

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**  
по выполнению лабораторных работ  
по дисциплине «Цифровые системы передачи»  
для студентов направления подготовки  
09.03.01 Информатика и вычислительная техника  
Направленность (профиль)  
"Программное обеспечение средств вычислительной  
техники и автоматизированных систем"

## СОДЕРЖАНИЕ

|                                                                                                               |            |
|---------------------------------------------------------------------------------------------------------------|------------|
| <b>ПРЕДИСЛОВИЕ .....</b>                                                                                      | <b>4</b>   |
| <b>ГЛАВА 1. КОДИРОВАНИЕ ИНФОРМАЦИИ .....</b>                                                                  | <b>7</b>   |
| 1.1. Дискретизация непрерывных сообщений по времени и уровню .....                                            | 8          |
| 1.2. Кодирование источника информации .....                                                                   | 11         |
| 1.3. Помехоустойчивое кодирование .....                                                                       | 18         |
| 1.4. Принципы создания систематических кодов .....                                                            | 30         |
| 1.5. Коды Хэмминга. ....                                                                                      | 34         |
| 1.6. Циклические коды.....                                                                                    | 38         |
| 1.7. Лабораторная работа 1. Исследование особенностей работы систем<br>кодирования информации .....           | 53         |
| <b>ГЛАВА 2. КОДИРОВАНИЕ В ЛИНИЯХ СВЯЗИ.....</b>                                                               | <b>57</b>  |
| 2.1. Униполярные коды.....                                                                                    | 57         |
| 2.2. Простейший биполярный код .....                                                                          | 58         |
| 2.3. Манчестерский код .....                                                                                  | 62         |
| 2.4. NRZI и другие коды .....                                                                                 | 66         |
| 2.5. Многоуровневые коды.....                                                                                 | 70         |
| 2.6. Кодирование скремблированием.....                                                                        | 72         |
| 2.7. Кодирование перемежением .....                                                                           | 80         |
| 2.8. Лабораторная работа 2. Исследование особенностей работы систем<br>линейного кодирования .....            | 84         |
| <b>ГЛАВА 3. МОДУЛЯТОРЫ И ДЕМОДУЛЯТОРЫ.....</b>                                                                | <b>88</b>  |
| 3.1. Амплитудная манипуляция .....                                                                            | 90         |
| 3.2. Частотная манипуляция.....                                                                               | 93         |
| 3.3. Фазовая манипуляция .....                                                                                | 95         |
| 3.4. Многопозиционная модуляция .....                                                                         | 97         |
| 3.5. Цифровая широтно-импульсная модуляция.....                                                               | 104        |
| 3.6. Лабораторная работа 3. Исследование особенностей работы<br>модулирующих и демоделирующих устройств ..... | 105        |
| <b>ГЛАВА 4. КАНАЛЫ ПЕРЕДАЧИ ИНФОРМАЦИИ.....</b>                                                               | <b>109</b> |
| 4.1. Типы каналов и линий связи .....                                                                         | 110        |
| 4.2. Каналы со случайными параметрами.....                                                                    | 121        |
| 4.3. Лабораторная работа 4. Исследование особенностей работы каналов<br>передачи информации.....              | 139        |
| <b>ГЛАВА 5. УПЛОТНЕНИЕ И РАЗДЕЛЕНИЕ КАНАЛОВ .....</b>                                                         | <b>143</b> |
| 5.1. Пространственное и дифференциальное разделение .....                                                     | 146        |
| 5.2. Частотное разделение .....                                                                               | 147        |
| 5.3. Временное разделение .....                                                                               | 149        |

|                                                                                                              |            |
|--------------------------------------------------------------------------------------------------------------|------------|
| 5.4. Фазовое разделение .....                                                                                | 153        |
| 5.5. Кодовое разделение .....                                                                                | 155        |
| 5.6. Разделение по уровню .....                                                                              | 167        |
| 5.7. Лабораторная работа 5. Исследование особенностей работы систем<br>уплотнения и разделения каналов ..... | 170        |
| <b>ГЛАВА 6. УСТРОЙСТВА ФИЛЬТРАЦИИ .....</b>                                                                  | <b>173</b> |
| 6.1. Создание IIR-фильтров .....                                                                             | 175        |
| 6.2. Создание FIR-фильтров .....                                                                             | 178        |
| 6.3. Адаптивные фильтры .....                                                                                | 181        |
| 6.4. Лабораторная работа 6. Исследование особенностей работы систем<br>фильтрации сигнала .....              | 185        |
| <b>ГЛАВА 7. УСТРОЙСТВА СИНХРОНИЗАЦИИ СИГНАЛА .....</b>                                                       | <b>188</b> |
| 7.1. Схемы реализации фазовой автоподстройки частоты .....                                                   | 188        |
| 7.2. Лабораторная работа 7. Исследование особенностей работы системы<br>фазовой автоподстройки частоты ..... | 197        |
| <b>ЛИТЕРАТУРА И ИСТОЧНИКИ .....</b>                                                                          | <b>199</b> |
| <b>ПРИЛОЖЕНИЕ 1. ИНСТРУМЕНТАЛЬНАЯ СРЕДА ВИЗУАЛЬНОГО<br/>МОДЕЛИРОВАНИЯ ДИНАМИЧЕСКИХ СИСТЕМ VISSIM .....</b>   | <b>201</b> |
| П1.1. Структура окна и система меню .....                                                                    | 201        |
| П1.2. Блоки контроля и управления .....                                                                      | 231        |
| П1.3. Библиотека компонентов Signal Producer .....                                                           | 252        |
| П1.4. Группа оформления и преобразования величин .....                                                       | 257        |
| П1.5. Арифметические функции и операции .....                                                                | 260        |
| П1.6. Математические и логические операции .....                                                             | 262        |
| П1.7. Каналы связи (Comm/Channels) .....                                                                     | 264        |
| П1.8. Операции с комплексными числами (Comm/Complex Math) .....                                              | 269        |
| П1.9. Демодуляторы (Comm/Demodulators) .....                                                                 | 270        |
| П1.10. Цифровые устройства (Comm/Digital) .....                                                              | 272        |
| П1.11. Кодеры/Декодеры (Comm/(Encode/Decode)) .....                                                          | 277        |
| П1.12. Измерительные преобразователи (Comm/Estimators) .....                                                 | 282        |
| П1.13. Фильтры (Comm/Filters) .....                                                                          | 289        |
| П1.14. Модуляторы типа Complex (Comm/Modulator-Complex) .....                                                | 297        |
| П1.15. Модуляторы типа Real (Comm/Modulator-Real) .....                                                      | 304        |
| П1.16. Элементы систем фазовой автоподстройки частоты (Comm/PLL) .....                                       | 312        |
| П1.17. Высокочастотные элементы (Comm/RF) .....                                                              | 314        |
| П1.18. Источники сигналов (Comm/Signal Sources) .....                                                        | 318        |
| П1.19. Блоки импортирования данных (Blocks/DDE) .....                                                        | 324        |
| П1.20. Интеграторы (Blocks/Integration) .....                                                                | 325        |
| П1.21. Линейные блоки (Blocks/Linear system) .....                                                           | 326        |

|                                                                                                                  |            |
|------------------------------------------------------------------------------------------------------------------|------------|
| П1.22. Матричные операции (Blocks/Matrix Operations).....                                                        | 328        |
| П1.23. Нелинейные блоки (Blocks/Nonlinear) .....                                                                 | 331        |
| П1.24. Блоки оптимизации (Blocks/Optimization) .....                                                             | 333        |
| П1.25. Генераторы шума (Blocks/Random Generator).....                                                            | 335        |
| П1.26. Блоки синхронизации (Blocks/Real Time).....                                                               | 337        |
| П1.27. Блоки задержки (Blocks/TimeDelay).....                                                                    | 337        |
| П1.28. Трансцендентные функции (Blocks/Transcendental) .....                                                     | 338        |
| П1.29. Другие компоненты меню Blocks .....                                                                       | 339        |
| П1.30. Блоки согласования (Comm/Multirate Support).....                                                          | 339        |
| П1.31. Турбокодеры (Comm/Turbo Codes).....                                                                       | 340        |
| П1.32. Векторные операторы (Comm/Vector Ops) .....                                                               | 343        |
| П1.33. Беспроводные компоненты спецификации 802.11<br>(Wireless/802.11) .....                                    | 345        |
| П1.34. Беспроводные компоненты спецификации 802.11a/g<br>(Wireless/802.11a/g) .....                              | 349        |
| П1.35. Беспроводные компоненты спецификации 802.11b<br>(Wireless/802.11b) .....                                  | 354        |
| П1.36. Беспроводные компоненты ближней связи (Wireless/Bluetooth) .....                                          | 356        |
| П1.37. Универсальные беспроводные компоненты (Wireless/Generic) .....                                            | 359        |
| П1.38. Беспроводные компоненты сверхширокополосной связи<br>(Wireless/UltraWideband).....                        | 359        |
| <b>ПРИЛОЖЕНИЕ 2. СРЕДА КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ<br/>И АНАЛИЗА СХЕМ ЭЛЕКТРОННЫХ УСТРОЙСТВ NI MULTISIM.....</b> |            |
| П2.1. Общие сведения о среде NI Multisim .....                                                                   | 361        |
| П2.2. Запуск программы Multisim 10 .....                                                                         | 363        |
| П2.3. Пользовательский интерфейс среды Multisim 10 .....                                                         | 363        |
| П2.4. Командные строки инструментальной линейки .....                                                            | 370        |
| П2.5. Измерительные приборы, источники питания и устройства<br>визуализации.....                                 | 374        |
| П2.6. Сборка схем электронных устройств .....                                                                    | 376        |
| П2.7. Установка параметров компонентов .....                                                                     | 378        |
| <b>ПРИЛОЖЕНИЕ 3. СОДЕРЖАНИЕ, ФОРМА И ПРАВИЛА<br/>ОФОРМЛЕНИЯ ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ .....</b>              |            |
|                                                                                                                  | <b>383</b> |

## ПРЕДИСЛОВИЕ

В настоящее время системы передачи дискретной – цифровой информации играют огромную роль в современной жизни и являются основой современной техники связи. Изучению вопросов передачи дискретных сообщений и предназначено данное учебное пособие.

Разработка любого радиоэлектронного устройства включает физическое или математическое моделирование. Физическое моделирование связано с большими материальными затратами, так как требует изготовления макетов и их трудоемкого исследования. Иногда чисто физическое моделирование просто невозможно из-за сложности устройства. В этом случае прибегают к математическому моделированию с использованием средств и методов вычислительной техники.

Наиболее простыми и легко осваиваемыми программами, позволяющими проводить моделирование элементов системы передачи дискретных сигналов являются программы VisSim разработанная и развиваемая компанией Visual Solutions (США) и Multisim, разработанная фирмой Electronics Workbench, ныне являющейся подразделением компании National Instruments (США).

Среда NI Multisim позволяет объединить процессы разработки электронных устройств и тестирования на основе технологии виртуальных приборов для учебных и производственных целей. База данных компонентов включает более 1200 SPICE-моделей элементов от ведущих производителей, таких как Analog Devices, Linear Technology и Texas Instruments, а также более 100 моделей импульсных источников питания.

Среда VisSim предназначена для построения, исследования и оптимизации виртуальных моделей физических и технических объектов, в том числе и телекоммуникационных систем. VisSim – это аббревиатура выражения Visual Simulator – визуальная, воспринимаемая зрением, среда и средство моделирования.

Программа представляет собой мощное, удобное в использовании, компактное и эффективное средство моделирования физических и технических объектов, систем и их элементов. Программа предоставляет разработчику развитый графический интерфейс, используя который он создает модель из виртуальных элементов с некоторой степенью условности так же, как если бы он строил реальную систему из настоящих элементов. Это позволяет создавать, а затем исследовать и оптимизировать модели систем широкого диапазона сложности.

Данное учебное издание состоит из семи глав, содержание которых позволяет достаточно подробно рассмотреть основополагающие принципы передачи дискретных сообщений.

В главе 1 изложены методы кодирования источника информации, а также основные подходы к помехоустойчивому кодированию. Глава 2 посвящена знакомству с особенностями построения систем линейного кодирования. Глава 3 содержит сведения о амплитудном, частотном, фазовом, многопозиционном и широтноимпульсном методах цифровой модуляции и демодуляции сигнала. В главе 4 рассмотрены особенности функционирования каналов и линий связи различного типа. В главе 5 изложены пространственный, дифференциальный, частотный, временной, фазовый, кодовый методы уплотнения и разделения каналов, а также метод разделения по уровню. Глава 6 посвящена анализу особенностей построения систем фильтрации сигналов. Глава 7 содержит сведения о методах синхронизации сигналов, в частности рассматриваются особенности функционирования системы фазовой автоподстройки частоты.

Предлагаемый материал включает теоретическую часть, достаточную для выполнения лабораторного практикума. Каждая из глав книги завершается лабораторной работой, выполнение которой позволяет углубить понимание процессов, происходящих при передаче дискретных сообщений.

Для облегчения работы со средами моделирования в приложениях приведены краткие описания программных комплексов VisSim 7.0 и Multisim 10.0.

При разработке данного учебного издания использовались материалы из книги Карлащук В. И. Электронная лаборатория на IBM PC. Лабораторный практикум на Electronics Workbench и VisSim по элементам телекоммуникационных систем. – М.: СОЛОН–ПРЕСС, 2005, 480 с.

Характерной особенностью данного учебного издания является реализация моделей элементов системы передачи дискретных сообщений с применением современных сред моделирования Multisim 10.0 компании National Instruments [18], VisSim 7.0 корпорации Visual Solutions [22], а также схем телекоммуникационных устройств, разрабатываемых на базе компонентов VisSim/Comm компанией Eritek Inc. [17].

## ГЛАВА 1. КОДИРОВАНИЕ ИНФОРМАЦИИ

В общем случае существует четыре типа кодирования [9].

**1. Кодирование источника информации** состоит в дискретизации по времени и уровню в случае непрерывных сообщений и при максимально возможном сжатии информации до дискретизации или после нее в целях экономии ресурсов каналов связи или памяти. Получаемый при этом код называется **первичным** или **натуральным**. Например, «чистая» дискретизация по времени и уровню используется в цифроаналоговых преобразователях (ЦАП) цифровых измерительных приборов с предварительным сжатием аналогового сигнала – в цифровой телефонии, со сжатием цифрового сигнала – в архиваторах ZIP, RAR и др.

**2. Кодирование канала** заключается в расширении кодового алфавита, позволяющем распознавать и исправлять ошибки после передачи через канал связи. В этом случае число исходных комбинаций первичного кода меньше числа закодированных. Такое кодирование в общем случае известно как помехоустойчивое, реализуемое корректирующими кодами. В сетевых технологиях оно известно как кодирование канала, кодирование на уровне канала или канальное кодирование.

**3. Кодирование линии связи** заключается в таком преобразовании закодированного сообщения, которое обеспечивало бы возможность надежной синхронизации и минимум искажений при его трансляции через линию связи (среду передачи) при асинхронной и синхронной передаче. В этом случае число исходных комбинаций, полученных при выполнении операций по п. 1 или 2, равно числу закодированных. Этот вид кодирования в сетевых технологиях известен как кодирование на физическом уровне.

**4. Криптографическое кодирование** состоит в засекречивании данных в целях исключения несанкционированного доступа к информации в процессе ее хранения и передачи.

Далее мы будем рассматривать процессы кодирования по п. 1 – 3.



### 1.1. Дискретизация непрерывных сообщений по времени и уровню

Дискретизация непрерывных сообщений производится при их преобразовании в цифровую форму (так называемая оцифровка) [9]. Выбор максимального (предельного) интервала дискретизации  $\Delta t$  определяется возможностью восстановления исходной непрерывной функции времени по ее дискретным отсчетам с заданной точностью. Большое внимание к этому параметру объясняется тем, что оно лежит в основе анализа ряда важнейших вопросов преобразования и передачи сообщений, в частности при исследовании вопросов устранения избыточности сообщений («сжатие информации»), пропускной способности каналов и т. п.

При определении  $\Delta t$  используется теорема Котельникова, предложенная и доказанная им в 1933 году применительно к конкретным проблемам передачи сообщений. Впервые теорема была сформулирована без доказательства англичанином Найквистом в 1928 году и в 1948-м использована под названием теоремы отсчетов (с доказательством) Шенноном при разработке математической теории связи. Напомним, что эта теорема формулируется следующим образом: если непрерывная функция времени  $x(t)$  имеет спектр, ограниченный полосой частот от нуля до  $F$  (например, синусоида частотой  $F$  с постоянной составляющей), то она полностью определяется последовательностью своих мгновенных значений, взятых в моменты времени, отсчитываемые через интервалы  $\Delta t = 1/2F$ .

Точное восстановление функции  $x(t)$  по ее дискретным отсчетам, по Котельникову, возможно только при бесконечной протяженности этой функции во времени. В действительности же все реальные сообщения имеют ограниченную продолжительность. Поэтому в реальных устройствах используется более высокая частота дискретизации. При этом, в зависимости от используемых при восстановлении (после передачи) сигнала методов интерполяции, интервал дискретизации определяется приближенными формулами:

- при ступенчатой интерполяции:  $\Delta t_c = 0,55\delta\Delta t$ ;
- при линейной:  $\Delta t_c = 1,35(\delta)^{1/2} \Delta t$ ;
- при параболической:  $\Delta t_c = (\delta)^{1/3} \Delta t$ ;
- при интерполяции  $k$ -го порядка  $\Delta t_c = (\delta)^{1/(k+1)} \Delta t$ ,

где  $\delta = \Delta x / x_{\max}$  – допустимая относительная погрешность равномерного приближения (близка к величине среднеквадратичной ошибки дискретизации);  $\Delta x$  – отклонение восстановленного сигнала от исходного;  $x_{\max}$  – максимальное значение интервала изменения исходного сигнала (полная шкала изменения мгновенных значений).

Из приведенных формул видно, что избыточность отсчетов зависит от требуемой точности восстановления исходного сообщения и вида применяемой при восстановлении интерполяции.

Заметим, что в истории электросвязи одной из первых причин дискретизации сигнала во времени была необходимость уплотнения во времени сигналов от различных источников. Исторически как в телеграфии, так и в телефонии многократные системы передачи с временным разделением каналов (ВД) появились раньше, чем системы с частотным разделением каналов. Телеграфная многократная система передачи с ВД была предложена еще в 1853 году малоизвестным американским изобретателем М. Б. Фармером. На прочную практическую основу метод ВД был поставлен Бодо почти 22 года спустя.

Дискретизация по времени является по существу операцией перемножения двух сигналов (дискретизируемого и импульсного). Для иллюстрации этого процесса в среде VisSim [20] разработана модель, представленная на рис. 1.1. На входы перемножителя Multiply подаются синусоидальный (дискретизируемый) сигнал с блока Sinusoid и импульсная последовательность с блока PulseTrain; результат представлен на дисплее плоттера Discrete time.

Любое непрерывное сообщение можно рассматривать как одну из реализаций случайного процесса, мгновенные значения которого находятся в

пределах некоторого диапазона, называемого **непрерывной шкалой мгновенных значений сообщения** и определяемого физическими ограничениями, характерными для реальных сообщений. Из-за влияния дестабилизирующих факторов (шумов, помех, аппаратной нестабильности и конечной разрешающей способности приемников) передача сообщений всегда сопровождается ошибкой. Это означает, что достаточно близкие по форме сообщения различить при приеме невозможно и, следовательно, их можно передавать в **квантованном** по амплитуде (уровню) виде с некоторой ошибкой  $\Delta x$ , равной **шагу квантования**.

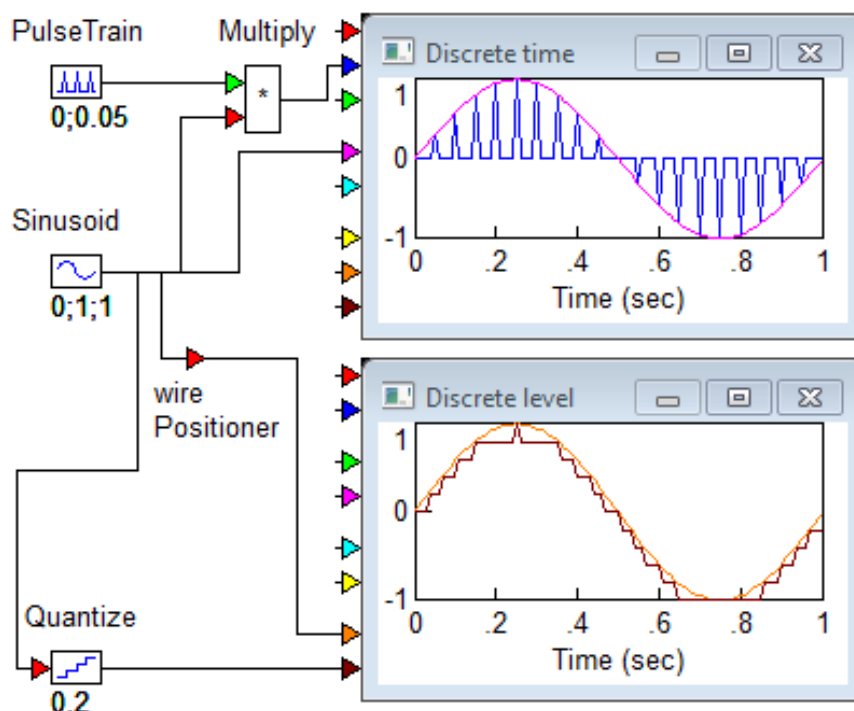


Рисунок 1.1. Дискретизация синусоидального сигнала по времени и уровню

В результате квантования непрерывное сообщение заменяется, сообщением ступенчатой формы. В качестве примера на дисплее плоттера Discrete level (рис. 1.1) показан процесс квантования синусоиды амплитудой 1 В с шагом квантования 0,2 В, задаваемым в диалоговом окне квантователя Quantize. Блок позиционирования соединений wirePositioner на рис. 1.1 служит для фиксации в показанном положении проводника (иначе он «спрячется» за плоттер).

В процессе квантования возникает ошибка квантования  $\Delta x/2$  (максимальное значение), которую из-за ее случайного характера часто называют шумом квантования и обычно оценивают величиной среднеквадратичной ошибки  $\sigma = \Delta x/2(3)^{1/2}$  (для наиболее часто используемого равномерного квантования).

## 1.2. Кодирование источника информации

Такое кодирование заключается в преобразовании дискретизированных по времени и по уровню сообщений в код (от французского code, которое произошло от латинского codex – кодекс – свод законов, система правил) [9]. Для этого используются разнообразные АЦП, на выходе которых для каждого дискретного значения информационного сигнала формируется **первичная кодовая комбинация**, число символов в которой определяет разрядность кода. Совокупность кодовых комбинаций для представления заданного количества дискретных сообщений называется **кодовым словарем** или **алфавитом кода**.

Если при кодировании используется система счисления с основанием  $m$  и числом разрядов  $n$ , то количество символов в одной комбинации составит  $Q = m \times n$ . Если все кодовые комбинации содержат одинаковое количество элементов, то код называется равномерным, при различном количестве – неравномерным. Примером неравномерного кода является телеграфный код С. Морзе (1837 год), в котором часто используемым буквам присваиваются короткие кодовые комбинации, а редко используемым – длинные. В равномерном коде длина всех комбинаций одинакова и, следовательно, одинакова их разрядность. Очевидно, что равномерные коды при передаче маловероятных дискретных сообщений менее эффективны, чем неравномерные. Однако техническая реализация равномерных кодов значительно проще, чем неравномерных, поэтому они получили наибольшее распространение.

Если имеется  $N$  различных элементов дискретных сообщений (чисел, букв, команд и т. п.), которым нужно сопоставить кодовый словарь равномерного кода с основанием  $m$  и разрядностью  $n$ , то количество кодовых комбинаций такого кода должно удовлетворять условию  $N_k = m^n \geq N$ . Если  $N_k = N$ , то все комбинации будут использованы. Однако такой случай довольно редок, так как  $N$  обычно не кратно степеням выбранной системы счисления. Поэтому, как правило,  $N_k > N$  и, следовательно, некоторая часть кодовых комбинаций не будет использована и окажется лишней (избыточной). Код, для которого выполняется условие

$$m^{n-1} < N \leq m^n, \quad (1.1)$$

называется простым, или **первичным**. Характерной особенностью такого кода является то, что в нем нельзя уменьшить разрядность кодовых комбинаций, поскольку при ее уменьшении на единицу число комбинаций становится меньше числа дискретных сообщений. Из (1.1) следует соотношение  $1/m < N/N_k \leq 1$ , что для равномерных первичных двоичных кодов соответствует условию  $0,5 < N/N_k \leq 1$ , а это означает, что в первичном двоичном коде неиспользуемая часть кодовых комбинаций всегда меньше половины общего числа кодируемых дискретных величин. Если первичный код применяется для передачи аналоговых сообщений в цифровой форме, то, как правило, используются все кодовые комбинации, что автоматически обеспечивается, например, при работе АЦП в составе цифровых измерителей электрических величин.

Все первичные коды, в зависимости от того, присваивается или не присваивается каждому разряду кодовой комбинации некоторый весовой коэффициент, разделяются на **взвешенные** и **невзвешенные**. Так как каждый разряд в кодовой комбинации занимает определенную позицию, взвешенные коды иногда называют позиционными, а невзвешенные — непозиционными. Наиболее распространенным видом взвешенных равномерных первичных

кодов является естественный или натуральный, в котором веса разрядов определяются последовательными степенями числа 2:  $2^{n-1}, 2^{n-2}, \dots, 2^0$ .

Наряду с достоинствами (простота реализации и удобство выполнения вычислений) натуральному коду присущ существенный недостаток, связанный с явлением, получившим название «неоднозначность отсчета» или «неопределенность границы считывания». Суть явления состоит в том, что при считывании сформированной кодовой комбинации в кодирующем устройстве может возникать ложная комбинация, которая существенно отличается от истинной. Например, при переходе от числа 7, которому соответствует комбинация 0111, к числу 8 (1000) происходит изменение состояний во всех разрядах (позициях) по сравнению с предыдущими. В этом случае из-за нестабильности синхронизирующих устройств возможно считывание некоторых промежуточных состояний. Такое положение может возникать при первичном кодировании, основанном, например, на счетно-импульсном методе, или при кодировании по методу считывания. Аналогичное положение возможно также при некоторых преобразованиях дискретных сообщений, когда требуется запоминать кодовые комбинации на некоторое время, а затем выводить (считывать) их из устройств памяти: если вывод информации начинается раньше, чем успевают установиться все состояния ячеек памяти, может быть считана любая из промежуточных комбинаций.

Натуральному коду присущ еще один недостаток: если сбой произошел в старшем разряде, то ошибка в передаче числа значительно больше, чем при неверном приеме элемента младшего разряда. С этой точки зрения невзвешенный код обладает преимуществом, поскольку для него подобные ошибки не зависят от разряда.

При отсутствии статистической взаимосвязи между элементами (буквами) сообщения конструктивные методы построения **эффективных кодов** были даны впервые Шенноном, Фэно и Хаффменом, однако такие коды вызывают затруднения при декодировании. Этого недостатка в некоторой степени лишены рефлексные коды (от латинского *reflexio* – отражение). Характерной

особенностью таких невзвешенных кодов является то, что любые две соседние  $n$ -значные кодовые комбинации отличаются друг от друга лишь в одной позиции. Наиболее часто применяемым является рефлексный код Грея (1878 год), который получается из натурального кода путем суммирования его двоичной комбинации по модулю 2 с той же комбинацией, сдвинутой на один разряд вправо, и последующим отбрасыванием младшего разряда сдвинутой комбинации. Например, для цифры 7 (0111) в коде Грея будем иметь:  $0111 \oplus 00111 = 0100$ .

Если составить таблицу натурального кода и кода Грея для десятичных чисел 0 – 15, то можно отметить следующие его особенности.

1. Каждая последующая комбинация всегда отличается от предыдущей только в одной позиции (в одном разряде).

2. Смена значений элементов в каждом разряде (1 на 0 или 0 на 1) при переходе от комбинации к комбинации в коде Грея происходит вдвое реже, чем в натуральном. Так, в натуральном коде смена элемента первого (младшего) разряда происходит с чередованием элементов 0 – 1 – 0 – 1 ..., второго – с чередованием 00 – 11 – 00 – 11 ..., третьего – с чередованием 0000 – 1111 – 0000 ... В коде Грея соответственно имеем: для первого разряда 11 – 00 – 11 ..., для второго 0000 – 1111 – 0000 ... Это свойство кода Грея позволяет при том же быстродействии схемы кодирования получать точность кодирования выше по сравнению с натуральным кодом.

3. При сложении двух соседних комбинаций кода Грея по модулю 2 ( $\text{mod } 2$ ) число единиц равно числу разрядов минус три, т. е. в нашем случае имеем одну единицу. В общем случае, для  $n$ -значного кода число единиц равно  $n - 3$ . Это свойство кода Грея иногда можно использовать для проверки правильности принятых комбинаций.

4. В коде Грея можно выделить оси симметрии (оси «отражения»), относительно которых наблюдается идентичность элементов в некоторых разрядах. Так, например, имеет место симметрия относительно оси, проведенной между числами 7 и 8. В комбинациях, симметричных

относительно этой оси, идентичны три символа младших разрядов, относительно осей 1 – 2, 5 – 6, 9 – 10, 11 – 12 имеется симметрия в одном символе младшего разряда, а относительно осей 3 – 4 и 11 – 12 – в двух символах. Эта особенность и послужила основанием для использования термина «рефлексный» (или отраженный) код.

По сравнению с натуральным, рефлексный код Грея позволяет уменьшить ошибки неоднозначности считывания, а также ошибки, вызванные помехами при передаче информации по каналам связи. Если, например, неправильное воспроизведение символа старшего разряда комбинации натурального кода приводит к значительной ошибке, то для кода Грея эта ошибка составляет величину одного младшего разряда.

Схема преобразователя 4-разрядного натурального кода в код Грея реализованного в среде NI Multisim [19] представлена на рис. 1.2 состоит из генератора слова XWG1 с индикаторами его состояния на логических пробниках N0 – N3 (отражают входной натуральный код), регистра сдвига на D-триггерах T0 – T3 с индикаторами их состояния C0 – C3, логических элементов ИСКЛЮЧАЮЩЕЕ ИЛИ (EOR) U3 – U6, ИЛИ (OR) U0 – U2, И (AND) U7 – U10, на выходе которых включены индикаторы кода Грея G0 – G3. Схема И (AND) Uс служит для формирования тактовых импульсов сдвига, поступающих на V-входы триггеров. На входы схемы Uс подаются логические сигналы 1 с выхода R (готовность данных) генератора слова и сигналы с выхода 15.

Кодовые комбинации в натуральном коде N0 – N3 задаются на дисплее генератора в ячейках, соответствующих выходам 0 – 3. Для вывода двоичной комбинации в коде Грея на индикаторы G0 – G3 используется логический сигнал с восьмого выхода генератора.

В кодовой комбинации № 0 (см. рис. 1.3) производится очистка регистра от предыдущей комбинации 14 входного кода путем сдвига вправо. При комбинации № 1, получаемой после нажатия кнопки STEP, в регистре



запоминается натуральный код 0001, при комбинации № 2 он сдвигается вправо на один разряд, т. е. становится равен 0000.

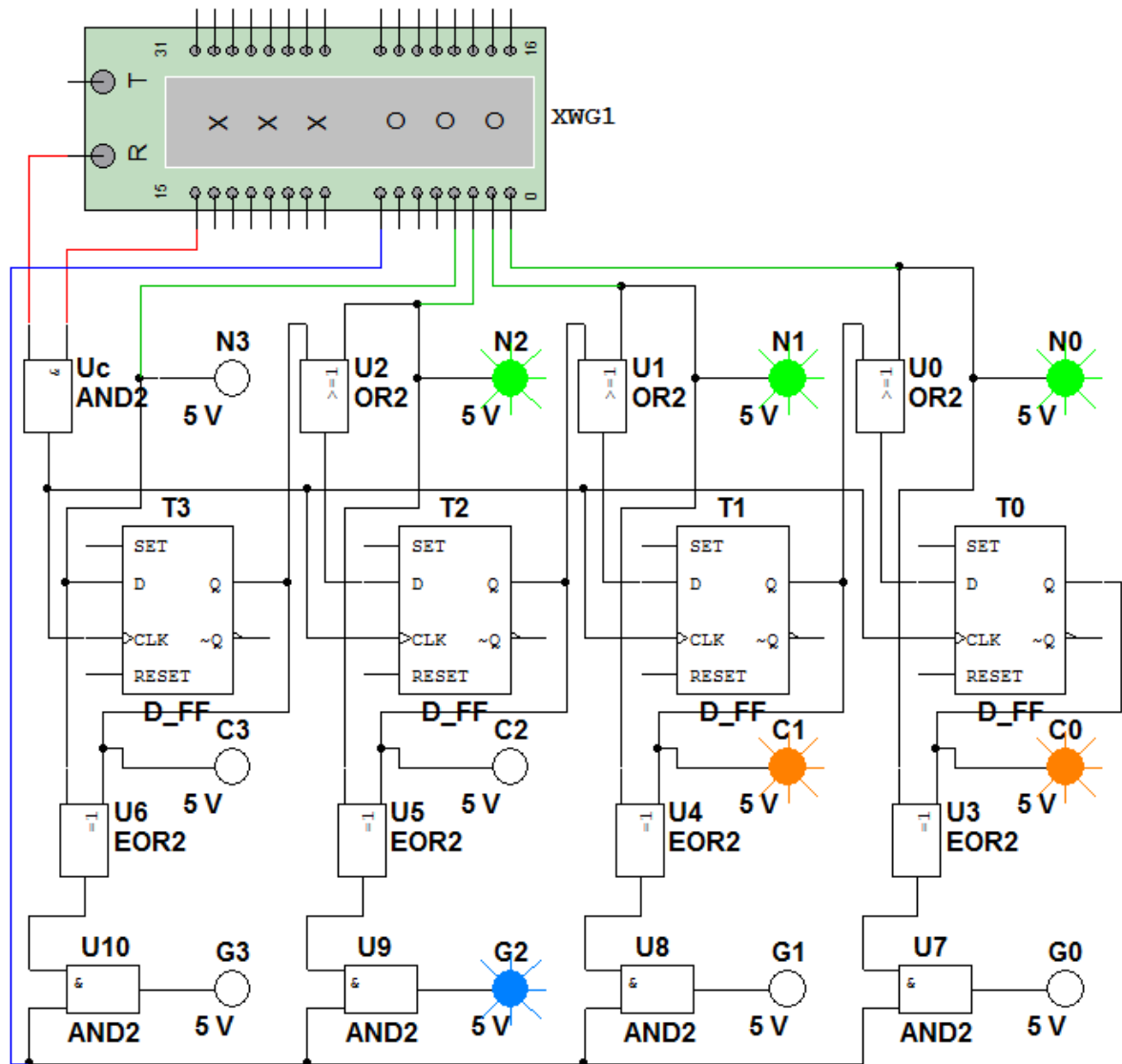


Рисунок 1.2. Преобразователь натурального кода в код Грея

При комбинации № 3 опять выдается код 0001, который с помощью элементов U3 – U6 суммируется по модулю 2 с запомненным на регистре при комбинации № 2 (выполняется операция  $0001 \oplus 0000 = 0001$ ). Поскольку при комбинации № 3 присутствует логическая единица в восьмом разряде, то полученный код Грея через U7 – U10 выводится на индикаторы G0 – G3.

При комбинациях 4 – 6 производится очистка регистра (три сдвига вправо). При комбинации № 7 вводится натуральный код 0010, при комбинации № 8 он сдвигается на один разряд вправо (код 0001), при комбинации № 9

выполняется операция  $0010 \oplus 0001 = 0011$  с выводом результата на индикаторы G0 – G3. Аналогичным образом производится преобразование натуральных кодов: 0011 в код Грея 0010; 0100 в код Грея 0110; 0101 в код Грея 0111; 0110 в код Грея 0101; 0111 в код Грея 0100.

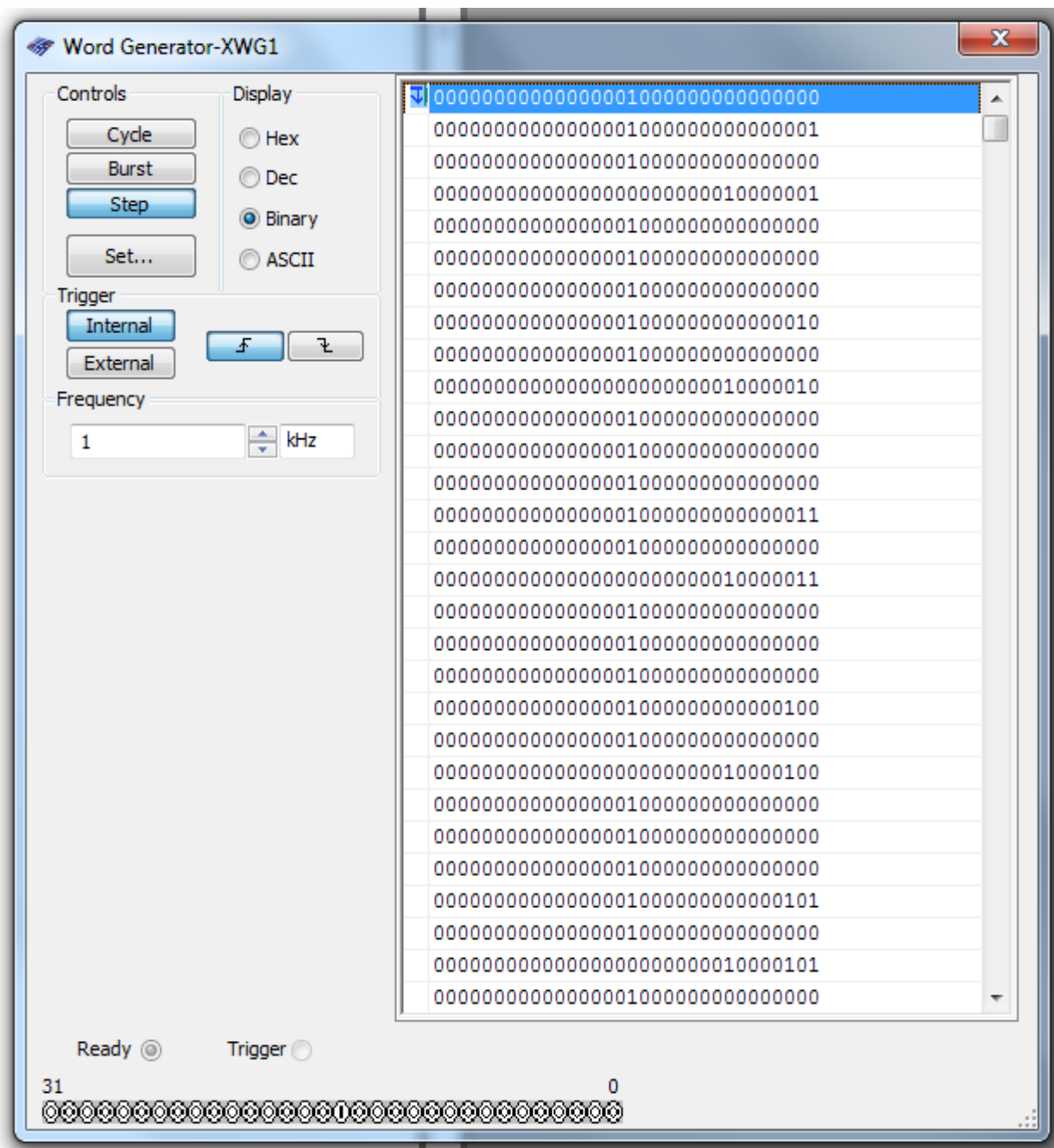


Рисунок 1.3. Окно генератора слова в преобразователе натурального кода в код Грея

Недостатком кода Грея и других рефлексных кодов являются трудности обработки данных с помощью ЭВМ и существенное усложнение

декодирующих устройств, что приводит к необходимости, предварительного преобразования рефлексных кодов в натуральный.

В случае кода Грея преобразование его в натуральный код производится по следующим правилам:

- преобразование начинается со старшего разряда кода Грея, который переносится в старший разряд натурального кода без изменения;
- каждый следующий символ натурального кода находится путем суммы по модулю 2 текущего символа кода Грея и символа натурального кода полученного на предыдущем шаге.

Кроме рассмотренных видов первичных кодов, в практике кодирования и отображения информации используется **двоично-десятичный** код с представлением каждой десятичной цифры 4-разрядной комбинацией натурального двоичного кода (так называемой **тетрадой**). Например, число 1975 в двоично-десятичном коде представляет последовательность тетрад в виде 0001 1001 0111 0101. Кодирование десятичных чисел тетрадами уступает по экономичности кодированию натуральным двоичным кодом. Если десятичное число содержит  $n$  разрядов, то при двоично-десятичном кодировании потребуется  $4n$  двоичных символов, а при натуральном двоичном кодировании –  $n \times \log_2 10 \approx 3,3n$  двоичных символов. Кроме того, при двоично-десятичном кодировании необходимо вводить разграничительные интервалы между тетрадами, что приводит к уменьшению быстродействия аппаратуры и скорости передачи информации.

### 1.3. Помехоустойчивое кодирование

Сигналы, непосредственно передаваемые по последовательным линиям (типа витой пары, коаксиального кабеля или телефонной линии), подвержены влиянию ряда факторов, воздействие которых может привести к возникновению ошибок в принятой информации [9]. Ошибки могут возникать вследствие влияния на канал связи наводок и помех естественного или

искусственного происхождения, а также вследствие изменения конфигурации системы передачи информации с временным нарушением или без нарушения целостности канала связи (например, в случае подключения новых абонентов к существующей локальной информационной сети).

Стандартные протоколы обмена информации в сетях предусматривают введение обязательного поля для размещения помехоустойчивого (корректирующего) кода. Если в результате обработки принятого пакета обнаружится несоответствие принятого и вновь вычисленного помехоустойчивого кода, с большой долей вероятности можно утверждать, что среди принятых бит имеются ошибочные. Передачу такого пакета нужно будет повторить (в расчете на случайный характер помех).

Имеется разрыв между требованиями к верности принимаемой информации и возможностями каналов связи. В частности, стандартами международных организаций ИТУ-Т и МСС установлено, что вероятность ошибки при телеграфной связи не должна превышать  $3 \times 10^{-5}$  (на знак), а при передаче данных –  $10^{-6}$  (на единичный элемент, бит). На практике допустимая вероятность ошибки при передаче данных может быть еще меньше –  $10^{-9}$ . В то же время каналы связи (особенно проводные каналы большой протяженности и радиоканалы) обеспечивают вероятность ошибки на уровне  $10^{-3} - 10^{-4}$  даже при использовании фазовых корректоров, регенеративных ретрансляторов и других устройств, улучшающих качество каналов связи.

Кардинальным способом снижения вероятности ошибок при приеме является введение избыточности в передаваемую информацию. В системах передачи информации без обратной связи данный способ реализуется в виде помехоустойчивого кодирования, многократной передачи информации или одновременной передачи информации по нескольким параллельно работающим каналам.

Помехоустойчивое кодирование доступнее, при прочих равных условиях позволяет обойтись меньшей избыточностью и за счет этого повысить скорость передачи информации. Этот вид кодирования является эффективным средством

повышения достоверности передачи сообщений на основе применения специальных кодов, которые обнаруживают и корректируют ошибки, вызванные действием помех. Код для обнаружения ошибок передачи называется **обнаруживающим**; исправление ошибок в этом случае обычно производится путем повторения искаженных сообщений по запросу, передаваемому по каналу обратной связи. Код называется **корректирующим**, если он позволяет не только обнаруживать, но и исправлять часть или все ошибки при приеме сообщений. Характерной особенностью таких кодов является высокая степень избыточности.

Все известные в настоящее время коды рассматриваемого класса могут быть разделены на **блочные** и **непрерывные**. Блочные коды характеризуются тем, что последовательность передаваемых символов разделена на блоки, в которых операции кодирования и декодирования производятся отдельно. Отличительной особенностью непрерывных кодов является то, что первичная информационная последовательность символов непрерывно преобразуется по определенному закону в другую последовательность, в которой проверочные символы перемежаются с информационными. Поскольку формирование проверочных символов в таких кодах ведется по рекуррентным правилам, то их часто называют **цепными** или **рекуррентными**.

Блочные и непрерывные коды могут быть **разделимыми** и **неразделимыми**. В разделимых кодах всегда можно выделить информационные и контрольные (проверочные) символы, которые являются избыточными и служат исключительно для коррекции ошибок. В неразделимых кодах такое разделение символов провести невозможно.

Наиболее многочисленный класс разделимых кодов составляют **линейные**. Основная их особенность состоит в том, что контрольные символы образуются как линейные комбинации информационных. В свою очередь, линейные коды могут быть разбиты на два подкласса: **систематические** и **несистематические**. Все двоичные систематические коды являются **групповыми** и характеризуются принадлежностью кодовых комбинаций к

группе, обладающей следующим свойством: сумма по модулю два любой пары комбинаций снова дает комбинацию, принадлежащую этой группе. Несистематические коды такими свойствами не обладают.

Среди систематических кодов наиболее известны **циклические**. К таким кодам относятся коды Хэмминга, которые исторически появились первыми, а также коды Боуза-Чоудхури-Хоквингэма (БЧХ), Рида-Соломона и др. Особенность циклических кодов заключается в том, что циклический сдвиг любой разрешенной кодовой комбинаций также является разрешенной комбинацией. Циклические коды обладают хорошими корректирующими свойствами, а реализация кодирующих и декодирующих устройств для таких кодов оказывается проще, чем для других систематических кодов.

Несмотря на то что история кодирования насчитывает несколько веков, в технике связи она свое начало берет с 1948 года, когда была опубликована работа Клода Шеннона, доказавшего принципиальную возможность безошибочной передачи сигналов при соответствующем представлении носителей информации, т. е. путем применения специальным образом построенных кодов. До этого времени считалось само собой разумеющимся, что единственной возможностью повышения помехоустойчивости приема сигналов является увеличение мощности передатчика или многократная передача по каналу связи одного и того же сообщения. Итак, начиная с середины XX века, многочисленные исследования были направлены на разработку корректирующих кодов, алгоритмов их декодирования и согласования с видами модуляции и характеристиками каналов связи.

Первые блочные коды, позволявшие корректировать одиночные ошибки, были разработаны в 1950-х годах М. Дж. Голеем и Р. Хэммингом (США). При блочном кодировании последовательность информационных символов разбивается на отдельные блоки определенной длины. Для каждого такого блока формируются дополнительные проверочные символы, которые образуются путем сложения по модулю 2 определенных информационных символов. Блоки информационных и проверочных символов передаются в

канал связи. Коды Хэмминга имели простой алгоритм декодирования и позволяли корректировать одну ошибку в кодовой комбинации. Однако такие коды были достаточно далеки от тех требований, которые вытекали из теории Шеннона. Несмотря на это, разработки Голея и Хэмминга дали толчок для огромного числа последующих работ в области алгебраической теории кодирования. Так, уже в 1954 году И. С. Ридом (США) было предложено использовать мажоритарное декодирование, при котором для каждого информационного символа формируется нечетное число оценок путем сложения по модулю 2 определенных комбинаций символов принятого кода. Решение об истинном значении принятого символа принимается по мажоритарному принципу: если большее количество оценок равно 1, то принимается именно такое решение. В 1963 году Дж. Л. Месси установил общие принципы построения и декодирования подобных кодов. Достоинством мажоритарно декодируемых кодов являются чрезвычайная простота и быстрдействие алгоритмов декодирования. Значительный вклад в создание теории построения мажоритарно декодируемых кодов внесли в 1965 году советские ученые В. Д. Колесников и Е. Т. Мирончиков.

Весьма интересный класс блочных кодов был предложен в 1954 году американским ученым Г. Форни. Это так называемые **каскадные** коды, которые формируются следующим образом: последовательность информационных символов длиной  $n = n_1 \times n_2$  записывается в буферную память в виде таблицы, имеющей  $n_1$  столбцов и  $n_2$  строк. Символы отдельных строк и столбцов кодируются с помощью корректирующих кодов и в виде дополнительных проверочных символов вместе с информационными передаются по каналу связи. Значительным вкладом в усовершенствование таких кодов были работы советских ученых Э. Л. Блоха и В. В. Зяблова (1976 – 1982 гг.), а также Д. Слепяна (США), который был первым, кто в 1956 году заложил фундамент теории линейных блочных кодов с проверкой на четность (математическая теория групп).

В 1955 году в США (П. Элайес) и СССР (Л. М. Финк и В. И. Шляпоберский) был предложен весьма важный класс **сверточных** или **рекуррентных** кодов, нашедший широкое применение в современной технике связи. Исследования, связанные с построением таких кодов и разработкой эффективных с вычислительной точки зрения алгоритмов их декодирования, заняли почти 20 лет. В этом классе кодов информационная последовательность символов разбивается на блоки, содержащие по  $m$  символов, которые поступают на линейный преобразователь, имеющий память на  $K$  подобных блоков ( $K$  – длина кодового ограничения). В этом преобразователе каждый блок из  $m$  поступивших символов с учетом содержащихся в памяти  $K$ -блоков преобразуются в  $n$  ( $n > m$ ) символов, передаваемых по каналу связи. При этом относительная скорость передачи информации составляет  $R = m/n$ .

Несмотря на то, что сверточные коды являются частным случаем блочных линейных кодов, введение сверточной структуры наделяет их рядом дополнительных свойств, которые существенно облегчают декодирование. Эти коды имеют древовидную или решетчатую структуру. Каждому ребру древовидной структуры соответствует определенная последовательность  $m$  информационных символов. По принятой последовательности символов для каждого ребра может быть найден его вес (метрика) – число, характеризующее его расстояние от принятой последовательности. Для измерения этого расстояния может быть использована метрика Хэмминга, если в демодуляторе принимается жесткое решение, или евклидова метрика, если декодирование осуществляется по методу максимума правдоподобия. Декодирование сверточных кодов состоит в прослеживании по кодовой решетке того пути, для которого расстояние от принятой последовательности символов имеет минимальное значение. При этом сверточная структура кода позволяет использовать рекуррентные алгоритмы, существенно упрощающие вычисления этого расстояния.

Для декодирования сверточных кодов Дж. Возенкрафтом (США) в 1957 году был предложен алгоритм последовательного декодирования, в



соответствии с которым в декодере просматриваются не все возможные пути по ребрам кодовой решетки сверточного кода, а наиболее вероятные. Если декодер выбрал на каком-то шаге неверный путь, то он вскоре обнаруживает, что при последующих выборах ребер происходит быстрое увеличение расстояния между выбранным путем и принимаемой последовательностью. Это является сигналом к тому, чтобы декодер сделал несколько шагов назад и начал исследовать альтернативные, более правдоподобные пути. При последовательном декодировании число вычислений на одно ребро является случайной величиной, и в памяти декодера должны храниться вычисленные расстояния для всех исследованных ветвей. Первые исследования алгоритма последовательного декодирования выполнили Дж. Возенкрафт и Б. Рейффен. В 1963 году его усовершенствовал Р. М. Фано, в 1966 году эффективную модификацию этого алгоритма предложил К. Ш. Зигангиров (СССР), позднее в 1969 году аналогичное предложение сделал Ф. Джелинек (США).

Значительным достижением в области теории кодирования явилась разработка в 1967 году А. Витерби (США) весьма эффективного с вычислительной точки зрения алгоритма декодирования сверточных кодов по максимуму правдоподобия. Этот алгоритм, в отличие от алгоритма последовательного декодирования, исследует все возможные пути по кодовой решетке на длине кодового ограничения  $k$ , поэтому он применим для декодирования сверточных кодов при сравнительно небольших значениях  $K = 1 - 10$ . Сверточные коды, алгоритм Витерби и алгоритм последовательного их декодирования получили в настоящее время весьма широкое распространение в магистральных радиорелейных, спутниковых и других системах связи.

В 1957 году американец Е. Преиндж первым ввел понятие циклического кода, а годом раньше (1956 год) Д. А. Хаффменом (США) было предложено использовать регистры сдвига для формирования корректирующих циклических кодов. Заметный вклад в разработку теории таких кодов внесли американские ученые Питерсон, Берлекамп и Касами.

Весьма важный и обширный класс линейных циклических кодов Боуза, Рой-Чоудхури (США, 1960 год) и Хоквингема (Франция, 1959 год), называемых теперь кодами БЧХ (по первым буквам имен разработчиков), позволяют корректировать в принятой кодовой комбинации многократные ошибки. Эти коды имеют следующие параметры:  $n = 2^m - 1$ ;  $k > 2^m - 1 - mt$ ;  $d > 2t + 1$  (здесь  $n$  – длина кода,  $k$  – число информационных символов в кодовой комбинации,  $t$  – количество корректируемых ошибок,  $d$  – минимальное хэммингово расстояние между кодовыми комбинациями). Декодирование кодов БЧХ производят на основе решения определенных алгебраических уравнений. Широкую известность и применение нашли циклические блочные коды И. С. Рида и Г. Соломона (США), разработанные в 1960 году.

Параллельно с перечисленными работами велись исследования по методам исправления пакетов ошибок, характерных для многих каналов связи. Оказалось, что задача коррекции таких ошибок может быть сведена к задаче коррекции независимых ошибок путем применения метода перемежения символов, который состоит в перестановке передаваемых символов таким образом, чтобы соседние символы передаваемой кодовой комбинации оказывались разнесенными в достаточной степени во времени так, чтобы их искажения в канале связи были независимыми. На приеме же производится восстановление исходного порядка символов. Эта простая идея, выдвинутая в 1960 году советскими учеными А. А. Харкевичем и Э. Л. Блохом и позднее (1970 год) Дж. Л. Рамсеем (США), сегодня применяется во многих системах радиорелейной и подвижной связи, в которых при приеме цифровых сигналов имеет место группирование ошибок.

Особо остро вопросы оптимального кодирования стоят в случае использования в цифровой системе связи аналоговых, например, стандартных телефонных каналов. В этом случае приходится, кроме кодера/декодера (кодека), дополнительно использовать модулятор/демодулятор (модем). При этом наименьшую вероятность ошибки можно получить, вычисляя в демодуляторе расстояния между принятым сигналом и всеми возможными

кодowymi комбинациями. Решение принимается в пользу той кодовой комбинации, которая находится на минимальном расстоянии от принятого сигнала. Однако для кодов большой длины, имеющих огромное количество кодовых комбинаций, сложность демодулятора столь сильно возрастает, что его практическая реализация становится невозможной. Поэтому использование кодов дает наилучшие результаты при разумном согласовании модема и кодека. Это возможно, если в демодуляторе принимается так называемое мягкое (предварительное) решение о принимаемом символе и если декодер получает дополнительную информацию, повышающую надежность принятого в демодуляторе решения. В простейшем случае передачи бинарных сигналов символы, для которых надежные решения не могут быть приняты в демодуляторе, стираются, и декодер извещается о тех позициях кода, на которых находятся стертые символы.

Идеи применения мягкого решения в демодуляторе для декодирования кодовых комбинаций зародились почти с самого начала возникновения теории кодирования. Первой работой в этом направлении можно считать выполненное в 1954 году американскими учеными Сильверманом и Болсером исследование помехоустойчивости приема с мягким решением кода Вагнера, содержащего всего один избыточный двоичный символ для проверки информационных символов на четность. На приемной стороне наряду с жестким (окончательным) решением о значениях всех символов дополнительно определяется наименее надежный из них. При декодировании этот символ изменяется на противоположный, если проверка принятой кодовой комбинации на четность не выполняется. При наличии всего одного проверочного символа в кодовой комбинации такой алгоритм декодирования позволял корректировать одиночные ошибки. Эта идея была позже в обобщенном виде применена к декодированию кодов БЧХ в канале со стираниями ненадежно принятых символов.

Для сверточных кодов их декодирование при мягком решении в демодуляторе можно осуществлять с использованием алгоритмов Витерби и

последовательного декодирования. Для кодов с пороговым декодированием применение мягкого решения исследовалось в США Дж. Л. Мессе и несколько позже в СССР Л. М. Финком и Б. Д. Каганом. Для блочных кодов в начале 1970-х годов американскими учеными Е. Велдоном и Д. Чейзом были предложены алгоритмы декодирования, использующие мягкое решение демодулятора и позволяющие приблизиться к решению по максимуму правдоподобия.

В 1980-х годах появилось новое научное направление – методы помехоустойчивого приема с использованием специально построенных сигналов, названных СКК. Это направление представляет собой синтез методов модуляции и кодирования. Полученные результаты позволили создать высокоэффективные цифровые системы связи с использованием аналоговых каналов связи с ограниченной полосой частот. В 1982 году Г. Унгербоеком (США) была предложена решетчатая кодовая модуляция (РКМ), основанная на сочетании многопозиционных сигналов и помехоустойчивых кодов. При этом ансамбль многопозиционных сигналов, содержащий  $M = 2n$  сигналов, разбивается на  $K = 2k + 1$  вложенных подансамблей с монотонно возрастающими расстояниями. Для формирования многопозиционных сигналов используются многопозиционная фазовая и квадратурная амплитудная модуляции, а для преобразования информационных символов – сверточное кодирование. Для оптимального приема сигналов РКМ используется метод максимального правдоподобия, реализуемый в виде алгоритма Витерби. Применение РКМ при заданных скорости передачи сигналов и вероятности ошибочного приема позволяет уменьшить энергетические затраты и необходимую полосу частот канала связи. Весьма обширные исследования помехоустойчивости различных методов приема СКК в гауссовом канале связи были выполнены С. Л. Портным, В. В. Зябловым, В. В. Гинзбургом и В. Л. Банкетом (СССР) в течение 1982 – 1991 гг.

Сравнительно новый тип корректирующих кодов (они предложены французским исследователем Клодом Берру в 1993 году) – турбокоды –

обеспечивают уровень помехоустойчивости, приближающийся к теоретически достижимым значениям (так называемому пределу Шеннона). Они относятся к классу параллельных каскадных кодов, в которых проверочная группа формируется двумя кодерами составных рекурсивных сверточных кодов.

Принципы создания корректирующих кодов основываются на использовании соотношения  $M > M_p$ , где  $M = 2^n$  – общее число комбинаций разрядности  $n$ ;  $M_p$  – число комбинаций, используемых для передачи информации и называемых **разрешенными**; остальные  $M - M_p$  комбинаций называются **запрещенными**. Поскольку на приемной стороне известно, какие комбинации являются разрешенными и какие запрещенными, то искаженная помехой комбинация переходит в разряд запрещенных, т. е. ошибка будет обнаружена и при определенных условиях исправлена. Естественным, что ошибки, приводящие к образованию другой разрешенной комбинации, не обнаруживаются.

Различие между комбинациями кода характеризуется **кодовым расстоянием** (расстоянием Хэмминга)  $d$ , которое определяется числом единиц в сумме этих комбинаций по модулю 2, например  $1001111101 \oplus \oplus 1100001010 = 0101110111$ , т. е.  $d = 7$ .

Для безызбыточного (с использованием всех комбинаций в качестве разрешенных) первичного натурального кода кодовое расстояние равно единице, т. е. в таком коде отсутствует возможность обнаружения ошибок.

Ошибка всегда обнаруживается, если ее кратность (число искаженных символов в кодовой комбинации)

$$g \leq d - 1. \quad (1.2)$$

Если  $g > d$ , то некоторые ошибки также обнаруживаются, однако полной гарантии здесь нет, так как ошибочная комбинация в этом случае может совпасть с какой-либо разрешенной. Из (1.2) видно, что минимальное кодовое расстояние, при котором обнаруживаются любые одиночные ошибки, равно 2.

Процедура исправления ошибок в процессе декодирования сводится к определению расстояния  $d_o$  между переданной разрешенной комбинацией и

принятой запрещенной комбинацией, равного кратности ошибок  $g$ . Если ошибки в символах комбинации происходят независимо относительно друг друга, то вероятность искажения  $g$  символов в  $n$ -значной комбинации равна:

$$(P_o)^g \times (1 - P_o)^{n-g}, \quad (1.3)$$

где  $P_o$  – вероятность искажения одного символа.

Так как обычно  $P_o \ll 1$ , то вероятность многократных ошибок уменьшается с увеличением их кратности, при этом возможно и уменьшение  $d_o$ . В этом случае принятая запрещенная комбинация заменяется ближайшей разрешенной. Аналогичное правило используется в теории потенциальной помехоустойчивости при оптимальном приеме дискретных сигналов, когда решение сводится к выбору того переданного сигнала, который в наименьшей степени отличается от принятого. При таком правиле декодирования будут исправлены все ошибки кратности

$$g \leq (d - 1)/2, \quad (1.4)$$

т. е. минимальное значение  $d$ , при котором еще возможно исправление любых одиночных ошибок, равно 3.

В некоторых случаях решающее устройство приемника выдает, кроме обычных символов 0 и 1, еще так называемый символ стирания  $\theta$ , который соответствует приему сомнительных сигналов, когда затруднительно принять определенное решение, какой из символов 0 или 1 был передан. Принятый символ в этом случае стирается. Однако при использовании корректирующего кода возможно восстановление стертых символов. Если в кодовой комбинации число символов  $\theta$  оказалось равным  $g$ , а остальные символы приняты без ошибок, то такая комбинация восстанавливается.

Естественно, корректирующая способность кода возрастает с увеличением  $d$ . Это значит, что при фиксированном числе разрешенных комбинаций  $M_p$  это возможно лишь за счет увеличения количества запрещенных комбинаций

$$M - M_p = 2^n - 2^k, \quad (1.5)$$

что, в свою очередь, требует избыточного числа символов  $r = n - k$ , где  $k$  – количество информационных символов.

Избыточность кода можно определить как

$$\chi = (n - k)/n = 1 - (\log M_p)/(\log M). \quad (1.6)$$

Поскольку количество всевозможных сочетаний  $g$  ошибочных символов в  $n$ -значной кодовой комбинации зависит от ее длины и определяется формулой числа сочетаний  $C_n^g = n!/g!(n - g)!$ , то с учетом (1.3) полная вероятность ошибки кратности  $g$ , учитывающая все сочетания ошибочных символов, будет определяться формулой  $P_{og} = C_n^g P_o^g (1 - P_o)^{n-g}$ , из которой можно получить выражения для определения вероятностей правильного приема  $P_{np}$  и коррекции ошибок  $P_{кор}$ , а затем и формулу для вероятности некорректируемых ошибок:

$$P_{ош} = 1 - P_{np} - P_{кор} = 1 - (1 - P_o)^n - \sum_g C_n^g P_o^g (1 - P_o)^{n-g}, \quad (1.7)$$

где суммирование производится по всем значениям кратности ошибок  $g$ , которые обнаруживаются и исправляются.

Из (1.7) видно, что при малом значении  $P_o$  и сравнительно небольших значениях  $n$  наиболее вероятны ошибки малой кратности, которые и необходимо корректировать в первую очередь.

Рассмотренные характеристики – вероятность  $P_{ош}$ , избыточность  $\chi$  и общее число символов  $n$  кодовой комбинации – определяют корректирующую способность кода. Общая задача, которая ставится при создании корректирующего кода, заключается в достижении наименьших значений  $P_{ош}$  и  $\chi$ . Целесообразность применения того или иного кода зависит также от сложности кодирующих и декодирующих устройств, которая, в свою очередь, зависит от  $n$ .

#### 1.4. Принципы создания систематических кодов

Такие коды, как отмечалось ранее, относятся к блочным разделимым, в которых операции кодирования осуществляются независимо в пределах каждой

комбинации, состоящей из информационных и контрольных символов [9]. Если обозначить информационные символы буквами  $c$ , а контрольные – буквами  $e$ , то любую кодовую комбинацию из  $k$  информационных и  $r$  контрольных символов можно представить последовательностью:  $c_1, c_2, \dots, c_k, e_1, e_2, \dots, e_r$ , где  $c$  и  $e$  в двоичном коде принимают значения 0 или 1.

Процесс кодирования на передающей стороне сводится к образованию в различных сочетаниях (и по желанию разработчика кода) контрольных символов  $e_j$  путем суммирования по модулю два информационных символов:

$$e_j = a_{j1}c_1 \oplus a_{j2}c_2 \oplus \dots \oplus a_{jk}c_k = \sum_{i=1}^k a_{ji}c_i, \quad (1.8)$$

где  $j = 1, 2, \dots, r$  – номер контрольного символа;  $a_{ji}$  – коэффициенты, равные 0 или 1 (устанавливаются в соответствии с выбранным алгоритмом формирования кода).

Процедура декодирования может осуществляться разными методами, один из них – метод контрольных чисел – состоит в формировании по (1.8) из принятых комбинаций  $c_{1п}, c_{2п}, \dots, c_{kп}, e_{1п}, e_{2п}, \dots, e_{rп}$  второй группы контрольных символов  $e_{jп} = \sum_{i=1}^k a_{ji}c_{iп}$  (здесь как и в (1.8), знак суммы обозначает суммирование по модулю 2).

Следующим этапом является сравнение обеих групп контрольных символов путем их суммирования по модулю 2:

$$(e_1 \oplus e_{1п}), (e_2 \oplus e_{2п}), \dots, (e_r \oplus e_{rп}) = x_1, x_2, \dots, x_r = X. \quad (1.9)$$

Полученное число  $X$  называется контрольным числом или **синдромом**. С его помощью можно обнаружить или исправить часть ошибок. Если ошибки в принятой комбинации отсутствуют, то все суммы  $x_j = e_j \oplus e_{jп}$  и, следовательно, контрольное число  $X$  будут равны нулю. При появлении ошибок некоторые значения  $x$  могут оказаться равными 1. В этом случае  $X \neq 0$ , что и позволяет обнаружить ошибки. Таким образом, контрольное число  $X$  определяется путем  $r$  проверок на четность.



Для исправления ошибок, кроме факта их возникновения, необходимо указать номер ошибочно принятых символов. С этой целью каждому сочетанию исправляемых ошибок в комбинации присваивается одно из контрольных чисел, что позволяет по известному контрольному числу определить местоположение ошибок и исправить их.

Поскольку синдром  $X$  записывается в двоичной системе, то общее количество различных контрольных чисел, отличающихся от нуля, равно  $(2^r - 1)$  и должно быть не меньше числа различных сочетаний ошибочных символов, подлежащих исправлению. Например, если код предназначен для исправления одиночных ошибок, то число различных вариантов таких ошибок равно  $(k + r)$ . В этом случае должно выполняться условие

$$2^r - 1 \geq k + r, \quad (1.10)$$

из которого при заданном количестве информационных символов  $k$  можно определить необходимое число контрольных символов  $r$ , с помощью которых исправляются все одиночные ошибки.

Рассмотрим **простейшие систематические коды**. Одним из кодов подобного типа является код с четным числом единиц. Каждая комбинация такого кода содержит, кроме информационных, один контрольный символ, выбираемый равным 0 или 1, так чтобы сумма единиц в комбинации всегда была четной. Примером могут служить пятизначные комбинации кода Бодо, к которым добавляется шестой контрольный символ: 10101,1 и 01100,0.

Правило вычисления контрольного символа для каждой комбинации рассматриваемого кода аналогично (1.8), т. е.  $e = \sum_{i=1}^k c_i$  и, согласно указанному алгоритму образования такого кода,

$$e \oplus \sum_{i=1}^k c_i = 0 \quad (1.11)$$

откуда следует, что в декодирующем устройстве сравнительно просто производится обнаружение ошибок путем проверки на четность. Нарушение четности имеет место при появлении однократных, трехкратных и других

ошибок нечетной кратности, что и позволяет их обнаруживать. Однако четные ошибки не изменяют четности суммы (1.11), поэтому такие ошибки не обнаруживаются.

К достоинствам кода относится простота кодирующих и декодирующих устройств, а также низкая избыточность  $\chi = 1/(1+k)$ , что, однако, определяет его основной недостаток – сравнительно низкую корректирующую способность. Вероятность необнаруженной ошибки для этого кода определяется формулой (1.7).

Заметно лучшими корректирующими способностями обладает инверсный код, который также применяется только для обнаружения ошибок. Если количество единиц в информационных символах четное, т. е. сумма этих символов

$$c_{\Sigma} = \sum_{i=1}^k c_i \quad (1.12)$$

равна нулю, то контрольные символы представляют собой простое повторение информационных (например, комбинация 11000, 11000). В противном случае, когда число единиц нечетное и сумма (1.12) равна 1, контрольные символы получаются из информационных инвертированием, т. е. путем замены всех 0 на 1, а 1 на 0 (например, комбинация 01101, 10010).

При декодировании происходит сравнение принятых информационных и контрольных символов. Если сумма единиц в принятых информационных символах четная, то соответствующие друг другу информационные и контрольные символы суммируются по модулю 2; при нечетной сумме производится такое же суммирование, но с инвертированными контрольными символами.

Анализ показывает, что при  $k \geq 5$  наименьшая кратность необнаруживаемой ошибки  $g = 4$ . Причем не обнаруживаются только те ошибки четвертой кратности, которые искажают одинаковые номера информационных и контрольных символов. Например, если передана

комбинация 10100, 10100, а принята 10111, 10111, то такая четырехкратная ошибка обнаружена не будет, так как здесь все значения  $x_j$  равны 0.

Сравнительно высокая обнаруживающая способность инверсного кода достигается ценой сравнительно большой избыточности, которая, как нетрудно определить, составляет 0,5.

### 1.5. Коды Хэмминга.

Следующей разновидностью систематических кодов являются **коды Хэмминга** [9]. К этому типу кодов обычно относят систематические коды с расстоянием  $d = 3$ , которые, согласно (1.4), позволяют исправить все одиночные ошибки.

Рассмотрим код Хэмминга, каждая комбинация которого содержит четыре информационных и три контрольных символа (обозначается как (7,4)); он соответствует неравенству (1.10) и имеет избыточность  $\chi = r/(k + r) = 3/7$ .

Если информационные символы  $c$  занимают в комбинации (в групповом коде) первые четыре места, то последующие три контрольных символа образуются, согласно (1.8), как

$$e_j = a_{j1}c_1 \oplus a_{j2}c_2 \oplus a_{j3}c_3 \oplus a_{j4}c_4. \quad (1.13)$$

Декодирование осуществляется путем трех проверок на четность согласно (1.9):

$$x_1 = e_1 \oplus e_{1п}; \quad x_2 = e_2 \oplus e_{2п}; \quad x_3 = e_3 \oplus e_{3п} \text{ и}, \quad (1.14)$$

где  $e_{jп} = \sum_{i=1}^4 a_{ji}c_{iп}$ .

Так как в (1.14)  $x$ , может принимать значения 0 или 1, то всего может быть восемь ( $2^r = 8$ ) контрольных чисел  $X$ : 000, 100, 010, 001, 011, 101, 110 и 111. Первое из них имеет место в случае правильного приема, а остальные семь формируются при искажении информационных и контрольных символов.

Если искажен один из контрольных символов  $e_j$  то, как следует из (1.14), контрольное число примет одно из трех значений: 100, 010 или 001. Остальные

четыре контрольных числа используются для выявления ошибок в информационных символах.

Порядок присвоения контрольных чисел ошибочным информационным символам может устанавливаться в любой последовательности. Если для  $c^1 - c^4$  выбрать соответственно комбинации 011, 101, 110 и 111, то контрольные числа будут определяться формулами:

$$x_1 = e_1 \oplus c_2 \oplus c_3 \oplus c_4; \quad x_2 = e_2 \oplus c_1 \oplus c_3 \oplus c_4; \quad x_3 = e_3 \oplus c_1 \oplus c_2 \oplus c_4.$$

При искажении одного из информационных символов становятся равными единице те суммы  $x_i$ , в которые входит этот символ.

Возможен и другой метод отыскания одиночных ошибок, предложенный впервые самим Хэммингом. При этом методе код строится так, что контрольное число в двоичной системе счисления сразу указывает номер искаженного символа. Для этого контрольные символы располагаются на месте первого (младшего)  $a_1$ , второго  $a_2$  и четвертого  $a_4$  разрядов комбинации группового кода.

Описанный метод отыскания одиночных ошибок использован в схеме кодирующего и декодирующего устройства, реализованной в среде NI Multisim и представленной на рис. 1.4. В качестве источника информационных символов  $a_3, a_5, a_6, a_7$  в схеме используется генератор слова, который совместно с элементами И (AND) U1с, U2с выполняет также функции формирователя тактирующих сигналов для записи в триггеры T3, T5 – T7 информационных символов и в триггеры T1, T2, T4 – контрольных символов, формируемых элементами ИСКЛЮЧАЮЩЕЕ ИЛИ (EOR) C1, C2, C4. Единичные выходы триггеров Q подключены к индикаторам кода на логических пробниках X1 – X7 и опознавателям ошибки на элементах ИСКЛЮЧАЮЩЕЕ ИЛИ D1, D2, D4 с индикаторами номера ошибочного разряда E0 – E2 в двоичном коде и ERROR – в десятичном. Элементы ИСКЛЮЧАЮЩЕЕ ИЛИ выполняют роль сумматоров по модулю 2.

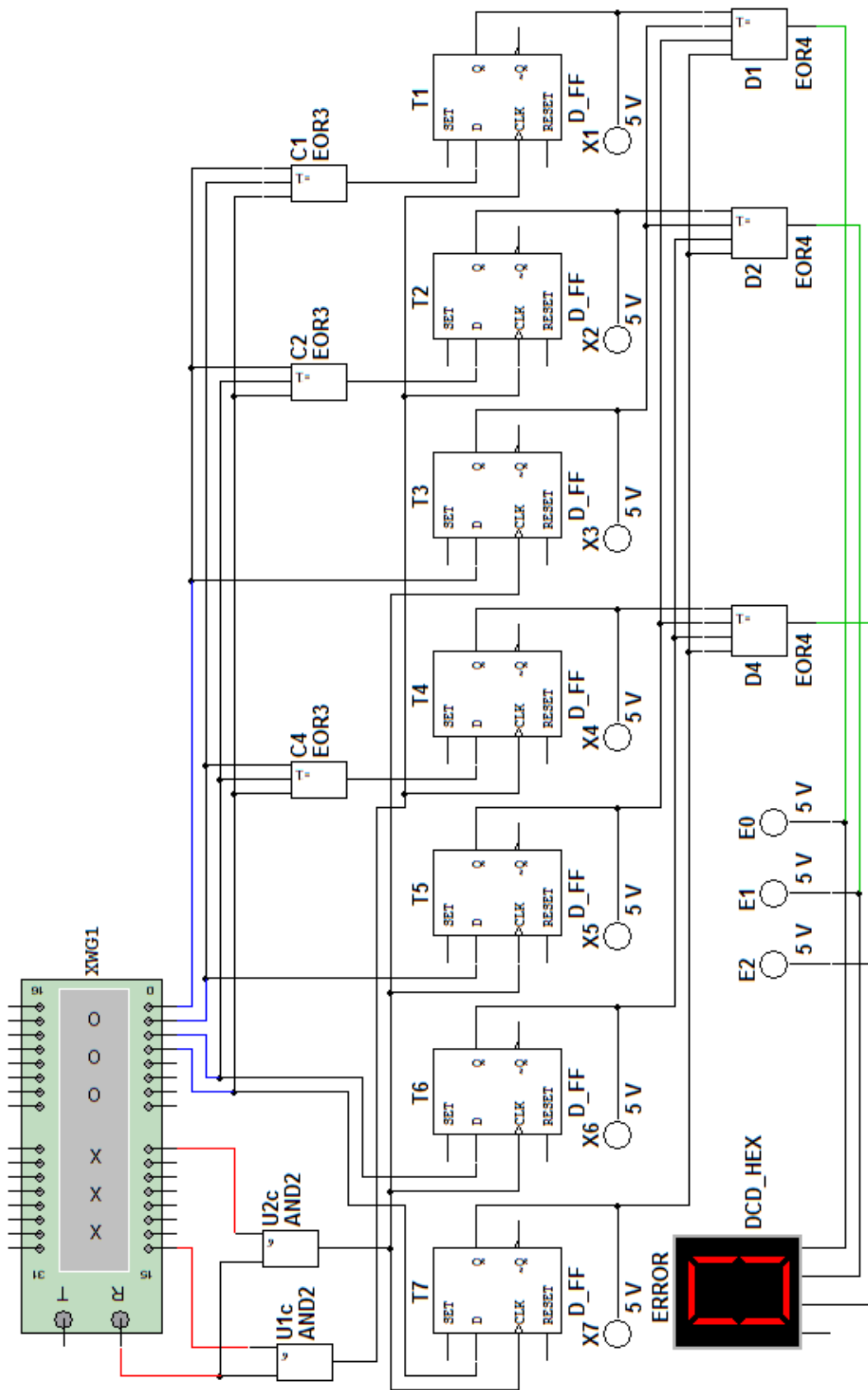


Рисунок 1.4. Кодирование и декодирование устройства кода Хэмминга

Контрольные символы формируются по правилам:

$$a1 = a3 \oplus a5 \oplus a7; \quad a2 = a3 \oplus a6 \oplus a7; \quad a4 = a5 \oplus a6 \oplus a7, \quad (1.15)$$

которые, как видно из схемы, реализуются элементами С1, С2 и С4 соответственно.

При первой комбинации (рис. 1.5) на выходе генератора формируется информационный код 0000, который обнуляет все триггеры под действием тактового сигнала с выхода R кодового генератора, поступающего на их тактовые V-входы через элементы U1с, U2с при разрешающих сигналах 1 на выходах 8 и 15 генератора.

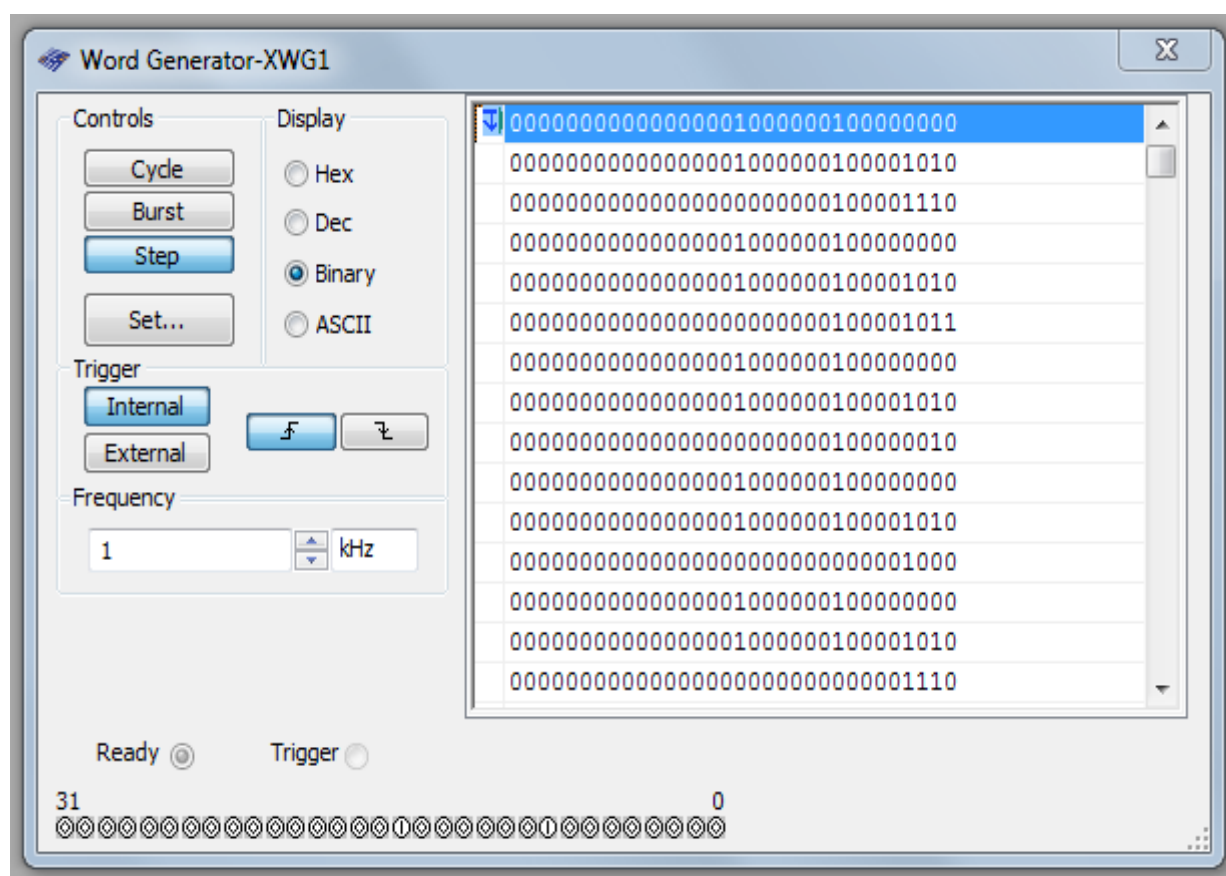


Рисунок 1.5. Окно генератора слова в кодере кода Хэмминга

При комбинации № 2 информационный код 1010 записывается в триггеры Т3, Т5 – Т7 и в соответствии с правилами (1.15) через С1, С2, С4 в виде контрольных символов – в триггеры Т1, Т2 и Т4, в результате чего формируется групповой код 1011010, который можно наблюдать на индикаторах Х1 – Х7.

Этот код поступает далее через линию связи в декодер приемника, который также содержит 7 триггеров и опознаватель, работающий по следующим правилам:

$$a1 \oplus a3 \oplus a4 \oplus a7 = 0; a2 \oplus a3 \oplus a6 \oplus a7 = 0; a4 \oplus a5 \oplus a6 \oplus a7 = 0, \quad (1.16)$$

которые реализуются элементами D1, D2, D4.

Поскольку в данной схеме роль приемных триггеров декодера играют триггеры кодера, т. е. T1 – T7, то при комбинации № 2, когда на декодер посылается тот же (безошибочный) информационный код 1010, индикатор ошибки ERROR будет показывать 0.

При комбинации № 3 вводится информационный код 1110, т. е. с ошибкой в разряде а6. В декодере этот код анализируется с помощью опознавателя и сигнал об ошибке в 6-м разряде через дешифратор может быть подан на счетный вход 6-го триггера декодера и переведет его в противоположное (исправленное) состояние. В нашем случае дешифратор находится внутри индикатора ERROR, который просто показывает номер разряда, в котором произошла ошибка.

При комбинации № 4 триггеры обнуляются, затем с генератора выдается код передачи 1010 (комбинация № 5), вводится ошибка 1011 (комбинация № 6), которая будет индцироваться цифрой 3 (ошибка в разряде а3 группового кода). Аналогичным образом определяются ошибки в других разрядах (комбинации № 9, 12, 15).

## 1.6. Циклические коды

Среди систематических кодов особое место занимают циклические, у которых строки образующих матриц могут быть получены циклическим сдвигом одной комбинации, называемой **образующей** для данного кода [9]. Сдвиг осуществляется справа налево, причем крайний левый символ каждый раз переносится в конец комбинации. Например, совокупность кодовых комбинаций, получающихся циклическим сдвигом одной шестиразрядной комбинации 010001, имеет вид: 010001, 100010, 000101, 001010, 010100,

101000. Как видно из этого примера, число возможных циклических кодов  $(n, k)$  значительно меньше числа различных групповых  $(n, k)$  кодов.

При описании циклических кодов удобно использовать запись любого  $n$ -разрядного двоичного числа в виде многочлена  $(n - 1)$ -й степени, содержащего фиктивную переменную  $x$  с показателем степени  $(n - 1)$  и коэффициентами при  $x$  (0 или 1 для двоичного кода). Например, двоичная кодовая комбинация 10101 может быть записана в виде многочлена  $G(x) = 1 \times x^4 + 0 \times x^3 + 1 \times x^2 + 0 \times x^1 + 1 \times x^0 = 1 \times x^4 + 1 \times x^2 + 1 \times x^0$ . Многочлен в таком виде называется **нормированным**, поскольку коэффициент при старшей степени равен единице. Таким образом, действия над кодовыми комбинациями сводятся к действиям над многочленами.

Вышеупомянутый циклический сдвиг некоторого образующего многочлена степени  $(n - 1)$  соответствует простому умножению на  $x$ . Умножив, например, многочлен  $(x^3 + x^2 + 1)$ , соответствующий комбинации 0001101, на  $x$ , получим многочлен  $(x^4 + x^3 + x)$ , соответствующий комбинации 0011010. Нетрудно убедиться, что кодовая комбинация, получающаяся сложением этих двух комбинаций, также будет соответствовать результату умножения многочлена  $(x^3 + x^2 + 1)$  на многочлен  $(x + 1)$ , если приведение подобных членов осуществлять по модулю 2. Действительно,  $(x^3 + x^2 + 0 + 1) \times (x + 1) = (x^4 + x^3 + 0 + x) \oplus \oplus (x^3 + x^2 + 0 + 1) = (x^4 + 0 + x^2 + 0 + x)$  и  $11010 \oplus 01101 = 10111$ . Из приведенного примера видно, что при соответствующем выборе образующего многочлена любая разрешенная кодовая комбинация циклического кода может быть получена в результате умножения образующего многочлена на некоторый другой многочлен. Это значит, что любой многочлен циклического кода делится на образующий многочлен без остатка. При этом ни один многочлен, соответствующий запрещенной кодовой комбинации, на образующий многочлен без остатка не делится. Это свойство позволяет обнаружить ошибку, а по виду остатка определить и вектор (код) ошибки. Поскольку умножение и деление многочленов весьма просто осуществляется



на регистрах сдвига с обратными связями, то это обстоятельство явилось причиной широкого применения циклических кодов.

Поскольку каждая разрешенная комбинация  $n$ -разрядного циклического кода является произведением двух многочленов, один из которых является образующим, то эти комбинации можно рассматривать как подмножество множества всех произведений многочленов степени не выше  $n - 1$ .

Следует заметить, что операция умножения многочленов по обычным арифметическим правилам с приведением подобных членов по модулю 2 может привести к нарушению условия замкнутости. Действительно, в результате умножения могут быть получены многочлены более высокой степени, чем  $(n - 1)$ , вплоть до  $2(n - 1)$ , а соответствующие им кодовые комбинации будут иметь число разрядов, превышающее  $n$ , и, следовательно, не будут относиться к рассматриваемому множеству. Поэтому операция символического умножения выполняется в следующем порядке.

1. Многочлены перемножаются по обычным правилам, но с приведением подобных членов по модулю 2.

2. Если старшая степень произведения не превышает  $(n - 1)$ , то оно является результатом символического умножения.

3. Если старшая степень произведения больше или равна  $n$ , то многочлен произведения делится на заранее определенный многочлен степени  $n$  и результатом символического умножения считается остаток от деления.

Степень остатка не будет превышать  $(n - 1)$ , и следовательно, этот многочлен будет принадлежать к рассматриваемому множеству  $n$ -разрядных кодовых комбинаций. Для того чтобы найти вид этого многочлена ( $n$ -й степени), остаток в виде многочлена степени  $(n - 1)$  умножается на  $x$ . Чтобы результат умножения и теперь соответствовал кодовой комбинации, образующейся путем циклического сдвига исходной кодовой комбинации, в нем необходимо заменить  $x$  на 1. Такая замена эквивалентна делению полученного при умножении многочлена на  $(x^n + 1)$  с записью в качестве результата остатка от деления, что обычно называется **взятием остатка** или

приведением по модулю  $(x^n + 1)$  (сам остаток в этом случае называется вычетом). Искомым многочленом  $n$ -й степени является, следовательно, многочлен  $(x^n + 1)$ .

Итак, при заданных операциях сложения и умножения все множество многочленов степени не выше  $(n - 1)$ , соответствующее множеству  $n$ -разрядных кодовых комбинаций, образует кольцо. Подмножество всех многочленов, кратных некоторому многочлену наименьшей степени  $g(x)$ , называется **идеалом**, а многочлен  $g(x)$  – **порождающим многочленом** идеала. При этом количество различных элементов в идеале определяется видом его порождающего многочлена. Если за порождающий многочлен взять 0, то весь идеал будет составлять только этот многочлен, так как умножение его на любой другой многочлен дает 0. Если за порождающий многочлен принять 1 ( $g(x) = 1$ ), то в идеал войдут все многочлены кольца. В общем случае число элементов идеала, порожденного простым многочленом степени  $(n - k)$ , составляет  $2^k$ .

Согласно определению циклического кода все многочлены, соответствующие его кодовым комбинациям, должны делиться на  $g(x)$  без остатка. Для этого достаточно, чтобы на  $g(x)$  делились без остатка многочлены, которые получаются циклическим сдвигом, что соответствует последовательному умножению  $g(x)$  на  $x$  с приведением по модулю  $(x^n + 1)$ .

Таким образом, чтобы  $g(x)$  мог породить идеал, а следовательно, и циклический код, он должен быть делителем многочлена  $(x^n + 1)$ . Если многочлен  $g(x)$  степени  $m = n - k$  является делителем  $(x^n + 1)$ , то любой элемент кольца либо делится на  $g(x)$  без остатка (тогда он является элементом идеала), либо в результате деления появляется остаток, представляющий собой многочлен степени не выше  $(m - 1)$ .

Любая принятая по каналу связи кодовая комбинация  $h(x)$ , содержащая ошибку, может быть представлена в виде суммы по модулю 2 неискаженной комбинации кода  $f(x)$  и вектора (кода) ошибки  $z(x)$ :  $h(x) = f(x) \oplus z(x)$ . При делении  $h(x)$  на образующий многочлен  $g(x)$  остаток, указывающий на наличие

ошибки, будет обнаружен только в том случае, если многочлен, соответствующий вектору ошибки, не делится на  $g(x)$ .

Вектор одиночной ошибки будет иметь единицу в искаженном разряде и нули во всех остальных разрядах. Ему будет соответствовать многочлен  $z(x) = x^i$ , который не должен делиться на  $g(x)$ . Остаток от деления любого многочлена на  $(x + 1)$  представляет собой многочлен нулевой степени и может принимать только два значения: 0 или 1. Все кольцо в данном случае будет состоять из идеала, содержащего многочлены с четным числом членов, и одного класса вычетов, соответствующего единственному остатку, равному единице. Таким образом, при любом числе информационных разрядов необходим только один проверочный разряд. Значение символа этого разряда как раз и обеспечивает четность числа единиц в любой разрешенной кодовой комбинации, а, следовательно, и делимость ее на  $(x + 1)$ .

Для исправления одиночной ошибки в принятой комбинации из  $n$  разрядов необходимо определить, какой из разрядов был искажен. Это можно сделать только в том случае, если каждой одиночной ошибке в определенном разряде соответствует свой класс вычетов и свой опознаватель. Поскольку в циклическом коде опознавателями ошибок являются остатки от деления многочленов ошибок на образующий многочлен кода  $g(x)$ , то  $g(x)$  должен обеспечить требуемое число различных остатков при делении векторов ошибок с единицей в искаженном разряде.

При степени многочлена  $m = n - k$  он может дать  $(2^{n-k} - 1)$  ненулевых остатков (нулевой остаток является опознавателем безошибочной передачи). Следовательно, необходимым условием исправления любой одиночной ошибки является выполнение неравенства  $2^{n-k} - 1 \geq n$ , откуда находится степень образующего многочлена кода  $m = n - k \geq \log_2(n + 1)$  и общее число символов в кодовой комбинации.

Наибольшие значения  $k$  и  $n$  при различных  $m$  можно найти с помощью таблицы, фрагмент которой приведен ниже.

Таблица 1.1. Зависимость значений  $k$  и  $n$  от величины  $m$ 

|     |   |   |   |    |    |    |     |     |     |      |
|-----|---|---|---|----|----|----|-----|-----|-----|------|
| $m$ | 1 | 2 | 3 | 4  | 5  | 6  | 7   | 8   | 9   | 10   |
| $n$ | 1 | 3 | 7 | 15 | 31 | 63 | 127 | 255 | 511 | 1023 |
| $k$ | 0 | 1 | 4 | 11 | 26 | 57 | 120 | 247 | 502 | 1013 |

Располагая выбранными значениями  $m$ ,  $n$  и используя таблицы неприводимых многочленов [9], можно выбрать образующий многочлен, после чего необходимо убедиться в том, что он обеспечивает заданное число остатков. Например, при  $n = 15$  и  $m = 4$  в таблице неприводимых многочленов [9] найдем один многочлен первой степени  $(x + 1)$ , один многочлен второй степени  $(x^2 + x + 1)$  и три многочлена четвертой степени:  $(x^4 + x + 1)$ ,  $(x^4 + x^3 + 1)$ ,  $(x^4 + x^3 + x^2 + x + 1)$ . Перемножив все многочлены, убеждаемся в том, что их произведением является двучлен  $(x^{15} + 1) = (x^n + 1)$ , т. е. один из сомножителей четвертой степени может быть принят за образующий многочлен кода. Выберем, например, многочлен  $(x^4 + x^3 + 1)$  или 11001 в двоичном коде. Чтобы убедиться, что каждому вектору ошибки соответствует отличный от других остаток, необходимо поделить каждый из этих векторов на 11001, начиная с вектора 00...00001 младшего разряда.

Для этого разряда такое деление выглядит следующим образом:

$$\begin{array}{r} 00...010000)11001 \\ \oplus \underline{11001} \\ 1001 \end{array}$$

Остаток 1001 и все последующие остатки, полученные аналогичным образом, делятся на образующий многочлен 11001 до тех пор, пока последний не достигнет значения 1001, после чего они начнут повторяться. Если количество различных остатков будет 15, то образованный таким  $g(x)$  код способен исправить любую одиночную ошибку. Аналогичные результаты получаются и в случае выбора  $g(x) = (x^4 + x + 1)$ . Однако использовать для тех же целей многочлен  $(x^4 + x^3 + x^2 + x + 1)$ , оказывается, нельзя, поскольку число различных остатков у него не 15, а только 5. Это объясняется тем, что он входит в разложение не только двучлена  $(x^{15} + 1)$ , но и двучлена  $(x^5 + 1)$ . Следовательно, в качестве образующего необходимо выбирать такой

неприводимый многочлен  $g(x)$  (или произведение таких многочленов), который, являясь делителем двучлена  $(x^n + 1)$ , не входит в разложение ни одного двучлена  $(x^z + 1)$ , степень которого  $z$  меньше  $n$  (в этом случае говорят, что многочлен  $g(x)$  принадлежит показателю степени  $n$ ).

Образующие многочлены кодов, способных обнаруживать одиночные, двойные и тройные ошибки, можно определить следующим образом. Если известен образующий многочлен  $p^m(x)$  кода длины  $n$ , позволяющего обнаруживать ошибки некоторой кратности  $q$ , то образующий многочлен  $g(x)$  код, способный обнаруживать ошибки следующей кратности  $(q + 1)$ , может быть получен умножением многочлена  $p^m(x)$  на многочлен  $(x + 1)$ , что соответствует введению дополнительной проверки на четность. При этом число символов в комбинациях кода за счет дополнительного проверочного символа увеличивается до  $(n + 1)$ .

Циклический код, исправляющий более одной ошибки в любом сочетании, в общем случае можно построить следующим образом.

1. По заданному значению  $k$  определяется необходимое для исправления одной ошибки значение  $(n - k)$  и строится  $(n, k)$  код.
2. Рассматривая  $(n, k)$  код как  $n$ -разрядный некорректирующий, определяется  $(n_1 - n)$  дополнительных разрядов для обеспечения исправления одной ошибки в этом коде и строится код  $(n_1, n)$ .
3. Повторяя данную процедуру  $t$  раз, можно получить код, исправляющий независимые ошибки кратности до  $t$  включительно.

Полученный описанным образом код, однако, оказывается неоптимальным по числу проверочных символов при данном значении  $k$ . Этого недостатка лишены циклические коды БЧХ и их подкласс – коды Рида-Соломона, которые требуют, однако, значительно более сложных устройств для обнаружения и исправления ошибок.

В случае циклических кодов, исправляющих пачки ошибок, образующий многочлен  $g(x)$  представляет собой произведение неприводимого многочлена степени  $m$  на другой многочлен, вид которого определяется типом кода, а

старшая степень  $c$  – длиной пачки ошибок, на исправление которой код рассчитан. Следовательно, показатель степени  $g(x)$  равен  $(c + m)$ .

В общем случае степени  $m$  и  $c$  не могут принимать произвольные значения и для различных типов кодов связаны различной функциональной зависимостью (длина кодовой комбинации  $n$  также является функцией  $m$  и  $c$ ).

В настоящее время из циклических кодов для исправления пачек ошибок наиболее известны коды Файра, Абрамсона, Миласа-Абрамсона и Рида-Соломона. Первые три разновидности кодов рассчитаны на исправление в кодируемом блоке одной пачки ошибок, а коды Рида-Соломона – нескольких пачек.

В кодах Файра образующий многочлен имеет вид:  $g(x) = R(x^m)(x^c + 1)$ , где  $R(x^m)$  – неприводимый многочлен степени  $m$ , а  $c$  – некоторое постоянное число. Общее число символов  $n$  в комбинации кода равно общему наименьшему кратному  $c$  и  $(2^m - 1)$ ; число проверочных символов (и максимальная длина пачки) –  $(c + m)$ .

При исследованиях циклического кода широко используется матричная форма представления. Полная образующая матрица циклического кода  $M_{n,k}^T = \|I_k^T C_{n-k,k}^T\|$  составляется из двух матриц: единичной транспонированной в канонической форме  $I_k^T$ , соответствующей  $k$  информационным разрядам, и дополнительной  $C_{n-k,k}^T$ , соответствующей проверочным разрядам.

Обычно строки дополнительной матрицы определяются путем вычисления многочленов  $r(x)$ . Для каждой строки матрицы  $I_k^T$  соответствующий остаток  $r(x)$  находится делением информационного многочлена  $a(x) \times x^m$  этой строки на образующий многочлен кода  $g(x)$ .

Дополнительную матрицу можно определить также делением на  $g(x)$  комбинации в виде единицы с рядом нулей и использования получающихся остатков в качестве строк дополнительной матрицы. Если степень какого-либо остатка  $r(x)$  оказывается меньше  $(n - k - 1)$ , то следующие за этим остатком строки матрицы получают путем циклического сдвига предыдущей строки

влево до тех пор, пока степень  $r(x)$  не станет равной  $(n - k - 1)$ . Деление производится до получения  $k$  строк дополнительной матрицы. Например, для упоминавшегося выше кода  $(15, 11)$  с образующим многочленом  $g(x) = x^4 + x^3 + 1$  полная образующая матрица имеет вид:

$$M_{15,11}^T = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{vmatrix} \quad (1.17)$$

Аппаратная реализация формирователей циклического кода наиболее просто осуществляется умножением многочлена исходного (информационного) кода на образующий многочлен  $g(x)$ . Однако такой способ имеет один недостаток: получающиеся в результате умножения комбинации кода не содержат информационных символов в явном виде и для их выделения требуется деление на образующий многочлен. Поэтому на практике используется метод, при котором в выходном коде под информационные символы  $k$  отводятся его старшие разряды, а под проверочные  $(n - k)$  – младшие. При этом применяется следующая процедура кодирования.

Многочлен  $a(x)$ , соответствующий  $k$ -разрядному входной коду, умножается на  $x^m$ , где  $m = n - k$ . Степень каждого одночлена, входящего в  $a(x)$ , увеличивается, что по отношению к комбинации кода означает добавление со стороны младших разрядов  $m$  нулей. Затем произведение  $a(x) \times x^m$  делится на образующий многочлен  $g(x)$ . В общем случае при этом получается некоторое частное  $q(x)$  той же степени, что и  $a(x)$ , и остаток  $r(x)$ , который прибавляется к  $a(x) \times x^m$ . В результате получается многочлен  $f(x) = a(x) \times x^m \oplus r(x)$ . Поскольку степень  $g(x)$  выбирается равной  $m$ , то степень остатка  $r(x)$  не превышает  $(m - 1)$ . В комбинации, соответствующей многочлену  $a(x) \times x^m$ ,  $m$  младших разрядов

нулевые и, следовательно, указанная операция сложения равносильна приписыванию  $r(x)$  к  $a(x)$  со стороны младших разрядов.

Таким образом, циклический код можно строить, приписывая к каждой комбинации входного кода остаток от деления соответствующего этой комбинации многочлена на образующий многочлен кода. Для кодов, число информационных символов в которых больше числа проверочных, рассмотренный способ реализуется наиболее просто.

Еще один способ кодирования связан с тем обстоятельством, что циклический код является разновидностью группового, что позволяет выражать его проверочные символы через суммы по модулю 2 определенных информационных символов путем решения рекуррентного уравнения

$$a_{i+k} = \sum_{j=0}^{k-1} h_j a_{i+j}, \quad \text{где } h_i \text{ — двоичные коэффициенты так называемого}$$

**генераторного многочлена**  $h(x) = (x^n + 1)/g(x)$ . Как и в предыдущем случае, проверочные символы размещаются на местах младших разрядов, а при одних и тех же информационных символах комбинации кода совпадают с комбинациями, получающимися при использовании предыдущего способа кодирования. Применение данного способа целесообразно для кодов с числом проверочных символов, превышающим число информационных, например для кодов БЧХ.

Основой кодирующих и декодирующих устройств циклических кодов являются регистры сдвига с обратными связями, позволяющие осуществлять как умножение, так и деление многочленов с приведением коэффициентов по модулю 2. Такие регистры также называют многотактными линейными переключательными схемами или **линейными кодовыми фильтрами Хаффмена**. Они состоят из триггерных ячеек памяти, сумматоров по модулю 2 и устройств умножения на коэффициенты многочленов множителя или делителя. В случае двоичных кодов для умножения на коэффициент, равный 1, требуется только наличие связи в схеме; если коэффициент равен 0, то связь отсутствует. Сдвиг информации в регистре осуществляется импульсами,



поступающими с тактового генератора кодера передатчика и синхронизированного с ним генератора декодера на приемной.

В качестве примера рассмотрим схемы кодера и декодера циклического кода (7, 4), выполненные в среде NI Multisim. Кодер построен на базе 4-разрядного регистра на триггерах T0 – T3 (рис. 1.6) с использованием порождающего (образующего) многочлена  $g(x) = x^3 + x^2 + 1$  или 1101.

Схема обратной связи на элементах ИСКЛЮЧАЮЩЕЕ ИЛИ U4 и U5 соответствует генераторному многочлену, что при выбранном  $g(x)$  равно  $h(x) = (x^7 + 1)/(x^3 + x^2 + 1) = x^4 + x^3 + x^2 + 1$  или 11101. Исходное состояние генератора слова (рис. 1.7), используемого в качестве источника входного кода  $U_i$ , синхронизирующих (с выхода R) и управляющих сигналов U&1 и U&2, отмечено на его дисплее (позиция № 0).

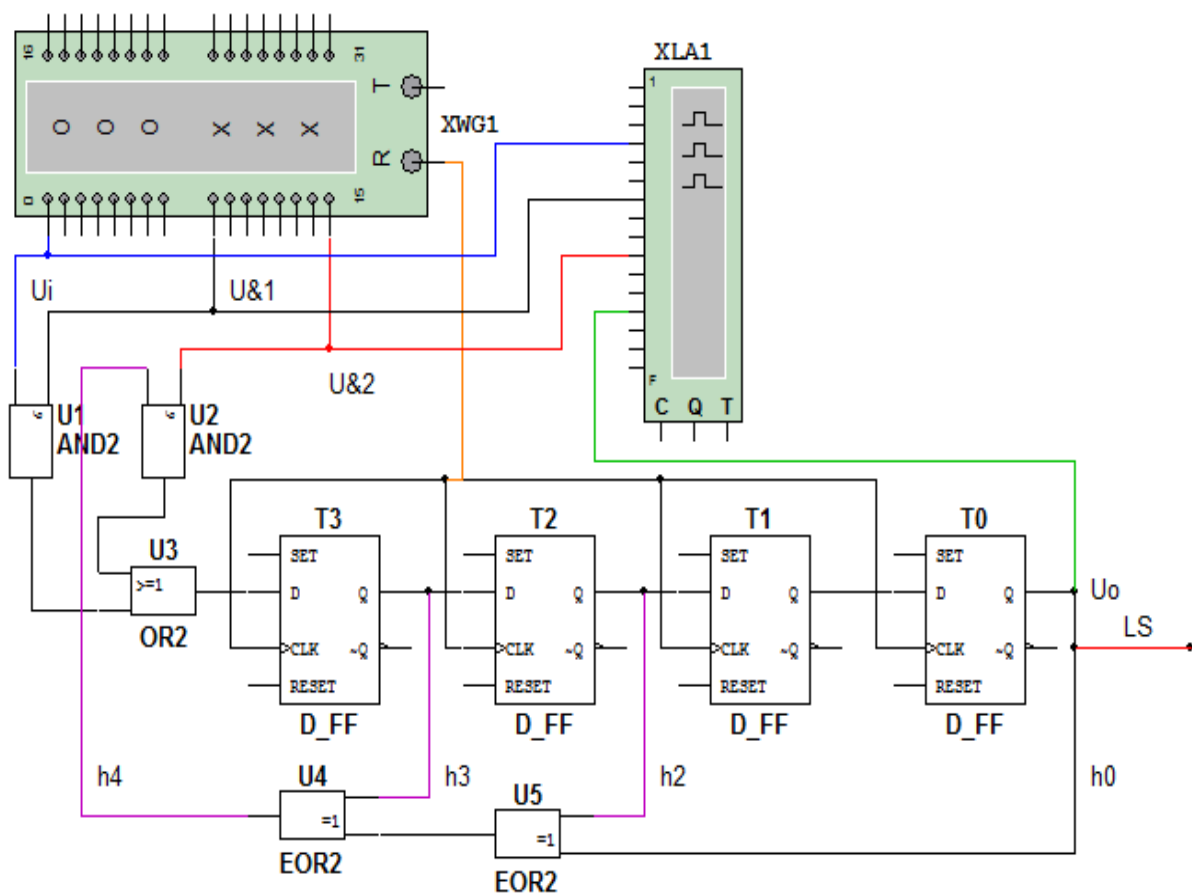


Рисунок 1.6. Кодер циклического кода

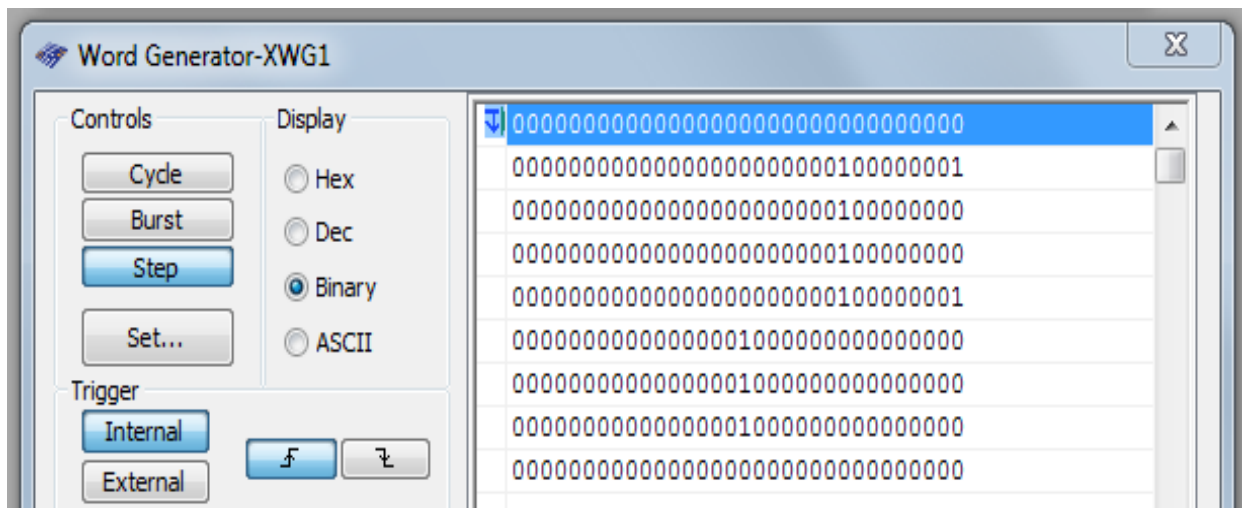


Рисунок 1.7. Окно генератора слова кодера циклического кода

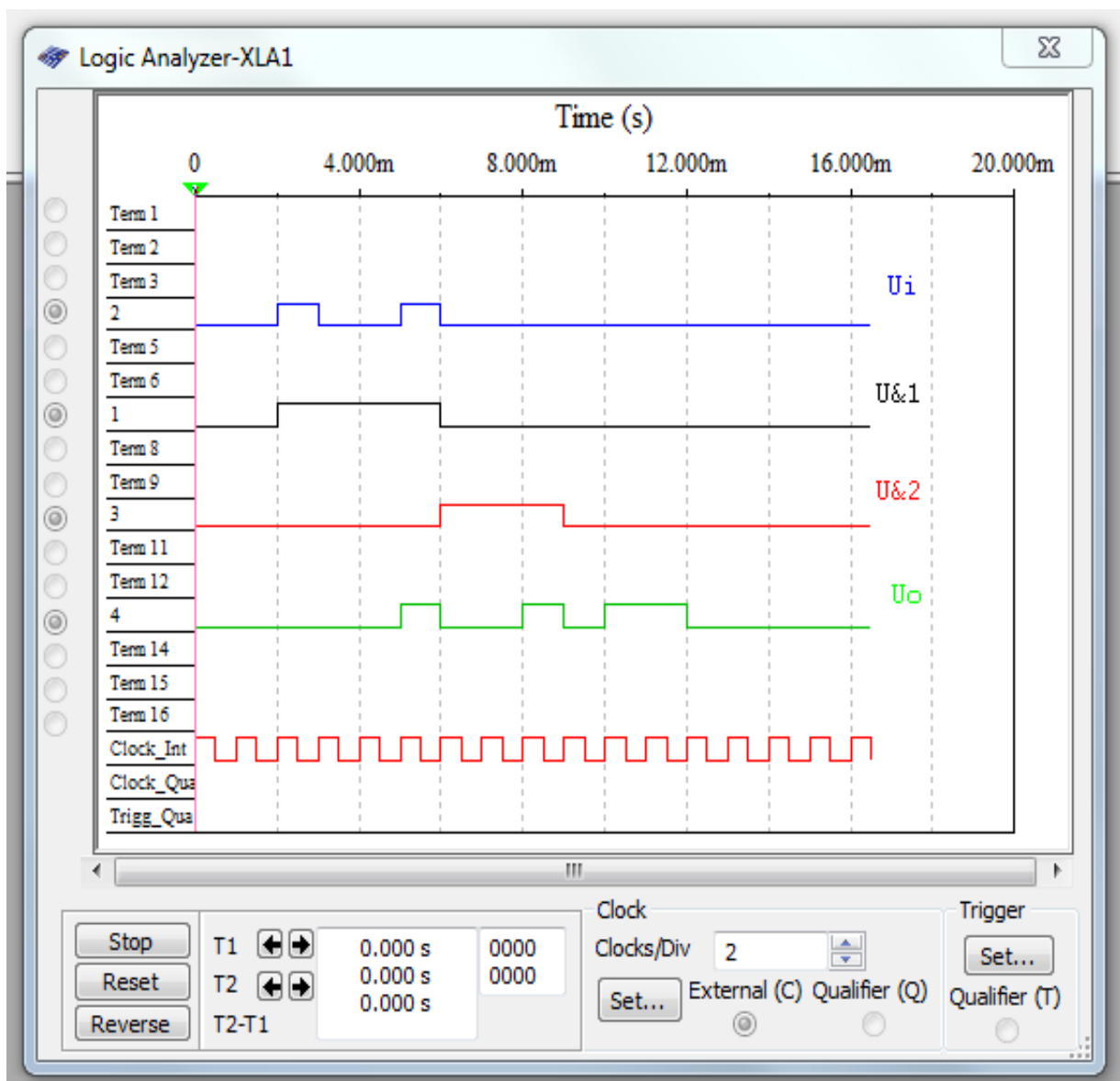


Рисунок 1.8. Окно логического анализатора кодера циклического кода

В позициях (тактах) 1 – 4, задаваемых кнопкой STEP, исходный код  $a(x) = 1001$  (см. рис. 1.7 и осциллограмму  $U_i$  на рис. 1.8) с нулевого выхода генератора последовательно, начиная со старшего разряда, поступает на вход регистра через элемент ИЛИ (OR)  $U_3$  и элемент И (AND)  $U_1$  на второй (разрешающий) вход которого поступают сигналы  $U\&1$  в указанных тактах с восьмого выхода генератора.

Начиная с пятого такта, разрешающие сигналы  $U\&2$  с пятнадцатого выхода генератора начинают поступать на элемент И  $U_2$ , в результате чего на вход регистра поступают сигналы обратной связи с  $U_4$ ,  $U_5$ . При этом формируется проверочный код 011, который вслед за информационным кодом 1001 поступает в линию связи (осциллограмма  $U_o$  на рис. 1.8) за счет сдвига информации в регистре под действием синхроимпульсов с генератора.

В декодере циклического кода в качестве входного используется код 1001011, полученный на выходе кодера, но с ошибкой в младшем разряде, т. е. 1000011 (рис. 1.9).

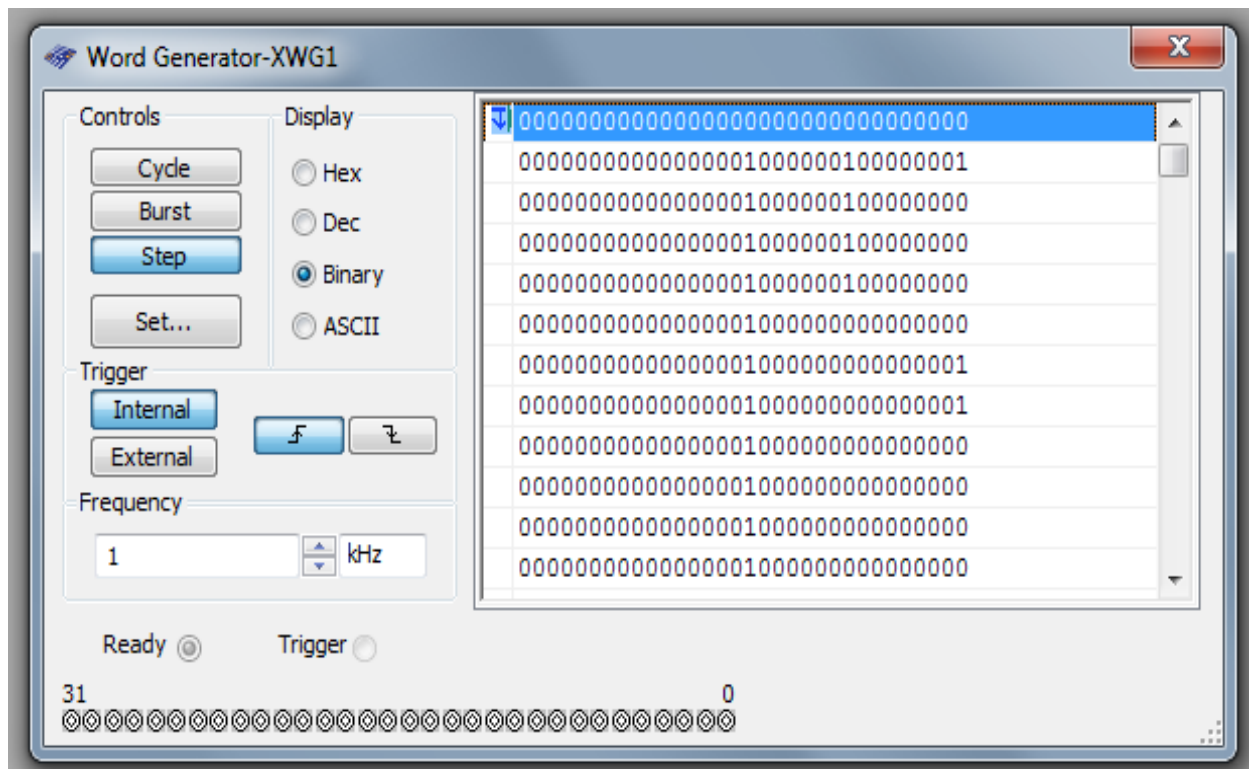


Рисунок 1.9. Окно генератора слова декодера циклического кода

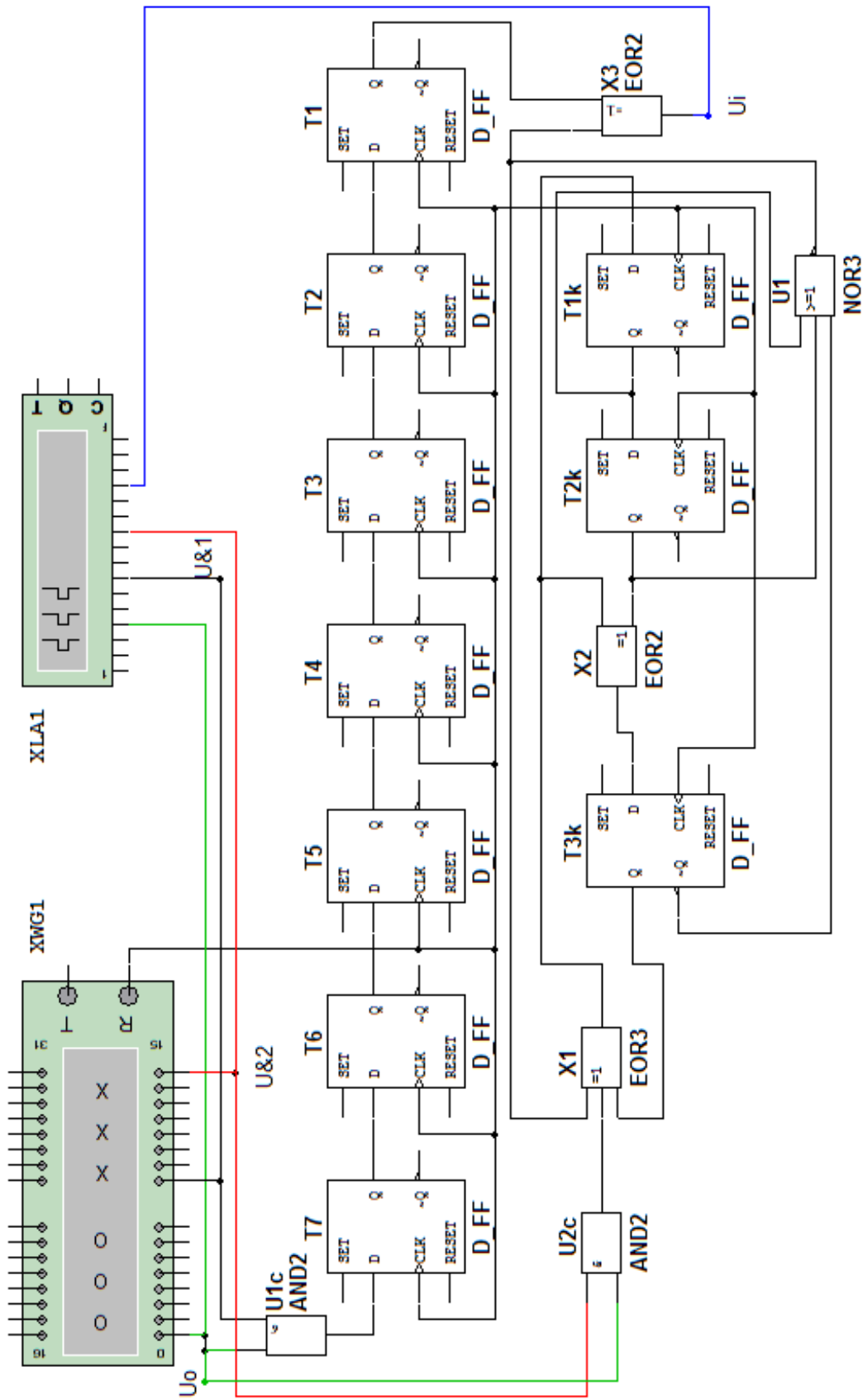


Рисунок 1.10 – Декодер циклического кода

Этот код поступает в тактах 1 – 7 с выхода 0 генератора слова (см. рис. 1.10 и осциллограмму  $U_0$  на рис. 1.11). Поскольку разрешающий сигнал  $U&1$  с восьмого выхода генератора поступает на элемент И (AND)  $U1c$  только в тактах 1 – 4, то на буферный регистр на триггерах  $T1 – T7$  поступает только информационный код 1000, который в дальнейшем сдвигается вправо.

