

Федосов В. П., Нестеренко А. К.

# Цифровая обработка сигналов в LabVIEW

Под редакцией Федосова В. П.



Москва, 2007

УДК 681.3.06(075.8)  
ББК 32.973.26-108.2\*\*\*  
Б 28\*\*\*

*Рецензенты:*

кафедра «Радиоэлектронные системы» Южно-Российского государственного университета экономики и сервиса, г. Шахты Ростовской области, заведующий кафедрой, канд. техн. наук, профессор **Марчук В. И.**;  
доктор техн. наук, профессор кафедры радиопизики Волгоградского государственного университета **Захарченко В. Д.**

**Федосов В. П., Нестеренко А. К.**

Цифровая обработка сигналов в LabVIEW: учеб. пособие / под ред. В. П. Федосова. – М.: ДМК Пресс, 2007. – 456 с.

ISBN 5-8327-0269-7

Учебное пособие содержит материалы для освоения программно-аппаратных средств National Instruments по спектральному анализу, применению окон сглаживания, узлов цифровых систем связи, синтезу и анализу цифровых фильтров, программированию и тестированию сигнальных процессоров, программированию ПЛИС, виброакустическому анализу, проектированию цифровых фильтров, сопряжению LabVIEW с другими программными средствами для цифровой обработки сигналов. Пособие предназначено для студентов, магистрантов, аспирантов и специалистов, изучивших начальный курс LabVIEW.

УДК 681.3.06(075.8)  
ББК 32.973.26-108.2\*

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

© Федосов В. П., Нестеренко А. К., 2007  
© Оформление, ДМК Пресс, 2007

ISBN 5-8327-0269-7



Предисловие .....	16
▼ 1	
Введение в цифровую обработку сигнала и анализ в LabVIEW .....	18
▼ 2	
Цифровая связь и LabVIEW .....	27
2.1. Обычный цифровой приемник .....	30
2.2. Приемник подвыборки .....	31
Резюме .....	36
▼ 3	
Получение сигнала в LabVIEW .....	38
3.1. Сигнал в обычном цифровом приемнике .....	38
3.2. Сигнал в цифровом приемнике с прореживанием выборки .....	43
3.2.1. Выбор частоты дискретизации .....	45
3.2.2. ОСШ при прореживании выборки .....	46
3.2.3. Прореживание выборки и спектральное размещение сигнала .....	53
3.3. Другие методы дискретизации .....	54
3.3.1. Цифровой осциллограф .....	55
3.3.2. Анализатор спектра радиосигнала .....	55
3.3.3. Карта дискретизации аналогового сигнала .....	56
3.3.4. Звуковая карта .....	56
Резюме .....	57
▼ 4	
Цифровая фильтрация(Digital Filtering) .....	60
4.1. Введение в фильтрацию .....	60
4.2. Преимущества цифровой фильтрации по сравнению с аналоговой фильтрацией .....	60

4.3. Классификация цифровых фильтров .....	61
4.3.1. Импульсная характеристика (Impulse Response) .....	61
4.3.2. Классификация фильтров по импульсной характеристике .....	62
4.3.3. Коэффициенты фильтра .....	63
4.3.4. Характеристики идеальных фильтров .....	63
4.3.5. Практические (неидеальные) фильтры .....	65
4.3.6. Полоса перехода .....	65
4.4. Пульсации АЧХ в полосе пропускания и в полосе ослабления .....	65
4.5. Частота дискретизации .....	67
4.6. КИХ-фильтры .....	67
4.6.1. Отводы (Taps) .....	68
4.6.2. Проектирование КИХ-фильтров .....	68
4.6.3. Проектирование КИХ-фильтров с помощью окон (Windowing) .....	72
4.6.4. Проектирование оптимальных КИХ-фильтров на основе использования алгоритма Parks-McClellan .....	73
4.6.5. Проектирование Equiripple КИХ-фильтров на основе использования алгоритма Parks-McClellan .....	73
4.6.6. Проектирование узкополосных КИХ-фильтров .....	74
4.6.7. Проектирование широкополосных КИХ-фильтров .....	76
4.7. БИХ(IIR)-фильтры .....	77
4.7.1. Каскадная форма БИХ-фильтрации .....	78
4.7.2. Фильтрация второго порядка .....	79
4.7.3. Фильтрация четвертого порядка .....	80
4.7.5. Типы БИХ-фильтров .....	81
4.7.6. Минимизация пиковой ошибки .....	81
4.7.7. Фильтры Баттерворта (Butterworth) .....	81
4.7.8. Фильтры Чебышева .....	82
4.7.9. Фильтры Чебышева II .....	82
4.7.10. Эллиптические фильтры .....	84
4.7.11. Фильтры Бесселя .....	85
4.7.12. Проектирование БИХ-фильтров .....	85
4.7.13. Характеристики БИХ-фильтра в LabVIEW .....	87
4.7.14. Переходный отклик .....	88
4.8. Сравнение КИХ- и БИХ-фильтров .....	88
4.9. Нелинейные фильтры .....	89
4.10. Выбор проекта цифрового фильтра .....	91

## ▼ 5

<b>Спектральный анализ</b> .....	93
5.1. Различия между частотной областью и временной областью .....	93
5.2. Отношения Парсеваля .....	95

5.3. Преобразование Фурье .....	95
5.4. Дискретное преобразование Фурье .....	97
5.4.1. Отношения между выборками из N отсчетов в области частот и в области времени .....	97
5.4.2. Пример ДПФ .....	98
5.4.3. Информация об амплитуде и фазе .....	100
5.4.4. Частотный интервал между выборками ДПФ .....	101
5.5. Основные принципы БПФ .....	103
5.5.1. Вычисление частотных составляющих .....	104
5.5.2. Быстродействие БПФ .....	104
5.5.3. Дополнение нулями .....	105
5.5.4. Виртуальный прибор БПФ (VI FFT) .....	106
5.5.5. Отображение частотной информации после преобразования .....	106
5.5.6. Двухстороннее БПФ, центрированное относительно постоянной составляющей .....	108
5.5.7. Математическое представление двустороннего БПФ, центрированного относительно постоянной составляющей .....	108
5.5.8. Создание двустороннего БПФ с постоянной составляющей в центре .....	109
5.6. Спектр мощности .....	111
5.6.1. Преобразование двустороннего спектра мощности в односторонний спектр мощности .....	112
5.6.2. Потеря фазовой информации .....	113
5.6.3. Вычисления на основе спектра .....	114
5.6.4. Оценка мощности и частоты .....	114
5.6.5. Вычисление шумового уровня и спектральная плотность мощности .....	115
5.6.6. Вычисление амплитудных и фазовых спектров .....	116
5.6.7. Вычисление амплитуды в вольтах среднеквадратических значений ( $V_{rms}$ ) и начальной фазы в градусах .....	118
5.6.8. Частотная характеристика .....	118
5.7. Перекрестный (взаимный) спектр мощности (Cross Power Spectrum) .....	119
5.7.1. Частотная характеристика и анализ четырехполюсников .....	120
5.7.2. Частотная характеристика .....	120
5.7.3. Импульсная характеристика .....	121
5.8. Функция когерентности .....	121
5.9. Работа с окнами .....	122
5.10. Усреднение для улучшения измерений .....	123
5.10.1. Среднеквадратическое усреднение (RMS Averaging) .....	123
5.10.2. Векторное усреднение (Vector Averaging) .....	124
5.10.3. Пиковое усреднение (Peak Hold) .....	124

5.11. Взвешивание (Weighting) .....	124
5.12. Обнаружение эхосигнала (Echo Detection) .....	125
<b>▼ 6</b>	
<b>Примеры спектрального анализа в LabVIEW</b> .....	128
6.1. Преобразования низкого уровня в области частоты .....	128
6.1.1. Простое БПФ .....	130
6.1.2. Улучшенное БПФ .....	132
6.2. Анализ результатов ДПФ .....	135
6.2.1. Спектральная утечка .....	135
6.2.2. Дискретизация формы окна .....	135
6.3. Спектральные преобразования высокого уровня .....	140
6.4. Добавление обычных кодов C к LabVIEW .....	143
6.5. Инструменты (Toolset) для спектральных измерений .....	147
Резюме .....	148
<b>▼ 7</b>	
<b>Окна сглаживания (Smoothing windows)</b> .....	149
7.1. Растекание спектра .....	149
7.1.1. Дискретизация целого числа периодов .....	151
7.1.2. Дискретизация с нецелым числом периодов .....	152
7.2. Сигналы с окнами .....	154
7.3. Характеристики сигналов с различными окнами сглаживания .....	158
7.3.1. Основной лепесток спектра окна сглаживания .....	159
7.3.2. Боковые лепестки спектра окна сглаживания .....	160
7.4. Прямоугольное окно (None) .....	161
7.5. Окно Хэннинга (Hanning) .....	161
7.6. Окно Хемминга (Hamming) .....	162
7.7. Окно Кайзера – Бесселя (Kaiser – Bessel) .....	163
7.8. Треугольное окно (Triangle) .....	164
7.9. Окно с плоской вершиной (Flat Top) .....	164
7.10. Экспоненциальное (Exponential) окно .....	165
7.11. Окно для спектрального анализа против окна для проектирования коэффициентов фильтра .....	166
7.11.1. Спектральный анализ .....	166
7.11.2. Окна для проектирования коэффициентов КИХ-фильтра .....	167
7.12. Выбор правильного окна сглаживания .....	168
7.13. Масштабирование окон сглаживания .....	169

**▼ 8**

<b>Многоскоростная обработка сигналов в LabVIEW</b> .....	171
8.1. Повышение частоты дискретизации .....	171
8.2. Уменьшение частоты дискретизации .....	173
8.3. Фильтры передискретизации .....	175
8.3.1. Фильтры полуполосы .....	177
8.3.2. Полифазные фильтры .....	179
Резюме .....	181

**▼ 9**

<b>Генерация сигналов в LabVIEW</b> .....	183
9.1. Основные функции .....	183
9.2. Синусоиды .....	185
9.2.1. Комплексный преобразователь .....	185
9.2.2. Функция sinc(x) .....	186
9.2.3. Линейно-частотно-модулированная (ЛЧМ) последовательность .....	188
9.3. Генерация моделей канала связи .....	190
9.3.1. Распределение Рэлея .....	190
9.3.2. Белый гауссовский шум .....	191
9.4. Формирование символов .....	193
Резюме .....	195

**▼ 10**

<b>Сборка узлов системы связи</b> .....	197
10.1. Модулятор .....	198
10.2. Демодулятор .....	200
10.3. Искажения в канале .....	204
10.4. Обнаружение и восстановление сигнала .....	210
10.4.1. Обнаружение и согласованная фильтрация .....	210
10.4.2. Пороговые решения .....	213
10.5. Синхронизация .....	213
10.5.1. Синхронизация по времени .....	213
10.5.2. Синхронизация по частоте .....	215
10.6. Модуляция в NI Toolset .....	216
Резюме .....	218

**▼ 11**

<b>Оптимизация обработки сигналов в LabVIEW</b> .....	219
---	-----

11.1. Общие руководящие принципы кодирования в LabVIEW .....	219
11.2. Подсказки в обработке сигналов .....	221
11.2.1. Линейная свертка на основе БПФ .....	221
11.2.2. Реальное БПФ .....	222
11.3. Дальнейшие применения цифровой обработки сигналов в LabVIEW .....	224
11.3.1. Корни дифференциального уравнения .....	226
11.3.2. Линейный прогнозирующий шифровальщик речи .....	226
Резюме .....	230

## ▼ 12

### **Сопряжение LabVIEW с другими программными продуктами для обработки цифровых сигналов. Пакет LabVIEW Simulation Interface Toolkit .....**

231

12.1. Тестирование моделей с использованием интерфейса LabVIEW .....	232
12.2. Дополнительные возможности по тестированию моделей .....	232
12.3. Импортирование моделей из Simulink в LabVIEW .....	235
12.4. Перенос модели в систему реального времени .....	235

## ▼ 13

### **Тестирование и программирование цифровых сигнальных процессоров (DSP) в LabVIEW .....**

237

13.1. Пакет LabVIEW DSP Test Toolkit .....	237
13.1.1. Функции автоматизации CCS .....	237
13.1.2. Обмен данными с кодом DSP .....	238
13.1.3. Функции углубленного тестирования сигнальных процессоров (DSP Test Advanced VIs) .....	238
13.2. Пакет LabVIEW DSP Module .....	239
13.2.1. Поддерживаемые аппаратные средства .....	239
13.2.2. Области применения .....	240

## ▼ 14

### **Виброакустический анализ в LabVIEW .....**

243

14.1. Функциональные возможности Sound and Vibration Toolkit .....	243
14.2. Поддерживаемые аппаратные средства .....	246
14.3. Перечень и описание виртуальных приборов и функций, входящих в пакет .....	246

14.3.1. Масштабирование .....	246
14.3.2. Калибровка .....	246
14.3.3. Тестирование (измерение) уровней сигналов .....	247
14.3.4. Взвешивание .....	248
14.3.5. Интегрирование .....	248
14.3.6. Генерация .....	249
14.3.7. Уровень вибрации .....	250
14.3.8. Уровень звука .....	250
14.3.9. Октавный анализ .....	250
14.3.10. Частотный анализ .....	251
14.3.11. Анализ переходных процессов .....	251
14.3.12. Каскадный график .....	251
14.3.13. Гармонический анализ .....	251
14.3.14. Анализ искажений .....	251
14.3.15. Чистый тон .....	253

## ▼ 15

### **Программирование ПЛИС (FPGA) в LabVIEW .....**

254

15.1. Реконфигурируемые устройства ввода/вывода (RIO) .....	255
15.2. Создание проектов .....	256
15.3. Последовательность программирования ПЛИС .....	258
Резюме .....	260

## ▼ 16

### **Инструментарий проектирования цифровых фильтров в LabVIEW .....**

261

16.1. Введение в инструментарий проектирования цифровых фильтров в LabVIEW .....	261
16.1.1. Краткий обзор процесса проектирования цифрового фильтра .....	261
16.1.2. Краткий обзор инструментария LabVIEW для проектирования цифрового фильтра .....	262
16.1.3. Обобщенный метод Ремеза и перечень р-х норм синтеза алгоритмов .....	262
16.1.4. Большой выбор структур фильтра .....	263
16.1.5. Специальное проектирование цифрового фильтра .....	263
16.1.6. Проектирование фильтра с фиксированной запятой .....	263
16.1.7. Создание кода для ПЛИС (FPGA) и цифровых сигнальных процессоров (DSP) .....	264
16.1.8. Проектирование многоскоростных цифровых фильтров .....	264

16.1.9. Всесторонние инструментальные средства анализа .....	264
16.1.10. Заметки к пользовательской расширенной библиотеке анализа в LabVIEW (Current LabVIEW Advanced Analysis Library Users) .....	264
16.2. Основы проектирования цифровых фильтров .....	265
16.2.1. Применение цифровых фильтров .....	265
16.2.2. Терминология цифрового фильтра .....	266
16.2.3. Свойства фильтра .....	266
16.2.4. КИХ- и БИХ-фильтры .....	267
16.2.5. Математические определения .....	267
16.2.6. Различие КИХ- и БИХ-фильтров .....	268
16.2.7. Свойства КИХ- и БИХ-фильтров. Технические требования для цифровых фильтров .....	269
16.2.8. Тип фильтра .....	269
16.2.9. Частота дискретизации .....	269
16.2.10. Технические требования к фильтру .....	269
16.2.11. Методы проектирования .....	270
16.2.12. Анализ цифрового фильтра .....	272
16.2.13. Полюса и нули .....	273
16.2.14. Поле нулей и полюсов .....	273
Резюме .....	274
16.3. Основы проектирования фильтра с плавающей запятой .....	275
16.3.1. Типичный процесс проектирования цифрового фильтра с плавающей запятой .....	275
16.3.2. Проектирование фильтров с плавающей запятой .....	276
16.3.3. Ввод технических характеристик фильтра .....	276
16.3.4. Ввод технических характеристик фильтра в цифровом управлении, основанном на текстовом интерфейсе .....	276
16.3.5. Ввод технических характеристик фильтра в графическом интерфейсе .....	278
16.3.6. Выбор метода проектирования .....	278
16.3.7. Анализ проекта фильтра .....	280
16.3.8. Частотная характеристика .....	280
16.3.9. Поле нулей и полюсов .....	280
16.3.10. Определение порядка фильтра .....	280
16.3.11. Пример. Проектирование ФНЧ цифрового фильтра согласно параметрам .....	281
16.3.12. Пример. Фильтрация .....	282
Резюме .....	282
16.4. Основы проектирования фильтра с фиксированной запятой .....	284

16.4.1. Типичный процесс проектирования цифрового фильтра с фиксированной запятой .....	284
16.4.2. Осуществление фильтров с фиксированной запятой .....	285
16.4.3. Выбор структуры фильтра .....	287
16.4.4. Выбор структуры для КИХ-фильтров .....	287
16.4.5. Выбор структуры для БИХ-фильтров .....	288
16.4.6. Использование лестничных структур .....	288
16.4.7. Метод проектирования структуры по умолчанию .....	289
16.4.8. Моделирование фильтров с фиксированной запятой .....	289
16.4.9. Квантователи моделирования .....	291
16.4.10. Установка $wl$ и $iwl$ .....	292
16.4.11. Установка режима переполнения .....	293
16.4.12. Установка режима округления .....	293
16.4.13. Установка квантователей вручную .....	293
16.4.14. Проверка достоверности фильтров с фиксированной запятой .....	294
16.4.15. Анализ фильтра с фиксированной запятой .....	294
16.4.16. Пример. Анализ фильтра с фиксированной запятой .....	295
16.4.17. Моделирование фильтра с фиксированной запятой .....	296
16.4.18. Пример. Моделирование фильтра с фиксированной запятой .....	298
16.4.19. Генерация кода .....	300
16.4.20. Извлечение целочисленных коэффициентов фильтра с фиксированной запятой .....	301
16.4.21. Производство кода C для фильтра с фиксированной запятой .....	301
16.4.22. Производство кода LabVIEW фильтра с фиксированной запятой .....	303
16.4.23. Структура целочисленных кодов LabVIEW Code SCTL-Optimized LabVIEW FPGA .....	303
16.4.24. Код SCTL-Optimized LabVIEW FPGA .....	303
16.4.25. Пример. Генерация кода LabVIEW для ПЛИС-Устройства из модели фильтра с фиксированной запятой .....	306
Резюме .....	308
16.5. Расширенные и специальные проекты фильтра .....	309
16.5.1. Фильтры с линейной ФЧХ и минимально-фазовые .....	309
16.5.2. Математическое определение .....	310
16.5.3. Типы КИХ-фильтров с линейной ФЧХ .....	310
16.5.4. Эксперимент с различными типами .....	311
16.5.5. Пример КИХ-фильтра с линейной ФЧХ .....	311
16.5.6. Преобразователи Гильберта (Hilbert) .....	312
16.5.7. Преобразователя Гильберта типа IV .....	313
16.5.8. Преобразователь Гильберта типа III .....	314
16.5.9. Дифференциаторы .....	315

16.5.10. Дифференциатор типа IV .....	316
16.5.11. Дифференциаторов типа III .....	316
16.5.12. Фильтр-пробка и пиковый фильтр .....	318
16.5.13. Пример фильтра-пробки .....	318
16.5.14. Гребенчатые фильтры .....	320
16.5.15. Пример гребенчатого фильтра .....	321
16.5.16. Фильтры с произвольной формой АЧХ .....	323
16.5.17. КИХ-фильтр с линейной ФЧХ и произвольной АЧХ .....	324
16.5.18. Пример фильтра с произвольной АЧХ .....	324
16.5.19. Пример инверсного компенсационного sinc-фильтра .....	325
16.5.20. БИХ-фильтр с произвольной АЧХ .....	326
16.5.21. Компенсатор групповой задержки .....	327
16.5.22. Пример компенсатора групповой задержки .....	327
16.5.23. Узкополосные КИХ-фильтры .....	329
16.5.24. Пример узкополосного фильтра .....	332
Резюме .....	332
16.6. Многоскоростные цифровые фильтры .....	334
16.6.1. Основы многоскоростной цифровой фильтрации .....	334
16.6.2. Повторная дискретизация .....	335
16.6.3. Децимация .....	335
16.6.4. Пример фильтра децимации .....	336
16.6.5. Интерполяция .....	337
16.6.6. Пример интерполирующего фильтра .....	339
16.6.7. Многоскоростные многоступенчатые фильтры .....	340
16.6.8. Многоступенчатый фильтр децимации .....	340
16.6.9. Гребенчатые каскадные фильтры-интеграторы (CIC) .....	341
16.6.10. Фильтр с нулевой ФЧХ .....	342
16.6.11. Разработка многоскоростного фильтра .....	344
16.6.12. Одноступенчатый проект многоскоростного фильтра .....	344
16.6.13. Проектирование многоступенчатых многоскоростных фильтров .....	346
16.6.14. Проектирование фильтра Найквиста .....	347
16.6.15. Фильтры Найквиста .....	348
16.6.16. Фильтры «поднятый косинус» .....	348
16.6.17. Фильтры полуполосы (halfband) .....	349
16.6.18. Анализ многоскоростных фильтров .....	350
16.6.19. Обработка сигналов в многоскоростном фильтре .....	350
Резюме .....	350
16.7. Современные методы проектирования фильтров .....	351
16.7.1. Проектирование методом Ремеза (Remez) .....	351

16.7.2. Использование ВП Remez Design .....	351
16.7.3. Определение заданной передаточной функции .....	352
16.7.4. Определение порядка фильтра .....	353
16.7.5. Выбор типа фильтра .....	353
16.7.6. Симметричные и несимметричные типы фильтра .....	353
16.7.7. Дифференциатор и типы фильтра Гильберта .....	354
16.7.8. Минимально-фазовый и максимально-фазовый типы фильтра .....	354
16.7.9. Аппроксимация АЧХ КИХ-фильтра .....	354
16.7.10. Проектирование минимально-фазовых и максимально-фазовых КИХ-фильтров .....	355
16.7.11. Проектирование минимально-фазовых и максимально-фазовых фильтров в ВП DFD Remez Design .....	357
16.7.12. Полосовой фильтр единственной точки по частоте .....	357
16.7.13. Точная установка значений передаточной функции .....	358
16.7.14. Ограничение пульсаций АЧХ .....	361
16.7.15. Метод Least Pth Norm Design .....	363
16.7.16. Использование ВП DFD Least Pth Norm Design .....	365
16.7.17. Определение заданной передаточной функции .....	365
16.7.18. Определение порядка фильтра .....	365
16.7.19. Определение порядка нормы .....	365
16.7.20. Определение ограничения радиуса полюса .....	366
16.7.21. Проектирование КИХ-фильтров с минимальной величиной r-й нормы и линейной ФЧХ .....	366
16.7.22. Проектирование БИХ-фильтров с приблизительно линейной ФЧХ .....	367
16.7.23. Проектирование минимально-фазового и максимально-фазового БИХ-фильтра .....	367
Резюме .....	369
16.8. Проектирование КИХ-фильтра .....	371
16.8.1. Описание задачи .....	371
16.8.2. Реализация фильтров в LabVIEW .....	371

## ▼ 17

<b>Машинное зрение в LabVIEW</b> .....	381
17.1. Интерактивный помощник NI Vision Assistant .....	381
17.2. Использование библиотеки IMAQ Vision в задачах машинного зрения .....	382
17.2.1. Последовательность создания приложений IMAQ Vision .....	383
17.2.2. Шаги по обработке изображений .....	386

## ▼ 18

**Использование LabVIEW в телекоммуникации.**

<b>Пакет NI Modulation Toolkit</b> .....	389
18.1. Использование функций библиотеки NI Modulation Toolkit при построении цифровых систем связи .....	390
18.2. Кодирование .....	390
18.3. Модуляция .....	392
18.4. Преобразование (повышение) частоты .....	393
18.5. Ухудшения (потери) .....	393
18.6. Преобразование (понижение) частоты .....	393
18.7. Демодуляция .....	393
18.8. Компенсация потерь .....	394
18.9. Декодирование .....	395
18.10. Измерение .....	395
18.11. Визуализация .....	395

## ▼ Приложение 1

**Разработка алгоритмов на основе использования LabVIEW**

<b>MathScript: Часть 1 – Ветви LabVIEW 8 MathScript</b> .....	397
П1.1. Что такое MathScript? .....	397
П1.2. Ветвь LabVIEW – программа MathScript .....	398
П1.3. Введение в LabVIEW и Графическое программирование (Graphical Programming) .....	398
П1.4. Использование ветви MathScript для разработки алгоритмов .....	402
П1.5. Окно LabVIEW MathScript .....	407

## ▼ Приложение 2

**Справочник виртуальных приборов ВП [10]**

<b>в телекоммуникациях</b> .....	411
----------------------------------	-----

## ▼ Приложение 3

**Лабораторная работа. Прохождение случайных колебаний**

<b>через линейные цепи и исследование явления нормализации</b> .....	440
Литература для углубленного изучения .....	446
Вопросы для самоконтроля .....	447
Описание средств для проведения исследования .....	447

Задание на предварительное внеаудиторное исследование .....	450
Задание на лабораторное исследование .....	451
Содержание отчета о проведении исследования .....	451
Контрольные вопросы .....	452

<b>Заключение</b> .....	453
-------------------------	-----

<b>Список литературы</b> .....	454
--------------------------------	-----

<b>Список литературы</b> .....	456
--------------------------------	-----

# Предисловие

Представляемое читателю учебное пособие не претендует на исчерпывающее изложение теоретических основ цифровой обработки сигналов и предназначено пользователям LabVIEW, имеющим начальные навыки составления блок-диаграмм виртуальных приборов, фронтальных панелей и опыт исследовательской работы с использованием LabVIEW. Пособие содержит материалы для освоения программных средств корпорации США National Instruments по спектральному анализу, применению окон сглаживания и узлов цифровых систем связи, по синтезу и анализу цифровых фильтров, программированию и тестированию сигнальных процессоров, программированию ПЛИС, виброакустическому анализу, проектированию цифровых фильтров, а также сопряжению LabVIEW с другими программными средствами для цифровой обработки сигналов.

В качестве примера приведены методические указания, блок-диаграмма и фронтальная панель лабораторной работы «Прохождение случайных сигналов через линейные цепи» по общепрофессиональной дисциплине «Радиотехнические цепи и сигналы» федерального компонента учебного плана по направлению «Радиотехника», в которой отражена исследовательская направленность инновационных образовательных программ.

Основой пособия послужили руководства: LabVIEW™ Digital Filter Design Toolkit User Manual, National Instruments, 2005; LabVIEW™ Analysis Concepts, National Instruments, 2004; LabVIEW™ and LabWindows™/CV Signal Processing Toolset User Manual, National Instruments, 2002 и электронное издание Cory L. Clark LabVIEW Digital Signal Processing and Digital Communication. – McGraw-Hill, 2005.

Использование новой информационной технологии в учебном процессе приведет к повышению уровня инженерной подготовки выпускников, заключающемуся в объединении знаний и навыков инженера и программиста. В настоящее время схемотехнические разработки в традиционном понимании отмирают и формируются новые возможности создания продукции на основе технологий с использованием цифровых систем и устройств. Инновационный подход в учебном процессе приведет к повышению требований и к преподавателям, формирующим интерес у студентов к изучаемым дисциплинам и способствующим превращению студентов в исследователей.

Сетевые возможности LabVIEW приведут к повышению роли самостоятельной работы студентов, позволят развить дистанционные методы в образовательном процессе. Кроме того, виртуальные измерительные приборы, созданные в LabVIEW, позволят освободить исследователя от рутинной работы получения и обработки данных, на основе наглядности повысят роль и понимание изучаемого явления в базовых дисциплинах специальностей и направлений.

Для изучения теоретических и практических (с примерами на MATLAB) основ цифровой обработки сигналов можно использовать учебную литературу, представленную в библиографическом списке, например [16, 17, 18].

Некоторые темы и определения повторяются в разных частях данного пособия, но каждый раз на другом уровне, способствующем лучшему усвоению тематики конкретного раздела или подраздела.

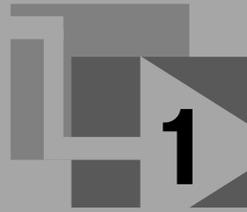
Выводы в разделах и подразделах приведены там, где, по мнению авторов, возникает необходимость в концентрации внимания на сведениях, приведенных в этих разделах, и обеспечении связи с другими разделами.

Книга предназначена для специалистов по цифровой обработке сигналов, а также для преподавателей, аспирантов и студентов, прошедших начальный курс обучения программированию в LabVIEW.

Первый раздел книги написан совместно В. П. Федосовым и Ф. А. Цветковым, разделы 12 – 15 и 18 написаны А. К. Нестеренко, остальные разделы – В. П. Федосовым.

Материалы [10, 36, 37, 38] предоставлены Погосом Сепояном – образовательной программой National Instruments в России и СНГ, г. Москва.

# Введение в цифровую обработку сигнала и анализ в LabVIEW



Цифровые сигналы находят применение всюду вокруг нас. Телефонные компании используют цифровые сигналы для передачи человеческого голоса. Радио, телевидение и аудиотехника постепенно преобразовываются в цифровые из-за превосходной точности представления исходных данных, малого шума и гибкости обработки сигналов. Данные передаются со спутников на наземные станции в цифровой форме. Изображения отдаленных планет и космоса часто обрабатываются в цифровой форме, чтобы убрать шум и извлечь полезную информацию. Экономические данные, результаты переписи населения и курсы ценных бумаг на фондовом рынке доступны всем в цифровой форме. Из-за многих преимуществ цифровой обработки сигналов (ЦОС) аналоговые сигналы все чаще преобразуют в цифровую форму, чтобы затем выполнить их обработку на компьютере.

Этот раздел является вводным в основы ЦОС и, в частности, в обработку сигналов и измерение их параметров и характеристик с помощью пакета программ LabVIEW. В этом пакете реализована концепция графического программирования алгоритмов преобразования и обработки цифровых сигналов. Программа алгоритма при этом выглядит как функциональная схема устройства обработки данных, а запрограммированная задача (приложение) предстает в виде виртуального прибора (ВП) или виртуального инструмента (Virtual Instrument). В LabVIEW уже заложен обширный набор способов обработки данных в виде готовых узлов (устройств), которые можно включить в виртуальный прибор, что существенно упрощает и ускоряет разработку приложений анализа данных.

**Дискретизация сигналов.** В большинстве практических приложений обработке подлежат сигналы, как правило, аналоговые, т.е. непрерывные во времени и принимающие бесчисленное множество значений; непосредственно к ним методы и алгоритмы ЦОС применить нельзя. Чтобы использовать цифровые методы обработки сигналов, необходимо сначала преобразовать аналоговый сигнал в его

цифровую форму. Для этого аналоговые сигналы сначала превращают в **дискретные сигналы** – в последовательность их отсчетов (дискретных значений), взятых через одинаковые интервалы времени, а затем каждый отсчет преобразуют в цифровой код (операция квантования по уровню). Только после этого к полученной последовательности цифровых кодов становится возможным применение методов и алгоритмов ЦОС.

***Примечание.** В российской математической и технической литературе совокупность отсчетов аналогового сигнала называется выборкой. В иностранной литературе под выборкой часто понимают один отсчет сигнала. В тексте далее, во избежание различных толкований, один отсчет сигнала будет также называться единичным измерением в выборке или выборкой единичного размера.*

Наиболее типична ситуация, когда аналоговый сигнал  $s(t)$  дискретизируется через постоянный интервал времени  $\Delta t$ , называемый периодом, интервалом или шагом дискретизации. Его обратная величина  $f_s = 1/\Delta t$  является частотой дискретизации. Круговая частота дискретизации определяется по формуле  $\Omega_s = 2\pi/\Delta t$ .

Каждое из дискретных значений  $x(t)$  в моменты  $t = 0, \Delta t, 2\Delta t, 3\Delta t, \dots$  является отсчетом или единичным измерением, а  $N$  таких отсчетов называется выборкой размера  $N$  (каждый отсчет можно также назвать выборкой единичного размера  $N = 1$ ). В реальных устройствах преобразование аналогового сигнала в цифровой осуществляется аналого-цифровым преобразователем (АЦП) (analog-digital converter – A/D).

Таким образом,  $x(0), x(\Delta t), x(2\Delta t), \dots$  являются единичными измерениями мгновенных значений аналогового сигнала в выборке бесконечного (теоретически) размера. Полученный дискретный сигнал может быть представлен набором выборочных данных

$$\{x(0), (\Delta t), x(2\Delta t), (3\Delta t), \dots, (k\Delta t), \dots\}.$$

На рис. 1.1 показан некоторый аналоговый сигнал на ограниченном интервале времени (непрерывная функция) и совокупность его отсчетов (точек), взятых через равные интервалы времени  $\Delta t$ .

Если моментам времени, в которые взяты отсчеты сигнала, присвоить номера, например,  $i = 0, 1, 2, \dots$ , то  $x[i] = x(i\Delta t)$  будет являться одним из этих отсчетов ( $i$ -м отсчетом).

Сигнал  $x(t)$ , таким образом, может быть представлен набором отсчетов (вектором-строкой)

$$\mathbf{X} = \{x(0), x(\Delta t), x(2\Delta t), x(3\Delta t), \dots, x(k\Delta t), \dots\} = \{x[0], x[1], x[2], x[3], \dots, x[k], \dots\}.$$

Очень важно при этом правильно выбрать интервал дискретизации  $\Delta t$ , т.е. решить задачу: как часто необходимо брать отсчеты аналогового сигнала, чтобы сохранить информацию, содержащуюся в нем, в дискретном сигнале без потерь или в пределах допустимых потерь (искажений).

На самом деле ответ очевиден: нужно брать отсчеты аналогового сигнала как можно чаще. При этом потери информации действительно можно сделать пренеб-

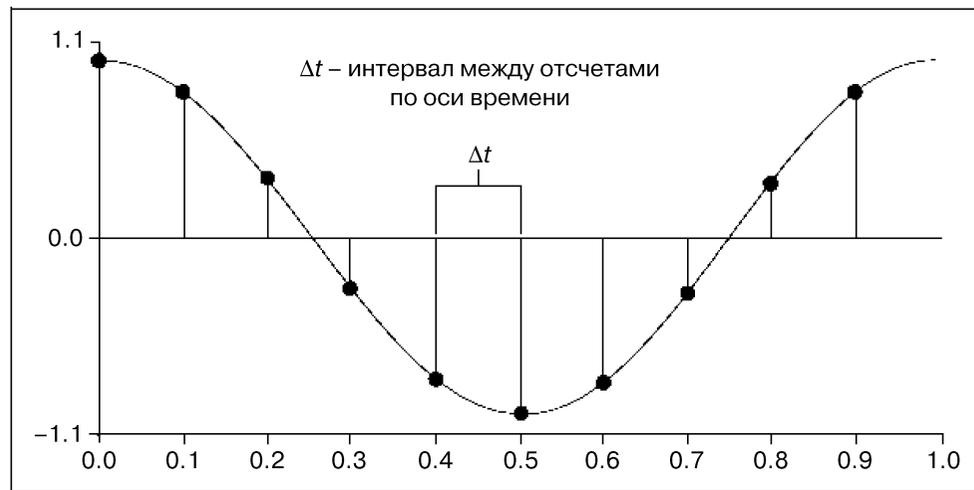


Рис. 1.1. Аналоговый сигнал и его выборка размера  $N = 10$

режимом малы. Однако такой подход заставляет повышать требования к узлам устройства ЦОС по быстродействию, а это, как следствие, приводит к неоправданно высокой стоимости этого устройства. Поэтому задача выбора частоты дискретизации сводится к нахождению такой **минимальной** частоты  $f_s$  дискретизации, при которой потери информации будут приемлемо малы.

Необходимо всегда иметь в виду, что последовательность  $\mathbf{X} = \{x[i]\}$  является совокупностью отсчетов и не содержит никакой информации о частоте дискретизации. Поэтому значения отсчетов, содержащиеся в  $\mathbf{X}$ , не несут никакой информации об абсолютных временных или частотных свойствах исходного аналогового сигнала.

В LabVIEW для работы с сигналами предусмотрено несколько типов данных – массивы (Array), осциллограммы (временные диаграммы) (Waveform) и динамические данные (Dynamic).

Данные типа Array представляют собой последовательность чисел, аналогичную набору отсчетов  $\mathbf{X}$ .

Данные типа Waveform содержат последовательность отсчетов, а также время получения первого отсчета, значение интервала дискретизации и комментарии к данным.

Данные типа Dynamic близки, по сути, к данным типа Waveform и также содержат последовательность отсчетов, сведения о дате и времени получения первого отсчета, значение интервала дискретизации, имя сигнала и др.

Обобщенная структурная схема устройства ЦОС представлена на рис. 1.2 со следующими обозначениями: АЦП – аналого-цифровой преобразователь, ЦВ – цифровой вычислитель; ЦАП – цифроаналоговый преобразователь; ФНЧ – сглаживающий фильтр нижних частот. На входе и выходе этого устройства сигналы аналоговые, соответственно  $s_{вх}(t)$  и  $s_{вых}(t)$ .

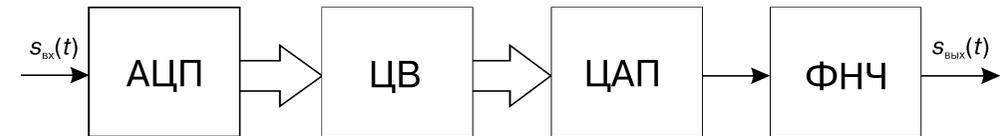


Рис. 1.2. Структурная схема устройства цифровой обработки сигналов

В частных случаях входной и выходной сигналы устройства ЦОС могут быть цифровыми, и тогда необходимость в использовании АЦП, ЦАП и ФНЧ отпадает.

Необходимые свойства устройства ЦОС обеспечиваются, в основном, заложенным в ЦВ алгоритмом преобразования последовательности отсчетов, поступающей с выхода АЦП. В определенном смысле устройство ЦОС универсально, так как заменой только алгоритма преобразования последовательности отсчетов в ЦВ можно существенно изменить свойства всего устройства ЦОС.

**Частота дискретизации и «ложные» частоты (alias).** Один из самых важных параметров устройства ЦОС – частота дискретизации, определяющая, как часто во времени выполняется аналого-цифровое (А/Д) преобразование. Она существенно зависит от свойств аналогового сигнала и допустимых потерь информации при дискретизации. Слишком малая частота дискретизации приводит к появлению в дискретном сигнале «ложных» составляющих [16, 17, 18], не содержащихся в исходном аналоговом сигнале, что вызывает значительные, часто недопустимые, искажения выходного сигнала  $s_{вых}(t)$  устройства ЦОС.

На рис. 1.3 и 1.4 показаны аналоговый гармонический сигнал с отмеченными на нем отсчетами и соответствующий дискретный сигнал при различных соотношениях частоты сигнала и частоты дискретизации.

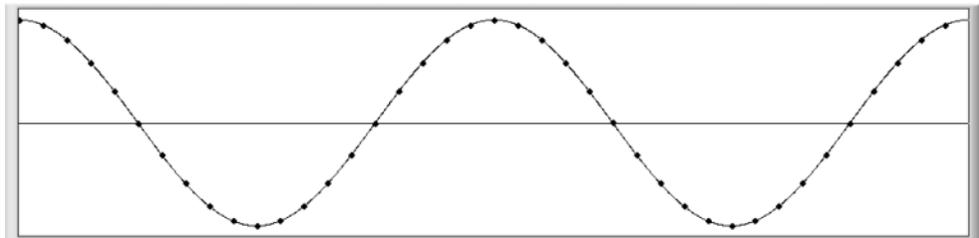
Из этих рисунков следует, что при частоте дискретизации  $f_s = 40$  кГц дискретный сигнал правильно представляет свойства аналогового сигнала с частотой  $f_0 = 2$  кГц и неправильно – с частотой  $f_0 = 42$  кГц. Следует заметить, что дискретные сигналы на рис. 1.3 и 1.4 одинаковы! То есть из аналогового сигнала с частотой  $f_0 = 42$  кГц сформировался дискретный сигнал, соответствующий аналоговому с частотой  $f_0 = 2$  кГц. Это можно описать и по-другому: произошел перенос высокочастотной составляющей в область низких частот – возникла составляющая с «ложной» низкой частотой (alias), которой нет в исходном аналоговом сигнале.

Для подробного ознакомления с данным эффектом можно запустить ВП `diskr_cos` и понаблюдать за получающимся дискретным сигналом при различных сочетаниях частоты аналогового сигнала и частоты дискретизации.

Частота «ложной» составляющей равна абсолютному значению наименьшей разности между частотой аналогового сигнала и частотами, кратными частоте дискретизации. Например, при частоте дискретизации  $f_s = 40$  кГц «ложные составляющие» с частотой 2 кГц будут получены при следующих частотах аналогового сигнала:

- $f_0 = 38$  кГц     $(2 = |38 - 40 \cdot 1|)$ ;
- $f_0 = 42$  кГц     $(2 = |42 - 40 \cdot 1|)$ ;
- $f_0 = 78$  кГц     $(2 = |78 - 40 \cdot 2|)$ ;

Аналоговый сигнал



Дискретный сигнал

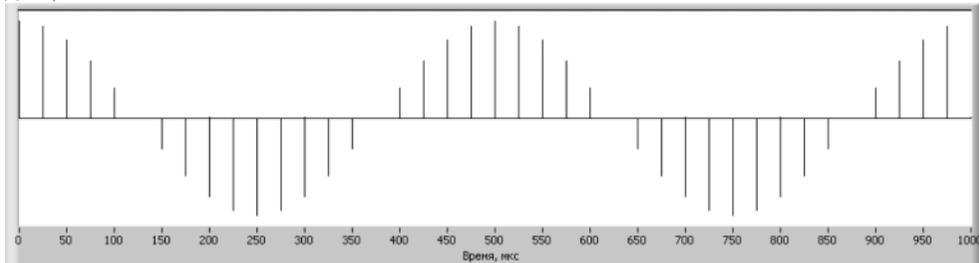
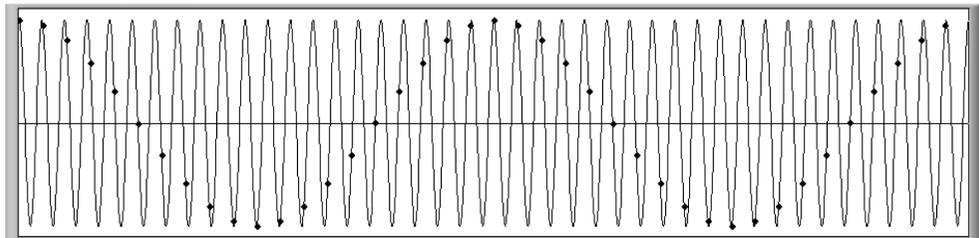


Рис. 1.3. Аналоговый гармонический и дискретный сигналы: частота сигнала  $f_0 = 2$  кГц, частота дискретизации  $f_s = 40$  кГц

Аналоговый сигнал



Дискретный сигнал

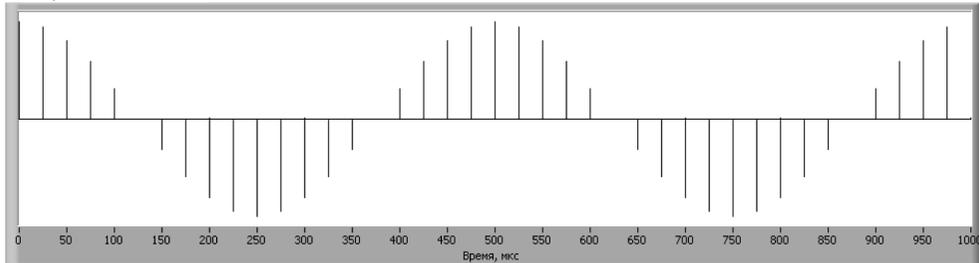


Рис. 1.4. Аналоговый гармонический и дискретный сигналы: частота сигнала  $f_0 = 42$  кГц, частота дискретизации  $f_s = 40$  кГц

- $f_0 = 82$  кГц ( $2 = |82 - 40*2|$ );
- $f_0 = 118$  кГц ( $2 = |118 - 40*3|$ );
- $f_0 = 122$  кГц ( $2 = |122 - 40*3|$ ) и т. д.

Если при частоте дискретизации  $f_s = 40$  кГц аналоговый сигнал будет состоять из гармонических составляющих, например, с частотами  $f_{01} = 3$  кГц,  $f_{02} = 28$  кГц,  $f_{03} = 36$  кГц,  $f_{04} = 45$  кГц,  $f_{05} = 79$  кГц, то дискретный сигнал будет иметь составляющие с частотами:

- $f_0 = 3$  кГц =  $|3 - 40*0|$  – «правильная» составляющая;
- $f_{a2} = 12$  кГц =  $|28 - 40*1|$  – «ложная» составляющая;
- $f_{a3} = 4$  кГц =  $|36 - 40*1|$  – «ложная» составляющая;
- $f_{a4} = 5$  кГц =  $|45 - 40*1|$  – «ложная» составляющая;
- $f_{a5} = 1$  кГц =  $|79 - 40*2|$  – «ложная» составляющая.

Несложно сделать вывод, что при частоте дискретизации  $f_s = 40$  кГц составляющие аналогового сигнала, преобразующиеся в «правильные» составляющие дискретного сигнала, должны иметь частоту меньше 20 кГц, а все составляющие аналогового сигнала с частотами больше 20 кГц преобразуются в «ложные» составляющие.

Эта граничная частота (20 кГц в рассмотренном примере) носит специальное название «частота Найквиста» (Nyquist):

$$f_N = \frac{f_s}{2}.$$

**Примечание.** Частоты «ложных» составляющих также не превышают частоту  $f_N$ .

Так как в дискретном сигнале невозможно различить «правильные» и «ложные» составляющие, то для получения дискретного сигнала, правильно описывающего свойства исходного аналогового сигнала, необходимо выполнение условия

$$f_{0i} < f_N,$$

где  $f_{0i}$  – частоты гармонических составляющих исходного аналогового сигнала.

**Увеличение частоты дискретизации для исключения «ложных» составляющих.** Согласно теореме отсчетов (теореме Котельникова или теореме Шеннона – Shannon Sampling Theorem), чтобы избежать появления «ложных» составляющих, необходимо использовать частоту дискретизации  $f_s$ , как минимум, в 2 раза большую, чем  $f_{0\max}$  – максимальная частота составляющей исходного сигнала:

$$f_s \geq 2f_{0\max}.$$

Важно заметить, что, хотя теоретически достаточно выполнения условия  $f_s = 2f_{0\max}$ , на практике всегда используется  $f_s > 2f_{0\max}$ , так как при этом существенно упрощается ФНЧ в устройстве ЦОС, восстанавливающий аналоговый сигнал  $s_{\text{вых}}(t)$  из дискретного сигнала.

**Фильтры для исключения сглаживания.** Даже при частоте дискретизации, равной  $2f_{0\max}$ , на вход устройства ЦОС вместе с аналоговым сигналом могут воздействовать помехи, например, от мощных линий электропередач или местных радиостанций, которые могут содержать составляющие с частотами выше частоты Найквиста.

ты Найквиста. Такие составляющие приведут к появлению «ложных» составляющих и исказят полезный сигнал. Поэтому из входного аналогового сигнала необходимо удалять составляющие с частотами выше частоты Найквиста прежде, чем этот сигнал поступит на вход АЦП.

Для этого используют **аналоговый** фильтр нижних частот (ФНЧ), включенный перед АЦП (Anti-alias Filter). Этот ФНЧ должен пропускать составляющие с частотами ниже частоты Найквиста и не пропускать составляющие с частотами выше ее. Тем самым будет предотвращено появление в дискретном сигнале «ложных» составляющих.

Аналоговый ФНЧ должен не исказить полезный аналоговый сигнал. Для этого он должен иметь плоскую амплитудно-частотную характеристику (АЧХ) в полосе частот ( $0 \dots f_N$ ) (необходимое, но не достаточное условие). В то же время для составляющих с частотами  $f > f_N$  этот ФНЧ должен иметь коэффициент передачи, равный нулю. Такой ФНЧ называется идеальным, его АЧХ показана на рис. 1.5, а (Ideal Anti-alias Filter).

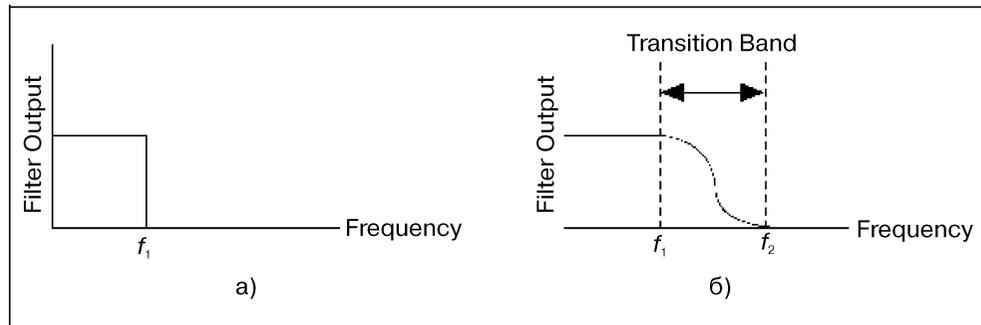


Рис. 1.5. Идеальный (а) и реальный (б) ФНЧ:

$f_1$  – максимальная частота полезной составляющей входного сигнала; составляющие входного сигнала с частотами  $f > f_1$  – помехи;  $f_1 \dots f_2$  – переходная полоса (Transition Band) реального ФНЧ (Practical Anti-alias Filter)

Идеальный ФНЧ наилучшим образом предотвращает появление «ложных» составляющих, однако он физически не осуществим. Для реальных ФНЧ характерна некоторая неравномерность АЧХ в полосе пропускания  $0 < f < f_1$ , наличие переходной полосы  $f_1 < f < f_2$ , в которой коэффициент передачи фильтра с ростом частоты плавно уменьшается до необходимо малого значения, и полосы подавления  $f_2 < f < \infty$ , для которой коэффициент передачи фильтра достаточно мал (см. рис. 1.5).

Поскольку в переходной полосе ФНЧ не полностью подавляет составляющие помех, то они будут создавать в дискретном сигнале «ложные» составляющие. Для исключения этого частоту дискретизации следует задавать исходя не из условия  $f_s \geq 2f_1$ , а из условия  $f_s \geq 2f_2$ . Это приводит к некоторому повышению требований к узлам устройства ЦОС по быстродействию, но обеспечивает необходимую точность обработки аналогового сигнала.

**Примечание.** Требования к ФНЧ определяются, в основном, наличием на входе устройства ЦОС смеси аналогового сигнала и помех, частотным составом помех и их интенсивностью. Если помехи отсутствуют, то ФНЧ перед АЦП не нужен.

**Использование логарифмических единиц.** На индикаторах приборов значение некоторой величины обычно отображают или в линейном, или в логарифмическом масштабе.

Линейный масштаб наиболее привычен для восприятия, так как широко используется для отображения графиков зависимостей в школьных курсах физики, математики, в быту.

Логарифмический масштаб используется реже, в основном тогда, когда нужно на одном графике одинаково подробно отобразить величину как в поддиапазоне малых, так и в поддиапазоне больших ее значений. В логарифмическом масштабе шкалу индикатора можно проградуировать непосредственно в значениях самой величины. В этом случае в начале шкалы будет обязательно ненулевое значение, а сама шкала будет неравномерная – сгущающаяся к концу. Иногда оказывается удобным индексировать не значение величины, а его отношение к некоторому значению, выбранному в качестве опорного. Тогда удобно использовать логарифмическую шкалу, проградуированную в децибелах (дБ), которая является равномерной.

Децибел – универсальная единица выражения отношений уровней величин, не зависящие от того, какими параметрами являются эти уровни: амплитудами, среднеквадратическими значениями (СКЗ) или мощностями.

Следующие формулы предназначены для определения отношений уровней величин в децибелах:

- для отношения мощностей величин

$$D = 10 \log_{10} \frac{P}{P_0};$$

- для отношения амплитуд или СКЗ величин

$$D = 20 \log_{10} \frac{A}{A_0}.$$

Значения мощности  $P_0$  и амплитуды или СКЗ  $A_0$  иногда называют опорными значениями, по сравнению с которыми определяются отношения уровней.

В ряде случаев за опорные значения принимают константы, значения которых определяются областью использования.

Например, в радиотехнике, акустике часто уровень величины выражается в отношениях к следующим константам:

- мощность 1 мВт на сопротивлении 50 Ом для радиосигналов (СКЗ = = 0,224 В) – уровень величины выражается в децибел-милливаттах (дБ·м);
- мощность 1 мВт на сопротивлении 600 Ом для аудиосигналов (СКЗ = = 0,775 В) – уровень величины выражается в децибел-милливаттах (дБ·м);

- дисперсия  $1 \text{ В}^2$  – уровень величины выражается в децибел-вольтах СКО ( $\text{дБ} \cdot \text{В}_{\text{rms}}$ );
- амплитуда  $1 \text{ В}$  – уровень величины выражается в децибел-вольтах ( $\text{дБ} \cdot \text{В}$ );

В табл. 1.1 приведены часто используемые отношения уровней величин в децибелах, отношения мощностей и отношения амплитуд (СКО).

Таблица 1.1 показывает, как можно сжать широкий диапазон амплитуд в небольшой диапазон чисел, используя логарифмический масштаб в децибелах.

Таблица 1.1. Децибелы и отношения мощностей и напряжений

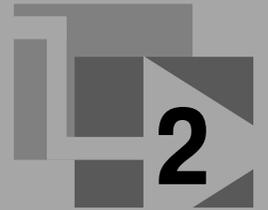
дБ	Отношение мощностей	Отношение напряжений
+60	1 000 000	1 000
+40	10 000	100
+30	1 000	31,6
+20	100	10
+10	10	3,16
+6	4	2
+3	2	1,414
+2	1,585	1,259
+1	1,259	1,122
0	1	1
-1	0,794	0,891
-2	0,631	0,794
-3	0,5	0,707
-6	0,25	0,5
-10	0,1	0,316
-20	0,01	0,1
-30	0,001	0,0316
-40	0,0001	0,01
-60	0,00001	0,001

**Пример.** Для приближенного, но быстрого перевода уровня, выраженного в децибелах, в отношение, например, напряжений можно пользоваться следующим алгоритмом. Для упрощения необходимо помнить, что величина в децибелах может быть представлена в виде суммы простых слагаемых, которым соответствуют известные отношения напряжений. После перевода слагаемых в отношения их необходимо перемножить для получения результата преобразования. Например, необходимо перевести 16 дБ в отношение напряжений. Представим число 16 в виде слагаемых  $6 + 6 + 3 + 1$ . Этим слагаемым соответствуют отношения напряжений: 2, 2, 1,41 и 1,1. После их перемножения получим приблизительно 6,2.

Для получения того же уровня 16 дБ в отношение по мощности: те же слагаемые в децибелах соответствуют множителям: 4, 4, 2, 1,21 (см. табл. 1.1). После перемножения получим приблизительно 38, что соответствует отношению напряжений 6,2, возведённому в квадрат.

Необходимо заметить, что значениям уровня в децибелах ниже 1 дБ соответствуют почти линейные шкалы отношений по мощности и по напряжению, что упрощает преобразование малых уровней, выраженных в децибелах, в отношение напряжений или мощностей.

## Цифровая связь и LabVIEW



Большинство людей представляют себе цифровую связь как нечто имеющее отношение к компьютерам, обеспечивающим разговор с абонентом. Однако они не понимают, что на самом деле цифровая связь заключается в посылке цифровых данных через некоторую среду, которая намного более подходит для аналоговых сигналов. Окружающий мир выражает себя в аналоговой форме, в то же время цифровая связь реально обеспечивает передачу совокупности дискретизированных форм аналоговых колебаний. Возможно, цифровая связь должна быть переименована в дискретную связь. Можно сказать, что каждая дискретная форма колебаний имеет связанное с ней цифровое представление. Так, в данном пособии под цифровой связью подразумевается любая система, где цифровые данные передаются из одного места в другое путем использования некоторого конечного набора сигналов. Цифровые системы связи имеются всюду. Мобильные телефоны, жесткие диски, модемы цифровой абонентской линии, спутниковое телевидение, проигрыватели компакт-дисков – все это примеры систем связи, в которых передаются цифровые данные. Для многих различных приложений связи, которые существуют, имеется много равномерных цифровых протоколов связи: GSM, множественный доступ с кодовым разделением каналов (CDMA), OFDM, 802.11 (b и g), Ethernet, APCO-25, также появляются протоколы типов EDGE и W-CDMA (множественного доступа с кодовым разделением каналов).

Беспроводные цифровые стандарты связи приведены в табл. 2.1, где представлены только цифровые системы связи, используемые сегодня, и каждая из них имеет свою собственную отдельную полосу частот, которая обеспечивает формат передачи данных и метод коллективного доступа. Эти системы становятся настолько распространенными, что каждая компания, вовлеченная в связь, должна продать изделие, которое поддерживает Bluetooth, 802.11, Инфракрасный канал, ИИЭР 1394, USB и т. д. Кажется, будто есть совпадения в структурах устройств связи, но имеются расхождения в протоколах связи. Испытание и развитие систем, соответствующих всем этим стандартам, являются дорогим и неизбежным

занятием, однако трудно покупать оборудование, которое может поддержать связь на высоком уровне. Так что же можно сделать, чтобы не отставать от быстро развивающихся технологий в цифровой связи? Лучшее моделирование и макетирование могли бы быть одним из решений этой проблемы, но как реально проверить целую систему?

Многие инженеры обращаются к пакету MATLAB или, возможно, даже к программированию в языке C, чтобы моделировать свои проекты связи перед началом производства. И есть для этого превосходные и самые мощные инструментальные средства. Однако при сочетании пакета программ и аппаратных средств программная среда LabVIEW действительно затмевает и то и другое. С общей точки зрения графическая среда LabVIEW проста для сборки и понимания. Посмотрев немного глубже, можно увидеть, что LabVIEW имеет мощный встроенный комплект инструментальных средств обработки сигналов, не требует кодирования типа распределения памяти или описания переменных, не использует никакого компилирования, имеет высокоинтегрированное инструментальное управление и превосходные утилиты отображения для изучения цифровых сигналов в различных точках системы связи, а также обеспечивает сбор цифровых данных. Можно видеть, как простые виртуальные приборы (ВП) LabVIEW могут быть сформированы и объединены, чтобы произвести гибкую и мощную цифровую систему тестирования системы связи. В результате формирования такой основанной на LabVIEW системы связи будут учтены новые стандарты интерфейса, которые будут быстро и легко объединены с программной средой. Можно также заметить, что всего за несколько лет пакет LabVIEW успел пройти длинный путь развития, что сказалось в повышении скорости обработки сигналов, но он ни в коем случае не является средой для обработки в реальном масштабе времени. При разработке частей цифровой системы связи важно понять, что эта система действительно слишком медленна для связи в реальном масштабе времени, но достаточно хороша для измерений инструментального типа. Формирование системы связи в LabVIEW сопровождается потерей в скорости обработки данных в обмен на гибкость, легкость кодирования и превосходные возможности интерфейса пользователя и дисплея данных.

Пакет LabVIEW всегда был одним из средств для инструментального управления и сбора цифровых данных или представления их на дисплее, и в настоящем пособии будет показано, как LabVIEW приспособлен для обработки любого сигнала, эффективно модулированного в цифровой форме, и как он может непосредственно использоваться в качестве испытательного прибора. Можно будет увидеть, что размывается граница между инструментальным и виртуальным приборами. Для иллюстрации перечисленных преимуществ узлы цифровой системы связи будут исследованы в среде LabVIEW и транслированы в цифровой приемник. Конечно, инструментальные средства LabVIEW, которые представлены в этом пособии, ни в коем случае не работают в реальном времени. Однако благодаря персональному компьютеру, работающему под управлением Windows, в ближайшем будущем появится много хороших средств обработки сигналов, еще более совершенных вследствие использования LabVIEW.

В табл. 2.1 использованы следующие наименования: Communication protocol – протокол системы связи; Frequency band (s) – диапазон(ы) частот; Channel bandwidth – полоса канала; Modulation – тип модуляции.

Таблица 2.1. Беспроводные цифровые стандарты связи

Communication protocol	Frequency band(s)	Channel bandwidth	Modulation
GSM	800/900 MHz	200 kHz	GMSK
CDMA	800/900 MHz and 1.9 GHz	1,25 MHz	QPSK
Digital cellular	800/PCS	30 kHz	p/4-DQPSK
TETRA	400/800 MHz	25 kHz	p/4-QPSK
802.11b	2,4 GHz	22 MHz	CCK
802.11g	2,4 GHz	22 MHz	OFDM
Bluetooth	2,4 GHz	1 MHz	GFSK
APCO-25	VHF, UHF, 800 MHz	12,5/25 kHz	C4FM
GPS	1,2 GHz и 1.5 GHz	20 MHz	Binary biphasе
CDMA2000	1,9 GHz	1,25 MHz	4-PSK
UMTS	900 MHz и 1,9 GHz	1,25 MHz	GMSK/8-PSK
EDGE	900 MHz и 1,9 GHz	200 kHz	GMSK/8-PSK

Итак, можно жертвовать скоростью автономного специализированного прибора, чтобы получить гибкость детектирования любого сигнала. Дело в том, что эта цифровая система связи больше соответствует системе отладки принципов, чем радиосвязи. Можно будет моделировать, проверять и измерять критические аспекты реальной системы связи. В различных разделах пособия будут представлены некоторые способы оптимизации скорости обработки в LabVIEW, а в разделе 9 будут предложены некоторые полезные подсказки для того, чтобы улучшить производительность системы связи.

Сначала необходимо исследовать, как LabVIEW можно использовать для того, чтобы получать цифровые сигналы связи. Часть этого пособия посвящена выполнению двух различных цифровых структур связи. Первая – обычный цифровой приемник, рассматриваемый в подразделе 2.1. В подразделе 2.2 представлен второй тип, который является полностью цифровым приемником. Оба приемника в настоящее время используются на практике, и оба имеют как положительные, так и отрицательные качества. В последующих разделах будут выделены преимущества и недостатки каждого типа цифрового приемника, что подскажет читателю, как лучше использовать LabVIEW в этих структурах.

Перед рассмотрением специфики каждой из систем цифровой связи необходимо определить, как частота в цифровой части системы соотносится к частоте в аналоговой ее части. Частота в цифровой части системы обозначается как относительная частота  $\Omega$ , выраженная в радианах:

$$\Omega = 2\pi \frac{f}{f_s}, \quad (2.1)$$

где  $f$  – текущая частота в аналоговой системе,  $f_s$  – частота дискретизации.

Так как цифровые выборки являются только последовательностью чисел, они не содержат никакой свойственной сигналам информации о времени. Поэтому, определяя период дискретизации (время между последовательными единичными выборками), можно связать частоту  $f$  с частотой  $\Omega$ , модуль которой есть число радиан на единичную выборку. Поэтому в последующих обозначениях под  $\Omega$  будет подразумеваться произвольная частота, которая используется в цифровой области, под  $f$  будет подразумеваться частота в аналоговой области.

## 2.1. Обычный цифровой приемник

На рис. 2.1 представлена функциональная схема обычного цифрового приемника. В действительности этот приемник является только частично цифровым, потому что во входной его части есть аналоговый узел обработки. Прежде чем нужный сигнал может перейти в цифровую область, он обрабатывается в ключевом устройстве, что обеспечивается аналого-цифровым (А/Д) преобразователем (АЦП). Стандартный АЦП может производить до 20 мегавыборок в секунду (Msps), а это означает, что он имеет входной фильтр (не показанный на рисунке) с граничной частотой 10 МГц. Если нужная радиочастота (RF) сигнала ниже 10 МГц, то АЦП не будет передавать сигнал к цифровому узлу обработки. По этой причине необходимо гетеродинирование. Входной радиосигнал сначала смешивается с сигналом местного гетеродина (LO – Local Oscillator), чтобы сдвинуть несущую частоту из некоторого высокого диапазона вниз, в диапазон промежуточной частоты (IF – Intermediate Frequency), более подходящей для учета ограничений в АЦП. Этот IF-сигнал затем проходит через аналоговый IF-фильтр перед преобразованием в цифровую форму, чтобы удалить любую нежелательную энергию сигнала и таким образом улучшить чувствительность приемника.

Так как считается, что смеситель имеет способ устранять зеркальную боковую полосу частот, в этом типе приемника сохраняется отношение сигнал/шум (ОСШ) исходного сигнала [1].

Это свойство может быть необходимо для приложений, где ОСШ изначально низко.

Обычный цифровой приемник, описанный выше, хотя и широко используется, становится громоздким, если осуществить его в программной среде типа LabVIEW. Преобразование с понижением частоты радиосигнала должно быть сделано в аппаратных средствах с аналоговым смесителем. Сигнал LO обычно генерируется некоторым управляемым контроллером – генератором напряжения и усиливается, чтобы обеспечить соответствующий уровень сигнала для смесителя. Это требует внешних аппаратных средств в дополнение к плате дискретизации. Также любые примеси в сигнале LO немедленно влияют на качество выходного сигнала смесителя. Дополнительные требования к стабильности LO и диапазону его настройки накладывают большие ограничения на полосу сигнала и выбор несущей частоты, которые могут быть изучены в этом типе приемника. Однако эти типы приемников доступны. Фактически компания National Instruments (NI), так же как другие компании, предлагает устройства для преобразования «вниз» час-

тоты радиосигналов, которые обеспечивают законченное решение типа «черного ящика» в LO при смещении радиосигнала по частоте и фильтрации на промежуточной частоте.

Стоит отметить, что прибор PXI-5660 является в некотором смысле гибридным устройством: в нем используются для получения нужного сигнала операции и обычного цифрового приемника, и приемника подвыборки. Детали устройства PXI-5660 будут объясняться позже более подробно. Однако с точки зрения непосредственного пользователя устройства серии 5660 могут показаться обычными цифровыми приемниками, которые также будут рассмотрены в этом пособии.

## 2.2. Приемник подвыборки

В то время как обычный цифровой приемник подходит для того, чтобы выполнять каждую операцию обработки сигналов, представленную в этом пособии, можно изменить его структуру на рис. 2.1, чтобы исключить большую часть аппаратных средств. Это возможно сделать, подключая АЦП к высокочастотной стороне LO-смесителя, а также за счет дискретизации радиосигнала непосредственно на частоте приема, создавая тем самым полностью цифровой приемник.

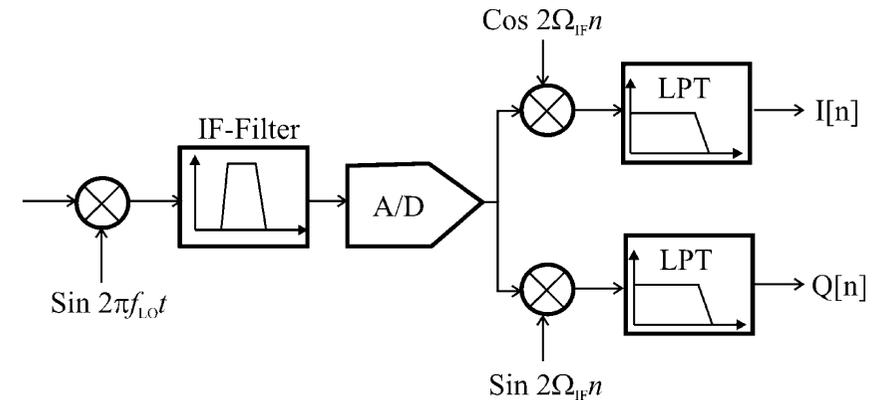


Рис. 2.1. Обычный цифровой приемник: LO – местный гетеродин; IF-Filter – фильтр на промежуточной частоте; LPF – ФНЧ;  $I[n]$  и  $Q[n]$  – мнимая и реальная части  $n$ -го комплексного числа

На рис. 2.2 показана структура полностью цифрового приемника. В этом типе системы АЦП должен производить дискретизацию радиосигнала непосредственно, а LO является простым цифровым смесителем. При этом подразумевается, что АЦП имеет полосу пропускания и скорость дискретизации, необходимые для сбора данных о радиосигнале. Как и прежде, если несущая частота радиосигнала достаточно низка, стандартные АЦП могут легко фиксировать отсчеты сигнала. Однако многие современные системы связи работают со спектром в 800 МГц в диапазо-

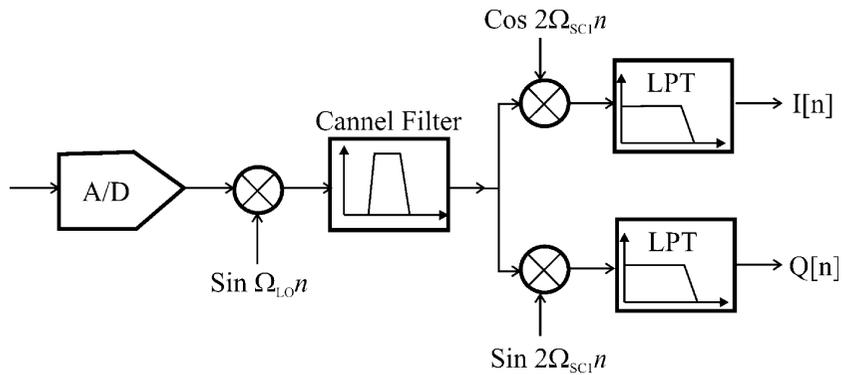


Рис. 2.2. Полностью цифровой приемник

не 1 ГГц. Не нарушая теорему Котельникова, АЦП должен быть способен произвести выборку с частотой вдвое выше наибольшей частоты входного сигнала, в данном примере более чем 2 гигавыборки в секунду (Gsp/s). В то же время есть доступные АЦП с множеством полос пропускания, производящие выборки по частоте в гигавыборках в секунду. Фактически прибор Acqiris выполняет функции цифрового преобразователя, который имеет аналоговую полосу пропускания 1 ГГц и дискретизирует сигналы со скоростью до 2 Gsp/s.

Чтобы дискретизировать сигналы в гигагерцовом диапазоне, требуется огромный объем ячеек памяти для сохранения выборки. При накоплении выборок (если реальные выборки имеют разрядность 8 бит и частоту дискретизации 1 Gsp/s) можно заполнить гигабайт памяти выборками радиосигнала только для фиксации отрезка времени в 1 с. Это не оставляет места в блоке памяти для операционной системы или любых других приложений. Вдобавок ко всему, любая обработка такого огромного числа единичных выборок требует длительных интервалов времени, что является нежелательным для любого псевдоиспытания в реальном масштабе времени. Так как же можно одновременно использовать в своих интересах большую аналоговую полосу пропускания некоторых из этих АЦП, не неся затрат, связанных с обработкой миллиардов выборок в секунду? Для этого можно, например, преднамеренно нарушить теорему дискретизации Найквиста, применив частоту намного меньшую, чем удвоенная частота относительно самой верхней составляющей в сигнале. Эта методика известна как дискретизация подвыборки, полосовая дискретизация или субдискретизация, но есть существенные ограничения на диапазон частот дискретизации, которые дадут желательный результат, не искажая спектра спектральными дублированиями.

Рассмотрим спектр реального сигнала, показанный на рис. 2.3. Для наглядности отображена только положительная половина спектра. Этот сигнал имеет полосу  $B$  спектра в пределах от  $f_L$  до  $f_H$ . Можно видеть, что сигнал центрирован относительно произвольной несущей частоты  $f_c$ . Ясно, что

$$f_L = f_c - \frac{B}{2}; \quad (2.2)$$

$$f_H = f_c + \frac{B}{2}. \quad (2.3)$$

Обычное правило бегунка для дискретизации аналогового сигнала на рис. 2.3 состоит в том, чтобы произвести выборку по частоте, вдвое большей самой высокой частоты, содержащейся в сигнале  $f_H$ . Это соответствует «сценарию», приведенному на рис. 2.4,а. Здесь показаны составляющие с положительными и отрицательными частотами, так же как и первые спектральные дублирования, обусловленные периодичностью дискретизации. Можно видеть, что дискретизация на некоторой частоте  $f_c$ , удвоенной относительно  $f_H$ , обеспечит защитную полосу между спектрами дискретизированного сигнала и половиной частоты дискретизации  $\pi$ . Частота дискретизации понижается до значения  $2f_H$  на рис. 2.4,б, и спектральные дубли перемещаются ближе к половине частоты дискретизации или далеко от множителей  $2\pi$ , не оставляя никаких защитных полос вообще.

На рис. 2.4, в частота дискретизации пересекла пороговое значение и фактически происходит нарушение критерия Найквиста (Nyquist) (стала ниже  $f_H$ ).

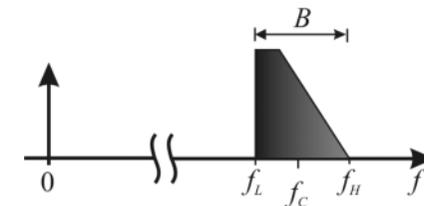


Рис. 2.3. Спектр реального сигнала в аналоговом пространстве

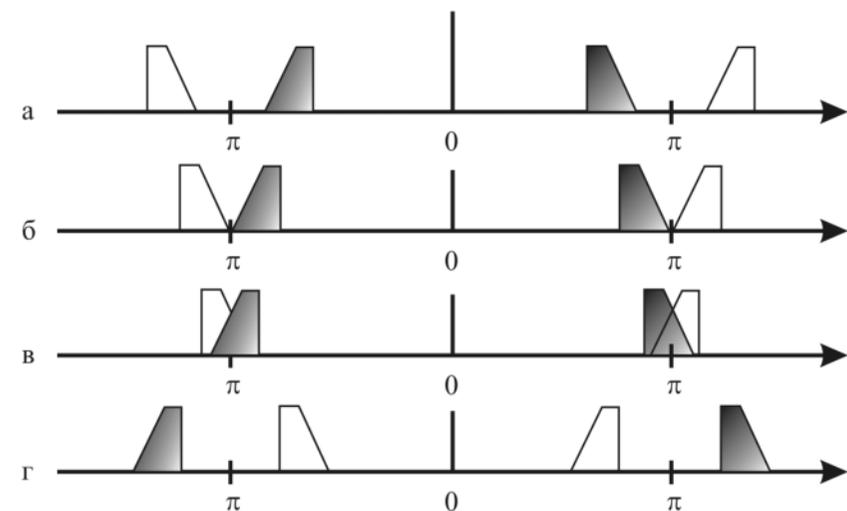


Рис. 2.4. Спектральные преобразования

В этом случае есть диапазоны частот дискретизации, которые приведут к нежелательному наложению частей спектра дискретного сигнала; однако на рис. 2.4, г продолжается снижение частоты дискретизации, и  $f_c$  теперь расположена ниже  $f_H$  без эффекта наложения спектра. Из рис. 2.4, г можно видеть, что спектральные наложения теперь поменялись местами и инвертированы по частоте. Фактически в [2] показано, что ориентация спектральных дублирований переходит от нормальной до инвертированной при каждом целочисленном сокращении частоты дискретизации. Как и при уменьшении частоты дискретизации, спектральные дублирования перемещаются за пределами периода  $2\pi$  и инвертированы относительно друг друга. Диапазон приемлемых частот дискретизации, при которых можно получить выборку сигнала без совмещения спектра, дается в [3]:

$$\frac{2f_H}{n} \leq f_s \leq \frac{2f_L}{n-1}, \quad (2.4)$$

где  $n$  обозначает, в какой степени подвыборка разлагает выборку на множители:

$$1 \leq n \leq I_g \left[ \frac{f_H}{B} \right], \quad (2.5)$$

где  $I_g \left[ \frac{f_H}{B} \right]$  – наибольшее целое число при округлении отношения  $f_H/B$ .

Из выражения (2.4) ясно, что есть много возможностей для того, чтобы определить, сколько можно получить подвыборок сигнала. Одна очевидная возможность заключается в дискретизации на самой низкой возможной частоте, или, иначе, в выборе наиболее возможного  $n$ . Подставляя выражение (2.3) в формулу (2.4), можно определить более низкий предел в этой формуле, чтобы показать абсолютную минимальную частоту дискретизации:

$$f_s \geq \frac{2f_c + B}{n}. \quad (2.6)$$

Используя выражение (2.5), можно определить, что наибольшим значением  $n$  всегда будет  $\left[ \frac{f_H}{B} \right]$ . Снова подставляя  $f_H$  из выражения (2.3), можно получить следующее неравенство:

$$f_s \geq \frac{2f_s + B}{\left( \frac{f_c + B/2}{B} \right)}. \quad (2.7)$$

Теперь можно перестроить знаменатель, вынося множитель  $1/2B$  за скобки:

$$f_s \geq \frac{2f_c + B}{1/2B(2f_c + B)}. \quad (2.8)$$

Наконец, можно переместить множитель  $2B$  в числитель и сократить на  $2f_c + B$  правую часть неравенства (2.8). В результате остается только неравенство

$$f_s \geq 2B, \quad (2.9)$$

из которого следует, что *абсолютно* минимальная частота дискретизации, по которой можно обеспечить выборку сигнала, является удвоенной относительно информационной полосы  $B$ .

Почему бы не дискретизировать аналоговый сигнал на самой возможно низкой частоте дискретизации, заданной в выражении (2.9) как  $2B$ ? Есть несколько соображений для выбора коэффициента  $n$  подвыборки.

Прежде всего, важно понять влияние четных и нечетных разложений на множители сокращений частоты  $f_s$ . Как упомянуто в [2], если значение коэффициента  $n$  является четным, результирующий спектр сигнала будет инвертирован, и спектр будет нормальным, если значение  $n$  нечетное. В зависимости от использования одна или другая ориентация может быть желательной. В случае, когда концы подвыборки сигнала имеют неправильную ориентацию, спектральная инверсия должна быть определена и исправлена просто, но, возможно, при использовании некоторых дополнительных шагов обработки сигнала. Кроме того, выбор  $n$  приведет к частоте дискретизации, при которой не будет наложения спектра. В [1] показано, что IF-частота может быть вычислена по формуле

$$\text{fix} \left( \frac{f_c}{f_s/2} \right) \begin{cases} \text{четный,} & f_{IF} = \text{rem}(f_c f_s); \\ \text{нечетный,} & f_{IF} = f_s - \text{rem}(f_c f_s), \end{cases} \quad (2.10)$$

где  $\text{fix} \left( \frac{f_c}{f_s/2} \right)$  – целочисленная часть отношения  $\frac{f_c}{f_s/2}$ .

Это означает, что проектировщик будет иметь возможность управлять размещением совмещаемых спектров. Третье и, возможно, самое важное соображение для выбора  $n$  – результирующее ухудшение ОСШ, произведенное подвыборкой. Как почти все в разработке, ничто не выполняется бесцельно. При выборе типа дискретизации компромисс заключается в получающемся ОСШ [3]. Субдискретизация подвергается также шум наложения в оттранслированной полосе сигнала; это неизбежно ухудшает ОСШ, но может быть приемлемо для передаваемых сигналов, что и будет рассматриваться далее в настоящем учебном пособии. В [3] ухудшение  $D$  в ОСШ определено приближенным равенством

$$D \approx 10 \log n. \quad (2.11)$$

Из выражения (2.11) очевидно, что при увеличении  $n$  (или уменьшении частоты дискретизации) можно «свернуть» больше шума в полосу пропускания сигнала. В разд. 3, где описаны некоторые аппаратные средства, можно фактически увидеть реальный дискретизированный сигнал, частота дискретизации которого выше частоты Найквиста и который субдискретизирован на частоте, совместимой с выражением (2.2). Можно также увидеть влияние поддискретизации на сигнал. Полный эффект  $D$  от разрушения сигнала будет зависеть от требуемого ОСШ для заданной системы связи.

В связи со всеми этими ограничениями на частоты дискретизации и ухудшение ОСШ субдискретизация кажется намного более сложной, чем дискретизация по Найквисту; однако этот тип дискретизации имеет два замечательных результата. Во-первых, можно чрезвычайно уменьшить требуемую частоту дискретизации и поэтому использовать намного меньший объем памяти, чтобы фиксировать и обрабатывать тот же самый радиосигнал. Во-вторых, если выбирать частоту дискретизации приближенно, совмещенная часть спектра закончится правее относительно промежуточной частоты IF и будет полностью устранена потребность в цифровой операции смещения частоты с помощью местного гетеродина LO (рис. 2.2). Остается единственный вопрос: действительно ли можно согласиться с уменьшенным ОСШ без серьезного воздействия на восстановленный сигнал?

**Пример. Субдискретизация сигнала GSM.** Предполагается, что делается попытка дискретизировать GSM-ячеистый сигнал. Несущая частота – 1 ГГц, и полоса сигнала – 30 кГц. Если взять за основу теорему Найквиста, надо произвести выборку со скоростью более чем 2 Gsp/s, но при субдискретизации можно использовать выражение (2.4), чтобы выбрать намного более низкую частоту дискретизации. Типовое пороговое значение частоты дискретизации все еще определяется теоремой Найквиста; надо помнить, что  $f_c$  должна всегда быть не менее чем вдвое больше информационной полосы сигнала (60 кГц в нашем случае). Стандарт GSM использует интервалы в 9 мс с чередованием запроса 3:1. Если нужно произвести дискретизацию 10 занятых интервалов, длительность сбора данных должна быть равной 270 мс. При скорости 2 Gsp/s это составляет 540 миллионов выборок, но если использовать при субдискретизации необходимую для этого частоту дискретизации, то число выборок может быть уменьшено согласно выражению (2.5):

$$1 \leq n \leq I_g \left[ \frac{1 \text{ ГГц} + 15 \text{ кГц}}{30 \text{ кГц}} \right], \quad \text{или} \quad 1 \leq m \leq 33333. \quad (2.12)$$

Если выбрано число  $n = 10\,000$ , которое означает, что можно получить выборку для SAME-сигнала (subscriber access maintenance entity – объект технического обслуживания абонентского доступа) на интервале в 270 мс, то в результате получится только 54 000 выборок!

## Резюме

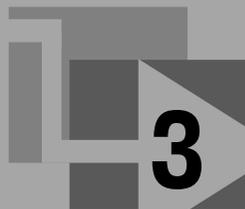
Пакет LabVIEW обеспечивает идеальную среду для того, чтобы моделировать и проверять цифровые системы связи, по нескольким причинам. Во-первых, его графический характер позволяет инженеру быстро проверять компоненты системы без всех непроизводительных затрат, обусловленных использованием обычного кода или системы компиляции. Во-вторых, LabVIEW был задуман для взаимодействия с физическими приборами, и, таким образом, накопление реальных сигналов является обычно прямым и эффективным. В-третьих, можно видеть, что LabVIEW имеет загрузки встроенных инструментальных средств обработки сиг-

налов, которые можно просто включить в ВП и запустить при использовании. И, наконец, можно сделать вывод, что большинство доступных аппаратных средств совместимо с LabVIEW.

Можно видеть также, что с помощью LabVIEW можно использовать обычный цифровой приемник и полностью цифровой приемник. Оба приемника способны к анализу каждого цифрового сигнала, представленного в этой книге. Однако каждый приемник является уникальным и может или не может использоваться для специфических приложений. Также была рассмотрена полноценность приемников субдискретизации. Из-за ограничений на скорость обработки миллионов (или миллиардов) выборок и требуемых объемов и конфигураций памяти субдискретизация является привлекательным способом получения цифрового сигнала связи. Есть некоторые очень определенные ограничения на допустимые частоты дискретизации для этих типов приемников, и есть также стоимость, связанная с субдискретизацией.

Можно будет позже получить по этой теме больше информации, но сначала перейдем к рассмотрению формирования цифрового приемника в разделе 3, выделяя при этом различные аппаратные устройства, чтобы получить необходимый сигнал. Здесь внимание будет сосредоточено на спектральных возможностях анализа LabVIEW. Понятия и инструментальные средства, описанные в этом разделе, будут использоваться в разделе 4 при анализе цифровых фильтров. LabVIEW имеет внушительный набор встроенных подпрограмм фильтрации, и можно группировать некоторые законченные средства проектирования фильтра на основе этих подпрограмм.

## Получение сигнала в LabVIEW



Прежде, чем начать выполнение любого вида цифровой обработки в LabVIEW, необходимо так или иначе получить сигнал, и это есть предмет анализа, представленного в этом разделе. В разделе 2 были выделены два типа цифровых приемников. Первый тип – обычный цифровой приемник, в котором гетеродинируется высокая частота (RF) сигнала «вниз» к частоте, подходящей для большинства АЦП. Второй тип – приемник с прореживанием выборки, для дискретизации сигнала RF непосредственно. Как было упомянуто выше, оба приемника способны к обработке сигналов, что будет исследовано в данном пособии, и оба имеют положительные и отрицательные свойства. В этом разделе сосредоточен материал по использованию каждой из перечисленных структур приемника. В других разделах, приведенных ниже, можно увидеть некоторые методы для проверки приемника на основе использования созданных в пределах LabVIEW испытательных сигналов. Как указано в разделе 1, LabVIEW особенно хорошо адаптирован для анализа физических явлений и устройств. Например, ввод сигналов можно организовать через специализированные приборы по общей шине интерфейса подключений (GPIB). Поскольку целью является формирование структуры приемника, можно использовать неограниченное число входных приборов для ввода физических дискретных сигналов и последующей их обработки. Виртуальные приборы (ВП), описываемые в разделе 3, будут использоваться как стандартные блоки для создания полной цифровой системы связи. Все ВП имеют имена и доступны для загрузки на вебсайте <http://www.MHEngineeringResources.com>. Также приложение 2 содержит полный список, в который внесен каждый ВП наряду с его описанием и блок-диаграммой.

### 3.1. Сигнал в обычном цифровом приемнике

Если рассматривать проект цифрового приемника, то его структура, вероятно, близка к структуре, показанной на рис. 2.1. В определенной полосе RF-сигнал с помощью местного гетеродина (LO) и преобразователя смещается к промежу-

точной частоте (IF) и фильтруется, затем используются два АЦП, которые объединены в одной интегральной схеме (ИС), с выходов которой снимаются квадратурные реальные и мнимые цифровые отсчеты (I и Q) видеосигнала.

Это хорошее решение для заранее определенного приемника, но для гибкого приемника в LabVIEW это несколько сложное решение. Чтобы быстро осуществлять данный приемник, необходимо иметь «черной ящик», в который поступает высокочастотный сигнал RF с некоторой заданной полосой пропускания и который передает цифровые отсчеты прямо в LabVIEW. Можно построить собственный RF-интерфейс, чтобы выполнить фильтрацию и смещение сигнала на промежуточную частоту, подходящую для некоторой имеющейся в наличии входной карты АЦП. Но основной смысл нашего пособия состоит в анализе цифровых возможностей LabVIEW, а не в разработке высокочастотных аппаратных средств ЭВМ.

Корпорация NI действительно производит некоторые модули для выполнения функций, необходимых для этого приемника. На самом деле, можно использовать понижающий преобразователь PXI-5600 RF и цифровой преобразователь PXI-5620 IF. В пакете NI эти два устройства поставляются вместе с анализатором PXI-5660 сигнала RF. Прежде чем продвинуться далее, необходимо дать краткий обзор семейства PXI-устройств. PXI-версия NI компактнее версии PCI. Платы выполнены в виде модулей и включаются в каркас PXI-шасси. Само шасси нуждается или в отдельном контроллере в виде модуля или в комплекте, который позволит автономному PC действовать как контроллер. Корпорация NI предлагает интерфейс для внешнего PC в виде MXI-решения, но оно требует для связи между PC и шасси PXI-прибора дорогого медного или оптического волоконного кабеля.

На рис. 3.1 показана обычная PXI-конфигурация компьютера. Как минимум, PXI-шасси нуждается в контроллере (встроенный PC или MXI-интерфейс PC), электропитании и инструменте для некоторого типа измерений или сбора данных. Чтобы строить обычный цифровой приемник на этой платформе, необходимы понижающий преобразователь PXI-5600 RF и цифровой преобразователь PXI-5620 IF наряду с PXI-шасси отдельно со своим электропитанием. Выбор

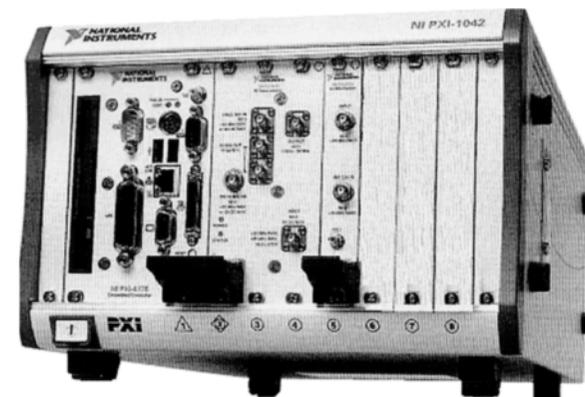


Рис. 3.1. Шасси PXI с модулями

вида контроллера будет зависеть от финансовых возможностей и специфических особенностей его применения.

Теперь, когда исследованы различные варианты для построения обычного цифрового приемника, можно рассмотреть создание рабочей установки для приема и анализа сигналов. Так как контроллер прозрачен для PXI-аппаратных средств ЭВМ, можно игнорировать фактический интерфейс контроллера, выбранного для определенного применения. Что является обычным, так это интерфейс LabVIEW к модулям PXI. При использовании любого из PXI-модульных инструментов может возникнуть необходимость (в дополнение к LabVIEW) в модулях NI Tuner и NI Score. Эти две части программного обеспечения обычно отправляют заказчику со всеми PXI-модулями, и они могут быть загружены с вебсайта NI. National Instruments также снабжает инструментарий спектральных измерений SMT (Spectral Measurements Toolset) пакетом 5660. Этот инструментарий имеет некоторые хорошо изменяющие масштаб изображения операции FFT, спектрограммы и некоторые очень хорошие виртуальные приборы, для того чтобы развивать пользовательские программы, связанные с системами связи. В разделе 6 пакет SMT исследуется более подробно. Если данный пакет инсталлирован и перезагружен в ПК, палитра функции в LabVIEW будет напоминать палитру, представленную на рис. 3.2.

Цифровая панель будет содержать все функции, необходимые для инициализации, и можно использовать дискретизатор PXI-5620. Корпорация NI присоединяет свой стандартный интерфейс ко всем цифровым NI Score, в которых имеются все операции. Панель анализатора высокочастотного сигнала содержит все операции, связанные с понижающим преобразователем PXI-5600 (обычно называемым NI Tuner), и анализатор спектра RFSA (Radio Frequency Signal Analyzer), а также содержит некоторые специальные функции с именами, которые начинаются с SMT. Эти функции – определенные комбинации NI Score и NI Tuner, предназначенные специально для установки аппаратных средств ЭВМ PXI-5660, чтобы ввести высокочастотные сигналы. Все, что выполнено в функциях спектрального анализа SMT, доступных в подпалитрах дискретизатора и RFSA, показано внизу на рис. 3.2.

Для неопытных пользователей LabVIEW или плохо знакомых с PXI-платформой лучше использовать прибор 5660, применяя функции SMT (Spectral Measurements Toolset), поскольку они обеспечивают довольно быструю и легкую разработку средств измерений. Потом, когда будет приобретен навык работы с этим прибором,



Рис. 3.2 Спектральная палитра функций измерения

ром, всегда можно возвращаться и изменять ВП, чтобы использовать более низкие требования к аппаратуре. Вероятно, позже можно обнаружить, что применение этих приборов действительно не требует использования всех особенностей ВП, и можно легко «скроить» другие файлы приборов, чтобы удовлетворить заданным потребностям пользователя.

Простые ВП, показанные на рис. 3.3 – это самые основные программы для получения цифровых данных от анализатора радиосигналов 5660. Такой файл ВП называется PXI Capture.vi и работает хорошо для быстрого получения и показа сигнала на дисплее. Подобно большинству совместимых с LabVIEW инструментальных драйверов, операции PXI могут быть разбиты на три группы: операции инициализации, операции сбора данных и операции способа перемещения. Имеются два блока sub-ВП слева на рис. 3.3 – Init5660.vi и Config5620.vi. Перечисленные операции соответствуют категории инициализации. Эти функции – просто реконфигурации стандартных примеров NI SMT для построения программы сбора цифровых данных.

Открытие ресурсов в LabVIEW выполняется почти так же, как открытие файлов на языке программирования C.

Ресурс инициализируется только однажды (подобно fopen в языке C) и после инициализации переходит к ручному или к ID-вводу. Управления вручную или с помощью ID-ввода *должны* использоваться в любых последующих запросах для доступа к этому инструменту. Это представление похоже на указатель файла в языке C. В петле программы непрерывно выполняется чтение из этого ресурса, и как только число циклов в петле заканчивается, ресурс закрывается. Можно заметить, что параметры инструмента PXI не позволяют пользователю выбирать частоту дискретизации. Далее в пособии можно будет увидеть, почему корпорация NI решила ввести управление частотой дискретизации. Пока можно мириться с фактом, что всегда непременно *известна* частота дискретизации, потому что видно, что один пользовательский вход соответствует длине времени ввода, и можно использовать операцию выбора размера множества, чтобы определить число полученных отсчетов. Следовательно, частота дискретизации – число отдельных отсчетов за секунду. Позже, в разделе 8, будет рассмотрена многоскоростная обработка сигнала, с помощью которой можно будет модифицировать некоторые инструменты для повторного приема данных на другой частоте дискретизации. Построение повторного приема означает, что аппаратные средства ЭВМ дают возможность работать с любой частотой дискретизации, но за счет увеличения объема вычислений в приемнике.

Итак, зачем NI обеспечивает управление частотой дискретизации? Информация NI об изделии PXI-5660 показывает, что производительность по времени устройств 5660 выше, чем у стандартных устройств. Чтобы ускорить готовность к получению выборки от аппаратных средств ЭВМ, в продукции NI присутствует дополнительная система дискретизации (DDC), осуществляющая прямое цифровое преобразование. Не зная определенно, что происходит в пакетах 5660, можно видеть, что выходные отсчеты поступают на частоте дискретизации, которая зависит от информационной полосы частот сигнала и не связана с несущей частотой.

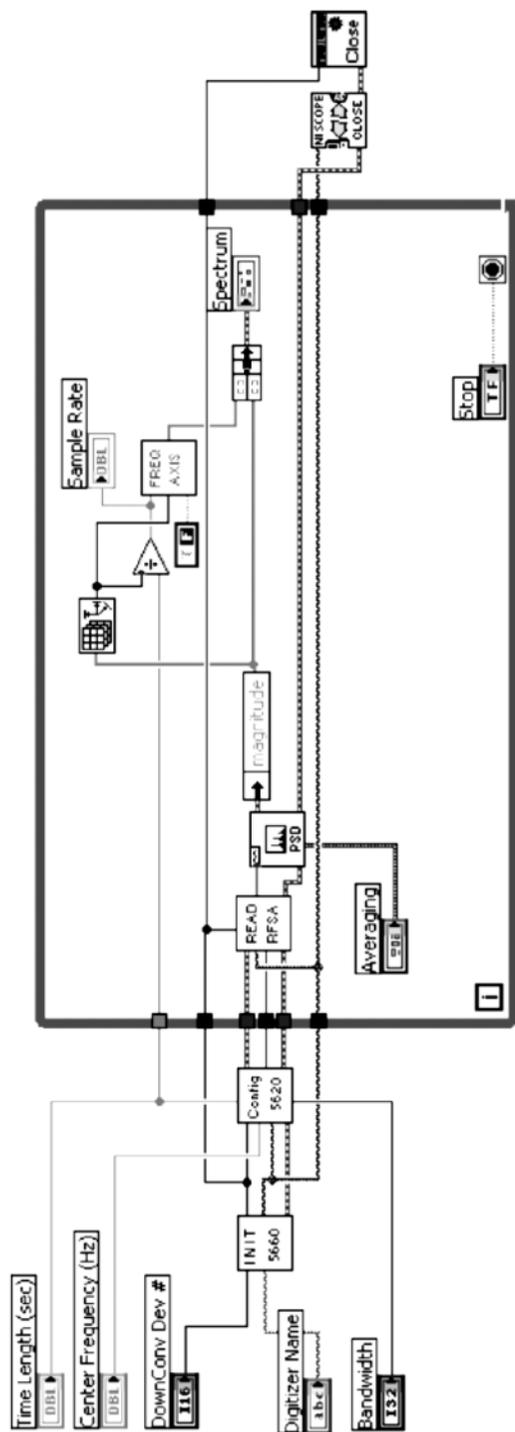


Рис. 3.3. Простой сбор данных в PXI-5660 на основе использования ВП PXI Capture.vi

Возвращаясь к описанному в разд. 2 прореживанию выборки сигнала, можно сказать, что аппаратные средства ЭВМ NI выполняют операции, входящие в выражения (2.2), (2.3) и (2.4). Другими словами, частота дискретизации может быть уменьшена, и дискретный сигнал может быть непосредственно приведен к частоте IF с помощью обработки дискретного сигнала (см. подраздел 3.2) собственными аппаратными средствами дискретизации в ЭВМ. Таким образом, из табл. 3.1 видно, что NI позволяет *ограничивать* управление использованием особенностей АЦП, отрегулировав желательную полосу частот сигнала.

Для получения дополнительной информации относительно ВП, описанных выше, можно обратиться к приложению 2.

Вероятно, для любого пользователя, плохо знакомого с PXI-платформой, лучше всего начать с примеров, предоставляемых с программным обеспечением NI. Для PXI-5660 примеры находятся обычно в папке C:\Program Files\National Instruments\LabVIEW 7.0\examples\Spectral Measurements Toolset\SMT Examples for RFSa.

Рассматривая содержимое этой папки, можно видеть, что некоторые очень полезные программы запускаются вместе с аппаратными средствами. Многие из таких ВП разработаны так, чтобы их можно было использовать в других приборах, и они содержат много элементов, которые не всегда связаны с предназначением этих ВП. Также надо помнить о возможности использования файлов помощи в LabVIEW для любых незнакомых функций.

## 3.2. Сигнал в цифровом приемнике с прореживанием выборки

В разделе 2 упоминался приемник с прореживанием выборки как полностью цифровой приемник. Это название связано с тем фактом, что единственной общей частью аппаратных средств ЭВМ во всей системе является АЦП и что *вся* обработка сигнала выполняется в цифровой форме.

Таблица 3.1. Полоса частот входного сигнала АЦП и наибольшая частота дискретизации

Полоса сигнала	Наивысшая возможная частота дискретизации
1,25 МГц	2 Msps
800 кГц	1 Msps
400 кГц	500 ksps
200 кГц	250 ksps
100 кГц	125 ksps
50 кГц	62 500 sps
25 кГц	31 250 sps
12,5 кГц	15 625 sps

**Примечание.** Единицы измерения частоты дискретизации: Msps – мега-выборки в секунду; ksps – киловыборки в секунду; sps – выборки в секунду.

Поэтому одним из самых важных параметров выбираемого АЦП всегда будет полоса частот входного аналогового сигнала. Если АЦП имеет фильтры с произвольными склонами АЧХ, которые ограничивают его входную полосу пропускания частотой ниже несущей частоты, то невозможно получить сигнал после дискретизатора. Можно подумать, что частота дискретизации АЦП – самая важная его характеристика, однако допустимо выбрать почти любую произвольную частоту дискретизации «вниз» к  $2B$  и все еще иметь точное цифровое представление исходного сигнала. Так, при выборе карты АЦП сначала ищут карту с достаточной аналоговой полосой пропускания, чтобы сохранить фронты сигнала, и затем выбирают такую карту, которая может обеспечить диапазон частот дискретизации.

Удивительно, что большинство карт АЦП имеет полосу пропускания только до 10 или 20 МГц. Это приемлемое ограничение, если несущая частота сигнала достаточно низкая, однако большинство систем беспроводной цифровой связи занимает диапазон от 800 МГц до 1 ГГц. Это делает выбор подходящей карты более жестким, но, конечно, возможным.

Устройство (Acqiris) имеет два входных канала, аналоговую полосу пропускания 1 ГГц и может обеспечить до 2 гигавыборок в секунду (Gsp/s) (отдельно на каждый канал). Карта АЦП поставляется с необходимыми драйверами для LabVIEW на компакт-диске. Установка карты проста, и когда она завершена, палитра функций LabVIEW напоминает палитру, представленную на рис. 3.4.

В дополнение к основным функциям драйвер карты Acqiris также обеспечивается примером ВП, чтобы пользователь мог быстро использовать карту. Примером может служить ВП Acquire.vi, который является простой программой. Блок-диаграмма ВП на рис. 3.5 показывает несколько измененную версию стандарта Acqiris. В этом ВП основные функции Acqiris были выделены в три группы, упомянутые выше, а добавление блока subVI, помещенного в петле программы, предназначено для непрерывного получения выборки сигнала.

Переходя к спектральной плотности мощности на палитре измерений сигнала в ВП Acqiris, можно вычислять спектр сигнала в течение времени записи каждого сигнала и показывать его на передней панели. Кроме того, ось частоты блока subVI добавляется, чтобы определить значения частоты на оси X для заданной частоты дискретизации. Эти две функции могут использоваться, чтобы вывести на дисплей результаты расчетов, как показано на рис. 3.6. Главное здесь то, что данный ВП может быть видоизменен за счет приведенных в палитре функций, в которых есть необходимость.

Далее необходимо внимательнее изучить рис. 3.6. Название ресурса PCI: INSTR0 является специфическим. В этом случае Acqiris-карта является PCI-устройством.

Если есть неуверенность в необходимости использования аппаратных средств ЭВМ, можно использовать National Instruments Measurement & Automation Explorer (MAX), чтобы рассмотреть все смонтированные NI-совместимые аппаратные средства ЭВМ. На рис. 3.5 показано управление для заданного числа отсчетов и частоты дискретизации. Вот только как определить частоту дискретизации? В разделе 1 даны некоторые руководящие принципы для выбора соответствующей

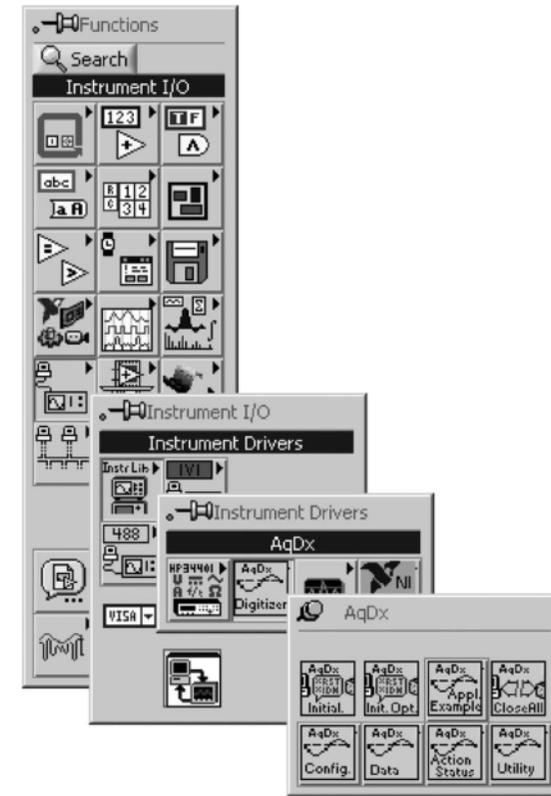


Рис. 3.4. Палитра функций цифрового преобразователя Acqiris

щей частоты прореживания выборки и описано воздействие этой частоты на входной сигнал. Ниже эти вопросы исследуются более подробно.

### 3.2.1. Выбор частоты дискретизации

В отличие от NI PXI-интерфейса функции цифрового преобразователя Acqiris позволяют пользователю определять частоту дискретизации. Это качество имеет особое значение при построении приемника с прореживанием выборки.

Единственное беспокойство для пользователя – какую выбрать частоту дискретизации. Самая низкая частота дискретизации выборки, пригодная к употреблению, задается в выражении (2.5) как  $2B$ , но мы можем также выбирать любую произвольную частоту до предельной для типовой карты, пока частота дискретизации не попадет в одну из запрещенных зон, где воображаемые составляющие приводят к наложению спектров. Эти зоны определены выражениями (2.3) и (2.4) и представлены графически в [3]. В зависимости от вида обработки входной сигнал уже переведен в цифровую форму на определенной частоте дискретизации, однако, может возникнуть потребность в дискретизации данных на другой частоте

те. Вне этих специфических требований обработки почему бы и не перейти на самую низкую частоту?

Есть действительно два главных соображения для выбора хорошей частоты прореживания выборки: ОСШ, разрушение и размещение подвыборки сигнала.

В разделе 2 теоретически представлено влияние обоих методов прореживания выборки на величину ОСШ в выражении (2.8) и на размещение спектра сигнала, определенное выражением (2.7). В последующих подразделах данного раздела будут показаны практические результаты этих преобразований сигналов и рассмотрено их воздействие на восстановленный сигнал.

### 3.2.2. ОСШ при прореживании выборки

Как известно из раздела 1, в обмен на снижение частоты дискретизации обнаруживается потеря в ОСШ. Почему частота дискретизации затрагивает ОСШ? Можно вспомнить, что прореживание выборки приводит фактически к явлению совмещения частей спектра дискретного сигнала.

Любой шум в системе связи приведет также к ложным составляющим в полосе сигнала, а это означает, что шум в полосе является теперь ложным и может занять ту же самую полосу, что и заданный сигнал.

При проведении эксперимента в лабораторных условиях должен быть незначительный шум в полосе приема, так что можно игнорировать ложный шум и рассматривать только шум, появляющийся в процессе работы АЦП. Обычный шум в АЦП включает шум квантования, тепловой шум и колебания положений отсчетов во времени.

В предположении, что все шумовые источники являются постоянными, необходимо определить изменения шумового порога, которые появляются из-за снижения частоты дискретизации.

Авторы Oppenheim и Schaffer [4] показали, что шум квантования может быть принят в широком смысле как постоянный белый шумовой процесс. Такой процесс имеет спектральную плотность мощности, показанную на рис. 3.7, а.

На рис. 3.7, б представлена другая спектральная плотность мощности того же самого белого шумового процесса, но с более низкой частотой дискретизации. В [5] авторы показали, что *полная* шумовая мощность не зависит от частоты дискретизации. Поэтому площади, ограниченные прямоугольниками на рис. 3.7, а и б, постоянны. Для того, чтобы полная шумовая мощность оставалась постоянной и в меньшем диапазоне частот (рис. 3.7, б), спектральная плотность  $A_2$  должна быть больше, чем спектральная плотность  $A_1$ . Наоборот, при увеличении частоты дискретизации, уменьшается спектральная плотность шума, чтобы обеспечить ту же самую полную шумовую мощность на увеличенной полосе пропускания, и, следовательно, это одна из выгод от прореживания выборки. Для приемника прореживания выборки эта увеличенная шумовая амплитуда имеет эффект поднятия шумового порога АЦП. Шумовой порог (NF) в АЦП может быть рассчитан по [5]:

$$NF(\text{дБ}) = 6,02 \times B + 1,8 + 10 \log \left( \frac{f_s}{2} \right), \quad (3.1)$$

где  $B$  – число битов в АЦП и  $f_s$  – частота дискретизации.

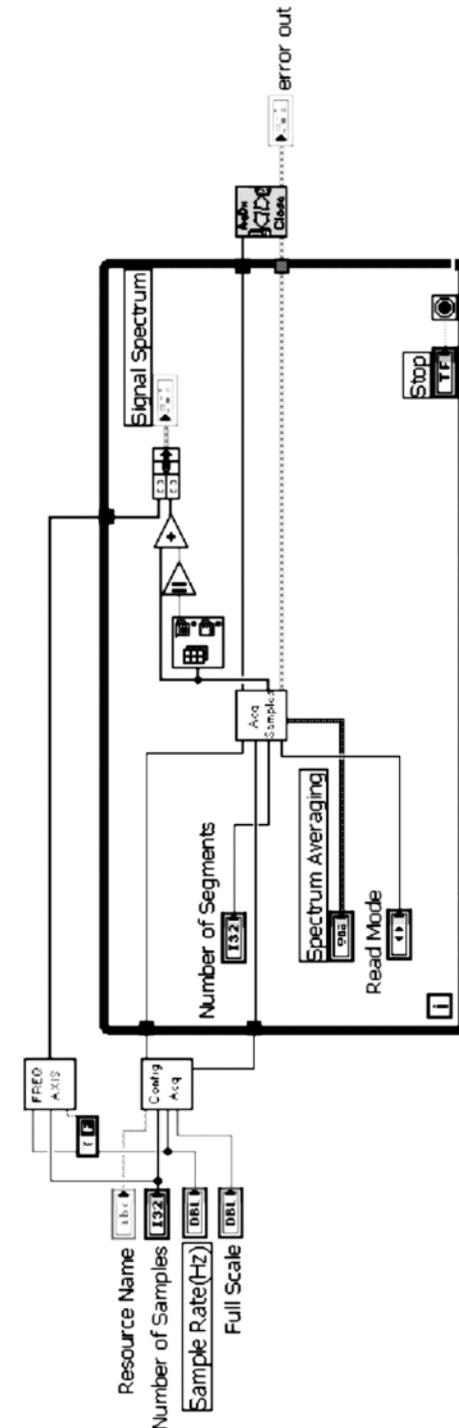


Рис. 3.5. Пример измененного ВП Acqiris

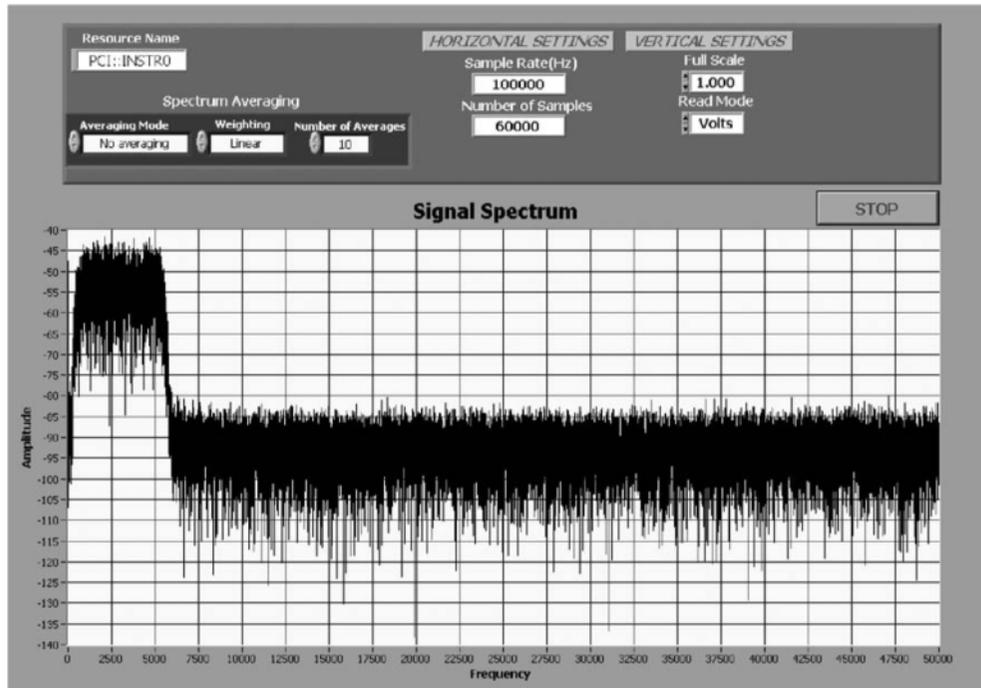


Рис. 3.6. Фронтальная панель для Acqiris

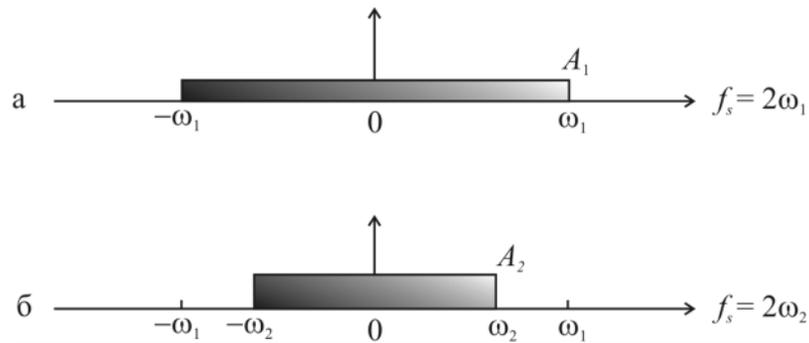


Рис. 3.7. Спектральная плотность мощности шума квантования

Из выражения (3.1) изменения в NF из-за понижения частоты дискретизации от  $f_1$  до  $f_2$  дают

$$\Delta NF(\text{дБ}) = 10 \log\left(\frac{f_1}{2}\right) - 10 \log\left(\frac{f_2}{2}\right) = 10 \log\left(\frac{f_1}{f_2}\right). \quad (3.2)$$

Можно легко увидеть эти изменения в шумовом пороге, исследуя рис. 3.8 и 3.9. На рис. 3.8 представлен спектр реального 16-QAM-сигнала с несущей частотой

$f_c = 99,003$  МГц и полосой пропускания 6 кГц, который дискретизирован со скоростью 200 Msps, превышающей частоту Найквиста. Шумовой порог на этом рисунке равен приблизительно – 70 дБ.

На рис. 3.9 – спектр того же самого 16-QAM-сигнала, прореженного на частоте 100 000 sps. Ясно, что здесь шумовой порог выше, чем на рис. 3.8, и что можно установить новый шумовой порог, приблизительно равный – 36 дБ. На основании выражения (3.2) увеличение шумового порога может быть рассчитано следующим образом:

$$\Delta NF = 10 \log\left(\frac{200000000}{100000}\right) = 33 \text{ дБ}. \quad (3.3)$$

Повышение шумового порога, только что описанного, – минимальная потеря в ОСШ, потому что здесь исключены любые другие шумы типа ложного шума за пределами полосы или острых выбросов в сигнале при аналого-цифровом преобразовании. Эффект потери в ОСШ (или увеличенный шумовой порог) – увеличенная вероятность ошибки в демодулируемом сигнале. Соотношения между ОСШ в символе и вероятностью ошибки символа для M-QAM-сигнала развиты в [6].