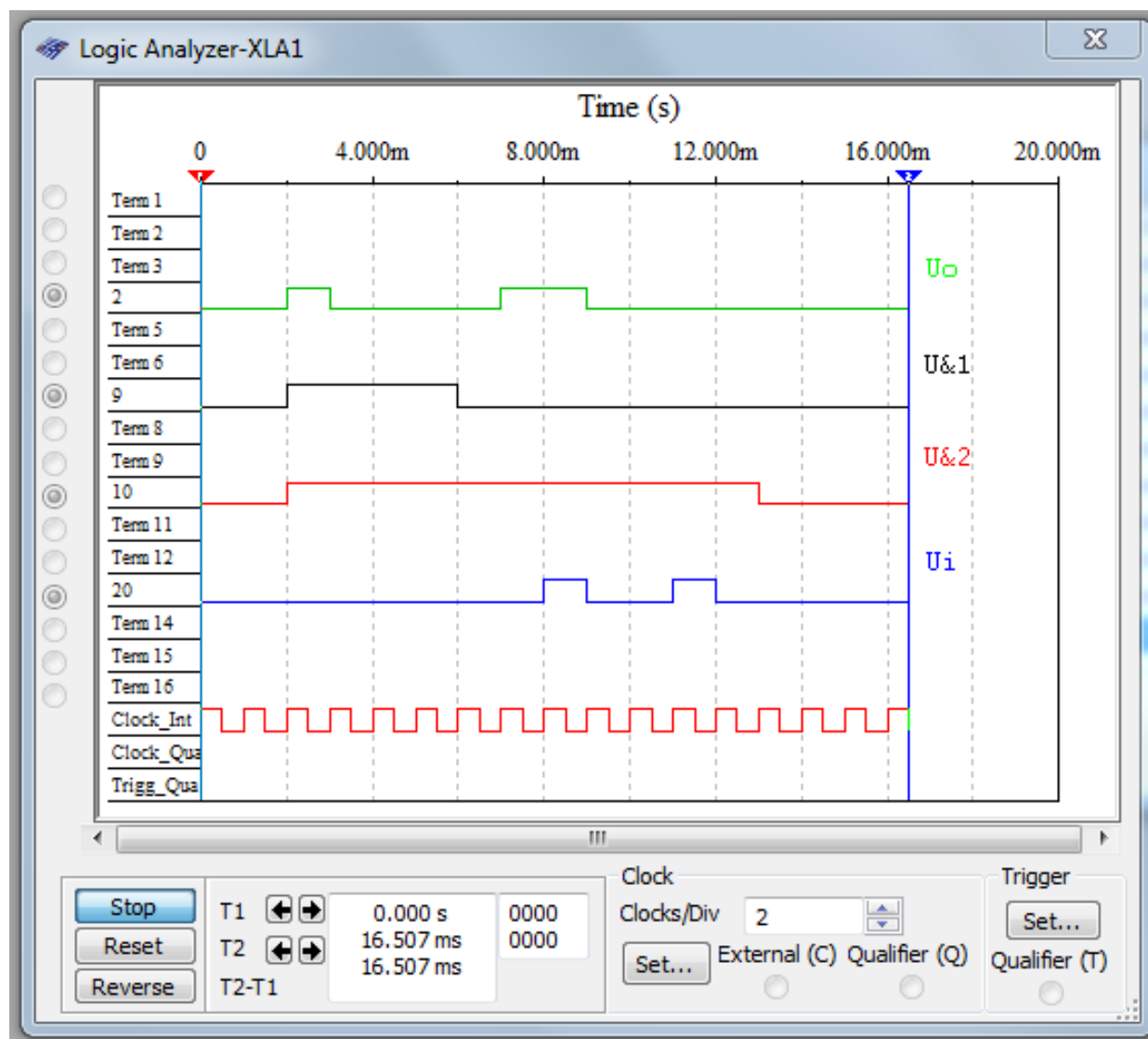


Рисунок 1.11. Окно логического анализатора декодера циклического кода

В то же время из-за того, что разрешающий сигнал  $U&2$  с выхода 15 генератора поступает на элемент И (AND)  $U&2$  в тактах 1 – 11, входной код 1000011 через ИСКЛЮЧАЮЩЕЕ ИЛИ (EOR)  $X1$  подается на  $(n - k = 3)$ -разрядный регистр на триггерах  $T1k – T3k$ , который совместно с  $X2$ , ИЛИ-НЕ

(NOR)  $U_1$  образует кодер, также основанный на использовании порождающего многочлена  $g(x) = x^3 + x^2 + 1$ .

Выходной код этого кодера с помощью элемента ИСКЛЮЧАЮЩЕЕ ИЛИ (EOR)  $X_3$  корректирует выходной код буферного регистра, устраняя возникшую ошибку (см. осциллограмму  $U_i$  на рис. 1.11).

## **1.7. Лабораторная работа 1. Исследование особенностей работы систем кодирования информации**

### **1.7.1. Цель и содержание**

Целью лабораторной работы является приобретение студентами практических навыков исследования принципов функционирования систем кодирования источника информации и кодирования канала.

#### **Содержание**

1. Исследование схемы дискретизации.
2. Исследование особенностей работы схемы преобразования натурального кода в код Грея.
3. Исследование моделей кодирующего и декодирующего устройства кода Хэмминга.
4. Исследование моделей кодирующего и декодирующего устройства циклического кода.

### **1.7.2. Аппаратура и материалы**

Для изучения материалов этой лабораторной работы необходимо:

1. Компьютер под управлением операционной системы Windows XP/Vista/7 с параметрами по умолчанию.
2. Установленная инструментальная среда визуального моделирования динамических систем VisSim/Comm 7.0.
3. Установленная среда проектирования NI Multisim 10.

4. Текстовый процессор (Microsoft Word 2003/2007/2010 или его аналог), предназначенный для создания и редактирования отчета по лабораторной работе.

### **1.7.3. Порядок выполнения работы**

1. Изучение теоретического материала.
2. Выполнение задания, предлагаемого в лабораторной работе.

#### **Задание**

1. Запустить на выполнение файл 1-1-1.VSM. Провести исследование особенностей работы схемы дискретизации для различных значений дискретизируемого сигнала и импульсной последовательности. Сделать выводы.

2. Используя правило формирования кода Грея из натурального кода, составить таблицу таких кодов для десятичных чисел 0 – 15.

Запустить на выполнение файл 1-2-1.ms10. Провести исследование особенностей работы схемы преобразования натурального кода в код Грея. Составить таблицу кодов для генератора слова, обеспечивающих преобразование натуральных кодов десятичных чисел 9, 10, 11, 13, 14, 15 в код Грея.

Используя правило преобразования кода Грея в натуральный код, составить схему модели, реализующей такое преобразование.

3. Запустить на выполнение файл 1-3-1.ms10. Провести исследование особенностей работы схемы кодирующего и декодирующего устройства кода Хэмминга. Определите ошибочные разряды в комбинациях № 9, 12, 15 на рис. 1.5. Ответ проверьте моделированием.

Сформировать ошибочные последовательности сигналов с генератора слова данной схемы для нерассмотренных комбинаций при входном коде 1010.

4. Запустить поочередно на выполнение файлы 1-3-2.ms10 и 1-3-3.ms10. Провести исследование особенностей работы схемы кодера и декодера

циклического кода. Определить взаимосвязь логики работы устройств и осциллограмм в определенных точках схемы.

#### **1.7.4. Вопросы для защиты работы**

1. Охарактеризуйте существующие типы кодирования информации.
2. На каких принципах основана дискретизация непрерывных сообщений по времени. В каких устройствах она впервые нашла применение?
3. Какая разница между дискретизацией и квантованием непрерывных сообщений?
4. Для чего требуется избыточность отсчетов при дискретизации непрерывных сообщений по времени, и какими параметрами она определяется?
5. Какие типы кодов получаются при кодировании источников информации?
6. Назовите и охарактеризуйте параметры первичного кода.
7. Что собой представляют взвешенные и невзвешенные коды? Назовите достоинства последнего. В каких случаях эти достоинства наиболее ярко проявляются?
8. Перечислите особенности кода Грея. К какому типу кодов он относится?
9. Как классифицируются корректирующие коды? Что послужило причиной их разработки? Опишите историю их развития.
10. Каким образом исправляются ошибки в кодах, которые обнаруживают только их?
11. В чем состоят основные принципы коррекции ошибок?
12. Дайте определения разрешенных и неразрешенных кодовых комбинаций.
13. Дайте определение кодового расстояния.
14. При каких условиях код может обнаруживать или исправлять ошибки?
15. Как используется корректирующий код в системах со стиранием?
16. Какие характеристики определяют корректирующие способности кода?



17. Каким образом составляются кодовые комбинации в систематических кодах?

18. Как формируются контрольные символы систематических кодов? Как определяется их необходимое количество?

19. Что такое синдром?

20. На чем основан принцип корректирования ошибок с использованием контрольного числа?

21. Назовите основные особенности кодов Хэмминга.

22. Как осуществляется процедура кодирования в семизначном коде Хэмминга?

23. Каким образом формируются циклические коды?

24. Запишите код 110101101 в виде многочлена.

25. Какой двоичный многочлен называется нормированным?

26. Что собой представляет порождающий многочлен и как он определяется?

27. Каким образом производится умножение и деление многочленов с приведением подобных членов по модулю 2?

28. Какое свойство порождающего многочлена позволяет использовать его для обнаружения и исправления ошибок циклического кода?

29. Что является опознавателем безошибочной передачи циклического кода?

30. Охарактеризуйте способы генерации циклического кода.

31. Найдите генераторный многочлен

$$h(x) = (x^n + 1)/g(x) = (x^7 + 1)/(x^4 + x^3 + x^2 + x + 1).$$

Напомним, что деление многочленов выполняется обычным (в столбик) образом, но с приведением подобных членов по модулю 2.

32. Каким образом размещаются информационные и проверочные биты в корректирующем циклическом коде?

## ГЛАВА 2. КОДИРОВАНИЕ В ЛИНИЯХ СВЯЗИ

Для согласования характеристик среды передачи информации – линии связи – с передаваемыми по ней цифровыми сигналами требуется их дополнительное преобразование, которое называется **кодированием в линии связи, линейным кодированием** или **кодированием на физическом уровне** [9]. К такому кодированию предъявляются следующие требования:

- надежное выделение сигналов синхронизации в приемнике;
- исключение постоянной или медленно меняющейся составляющих в линии связи (ЛС), приводящих к искажению цифровых сигналов по амплитуде;
- обеспечение минимальных частотных искажений сигналов;
- сужение или выравнивание энергетического спектра передаваемых сигналов в полосе пропускания ЛС, что способствует повышению отношения сигнал/шум;
- повышение помехоустойчивости путем нарушения статистических связей между элементами пакетных помех;
- обеспечение скорости передачи на уровне этого параметра для исходного кода;
- универсальность алгоритма преобразования (код должен работать при любой форме передаваемых символов);
- кодирование и декодирование должны быть достаточно простыми, чтобы уменьшить стоимость оборудования;
- процесс декодирования должен обеспечивать однозначную идентификацию исходных данных.

### 2.1. Униполярные коды

Эти коды относятся к классу потенциальных; информация в этом случае передается уровнями напряжения или тока сигнала. Различают две формы таких кодов [9]:

**RZ** (Return to Zero – с возвратом к нулю) – уровень возвращается к нулю в конце каждого кодового импульса; в этом случае биты 1 разделяются короткими паузами; передача нулевого бита производится нулевым уровнем сигнала;

**NRZ** (Non Return to Zero – без возврата к нулю) – самый простой способ линейного кодирования; в этом случае биты 1 не разделяются, например код 111 представляется одним импульсом длительностью, равной суммарной длительности трех битов; передача нулевого бита производится нулевым уровнем сигнала.

При использовании униполярных кодов возникают две проблемы. Во-первых, при передаче последовательности нулей или единиц отсутствуют изменения сигнала, что затрудняет его синхронизацию (в случае RZ-кода эта проблема частично решается за счет разделения при передаче единичных битов), и, во-вторых, присутствует постоянная составляющая.

## 2.2. Простейший биполярный код

В таком коде логический нуль передается отрицательным уровнем, равным по величине положительному уровню логической единицы (возможен и обратный вариант) [9]. Он отличается от униполярного кода заметно меньшим уровнем постоянной составляющей и незначительным улучшением характеристик самосинхронизации. Кроме того, биполярный код может быть как с возвратом к нулю, так и без него.

Наибольшее распространение биполярный код получил в системах асинхронной передачи, работающих в основной полосе частот информационного сигнала (без несущей). В таких системах передатчик и приемник оснащаются тактовыми генераторами с периодом следования сигналов (импульсов)  $T$ . Передатчик сначала группирует биты в короткие слова по  $K$  бит (например,  $K = 8$ ) и затем устанавливает в ЛС нулевое значение сигнала. Для передачи первого кодового слова передатчик устанавливает напряжение  $+V$  вольт на время  $T$  для каждого двоичного нуля в кодовом слове

и  $-V$  вольт для каждой двоичной единицы, что примерно соответствует амплитудной манипуляции (АМн), но без несущей. После передачи первого кодового слова передатчик опять устанавливает в ЛС нулевое значение сигнала. Такой переход заставляет приемник запустить его тактовый генератор в целях определения момента начала приема информации. После приема одного слова приемник переходит в режим ожидания следующего перехода сигнала с  $+V$  или  $-V$  в нулевое значение. Таким образом, передача и прием слов длительностью  $K$  бит при асинхронной передаче могут начаться в произвольный момент времени. При этом понятие «основная полоса» означает, что полоса частот определяется временными характеристиками информационного сигнала (иногда используется понятие «передача в полосе частот модулирующих сигналов»). Отсутствие постоянной синхронизации при асинхронном методе передачи ограничивает размер слова и скорость передачи, поскольку различие значений тактовой частоты вызывает ошибки в работе приемника при достаточно большой последовательности посылок.

Описанный метод передачи используется в телекоммуникациях при последовательной асинхронной передаче двоичных данных между терминалом и коммуникационным устройством (интерфейс RS-232). В случае оптической ЛС, передатчик посылает слова путем переключения источника света в положение «Включено» при передаче 1 и «Выключено» для каждого 0. Поскольку при этом логическая 1 будет заполнена высокочастотным «оптическим» сигналом, то в этом случае будем иметь случай «настоящей» АМн. Для подготовки приемника к приему передатчик добавляет в начало каждого слова ведущую 1, которая называется **стартовым битом** и используется приемником для обнаружения начала принимаемого кодового слова. Такая процедура передачи применяется, например, в устройствах дистанционного управления телевизорами.

В случае радиоканала для передачи могут быть использованы сигналы с амплитудной, частотной или фазовой манипуляцией. При этом принцип синхронизации аналогичен рассмотренному для оптической ЛС, хотя возможен

способ и с условным нулем в виде определенного значения частоты в случае частотной манипуляции (ЧМн), амплитуды для АМн или фазы при фазовой манипуляции (ФМн).

Схема формирователя биполярного кода из униполярного, выполненная в среде NI Multisim, содержит (рис. 2.1): на передающей стороне (в кодере) – D-триггер T1, элементы И (AND) U1, U2, инвертор U3, операционный усилитель (ОУ) O1 с напряжениями насыщения  $V_{sw+} = 5\text{ В}$  и  $V_{sw-} = -5\text{ В}$  (остальные параметры – по умолчанию); на приемной стороне (в декодере) – инвертор U4 и диодный ограничитель сигналов отрицательной полярности (резистор R1 и диод D1).

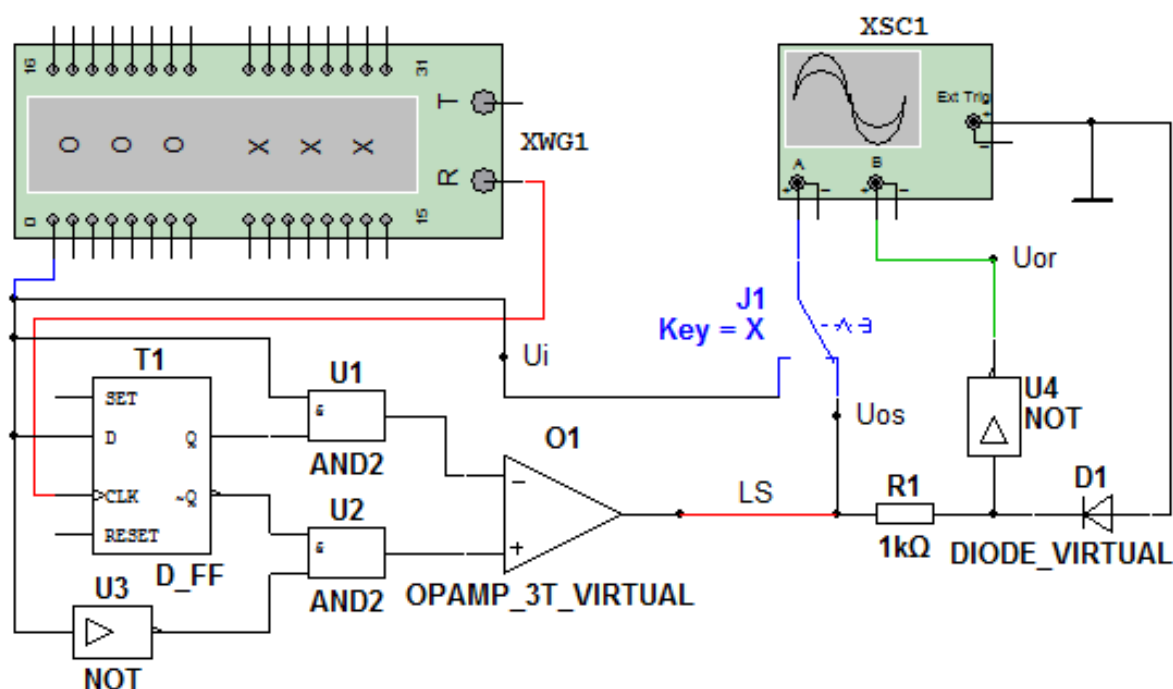


Рисунок 2.1. Схема биполярного кодера/декодера

В качестве источника входного кода  $U_i = 0110100101101$  используется нулевой выход генератора слова (рис. 2.2); синхроимпульс для тактирования триггера подается с выхода R генератора.

Из сравнения осциллограмм  $U_{os}$  и сигнала  $U_{or}$  в линии связи (рис. 2.3) видно, что для декодирования  $U_{os}$  достаточно «отрезать» отрицательные импульсы и затем оставшиеся положительные инвертировать. Данная операция реализуется диодным ограничителем и инвертором U4.

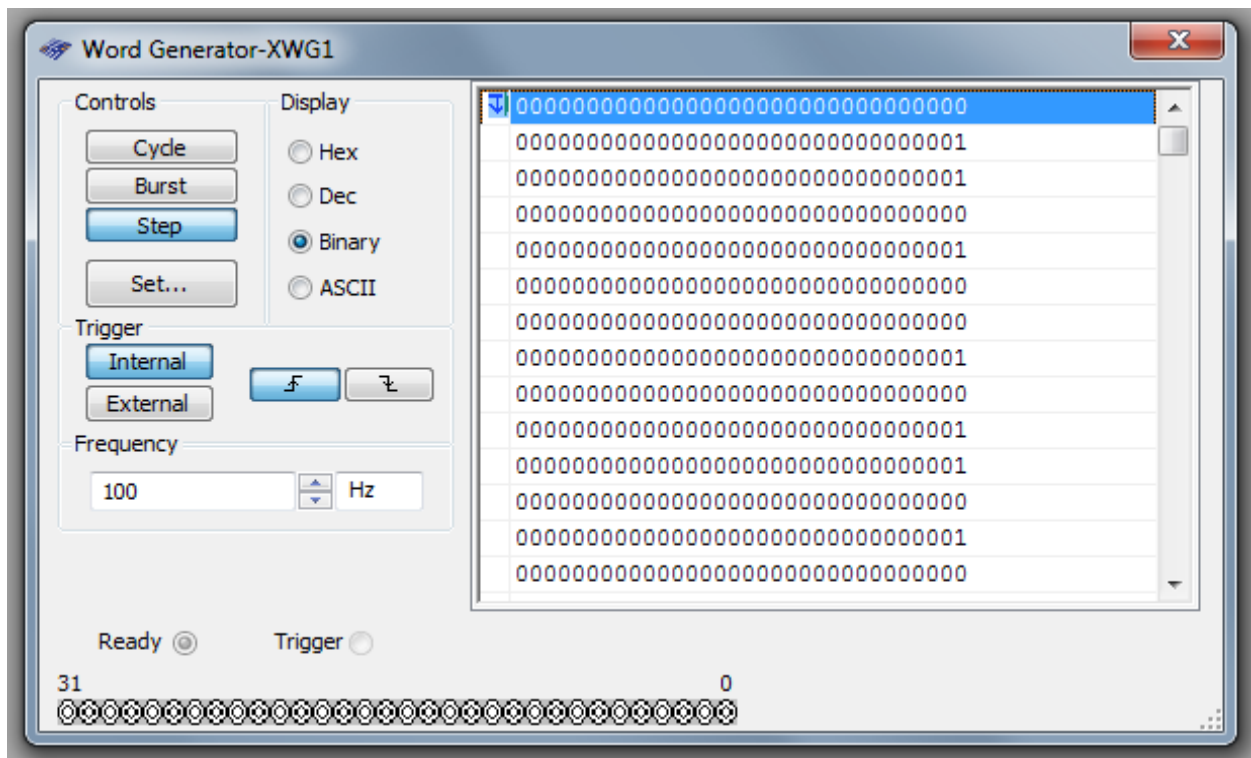


Рисунок 2.2. Окно генератора слова биполярного кодера/декодера

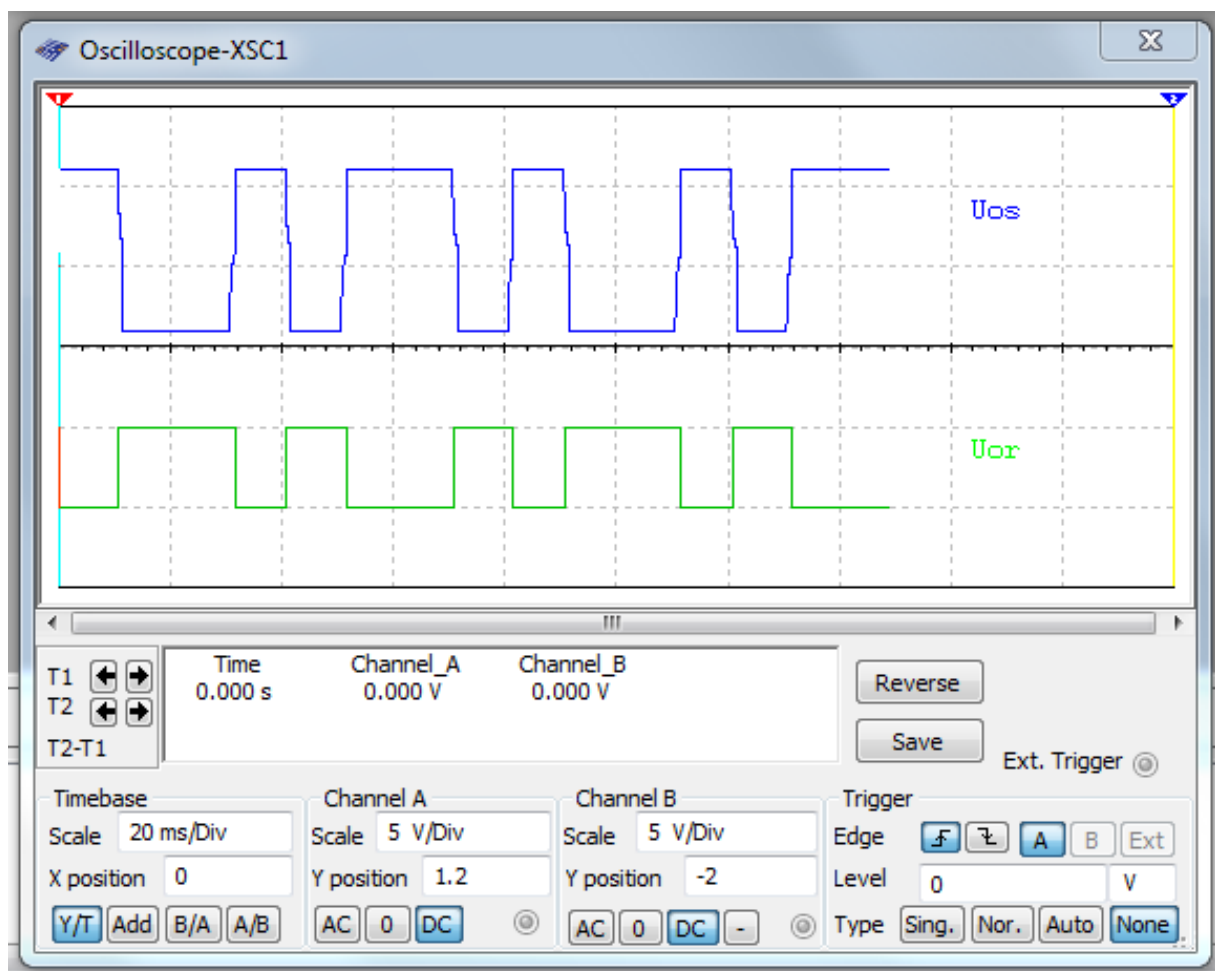


Рисунок 2.3. Осциллограммы биполярного кодера/декодера

Как видно из осциллограммы  $U_{or}$ , сигнал на приемной стороне полностью соответствует входному, в чем можно убедиться подключением канала А осциллографа к нулевому выходу генератора слова с помощью ключа X (после нажатия одноименной клавиши клавиатуры).

Формирование кодированного сигнала  $U_{os}$  происходит следующим образом. При нулевом значении входного бита он инвертируется инвертором  $U_3$ ; поскольку при этом в триггер  $T_1$  записывается 0, то на его инверсном выходе  $Q' = 1$ , в результате сигнал логической 1 с выхода И  $U_2$  передается без инверсии на выход ОУ. При единичном значении входного бита элемент И  $U_2$  блокируется сигналом 0 с выхода  $U_3$ , триггер  $T_1$  переводится в единичное состояние ( $Q = 1$ ), в результате чего формируемый на выходе  $U_1$  бит 1 поступает на инвертирующий вход ОУ и передается на его выход с инверсией ( $U_{os} = -5B$ ).

### 2.3. Манчестерский код

В этом коде кодирование осуществляется фронтом импульса [9]. Перепад потенциала происходит в середине ( $T/2$ ) тактового импульса, при этом единица кодируется перепадом от низкого потенциала к высокому, а ноль – наоборот. В начале каждого такта в случае появления нескольких нулей или единиц подряд может возникать так называемый служебный перепад потенциала, задаваемый в принудительном порядке передатчиком, т. е. передатчик формирует последовательность двоичных символов таким образом, чтобы сигналы в линии связи меняли свое состояние (1 – 0 или 0 – 1) не реже чем через  $T$  секунд. Такие переходы приемник использует для синхронизации с помощью системы фазовой автоподстройки частоты. Для обеспечения начальной синхронизации передатчик посылает набор дополнительных бит, который называется **синхронизирующим заголовком** или синхрокомбинацией и передается перед информационным сигналом. Примером такой синхрокомбинации является последовательность 0101010...010111. Если приемник входит в синхронизм во время получения синхрокомбинации, то он может определить начало

информационной последовательности путем определения последнего бита синхροкомбинации.

Большинство локальных сетей, включая сети Ethernet и Token Ring, использует витую пару и манчестерский код. Это означает, что при скорости передачи 10 Мбит/с, для передачи бита 0 интерфейс устанавливает напряжение в проводах равным  $-0,85$  В на  $0,05$  мкс, и после этого  $+0,85$  В на  $0,05$  мкс. Подобным же образом интерфейс передает бит 1 установкой напряжения сначала  $+0,85$  В на  $0,05$  мкс, а затем  $-0,85$  В на  $0,05$  мкс. Передача каждого бита, таким образом, занимает  $0,1$  мкс.

Напряжение в момент после середины интервала передачи бита положительно, если передается 0, и отрицательно в случае 1. В манчестерском коде в середине интервала передачи каждого бита происходит переход напряжения от одного разрешенного уровня к другому. Приемник использует эти переходы для определения момента времени, когда он должен измерить напряжение сигнала, чтобы определить передаваемый бит. Для этого приемник использует фазовую автоподстройку частоты (ФАПЧ) генератора тактовых импульсов, который корректирует свою частоту таким образом, чтобы окончание периода попадало точно, в середину интервала передачи бита. Сразу после окончания периода приемник измеряет напряжение сигнала и в зависимости от знака напряжения решает, принята 1 или, напротив, 0. Для запуска ФАПЧ физический уровень добавляет к началу пакета 64-битную строку 101010...101011, которая называется **преамбулой** (определителем начала кадра).

Один из вариантов схемной реализации формирователя манчестерского кода в среде NI Multisim показан на рис. 2.4. В качестве источника входной последовательности 010011010011010 приведенной на рис. 2.5 (осциллограмма  $U_i$ ) используется левый выход генератора слова, курсор которого в исходном состоянии находится в верхней позиции при выключенной кнопке моделирования.





генератора слова длительностью, равной половине длительности бита  $T/2$ . После первого нажатия кнопки Step в триггер T1 записывается 0, т. е. на его инверсном выходе  $Q' = 1$ . Поскольку после записи наступает пауза синхроимпульса (нулевого уровня и длительностью  $T/2$ ), то после ее инвертирования элементом НЕ (NOT) U3 сигнал  $Q' = 1$  через логические элементы И (AND) U2 и ИЛИ (OR) U4 передается на выход  $U_o$  (см. осциллограмму  $U_o$ ). Таким образом, входной нулевой бит передается на выход через полпериода тактовой частоты в виде перехода 0 – 1, т. е. от низкого уровня к высокому (рис. 2.6).

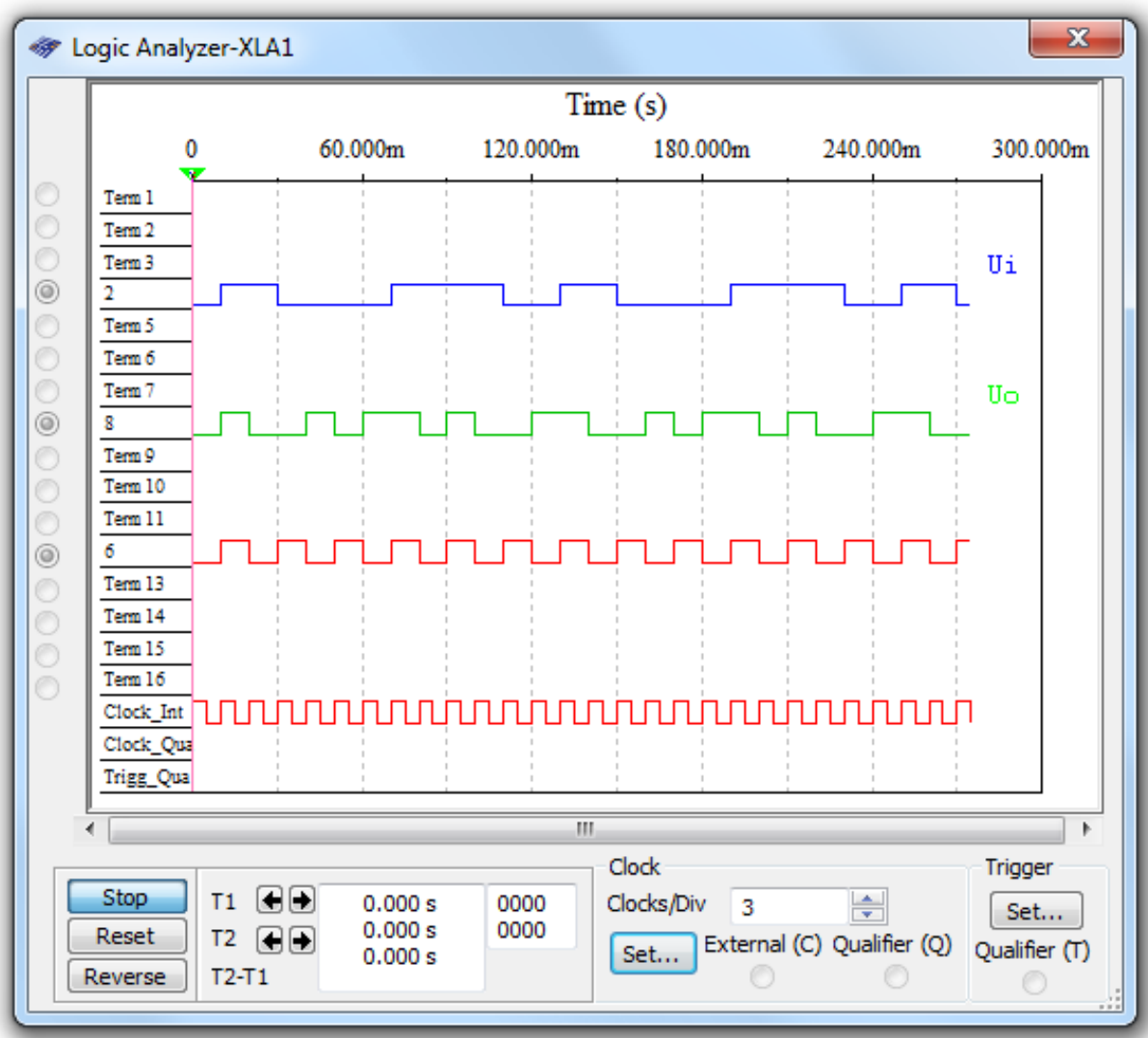


Рисунок 2.6. Осциллограммы формирователя манчестерского кода

После второго нажатия кнопки Step курсор дисплея перемещается в положение 1. При этом в триггер T1 записывается входной бит 1, т. е. на его прямом выходе  $Q = 1$ , который в течение следующих полпериода тактовой частоты, равной длительности тактового импульса, проходит на выход  $U_0$  через логические элементы И (AND) U1 и ИЛИ (OR) U4. После окончания действия тактового импульса, «сопровождающего» единичный входной бит, выходной сигнал претерпевает переход 1 – 0 (от высокого уровня к низкому), что является признаком единичного бита в манчестерском коде. При поступлении на вход формирователя двух нулевых битов (в позициях 2 и 3) они передаются на выход в виде двух переходов 0 – 1, имеющийся при этом переход 1 – 0 при передаче первого нулевого бита из этой пары не считается за 1, поскольку длительность информационного импульса равна длительности тактового. Аналогичным образом происходит и передача пары единичных битов (см. осциллограммы  $U_i$  и  $U_0$  для позиций 4, 5 и 10, 11).

Вследствие того, что при передаче каждого бита производится дополнительное изменение значения сигнала в течение периода  $T$ , манчестерский код удваивает полосу частот сигнала. Это приводит к тому, что при передаче по ЛС в виде витой пары информации со скоростью 100 Мбит/с уровень побочного излучения становится недопустимо высоким. Поэтому при таких скоростях в сетях Ethernet и FDDI (Fiber Distributed Data Interface – волоконно-оптический распределенный интерфейс данных) с использованием витой пары применяются другие самосинхронизирующиеся коды.

## 2.4. NRZI и другие коды

Код NRZI (Non Return to Zero with one Inverted – без возврата к нулю с инверсией единиц) является простейшей реализацией принципа кодирования сменой уровня сигнала или дифференциального кодирования [9]. При таком кодировании при передаче нуля уровень сигнала не меняется, оставаясь таким же, как и в предыдущем такте. При передаче единицы потенциал инвертируется на противоположный. По сравнению с NRZ код NRZI обладает лучшей

самосинхронизацией в том случае, если в кодируемой информации логических единиц больше, чем логических нулей. Поскольку в передаваемом сообщении может быть любое сочетание 1 и 0, в том числе и с малым количеством единиц, NRZI применяется в комбинации с находящимся в передатчике линейным кодом, предусматривающим введение единиц в исходный поток битов на передающем конце и устранение их при приеме.

Схема формирователя NRZI кода реализованная в среде NI Multisim представлена на рис. 2.7.

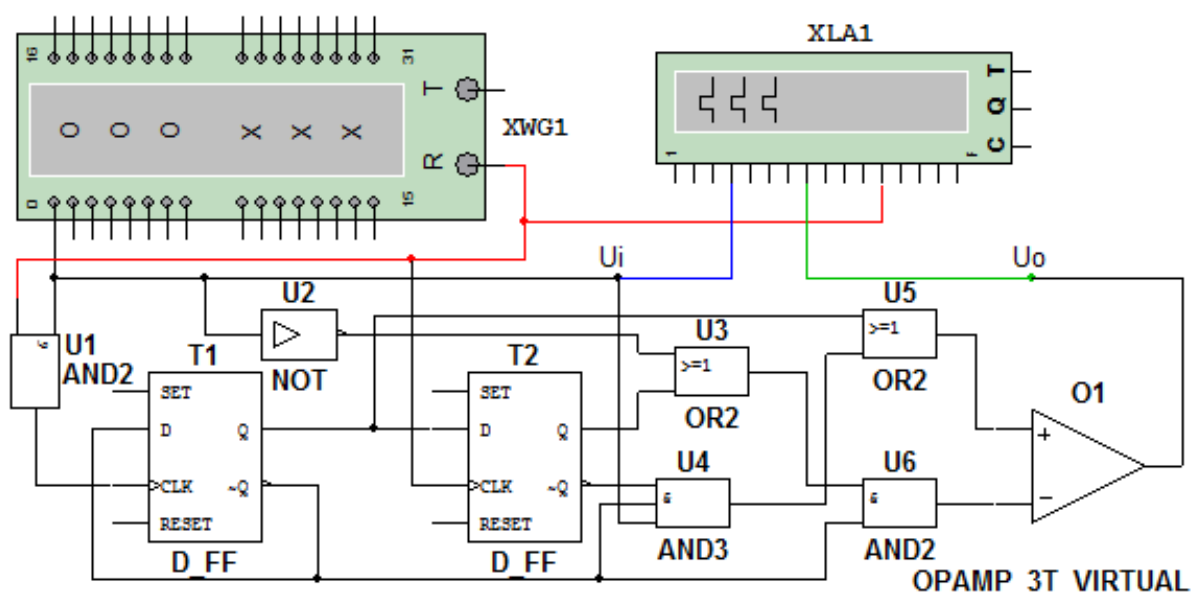


Рисунок 2.7. Формирователь биполярного кода NRZI

Схема формирователя NRZI кода содержит два D-триггера, один из которых (T1) работает в счетном режиме, фиксируя переходы 0 – 1 и 1 – 1, а другой – в качестве ячейки памяти, тактируемой импульсами синхронизации с выхода Clk генератора слова (рис. 2.8) с настройками подобными предыдущей схеме. Операционный усилитель O1 выполняет те же функции, что и на рис. 2.1, и имеет аналогичные настройки.

При нулевом входном бите (такт 0) формируемый на выходе элемента НЕ (NOT) U2 сигнал 1 через элемент ИЛИ (OR) U3 поступает на вход И (AND) U6. Поскольку T1 находится в нулевом состоянии ( $Q' = 1$ ), то сигнал 1 с выхода U6

поступает на инвертирующий вход ОУ О1, в результате чего на его выходе формируется  $U_o = -5 \text{ В}$  (рис. 2.9).

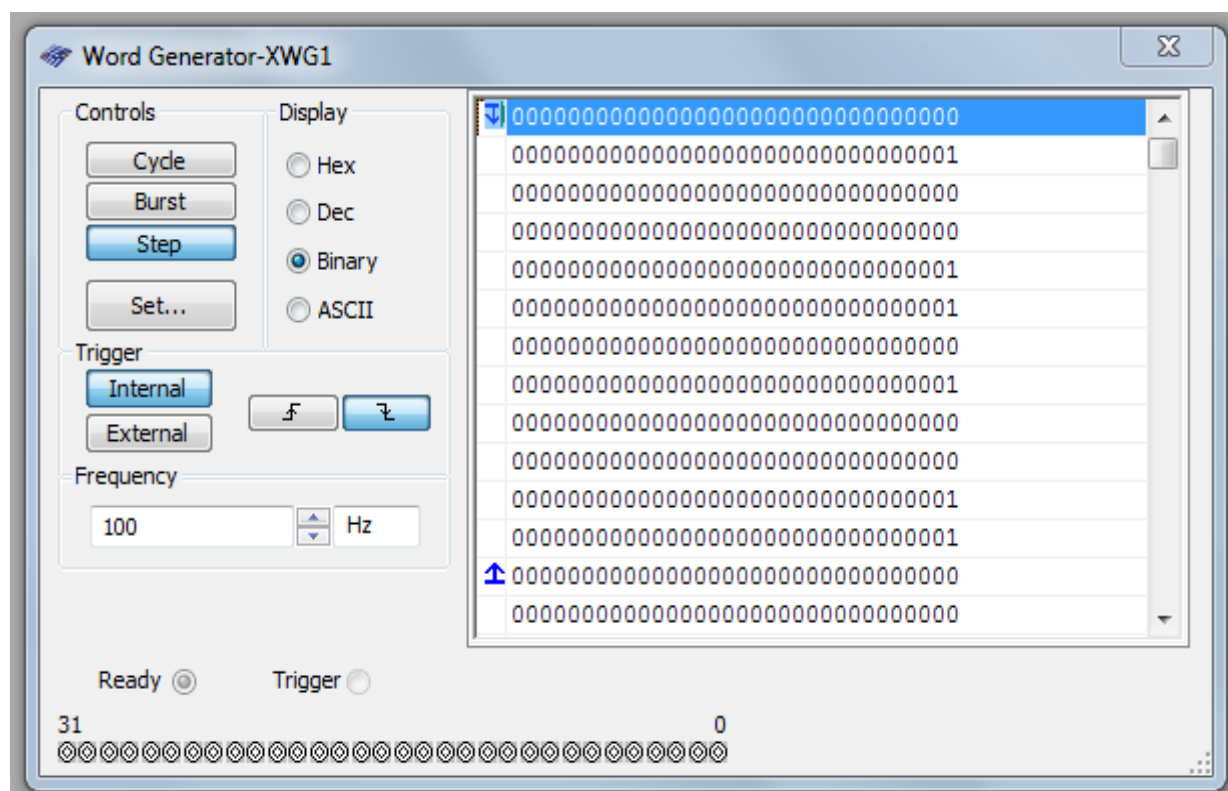


Рисунок 2.8. Окно генератора слова формирователя биполярного кода NRZI

Во время первого такта на вход  $U_i$  поступает бит 1, переводящий триггер  $T1$  в единичное состояние, которое запоминается в  $T2$ . Сигнал  $Q = 1$  с триггера  $T1$  через ИЛИ (OR)  $U5$  поступает на неинвертирующий вход ОУ и на его выходе формируется сигнал 1 (+5 В). В тактах 2 и 3 входные биты нулевые, поэтому состояние триггера  $T1$  не меняется и на выходе ОУ в этих тактах формируются 1.

В такте 2 триггер  $T2$  переводится в единичное состояние и его выходной сигнал  $Q = 1$  через ИЛИ (OR)  $U3$  поступает на И (AND)  $U6$ . В такте 4 входной бит 1 переводит  $T1$  в нулевое состояние и его выходной сигнал  $Q' = 1$  через  $U6$  поступает на ОУ, в результате чего формируется  $U_o = -5 \text{ В}$ . В пятом такте триггер  $T2$  переводится в нулевое состояние, в результате через  $U3$  блокируется  $U6$ ; в этом же такте входной бит 1 переводит  $T1$  в единичное состояние и его выходной сигнал  $Q = 1$ , поступая через  $U5$  на ОУ, формирует  $U_o = 5 \text{ В}$ .

В других тактах работы кодера процесс формирования выходной последовательности  $U_o$  происходит аналогично описанному.

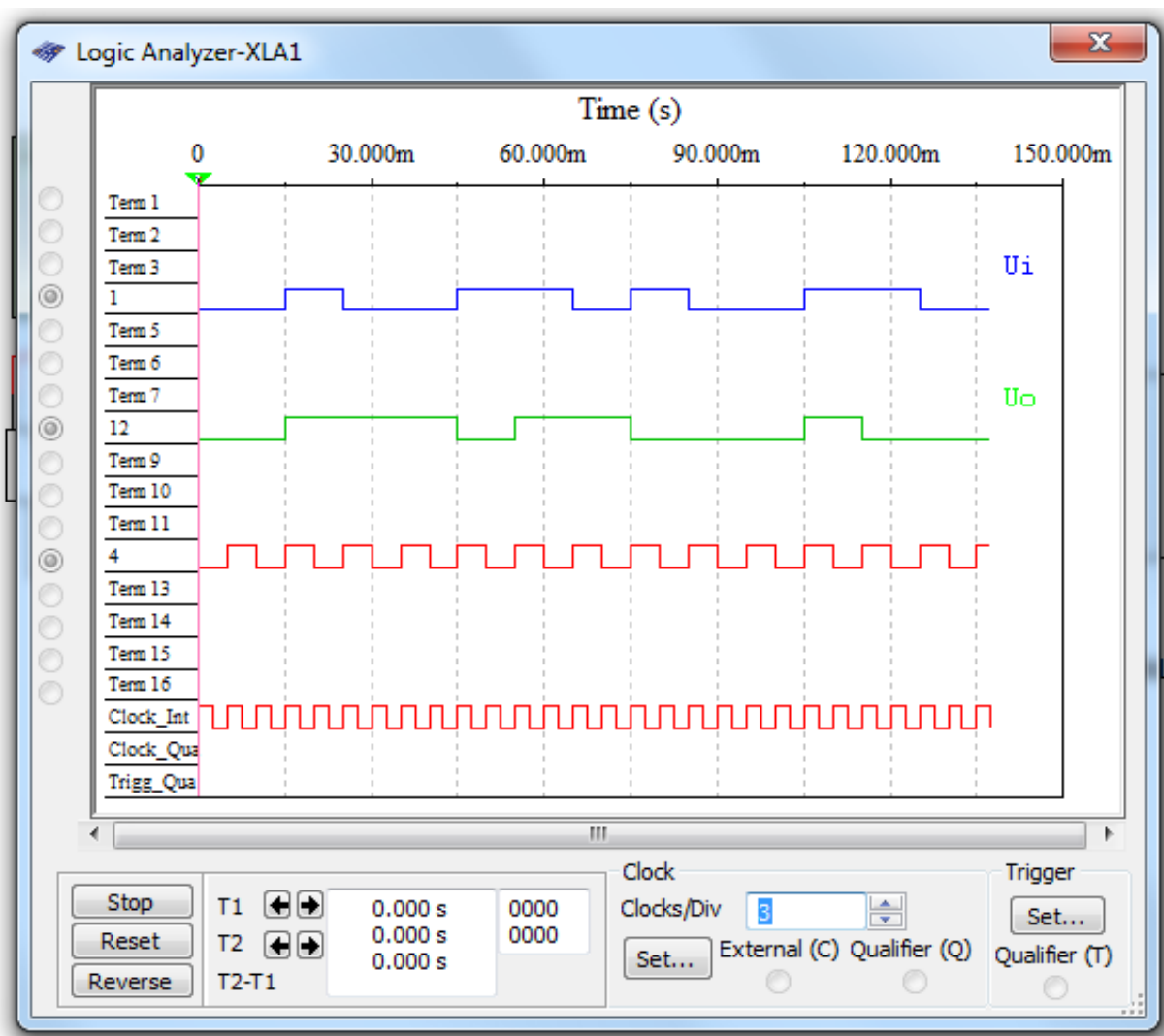


Рисунок 2.9 – Осциллограммы формирователя биполярного кода NRZI

Из других биполярных кодов получили известность следующие типы.

**AMI** (Alternate Mark Inversion) – код с чередованием полярности единиц; в этом коде нули передаются нулевыми уровнями напряжения, а единицы – чередующимся изменением полярности напряжения. Чередование полярностей сигнала при передаче единиц гарантирует отсутствие постоянной составляющей, однако проблема синхронизации остается, особенно при длинных последовательностях нулевых битов, поскольку при этом сигнал остается на нулевом уровне.

**Millar code** – код Миллара: единицы кодовой последовательности представляются в середине битового интервала переходом низкий (отрицательный) уровень – высокий (положительный) уровень; одиночные нулевые биты (например, в комбинации 101) своей полярности внутри интервала не изменяют, при нескольких нулевых битах (например, в комбинации 10001) переход полярности выполняется между каждой парой таких бит. Достоинством такого кода является снижение средней скорости передачи, однако тот факт, что переходы могут происходить как в середине битовых интервалов, так и на их концах, делает код более сложным для обнаружения. Кроме того, фазовый сдвиг сигнала на половину битового интервала может заканчиваться записями ложных битов.

**CMI (Coded Mark Inversion)** – кодирование с инверсией кодовых посылок: нулевой бит передается в середине его интервала переходом низкий (отрицательный) уровень – высокий (положительный) уровень, а единица – чередующимися переходами в течение битового интервала с верхнего уровня на нижний и обратно, т. е. этот код напоминает инверсный вариант кода Миллара.

**HDB3 (High Density Bipolar 3)** – биполярный код 3 высокой плотности, в котором 4-элементный ряд кодовых нулей заменяется специальными кодами 000V или V00V, где V – нормальный АМІ-импульс, а V – «импульс-нарушитель», который имеет ту же полярность, что и предыдущий импульс. Код 000V используется, если имеется нечетное число стандартных АМІ-импульсов с момента последнего нарушения, а V00V – если их число четное. Отличительной особенностью HDB3-кода является достижение минимального значения постоянной составляющей.

## 2.5. Многоуровневые коды

Такие коды при заданной длительности  $\tau_m$  сигналов позволяют увеличить битовую скорость передачи  $R = \log m / \tau_m$  при заметно меньшей сигнальной,

определяющей полосу пропускания канала  $F = 1/2 \tau_m$ . Естественно, требование надежного выделения сигналов синхронизации и в этом случае остается не менее важным. Приведем краткие характеристики некоторых многоуровневых кодов [9].

**MLT-3** (Multi Level Transition-3) – многоуровневый троичный или квазитроичный код с чередованием полярности единиц. Сигнал в таком коде меняется от  $-1$  до  $0$ , затем от  $0$  до  $+1$ , потом в обратном направлении – от  $+1$  до  $0$  и от  $0$  до  $-1$ . Каждый единичный бит исходного кода представляется переходом к следующему уровню сигнала, а нулевой бит – просто нулевым уровнем. Код используется в сетях Ethernet, работающих на скоростях порядка 100 Мбит/с, и FDDI-сетях, работающих с такими же скоростями на проводной (не оптоволоконной) линии связи.

**2B1Q** – код с 4-уровневым сигналом; каждые два последовательных бита (2B) передаются за один битовый интервал сигнала, который может иметь четыре состояния (1Q). Паре 00 соответствует потенциал  $-2,5$  В, 01 соответствует  $-0,833$  В, 11 –  $+0,833$  В, 10 –  $+2,5$  В. Двухразрядные последовательности 00, 01, 11, 10 называются кодами Грея (Gray encoding). У данного кода сигнальная скорость в два раза ниже, чем в NRZ и AMI кодах, а спектр сигнала в два раза уже. Поэтому с помощью 2B1Q кода можно по одной и той же линии предавать данные в два раза быстрее. Однако реализация этого метода требует более мощного передатчика и более сложного приемника, который должен различать не два уровня, а четыре.

**4B3T** (4 Binary 3 Ternary) – код, где группа из 4-х двоичных символов преобразуется в соответствующую группу из 3-х троичных символов. 4 бита исходного кода с помощью кодирующей таблицы представляются тремя уровнями:  $+V$ ,  $0$  и  $-V$ . Такая схема кодирования рекомендована для широкополосных систем стандарта IEEE 802.3 CSMA/CD; по сравнению с манчестерским кодированием она позволяет более чем в 2,6 раза повысить пропускную способность канала при одинаковой битовой скорости передачи.



**4В/5В** – код, в котором биты передаваемого сигнала сначала группируются в 4-битовые слова, после чего кодер преобразует каждое из 16 возможных 4-битовых слов в 5-битовые по принципу поддержания достаточного числа переходов между нулями и единицами. Это значит, что если в исходной последовательности из четырех бит имеется 16 различных комбинаций нулей и единиц, то в группе из пяти бит таких комбинаций будет уже 32. Из этих 32-х можно выбрать 16 комбинаций с нужным чередованием 1 и 0, а остальные 16 сделать запрещенными. При отсутствии информационных сигналов передатчик продолжает передачу резервных 5-битовых слов, называемых «пустыми символами», чтобы обозначить отсутствие цифрового сигнала и обеспечить синхронизацию тактового генератора приемника. Код 4В/5В используется в сетях Ethernet при скорости 100 Мбит/с и проводных FDDI-сетях в качестве дополнительного к MLT-3.

В некоторых случаях применяется код 6В/8В, в котором биты передаваемого сигнала сначала группируются в 6-битовые слова, после чего кодер преобразует их в 8-битовые комбинации по тому же принципу поддержания достаточного числа переходов между нулями и единицами, что и в коде 4В/5В. Однако в 6В/8В выбор нужных комбинаций заметно шире – 64 из 256 возможных. В гигабитной сети Ethernet стандарта 1000Base-X (случай оптоволоконной линии связи) используется код 8В/10Т, также аналогичный по алгоритму формирования коду 4В/5В.

**РАМ-5** – система кодирования с использованием пяти уровней:  $-2$ ,  $-1$ ,  $0$ ,  $+1$ ,  $+2$ ; применяется в гигабитной сети Ethernet (1000Base-T) под названием 4D/РАМ-5, где цифра 4 означает, что передача ведется одновременно по четырем витым парам. В 4D/РАМ-5 уровням  $+2/-2$  соответствуют напряжения  $+1/-1$  В, а уровням  $+1/-1$  – напряжения  $+0,5/-0,5$  В.

## 2.6. Кодирование скремблированием

Применение в технике связи сложных сигналов обеспечивает высокую достоверность передачи цифровой информации, помехоустойчивость к

организованным помехам и эффективное использование выделенного диапазона частот [9]. Кроме того, применение сложных сигналов обеспечивает равномерность спектра и более надежное выделение сигналов тактовой частоты в приемнике непосредственно из принимаемого сигнала, а также позволяет снизить взаимное влияния каналов в случае многоканальной связи.

Аппаратная реализация отмеченных достоинств достигается использованием различных аппаратно-программных и, в частности, **скремблирования** (от англ. *scramble* – перемешивание), при котором сигналу придаются характеристики псевдослучайного с равномерным спектром. Применение скремблирования позволяет уменьшить влияние структуры исходного цифрового сигнала на спектр сигнала в линии связи, что облегчает декодеру выделение несущей частоты и декодирование.

Для скремблирования, как правило, используются М-последовательности, формируемые (генерируемые) двоичным регистром сдвига с логической обратной связью. Такие устройства называются скремблерами; в качестве логической обратной связи в них используются элементы ИСКЛЮЧАЮЩЕЕ ИЛИ. При этом скремблирование – суммирование по модулю 2 исходного  $U_i$  (см. рис. 2.10) и псевдослучайного двоичного  $U_b$  сигналов – производится на передающей стороне.

На приёмной стороне осуществляется обратное преобразование – выделение из принятой последовательности  $U_s$  исходной информационной последовательности  $U_d$ , которая в случае правильного приема идентична последовательности  $U_i$  ( $U_d \equiv U_i$ ). Данная операция выполняется дескремблером, аналогичным по схеме скремблеру.

На практике находят применение два типа скремблеров – дескремблеров (СД): самосинхронизирующиеся и аддитивные (с начальной установкой состояния регистров). В самосинхронизирующихся СД (рис. 2.10 а) на передающей и приемной стороне используется одинаковый по составу набор из регистра Registr и элементов ИСКЛЮЧАЮЩЕЕ ИЛИ. При этом скремблер управляется самой скремблируемой последовательностью  $U_i$ , а дескремблер –

скремблированной последовательностью  $U_s$ , однозначно связанной с  $U_i$ . Следовательно, в данном случае не требуется специальной установки состояний скремблера и дескремблера, поскольку они оказываются идентичными в результате записи в их регистры сдвига скремблированной последовательности.

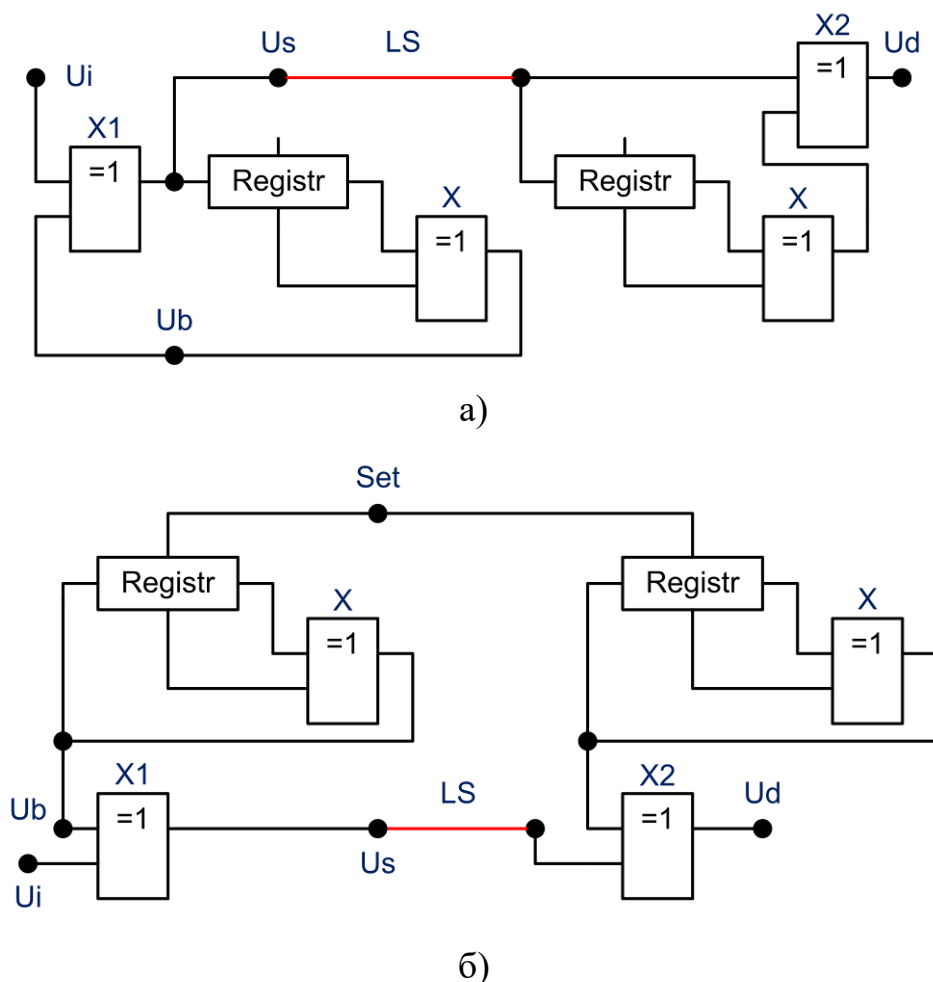


Рисунок 2.10. Структурные схемы системы скремблер – дескремблер:  
а) самосинхронизирующий тип; б) аддитивный тип

При потере синхронизма между скремблером и дескремблером время ее восстановления не превышает числа тактов, равного числу ячеек регистра Registr. Одним из недостатков самосинхронизирующегося СД является так называемый эффект размножения ошибок, который зависит от числа  $a$  обратных связей (для схем на рис. 2.10  $a = 1$ ); в большинстве случаев  $a = 2$ . Второй недостаток таких СД связан с возможностью появления на его входе так называемых «критических ситуаций», когда выходная последовательность

приобретает периодический характер с периодом, меньшим длины  $M$ . Для предотвращения этого в скремблере и дескремблере предусматриваются специальные дополнительные схемы контроля, которые выявляют периодичность элементов на входе и нарушают ее.

Недостатки, присущие самосинхронизирующимся СД, практически отсутствуют при аддитивном скремблировании (рис. 2.10 б). Однако в этом случае требуется предварительная и идентичная установка состояний регистров скремблера и дескремблера (вход Set). Как и в самосинхронизирующемся СД, суммирование сигналов на входе производится элементом XOR, но результирующий сигнал  $U_s$  не поступает на вход регистра. Аналогичным образом производится восстановление сигнала в дескремблере, т. е. скремблированная последовательность также не проходит через регистр сдвига, поэтому размножения ошибок не происходит. Поскольку суммируемые в скремблере сигналы в этом случае независимы, то и критические ситуации исключаются. Отсутствие эффекта размножения ошибок и необходимость специальной защиты от нежелательных ситуаций делают способ аддитивного скремблирования предпочтительнее и экономически эффективнее, если не учитывать затрат на решение задачи взаимной синхронизации пары скремблер – дескремблер.

Перейдем к рассмотрению модели самосинхронизирующегося скремблера, выполненной в среде NI Multisim (рис. 2.11). Как видно из данной схемы, 4-разрядный регистр сдвига составлен из D-триггеров  $U1 - U4$ , тактируемых сигналами синхронизации  $U_t$  с выхода  $R$  генератора слова, лицевая панель которого показана на рис. 2.12.

Входная (информационная) битовая последовательность  $U_i$  в виде кода 101 с нулевого выхода генератора через буфер  $U7$  поступает на один вход элемента ИСКЛЮЧАЮЩЕЕ ИЛИ (EOR)  $U6$ . На второй вход элемента  $U6$  поступают сигналы  $U_b$  обратной связи с выхода элемента EOR  $U5$ , подключенного к единичным выходам триггеров  $U3, U4$ .

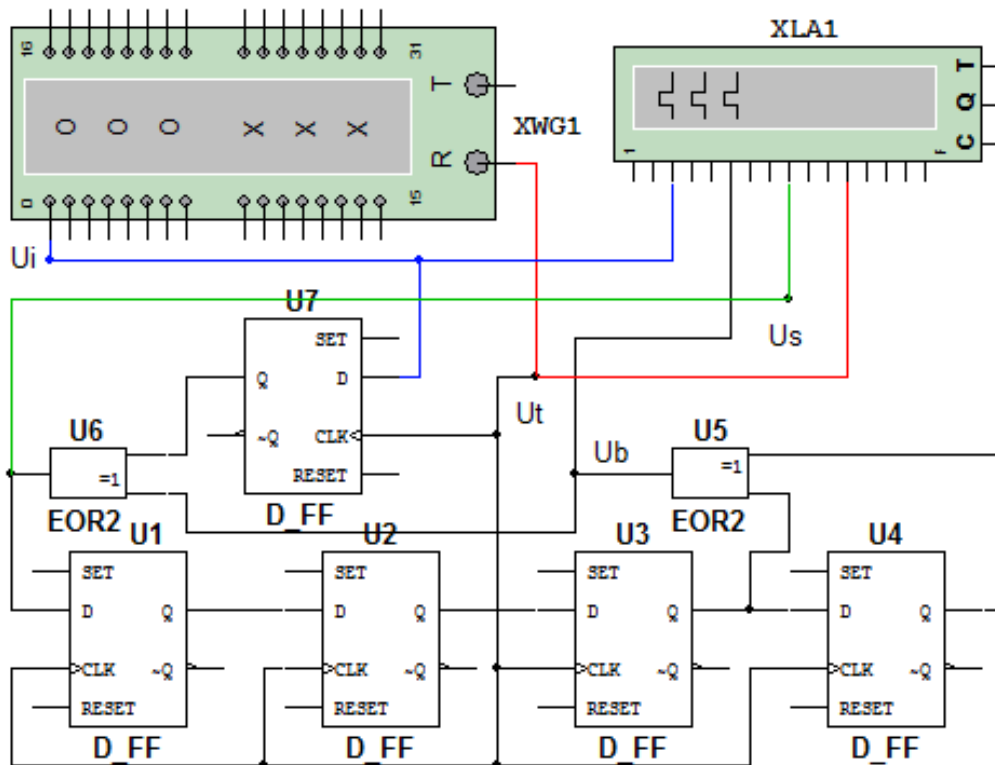


Рисунок 2.11. Структурная схема модели скремблера

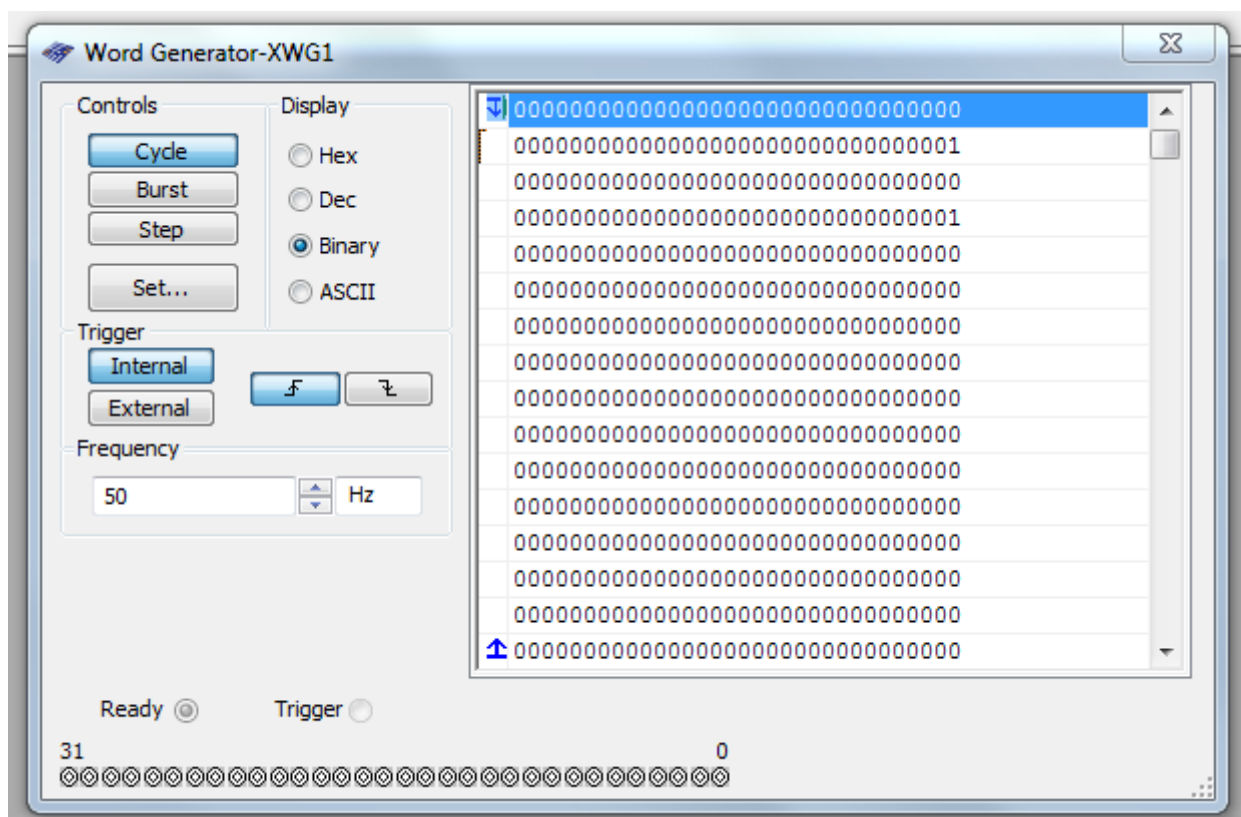


Рисунок 2.12. Окно генератора слова модели скремблера

Буфер U7 организован на D-триггере и используется в данном случае для временного согласования входных сигналов  $U_i$  и сигналов обратной связи  $U_b$  (задерживает генерацию последовательности  $U_b$  на один такт). Выходной сигнал скремблера находится по формуле

$$U_s(t) = U_i(t) \oplus U_b(t - 1) = U_i(t) \oplus Q_3(t - 1) \oplus Q_4(t - 1).$$

На первом такте (позиция 1) выходной сигнал скремблера  $U_{s1} = 1 \oplus 0 \oplus 0 = 1$ ; при этом в триггер U1 записывается 1. На втором такте  $U_{s2} = 0 \oplus 0 \oplus 0 = 0$ ; при этом U1 обнуляется, а в U2 записывается 1. На третьем такте  $U_{s3} = 1 \oplus 0 \oplus 0 = 1$ ; при этом U2 обнуляется, а в U1 и U3 записываются 1. На четвертом такте  $U_{s4} = 0 \oplus 1 \oplus 0 = 1$ ; при этом U3 обнуляется, а в U1, U2 и U4 записываются 1. На пятом такте  $U_{s5} = 0 \oplus 0 \oplus 1 = 1$ ; при этом U4 обнуляется, а в U1, U2 и U3 записываются 1. На шестом такте  $U_{s6} = 0 \oplus 1 \oplus 0 = 1$ ; при этом в U1, U2, U3 и U4 записываются 1. На седьмом такте  $U_{s7} = 0 \oplus 1 \oplus 1 = 0$ ; при этом U1 обнуляется, а в U2, U3 и U4 записываются 1. На восьмом такте  $U_{s8} = 0 \oplus 1 \oplus 1 = 0$ ; при этом U1, U2 обнуляются, а в U3 и U4 записываются 1. Далее работа скремблера продолжается и за 15 тактов работы формирует код 101111000100110, что и отображается осциллограммой  $U_s$  на рис. 2.13.

Из сравнения осциллограмм  $U_i$  и  $U_s$  видно, что коэффициент заполнения (отношение суммарной длительности импульсов к суммарной длительности пауз) в  $U_s$  больше, чем в  $U_i$ , а это значит, что скремблированная последовательность  $U_s$  «ближе» к гармонической (т. е. имеет более равномерный спектр), а наличие большего количества «холостых» импульсов способствует более эффективному выделению сигналов синхронизации.

Приведенные результаты моделирования касаются особенностей работы скремблера на передающей стороне. На приёмной стороне дескремблер из сигнала  $U_s$  выделяет последовательность  $U_d$ , идентичную информационной последовательности  $U_i$ . Схема дескремблера приведена на рис. 2.14 и аналогична рассмотренной схеме скремблера.

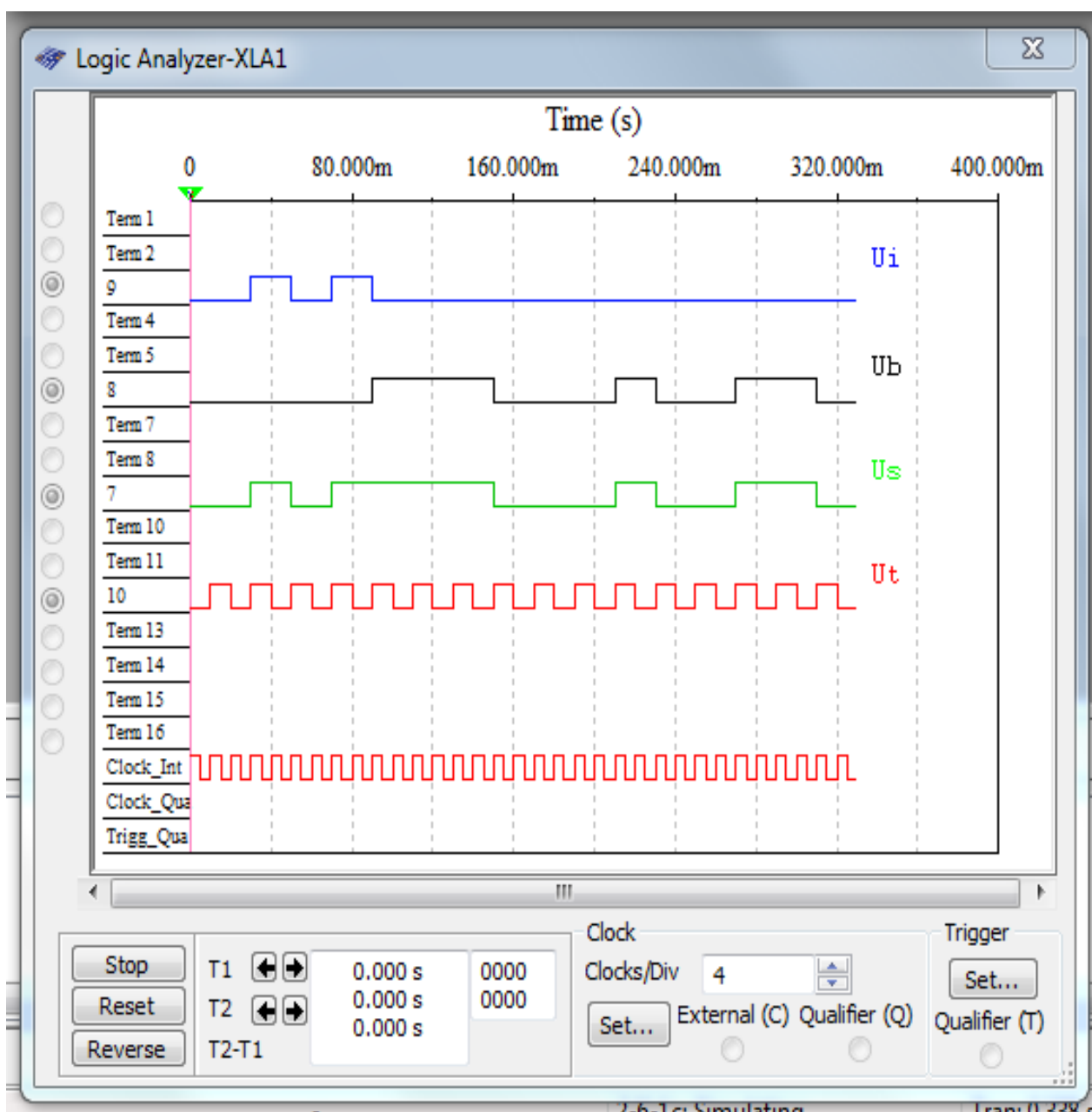


Рисунок 2.13. Результаты моделирования скремблера

В этом случае регистрируемая осциллографом последовательность на выходе дескремблера (без учета временных сдвигов)  $U_d = U_s \oplus Q_3 \oplus Q_4 = U_i \oplus Q_3 \oplus Q_3 \oplus Q_4 \oplus Q_4 = U_i$ , т. е. на выходе дескремблера исходная информационная последовательность полностью восстанавливается, что и подтверждается осциллограммами на рис. 2.16. Объясняется это тем, что слагаемые  $(Q_3 \oplus Q_3)$  и  $(Q_4 \oplus Q_4)$  при любом (но одинаковом) значении их составляющих всегда дают 0 (используются свойства элемента ИСКЛЮЧАЮЩЕЕ ИЛИ:  $1 \oplus 1 = 0$  и  $0 \oplus 0 = 0$ ).

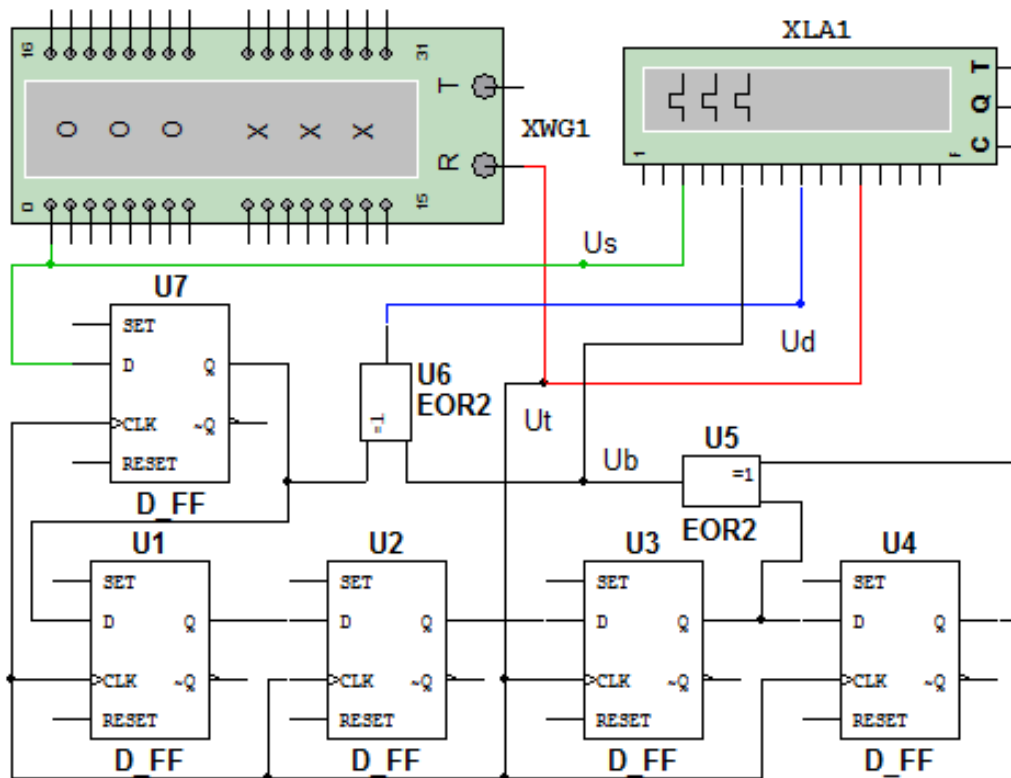


Рисунок 2.14. Структурная схема модели самосинхронизирующегося дескремлера

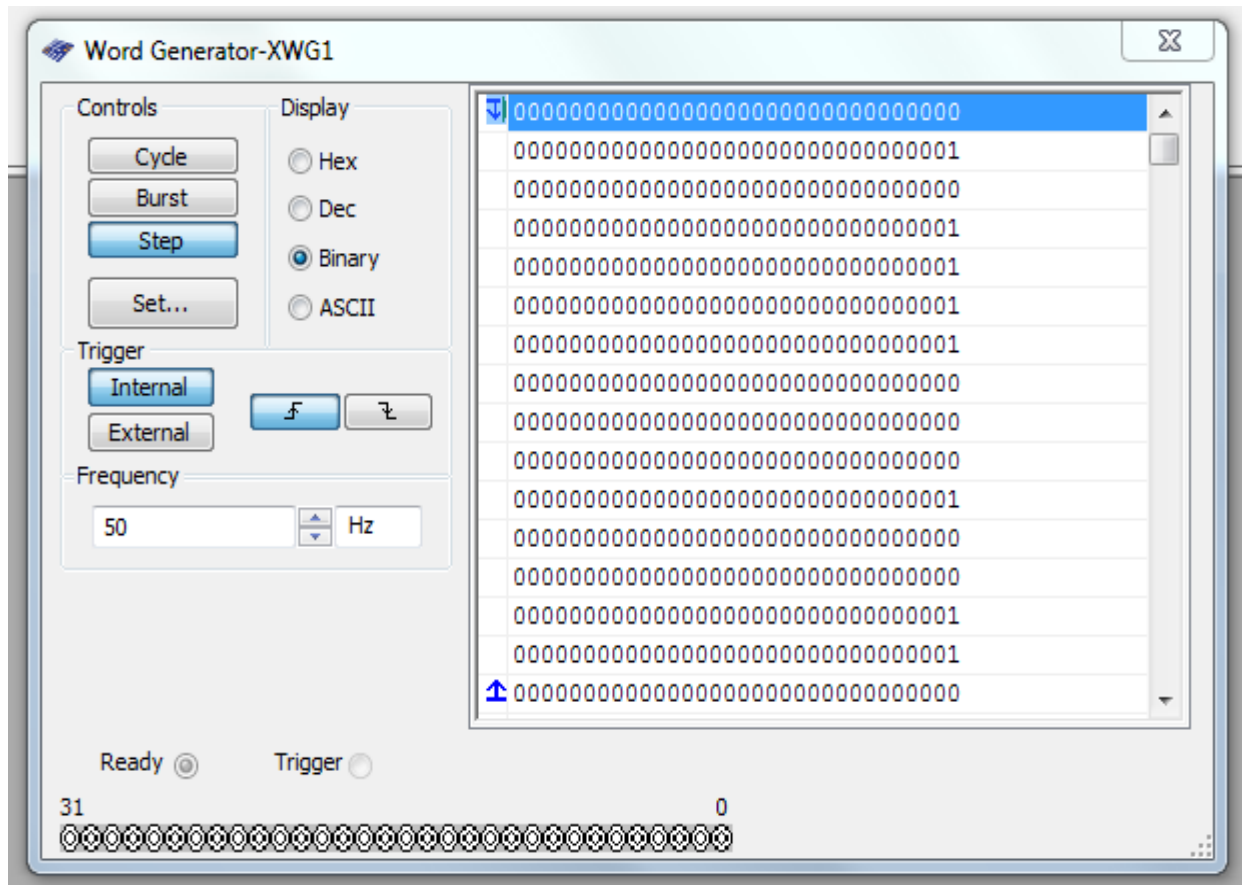


Рисунок 2.15. Окно генератора слова модели дескремлера



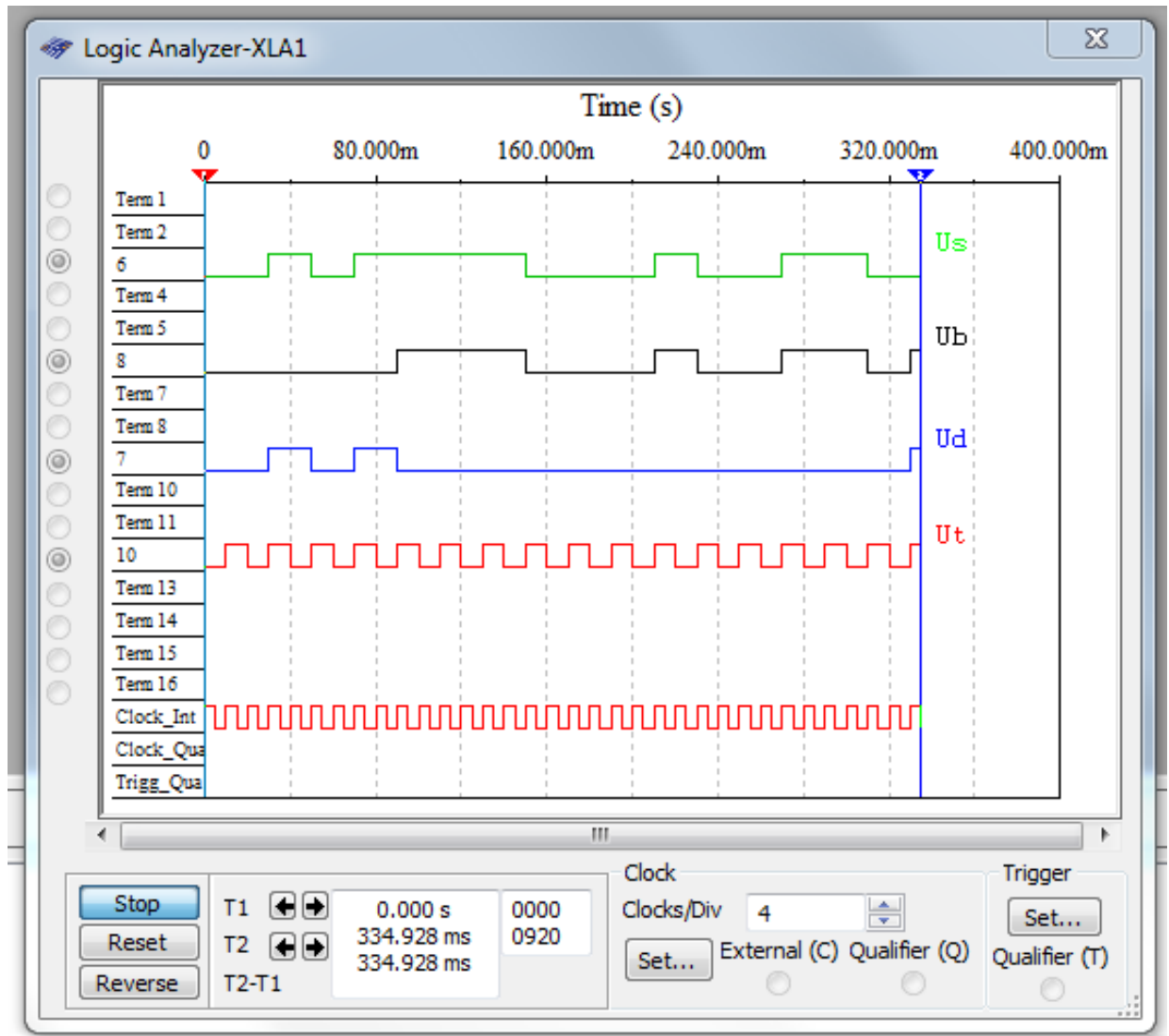


Рисунок 2.16. Результаты моделирования дескремблера

## 2.7. Кодирование перемежением

В некоторых системах передачи информации, например с кодовым разделением каналов (CDMA), в зависимости от условий эксплуатации и вида информационных сигналов, используются различные сочетания кодирующих устройств [9].

Так, для передачи речевого сигнала, когда вероятность ошибки на бит может быть достаточно большой, обычно используется сверточное кодирование. При передаче данных, когда требуется более высокая надежность, применяется каскадное кодирование, при котором первым (внешним) обычно

является кодер Рида – Соломона, а вторым (внутренним) – сверточный. Однако в случае каналов с высоким уровнем помех и замираний, приводящих к появлению пакетов ошибок, эффективность применения корректирующих кодов заметно снижается. Попытки создания специальных кодов, способных исправлять подобные серии ошибок, предпринимались неоднократно (коды Хагельбаргера, Файера и др.), однако из-за огромной вычислительной сложности, экспоненциально возрастающей с увеличением длины пакета ошибок, они не нашли практического применения.

Из этой ситуации был найден достаточно простой выход: выполнение операции декодирования в два этапа позволяет почти полностью избавиться от помех. При этом на первом этапе производится декорреляция пакетов ошибок, в результате которой они преобразуются в группу случайных одиночных ошибок. На втором этапе сигнал обрабатывается с помощью классических методов борьбы со случайными ошибками (сверточные коды, турбокоды), что приводит к их полному подавлению.

Для борьбы с замираниями и связанными с ними пакетами ошибок используется процедура перемежения. Она состоит в перестановке (перемежении) символов кодированной последовательности до ее модуляции и восстановлении (деперемежении) исходной последовательности после демодуляции. Перестановка позволяет так разнести рядом стоящие символы, чтобы они оказались разделены группой других символов, передаваемых в том же блоке данных. Данная операция не вносит избыточности, а только изменяет порядок следования символов или бит. Однако чем больше глубина перемежения (максимальное расстояние, на которое разносятся соседние символы входной последовательности), тем больше задержка.

Принцип работы перемежителя заключается в том, что исходная кодовая последовательность из  $L$  символов преобразуется в матрицу размером  $L = [R \times C]$ , где  $R$  – число строк, а  $C$  – число столбцов. Операция блочного перемежения заключается в последовательной построчной записи входных данных и считывании этой информации по столбцам. В результате порядок

следования символов в выходной последовательности изменяется, например  $k$ -я строка матрицы будет выглядеть как  $\{k, C + k, 2C + k, (N - 1)C + k\}$ , откуда видно, что два любых соседних символа входной последовательности будут разнесены в линии связи на  $C - 1$  символов.

Максимальное расстояние, на которое разносятся соседние символы, определяется как  $d = T \times C$ , где  $T$  – длительность символа. Если время, в течение которого происходил сбой сигнала, меньше глубины перемежения, любой пакет ошибок будет преобразован в группу из  $C$  одиночных ошибок, которые затем легко устраняются сверточным кодером. В общем случае выбор глубины перемежения зависит от двух факторов. С одной стороны, чем больше расстояние между соседними символами, тем большей длины пакет ошибок может быть исправлен. С другой стороны, чем больше глубина перемежения, тем сложнее аппаратно-программная реализация оборудования и больше задержка сигнала.

Заметим, что различают внешнее и внутреннее перемежение. Внутреннее перемежение обычно производится в пределах одного кадра, а внешнее – в пределах интервала из нескольких кадров.

Схема для исследования процесса перемежения/деперемежения реализованная в среде VisSim/Comm [17, 21] представлена на рис. 2.12.

Схема исследования процесса перемежения/деперемежения содержит:

- источник равномерно распределенных случайных символов Random Sym с параметрами: число символов в последовательности (Number of Symbols) – 2 (чисто бинарная последовательность); скорость следования символов (Symbol Rate) – 5 симв/с; характеристики выходного сигнала источника представлены на рис. 2.12 осциллограммой R и спектром Spectrum R;

- блоки перемежения/деперемежения (Interleaver/Deinterleaver) с параметрами (для обоих блоков): число строк матрицы  $R = 2$ ; число столбцов  $C = 2$  (приведенные значения являются минимально возможными);

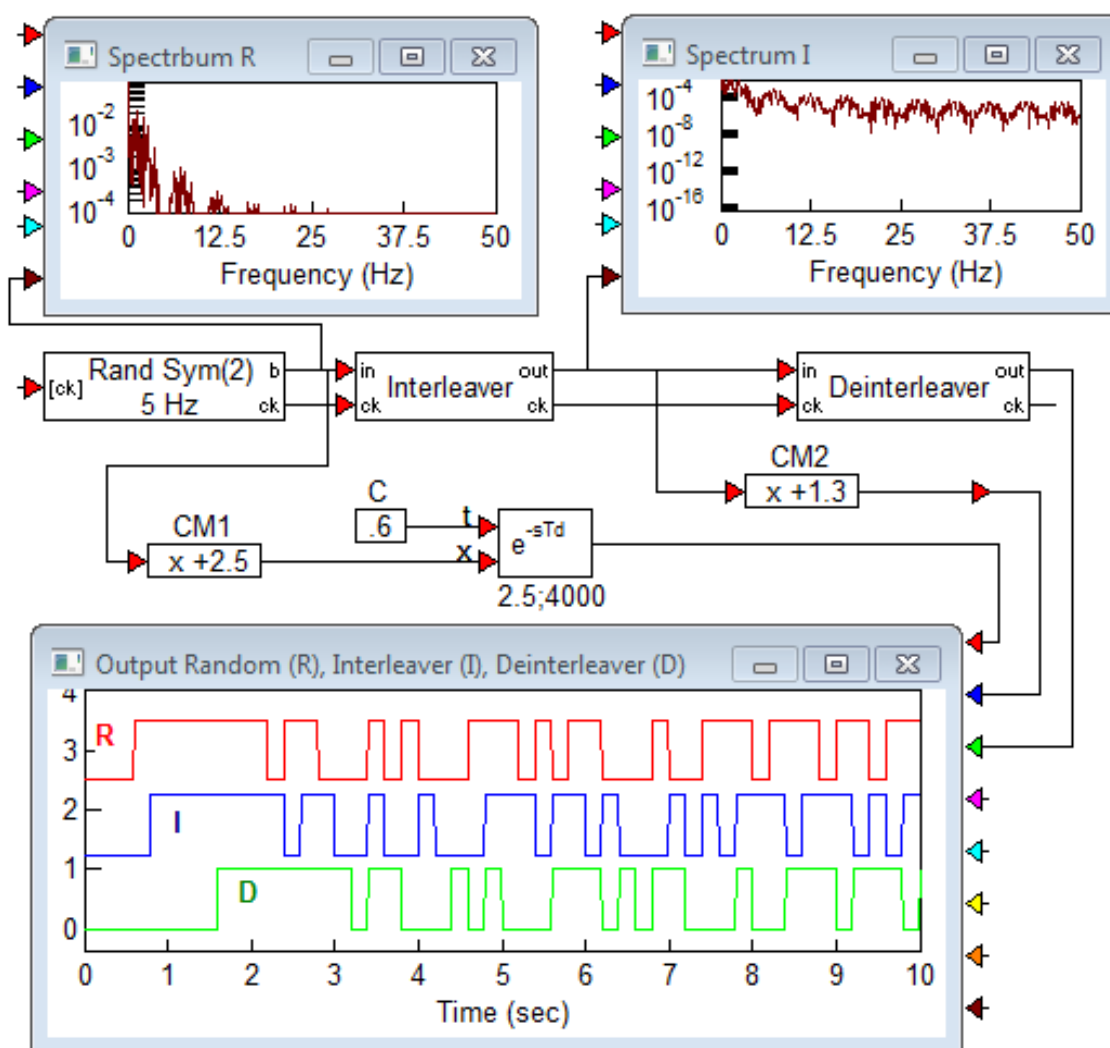


Рисунок 2.12. Схема кодирования перемежением

- характеристики выходных сигналов представлены на рис. 2.12 осциллограммой D для деперемежителя, осциллограммой I и спектром Spectrum I для перемежителя;
- блоки CM1, CM2 смещения осциллограмм сигналов; в качестве таких блоков использован полиномиальный источник Polynomial с различными значениями коэффициентов  $x^0$  Coefficient, значения которых индицируются в значке блока при включенной опции Show Formula (показать формулу);
- блок  $e^{-sTd}$  задержки входного сигнала  $x$  на время, задаваемое по входу  $t$  источником постоянной величины  $C$ , равном в данном случае 0,6 с; это время задержки сигнала в перемежителе и введено для удобства сравнения осциллограмм R и I; другой параметр блока – начальное напряжение (Initial

Condition (IC)) – выбрано равным напряжению смещения осциллограммы R, т. е. 2,5 В, иначе искажается ее начальный участок.

Анализ полученных результатов моделирования позволяет сделать следующие выводы:

- перемежение приводит к выравниванию спектра сигнала в линии связи, что является показателем более эффективного ее использования;
- информационная последовательность символов на выходе деперемежителя полностью совпадает с входной;
- при выбранных параметрах перемежителя задержка его выходного сигнала равна  $\Delta T_i = (C + 1) \times T = (2 + 1) \times 0,2 = 0,6 \text{ с}$ , где  $T = 0,2 \text{ с}$  – длительность символа (измерения проводятся в режиме максимального окна графопостроителя, включаемом кнопкой Ц);
- при тех же параметрах деперемежителя задержка его выходного сигнала относительно входного равна  $\Delta T_d = (2C + 1) \times T = (2 \times 2 + 1) \times 0,2 = 1 \text{ с}$ , т. е. суммарная задержка составляет  $(3C + 2) \times T = 1,6 \text{ с}$ .

## **2.8. Лабораторная работа 2. Исследование особенностей работы систем линейного кодирования**

### **2.8.1. Цель и содержание**

Целью лабораторной работы является приобретение студентами практических навыков исследования принципов функционирования систем линейного кодирования.

#### **Содержание**

1. Исследование модели биполярного кодера/декодера.
2. Исследование особенностей работы схемы формирователя манчестерского кода.
3. Исследование модели кодера биполярного кода NRZI.
4. Исследование моделей самосинхронизирующихся скремблера и дескремблера.

5. Исследование особенностей работы схемы кодирования перемежением.

### **2.8.2. Аппаратура и материалы**

Для изучения материалов этой лабораторной работы необходимо:

1. Компьютер под управлением операционной системы Windows XP/Vista/7 с параметрами по умолчанию.
2. Установленная инструментальная среда визуального моделирования динамических систем VisSim/Comm 7.0.
3. Установленная среда проектирования NI Multisim 10.
4. Текстовый процессор (Microsoft Word 2003/2007/2010 или его аналог), предназначенный для создания и редактирования отчета по лабораторной работе.

### **2.8.3. Порядок выполнения работы**

1. Изучение теоретического материала.
2. Выполнение задания, предлагаемого в лабораторной работе.

#### **Задание**

1. Запустить на выполнение файл 2-2-1.ms10. Провести моделирование биполярного кодера/декодера для входного кода 0101101101101. Имейте в виду, что код с выхода генератора слова выдается начиная с первого такта после начального (исходного) состояния, устанавливаемого командой Set Cursor контекстного меню на его лицевой панели. Сделать выводы.

2. Запустить на выполнение файл 2-3-1.ms10. Провести моделирование схемы формирователя манчестерского кода и проанализировать полученные результаты при входной последовательности 010000011111010.

Разработать схему декодера манчестерского кода и провести ее моделирование.

3. Запустить на выполнение файл 2-4-1.ms10. Провести моделирование и анализ работы кодера биполярного кода NRZI для входного кода 0101101101101.

4. Запустить на выполнение файл 2-6-1.ms10. Провести моделирование и анализ работы самосинхронизирующегося скремблера для входного кода 10111.

Запустить на выполнение файл 2-6-2.ms10. Провести моделирование и анализ работы самосинхронизирующегося дескремблера для выходного сигнала скремблера при информационной посылке 10111.

5. Запустить на выполнение файл 2-7-1.VSM. Провести моделирование и анализ работы схемы кодирования перемежением. Путем сравнения осциллограмм R и I проверить, что максимальное расстояние, на которое разносятся соседние символы, равно  $d = T \times C = 0,2 \times 2 = 0,4$  с.

Провести моделирование при  $R = C = 3, 4, 5$  и  $6$  (произведение  $R \times C$  может принимать значения от 4 до 32767). Проверить, выполняются ли при этих значениях соотношения  $\Delta T_i = (C + 1) \times T$  и  $\Delta T_d = (2C + 1) \times T$ .

#### 2.8.4. Вопросы для защиты работы

1. Назовите отличия униполярных кодов от биполярных.
2. Прокомментируйте особенности и области применения интерфейса RS-232.
3. Какие параметры имеет кабель для соединения двух устройств с использованием интерфейса RS-232? Какой длины он бывает?
4. Назовите длину пакета применяемого для обмена данными с использованием интерфейса RS-232.
5. Опишите принцип формирования манчестерского кода.
6. Какие переходы сигнала в манчестерском коде используются в приемнике для синхронизации?
7. Какие значения напряжения используются в качестве низкого и высокого уровней в сети Ethernet?
8. Какой недостаток манчестерского кода обусловлен тем обстоятельством, что его сигналы дважды меняют свое состояние за период тактовой частоты?

9. Поясните принципы дифференциального кодирования.
10. Разработайте схему NRZI-декодера и прокомментируйте его работу.
11. Используя описанный в тексте алгоритм АМІ-кодирования и принципы построения предыдущих схем, разработайте схему АМІ-кодера/декодера.
12. Какой положительный эффект можно получить при использовании сложных сигналов, и в частности псевдослучайных бинарных последовательностей? Назовите параметры таких последовательностей.
13. В чем заключается процесс скремблирования цифровых информационных сигналов?
14. Охарактеризуйте существующие типы скремблирующих устройств.
15. Чем объясняются такие положительные свойства скремблирования, как получение более равномерного спектра сигнала в линии связи и более быстрого восстановления синхронизации?
16. Какие требования предъявляются к дескремблеру приемника?
17. В чем заключается процедура перемежения/деперемежения?
18. В каком месте цифровой системы передачи информации ставится перемежитель и деперемежитель?



### ГЛАВА 3. МОДУЛЯТОРЫ И ДЕМОДУЛЯТОРЫ

Для передачи сообщений по каналу связи, кроме кодирования, во многих случаях требуется еще одно преобразование информационного сигнала, которое называется модуляцией на передающей стороне и демодуляцией – на приемной [9]. Особенно ярко необходимость в модуляции проявляется в радиоканалах, в которых немодулированный информационный сигнал не может передаваться в силу условий распространения радиоволн в свободном пространстве: для этого требуется высокочастотный сигнал-носитель, который с помощью антенно-фидерных устройств приемлемых размеров способен излучаться в эфир. Такими же свойствами обладает и сигнал-носитель, промодулированный информационным сигналом.

**Цифровая модуляция** – это процесс преобразования цифровых символов в сигналы, совместимые с характеристиками канала [6]. При **низкочастотной модуляции** (baseband modulation) эти сигналы обычно имеют вид импульсов заданной формы. В случае **полосовой модуляции** (bandpass modulation) импульсы заданной формы модулируют синусоиду, называемую несущей волной (carrier wave), или просто несущей (carrier); для радиопередачи на нужное расстояние несущая преобразуется в электромагнитное поле.

Полосовая модуляция (аналоговая или цифровая) – это процесс преобразования информационного сигнала в синусоидальный сигнал; при цифровой модуляции синусоида на интервале  $T$  называется цифровым символом. Синусоиды могут отличаться по амплитуде, частоте и фазе. Таким образом, полосовую модуляцию можно определить как процесс варьирования амплитуды, частоты или фазы (или их комбинаций) радиочастотной несущей согласно передаваемой информации. В общем виде несущая записывается следующим образом:

$$s(t) = A(t)\cos\theta(t), \quad (3.1)$$

где  $A(t)$  – переменная во времени амплитуда,  $\theta(t)$  – переменный во времени угол, который удобно записывать в виде

$$\theta(t) = \omega_0 t + \varphi(t), \quad (3.2)$$

так, что

$$s(t) = A(t) \cos[\omega_0 t + \varphi(t)], \quad (3.3)$$

где  $\omega$  – угловая частота несущей, а  $\varphi(t)$  – ее фаза. Частота может записываться как переменная  $f$  или как переменная  $\omega$ . В первом случае частота измеряется в герцах (Гц), а втором – в радианах в секунду (рад/с). Эти параметры связаны следующим соотношением  $\omega = 2\pi f$ .

К основным типам полосовой цифровой модуляции/демодуляции можно отнести: амплитудную манипуляцию (amplitude shift keying – ASK), частотную манипуляцию (frequency shift keying – FSK), фазовую манипуляцию (phase shift keying – PSK) и комбинации этих модуляций.

Если для детектирования сигналов приемник использует информацию о фазе несущей, процесс называется **когерентным детектированием** (coherent detection); если подобная информация не используется, процесс именуется **некогерентным детектированием** (noncoherent detection). Вообще, в цифровой связи термины «демодуляция» и «детектирование» часто используются как синонимы, хотя демодуляция делает акцент на восстановлении сигнала, а детектирование – на принятии решения относительно символического значения принятого сигнала. При идеальном когерентном детектировании приемник содержит прототипы каждого возможного сигнала. Эти сигналы-прототипы дублируют алфавит переданных сигналов по всем параметрам, даже по радиочастотной фазе. В этом случае говорят, что приемник автоматически подстраивается под фазу входного сигнала. В процессе демодуляции приемник перемножает и интегрирует входной сигнал с каждым прототипом (определяет корреляцию).

Ниже приводятся результаты испытаний моделей модуляторов реализованных в системе VisSim с использованием библиотек Comm/Modulator.

### 3.1. Амплитудная манипуляция

Первооткрывателем амплитудной модуляции (АМ) считается американский инженер Р. Фессенден (1906 год) [9]. Этот вид модуляции с 1920 года стал основным в радиовещании. Разновидность АМ – с одной боковой полосой – изобрел американский ученый Дж. Р. Карсон (1915 год). Такой вид АМ отличается почти в два раза меньшей полосой занимаемых частот канала связи и широко применяется до сих пор в системах многоканальной связи и в телевидении. Еще одна разновидность АМ – полярная модуляция – была разработана С. И. Тетельбаумом (СССР, 1939 год) и Л. Канном (США, 1961 год). Суть этого метода модуляции заключается в том, что положительная полуволна несущей частоты модулировалась по амплитуде одним сообщением, а отрицательная – другим. В СССР такой вид АМ использовался в системе стереофонического радиовещания.

Амплитудно-манипулированный сигнал (АМн – ASK), изображенный на рис. 3.1, описывается выражением

$$s_i(t) = \sqrt{\frac{2E_i(t)}{T}} \cos(\omega_0 t + \varphi) \quad 0 \leq t \leq T \quad (3.4)$$

$$i = 1, \dots, M,$$

где параметр  $E$  – это энергия символа,  $T$  – время передачи символа,  $0 < t < T$ , амплитудный член  $\sqrt{\frac{2E_i(t)}{T}}$  может принимать  $M$  дискретных значений, а фазовый член  $\varphi$  – это произвольная константа [6].

На рис. 3.1 параметр  $K$  выбран равным 2, что соответствует двум типам сигналов, амплитуды которых равны 0,5 и 1.

Передача сигналов в двухуровневой манипуляции ASK – это одна из первых форм цифровой модуляции, изобретенных для беспроводной телеграфии. В настоящее время простая схема ASK в системах цифровой связи уже не используется.

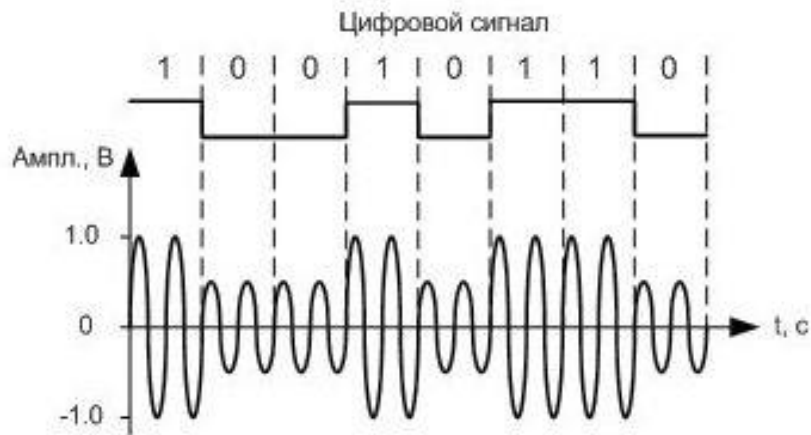


Рисунок 3.1. Сигнал амплитудной манипуляции

Перейдем к результатам моделирования системы амплитудной манипуляции с использованием ASK-модуляторов типа Real и Complex, схема включения которых содержит (рис. 3.2, 3.3): источник равномерно распределенных случайных символов Rand. Sym с частотой следования символов 5 Гц и параметром Number of Symbols определяющим число битов (уровней) в символе; источник запускающего сигнала Impulse; регистраторы D1 – D7 типа Display; графопостроители (плоттеры); преобразователи Crplx комплексных сигналов в их составляющие; блоки прямого FFT и обратного IFFT Фурье-преобразования; сумматоры. Параметры ASK-модуляторов: несущая частота – 10 Гц, начальная фаза –  $30^\circ$  (для модулятора Complex), остальные – по умолчанию.

Из рассмотрения результатов на рис. 3.2, а можно сделать следующие выводы. При  $M = 1$  на выходе sig модулятора Real будет немодулированный сигнал несущей амплитудой 1 В (см. дисплей плоттера «АМн при M=1»); при  $M = 2$  на выходе z модулятора Complex будет модулированный сигнал с максимальным размахом  $U_M = 2$  В (см. дисплей плоттера «АМн при M=2»).

Как видно из осциллограммы спектра на рис. 3.3, он содержит явно выраженный максимум на частоте несущей 10 Гц.

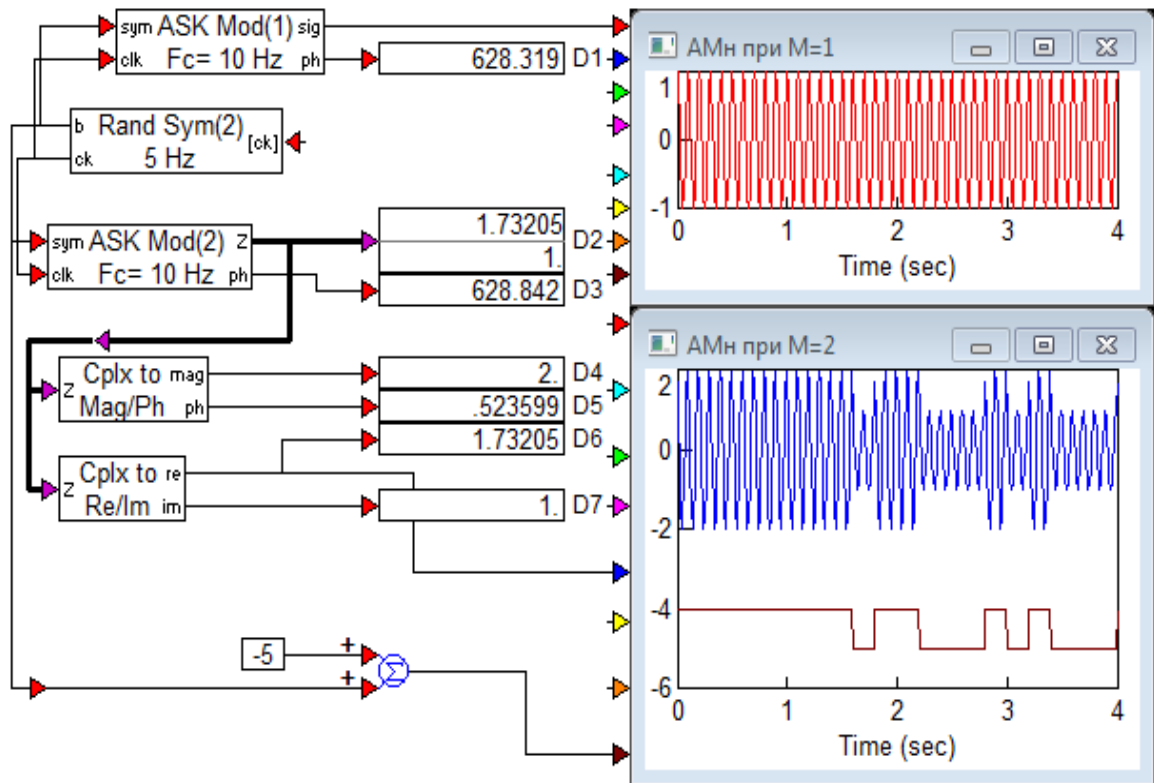


Рисунок 3.2. Схема ASK-модулятора

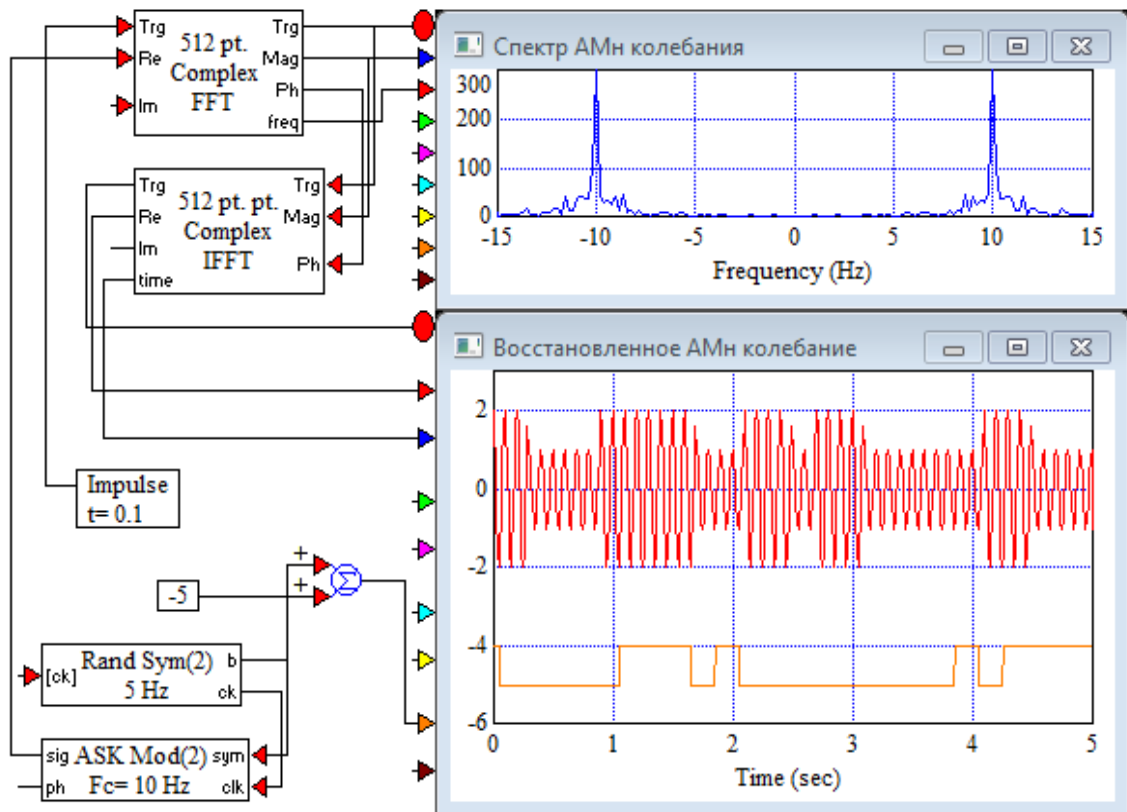


Рисунок 3.3. Схема ASK-модулятора с преобразованием Фурье

### 3.2. Частотная манипуляция

Изобретение частотной модуляции (ЧМ) относится к первым годам XX века (США, Корнелиус Д. Эрст, 1902 год) [9]. Однако в течение почти 30 лет, до работ знаменитого американского инженера Э. Х. Армстронга (1935 год), она не находила практического применения. Этот вид модуляции получил широчайшее применение в огромном числе систем связи самого различного назначения: подвижной, радиорелейной, спутниковой, в стереофоническом радиовещании.

При частотной (тональной) манипуляции (ЧМн – FSK) каждому уровню (посылке) модулирующего сигнала задается гармонический сигнал определенной частоты (тона). Первое применение ЧМн было отмечено в 1915 году (США, В. Паульсен), в дальнейшем работы по ЧМн были продолжены Э. Х. Армстронгом (1927 год) [9]. В СССР работы по многопозиционной ЧМн для телеграфии проводились академиком А. Н. Щукиным. На основе проведенных исследований им была разработана система двойного частотного телеграфирования.

Широкое распространение ЧМн нашла в модемных протоколах связи. Например, в двухканальном (дуплексном) модеме V.21 на нижнем канале (его обычно использует вызывающий модем) сигнал логической 1 передается частотой 980 Гц, а 0 – 1180 Гц. На верхнем канале (в случае «отвечающего» модема) 1 передается частотой 1650 Гц, а 0 – 1850 Гц. При этом длительность выходной посылки задается длительностью входной.

Общее аналитическое выражение для частотно-манипулированного сигнала (ЧМн – FSK) имеет следующий вид [6]:

$$s_i(t) = \sqrt{\frac{2E}{T}} \cos(\omega_i t + \varphi) \quad 0 \leq t \leq T \quad (3.5)$$

$$i = 1, \dots, M.$$

Здесь частота  $\omega_i$  может принимать  $M$  дискретных значений, а фаза  $\varphi$  является произвольной константой. Схематическое изображение ЧМн-сигнала

дано на рис. 3.4, где можно наблюдать типичное изменение частоты (тона) в моменты переходов между символами.