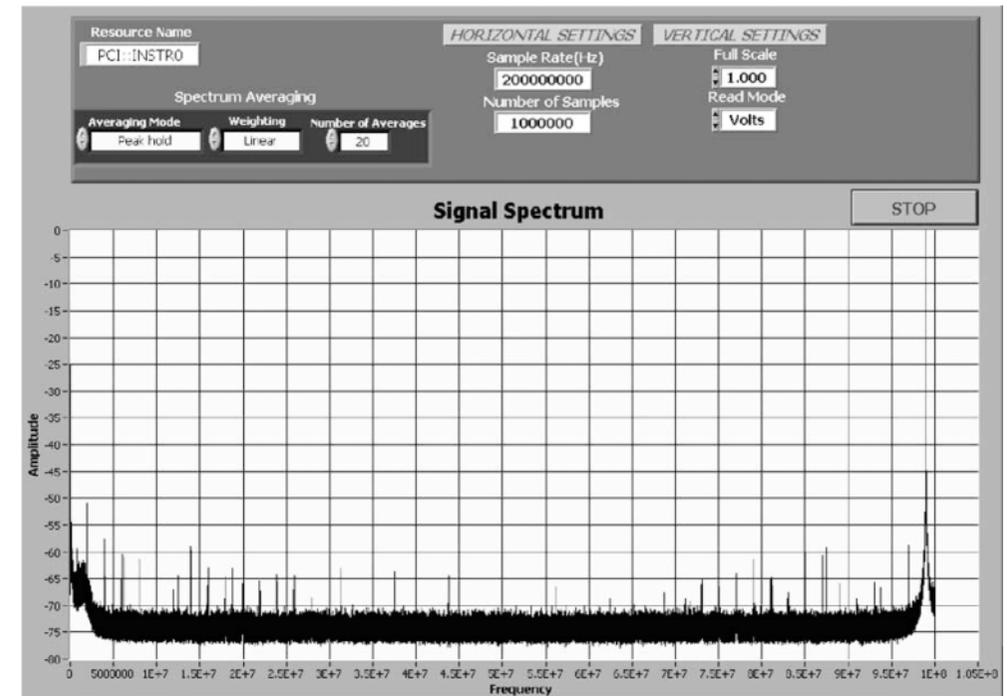


Рис. 3.8. Спектр 16-QAM-сигнала с частотой дискретизации 200 МГц ( $f_c = 99,003$  МГц)

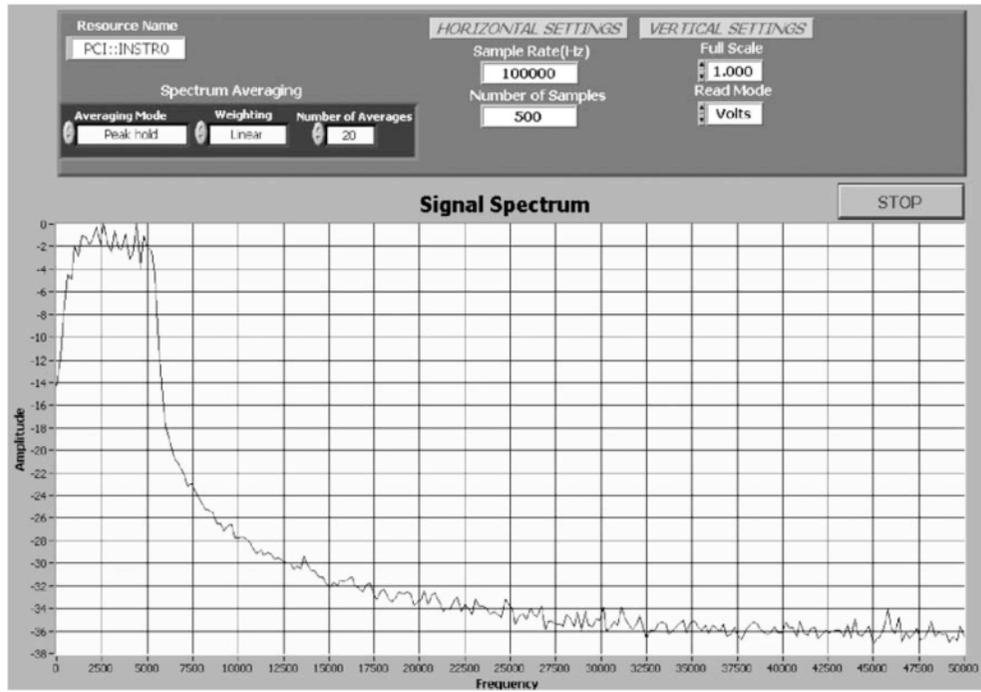


Рис. 3.9. Спектр дискретного 16-QAM-сигнала при использовании прореживания до 100 кГц ( $f_c = 99,003$  МГц)

Для 16-QAM-сигнала вероятность ошибки символа равна

$$P_M = 1 - \left[ 1 - \frac{3}{2} Q \left( \sqrt{\frac{3 E_{AV}}{15 N_0}} \right) \right]^2, \quad (3.4)$$

где отношение  $E_{AV}/N_0$  определено как среднее ОСШ в символе.

Функция  $Q(\cdot)$  известна как дополнительная ошибка функции (3.4) и имеет свойство монотонного увеличения с ростом ее аргумента.

Косвенно это означает, что как только ОСШ в символе (или  $E_{AV}/N_0$ ) увеличивается, то  $1 - Q(\text{ОСШ})$  уменьшается и снижает вероятность ошибки символа. Теперь возникает вопрос: когда используется слишком высокое прореживание сигнала и полностью ли разрушается при этом ОСШ? Один способ узнать это состоит в том, чтобы только контролировать частоту ошибки в битах, передавая известную последовательность битов, и просто складывать ошибки битов в демодуляторе. Другой, лучший, способ состоит в том, чтобы удостовериться, что выбранные модуляция и частота символа могут компенсировать уменьшение ОСШ. В [7] имеется много информации на эту тему: как  $E_{AV}/N_0$  соотносится с полным ОСШ и почему  $E_{AV}/N_0$  (или на самом деле  $E_b/N_0$ ) – хорошая мера качества для цифровой системы связи. Эти соотношения имеют следующий вид:

$$SNR = \frac{E_{AV}}{N_0} \times \frac{R_S}{W} = \frac{E_b}{N_0} \times \frac{R_b}{W}, \quad (3.5)$$

где  $W$  – полоса, занимаемая сигналом;  $R_S$  – частота символа;  $R_b$  – частота битов;  $E_{AV}$  – энергия в символе;  $E_b$  – энергия, содержащаяся в 1 бите;  $N_0$  – шумовая спектральная плотность мощности. Теперь можно использовать выражение (3.4) в обратном смысле, чтобы получить требуемое отношение  $E_{AV}/N_0$  для заданной вероятности ошибки символа, и затем решить уравнение (3.5) для определения необходимого ОСШ при заданной частоте ошибки. Пока можно сохранить это ОСШ, передать данные с заданной частотой и зафиксировать вероятность небольшого значения ошибки ниже требуемого порога. Можно рассмотреть следующий пример, который иллюстрирует отношение между вероятностью  $P_M$  ошибки и ОСШ.

**Пример. Зависимость ОСШ от вероятности ошибки.** Для этого примера можно использовать сигналы, показанные на рис. 3.7 и 3.8. Такую форму имел 16-QAM-сигнал, выданный произвольным генератором формы колебания со следующими параметрами: частота символа  $R_S$  – 4800 символов в секунду, дополнительный коэффициент полосы пропускания – 0,2. Необходимо, чтобы система имела *максимальную* частоту ошибки символа  $10^{-3}$ . На первом шаге следует решить уравнение (3.4), чтобы рассчитать функцию  $Q(\cdot)$  в зависимости от вероятности ошибки символа:

$$Q \left( \sqrt{\frac{3 E_{AV}}{15 N_0}} \right) = \frac{2}{3} (1 - \sqrt{1 - P_M}). \quad (3.6)$$

Теперь можно подставить значение  $P_M$ :

$$Q \left( \sqrt{\frac{3 E_{AV}}{15 N_0}} \right) = \frac{2}{3} (1 - \sqrt{1 - P_M}) = 0,0003334. \quad (3.7)$$

Далее можно использовать таблицу для функции  $Q(\cdot)$  в [7], чтобы найти величину дополнительной ошибки, которая приводит к значению, полученному с помощью выражения (3.7). Можно заметить, что в этой таблице значения представлены только четырехзначными десятичными числами, но для нескольких аргументов функция  $Q(\cdot)$  приводит к тому же результату: 0,0003. Можно легко получить это значение, выбирая *наибольшую* величину аргумента, которое даст 0,0003 при условии, что наибольшее значение аргумента обусловит наибольшее необходимое отношение  $E_{AV}/N_0$  и, следовательно, приведет к самой низкой вероятности ошибки. Так что из таблицы в [7] получается значение 3,48, и теперь можно получить следующее значение для  $E_{AV}/N_0$ :

$$\frac{E_{AV}}{N_0} = (\arg(Q))^2 \times \frac{15}{3} = 60,552. \quad (3.8)$$

Наконец, необходимо разделить полученное ОСШ в символе на отношение частоты символа к полосе сигнала, как показано в выражении (3.5). Однако перед использованием выражения (3.5) надо определить полосу  $W$  сигнала.

Снова можно заглянуть в [7], где дана полезная формула

$$W = \frac{1}{2}(1 + \alpha)R_s. \quad (3.9)$$

Теперь используем  $\alpha = 0,2$  – дополнительную полосу пропускания из выражения (3.8) и подставим выражение (3.9) в формулу (3.5) для  $W$ . Параметр  $R_s$  сокращается, и получается выражение

$$SNR_{req} = \frac{E_{AV}}{N_0} \times \frac{2}{1,2} = 100,92 \approx 20 \text{ дБ.}$$

Этот пример является академическим, в нем использовано много ссылок на описания цифровых систем связи, которые содержат зависимости требуемого ОСШ в символе (или в бите) от вероятности ошибки в битах для некоторых из более общих схем модуляции. Вместо того, чтобы выполнять вычисления для каждой системы или вида модуляции, полезно обратиться к этим зависимостям. В данном примере определено, что 16-QAM-система требует 20 дБ отношения сигнал/шум, чтобы остаться на уровне или ниже одной ошибки на 1000 символов.

На первый взгляд, в этом подразделе представлены два противоположных представления зависимостей между ОСШ и частотой дискретизации. С одной стороны, есть в [4] высказывание, что полная шумовая мощность не зависит от частоты дискретизации. С другой стороны, 16-QAM-сигналу обеспечено повышение шумового порога, и авторы в [3] заявляют, что ОСШ ухудшается в прореженном сигнале. Но если полная шумовая мощность остается постоянной, то *как* изменится ОСШ, когда это есть просто отношение мощности сигнала к шумовой мощности? На рис. 3.10, а показано, что в сигнале с односторонней полосой пропускания  $B$  при передаче его через АЦП выравниваются шумы квантования  $A_1$  и  $A_2$  при соответствующих частотах дискретизации  $2\omega_1$  и  $2\omega_2$ . Для максимизации ОСШ в восстановленном сигнале в большинстве приемников следовало бы после

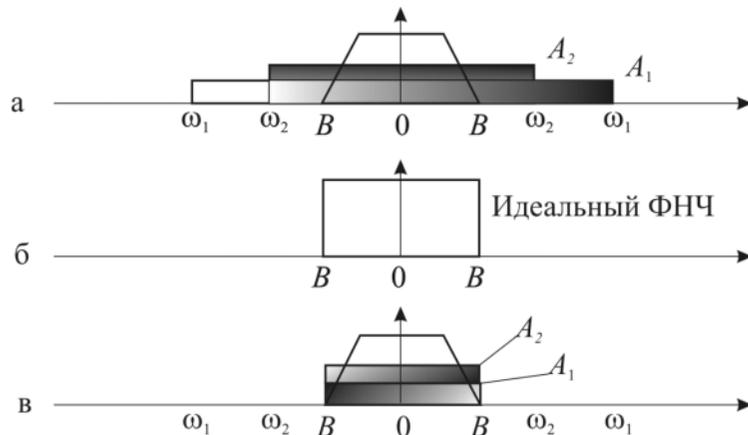


Рис. 3.10. Влияние прореживания на полную шумовую мощность

АЦП использовать ФНЧ, чтобы удалить любой дополнительный шум. На рис. 3.10, б показана идеальная версия такого ФНЧ с граничной частотой  $B$ . Результирующий сигнал и шум квантования после применения идеального ФНЧ показаны на рис. 3.10, в. При частоте дискретизации  $2\omega_1$  полная шумовая мощность получается равной  $A_1 \times 2B$ . И точно так же при частоте дискретизации  $2\omega_2$  полная шумовая мощность равна  $A_2 \times 2\tilde{B}$ .

Так как известно, что  $A_2 > A_1$ , полная шумовая мощность после фильтрации увеличивается, *понижая* частоту дискретизации от  $2\omega_1$  до  $2\omega_2$ . Принимая, что мощность сигнала не изменилась, ОСШ уменьшают (так как увеличилась шумовая мощность), изменяя частоту дискретизации.

### 3.2.3. Прореживание выборки и спектральное размещение сигнала

Другой очень важный аспект прореживания выборки – спектральное размещение прореживания, трансформация спектра. На рис. 3.11 показан вариант изменения масштаба сигнала, представленного на рис. 3.8. Можно заметить, что сигнал сосредоточен вблизи частоты  $f_c = 99,003$  МГц и что частота дискретизации равна 200 МГц. Внимательно посмотрев на подвыборку сигнала на рис. 3.9, можно увидеть, что сигнал теперь переместился к частоте 3000 Гц.

Используя выражение (2.7), получаем четное число

$$\text{fix} \left[ \frac{f_c}{f_s/2} \right] = \text{fix} \left[ \frac{90,003 \text{ МГц}}{100 \text{ кГц}/2} \right] = 1800. \quad (3.10)$$

Тогда новая промежуточная частота,  $f_{IF}$  задается выражением (2.8):

$$f_{IF} = \text{rem}(f_c, f_s) = 3000. \quad (3.11)$$

Можно заметить, что несущая частота была выбрана равной 99,003 МГц, а на самом деле равна 0,003 МГц, что дает снижение до 3000 Гц значения частоты  $f_{IF}$ . В этой конфигурации теперь появляется необходимость смещать центрированную на частоте  $f_{IF}$  форму колебания «вниз» к постоянному току в программном обеспечении. Смещение – дополнительный шаг, но только почему не перемещают несущую частоту точно на 99 МГц и позволяют мнимым составляющим спектра после процесса подвыборки формы колебания смещаться вправо к нулевому значению? Для этого есть определенная причина, что проиллюстрировано на рис. 3.12.

Как показано на рис. 3.12, а, реальный сигнал будет иметь четный симметричный спектр.

Если позволено мнимым составляющим после процесса подвыборки сигнала смещаться к постоянному току, то две половины спектра сигнала накладываются друг на друга (рис. 3.12, б). По этой причине необходимо соответственно выбрать промежуточную частоту и затем выполнить некоторое необходимое окончательное смещение в программном обеспечении.

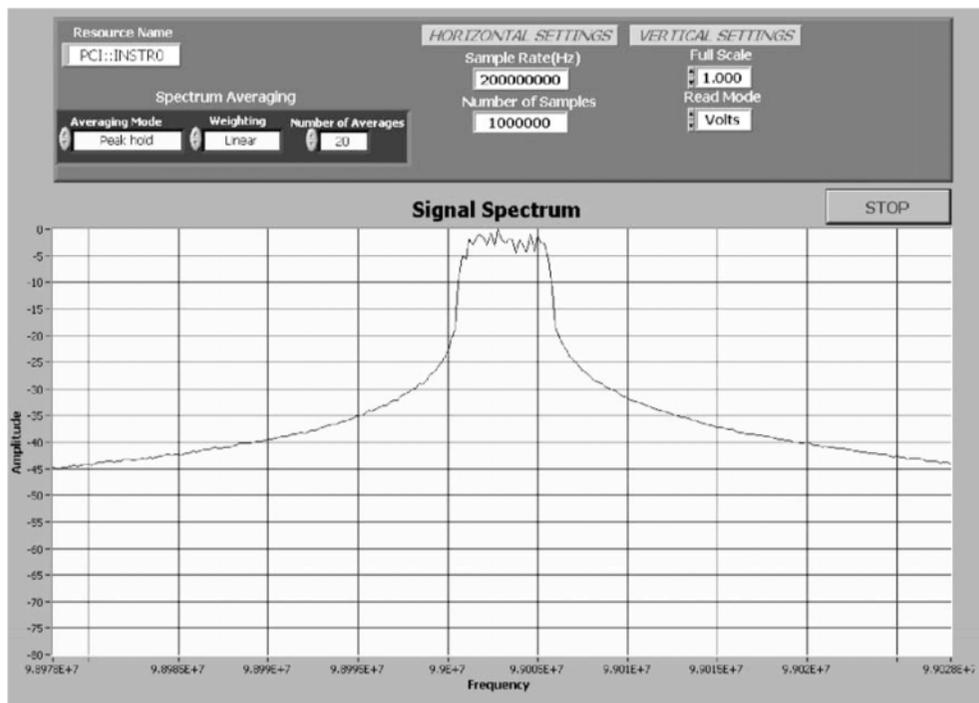


Рис. 3.11. Спектр реального 16-QAM-сигнала, дискретизированного с частотой 200 МГц ( $f_c = 99,003$  МГц)

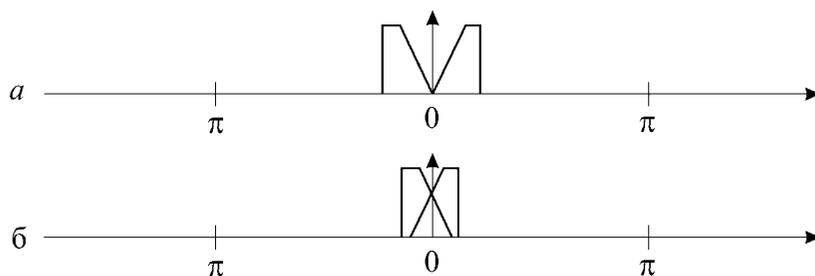


Рис. 3.12. Спектральное размещение совмещенных спектров

### 3.3. Другие методы дискретизации

Два варианта цифрового приемника, описанные в подразделах 3.1 и 3.2, – не единственные устройства передачи сигнала в пакет LabVIEW. Есть также много других методов, но в этом пособии невозможно охватить их все. Ниже приведены некоторые другие методы передачи сигналов, но без подробного объяснения.

### 3.3.1. Цифровой осциллограф

Подготовка отсчетов сигнала для показа на экране является одной из самых простых обработок сигнала. Если имеется в распоряжении исследователя LabVIEW и средства накопления цифровых данных, то это уже половина обработки сигнала для построения цифрового осциллографа. Большие возможности на этом пути открываются, если есть интерфейс GPIB и/или Ethernet. При подключении его к персональному компьютеру остается только обеспечить построение АП, чтобы читать буфер данных из прибора. Главным в этом методе будет ограничение полосы пропускания осциллографа и скорости GPIB или Ethernet для того, чтобы передать накопленные данные.

Есть некоторые специфические особенности инструмента типа формата данных и накопления рекордного их объема, которые необходимо знать. Вероятно, также придется в цифровой форме смещать дискретный сигнал «вниз» к основной полосе частот ВП и иметь некоторые методы преобразования для того, чтобы фактически привести это смещение к величине, которая приводится в разделе 6. Много возможностей имеет единственная опция частоты дискретизации, обычно около 100 Msps. Это означает, что можно обработать *большое* количество данных. Однако есть одно предостережение: этот метод может быть чрезвычайно «медленным» из-за использования GPIB или даже Ethernet. Все-таки это элементарный метод, но он может быть использован так, чтобы работать быстро. Недостаток такого вида обработки заключается в том, что аналоговая полоса пропускания осциллографа обычно составляет 100 или 200 МГц.

### 3.3.2. Анализатор спектра радиосигнала

Анализ спектра является одной из самых распространенных операций, и разработано много приложений для ее выполнения. В отличие от осциллографа большинство анализаторов спектра имеет превосходные частотные характеристики и большие входные полосы пропускания. Конечно, прибор должен будет иметь возможности подключения Ethernet или GPIB для накопления данных. Обычно задается время накопления и вычисления, известна полоса частот сигнала, а значит, и частота дискретизации данных. Такие данные уже находятся в области частот, которые помогут сделать более легкими некоторые операции фильтрации при реализации обработки в ВП. Ключевым пунктом является то, что *пользователь* управляет интервалом накопления и решением, выбирая частоту дискретизации. Другое, что надо помнить, – это то, что при управлении несущей частотой аппаратные средства прибора смещают сигнал к некоторой промежуточной частоте. Может пройти много времени ожидания из-за задержки данных, проходящих по шине GPIB, и много времени обработки, которое требуется для анализатора спектра. Но несмотря на невысокую скорость, это очень легкий способ начать накапливать данные и выполнять некоторую обработку в LabVIEW.

### 3.3.3. Карта дискретизации аналогового сигнала

Корпорация NI (среди многих других изготовителей) производит несколько аналоговых дискретизирующих устройств. Некоторые из них разработаны для подключения в PCI-разъем PC, другие являются внешними коробками с интерфейсами USB. Преимущество большинства этих изделий – легкость, с которой LabVIEW может связаться с ними и с множеством готовых для применения примеров ВП, встроенных в LabVIEW и имеющихся на веб-сайте NI. Однако ни одна из карт NI не имеет аналоговые полосы пропускания более чем 150 МГц, которые строго ограничивают диапазон сигналов. Можно анализировать радиосигналы без некоторых дополнительных аппаратных средств ЭВМ, чтобы перевести некоторый сигнал в диапазон, которому может соответствовать АЦП.

Как показано на рис. 3.13, в LabVIEW целая палитра посвящена установлению связи компьютера с аналоговыми входными устройствами. В зависимости от изготовителя они *могут* обеспечить управление специфическими устройствами, если изготовитель не разработал своих собственных драйверов в LabVIEW. Эти специфические устройства довольно легко осуществить. Есть конфигурация для установки таких параметров, как число отсчетов, число каналов (для многоканальных устройств) и разновидности соединения и размещения буфера для данных. Функции «запустить» и «читать» обеспечивают получение и чтение данных из буфера.

Использование этих функций в большой степени зависит от специфических аппаратных средств ЭВМ. Для получения информации в этом случае необходимо пользовательское руководство и наличие в библиотеке LabVIEW примеров ВП, с которых нужно начать освоение пакета.

### 3.3.4. Звуковая карта

Пакет LabVIEW также имеет способность делать запись данных из звуковой карты, установленной в PC (только в Windows). Есть некоторые ограничения в использовании этого метода получения данных, но с некоторых пор почти все PC имеют звуковые карты, и это осуществимый метод для того, чтобы пробовать обрабатывать реальные сигналы на основе LabVIEW. Нужно иметь в виду, что могут быть ограничения по частоте возможностей аппаратных средств ЭВМ в данном случае, так как звуковая карта будет иметь верхний предел по полосе пропускания и ограниченную частоту дискретизации. Однако LabVIEW позволяет иметь частоту дискретизации до 44,1 кГц с двумя каналами входа и 8 или 16 битами на отсчет.

Конечно, низкая полоса пропускания звуковой карты не будет позволять непосредственно накапливать большинство радиосигналов, но можно экспериментировать с дискретизацией других типов сигналов, используя этот метод.

На рис. 3.14 показано местоположение функций ввода звука в LabVIEW. Как и во всех устройствах дискретизации, есть конфигурация начала чтения и даже функция остановки. Этот метод ввода *может* быть приемлемым, если спектр сиг-

нала находится в основной полосе частот на выходе цифрового обработчика сигналов устройства передатчика.

Блок-диаграмма ВП накопления с выхода звуковой карты показана на рис. 3.15.

Здесь входное управление установки позволяет пользователю выбирать частоту дискретизации, число битов в отсчете, а также накопление моно- или стереосигналов. По умолчанию буфер накопления имеет 8192 байта по длине, хотя это может быть определено пользователем. Как только формируется звуковая карта, петля непрерывно читает из буфера до остановки пользователем.

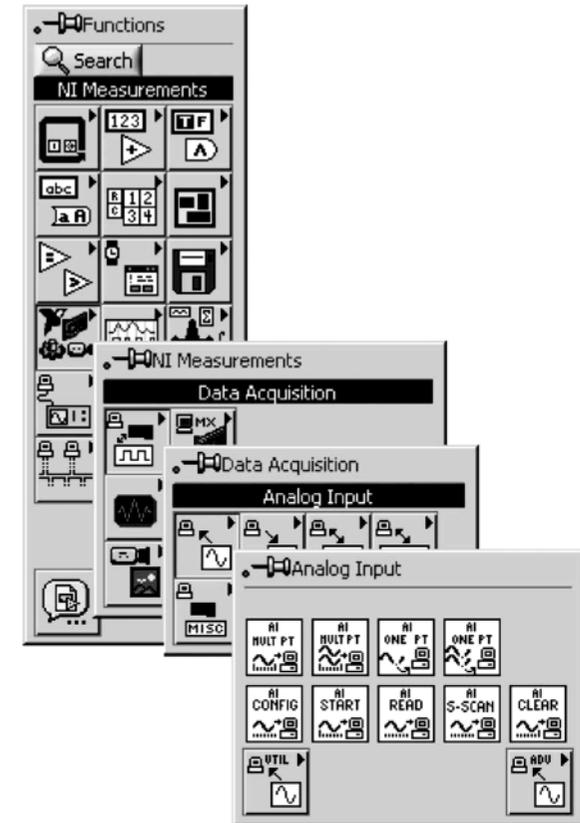


Рис. 3.13. Палитра LabVIEW ввода аналогового сигнала

## Резюме

В этом разделе описано несколько методов накопления в цифровой форме модулированных сигналов с использованием LabVIEW и различных типов аппаратных средств ЭВМ. Как и во всех разработках, есть возможность обмена между типами аппаратных средств ЭВМ по сложности, стоимости и реализации каждой

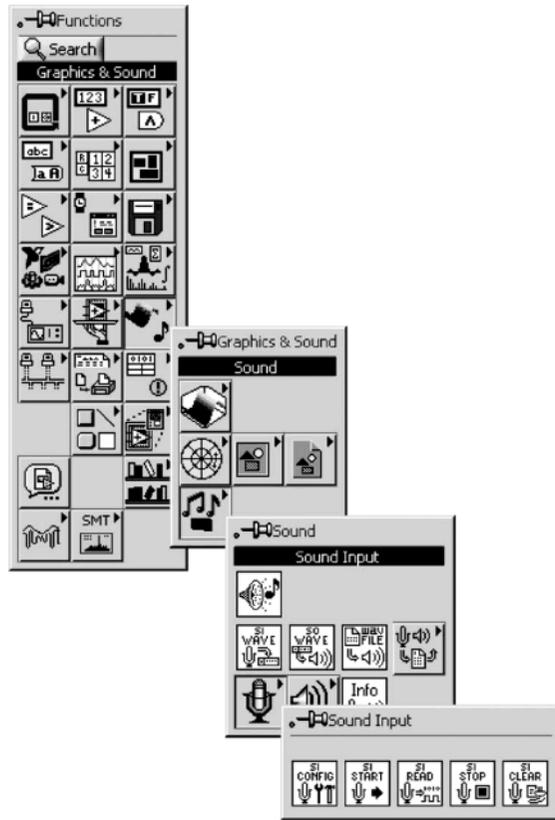


Рис. 3.14. Палитра функций для ввода звука

системы. PXI-изделия NI, имеющиеся в наличии, являются совместимыми с LabVIEW приборами, которые действительно хороши для цифровых систем связи. Недостатки этих изделий – высокая цена и недостаточная способность к перестройке.

В подразделе 3.2 использовалась автономная типовая карта, чтобы показать, как строить полностью цифровой приемник с отдельной частью аппаратных средств ЭВМ.

Теперь, когда известно, какие имеются типы аппаратных средств ЭВМ для дискретизации, остается выбрать метод и начать разработку ВП. Выбор LabVIEW для обработки цифрового сигнала должен сузить ее возможности, хотя и весьма незначительно.

Прибор PXI-5660 действительно осуществляет прореживание, но при этом будет наблюдаться некоторое ухудшение ОСШ, хотя не такое, как при использовании в приемнике выборки из выборки (подвыборки).

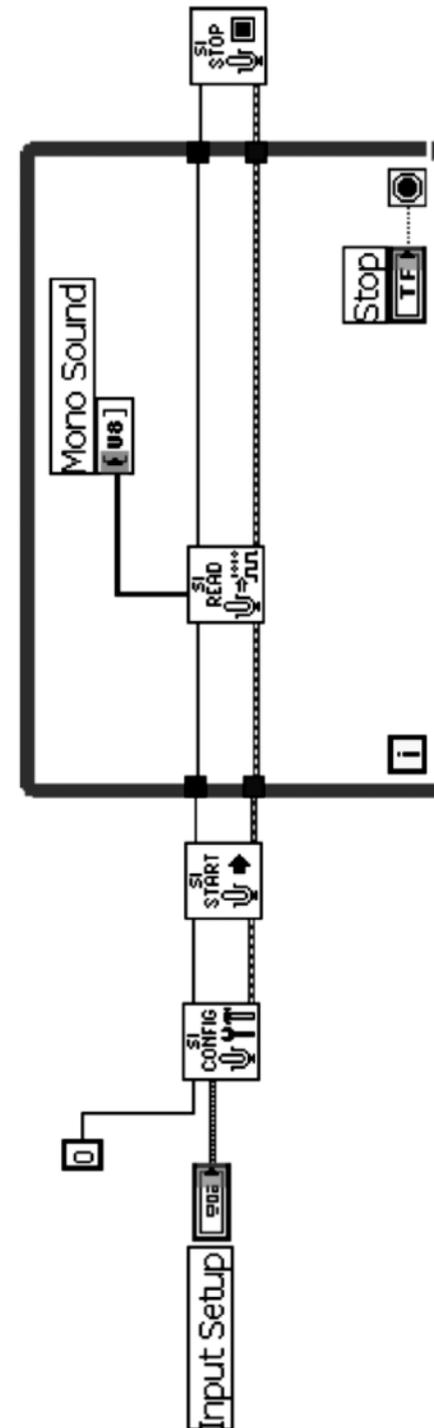


Рис. 3.15 Блок-диаграмма ВП SoundCardCapture.vi

# Цифровая фильтрация (Digital Filtering)



В этом разделе вводится понятие фильтрации, сравниваются аналоговые и цифровые фильтры, описываются фильтры с конечной импульсной характеристикой (КИХ) и бесконечной импульсной характеристикой (БИХ), а также поясняется, как выбрать соответствующий цифровой фильтр для конкретного применения.

## 4.1. Введение в фильтрацию

Процесс фильтрации изменяет содержание частотных составляющих сигнала. Примером может являться регулировка тембра на низких и высоких частотах в усилителях для звуковых сигналов. Такое регулирование обеспечивается фильтрами нижних и верхних частот. При этом изменение амплитуд составляющих низких и верхних частот обеспечивается фильтрами звукового сигнала. Фильтрацией можно также снизить уровень шума. Процесс фильтрации предполагает, что можно выделить нужный сигнал из смеси с другими сигналами. При этом классическая линейная фильтрация основана на том, что составляющие нужного сигнала отличаются в частотной области от остальных сигналов и помех.

## 4.2. Преимущества цифровой фильтрации по сравнению с аналоговой фильтрацией

Аналоговый фильтр имеет аналоговый сигнал как на входе  $x(t)$ , так и на выходе  $y(t)$ . Сигналы  $x(t)$  и  $y(t)$  – функции непрерывной переменной  $t$  и могут иметь бесконечное число мгновенных значений.

Используя современные средства получения выборки и цифровой обработки сигналов, можно заменить аналоговый фильтр цифровым в приложениях, которые требуют универсальности и возможности перепрограммирования. К таким

приложениям можно отнести обработку в аудиотехнике, телекоммуникации, геофизике и при медицинском мониторинге.

Цифровые фильтры имеют следующие преимущества по сравнению с аналоговыми фильтрами:

- цифровые фильтры имеют программное обеспечение, которое делает их легкими в настройке и при проверке;
- цифровые фильтры требуют только арифметических действий умножения и сложения-вычитания;
- цифровые фильтры не изменяют своих параметров и характеристик с изменением температуры или влажности и не требуют компонентов высокой точности;
- цифровые фильтры имеют превосходное отношение характеристики/стоимость;
- цифровые фильтры не меняют своих свойств при изменении условий эксплуатации или из-за старения.

## 4.3. Классификация цифровых фильтров

Можно классифицировать цифровые фильтры по следующим типам:

- фильтры с конечной импульсной характеристикой [КИХ(FIR)-фильтры], также известные как фильтры скользящего среднего значения (СС) (ARMA);
- фильтры с бесконечной импульсной характеристикой [БИХ(IIR)-фильтры], также известные как фильтры авторегрессионного скользящего среднего значения (АСС) (ARMA);
- нелинейные фильтры.

Традиционная классификация фильтров начинается с их классификации в соответствии с видом их импульсных характеристик.

### 4.3.1. Импульсная характеристика (Impulse Response)

Пусть на вход фильтра воздействует короткий по длительности единичный импульс прямоугольной формы в виде одного отсчета на оси времени. Математически этот импульс можно представить в виде уравнения

$$\begin{aligned} x_0 &= 1; \\ x_i &= 0 \text{ для всех } i \neq 0. \end{aligned} \quad (4.1)$$

Тогда отклик дискретного фильтра на это воздействие является его импульсной характеристикой  $h(i\Delta t)$ , зависящей от типа фильтра.

На рис. 4.1 представлена возможная импульсная характеристика дискретного фильтра.

Фурье-преобразование импульсной характеристики есть комплексная передаточная функция фильтра или его частотный отклик (frequency response). Комплексная передаточная функция фильтра характеризует прохождение составляющих сигнала с различными частотами на выход фильтра. Другими словами, она

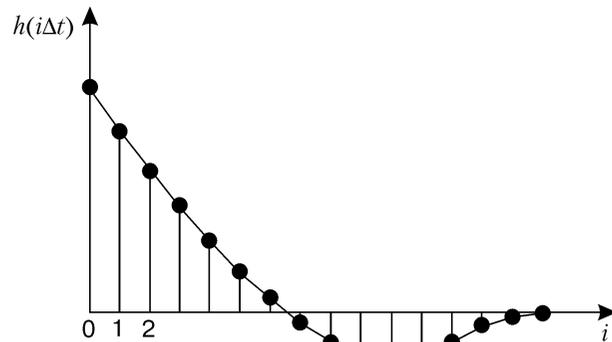


Рис. 4.1. Импульсная характеристика дискретного фильтра

отражает зависимость передаточной функции фильтра от частоты. Для идеального фильтра модуль передаточной функции – единица в полосе пропускания (полосе прозрачности) и ноль в полосе задержания. Идеальный фильтр пропускает все составляющие с частотами, соответствующими полосе пропускания, на выход без изменения, но не пропускает ни одну из составляющих с частотами, соответствующими полосе задержания.

### 4.3.2. Классификация фильтров по импульсной характеристике

Вид импульсной характеристики фильтра определяет, является ли он КИХ-фильтром или БИХ-фильтром. Выходной сигнал КИХ-фильтра зависит только от поступивших на вход отсчетов входного сигнала в предыдущие моменты времени и отсчета в рассматриваемый момент. Выходной сигнал БИХ-фильтра зависит от поступивших на вход отсчетов входного сигнала в предыдущие моменты времени и отсчета в рассматриваемый момент, а также и от отсчетов выходного сигнала в предыдущие моменты.

На примере кассового аппарата можно проиллюстрировать различия между процессами в КИХ- и БИХ-фильтрах. Для этого примера верны следующие определения:

- $x[k]$  – стоимость текущего изделия, введенная в кассовый аппарат;
- $x[k - 1]$  – стоимость предыдущего изделия, введенная в кассовый аппарат;
- $1 \leq k \leq N$ ;
- $N$  – общее количество изделий, стоимость которых введена в кассовый аппарат.

Следующие утверждения описывают действия кассового аппарата:

- кассовый аппарат добавляет к накопленной в нем сумме стоимость каждого изделия, чтобы определить текущую общую стоимость  $y[k]$  изделий;
- стоимость  $y[k]$  до  $k$ -го изделия включительно вычисляется по формуле

$$y[k] = x[k] + x[k - 1] + x[k - 2] + \dots + x[1]; \quad (4.2)$$

- общая стоимость  $N$  изделий есть  $y[N]$ ;
- так как  $y[k]$  равняется общей стоимости изделий до  $k$ -го, а  $y[k - 1]$  равняется общей стоимости изделий до  $(k - 1)$ -го, то уравнение (4.2) может быть переписано в следующем виде:

$$y[k] = x[k - 1] + x[k]; \quad (4.3)$$

- с учетом налога 8,25 % уравнения (4.2) и (4.3) переписываются в виде:

$$y[k] = 1,0825x[k] + 1,0825x[k - 1] + 1,0825x[k - 2] + 1,0825x[k - 3] + \dots + 1,0825x[1]; \quad (4.4)$$

$$y[k] = y[k - 1] + 1,0825x[k]. \quad (4.5)$$

Уравнения (4.4) и (4.5) тождественно описывают алгоритм работы кассового аппарата.

Однако уравнение (4.4) описывает алгоритм работы кассового аппарата только с использованием входных отсчетов  $x[k]$ , в то время как уравнение (4.5) описывает алгоритм с использованием как входных отсчетов  $x[k]$ , так и выходных отсчетов  $y[k]$ . Уравнение (4.4) представляет нерекурсивные, или КИХ-операции, в то время как уравнение (4.5) представляет рекурсивные, или БИХ-операции.

Уравнения, которые описывают операции фильтра и имеют такую же форму, как уравнения (4.2), (4.3), (4.4) и (4.5), называются разностными уравнениями.

КИХ-фильтры – самые простые фильтры при проектировании. Если одиночный импульс воздействует во вход КИХ-фильтра и все последующие входные отсчеты – нуль, выходной сигнал КИХ-фильтра становится равным нулю после конечного времени. Поэтому отклики КИХ-фильтров конечны во времени. Время, требуемое для достижения выходным сигналом КИХ-фильтра нулевого значения, определяется числом коэффициентов фильтра.

Поскольку БИХ-фильтры оперируют потоками прошлых входных и выходных отсчетов, то отклик БИХ-фильтра на единичный импульс никогда теоретически не достигает нуля – отклик бесконечен по длительности.

### 4.3.3. Коэффициенты фильтра

В уравнении (4.4) отсчеты входного сигнала умножаются на коэффициенты, которые в данном примере одинаковы и равны 1,0825. В уравнении (4.5) константы-множители равны 1,0000 для  $y[k - 1]$  и 1,0825 для  $x[k]$ . Константы-множители – коэффициенты фильтра.

Для БИХ-фильтра коэффициенты, на которые умножаются входные отсчеты, называются прямыми (forwards) коэффициентами. Коэффициенты, на которые умножаются выходные отсчеты, называются обратными (reverse) коэффициентами.

### 4.3.4. Характеристики идеальных фильтров

Следующая классификация фильтра базируется на диапазонах частот, которые являются полосами прозрачности или заграждения:

- фильтры нижних частот (ФНЧ) (Lowpass) передают составляющие в нижнем диапазоне частот и уменьшают по амплитуде составляющие верхних частот;
- фильтры верхних частот (ФВЧ) (Highpass) передают составляющие в верхнем диапазоне частот и уменьшают по амплитуде составляющие нижних частот;
- полосовые фильтры (ПФ) (Bandpass) передают составляющие по частоте, соответствующие некоторой полосе частот;
- заграждающие фильтры (ЗФ) (Bandstop) уменьшают амплитуды составляющих в некоторой полосе частот.

На рис. 4.2 показаны следующие зависимости коэффициентов передачи от частоты идеальных фильтров:

- ФНЧ (Lowpass) пропускает все частоты ниже  $f_c$ ;
- ФВЧ (Highpass) пропускает все частоты выше  $f_c$ ;
- ПФ (Bandpass) пропускает все частоты между  $f_{c1}$  и  $f_{c2}$ ;
- ЗФ (Bandstop) уменьшает все частоты между  $f_{c1}$  и  $f_{c2}$ .

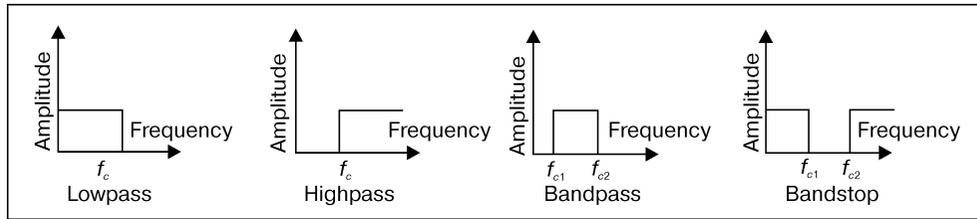


Рис. 4.2. Частотные характеристики идеальных фильтров

Частоты  $f_c$ ,  $f_{c1}$ , и  $f_{c2}$  определяют граничные частоты (частоты среза) для различных фильтров. Цель проектирования фильтров заключается в определении этих граничных частот.

Полоса пропускания идеального фильтра – диапазон частот гармонических составляющих входного сигнала, которые проходят через фильтр без изменения амплитуд. Полоса задержания идеального фильтра – диапазон частот гармонических составляющих, которые не проходят через фильтр. На рис. 4.3 показаны

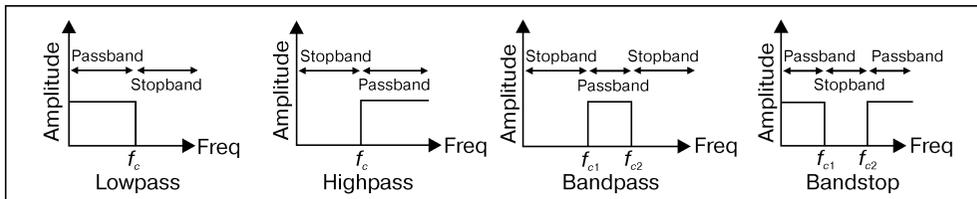


Рис. 4.3. Полоса пропускания (Passband) и полоса задержания (Stopband) для различных фильтров

полоса пропускания (ПП) (Passband) (полоса прозрачности) и полоса задержания (ПЗ) (Stopband) для каждого типа идеального фильтра.

Фильтры, частотные характеристик которых показаны на рис. 4.3, имеют следующие полосы пропускания и полосы задержания:

- ФНЧ и ФВЧ имеют одну полосу пропускания и одну полосу задержания;
- ПФ имеет одну полосу пропускания и две полосы задержания;
- ЗФ имеет две полосы пропускания и одну полосу задержания.

Практически реализуемых идеальных фильтров не существует.

### 4.3.5. Практические (неидеальные) фильтры

Идеальный фильтр имеет модуль передаточной функции (коэффициент передачи), равный единице (0 дБ) в полосе пропускания, и коэффициент передачи, равный нулю ( $-\infty$  дБ) в полосе задержания. Однако реальные фильтры не могут обеспечить параметры и характеристики идеального фильтра. Практически существует конечная полоса перехода между полосой пропускания и полосой прозрачности. В полосе перехода (Transition Regions) коэффициент передачи фильтра изменяется постепенно от единицы (0 дБ) в полосе пропускания до нуля ( $-\infty$  дБ) в полосе задержания.

### 4.3.6. Полоса перехода

На рис. 4.4 показаны полосы пропускания, полосы прозрачности и полосы перехода (Transition Regions) для каждого типа практического фильтра.

В каждом графике рис. 4.4 ось  $X$  представляет частоту, а ось  $Y$  – величину модуля передаточной функции фильтра. Полоса пропускания – область частот, в пределах которой модуль передаточной функции фильтра изменяется от 0 дБ до  $-3$  дБ.

## 4.4. Пульсации АЧХ в полосе пропускания и в полосе ослабления

Во многих применениях можно позволить модулю передаточной функции фильтра в полосе пропускания отличаться от единицы. Эти отличия могут быть определены как «пульсации» коэффициента передачи в полосе пропускания – «рябь» в полосе пропускания, обусловленная различием между фактическим значением коэффициента передачи и желательной единицей. Практически в полосе задержания ослабление (Stopband) не может быть бесконечно, поэтому необходимо определить удовлетворительное значение коэффициента передачи. Измеряются и пульсации полосы пропускания, и ослабление в полосе задержания в децибелах. Следующее уравнение определяет связь децибелов с отношением по амплитуде:

$$K[\text{дБ}] = 20 \log \left( \frac{A_0(f)}{A_i(f)} \right), \quad (4.6)$$

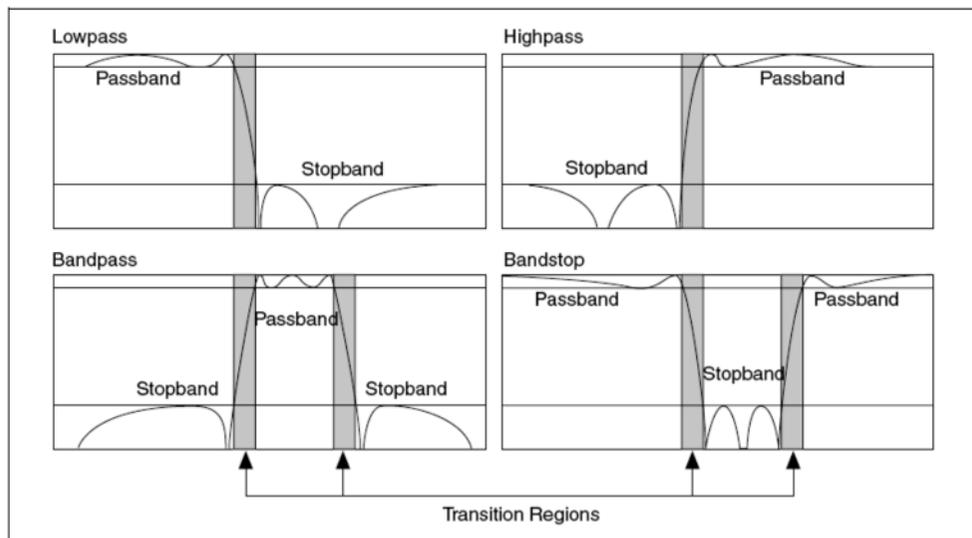


Рис. 4.4. Частотные характеристики неидеальных фильтров

где  $\log$  – логарифм с основанием десять;  $A_i(f)$  – амплитуда составляющей с частотой  $f$  до фильтрации и  $A_0(f)$  – амплитуда составляющей с той же частотой  $f$  после фильтрации.

Когда пульсации в полосе пропускания или в полосе задержания известны, то можно использовать выражение (4.6) для определения отношения амплитуд спектральных составляющих на входе и выходе.

Отношение амплитуд показывает, как близко к идеальному поведению коэффициента передачи при изменении частоты в полосе пропускания или полосе задержания. Например, для пульсаций в полосе пропускания – 0,02 дБ выражение (4.6) позволяет определить коэффициент передачи фильтра в этой полосе:

$$-0,02 = 20 \log \left( \frac{A_0(f)}{A_i(f)} \right), \quad (4.7)$$

откуда

$$\frac{A_0(f)}{A_i(f)} = 10^{-0,001} = 0,9977. \quad (4.8)$$

Выражения (4.7) и (4.8) показывают, что отношение амплитуд составляющих на входе и выходе фильтра близко к единице и является идеальным для полосы пропускания.

При разработке практического фильтра делаются попытки приблизиться к идеальной характеристике с некоторыми ограничениями. В табл. 4.1 приведены сравнительные характеристики идеальных и практических фильтров.

Таблица 4.1. Характеристики идеальных и практических фильтров

Характеристика	Идеальный фильтр	Практический фильтр
Полоса пропускания	Плоская и постоянная	Имеются пульсации
Полоса задержания	Плоская и постоянная	Имеются пульсации
Переходная полоса	Отсутствует	Имеет переходный участок

При разработке практического фильтра приходится идти на компромисс, делая попытку выделить желательную характеристику фильтра за счет менее желательной характеристики. Компромиссы, которые можно допустить, зависят от того, является ли данный фильтр КИХ- или БИХ-фильтром, а также от разрабатываемых алгоритмов.

## 4.5. Частота дискретизации

Выбор частоты дискретизации выборки важен для успеха фильтрации. Максимальная частота составляющих заданного сигнала обычно определяет частоту дискретизации выборки. Вообще говоря, можно выбрать частоту дискретизации выборки в 10 раз выше, чем самая высокая частота составляющих заданного сигнала. При этом необходимо руководствоваться практической необходимостью.

Можно сделать исключения при определении частоты дискретизации выборки, когда граничные частоты фильтра должны быть очень близки к нулевой частоте или к частоте Найквиста (Nyquist).

Фильтры с граничными частотами, близкими к нулевой частоте или частоте Найквиста, могут иметь медленный характер сходимости (convergence) к этим точкам по частоте. Можно выполнить следующие действия, чтобы преодолеть медленную сходимость:

- если граничная частота фильтра слишком близка к частоте Найквиста, то можно увеличить частоту дискретизации выборки;
- если граничная частота фильтра слишком близка к нулевой частоте, то можно уменьшить частоту дискретизации выборки.

Вообще говоря, необходимо регулировать частоту дискретизации выборки, если появляются проблемы с представлением сигналов.

## 4.6. КИХ-фильтры

Фильтры с конечной импульсной характеристикой являются цифровыми фильтрами, которые имеют ограниченную во времени импульсную характеристику. КИХ-фильтры работают только по текущему и прошлым входным отсчетам и являются самыми простыми фильтрами при разработке. КИХ-фильтры также известны как нерекурсивные (трансверсальные) фильтры, как фильтры свертки или как фильтры скользящего среднего значения.

КИХ-фильтры выполняют свертку своих коэффициентов с последовательностью входных отсчетов и производят последовательность выходных отсчетов,

одинаково пронумерованную с входными отсчетами. Конечная свертка, которую выполняет КИХ-фильтр, определяется по уравнению

$$y_i = \sum_{k=0}^{n-1} h_k x_{i-k} \quad (4.9)$$

где  $x$  – последовательность входных отсчетов, поступающая на вход фильтра;  $y$  – последовательность отсчетов на выходе фильтра;  $h$  – коэффициенты КИХ-фильтра.

КИХ-фильтры имеют следующие особенности:

- могут обеспечить линейную ФЧХ при симметрии коэффициентов фильтра в реализации проекта;
- всегда устойчивы;
- позволяют фильтровать сигналы на основе использования свертки. Поэтому, вообще говоря, можно связывать задержку (delay) с последовательностью выходных отсчетов, как показано в следующем уравнении:

$$\text{delay} = \frac{n-1}{2},$$

где  $n$  – число коэффициентов КИХ-фильтра.

На рис. 4.5 показана типичная амплитудно-частотная (АЧХ) (Magnitude Response) и фазочастотная (ФЧХ) (Phase Response) характеристики КИХ-фильтра в зависимости от циклической частоты в герцах. Прерывистый характер фазовой характеристики следует из введенного интервала однозначности по фазе, равного  $\pi$ .

Однако фазочастотная характеристика КИХ-фильтра явно линейна.

### 4.6.1. Отводы (Taps)

Термины *отвод* и *отводы* часто появляются в описаниях КИХ-фильтров и операций при КИХ-фильтрации. На рис. 4.6 показан процесс ответвления сигналов (Tapping) при получении отсчетов выходного сигнала.

На рис. 4.6 представлены изменения  $n$  отсчета в сдвиговом регистре, содержащем входные отсчеты  $[x_n, x_{n-1}, \dots]$ . Термин *отвод* обозначает процесс ответвления на выходе сдвигового регистра для получения слагаемого  $h_k x_{i-k}$  в уравнении (4.9). Термин *отводы* обычно обозначает номера коэффициентов для фильтра КИХ.

### 4.6.2. Проектирование КИХ-фильтров

Разработка КИХ-фильтров имеет целью приближение к желательной комплексной передаточной функции дискретно-временной системы. Имеются самые общие методы приближения к желательной амплитудно-частотной характеристике, в то время как выдерживается линейность фазовой характеристики.

Линейность фазовой характеристики подразумевает, что все частотные составляющие в системе имеют одинаковую задержку во времени.

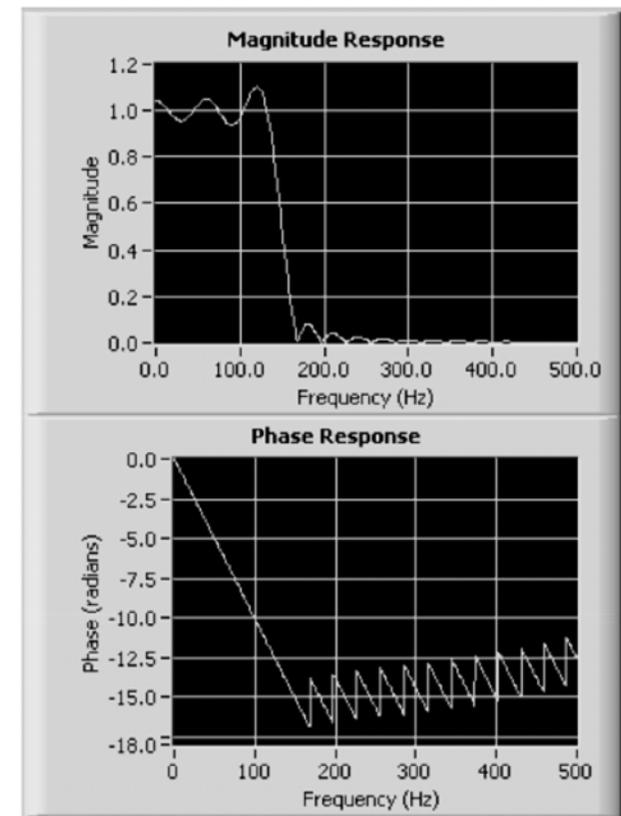


Рис. 4.5. АЧХ и ФЧХ КИХ-фильтра в зависимости от циклической частоты

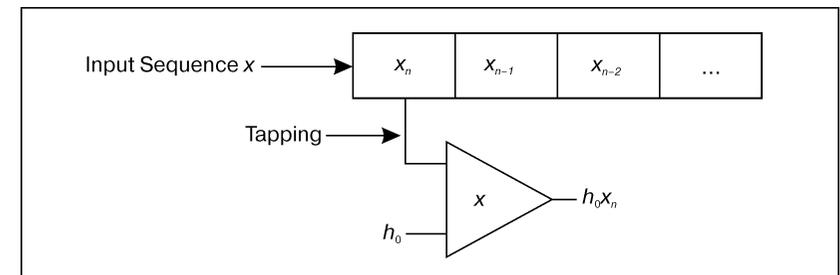


Рис. 4.6. Ответвление входной последовательности (Input Sequence)

На рис. 4.7 показана блок-диаграмма виртуального прибора (VI), который позволяет получить передаточную функцию полосового КИХ-фильтра с равномерными пульсациями (Equiripple).

В виртуальном приборе, представленном блок-диаграммой на рис. 4.7, выполняются следующие шаги для вычисления передаточной функции фильтра.

1. Пропускается импульсный сигнал через фильтр.
2. Передается далее фильтрованный сигнал от блока фильтра к блоку быстрого преобразования Фурье (БПФ) (FFT VI). Структура блока определяет тип фильтра: ФНЧ, ФВЧ, ПФ или ЗФ. Сигнал на выходе блока является импульсом отклика фильтра.
3. Используется ВП БПФ (FFT VI), чтобы выполнить Фурье-преобразование импульсного отклика и вычислить частотную характеристику фильтра. Далее эту характеристику сравнивают с Фурье-преобразованием пары  $h(t) \Leftrightarrow H(f)$ , где  $h(t)$  – импульсная характеристика;  $H(f)$  – передаточная функция фильтра.
4. Используется функция перебора подмножества (Array Subset), чтобы уменьшить количество данных, полученных после БПФ (FFT VI). Половина реального результата FFT избыточна, поэтому в виртуальном приборе необходимо обрабатывать только половину данных, полученных от блока FFT VI.
5. Используется функция Complex To Polar, чтобы получить форму данных, полученных от FFT VI в экспоненциальном виде  $A_n e^{j\phi_n}$ , где  $A_n$  – амплитуда  $n$ -й составляющей;  $\phi_n$  – ее начальная фаза. Экспоненциальную форму комплексных чисел с выхода FFT VI легче интерпретировать, чем компоненты FFT, представленные в прямоугольных координатах (в алгебраическом виде).
6. Начальные фазы в радианах преобразовываются в градусы.
7. Преобразовываются амплитуды в децибелы.

На рис. 4.8 показаны амплитудно-частотная и фазочастотная характеристики, полученные в виртуальном приборе, представленном на рис. 4.7.

На рис. 4.8 прерывистый характер фазовой характеристики следует из неоднозначности измерений угла, выходящего за пределы интервала  $\pi$ . Однако фазо-

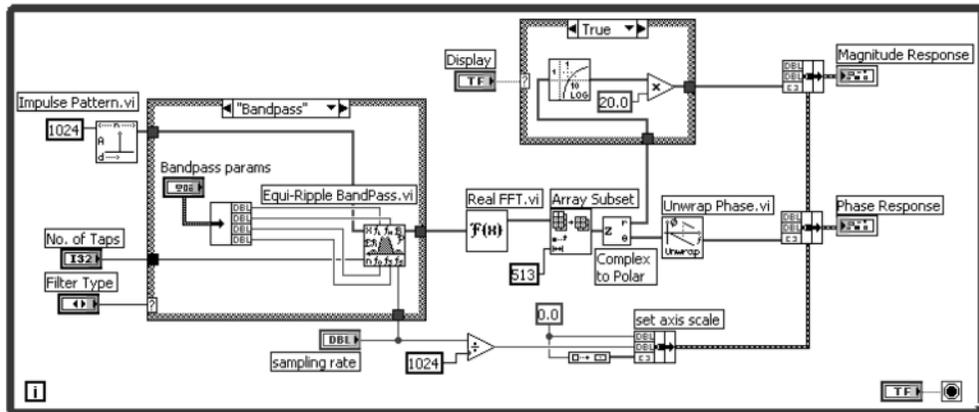


Рис. 4.7. Виртуальный прибор для полосового КИХ-фильтра с равномерными пульсациями АЧХ (Equiripple)

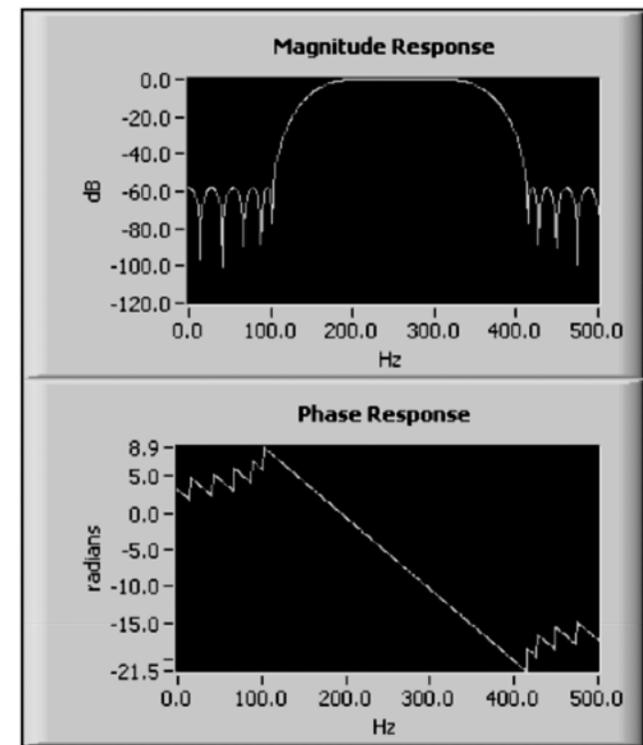


Рис. 4.8. Амплитудно-частотная и фазочастотная характеристики полосового КИХ-фильтра с равномерными пульсациями (Equiripple)

вая характеристика линейна, потому что составляющие с разными частотами имеют одинаковую задержку на временной оси.

Поскольку КИХ-фильтры имеют АЧХ с пульсациями, их проектирование имеет следующие особенности:

- проектирование фильтра с АЧХ, максимально близкой к идеальной;
- проектирование фильтра, который распределяет пульсации на АЧХ желательным способом.

Например, ФНЧ имеет идеальную характерную АЧХ. Специфическое применение этого фильтра могло бы позволить некоторые пульсации АЧХ в полосе пропускания и большие значения пульсаций в полосе задержания. Алгоритм проектирования фильтра должен позволить балансировать между требованиями к пульсациям и созданием самой узкой полосы перехода от полосы прозрачности к полосе задержания.

Самые общие методы для разработки КИХ-фильтров – это метод «окон» и алгоритм Parks-McClellan, также известный как Remez Exchange.

### 4.6.3. Проектирование КИХ-фильтров с помощью окон (Windowing)

Windowing – самая простая методика для разработки КИХ-фильтров. При проектировании КИХ-фильтров с использованием Windowing применяют обратное БПФ к желательной передаточной функции и применяют окно сглаживания к результату. Окно сглаживания – окно в области времени.

Необходимо выполнить следующие шаги, чтобы спроектировать КИХ-фильтр с помощью окон (Windowing):

- 1) выбрать идеальную частотную передаточную функцию;
- 2) вычислить импульсную характеристику для идеальной частотной передаточной функции;
- 3) усечь импульсную характеристику, чтобы определить конечное число коэффициентов; чтобы выполнить ограничение линейности фазовой характеристики, необходимо выдержать симметрию относительно центрального коэффициента;
- 4) применить симметричное окно сглаживания.

Усечение идеального отклика импульса приводит к явлению Гиббса (Gibbs).

Явление Гиббса проявляется как колебательное поведение отсчетов около граничных частот в частотной характеристике фильтра КИХ. Можно уменьшить эффекты явления Гиббса, используя окно сглаживания, чтобы сгладить усечение идеальной импульсной характеристики. Для того чтобы уменьшить коэффициенты КИХ в начале и конце последовательности коэффициентов, можно уменьшить высоту боковых лепестков в частотной характеристике. Однако уменьшение высоты боковых лепестков заставляет главный лепесток расширяться, и заканчиваться более широкой полосой перехода к частотам полосы задержания.

Выбор окна сглаживания требует компромисса между высотой боковых лепестков около граничных частот и шириной полосы перехода. Уменьшение высоты боковых лепестков около граничных частот увеличивает ширину полосы перехода (снижает крутизну АЧХ). Уменьшение ширины полосы перехода увеличивает высоту боковых лепестков около граничных частот.

Проектирование КИХ-фильтров на основе окон (Windowing) имеет следующие недостатки:

- неэффективность:
  - неравномерное распределение пульсаций;
  - более широкая полоса перехода, чем в других методах проектирования;
- трудность в определении параметров – определение граничной частоты, около которой имеются специфические затухания.

Проектировщики фильтра должны определить:

- идеальную граничную частоту;
- частоту дискретизации;
- число отсчетов сигналов;
- тип окна.

Проектирование КИХ-фильтров на основе Windowing не требует большого количества вычислительных ресурсов. Поэтому Windowing – самая быстрая техника для того, чтобы разработать КИХ-фильтры. В то же время Windowing – не самая лучшая техника для разработки КИХ-фильтров.

### 4.6.4. Проектирование оптимальных КИХ-фильтров на основе использования алгоритма Parks-McClellan

В алгоритме Parks-McClellan или Remez Exchange при проектировании коэффициентов КИХ-фильтра используется итеративная (iterative) методика, основанная на критерии минимума ошибок. Можно использовать алгоритм Parks-McClellan, чтобы проектировать фильтр с оптимальной линейной фазовой характеристикой при определении коэффициентов КИХ-фильтра. Фильтры, которые проектируются на основе алгоритма Parks-McClellan, оптимальны, потому что они минимизируют максимальную ошибку между фактической частотной характеристикой фильтра и идеальной частотной характеристикой.

Оптимальные КИХ-фильтры уменьшают неблагоприятные эффекты на граничных частотах. Проектирование оптимальных КИХ-фильтров также предполагает больше возможностей для контроля ошибок приближения в различных полосах на оси частот, чем другие методы проектирования КИХ-фильтров, как, например проектирование КИХ-фильтров на основе Windowing, которое не обеспечивает никакого контроля над ошибками приближения в различных полосах по оси частот.

Оптимальные КИХ-фильтры, проектируемые на основе использования алгоритма Parks-McClellan, имеют:

- частотную характеристику со взвешенными пульсациями, равномерно распределенными по полосе пропускания и полосе задержания;
- узкую полосу перехода.

КИХ-фильтры, проектируемые на основе использования алгоритма Parks-McClellan, имеют оптимальную импульсную характеристику. Однако процесс такого проектирования сложен, требует большого количества вычислительных ресурсов и намного более длителен, чем проектирование КИХ-фильтров на основе Windowing.

### 4.6.5. Проектирование Equiripple КИХ-фильтров на основе использования алгоритма Parks-McClellan

Можно использовать алгоритм Parks-McClellan, чтобы проектировать КИХ-фильтры, имеющие АЧХ с равномерными пульсациями (Equiripple). Фильтры проекта Equiripple имеют АЧХ, одинаковые по уровню величины незначительных пульсаций в полосе пропускания и полосе задержания. Кроме того, этот алгоритм позволяет разрабатывать фильтры с линейной фазовой характеристикой.

Необходимо определить следующие характеристики и параметры фильтра, чтобы проектировать Equiripple-КИХ-фильтр:

- граничная частота;
- число отсчетов сигнала;
- тип фильтра: ФНЧ, ФВЧ, ПФ или ЗФ;
- полоса прозрачности;
- полоса задержания.

Граничная частота для Equiripple-фильтров определяет края полос пропускания и задержания или края и той и другой одновременно. Пульсации в полосе пропускания и полосе задержания Equiripple-фильтров определяются следующими параметрами частотной характеристики:

- в полосе пропускания АЧХ больше или равна единице;
- в полосе задержания частотная характеристика меньше или равна заданному затуханию.

Например, если разрабатывается ФНЧ, граничная частота его полосы пропускания – самая высокая частота, для которой условия определения полосы пропускания являются справедливыми.

Точно так же граничная частота полосы задержания – самая низкая частота, для которой условия определения полосы задержания являются справедливыми.

#### 4.6.6. Проектирование узкополосных КИХ-фильтров

Использование обычных методов для проектирования КИХ-фильтров с особенно узкими полосами пропускания может приводить к большим размерам фильтра (большому числу его коэффициентов). КИХ-фильтры с большим числом коэффициентов часто требуют трудоёмкого проектирования, имеют длительное время выполнения операций и восприимчивы к числовой погрешности коэффициентов. В некоторых случаях обычные методы проектирования фильтров типа алгоритма Parks-McClellan не могут обеспечить приемлемо узкую полосу пропускания КИХ-фильтров.

Техника проектирования фильтров на основе интерполирования конечной импульсной характеристики (IFIR) предлагает эффективный алгоритм для того, чтобы разрабатывать узкополосные КИХ-фильтры. Использование IFIR-техники обеспечивает разработку узкополосных фильтров, которые требуют меньшего количества коэффициентов и вычислений, чем фильтры, которые проектируются на основе применения алгоритма Parks-McClellan. Виртуальный прибор Narrowband Coefficients VI использует IFIR-технику, чтобы генерировать коэффициенты узкополосного КИХ-фильтра.

Необходимо определить следующие параметры для описания характеристик узкополосных фильтров:

- тип фильтра: ФНЧ, ФВЧ, ПФ или ЗФ;
- пульсации в полосе пропускания в линейном масштабе;
- частоту дискретизации;

- полосу пропускания, которая относится к ширине полосы пропускания для узкополосных и заграждающих фильтров;
- полосу частот заграждения, которая относится к ширине полосы заграждения для заграждающих фильтров, и полосы заграждения у остальных фильтров;
- частоту центра АЧХ для ПФ и ЗФ;
- ослабление в полосе заграждения ЗФ в децибелах.

На рис. 4.9 показана блок-диаграмма виртуального прибора для оценки частотной характеристики узкополосного КИХ-ПФ-фильтра на основе преобразования импульсной характеристики в область частот.

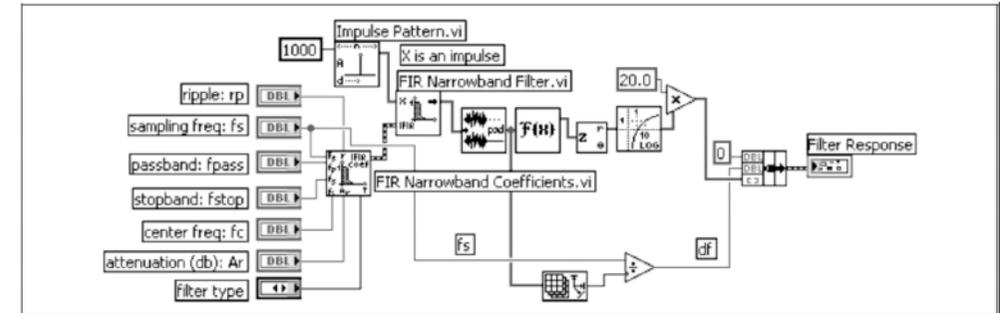


Рис. 4.9. Оценка частотной характеристики узкополосного КИХ-фильтра (ПФ)

На рис. 4.10 показана характеристика фильтра от нуля до частоты Найквиста (Nyquist), блок определения которой встраивается в виртуальный прибор, представленный на рис. 4.9.

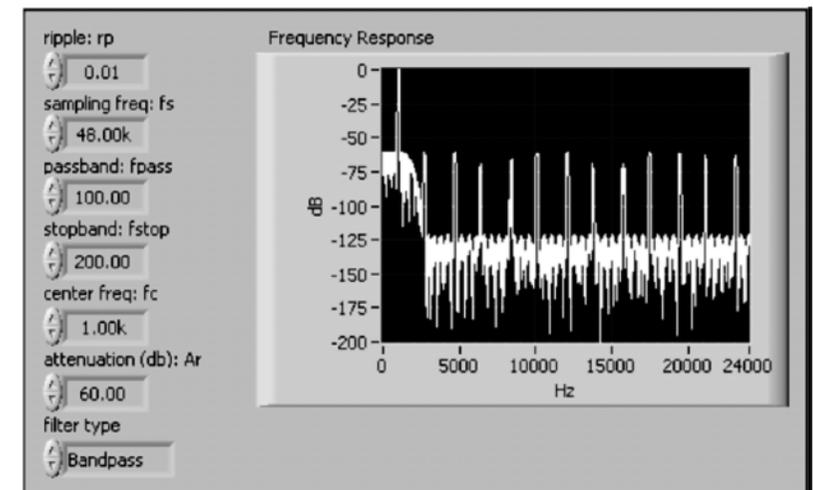


Рис. 4.10. Оценка частотной характеристики узкополосного КИХ-фильтра (ПФ) от нуля до частоты Найквиста

На рис. 4.10 узкая полоса пропускания сосредотачивается приблизительно около 1 кГц. Центр узкой полосы пропускания, равной 1 кГц в характеристике фильтра устанавливается групповым контролем на лицевой панели, показанной на рис. 4.10.

На рис. 4.11 представлена более подробно характеристика КИХ-ПФ-фильтра.

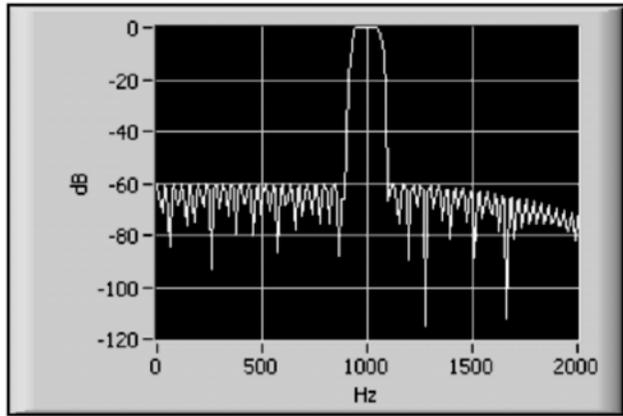


Рис. 4.11. Деталь оценки частотной характеристики узкополосного полосового КИХ-фильтра

Узкая полоса пропускания сосредотачивается приблизительно около частоты 1 кГц и обеспечивает ослабление сигнала на 60 дБ за пределами полосы пропускания.

Можно обратиться к работам P.P. Vaidyanathan [34, 35] и Y. Neuvo [28], а также к другим, представленным в библиографическом списке работам за дополнительной информацией о проектировании IFIR-фильтров.

#### 4.6.7. Проектирование широкополосных КИХ-фильтров

Можно также использовать ту же IFIR-технику, чтобы разработать широкополосный КИХ-ФНЧ и широкополосный КИХ-ФВЧ. Широкополосная АЧХ КИХ-ФВЧ имеет граничную частоту, расположенную вблизи частоты Найквиста. Широкополосная АЧХ КИХ-ФВЧ имеет граничную частоту около нуля. Можно использовать виртуальный прибор FIR Narrowband Coefficients VI, чтобы проектировать широкополосные КИХ-ФНЧ и широкополосные КИХ-ФВЧ. На рис. 4.12 показана частотная характеристика виртуального прибора, представленного на рис. 4.9 блок-диаграммой. Эта блок-диаграмма может встраиваться в другие виртуальные приборы, когда надо оценить частотную характеристику широкополосного КИХ-ФНЧ.

На рис. 4.12 элементы управления на передней панели виртуального прибора позволяют установить узкую полосу пропускания между границей полосы задерж-

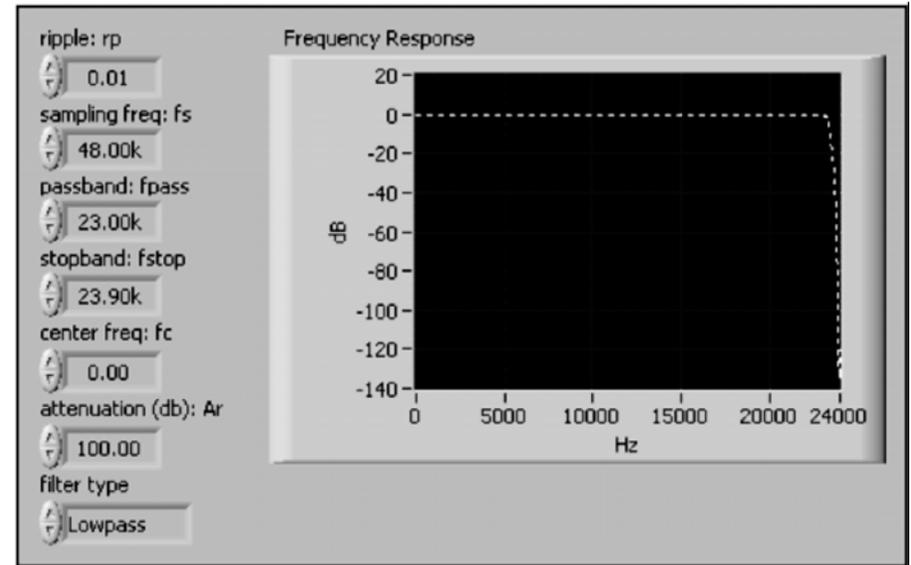


Рис. 4.12. Частотная характеристика широкополосного КИХ-фильтра (ФНЧ) от нуля до частоты Найквиста

жания 23,9 кГц и частотой Найквиста 24 кГц. Однако частотная характеристика фильтра изменяется от нуля до 23,9 кГц, что делает этот фильтр широкополосным.

## 4.7. БИХ(IIR)-фильтры

Фильтры с бесконечной импульсной характеристикой (БИХ), также известные как рекурсивные фильтры или как фильтры авторегрессионного скользящего среднего значения (ARMA), производят вычисления на основе потока предыдущих входных отсчетов и потока предыдущих выходных отсчетов. Импульсная характеристика БИХ-фильтра – отклик произвольного БИХ-фильтра на импульс, который определяется уравнением (4.1). Теоретически импульсная характеристика БИХ-фильтра никогда не достигает нуля, и получается бесконечный по длительности отклик.

БИХ-фильтры характеризует следующее общее разностное уравнение:

$$y_i = \frac{1}{a_0} \left( \sum_{j=0}^{N_b-1} b_j x_{i-j} - \sum_{k=1}^{N_a-1} a_k y_{i-k} \right), \quad (4.10)$$

где  $\{b_j\}$  – набор прямых коэффициентов;  $N_b$  – число прямых коэффициентов;  $\{a_k\}$  – набор обратных коэффициентов;  $N_a$  – число обратных коэффициентов.

Уравнение (4.10) описывает фильтр с импульсной характеристикой теоретически бесконечной длины для коэффициентов, отличных от нуля. Однако в прак-

тических применениях импульсная характеристика устойчивого БИХ-фильтра сходится к нулю и имеет конечное число отсчетов.

В большинстве проектов БИХ-фильтров и во всех БИХ-фильтрах в LabVIEW коэффициент  $a_0 = 1$ . Выходной отсчет при текущем индексе  $i$  есть сумма масштабируемых текущих отсчетов и предыдущих входных отсчетов, а также масштабируемых предыдущих выходных отсчетов:

$$y_i = \sum_{j=0}^{N_b-1} b_j x_{i-j} - \sum_{k=0}^{N_a-1} a_k y_{i-k}, \quad (4.11)$$

где  $x_i$  – текущий входной отсчет;  $x_{i-j}$  – прошлые входные отсчеты;  $y_{i-k}$  – прошлые выходные отсчеты.

БИХ-фильтры могут иметь пульсации в полосе пропускания, полосе задержания или в обеих полосах. Также БИХ-фильтры имеют нелинейную фазовую характеристику.

### 4.7.1. Каскадная форма БИХ-фильтрации

Прямая форма передаточной функции БИХ-фильтра определяется следующим уравнением:

$$H(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_{N_b-1} z^{-(N_b-1)}}{1 + a_1 z^{-1} + \dots + a_{N_a-1} z^{-(N_a-1)}}. \quad (4.12)$$

Прямая форма (4.12) чувствительна к ошибкам, обусловленным ограниченным числом уровней квантования коэффициентов, и зависит от пределов точности в вычислениях. Также фильтр с первоначально устойчивым проектом может стать неустойчивым с увеличением числа коэффициентов. При этом порядок фильтра пропорционален числу его коэффициентов. Естественно, при увеличении числа коэффициентов порядок фильтра увеличивается. Следовательно, при увеличении порядка фильтр становится более нестабильным.

Можно уменьшить чувствительность фильтра к ошибке в записи уравнения (4.12), преобразовав отношение полиномов в  $z$ -образе  $H(z)$  передаточной функции к форме записи, в которой функция передачи прямой формы подразделяется на секции более низкого порядка, соответствующие блокам фильтра. Для этого можно рассчитать нули и полюсы  $H(z)$ , приравняв к нулю числитель и знаменатель правой части выражения (4.12), а затем записать его в форме, обеспечивающей представление системы в виде каскадного соединения звеньев первого порядка. Однако при этом часть звеньев может оказаться с комплексными коэффициентами. Поэтому при каскадной реализации вещественных фильтров их представляют в виде набора секций второго порядка. В этом случае пары комплексно-сопряженных нулей и полюсов обеспечивают образование блоков или секций второго порядка (структура фильтра типа II). Такой подход к реализации фильтров обеспечивает снижение зависимости результата фильтрации от конечной точности вычислений.

Факторизация выражения (4.12) в виде секций второго порядка приводит к тому, что передаточная функция фильтра становится зависящей от функций фильтра второго порядка, как показано в следующем уравнении:

$$H(z) = \prod_{k=1}^{N_s} \frac{b_{0k} + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}}, \quad (4.13)$$

где  $N_s$  – число секций;  $N_s = \left\lceil \frac{N_a}{2} \right\rceil$  – наибольшее число, меньшее или равное  $\frac{N_a}{2}$ ,

$N_a \geq N_b$ .

Можно описать структуру фильтра, описанную выражением (4.13), как каскадное соединение фильтров второго порядка. На рис. 4.13 представлена структура каскадной фильтрации.

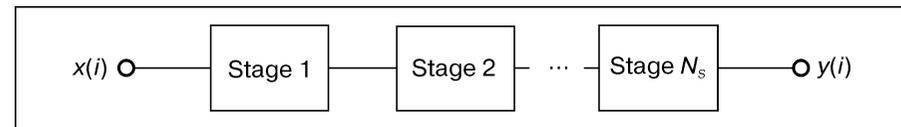


Рис. 4.13. Блоки каскадной фильтрации

Осуществляется каждый индивидуальный блок фильтрации на рис. 4.13 в виде фильтра типа II с каскадной формой соединения блоков. Используется такой вид фильтров по следующим причинам:

- каскадная структура фильтров типа II требует минимального числа арифметических действий;
- каскадная структура фильтров типа II требует минимального числа элементов задержки или внутренних состояний фильтра;
- каждый  $k$ -й каскад имеет один вход, один выход и два предыдущих внутренних состояния  $s_k[i-1]$  и  $s_k[i-2]$ .

Если  $n$  – число отсчетов во входной последовательности, процедуры операций фильтрации могут быть показаны в уравнениях

$$\begin{aligned} y_0[i] &= x[i]; \\ s_k[i] &= y_{k-1}[i-1] - a_{1k} s_k[i-1] - a_{2k} s_k[i-2], \quad k = 1, 2, \dots, N_s; \\ y_k[i] &= b_{0k} s_k[i] + b_{1k} s_k[i-1] + b_{2k} s_k[i-2], \quad k = 1, 2, \dots, N_s \end{aligned}$$

для каждого отсчета  $i = 0, 1, 2, \dots, n-1$ .

### 4.7.2. Фильтрация второго порядка

ФНЧ и ФВЧ, которые имеют единственную граничную частоту, можно проектировать в виде секций фильтров второго порядка непосредственно. В результате БИХ-ФНЧ или ФВЧ содержат в качестве каскадов фильтры второго порядка.

Каждый каскад фильтра второго порядка имеет следующие характеристики:

- $k = 1, 2, \dots, N_s$ , где  $k$  – номер секции фильтра второго порядка и  $N_s$  – общее количество секций в виде фильтра второго порядка;

- каждый каскад фильтра второго порядка имеет два обратных коэффициента ( $a_{1k}, a_{2k}$ );
- общее количество обратных коэффициентов равняется  $2N_s$ ;
- каждый каскад фильтра второго порядка имеет три прямых коэффициента ( $b_{0k}, b_{1k}, b_{2k}$ );
- общее количество прямых коэффициентов равняется  $3N_s$ .

В сигналах, обрабатываемых в ВП с **обратными коэффициентами (Reverse Coefficients)** и **прямыми коэффициентами (Forward Coefficients)**, вектор параметров этих ВП содержит коэффициенты для одной секции фильтра второго порядка, коэффициенты для следующей секции фильтра второго порядка и т. д. Например, БИХ-фильтр с двумя секциями фильтра второго порядка должен иметь четыре обратных коэффициента и шесть прямых коэффициентов:

- общее количество обратных коэффициентов равно  $2N_s = 2 \cdot 2 = 4$ ;
- обратные коэффициенты  $\{a_{11}, a_{21}, a_{12}, a_{22}\}$ ;
- общее количество прямых коэффициентов  $3N_s = 3 \cdot 2 = 6$ ;
- прямые коэффициенты  $\{b_{01}, b_{11}, b_{21}, b_{02}, b_{12}, b_{22}\}$ .

### 4.7.3. Фильтрация четвертого порядка

Для ПФ и ЗФ, которые имеют две граничные частоты, секции в виде фильтра четвертого порядка – более приемлемая форма проекта фильтра, чем секции в виде фильтра второго порядка. БИХ-ПФ или БИХ-ЗФ состояются из фильтров четвертого порядка.

Каждый каскад в виде фильтра четвертого порядка имеет следующие характеристики:

- $k = 1, 2, \dots, N_s$ , где  $k$  – число каскадов в виде фильтров четвертого порядка и  $N_s$  – общее количество каскадов фильтров четвертого порядка;
- $N_s = \left\lceil \frac{N_a + 1}{4} \right\rceil$ ;
- каждый каскад фильтра четвертого порядка имеет четыре обратных коэффициента ( $a_{1k}, a_{2k}, a_{3k}, a_{4k}$ );
- общее количество обратных коэффициентов равняется  $4N_s$ ;
- каждый каскад фильтра четвертого порядка имеет пять прямых коэффициентов ( $b_{0k}, b_{1k}, b_{2k}, b_{3k}, b_{4k}$ );
- общее количество прямых коэффициентов равняется  $5N_s$ .

При выполнении каскадных секций в виде фильтров четвертого порядка используется та же самая методика, что и при фильтрации на основе секций в виде фильтров второго порядка.

Следующие уравнения показывают, как действует фильтрация на выходные отсчеты секций четвертого порядка:

$$y_0[i] = x[i];$$

$$s_k[i] = y_{k-1}[i-1] - a_{1k}s_k[i-1] - a_{2k}s_k[i-2] - a_{3k}s_k[i-3] - a_{4k}s_k[i-4];$$

$$y_k[i] = b_{0k}s_k[i] + b_{1k}s_k[i-1] + b_{2k}s_k[i-2] - b_{3k}s_k[i-3] - b_{4k}s_k[i-4],$$

где  $k = 1, 2, \dots, N_s$ .

### 4.7.5. Типы БИХ-фильтров

Цифровые БИХ-проекты фильтров основываются на классических аналоговых проектах и включают следующие типы фильтров:

- фильтры Баттерворта;
- фильтры Чебышева;
- фильтры Чебышева II, также известные как инверсные чебышевские фильтры или фильтры Чебышева типа II;
- эллиптические фильтры, также известные как фильтры Кауэра (Cauer);
- фильтры Бесселя.

Проекты БИХ-фильтров отличаются по степени резкости перехода между их полосой пропускания и полосой задержания, и в этих областях показывают различие характеристик в полосе пропускания или полосе задержания.

### 4.7.6. Минимизация пиковой ошибки

Фильтры Чебышева, фильтры Чебышева II и эллиптические фильтры минимизируют пиковую ошибку вычислений, обеспечивая максимально допустимую ошибку в их частотных характеристиках.

Максимально допустимая ошибка – максимальная абсолютная величина различия между идеальной частотной характеристикой фильтра и его фактической частотной характеристикой. Величина пульсаций АЧХ в децибелах, допускаемая в частотной характеристике фильтра, определяет максимальную допустимую ошибку. В зависимости от типа фильтр минимизирует пиковую ошибку в полосе пропускания, полосе задержания или в обеих полосах.

### 4.7.7. Фильтры Баттерворта (Butterworth)

АЧХ фильтров Баттерворта имеют следующие особенности:

- гладкий вид на всех частотах;
- монотонное уменьшение от указанных граничных частот;
- максимальная плоскость вершины с идеальным единичным значением в полосе пропускания и нулевым значением в полосе задержания;
- частота половинной мощности спектральной составляющей или частота снижения АЧХ на 3 дБ, соответствующая указанным граничным частотам.

Преимущество фильтров Баттерворта: гладкость их АЧХ, монотонность снижения частотной характеристики в полосе перехода. На рис. 4.14 представлена частотная характеристика ФНЧ Баттерворта.

Как показано на рис. 4.14, после того как определена граничная частота фильтра Баттерворта, LabVIEW устанавливает крутизну перехода АЧХ, пропорциональную порядку фильтра. Повышение порядка фильтра Баттерворта приближает к идеальной частотную характеристику ФНЧ.

Фильтры Баттерворта не всегда обеспечивают хорошее приближение их АЧХ к идеальной частотной характеристике из-за медленного спада между полосой пропускания и полосой задержания.

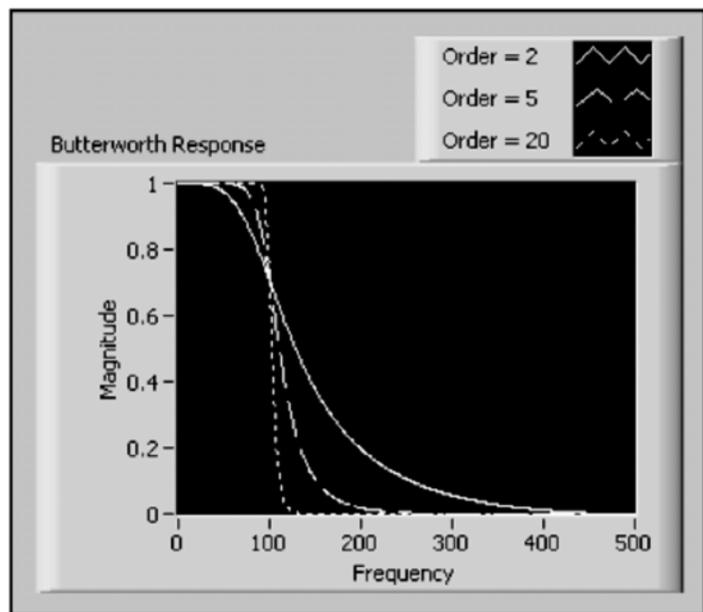


Рис. 4.14. Частотная характеристика ФНЧ Баттерворта

### 4.7.8. Фильтры Чебышева

Фильтры Чебышева имеют следующие особенности:

- минимизация пиковой ошибки в полосе пропускания;
- неизменные пульсации частотной характеристики в полосе пропускания;
- монотонно убывающая частотная характеристика к полосе задержания;
- более крутой спад АЧХ по сравнению с фильтром Баттерворта (фильтр Чебышева может обеспечить более крутой переход между полосой пропускания и полосой задержания в фильтре более низкого порядка).

Крутой переход между полосой пропускания и полосой задержания фильтра Чебышева обеспечивает меньшие абсолютные ошибки и более высокие скорости выполнения операций, чем фильтр Баттерворта.

На рис. 4.15 показана частотная характеристика ФНЧ Чебышева.

На этом рисунке заданная максимально допустимая ошибка вычислений ограничивает пульсации частотной характеристики ФНЧ в полосе пропускания. Кроме того, обеспечивается крутой спад частотной характеристики в полосе задержания.

### 4.7.9. Фильтры Чебышева II

Фильтры Чебышева II имеют следующие характеристики:

- минимизация пиковой ошибки в полосе задержания;

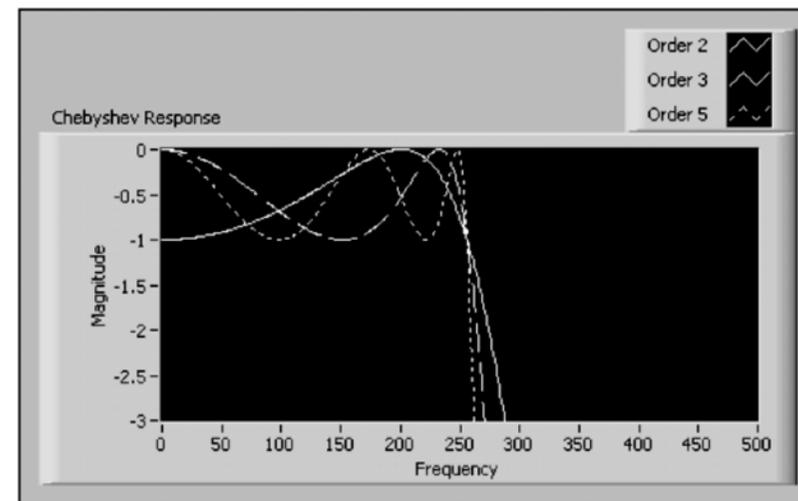


Рис. 4.15. Частотная характеристика ФНЧ Чебышева

- неизменная величина пульсаций частотной характеристики в полосе задержания;
- монотонно убывающая частотная характеристика в полосе пропускания;
- более крутой спад АЧХ, чем у фильтра Баттерворта.

Фильтры Чебышева II подобны фильтрам Чебышева, однако отличаются следующими особенностями:

- минимизируют пиковую ошибку в полосе задержания, а не в полосе пропускания, что является преимуществом фильтров Чебышева II по отношению к фильтрам Чебышева;
- имеют неизменные пульсации частотной характеристики в полосе задержания в отличие от полосы пропускания;
- имеют монотонно убывающую частотную характеристику в полосе пропускания в отличие от полосы задержания.

На рис. 4.16 показаны частотные характеристики ФНЧ Чебышева II. Максимальная допустимая ошибка ограничивает неизменную по частоте пульсацию частотной характеристики в полосе задержания. Кроме того, проявляется гладкий монотонный спад АЧХ к полосе задержания.

Фильтры Чебышева II имеют то же самое преимущество по сравнению с фильтрами Баттерворта, которое имеют фильтры Чебышева, – более крутой переход между полосой пропускания и полосой задержания у фильтров более низкого порядка, отличаются меньшей абсолютной ошибкой и большей скоростью выполнения операций.

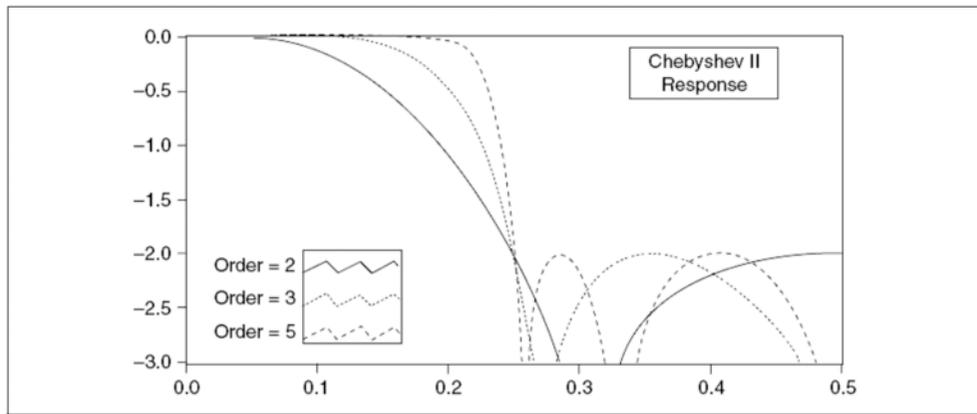


Рис. 4.16. Частотные характеристики ФНЧ Чебышева II для разных порядков фильтров

#### 4.7.10. Эллиптические фильтры

Эллиптические фильтры имеют следующие характеристики:

- минимизация пиковой ошибки в полосе пропускания и в полосе задержания;
- неизменные по величине пульсации в полосе пропускания и в полосе задержания.

По сравнению с фильтрами Баттерворта того же порядка или фильтрами Чебышева эллиптические фильтры обеспечивают самый крутой переход между полосой пропускания и полосой задержания, который объясняет их широкое распространение в практике.

На рис. 4.17 показаны частотные характеристики эллиптического ФНЧ для разных порядков фильтра.

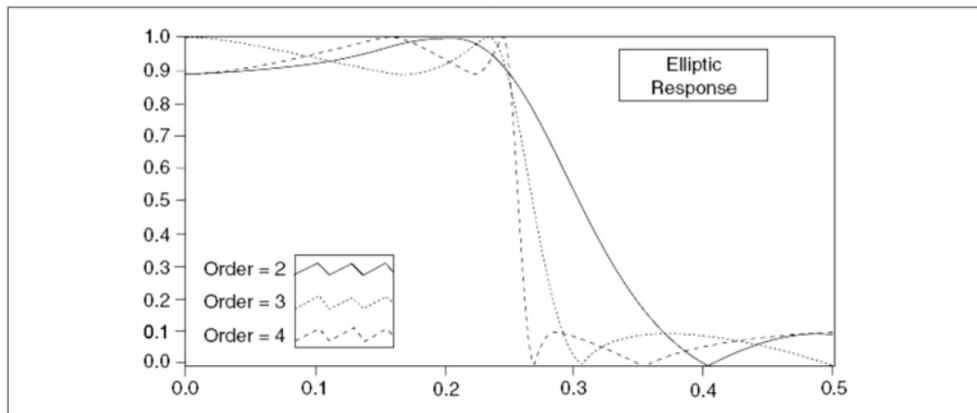


Рис. 4.17. Частотные характеристики эллиптического ФНЧ для его различных порядков

На рис. 4.17 одна и та же максимальная допустимая ошибка ограничивает пульсации и в полосе пропускания, и в полосе задержания. В этом случае даже эллиптические фильтры младшего порядка имеют крутой переход.

#### 4.7.11. Фильтры Бесселя

Фильтры Бесселя имеют следующие характеристики:

- максимально плоскую частотную характеристику и по амплитуде, и по фазе;
- почти линейную фазовую характеристику в полосе пропускания.

Можно использовать фильтры Бесселя, чтобы уменьшать искажения, обусловленные нелинейностью ФЧХ, свойственное всем БИХ-фильтрам. Старшие по порядку БИХ-фильтры и БИХ-фильтры с крутым спадом имеют явно выраженное искажение линейности ФЧХ, особенно в областях перехода АЧХ. Можно также получить линейную ФЧХ секции с БИХ-фильтрами.

На рис. 4.18 показаны АЧХ и ФЧХ ФНЧ Бесселя. АЧХ гладкая и монотонно убывает на всех частотах.

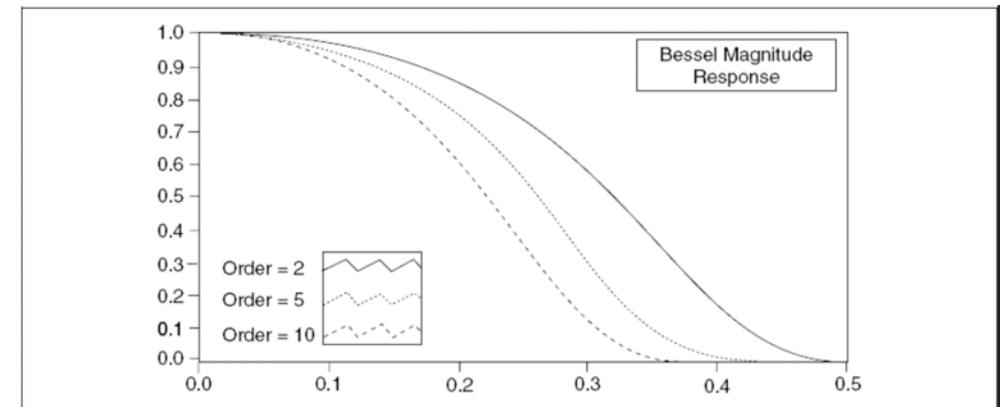


Рис. 4.18. Частотные характеристики ФНЧ Бесселя для разных порядков фильтров

На рис. 4.19 показаны фазочастотные характеристики ФНЧ Бесселя. В полосе пропускания ФЧХ почти линейные.

Подобно фильтрам Баттерворта, фильтры Бесселя требуют, чтобы старшие по порядку фильтры минимизировали пиковую ошибку, чем объясняется их ограниченное использование.

#### 4.7.12. Проектирование БИХ-фильтров

Для выбора БИХ-фильтра необходимо знать его характеристики. На рис. 4.20 показана блок-диаграмма виртуального прибора, который формирует частотную характеристику БИХ-фильтра.

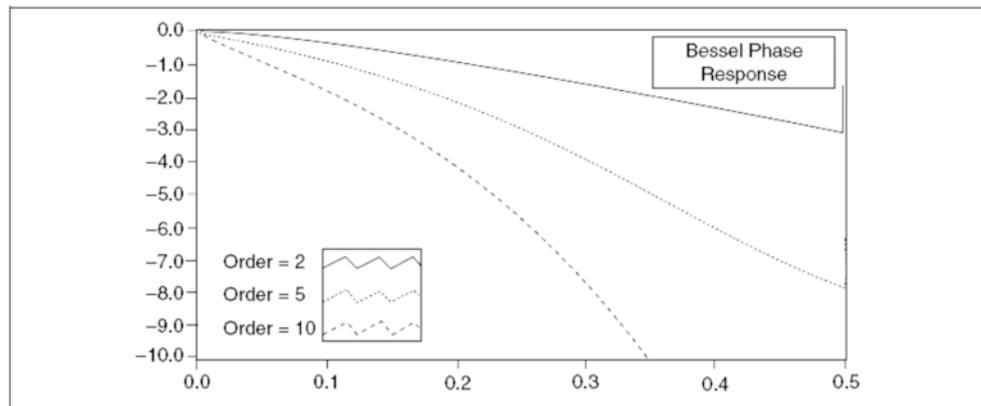


Рис. 4.19. Фазовые характеристики ФНЧ Бесселя для разных порядков фильтров

Поскольку одна и та же математическая теория используется при разработке БИХ- и КИХ-фильтров, блок-диаграмма виртуального прибора, показанная на рис. 4.20, которая формирует частотную характеристику БИХ-фильтра, и блок-диаграмма виртуального прибора на рис. 4.7, которая формирует частотную характеристику КИХ-фильтра, имеют общие модули. Главное различие между двумя виртуальными приборами – это то, что структура, показанная слева на рис. 4.20, определяет БИХ-проект фильтра и тип фильтра вместо того, чтобы определять только его тип. Виртуальный прибор с блок-диаграммой, показанной на рис. 4.20, вычисляет частотные характеристики БИХ-фильтра с помощью таких же операций, как и ВП, приведенные в этом подразделе.

На рис. 4.21 показаны АЧХ и ФЧХ эллиптического БИХ-ПФ.

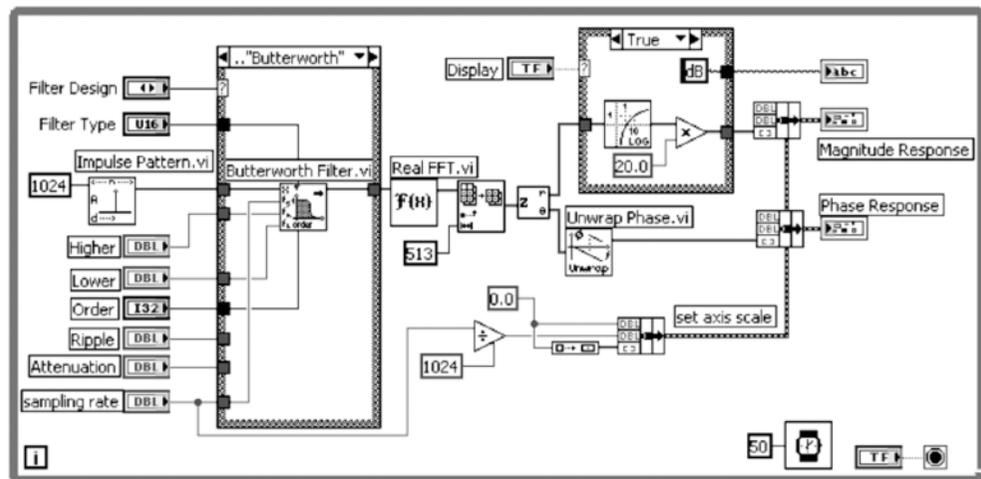


Рис. 4.20. Блок-диаграмма БИХ-фильтра

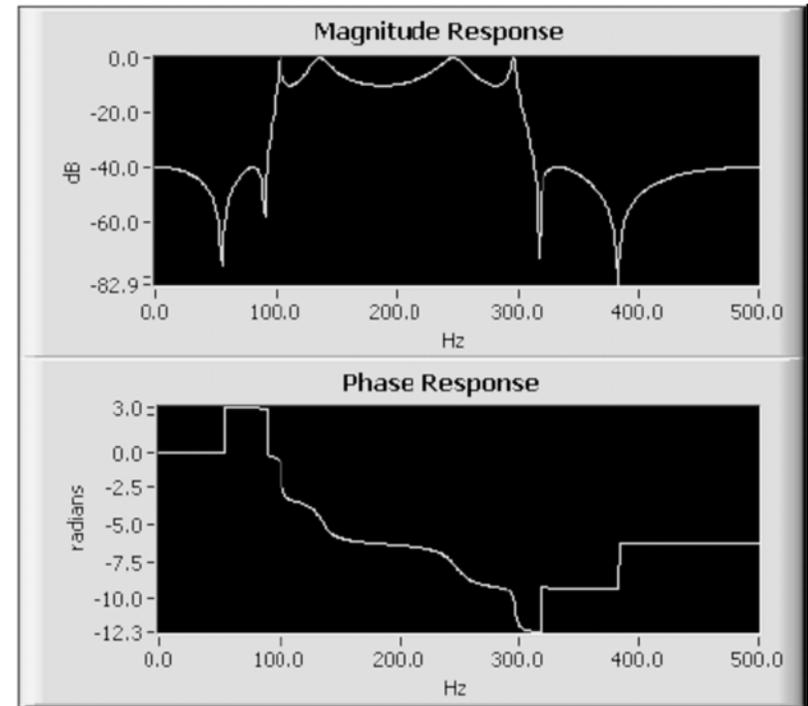


Рис. 4.21. АЧХ и ФЧХ эллиптического БИХ-ПФ

На рис. 4.21 ФЧХ фильтра явно нелинейная. Решая, использовать БИХ- или КИХ-фильтр, чтобы обработать данные, нужно помнить, что БИХ-фильтры обеспечивают нелинейную ФЧХ.

### 4.7.13. Характеристики БИХ-фильтра в LabVIEW

БИХ-фильтры в LabVIEW имеют следующие особенности:

- интерпретируют величины в отрицательных индексах в уравнении (4.10), так как ноль устанавливается в первый раз, когда вызывается виртуальный прибор;
- переходные характеристики показывают задержку, пропорциональную порядку фильтра, перед тем как они достигнут устойчивого состояния;
- число отсчетов в отфильтрованной последовательности равняется числу отсчетов во входной последовательности;
- фильтр сохраняет внутренние установки своего состояния, когда процесс фильтрации закончился.

### 4.7.14. Переходный отклик

Переходный отклик после запуска виртуального прибора наблюдается из-за того, что начальное состояние фильтра нулевое или есть отсчеты с отрицательными индексами. Продолжительность переходного отклика зависит от типа фильтра.

Продолжительность переходного отклика, выраженная в числе отсчетов, для ФНЧ и ФВЧ равняется порядку фильтра:

задержка = порядок.

Продолжительность переходного отклика, выраженная в числе отсчетов, для ПФ и ЗФ равняется удвоенному порядку фильтра:

задержка =  $2 \times$  порядок.

Можно устранить переходный отклик виртуального прибора БИХ-фильтра (IIR filter VI), обеспечивая установку памяти. Чтобы обеспечивать установку памяти для непрерывной фильтрации, надо подсоединить с помощью проводника клемму TRUE к входу `init/cont` ВП БИХ-фильтра.

На рис. 4.22 показан переходный отклик и устойчивое состояние БИХ-фильтра.

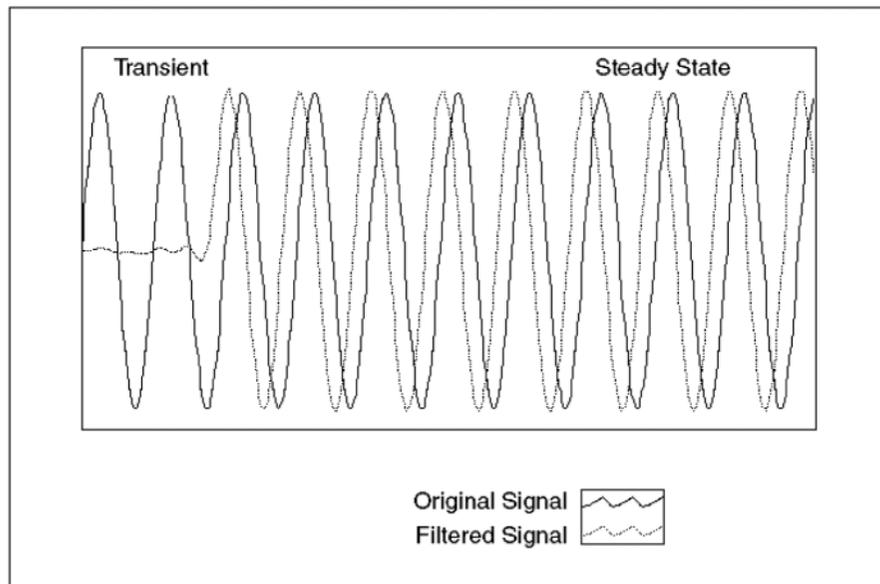


Рис. 4.22. Переходный отклик и устойчивое состояние БИХ-фильтра

## 4.8. Сравнение КИХ- и БИХ-фильтров

Поскольку проектирование цифровых фильтров основано на компромиссах для обеспечения желательной характеристики фильтра, результаты сравнения КИХ- и БИХ-фильтров могут помочь произвести отбор соответствующего проекта фильтра для его специфического применения.

БИХ-фильтры могут обеспечить тот же самый уровень ослабления, что и КИХ-фильтры, но с гораздо меньшим числом коэффициентов. Поэтому БИХ-фильтр может обеспечить фильтрацию значительно быстрее и эффективнее, чем КИХ-фильтр.

Можно проектировать КИХ-фильтры так, чтобы обеспечить линейность ФЧХ. В то же время БИХ-фильтры обеспечивают нелинейную ФЧХ. КИХ-фильтры возможно использовать тогда, когда требуются линейные ФЧХ. Можно применять БИХ-фильтры тогда, когда не требуется информация об ФЧХ, таких, например, как формирование сигнала для контроля и управления объектом.

## 4.9. Нелинейные фильтры

Окна сглаживания, БИХ-фильтры и КИХ-фильтры линейны, потому что они удовлетворяют принципу суперпозиции и принципу пропорциональности:

$$L\{ax(t) + by(t)\} = aL\{x(t)\} + bL\{y(t)\}, \quad (4.14)$$

где  $a$  и  $b$  – константы;  $x(t)$  и  $y(t)$  – сигналы;  $L\{\bullet\}$  – линейная операция фильтрации, причем связь входов и выходов осуществляется с помощью операции свертки, как показано в уравнениях (4.9) и (4.11).

Нелинейный фильтр не удовлетворяет уравнению (4.14). Также нельзя получить выходные сигналы нелинейного фильтра с помощью операции свертки, потому что набор коэффициентов не может характеризовать импульсную характеристику фильтра. Нелинейные фильтры обеспечивают специфические характеристики фильтрации, которые трудно получить, используя линейные методы фильтрации.

Медианный фильтр (Median Filter VI) есть нелинейный фильтр, имеющий одновременно низкочастотные и высокочастотные характеристики. Характеристики ФНЧ позволяют медианному фильтру удалять высокочастотный шум. Кроме того, высокочастотные характеристики позволяют медианному фильтру обнаруживать грани, которые представляют граничную информацию.

### Пример. Анализ при помощи медианного фильтра импульса на фоне шума

Виртуальные приборы «Параметры импульса» (Pulse Parameters VIs) анализируют входную последовательность для импульса, копируют и определяют лучший набор параметров импульса для его описания. После того как эти виртуальные приборы заканчивают модальный анализ, чтобы определить основание и вершину входной последовательности, различение шума и сигнала становится трудной операцией без дополнительного количества информации. Поэтому, чтобы точно определить параметры импульса, пиковое значение шумовой части входной последовательности должно быть меньше или равно 50% ожидаемой амплитуды импульса. В некоторых практических применениях 50%-ного отношения амплитуды импульса к возможному пиковому значению шума трудно достигнуть. Достижение необходимого отношения амплитуды импульса к пику шума требуется, чтобы с помощью операции предварительной обработки была извлечена информация об импульсе.

Если импульс скрыт в шуме, ожидаемое пиковое значение которого превышает 50% ожидаемой амплитуды импульса, можно использовать ФНЧ, чтобы удалить часть нежелательного шума. Однако фильтр также задерживает сигнал во времени и сглаживает фронты импульса, которые содержат высокочастотную информацию. Виртуальный прибор «Медианный фильтр» (Median Filter VI) может извлечь импульс более эффективно, чем ФНЧ, потому что медианный фильтр удаляет высокочастотный шум при сохранении информации о фронтах импульса.

На рис. 4.23 показана блок-диаграмма виртуального прибора, который генерирует и анализирует импульс на фоне шума.

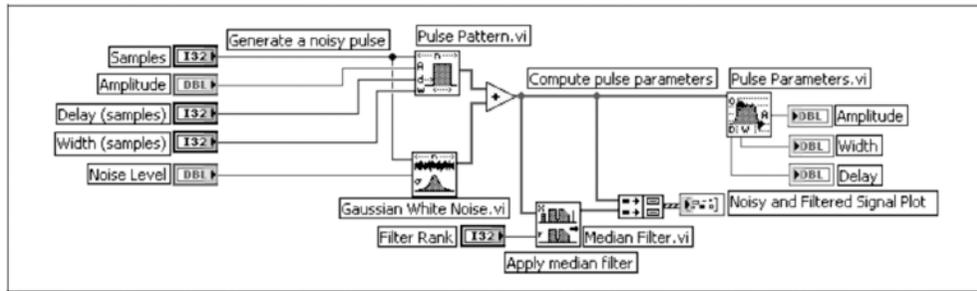


Рис. 4.23. Использование медианного фильтра (Median Filter.vi) для извлечения информации об импульсе

Виртуальный прибор на рис. 4.23 генерирует импульс на фоне шума с ожидаемым пиковым значением, превышающим 100% ожидаемой амплитуды импульса. Сигнал, который генерирует этот виртуальный прибор, имеет следующие идеальные параметры импульса:

- амплитуда – 5,0 В;
- задержка – 64 выборки;
- длительность – 32 выборки.

На рис. 4.24 представлен импульс на фоне шума, отфильтрованный импульс и оцененный импульс, параметры которого измерены с помощью виртуального прибора с блок-диаграммой, показанной на рис. 4.23.

На рис. 4.24 можно проследить сигнал импульса, выделенный медианным фильтром, даже в том случае, когда среднее квадратическое значение шума превышает пиковое значение импульса.

Можно удалить высокочастотный шум с помощью медианного фильтра (Median Filter.vi), чтобы достигнуть 50%-ного отношения пикового значения импульса к ожидаемому максимуму шума, а виртуальный измеритель параметров импульса (Pulse Parameters VI) должен завершить анализ с предельной точностью.

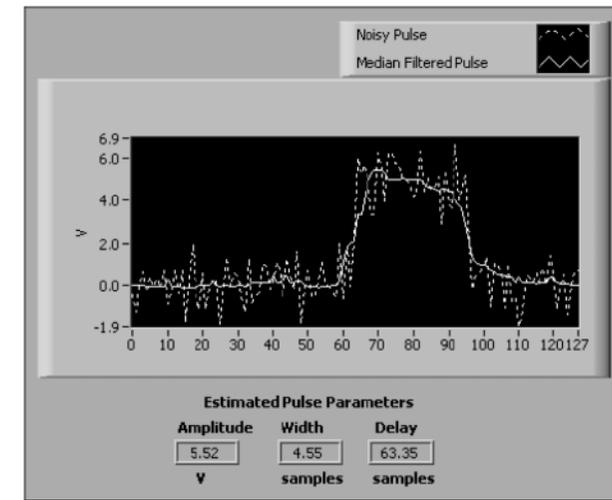


Рис. 4.24. Импульс на фоне шума и отфильтрованный в медианном фильтре (Median Filter.vi) импульс

## 4.10. Выбор проекта цифрового фильтра

Необходимо ответить на следующие вопросы, чтобы обоснованно выбрать фильтр:

- требуется ли для решения задачи анализа линейная ФЧХ?
- могут ли условия задачи анализа допустить пульсации частотной характеристики?
- требует ли задача анализа узкой полосы перехода?

Рис 4.25 можно использовать как руководство, чтобы выбрать соответствующий фильтр для анализа в приложениях обработки сигналов.

На рис. 4.25 использованы следующие обозначения: Linear phase – линейная ФЧХ; Ripple acceptable – приемлемые пульсации АЧХ; Narrow transition band – узкая полоса перехода; Narrowest possible transition region? – нужна ли узкая полоса перехода; Ripple in Passband – пульсации АЧХ в полосе пропускания; Ripple in Stopband – пульсации АЧХ в полосе задержания; Multiband filter specifications – многополосные параметры фильтра.

Однако может возникнуть необходимость в экспериментировании с несколькими типами фильтров, чтобы найти лучший из них для конкретной задачи.

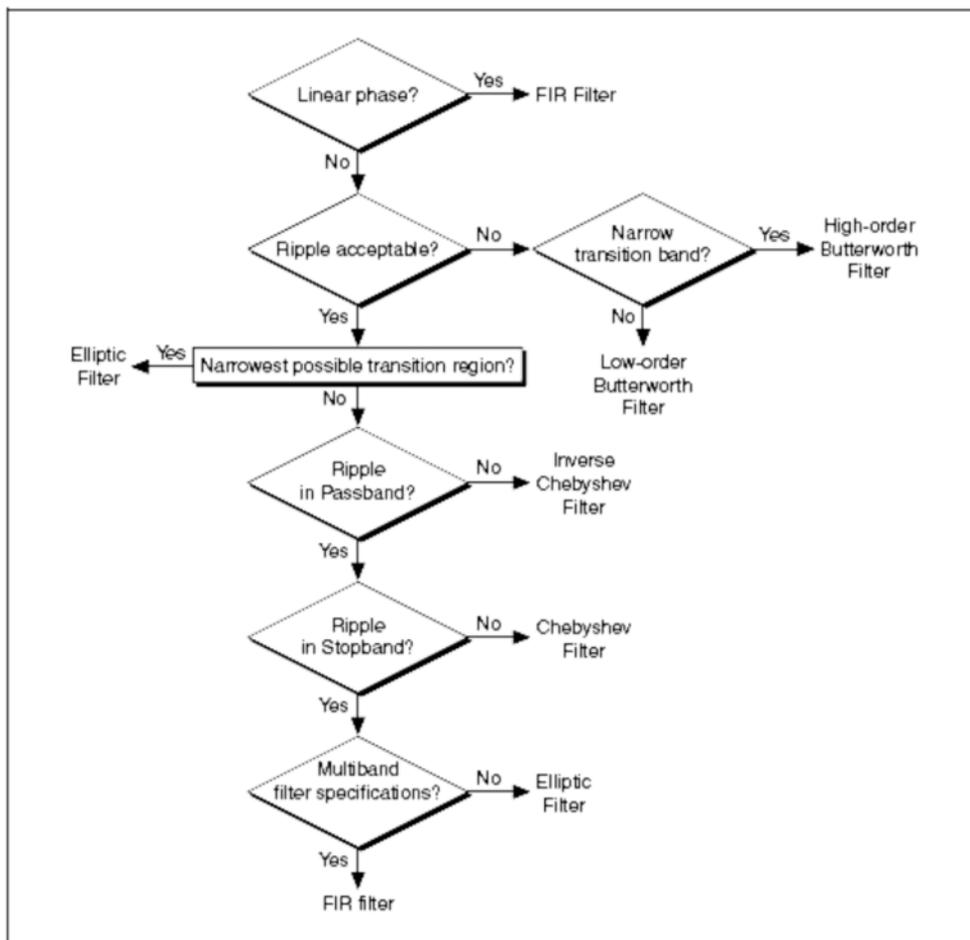


Рис. 4.25. Структурная схема для выбора типа фильтра

## Спектральный анализ

5

В этом разделе представлены основные принципы дискретного преобразования Фурье (ДПФ) (Discrete Fourier Transform) (DFT), быстрого преобразования Фурье (БПФ) (Fast Fourier Transform) (FFT), основы анализа видеосигнала, вычисления, выполняемые для определения спектра мощности, и использование функции базового БПФ (FFT-based) для сетевого измерения спектра. Можно использовать систему поиска NI Example Finder, чтобы найти примеры использования ВП цифровой обработки сигналов и ВП анализа измерения и выполнить БПФ для спектрального анализа сигналов.

### 5.1. Различия между частотной областью и временной областью

Представление сигнала в области (domain) времени дает совокупность его мгновенных значений в интервале времени, в течение которого была получена выборка. Однако во многих случаях необходимо знать гармонические составляющие сигнала, а не значения единичных отсчетов в выборке.

Теорема Фурье устанавливает, что периодический сигнал любой формы в области времени может быть представлен взвешенной суммой синусов и косинусов с частотами, определяемыми периодом анализируемого сигнала. Тот же самый сигнал может быть представлен в области частот как пара: амплитуда и значение начальной фазы для каждой гармонической составляющей.

Можно синтезировать любую форму сигнала, добавляя гармонические составляющие, каждую со специфической амплитудой и начальной фазой. На рис. 5.1 показана оригинальная форма сигнала, обозначенного как *Sum*, и его гармонические составляющие. Частота основной (первой) гармонической составляющей (гармоники) равна  $f_0$ , вторая гармоника имеет частоту  $2f_0$ , и третья гармоника – частоту  $3f_0$ .

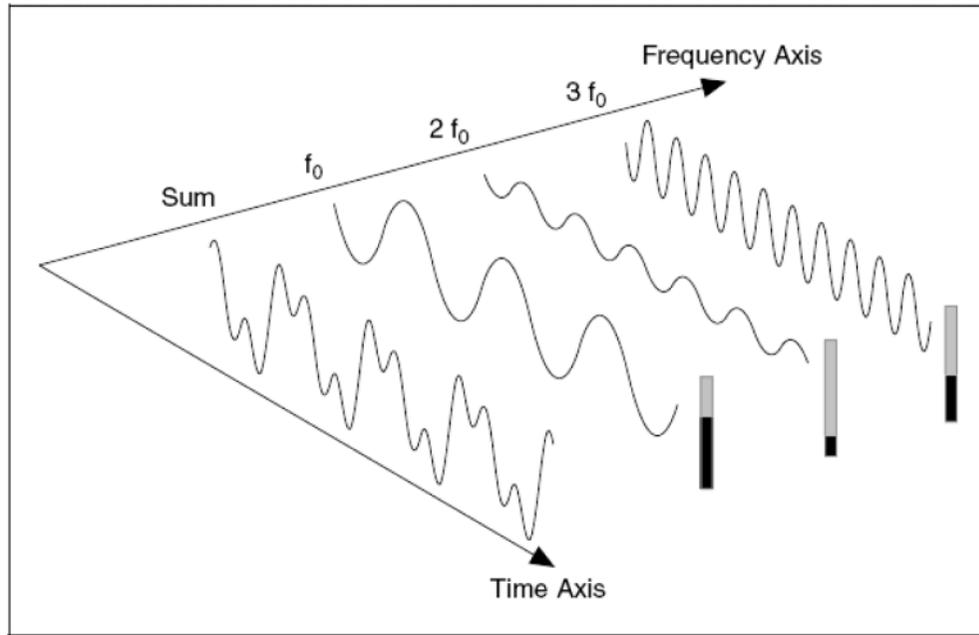


Рис. 5.1. Сигнал, сформированный сложением трех гармонических составляющих

В частотной области можно сформировать концептуально гармонические составляющие, которые необходимо сложить, чтобы синтезировать сложный по форме сигнал в области времени. На рис. 5.1 представлены простые гармонические составляющие, которые являются функциями времени и являются отдельными линиями в частотной области. Высота каждой линии есть амплитуда колебания во времени для этой гармонической составляющей. Представление сигнала в виде индивидуальных гармонических составляющих есть его представление в частотной области, которое может дать больше представления о сигнале и системе, в которой он был сгенерирован.

Выборки сигнала, полученные от DAQ-устройства (data acquisition – процесс получения цифровых данных или от АЦП, или от разъема ввода цифровых данных), составляют представление сигнала в области времени. Некоторые измерения, такие как оценка гармонических искажений, являются трудными для количественных определений на основе исследований формы сигнала как функции времени на экране осциллографа. Когда тот же самый сигнал отображен в области частот с помощью анализатора БПФ (Analyzer FFT), также известного как «Динамический анализатор сигнала» (Dynamic Signal Analyzer), можно легко измерить частоты и амплитуды гармоник и оценить, например, коэффициент нелинейных искажений усилителя.

## 5.2. Отношения Парсеваля

Теорема Парсеваля (для сигналов с ограниченной энергией) определяет, что полная энергия, вычисленная в области времени, должна равняться полной энергии, вычисленной в области частот. Это есть условие сохранения энергии. Следующее уравнение определяет непрерывную форму отношений Парсеваля:

$$\int_{-\infty}^{\infty} x(t)x^*(t)dt = \int_{-\infty}^{\infty} |X(f)|^2 df,$$

где  $x^*(t)$  – сигнал, комплексно-сопряженный с исходным сигналом  $x(t)$ ;  $X(f)$  – спектральная плотность сигнала.

Следующее уравнение определяет дискретную форму отношений Парсеваля:

$$\sum_{i=0}^{n-1} |x_i|^2 = \frac{1}{n} \sum_{k=0}^{n-1} |X_k|^2, \quad (5.1)$$

где  $x_i \leftrightarrow X_k$  – дискретная ДПФ (БПФ)-пара;  $n$  – число элементов в последовательности.

На рис. 5.2 показана блок-диаграмма ВП, который демонстрирует отношения Парсеваля.

Виртуальный прибор на рис. 5.2 производит реальную входную последовательность. Верхнее ответвление на блок-диаграмме обеспечивает вычисление энергии сигнала в области времени с учетом левой части уравнения (5.1). Нижнее ответвление на блок-диаграмме дает возможность преобразовывать сигнал из области времени в область частот и вычислять энергию сигнала в области частот с учетом правой части уравнения (5.1).

На рис. 5.3 представлены результаты вывода результатов расчетов на лицевую панель ВП, блок-диаграмма которого показана на рис. 5.2. Общее вычисление энергии в области времени равняется результату вычисления энергии в области частоты.

## 5.3. Преобразование Фурье

Преобразование Фурье (для сигналов с ограниченной энергией) используется в методе исследования сигналов в терминах области частот. Самые обычные приложения преобразования Фурье – анализ линейных, инвариантных ко времени систем и спектральный анализ сигналов.

Следующее уравнение определяет двухстороннее прямое преобразование Фурье, результатом которого является нахождение спектральной плотности  $X(f)$ :

$$X(f) = F\{x(t)\} = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft} dt.$$

Двухстороннее обратное преобразование Фурье определяется следующим выражением:

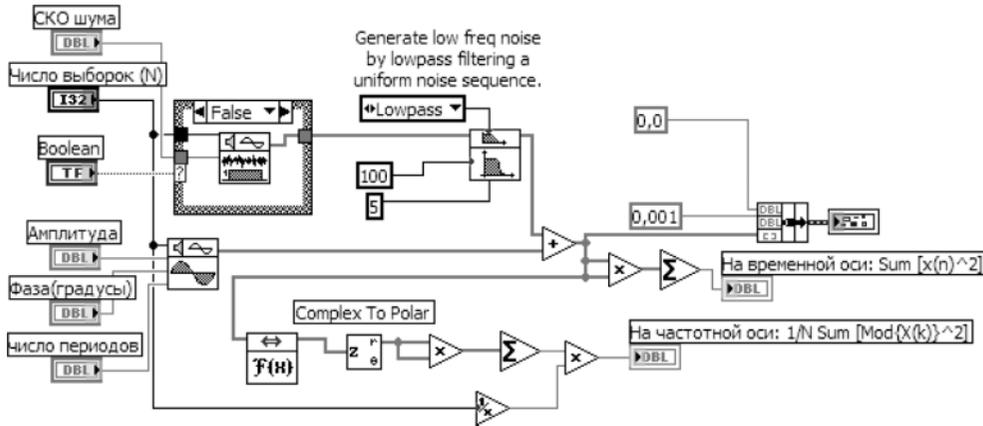


Рис. 5.2. ВП, иллюстрирующий теорему Парсеваля

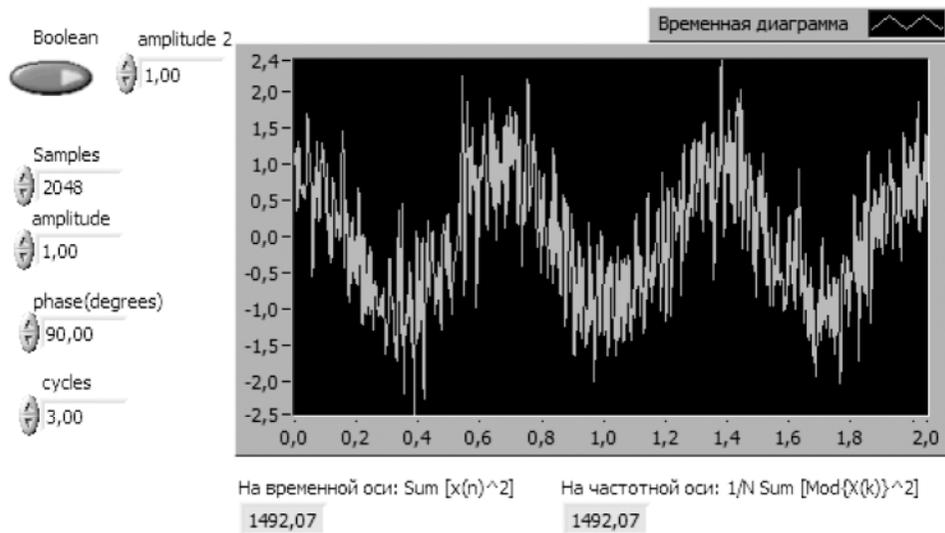


Рис. 5.3. Результаты установки параметров и вычислений по теореме Парсеваля на лицевой панели ВП

$$x(t) = F^{-1}\{X(f)\} = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft} df.$$

Понятие «двухсторонний» означает, что математическая реализация прямого и обратного преобразований Фурье использует все отрицательные и положительные частоты и длительность действия сигнала. В то же время понятие «односторонний» означает, что математическая реализация преобразования связывает

только положительные частоты и зависимость мгновенных значений сигнала от времени. При этом различают физический (односторонний) и математический (двухсторонний) спектры.

Пара преобразований Фурье состоит в представлении сигнала и в области частот, и в области времени. Следующее отношение обычно символически обозначает пару преобразований Фурье:

$$x(t) \Leftrightarrow X(f).$$

## 5.4. Дискретное преобразование Фурье

Алгоритм, который обеспечивает преобразование выборки сигнала из области времени в область частот, является дискретным преобразованием Фурье (ДПФ). Необходимо отметить, что ДПФ по умолчанию исходный ограниченный во времени сигнал превращает в периодический с периодом, равным времени получения выборки. Фактически исходный сигнал представляется в виде ряда Фурье: набора гармоник с частотами, кратными частоте первой гармоники, и амплитудами, определяемыми с помощью преобразования Фурье исходного сигнала в пределах его периода. ДПФ широко используется в спектральном анализе, медицине, при численном анализе, в прикладной механике, акустике и т.д. На рис. 5.4 показан результат использования ДПФ (DFT) для преобразования данных из области времени (Time Domain) в область частот (Frequency Domain).

Предположим, что получена выборка сигнала из  $N$  единичных измерений (отсчетов) от DAQ-устройства. Если применяется ДПФ к  $N$  единичным измерениям, представляющим сигнал в области времени, результат преобразования также состоит из  $N$  единичных измерений (отсчетов), но информация, которую они содержат, имеет представление в области частот.

### 5.4.1. Отношения между выборками из $N$ отсчетов в области частот и в области времени

Если выборка сигнала произведена на заданной частоте дискретизации, уравнение (5.2) определяет интервал  $\Delta t$  времени между отсчетами в выборке:

$$\Delta t = \frac{1}{f_s}, \tag{5.2}$$

где  $f_s$  – частота дискретизации, т. е. число отсчетов в секунду (S/s).

Интервал между отсчетами в выборке соответствует наименьшей частоте, которую система может реализовать через ДПФ или связанные подпрограммы.

ДПФ определяется следующим выражением, в котором отсчеты по времени преобразуются в отсчеты в области частот:

$$X[k] = \sum_{i=0}^{N-1} x[i] e^{-j2\pi ik/N} \text{ для } k = 0, 1, 2, \dots, N-1, \tag{5.3}$$

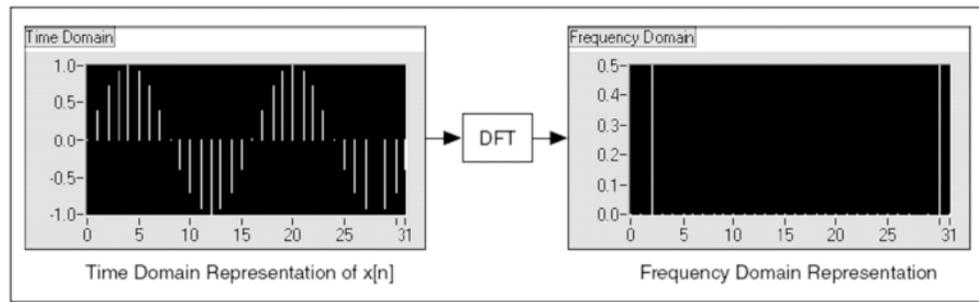


Рис. 5.4. Дискретное преобразование Фурье

где  $x[i]$  – представление выборки сигнала в области времени;  $N$  – общее число отсчетов в выборке. Необходимо обратить внимание, что, как отмечалось выше, и область времени  $x$ , и вектор  $X$  в области частоты имеют общее число отсчетов  $N$ .

Подобный интервалу времени  $\Delta t$  между выборками  $x$  в области времени, имеется интервал по частоте, или разрешающая способность по частоте, между компонентами  $X$  в области частот, который определяется уравнением

$$\Delta f = \frac{f_s}{N} = \frac{1}{N\Delta t}, \quad (5.4)$$

где  $\Delta f$  – разрешающая способность по частоте;  $f_s$  – частота дискретизации;  $N$  – число отсчетов выборки;  $N\Delta t$  – полное время получения выборки.

Чтобы улучшить разрешающую способность по частоте, т. е. уменьшить  $\Delta f$ , необходимо увеличить  $N$  и сохранить постоянной или уменьшить  $f_s$ , а также сохранить постоянным  $N$ . Оба подхода эквивалентны увеличению длительности  $N\Delta t$ , которая является продолжительностью выборки.

### 5.4.2. Пример ДПФ

Здесь представлен пример использования уравнения (5.3) для ДПФ постоянного напряжения (сигнала в виде постоянной составляющей). Используются следующие предположения:

- $X[0]$  соответствует компоненте постоянного напряжения, или среднему значению, сигнала;
- сигнал постоянного напряжения имеет неизменную во времени величину +1 В;
- число отсчетов в выборке – 4;
- каждый из отсчетов имеет значение +1, как показано на рис. 5.5;
- результирующая временная последовательность для выборки из четырех отсчетов дается уравнением  $x[0] = x[1] = x[2] = x[3] = 1$ .

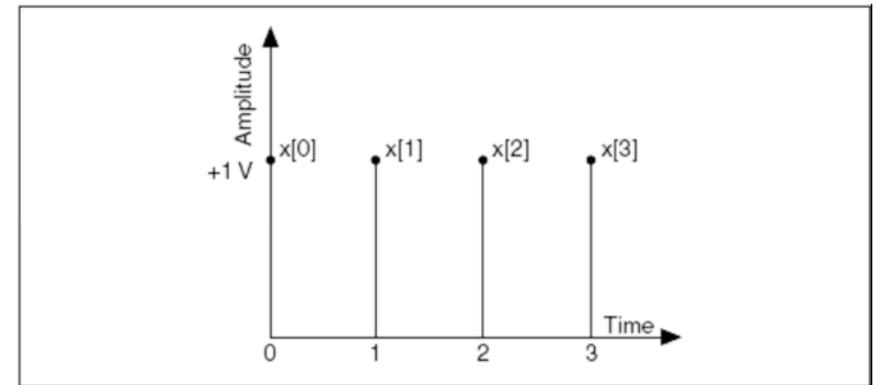


Рис. 5.5. Последовательность временных выборок для ДПФ

При вычислении ДПФ используют тождество Эйлера, которое задается следующим выражением:

$$\exp(-j\theta) = \cos(\theta) - j\sin(\theta).$$

Если использовать уравнение (5.3), чтобы выполнить ДПФ последовательности, показанной на рис. 5.5, и применить к нему тождество Эйлера, можно получить следующие уравнения:

$$X[0] = \sum_{i=0}^{N-1} x_i e^{-j2\pi i 0 / N} = x[0] + x[1] + x[2] + x[3] = 4;$$

$$X[1] = x[0] + x[1] \left( \cos\left(\frac{\pi}{2}\right) - j\sin\left(\frac{\pi}{2}\right) \right) + x[2](\cos(\pi) - j\sin(\pi)) +$$

$$+ x[3] \left( \cos\left(\frac{3\pi}{2}\right) - j\sin\left(\frac{3\pi}{2}\right) \right) = (1 - j - 1 + j) = 0;$$

$$X[2] = x[0] + x[1](\cos(\pi) - j\sin(\pi)) + x[2](\cos(2\pi) - j\sin(2\pi)) +$$

$$+ x[3](\cos(3\pi) - j\sin(3\pi)) = (1 - 1 + 1 - 1) = 0,$$

где  $X[0]$  – компонента постоянного напряжения;  $N$  – число отсчетов в выборке.

Кроме отсчетов постоянного напряжения в дискретные моменты все другие значения на оси времени, как показано на рис. 5.5, являются нулевыми. Однако расчетное значение  $X[0]$  зависит от значения  $N$ . Поскольку в этом примере  $N = 4$ , то и  $X[0] = 4$ . Аналогично, если  $N = 10$ , вычисление приводит к  $X[0] = 10$ . Эта зависимость  $X[\cdot]$  от  $N$  также выполняется для других частотных компонент. Поэтому обычно производится деление результата ДПФ на  $N$ , чтобы получить правильную величину амплитуд спектральных составляющих.

### 5.4.3. Информация об амплитуде и фазе

$N$  выборок входного сигнала приводят к  $N$  выборкам ДПФ. Уравнение (5.3) показывает, что независимо от того, реальный входной сигнал  $x[i]$  или комплексный, значение  $X[k]$  всегда является комплексным, хотя его мнимая часть может быть и нулевой. Другими словами, каждая частотная компонента имеет амплитуду и начальную фазу. Есть два вида графического представления результатов спектрального анализа: амплитудная спектральная диаграмма и фазовая спектральная диаграмма.

Обычно отображается в виде графика амплитудная спектральная диаграмма. Амплитуда есть квадратный корень из суммы квадратов реальных и мнимых частей комплексной амплитуды.

Начальная фаза отсчитывается относительно начала отсчета времени или относительно колебания по закону косинуса первой гармоники, начинающейся в начале отсчета времени. Измерения начальной фазы первой гармоники устойчивы, только если входной сигнал имеет синхронизированный (triggered) вид. Двухканальные измерения фазы определяют разность фаз между каналами, если каналы синхронизированы. В дополнительной синхронизации в каждом канале обычно нет необходимости.

Начальная фаза есть арктангенс отношения мнимой и вещественной частей комплексной амплитуды и обычно соответствует интервалу между  $\pi$  и  $-\pi$  радиан, или  $180^\circ$  и  $-180^\circ$ . Это обстоятельство приводит к неоднозначности в представлениях фазовой спектральной диаграммы

Для реальных сигналов ( $x[i]$  – реальный сигнал для всех  $i$ ), подобных тем, которые получают на выходе одного канала DAQ-устройства, ДПФ симметрично и его свойства задаются следующими уравнениями:

$$|X[k]| = |X[N - k]|;$$

начальная фаза ( $X[k]$ ) = минус начальная фаза ( $X[N - k]$ ).

Амплитуда  $X[k]$  – четная симметричная, а фаза ( $X[k]$ ) – нечетная симметричная. Четный симметричный сигнал симметричен относительно оси  $Y$ , и нечетный симметричный сигнал симметричен относительно начала координат. На рис. 5.6 показаны четная и нечетная симметрии.

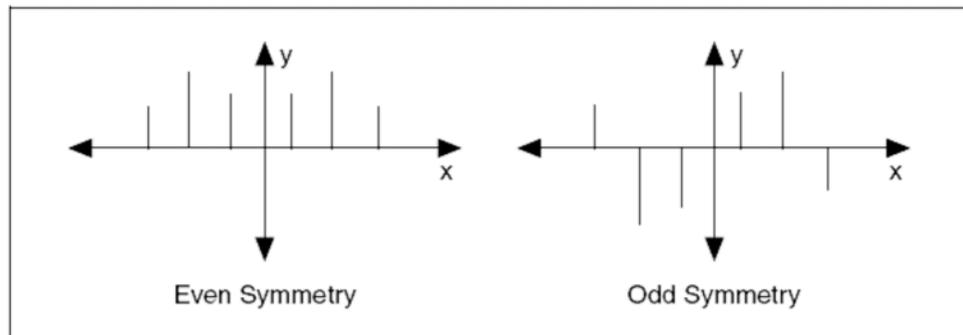


Рис. 5.6. Симметрия сигнала (Odd – нечетная, Even – четная)

По причине таких симметрий  $N$  выборок ДПФ содержат повторную информацию. Из-за этого повторения информации можно вычислить или отобразить только половину выборок дискретного преобразования Фурье, так как другую половину можно получить, используя это повторение. Если входной сигнал сложен, ДПФ несимметрично, и нельзя использовать одну половину выборок, чтобы получить другую половину.

### 5.4.4. Частотный интервал между выборками ДПФ

Если интервал между отсчетами в выборке по оси времени есть  $\Delta t$  и первый отсчет в выборке данных ( $k = 0$ ) соответствует 0,  $k$ -й отсчет в выборке данных, где  $k > 0$ , есть целое число, соответствующее моменту  $k\Delta t$ . Точно так же, если разрешающая способность по частоте есть  $\Delta f$ , то  $k$ -й отсчет в выборке ДПФ соответствует точке  $k\Delta f$  на оси частот. Однако это допустимо только для первой половины частотных компонент. Другая половина представляет собой компоненты отрицательных частот.

В зависимости от того, является ли число  $N$  отсчетов в выборе четным или нечетным, можно иметь различную интерпретацию частоты, соответствующей  $k$ -му отсчету в выборке ДПФ. Например, если  $N = 8$ , то  $p$  представляет собой индекс частоты Найквиста:  $p = N/2 = 4$ .

В табл. 5.1 приведены целые числа  $\Delta f$ , которым соответствует каждый элемент комплексной выводимой последовательности  $X$ .

Таблица 5.1.  $X[p]$  для  $N = 8$

$X[p]$	$\Delta f$
$X[0]$	Постоянный ток
$X[1]$	$\Delta f$
$X[2]$	$2\Delta f$
$X[3]$	$3\Delta f$
$X[4]$	$4\Delta f$ (частота Найквиста)
$X[5]$	$-3\Delta f$
$X[6]$	$-2\Delta f$
$X[7]$	$-\Delta f$

Отрицательные величины во втором столбце за пределами частоты Найквиста представляют собой отрицательные частоты, т. е. элементы со значениями индексов больше  $p$ .

Для  $N = 8$   $X[1]$  и  $X[7]$ ,  $X[2]$  и  $X[6]$ , а также  $X[3]$  и  $X[5]$  имеют одинаковую величину. Различие в том, что  $X[1]$ ,  $X[2]$  и  $X[3]$  соответствуют положительным (positive) компонентам частоты, в то время как  $X[5]$ ,  $X[6]$  и  $X[7]$  соответствуют отрицательным (negative) компонентам частоты,  $X[4]$  соответствует частоте Найквиста (табл. 5.1).

Рисунок 5.7 иллюстрирует комплексную последовательность отсчетов для вектора  $X$  при  $N = 8$ .

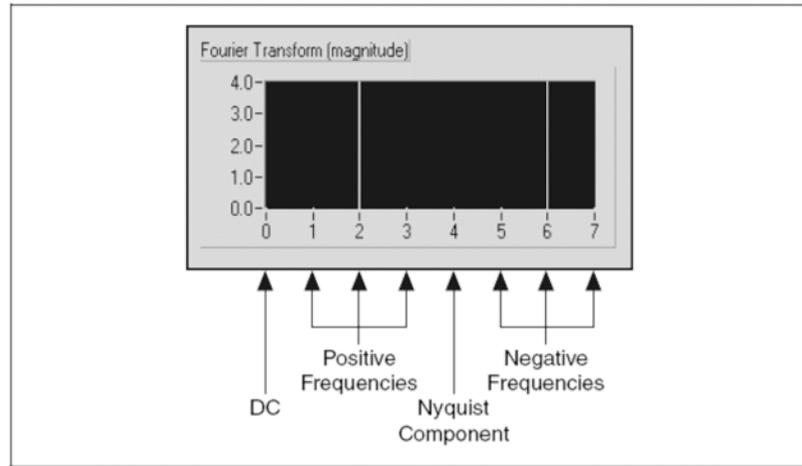


Рис. 5.7. Комплексная последовательность  $X$  для  $N = 8$

Представление спектра, в котором имеются гармонические составляющие с положительными и отрицательными частотами, есть продукт двустороннего преобразования (математический спектр).

Когда  $N$  – нечетное число, гармоническая составляющая с частотой Найквиста отсутствует в спектре. В табл. 5.2 перечислены значения составляющих при частотах, выраженных в целых числах  $\Delta f$  для  $X[p]$ , когда  $N = 7$  и  $p = (N - 1)/2 = (7 - 1)/2 = 3$ .

Таблица 5.2.  $X[p]$  для  $N = 7$

$X[p]$	$\Delta f$
$X[0]$	Постоянный ток
$X[1]$	$\Delta f$
$X[2]$	$2\Delta f$
$X[3]$	$3\Delta f$
$X[4]$	$-3\Delta f$
$X[5]$	$-2\Delta f$
$X[6]$	$-\Delta f$

Для  $N = 7$   $X[1]$  и  $X[6]$  имеют одинаковую величину;  $X[2]$  и  $X[5]$  имеют также одинаковое значение;  $X[3]$  и  $X[4]$  также равны между собой. Однако значения  $X[1]$ ,  $X[2]$  и  $X[3]$  соответствуют положительным частотам, в то время как  $X[4]$ ,  $X[5]$  и  $X[6]$  соответствуют отрицательным частотам. Поскольку число  $N$  нечетное, гармоническая составляющая с частотой Найквиста имеет нулевую амплитуду.

На рис. 5.8 показана комплексная последовательность  $X[p]$  для  $N = 7$ .

На рис. 5.8 показан также результат двустороннего преобразования Фурье (представлены положительные и отрицательные частоты).

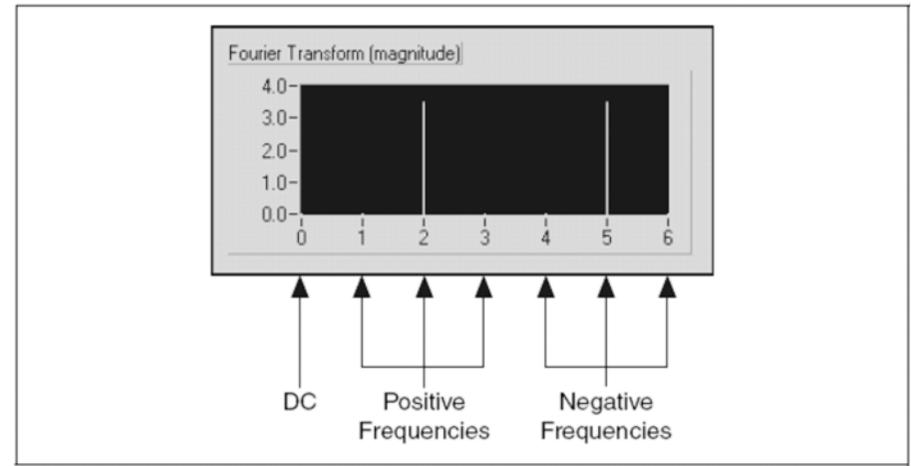


Рис. 5.8. Комплексная последовательность  $X[p]$  для  $N = 7$  с положительными (Positive) и отрицательными (Negative) частотами

## 5.5. Основные принципы БПФ

Непосредственно выполнение ДПФ на  $N$  отсчетов в выборке данных требует приблизительно  $N^2$  комплексных операций, и поэтому данный процесс занимает много времени. Быстрое преобразование Фурье представляет собой быстрый алгоритм для вычисления ДПФ. ДПФ определяется уравнением

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\left(\frac{2\pi nk}{N}\right)}$$

Следующие понятия являются базовыми функциями для основанного на БПФ анализа спектра сигнала:

- БПФ;
- спектр мощности;
- перекрестный (взаимный) (cross) спектр мощности.

Можно использовать базовые функции БПФ (FFT-based) как стандартные блоки для того, чтобы измерять дополнительные характеристики типа частотной, импульсной характеристик, а также последовательностей для амплитудного и фазового спектров.

БПФ и спектр мощности используются для того, чтобы произвести измерения частоты стационарных или нестационарных сигналов. С помощью БПФ производится усреднение частотных компонент сигнала по всей выборке. Поэтому можно использовать БПФ для анализа стационарного сигнала или в случаях, когда есть необходимость измерить только значение средней энергии в каждой точке по частоте.

БПФ эквивалентно набору параллельных фильтров с полосой пропускания  $\Delta f$ , центрированных в каждом приращении частоты от постоянной составляющей до  $(F_s / 2) - (F_s / N)$ . Поэтому частотные линии также известны как отсчеты по частоте, частотные дискреты (bins) или БПФ-дискреты.

### 5.5.1. Вычисление частотных составляющих

Каждая гармоническая составляющая есть точечная выборка сигнала, заданного в области времени с множителем в виде экспоненциального комплекса на этой частоте, которая определяется следующим выражением:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\left(\frac{2\pi nk}{N}\right)} = \sum_{n=0}^{N-1} x[n] \left[ \cos\left(\frac{2\pi nk}{N}\right) - j \sin\left(\frac{2\pi nk}{N}\right) \right].$$

Постоянная составляющая – это точечная выборка  $x(n)$  с весовым коэффициентом  $[\cos(0) - j\sin(0)]$  или 1,0.

Первый частотный дискрет, или частотная составляющая, является точечной оценкой  $x(n)$  с весовым коэффициентом  $\cos(2\pi n/N) - j\sin(2\pi n/N)$ . Здесь  $\cos(2\pi n/N)$  – отдельный период косинусоидального колебания и  $\sin(2\pi n/N)$  – отдельный период синусоидального колебания.

Вообще, отсчет с номером  $k$  – точечная выборка  $x(n)$  с  $k$  периодами косинусной составляющей для вещественной части  $X(k)$  и синусной составляющей для мнимой части  $X(k)$ .

Использование БПФ для спектрального анализа подразумевает два важных отношения.

Первое отношение связывает самую высокую по частоте спектральную составляющую, которая может быть проанализирована в частотной выборке, с частотой дискретизации. Частота этой составляющей определяется следующим уравнением:

$$F_{\max} = \frac{f_s}{2},$$

где  $F_{\max}$  – самая высокая частота в спектре, которая может быть проанализирована.

Второе отношение связывает разрешающую способность по частоте с полным временем выборки, которое связано с частотой дискретизации и размером блока БПФ:

$$\Delta f = \frac{1}{T} = \frac{f_s}{N},$$

где  $\Delta f$  – разрешающая способность по частоте;  $T$  – время измерения;  $f_s$  – частота дискретизации;  $N$  – размер блока отсчетов БПФ.

### 5.5.2. Быстродействие БПФ

Когда размер входной последовательности есть число по основанию два ( $N = 2^m$ ), можно осуществить ДПФ приблизительно с числом операций, равным  $M \log_2(N)$ .

Эти операции позволяют выполнить ДПФ намного быстрее. В литературе по цифровой обработке сигналов приведены алгоритмы для более быстрого ДПФ (как БПФ). Обычно размеры входной последовательности определяются числом по основанию 2 и составляют 512, 1024 и 2048.

Когда размер входной последовательности не определяется числом по основанию два, но может быть факторизован как произведение маленьких простых чисел, в виртуальных приборах (VIs), основанных на БПФ, используется смешанное основание системы счисления (Cooley-Tukey-алгоритм), чтобы эффективно осуществить ДПФ входной последовательности. Например, следующее уравнение определяет размер входной последовательности и устанавливает размеры  $N$  как произведение маленьких простых чисел:

$$N = 2^m 3^k 5^j \text{ для } m, k, j = 0, 1, 2, \dots \quad (5.5)$$

Для размера исходной последовательности, определенного уравнением (5.5), виртуальный прибор (VIs) базового БПФ (FFT-based) может произвести ДПФ со скоростями, сопоставимыми со скоростями выполнения БПФ, размер входной последовательности которого есть число по основанию два. Обычные размеры входных последовательностей, которые факторизуются как множители малых простых чисел, включают следующие значения: 480, 640, 1000 и 2000.

### 5.5.3. Дополнение нулями

Дополнение нулями – методика, часто используемая для того, чтобы делать входную последовательность равной числу по основанию два. В этой методике добавляются нули до конца входной последовательности так, чтобы общее количество выборок было равно следующему большему числу с основанием два. Например, если имеется 10 выборок сигнала, то можно добавить шесть нулей, чтобы сделать общее количество выборок равным 16, которое является числом по основанию два. На рис. 5.9 показано дополнение нулями 10 выборок сигнала, чтобы сделать общее количество выборок равным 16.

Добавление нулей к исходной выборке для обеспечения числа отсчетов с основанием два не улучшает основную разрешающую способность по частоте, связанную с сигналом в области времени. Единственный способ улучшить разрешающую способность по частоте для сигнала в области времени состоит в том, чтобы увеличить время получения исходной выборки или обеспечить более длинные отрезки времени между отсчетами.

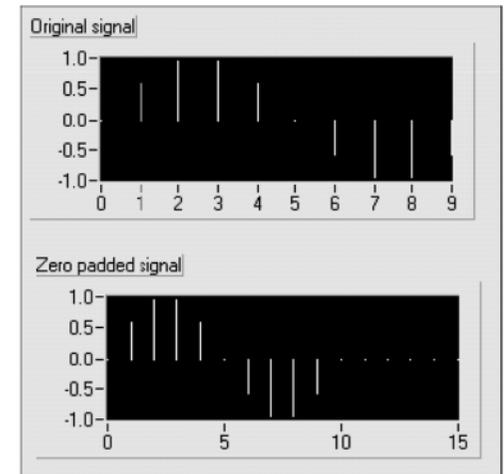


Рис. 5.9. Дополнение нулями

Дополнительно к приведению общего числа выборок к числу по основанию два так, чтобы стало возможным более быстрое вычисление на основе применения БПФ к выборке с дополнением нулями, можно интерполировать результат применения БПФ, что может обеспечить более высокую разрешающую способность по частоте на графике.

### 5.5.4. Виртуальный прибор БПФ (VI FFT)

В полиморфном виртуальном приборе БПФ (Polymorphic FFT VI) используются две разновидности БПФ: реальное БПФ (Real FFT) и комплексное БПФ (Complex FFT).

Различие между этими двумя типами преобразований состоит в том, что виртуальный прибор реального БПФ производит БПФ от сигнала с реальными отсчетами, тогда как виртуальный прибор комплексного БПФ вычисляет БПФ сигнала с комплексными отсчетами. Однако выводы для обоих видов виртуальных приборов БПФ комплексные.

Большинство реальных сигналов имеет реальные отсчеты в выборке. Поэтому можно использовать ВП реального БПФ для большинства приложений. Можно также использовать ВП комплексного БПФ, полагая отсчеты мнимой части сигнала нулевыми.

Для комплексного БПФ используются отсчеты сигнала, содержащие реальные и мнимые части.

Сигнал, состоящий из реальной и мнимой частей, встречается часто в области телекоммуникаций и радиолокации, где модулируется форма колебания показательным комплексом, а сигнал является узкополосным, что позволяет перейти к методам анализа и обработки по комплексной огибающей. Процесс умножения входных отсчетов на комплексные экспоненциальные колебания приводит к комплексному сигналу, как показано на рис. 5.10.

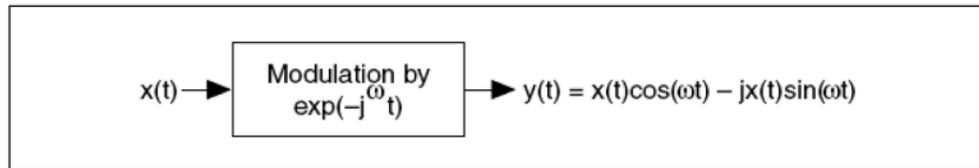


Рис. 5.10. Модуляция комплексными экспоненциальными колебаниями

### 5.5.5. Отображение частотной информации после преобразования

Дискретная интерпретация преобразования Фурье заключается в преобразовании цифрового сигнала в коэффициенты ряда Фурье или гармоники. К сожалению, ни время, ни шкала частот непосредственно не связаны с операцией БПФ. Поэтому необходимо определить интервал  $\Delta t$  получения отсчетов в выборке.

Поскольку полученный при дискретизации массив единичных измерений в выборке представляет эквидистантные во времени отсчеты выборки, можно определить соответствующую частоту в герцах. В следующем уравнении определяется частота  $f_s$  дискретизации исходного сигнала через интервал  $\Delta t$ :

$$f_s = \frac{1}{\Delta t}.$$

На рис. 5.11 показана блок-диаграмма ВП, который должным образом отображает информацию о частотах гармонических составляющих сигнала по заданному интервалу дискретизации  $1,000 \times 10^{-3}$  с и рассчитывает амплитуды гармонических составляющих через интервал  $\Delta f$  по частоте.

На рис. 5.12 показан дисплей и индикатор интервала  $\Delta f$ , которые определяются виртуальным прибором, представленным блок-диаграммой на рис. 5.11.

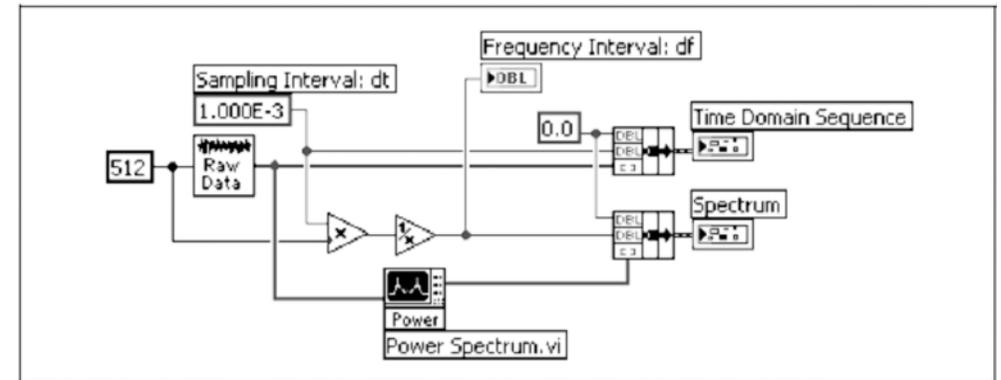


Рис. 5.11. Блок-диаграмма виртуального прибора для отображения информации о частоте

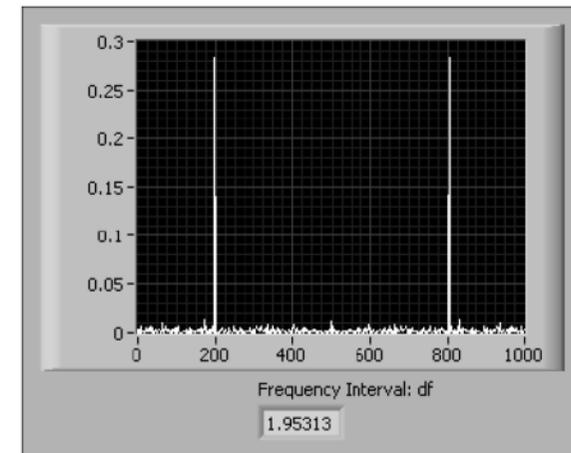


Рис. 5.12. Отображенная информация о частоте на лицевой панели виртуального прибора

Два других обычных способа представления информации о частоте обеспечивают отображение постоянной составляющей в центре или показывают односторонние спектры.

### 5.5.6. Двухстороннее БПФ, центрированное относительно постоянной составляющей

Двухсторонний спектр, центрированный относительно постоянной составляющей после применения БПФ, обеспечивает отображение спектра и с положительными, и с отрицательными частотами. В большинстве учебников, в которых рассматривается преобразование Фурье и его свойства, представляется таблица прямого и обратного преобразований Фурье. Можно использовать свойство преобразования Фурье сдвига спектра по частоте, чтобы получить двухстороннее, центрированное относительно постоянной составляющей представление. В этом случае компонента постоянной составляющей находится в середине лицевой панели.

### 5.5.7. Математическое представление двустороннего БПФ, центрированного относительно постоянной составляющей

Если есть пара преобразований Фурье  $x(t) \Leftrightarrow X(f)$ , тогда

$$x(t)e^{j2\pi f_0 t} \Leftrightarrow X(f - f_0).$$

Интервал дискретизации по времени определяется по формуле

$$\Delta t = \frac{1}{f_s},$$

где  $f_s$  – частота дискретизации исходного сигнала.

Расчет значения частоты  $f_0$ , равного частоте, соответствующей составляющей с частотой Найквиста  $f_N$ , показан в следующем уравнении:

$$f_0 = f_N = \frac{f_s}{2} = \frac{1}{2\Delta t}.$$

Частота  $f_0$  определена в точке, соответствующей  $f_N$ , для того чтобы постоянная составляющая появилась в месте расположения гармоники с частотой Найквиста. Для этого необходимо, чтобы сдвиг по частоте всех составляющих равнялся  $f_N$ .

Установка  $f_0$  по оси частот в точке, соответствующей  $f_N$ , задается парой дискретных преобразований Фурье:

$$x_i e^{j i \pi} \Leftrightarrow X_{k - \frac{n}{2}},$$

где  $n$  – число элементов в дискретной последовательности;  $x_i$  –  $i$ -й элемент последовательности в области времени;  $X_k$  – представление  $x_i$  в области частоты.

Распространение экспоненциального показателя во временной последовательности производится следующим выражением:

$$e^{j i \pi} = \cos(i\pi) + j \sin(i\pi) = \begin{cases} 1, & \text{если } i \text{ четное;} \\ -1, & \text{если } i \text{ нечетное.} \end{cases} \quad (5.6)$$

Уравнение (5.6) представляет собой последовательность чередований  $+1$  и  $-1$ . В то же время это уравнение обеспечивает преобразование в отрицательные значения нечетных элементов исходной последовательности в области времени, и затем выполняется БПФ новой последовательности и производится определение спектра, у которого компонента постоянной составляющей появляется в центре последовательности по частоте.

Поэтому, если исходная последовательность есть

$$X = \{x_0, x_1, x_2, x_3, \dots, x_{n-1}\},$$

тогда последовательность

$$Y = \{x_0, -x_1, x_2, -x_3, \dots, x_{n-1}\} \quad (5.7)$$

создает спектр с постоянной составляющей в центре.

### 5.5.8. Создание двустороннего БПФ с постоянной составляющей в центре

Можно модулировать сигнал частотой Найквиста непосредственно, без дополнительных преобразований. На рис. 5.13 показана блок-диаграмма виртуального прибора, обеспечивающего сдвиг спектра к частоте Найквиста (Nyquist Shift VI). Этот ВП расположен в библиотеке `labview\examples\analysis\dsp\mpl.llb` и создает последовательность, показанную в уравнении (5.7).

На рис. 5.13 в петле программы выполняются итерации отсчетов входной последовательности и поочередно умножаются элементы массива на  $1$  и  $-1$ , пока не будет обработан полностью входной массив.

На рис. 5.14 показана блок-диаграмма виртуального прибора, который создает последовательность в области времени и использует ВП Nyquist Shift и Power Spectrum, чтобы произвести спектр с постоянной составляющей в центре.

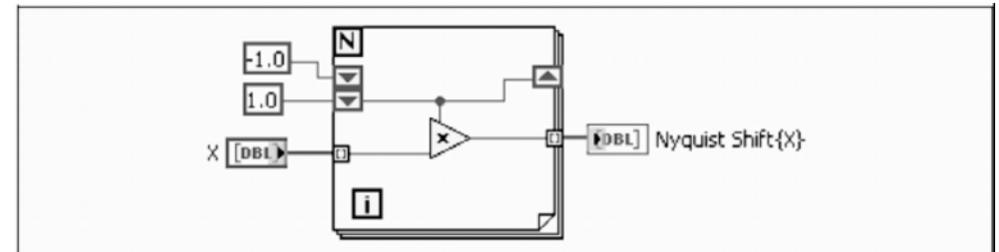


Рис. 5.13. Блок-диаграмма виртуального прибора для сдвига на частоту Найквиста (Nyquist Shift VI)

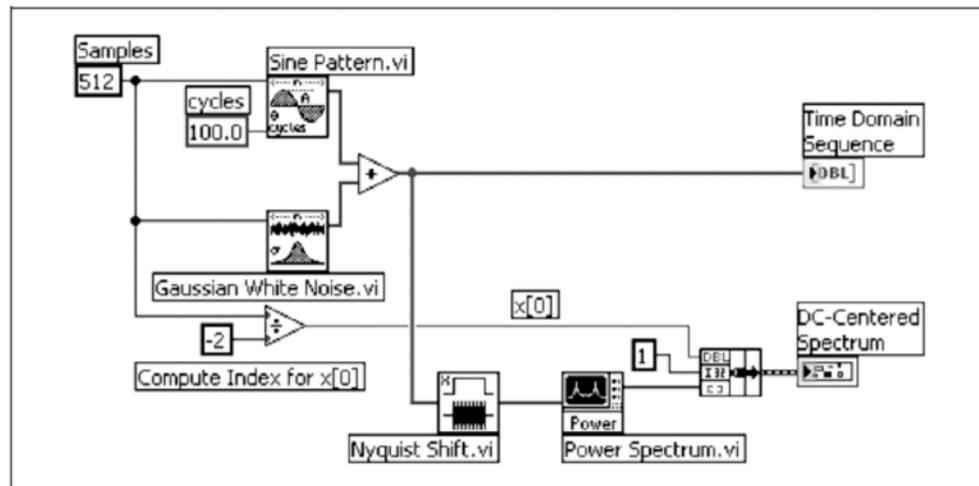


Рис. 5.14. Производство последовательности в области времени и спектра с постоянной составляющей в центре

В виртуальном приборе Nyquist Shift VI на рис. 5.14 предпроцессорная последовательность в области времени делает отрицательным каждый второй элемент в последовательности.

Виртуальный прибор Power Spectrum VI преобразует данные в области частоты. Чтобы отображать ось частот для преобразованных данных правильно, необходимо снабдить эти данные отсчетом  $x_0$ , который является значением начала оси  $X$  по частоте. Для получения спектра с постоянной составляющей в центре вычисляют значение  $x_0$  по следующему выражению:

$$x_0 = -\frac{n}{2}.$$

На рис. 5.15 показана последовательность в области времени и виртуальный прибор, представленный блок-диаграммой на рис. 5.14, для получения спектра с постоянной составляющей в центре на лицевой панели.

На рис. 5.15 представлен также дисплей со спектром, в котором постоянная составляющая сигнала расположена в центре (в точке  $f = 0$ ). Полный формат рисунка напоминает обычное представление спектра в таблицах пары преобразований Фурье.

Можно создать спектры с постоянной составляющей в центре для входной последовательности четного размера, делая отрицательными нечетные элементы входной последовательности.

В то же время невозможно создать спектры с постоянной составляющей в центре, непосредственно делая отрицательными нечетные элементы входной последовательности в области времени, содержащей нечетное число элементов, потому что составляющая с частотой Найквиста появляется между двумя отсчетами частоты. Чтобы создавать спектры с постоянной составляющей в центре для входных

Рис. 5.15. Необработанная последовательность в области времени и спектр с постоянной составляющей в центре

последовательностей нечетного размера, необходимо перемещать массив отсчетов БПФ на величину, определяемую следующим отношением:

$$\frac{n-1}{2}.$$

Для спектра с постоянной составляющей в центре, преобразованного из входной последовательности нечетного размера, следующее уравнение позволяет вычислить значение  $x_0$ :

$$x_0 = -\frac{n-1}{2}.$$

## 5.6. Спектр мощности

ДПФ или БПФ реального сигнала – комплексное число, имеющее реальную и мнимую части. Можно определить мощность (выделяемую на сопротивлении в 1 Ом) каждой спектральной составляющей, полученной с помощью ДПФ или БПФ, возводя в квадрат амплитуду этой спектральной составляющей. Таким образом, мощность  $k$ -й спектральной составляющей есть  $k$ -й элемент результата ДПФ или БПФ:

$$\text{мощность} = |X[k]|^2,$$

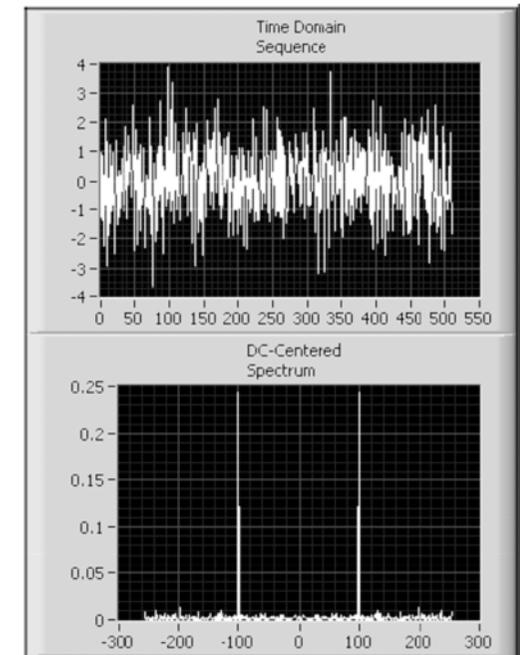
где  $|X[k]|$  – амплитуда частотной составляющей.

Спектр мощности определяет массив, который содержит двусторонний спектр мощности сигнала в области времени, и показывает мощность каждой из спектральных составляющих. Можно использовать следующее уравнение, чтобы вычислить двусторонний спектр мощности (power spectrum) на основе БПФ:

$$\text{Power Spectrum } S_{AA}(f) = \frac{\text{FFT}(A) \times \text{FFT}^*(A)}{N}, \quad (5.8)$$

где БПФ\*( $A$ ) ( $\text{FFT}^*(A)$ ) обозначает комплекс, сопряженный с БПФ( $A$ ). Комплекс, сопряженный с БПФ( $A$ ), есть следствие наличия отрицательной мнимой части БПФ( $A$ ).

Значения элементов в массиве спектра мощности пропорциональны квадрату амплитуды каждой спектральной составляющей сигнала в области времени. Поскольку ДПФ или БПФ реального сигнала симметричны, мощность составляющей при положительной частоте  $k\Delta f$  та же самая, что и мощность при соответ-



вующей отрицательной частоте  $-k\Delta f$ , исключая постоянную составляющую и составляющую с частотой Найквиста. Полная мощность постоянной составляющей есть  $|X[0]|^2$ . В то же время полная мощность составляющей с частотой Найквиста есть  $|X[N/2]|^2$ .

График двустороннего спектра мощности показывает амплитуды спектральных составляющих с отрицательными и положительными частотами по высоте, определяемой следующим отношением:

$$\frac{A_k^2}{4},$$

где  $A_k$  – амплитуда синусоидальной составляющей с частотой, имеющей номер  $k$ .

Постоянная составляющая в спектре мощности имеет высоту  $A_0^2$  ( $A_0$  – значение постоянной составляющей в исходном сигнале).

На рис. 5.16 показан спектр мощности, полученный из сигнала в области времени, который состоит из гармонического колебания с амплитудой  $3V_{\text{rms}}$  и частотой 128 Гц и гармонического колебания с амплитудой  $3V_{\text{rms}}$  и частотой 256 Гц, а также из постоянной составляющей 2В. Колебания с амплитудами  $3V_{\text{rms}}$  имеют пиковое значение напряжения  $3\sqrt{2}V \approx 4,2426V$  ( $\text{rms}$  – среднеквадратическое значение). Спектр мощности вычислен на основе базового БПФ, как показано в уравнении (5.8).

### 5.6.1. Преобразование двустороннего спектра мощности в односторонний спектр мощности

Большинство приборов для анализа спектра отображает только положительную половину спектра, потому что спектр реального сигнала является симметричным относительно постоянной составляющей. Таким образом, информация о составляющих с отрицательными частотами избыточна. Двусторонние результаты спектральных преобразований включают положительную и отрицательную половины спектра, как показано на рис. 5.16.

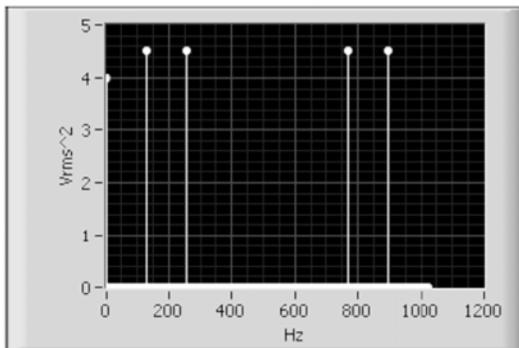


Рис. 5.16. Двусторонний спектр мощности сигнала

Двусторонний спектр мощности отображает половину энергии в положительной области частот и половину энергии в отрицательной области частот. Поэтому, чтобы преобразовать двусторонний спектр в односторонний спектр, необходимо отказаться от второй половины массива и умножить мощность каждой составляющей на два, кроме постоянной составляющей ( $DC$ ), как показано в следующих уравнениях:

$$G_{AA}(i) = S_{AA}(i), i = 0(DC);$$

$$G_{AA}(i) = (2S_{AA}(i)), i = 1, \dots, \frac{N}{2} - 1,$$

где  $S_{AA}(i)$  – двусторонний спектр мощности;  $G_{AA}(i)$  – односторонний спектр мощности;  $N$  – длина массива двустороннего спектра мощности. Можно отказаться от остатка двустороннего  $S_{AA}$  спектра мощности с номерами составляющих от  $N/2$  до  $N - 1$ .

Значения амплитуд гармоник без постоянной составляющей в одностороннем спектре дают высоту пиков в соответствии со следующим отношением:

$$\frac{A_k^2}{2}. \quad (5.9)$$

Уравнение (5.9) эквивалентно следующему выражению:

$$\left( \frac{A_k}{\sqrt{2}} \right)^2,$$

где  $\frac{A_k}{\sqrt{2}}$  – среднеквадратическое значение синусоидальной составляющей с частотой, имеющей индекс  $k$ .

Модули спектра мощности часто соответствуют величине квадрата среднеквадратического значения, которое является единицей измерения сигнала в области времени. Например, односторонний спектр мощности сигнала-напряжения измеряется в вольтах среднеквадратического значения, возведенного в квадрат:  $V_{\text{rms}}^2$ .

На рис. 5.17 показан односторонний спектр сигнала-напряжения, двусторонний спектр которого показан на рис. 5.16.

На рис. 5.17 высота спектральных составляющих, начиная с первой гармоники, вдвое больше высоты спектральных составляющих, показанных на рис. 5.16, кроме постоянной составляющей. Кроме того, спектр на рис. 5.17 размещается на частотном интервале, который вдвое меньше, чем интервал на рис. 5.16.

### 5.6.2. Потеря фазовой информации

Поскольку мощности составляющих получены возведением в квадрат их амплитуд после ДПФ или БПФ, спектр мощности всегда реален. Недостаток получения мощности таким путем состоит в том, что информация о фазе теряется. Если необходима информация о фазе, надо использовать ДПФ или БПФ, которые дают комплексный результат.

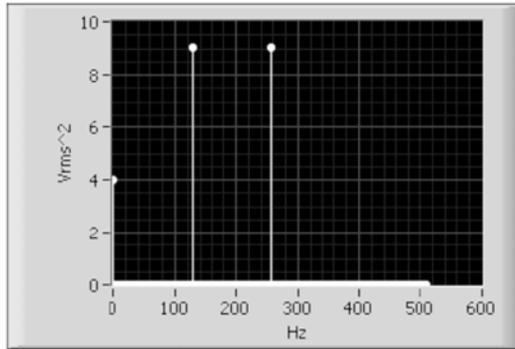


Рис. 5.17. Односторонний спектр мощности

Можно использовать спектр мощности там, где информация о фазе не нужна, например при вычислении мощности гармоник в сигнале. Можно подать синусоидальный сигнал на вход нелинейной системы и определить его мощность на выходе системы.

### 5.6.3. Вычисления на основе спектра

Когда имеется амплитудный спектр или спектр мощности, можно вычислить несколько полезных параметров входного сигнала: мощность и частоту, шумовой уровень и мощность шума, спектральную плотность мощности.

### 5.6.4. Оценка мощности и частоты

Если спектральная составляющая сигнала находится на оси частот между двумя частотными линиями, то после ДПФ ее мощность распределяется среди смежных частотных линий. Фактически максимальное значение мощности спектральной составляющей находится между двумя частотными линиями.

Можно оценить фактическую частоту (estimate frequency) по дискретной спектральной составляющей с большей разрешающей способностью, чем  $\Delta f$ , определяя среднее взвешенное число частот вокруг обнаруженного пика в спектре мощности, как показано в следующем уравнении:

$$\text{Estimate Frequency} = \frac{\sum_{i=j-3}^{j+3} (\text{Power}(i)(i\Delta f))}{\sum_{i=j-3}^{j+3} \text{Power}(i)},$$

где  $j$  – индекс массива очевидного пика на частоте, представляющей интерес.

Промежуток  $j \pm 3$  разумен, потому что такой выбор индексов обеспечивает учет составляющих на интервале частот, превышающем ширину основных лепестков окон сглаживания, приведенных в табл. 7.3.

Можно оценить мощность в единицах измерения  $V_{\text{rms}}^2$  дискретного пикового компонента частоты, определяя сумму мощностей ближайших по частоте составляющих относительно максимальной. Другими словами, вычисляется область под пиком. Можно использовать следующее уравнение, чтобы оценить мощность (Estimate Power) дискретного пикового компонента частоты:

$$\text{Estimate Power} = \frac{\sum_{i=j-3}^{j+3} \text{Power}(i)}{\sum_{i=j-3}^{j+3} \text{Power}(i)}. \quad (5.10)$$

Использование уравнения (5.10) допустимо только для спектра, составленного из дискретных частотных составляющих. Оно не имеет смысла для непрерывного спектра. Точно так же, если два или больше пика по частоте находятся в пределах шести линий друг около друга, то они способствуют смещению в большую сторону оцененных мощностей и искажениям фактических частот. Можно снизить влияние этого эффекта, уменьшая номера отсчетов, входящих в уравнение (5.10). Если на спектральной диаграмме есть два пика, сгруппированных в пределах шести отсчетов, то, вероятно, они уже смешиваются друг с другом из-за спектральной утечки.

Если необходимо определить полную мощность в данном диапазоне частоты, суммируются мощности каждого отсчета, включенного в частотный диапазон, и результат сложения делится на шумовую полосу пропускания окна сглаживания. Можно обратиться к разд. 5 для получения информации о шумовой полосе пропускания мощности окон сглаживания.

### 5.6.5. Вычисление шумового уровня и спектральная плотность мощности

Измерение шумовых уровней зависит от полосы пропускания измерителя. При анализе шумового порога спектра мощности измеряется узкополосный шумовой уровень в каждом дискрете БПФ. Поэтому шумовой порог данного спектра мощности зависит от  $\Delta f$  спектра, который, в свою очередь, зависит от частоты дискретизации и числа единичных отсчетов в выборке. Другими словами, шумовой уровень в каждом дискрете по частоте эквивалентен шумовому уровню, полученному с помощью фильтра с полосой  $\Delta f$  (Гц), центрированной относительно этого дискретного значения частоты. Поэтому для данной частоты дискретизации удвоение числа полученных отсчетов выборки уменьшает шумовую мощность, которая имеется в каждом частотном дискрете, на 3 дБ. Теоретически дискретные спек-

ральные составляющие имеют нулевую полосу пропускания и поэтому не масштабируются числом точек или диапазоном частот БПФ.

Чтобы вычислить отношение сигнал/шум (SNR), сравнивают пиковую мощность на частотах, представляющих интерес, с шумовым уровнем для широкой полосы. Вычисляют широкополосный шумовой уровень, производя суммирование всех отсчетов спектра мощности, исключая любые пики и постоянную составляющую, и делят сумму на эквивалентную шумовую полосу пропускания всего окна.

При делении шумового уровня на полосу  $\Delta f$  и стремлении  $\Delta f$  к нулю спектры для шумовых измерений часто отображаются в нормализованном формате, названном значением спектральной плотности мощности. Мощность для получения значения спектральной плотности нормализуют относительно ширины спектра, взвешенного прямоугольным по АЧХ фильтром с полосой в 1 Гц, используемым для измерений шумового уровня. Уровень в каждом дискретном значении по частоте эквивалентен уровню, рассчитанному на основе использования фильтра с полосой пропускания в 1 Гц и с АЧХ, центрированной относительно этого же значения по частоте.

Можно использовать следующую формулу, чтобы вычислить спектральную плотность мощности (Power Spectral Density):

$$\text{Power Spectral Density} = \frac{\text{Power Spectrum in } B_{\text{rms}}^2}{\Delta f \times \text{Noise Power Bandwidth of Window}}$$

Можно также использовать следующее выражение, чтобы вычислить значение спектральной плотности:

$$\text{Amplitude Spectral Density} = \frac{\text{Amplitude Spectrum in } B_{\text{rms}}}{\sqrt{\Delta f \times \text{Noise Power Bandwidth of Window}}}$$

Спектральный формат плотности мощности является приближением для случайных, или шумовых, сигналов.

В то же время для дискретных спектральных составляющих спектральный формат плотности мощности не является приближением, потому что они теоретически занимают нулевую полосу на оси частот.

### 5.6.6. Вычисление амплитудных и фазовых спектров

Спектр мощности соответствует мощности, рассчитанной как квадрат амплитуды для каждой частоты, но не имеет никакой информации о начальной фазе. Поскольку спектр мощности теряет информацию о начальной фазе, можно попытаться использовать БПФ, чтобы определить и частоту, и информацию о фазе сигнала.

Информация о начальной фазе, которую БПФ обеспечивает, есть фаза относительно начала отсчета сигнала в области времени. Поэтому необходимо начинать выборку от некоторого момента в сигнале, чтобы получить непротиворечивые

сведения о начальной фазе. Колебание по закону синуса имеет начальную фазу, равную  $-90^\circ$ . Колебание по закону косинуса имеет начальную фазу, равную  $0^\circ$ . Обычно основным интересом для анализа спектра сигнала представляет измерение разности фаз между составляющими спектра или разности фаз между двумя гармоническими колебаниями, полученными одновременно. Можно рассмотреть разность фаз между двумя сигналами, используя некоторые из расширенных функций БПФ.

В результате БПФ получают двусторонний спектр в комплексной форме с реальными и мнимыми частями. Необходимо масштабировать и преобразовать двусторонний спектр в полярную форму, чтобы получить амплитуду и фазу каждой гармонической составляющей сигнала. Ось частоты полярной формы идентична оси частоты двустороннего спектра мощности.

Амплитуда спектральных составляющих БПФ связана с номером единичных отсчетов сигнала в области времени. Можно использовать следующие уравнения, чтобы вычислить амплитуду и фазу каждой составляющей после применения БПФ:

$$\begin{aligned} \text{Amplitude spectrum in quantity peak} &= \frac{\text{Magnitude}[FFT(A)]}{N} = \\ &= \frac{\sqrt{[\text{real}[FFT(A)]]^2 + [\text{imag}[FFT(A)]]^2}}{N}; \end{aligned} \quad (5.11)$$

$$\text{Phase spectrum in radians} = \text{Phase}[FFT(A)] = \arctg\left(\frac{\text{imag}[FFT(A)]}{\text{real}[FFT(A)]}\right), \quad (5.12)$$

где функция арктангенса определяет значения фазы в интервале между  $-\pi$  и  $+\pi$  в полном диапазоне, составляющем  $2\pi$  радиан.

Следующее отношение определяют функцию преобразования прямоугольных координат в полярные:

$$\frac{FFT(A)}{N}. \quad (5.13)$$

Использование функции (5.13) позволяет преобразовывать комплексный спектр в амплитудный ( $r$ ) и фазовый ( $\phi$ ) спектры, что эквивалентно использованию выражений (5.11) и (5.12).

Двусторонний амплитудный спектр фактически содержит половину амплитудных значений при положительных и отрицательных частотах. Чтобы преобразовать спектр в одностороннюю форму, необходимо умножить каждую амплитуду, кроме постоянной составляющей, на два и отказаться от второй половины массива. Амплитуды составляющих в одностороннем спектре имеют максимальные значения для каждой гармонической составляющей спектра, которые описывают сигнал в области времени.

Чтобы получить односторонний спектр начальных фаз, необходимо также отказаться от второй половины массива.

### 5.6.7. Вычисление амплитуды в вольтах среднеквадратических значений ( $B_{rms}$ ) и начальной фазы в градусах

Чтобы рассматривать спектр амплитуд в вольтах среднеквадратических значений ( $B_{rms}$ ), необходимо разделить все амплитуды гармоник спектра, кроме постоянной составляющей, на  $\sqrt{2}$  после преобразования полученного спектра в одностороннюю форму. Поскольку выполняется умножение всех амплитуд, кроме постоянной составляющей, на два, чтобы перейти от двустороннего спектра к одностороннему амплитудному спектру, можно вычислить среднеквадратический амплитудный спектр непосредственно на основе двустороннего амплитудного спектра, умножая все его компоненты, кроме постоянной составляющей, на  $\sqrt{2}$ , а затем отказаться от второй половины массива. Следующие уравнения показывают в целом вычисления при переходе от двустороннего БПФ к одностороннему амплитудному спектру:

$$\text{Amplitude Spectrum } B_{rms} = \sqrt{2} \frac{\text{Magnitude}[FFT(A)]}{N} \text{ для } i = 1 \text{ до } \frac{N}{2} - 1;$$

$$\text{Amplitude Spectrum } B_{rms} = \frac{\text{Magnitude}[FFT(A)]}{N} \text{ для } i = 0 \text{ (DC),}$$

где  $i$  – номер дискретного значения по частоте, или индекс массива  $A$  после применения БПФ.

Величина в  $B_{rms}$  дает среднеквадратическое значение напряжения каждой гармонической составляющей как функции в области времени.

Амплитудный спектр тесно связан со спектром мощности. Можно вычислить односторонний спектр мощности возведением в квадрат одностороннего среднеквадратического амплитудного спектра. Наоборот, можно определить амплитудный спектр, вычисляя квадратный корень из мощности составляющих спектра.

Можно использовать следующее уравнение, чтобы рассчитать спектр фазы (*Phase Spectrum*) в градусах (*degree*):

$$\text{Phase Spectrum in Degrees} = \frac{180}{\pi} \text{Phase FFT}(A).$$

### 5.6.8. Частотная характеристика

При анализе двух одновременно дискретизированных выборок на выходах двух каналов часто необходимо определить различия между этими двумя каналами, а не свойства каждого.

В типичном двухканальном анализаторе, как показано на рис. 5.18, мгновенный спектр вычисляется на основе использования оконной функции и БПФ для каждого канала. Усредненный оконной функцией БПФ-спектр (Average FFT), автоспектр мощности (Auto Spectrum) и перекрестный спектр (Cross Spectrum)

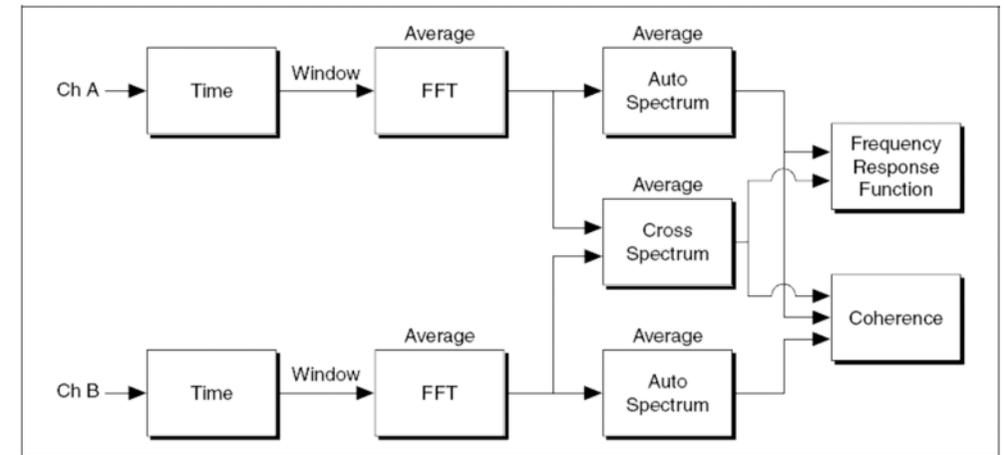


Рис. 5.18. Двухканальный спектральный анализатор

мощности вычисляются и используются в оценке частотной характеристики (Frequency Response Function). Также можно использовать функцию когерентности (Coherence), чтобы проверить законность вычислений частотной характеристики.

Частотная характеристика описывается модулем  $|H|$  и фазой  $\angle H$  на каждой частоте. Комплексная передаточная функция системы определяется отношением комплексной амплитуды гармонического сигнала на выходе к входной комплексной амплитуде на каждой частоте. Фаза системы – разность фаз выходного и входного гармонических колебаний на каждой частоте.

## 5.7. Перекрестный (взаимный) спектр мощности (Cross Power Spectrum)

Перекрестный спектр мощности обычно не используется как прямое измерение, но является важным стандартным блоком для других измерений.

Можно использовать следующее уравнение, чтобы вычислить двусторонний перекрестный спектр мощности двух сигналов  $A$  и  $B$ , заданных в области времени:

$$\text{Cross Power Spectrum } S_{AB}(f) = \frac{FFT(B) \times FFT^*(A)}{N^2}.$$

Перекрестный спектр мощности имеет двустороннюю комплексную форму и содержит реальную и мнимую части. Чтобы преобразовать составляющие перекрестного спектра мощности в амплитудный и фазовый спектры, необходимо использовать функцию преобразования прямоугольных координат в полярные в соответствии с формулой (5.13).

Односторонний перекрестный спектр мощности приводится к совокупности среднеквадратических значений амплитуд и разностей фаз между двумя гармо-

ническими сигналами  $A$  и  $B$ . Модулем одностороннего перекрестного спектра мощности является среднеквадратическое значение составляющих (rms) в квадрате с единицей измерения, например,  $V_{\text{rms}}^2$ .

Спектр мощности эквивалентен перекрестному спектру мощности, когда сигналы  $A$  и  $B$  одинаковы. Поэтому он называется часто автоспектром мощности или автоспектром.

### 5.7.1. Частотная характеристика и анализ четырехполюсников

Можно использовать следующие характеристики, чтобы определить частотные свойства четырехполюсника:

- частотную характеристику;
- импульсную характеристику;
- функцию когерентности.

### 5.7.2. Частотная характеристика

На рис. 5.19 показана структура метода измерения частотной характеристики четырехполюсника.

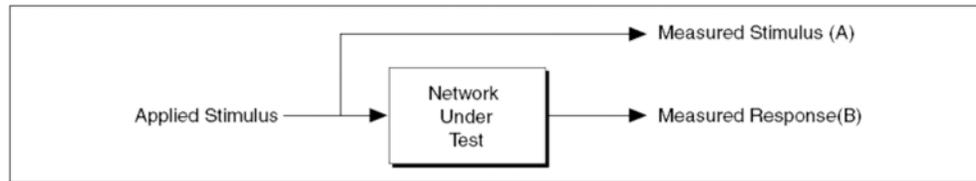


Рис. 5.19. Структурная схема метода анализа четырехполюсника: *Applied Stimulus* – приложенное воздействие; *Measured stimulus* – измеренное воздействие; *Network Under Test* – тестируемый четырехполюсник; *Measured Response* – измеренный отклик

Согласно рис. 5.19, при испытании на вход четырехполюсника подается сигнал и измеряется сигнал отклика. По измеренным параметрам входного сигнала и сигнала отклика вычисляется частотная характеристика. Частотная характеристика показывает зависимость модуля (амплитудно-частотная характеристика) и фазы (фазочастотная характеристика) четырехполюсника от частоты. Используется следующее выражение для вычислений частотной характеристики:

$$H(f) = \frac{S_{AB}(f)}{S_{AA}(f)}, \quad (5.14)$$

где  $H(f)$  – частотная характеристика;  $A$  – гармонический сигнал на входе;  $B$  – гармонический сигнал на выходе;  $S_{AB}(f)$  – перекрестный спектр мощности сигналов  $A$  и  $B$ ;  $S_{AA}(f)$  – спектр мощности сигнала  $A$ .

Частотная характеристика – это двусторонняя комплексная форма, имеющая реальную и мнимую части. Чтобы преобразовать комплексную передаточную функцию в АЧХ и ФЧХ, можно использовать функцию преобразования прямоугольных координат в полярные координаты по уравнению (5.13). Чтобы представить эти характеристики в односторонней форме, необходимо отказаться от второй половины массива частотной характеристики.

Для уменьшения погрешности измерения можно взять несколько вычисленных частотных характеристик и определить среднюю характеристику. Для нахождения средней частотной характеристики необходимо выполнить следующие шаги:

- 1) вычислить среднее число  $S_{AB}(f)$ , находя сумму отсчетов характеристики в комплексной форме и деля эту сумму на число измерений;
- 2) вычислить средний спектр  $S_{AA}(f)$ , находя сумму амплитуд составляющих и деля ее на число измерений;
- 3) заменить в уравнении (5.14) характеристики  $S_{AB}(f)$  и  $S_{AA}(f)$  на средние характеристики  $S_{AB}(f)$  и  $S_{AA}(f)$ .

### 5.7.3. Импульсная характеристика

Импульсная характеристика четырехполюсника – представление в области времени его комплексной передаточной функции.

Импульсная характеристика четырехполюсника – сигнал в области времени на выходе, полученный при подаче на вход четырехполюсника  $\delta$ -импульса в момент времени  $t = 0$ .

Чтобы вычислить импульсную характеристику четырехполюсника, необходимо применить обратное БПФ к двусторонней комплексной передаточной функции по уравнению (5.14). Чтобы вычислять среднюю импульсную характеристику четырехполюсника, необходимо применить обратное БПФ к средней комплексной передаточной функции.

## 5.8. Функция когерентности

Функция когерентности обеспечивает определение относительного качества измерения комплексной передаточной функции: какая часть энергии отклика коррелирована с энергией входного сигнала. Если присутствует дополнительный сигнал в отклике (или от чрезмерного шума, или от другого сигнала), качество измерения отклика четырехполюсника невысокое. Можно использовать функцию когерентности, чтобы идентифицировать и чрезмерный шум, и множественные источники сигнала, влияющие на выходной сигнал. Следующее уравнение используется для вычисления функции когерентности:

$$\gamma^2(f) = \frac{(\text{Magnitude of Average } S_{AB}(f))^2}{(\text{Average } S_{AA}(f))(\text{Average } S_{BB}(f))}, \quad (5.15)$$

где  $S_{AB}$  – перекрестный спектр мощности;  $S_{AA}$  – спектр мощности  $A$ ;  $S_{BB}$  – спектр мощности  $B$ .

Выражение (5.15) позволяет определить коэффициент когерентности, имеющий величину между нулем и единицей для разных частот. Нулевое его значение для данной линии по частоте показывает отсутствие корреляции между откликом и входным сигналом. Единичное значение для данной линии по частоте указывает, что 100 % энергии отклика получается из энергии входного сигнала и что никакой другой сигнал не присутствует на этой частоте.

Для получения правильного результата функция когерентности требует определения среднего значения для двух или более измерений входного и выходного сигналов. Если используется только одно измерение, функция когерентности регистрирует единичное значение на всех частотах.

## 5.9. Работа с окнами

В практических приложениях спектрального анализа получается только конечное число выборок сигнала, в то же время БПФ предполагает, что временные отсчеты по времени носят периодический характер. Если имеется целое число периодов в исходной выборке по времени, то повторение этой выборки по оси времени имеет граничные значения без разрывов непрерывности. Однако на практике в выборке обычно не укладывается целое число периодов. В случае нецелого числа периодов такое повторение приводит к разрывам непрерывности на границах. Эти искусственные разрывы не присутствуют первоначально в исходном сигнале и приводят к «размазыванию», или «утечке», энергии спектральной составляющей с конкретной частотой на все другие частоты. Это явление называется спектральной утечкой. Величина утечки зависит от значения разрыва однородности на границах выборки: больший разрыв вызывает большие утечки.

Сигнал, который является строго периодическим в выборках по времени, составлен из гармонических составляющих с точными кратными периодами в пределах выборки по времени. Такой строго периодический сигнал имеет спектр с энергией, сосредоточенной только в точных дискретных отсчетах частоты.

Сигнал, который не является периодическим в выборке по времени, имеет спектр с разбиением энергии или распространенный по многим линиям частоты. Спектр БПФ моделирует область времени так, как будто выборка по времени повторяется всегда во времени. Это предполагает, что проанализированная выборка – это только один период бесконечно повторяющегося периодического сигнала.

Поскольку величина утечки зависит от величины разрыва однородности на границах выборки, можно использовать дополнительную обработку сигнала окнами на временной оси, чтобы уменьшить размер неоднородности и спектральную утечку. Обработка окнами заключается в умножения сигнала в области времени на колебание другой формы, известное как окно, амплитуда которого снижается постепенно и гладко к нулю на границах выборки. Результат – обработанный окном сигнал с очень малыми или с полным отсутствием непоследовательностей на границах выборки, за счет чего уменьшается спектральная утечка. Можно выбрать окно из числа многих известных типов окон. То окно, которое

необходимо выбрать, зависит от вида приложения спектрального анализа и небольшого количества предшествующей информации о сигнале, который анализируется.

## 5.10. Усреднение для улучшения измерений

Выполнение усреднений последовательных измерений часто повышает их точность. Усреднение обычно выполняется по результатам измерений индивидуальных спектров, но не непосредственно по отсчетам сигнала во времени.

Можно выбрать режим усреднения из числа следующих видов:

- среднеквадратическое (rms) усреднение;
- векторное усреднение;
- пиковое усреднение.

### 5.10.1. Среднеквадратическое усреднение (RMS Averaging)

Среднеквадратичное (RMS) усреднение уменьшает колебания сигнала, но не шумовой порог. Шумовой порог не уменьшается, потому что при среднеквадратическом усреднении усредняется значение энергии или мощности сигнала. Среднеквадратичное усреднение приводит также к тому, что усредненные среднеквадратические значения измерений для единственного канала имеют нулевую фазу. Среднеквадратическое усреднение для двухканальных измерений должно сохранять важную информацию о разности фаз.

Усредненные среднеквадратические значения измерений вычисляются согласно следующим уравнениям:

- спектр БПФ  $\sqrt{\langle X^* X \rangle}$ ;
  - спектр мощности  $\langle X^* X \rangle$ ;
  - взаимный спектр  $\langle X^* Y \rangle$ ;
  - частотная характеристика  $H1 = \frac{\langle X^* Y \rangle}{\langle X^* X \rangle}$ ;
- $$H2 = \left\langle \frac{Y^* Y}{Y^* X} \right\rangle;$$
- $$H3 = \frac{(H1 + H2)}{2},$$

где  $X$  – комплексное БПФ сигнала  $x$  (входного);  $Y$  – комплексное БПФ сигнала  $y$  (выходного);  $X^*$  – комплекс, сопряженный с  $X$ ;  $Y^*$  – комплекс, сопряженный с  $Y$ ;  $\langle X \rangle$  – усредненное  $X$  (реальные и мнимые части усредняются отдельно).

### 5.10.2. Векторное усреднение (Vector Averaging)

Векторное усреднение устраняет помеху от синхронных сигналов. При выполнении векторного усреднения вычисляется среднее значение комплексных величин непосредственно. Вещественная часть усредняется отдельно от мнимой части. При этом можно уменьшить шумовой порог для случайных сигналов, потому что случайные сигналы не имеют фазовой когерентности от одного отсчета времени к следующему. Реальные и мнимые части усредняются отдельно, что уменьшает уровень шума, но обычно требуют синхронизации (trigger):

- спектр БПФ  $\langle X \rangle$ ;
- спектр мощности  $\langle X^* \rangle \langle X \rangle$ ;
- взаимный спектр  $\langle X^* \rangle \langle Y \rangle$ ;
- частотная характеристика  $\frac{\langle Y \rangle}{\langle X \rangle}$  ( $H1 = H2 = H3$ ),

где  $X$  – комплексное БПФ сигнала  $x$  (входного);  $Y$  – комплексное БПФ сигнала  $y$  (отклик);  $X^*$  – комплекс, сопряженный с  $X$ ;  $\langle X \rangle$  – среднее число  $X$  (реальные и мнимые части усредняются отдельно).

### 5.10.3. Пиковое усреднение (Peak Hold)

Пиковое усреднение сохраняет пиковые уровни усредненных величин. Пиковое усреднение по частоте выполняется в каждом дискрете отдельно с сохранением пиковых уровней от одного отсчета БПФ до следующего:

- спектр БПФ  $MAX \sqrt{\langle X^* X \rangle}$ ;
- спектр мощности  $MAX \langle X^* X \rangle$ ,

где  $X$  – комплекс БПФ сигнала  $x$  (входного);  $X^*$  – комплекс, сопряженный с  $X$ .

## 5.11. Взвешивание (Weighting)

При выполнении среднеквадратичного или векторного усреднения можно взвешивать каждый новый спектральный отсчет, используя линейное или экспоненциальное взвешивание.

Линейное взвешивание объединяет  $N$  спектральных отсчетов с равными весовыми коэффициентами. Когда усреднение чисел завершено, анализатор прекращает усреднять и представляет усредненные результаты.

Экспоненциальное взвешивание подчеркивает новые спектральные данные больше, чем старые, и в результате образуется непрерывный процесс.

При усреднении весовые коэффициенты применяются согласно следующему выражению:

$$Y_i = \frac{N-1}{N} Y_{i-1} + \frac{1}{N} X_i,$$

где  $X_i$  – результат анализа, выполненного на  $i$ -м блоке;  $Y_i$  – результат процесса взвешивания от  $X_1$  до  $X_i$ ;  $N = i$  для линейного взвешивания;  $N$  – константа для экспоненциального взвешивания ( $N = 1$  для  $i = 1$ ).

## 5.12. Обнаружение эхосигнала (Echo Detection)

Обнаружение эхосигнала обеспечивается на основе использования преобразования Гильберта (Hilbert) – обычное измерение для анализа систем модуляции. Преобразование Гильберта позволяет определять аналитический (комплексный) сигнал, найти комплексную огибающую и выполнить обработку по комплексной огибающей. Такие операции имеют смысл для узкополосных сигналов, когда их полоса много меньше центральной частоты спектра.

Следующее уравнение описывает сигнал в области времени:

$$x(t) = Ae^{-t/\tau} \cos(2\pi f_0 t); \quad (5.16)$$

следующее уравнение показывает преобразование Гильберта сигнала в области времени:

$$x_H(t) = -Ae^{-t/\tau} \sin(2\pi f_0 t), \quad (5.17)$$

где  $A$  – максимальное значение сигнала при  $t = 0$ ;  $f_0$  – центральная частота спектра сигнала (обычно резонансная частота узкополосного фильтра);  $\tau$  – постоянная времени затухания амплитуд отклика.

Следующее уравнение представляет натуральный логарифм амплитуды аналитического сигнала  $x_A(t)$ :

$$\ln|x_A(t)| = \ln|x(y) + jx_H(t)| = -\frac{1}{\tau} \ln A. \quad (5.18)$$

При использовании уравнения (5.18) получается линия с наклоном  $m = -\frac{1}{\tau}$ .

Поэтому можно определить постоянную времени системы из графика  $\ln|x_A(t)|$ .

На рис. 5.20 показаны колебания в области времени, содержащие эхосигнал.

Следующие условия делают трудной локацию эхосигнала, что следует из анализа рис. 5.20:

- запаздывание между излучаемым колебанием и эхосигналом имеет малое значение относительно постоянной времени затухания системы;
- амплитуда эхосигнала мала по сравнению с амплитудой сигнала источника.

Можно сделать эхосигнал видимым, изображая график величины  $x_A(t)$  в логарифмическом масштабе, как показано на рис. 5.21.

На рис. 5.21 неоднородность явно видна и указывает местоположение и запаздывание эхосигнала.

На рис. 5.22 показана секция блок-диаграммы виртуального прибора, который использовался для выполнения рис. 5.20 и 5.21.

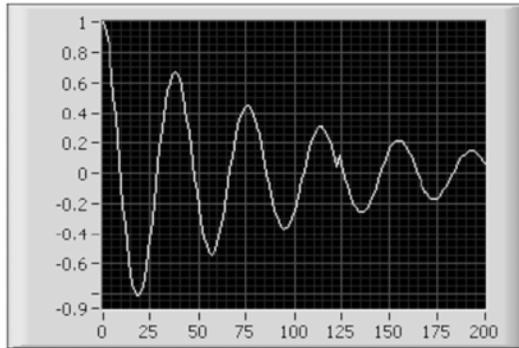


Рис. 5.20. Эхосигнал

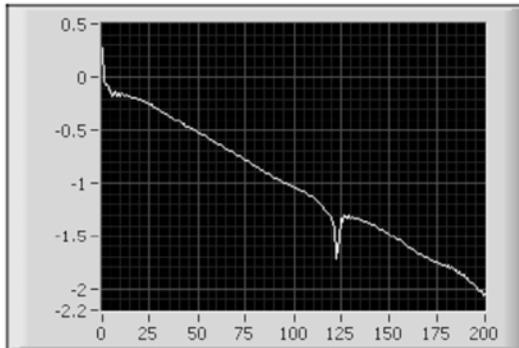
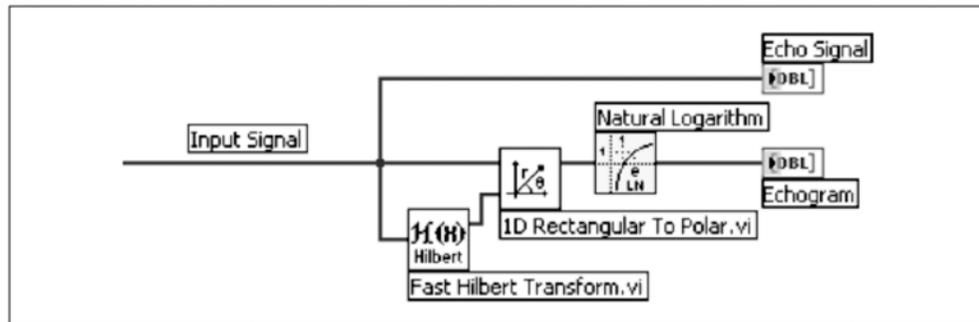
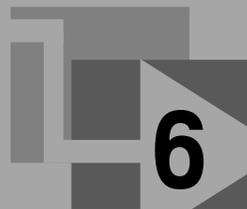
Рис. 5.21. Эхограмма амплитуды сигнала  $x_A(t)$ 

Рис. 5.22. Блок-диаграмма детектора эхосигнала

В виртуальном приборе, блок-диаграмма которого представлена на рис. 5.22, обеспечивается выполнение следующих шагов для обнаружения эхосигнала:

- 1) обрабатывается входной сигнал в виртуальном приборе с быстрым преобразованием Гильберта (Fast Hilbert Transform VI), чтобы сформировать аналитический сигнал  $x_A(t)$ ;
- 2) вычисляется величина  $x_A(t)$  в виртуальном преобразователе (1D Rectangular To Polar.vi) прямоугольных координат в полярные;
- 3) вычисляется натуральный логарифм от  $x_A(t)$ , для того чтобы обнаружить присутствие эхосигнала;
- 4) индицируются исходный сигнал (Echo signal) и эхограмма (Echoграм).

# Примеры спектрального анализа в LabVIEW



Перед дальнейшим изучением цифровой обработки сигналов связи необходимо исследовать некоторые из особенностей спектрального анализа в LabVIEW.

Инструментарий спектрального анализа будет нужен позже, когда будут исследоваться спектры входных сигналов в различных узлах системы связи. И, конечно, этот инструментарий также будет полезен при анализе частотных характеристик фильтров. Большинство версий LabVIEW включает все спектральные методы и алгоритмы обработки сигналов, в которых когда-либо возникнет необходимость. Среди стандартных функций – реальное БПФ, комплексное БПФ и преобразования Гильберта (Hilbert) и Хартли (Hartley), спектр мощности а также инверсии этих функций.

Есть также много расширенных спектральных функций, которые основываются на упомянутых выше стандартных функциях. В последующих подразделах некоторые из обычных стандартных спектральных функций будут преобразованы в полезные виртуальные приборы (ВП) (VI). В случаях, когда необходимо использовать конкретный алгоритм, основанный на БПФ, или когда представленный пакет LabVIEW не включает спектральные преобразования, которые требуются (типа дискретного косинусного преобразования или вейвлет-анализа), пакет LabVIEW обеспечивает возможность преобразования программы на основе собственного компилятора в код C через кодовый узел интерфейса.

Это стоит упомянуть по причине наличия обширных библиотек в кодах C, доступных в Интернете, чтобы выполнить некоторые из сложных спектральных преобразований.

## 6.1. Преобразования низкого уровня в области частоты

Для начала необходимо ознакомиться с основными преобразованиями в области частоты, показанными на рис. 6.1.

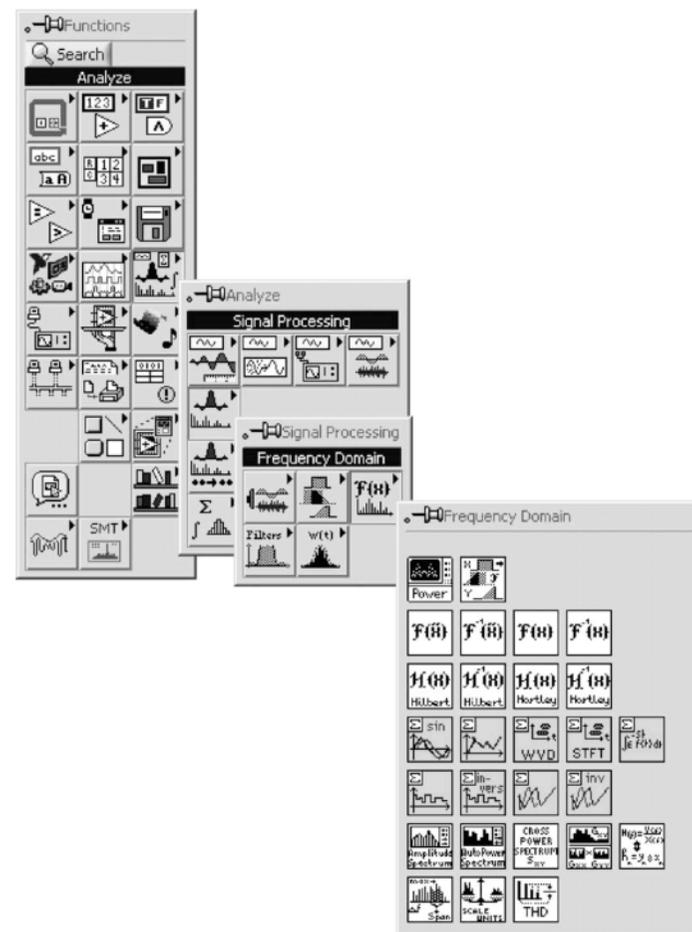


Рис. 6.1. Преобразования низкого уровня в области частоты

Из этих преобразований, вероятно, наиболее используемое для цифровых систем связи – комплексное преобразование Фурье  $F(x)$  и реальное преобразование Фурье  $F(x)$ . В цифровых системах связи приходится иметь дело и с реальными, и с комплексными типами сигналов, и поэтому необходимо использовать обе функции преобразования Фурье в LabVIEW. Пока размер входного сигнала есть число по основанию 2, эти функции производят распределенное на две половины БПФ, т. е. выполняют дискретное преобразование Фурье. Так как оба эти преобразования приводят к идентичным результатам, мы будем пробовать использовать более эффективное выполнение БПФ, приспособливаясь к длине входной выборки.

Кроме оперирования с комплексными отсчетами входного сигнала, в чем различия между реальной и комплексной версиями БПФ? С учетом свойства преобразования Фурье преобразование реального сигнала приведет к получению спектральной плотности, симметричной относительно 0 Гц в диапазоне от  $-f_s / 2$  до

$f_s / 2$ . Это означает, что две половины спектра являются зеркальными. С другой стороны, преобразование Фурье комплексного сигнала приведет к полностью независимым спектральным составляющим в каждой половине диапазона. Это говорит о том, что результат реального преобразования Фурье в LabVIEW имеет только положительную половину спектра сигнала, а результат комплексного преобразования Фурье - обе половины. Две половины спектра в комплексном случае представляют результат в диапазоне частот от 0 до  $2\pi$  и поэтому требуют некоторого небольшого перемещения, чтобы анализировать сигнал в диапазоне от  $-\pi$  до  $\pi$ . Эта операция будет показана более подробно в следующем пункте.

### 6.1.1. Простое БПФ

Простое стандартное БПФ, представленное на рис. 6.2, говорит нам о нескольких интересных особенностях использования в LabVIEW комплексного БПФ. Спектр сигнала после комплексного БПФ начинается с компоненты постоянного тока, занимает половину выборки, а затем отображается зеркально на второй половине выборки.

Такие результаты типичны для БПФ и легко могут быть изменены при разделении множества отсчетов спектра на две половины, которые располагаются зеркально относительно друг друга (рис. 6.3).

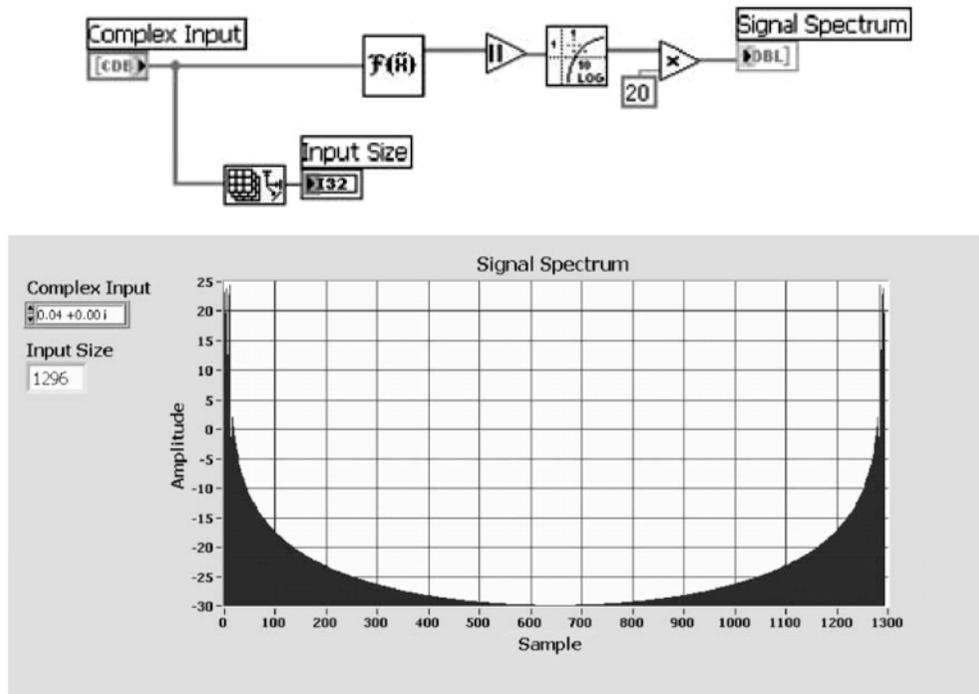


Рис. 6.2. ВП для вычисления простого БПФ

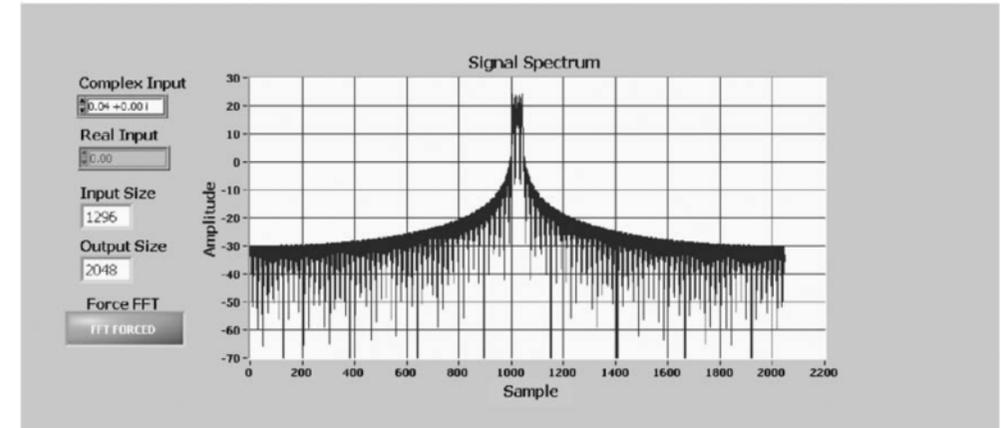
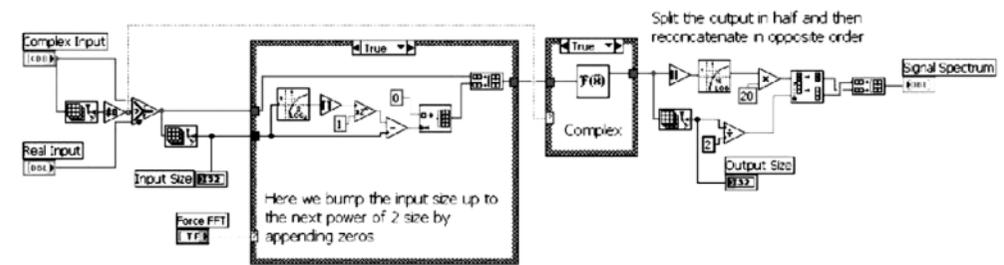


Рис. 6.3. Виртуальный прибор AdvFFT.vi

Другая известная особенность (или недостаток) БПФ – это то, что размер входной выборки не равен числу по основанию 2. В этом случае комплексное стандартное БПФ фактически является комплексным дискретным преобразованием Фурье (ДПФ). Если необходимо гарантировать, чтобы БПФ всегда выполнялось, необходимо дополнить входные данные, приближая размер выборки к следующему по величине числу по основанию 2, как показано в улучшенном ВП на рис. 6.3. Здесь имеет значение, что выполняется: БПФ или ДПФ. Из сравнительного анализа БПФ и ДПФ, приведенного в [4], можно сделать вывод, что вычисление дискретного преобразования Фурье вообще требует  $N^2$  комплексных умножений, где  $N$  – число точек в дискретном преобразовании Фурье. Наоборот, БПФ требует только  $(N/2) \log_2 N$  комплексных умножений. Для примера можно предположить, что размер выборки входного сигнала – 200 000 единичных отсчетов. Если необходимо исследовать спектр этого сигнала на основе дискретного преобразования Фурье, потребуется 40 000 000 000 комплексных умножений. Это довольно много операций для выполнения любым компьютером. Используя БПФ, можно сократить число операций комплексного умножения до числа, приблизительно равного 1 760 000. Имеется различие, большее чем в 10 000 раз, между этими двумя реализациями, и, таким образом, преимущество использования БПФ очевидно.

### 6.1.2. Улучшенное БПФ

На рис. 6.3 показана блок-диаграмма и передняя панель для улучшенного ВП БПФ, названного AdvFFT.vi. Как упоминалось выше, эта функция теперь имеет возможность заставить обычный LabVIEW БПФ *всегда* перед вычислением БПФ добавлять в конец входной выборки соответствующее число нулей, чтобы дополнить число  $N$  до следующего значения по основанию 2. Необходимо заметить также, что спектр, показанный на передней панели, теперь изменен так, что имеет центрированный относительно постоянной составляющей характер, и при увеличении количества отрицательных по частоте отброшенных составляющих способствует увеличению значений амплитуд составляющих с положительными частотами в правой части. В дополнение к этим особенностям ВП AdvFFT.vi также имеет способность принять реальный или комплексный входной сигнал и будет вызывать из библиотеки соответствующую БПФ-функцию. Кроме того, имеется псевдополиморфный вход ВП для обращения к реальному или комплексному БПФ и возможность отображать на передней панели результат смещения спектра. Можно вызывать этот специфический ВП регулярно и анализировать сигнал в различных точках системы связи, так же как и исследовать выходные сигналы фильтров (см. разд. 4).

Здесь есть необходимость уменьшить число головоломок, возникающих при использовании ВП AdvFFT. Это относится к генератору значений для частотной оси.

Таким образом, неизвестны отношения между компонентами сигнала, представленными на рис. 6.3, и связь с ними величин амплитуд спектральных составляющих. Известно [4], что ДПФ вычисляется так:

$$X(k) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi nk}{N}}, \quad k = 0, \dots, N-1. \quad (6.1)$$

При использовании выражения (6.1) можно увидеть, что с помощью ДПФ вычисляются спектральные составляющие входного сигнала  $x(n)$  в  $N$  эквидистантно расположенных по частоте точках, определенных как  $e^{-j(2\pi k/N)}$  для  $k = 0, \dots, N-1$ . Это означает, что будут получены  $N$  дискретных составляющих по частоте (обычно известные как точки ДПФ на частотах), в которых оцениваются спектральные составляющие  $x(n)$ . Необходимо иметь в виду, что, если нужно вычислить непрерывное преобразование Фурье от сигнала, необходимо оценить содержание спектральных составляющих сигнала в бесконечном числе точек по частоте. По этой причине во многих литературных источниках ДПФ именуется как дискретизация результата непрерывного преобразования Фурье. Поскольку мы только *смотрим* на спектральные составляющие выборки  $x(n)$  по  $N$  дискретным отсчетам по частоте, любое спектральное содержание  $x(n)$ , *не соответствующее* этим дискретным значениям, не будет учтено должным образом. Есть ли что-нибудь, что можно сделать в этом направлении? Один из способов разрешения этой проблемы состоит в том, чтобы расположить по частоте эти  $N$  дискретных отсчетов ближе друг к дру-

гу, увеличивая значение  $N$ . Это легко может быть обеспечено нулями, дополняющими входной сигнал, как это было сделано в AdvFFT VI при использовании функции БПФ. Так что здесь мы получаем два преимущества от дополнения нулями: повышение скорости БПФ и увеличение спектрального разрешения, вызванные использованием непрерывного преобразования Фурье при более высокой скорости дискретизации его результатов. Как всегда, будет компромисс между требуемым спектральным разрешением и временем вычисления БПФ. Таким образом, возникает вопрос: как точно связан множитель  $e^{-j(2\pi k/N)}$  с интервалом разрешения ДПФ по частоте, который имеет смысл в решаемой задаче? Здесь *необходимо* точно знать обычную выборку  $x(n)$  анализируемого сигнала. Выборка  $x(n)$  получена из входного аналогового сигнала дискретизацией с частотой  $f_s$ , и можно рассчитать интервал по частоте между последовательными отсчетами ДПФ:

$$\Delta f = \frac{f_s}{N}. \quad (6.2)$$

Используя выражение (6.2), можно легко выразить ось частоты для подготовленного спектра в ВП AdvFFT. На рис. 6.4 показан ВП FreqAxis.vi, и с помощью его единственной операции можно получить значения на оси частот, основанные на интервале между отсчетами, данным в выражении (6.2). Входы этого ВП:  $N$  (число отсчетов),  $f_s$  (обычная частота дискретизации) и булевы значения, чтобы решить, произвести одностороннюю или двухстороннюю ось частоты в спектре.

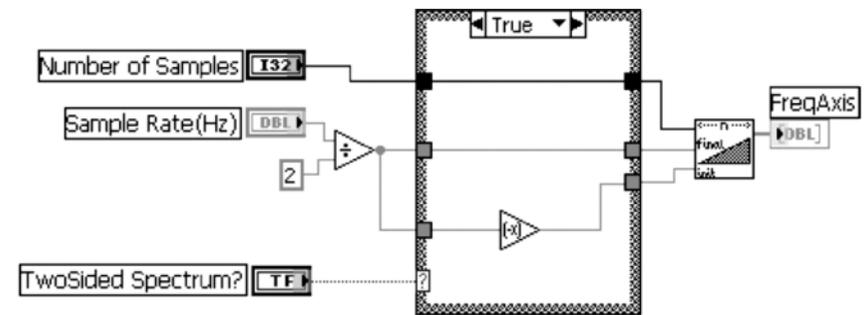
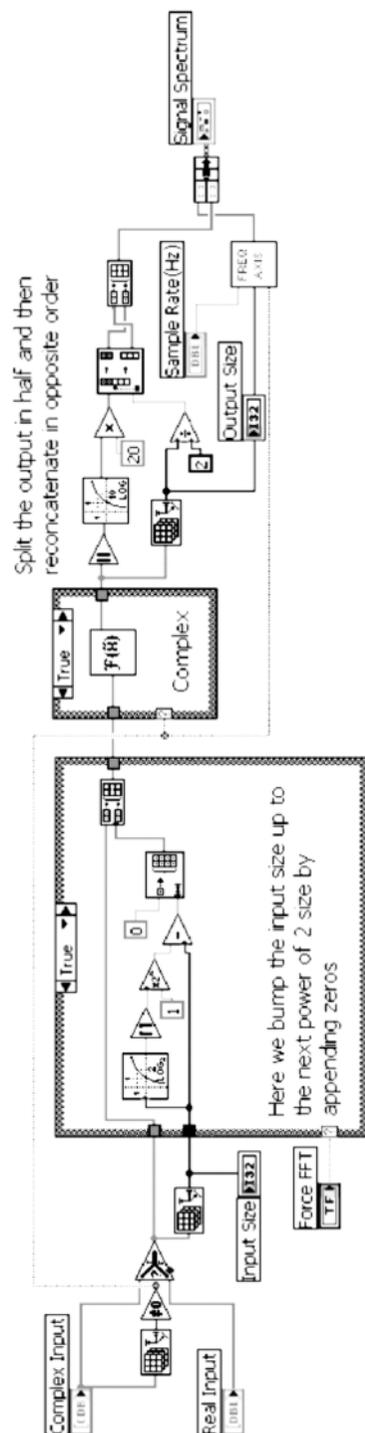


Рис. 6.4. ВП FreqAxis.vi для формирования оси частот, основанной на выражении (6.2)

Заключительная блок-диаграмма для AdvFFT.vi, дополненная ВП FreqAxis, показана на рис. 6.5. График спектра строится в декартовой системе координат XY, чтобы разместить входные отсчеты на двух осях. Теперь, если отредактировать соединительную панель, чтобы связать комплексный вход, реальный вход и ввод частоты дискретизации, а также подсоединить вход БПФ к входным терминалам ВП, а спектр сигнала с его выхода – к одному из терминалов выхода, можно использовать AdvFFT в других ВП в виде блока (sub-VI).

Рис. 6.5. Окончательная форма блок-диаграммы ВП `advFFT.vi`

## 6.2. Анализ результатов ДПФ

Столь же важным является способность понять результаты, которые получаются после использования ДПФ. Ранее упоминалось, что произвольные по частоте спектральные компоненты входного сигнала  $x(n)$  не всегда точно соответствуют одному из  $N$  центров отсчетов ДПФ по частоте и не будут должным образом проявляться в результате использования ДПФ. Смещение спектральной составляющей относительно точек по частоте через интервал разрешения приводит к следующим последствиям: амплитуда составляющей не будет правильно отражена в спектре из-за формы окна дискретизации входной выборки, что приведет к появлению боковых лепестков в спектре, и энергия составляющей распределяется по *всем* другим точкам частоты ДПФ. Такое явление называется спектральной утечкой [8].

Проанализируем эти два явления по одному, начиная с последнего.

### 6.2.1. Спектральная утечка

Спектральная утечка является распределением энергии из одной точки на оси частот по другим частотным составляющим ДПФ. Утечка проявляется из-за недостатка ортогональности между некоторыми компонентами частоты в исходном сигнале и множеством базисных функций (векторов) в ДПФ [8]. ДПФ, по существу, обеспечивает вычисление проекций входного сигнала  $x(n)$  на ортогональный набор базисных функций ДПФ, составленный из синусов и косинусов по  $N$  дискретным отсчетам частоты, равномерно распределенным на интервале от 0 до  $f_s$ . При этом расположение произвольной спектральной составляющей сигнала  $x(n)$  с частотой, *не точно* соответствующей одному из  $N$  дискретных значений частоты в семействе базисных функций ДПФ, приведет к спектральным составляющим, отличным от нуля *на всех* значениях частоты в заданном базисе [8]. Это явление будет иметь место, если не делается ничего, чтобы уменьшить амплитуды спектральных составляющих сигнала  $x(n)$  с частотами, не соответствующими сетке частот ДПФ, в результате чего *полное* вычисление ДПФ приводит к неточным результатам. Способ, которым уменьшается произвольная часть спектральных составляющих сигнала, заключается в использовании фильтрации. Существует класс операций фильтрации, обычно применяемых только для этой цели и известных как применение взвешивающих функций.

### 6.2.2. Дискретизация формы окна

Можно было бы задать себе вопрос: какое окно использовать? Действительно, даже если не использовать никакого окна, оно все равно уже применено: это окно дискретизации выборки. Данное окно имеет прямоугольную форму по времени, и оно подразумевается даже тогда, когда нет необходимости использовать его при обработке.

Существование подразумеваемого окна основано на том факте, что при ДПФ используется конечная сумма.

Для учета в обработке сигнала взвешивающей функцией необходимо изменить выражение для ДПФ (6.1), чтобы включить окно, обозначаемое как  $w(n)$ :

$$X(k) = \frac{1}{N} \sum_{n=0}^{N-1} x(n)w(n)e^{-j\frac{2\pi nk}{N}}, \quad k=0,1,2,\dots, N-1. \quad (6.3)$$

Из свойств преобразования Фурье известно, что умножению в области времени соответствует круговая свертка в области частот. Можно интерпретировать окно в области времени как фильтр в области частот. В этом случае окно есть не что иное, как фильтр, сосредоточенный в каждом дискретном отсчете по частоте ДПФ. Этот фильтр обеспечивает значительное уменьшение амплитуды для любых спектральных составляющих сигнала  $x(n)$  вдали от дискрета по частоте. Хотя прямоугольное окно имеет очень узкий главный лепесток, его главный недостаток заключается в том, что боковые лепестки соответствуют только  $-13,5$  дБ относительно главного лепестка и не производят достаточного уменьшения спектральной утечки, описанной выше. Так как нет другого выбора, кроме как использовать окно, можно выбрать окно, требуемое для решения задачи. Произведены большие исследования в области разработки окон для ДПФ (для различных применений), и в результате есть множество окон для выбора, среди них окна Хэмминга (Hamming), Ханна (Hann), Блэкмана – Хэрриса (Blackman – Harris) и др.

Как правило, выбор окна производится на основе компромисса между высотой боковых лепестков и шириной главного лепестка. Выбранное окно должно наилучшим образом подходить для данного сигнала. Окно типа А может особенно хорошо подходить для обработки речи и не может быть желательным для цифровых систем связи. Некоторые окна имеют основание, в то время как другие стремятся к нулю на границах. Различные окна отличаются, в основном, величиной ослабления боковых лепестков и шириной главного лепестка.

Для более полного ознакомления со списком окон с более детальным пояснением их параметров (высот боковых лепестков и ширины главного лепестка) можно обратиться к [9]. С помощью рис. 6.6 можно визуально ознакомиться с результатом воздействия окна на входной сигнал в спектральной области.