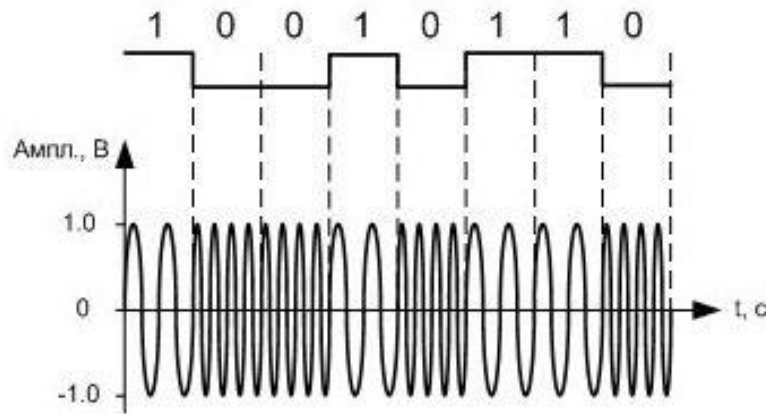


Рисунок 3.4. Сигнал частотной манипуляции

Такое поведение характерно только для частного случая FSK, называемого частотной манипуляцией без разрыва фазы (continuous-phase FSK – CPFSK). В общем случае многочастотной манипуляции (multiple frequency shift keying – MFSK) переход к другому тону может быть довольно резким, поскольку непрерывность фазы здесь не обязательна. В приведенном примере  $M = 2$ , что соответствует такому же числу типов сигналов (двоичной передаче).

Минимальная полоса частот, занимаемая ЧМн сигналом, определяется следующим выражением:

$$\Delta f = \Delta f_p + 1/\tau, \quad (3.6)$$

где  $\Delta f_p$  – интервал разноса посылок  $u_1(t)$  и  $u_2(t)$  по частоте,  $\tau$  – длительность входной посылки.

Величина  $\Delta f_p$  в среде VisSim определяется параметром Frequency Spacing модулятора FSK. Кроме FSK-модуляторов, новым компонентом на рис. 3.5, является источник прямоугольных сигналов Rect. Pulses из библиотеки Comm/Signal Sources с параметрами по умолчанию. Параметры модуляторов выбраны также по умолчанию.

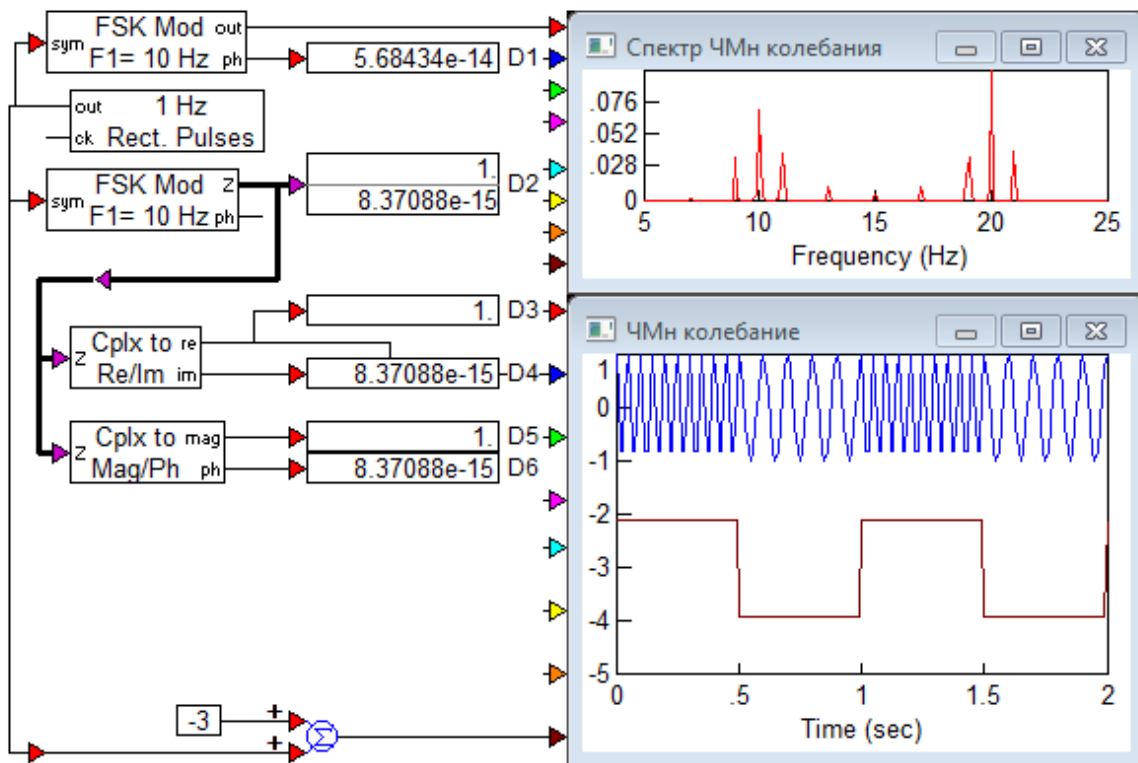


Рисунок 3.5. Схема FSK-модулятора

В результате анализа спектра сигнала (рис. 3.5) можно сделать вывод, что его ширина соответствует выражению (3.6) для случая прямоугольных сигналов  $\tau = 0,5$  с:  $\Delta f = \Delta f_p + 1/\tau = \Delta f_p + 2\Omega = 10 + 2 = 12$  Гц.

### 3.3. Фазовая манипуляция

Фазовая модуляция (ФМ), предложенная американским ученым Г. Найквистом в 1928 году, долгое время не находила применения из-за явления «обратной работы», возникающего вследствие невозможности восстановления на приеме опорного колебания, строго синфазного с несущей частотой принимаемого сигнала [9]. С 1930-х годов в течение почти 40 лет велись интенсивные научные исследования по разработке методов синхронного приема сигналов ФМ, в которых принимали активное участие ученые и инженеры Франции, СССР, США и других стран. В настоящее время этот вид модуляции широко используется в разнообразных цифровых системах передачи информации.



Фазовая манипуляция (ФМн – PSK) была разработана в начале развития программы исследования дальнего космоса; сейчас схема PSK широко используется в коммерческих и военных системах связи [6]. Фазоманипулированный сигнал имеет следующий вид

$$s_i(t) = \sqrt{\frac{2E}{T}} \cos[\omega_0 t + \varphi_i(t)] \quad 0 \leq t \leq T \quad (3.7)$$

$$i = 1, \dots, M,$$

здесь фазовый член  $\varphi_i(t)$  может принимать  $M$  дискретных значений, обычно определяемых следующим образом:

$$\varphi_i(t) = \frac{2\pi i}{M} \quad i = 1, \dots, M.$$

На рис. 3.6 приведен пример двоичной ( $M = 2$ ) фазовой манипуляции (binary PSK – BPSK).

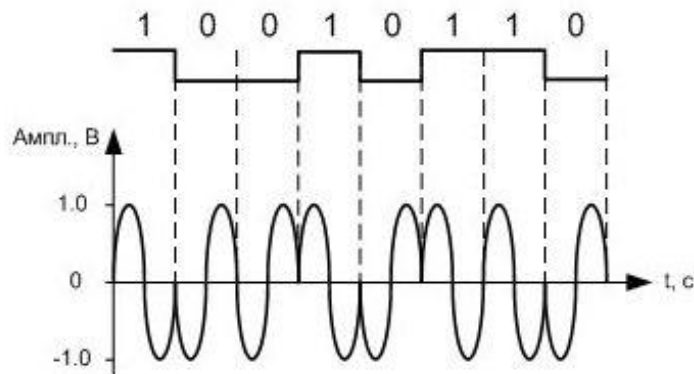


Рисунок 3.6. Сигнал фазовой манипуляции

Работа схемы модуляции заключается в смещении фазы модулируемого сигнала  $s_i(t)$  на одно из двух значений, нуль или  $\pi$  ( $180^\circ$ ). На рис. 3.6 явно видны характерные резкие изменения фазы при переходе между символами. Если модулируемый поток данных состоит из чередующихся нулей и единиц, такие резкие изменения будут происходить при каждом переходе.

Выходное напряжение PSK-модуляторов имеет спектр колебания, который содержит множество боковых частот, а сам процесс ФМн сопровождается и частотной манипуляцией.

Перейдем к результатам моделирования системы фазовой манипуляции с использованием PSK-модуляторов типа Real и Complex, схема включения которых представлена на рис. 3.7. Сравнивая полученные результаты можно сделать вывод, что если при ЧМн частота модулированного сигнала линейно зависит от величины модулирующего сигнала при интегральной зависимости от фазы, то при ФМн линейное изменение фазы модулированного сигнала сопровождается изменением частоты  $\omega(t) = d\phi/dt$ .

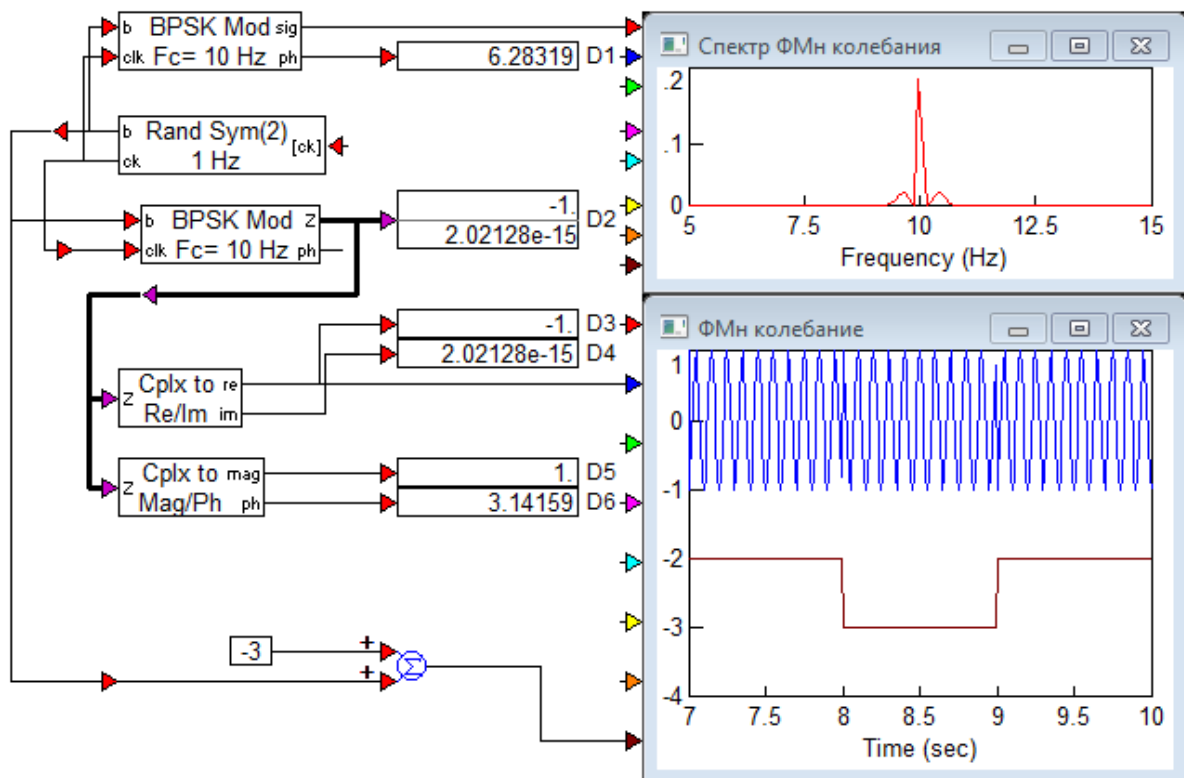


Рисунок 3.7. Схема PSK-модулятора

### 3.4. Многопозиционная модуляция

Такие модуляторы используются в тех случаях, когда канал передачи цифровой информации обладает ограниченной полосой пропускания [9]. В данном случае используют многопозиционные сигналы, в частности амплитудно-манипулированные (АМн) или квадратурные АМн (КАМн) колебания. Первые в программе VisSim называются PAM (Pulse Amplitude Modulation – с импульсной модуляцией (манипуляцией) амплитуды), вторые –

QAM (Quadrature Amplitude Modulation – с одновременной манипуляцией амплитуды и фазы несущего гармонического сигнала).

Идея использования многопозиционных сигналов для снижения требуемой полосы пропускания линии связи заключается в разбиении сообщения в виде двоичной последовательности на блоки (посылки), каждый из которых содержит комбинации из  $m$  двоичных символов (1 или 0), количество которых соответствует возможным состояниям выходного сигнала модулятора.

В случае 4-РАМ (рис. 3.8а) модулированный сигнал представляется четырьмя выходными уровнями несущего колебания (в скобках указаны соответствующие им уровни входных сигналов  $x$ ):  $-3$  ( $x \leq 0,5$  В),  $-1$  ( $0,5 < x \leq 1,5$  В),  $+3$  ( $1,5 < x \leq 2,5$  В) и  $+1$  ( $x > 2,5$  В) с интервалом  $\alpha$  равным например, 2 В. Эти уровни отображены точками 0 – 3 на вещественной оси I комплексной плоскости IQ (I, Q – ось мнимых и действительных значений). Следовательно, в случае 4-РАМ каждый блок будет содержать двухзначные комбинации 00 – 11, а это значит, что скорость двоичного потока на выходе модулятора (и в линии связи) будет в два раза ниже, чем на его входе.

В случае 8-РАМ (рис. 3.8б) модулирующий сигнал представляется уже восьмью уровнями:  $\pm 7$ ,  $\pm 5$ ,  $\pm 3$  и  $\pm 1$ , что позволяет увеличить длину каждого блока до трех символов (000 – 111) и, следовательно, уменьшить скорость передачи в 4 раза. Если, кроме модуляции амплитуды, использовать и модуляцию фазы (16-QAM на рис. 3.8в), то число символов в пачке увеличивается до четырех и, следовательно, скорость передачи можно уменьшить еще в 2 раза.

Использование многопозиционной QAM в чистом виде сталкивается с серьезными проблемами, связанными с недостаточной помехоустойчивостью при увеличении скорости. Поэтому во всех современных высокоскоростных протоколах этот вид модуляции применяется в сочетании с методами решетчатого кодирования, что позволяет снизить требования к отношению сигнал/шум в канале. При этом пространство сигналов расширяется путем добавления к информационным битам вспомогательных, которые образуются

благодаря сверточному кодированию части информационных бит и введению элементов запаздывания. Расширенная таким образом группа подвергается многопозиционной амплитудно-фазовой модуляции.

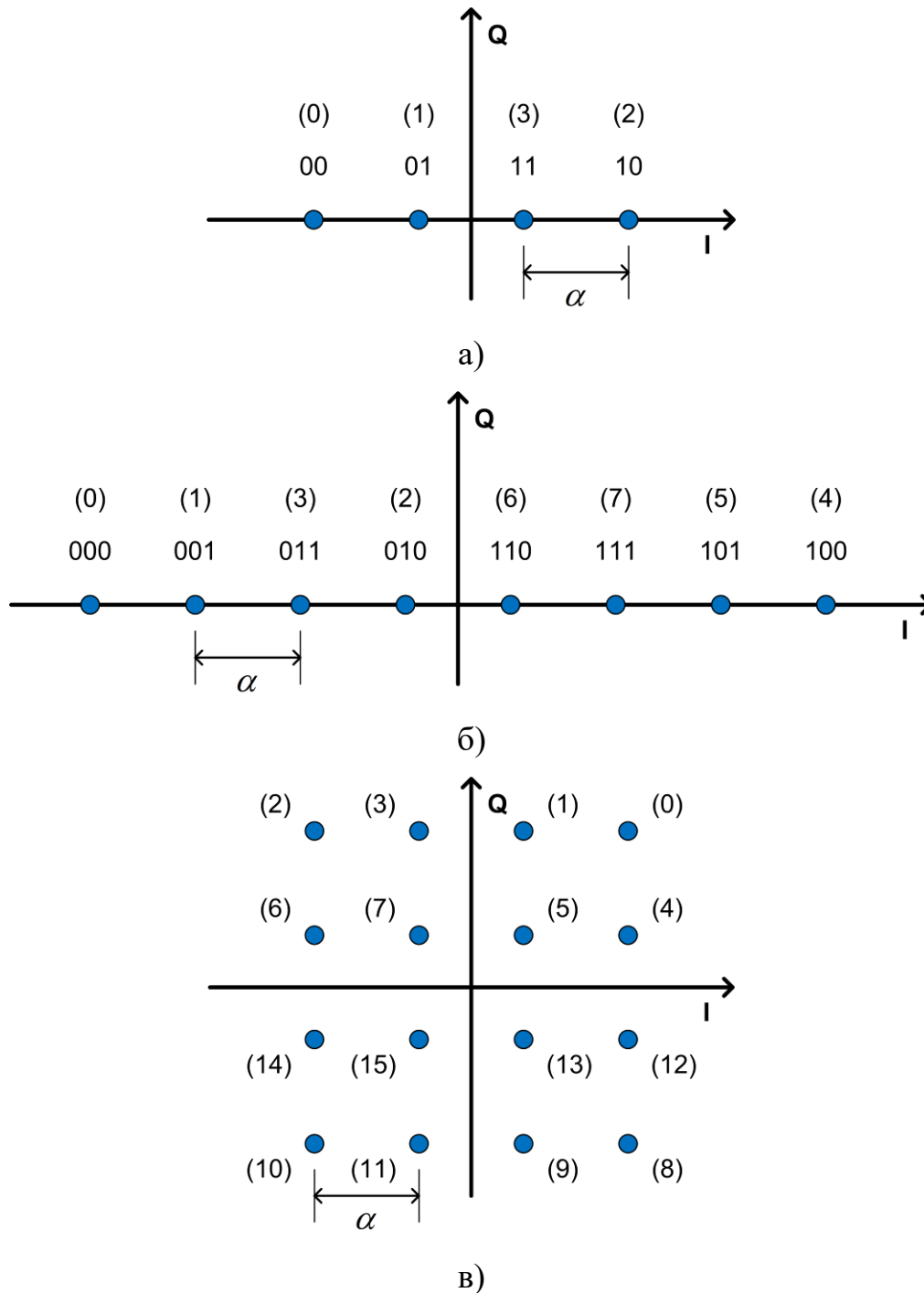


Рисунок 3.8. Принципы формирования PAM и QAM сигналов:  
а) 4-PAM; б) 8-PAM; в) 16-QAM

При демодуляции производится декодирование сигнала по алгоритму Виттерби, позволяющему, в частности, за счет введенной избыточности

выбрать из сигнального пространства наиболее приемлемый вид принятого сигнала и тем самым определить значения информационных бит.

Рассмотрим результаты моделирования 4-РАМ модулятора (рис. 3.9), схема включения которого, кроме рассмотренных ранее, содержит: собственно блок 4РАМ, в диалоговом окне которого можно выбрать тип модулятора (РАМ или QAM); источник прямоугольных импульсов Rect. Pulses с параметрами по умолчанию, за исключением Low Level = 0 (однополярный режим работы) и High Level =  $X_i$  – уровень выходного напряжения, варьируемый в соответствии с логикой работы модулятора.

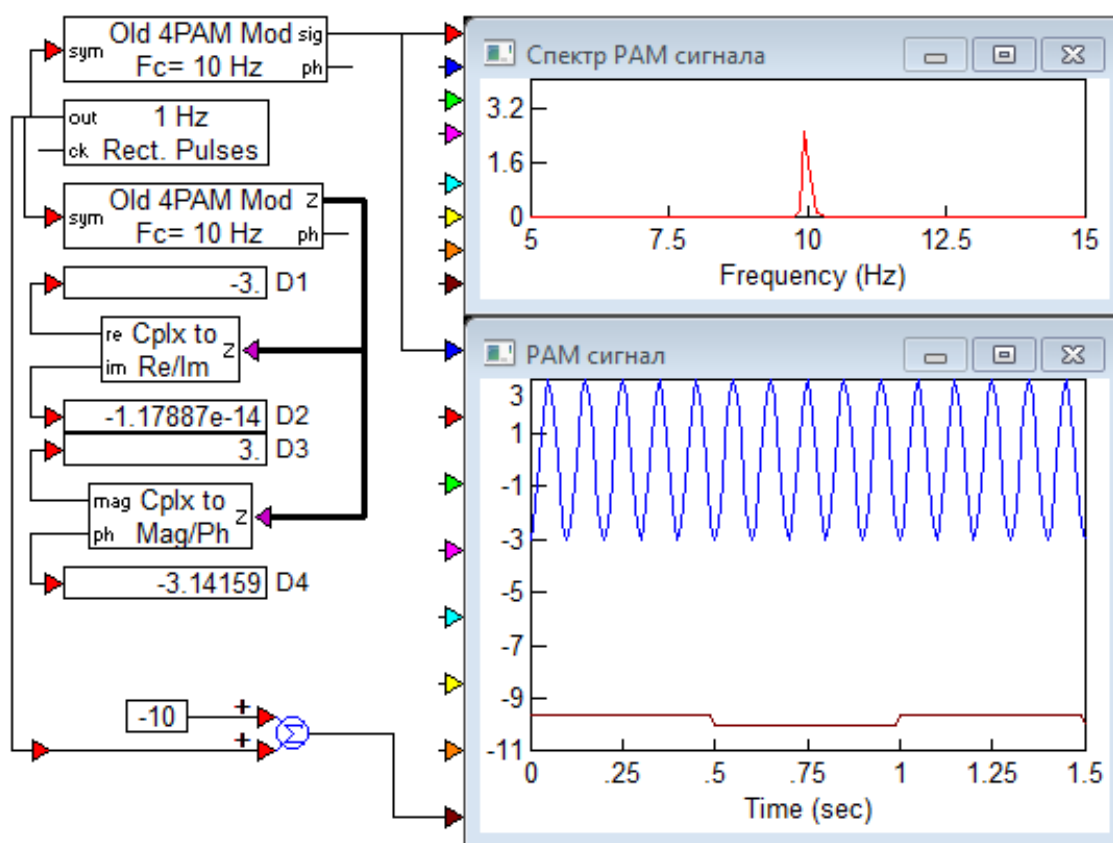


Рисунок 3.9. Схема 4-РАМ модулятора

Выходное напряжение модулятора может быть записано в виде:

$$U_i(j\omega) = U_i(X_i) e^{j(\omega t + \theta + \varphi)} = A_i e^{j(\theta + \varphi)} \text{ при } 0 \leq t \leq \tau, \quad (3.8)$$

где  $A_i = U_i(X_i)e^{j\omega t}$  – комплексная амплитуда;  $U_i(X_i)$  – амплитуда модулированного сигнала  $U_i$  при амплитуде модулирующего сигнала  $X_i$ ;

$\omega = 2\pi f$ ,  $f$  – угловая (рад/с) и циклическая (Гц) частота несущего колебания;  
 $\varphi = 0$  – фазовый угол несущей, рад;  $\theta$  – фазовый угол модулированного сигнала (измеряется блоком D4), рад;  $\tau = 0,5$  с – длительность модулирующего импульса амплитудой  $X_i$ . Результаты моделирования для всех четырех комбинаций  $X_i$  приведены в табл. 3.1.

Таблица 3.1. Результаты моделирования работы 4-РАМ модулятора

| № п/п | Параметры      | Результаты моделирования |         |         |         |
|-------|----------------|--------------------------|---------|---------|---------|
| 1.    | $U_i(X_i)$ , В | $-3(0,4)$                | $-1(1)$ | $+3(2)$ | $+1(3)$ |
| 2.    | $D1(re)$ , В   | $-3$                     | $-1$    | $3$     | $1$     |
| 3.    | $D2(im)$ , В   | $0$                      | $0$     | $0$     | $0$     |
| 4.    | $D3(mag)$ , В  | $3$                      | $1$     | $3$     | $1$     |
| 5.    | $D4(ph)$ , рад | $-\pi$                   | $-\pi$  | $0$     | $0$     |
| 6.    | Код            | 00                       | 01      | 10      | 11      |

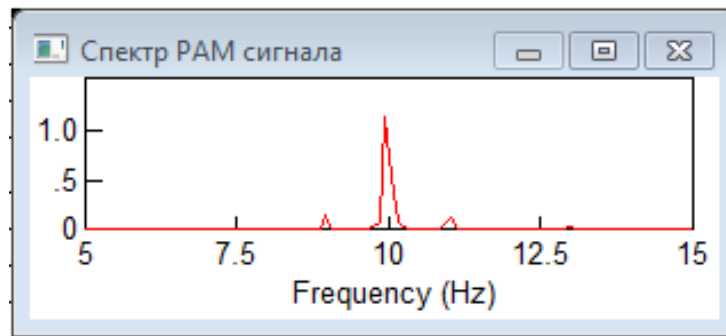
Для первого столбца ( $U_i(X_i) = -3(0,4)$ )  $\theta = -\pi$ , в этом случае из (3.8) при  $\varphi = 0$  получаем

$$U_1(j\omega) = A_1 e^{j\theta} = A_1 (\cos\theta + j\sin\theta) = A_1 [\cos(-\pi) + j\sin(-\pi)] = 3[-1 + j \times 0],$$

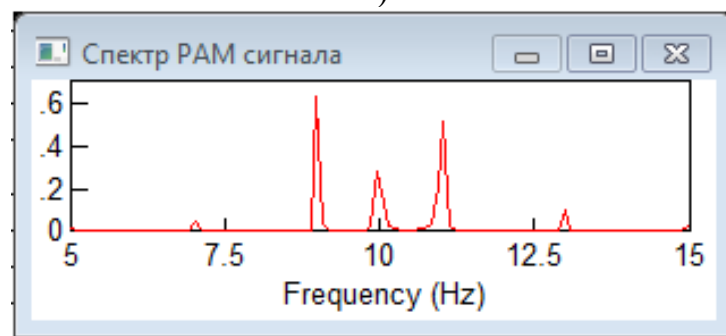
откуда следует: действительная часть комплекса  $\text{Re} = U_i(X_i) = -3$  В (показания регистратора D1), мнимая часть  $\text{Im} = 0$  (D2), модуль комплекса  $\text{mag} = 3$  В (D3),  $\text{ph} = \theta = \arctg(-\sin\theta/\cos\theta) = -\pi$  (D4). Таким образом, полученные данные соответствуют друг другу. Заметим, что для рассматриваемого случая модуляция амплитуды несущей отсутствует, – модулируется только фаза несущей поворотом ее на  $180^\circ$ , что является ее отличительным признаком от комбинации  $U_i(X_i) = +3(2)$  В (третий столбец). Одновременная модуляция амплитуды и фазы имеет место только для комбинации  $U_i(X_i) = -1(1)$  В (второй столбец). При комбинации  $U_i(X_i) = +1(3)$  В (четвертый столбец) модулируется только амплитуда несущей, чем она и отличается от предыдущей комбинации.

Что касается спектра, то его ширина для каждой комбинации различна. Наименьшей шириной обладает комбинация  $U_i(X_i) = -3(0,4)$  (рис. 3.9), от нее незначительно отличается комбинация  $U_i(X_i) = -1(1)$  (рис. 3.10а). Несколько

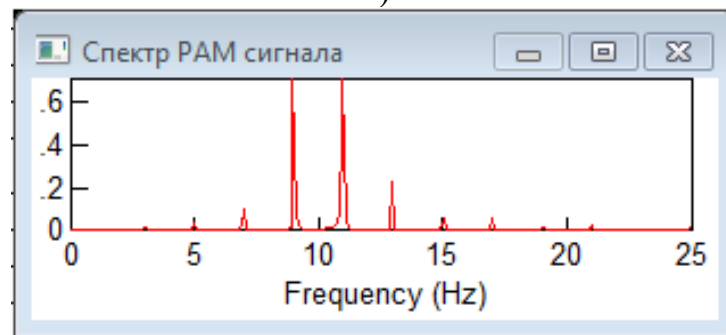
большей шириной спектра отличаются комбинации  $U_i(X_i)=+1(3)$  (рис. 3.10б) и  $U_i(X_i)=+3(2)$  (рис. 3.10в).



а)



б)



в)

Рисунок 3.10. Результаты измерений спектра 4-РАМ сигнала:  
а) комбинация кода 01; б) комбинация кода 11; в) комбинация кода 10

В общем же случае максимальная ширина спектра ориентировочно может быть определена по формуле:  $\Delta f = f + (1 + 2^{n-1})/\tau$ , где  $n$  – десятичный эквивалент формируемого кода. Например, для кода 00

$$\Delta f = f + (1 + 2^{-1})/\tau = 10 + 1,5/0,5 = 13 \text{ Гц},$$

для кода 11

$$\Delta f = f + (1 + 2^2)/\tau = 10 + 4/0,5 = 18 \text{ Гц}.$$

Однако наиболее высокоэнергетическая часть спектра расположена в полосе  $10 \pm 2$  Гц, что особенно заметно при линейном масштабе по оси амплитуд.

Рассмотрим особенности формирования сигнала 16-QAM модуляции представленного на рис. 3.11.

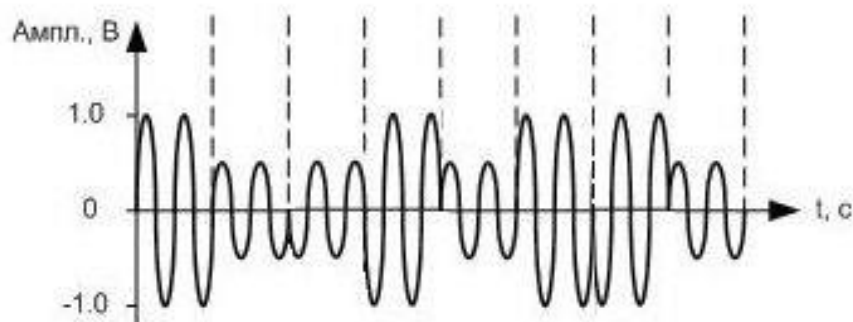


Рисунок 3.11. Сигнал амплитудно-фазовой манипуляции

Схема включения модулятора 16-QAM одинакова с приведенной на рис. 3.9, а тип модулятора выбирается в диалоговом окне блока 4РАМ. При этом настройка блока сводится к установке в строке параметра Constellation Rotation (циклический сдвиг между каналами по фазе) значения  $22,5^\circ$  ( $\sim 0,3927$  рад).

Поскольку показания регистраторов D1 – D4, как было показано выше, соответствуют друг другу, то ограничимся только рассмотрением результатов, полученных регистраторами D3 и D4, показания которых для варьируемых значений амплитуды  $X_i$ , импульсного генератора приведены в табл. 3.2 с округлением до второго знака в дробной части. Из этой таблицы видно, что исследуемый модулятор обеспечивает на выходе 16 неповторяющихся комбинаций амплитуда/фаза.

Что же касается спектра выходных сигналов модулятора 16-QAM, то для каждой комбинации они несколько отличаются друг от друга, но в большинстве случаев основная часть амплитуд сосредоточена в полосе  $10 \pm 2$  Гц.



Таблица 3.2. Результаты моделирования работы 16-QAM модулятора

| №<br>п/п | $X_i$ | D3(mag),<br>В | D4(ph),<br>рад | №<br>п/п | $X_i$ | D3(mag),<br>В | D4(ph),<br>рад |
|----------|-------|---------------|----------------|----------|-------|---------------|----------------|
| 1.       | 0,1   | 4,24          | 1,18           | 9.       | 8     | 4,24          | – 0,39         |
| 2.       | 1     | 3,16          | 1,64           | 10.      | 9     | 3,16          | – 0,86         |
| 3.       | 2     | 4,24          | 2,75           | 11.      | 10    | 4,24          | – 1,96         |
| 4.       | 3     | 3,16          | 2,28           | 12.      | 11    | 3,16          | – 1,51         |
| 5.       | 4     | 3,16          | 0,71           | 13.      | 12    | 3,16          | 0,07           |
| 6.       | 5     | 1,41          | 1,18           | 14.      | 13    | 1,41          | – 0,39         |
| 7.       | 6     | 3,16          | – 3,07         | 15.      | 14    | 3,16          | – 2,43         |
| 8.       | 7     | 1,41          | 2,75           | 16.      | 15    | 1,41          | – 1,96         |

### 3.5. Цифровая широтно-импульсная модуляция

Рассматриваемые модуляторы относятся к наиболее часто используемым на практике модуляторам с односторонней широтно-импульсной модуляцией (ОШИМ), при которой положение одного фронта импульса определяется тактовым импульсом, а положение другого фронта изменяется в зависимости от величины входного сигнала [9].

Различают ОШИМ первого и второго рода (ОШИМ-1 и ОШИМ-2). При ОШИМ-1 длительность импульсов определяется значением входного сигнала в момент появления модулированного фронта импульса. При ОШИМ-2 длительность импульсов пропорциональна значению входного сигнала в момент, соответствующий посылке тактового импульса.

Результаты испытания широтно-импульсного модулятора для случая импульсного модулирующего сигнала приведены на рис. 3.12.

В качестве источника такого сигнала использован генератор импульсов с параметрами: Low Level = 0 (режим однополярных положительных импульсов), амплитуда (High Level) – 0,5 В, частота следования (Pulse Frequency) – 2 Гц, длительность импульса (Duration) – 0,04 с, остальные – по умолчанию. Параметры модулятора: длительность прямоугольного импульса несущей (Pulse Width) – 0,04 с, частота следования (Symbol Rate) – 10 Гц, остальные – по умолчанию.

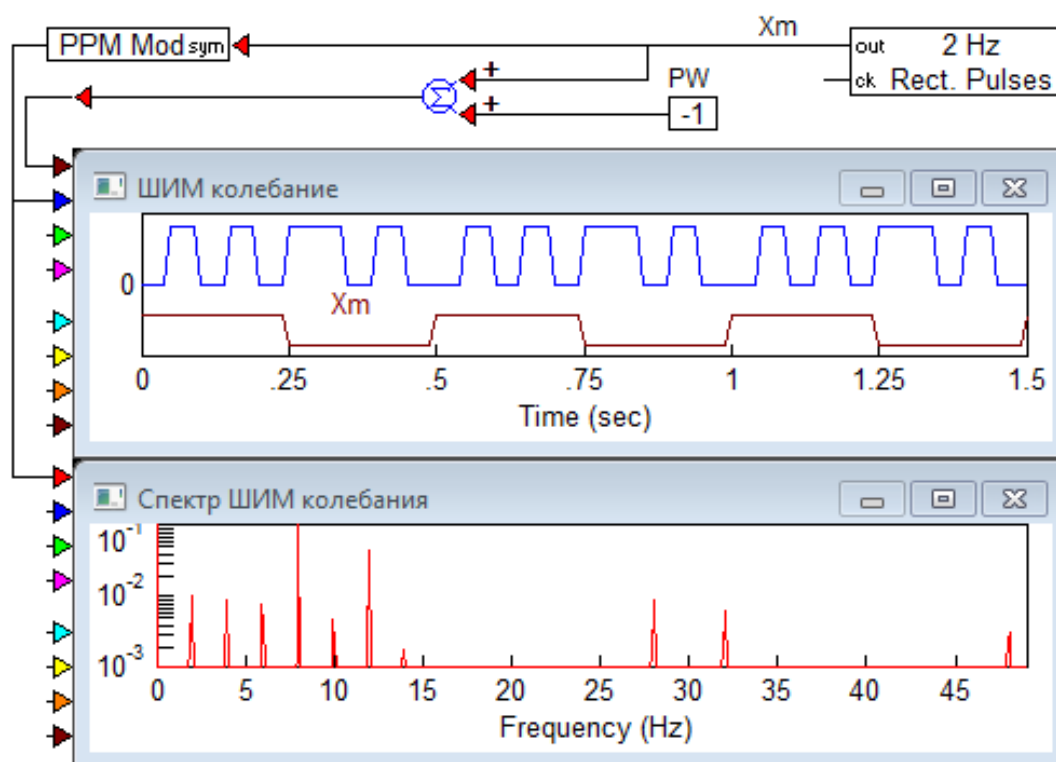


Рисунок 3.12. Схема широтно-импульсного модулятора с импульсным моделирующим сигналом

### 3.6. Лабораторная работа 3. Исследование особенностей работы модулирующих и демодулирующих устройств

#### 3.6.1. Цель и содержание

Целью лабораторной работы является приобретение студентами практических навыков исследования принципов функционирования модуляторов и демодуляторов в системах передачи дискретных сигналов.

#### Содержание

1. Исследование схемы амплитудного манипулятора.
2. Исследование особенностей работы схемы частотного манипулятора.
3. Исследование схемы фазового манипулятора.
4. Исследование особенностей работы схем многопозиционных манипуляторов.
5. Исследование особенностей работы схемы цифрового широтно-импульсного модулятора.

### **3.6.2. Аппаратура и материалы**

Для изучения материалов этой лабораторной работы необходимо:

1. Компьютер под управлением операционной системы Windows XP/Vista/7 с параметрами по умолчанию.
2. Установленная инструментальная среда визуального моделирования динамических систем VisSim/Comm 7.0.
3. Текстовый процессор (Microsoft Word 2003/2007/2010 или его аналог), предназначенный для создания и редактирования отчета по лабораторной работе.

### **3.6.3. Порядок выполнения работы**

1. Изучение теоретического материала.
2. Выполнение задания, предлагаемого в лабораторной работе.

#### **Задание**

1. Запустить на выполнение файл 3-1-1.VSM. Провести моделирование схемы амплитудного манипулятора, изменяя количество уровней модулированного сигнала.

2. Запустить на выполнение файл 3-1-2.VSM. Путем моделирования установить зависимость соотношения между интенсивностями несущей и боковых частот от количества уровней модулированного сигнала. Для удобства отсчета увеличьте (стандартным образом) размер графопостроителя «Спектр АМн колебания» по вертикали и горизонтали.

Установить зависимость формы спектра и временного смещения восстановленного АМн колебания относительно модулирующего от времени моделирования путем увеличения параметра Impulse Time источника Impulse.

Провести испытания амплитудных модуляторов в узкополосном режиме (при  $F_c = 0$ ). Сравнить спектры модулированных сигналов с показанными на рис. 3.3.

3. Запустить на выполнение файл 3-2-1.VSM. Провести моделирование схемы частотного манипулятора. Исследовать зависимость формы спектра ЧМн

сигнала от амплитуды модулирующего сигнала (0,1 и 0,3 В) и уровнем смещения между тонами. Провести сравнительный анализ полученных результатов.

4. Запустить на выполнение файл 3-3-1.VSM. Провести моделирование схемы фазового манипулятора. Исследовать зависимость формы спектра ФМн сигнала от частоты и амплитуды модулирующего сигнала (0,1 и 0,3 В). Провести сравнительный анализ полученных результатов.

5. Запустить на выполнение файл 3-4-1.VSM. Провести моделирование схемы модулятора 8-РАМ при следующих разрешенных для него комбинациях  $U_i(X_i) = -7(0,4), -5(1), -1(2), -3(3), 7(4), 5(5), 1(6), 3(7)$ . Провести сравнительный анализ полученных результатов.

6. Запустить на выполнение файл 3-4-2.VSM. Провести моделирование схемы модулятора 16-QAM при  $X_i > 15$  В, а также при его дробных значениях. Провести сравнительный анализ спектров выходных сигналов модулятора при различных значениях  $X_i$ .

7. Запустить на выполнение файлы 3-5-1.VSM. Провести исследование особенностей работы цифрового широтно-импульсного модулятора. Установить зависимость формы спектра ОШИМ сигнала от параметров модулирующего сигнала.

### 3.6.4. Вопросы для защиты работы

1. Прокомментируйте особенности реализации цифровой модуляции.
2. Какими параметрами определяются характеристики частотных модуляторов?
3. Прокомментируйте осциллограмму спектра приведенную на рис. 3.3.
4. Каким образом можно определить число боковых частот и ширину спектра ЧМн колебания?
5. Какими эффектами сопровождается фазовая манипуляция?
6. Имеются ли отличия в ширине спектра ФМн и ЧМн колебания?

7. Что собой представляют многопозиционные модулированные сигналы и какую роль они выполняют в цифровых системах передачи информации?
8. Какие параметры несущего сигнала изменяются при ШИМ?
9. Охарактеризуйте ОШИМ-1 и ОШИМ-2. Перечислите недостатки ОШИМ-2.

## ГЛАВА 4. КАНАЛЫ ПЕРЕДАЧИ ИНФОРМАЦИИ

В системах передачи информации различают канал и линию связи (ЛС).

**Канал** – это тракт движения сигнала в многократной системе передачи со множеством входных и выходных устройств. **Линия связи** представляет физическую среду, по которой передаются сигналы. Многоканальные системы могут иметь одно- или двухпроводную ЛС. Так, в телефонии двухпроводная линия может использоваться для передачи сообщений большого числа абонентов. Соответственно различают аппаратуру, канала и аппаратуру линии, общую для всех каналов.

В зависимости от формы передаваемых сигналов и эксплуатационных требований различают каналы [9]:

- цифровые – с цифровым (импульсным) сигналом на входе и выходе;
- аналоговые – с непрерывным сигналом на входе и выходе;
- коммутируемые – когда они предоставляются потребителю (например, абоненту телефонной станции) на время соединения; такие каналы содержат в своем составе коммутационное оборудование;
- выделенные – арендуются у телефонных компаний или прокладываются самой заинтересованной организацией; являются двухточечными и их качество заметно выше качества коммутируемых из-за отсутствия влияния коммутационной аппаратуры;
- двухпроводные – используются как для передачи, так и для приема; позволяют сэкономить на стоимости кабелей, но требуют усложнения каналообразующей аппаратуры и аппаратуры пользователя для решения задачи разделения принимаемого и передаваемого сигналов;
- четырехпроводные – комбинация из двух двухпроводных для приема и передачи; характеризуются практически полным отсутствием взаимного влияния сигналов, передаваемых во встречном направлении.

В системах передачи информации часто используют такие понятия, как:

DTE (Data Terminal Equipment) – оконечное оборудование данных (ООД), которое может быть как источником информации, так и ее получателем, например персональный компьютер, планшет, терминал, устройство сбора данных, кассовый аппарат, коммуникатор, приемник глобальной навигационной системы, модем и т. п.;

DCE (Data Communications Equipment) – аппаратура передачи данных (АПД), обеспечивающая возможности передачи информации между двумя или большим числом DTE по каналу определенного типа, например по телефонному.

#### **4.1. Типы каналов и линий связи**

В технике передачи информации находят применение следующие каналы связи: механические, акустические, оптические, электрические и радиоканалы, которые различаются по используемым ЛС и физической природе сигналов [9].

Механические каналы применяются для передачи на короткие расстояния (до 500 м). К ним относятся: чисто механические каналы (например, трос управления дроссельной заслонкой карбюратора двигателя); гидравлические (передающая среда – жидкость); пневматические (передающая среда – воздух).

Акустические каналы делятся на каналы звукового диапазона (до 20 кГц) и ультразвукового диапазона (свыше 20 кГц); находят применение в технике автоматического контроля, обнаружения и связи: акустический контроль состояния работающих механических объектов, ультразвуковая дефектоскопия, обнаружение объектов (подводных лодок, самолетов), гидролокация, акустическая связь и др.

##### **4.1.1. Электрические каналы**

Наиболее известными представителями таких каналов являются каналы с применением проводных ЛС целевого или многофункционального назначения [9]. Шкала частот, занимаемая сигналами в электрическом канале связи, условно делится на четыре диапазона: 0 – 200 Гц – подтональные

частоты; 300 – 3400 Гц – тональные частоты; 4000 – 8500 Гц – надтональные частоты; свыше 10 кГц – высокие частоты. При одновременном использовании линии для целей связи и передачи телемеханической информации заданные диапазоны частот распределяются следующим образом: тональные частоты используются для телефонии и тональной телеграфии, высокие частоты – для высокочастотной телефонии, подтональные и надтональные частоты – для телемеханической информации.

Электрическая линия связи (ЛС) представляет собой электрическую цепь с распределенными параметрами и характеризуется первичными и вторичными параметрами.

К **первичным** относятся: погонные (на 1 метр или километр длины) активное сопротивление  $R$  проводов и потерь из-за поверхностного эффекта (токи при высоких частотах «вытесняются» к поверхности проводника), Ом/км; проводимость  $G$  (утечка между проводами, измеряется в сименсах (1/Ом)), См/км; индуктивность  $L$ , Гн/км; емкость  $C$  между проводами, Ф/км.

К **вторичным** параметрам относятся: волновое сопротивление  $Z = [(R + j\omega L)/(G + j\omega)]^{1/2}$  и постоянная распространения  $\gamma = \beta + j\alpha = [(R + j\omega L)(G + j\omega)]^{1/2}$ , где  $\omega$  – частота сигнала;  $\beta$  – коэффициент затухания (на практике используется в виде  $\ln \beta$ , измеряемого в неперах (Нп));  $j\alpha$  – коэффициент сдвига фаз.

Для того чтобы вся проходящая по линии энергия поступала в нагрузку, нужно чтобы сопротивление нагрузки было равно волновому сопротивлению, иначе возникает эффект отражения, при котором часть энергии, отражаясь от конца линии, возвращается к источнику сигнала.

Проводные ЛС делятся на кабельные и воздушные. Кабельные ЛС обладают большей надежностью, меньшей зависимостью параметров от условий внешней (среды, малой проводимостью изоляции (около  $10^{-7}$  См/км), однако уступают воздушным по значениям активного сопротивления  $R$  и емкости  $C$ .



При построении современных телефонных сетей применяются абонентские кабельные линии на базе витых пар, которые обеспечивают скорость передачи 50 Мбит/с на расстояние около 100 м и 4 Мбит/с на расстояние около 1500 м. В отечественной практике абонентские линии в подавляющем большинстве представлены симметричными нескрученными парами, входящими в состав абонентских многопарных кабелей с жилами малого диаметра, или отдельными парами проводов.

Для высокоскоростной передачи данных (до 100 Мбит/с) на относительно короткие расстояния (десятки метров), например в сетях Ethernet, обычно используются кабели типа UTP (Unshielded Twisted Pair – неэкранированная витая пара) пяти категорий: категория 1 – для передачи речи или низкоскоростных данных (до 56 Кбит/с); 2 – до 1 Мбит/с; 3 – до 16 Мбит/с; 4 – до 20 Мбит/с; 5 – до 125 Мбит/с (состоит из четырех витых пар проводов). Характеристики таких кабелей можно улучшить за счет внешней экранирующей оболочки и использования соответствующего линейного кодирования.

Коаксиальный кабель по сравнению с витой парой позволяет работать в более широкой полосе частот (около 400 МГц на расстоянии 2 км). При использовании эффективного способа модуляции (0,5 Гц на бит/с) по коаксиальному кабелю можно передавать сигналы со скоростью 800 Мбит/с на расстояние около 2 км, что достаточно, например, для передачи 500 телевизионных каналов, предварительно сжатых до 1,5 Мбит/с.

К проводным каналам можно отнести так называемые **индуктивные контуры** (inductive loops), которые состоят из двух индуктивно связанных катушек и используются для передачи информации на очень малые расстояния (до нескольких сантиметров). В качестве примера такой среды передачи данных можно привести карточки со встроенными микропроцессорами (smart-карты), широко использующиеся для оплаты товаров (услуг) или для доступа в закрытые помещения.

#### 4.1.2. Радиоканалы

Радиолинии используются в тех случаях, когда требуется организовать связь с подвижными объектами или необходимо избежать затрат на прокладку кабельных линий [9]. При передаче информационного сигнала его спектр путем модуляции несущего колебания переносится в область радиочастот, которые могут распространяться в радиолинии – свободном пространстве. При использовании отличающихся по частоте несущих возможна организация многоканальной связи.

По диапазону частот различаются радиоканалы: длинноволнового диапазона (длина волны  $> 1000$  м); средневолнового (200 – 1000 м); промежуточного (50 – 200 м); коротковолнового (10 – 50 м); ультракоротковолнового ( $< 10$  м) диапазонов.

На распространение радиоволн влияют отражающие и поглощающие свойства земной поверхности и атмосферы, особенно ионосферы, расположенной выше стратосферы. Ионосфера состоит из заряженных частиц газов – электронов и ионов, образующихся в результате воздействия солнечных лучей, космического излучения и метеоритных частиц. Концентрация ионов имеет два максимума. Первый, называемый слоем  $E$ , расположен на высоте 110 – 130 км; второй (под названием  $F$ ) – на высоте около 300 км. Изменение концентрации ионов с высотой вызывает непрерывное изменение угла преломления радиоволн, в результате чего волны распространяются криволинейно. Если направление распространения становится горизонтальным, не достигнув уровня максимальной ионизации, то происходит отражение радиоволны в сторону земли. Часть волн отражается слоем  $E$ , другая часть – слоем  $F$ , третья часть «пробивает» оба слоя и выходит за пределы земной атмосферы. Преломляющая способность ионосферы уменьшается с уменьшением длины волны: чем больше длина волны, тем меньшая степень ионизации требуется для ее отражения.

Волны, распространяющиеся в результате отражения от ионосферы, носят название **пространственные**, а распространяющиеся вдоль поверхности земли

благодаря дифракции – **поверхностные**. Чем меньше длина волны, тем быстрее затухает поверхностная волна (вследствие потерь в земной поверхности) и тем медленнее затухает пространственная. Сигналы длинноволнового диапазона полностью отражаются слоем  $E$ . Ультракоротковолновые сигналы не отражаются и выходят за пределы земной атмосферы, что позволяет использовать их для космической связи. При наземном использовании ультракоротких волн приемно-передающие станции располагаются в пределах прямой видимости. Если она ограничена только кривизной поверхности земли, то расстояние между станциями (в километрах) рассчитывается по формуле  $L = 7,2(h)^{1/2}$ , где  $h$  – высота антенной системы, м. В общем случае мощность радиосигнала, отнесенная к единице площади, убывает с расстоянием, в одних случаях – обратно пропорционально квадрату расстояния, а в других – его кубу.

Модель канала в среде VisSim, имитирующая затухание сигнала в свободном пространстве, определяется блоком Prop Loss (рис. 4.1).

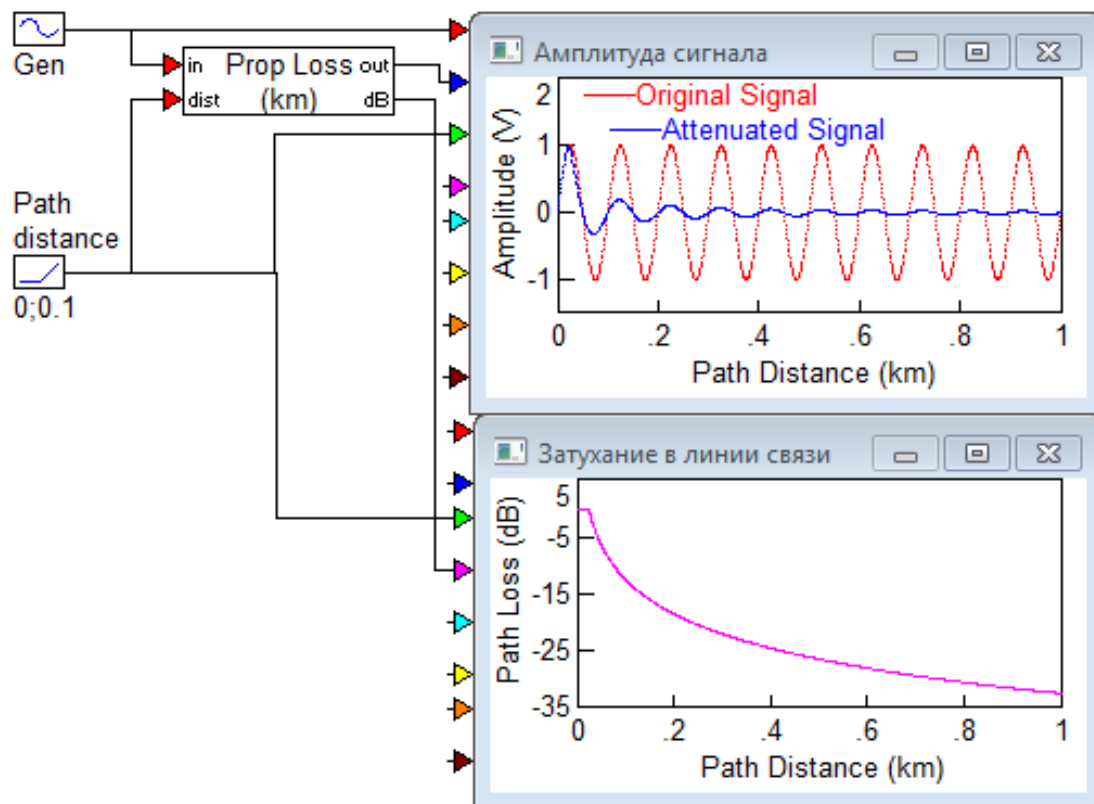


Рисунок 4.1. Модель канала, имитирующая затухание сигнала

Входной сигнал  $x(t)$  имитируется источником синусоидального сигнала Gen (осциллограмма 1). Выходной сигнал модели out определяется выражением  $y(t) = c x(t) / 4\pi f d$ , где  $c$  – скорость света.  $d$  – расстояние между передатчиком и приемником (в километрах или милях), имитируемое источником линейно изменяющегося сигнала Path distance;  $f$  – частота сигнала (в данном случае 1 МГц). На осциллограммах 2 и 3 показаны соответственно зависимости амплитуды сигнала от расстояния в абсолютных (на выходе out) и в относительных единицах дБ (на выходе dB).

Приведем примеры использования радиоканалов.

**Радиовещание и телевидение** – средства массового распространения информации. Ограниченный диапазон частот, в которых изначально работали эти системы (аналоговая передача), вынуждает компании осваивать спутниковую технику и цифровые методы передачи информации.

**Микроволновые радиоканалы** – используют частоты 4, 6 и 11 ГГц; работают по принципу «точка-точка» в зоне прямой видимости между передатчиком и приемником, что приводит к необходимости установки антенн на высоких мачтах с ретрансляторами через каждые 40 – 50 км; обеспечивают скорости передачи порядка 100 Мбит/с.

**WLL (Wireless Local Loop)** – беспроводные локальные каналы; являются экономически выгодной альтернативой проводных каналов в тех случаях, когда трудно использовать кабельные линии связи или необходимо обеспечить мобильность в пределах помещений, здании или территорий предприятий. Среди беспроводных локальных технологий наибольшее распространение получили стандарты IEEE 802.11 и Bluetooth.

IEEE 802.11 – набор стандартов связи, для коммуникации в беспроводной локальной сетевой зоне частотных диапазонов 2,4; 3,6 и 5 ГГц, более известен по названию Wi-Fi. Одна из последних версий стандарта Wi-Fi – IEEE 802.11n обеспечивает скорость передачи данных до 54 Мбит/с.

Радиосвязь Bluetooth осуществляется в диапазоне 2,4 ГГц. В Bluetooth применяется метод расширения спектра со скачкообразной перестройкой

частоты FHSS (Frequency Hopping Spread Spectrum), который обеспечивает устойчивость к широкополосным помехам. Стандарт Bluetooth 4.0 предоставляет скорость передачи данных в 1 Мбит/с.

**WiMAX** (Worldwide Interoperability for Microwave Access) – телекоммуникационная технология, разработанная с целью предоставления универсальной беспроводной связи на больших расстояниях для широкого спектра устройств (от рабочих станций и портативных компьютеров до мобильных телефонов). Она основана на стандарте IEEE 802.16, который также называют Wireless MAN. Используемые частоты и соответствующие скорости передачи данных 1,5 – 11 ГГц (до 75 Мбит/с), 2,3 – 13,6 ГГц (до 40 Мбит/с).

**Сотовые радиосистемы** получили широкое распространение при организации мобильной телефонной связи. Создание таких систем началось в 1970-х годах с разработки аналоговых устройств, в которых использовалась частотная модуляция для обеспечения передачи нескольких телефонных каналов. При этом обслуживаемый регион делится на соты, и достаточно удаленные друг от друга соты могут использовать одни и те же частоты. Таким образом, если 7 сот периодически чередуются для покрытия зоны обслуживания и в каждой используется свой набор из  $N$  частот, то во всей области используется  $7N$  частот. Плотность активных пользователей, которых система может обслуживать, составляет  $N/S$  пользователей на квадратный километр, где  $S$  – площадь соты.

Аналоговые системы постепенно были дополнены или заменены цифровыми. Преимущества цифровых систем состоят в возможности сжатия речи в целях сужения полосы частот, занимаемой телефонным каналом. Кроме того, цифровая передача обладает большей помехозащищенностью.

В основе работы цифровых сотовых систем лежит принцип ограничения полосы рабочих частот, которые распределяются в пределах территории, обслуживаемой данной системой. При этом применяется метод множественного доступа с частотным разделением каналов FDMA (Frequency

Division Multiplexing Access), совмещаемый с методом множественного доступа с временным разделением каналов TDMA (Time Division Multiple Access).

При использовании TDMA полоса рабочих частот вначале делится на несколько радиоканалов. Каждый радиоканал затем разбивается на ряд абонентских каналов при помощи временного разделения. Таким образом, радиоканал передает цифровой поток, который делится на циклы, а каждый цикл, в свою очередь, разбивается на  $K$  канальных интервалов. Канальные интервалы соответствуют различным телефонным каналам.

В другом методе, называемом методом множественного доступа с кодовым разделением каналов CDMA (Code Division Multiple Access), сигналы различных абонентов занимают всю полосу частот. Однако эти сигналы формируются путем умножения потоков битов пользователей на различные псевдослучайные импульсные последовательности. При получении абонентом цифрового потока его приемник умножает принятый сигнал на псевдослучайную последовательность, закрепленную за данным абонентом. Сигналы других абонентов, умноженные на принадлежащую данному абоненту псевдослучайную последовательность, воспринимаются приемником как быстро меняющийся шум, от которого можно избавиться усреднением (фильтрацией).

Развитием CDMA является стандарт LTE (Long Term Evolution), который обеспечивает скорость 173 Мбит/с на приём и 58 Мбит/с на передачу.

**Спутниковые широкополосные радиосистемы** реализуются с использованием спутников различных типов, определяемых местоположением их орбит. Спутники с геостационарной орбитой (35,784 км выше экватора) совершают оборот за то же время, что и Земля, и поэтому кажутся постоянно висящими в одной и той же точке небосвода. Такие спутники используются для фиксированных двухточечных и широковещательных коммуникаций с использованием параболических антенн с достаточно высокими требованиями по фокусировке и усилению, что не всегда приемлемо, особенно для бытовой аппаратуры и мобильных средств. Поэтому все чаще стали использоваться

коммуникационные спутники с низкими околоземными орбитами. Они обеспечивают намного меньшие задержки и меньшее ослабление сигналов, однако каждый такой спутник охватывает лишь небольшую область земной поверхности, что приводит к необходимости увеличения их количества.

#### 4.1.3. Оптические каналы

По диапазону используемых длин волн такие каналы делятся на каналы [9]: видимой части спектра (с длиной волны 0,3 – 0,75 мкм); инфракрасной (ИК) (0,75 – 1000 мкм); ультрафиолетовой (УФ) части спектра (< 0,3 мкм). Наибольшее распространение получили ИК-каналы вследствие меньшего ослабления И К излучения атмосферой по сравнению с видимой и УФ областью спектра, а также возможности его распространения ночью. В качестве ЛС в оптических каналах используется свободное пространство или оптическое волокно (ВОЛС – волоконно-оптические ЛС). Первый тип используется для организации локальных беспроводных сетей, например для связи между пультом управления и телевизором, между компьютером и клавиатурой, мышью и т. п. Разработкой стандартов инфракрасных ЛС занимается промышленная ассоциация IrDA (ассоциация инфракрасной передачи данных), образованная в июне 1993 года. Разработанные этой ассоциацией стандарты распространяются на скорости передачи 2400 бит/с, 115,2 Кбит/с, 1,152 Мбит/с и 4 Мбит/с. Передатчики IrDA используют ПК излучение с длиной волны 0,86 мкм и работают на расстояниях порядка нескольких метров. При низких скоростях передачи используется АМн сигналы, а на высоких – ФМн.

ВОЛС используются в высокоскоростных системах передачи информации (около 10 Гбит/с при расстояниях 100 км и более). Кроме того, по одному и тому же волокну могут одновременно передаваться оптические сигналы в различных областях спектра, что позволяет увеличить количество передаваемой информации.

Канал ВОЛС состоит из передатчика, оснащенного источником света, оптического волокна и оптического детектора в составе приемника. Передатчик преобразует цифровую информацию в оптический сигнал, используя, например, АМн, после чего он проходит по волокну и попадает на детектор (оптоэлектронный преобразователь) приемника, в котором производится обратное преобразование оптического сигнала в исходный цифровой поток.

Оптическое волокно представляет собой длинный цилиндр, изготовленный из стекла или пластика. В сетях связи используются три типа волокон: ступенчатое, градиентное (GRIN) и одномодовое (SMF), которые различаются показателем преломления и коэффициентом дисперсии  $D \leq 1/2R \times L$ , где  $R$  – скорость передачи, Мбит/с;  $L$  – длина ЛС, км.

ВОЛС со ступенчатым волокном обеспечивает  $R \times L \leq 100$  Мбит/с  $\times$  км при затухании 10 – 50 дБ/км, с градиентным –  $R \times L \leq 1$  Гбит/с  $\times$  км при затухании 7 – 15 дБ/км, с одномодовым –  $R \times L \leq 200$  Гбит/с  $\times$  км при затухании 0,2 – 1 дБ/км. Одномодовые волокна с компенсированной дисперсией обеспечивают  $R \times L \geq 1000$  Гбит/с  $\times$  км. Различие значений коэффициента дисперсии  $D$ , вытекающее из приведенных данных, объясняется различием пути распространения оптического сигнала в волокне, что, в свою очередь, объясняется различиями в конструкции волокон.

В качестве источников световых сигналов в системах с ВОЛС используются светодиоды LED (Light Emitting Diode) и лазерные диоды LD. Основными характеристиками этих источников являются интенсивность (мощность) излучения, длина волны светового сигнала и максимальная частота модуляции. Лазерные диоды допускают модуляцию на более высокой частоте, чем светодиоды, и имеют меньшую ширину спектра, что обеспечивает меньшую дисперсию и позволяет нескольким диодам одновременно работать по оптическому волокну в окне 100 нм на волне 1,55 мкм.

Работа светодиодов основана на рекомбинации электронов и дырок в полупроводнике, при которой высвобождаемая электроном энергия излучается в виде фотонов. Типовой светодиод обеспечивает световую мощность в



несколько милливатт при токе инжекции около 50 мА. При изменении тока инжекции оптическое излучение светодиода преобразуется в импульсы с соответствующей скоростью следования. Поскольку для рекомбинации пар электрон – дырка требуется определенное время, то значения граничной частоты, при которых излучаемая мощность составляет 70 % от максимального значения, находятся в пределах от 1 кГц до 100 МГц.

Лазерный диод отличается от светодиода тем, что р-п-переход помещается между двумя параллельными и частично отражающими поверхностями А и В, расстояние между которыми кратно половине длины волны светового излучения, генерируемого в процессе рекомбинации. Такая конструкция позволяет за счет интерференции прямых и отраженных волн получить световой поток высокой монохромности. Граничная частота лазерных диодов на несколько порядков выше, чем у светодиодов. Некоторые лазерные диоды допускают модуляцию с частотой до 11 Гбит/с. Однако мощность оптического сигнала сильно зависит от температуры, поскольку нагревание увеличивает генерацию свободных электронов и дырок, что требует применения сложных схем температурной стабилизации и подстройки тока инжекции.

В качестве детектора светового сигнала (оптоэлектронного преобразователя) используются полупроводниковые фотодиоды, PIN-диоды и лавинные фотодиоды. Основным параметром таких приборов является квантовая эффективность, показывающая, какая часть падающих на детектор фотонов преобразуется в поток электронов.

Быстродействие фотодиодов ограничивается частотой около 1 МГц. Более высоким быстродействием (до 20 ГГц) обладают PIN-диоды. Такие диоды составлены из трех слоев: *p*-легированного, беспримесного (не легированного) и *n*-легированного, что и определило название PIN. Поток фотонов, поступающих в высокоомный беспримесный слой обратносмещенного диода, порождает пары электрон – дырка, которые за счет сильного электрического поля в этом слое с большой скоростью устремляются соответственно к аноду и катоду диода. В беспримесном *i*-слое лавинного фотодиода напряженность

электрического поля достигает таких значений, что порождаемые электроны вызывают лавинную вторичную эмиссию электронов в  $i$ -слое, что в конечном счете приводит к существенному повышению чувствительности (отдельный фотон может «родить» до 100 пар электрон – дырка).

Как и в других средах, в оптоволокне можно использовать мультиплексирование с частотным разделением каналов TDM, только в этом случае его называют WDM (Wavelength Division Multiplexing) – мультиплексированием с разделением длин волн.

## 4.2. Каналы со случайными параметрами

Каналы со случайными параметрами можно разделить на каналы с прямой и отраженно-рассеянной волной [9]. В каналах первой группы имеет место прямая геометрическая видимость между передатчиком и приемником (например, наземные каналы оптической связи, каналы телевизионного вещания, гидроакустические каналы и т. п.).

В каналах второй группы прямая видимость между передатчиком и приемником отсутствует, и связь обеспечивается за счет использования свойств неидеальных сред рассеивать и переизлучать энергию сигнала. Каналы этой группы являются типичными каналами со случайными параметрами. Причиной случайного изменения параметров канала, может быть, действие целого ряда изменяющихся факторов (температуры, давления, плотности среды, наличия различных примесей и т. п.). Под влиянием этих факторов в среде распространения происходят сложные физико-химические процессы, в результате которых ее структура и характеристики претерпевают непрерывные случайные изменения.

Для удобства исследований в теории распространения сигналов принято различать макро- и микроструктуру среды. **Макроструктура** отражает основные (глобальные) особенности строения среды и ее физические свойства, обусловленные постоянно действующими факторами. Макроструктура среды и ее свойства меняются под действием метеорологических условий, а также

суточных, сезонных и годовых изменений температуры. Изменения макроструктуры во времени протекают довольно медленно, и их средний период может составлять десятки минут.

**Микроструктура** отражает местные (локальные) особенности строения среды, обусловленные действием таких случайных факторов, как сложные вихревые (турбулентные) процессы и различные физико-химические превращения, происходящие на отдельных участках среды.

Изменения структуры среды и ее свойств оказывают существенное влияние на процесс распространения сигнала и приводят к случайным изменениям его уровня на входе приемника вплоть до невозможности его обнаружения или так называемого замирания. При этом различают две составляющие такого явления: одна обусловлена весьма медленными изменениями макроструктуры среды, а другая — более быстрыми.

В общем случае модель распространения сигналов в каналах связи различной физической природы можно представить в виде трех последовательно соединенных пассивных линейных четырехполюсников.

Первый четырехполюсник определяет ослабление сигнала при распространении в среде (например, в свободном пространстве, в оптической линии связи, в проводной среде и т. п.). Его можно рассматривать как последовательное соединение ослабителя (аттенюатора) и линейного фильтра с равномерной частотной и линейной фазовой характеристиками на всех частотах. Ослабление сигнала в идеальной среде зависит только от геометрии распространения (дальности связи и характеристик приемной и передающей антенн). Величина этого ослабления может быть легко рассчитана по формуле дальности связи. При этом форма сигнала на выходе первого четырехполюсника остается такой же, как и на его входе, меняется только уровень этого сигнала.

Второй четырехполюсник отражает специфические особенности и физические свойства конкретной среды распространения. Он учитывает связанные с поглощением энергии сигнала в среде и их зависимость от частоты,

а также медленные изменения структуры и свойств среды во времени и пространстве. В результате этих изменений уровень сигнала на выходе второго четырехполосника испытывает медленные замирания, которые наблюдаются на достаточно длительных интервалах времени (десятки минут и более).

Третий четырехполосник модели учитывает сравнительно быстрые случайные изменения микроструктуры среды, включая тепловые шумы. При исследовании помехоустойчивости каналов связи чаще всего ограничиваются учетом **стационарных** случайных процессов, вероятностные характеристики которых не зависят от сдвига на произвольную величину всех временных аргументов. Это означает, что функция распределения стационарного процесса зависит только от одной временной переменной  $\tau = t_1 - t_2$ , и, следовательно, его корреляционная функция  $B(\tau) = B(t_1 - t_2) = B(-\tau)$ , поскольку для любого случайного процесса она обладает свойством симметрии.

В реальных каналах связи сигналы и многие помехи представляют собой, как правило, нестационарные случайные процессы хотя бы потому, что они ограничены во времени. Тем не менее на ограниченном временном интервале их чаще всего можно считать стационарными. Кроме того, в целях упрощения анализа принимается, что такие процессы являются **эргодическими**. Это означает, что любая их вероятностная характеристика, полученная усреднением по множеству возможных реализаций, с вероятностью, сколь угодно близкой к единице, равна полученной усреднением за достаточно большой промежуток времени из одной единственной реализации случайного процесса, что позволяет получить приближенное значение математического ожидания, дисперсии и др.

В телекоммуникационных системах влияние разнообразных дестабилизирующих факторов принято характеризовать **коэффициентом ошибок**. Например, коэффициент ошибок для типовой волоконно-оптической линии составляет около  $10^{-12}$ . Это значит, что в такой линии в среднем неправильно передается 1 из  $10^{12}$  бит. Если скорость передачи равна 155 Мбит/с, то один ошибочно переданный бит появится в среднем в течение

$10^{12}/155 \times 10^6$  с или около 2 ч. Проводные линии связи характеризуются коэффициентом ошибки на уровне  $10^{-7}$ .

Для цифровых каналов коэффициент ошибок принято обозначать как BER (Bit Error Rate – коэффициент ошибок на бит) или PER (Packet Error Rate – коэффициент ошибок на пакет). PER для данной линии определяет вероятность того, что хотя бы один из  $N$  битов данного пакета не будет принят правильно. Соотношение между коэффициентами ошибок на пакет и бит определяется формулой:

$$PER = 1 - (1 - BER)^N. \quad (4.1)$$

Если  $N \times BER \gg 1$ , то  $PER = N \times BER$ . Например, если  $N = 10^5$  и  $BER = 10^{-7}$ , то  $PER = 10^{-2}$ .

Если коэффициент ошибок для линий на базе коаксиальных или волоконно-оптических кабелей находится в пределах  $10^{-7} - 10^{-12}$ , то для радиолиний он составляет около  $10^{-3}$ .

#### 4.2.1. Бинарный симметричный канал

Бинарный симметричный канал относится к классу дискретных. Математически он описывается алфавитом кодовых символов на входе, алфавитом кодовых символов на выходе, их вероятностными характеристиками и скоростью передачи [9]. Если вероятности появления сигналов на входе и выходе канала не меняются во времени и не зависят от того, какие символы передавались и принимались ранее, то такой дискретный канал называют **однородным** (стационарным) и без памяти. Если эти вероятности зависят от времени, то канал называется **неоднородным** (нестационарным), если же они зависят от того, какие символы передавались и были приняты ранее, то канал называют **каналом с памятью** (такой канал описывается цепью Маркова и используется в системах связи с обратной связью).

Если в однородном дискретном канале алфавиты на входе и выходе одинаковы (характерно для большинства дискретных систем связи), то такой

канал называют симметричным каналом **без стирания**, в противном случае он называется каналом **со стиранием**. В каналах со стиранием дополнительный символ в выходной кодовой комбинации используется в системах с обратной связью для повышения помехоустойчивости канала. Модель наиболее простого двоичного симметричного канала без памяти и стирания представлена в системе моделирования VisSim блоком BSC (Binary Symmetric Channel). Если вероятность ошибочного приема одиночного символа в таком канале равна  $p$ , то вероятность того, что из  $n$  возможных  $q$  символов на приеме зарегистрировано ошибочно, определяется формулой Бернулли:

$$p(q) = C_n^q p^q (1-p)^{n-q}, \quad (4.2)$$

где  $C_n^q$  – число сочетаний из  $n$  элементов по  $q$ . Из (4.2) следует, что при  $p < 0,5$  с ростом  $q$  вероятность  $p(q)$  уменьшается, т. е. в рассматриваемом канале наиболее вероятными являются реализации, в которых все  $n$  символов принимаются безошибочно независимо от длины цепочки символов  $n$ , если  $n \times p \ll 1$ .

В дискретных каналах, кроме ошибочного приема символов и стираний, возможна еще одна разновидность искажений, которая связана с нарушением синхронизации переданных и принятых символов: если в приемнике за начало цепочки будет принят момент времени, опережающий действительный момент прихода символов, то будет зарегистрирован ряд лишних символов; если же за начало будет принят запаздывающий относительно действительного момент времени, то ряд переданных символов выпадает.

Количество символов  $S$ , передаваемых в среднем по дискретному каналу, чаще всего устанавливается с учетом ширины полосы пропускания  $F$  непрерывного канала, который является составной частью дискретного. Согласно теореме Котельникова непрерывный сигнал длительности  $T$  с полосой частот  $F$  описывается  $2F \times T$  отсчетами, например за 1 с при  $F = 1$  Гц – двумя отсчетами. Отсюда следует, что если передается дискретный двоичный сигнал (каждый его отсчет принимает 2 значения), то за 1 с в полосе 1 Гц

нельзя передать больше чем 2 импульса, т. е. количество символов в секунду  $C < 2F$  (так называемый предел Найквиста). Преодоление этого ограничения в принципе возможно путем учета переходных процессов, однако это связано с большими аппаратными затратами.

Схема для испытаний бинарного симметричного канала (рис. 4.2) содержит источник Rect. Pulses однополярных прямоугольных импульсов (при нулевом уровне Low Level) длительностью (Duration) импульса – 0,2 с и частотой следования 1 Гц, блок смещения осциллограммы входных сигналов в составе источника Gen и сумматора, канал BSC с вероятностью ошибки передачи одиночного бита (Channel Error Probability) – 0,1, измеритель количества ошибочных битов на символ SER с регистраторами количества ошибочных битов D1 и количества 8-битовых символов D2. Модель канала BSC характерна тем что сигналы с амплитудой  $> 0,5$  передаются на выход в виде логической единицы, в противном случае в виде логического нуля. Параметры блока SER: режим работы (Output Mode) – анализ потока символов (опция Symbol Error Rate); количество бит в символе (Bits per Symbol) – 8; остальные – по умолчанию.

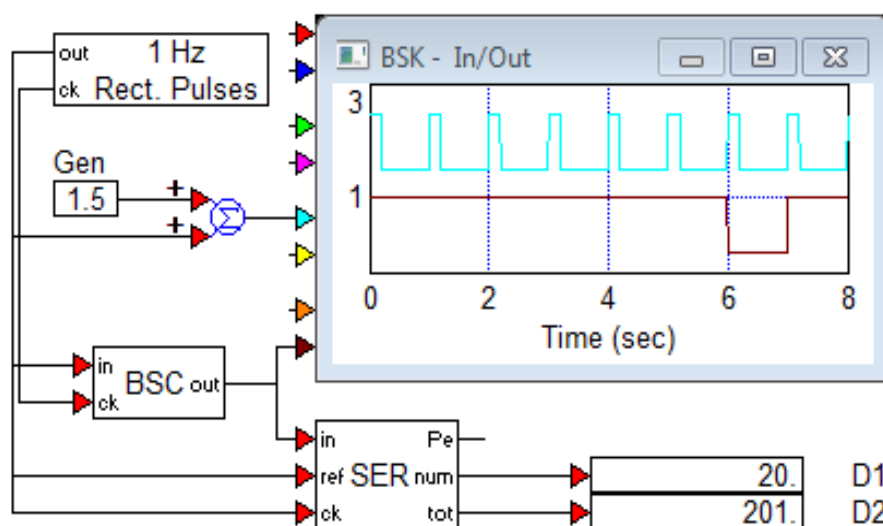


Рисунок 4.2. Схема для испытания бинарного симметричного канала

Как следует из осциллограммы на рис. 4.2, при вероятности безошибочной передачи  $p = 1 - 0,1 = 0,9$  количество поступивших за 8 с на вход BSC канала

импульсов (битов)  $n = 8$ , количество ошибочных битов на выходе канала  $q = 1$ , что близко к показаниям регистраторов D1, D2 (в среднем 0,0995 бита на 1 символ). Для формулы (4.2) рассчитываем число сочетаний  $C_n^q = n!/(n-q)! \times q! = 8!/7! \times 1! = 8$ . Рассчитывая другие составляющие формулы (4.2), получаем вероятность безошибочной передачи 8-битового символа  $p(q) = 0,383$ . Результаты расчета для рассмотренного случая, а также для вероятности ошибок 0,5 и 0,7 ( $p = 0,5$  и 0,3) приведены в табл. 4.1. Осциллограммы при данных вероятностях представлены на рис. 4.3 – 4.4.

Таблица 4.1. Результаты расчетов бинарного симметричного канала

| Осциллограмма | $n$ | $q$ | $q_{d1}$ | $p$ | $n!/(n-q)!q!$ | $p(q)$ |
|---------------|-----|-----|----------|-----|---------------|--------|
| рис. 4.2      | 8   | 1   | 0,1      | 0,9 | 8             | 0,383  |
| рис. 4.3      | 8   | 3   | 0,3      | 0,5 | 56            | 0,219  |
| рис. 4.4      | 8   | 5   | 0,5      | 0,3 | 56            | 0,254  |

Как видно из осциллограммы на рис. 4.3, при  $p = 0,5$  количество ошибочно принятых битов  $q = 3$  (1-й, 6-й и 8-й) (по показаниям D1 их 3), при  $p = 0,3$  (рис. 4.4) –  $q = 5$  (1-й, 4-й, 5-й, 6-й и 8-й) (по показаниям D1 их 5). Таким образом, имеет место соответствие между результатами осциллографических измерений и измерений с помощью блока SER при различных параметрах бинарного симметричного канала.

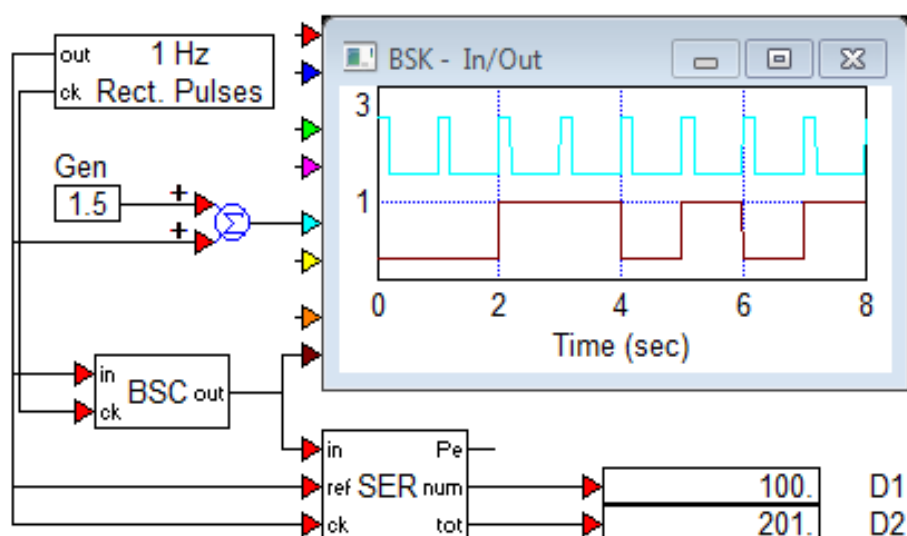


Рисунок 4.3. Осциллограммы сигналов при ошибке передачи 0,5



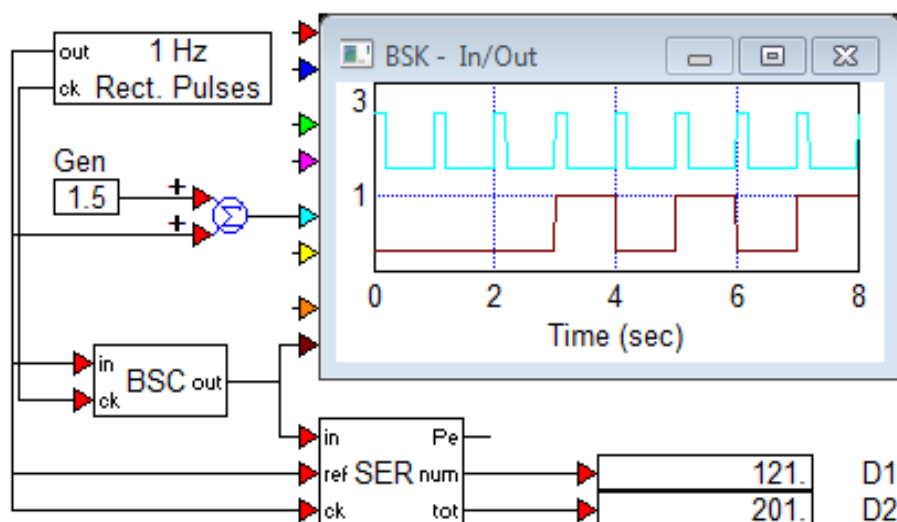


Рисунок 4.4. Осциллограммы сигналов при ошибке передачи 0,7

#### 4.2.2. Многолучевые каналы

Сложные и непрерывные изменения микроструктуры среды приводят к появлению в ней неоднородностей двух типов: слоистых и глобулярных [9].

**Слоистые неоднородности** возникают при условиях, когда коэффициент преломления претерпевает скачок, например при наличии протяженных участков с различными температурами или плотностями. **Глобулярные неоднородности** являются результатом действия турбулентных процессов в среде, под действием которых в среде образуются локальные неоднородности (глобулы), в пределах которых коэффициент преломления несколько отличается от его значений в окружающей среде.

В общем случае форма таких неоднородностей произвольна, однако при построении упрощенных моделей среды принимают, что они имеют сферическую или эллипсоидальную форму. Такие неоднородности непрерывно возникают, перемещаются и исчезают. Интенсивность этого процесса, размер образующихся неоднородностей и их количество в единице объема зависят от характера и интенсивности турбулентных процессов.

Распространение сигнала в среде с неоднородностями сопровождается процессами отражения и рассеивания его энергии. В результате излученный сигнал, пройдя через среду, попадает на вход приемника по многим путям

(лучам). Это явление называется **многопутевым распространением**, или **многолучевостью**. Так как неоднородности среды, выполняющие роль элементарных «отражателей» или «рассеивателей» энергии сигнала, непрерывно и случайным образом меняются, то время прохождения сигнала от передатчика к приемнику по отдельным лучам различно и случайно. Это приводит к тому, что сигнал на входе приемника представляет собой сумму отдельных колебаний, время прихода и амплитуды которых случайны. Интерференция таких составляющих является основной причиной флуктуации параметров сигнала, принимаемого в условиях многолучевости. Наряду с отражением и рассеиванием происходит также поглощение энергии сигнала в зависимости от свойств среды, природы сигнала (электромагнитное излучение, акустические колебания и т. п.), несущей частоты сигнала и ширины его спектра, размеров неоднородностей и т. п.

Динамический характер микроструктуры среды приводит к случайным изменениям числа лучей, а также амплитуды и времени запаздывания каждой составляющей. Это, в свою очередь, вызывает случайные изменения многолучевого сигнала, которые проявляются в виде флуктуации его амплитуды и времени запаздывания.

Ряд таких важных параметров микроструктуры, как средний размер неоднородностей, их среднее число в единице объема среды (плотность неоднородностей), их средняя скорость перемещения в среде и т. п., полностью определяется состоянием и параметрами макроструктуры, которые остаются практически неизменными в течение длительных интервалов времени (от нескольких минут до нескольких часов и более). Это дает основание считать, что на таких интервалах времени указанный случайный процесс стационарен.

Для описания стационарных случайных процессов, которые используются в качестве математических моделей замираний сигналов в средах со случайно изменяющейся микроструктурой, используют ряд рассматриваемых далее характеристик и определений.

Время запаздывания сигнала, пришедшего на вход приемника по  $k$ -му лучу с номером  $k$ , можно представить как  $T_z = T_o + \Delta T$ , где  $T_o$  – среднее по всем лучам время запаздывания;  $\Delta T$  – случайное отклонение времени запаздывания по  $k$ -му лучу. Время многолучевого растяжения сигнала  $T_p = \max \Delta T - \min \Delta T$  характеризует разницу между максимальным и минимальным временем запаздывания при многолучевом распространении сигнала.

Непрерывные изменений микроструктуры среды приводят к тому, что величина  $\Delta T$  может принимать любое значение внутри интервала многолучевого растяжения  $T_p$  (случай непрерывной (дисперсной) многолучевости). Наряду с непрерывной может наблюдаться одновременно и дискретная многолучевость, вызванная слоистыми неоднородностями и границами раздела сред (например, тропосфера/ионосфера, тропосфера/земная поверхность, море/атмосфера и т. п.).

Эффект многолучевого растяжения приводит к тому, что каждой переданной посылке сигнала длительностью  $\tau_o$  на входе приемника соответствует посылка длительностью, увеличенной на время растяжения  $T_p$ , что вызывает взаимное перекрытие принимаемых символов или так называемую **межсимвольную интерференцию**. Заметим, что это явление является основным источником ошибок и в проводных ЛС (типа рассмотренного в предыдущем BSC канале), где такие явления вызываются паразитными реактивностями (рассредоточенными емкостями и индуктивностями), которые ограничивают полосу пропускания канала. Очевидно, что для исключения таких межсимвольных искажений необходимо выбирать  $\tau_o \gg T_p$ , однако, приводит к ограничению скорости передачи  $R = 1/\tau_o \ll 1/T_p$ .

Время многолучевого растяжения сигнала зависит от типа канала, протяженности линии связи, состояния среды, диаграмм направленности антенн и др. Например, в линиях дальней коротковолновой радиосвязи протяженностью 1500 – 4000 км это время составляет 1 – 3 мс. В случае

применения антенн с большим коэффициентом направленного действия в тропосферных линиях протяженностью 600 – 1000 км это время составляет около 5 мкс, в ионосферных линиях протяженностью 1000 – 3000 км – до 100 мкс.

К многолучевым каналам относятся и каналы с **доплеровским растяжением** спектра сигнала. Такие каналы характеризуются непрерывным изменением микроструктуры среды в виде случайных перемещений элементарных неоднородностей (отражателей и рассеивателей), что приводит к случайному доплеровскому смещению частоты спектральных составляющих сигнала. Так как неоднородности перемещаются с различными скоростями, а сигнал приходит на вход приемника по нескольким путям, то его спектр отличается от спектра переданного сигнала. Каждая спектральная составляющая принятого сигнала приобретает случайные изменения, что эквивалентно ее «размытию» в некоторой области частот  $\Delta f_d$ , называемой **доплеровским растяжением частоты**. Диапазон частот, в котором доплеровское растяжение примерно одинаково для всех составляющих спектра флуктуирующего сигнала, называется полосой **когерентности** среды или **полосой частот гладких замираний**. Для таких замираний характерна сильная (близкая к единице) корреляция замираний всех спектральных составляющих, в результате чего форма спектра сигнала остается неизменной, меняется лишь его масштаб. В этом случае говорят, что замирания протекают во времени **когерентно**.

Для выполнения условия когерентности необходимо, чтобы величина доплеровского растяжения  $\Delta f_d$  была значительно меньше ширины спектра сигнала  $\Delta f_c$ . Поскольку  $\Delta f_d \approx 1/\tau_o$ , то это условие можно записать как  $\tau_o \ll 1/\Delta f_d$ . Невыполнение этого условия приводит к тому, что скорость замираний спектральных составляющих сигнала становится разной, т. е. замирания на разных частотах становятся слабо коррелированными, что приводит к искажению формы сигнала. Такие явления называют, **селективными замираниями**.

Для снижения влияния многолучевого и доплеровского растяжений необходимо выполнить условия:  $\tau_o \ll T_p \ll 1/\Delta f_\partial$  или  $\tau_o \Delta f_\partial \ll T_p \Delta f_\partial \ll 1$ , на основании которых определяется **коэффициент растяжения**  $k_p = T_p \Delta f_\partial \ll 1$ . Для большинства каналов это условие выполняется. Однако в некоторых каналах, таких, например, как гидроакустические и каналы с использованием отражений от дипольных поясов, коэффициенты растяжения могут приближаться к единице, и в таких случаях замирания создают серьезные затруднения в приеме сигналов.

Модель многолучевого канала, реализованная в среде VisSim, представлена на рис. 4.5. Модель, кроме ранее рассмотренных, содержит следующие блоки:

1. Блоки 1 и 2 Variance для определения дисперсии (выход var) и среднего значения (выход mean) сигналов поступающих на вход in.
2. Преобразователь Cplx to Re/Im комплексного сигнала Z в его вещественную и мнимую составляющие.
3. Преобразователь Re/Im to Cplx вещественной и мнимой составляющих на входе канала в комплекс (без параметров).

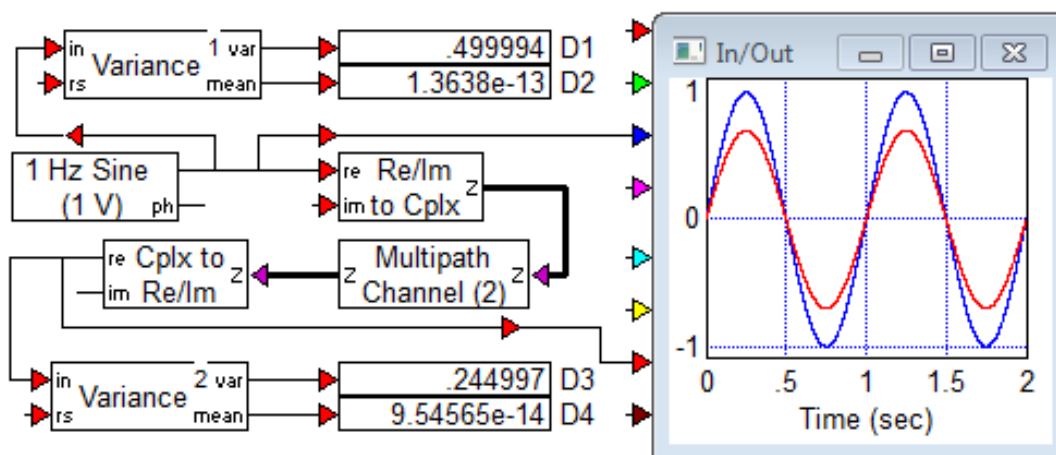


Рисунок 4.5. Схема для испытания многолучевого канала

4. Канал Multipath, выходной сигнал которого описывается формулой:

$$Z(t) = \sum_{k=1}^n a_k x(t - t_k) e^{j\varphi_k}, \quad (4.3)$$

где  $n = 1 - 4$  – число сигналов, поступающих на приемник с разных направлений (параметр Number of Paths в диалоговом окне; в данном случае  $n = 2$ );  $a_k$  – относительный вклад сигнала данного направления (параметр Weight;  $a_1 = 0,5$ ,  $a_2 = 0,2$ );  $x(t - t_k)$  – входной сигнал канала;  $t_k$  – задержка сигнала для каждого из 4-х направлений (параметр Path Delay; в нашем случае  $t_1 = t_2 = 0$ );  $\varphi_k$  – значение фазы сигнала для каждого направления (параметр Phase Rotation; для схемы на рис. 4.5  $\varphi_1 = \varphi_2 = 0^\circ$ ).

Таким образом, в модели многолучевого канала производится суммирование двух колебаний:  $x_1(t) = X_1 \sin(2\pi ft)$  и  $x_2(t) = X_2 \sin(2\pi ft)$ . Как известно из тригонометрии, амплитуда суммарного колебания равна  $X = [X_1^2 + X_2^2 + 2X_1X_2 \cos 0^\circ]^{1/2}$ . Поскольку величины  $X_1$  и  $X_2$  составляют соответственно 0,5 и 0,2 от амплитуды входного сигнала (1 В), то  $X_1 = 0,5$  В и  $X_2 = 0,2$  В. В таком случае  $X = [0,25 + 0,04 + 0,2]^{1/2} = 0,7$  В, что соответствует действующему значению  $0,7/(2)^{1/2} = 0,495$ . Поскольку дисперсия выходного сигнала, измеряемая регистратором D3, равна квадрату его действующего значения, то последнее равно  $(0,245)^{1/2} = 0,495$  В и совпадает с расчетным.

Если в блоке Multipath Cannel установить  $\varphi_2 = 90^\circ$ , то с помощью D3 получим дисперсию  $0,125 \text{ В}^2$  (рис. 4.6), т. е. действующее значение  $X$  равно  $0,354$  В, а амплитудное –  $0,354(2)^{1/2} = 0,5$  В. Расчетное же значение  $X = [X_1^2 + X_2^2 + 2X_1X_2 \cos 90^\circ]^{1/2} = [0,25 + 0,04 + 0]^{1/2} = 0,538$  В больше «экспериментального» и совпадает только в том случае, если пренебречь слагаемым  $[X_2^2 + 2X_1X_2 \cos 90^\circ]$ , т. е. блок Multipath Cannel «пренебрегает» сигналом с фазой, отличной от нуля. Этот факт отражается и на осциллограмме на рис. 4.6, где выходной сигнал по фазе совпадает с входным.

Если в блоке Multipath Cannel установить  $\varphi_2 = 0^\circ$ , и время задержки прихода сигнала по второму направлению  $t_2 = 0,125$  с, что, казалось бы, эквивалентно  $\varphi_2 = 90^\circ$ , то получим (с помощью D3) дисперсию  $0,215858 \text{ В}^2$ , что соответствует амплитуде  $0,657$  В. Однако это больше расчетного  $0,538$  В.

Осциллограмма для рассматриваемого случая показана на рис. 4.7, где с помощью визира, вызываемого кнопкой Read Coordinates на закладке Options диалогового окна плоттера, измерены координаты пересечения выходным сигналом оси времени.

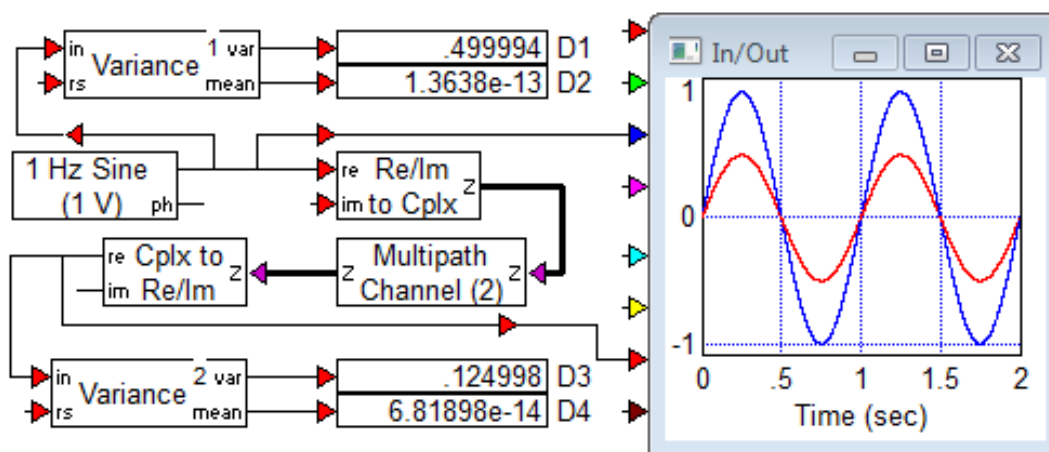


Рисунок 4.6. Состояние модели многолучевого канала при  $\varphi_2 = 90^\circ$

Из полученных данных видно, что задержка сигнала на выходе канала меньше установленного и составляет 0,048 с; более точное значение, которое получается в увеличенном масштабе плоттера (после нажатия его кнопки ) , равно 0,0336 с, что эквивалентно запаздыванию по фазе  $360 \times 0,0336 / 1 = 12,1^\circ$ . Таким образом, алгоритм функционирования блока Multipath Cannel более сложный, чем это дается формулой (4.3), приводимой в справочной системе программы.

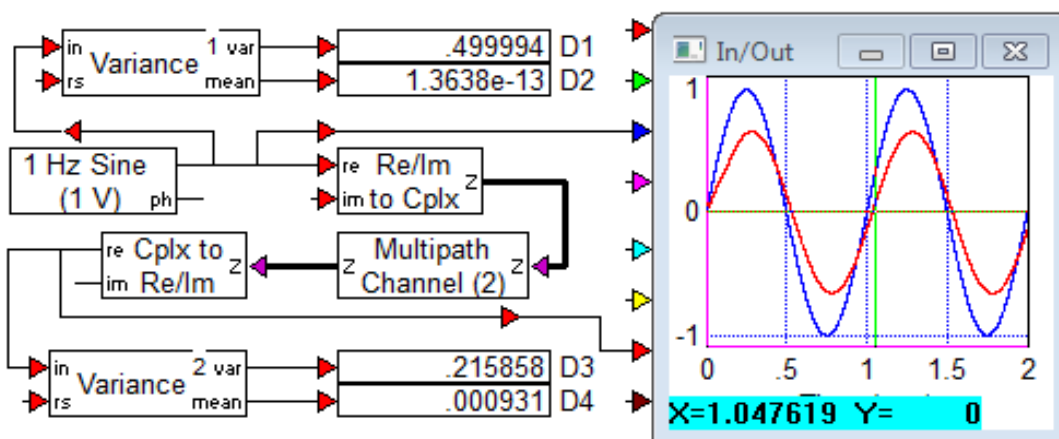


Рисунок 4.7. Состояние модели многолучевого канала при  $t_2 = 0,25$  с

### 4.2.3. Райсовский и релеевский каналы

Для характеристики изменений микроструктуры среды используются понятия глубины и скорости замирания флуктуирующего сигнала на входе приемника [9]. **Глубину замираний** принято характеризовать изменением уровня огибающей относительно ее **медианного значения**, определяемого уровнем огибающей сигнала, при котором в течение сеанса связи суммарное время пребывания процесса, как выше, так и ниже этого уровня, одинаково. Глубина замираний в линиях связи большой протяженности может достигать 20 – 30 дБ.

**Скорость замираний** принято характеризовать средним числом односторонних пересечений (снизу вверх или сверху вниз) в секунду огибающей сигнала с медианным уровнем. Эту величину можно также трактовать как среднюю частоту замираний или как среднее число выбросов огибающей, превышающих медианный уровень в течение 1 с. Для измерения скорости замираний обычно используют излучение непрерывной немодулированной несущей с фиксацией частоты пересечения огибающей принимаемого флуктуирующего сигнала с ее медианным уровнем. По записям флуктуирующей огибающей можно получить энергетический спектр или корреляционную функцию замираний, а по их значениям – скорость замираний. При этом нормированную корреляционную функцию замираний, построенную по экспериментальным данным, аппроксимируют выражением:  $\rho(\tau) = \exp(-\tau^2/2\tau_\phi)^2$ , где  $\tau_\phi$  – параметр, однозначно связанный со скоростью замираний. Если время корреляции процесса определять по используемой в статистической радиотехнике формуле  $\tau_k = \int_0^\infty \rho(\tau) d\tau$ , то для указанной аппроксимации будем иметь  $\tau_k = \tau_\phi (\pi/2)^{1/2} \approx 1,25\tau_\phi$  и  $\rho(\tau_k) \approx 0,45$ . При этом полученное время корреляции можно рассматривать как средний период замираний.



Многочисленные экспериментальные исследования показывают, что длительность замираний зависит от частоты несущего сигнала, протяженности ЛС, диаграмм направленности антенн, типа канала и может колебаться от нескольких секунд до 0,1 – 0,3 с. Кроме того, установлено, что на интервале, равном длительности посылки сигнала, микроструктура среды не успевает существенно измениться, так как за это время могут произойти лишь относительно небольшие случайные перемещения неоднородностей. Поэтому амплитуду сигнала по каждому лучу в течение длительности посылки практически можно считать неизменной случайной величиной, значение которой медленно изменяется от посылки к посылке. Фаза же этого сигнала даже при незначительных перемещениях неоднородностей может меняться значительно, поворачиваясь на угол  $2\pi$  каждый раз, когда путь, по которому проходит этот сигнал, изменяется всего лишь на длину волны несущего колебания.

Неоднородности на пути каждого луча перемещаются случайно и изменения фаз сигналов, приходящих по каждому лучу, происходят также случайно. В некоторых случаях замирающий сигнал может содержать регулярную составляющую, обусловленную отражением и рассеянием сигналов от слоистых неоднородностей и границ раздела сред с различающимися физическими свойствами.

Кроме того, при определенных условиях по некоторым направлениям возможно и когерентное распространение сигнала, когда фазы по отдельным лучам незначительно отклоняются от своего среднего значения. В таких случаях сигнал на входе приемника на интервале  $0 \leq t \leq \tau$  ( $\tau$  – длительность посылки) может быть представлен в виде суммы регулярной  $s_p(t)$  и флуктуирующей  $s_\phi(t)$  составляющих, т. е.  $s_x(t) = S_x f_x(t) = s_p(t) + s_\phi(t)$ , где  $S_x$  – огибающая посылки;  $f_x(t)$  – сигнальная функция, учитывающая высокочастотное заполнение посылки и соответствующие случайные сдвиги фазы при распространении.

Если ни одна составляющая многолучевого сигнала не содержит регулярной части (случай диффузно отраженных лучей), то канал с такими условиями распространения называют **релеевским**. При этом распределение флуктуации подчиняется закону Релея:

$$w(r) = (r/\sigma^2) \exp[-0,25(r/\sigma)^2], \quad (4.4)$$

где  $\sigma^2$  – дисперсия огибающей элементарной площадки на расстоянии  $r \geq 0$  (в полярных координатах) от источника помехи. Примерами релеевского канала являются каналы связи с рассеянной волной (тропосферные, ионосферные, метеорные и др.). Если многолучевой сигнал содержит, кроме диффузно отраженных, и регулярный (незамирающий) луч, то распределение флуктуации подчиняется обобщенному закону Релея (или закону Райса) (к райсовским каналам, например, относится ряд каналов с прямой волной и остронаправленными диаграммами антенн):

$$w(r) = (r/\sigma^2) \exp[-(r^2 + \alpha^2)/\sigma^2] \times I_0(\alpha r/\sigma^2), \quad (4.5)$$

где  $\alpha = (x^2 + y^2)^{1/2}$  – огибающая (среднеквадратичное значение ожидания квадратурных компонент (по осям  $X$  и  $Y$ )) регулярной составляющей;  $I_0(\alpha r/\sigma^2)$  – модифицированная функция Бесселя первого рода нулевого порядка.

Следует заметить, что более гибким инструментом при аппроксимации экспериментально получаемых распределений является закон Накагами, предложенный в 1943 году. Он известен под названием « $m$ -распределение» или «распределение Накагами» и выглядит следующим образом:

$$w(r) = [2/\Gamma(m)] (m/2\sigma^2)^m (r)^{2m-1} \times \exp[-(r/2\sigma^2)^2], \quad (4.6)$$

где  $\Gamma(m)$  – гамма-функция.

При  $m = 0,5$  из формулы (4.6) получается нормальный закон с односторонней плотностью распределения, при  $m = 1$  – закон распределения Релея (4.4). При  $m > 1$  распределение Накагами становится достаточно хорошей аппроксимацией закона Райса (4.5). Таким образом, распределение Накагами позволяет описать более широкий класс распределений, чем, например, закон Райса.

Рассмотренные процессы достаточно наглядно иллюстрируются в среде VisSim/Comm моделью канала Rice/Rayleigh, имитирующей тропосферное и ионосферное рассеивание. Тип модели выбирается в диалоговом окне параметром Rice Factor ( $r$ ): при  $r = 0$  модель является релейской, при  $r = 1$  – райсовской. Схемы для исследования каналов, содержащие уже известные нам блоки, представлены на рис. 4.8 – для релейской модели и на рис. 4.9 – для райсовской.

Результаты моделирования в виде осциллограмм входного сигнала, выходного и его фазы получены при следующих значениях параметров моделей: действующее значение доплеровского смещения частоты (RMS Doppler Shift) – 0,5 Гц; среднеквадратичное значение затухания канала (RMS Fade Level) – 0 дБ.

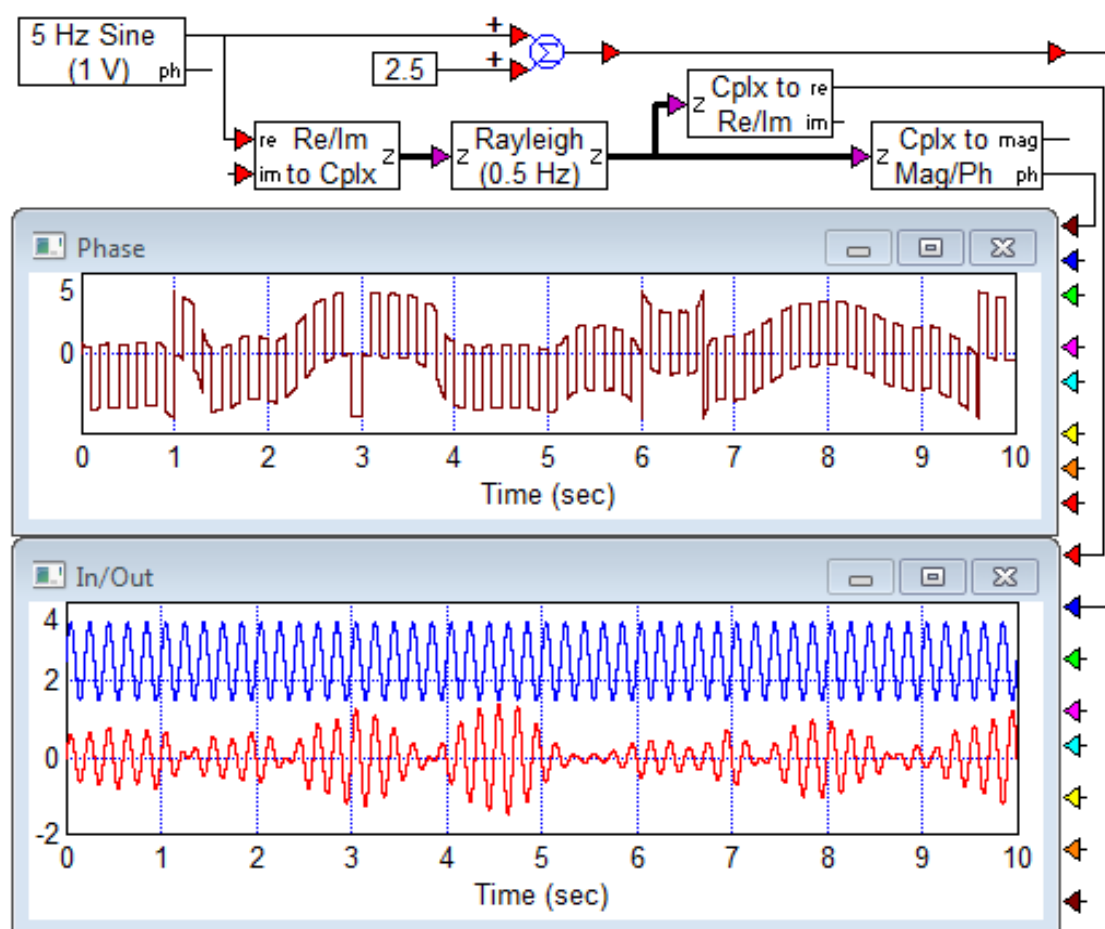


Рисунок 4.8. Результаты испытания канала Rayleigh

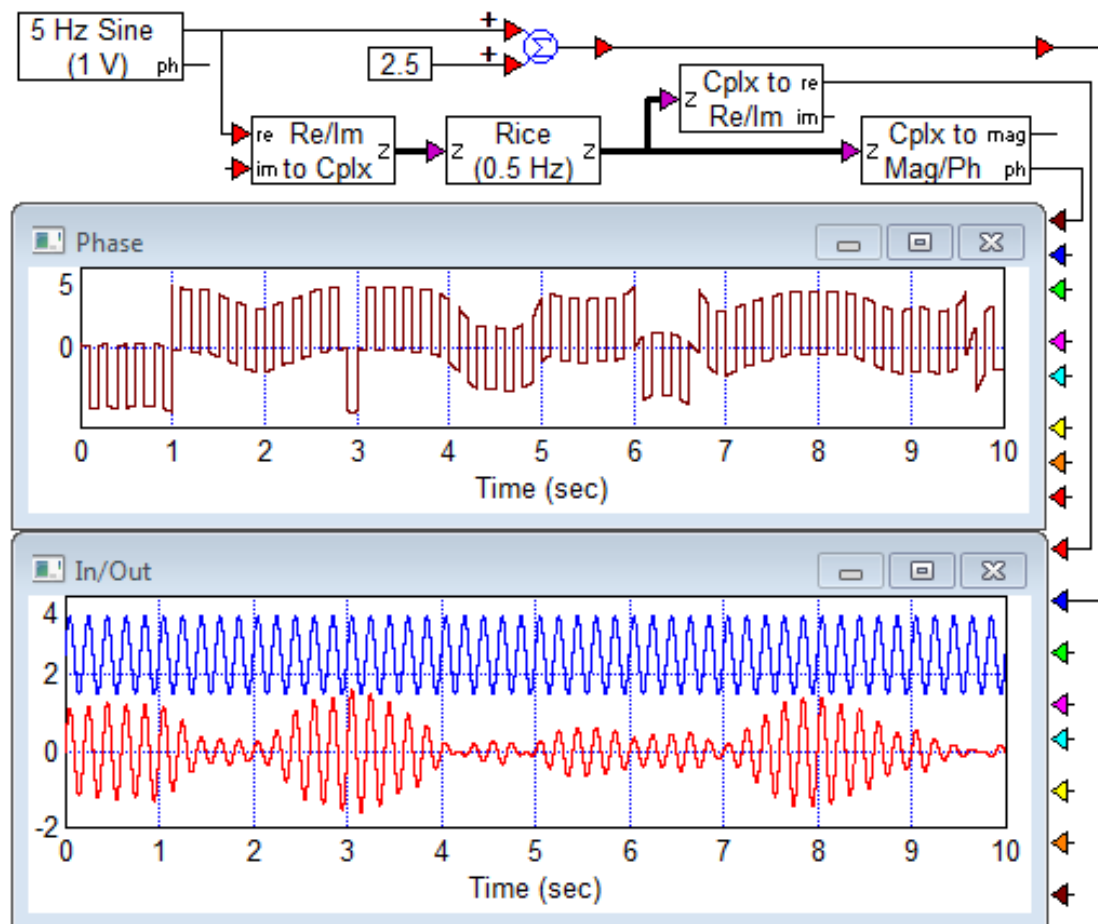


Рисунок 4.9. Результаты испытания канала Rice

Из сравнения результатов измерений на рис. 4.8 и 4.9 следует, что

- изменение фазы в райсовском канале меньше, чем в релеевском;
- сигналы на выходе каналов по форме напоминают АМ-сигнал со случайно изменяющимся коэффициентом модуляции, несмотря на то, что испытательные сигналы имеют детерминированный характер;
- уровень замираний в райсовском канале меньше, чем в релеевском.

### 4.3. Лабораторная работа 4. Исследование особенностей работы каналов передачи информации

#### 4.3.1. Цель и содержание

Целью лабораторной работы является приобретение студентами навыков исследования принципов функционирования каналов передачи информации.

## Содержание

1. Моделирование канала с затуханием сигнала в свободном пространстве.
2. Исследование особенностей работы схемы бинарного симметричного канала.
3. Исследование особенностей работы схемы многолучевого канала.
4. Исследование моделей райсовского и релеевского каналов.

### 4.3.2. Аппаратура и материалы

Для изучения материалов этой лабораторной работы необходимо:

1. Компьютер под управлением операционной системы Windows XP/Vista/7 с параметрами по умолчанию.
2. Установленная инструментальная среда визуального моделирования динамических систем VisSim/Comm 7.0.
3. Текстовый процессор (Microsoft Word 2003/2007/2010 или его аналог), предназначенный для создания и редактирования отчета по лабораторной работе.

### 4.3.3. Порядок выполнения работы

1. Изучение теоретического материала.
2. Выполнение задания, предлагаемого в лабораторной работе.

#### Задание

1. Запустить на выполнение файл 4-1-1.VSM. Провести моделирование канала с затуханием сигнала в свободном пространстве. Охарактеризовать особенности работы отдельных элементов схемы.
2. Запустить на выполнение файл 4-2-1.VSM. Провести моделирование схемы бинарного симметричного канала при вероятности ошибки передачи одиночного бита (параметр Channel Error Probability) BSC канала 0,2; 0,4 и 0,8. Провести для этих случаев расчет вероятности  $p(q)$  безошибочной передачи последовательности из 8 бит.

3. Запустить на выполнение файл 4-2-2.VSM. Провести моделирование схемы многолучевого канала. Путем варьирования фазы  $\varphi_2$  и параметра  $a_2$  установить их значения, при которых блок Multipath Channel начинает «учитывать» сигнал второго направления.

Варьируя значения задержки сигнала второго направления  $t_2$ , получить зависимость действующего значения выходного напряжения канала (по показаниям D3 на рис. 4.5) от этого параметра.

4. Запустить на выполнение файл 4-2-3.VSM. Провести моделирование схем райсовского и релеевского каналов. Используя результаты осциллографических измерений в режиме Read Coordinates (закладка Options диалогового окна плоттера), определить глубину замираний выходного сигнала райсовского и релеевского каналов.

Используя результаты осциллографических измерений определить скорость замираний выходного сигнала райсовского и релеевского каналов.

Провести моделирование каналов при доплеровском смещении частоты (RMS Doppler Shift) 1 Гц.

#### 4.3.4. Вопросы для защиты работы

1. Проведите классификацию каналов по форме передаваемых сигналов и эксплуатационным требованиям.
2. Охарактеризуйте области применения радиоканалов.
3. Какие устройства применяют в качестве источников и детекторов световых сигналов в системах с ВОЛС?
4. Проведите классификацию каналов со случайными параметрами.
5. Назовите виды искажений в дискретных каналах.
6. Какие виды неоднородностей вызывают замирания сигналов в свободном пространстве?
7. Чем вызывается многолучевое распространение сигнала? Для каких типов сигналов оно характерно?

8. Что такое межсимвольная интерференция сигналов? Чем она вызывается в случае проводных линий связи?
9. Охарактеризуйте многолучевые каналы с доплеровским растяжением спектра сигнала и полосу когерентности для таких каналов?
10. Что понимается под глубиной замираний?
11. Что такое скорость замираний?
12. Назовите отличия райсовского канала от релеевского. Приведите примеры реальных каналов связи, которые могут быть описаны данными моделями.

## ГЛАВА 5. УПЛОТНЕНИЕ И РАЗДЕЛЕНИЕ КАНАЛОВ

Системы телекоммуникаций являются, как правило, многоканальными, т. е. в них по одной физической ЛС путем ее **уплотнения** передается большое число независимых сообщений [9]. Многоканальная передача информации характерна практически для всех частотных диапазонов проводных, радио- и оптоволоконных систем. Уплотнение одного канала с высокой пропускной способностью особенно эффективно при передаче на дальние расстояния объемных потоков независимых сообщений, так как строить в этих условиях большое количество независимых каналов с малыми пропускными способностями невыгодно.

Общая идея уплотнения линии связи заключается в том, что индивидуальные сигналы соответствующих сообщений объединяются в блоке уплотнения и образуют суммарный или **групповой сигнал**, который до подачи в канал преобразуется при помощи дополнительной модуляции того или иного вида в линейный сигнал. В месте приема он восстанавливается после первой демодуляции, после чего суммарный сигнал (сигнал всех каналов + помеха) поступает на устройство разделения каналов, а затем сигналы отдельных каналов подвергаются индивидуальной обработке (демодуляции и декодированию). В некоторых случаях функции разделения каналов и обработки индивидуального сигнала выполняются совместно.

По способу образования группового сигнала и принципам разделения каналов на приеме системы уплотнения делятся на отдельные, комбинационные и смешанные.

В **раздельных** системах уплотнения групповой сигнал образуется путем линейного сложения индивидуальных сигналов на передающей стороне. При этом на приемной стороне возможны 4 способа линейного разделения: по частоте, по времени, по фазе и по форме сигнала. Иногда используются нелинейные схемы разделения канальных сигналов по их уровням. Очевидно, что для каждого индивидуального канала сигналы всех остальных каналов (как



и аддитивный шум) представляют собой помехи, которые называют **переходными**. При большом числе независимых каналов уплотнения переходные помехи имеют характер нормального флуктуационного шума. При этом переходную помеху нельзя уменьшить повышением мощности сигнала, так как это повлечет за собой рост мощности помехи. Системы раздельного уплотнения применяются для передачи как дискретных, так и непрерывных сообщений. Они нашли преимущественное применение на практике вследствие простоты каналообразующей аппаратуры.

**Комбинационные** системы уплотнения применяются для многоканальной передачи дискретных сообщений. В этих системах как групповой сигнал, так и его отдельные составляющие не могут считаться суммой индивидуальных сигналов. Групповой сигнал здесь определяется совокупностью сочетаний символов в индивидуальных каналах.

**Смешанные** системы уплотнения характеризуются одновременным использованием раздельных и комбинационных принципов уплотнения.

Кроме рассмотренных классификационных признаков, системы уплотнения делятся на синхронные и асинхронные.

В **синхронных** системах информация от отдельных источников передается с одинаковыми или кратными скоростями и строго соблюдается смена начала и конца элементов сигнала, длительность которых одинакова. Современные достижения техники частотной и фазовой стабилизации и взаимной синхронизации делают синхронные системы уплотнения весьма перспективными при передаче дискретных сообщений.

В **асинхронных** системах уплотнения отдельные источники имеют относительно свободный доступ в канал, скорость выдачи информации может варьироваться в широких пределах, при этом отпадает необходимость в устройствах согласования с уплотняемым каналом и заметно понижаются требования к стабильности аппаратуры. Однако асинхронные системы уплотнения менее эффективно используют частотные и временные интервалы, отведенные для связи.