На рис. 6.6 показаны спектры двух различных функций: прямоугольного окна (штриховая линия) и окна Ханна (Hann) (сплошная линия). Ось частоты на рисунке проградуирована в относительных  $f_s/N$  единицах разрешения по частоте базового ДПФ. Как упоминалось выше, умножение отсчетов сигнала на функцию окна в области времени соответствует круговой свертке в области частоты. Это означает, что спектр окна будет распределен по всем дискретным частотам анализа ( $nf_s/N$  для  $n = 0, 1, 2, \dots, N-1$ ) после применения ДПФ. Можно отметить, что главный лепесток окна Ханна (Hann) охватывает первый и второй отсчеты ДПФ по частоте (один дополнительный отсчет с каждой стороны при анализе двухстороннего спектра), но боковые лепестки уменьшаются быстрее по сравнению с прямоугольным окном. При осуществлении ДПФ это означает, что произволь-

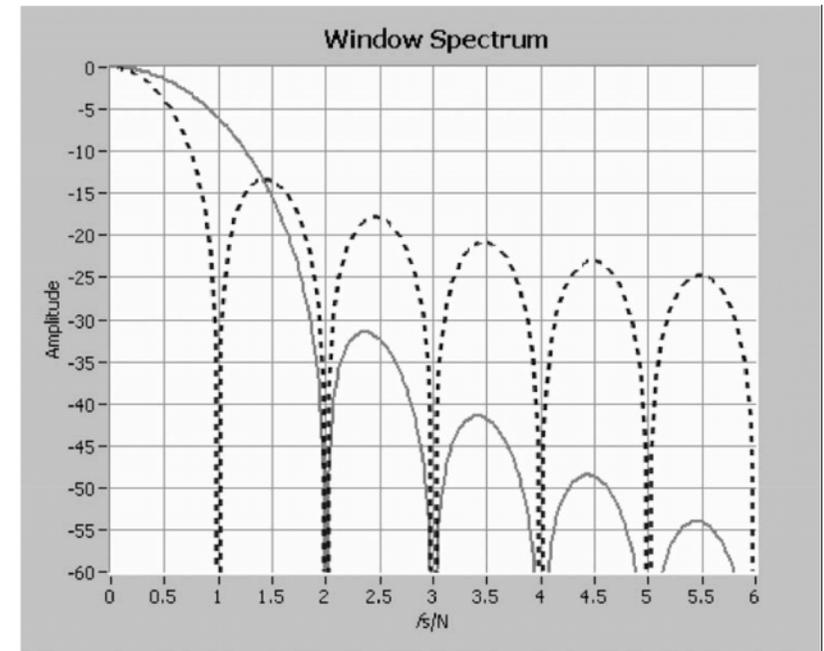


Рис. 6.6. Спектры прямоугольного окна (штриховая линия) и окна Ханн (сплошная линия)

ная энергия сигнала, содержащаяся в соседнем дискрете, выше, а в следующем дискрете – немного ниже и добавляется к спектральной составляющей ДПФ, рассчитанной для данного дискрета (в этом причина спектральной утечки). Преимущество окна Ханна (или любого непрямоугольного окна) – это то, что вклады энергии от дискретов далее чем через один дискрет существенно уменьшены по сравнению со стандартным прямоугольным окном.

И, наконец, понятие потерь, обусловленных неравномерностью отсчетов на краях выборки, также проиллюстрировано на рис. 6.6. Из более раннего обсуждения следует, что амплитуды спектральных составляющих сигнала далеко от центров дискретов *сформированы* окном дискретизации. При использовании, например, прямоугольного окна спектральная составляющая по частоте посередине между двумя центрами отсчетов будет уменьшена почти на 5 дБ. Имеет ли на практике значение то, что выходные отсчеты ДПФ уменьшается по амплитуде при некоторых частотах? Такие изменения имеют значение в применениях типа обработки речи, где амплитуды некоторых компонентов частоты должны быть измерены точно. Важно держать эти изменения в памяти при использовании дискретного преобразования Фурье.

Теперь, когда разъяснены результаты применения ДПФ, можно иначе оценивать итоги использования взвешивания в LabVIEW, чтобы улучшить результат применения операции ДПФ в ВП AdvFFT.vi.

На рис. 6.7 показано, где расположены окна, используемые LabVIEW в палитре «Функции» (Functions). LabVIEW для обработки сигналов включает некоторые из самых распространенных функций: окна Ханна (Hann), Хэмминга (Hamming), Блэкмена (Blackman), Блэкмана – Хэрриса (Blackman-Harris) и Кайзера (Kaiser), треугольник.

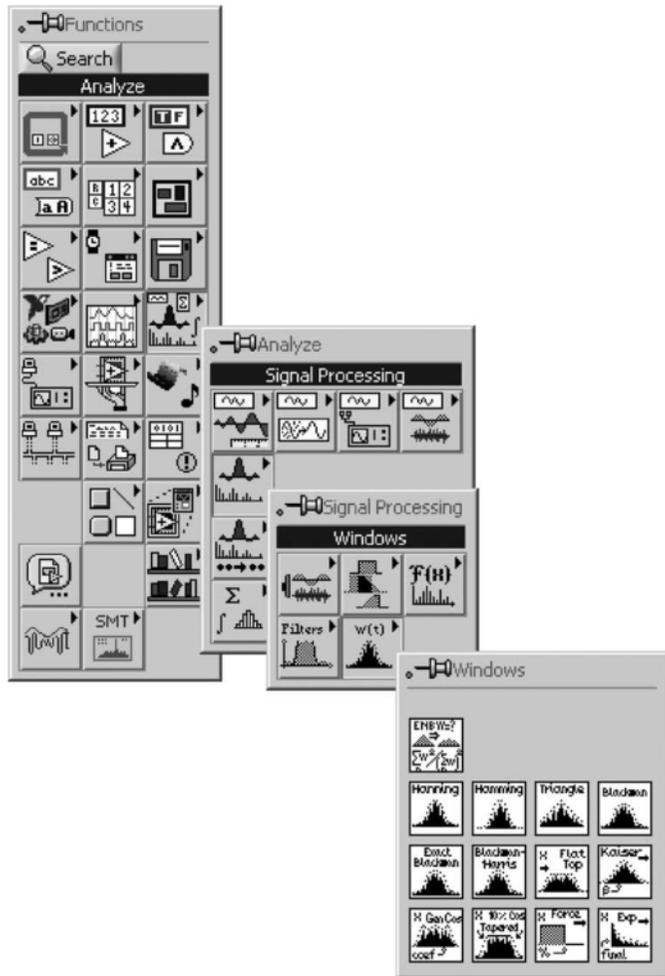


Рис. 6.7. Весовые функции в LabVIEW

Конечно, можно всегда добавлять в ВП оригинальный алгоритм, если требуется специализированная функция окна, используя следующие примеры и заменяя коэффициенты окна необходимыми значениями.

Если проанализировать блок-диаграмму любой из весовых функций на рис. 6.7, можно видеть, что все они, по существу, одинаковы. Каждый из них содержит мо-

дули ВП (sub-VIs) с именами General Cosine Window.vi и Windowcoeffs.vi. Все прежде упомянутые окна сформированы в результате суммирования взвешенных косинусов.

Выбор окна эквивалентен отбору множества коэффициентов с помощью переключателя в ВП Windowcoeffs.vi. Можно использовать этот блок ВП (sub-VI) самостоятельно или вставить его в блок-диаграмму, как показано в верхней части рис. 6.8, тем самым подвергая весовой обработке входной сигнал перед отправкой его в ВП AdvFFT.vi с целью уменьшения спектральной утечки при осуществлении ДПФ, а также улучшения разрешающей способности ДПФ двух близких по частоте спектральных составляющих. На рис. 6.8 показан спектр входного сигнала без дополнительного окна (т. е. применено прямоугольное окно) и с дополнительным окном Блэкмана – Хэрриса (Blackman – Harris). Штриховая линия – применено прямоугольное окно и сплошная линия – тот же самый сигнал с дополнительным окном Блэкмана – Хэрриса (Blackman – Harris). Некоторые выво-

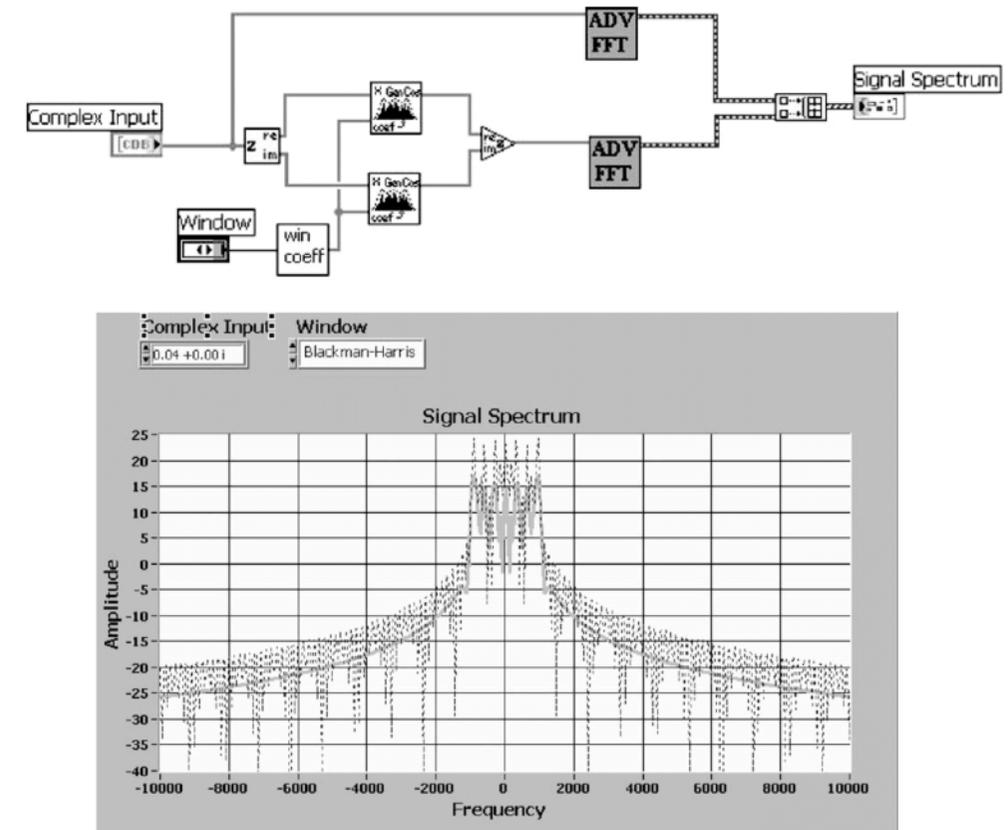


Рис. 6.8. Эффект от весовой обработки входных отсчетов прямоугольным окном (пунктир) и окном Блэкмана – Хэрриса (Blackman – Harris) (сплошная линия)

ды можно сделать уже сейчас об эффекте применения окна Блэкмана – Хэрриса. Во-первых, пиковое значение спектра взвешенного сигнала на (5–7) дБ меньше, чем это же значение без взвешивания. Во-вторых, спектр взвешенного сигнала (сплошная линия) заметно более гладок, чем спектр невзвешенного сигнала. В этом случае окно Блэкмана – Хэрриса имеет боковые лепестки, уровень которых ниже приблизительно на 61 децибел. Это подавление боковых лепестков очистило сигнал, показанный на рис. 6.8, и обеспечило исследователю более гладкий участок спектра.

Итак, теперь известно немного больше о функциях окна и известно также, как их применить в LabVIEW. Возникает существенный вопрос: какое окно выбрать?

Ранее обсуждался компромисс между выбором уровня ослабления боковых лепестков и шириной главного лепестка, но это не может быть единственным показателем при анализе результатов спектральных преобразований. Некоторые окна могут иметь уклон боковых лепестков, которые уменьшаются со скоростью 6 дБ на октаву, а спектры других окон могут иметь уклон 18 дБ на октаву. Важно иметь в виду в этом случае, что какое бы окно ни было выбрано, оно должно быть лучше, чем прямоугольное окно. Для более полного сравнения различных окон можно обратиться к [8].

### 6.3. Спектральные преобразования высокого уровня

В дополнение к функциям низкого уровня для обработки сигнала, описанным в подразд. 6.1, пакет LabVIEW также содержит некоторые полезные ВП более высокого уровня, которые расширяют основные функциональные возможности спектральных инструментов низкого уровня с особенностями типа весовой обработки, усреднения и вычисления спектральной плотности. Местоположение этих функций в библиотеке LabVIEW показано на рис. 6.9.

Если рассмотреть высокоуровневые функции подпалитры измерений формы колебания, можно видеть, что на самом низком уровне они все построены на основе БПФ или ВП Power Spectrum, показанного на рис. 6.1. Необходимо пройти несколько слоев в глубину палитры, чтобы видеть эти основные функции, но они являются ядром каждой из функций высокого уровня. Остальные коды LabVIEW связаны с некоторым форматированием и распаковыванием групп кодов, а также пользовательским управлением. Все эти функции LabVIEW предназначены для того, чтобы заинтересовать пользователя, и цель NI состоит в том, чтобы функции были выражены в простом виде для поддержки различных применений.

Большинство функций, упомянутых выше, предназначено для поддержки быстрых измерений параметров и характеристик сигналов, полученных с карты АЦП. Это означает, что данные сигналы будут иметь реальный характер, и поэтому используемые функции обычно основаны на обработке только реальных входных отсчетов в области времени. Как только входной реальный сигнал преобразовывается в комплексный, многие из используемых функций становятся непригод-

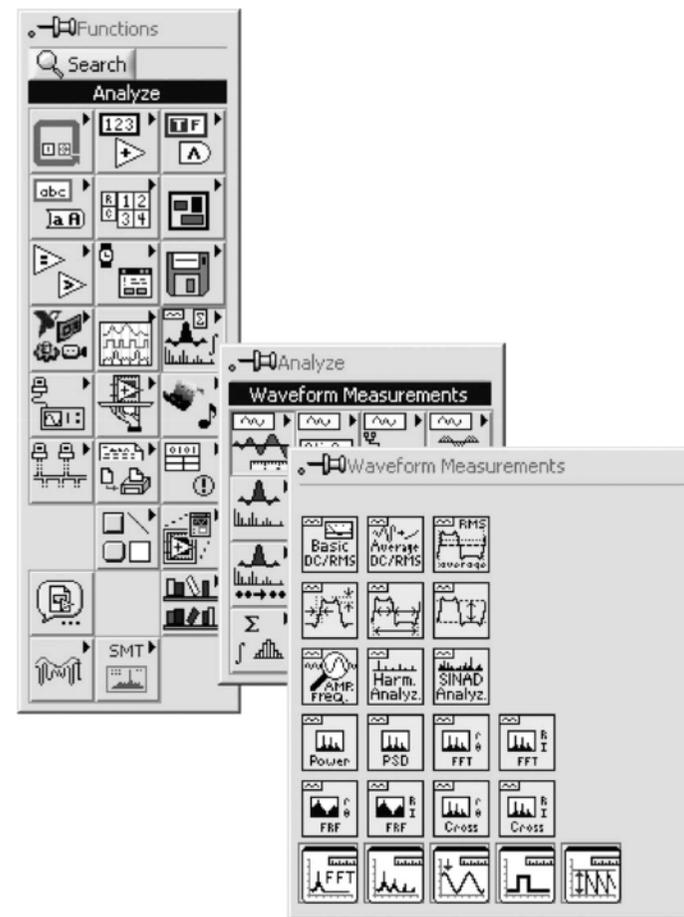


Рис. 6.9. Инструменты высокого уровня (High-level) для измерения формы колебаний

ными для применения в последующих узлах системы связи. В разд. 3 были показаны некоторые примеры реальных входных сигналов, полученных с выхода карты АЦП. Спектр этих сигналов был показан на передней панели ВП при измерении спектральной плотности мощности в соответствии с блок-диаграммой, показанной в нижней части рис. 6.10.

Некоторые из удачно выполненных функций для измерения формы колебаний позволяют производить усреднение, удерживать пиковое значение и обработку входных отсчетов весовой функцией. Эти возможности особенно полезны, если необходимо просто показать или записать прямые измерения на основе спектра входного сигнала без любой дальнейшей обработки. Как и все ВП, разработанные корпорацией National Instruments, эти инструменты особенно полезны как руководство по программированию, чтобы обеспечить построение пользовательских специальных приборов анализа спектра.

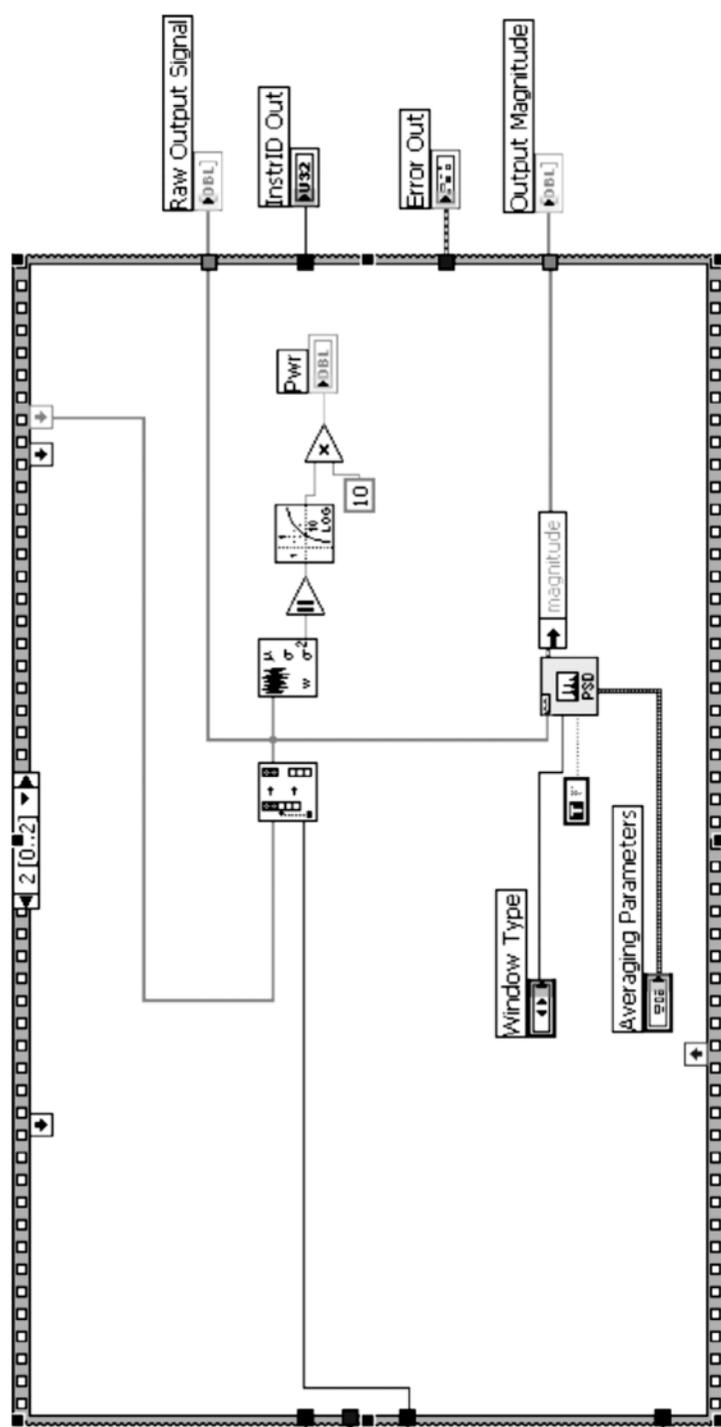


Рис. 6.10. Блок-диаграмма ВП SampleLoop.vi, показывающего спектральную плотность мощности

## 6.4. Добавление обычных кодов С к LabVIEW

В этом подразделе кратко описаны способы распространения собственных LabVIEW возможностей на функции пользователя в кодах С. Причина, по которой эта тема рассматривается, состоит в том, что многие инженеры могут иметь собственные испытанные и истинные функции в кодах С для того, чтобы произвести дискретное косинусное преобразование, или вейвлет-преобразование, или, возможно, другой тип преобразований, которые разработаны и составляют собственность пользователя. Это ни в коем случае не подразумевает, что только спектральная обработка в кодах С должна быть включена в пользовательский ВП, однако эти типы алгоритмов обычно в вычислительном отношении интенсивны и могут быть оптимизированы в кодах С. Есть также другие причины, по которым пользователь может желать включить функцию в кодах С в LabVIEW: скорость вычислений, компактность, а также способность к развитию.

Ниже описаны основные шаги при включении программы в кодах С для использования БПФ.

Первый шаг, с которого необходимо начинать, – наличие соответствующей динамической библиотеки связи (DLL) для функции, которую пользователь желает объединить с LabVIEW. Для этих целей создан поисковый вебсайт библиотеки FFT с очень простым интерфейсом, который был создан Murphy McCauley и может быть загружен по адресу [www.fullspectrum.com/deeth](http://www.fullspectrum.com/deeth).

Чтобы включить программу в кодах С в LabVIEW, на первом шаге необходимо исследовать опытный образец этой программы в виде заголовка файла (в этом случае FFTLib.h). В документации для библиотеки DLL можно заметить, что автор обеспечил некоторую оболочку функции, названную `fftDouble` и `fftFloat`, чтобы загрузить DLL и вызвать необходимые функции `fft_double()` и `fft_float()`. В этом случае LabVIEW будет обращаться для загрузки в библиотеку, так что можно устранить любые запросы к оболочкам и обращаться прямо к реальной функции `fft_double()`. Можно удалить ненужные данные из файла `FFTLib.c` и увидеть, что функция `fftDouble()` просто передает все свои аргументы функции `fft_double()`. При наличии этого файла в памяти компьютера можно решить, как вызвать функцию `fft_double()` для рассмотрения функции-прототипа из библиотеки `FFTLib.h`, представленной в следующей таблице.

Выборка кодов из `FFTLib.h`

BOOL <code>fftDouble()</code>		
Unsigned	NumSamples	/* must be a power of 2 */
Int	InverseTransform	/* 0 = forward FFT, 1 = inverse FFT */
Double	*RealIn	/* array of input's real samples */
Double	*ImagIn	/* array of input's imag samples */
Double	*RealOut	/* array of output's real */
Double	*ImageOut);	/* array of output's imaginaries */

Теперь, когда известно, как вызвать функцию FFT, можно работать в LabVIEW. Первое необходимое действие состоит в том, чтобы поместить в ВП запрошенный

узел функции из библиотеки. Этот узел можно найти в продвинутой подпалитре, показанной на рис. 6.11.

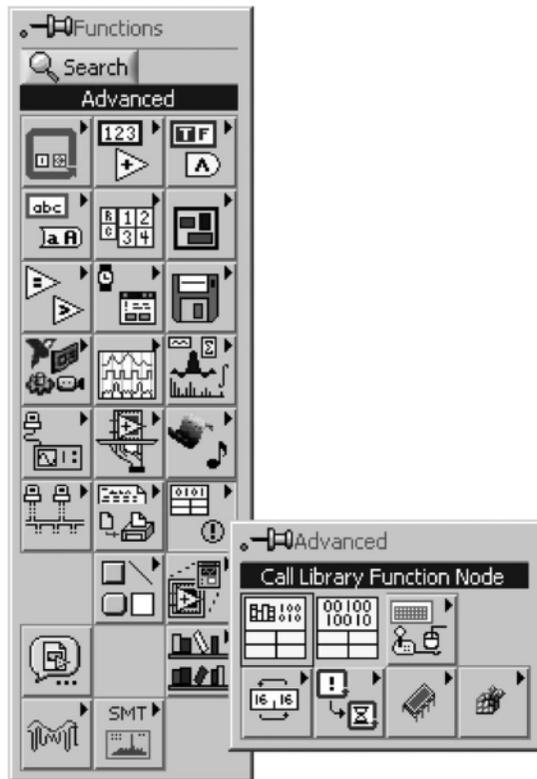


Рис. 6.11. Продвинутая подпалитра функций

После установки запрошенного узла надо щелкнуть правой кнопкой мыши на нем и выбрать функцию из возникшего внизу меню, как показано на рис. 6.12. Это «выскочившее» окно позволяет обозначить функцию для определения пути, следуя которому LabVIEW может ее найти. Большая часть из требуемой информации является прямой. Очевидно, необходимо будет указать в LabVIEW на соответствующую библиотеку DLL и выбрать имя функции. Узел проверки определяет установку всех входных параметров согласно функции-прототипу, указанной в приведенной выше выборке кодов.

Необходимо заметить, что, поскольку добавляется каждый параметр с помощью кнопки Add-a-parameter-after, LabVIEW автоматически строит функцию-прототип в базовом окне. Быстрый анализатор должен подтвердить, поступила ли информация о функции, и затем можно начать связывать входы и выходы.

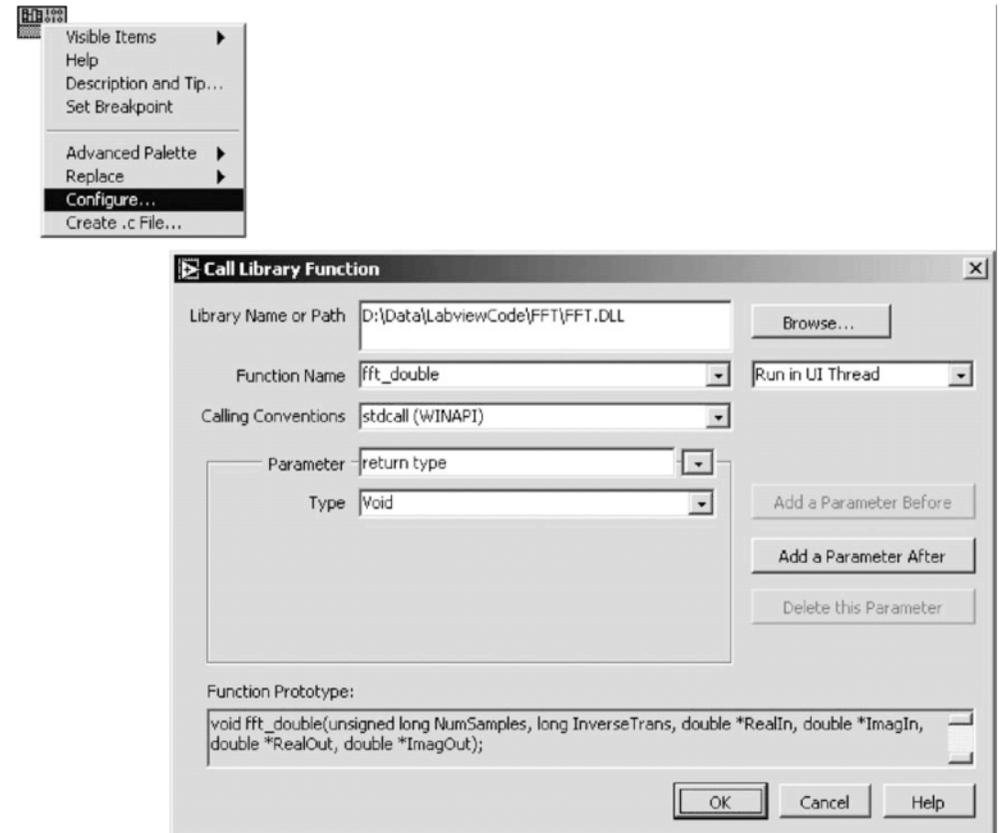


Рис. 6.12. Конфигурация узла вызова функции из библиотеки

Законченный ВП FFT DLL показан на рис. 6.13. Устанавливаются входные параметры: число точек, направление преобразования (прямое или обратное), затем маркируются указателями реальная часть входа и мнимая часть входа. LabVIEW обращается к форме переданных аргументов, основанных на функции-прототипе, и все пользователи должны передать соответствующие сигналы на терминалы. Что касается выходных сигналов, то должно быть зарезервировано место для реальных и мнимых множеств с нулевыми значениями для каждого из них.

Способность включать DLLs непосредственно в ВП делает обработку сигнала в LabVIEW универсальной. Как правило, инженерами используются традиционные процедуры в кодах C, которые являются алгоритмически интенсивными, но не всегда имеют развитый пользовательский интерфейс. Используя LabVIEW для управления и реализации графических возможностей, можно построить много мощных инструментов обработки сигнала на основе традиционных DLLs.

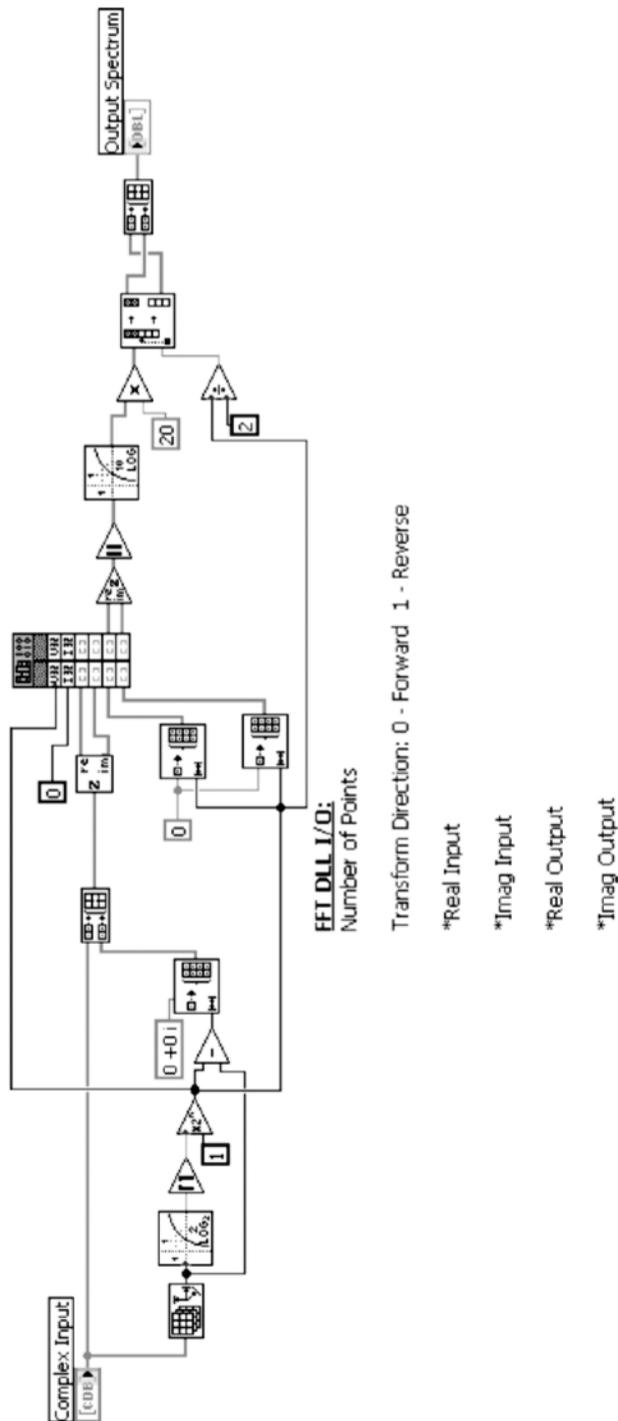


Рис. 6.13. Блок-диаграмма завершеного узла FFT DLL для вызова функции из библиотеки

## 6.5. Инструменты (Toolset) для спектральных измерений

В этом подразделе рассматривается один из дополнительных наборов инструментов NI Spectral Measurements (SMT). Этот набор оформлен как часть пакета анализатора PXI-5660 RF, рассмотренного в разд. 2. Пакет инструментов SMT содержит функции для измерения мощности в полосе, занятую полосу пропускания, внеполосные спектральные составляющие канала, находит как спектральные пики, так и усредненный спектр сигнала. Имеется также некоторый очень удобный ВП, который фиксирует и отображает спектр входного сигнала и позволяет устанавливать несущую частоту, диапазон и усредненные параметры.

На рис. 6.14 показано местоположение подпалитры SMT в пределах управления LabVIEW.

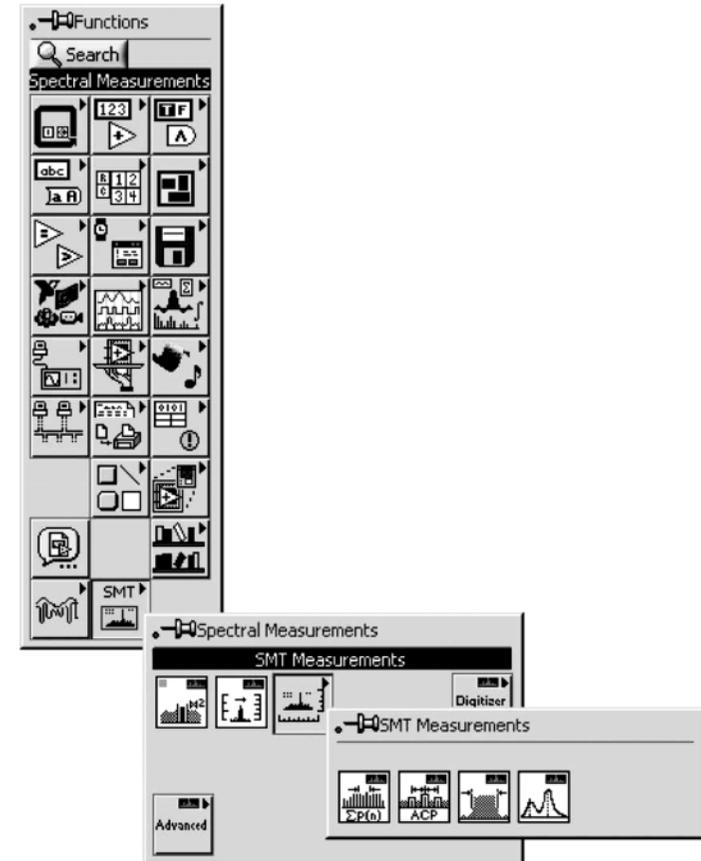


Рис. 6.14. Местоположение инструментария спектральных измерений

Все эти функции тесно связаны с аппаратными средствами 5660 ЭВМ, и по этой причине NI объединяет SMT и 5660. Рассматривая подпалитру SMT, можно заметить много ВП низкого уровня, которые позволяют формировать Tuner NI (PXI-5600) и NI Scope (PXI-5620) независимо от функций SMT или RFSA. Как и во многих других случаях, корпорация NI снабдила инструменты функциями SMT, предназначенными для пользователя, чтобы легко формировать аппаратные средства ЭВМ 5660 для возможных измерений, но пользователи также могут строить свои собственные средства, используя функции низкого уровня.

## Резюме

В этом разделе рассмотрены некоторые из особенностей спектрального анализа, встроенного в пакет LabVIEW. При исследовании возможностей LabVIEW были показаны некоторые специфические недостатки ДПФ: потери информации из-за неравномерностей на краях выборки (scalloping), спектральная утечка и ограничение по частоте результатов спектрального анализа. Так как ДПФ есть дискретизация непрерывного преобразования Фурье, эти недостатки не могут быть устранены, но их отрицательное влияние может быть уменьшено за счет надлежащего использования весовых функций (windowing) и функций калибровки ДПФ. Также представлен ВП, названный AdvFFT.vi, для расширения функциональных возможностей основной операции ДПФ за счет использования БПФ и изменения выходного спектра для целей отображения. В заключение этого раздела было показано, как используется DLL-библиотека в ВП LabVIEW. Это расширяет функциональные возможности обработки в LabVIEW, поскольку есть огромное число библиотек обработки сигналов, доступных в Интернете. Очевидно, что LabVIEW имеет некоторые очень мощные инструменты спектрального анализа, применение которых для анализа и синтеза цифровых фильтров, а также для многоскоростной обработки дискретных и цифровых сигналов будет показано в следующих разделах.

## Окна сглаживания (Smoothing windows)



В этом разделе подробно описывается явление, называемое *растеканием спектра*, поясняется, как использовать окна сглаживания, чтобы уменьшить растекание спектра, приводятся различные типы сглаживающих окон, объясняется, как правильно выбрать тип сглаживающего окна, описываются различия между сглаживающими окнами, используемыми для спектрального анализа, и сглаживающими окнами, используемыми для разработки коэффициентов цифрового фильтра, а также поясняется важность масштабирования окон сглаживания.

Применение окна сглаживания к сигналу – это его обработка с помощью окна. Используется такая обработка для выполнения следующих операций анализа:

- определение продолжительности наблюдения;
- уменьшение растекания спектра;
- отделение в спектре сигнала гармонической составляющей с малой амплитудой от составляющей с большой амплитудой, частоты которых очень близки друг к другу;
- определение коэффициентов КИХ-фильтра.

Окна в ВП обеспечивают простой метод улучшения спектральных характеристик сигнала по его выборке. Можно использовать NI Example Finder, чтобы найти примеры использования Окон (Windows VIs) в библиотеке виртуальных приборов LabVIEW.

### 7.1. Растекание спектра

Согласно теореме дискретизации Шеннона (Котельникова) (Shannon Sampling Theorem), можно полностью восстановить непрерывный во времени сигнал из дискретного с эквидистантными по времени единичными выборками, если самая высокая частота составляющей во временной функции сигнала меньше половины частоты дискретизации выборки. Половина частоты дискретизации выборки рав-

няется частоте Найквиста (Nyquist). Выборочная теорема Шеннона является мостом между непрерывными во времени сигналами и цифровыми сигналами во времени. Можно обратиться к разд. 1 для получения дополнительной информации о теореме дискретизации Шеннона.

В практических приложениях, производя выборку сигнала и оцифровывая сигнал во времени, ограничивают высокочастотные составляющие сигнала, даже когда соблюдаются теорема дискретизации Шеннона и условия дискретизации. Даже когда дискретизация выполняется по критерию Найквиста, ограничения выборки могут вызвать утечку энергии, называемую растеканием спектра. Поэтому даже при условии, что используются надлежащие методы дискретизации сигнала, измерения не могут привести к масштабируемому одностороннему спектру из-за его растекания. При растекании спектра энергия спектральной составляющей сигнала на одной частоте как бы просачивается на все другие частоты.

Растекание спектра следует из предположения, что в алгоритмах БПФ и ДПФ отсчет времени точно повторяется в течение всего времени. Таким образом, сигналы в отсчетах по времени являются периодическими с промежутками, которые соответствуют длине отсчетов времени (интервалу выборки). Когда используется БПФ или ДПФ, чтобы измерить содержание составляющих по частоте в исходных данных, преобразования предполагают, что конечный набор данных соответствует одному периоду периодического сигнала. Поэтому ограниченность времени получения выборки приводит к появлению «обрезанной» формы колебания с отличающимися спектральными характеристиками относительно исходного непрерывного по времени сигнала, и на границах выборки могут появиться резкие (скачкообразные) изменения в измеренных данных. Резкие переходы приводят к нарушению непрерывности. Для иллюстрации на рис. 7.1 представлены нарушения непрерывности на границах выборки.

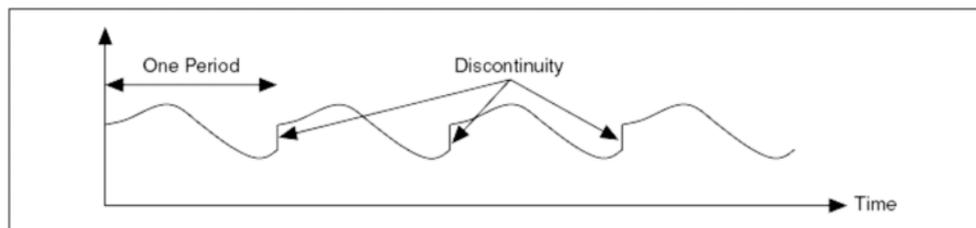


Рис. 7.1. Периодическая форма колебания, созданная из выборки:  
One Period – длительность выборки (один период);  
Discontinuity – разрыв непрерывности

Нарушения непрерывности, показанные на рис. 7.1, обуславливают утечку спектральной информации: спектр расширяется искусственно, появляются составляющие, отсутствующие в исходном сигнале. Растекание спектра способствует образованию дискретно-разового спектра, который появляется как смазанная версия первоначального непрерывно-разового спектра (без усреднения).

### 7.1.1. Дискретизация целого числа периодов

Растекание спектра происходит только тогда, когда выборка данных состоит из нецелого числа периодов. На рис. 7.2 показано синусоидальное колебание, дискретизированное на целом числе периодов, и преобразование Фурье синусоидального колебания.

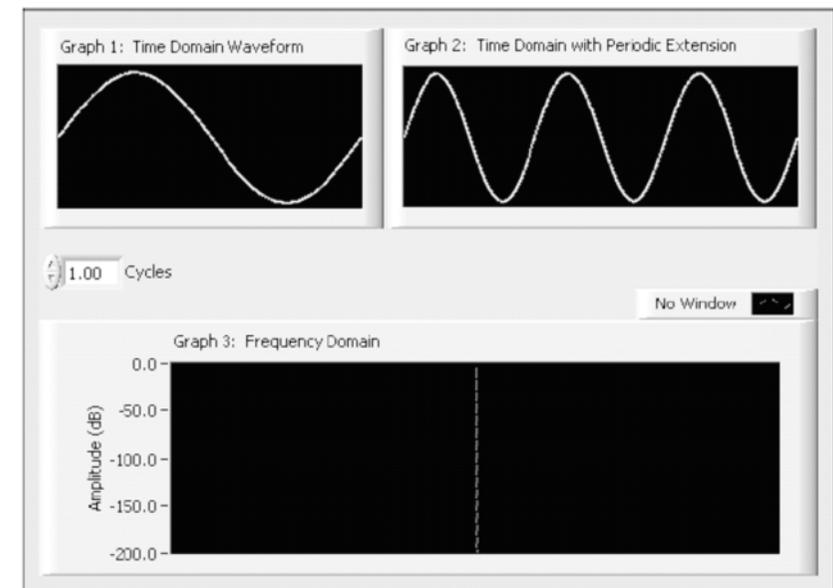


Рис. 7.2. Синусоидальное колебание и связанное с ним преобразование Фурье

На рис. 7.2 график 1 (Graph 1) показывает выборочную форму колебания в области времени. График 2 (Graph 2) показывает периодическую форму синусоидального колебания, полученного из графика 1 (Graph 1). На графике 2 (Graph 2) исходное колебание дополняется бесконечными повторениями его компонентов, соответствующих выборке, чтобы подтвердить предположение о периодичности этого колебания для преобразования Фурье. График 3 (Graph 3) показывает спектральное представление формы колебания.

Поскольку отсчет времени на Graph 2 является периодическим без нарушения непрерывности на границах периодов, его спектр появляется в Graph 3 как одиночная линия, показывая частоту синусоидального колебания. Форма колебания на Graph 2 не имеет никакого нарушения непрерывности, потому что набор данных образован от целого числа периодов (в этом случае одного).

Следующие методы дискретизации гарантируют, что всегда получится целое число периодов в пределах выборки:

- осуществление синхронной выборки относительно измеряемого сигнала, в результате чего можно преднамеренно получить целое число периодов;

- фиксирование переходного сигнала, который полностью соответствует сигналу внутри отсчетов времени (выборки).

### 7.1.2. Дискретизация с нецелым числом периодов

Обычно априори неизвестный измеряемый сигнал является стационарным.

Стационарный сигнал существует до, в течение и после интервала сбора данных.

При измерении стационарного сигнала можно гарантировать, что получится целое число периодов. Если отсчеты времени содержат нецелое число периодов, растекание спектра происходит из-за того, что спектральные составляющие для нецелого числа отсчетов сигнала не соответствуют точно одной из линий спектра на оси частот. Растекание спектра искажает измерение таким образом, что энергия от данной спектральной составляющей, кажется, распространяется по смежным точкам частоты или линиям спектра, приводя к «размазыванию» спектра сигнала. В этом случае можно использовать окна сглаживания, чтобы уменьшить нежелательные эффекты, появляющиеся после выполнения БПФ при нецелом числе периодов.

Из-за предположения о периодичности формы колебания искусственные нарушения непрерывности между последовательными периодами выборок происходят, когда производится выборка нецелого числа периодов. Искусственные нарушения непрерывности проявляются в виде сравнительно высоких частот в спектре сигнала, которые не присутствуют в первоначальном сигнале. Высокие частоты составляющих, обусловленных нарушением непрерывности, могут быть намного выше, чем частота Найквиста, и появляются боковые составляющие где-то между частотами 0 и  $f_s / 2$ . Так проявляется растекание спектра. Спектр, который получается при использовании ДПФ или БПФ, есть «размазанные» версии спектра, а не фактический спектр исходного сигнала.

На рис. 7.3 представлено синусоидальное колебание, выборка из которого содержит нецелое число периодов, и преобразование Фурье этого колебания.

На рис. 7.3 временная диаграмма (Graph 1: Time Domain Waveform) состоит из 1,25 периода синусоидального колебания. На временной диаграмме (Graph 2: Time Domain with Periodic Extension) представлены повторения исходного отрезка сигнала во времени с периодом, равным длительности выборки, для подтверждения предположения о периодичности исходного сигнала при использовании ДПФ. Диаграмма (Graph 3: Frequency Domain) показывает спектральное представление формы колебания. Энергия как бы «просачивается» от одной линии БПФ к другим, вызывая спектральную утечку.

Можно также заметить, что растекание спектра происходит из-за конечного числа отсчетов по времени исходного сигнала.

Чтобы преодолеть растекание спектра, можно взять теоретически бесконечное число отсчетов по времени от  $-\infty$  ( $-\infty$ ) до  $+\infty$  ( $+\infty$ ). С бесконечным числом отсчетов по времени производится БПФ, в результате определяется одна единственная линия на частоте, равной частоте спектральной составляющей входного сигнала. Однако сигнала, распространяющегося в течение бесконечного

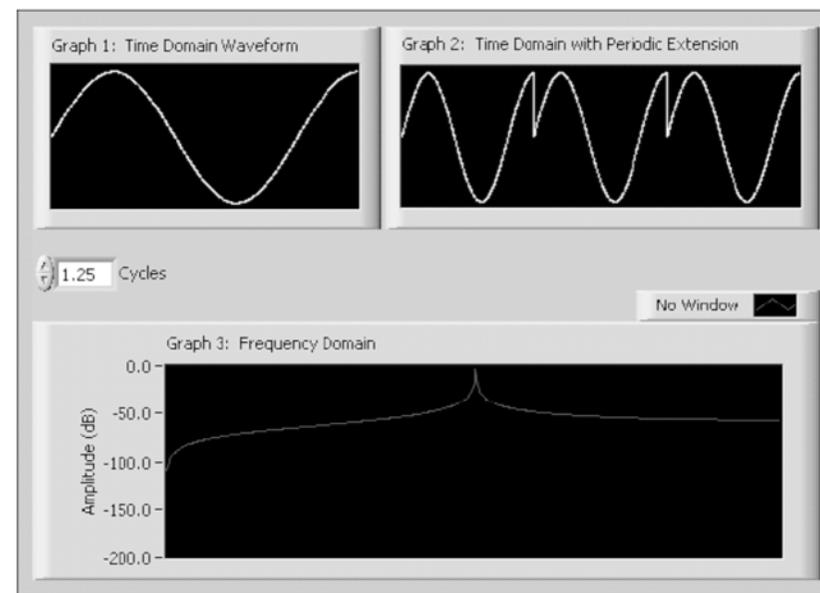


Рис. 7.3. Спектральное представление, когда выборка содержит нецелое число периодов

времени, практически не существует. Чтобы преодолеть ограничения, возникающие из-за конечного числа отсчетов времени, для уменьшения спектральной утечки используется работа с окнами.

В дополнение к появлению ошибок в измерениях амплитуды растекание спектра может затенить смежные по частоте спектральные составляющие с малыми амплитудами. На рис. 7.4 представлен спектр двух близких по частоте спектраль-

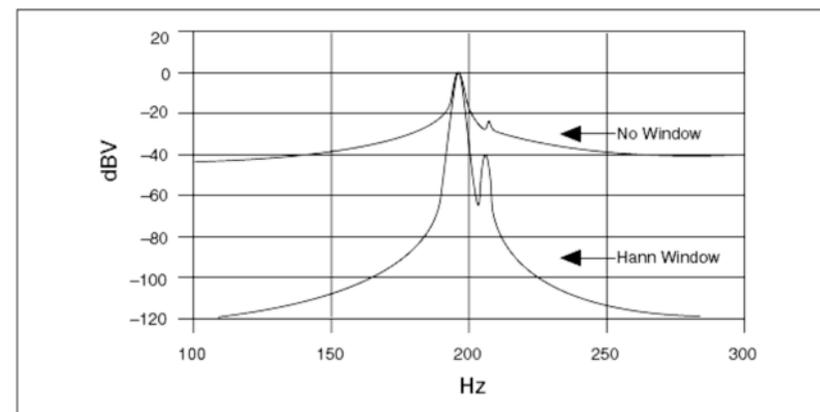


Рис. 7.4. Растекание спектра, затеняющее смежные спектральные составляющие

ных составляющих, когда не используется окно сглаживания и когда применяется окно Ханна.

Второй пик на графике спектра выделяется более заметно для сигнала с применением окна, чем для сигнала без применения сглаживающего окна.

## 7.2. Сигналы с окнами

Окна сглаживания используются, чтобы улучшить спектральные характеристики дискретного сигнала. При выполнении преобразования Фурье или при спектральном анализе в случае использования исходных данных конечной длины по времени можно использовать окна сглаживания, чтобы уменьшить нарушения непрерывности колебаний обрезанной формы и таким образом добиться снижения спектральной утечки. Величина спектральной утечки зависит от амплитуды неоднородности. Если неоднородность становится большой, увеличивается растекание спектра и наоборот. Сглаживающие окна уменьшают амплитуду непрерывности на границах каждого периода и действуют подобно пропусканию сигнала через предварительный ФНЧ с узкой полосой пропускания.

В процессе работы с окнами происходит перемножение отчетов сигнала с отсчетами окна сглаживания конечной длины, значения которых изменяются гладко и постепенно снижаются до нуля на границах. Длина, или интервал времени, окна сглаживания определена числом единичных выборок. В соответствии со свойствами преобразования Фурье умножение в области времени эквивалентно свертке в области частоты. Поэтому спектр сигнала с окном есть свертка спектра исходного сигнала со спектром окна сглаживания. Работа с окнами изменяет форму сигнала в области времени, в результате изменяется спектр этого сигнала в соответствии со свойством преобразования Фурье.

На рис. 7.5 представлена свертка спектра исходного сигнала со спектром окна сглаживания.

Даже если специально не применяется окно сглаживания к сигналу, эффект воздействия окна все же имеет место. Так, получение конечного числа отчетов входного сигнала эквивалентно эффекту умножения сигнала в области времени на прямоугольное окно. Это окно имеет форму прямоугольника и одинаковую для всех отсчетов выборки высоту. Умножение входного сигнала в области времени на прямоугольное окно эквивалентно свертке спектра сигнала со спектральной плотностью прямоугольного окна, которая имеет характеристику, подобную функции sinc.

На рис. 7.6 показан результат применения окна Хэмминга к сигналу в области времени.

На рис. 7.6 форма колебания сигнала во времени в соответствии с формой окна постепенно сужается, чтобы обнулить отсчеты в начале и конце выборки, поэтому окно Хемминга сглаживает неоднородности в форме колебания на границах перехода. Применение окна сглаживания к данным в области времени обеспечивает минимизацию спектральной утечки перед преобразованием этих данных из области времени в область частот.

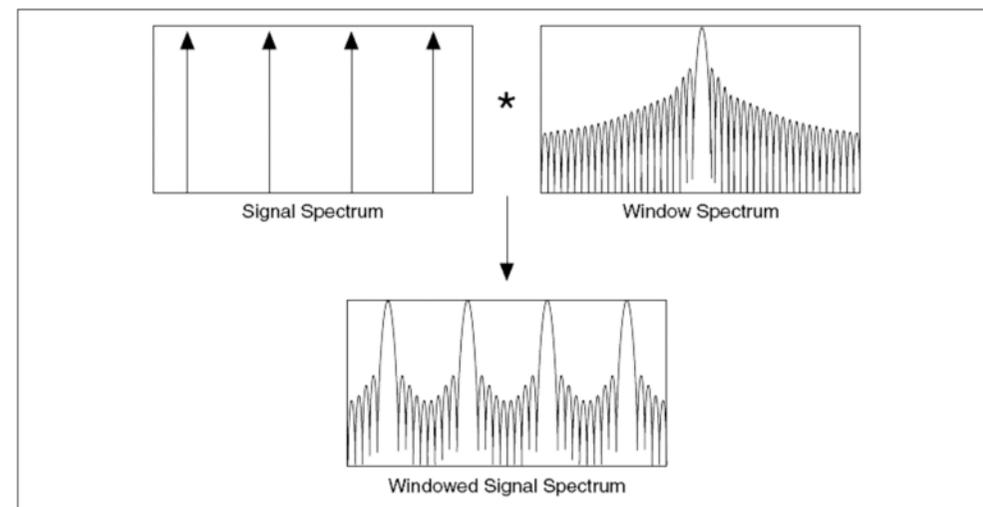


Рис. 7.5. Спектральные характеристики сигнала с окном сглаживания

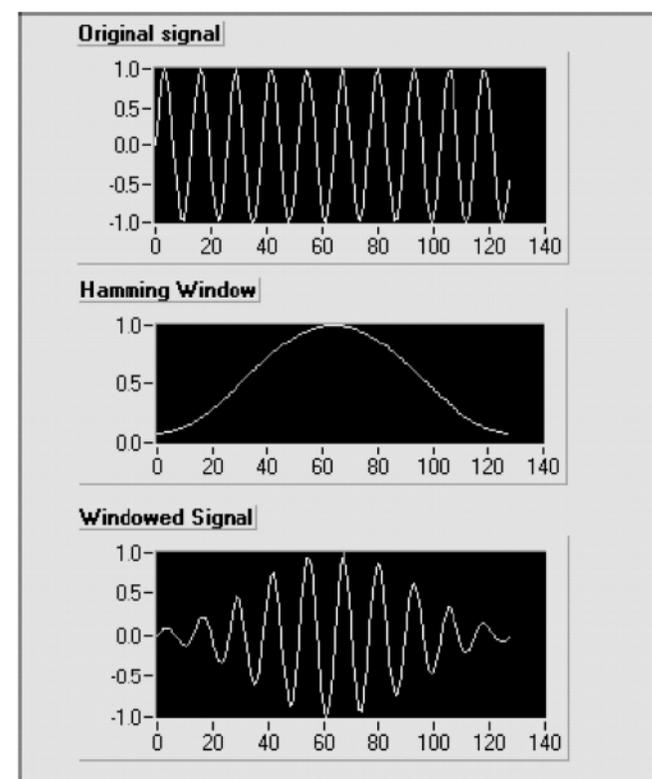


Рис. 7.6. Сигнал с окном при использовании окна Хемминга

На рис. 7.7 показаны эффекты от применения окна сглаживания Хэннинга (Hanning) и окна с плоской вершиной (Flat top), а также эффект без применения окна (униформа) (Uniform).

Набор данных для сигнала, представленного на рис. 7.7, состоит из 256 целых периодов, представленных в виде 1024 отчетов. Если составляющие спектра исходного сигнала соответствуют точно линии частоты, что имеет место, когда получается целое число периодов, на графике виден только основной лепесток спектра. Окна сглаживания имеют основной лепесток вокруг частоты, представляющей интерес. Основной лепесток (если анализируется гармонический сигнал) соответствует характеристике окна в области частоты. Прямоугольное (Uniform) окно имеет самый узкий лепесток. Окно Хэннинга и окно с плоской вершиной приводят к некоторому размытию основного (главного) лепестка. Окно с плоской вершиной имеет более широкий основной лепесток, чем окно Хэннинга или прямоугольное окно (Uniform). Для целого числа периодов все окна сглаживания приводят к одному и тому же максимальному значению амплитуды и обеспечивают ее превосходную точность. Побочные лепестки не появляются, потому что спектр окна сглаживания приближается к нулю в интервалах  $\Delta f$  с обеих сторон основного лепестка.

На рис. 7.7 в таблице показаны значения амплитуд в точках, соответствующих частотам 254, 255, 256, 257 и 258 Гц для каждого окна сглаживания. Ошибка

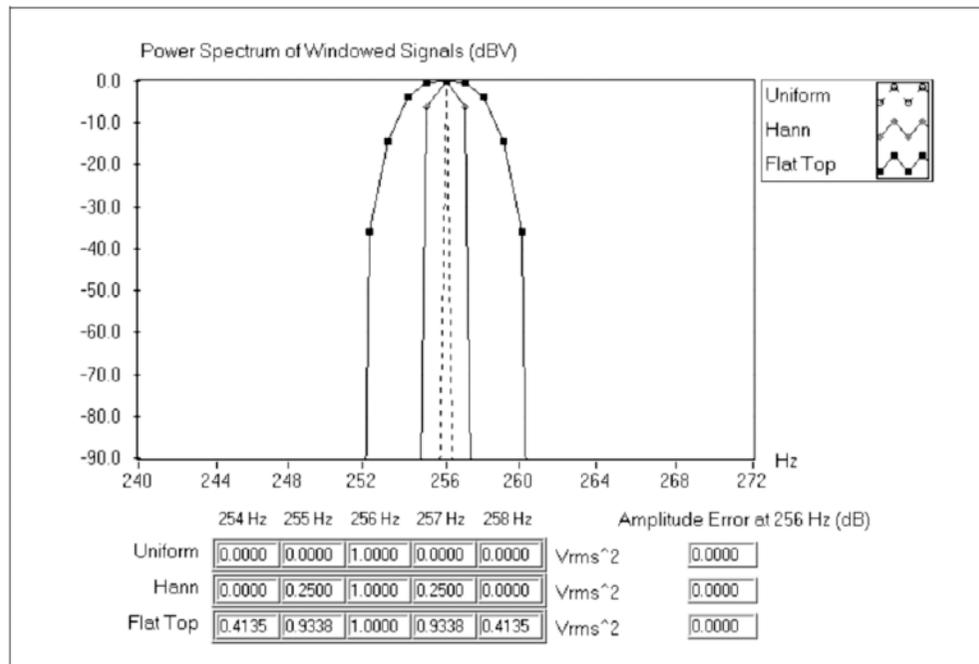


Рис. 7.7. Спектр мощности сигнала  $1V_{rms}$  с частотой 256 Гц с окном Хэннинга (Hanning), окном с плоской вершиной (Flat Top) и без окна (Uniform)

в амплитуде составляющей с частотой 256 Гц равняется 0 дБ для каждого окна сглаживания. График показывает значения спектра между 240 и 272 Гц. Фактические значения в результирующем массиве спектра для каждого окна сглаживания в 254 Гц и через 258 Гц показаны ниже графика. Интервал  $\Delta f$  равен 1 Гц.

Если отсчеты сигнала по времени не содержат целого числа периодов в пределах выборки, непрерывный спектр окна сглаживания смещает центр основного лепестка на интервал  $\Delta f$ , который соответствует различию между составляющими спектра и частотами отсчетов БПФ. Этот сдвиг приводит к появлению побочных лепестков в спектре. Кроме того, ошибка в амплитуде проявляется в пиковом значении на основной частоте, потому что сглаживается единичная выборка в основном лепестке (от центра) и распределяется («размывается») спектр по соседним частотам. На рис. 7.8 показан эффект от спектральной утечки на примере сигнала, выборка которого состоит из 256,5 периодов.

На рис. 7.8 для нецелого числа периодов использование окон Hanning и Flat top приводит к намного меньшей спектральной утечке, чем применение прямоугольного (Uniform) окна.

Кроме того, ошибка в амплитуде меньше для окон Хэннинга и с плоской вершиной.

Окно с плоской вершиной (Flat top) по сравнению с окном Хэннинга обеспечивает большую точность амплитуды, но приводит к появлению более высоких боковых лепестков и к большей «расплюснутости» основного лепестка.

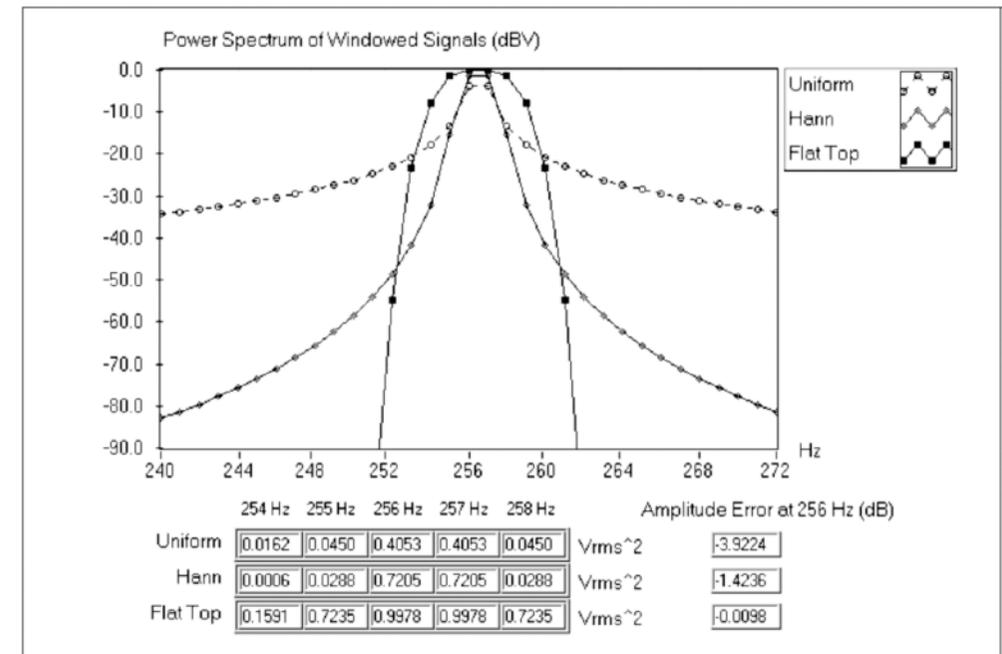


Рис. 7.8. Спектр мощности сигнала  $1V_{rms}$  с частотой 256,5 Гц с окнами: прямоугольным, Хэннинга и с плоской вершиной

На рис. 7.9 показана блок-диаграмма ВП, который измеряет спектр сигнала, составленного из суммы двух синусоид, с окном и без окна.

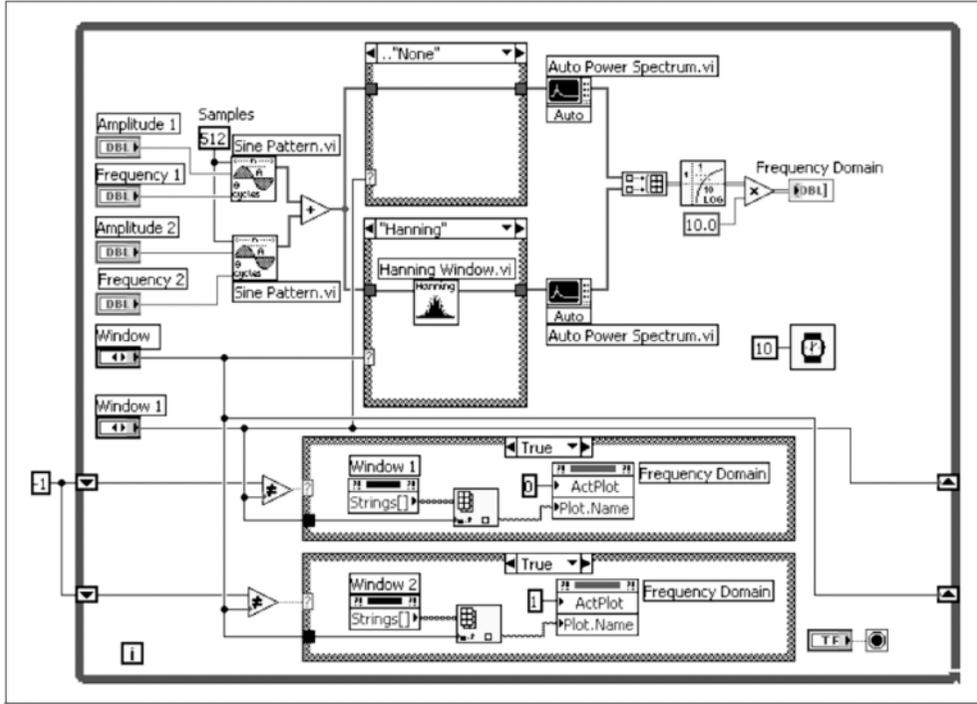


Рис. 7.9. Блок-диаграмма ВП для измерения спектра сигнала, составленного из суммы двух синусоид

На рис. 7.10 показаны амплитуды и частоты этих двух синусоид и результаты их измерения. Частоты составляющих определяются значениями их периодов.

На рис. 7.10 спектр сигнала без окна показывает утечку, которая превышает 20 дБ на частоте синусоиды с меньшей частотой.

Можно применить более сложные методы, чтобы получить более точное описание исходного непрерывного сигнала в области частоты.

Однако в большинстве случаев применения окно сглаживания достаточно, чтобы получить нужное представление о частотных свойствах сигнала.

### 7.3. Характеристики сигналов с различными окнами сглаживания

Чтобы упростить выбор окна сглаживания, необходимо определить различные характеристики так, чтобы можно было сравнивать окна сглаживания. Фактический график окна сглаживания показывает, что частотная характеристика окна

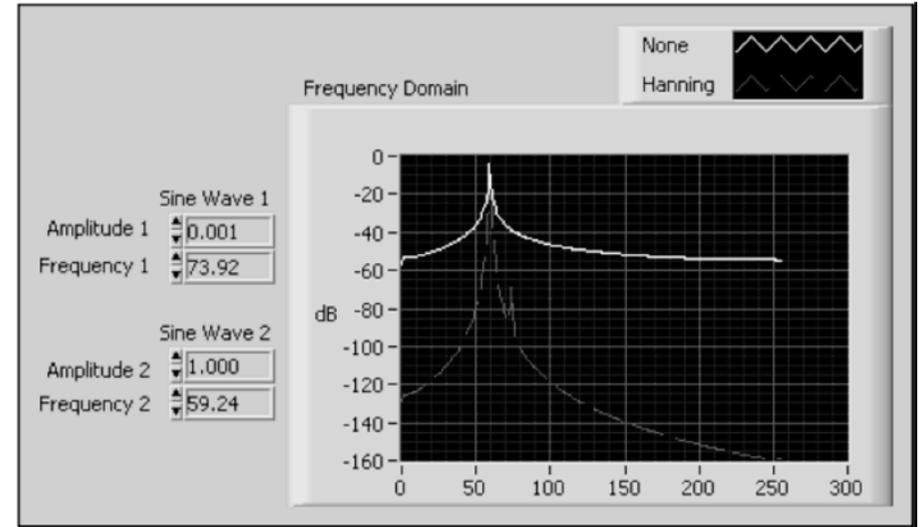


Рис. 7.10. Спектры суммы двух синусоид с окнами и без них

есть непрерывный спектр с основным лепестком и несколькими боковыми лепестками. На рис. 7.11 показан типичный спектр окна сглаживания.

#### 7.3.1. Основной лепесток спектра окна сглаживания

Центр основного лепестка спектра окна сглаживания находится в каждой спектральной составляющей сигнала на оси частот. В соответствии с общепринятыми правилами ширина основного лепестка определяется на уровнях  $-3$  дБ и  $-6$  дБ

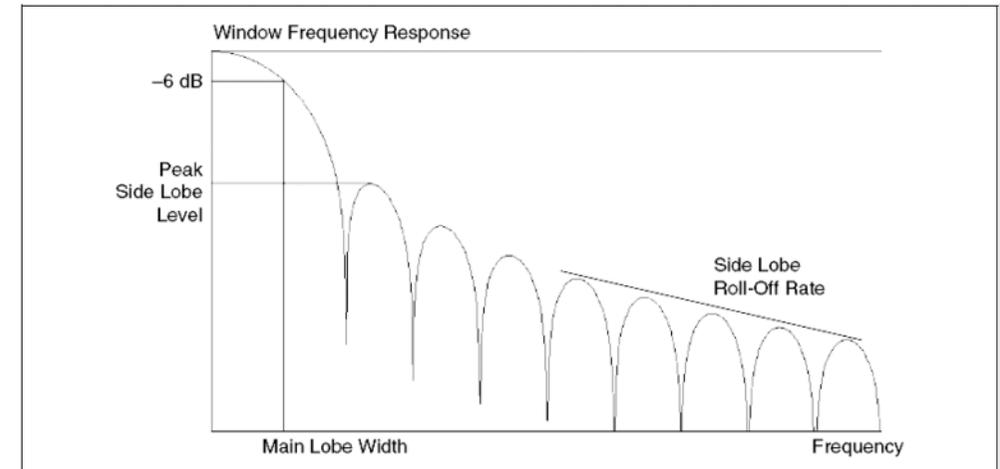


Рис. 7.11. Обобщенный спектр окон сглаживания

ниже максимума основного пика лепестка. Единица измерения для ширины основного лепестка спектра БПФ – точки или частотные линии.

Ширина основного лепестка спектра окна сглаживания ограничивает разрешающую способность частоты сигнала с окном. Поэтому способность БПФ отличать две близко расположенные спектральные составляющие увеличивается, так как основной лепесток окна сглаживания сужается. Поскольку основной лепесток сужается и разрешающая способность по спектру улучшается за счет распространения энергии окна в его боковые лепестки, увеличивается спектральная утечка и уменьшается точность в оценке амплитуды. Обеспечивается компромисс между точностью оценки амплитуды и разрешающей способностью по спектру.

### 7.3.2. Боковые лепестки спектра окна сглаживания

Боковые лепестки спектра окна сглаживания располагаются с каждой стороны основного лепестка и приближаются к нулю при подходе к частоте с множителем  $f_s/N$  со стороны основного лепестка. Характеристики боковых лепестков окна сглаживания непосредственно влияют на степень «просачивания» энергии смежных компонент спектра входного сигнала в смежные линии частоты после применения ДПФ. Боковой лепесток спектра сильного синусоидального сигнала может «пересилить» основной лепесток спектра близлежащего слабого синусоидального сигнала.

Максимальный уровень первого бокового лепестка и боковой склон (roll-off) лепестков с обеих сторон от основного лепестка характеризуют параметры боковых лепестков спектра окна сглаживания. Максимальный уровень первого бокового лепестка – наибольший уровень боковых лепестков в децибелах относительно пика основного лепестка. Боковой склон лепестков есть асимптотический спад уровня лепестков (в дБ/декада) по частоте. В табл. 7.1 приводятся характеристики нескольких основных окон сглаживания.

Таблица 7.1. Характеристики окон сглаживания

Сглаживающее окно	Уровень –3дБ основного лепестка (строки БПФ)	Уровень –6дБ основного лепестка (строки БПФ)	Максимальный уровень боковых лепестков (дБ)	Склон боковых лепестков (дБ/декада)
Прямоугольное (Uniform)	0,88	1,21	-13	20
Хэннинга	1,44	2,00	-32	60
Хэмминга	1,30	1,81	-43	20
Блэкмана – Харриса	1,62	2,27	-71	20
Точное				
Блэкмана	1,61	2,25	-67	20
Блэкмана	1,64	2,30	-58	60
С плоской вершиной (Flat Top)	2,94	3,56	-44	20

## 7.4. Прямоугольное окно (None)

Прямоугольное окно имеет значение, равное единице по всей его длине. Следующее уравнение определяет прямоугольное окно:

$$w(n) = 1,0 \text{ для } n = 0, 1, 2, \dots, N - 1,$$

где  $N$  – длина окна;  $w$  – значение окна.

Применение прямоугольного окна эквивалентно неиспользованию любого окна, потому что прямоугольная функция только выделяет сигнал в пределах конечного интервала времени (времени выборки). Прямоугольное окно имеет самый высокий уровень спектральной утечки.

На рис. 7.12 показано прямоугольное окно для  $N = 32$ .

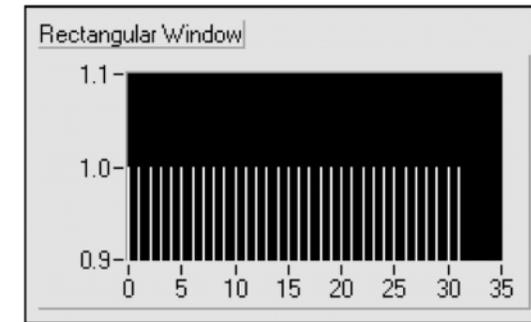


Рис. 7.12. Прямоугольное окно

Прямоугольное окно полезно в том случае, когда необходимо анализировать переходные процессы, имеющие продолжительность короче, чем это окно. Переходные процессы – сигналы, которые существуют только в течение сравнительно короткого отрезка времени. Прямоугольное окно также используется, чтобы измерить трэкинг (tracking), при этом эффективная частота дискретизации выбирается пропорциональной скорости вращающегося вала в машине. При измерении трэкинга прямоугольное окно обнаруживает основной режим вибрации машины и ее гармоники.

## 7.5. Окно Хэннинга (Hanning)

Окно Хэннинга имеет форму, подобную половине периода косинусоидального колебания. Следующее уравнение определяет окно Хэннинга:

$$w(n) = 0,5 - 0,5 \cos \frac{2\pi n}{N} \text{ для } n = 0, 1, 2, \dots, N - 1,$$

где  $N$  – длина окна;  $w$  – значение окна.

На рис. 7.13 представлено окно Хэннинга при  $N = 32$ .

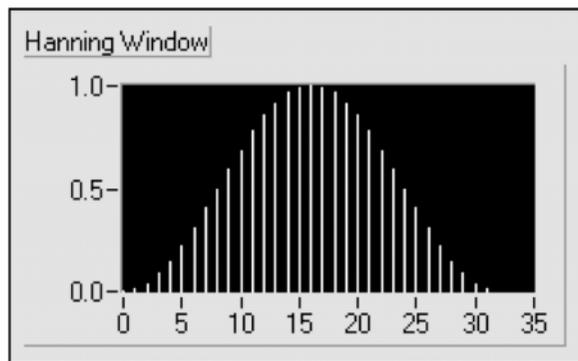


Рис. 7.13. Окно Хэннинга

Окно Хэннинга используют в том случае, когда необходимо анализировать переходные процессы, протекающие дольше, чем продолжительность времени окна, и также для универсальных приложений.

## 7.6. Окно Хемминга (Hamming)

Окно Хэмминга – управляемая версия окна Хэннинга.

Форма окна Хэмминга также подобна отрезку косинусоидального колебания.

Следующее уравнение определяет окно Хэмминга:

$$w(n) = 0,54 - 0,46 \cos \frac{2\pi n}{N} \text{ для } n = 0, 1, 2, \dots, N - 1,$$

где  $N$  – длина окна;  $w$  – значение окна.

На рис. 7.14 показано окно Хэмминга при  $N = 32$ .

Окна Хэннинга и Хэмминга подобны друг другу, как показывает сравнение рис. 7.13 и 7.14. Однако в области времени окна Хэмминга не приближается так близко к нулю около граней выборки, как это делается в окне Хэннинга.

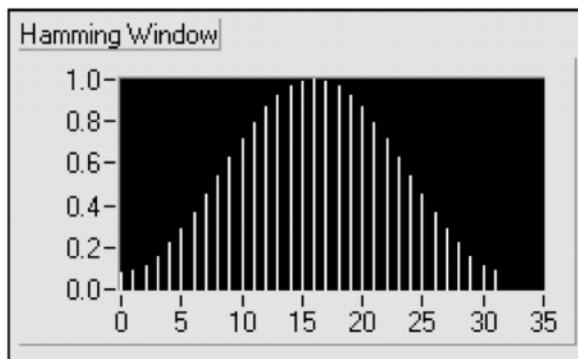


Рис. 7.14. Окно Хемминга

## 7.7. Окно Кайзера – Бесселя (Kaiser – Bessel)

Окно Кайзера – Бесселя – гибкое окно сглаживания, форму которого можно изменить, регулируя ввод параметра **бета** (beta). Таким образом, в зависимости от применения можно изменять форму окна для управления величиной спектральной утечки.

На рис. 7.15 показано окно Кайзера – Бесселя для различных значений параметра **бета**.

При малых значениях параметра **бета** форма окна Кайзера – Бесселя близка к форме прямоугольного окна.

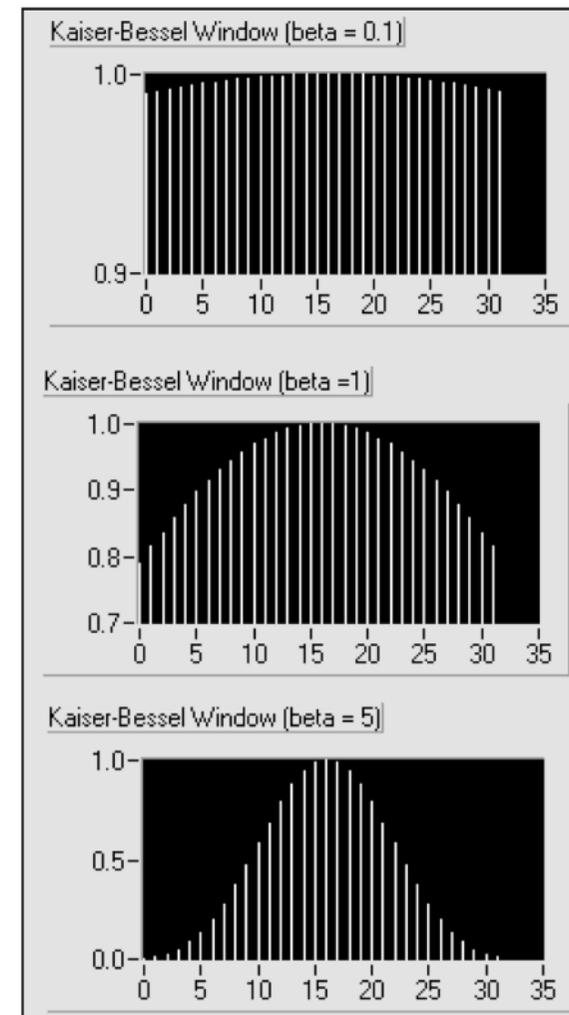


Рис. 7.15. Окно Кайзера – Бесселя (Kaiser – Bessel)

Для параметра **бета**, равного 0, действительно получается прямоугольное окно. В то же время при увеличении этого параметра амплитуды отсчетов окна снижаются до крайних значений окна.

Окно Кайзера – Бесселя применяется для того, чтобы обнаружить два сигнала почти одной и той же частоты, но со значительно различающимися амплитудами.

## 7.8. Треугольное окно (Triangle)

Форма треугольного окна ясна из названия. Следующее уравнение определяет треугольное окно:

$$w(n) = 1 - \left| \frac{2n - N}{N} \right| \text{ для } n = 0, 1, 2, \dots, N - 1,$$

где  $N$  – длина окна;  $w$  – значение окна.

На рис. 7.16 представлено треугольное окно при  $N = 32$ .

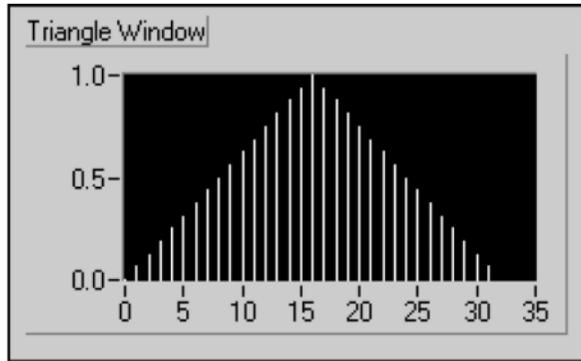


Рис. 7.16. Треугольное окно

## 7.9. Окно с плоской вершиной (Flat Top)

Окно с плоской вершиной по сравнению с другими окнами обеспечивает самую высокую точность оценки амплитуды спектральной составляющей, равную  $\pm 0,02$  дБ, для выборок сигналов, соответствующих целому числу периодов. Однако, поскольку окно с плоской вершиной имеет широкий основной лепесток в спектре, это приводит к низкой разрешающей способности спектрального анализа по частоте.

Следующее уравнение определяет окно с плоской вершиной:

$$w(n) = \sum_{k=0}^4 (-1)^k / a_k \cos(k\omega),$$

где  $\omega = \frac{2\pi n}{N}$ ;  $a_0 = 0,215578948$ ;  $a_1 = 0,416631580$ ;  $a_2 = 0,277263158$ ;  $a_3 = 0,083578947$ ;  $a_4 = 0,006947368$ .

На рис. 7.17 показано окно с плоской вершиной (Flat Top)

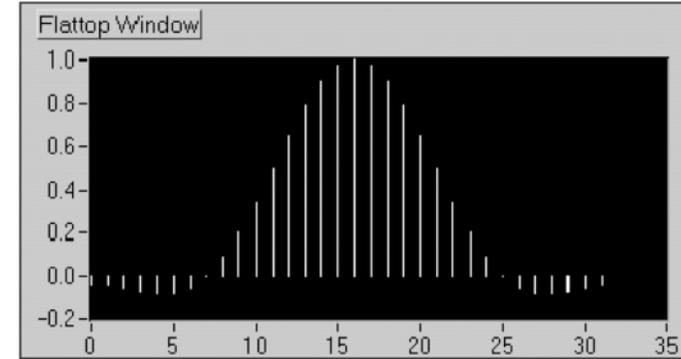


Рис. 7.17. Окно с плоской вершиной (Flat Top)

Окно с плоской вершиной (Flat Top) наиболее применимо при точном измерении амплитуды отдельных составляющих спектра при малом уровне амплитуд соседних составляющих в спектре анализируемого сигнала.

## 7.10. Экспоненциальное (Exponential) окно

Форма экспоненциального окна – это затухающая экспонента.

Следующее уравнение определяет экспоненциальное окно:

$$w(n) = e^{\left(\frac{n \ln(f)}{N-1}\right)} = f^{\left(\frac{n}{N-1}\right)} \text{ для } n = 0, 1, 2, \dots, N - 1,$$

где  $N$  – длина окна;  $w$  – значения окна;  $f$  – конечное значение частоты.

Начальное значение окна равно единице и постепенно уменьшается к нулю.

Можно корректировать конечное значение единичного отсчета экспоненциального окна между 0 и 1.

На рис. 7.18 показано экспоненциальное окно при  $N = 32$  с конечным значением 0,1.

Экспоненциальное окно применяют в тех случаях, когда необходимо анализировать переходные сигналы отклика, продолжительность которых больше, чем длина окна. Экспоненциальное окно «заглушает» конец сигнала, гарантируя, что мгновенные значения сигнала уменьшаются к концу выборки по времени. Можно применять экспоненциальное окно к сигналам, у которых спад происходит по экспоненте типа отклика структур, подобного удару с затуханием. Этот отклик обуславливается внешним кратковременным воздействием.

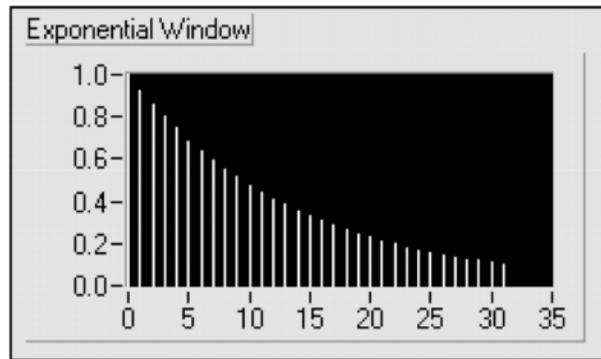


Рис. 7.18. Экспоненциальное окно

## 7.11. Окно для спектрального анализа против окна для проектирования коэффициентов фильтра

Спектральный анализ и проектирование коэффициентов фильтра предъявляет различные требования к окну. Спектральный анализ требует окна с четным спектром ДПФ, в то время как проектирование коэффициентов фильтра требует окна, симметричного относительно его середины.

### 7.11.1. Спектральный анализ

Окна сглаживания, предназначенные для спектрального анализа, должны иметь четный спектр после применения ДПФ.

Окно сглаживания имеет четное ДПФ, результат которого есть скалярные числа с нулевыми синусоидальными составляющими. Другими словами, результат ДПФ не имеет мнимых компонент в своей последовательности.

На рис. 7.19 и 7.20 представлено окно Хэннинга для типового размера выборки, равного 8, и выборка для одного периода синусоидального колебания.

На рис. 7.19 показано четное ДПФ окна Хэннинга, несимметричное относительно его середины. Последний отсчет окна не равен его первому отсчету и подобен одному законченному периоду модели синусоидального колебания, показанного на рис. 7.20.

Сглаживающие окна для спектрального анализа называются спектральными и включают следующие типы окон:

- масштабируемое окно области времени;
- окно Хэннинга (Hanning);
- окно Хэмминга (Hamming);
- окно треугольное (Triangle);

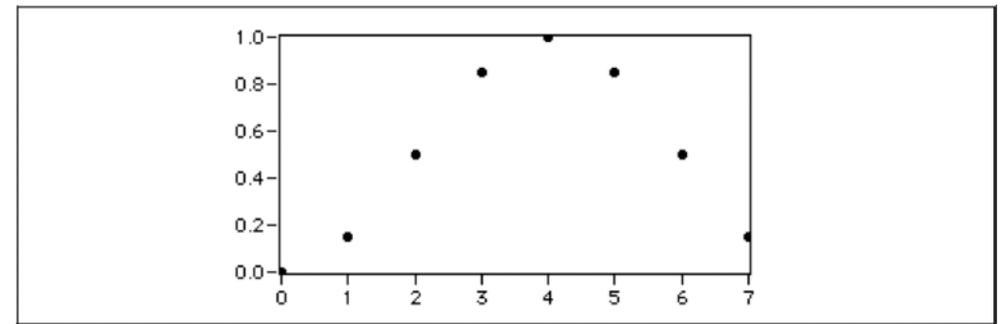


Рис. 7.19. Окно Хэннинга для выборки размера 8

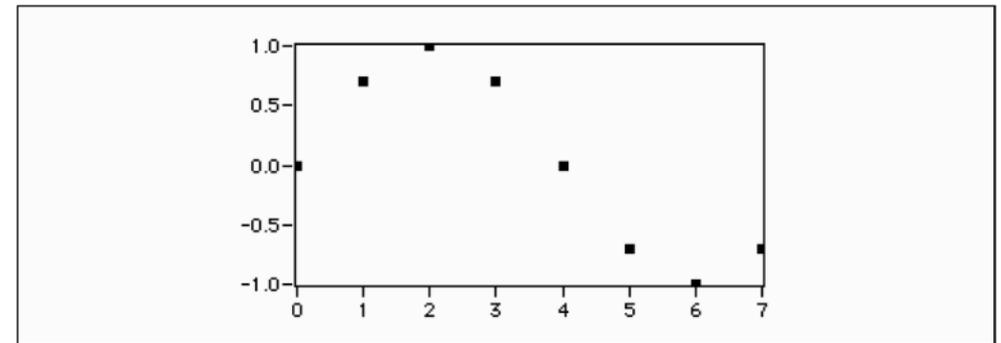


Рис. 7.20. Выборка размера 8 одного периода синусоидального колебания

- окно Блэкмена (Blackman);
- окно точное Блэкмена (Exact Blackman);
- окно Блэкмена – Хэрриса (Blackman – Harris);
- окно с плоской вершиной (Flat Top);
- окно Кайзера – Бесселя (Kaiser – Bessel);
- общее окно косинуса;
- суженное окно косинуса.

### 7.11.2. Окна для проектирования коэффициентов КИХ-фильтра

Проектирование коэффициентов КИХ-фильтра требует использования окна, которое является симметричным относительно его середины.

Уравнения, приводимые ниже, иллюстрируют различие между спектральным окном и симметричным окном для проектирования коэффициентов фильтра.

Следующее уравнение определяет окно Хэннинга для спектрального анализа:

$$w[i] = 0,5 \left( 1 - \cos \left( \frac{2\pi i}{N} \right) \right) \text{ для } i = 0, 1, 2, \dots, N-1, \quad (7.1)$$

где  $N$  – длина окна;  $w$  – значения окна.

Следующее уравнение определяет симметричное окно Хэннинга для проектирования коэффициентов фильтра:

$$w[i] = 0,5 \left( 1 - \cos \left( \frac{2\pi i}{N-1} \right) \right) \text{ для } i = 0, 1, 2, \dots, N-1, \quad (7.2)$$

где  $N$  – длина окна;  $w$  – значения окна.

Изменяя спектральное окно, как показано в уравнении (7.2), можно определить симметричное окно для того, чтобы рассчитать коэффициенты фильтра. Можно обратиться к разд. 4 для получения дополнительной информации о проектировании цифровых фильтров.

## 7.12. Выбор правильного окна сглаживания

Выбор окна сглаживания – не простая задача. Каждое окно сглаживания имеет свои собственные, отличные от других характеристики и степень пригодности для различных применений. Чтобы выбрать окно сглаживания, необходимо оценить содержание и величину амплитуд спектральных составляющих сигнала. Если же сигнал содержит большие по амплитуде побочные компоненты спектральных составляющих с частотами, сильно отличающимися от составляющей, представляющей интерес, необходимо выбирать окно сглаживания с крутым склоном боковых лепестков относительно главного лепестка. Если сигнал содержит побочные спектральные компоненты с большими амплитудами и с частотами, близкими по частоте к составляющей, представляющей интерес, необходимо выбирать окно сглаживания с низким максимальным уровнем боковых лепестков. Можно обратиться к табл. 7.1 для получения информации о склоне боковых лепестков относительно главного лепестка и максимальном уровне боковых лепестков для различных окон сглаживания, применяемых в цифровой обработке сигналов.

Если компонента сигнала, представляющая интерес, содержит две или более спектральные составляющие, очень близко расположенные относительно друг друга по частоте, важна спектральная разрешающая способность. В этом случае лучше выбирать окно сглаживания с очень узким основным лепестком. Если же точность оценки амплитуды отдельной спектральной составляющей более важна, чем точное определение местоположения компоненты в данном месте по оси частот, надо выбирать окно сглаживания с широким основным лепестком. Если спектр сигнала является довольно плоским или широкополосным по частоте, необходимо использовать прямоугольное окно или не использовать никакого окна. Вообще применение окна Хэннинга удовлетворительно в 95 % практических случаев. Оно имеет хорошую разрешающую способность по частоте и существенно уменьшает спектральную утечку. В том случае, если неизвестен характер сигнала, но необходимо применить окно сглаживания, надо начать с окна Хэннинга.

В табл. 7.2 приведен список различных типов сигналов и соответствующих окон, которые можно использовать для их спектрального анализа.

Таблица 7.2. Виды сигналов и рекомендуемые окна для их анализа

Тип сигнала	Окно
Переходные процессы, продолжительность которых короче, чем длина окна	Rectangular
Переходные процессы, продолжительность которых длиннее, чем длина окна	Exponential, Hanning
Универсальные приложения	Hanning
Спектральный анализ (измерения частотной характеристики)	Хэннинга (для случайного возбуждения), прямоугольное (для псевдослучайного возбуждения)
Сочетание двух тонов с очень близкими частотами, но сильно отличающимися амплитудами	Kaiser – Bessel
Сочетание двух тонов с очень близкими частотами и с почти равными амплитудами	Rectangular
Измерения амплитуды единственного тона	Flat Top
Синусоидальные колебания или комбинация таких колебаний	Хэннинг
Синусоидальные колебания (при этом важна точность оценки амплитуды)	Flat Top
Узкополосный случайный сигнал (сигнал вибрации)	Хэннинг
Широкополосный случайный сигнал (белый шум)	Uniform
Близко расположенные отрезки синусоидального колебания	Униформа, Хэмминг
Возбуждающий сигнал («сокрушительный удар»)	Force
Сигнал отклика	Exponential
Сигнал неизвестного содержания	Хэннинг

Для правильного выбора окна сглаживания необходимо иметь достаточный уровень априорной информации о спектральном составе сигнала. При наличии априорной неопределенности параметров спектра приходится экспериментировать с различными окнами сглаживания для поиска приемлемого окна для решаемой задачи. Для этого необходимо иметь возможность сравнивать различные окна по их параметрам и характеристикам.

## 7.13. Масштабирование окон сглаживания

Применение окна сглаживания к сигналу в области времени заключается в умножении отсчетов сигнала в области времени в пределах окна сглаживания на отсчеты выбранного окна, и вводится искажение в сигнал, возникающее из-за окна сглаживания. В частности, окно сглаживания изменяет полную амплитуду сигнала. При использовании нескольких окон сглаживания для одного и того же сигнала необходимо обеспечить масштабирование каждого окна сглаживания за счет деления массива отсчетов сигнала с окном посредством синхронного усиления

значений отсчетов окна некоторым амплитудным спектром внутри конкретного ограниченного во времени окна. Графики на рис. 7.7 и 7.8 показывают результат применения масштабируемых окон сглаживания к сигналу в области времени.

БПФ эквивалентно набору параллельных фильтров с одинаковой полосой пропускания, равной  $\Delta f$ . Из-за расширяющего эффекта окна сглаживания увеличивается эффективная полоса пропускания фильтров в наборе для спектрального анализа на величину, известную как эквивалентная полоса пропускания шумовой мощности (ENBW – Equivalent Noise-power BandWidth) окна сглаживания.

Мощность каждой спектральной составляющей равняется сумме мощностей смежных отсчетов в спектре вокруг этой составляющей, умноженной на коэффициент масштабирования, она определяет ENBW окна сглаживания. Для уменьшения погрешности анализа необходимо принять во внимание коэффициент масштабирования при выполнении вычислений, основанных на использовании спектра мощности. Можно обратиться к разд. 5 для получения информации о выполнении операций над спектром мощности.

В табл. 7.3 приведен список коэффициентов масштабирования (Scaling Factor или Coherent Gain), также известных как синхронное усиление, значение ENBW и погрешности оценки амплитуды спектральной составляющей для самого худшего случая (Worst-Case Amplitude Error), обусловленного смещением этой составляющей по частоте, для нескольких популярных окон сглаживания.

Таблица 7.3. Корректирующие коэффициенты и максимальные ошибки в оценке амплитуды (Worst-Case Amplitude Error) для окон сглаживания

Окно	Scaling Factor (Coherent Gain)	ENBW	Worst-Case Amplitude Error (дБ)
Прямоугольное (Uniform)	1,00	1,00	3,92
Хэннинга	0,50	1,50	1,42
Хэмминга	0,54	1,36	1,75
Блэкмана – Хэрриса	0,42	1,71	1,13
Точное Блэкмана	0,43	1,69	1,15
Блэкмана	0,42	1,73	1,10
С плоской вершиной (Flat Top)	0,22	3,77	<0,01

## Многоскоростная обработка сигналов в LabVIEW



Часто в области цифровых систем связи сталкиваются с проблемой дискретизации цифровых данных на какой-то частоте, которую необходимо заменить на другую частоту. Возможно, это необходимо из-за ограниченности аппаратных средств ЭВМ или некоторых требований к обработке, выполнение которых вынуждает производить дискретизацию данных на некоторой частоте, но затем возникает необходимость работать с другой частотой. Это случай многоскоростной обработки сигнала. Используя многоскоростную обработку сигналов, можно увеличить частоту дискретизации (Upsample), снизить ее (Downsample) или же обеспечить комбинацию того и другого.

Как правило, повышение или понижение частоты дискретизации производят в целое число раз относительно стартовой частоты дискретизации. Комбинируя эти две операции, можно достичь любого нецелого числа относительно исходной частоты. Далее будут показаны некоторые довольно простые способы изменения частоты дискретизации сигналов и приведены некоторые очень изящные методы, с помощью которых можно эффективно выполнять операции смены частоты дискретизации.

### 8.1. Повышение частоты дискретизации

Наиболее прямой путь повышения частоты дискретизации данных до более высокой частоты есть метод добавление нулей в сигнал, заключающийся в расположении нулей между каждыми двумя соседними отсчетами, чтобы увеличить число дискретных значений, размещенных на данном отрезке времени. Процесс заполнения выборки сигнала нулями не добавляет никаких спектральных составляющих в исходный сигнал и является хорошим способом увеличения частоты дискретизации данного цифрового сигнала.

Теперь можно построить ВП для повышения частоты дискретизации сигнала. На рис. 8.1 показана блок-диаграмма ВП Upsample.vi. Этот ВП инициализирует

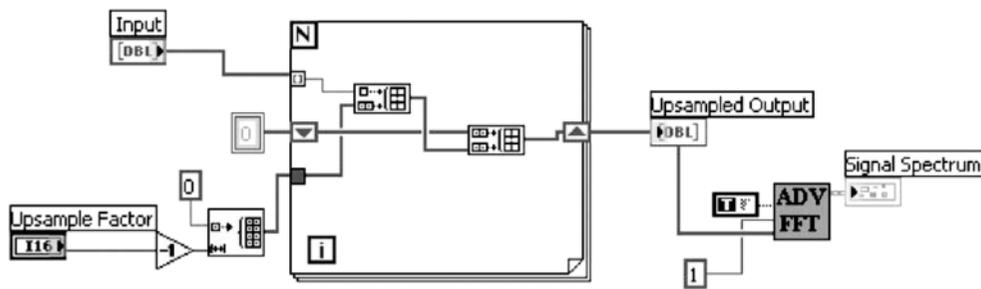


Рис. 8.1. Блок-диаграмма ВП Upsample.vi

множество  $(I - 1)$  нулей и вставляет их между каждыми двумя исходными входными единичными отсчетами.

На графике рис. 8.2 представлен результат БПФ выборки с прореживанием нулями. Если использовать фильтр, представленный в разд. 4 на основе ВП KaiserFIR.vi, то никакая спектральная информация не добавляется в исходный сигнал, но изменяется местоположение частот дискретных спектральных составляющих. При работе с цифровыми сигналами важно помнить, что частота диск-

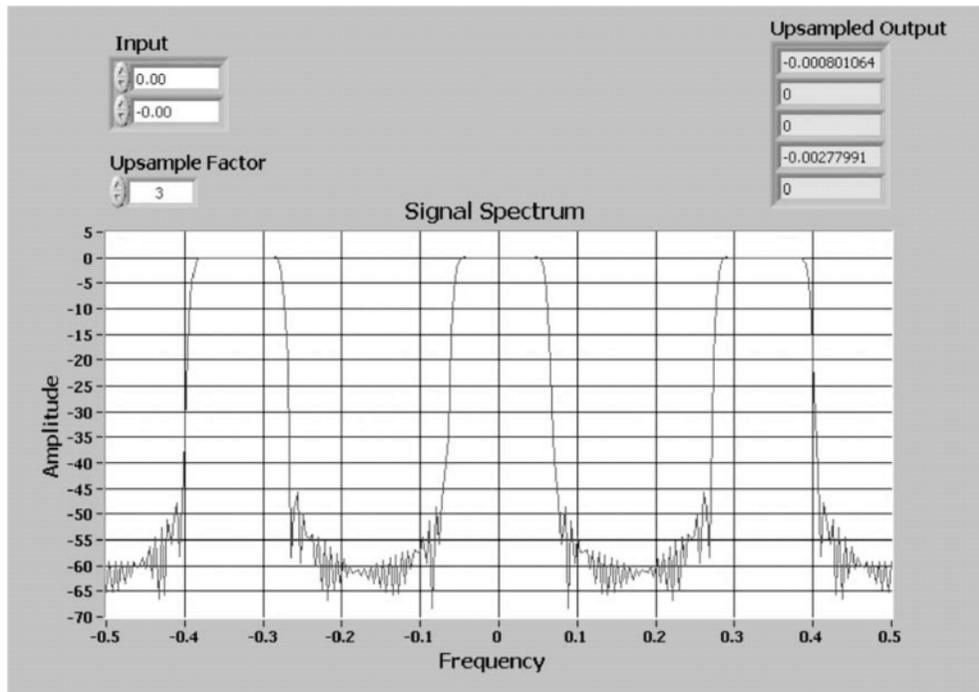


Рис. 8.2. Передняя панель ВП Upsample.vi: три спектральных отсчета при повышении частоты дискретизации в 3 раза

ретных гармоник зависит от частоты дискретизации. Теперь, когда «наполнен» нулями входной сигнал, имеется в три раза больше отсчетов на том же самом отрезке времени. Фактически рассматривается больший отрезок периодического спектра исходного сигнала. Перед повышением частоты дискретизации можно было видеть спектр сигнала в пределах от  $-f_s/2$  до  $f_s/2$ .

Теперь, когда частота  $f_s$  стала больше, в то время как спектральное содержание сигнала осталось тем же самым, можно видеть трехкратное повторение исходного периодического спектра.

Могут иметь место некоторые интересные применения этих методов, когда желательно сохранить спектральные отображения в зависимости от повышения частоты дискретизации, но в большинстве случаев необходимо фильтровать полученный сигнал, чтобы удалить это нежелательное спектральное дублирование. Прежде чем будет показана фильтрация передискретизированных сигналов, желательно рассмотреть обратное действие: уменьшение частоты дискретизации.

## 8.2. Уменьшение частоты дискретизации

Если говорить о повышении частоты дискретизации как о сжатии большего периодического спектра в интервале от  $-f_s/2$  до  $f_s/2$ , то можно понять, что случается, когда уменьшается частота дискретизации. Спектр входного сигнала расширяется, что приводит к изменению масштаба некоторой части исходного спектра и к заполнению спектральными составляющими интервала от  $-f_s/2$  до  $f_s/2$ .

Опасность этой операции состоит в том, что «крупный план» спектра может превысить границу – частоту Найквиста  $f_s/2$ . Превышение приведет к совмещению участков спектра, и тогда будет невозможно вернуть необходимую часть сигнала. Также нужно иметь в виду, что при этом теряется часть исходного сигнала, которая вносит вклад в его полную мощность в исходном спектре. Это означает, что будет потеряна мощность сигнала пропорционально коэффициенту уменьшения частоты дискретизации.

На рис. 8.3 показан спектр сигнала с уменьшенной частотой дискретизации наряду со спектром исходного сигнала.

Можно заметить, что коэффициент уменьшения частоты дискретизации сигнала был равен 2, и, следовательно, полоса, занимаемая сигналом, удвоилась, в то время как его амплитуда уменьшилась наполовину ( $1/2 \sim -6$  дБ). Если бы необходимо было увеличить коэффициент уменьшения частоты дискретизации больше чем в 3 раза, тогда началось бы совмещение частей спектра дискретного сигнала. Это происходит потому, что исходный сигнал содержит всю спектральную информацию в полосе до 2 кГц. Если сигнал расширяется в 3 раза, то составляющие до 2 кГц перемещаются вправо к частоте 6 кГц, и происходит совмещение частей спектра за пределами половины частоты дискретизации, составляющей 5 кГц (из разд. 4 известно, что частота дискретизации для коэффициентов этого фильтра была равна 10 кГц).

Из этого примера видно, как можно избежать совмещения частей спектра при уменьшении частоты дискретизации. Из анализа можно сделать вывод: совмеще-

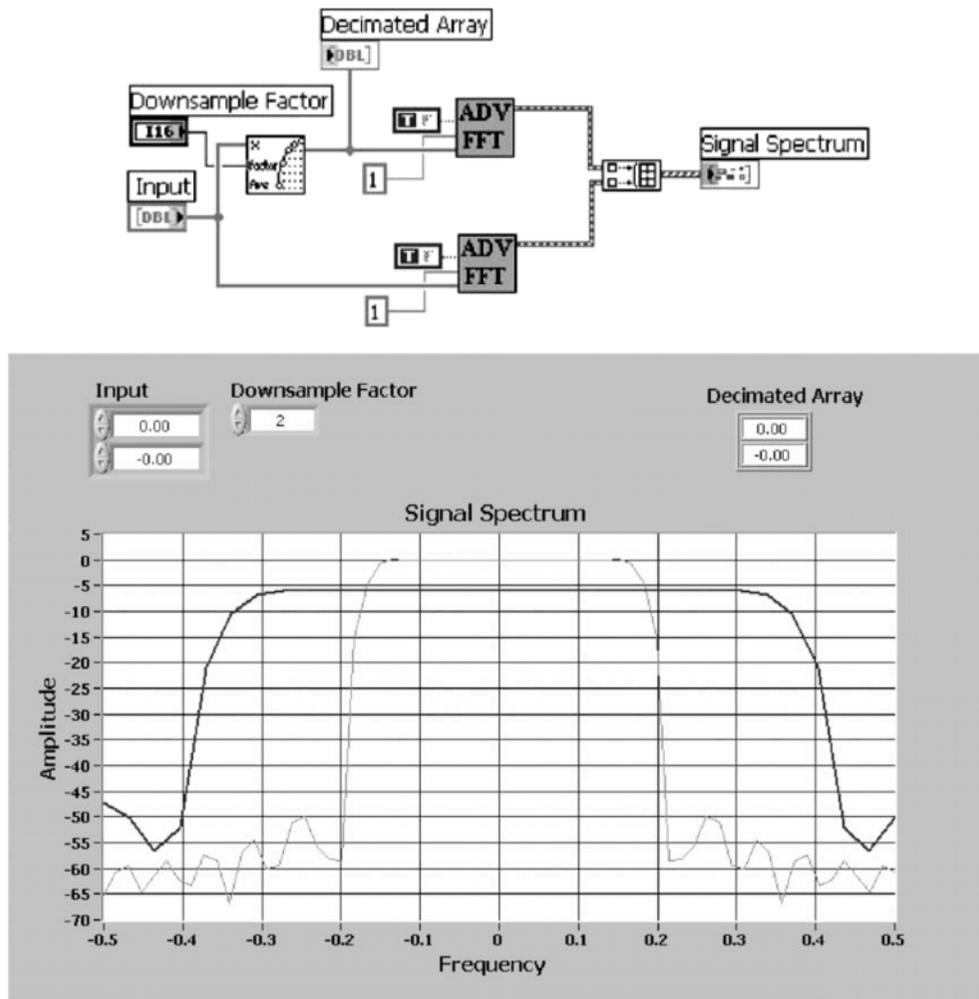


Рис. 8.3. Блок-диаграмма и передняя панель ВП Downsample.vi

ния не будет, пока исходная полоса, занимаемая сигналом, меньше чем  $1/2$  от половины частоты дискретизации, или, более точно, чем  $1/4$  от исходной частоты дискретизации. Но иногда исходный сигнал может содержать шум, который дает существенные спектральные составляющие на частотах больших, чем  $1/4$  частоты дискретизации. Когда уменьшается частота дискретизации зашумленного сигнала, шум может давать ложные составляющие и искажать текущий сигнал. Конечно, это явление не ограничено делением частоты дискретизации на коэффициент, равный 2. Для произвольного коэффициента уменьшения частоты дискретизации  $M$  сигнал должен быть *сначала* ограничен полосой  $f_s / 2M$  [4]. Именно поэтому необходимо пропускать сигнал через фильтр, устраняющий ложные со-

ставляющие перед операцией уменьшения частоты дискретизации. Теперь, когда приведен краткий обзор операций повышения и уменьшения частоты дискретизации, можно закончить анализ операций передискретизации, представляя ее как процесс фильтрации.

### 8.3. Фильтры передискретизации

Поскольку с увеличением и с уменьшением частоты дискретизации изменяется спектр дискретного сигнала, возникает необходимость в некоторой фильтрации, чтобы сохранить спектр исходного сигнала после операции передискретизации. Из предыдущего рассмотрения ясно, что в случае операции уменьшения частоты дискретизации лучше иметь фильтр для сигнала *на входе*, чтобы быть уверенным в том, что не получится никакого совмещения спектра.

Для случая повышения частоты дискретизации нет никакого совмещения частей спектра, но, если применяется фильтр перед операцией повышения частоты дискретизации, дополнительные спектральные составляющие будут все же обнаруживаться неизменными. Следовательно, для операции повышения частоты дискретизации фильтр должен быть применен *позже* передискретизации.

На рис. 8.4 и 8.5 показаны структуры операций уменьшения и повышения частоты дискретизации. В этом традиционном типе конфигурации структуры обработки сигнала фильтр уменьшения частоты дискретизации используется на входной частоте дискретизации, в то время как фильтр повышения частоты дискретизации должен применяться на более высокой выходной частоте дискретизации. Как можно было видеть в подразд. 8.2, при уменьшении частоты дискретизации граничная частота фильтра должна ограничить входную полосу пропускания сигнала  $f_s / 2M$ . Фильтр повышения частоты дискретизации, с другой стороны, должен иметь верхнюю граничную частоту, равную половине исходной частоты дискретизации или половине интерполированной частоты дискретизации  $f_i$ , разделенной на  $2I$ .

До сих пор операции передискретизации изменяли частоту вверх или вниз в целое число раз. При нецелом числе повышение и уменьшение частоты дискре-

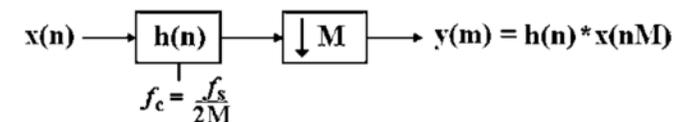


Рис. 8.4. Структура операции уменьшения частоты дискретизации

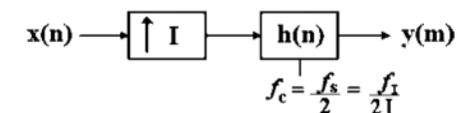


Рис. 8.5. Структура операции повышения частоты дискретизации

тизации может быть объединено для достижения почти произвольного коэффициента. Например, чтобы обеспечить изменение частоты в 2,35 раза, можно увеличить частоту в 12 раз и уменьшить ее в 5 раз. Конечно, если потребовать более высокой точности, то надо увеличить частоту в 235 раз и уменьшить ее в 100 раз, но на это потребуется больший объем вычислений. При объединении этих двух операций соответствующий фильтр может быть выбран так, что он будет и удалять дополнительные спектральные составляющие при повышении частоты, и предотвращать совмещение частей спектра при ее уменьшении. Так как повышение частоты требует фильтрации позже дискретизации, а снижение частоты требует фильтрации до дискретизации, повышение частоты должно всегда выполняться сначала. Эта объединенная операция показана на рис. 8.6.

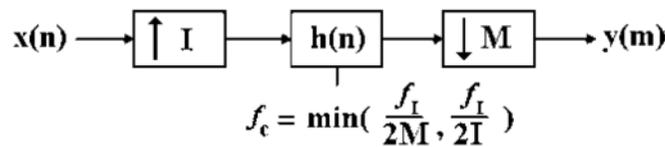


Рис. 8.6. Рациональное соотношение частот при передискретизации

Объединение операций повышения и уменьшения частоты (рис. 8.6), приведет к очевидной экономии в вычислениях. Раздельное выполнение этих двух операций потребовало бы фильтра для каждой операции. Объединение операций позволяет использовать единственный фильтр, чтобы и удалить дополнительные спектральные составляющие при повышении частоты, и предотвратить совмещение частей спектра дискретного сигнала из-за последующего ее уменьшения. Для того чтобы этот фильтр удалял безошибочно часть спектра, частота среза должна быть выбрана как меньшая из двух частот:  $f_t / 2M$  или  $f_t / 2I$ . Для того чтобы при использовании единственного фильтра можно было сэкономить время вычисления в большей степени, есть некоторые специальные условия, касающиеся передискретизации последовательности, которые можно использовать в своих интересах. Для этого необходимо подробнее рассмотреть случай повышения частоты дискретизации.

Из разд. 8.1 следует, что при повышении частоты выполняется вставка  $(I - 1)$  нулей между каждыми двумя единичными отсчетом входной последовательности. Если подставлять нули в фильтре, то это приводит к большим затратам времени, даже если точно известно их расположение в выборке. Можно не производить вычисления на нулевых входных отсчетах, если тщательно перестроить путь потока входной последовательности через фильтр. Фактически можно не вставлять нули во входную последовательность вообще, и все же в этой последовательности может быть повышена частота дискретизации на основе свойств фильтрации.

Такой тип фильтра известен как полифазный фильтр, и его сущность будет рассмотрена в подразд. 8.3.2.

Противоположная ситуация имеет место в случае уменьшения частоты дискретизации, когда сигнал сначала фильтруется, и тогда исключается каждый

$M$ -й отсчет. Оба случая будут рассмотрены ниже наряду с применением LabVIEW для выполнения полифазной интерполяции.

Есть также другой тип фильтра передискретизации выборки, известный как фильтр полуполосы. Этот фильтр имеет особенность, заключающуюся в том, что каждый второй коэффициент в области времени является нулем, и это может использоваться для сокращения требуемых вычислений в два раза [11].

Теперь можно рассмотреть эти два эффективных фильтра передискретизации выборки.

### 8.3.1. Фильтры полуполосы

Как уже упоминалось, фильтр полуполосы имеет такое свойство, что каждый его выходной отсчет чередуется с нулевым значением (кроме нулевого индекса выборки). Фильтр полуполосы рассчитывается теми же самыми методами проектирования фильтра, которые будут рассматриваться в разд. 16 (методами окна или Parks-McClellan), при этом устанавливается граничная частота фильтра, равная  $1/4$  частоты дискретизации.

На рис. 8.7 и 8.8 показаны соответственно блок-диаграмма и передняя панель ВП Halfband.vi.

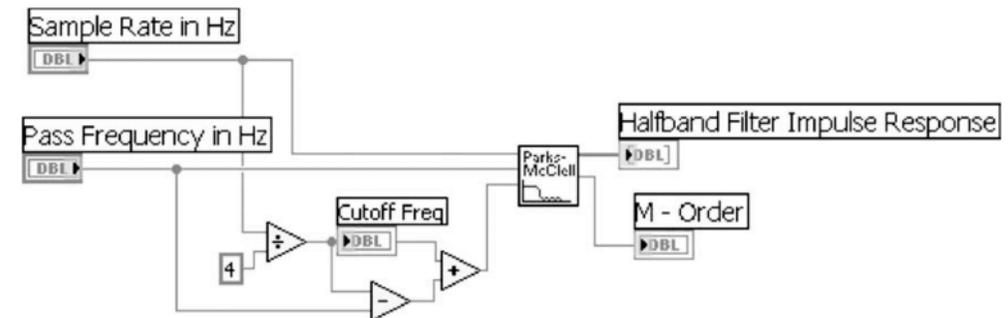


Рис. 8.7. Блок-диаграмма ВП Halfband.vi:

Из блок-диаграммы на рис. 8.7 можно видеть, что граничная частота фильтра определена равной  $1/4$  частоты дискретизации, а частота среза (Cutoff Freq) вычислена по значениям граничной частоты и полосы пропускания. Эти величины установлены в ВП MPR.vi, и импульсная характеристика фильтра показана на рис. 8.8. Импульсная характеристика имеет явную особенность, заключающуюся в том, что каждый второй отсчет равен 0. Можно использовать эти нули, чтобы сократить требуемое число умножений наполовину. Очевидно, что при построении фильтра полуполосы подразумевается, что коэффициент изменения частоты дискретизации соответствует 2 или 0,5. Для больших изменений частоты дискретизации каскады фильтров передискретизации в два раза, сопровождаемые фильтрами полуполосы, могут быть объединены.



В средней блок-диаграмме ВП на рис. 8.10 для сравнения с другими методами обычным методом повышается частота дискретизации входного сигнала, т. е.  $(I - 1)$  нулей заполняют интервалы между каждым реальным и мнимым входными отсчетами и результирующий сигнал сворачивается с коэффициентами фильтра, вычисленными с помощью алгоритма Parks-McClellan. Это означает, что сигнал, который проходит через фильтр, уже имеет повышенную частоту дискретизации, так что фильтрация происходит при более высокой частоте. Если исходная вход-

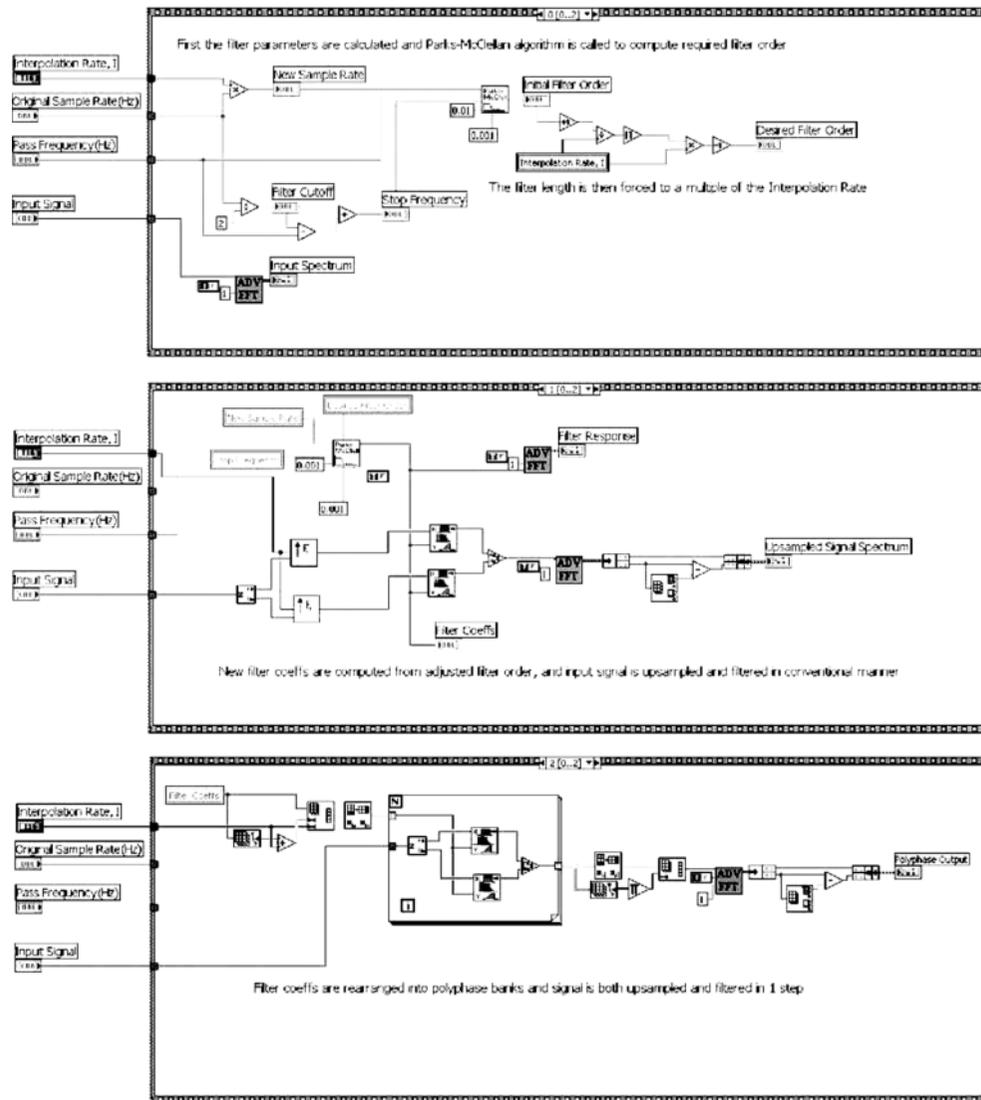


Рис. 8.10. Блок-диаграммы ВП PolyphaseInterpolator.vi

ная последовательность имеет продолжительность  $K$  отсчетов, а фильтр –  $N$  отсчетов, свертка требует  $(K + N - 1)$  умножений (комплексных).

Если на входе выполняется первое повышение частоты дискретизации в  $I$  раз, свертка требует  $(IK + N - 1)$  умножений. На каждый выходной отсчет это традиционное повышение частоты дискретизации, сопровождаемое фильтрацией, требует  $N$  комплексных умножений. Полифазный фильтр разделен на блоки длиной  $N/I$ , и каждый выход, таким образом, требует меньшего объема вычислений.

Наконец, в последней (нижней) структуре выполнена полифазная вставка. На первом шаге необходимо перестроить коэффициенты фильтра в  $I$  блоках фильтрации, как в выражении (4.9), используя изменяющие и перемещающие операторы. Теперь входная последовательность свернута с каждым блоком коэффициентов фильтра, и выходные отсчеты чередуются, чтобы произвести соответствующую отфильтрованную последовательность.

На рис. 8.11 показана передняя панель ВП, производящего полифазные интерполяции.

Входной спектр и АЧХ фильтра показаны в верхней части рисунка, а выходные спектры при обычном и полифазном повышении частоты дискретизации показаны в нижней его части. Два выходных спектра, на первый взгляд, соответствуют друг другу. Этот пример должен убедить читателя, что разделение полифазных коэффициентов фильтра действительно способствует появлению той же самой последовательности, что и у обычного фильтра повышения частоты дискретизации, при этом такая операция обеспечивает существенную экономию вычислений на каждом выходном отсчете, особенно если коэффициент интерполяции большой.

Из предыдущих рассуждений ясно, что полифазное уменьшение частоты дискретизации может быть построено на основе того же самого принципа, а именно: уменьшение частоты дискретизации может произойти *прежде, чем* выполнена операция фильтрации, а сам фильтр может быть разделен на  $M$  блоков. При этом каждый блок фильтра будет оперировать последовательностью с более низкой частотой дискретизации, для которой требуется меньший объем вычислений.

## Резюме

В этом разделе представлены некоторые методы обычных изменений частоты дискретизации.

Изменение частоты дискретизации может быть очень простым действием, которое осуществляется или вставкой нулей во входную последовательность сигнала, или удалением некоторых значений этой последовательности. Были рассмотрены построенные в LabVIEW два фильтра, эффективно изменяющие частоту дискретизации: фильтр полуполосы и полифазный фильтр. Только операции полифазной интерполяции показали, как следует выполнять полифазное уменьшение частоты дискретизации. Конечно, есть более эффективные схемы фильтрации, такие, например, как фильтры Гильберта, которые здесь не были рассмотрены. Для получения дальнейшей информации относительно эффективных фильтров пере- дискретизации выборки можно обратиться к [9].

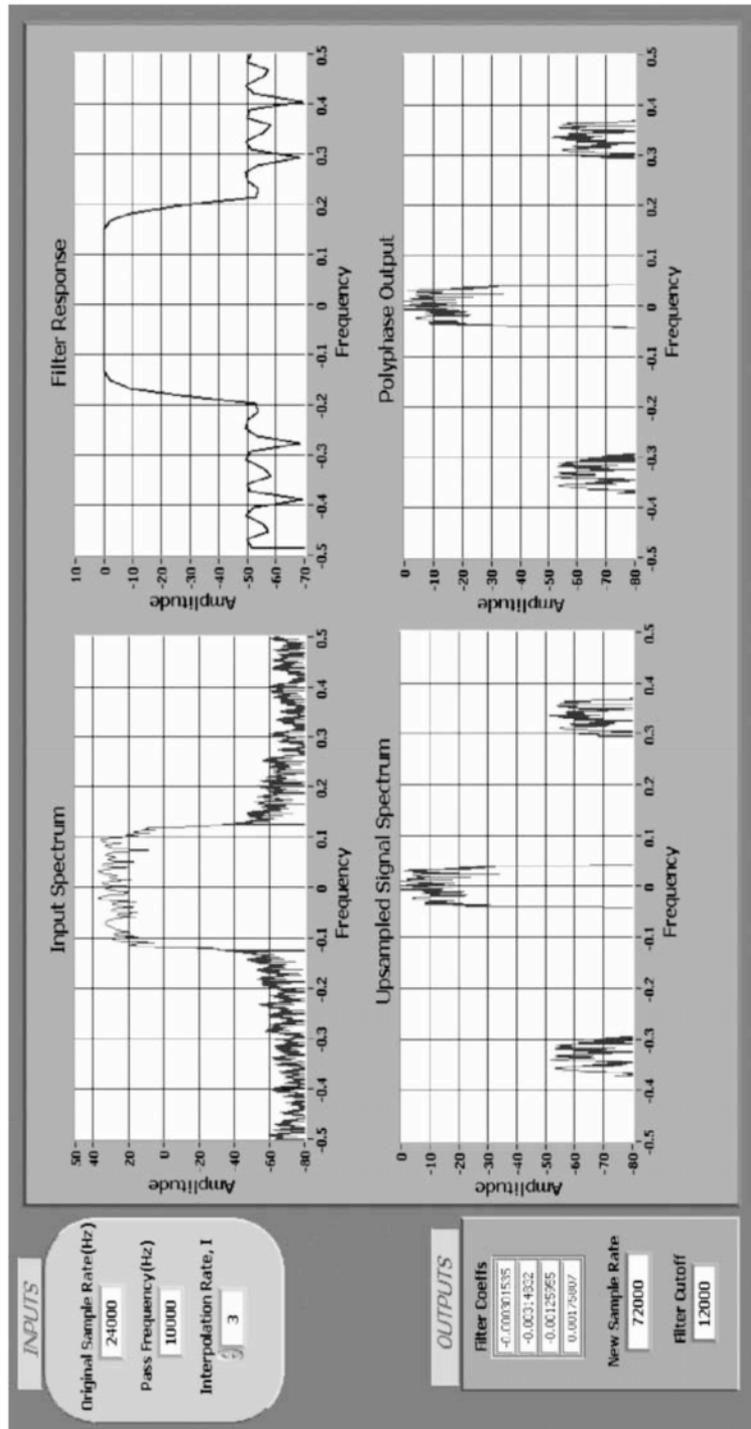


Рис. 8.11. Передняя панель ВП PolyphaseInterpolator.vi

## Генерация сигналов в LabVIEW

9

В этом разделе будет показано, как генерируются обычные сигналы, с которыми сталкиваются в различных системах связи. В разделе представлены некоторые из наиболее употребимых сигналов, которые используются для работы с цифровыми сигналами в цифровых сигнальных процессорах (Digital Signal Processor – DSP) и цифровых системах связи. В библиотеку LabVIEW уже включены некоторые функции для того, чтобы получать большинство этих простых сигналов. В цифровых сигнальных процессорах и системах связи часто используются синусоидальные, импульсные, пилообразные колебания и белые гауссовские шумы. На рис. 9.1 показано местоположение этих функций в палитре, где представлены различные ВП для генерации сигнала.

### 9.1. Основные функции

Некоторые из самых полезных сигналов, используемых в DSP, являются также и самыми простыми.

Три из наиболее широко используемых импульсных сигналов следующие:  $\delta$ -импульс, пилообразный импульс и ступенчатый импульс. На рис. 9.2 показана блок-диаграмма ВП BasicFcns.vi, в котором используются встроенные функции LabVIEW, для того чтобы произвести эти три сигнала. Фактически эти ВП называют Impulse Pattern.vi, Ramp Pattern.vi и Pulse Pattern.vi.

Функция импульса представлена следующим выражением и может использоваться как входной сигнал при извлечении коэффициентов фильтра для операции фильтрации:

$$\delta[n - n_D] = \begin{cases} 1, & n = n_D; \\ 0, & n \neq n_D. \end{cases} \quad (9.1)$$

Она может также использоваться для получения плоского шумового спектра.

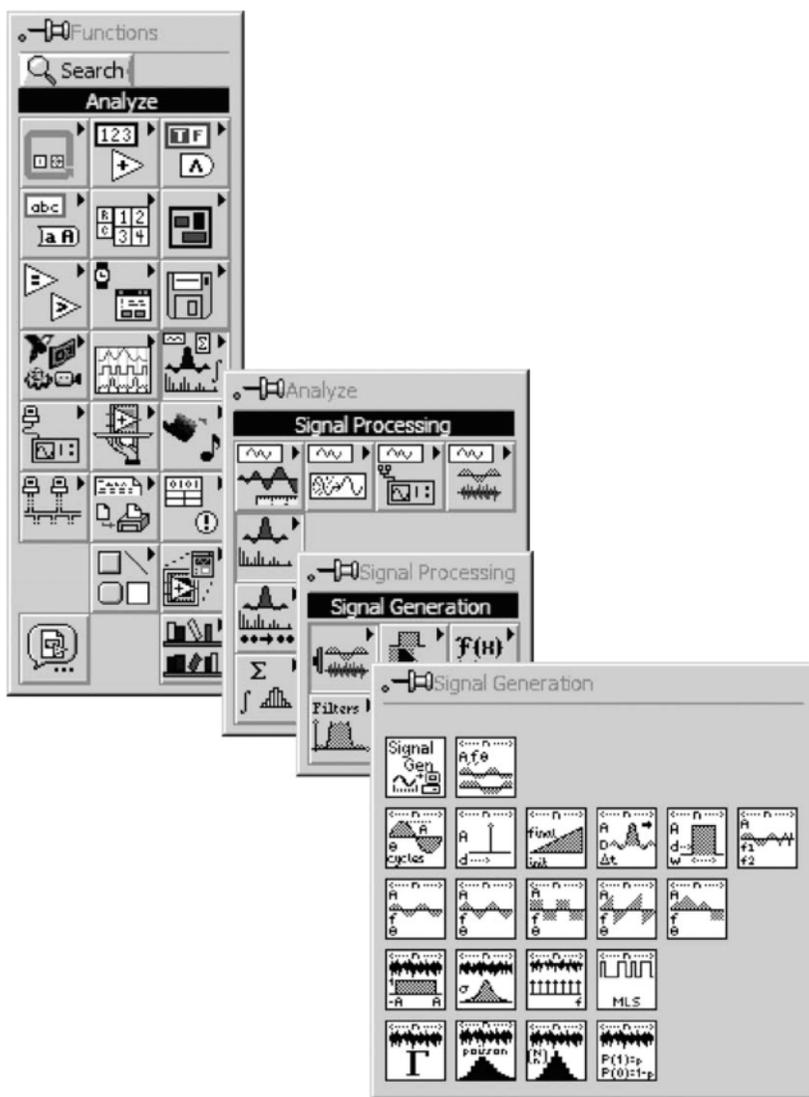


Рис. 9.1. Палитры ВП генераторов сигналов

Этот ВП имеет три ввода параметров: число отсчетов, амплитуда импульса и задержка  $n_D$  в выражении.

Пилообразная функция применяется для получения любой последовательности линейного типа. Например, она используется для создания оси частот при изображении графиков спектра исследуемых сигналов. Такие ВП также имеют три ввода параметров: число отсчетов, номер отсчета в начале и номер отсчета в конце сигнала. Интересная особенность ВП Ramp Pattern.vi состоит в том, что он способен к формированию или нарастающего, или спадающего характера пи-

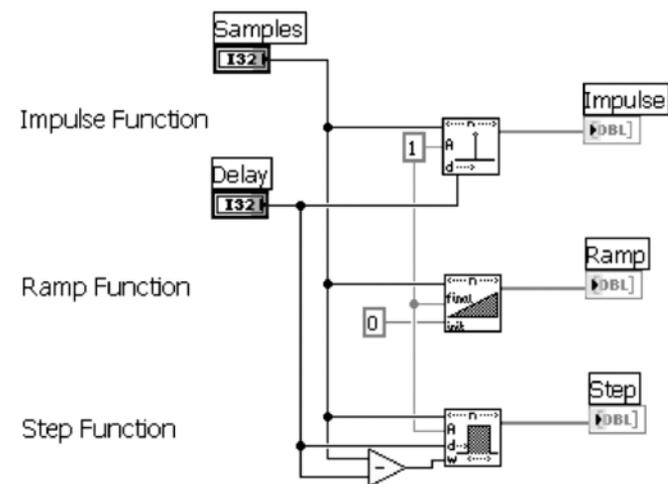


Рис. 9.2. Блок-диаграмма ВП BasicFcns.vi

лообразной функции в зависимости от соотношения между номерами отсчетов конца и начала сигнала.

ВП ступенчатой функции, или Pulse Pattern.vi, имеет четыре ввода параметров: число отсчетов, пиковое значение импульса, задержка и длительность. Уравнение для ступенчатой функции следующее:

$$u[n - n_D] = \begin{cases} 1, & n_D \leq n \leq (n_D + n_W - 1); \\ 0, & n < n_D; n \geq (n_D + n_W), \end{cases} \quad (9.2)$$

где  $n_W$  – длительность импульса.

Участки в области времени всех трех функций показаны на рис. 9.3.

## 9.2. Синусоиды

Другой очень важный класс сигналов в DSP и цифровых системах связи – синусоидальные колебания. Уже были представлены некоторые примеры таких колебаний типа поднятого косинуса и функция  $\text{sinc}(x)$ . Здесь также будет рассмотрена разработка комплексного преобразователя и пример последовательности отсчетов для линейной модуляции частоты.

### 9.2.1. Комплексный преобразователь

Поскольку LabVIEW имеет программу для производства модели синуса, можно использовать ее как основание для того, чтобы построить комплексный преобразователь. Необходимо также знать частоту дискретизации, желательную длину и частоту преобразования. LabVIEW производит модель синуса как синусоидальное колебание заданного числа периодов (представленные части периодов) в ука-

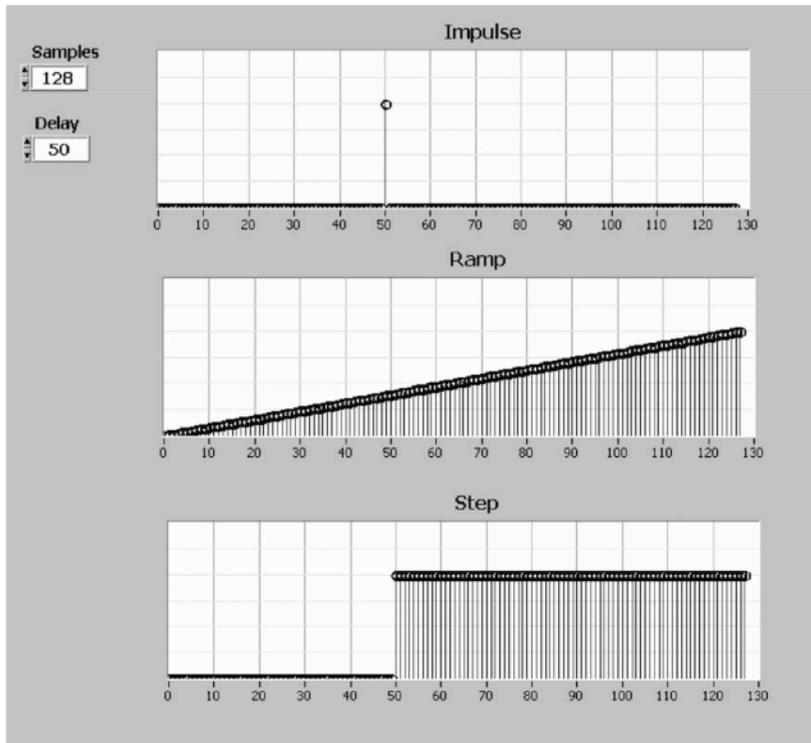


Рис. 9.3. Фронтальная панель ВП BasicFcns.vi

занном числе единичных отсчетов. ВП, блок-диаграмма которого показана на рис. 9.4, делит длительность преобразования с учетом частоты дискретизации, чтобы получить отрезок времени в виде числа периодов дискретизации.

Тогда время, умноженное на частоту преобразования, дает число периодов. Можно произвести косинусное колебание, заменив в генераторе сдвиг по фазе модели синуса на 90°. Наконец, для компактности можно объединить в комплексное число результаты синусного и косинусного преобразований в виде мнимой и реальной частей этого комплексного числа.

### 9.2.2. Функция sinc(x)

Другая полезная функция, которая была кратко описана в разд. 7, – sinc-функция, вырабатываемая ВП, названным SincFcn.vi. Этот ВП вычисляет sinc-форму колебания по следующей формуле [4]:

$$h[n] = \frac{\sin[\omega_c(n - M/2)]}{\pi(n - M/2)} \quad (9.3)$$

Блок-диаграмма ВП, который вычисляет выражение (9.3), показана на рис. 9.5. Можно заметить, что есть исключительная точка, где функция sinc(x) вела бы

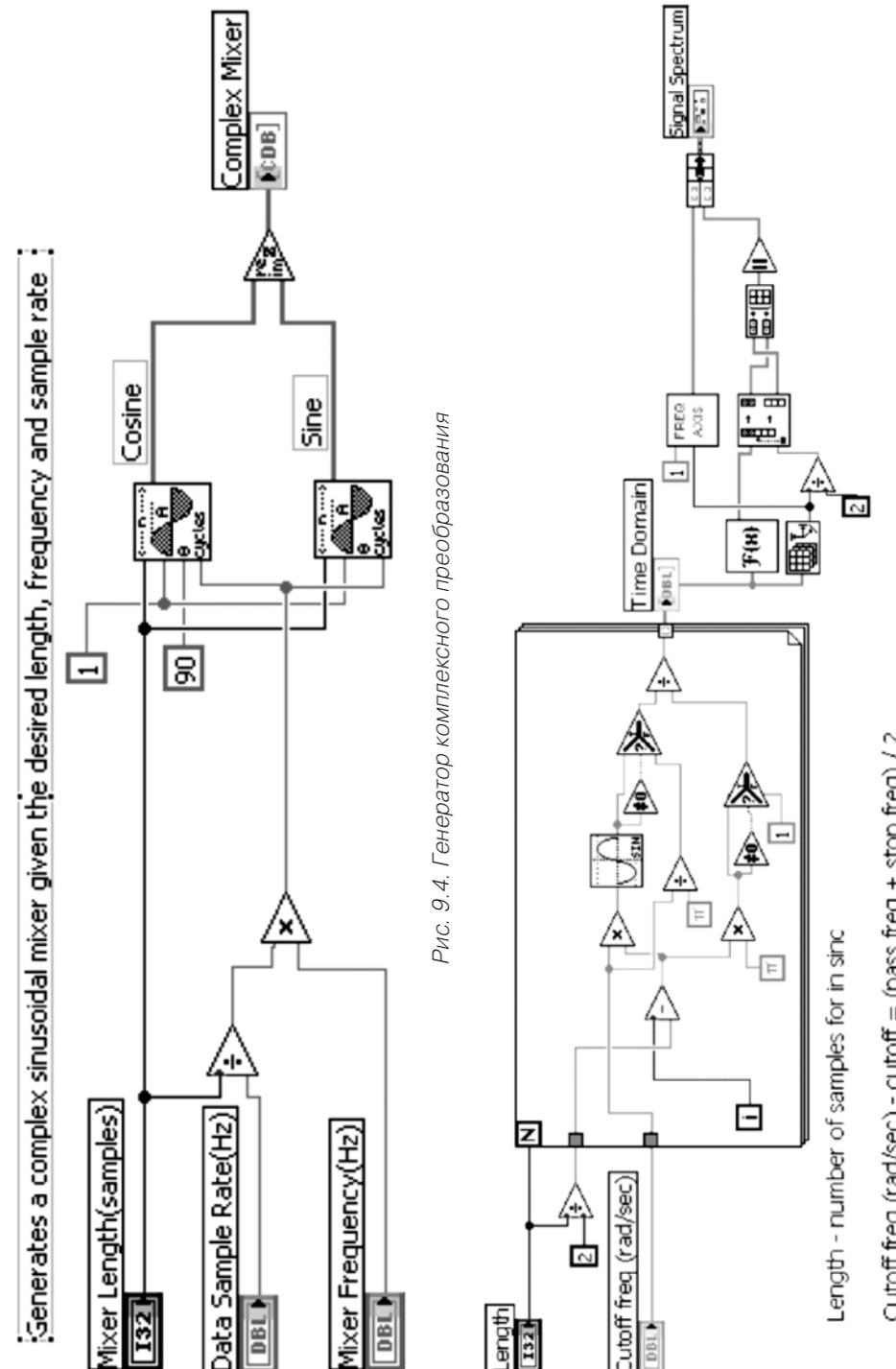


Рис. 9.4. Генератор комплексного преобразования

Length - number of samples for in sinc  
Cutoff freq (rad/sec) - cutoff = (pass freq + stop freq) / 2

Рис. 9.5. Блок-диаграмма ВП SincFcn.vi

себя как первый замечательный предел в математике, если бы этим поведением не управляли так или иначе. Это точка, где  $n = M/2$ , в которой согласно выражению (9.3) производится деление 0/0. Эта операция контролируется в ВП двумя селекторами, которые устанавливают максимальное значение функции  $\text{sinc}(x)$  в определенное положение в последовательности отсчетов.

Передняя панель ВП SincFcn.vi показана на рис. 9.6. Здесь используется усеченная функция  $\text{sinc}(x)$  (необходимо помнить, что реальная функция  $\text{sinc}(x)$  имеет бесконечную продолжительность). При увеличении длительности колебания  $\text{sinc}(x)$  его спектр приближается по форме к АЧХ идеального ФНЧ.

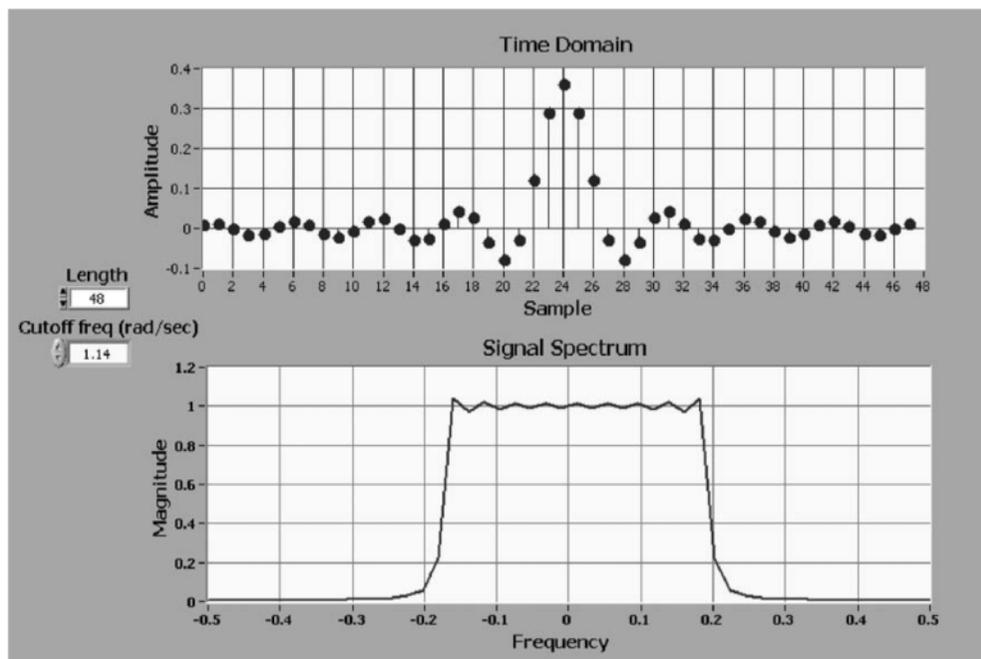


Рис. 9.6. Фронтальная панель ВП SincFcn.vi

### 9.2.3. Линейно-частотно-модулированная (ЛЧМ) последовательность

Последовательность ЛЧМ – интересный сигнал, который имеет применения в звуковых, радарных и в сверхширокополосных (СШП) системах связи. Есть две различные формы ЛЧМ-последовательности – линейная и показательная, которые относятся к коэффициенту в показателе частоты, в соответствии с которыми изменяется частота ЛЧМ-сигнала. Более простой из этих двух – линейный ЛЧМ-сигнал, который будет рассмотрен ниже. Следующее уравнение является формулой для определения линейной ЛЧМ-последовательности:

$$w[n] = \sin\left(\frac{2\pi f}{f_s} n^k\right). \tag{9.4}$$

Блок-диаграмма на рис 9.7 показывает, как ВП для генерации ЛЧМ-последовательности может быть осуществлен.

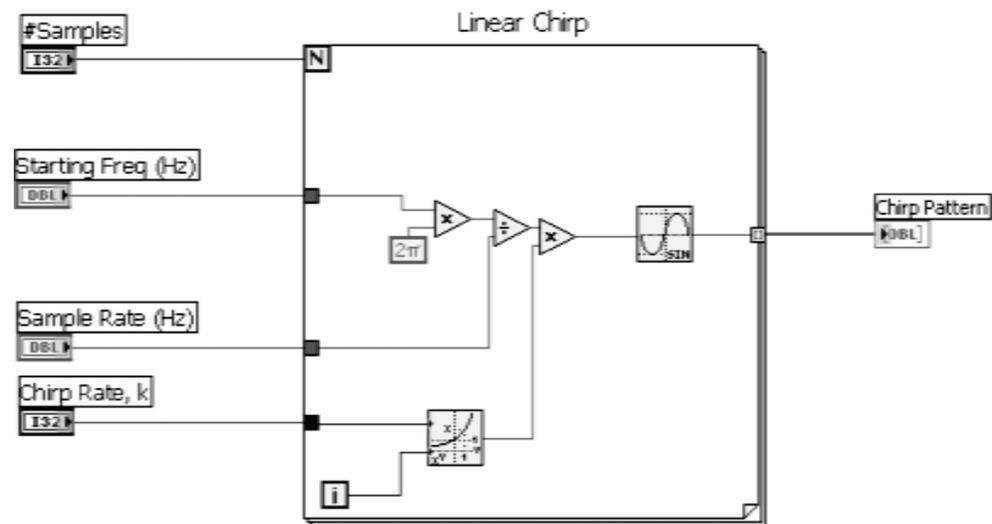


Рис. 9.7. Блок-диаграмма ВП Chirp.vi

ВП Chirp.vi линейно увеличивает частоту колебания от заданной частоты в начале сигнала до значения, определяемого числом единичных отсчетов. Мгновенная частота колебания определяется производной от полной мгновенной фазы по времени. Произведенная модель ЛЧМ-последовательности показана на рис. 9.8.

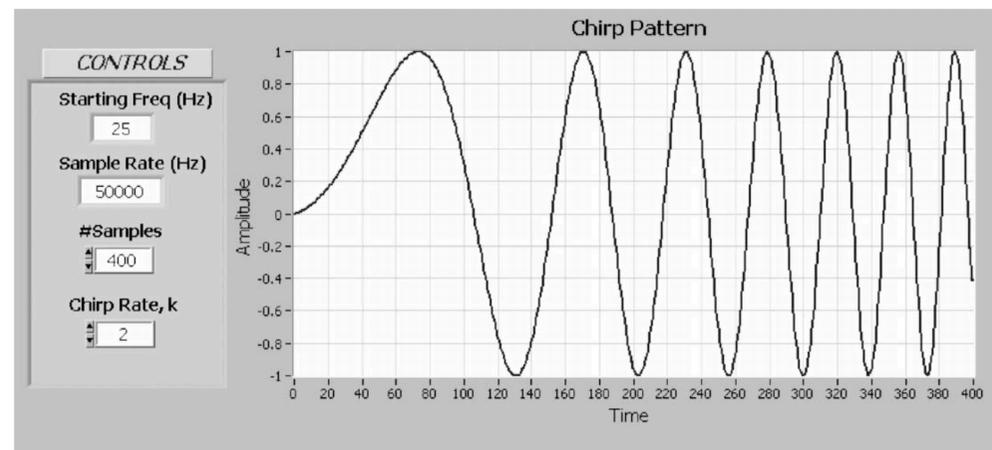


Рис. 9.8. Фронтальная панель ВП Chirp.vi

## 9.3. Генерация моделей канала связи

Важная часть цифровых систем связи моделирует восстановление сигнала на фоне шума и других помех в канале связи. Чтобы реально обеспечить влияние помехи в системе связи через ее синхронизацию, необходимо генерировать модель канала.

Изменения амплитуды и фазы сигнала, обусловленные его распространением в канале связи, могут вообще разрушить сигнал и передаваемую информацию. Есть несколько различных типов разрушения информации, так же как и нескольких различных моделей, чтобы моделировать условия разрушения.

Для простоты можно рассмотреть только ограниченные разрушения. Рекомендации, приведенные в [12], показывают, что для двумерных изотропных окружающих сред рассеивания модуль полученного комплексного сигнала  $g(t)$  может быть распределен по закону Рэля как случайная переменная в любой заданный момент времени, а его фаза равномерно распределена от  $-\pi$  до  $\pi$ .

В подразд. 9.3.1 будет объяснено, как строить рэлеевскую модель канала связи. В дополнение к рассеиванию есть другие источники шума, которые можно рассмотреть. Например, приемник будет иметь некоторый собственный тепловой шум и может быть добавление мешающего узкополосного случайного сигнала с другой радиочастотой (RF – Radio Frequency), обусловленное недостаточной электромагнитной совместимостью каналов. Шумы этих посторонних источников будут моделироваться в пункте 9.3.2 как гауссовский шум.

### 9.3.1. Распределение Рэля

Рэлеевская огибающая в LabVIEW производится следующим образом. На рис. 9.1 показано, что LabVIEW содержит функцию для получения отсчетов белого гауссовского шума (а также одностороннего белого шума, шума с распределениями Гамма, Пуассона и биномиального шума). Из библиографических источников [12, 13] известно, что, если мы имеем случайную переменную  $R$ , определенную как

$$R = \sqrt{[g_I(t)]^2 + [g_Q(t)]^2} \quad (9.5)$$

и распределенную согласно закону Рэля, тогда мнимая  $g_I(t)$  и реальная  $g_Q(t)$  составляющие имеют нулевые средние и гауссовские распределения (когда число путей рассеивания увеличивается). Связь выражения (9.5) с восстановленным аналоговым аналитическим сигналом следующая:  $g(t) = g_I(t) + jg_Q(t)$ , и это есть его комплексная огибающая, а  $R = |g(t)|$  есть модуль этой огибающей. Можно теперь использовать эту информацию и разработать ВП для получения белого гауссовского шума, чтобы построить блок-диаграмму ВП Rayleigh.vi, показанную на рис. 9.9.

На входе ВП Rayleigh.vi устанавливаются два параметра: число единичных отсчетов и стандартное среднеквадратическое отклонение (СКО) рассеивания огибающей. Так как стандартное отклонение есть квадратный корень из мощности сигнала, СКО делится на  $\sqrt{2}$ . Как реальная, так и мнимая части огибающей Рэля содержат половину полной мощности сигнала. Термин «рассеивание» возник в свя-

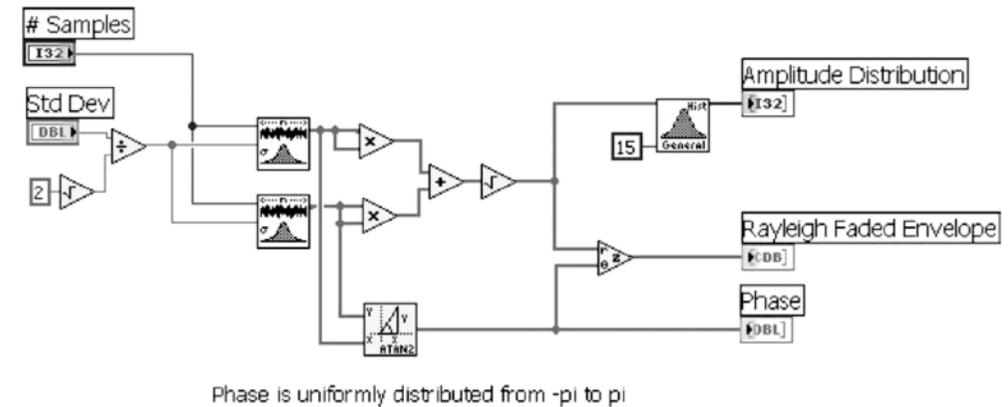


Рис. 9.9. Блок-диаграмма ВП Rayleigh.vi

зи с тем, что малые замирания огибающей есть результат приема многократных преотражений переданного сигнала от препятствий на пути его распространения. Рассеивание может быть обусловлено любыми объектами в окружающей среде (здания, деревья, автомобили и т. д.). На рис. 9.9 представлены две последовательности  $g_I(t)$  и  $g_Q(t)$ , сформированные из случайных чисел с гауссовским распределением, они возводятся в квадрат, суммируются, и затем извлекается квадратный корень из этой суммы, что соответствует рэлеевской огибающей. Наконец, арктангенс отношения  $g_Q(t)/g_I(t)$  дает равномерно распределенную фазу на интервале  $[-\pi, \pi]$ .

На рис. 9.10 можно видеть, что гистограмма модуля комплексной огибающей действительно соответствует закону Рэля.

Теперь, когда имеется модель для модуля коэффициента передачи канала и его аргумента на основе ВП Rayleigh.vi, можно начать строить канал с замираниями огибающей для распространения модулированного сигнала. Чтобы сделать это, нужно просто определить необходимое число рэлеевских путей распространения сигнала, подвести итог по их вкладам в сумму и умножить полученный результат на модулированный сигнал.

Надо иметь в виду, что при моделировании рэлеевских замираний огибающей не учитывается прохождение компоненты сигнала в пределах прямой видимости (по прямому пути). Этот тип распределения учитывается в обобщенном распределении Рэля – Райса, для которого распределение Рэля является частным случаем. В этом пункте делается попытка только показать возможности моделирования канала в LabVIEW, а учет компоненты сигнала прямой видимости в канале связи предлагается сделать самому читателю.

### 9.3.2. Белый гауссовский шум

Кроме замираний (мультипликативная помеха) в канале связи будут помехи от RF-сигналов соседних каналов и от аддитивного теплового шума в приемнике. Можно моделировать эти шумы как белые гауссовские шумовые процессы, какие

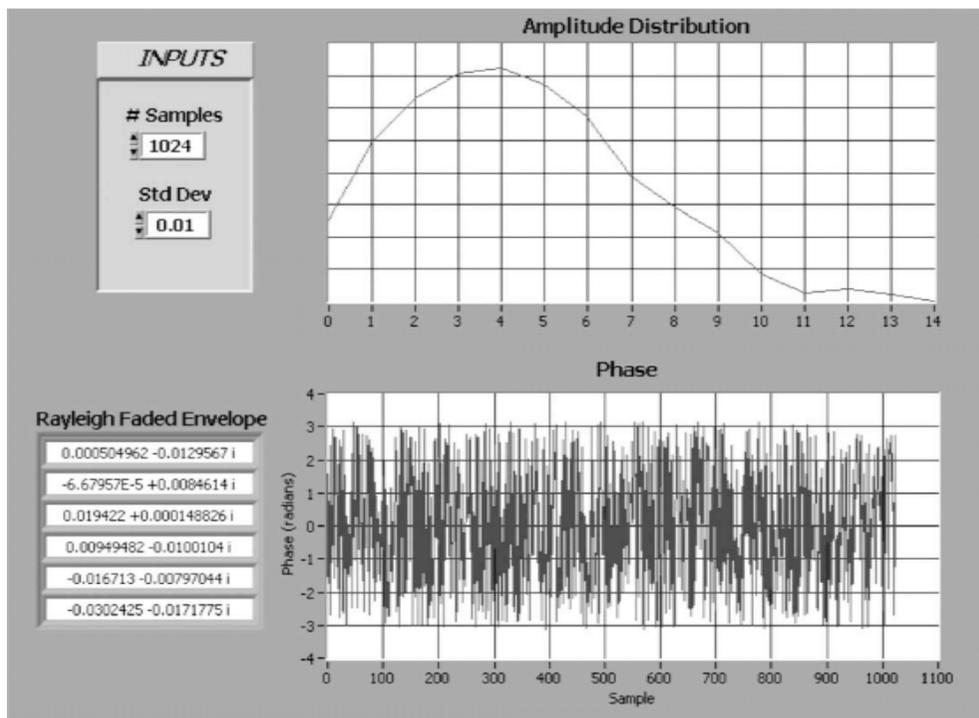


Рис. 9.10. Передняя панель ВП Rayleigh.vi

LabVIEW может легко произвести на основе использования ВП белого гауссовского шума, который применялся в подразд. 9.3.1. Блок-диаграмма ВП AWGN.vi показана на рис. 9.11, а лицевая панель – на рис. 9.12.

Этот ВП просто формирует как реальную, так и комплексную белые гауссовские шумовые последовательности с заданной дисперсией. Экспертиза шумового спектра показывает, что он является плоским по частоте, что характерно для белого шума.

Последний шаг перед моделированием целой цифровой структуры связи заключается в генерации в цифровой форме модулируемого сигнала. Это означает, что необходимо выбрать цифровую схему модуляции, с помощью которой производится QAM (квадратурная амплитудная модуляция), QPSK (фазовая манипуляция с четвертичными сигналами), PAM (амплитудно-импульсная модуляция) или любая другая модуляция. Возникает также необходимость в создании реальной карты символов для выбранного вида модуляции. Создание символов столь же просто, как и использование в LabVIEW генератора случайных чисел (рис. 9.13). Генератор случайных чисел производит равномерно распределенные числа в интервале между 0 и 1. Двойная точность случайного числа обеспечивается, когда оно приводится к 16-битовым целым числам за счет округления до самого близкого целого числа.

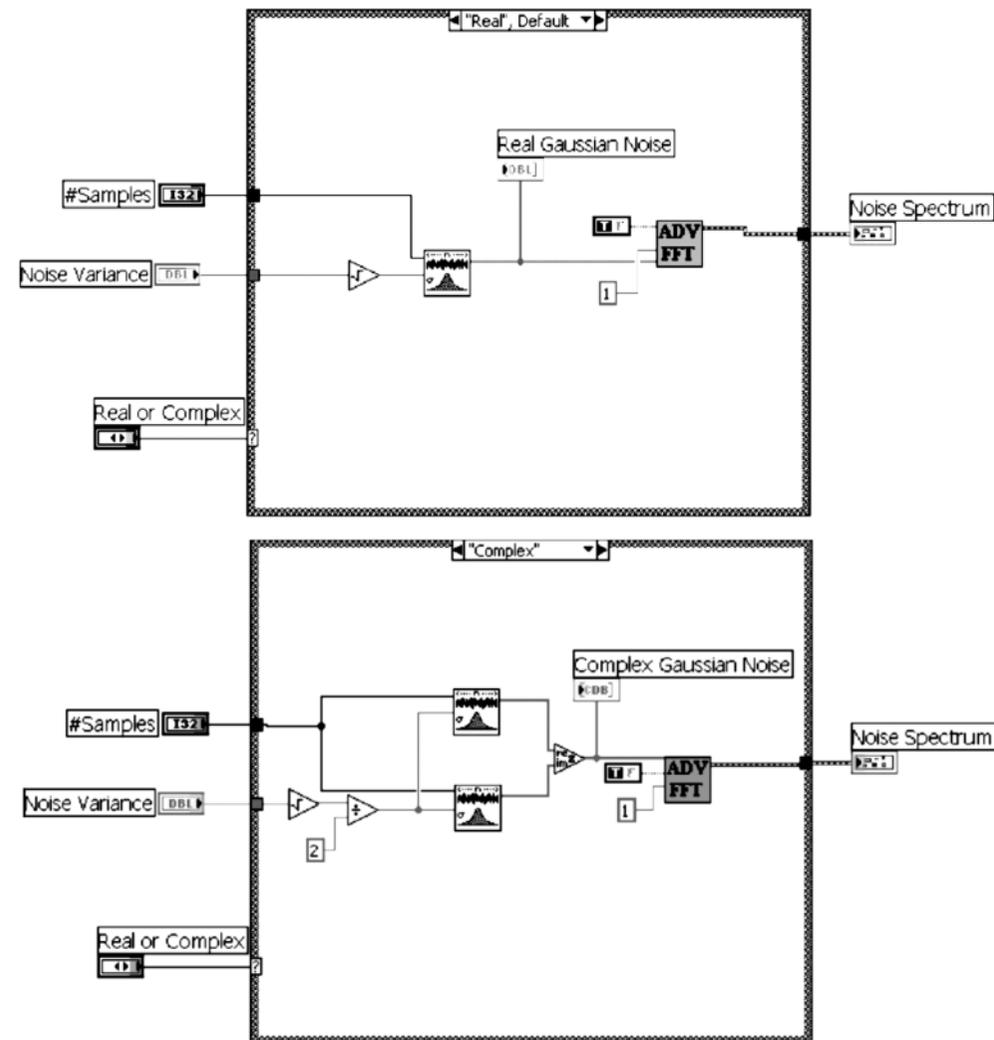


Рис. 9.11. Блок-диаграммы ВП AWGN.vi

## 9.4. Формирование символов

Как только будет определено количество битов, их необходимо нанести на карту символов модуляции. Для этой операции была выбранная 16-битовая схема формирования сигнала QAM. Чтобы наносить на карту биты в символах, следует использовать таблицу символов в кодах Грея для выбора соответствующего символа, индексируя множество символов для передачи по каналу связи [6]. Код Грея позволяет уменьшить величину ошибки при нанесении на карту соседних симво-

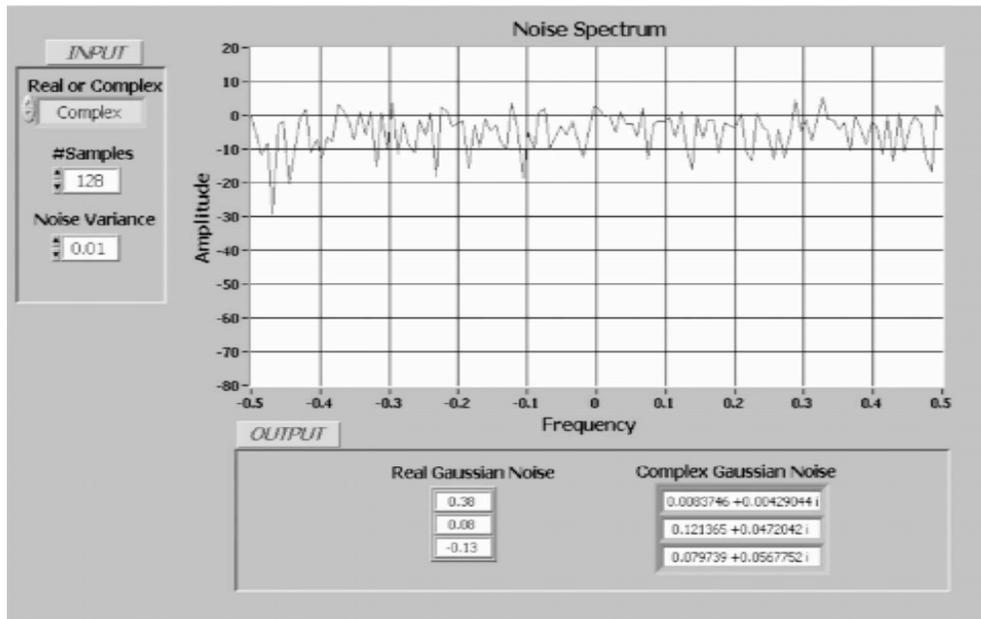


Рис. 9.12. Лицевая панель ВП AWGN.vi

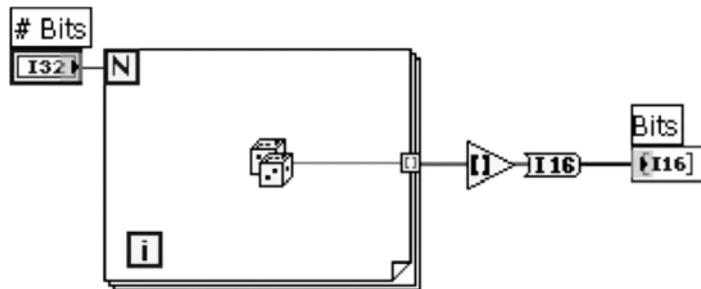


Рис. 9.13. Блок-диаграмма ВП GeneratorBits.vi

лов за счет удаления значений, которые отличаются только единственным битом. Это означает, что, если соседний символ выбран неправильно, ошибка будет равна только 1 бит. Так как код Грея находится на стадии проекта, это хороший повод для того, чтобы сократить величину ошибки с нулевой стоимостью в терминах вычисления. На рис. 9.14 показан 16-QAM-построитель карты символов с таблицей символов кода Грея.

С помощью 16-QAM-модуляции заносится на карту 4 бита для каждого символа, и можно видеть на рис. 9.14, что входное множество битов разделено в индексе 4 с ближайшими 4 битами, преобразованными к числовому индексу для таблицы символов, а остающиеся биты передаются в регистр изменения для следующего

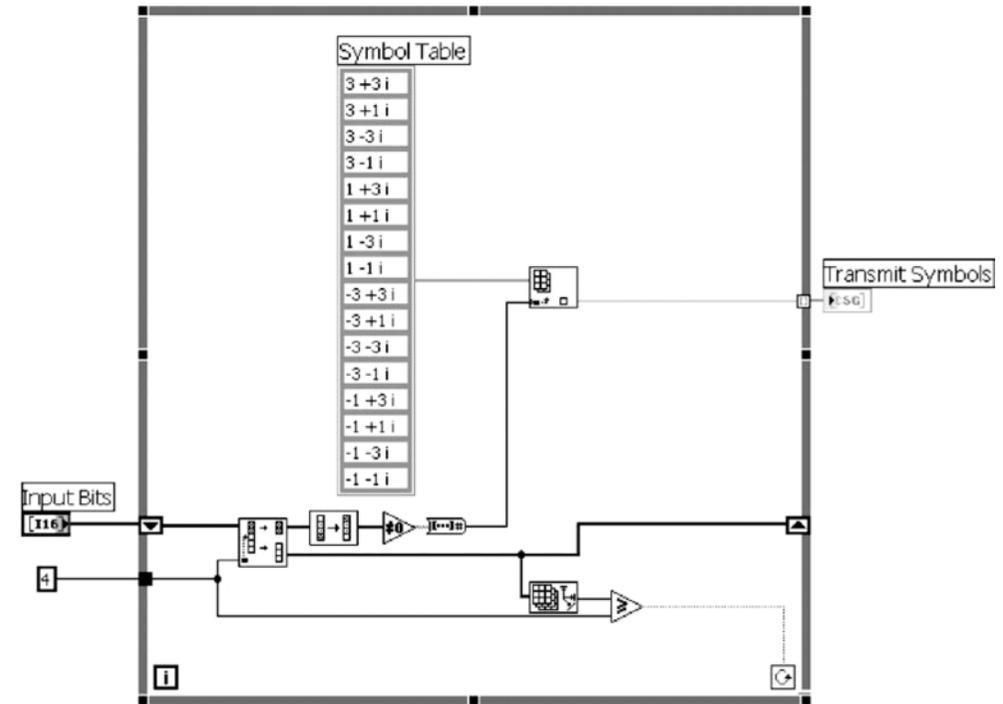


Рис. 9.14. Блок-диаграмма ВП SymbolMapper.vi

повторения цикла. Множество величиной в 4 бита сначала полностью реверсируется, что определяется видом булевого множества в операциях, т.е. функциях от числа. Можно тогда рассмотреть эту совокупность, связывающую реальные и мнимые части символов, и создать X-Y-график, как показано на блок-диаграмме ВП (рис. 9.15). Передняя панель ВП X-Y-графика показана на рис. 9.16.

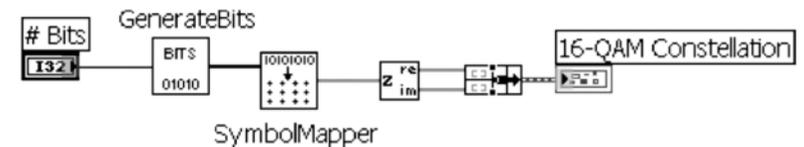


Рис. 9.15. Подготовка совокупности

## Резюме

В этом разделе представлены некоторые из генераторов сигнала LabVIEW.

Конечно, возможности LabVIEW для построения модели цифрового канала связи рассмотрены только поверхностно, но можно надеяться, что все-таки здесь приведено достаточно информации, чтобы продолжить изучение этого пакета.

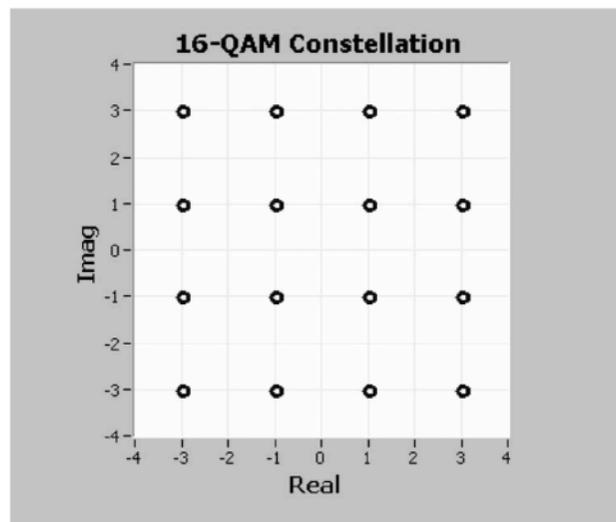


Рис. 9.16. 16-QAM-совокупность X-Y-графика

## Сборка узлов системы связи



В разд. 6–9 было рассмотрено большинство функций базового инструментария LabVIEW DSPtools, в котором может возникнуть необходимость, когда придется иметь дело с цифровыми сигналами связи. В этом разделе можно увидеть, как все эти функции объединяются в полный цифровой пакет связи. Необходимо начать с создания формы передаваемого колебания, как показано в блок-схеме на рис. 10.1. В предыдущих разделах уже были представлены все необходимые ВП, в которых выполнялось формирование модулированного сигнала в цифровой форме, и из этой информации можно получить, по крайней мере, поверхностное представление о работе основных частей блок-схемы модулятора, показанной на рис. 10.1. Теперь можно объединить все эти ВП в виде блоков в рабочий модулятор.

Процесс, обратный цифровой модуляции, представлен на рис. 10.2. Блоки демодулятора такие же, как в модуляторе, однако некоторые из приборов для демодуляции еще не были рассмотрены и будут представлены ниже в этом разделе.

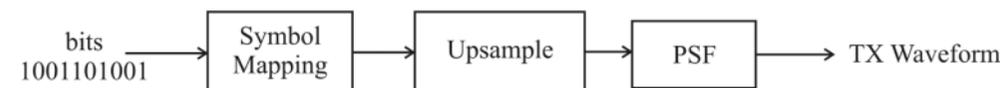


Рис. 10.1. Блок-схема цифрового генератора модулированных колебаний:  
bits – биты входной информации; Symbol Mapping – отображение символов на карте;  
Upsample – повышение частоты дискретизации; PSF (Point Spread Function) – функция  
рассеяния в точке; TX (Time Slot) Waveform – квантованный по времени сигнал

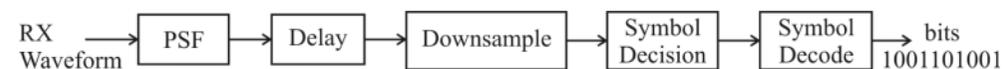


Рис. 10.2. Блок-схема процесса демодуляции модулируемого колебания в цифровой  
форме: Delay – задержка; Downsample – снижение частоты дискретизации; Symbol  
Decision – определение вида символа; Symbol Decode – декодирование символов

После того как создана рабочая схема модуляции/демодуляции, можно моделировать шум на радиочастоте, формировать канал с замираниями, а также строить ВП Channel.vi на основе использования модели канала из разд. 7.

## 10.1. Модулятор

При формировании модулированных колебаний (рис. 10.1) можно применить простой ВП GenerateBits.vi (см. разд. 8), чтобы смоделировать случайную последовательность битов. В подразд. 9.4 был исследован метод отображения этих битов в виде точек на карте кодов Грея для совокупности 16-QAM.

Затем повышается частота дискретизации сигнала основной полосы частот с учетом желательной частоты дискретизации, и данные пропускаются через фильтр для формирования импульсов (см. рис. 10.1). Необходимо помнить, что сигнал будет склонен к межсимвольной интерференции (ISI), если не обеспечить выполнение определенных требований к форме импульса.

Теперь, когда выбраны компоненты модулятора, необходимо исследовать блок-диаграмму ВП Modulator.vi на рис. 10.3.

В начале блок-диаграммы на рис. 10.3 приведены знакомые функции из разд. 9, для того чтобы производить биты и наносить их на карту совокупности 16-QAM. После их запуска в разрабатываемый ВП вызывается ВП NyquistPulse.vi из библиотеки LabVIEW, чтобы спроектировать фильтр для формирования импульса с заданными длительностью, частотой дискретизации и частотой повторения символа, а также коэффициента (Alpha), определяющего склон АЧХ фильтра при переходе от полосы прозрачности к полосе задержания. Здесь формирование импульса производится на желательной частоте дискретизации выходного сигнала, а не на входной частоте символов. Из разд. 4 известно, что задержка импульса формирующим фильтром равна  $M/2$  ( $M$  есть порядок фильтра). И также известно, что  $M = N - 1$ , где  $N$  – длина последовательности коэффициентов фильтра. Таким образом, возникает задержка сигнала в фильтре на число  $N$  элементов длины выборки [4]. Теперь исходная последовательность в виде реальных и мнимых частей символов входного колебания свертывается с формой колебания, соответствующего импульсной характеристике фильтра, при повышенной частоте дискретизации. На заключительном шаге производится спектр в ВП AdvFFT.vi и заменяется (нормируется) значение его максимума на оси Y на 0 дБ. Необходимо обратить внимание также, что повышение частоты дискретизации могло также быть произведено на основе полифазного приближения из разд. 8.

На рис. 10.4 показана передняя панель ВП Modulator.vi.

Необходимо обратить внимание на некоторые важные элементы на рис. 10.4. Во-первых, можно заметить, что есть пять отсчетов на интервалах времени от пикового значения выходного отфильтрованного импульса до первых нулевых значений с обеих сторон относительно максимума отклика фильтра. Это означает, что частота дискретизации в пять раз больше частоты символа. Во-вторых, односторонняя полоса пропускания переданного сигнала равна приблизительно 3000 Гц. Необходимо помнить, что требуемая односторонняя полоса спектра модулирует-

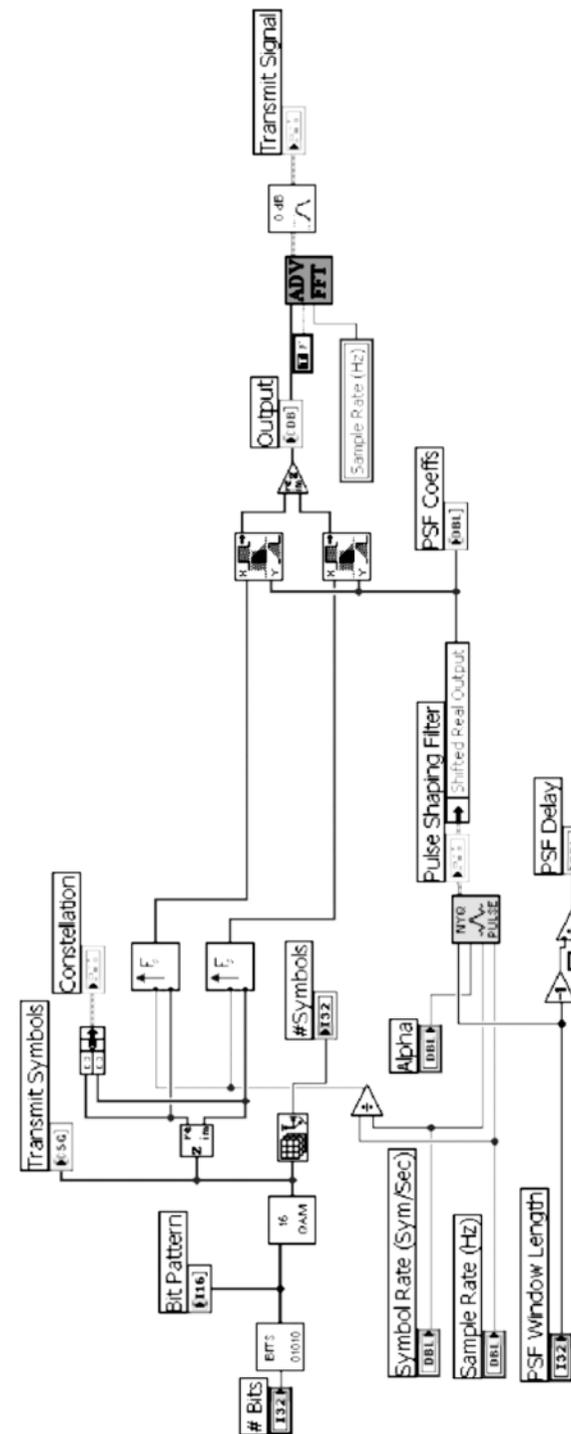


Рис. 10.3. Блок-диаграмма ВП Modulator.vi

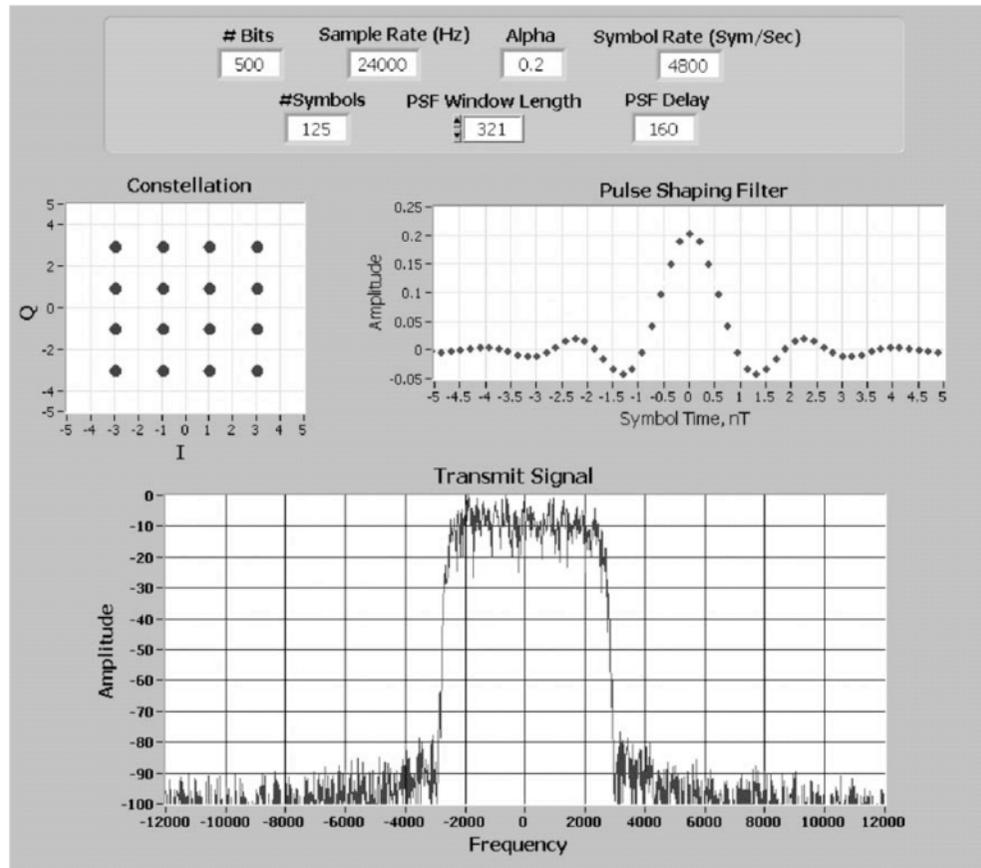


Рис. 10.4. Передняя панель ВП Modulator.vi

мого сигнала в цифровой форме с частотой 4800 символов в секунду должна быть: 2400 Гц плюс избыточная полоса пропускания формирующего фильтра. В этом примере избыточная полоса пропускания составляет 20 % от односторонней полосы пропускания, т. е. 2880 Гц.

Следует демодулировать этот сигнал, чтобы знать наверняка, что всё работает должным образом.

## 10.2. Демодулятор

Прежде чем начать исследовать структуру демодулятора, необходимо быстро проанализировать область времени и отобразить форму колебаний, как показано на рис. 10.5, чтобы удостовериться в правильности посылки сигнала по каналу связи.

Анализ рис. 10.5 показывает, что переданная форма колебаний соответствует приблизительно нулевым значениям для некоторых интервалов в начале и конце

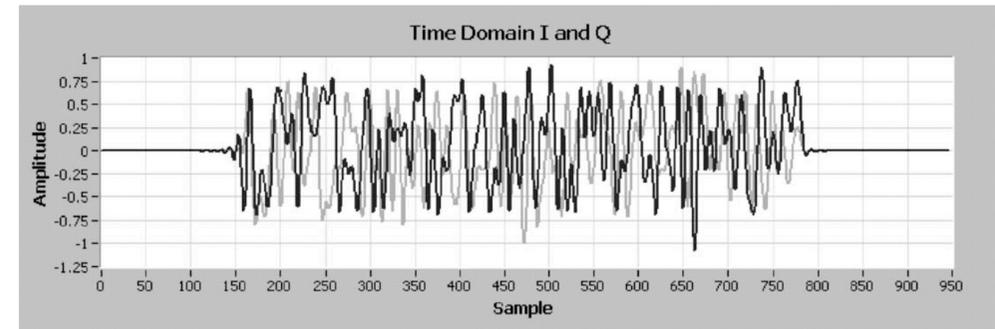


Рис. 10.5. Переданные I и Q формы колебания

передачи. Надо помнить, что выход операции свертки имеет длину  $L + N - 1$ , где  $L$  – длина входной формы колебания (625 отсчетов) и  $N$  – длина импульсной характеристики фильтра (321 отсчет). Поскольку эти две формы колебаний «скользят» относительно друг друга в свертке, получается  $(N - 1)/2$  нулей, так как значения отсчетов коэффициентов фильтра в начале и конце его импульсной характеристики отличаются от нуля и их огибающая повторяет по форме функцию квадратного корня из косинуса до того момента, когда первый входной отсчет будет соответствовать главному лепестку импульсной характеристики фильтра формирования. В конце колебания его форма повторяется, но в инверсном виде. Это означает, что передано  $(N - 1)$  отсчетов, которые действительно не содержат никакой информации. Можно избавиться от них, как только приемник закончит формирование половины импульса.

Из анализа блок-схемы на рис. 10.2 следует, что первая операция, которую приемник должен выполнить, – это закончить фильтрацию другой половины импульса. Необходимо помнить, что форма квадратного корня из функции поднятого косинуса должна с обеих сторон соответствовать выборке из импульсной характеристики фильтра, чтобы привести форму его отклика к форме функции поднятого косинуса. После этой свертки добавляется  $(N - 1)/2$  нежелательных отсчетов по направлению к фронту колебания, и в итоге получается общее количество  $(N - 1)$  отсчетов. По этой причине выходной сигнал задерживается на  $(N - 1)$  отсчетов, и точно в конце  $(N - 1)$ -й позиции появляются дополнительные отсчеты. Теперь можно снизить частоту дискретизации сигнала и обеспечить обработку переданных символов. Возможно, необходимо выполнить некоторые вычисления на основе восстановленных символов, чтобы определить передаточную функцию фильтра, формирующего импульсный сигнал, шум канала и т. д.

На рис. 10.6 показана блок-диаграмма ВП Demodulator.vi, который построен в соответствии со всем описанным в предыдущем абзаце.

ВП ScaleSymbols.vi, показанный на рис. 10.6, обеспечивает вычисления, упомянутые выше в этом подразделе. Блок-диаграмма ВП ScaleSymbols.vi представлена на рис. 10.7. Так как известно значение наибольшего переданного символа, этот ВП просто повышает (или снижает) наибольшую величину отсчета во входной выборке до этого значения.

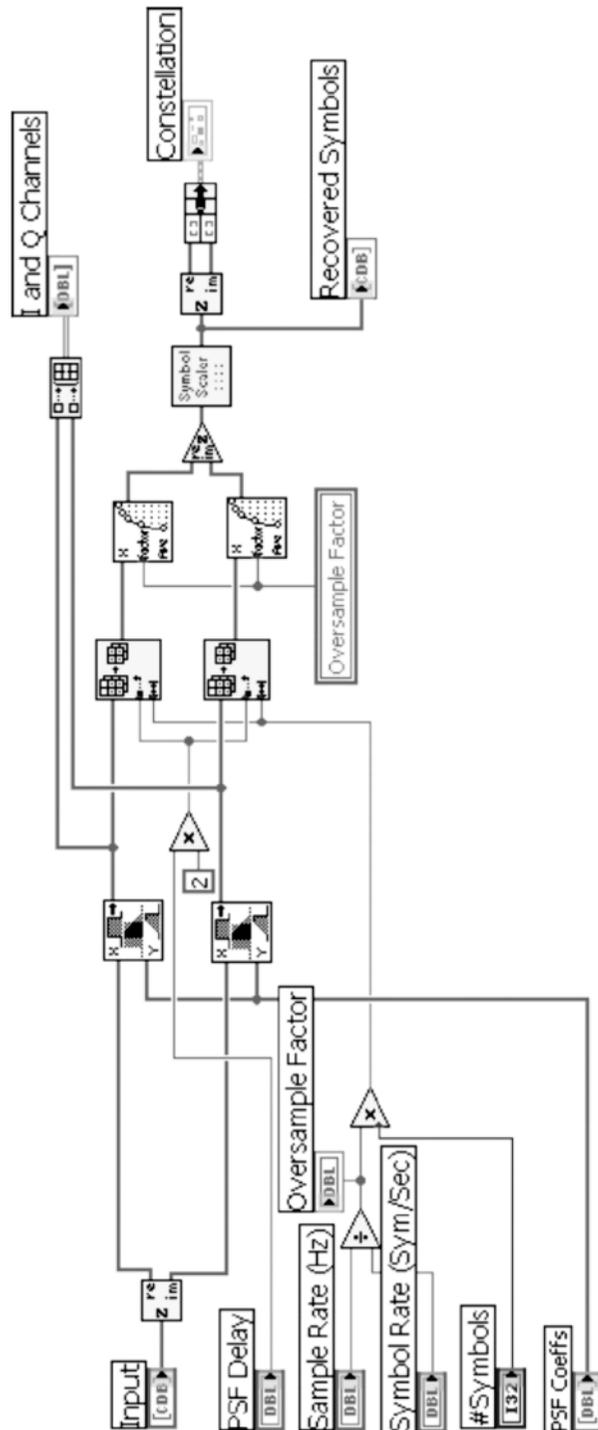


Рис. 10.6. Блок-диаграмма ВП Demodulator.vi

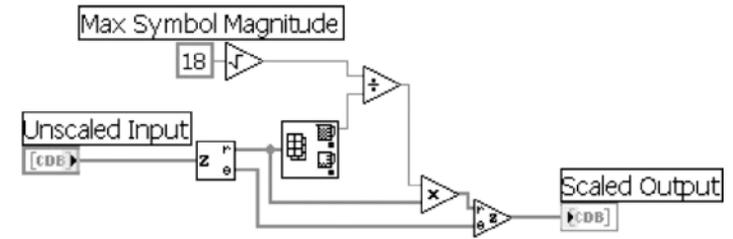


Рис. 10.7. Блок-диаграмма ВП ScaleSymbols.vi: Max Symbol Magnitude – максимальная амплитуда символов; Unscaled Input – масштабирование входного сигнала; Scaled Output – масштабирование выходного сигнала

Передняя панель демодулятора показана на рис. 10.8. Можно заметить, что восстановленные символы – это не совсем точные копии возможных точек входной совокупности. В демодуляторе не используется никаких критериев достовер-

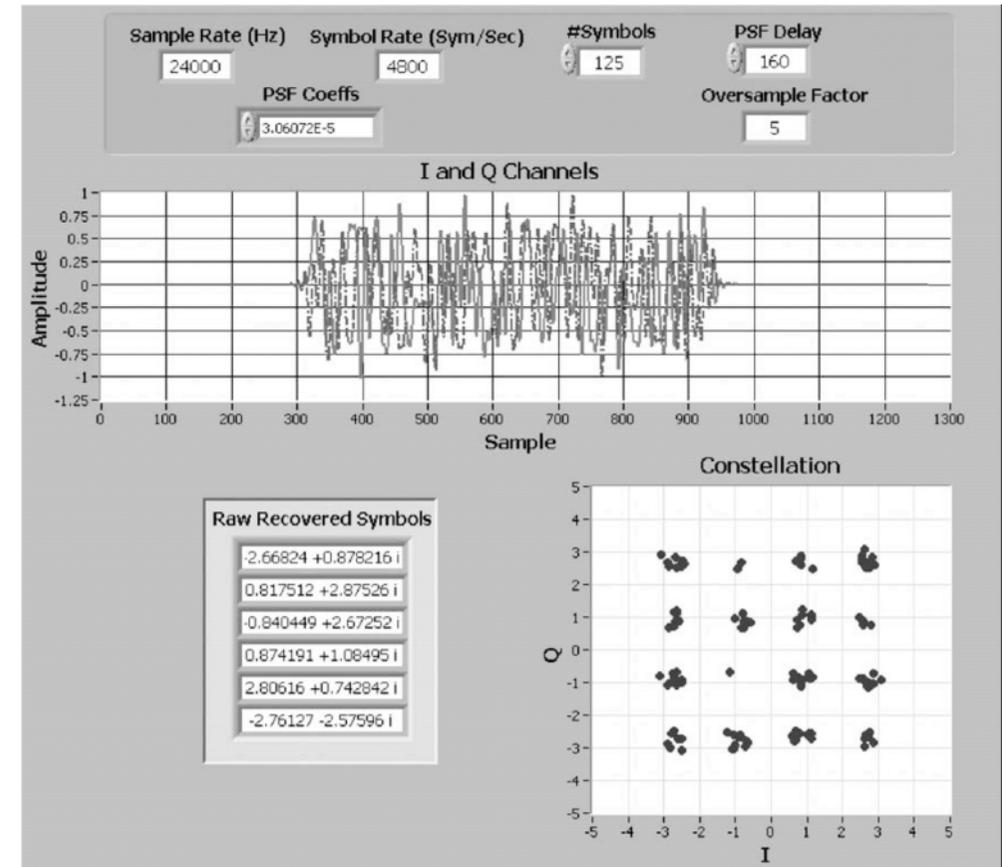


Рис. 10.8. Передняя панель ВП Demodulator.vi



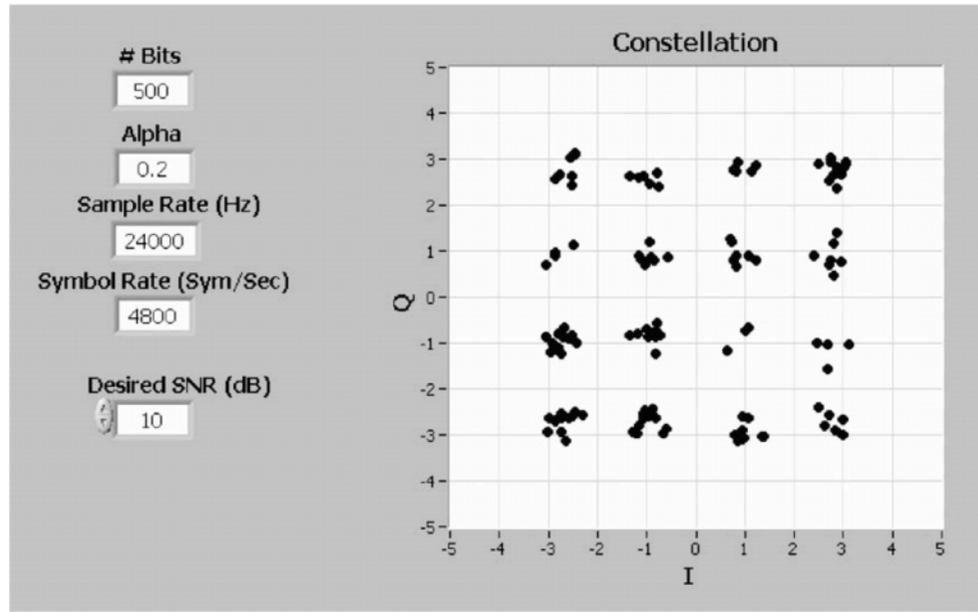


Рис. 10.10. Передняя панель ВП SimpleNoisySystem.vi

с долей этой шумовой мощности так, чтобы суммирование отдельных компонентов сигнала привело к установлению заданной шумовой мощности.

Рэлеевские замирания – это усредненная модель колебания сигнала, поэтому огибающая с замираниями заменяется средним ее значением, вычисленным на основе отсчетов входного сигнала. Наконец, можно обратиться в ВП, который обеспечивает формирование огибающей сигнала с замираниями, как показано на рис. 10.11.

Так как же воздействуют рэлеевские замирания на полученную форму колебания? Надо проанализировать процедуру ввода замирания при использовании ВП Channel.vi, представленного на рис. 10.11. Уже были описаны все блоки на блок-диаграмме этого ВП, кроме датчика символа, помещенного после демодулятора. Этот ВП будет рассмотрен более подробно в подразд. 10.4.2. Его работа заключается в определении вида символа с учетом заранее заданных границ символа.

Передняя панель ВП этой простой модели канала с замираниями показана на рис. 10.13. Величины ошибок в передаче символа приведены в нижней части рисунка. Для ОСШ в 40 дБ имеется единственная ошибка передачи символа, равная 2, в символе 11. В разд. 11 можно будет видеть, как исправляют эти ошибки в передаче символа.

В этом подразделе рассмотрен только статический случай линии связи без учета взаимного перемещения в пространстве передатчика и приемника. Конечно, учитывая сегодняшнее повсеместное использование беспроводных устройств связи, относительным движением передатчика и приемника невозможно пренеб-

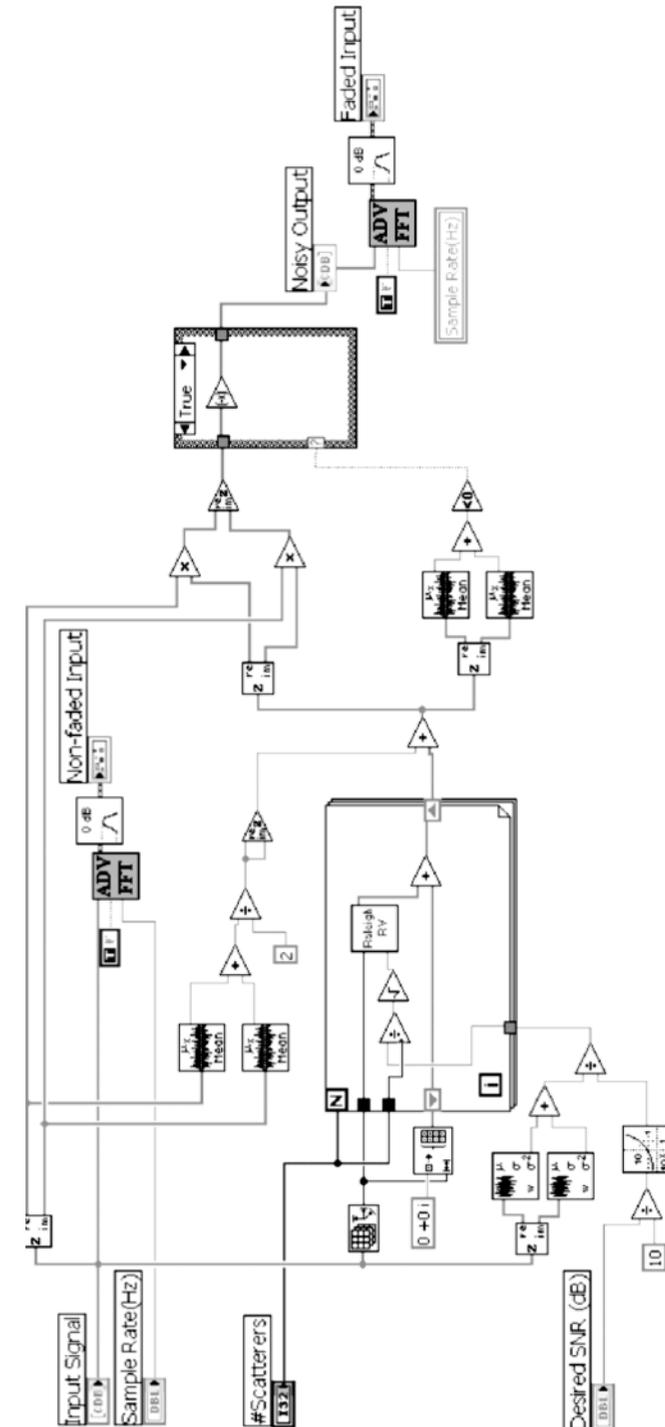


Рис. 10.11. Блок-диаграмма ВП Channel.vi

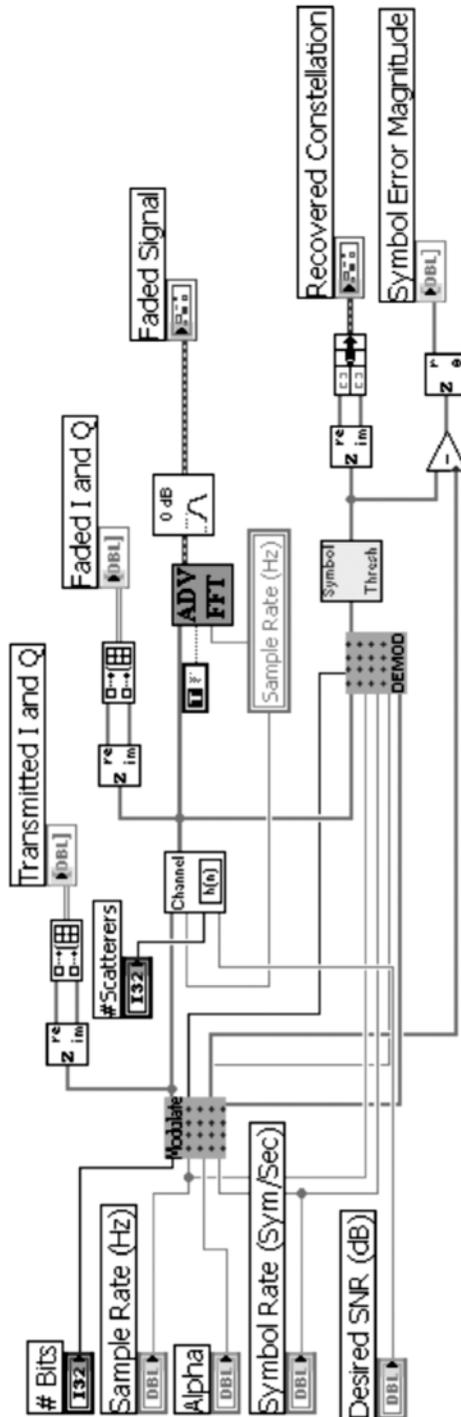


Рис. 10.12. Блок-диаграмма ВП FadeSimulation.vi

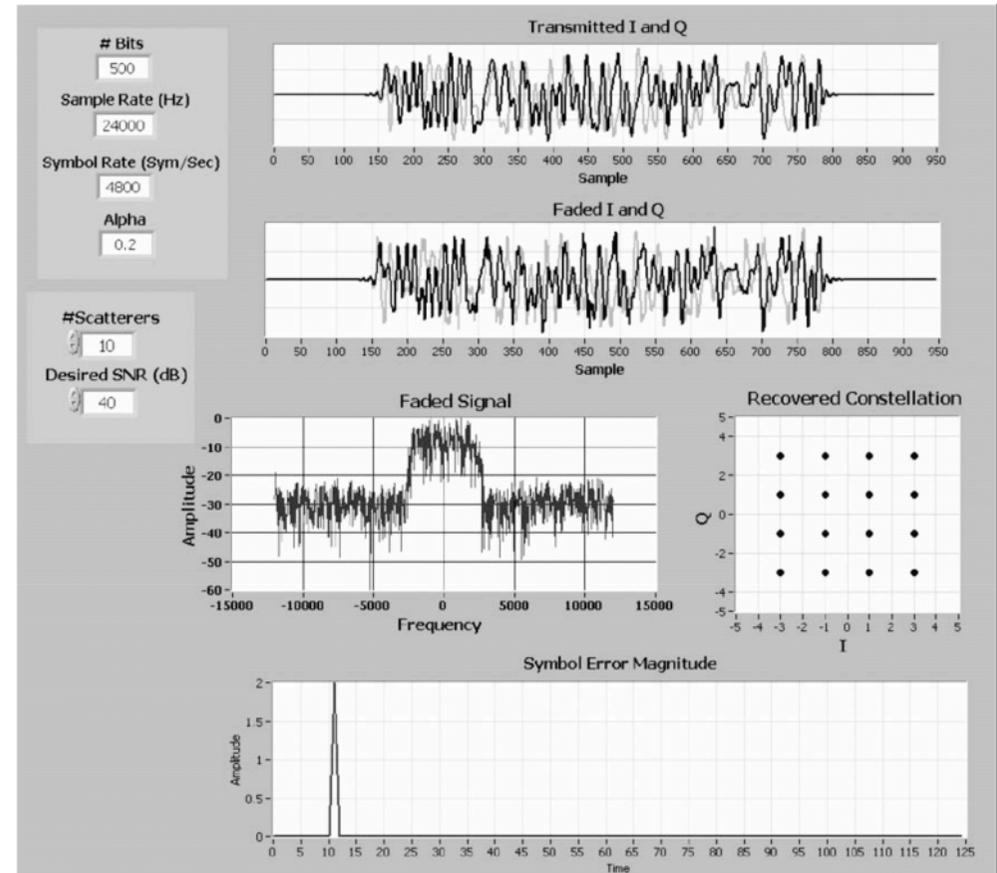


Рис. 10.13. Передняя панель ВП FadeSimulation.vi

регать. Однако можно легко учесть воздействие на передаваемый сигнал любого относительного движения между основной станцией и мобильной единицей с помощью доплеровского приращения частоты полученного сигнала, заданного следующим выражением [13]:

$$\omega_{di} = \frac{\omega_c v}{c} \cos \psi_i, \quad (10.1)$$

где  $v$  – скорость мобильной единицы;  $\omega_c$  – несущая частота РФ;  $c$  – скорость света;  $\psi_i$  – равномерно распределенные начальные фазы полученной отраженной волны.

Эффект Доплера может быть учтен и в выражении для комплексной огибающей  $g(t)$ :

$$g_D(t) = g(t)e^{j\omega_{di}t}. \quad (10.2)$$

Исследование влияния эффекта Доплера на цифровую форму модулированного сигнала выходит за рамки данного учебного пособия. Для проведения этих

исследований равномерно распределенные фазы колебаний легко устанавливаются, так же как и несущая частота, и радиальная скорость, которые могут быть заданы в эксперименте по изучению канала связи.

Однако эта модель канала ни в коем случае не может считаться полной. Исследователи, занимаясь изучением только мелкомасштабных замираний рэлеевской огибающей, пренебрегают эффектами потерь распространения сигнала, поэтому на сегодняшний день нет никаких инструментов для учета компоненты сигнала, поступающего по линии прямой видимости. Однако выше было показано, как можно построить модель канала с некоторыми довольно простыми стандартными блоками в LabVIEW.

## 10.4. Обнаружение и восстановление сигнала

После предварительного изучения отдельных узлов системы связи можно заняться основной ее функцией – восстановлением формы переданного колебания. Были исследованы помехи от соседних символов (ISI – Intersymbol Interference), выполнено моделирование рэлеевских замираний в канале связи и даже добавлено к сигналу немного гауссовского шума. Можно было бы предположить, что сигнал не может быть восстановлен после таких искажений, однако это не так: сигнал все еще присутствует где-то в отсчетах выборки.

На первый взгляд, на выходе канала должна быть фильтрация сигнала на фоне шума и любой интерференции, которые могут иметь место при распространении сигнала в канале. Необходимо осторожно выбирать полосу пропускания фильтра, чтобы она не была слишком узкой, потому что ошибка по частоте (любая, если она есть) может только «вытолкнуть» сигнал за пределы полосы пропускания фильтра.

Конечно, должны использоваться некоторые операции на входе приемника, чтобы можно было настроиться на частоту колебаний в основной полосе частот, с которой имеют дело в этом случае. Эти операции включают аналого-цифровое (A/D) преобразование, смещение частоты входного сигнала к промежуточной частоте (IF), преобразование частоты сигнала к заданной для обработки полосе частот и, возможно, несколько этапов передискретизации. Детали A/D-преобразования были рассмотрены в разд. 3. Если необходимо, чтобы выполнялось преобразование «вниз» по частоте, в результате которого частота колебаний соответствовала бы некоторой другой частоте (кроме нулевой), то можно использовать ВП ComplexMixer.vi.

### 10.4.1. Обнаружение сигнала и согласованная фильтрация

Название «согласованный фильтр» может ввести в некоторое заблуждение. Это не фильтр в традиционном смысле. Обычный фильтр предназначен только для удаления некоторой части спектра сигнала, которая является нежелательной, а оставшаяся часть спектра передается насколько возможно неизменной. Вообще говоря, цель применения фильтра состоит в том, чтобы сохранить желательный сигнал с наименьшими искажениями. На самом деле согласованный фильтр не выполняет ничего подобного. Фактически выходной сигнал такого фильтра

не является подобным по форме полученному в шумах сигналу [7, 19]. Цель применения согласованного фильтра состоит в том, чтобы просто вычислить показатель, который помогает решить, действительно ли присутствует нужный сигнал во входной смеси сигнала с шумом. Ниже можно увидеть, как согласованный фильтр может быть осуществлен в LabVIEW.

Предположим, что имеется зашумленное колебание (рис. 10.14), и нужно решить, присутствует ли в нём ЛЧМ-импульс с  $k = 2$  или  $k = 3$ .

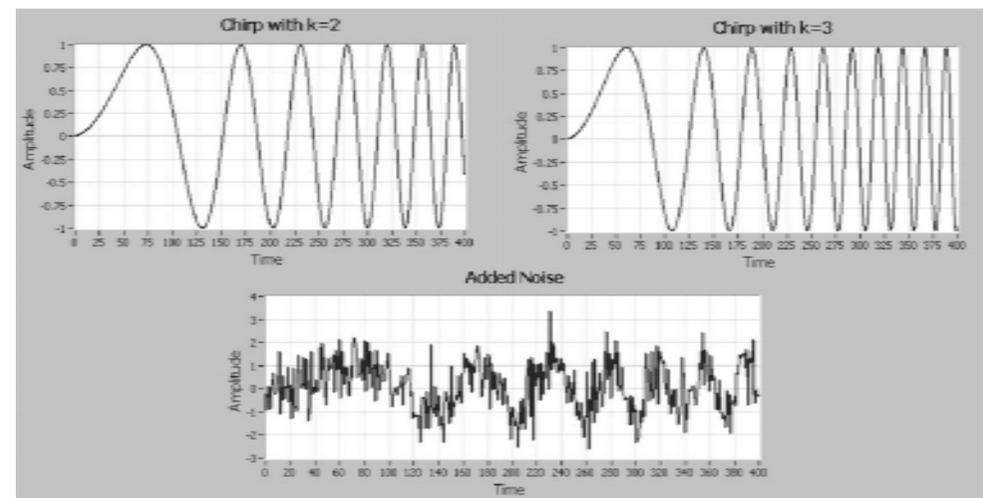


Рис. 10.14. Передняя панель ВП MatchedFilterDetection.vi

При некотором очень внимательном рассмотрении можно сказать, что это сигнал, который был получен на фоне шума. Вероятно, следует сопоставить пики в зашумленном сигнале с пиками в каждом из заданных сигналов и сделать вывод, какой из них является наиболее близким к полученному сигналу. Можно выполнить почти те же самые действия при согласованной фильтрации. При этом нужно знать заранее, что зашумленный сигнал может быть только одним из двух сигналов. Поэтому, если использовать эти два известных сигнала как шаблоны, можно определять корреляцию заданных шаблонов с зашумленными сигналами, и наиболее вероятным будет тот переданный сигнал, с которым будет самая высокая корреляция.

В блок-диаграмме, показанной на рис. 10.15, ЛЧМ-сигнал для  $k = 2$  искажен шумом в ВП AWGN. Если известен набор сигналов, который содержит только ЛЧМ-сигналы для  $k = 2$  и  $k = 3$ , то можно просто использовать эти два сигнала в качестве опорных для расчета взаимной корреляции между полученным сигналом и опорными. Результаты такого расчета показаны на рис. 10.16.

Из анализа выходного сигнала согласованного фильтра очевидно, что ЛЧМ-сигнал при  $k = 2$  есть нужный сигнал. Этот простой пример с двумя сигналами может быть легко распространен на случай 16-QAM-сигнала с некоторыми незначитель-

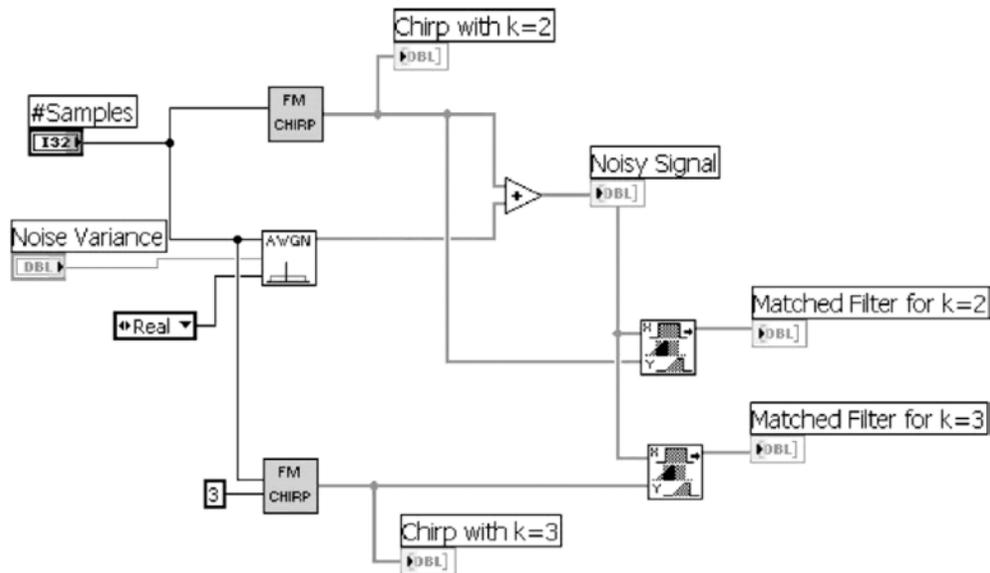
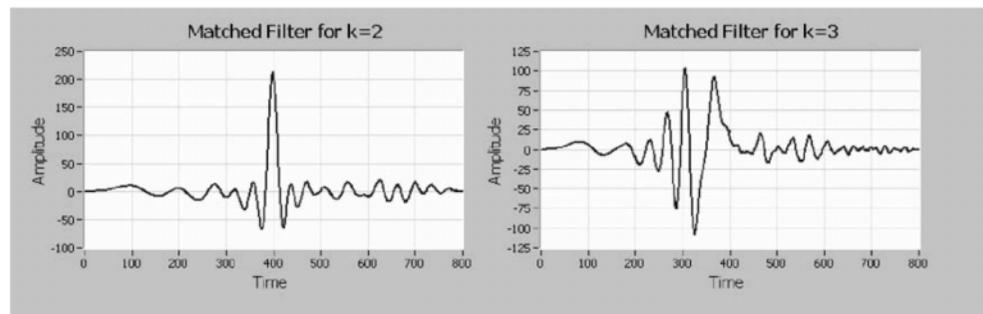


Рис. 10.15. Блок-диаграмма ВП MatchedFilterDetection.vi

Рис. 10.16. Выходной сигнал согласованного фильтра (Matched Filter) при  $k = 2$  и  $k = 3$ 

ными изменениями. При этом лучше нормализовать корреляцию, так как амплитуды совокупности точек в 16-QAM-сигнале изменяются. Указанное изменение амплитуды может быть причиной ошибки вычисления корреляции. Нормированная корреляция может быть вычислена по следующей формуле:

$$\text{Нормированная корреляция} = \frac{\sum xy}{\sum |x||y|} \quad (10.3)$$

## 10.4.2. Пороговые решения

Другой способ определения вида символа состоит в том, чтобы просто установить пороговые границы решения. В случае 16-QAM-сигнала любые его I- или Q-значения, например, равные +2, если принимаются по величине большими, чем +2, устанавливаются равными +3. Если эти значения находятся где-то между 0 и +2, то устанавливаются равными +1, и т. д. Можно рассмотреть простой декодер в LabVIEW, показанный на рис. 10.17. Это блок-диаграмма ВП Symbol\_threshold\_det.vi, которая уже была представлена на рис. 10.12 в ВП FadeSimulation.vi. Этот ВП содержит цикл обработки, основанный на необработанных принятых символах, и ряд селекторов для I- и Q-каналов. Можно видеть (см. рис. 10.12), что любое значение Q-символа есть +2; если это значение будет большим, то оно устанавливается равным +3.

## 10.5. Синхронизация

Синхронизация – чрезвычайно важный этап процесса восстановления модулированного сигнала в цифровой форме. В двух последующих подразделах рассматриваются важные аспекты синхронизации сигнала по времени и по частоте и то, как синхронизация обеспечивает возможность восстановления сигнала. Большинство цифровых стандартов уже включает оснащение для синхронизации по времени и по частоте, теперь можно будет увидеть для чего.

### 10.5.1. Синхронизация по времени

В предыдущих подразделах при анализе формы колебания не было никаких проблем с выбором начала отсчета времени. Было определено известно, в какой момент колебание начинается и с какого момента начнет применяться импульсная характеристика формирующего фильтра, что не может быть достигнуто вслепую так как возможна ситуация, когда приемник начнет обрабатывать входной сигнал во время его передачи.

Способен ли приемник в любое время восстанавливать символы, а если способен, то как он распознает, какой символ является началом передачи? Очевидно, что это огромная проблема в мире цифровых коммуникаций, потому что если неизвестно, какой символ был началом передачи, то нет никакого способа, с помощью которого остальная часть битов сигнала была бы должным образом восстановлена. Конечно, можно применить в решении проблемы синхронизации по времени то, что уже известно о согласованном фильтре.

Выходной сигнал согласованного фильтра максимален при наличии на входе заданного сигнала. Эта его особенность может быть использована для определения момента синхронизации по времени. Для этого можно использовать в начале передачи любой сигнал, который выбран заранее. Хорошо, если выбранная для сравнения копия сигнала не будет соответствовать ни одному из сигналов передачи данных, в противном случае получатся ложные сопоставления форм сигнала и копии и момент синхронизации будет выбран неверно. Так что необходимо ис-

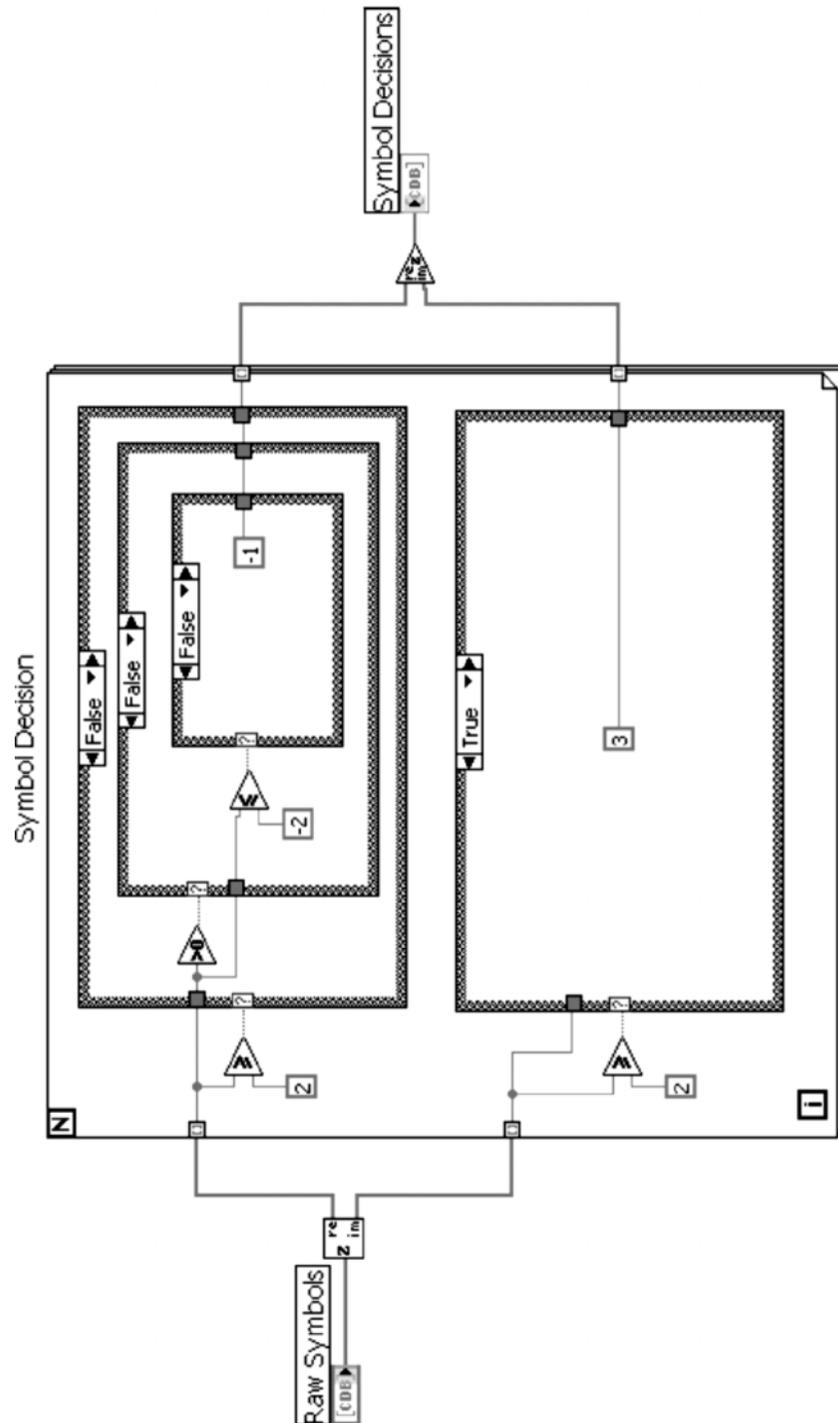


Рис. 10.17. Блок-диаграмма ВП Symbol\_threshold\_det. vi.

пользовать сигнал, который существенно отличается по форме от изменений во времени передаваемых с помощью сигнала символов и который не будет иметь почти никакой корреляции с любым из передаваемых символов. Этот сигнал следует добавлять в начале передачи группы символов. Тогда можно использовать в приемнике согласованный фильтр (согласованный с этим отличающимся сигналом). В момент, соответствующий максимальному значению отклика согласованного фильтра, отмечается положение вложенного символа синхронизации и таким образом зафиксировано начало передачи группы символов. Это тот случай, когда нормализованная корреляция в соответствии с выражением (10.3) будет чрезвычайно существенной, чтобы помочь избежать ложных определений момента синхронизации из-за больших выбросов амплитуды в полученной смеси сигнала и шума.

### 10.5.2. Синхронизация по частоте

Правильное определение частоты передаваемого сигнала столь же важно, как и синхронизация по времени, рассмотренная выше. Эффект от маленькой ошибки определения частоты, равной 10 Гц, для сигнальной совокупности показан на рис. 10.18.

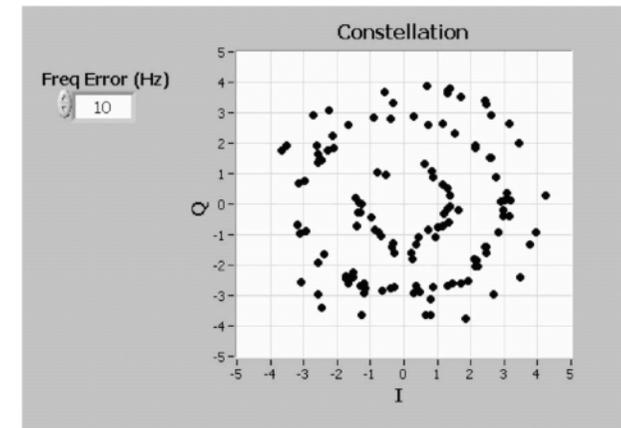


Рис. 10.18. Влияние ошибки по частоте на восстановленные символы

В демодулированной совокупности символов видны концентрические кольца. Так как единица частоты – радианы в секунду, любая ошибка по частоте обуславливает влияние вращения фазы на восстановленные символы, которое увеличивается с возрастанием времени, в результате наблюдается вращение символов со скоростью, пропорциональной ошибке по частоте.

Точно так же, как и в случае синхронизации по времени, чтобы оценить ошибку по частоте, можно включать определенные символы в известные на приемном конце линии моменты в передаваемую последовательность. Интервалы между этими символами позволят определить фазы (в противоположность случайным

символам данных, которые имеют случайные фазы), и данные фазы могут использоваться для вычисления ошибки их определения при восстановлении символов. Моменты времени, в которых эти символы присутствуют в переданном сигнале, изменяются в зависимости от стандарта (GSM, APCO25 и некоторые другие), но обычно имеется несколько таких пилот-символов, распределенных по интервалу передачи данных. В результате накопления ошибки фазы в течение передачи производится с высокой точностью оценка ошибки по частоте.

## 10.6. Модуляция в NI Toolset

В этом подразделе стоит напомнить, что инструментарий NI Toolset весьма похож на инструментальную панель Matlab, однако компания National Instruments обеспечивает добавление некоторых функций в инструментарий цифровой связи.

Из разд. 6 известно, что NI имеет инструментарий для спектральных измерений (toolset), который можно охарактеризовать как пакет, ориентированный больше на обнаружение сигнала. С другой стороны, инструментарий модуляции содержит несколько очень полезных функций для того, чтобы измерять и анализировать сигналы связи, некоторые из которых будут кратко упомянуты в этом подразделе. В настоящее время данный инструментарий доступен для пользователя только при использовании генератора сигнала PXI-5670, но, возможно, в будущем этот инструментарий будет доступен и без него. Типичные для LabVIEW панели этого инструментария показаны на рис. 10.19.

Инструментарий для модуляции поставляется с аналоговыми и цифровыми инструментами модуляции, а также с некоторыми сервисными функциями, чтобы выполнять вспомогательные действия, которые производятся вместе с цифровыми коммуникациями. Здесь будут только упомянуты некоторые из этих функций, чтобы дать читателю представление о том, каковы возможности инструментария (за дополнительной информацией можно обратиться к сайту [www.ni.com](http://www.ni.com)).

Можно начать рассмотрение NI Toolset с аналоговых инструментов модуляции. Четыре аналоговые функции, которые включены в инструментарий, показаны на рис. 10.20. Они обеспечивают модуляцию колебаний по амплитуде, частоте и фазе; и есть также один ВП, который позволяет преобразовывать входной сигнал при повышении частоты в пределах основной полосы частот.



Рис. 10.19. Вид палитры инструментария модуляции

### Analog Modulation Tools

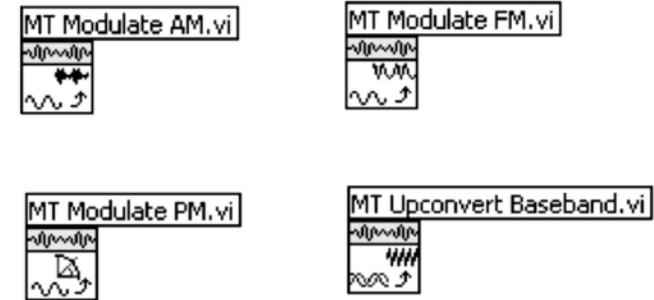


Рис. 10.20. Аналоговые инструменты инструментария модуляции

Несколько из наиболее интересных цифровых инструментов модуляции показаны на рис. 10.21. С помощью этих функции может быть сформирован QAM-сигнал, добавлен AWGN-шум, использованы замирания и даже выполнена фракционная передискретизация выборки. Не показаны на рис. 10.20 и 10.21 некоторые другие функции для модуляции и демодуляции FSK- и MSK-сигналов.

Из цифровых функций модуляции (см. рис. 10.21) ВП MT Fraction Resample.vi выполняет эффективную передискретизацию и операцию фильтрации. Пользователь должен только ввести форму колебания, которое будет повторно передискретизировано на текущую частоту дискретизации и желательную частоту дискретизации.

### Digital Modulation Tools

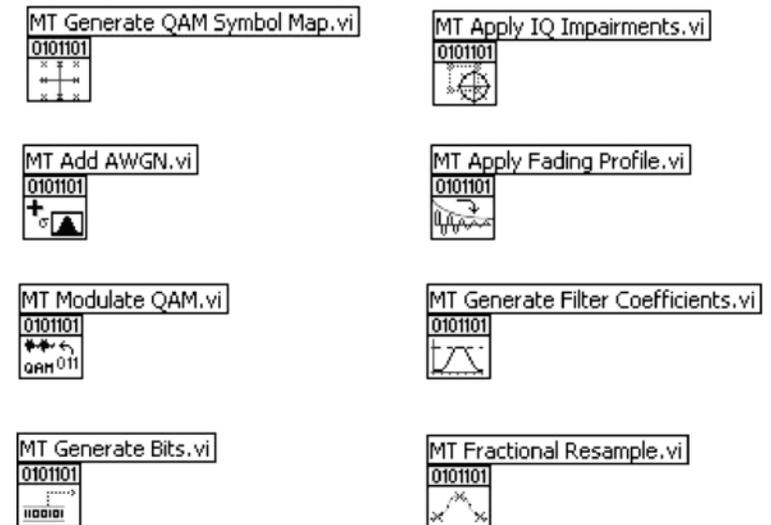


Рис. 10.21. Цифровой инструментарий модуляции (Digital Modulation Tools) (MT)

Основная цель корпорации NI при разработке описанного инструментария состоит в том, чтобы расширить использование PXI-аппаратных средств для формирования радиосигналов, некоторые из которых были упомянуты в разд. 3. В этом же подразделе не приведены детальные сведения относительно использования любого из рассмотренных ВП, но информация, приведенная в данном пособии, поможет читателю понять то, что выполняется каждым из этих ВП. Help NI также позволит получить достаточное представление, для чего эти ВП были предназначены.

## Резюме

В разд. 10 показано применение всего рассмотренного в предыдущих разделах в области цифровой системы связи. Вначале были рассмотрены модуляторы и демодуляторы, затем искажения в канале связи, обусловленные совокупностью белого гауссовского шума и рэлеевских замираний, и, наконец, было описано влияние этих искажений на восстановленные символы. Затем были исследованы некоторые методы определения вида символа.

Была также отмечена важность синхронизации и по времени, и по частоте.

Наконец, в конце раздела был приведен краткий обзор инструментария модуляции NI.

# Оптимизация обработки сигналов в LabVIEW



Следует заметить, что большинство алгоритмов обработки длительных сигналов, созданных в LabVIEW, довольно «медленны». Имеются непроизводительные расходы операционной системы персонального компьютера (PC), связанные с конфигурацией LabVIEW, обновлением фронтальной панели на дисплее, а также с фильтрацией и обработкой данных, которые определяются демодуляцией входного сигнала. Основной способ уменьшить *огромные* накопления в памяти при обработке сигнала состоит в том, чтобы понизить частоту дискретизации поступающего сигнала. Конечно, это не всегда выполнимо, поэтому в данном разделе приводятся некоторые общие подсказки для того, чтобы упростить кодирование в LabVIEW, а также некоторые лаконичные приемы для повышения скорости обработки сигнала.

## 11.1. Общие руководящие принципы кодирования в LabVIEW

В большинстве сред программирования есть некоторые простые средства, которые можно использовать для максимизации эффективности кодов. Приведенные ниже рекомендации не являются исчерпывающими, но содержат некоторые приемы [10], которые упрощают программы и экономят время их работы при небольших затратах времени на разработку.

1. *Часы, которые размещаются в цикле.* Повторные вычисления должны всегда выполняться вне цикла. Это должно быть стандартной практикой для любого языка программирования, включая LabVIEW. Необходимо исключить соблазн разместить в цикле связанные операции (или запросы блоков ВП), если они не зависят от цикла или должны обновляться при каждом повторе.

Можно обратиться к рис. 11.1, на котором показана простая блок-диаграмма ВП свертки.

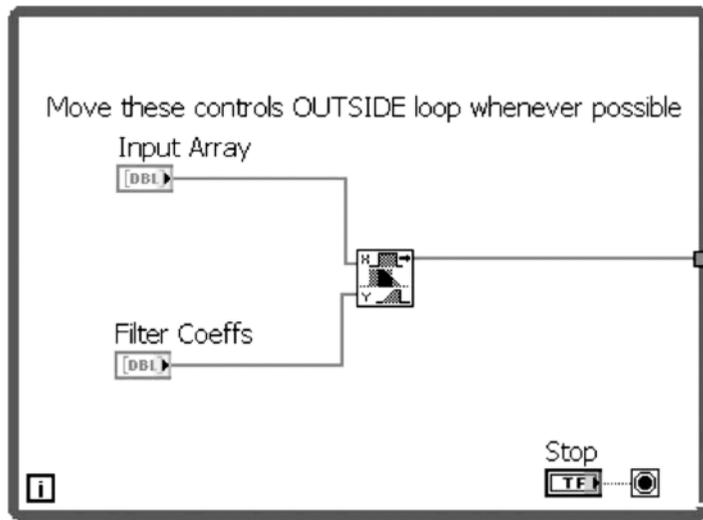


Рис. 11.1. Блок-диаграмма ВП свертки

Оба входных органа управления помещены на рис. 11.1 в цикл блок-диаграммы. Это означает, что при каждом повторении цикла будут просмотрены входные отсчеты. Так как коэффициенты фильтра, возможно, не будут изменяться, то это управление должно быть помещено вне цикла. Если входные отсчеты, как ожидается, изменяются при каждом повторении цикла, то это управление должно остаться в цикле.

2. *Предварительные вычисления должны использоваться в максимальной возможной степени.* Это относится к расчету коэффициентов фильтра, комплексных величин преобразователя или любому другому набору величин, которые не будут изменяться в процессе выполнения.

Нет, конечно, никакого смысла в пересчете набора коэффициентов фильтра или в использовании БПФ (FFT) для фиксированной последовательности отсчетов.

3. *Необходимо избегать применения глобальных и локальных переменных.* Надо использовать местные последовательности вместо местных переменных и пытаться избегать глобальных переменных в критически быстрых ситуациях.

Нарушение этой рекомендации приводит к большим перебоям в обработке сигнала из-за стремления сделать удобочитаемыми блок-диаграммы за счет использования глобальных и местных переменных. Виртуальные приборы сами присваивают обозначения переменным, что немедленно сказывается на продолжительности обработки сигналов, когда ВП принимают или записывают данные. И делается это для того, чтобы в блок-диаграмме не использовались соединительные провода, однако такие ВП чрезвычайно «медленны». Намного более эффективно использовать взамен глобальных локальные переменные. Известно [10], что каждая глобальная переменная в LabVIEW сохраняется отдельно в памяти. Если используются большие множества, определенно нет необходимости отсылать их к глобальным или местным переменным.