Для оценки эффективности систем уплотнения используются следующие показатели.

1. Энергетический коэффициент уплотнения  $\eta(n) = P_n / P_1$ , также называемый «энергетической ценой» – показывает, во сколько раз необходимо увеличить среднюю мощность многоканального сигнала  $P_n$  и сигнала одного канала  $P_1$  чтобы при уплотнении и заданной вероятности ошибки при передаче дискретных сообщений или отношении сигнал/помеха на выходе детектора при передаче непрерывных сообщений можно было передавать  $n$  сообщений вместо одного. Например, для большинства систем с частотным уплотнением можно считать, что в линейном сигнале сигналы индивидуальных каналов независимы и  $P_n \approx nP_1$ , т. е. энергетическая цена уплотнения  $\eta(n) = n$ . В случае же использования радиолинии пиковая мощность каждого  $P_{1m}$  и суммарного  $P_{nm}$  каналов связана соотношением:  $P_{1m} = P_{nm} / n^\chi$ , где  $\chi = 1 - 2$  – коэффициент, характеризующий нелинейность передатчика.

2. Коэффициент уплотнения по частоте («спектральная цена»)  $\lambda(n) = F_n / F_1$  – показывает, во сколько раз возрастает полоса частот, занимаемая многоканальным сообщением  $F_n$  по сравнению с одноканальным  $F_1$  при условии сохранения вида модуляции и постоянстве  $F_1$  при передаче любого из сообщений. При частотном уплотнении  $\lambda(n) > n$ , кроме того, это условие усиливается из-за необходимости оставлять защитные частотные интервалы между сигналами каждого канала.

3. Количество информации, передаваемой системой за время равное 1 с в полосе частот 1 Гц.

Процедура разделения каналов является неотъемлемой частью многоканальных систем и заключается в том, чтобы сигнал каждого источника информации, передаваемый по линии связи в виде составного (суммарного) сигнала нескольких каналов, мог попасть в свой приемник. Признаками разделения могут служить принадлежность к определенной электрической цепи, частота или фаза носителя, положение на временной оси, форма и т. д.

### 5.1. Пространственное и дифференциальное разделение

Пространственное разделение является простейшим. При таком разделении каждому каналу отводится индивидуальная ЛС и, естественно, в этом случае об уплотнении не может быть и речи [9].

Дифференциальное разделение используется, например, для разделения в телефонных каналах связи разговорного тракта и сигналов управления (вызов, отбой и т. п.). В такой системе (рис. 5.1), схема которой выполнена в среде NI Multisim, на передающей стороне используется дифференциальный трансформатор T1, а на приемной – такой же по конструкции T2.

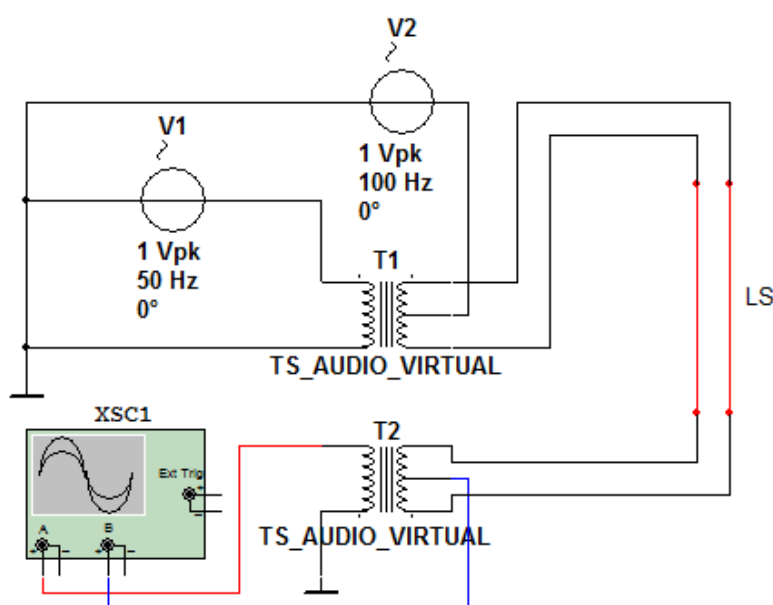


Рисунок 5.1. Схема дифференциального разделения каналов

Источник сигнала V1 частотой 50 Гц первого канала подключается к первичной обмотке T1, источник V2 частотой 100 Гц второго канала – между средней точкой вторичной обмотки T1 и общей шиной (землей). На приемной стороне такой же трансформатор T2 подключается к линии связи LS в «перевернутом» виде, поскольку теперь уже первичная обмотка имеет отвод от своей середины, с которого снимается сигнал второго канала (регистрируется в канале B осциллографа); сигнал первого канала при этом снимается со вторичной обмотки T2 (канал A осциллографа).

Как видно из осциллограмм на рис. 5.2, сигналы передаются в приемник получателя без искажений. Отсутствие взаимного влияния каналов объясняется тем, что для сигнала второго канала (100 Гц) полуобмотки трансформаторов включены встречно, т. е. магнитные потоки от протекающих по этим полуобмоткам токов взаимно уничтожаются. По отношению же к первому каналу они включены согласно.

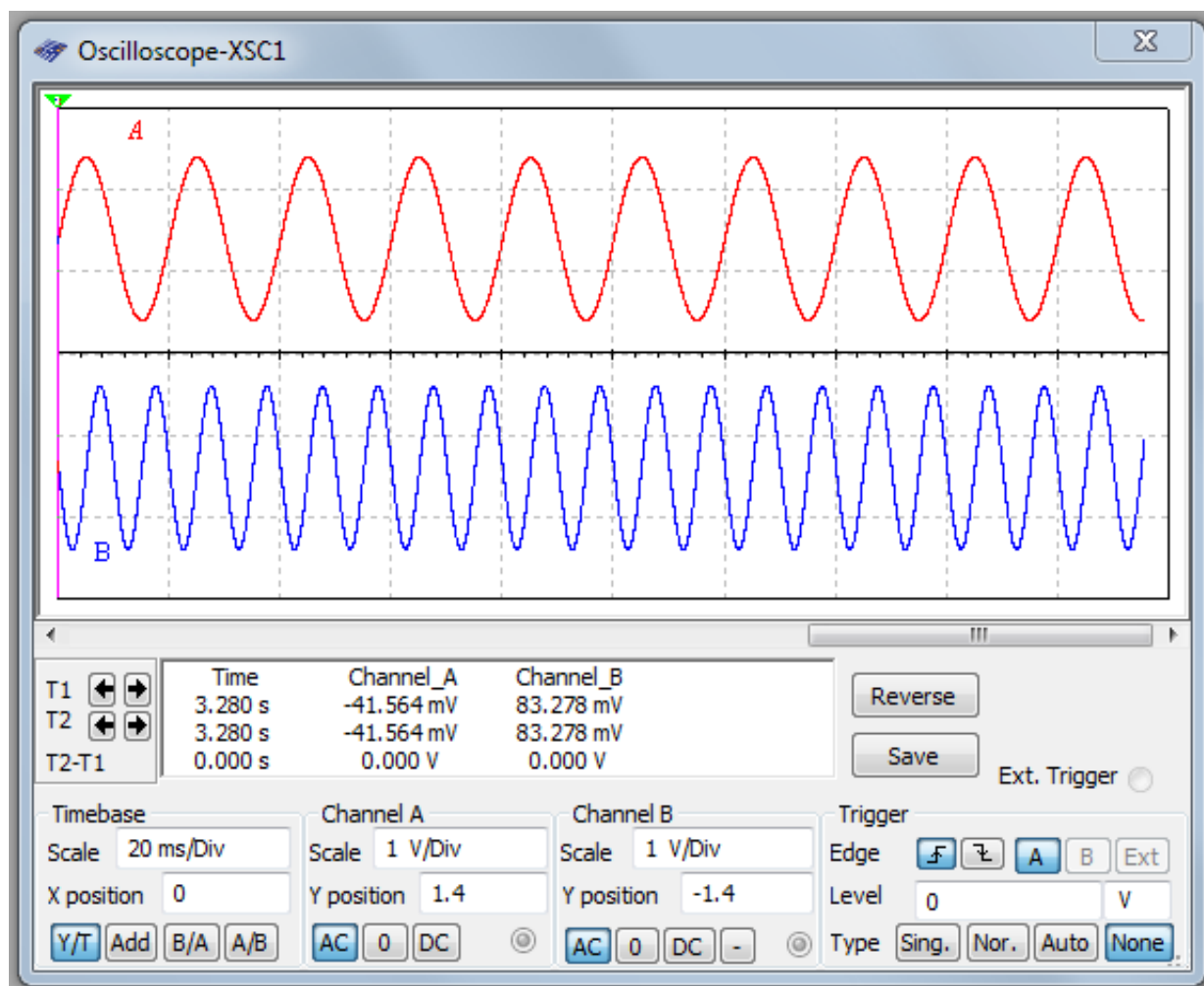


Рисунок 5.2. Осциллограммы на приемной стороне

## 5.2. Частотное разделение

При частотном разделении для различных каналов отводятся непересекающиеся участки на частотной шкале [9]. Спектры сигналов и соответствующих каналов при этом должны укладываться в пределы полосы пропускания ЛС. Принцип частотного разделения поясняется схемой

реализованной в среде VisSim (рис. 5.3), где передающая сторона представлена двумя источниками несущих сигналов частотой 5 и 10 Гц, которые могут быть промодулированы информационными сигналами, а также сумматором СУМ1. Приемная сторона представлена двумя полосовыми фильтрами Bandpass FIR с центральной частотой 5 Гц для первого канала и 10 Гц для второго при полосе пропускания 2 Гц (определяется параметрами Cutoff Freq 1 и Cutoff Freq 2, значения которых индицируются на значке блока).

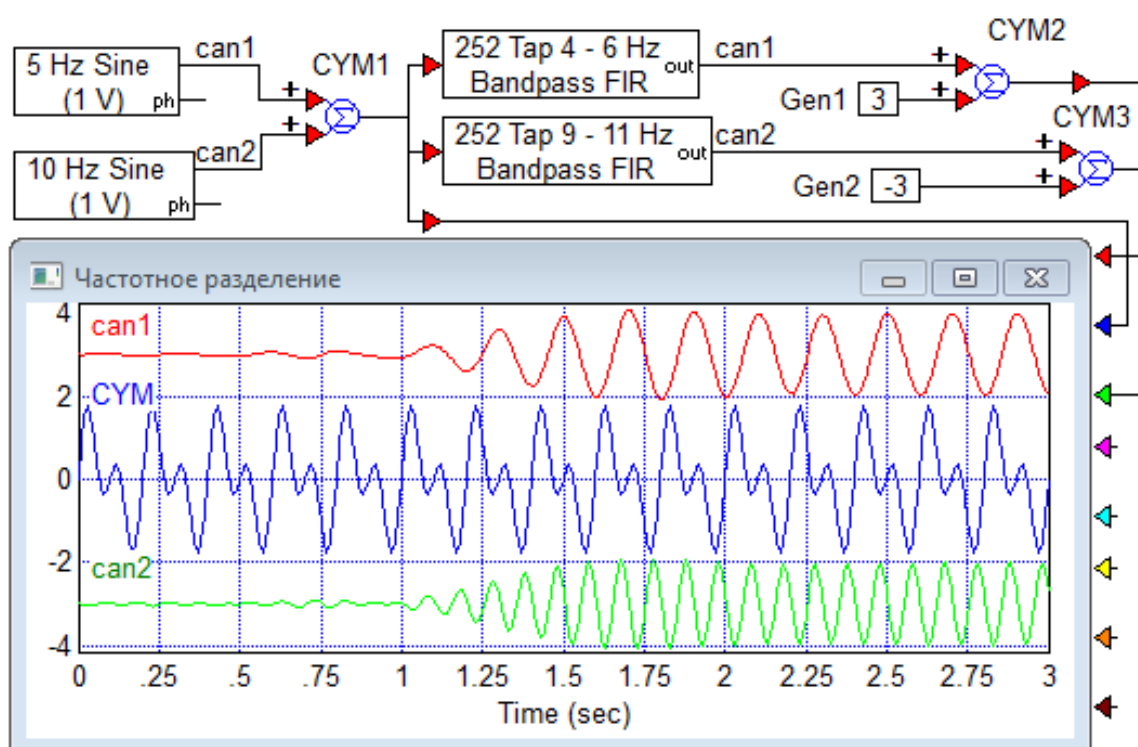


Рисунок 5.3. Частотное разделение каналов

Получателем на приемной стороне является плоттер, осциллограммы которого отображают, если не учитывать переходные процессы в фильтрах, принятые сигналы, а также сигнал СУМ на входе приемника.

Преимуществами частотного разделения являются возможность одновременной передачи сигналов, относящихся к разным каналам и возможность передачи сигналов от рассредоточенных объектов. К недостаткам относится сравнительно большое взаимное влияние каналов из-за перекрытия спектров сигналов, неидеальности полосовых фильтров и паразитных

комбинационных составляющих, вызванных нелинейностью электрических цепей (так называемая перекрестная модуляция).

В зависимости от назначения и типа ЛС в системах с частотным уплотнением (разделением) используются различные виды модуляции, как для информационных сигналов каждого канала, так и для суммарного сигнала, что отображается в таких названиях, как ЧМ-АМ, ЧМ-ЧМ и др. Первая аббревиатура обозначает вид модуляции несущей для каждого канала (ее называют поднесущей), вторая – вид модуляции несущей суммарного (многоканального) сигнала. Частоты поднесущих выбираются так, чтобы спектры сигналов каждого канала не перекрывались и, кроме того, были разделены защитными интервалами. Несмотря на большое разнообразие практических систем, при частотном уплотнении длительность канальных сигналов определяется длительностью сообщений, которые во времени могут иметь произвольное взаимное расположение, т. е. системы с классическим частотным уплотнением относятся к разряду асинхронных.

### 5.3. Временное разделение

При временном разделении несущие сигналы отдельных каналов передаются только в отведенные для них непересекающиеся отрезки времени, которые задаются коммутаторами-распределителями [9]. В модели выполненной в среде NI Multisim (рис. 5.4) такие распределители представлены ключами J1t, J2t на передающей стороне и J1r, J2r – на приемной.

Ключи управляются противофазными источниками V1t, V2t на передающей стороне и синхронизированными с ними V1s, V2s – на приемной. Поскольку напряжения срабатывания ключей выбраны достаточно низкими (0,1 В), то практически в течение всего положительного полупериода V1t и V1r ключи J1t и J1r замкнуты (с небольшим «защитным» интервалом, обусловленным порогом срабатывания 0,1 В). При этом на первый вход сумматора на операционном усилителе (ОУ) О1 поступает очередная выборка сигнала первого канала (осциллограмма U1d на рис. 5.5).

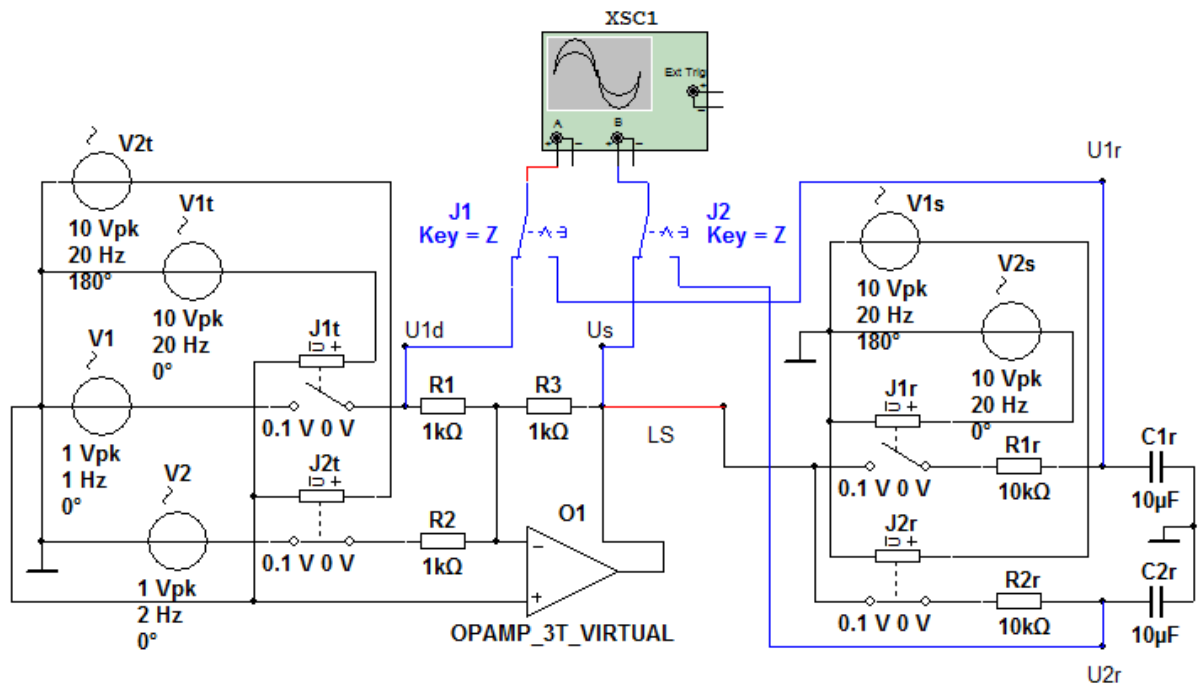


Рисунок 5.4. Схема моделирующая временное разделение каналов

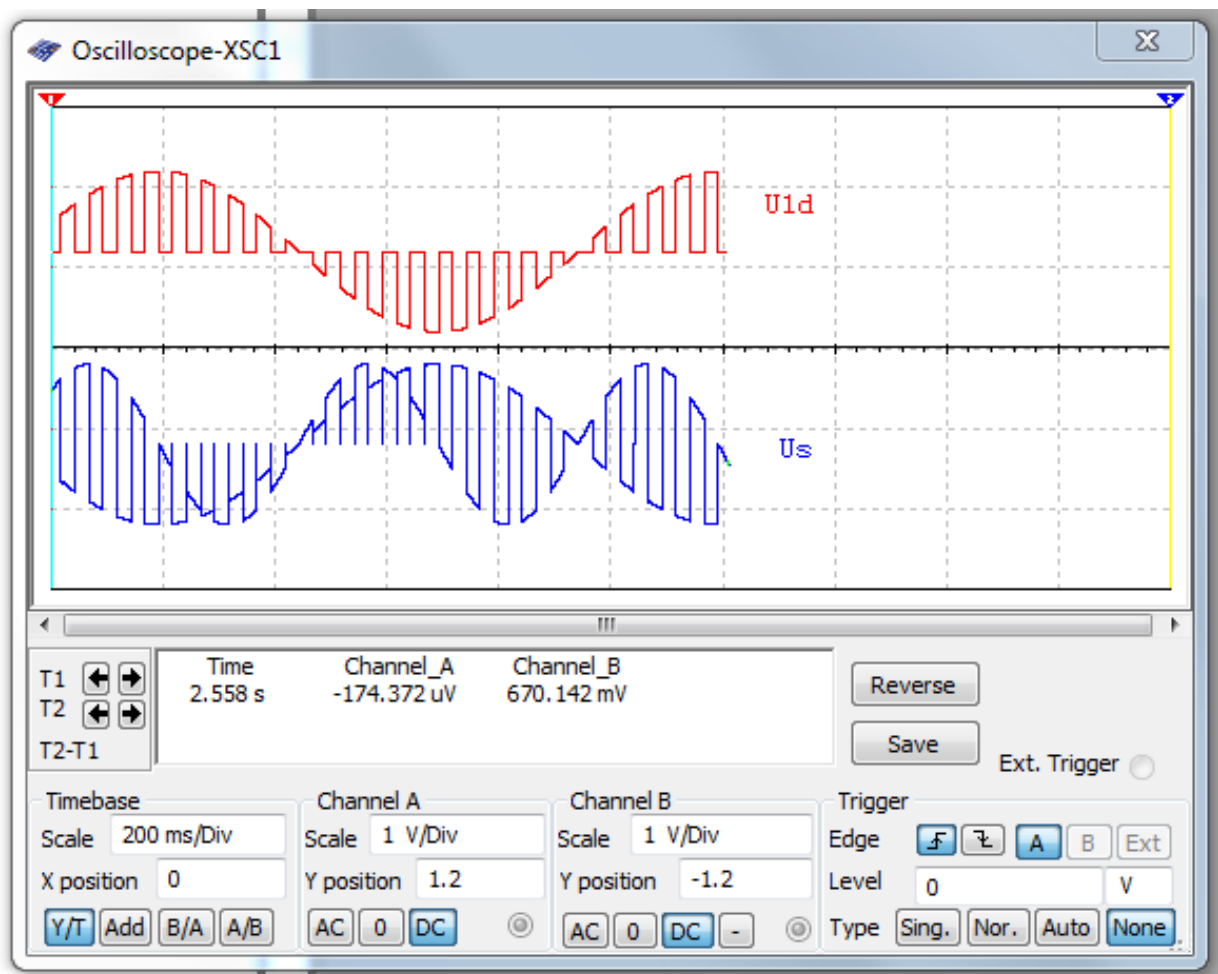


Рисунок 5.5. Осциллограммы на передающей стороне

С выхода сумматора суммарный сигнал (осциллограмма  $U_s$  на рис. 5.5) через линию связи и ключ  $J_{1r}$  поступает на низкочастотный RC-фильтр первого канала (элементы  $R_{1r}$  и  $C_{1r}$ ), полученный после фильтра сигнал представлен на осциллограмме  $U_{1r}$  (рис. 5.6). В течение положительного полупериода сигналов генераторов  $V_{2t}$  и  $V_{2r}$ , следующего непосредственно за рассмотренным (вследствие противофазности напряжений источников управления), замкнутыми оказываются ключи  $J_{2t}$  и  $J_{2r}$ , в результате чего очередная выборка сигнала генератора  $V_2$  второго канала через сумматор и ЛС поступает на низкочастотный RC-фильтр второго канала (элементы  $R_{2r}$  и  $C_{2r}$ ), что отражено на осциллограмме  $U_{2r}$  (рис. 5.6).

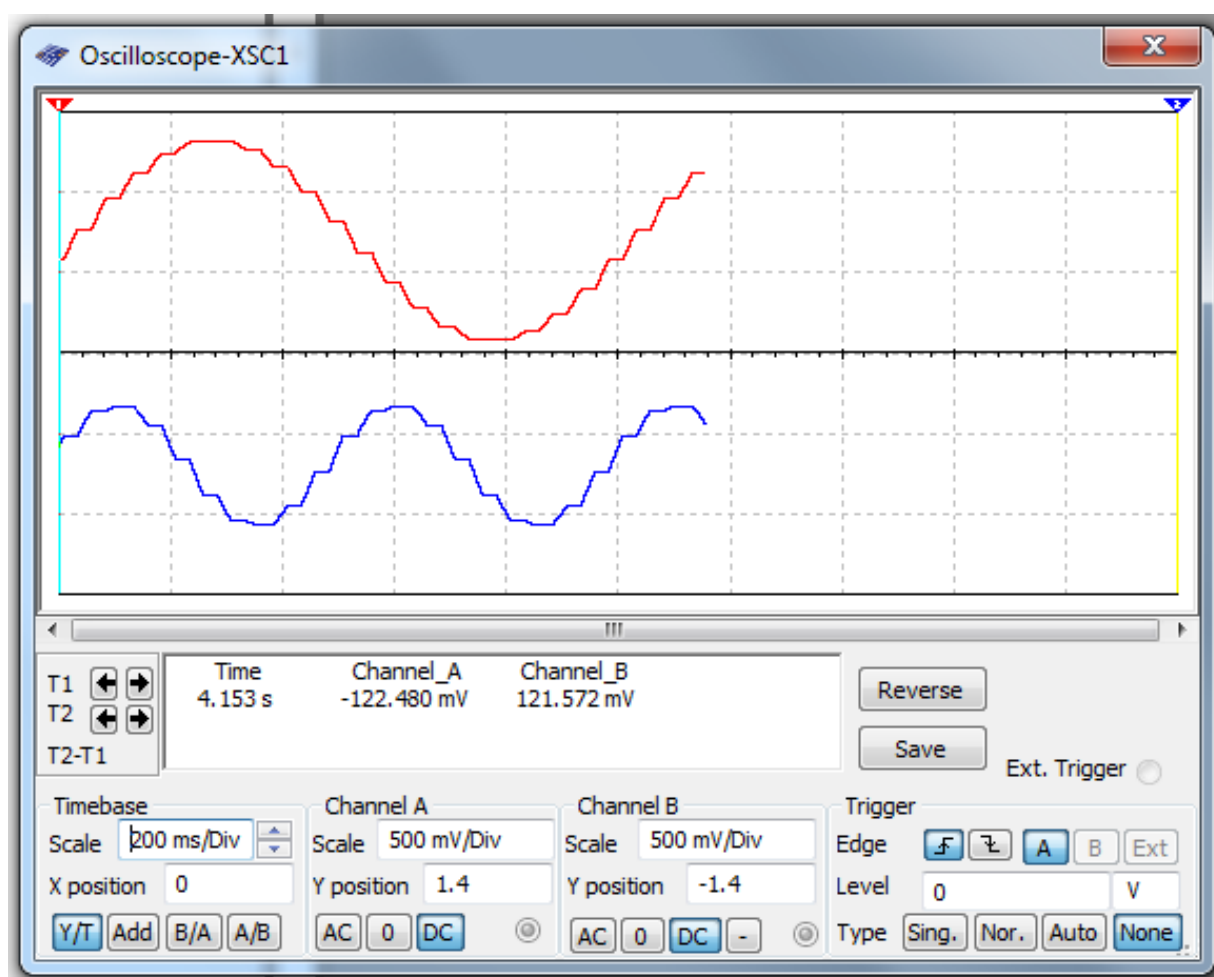


Рисунок 5.6. Осциллограммы на приемной стороне

Из проведенного рассмотрения видно, что временное разделение каналов сводится к использованию принципов амплитудно-импульсной модуляции



(АИМ) с соответствующим выбором интервалов дискретизации. В практических системах временного уплотнения/разделения выборки АИМ кодируются, при этом система называется КИМ (кодоимпульсная модуляция).

Достоинством систем временного уплотнения является сравнительная простота аппаратной реализации (особенно в случае АИМ) и незначительное взаимное влияние каналов, что позволяет строить системы с большим количеством каналов. Правда, из-за ограниченности полосы пропускания реального канала импульсные сигналы на его выходе могут «расплываться», что вызывает переходные помехи между соседними каналами.

Для снижения влияния переходных помех расширяют полосу пропускания группового тракта с введением защитных временных интервалов между импульсами индивидуальных каналов или уменьшают число каналов.

В случае радиоканала причиной переходных помех может быть также многолучевое распространение радиоволн, в результате чего запаздывающий луч, несущий сигнал одного канала, может интерферировать с лучом другого канала. Для защиты от таких помех часто вводят защитный интервал, равный максимальному времени запаздывания между лучами с соизмеримой интенсивностью.

Энергетическая цена временного уплотнения  $\eta(n)=n$ . При передаче непрерывных сообщений и использовании ФИМ-АМ, КИМ-АМ, для которых характерно постоянство длительности отдельных импульсов, это объясняется тем, что число излучаемых импульсов в единицу времени пропорционально  $n$ , а необходимая энергия каждого импульса (для обеспечения заданной верности) определяется условием превышения порога помехоустойчивости, т. е. зависит лишь от спектральной плотности помехи  $G_{ш}$ . Поскольку с изменением числа каналов уплотнения  $G_{ш}$  не меняется, то необходимая энергия сигнала в единицу времени (его средняя мощность) пропорциональна  $n$ .

При передаче дискретных сообщений для сохранения неизменной вероятности ошибки при флуктуационной помехе с заданной спектральной плотностью  $G_{ш}$  необходимо сохранение постоянства энергии посылки.

Поскольку при временном уплотнении длительность сигнала сокращается в  $n$  раз, то для выполнения этого условия необходимо увеличивать мощность сигнала также в  $n$  раз, т. е.  $\eta(n)=n$ . Эти же соображения позволяют считать, что спектральная цена временного уплотнения  $\lambda(n) \approx n$ , если пренебречь межсимвольной интерференцией. Количество информации, передаваемой системой временного и частотного уплотнения за время равное 1 с в полосе частот – 1 Гц, примерно одинаково и при дискретных сообщениях в двоичном коде меньше 1 бит/с  $\times$  Гц.

Следует отметить, что одной из важных проблем систем временного уплотнения является обеспечение синхронной работы тактовых генераторов (источники V1t, V2t, V1s, V2s на рис. 5.4) на передающей и приемной стороне.

#### 5.4. Фазовое разделение

Фазовое разделение применено в двухканальной системе с несущими синусоидальными сигналами (рис. 5.7). Несущие синусоидальные сигналы генерируются источниками V1, V2, амплитуды которых могут быть промодулированы информационными сигналами, а фазы различаются на  $90^\circ$ , т. е.  $U_1(t)=U_1 \sin \omega t$  и  $U_2(t)=U_2 \sin(\omega t + 90^\circ)=U_2 \cos \omega t$ , и на выходе сумматора на О1 (в линии связи LS) будем иметь линейную комбинацию  $U_s(t)=U_1 \sin \omega t + U_2 \cos \omega t$ .

На приемной стороне сигналы разделяются с помощью двух фазовых двухполупериодных детекторов, выполненных на двух дифференциальных усилителях (О2 и 3), управляемых ключами J1 и J2, которые, в свою очередь, управляются источниками напряжений V1s и V2s, синхронизированными с V1, V2 и также сдвинутыми по фазе на  $90^\circ$ .

На выходе детекторов из композитного сигнала  $U_s(t)$  выделяются  $U_{1r}(t)$ ,  $U_{2r}(t)$  (см. осциллограммы на рис. 5.8), средние значения которых за два полупериода (выпрямление двухполупериодное) соответственно равны  $2(2)^{1/2} (0,71 \times U_1)/\pi = 1,280 \text{ В}$  и  $2(2)^{1/2} (0,71 \times U_2)/\pi = 0,640 \text{ В}$ , что примерно

соответствует показаниям подключенных к выходам ОУ вольтметров. Результаты измерений соответствуют действующим значениям и отличаются от максимальных, которые приведены на рис. 5.8.

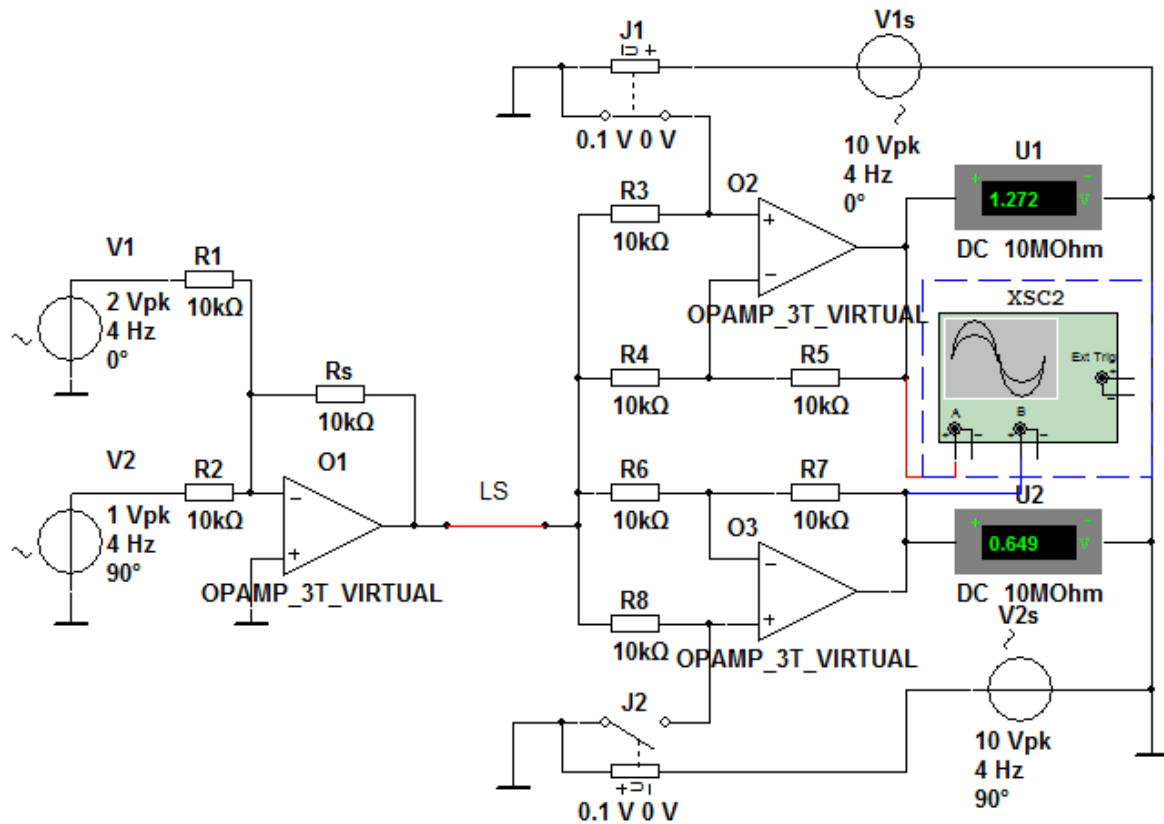


Рисунок 5.7. Схема модели фазового разделение каналов

Очевидно, что в рассматриваемой системе уплотнения при формировании композитного сигнала сигналы каждого канала остаются независимыми, поэтому мощность композитного сигнала  $P_n \approx nP_1$  и, следовательно, энергетический коэффициент уплотнения  $\eta(n) = n = 2$ . Коэффициент уплотнения по частоте  $\lambda(n) = F_n/F_1 = 1$ , поскольку частота несущей не меняется. Это значит, что многоканальные системы с фазовым уплотнением весьма выгодно используют полосу частот линии связи, что и обусловило их преимущественное применение в телекоммуникационных системах с использованием узкополосных телефонных ЛС.

Следует отметить, что одной из важных проблем систем фазового уплотнения является, как и в случае временного уплотнения, обеспечение

синхронной работы задающих генераторов несущей на передающей стороне и источников опорных сигналов на приемной [9].

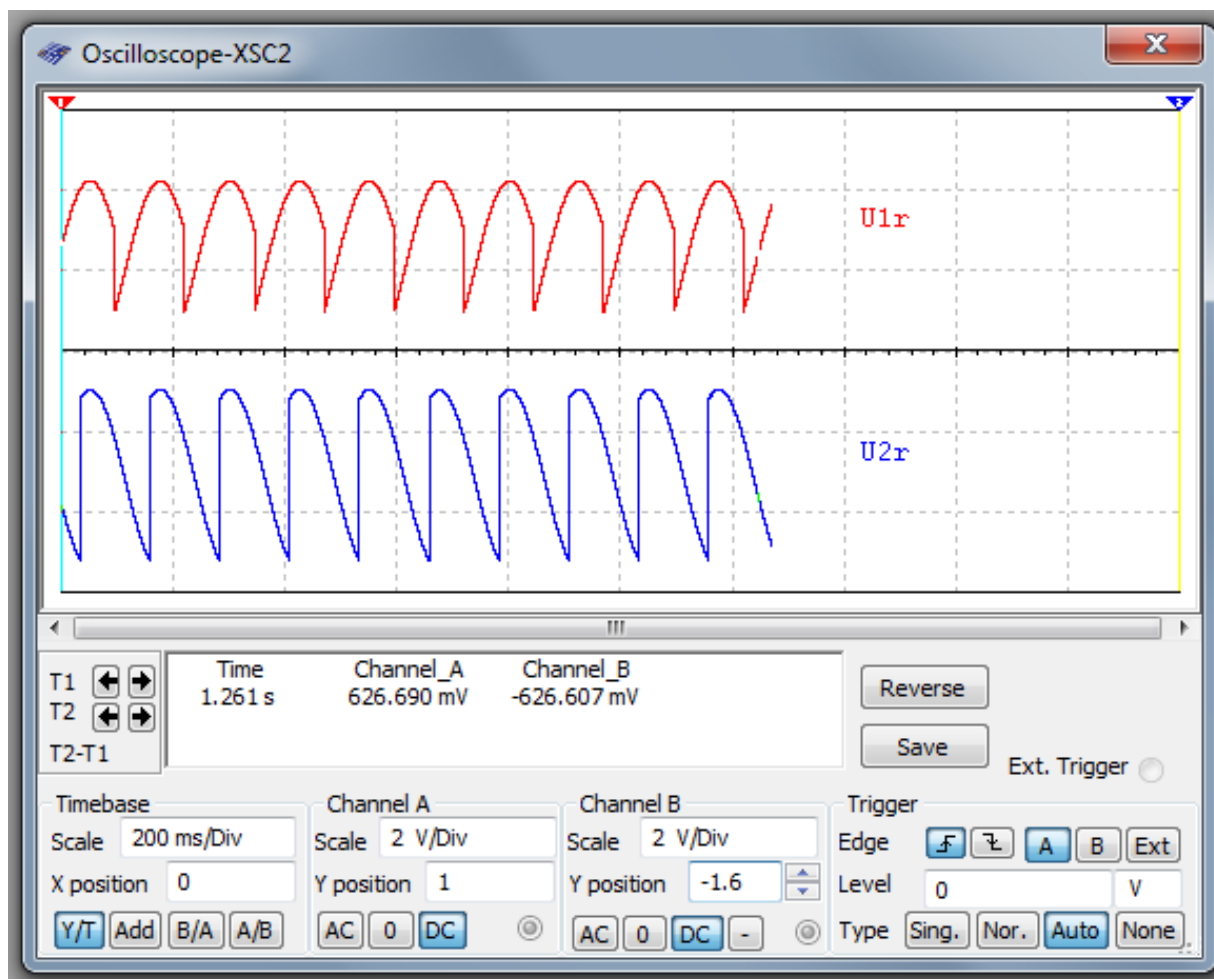


Рисунок 5.8. Осциллограммы на приемной стороне

### 5.5. Кодовое разделение

При кодовом (адресном) разделении адрес нужного канала указывается кодированным сигналом, посылаемым в линию связи [9]. Разделение на приемной стороне осуществляется декодирующим устройством, направляющим сообщения по выбранному каналу. Код адреса может быть как последовательным, так и параллельным. В последнем случае используется отдельная линия связи или индивидуальный канал на каждый разряд кода. Кодовое разделение каналов позволяет производить опрос каналов в произвольном порядке с заданным приоритетом, что важно, например, в

системах автоматического управления, в которых отдельные датчики контроля могут опрашиваться с различной частотой.

Схема, иллюстрирующая принцип кодового разделения каналов, содержит (рис. 5.9) источники входных (канальных) сигналов  $V0 - V3$ , формирователь адреса (источник тактовых сигналов  $V5$ , двухразрядный счетчик на триггерах  $U15$  и  $U16$ , дешифратор на логических элементах НЕ  $U13$ ,  $U14$  и трехвходовых элементах И  $U9 - U12$ ), мультиплексор (элементы И  $U0 - U3$  и ИЛИ  $U4$ ), ЛС и демультиплексор на элементах И  $U5 - U8$ .

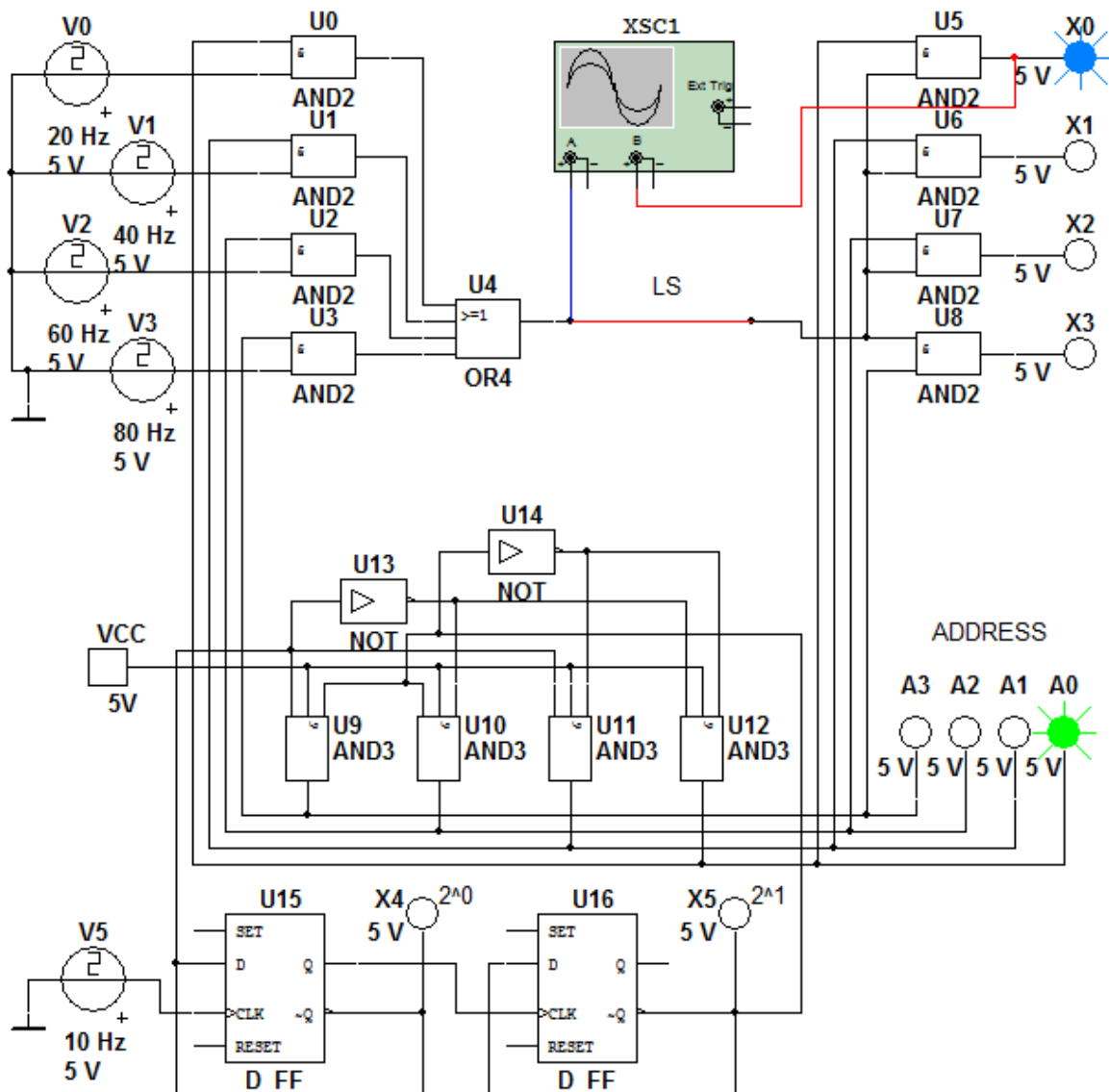


Рисунок 5.9. Модель с кодовым разделением каналов

Работа демультиплексора индицируется логическими пробниками  $X0 - X3$ , отражающими получение сообщения. Результат формирования адреса

представляется на логических пробниках А0 – А4. В практических конструкциях формирователь адреса должен быть как на передающей, так и приемной стороне. Естественно, что их работа должна быть строго синхронизированной.

В исходном состоянии триггеры счетчика находятся в нулевом состоянии (код адреса 00 на выходах  $2^0$ ,  $2^1$ ). При этом сигнал логической 1 на выходе 0 дешифратора деблокирует элементы И U0 на передающей и U5 на приемной стороне и сигналы с частотой 20 Гц (условный код 11 информационного сигнала) через элемент ИЛИ U4 и линию связи LS поступают на выход X0 демультиплексора приемника. Поскольку при длительности развертки осциллографа 50 мс/дел и количестве делений 10 время регистрации составляет 0,5 с, то на его экране мы увидим две посылки кода 11 (осциллограмма U0 на рис. 5.10), сопровождающиеся короткой импульсной помехой, вызванной временной несогласованностью логических элементов. Из осциллограммы Us можно видеть положение посылок источника U0 в композитном сигнале в ЛС.

При поступлении на счетчик первого импульса с генератора V5 на его выходах формируется код адреса 01, в результате чего код 1111 источника V1 передается на выход 1 демультиплексора приемника; при втором импульсе V5 формируется код адреса 10 и код 111111 источника V2 передается на выход 2 демультиплексора, при третьем импульсе формируется код адреса 11 и на выход 3 демультиплексора передается код 11111111 источника UV3.

Метод кодового разделения канатов в сочетании с временным широко используется в телекоммуникационных системах. Рассмотрим процесс адресации абонентов (пользователей) на примере сети Интернет, в которой используется принцип кодового разделения каналов. Поскольку в этой сети, как и в большинстве цифровых систем связи, информация передается пакетами (блоками), то для его доставки по назначению сеть должна до выбора пути знать пункт назначения (адрес абонента). Поэтому каждому узлу связи присваивается уникальный сетевой адрес и уникальное имя.

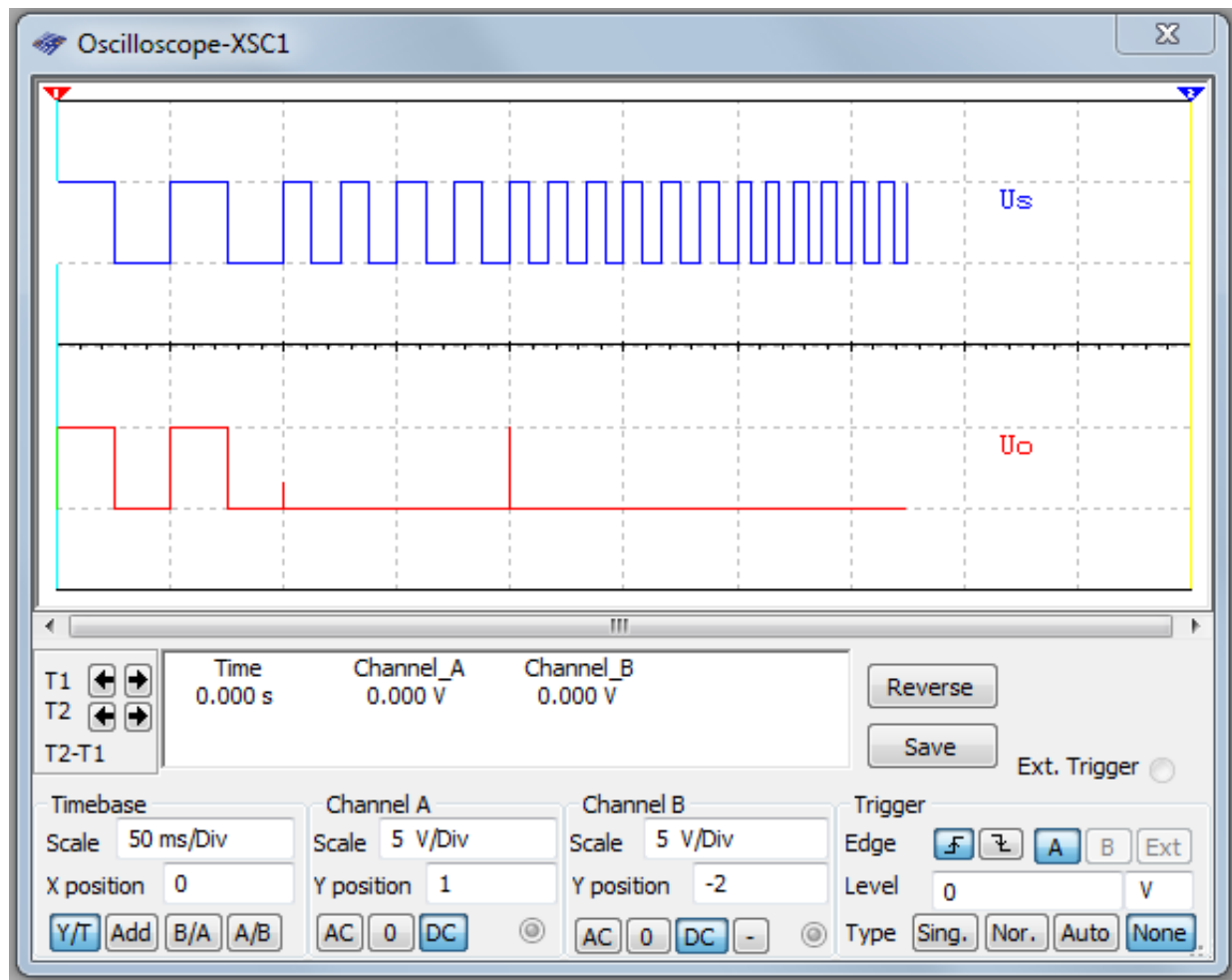


Рисунок 5.10. Осциллограммы модели с кодовым разделением

В сети Интернет имя имеет иерархический характер, основанный на доменной структуре; сетевые адреса также имеют иерархическую структуру, связанную с географией, т. е. с топологией сети и ее привязкой к местности. При этом сервер сети (ее центральный компьютер) должен уметь преобразовывать сетевое имя в сетевое местоположение. Процесс присвоения адреса подключенному к сети компьютеру регламентируется протоколом разрешения адресов ARP (Address Resolution Protocol), который применяется в сетях с набором (стеком) протоколов Интернета TCP/IP (Transmission Control Protocol/Internet Protocol). Возможен случай, когда сетевой адрес находится на сервере (например, в локальных вычислительных системах) и присваивается после запроса обратившегося к нему компьютера в соответствии с так называемым протоколом обратного разрешения адресов RARP (Reverse Address Resolution Protocol).

Чтобы найти сетевой адрес получателя, передающий компьютер использует службу каталога Интернета аналогично поиску номера телефона в справочнике или по запросу в справочной службе телефонной станции. Чтобы обратиться к абоненту другого города, надо набрать его код и затем телефон справочной службы местной телефонной станции. Таким образом, вся база данных (БД) телефонных номеров распределена между операторами по кодам городов или областей. Это распределение локализует звонки в справочные службы и упрощает поддержание и обновление БД. Имена хостов (узлов связи компьютерной сети) в Интернете точно так же сгруппированы в домены. Деление на домены при этом не носит географического характера, а основывается на иерархической организационной структуре уполномоченных, которые управляют именами узлов. Когда хосту требуется адрес какого-либо другого хоста, он запрашивает поиск у локального сервера имен, выполняющийся на компьютере с адресом, который должен быть заранее известен.

Сначала в процессе поиска проверяется, принадлежит ли запрашиваемое имя домену, за который отвечает данный сервер: Если это так, то он дает ответ с уже готовым сетевым адресом. В противном случае он определяет, какой другой сервер имен является ответственным за этот домен имен, и пересылает ему запрос. После получения адреса хост сохраняет его во временной базе данных для использования в дальнейшем. Такой метод декомпозиции (разделения узлов на домены) может быть развит декомпозицией доменов на субдомены, субдоменов на субсубдомены и т. д. Доменами являются имена файлов с расширениями .com (частные компании), .edu (образовательные учреждения), .gov (правительственные структуры), .int (международные организации.), .mil (военные организации), .net (поставщики услуг Интернета), .org (некоммерческие организации). Государства тоже имеют свои домены: .be (Бельгия), .ca (Канада), .ch (Швейцария), .in (Индия), .jp (Япония), .ru (Россия), .uk (Великобритания) и т. п.



Чтобы подключиться к Интернету, пользователь запрашивает у своего поставщика услуг список имен. Тот, в свою очередь, получает имена от своего интернет-регистратора. Каждый домен поддерживает собственный сервер имен домена DNS (Domain Name Server), на котором хранится БД имен и адресов домена Интернет. Сетевой ресурс (файл, доступ к которому должен осуществляться по определенному протоколу) определяется его идентификатором. Наиболее популярным является унифицированный локатор ресурса URL (Uniform Resource Locator), который определяет протокол передачи (например, файловый FTP или гипертекстовый HTTP) и местоположение ресурса.

На первом этапе использовалась классовая адресация типа «сеть.хост», например в виде десятичных чисел 128.32.0.56 (двоичный эквивалент 10000000.00100000.00000000.00111000), где 128 – номер сети; 32.0.56 – номер хоста. При этом деление сетей на классы выглядело следующим образом: большие, класса А (адрес сети составляет 8 бит, адрес хоста сети – 24 бита); средние, класса В (16/16 бит); малые, класса С (24/8 бит). Четвертый класс адресов D зарезервирован для групповой адресации.

Следующим этапом было разделение большой сети, например класса В на подсети. В этом случае приведенный выше пример выглядит следующим образом: 128.32.134 – адрес подсети, а 128.32.134.56 – адрес компьютера в этой подсети, т. е. подсеть определяется третьим байтом, а хост в пределах этой сети – последним байтом. При таком разделении сети хост содержит IP-адрес и маску подсети. Например, компьютер 128.32.134.56 имеет маску подсети 255.255.255.00, т. е. 24 единицы и 8 нулей. При этом подсеть хоста определяется побитовой операцией И (AND) IP-адреса и маски М. Например, если компьютер Е 128.32.152.26 с маской ME 255.255.255.0 хочет послать пакет компьютеру Р с адресом 128.32.134.56 и маской РМ, то он выясняет, что  $P \otimes ME = 128.32.152.26 \otimes 255.255.255.0 = 128.32.152.0$  не соответствует  $E \otimes ME = 128.32.134.56 \otimes 255.255.255.0 = 128.32.134.0$ . Тем самым компьютер Е узнает, что они с компьютером Р находятся в разных подсетях. Поэтому он

посылает пакет маршрутизатору Н подсети, который, выполняя операцию  $H \otimes MP$ , находит, например, что она соответствует  $P \otimes ME$  и, следовательно, компьютер Р находится с ним в одной подсети, после чего маршрутизатор Н просматривает таблицу разрешения адресов ARP или использует протокол ARP для поиска локального Ethernet-адреса (MAC-адреса (Media Access Control) – управление доступом к среде) хоста Р и посылает ему пакет по ЛВС.

Нами рассмотрена четвёртая версия IP-протокола – IPv4, которая при длине адреса 32 бита столкнулась с проблемой малого адресного пространства. Новая версия протокола IPv6, призвана решить данную проблему, за счёт использования длины адреса 128 бит. В настоящее время протокол IPv6 уже используется в нескольких тысячах сетей по всему миру, но пока ещё не получил столь широкого распространения в Интернете, как IPv4. В России используется почти исключительно в тестовом режиме некоторыми операторами связи, а также регистраторами доменов для работы DNS-серверов.

К вопросам кодовой адресации непосредственно примыкают и способы маршрутизации сообщений. В телекоммуникационных сетях между любыми ее двумя узлами имеется, как правило, несколько путей, по которым могут быть переданы сообщения. Основной задачей маршрутизации является выбор определенного пути из указанного множества. Выбор осуществляется с помощью матриц (таблиц) маршрутов, которые хранятся в каждом маршрутизаторе.

При статическом распределении информации может использоваться статическая (фиксированная) маршрутизация. Однако наиболее эффективное использование ресурсов сети достигается при адаптивной маршрутизации, когда план распределения информации изменяется в соответствии с изменяющимися условиями работы сети (перегрузками на отдельных направлениях или участках сети, повреждениями каналов и др.).

Оптимизация маршрутизации может производиться как по общесетевым, так и по локальным критериям. К первым относятся средняя задержка при передаче сообщений (пакетов) по сети и средняя вероятность своевременной

доставки, максимально допустимые значения длины и стоимости пути и др. Локальными критериями могут быть задержка при передаче между группой (парой) абонентов, вероятности потерь на отдельных направлениях связи и др. В общем случае выбор критерия оптимальности в алгоритмах сетевого управления неоднозначен. Чаще всего предпочтение отдается критериям, связанным с коэффициентами использования пропускной способности сети. Это означает, что оптимальными считаются такие решения по маршрутизации или управлению потоками, которые при выполнении требований к характеристикам доставки информации позволяют максимально использовать пропускную способность сети.

В зависимости от того, где производится расчет маршрутных матриц и принимается решение о маршруте для данного узла, различаются: **централизованная** маршрутизация (решение о маршрутах принимается в сетевом маршрутном центре); **распределенная** (децентрализованная) (решение принимается на каждом узле связи, с учетом данных, полученных от других узлов); **изолированная** (решение принимается на каждом узле связи самостоятельно); **смешанная** маршрутизация, представляющая то или иное сочетание централизованной и распределенной.

Еще одним классификационным параметром является момент времени, когда принимается решение об изменении маршрутов в узлах сети. Здесь можно выделить **детерминированные** (пособытийные) методы динамического управления, когда маршрутные матрицы изменяются при зафиксированных неисправностях оборудования или перегрузках на участках сети, при ее запуске, по команде операторов, периодически и т. п., и **статистические** методы, когда изменение маршрутов происходит непрерывно по результатам обслуживания предыдущего вызова или группы вызовов.

Детерминированные методы адаптивной маршрутизации могут быть **разовыми**, когда коррекция плана распределения информации производится после поступления каждой заявки на соединение, или **групповыми**, когда коррекция производится после поступления некоторой группы вызовов.

К разовым относится волновой метод, к групповым – метод рельефов и матричный метод. Метод рельефов осуществляет распределенную маршрутизацию, а матричный – централизованную.

К распределенной маршрутизации относятся стратегии кратчайшего пути с весом линий, равным текущей задержке передачи с учетом ожидания в очереди на обслуживание. В сети ARPANET (предшественнике Интернета) такая стратегия получила название «кратчайшая очередь + смещение».

Примерами изолированной маршрутизации являются стратегия «кратчайшая очередь + нулевое смещение», когда узел направляет сообщение в линию с минимальным ожиданием в очереди; стратегия «горячей картошки» – путем ранжирования различных исходящих линий с назначением наивысшего ранга линии с минимальной очередью, а также волновая стратегия.

Примером смешанной маршрутизации является дельта-маршрутизация, при которой маршрутизатор определяет совокупность альтернативных путей для каждой пары источник – адресат и пересылает ее на узлы связи (хосты).

Для расчета маршрутов при всех типах маршрутизации применяются два класса алгоритмов:

- а) алгоритмы определения кратчайшего пути для заданной пары источник – адресат;
- б) алгоритмы вероятностного выбора возможных маршрутов, минимизирующих (или максимизирующих) выбранный критерий оптимальной маршрутизации с учетом средней загрузки сети.

В алгоритмах кратчайшего пути используются методы нумерации узлов и ветвей и матричные методы. Методы нумерации, как правило, являются итерационными и определяют кратчайший путь до заданного узла с учетом всех остальных узлов сети. Матричные методы позволяют найти кратчайшие пути между всеми узлами сети одновременно, используя операции над матрицами весов ветвей сети. В сетях с коммутацией пакетов широкое распространение получили методы Форда-Фалкерсона и Дейкстры, известные в

Интернете как OSPF (Open Shortest Path First – протокол маршрутизации данных по алгоритму выбора кратчайшего пути).

Итерационный алгоритм Форда-Фалкерсона, относящийся к методам нумерации узлов и ветвей, является одним из алгоритмов, предложенных Флойдом и предназначенных, для распределенной маршрутизации. Алгоритм Дейкстры относится к матричным методам и также является итерационным. Он поддерживает централизованную маршрутизацию, так как для его работы нужна полная информация о структуре сети (в ARPANET он является основой децентрализованной маршрутизации, но при этом каждый узел имеет собственную базу данных с информацией по всей сети).

К классу алгоритмов определения кратчайших путей относится также групповой распределенный метод рельефов, в котором используются результаты обмена служебной информацией по состоянию всех узлов сети.

Алгоритмы вероятностного выбора маршрутов сводятся к реализации методов решения задач оптимального распределения информационных потоков по ветвям сети. Для решения таких задач используются различные итерационные алгоритмы: метод девиации потоков, характеризующийся сравнительно малым объемом вычислений при довольно медленной сходимости; метод экстремальных потоков, отличающийся ускоренной сходимостью; метод Галлагера и др.

При реализации алгоритма Дейкстры необходимые сведения по плану сети извлекаются из базы данных маршрутизаторов, поскольку каждому из них при настройке устанавливается определенный сетевой адрес. При этом работающий в сети маршрутизатор может рассчитать метрику каждого исходящего канала (под метрикой понимается математический термин, обозначающий расстояние между двумя точками (элементами) данного пространства (множества), определяемого по тому или иному правилу).

В простейшем случае метрика равна 1 для работоспособного канала и бесконечно велика для неработоспособного. Более точная метрика учитывает пропускную способность канала и среднюю задержку при прохождении через

его буфер. Метрика выбирается администратором сети, и все маршрутизаторы данной автономной сети (АС) используют одну метрику.

Маршрутизаторы АС могут также выполнять алгоритм Дейкстры с несколькими различными наборами метрик. Например, метрики могут учитывать порознь надежность, пропускную способность и задержку. В результате выполнения алгоритма Дейкстры для каждого из трех наборов метрик все маршрутизаторы сети знают самый надежный путь с наибольшей пропускной способностью и минимальной задержкой до любого другого маршрутизатора АС. В пересылаемом пакете может быть указано, по какому критерию маршрутизаторам следует выбирать путь.

Кроме того, маршрутизатор  $i$  подготавливает сообщение, перечисляющее метрики исходящих каналов, присваивает сообщению  $M$  порядковый номер  $s$  и рассылает его во все свои исходящие каналы. Когда маршрутизатор  $j$  получает подобное сообщение, он проверяет его отправителя. Если  $i = j$ , то маршрутизатор отбрасывает пакет, в другом случае он проверяет наибольший порядковый номер маршрутизирующих сообщений от  $i$ , которые были получены прежде. Если этот номер меньше  $s$ , то это означает, что  $M$  – новое сообщение от  $i$ . В таком случае маршрутизатор  $j$  обновляет сообщение наибольшего порядкового номера, сохраняет данное сообщение для своих нужд и посылает копию этого сообщения во все исходящие каналы, за исключением того, от которого оно пришло. Такой метод распространения служебных сообщений по всем другим маршрутизаторам называется **лавинной маршрутизацией**.

После получения сообщения маршрутизатор может построить (или обновить) схему сети и выполнить по ней алгоритм Дейкстры. Маршрутизатор может послать новое сообщение, если существенно изменилась метрика канала.

Кроме протокола OSPF, в сети Интернет для маршрутизации между АС используется протокол граничных маршрутизаторов BGP (Border Gateway Protocol), который, в отличие от OSPF, является **распределенным**. Поэтому маршрутизаторы выполняют этот протокол при различной степени

информированности о сети. Другое отличие заключается в том, что в BGP используется алгоритм **предпочтительного пути**, а не **алгоритм метрик**.

АС может быть соединена с другими АС несколькими граничным маршрутизаторами. В таком случае администратор сети выбирает внутри АС один маршрутизатор для реализации BGP, так называемый **спикер BGP**. Как и в случае OSPF, BGP может иметь несколько разных метрик. Кроме того, каждая АС может поддерживать списки АС, через которые она не хочет передавать пакеты.

Достоинством протокола BGP является его экономность, поскольку вместо лавинного заполнения сети каждая АС посылает лишь небольшие сообщения своим соседям. Кроме того, характерный для распределенных алгоритмов риск заикливания из-за несогласованности метрик в BGP может быть уменьшен за счет объявления путей в дополнение к метрикам.

Некоторое отношение к маршрутизации имеют два интересных «мобильных» протокола Интернет: DHCP (Dynamic Host Configuration Protocol – протокол динамического конфигурирования хостов) и MIP (mobile IP – мобильный интернет-протокол), дополняющие друг друга и позволяющие временно подключиться и работать в сети Интернет.

Принцип работы протокола DHCP, реализующего алгоритм plug and play (включил и работай), заключается в том, что в случае временного подключения к сети нового, например портативного, компьютера, для получения сетевого адреса с него посылается специальное широковещательное DHCP-сообщение типа «Маршрутизатор, дай мне сетевой адрес!». На эту просьбу маршрутизатор реагирует тем, что из фонда свободных сетевых адресов выдает один из них «новичку» на определенное время (например, 20 минут). После этого портативный компьютер может использовать стандартные приложения Интернета.

Когда остаток времени жизни, назначенного маршрутизатором, приближается к 0, портативный компьютер запрашивает у маршрутизатора еще 20 минут, которые, как правило, выдаются (речь идет о сети Ethernet

Калифорнийского университета в Беркли). Если «новичок» выключает свой компьютер, то после того, как его время жизни в сети истечет, Сетевой адрес возвращается опять в фонд свободных адресов. Такой же механизм может быть использован и в случае беспроводной сети Ethernet.

Протокол DHCP, совмещенный с протоколом MIP, позволяет автоматически перенаправить электронную почту «новичка» с его домашнего компьютера на портативный, подключенный, как мы говорили, в совершенно другом месте. При этом предполагается, что «домашний» маршрутизатор «новичка» знает о его существовании. В таком случае портативный компьютер автоматически регистрируется с помощью агента в сети, к которой он подключается. Этот агент выделяет «новичку» временный сетевой адрес, вступает в контакт с агентом на «домашнем» маршрутизаторе и информирует его о новом местоположении хозяина. Агент на домашнем маршрутизаторе перехватывает все пакеты, предназначенные портативному компьютеру «новичка», и пересылает их удаленному агенту, который доставляет их по временному адресу. Когда истекает время жизни временного адреса, удаленный агент просит домашнего агента прекратить пересылку пакетов до тех пор, пока хозяин не вернется или не объявится в новом месте.

## 5.6. Разделение по уровню

В системах с разделением по уровню параметром разделения служит амплитуда сигналов, а полезная информация может содержаться, например, в их длительности [9].

В качестве примера рассмотрим двухканальную систему, выполненную в среде NI Multisim (рис. 5.11). Данная схема содержит на передающей стороне два источника однополярных импульсных сигналов амплитудой  $U_1 = 5 \text{ В}$ ,  $U_2 = 2,5 \text{ В}$  и коэффициентом заполнения 70 и 20 % (осциллограмма на рис. 5.12). Оба сигнала, линейно складываясь в сумматоре S1, образуют композитный сигнал  $U_s = U_1 + U_2$ , поступающий в ЛС (рис. 5.12).



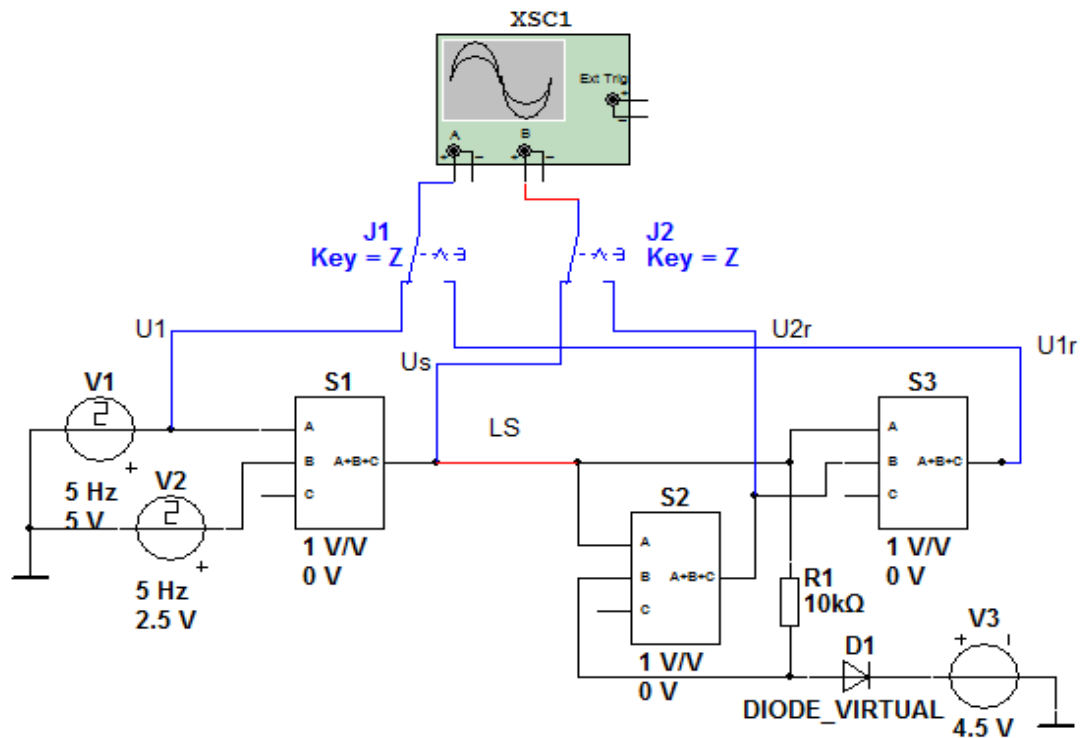


Рисунок 5.11. Модель с разделением каналов по уровню

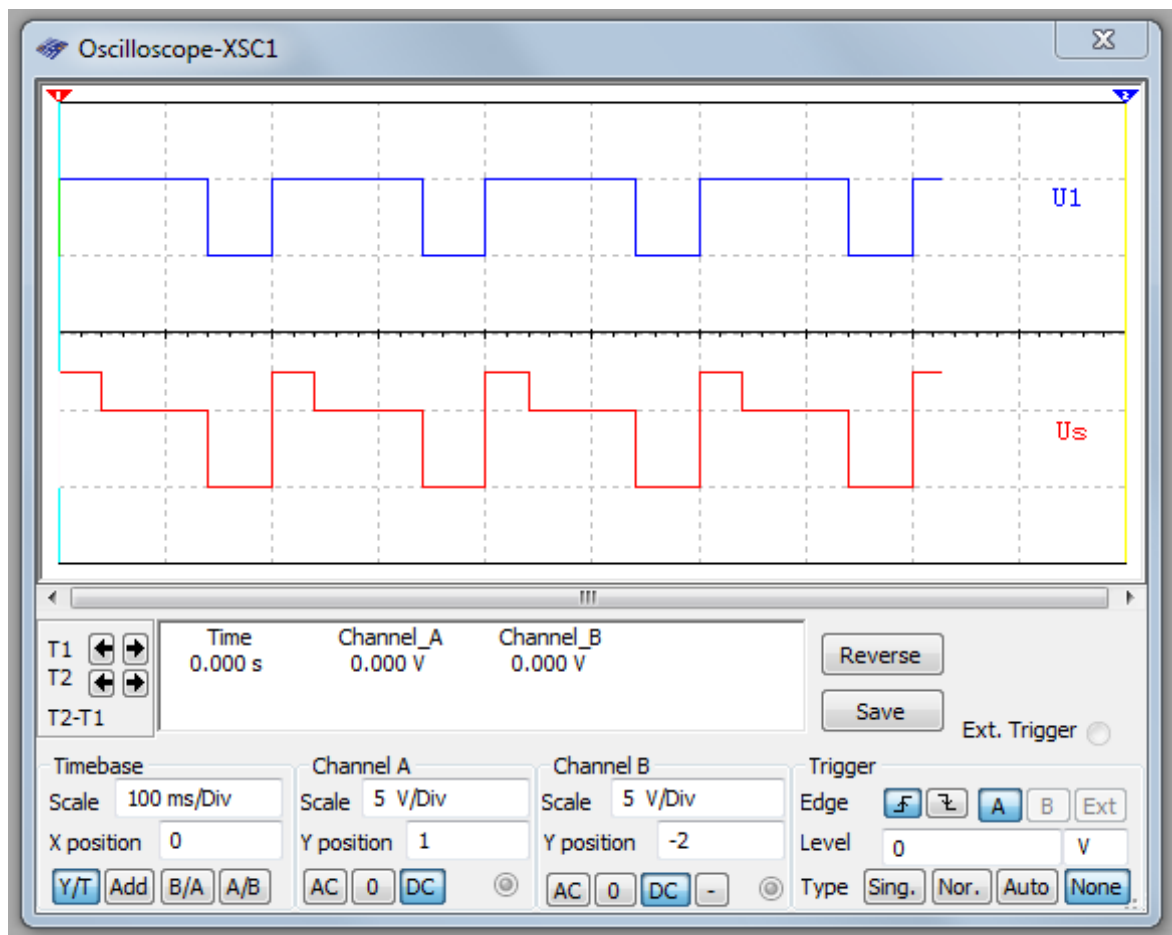


Рисунок 5.12. Осциллограммы модели на передающей стороне

На приемной стороне с помощью диодного ограничителя (резистор  $R1$ , диод  $D1$  и источник постоянного напряжения  $V3$ ) формируется напряжение  $U_b = U_0 + U_d \approx 5 \text{ В} = U_1$  ( $U_d \approx 0,5 \text{ В}$  – падение напряжения на открытом диоде). Напряжение  $U_b$  подается на вход  $B$  сумматора  $S2$  и за счет выбора коэффициента передачи по входу  $B$  ( $K_B = -1$ ) вычитается из  $U_s$ , в результате чего на его выходе формируется напряжение  $U_{2r} = U_s - U_b \approx U_s - U_1 = U_2$  (см. осциллограмму на рис. 5.13, которая получается при переводе сдвоенного ключа  $J1 - 2$ ).

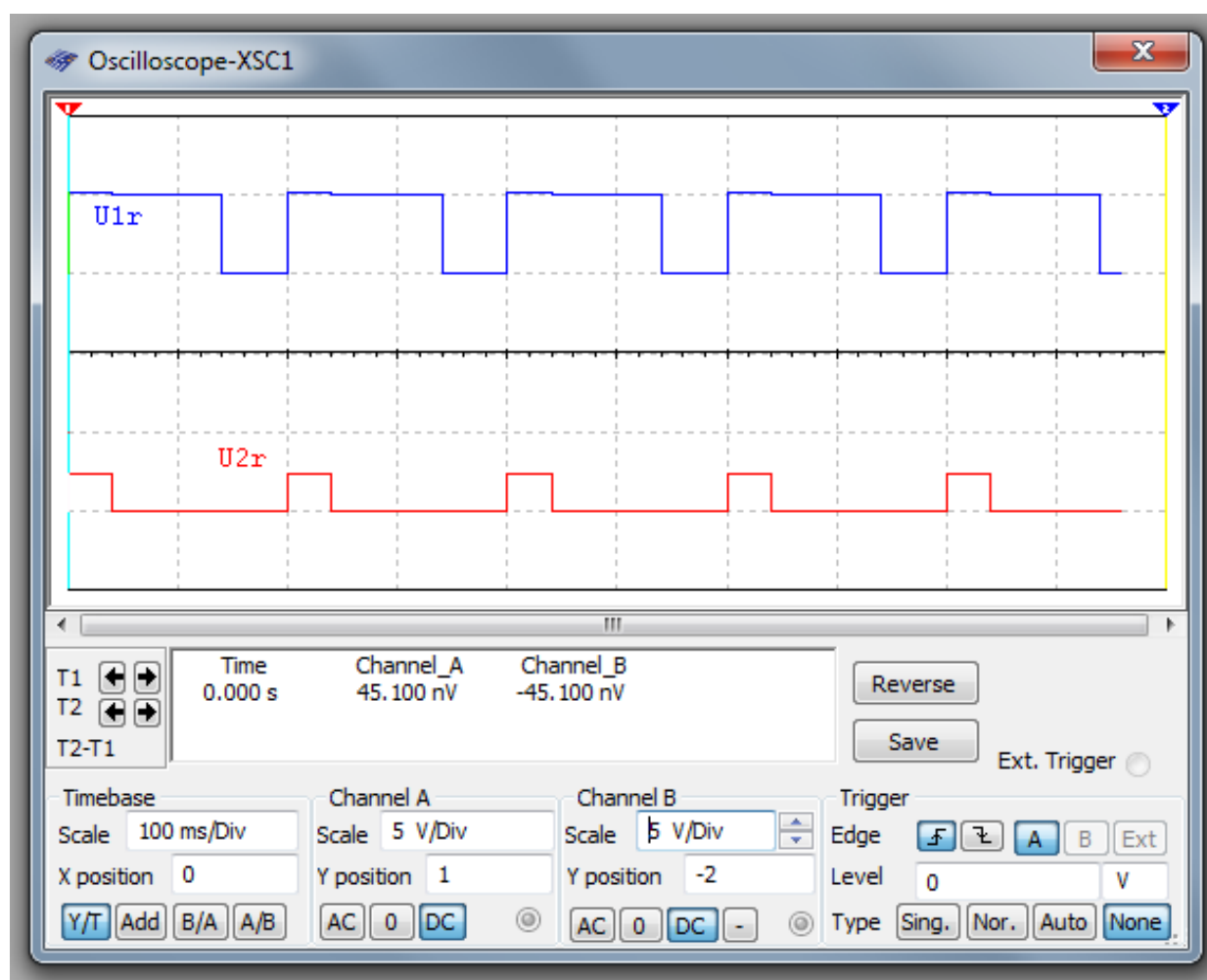


Рисунок 5.13. Осциллограммы модели с разделением каналов по уровню на приемной стороне

Для выделения  $U_{1r}$  необходимо, очевидно, выполнить операцию  $U_s - U_{2r} = U_{1r}$ , которая осуществляется сумматором  $S3$  при  $K_B = -1$ . Из осциллограммы на рис. 5.13, в видно, что вершина  $U_{1r}$  несколько искажена

вследствие ошибок формирования сигнала  $U_b$  на выходе диодного ограничителя и последующей операции вычитания в сумматоре S2.

Увеличение числа каналов рассматриваемой системы достигается увеличением числа диодных ограничителей, схем вычитания и ограничивается их точностью и стабильностью.

## **5.7. Лабораторная работа 5. Исследование особенностей работы систем уплотнения и разделения каналов**

### **5.7.1. Цель и содержание**

Целью лабораторной работы является приобретение студентами практических навыков исследования принципов функционирования систем уплотнения и разделения каналов.

#### **Содержание**

1. Исследование схемы дифференциального разделения каналов.
2. Исследование особенностей работы схемы частотного разделения каналов.
3. Исследование модели временного разделения каналов.
4. Исследование особенностей работы схемы с фазовым разделением каналов.
5. Исследование схемы с кодовым разделением каналов.
6. Исследование особенностей работы схемы с разделением каналов по уровню.

### **5.7.2. Аппаратура и материалы**

Для изучения материалов этой лабораторной работы необходимо:

1. Компьютер под управлением операционной системы Windows XP/Vista/7 с параметрами по умолчанию.
2. Установленная инструментальная среда визуального моделирования динамических систем VisSim/Comm 7.0.

3. Установленная среда схемотехнического проектирования NI Multisim 10.

4. Текстовый процессор (Microsoft Word 2003/2007/2010 или его аналог), предназначенный для создания и редактирования отчета по лабораторной работе.

### **5.7.3. Порядок выполнения работы**

1. Изучение теоретического материала.
2. Выполнение задания, предлагаемого в лабораторной работе.

#### **Задание**

1. Запустить на выполнение файл 5-1-1.ms10. Провести анализ схемы дифференциального разделения каналов. Определить зависимость характеристик сигнала во вторичной обмотке T2 от изменения параметров источника сигнала V2. Сделать выводы.

2. Запустить на выполнение файл 5-2-1.VSM. Провести моделирование схемы частотного разделения каналов. Организовать работу программы при частотах несущих сигналов 15 и 23 Гц.

3. Запустить на выполнение файл 5-3-1.ms10. Провести моделирование схемы временного разделения каналов. Описать зависимость формы осциллограмм на приеме и передаче от параметров модели.

4. Запустить на выполнение файл 5-4-1.ms10. Провести моделирование схемы с фазовым разделением каналов. Провести модернизацию модели, которая приведет к взаимной смене параметров каналов.

5. Запустить на выполнение файл 5-5-1.ms10. Провести моделирование схемы с кодовым разделением каналов. Снять осциллограммы U1, U2, U3 на приемной стороне.

6. Запустить на выполнение файл 5-6-1.ms10. Провести моделирование схемы с разделением каналов по уровню. Провести доработку модели до 3-канальной системы. Снять осциллограммы в точках U1, U2, U3, Us, U1r, U2r, U3r. Сделать выводы.

#### 5.7.4. Вопросы для защиты работы

1. Назовите особенности пространственного и дифференциального разделения каналов.
2. Прокомментируйте принципы работы дифференциального трансформатора.
3. Назовите преимущества и недостатки систем с частотным разделением каналов.
4. Прокомментируйте понятие поднесущей.
5. Перечислите и прокомментируйте преимущества и недостатки систем с временным разделением каналов.
6. Назовите пути устранения искажения импульсных сигналов при ограниченности полосы пропускания реального канала.
7. Перечислите и прокомментируйте преимущества и недостатки систем с фазовым разделением каналов.
8. Назовите функциональное назначение основных элементов модели канала с фазовым разделением.
9. Перечислите и прокомментируйте преимущества и недостатки систем с кодовым разделением каналов.
10. Назовите особенности применения технологии кодового разделения каналов в сети Интернет.
11. Чем отличается четвертая версия IP-протокола – IPv4 от IPv6?
12. Перечислите и прокомментируйте преимущества и недостатки систем с разделением каналов по уровню.
13. Как изменяются параметры работы схемы с разделением каналов по уровню при переводе сдвоенного ключа Z?

## ГЛАВА 6. УСТРОЙСТВА ФИЛЬТРАЦИИ

Фильтрами называют устройства, обеспечивающие выделение из входного сигнала тех или иных временных (time domain) или частотных (frequency domain) составляющих [9].

В течение длительного времени, начиная с первых шагов развития, в электросвязи и радиотехнике широкое распространение получили **аналоговые фильтры** на основе индуктивных и емкостных элементов, а также электромеханических и пьезокерамических резонаторов. Такие элементы имеют большие габариты, достаточно сложную технологию изготовления и, в связи достижениями микроэлектроники, успешно заменяются **цифровыми фильтрами**, построенными на элементах временной задержки и цифровых блоках. Такие фильтры являются основой цифровых сигнальных процессоров, обеспечивающих резкое снижение стоимости и повышение качества фильтрации.

Цифровой фильтр – это фактически дискретно-временная система, преобразующая цифровой входной сигнал в модифицированный цифровой выходной. Математически он описывается системой дифференциальных уравнений, которая обычно представляется в конечно-разностном виде:

$$y(k) = \sum_{i=1}^M a_i x(k-i) - \sum_{j=1}^N b_j y(k-j). \quad (6.1)$$

Это уравнение представляет отношение между  $k$ -м отсчетом выходного сигнала,  $N$  предыдущими и  $M$  последующими значениями отсчетов входного сигнала.

Фильтры могут задаваться также передаточными характеристиками. Например, если уравнение дискретного фильтра класса time domain задано в виде  $y(k) = x(k) - 0,2y(k-1) - 0,8y(k-2)$ , то он может быть создан как в виде устройства с отдельными блоками, так и в виде блока с передаточной функцией  $y(z) = z^2 / (z^2 + 0,2z) + 0,8$ .

Согласно дуальности (двойственности) понятий времени и частоты временные и частотные фильтры могут быть преобразованы друг в друга.

Наиболее распространены частотные фильтры, характеризующиеся АЧХ, ФЧХ характеристиками и реализованными в них математическими функциями. По виду АЧХ фильтры делятся на фильтры низкой частоты (ФНЧ) – Low Pass; фильтры высокой частоты (ФВЧ) – High Pass; полосовые фильтры (ПФ) – Band Pass; полосовые заградительные фильтры (ЗФ) – Stop Pass. По используемым математическим функциям фильтры делятся на фильтры Баттерворта (Butterworth), Бесселя (Bessel), Чебышева (Chebyshev) и др.

**Фильтр Бесселя** обеспечивает одинаковую временную задержку сигналов всех частот, максимально плоскую АЧХ (без колебаний в полосе пропускания и задержки) и потому часто используется для частотной селекции. Фильтр не искажает сигнал, спектр которого лежит в пределах полосы пропускания, однако переходная характеристика этого фильтра имеет небольшое перерегулирование (выброс).

**Фильтр Баттерворта** имеет максимально плоскую АЧХ, что делает его предпочтительным для частотной селекции; все производные фильтра от первой до  $(2n - 1)$ -й равны нулю. Нормированная АЧХ такого фильтра  $n$ -го порядка задается аппроксимацией:  $W(jm) = (m^{2n-1})^{-1/2}$ , где  $m = \omega/\omega_c$  – нормированная частота (относительно частоты среза  $\omega_c$ ).

**Фильтры Чебышева** (прямой и инверсный) обеспечивают максимальное подавление сигнала в области частот выше частоты среза. Они имеют колебательную АЧХ, но эти колебания одинаковы по уровню. Наилучшей для этих фильтров считается, следующая аппроксимация:

$W(j\omega) = [1 + c^2(T_n(\omega))^2]^{-1/2}$ , где  $c$  – коэффициент, определяющий неравномерность АЧХ в полосе пропускания;  $T_n(\omega)$  – полином Чебышева первого рода  $n$ -го порядка. В полосе пропускания подкоренное выражение колеблется между 1 и  $1/(1 + \omega^2)$ .

В среде моделирования VisSim возможно создание цифровых фильтров, которые делятся на два больших класса:

1. IIR (Infinity Impulse Responce) – рекурсивные фильтры с бесконечной импульсной характеристикой.
2. FIR (Finite Impulse Responce) – нерекурсивные фильтры с конечной импульсной характеристикой, для которых в формуле (6.1) все  $b_j = 0$ .

### 6.1. Создание ИИР-фильтров

Проектирование фильтров в VisSim/Comm по существу сводится к редактированию имеющихся в этом приложении библиотечных моделей [9]. Рассмотрим модель ИИР-фильтра Чебышева, созданную на основе демофайла Filter\_Viewer.vsm, схема включения которой приведена на рис. 6.1. В качестве испытательного в схеме используется синусоидальный сигнал вырабатываемый блоком Gen (осциллограмма 1). Сигнал на выходе ИИР-фильтра Чебышева (блок Filter) представлен осциллограммой 2.

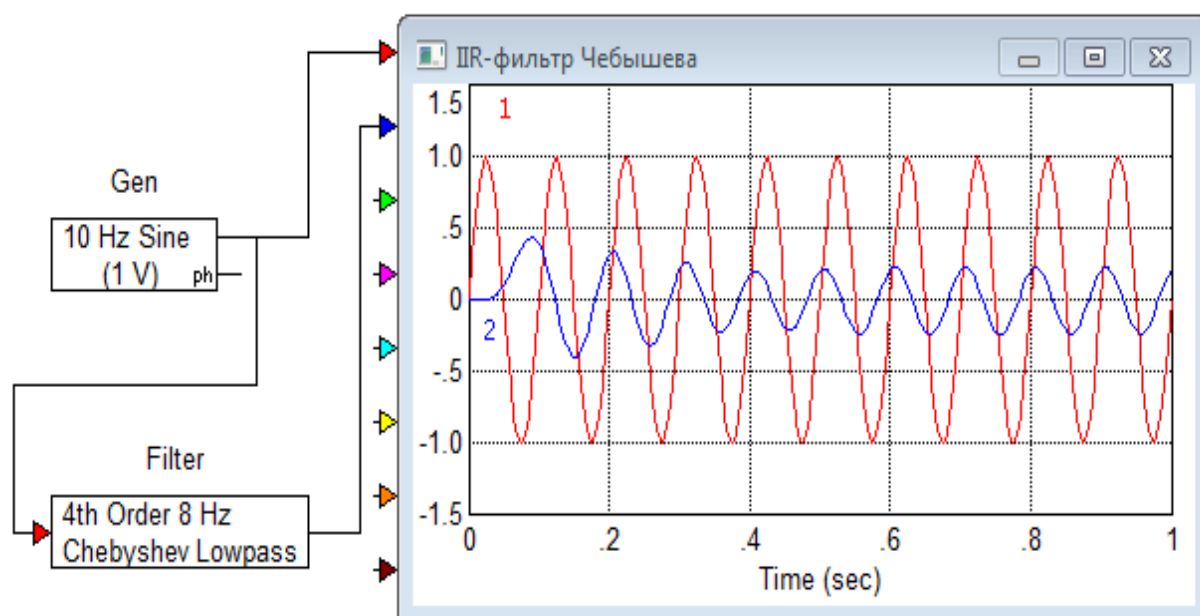


Рисунок 6.1. Схема включения модели ИИР-фильтра Чебышева



Редактирование и просмотр характеристик фильтра осуществляются в его диалоговом окне (рис. 6.2), вызываемом, как обычно, правым щелчком мыши по значку блока.

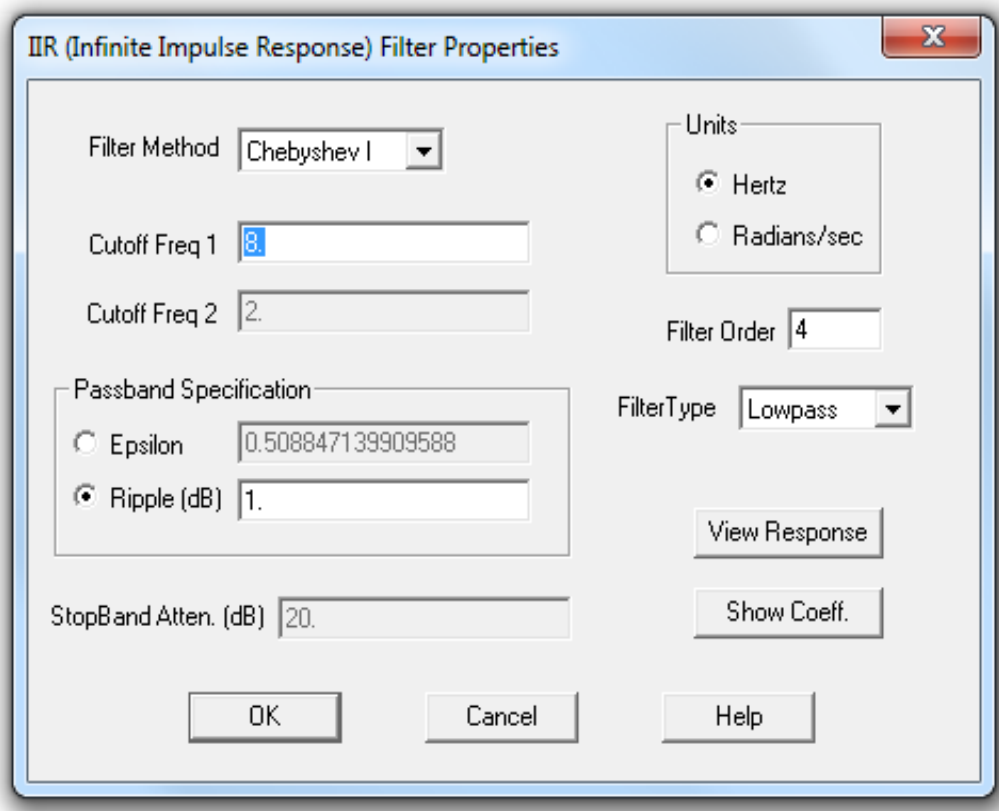


Рисунок 6.2. Диалоговое окно IIR-фильтра Чебышева

В этом окне можно выбрать следующие параметры фильтра:

- Filter Method – в этом разделе можно выбрать тип фильтра;
- Cutoff Freq – частота отсечки;
- Passband Specification – спецификация полосы пропускания;
- StopBand Atten. – затухание в области непропускания; задается только для фильтров Elliptic и Chebyshev II (с инверсией);
- Filter Order – порядок фильтра;
- Filter Type – тип фильтра по виду АЧХ: Low Pass (низкочастотный); High Pass (высокочастотный); Band Pass (полосовой); Band Stop (полосовой заградительный);

– Show Coef. (Показать коэффициенты) – кнопка вызова окна с коэффициентами передаточной характеристики заданного фильтра (рис. 6.3).

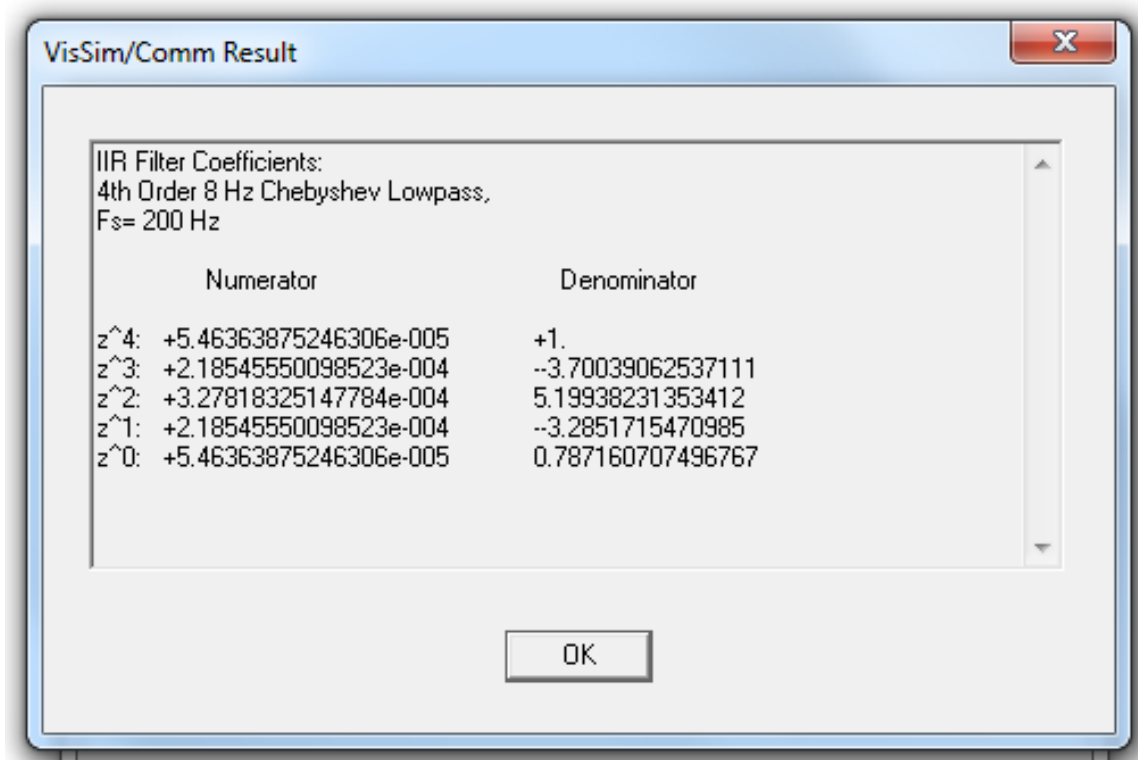


Рисунок 6.3. Таблица коэффициентов передаточной функции IIR-фильтра Чебышева

С помощью кнопки View Response на рис. 6.2 вызывается окно (рис. 6.4) для получения следующих характеристик фильтра (Response Type): Impulse – импульсной; Gain – амплитудно-частотной; Gain (dB) – логарифмической; Phaspr. – фазочастотной; Group Delay – групповой задержки; Cum Area – кумулятивной характеристики ( $\text{Gain}^2$ ).

Второстепенные параметры и настройки: Print – вывод графика на принтер; Copy to Clip. – поместить в буфер обмена; Set Bouns – вызов окна установки пределов по осям графика-характеристики; Imp. Response Size – количество расчетных точек для импульсной характеристики; FFT Size – количество выборок прямого преобразования Фурье; Markers – использование маркеров для обозначения расчетных точек на графике; Title – вывод названия фильтра; Monochrome – монохромный режим; Grid Lines – вывод сетки.

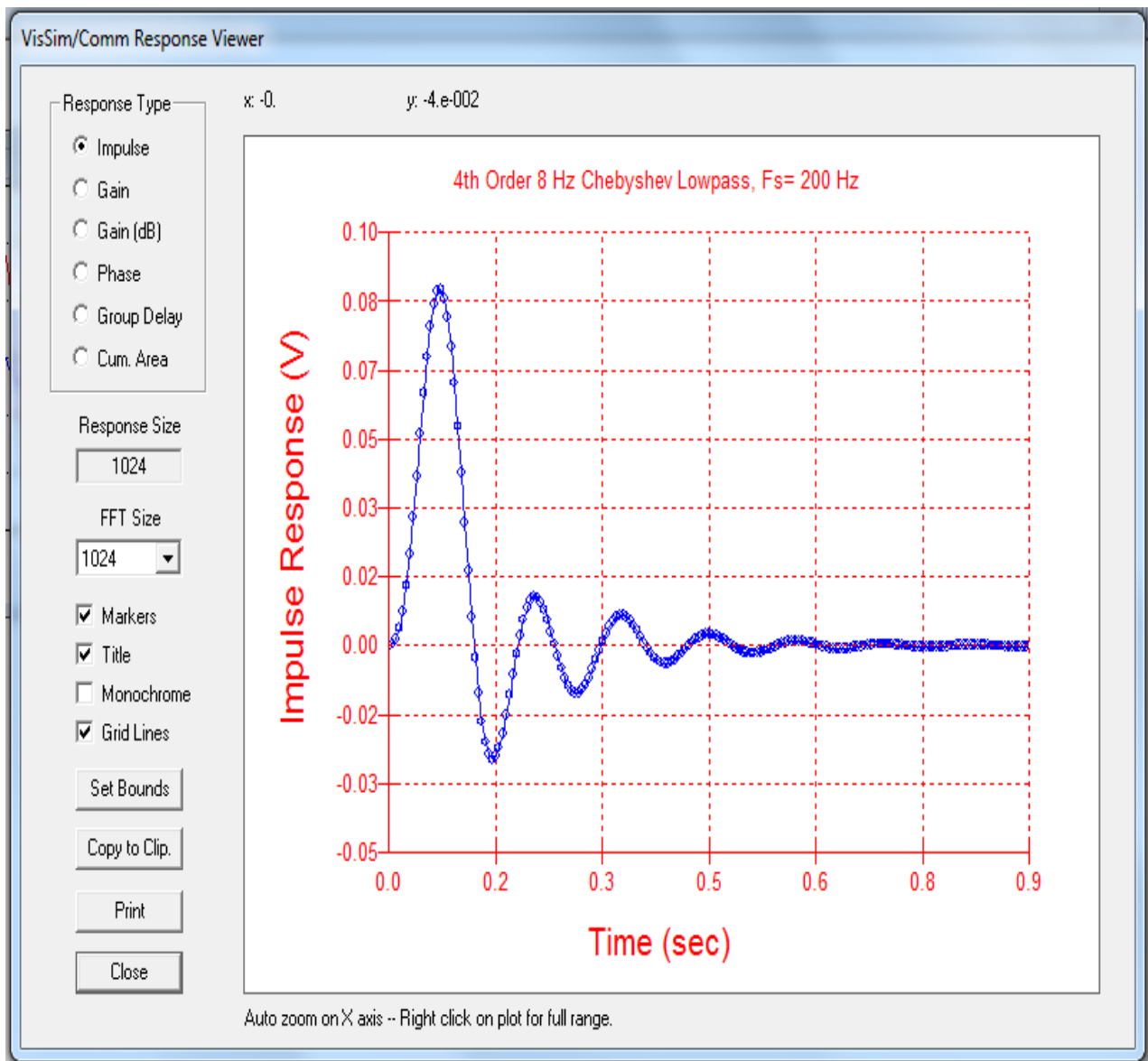


Рисунок 6.4. Окно характеристик фильтра

## 6.2. Создание FIR-фильтров

Рассмотрим процесс создания нерекурсивного фильтра с конечной импульсной характеристикой [9]. Модель FIR-фильтра создана на основе демофайла FIR\_Filter.vsm, схема включения которой приведена на рис. 6.5. В данной модели на вход блока Filter подается короткий испытательный импульс с источника (блок Gen), используемый также для запуска блока прямого преобразования Фурье (Complex FFT), позволяющего получить осциллограммы АЧХ (2) и ФЧХ (3). При этом блок  $\text{rad} \rightarrow \text{deg}$  используется для преобразования радиан в градусы. Поскольку испытательный сигнал

эквивалентен дельта-функции, то на выходе фильтра сигнал представляет собой импульсную характеристику (осциллограмма 1).

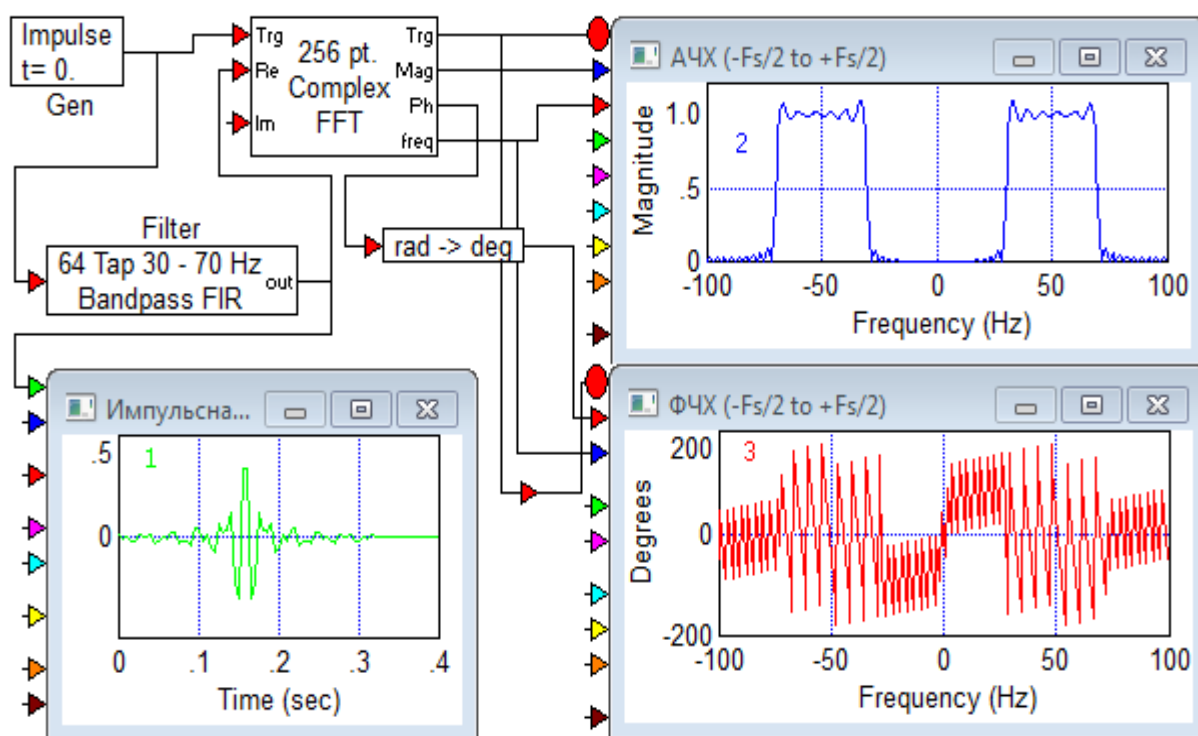


Рисунок 6.5. Схема для испытания FIR-фильтра

В диалоговом окне рассматриваемого фильтра (рис. 6.6) можно отредактировать следующие параметры:

- Filter Type – выбор типа фильтра по виду АЧХ: низкочастотный (Lowpass), высокочастотный (Highpass), полосовой (Bandpass), заградительный (Bandstop), косинусно нарастающий (Raised Cosine), усеченный косинусно нарастающий (Root Raised Cosine), Гильберта (Hilbert) или Гаусса (Gaussian);
- Rolloff Factor – параметр для фильтров Raised Cosine и Root Raised Cosine (0 – 1); нулевое значение выбирается для случая усеченной  $\sin(x)/x$  импульсной характеристики;
- Number of Taps – число выборок (1 – 32767); для ВЧ (Highpass) и заградительных (Bandstop) фильтров это число должно быть нечетным;
- Cutoff Freq 1 – частота среза (отсечки) для фильтров lowpass, raised cosine; root raised cosine, hilbert, gaussian или highpass; для фильтров bandpass и bandstop рекомендуется выбирать более низкое значение этой частоты; частота

отсечки для фильтров raised cosine, root raised cosine и gaussian определяется по уровню 3 дБ, для остальных – по уровню 6 дБ;

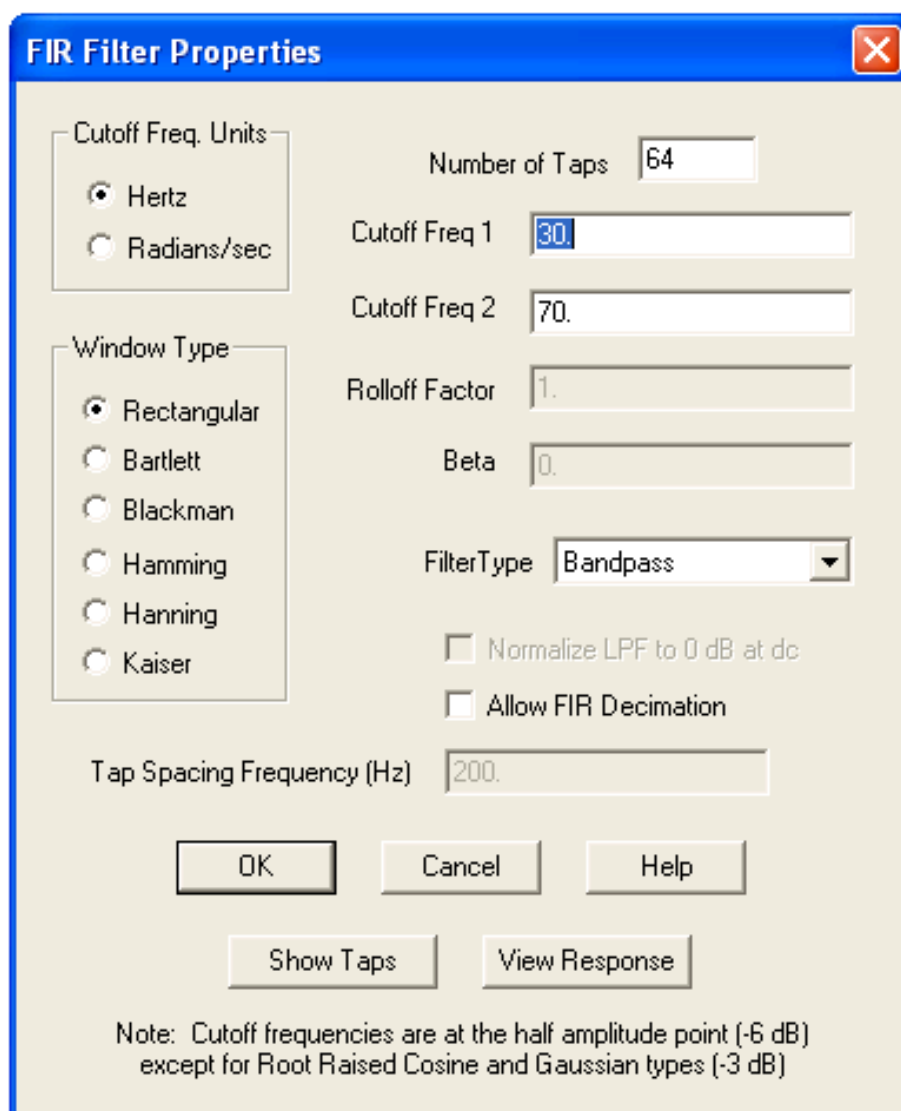


Рисунок 6.6 – Диалоговое окно FIR-фильтра

- Cutoff Freq 2 – частота отсечки (по уровню 6 дБ) для фильтров bandpass и bandstop;
- Window Type – опциональный выбор оконного режима; в случае выбора режима Kaiser указывается значение параметра Beta;
- Units – выбор единиц измерения частоты: Гц (Hertz) или рад/с (Radians/Sec);
- Normalize LPF to 0 dB at dc – нормализация импульсной характеристики относительно ее значения на постоянном токе (0 дБ);

используется в случае небольших значений частоты и числа выборок для фильтров lowpass, raised cosine, root raised cosine, hilbert и gaussian;

- Tap Spacing Frequency – частота следования выборок, Гц; установка параметра доступна после активизации опции Allow FIR Decimation;

- View Response – кнопка для просмотра отклика (реакции) фильтра; вызывается окно, аналогичное показанному на рис. 6.4;

- Show Taps – кнопка для вывода текущих значений выборок, которые могут быть сохранены в файле после нажатия кнопки Save To File; окно по внешнему виду аналогично показанному на рис. 6.3.

### 6.3. Адаптивные фильтры

Адаптивные фильтры наибольшее распространение нашли в устройствах под названием **эквалайзер**, которые, в частности, широко применяются в технике модемной связи [9]. Выполняемые эквалайзером функции заключаются в компенсации частотных и нелинейных искажений.

Частотные искажения вызываются распределенными реактивными и активными сопротивлениями линии связи и проявляются в «наезде» импульсов друг на друга.

Нелинейные искажения вызываются особенностями каналообразующей аппаратуры цифровых телефонных каналов связи, которые изначально разрабатывались и использовались в первую очередь для передачи речевых сообщений. В таких каналах перед оцифровкой аналогового сигнала на передающей стороне используется его сжатие (компрессия) по логарифмическому закону и растяжение (декомпрессия) после декодирования на приемной (компандирование сигнала).

Использование логарифмического компандирования примерно соответствует чувствительности человеческого уха и позволяет при относительно небольшом количестве уровней квантования закодировать звук человеческого голоса с сохранением его разборчивости. Если модем будет

использоваться для передачи по такому каналу равномерно распределенных уровней, то символы с большой амплитудой будут вызывать перегрузки приемной аппаратуры и, следовательно, нелинейные искажения. Поскольку наравне с цифровыми широко распространены и аналоговые телефонные каналы, то модемы, как универсальные устройства, должны распознавать тип канала и, соответственно, менять уровень и тип кодирования (линейный или логарифмический).

Модель адаптивного фильтра создана на основе демофайла Equalizer.vsm и содержит (рис. 6.7) источник испытательного сигнала в виде 16-позиционного квадратурного амплитудного модулятора, выполненного в виде субблока 16 QAM Source, состоящего из генератора 16-уровневых случайных символов и собственно модулятора 16 QAM Mod (рис.6.8). Тактовый сигнал генератора передается в другие части схемы через переменную *clk*.

Комплексный модулированный сигнал подается на вход низкочастотного фильтра Чебышева, который содержит (см. рис. 6.9) два «простых» фильтра Чебышева, используемых для фильтрации действительной и мнимой части модулированного сигнала, полученного на выходе преобразователя Cplx to Re/Im и перемноженного с сигналом регулятора Gain Factor для изменения уровня. Выходные сигналы фильтров преобразуются в комплексный сигнал блоком Re/Im to Cplx и подаются на вход эквалайзера.

Составляющие комплексного сигнала отображаются сигнальным созвездием в виде крестообразных точек концов вектора каждого из многопозиционных сигналов на комплексной плоскости (осциллограмма 2) и так называемой Eye-диаграммой (осциллограмма 1), напоминающей анатомированные глаза, где каждая складка «мышцы» отображает составляющие комплексного сигнала на плоскости  $(y, t)$ .

Модель работает в режиме рестарта, и отображаемые данные получаются за три пуска, которые задаются субблоком управления перезапуском и остановом STOP BLOCK. Из полученных осциллограмм видна явная «неупорядоченность» многопозиционного модулированного сигнала.

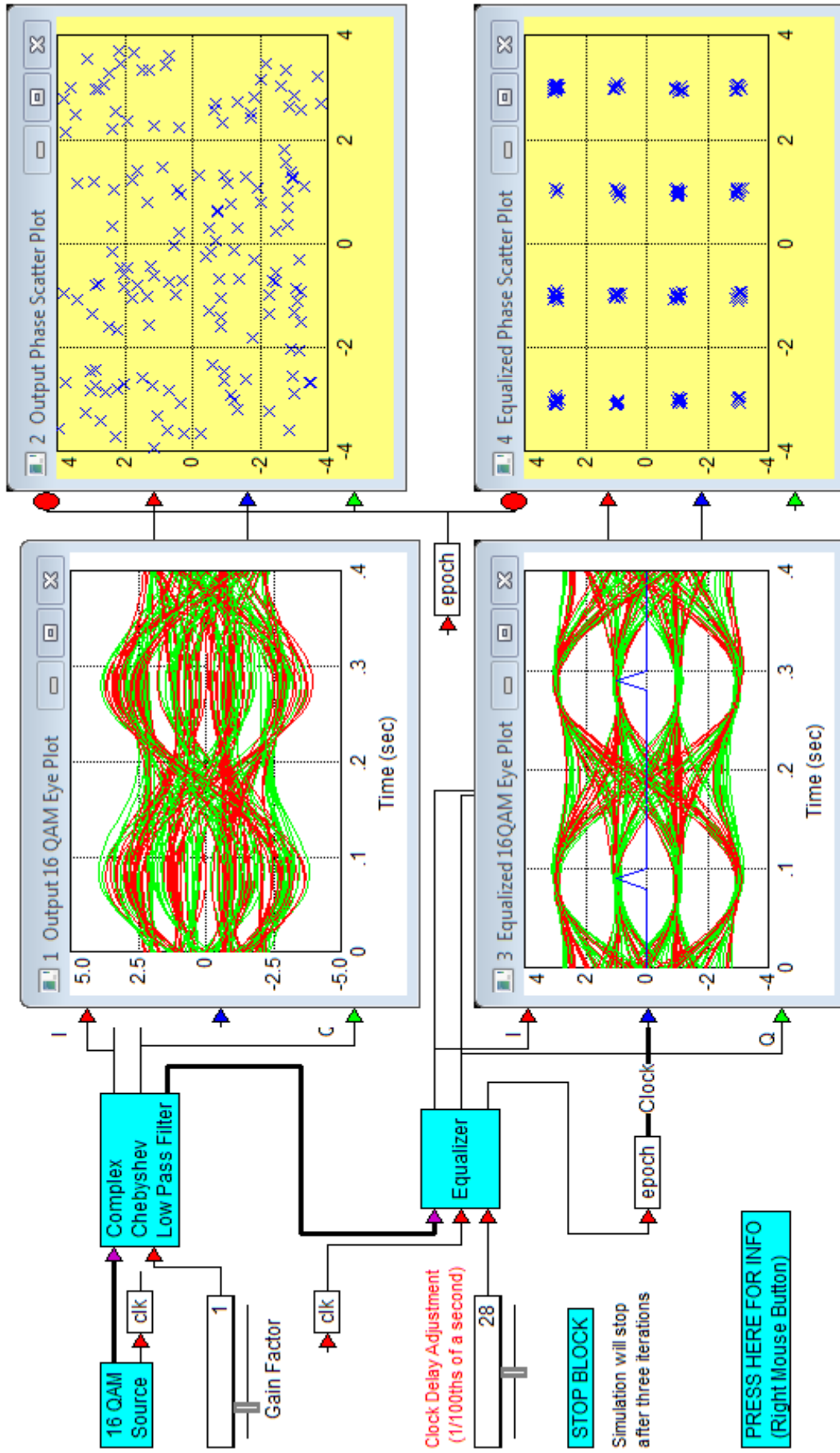


Рисунок 6.7. Схема системы с адаптивным фильтром эквалайзера



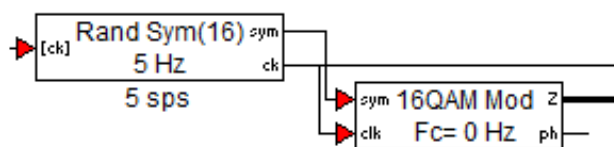


Рисунок 6.8. Субблок 16 QAM Source

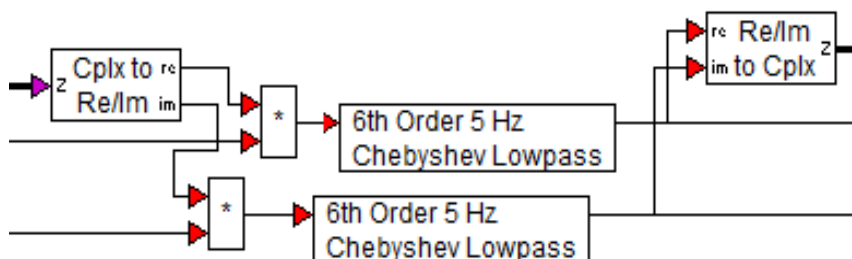


Рисунок 6.9. Субблок Complex Chebyshev Low Pass Filter

Субблок эквалайзера содержит (см. рис. 6.10) собственно эквалайзер, на входы которого поступают: In – модулированные сигналы с выхода фильтра Чебышева; Clk – тактовые сигналы, задержанные блоком задержки на время, определяемое положением движка регулятора (каждое деление – 0,01 с); Init – векторный сигнал с субблока External Taps в виде кода 0000100000, задающий начальное состояние регистра сдвига эквалайзера (первичная настройка при установлении соединения, например, двух модемов); на вход Eg – сигнал ошибки с субблока Error Calc (рис. 6.11), который формируется в результате сравнения сумматором модулированных сигналов с выхода Out эквалайзера и вторично модулированных блоком 16QAM Mod после детектирования блоком 16QAM Detect исходного модулированного сигнала.

Адаптивность эквалайзера определяется использованием отрицательной обратной связи, эффективность которой в процессе передачи информации определяется выбором задержки тактового сигнала с помощью более общего контура обратной связи, работающего на принципе определения ошибочных битов на приемной стороне. В данном случае роль такого контура выполняет регулятор Clock Delay Adjustment; при задержке 0,28 с (см. дисплей регулятора) «глаза» стали более «выразительными» (осциллограмма 3), а сигнальное созвездие заняло положенные ему 16 позиций (осциллограмма 4).

На осциллограмме 3 показаны также (в качестве «зрочков») два тактовых импульса, которые через переменную *epoch* передаются на входы запуска плоттеров сигнальных созвездий.

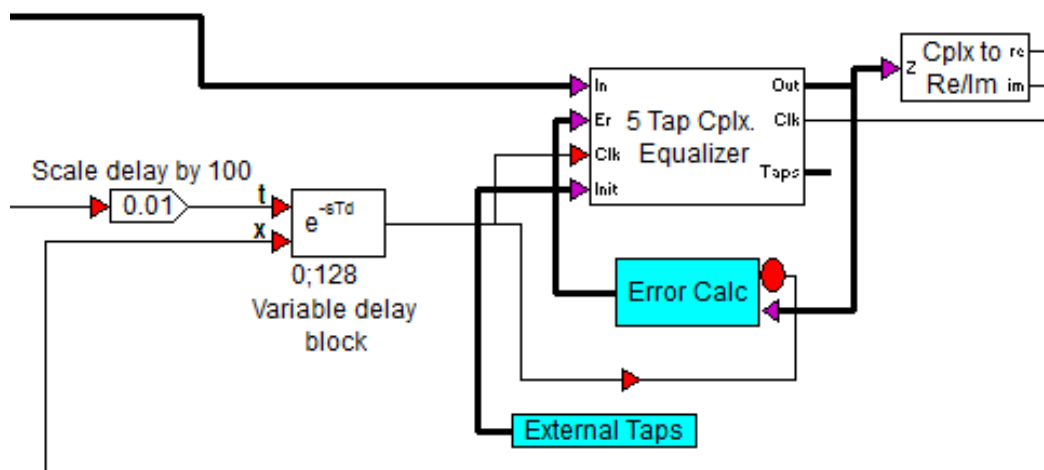


Рисунок 6.10. Субблок Equalizer

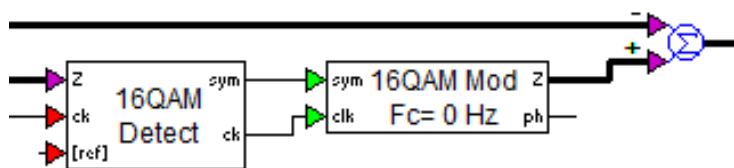


Рисунок 6.11. Субблок Error Calc

## 6.4. Лабораторная работа 6. Исследование особенностей работы систем фильтрации сигнала

### 6.4.1. Цель и содержание

Целью лабораторной работы является приобретение студентами практических навыков исследования принципов функционирования систем фильтрации сигнала.

#### Содержание

1. Исследование особенностей работы схемы IIR-фильтра Чебышева.
2. Исследование особенностей работы схемы FIR-фильтра.
3. Исследование особенностей работы схемы адаптивной фильтрации.

### **6.4.2. Аппаратура и материалы**

Для изучения материалов этой лабораторной работы необходимо:

1. Компьютер под управлением операционной системы Windows XP/Vista/7 с параметрами по умолчанию.
2. Установленная инструментальная среда визуального моделирования динамических систем VisSim/Comm 7.0.
3. Текстовый процессор (Microsoft Word 2003/2007/2010 или его аналог), предназначенный для создания и редактирования отчета по лабораторной работе.

### **6.4.3. Порядок выполнения работы**

1. Изучение теоретического материала.
2. Выполнение задания, предлагаемого в лабораторной работе.

#### **Задание**

1. Запустить на выполнение файл 6-1-1.VSM. Проанализировать работу схемы IIR-фильтра Чебышева с различными настройками. Провести испытания и сравнительный анализ фильтров Баттерворта, Бесселя, Чебышева и эллиптического фильтра с различными типами АЧХ.

2. Запустить на выполнение файл 6-2-1.VSM. Провести моделирование схемы FIR-фильтра. Получить АЧХ, ФЧХ и импульсную характеристику для всех фильтров, перечисленных в разделе Filter Type диалогового окна на рис. 6.6. Ознакомится с другими характеристиками этих фильтров в режиме View Response.

3. Запустить на выполнение файл 6-3-1.VSM. Провести моделирование схемы адаптивного фильтра. Изменяя регулятором Gain Factor (рис. 6.7) уровень входного сигнала, установить его влияние на Eye-диаграмму и сигнальное созвездие на выходе фильтра Чебышева и эквалайзера.

Изменяя регулятором Clock Delay Adjustment (рис. 6.7) задержку синхросигнала, установить ее влияние на Eye-диаграмму и сигнальное созвездие на выходе фильтра Чебышева и эквалайзера.

Подключить к выходам блоков перемножителей субблока Complex Chebyshev Low Pass Filter (рис. 6.9) плоттеры с настройками, используемыми в плоттерах 1 Output 16 QAM Eye Plot и 2 Output Phase Scatter Plot. Полученные осциллограммы сравнить с показанными на рис. 6.7.

#### **6.4.4. Вопросы для защиты работы**

1. Назовите типы цифровых фильтров.
2. Какие алгоритмы используются при создании цифровых фильтров?
3. Какими параметрами характеризуются фильтры?
4. Как классифицируются фильтры по виду АЧХ?
5. Дайте краткую характеристику фильтров Баттерворта, Бесселя и Чебышева.

## ГЛАВА 7. УСТРОЙСТВА СИНХРОНИЗАЦИИ СИГНАЛА

### 7.1. Схемы реализации фазовой автоподстройки частоты

Вопросы синхронизации наиболее остро стоят в синхронных системах передачи информации [9]. Для надежного выделения на приемной стороне синхроимпульсов используются различные методы, включая и специальное кодирование.

Рассмотрим технологию фазовой автоподстройки частоты (ФАПЧ – PLL, Phase Locked Loop). Принцип ФАПЧ основан на сравнении в схеме усиления/управления входящего и выходящего частотных сигналов, и изменении частоты выходящего сигнала до исчезновения разницы фаз со входящим. На рис. 7.1 показаны схемы ФАПЧ, где выходной сигнал снимается с зажима  $U_o$ , а выделенный синхроимпульс подается на зажим  $U_i$ .

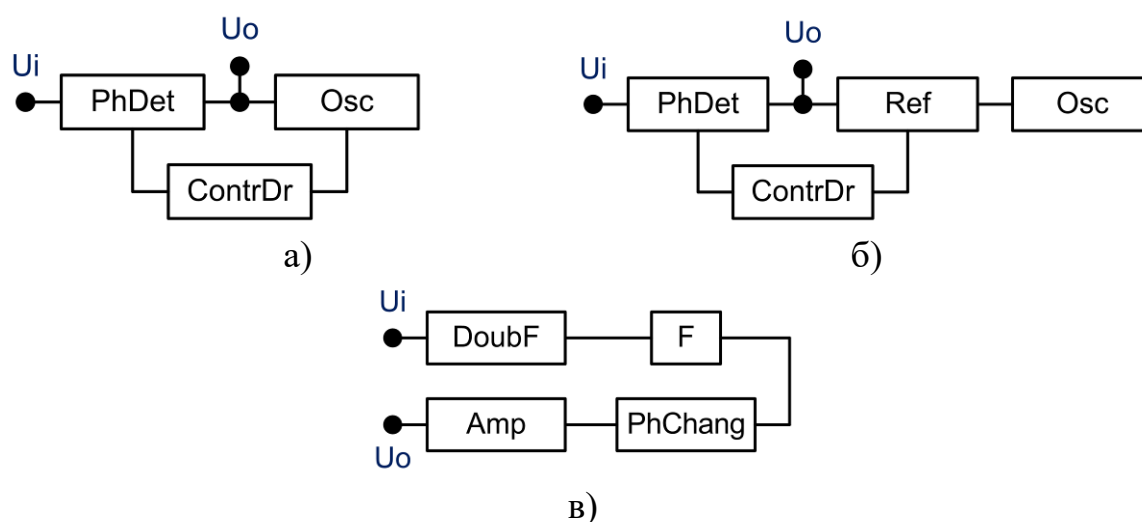


Рисунок 7.1. Схемы фазировки синхросигналов в приемнике

В схеме на рис. 7.1а выделенный в приемнике из информационной последовательности сигнал синхронизации поступает на вход  $U_i$  фазового детектора PhDet, а на его другой вход  $U_o$  – сигналы тактового генератора Osc. Если эти два сигнала смещены по фазе, то разность фаз преобразуется детектором в пропорциональный ей сигнал, который через блок управления ContrDr воздействует на генератора Osc таким образом, чтобы его фаза (соответственно, и частота) отличалась на величину, определяемую

коэффициентом усиления в петле отрицательной обратной связи. Распространенной реализацией такой схемы является использование управляемого напряжением генератора и усилителя выходного сигнала детектора в качестве блока управления.

В схеме на рис. 7.1б на вход детектора  $U_0$  подается сигнал с преобразователя Ref, в качестве которого используется двоичный или реверсивный счетчик. В первом случае, в зависимости от соотношения фаз на входах фазового детектора, блок управления ContrDr добавляет или вычитает некоторое количество импульсов из содержимого счетчика (возможен вариант и программируемого счетчика, коэффициент счета которого  $K$  меняется блоком управления). Во втором случае блоком управления осуществляется переключение направления счета – суммирование или вычитание. Очевидно, что частота генератора Osc должна быть выше тактовой в  $K$  раз.

Схема на рис. 7.1в относится к классу разомкнутых (без обратной связи), т. е. предполагается, что тактовая частота надежно выделяется из информационного потока. В этой схеме частота тактового сигнала удваивается блоком DoubF и через узкополосный фильтр F, настроенный на частоту тактового сигнала, и фазовращатель PhChang подается на усилитель-ограничитель Amp для формирования прямоугольных импульсов тактовой частоты.

Удвоение частоты вводится в целях повышения надежности выделения сигналов тактовой частоты в случае нарушения периодичности следования информационных нулей и единиц. Фазовращатель предназначен для компенсации задержки сигнала в удвоителе и фильтре. Устойчивая работа схемы обеспечивается при изменении тактовой частоты передатчика, не превышающем полосы пропускания фильтра.

Рассмотрим четыре модификации схемы ФАПЧ (рис. 7.1а) реализованные в среде VisSim (рис. 7.2) с использованием демофайла ChargePump\_PLL.vsm. Во всех модификациях в качестве  $U_i$  использованы источники синусоидального сигнала частотой 1,9 Гц и начальной фазой  $30^\circ$ .

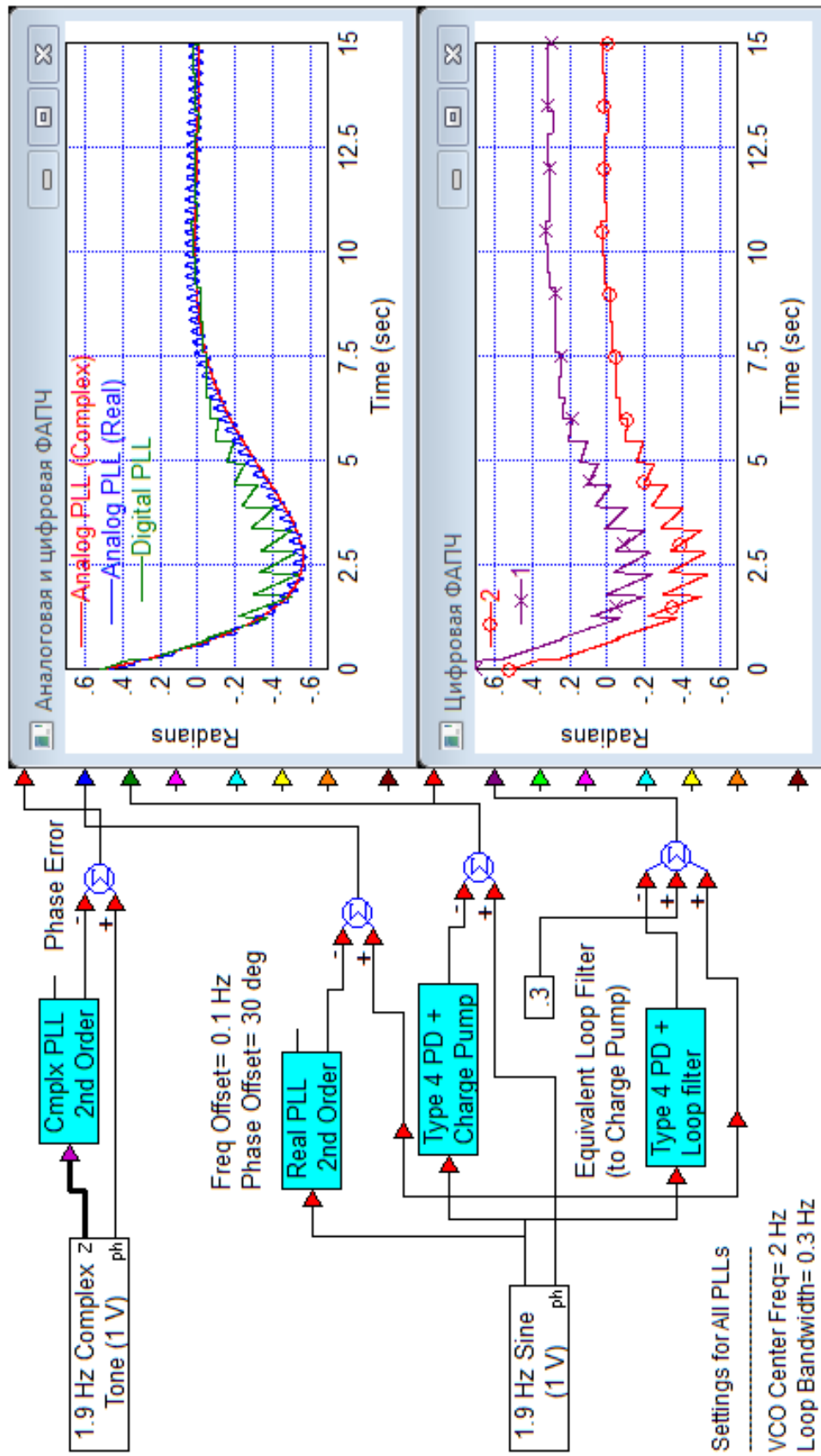


Рисунок 7.2. Схемы автоподстройки частоты и фазы

В качестве подстраиваемого генератора Osc используется источник VCO (Voltage Controlled Oscillator) с управляемой частотой и параметрами: начальная частота (Center Frequency) – 2 Гц; начальная фаза (Initial Phase) –  $0^\circ$ . Таким образом, в исходном состоянии разность тактовых частот передатчика и приемника составляет 0,1 Гц, а разность фаз –  $30^\circ$ .

Результаты моделирования представлены осциллограммами переходных процессов подстройки фазы для каждого из четырех вариантов. Необходимая при этом разность фаз формируется сумматорами. Из приведенных осциллограмм видно, что время выхода в установившийся режим для всех вариантов составляет около 7,5 с при перерегулировании около 0,5 рад ( $30^\circ$ ).

Перейдем теперь к рассмотрению каждого варианта схемы подстройки.

Первый вариант (для комплексного входного сигнала) реализуется субблоком Cmplx PLL 2nd Order (рис. 7.3). В этой схеме фазовый детектор выполнен на перемножителе комплексных сигналов Cplx Mult; его выходной сигнал поступает на преобразователь Cplx to Re/Im, где выделяется его мнимая составляющая, которая через согласующий усилитель Kd и фильтр PLL с полосой пропускания 0,3 Гц поступает на управляющий вход VCO. Выходной сигнал фазы (VCO Phase) подается на сумматор для формирования осциллограммы Analog PLL (Complex), а комплексный сигнал через преобразователь комплексной величины в комплексно сопряженную (Cplx Conj) – на второй вход фазового детектора. Действительная часть входного и выходного комплексного сигналов отображается на дисплее плоттера IN/OUT; по виду осциллограмм можно судить о процессе подстройки частоты. Однако процесс подстройки по фазе более нагляден. Нетрудно заметить, что осциллограмма Analog PLL (Complex) переходного процесса для рассматриваемого варианта (см. рис. 7.2) отличается плавностью кривой (без пульсаций).

Второй вариант (для действительных сигналов) реализуется субблоком Real PLL 2nd Order (рис. 7.4).



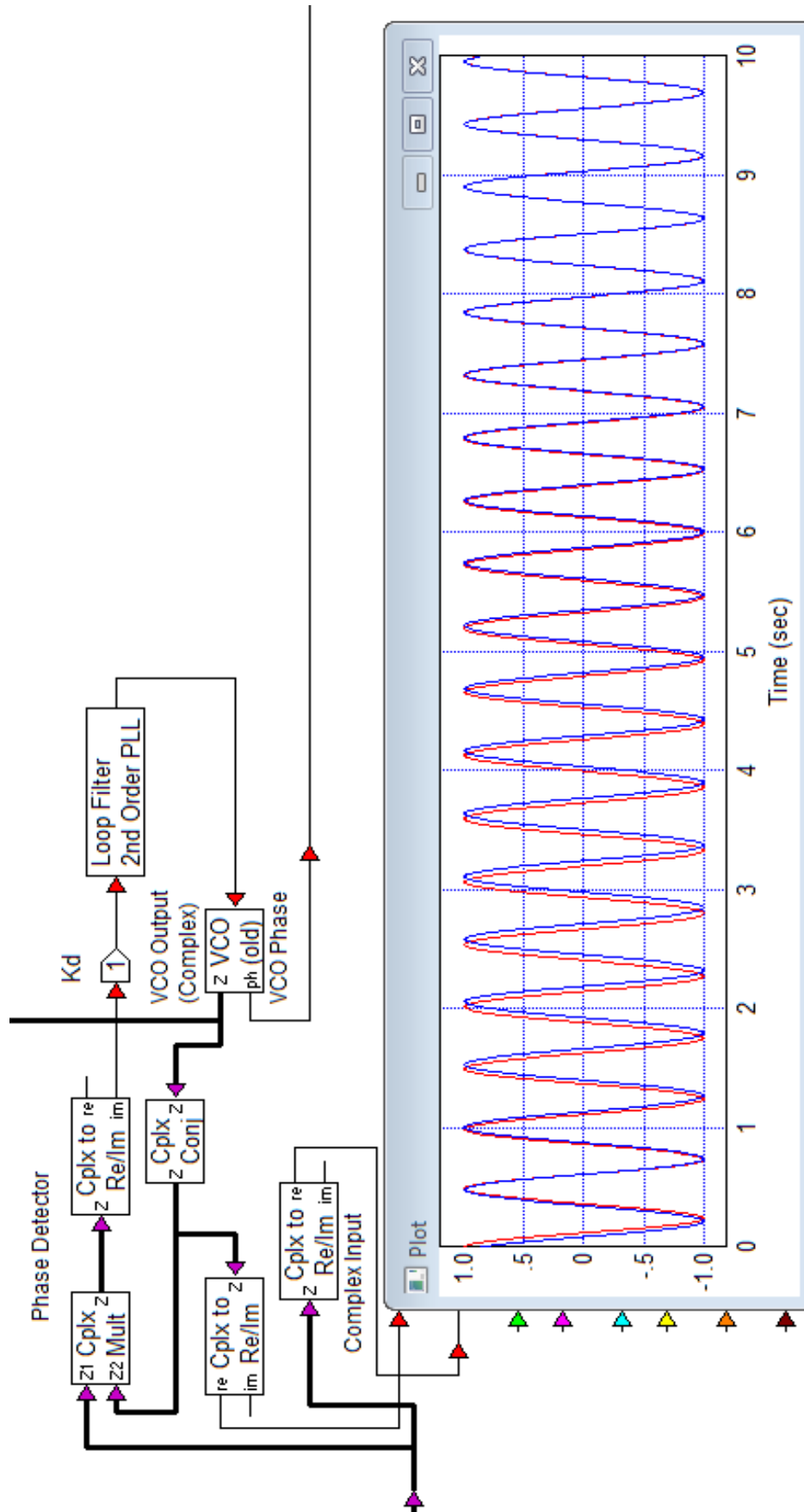


Рисунок 7.3. Схема субблока Cmplx PLL 2nd Order

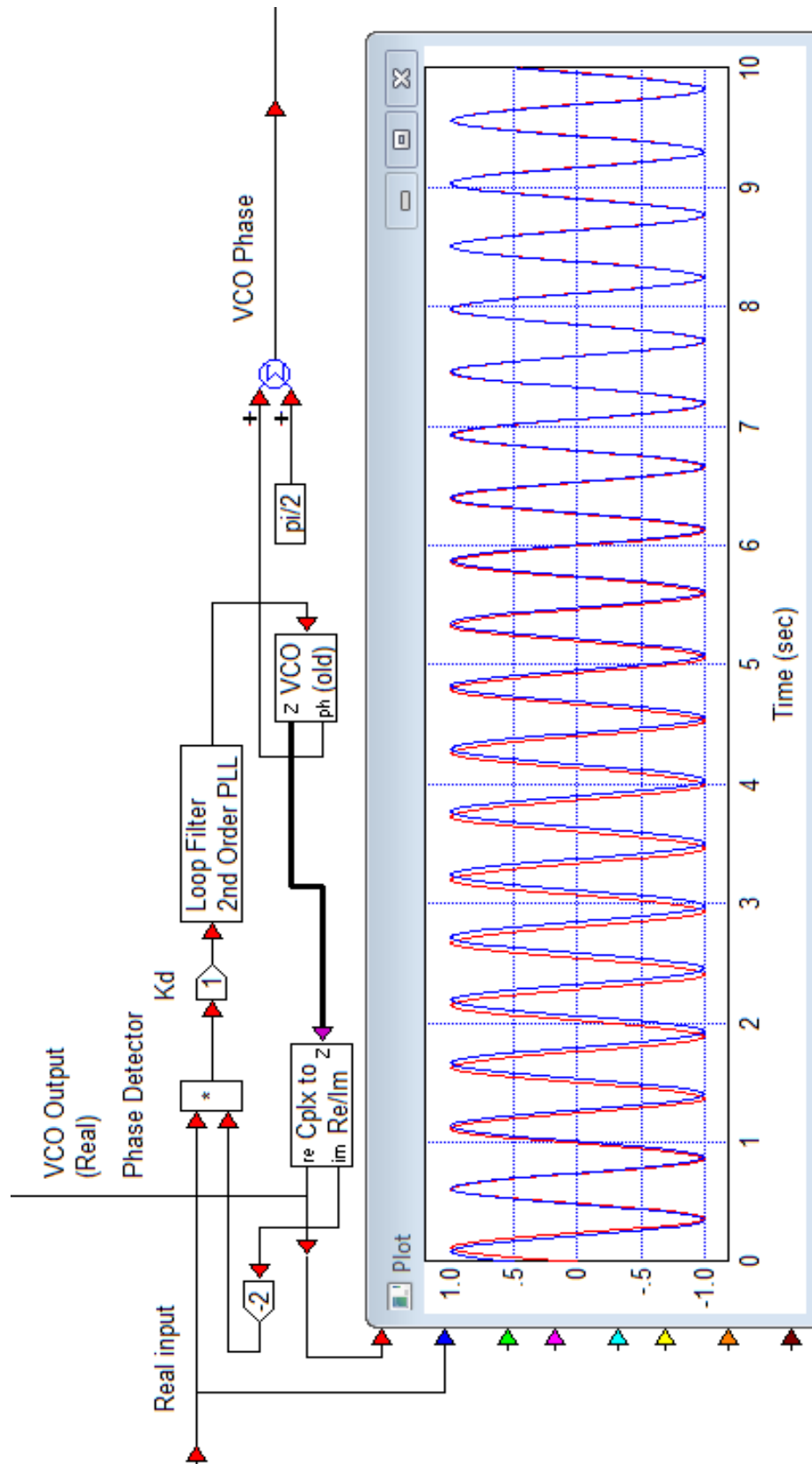


Рисунок 7.4. Схема субблока Real PLL 2nd Order

В этой схеме фазовый детектор выполнен на перемножителе сигнала с генератора Sine и сигнала тактовой частоты, поступающего по цепи обратной связи: усилитель  $K_d$  – фильтр (Loop Filter) – VCO – преобразователь Cplx to Re/Im – инвертирующий усилитель. В этой схеме начальная фаза VCO выбрана равной  $-90^\circ$ . Поэтому в целях удобства сравнения осциллограммы Analog PLL (Real) на рис. 7.2 с другими вариантами (она отличается повышенными пульсациями) к выходному сигналу фазы (VCO Phase) добавляется с помощью сумматора и источника число  $\pi/2$  ( $90^\circ$ ).

Третий вариант (также для действительных сигналов) является цифровым и реализуется субблоком Type 4PD + Charge Pump (рис. 7.5). В этой схеме используется фазовый детектор типа 4 повышенной чувствительности, на выходы UP и DN которого выводится сигнал ошибки (рассогласования) по фазе и частоте в виде логических 0, 1 (осциллограмма на плоттере 1). Сигнал рассогласования поступает далее на активный фильтр первого порядка (Charge Pump), являющийся одновременно сравнивающим устройством для сигналов фаза/частота на входах UP и DN. Выходной сигнал блока Charge Pump (осциллограмма на плоттере 2) является управляющим для VCO, выходные сигналы которого поступают через преобразователь Cplx to Re/Im на вход детектора, а по фазе – на вход сумматора, выходной сигнал которого подается как на верхний (осциллограмма Digital PLL), так и на нижний плоттер (осциллограмма 2). Из этих осциллограмм видно, что в переходном режиме сигнал рассогласования по фазе отличается большими пульсациями.

Четвертый вариант реализуется субблоком Type 4PD + Loop filter (рис. 7.6) и является полным аналогом третьего: в нем блок Charge Pump заменен его эквивалентной схемой, состоящей из инвертирующего усилителя, сумматора, усилителя и фильтра. Осциллограмма 1 для этого варианта смещена источником  $C$  вверх на 0,3 рад и полностью совпадает с осциллограммой 2, в чем можно убедиться, выбрав сигнал смещения равным нулю.

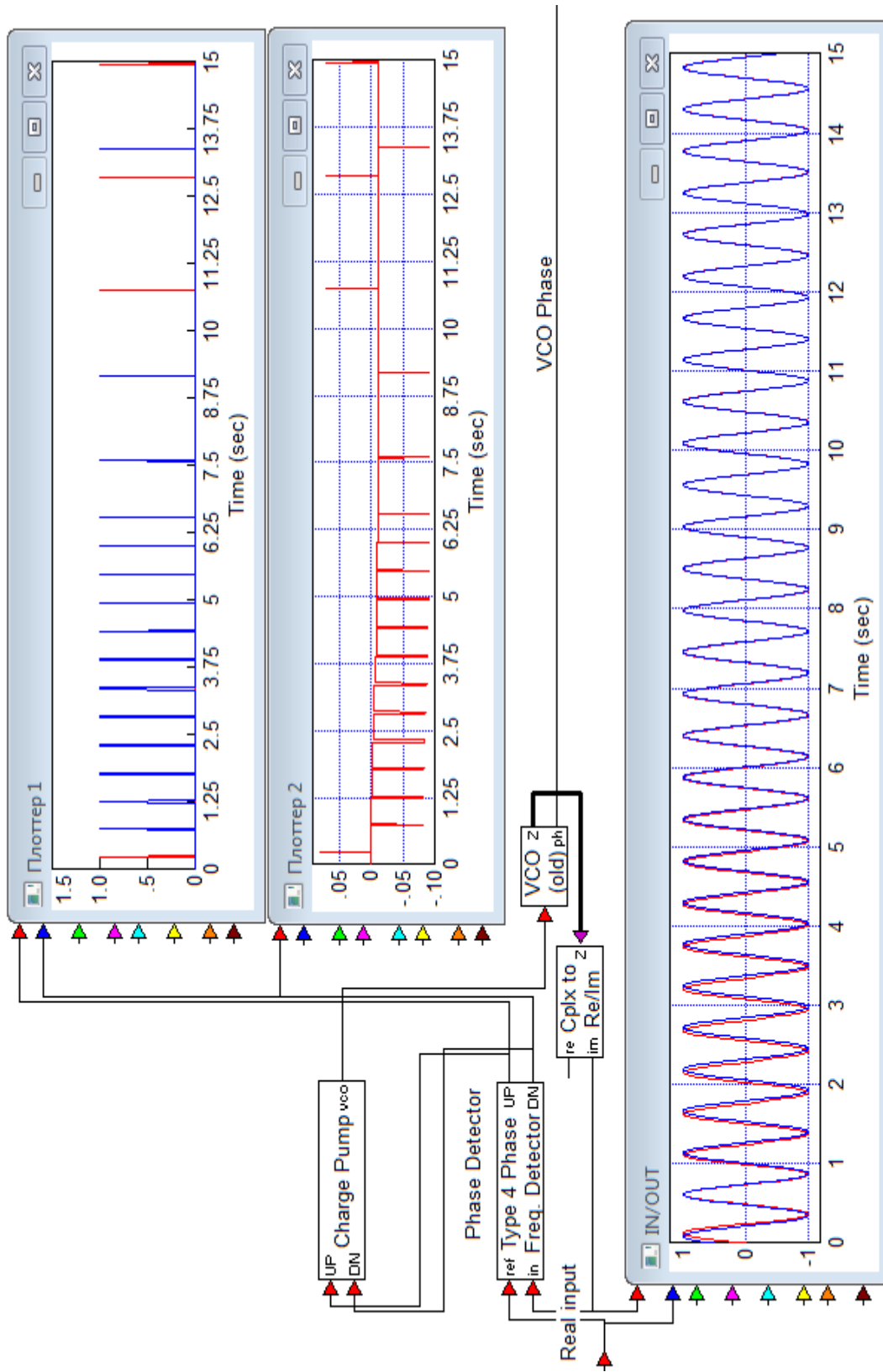


Рисунок 7.5. Схема субблока Type 4PD + Charge Pump

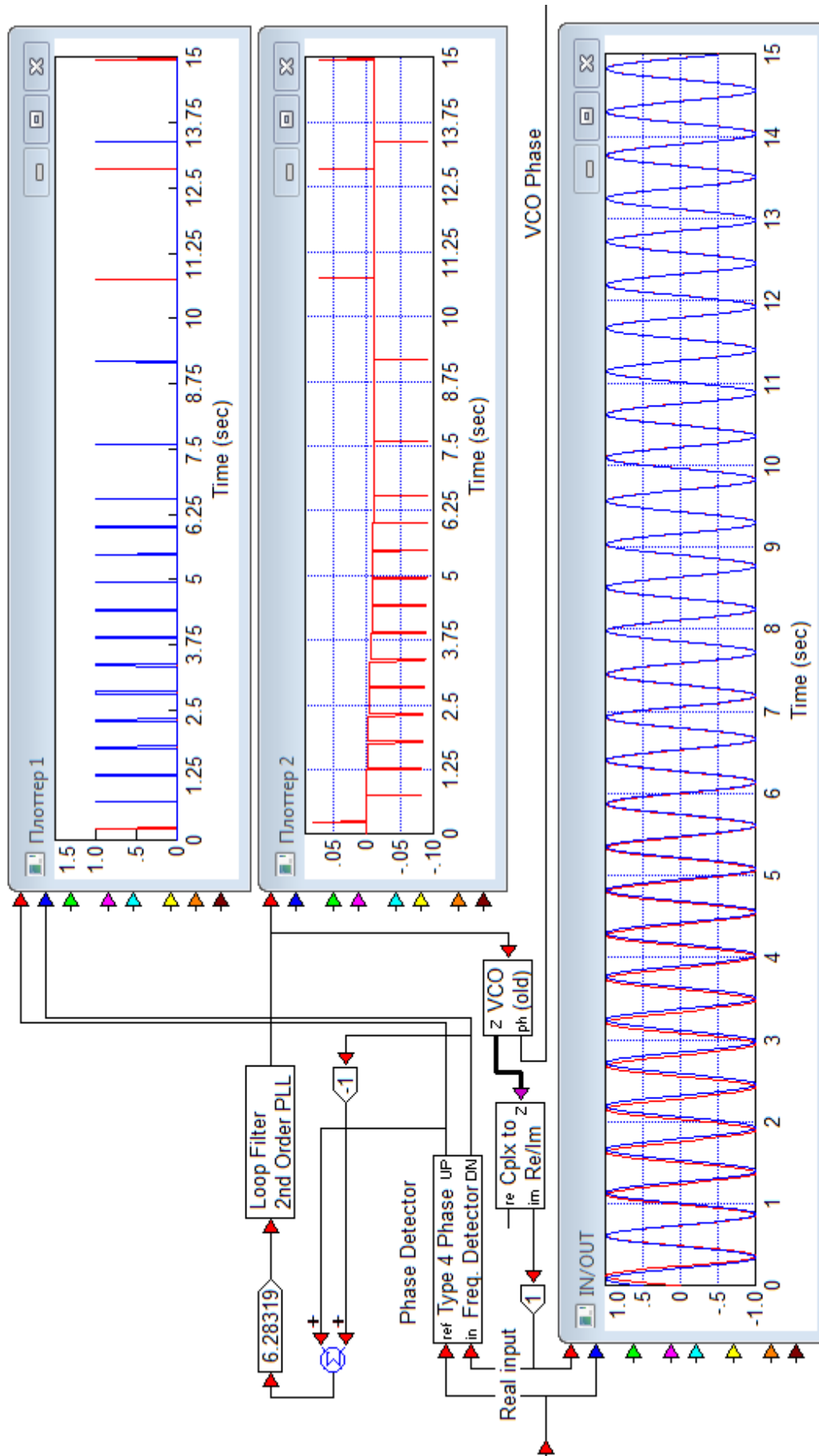


Рисунок 7.6. Схема субблока Type 4PD + Loop filter

## **7.2. Лабораторная работа 7. Исследование особенностей работы системы фазовой автоподстройки частоты**

### **7.2.1. Цель и содержание**

Целью лабораторной работы является приобретение студентами практических навыков исследования принципов синхронизации сигнала на примере функционирования системы фазовой автоподстройки частоты.

#### **Содержание**

1. Исследование схемы фазовой автоподстройки частоты.
2. Исследование переходных процессов в схемы фазовой автоподстройки частоты.

### **7.2.2. Аппаратура и материалы**

Для изучения материалов этой лабораторной работы необходимо:

1. Компьютер под управлением операционной системы Windows XP/Vista/7 с параметрами по умолчанию.
2. Установленная инструментальная среда визуального моделирования динамических систем VisSim/Comm 7.0.
3. Текстовый процессор (Microsoft Word 2003/2007/2010 или его аналог), предназначенный для создания и редактирования отчета по лабораторной работе.

### **7.2.3. Порядок выполнения работы**

1. Изучение теоретического материала.
2. Выполнение задания, предлагаемого в лабораторной работе.

#### **Задание**

1. Запустить на выполнение файл 7-1-1.VSM. Провести анализ схемы фазовой автоподстройки частоты. Установить зависимость времени переходного процесса (на уровне 5 % от установившегося значения) при рассогласовании по фазе  $30^\circ$  от частоты, выбрав ее значения равными 0,2; 0,3 и

0,4 Гц (производится установкой в диалоговых окнах источников Complex Tone и Sine значений частоты 1,8; 1,7 и 1,6 Гц).

2. Выполнить задание по п. 1 при рассогласовании по частоте 0,1 Гц и фазе 40, 50 и 60° (значения устанавливаются непосредственно в диалоговых окнах источников Complex Tone и Sine).

3. Установить зависимость формы и времени переходного процесса при рассогласовании по фазе 30° и частоте 0,1 Гц от полосы пропускания фильтров (параметр Loop Bandwidth блоков Loop filter и Charge Pump) при ее значениях 0,1, 0,2, 0,5 и 0,6 Гц.

#### **7.2.4. Вопросы для защиты работы**

1. Каким образом осуществляется синхронизация в асинхронных системах?

2. Прокомментируйте особенности реализации схем фазовой автоподстройки частоты.

## ЛИТЕРАТУРА И ИСТОЧНИКИ

### *Основная литература*

1. Будылдина, Н. В. Основы передачи дискретных сообщений : учебное пособие / Н. В. Будылдина. – Екатеринбург : УрТИСИ ГОУ ВПО «СибГУТИ», 2009. – 142 с.
2. Карлащук, В. И. Электронная лаборатория на IBM PC. Инструментальные средства и моделирование элементов практических схем / В. И. Карлащук, С. В. Карлащук. – М. : Солон-Пресс, 2008. – 139 с.
3. Марченко, А. Л. Основы преобразования информационных сигналов : учебное пособие / А. Л. Марченко, Е. А. Марченко – М. : Горячая линия – Телеком, 2008. – 280 с.
4. Олифер, В. Г. Компьютерные сети: принципы, технологии, протоколы : учебник – 4-е изд. / В. Г. Олифер, Н. А. Олифер. – СПб.: Питер, 2012. – 944 с.

### *Дополнительная литература и источники*

5. Введение в Multisim. Трехчасовой курс [электронный ресурс] // <http://www.ni.com/russia/>.
6. Дьяконов, В. П. VisSim + Mathcad + MATLAB. Визуальное математическое моделирование / В. П. Дьяконов. – М. : Солон-Пресс, 2004. – 384 с.
7. Карлащук, В. И. Электронная лаборатория на IBM PC. Лабораторный практикум на Electronics Workbench и VisSim по элементам телекоммуникационных систем / В. И. Карлащук. – М. : Солон-Пресс, 2005. – 480 с.
8. Карлащук, В. И. Электронная лаборатория на IBM PC. Том 1. Моделирование элементов аналоговых систем на Electronics Workbench и MATLAB / В. И. Карлащук. – 6-е изд., пер. и доп. – М. : Солон-Пресс, 2006. – 720 с.
9. Карлащук, В. И. Электронная лаборатория на IBM PC. Том 2. Моделирование элементов телекоммуникационных и цифровых систем на VisSim и Electronics Workbench / В. И. Карлащук. – 6-е изд., пер. и доп. – М. : Солон-Пресс, 2006, – 640 с.



10. Ломовицкий, В. В. Основы построения систем и сетей передачи информации : учеб. пособие для вузов / В. В. Ломовицкий, А. И. Михайлов, К. В. Шестак, В. М. Щекотихин. – М. : Горячая линия – Телеком, 2005. – 384 с.
11. Марченко, М. В. Системы документальной электросвязи : учебное пособие / М. В. Марченко. – Ульяновск : УлГТУ, 2007. – 90 с.
12. Передача дискретных сообщений : учебник для вузов / В. П. Шувалов, Н. В. Захарченко, В. О. Шварцман и др. ; под ред. В. П. Шувалова. – М. : Радио и связь, 1990. – 464 с.
13. Прокис, Д. Цифровая связь / Д. Прокис; под ред. Д. Д. Кловского. – М. : Радио и связь, 2000. – 800 с.
14. Скляр, Б. Цифровая связь. Теоретические основы и практическое применение. – 2-е изд., испр. : пер. с англ. / Б. Скляр – М. : Издательский дом «Вильямс», 2007. – 1104 с.
15. Хемминг, Р. В. Цифровые фильтры: пер. с англ. / Р. В. Хемминг. – М. : Недра, 1987. – 224 с.
16. Шварцман, В. О. Теория передачи дискретной информации / В. О. Шварцман, Е. А. Емельянов. – М. : Связь, 1979. – 424 с.
17. Eritek, Inc. [электронный ресурс] // <http://www.eritek.com/>.
18. Представительство корпорации National Instruments в России СНГ и Балтии [электронный ресурс] // <http://russia.ni.com/>.
19. NI Multisim [электронный ресурс] // <http://www.ni.com/multisim/>.
20. VisSim [электронный ресурс] // <http://www.vissim.com/products/vissim.html/>.
21. VisSim/Comm™ [электронный ресурс] // <http://www.vissim.com/products/vissim/comm.html/>.
22. Visual Solutions, Inc. [электронный ресурс] // <http://www.vissim.com/>.
23. ГОСТ 2.105-95 «Общие требования к текстовым документам». – ГОСТ 2.105-95 «Общие требования к текстовым документам». – М. : Изд-во стандартов, 1998.

## **ПРИЛОЖЕНИЕ 1. ИНСТРУМЕНТАЛЬНАЯ СРЕДА ВИЗУАЛЬНОГО МОДЕЛИРОВАНИЯ ДИНАМИЧЕСКИХ СИСТЕМ VISSIM**

### **П1.1. Структура окна и система меню**

Пакет моделирования и имитации VisSim 7.0 от компании Visual Solutions [20, 22] сочетает интуитивный интерфейс блок-диаграмм с мощным процессором динамической имитации для создания быстрых и точных решений для самых сложных систем.

Для моделирования телекоммуникационных систем используют версию VisSim/Comm 7.0, которая содержит библиотеку с блоками источников сигналов, кодеров, модуляторов, каналов, демодуляторов, декодеров, приемников и др [17, 21].

Окно программы VisSim/Comm 7.0 (рис. П1.1) содержит поле меню, набор мнемокнопок и рабочее поле из двух частей. В левой в виде дерева перечислены составные блоки (субблоки) схемы модели, расположенной в правой части. Дерево может быть использовано для оперативной навигации по блок-схеме.

Выбранный блок или субблок с его содержимым отображается в правой части окна. При необходимости изменения соотношения размеров подокон дерева и рабочей области, разделяющая их линия курсором мыши перемещается вправо-влево.

Наборы кнопок сгруппированы по функциональному назначению, причем каждая группа может быть снята со своей «полки» и перемещена курсором мыши в любое место окна, в том числе и на любую «полку».

Первые семь кнопок первой (верхней) «полки» являются типичными для Windows-систем, назначение других кнопок, а также строки состояния, которая расположена в нижней части окна будет обсуждено при рассмотрении связанных с ними команд.

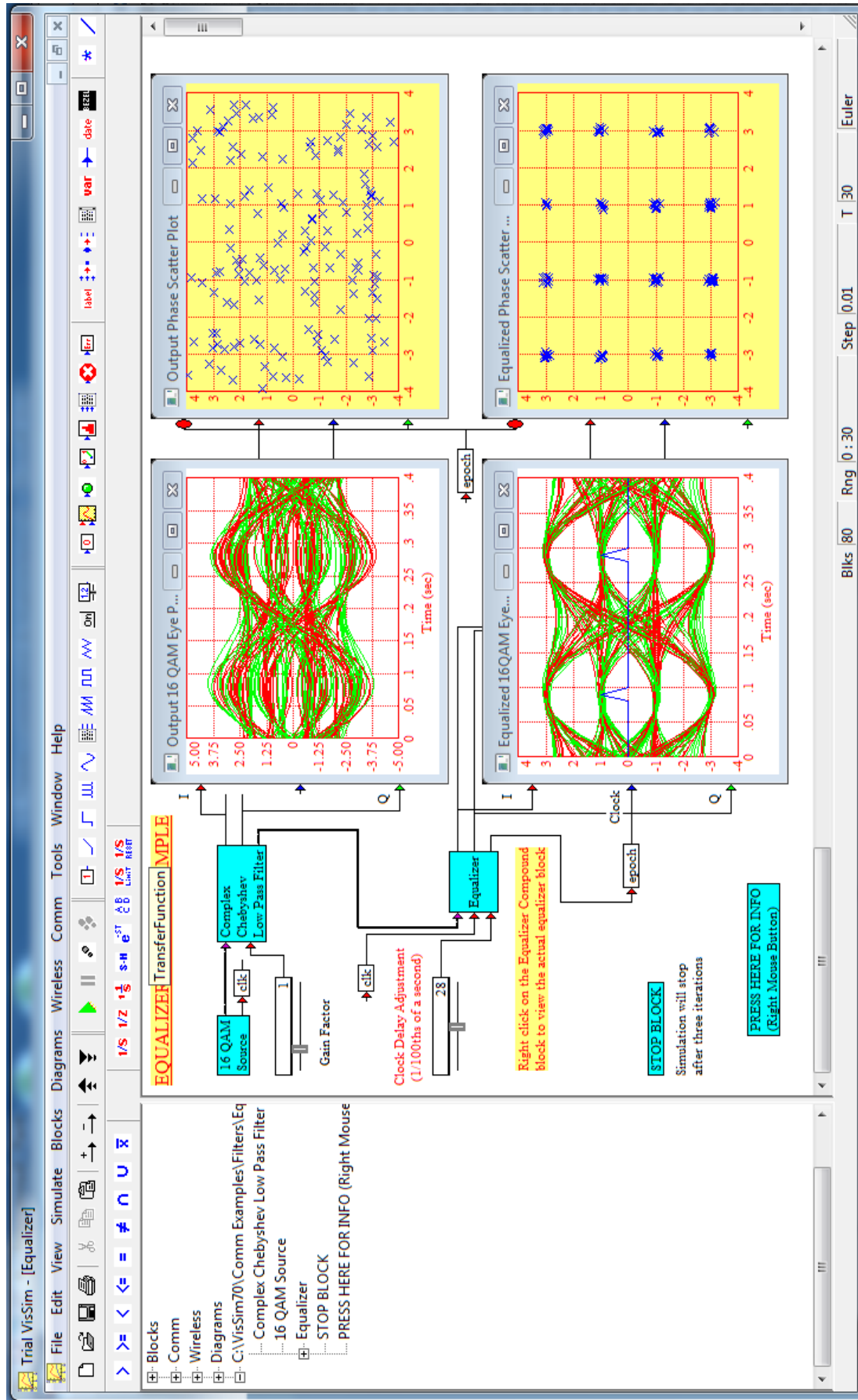


Рисунок П1.1. Окно программы VisSim/Comm

### П1.1.1. Меню File

**New** (Ctrl + N), **Open** (Ctrl + O), **Close**, **Save** (Ctrl + S), **Save As**, **Save All** – набор команд для работы с файлами, включая экспорт файлов с расширениями .DAT, .M, .MAP, .MAT и .WAV; файлы программы имеют расширение .VSM.

**Add** – добавить в текущую схему модели данные из выбранного в стандартном окне файла; после выбора команды переносимая схема имеет габариты, очерченные штрихпунктирной линией, что позволяет разместить ее без наложения на основную схему.

**Save Embedded Files** – запись внесенных изменений в текущую схему модели во все блоки проекта, в которых она используется.

**Save As Metafile** – записать изображение текущей схемы модели в формате метафайла Windows.

**Send** – передача по электронной почте текущей (находящейся в рабочем окне) схемы модели.

**Diagram Information** – информация о текущей модели: название, имя автора, комментарий, режим доступа (пароль, только для чтения) и др.

**Page Setup**, **Print** (Ctrl + P), **Printer Setup**, **Print Preview** – набор команд для вывода на печать.

Команда Page Setup имеет следующие особенности.

**Fit diagrams to page** – выровнять схему под выбранные размеры страницы; при выборе этой опции каждый уровень в окне дерева блок-схемы печатается на отдельной странице. Случай невозможности размещения данных на одной странице в заданном масштабе сопровождается предупредительным сообщением, после чего печать выполняется с максимальным уменьшением.

**Tile printed page for large diagrams** – специальный режим для больших блок-схем; при выборе этой опции каждый уровень в окне дерева блок-схемы печатается только в заданном режиме, однако отдельные элементы блок-схемы могут оказаться на разных страницах.

При оформлении верхнего (Header) и нижнего (Footer) колонтитулов в целях соблюдения стандартов по оформлению документации наравне с

обычным текстом могут быть использованы следующие спецсимволы: \$f – имя файла; \$F – путь к файлу; \$H – путь к блоку; \$D – дата; \$I – метод интегрирования; \$O – оптимизация; \$p – номер страницы; \$G – диапазон моделирования; \$S – размер шага; \$L, \$C, \$R – выравнивание по левому краю, по центру и правому краю.

Результаты выбранных настроек могут быть проконтролированы в окне команды Print Preview.

**Create Report** – создание отчета в формате .txt по текущей схеме; в диалоговом окне команды опционально выбираются автоматически включаемые в отчет такие сведения, как переменные (Include Variables), метки выводов (Include Connector Labels), комментарии (Include Comments), справочные файлы (Include Reference Files), псевдонимы (Include Aliases), а также возможность сортировки данных по уровню значимости (Sort by Level).

**Real Time Config** – конфигурирование системы для работы с аппаратными устройствами ввода/вывода, однако для этого необходимо использовать дополнительно специальные программные продукты (Commsim/Real-Time, Commsim/Real-TimePRO или Commsim DACQ).

**Поле со списком использованных файлов.**

**Exit** (Alt + F4) – выход из программы.

### П1.1.2. Меню Edit

**Undo** (Ctrl + Z), **Redo** (Ctrl + Y) – одноступенчатые команды отмены/восстановления последнего выполненного при редактировании действия.

**Cut** (Ctrl + X), **Copy** (Ctrl + C), **Delete** (Del) – команды вырезания, копирования (с сохранением данных в буфере обмена) и удаления выделенных блоков или частей текущей схемы модели. Выделение, сопровождающееся «зачернением» выделяемого простого блока или «покраснением» составного, производится левой кнопкой мыши при нажатой клавише Shift или по правилу прямоугольника, при котором курсор мыши ставится в левый верхний угол

выделяемой области и при нажатой левой кнопке протягивается по ее диагонали.

**Paste** (Ctrl + V) – вставка содержимого буфера обмена; после выполнения команды прицепленный к курсору мыши вставляемый объект с габаритами, очерченными штрихпунктирной линией, размещается (после нажатия левой кнопки мыши) в нужном месте.

**Paste Link** – вставка из буфера обмена блока со связями DDE.

**Select All** (Ctrl + A) – выбор всех объектов.

**Clear Errors** (Ctrl + E) – сброс флага ошибки, возникающей в процессе моделирования и сопровождаемой сообщением о ее типе и красной подсветкой блока с ошибочным результатом. Рассматриваемая команда служит для того, чтобы при повторном запуске моделирования получить указанное сообщение еще раз, например для идентификации ошибки. Сброс ошибки отдельного блока может осуществляться также щелчком правой кнопки мыши по его значку, однако, если ошибок много, то целесообразно пользоваться командой Clear Errors.

**Flip Horizontal** (Ctrl + Left) – поворот одного или нескольких выделенных блоков по горизонтали (входы и выходы меняются местами).

**Create Compound Block** – создание субблока (составного блока) из предварительно выделенных блоков. При этом субблок снабжается выводами для каждой связи с внешними блоками, а также для каждого неподключенного вывода на внутренних блоках. В диалоговом окне команды (рис. П1.2) вводятся: имя блока (Compound Name); элементы защиты (Protection): закрыть (Locked), запретить внесение изменений (Read Only) или использовать пароль (Password); выбрать изображение для значка (Select Image) в формате BMP, например из VISSIM\BITMAP\DIAGRAM; использовать режим дисплея оператора (Hide in Display Mode), при котором субблок становится невидимым и, следовательно, недоступным для изменения его настроек. Для изменения цветового заполнения иконки блока используется команда Color Compound Blocks.

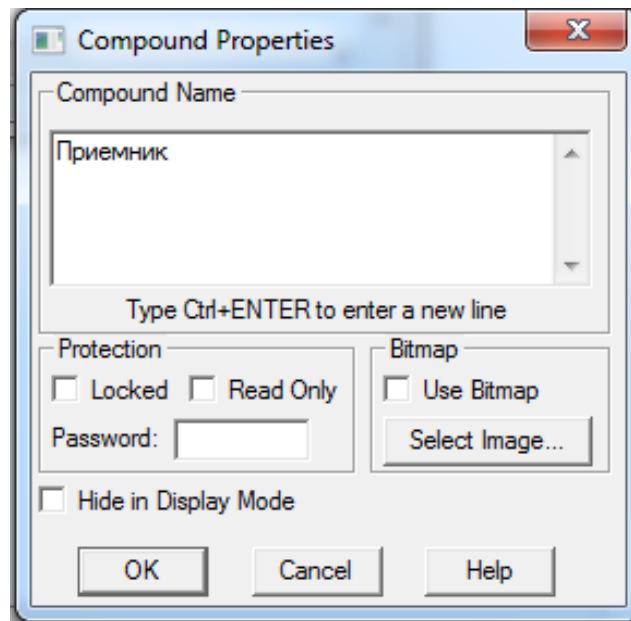


Рисунок П1.2. Окно создания субблока

**Dissolve Compound Block** – команда декапсуляции (разборки) субблока; при выборе команды курсор приобретает многострелочную форму в виде захвата, которым и осуществляется демонтаж выбранного блока.

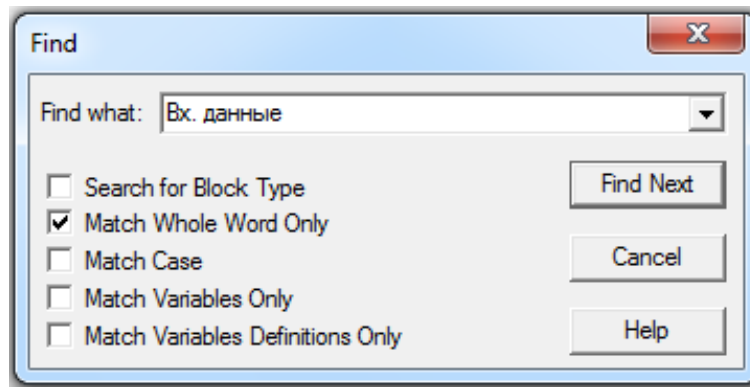
**Auto Connect** – автоматическое соединение выводов блоков, расположенных горизонтально друг за другом.

**Auto Disconnect** – автоматическое удаление соединения блоков, выполненного предыдущей командой.

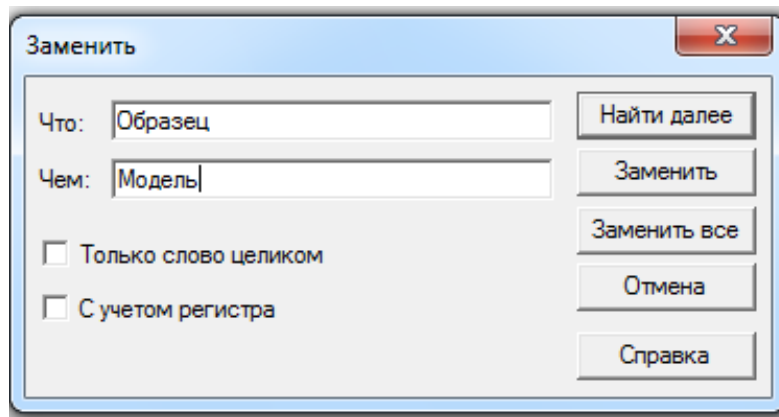
**Align** – команды выравнивания группы блоков: по левой (Left), правой (Right), или верхней (Top) граням; по входам (Input), выходам (Output) или в столбик (Stack) по вертикали.

**Find** (Alt + F3) – команда поиска элементов схем (рис. П1.3а):

- Find what – строка ввода для поиска (тип блока, имя, метка, текстовая строка, фрагмент строки и т. п.);
- Search for Block Type – поиск блоков определенного типа, которые можно выбрать в этом случае из списка в поле строки Find what;
- Match Whole Word Only – искать только целые слова;
- Match Case – поиск элементов, аналогичных указанным в поле Find what, но в другом регистре;



а)



б)

Рисунок П1.3. Окна команд Find и Replace

- Match Variable Only – искать только переменные (позволяет проследить места использования переменных);
- Match Variable Definitions Only – искать только определения (блоки) переменных;
- Find Next – кнопка для выполнения очередного шага поиска.

**Replace** – команда замены элементов схемы (рис. П1.3б), найденных, например, с помощью предыдущей команды.

**Block Properties** (Ctrl + RtBtn) – команда вызова диалогового окна с параметрами блока; в случае простого блока выполняется также установкой курсора мыши на значок блока и нажатием ее правой кнопки, когда курсор приобретает крестообразную форму, а в случае составного – при нажатой клавише Ctrl. В общем же случае выбор команды сопровождается курсором в виде гаечного ключа, с помощью которого вызывается диалоговое окно с параметрами выбранного блока. В случае составного блока (субблока)



диалоговое окно его свойств отличается от показанного на рисунке П1.2 следующими дополнительными параметрами:

- **Create Dialog from contained Dialog Constants** – опция, указывающая на возможность применения блоков DialogConstant для реконфигурации составного блока путем изменения его параметров и начальных условий без перехода к его компонентам в режиме дисплея оператора (включена опция Hide in Display Mode на рис. П1.2), при котором субблок недоступен; блоки DialogConstant в количестве, определяемом количеством настроек, включаются в состав субблока;

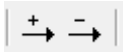
- **Enable Execution** – опция, разрешающая использование внешней синхронизации; при ее выборе на иконке составного блока формируется дополнительный ввод круглой формы красного или зеленого цвета; если на этом вводе значение сигнала  $\geq 1$ , то VisSim выполняет вычисления для всех внутренних блоков; если  $< 1$ , то такие вычисления не выполняются, а для других (внешних) блоков используются значения, полученные в предыдущий момент синхронизации; эта возможность позволяет в некоторых случаях сократить время моделирования;

- **Local Time Step** – опция, позволяющая назначить составному блоку локальный шаг модельного времени больший, чем для всей схемы модели, что позволяет уменьшить общее время моделирования;

- **Use Local Bounds** – опция, разрешающая использование сигналов с настраиваемыми параметрами, например, временем начала (Start) и окончания (End);

- **Contained Block Count** – строка, в поле которой индицируется общее количество библиотечных блоков, входящих в субблок;

- **Contained Computations Block** – строка для индикации числа вычислительных блоков.

**Add Connector, Remove Connector** – команды добавления и удаления выводов с правой или левой грани значка блока; выполняются также с помощью кнопок . При выборе команды добавления, курсор в

показанном на кнопке виде подводится к грани значка (снаружи для левой грани и изнутри для правой), и после появления перед острием зигзагообразной («змеиной») линии нажимается левая кнопка мыши; в случае команды удаления сигналом к нажатию кнопки является штрихпунктирный прямоугольник, формируемый вокруг удаляемого вывода (будет удален и подключенный к нему проводник).

**Toggle Tag** (Ctrl + F2) – вставка закладки (отметки) в виде зеленого треугольника в верхнем левом углу текущего уровня схемы (диаграммы) (при печати игнорируется).

**Goto Tag** (F2) – переход к уровню схемы с закладкой.

**Goto Next Change** – переход к уровню схемы с внесенными изменениями.

**Reset Bitmap Scaling** – команда согласования изображений элементов схемы модели с разрешением монитора.

**Repaint Screen** – обновить экран; удаляются все внесенные в процессе редактирования изменения.

**Preferences** – набор команд настройки, окно которых состоит из пяти закладок (Preferences, Path Aliases, Local Path Aliases, Addons, Menu Directories).

Первая закладка – **Preferences** – содержит следующие настройки (рис. П1.4):

- Show Horizontal Scroll Bar, Show Vertical Scroll Bar – показывать горизонтальные и вертикальные полосы прокрутки.

- High Precision Display – установка возможности увеличения разрядности отображаемых чисел от 6 до 15 десятичных цифр в регистраторах типа Display;

- Snap to Grid – опция привязки размещения блоков к сетке с размером ячеек 1/2 от размера выбранного шрифта;

- Auto Connect Radius – установка расстояния в дюймах между курсором мыши с проводником и выводом блока, при котором подключение к нему заканчивается автоматически;

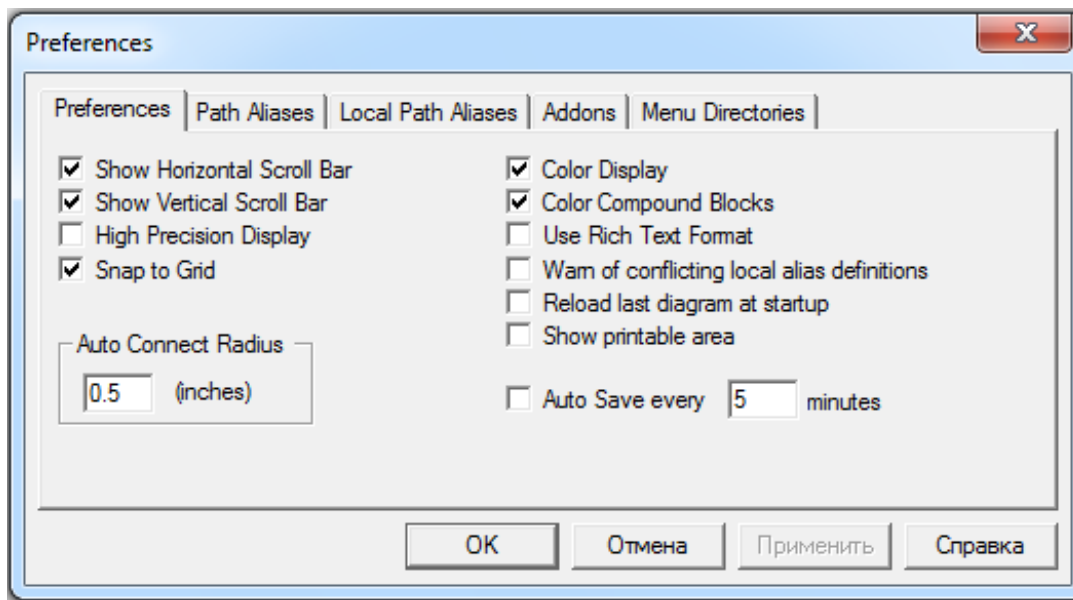


Рисунок П1.4. Окно предустановок

– Color Display, Color Compound Blocks, – включение/выключение цветного режима; черно-белый режим отображения блок-схемы может потребоваться в случаях, если необходимо визуально проконтролировать ее вид перед распечаткой на черно-белом принтере и при использовании монохромного дисплея;

– Use Rich Text Format – использовать расширенный текстовый формат (например, для блока комментариев);

– Warn of conflicting local alias definitions – установка приоритета локальных текстовых переменных (псевдонимов) перед глобальными; этот параметр используется в случае создания сложных проектов, состоящих из нескольких файлов; для этого в VisSim имеется нестандартный механизм назначения или смены рабочих каталогов, заключающийся в замене абсолютных (в формате DOS) путей к рабочим каталогам текстовыми переменными (alias – псевдонимами), т. е. при ссылке на файлы в проекте допустимо указывать не пути, а псевдонимы; при этом можно определить несколько псевдонимов: для каталога растровых BMP-изображений, для внедряемых VSM-файлов, для M-файлов (файлы для определения нелинейных функций), для импортируемых файлов данных; в VisSim имеются глобальные и локальные псевдонимы – первые хранятся в файле vissim.ini и актуальны для

любой блок-схемы, вторые хранятся в файле конкретной блок-схемы и актуальны только для нее; возможна ситуация, когда локальный и глобальный псевдонимы имеют одинаковое имя; в этом случае установка рассматриваемой опции предписывает локальному псевдониму более высокий приоритет;

- Reload last diagram at startup – автоматическая загрузка схемы, сохраненной в предыдущей сессии работы с программой;
- Show printable area – показывать пригодную для печати часть схемы;
- Auto save every – включение режима автосохранения через заданное в строке опции время (по умолчанию 5 мин).

**Path Aliases** – закладка для создания глобальных псевдонимов;

**Local Path Aliases** – закладка для создания локальных псевдонимов. При создании псевдонима пути (глобального или локального) используется формат: псевдоним=путь\_MS\_\_DOS. При этом псевдоним не может содержать запятую (,), точку с запятой (;), двоеточие (:) или знак доллара (\$), который используется специальным образом (см. ниже). Примеры: BmpDir=C:\BITMAPS\PUMP.BMP – этот псевдоним указывает на определенный файл; BmpDir=C:\BITMAPS – псевдоним указывает на директорию; при использовании в блок-схеме его имя и имя файла разделяются обратной наклонной чертой; BmpDir=C:\BITMAPS\ – псевдоним указывает на директорию, при использовании в схеме его имя указывается в круглых скобках: (BmpDir). Использование псевдонима пути возможно в любом блоке, в котором имеется ссылка на файл. При этом псевдониму должен предшествовать символ доллара (\$), с тем чтобы VisSim мог различать, где используется MS\_DOS-путь, а где псевдоним. Для создания в составном блоке ссылки на файл с применением псевдонима используется диалоговое окно блока;

**Addons** – закладка для выбора пути к дополнительным DLL-файлам (кроме установленных при инсталляции программы).

**Menu Directories** – расположение каталогов меню.

**Tool Bar** – создание кнопок пользователя (в VisSim/Comm 7.0 находится в закладке Customize меню Tools); перед выбором команды в меню View

активизируется опция Toolbar. Вызов команды сопровождается появлением диалогового окна (рис. П1.5).

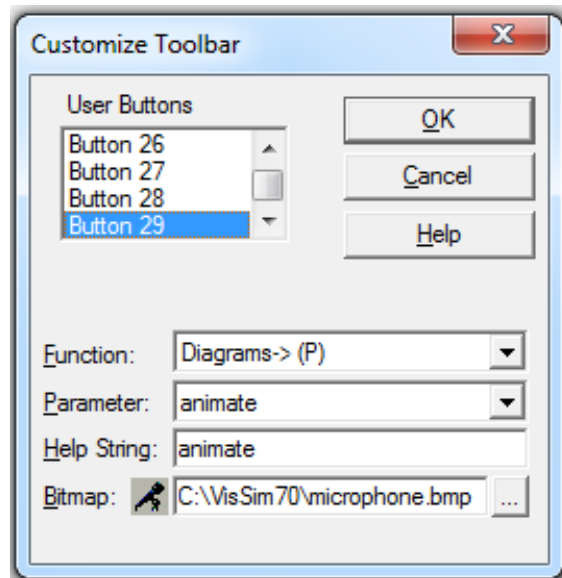


Рисунок П1.5. Окно создания кнопок пользователя

В окне создания кнопок пользователя имеются следующие элементы:

- User Buttons – выбор номера кнопки (0 – 29);
- Function – выбираются необходимые команды VisSim, которые необходимо сгруппировать в требуемом сочетании;
- Parameter – выбираются параметры команды, помеченной в упоминавшемся списке символами (P);
- Help Siring – содержание подсвечиваемой подсказки о названии команды;
- Bitmap – выбор изображения для кнопки; точечный рисунок должен иметь размер 16 пикселей в ширину и 15 пикселей в высоту.

Для удаления пользовательской кнопки, отмеченной в списке User Buttons, в строке Function выбирается первая позиция (none).

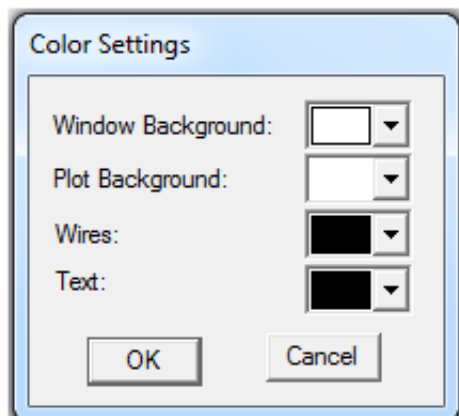
### П1.1.3. Меню View

**Fonts** – команда изменения параметров шрифтов с использованием стандартного окна.

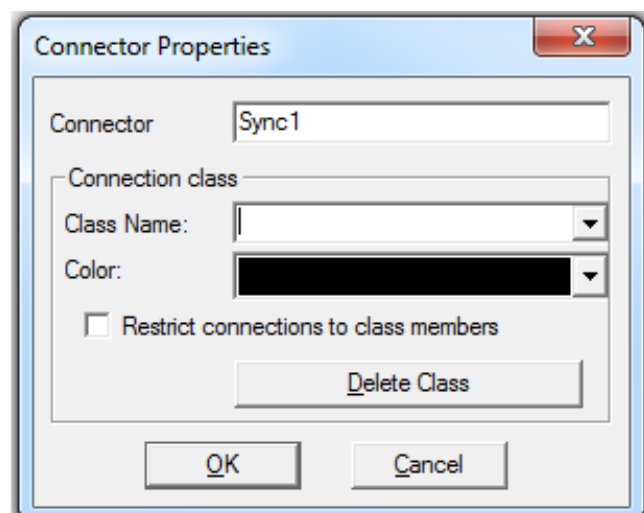
**Colors** – выбор в диалоговом окне (рис. П1.6а) из предлагаемой палитры цветового оформления фона рабочего поля программы (Window Background), дисплея виртуальных приборов (Plot Background), соединительных проводников (Wires) и всех текстовых надписей (Text).

**Block Labels** – показывать краткие характеристики блоков или метки, располагаемые под их значком.

**Connector Labels** – показывать метки выводов блоков, которые оформляются следующим образом; к выводу блока подводится курсор и после того, как он примет вид вертикальной стрелки ↑, производится двойной щелчок левой кнопкой мыши, в результате чего вызывается окно (рис. П1.6б), в которое вводится наименование вывода (в строке Connector), например вывод или ввод синхронизации, класс вывода (Class Name) (в данном случае класс синхронизирующих цепей блок-схемы) и его цвет (Color), что позволяет более оперативно ориентироваться при «чтении» схемы; если необходимо предотвратить случайные подключения между разными классами цепей схемы, то активизируется опция Restrict connection to class members (Ограничить подключения к элементам класса); при необходимости исключения данного вывода из наименованного класса используется кнопка Delete Class.



а)



б)

Рисунок П1.6. Окна выбора цветового оформления (а) и наименования выводов блоков (б)

**Display Mode** (Alt + D) – включение режима «Дисплей оператора», который не позволяет обслуживающему персоналу редактировать параметры схемы модели, являющейся частью реального производственного процесса.

В этом режиме проводники и составные блоки скрываются, а для оператора доступными являются только определенные функции управления (ознакомление с текущими данными, ввод данных по требованию, отдельные регулировки и т. п.). Режим используется также для анимационной визуализации результатов моделирования.

**Data Types** – режим контроля типа данных обрабатываемых сигналов путем раскраски выводов блоков следующим образом: вывод с данными типа double (двойной точности с плавающей точкой) обозначается красным цветом, int (целое число со знаком) – зеленым, unsigned int (целое число без знака) – синим, смесь данных (шинный проводник) – малиновым.

**Presentation Mode** – включение режима презентации, при котором треугольники на входах, определяющие направление распространения сигналов, становятся меньше, а треугольники выходов скрываются, отчего блок-схема становится менее загроможденной и более презентабельной.

**Zoom In, Zoom Out** – увеличение или уменьшение изображения текущей блок-схемы; дублируются кнопками  .

**Control Panel** – включение/выключение контрольной панели в виде небольшого ярлыка, который может быть размещен в любом месте экрана.

**Status Bar** – включение/выключение строки состояния, находящейся в нижней части рабочего окна на рисунке П1.1; содержит информацию о текущей блок-схеме: количество блоков (Blks), временной диапазон (Rng) и конечное время (T) моделирования, метод интегрирования, размер шага (Step) и др. При перемещении курсора мыши по элементам меню, кнопкам, блокам модели в строке состояния отображается их краткое описание и возможности использования специальных клавиш и их комбинаций.

**Tool Bar** – включение/выключение элементов инструментальной панели путем выбора их в диалоговом окне.

### П1.1.4. Меню Simulate

**Go** (F5) – запуск моделирования; дублируется кнопкой , а также одноименной кнопкой панели управления, вызываемой командой View/Control Panel.

**Stop** (F4) – остановка моделирования; дублируется одноименной кнопкой панели управления. При аварийной остановке (при наличии ошибки) автоматически активизируется кнопка .

**Single Step** (F10) – пошаговое моделирование; дублируется одноименной кнопкой панели управления или кнопкой .

**Continue** – продолжить моделирование (после приостановки); дублируется кнопкой Cont панели управления или кнопкой .

**Reset Sim** – сброс данных, содержащихся после моделирования в блоках с памятью (интеграторы, регистры, элементы задержки и др.); дублируется одноименной кнопкой панели управления.

**Simulation Properties** – набор настроек моделирования, содержащихся в пяти закладках окна приведенного на рисунке П1.7.

Закладка **Range** содержит следующие настройки:

- Start/End – начало/конец моделирования;
- Frequency – частота выборок при формировании осциллограмм;
- Run in Real Time – запуск в реальном режиме времени, при котором временные параметры моделирования совпадают с реальным временем (например, с обычной секундой); используется при работе с аппаратными средствами ввода/вывода и дополнительном программном обеспечении;
- Auto Restart – опция непрерывного режима работы программы путем автоматического перезапуска процесса моделирования; используется в случае управления реальными объектами или в режиме обучения нейронных сетей;
- Retain States – сохранение в конце цикла моделирования состояния блоков с памятью; используется в режиме Auto Restart.



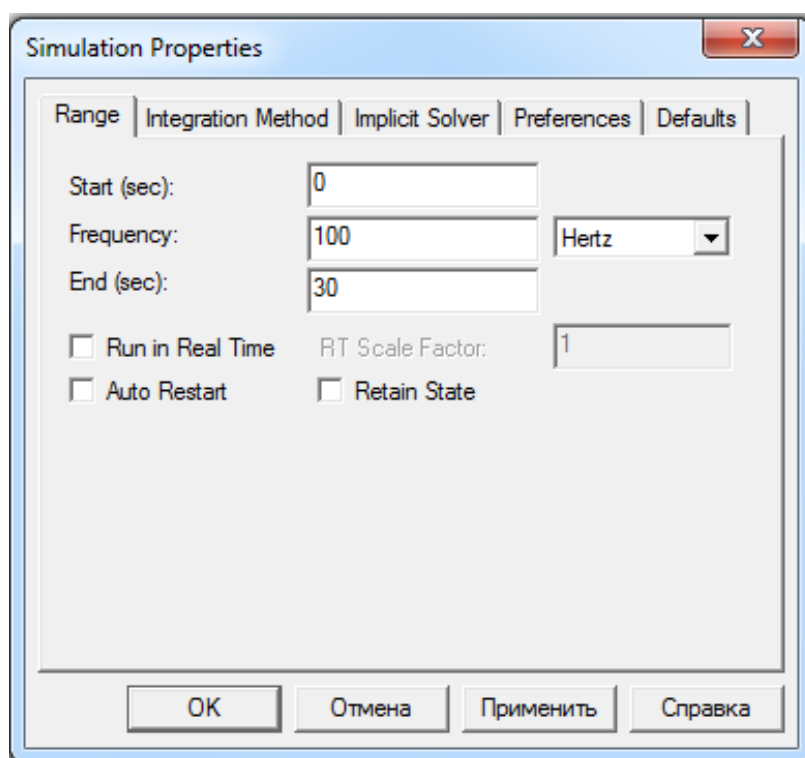


Рисунок П1.7. Окно настроек моделирования

Закладка **Integration Method** содержит настройки, связанные с выбором методов численного интегрирования, которые могут быть разделены на простые (с постоянным шагом интегрирования) и адаптивные (с переменным шагом). Первые обеспечивают высокую скорость интегрирования, однако отличаются невысокой точностью и нестабильностью процесса моделирования.

Адаптивные алгоритмы отличаются более высокой точностью расчета при меньшем быстродействии. При их использовании выбираются следующие параметры:

- Min Step Size – минимальный размер шага моделирования; определяет промежуток времени, по истечении которого обновляется значение интеграла входной функции, отображаемое с помощью соответствующего регистрирующего блока;
- Max Truncation Error, Max Iteration Count – максимальная ошибка и максимальное количество итераций для ее достижения.

С помощью рассматриваемой закладки выбираются следующие методы численного интегрирования:

– Euler – одношаговый метод Эйлера, при котором оценка значения вычисленного интеграла выполняется один раз за шаг моделирования; при оптимальном выборе шага метод обеспечивает самую высокую скорость моделирования, однако критичен к выбору шага и нередко приводит к расхождению решения;

– Trapezoidal – метод трапеций; имеет на порядок меньшую погрешность по сравнению с методом Эйлера; используется для решения многих задач моделирования, если результаты представляются гладкими и не слишком быстро изменяющимися кривыми; оценка значений интеграла выполняется дважды за шаг моделирования;

– Runge Kutta 2d order – метод Рунге-Кутты второго порядка; для оценки значения интеграла в конечной точке шага рассчитывается производная в его середине; по точности вычислений эквивалентен методу трапеций;

– Runge Kutta 4th order – метод Рунге-Кутты четвертого порядка; имеет точность четвертого порядка; для оценки значения интеграла рассчитываются производные четыре раза на каждом шаге: в начальной точке, дважды в середине и в конце шага;

– Adaptive Runge Kutta 5th order – адаптивный метод Рунге-Кутты пятого порядка (с точностью пятого порядка); при увеличении приращений входных координат алгоритм автоматически уменьшает размер шага;

– Adaptive Bulirsh-Stoer – адаптивный метод Булирша-Стоера; для оценки значения интеграла в конечной точке шага используется полиномиальная экстраполяция на основе серии предыдущих расчетов; метод обладает малой погрешностью для гладких функций, когда координаты меняются с постоянной скоростью, ускорением или постоянным приращением ускорения; этот метод и метод Рунге-Кутты пятого порядка могут быть полезными для обеспечения устойчивой работы модели, например в случае, если сиг-пал на входе интегратора содержит разрывы;

– Backward Euler (Stiff) – обратный метод Эйлера; наиболее эффективен при моделировании систем с большой разницей частот собственных колебаний

(так называемых жестких моделей); для таких систем другие алгоритмы требовали бы установки существенно меньшего размера шага для стабильного моделирования. При использовании этого метода шаг  $\Delta t$  целесообразно выбирать из условия  $\Delta t \approx (0,1 - 0,2)/\Delta f$ , где  $\Delta f$  – разность собственных частот;

- Adaptive Adams-Moulton – адаптивный многошаговый метод Адамса-Мултона;

- Adaptive BDF – адаптивный многошаговый метод с применением дифференцирования назад.

Закладка **Implicit Solver** служит для выбора параметров моделирования при решении неявных и имплекативных, например алгебраических, уравнений. К этим параметрам относятся:

- None – отказ от применения решателя имплекативных уравнений;
- Newton-Raphson – применение решателя на основе метода Ньютона-Рафсона;

- User Defined – применение заданного пользователем решателя;
- Suppress Convergence Warnings – запрет на вывод сообщений о сходимости;

- Max Iteration Count – максимальное число итераций при заданных значениях погрешности расчета (Error Tolerance), релаксации (Relaxation) и возмущения (Perturbation).

Закладка **Preferences** служит для выбора следующих параметров моделирования:

- Checkpoint State – опция сохранения в файле состояния исследуемой модели (текущие значения всех координат, результаты визуализации, текущее время моделирования и др.) на момент приостановки процесса моделирования и возможности его возобновления при очередном сеансе работы с программой; режим полезен в случае продолжительного по времени моделирования; опция должна быть активизирована перед запуском моделирования;

- Notify Simulation End – разрешение на вывод сообщения о приостановке моделирования;

- Warn Nonintegral Clock – разрешение на вывод сообщения о несоответствии (некратности) периода синхронизации шагу моделирования; опция включена по умолчанию, поскольку в противном случае такая ошибка моделирования может быть не обнаружена;
- Warn Nonintegral Delay – разрешение на вывод сообщения о несоответствии времени задержки соответствующего блока (например, линии задержки) шагу моделирования; по указанным выше причинам эта опция включена по умолчанию;
- Checkpoint During Sim – опция определения контрольных точек при симуляции;
- Check Connections – вывод сообщения при пуске моделирования о неподключенных входах блоков, которые при этом помечаются красным цветом; в окне сообщения можно выбрать кнопку остановки моделирования, проигнорировать первый найденный неподключенный вход и продолжить проверку блок-схемы или все-таки провести моделирование;
- Warn Numeric Overflow – вывод сообщения о числовом переполнении в случае, если при преобразовании старших типов данных в младшие, (например, double к int) происходит переполнение;
- Propagate Integer Types – вывод сообщения о числовом переполнении при преобразовании целочисленных типов данных языка Си (от младших типов к старшим), используемого в VisSim;
- Raise Real-time Priority – приоритет задач реального времени; опция активизируется в случае управления реальным объектом, когда этому процессу присваивается более высокий приоритет для надежного осуществления процессов выборки и записи в каналы специализированных плат ввода/вывода с запретом прерываний от других процессов, которые выполняются одновременно;
- Optimize matrix usage – оптимизация использования матриц;
- Startup Script File – поиск скриптового файла;

– **Random Seed** – начальное число (0 – 65535) для генераторов псевдослучайных последовательностей; каждое новое начальное число изменяет характер последовательности. Если опция **Auto Restart** (закладка **Range**) активизирована, то программа генерирует новую последовательность автоматически при каждом запуске моделирования;

– **Frequency Units** – выбор единиц измерения частоты (Гц или рад/с) во всех блоках с таким параметром.

Закладка **Defaults** определяет настройки параметров моделирования умолчанию. К таким параметрам относятся упоминавшиеся ранее (закладки **Range** и **Integration Method**) метод интегрирования, начальное и конечное время моделирования, шаг интегрирования и максимальное число отображаемых точек для графопостроителя (**Max Plotted Points**).

**Optimization Properties** – выбор параметров оптимизации (рис. П1.8). Данное окно позволяет устанавливать следующие параметры оптимизации:

– **Powell** – безградиентный метод Поувелла, в котором вычисления производных выполняются по упрощенным разностным формулам, что обеспечивает повышенную скорость оптимизации;

– **Polak-Rabiere** – градиентный метод Полака–Райбера для поиска минимума функций с квадратичной или близкой к ней зависимостью в окрестности точки минимума;

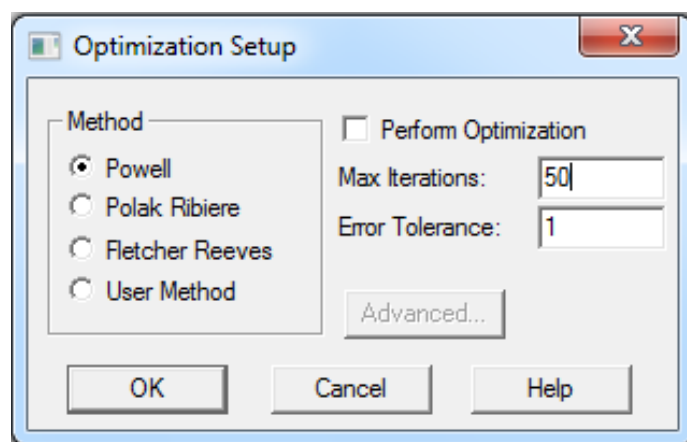


Рисунок П1.8. Окно выбора параметров оптимизации

- Fletcher Reeves – градиентный метод Флетчера–Ривеса, отличающийся лучшей сходимостью, чем метод Полака–Райбера, но обладающий несколько меньшей скоростью поиска;
- User Method – оптимизация методом пользователя;
- Perform Optimization – признак использования при моделировании средств оптимизации;
- Max Iterations – максимальное число итераций;
- Error Tolerance – пороговое значение погрешности, определяемой разностью значений оптимизируемой зависимости на двух последних итерациях; увеличение этого параметра способствует уменьшению количества итераций и улучшению сходимости. Получаемое при этом грубое решение может быть использовано в качестве исходного для более точного поиска минимума;
- Advanced – кнопка для будущего применения.

Оптимизация модели заключается в таком подборе ее параметров, при котором обеспечивается минимум некоторой целевой функции, например для поиска корней нелинейных уравнений или оптимизации системы управления по заданному параметру качества. В рассматриваемой программе задача оптимизации решается с использованием специальных блоков с изменяемыми в процессе моделирования параметрами.

**Snap States** – фиксация текущего состояния модели в заданном промежутке времени с запоминанием мгновенных значений всех ее координат на момент остановки моделирования; если в таком состоянии схема модели будет сохранена на диске, то будут сохранены и начальные условия, перезаписанные в результате выполнения снимка текущего положения модели в пространстве состояний.

**Reset Properties** – сброс начальных условий в блоках модели, обладающих эффектом памяти; в VisSim/Comm 7.0 эта команда переименована в Reset States.

### П1.1.5. Меню Blocks, Comm и Wireless

Эти меню содержат библиотеки блоков, рассматриваемые в П1.3, за исключением пункта Comm Module Random Seed – установка случайного начального числа с помощью модуля связи, реализуемого DLL-файлом. Параметры команды (рис. П1.9):

- Enable DLL Random Seed – разрешение на использование указанного в строке Seed значения начального числа перед каждым пуском моделирования; в случае деактивации этой опции начальное число для генератора случайных последовательностей будет определяться состоянием генератора на момент окончания предыдущего сеанса моделирования;
- Reset Seed on Auto Restart – опция использования установленного значения Seed в режиме авторестарта.

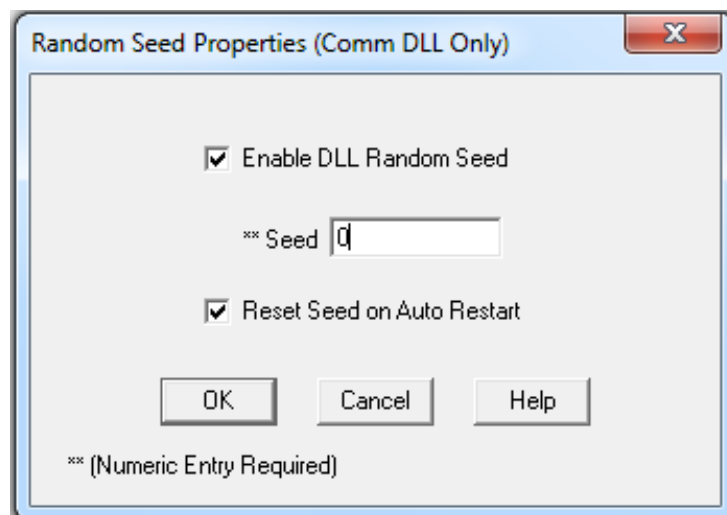


Рисунок П1.9. Окно команды Comm/Comm Module Random Seed

### П1.1.6. Меню Tools

**Code Gen** – параметры генераторов по умолчанию.

**Compile Matlab Function** – компиляция функций MATLAB; эта и следующие две команды активны в случае, если при инсталляции Vissim/Comm указан путь к файлу matlab.exe, т. е. при наличии на компьютере MATLAB.

**Model Wizard** – создание модели с помощью помощника.

**Import** – импорт данных из MATLAB в виде таблиц (Look-Up Table) и блок-схем Simulink (Simulink Diagram).

**Insert Mathcad Object** – вставка Mathcad-объекта в модель VisSim:

- New – вставка нового Mathcad-объекта; команда сопровождается вызовом программы Mathcad, если она установлена на компьютере;
- From File – вставка Mathcad-объекта из файла с использованием стандартного окна загрузки.

**Customize** – параметры пользовательских настроек:

- Menu – параметры вставки пользовательского меню;
- User Tool Bar – создание пользовательской панели инструментов (рис. П1.5), которая содержит кнопки для наиболее часто используемых команд; существует возможность выбора изображения для кнопок;
- Tool Bar – параметры отображения панелей инструментов.

**Tutorial Creator** – создание обучающих подсказок:

- Tutorial – создание блока обучающих подсказок;
- Design Mode – форматирование окна текстовой подсказки;
- New Message – создание текстовой подсказки;
- Start New Part – старт новой части обучающих подсказок;
- Start New Module – старт нового модуля обучающих подсказок;
- Organize – организация последовательности обучающих подсказок;
- Preferences – коррекция размера, вида, ориентации, фона, шрифта текстовой подсказки;
- About vsm Tutor – сведения о версии VisSim Tutor Addon.

**VML Workspace** – окно для программирования на языке высокого уровня VML (VisSim Matrix Language), для манипуляции матрицами и векторами.

### П1.1.7. Меню Analyze

Это меню считается расширением VisSim и активизируется в том случае, если сначала устанавливается базовая часть VisSim, а затем VisSim/Comm. Это



меню к задачам, решаемым VisSim/Comm, непосредственного отношения не имеет. Поэтому ограничимся самыми краткими сведениями по его командам, перечисленным ниже.

**Linearize** – вывод окна с данными линеаризованной модели.

**Select Input/Output Point** – селекция входов и выходов модели.

**Transfer Function Info** – вывод окна с информацией о передаточной функции подсистемы.

**Compensator Design** – конструирование компенсатора (относится к анализу и синтезу регуляторов).

**Select Compensator Block** – выделение блока компенсатора.

**Root Locus** – вывод диаграммы нулей и полюсов.

**Root Locus Options** – установка опции нулей и полюсов.

**Nyquist Responce** – построение диаграммы Найквиста.

**Frequence Responce** – постройка АЧХ и ФЧХ.

**Frequence Range** – установка пределов шкалы частот.

**Preferences** – вывод окна установок меню Analyze:

- Digits of Display Precision – установка числа знаков отображения чисел;
- Frequence Unit – выбор единицы измерения, частоты (рад/с или Гц);
- Phase – выбор единицы измерения фазы (в радианах или градусах);
- Magnitude – выбор единицы измерения отношения сигналов на выходе

и входе;

– Zero Theresold – установка пороговых значений числовых данных, которые принимаются за нулевые.

**Reuse Plot** – построение нескольких графиков.

**About** – вывод окна с данными о версии расширения Analyze.

### П1.1.8. Меню Window и Help

Меню Window содержит стандартные команды работы с окнами, включая перечень открытых в данный момент схем (диаграмм) моделей.

Меню Help состоит из следующих пунктов.

**Contents** – вывод окна указателя справочной системы.

**Search** – вывод окна указателя для поиска.

**Wiring** – справка по межблочным соединениям.

**Simulating** – справка по методам моделирования.

**License Manager** – менеджер лицензии.

**About VisSim** – сведения о программе VisSim.

**About VisSim/Comm Module** – сведения о модуле Comm.

**Comm Module Help** – вызов руководства пользователя модуля Comm.

**Comm Module Updates** – сведения об обновлениях модуля Comm.

**Purchase Information** – информация о покупке программы.

**About Turbo Codes Module** – информация о модуле турбокодеров.

**About Wireless Module** – информация о модуле беспроводных компонентов.

**Wireless Module Help** – справка об использовании модуля беспроводных компонентов.

### П1.1.9. Технология подготовки схем моделей

Подготовка схем (диаграмм) моделей выполняется в следующем порядке:

1. На листке бумаги составляется схема модели с примерным расположением блоков из специализированных библиотек VisSim.

2. В соответствии с составленной схемой на рабочее поле программы из библиотек переносятся нужные блоки. Для этого щелчком левой кнопки мыши по названию соответствующего блока в меню Blocks, Comm, Wireless или мнемик-ножке на рисунке П1.1 он «прицепляется» к курсору мыши в виде штрихпунктирной прямоугольной рамки и переносится на рабочее поле, после чего производится вторичный щелчок левой копкой и рамка превращается в выбранный блок VisSim. Затем на его значок устанавливается курсор мыши, и, когда он примет форму крестообразных стрелок, нажимается левая кнопка мыши, после чего выполняется перемещение с удержанием кнопки.

Аналогичным образом перемещаются метки, например с комментариями, и составные блоки (субблоки).

2.1. Вставка блока или группы блоков в текущую схему может осуществляться методом копирования. Если «размножается» один блок текущей схемы, то он сначала выделяется левым щелчком мыши по его значку при нажатой клавише Shift, после чего командой Edit/Copy (Ctrl + C) производится копирование в буфер обмена. Затем выполняется команда Edit/Paste (Ctrl + V), в результате чего скопированный блок в виде штрихпунктирного прямоугольника курсором мыши переносится в нужное место с последующим нажатием ее левой кнопки.

Копирование одного блока возможно также без предварительного выделения щелчком мыши по его значку при нажатых клавишах Ctrl и Shift. При копировании группы блоков целесообразно использовать другой способ их выделения, при котором курсор мыши ставится в левый верхний угол выделяемой области и при нажатой левой кнопке протягивается по ее диагонали, после чего выполняются вышеперечисленные команды. Аналогичным образом в схему вставляются группы простых (библиотечных) и составных блоков как из текущей схемы, так и из других схем. В последнем случае схему-донор целесообразно предварительно открыть и затем вызывать ее из меню Window.

2.2. При перемещении или копировании блоков значения их параметров остаются неизменными; при копировании внутренние соединения в группе блоков сохраняются, а внешние, идущие к другим блокам, – отсекаются. Удаление блоков из схемы производится клавишей Del или командой вырезания Cut (Ctrl + X) после их выделения; при этом удаляются и подключенные к ним проводники.

2.3. Для большинства блоков в их диалоговых окнах в поле строки Label может быть определена текстовая метка, например с его названием или функциональным назначением. Такие метки появляются под значком блока, если активизирована команда View/Block Labels (при отсутствии метки в этом

месте индицируются параметры блока в сокращенном виде). Для блоков без параметра Label его метка может быть оформлена с помощью блока Label. Этот же блок может быть использован для оформления различных поясняющих надписей, заголовков и т. п.

2.4. Для добавления и удаления выводов с правой или левой стороны блока используются соответственно команды Edit/Add Connectors и Edit/Remove Connectors, которые представлены кнопками в инструментальной панели.

2.5. Если блок в целях удобства соединения с другими блоками необходимо развернуть на  $180^\circ$ , то он выделяется, после чего нажимаются клавиши Ctrl + ←.

2.6. Установка параметров каждого блока производится в диалоговом окне, вызываемом правым щелчком мыши по его значку (иконке) или, что менее удобно, командой Edit/Block Properties, при выборе которой курсор приобретает форму гаечного ключа, которым и вызывается окно. Для некоторых блоков (button, compound, DDE, DDEsend, DDEreceive, embed, label, userFunction и variable) быстрый доступ к их диалоговым окнам осуществляется правым щелчком мыши при удержании клавиши Ctrl.

При вводе числовых данных значения  $\geq 10^6$  или  $\leq 10^{-6}$  отображаются в показательной форме, например число 6000000,0 записывается в виде 6Е6 или 6Е6. Можно также вводить числовые данные в качестве параметров блоков в виде математических выражений с использованием арифметических операторов «+», «-», «\*», «/», константы «pi», математических функций acos, asin, atan2, cos, cosh, exp, fabs, log, log10, pow, sin, sinh, sqrt, tan, tanh с учетом обычных правил их приоритетов в расчетах.

Кроме того, среда VisSim способна распознать выражения, написанные на языке Си, результатом которых являются числовые данные. Например, если ввести row(2,3) вместо коэффициента усиления в блоке gain, то VisSim его вычислит (8) и примет к исполнению. Числовые расчеты в VisSim при моделировании всегда проводятся с точностью до 15 значащих десятичных

цифр (имеется в виду мантисса). Это значение можно уменьшить с 15 до 6 значащих десятичных цифр, используя команду Edit/Preference/High Precision Display.

2.7. Поиск и замена блоков при редактировании схемы модели производится командой Edit/Find (Alt + F3) и Edit/Replace.

3. При соединении блоков используются два типа проводников: одиночный (FlexWire – простой) для передачи одиночного сигнала и шинный («многожильный», более толстый) – для передачи совокупности сигналов при выполнении векторных или матричных операций. Шинный проводник также используется для устранения загромождения схемы многочисленными проводниками. Собрать одиночные проводники в шину и вывести их из нее можно при помощи парных блоков scalarToVec и vecToScalar соответственно.

3.1. Для прокладки соединительного проводника между входом одного и выходом другого блока курсор мыши подводится к одному из этих выводов, и, когда он примет форму вертикальной стрелки ↑, нажимается левая кнопка, после чего курсор с проводником зеленого цвета подводится к выводу другого блока и кнопка мыши отпускается. При этом проводник меняет свой цвет на установленный по умолчанию черный. При соединении выхода одного блока со входами нескольких других проводники прокладываются, начиная со входов каждого и заканчивая выходом первого.

Удаление проводника производится аналогичным образом, т. е. начиная со входа, когда курсор приобретает форму стрелки ↑, и заканчивая выходом. Заметим, что, если левую кнопку мыши отпустить не у выхода блока, то проводник может «перескочить» на свободный вход другого близлежащего блока. Возможность такого самопроизвольного соединения определяется параметром Auto Connect Radius, устанавливающим расстояние в дюймах между курсором мыши с проводником и выводом блока, при котором подключение к нему заканчивается автоматически.

3.2. Кроме шинных проводников с применением блоков преобразования скалярных величин в вектор ScalarToVec и вектора в набор скалярных величин

VecToScalar, для упрощения схем моделей используется блок присвоения имени переменной Variable, позволяющий существенно уменьшить количество проводников схемы модели, если какой-либо сигнал используется многократно. В этом случае к выходу источника такого сигнала достаточно подключить блок Variable, указав в его диалоговом окне имя переменной, после чего такой блок можно использовать в качестве источника сигнала в любой части схемы модели, включая подсхемы (субблоки). Такой механизм подобен использованию символа заземления в электрических принципиальных схемах, повышающих ее наглядность за счет сокращения проводов.

3.3. Для фиксации на схеме положения проводников используется блок WirePositioner, направление выводов которого может меняться на 180°. Изменяя положение этого блока, простой или шинный проводник можно зафиксировать в нужном положении.

3.4. Для определения параметров выводов блоков и подсоединенных к нему проводников двойным щелчком мыши (при курсоре типа ↑) по выбранному выводу вызывается диалоговое окно, в котором выбираются:

- Connector – название соединения;
- Class Name – название класса соединения; например, название Sync означает, что соединение относится к классу цепей синхронизации;
- Color – цвет проводников данного класса; выбирается из предлагаемой палитры; если класс назначен определенному выводу, то, куда бы он ни был подключен, проводники будут окрашены в цвет класса;
- Restrict Connections to class member – опция ограничения подключений к элементам класса;
- Delete class – кнопка стирания присвоения статуса класса.

3.5. Изменение цвета соединительных проводников схемы осуществляется командой View/Colors, в диалоговом окне которой выбирается цвет из предлагаемой палитры.

3.6. Скрытие проводников схемы производится выбором режима «Дисплей оператора», активизируемого командой View/Display Mode.

4. Создание больших иерархических схем достигается объединением некоторой совокупности блоков в одном составном с различным уровнем вложенности. На верхнем уровне составные блоки могут отражать главные компоненты системы и их связи, а на нижнем – логику работы каждого компонента и его частей. Составные блоки (субблоки) позволяют применить модульный подход при разработке большой модели, т. е. проектирование и проверка функционально независимых компонентов может быть выполнена независимо и в разных рабочих файлах.

4.1. Для создания составного блока часть схемы, которую необходимо включить в состав такого блока, выделяется по правилу прямоугольника, после чего вызывается команда *Edit/Create Compound Block*, в диалоговом окне которой выбираются соответствующие параметры составного блока. При этом составной блок снабжается выводами для каждой связи с внешними блоками, а также для каждого неподключенного вывода блоков, оказавшихся внутри составного.

В свою очередь, такой составной блок в сочетании с простыми (библиотечными) блоками и другими составными может быть объединен в составной блок (субблок) второго уровня и т. д. Вызов диалогового окна субблока (например, в целях просмотра его параметров или редактирования) производится щелчком правой кнопкой мыши по его значку (иконке) при нажатой клавише *Ctrl*.

4.2. Все операции работы с простыми блоками практически полностью применимы к составным. Для установки дополнительных параметров составного блока используется команда *Edit/Block properties*.

4.3. Для демонтажа (декапсуляции) выделенного составного блока применяется команда *Edit/Dissolve Compound Block*, при выборе которой курсор приобретает многострелочную форму в виде захвата, которым и осуществляется демонтаж выбранного блока с сохранением всех связей с другими блоками и субблоками.

## П1.2. Блоки контроля и управления

В этой главе рассматриваются модели устройств из библиотек Blocks/Signal Consumer и Blocks/Signal Producer, часть которых представлена на приведенной ниже линейке мнемокнопок.

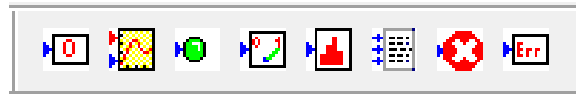


Рисунок П1.10. Внешний вид линейки мнемокнопок

### П1.2.1. Многофункциональный графопостроитель (Plot)

Этот прибор является наиболее универсальным и наиболее часто используемым. Он имеет 8 каналов с различными способами разверток. Одна из возможных и наиболее простых схем применения прибора в качестве трехканального осциллографа показана на рисунке П1.11, где в качестве источников входных сигналов используются следующие блоки: функциональный генератор Frequency Waveform Generator 6 в режиме треугольных импульсов (осциллограмма 1); источник случайных символов Random Symbol 7 (осциллограмма 2); источник качающейся частоты Sweep 8 (осциллограмма 3).

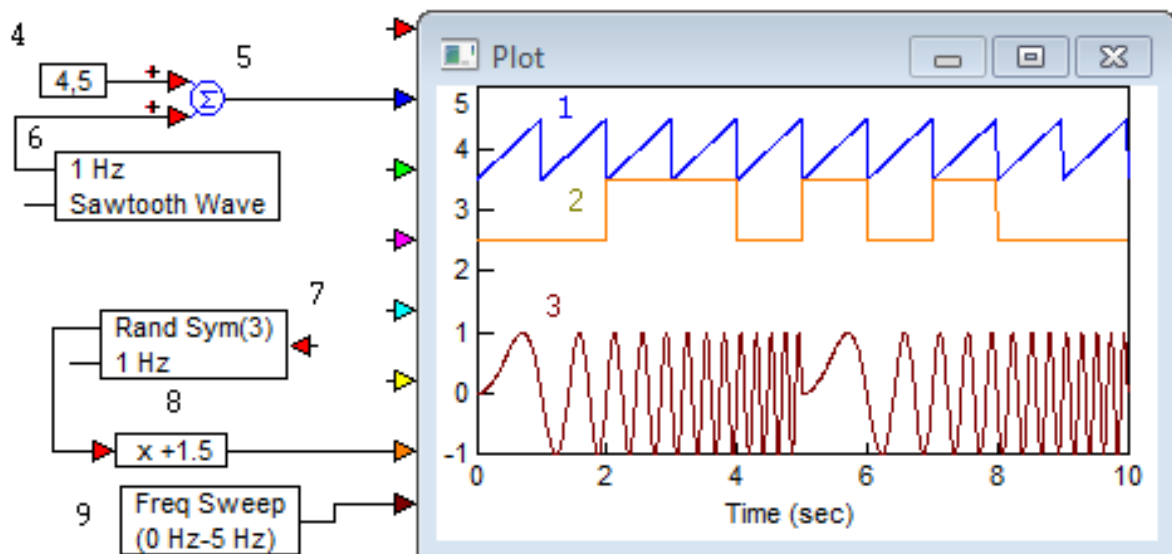


Рисунок П1.11. Схема включения графопостроителя (плоттера)



Для смещения осциллограмм по вертикали может использоваться полиномиальный источник Polynomial 8 (разд. 3.14) или сумматор 2 (разд. 3.3) с источником смещения 4 типа Constant (разд. 3.1).

Окно настроек графопостроителя содержит пять закладок следующего содержания.

**Options** – закладка выбора режимов работы плоттера; она содержит следующие настройки (рис. П1.12):

– Fixed Bounds – фиксация масштаба по оси Y, выбранного на закладке Axis; при деактивации опции включается режим автомасштабирования;

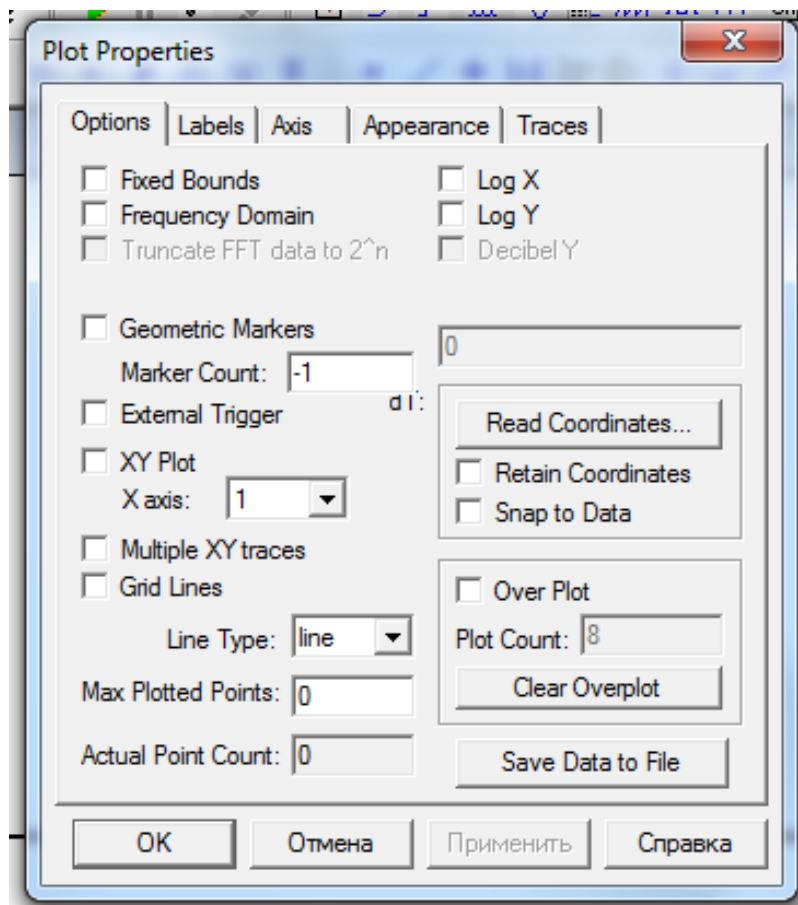


Рисунок П1.12. Окно настроек графопостроителя, закладка Options (а) и Labels (б)

– Frequency Domain – опция активизации процесса выполнения операции быстрого преобразования Фурье (БПФ) с отображением осциллограммы плотности спектра мощности сигнала; точность спектрального анализа зависит от количества зарегистрированных точек, которое должно быть кратно степеням двойки (8, 16, 32, 64, 128, ...), иначе спектр будет зашумлен; точность

спектрального анализа может быть повышена за счет вариации времени и шага моделирования;

- Truncate FFT data to  $2^n$  – БПФ по  $2^n$  точкам; при активной опции (только в режиме Frequency Domain) последовательность накопленных выборок усекается до ближайшей степени двойки. Если опция не активна, то последовательность данных дополняется нулями до количества, кратного ближайшей степени двойки;

- Over Plot – включение режима сохранения указанного в строке Plot Count количества последних точек (следов) осциллограмм при повторном моделировании; позволяет отслеживать воздействие вариации параметров на модель;

- Clear Overplot – кнопка очистки (удаления) всех точек запомненных осциллограмм;

- Geometric Markers – включение режима отображения осциллограмм геометрическими маркерами, форма которых выбирается на закладке Traces;

- Marker Count – количество маркеров на осциллограмму; при нулевом значении маркером будет оформляться каждая выборка;

- External Trigger – включение режима внешней синхронизация, при котором в левом верхнем углу прибора с внешней стороны формируется дополнительный вывод круглой формы, к которому подключается источник синхросигналов с логическими уровнями 0 или 1 (активный – 1 уровень);

- XY Plot and X Axis – режим XY-плоттера, при котором один из входных сигналов используется для управления горизонтальной разверткой;

- Multiple XY Traces – активизация опции позволяет задействовать до четырех независимых XY-плоттеров; при неактивной опции развертка по горизонтали осуществляется одним входным сигналом для семи каналов вертикального отклонения;

- Line Type – выбор для отображения осциллограмм типа линии: в виде линии (line), точек (point) или ступенек (discrete);

- Max Plotted Points – максимальное число регистрируемых и отображаемых на экране выборок (до 250 млн);
- Actual Point Count – действительное количество выборок, использованных при формировании осциллограммы;
- Log X, Log Y – переключения между линейным и логарифмическим масштабами представления данных по осям X и Y;
- Decibel Y – активизации этой опции приводит к оформлению меток по оси Y в децибелах ( $20\text{Lg}(Y)$ ); значение в логарифмическом масштабе 20 дБ соответствует изменению в линейном масштабе в 10 раз;
- Grid Lines – включение/выключение координатной сетки, интервал между линиями которой устанавливается автоматически в зависимости от размера графика (осциллограммы);
- Read Coordinates – кнопка включения измерителя координат графика в виде зеленого перекрестия, позиционируемого мышью или клавишами курсор клавиатуры с одновременным отображением текущих координат в нижней части дисплея; при нажатии левой кнопки мыши или клавиши ENTER положение перекрестия фиксируется; повторное нажатие левой кнопки мыши приводит к обновлению дисплея и стиранию перекрестия;
- Snap to Data – опция режима повышенной точности отсчета координат; в этом режиме перекрестие программно фиксируется на ближайшей точке дисплея;
- Retain Coordinates – эта опция позволяет запомнить предыдущее положение перекрестия и определить разницу между координатами двух точек;
- Save Data To File – кнопка вызова диалогового окна для записи полученных данных в файл.

**Labels** – закладка текстового оформления графиков и осциллограмм (рис. П1.13) позволяет ввести следующие данные:

- Title – в этой строке вводится название графопостроителя по назначению (он располагается в шапке прибора);

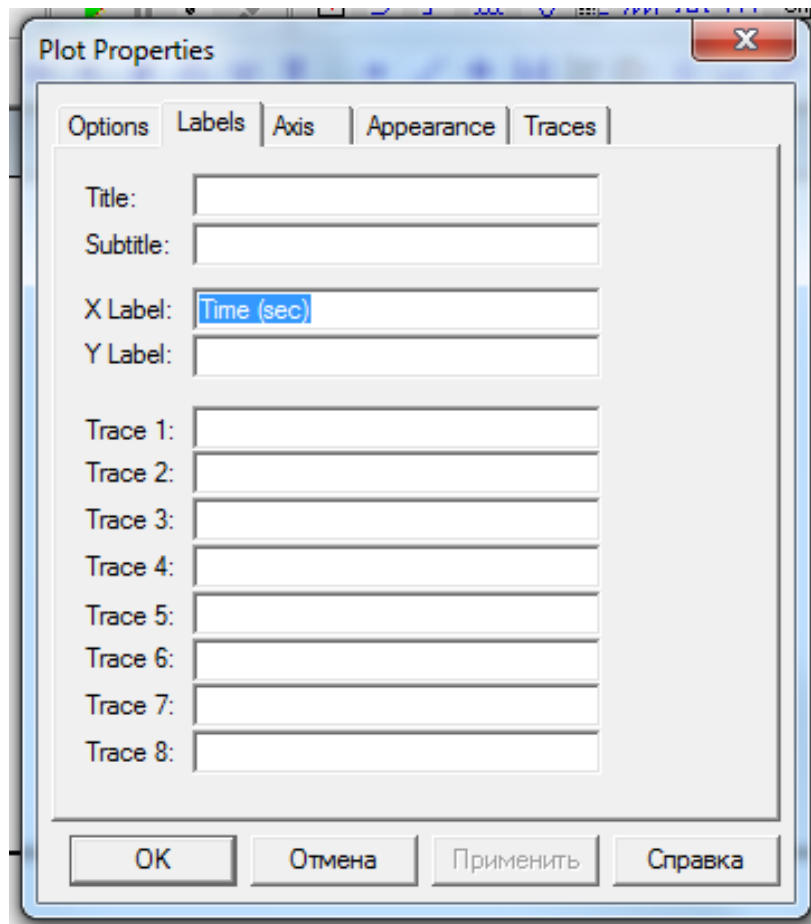


Рисунок П1.13. Окно настроек графопостроителя, закладка Labels

- Subtitle – подзаголовок, например название семейства осциллограмм, над которыми и располагается, занимая верхнюю часть дисплея; обе строки ввода позволяют ввести до 80 алфавитно-цифровых символов. Заголовок появляется в строке названия окна осциллографа; подзаголовок отображается в верхней области дисплея;

- X Label, Y Label – надписи осей X и Y (например, время и амплитуда);

- Trace 1, Trace 2, ... – названия осциллограмм соответствующих каналов; надписи располагаются в верхней части дисплея непосредственно под заголовком или подзаголовком, если такой имеется; надписи следуют с небольшим смещением вправо (елочкой) сверху вниз в соответствии с номером канала и его цветом.

Каждая из перечисленных строк позволяет ввести до 80 алфавитно-цифровых символов и усекается многоточием, если текст не помещается в выбранном размере дисплея.

**Axis** – закладка для выбора следующих параметров (рис. П1.14):

– Y Upper Bound, Y Lower Bound, X Upper Bound, X Lower Bound – верхние и нижние значения размеров осей Y и X; введенные значения фиксируются, если на закладке Options активизирована опция Fixed Bounds;

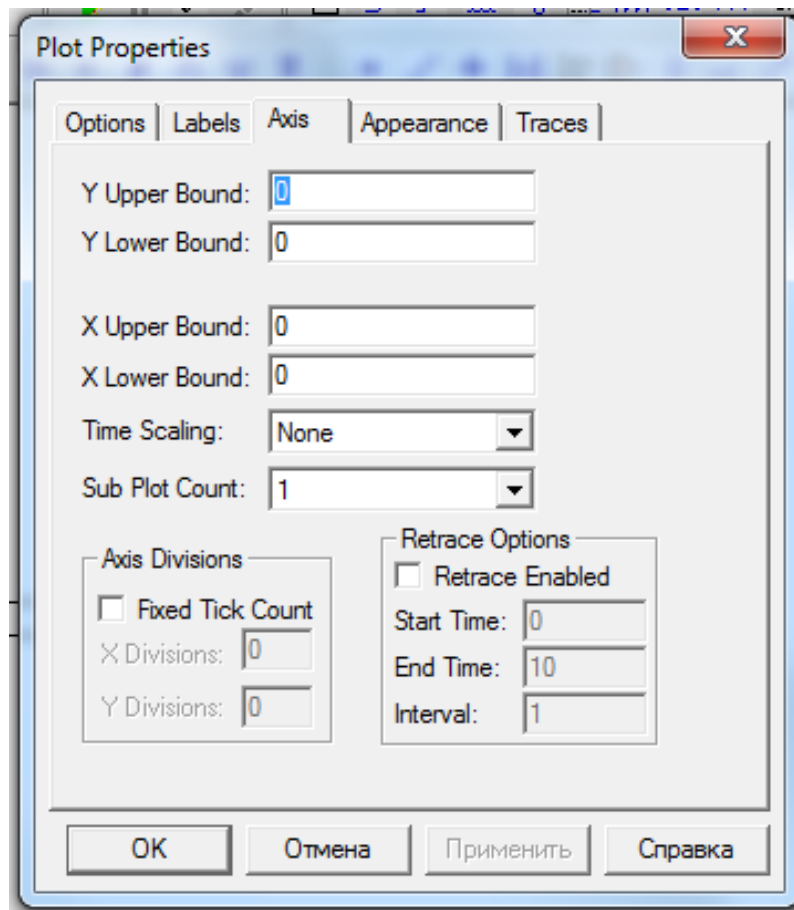


Рисунок П1.14. Окно настроек графопостроителя, закладка Axis

– Time Scaling – выбор из списка единицы времени: от микросекунды до года;

– Axis Divisions – установка размеров сетки; при активизация опции Fixed Tick Count режим автомасштабирования сетки отключается; при этом ее размеры устанавливаются в поле строк X Divisions и Y Divisions (количество отрезков, на которые сетка поделит соответствующие оси); если в поле вводится 0, то режим автомасштабирования сетки не отключается;

– Retrace Options – режим повторной развертки; при активизация опции Retrace Enabled (Обратный ход) реализуется классический для осциллографов

режим работы, при котором его луч, начиная с момента Start Time, пробегает дисплей и в момент End Time быстро возвращается в исходное нулевое положение; процесс повторяется через промежуток времени, задаваемый в поле Interval.

**Traces** – закладка для установки цветового оформления осциллограмм и графиков для каждого из 8 каналов в режиме использования линий (колонок Color) и линий с маркерами в виде разнообразных геометрических фигур (колонок Marker) (рис. П1.15). Выбор производится из предлагаемого набора палитры цветов или маркеров после нажатия треугольной кнопки для соответствующего канала.

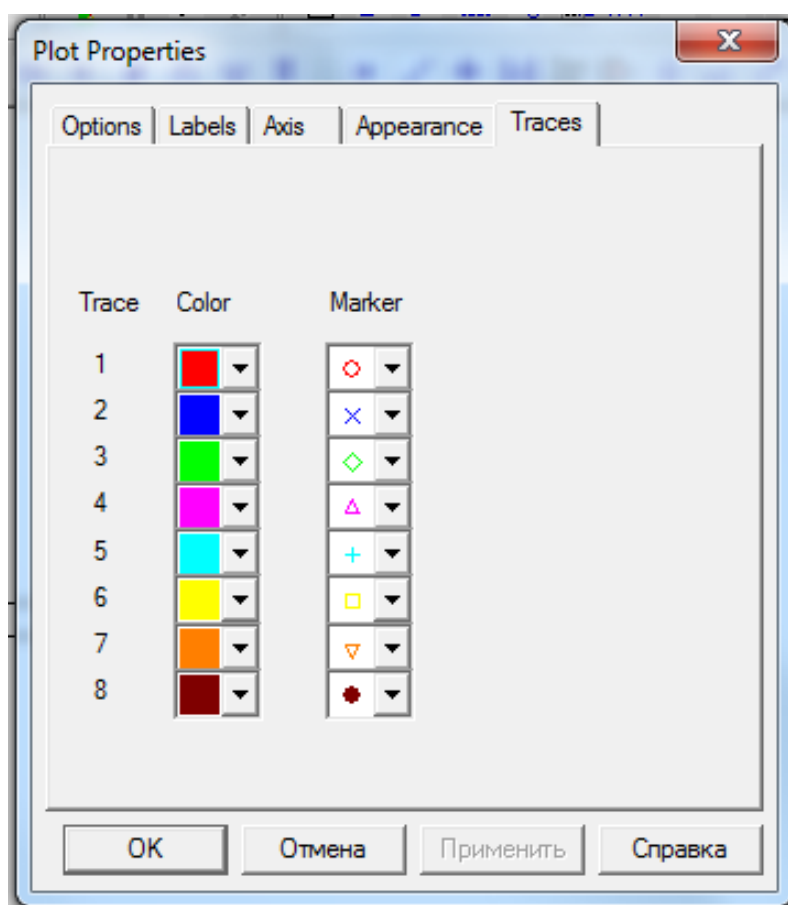


Рисунок П1.15. Окно настроек графопостроителя, закладка Traces

Кроме рассмотренного графопостроителя в меню Blocks/Signal Consumer доступен блок **Plot3D** для отображения векторных и матричных данных в виде трехмерных объектов.