4. *Желательно минимизировать число обновлений дисплея и количество данных на фронтальной панели.* Графика LabVIEW является развитой графикой, и поэтому ее удобно использовать интенсивно, однако вывод данных для показа на экран является *медленным* процессом. Если график не является необходимым для использования, желательно его не выводить на экран. Следует стремиться свести информацию на дисплее к минимуму, чтобы увеличить скорость выполнения вычислительных операций.

5. *Необходимо использовать встроенные ВП при любой возможности.* Если хорошо разобраться в большинстве встроенных в LabVIEW функций, можно заметить, что если основной ВП не является наилучшим, то выгоднее вместо него использовать типовую программу на языке С.

Такие обращения к типовым программам на языке С приведут к существенному повышению скорости вычислений по сравнению с использованием эквивалентного ВП в LabVIEW. В этом есть определенный смысл, однако можно соблазниться и включить в блок-диаграмму собственный ВП, чтобы выполнить некоторую функцию, но сначала необходимо проверить, имеет ли LabVIEW такой же ВП.

6. *Необходимо разрабатывать исполняемые приложения.* Пакет LabVIEW имеет способность встраивать ВП в исполняемое приложение. Это добавление к программному обеспечению известно как «разработчик приложения», что позволяет любому пользователю *управлять* приложением, не имея копии LabVIEW. Исходная программа должна расходовать время вычислений на обращение к пакету LabVIEW. В то же время, даже если с помощью этого приема и не ускорится работа приложения, зато уменьшатся требуемые объемы памяти, так как полный пакет LabVIEW не загружен, что, таким образом, даст PC некоторый запас памяти для более быстрого выполнения операций. Конечно, если PC очень мощный и с большим объемом памяти, то можно не заметить в этом случае больших преимуществ. Однако все-таки стоит делать такие попытки, а преимущество в разработке исполняемых приложений заключается в том, что никто посторонний не может увидеть или изменить созданные блок-диаграммы.

## 11.2. Подсказки в обработке сигналов

Кроме использования упомянутых выше подсказок для того, чтобы обеспечить высокую скорость обработки сигнала, есть несколько способов увеличения скорости операций в обычном цифровом сигнальном процессоре (DSP).

Имеется две операции, которые заслуживают большего внимания, чем другие: фильтрация большого входного массива и вычисление БПФ. Далее в тексте приводятся некоторые краткие подсказки для того, чтобы эффективно выполнять операции в LabVIEW.

### 11.2.1. Линейная свертка на основе БПФ

Можно вспомнить, что свертка в области времени соответствует умножению в области частоты. Это означает, что можно выполнить фильтрацию в области частоты, умножая результат дискретного преобразования Фурье коэффициен-

тов фильтра на результат применения ДПФ к сигналу. Теперь предположим, что входной сигнал – довольно длинная последовательность, и тогда использование БПФ для выполнения ДПФ приведет к существенной экономии времени преобразования. Прежде чем рассмотреть выполнение этой операции в LabVIEW, необходимо рассмотреть ограничения на использование ДПФ для линейной свертки.

Линейная свертка основана на использовании входных выборок длиной  $L$  и  $N$ , и с ее помощью производят выходную выборку длиной  $L + N - 1$  [4]. Чтобы использовать БПФ для получения выходной выборки такой же длины, достаточно просто добавить число входных выборок, чтобы получить число по основанию 2.

Блок-диаграмма на рис. 11.2 показывает, что размеры выборок входного сигнала и число коэффициентов фильтра дополняются до следующего числа по основанию 2. Выбор числа отсчетов выборки по основанию 2 обеспечивается увеличением числа отсчетов как во множестве коэффициентов фильтра, так и во входном множестве. Выходные отсчеты двух БПФ перемножаются, и производится обратное БПФ. Для четных чисел больших выборок БПФ коэффициентов фильтра должно быть выполнено только однажды, в то время как БПФ входных данных выполняется на каждом интервале выборки.

### 11.2.2. Реальное БПФ

БПФ выполняется довольно быстро, но оно может производиться быстрее, если четные и нечетные компоненты реального сигнала упаковывать в виде реальных и мнимых (соответственно) частей для применения входного множества комплексного БПФ. Когда это сделано,  $N$ -точечное реальное БПФ может быть заменено на  $N/2$ -точечное комплексное БПФ [11]. На первом шаге надо разделить входную последовательность отсчетов на четные и нечетные индексированные компоненты. Затем повторно собирают эти две составляющие последовательности в одну комплексную последовательность.

Теперь можно выполнить  $N/2$ -точечное комплексное БПФ. Однако небольшую манипуляцию необходимо проделать с выходными отсчетами БПФ, а именно применить  $N/2$ -точечное БПФ, которое, как предполагается, представляет собой БПФ последовательности данных из  $N$  точек.

Использование свойств преобразования Фурье позволяет получить другую часть результата БПФ [11]. На рис. 11.3 показана блок-диаграмма ВП FastRealFFT.vi.

В ВП FastRealFFT.vi используются уравнения из [11] для того, чтобы разделить это комплексное обратное БПФ на компоненты спектра реального сигнала. При этом используется промежуточный дополнительный шаг, в котором отсчеты должны быть сохранены в памяти ПК, а затем выполняются операции разделения результатов комплексного обратного БПФ на четные и нечетные компоненты реального сигнала. Этот пример не был оптимизирован, но показывает, как комплексное БПФ может использоваться для реализации БПФ реального сигнала.

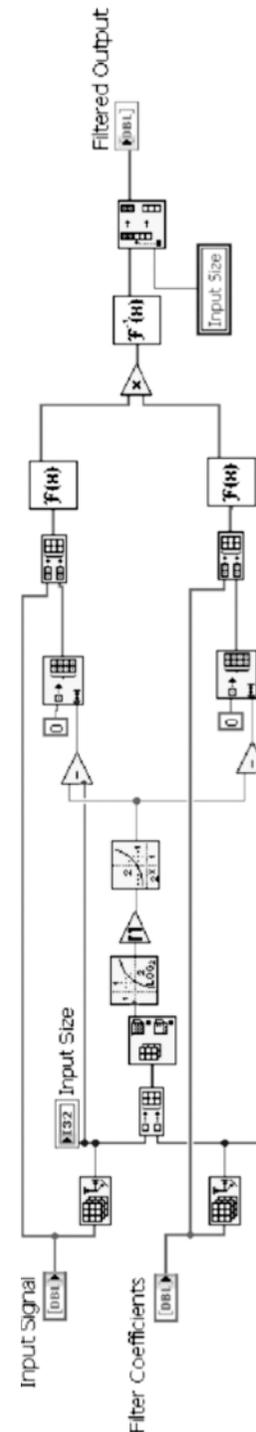


Рис. 11.2. Блок-диаграмма ВП Conv\_with\_FFT.vi

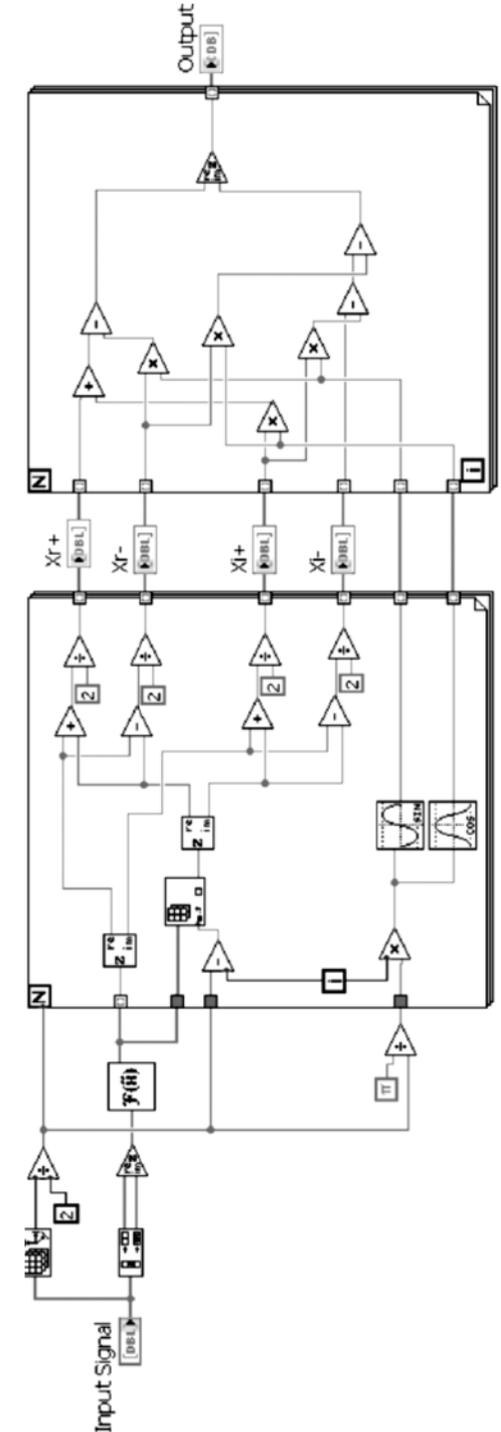


Рис. 11.3. Блок-диаграмма ВП FastRealFFT.vi

### 11.3. Дальнейшие применения цифровой обработки сигналов в LabVIEW

Инструменты и функции, которые были показаны ранее в данном разделе, – это только малая часть возможностей LabVIEW. LabVIEW имеет палитры, полные математических функций. Например, имеется доступ к функциям для аппроксимации кривых, различным операциям вычисления (включая решение дифференциальных уравнений, нахождение корней, приближение функций), а также к специальным функциям, подобным Бесселевой функции и дополнительной функции ошибки. Расположение этих функций в палитре показано на рис. 11.4.

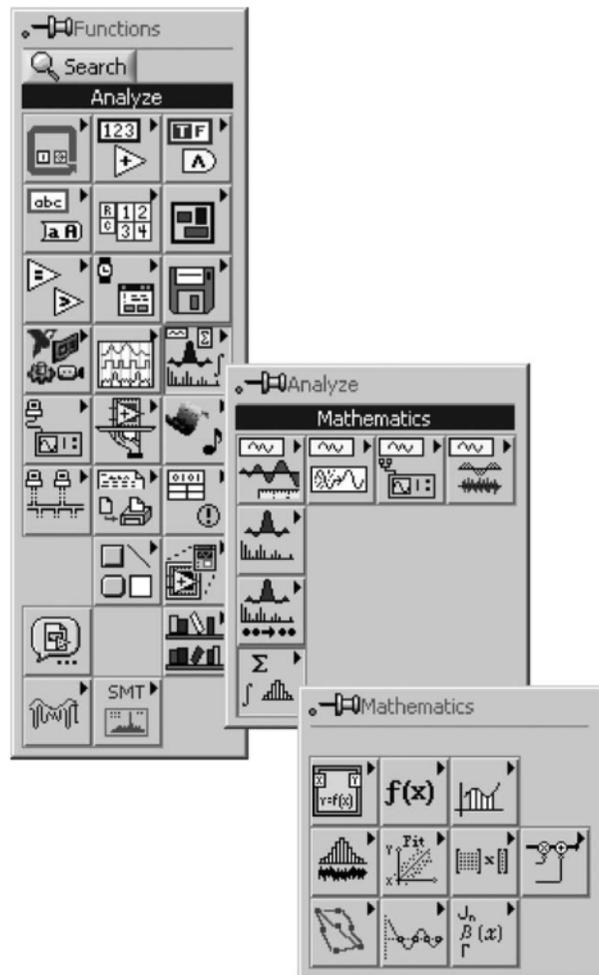


Рис. 11.4. Расположение функций в палитре математики в LabVIEW

Некоторые из этих функций будут более полезны, чем другие, в частности определение нулей математической функции и линейные функции алгебры. Подпалитры перечисленных функций показаны на рис. 11.5 и 11.6. В следующих двух подразделах используются некоторые из этих функций.

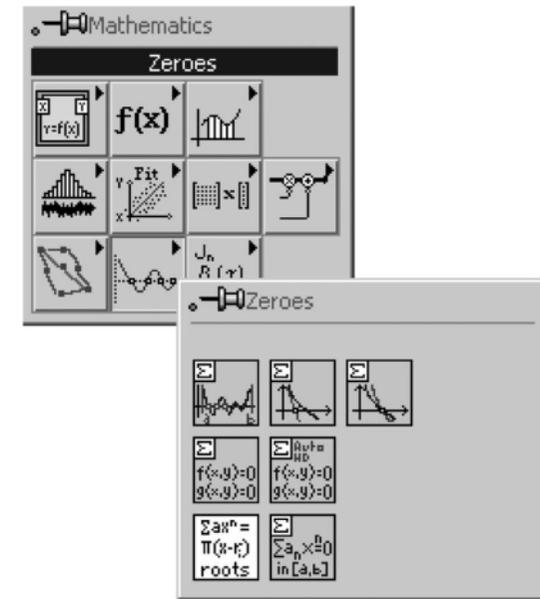


Рис. 11.5. Местоположение нулей функции в LabVIEW

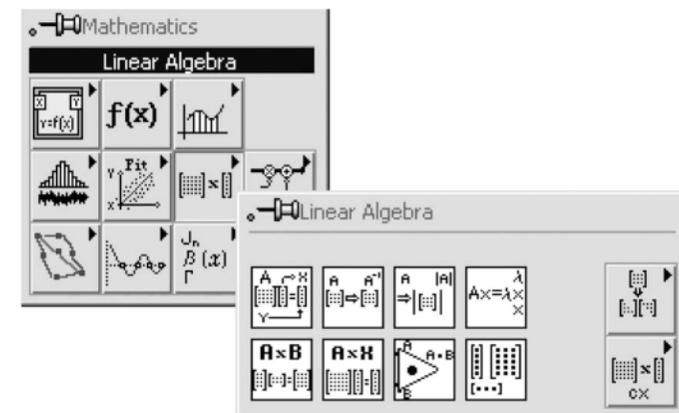


Рис. 11.6. Местоположение в LabVIEW линейных алгебраических функций

### 11.3.1. Корни дифференциального уравнения

LabVIEW имеет встроенную функцию для нахождения комплексных корней полинома.

Эта функция может быть чрезвычайно полезной, когда необходимо исследовать корни дифференциального уравнения, которое может описать функцию передачи фильтра. На рис. 11.7 показана блок-диаграмма ВП PoleZero.vi, в котором используется встроенный ВП Complex Polynomial Roots.vi, чтобы решить дифференциальное уравнение, введенное пользователем. На рис. 11.8 показана фронтальная панель ВП PoleZero. Здесь можно видеть, что коэффициенты дифференциального уравнения *должны* быть введены на спускающихся панелях z. График показывает поле расположения полюсов и нулей для данного уравнения и расположение их относительно круга единичного радиуса.

### 11.3.2. Линейный прогнозирующий шифровальщик речи

Во всех цифровых системах связи используется некоторая форма кодирования речи, чтобы уменьшить требуемую скорость передачи голосового сигнала в битах. Здесь будет рассмотрен один из наиболее общих алгоритмов кодирования речи – линейное прогнозирующее кодирование (LPC).

Работа LPC-вокодера базируется на предпосылке, что на коротком интервале времени речь может быть смоделирована как выходной сигнал линейного фильтра, периодически возбуждаемого отдельным импульсом [14]. Есть еще много тонкостей в практическом осуществлении этого вида кодирования речи. Для более подробного знакомства с LPC можно обратиться к [15], где есть глава, посвященная этой теме. В этом подразделе представлено выполнение в LabVIEW только центральной части LPC-вокодера – генератора коэффициентов.

Чтобы смоделировать короткое звучание, голосовой сигнал сегментируют как выходные сигналы линейных фильтров, и необходимо найти коэффициенты фильтров, которые наиболее близко соответствуют спектру каждого сегмента.

Такая задача может быть решена за счет применения критерия минимизации ошибки, который выражается следующим уравнением [15]:

$$\sum_{k=1}^p \alpha_k R_n(|i-k|) = R_n(i), \quad 1 \leq i \leq p, \quad (11.1)$$

где  $p$  – число полюсов в модели фильтра;  $R_n$  задается выражением [15]

$$R_n(k) = \sum_{m=0}^{N-1-k} s_n(m)s_n(m+k); \quad (11.2)$$

где  $s_n(m)$  – взвешенный сигнал речи.

Имеются очень эффективные методы для решения системы линейных уравнений, один из которых – рекурсия Levinson-Durbin. Но так как LabVIEW имеет не-

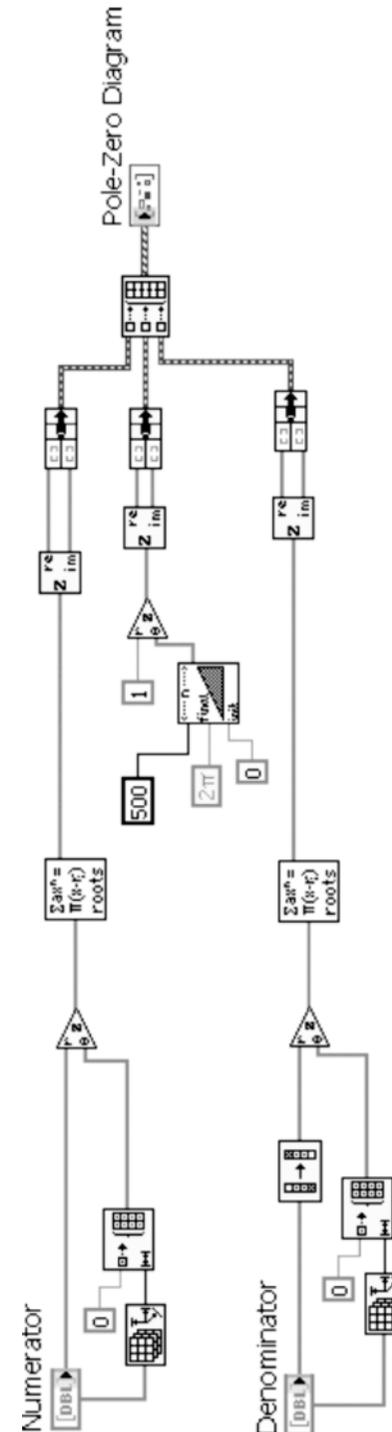


Рис. 11.7. Блок-диаграмма ВП PoleZero.vi

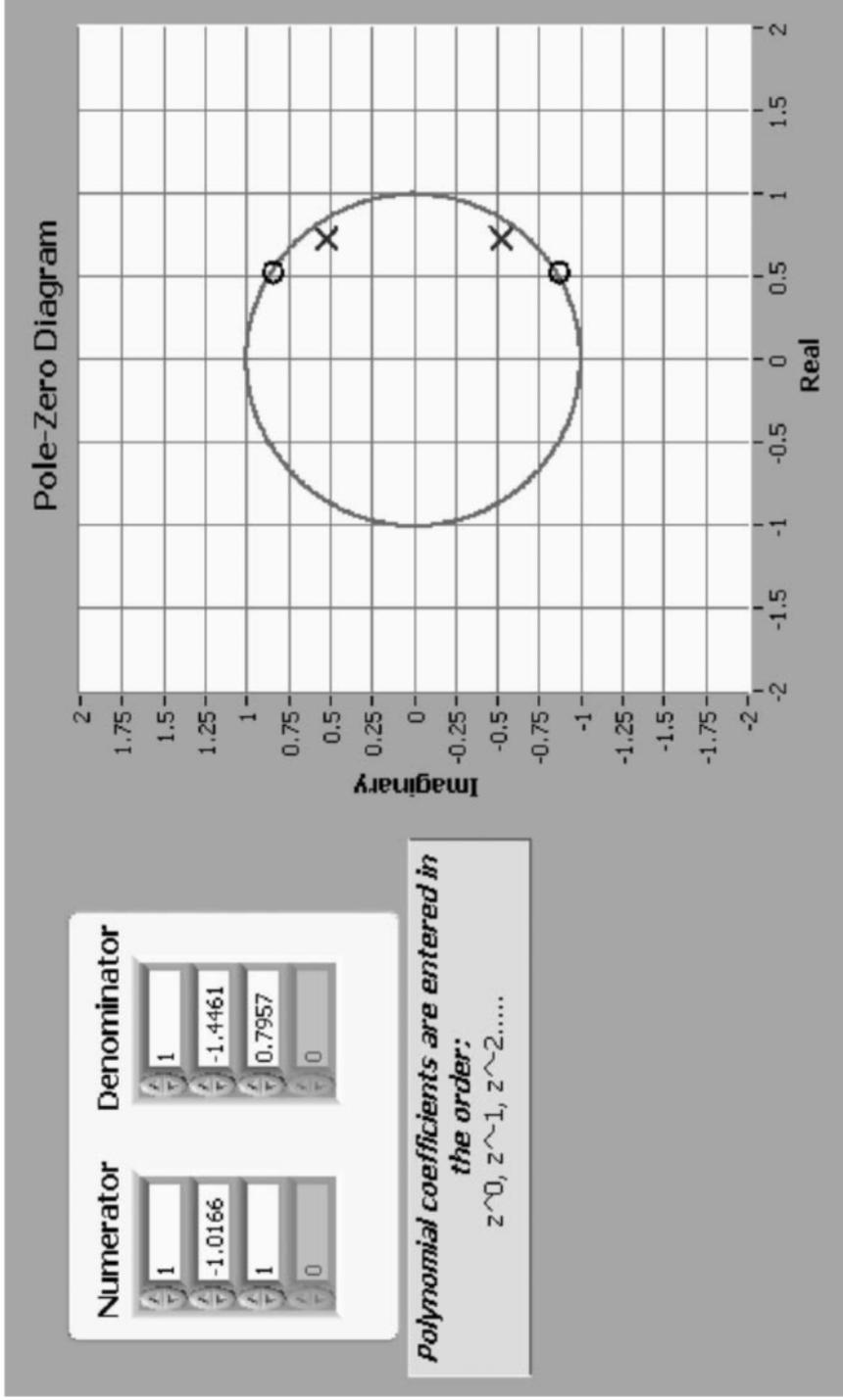


Рис. 11.8. Фронтальная панель ВП PoleZero.vi

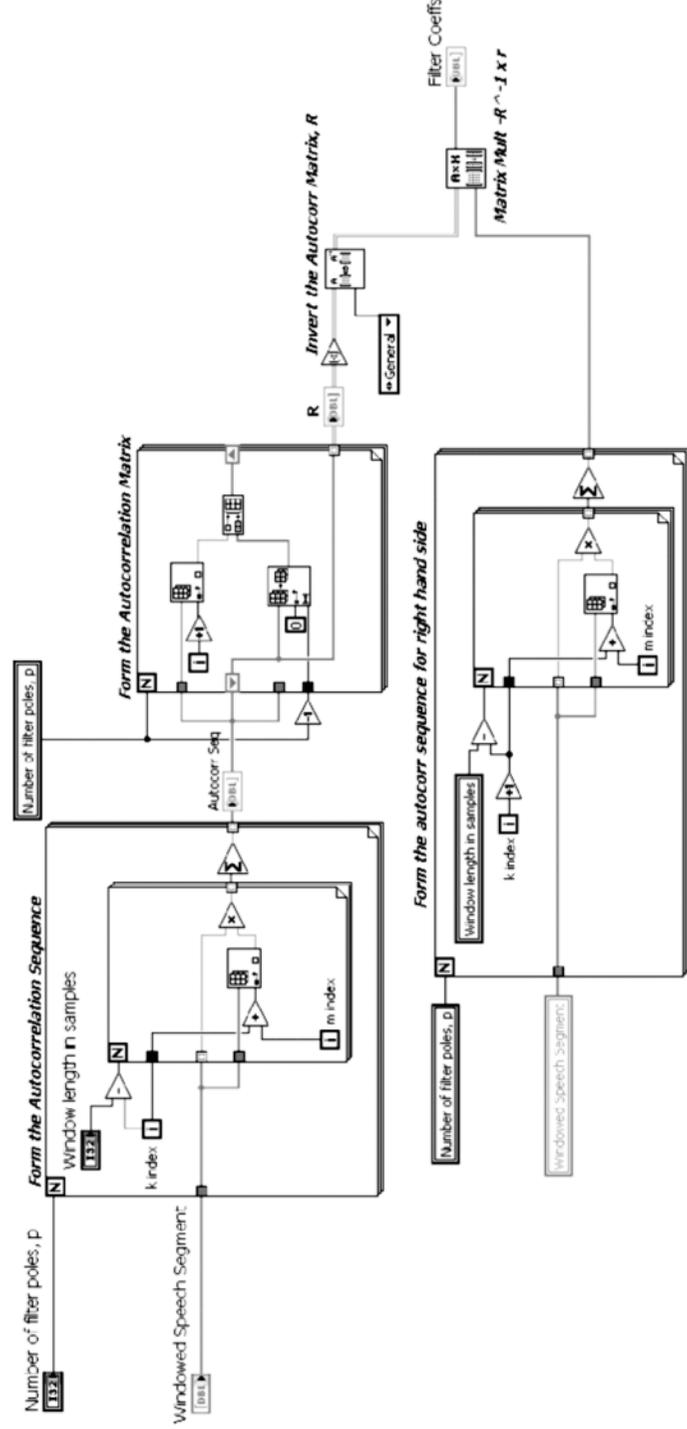


Рис. 11.9. Блок-диаграмма ВП LPCcoeffs.vi

которые встроенные инструменты манипуляции матрицами, можно только инвертировать матрицу автокорреляции, как показано в блок-диаграмме LPCcoeffs.vi на рис. 11.9.

На первом шаге работы ВП LPCcoeffs.vi должен вычислить за малое время значения функции автокорреляции  $R_n(k)$  по выражению (11.2). Как только будет получена последовательность автокорреляции, можно использовать эти значения, чтобы сформировать матрицу автокорреляции (которая является Теплицевой матрицей).

Тогда вызывают функцию инверсии матрицы в LabVIEW для инвертирования (символ отрицания) матрицы автокорреляции. В то же самое время формируется другая последовательность автокорреляции (на этот раз с задержкой от 1 до  $p$ ), и эта последовательность есть матрица, умноженная на инвертированную матрицу автокорреляции. Результат этого умножения – последовательность из  $p$  коэффициентов фильтра.

ВП LPCcoeffs.vi может быть затем включен в другой ВП, который будет принимать выборку аудиосигналов из звуковой карты (можно использовать ВП sndcard.vi, представленный в разд. 3), выполнит весовую обработку и далее использует коэффициенты фильтра из первого ВП, чтобы восстановить структуру фрагмента голоса (или преобразовать коэффициенты фильтра в коэффициенты отражения для реализации его структуры). Конечно, есть многочисленные LPC-вокодеры, применяемые для распознавания вокализованных и невокализованных звуков, передаточные функции фильтров и некоторый класс алгоритмов обнаружения высоты звука, однако необходимо начинать с использования ВП LPCcoeffs.vi.

## Резюме

В этом разделе представлены некоторые методы кодирования в базисе LabVIEW, которые можно использовать для увеличения скорости программ. Также показано несколько вычислительных приемов для некоторых самых общих вычислений в DSP. Наконец, завершено рассмотрение некоторых математических возможностей LabVIEW и показаны два применения DSP в LabVIEW, не связанные с цифровой связью.

# Сопряжение LabVIEW с другими программ- ными продуктами для обработки цифровых сигналов. Пакет LabVIEW Simulation Interface Toolkit



Пакет LabVIEW Simulation Interface Toolkit (набор инструментов для организации интерфейса моделирования) предназначен для осуществления тесного взаимодействия LabVIEW и Simulink, программного продукта компании MathWorks с целью повышения скорости разработки приложений.

Модуль Simulation Interface Toolkit представляет собой запатентованную технологию для отображения и управления моделями Simulink. Кроме того, он предоставляет возможность переносить модели в LabVIEW, что позволяет ввести в модель ввод/вывод сигналов в устройствах сбора данных компании National Instruments. Причем интерфейс пользователя остается неизменным. Таким образом, данный программный продукт включает в себя следующие функции:

- построение собственного интерфейса пользователя в LabVIEW для интерактивного тестирования моделей Simulink;
- импорт моделей Simulink в LabVIEW;
- добавление в модели Simulink функций ввода/вывода.

Используя перечисленные возможности, разработчики управляющих систем могут в кратчайшие сроки переходить от моделирования к тестированию алгоритмов на измерительном устройстве. Модель процесса совместного применения Simulink и LabVIEW представлена на рис. 12.1.

## Подключение LabVIEW к Simulink



Рис. 12.1. Модель процесса разработки системы с применением Simulink и LabVIEW

## 12.1. Тестирование моделей с использованием интерфейса LabVIEW

Модуль Simulation Interface Toolkit (SIT) обладает инструментами, необходимыми для создания собственного интерфейса для моделей Simulink, для этого предназначен встроенный Менеджер подключений (SIT Connection Manager). Использование собственного интерфейса пользователя для моделирования, анализа и тестирования моделей можно представить как процесс, состоящий из четырех основных шагов.

**Шаг 1.** Модуль SIT добавляет компонент NISink в набор компонентов Simulink, который можно поместить в рабочее окно модели, как показано на рис. 12.2.

**Шаг 2.** Необходимо создать собственный интерфейс, используя библиотеки встроенных в LabVIEW элементов управления и отображения, как, например, на рис.12.3.

**Шаг 3.** Используя SIT Connection Manager, окно которого показано на рис. 12.4, необходимо подключить элементы управления и отображения, размещенные на лицевой панели, к параметрам и узлам NISinks на блок-диаграмме Simulink.

**Шаг 4.** Запуск приложения и анализ работы модели.

На рис. 12.3 представлена лицевая панель ВП Pass\_Susp.vi, обеспечивающая управлением амплитуды и частоты отклика.

На рис. 12.5 представлено окно пользователя в процессе работы по отображению входного воздействия и отклика моделируемой цепи.

## 12.2. Дополнительные возможности по тестированию моделей

Менеджер подключений SIT Connection Manager может работать в сетевом режиме. Таким образом, существует возможность подключать интерфейс пользователя, созданный в LabVIEW, к моделям Simulink, запущенным на различных компьютерах в сети. Эта функция позволяет управлять работой всех моделей с одного компьютера и осуществлять их тестирование из одного и того же окна пользователя. На рис. 12.6 структура сетевого управления моделированием.

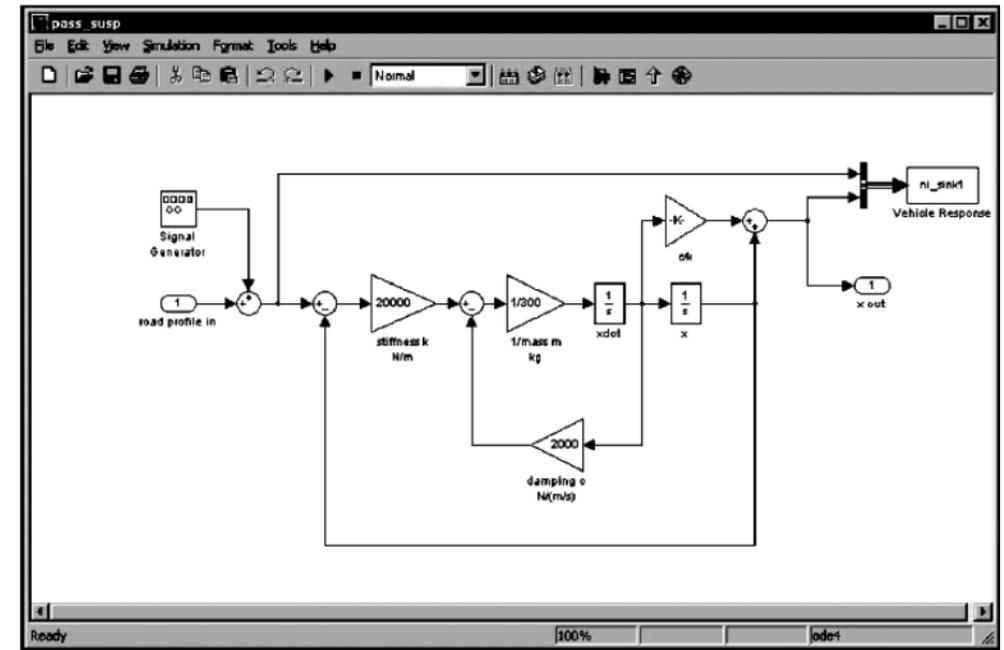


Рис. 12.2. Рабочее окно Simulink

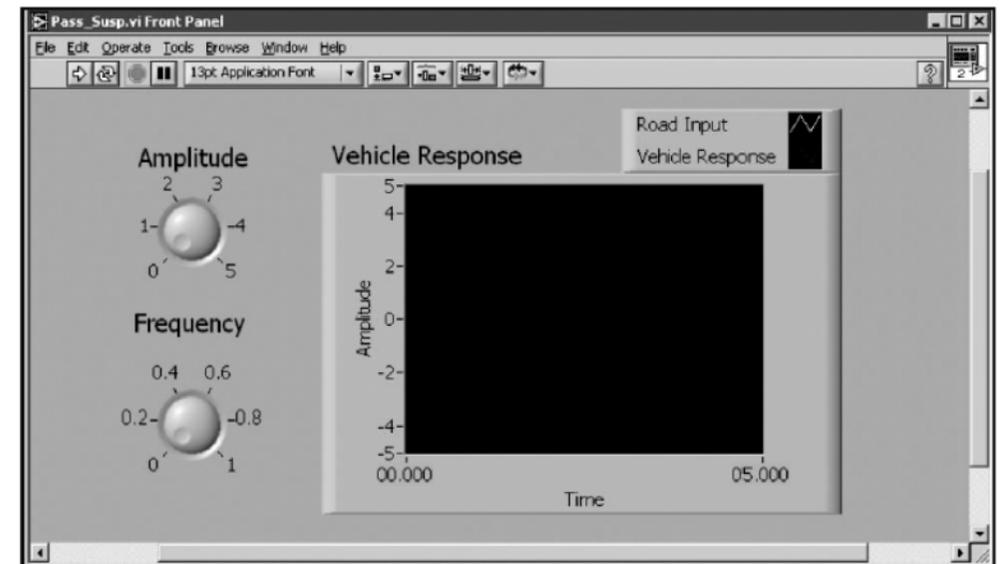


Рис. 12.3. Окно пользователя

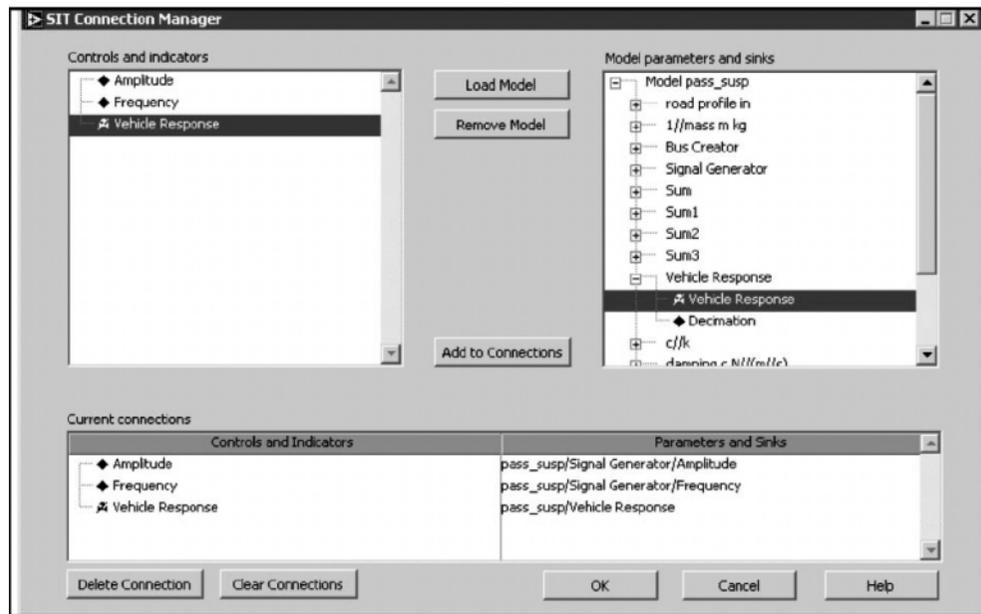


Рис. 12.4. Окно Менеджера подключений (SIT Connection Manager)

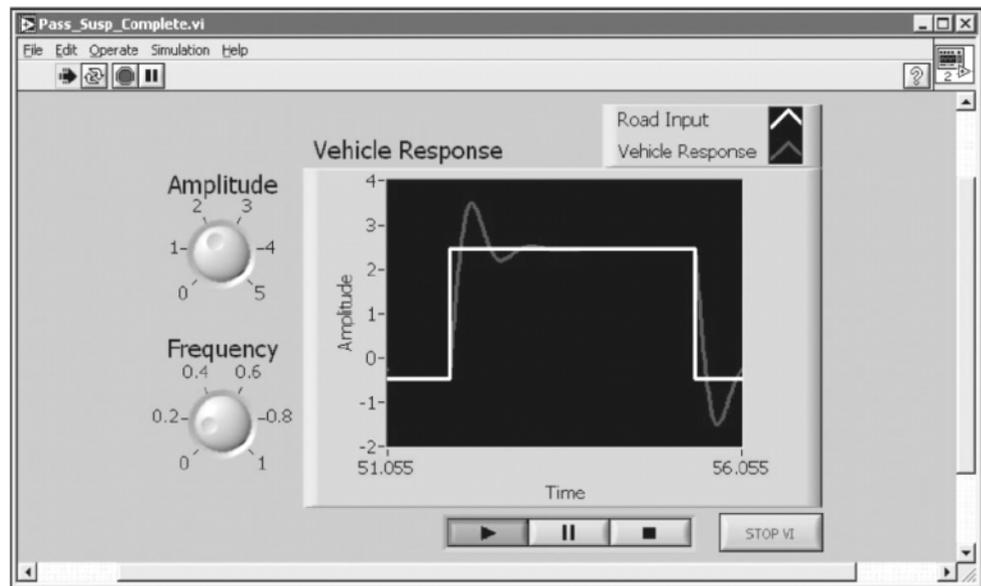


Рис. 12.5. Окно пользователя в процессе работы

Для автоматизации процесса тестирования, используя библиотеки SIT, можно создать симулятор, который будет автоматически запускать модель Simulink с различными характеристиками и записывать результаты. Кроме того, используя функциональные возможности LabVIEW, можно подавать на вход модели различные сигналы и анализировать результаты, что снижает необходимость ручного тестирования, требуемого на этапе проверки алгоритмов.

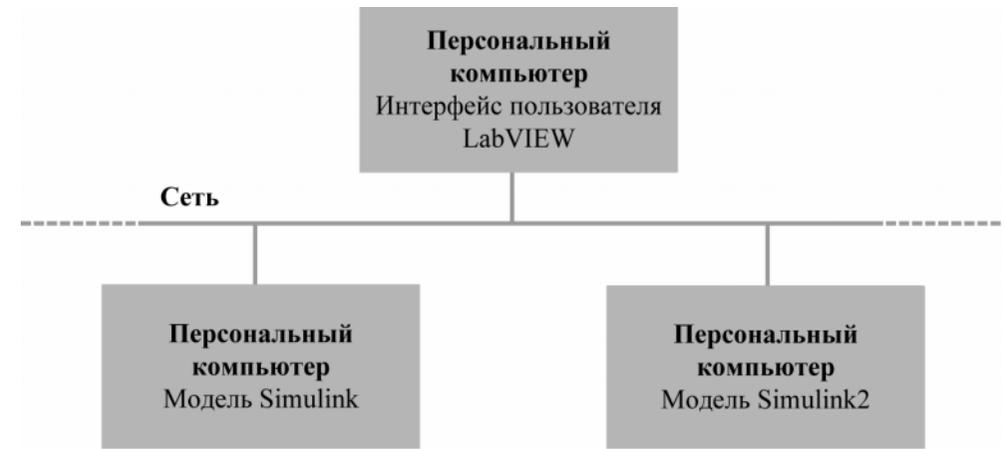


Рис. 12.6. Сетевая модель управления тестами

## 12.3. Импорт модели из Simulink в LabVIEW

С помощью пакета Simulation Interface Toolkit можно переносить модели в рабочую среду LabVIEW. SIT включает в себя встроенный компилятор для конвертирования моделей Simulink в файлы DLL и создает примеры подключений к ним в LabVIEW. Генерируемые примеры адаптированы под среду Simulink, и они позволяют наделять модели базовыми функциями по обмену данными с устройствами сбора данных с минимальными модификациями исходного кода.

## 12.4. Перенос модели в систему реального времени

Используя возможности Simulation Interface Toolkit, разработчики могут легко переходить от тестирования модели к работе в системах реального времени. Для этого необходимо выбрать в меню доступную систему, после чего необходимые для работы файлы автоматически будут загружены в устройство без потери функ-

циональности модели. В настоящее время существует множество систем реального времени, позволяющих загрузить в них модели Simulink. Это PXI-, CompactRIO- и CompactFieldPoint-устройства. Кроме того, можно использовать для работы в режиме реального времени платы PCI-7041/6040E, устанавливаемые непосредственно в слот PCI персонального компьютера.

# Тестирование и программирование цифровых сигнальных процессоров (DSP) в LabVIEW



## 13.1. Пакет LabVIEW DSP Test Toolkit

С помощью инструментов пакета для тестирования сигнальных процессоров (LabVIEW DSP Test Toolkit) разработчики легко могут проводить испытания и осуществлять отладку систем на базе DSP. Благодаря этому ошибки можно отслеживать непосредственно в процессе проектирования, кроме того, значительно сокращается время разработки системы.

Данный пакет предназначен для автоматизации TI's Code Composer Studio™ (CCS), интегрированной среды разработки управляющих команд компании Texas Instruments. Таким образом, инженеры получают возможность создавать тестирующие системы для программного кода DSP. Виртуальные приборы, включенные в библиотеку LabVIEW DSP Test Toolkit, позволяют также осуществлять обмен данными в реальном времени по технологии RTDX™ с платами, входящими в стартовый пакет для программирования сигнальных процессоров фирмы TI (DSP Starter Kit (DSK)).

### 13.1.1. Функции автоматизации CCS

Функции пакета LabVIEW DSP Test Toolkit позволяют работать с проектами CCS. Это означает, что можно открыть файл проекта (.pjt-файл) скомпоновать его, а затем загрузить результат в плату DSK.

На рис. 13.1 показано, как автоматизировать процесс компиляции кода DSP и загрузить его в плату DSK. Для открытия проекта необходимо указать расположение .pjt-файла на соответствующем входе ВП «Открыть проект CCS» (CCS Open Project VI). Для компоновки проекта используется ВП CCS Build VI, который создает код DSP. ВП CCS Download Code VI загружает код в плату DSK. Далее для запуска загруженного кода применяется ВП CCS Run VI. Виртуальные

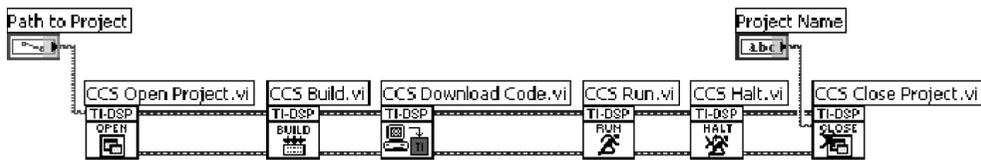


Рис. 13.1. Автоматизация CCS

приборы CCS Halt VI и CCS Close Project VI останавливают выполнение кода и закрывают проект CCS.

### 13.1.2. Обмен данными с кодом DSP

После создания кода DSP и загрузки на плату DSK, можно разработать в LabVIEW тестирующую систему, для проверки корректности его работы. Использование системы тестирования позволяет передать результаты симуляции в процессор и проанализировать выходные данные. Для этого в LabVIEW предустановлена палитра функций, использующих технологию RTDX. Входными и выходными данными могут быть любые тестовые величины, которые необходимо передать в код DSP и считать для анализа.

На рис. 13.2 показана блок-диаграмма, поясняющая процесс передачи и приема данных с кода DSP, запущенного на плате DSK. Функциональный генератор (Basic Function Generator VI) генерирует синусоиду, далее этот сигнал подается на вход ВП CCS RTDX Write Array I16, записывающий массив данных в процессор, затем без прерывания выполнения кода, результат считывается с выходного канала ВП CCS RTDX Read Array I16 VI.

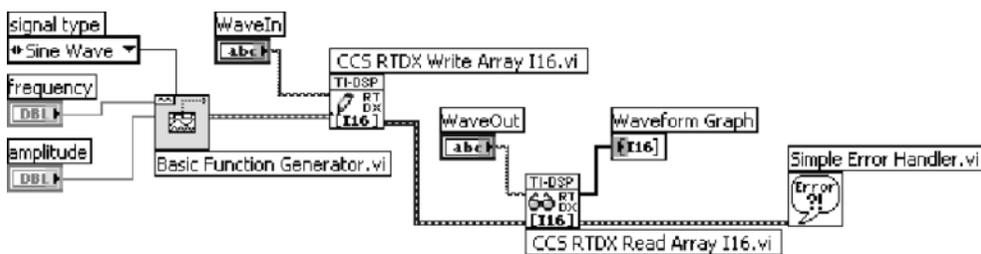


Рис. 13.2. Обмен данными с сигнальным процессором

### 13.1.3. Функции углубленного тестирования сигнальных процессоров (DSP Test Advanced VIs)

ВП углубленного тестирования выделены синим цветом и используют функции низкого уровня CCS и RTDX, такие как запуск CCS, включение или отключение коммуникации по технологии RTDX.

На рис. 13.3 показана блок-диаграмма ВП «Загрузка кода CCS» (CCS Download Code VI), построенная с использованием функций углубленного тестирования. ВП CCS Get Build Path VI возвращает полный путь к скомпилированному файлу с расширением .out. Узел вызова (Invoke node) вызывает метод, по которому загружается указанный файл в плату DSK. Функция информации о файле/директории (File/Directory Info function) возвращает размер .out-файла, используя данные о котором, ВП вычисляет необходимое время загрузки. Во время выполнения программы ВП CCS Status VI отображает пользовательские данные о текущем состоянии операции.

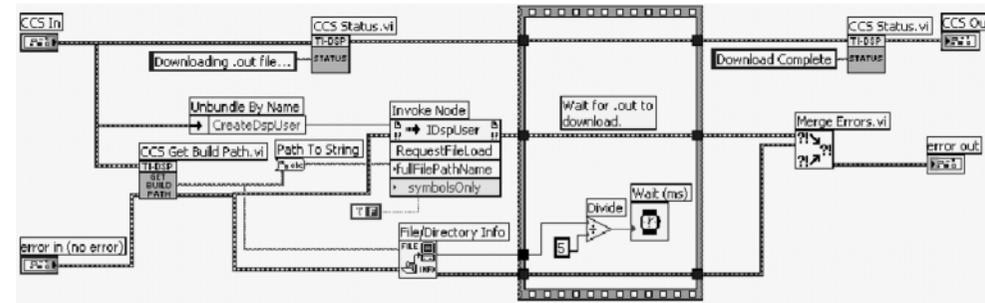


Рис. 13.3. Использование функций углубленного тестирования

## 13.2. Пакет LabVIEW DSP Module

LabVIEW DSP-модуль предназначен для проектирования систем на базе сигнальных процессоров. Его использование позволяет применить такие методы цифровой обработки сигналов (ЦОС), как спектральный анализ и фильтрация для программирования сигнальных процессоров в LabVIEW.

При использовании графического программирования время разработки приложений для поддерживаемых аппаратных средств значительно снижается, так как нет необходимости в знании текстовых языков программирования.

Кроме того, модуль оснащен удобной справкой, что позволяет создавать более производительный код.

### 13.2.1. Поддерживаемые аппаратные средства

LabVIEW DSP-модуль поддерживает различные устройства, включая NI SPEEDY-33, TMS320C6713 DSK и TMS320C6711 DSK. Он адресует аналого-цифровое преобразование, цифроаналоговое преобразование или цифровой ввод/вывод в целевые устройства для выполнения единичных или многоточечных операций. При создании приложений инженеры могут конфигурировать порт, тактирование, размер буфера и некоторые другие опции. В том случае, когда необходимо пере-

ключиться на другое аппаратное средство, узлы ввода/вывода автоматически изменяются для работы с новым устройством.

### 13.2.2. Области применения

LabVIEW DSP-модуль предназначен для использования в любых инженерных задачах, в решении которых, как правило, применяют сигнальные процессоры. Его можно применять для изучения фундаментальных основ ЦОС с применением аппаратных средств, проектирования систем связи, звуковой обработки и управления двигателями.

На рис. 13.4–13.6 показан процесс проектирования типичного приложения с использованием DSP-модуля. После создания виртуального прибора необходимо его скомпилировать и загрузить полученный код в выбранное устройство. Данный программный код очень компактен и эффективен для большинства приложений с доступной памятью на 32 кб слов. При работе с устройствами NI SPEEDY-33, TI C6711 DSK и C6713 DSK можно загрузить ВП во флэш-память для испытания реальных систем. Одно из преимуществ графического программирования процессоров с использованием DSP-модуля – это наличие лицевой панели прибора, которая является интерактивным интерфейсом пользователя при обмене данными с устройством. В процессе создания ВП пользователь легко может добавлять элементы управления и контроля для удобства отладки и проверки правильности работы кода.

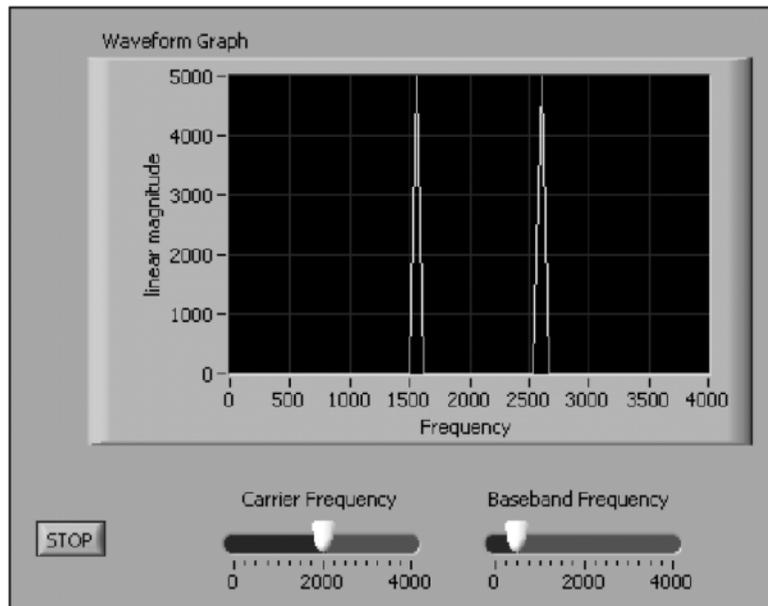


Рис. 13.4. Лицевая панель ВП Heterodyne VI

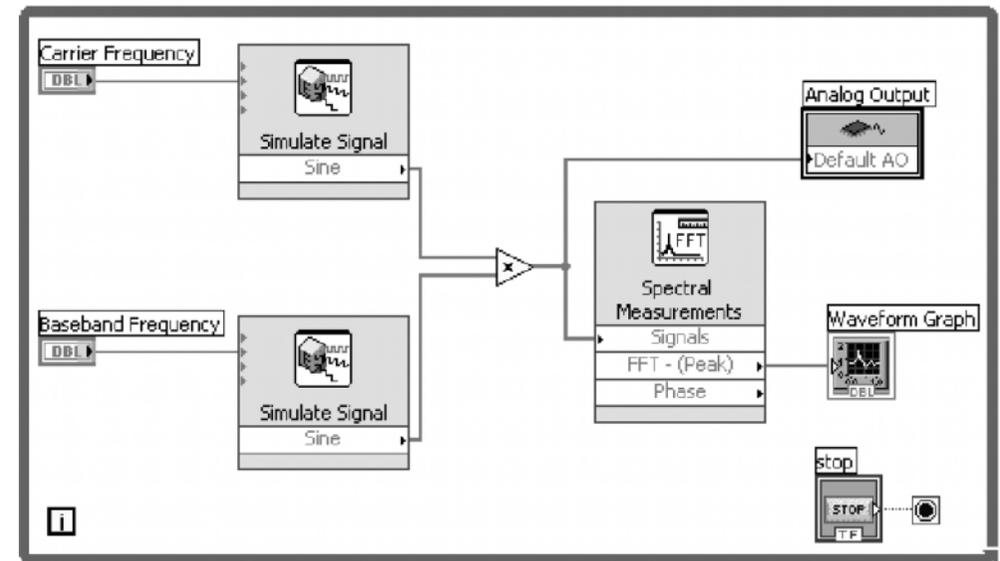


Рис. 13.5. Блок-диаграмма ВП Heterodyne VI

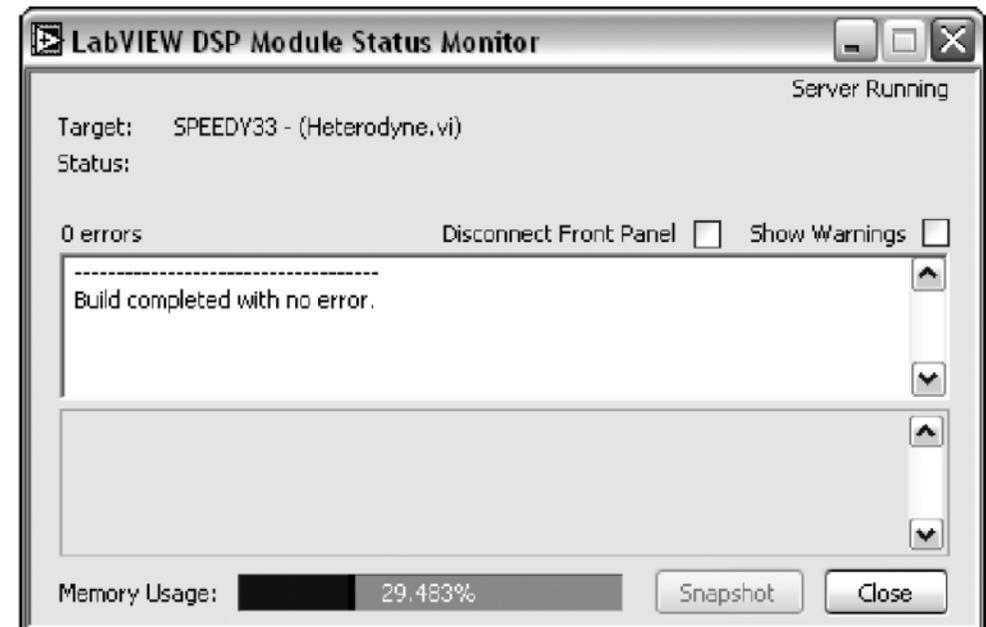


Рис. 13.6. Окно LabVIEW DSP Module Status Monitor

На рис. 13.4 представлена лицевая панель ВП Гетеродин (Heterodyne VI). Как видно, создание интерфейса пользователя в DSP-модуле осуществляется идентично работе с LabVIEW. График на лицевой панели прибора отображает сигнал гетеродина. Элементы управления, размещенные внизу экрана, позволяют изменять его частотные настройки.

На рис. 13.5 показана блок-диаграмма ВП Heterodyne VI. В ней используются следующие приборы:

- **Simulate Signal Express VI** – экспресс ВП, генерирующий синусоидальный сигнал. В одном случае он генерирует сигнал несущей частоты, а в другом – частоты гетеродина. Далее два синусоидальных сигнала перемножаются, формируя смешанный сигнал, который подается на вход узла аналогового вывода;
- **Spectral Measurements Express VI** – вычисляет быстрое преобразование Фурье и отображает спектр сигнала на графике, расположенном на лицевой панели;
- **Analog Output Elemental I/O Node** – записывает данные в цифроаналоговый преобразователь, также называемый кодеком (CODEC), в устройстве SPEEDY-33. Пользователь может самостоятельно определять, каким образом данные пишутся в аналоговый выход, двойным нажатием левой кнопки мыши в поле узла Elemental I/O Node.

Для того чтобы откомпилировать, загрузить и запустить ВП в устройстве SPEEDY-33, необходимо нажать кнопку Run. После этого появляется окно статуса выполнения указанных операций (LabVIEW DSP Module Status Monitor) (рис. 13.6).

Во время выполнения кода в DSP на переднем плане автоматически отображается лицевая панель ВП. Пользователь может изменять несущую частоту и частоту гетеродина, используя элементы управления. После внесения изменений ВП автоматически записывает данные в устройство без затрат времени на модификацию кода. При этом на графике динамически отображается спектр сигнала. Кроме того, если подключить наушники к аналоговому выходу устройства SPEEDY-33, можно услышать все изменения по частоте. Для завершения выполнения программы необходимо нажать кнопку Stop.

## Виброакустический анализ в LabVIEW

14

Пакет *Sound and Vibration Toolkit* (комплект инструментов для измерения звука и вибраций) National Instruments добавляет в LabVIEW библиотеку компонентов и элементов отображения для выполнения измерений звуковых частот, октавного анализа, анализа гармонических сигналов, измерений уровня звука, частотного анализа и анализа переходных процессов. Он также предоставляет удобную основу для использования ПО с более чем пятьюдесятью примерами для виброакустического анализа. Этот пакет является одним из наиболее функционально разработанных пакетов программ для звукового и вибрационного приложений.

Основанный на библиотеке функций, обладающих гибкой функциональностью, *Sound and Vibration Toolkit* демонстрирует, каким образом можно использовать программный подход для проведения уникальных измерений. Например, можно использовать цифровой мультиметр для выполнения высоковольтных звуковых измерений, преобразователь частоты – для измерения искажения модулированного радиосигнала или 24-разрядное устройство сбора динамических сигналов (DSA) – для тестирования громкоговорителя.

### 14.1. Функциональные возможности *Sound and Vibration Toolkit*

Анализ вибраций и звука часто начинается с записи сигналов с микрофонов, акселерометров, датчиков смещения или тахометров. Чтобы получить данные с этих и других датчиков, можно использовать любую плату, измеряющую динамические сигналы (DSA) или осуществляющую сбор данных (DAQ), так как данные устройства являются аппаратными и частотно независимы. После записи данных полученный сигнал можно отображать в единицах измерения или в децибелах. Все стандартные процедуры отслеживают эти параметры для выполнения одноканальных и многоканальных измерений сигналов и получения масштабирован-

ных и калиброванных результатов. Полный перечень возможных измерений показан на рис. 14.1.

Пакет *Sound and Vibration Toolkit LabVIEW* включает в себя набор экспресс-виртуальных приборов (Express VI), которые облегчают выполнение измерений и анализа вибрационных и звуковых сигналов. В их число входят 10 ВП:

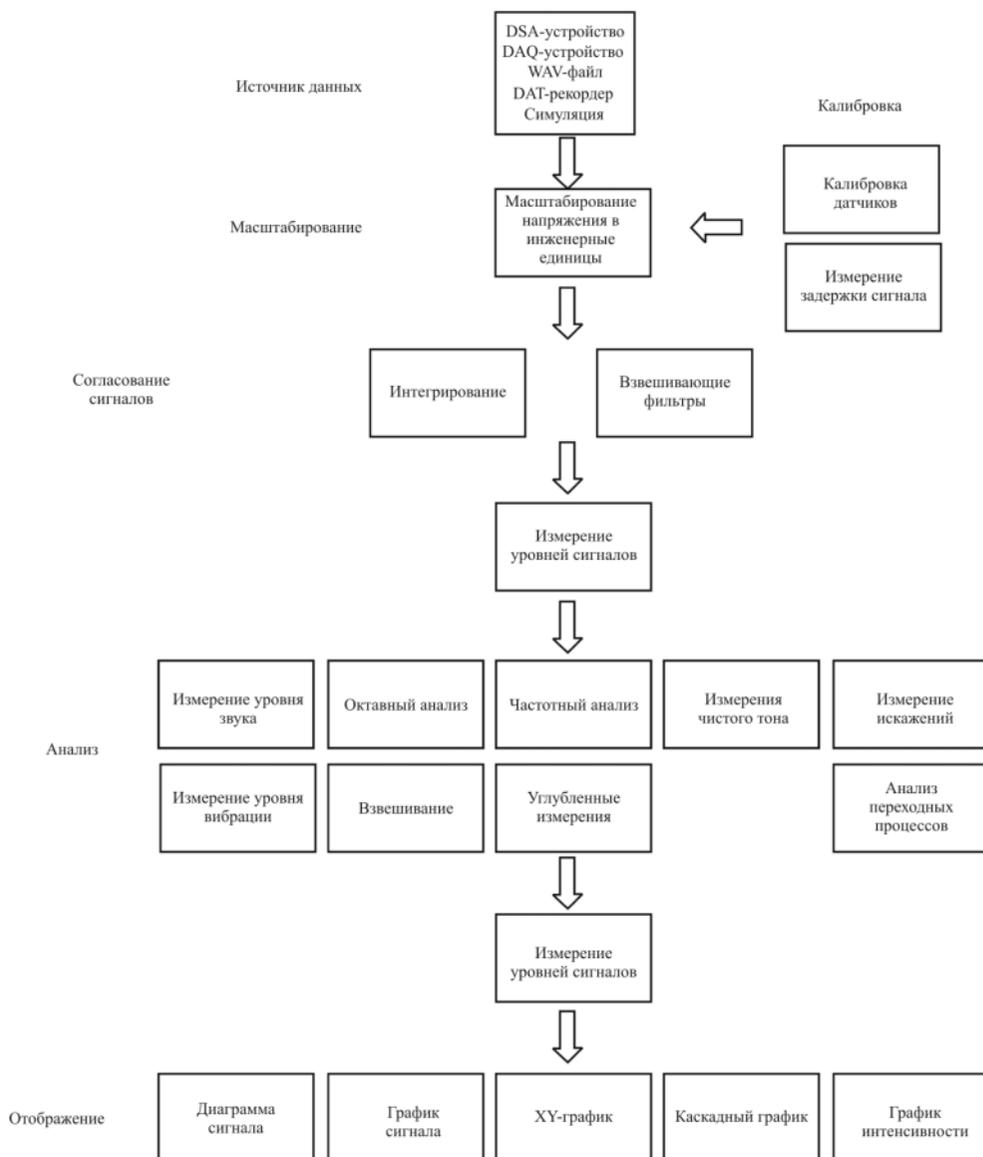


Рис. 14.1. Применение пакета *Sound and Vibration Toolkit* для виброакустического анализа

- для октавного анализа с А-, В- или С-взвешиванием частотных характеристик;
- для измерений уровня вибрации с однократным или двойным интегрированием;
- для измерения уровня звука с А-, В- или С-взвешиванием;
- для измерения спектра мощности;
- для измерения спектра мощности с изменением масштаба отображения;
- для измерения пиковых отклонений;
- для измерения мощности в заданной полосе частот;
- для измерения частотных характеристик;
- для измерения уровней сигналов;
- для фильтрации.

Пакет *Sound and Vibration Toolkit* включает в себя множество палитр функций для таких акустических измерений таких параметров, как усиление, фаза, анализ искажений и анализ гармонических сигналов. Гармонический анализ – это эффективный метод измерения частотных характеристик. С помощью *Sound and Vibration Toolkit* можно легко генерировать сигналы и исследовать отклик на выходе измерительного устройства. В отличие от обычных частотных характеристик гармонический анализ позволяет также измерять частотно-зависимые искажения и учитывать индивидуальные для устройства требования к настройкам. Помимо акустических измерений этот метод анализа является превосходным инструментом для многих приложений тестирования механических вибраций. Другие звуковые измерения, проводимые пакетом *Sound and Vibration Toolkit*, включают в себя усиление, фазу, суммарный коэффициент гармоник, интермодуляционное искажение, динамический диапазон и линейность фазочастотной характеристики.

Он также включает в себя примеры проведения указанных измерений, так что пользователь может быстро объединять аналоговый вывод, аналоговый ввод и математический анализ данных с целью построения собственных приложений. Кроме того, в библиотеку пакета входят примеры по генерации сигналов, которые необходимы для обеспечения сигнала возбуждения, требуемого в большинстве виброакустических измерений. Например, в *Sound and Vibration Toolkit* включена библиотека из 33 видов сигналов.

Пакет для виброакустического анализа оптимизирует LabVIEW для выполнения измерений шума и вибраций. Например, спектральный анализ можно проводить с изменением масштаба отображения, т. е. анализировать тот частотный диапазон, который необходимо исследовать. Интегрирование сигналов может проводиться как во временной, так и в частотной области, что позволяет легко преобразовывать сигналы, полученные с акселерометров, в скорость и/или смещение для дополнительного анализа. Кроме того, с помощью предустановленных функций обеспечивается линейное усреднение, экспоненциальное усреднение и фиксация пика. Благодаря встроенному октавному анализу можно выполнять измерения в нескольких частотных диапазонах с любой частотой выборки.

## 14.2. Поддерживаемые аппаратные средства

Специально разработанные для виброакустического анализа устройства DSA NI 44xx являются полностью совместимыми с данным программным пакетом и соответствуют следующим стандартам на звуковой и октавный анализ:

- IEC 61260 : 1995, класс 1;
- IEC 61672 : 2002, класс 1;
- ANSI S1.11 – 2004, класс 1;
- ANSI S1.4 – 1983;
- ANSI S1.42 – 1986.

Функции пакета **Sound and Vibration Toolkit** доступны как для DSA-устройств, так для многофункциональных устройств сбора данных NI (DAQ), их можно использовать совместно с драйверами **Traditional** NI-DAQ и NI-DAQmx, предустановленными в LabVIEW.

## 14.3. Перечень и описание виртуальных приборов и функций, входящих в пакет

Предустановленные в **Sound and Vibration Toolkit** ВП подходят для выполнения любых звуковых и вибрационных измерений. Они выполняют выбранный тип анализа и выводят результаты на стандартные элементы отображения LabVIEW в выбранных единицах измерения, таких как амплитуда/фаза, действительная/мнимая часть и включение/выключение отображения в дБ.

### 14.3.1. Масштабирование

ВП **Scaling** VI используется для изменения масштаба данных во временной области для того, чтобы установить выбранные единицы отображения. ВП **Scaling** VI предназначен для работы с измерительными каналами, не масштабированными в Measurement & Automation Explorer (Помощник для измерения и автоматизации). ВП **Scale Voltage to EU** VI масштабирует исходный сигнал для отображения в инженерных единицах.

### 14.3.2. Калибровка

Для калибровки в **Sound and Vibration Toolkit** предустановлены ВП **Calibration** VIs, с помощью которых осуществляется калибровка микрофонов, акселерометров и динамических датчиков. При помощи указанных ВП можно выполнять калибровку выбранного канала и измерять задержку распространения в измерительном устройстве (рис. 14.2).

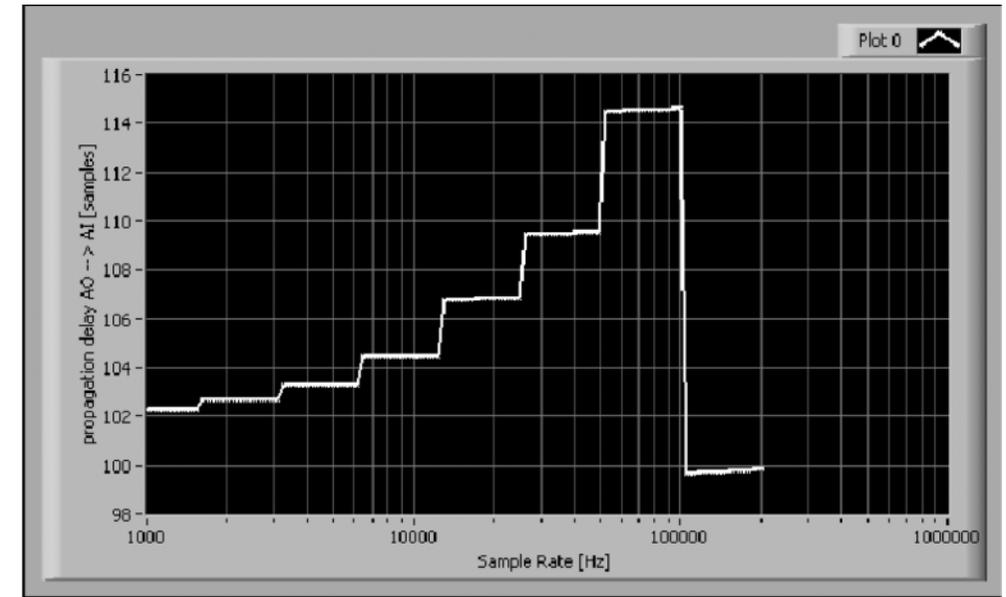


Рис. 14.2. Зависимость задержки распространения сигнала от частоты выборки в DSA устройстве PXI-4461, построенная с использованием **Sound and Vibration Toolkit**

### 14.3.3. Тестирование (измерение) уровней сигналов

ВП **Limit Testing** VI предназначен для обнаружения сигналов по заданному уровню во временной и частотной областях при анализе гармонического спектра, двухкоординатных данных, пиковых отклонений и результатов скалярных измерений (рис. 14.3 и 14.4). Данный ВП применяется для анализа предельных значений любого типа результатов измерений, полученных с помощью **Sound and Vibration Toolkit**.

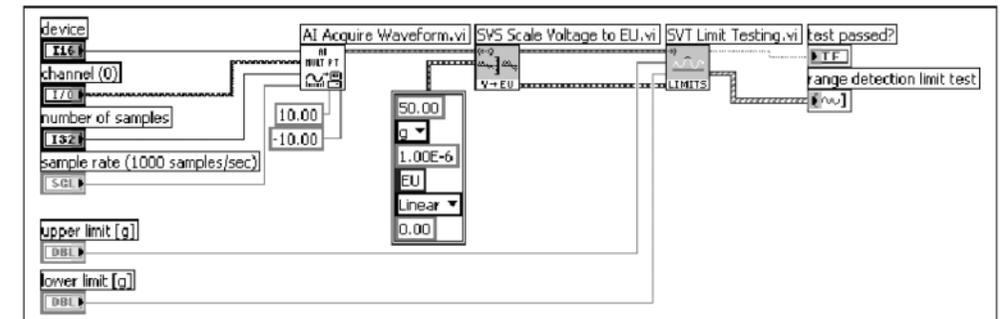


Рис. 14.3. Пример блок-диаграммы прибора, измеряющего уровни сигнала

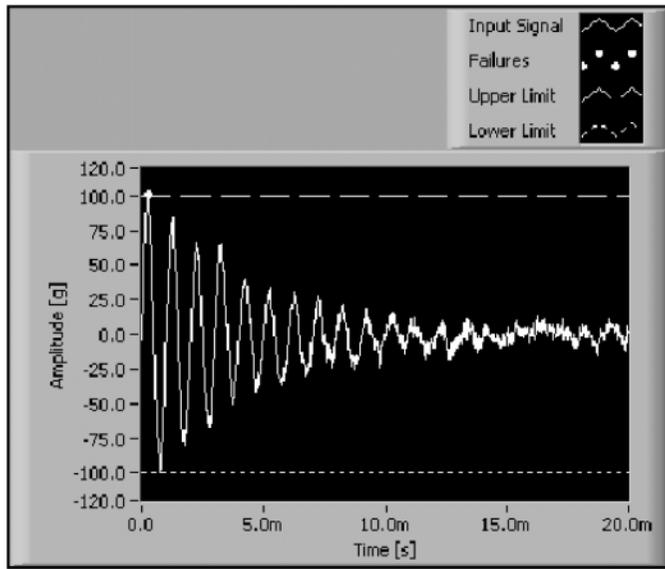


Рис. 14.4. Лицевая панель прибора, измеряющего уровни сигнала

### 14.3.4. Взвешивание

ВП *Weighting VIs* предназначены для выполнения взвешивания данных во временной области. Их использование позволяет применить взвешивающие фильтры А-, В- или С-типа к сигналу во временной области (рис. 14.5 и 14.6). Кроме того, для приложений радиосвязи доступны ITU-R 468-4 и *Dolby*-фильтры, а для задач телекоммуникации имеются фильтры С-сообщений и ССITТ.

### 14.3.5. Интегрирование

ВП *Integration VI* позволяет выполнять интегрирование во временной области с целью преобразования данных об ускорении в данные о скорости (однократное

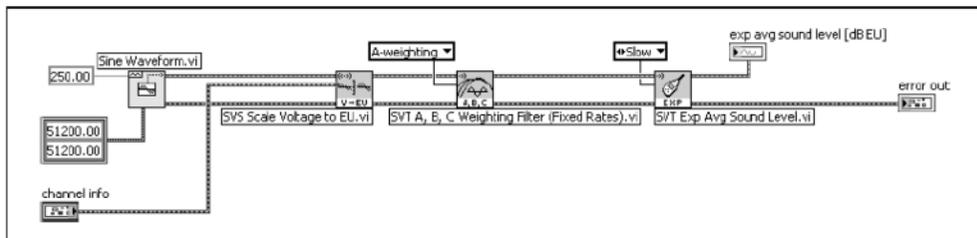


Рис. 14.5. Блок диаграмма прибора, осуществляющего взвешивание амплитуд синусоид по А-типу частотной характеристики

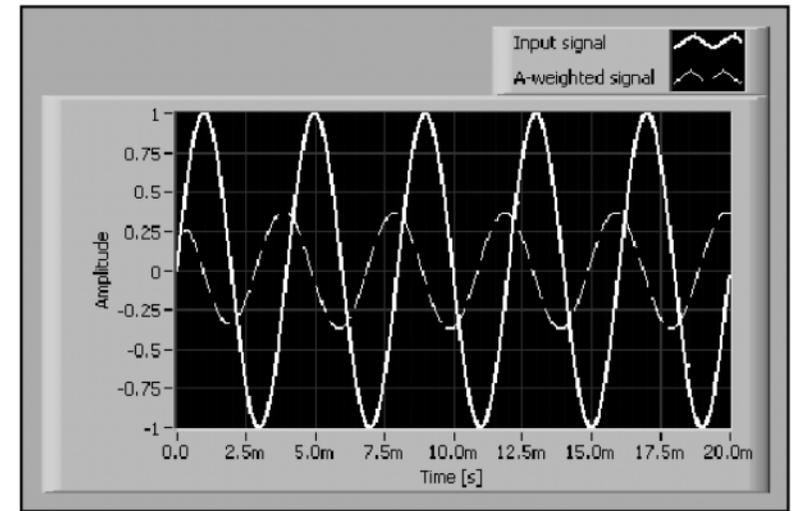


Рис. 14.6. Входной гармонический сигнал (непрерывная линия) и А-взвешенный сигнал (пунктирная линия) на лицевой панели прибора

интегрирование), ускорение в смещение (двойное интегрирование) и скорость в смещение (рис. 14.7).

### 14.3.6. Генерация

ВП *Generation VIs* предназначены для генерации тестовых сигналов в задачах измерения звука и вибраций. Например, ВП *SVT Pink Noise Waveform VI* помогает

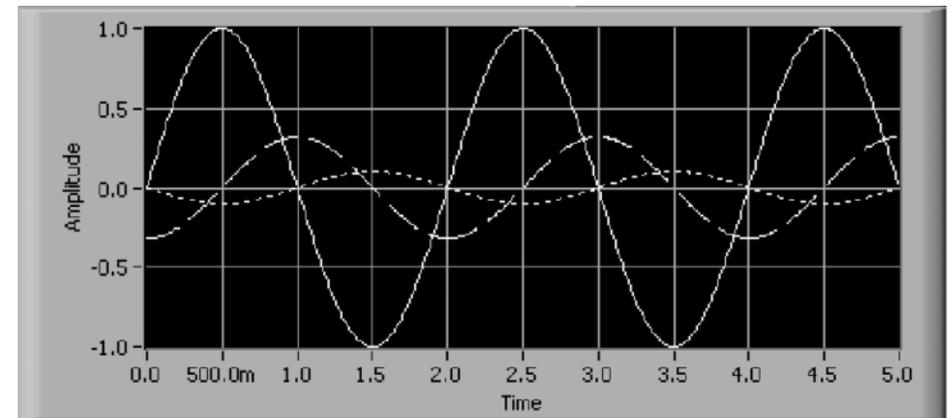


Рис. 14.7. Лицевая панель прибора, интегрирующего синусоиду с частотой 0,5 Гц. Показаны ускорение (непрерывная линия), скорость (прерывистая линия) и смещение (пунктирная линия)

генерировать стандартный непрерывный «розовый» шум, у которого спектральная плотность мощности уменьшается по частоте на 3 дБ на октаву.

### 14.3.7. Уровень вибрации

ВП *Vibration Level* VIs предназначены для измерения уровня вибрации и отображения результатов в линейном масштабе. Данные ВП наделены функциями, которые обычно используются для анализа вибраций, таких как измерения коэффициента отклонения. Доступные режимы усреднения включают в себя усреднение СКО, экспоненциальное усреднение и пиковую фиксацию.

### 14.3.8. Уровень звука

ВП *Sound Level* VIs предназначены для измерения уровня звука и отображения результатов в децибелах (дБ). Эти приборы обеспечивают проведение обычных измерений уровня звука, включая непрерывное эквивалентное усреднение (Leq), экспоненциальное усреднение и фиксацию пиковых значений.

### 14.3.9. Октавный анализ

ВП *Octave Analysis* VIs предназначены для измерения октавного спектра, 1/3-октавного (терц-октавного) и 1/n-октавного (рис. 14.8). Можно также выполнять расширенные измерения, такие как взвешивание и контроль по диапазону значений. ВП *Octave Analysis* VIs обладают полным набором инструментов для выполнения фракционно-октавного анализа, включая анализ полного, 1/3-, 1/6-, 1/12- и 1/24-октавного диапазона. Кроме того, они могут производить анализ любой частоты выборки и любого количества фракционно-октавных диапазонов.

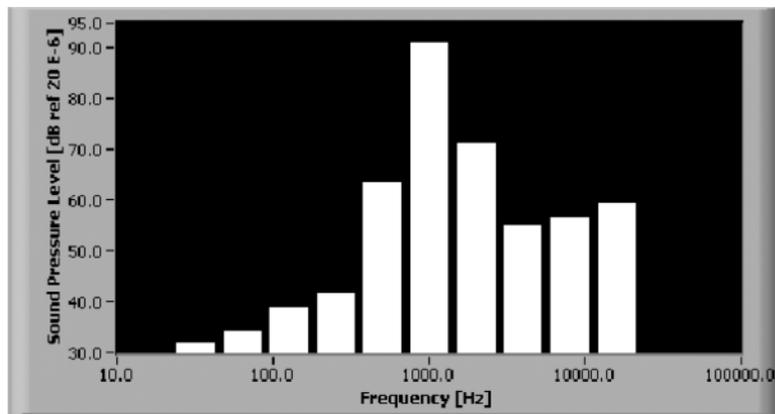


Рис. 14.8. Лицевая панель прибора, вычисляющего полнооктавный спектр

## 14.3.10. Частотный анализ

ВП *Frequency Analysis* VIs предназначены для проведения частотного анализа в заданной полосе частот с изменением масштаба отображения. Одноканальные измерения, проводимые с помощью данных ВП, включают в себя измерения спектра мощности, спектральной плотности мощности, амплитудного и фазового спектра и спектра действительной и мнимой составляющих комплексного (аналитического) сигнала или его комплексной огибающей. Указанные функции можно применять также и при многоканальных измерениях. Двухканальные измерения включают в себя анализ взаимного спектра сигналов, частотных характеристик и когерентной выходной мощности. Кроме того, для частотного анализа можно применять другие ВП, входящие в *Sound and Vibration Toolkit*, благодаря которым выполняются такие измерения, как пиковое детектирование, вычисление мощности в заданной полосе, преобразование единиц измерений и контроль по диапазону значений.

### 14.3.11. Анализ переходных процессов

ВП *Transient Analysis* VIs предлагают два метода получения информации о неустановившихся сигналах. Один из них это быстрое преобразование Фурье (STFT) для получения информации о частоте как зависимости от времени или от скорости изменения. Второй – это анализ спектра отклика на мгновенное воздействие (SRS).

### 14.3.12. Каскадный график

Благодаря ВП *Waterfall Display* VIs пользователь может выводить на экран результаты частотного и октавного анализов в виде каскадных графиков. ВП *Waterfall Display* VIs генерируют и управляют ими (рис. 14.9).

### 14.3.13. Гармонический анализ

ВП *Swept-Sine* VIs предназначены для анализа гармонических сигналов. С их помощью можно выполнять такие измерения, как оценка уровня среднеквадратического значения, амплитудных и фазовых характеристик, суммарного коэффициента гармоник и частного коэффициента гармоник. Данный вид анализа обычно используется в целях описания частотной характеристики тестируемого устройства (*Device under test* – DUT). На рис. 14.10 представлена лицевая панель ВП, вычисляющего спектр мощности.

### 14.3.14. Анализ искажений

ВП *Distortion* VIs позволяют выполнять анализ искажений блока данных во временной области. ВП SNR VI, SINAD VI, THD+N VI и THD VI выводят комплексный спектр и результаты измерений. Кроме того, их можно использовать совме-

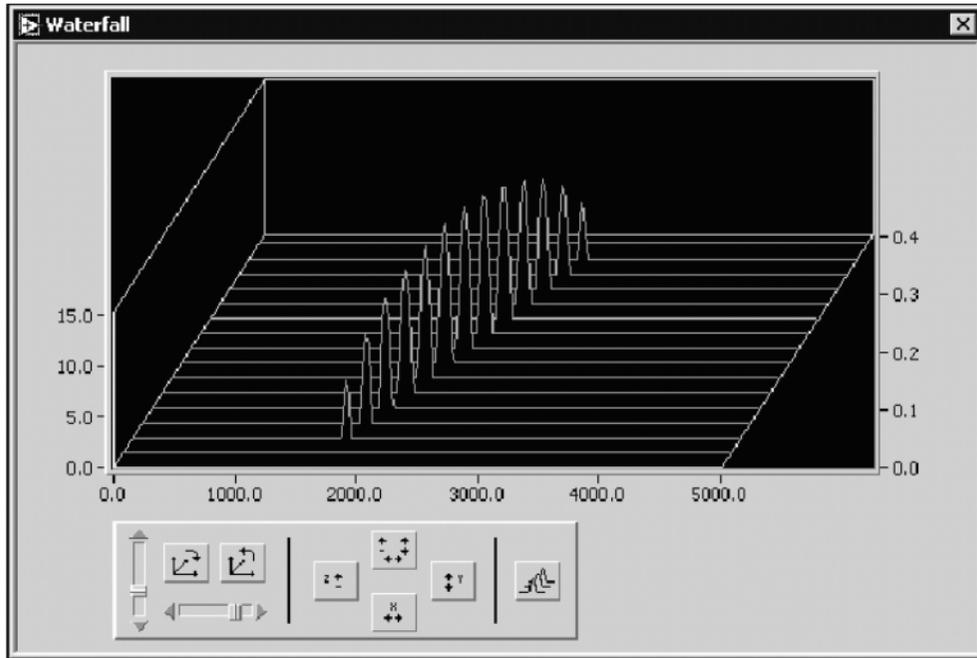


Рис. 14.9. Пример использования каскадного графика при спектральном анализе

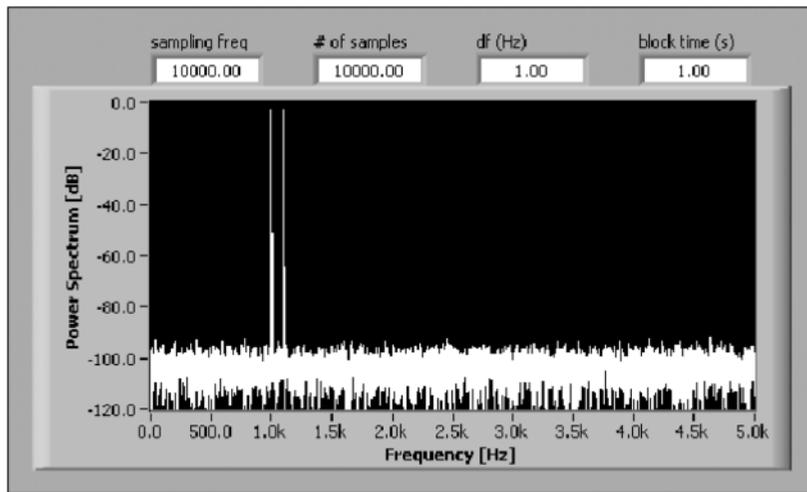


Рис. 14.10. Лицевая панель прибора, вычисляющего спектр мощности сигнала с установленными параметрами: временем измерения, частотой дискретизации и количеством точек

стно для повышения производительности. Благодаря ВП *Distortion VIs*, возможны вычисления гармонических искажений, интермодуляционных искажений и нелинейности фазы в тестируемом устройстве.

### 14.3.15. Чистый тон

ВП *Single-Tone VIs* позволяют выполнять измерения чистого тона. С их помощью можно измерять амплитуду и фазу, перекрестные помехи, динамический диапазон, шум в канале и динамический диапазон без помех. Каждый ВП выводит комплексный спектр и результаты измерений. Эти функции также можно применять последовательно для повышения производительности. Данный вид анализа применяется, как правило, для измерения линейной и шумовой характеристик тестируемого устройства.

# Программирование ПЛИС (FPGA) в LabVIEW

# 15

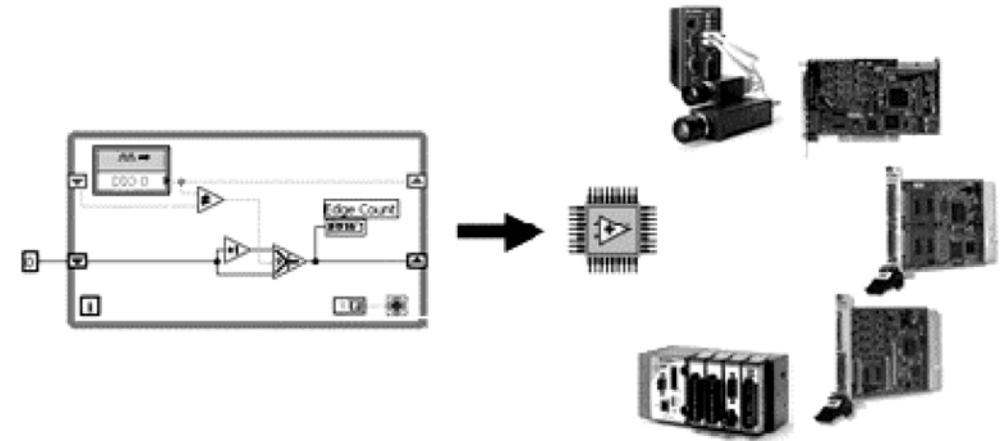


Рис. 15.1. Блок-диаграмма ВП в модуле LabVIEW FPGA и реконфигурируемые устройства

Применение LabVIEW FPGA-модуля позволяет использовать графический интерфейс LabVIEW для программирования ПЛИС (FPGA), используемых в реконфигурируемых устройствах ввода/вывода корпорации National Instruments (NI RIO) [20, 21]. Совместное использование указанных средств предоставляет гибкую платформу для создания измерительных и управляющих систем, которые раньше требовали индивидуальной разработки аппаратных средств.

ПЛИС – это микросхема, которая состоит из множества логических вентилей. В отличие от традиционных микросхем, обладающих заданной функциональностью, ПЛИС можно конфигурировать и реконфигурировать применительно к каждой конкретной задаче. Подобные устройства используются в приложениях, где применение специализированных интегральных микросхем (ASIC) проблематично, либо система требует реконфигурации после сдачи в эксплуатацию. Связано это с тем, что в ряде задач требуется выполнение устройством обработки сигналов по специфическим алгоритмам. В настоящее время ПЛИС используются во многих отраслях, таких как автомобилестроение, авиация, бытовая техника и т. д. [16].

Исторически программирование ПЛИС ограничивалось инженерами, обладающими глубокими познаниями VHDL либо других средств проектирования низкого уровня. Для того чтобы запрограммировать ПЛИС с помощью FPGA-модуля, никаких дополнительных знаний кроме владения LabVIEW не требуется. Кроме того, параллелизм графического программирования данной среды идеально подходит для архитектуры ПЛИС в приложении к синхронным и асинхронным параллельным задачам. На рис. 15.1 показаны устройства National Instruments, программируемые с помощью LabVIEW FPGA-модуля.

## 15.1. Реконфигурируемые устройства ввода/вывода (RIO)

Реконфигурируемые (Reconfigurable) устройства National Instruments (NI) на шине PXI или PCI позволяют проводить высокоскоростную обработку аналоговых и цифровых сигналов с использованием различных протоколов обмена данными, датчиков и других измерительных устройств.

Платформа NI CompactRIO основана на RIO-технологии и представляет собой компактную модульную систему, предназначенную для создания мобильных измерительных комплексов с высокими характеристиками, работающую в жестких промышленных условиях [16]. Данная система работает под управлением операционной системы реального времени и легко реконфигурируется посредством простой замены измерительных модулей, установленных в слоты шасси.

NI Compact Vision System – это система машинного зрения, предназначенная для автономной работы и применяемая в автоматизированных системах различного назначения. Данная система осуществляет обмен данными по технологии FireWire, а также обладает встроенными сетевыми возможностями для распределения выполнения задачи между несколькими устройствами.

Устройства тактирования и синхронизации, такие как NI PXI-6653 и NI PXI-6652 синхронизируют работу различных устройств. Программируемые с помощью LabVIEW FPGA-модуля, они позволяют использовать различные методы синхронизации, управляемый запуск триггера, а также высокоскоростные аналоговые таймеры.

## 15.2. Создание проектов

При использовании LabVIEW FPGA-модуля конфигурирование ПЛИС осуществляется посредством программирования в LabVIEW. Данное свойство максимально удобно для инженеров, поскольку используется тот же самый графический интерфейс.

При программировании реконфигурируемых устройств LabVIEW отображает только те функции, которые могут быть заложены в ПЛИС. Палитра функций модуля LabVIEW FPGA представлена на рис. 15.2 и включает наряду с типичными структурами (While, For, Case, Sequence) и функциями специальные математические операции, функции генерации и анализа, линейного и нелинейного контроля, сравнительной логики, манипуляции с массивами и кластерами, аналогового и цифрового ввода/вывода и тактирования.

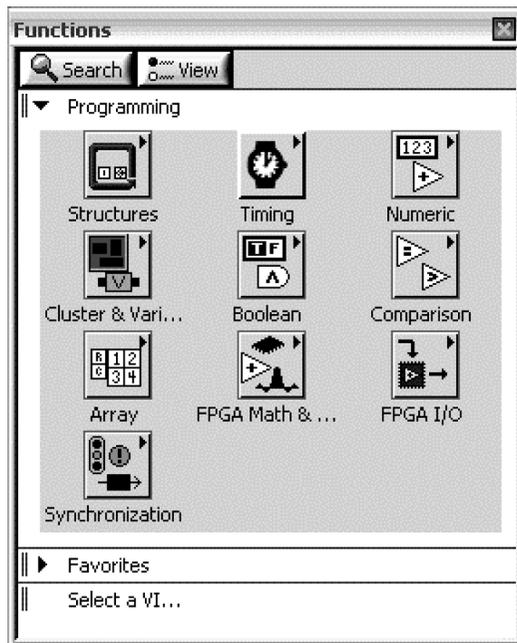


Рис. 15.2. Палитра функций модуля LabVIEW FPGA

На рис. 15.3 показана блок-диаграмма простейшего счетчика по нарастающему фронту. Работа реконфигурируемого устройства в данном случае происходит следующим образом: с цифровой линии непрерывно считывается ее статус от одной итерации к другой; когда статус линии изменяется, показание счетчика увеличивается на единицу. Размер таймера определяется типом данных I32, т. е. данный счетчик является 32-битным.

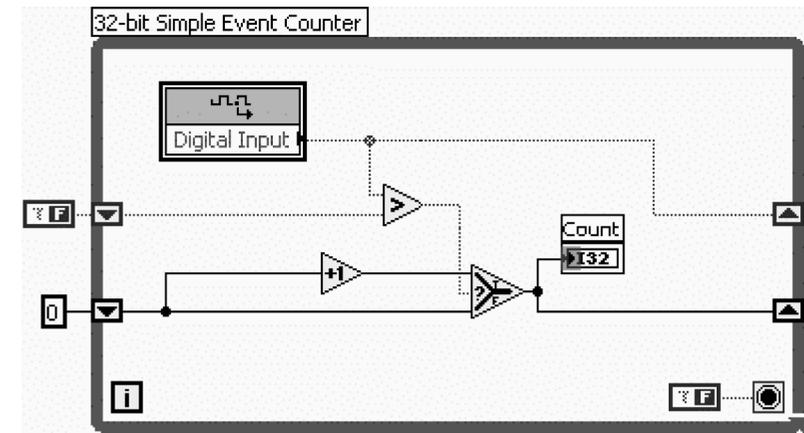


Рис. 15.3. Пример простейшей программы FPGA

В типичных устройствах сбора данных счетчик представляет интегральную микросхему, и программирование ведется с использованием ВП NI-DAQmx VIs. В LabVIEW FPGA-модуле создается свое собственное приложение в выделенной части ПЛИС. Использование подобного счетчика непосредственно в LabVIEW проблематично, поскольку требует программных вызовов из операционной системы. В связи с тем, что LabVIEW FPGA использует блок-диаграмму для выполнения приложения непосредственно в устройстве сбора данных на базе ПЛИС, данный счетчик будет работать аналогично построенной структуре.

Хотя данный пример показывает простейшую структуру счетчика, тем не менее, добавление дополнительной функциональности требует простой замены исходного кода LabVIEW. В отличие от традиционных процессоров, ПЛИС имеет параллельную структуру, добавление дополнительных программных циклов в которую не затрагивает работу других независимых структур.

На рис. 15.4 код изменен для генерации цифрового импульса в каждой четвертой итерации цикла. При этом код помещен в специальную структуру LabVIEW – Single Cycle Timed Loop. Структура Single Cycle Timed Loop выполняется каждый тактовый цикл таймера ПЛИС. Это означает, что тактовая частота счетчика определяется тактовой частотой ПЛИС, таким образом, значение по умолчанию – 40 МГц или 25 наносекунд. Код, сгенерированный структурой Single-Cycle Timed Loop, более эффективен, чем код, созданный с помощью структуры While Loop, так как в ПЛИС он использует гораздо меньше вентилях. Однако некоторые операции не поддерживаются данной структурой, например, аналоговый ввод/вывод. В этом случае необходимо использовать цикл While. Как показано на рис. 15.4, в основном цикле производится чтение данных с аналогового входа (AI), затем осуществляются PID-вычисления и запись в аналоговый выход (AO). Тактирование осуществляется с помощью структуры Loop Timing VI, входным параметром которой могут быть миллисекунды, микросекунды либо такт системных часов (25 нс). В данном случае установлено значение – 10 мкс (100 кГц).

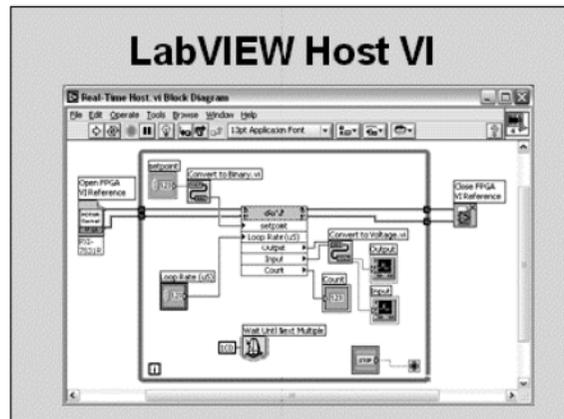
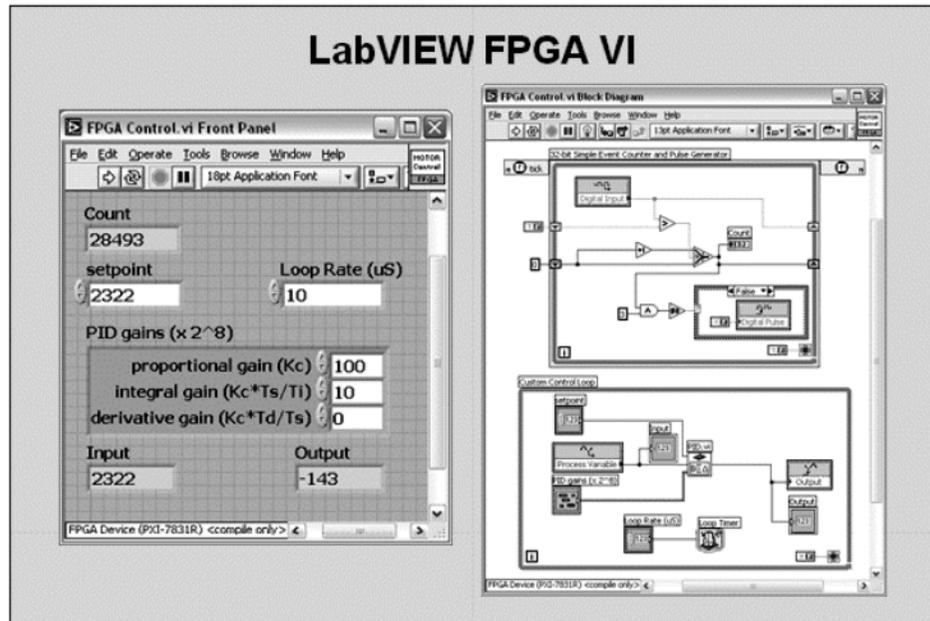


Рис. 15.4. Проекты FPGA VI и Host VI для приложений счетчика и аналогового контроля

### 15.3. Последовательность программирования ПЛИС

После создания проекта FPGA VI необходимо скомпилировать код для выполнения в ПЛИС. Аналогично другим пакетам для программирования ПЛИС, компиляция проекта FPGA VI может занимать от одной минуты до нескольких ча-

сов, в зависимости от сложности кода и параметров персонального компьютера. Для повышения производительности работы реконфигурируемого устройства можно использовать режим эмуляции (bit-accurate emulation mode), что позволяет проверить логику проекта до инициализации процесса компиляции. В этом режиме доступны те же инструменты отладки, что и в LabVIEW, такие как выполнение в замедленном режиме с отображением потока данных (execution highlighting), пробники (probes) и контрольные точки (breakpoints).

После того как FPGA-код скомпилирован, нужно создать ВП Host VI для интеграции реконфигурируемого устройства в среду тестирования и контроля. На рис. 15.5 показана последовательность создания приложений. ВП Host VI использует элементы управления и индикации на лицевой панели ВП FPGA VI для передачи данных между ПЛИС в измерительном устройстве и обрабатывающим ядром LabVIEW. Данные элементы представлены как регистраторы данных. Обрабатывающим ядром может быть как контроллер, работающий под управлением операционной системы Windows (PC и PXI), так и работающий под управлением операционной системы реального времени RTOS (PC, PXI, компактные системы машинного зрения, CompactRIO). В приведенном выше примере происходит обмен данными о скорости выполнения цикла, аналоговом вводе/выводе и значении счетчика между кодом FPGA VI в ПЛИС и ВП Host VI на главном компьютере.

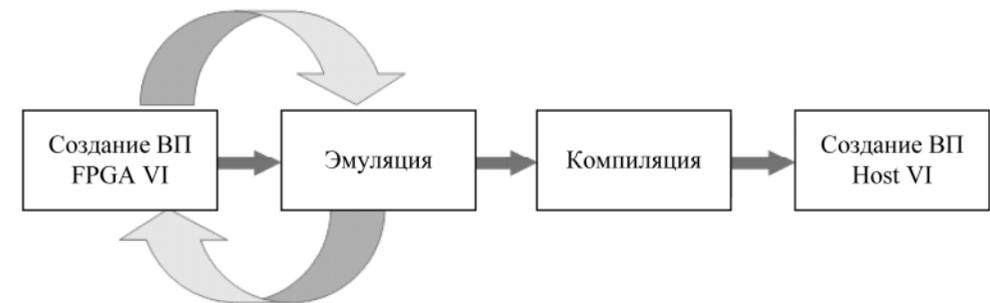


Рис. 15.5. Последовательность программирования ПЛИС

На рис. 15.6 показан набор функций, предназначенных для обмена данными с ВП FPGA VI. Первое, что необходимо сделать при создании ВП Host VI, – открыть ссылку на ВП FPGA VI и реконфигурируемое устройство. Функция ссылки (Open FPGA VI Reference function) загружает и запускает в ПЛИС откомпилированный FPGA-код. После установки соединения происходит запись в управляющие элементы и чтение данных с индикаторов (Read/Write Control function).

Перечень доступных элементов управления и индикации можно выбрать из списка вручную, после указания ссылки на FPGA VI. Для непрерывной записи/чтения данных необходимо поместить функцию обмена данными в цикл While (While Loop). И, наконец, последняя функция, необходимая для создания ВП

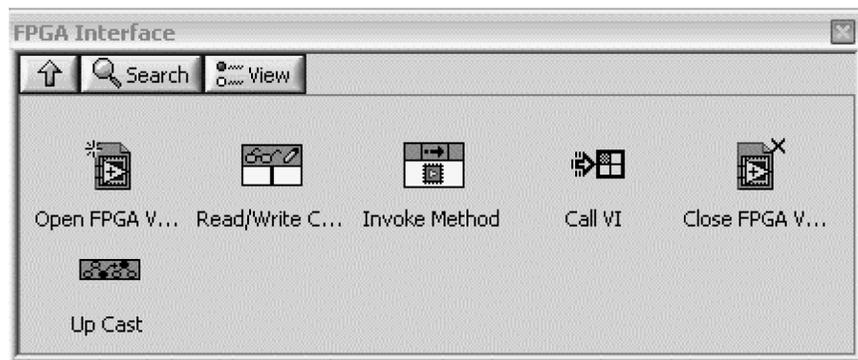


Рис. 15.6. Палитра функций FPGA

Host VI, – функция закрытия ВП FPGA VI (Close FPGA VI Reference function). При ее выполнении происходит остановка выполнения кода в ПЛИС и закрывается ссылка на устройство. После этого можно загрузить в ПЛИС другой откомпилированный код, для изменения функциональности реконфигурируемого устройства.

ВП Host VI может выполнять также любые математические операции, которые не влияют на работу ПЛИС.

## Резюме

Применяя LabVIEW FPGA-модуль, разработчик получает возможность использовать преимущества в скорости и синхронизации ПЛИС наряду с традиционными функциями, такими как сбор данных, управление движением, промышленные коммуникации и т. д. Интуитивно понятный интерфейс графического программирования LabVIEW предоставляет широкие возможности по созданию приложений измерения и контроля с помощью аппаратных средств.

# Инструментарий проектирования цифровых фильтров в LabVIEW



## 16.1. Введение в инструментарий проектирования цифровых фильтров в LabVIEW

Инструментарий проектирования (синтеза) цифровых фильтров (ИПЦФ) в LabVIEW (LabVIEW Digital Filter Design Toolkit) состоит из нескольких расширенных средств для проектирования фильтра, анализа и моделирования цифровых фильтров с фиксированной и с плавающей запятой. В этом подразделе обсуждается общий подход к процессу проектирования цифрового фильтра и вводятся инструментальные средства в ИПЦФ, которые можно использовать для разработки цифровых фильтров.

***Примечание.** Если есть необходимость в использовании виртуальных приборов (ВП) из палитры **Analyze** в полной (LabVIEW Full) или профессиональной (Professional Development) версии (System) LabVIEW для разработки программ проектирования фильтра, можно обратиться к пункту **Заметки к пользовательской расширенной библиотеке анализа в LabVIEW** этого подраздела.*

### 16.1.1. Краткий обзор процесса проектирования цифрового фильтра

Проектирование цифрового фильтра выполняется следующими тремя этапами:

1. Определение метода проектирования и параметров цифрового фильтра или характеристик, которыми цифровой фильтр должен обладать. Для этого необходимо обратиться к подразд. 16.2 для получения общей информации о технических требованиях для цифровых фильтров. Можно также об-

ратиться к подразд. 16.3, чтобы узнать, как использовать ИПЦФ для ввода технических требований для цифрового фильтра.

2. Анализ характеристик цифрового фильтра с плавающей запятой, который спроектирован на первом этапе. Для этого необходимо обратиться к подразд. 16.2 для получения информации об оценке характеристик цифрового фильтра.
3. Для проектирования фильтров с фиксированной запятой на основе алгоритма с арифметикой конечной точности можно обратиться к подразд. 16.4, чтобы изучить реализуемость цифровых фильтров с фиксированной запятой в ИПЦФ.

Процесс проектирования является итерационным. Необходимо экспериментировать с различными техническими требованиями при проектировании или выбирать методы разработки соответствующего цифрового фильтра для приложения. Иногда, возможно, потребуется пересмотреть технические требования или изменить метод проектирования после проверки фильтра, особенно при разработке фильтров с фиксированной запятой из-за конечной точности коэффициентов.

## 16.1.2. Краткий обзор инструментария LabVIEW для проектирования цифрового фильтра

ИПЦФ включает в себя большое разнообразие инструментальных средств для обеспечения помощи в проектировании цифровых фильтров, не требуя больших знаний в цифровой обработке сигналов или цифровых методов фильтрации. Например, ИПЦФ содержит ВП (Express VIs), которые можно использовать, чтобы обеспечить взаимодействие в графическом виде с техническими требованиями для фильтра при проектировании соответствующих цифровых фильтров.

В дополнение к инструментальным средствам, которые помогают быстро создавать цифровые фильтры, ИПЦФ включает инструментальные средства для обычного и разветвленного синтеза цифрового фильтра, преобразования фильтра с плавающей запятой к фильтру с фиксированной запятой, анализ фильтра и моделирование на настольном компьютере. Все эти операции описаны ниже.

Для получения полной информации можно обратиться к *Справке LabVIEW (LabVIEW Help)*, выбирая **Help VI, Function, & How-To Help**.

## 16.1.3. Обобщенный метод Ремеза и перечень $p$ -х норм синтеза алгоритмов

Можно использовать алгоритмы типа обобщенного метода Ремеза (Remez) и перечень методов  $p$ -х норм, чтобы определить произвольные АЧХ и ФЧХ для цифрового фильтра. Инструментарий включает ВП для автоматической оценки порядка фильтра. Можно обратиться к подразд. 7.7, для получения информации об использовании ВП DFD Remez Design и ВП DFD Least Pth Norm Design.

## 16.1.4. Большой выбор структур фильтра

Когда выполняется проектирование цифровых фильтров с помощью ИПЦФ, можно выбрать одну из 23 возможных структур цифрового фильтра, которые располагаются в ряд от прямой формы к лестничной AP (Lattice AR) (авторегрессионного значения), лестничной CC (Lattice MA – Moving Average) (скользящего среднего значения), и лестничной APCC (Lattice ARMA) (авторегрессионного и скользящего среднего значения).

Структура фильтра есть математический эквивалент операций, когда используется вычисление с плавающей запятой. Однако различные структуры могут иметь заметно отличающееся исполнение в варианте с фиксированной запятой и вести к различной сложности вычисления и использования памяти в реализациях с плавающей запятой или с фиксированной запятой. Выбор соответствующей структуры фильтра является критическим при проектировании цифрового фильтра, особенно для цифровых фильтров с фиксированной запятой, в которых точность коэффициентов фильтра и операций фильтрации более ограничена, чем для цифровых фильтров с плавающей запятой. Для получения информации о применении различных структур фильтра можно обратиться к подразд. 16.4.

## 16.1.5. Специальное проектирование цифрового фильтра

ВП Special Filter Design обеспечивает проектирование БИХ-фильтров пробки/пиковые (notch/peak), гребенчатых БИХ-фильтров, фильтров с максимально плоской АЧХ, узкополосных фильтров и компенсаторов групповой задержки. Можно обратиться к подразд. 16.5 для получения информации об использовании ВП Special Filter Design.

## 16.1.6. Проектирование фильтра с фиксированной запятой

ВП Fixed-Point Tools помогают квантовать коэффициенты фильтра, создавать по шаблону, моделировать и анализировать проект фильтра с фиксированной запятой. Имеется возможность сохранять окончательную информацию о фильтре с фиксированной запятой в коде  $C$  для реализации на сигнальных процессорах (DSP) или в коде LabVIEW для выполнения на NI ПЛИС (FPGA)-устройствах, использующих ВП DFD FXP Code Generator. ВП Fixed-Point Tools, встроенные в ИПЦФ, являются важной частью внедрения NI в системное развитие. Для получения информации о разработке фильтров с фиксированной запятой на основе ВП Fixed-Point Tools можно обратиться к подразд. 16.4.

### 16.1.7. Создание кода для ПЛИС (FPGA) и цифровых сигнальных процессоров (DSP)

Использование ВП DFD FXP Code Generator для автоматической генерации кодов C и кодов LabVIEW позволяет развернуть DSP- и NI FPGA-устройства.

Можно генерировать код LabVIEW, оптимизированный только для NI FPGA-устройства, на основе использования кодовой опции SCTL-Optimized LabVIEW FPGA. Для получения информации о производстве кода на основе ВП DFD FXP Code Generator можно обратиться к подразд. 16.4.

### 16.1.8. Проектирование многоскоростных цифровых фильтров

ВП Multirate Filter Design помогает проектировать, анализировать и реализовывать одноступенчатые многоскоростные фильтры, многоступенчатые многоскоростные фильтры, фильтры выделения боковой полосы, фильтры Найквиста, фильтры поднятого косинуса и многокаскадные накопительные гребенчатые (CIC) фильтры. Для получения информации об использовании ВП Multirate Filter Design можно обратиться к подразд. 16.6.

### 16.1.9. Всесторонние инструментальные средства анализа

Использование ВП Filter Analysis позволяет оценить характеристики фильтров, которые проектируются в ВП Filter Design. Можно также исследовать частотную характеристику, групповую задержку, задержку фазы, импульсную передаточную функцию, ступенчатый отклик и размещение нулей и полюсов передаточной функции на плоскости (на поле) комплексного переменного. Для общей информации об оценке характеристик цифровых фильтров, которые создаются с помощью Digital Filter Design Toolkit, можно обратиться к подразд. 16.2.

### 16.1.10. Заметки к пользовательской расширенной библиотеке анализа в LabVIEW (Current LabVIEW Advanced Analysis Library Users)

Расширенная библиотека средств анализа в LabVIEW (LabVIEW Advanced Analysis Library), которая является доступной через палитру **Analyze**, является комплектом ВП, которые выполняют измерения временных диаграмм, обеспечивают улучшение формы диаграмм, контроль формы диаграмм, генерацию сигналов, обработку сигналов, точечную оценку и другие математические вычисления.

Расширенная библиотека анализа содержит некоторые ВП для проектирования цифровых фильтров, подобных ВП в ИПЦФ (Digital Filter Design Toolkit). Например, ВП Butterworth Coefficients коэффициентов Баттерворта в расширенной библиотеке анализа (Advanced Analysis Library) подобен ВП DFD Butterworth Design, а ВП Parks-McClellan VI в расширенной библиотеке анализа подобен ВП DFD Remez Design.

Однако ВП в ИПЦФ обеспечивают большие возможности, включая, но не без ограничений, поддержку произвольных технических требований по фазе и амплитуде при проектировании фильтра с фиксированной запятой.

Хотя различные ВП имеют похожие функциональные возможности, получаемые результаты не могут быть совершенно одинаковыми, потому что различаются алгоритмы проектирования. Для получения информации об использовании ВП LabVIEW Analyze с ВП Digital Filter Design Toolkit можно обратиться к NI Web site в ni.com/info и ввести информацию с кодом dfdfinfo.

## 16.2. Основы проектирования цифровых фильтров

Этот подраздел содержит вводную информацию о проектировании цифрового фильтра для пользователей, которые, возможно, не имели специального обучения цифровой обработке сигналов (DSP) или обучения проектированию цифрового фильтра. Можно применять цифровые фильтры, не зная или не понимая большую часть математики по цифровой фильтрации. Ниже приведены общие приложения цифровых фильтров, стандартная терминология, принципы проектирования и методы анализа, для того чтобы проектировать и анализировать характеристики цифрового фильтра.

### 16.2.1. Применение цифровых фильтров

Фильтры – элементы обработки сигналов, которые изменяют спектр входного сигнала. Можно использовать фильтры для следующих приложений:

- уменьшение шума во входном сигнале, если шумовая мощность и мощность сигнала сконцентрированы на различных частотах. Например, можно использовать фильтр-пробку, чтобы уменьшить сетевую наводку 50 Гц (или 60 Гц для США) в сигнале;
- выделение спектральных составляющих одного сигнала относительно составляющих другого сигнала, основанное на различии по спектру и концентрации составляющих по частоте. Например, можно использовать полосовой фильтр, чтобы извлечь сигнал нужной радиостанции из широкополосного входного радиосигнала;
- изменение гармонического спектра входного сигнала. Например, можно использовать фильтр с частотной характеристикой типа А, чтобы обеспечить уровень сигнала на разных частотах, соответствующий характеристи-

ке чувствительности человеческого уха. Другой пример – можно использовать компенсационный фильтр, чтобы изменить величину и искажение фазы, вызванное передачей сигнала через линейный, инвариантный по времени канал связи.

Чтобы реализовать системы цифровой обработки сигналов, можно использовать арифметику с плавающей запятой или с фиксированной запятой. Хотя разработка фильтра с плавающей запятой обычно проще в процессе проектирования, в то же время реализация фильтра с фиксированной запятой часто приводит к меньшим затратам и более эффективна, чем фильтра с плавающей запятой.

Проекты фильтра с плавающей запятой являются более приемлемыми в приложениях, которые работают на настольных компьютерах, а проекты фильтра с фиксированной запятой являются часто более эффективными в специализированных приложениях, в которых важно снизить стоимость или потребляемую мощность.

## 16.2.2. Терминология цифрового фильтра

Когда начинается проектирование цифрового фильтра, необходимо сформулировать технические требования, которые определяют характеристики, необходимые в разрабатываемом фильтре. Ниже сформулированы определения характеристик цифровых фильтров, которые создаются на основе использования ИПЦФ.

## 16.2.3. Свойства фильтра

Фильтры, которые могут быть спроектированы на основе использования ИПЦФ:

- **Линейный фильтр** – процессы в таком фильтре описываются линейными уравнениями с постоянными коэффициентами. Сигналы на выводах линейных фильтров – линейные преобразования от входных сигналов. Нельзя использовать эти цифровые фильтры, чтобы исключить нелинейное искажение, вызванное прохождением сигнала через нелинейный ограничитель или узел ограничения цифровых отсчетов.
- **Инвариантный во времени фильтр** – частотная характеристика такого цифрового фильтра не зависит от времени. Альтернативными являются адаптивные фильтры, которые не рассматриваются ИПЦФ и могут изменять свою частотную характеристику во времени в ответ на различное изменяющееся во времени управляющее воздействие (параметрические системы, в которых процессы описываются дифференциальными уравнениями с переменными коэффициентами).
- **Причинный (каузальный) фильтр** – сигнал на выходе цифрового фильтра не может измениться в ответ на изменение входного сигнала *до начала* изменений входного сигнала. ИПЦФ создает только цифровые фильтры, которые реализуются как каузальные в приложениях, но можно использовать их в некоторых некаузальных приложениях. Например, обычная DSP-техника для линеаризации ФЧХ фильтра должна передать входной сигнал через

фильтр, изменить на обратный по времени входной сигнал и затем передать его через фильтр снова. В этом случае, фильтр действует как некаузальный.