### П1.2.2. Построитель гистограмм (Histogram)

Этот прибор позволяет визуализировать в виде гистограммы вероятностное распределение величины  $X$  в процессе моделирования. При этом горизонтальная ось гистограммы разбивается на одинаковые интервалы в заданном диапазоне значений  $X$ , а по вертикальной оси размещается количество ее дискретных значений, попавших в тот или иной интервал (bin). В качестве примера на рисунке П1.16 показана гистограмма распределения сигналов источника шума с нормальным (гауссовым) распределением со средним значением (Mean)  $M = 10$ , среднеквадратичным отклонением (Standard Deviation)  $SD = 2$  и значениями параметров рассматриваемого прибора, показанных в диалоговом окне на рисунке П1.17.

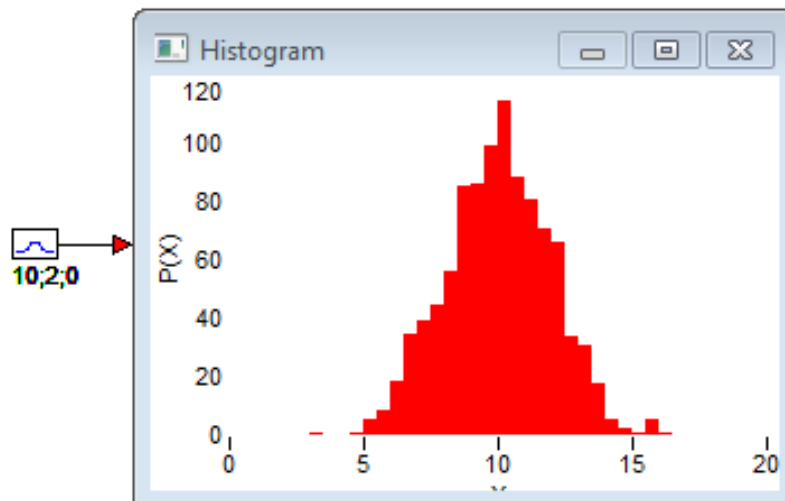


Рисунок П1.16. Гистограмма распределения сигналов источника

Назначение параметров диалогового окна построителя:

- Title – название (заголовок) гистограммы;
- Vertical Label, Horizontal Label – название вертикальной и горизонтальной осей;
- Bin Count – общее число интервалов в заданном диапазоне значений  $X$ : от максимального (Max Bin) до минимального (Min Bin);
- Max Bin Height – максимальное количество значений  $X$  по вертикальной оси;

- Autoscale – опция автомасштабирования гистограммы;
- Show Out-of-Range Data – разрешение на размещение данных, превышающих значение Max Bin Height, в двух крайних интервалах гистограммы, что позволяет судить о грубых ошибках, которые не подчиняются ожидаемому закону распределения.

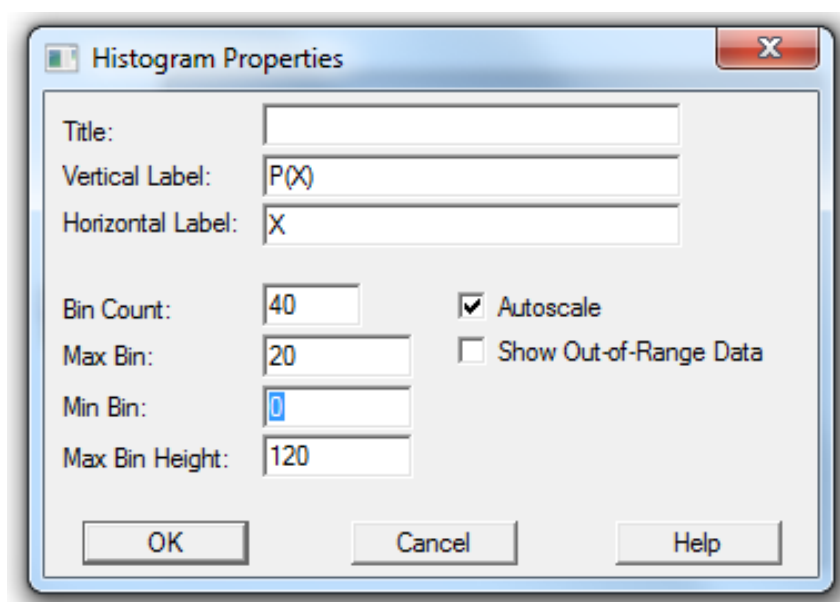


Рисунок П1.17. Параметры построителя гистограмм

### П1.2.3. Стрелочно-шкальный прибор (Meter)

Прибор является аналоговым (стрелочным или шкальным) указателем для визуализации постоянных или медленно меняющихся величин. В качестве источников контролируемых сигналов в примере на рис. П1.18 используются источники постоянных сигналов 1, 2 типа Constant.

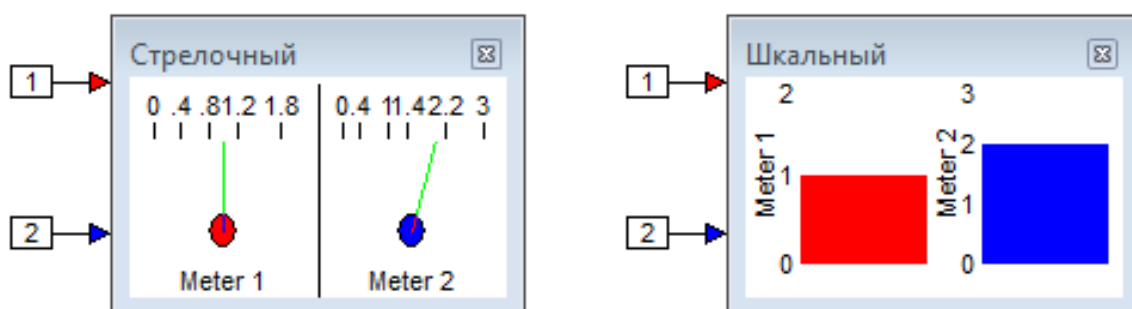


Рисунок П1.18. К применению стрелочно-шкального прибора



Назначение параметров диалогового окна (рис. П1.19):

- Style – опции переключения из стрелочного (Gauge) режима в шкальный (Bar);
- Meter # – выбор режима настроек (Settings for Meter) для каждого из восьми возможных приборов (в одном корпусе); формирование или удаление дополнительных выводов осуществляется инструментами, вызываемыми кнопками  ;

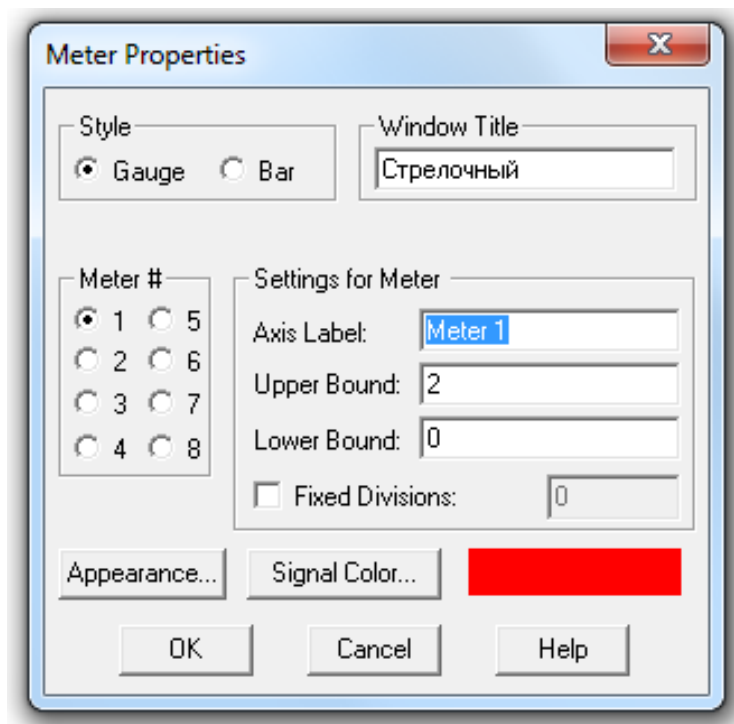


Рисунок П1.19. Параметры стрелочно-шкального прибора

- Window Title – поле для ввода общего названия блока;
- Axis Label – ввод имени для каждого прибора в блоке (располагается в нижней части стрелочного прибора или вертикально слева в шкальном);
- Upper Bound, Lower Bound – верхнее и нижнее значения контролируемой величины;
- Fixed Divisions – сохранение (значение 0) или удаление (значение 1) промежуточных значений шкалы приборов;
- Appearance – вызов окна для выбора цвета переднего плана (Foreground) и фона (Background);

– Signal Color – выбор цвета раскраски области подвески у стрелочных приборов и полос у шкальных.

#### П1.2.4. Графопостроитель-самописец (StripChart)

Этот прибор является аналогом ленточного самописца и предназначен для регистрации длительных процессов. Он во многом напоминает рассмотренный в разделе П1.2.1 графопостроитель с аналогичным, но несколько меньшим набором параметров и отличается от него «растянутой» временной шкалой и, соответственно, линейкой прокрутки для перемещения окна просмотра. Это отличие иллюстрируется в демонстрационном примере на рисунке П1.20, на котором показаны осциллограммы амплитудно-модулированных колебаний (АМК), зарегистрированные графопостроителем-самописцем StripChart и графопостроителем Plot.

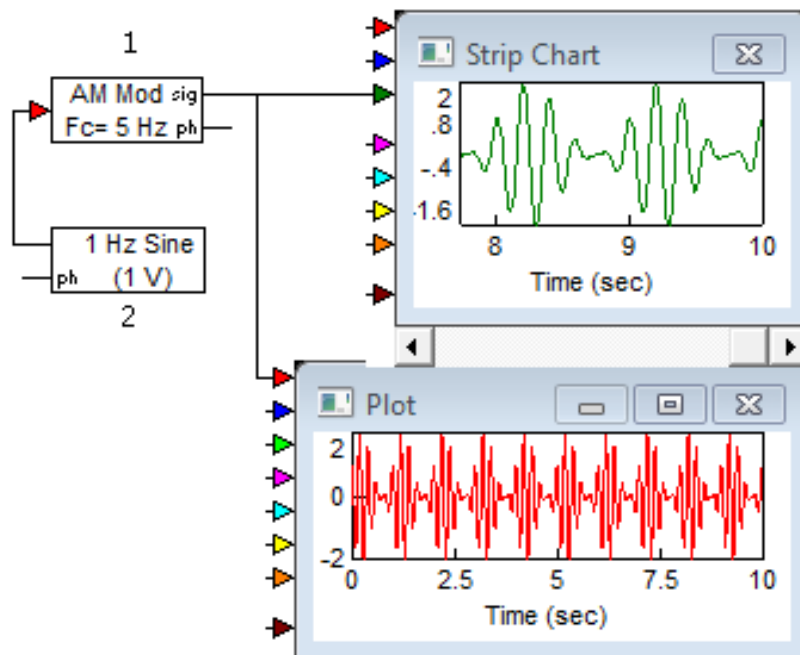


Рисунок П1.20. К применению прибора StripChart

В качестве источника АМК использован амплитудный модулятор 1 с синусоидальным источником 2 модулирующего сигнала. Как видно из рисунка П1.20, осциллограмма StripChart оказалась «растянутой» почти в 5 раз по сравнению с осциллограммой на дисплее Plot.

### П1.2.5. Алфавитно-цифровой регистратор (Display)

Варианты включения рассматриваемого прибора в сочетании с источником постоянного сигнала Constant показаны на рисунке П1.21, где видно, что блок Constant может выполнять следующие функции:  $y = \text{const}$  (1);  $y = \text{«текст»}$  (2);  $y = [\text{const } ij]$  (3). При этом ввод матрицы или вектора в строке Value диалогового окна блока Constant осуществляется с помощью квадратных скобок, данные строки разделяются пробелом или запятой, столбцы – символом «;» (4). Для генерации алфавитно-цифровой строки она в поле ввода Value включается в кавычки.

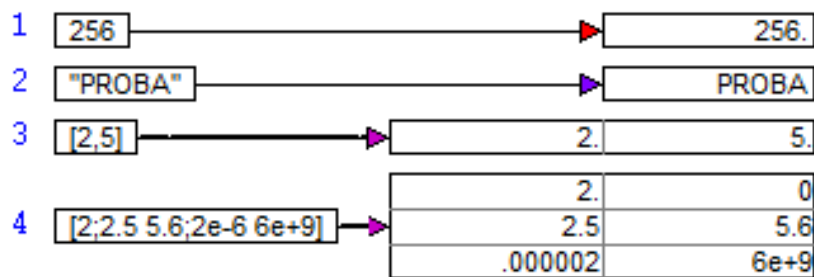


Рисунок П1.21. К применению прибора Display

В зависимости от задаваемого в окне блока Constant значения сигнала (параметр Value), иконка блока Display и соединительный проводник принимают разные формы, в частности для сигналов в виде простого числа или символьной строки – в виде одиночного (тонкого) проводника и иконки обычной формы. При векторной или матричной форме выходного сигнала источника Constant проводник превращается в «многожильный» (шину), а Display автоматически становится многооконным с количеством окон, определяемых количеством элементом вектора или матрицы.

В диалоговом окне блока Display (рис. П1.22) приведены следующие параметры:

- Value – строка для отображения текущего значения контролируемой величины;

- Total Digits – разрядность отображаемых величин (1 – 64);
- Specify Fractional Digits – отображать дробную часть числа;
- Fractional Digits – разрядность отображаемой дробной части числа (1 – 32);
- Use Hex Notation – использовать шестнадцатеричную систему счисления;
- Allow Room For Exponential Notation – разрешение на расширение значка блока при экспоненциальной нотации; при деактивации установленной по умолчанию опции иконка блока укорачивается, однако в случае переполнения разрядной сетки на его дисплее появляется изображение #####, что говорит о необходимости установки указанной опции;
- Show Leading Zero for Fractions – отображать начальное нулевое значение;
- Attributes – опции установки атрибутов: шрифта, фоновой окраски, а также возврат к параметрам по умолчанию.

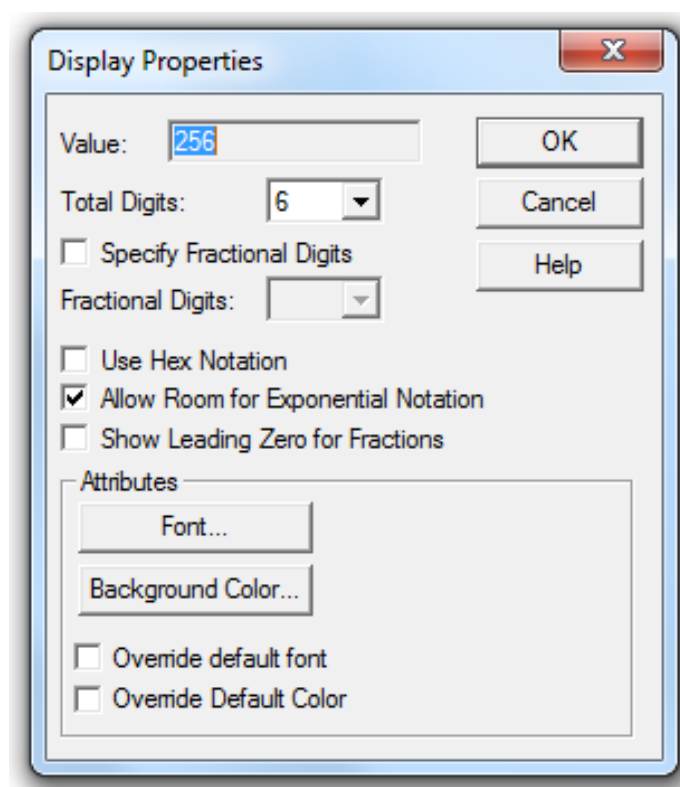


Рисунок П1.22. Диалоговое окно блока Display

### П1.2.6. Кнопка управления (Button) и остановка моделирования (Stop)

Блок Button является кнопкой, реагирующей на нажатие курсора мыши в процессе моделирования. Этот блок в исходном состоянии имеет вид прямоугольника с надписью On/Off (см. рис. П1.23). Однако размеры кнопки и «внутреннее содержание» могут быть изменены путем выбора названия кнопки (строка Button Name на рисунке П1.24) и соответствующего изображения BMP-формата (строка Bmp Path и окно Preview на рисунке П1.24).

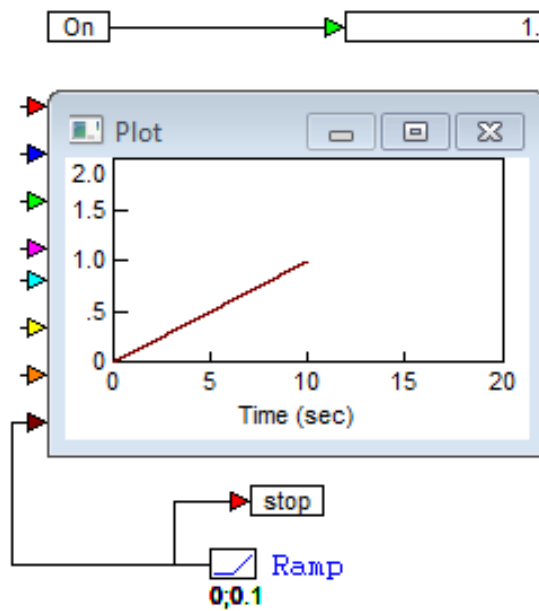


Рисунок П1.23. К применению блоков Button и Stop

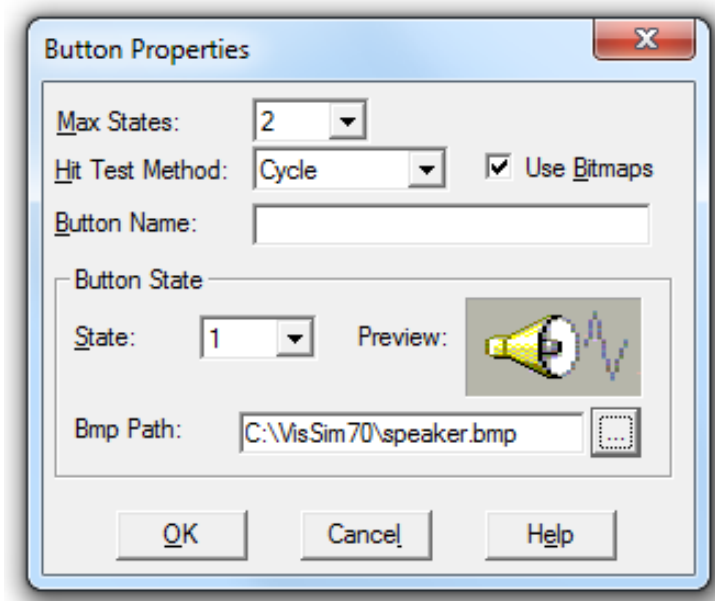


Рисунок П1.24. Диалоговое окно блока Button

Другие параметры диалогового окна, блока, вызываемого правым щелчком при нажатой клавише Ctrl, имеют следующее назначение:

1. Max States – максимальное число состояний выходного сигнала кнопки  $y = \text{state} - 1$ , где  $\text{state} = 2 - 16$ .
2. Button State – опции установки соответствия состояния кнопки и соответствующего изображения BMP-формата.
3. Hit Test Method – раздел выбора направления и порядка следования выходных сигналов кнопки при щелчке правой кнопкой мыши:
  - Cycle – циклическое изменение выходного сигнала с шагом +1;
  - Pie – выбор состояния при круговом движении курсора мыши перед нажатием ее правой кнопки;
  - Vertical, Horizontal – выбор состояния при вертикальном и горизонтальном перемещении курсора мыши;
  - Push Button – выбор состояния «лобовым» нажатием на кнопку.

Блок остановки моделирования Stop управляет процессом моделирования в зависимости от значения входного сигнала  $x$  с использованием следующего алгоритма:

- завершение моделирования с запретом авторестарта, если  $x > 2$ ;
- завершение моделирования без запрета авторестарта, если  $x > 1$ ;
- моделирование в обычном режиме в других случаях.

В демонстрационном примере на рисунке П1.23 в качестве контролируемого (входного) сигнала  $x$  используется источник линейно изменяющегося сигнала Ramp с параметрами: скорость изменения (Slope) выходного сигнала – 0,1 В/с; время задержки (Time Delay) – 0 с. Как видно из осциллограммы на рисунке П1.23, блок Stop срабатывает при входном напряжении  $0,1 \times 10 = 1$  В, при котором моделирование прекращается.

### **П1.2.7. Регулятор (Slider) и индикатор ошибки (Error)**

Блок Slider генерирует постоянный сигнал, величина которого может меняться перемещением курсором мыши его движка в виде спаренных

вертикальных отрезков (см. рис. П1.25). Для более точного позиционирования движка можно увеличить размер блока активацией опции High Precision Display в диалоговом окне команды Edit/Preferences. Применение блока напоминает регулировку напряжения реостатом или потенциометром.

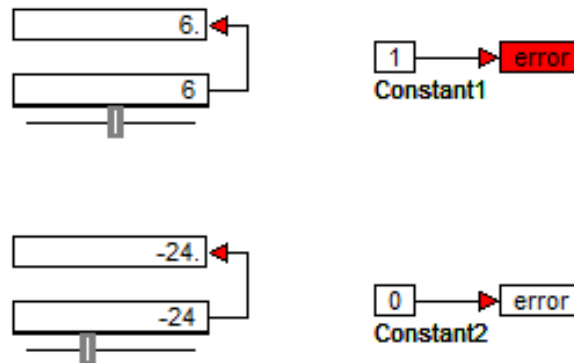


Рисунок П1.25. К применению блоков Slider и Error

Параметрами блока Slider являются (рис. П1.26):

- Current Value – текущее значение выходного сигнала;
- Upper Bound, Lower Bound – нижнее и верхнее значение выходного сигнала;
- Increment – приращение выходного сигнала (при активации опции % – в процентах) при каждом щелчке мышью по оси движка.

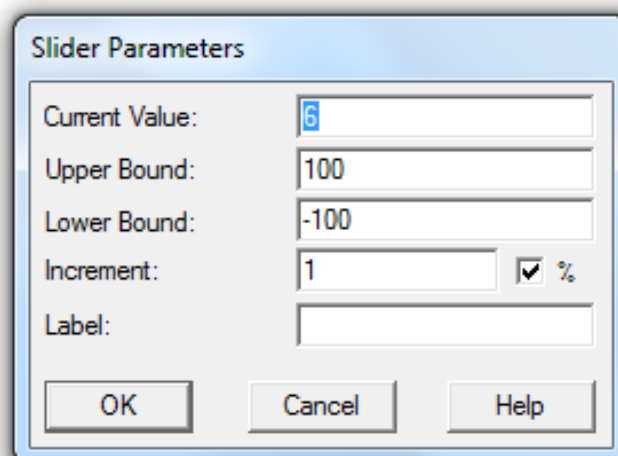


Рисунок П1.26. Диалоговое окно блока Slider

Для изменения сигнала с произвольным приращением движок регулятора просто перемещается курсором. Для активации программного (автоматического) изменения сигнала с фиксированным приращением нажимается левая кнопка мыши в требуемой конечной позиции на оси движка, и удерживается или отпускается за пределами блока.

Блок Error позволяет индцировать отклонение контролируемого сигнала от нулевого значения заполнением его иконки красным цветом, что иллюстрируется на рисунке П1.25, где в качестве источника контролируемого сигнала используются блоки Constant1 и Constant2 с выходным напряжениями соответственно 1 и 0 В.

### П1.2.8. Световой индикатор (Light)

Этот блок является имитатором светодиодного индикатора (рис. П1.27).

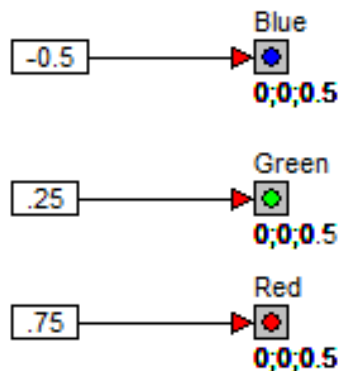


Рисунок П1.27. К применению блока Light

Блок Light обладает следующими параметрами:

1. Properties – в этом разделе в поле Value индцируется текущее значение входного сигнала  $x$ , его нижнее (Lower Bound,  $lb$ ) и верхнее (Upper Bound,  $ub$ ) значения.
2. Settings – выбор режима работы:
  - Lower – соответствует неравенству  $x > lb$  и красному свечению;
  - Safe – соответствует неравенству  $lb \leq x \leq ub$  и зеленому свечению;
  - Upper – соответствует неравенству  $x < ub$  и синему свечению.



3. Associations – ассоциация срабатывания индикатора (кроме цвета свечения) со звуковым сигналом (кнопки Sound для выбора WAV-файла и Play Sound – для его тестирования) или изображением в BMP-формате (кнопка Image).

4. Beep If Value Exceeds Upper Bound – опция на разрешение звукового сигнала при превышении входным сигналом верхнего уровня.

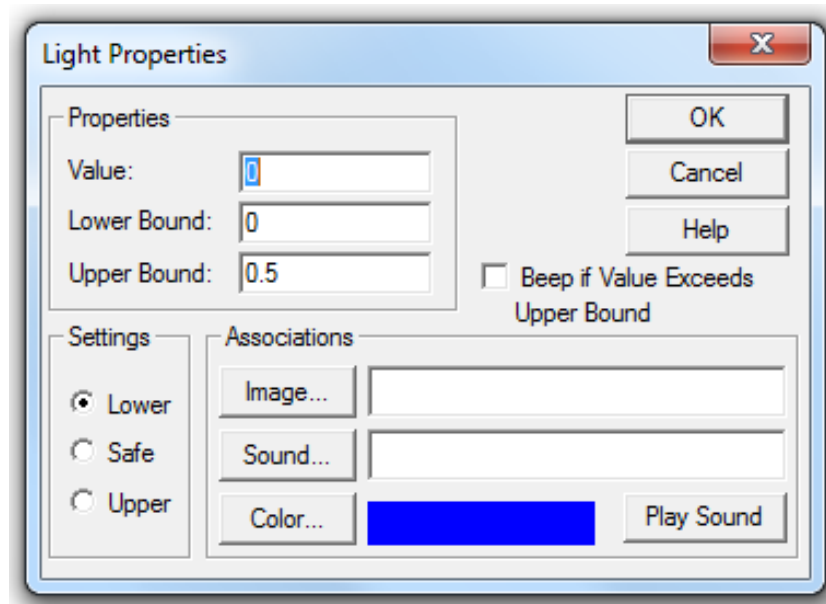


Рисунок П1.28. Диалоговое окно блока Light

### П1.2.9. Средства анимации

При создании анимации обычно реализуется принцип, применяемый при создании мультипликационных фильмов и заключающийся в формировании последовательности кадров (фреймов) со смещенными элементами изображения. Библиотека блоков VisSim позволяет проводить три типа анимаций: блок Animation служит для покадрового показа ряда изображений, представленных BMP-файлами; блок lineDraw служит для анимации отрезка прямой lineDraw путем пошагового изменения координат его начала и конца; блок Animation3D служит для отображения 3D-анимаций.

Реализация анимации в VisSim возможна, только в режиме Display Mode (меню View). В этом режиме доступна только часть блоков; соединения и составные блоки скрываются; не допускается редактирование блок-схемы,

перемещение блоков и др.; остается только возможность внесения изменений в процесс моделирования с помощью управляющих элементов: ползунковых регуляторов, кнопок и констант. После выполнения анимации рабочее поле можно обновить командой Edit/Repaint Screen.

### Анимация линии (LineDraw)

Блок анимации линии в наибольшей степени подходит для визуализации движения рычажных механизмов, маятников и других механических систем. В частности, на рисунке П1.29 приведена схема моделирования маятника.

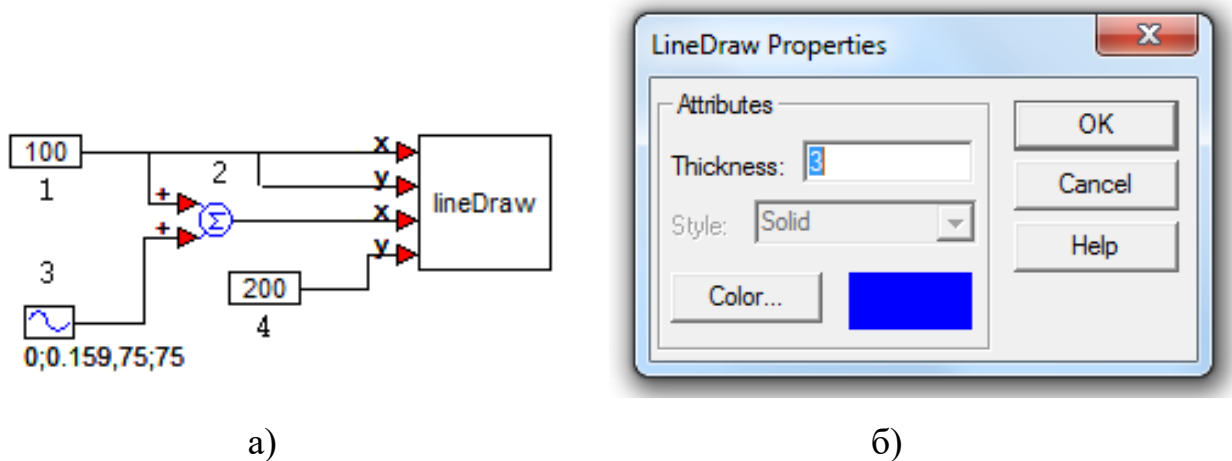


Рисунок П1.29. К применению блока LineDraw

Блок имеет два входа для задания координат  $x$  и  $y$  начала и конца отрезка прямой. Верхняя точка отрезка в модели имеет фиксированные координаты  $(100, 100)$ , задаваемые источником 1. Нижняя точка имеет фиксированную координату  $y = 200$ , задаваемую источником 3, и координату

$$x = 100 + 75\sin(\omega t) = 100 + 75\sin(2\pi \times f \times t) = 100 + 75\sin(2\pi \times 0,159 \times t),$$

формируемую сумматором 2. В приведенной формуле 100 – смещение, задаваемое источником 1;  $\omega = 2\pi \times f = 0,1$  рад/с – угловая частота колебаний маятника, задаваемая источником 3 синусоидального сигнала частотой  $f = 0,159$  Гц и амплитудой 75 В, определяющей размах колебаний маятника.

Параметры блока LineDraw (окно на рис. П1.29б):

- Thickness – ширина линии, пиксел;

- Style – тип линии: сплошная (Solid), штрихпунктирная (Dash) или в виде точек (Dot); последние два типа доступны только для однопиксельной ширины;
- Color – выбор цвета линии.

После выполнения команды View/Display Mode все проводники из схемы модели удаляются (рис. П1.30) и какие-либо настройки становятся недоступными. После выполнения моделирования «стержень» маятника фиксируется в одном из крайних положений. Параметры моделирования показаны на рисунке П1.31.



Рисунок П1.30. Результат работы схема моделирования маятника

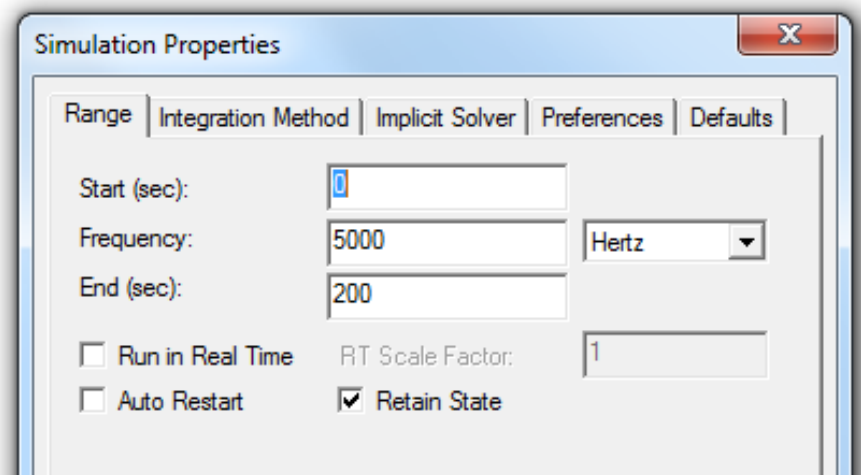
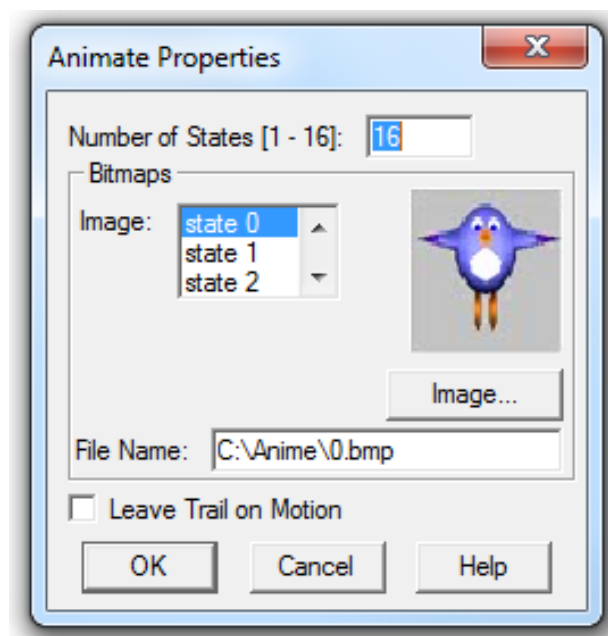


Рисунок П1.31. Параметры процесса моделирования

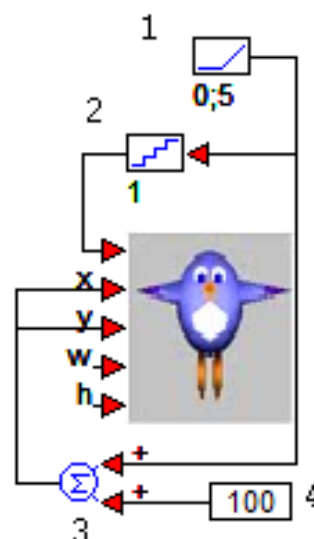
### Блок анимации Animation

Этот блок служит для создания анимации объектов произвольного вида. Для этого предварительно создается набор кадров (состояний) объекта в виде BMP-файлов, которые затем задаются для каждого состояния (State) в диалоговом окне блока (рис. П1.32a). Число кадров анимации (Number of States)

– от 1 до 16; имя файла для каждого кадра и путь к нему указывается в строке File Name или выбирается кнопкой Image. После загрузки файла изображение кадра появляется в диалоговом окне блока. Опция Leave Trail on Motion (Оставлять след от движения) позволяет усилить эффект движущегося объекта.



а)



б)

Рисунок П1.32. К применению блока Animation

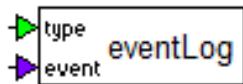
Блок анимации имеет пять входов (см. рис. П1.32б). Верхний (безымянный) предназначен для выбора номера отображаемого кадра (0 – 15), который в нашем случае производится линейно-ступенчатыми сигналами с блока 2 (квантователь Quantize); при изменении его выходного сигнала на шаг квантования 1 В производится выбор очередного кадра. На вход квантователя подается линейно изменяющийся сигнал с источника 1 (источник Ramp) с параметрами: время задержки (Time Delay) – 0; скорость изменения (Slope) – 4 В/с, т. е. смена кадров происходит четыре раза в секунду (в случае дробного значения сигнала оно округляется до ближайшего целого). Одновременно с источника Ramp линейно изменяющийся сигнал через сумматор 3 с источником смещения 4 подается на координатные входы x и y, определяющие координаты (в пикселях) относительно левого верхнего угла рабочего окна

дисплея. Входы *w* и *h* блока могут быть использованы для ограничения кадра по ширине и высоте соответственно (для создания эффекта приближения или удаления объекта). Работа модели может быть проверена и без перехода в режим Display Mode.

### Блок анимации Animation3D

Этот блок служит для отображения 3D-анимаций. Он вызывается одноименной командой из меню Blocks/Animation. Необходимая информация вводится с блока world3D, в окне которого данные представляются на языке VML, с блока mesh3D (с входными сигналами в виде матрицы 4×4 и постоянной, определяющей число анимаций) или с блока camera3D (3D-камера с входным сигналом в виде матрицы 4×4 для управления положением камеры в пространстве) и использования во всех трех случаях блока light3D световых спецэффектов. Все перечисленные блоки вызываются одноименными командами из меню Blocks/Animation.

#### П1.2.10. Средства регистрации событий



**EventLog** – блок регистрации событий, возникающих при моделировании. Для отображения регистрируемых блоком EventLog событий используется блок-дисплей **EventDisplay**. Данный блок на рабочем поле представляется в виде таблицы типа сообщений об ошибках при моделировании (в столбце Event type), время (Date and Time) события, блок (в столбце Source), с работой которого связано событие, и его краткое описание (в столбце Event).

#### П1.3. Библиотека компонентов Signal Producer

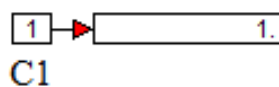
Библиотеки компонентов размещаются как в поле мнемонических кнопок рабочего окна на рисунке П1.1, так и в меню Comm и Blocks, а также (для VisSim/Comm) меню Wireless.

Внешний вид линейки мнемоблоков источников измерительных сигналов приведен на рис. П1.33. Полный список блоков этой группы находится в меню Blocs/Signal Producer.



Рисунок П1.33. Линейка мнемоблоков группы Signal Producer

**1** — **Constant** – источник постоянного сигнала; параметры (в диалоговом окне, вызываемом щелчком правой кнопки мыши): Value – выходная величина в виде алфавитно-цифровых строк (по умолчанию 1 – см. рис. П1.34а) или матриц (рис. П1.34б); Label – метка (наименование) блока, вводится латиницей или кириллицей. Поскольку метка в большинстве случаев является обязательным атрибутом каждого блока и имеет одинаковое назначение, то в дальнейшем мы не будем упоминать об этом параметре.



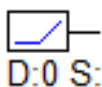
а)



б)

Рисунок П1.34. К применению блоков Constant и Display

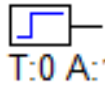
Как видно из рисунка П1.34б, ввод матрицы осуществляется с помощью квадратных скобок, данные строки разделяются пробелом, столбцы – символом «;», при этом линия связи с приемником данных изображается утолщенной линией (после запуска моделирования), а размеры значка блока Display (приемника данных), в отличие от MATLAB, изменяются автоматически в зависимости от формата входных данных.



**D:0 S:1 Ramp** – источник линейно изменяющегося сигнала; параметры:

Slope (S) – скорость изменения; Time Delay (D) – время задержки в единицах модельного времени; Label – метка блока, при введении которой параметры S,

$D$  не индицируются (это относится и к другим блокам рассматриваемой группы). В свою очередь, основные параметры блока в сокращенном виде выводятся только при выборе команды View/Block Labels. Выходной сигнал блока  $y = S \times (t - D)$ .



**Step** – потенциальный скачок с параметрами: Time Delay

( $T$ ) – время задержки; Amplitude ( $A$ ) – амплитуда скачка. Выходной сигнал блока  $y = A \times l(t - T)$ , где  $l(t - T)$  – единичная ступенчатая функция;  $t$  – текущее время;  $y = 0$ , если  $t < T$ , иначе  $y = 1 \times A$ .



**PulseTrain** – источник импульсов положительной полярности

единичной амплитуды с параметрами: Time Delay ( $D$ ) – время задержки; Time Between Pulses ( $I$ ) – время между импульсами; выходной сигнал блока  $y = d_0[nT - t]$ , где  $d_0[nT]$  – единичная импульсная решетчатая функция;  $n$  – номер дискретного момента времени;  $T$  – интервал следования импульсов;  $t$  – время запуска генератора (чистое запаздывание);  $y = 1$ , если время моделирования совпадает с моментами времени  $nT$ , в противном случае  $y = 0$ .



**Sinusoid** – источник синусоидального сигнала единичной

амплитуды с параметрами: Time Delay ( $D$ ) – время задержки; Frequency ( $F$ ) – частота (по умолчанию 0,159154943091895 Гц или 1 рад/с); Amplitude ( $A$ ) – амплитуда; выходной сигнал блока  $y = A \times \sin[\omega(t - D)]$ , где  $\omega = 2\pi F$  – круговая частота;  $t$  – текущее время.



**Import** – импорт данных в виде файлов с расширениями .DAT,

.M, .MAT, .WAV с параметрами:

- File Name – имя файла;
- Type – тип данных: char (символьный), unsigned char (беззнаковый символьный), short (короткий в фиксированном формате (5 знаков)), unsigned

short (беззнаковый короткий), long (длинный (15 знаков)), unsigned long (беззнаковый длинный), float (с плавающей точкой), double (с плавающей точкой двойной точности), matrix double (матричное двойной точности), scaled int (масштабированное целое), string (строковый);

- Dimension – формат данных: Scalar (скаляр), Vector (вектор), Matrix, (матрица);

- Start Column – номер столбца матрицы в случае сигнала, заданного матрицей;

- Start Row – номер строки матрицы в случае сигнала, заданного матрицей;

- Interpolate – опция интерполяции данных (определение значения функции в промежуточной точке по ее значениям в двух соседних точках отсчета);

- Extrapolate – опция экстраполяции данных (определение значения функции по ее значениям в двух предыдущих точках);

- Use Comma for Decimal Point – использовать запятую отделяющую целую часть от дробной в десятичной дроби;

- Play Sound – кнопка для прослушивания WAV-файла;

- Data Point Time Delta – опциональный выбор значения временного интервала дискретизации данных во входном файле: Fixed Interval (фиксированный интервал) или Time Data Column (по количеству выборок за интервал);

- Data File Info – информация о времени моделирования (начало, конец), объемах пакета и др.

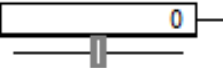
 — **SawTooth** – источник пилообразных импульсов; параметры: время задержки (Time Delay), частота следования (Frequency), амплитуда (Amplitude).

 — **SquareWave** – источник прямоугольных импульсов; параметры аналогичны источнику SawTooth.

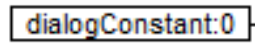
 — **TriangleWave** – источник треугольных импульсов; параметры аналогичны источнику SawTooth.



 **Button** – блок кнопки, реагирующей на нажатие курсора мыши в процессе моделирования.

 **Slider** – ручное или автоматическое изменение выходного постоянного сигнала в процессе моделирования с заданным приращением (1 – 10 %).

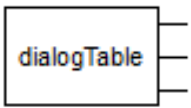
Далее рассматриваются источники сигналов из меню Blocs/Signal Producer, не вошедшие в мнемоническую группу.

 **DialogConstant** – блок генерации постоянной величины (константы), которая может быть параметром или начальным условием; совместно с сумматорами используется для создания диалоговых окон настроек составных блоков, предназначенных для их реконфигурации путем изменения параметров и начальных условий схемы модели без перехода внутрь ее составных блоков; параметры:

- Name – введенный в этой строке текст будет отображаться при вызове диалогового окна настроек составного блока (командой Edit/Block Compound properties); рядом с текстом будет или строка ввода, или переключатель, или список выбора;

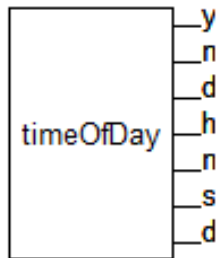
- Type – выбор типа генерируемой блоком константы из списка: Double (число с плавающей точкой двойной точности), Boolean (логические 0 или 1), Integer List (целочисленные значения); в диалоговом окне настроек составного блока будет либо строка ввода, либо переключатель, либо список выбора;

- Value – значения Double, Integer List или Boolean в соответствии с правилами их ввода.

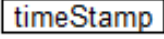
 **DialogTable** – блок для табличного представления параметров выбранного класса моделей (например, двигателей постоянного тока разных типов). Данные заносятся в предлагаемую таблицу из семи полей (столбцов) и могут быть представлены файлом в формате Microsoft Excel, Lotus 1-2-3 и т. п., а также с помощью редактора Notepad.

 — **Parabola** – блок генерации сигнала с постоянным ускорением; выходной сигнал блока  $y = a \times (t - \tau)^2$ , где:  $a$  – ускорение (параметр Slope Rate);  $\tau$  – временное смещение (параметр Time Delay);  $t$  – текущее время моделирования (отличается от реального).

 — **RealTime** – блок отсчета времени в реальном масштабе (в миллисекундах).



**TimeOfDay** – блок выводит на экран текущую дату и время.

 — **TimeStamp** – данный блок производит формирование уникальной строки в виде %M%d%y%h:%n:%s с метками времени для экспорта данных в файл с уникальным именем.

Источники измерительных сигналов представлены также в меню Comm/Signal Sources.

## П1.4. Группа оформления и преобразования величин

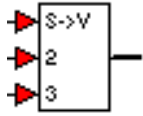
Внешний вид линейки мнемоблоков группы оформления и преобразования величин приведен на рис. П1.35. Полный список блоков этой группы находится в меню Blocks/Annotation.



Рисунок П1.35. Линейка мнемоблоков группы Annotation

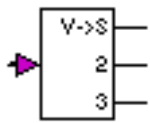
 **Label** – подготовка комментариев в вызываемом правым щелчком мыши окне с возможностью выбора атрибутов шрифта и цвета фона. Для большинства блоков метки определяются в их диалоговых окнах и активизируются (выводятся на экран) командой View/Block Labels; если при

этом метка не определена, то вместо нее отображаются в сокращенном виде параметры блока.



**ScalarToVec** – блок преобразования скалярных величин в вектор;

параметры: Rows – число строк; Cols – число столбцов. Блок позволяет, во-первых, объединить несколько проводников в шину, что повышает качество («читаемость») оформления блок-схем, и, во-вторых, сигналы, собранные в шину могут быть использованы в операциях векторной и матричной алгебры; для вывода отдельных сигналов из шины используется блок VecToScalar.



**VecToScalar** – блок преобразования вектора в набор скалярных

величин (без параметров, не считая метки).

Комментарий

Coment – окно-ярлык для комментариев с

возможностью изменения его размеров стандартным способом; вход в режим ввода текста или редактирования производится правым щелчком мыши по значку блока, выход – левым щелчком по свободному участку рабочего поля.



**Variable** – блок присвоения имени переменной; позволяет,

например, существенно уменьшить количество проводников схемы модели, если какой-либо сигнал используется многократно; в этом случае к выходу источника такого сигнала достаточно подключить такой блок, указав в его диалоговом окне имя переменной, после чего его можно использовать в качестве источника сигнала в любой части схемы модели, включая подсхемы (субблоки). Для поиска дубликатов переменной используются кнопки Find Def или Find Ref диалогового окна.

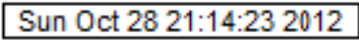
При объявлении переменных не рекомендуется использовать имена 'X'; '1', '2', '3' чтобы исключить возможные конфликты с другими блоками. Для ограничения действия переменной ее имя предваряется двоеточием (:имя) в случае ее использования только на текущем уровне иерархии блок-схемы (локальная переменная) или двойным двоеточием (::имя), если она определена

на любом уровне блок-схемы и может быть переопределена на более низких уровнях (переопределяемая переменная). Переменная без указанных атрибутов считается глобальной и может быть использована в любом месте блок-схемы, т. е. на любом иерархическом уровне.

В VisSim имеются несколько встроенных переменных, назначение которых заключается в следующем:

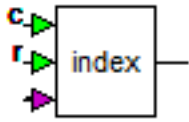
- \$firstPass, \$lastPass – формирование импульса единичной амплитуды на первом и последнем шаге моделирования соответственно;
- \$randomSeed – хранение иницирующего число для Random-генератора, которое устанавливается в диалоговом окне настроек моделирования;
- \$runCount – номер текущего сеанса моделирования в режиме авторестарта;
- \$timeStep, \$timeStart, \$timeStop – шаг, начальное и конечное время моделирования соответственно.

➡ **WirePositioner** – блок позиционирования одиночных и шинных соединительных проводников.

 **Date** – ярлык текущих даты и времени (по данным системных часов ПК).



**Bezel** – блок для размещения отдельных элементов схемы модели с возможностью изменения его размеров, выбора цвета фона кнопкой Select Color или BMP изображения с указанием пути к нему в строке File Name или выбора кнопкой Image. Может использоваться, например, для оформления панели оператора в режиме Display Mode.



**Index** – блок адресации (по входам  $C$  и  $r$ ) отдельных элементов входной матрицы (добавлен в программе VisSim/Comm). Если адресуется элемент вне границ матрицы, то выводится предупреждающее сообщение: Index n too big (Индекс  $n$  слишком большой).

### П1.5. Арифметические функции и операции

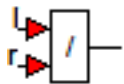
Внешний вид линейки мнемокнопок группы арифметических функций и операций приведен на рис. П1.36. Полный список блоков этой группы находится в меню Blocks/ Arithmetic.



Рисунок П1.36. Линейка мнемокнопок группы Arithmetic

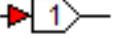
➔ **abs** – **Abs** – блок определения абсолютной величины входного сигнала в виде скаляра, вектора или матрицы.

➔ **convert** – **Convert** – преобразование типа данных входного сигнала в данные типа char, unsigned char, short, unsigned short, int, long, unsigned long, float, double; преобразования могут сопровождаться переполнениями; для отслеживания подобных ситуаций необходимо активизировать опцию Warn Numeric Overflow в окне команды Simulation/Simulation Properties/Preferences (разрешение на вывод сообщения о числовом переполнении).

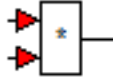


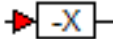
**Divide** – блок деления  $I$  на  $r$  (без параметров). Если входной сигнал  $r$  не подан или равен нулю, то VisSim выводит информационное сообщение о факте деления на ноль и устанавливает флаг ошибки, раскрашивая блок красным цветом. Возможно выполнение операции для шинных проводников, т. е. для матриц-шин одинакового размера. При этом матрицы-шины будут

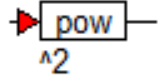
поделены поэлементно. Можно так же делить все сигналы в шине на один масштабирующий сигнал или выполнять обратную операцию.

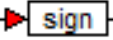
 **Gain** – блок умножения входной величины на параметр Gain (коэффициент передачи); является безынерционным звеном, поэтому его нельзя охватывать цепью отрицательной обратной связи, поскольку получается так называемая алгебраическая петля.

 **Inverse** – преобразование типа  $1/X$  входной величины  $X$  (скаляра, вектора или матрицы) без параметров.

 **Multiply** – блок перемножения двух скалярных или векторных величин (без параметров). Для векторного перемножения сцепленных матриц (число столбцов первой матрицы равно числу строк второй) блок используется в категории «матричные операции».

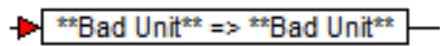
 **Negate** – инвертирование входной величины (скаляра, вектора или матрицы) без параметров.

 **Pow** – возведение входной величины (скаляра, вектора или матрицы) в степень, задаваемую параметром Exponent или сигналом  $x_2$  на втором входе, формируемом инструментами, вызываемыми кнопками . Возможно выполнение этой операции для шинных проводников, т. е. для матриц-шин одинакового размера. При этом они возводятся в степень поэлементно. Можно также возводить в одну степень все сигналы в шине или один сигнал возвести во множество степеней.

 **Sign** – определение знака скалярного входного сигнала: при положительном значении на выходе выдается  $+1$ , при отрицательном  $-1$  и ноль – при нулевом значении.

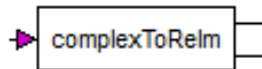
 **SummingJunction** – сумматор входных величин (скаляр, вектор или матрица) без параметров; добавление или удаление выводов производится инструментами, вызываемыми кнопками ; изменение знака суммирования

(+) на знак вычитания (–) и оборот производится правым щелчком мыши по соответствующему выводу с предварительно нажатой клавишей Ctrl. Возможно выполнение операции для шинных проводников (матриц-шин). Матрицы-шины суммируются поэлементно. Если размеры матриц не совпадают, то VisSim увеличивает меньшую, дополняя ее нулями. Можно также просуммировать все сигналы в шине с одиночным сигналом.



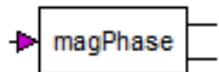
**UnitConversion** – преобразование единиц

измерения ускорения (acceleration), площади (area), емкости (capacitance), заряда (charge), проводимости (conductivity), тока (current), энергии (energy), расхода (flow rate), силы (force), индуктивности (inductance), магнитного потока (magnetic flux), массы (mass), положения (position), мощности (power), давления (pressure), скорости (speed), температуры (temperature), объема (volume) и других величин, задаваемых параметром Class; преобразование единиц измерения из одной системы (From) в другую (To), включая национальные.



**ComplexToReIm** – блок определения мнимой (imag) и

действительной (real) части комплексной величины.



**MagPhase** – блок определения модуля (mag) и аргумента

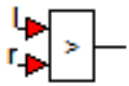
(phase) комплексной величины.

## П1.6. Математические и логические операции

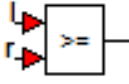
Внешний вид линейки мнемокнопок группы математических и логических операций приведен на рис. П1.37. Полный список блоков этой группы находится в меню Blocks/Boolean.



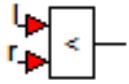
Рисунок П1.37. Линейка мнемокнопок группы Boolean



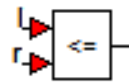
операция  $I > r$ ; если  $I > r$ , то на выходе формируется логическая 1; если  $I < r$ , то – 0. В диалоговом окне этого и остальных блоков, за исключением блока Not, могут быть выбраны любые из перечисляемых ниже операторов.



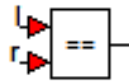
операция  $I \geq r$  (больше или равно).



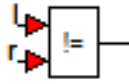
операция  $I < r$  (если меньше, то 1; если  $I > r$ , то 0).



операция  $I \leq r$  (меньше или равно).

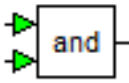


операция  $I = r$  (если равно).

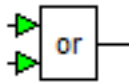


операция  $I \neq r$  (не равно).

Следует заметить, что в случае входных величин в виде чисел с плавающей точкой (float, double long double) результат на выходе перечисленных блоков может быть неверным, так как в математических сопроцессорах значения в младших знаковых местах мантиссы чисел с плавающей точкой могут различаться даже при повторении одной и той же совокупности машинных команд, и вероятность отличий тем больше, чем больше выполнено математических вычислений до операции сравнения. При этом при визуализации мантисса (до 20 знаковых мест) выводится не полностью (0,500000000...0000000xx) и обнаружить ошибку сложно.



логическая операция **И** (поразрядное логическое умножение входных сигналов).

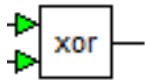


логическая операция **ИЛИ** (поразрядное логическое сложение входных сигналов).



логическая операция **НЕ**.

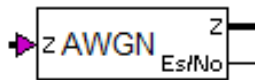




логическая операция **ИСКЛЮЧАЮЩЕЕ ИЛИ** (поразрядное логическое сложение по модулю 2 входных сигналов).

Далее рассматриваются компоненты, содержащиеся в меню Comm и в программе VisSim/Comm 7.0.

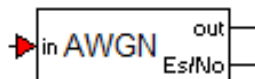
### П1.7. Каналы связи (Comm/Channels)



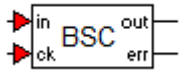
**AWGN (Additive White Gaussian Noise) Complex** – канал с аддитивным белым шумом с нормальным (гауссовым) распределением и комплексными входным и выходным сигналами; параметры:

- Number of Runs – количество пусков (до 10) с возможностью выбора параметра SNR (Signal to Noise Ratio – отношение сигнал/шум) для каждого пуска;
- Es/No – значение отношения сигнал/шум в децибелах; при известном количестве битов на символ может быть преобразовано в отношение Eb/No;
- Ref. Average Complex Signal Power – значение средней мощности сигнала, принимаемое в качестве опорного при определении отношения Es/No: 1 Вт на нагрузке 1 Ом или 13 дБм на нагрузке 50 Ом;
- Symbol Rate – частота следования входных сигналов (символов), Гц;
- Ref. Power Units – выбор режима измерения мощности: в децибелах мощности на нагрузке 50 Ом (опция dBm in 50 Ohms) или в ваттах на нагрузке 1 Ом (опция Watts in 1 Ohm).

Блок может быть использован в качестве источника шумового сигнала (при отсутствии входного), а также в сочетании с блоком BER.



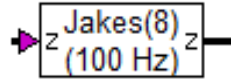
**AWGN (Additive White Gaussian Noise) Real** – блок аналогичен предыдущему, но для скалярных входного и выходного сигналов.



**BSC** (Binary Symmetric Channel) – бинарный симметричный

канал передачи цифровых сигналов; сигналы с уровнем  $> 0,5$  передаются на выход в виде логической 1, в противном случае – в виде логического нуля; параметры:

– Channel Error Probability – вероятность ошибочной передачи сигнала ( $0,9999 - 3,1 \times 10^{-8}$ ).



**Jakes Mobile** – Jakes-модель канала с замираниями,

характерными при связи с подвижными объектами и вызванными эффектами Рэлея (суммирование многократно отраженных комплексных синусоид) и Допплера (девиация несущей частоты).

Выходной сигнал блока

$$y(t) = x(a + jb)(2N_0 + 1)^{1/2},$$

где  $a = \sum_{k=1}^N \cos \varphi_k \cos \omega_k t + 2^{1/2} \sin \varphi_m \cos \omega_m t$ ;

$$b = \sum_{k=1}^N \sin \varphi_k \cos \omega_k t + 2^{1/2} \sin \varphi_m \cos \omega_m t$$

$$\omega_k = \omega_m \cos(2\pi k / N_0);$$

$$\varphi_k = \pi k / (N + 1);$$

$N_0 = 2(2N + 1)$  – общее число термов;

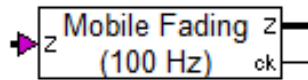
$N$  – отсчет доплеровского сигнала за промежуток времени, равный около  $3\pi/\omega_m$ ;

$\varphi_m$  – начальная фаза сигнала Допплера при максимальном значении частоты (параметр Doppler Frequency).

Параметры:

– Number of Terms – число  $N$  термов, которые нужно использовать в Jakes-модели в дополнение к первичному терму (отсчету) доплеровского сигнала;

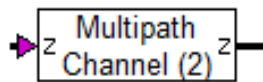
– Doppler Frequency – максимальная частота Доплера ( $\omega_m/2\pi$ ), Гц; определяется скоростью перемещения объекта.



**Mobile Fading** – аналогичен Jakes-каналу с тем лишь отличием, что замирание имеет не аддитивный, а мультипликативный характер (перемножается, а не складывается с входным сигналом). Параметры:

– Number of Taps – число выборок внутреннего FIR фильтра с учетом медленных (постепенных) изменений параметров;

– Doppler Shift – максимальное смещение доплеровской частоты, Гц; определяется скоростью перемещения объекта.



**Multipath** – канал, имитирующий взаимодействие прямого и многократно отраженного сигналов в приемнике получателя; параметры:

– Number of Paths – число сигналов, поступающих на приемник с разных направлений (до четырех);

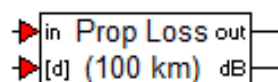
– Initial Condition (Real), Initial Condition (Imag) – начальные значения вещественной и мнимой составляющих для сдвигового регистра модели;

– Delay Mode – задержка сигнала: в единицах шага моделирования (опция Sim Steps) или секундах (опция Seconds);

– Path Delay – установка задержки сигнала для каждого из четырех направлений;

– Weight – относительный вклад сигнала данного направления;

– Phase Rotation – значение фазы для каждого пришедшего сигнала в градусах.



**Propagation Loss** – канал, имитирующий затухание сигнала в свободном пространстве согласно формуле:  $y(t) = c \times x(t) / 4\pi f d$ , где  $c$  – скорость света,  $d$  – расстояние между передатчиком и приемником (в километрах или милях),  $f$  – частота сигнала, МГц.

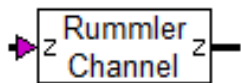
Параметры:

- Path Distance – расстояние  $d$ ; используется в режиме Internal Distance Mode; Frequency – расчетное значение частоты, МГц;
- Distance Mode – выбор режима определения расстояния; указанный в диалоговом окне через параметр Path Distance (External) или полученный извне по входу [d] (Internal);
- Distance Units – выбор единицы измерения расстояния: в километрах (kilometers) или милях (miles).



**Rice/Rayleigh** – канал с имитацией тропосферного или ионосферного рассеивания; параметры:

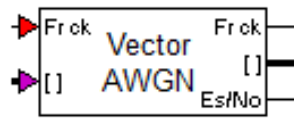
- Rice Factor ( $r$ ) – тип модели; при  $r = 0$  модель является рэлеевской (чистое рассеивание); при  $r = 1$  – райсовской;
- RMS Doppler Shift (Bd) – действующее значение доплеровского смещения частоты, Гц; период такого смещения приблизительно равен времени когерентного распространения, когда фаза незначительно отличается от своего среднего значения;
- RMS Fade Level (LdB) – среднеквадратичное значение затухания (потерь) канала, дБ (по умолчанию 0 дБ); положительные значения параметра указывают на потери сигнала.



**Rummler Multipath** – канал микроволнового (оптического) диапазона, имитирующий взаимодействие прямого и многократно отраженных цифровых сигналов; параметры:

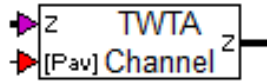
- Reflected Path Delay – задержка отраженного сигнала относительно прямого (по умолчанию 6,3 нс);
- Shape Parameter – затухание отраженного сигнала в относительных единицах (по отношению к входному сигналу);
- Fade Null Frequency – частота затухающих колебаний отраженного сигнала, Гц;

– Overall Attenuation Parameter – затухание прямого сигнала в относительных единицах.



**Vector AWGN** – комплексная версия канала AWGN;

в состав параметров, кроме перечисленных для канала AWGN, дополнительно включен Vector Size – размерность вектора.



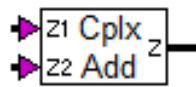
**TWTA** (Traveling Wave Tube Amplifier) – ламповый

усилитель бегущей волны, который часто используется для моделирования каналов спутниковой связи; выводы: z – вход и выход комплексного сигнала; Pav – управление средним уровнем мощности входного сигнала.

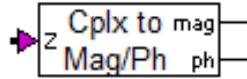
Блок реализует формулу:  $y(t) = A \times G(r) \times e^{j\Phi(r)} x(t)$ , где  $A$  – коэффициент масштабирования;  $G(r) = r \times \alpha_a / (1 + r^2 \beta_a)$  – усиление по амплитуде, дБ;  $\Phi(r) = r \times \alpha_\phi / (1 + r^2 \beta_\phi)$  – сдвиг по фазе;  $r = (|x_1|^2 / P_{av})^{1/2}$ ;  $x_1$  – номинальное значение входного сигнала; Pav – средняя мощность входного сигнала. Параметры:

- Operating Point – рабочая точка TWTA, дБ; определяется относительно мощности при номинальном значении входного сигнала  $x_1$  принятой за 0 дБ;
- Saturation Gain – усиление в области насыщения, дБ;
- Average Power Mode – режим установки средней мощности входного сигнала: внутренний (Internal) или внешний (External);
- Average Input Power – средняя мощность входного сигнала, Вт; параметр задается только в режиме Internal;
- Alpha\_a ( $\alpha_a$ ), Beta\_a ( $\beta_a$ ) – коэффициенты усиления по амплитуде (см. приведенную выше формулу);
- Alpha\_phi ( $\alpha_\phi$ ), Beta\_phi ( $\beta_\phi$ ) – коэффициенты сдвига по фазе (см. приведенную выше формулу).

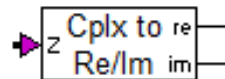
## П1.8. Операции с комплексными числами (Comm/Complex Math)



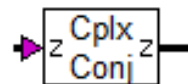
**Addition** – сумматор двух комплексных величин (без параметров).



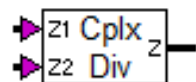
**Complex to Mag/Phase** – преобразователь комплексной величины в ее модуль и аргумент (фазу) (без параметров).



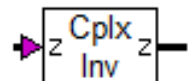
**Complex to Real/Imag** – преобразователь комплексной величины в ее вещественную и мнимую составляющие (без параметров).



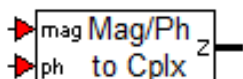
**Conjugate** – преобразователь комплексной величины в комплексно сопряженную (без параметров).



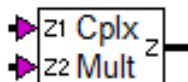
**Division** – устройство деления двух комплексных величин (без параметров).



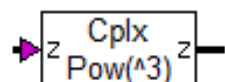
**Inverse** – преобразователь комплексной величины в обратную (типа 1/X) (без параметров).



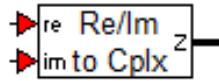
**Mag/Phase to Complex** – преобразователь модуля и аргумента (фазы) в комплекс (без параметров).



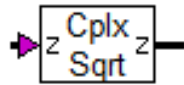
**Multiplication** – блок перемножения двух комплексных величин.



**Power** – блок возведения входного комплексного сигнала в заданную степень (параметр Exponent).

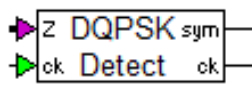


**Real/Imag to Complex** – преобразователь вещественной и мнимой составляющих в комплекс (без параметров).



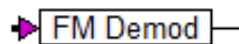
**Square Root** – блок извлечения квадратного корня из комплексной величины (без параметров).

### П1.9. Демодуляторы (Comm/Demodulators)



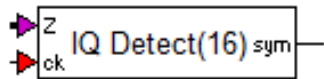
**DiffPSK Detector** – блок детектирования DQPSK (Differential Quadrature Phase Shift Keying) дифференциальных и квадратурных фазоманипулированных сигналов; параметры:

- DPSK Type – выбор типа демодулятора: DBPSK (бинарный), DQPSK (квадратурный), Pi4DQPSK (квадратурный 8-позиционный), D8PSK (дифференциальный 8-позиционный), D16PSK (16-позиционный) или D32PSK (32-позиционный);
- Initial Phase – начальная фаза, град.;
- Select File – выбор map-файла настроек и пути к нему (DPSK File Path) с помощью кнопки Browse File; используется в случае корректировки параметров модели с использованием текстового редактора Notepad (структура текстового файла имеется в справочной системе).



**FM Demodulator** – детектор частотно-модулированных сигналов; параметры:

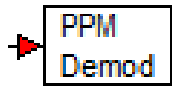
- FM Deviation Index – индекс модуляции;
- Offset – постоянная составляющая; определяется отношением несущей частоты и индекса модуляции;
- Overflow Value – допустимое значение переполнения (например, при делении на ноль).

**IQ Detector**

– детектор модулированного

комплексного сигнала ((I, Q) – комплексная плоскость); выходы: z – вход; ck – тактовый вход; sym – выход; обычно используется в сочетании с блоком Integrate & Dump ; параметры:

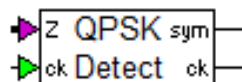
- Constellation Size – размер сигнального созвездия на IQ-плоскости; определяется количеством точек, отображающих концы векторов модулированного сигнала;
- Select File, Browse File, File Path – выбор файла с данными детектора и пути к нему.

**PPM Demodulator**

– детектор широтно-импульсных (Pulse

Position Modulated) сигналов; параметры:

- Number of Levels – количество возможных значений (уровней) сигнала (символа);
- Pulse Width – длительность прямоугольного импульса, с;
- Symbol Rate – скорость следования сигналов, символ/с;
- Frame Start Delay – начальная задержка фрейма (выборки), с.

**PSK (Phase Shift Keying) Detector**

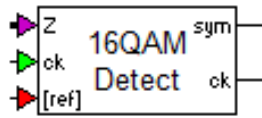
– набор демодуляторов

фазоманипулированных (ФМн) (не дифференциальных) сигналов; параметры:

- PSK Type – выбор типа PSK: BPSK (бинарный), QPSK (квадратурный 8-позиционный), 8-PSK (8-позиционный), 16-PSK (16-позиционный), 32-PSK (32-позиционный) и SQPSK (Staggered QPSK – с относительной (смещенной) квадратурной фазовой манипуляцией (ОФМн));
- Constellation Rotation – циклический сдвиг составляющих относительно 4 значения по умолчанию, град.;
- Channel Rotation – циклический сдвиг фазы относительно полученной из канала связи, рад;



– PSK File Path – выбор пути к файлу настроек с использованием кнопок выбора файла (Select File) и пути к нему (Browse File).



**QAM/PAM Detector** – детектор квадратурных

амплитудно-модулированных и импульсно-модулированных сигналов; набор параметров аналогичен предыдущему блоку, за исключением:

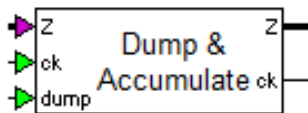
– QAM/PAM Type – выбор типа модуляции: 16-QAM, 32-QAM, 64-QAM, 128-QAM, 256-QAM, 4-PAM, 8-PAM или 16-PAM (числа обозначают количество используемых уровней амплитуды);

– Amplitude Reference – выбор способа установки уровней амплитуды сигнала: внутренняя (Internal) или внешняя (External) с использованием входа ref;

– ck – вход и выход синхросигналов;

– Constellation Spacing – интервал посылок; параметр Доступен только в режиме Internal.

### П1.10. Цифровые устройства (Comm/Digital)

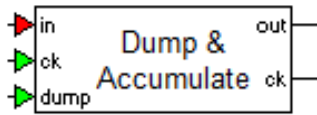


**Accumulate Dump (Complex)** – накопитель

комплексных сигналов с разгрузкой (передачей на выход); выводы: Z – вход и выход информационного сигнала; ck – тактовый вход и выход; dump – вход сигнала разгрузки; параметры:

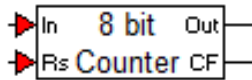
– Dump Mode – режимы работы: с первоначальным накоплением (Accumulate First) и последующей разгрузкой или с первоначальной разгрузкой (Dump First) и последующим накоплением;

– Average by the Input Count – опция вывода среднего значения выборки за период накопления.



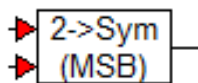
**Accumulate Dump (Real)** – накопитель скалярных

сигналов с разгрузкой; назначение выводов и набор параметров аналогичных предыдущему пункту.



**Binary Counter** – двоичный счетчик с параметрами:

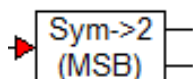
- Number of Bits – разрядность счетчика (1–31);
- Counter Initial Value – начальное состояние счетчика;
- Threshold – порог чувствительности по счетному входу;
- Edge Mode – способ запуска: по переднему (Rising Edge) или заднему (Falling Edge) фронту входного импульса.



**Bits to Symbol** – преобразователь  $n$ -битного параллельного

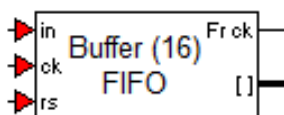
кода в символы с параметрами:

- Bit Order – порядок преобразования входных битов: поступающий на первый (верхний) вход бит является младшим (LSB First) или старшим (MSB First);
- Number of Input Bits – количество  $n$  входных битов (2 – 16), определяющих количество входов.



**Symbol to Bits** – блок, противоположный по выполняемым

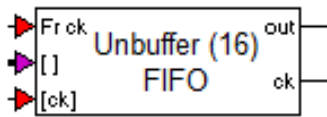
функциям блоку Bits to Symbol с аналогичным набором параметров.



**Buffer** – накопитель данных, поступающих

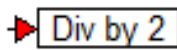
последовательно на вход *in* и сопровождаемых тактовыми импульсами по входу *ck*; после заполнения буфера на его выходе *Fr ck* формируется тактовый импульс, под действием которого накопленные данные параллельным кодом выдаются на шину *[]*; вход *rs* используется для сброса накопленных данных сигналом логической 1; параметры:

- Output Mode – выбор алгоритма работы буфера: FIFO (First In First Out – первым вошел и первым вышел) или LIFO (Last In First Out – последним вошел и первым вышел); вышедший первым бит становится на выходе буфера первым элементом вектора;
- Buffer Size – размер буфера (1 – 8192); указывается на значке блока в круглых скобках.



**Unbuffer** – блок по выполняемым функциям противоположен предыдущему; параметры (кроме указанных для Buffer):

- Timing – способ синхронизации: внутренний (Internal) или внешний (External);
- Output Rate – скорость вывода данных в режиме Internal, Гц.



**Divide by N** – двоичный счетчик с коэффициентом пересчета  $N$ ; параметры:

- Divide by – выбор коэффициента  $N$  (только целое значение);
- Initial Delay – начальная задержка (в единицах числа входных импульсов);
- Threshold – порог чувствительности по входу;
- Output Mode – режим выхода: прямоугольный импульс длительностью, равной начальной задержке (Rectangular Pulses), или импульс с длительностью входного (Impulse Output).



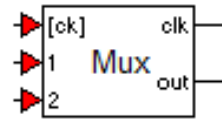
**D Flip Flop** – D-триггер с параметрами:

- Initial State – начальное состояние (0 или 1);
- Threshold – порог чувствительности (0,5);
- Edge Mode – режим запуска: по переднему (Rising Edge) или заднему (Falling Edge) фронту входного сигнала.



**JK Flip Flop** – JK-триггер с набором параметров

аналогичным набору для блока D Flip Flop.



**Mux/Demux** – мультиплексор/демультиплексор; выводы:

[ck], clk – тактовые вход и выход; out – выход; 1, 2, ..., n – информационные входы; параметры:

- Mode – выбор режима мультиплексора (Multiplexer) или демультиплексора (Demultiplexer);

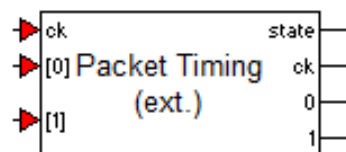
- Number of Lines – число входов в режиме мультиплексора (Multiplexer Mode) или число выходов в режиме демультиплексора (Demultiplexer Mode);

- Initial Position – начальное состояние мультиплексора (или демультиплексора); определяет выбранный канал до прихода первого тактового импульса;

- Timing – выбор типа синхронизации: внутренняя (External) или внешняя (Internal);

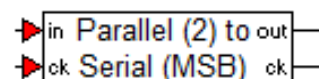
- Switch Rate – скорость переключения, Гц;

- Start Time – время старта при внешней синхронизации, с; определяет задержку после поступления импульса синхронизации.



**Packet Timing** – блок пакетной синхронизации

генерирует синхросигналы, для использования при формировании пакетов фиксированной длины или пакетных структур переменной длины.

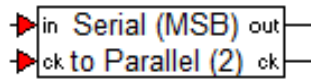


**Parallel to Serial** – преобразователь потока символов

в последовательный поток битов; параметры:

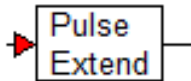
- Bit Order – порядок преобразования входных символов: начиная с их младшего бита (LSB First) или старшего (MSB First);

- Bits per Symbol – количество бит во входных символах (2–16);
- Output Bit Rate – скорость следования входных символов (битрейт).



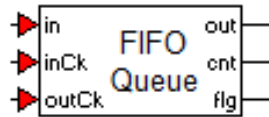
**Serial to Parallel** – преобразователь потока символов

из последовательного в параллельный; набор параметров аналогичен блоку Parallel to Serial.



**Pulse Extend** – блок расширения импульса; используется,

чтобы увеличить продолжительность тактового импульса до требуемого интервала, представленного количеством шагов моделирования или временем в секундах. Этот блок сохраняет амплитуду исходного входного импульса.



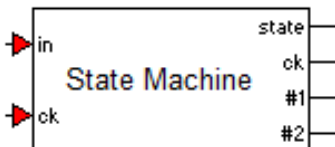
**Queue** – блок организации очереди; выводы: in – вход;

inCk, outCk – тактовые входы для ввода данных в очередь и вывода из очереди;

out – выход; cnt – счетчик очереди; flg – флаг переполнения; параметры:

- Queue Size – размер буфера (1 – 32766);
- Empty Output – величина, определяющая конец очереди;
- Queue Type – выбор типа очереди: FIFO (first in first out – первым вошел и первым вышел) или LIFO (last in first out – последним вошел и первым вышел; такие устройства известны также под названием «стек»).

Значения сигналов на выходе flg: 0 – нормальное состояние буфера; 1 – переполнение в процессе моделирования; 2 – переполнение при сдвиге.

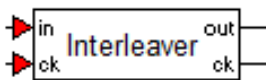


**State Machine** – блок фиксации состояний

целочисленных входных сигналов. Выводы: in – вход; ck – вход и выход синхронизации; state – текущее состояние; #1, #2, ..., #10 – выходы прогнозируемых состояний. Параметры:

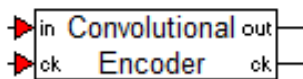
- Number of Inputs – число  $k$  значений входных дискретных сигналов (0 – 99999);
- Number of Outputs – число выходов  $n < 10$ ; задание  $n$  с помощью глобальной переменной не допускается;
- Number of States – число внутренних состояний (0 – 99999);
- Select File – выбор пути (в строке State Map File Path или кнопкой Browse File) к файлу с указанными выше параметрами; для редактирования содержимого файла может быть использован редактор Notepad.

### П1.11. Кодеры/Декодеры (Comm/(Encode/Decode))



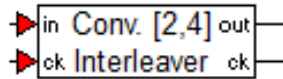
**Interleaver** – блок перемежения (чередования) или деперемежения (дечередования) битовых последовательностей; параметры:

- Mode – режимы работы блока: перемежение (Interleave) или деперемежение (Deinterleave);
- Rows ( $R$ ), Columns ( $C$ ) – число строк и столбцов матрицы (2 – 16383); произведение  $R \times C = 4 - 32767$ .



**Convolutional Encoder** – сверточный кодер; выводы: in, out – вход и выход; ck – вход и выход синхросигнала; параметры:

- Input Bits – число входных битов  $k$  (1 – 7);
- Coded Bits – число выходных битов  $n$  (1 – 8);
- Constraint Length – длина  $L$  битовой последовательности, определяемая размером  $k$  внутреннего буфера;
- Input Bit Rate – скорость битовой последовательности на входе, Гц;
- Generator Coefficients – значение образующего коэффициента в восьмеричном формате для каждого из восьми возможных состояний генератора кода.



**Convolutional Interleaver** – блок чередования или

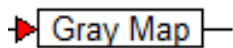
дечередования битовых последовательностей; рекомендуется использовать совместно с кодером Convolutional Encoder и декодерами Viterbi Decoder (Hard) или Viterbi Decoder (Soft); выводы: in, out – вход и выход информационных сигналов (0, 1); ck – тактовые вход и выход; параметры:

- Mode – опциональный выбор режима работы: перемежение (Interleave) или деперемежение (Deinterleave);
- Rows – число строк матрицы перемежения;
- Row Increment – приращение номера ячейки при переходе от одной к другой строке.

Например, перемежение в режиме [3, 2] означает, что глубина перемежения Depth = 3, а приращение Increment = 2; в этом случае входная и выходная последовательности могут иметь вид:

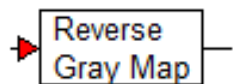
Input: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19 ...

Output: 1, 0, 0, 4, 0, 0, 7, 2, 0, 10, 5, 0, 13, 8, 3, 16, 11, 6, 19, 14, 9 ...

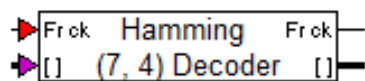


**Gray Map** – кодирование входных сигналов с

предварительным приведением их к целочисленным значениям; при этом на выходе кодера выделяются символы, которые отличаются только одним битом; например, если на входе имеется поток символов 0, 1, 2, 3, 4, 5, 6, 7, то на выходе блока будет последовательность 0, 1, 3, 2, 6, 7, 5, 4.

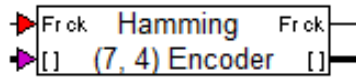


**Gray Reverse Map** – «инверсный» вариант Gray Map.



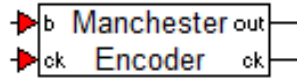
**Hamming Decoder** – декодер Хемминга; выводы:

[ ] – информационные вход ( $n$ ) и выход ( $k$ ); Fr ck – тактовые вход и выход; параметры: Hamming Code Size – размер кода в форме  $(n, k)$ , где  $n$  – общее число бит,  $k$  – число информационных бит;  $(n - k)$  – число бит для проверки на четность; в диалоговом окне можно задать следующие коды Хемминга: (3, 1), (7, 4), (15, 11), (31, 26), (63, 57), (127, 120) и (255, 247).



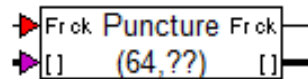
**Hamming Encoder** – кодер Хемминга; выводы:

[ ] – информационные вход ( $k$ ) и выход ( $n$ ); Fr ck – тактовые вход и выход; параметры аналогичны блоку Hamming decoder.



**Manchester Encoder** – кодер манчестерского кода;

выводы: b – входной поток битов; out – выход; ck – вход и выход синхросигнала; параметры: Input Rate – определяет входную скорость передачи для блока в бит/с; Output Mid-Period Clock Pulse – генерация выходных тактовых импульсов на границах бита и в середине (точке перехода) импульса; Output Mode – количество уровней сигнала.



**Puncture** – сжатие входных данных по заданному

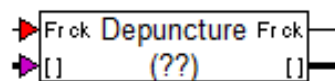
алгоритму (с использованием трафарета); выводы: [ ] – входной  $N$  и выходной  $M$  потоки данных; Fr ck – вход и выход тактовых сигналов выборок (фреймов), период следования которых определяет глубину анализа входных данных; параметры:

- Number of Encoder Outputs – число потоков данных на выходе кодера; используется для расчета эффективной скорости кода при заданном шаге анализа входного потока (так называемый шаг прокалывания);

- Input Vector Size – размер  $N$  буфера входных данных (1 – 100000); определяет период прокалывания;

- Compute Output Vector Size – размер выходного вектора сжатых данных;

- Select File – вызов файла с алгоритмом сжатия (Puncture Pattern) при необходимости его редактирования (в редакторе Notepad).

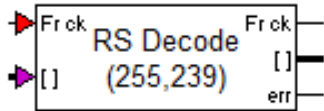


**Depuncture** – восстановление данных, сжатых

блоком Puncture; выводы: [ ] – входной  $M$  и выходной  $N$  потоки данных; Fr ck – вход и выход тактовых сигналов выборок (фреймов); параметры:



- Number of Encoder Outputs – число потоков закодированных данных;
- Output Vector Size – объем (размер)  $N$  выходного буфера (1 – 100000);
- Compute Input Vector Size – размер  $M$  входного вектора данных;
- Select File – выбор пути к файлу с настройками (в строке Puncture Pattern File Path или кнопкой Browse File).



**Reed-Solomon Decoder** – декодер Рида-Соломона;

выводы: [ ] – входной  $n$  и выходной  $k$  поток данных; Fr ck – вход и выход тактовых сигналов декодирования выборок; err – признак ошибки декодирования (сигнал уровня – 1); параметры:

- Symbol Size – количество  $M$  бит в символе; определяется как  $2^m - 1$ , где  $M = 3 - 10$ ;
- Information Symbols – количество  $k$  информационных битов;
- Coded Symbols – количество  $n$  бит на входе декодера. Как видно из значка блока,  $n = 255$  и  $k = 239$ . Такой код обозначается как RS(255, 239); он содержит  $n - k = 255 - 239 = 16$  проверочных бита, поэтому его называют также CRC-16 (CRC – Cyclic Redundancy Check – циклический избыточный код); он способен обнаружить  $(n - k)$  ошибок и  $(n - k)/2$  из них исправить;
- Advanced Settings – кнопка расширенных параметров настройки; позволяет выбрать параметр First root of generator polynomial – первый член ( $b_0$ ) образующего многочлена; значение по умолчанию – 1.

Образующие многочлены, используемые рассматриваемым RS блоком при различных значениях  $M$ , имеют следующий вид:

для  $M = 3$ :  $1 + x + x^3$ ;

$M = 4$ :  $1 + x + x^4$ ;

$M = 5$ :  $1 + x^3 + x^5$ ;

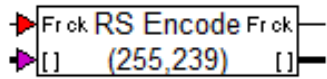
$M = 6$ :  $1 + x + x^6$ ;

$M = 7$ :  $1 + x^3 + x^7$ ;

$M = 8$ :  $1 + x^2 + x^3 + x^4 + x^8$ ;

$M = 9$ :  $1 + x^4 + x^9$ ;

$M = 10$ :  $1 + x^3 + x^{10}$ .



**Reed-Solomon Encoder** – кодер Рида-Соломона;

обозначения выводов и набор параметров аналогичен блоку Reed-Solomon Decoder с тем отличием, что на его вход поступает блок из  $k$  информационных битов, а на выходе формируется  $n$  битов, представляющих сумму информационных и проверочных битов, значение которых определяется путем деления входного кода (в алгебраической форме) на соответствующий образующий многочлен. На приемной стороне в декодере выполняется такая же операция с использованием такого же образующего многочлена, поэтому сравнение полученного результата с проверочными битами позволяет обнаружить  $(n - k)$  ошибок и  $(n - k)/2$  из них исправить.



**Trellis Encoder** – решетчатый кодер; выводы: in – вход,

ck – тактовые вход и выход, I, Q – составляющие выходного комплексного сигнала (I, Q); используется в сочетании с блоком I/Q Modulator; параметры:

- Input Bits – число битов входных данных  $k$  (1 – 7);
- Output Bits – число битов выходных данных  $n$  (1 – 8);
- Number of States – число состояний сигнально-кодовой конструкции;
- Select File, Browse File – выбор файла настроек и пути к нему.



**Trellis Decoder** – решетчатый декодер; выводы: I, Q –

входы комплексного сигнала (I, Q), ck – тактовые вход и выход, out – выход; набор параметров аналогичен блоку Trellis Encoder, за исключением параметра Trellis Truncation Length – длина усечения декодированной решетки с предельным значением в 100 точек сигнального созвездия.



**Viterbi Decoder (Hard)** – жесткое (окончательное)

декодирование сигналов, полученных с помощью Convolutional Encoder;

выводы: in, out – вход и выход битового потока (0, 1); ck – тактовые вход и выход; Mtrc – сигнал метрики (расстояние Хемминга); параметры:

- Input Bits – число информационных битов  $k$  (1 – 7);
- Coded Bits – число закодированных битов  $n$  (1 – 8);
- Constraint Length – длина  $L$  последовательности, определяемая размером  $k$  внутреннего буфера; максимальное значение произведения  $k \times L = 15$  при объеме  $L = 1$ ;
- Output Bit Rate – скорость битовой последовательности на выходе, Гц;
- Trellis Truncation Length – длина усечения решетки ( $M$ ) в словах размера  $k$  ( $k \times M \leq 96$ );
- Generator Coefficients – значение образующего коэффициента в восьмеричном формате для каждого из восьми возможных генераторов кода.



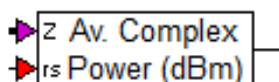
**Viterbi Decoder (Soft)** – вариант Viterbi Decoder для

мягкого (предварительного) декодирования; параметры:

- No. Information Bits – число информационных битов  $k$  (1 – 7);
- No. Coded Output Bits – число закодированных битов  $n$  (1 – 8);
- Quantization Bits – число битов квантования, используемых в процессе декодирования;
- Metric File Path – определение пути к файлу с алгоритмом поиска оптимальной метрики декодирования.

Другие параметры аналогичны блоку Viterbi Decoder (Hard).

## П1.12. Измерительные преобразователи (Comm/Estimators)



**Average Power (Complex)** – блок измерения средней

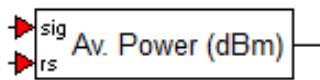
мощности комплексного сигнала, поступающего на вход  $z$  ( $rs$  – вход сброса логическим сигналом 1); параметры:

- Load – выбор сопротивления нагрузки: 1 или 50 Ом;

- Output Units – выбор единицы измерения мощности: дБм (dBm), ватт (Watts) или дБВт (dBW);
- Mode – выбор режима измерения; с момента запуска моделирования (Running) или с использованием окна с заданным количеством шагов модельного времени (Sliding Window);
- Window Size – количество шагов модельного времени (только для режима Sliding Window);
- Shift Reg. Initial Value – начальное состояние регистра буфера перед запуском моделирования (только для режима Sliding Window).

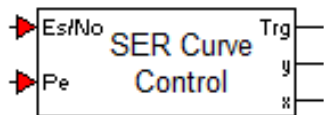
Выходной сигнал блока:  $y_k = \frac{1}{N} \sum_{i=k-N-1}^k x_{li}^2$  где  $k$  – общее число шагов

модельного времени;  $x_{li} = x_i - M$  – центрированное значение входного сигнала на  $i$ -м шаге моделирования;  $x_i$ ,  $M$  – значение входного сигнала на  $i$ -м шаге моделирования и его среднее значение;  $N$  – размер окна в шагах модельного времени или число выборок (реализаций) в режиме Running.



**Average Power (Real)** – блок аналогичен блоку

Average Power, но для действительных (скалярных) сигналов.

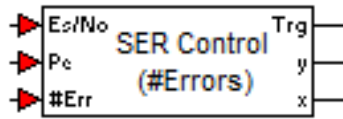


**BER Curve Control** – блок управления индикацией

ошибок передачи битов BER (Bit Error Rate) или (по выбору) символов SER (Symbol Error Rate), определяемых в результате анализа данных, полученных при заданном количестве пусков (измерений); выводы: Trg – сигнал синхронизации для построения BER-осциллограммы, y, x – данные для BER-осциллограммы по ошибкам и скорости передачи (скорость учитывается в логарифмическом масштабе); Pe – признак ошибочных битов; параметры:

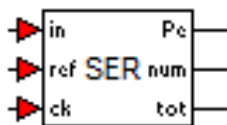
- Number of Runs – количество пусков (1 – 10);
- Mode – выбор режима работы: измерение ошибки в битах (Bit Error Rate, соответствует опции Eb/No блока AWGN) или в символах (Symbol Error Rate, опция Es/No в блоке AWGN);

- Duration – продолжительность каждого пуска, с;
- Suppress Result Notification – опция запрета вывода на дисплей результатов каждого пуска;
- Show Results – кнопка вывода результатов измерения.



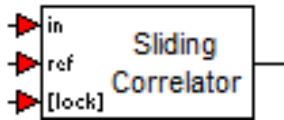
**BER Control (#Errors)** – блок контроля количества ошибок передачи битов BER (Bit Error Rate) или (по выбору) символов SER (Symbol Error Rate), определяемых в результате анализа данных, полученных при заданном количестве пусков (измерений); выводы: Trg – сигнал синхронизации для построения BER-осциллограммы; y, x – данные для BER-осциллограммы по ошибкам как функция отношения сигнал/шум  $E_b/N_0$ ;  $P_e$  – вход сигналов с блока; #Err – вход сигнала зарегистрированной ошибки; параметры:

- Number of Runs – количество пусков (1 – 10);
- Mode – выбор режима работы: измерение ошибок в битах (Bit Error Rate, соответствует опции  $E_b/N_0$  блока AWGN ) или в символах (Symbol Error Rate, опция  $E_s/N_0$  в блоке AWGN);
- Max Errors – максимальное число ошибок для каждого выполненного пуска;
- Show Results – отображение полученного результата;
- Notification – запрет вывода данных в конце каждого пуска.



**Bit/Symbol Error Rate** – измеритель количества ошибочных битов BER при передаче отдельной битовой последовательности (режим BER) или символов (режим SER) путем сравнения восстановленного потока данных (вход in) с опорным (образцовым) (вход ref), синхронизированных по тактовому входу ck; выходы:  $P_e$  – признак ошибочных битов (для блока BER Curve Control); tot, num – общее количество битов или символов (в режиме SER) и количество ошибочных битов; параметры:

- Count Start Delay – начальная задержка в единицах числа символов;
- Output-Mode – режим работы: анализ потока символов (опция Symbol Error Rate) или битов (опция Bit Error Rate);
- Bits per Symbol – количество бит в символе (только для режима Bit Error Rate) (не более 8).

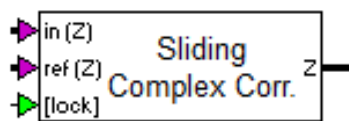


**Correlation (Real)** – вычисление взаимной корреляции и автокорреляции сигналов на входах in и ref с управлением по входу [lock]; параметры:

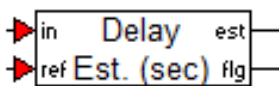
- Mode – режимы работы блока: стандартный (Standard) и так называемый пропущенный режим (Gated Control), когда анализ проводится при значении сигнала на входе ref, запомненного в буфере блока в момент перехода сигнала на входе [lock] с 1 в 0;
- Window Size – размер буфера корреляций в шагах моделирования.

Выходной сигнал блока  $y_k = \frac{1}{N} \sum_{i=0}^{N-1} x_{1(k-i)} x_{2j}$ , где  $j = k - i$  – для режима

Standard (измерение взаимной корреляции между сигналами на входах in и ref);  $j = j_0 - i - k$  – для режима Gated Control (измерение автокорреляции сигнала на входах in и ref с запоминанием сигнала на входе ref в течение  $j_0$  шагов модельного времени).

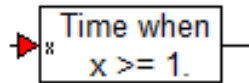


**Correlation (Complex)** – блок аналогичен блоку Correlation (Real), но для комплексных сигналов.

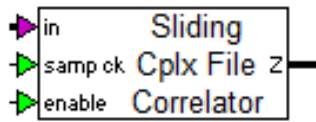


**Delay Estimator** – оценка временной задержки распространения исследуемого сигнала (вход in) относительно опорного (вход ref); выходы: est – временная задержка, flg – флаг состояния процесса моделирования (0 – в начале и в процессе моделирования и 1 – в конце); параметры:

- Window Size – размер окна корреляции (в единицах модельного времени);
- Start Time – время начала корреляционного анализа, с;
- Max. Delay – максимально возможное время измеряемой задержки, с.

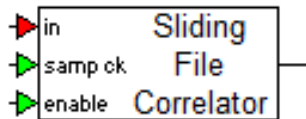


**Event Time** – таймер, фиксирующий момент времени от начала моделирования, при котором входной сигнал  $x$  достигает значения, соответствующего условиям, задаваемым оператором Event Mode ( $=$ ,  $\geq$ ,  $\leq$ ,  $>$  или  $<$ ) и значением входной величины Threshold Value (по умолчанию 1).

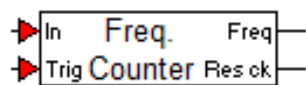


**File Correlation (Cplx)** – вычисление взаимной корреляции между входным комплексным сигналом на входе  $in$  и фиксированным опорным сигналом определенным через внешний файл.

Взаимная корреляция выполняется по подвижному окну размером  $N$ , где  $N$  – определяемый пользователем параметр. Выборка значений из файла может быть загружена или в порядке возрастания или в порядке убывания.



**File Correlation (Real)** – блок аналогичен блоку File Correlation (Cplx), но для действительных (скалярных) сигналов.

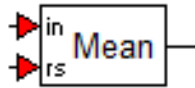


**Frequency Counter** – цифровой частотомер, подсчитывающий количество переходов между уровнями сигнала (с низкого на высокий, с высокого на низкий) на определенном интервале времени; выводы:  $in$  – входной сигнал,  $Trig$  – сигнал триггера,  $Freq$  – значение частоты,  $Res\ ck$  – синхроимпульсы; параметры:

- Time Span – измерительный интервал в режиме Time Duration;
- High Threshold – значение напряжения входного сигнала соответствующее высокому уровню;

– Low Threshold – значение напряжения входного сигнала соответствующее низкому уровню;

– Time Span Mode – определение интервала измерений; при External Enable измерительным интервалом управляют внешне через вход Trig (при  $> 0,5$  начало, при  $< 0,5$  окончание измерения); при Time Duration интервал определен параметром Time Span.



**Mean** – блок измерения среднего значения входного сигнала (вход in); rs – вход сброса сигналом 1; параметры:

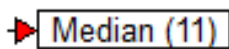
– Mode – выбор режима измерения: с момента запуска моделирования (Running) или с помощью подвижного окна с заданным количеством шагов  $N$  модельного времени (Sliding Window);

– Window Size – количество шагов (для режима Sliding Window);

– Shift Reg. Initial Value – начальное состояние регистра буфера перед запуском моделирования (только для режима Sliding Window).

Выходной сигнал блока:  $y_k = \frac{1}{N} \sum_{i=k-N+1}^k x_i$ , где  $k$  – общее число шагов

модельного времени;  $x_i$  – значение входного сигнала на  $i$ -м шаге моделирования;  $N$  – размер окна в шагах модельного времени или число выборок (реализаций) в режиме Running.

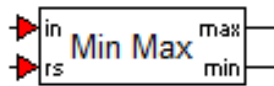


**Median** – вычисление медианы входного сигнала путем сортировки исходных данных в восходящем порядке; параметры:

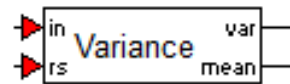
– Window Size – размер  $N$  подвижного окна; при четном  $N$  возвращается среднее из двух средних значений меньшей и большей половины, при нечетном – самое близкое к середине данных (геометрически медиана определяется как абцисса точки, в которой площадь, ограниченная кривой распределения, делится пополам);

– Shift Register Initial Value – начальное состояние сдвигового регистра.



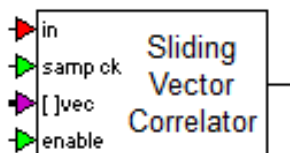


**MinMax** – этот блок выводит минимальные и максимальные значения входного сигнала с начала моделирования или после сигнала сброса; выходы: in – входной сигнал, rs – вход сброса логическим сигналом 1, max – максимальное значение, min – минимальное значение. У этого блока нет внутренних параметров.



**Variance** – определение дисперсии (выход var) и среднего значения (выход mean) входного сигнала (вход in); rs – вход сброса логическим сигналом 1. Выходной сигнал блока

$$y_k = \frac{1}{N} \sum_{i=k-N-1}^k x_i^2 - \left( \frac{1}{N} \sum_{i=k-N-1}^k x_i \right)^2.$$



**Vector Correlation** – этот блок выполняет оценку взаимной корреляции между входным сигналом in и внешним вектором данных [ ]vec. Корреляция выполняется по раздвижному окну размера N, где N представляет размер внешнего вектора данных, а также определяемый пользователем параметр.

В целях повышения эффективности используется вход enable (0, 1) для определения когда вычисление корреляции выполняется (при уровне  $\geq 1$ ). Вход samp ck – импульсный синхросигнал (0, 1) для входного сигнала.

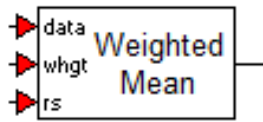
Выходной сигнал блока

$$y_k = \sum_{i=0}^{N-1} (x_{1(k-1)} \times x_3[i]),$$

где N – параметр Window Size,

$x_3[i]$  – i-й элемент вектора.

Window Size – размер подвижного окна; это значение должно соответствовать размеру внешнего вектора данных.



**Weighted Mean** – блок вычисления взвешенного

среднего значения входного сигнала; входы: data, whgt – входы контролируемого и «взвешивающего» сигналов; rs – сигнал сброса; параметры:

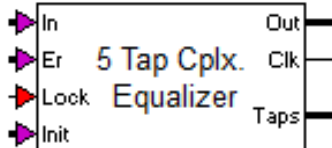
- Mode – работа в режиме Sliding Window или с учетом всех выборок моделирования после последнего сброса или начала моделирования (Running);
- Window Size – размер буфера подвижного окна в шагах моделирования; параметр доступен только в режиме Sliding Window.

Выходной сигнал блока

$$y_k = \left( \sum_{i=k-N-1}^k x_i w_i \right) / \left( \sum_{i=k-N-1}^k w_i \right),$$

где  $w_i$  – значение «взвешивающего» сигнал на входе whgt.

### П1.13. Фильтры (Comm/Filters)

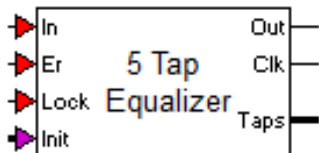


**Adaptive Equalizer (Cplx)** – адаптивный фильтр

(эквалайзер) типа FS (Fractionally Spaced – дробно пространственный) для комплексных сигналов; входы: Out – выход; Clk – сигнал синхронизации; Taps – векторный выход; параметры:

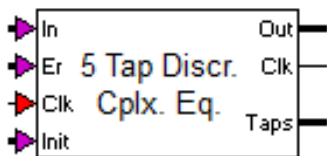
- Number of Taps – число выборок (1 – 32767);
- Taps per Symbol – число выборок на символ;
- Mu – коэффициент обратной связи (коэффициент сходимости) при отработке сигнала ошибки на входе er в процессе адаптации (типичное значение 0,005);
- Symbol Rate – скорость следования символов на входе in, Гц;
- Input Time Delay – временная задержка сигнала ошибки относительно входного; служит для синхронизации работы сравнивающего элемента в режиме Internal Timing;.

- Initialized Tap Value – начальное значение выборки в режиме Internal Tap Initialization; при внешней инициализации (вход init) используется нулевое значение;
- Initialized Tap Offset – смещение от центра выборки инициализации Initialized Tap Value;
- Show Taps – кнопка для вывода текущего значения выборки;
- Timing – опции выбора типа синхронизации: внешняя (External) (вход Lock при уровне сигнала  $> 0,5$ , при уровне  $\leq -1$  производится сброс) или внутренняя (Internal);
- Tap Initialization – режим выбора инициализации: внешняя (External) (вход init) или внутренняя (Internal) с учетом параметра Initialized Tap Offset.



**Adaptive Equalizer (Real)** – вариант фильтра

Adaptive Equalizer для скалярных сигналов.



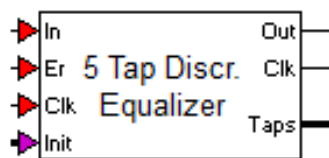
**Discrete Equalizer (Cplx)** – адаптивный дискретный

фильтр (эквалайзер) типа FS (Fractionally Spaced – дробно интервальный) для комплексных сигналов. Это блок подобен блоку Adaptive Equalizer за исключением того, что в нем не используется внутренняя задержка в регистре сдвига. Этот блок использует алгоритм сходимости Least Mean Square (LMS, метод наименьших квадратов), чтобы подстроить значения выборок с целью уменьшения средней погрешности.

Выводы: in – входной сигнал; Er – сигнал ошибки; Clk – сигнал синхронизации/блокировка ввода (высокий уровень  $> 0,5$ , последовательность импульсов) и сигнал сброса ( $\leq -1$ , импульс); Init – вход внешней инициализации (вектор); Out – выходной сигнал; Taps – векторный выход; параметры:

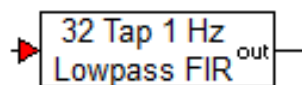
- Number of Taps – число выборок (1 – 32767);

- $\mu$  – коэффициент обратной связи (коэффициент сходимости) при обработке сигнала ошибки на входе  $e[n]$  в процессе адаптации (типичное значение 0,005);
- Sample Rate – определяет входную частоту дискретизации эквалайзера в Герцах для режима Internal Timing Mode;
- Input Time Delay – временная задержка сигнала ошибки относительно входного; служит для синхронизации работы сравнивающего элемента в режиме Internal Timing;
- Initialized Tap Value – начальное значение выборки в режиме Internal Tap Initialization; при внешней инициализации (вход init) используется нулевое значение;
- Initialized Tap Offset – смещение от центра выборки инициализации Initialized Tap Value;
- Tap Spacing (1 – 10) – определяет интервал между последовательными выборками эквалайзера;
- Show Taps – кнопка для вывода текущего значения выборки;
- Timing – опции выбора типа синхронизации: внешняя (External) или внутренняя (Internal);
- Tap Initialization – режим выбора инициализации: внешняя (External) (вход Init) или внутренняя (Internal) с учетом параметров Initialized Tap Value и Initialized Tap Offset.



**Discrete Equalizer (Real)** – вариант фильтра Discrete

Equalizer для скалярных сигналов.



**FIR** – фильтр типа FIR (Finite Impulse Response – с

конечной импульсной характеристикой); параметры:

- Filter Type – выбор типа фильтра по виду АЧХ: низкочастотный (Lowpass), высокочастотный (Highpass), полосовой (Bandpass), заградительный

(Bandstop), косинусно нарастающий (Raised Cosine), усечённый косинусно нарастающий (Root Raised Cosine), Гильберта (Hilbert) или Гаусса (Gaussian);

- Rolloff Factor – параметр для фильтров Raised Cosine и Root Raised Cosine (0 – 1); нулевое значение выбирается для случая усеченной  $\sin(x)/x$  импульсной характеристики;

- Number of Taps – число выборок (1 – 32767); для высокочастотных (Highpass) и заградительных (Bandstop) фильтров это число должно быть нечетным;

- Cutoff Freq 1 – частота отсечки для фильтров lowpass, raised cosine, root raised cosine, hilbert, gaussian или highpass; для фильтров bandpass и bandstop рекомендуется выбирать более низкое значение этой частоты; частота отсечки для фильтров raised cosine, root raised cosine и gaussian определяется по уровню 3 дБ, для остальных – по уровню 6 дБ;

- Cutoff Freq 2 – частота отсечки (по уровню 6 дБ) для фильтров bandpass и bandstop;

- Window Type – опциональный выбор оконного режима; в случае выбора режима Kaiser можно выбрать значение параметра Beta;

- Units – выбор единиц измерения частоты: Гц (Hertz) или рад/с (Radians/Sec);

- Normalize LPF to 0 dB at dc – нормализация импульсной характеристики относительно ее значения на постоянном токе (0 дБ); используется в случае небольших значений частоты и числа выборок для фильтров lowpass, raised cosine, root raised cosine, hilbert и gaussian;

- Tap Spacing Frequency – частота следования выборок, Гц; установка параметра доступна после активизации опции Allow FIR Decimation;

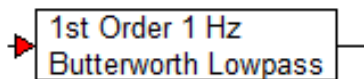
- View Response – кнопка для просмотра отклика (реакции) фильтра;

- Show Taps – кнопка для вывода текущих значений выборок, которые могут быть сохранены в файле после нажатия кнопки Save To File.



**File FIR** – фильтр, конструируемый пользователем на основании выборок (до 32767), представленных файлом; параметры:

- Select File, Browse File – кнопки выбора файла и пути к нему;
- Tap Spacing Frequency – частота выборок, Гц; должна быть согласована с модельным временем;
- Taps File Path – путь к файлу.



**IIR** – фильтры типа IIR (Infinite Impulse Response

– с бесконечной импульсной характеристикой), являющиеся цифровыми аналогами фильтров Чебышева, Баттерворта и др.; параметры:

- Filter Method – выбор алгоритма фильтра: эллиптического (Elliptic), Чебышева двух типов (Chebyshev Type I and Type II), Баттерворта (Butterworth) или Бесселя (Bessel);

- Filter Type – выбор типа фильтра: низкочастотный (Lowpass), высокочастотный (Highpass), полосовой (Bandpass) или заградительный (Bandstop);

- Filter Order – порядок фильтра (1 – 20); для фильтров Bandpass и Bandstop он должен быть четным;

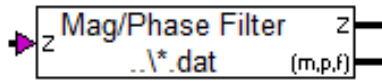
- Cutoff Freq 1 – частота отсечки для фильтров Lowpass и Bandpass; для Bandpass и Bandstop рекомендуется выбирать более низкое значение этой частоты;

- Cutoff Freq 2 – верхняя частота отсечки (среза) (по уровню 6 дБ) для фильтров bandpass и bandstop;

- Passband Specification – выбор спецификации полосы пропускания: при заданных параметром Epsilon относительном или ослаблении в децибелах Ripple;

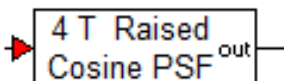
- Stopband Attenuation – затухание в полосе пропускания для фильтров Chebyshev Type II и Elliptic;

- Units – выбор единиц измерения частоты: Гц (Hertz) или рад/с (Radians/Sec);
- Show Coeff. – кнопка визуализации полиномиальных коэффициентов числителя и знаменателя для фильтров выше второго порядка;
- View Response – кнопка для просмотра отклика (реакции) фильтра.



**MagPhase** – фильтр типа FIR с промежуточными прямым и обратным БПФ (FFT) с параметрами, заданными в файле; выходы: z – вход и выход комплексного сигнала; (m,p,f) – векторный выход амплитуды, фазы и частоты; параметры:

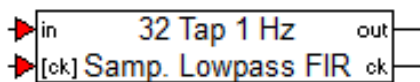
- Equivalent FIR Filter Tap Length – эквивалентная длина импульсной передаточной функции фильтра;
- Phase Units – единицы измерения фазы: в градусах (Degrees), в единицах группового времени задержки (Group Delay) или радианах (Radians);
- Filter File Data – данные для файла в виде  $[0, fs/2]$  – для реального и  $[-fs/2, +fs/2]$  – для комплексного представления;
- Add Linear Phase – опция автоматического увеличения на 0,5 длительности импульсной передаточной функции фильтра;
- Normalize Phase – автоматическая нормализация фазовой характеристики из условия нулевой фазы в точке нулевой частоты; опция активна при выборе режима Group Delay;
- Select File, Browse File, Filter File Path – команды вызова и загрузки файла с данными.



**Pulse Shaping Filter** – формирователь последовательности импульсов требуемой формы на принципе FIR. Параметры:

- Pulse Span – размер интервала формирования импульсов в единицах периода выборок  $T$ ; например, в случае интервала  $4T$  выборки будут располагаться на интервале  $[-2/T, 2/T - t]$ , где  $t$  – шаг моделирования;
- Samples per Symbol – число выборок на входной символ;

- Rolloff Factor – параметр формы (альфа) (0 – 1);
- Beta – параметр формы окна типа Kaiser;
- BT Product – параметр гауссова фильтра;
- Filter Type – выбор из предлагаемого списка типа;
- Window Type – тип окна для выбранного типа фильтра;
- Apply Inverse Sine – опция инверсии Sine функции в случае прямоугольного импульса;
- Show Taps – отображение формируемого сигнала в режиме увеличивающейся задержки; при удовлетворительных результатах они могут быть сохранены в файле (кнопка Save To File);
- View Response – кнопка для просмотра отклика (реакции) фильтра, эквивалентного формируемому сигналу.



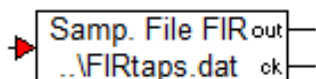
**Sampling FIR** – фильтр на основе

использования отбора образцов (сэмплов) входного сигнала по заданному алгоритму и их размножение с помощью сдвигового регистра; выводы: in, out – вход и выход сигнала; [ck], ck – тактовые вход и выход; параметры:

- Number of Taps – число выборок (1 – 32767); для фильтров типа highpass и bandstop это число должно быть нечетным;
- Cutoff Freq 1 – частота среза для фильтров типа Lowpass, Raised Cosine, Root Raised Cosine, Hilbert, Gaussian и Highpass; для фильтров типа Bandpass и Bandstop ее значение выбирается более низким;
- Cutoff Freq 2 – верхнее значение частоты среза для фильтров типа Bandpass и Bandstop;
- Window Type – выбор типа окна; при выборе Kaiser окна активизируется опция Beta (параметр формы);
- Units – выбор единиц измерения частоты: Гц (Hertz) или рад/с (Radians/Sec);
- Rolloff Factor – коэффициент формы косинусоидального фильтра (0 – 1);

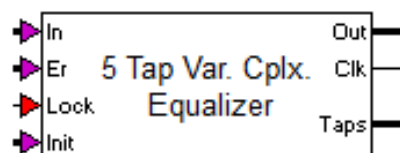


- Filter Type – выбор типа фильтра: lowpass, highpass, bandpass, bandstop, raised cosine, root raised cosine, Hilbert, Gaussian; при выборе фильтров Raised Cosine или Root Raised Cosine необходимо выбрать параметр Rolloff Factor; Normalize LPF to 0 dB at dc – нормализация импульсной передаточной функции фильтра к единичному значению (0 дБ) на нулевой частоте;
- Sampling Frequency – частота дискретизации; должна быть кратна частоте (шагу) моделирования;
- Initial Delay – начальная задержка фильтра в шагах моделирования или секундах (в зависимости от параметра Delay Mode);
- Delay Mode – выбор задержки в шагах моделирования (Sim Steps) или секундах (Seconds);
- Output Mode – эта опция выбирается в случае, когда частота дискретизации кратна частоте (шагу) моделирования;
- Interpolated – использование линейной интерполяции при формировании передаточной характеристики фильтра;
- Held – отображение интерполированных точек;
- Show Taps – отображение характеристик фильтра в режиме увеличивающейся задержки.



**Sampling File FIR** – фильтр, конструируемый

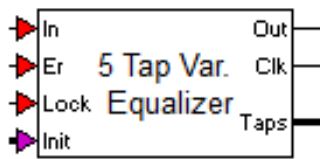
пользователем на основании выборок (до 32767), представленных в файле.



**Var. Adaptive Equalizer (Cplx)** – адаптивный

фильтр (эквалайзер) типа FS (Fractionally Spaced – дробно интервальный) для комплексных сигналов подходящий для использования с аналоговыми и цифровыми сигналами. В отличие от блока Adaptive Equalizer (Cplx), эта версия позволяет пользователю устанавливать неравномерный интервал между выборками фильтра.

Этот блок использует алгоритм сходимости Least Mean Square (LMS, метод наименьших квадратов), чтобы подстроить значения выборок с целью уменьшения средней погрешности.

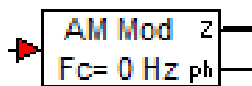


**Var. Adaptive Equalizer (Real)** – вариант фильтра

Var. Adaptive Equalizer для скалярных сигналов.

### П1.14. Модуляторы типа Complex (Comm/Modulator-Complex)

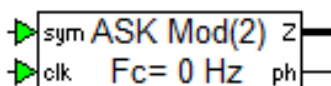
В этом разделе приводятся краткие данные моделей модуляторов с выходным сигналом в комплексной форме.



**AM** (Amplitude Modulation) Modulator – амплитудный

модулятор; выводы:  $z$  – выход амплитудно-модулированного сигнала;  $ph$  – фаза немодулированной несущей, рад; на этом выходе формируется сигнал, равный произведению угловой частоты  $\omega$  и времени моделирования  $t$  (по умолчанию 10 с); параметры:

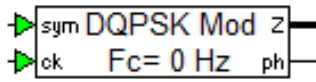
- Translation Frequency, Amplitude, Initial Phase – частота (Гц), амплитуда (В) и начальная фаза (град.) несущей;
- Modulation Factor – коэффициент модуляции;
- Phase Output Mode – режим по выходной фазе: при немодулированной (Unwrapped) или в режиме модулированной (Wrapped) несущей в диапазоне  $[0, 2\pi]$ .



**ASK** (Amplitude Shift Keying) Modulator –

модулятор с амплитудной манипуляцией; выводы:  $sym$  – входной символ (целое число  $[0, \dots, N-1]$ , где  $N$  – размер совокупности),  $clk$  – тактовый сигнал;  $z$  – выход амплитудно-модулированного сигнала;  $ph$  – фаза немодулированной несущей, рад.

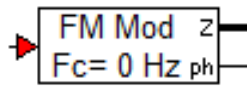
В блоке ASK информация передается путем изменения амплитуды несущего сигнала на дискретных уровнях согласно входному значению данных. Частота несущего сигнала остается постоянной. Этот блок принимает число символа как свой ввод.



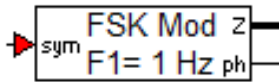
### Differential QPSK (Quadrature Phase Shift Keying)

Modulator – включает семейство дифференциально-квадратурных модуляторов фазоманипулированных сигналов; выходы: sym, ck – входы модулирующего и тактового сигналов; z, ph – выход комплексного модулированного сигнала и фазы несущей; параметры:

- DQPSK Type – выбор типа модулятора PSK: DQPSK (дифференциально-квадратурный 4-позиционный), DBPSK (Differential Binary PSK – дифференциальный бинарный (2-позиционный) PSK), D8PSK, D16PSK и D32PSK (соответственно дифференциальные 8-, 16- и 32-позиционные PSK) или Pi4DQPSK (дифференциально-квадратурный 4-позиционный PSK; отличается от DQPSK смещением фазы на  $45^\circ$  ( $\pi/4$ ));
- Translation Frequency, Amplitude, Initial Phase – частота (Гц), амплитуда (В) и начальная фаза (град.) несущей;
- Gain Imbalance – дисбаланс данных по вещественной оси Q относительно мнимой оси I;
- Constellation Rotation – циклический сдвиг по фазе сигнального созвездия (сигнальных точек) на плоскости (I, Q), град.;
- Phase Imbalance – дисбаланс данных по мнимой оси I относительно вещественной оси Q; например, дисбаланс в  $10^\circ$  означает, что угол между осями I и Q составляет  $80^\circ$  (вместо  $90^\circ$ );
- Phase Output Mode – режим по выходной фазе: при немодулированной (Unwrapped) или в режиме модулированной (Wrapped) несущей в диапазоне  $[0, 2\pi]$ ;
- Select File, Browse File, DPSK File Path – выбор файла настроек и пути к нему.

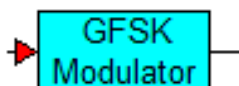


**FM** (Frequency Modulation) – частотный модулятор, на выходе *z* которого формируется сигнал постоянной амплитуды, частота которого меняется в соответствии с уровнем входного (модулирующего) сигнала; выход *ph* – фаза модулированного сигнала, за исключением индекса модуляции (FM Deviation Index вместо Modulation Factor), определяющего изменение (девиацию) частоты выходного сигнала, Гц/В.