- **Цифровой фильтр** – ожидаемый сигнал ввода фильтра есть ряд дискретных цифровых значений. Набор дискретных цифровых коэффициентов определяет частотную характеристику фильтра. Цифровые фильтры имеют много преимуществ перед аналоговыми фильтрами. Например, частотные характеристики цифровых фильтров вообще не зависят от изменения параметров электронных компонентов или шума источника электропитания или снижения его напряжения.

## 16.2.4. КИХ- и БИХ-фильтры

Можно проектировать и КИХ- и БИХ-цифровые фильтры, используя ИПЦФ:

- **КИХ-фильтр** – после того как входной сигнал становится нулевым, сигнал на выходе фильтра может быть ненулевым только для конечного числа дискретных отсчетов, затем они также становятся нулевыми.
- **БИХ-фильтр** – сигнал на выходе фильтра может быть ненулевым бесконечно после того, как обнуляется входной сигнал.

Выбор между КИХ- и БИХ-фильтрами затрагивает процесс проектирования и реализации фильтра, как объяснено ниже.

## 16.2.5. Математические определения

В области времени процесс фильтрации для КИХ-фильтров может быть определен математически, для него свертка последовательности  $N + 1$  коэффициентов фильтра с последовательностью входных данных выборки  $x[n]$  есть

$$y[n] = \sum_{k=0}^N b_k x[n-k], \quad (16.1)$$

где  $y[n]$  – выходной отсчет фильтра;  $N$  – порядок фильтра. Число коэффициентов для КИХ-фильтра определено как число отсчетов входного сигнала и равно  $N + 1$ .

Если единственное ненулевое значение появляется на входе фильтра, а все последующие входные отсчеты являются нулевыми, выходные отсчеты фильтра в уравнении (16.1) становятся нулевыми после того, как фильтр обработает  $N + 1$  отсчетов входной выборки данных. Поскольку продолжительность ненулевого отклика в этом случае обязательно конечна для конечного числа  $N + 1$ , фильтр по уравнению (16.1) есть КИХ-фильтр.

В 1962 г. Чарльз Рейдер и Бернард Гоулд [22] в MIT Лаборатории Линкольна, разработали цифровой вокодер (кодер голоса) и предложили идею использования рекурсивных цифровых фильтров. Это новшество привело к БИХ-фильтру, у которого отклик в области времени определяется следующим уравнением:

$$\sum_{i=0}^M a_i y[n-i] = \sum_{k=0}^N b_k x[n-k], \quad (16.2)$$

где  $N$  – порядок числителя передаточной функции, и  $M$  – порядок ее знаменателя. Фильтр по уравнению (16.2) оперирует входным отсчетом  $x[n]$  и предыдущими входными отсчетами  $x[n - k]$  и выводит  $y[n]$  и предыдущие  $y[n - i]$  выходные отсчеты. Импульсная передаточная функция фильтра в этом случае бесконечна по времени, потому что фильтр может генерировать ненулевые выходные отсчеты далеко в будущем в ответ на единственную ненулевую входную выборку.

Для большинства приложений фильтрации КИХ-фильтры достаточны. Вообще КИХ-фильтры шире используются ввиду их высокой точности, и они более устойчивы в цифровой форме. Однако в некоторых случаях КИХ-фильтры становятся непрактично большими. Так, если есть необходимость иметь в большом количестве коэффициенты, например для КИХ-фильтра сотни или более, то КИХ-фильтр будет слишком труден или дорог при реализации, потому что проектирование может потребовать большего объема памяти, большей величины мощности питания, большего значения времени обработки и больше затрат технического времени для оптимизации проекта.

## 16.2.6. Различие КИХ- и БИХ-фильтров

Первое различие между КИХ- и БИХ-фильтрами состоит в том, что импульсная характеристика является конечной или бесконечной соответственно. Это различие вряд ли будет важным при проектировании фильтра, но КИХ- и БИХ-фильтры имеют и другие различия, которые могли бы повлиять на проект. Например, реализация КИХ-фильтра обычно требует большего количества операций умножения и суммирования, чем БИХ-фильтра с подобной задачей фильтрации, из-за того, что архитектура вычислительной системы часто лучше подходит для выполнения КИХ-фильтрации, причем скорость вычисления в БИХ-фильтре – не обязательно больше, чем в КИХ-фильтре. В табл. 16.1 сравниваются свойства каузальных КИХ- и БИХ-цифровых фильтров.

Таблица 16.1. Свойства КИХ- и БИХ-фильтров

Свойства	КИХ-фильтр	БИХ-фильтр
Возможность приближения ФЧХ к линейной	Возможно	Невозможно
Стабильность	Всегда стабильна	Условно стабильна
Фиксированная запятая	Легко выполняема	Могут быть сложности при выполнении
Вычислительная сложность	Больше вычислений	Меньше вычислений
Обычно заданная точность	Точность ниже заданной	Точность выше заданной
Нулевые по входу	Нельзя производить	Можно производить
предельные циклы*	предельные циклы	предельные циклы

\* «Ограничение нулевых по входу предельных циклов определяется в том смысле, что выходной сигнал может продолжить колебаться неопределенно периодическим образом, в то время как вход остается равным нулю. Это последствия или нелинейного ограничивающего устройства в петле обратной связи БИХ-фильтра, или выход за пределы условий» (Оппенгейм и Шафер)

## 16.2.7. Свойства КИХ- и БИХ-фильтров.

### Технические требования для цифровых фильтров

Чтобы проектировать цифровой фильтр с помощью ИПЦФ, необходимо определить тип фильтра, подобрать частоту дискретизации, свойства фильтра и выбрать метод проектирования.

## 16.2.8. Тип фильтра

Можно создать ФНЧ, ФВЧ, полосовой и ЗФ:

- фильтр ФНЧ пропускает составляющие спектра с низкими частотами и уменьшает амплитуды составляющих с высокими частотами;
- фильтр ФВЧ пропускает составляющие спектра с высокими частотами и уменьшает амплитуды составляющих с низкими частотами;
- полосовой фильтр пропускает составляющие в заданном диапазоне частот;
- фильтр ЗФ уменьшает амплитуды составляющих в заданном диапазоне частот и пропускает все составляющие с частотами за пределами этого диапазона.

## 16.2.9. Частота дискретизации

Символ  $f_s$  обозначает частоту дискретизации, определяющую предполагаемую скорость, в соответствии с которой получена выборка входного сигнала для фильтра. Половину частоты дискретизации называют частотой Найквиста (Nyquist). В этом инструментарии по умолчанию устанавливается значение частоты, равное 1, которая является нормализованной частотой дискретизации.

## 16.2.10. Технические требования к фильтру

Для большинства цифровых фильтров обычно проектируется цифровая передаточная функция. Технические требования для цифрового фильтра обычно включают искомые АЧХ, ФЧХ и допустимое отклонение для каждой. На рис. 16.1 представлены технические требования для ФНЧ.

На рис. 16.1 использованы следующие обозначения: passband edge frequency – граница полосы пропускания; stopband edge frequency – граница полосы задержания; stopband attenuation – затухание в полосе задержания; transition band – переходная полоса;  $\delta_p$  – уровень пульсаций АЧХ в полосе пропускания;  $\delta_s$  – уровень подавления в полосе задержания.

При проектировании фильтра необходимо определить следующие параметры в технических требованиях к фильтру: коэффициент передачи в полосе пропускания, максимальные пульсации АЧХ в полосе пропускания, граничные частоты полосы пропускания, граничную частоту полосы задержания и минимальное ослабление в полосе задержания (обратно последнему – максимум коэффициента передачи в полосе задержания). Уровень пульсаций АЧХ в полосе пропускания

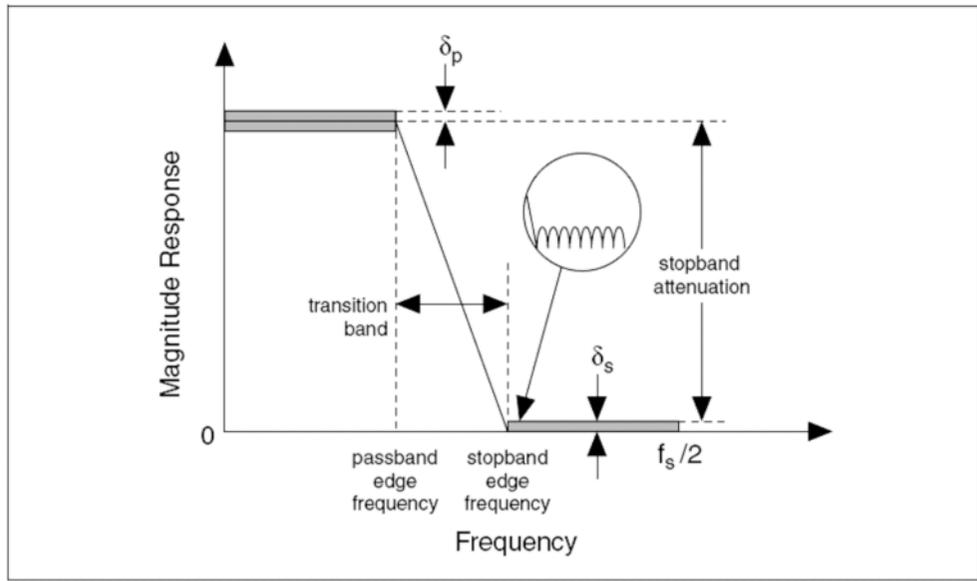


Рис. 16.1. Технические требования к ФНЧ

определяет максимально допустимое отклонение  $\delta_p$  коэффициента передачи от желательного в полосе пропускания. Минимальное ослабление в полосе задержания указывает максимально допустимое отклонение  $\delta_s$  коэффициента передачи от нуля.

Необходимо также обращать внимание на полосу перехода между полосой пропускания и полосой задержания.

В идеальном проекте цифровой фильтр имеет заданный коэффициент передачи в полосе пропускания и нулевой коэффициент передачи ( $-\infty$  дБ) в полосе задержания. В реальном исполнении конечная полоса перехода между полосой пропускания и полосой задержания, которая известна как полоса перехода, всегда существует. Коэффициент передачи фильтра в полосе перехода не определен. Этот коэффициент обычно изменяется постепенно в пределах полосы перехода от 1 (0 дБ) в полосе пропускания до 0 ( $-\infty$  дБ) в полосе задержания.

На рис. 16.2–16.4 представлены АЧХ ФВЧ, полосового и ЗФ.

### 16.2.11. Методы проектирования

ИПЦФ обеспечивает несколько методов проектирования, включая эллиптический (Elliptic), Чебышева (Chebyshev), инверсный Чебышева (Inverse Chebyshev) и Баттерворта (Butterworth) для БИХ-цифровых фильтров. Каждая опция проектирования предлагает различные характеристики. Например, для фильтра Баттерворта характерна гладкая АЧХ на всех частотах и также монотонное уменьшение или увеличение в полосе перехода. Однако фильтры Баттерворта не всегда

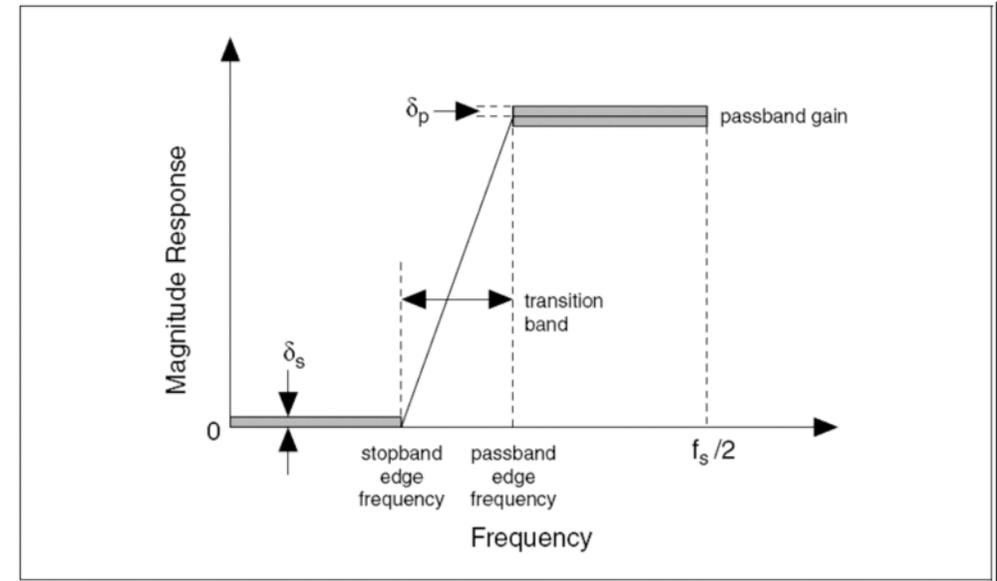


Рис. 16.2. Технические требования к ФВЧ

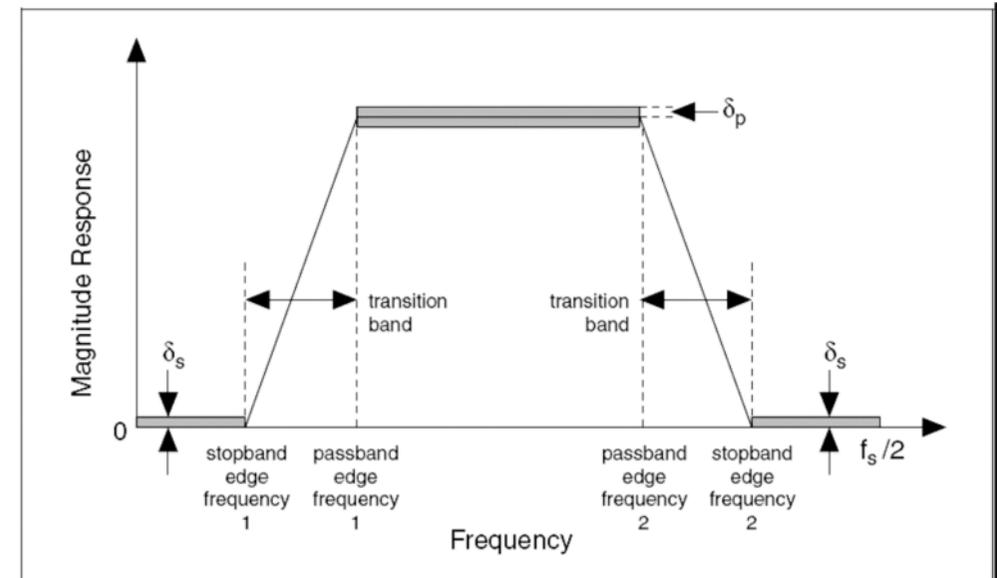


Рис. 16.3. Технические требования к ПФ

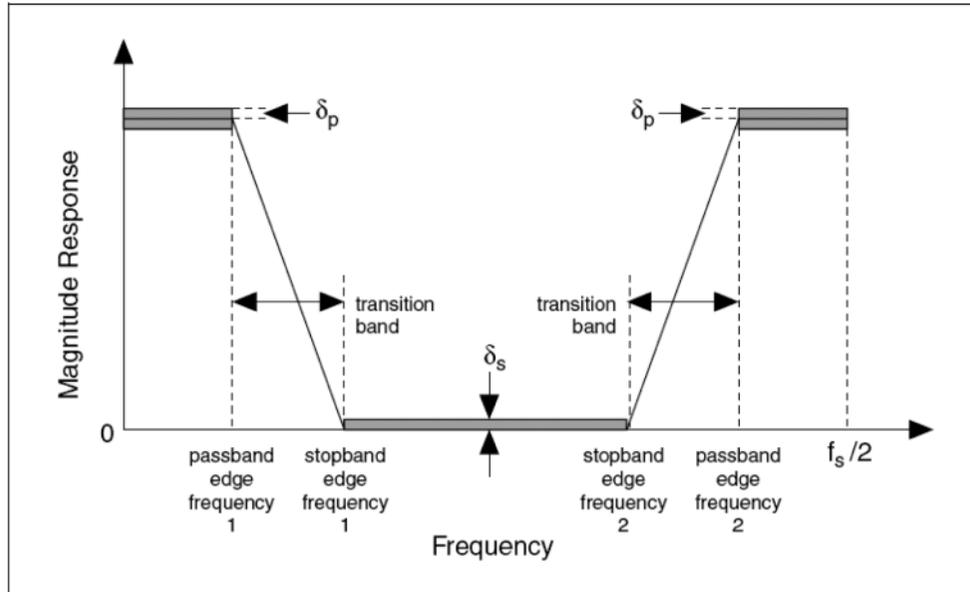


Рис. 16.4. Технические требования к ЗФ

обеспечивают приемлемое точное приближение к идеальной АЧХ фильтра, потому что этот фильтр имеет медленный склон характеристики в полосе перехода. Если есть необходимость в более крутом склоне АЧХ от полосы пропускания к полосе задержания, чем фильтр Баттерворта может обеспечить, надо использовать фильтр Чебышева, инверсию Чебышева, или эллиптический проект. Для получения окончательной информации о различных методах проектирования можно обратиться к **Справке LabVIEW (LabVIEW Help)** в ВП проектирования фильтра (Filter Design Vis).

Инструментарий также обеспечивает несколько методов проектирования, включая метод окна Кайзера (Kaiser Window) и алгоритм Ремеза (Remez), для КИХ-цифровых фильтров. Методы, основанные на работе с окнами, являются классическими, в то время как алгоритм Ремеза усложнен, но обеспечивает оптимальные результаты.

Можно использовать ВП DFD Classical Filter Design Express, чтобы экспериментировать в интерактивном режиме с методами проектирования. Для получения дополнительной информации о методах дизайна и использования DFD Classical Filter Design Express можно обратиться к подразд. 16.3.

## 16.2.12. Анализ цифрового фильтра

После того как цифровой фильтр будет спроектирован, необходимо анализировать характеристики цифрового фильтра, оценивая АЧХ, ФЧХ и групповую задержку, импульсную передаточную функцию или положение полюсов и нулей.

ИПЦФ содержит ВП Filter Analysis, чтобы помочь оценить характеристики фильтра. Ниже представлена информация о поле полюсов и нулей при использовании ВП DFD Plot Pole-Zero. Для получения информации о ВП Filter Analysis можно обратиться к **Справке LabVIEW (LabVIEW Help)**.

## 16.2.13. Полюса и нули

Z-преобразование (*z-transform*) передаточной функции КИХ-фильтра определено как

$$H(z) = \sum_{k=0}^N b_k z^{-k} = K \prod_{k=1}^N (1 - z_k z^{-1}), \quad (16.3)$$

где  $b_k$  – коэффициенты фильтра;  $N$  – порядок КИХ-фильтра;  $K$  – коэффициент передачи. В выражении (16.3)  $z_k$  – корни полинома  $H(z)$ .

$H(z_k) = 0$  для всех  $z_k$ , поэтому  $z_k$  называют нулями передаточной функции  $H(z)$  фильтра. Число нулей в передаточной функции фильтра должно равняться порядку фильтра и равно  $N$ .

Точно так же z-преобразование БИХ-фильтра по выражению (16.2) определено как

$$H(z) = \frac{\sum_{k=0}^N b_k z^{-k}}{1 + \sum_{i=1}^M a_i z^{-i}} = K \frac{\prod_{k=1}^N (1 - z_k z^{-1})}{\prod_{k=1}^M (1 - p_k z^{-1})}, \quad (16.4)$$

где  $z_k$  и  $p_k$  представляют корни полинома числителя и полинома знаменателя соответственно;  $p_k$  – полюса БИХ-фильтра с передаточной функцией  $H(z)$ .

Следует заметить, что  $H(z)$  БИХ-фильтров имеет полюса и нули, а  $H(z)$  КИХ-фильтров имеет только нули.

## 16.2.14. Поле нулей и полюсов

С математической точки зрения, поле нулей и полюсов и комплексная передаточная функция обеспечивают одну и ту же информацию. На основе комплексной передаточной функции можно получить поле нулей и полюсов, и наоборот, из поля нулей и полюсов, можно определить комплексную передаточную функцию.

Поле нулей и полюсов, в частности для БИХ-фильтра, иллюстрирует рис. 16.5. Полуокруг соответствует условию  $|z| = 1$  или кругу единичного радиуса (модуля). Маленькие окружности по полуокругу представляют положение нулей. Каждый знак ( $\infty$ ) представляет положение полюса.

Поле нулей и полюсов и передаточная функция характеризуют цифровые фильтры по следующим возможностям:

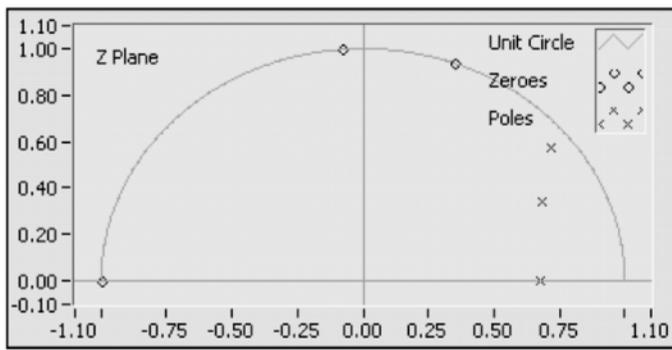


Рис. 16.5. Типичное поле нулей и полюсов

- можно определить, устойчив ли цифровой фильтр, т. е., ограничен ли по величине отсчетов выходной сигнал цифрового фильтра для всех возможных ограниченных отсчетов входного сигнала. Необходимое и достаточное условие для фильтров БИХ, чтобы быть устойчивым состоит в том, что все полюса должны находиться в круге единичного радиуса. В то же время КИХ-фильтры всегда устойчивы, потому что их передаточные функции не имеют полюсов;
- можно определить, достаточно ли близки пары нуль-полюс, чтобы эффективно уравновесить друг друга. Можно попробовать устранить близкие пары и проверить результирующую передаточную функцию. Меньше пар нуль-полюс означают меньше вычислений.

## Резюме

1. Чтобы проектировать цифровой фильтр, сначала необходимо ввести технические требования для фильтра, который нужно спроектировать. Технические требования включают тип фильтра, частоту дискретизации, пульсации АЧХ в полосе пропускания, полосу пропускания и граничные частоты, ослабление в полосе задержания и метод проектирования. Для получения информации о вводе технических требований для фильтра при использовании ВП DFD Classical Filter Design Express можно обратиться к подразд. 16.3.
2. Можно создать КИХ- и БИХ-фильтры, используя ИПЦФ. В большинстве случаев КИХ-фильтры лучше используют точность и более устойчивы в цифровой форме. Для получения информации о других характеристиках КИХ- и БИХ-фильтров, которые могли бы изменить проект фильтра, следует обратиться к табл. 16.1.
3. Можно создать ФНЧ, ФВЧ, полосовые, и ЗФ цифровые фильтры, используя несколько различных методов проектирования. Для получения информации о проектировании КИХ и БИХ-фильтров на основе использования

ВП Advanced FIR Filter Design и Advanced IIR Filter Design следует обратиться к **Справке LabVIEW (LabVIEW Help)**.

4. После того как спроектирован цифровой фильтр, необходимо проанализировать характеристики полученного цифрового фильтра, чтобы удостовериться, выполнены ли потребности применения фильтра. Для получения информации о ВП Filter Analysis обратиться к **Справке LabVIEW (LabVIEW Help)**.

## 16.3. Основы проектирования фильтра с плавающей запятой

В этом подразделе объясняется, как использовать ВП DFD Classical Filter Design Express для проектирования классических цифровых фильтров с плавающей запятой.

### 16.3.1. Типичный процесс проектирования цифрового фильтра с плавающей запятой

На рис. 16.6 представлен процесс проектирования цифрового фильтра с плавающей запятой. Сначала необходимо разработать технические характеристики фильтра. Затем анализируются характеристики полученного фильтра, чтобы определить, выполняет ли фильтр требования системы. Если фильтр не выполняет эти требования, можно изменить технические характеристики и повторить процесс. После того как соответствующий фильтр будет спроектирован, можно использовать фильтр в системе обработки сигналов.

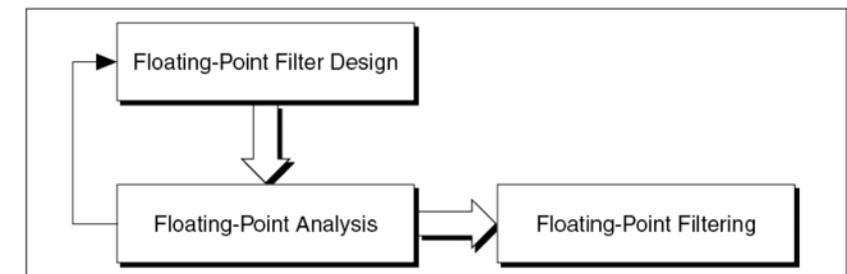


Рис. 16.6. Процесс проектирования цифрового фильтра с плавающей запятой

На рис. 16.6 использованы следующие обозначения:  
 Floating-Point Filter Design – проект фильтра с плавающей (точкой) запятой;  
 Floating-Point Analysis – анализ фильтра с плавающей запятой;  
 Floating-Point filtering – фильтрация с плавающей запятой.

Для преодоления специфической проблемы при проектировании можно использовать несколько различных методов и выбрать тот, который приведет к приемлемому результату. Чтобы достичь лучших результатов, можно экспериментировать на основе нескольких различных подходов.

### 16.3.2. Проектирование фильтров с плавающей запятой

Основная цель проектирования цифрового фильтра состоит в том, чтобы вычислить набор коэффициентов, так что характеристики полученного цифрового фильтра выполняют набор predetermined технических характеристик фильтра. Ниже описано, как использовать ВП DFD Classical Filter Design Express, чтобы проектировать цифровые фильтры с плавающей запятой.

### 16.3.3. Ввод технических характеристик фильтра

Можно использовать ВП DFD Classical Filter Design Express, чтобы в интерактивном режиме конфигурировать классический цифровой фильтр. После того как будет найдена палитра **Functions** ВП Express и размещена в блок-схеме, диалоговое окно **Configure Classical Filter Design** появляется на лицевой панели, как показано на рис. 16.7.

В диалоговом окне технических характеристик можно выбрать тип фильтра и проектировать метод на опускающемся меню, затем ввести технические характеристики фильтра или через основанный на тексте интерфейс цифрового управления на левой стороне диалогового окна технических характеристик, или через интерактивный графический интерфейс на правой стороне диалогового окна технических характеристик. Результаты эквивалентны.

Для получения информации о выборе технических характеристик фильтра можно обратиться к подразд. 16.2, затем к подразд. 16.6 и 16.7 для получения информации о вводе технических характеристик фильтра в диалоговое окно, показанное на рис. 16.7.

### 16.3.4. Ввод технических характеристик фильтра в цифровом управлении, основанном на текстовом интерфейсе

Классические технические характеристики цифрового фильтра включают диапазон частот и ограничения по уровню пульсаций АЧХ. Можно определить максимальное допустимое отклонение  $\delta_p$  коэффициента передачи от единицы в полосе пропускания как уровень пульсаций в полосе пропускания (**Passband Ripple**) в цифровом управлении ВП. Можно определить максимальное допустимое отклонение  $\delta_s$  от нулевого значения в полосе задержания (**Stopband Attenuation**) в цифровом управлении.

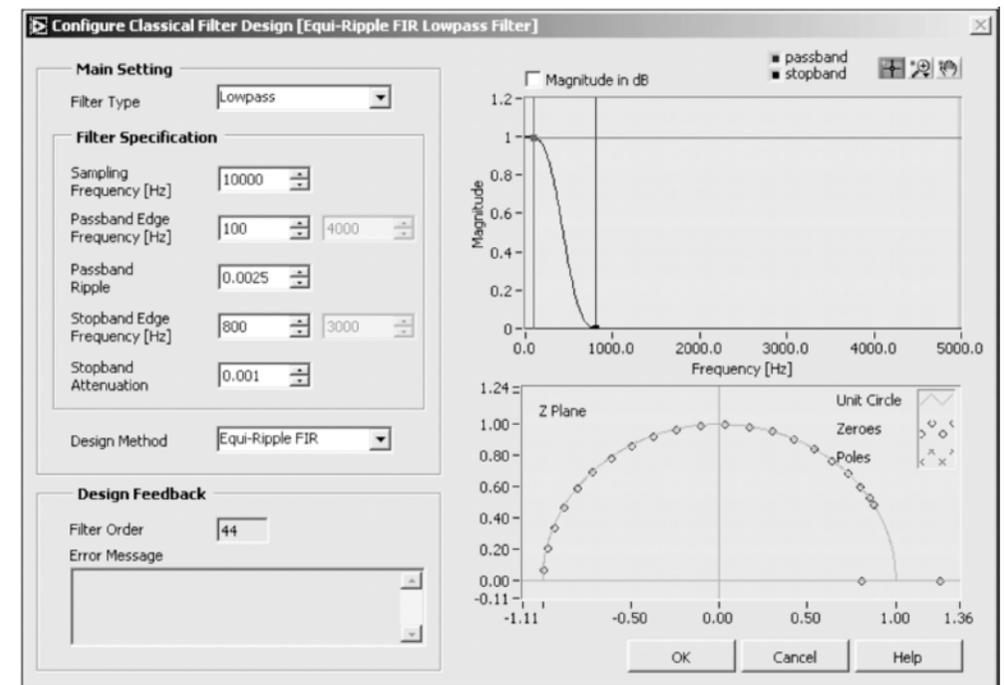


Рис. 16.7. Диалоговое окно технических характеристик ВП DFD Classical Filter Design Express

Можно установить эти отклонения или в логарифмическом, или в линейном масштабе. В ВП DFD Classical Filter Design Express используется логарифмический масштаб по умолчанию.

Чтобы использовать линейный масштаб, необходимо удалить пометку из **Magnitude in dB** диалогового окна технических характеристик.

Отношения между логарифмическим и линейным масштабами пульсаций АЧХ в полосе пропускания (Passband ripple) и уровнем затухания в полосе задержания (Stopband attenuation) представлены следующими формулами:

$$\text{Passband ripple (dB)} = -20 \log_{10}(1 - \delta_p);$$

$$\text{Stopband attenuation (dB)} = -20 \log_{10}(\delta_s).$$

Например, если коэффициент передачи в полосе пропускания слегка колеблется и уровень пульсаций АЧХ равен 0,01 дБ (т. е.  $0,01 \text{ дБ} = -20 \log_{10}(1 - \delta_p)$ ), то  $\delta_p = 0,00115$ . Точно так же, если ослабление в полосе задержания равно 60 дБ (т. е.  $60 \text{ дБ} = -20 \log_{10}(\delta_s)$ ), то  $\delta_s = 0,001$ .

### 16.3.5. Ввод технических характеристик фильтра в графическом интерфейсе

Правая сторона рис. 16.7 отображает частотную характеристику разработанного цифрового фильтра. Ось значений коэффициента передачи может быть выражена в линейном или логарифмическом масштабах. Чтобы использовать линейный масштаб, надо удалить пометку в переключателе **Magnitude in dB**, или сохранить пометку в переключателе **Magnitude in dB**, чтобы поддержать логарифмический масштаб. Ось частоты (**Frequency**), в герцах, охватывает диапазон от 0 до половины частоты дискретизации, которая является частотой Найквиста.

График **Magnitude** содержит набор курсоров, которые можно использовать, чтобы определить полосу пропускания и полосу задержания. Для изменения полосы пропускания можно использовать курсор **Passbands**. Расстояние между единицей и горизонтальным курсором полосы пропускания (**Passbands**) определяет максимальный уровень пульсаций АЧХ в полосе пропускания. Местоположение вертикального курсора **Passbands** указывает граничную частоту полосы пропускания. Курсоры **Stopband** работают тем же самым способом для установки характеристик полосы задержания (**Stopbands**).

При работе с курсорами, чтобы установить технические характеристики фильтра, необходимо придерживаться набора правил для ввода допустимых технических характеристик. Если любое правило нарушено, ВП DFD Classical Filter Design Express выведет на дисплее сообщение в индикаторе **Error Message** диалогового окна **Configure Classical Filter Design** с предложениями переустановить курсоры.

Правила установки следующие:

- устанавливать горизонтальные курсоры только в диапазоне (0, 1) в линейном масштабе или (-inf, 0 дБ) в логарифмическом масштабе;
- устанавливать горизонтальный курсор полосы пропускания выше горизонтального курсора полосы задержания;
- нельзя обменивать позиции вертикальных курсоров.

### 16.3.6. Выбор метода проектирования

После того как введены заданные технические характеристики цифрового фильтра в цифровое управление или графический интерфейс, необходимо выбрать метод проектирования. ВП DFD Classical Filter Design Express обеспечивает следующие методы дизайна на основе КИХ-фильтрации:

- окно Кайзера (Kaiser Window);
- окно Дольфа – Чебышева (Dolph – Chebyshev Window);
- КИХ с равномерной пульсацией в полосе пропускания (Equi-Ripple FIR).

Метод окна Кайзера и метод окна Дольфа – Чебышева проще при проектировании, чем метод Equi-Ripple FIR, но этот метод (Equi-Ripple FIR) приводит к оптимальным фильтрам и часто производит лучшие результаты для большинства проблем проектирования КИХ-фильтров.

В дополнение к методам на основе КИХ-фильтрации ВП DFD Classical Filter Design Express поддерживает следующие основанные на КИХ-фильтрации методы проектирования:

- Баттерворта (Butterworth);
- Чебышева (Chebyshev);
- инверсный Чебышева (Inverse Chebyshev);
- эллиптический (Elliptic).

На рис. 16.8 представлены частотные характеристики типичного ФНЧ, разработанного четырьмя основанными на КИХ-фильтрации методами. В каждой характеристике порядок фильтра равен 5 и в числителе, и в знаменателе передаточной функции.

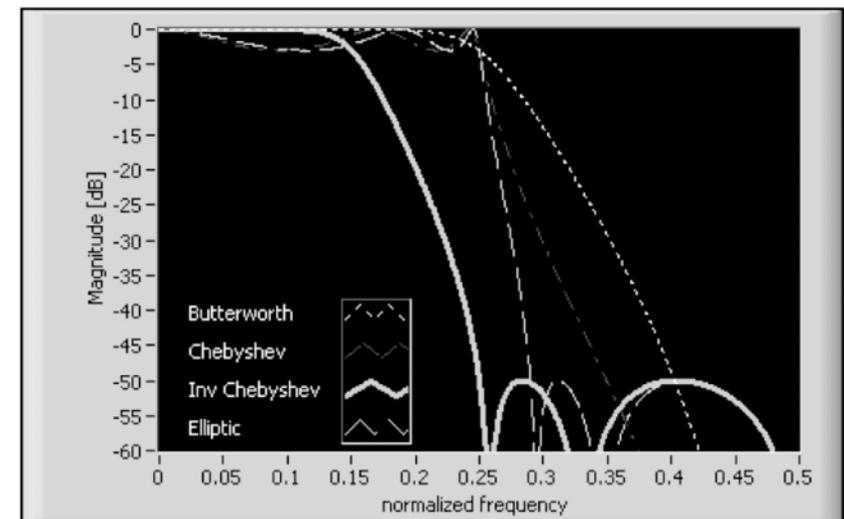


Рис. 16.8. Сравнение частотных характеристик классических БИХ-фильтров

В табл. 16.2 приведены основные особенности четырех основанных на КИХ-фильтрации методов проектирования, так что можно определить, какой метод проектирования БИХ-фильтра можно использовать в заданных условиях.

Таблица 16.2. Сравнение классических БИХ-фильтров

БИХ-фильтр	Неравномерность в полосе пропускания	Неравномерность в полосе задержания	Переходная полоса для фиксированного порядка	Порядок для заданных параметров фильтра
Баттерворта	Нет	Нет	Широкая	Высокий
Чебышева	Есть	Нет	Узкая	Низкий
Инверсный Чебышева	Нет	Есть	Узкая	Низкий
Эллиптический	Есть	Есть	Узкая	Низкий

Когда проектируется цифровой фильтр с помощью ВП DFD Classical Filter Design Express, форма частотной характеристики управляется выбранным методом проектирования и заданными техническими характеристиками фильтра. Нельзя изменить ФЧХ даже в том случае, когда ФЧХ для фильтров, разработанных в этом ВП на основе методов КИХ-фильтрации, являются линейными. Если необходимо определить АЧХ и ФЧХ, можно использовать ВП Advanced IIR Filter Design, Advanced FIR Filter Design, или Special Filter Design. Для получения информации о проектировании специальных фильтров можно обратиться к подразд. 16.5.

### 16.3.7. Анализ проекта фильтра

После того как введены заданные технические характеристики цифрового фильтра, можно немедленно анализировать полученные характеристики на появляющемся диалоговом окне параметров, оценивая частотную характеристику, график нулей и полюсов и порядок фильтра.

### 16.3.8. Частотная характеристика

Комплексная передаточная функция определяется как  $H(e^{j2\pi f})$ , а АЧХ задается ее модулем  $|H(e^{j2\pi f})|$ . Для дискретно-временных систем  $H(e^{j2\pi f})$  является периодической функцией с периодом  $f_s$ . Для цифровых фильтров с реальными коэффициентами АЧХ симметрична относительно значений частоты:  $0, \pm f_s, \pm 2f_s, \dots$  Поэтому передаточная функция рассчитывается только для интервала  $[0, f_s / 2]$ , т. е. для частот между 0 и частотой Найквиста. На графике АЧХ в диалоговом окне **Configure Classical Filter Design** показана толстой вертикальной линией (см. рис. 16.7), чтобы указать местоположение  $f_s / 2$ .

### 16.3.9. Поле нулей и полюсов

В диалоговом окне **Configure Classical Filter Design** знак (x) представляет полюс, а кружок представляет нуль. Полюса каузального устойчивого фильтра должны быть в круге единичного модуля. Для получения дополнительной информации о полюсах и нулях можно обратиться к подразд. 16.2.

### 16.3.10. Определение порядка фильтра

ВП DFD Classical Filter Design Express автоматически вычисляет минимальный порядок фильтра, необходимый для выполнения заданных характеристик фильтра и отображает значение порядка в индикаторе **Filter Order**. Учитывая те же самые характеристики, различные алгоритмы проектируют цифровые фильтры с различными порядками фильтра. Можно оценить вычислительную сложность и стоит до некоторой степени фильтр, основанный на порядке. Если же имеются строгие требования для системы, необходимо установить порядок фильтра, который может помочь определить, приемлем ли спроектированный фильтр.

## 16.3.11. Пример. Проектирование ФНЧ цифрового фильтра согласно параметрам

Необходимо выполнить следующие шаги, чтобы экспериментировать с ФНЧ с плавающей запятой, разработанным с помощью ВП DFD Classical Filter Design Express для следующих установленных параметров и характеристик:

Параметры и характеристики	Значения
Тип фильтра (Filter Type)	ФНЧ Lowpass
Метод проектирования (Design Method)	Равномерные пульсации Equi-Ripple
Частота дискретизации (Sampling Frequency)	10 кГц 10 kHz
Полоса пропускания (Passband Frequency)	От 0 до 100 Гц 0 to 100 Hz
Пульсации в полосе задержания (Passband Ripple)	$\pm 0,0025$
Полоса задержания (Stopband Frequency)	От 800 Гц до 5 кГц 800 Hz to 5 kHz
Затухание в полосе (Stopband Attenuator)	60 дБ 60 dB

Последовательность операций:

1. Открыть ВП Lowpass Step 1\_Design Lowpass, расположенный в директории: examples\Digital каталоге Filter Design\CaseStudy1.
2. Исследовать блок-диаграмму, показанную на рис. 16.9.

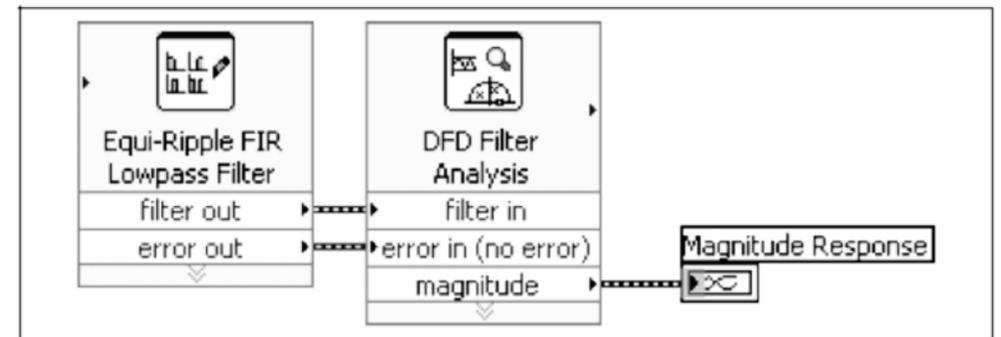


Рис. 16.9. Блок-диаграмма ВП Lowpass Step 1\_Design Lowpass

3. Дважды щелкнуть на блоке Equi-Ripple FIR Lowpass Filter ВП Express. Появляется диалоговое окно **Configure Classical Filter Design**.

**Примечание.** Этот ВП Express был создан с помощью ВП DFD Classical Filter Design Express. После того как введены параметры и характеристики, в диалоговом окне конфигурации можно нажать кнопку **OK** на имени прибора Express VI, в результате на блок-диаграмме изменяется надпись, отражающая тип фильтра, который разработан. В этом примере название изменено на *Equi-Ripple FIR Lowpass Filter*.

4. Проверить, что параметры в диалоговом окне технических характеристик соответствуют параметрам таблицы, перечисленным выше. Поскольку ослабление 60 дБ в полосе задержания эквивалентно коэффициенту передачи в линейном масштабе 0,001, цифровое управление **Stopband Attenuation** содержит число 0,001, и пометка удалена из **Magnitude in dB**, чтобы перейти на линейный масштаб.
5. Можно использовать графики АЧХ и поле нулей и полюсов (pole-zero), чтобы убедиться, что характеристики разработанного фильтра удовлетворяют техническим требованиям.
6. Нажать кнопку **OK**, чтобы закрыть диалоговое окно **Configure Classical Filter Design**.
7. Запустить ВП и исследовать АЧХ полученного фильтра на лицевой панели.

### 16.3.12. Пример. Фильтрация

Необходимо пройти следующие шаги, чтобы использовать ФНЧ в задаче фильтрации.

1. Открыть ВП Lowpass\_Step 2\_Perform Lowpass Filtering, расположенный в каталоге examples\Digital Filter Design\CaseStudy1.
2. Исследовать блок-диаграмму ВП, показанную на рис. 16.10.

В этом примере на входы ВП DFD Filtering поступают как выходной сигнал ФНЧ цифрового фильтра (Equi-Ripple FIR Lowpass Filter), созданного в ВП DFD Classical Filter Design Express, так и сигнал, который представляется обычным сигналом (Multitone Generator.vi) в системе. ВП DFD Filtering выводит отфильтрованный сигнал на лицевую панель.

3. Необходимо переключиться к лицевой панели и запустить ВП. Заметим, что на лицевой панели ВП имеется исходный входной сигнал (Signal In) и отфильтрованный сигнал (Signal Out) в виде графиков, как показано на рис. 16.11.

## Резюме

1. Использование ВП DFD Classical Filter Design Express позволяет проектировать цифровой фильтр быстро и в интерактивном режиме.
2. Можно ввести заданные технические характеристики фильтра в интерактивное диалоговое окно **Configure Classical Filter Design**, которое открывается автоматически, когда загружается ВП DFD Classical Filter Design Express в виде блок-схемы, или двойной щелчок левой кнопки мыши на выведенной блок-диаграмме ВП DFD Classical Filter Design Express.

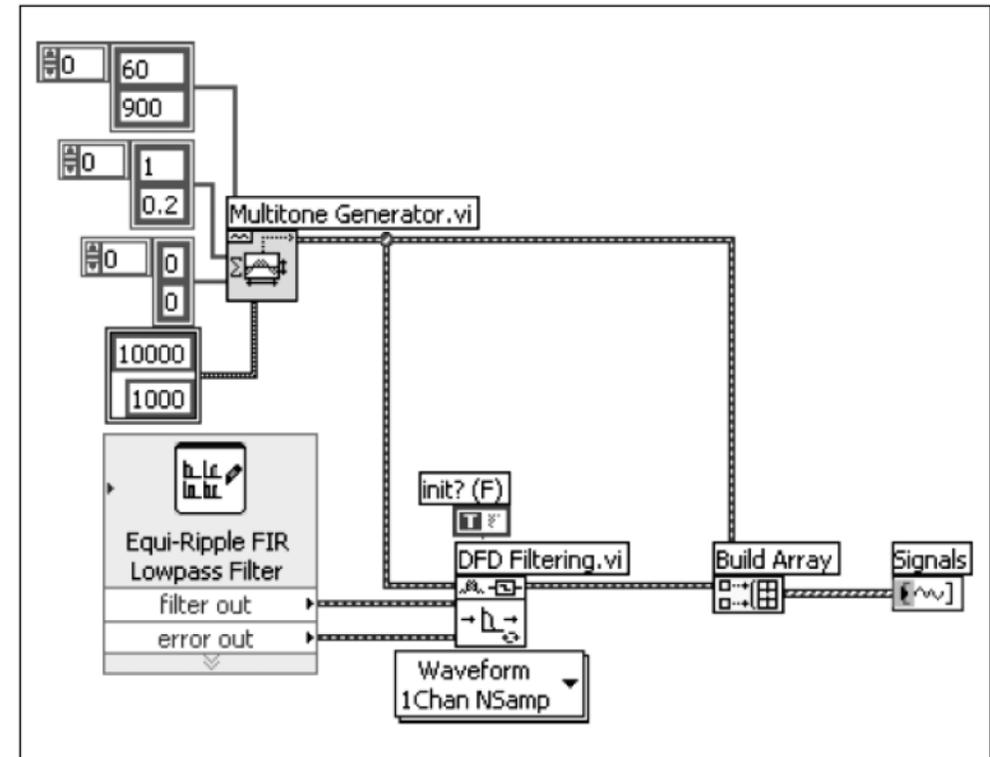


Рис. 16.10. Блок-диаграмма ВП Lowpass Step 2\_Perform Lowpass Filtering

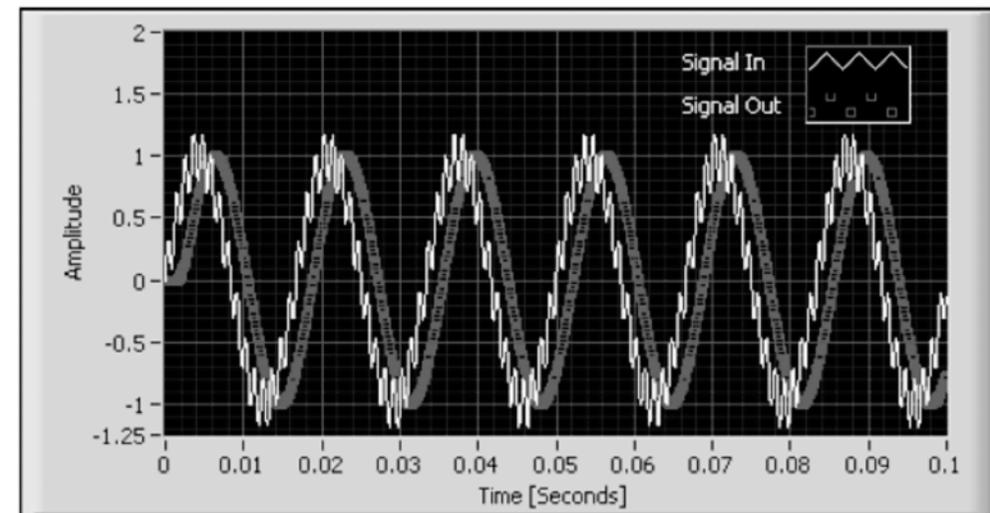


Рис. 16.11. Исходный и отфильтрованный сигналы

3. Можно использовать основанный на тексте интерфейс цифрового управления или графический интерфейс прибора Express VI, чтобы ввести заданные технические характеристики. Оба интерфейса обеспечивают одинаковый результат и можно ясно видеть зависимости между задаваемыми пользователем параметрами и характеристиками полученного цифрового фильтра с плавающей запятой в виде АЧХ и поля распределения нулей и полюсов.
4. Виртуальный прибор DFD Classical Filter Design Express.vi обеспечивает три метода проектирования на основе КИХ-фильтрации и четыре метода дизайна, основанных на БИХ-фильтрации. Можно в интерактивном режиме экспериментировать с этими методами проектирования в диалоговом окне **Configure Classical Filter Design**.
5. Если необходимо определить АЧХ и ФЧХ или проектировать специальный фильтр, надо использовать ВП Advanced IIR Filter Design, Advanced FIR Filter Design, или Special Filter Design.

Для получения информации о проектировании специальных фильтров можно обратиться к подразд. 16.5.

## 16.4. Основы проектирования фильтра с фиксированной запятой

В этом подразделе объясняется, как разработать цифровой фильтр с фиксированной запятой на основе библиотеки фильтров с плавающей запятой, используя ВП DFD Fixed-Point Tools. Предполагается, что читатель знаком с содержанием подразд. 16.3, а также с методикой создания цифровых фильтров с плавающей запятой на основе ИПЦФ.

### 16.4.1. Типичный процесс проектирования цифрового фильтра с фиксированной запятой

Базовые системы обработки сигналов с фиксированной запятой типа цифровых сигнальных процессоров с фиксированной запятой (DSPs) и программируемые в условиях эксплуатации вентильные матрицы (ПЛИС) (FPGAs) являются обычно более эффективными и менее дорогими, чем альтернативные системы с плавающей запятой. Однако системы с фиксированной запятой вообще трудны в проектировании.

Например, если необходимо иметь дело с коэффициентами фильтра более грубого квантования, то обычно представляют системы с фиксированной запятой.

Чтобы проектировать цифровой фильтр с фиксированной запятой, сначала необходимо спроектировать фильтр с плавающей запятой, названный рекомендуемым фильтром, который обеспечивает заданные параметры.

Затем необходимо изменить фильтр с плавающей запятой, чтобы приспособить ограниченную точность заданной платформы, в то же время попытаться вы-

полнить заданные параметры. На рис. 16.12 показан процесс проектирования фильтра с фиксированной запятой.

Здесь: Floating-point Filter Design – проект фильтра с плавающей запятой (точкой); Floating-point Analysis – анализ фильтра с плавающей запятой; Floating-point Filtering – фильтрация с плавающей запятой; Structure Selection – структурная селекция; Fixed-point modeling – моделирование с фиксированной запятой; Fixed-point Analysis – анализ с фиксированной запятой; Simulation – имитация; Fixed-point Target (DSP or FPGA – адрес проекта с фиксированной запятой (цифровой сигнальный процессор или ПЛИС).

Заметим, что после того, как будет спроектирован и проанализирован рекомендуемый фильтр с плавающей запятой, выбирают структуру фильтра и моделируют проект с фиксированной запятой. Преобразование фильтра с плавающей запятой к форме с фиксированной запятой может значительно изменить характеристики и работу фильтра. Необходимо анализировать фильтр и имитировать процесс фильтрации с данными, подобными данным в фактической системе.

Арифметика с фиксированной запятой может иметь следующие вредные эффекты в работе фильтра:

- ухудшение отношения сигнал-шум из-за уменьшенной точности отсчетов входного сигнала;
- искажения частотной характеристики из-за ограниченного представления разрядности коэффициентов фильтра;
- ограничение периодов в БИХ-фильтрах из-за арифметики конечной точности;
- переполнение или отсечение из-за недостаточной разрядности в канале передачи данных.

Наконец, можно генерировать код для адресата с фиксированной запятой.

### 16.4.2. Осуществление фильтров с фиксированной запятой

Необходимо завершить следующие шаги, чтобы преобразовать рекомендуемый фильтр с плавающей запятой в фильтр с фиксированной запятой, как показано на рис. 16.12.

1. Выбор структуры выполнения для фильтра. Для получения информации о выборе и преобразовании структур фильтра можно обратиться ниже к пункту **Выбор структуры фильтра**.
2. Моделирование фильтра с фиксированной запятой. Чтобы преобразовать фильтр с плавающей запятой в фильтр с фиксированной запятой, необходимо квантовать коэффициенты фильтра с плавающей запятой, входной сигнал, выходной сигнал и результаты промежуточных операций. Квантование есть процесс приближения каждого значения с плавающей запятой к значению с фиксированной запятой, которое может использоваться в математических операциях с фиксированной запятой. Для получения

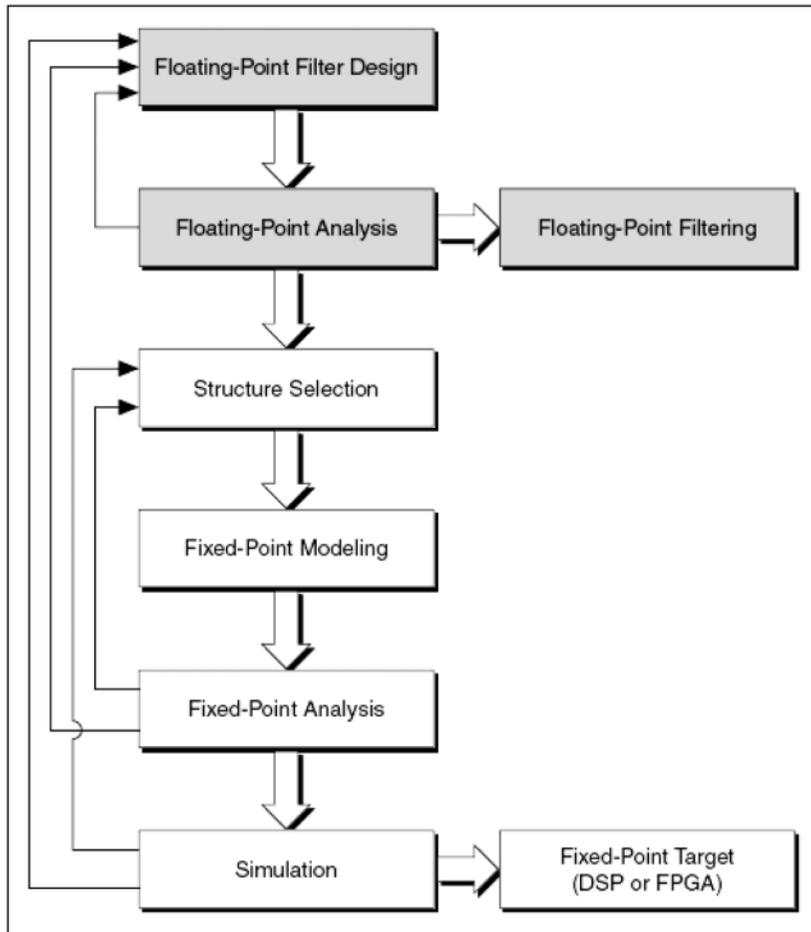


Рис. 16.12. Процесс проектирования цифрового фильтра с фиксированной запятой

информации о квантовании можно обратиться ниже к пункту **Моделирование фильтров с фиксированной запятой**.

3. Проверка достоверности модели фильтра с фиксированной запятой на основе анализа и моделирования с фиксированной запятой. Если фильтр с фиксированной запятой не выполняет заданные технические требования, можно изменить параметры настройки квантования, изменить структуру выполнения или совершенствовать параметры фильтра для рекомендованного фильтра. Для получения информации о проверке достоверности фильтров с фиксированной запятой можно обратиться ниже к пункту **Подтверждение правильности фильтров с фиксированной запятой**.
4. Генерация кода операций смоделированного фильтра с фиксированной запятой для определенного аппаратного адресата. Для получения информа-

ции об извлечении коэффициентов фильтра и автоматической генерации кодов C или LabVIEW можно обратиться ниже к пункту **Генерация кода**.

Ниже в этом подразделе также приведены примеры и упражнения, которые демонстрируют каждый шаг проектирования фильтра.

### 16.4.3. Выбор структуры фильтра

Структура фильтра определяет, как используется арифметически заданный набор коэффициентов фильтра для обработки входного потока сигнала. Для указанного цифрового фильтра доступны множества математически эквивалентных структур алгоритмов. Следует отметить, что для цифрового фильтра с плавающей запятой различие структур алгоритмов в большинстве случаев незначительно влияет на поведение фильтра. В то же время для цифрового фильтра с фиксированной запятой отличия в структуре алгоритма могут привести к различным результатам.

Можно создать КИХ-фильтры со следующими структурами:

- КИХ-форма прямая (FIR Direct Form);
- КИХ-форма прямая транспонированная (FIR Direct Form Transposed);
- Симметричная КИХ (FIR Symmetric);
- Несимметричная КИХ (FIR Antisymmetric);
- Лестничная СС (минимально-фазовая) (Lattice MA (minimum phase));
- Лестничная СС (максимально-фазовая) (Lattice MA (maximum phase)).

Можно создать фильтры БИХ со следующими структурами:

- БИХ-прямая форма (IIR Direct Form);
- БИХ-форма каскадная секционная второго порядка (IIR Cascaded Second Order Sections Form);
- Все пропускающая лестничная (Lattice Allpass);
- Лестничная АР (Lattice AR);
- Лестничная АРСС (Lattice ARMA).

Когда выбирается структура фильтра, необходимо балансировать множеством факторов, включая тип фильтра (КИХ или БИХ), ресурсы выполнения, вычислительную сложность и чувствительность к квантованию коэффициентов.

### 16.4.4. Выбор структуры для КИХ-фильтров

Для КИХ-фильтра структура Direct Form КИХ – самая простая и наиболее прямая структура из набора типов функций преобразования фильтра. Структура Direct Form Transposed КИХ – альтернативная прямая реализация формы для фильтров КИХ. Обе формы прямых структур имеют одинаковую вычислительную сложность. Структура Direct Form КИХ требует меньших затрат памяти для того, чтобы хранить внутренние состояния. Структура Direct Form Transposed КИХ имеет лучшее время выполнения операций, если фильтр реализован параллельно. Можно использовать структуру Direct Form КИХ, если необходимо реализовать КИХ-фильтр, использующий последовательную структуру циклов.

Для линейности ФЧХ КИХ-фильтра используют структуры FIR Symmetric или FIR Antisymmetric, которые основаны на симметрии или несимметрии коэффициентов фильтра с целью уменьшения вычислительной сложности и использованной памяти.

### 16.4.5. Выбор структуры для БИХ-фильтров

Структура IIR Direct Form – самая простая БИХ-структура из набора передаточных функций. Эта структура обычно имеет малое число операций в реализации фильтра. Однако его чувствительность к конечной разрядности кода ограничивает ее использование в реализации с фиксированной запятой. Можно использовать IIR Cascaded Second Order Sections Form, чтобы эта структура каскадов привела к снижению воздействия конечной разрядности кода коэффициентов. Структура IIR Cascaded Second Order Sections Form II имеет ту же самую вычислительную сложность, как Form I, но Form I требует большего объема памяти для того, чтобы сохранять внутренние состояния фильтра. Форма II – наиболее часто используемая структура. Преимущества и недостатки использования Form I и Form II и их измененных копий такие же, как и у КИХ-структур FIR Direct Form и FIR Direct Form Transposed.

### 16.4.6. Использование лестничных структур

Лестничные структуры – хорошие альтернативы для реализации фильтра с фиксированной запятой. Такие структуры могут обеспечить стабильность фильтров БИХ с фиксированной запятой, пока обратные коэффициенты лестничной структуры имеют модули меньше единицы, независимо от того, как ограничена арифметическая точность.

ИПЦФ обеспечивает следующие три категории лестничных (Lattice) структур:

- **Базовый тип секции** – два перемножителя в секции лестничной структуры. Эта категория предполагает самую общую лестничную структуру.
- **Тип, имеющий один перемножитель на секцию** – один перемножитель в лестничной секции. Эта категория экономит некоторые ресурсы аппаратных адресатов, подобных ПЛИС.
- **Тип нормализованной секции** – четыре перемножителя в лестничной секции. Эта категория автоматически масштабирует внутренние сигналы, что помогает снизить влияние квантования коэффициентов в каждой лестничной секции за счет увеличения сложности реализации.

Для получения информации об использовании в структуре фильтра всепропускающей лестничной структуры (Lattice Allpass), Lattice AR, Lattice MA и Lattice ARMA можно обратиться к *Справке LabVIEW (LabVIEW Help)*.

В табл. 16.3 приведены заданные по умолчанию структуры фильтра, которые использует ВП Filter Design.

Можно использовать ВП DFD Convert Structure для выбора различных структур фильтра на основе учета следующих запретов:

- Нельзя преобразовать структуру или форму БИХ из структуры БИХ.
- Нельзя преобразовать всепропускающую лестничную структуру (Lattice Allpass) из структуры Lattice AR.
- Нельзя преобразовать КИХ-фильтр к КИХ-фильтру с симметричной структурой, только если КИХ-фильтр имеет симметричные коэффициенты.
- Нельзя преобразовать КИХ-фильтр к КИХ-фильтру с несимметричной структурой, только если КИХ-фильтр имеет несимметричные коэффициенты.
- Нельзя преобразовать КИХ-фильтр к лестничной структуре фильтра Lattice MA (минимально-фазовый), только если КИХ-фильтр минимально-фазовый.
- Нельзя преобразовать КИХ-фильтр к лестничной структуре фильтра Lattice MA (максимально-фазовой), только если КИХ-фильтр максимально-фазовый.
- Если необходимо преобразовать структуру фильтра к всепропускающей лестничной структуре (Lattice Allpass), надо использовать всепропускающий (Allpass) фильтр.
- Если необходимо преобразовать структуру фильтра к лестничной структуре Lattice AR, надо использовать всеполосный БИХ-фильтр.

### 16.4.7. Метод проектирования структуры по умолчанию

На рис. 16.13 показано, как использовать ВП DFD Convert Structure для изменения структуры фильтра.

Различные структуры реализации могут вести к заметно отличающимся результатам.

После того как завершено моделирование, выполнен анализ и пройдены шаги моделирования, описанные ниже в этом подразделе, необходимо оценить эффект при других структурах фильтра, изменяя ввод **заданной структуры (target structure)** ВП DFD Convert Structure и запуская виртуальный прибор снова.

### 16.4.8. Моделирование фильтров с фиксированной запятой

В реализации фильтра с фиксированной запятой квантование выполняется для коэффициентов, промежуточных операндов и результатов. Необходимо моделировать все квантователи правильно в реализации фильтра с фиксированной запятой, как показано на рис. 16.14.

Эта модель фильтра с фиксированной запятой содержит следующие переменные:

- $Q_c$  – квантователь коэффициента для рекомендуемых коэффициентов фильтра с плавающей запятой;
- $Q_i$  – квантователь для входного сигнала фильтра с фиксированной запятой;
- $Q_o$  – квантователь для выходного сигнала фильтра с фиксированной запятой;

Таблица 16.3. Методы проектирования фильтра по заданной по умолчанию структуре

Метод проектирования	Структура по умолчанию
Баттерворта (Butterworth)	IIR Cascaded Second Order Sections Form II
Чебышева (Chebyshev)	IIR Cascaded Second Order Sections Form II
Инверсный Чебышева (Inverse Chebyshev)	IIR Cascaded Second Order Sections Form II
Эллиптический (Elliptic)	IIR Cascaded Second Order Sections Form II
Бесселя (Bessel)	IIR Cascaded Second Order Sections Form II
Произвольная групповая задержка (Arbitrary Group Delay)	IIR Direct Form II
КИХ с наименьшей $p$ -й нормой (Least Pth Norm IIR)	IIR Direct Form II
БИХ с наименьшей $p$ -й нормой (Least Pth Norm FIR)	FIR Direct Form
Окно Кайзера (Kaiser Window)	FIR Direct Form
Окно Дольфа – Чебышева (Dolph-Chebyshev Window)	FIR Direct Form
БИХ с окном (Windowed FIR)	FIR Direct Form
Ремез/с равномерными пульсациям (Remez/Equi-Ripple)	FIR Direct Form
БИХ Пик Метки (IIR Notch Peak)	IIR Direct Form II
БИХ лестничный (IIR Comb)	IIR Direct Form II
БИХ максимально плоский (Maxflat IIR)	IIR Cascaded Second Order Sections Form II
Компенсатор групповой задержки (Group Delay Compensator)	IIR Cascaded Second Order Sections Form II

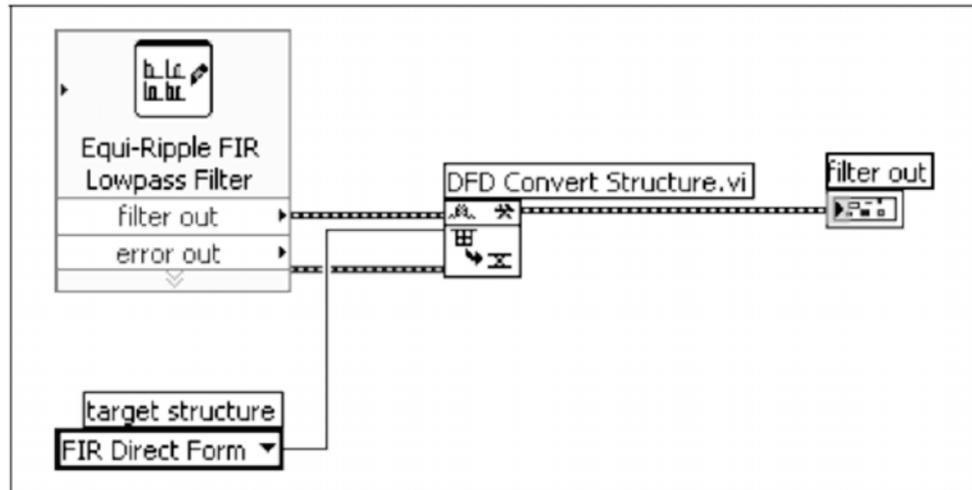


Рис. 16.13. Блок-диаграмма преобразователя структуры цифрового фильтра

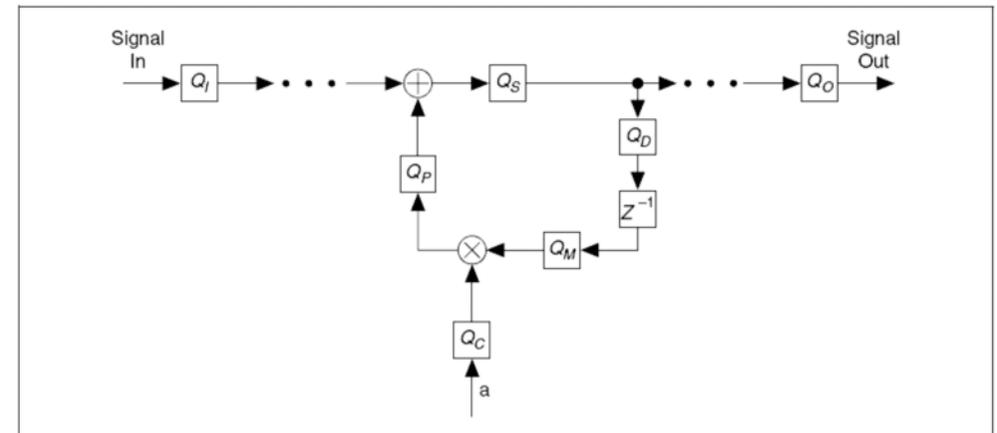


Рис. 16.14. Модель фильтра с фиксированной запятой

- $Q_M$  – квантователь на входе перемножителя для сомножителя с фиксированной запятой, который умножает квантуемый сомножитель на квантуемый коэффициент;
- $Q_P$  – квантователь результата на выходе перемножителя с фиксированной запятой;
- $Q_S$  – квантователь суммы на выходе сумматора с фиксированной запятой;
- $Q_D$  – квантователь задержки для ввода элемента задержки.

Каждый квантователь имеет различное влияние на фильтр с фиксированной запятой в зависимости от структуры фильтра. Необходимо моделировать и имитировать поведение фильтра с фиксированной запятой, через испытания и ошибки, прежде чем можно использовать в приложениях полученный фильтр с фиксированной запятой.

Несмотря на то, что можно определить эффект от квантования коэффициента во времени разработки, нельзя определить эффект от квантования данных, пока не будет опробована фильтрация с фактическими данными. Например, фактические данные могли бы быть слишком большими или маленькими по величине в пределах периода. Поэтому важно моделировать процесс фильтрации с образцами данных, который имеют подобные характеристики сигнала с фактическими данными.

### 16.4.9. Квантователи моделирования

На рис. 16.15 показано диалоговое окно **Configure FXP Modeling for Code Generation**, которое появляется автоматически, когда вызывается блок-диаграмма ВП DFD FXP Modeling for CodeGen Express. Можно ввести свойства квантования для коэффициентов фильтра, входных и выходных отсчетов, операндов в этом диалоговом окне.

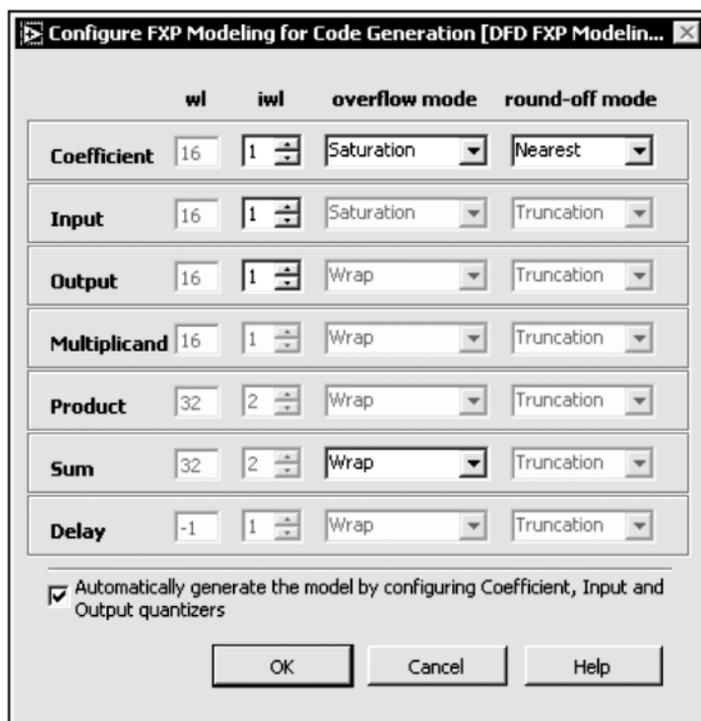


Рис. 16.15. Конфигурация FXP Modeling – для диалогового окна Code Generation: overflow mode – вид переполнения; round-off mode – вид округления; Saturation – насыщение; Nearest – ближайший; Truncation – усечение; Wrap – виток

## 16.4.10. Установка $wl$ и $iwl$

Длина слова  $wl$  определяет, каким числом битов обычно представляют число с фиксированной запятой. Целочисленная длина слова  $iwl$  определяет число битов, включая знаковый разряд, обычно представляет целую часть числа с фиксированной запятой. Различие в битах между  $wl$  и  $iwl$  определяется значением точности.

Поскольку ВП DFD FXP Code Generator поддерживает только 16-разрядные системы с 32-разрядными сумматорами и перемножителями, когда генерируется код для модели, прибор DFD FXP Modeling для CodeGen Express вычисляет  $wl$  и отображает его в диалоговом окне конфигурации. Необходимо только определить  $iwl$ .

Для чисел без знака и со знаком большее значение  $iwl$  вообще приводит к более широкому диапазону за счет точности. Надо выбрать наименьшее значение  $iwl$ , которое обеспечивает достаточный диапазон, чтобы охватить возможные значения.

## 16.4.11. Установка режима переполнения

Числа с фиксированной запятой могут представить только числа конечного диапазона.

Переполнение происходит, когда значение является большим, чем максимально представимое число в пределах диапазона, и потеря значимости происходит, когда это значение ниже, чем минимально представимое число в пределах диапазона. Можно управлять переполнением и потерей значимости, используя один из двух режимов:

- **Насыщение** – в режиме насыщения квантователь преобразовывает указанное число к максимально представимому числу, соответствующему переполнению или к минимальному представимому номеру для потери значимости.
- **Виток** – в режиме витка квантователь переносит указанное значение от максимально представимого числа до минимально представимого числа для переполнения и от минимально представимого числа до максимально представимого числа для потери значимости.

Для большинства приложений режим насыщения предпочтителен по сравнению с режимом витка, потому что размер ошибки не увеличивается так резко с насыщением, по сравнению с тем, как это делается с витком, когда происходят переполнение или потеря значимости. В режиме витка, когда происходят переполнение или потеря значимости, абсолютное значение ошибки –  $2iwl$ , которая будет больше, чем полный доступный динамический диапазон.

## 16.4.12. Установка режима округления

Поскольку числа с фиксированной запятой представляют собой дискретные значения с ограниченной точностью, не каждое значение в заданном диапазоне может быть представлено точно. Округление определяет, какое число с фиксированной запятой является соответствующим представителем для указанного значения, но в то же самое время с потерей точности.

Можно использовать один из следующих режимов, чтобы определить, как нужно округлять, чтобы перейти:

- к ближайшему (**Nearest**) – результат округляется к самому близкому представимому числу. Если к двум самым близким представимым числам расстояние равно, то этот вид режима обеспечивает округление к самому близкому представимому числу, наименьший значащий бит которого есть 0.
- к усечению (**Truncation**) – округление к самому близкому представимому числу, меньшему, чем первоначальное значение. Такой вид округления является самым популярным округлением в аппаратных средствах.

## 16.4.13. Установка квантователей вручную

По умолчанию, в ВП DFD FXP Modeling для CodeGen Express автоматически устанавливается  $iwl$  для перемножителя, результата, суммы и задержки, основанные на квантователях коэффициента, входа, выхода и предопределенных внут-

ренных отношениях среди всех семи квантователей (см. рис. 16.14). Если необходимо изменить **iw1** для всех семи квантователей вручную, надо удалить пометку из переключателя **Automatically generate the model by configuring Coefficient, Input and Output quantizers**.

### 16.4.14. Проверка достоверности фильтров с фиксированной запятой

После моделирования фильтра с фиксированной запятой необходимо проверить правильность поведения смоделированного фильтра. Проверка правильности состоит из анализа фильтра с фиксированной запятой и проведения моделирования.

### 16.4.15. Анализ фильтра с фиксированной запятой

ВП DFD Filter Analysis Express выполняет анализ фильтра с фиксированной запятой в области частоты. Этот ВП вычисляет частотную характеристику и распределение нулей и полюсов для данного набора коэффициентов, структуры фильтра и параметров квантования. Можно использовать результаты расчетов, чтобы оптимизировать фильтр с фиксированной запятой. Имеется возможность выполнить анализ итерационно и скорректировать проект фильтра, до тех пор, пока результаты расчетов не будут удовлетворительными.

Можно использовать ВП DFD Filter Analysis Express, чтобы наблюдать отклик фильтра с фиксированной запятой. Для проверки устойчивости фильтра с фиксированной запятой необходимо проверить, чтобы все полюса находились в пределах круга с единичным радиусом и чтобы фильтр обеспечивал удовлетворительную частотную характеристику. Если характеристики фильтра с фиксированной запятой не удовлетворяют требованиям, то необходимо попробовать один или больше следующих вариантов:

- возвратиться к моделированию по шагам и изменить параметры настройки квантователя коэффициентов;
- изменить структуру реализации;
- изменить параметры рекомендуемого фильтра с плавающей запятой, чтобы увеличить разрядность для выходных отсчетов с конечной точностью;
- для фильтров БИХ уменьшить ограничение радиуса полюса рекомендуемого фильтра с плавающей запятой.

Если наблюдаются искажения в отклике, надо найти условия переполнения или потери значимости. Необходимо использовать ВП DFD FXP Coef Report, чтобы определить, были ли обнаружены переполнение или потеря значимости в течение анализа. Если обнаружены любые проявления переполнения или потери значимости в значениях коэффициентов, можно возвратиться к шагу моделирования и увеличить **iw1** для квантователя коэффициентов, чтобы попробовать устранить переполнение и потерю значимости и улучшить отклик.

**Примечание.** Потеря значимости и переполнение не всегда влияют на отклик фильтра, особенно для БИХ-фильтров. Если отклик фильтра удовлетворителен, нет необходимости вносить изменения, чтобы избежать переполнения или потери значимости.

### 16.4.16. Пример. Анализ фильтра с фиксированной запятой

Необходимо выполнить следующие шаги, чтобы анализировать смоделированный фильтр с фиксированной запятой, созданный из рекомендуемого фильтра с плавающей запятой.

1. Открыть и запустить ВП Lowpass\_Step 3\_Analyze Quantized Lowpass, расположенный в каталоге examples\Digital Filter Design\CaseStudy1.
2. Исследовать лицевую панель. Сравнить АЧХ рекомендуемого фильтра с плавающей запятой и смоделированного фильтра с фиксированной запятой. Даже если АЧХ фильтра с фиксированной запятой напоминает АЧХ рекомендуемого фильтра с плавающей запятой, можно скорректировать модель, чтобы эта характеристика лучше соответствовала заданной.
3. Выполнить обзор вывода **Coefficients Report**. Выводится сообщение о величинах рекомендуемого и квантуемого значения для каждого коэффициента фильтра. Сообщение также перечисляет число потерь значимости и переполнения после информации о коэффициентах фильтра.
4. Исследовать блок-диаграмму, показанную на рис. 16.16.
5. Дважды щелкнуть на ВП DFD FXP Modeling для CodeGen Express, чтобы открыть диалоговое окно параметров. Попробовать ввести маленькие значения **iw1** в управлении **Coefficients**, щелкая кнопку **OK** и запуская виртуальный прибор.

**Совет.** Желательно попробовать различные режимы округления, чтобы сделать АЧХ квантуемого фильтра соответствующей параметрам настолько близко, насколько возможно.

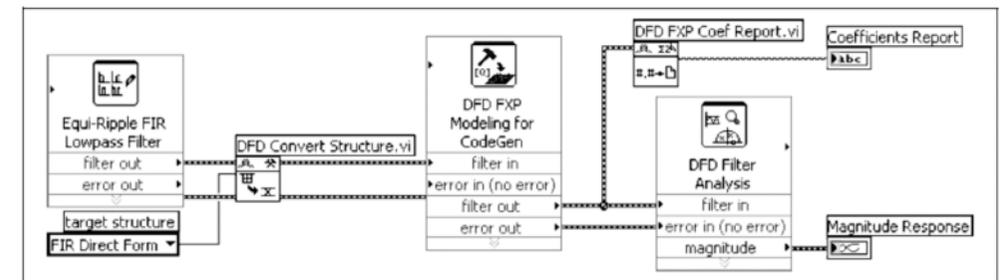


Рис. 16.16. Блок-диаграмма ВП Lowpass\_Step 3\_Analyze Quantized Lowpass VI

На рис. 16.17 и 16.18 показаны АЧХ для двух различных **iw1** параметров настройки, где FLP – фильтр с плавающей запятой; FXP – фильтр с фиксированной запятой. Можно обеспечить АЧХ, подобную рис. 16.17, если установить **iw1** в значение  $-2$ , и получить АЧХ, подобную рис. 16.18, если установить **iw1** в значение  $-3$ .

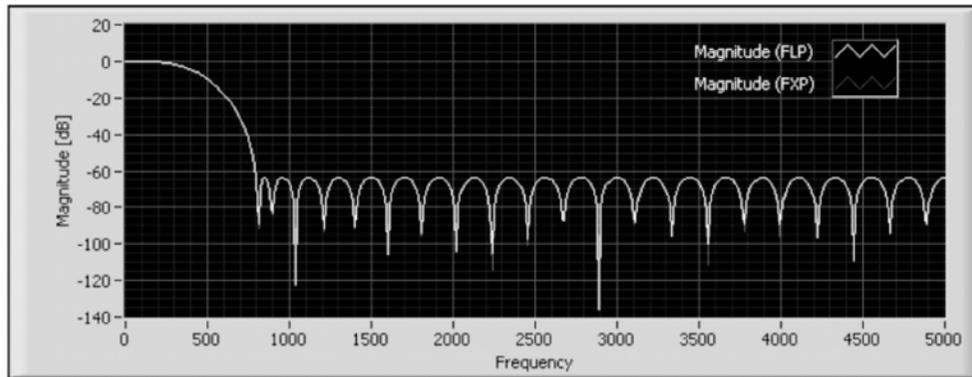


Рис. 16.17. АЧХ фильтра, когда  $iw1 = -2$

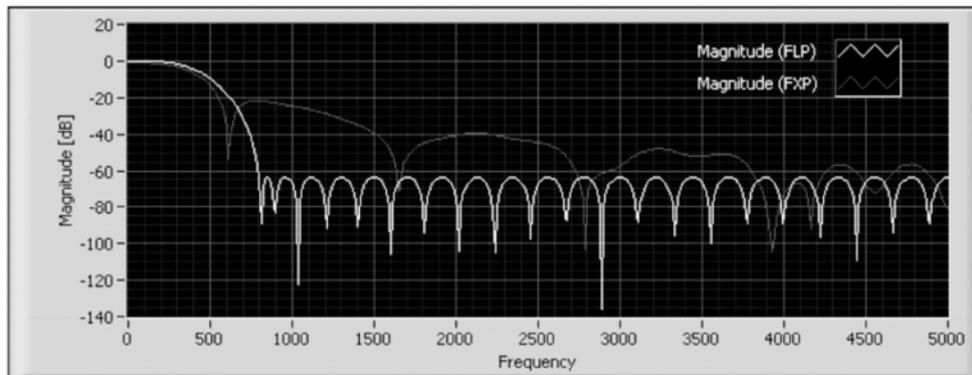


Рис. 16.18. АЧХ фильтра, когда  $iw1 = -3$

Необходимо обратить внимание, что АЧХ на рис. 16.18 искажена, в то время как АЧХ на рис. 16.17 близко соответствует рекомендуемой характеристике.

Поэтому можно установить, что **iw1** =  $-2$  есть лучшая опция.

## 16.4.17. Моделирование фильтра с фиксированной запятой

После того как выполнен анализ влияния квантования коэффициентов на характеристики фильтра, необходимо моделировать процесс фильтрации, чтобы про-

верить, работает ли, как ожидалось, полная модель с фиксированной запятой в окончательной реализации фильтра.

Можно использовать ВП DFD FXP Simulation и DFD FXP Simulation с ВП State, чтобы облегчить эту оценку.

Подобно всем другим частям процесса проектирования, моделирование есть эмпирический процесс. Надо выполнить следующие шаги, чтобы моделировать фильтрацию, обработать и оценить результаты.

1. Ввести параметры настройки для Входного квантователя ВП DFD FXP Modeling для CodeGen Express, основанного на диапазоне значений входного сигнала.
2. Моделировать поведение фильтра с фиксированной запятой с типичными входными данными, используя ВП DFD FXP Simulation или DFD FXP Simulation with State. Можно подключить на вход образцовый импульс, использовать ступенчатый сигнал, однородный белый шум или заданный пользователем сигнал к сигнальному входу ВП DFD FXP Simulation.
3. Использовать ВП DFD FXP Simulation Report для контроля поведения фильтра с фиксированной запятой в течение процесса моделирования, наблюдая вывод **filtering text report**.

Сообщение содержит статистическую информацию для всех квантователей, кроме квантователя коэффициента в модели с фиксированной запятой. Каждый квантователь имеет пять входов данных: **Max Value**, **Min Value**, **#Overflows**, **#Underflows** и **#Operations**.

4. Если наблюдается переполнение или потеря значимости в текстовом сообщении фильтрации или если результат моделирования не соответствует рекомендуемому фильтру с плавающей запятой, надо попробовать сделать следующие корректировки:
  - возвратиться к моделированию по шагам и увеличить **iw1** для заданных квантователей, чтобы устранить переполнение и потерю значимости как в **#Overflows**, так и в **#Underflows** при равенстве 0 или при падении ниже соответствующего порогового значения. Можно использовать установки **Max Value** и **Min Value** для оценки **iw1**;
  - возвратиться к моделированию по шагам и уменьшить **iw1** для заданных квантователей, чтобы повысить точность при уходе от переполнения или потери значимости;
  - изменить структуру реализации;
  - корректировать параметры и перепроектировать фильтр с плавающей запятой.

**Примечание.** Ввод **#Operations** для квантователей результата и суммы обеспечивает информацию о вычислительных требованиях к фильтру с фиксированной запятой. Меньшее значение означает более быструю вычислительную скорость. Если несколько структур фильтра удовлетворяют требованиям реализации фильтра, надо выбрать структуру фильтра, квантователь результата которой имеет наименьшее значение **#Operations**.

## 16.4.18. Пример. Моделирование фильтра с фиксированной запятой

Необходимо выполнить следующие шаги, чтобы моделировать другие квантователи и моделировать их влияние на фильтр, который анализировался в предыдущем примере:

1. Загрузить ВП Lowpass\_Step 4\_Model и Simulate FXP Lowpass, расположенные в examples\Digital каталоге Filter Design\CaseStudy1.
2. Исследовать блок-диаграмму, показанную на рис. 16.19.

Этот пример основан на использовании ВП DFD FXP Modeling для CodeGen Express при моделировании фильтра с фиксированной запятой; ВП DFD FXP Simulation – при моделировании процесса фильтрации, а также ВП DFD FXP Simulation Report.vi – для получения статистики процесса моделирования; ВП DFD Filtering.vi – для получения рекомендуемого результата для фильтра с плавающей запятой.

3. Запустить ВП. Наблюдать результат на индикаторе **Filtering Text Report** лицевой панели. Можно использовать эти данные, чтобы оптимизировать модель фильтра с фиксированной запятой. Например, можно убедиться, что **#Overflows** и **#Underflows** равны 0.

На рис. 16.20 показано текстовое сообщение результата фильтрации, отдельно – сообщения для чисел переполнения и потери значимости квантователя **Суммы**.

4. Чтобы устранять переполнение и потерю значимости в этом примере, необходимо дважды щелкнуть левой кнопкой мыши на ВП DFD FXP Modeling.vi для CodeGen Express, чтобы изменить параметры настройки квантователя. Удалить пометку из окон переключателей **Automatically generate the model by configuring Coefficient, Input and Output quantizers** и увеличить значение **iw1** квантователя суммы от  $-1$  до  $0$ . Нажать кнопку **OK**.

***Примечание.** При устранении переполнения и потери значимости можно улучшить реализацию цифрового фильтра, однако это не является необходимым или достаточным, чтобы гарантировать хорошую реализацию. Необходимо моделировать процесс фильтрации с представителем данных заданной системы. Можно также проверить процесс фильтрации на одностороннем белом шуме в полном масштабе и с гармоническими сигналами.*

5. Запустить ВП снова и проконтролировать индикатор **Filtering Text Report**. Квантователь суммы больше не должен испытывать переполнение или потерю значимости.
6. Исследовать результат моделирования, как показано на рис. 16.21. Результаты моделирования фильтра с фиксированной запятой соответствуют результатам для рекомендуемого фильтра с плавающей запятой.

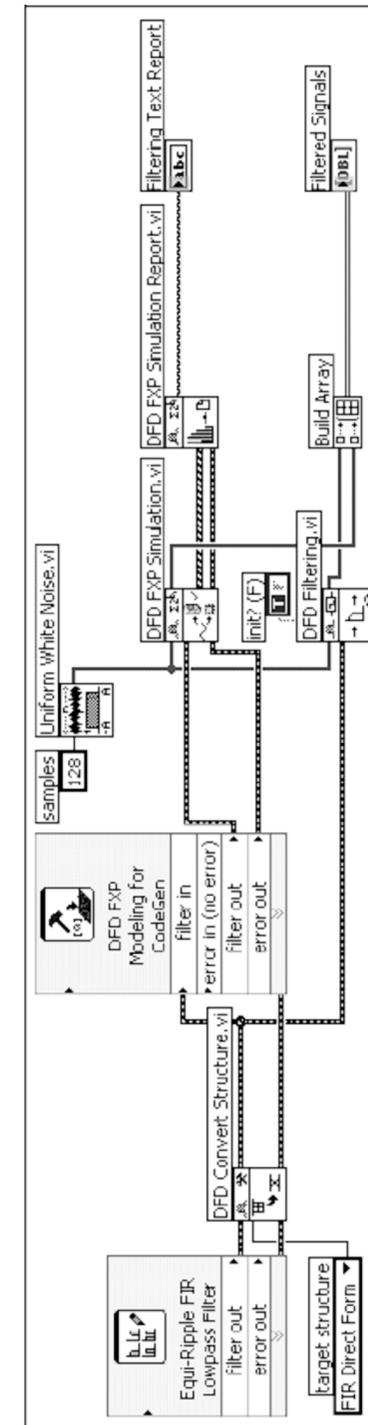


Рис. 16.19. Блок-диаграмма ВП Lowpass\_Step 4\_Model и Simulate FXP Lowpass

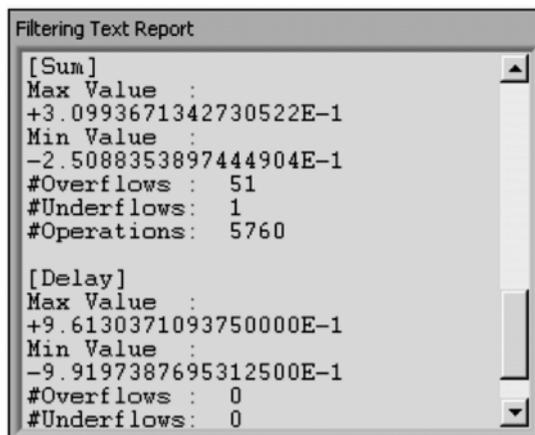


Рис. 16.20. Текстовое сообщение фильтрации

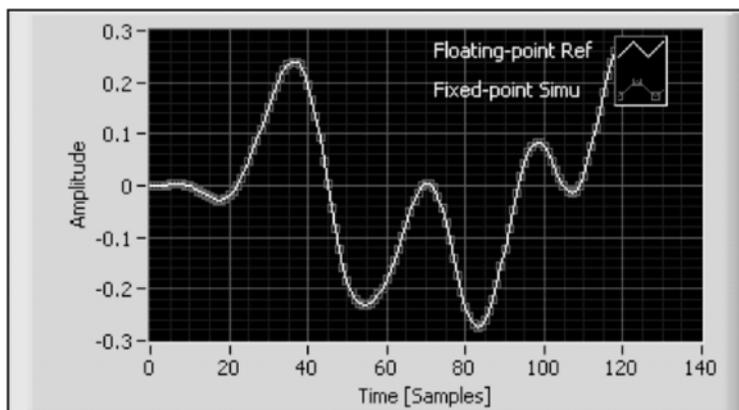


Рис. 16.21. Результат моделирования: для рекомендуемого фильтра с плавающей запятой (Floating-point Ref) и для моделируемого фильтра с фиксированной запятой (Fixer-point Simu)

## 16.4.19. Генерация кода

После того как получена соответствующая модель фильтра с фиксированной запятой, можно реализовать итоговый фильтр с фиксированной запятой на заданных аппаратных средствах. ИПЦФ обеспечивает следующие три варианта ввода в действие фильтра:

- извлекают коэффициенты фильтра с фиксированной запятой и используют коэффициенты в механизме реализации фильтра;
- используют ВП DFD FXP Code Generator для автоматической генерации кода С с фиксированной запятой. Тогда можно использовать компилятор

DSP C compiler, чтобы генерировать цифровой объектный код фильтра для универсального сигнального процессора (general-purpose DSP);

- используют ВП DFD FXP Code Generator, чтобы автоматически генерировать код LabVIEW с фиксированной запятой или код LabVIEW, оптимизированный для модуля LabVIEW FPGA Module. Затем можно использовать адрес модуля LabVIEW FPGA Module и реализовывать фильтр в устройстве NI FPGA.

## 16.4.20. Извлечение целочисленных коэффициентов фильтра с фиксированной запятой

Если имеется механизм реализации фильтра, для которого необходимы только коэффициенты фильтра, можно извлечь коэффициенты фильтра. Можно также сохранить коэффициенты в текстовом файле и направить их адресату реализации. Можно обратиться к Get Integer Coef ВП Fixed-point Filter, расположенному в каталоге examples\Digital Filter Design\Fixed-Point Filters, для изучения примера, который демонстрирует, как извлечь целочисленные коэффициенты с фиксированной запятой.

## 16.4.21. Производство кода С для фильтра с фиксированной запятой

Это обычная операция программирования заданных сигнальных процессоров (DSP), использующих коды С. Чтобы сгенерировать код С из модели фильтра с фиксированной запятой, можно использовать ВП DFD FXP Code Generator VI и выбрать **С Code** из полиморфных внутренних селекторов виртуального прибора. Можно компилировать сгенерированный код, чтобы работать на сигнальном процессоре с фиксированной запятой.

**Примечание.** Код С может привести к менее компактной и менее эффективной реализации, чем в режиме ручного создания (hand-crafted) кодовых сборок (assembly-coded), которые обычно требуют меньшего числа кодов и памяти данных и могут работать быстрее. Если необходимо улучшить реализацию фильтра, можно транслировать код С к ассемблерному коду вручную.

В табл. 16.4 представлен список квантователей и параметры, которые поддерживает ИПЦФ.

**Примечание.** Если генерируется код С с помощью ВП DFD FXP Modeling для CodeGen, необходимо установить значение режима переполнения (**overflow**) квантователя суммы в положение **Wrap**. Если генерируется код LabVIEW, можно установить значения режима переполнения квантователя суммы или **Wrap**, или **Saturation**.

Таблица 16.4. Ограничения на квантователи для генерации объектного кода C

Источник (Source)	Разрядность (wl)	Режим переполнения (overflow mode)	Режим потери значимости (round-off mode)
Коэффициент	16	Нет ограничений	Нет ограничений
Вход	16	Нет ограничений	Нет ограничений
Выход	16	Петля (Wrap)	Округление (Truncation)
Перемножитель	16	Петля	Округление
Результат	32	Петля	Округление
Сумма	32	Петля	Округление
Задержка*	16 или 32	Петля	Округление

\* Разрядность (wl) задержки зависит от избранной структуры фильтра. Преобразованные структуры фильтра имеют разрядность 32, а другие структуры – 16. Преобразованные структуры включены в виртуальные приборы: FIR Direct Form Transposed, IIR Direct Form I Transposed, IIR Direct Form II Transposed, IIR Cascaded Second Order Sections Form I Transposed и IIR Cascaded Second Order Sections Form II Transposed

ВП DFD FXP C Code Generator производит три файла в приборе – **filename** – строка, к которой надо обращаться для ввода имени фильтра:

- *nidfdtyp.h* содержит определения типов данных, используемых в исходных файлах C, которые генерирует DFD FXP C Code Generator.
- *filename.h* содержит определения типов, объявления глобальной переменной и функциональные прототипы.
- *filename.c* содержит код, в котором реализован фильтр, включая коэффициенты фильтра, информацию о структуре выполнения и параметрах настройки квантователя в модели с фиксированной запятой и следующих функциях:
  - *filename\_State filename\_CreateState()* – создает пространство памяти, которое должно будет сохранить внутренние установки фильтра;
  - пробел *filename\_DisposeState* (состояние *filename\_State*) – избавляет от пространства памяти, которая имеет обыкновение сохранять внутренние состояния фильтра;
  - пробел *filename\_InitState* (состояние *filename\_State*) – переводит внутренние состояния к нулям. Эту функцию вызывают для первого блока, когда обработана большая последовательность данных, состоящая из множественных блоков данных;
  - I16 *filename\_Filtering* (состояние I16 *sampleIn*, *filename\_State*) – реализовывает фильтр с фиксированной запятой;
  - статический I16 *filename\_Coeff[]* – содержит квантуемые коэффициенты фильтра с фиксированной запятой.

## 16.4.22. Производство кода LabVIEW фильтра с фиксированной запятой

Чтобы генерировать код LabVIEW, необходимо использовать ВП DFD FXP Code Generator и выбрать **LabVIEW Code** из многих образцов селектора ВП. Можно выбрать один из следующих типов кода LabVIEW при вводе типа кода (**code type**):

- **Integer LabVIEW code** – только целые числа кода LabVIEW, который можно выполнить на любой платформе или другом адресате, который основан на ВП LabVIEW;
- **SCTL-Optimized LabVIEW FPGA code** – специально оптимизированный код, чтобы запустить NI Reconfigurable I/O (RIO) – устройства типа NI PXI-7831R. Необходимо иметь модуль LabVIEW FPGA, установленный на компьютере, от которого загружается ВП на ПЛИС-устройство.

В табл. 16.5 приведен список структур фильтров текущей реализации с помощью ИПЦФ для генерации объектного кода LabVIEW.

## 16.4.23. Структура целочисленных кодов LabVIEW Code SCTL-Optimized LabVIEW FPGA

В табл. 16.6 приведен список чисел перемножителей, используемых для каждой структуры поддерживаемого фильтра, и оценка времени выполнения в числе дискрет времени.

Один дискрет времени – это один тактовый период, длина которого определяется тактовой частотой, для которой эти ВП откомпилированы. Время выполнения может изменяться по команде управления.

## 16.4.24. Код SCTL-Optimized LabVIEW FPGA

SCTL (Single-Cycle Timed Loop) – петлевая структура, в которой повторяется секция кода каждый тактовый период по умолчанию FPGA clock до условного предела, когда на вход поступает специфическое булево значение.

Можно использовать SCTL, чтобы увеличить скорость выполнения и уменьшить использование логики в ПЛИС, а также флуктуации в приложениях FPGA.

**Примечание.** Использование опции кодирования SCTL-Optimized LabVIEW FPGA возможно только для целей применения ПЛИС. Если работать на сгенерированном коде SCTL-Optimized LabVIEW FPGA на LabVIEW, предназначенном для Windows, можно столкнуться с проблемами синхронизации, даже если имеется установленный модуль LabVIEW FPGA.

Оптимизированная программа не всегда может компилировать успешно. Например, вычислительные ресурсы в приборе ПЛИС могут быть недостаточны,

Таблица 16.5. Структуры фильтров, в настоящее время реализованные для генерации объектного кода LabVIEW

Структура	Целочисленный код LabVIEW	SCTL-оптимизированный код LabVIEW ПЛИС (FPGA)
FIR Direct Form	Внедрен	Внедрен
FIR Direct Form Transposed	Внедрен	Внедрен
FIR Symmetric	Внедрен	Внедрен
FIR Antisymmetric	Внедрен	Внедрен
IIR Direct Form I	Внедрен	Не внедрен
IIR Direct Form I Transposed	Внедрен	Не внедрен
IIR Direct Form II	Внедрен	Не внедрен
IIR Direct Form II Transposed	Внедрен	Не внедрен
IIR Cascaded Second Order Sections Form I	Внедрен	Внедрен
IIR Cascaded Second Order Sections Form I Transposed	Внедрен	Внедрен
IIR Cascaded Second Order Sections Form II	Внедрен	Внедрен
IIR Cascaded Second Order Sections Form II Transposed	Внедрен	Внедрен
Lattice Allpass (basic sections)	Внедрен	Не внедрен
Lattice Allpass (one multiplier sections)	Внедрен	Не внедрен
Lattice Allpass (normalized sections)	Внедрен	Не внедрен
Lattice AR (basic sections)	Внедрен	Не внедрен
Lattice AR (one multiplier sections)	Внедрен	Не внедрен
Lattice AR (normalized sections)	Внедрен	Не внедрен
Lattice MA (minimum phase)	Внедрен	Внедрен
Lattice MA (maximum phase)	Внедрен	Внедрен
Lattice ARMA (basic sections)	Внедрен	Внедрен
Lattice ARMA (one multiplier sections)	Внедрен	Внедрен
Lattice ARMA (normalized sections)	Внедрен	Внедрен
IIR Cascaded Second Order Sections Form I	Внедрен	Внедрен
IIR Cascaded Second Order Sections Form I Transposed	Внедрен	Внедрен
IIR Cascaded Second Order Sections Form II	Внедрен	Внедрен
IIR Cascaded Second Order Sections Form II Transposed	Внедрен	Внедрен
Lattice Allpass (basic sections)	Внедрен	Не внедрен
Lattice Allpass (one multiplier sections)	Внедрен	Не внедрен
Lattice Allpass (normalized sections)	Внедрен	Не внедрен
Lattice AR (basic sections)	Внедрен	Не внедрен
Lattice AR (one multiplier sections)	Внедрен	Не внедрен
Lattice AR (normalized sections)	Внедрен	Не внедрен
Lattice MA (minimum phase)	Внедрен	Внедрен
Lattice MA (maximum phase)	Внедрен	Внедрен
Lattice ARMA (basic sections)	Внедрен	Внедрен
Lattice ARMA (one multiplier sections)	Внедрен	Внедрен
Lattice ARMA (normalized sections)	Внедрен	Внедрен

Таблица 16.6. Число перемножителей и оценка времени выполнения целочисленного кода LabVIEW

Структура	Число перемножителей	Оценка времени выполнения (дискрет)
FIR Direct Form	1	$4order+8$
FIR Direct Form Transposed	2	$5order+4$
FIR Symmetric (even order)	2	$3.5order+5$
FIR Symmetric (odd order)	1	$3.5order+7.5$
FIR Antisymmetric	1	$\lceil order/2 \rceil +4$
IIR Direct Form I	2	
IIR Direct Form I Transposed	3	$5order+5$
IIR Direct Form II	2	$4(order_{num}+order_{den})+9$
IIR Direct Form II Transposed	3	$5(order_{num}+order_{den})+5$
IIR Cascaded Second Order Sections Form I	5	$9\lceil order/2 \rceil +4$
IIR Cascaded Second Order Sections Form I Transposed	5	$7\lceil order/2 \rceil +4$
IIR Cascaded Second Order Sections Form II	5	$9\lceil order/2 \rceil +4$
IIR Cascaded Second Order Sections Form II Transposed	5	$8\lceil order/2 \rceil +4$
Lattice Allpass (basic sections)	4	$8order-2$
Lattice Allpass (one multiplier sections)	2	$7order$
Lattice Allpass (normalized sections)	8	$6order$
Lattice AR (basic sections)	2	$8order+4$
Lattice AR (one multiplier sections)	2	$7order+5$
Lattice AR (normalized sections)	5	$6order+5$
Lattice MA (minimum phase)	2	$5order+4$
Lattice MA (maximum phase)	2	$5order+4$
Lattice ARMA (basic sections)	4	$9order+6$
Lattice ARMA (one multiplier sections)	3	$8order+6$
Lattice ARMA (normalized sections)	6	$7order+6$

$order$  – полный порядок фильтра;

$order_{num}$  – порядок числителя передаточной функции фильтра;

$order_{den}$  – порядок знаменателя передаточной функции фильтра;

$\lceil x \rceil$  – наименьшее целое число, большее или равное  $x$ .

чтобы обеспечить требования фильтра с фиксированной запятой. Это может привести к неудачной компиляции.

Другая потенциальная причина отказа в компиляции – если фильтр с фиксированной запятой слишком усложнен, чтобы реализовать его в Single-Cycle Timed Loop, или, если тактовая частота проекта слишком высока. Если произошел отказ во время компиляции, можно преобразовать структуру фильтра к такой, которая потребляет меньше ресурсов или установить тактовую частоту проекта более низкой. Если никакие решения не сработают, можно попробовать переключиться на

целочисленный код LabVIEW, который может произвести жизнеспособный проект в ущерб эффективности.

**Примечание.** Если увеличена тактовая частота устройства FPGA, то меньше кодов выполняется в Single-Cycle Timed Loop, потому что тактовый цикл будет короче.

В табл. 16.7 приведен список числа множителей, используемых для каждой структуры поддерживающего фильтра, и оценка времени выполнения в тактах. Один такт – один тактовый период, длина которого определена тактовой частотой, для которой эти ВП откомпилированы. Время выполнения может изменяться по внешней команде.

### 16.4.25. Пример. Генерация кода LabVIEW для ПЛИС-Устройства из модели фильтра с фиксированной запятой

Чтобы сгенерировать код SCTL-Optimized LabVIEW FPGA, который использует LabVIEW FPGA Module Single-Cycle Timed Loop, необходимо завершить следующие шаги, обеспечивающие улучшение выполнения расчетов.

1. Открыть ВП Lowpass\_Step 5\_Generate Lowpass for FPGA, расположенный в examples\Digital каталоге Filter Design\CaseStudy1.
2. Исследовать блок-диаграмму ВП, показанную на рис. 16.22. В этом примере используется ВП DFD FXP LabVIEW Code Generator VI, чтобы генерировать код LabVIEW для программирования ПЛИС.
3. Вызвать ВП SCTL FPGA Lowpass Filter.v в VI path на лицевой панели и запустить ВП. Когда ВП закончит генерацию кода SCTL-optimized LabVIEW FPGA, это отображается на его лицевой панели.

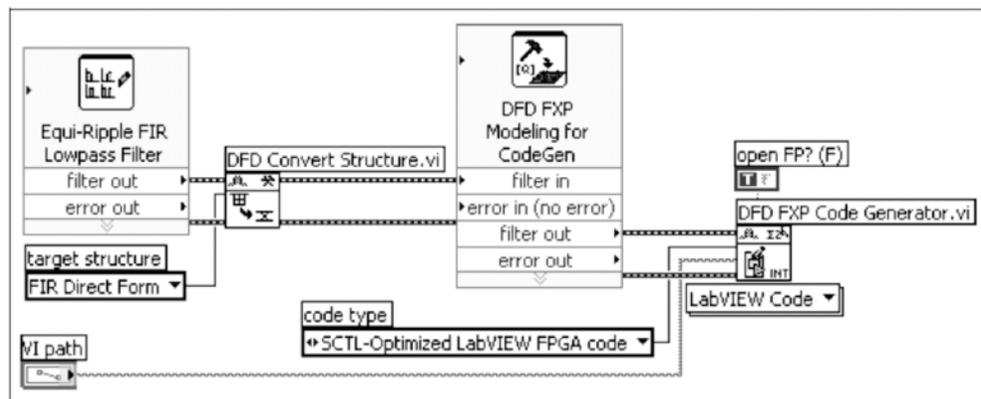


Рис. 16.22. Блок-диаграмма ВП Lowpass\_Step 5\_Generate Lowpass для ПЛИС

4. Рассмотреть блок-диаграмму ВП SCTL FPGA Lowpass Filter VI, показанную на рис. 16.23.

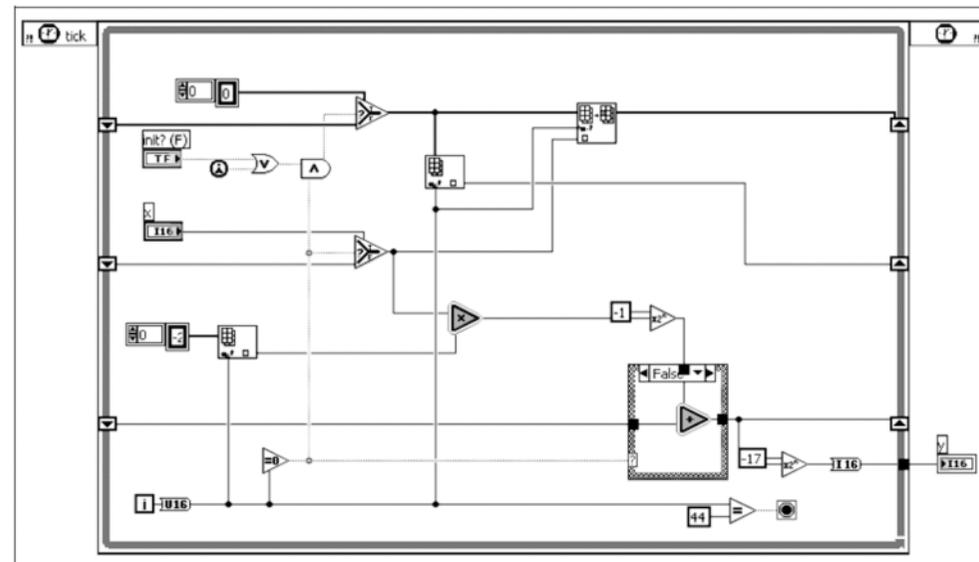


Рис. 16.23. Блок-диаграмма сгенерированного кода SCTL-Optimized LabVIEW FPGA

Таблица 16.7. Число перемножителей и оценка времени выполнения кода SCTL-Optimized LabVIEW FPGA

Структура	Число перемножителей	Оценка времени выполнения (такт)
FIR Direct Form	1	IIR Cascaded Second
Order Sections Form I Transposed		
FIR Direct Form Transposed	1	$order+1$
FIR Symmetric	1	$\lceil order/2 \rceil + 2$
FIR Antisymmetric	1	$\lceil order/2 \rceil + 2$
IIR Cascaded Second Order Sections Form I	1	$5 \lceil order/2 \rceil + 1$
IIR Cascaded Second Order Sections Form I Transposed	5	$\lceil order/2 \rceil + 2$
IIR Cascaded Second Order Sections Form II	5	$2 \lceil order/2 \rceil + 2$
IIR Cascaded Second Order Sections Form II Transposed	5	$2 \lceil order/2 \rceil + 2$
Lattice MA (minimum phase)	2	$order+2$
Lattice MA (maximum phase)	2	$order+2$
Lattice ARMA (basic sections)	3	$order+5$
Lattice ARMA (one multiplier sections)	2	$order+4$
Lattice ARMA (normalized sections)	5	$order+4$

$order$  – полный порядок фильтра;

$\lceil x \rceil$  – наименьшее целое число, большее или равное  $x$

Можно использовать этот код LabVIEW, как показано на рис. 16.24. Можно также успешно компилировать этот код и выполнить его на приборе NI FPGA.

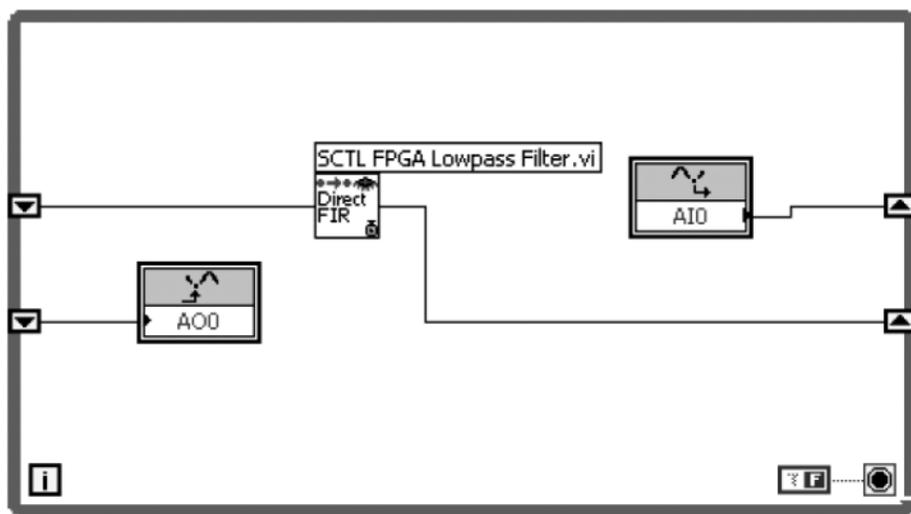


Рис. 16.24. Код фильтра ФНЧ LabVIEW, реализованный на ПЛИС

## Резюме

1. Процесс проектирования цифрового фильтра с фиксированной запятой выполняется в пять шагов:
  - выбор структуры фильтра;
  - моделирование квантователей;
  - анализ модели;
  - моделирование процесса фильтрации с образцовыми входными сигналами;
  - генерация кодов C или LabVIEW, чтобы компилировать и загружать в DSP или ПЛИС.
2. Можно выполнить каждый шаг в процессе использования следующих ВП:
  - DFD Convert Structure;
  - DFD FXP Modeling for CodeGen Express;
  - DFD Filter Analysis Express;
  - DFD FXP Simulation и DFD FXP Simulation Report;
  - DFD FXP Code Generator.
3. Для большинства фильтров с плавающей запятой структуры фильтра математически эквивалентны. В то же время, для большинства фильтров с фиксированной запятой изменение структуры может повлиять на эффективность и работу фильтра. Если не известно, какая выбрана структура фильтра, лучше использовать для метода проектирования структуры, заданные по умолчанию, как показано в табл. 16.3.

4. Если при анализе и моделировании процесса фильтрации и фильтра не выполняются заданные параметры, то необходимо изменить параметры настройки квантования, структуру фильтра или исходные параметры для исходного фильтра и повторить анализ и моделирование.
5. Для получения дополнительной информации о работе с ВП Fixed-Point Tools можно провести эксперимент с примерами в каталоге examples\Digital Filter Design\Fixed-Point Filters и обратиться к *LabVIEW Help*.

## 16.5. Расширенные и специальные проекты фильтра

В этом подразделе объясняется, как проектировать расширенные или специальные цифровые фильтры, используя следующие ВП: DFD Advanced IIR Filter Design, DFD Advanced FIR Filter Design и DFD Special Filter Design VIsDFD Advanced IIR Filter Design, DFD Advanced FIR Filter Design, и DFD Special Filter Design. Для получения окончательной информации об использовании ВП Filter Design можно обратиться к *LabVIEW Help*.

### 16.5.1. Фильтры с линейной ФЧХ и минимально-фазовые

Линейная ФЧХ – это такая характеристика, в соответствии с которой цифровые фильтры обеспечивают одинаковую задержку для всех гармонических составляющих входного сигнала. Это означает, что групповая задержка имеет постоянное значение, независимое от частоты. Фильтры с линейной ФЧХ полезны в таких применениях, где необходимо минимизировать длительность протяженного во времени сигнала.

Нелинейная ФЧХ или дисперсия, когда задержка зависит от частоты составляющей, может иметь как негативное, так и положительное влияние на формирование выходного сигнала. В некоторых случаях, например в аудиоприложениях при записи и воспроизведении фонограмм, можно допустить некоторое искажение фазы составляющих, часто незаметное слушателям. Положительное влияние дисперсии может быть в согласованной фильтрации при обнаружении частотно-модулированных сигналов, когда необходимо обеспечить сжатие длительности сигнала путем сведения составляющих спектра к одному моменту времени. Однако искажение фазы составляющих может быть и вредно в некоторых приложениях.

Например, в цифровых приложениях связи распространение сигнала, вызванное искажением фазы, может вызвать интерференцию (наложение) по времени концентраций информационных символов.

Минимально-фазовый фильтр – тип фильтра с нелинейной ФЧХ, который оптимально минимизирует групповую задержку на всех частотах для данной АЧХ за счет искажения фазы. Минимально-фазовые фильтры могут быть полезны

в приложениях управления, в которых уменьшение задержки более важно, чем уменьшение расширения сигнала по длительности и искажения формы сигнала.

## 16.5.2. Математическое определение

Комплексная передаточная функция цифрового фильтра  $H(f)$  может быть выражена в обозначениях модуля и аргумента в следующем уравнении:

$$H(f) = |H(f)|e^{i\Phi(f)}, \quad (16.5)$$

где  $|H(f)|$  и  $\Phi(f)$  являются реальными оценками функций частоты  $f$ , которые представляют АЧХ и ФЧХ соответственно.

Только КИХ-фильтры могут обладать строго линейной ФЧХ. Можно проектировать КИХ-фильтры с линейной ФЧХ, используя методы проектирования на основе окон или метод проектирования Ремеза (Remez). Метод Remez в ВП DFD Remez Design более мощнее и гибче, чем методы проектирования на основе окон.