**FSK** (Frequency Shift Keying) Modulator – модулятор с частотной (тональной) манипуляцией, в котором каждый уровень входного сигнала преобразуется в сигнал определенной частоты (тона), в частности в случае двоичного сигнала таких частот будет две; используется, например, в некоторых модемах для передачи двоичных (бинарных) последовательностей; параметры:

- Number of Tones – количество тонов;
- Lowest Frequency – наименьшее значение частоты тона, Гц; в случае двоичного сигнала это значение соответствует логическому нулю;
- Frequency Spacing – частотное различие (смещение) между тонами, Гц;
- Amplitude – амплитуда несущей, В;
- Initial Phase – начальная фаза несущей, град.; используется в аналоговом режиме (Continuous Phase – непрерывной фазы);
- Phase Mode – режим работы по фазе: непрерывный (Continuous) или дискретный (Discontinuous), например двоичный (сигнал логической единицы передается частотой Lowest Frequency + Frequency Spacing);
- Phase Output Mode – режим по выходной фазе: при немодулированной (Unwrapped) или в режиме модулированной (Wrapped) несущей в диапазоне  $[0, 2\pi]$ .



**GFSK** Modulator – разновидность FSK модулятора с биполярным (без возврата к нулю) входным сигналом NRZ  $[-1, +1]$ ; содержит

(рис. П1.38) частотный модулятор с девиацией частоты  $FMdev$  (задается одноименной переменной), на входе которого включен фильтр Gaussian FIR с частотой среза  $Cutoff = BT \times SymRate$ , где  $SymRate = R$  – скорость входного потока, симв/с;  $BT$  – задержка в единицах символов; оба параметра задаются одноименными переменными. Кроме того, параметр  $BT$  должен выбираться из условия  $BT = (N - 1)/2$ , где  $N$  – число шагов моделирования. Приведенные значения настроек соответствуют случаю применения блока в системах беспроводной ближней связи (Bluetooth) с девиацией частоты в диапазоне 140 – 175 кГц.

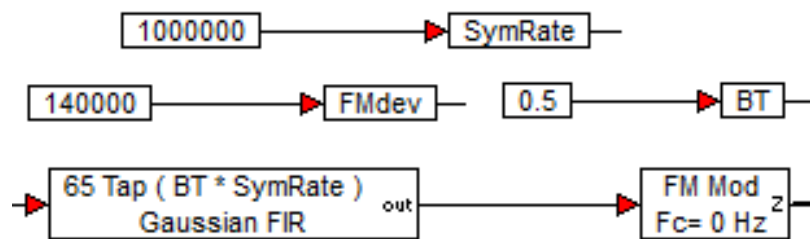
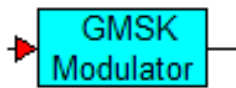


Рисунок П1.38. Функциональная схема GFSK модулятора



**GMSK Modulator** – разновидность MSK модулятора с биполярным входным сигналом NRZ  $[-1, +1]$  (рис. П1.39); отличается от GFSK тем, что девиация частоты частотного модулятора выбирается из условия  $FMdev = 0,25R$ .

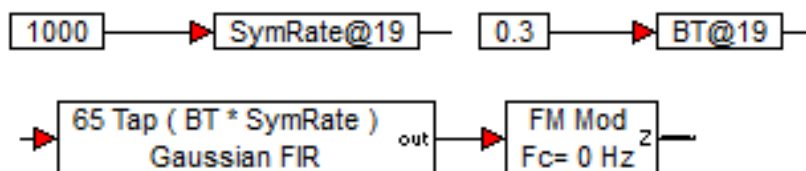
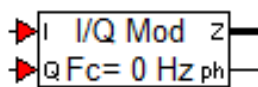


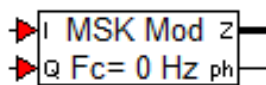
Рисунок П1.39. Функциональная схема GMSK модулятора



**I/Q Modulator** – амплитудно-фазовый модулятор с модуляцией амплитуды и фазы несущей; выходы: I, Q – входы мнимой и вещественной составляющих модулирующего сигнала; z – выход комплексного

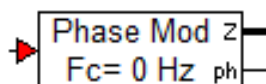
модулированного сигнала;  $ph$  – фаза немодулированной составляющей; параметры:

- Carrier Frequency – несущая частота  $F_c$ , Гц; выбирается равной нулю при комплексном представлении огибающей;
- Amplitude Factor – масштабный коэффициент несущей относительно амплитуды вектора на плоскости (I, Q);
- Initial Phase – начальная фаза модулированной несущей, град.;
- Phase Output Mode – режим по выходной фазе: при немодулированной (Unwrapped) или в режиме модулированной (Wrapped) несущей в диапазоне  $[0, 2\pi]$ .

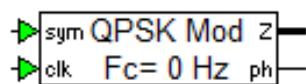


**MSK** (Minimum Shift Keying) Modulator – разновидность FSK модулятора с минимальным сдвигом тональных частот для 1 и 0; используются в высокоскоростных модемах; отличается более низким уровнем энергии боковых лепестков спектра (более узким спектром) по сравнению с QPSK и SQPSK модуляцией; параметры:

- Carrier Frequency, Amplitude – частота (Гц) и амплитуда (В) несущей;
- Constellation Rotation – циклический сдвиг сигнального созвездия на плоскости (I, Q), град.;
- Data Rate – скорость поступления данных, симв/с;
- Data Start Time – начальное время поступления данных, с;
- Phase Output Mode – режим по выходной фазе: при смодулированной (Unwrapped) или в режиме модулированной (Wrapped) несущей в диапазоне  $[0, 2\pi]$ .

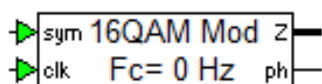


**PM** (Phase Modulation) Modulator – фазовый модулятор, в котором модулирующий сигнал изменяет фазу несущей; PM Modulation Index – индекс фазовой модуляции, рад/В.



**PSK** (Phase Shift Keying) Modulator – набор модуляторов фазоманипулированных (ФМн) сигналов; параметры:

- PSK Type – выбор типа модулятора: BPSK (Binary PSK – бинарный (двухпозиционный) ФМн), QPSK (Quadrature PSK – квадратурный ФМн), 8PSK (8-позиционный ФМн), 16PSK (16-позиционный) или 32PSK (32-позиционный);
- Amplitude, Translation Frequency – амплитуда (В) и частота (Гц) несущей;
- Constellation Rotation – значение циклического сдвига фазы, град.; положительные значения соответствуют вращению против часовой стрелки;
- Gain Imbalance – нестабильность коэффициента передачи по напряжению относительно его номинального значения, дБ;
- Phase Imbalance – нестабильность фазы выходного сигнала, град.;
- Select File, Browse File – кнопки выбора файла с данными PSK-карты и пути к нему;
- Phase Output Mode – режим по выходной фазе: при немодулированной (Unwrapped) или в режиме модулированной (Wrapped) несущей в диапазоне  $[0, 2\pi]$ .

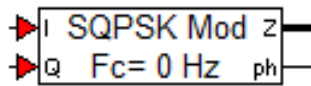


**QAM/PAM** (Quadrature Amplitude Modulation/Pulse Amplitude Modulation) Modulator – семейство амплитудно-манипулированных (АМн) и квадратурных АМн модуляторов; параметры:

- QAM/PAM Type – выбор типа модулятора: 4PAM, 8PAM или 16PAM – 4-, 8- и 16-позиционные АМн; известны также под названием ASK (Amplitude Shift Keying); 16-QAM 32-QAM, 64-QAM, 128-QAM или 256-QAM – квадратурные 16-, 32-, 64-, 128- и 256-позиционные АМн;
- Translation Frequency – несущая частота, Гц;
- Constellation Spacing – циклический сдвиг между каналами (позициями) по уровню модулированного напряжения, В;
- Constellation Rotation – циклический сдвиг между каналами по фазе (в режиме QAM), град.;
- Initial Phase – начальная фаза, град.; положительные значения соответствуют вращению вектора против часовой стрелки; для QAM шаг

размещения вектора на плоскости (I, Q) кратен ( $I/2n$ ,  $Q/2n$ ), для PAM – ( $I/2n$ ,  $Q = 0$ ), где  $n$  – число позиций (каналов), I, Q – вещественная и мнимая оси комплексной плоскости;

- Gain Imbalance – нестабильность коэффициента передачи по напряжению относительно его номинального значения, дБ;
- Phase Imbalance – нестабильность фазы выходного сигнала, град.;
- Select File, Browse File – кнопки выбора файла с данными QAM-карты и пути к нему;
- Phase Output Mode – режим по выходной фазе: при немодулированной (Unwrapped) или в режиме модулированной (Wrapped) несущей в диапазоне  $[0, 2\pi]$ .



**SQPSK** (Staggered Quadrature Phase Shift Keying)

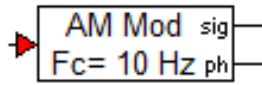
Modulator – модулятор с относительной (смещенной) квадратурной фазовой манипуляцией (ОФМн), известен также под названием OQPSK (Offset QPSK); в таких модуляторах фаза несущей каждого из четырех каналов смещена на  $90^\circ$  при постоянном значении амплитуды; входы: I, Q – цифровые данные (0, 1); z, ph – выходы модулированного сигнала (в виде вектора) и фаза немодулированной несущей, рад; параметры:

- Translation Frequency, Amplitude – частота (Гц) и амплитуда (В) несущей;
- Constellation Rotation – циклический сдвиг между каналами по фазе, град.; положительные значения соответствуют вращению против часовой стрелки;
- Data Rate – скорость поступления данных, симв/с;
- Phase Output Mode – режим по выходной фазе: при немодулированной (Unwrapped) или в режиме модулированной (Wrapped) несущей в диапазоне  $[0, 2\pi]$ .

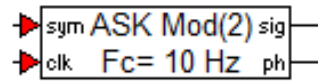


### П1.15. Модуляторы типа Real (Comm/Modulator-Real)

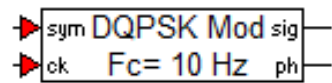
В этом разделе приводятся краткие сведения по моделям модуляторов с входным и выходным сигналами в скалярной форме.



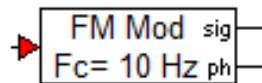
**AM (Re)** аналогичен модулятору AM (Amplitude Modulation).



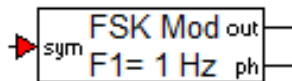
**ASK (Re)** аналогичен модулятору ASK (Amplitude Shift Keying).



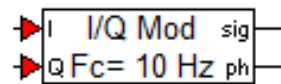
**Differential QPSK (Re)** аналогичен модулятору Differential QPSK (Differential Quadrature Phase Shift Keying).



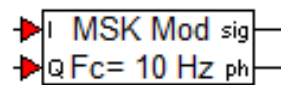
**FM (Re)** – аналогичен модулятору FM (Frequency Modulation).



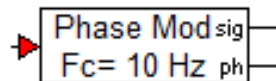
**FSK (Re)** – аналогичен модулятору FSK (Frequency Shift Keying).



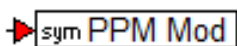
**I/Q (Re)** – аналогичен модулятору I/Q Modulator.



**MSK (Re)** – аналогичен модулятору MSK (Minimum Shift Keying).



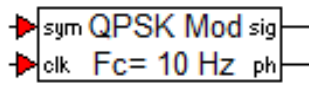
**PM (Re)** – аналогичен модулятору PM (Phase Modulation).



**PPM (Pulse Position Modulation)** – широтно-импульсный модулятор; параметры:

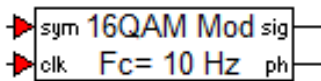
– Number of Levels – количество возможных уровней (каналов) входных сигналов;

- Pulse Width – длительность прямоугольного импульса несущей, с;
- Symbol Rate – скорость входного потока данных, Гц;
- Pulse Amplitude – амплитуда импульса, В;
- Data Frame Start – время поступления первой пачки данных, с.



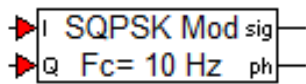
**PSK (Re)** – аналогичен модулятору PSK (Phase Shift

Keying).



**QAM/PAM (Re)** – аналогичен модулятору

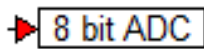
QAM/PAM (Quadrature Amplitude Modulation/Pulse Amplitude Modulation).



**SQPSK (Re)** – аналогичен модулятору SQPSK

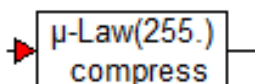
(Staggered Quadrature Phase Shift Keying).

#### П1.3.14. Преобразователи сигналов (Comm/Operatops)



**A/D Converter** (Analog to Digital Converter) – аналого-цифровой преобразователь (АЦП) с параметрами:

- Number of Bits – разрядность АЦП,  $n$ ;
- Output Mode – режим выхода: беззнаковый (Unsigned) целочисленный  $[0, 2^n - 1]$  или с индикацией знака (Signed) данных  $[-2^{n-1} - 1, 2^{n-1} - 1]$ ;
- Zero Level – тип нулевого уровня: средний (Midlevel), равный  $0 \pm MP$  в режиме Signed ( $MP$  – цена младшего разряда) и  $n/2$  в режиме Unsigned, или смещенный (Offset) между  $0 - 1$  в режиме Signed и между  $(n/2 - 1) - (n/2)$  в режиме Unsigned;
- Max Amplitude – максимальная амплитуда входного сигнала, В.



**Compander** – компандер/экспандер для сжатия/

растяжения сигналов по динамическому диапазону (амплитуде); параметры:

- Compander Mode – режим работы: сжатие (Compress) или растяжение (Expand) сигнала по амплитуде;

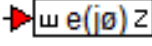
– **Compander Type** – тип сжатия/растяжения:  $\mu$ -Law ( $\mu$  Value = 255 по умолчанию) или A-Law (A Value = 87,6 по умолчанию);

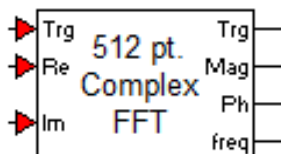
– **Max Input Level** – максимальный уровень входного сигнала; нормализуется к значениям  $[-1, 1]$ ; в случае режима Expand параметр используется для масштабирования по выходу.

Выходной сигнал блока в режиме  $\mu$ -Law:  $y = \text{sgn}(x) \times \ln(1 + \mu \times |x|) / \ln(1 + \mu)$  при  $|x| \leq 1$ , где  $|x|$  – нормированное (относительно  $x_{\max}$ ) значение входного сигнала.

Выходной сигнал блока в режиме A-Law:  $y = \text{sgn}(x) \times \ln(1 + A \times |x|) / \ln(1 + A)$  при  $(1/A) < |x| \leq 1$  и  $y = \text{sgn}(x) \times (A \times |x|) / (1 + \ln A)$  при  $0 \leq |x| \leq (1/A)$ .

Рассматриваемый блок используется при так называемом нелинейном кодировании. Наибольшее распространение оно нашло в цифровых системах телефонной связи с импульсно-кодовой модуляцией (ИКМ), а также в устройствах, которые используют такие каналы, например, в модемах. Сущность такого кодирования заключается в сжатии (компрессии) аналогового (речевого) сигнала на передающей стороне перед его преобразованием в цифровую форму (так называемой оцифровкой) и растяжении (декомпрессии) после декодирования на приемной. Такое нелинейное преобразование обычно осуществляется по логарифмическому закону, что соответствует чувствительности человеческого уха и позволяет при относительно небольшом количестве уровней квантования закодировать звук человеческого голоса с обеспечением его разборчивости на приемлемом уровне.

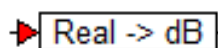
 **Complex Exponential** – блок, реализующий функцию  $y = e^{jx(t)}$  (без параметров).



**Complex FFT/IFFT** – блок прямого FFT (Forward Fast Fourier Transform) и обратного IFFT (Inverse Forward Fast Fourier Transform) преобразования Фурье; входы: Trg – триггерный (уровень сигнала  $> 0,5$ );

Re, Im – действительная и мнимая составляющие комплексного входного сигнала; выходы: Trg – триггерный (0 или 1); Mag, Ph – амплитуда и фаза (выходного сигнала); freq – частота (Hz) в режиме FFT или время (с) в режиме FFT; параметры:

- FFT Mode – выбор режима работы блока: Forward FFT или Inverse FFT;
- FFT Window Type – опциональный выбор вида оконного режима работы (только для FFT);
- FFT Representation – представление FFT данных в виде амплитуда/фаза (Mag/Phase) или в комплексном виде (Real/Imaginary);
- FFT Size – количество отображаемых точек  $N$  для FFT или IFFT (8 – 1048576);
- Unwrap Phase – диапазон изменения аргумента фазовой характеристики (по умолчанию от  $-\pi$  до  $+\pi$ );
- Remove Linear Phase – опция удаления вводимой задержки при расчетах фазовой характеристики;
- Delay – установка временной задержки в случае активизации опции Remove Linear Phase;
- Number of FFT Averages – количество усредняемых точек при расчете FFT спектра;
- Output Freq. Units – выбор единиц измерения частоты (только для FFT);
- Output Mode – режим вывода (только для FFT): стандартный (Standard), в виде энергетического спектра (Power Spectrum) в дБм на нагрузке 50 Ом или спектральной плотности мощности (Spectral Density) в дБм/Гц на нагрузке 50 Ом.



**Conversions** – блок для преобразования (по выбору) комплекса в его модуль и аргумент (Real/Imag to Mag/Phase) и обратно (Mag/Phase to Real/Imag), в децибелы (Real to Decibels) и обратно (Decibels to Real), мощности в дБ мощности (Power to Decibels) и обратно (Decibels to

Power), герц в рад/с (Hertz to Rad/sec) и обратно (Rad/sec to Hertz), градусов в радианы (Degrees to Radians) и обратно (Radians to Degrees).

➡ **8 bit DAC** — **D/A Converter** (Digital to Analog Converter) – цифро-аналоговый преобразователь (ЦАП) с параметрами: количество битов ( $N$ ) квантования входного сигнала  $x$  и минимальный  $A_{\min}$  и максимальный  $A_{\max}$  уровень амплитуды выходного сигнала  $y$ :

$$y = A_{\min} + x \times \Delta q \text{ (беззнаковый режим выхода, unsigned mode),}$$

$$y = A_{\min} + \left( x + \frac{k}{2} \right) \times \Delta q \text{ (режим с индикацией знака, signed mode),}$$

где  $\Delta q = \frac{A_{\max} - A_{\min}}{k - 1}$ .

➡ **z Delay (1) z** — **Delay (Complex)** – блок задержки комплексного сигнала: параметры:

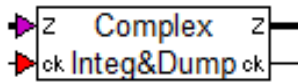
- Delay – значение задержки в секундах или в единицах модельного времени;
- Initial Condition (Real), Initial Condition (Imaginary) – начальные значения действительной и мнимой составляющих на этапе задержки;
- Delay Mode – режим задержки в секундах (Seconds) или в единицах модельного времени (Sim Steps).

➡ **Delay (1)** — **Delay (Real)** – блок задержки сигнала (для действительной составляющей); набор параметров аналогичен **Delay (Complex)**.

➡ **+3. dB** — **Gain (dB)** – усилитель с коэффициентом усиления Gain (ДБ).

➡ **sym IQ Map(16)** — **IQ Mapper** – блок воспроизведения СКК (специально построенных сигналов), представленного в файле; выходы: sym – вход; I, Q – выходы мнимой и вещественной составляющих комплексного сигнала; параметры:

- Constellation Size – количество точек сигнального созвездия;
- Select File, Browse File, File Path – выбор файла с СКК.



**Integrate & Dump (Complex)** – интегратор с

периодическим сбросом; используется при детектировании цифровых сигналов;

выводы:  $z$  – вход и выход комплексного сигнала;  $ck$  – вход и выход сигнала синхронизации; параметры:

- Reset Value – сигнал сброса и ввод начальных условий интегратора перед моделированием; в рабочем режиме сброс блока осуществляется тактовыми сигналами (вход  $ck$ ), сопровождающими каждый дамп (пачку, символ, кадр и т. п.) информационного сигнала;

- Scale Factor – параметр масштабирования выходного сигнала; определяется скоростью вывода информации в приемном устройстве;

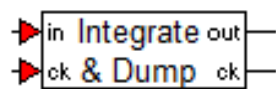
- Dump Timing – управление интегратором и начальной задержкой: внутреннее (Internal) или внешнее (External) (вход  $ck$ );

- Dump Rate – частота следования сигналов сброса (Гц) (в режиме Internal Dump Timing mode);

- Output Mode – режим вывода данных: постоянный уровень между тактовыми импульсами сброса (Held) или выходной сигнал (Continuous);

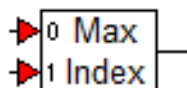
- Integration Method – метод интегрирования: Эйлера (Euler), метод трапеций (Trapezoidal) или обратных разностей (Backward Difference);

- Initial Delay – задержка сброса, с (режим Internal Dump Timing mode).



**Integrate & Dump (Real)** – интегратор скалярных

сигналов с периодическим сбросом; параметры аналогичны Integrate & Dump (Complex).



**Max Index** – блок определения максимального по амплитуде

сигнала из возможных 16; параметры:

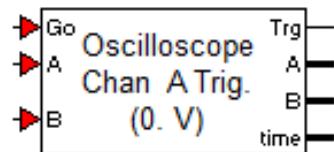
- Number of Inputs – число входов (2 – 16);

- Index Mode – режим индексирования: с момента моделирования (Start at Zero) или начиная с первого сигнала после старта (Start at One).

→ **Real Mod(2.)** — **Modulo** — блок деления по заданному модулю;

параметры:

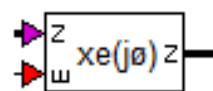
- Modulo — значение модуля;
- Modulo Mode — результат операции: целое (Integer) или вещественное (Real) число;
- Tolerance — погрешность определения результата типа Integer.



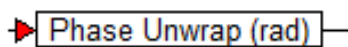
**Oscilloscope (Core)** — блок эмуляции входных цепей

двухканального осциллографа; обозначения выводов: Go — вход запуска (уровень сигнала  $> 0,5$ ); A, B — входы и векторные выходы каналов A и B; Trg — выход (0 или 1) для запуска блока plot; time — сигнал развертки (ось X) блока plot; параметры:

- Trigger Channel — запуск: сигналом канала A (A Channel) или B (B Channel); Triggering Mode — ждущий режим: запуск по нарастанию (Rising Edge) или спаду (Falling Edge) сигнала заданного уровня;
- Time Span — длительность развертки, с;
- Threshold — уровень сигнала запуска, В.



**Phase Rotate** — фазовращатель (без параметров).



**Phase Unwrap** — фазовращатель с запоминанием

предыдущих средневзвешенных значений фазы; удовлетворительная работа блока обеспечивается при разности значений фаз в  $45^\circ$ ; параметр: Units — выбор единицы измерения фазы: в градусах (Degrees) или радианах (Radians).

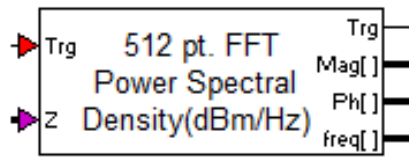


**Polynomial** — блок, реализующий полиномиальную функцию

$y(x) = ax^5 + bx^4 + cx^3 + dx^2 + ex + f$ ; параметры:

- $x^5$  Coefficient ...  $x^0$  Coefficient — значения коэффициентов  $a \dots f$ ;
- Show Formula — опция для отображения формулы внутри значка блока;

– Decimal Digits – число десятичных знаков, используемых для отображения значений коэффициентов полинома (при опции Show Formula).



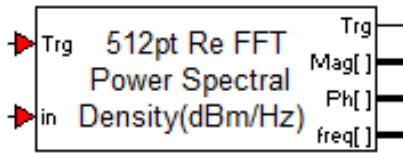
**Spectrum (Complex)**

– измеритель

комплексного энергетического спектра входного сигнала (вход z) в непрерывном режиме или на выбранных участках путем подачи в заданное время сигнала 1 на вход Trg; выходы синхронизации Trg, значения модуля Mag[ ], фазы Ph[ ] и частоты freq[ ] используются для вывода данных на блок plot; параметры:

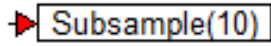
- Trigger Mode – режим запуска: непрерывный (Continuous) или ждущий (Triggered);
- FFT Window Type – опциональный выбор оконного режима работы;
- FFT Size – количество отображаемых точек  $N$  для FFT или IFFT (8 – 1048576);
- Spectral Output – режим выхода: амплитуда/фаза (Mag/Phase) или в комплексном виде (Real/Imaginary);
- Unwrap Phase – диапазон изменения аргумента фазовой характеристики ( $-\pi \dots +\pi$  по умолчанию);
- Remove Linear Phase – опция удаления вводимой задержки при расчетах фазовой характеристики;
- Delay – установка временной задержки в случае активизации опции Remove Linear Phase;
- Number of FFT Averages – количество усредняемых точек при выводе осциллограммы FFT спектра;
- Output Freq. Units – выбор единиц измерения частоты (для FFT);
- Power Spectrum Units – единицы измерения мощности и ее спектральной плотности: Вт (Watts), дБм (dBm), дБм/Гц (DBm/Hz);
- Load – нагрузка: 1 Ом (1 Ohm) или 50 Ом (Ohms).





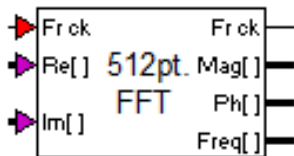
**Spectrum (Real)** – блок аналогичный Spectrum

(Complex), но для действительных сигналов (для положительных частот).



**Subsample** – блок масштабирования; параметры:

- Decimate by – целочисленный параметр масштабирования; например, при значении 5 будет фиксироваться каждый пятый шаг моделирования;
- Offset – смещение от первого шага моделирования для уменьшения размера изображения; при нулевом значении параметра такое уменьшение начинается с первого шага моделирования;
- Output Mode – режим вывода: выходной сигнал с нулевым значением между расчетными точками (Pulsed Output) или постоянным уровнем (Held Output).

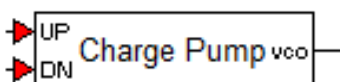


**Vector FFT** – блок аналогичен блоку Complex

FFT/IFFT, для сигналов в виде вектора размера N (Re[ ], Im[ ] на входе и Mag[ ], Ph[ ] на выходе), синхронизируемых тактовой частотой Frck на входе и Freq[ ] на выходе; набор параметров аналогичен блоку Complex FFT/IFFT за исключением:

- Use Simulation Time Step – установка шага модельного времени;
- Sampling Frequency – частота посылок (выборок).

### П1.16. Элементы систем фазовой автоподстройки частоты (Comm/PLL)



**Charge Pump** – активный фильтр первого порядка

для цифровых PLL (Phase Locked Loops – систем фазовой автоподстройки частоты (ФАПЧ)), являющийся одновременно сравнивающим устройством для сигналов фаза/частота на входах UP и DN; выходной сигнал блока (вывод vco)

используется для управления источником VCO (Voltage Controlled Oscillator – генератор, управляемый напряжением); параметры:

- Natural Frequency – характеристическая частота, рад/с; определяет время выхода на режим ФАПЧ;
- Loop Bandwidth – полоса пропускания цепи обратной связи ФАПЧ (обычно на уровне – 3 дБ), Гц; является определяющим для параметра Natural Frequency;
- Damping Factor – уровень определения параметра Loop Bandwidth (по умолчанию – 0,7071068, т. е. – 3 дБ);
- VCO Gain – коэффициент преобразования ( $K_o$ ) VCO, Гц/В;
- Detector Gain – коэффициент передачи фазового детектора (для типа 3 определяется как  $K_d = 1/\pi$  и  $K_d = 1/2\pi$  для типа 4), В/рад; например, при  $K_d = 1$  ошибка в 0,1 В по напряжению соответствует ошибке 0,1 рад по фазе;
- Specification Method – выбор постоянной времени подстройки по полосе частот (Loop Bandwidth) или характеристической частоте (Natural Frequency);
- Update – кнопка модификации полосы пропускания контура автоподстройки (по значению параметров Loop Bandwidth или Natural Frequency) в соответствии со значениями постоянных времени  $Tau1$  и  $Tau2$ , указанных в диалоговом окне блока и временно вступающих в силу после нажатия кнопки ОК.

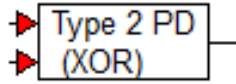


**Loop Filter (2nd Order PLL)** – фильтр первого порядка для ФАПЧ второго порядка; набор параметров аналогичен блоку Charge Pump за исключением параметра Loop Type – тип фильтра: активный (Active) с коэффициентом передачи  $K(p) = (1 + pT_2)/pT_1$  или пассивный (Passive) с  $K(p) = (1 + pT_2)/(1 + pT_1)$ .

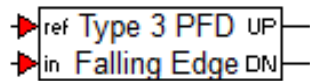


**Loop Filter (3rd Order PLL)** – оптимальный фильтр Винера второго порядка для ФАПЧ третьего порядка; набор параметров

аналогичен блоку Charge Pump; выход рассчитан на подключение VCO блока, а вход – на подключение к фазовому детектору. Для устойчивой работы модели рекомендуется выбирать шаг моделирования намного меньше периода собственной частоты ФАПЧ.

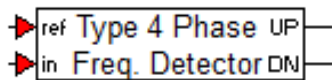


**Type 2 Phase Detector** – цифровой фазовый детектор с функциями логического элемента ИСКЛЮЧАЮЩЕЕ ИЛИ (XOR); параметр: Threshold – чувствительность по входу.



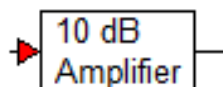
**Type 3 Phase Detector** – цифровой фазовый детектор с внутренним JK-триггером; выводы: ref – вход синхронизации; in – сигнал с VCO; UP, DN – выходы сигналов ошибки (рассогласования) [0, 1] и [0, 1]; параметры:

- Initial State – исходное состояние внутреннего JK-триггера;
- High Threshold, Low Threshold – уровни входного напряжения, принимаемые в качестве сигналов высокого и низкого уровня (например, 1 и 0);
- Edge Mode – режим запуска: по нарастающей (Rising Edge) или спадающей (Falling Edge) части запускающего сигнала.



**Type 4 Phase Detector** – цифровой фазовый детектор типа 4 повышенной чувствительности; назначение выводов и набор параметров аналогичны Type 3 Phase Detector.

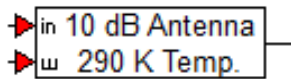
### П1.17. Высокочастотные элементы (Comm/RF)



**Amplifier** – нелинейный усилитель с амплитудной характеристикой, аппроксимируемой полиномом Тейлора:  $y = ax + bx^2 + cx^3 + dx^4 + ex^5$ ; параметры:

- Gain – коэффициент усиления, дБ;
- 1 dB Compression Point – уровень сжатия выходной мощности, дБм;

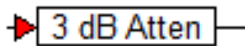
- IP2 – уровень выходной мощности (в дБм) при аппроксимации второго порядка (20 – 25 дБ по напряжению);
- Output Bias – смещение выходного постоянного уровня в положительную (Positive) или отрицательную (Negative) сторону в точке IP2;
- IP3 – уровень выходной мощности (в дБм) при аппроксимации третьего порядка (10 – 15 дБ по напряжению);
- IP4 – уровень выходной мощности (в дБм) при аппроксимации четвертого порядка;
- Noise Figure – эквивалентный коэффициент шума усилителя в децибелах для случая, если выбрана опция Add Noise (добавление белого шума).



**Antenna** – блок моделирует высокочастотную антенну с параметрами шумовой температуры (*noise*) и усиления (*gain*); шум прибавляется к выходному сигналу  $y$  на основе указанной шумовой температуры и не связан с параметром усиления антенны; блок может также быть смоделирован как бесшумный; импеданс по умолчанию 50 Ом; усиление антенны может быть определено или как фиксированное (fixed mode), или как функция от угла падения (pattern mode), который задается сигналом на втором входе  $x_2$ ; уровень амплитуды выходного сигнала  $y$ :

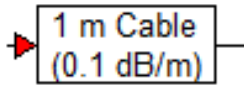
$$y = x \times 10^{(gain/20)} + noise \text{ (режим fixed mode),}$$

$$y = x \times 10^{(gain(x_2)/20)} + noise \text{ (режим pattern mode).}$$



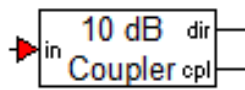
**Attenuator** – пассивный аттенюатор (делитель напряжения); параметры:

- Loss – коэффициент ослабления, дБ;
- Physical Temperature – установка температуры в градусах Кельвина для случая выбора опции Add Noise (добавление белого шума).



**Cable** – блок моделирует пассивный высокочастотный

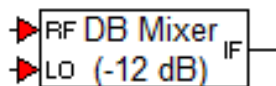
кабель с указанным коэффициентом ослабления на единичное расстояние ( $loss/meter$ ); шум прибавляется к выходному сигналу  $y$  на основе указанной длины кабеля ( $cable\ length$ ) и его физической температуры (Phys. Temperature); блок может также быть смоделирован как бесшумный; импеданс по умолчанию 50 Ом; уровень амплитуды выходного сигнала  $y = x \times 10^{(-loss/20)} + noise$ , где  $loss = cable\ length \times loss/meter$ .



**Coupler** – высокочастотный соединитель с 50-омным

импедансом и дистанционным переключением направления передачи сигнала (вывод  $dir$ ); параметры:

- Output Coupling – выходные параметры соединения;
- Direct Path Loss – потери в прямом направлении, дБ;
- Coupled Loss – общие потери соединения, дБ;
- Coupling Mode – конфигурация использования блока: в качестве входного (Input Coupling) или выходного (Output Coupling) соединителя;
- Noise Figure – уровень шумов (при включенной опции Add Noise), дБ.

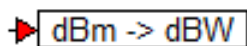


**Double Balanced Mixer** – двойной нелинейный

балансный смеситель; параметры:

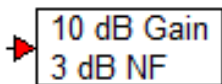
- Conversion Loss – потери преобразования, дБ; (отношение сигнала промежуточной частоты IF к сигналу радиочастоты RF);
- 1 dB Compression Point – уровень мощности входного сигнала (дБм), при котором потери преобразования увеличиваются на 1 дБ;
- IP3 – минимальный (теоретический) уровень мощности входного сигнала (дБм), при котором уровень интермодуляционных искажений не превышает 10 – 15 дБ;

- LO Power – уровень сигнала на входе LO (дБм); используется для масштабирования IF сигнала;
- RF Power – уровень сигнала на входе RF (дБм); используется для масштабирования IF сигнала;
- LO 3rd Harmonic, LO 5th Harmonic – уровни испытательных сигналов на входе LO для третьей и пятой гармоник;
- DC Offset – постоянная составляющая на выходе смесителя; определяется в случае, когда смеситель используется в качестве фазового детектора;
- RF Isolation – ослабление сигнала, вызванное сопротивлением утечки между выводами RF и IF, дБ;
- LO Isolation – ослабление сигнала, вызванное сопротивлением утечки между выводами LO и IF, дБ;
- Noise Figure – уровень шумов (опция Add Noise), дБ;
- Add Noise – опция добавки аддитивного белого шума.



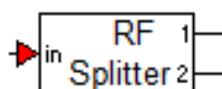
#### **RF Conversions**

– блок реализует множество преобразований, относящихся к схемам RF; требуемое преобразование выбирается в диалоговом окне установки блока.



#### **RF Gain**

– блок моделирует идеальный элемент усиления RF; шум прибавляется к выходному сигналу у на основе указанного коэффициента шума или шумовой температуры устройства и усиления по току; блок может также быть смоделирован как бесшумный; импеданс по умолчанию 50 Ом.



#### **Splitter/Combiner**

– высокочастотный разветвитель с 50-омным импедансом; параметры:

- Splitter Mode – режим работы: без поворота фазы (Split 0), с поворотом фазы на 180° (Split 180) или комбинированным (Combiner) разветвлением;

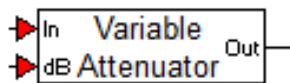
- Number of Connections – количество выходов (в режиме Splitter Mode) или входов (в режиме Combiner Mode) (2 – 16 и 2 – 180 для Split Mode);
- Additional Loss – дополнительные потери (дБ), связанные с числом входов/выходов;
- Noise Figure – эквивалентный коэффициент шума в децибелах для случая, если выбрана опция Add Noise (добавление белого шума).



**Switch** – высокочастотный ключ с 50-омным

импедансом; параметры:

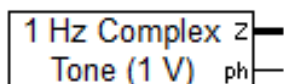
- Switch Mode – режим работы:  $N$  входов и один выход (Input Mode) или  $N$  выходов и один вход (Output Mode);
- Number of Connections – количество соединений (2 – 8);
- Switch Loss – ослабление сигнала в ключе, дБ;
- Isolation – ослабление сигнала, вызванное сопротивлением утечки между входом и выходом ключа, дБ; параметр не используется, если установлена опция Perfect Isolation – сопротивление утечки пренебрежимо мало;
- Noise Figure – эквивалентный коэффициент шума в децибелах для случая, если выбрана опция Add Noise (добавление белого шума).



**Variable Attenuator** – переменный пассивный

аттенюатор с управлением по входу dB; параметр Phys. Temperature – температура аттенюатора (по умолчанию – 290 K); используется при включенной опции Add Noise (добавка белого (теплового) шума).

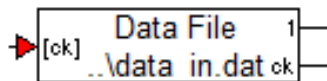
## П1.18. Источники сигналов (Comm/Signal Sources)



**Complex Tone** – источник комплексного сигнала;

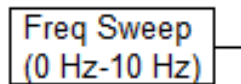
параметры:

- Frequency – частота, Гц;
- Amplitude/Power – установка режима измерения выходных параметров: по напряжению (V) или комплексной мощности на нагрузке 50 Ом (дБм);
- Initial Phase – начальная фаза, град.;
- Units – единицы измерения: напряжения (Volts) или мощности (dBm);
- Phase Output Mode – режим по выходной фазе: при немодулированной (Unwrapped) или модулированной (Wrapped) фазе в диапазоне  $[0, 2\pi]$ .



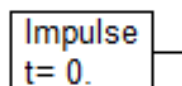
**File Data** – источник с данными в загружаемом файле; выходы: [ck], ck – входы и выходы синхронизации: 1 – выход сигнала; параметры:

- Timing – тип синхронизации: внутренняя (Internal) или внешняя (External); Repeat at EOF – опция повторения выходной последовательности сигналов после использования данных, представленных в файле;
- Number of Outputs (n) – число выходов (1 – 5);
- Data Rate – скорость следования выходных сигналов (Гц) в режиме Internal Timing mode;
- Start Time – время старта (с) в режиме Internal Timing mode;
- Empty Value – значение выходного сигнала после достижения отметки EOF (конец файла);
- Select File, Browse File – кнопки выбора файла данных и пути к нему.



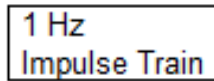
**Frequency Sweep** – источник качающейся частоты (сви́п-генератор); параметры:

- Start Frequency, Stop Frequency – начальная и конечная частота, Гц;
- Sweep Duration – продолжительность повторяющегося сигнала, с;
- Amplitude, Initial Phase – амплитуда (В) и начальная фаза (град.).



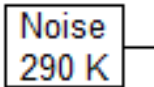
**Impulse** – источник одиночного импульса задаваемой длительности (параметр Impulse Time) амплитуды (Amplitude).





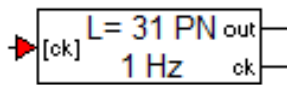
**Impulse Train** – источник импульсной последователь-

ности с задаваемыми частотой (Pulse Frequency), амплитудой (Amplitude) и временем старта (Start Time); блок реализует формулу:  $y = d_0[nT - t]$ , где  $d_0[nT]$  – единичная импульсная решетчатая функция;  $n$  – номер дискретного момента времени;  $T$  – период следования импульсов;  $t$  – время запуска генератора (чистое запаздывание);  $y = 1$ , если время моделирования совпадает с моментами времени  $nT$ , в противном случае  $y = 0$ .



**Noise** – источник гауссова шума; параметры:

- Load – нагрузка: 1 Ом (1 Ohm) или 50 Ом (50 Ohm);
- Noise Units – единицы измерения плотности шума (Noise Density или Noise Temperature – шумовая температура): дБм/Гц (dBm/Hz), Вт/Гц (Watts/Hz) или в градусах Кельвина (Degrees Kelvin).

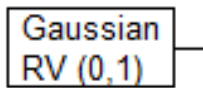


**PN (Pseudo Noise) Sequence** – генератор псевдо-

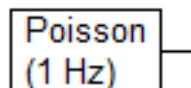
случайной последовательности, известной также как PRBS (Pseudo Random Binary Sequence – псевдослучайная бинарная последовательность); выходы: [ck], ck – входы и выходы синхронизации; out – выход сигнала; параметры:

- Shift Register Size – разрядность  $n$  сдвигового регистра (2 – 28); определяет длину последовательности  $2^n - 1$  ( $2^n$  – для увеличенных последовательностей);
- Sequence Offset – исходное состояние последовательности (от 0 до  $2n - 1$ );
- Initial State – исходное состояние внутреннего сдвигового регистра в восьмеричном формате;
- Generator Coefficient – параметр определяет образующий код для PN последовательности в восьмеричном формате; значение по умолчанию определяется при выборе опции Default Generator Coefficient;

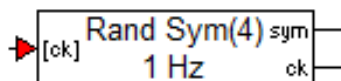
- Zero Augmented Sequence – последовательность с дополнительным нулем;
- Output Mode – режим выхода: биполярный сигнал (Bilevel)  $(-1, 1)$  или бинарный (Binary)  $(0, 1)$ ;
- Timing – вид синхронизации: внутренняя (Internal) или внешняя (External);
- Bit Rate – скорость следования сигналов (бит/с) для Internal Timing mode;
- Start Time – время старта, с.



**Random Distribution** – блок генерирует значения случайной переменной согласно выбранного распределения функции, включая равномерное, нормальное, экспоненциальное и распределение Рэлея.

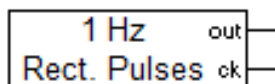


**Poisson Arrivals** – блок генерирует импульсную последовательность с интервалами между импульсами, распределенными по экспоненте, согласно указанной средней частоте поступления.



**Random Symbols** – источник равномерно распределенных случайных символов (на выходе sym); набор параметров:

- Number of Symbols – число битов (уровней) в символе;
- Symbol Rate – скорость следования символов, симв/с.



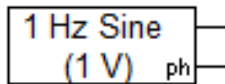
**Rectangular Pulses** – источник двухполярных или однополярных прямоугольных импульсов (out, ck – выходы сигналов генератора и синхронизации); параметры:

- Pulse Frequency – частота следования импульсов, Гц; величина, обратная периоду  $T$ , с;
- Pulse Mode – способ задания параметров импульса: Duration – по длительности  $\tau$  высокого уровня  $(0 < \tau < T)$  или Duty Cycle – в виде

относительных единиц – коэффициента заполнения  $100 \times (\tau/T)$  ( $0 < 100(\tau/T) < 100$ ), %;

– Low Level, High Level – нижнее (– 1) и верхнее (1) значения амплитуды сигнала, В; нулевое значение одного из этих параметров соответствует однополярному режиму источника;

– Start Time – время задержки (старта), с.



**Sinusoid** – источник гармонического сигнала; на выходе ph

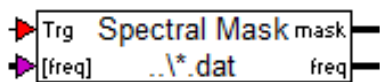
формируется сигнал, равный (в радианах) произведению угловой частоты  $\omega$  и времени моделирования  $t$  (по умолчанию 10 с); параметры:

– Frequency, Amplitude (Power), Initial Phase – частота (Гц), амплитуда (В) или мощность (дБм), начальная фаза (град.) сигналов;

– Amplitude/Power – работа в режиме источника напряжения или мощности;

– Output Mode – форма выходного сигнала: синусоидальная (Sine) или косинусоидальная (Cosine);

– Units – единицы измерения: амплитуды (Volts), мощности (dBm).



**Spectral Mask** – блок генерации сигналов с

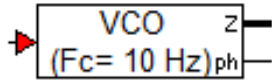
заданным в файле спектром; для контроля спектра рекомендуется применять блок Spectrum Analyzer; входы: Trg – вход запуска ( $> 0,5$ ); [freq] – вектор управления параметрами настройки (используется опционально, когда есть необходимость оперативно изменять параметры генератора); mask – выход; freq – дополнительный выход для плоттера; параметры:

– Mask Data Range – форматы задания параметров маски: в скалярном виде  $[0, fs/2]$  или комплексном  $[- fs/2, + fs/2]$ ;

– Frequency Source – управление параметрами генератора: External – внешнее (по входу [freq]) или Internal – внутреннее (с использованием значений шага моделирования и заданного количества  $N$  выборок БПФ);

– Spectrum Size – количество  $N$  выборок БПФ (8 – 16384);

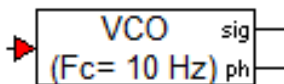
- Select File, Browse File, Mask File Path – выбор файла настроек.



**VCO (Voltage Controlled Oscillator) (Complex)** –

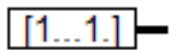
источник комплексного сигнала (на выходе  $z$ ) с управляемой частотой; параметры:

- Center Frequency – центральная (начальная) частота, Гц;
- Amplitude, Initial Phase – амплитуда (В) и начальная фаза (град., выход  $ph$ );
- VCO Gain – коэффициент передачи VCO, Гц/В;
- Integration Method – метод интегрирования: Эйлера (Euler), метод трапеций (Trapezoidal) или обратных разностей (Backward Difference);
- Phase Output Mode – режим по выходной фазе: при немодулированном (Unwrapped) или модулированном (Wrapped) сигнале в диапазоне  $[0, 2\pi]$ .



**VCO (Real)** – блок аналогичен блоку VCO (Complex),

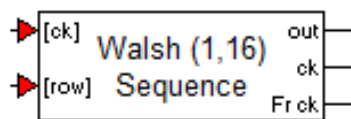
но для скалярных сигналов.



**Vector Constant** – источник сигнала в векторной форме;

параметры:

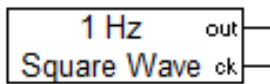
- Vector Size – размер вектора (1 – 1048576);
- Constant Value – спецификация для всех векторных элементов.



**Walsh Sequence** – источник повторяющихся

ортогональных последовательностей Уолша, используемых в системах множественного доступа с кодовым разделением каналов (CDMA). Выводы:  $[ck]$ ,  $ck$  – вход и выход тактовых импульсов;  $Fr\ ck$  – выход тактовых импульсов фреймов;  $[row]$  – вход сигналов выбора строки;  $out$  – выход последовательностей Уолша; параметры:

- Row Selection – способ выбора строки  $K$  матрицы размера  $(N \times N)$ , где  $N$  – длина последовательности: внутренний (Fixed (Internal)) или внешний (External Input) с использованием входа [row];  $K = 0 - 2^{N-1}$ ;
- Matrix Output Row – выбор строки  $K$  в режиме Fixed (Internal);
- Sequence Offset – смещение старта последовательности  $(0 - (N - 1))$ ;
- Sequence Length – длина  $N$  последовательности  $(2 - 256)$ ;
- Output Mode – режим выхода: биполярный (Bilevel)  $(-1, 1)$  или однополярный (Binary)  $(0, 1)$ ;
- Timing – синхронизация: внутренняя (Internal) или внешняя (External);
- Bit Rate – средняя скорость последовательности, бит/с; задается только в режиме Internal Timing mode;
- Start Time – время старта последовательности.



**Waveform Generator** – универсальный функциональный

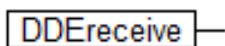
генератор; параметры:

- Waveform Type – форма выходных сигналов: прямоугольная (Square Wave), треугольная (Triangle Wave) или зубчатая (Sawtooth Wave);
- Waveform Frequency – частота следования сигналов, Гц;
- Peak-to-Peak Amplitude – размах выходных сигналов, В;
- Offset – постоянная составляющая, В;
- Start Time – стартовое время, с.

Далее рассматриваются библиотеки компонентов из меню Blocks.

### П1.19. Блоки импортирования данных (Blocks/DDE)

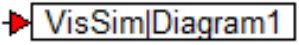
Блоки DDE (Dynamic Data Exchange) позволяют организовать в процессе моделирования двусторонний динамический обмен данными между VisSim и другими приложением Windows.

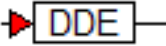


**DDEreceive** – блок приема данных из приложения

Windows или внешнего источника (сервера) в схему модели; параметры:

- ServerTopic – адрес сервера или путь к файлу данных;
- Receive Item – спецификация принимаемых данных;
- Custom Update Interval – опция установки времени, в течение которого могут приниматься данные;
- Synchronous Operation – данные для синхронизации буфера блока;
- Output Dimension – тип данных: скаляр, вектор или матрица;
- Bitmap – выбор для значка блока нового изображения, содержащегося в предварительно подготовленном BMP-файле;
- Paste Link – кнопка установки связи текущей схемы модели с импортируемыми данными.

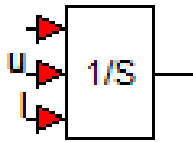
 **DDEsend** – блок передачи данных текущей схемы модели в приложение Windows; отличается параметром Send Item (спецификация передаваемых данных).

 **DDE** – комбинированный блок передачи/приема данных, объединяющий функции двух предыдущих блоков.

## П1.20. Интеграторы (Blocks/Integration)

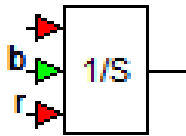
 **Integrator** – блок интегрирования, реализующий один из методов численного интегрирования, установленный в окне команды Simulate/Simulation Properties; параметры:

- Initial Condition – начальное значение выходного сигнала (начальное значение интеграла на первом шаге моделирования); задание начальных условий требуется, например, при моделировании цепей с реактивными элементами (индуктивностями и конденсаторами);
- ID – идентификационный номер интегратора в моделируемой системе (порядок любой системы (блок-схемы) равен числу интеграторов в ней);
- Checkpoint State – фиксированное состояние интегратора на момент приостановки процесса моделирования.



**LimitedIntegrator** – интегратор с насыщением и аналогичным

набором параметров блока Integrator; два дополнительных входа используются для установки верхнего (u) и нижнего (l) уровней ограничения.



**ResetIntegrator** – при  $|b| < 1$  блок выполняет функции

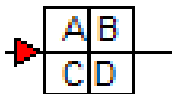
аналогичные простому блоку интегрирования или интегратор со сбросом, при  $|b| > 1$  (сброс до уровня сигнала на входе r).



**Derivative** – цифровой интегратор; параметры аналогичны

блоку Integrator.

### П1.21. Линейные блоки (Blocks/Linear system)



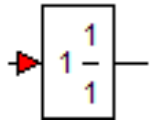
**StateSpace** – блок преобразования коэффициента передачи

объекта в систему дифференциальных уравнений для переменных состояния в виде:  $u' = A \times u + B \times x$ ;  $y = C \times u + D \times x$ , где  $x$  – вектор входных переменных;  $y$  – вектор выходных переменных;  $u$  – вектор переменных состояния (фазовых координат системы);  $A$  – матрица коэффициентов системы;  $B$  – матрица входных коэффициентов (матрица управления);  $C$  – матрица выходных коэффициентов;  $D$  – матрица коэффициентов пропорциональных каналов (матрица компенсации); данные готовятся в виде файла (имя в строке File Name), в формате MATLAB (опция .mat File) или Commsim (.m File); другие параметры:

- $dT$  – шаг дискретизации в случае выбора опции Discrete (замещение в блоке матрицы интеграторов  $[1/S]$  матрицей регистров задержки  $[1/Z]$ ); шаг дискретизации по умолчанию равен текущему шагу моделирования;

- Initial State – начальное состояние;

– Input Count, Output Count, State Count – число входов, выходов и состояний.



**TransferFunction** – блок формирования системы с заданным коэффициентом передачи; параметры:

– Polynomial – опция представления коэффициента передачи в виде отношения полиномов  $F(p)$ , коэффициенты которых (при операторах  $p^k$ ) через пробел в убывающем порядке заносятся в строках Denominator (знаменатель) и Numerator (числитель);

– .mat File, .m File – опция представления коэффициента передачи текстовым файлом в формате MATLAB или VisSim; его имя должно быть занесено в строке File;

– Discrete – представление данных в дискретной форме с временем (шагом) дискретизации  $dT$ ;

– Poles and Zeros – ввод коэффициента передачи в виде разложения полиномов путем задания полюсов (корней знаменателя), нулей (корней числителя) и коэффициента передачи Gain;

– Tapped Delay – задержка при использовании фильтров типа IIR или FIR, диалоговые окна которых вызываются кнопками IIR Filter и FIR Filter;

– Convert  $S \rightarrow Z$ , Convert  $Z \rightarrow S$  – преобразование коэффициента передачи из непрерывной в дискретную форму и наоборот;

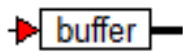
– Display Filter Method – опция использования фильтра при графическом отображении результатов моделирования;

– Use 32-Bit Precision – опция использования 32-битовой обработки вместо 64-битовой по умолчанию;

– Initial Value – начальное состояние блока.

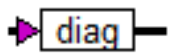


## П1.22. Матричные операции (Blocks/Matrix Operations)

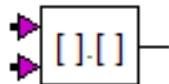


**Buffer** – буфер типа FIFO (первым вошел – первым вышел)

для приема данных в виде скалярной последовательности разрядностью  $\text{Buffer Length}$  в течение заданного промежутка времени  $dT$  (периода стробирования; шаг моделирования может быть в целое число раз меньше или равен ему), а также преобразования ее в вектор (матрицу-строку); основное назначение блока – группировка последовательных данных в параллельные для операций цифровой обработки сигналов.



**Diag** – создание, квадратной матрицы с вектором  $V$  на диагонали и возможностью его смещения (параметр Diagonal Offset).



**Dot Product** – блок перемножения двух матриц с выходом в

виде скаляра; вычисляет сумму всех произведений соответствующих элементов в двух матрицах-столбцах или строках.



**Fft** – блок быстрого преобразования Фурье (БПФ) данных из

временной области в частотную на каждом шаге моделирования. Если на вход блока подается количество выборок, не кратное степеням двойки (8, 16, 32, 64, 128, ...), то блок автоматически дополняет массив выборок нулями. Размерность выходного массива ( $n$ ) совпадает с размерностью входного; возвращаемые данные – пары ортогональных коэффициентов ряда Фурье, пропорциональные вещественным и мнимым составляющим каждой гармоники в количестве  $n/2$ .



**Ifft** – блок обратного быстрого преобразования Фурье (ОБПФ)

данных из частотной во временную область; другие сведения аналогичны блоку `fft`.



**Index** – блок индексации для определения одного из

элементов матрицы, подаваемой на нижний вход, по адресу, определяемому

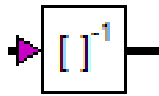
номерами столбца  $c(\text{column})$  и строки  $r(\text{row})$ ; если адресуется элемент вне границ матрицы, то выдается сообщение: `Index n too big` (индекс  $n$  слишком большой) с установкой флага ошибки и закрашиванием блока красным цветом.

При адресации используются правила:

- если индексирующий сигнал имеет тип с плавающей точкой, то округление для индексации выполняется отбрасыванием дробной части, например сигнал 1,999 будет округлен до 1 (будет индексировать первый, а не второй элемент);

- при неподключенной входной матрице при любых значениях индексирующих сигналов блок будет возвращать ноль без предупреждения об ошибке;

- при адресации элемента вне границ входной матрицы и игнорировании предупреждающих сообщений блок будет возвращать число, близкое к нулю, например  $1.06983\text{e}-306$ .



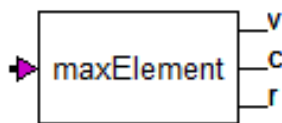
**Invert** – блок инвертирования невырожденной квадратной

матрицы; обратная матрица получается как частное от деления транспонированной матрицы алгебраических дополнений элементов входной матрицы на определитель той же входной матрицы. 1



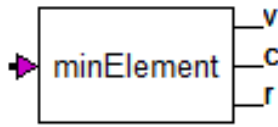
**IndexedAssign** – блок определяет значение элемента

матрицы. Входы  $c$  и  $r$  указывают столбец и строку матрицы на пересечении которых находится определяемый элемент; нижний входной соединитель представляет значение элемента; на оставшийся вход блока подается матрица.



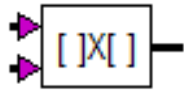
**MaxElement** – блок имеет один вход для ввода

матрицы и три вывода; вывод  $v$  – значение максимального элемента, выходы  $c$  и  $r$  – строка и столбец найденного максимального элемента.



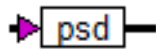
**MinElement** – блок имеет один вход для ввода матрицы

и три вывода; вывод  $v$  – значение минимального элемента, выходы  $c$  и  $r$  – строка и столбец найденного минимального элемента.



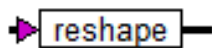
**Multiply** – блок перемножения двух сцепленных матриц

(число столбцов в первой матрице должно быть равно числу строк во второй).



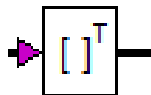
**Psd** – блок для преобразования на каждом шаге моделирования

вектора сигнала в вектор значений его энергетического спектра с использованием БПФ для массива выборок с соответствующим масштабированием. При увеличении амплитуды входного сигнала в 10 раз спектр увеличивается на 40 дБ, поскольку мощность увеличивается пропорционально квадрату сигнала. Если на вход блока подается количество выборок  $n$ , не кратное степеням двойки (8, 16, 32, 64, 128, ...), то блок автоматически дополняет массив выборок нулями. Размерность выходного массива ( $n/2$ ) в два раза меньше размерности входного. Шаг частоты между спектральными полосами (для выходных данных блока) определяется по формуле  $f/n$ , где  $f$  – частота стробирования буфера.



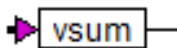
**Reshape** – преобразователь формы матрицы заданной

размерности Dim1 и Dim2.



**Transpose** – транспонирование входной матрицы (зеркальное

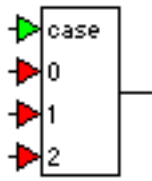
отображение матрицы относительно главной оси с заменой строк на столбцы).



**Vsum** – блок суммирования значения всех элементов

матрицы.

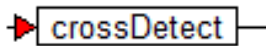
### П1.23. Нелинейные блоки (Blocks/Nonlinear)



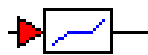
**Case** – блок коммутации входного сигнала case (скаляр, вектор или матрица) по задаваемым  $n = 0, 1, 2, \dots$  адресам (или условиям отбора).

При обработке сигналов адресации действуют правила:

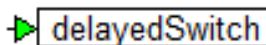
- адреса начинаются с нуля (вход 0);
- если сигнал адреса является числом с плавающей точкой, то выполняется округление отбрасыванием дробной части; например, сигнал 0,999 будет округлен до 0, т. е. будет подключен нулевой, а не первый канал;
- при выходе сигнала за адресный диапазон мультиплексора выдается соответствующее сообщение с установкой флага ошибки и закрашиванием блока красным цветом.



**CrossDetect** – компаратор с задаваемым Cross Point уровнем срабатывания и выходным сигналом + 1 при нарастающем входном, – 1 – при спадающем и 0 – при необнаружении сигнала.



**Deadband** – модель с зоной нечувствительности (параметр Dead Band (DB)), например, модель зазора в механических, системах (в электронных усилителях данный вид нелинейности называется искажением типа «ступенька»). Выходной сигнал  $y = 0$ , если  $|x| \leq DB/2$ , иначе  $y = x - (\text{sign } x) DB/2$ . Если абсолютная величина входного  $x$  сигнала меньше или равна вычитаемой константе, то значение выходного сигнала будет равно нулю.



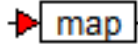
**DelayedSwitch** – блок обеспечивает задержку при переходе с 0 на 1 (on) и с 1 на 0 (off).



**Int** – блок выделения целой части дробного числа.



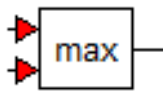
**Limit** – модель звена с заданными верхним (Upper Bound,  $ub$ ) и нижним (Lower Bound,  $lb$ ) уровнями ограничения; работа блока описывается формулами:  $y = x$ , если  $lb \leq x \leq ub$ ;  $y = lb$ , если  $x < lb$ ;  $y = ub$ , если  $x > ub$ .



**Map** – табличное представление нелинейной функции MAP-файлом; параметры:

- Map File Name – имя файла с расширением .MAP; вводится вручную или с помощью кнопок Select File и Browse Data;
- Type – выбор типа данных из предлагаемого списка;
- Map Dimensions – выбор размерности табличных данных: одномерные (1-D) (в MAP-файле записываются, например, в виде  $10 \times 1 [1:100]$  – таблица из 10 элементов, значения которых находятся в пределах 1 – 100), двухмерные (2-D) (например, запись типа  $10 \times 50[10:20, -10:10]$  обозначает таблицу (матрицу) из 10 столбцов (данные – от 10 до 20) и 50 строк (данные – от – 10 до 10)) или трехмерные (3-D);
- Interpolate – опция интерполяции (определение значения функции в промежуточной точке по ее значениям в двух соседних);
- Extrapolate – опция экстраполяции (определение значения функции в дополнительной точке по ее значениям в двух предыдущих).

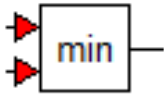
Подготовка блока Map с данными MAP-файла можно проводить с использованием помощника Lookup Table wizard, вызываемого командой Tools/Import/Lookup Table; в качестве примера можно использовать файл Commsim\Samples\Amam.map.



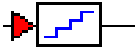
**Max** – определитель максимального из двух входных сигналов.



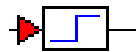
**Merge** – блок условного оператора: при  $b = 1$  выходной сигнал  $y = t$ , если  $|t| > 1$  и  $y = f$ , если  $|t| < 1$  (названия выводов – от слов boolean, true и false).



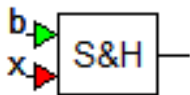
**Min** – определитель минимального из двух входных сигналов.



**Quantize** – квантователь с заданным уровнем (шагом) квантования (Resolution Q); при положительном значении параметра квантуемый сигнал округляется в меньшую сторону (например, при значении входного сигнала 1,9 на выходе будет 1, при – 1,9 будет 2), при отрицательном значении – наоборот. Блок может быть описан выражением:  $y = [\text{целая\_часть}(x/\Delta y)] \Delta y - q$ , где  $\Delta y$  – величина шага квантования;  $q = \Delta y$ , если  $\text{sign}(\Delta y) \Delta x < 0$ , иначе  $q = 0$ .

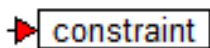


**Relay** – модель трехпозиционного реле с заданным порогом срабатывания Dead Band (DB); выходной сигнал  $y = -1$  при входном  $x < DB/2$ ,  $y = 1$  при  $x > DB/2$  и  $y = 0$  в других случаях.



**SampleHold** – модель схемы выборки и хранения (СВХ) для запоминания на требуемый промежуток времени мгновенного значения входного сигнала  $x$  с возможностью установки начального состояния Initial Condition (IC). Логический вход блока  $b$ , на который подается управляющий сигнал, работает по уровню. Если модуль управляющего сигнала (на входе  $b$ ) больше или равен 1, то блок работает в режиме выборки (повторяет входной сигнал  $x$ ); если модуль управляющего сигнала меньше 1, то блок работает в режиме хранения. В математике подобная операция называется экстраполяцией нулевого порядка. В аналого-цифровой технике устройства СВХ удерживают в неизменном виде выбранное мгновенное значение аналогового сигнала для обеспечения корректной работы большинства АЦП.

#### П1.24. Блоки оптимизации (Blocks/Optimization)



**Constraint** – элемент оценки качества системы с обратной связью с индексом ID, содержащей интегратор, элемент задержки и блок

имитации объекта управления (например, П- или ПИ-регулятор). В более общем случае блок является датчиком для неявных решателей (решение итерационными методами уравнений, не содержащих производных (неявных уравнений)); используется в паре с блоком ParameterUnknown.

→ **cost** **Cost** – блок оценки качества, параметрической оптимизации (без параметров); является датчиком для оптимизаторов с автоматическим подбором параметров системы в итерационном процессе; также используется в паре с блоком по ParameterUnknown.

→ **globalConstraint** **GlobalConstraint** – блок оптимизации с использованием файла с именем VOPT.DLL, который должен постоянно находиться в каталоге Commsim; является дополнительным датчиком, который можно использовать для создания связи с оригинальным оптимизатором, написанным на языке высокого уровня (Си++, Паскаль, Фортран и др.) и откомпилированным в dll-библиотеку.

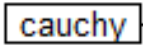
→ **parameterUnknown** **ParameterUnknown** – блок рассчитан для совместной работы с блоком Cost; используется оптимизаторами для подстановки в блок-схему подбираемых ими в итерационном процессе повторного моделирования значений параметров системы, минимизирующих заданную функцию.

→ **unknown** **Unknown** – блок рассчитан для совместной работы с блоком Constraint, т. е. используется неявными решателями для подстановки в блок-схему подбираемых ими в итерационном процессе значений неизвестных величин. Условием возможности получения численного решения неявного уравнения является прямая или косвенная зависимость значения сигнала на входе блока (или блоков) Constraint от выходных значений всех других используемых блоков.

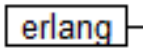
### П1.25. Генераторы шума (Blocks/Random Generator)



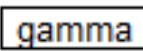
**Beta** – генератор случайных чисел стандартного  $\beta$ -распределения. Бета-распределение является непрерывным распределением с конечными верхним и нижним граничными значениями.



**Cauchy** – генератор случайных чисел подчиняющихся закону распределения Коши. Распределение Коши – это неограниченное непрерывное распределение с ярко выраженным центральным максимумом и достаточно широкими хвостами, коорые имеют гораздо больший вес, чем хвосты нормального распределения.



**Erlang** – генератор случайных чисел с распределением Эрланга; используется для исследования вероятностных процессов, например, в телефонии. Распределение Эрланга является непрерывным распределением,ограниченным снизу.



**Gamma** – генератор случайных чисел Гамма-распределения. Гамма-распределение – это непрерывное распределение, ограниченное снизу.



**Gaussian** – источник шума с нормальным (гауссовым) распределением; параметры: Mean, Standard Deviation – математическое ожидание и среднеквадратичное отклонение (стандартное отклонение или разброс) нормального закона распределения. При повторном моделировании псевдослучайная последовательность повторяется, т. е. значения ее выборок для  $n$ -ого шага моделирования неизменны. Для ее изменения необходимо произвольно изменить параметр Random в окне команды Simulation Properties/Seed Preferences.



**Uniform** – источник шума с равномерным распределением в диапазоне 0 – 1; параметр Time Delay – задержка запуска, с. Равномерное распределение является непрерывным распределением, ограниченным с обеих сторон, т. е. значение лежит в интервале [min, max].



**pareto** — **Pareto** – генератор случайных чисел подчиняющихся закону распределения по Парето. Распределение по Парето является непрерывным распределением, ограниченным снизу.

**PRBS** — **PRBS** (Pseudo Random Block Sequence) – источник бинарной последовательности заданной амплитуды с псевдослучайной и равновероятной сменой знака; при повторном запуске регистр сдвига генератора обнуляется, в результате вся псевдослучайная последовательность повторяется, и знак ее выборки для  $n$ -ого шага моделирования остается неизменным; параметры:

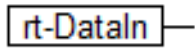
- Register Length – разрядность регистра; параметр определяет период повторения последовательности; если  $N$  – количество разрядов регистра, то псевдослучайная последовательность повторится через  $(2^N - 1)$  выборки;
- Amplitude – амплитуда;
- Sample Interval – пауза между последовательностями (период синхронизации регистра сдвига).

**rayleigh** — **Rayleigh** – генератор случайных чисел распределения Рэлея. Распределение Рэлея – это непрерывное распределение, ограниченное снизу. Оно представляет собой особый случай распределения Вейбулла.

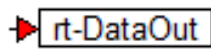
**triangular** — **Triangular** – генератор случайных чисел треугольного распределения; в диалоговом окне задаются начальное (Start), максимальное (Peak) и конечное (End) значения распределения. Треугольное распределение – это непрерывное распределение, ограниченное с обеих сторон, которое часто применяется при недостаточной информации или полном ее отсутствии.

**weibull** — **Weibull** – генератор случайных чисел подчиняющихся закону распределения Вейбулла. Распределение Вейбулла – это непрерывное распределение, ограниченное снизу. Так как оно дает одно из предельных распределений для экстремумов, оно также называется распределением Фреше и распределением Вейбулла – Гнеденко.

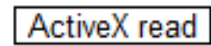
## П1.26. Блоки синхронизации (Blocks/Real Time)



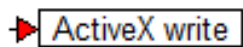
**Rt-DataIn** – блок для согласования работы с аппаратными устройствами ввода/вывода (УВВ) при дополнительном использовании специальных программ (Commsim/Real-Time, Commsim/Real-TimePRO или Commsim DACQ); осуществляет чтение в блок-схему сигнала, поступающего из аппаратных средств сопряжения компьютера с внешними объектами; необходимые при этом настройки создаются при выполнении команды File/Real Time Config.



**Rt-DataOut** – аналогично Rt-DataIn, но для записи во внешнее УВВ.

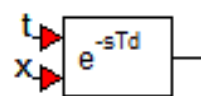


**ActiveX read** – блок чтения данных в реальном масштабе времени из приложения Windows (Excel, Microsoft Visual Basic и др.), организованных с помощью программы ActiveX.



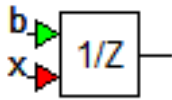
**ActiveX write** – аналогично ActiveX read, но в режиме записи.

## П1.27. Блоки задержки (Blocks/TimeDelay)



**TimeDelay** – блок задержки аналогового входного сигнала  $x$  на время, задаваемое по входу  $t$ ; в блоке используется буфер с памятью, работающий по принципу FIFO; параметры:

- Initial Condition (IC) – начальные условия; этим значением заполняется буфер на первом шаге моделирования;
- Max Buffer Size (M) – максимальный размер буфера в единицах шагов модельного времени, т. е. максимальное время задержки, определяется произведением этого, параметра на интервал дискретизации, который определяется параметром Min Step Size команды Simulate/Simulation Properties/Integration Method.



**UnitDelay** – блок задержки выборки входного сигнала  $x$ ,

тактируемый логическим сигналом  $b$ ; между синхроимпульсами выходное значение поддерживается неизменным с помощью экстраполятора нулевого порядка; при моделировании дискретных систем блок выполняет такие же функции, что и интегратор при моделировании непрерывных; параметры:

- Checkpoint State – опция сохранения состояния блока для очередного сеанса моделирования;
- Initial Condition – начальное значение выходного сигнала до момента подачи синхроимпульсов; запоминается одновременно в регистре и экстраполяторе.

### П1.28. Трансцендентные функции (Blocks/Transcendental)

→ **acos** – **Acos** – арккосинус входного сигнала, рад.

→ **asin** – **Asin** – арксинус входного сигнала, рад.

→ **atan2** – **Atan2** – вычисление фазы между двумя входными сигналами, рад.

→ **bessel** – **Bessel** – преобразователь входного сигнала в функцию Бесселя первого рода заданного порядка Order.

→ **cos** – **Cos** – косинус входного сигнала, рад.

→ **cosh** – **Cosh** – косинус гиперболический входного сигнала, рад.

→ **exp** – **Exp** – экспонента ( $y = e^x$ ).

→ **ln** – **Ln** – натуральный логарифм (с основанием  $e = 2,7182828$ ) входного сигнала.

→ **log10** – **Log10** – десятичный логарифм (с основанием 10) входного сигнала.

→ **sin** – **Sin** – синус входного сигнала, рад.

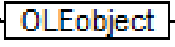
- ➔  — **Sinh** – синус гиперболический входного сигнала, рад.
- ➔  — **Sqrt** – корень квадратный из входной положительной величины.
- ➔  — **Tan** – тангенс входного сигнала, рад.
- ➔  — **Tanh** – тангенс гиперболический входного сигнала, рад.

### П1.29. Другие компоненты меню **Blocks**

➔  — **UserFunction** – подготовка функций пользователя в виде файлов DLL (Dynamic Link Library – динамически загружаемая библиотека), с использованием помощника DLL Wizard.

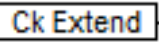
➔  — **Expression** – блок подготовки моделей на языке СИ; параметры:

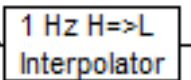
- Expression Text – окно для ввода текста программы;
- Parse Errors – поиск и регистрация синтаксических ошибок.

➔  — **OLEobject** – блок для связи текущей схемы с объектами из приложений Windows; позволяет оперативно изменять параметры используемого объекта.

### П1.30. Блоки согласования (**Comm/Multirate Support**)

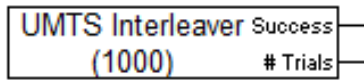
➔  — **Clock Edge** – формирователь синхроимпульсов из медленно меняющихся сигналов.

➔  — **Clock Extend** – формирователь синхроимпульсов пониженной частоты с опцией Local Sim Rate of Next Block (формирование шага моделирования для следующего блока), для случая применения составных блоков.

➔  — **Interpolator** – интерполятор повышенной точности; параметры:

- Local Sim Rate of Next Block – опция формирования шага моделирования для следующего блока (при применении составных блоков);
- Mode – режим формирования «нового» шага моделирования путем его увеличения (High to Low) или уменьшения (Low to High).

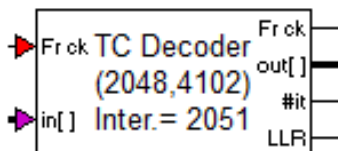
### П1.31. Турбокодеры (Comm/Turbo Codes)



**TS (Turbo Code) Interleaver Generator** –

генератор для испытания турбокодеров и декодеров; выходы: Success – выходной сигнал и признак успешных испытаний при установлении потенциального сигнала 1; #Trials – выход счетчика числа испытательных пусков; параметры:

- Interleaver Size – глубина перемежения  $K$ ; для кодеров, реализующих алгоритм S-Random, она выбирается в пределах 1 – 100000, для UMTS кодеров – 40 – 5114;
- Interleaver Type – тип перемежения: UMTS – для UMTS-кодеров или S-Random – для кодеров с алгоритмом S-Random;
- Minimum Spacing – минимальное значение пусков в режиме S-Random; ориентировочно определяется как  $K^{1/2}/2$ ;
- Max Trials – максимальное количество пусков в режиме S-Random;
- Save As, Browse File, Interleaver File Path – команды работы с файлами настройки.



**Turbo Code Decoder** – турбодекодер для

декодирования PCCC (Parallel Concatenated Convolutional Code – параллельного составного сверточного кода) и RSC (Recursive Systematic Convolutional – рекурсивного систематического сверточного). Как и все подобные устройства, декодер принимает закодированный входной вектор размера  $n$  и выводит

декодированный вектор данных размера  $k$ . Выводы:  $Fr\ sk$  – тактовые вход и выход;  $in[ ]$  – входной код размера  $n$ ;  $out[ ]$  – выходной код размера  $k$ ;  $\#it$  – число выполненных итераций декодирования; LLR (Log Likelihood Ratio) – минимальное количество регистрируемых отношений правдоподобия; параметры:

- Output Block Size – размер выходного кода  $k$  (1 – 100000);
- Trellis Termination – использование сигнально-кодовой решетки (конструкции): Not Terminated – биты завершения решетки не используются; размер блока после перемежения равен входному блочному размеру ( $k$ );
- Terminate Upper – выбор этой опции указывает, что первый кодер должен закончить работу с использованием дополнительных битов; в этом случае размер блока после перемежения увеличивается от  $k$  до  $(k + L - 1)$ , где  $L$  – длина ограничения кода;
- Terminate Both – выбор этой опции указывает, что первый и второй кодеры должны закончить работу с использованием дополнительных битов (согласно требованиям UMTS спецификации); в этом случае размер блока после перемежения равняется входному блочному размеру;
- Generator #1 – код генератора обратной связи для обоих RSC кодеров (в восьмеричном формате);
- Generator #2 – код генератора упреждения для обоих RSC кодеров (в восьмеричном формате);
- Constraint Length – ограничение длины ( $L$ ) кода RSC кодеров (значение  $L$  ограничено значениями 3, 4 и 5; в диалоговом окне параметр обозначен буквой K);
- Decoder Type – типы декодеров:
  - а) Linear Approx. log-MAP – декодер линейного приближения с регистрационной картой для вычисления внутренней метрики декодера;
  - б) Max-log-MAP – декодер с использованием метода «максимальная регистрационная карта» для вычисления внутренней метрики декодера; самый

быстрый алгоритм декодирования, однако потери составляют около 0,4 дБ по сравнению с Linear Approx. log-MAP;

в) Constant-log-MAP – декодер с использованием метода «постоянная регистрационная карта»; также относится к классу быстродействующих алгоритмов (потери кодирования составляют около 0,03 дБ по сравнению с Linear Approx. log-MAP);

г) Log-MAP (piecewise linear) – декодер кусочно-линейного приближения; алгоритм характеризуется повышенным быстродействием по сравнению с Linear Approx. log-MAP;

д) Log-MAP (C calls) – декодер с использованием стандартной функции C; отличается самым низким быстродействием при достаточно высоком качестве декодирования;

- Code Rate – скорость выходного кода: двойная (Rate 1/2), тройная (Rate 1/3).

- Halting Mode – режимы завершения декодирования:

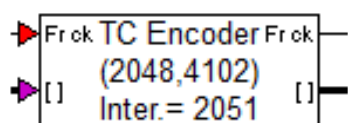
- LLR Halting – остановка декодирования после выполнения заданного количества итераций с формированием сигнала останова на выходе LLR;

- Perfect Halting – остановка после обнаружения нулевой ошибки на дополнительном выходе модели, формируемом в этом режиме;

- Halt LLR Threshold – останов по заданному значению LLR (типичное значение этого параметра – 10);

- Max Iterations – максимальное число итераций декодирования (типичные значения – 4 – 30);

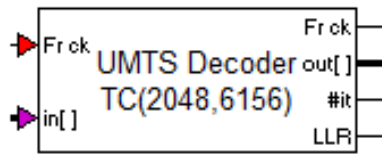
- Select File, Browse File, Interleaver File Path – команды работы с файлами настройки.



**Turbo Code Encoder** – кодер PCCC и RSC кодов;

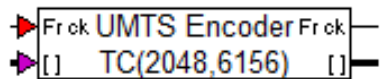
выводы: Fr ck – вход и выход синхросигналов фрейма; [ ] – вход фрейма

(блока) информационного кода размером  $k$  и выход размером  $n$ ; набор параметров аналогичен блоку Turbo Code Decoder.



**UMTS Turbo Decoder** – турбодекодер

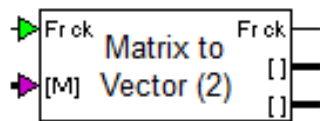
спецификации UMTS для декодирования PCCC кодов скоростью  $1/3$  и размером  $3k + 12$ , где  $k$  – размер информационного блока. Обозначения выводов и набор параметров аналогичны блоку Turbo Code Decoder.



**UMTS Turbo Encoder** – турбокодер

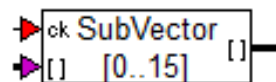
спецификации UMTS для генерирования PCCC кодов; параметры аналогичны TC (Turbo Code) Decoder.

### П1.32. Векторные операторы (Comm/Vector Ops)



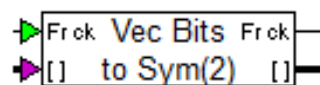
**Matrix to Vector** – блок осуществляющий

преобразование матрицы размера  $M \times N$  (вход  $[M]$ ) в  $N$  или менее независимых векторов (выходы  $[ ]$ ), количество которых задается параметром Number of Output Vectors; выходы Fr ck – тактовые вход и выход.



**SubVector** – блок для извлечения вектора столбца

размера  $N$  из большего или равного по размеру вектора столбца блока  $L$ . Размер  $N$  выходного вектора задается параметром Output Vector Size, а его местоположение в пределах входного вектора – параметром Offset (от 0 до  $N - 1$ ).



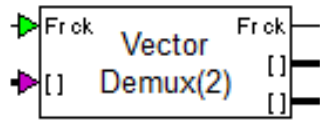
**Vector Bits to Symbol** – блок для объединения

смежных элементов вектора столбца в меньший вектор столбца, составленный из символов. Параметры:

- Number of Bits per Symbol – число  $K$  бит на символ;



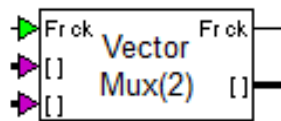
– Bit Order – порядок объединения битов начиная с младшего бита (LSB First) или старшего (MSB First).



**Vector Demux** – блок преобразования вектора

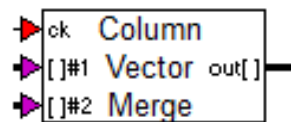
столбца в 2 или более меньших по размеру векторов. Параметры:

- Number of Output Vectors – количество  $K$  выходных векторов;
- Padding Mode – режим дополнения: выключен (Off) или включен (On) с указанием значения дополнения (Pad Value) в последнем случае.



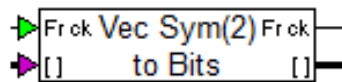
**Vector Mux** – блок объединения  $N$  входных векторов

(задается параметром Number of Input Vectors) в один.



**Vector Merge** – блок добавляет в конец вектор размера

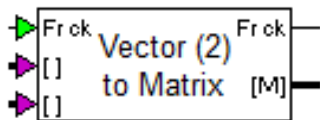
$M$  (вход  $[ ]#1$ ) вектор размера  $L$  (вход  $[ ]#2$ ). На выходе  $out[ ]$  формируется вектор размера  $N = L + M$ .



**Vector Symbol to Bits** – преобразование вектора

символов в вектор битов; параметры:

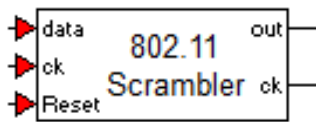
- Number of Bits per Symbol – число  $K$  бит на символ;
- Bit Order – порядок расчленения символов, начиная с младшего бита символьной группы (LSB First) или старшего (LSB Last).



**Vector to Matrix** – преобразование двух и более (до

16) входных векторов размера  $M$  (количество векторов  $N$  задается параметром Number of Input Vectors, входы  $[ ]$ ) в вектор размера  $M \times N$  (выход  $[M]$ ).

### П1.33. Беспроводные компоненты спецификации 802.11 (Wireless/802.11)



**802.11 Scrambler** – устройство скремблирования;

выводы: data, out – вход и выход; ck – вход и выход синхросигналов; Reset – сброс регистра сдвига. Параметр Register Initialization – виды инициализации регистра сдвига согласно спецификации 802.11: с длинной преамбулой (Long Preamble) (заносимое число hex 0×6C (двоичное 1101100)) и короткой (Short Preamble) (заносимое число – hex 0×1B (001101)).

Основным элементом рассматриваемого блока является 7-разрядный регистр сдвига (рис. П1.40) с логическими элементами ИСКЛЮЧАЮЩЕЕ ИЛИ (XOR) в цепи обратной связи, выполненной в соответствии с многочленом  $p(x) = x^7 + x^4 + 1$  (двоичный эквивалент 10010001).

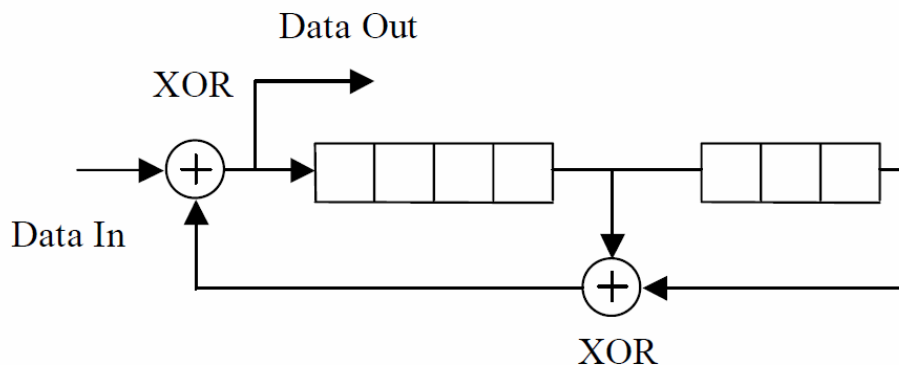
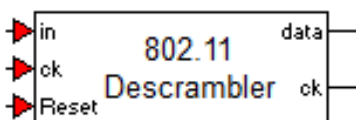


Рисунок П1.40. Схема регистра сдвига блока 802.11 Scrambler



**802.11 Descrambler** – дескремблирующее

устройство спецификации IEEE 802.11 на базе 7-разрядного регистра сдвига (рис. П1.41); выводы: in, data – вход и выход; назначение других выводов и набор параметров аналогичны блоку 802.11 Scrambler.

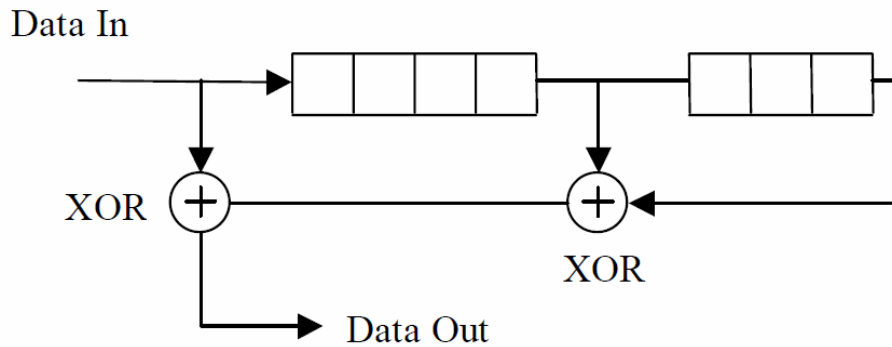


Рисунок П1.41 – Схема регистра сдвига блока 802.11 Descrambler

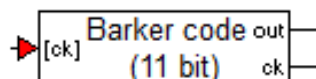


**802.11 Hop Generator** – блок управления

генератором дискретных частот Frequency Hop для формирования сигналов с псевдослучайным изменением несущих частот радиоканала, значения которых зависят от региона (страны) в соответствии с требованиями стандарта 802.11. Выводы: Ptrn # – вход управления выбора частоты; [ck], ck – вход и выход синхронизации; Hop Indx – значение индекса частоты; Chan # – номер канала.

Параметры:

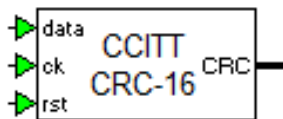
- Region – разбиение на географические регионы согласно спецификации 802.11: Северная Америка/Канада, Европа, Япония, Испания и Франция;
- Timing Mode – режимы синхронизации: внутренняя (Internal) или внешняя (External);
- Initial Hop Index – начальное значение индекса частоты;
- Hop Rate – скорость переключения частот в режиме Internal Timing, кГц;
- Start Time – время запуска в режиме Internal Timing, с.



**Barker Sequence** – формирователь кода Баркера с

квазиидеальными корреляционными характеристиками; используется, например, в системах множественного доступа с кодовым разделением каналов CDMA в соответствии со спецификацией IEEE 802.11 2,4 ГГц DSSS (код длиной  $N = 11$ ). Параметры:

- Sequence Length – длина  $N$  кода Баркера; выбирается из ряда: 3, 4, 5, 7, 11 и 13;
- Sequence Offset – смещение кодовой последовательности Баркера (от 0 до  $N - 1$ );
- Output Mode – форма выходного сигнала: двухуровневая (Bilevel)  $(-1, 1)$  или однополярная (Binary)  $(0, 1)$ ;
- Timing – синхронизация: внутренняя (Internal) и внешняя (External);
- Bit Rate – скорость кодовой последовательности, бит/с; выбирается только в режиме Internal Timing;
- Start Time – время старта, с.



**CRC-16 Generator** – 16-разрядный формирователь

проверочных битов CCITT-16 CRC (Cyclic Redundancy Code – циклический избыточный код), устанавливаемый как на передающей, так и на приемной стороне. Сравнение значений проверочных битов, вычисленных с помощью идентичных блоков на передающей и приемной стороне, позволяет выявить ошибки при передаче кода по линии связи и часть из них исправить.

Блок выполнен на базе 16-разрядного регистра сдвига (рис. П1.42) с организацией обратной связи в соответствии с многочленом  $p(x) = x^{16} + x^{12} + x^5 + 1$ . Выходные данные в виде 16-разрядного проверочного кода снимаются с выходов триггеров регистра.

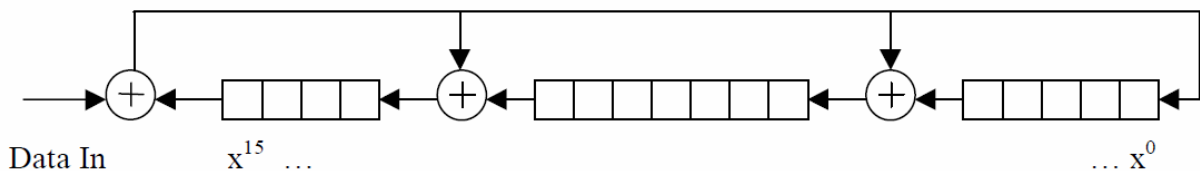
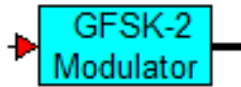


Рисунок П1.42. Схема 16-разрядного регистра сдвига блока CRC-16 Generator

Назначение выводов модели: data – вход информационного кода; ck – вход синхронизации; rst – вход сброса регистра сдвига; CRC – выход последовательности проверочных битов. Параметр Output Mode – режим

работы блока: регулярный с прямым выводом данных (Regular) или в виде дополнения (One's Complement) (используется в 802.11).



**GFSK-2** – разновидность FSK модулятора с

преобразователем  $(2x - 1)$  бинарного (с возвратом к нулю) сигнала в двухуровневый (без возврата к нулю) входной сигнал NRZ  $[-1, +1]$ ; содержит (см. рис. П1.43) частотный модулятор с девиацией частоты  $Fmdev$  (задается одноименной переменной), на входе, которого включен фильтр Gaussian FIR с частотой среза  $Cutoff = BT \times SymRate$ , где  $SymRate = R$  – скорость входного потока, симв/с;  $BT$  – задержка в единицах символов; оба параметра задаются одноименными переменными. Кроме того, параметр  $BT$  должен выбираться из условия  $BT = (N - 1)/2$ , где  $N$  – число шагов моделирования. В качестве блока  $(2x - 1)$  используется полиномиальный источник из раздела преобразователи сигналов. Приведенные значения настроек в виде переменных соответствуют требованиям спецификации IEEE 802.11 для режима 1 Мбит/с при девиации частоты 160 кГц.

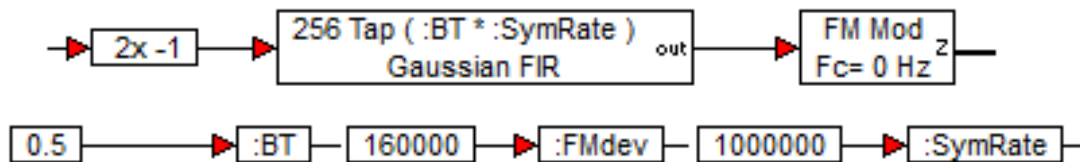
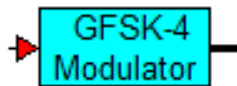


Рисунок П1.43. Модулятор GFSK-2



**GFSK-4** – разновидность FSK модулятора; отличается от

GFSK-2 тем, что на входе фильтра включен преобразователь case бинарного сигнала в 4-уровневый  $[-1, -0,3333, +0,3333, +1]$  (рис. П1.44).

Приведенные значения настроек в виде переменных соответствуют требованиям спецификации IEEE 802.11 для режима 2 Мбит/с при девиации частоты 216 кГц.

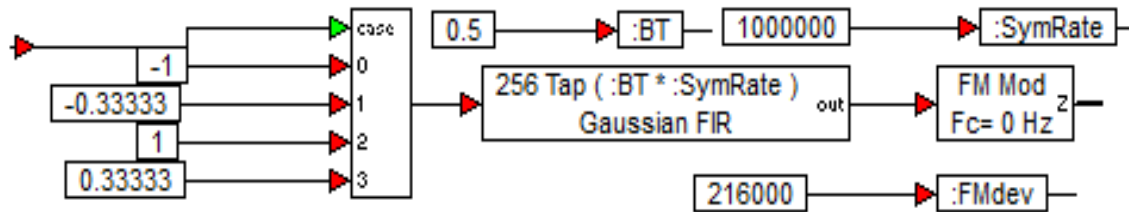
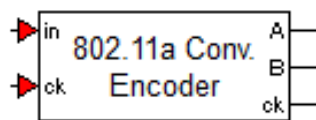


Рисунок П1.44. Модулятор GFSK-4

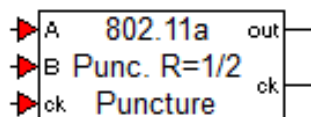
### П1.34. Беспроводные компоненты спецификации 802.11a/g (Wireless/802.11a/g)



**802.11a/g Convolutional Encoder** – сверточный кодер

спецификации IEEE 802.11a/g с параметрами:

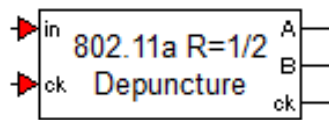
- Generator #1 – значение первого образующего коэффициента в восьмеричном формате; значение по умолчанию – 133 (двоичное 1011011);
- Generator #2 – значение второго образующего коэффициента в восьмеричном формате; значение по умолчанию – 171 (двоичное 1111001);
- Constraint Length – допустимая длина кода  $k$  (по умолчанию – 7).
- Скорость кода  $R = 1/2$ . Выводы: in – информационный вход; ck – вход и выход синхронизации; A, B – выходы (для подключения к блоку 802.11a Puncture).



**802.11a/g Puncture** – преобразователь выходного

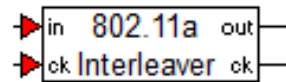
сигнала сверточного кодера по 802.11a/g Convolutional Encoder в последовательный код; выполняет также функцию сжатия и скремблирования данных; выводы: A, B – входы; ck – вход и выход синхронизации; out – выход.

- Code Rate – скорость выходного кода: Rate 1/2, Rate 2/3, Rate 3/4;
- Input Symbol Rate – скорость входного кода (для пары A B).



**802.11a/g Depuncture** – блок, противоположный по

выполняемым функциям предыдущему блоку; назначение «перевернутых» выводов и набор параметров аналогичны 802.11a/g Puncture; добавлен параметр стирания Erasure Value, определяющий значение для всех фиктивных битов, вставленных при восстановлении.

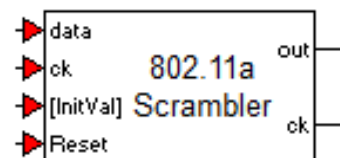


**802.11a/g Interleaver** – перемежитель/деперемежитель

спецификации 802.11a/g; обычно включается после сверточного кодера на передающей стороне и до него – на приемной. Длина обрабатываемых блоков зависит от скорости передачи и представлена следующими значениями: при скорости (6 – 9) Мбит/с длина блока составляет 48 бит, при (12 – 18) Мбит/с – 96 бит, при (24 – 36) Мбит/с – 192 бита, при (48 – 54) Мбит/с – 288 бит.

Параметры:

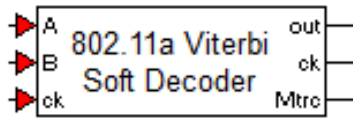
- 802.11a Rate Mode – скорость передачи данных согласно требованиям IEEE 802.11a/g (6 – 54 Мбит/с);
- Mode – работа в режиме перемежителя (Interleave) или деперемежителя (Deinterleave).



**802.11a/g Scrambler** – скремблер/дескремблер IEEE

802.11a/g с 7-разрядным регистром обратной связи; входы: data – вход данных; ck – вход и выход синхронизации; [InitVal] – код инициализации регистра (только в режиме Internal); Reset – обнуление регистра; out – выход.

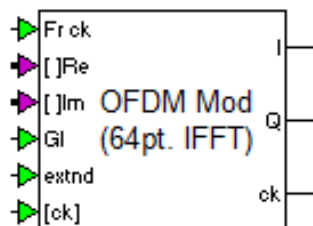
Параметр Register Initialization – режим синхронизации регистра: внешний (External) (по входу [InitVal]) или внутренний (Internal) с использованием выбранного значения параметра Register Init Value.



**802.11a/g Viterbi Decoder** – декодер Витерби

предварительного (мягкого) декодирования для совместной работы с блоком Depuncture. Выводы: A, B – входы кода скорости 1/2 в двухуровневом формате (1, -1) с блока Depuncture; ck – вход и выход синхронизации; out – выход в формате (0, 1); Mtrc – выход сигнала метрики. Параметры:

- Generator #1 – значение первого образующего коэффициента в восьмеричном формате; значение по умолчанию – 133 (двоичное 1011011);
- Generator #2 – значение второго образующего коэффициента в восьмеричном формате; значение по умолчанию – 171 (двоичное 1111001);
- Constraint Length – допустимая длина кода k;
- Trellis Truncation Length – длина (размер)  $M$  сигнально-кодовой конструкции ( $M \leq 96$ );
- Quantization Bits – число битов квантования, используемых в процессе декодирования;
- Select File, Browse File, Metric File Path – команды работы с файлами настройки.



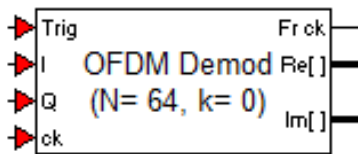
**OFDM Modulator** – широкополосный модулятор

OFDM (Orthogonal Frequency Division Multiplexed); спецификации IEEE 802.11a/g с ортогональным частотно-мультиплексированным разделением основного потока битов на ряд несущих. Выводы: Fr ck, ck – вход и выход синхронизации фрейма; [ck] – синхросигнал селектора охранного интервала; [ ]Re, [ ]Im – вектор размера  $2N$  вещественной и мнимой составляющих поднесущих входного фрейма; GI (Guard Interval) – данные с селектора охранного интервала {0, 1, 2, 3, 4, 5}; extnd – сигнал продленного периода БПФ (разрешение – сигнал высокого уровня); I, Q – выходные сигналы



(с возможностью отображения на плоскости (I, Q)). Вектор данных селектора охранного интервала  $\{0, 1, 2, 3, 4, 5\}$  определяется размером выборки БПФ, деленной соответственно на 2 (для  $GI = 1$ ), 4 ( $GI = 2$ ), 8 ( $GI = 3$ ), 16 ( $GI = 4$ ) и 32 (для  $GI = 5$ ). При этом размер выходного фрейма увеличивается на выбранное значение  $GI$ . Параметры блока:

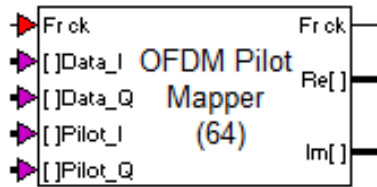
- Timing – синхронизация: внутренняя (Internal) или внешняя (External);
- Use Simulation Sample Rate – синхронизация скорости работы OFDM с шагом моделирования;
- Output Rate – выходная скорость кодовых блоков OFDM; используется в режиме Internal Timing; в этом режиме она по умолчанию равна скорости моделирования;
- FFT Size – размер выборки ОБПФ (16 – 64 Кбит); определяет количество поднесущих;
- Overlap Size – размер перекрытия между последовательностями OFDM (0 – 3 выборки).



**OFDM Demodulator** – демодулятор OFDM;

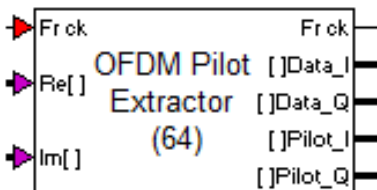
выводы: I, Q – входы; Trig – триггерный вход запуска; ck, Fr ck – вход и выход синхронизации фрейма; Re[ ], Im[ ] – вектор размера  $2N$  действительной и комплексной составляющих поднесущих выходного фрейма. Параметры:

- FFT Size – размер выборки ПБПФ (FFFT); выбирается в диапазоне 16 – 64 Кбит и должен быть равен размеру выборки ОБПФ (IFFT) модулятора передатчика;
- FFT Wraparound Offset (параметр  $k$ ) – циклическое смещение, которое нужно применить к полученным данным до выполнения операции FFFT в целях исключения выборки  $GI$  (защитного интервала).



**OFDM Pilot Map** – блок контроля OFDM

модулятора; назначение выводов и набор параметров аналогичен предыдущему блоку.



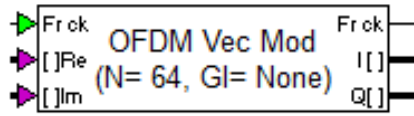
**OFDM Pilot Extract** – блок контроля OFDM

демодулятора путем анализа составляющих Re, Im и сопоставления полученных данных с данными контрольного пилот-сигнала. Выводы: Re[ ], Im[ ] – входные данные (с выхода OFDM демодулятора); Fr ck – вход и выход синхросигнала фрейма данных; [ ]Data\_I, [ ]Data\_Q – выходные данные, соответствующие выходным сигналам демодулятора; [ ]Pilot\_I, [ ]Pilot\_Q – выходные данные, соответствующие контрольному сигналу.

Далее приводятся ориентировочные значения параметров блока:

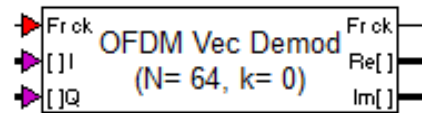
- Input Insertion Point – режим контроля входного сигнала, начиная с нулевой точки (Begin at 0 Location) или с точки  $N/2$  (Begin at  $N/2$  Location), где  $N$  – размер выборки БПФ;
- Number of Data Subcarriers – размер данных I и Q несущих выходного вектора;
- Number of Data Subcarriers – размер экспериментальных данных для I и Q несущих выходного вектора;
- Data Subcarrier Weight – определение веса несущих I и Q в целях их масштабирования;
- Pilot Subcarrier Weight – определение веса несущих I и Q в целях их масштабирования с использованием контрольного сигнала;
- FFT Size – размер БПФ (16 – 64 Кбит), используемый для сортировки данных по значению  $N$ ;

– Select File, Browse File, Subcarrier Mapping File – команды работы с файлами настройки и контроля.



**OFDM Vector Modulator** – широкополосный

OFDM модулятор, отличающийся от ранее рассмотренного OFDM Modulator векторным выходом.



**OFDM Vector Demodulator** – широкополосный

OFDM демодулятор с векторным входом.

### П1.35. Беспроводные компоненты спецификации 802.11b (Wireless/802.11b)



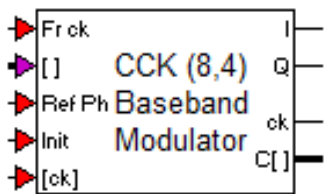
**802.11b High Rate Hop Generator** – блок

управления генератором дискретных частот Frequency Hop для формирования сигналов с псевдослучайным изменением несущих частот радиоканала High Rate (5,5 и 11 Мбит/с) с набором 1 и 2 для Северной Америки и Европы (кроме Франции и Испании). В режиме High Rate для Северной Америки набор 1 включает каналы 1, 6, 11 в различной последовательности, набор 2 – каналы 1, 3, 5, 7, 9 и 11; для Европы (кроме Франции и Испании) набор 1 включает каналы 1, 7, 13 в различных комбинациях, набор 2 – каналы 1, 3, 5, 7, 9, 11 и 13.

Выводы: Ptrn # – ввод выбора номера образца; ck – вход и выход тактового сигнала; Hop Index – индекс частотного диапазона (0 – 78); Hi/Low Rate – флаг высокой/низкой скорости передачи; Chan # – номер канала (1 – 13) в режиме High Rate (5,5 и 11 Мбит/с) и (2 – 80) в режиме Low Rate (1 и 2 Мбит/с).  
Параметры:

– Region – выбор региона: Северная Америка (США и Канада) или Европа (кроме Франции и Испании);

- Timing Mode – способ синхронизации: внутренний (Internal) или внешний (External);
- Hop Set – выбор набора: 1 или 2;
- Initial Hop Index – начальное значение индекса несущей частоты в режиме Internal;
- Hop Rate – скорость переключения (изменения) несущих частот в режиме Internal;
- Start Time – время старта (только в режиме Internal Timing).



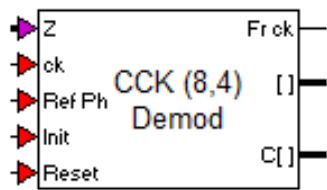
**CCK (Complementary Code Keying) Modulator** –

комплементарный кодо-манипулированный модулятор формирования ортогональных QPSK- и DQPSK-сигналов в режиме High Rate (5,5 и 11 Мбит/с) согласно спецификации 802.11b и обеспечения совместимости с 802.11 для Low Rate (1 и 2 Мбит/с).

Выводы: Fr ck – сигнал синхронизации входного фрейма (вектора); [ ] – входной вектор размером 4 или 8 бит; Ref Ph – вектор {0, 1, 2, 3} фазы; Init – начальное значение фазы; [ck] – сигнал синхронизации битов; I, Q – выходы для отображения на плоскости (I, Q) сигнально-кодовой конструкции; C[ ] – выход сигнального вектора {0, 1, 2, 3}. Параметр CCK Mode – режим формирования CCK сигналов:

- (8, 4) CCK – при 4-битной входной последовательности (для скорости 5,5 Мбит/с); в этом случае четная пара битов 00, 01, 10 и 11 кодируется соответственно значением фазы 0,  $\pi/2$ ,  $\pi$  и  $3\pi/2$ , а нечетная пара тех же битов – соответственно значением фазы  $\pi$ ,  $3\pi/2$ , 0 и  $\pi/2$ ;
- (8, 8) CCK – при 8-битной входной последовательности (для скорости 11 Мбит/с); в этом случае каждая пара битов 00, 01, 10 и 11 кодовых комбинаций (8, 4) CCK кодируется в дифференциальном режиме соответственно значением фазы 0,  $\pi/2$ ,  $\pi$  и  $3\pi/2$ .

- Timing Mode – способ синхронизации: внутренний или внешний;
- Chip Output Rate – скорость выходного потока, бит/с; параметр доступен только в режиме Internal Timing.



**CCK Demodulator** – демодулятор CCK

последовательностей с выхода блока CCK (Complementary Code Keying) Modulator.

Выводы: Fr ck – сигнал синхронизации выходного фрейма (вектора); [ ] – выходной вектор размером 4 или 8 бит; C[ ] – выход сигнального вектора {0, 1, 2, 3}; z – входной комплексный сигнал; ck – вход синхронизации; Ref Ph – строб вектора фазы {0, 1, 2, 3}; Init – начальное значение фазы; Reset – сигнал обнуления счетчика фреймов. Набор параметров аналогичен CCK (Complementary Code Keying) Modulator.

### П1.36. Беспроводные компоненты ближней связи (Wireless/Bluetooth)



**Bluetooth GFSK** – разновидность FSK модулятора с

преобразователем  $(2x - 1)$  бинарного (с возвратом к нулю) сигнала в двухуровневый (без возврата к нулю) входным сигналом NRZ  $[-1, +1]$ ; содержит (см. рис. П1.45) частотный модулятор с девиацией частоты  $F_{mdev}$  (задается одноименной переменной), на входе, которого включен фильтр Gaussian FIR с частотой среза  $Cutoff = BT \times SymRate$ , где  $SymRate = R$  – скорость входного потока, симв/с;  $BT$  – задержка в единицах символов; оба параметра задаются одноименными переменными. Кроме того, параметр  $BT$  должен выбираться из условия  $BT = (N - 1)/2$ , где  $N$  – число шагов моделирования. В качестве блока  $(x - 1)$  используется полиномиальный источник. Приведенные

значения настроек в виде переменных соответствуют требованиям спецификации Bluetooth для режима 1 Мбит/с при девиации частоты 150 кГц.

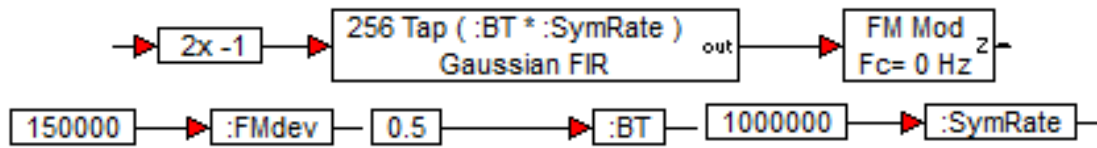
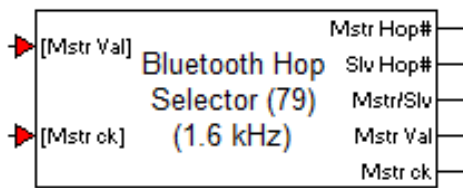


Рисунок П1.45. Схема модулятора Bluetooth GFSK



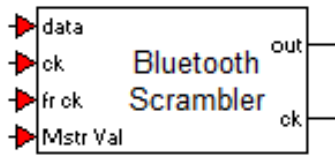
**Bluetooth Hop Generator** – блок управления

генератором дискретных частот Frequency Hop, в соответствии с требованиями стандарта Bluetooth.

Выводы: [Mstr Val] – вход счетчика для определения времени работы (активности) блока в режиме Master (необязательный параметр); [Mstr ck] – вход тактового сигнала в режиме Master (необязательный параметр); Mstr Hop# – номер используемого канала в режиме Master (0 – 78 или 0 – 23); Slv Hop# – номер используемого канала в режиме Slave (Раб) (0 – 78 или 0 – 23); Mstr/Slv – признак (флаг) режима работы блока (0 = Master; 1 = Slave); Mstr Val – выход счетчика времени работы блока в режиме Master (228 временных интервалов общей длительностью 625 мкс по умолчанию); Mstr ck – выход тактового сигнала в режиме Master (через каждые 625 мкс). Параметры:

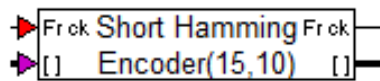
- Device Address – шестнадцатеричный адрес устройства;
- Timing Mode – способ синхронизации: внутренний (Internal) или внешний (External);
- Hop Mode – выбор режима формирования частотных каналов: последовательность из 79 шаблонов (79 Hops) (используется в США и в большинстве стран Европы) или из 23 (23 Hops) (Япония, Испания и Франция);

- Ink Master Clock – начальное значение частотного канала для Bluetooth Master; параметр доступен только в режиме Internal Timing;
- Hop Rate – скорость переключения частот (по умолчанию – 1,6 кГц);
- Start Time – время старта (только в режиме Internal Timing).



**Bluetooth Scrambler** – скремблер/дескремблер

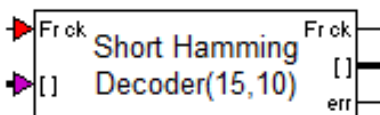
стандарта Bluetooth; выполнен на базе 7-разрядного регистра сдвига с обратными связями в соответствии с многочленом  $p(x) = x^7 + x^4 + 1$ . Входы: data – вход данных; ck – тактовые вход и выход; fr ck – сигнал синхронизации входного фрейма; Mstr Val – сигнал с выхода счетчика времени работы блока Bluetooth Hop Generator в режиме Master; out – выход данных.



**Shortened Hamming Encoder** – формирователь

кода Хемминга (15, 10) с выходной скоростью 2/3 (10/15). Выводы: Fr ck – сигнал синхронизации входного и выходного фрейма (вектора); [ ] – входной (размером 10) и выходной (размером 15) векторы сигналов. Формируемый на выходе блока код Хемминга описывается матрицей:

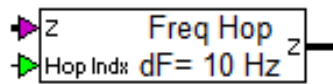
$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}.$$



**Shortened Hamming Decoder** – декодер кода

Хемминга (15, 10). Выводы: Fr ck – сигнал синхронизации входного и выходного фрейма (вектора); [ ] – входной (размером 15) и выходной (размером 10) векторы; err – выход сигнала ошибки.

### П1.37. Универсальные беспроводные компоненты (Wireless/Generic)

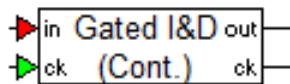


**Frequency Hop** – управляемый формирователь

(генератор) сетки частот для беспроводных систем связи; выводы:  $z$  – комплексные сигналы управления и выхода; Hop Indx – сигнал номера (индекса) канала. Параметры:

- Number of Hops – число фиксированных каналов  $N$ ; используется только для внутренней проверки диапазона частот;
- Lowest Hop Channel – канал с минимальной частотой (нулевой канал) относительно центральной частоты выделенного диапазона, Гц;
- Hop Carrier Spacing – разнос (смещение) несущих частот  $f$  между смежными каналами, Гц;
- Initial Phase – начальная фаза несущей, град.; опция доступна только в режиме Continuous Phase;
- Phase Mode – режим по фазе: непрерывный (Continuous) (например, в случае использования VCO – источника с управляемой частотой) или дискретный (Discontinuous) (например, в многофазных или многодиапазонных генераторах).

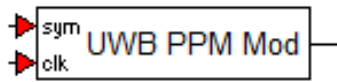
### П1.38. Беспроводные компоненты сверхширополосной связи (Wireless/UltraWideband)



**Gated Integrate & Dump** – блок обеспечивает функции

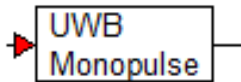
интегрирования и вывода, где интеграционный процесс активен только в течение узкого определяемого пользователем временного окна; пользователь может определить длительность окна, а также время его начала относительно поступления последнего импульса на вход синхронизации ck; выводы: in – вход; ck – вход и выход синхронизации; out – выход.





**UWB (UltraWideband) PPM Modulator** – блок

реализует фазоимпульсную модуляцию (Pulse Position Modulator, PPM), где в роли несущего сигнала используют опорную или базовую, последовательность импульсов, повторяющихся через строго определенные промежутки времени; кодировать информацию можно посредством временного сдвига импульса относительно опорного сигнала в зависимости от значения символа данных  $k$  (в диапазоне  $[0, N - 1]$  на входе sym.



**UWB Pulse** – блок генерирует различные

сверхширокополосные импульсы, включая импульс Гаусса и его производные по времени; блок генерирует импульс каждый раз, при поступлении входного импульса с амплитудой  $> 0,5$ ; также блок обнаруживает знак входного импульса и генерирует выходную форму волны того же знака; формы импульса:

- Monopulse – моноимпульс (видеоимпульс);
- Gaussian Pulse – импульс Гаусса (колоколообразный импульс);
- Gaussian Monocycle – моноцикл Гаусса, т. е. первая производная импульса Гаусса;
- Gaussian Doublet – дублет Гаусса, т. е. вторая производная импульса Гаусса, сигнал также называют «мексиканской шляпой»;
- Gated Sinusoid – стробированная синусоида; генерация синусоидального сигнала на отрезке времени  $[t_0, T]$ , где  $t_0$  – время смещения начала генерации выходного сигнала от момента прихода входного сигнала;  $T$  – длительность генерации выходного сигнала.

## ПРИЛОЖЕНИЕ 2. СРЕДА КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ И АНАЛИЗА СХЕМ ЭЛЕКТРОННЫХ УСТРОЙСТВ NI MULTISIM

### П2.1. Общие сведения о среде NI Multisim

**NI Multisim** – это современная программная среда компьютерного моделирования и комплексного анализа схем электронных устройств. При помощи данной программной среды можно спроектировать и проверить принципиальные схемы электронных устройств и обнаружить в них ошибки еще до создания экспериментальных образцов, а также интерактивно исследовать созданные схемы с помощью средств анализа и виртуальных приборов.

Особенностью среды NI Multisim 10 (MS 10) является наличие контрольно-измерительных приборов, по внешнему виду и характеристикам приближенных к их промышленным аналогам, а также наличие в ее библиотеках более 16000 электронных компонентов и узлов, что позволяет создавать не только простые, но и сложные схемы электронных устройств.

Возможность изменения параметров компонентов схем без остановки процесса моделирования, простота замены и добавления новых компонентов, широкий набор средств анализа – все это способствовало популяризации программной среды NI Multisim 10.

Скачать и установить на своем компьютере программную среду NI Multisim 10 (рис. П2.1) для операционной системы MS Windows, а также скачать описание среды (Multisim™ User Guide) и ее компонентов (Multisim™ Component Reference Guide) можно с сайта компании National Instruments [18, 19] или приобрести компакт-диск с записанной программной средой MS 10 в представительстве корпорации National Instruments в России СНГ и Балтии [18].

Multisim Power Pro Edition

Version 10.1.1



10.1.1

[ni.com/multisim](http://ni.com/multisim)

Initializing: plugin WGEN ...

©2009 National Instruments. All rights reserved.

Рисунок П2.1. Первоначальное окно среды NI Multisim 10

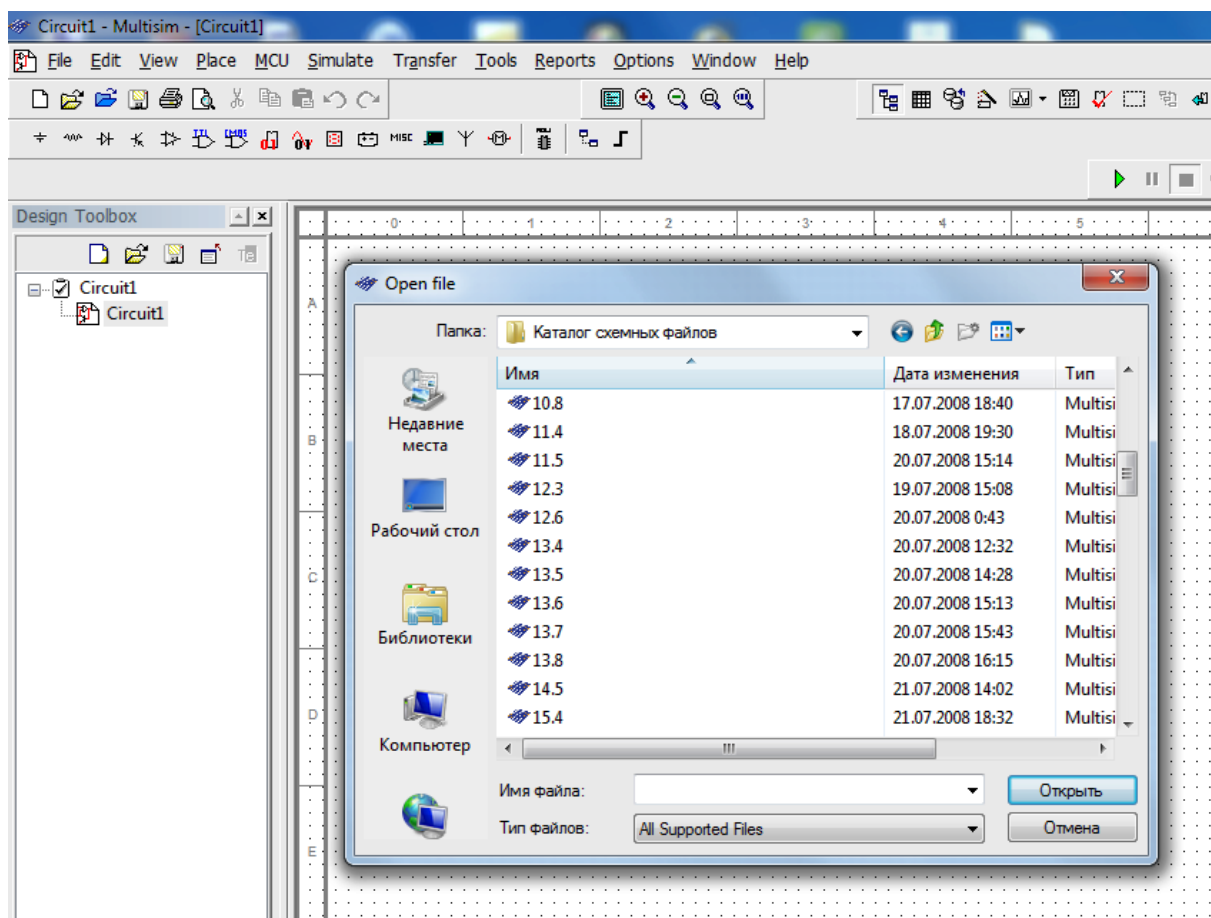


Рисунок П2.2. Окно «Open file» среды NI Multisim 10

## П2.2. Запуск программы Multisim 10

Для того чтобы начать работать с моделями электронных компонентов, запустите программу MS 10, щелкните мышью на кнопке меню File и в открывшемся интерфейсе пользователя – на закладке Open. В открывшемся диалоговом окне выбора программ (рис. П2.2) можно выбрать любой схемный файл, и после щелчка мышью на его имени схема будет выведена на рабочее поле среды MS 10.

После установки программного обеспечения и запуска программной среды MS 10 необходимо провести как глобальные настройки управления свойствами среды, так и настройки пользовательского интерфейса и используемых в работе компонентов и инструментария.

## П2.3. Пользовательский интерфейс среды Multisim 10

Программная среда MS 10 имеет простой пользовательский интерфейс, выполненный в традиционном для MS Windows виде, и состоит из:

- строки заголовка, расположенной в верхней части экрана (рис. П2.3);
- строки главного меню;
- командных строк инструментальной линейки (управляющих кнопок – пиктограмм) библиотек электронных элементов, узлов и приборов.

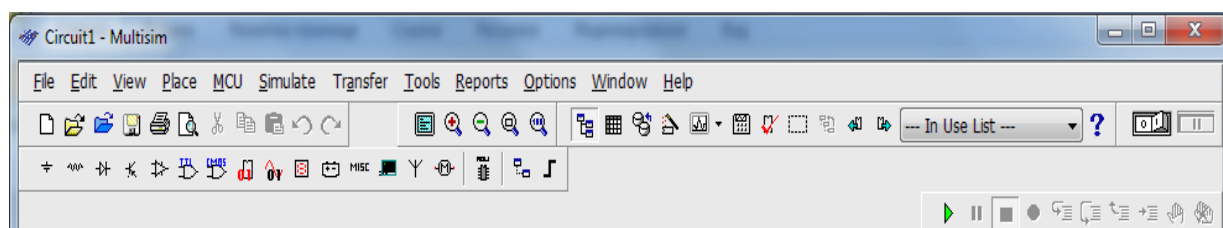


Рисунок П2.3. Пользовательский интерфейс Multisim 10

Пользовательский интерфейс MS 10 можно изменить, выбрав в команде меню Options (Опции) пункт Customize User Interface (Настроить пользовательский интерфейс). В частности, с помощью диалогового окна Customize (Настройка) можно настроить систему меню и назначить горячие

клавиши клавиатуры любой его команде, изменить или создать новые панели инструментов и закрепить их в любом месте.

С помощью опции Options → Sheet Properties (Настройка свойств листа) и ее закладок Circuit (Схема), Workspace (Рабочая область), Wiring (Соединение), PCB (Печатная плата) и Visibility (Доступность) можно изменить свойства каждого схемного документа (листа): размер и цвет схемы, шрифт текста описания схемы, настройки печатной платы, отображение или сокрытие слоев комментариев и другие параметры листа.

Глобальные настройки управления свойствами среды MS 10 выполняются с помощью опции Global Preferences (Глобальные настройки) той же команды Options (рис. П2.4) со следующими закладками:

- Paths (Путь), в диалоговом окне которой можно указать путь к файлам баз данных и осуществить другие настройки;
- Save (Сохранить), в окне которой можно задать период автоматического сохранения и выполнить другие настройки;
- Parts (Компоненты), в диалоговом окне которой можно выбрать режим размещения компонентов и стандарт символов (ANSI или DIN);
- General (Общее), в окне которой можно изменить конфигурацию компонентов, колеса мыши и инструментов соединения компонентов.

Для вывода на экран графических изображений компонентов электронных схем, наиболее приближенных к их условным графическим обозначениям, установленных российскими ГОСТ, необходимо щелкнуть мышью на кружке стандарта DIN и кнопке ОК (рис. П2.4). В этом же окне можно выбрать режим моделирования (Digital Simulation Settings): идеальная или реальная (более точная) цифровая эмуляция.

Кроме команды Options, строка главного меню содержит несколько команд (рис. П2.3), в каждой из которых больше десятка опций со своими закладками и кнопками. Для команд характерны как общеизвестные опции, такие как Open, Save, Copy, Paste и др., так и опции, типичные лишь для среды MS 10.

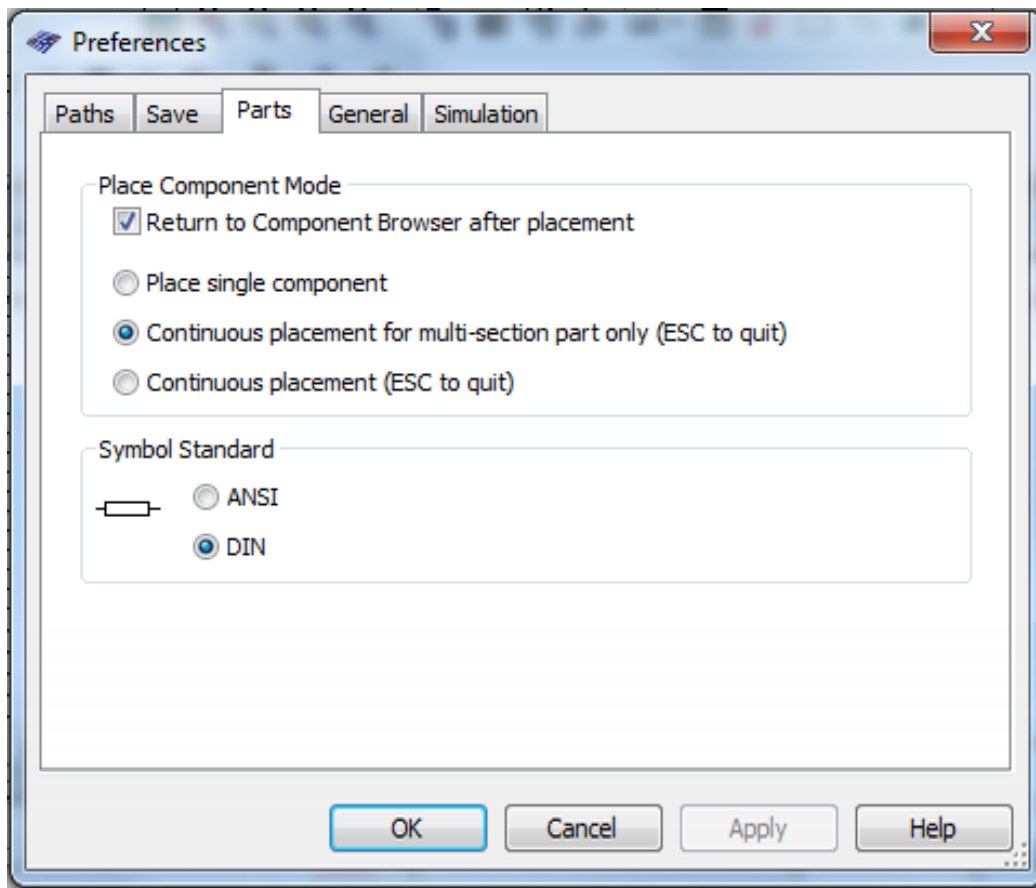


Рисунок П2.4. Окно «Global Preferences» вкладки Options

Отметим назначение отдельных меню и их опций, наиболее часто используемых при проектировании электронных схем.

Опции меню File предназначены для открытия, закрытия, сохранения и печати смоделированных электронных схем. Опции меню Edit (Редактирование) можно использовать для изменения объектов схем на лицевой панели или в блоке программы.

Опции команды View предназначены для вывода по краям рабочего поля размерных полос и точечной сетки на рабочем поле (Show Border, Show Page Bounds, Show Grid), пиктограмм библиотек в инструментальной линейке (Toolbar, рис. П2.5) и других пиктограмм и панелей. Так, при щелчке мышью на опции Virtual в инструментальной линейке библиотек добавляется панель виртуальных приборов.

В меню Place после щелчка мышью на опции AText можно вносить (с помощью клавиш клавиатуры) на рабочее поле программы текстовую

информацию, в том числе на русском языке. К сожалению, кириллический шрифт не поддерживается программой MS 10 и при сохранении (Save) схем русские буквы заменяются соответствующими символами расширенного двоично-десятичного кода EBCDIC.

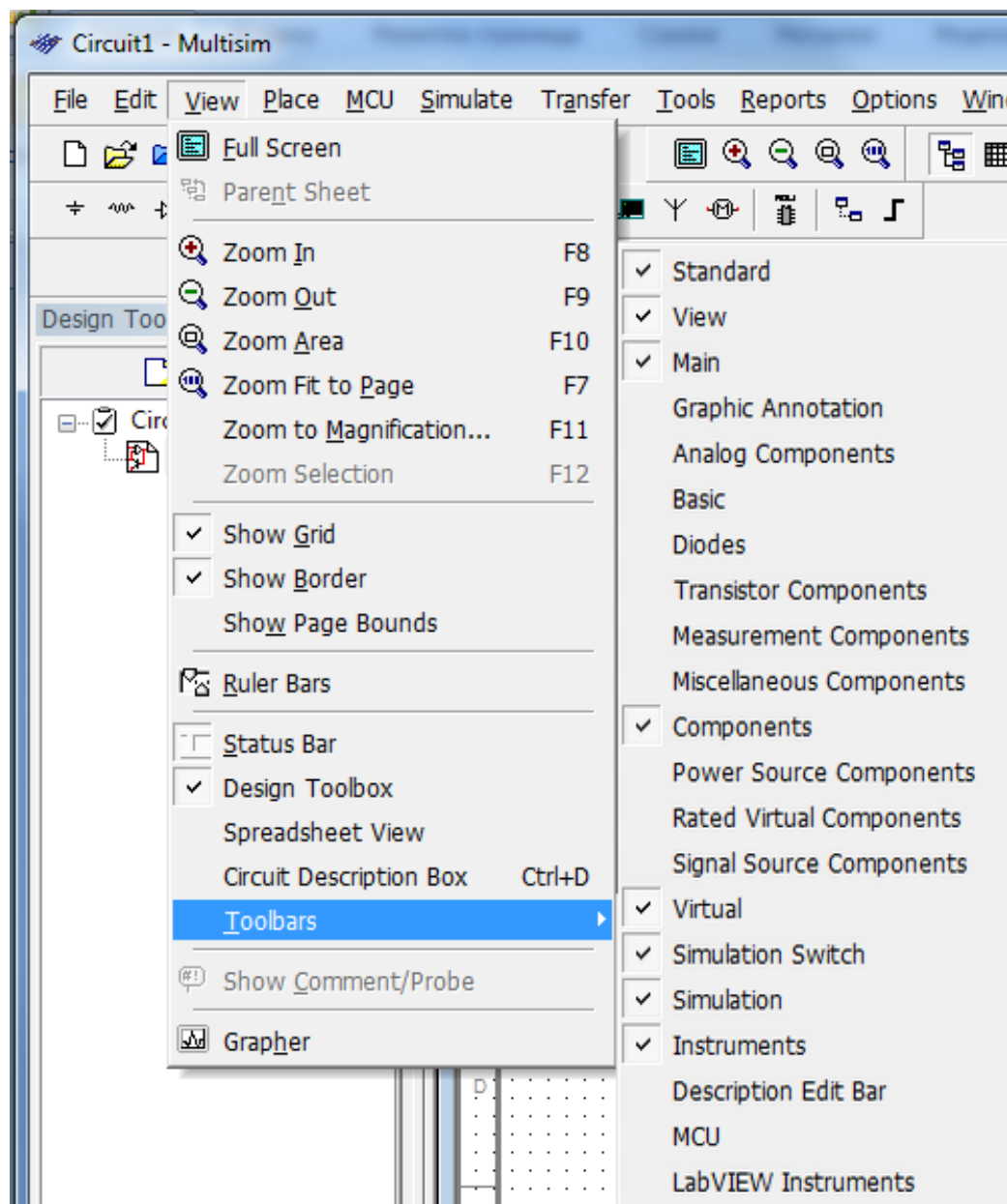


Рисунок П2.5. Окно опций команды View

В меню Transfer (Обмен данными) размещены опции, позволяющие экспортировать разработанные схемы устройств в другие приложения, в частности в программу объемного изображения Ultiboard 10 среды MS 10.

Посредством опций команды Tools (Инструменты) можно конвертировать схемы, спроектированные в средах MS 8 и MS 9, в среду MS 10, выводить на экран из библиотеки MS 10 (Tools → Circuit Wizard) средства разработки усилителей, фильтров, устройств на базе таймера 555 и выводить готовые схемы на операционном усилителе.

Опция Analyses в команде Simulate (Эмуляция) предназначена для выбора методов анализа и установки параметров моделирования (рис. П2.6). Перед запуском эмуляции схемы необходимо убедиться, что в схеме есть источник энергии и заземление.

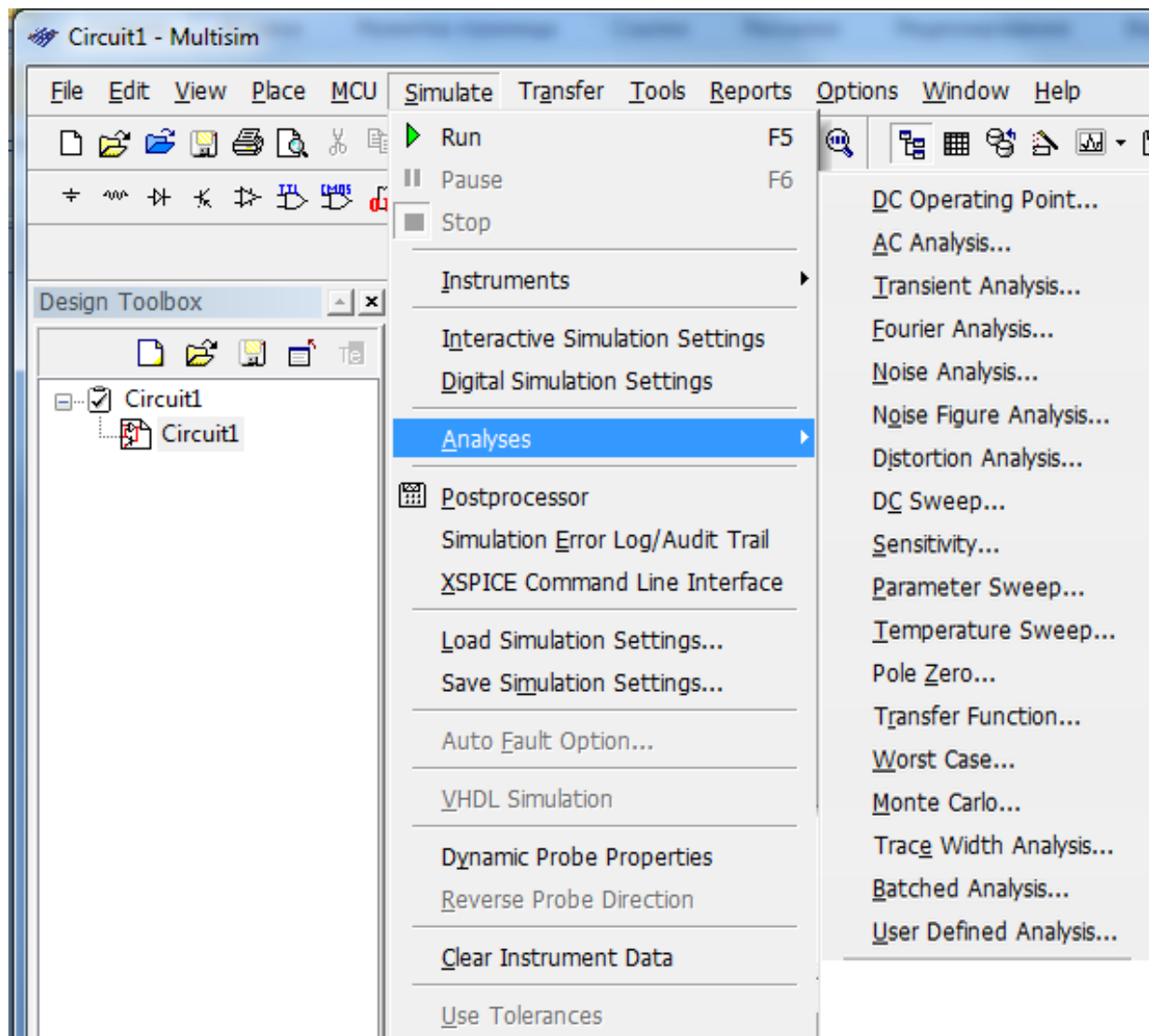


Рисунок П2.6. Окно опций команды Simulate



В среду MS 10 включено множество средств анализа схем устройств (см. список всех функций анализа на рис. П2.6), выбор и настройка параметров которых будут описаны при разработке схем. Результаты анализа отображаются на графиках программ Postprocessor и Grapher.

Настройки эмуляции можно изменить с помощью диалоговых окон опции Simulate → Interactive Simulation Settings (Эмуляция → Настройки интерактивной эмуляции). Некоторые из настроек интерактивной эмуляции приведены на рис. П2.7:

- по умолчанию установлено время окончания эмуляции через  $1e + 30$  с;
- шаг по времени генерируется автоматически.

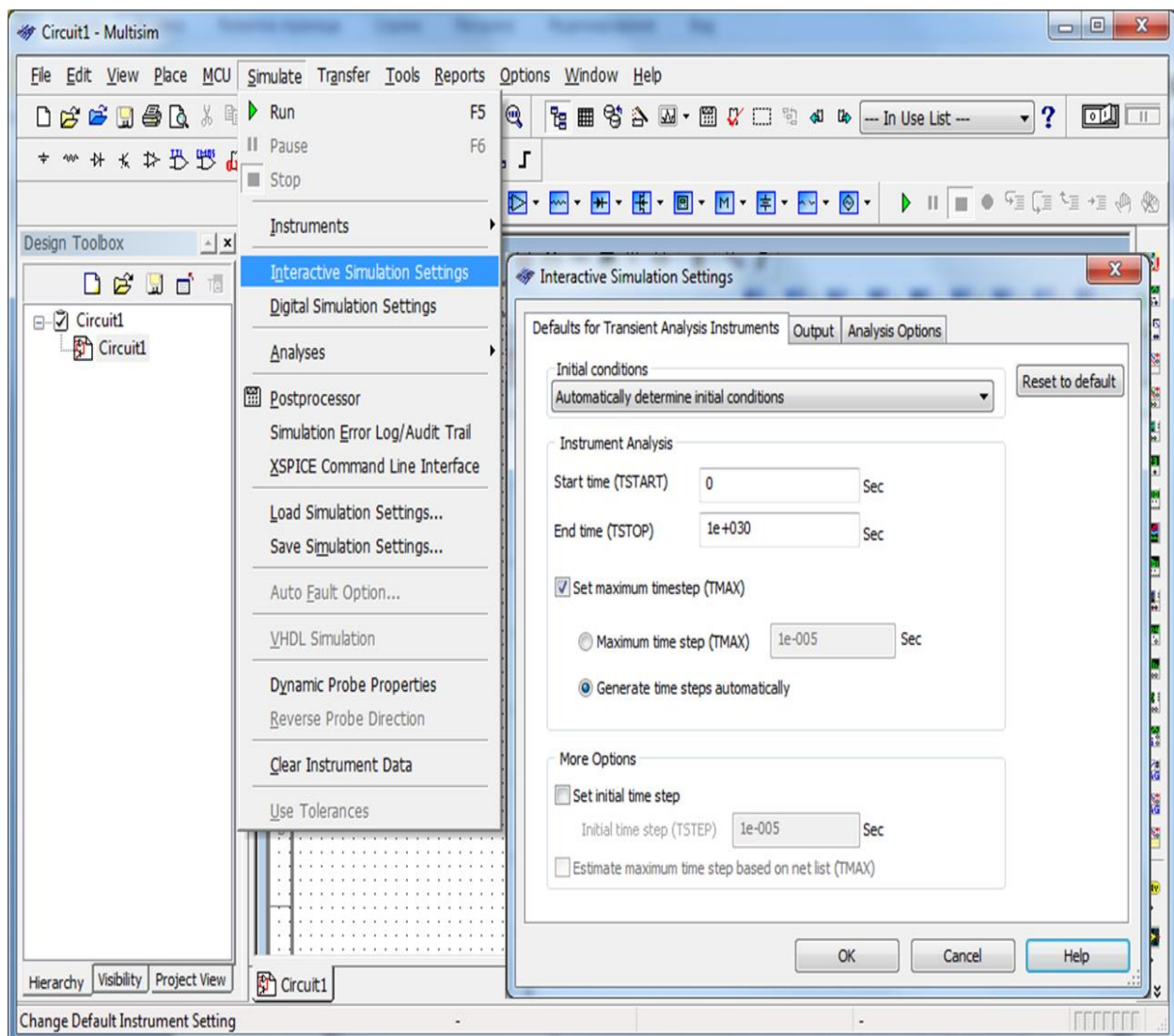


Рисунок П2.7. Окно настроек интерактивной эмуляции

При возникновении ошибок при эмуляции нужно запустить Советника эмуляции (Simulation Error Log → Audit Trail). Типичными ошибками являются слишком малое время шага эмуляции (Time step error) и ошибки в сингулярной матрице (Singular matrix). Для исправления ошибок нужно увеличить максимальный шаг по времени (TMAX, см. рис. П2.7), например с  $TMAX = 1 \times 10^{-5}$  с на  $TMAX = 1 \times 10^{-3}$  с, увеличить относительный допуск ошибки (Relative error tolerance), изменить начальные условия и другие настройки, а при ошибке Singular matrix нужно уменьшить порог диагонального элемента (PIVTOL) и увеличить допустимый уровень заполнения (PIVREL), равные по умолчанию  $1 \times 10^{-13}$ .

В среду MS 10 встроены эмулятор тысяч моделей компонентов SPICE и эмулятор XSPICE, предназначенный для эффективного эмулирования цифровых компонентов, а также конструктор моделей (Model Makers), который автоматически сгенерирует модель на основании данных Databook.

Конструктор моделей запускается при редактировании компонентов из базы данных на закладке окна свойств компонента (Model → Component Properties → (Add → Edit) → Start Model Maker). Начальные настройки моделей не фиксированы, они соответствуют определенной модели. С помощью данных Databook можно выбрать другие численные значения.

В среде MS 10 имеются три базы данных (Tools → Database → Database Manager – Инструменты → База данных → Проводник баз данных):

- Master Database (Главная база данных), содержащая компоненты, из которой можно только считывать информацию;
- User Database (Пользовательская база данных), предназначенная для хранения компонентов на персональном компьютере;
- Corporate Database (Корпоративная база данных), компоненты которой доступны только пользователям корпоративной сети.

Все базы данных разделяются на группы, а они – в свою очередь, на семейства. Когда пользователь выбирает компонент и размещает его на рабочее поле, создается новая копия. Все изменения в ней не затрагивают информацию,

хранящуюся в базе данных. Параметры компонентов можно изменить и в базе данных, в любое время восстановить их в том виде, как они были записаны изначально, или добавить в нее свой разработанный компонент.

В команду Reports включены опции формирования таблиц с подробной информацией об элементах и приборах, включенных в исследуемую схему электронного устройства.

Опции команды Windows используются для быстрого размещения на рабочем поле открытых окон участков схем и спроектированных приборов.

Среда MS 10 содержит функции и режимы работы для целей обучения. Эти функции включают назначение дефектов компонентов схемы, установку глобальных ограничений, распространяемых на все проекты и локальных – для конкретной схемы и для вывода конкретного элемента. Назначить дефект определенному выводу установленного компонента можно на закладке Fault (Дефект) в диалоговом окне свойств этого элемента: None (Нет дефекта), Open (Обрыв), Short (Короткое замыкание), Leakade (Утечка).

## **П2.4. Командные строки инструментальной линейки**

Число выводимых на экран панелей библиотек с соответствующим набором пиктограмм компонентов в инструментальной линейке (рис. П2.3) зависит от их выбора пользователем в диалоговом окне (рис. П2.5), открываемом после щелчка мышью на кнопке View, наведения указателя мыши на опцию Toolbars и выбора панели Components.

Строки инструментальной линейки содержат управляющие кнопки, часть из которых необходима для запуска/выключения программы (кнопка  – Run Simulation) или ее остановки (кнопка  – Pause), а другая часть (кнопки библиотек инструментов и приборов, масштабирования и др.) используется при проектировании схем.

В MS 10 имеется несколько типов библиотек компонентов:

- Family Components – библиотеки с номинальными значениями параметров компонентов фирм-производителей;
- Virtual Components – библиотеки виртуальных компонентов, значения параметров которых устанавливаются пользователем;
- Rated Virtual Components – виртуальные компоненты с ограничениями, которые могут быть выведены из строя («сгорят»), если в процессе моделирования схемы на них подаются сигналы с параметрами, превышающими установленные (в окне свойств каждого компонента на закладке Value) пользователем допустимые значения;
- 3D Virtual Parts – трехмерные (3D) виртуальные компоненты.

После щелчка мышью на одной из 17 пиктограмм панели компонентов фирм-производителей (рис. П2.8 вверху) в открывшемся окне пользователь выбирает в меню Group нужную ему библиотеку компонентов, например библиотеку Basic, а в открывшемся окне (левый столбец внизу) – например, компонент Resistor (Резистор). Во втором столбце необходимо щелкнуть мышью на значении требуемого сопротивления (например, 1 k – 1 кОм), выбрать модель (тип) резистора, процент отклонения сопротивления от номинального значения и щелкнуть мышью на кнопке ОК.

Рассмотрим значения наименований в библиотеке компонентов.

1. Source – источники энергии.
2. Basic – базисные компоненты.
3. Diode – диоды.
4. Transistors – биполярные и полевые транзисторы.
5. Analog – аналоговые микросхемы.
6. TTL – логические микросхемы семейства ТТЛ.
7. CMOS – КМОП-микросхемы.
8. Misc digital – цифровые микросхемы.
9. Mixed – аналого-цифровые и цифроаналоговые преобразователи.
10. Indicator – индикаторные устройства.
11. Power – силовые устройства.

12. Miscellaneous – компоненты смешанного типа.
13. Advanced Peripherals – периферийные устройства.
14. RF – радиоэлементы.
15. Electromechanical – электромеханические устройства.
16. MCU Modul – функциональные узлы цифровой электроники.
17. Hierarchical Block – блок-схемы.

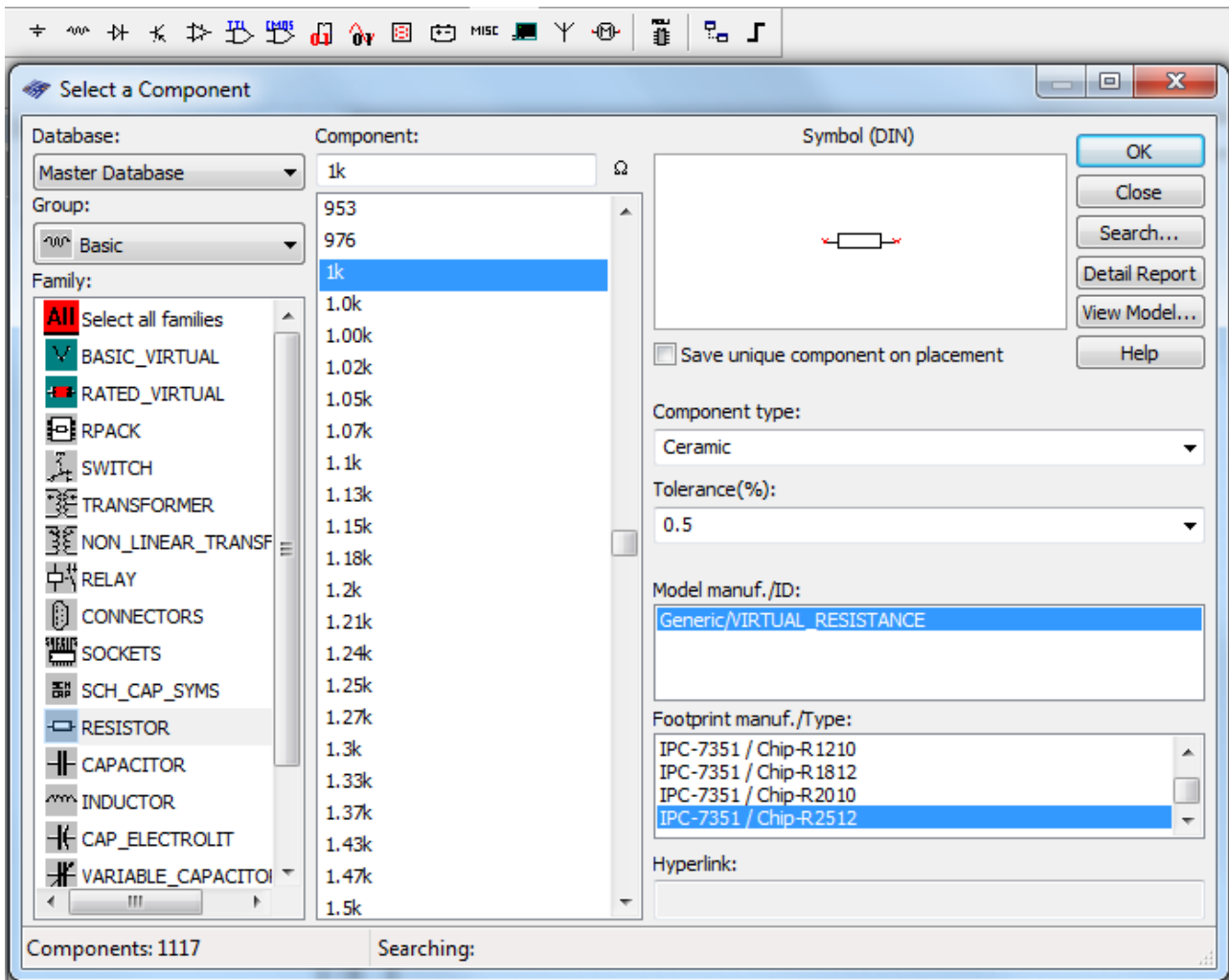


Рисунок П2.8. Окно библиотеки компонентов

При наведении курсора на графическое изображение компонента (рис. П2.8) оно как бы закрепляется за курсором мыши и после щелчка мышью в любой точке рабочего поля среды MS 10 переносится на экран. После двойного щелчка мышью на изображении компонента на рабочем поле MS 10 в открывшемся окне-закладке можно установить другие параметры компонента,

например температуру испытания, изменить (закладка Value) или оставить (заданное по умолчанию) значение основного параметра (например, 1 k резистора), заменить этот резистор на другой (опция Replace) и выбрать сопротивление из шкалы номиналов.

Использование этих библиотек целесообразно на завершающем этапе разработки и испытания схем электронных устройств с формированием таблиц с параметрами компонентов фирм-производителей.

На начальном этапе, при сборке и испытании схем базовых узлов и исследовании влияния параметров компонентов на получаемые характеристики устройств удобны библиотеки виртуальных компонентов (рис. П2.9, верхняя строка), выводимые в инструментальную линейку после щелчка мышью на кнопке меню View, выбора опции Toolbars и щелчка мышью на закладке Virtual.

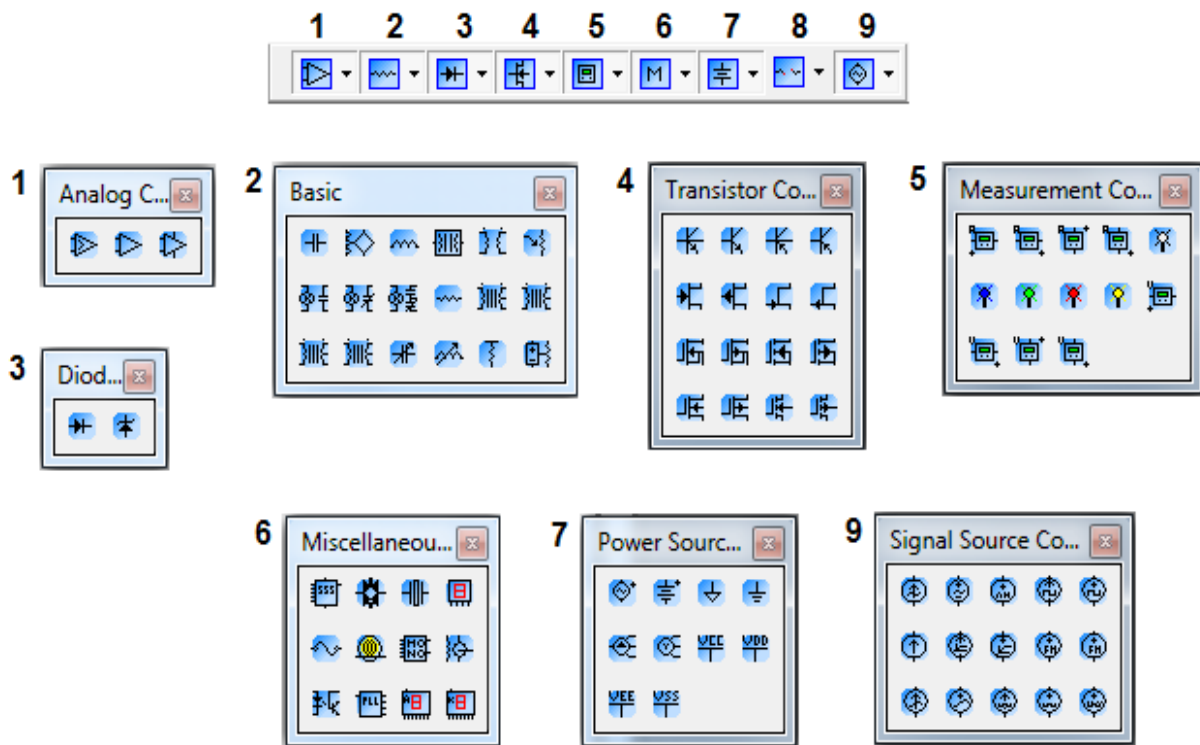


Рисунок П2.9. Окно библиотеки виртуальных компонентов

На панели Virtual размещены следующие виртуальные библиотеки и входящие в них компоненты и устройства (рис. П2.9):

1. Analog Components – аналоговые микросхемы: компаратор, операционные усилители с тремя и с пятью выводами;
2. Basic Components – базовые компоненты: конденсатор, индуктивные элементы, трансформаторы, потенциометр, реле, резистор, конденсатор с переменной емкостью, индуктивная катушка с переменной индуктивностью и др.;
3. Diod Components – диоды и стабилитроны;
4. Transistor Components – транзисторы биполярные и полевые;
5. Measurement Components – измерительные приборы: амперметры, вольтметры, световые индикаторы;
6. Miscellaneous Components – устройства смешанного типа: таймер 555, выключатели, предохранители, лампа накаливания, двигатель постоянного тока, семисегментные индикаторы;
7. Power Source Components – источники напряжения, трехфазные генераторы, источники питания цифровых устройств, заземление;
8. Rated Virtual Components – виртуальные компоненты с ограничениями параметров и с выбором скорости выхода из строя при превышении установленных параметров (Animation Delay Factor – множитель задержки для анимации; параметр не имеет отношения к процессам реального времени): транзисторы, конденсатор, диод, катушка индуктивности, двигатель постоянного тока, реле трех типов, резистор;
9. Signal Source Components – генераторы: синусоидального напряжения и тока, амплитудно-модулированных сигналов (тока и напряжения), тока и напряжения специальных форм (в виде меандра, прямоугольных, экспоненциальных и линейно нарастающих импульсов).

## **П2.5. Измерительные приборы, источники питания и устройства визуализации**

Справа от рабочего окна (поля) программной среды MS 10 после щелчков мышью на пиктограммах команд Simulate → Instruments выводится

вертикальная линейка кнопок-пиктограмм источников питания, измерительных приборов и осциллографов (рис. П2.10).

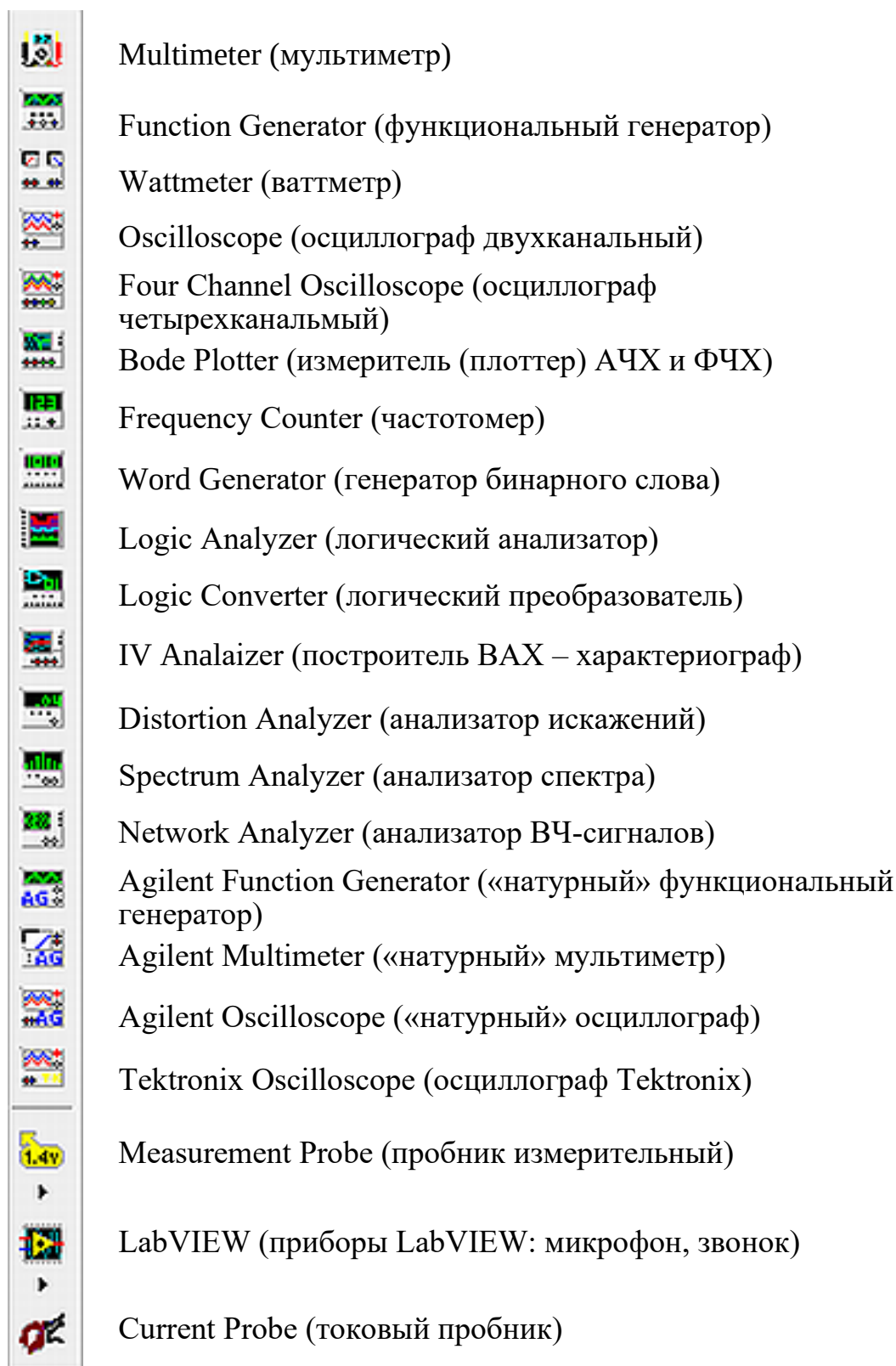


Рисунок П2.10. Линейка кнопок-пиктограмм



Для подключения прибора к схеме нужно щелкнуть мышью на пиктограмме прибора, затем на рабочем поле программы MS 10 и подключить выводы прибора к узлам схемы. Некоторые приборы нужно заземлить (подключить к общей точке), иначе их показания будут неверными. Расширенное изображение прибора выводится на рабочее поле после двойного щелчка мышью на его уменьшенном изображении или после выполнения команд Tools → Zoom. Закрыть увеличенное окно прибора можно щелчком мыши на кнопке Close, расположенной в правом верхнем углу окна.

## **П2.6. Сборка схем электронных устройств**

Процесс создания схем для испытания электронных узлов и устройств начинается с «переноса» и размещения на рабочем поле среды MS 10 нужных компонентов и приборов из ее библиотек и последующего их соединения «проводниками» между собой. Библиотека раскрывается после щелчка мышью на ее пиктограмме.

Чтобы «перетащить» виртуальный компонент или прибор на рабочее поле, нужно поместить указатель (курсор) мыши на изображение компонента и щелкнуть левой клавишей. Затем выбрать место на рабочем поле и щелкнуть мышью. Для перемещения компонента в другое место рабочего поля необходимо нажать левую клавишу мыши на его изображении и, не отпуская клавишу, передвинуть компонент в нужное место. Для более точного расположения элемента или прибора его нужно выделить, а затем переместить, нажимая на соответствующие клавиатурные клавиши (со стрелками) управления курсором.

Для соединения выводов (зажимов) элементов и/или приборов нужно подвести курсор мыши к концу требуемого вывода (при этом указатель (стрелка) мыши преобразуется в небольшой кружок-точку), щелкнуть левой клавишей (или нажать левую клавишу) и перемещать тянущуюся за курсором линию к требуемому выводу другого компонента до появления на его конце точки (рис. П2.11). После щелчка на точке мышью два вывода будут соединены

линией («проводником»). Аналогично можно соединить любой вывод элемента (прибора) с любым проводником, на котором появляется (при соединении) точка.

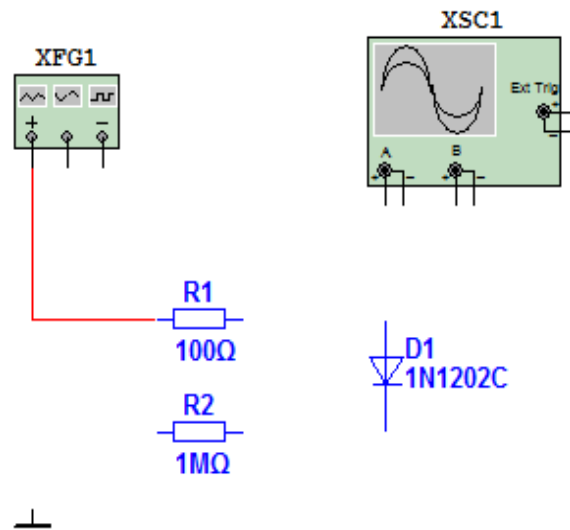


Рисунок П2.11. Соединение выводов элементов и приборов

Следует отметить, что прокладка соединительных проводников производится автоматически, причем препятствия – компоненты и другие проводники – огибаются по ортогональным направлениям (по горизонтали или по вертикали). Если необходимо переместить отдельный сегмент проводника, к нему подводится курсор мыши, нажимается левая клавиша и, после выделения проводника, производится нужное перемещение.

Для разрыва проводника нужно подвести курсор мыши к требуемому выводу компонента до появления крестика, щелкнуть мышью на крестике, а затем щелкнуть два раза на свободном пространстве поля. В этом месте на конце отсоединенного проводника появится точка, используя которую в дальнейшем можно соединить этот проводник с другим выводом любого компонента или с другим проводником. С другой стороны, если подвести курсор к концу вывода, щелкнуть мышью на появившемся крестике, подвести курсор к другому выводу и щелкнуть на нем, то получим новое присоединение этого вывода.

Чтобы удалить проводник, нужно щелкнуть на нем правой клавишей мыши (на концах соединений этого проводника появятся квадратики) и в открывшемся окне щелкнуть левой клавишей мыши на опции Delete (рис. П2.12) или нажать на клавиатурную клавишу Del. В этом же окне размещены опции выбора цвета проводника (Segment Color) и выбора шрифта (Font) и его номера. Если выбрать синий и красный цвета проводов, подходящих к входам осциллографа, логического анализатора и др., то графики кривых на экранах указанных приборов окрашиваются в заданные цвета.

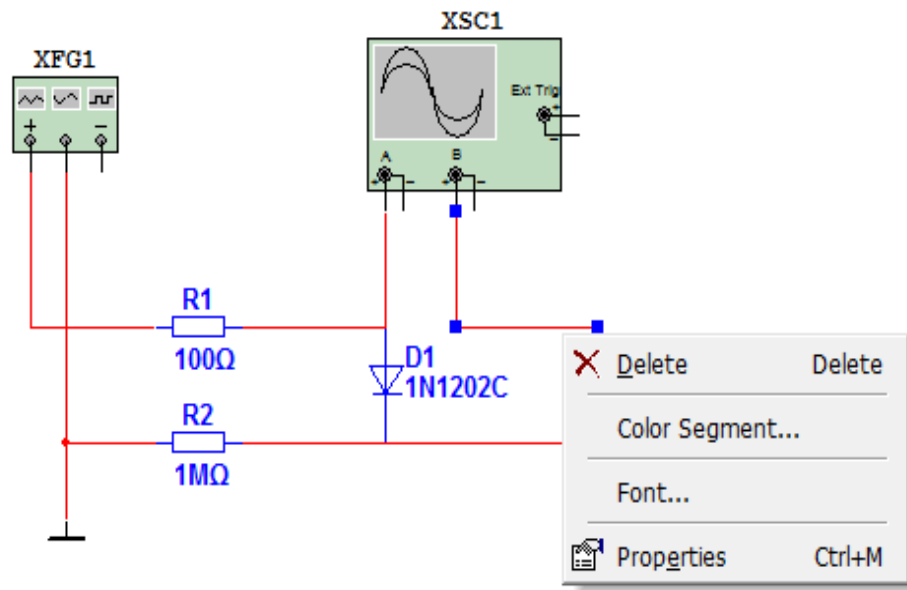


Рисунок П2.12. Разрыв соединения выводов элементов и приборов

Выделенную мышью группу компонентов с нажатой левой клавишей можно перемещать и удалять аналогично перемещению и удалению одного элемента. Для выделения всей схемы (кадра) нужно щелкнуть мышью на опции Select All в меню Edit. Для поворота компонента, например на 90°, его нужно выделить и нажать (одновременно) клавиатурные клавиши Ctrl+R.

## П2.7. Установка параметров компонентов

Для установки требуемых параметров элемента или прибора нужно навести на его изображение курсор мыши и щелкнуть два раза левой клавишей.

В появившемся окне (рис. П2.13) можно изменить (с помощью закладки Value) значения параметров прибора (в данном примере диода VD1), его буквенное обозначение (закладка Label), после чего нужно подтвердить (нажать кнопку ОК) или отменить (Cancel) выбранные значения (установку) параметров.

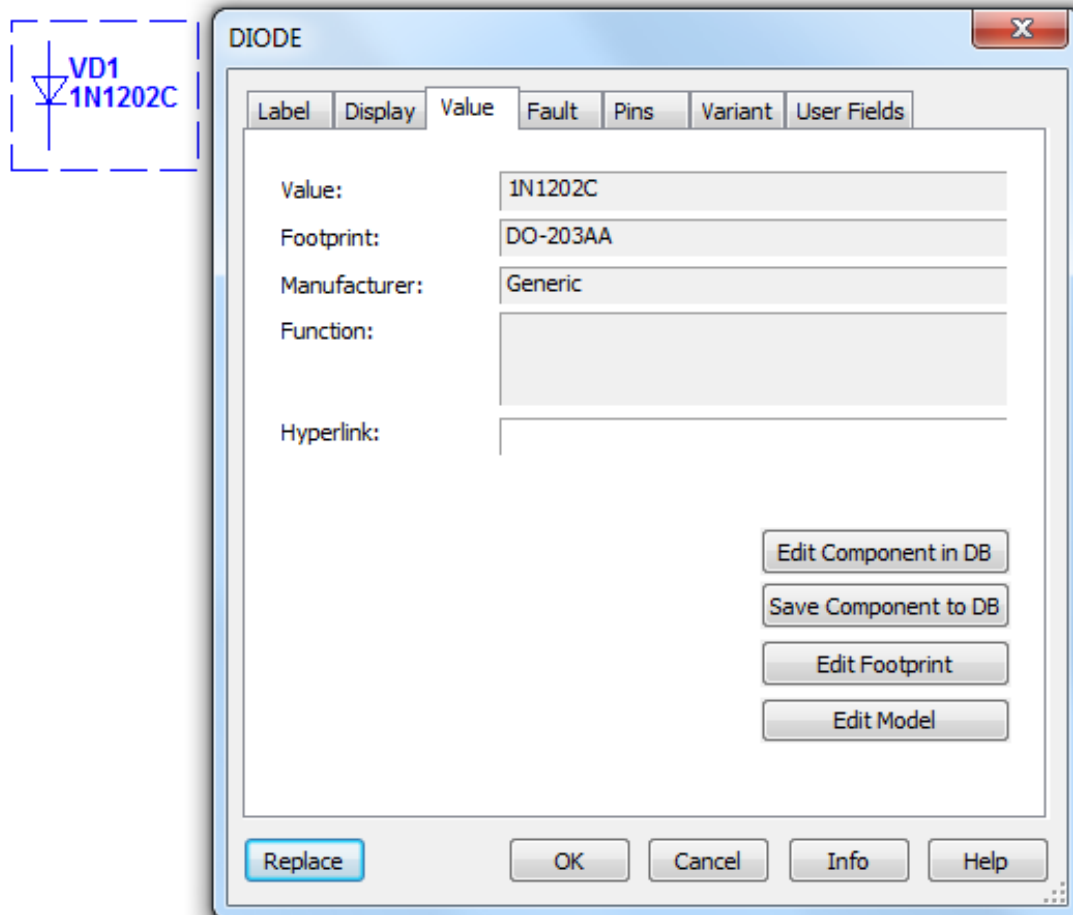


Рисунок П2.13. Окно установки параметров прибора

Чтобы изменить цвет компонента, нужно навести курсор мыши, и щелкнуть на изображении компонента правой клавишей. В появившемся окне (рис. П2.14), щелкая мышью на соответствующих закладках, в открывающихся окнах можно выбрать цвет (Change Color) компонента, вырезать (Cut) или удалить (Delete) его, выполнить другие операции, относящиеся к этому компоненту.

Предусмотрена возможность изменения параметров интерактивных компонентов, управляемых буквенными или цифровыми клавиатурными

клавишами (английский шрифт). Так, для изменения сопротивления потенциометра R1 нужно кратковременно нажимать на назначенную в закладке Key клавишу C (рис. П2.15). При этом дискретно уменьшается (в данном случае на 5 %) значение сопротивления предварительно выделенного (см. пунктирную рамку на рис. П2.15) потенциометра. При одновременном нажатии клавиш Shift+C сопротивление потенциометра R1 увеличивается.

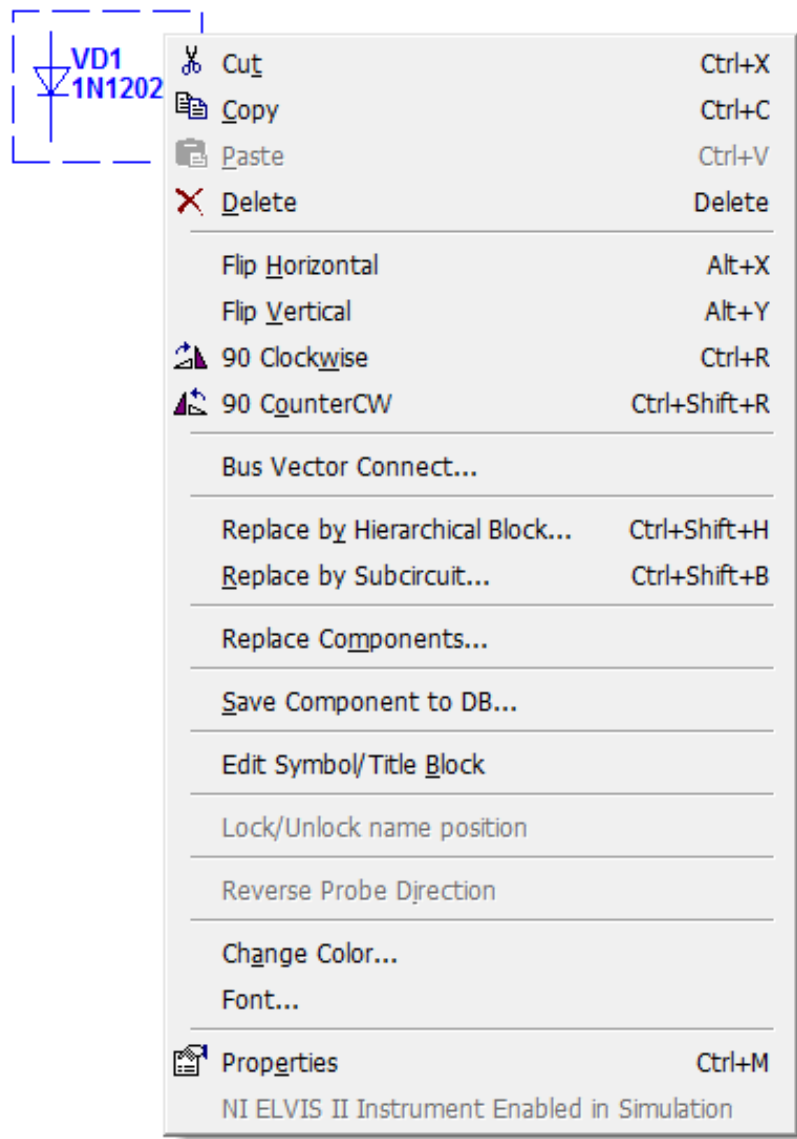


Рисунок П2.14. Окно выполнения операций

Изменить параметры приведенных на рис. П2.15 компонентов можно также с помощью движков, которые всплывают при подведении указателя мыши к надписи Key: при перемещении мышью движка влево/вправо

уменьшается/увеличивается значение параметра (емкости  $C_1$ , индуктивности  $L_1$ , сопротивления  $R_1$ ) соответствующего компонента.

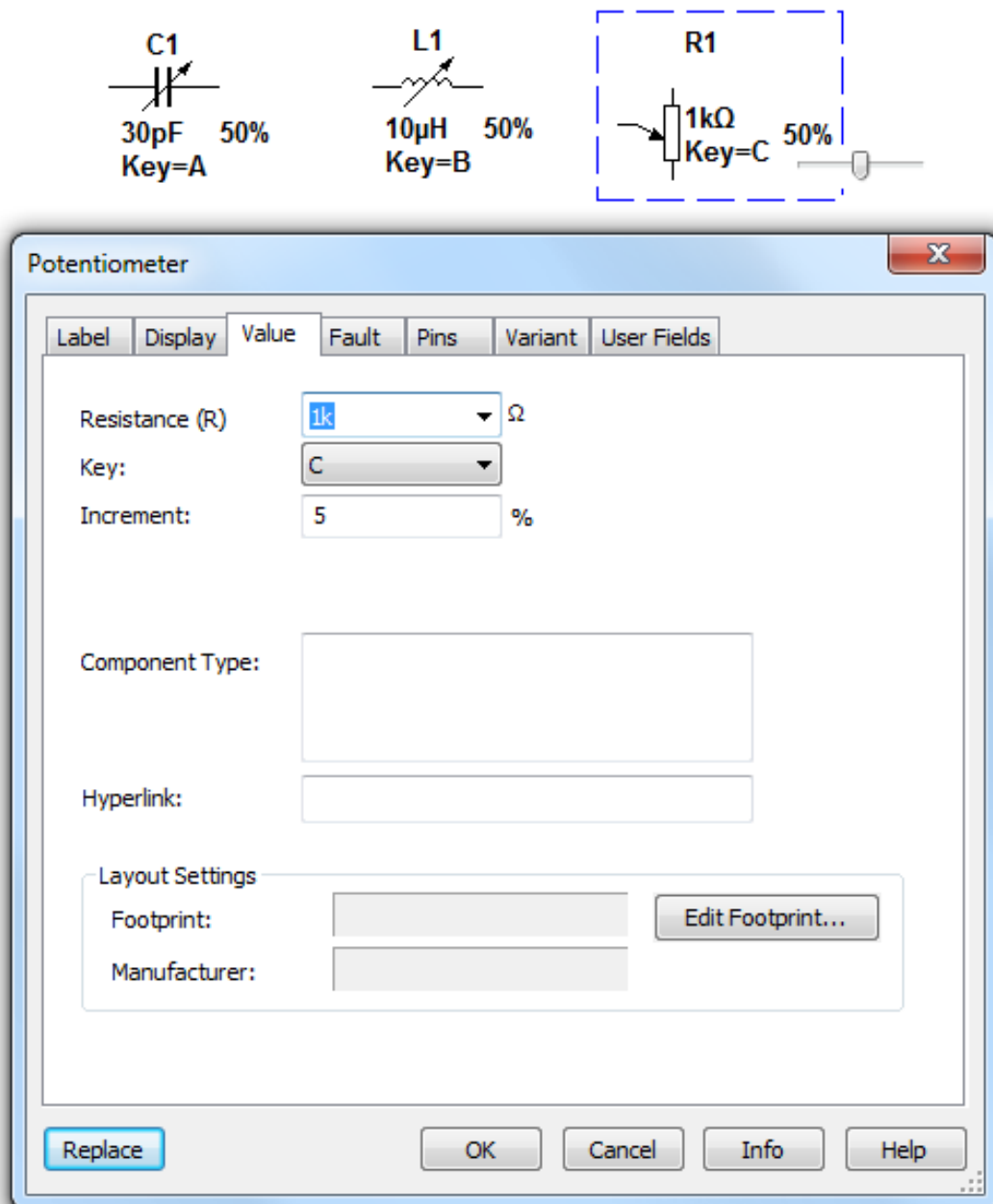


Рисунок П2.15. Окно изменения параметров интерактивных компонентов

Таким образом, как в реальном эксперименте, параметры компонентов схемы можно изменять, не прерывая процесса моделирования и наблюдения за изменениями параметров (по показаниям приборов) и кривых зависимостей электрических величин во времени на экране осциллографа.

Для функционирования буквенных и цифровых клавиатурных клавиш (А, В, С, Z; 1, 2, 9) нужно щелкнуть мышью на свободном участке рабочего поля среды MS10 для закрытия активных окон приборов и осциллографа.

Для управления измерительными приборами, осциллографом и снятия показаний надо навести курсор мыши на изображение соответствующего прибора и щелкнуть два раза левой клавишей мыши. Появится увеличенное изображение панели прибора с закладками установки параметров и режимов его работы. С помощью визирных линий (визиров), расположенных слева и справа в окне осциллографа или плоттера, можно получить координаты точек на снимаемых временных, частотных или вольтамперных характеристиках электронных устройств.

### **ПРИЛОЖЕНИЕ 3. СОДЕРЖАНИЕ, ФОРМА И ПРАВИЛА ОФОРМЛЕНИЯ ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ**

#### **Содержание**

1. Титульный лист с указанием названия лабораторной работы, ФИО студента.
2. Цель работы.
3. Основная часть, в которой должны присутствовать комментарии о производимых действиях при выполнении задания, предлагаемого в лабораторной работе.
4. Выводы по результатам выполнения задания, предлагаемого в лабораторной работе.

Текст отчета подготавливается в текстовом редакторе, согласно ГОСТ 2.105-95 «Общие требования к текстовым документам», и предоставляется в электронном и в распечатанном виде в одном экземпляре. К отчету прилагаются файлы, созданные в ходе выполнения задания, предлагаемого в лабораторной работе.



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

по организации и проведению самостоятельной работы  
по дисциплине «Цифровые системы передачи»  
для студентов направления подготовки  
09.03.01 Информатика и вычислительная техника  
Направленность (профиль)

"Программное обеспечение средств вычислительной техники  
и автоматизированных систем"

## ОГЛАВЛЕНИЕ

|                                                                                                                                |           |
|--------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>ВВЕДЕНИЕ.....</b>                                                                                                           | <b>4</b>  |
| <b>1 ОБЩАЯ ХАРАКТЕРИСТИКА САМОСТОЯТЕЛЬНОЙ<br/>РАБОТЫ СТУДЕНТА ПРИ ИЗУЧЕНИИ ДИСЦИПЛИНЫ<br/>«ЦИФРОВЫЕ СИСТЕМЫ ПЕРЕДАЧИ».....</b> | <b>7</b>  |
| <b>2 ПЛАН-ГРАФИК ВЫПОЛНЕНИЯ САМОСТОЯТЕЛЬНОЙ<br/>РАБОТЫ .....</b>                                                               | <b>12</b> |
| <b>3 МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ИЗУЧЕНИЮ<br/>ТЕОРЕТИЧЕСКОГО МАТЕРИАЛА .....</b>                                              | <b>15</b> |
| <b>4 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПОДГОТОВКЕ К<br/>ЛАБОРАТОРНЫМ РАБОТАМ .....</b>                                                  | <b>29</b> |
| <b>СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ.....</b>                                                                                    | <b>38</b> |

## ВВЕДЕНИЕ

Основная задача высшего образования заключается в формировании творческой личности специалиста, способного к саморазвитию, самообразованию, инновационной деятельности. Решение этой задачи вряд ли возможно только путем передачи знаний в готовом виде от преподавателя к студенту. Для решения этой задачи в учебные планы всех специальностей включена самостоятельная работа, составляющая значительную часть учебного времени обучаемого.

В связи с этим, обучаемому из пассивного потребителя знаний необходимо превратиться в активного их творца, умеющего сформулировать проблему, проанализировать пути ее решения, найти оптимальный результат и доказать его правильность. Происходящая в настоящее время реформа высшего образования связана по своей сути с переходом парадигмы обучения к парадигме образования. В этом плане следует признать, что самостоятельная работа студентов является не просто важной формой образовательного процесса, а должна стать его основой. Это предполагает ориентацию на активные методы овладения знаниями, развитие творческих способностей студентов, переход от поточного к индивидуализированному обучению с учетом потребностей и возможностей личности.

В широком смысле под самостоятельной работой следует понимать совокупность всей самостоятельной деятельности студентов как в учебной аудитории, так и вне ее, в контакте с преподавателем и в его отсутствие.

Самостоятельная работа *в контакте с преподавателем* реализуется:

- непосредственно в процессе аудиторных занятий – на лекциях, практических и семинарских занятиях, при выполнении лабораторных работ;
- вне рамок расписания – на консультациях по учебным вопросам, в ходе творческих контактов, при ликвидации задолженностей, при выполнении студентом учебных и творческих задач.

Видами заданий для *внеаудиторной* самостоятельной работы могут быть:

**1. Для овладения знаниями:**

– чтение текста (учебника, первоисточника, дополнительной литературы);

– составление плана текста;

– графическое изображение структуры текста;

– конспектирование текста;

– работа со словарями и справочниками;

– работа с нормативными документами;

– учебно-исследовательская работа;

– использование аудио- и видеозаписей, компьютерной техники,

Интернет и др.

**2. Для закрепления и систематизации знаний:**

– работа с конспектом лекции (обработка текста);

– повторная работа над учебным материалом (учебника, первоисточника, дополнительной литературы, аудио- и видеозаписей);

– составление плана и тезисов ответа;

– составление таблиц для систематизации учебного материала;

– изучение нормативных материалов;

– ответы на контрольные вопросы;

– аналитическая обработка текста (аннотирование, рецензирование, реферирование, конспект, анализ и др.);

– подготовка сообщений к выступлению на семинаре, конференции;

– подготовка рефератов, докладов;

– составление библиографии;

– тестирование и др.

**3. Для формирования умений:**

– решение задач и упражнений по образцу;

- решение вариантных задач и упражнений;
- решение ситуационных производственных (профессиональных) задач;
- подготовка к деловым играм;
- проектирование и моделирование разных видов и компонентов профессиональной деятельности;
- подготовка курсовых и дипломных работ (проектов);
- экспериментальная работа;
- рефлексивный анализ профессиональных умений, с использованием аудио- и видеотехники и др.

Границы между этими видами работ достаточно размыты, а сами виды самостоятельной работы пересекаются.

## **1 ОБЩАЯ ХАРАКТЕРИСТИКА САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТА ПРИ ИЗУЧЕНИИ ДИСЦИПЛИНЫ «ЦИФРОВЫЕ СИСТЕМЫ ПЕРЕДАЧИ»**

Целью самостоятельной работы студента при изучении дисциплины «Цифровые системы передачи» является формирование у обучающихся личностных качеств, а также набора профессиональных компетенций будущего бакалавра в области применения систем связи, функционирующих на основе цифровых технологий и в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.01 – Информатика и вычислительная техника.

Задачи самостоятельной работы:

- изучение основных способов построения и принципов функционирования волоконно-оптических систем связи;
- изучение основных способов построения и принципов функционирования цифровых систем передачи;
- формирование у студентов умений, позволяющих самостоятельно проводить теоретический анализ процессов в современных цифровых системах передачи;
- приобретение навыков принятия и обоснования конкретных технических решений при конструировании элементов цифровых систем передачи.

В рабочей программе дисциплины «Цифровые системы передачи» определены следующие виды самостоятельной работы:

- подготовка к лабораторной работе;
- самостоятельное изучение литературы.

Целью вида самостоятельной работы «Подготовка к лабораторной работе» по дисциплине «Цифровые системы передачи» является закрепление студентами теоретического материала по специальности и выработка навыков самостоятельной профессиональной и научно-

исследовательской деятельности в области программного обеспечения робототехнических и интеллектуальных систем.

Задачи вида самостоятельной работы «Подготовка к лабораторной работе» по дисциплине «Цифровые системы передачи» обусловлены необходимостью получения студентом знаний, умений, навыков согласно требованиям ФГОС ВО по направлению подготовки 09.03.01 – Информатика и вычислительная техника, на основе которых формируются соответствующие результаты обучения по дисциплине (модулю), соотнесённые с планируемыми результатами освоения образовательной программы.

| Код, формулировка компетенции                                                                                                     | Код, формулировка индикатора                                                                                                                                                                                                                                                                                                                                                                                                       | Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов                                                                                                                                     |
|-----------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ПК-7. Способен осуществлять обслуживание сетевых устройств и серверных операционных систем информационно-коммуникационной системы | ИД-1 <sub>ПК-7</sub> понимает общие принципы функционирования аппаратных, программных и программно-аппаратных средств администрируемой сети; ориентируется в архитектурах аппаратных, программных и программно-аппаратных средств администрируемой сети; понимает принципы установки и настройки программного обеспечения.                                                                                                         | На основе понимания общих принципов функционирования аппаратных, программных и программно-аппаратных средств цифровой системы передачи, может определить архитектуру системы и провести установку и настройку программного обеспечения на её элементах. |
|                                                                                                                                   | ИД-2 <sub>ПК-7</sub> организывает инвентаризацию периферийных и абонентских технических средств; идентифицирует права пользователей по доступу к программно-аппаратным средствам информационных служб инфокоммуникационной системы и/или ее составляющих; применяет специальные программно-аппаратные средства контроля доступа пользователей к программно-аппаратным средствам информационных служб инфокоммуникационной системы. | Способен организовать инвентаризацию абонентских технических средств в цифровых системах передачи.                                                                                                                                                      |
|                                                                                                                                   | ИД-3 <sub>ПК-7</sub> осуществляет конфигурирование УАТС, периферийных и абонентских устройств.                                                                                                                                                                                                                                                                                                                                     | Способен осуществить конфигурирование абонентских устройств в цифровых системах передачи.                                                                                                                                                               |

|                                                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ПК-8. Способен осуществлять администрирование процесса контроля производительности сетевых устройств и программного обеспечения, проводить регламентные работы на сетевых устройствах и программном обеспечении инфокоммуникационной системы | ИД-1 <sub>ПК-8</sub> понимает устройство и принцип работы кабельных и сетевых анализаторов; понимает модель OSI/ISO, модели IEEE; понимает назначение протоколов канального, сетевого, транспортного и прикладного уровней модели взаимодействия открытых систем.                                                                                                                                                                                | На основе понимания устройства и принципа работы кабельных и сетевых анализаторов, модели OSI/ISO, модели IEEE, назначения протоколов канального, сетевого, транспортного и прикладного уровней модели взаимодействия открытых систем, способен осуществлять администрирование процесса контроля производительности сетевых устройств в цифровых системах передачи. |
|                                                                                                                                                                                                                                              | ИД-2 <sub>ПК-8</sub> использует нормативно-техническую документацию в области инфокоммуникационных технологий; использует современные методы контроля производительности инфокоммуникационных систем; осуществляет мониторинг администрируемых сетевых устройств; планирует восстановление сетевой инфокоммуникационной системы; восстанавливает параметры программного обеспечения сетевых устройств; планирует модернизацию сетевых устройств. | На основе использования нормативно-технической документации в области инфокоммуникационных технологий, а также современных методов контроля производительности инфокоммуникационных систем, способен осуществлять мониторинг администрируемых сетевых устройств в цифровых системах передачи.                                                                       |
|                                                                                                                                                                                                                                              | ИД-3 <sub>ПК-8</sub> использует современные измерительные приборы и программное обеспечение; устанавливает операционные системы сетевых устройств; конфигурирует операционные системы сетевых устройств администрируемой сети; осуществляет разборку и сборку администрируемых сетевых устройств.                                                                                                                                                | Способен проводить регламентные работы на сетевых устройствах и программном обеспечении в цифровых системах передачи с использованием современных измерительных приборов и программного обеспечения.                                                                                                                                                                |

Целью вида самостоятельной работы «Самостоятельное изучение литературы» по дисциплине «Цифровые системы передачи» является расширение и закрепление знаний, приобретаемых студентом на традиционных формах занятий и выработка навыков самостоятельной работы с литературой и источниками в области программного обеспечения робототехнических и интеллектуальных систем.



Задачи вида самостоятельной работы «Самостоятельное изучение литературы» по дисциплине «Цифровые системы передачи» обусловлены необходимостью получения студентом знаний, умений, навыков согласно требованиям ФГОС ВО по направлению подготовки 09.03.01 – Информатика и вычислительная техника, на основе которых формируются соответствующие результаты обучения по дисциплине (модулю), соотнесённые с планируемыми результатами освоения образовательной программы.

| Код, формулировка компетенции                                                                                                     | Код, формулировка индикатора                                                                                                                                                                                                                                                                                                                                                                                                       | Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов                                                                                                                                     |
|-----------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ПК-7. Способен осуществлять обслуживание сетевых устройств и серверных операционных систем информационно-коммуникационной системы | ИД-1 <sub>ПК-7</sub> понимает общие принципы функционирования аппаратных, программных и программно-аппаратных средств администрируемой сети; ориентируется в архитектурах аппаратных, программных и программно-аппаратных средств администрируемой сети; понимает принципы установки и настройки программного обеспечения.                                                                                                         | На основе понимания общих принципов функционирования аппаратных, программных и программно-аппаратных средств цифровой системы передачи, может определить архитектуру системы и провести установку и настройку программного обеспечения на её элементах. |
|                                                                                                                                   | ИД-2 <sub>ПК-7</sub> организывает инвентаризацию периферийных и абонентских технических средств; идентифицирует права пользователей по доступу к программно-аппаратным средствам информационных служб инфокоммуникационной системы и/или ее составляющих; применяет специальные программно-аппаратные средства контроля доступа пользователей к программно-аппаратным средствам информационных служб инфокоммуникационной системы. | Способен организовать инвентаризацию абонентских технических средств в цифровых системах передачи.                                                                                                                                                      |
|                                                                                                                                   | ИД-3 <sub>ПК-7</sub> осуществляет конфигурирование УАТС, периферийных и абонентских устройств.                                                                                                                                                                                                                                                                                                                                     | Способен осуществить конфигурирование абонентских устройств в цифровых системах передачи.                                                                                                                                                               |

|                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                                                                            |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>ПК-8. Способен осуществлять администрирование процесса контроля производительности сетевых устройств и программного обеспечения, проводить регламентные работы на сетевых устройствах и программном обеспечении инфокоммуникационной системы</p> | <p>ИД-1<sub>ПК-8</sub> понимает устройство и принцип работы кабельных и сетевых анализаторов; понимает модель OSI/ISO, модели IEEE; понимает назначение протоколов канального, сетевого, транспортного и прикладного уровней модели взаимодействия открытых систем.</p>                                                                                                                                                                                | <p>На основе понимания устройства и принципа работы кабельных и сетевых анализаторов, модели OSI/ISO, модели IEEE, назначения протоколов канального, сетевого, транспортного и прикладного уровней модели взаимодействия открытых систем, способен осуществлять администрирование процесса контроля производительности сетевых устройств в цифровых системах передачи.</p> |
|                                                                                                                                                                                                                                                     | <p>ИД-2<sub>ПК-8</sub> использует нормативно-техническую документацию в области инфокоммуникационных технологий; использует современные методы контроля производительности инфокоммуникационных систем; осуществляет мониторинг администрируемых сетевых устройств; планирует восстановление сетевой инфокоммуникационной системы; восстанавливает параметры программного обеспечения сетевых устройств; планирует модернизацию сетевых устройств.</p> | <p>На основе использования нормативно-технической документации в области инфокоммуникационных технологий, а также современных методов контроля производительности инфокоммуникационных систем, способен осуществлять мониторинг администрируемых сетевых устройств в цифровых системах передачи.</p>                                                                       |
|                                                                                                                                                                                                                                                     | <p>ИД-3<sub>ПК-8</sub> использует современные измерительные приборы и программное обеспечение; устанавливает операционные системы сетевых устройств; конфигурирует операционные системы сетевых устройств администрируемой сети; осуществляет разборку и сборку администрируемых сетевых устройств.</p>                                                                                                                                                | <p>Способен проводить регламентные работы на сетевых устройствах и программном обеспечении в цифровых системах передачи с использованием современных измерительных приборов и программного обеспечения.</p>                                                                                                                                                                |

## 2 ПЛАН-ГРАФИК ВЫПОЛНЕНИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

На первом этапе необходимо ознакомиться с рабочей программой дисциплины, в которой рассмотрено содержание тем дисциплины лекционного курса, взаимосвязь тем лекций с лабораторными занятиями, темы и виды самостоятельной работы. По каждому виду самостоятельной работы предусмотрены определённые формы отчетности.

Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием количества часов и видов занятий

| № | Раздел (тема) дисциплины и краткое содержание                                                                                                                                                                                                                                                                                                                                                                                         | Формируемые компетенции, индикаторы | очная форма                                                                                   |                      |                     |                               | Формы текущего контроля успеваемости      |
|---|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------|-----------------------------------------------------------------------------------------------|----------------------|---------------------|-------------------------------|-------------------------------------------|
|   |                                                                                                                                                                                                                                                                                                                                                                                                                                       |                                     | Контактная работа обучающихся с преподавателем /из них в форме практической подготовки, часов |                      |                     | Самостоятельная работа, часов |                                           |
|   |                                                                                                                                                                                                                                                                                                                                                                                                                                       |                                     | Лекции                                                                                        | Практические занятия | Лабораторные работы |                               |                                           |
| 1 | <b>Тема 1. Построение цифровых систем передачи на основе импульсно-кодовой модуляции с временным разделением каналов.</b><br>Основные понятия и определения. Классификация цифровых систем передачи. Аналого-цифровое и цифро-аналоговое преобразование сигнала в ЦСП ИКМ-ВРК. Квантование. Кодирование квантовых сигналов. Групповой ИКМ сигнал. Обобщенная структурная схема оконечной станции цифровой системы передачи с ИКМ-ВРК. | ПК-7, ПК-8                          | 6.00                                                                                          | -                    | 4.00/4.00           | 10                            | собеседование, защита лабораторной работы |

|   |                                                                                                                                                                                                                                                                                                                                                                                                                                                       |            |      |   |               |    |                                           |
|---|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|------|---|---------------|----|-------------------------------------------|
| 2 | <p><b>Тема 2. Основные функциональные узлы цифровых систем передачи с ИКМ-ВРК.</b></p> <p>Канальные амплитудно-импульсные модуляторы и селекторы. Кодеры и декодеры с линейной шкалой квантования. Кодеры и декодеры с нелинейной шкалой квантования. Генераторное оборудование цифровых систем передачи.</p>                                                                                                                                         | ПК-7, ПК-8 | 4.00 | - | 6.00/<br>6.00 | 10 | собеседование, защита лабораторной работы |
| 3 | <p><b>Тема 3. Временное группообразование или мультиплексирование в ЦСП ИКМ-ВРК.</b></p> <p>Иерархии и стандарты ЦСП ИКМ-ВРК. Объединение цифровых потоков в плезиохронной цифровой иерархии. Асинхронное объединение цифровых потоков. Синхронное объединение цифровых потоков. Функциональные узлы оборудования временного группообразования. Объединение цифровых потоков в синхронной цифровой иерархии. Функциональные блоки аппаратуры SDH.</p> | ПК-7, ПК-8 | 6.00 | - | 4.00/<br>4.00 | 10 | собеседование                             |
| 4 | <p><b>Тема 4. Цифровые разностные системы передачи.</b></p> <p>Дифференциальная импульсно-кодовая модуляция. Дельта-модуляция. Дельта-модуляция с инерционным компандированием. Полосное кодирование. Вокодеры. Ввод сигналов передачи данных в каналы и тракты цифровых систем передачи с ИКМ-ВРК.</p>                                                                                                                                               | ПК-7, ПК-8 | 4.00 | - | 6.00/<br>6.00 | 10 | собеседование                             |

|   |                                                                                                                                                                                                                                                                                                                                                                                                                                         |            |             |   |                 |       |               |
|---|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|-------------|---|-----------------|-------|---------------|
| 5 | <b>Тема 5. Синхронизация в цифровых системах передачи.</b><br>Основные понятия и определения. Виды синхронизации. Требования к системам синхронизации. Тактовая синхронизация. Тактовая сетевая синхронизация. Общие принципы построения цикловой синхронизации. Приемники синхросигнала с задержкой контроля и одноразрядным сдвигом. Приемники синхросигнала со скользящим поиском. Оценка параметров системы цикловой синхронизации. | ПК-7, ПК-8 | 6.00        | - | 4.00/<br>4.00   | 10    | собеседование |
|   | <b>Тема 6. Линейный тракт цифровых систем передачи по электрическим кабелям.</b><br>Основные понятия и определения. Структура линейного тракта и его основные параметры. Линейные коды. Скремблирование цифрового сигнала. Регенерация цифрового сигнала.                                                                                                                                                                               | ПК-7, ПК-8 | 4.00        | - | 6.00/<br>6.00   | 10    | собеседование |
|   | <b>Тема 7. Линейный тракт цифровых систем передачи по оптическим кабелям.</b><br>Обобщенная структурная схема цифровой волоконно-оптической системы передачи. Принципы построения двухсторонних линейных трактов ЦВОСП. Методы уплотнения волоконно-оптических линий связи. Источники оптического излучения. Приемники оптического излучения. Модуляция и демодуляция оптической несущей.                                               | ПК-7, ПК-8 | 6.00        | - | 6.00/<br>6.00   | 12    | собеседование |
|   | ИТОГО                                                                                                                                                                                                                                                                                                                                                                                                                                   |            | 36.00/<br>0 | - | 36.00/<br>36.00 | 72.00 |               |

### **3 МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ИЗУЧЕНИЮ ТЕОРЕТИЧЕСКОГО МАТЕРИАЛА**

При работе с рекомендованной литературой и источниками студенту необходимо, в первую очередь, обратить внимание на следующие моменты:

1. Внимательно изучить материалы, характеризующие курс и тематику самостоятельного изучения, что изложено в учебно-методическом комплексе по дисциплине. Это позволит четко представить как круг, изучаемых тем, так и глубину их постижения.

2. Составить подборку литературы, достаточную для изучения предлагаемых тем. В учебно-методическом комплексе или методических рекомендациях обязательно представлены основной и дополнительный списки литературы, доступные для студентов. Они носят рекомендательный характер, это означает, что всегда есть литература, которая может не входить в данный список, но является необходимой для освоения темы. При этом следует иметь в виду, что нужна литература различных видов:

- учебники, учебные и учебно-методические пособия;
- монографии, сборники научных статей, публикации в журналах;
- справочная литература – энциклопедии, словари, тематические, терминологические справочники;

3. Обучаемый должен совершать собственные интеллектуальные усилия, а не только механически заучивать понятия и положения.

**Вопросы для собеседования по вопросам темы выносимой на самостоятельное изучение**

**Тема 1. Построение цифровых систем передачи на основе импульсно-кодовой модуляции с временным разделением каналов**

1. Прокомментируйте необходимость процесса дискретизации при построении систем передачи с временным разделением каналов (СП с ВРК).

2. Для периодической последовательности прямоугольных импульсов (ПППИ) длительностью  $\tau_{\text{и}} = 244 \times 10^{-9} \text{ с}$ , периодом следования  $T_{\text{д}} = 125 \times 10^{-6} \text{ с}$  и амплитудой  $A_{\text{м}} = 4 \text{ В}$  определить частоту следования (или тактовую частоту)  $F_{\text{д}}$  и амплитудный спектр ПППИ для  $n = 5$ , найти полосу частот  $\Delta F$ , необходимую для передачи заданной ПППИ; изобразить спектр амплитуд ПППИ.

3. Частота следования ПППИ равна  $F_{\text{д}} = 8000 \text{ Гц}$ , при скважности  $q = 40$ . Определить ширину полосы частот  $\Delta F$ , необходимую для передачи такой ПППИ по тракту с частотной характеристикой идеального фильтра нижних частот, а также отношение амплитуды 1-ой гармоники к амплитуде 10-ой гармоники.

4. Назовите основные условия теоремы Найквиста-Котельникова.

5. Определить период дискретизации речевого и телевизионного сигналов, занимающих полосы частот  $0,2 \dots 6,8 \text{ кГц}$  и  $0,0000050 \dots 6,5 \text{ МГц}$ .

6. Назовите принципы реализации амплитудно-импульсной модуляции, спектральный состав АИМ-1 и АИМ-2, их отличия и сходства.

7. Определить минимальное значение частоты дискретизации  $F_{\text{д}}$  сигнала, для которого  $F_{\text{н}} = 60 \text{ кГц}$  и  $F_{\text{в}} = 108 \text{ кГц}$ .

8. Дискретизации подвергается сигнал с полосой частот  $60,6 \dots 107,7 \text{ кГц}$ . Ширина полосы расфильтровки фильтров, используемых для восстановления (демодуляции) дискретизированного АИМ сигнала, равна  $\Delta F_{\text{р}} = 1,8 \text{ кГц}$ . Найти наименьшее значение частоты дискретизации  $F_{\text{д}}$ , при котором не будет наблюдаться искажений от дискретизации.

9. Дискретизации во времени (или амплитудно-импульсной модуляции) подвергается сигнал, занимающий полосу частот  $60 \dots 84 \text{ кГц}$ . При каких из перечисленных ниже значений частоты дискретизации искажений наблюдаться не будет: 48, 96, 144 и 192 кГц? Параметры фильтров, используемых для демодуляции (восстановления) АИМ сигнала, принять идеальными.

10. Дискретизации подвергается сигнал с полосой частот  $0,3...3,4$  кГц. Определить минимальное значение частоты дискретизации, при котором возможно восстановление исходного сигнала из периодической последовательности отсчетов фильтром нижних частот, если крутизна его характеристики затухания в переходной области равна  $S_{\phi} = 0,05$  дБ/Гц, а затухание в полосе эффективного задерживания равно  $A_{\text{пэз}} = 60$  дБ.

11. Объясните необходимость квантования отсчетов. Перечислите виды квантования.

12. Синусоидальный сигнал с амплитудой  $U_m = 1$  В следует преобразовать в цифровую форму таким образом, чтобы получить защищенность от шумов квантования  $A_{\text{кв}}$  на уровне не менее 30 дБ. Определить величину шага квантования, число шагов квантования и разрядность кода при равномерном кодировании.

13. Равномерному квантованию подвергается телевизионный сигнал от одного источника, для которого коэффициент  $k = \sqrt{3}$ . Определить разрядность кода, если требуется обеспечить защищенность от шумов квантования не ниже 30 дБ.

14. Равномерному квантованию и последующему кодированию 6-разрядным кодом подвергается групповой телефонный сигнал от одного источника, для которого  $k = 4$ . Определить полученную при этом защищенность сигнала от шумов квантования.

15. Назовите типы кодов, применяемых для кодирования отсчетов АИМ-2 сигналов.

16. На вход кодера с линейной шкалой квантования с шагом, равным  $\delta = 10$  мВ, поступают импульсы АИМ-2 с амплитудами  $U_{\text{отс1}} = +12$  мВ,  $U_{\text{отс2}} = +98$  мВ,  $U_{\text{отс3}} = +170$  мВ,  $U_{\text{отс4}} = -50$  мВ,  $U_{\text{отс5}} = -137$  мВ,  $U_{\text{отс6}} = -412$  мВ. Кодирование осуществляется в симметричном коде. Определить разрядность кода, структуру кодовых комбинаций, образующихся на выходе кодера или аналого-цифрового преобразователя (АЦП), для заданных отсчетов АИМ-2 сигнала, мощность шумов



квантования на единичном сопротивлении и ожидаемую при этом защищенность от шумов квантования, если исходный сигнал представляет речевой сигнал от единственного источника.

17. Решить задачу 16 для следующих условий:  $U_{\text{отс1}} = 605 \text{ мВ}$ ,  $U_{\text{отс2}} = 37 \text{ мВ}$ ,  $U_{\text{отс3}} = 621 \text{ мВ}$ ,  $U_{\text{отс4}} = 512 \text{ мВ}$ ,  $U_{\text{отс5}} = 1007 \text{ мВ}$ ,  $U_{\text{отс6}} = 1412 \text{ мВ}$ . Кодирование осуществляется натуральным кодом.

18. Решить задачу 16 для инверсного кода.

19. Решить задачу 17 для обратного кода.

20. Решить задачу 16 для следующих условий:  $U_{\text{отс1}} = 72 \text{ мВ}$ ,  $U_{\text{отс2}} = 128 \text{ мВ}$ ,  $U_{\text{отс3}} = 86 \text{ мВ}$ ,  $U_{\text{отс4}} = 53 \text{ мВ}$ ,  $U_{\text{отс5}} = 12 \text{ мВ}$ ,  $U_{\text{отс6}} = 32 \text{ мВ}$ . Кодирование осуществляется кодом Грея.

21. На вход 8-разрядного декодера, шаг квантования которого постоянен и равен 8 мВ, поступает следующая последовательность кодовых комбинаций: 11110111, 00001000, 10010101, 01010101. Декодер рассчитан на декодирование кодовых комбинаций, построенных по законам симметричного кода. Определить полярность и амплитуду импульсов отсчетов (АИМ-2), образующихся на выходе декодера или цифро-аналогового преобразователя (ЦАП).

22. Решить задачу 21 для следующих кодовых комбинаций: 01001100, 10001100, 00110100 и 10101010, построенных по законам натурального (простого) кода.

23. Объясните необходимость неравномерного квантования, методы неравномерного квантования. Основные законы компандирования, их достоинства и недостатки. Способы реализации.

24. Построить график защищенности от шумов квантования для  $\mu$ -закона компандирования при  $\mu = 127$ .

25. На вход нелинейного кодера (НК) с характеристикой компандирования А-87,6/13 после линейного квантования с шагом 80 поступают отсчеты, равные  $-273\delta_0$ ,  $+1613\delta_0$ ,  $+113\delta_0$ ,  $-23\delta_0$ ,  $+12\delta_0$ ,  $-43\delta_0$ ,  $+183\delta_0$ ,  $+613\delta_0$ . Определить структуру кодовых комбинаций на выходе НК и

величину шумов квантования для каждого отсчета. Для отсчета, равного  $+613\delta_0$ , изобразить граф алгоритма неравномерного квантования-кодирования.

26. На вход нелинейного декодера (НДК) с характеристикой компандирования А-87,6/13 поступают кодовые комбинации вида 00001010, 10011101, 10101111, 01001000, 11011100, 01100111, 10111010, 01110001. Определить амплитуды отсчетов на выходе НДК, если шаг квантования при линейном квантовании (или шаг квантования центрального сегмента) равен  $\delta_0$ .

## **Тема 2. Основные функциональные узлы цифровых систем передачи с ИКМ-ВРК**

1. Назовите основные способы снижения затухания при дискретизации аналоговых сигналов.

2. В чем сущность резонансного способа дискретизации?

3. Процесс кодирования в кодерах последовательного счета. Достоинства и недостатки кодеров последовательного счета.

4. Процесс кодирования в линейных кодерах поразрядного кодирования или взвешивающего типа.

5. Назначение инверторов  $DD_1$  и  $DD_2$  линейного кодера взвешивающего типа для двухполярных сигналов. Назначение устройства формирования ИКМ сигнала.

6. Назначение компандерной логики в схеме нелинейного кодера.

7. Назначение экспандерной логики в схеме нелинейного декодера.

8. Изобразите структурную схему цифрового регистра или преобразователя последовательного кода в параллельный.

9. На вход нелинейного кодера поступает АИМ-2 сигнал с амплитудой  $-1018\delta_0$ . Определить структуру кодовой комбинации на выходе нелинейного кодера.

10. На вход нелинейного декодера поступает ИКМ сигнал вида 00111101. Определите амплитуду АИМ-2 отсчета на выходе нелинейного декодера.

11. Сформулируйте основные требования к генераторному оборудованию ЦСП.

12. Укажите назначение основных элементов структурной схемы генераторного оборудования ЦСП.

13. Поясните работу схемы задающего генератора, изображенного на рис. 2.16.

14. Разработайте функциональную схему ЗГ с использованием двух инверторов.

15. Попробуйте изобразить временные диаграммы работы схемы в режиме внешней синхронизации.

16. Изобразите функциональную схему ДК для формирования канальных последовательностей импульсов с использованием кольцевого счетчика.

17. Изобразите функциональную схему ДК для формирования канальных последовательностей импульсов с использованием соответствующего разрядного счетчика.

### **Тема 3. Временное группообразование или мультиплексирование в ЦСП ИКМ-ВРК**

1. Объясните необходимость иерархии цифровых систем передачи на основе ИКМ-ВРК.

2. Перечислите способы объединения цифровых потоков, проведите их сравнительный анализ и назовите область применения.

3. Прокомментируйте сущность асинхронного объединения цифровых потоков и причины появления неоднородностей в агрегатном цифровом потоке.

4. Изобразите схему мультиплексирования цифровых потоков плезиохронной цифровой иерархии североамериканского стандарта.
5. Изобразите схему мультиплексирования цифровых потоков плезиохронной цифровой иерархии европейского стандарта.
6. Сущность согласования скоростей. Способы согласования скоростей, их достоинства и недостатки.
7. Изложите основные принципы формирования цикла передачи ЦСП ИКМ- ВРК.
8. Структурная схема тракта передачи оборудования асинхронного сопряжения.
9. Структурная схема тракта приема оборудования асинхронного сопряжения
10. Поясните назначение символов потока E1.
11. Поясните принцип формирования и назначение символов потока E2.
12. Поясните принцип формирования и назначение символов потока E3.
13. Поясните принцип формирования и назначение символов потока E4.
14. Сущность мультиплексирования цифровых потоков плезиохронной цифровой иерархии в технологии синхронной цифровой иерархии.
15. Формирование синхронного транспортного модуля STM-1 на основе потока E1.
16. Формирование синхронного транспортного модуля STM-1 на основе потока E3.
17. Формирование синхронного транспортного модуля STM-1 на основе потока E4.
18. Принцип функционирования запоминающего устройства блока асинхронного сопряжения.

19. Принцип функционирования временного детектора блока асинхронного сопряжения.

20. Принцип действия передатчика команд согласования скоростей блока асинхронного сопряжения.

21. Принцип функционирования приемника команд согласования скоростей блока асинхронного сопряжения.

22. Принцип функционирования устройства фазовой автоподстройки частоты блока асинхронного сопряжения.

23. Модульная структура аппаратуры мультиплексирования синхронного транспортного модуля STM-1 на основе компонентных потоков Е1.

24. Модульная структура аппаратуры мультиплексирования синхронного транспортного модуля STM-1 на основе компонентных потоков Е4.

#### **Тема 4. Цифровые разностные системы передачи**

1. Назовите основные недостатки передачи сигналов методом ИКМ с временным разделением каналов и пути их устранения.

2. Основные процессы формирования сигнала дифференциальной импульсно-кодовой модуляции (ДИКМ).

3. Разновидности ДИКМ, алгоритмы формирования их сигналов, достоинства, недостатки и область применения.

4. Основные процессы формирования сигнала дельта-модуляции (ДМ).

5. Физическая сущность явления перегрузки при формировании сигнала ДМ. Способы борьбы с явлением перегрузки.

6. Разновидности ДМ, алгоритмы формирования их сигналов, достоинства, недостатки и область применения.

7. Назовите основные параметры речевого сигнала и поясните их физическую сущность.

8. Принципы функционирования цифрового полосного вокодера.
9. Принципы функционирования цифрового формантного вокодера.
10. Принципы функционирования цифрового гармонического вокодера.
11. Принципы функционирования цифрового полувокодера.
12. Назовите основные параметры сигналов и каналов передачи данных и поясните их физическую сущность.
13. Основные методы ввода сигналов передачи данных (ПД) в каналы и тракты цифровых систем передач на основе ИКМ-ВРК.
14. Метод простого наложения, его реализация, достоинства, недостатки и область применения.
15. Метод скользящего индекса, его реализация, достоинства, недостатки и область применения.
16. Метод асинхронного ввода сигналов ПД, его реализация, достоинства, недостатки и область применения.
17. Метод фиксированного индекса, его реализация, достоинства, недостатки и область применения.

## **Тема 5. Синхронизация в цифровых системах передачи**

1. Назовите основные виды синхронизации в ЦСП ИКМ-ВРК и их назначение.
2. Основные требования, предъявляемые к системам синхронизации, и их физическая сущность.
3. Способы организации тактовой синхронизации.
4. Сущность пассивной фильтрации тактовой частоты.
5. Сущность активной фильтрации тактовой частоты.
6. Структурная схема выделителя тактовой частоты и временные диаграммы его работы
7. Фазовые дрожания, причины их возникновения, их оценка и влияние на качество передачи различных сообщений.

8. Проскальзывания, причины их возникновения и влияние на качество передачи различных сообщений.
9. Принципы нормирования количества проскальзываний.
10. Необходимость тактовой сетевой синхронизации.
11. Принципы построения тактовой синхронизации.
12. Требования к источникам сигналов тактовой синхронизации, их классификация.
13. Обобщенная схема передачи сигналов тактовой синхронизации в сети синхронной цифровой иерархии.
14. Причины нарушения цикловой синхронизации оконечных станций ЦСП ИКМ-ВРК и их влияние на качество передачи различных сообщений.
15. Основные блоки системы цикловой синхронизации и требования к ним.
16. Синхросигналы цикловой синхронизации (ЦС) и их основные параметры.
17. Классификация синхросигналов ЦС и способы их передачи.
18. Основные характеристики приемников синхросигналов ЦС и их классификация.
19. Алгоритм функционирования приемника синхросигнала ЦС с задержкой контроля и одноразрядным сдвигом.
20. Алгоритм функционирования приемника синхросигнала ЦС со скользящим поиском и одноразрядным сдвигом.
21. Алгоритм функционирования неадаптивного приемника синхросигнала ЦС.
22. Алгоритм функционирования адаптивного к вероятности ошибки приемника синхросигнала ЦС.
23. Алгоритм функционирования полностью адаптивного приемника синхросигнала ЦС.
24. Основные принципы определения структуры синхросигнала ЦС.

25. Основные принципы определения емкости накопителей по выходу из синхронизма и по входу в синхронизм приемника синхросигнала ЦС.

## **Тема 6. Линейный тракт цифровых систем передачи по электрическим кабелям**

1. Обобщенная структурная схема ЦЛТ, назначение основных блоков и требования к ним.

2. Назовите основные причины искажения сигналов в цифровых линейных трактах (ЦЛТ).

3. Назовите основные параметры ЦЛТ. Принципы их оценки.

4. Основные требования, предъявляемые к структуре линейного цифрового сигнала (ЛЦС). Их физическая сущность.

5. Основные параметры ЛЦС и их физическая сущность.

6. На вход преобразователя кода передачи ( $ПК_{пер}$ ), работающего в коде с чередованием полярности импульсов (ЧПИ), поступает следующая реализация ИКМ группового сигнала: ...0111000101010... Каким будет ЛЦС?

7. Предыдущую задачу решить для случая, когда  $ПК_{пер}$  работает в парно-избирательном коде.

8. На вход преобразователя кода передачи ( $ПК_{пер}$ ), работающего в коде с высокой плотностью единиц типа HDB-3, поступает следующая реализация ИКМ группового сигнала: ...01110000010000101011100001... Каким будет ЛЦС?

9. Предыдущую задачу решить для случая, когда  $ПК_{пер}$  работает:

а) в коде абсолютного биимпульсного сигнала (АБС);

б) в коде относительного биимпульсного сигнала (ОБС);

в) в коде CMI (Coding Mark Inversion);

г) в алфавитном коде типа 4ВЗТ.

10. На вход преобразователя кода приема  $ПК_{прм}$  поступает следующая реализация ЛЦС ...+ – 0 + 0 – – + + 0 – + – 0 0. Известно, что  $ПК_{пер}$  работает с ПИТ кодом. Какой вид примет эта реализация на выходе  $ПК_{прм}$ ?



11. На вход преобразователя кода приема ПК<sub>прм</sub> поступает следующая реализация ЦЛС ...0 – 0 + 0 – 0 0 0 0 + – + –... Известно, что ПК<sub>пер</sub> работает с ПИТ кодом. Какой вид примет эта реализация на выходе ПК<sub>прм</sub>?

12. Предыдущую задачу решить для случая, когда ПК<sub>пер</sub> работает в коде ЧПИ.

13. На вход преобразователя кода приема ПК<sub>прм</sub> поступает следующая реализация ЦЛС: + – + – + + + 0 + – + –. Известно, что ПК<sub>пер</sub> работает в алфавитном коде 4ВЗТ. Какой вид примет эта реализация на выходе ПК<sub>прм</sub>?

14. Назначение линейного регенератора (ЛР). Основные этапы процесса регенерации.

15. Поясните с помощью временных диаграмм процесс регенерации ЦЛС.

16. Перечислите основные типы помех и искажений в линейном цифровом тракте и принципы их оценки.

17. По ЦЛТ передаются сигналы типа АБС или ОБС. Определить минимально-допустимую защищенность для получения вероятности ошибки одиночного регенератора не более  $0,5 \times 10^{-11}$ .

18. По ЦЛТ передаются сигналы типа кода ЧПИ. Определить минимально-допустимую защищенность для получения вероятности ошибки одиночного регенератора не более  $0,5 \times 10^{-11}$ .

19. По ЦЛТ передаются сигналы типа кода ЧПИ. Определить вероятность ошибки одиночного регенератора, если соотношение сигнал-помеха на входе решающего устройства ЛР не превышает величины 250.

20. По ЦЛТ передаются сигналы типа кода АБС (ОБС). Определить вероятность ошибки одиночного регенератора, если соотношение сигнал-помеха на входе решающего устройства ЛР не превышает величины 250.

21. Что такое глаз-диаграмма? Ее основные параметры и их влияние на качество регенерации ЦЛС.

22. Назовите основные дестабилизирующие факторы, снижающие помехоустойчивость одиночного регенератора.

## **Тема 7. Линейный тракт цифровых систем передачи по оптическим кабелям**

1. Назовите основные достоинства ВОСП.
2. Назовите основные элементы обобщенной структурной схемы ВОСП.
3. Поясните методы мультиплексирования, используемые в ВОСП.
4. Поясните особенности распространения сигнала по оптическому волокну (ОВ)
5. В чем заключается сущность и каковы возможности спектрального уплотнения?
6. Сравните возможности различных технологий спектрального уплотнения.
7. Какие требования предъявляются к источникам оптического излучения?
8. Назовите основные типы и характеристики источников оптического излучения.
9. Какие требования предъявляются к приемникам оптического излучения?
10. Назовите основные типы и характеристики приемников оптического излучения.
11. Назовите способы модуляции оптического излучения.
12. Назовите типы оптических модуляторов и поясните принципы их работы.

### **Критерии оценивания компетенций**

Оценка «отлично» выставляется студенту, если он выполняет в полном объеме и безошибочно задания оценочных средств по дисциплине, относящиеся к данной компетенции (индикаторам достижения компетенции), при этом демонстрирует на высоком уровне способность решать стандартные

и нестандартные прикладные практические задачи проектирования информационно-коммуникационных систем и сетей.

Оценка «хорошо» выставляется студенту, если он выполняет, допуская незначительные ошибки, задания оценочных средств по дисциплине, относящиеся к данной компетенции (индикаторам достижения компетенции), при этом демонстрирует на среднем уровне способность решать только стандартные прикладные практические задачи проектирования информационно-коммуникационных систем и сетей.

Оценка «удовлетворительно» выставляется студенту, если он выполняет, допуская ошибки, исправление которых осуществляет с помощью наводящих вопросов преподавателя, задания оценочных средств по дисциплине, относящиеся к данной компетенции (индикаторам достижения компетенции), при этом демонстрирует на минимальном уровне способность решать стандартные прикладные практические задачи проектирования информационно-коммуникационных систем и сетей.

Оценка «неудовлетворительно» выставляется студенту, если он, выполняя задания оценочных средств по дисциплине, не достигает минимального уровня сформированности компетенции (индикаторам достижения компетенции), при этом не демонстрирует способность решать стандартные прикладные практические задачи проектирования информационно-коммуникационных систем и сетей, допускает грубые ошибки, которые не может исправить даже с помощью наводящих вопросов преподавателя.

## **4 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПОДГОТОВКЕ К ЛАБОРАТОРНЫМ РАБОТАМ**

При проведении лабораторных работ по дисциплине «Цифровые системы передачи» студент должен знать и владеть навыками, содержание, которых представлено в программе по дисциплине.

### **5.1. Методические указания при подготовке к лабораторной работе**

Подготовку к лабораторной работе рекомендуется проводить в следующей последовательности:

- уяснить тему и цель, предстоящего занятия;
- изучить теоретический материал в соответствии с темой занятия (рекомендуется использовать рекомендованную литературу, конспект лекций, учебное пособие (практикум)).

### **5.2. Методические указания при выполнении задания лабораторной работы**

При выполнении задания лабораторной работы студенты должны строго соблюдать, установленные правила охраны труда.

При выполнении задания студентам рекомендуется:

- уяснить цель, выполняемых заданий и способы их решения;
- задания, выполнять в той последовательности, в которой они указаны в практикуме;
- при выполнении задания и изучении теоретического материала использовать помощь преподавателя;
- оформить отчет по лабораторной работе;
- ответить на контрольные вопросы.

### **5.3. Методические указания при подготовке к защите лабораторной работы**

При подготовке к защите лабораторной работы студентам рекомендуется:

- подготовить отчет по лабораторной работе;
- уметь обосновать, сделанные выводы;
- закрепить знания теоретического материала по теме занятия (рекомендуется использовать контрольные вопросы);
- знать порядок проведения расчетов (проводимых исследований).

Защита лабораторной работы осуществляется путем собеседования студента с преподавателем. При собеседовании студент представляет на проверку отчет по лабораторной работе.

Собеседование со студентом, претендующим на оценку «отлично» проводится по вопросам повышенного уровня обученности.

Собеседование со студентом, не претендующим на оценку «отлично» проводится по вопросам базового уровня, которые приведены в фонде оценочных средств по дисциплине.

#### **Вопросы (задания) для проверки уровня обученности**

#### **Лабораторная работа 1. Исследование особенностей работы систем кодирования информации**

1. Охарактеризуйте существующие типы кодирования информации.
2. На каких принципах основана дискретизация непрерывных сообщений по времени. В каких устройствах она впервые нашла применение?
3. Какая разница между дискретизацией и квантованием непрерывных сообщений?
4. Для чего требуется избыточность отсчетов при дискретизации непрерывных сообщений по времени, и какими параметрами она определяется?

5. Какие типы кодов получаются при кодировании источников информации?
6. Назовите и охарактеризуйте параметры первичного кода.
7. Что собой представляют взвешенные и невзвешенные коды? Назовите достоинства последнего. В каких случаях эти достоинства наиболее ярко проявляются?
8. Перечислите особенности кода Грея. К какому типу кодов он относится?
9. Как классифицируются корректирующие коды? Что послужило причиной их разработки? Опишите историю их развития.
10. Каким образом исправляются ошибки в кодах, которые обнаруживают только их?
11. В чем состоят основные принципы коррекции ошибок?
12. Дайте определения разрешенных и неразрешенных кодовых комбинаций.
13. Дайте определение кодового расстояния.
14. При каких условиях код может обнаруживать или исправлять ошибки?
15. Как используется корректирующий код в системах со стиранием?
16. Какие характеристики определяют корректирующие способности кода?
17. Каким образом составляются кодовые комбинации в систематических кодах?
18. Как формируются контрольные символы систематических кодов? Как определяется их необходимое количество?
19. Что такое синдром?
20. На чем основан принцип корректирования ошибок с использованием контрольного числа?
21. Назовите основные особенности кодов Хэмминга.

22. Как осуществляется процедура кодирования в семизначном коде Хэмминга?

23. Каким образом формируются циклические коды?

24. Запишите код 110101101 в виде многочлена.

25. Какой двоичный многочлен называется нормированным?

26. Что собой представляет порождающий многочлен и как он определяется?

27. Каким образом производится умножение и деление многочленов с приведением подобных членов по модулю 2?

28. Какое свойство порождающего многочлена позволяет использовать его для обнаружения и исправления ошибок циклического кода?

29. Что является опознавателем безошибочной передачи циклического кода?

30. Охарактеризуйте способы генерации циклического кода.

31. Найдите генераторный многочлен

$$h(x) = (x^n + 1)/g(x) = (x^7 + 1)/(x^4 + x^3 + x^2 + x + 1).$$

Напомним, что деление многочленов выполняется обычным (в столбик) образом, но с приведением подобных членов по модулю 2.

32. Каким образом размещаются информационные и проверочные биты в корректирующем циклическом коде?

## **Лабораторная работа 2. Исследование особенностей работы систем линейного кодирования**

1. Назовите отличия униполярных кодов от биполярных.

2. Прокомментируйте особенности и области применения интерфейса RS-232.

3. Какие параметры имеет кабель для соединения двух устройств с использованием интерфейса RS-232? Какой длины он бывает?

4. Назовите длину пакета применяемого для обмена данными с использованием интерфейса RS-232.

5. Опишите принцип формирования манчестерского кода.
6. Какие переходы сигнала в манчестерском коде используются в приемнике для синхронизации?
7. Какие значения напряжения используются в качестве низкого и высокого уровней в сети Ethernet?
8. Какой недостаток манчестерского кода обусловлен тем обстоятельством, что его сигналы дважды меняют свое состояние за период тактовой частоты?
9. Поясните принципы дифференциального кодирования.
10. Разработайте схему NRZI-декодера и прокомментируйте его работу.
11. Используя описанный в тексте алгоритм АМІ-кодирования и принципы построения предыдущих схем, разработайте схему АМІ-кодера/декодера.
12. Какой положительный эффект можно получить при использовании сложных сигналов, и в частности псевдослучайных бинарных последовательностей? Назовите параметры таких последовательностей.
13. В чем заключается процесс скремблирования цифровых информационных сигналов?
14. Охарактеризуйте существующие типы скремблирующих устройств.
15. Чем объясняются такие положительные свойства скремблирования, как получение более равномерного спектра сигнала в линии связи и более быстрого восстановления синхронизации?
16. Какие требования предъявляются к дескремблеру приемника?
17. В чем заключается процедура перемежения/деперемежения?
18. В каком месте цифровой системы передачи информации ставится перемежитель и деперемежитель?

### **Лабораторная работа 3. Исследование особенностей работы модулирующих и демоделирующих устройств**

1. Прокомментируйте особенности реализации цифровой модуляции.



2. Какими параметрами определяются характеристики частотных модуляторов?
3. Каким образом можно определить число боковых частот и ширину спектра ЧМн колебания?
4. Какими эффектами сопровождается фазовая манипуляция?
5. Имеются ли отличия в ширине спектра ФМн и ЧМн колебания?
6. Что собой представляют многопозиционные модулированные сигналы и какую роль они выполняют в цифровых системах передачи информации?
7. Какие параметры несущего сигнала изменяются при ШИМ?
8. Охарактеризуйте ОШИМ-1 и ОШИМ-2. Перечислите недостатки ОШИМ-2.

#### **Лабораторная работа 4. Исследование особенностей работы каналов передачи информации**

1. Проведите классификацию каналов по форме передаваемых сигналов и эксплуатационным требованиям.
2. Охарактеризуйте области применения радиоканалов.
3. Какие устройства применяют в качестве источников и детекторов световых сигналов в системах с ВОЛС?
4. Проведите классификацию каналов со случайными параметрами.
5. Назовите виды искажений в дискретных каналах.
6. Какие виды неоднородностей вызывают замирания сигналов в свободном пространстве?
7. Чем вызывается многолучевое распространение сигнала? Для каких типов сигналов оно характерно?
8. Что такое межсимвольная интерференция сигналов? Чем она вызывается в случае проводных линий связи?
9. Охарактеризуйте многолучевые каналы с доплеровским растяжением спектра сигнала и полосу когерентности для таких каналов?

10. Что понимается под глубиной замираний?
11. Что такое скорость замираний?
12. Назовите отличия райсовского канала от релеевского. Приведите примеры реальных каналов связи, которые могут быть описаны данными моделями.

### **Лабораторная работа 5. Исследование особенностей работы систем уплотнения и разделения каналов**

1. Назовите особенности пространственного и дифференциального разделения каналов.
2. Прокомментируйте принципы работы дифференциального трансформатора.
3. Назовите преимущества и недостатки систем с частотным разделением каналов.
4. Прокомментируйте понятие поднесущей.
5. Перечислите и прокомментируйте преимущества и недостатки систем с временным разделением каналов.
6. Назовите пути устранения искажения импульсных сигналов при ограниченности полосы пропускания реального канала.
7. Перечислите и прокомментируйте преимущества и недостатки систем с фазовым разделением каналов.
8. Назовите функциональное назначение основных элементов модели канала с фазовым разделением.
9. Перечислите и прокомментируйте преимущества и недостатки систем с кодовым разделением каналов.
10. Назовите особенности применения технологии кодового разделения каналов в сети Интернет.
11. Чем отличается четвертая версия IP-протокола – IPv4 от IPv6?
12. Перечислите и прокомментируйте преимущества и недостатки систем с разделением каналов по уровню.

13. Как изменяются параметры работы схемы с разделением каналов по уровню при переводе сдвоенного ключа  $Z$ ?

### **Лабораторная работа 6. Исследование особенностей работы систем фильтрации сигнала**

1. Назовите типы цифровых фильтров.
2. Какие алгоритмы используются при создании цифровых фильтров?
3. Какими параметрами характеризуются фильтры?
4. Как классифицируются фильтры по виду АЧХ?
5. Дайте краткую характеристику фильтров Баттерворта, Бесселя и Чебышева.

### **Лабораторная работа 7. Исследование особенностей работы системы фазовой автоподстройки частоты**

1. Каким образом осуществляется синхронизация в асинхронных системах?
2. Прокомментируйте особенности реализации схем фазовой автоподстройки частоты.

### **Критерии оценивания компетенций**

Оценка «отлично» выставляется студенту, если он выполняет в полном объеме и безошибочно задания оценочных средств по дисциплине, относящиеся к данной компетенции (индикаторам достижения компетенции), при этом демонстрирует на высоком уровне способность решать стандартные и нестандартные прикладные практические задачи проектирования информационно-коммуникационных систем и сетей.

Оценка «хорошо» выставляется студенту, если он выполняет, допуская незначительные ошибки, задания оценочных средств по дисциплине, относящиеся к данной компетенции (индикаторам достижения компетенции), при этом демонстрирует на среднем уровне способность решать только

стандартные прикладные практические задачи проектирования информационно-коммуникационных систем и сетей.

Оценка «удовлетворительно» выставляется студенту, если он выполняет, допуская ошибки, исправление которых осуществляет с помощью наводящих вопросов преподавателя, задания оценочных средств по дисциплине, относящиеся к данной компетенции (индикаторам достижения компетенции), при этом демонстрирует на минимальном уровне способность решать стандартные прикладные практические задачи проектирования информационно-коммуникационных систем и сетей.

Оценка «неудовлетворительно» выставляется студенту, если он, выполняя задания оценочных средств по дисциплине, не достигает минимального уровня сформированности компетенции (индикаторам достижения компетенции), при этом не демонстрирует способность решать стандартные прикладные практические задачи проектирования информационно-коммуникационных систем и сетей, допускает грубые ошибки, которые не может исправить даже с помощью наводящих вопросов преподавателя.

## **СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ**

### **Перечень основной литературы**

1. Берлин, А. Н.; Высокоскоростные сети связи / А.Н. Берлин. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 452 с. : ил. - <http://biblioclub.ru/>.

2. Сутягина, Л.Н.; Проектирование городской наложенной мультисервисной сети связи общего пользования Электронный ресурс : учебно-методическое пособие / Л.Н. Сутягина. - Самара : Поволжский государственный университет телекоммуникаций и информатики, 2016. - 39 с. - Книга находится в базовой версии ЭБС IPRbooks.

3. Дьяконов, В. П. VisSim+Mathcad+MATLAB. Визуальное математическое моделирование / В. П. Дьяконов. - VisSim+Mathcad+MATLAB. Визуальное математическое моделирование, 2022-05-25. - Москва : СОЛОН-Пресс, 2017. - 384 с. - Электронный ресурс. - Книга находится в премиум-версии ЭБС IPR BOOKS. экземпляров неограничено.

4. Карлащук, В. И. Электронная лаборатория на IBM PC. Лабораторный практикум на Electronics Workbench и VisSim по элементам телекоммуникационных систем : практическое пособие / В.И. Карлащук. - Москва : СОЛОН-ПРЕСС, 2008. - 480 с. - <http://biblioclub.ru/>, экземпляров неограничено.

### **Перечень дополнительной литературы**

1 Голиков, А. М.; Транспортные и мультисервисные системы и сети связи : учебное пособие / А.М. Голиков ; Министерство образования и науки Российской Федерации ; Томский Государственный Университет Систем Управления и Радиоэлектроники (ТУСУР). - Томск : Томский государственный университет систем управления и радиоэлектроники, 2015. - 102 с. : схем. - <http://biblioclub.ru/>.

2 Кузьмич, Р. И.; Вычислительные системы, сети и телекоммуникации Электронный ресурс : Учебное пособие / Р. И. Кузьмич, А. Н. Пупков, Л. Н. Корпачева. - Красноярск : Сибирский федеральный университет, 2018. - 120 с. - Книга находится в премиум-версии ЭБС IPR BOOKS. - ISBN 978-5-7638-3943-2

3 Фокин, В. Г.; Проектирование оптической сети доступа : учебное пособие / В.Г. Фокин ; Министерство связи и массовых коммуникаций Российской Федерации ; Федеральное агентство связи ; ФГОБУ ВПО «СибГУТИ». - Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2012. - 311 с. : ил., табл., схем. - <http://biblioclub.ru/>.