Уравнение (16.5) преобразуется к следующему уравнению для линейной ФЧХ цифрового КИХ-фильтра:

$$H(f) = (j)^m A(f) e^{-jm\pi f}, \quad m = 0, 1. \quad (16.6)$$

В уравнении (16.6) использованы следующие обозначения:

- $N$  – порядок фильтра, который равен числу отводов фильтра минус один;
- $f$  – нормализованная частота с диапазоном  $[0, 0,5]$ ;
- если коэффициенты фильтра  $h(n)$  (где  $n = 0, 1, \dots, N$ ) симметричны,  $h(n) = h(N - n)$ , то  $m$  должно быть равным 0;
- если коэффициенты фильтра несимметричны,  $h(n) = -h(N - n)$ , то  $m$  в этом случае должно быть равным 1.

## 16.5.3. Типы КИХ-фильтров с линейной ФЧХ

В табл. 16.8 приведен список из четырех типов КИХ-фильтров с линейной ФЧХ и характеристики каждого из них.

**Примечание.** Для получения дополнительной информации о КИХ-фильтрах с линейной ФЧХ имеет смысл обратиться к книге Digital Filter Design авторов T.W. Parks и C. S. Burrus [31]. Можно обратиться также к библиографическому списку литературных источников, которые содержат подробную информацию о теории и алгоритмах, реализованных в ИПЦФ.

Можно использовать ВП DFD Remez Design для проектирования КИХ-фильтра с линейной ФЧХ. Установить ввод порядка (**Order**) и ввод типа фильтра (**filter type**) согласно столбцу Классификация табл. 16.9. ВП DFD Remez Design поможет спроектировать соответствующий тип КИХ-фильтра с линейной ФЧХ, основанный на этих вводах.

Таблица 16.8. Типы КИХ-фильтров с линейной ФЧХ (где  $f$  – нормированная частота)

Тип	Классификация	Частотные характеристики
I	Четный порядок (нечетное число отсчетов) Симметрия	$A(f)$ четная относительно $f = 0$ и нечетная относительно $f = 0,5$ $A(f)$ периодическая с периодом 1
II	Нечетный порядок (четное число отсчетов) Симметрия	$A(f)$ четная относительно $f = 0$ и $f = 0,5$ $A(f)$ ограничена от 0 до $f = 0,5$ $A(f)$ периодическая с периодом 2
III	Четный порядок (нечетное число отсчетов) Несимметричный	$A(f)$ нечетная относительно $f = 0$ и $f = 0,5$ $A(f)$ ограничена 0 от 0 до $f = 0,5$ $A(f)$ периодическая с периодом 1
IV	Нечетный порядок (четное число отсчетов) Несимметричный	$A(f)$ нечетная относительно $f = 0$ и четная относительно $f = 0,5$ $A(f)$ ограничена 0 при $f = 0,5$ $A(f)$ периодическая с периодом 2

Можно использовать следующие рекомендации, чтобы определить, какой тип КИХ-фильтра с линейной ФЧХ нужно проектировать:

- типы III и IV не могут быть ФНЧ-подобными фильтрами;
- типы II и III не могут быть ФВЧ-подобными фильтрами;
- типы III и IV работают хорошо для дифференциаторов или преобразователей Гильберта (Hilbert), потому что они могут дать постоянный по частоте сдвиг по фазе на  $90^\circ$ .

## 16.5.4. Эксперимент с различными типами

Некоторое число типов фильтров могут обеспечить приемлемый результат в определенной задаче, но лишь один из них даст лучший среди них результат. Можно выбрать, например, тип фильтра, имеющего гладкую АЧХ.

Для примера рис. 16.25 иллюстрирует типы симметрии АЧХ для каждого типа КИХ-фильтров с линейной ФЧХ при частоте дискретизации  $f_s = 1$ . Надо обратить внимание, что в этом примере Типы III или IV фильтров приводят к самой гладкой АЧХ.

## 16.5.5. Пример КИХ-фильтра с линейной ФЧХ

На рис. 16.26 показана АЧХ, удовлетворяющая требованию взвешивающего фильтра ITU-468. Необходимо обратить внимание, что АЧХ имеет нулевое значения для постоянного тока и маленькие, но ненулевые значения на высоких частотах. Можно обратиться к рис. 16.25, чтобы определить подходящий тип фильтра для приближения к нужной АЧХ. Тип IV (нечетный порядок, несимметричный) – лучший выбор для такого приближения к заданной АЧХ.

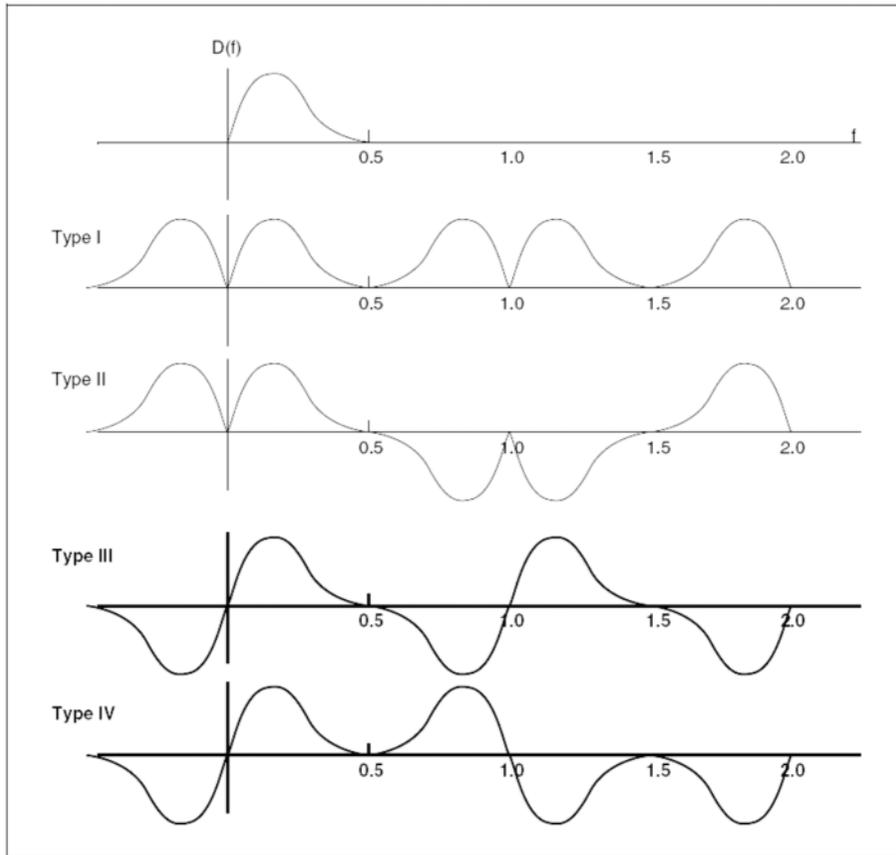


Рис. 16.25. Идеальные АЧХ для различных типов КИХ-фильтров, имеющих линейную ФЧХ

### 16.5.6. Преобразователи Гильберта (Hilbert)

Идеальная АЧХ преобразователя Гильберта равна

$$H(f) = -j\text{sgn}(f). \quad (16.7)$$

Комплексная передаточная функция обеспечивает сдвиг по фазе минус  $90^\circ$  для положительных частот и сдвиг по фазе  $90^\circ$  для отрицательных частот.

Если сравнить уравнения (16.7) и (16.6), можно видеть, что идеальный преобразователь Гильберта имеет линейную ФЧХ и относится к типу III или IV КИХ-фильтров с линейной ФЧХ.

Можно использовать ВП DFD Remez Design, чтобы проектировать преобразователь Гильберта, устанавливая на входе **filter type** в положение **Hilbert**. Ниже объясняется, как создать преобразователь Гильберта типов IV и III.

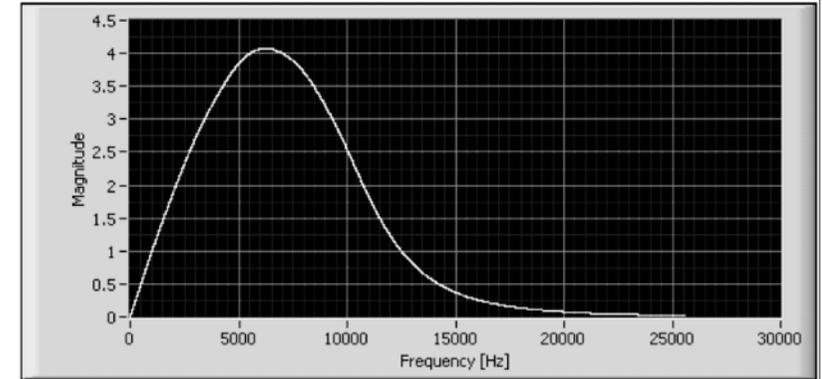


Рис. 16.26. Заданная АЧХ для взвешивающего фильтра ITU-468

### 16.5.7. Преобразователя Гильберта типа IV

Тип IV (нечетный порядок, несимметричный) позволяет делать фильтры – идеальные преобразователи Гильберта.

Когда проектируется преобразователь Гильберта типа IV в ВП DFD Remez Design, можно определить единственную полосу частот, которая содержит две точки с равными весами. Например, нужно спроектировать фильтр для диапазона частот  $[0,1; 0,5]$  с весом  $[-1, -1]$ . Чтобы спроектировать преобразователь Гильберта типа IV 11 порядка на указанную полосу, надо ввести параметры в ВП DFD Remez Design, показанные на рис. 16.27.

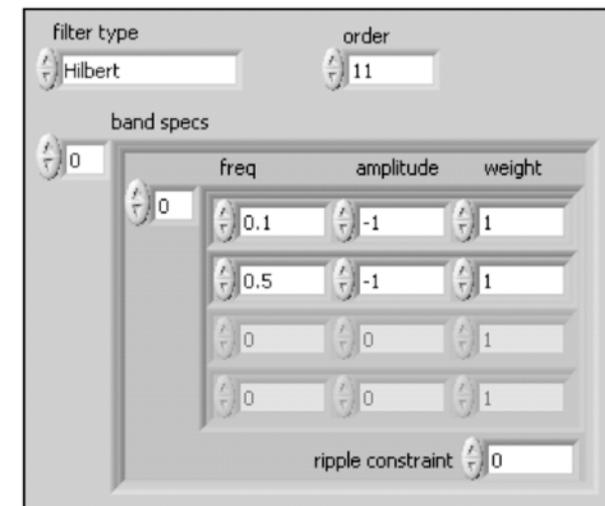


Рис. 16.27. Параметры преобразователя Гильберта типа IV

На рис. 16.28 показаны АЧХ и импульсная характеристика разработанного преобразователя Гильберта типа IV.

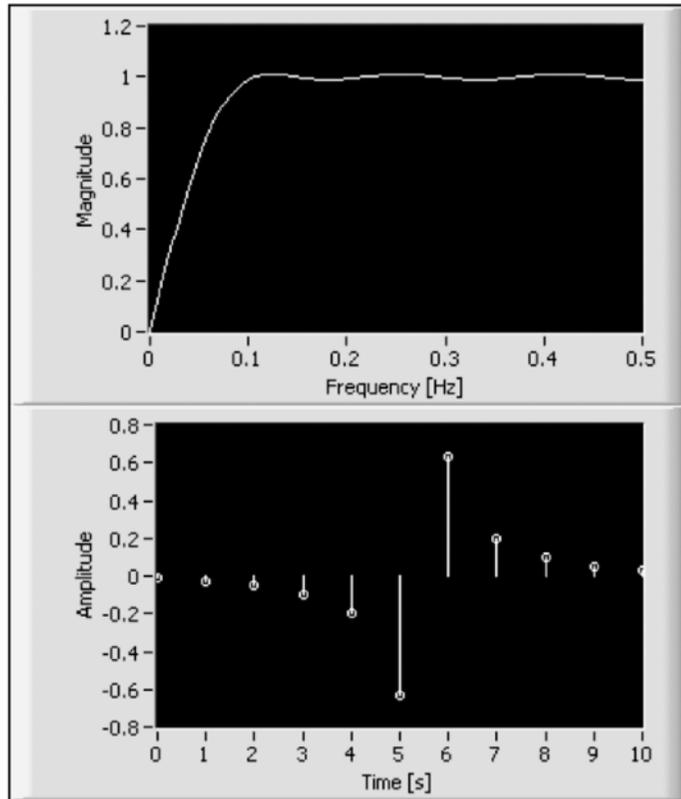


Рис. 16.28. АЧХ и импульсная характеристика разработанного преобразователя Гильберта типа IV

### 16.5.8. Преобразователь Гильберта типа III

Преобразователь Гильберта типа III (четный порядок, несимметричный) нужен, если необходимо отфильтровать составляющие с верхними частотами. Фильтры типа III обнуляют амплитуду составляющей с частотой Найквиста, равной 0,5 Гц. Изменяя диапазон частоты для предыдущего примера до [0,05; 0,45] и поддерживая соответствующие амплитуды составляющих, умножая их на весовые коэффициенты в интервале  $[-1, -1]$ , можно проектировать полосовой преобразователь Гильберта. Для этого примера проектируется преобразователь Гильберта типа III с порядком 12, вводя параметры, показанные на рис. 16.29, в ВП DFD Remez Design.

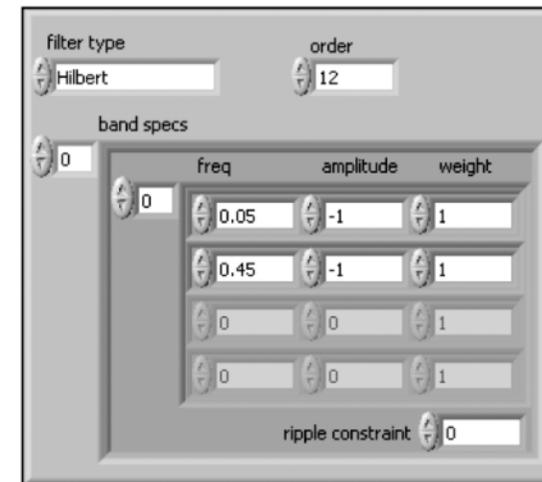


Рис. 16.29. Параметры преобразователя Гильберта типа III

На рис. 16.30 показана АЧХ окончательного преобразователя Гильберта типа III.

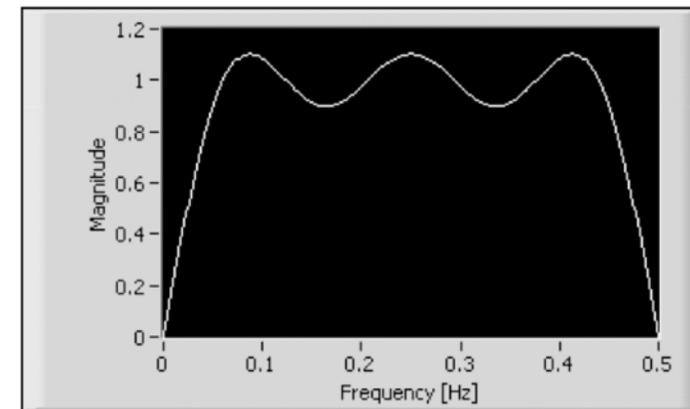


Рис. 16.30. АЧХ преобразователя Гильберта типа III

### 16.5.9. Дифференциаторы

Идеальная комплексная передаточная функция дифференциатора равна

$$H(f) = jf. \quad (16.8)$$

Передаточная функция обеспечивает постоянный сдвиг по фазе на  $90^\circ$  на всех частотах заданной полосы.

Если сравнить уравнения (16.8) и (16.6), можно видеть, что дифференциатор имеет линейную ФЧХ и является КИХ-фильтром типов III или IV.

Можно использовать ВП DFD Remez Design, чтобы спроектировать дифференциатор, устанавливая на вводе **filter type** опцию **Differentiator**. Такой дифференциатор находит применение, поскольку его передаточная функция в полосе пропускания дифференциатора имеет склон по частоте, малую ошибку аппроксимации на частотах, где АЧХ имеет малые абсолютные значения. ВП DFD Remez Design использует весовую функцию в полосе, обратно пропорциональную частоте, чтобы обеспечить постоянные по величине (в процентах) пульсации АЧХ. Ниже поясняется, как создать дифференциатор типов IV и III.

### 16.5.10. Дифференциатор типа IV

Когда необходимо иметь фильтр с передаточной функцией дифференциатора в диапазоне частот от постоянного тока до частоты  $f_s/2$ , единственным решением является выбор дифференциатора типа IV. Поскольку дифференциатор типа III обнуляет составляющую с частотой Найквиста, этот тип дифференциаторов не может обеспечить точную передаточную функцию дифференциатора вблизи частоты Найквиста.

Например, можно создать дифференциатор типа IV с порядком 19, вводя параметры, показанные на рис. 16.31, в ВП DFD Remez Design.

На рис. 16.32 показана АЧХ окончательного дифференциатора типа IV.

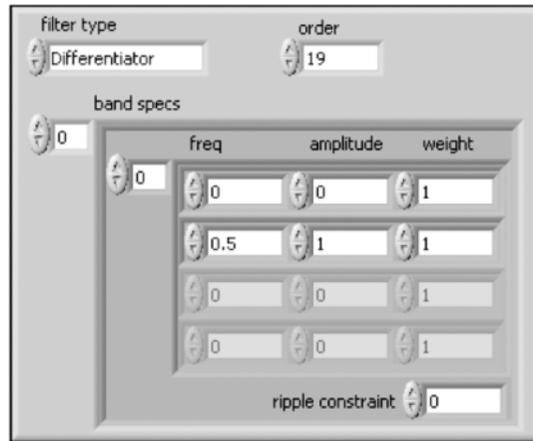


Рис. 16.31. Параметры дифференциатора типа IV

### 16.5.11. Дифференциаторов типа III

Если необходимо использовать дифференциатор в комбинации с фильтром ФНЧ, можно использовать дифференциатор типа III, имеющий передаточную функцию дифференциатора в полосе пропускания ФНЧ, и фильтр, который имеет полосу задержания, простирающуюся до частоты Найквиста.

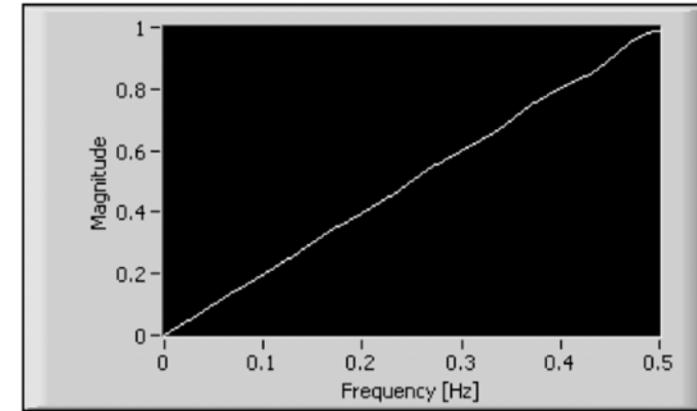


Рис. 16.32. АЧХ дифференциатора типа IV

Параметры, показанные на рис. 16.33, предназначены для проектирования дифференциатора типа III с порядком 20, который охватывает диапазон частот от 0 до 0,45 с заданными значениями 0 и 0,9 соответственно. Это – параметры для однополосного фильтра без указания заданного значения АЧХ при частоте 0,5 Гц.

На рис. 16.34 представлена АЧХ финального дифференциатора типа III.

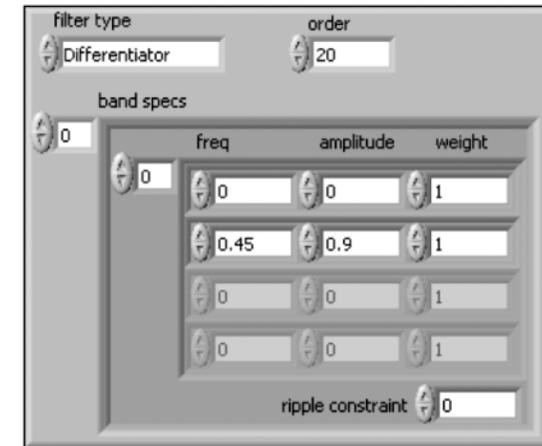


Рис. 16.33. Параметры дифференциатора типа III

Фильтры типа IV требуют меньшего числа коэффициентов фильтра и имеют меньший уровень пульсаций АЧХ из-за того, что дифференциатор типа III обнуляет спектральную составляющую с частотой 0,5 Гц.

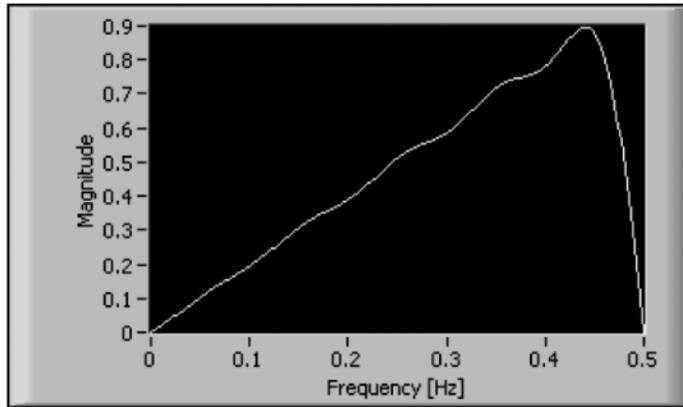


Рис. 16.34. АЧХ дифференциатора типа III

### 16.5.12. Фильтр-пробка и пиковый фильтр

Фильтр-пробка используется, чтобы подавить помеховый гармонический сигнал на заданной частоте типа сетевых наводок. Пиковый фильтр используют, чтобы усилить сигнал на заданной частоте.

ВП DFD IIR Notch Peak Design позволяет проектировать БИХ-фильтр-пробку второго порядка, или пиковый фильтр, используя метод билинейного преобразования. Уравнение

$$H(z) = \frac{b_0 \left( 1 + \frac{b_1}{b_0} z^{-1} + z^{-2} \right)}{1 + a_0 z^{-1} + a_1 z^{-2}} \quad (16.9)$$

описывает  $z$ -преобразование передаточной функции фильтра-пробки.

Уравнение

$$H(z) = \frac{b_0 (1 + z^{-2})}{1 + a_0 z^{-1} + a_1 z^{-2}}, \quad (16.10)$$

где  $a_0$ ,  $a_1$ ,  $b_0$ , и  $b_1$  – коэффициенты фильтра, описывает  $z$ -преобразование передаточной функции пикового фильтра.

### 16.5.13. Пример фильтра-пробки

На рис. 16.35 показана временная диаграмма сигнала с помехой электрокардиограммы ((ECG)), при частоте дискретизации 333 Гц.

На рис. 16.36 представлен спектр того же самого сигнала с помехой. На этом рисунке можно идентифицировать сигнал сетевой наводки в 60 Гц.

Чтобы удалить наводку в 60 Гц, можно спроектировать фильтр-пробку, используя ВП DFD IIR Notch Peak Design со следующими параметрами.

Входы ВП DFD IIR Notch Peak Design	Значение
Тип фильтра	Пробка
$f_0$	60 Гц
Добротность $Q$	40
$f_s$	333 Гц

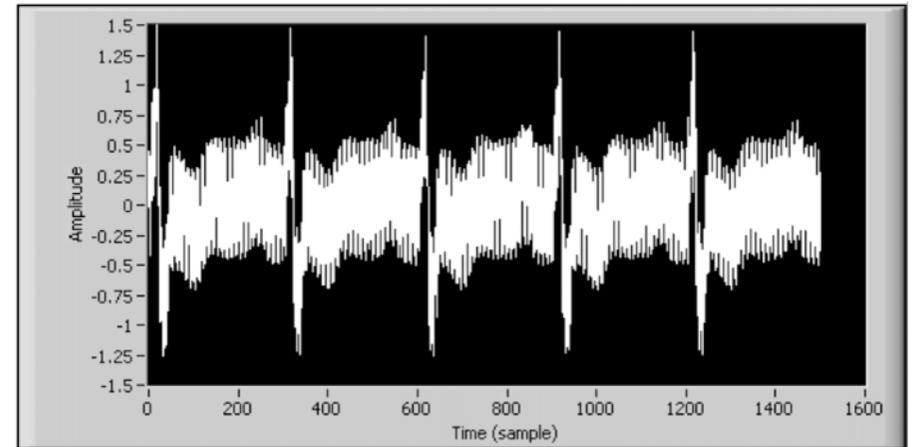


Рис. 16.35. Временная диаграмма сигнала электрокардиограммы в смеси с помехой

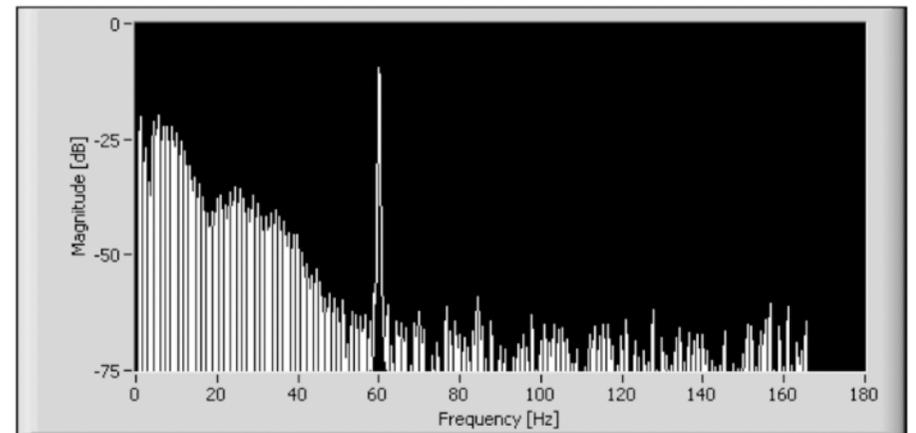


Рис. 16.36. Спектр сигнала кардиограммы с помехой

На рис. 16.37 показана АЧХ полученного фильтра-пробки.

Из рис. 16.38 следует, что отфильтрованный сигнал кардиограммы близок к исходным выборкам без сетевой наводки.

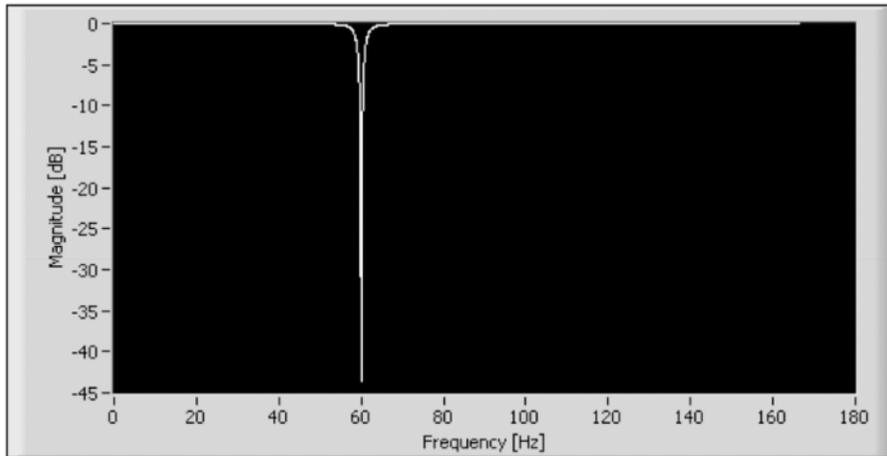


Рис. 16.37. АЧХ фильтра-пробки на 60 Гц

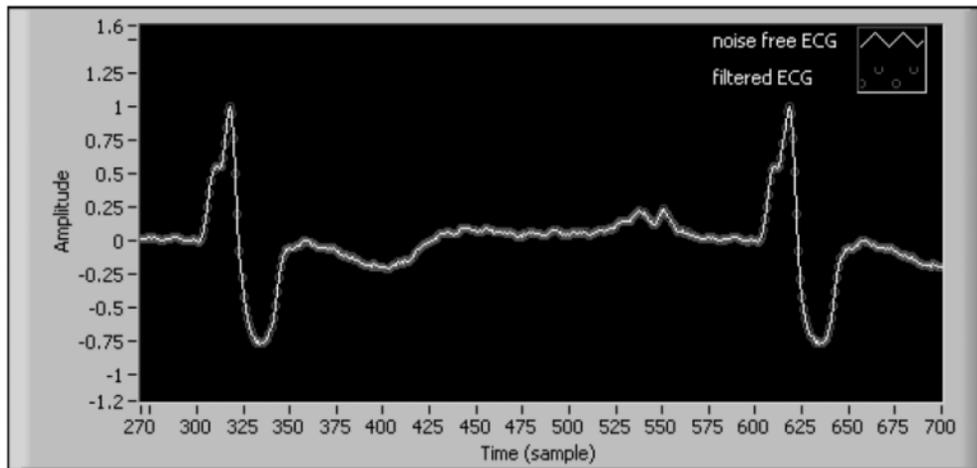


Рис. 16.38. Свободная от сетевой наводки (исходная) и отфильтрованная кардиограммы: *noise free ECG* – кардиограмма без сетевой помехи; *filtered ECG* – отфильтрованная кардиограмма

### 16.5.14. Гребенчатые фильтры

Используя ВП DFD IIR Notch Peak Design, можно спроектировать фильтр-пробку и пиковые фильтры, которые подавляют наводку или выделяют гармонический сигнал с одной заданной частотой.

Однако можно подавить шум или выделить сигнал на большем числе частот. Например, необходимо устранить составляющие наводки от сети на основной час-

тоте и ее гармоники. В этой ситуации можно использовать гребенчатый фильтр, имеющий максимальные и минимальные значения АЧХ равномерно по оси частот.

Гребенчатые фильтры также могут разделить два сигнала с различными частотами гармоник. Например, в системе цветного телевидения фильтры гребенки отделяют компоненты сигнала яркости и компоненты информации о цвете из составного видеосигнала.

В табл. 16.9 приведен список разных типов гребенчатых фильтров с их z-преобразованием передаточной функции и графиками АЧХ. Необходимо обратить внимание, что можно создать два типа гребенки: с нулями в АЧХ или максимумами на заданных кратных частотах. Один тип имеет нули, другой – пики на частотах  $kf_s/N$ , в то время как другой тип имеет нули или пики на частотах  $(k + 1/2)f_s/N$ , где  $k$  – целое число в диапазоне  $[0, N - 1]$ .

### 16.5.15. Пример гребенчатого фильтра

На рис. 16.39 показана временная диаграмма сигнала кардиограммы в смеси с помехой и с частотой дискретизации 480 Гц.

На рис. 16.40 представлен спектр того же самого сигнала в смеси с помехой. В спектре можно идентифицировать гармоники с частотами 60, 120 и 180 Гц.

Чтобы удалить такую помеху, можно спроектировать гребенчатый фильтр-пробку, используя ВП DFD IIR Comb Design, подбирая  $f_0$  и различные полосы пропускания (**Bandwidth**), а также устанавливая следующие параметры.

Входы ВП DFD IIR Comb Design	Величины
Тип фильтра	Фильтр-пробка, тип I
$f_0$	60 Гц
$\Delta f$	0,5
$f_s$	480

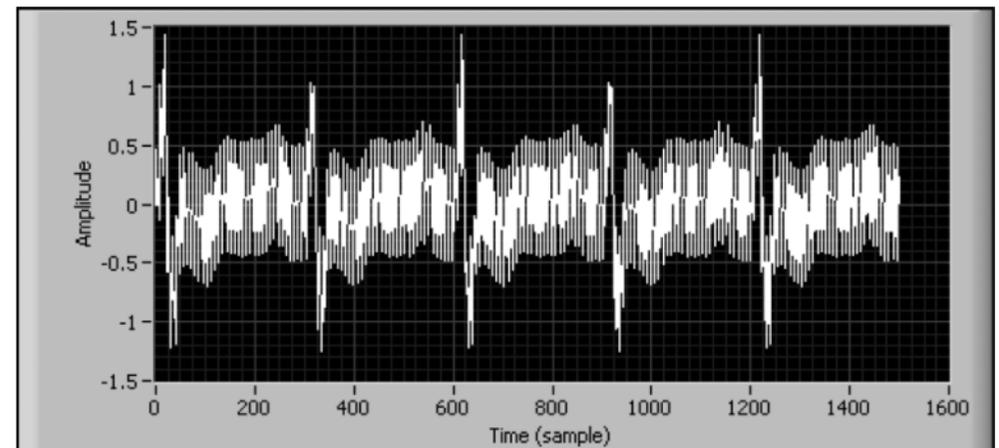
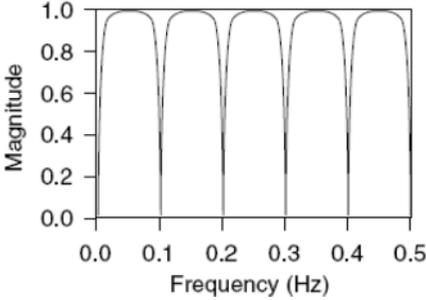
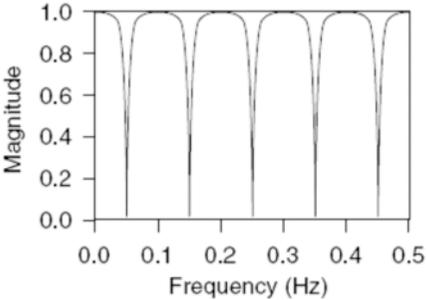
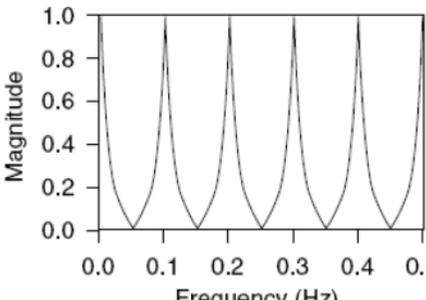
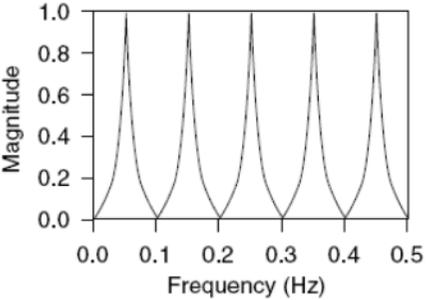


Рис. 16.39. Временная диаграмма сигнала кардиограммы в смеси с помехой

Таблица 16.9. Типы гребенчатых фильтров с нулями или максимумами в АЧХ

Фильтр	Передаточная функция	АЧХ (N = 10)
Фильтр-пробка, тип I	$\frac{b(1-z^{-N})}{1-az^{-N}}$	
Фильтр-пробка, тип II	$\frac{b(1+z^{-N})}{1+az^{-N}}$	
Фильтр пиков, тип I	$\frac{b(1+z^{-N})}{1-az^{-N}}$	
Фильтр пиков, тип II	$\frac{b(1-z^{-N})}{1+az^{-N}}$	

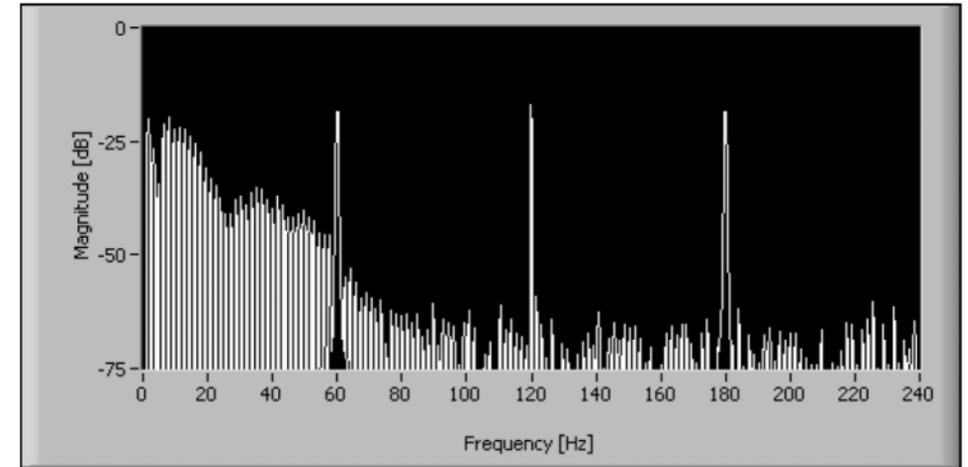


Рис. 16.40. Спектр зашумленного сигнала кардиограммы

На рис. 16.41 изображена АЧХ полученного гребенчатого фильтра.

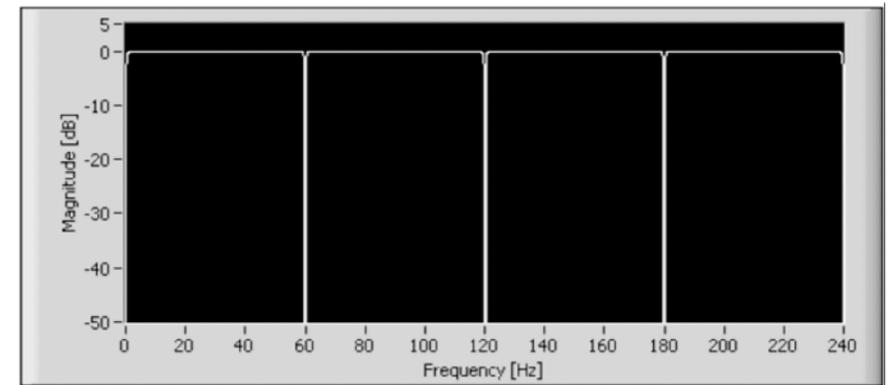


Рис. 16.41. АЧХ гребенчатого фильтра-пробки

Отфильтрованный сигнал кардиограммы, близкий к исходным выборкам без помехи, показан на рис. 16.42.

### 16.5.16. Фильтры с произвольной формой АЧХ

Если необходимо спроектировать КИХ-фильтр с линейной ФЧХ и произвольной формой АЧХ, можно использовать ВП DFD Remez Design. Если же надо спроектировать фильтр, в котором линейная ФЧХ не требуется, но важно минимизиро-

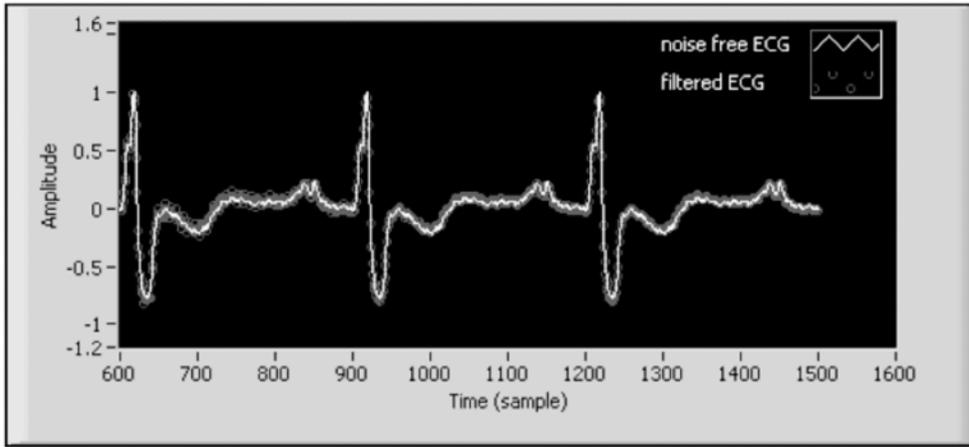


Рис. 16.42. Кардиограмма ECG без помехи и отфильтрованная ECG

вать порядок фильтра, можно использовать ВП DFD Least Pth Norm Design, чтобы проектировать БИХ-фильтр.

Произвольная форма АЧХ описывается множеством точек, используя кусочно-линейную интерполяцию. Число точек, которое необходимо обеспечить, чтобы описать форму, зависит от заданной формы АЧХ в некоторой полосе частот. Точки на оси частот распределяются неравномерно. Необходимо использовать больше точек, где АЧХ сильно изогнута, и меньше точек, где АЧХ более гладкая по частоте.

### 16.5.17. КИХ-фильтр с линейной ФЧХ и произвольной АЧХ

Чтобы спроектировать КИХ-фильтр с линейной ФЧХ и АЧХ произвольной формы, можно использовать ВП DFD Remez Design и установить при вводе типа фильтра (**filter type**) симметричный (**Symmetric**) или несимметричный (**Antisymmetric**) согласно табл. 16.8. Затем надо описать форму фильтра, определяя множество точки АЧХ при вводе параметров полосы (**band specs**).

### 16.5.18. Пример фильтра с произвольной АЧХ

В этом примере приведен фильтр ФНЧ с диапазоном частот полосы пропускания от 0 до 0,25 и полосой задержания в диапазоне частот от 0,3 до 0,5. Три точки на частотах 0; 0,1 и 0,25 с ожидаемыми значениями АЧХ 1, 2 и 1 соответственно описывают форму полосы пропускания. Чтобы проектировать этот фильтр, можно ввести параметры, показанные на рис. 16.43, на фронтальной панели ВП DFD Remez Design.

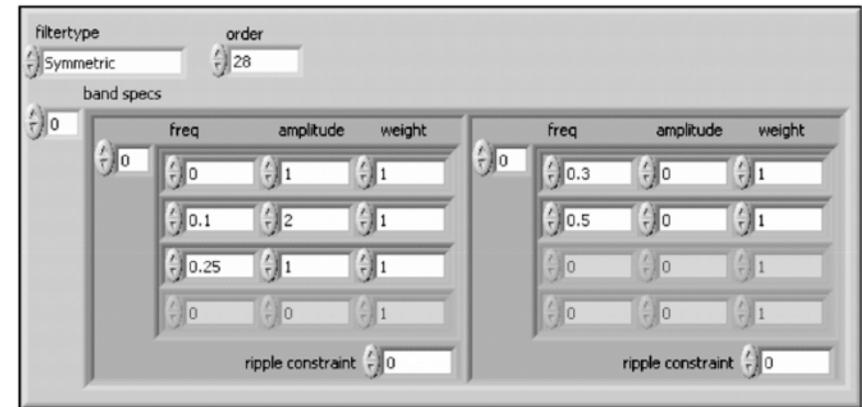


Рис. 16.43. Параметры КИХ-фильтра ФНЧ 28-го порядка с произвольной формой АЧХ

На рис. 16.44 представлена АЧХ разработанного фильтра.

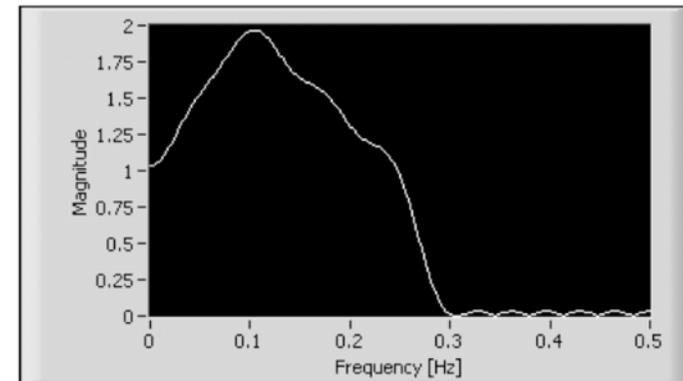


Рис. 16.44. АЧХ КИХ-фильтра ФНЧ 28-го порядка произвольной формы

### 16.5.19. Пример инверсного компенсационного sinc-фильтра

В этом примере приведен компенсационный ФНЧ sinc-фильтр, предназначенный для исправления спада амплитуд, вызванного использованием сглаживающего нулевого порядка в цифроаналоговом преобразователе (ЦАП).

На рис. 16.45 приведена АЧХ ФНЧ с диапазоном частот полосы пропускания от 0 до 0,2 и диапазоном полосы задержания по частоте от 0,3 до 0,5.

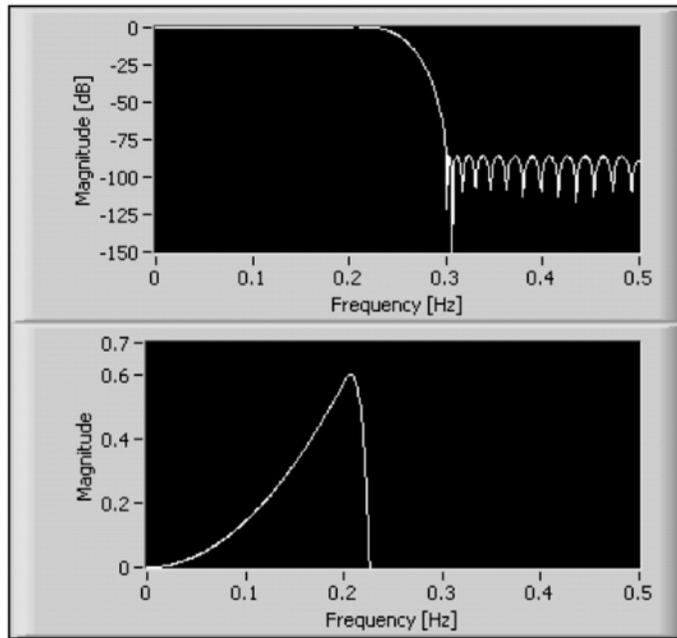


Рис. 16.45. Инверсный компенсационный sinc-фильтр 50-го порядка

Форма передаточной функции в полосе пропускания описана 100 отдельными точками, распределенными равномерно по частоте, с соответствующей инверсией функциональных значений sinc. Разработанный фильтр имеет 50-й порядок и одинаковый весовой коэффициент между полосой пропускания и полосой задержания. На нижнем графике рис. 16.45 показано, как инверсия формы sinc аппроксимируется в полосе пропускания.

### 16.5.20. БИХ-фильтр с произвольной АЧХ

Если необходимо спроектировать БИХ-фильтр с произвольной АЧХ, а форма ФЧХ не имеет значения, можно использовать ВП DFD Least Pth Norm Design и установить ввод типа фильтра (**filter type**) на **Minimum Phase** или на **Maximum Phase**. ВП DFD Least Pth Norm Design игнорирует все другие вводы параметров ФЧХ, включая групповую задержку (**group delay**) и ФЧХ (**phase**) на вводе параметров полосы пропускания (**band specs**). Можно при этом определить форму АЧХ, вводя множество точек на вводе **band specs**.

Например, можно спроектировать минимально-фазовый БИХ-фильтр с той же самой произвольной АЧХ, как и в предыдущем примере. Установить **filter type** на **Minimum Phase** и ввести те же самые параметры полосы (**band specs**) ВП DFD Least Pth Norm Design. На рис. 16.46 представлена АЧХ разработанного фильтра.

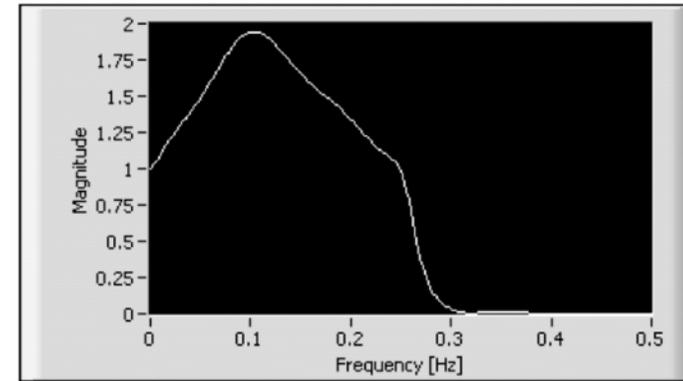


Рис. 16.46. ФНЧ БИХ-фильтра с произвольной формой АЧХ

### 16.5.21. Компенсатор групповой задержки

БИХ-фильтры, которые проектируются на основе использования окон Баттерворта, Чебышева, эллиптических обычно имеют непостоянную групповую задержку. Это означает, что они имеют нелинейную ФЧХ или искажение фазы. Самое большое отклонение от постоянной групповой задержки обычно соответствует границе полосы пропускания или где-нибудь в полосе перехода.

Заданный фильтр с искажением фазы можно расположить каскадом с всепропускающим фильтром, чтобы линеаризовать ФЧХ в заданном диапазоне частоты при обеспечении неизменной АЧХ.

Обозначим  $\tau_o(f_i)$  и  $\tau_{ap}(\overset{\gamma}{a}, f_i)$  как групповые задержки заданного фильтра и разработанного всепропускающего фильтра в  $i$ -й точке частоты соответственно. Вектор коэффициентов  $\overset{\gamma}{a}$  всепропускающего фильтра определяется следующим уравнением:

$$\min \sum_i \left| \tau_{ap}(\overset{\gamma}{a}, f_i) + \tau_o(f_i) - \tau \right|^p, \quad (16.11)$$

где  $\tau$  – заданная групповая задержка во всех определяемых пользователем диапазонах частот.

### 16.5.22. Пример компенсатора групповой задержки

Эллиптический полосовой фильтр 4-го порядка с полосой пропускания от 0,3 до 0,4 имеет непостоянную групповую задержку в указанной полосе пропускания. На рис. 16.47 показано, как компенсировать групповую задержку фильтром-компенсатором 8-го порядка в указанной полосе пропускания, чтобы сделать задержку постоянной на основе ВП DFD Group Delay Compensator.

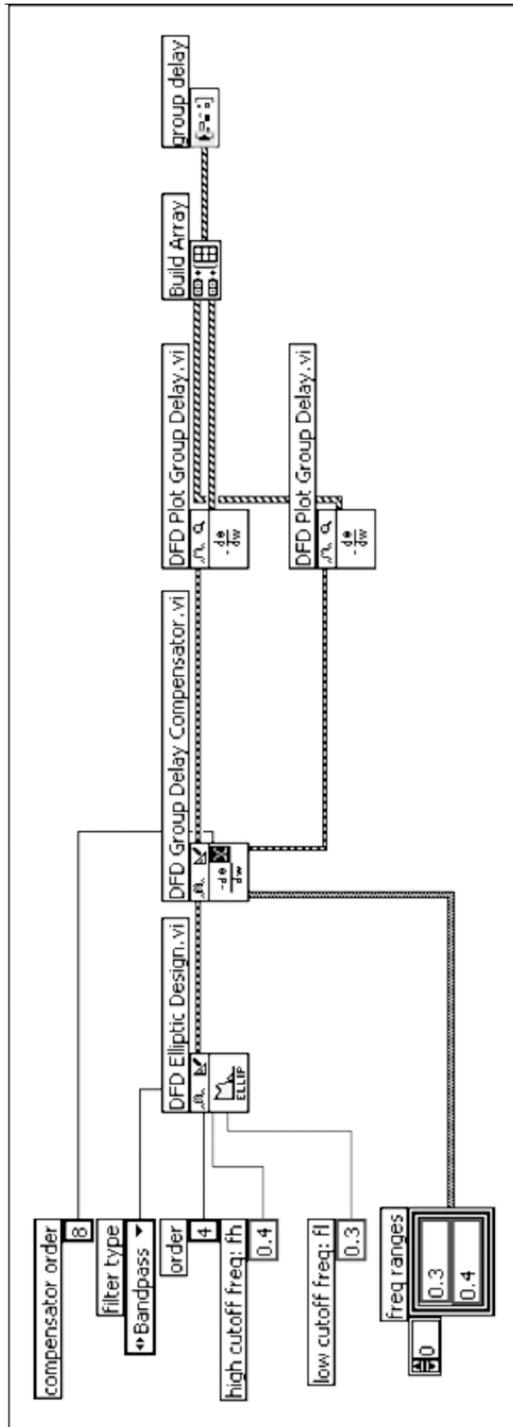


Рис. 16.47. Блок-диаграмма ВП компенсатора групповой задержки

Блок-диаграмма на рис. 16.47 основана на ВП DFD Plot Group Delay, чтобы проверить зависимость групповой задержки от частоты. На рис. 16.48 показана такая зависимость для исходного фильтра групповой задержки и фильтра, у которого компенсируют эту задержку. Надо обратить внимание, что компенсатор приблизительно линейризует ФЧХ в полосе пропускания за счет увеличенной задержки через фильтр и увеличенных вычислений при фильтрации.

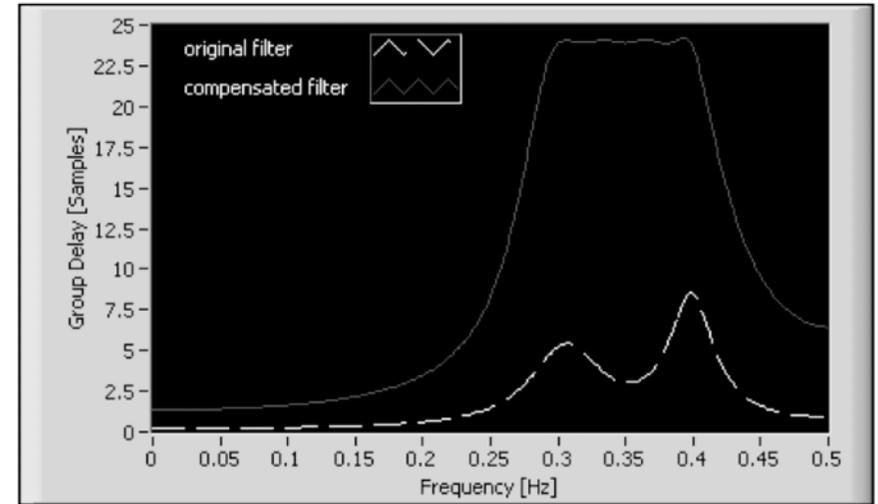


Рис. 16.48. Зависимости групповой задержки от частоты исходного фильтра и скомпенсированного фильтра

### 16.5.23. Узкополосные КИХ-фильтры

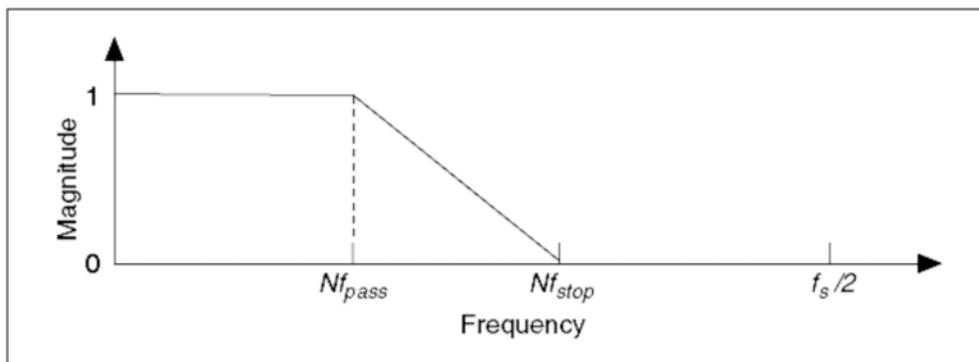
Порядок КИХ-фильтра обратно пропорционально зависит от его полосы перехода.

Обычные КИХ-фильтры с узкими полосами перехода и высокими порядками могут быть слишком сложными для реализации. Можно рассмотреть разработанные узкополосные фильтры на основе использования БИХ-фильтров. Однако узкополосные БИХ-фильтры обычно имеют нелинейную ФЧХ, особенно около полосы перехода, и чувствительны в цифровой форме. Можно обеспечить заданные параметры фильтра, используя специальные методы дизайна узкополосных КИХ-фильтров вместо того, чтобы использовать БИХ-фильтры.

ВП DFD Narrowband Filter Design использует интерполированные КИХ (IFIR) методы и методы маски АЧХ, чтобы проектировать узкополосные фильтры КИХ со значительно меньшей вычислительной сложностью, чем обычные решения КИХ. Чтобы лучше понять работу этих методов, предположим, что заданный узкополосный фильтр имеет АЧХ, показанную на рис. 16.49.

Рис. 16.49. АЧХ заданного фильтра с передаточной функцией  $H(z)$ 

Первый шаг заключается в проектировании формирующего фильтра с более широкой полосой перехода, как показано на рис. 16.50.

Рис. 16.50. АЧХ формирующего фильтра с передаточной функцией  $G(z)$ 

Следующий шаг заключается в проектировании интерполяционного фильтра с АЧХ, приведенной на рис. 16.51. Необходимо обратить внимание, что коэффициенты интерполирующего фильтра  $G(z^N)$  созданы путем расстановки  $N - 1$  нулей между каждыми двумя смежными коэффициентами функции  $G(z)$ .

АЧХ первого графика  $G(z^N)$  на рис. 16.51 такая же, как для заданного фильтра  $H(z)$  на рис. 16.49. Чтобы удалить нежелательные изображения  $G(z^N)$ , необходимо расположить каскадом интерполяционный фильтр  $G(z^N)$  с фильтром маски  $I(z)$ , АЧХ которого представлена на рис. 16.52.

Располагая каскадом интерполяционный фильтр  $G(z^N)$  и фильтр маски  $I(z)$ , как показано на рис. 16.53, можно получить заданный узкополосный фильтр  $H(z)$ .

На рис. 16.50 и 16.52 показано, что  $I(z)$  и  $G(z)$  имеют намного более широкую полосу перехода, чем исходный фильтр  $H(z)$ . Следовательно, общий порядок фильтров  $I(z)$  и  $G(z)$  ниже, чем порядок фильтра  $H(z)$ , что делает каскадные фильтры в вычислительном отношении более эффективными.

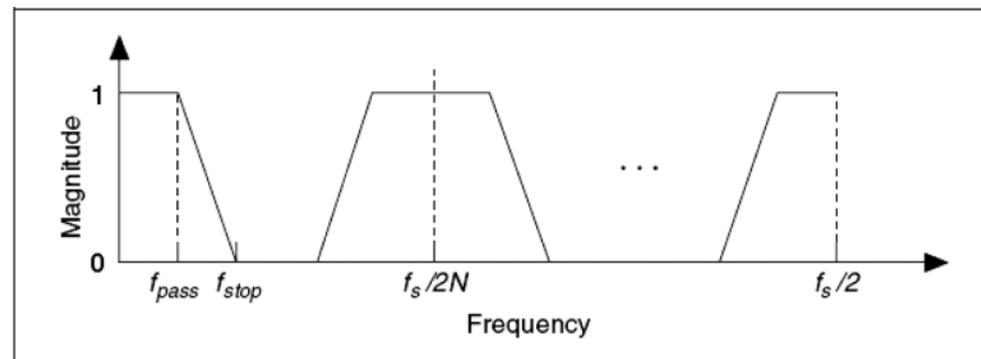
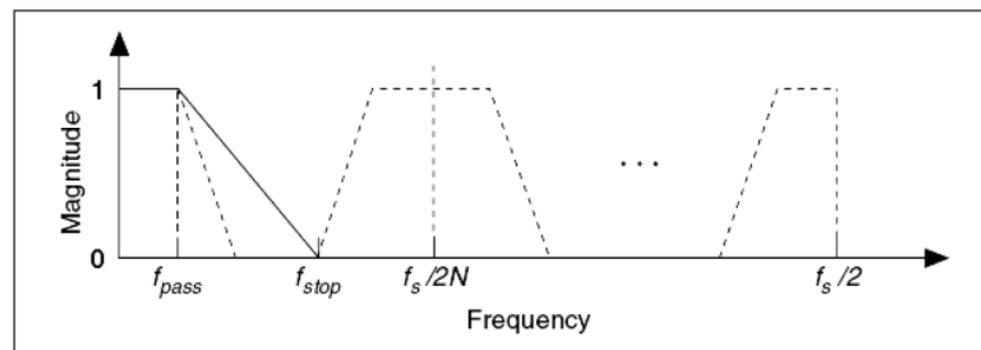
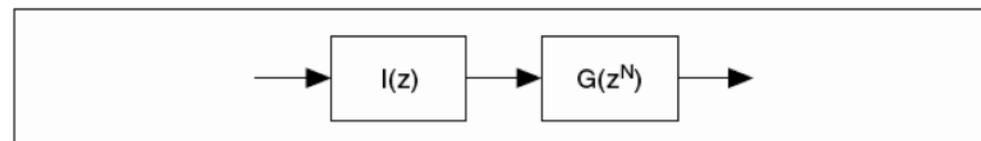
Рис. 16.51. АЧХ интерполяционного фильтра с передаточной функцией  $G(z^N)$ Рис. 16.52. АЧХ фильтра маскировки  $I(z)$ 

Рис. 16.53. Двухступенчатая структура узкополосного фильтра

Точно так же, если ФНЧ-фильтр маски  $I(z)$  имеет узкую полосу, можно делать его более эффективным, используя структуру двухступенчатого узкополосного фильтра, показанную на рис. 16.53. На рис. 16.54 изображена структурная схема результирующего фильтра с тремя каскадами. Гребенчатый каскадный интегратор (Cascaded Integrator Comb – CIC) – фильтр, который используется как первый каскад маски ФНЧ, в этом случае из-за его низкочастотного характера и эффективного выполнения.

Фильтры CIC будут рассмотрены ниже, в подразд. 16.6.

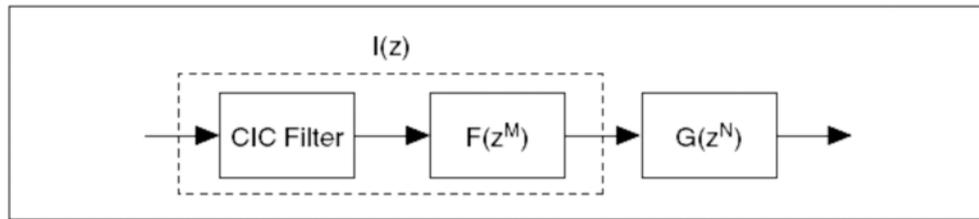


Рис. 16.54. Структура узкополосная фильтра с тремя каскадами

### 16.5.24. Пример узкополосного фильтра

В этом примере частота дискретизации входного сигнала равна 5 кГц. Сигнал имеет полезную информацию на частотах ниже 100 Гц и шум на частотах выше чем 120 Гц. Чтобы подавить шум, можно применить узкополосный фильтр ФНЧ со следующими параметрами.

Параметры	Величина
Диапазон полосы пропускания (Passband Range)	0–100 Гц
Неравномерность в полосе (Passband Ripple)	0,05 дБ
Полоса задержания (Stopband Range)	120–2500 Гц
Затухание в полосе задержания (Stopband Attenuation)	60 дБ

Если проектируется ФНЧ КИХ-фильтр при использовании ВП DFD Remez Design, окончательный проект фильтра использует 689 отсчетов сигнала. Учитывая те же самые параметры, ВП DFD Narrowband Filter Design вырабатывает узкополосный фильтр с тремя каскадами, как показано на рис. 16.54. Окончательный CIC-фильтр имеет 5 каскадов, интерполирующий фильтр  $F(z^M)$  имеет 18 ненулевых коэффициентов, а интерполирующий фильтр  $G(z^N)$  имеет 27 ненулевых коэффициентов. Узкополосный CIC-фильтр на 78 % менее сложен в вычислительном отношении, чем одноступенчатый ФНЧ КИХ-фильтр.

На рис. 16.55 представлена АЧХ разработанного узкополосного фильтра.

Чтобы получить график АЧХ узкополосного КИХ-фильтра, можно использовать ВП DFD Plot Narrowband Freq Response. Для сравнительного анализа узкополосной фильтрации, можно использовать ВП DFD Narrowband Filtering.

## Резюме

1. Цифровые фильтры с линейной ФЧХ позволяют гармоническим составляющим входного сигнала пройти через фильтр с одинаковой задержкой. Можно спроектировать четыре типа фильтров с линейной ФЧХ, используя ВП DFD Remez Design. Можно обратиться к табл. 16.9 для ознакомления со списком типов фильтров и характеристик каждого из них.

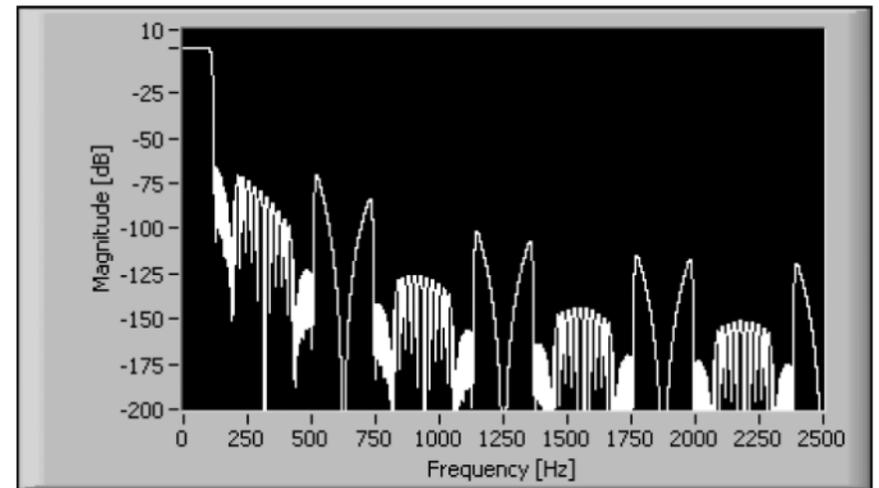


Рис. 16.55. АЧХ узкополосного фильтра

2. Преобразователи Гильберта имеют сдвиг по фазе минус  $90^\circ$  для положительных частот и  $90^\circ$  сдвиг по фазе для отрицательных частот в их передаточной функции. Можно спроектировать преобразователь Гильберта типа IV или III, используя ВП DFD Remez Design. Выбрать при вводе **filter type** положение **Hilbert**.
3. Дифференциаторы имеют постоянный сдвиг по фазе  $90^\circ$  на всех частотах. Можно спроектировать дифференциатор типа IV или III, используя ВП DFD Remez Design. Для этого надо выбрать при вводе **filter type** положение **Differentiator**.
4. Фильтры-пробки подавляют помеху на заданной частоте, а пиковые фильтры усиливают сигнал на специфической частоте. Можно спроектировать фильтр-пробку или максимизирующий фильтр, используя ВП DFD IIR Notch Peak Design.
5. Гребенчатые фильтры подавляют помеху или выделяют сигнал. Например, можно использовать гребенчатый фильтр, чтобы подавить или выделить сигнал на одной основной частоте и на частотах гармоник. Можно спроектировать гребенчатый фильтр, используя ВП DFD IIR Comb Design. Для изучения списка различных типов гребенчатых фильтров можно обратиться к табл. 16.10.
6. Фильтры с произвольной АЧХ позволяют обеспечить произвольную форму частотной характеристики. Можно использовать ВП DFD Remez Design, чтобы спроектировать КИХ-фильтр с линейной ФЧХ и с произвольной формой АЧХ. Можно также использовать ВП DFD Least Pth Norm Design, чтобы спроектировать БИХ-фильтр, в котором линейная ФЧХ не требуется, но важнее результирующий порядок фильтра.

7. Компенсаторы групповой задержки линеаризуют ФЧХ в указанном диапазоне частот при сохранении неизменной АЧХ. Можно спроектировать компенсатор групповой задержки, используя ВП DFD Group Delay Compensator.
8. Узкополосные фильтры трудно проектировать, потому что порядок КИХ-фильтра обратно пропорционально связан с его переходной полосой пропускания. ВП DFD Narrowband Filter Design основан на использовании интерполяционных КИХ-методов и методов маски АЧХ, чтобы проектировать узкополосные КИХ-фильтры со значительно меньшей вычислительной сложностью, чем обычные решения КИХ-фильтров.

## 16.6. Многоскоростные цифровые фильтры

В этом подразделе обсуждается, как проектировать, анализировать и использовать многоскоростные цифровые фильтры, используя ИПЦФ в LabVIEW. Можно применять многоскоростные цифровые фильтры в цифровых системах обработки сигналов, в которых различные частоты дискретизации требуются для различных частей потока отсчетов входного сигнала или для уменьшения вычислительной сложности в системах с однородной частотой дискретизации (см. разд. 8).

*Примечание.* Можно проектировать многоскоростные фильтры с плавающей запятой, используя ИПЦФ, но этот инструментарий в настоящее время не содержит инструментальные средства для проектирования многоскоростных фильтров с фиксированной запятой.

### 16.6.1. Основы многоскоростной цифровой фильтрации

Цифровой многоскоростной фильтр преобразовывает исходную выборку, дискретизированную с одной частотой, к новой частоте дискретизации при минимальных искажениях информации в сигнале. Имеются следующие три варианта изменения частоты дискретизации:

1. Повторная дискретизация – частота дискретизации сигнала преобразовывается в новую частоту, которая отличается от исходной в нецелое число раз. Например, можно использовать повторную дискретизацию, чтобы преобразовать сигнал из профессиональной аудиосистемы, с частотой дискретизации 48 кГц до частоты дискретизации для компакт-диска, равной 44,1 кГц.
2. Децимация (снижение частоты дискретизации) – частота дискретизации преобразована в более низкую частоту, которая отличается от исходной в целое число раз (см. подразд. 8.2).

Некоторая информация в спектральных составляющих с высокой частотой может быть потеряна низкочастотной фильтрацией, связанной с децимацией, но потеряно столько, сколько требуется для приспособления к новой частоте дискретизации.

Децимация может быть полезна в приложениях, где полоса Найквиста для цифрового сигнала намного шире, чем необходимо. Можно использовать многоскоростные цифровые фильтры, чтобы удалить составляющие в избыточной полосе пропускания и уменьшить частоту дискретизации, что приводит к снижению требуемых для обработки и сохранения сигнала вычислительных ресурсов.

3. Интерполяция (увеличение частоты дискретизации) – частота дискретизации преобразована к более высокой величине, которая отличается от исходной в целое число раз (см. подразд. 8.1).

Спектр сигнала на выходе такой же, как спектр входного сигнала, за исключением того, что добавляется высокочастотная область с нулевой плотностью мощности.

Многоскоростные фильтры обычно выполняются на основе использования КИХ-фильтров вместо БИХ-фильтров. КИХ-фильтры могут обеспечить превосходную стабильность, линейность ФЧХ и характеристики конечной точности. Также структуры КИХ-фильтра содержат только прямые пути для сигнала, что позволяет упростить выполнение децимации и интерполяции при фильтрации из-за того, что можно уменьшить число сомножителей, требуемое для перемножения при интерполяции или децимации.

### 16.6.2. Повторная дискретизация

Повторная дискретизация – процесс преобразования данных, дискретизированных с одной частотой к новой частоте, связанной с исходной заданным множителем. Рациональная повторная дискретизация, по которой прежняя частота и новая частота отличаются мультипликативным рациональным множителем, является обычной методикой повторной дискретизации. Например, если входной сигнал дискретизирован с частотой 100 Гц, а новая заданная частота – 150 Гц, эти две частоты связаны через мультипликативный рациональный коэффициент, равный 150/100, что эквивалентно 3/2. В этом случае можно повторно произвести дискретизацию данных первой интерполяцией к 3 и затем децимацию на 2.

Когда повторно производится дискретизация, интерполяция и затем децимация, многие из интерполированных единичных отсчетов данных отвергаются. Использование ВП Align and Resample позволяет произвести повторную дискретизацию за один шаг.

### 16.6.3. Децимация

Децимация есть процесс уменьшения частоты дискретизации сигнала до значения, связанного целочисленно с исходной частотой дискретизации. На рис. 16.56 показан типичный фильтр децимации на  $M$  сверток, который содержит ФНЧ КИХ-фильтр  $H(z)$ , сопровождаемый дециматором на  $M$  сверток. Дециматор передает на выход каждый  $M$ -й отсчет и отбрасывает остальные отсчеты, чем изменяется частота дискретизации  $f_s$  на  $f_s/M$ .

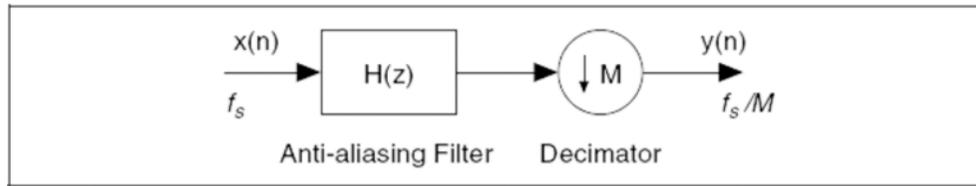


Рис. 16.56. Фильтр децимации

ФНЧ КИХ-фильтр с передаточной функцией  $H(z)$ , расположенный перед дециматором (decimator), является фильтром (Anti-aliasing Filter), исключаяющим высокочастотные составляющие спектра с частотами выше частоты Найквиста. Чтобы исключить совмещение спектральных составляющих, эта система использует ФНЧ  $H(z)$ , фильтрующий сигнал перед дециматором на  $M$  сверток, чтобы подавить составляющие с частотами выше частоты  $f_s/2M$ . В результате остаются составляющие спектра в диапазоне частот от 0 до  $f_s/2M$ . Эта система обеспечивает те же самые результаты, что и аналоговый фильтр сглаживания с частотой среза  $f_s/2M$ , сопровождающий АЦП с частотой дискретизации  $f_s/M$ . Поскольку система, показанная на рис. 16.56, выполняется в цифровой области  $H(z)$  называется передаточной функцией цифрового фильтра сглаживания.

На рис. 16.57 представлены потенциально вредные эффекты, которые проявляются, если не использовать фильтр сглаживания перед дециматором. На этом рисунке показан спектр исходного цифрового сигнала с частотой дискретизации  $f_s$  и спектры сигналов, получающихся после непосредственной децимации исходного сигнала в 2, 3 и  $M$  раз. Необходимо обратить внимание на перекрывающиеся по частоте изображения на графиках рис. 16.57, б, в и г.

### 16.6.4. Пример фильтра децимации

Можно использовать ВП Multirate Filter Design при установке ввода режима фильтрации (**filtering mode**) в положение децимации (**Decimation**), чтобы спроектировать фильтр децимации. Например, рассмотрим синусоидальное колебание с частотой 40 Гц при частоте дискретизации 500 Гц, которое загрязнено высокочастотным шумом. Цель состоит в том, чтобы удалить высокочастотный шум и проредить отсчеты сигнала с коэффициентом 5. Чтобы создать соответствующий фильтр децимации, можно использовать ВП DFD Nyquist Design со следующими параметрами фильтра

Вводы ВП DFD Nyquist Design	Величина
Коэффициент (factor)	5
Склон АЧХ (roll off)	0,1
Затухание в полосе задержания (Stopband attenuation)	80 дБ
Режим фильтрации (filtering mode)	Децимация (Decimation)

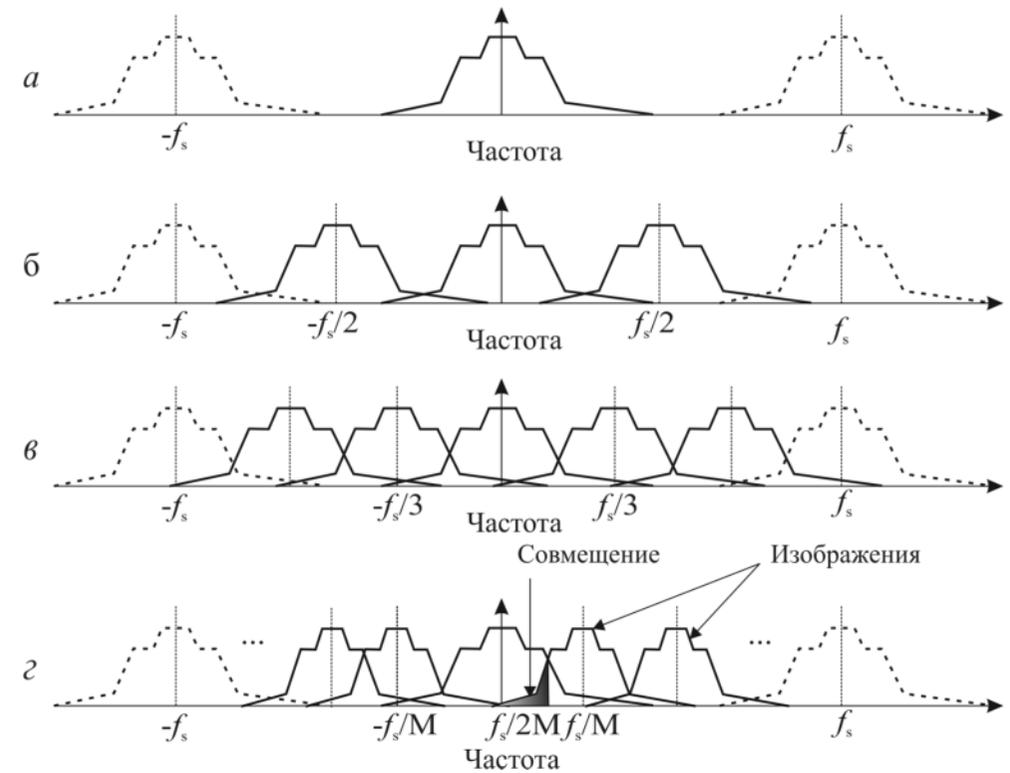


Рис. 16.57. Децимация без фильтра сглаживания:

а – спектр исходного сигнала; б – спектр после прямой децимации исходного сигнала в 2 раза; в – спектр после прямой децимации исходного сигнала в 3 раза; г – спектр после прямой децимации исходного сигнала в  $M$  раз

Если используется значение порядка (**order**) по умолчанию, равное 1, то ВП DFD Nyquist Design использует ввод ослабления в полосе затухания (**Stopband attenuation**), чтобы оценить порядок фильтра. Для получения информации о ВП DFD Nyquist Design можно обратиться к *LabVIEW Help*.

### 16.6.5. Интерполяция

Интерполяция – процесс увеличения частоты дискретизации сигнала к более высокому значению, целочисленно связанному с исходной частотой дискретизации. На рис. 16.58 показан обычный фильтр интерполяции на  $M$  сверток, где  $M$  – заданное целочисленное увеличение частоты дискретизации.

Система содержит расширитель (экспандер) (Expander) на  $M$  сверток перед ФНЧ, КИХ-фильтр с передаточной функцией  $H(z)$  – фильтр обратного отображения отсчетов. Экспандер на  $M$  сверток вставляет  $M - 1$  нулей между отсчетами

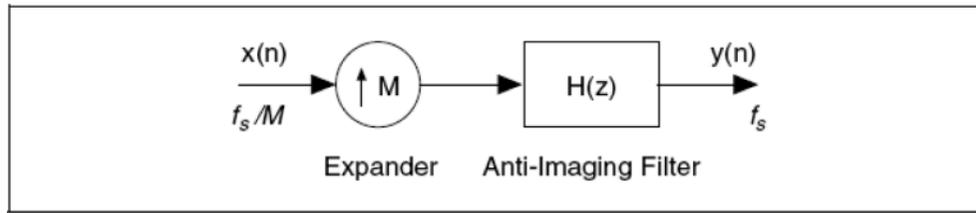


Рис. 16.58. Интерполяционный фильтр

в последовательности для исходного сигнала. Такая операция изменяет частоту дискретизации  $f_s/M$  на частоту  $f_s$ .

На рис. 16.59 показан спектр исходного цифрового сигнала, дискретизированного с частотой  $f_s/M$ , и спектры сигналов от прямой интерполяции исходного сигнала в 2, 3 и  $M$  раз без фильтра обратного отображения отсчетов, показанного на

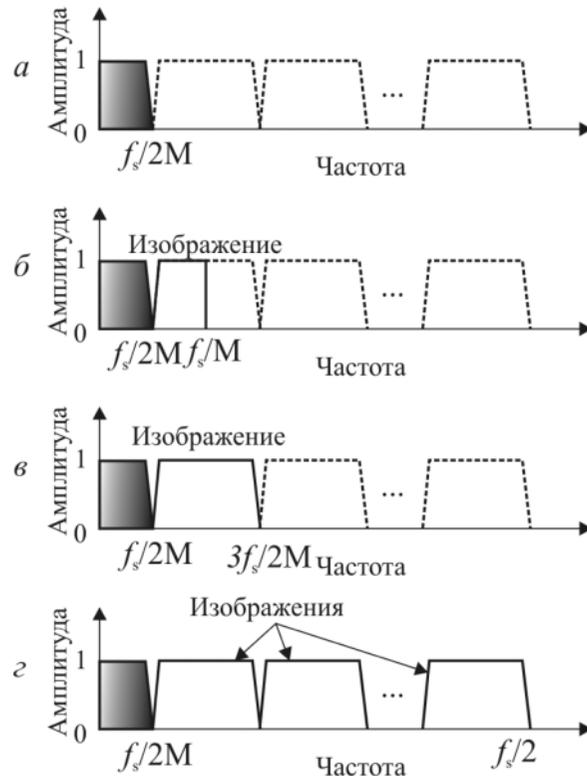


Рис. 16.59. Интерполяция без фильтра обратного отображения отсчетов:  
 а – спектр исходного сигнала; б – спектр после прямой интерполяции исходного сигнала в 2 раза; в – спектр после прямой интерполяции исходного сигнала в 3 раза; г – спектр после прямой интерполяции исходного сигнала в  $M$  раз

рис. 16.58. Необходимо обратить внимание на множественные изображения, появляющиеся в диапазоне от 0 до половины частоты дискретизации выходного сигнала на рис. 16.59, б, в, и г, которые демонстрируют эффект интерполяции за счет вставления нулевых значений.

Чтобы устранить отображения (images – см. рис. 16.59) спектра, эта система использует ФНЧ с передаточной функцией  $H(z)$  после экспандера, чтобы устранить отображения в диапазоне от  $f_s/2M$  до  $f_s/2$ . В области времени эффект использования  $H(z)$  должен заменить выборки чередованием исходных значений нулями, введенными экспандером для решения задачи интерполяции.

При замене вставленных нулей интерполированными значениями ФНЧ обратного отображения отсчетов  $H(z)$  может изменить исходные значения. Можно использовать интерполяционный фильтр Найквиста вместо фильтра с передаточной функцией  $H(z)$ , чтобы сохранить исходные значения. Характеристики фильтра Найквиста (Котельникова) в области времени гарантируют, что исходные значения не изменились в результате интерполяции.

### 16.6.6. Пример интерполирующего фильтра

Можно использовать ВП Multirate Filter Design при вводе режима фильтрации (**filtering mode**), устанавливая на интерполяции (**Interpolation**), чтобы проектировать интерполирующие фильтры. Например, рассмотрим устройство сбора данных, которое непрерывно получает множественные блоки сигнала с частотой дискретизации 500 Гц. Цель состоит в том, чтобы интерполировать блоки сигнала интерполирующим КИХ-фильтром с коэффициентом 5 и сохранить исходные единичные отсчеты после интерполяции. Чтобы создать интерполирующий фильтр, можно использовать ВП DFD Nyquist Design со следующими параметрами:

Вводы ВП DFD Nyquist Design	Величина
Коэффициент (factor)	5
Склон АЧХ (roll off)	0,1
Затухание в полосе задержания (Stopband attenuation)	80 дБ
Режим фильтрации (filtering mode)	Сглаживание (Interpolation)

Если использовать значение порядка (**order**) по умолчанию, равное 1, то ВП DFD Nyquist Design использует ввод величины ослабления в полосе задержания (**stopband attenuation**), чтобы оценить порядок фильтра. Для получения информации о ВП DFD Nyquist Design можно обратиться к *LabVIEW Help*.

Можно использовать ВП DFD MRate Filtering, чтобы непрерывно обрабатывать блоки сигнала. На рис. 16.60 показан входной и интерполированный сигналы после фильтрации. Необходимо установить нулевую ФЧХ (**zero phase?**) при вводе в положение «ИСТИНА» (TRUE), чтобы не было никакой задержки между сигналами входа и выхода.

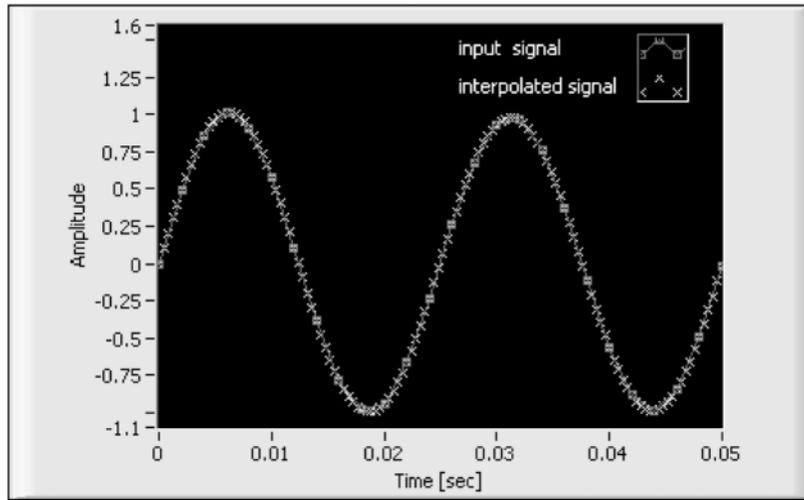


Рис. 16.60. Фильтрация с интерполяцией при нулевой ФЧХ без задержки: input signal – входной сигнал; interpolated signal – интерполированный сигнал

### 16.6.7. Многоскоростные многоступенчатые фильтры

Для одноступенчатых многоскоростных фильтров, рассмотренных выше, полоса перехода ФНЧ КИХ-фильтра с передаточной функцией  $H(z)$  обратно пропорционально связана с коэффициентом  $M$  децимации или интерполяции. Чем больше  $M$ , тем более узкой должна быть полоса перехода. Требуемый порядок ФНЧ  $H(z)$  растет с возрастанием  $M$ , чтобы сохранить тот же самый уровень подавления совмещения спектра или отображения. Для большего значения  $M$  необходим больший порядок ФНЧ с передаточной функцией  $H(z)$ , который может стать слишком дорогим при реализации.

Многоступенчатые фильтры более эффективны, чем одноступенчатые фильтры. Они позволяют постепенно увеличить или уменьшить частоту дискретизации, передавая сигнал через две или больше ступени повторной дискретизации. Каждая ступень имеет более низкую децимацию или коэффициент интерполяции, который дает возможность требуемой полосе перехода каждой ступени быть существенно уменьшенной. Поскольку каждая ступень содержит меньше операций, можно значительно уменьшить порядок фильтра каждой ступени и, таким образом, снизить сложность системы.

### 16.6.8. Многоступенчатый фильтр децимации

Чтобы проектировать многоступенчатые фильтры, можно использовать ВП DFD NStage MRate Filter Design. Например, предположим, что необходимо проредить отсчеты сигнала с коэффициентом 50 при исходной частоте дискретизации – 100 кГц, а фильтр сглаживания имеет следующие параметры:

- полоса пропускания [0, 100] Гц;
- полоса задержания [800 Гц, 50 кГц];
- неравномерность в полосе пропускания 0,01;
- ослабление в полосе задержания 60 дБ.

Можно использовать ВП DFD NStage MRate Filter Design со следующими параметрами фильтра, чтобы создать многоступенчатый фильтр децимации.

Входы ВП DFD NStage MRate Filter	Значения
Ручное разложение на множители (manual factorization)	$M1 = 10, M2 = 5$
Коэффициент (factor)	50
Граничные частоты (freq specs)	
Нижняя граничная частота (fpass)	100 Гц
Верхняя граничная частота (fstop)	800 Гц
Неравномерности (ripple specs)	
в полосе пропускания (pass band)	0,01
в полосе задержания (stop band)	0,001
Частота дискретизации ( $f_s$ )	100 кГц
Вид фильтрации (filtering mode)	Децимация (Decimation)

Этот фильтр содержит два каскада. Первый каскад прореживает отсчеты с коэффициентом 10 и второй каскад прореживает отсчеты с коэффициентом 5. Оба каскада способствуют полной децимации с коэффициентом 50.

**Примечание.** Общий результат ввода всех коэффициентов при ручном разложении на множители (**manual factorization**) должен равняться вводимому значению коэффициента (**factor**).

Можно использовать ВП DFD Plot NStage MRate Freq Response для анализа АЧХ разработанного многоступенчатого фильтра децимации, как показано на рис. 16.61.

### 16.6.9. Гребенчатые каскадные фильтры-интеграторы (CIC)

Можно реализовать многоскоростные фильтры на основе эффективного использования гребенчатых каскадных фильтров-интеграторов (CIC). Фильтры CIC ФНЧ имеют следующий тип функции преобразования:

$$H_L(z) = H_I^N(z) H_C^N(z) = \frac{(1 - z^{-RM})^N}{(1 - z^{-1})^N} = \left( \sum_{k=0}^{RM-1} z^{-k} \right)^N, \quad (16.12)$$

где  $M$  – коэффициент преобразования частоты дискретизации;  $R$  – дифференциальная задержка;  $N$  – число каскадов.  $R$  может иметь любое положительное целочисленное значение, однако  $R$  обычно ограничивается 1 или 2.

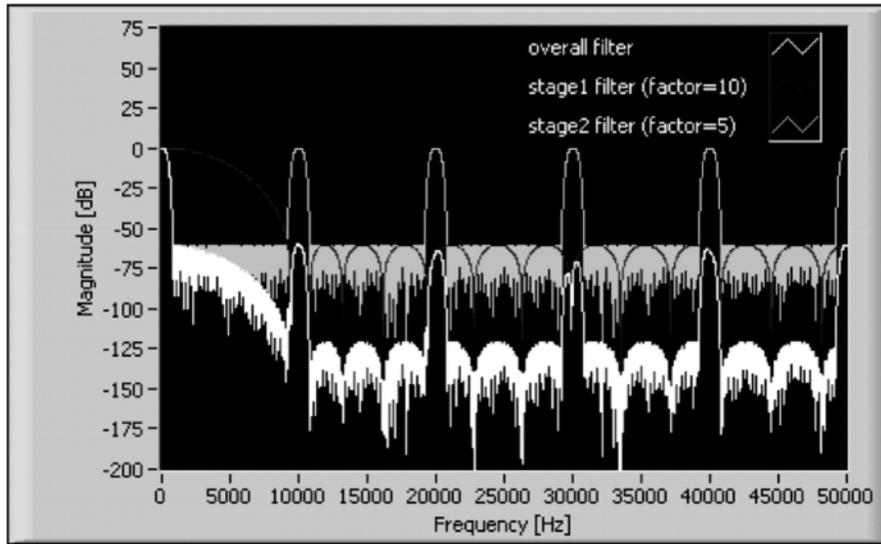


Рис. 16.61. АЧХ многоступенчатого фильтра децимации:

overall filter – всепропускающий фильтр; stage 1 filter (factor = 10) – первый каскад фильтра (коэффициент = 10); stage 2 filter (factor = 5) – второй каскад фильтра (коэффициент = 5)

Уравнение (16.12) показывает, что фильтр СИС эквивалентен  $N$ -каскадному КИХ-фильтру. Каждый КИХ-фильтр имеет прямоугольную импульсную передаточную функцию. Все коэффициенты КИХ-фильтров одинаковы и поэтому симметричны, так что фильтр СИС имеет линейную ФЧХ и постоянную групповую задержку. Фильтры СИС не требуют операции умножения, так что можно создать их эффективно на основе алгоритмов с фиксированной запятой.

В некоторых приложениях может появиться необходимость в фильтре ФВЧ. Например, нужен ФВЧ, в котором заданная полоса пропускания является смежной с частотой Найквиста, а не примыкает к постоянной составляющей. Передаточная функция ФВЧ СИС-фильтра представлена уравнением

$$H_H(z) = H_I^N(z)H_C^N(z) = \frac{(1 - (-z)^{-RM})^N}{(1 - (-z)^{-1})^N} = \left( \sum_{k=0}^{RM-1} (-z)^{-k} \right)^N. \quad (16.13)$$

Можно использовать ВП DFD NStage MRate Filter Design, устанавливая ввод СИС? в положение TRUE для проектирования многоскоростного фильтра с тремя каскадами, один каскад которого – фильтр СИС.

### 16.6.10. Фильтр с нулевой ФЧХ

Все многоскоростные фильтры, проектируемые в ИПЦФ, исключая, по возможности, фильтры СИС нечетного порядка, имеют четный порядок и линейную ФЧХ

КИХ-фильтра. При линейной ФЧХ выходные сигналы КИХ-фильтра имеют постоянную групповую задержку на число отсчетов, равное половине порядка фильтра, как показано на рис. 16.62.

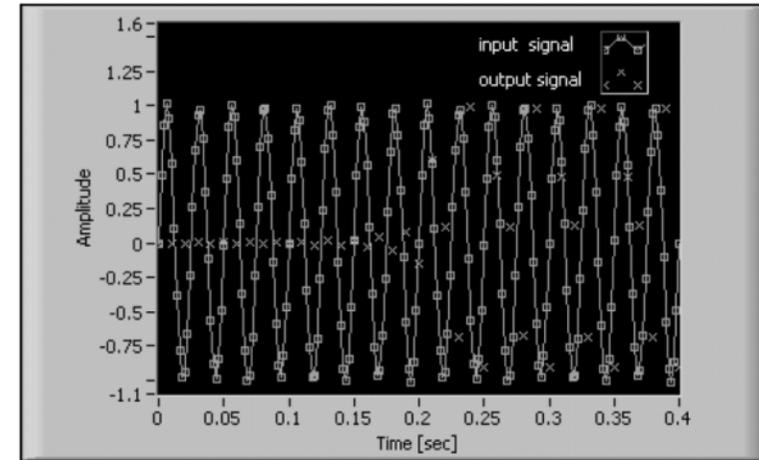


Рис. 16.62. Фильтрация с задержкой

Если необходимо устранить задержку между сигналами входа и выхода, можно разработать фильтр с нулевой ФЧХ. Чтобы выбрать нулевую опцию ФЧХ, надо установить при вводе опцию **zero phase?** ВП DFD MRate Filtering for Single Block или ВП DFD MRate Filtering for Single Block в положение TRUE, тогда сигнал на выходе не будет иметь никакой задержки по сравнению с входным сигналом, как представлено на рис. 16.63.

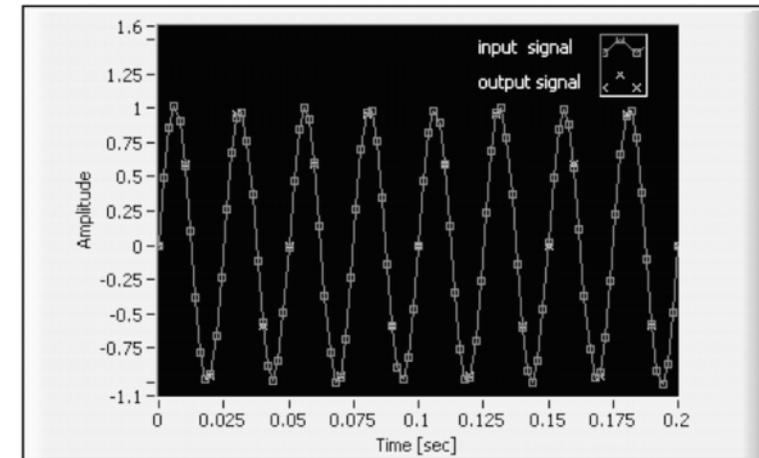


Рис. 16.63. Выходной и входной сигналы для фильтра с нулевой ФЧХ

Поскольку нетривиальные фильтры с нулевой ФЧХ не должны быть каузальными, невозможно обеспечить нулевую ФЧХ при обработке сигналов в реальном масштабе времени. ВП Multirate Processing обеспечивают нулевую ФЧХ путем подбора и настройки данных. Для однокаскадной обработки в ВП дополняются входной блок данных в начале и в конце и урезаются выходные данные так, чтобы задержка между входом и выходом была нулевой. Для непрерывной обработки в ВП урезают начальный переход так, что задержка между входным и выходным сигналами была нулевой.

Если установить на вводе **zero phase?** в положение TRUE, необходимо также установить **extension type**, который определяет, каким способом в ВП дополняется блок входных данных. Имеется следующий выбор для **extension type**:

- опция **zero padding** использует нули для дополнения входных данных. При этом обеспечивается управление резким переходом между дополняемыми нулями и входными данными, что вызывает большие искажения вблизи перехода;
- опция **periodic** добавляет для дополнения данных одну копию блока отсчетов входного сигнала до и другую копию после этого блока отсчетов;
- опция **symmetric** также используется для дублирования входных данных, чтобы их дополнить, за исключением того, что эти ВП ставят зеркально отраженный влево блок в начале исходной выборки и зеркально отраженный вправо блок – в конце этой выборки.

**Примечание.** Многоскоростная фильтрация с нулевой ФЧХ обеспечивается в фильтрах с четными порядками. Все проектируемые с помощью ИПЦФ многоскоростные фильтры – четного порядка, кроме фильтров CIC – нечетного порядка.

### 16.6.11. Разработка многоскоростного фильтра

В этом пункте описывается, как проектировать одноступенчатые многоскоростные фильтры, многоступенчатые многоскоростные фильтры и фильтры Найквиста на основе использования ИПЦФ. Можно обратиться к NI Example Finder для поиска примеров проектирования многоскоростных фильтров.

### 16.6.12. Одноступенчатый проект многоскоростного фильтра

На рис. 16.64 приведено возможное наложение спектра сигнала при изменении частоты дискретизации, которое следует из перекрытия с его зеркальным спектром (см. разд. 3).

Когда проектируются одноступенчатые фильтры, надо учесть вид режима фильтрации для обеспечения параметров фильтра. Для заданного коэффициента децимации/интерполяции  $M$  можно использовать следующие параметры фильтра:

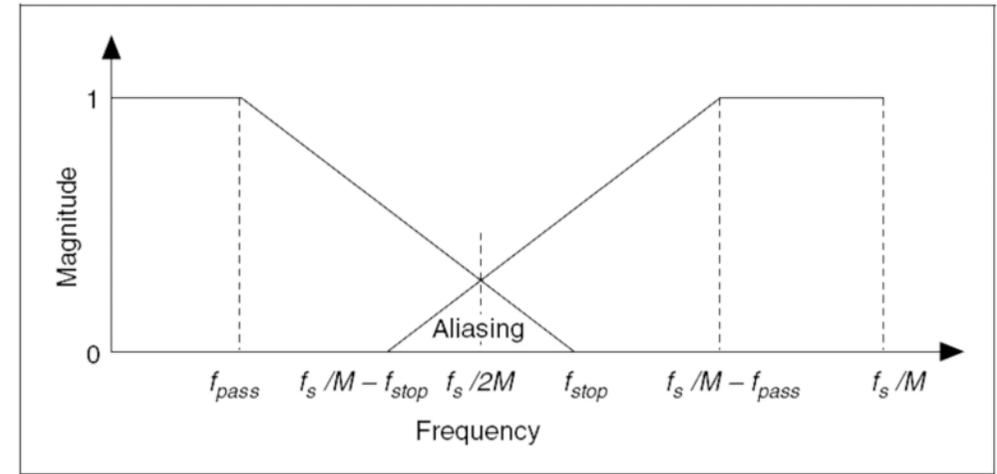


Рис. 16.64. Совмещение спектра (Aliasing) при изменении частоты дискретизации

Параметр фильтра (Filter Specification)	Величина (Value)
Диапазон полосы пропускания (Passband Range)	$[0, f_{pass}]$ , где $0 < f_{pass} < f_s/2M$
Диапазон полосы задержания (Stopband Range)	$[f_{stop}, f_s/2]$ , где $f_{pass} < f_{stop} < f_s/M - f_{pass}$

Как правило,  $f_{pass}$  – самая высокая частота, представляющая интерес в исходном сигнале.

Если  $f_{stop} < f_s/2M$ , то полоса перехода свободна от наложения спектров. Если  $f_s/M$  соответствует условию  $f_{pass} > f_{stop} > f_s/2M$ , то полоса перехода содержит наложение спектров. Если наложение спектров проявляется в полосе перехода, то полоса предварительного фильтра слишком широкая и необходимо использовать фильтры более низкого порядка, что может значительно уменьшить вычислительную сложность в операциях фильтрации.

Можно использовать следующие рекомендации, чтобы решить, можно ли позволить совмещение спектров в полосе перехода:

- если фильтр децимации – однокаскадный цифровой фильтр в системе и необходимо, чтобы сигнал на выходе был свободным от совмещения спектров, надо обеспечить такие параметры для фильтра децимации, чтобы сделать этот фильтр достаточным для предотвращения совмещения спектров;
- если имеются другие каскады ДСП-фильтров, которые следуют за фильтром децимации, для удаления шума вне заданного диапазона частот можно спроектировать фильтр с совмещением спектров в полосе перехода;
- если полоса перехода не важна, можно спроектировать фильтр с совмещением спектров в полосе перехода. Например, если известно, что сигналы не имеют никакой спектральной мощности на этих частотах, нет необходимости беспокоиться об исключении совмещения спектров.

Для получения информации об использовании ВП DFD MRate Filter Design при проектировании одноступенчатых многоскоростных фильтров можно обратиться к *LabVIEW Help*.

### 16.6.13. Проектирование многоступенчатых многоскоростных фильтров

Многокаскадная фильтрация более эффективна, чем однокаскадная, потому что можно изменить частоту дискретизации во многих каскадах одновременно, а не только в одном каскаде, кроме того случая, когда коэффициент преобразования частоты дискретизации является простым числом.

Использование многих каскадов уменьшает число вычислительных операций и использование памяти.

В многокаскадной системе децимации полный коэффициент  $M$  децимации разлагается на множители, как  $M = M_1 M_2 \dots M_N$ , где  $M_i$  – коэффициент децимации  $i$ -го каскада.

На рис. 16.65 показан такой процесс децимации в  $N$  каскадах.

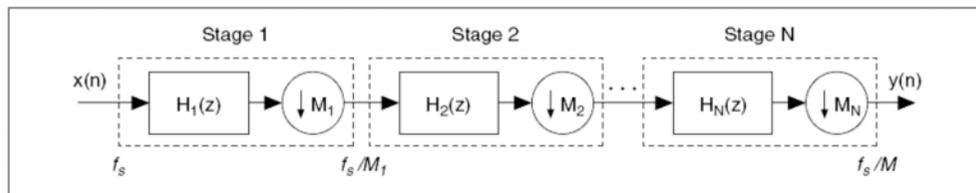


Рис. 16.65. Многокаскадная архитектура фильтра децимации

На рис. 16.66 иллюстрируется аналогичный  $N$ -каскадный процесс интерполяции.

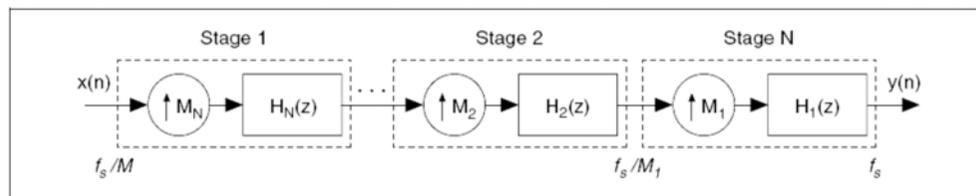


Рис. 16.66. Многокаскадная архитектура интерполяционного фильтра

При проектировании многокаскадных многоскоростных фильтров можно использовать ВП DFD NStage MRate Filter Design. Этот ВП воспринимает некоторые параметры проектов однокаскадных многоскоростных фильтров, производит дополнительные операции, необходимые для разработки многокаскадных фильтров, включая фильтры, передаточные функции которых представлены в виде произведения  $M$  сомножителей, прозрачных для пользователя. Альтернативно мож-

но использовать ВП DFD NStage MRate Filter Design, чтобы разложить вручную на множители передаточную функцию многоскоростного фильтра.

При разложении передаточной функции на сомножители вручную можно опираться на следующие рекомендации:

- использовать два или три каскада для оптимальных или подоптимальных результатов;
- использовать для самой высокой частоты дискретизации наибольший коэффициент. Производить децимацию в порядке от наибольшего к наименьшему коэффициенту и интерполировать в последовательности от наименьшего к наибольшему коэффициенту.

Для получения информации об использовании ВП DFD NStage MRate Filter Design при проектировании многоступенчатых многоскоростных фильтров можно обратиться к *LabVIEW Help*.

### 16.6.14. Проектирование фильтра Найквиста

Фильтры Найквиста (Nyquist) имеют следующие параметры АЧХ:

Параметр фильтра (Filter Specification)	Величина (Value)
Диапазон полосы пропускания (Passband Range)	$[0, f_s/2M - \epsilon]$
Диапазон полосы задержания (Stopband Range)	$[f_s/2M + \epsilon, f_s/2]$

Можно косвенно определить параметр  $\epsilon$  по склону АЧХ, который определен как  $\alpha = \frac{\epsilon}{f_s/(2M)}$ .

На рис. 16.67 представлена АЧХ фильтра Найквиста.

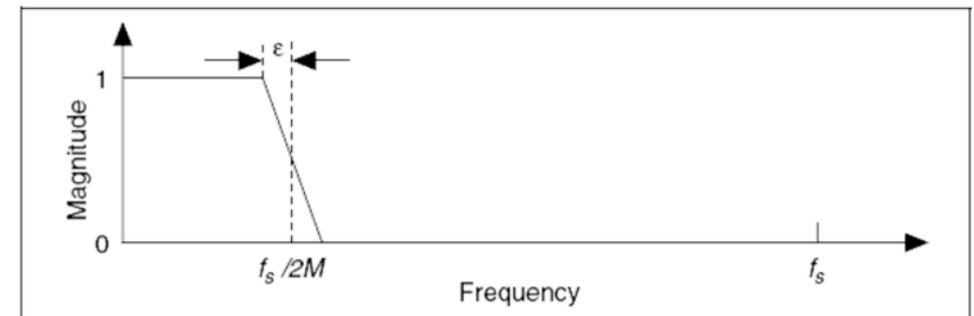


Рис. 16.67. АЧХ фильтра Найквиста

ИПЦФ обеспечивает три ВП для того, чтобы создать фильтры Найквиста: DFD Nyquist Design, DFD Raised Cosine Design и DFD Halfband Design. Для получения информации об использовании этих ВП можно обратиться к *LabVIEW Help*.

### 16.6.15. Фильтры Найквиста

Фильтры Найквиста, также называемые фильтрами полосы  $M$ , являются специальным типом многоскоростного КИХ-фильтра. Коэффициенты фильтра Найквиста имеют периодические нули через каждые  $M$  значений за исключением средних коэффициентов. На рис. 16.68 показаны коэффициенты фильтра Найквиста с отношением частот дискретизации, равным 4.

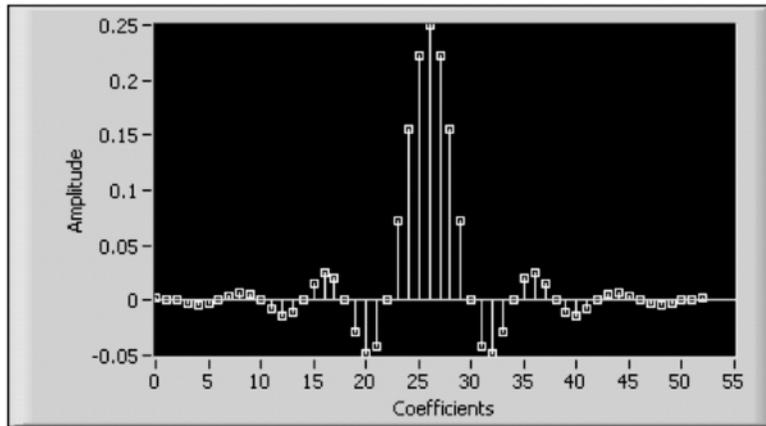


Рис. 16.68. Коэффициенты фильтра Найквиста ( $M = 4$ )

Можно использовать фильтры Найквиста, чтобы удалить отображения спектра при интерполяции. Фильтры Найквиста изменяют интерполированные нули, но не изменяют исходные выборки.

Z-преобразование ( $z$ -transform) фильтра Найквиста  $H(z)$  удовлетворяет следующему уравнению:

$$\sum_{k=0}^{M-1} H(zW^k) = Mc = 1, \quad (16.14)$$

где  $W = e^{-j2\pi/M}$  и  $c = 1/M$ . Передаточная функция  $H(zW^k)$  есть сдвинутая копия передаточной функции  $H(z)$ , так что передаточные функции всех  $M$  равномерно сдвинутых копий  $H(z)$  добавляются через постоянное число отсчетов по частоте.

Когда проектируются фильтры с помощью ВП DFD Nyquist Design, параметры – коэффициент (**factor**) и склон АЧХ (**roll off**) – вводятся вместо полосы пропускания и полосы задержания, как с другими ВП Filter Design.

### 16.6.16. Фильтры «поднятый косинус»

Фильтр «поднятый косинус» – специальный случай фильтра Найквиста. Как и для других фильтров Найквиста, коэффициенты этого фильтра имеют периоди-

ческое нулевое значение через каждый  $M$ -й коэффициент кроме средних коэффициентов.

Идеальная передаточная функция фильтра «поднятого косинуса» состоит из единичных коэффициентов на низких частотах, значений, по величине соответствующих поднятой форме косинуса в середине, и нулевых коэффициентов на высоких частотах. АЧХ фильтра поднятого косинуса математически описывает следующее уравнение:

$$H(f) = \begin{cases} 1 & \text{при } f \leq f_c(1-\alpha); \\ \frac{1 + \cos\left(\frac{\pi(f - f_c(1-\alpha))}{2\alpha f_c}\right)}{2} & \text{при } f_c(1-\alpha) \leq f \leq f_c(1+\alpha); \\ 0 & \text{при } f > f_c(1+\alpha), \end{cases} \quad (16.15)$$

где  $f_c$  – частота среза и  $\alpha$  – склон АЧХ, который удовлетворяет неравенству  $0 \leq \alpha \leq 1$ .

В цифровых системах связи, если необходимо разделить передаточную функцию фильтра «поднятый косинус», равномерно между фильтром передатчика и фильтром приемника, надо использовать квадратный корень из передаточной функции фильтра «поднятого косинуса». АЧХ корня из передаточной функции фильтра поднятого косинуса дается в уравнении

$$H(f) = \begin{cases} 1 & \text{при } f \leq f_c(1-\alpha); \\ \sqrt{\frac{1 + \cos\left(\frac{\pi(f - f_c(1-\alpha))}{2\alpha f_c}\right)}{2}} & \text{при } f_c(1-\alpha) \leq f \leq f_c(1+\alpha); \\ 0 & \text{при } f > f_c(1+\alpha). \end{cases} \quad (16.16)$$

Для получения информации об использовании ВП DFD Raised Cosine Design при проектировании фильтра «поднятого косинуса» и корня из передаточной функции фильтра «поднятого косинуса» можно обратиться к *LabVIEW Help*.

### 16.6.17. Фильтры полуполосы (halfband)

Фильтры полуполосы – фильтры Найквиста, разработанные определенно для  $M = 2$ . Почти половина коэффициентов фильтра в фильтре полуполосы – нули, которые значительно уменьшают объем вычислений, требуемых для фильтрации. Для получения информации об использовании ВП DFD Halfband Design при проектировании фильтров полуполосы можно обратиться к *LabVIEW Help*.

## 16.6.18. Анализ многоскоростных фильтров

Можно использовать ВП Multirate Filter Analysis для анализа одноступенчатых и многоступенчатых многоскоростных фильтров. Для получения информации о ВП Multirate Filter Analysis можно обратиться к *LabVIEW Help*.

## 16.6.19. Обработка сигналов в многоскоростном фильтре

Можно использовать ВП Multirate Processing для обработки сигналов в многоскоростных фильтрах, разработанных с помощью ИПЦФ. ВП Multirate Processing могут обработать данные входного сигнала тремя способами:

- как единственный блок данных;
- как последовательность блоков данных;
- как последовательность блоков данных с сохраненными внутренними установками фильтра. Используется для обработки отдельного блока данных ВП DFD MRate Filtering for Single Block или ВП DFD NStage MRate Filtering for Single Block.

При обработке отдельного блока данных ВП расширяют входной блок сигнала с обоих концов, чтобы гарантировать такую же длину блока выходного отфильтрованного сигнала.

Можно использовать другой ВП Multirate Processing, чтобы непрерывно обрабатывать множественные блоки сигнала. Этот ВП автоматически сохраняет внутренние установки фильтра между блоками и позволяет сохранять и восстанавливать установки фильтра, не приводя к ложным выбросам в обработанных данных.

Можно достичь тех же самых результатов, когда множественные блоки обрабатываются вместе как один единственный блок.

## Резюме

1. Многоскоростной фильтр преобразовывает выборки данных от исходной частоты к новой частоте дискретизации. Можно использовать децимацию или методы интерполяции для преобразования частоты дискретизации к новой частоте.
2. При проектировании многоскоростных фильтров используют тот же самый процесс проектирования, как и для базовых фильтров:
  - ввод параметров фильтра на основе использования ВП Multirate Filter Design;
  - анализ характеристик многоскоростного фильтра, используя ВП Multirate Filter Analysis;
  - использование многоскоростных фильтров для приложений обработки сигналов на основе ВП Multirate Processing.
3. Можно создавать фильтры децимации и интерполяции, устанавливая режимы фильтрации (**filtering mode**) на входе в ВП Multirate Filter Design.

4. Можно создать многокаскадные многоскоростные цифровые фильтры, используя ВП DFD NStage MRate Filter Design. Многокаскадная фильтрация более эффективна, чем однокаскадная фильтрация, потому что можно изменить частоту дискретизации в нескольких каскадах, а не в одном каскаде. Когда конфигурируется многокаскадный фильтр, используются два или три каскада, и наибольший коэффициент соответствует самой высокой частоте дискретизации.
5. Фильтры Найквиста обеспечивают гарантию, что при интерполяции не изменяются исходные значения.

## 16.7. Современные методы проектирования фильтров

В этом подразделе приведен краткий обзор современных методов для проектирования цифровых фильтров в ВП DFD Remez Design и ВП DFD Least Pth Norm Design. Если есть необходимость познакомиться с методами или понятиями, обсуждаемыми в этом подразделе, можно обратиться к материалу *Digital Filter Design* by Parks and Burrus [31] для получения дополнительной информации.

### 16.7.1. Проектирование методом Ремеза (Remez)

Алгоритм Ремеза формирует КИХ-фильтры, при синтезе которых минимизируется максимальная ошибка между заданной АЧХ и АЧХ разработанного фильтра. Такое определение ошибки приводит к формированию фильтров с равномерной пульсацией АЧХ или к определению Чебышевской ошибки.

Несколько ВП Filter Design используют алгоритм Ремеза. ВП DFD Remez Design расширяет приложение алгоритма Ремеза, чтобы обеспечить разработку расширенных, специализированных проектов цифровых фильтров следующего типа:

- проектирование КИХ-фильтра типов I–IV с линейной ФЧХ;
- проектирование дифференциаторов;
- проектирование преобразователей Гильберта;
- проектирование КИХ-фильтров, имеющих произвольную форму АЧХ;
- проектирование фильтров с АЧХ, приближенной к оптимальной (минимально-фазовый или максимально-фазовый);
- проектирование по параметрам единственной точки по частоте (фильтр-пробка или пик-фильтр);
- точная установка значений передаточной функции;
- параметры ограничения по пульсациям АЧХ в полосе.

### 16.7.2. Использование ВП Remez Design

Чтобы проектировать цифровой фильтр, используя ВП DFD Remez Design, необходимо ввести параметры для заданной АЧХ, порядок фильтра и тип фильтра.

Ниже поясняются специальные соображения при вводе параметров фильтра. Для получения окончательной ссылочной информации о ВП DFD Remez Design можно обратиться к *LabVIEW Help*.

### 16.7.3. Определение заданной передаточной функции

Можно использовать массив параметров полосы (**band specs**), чтобы определить заданную передаточную функцию фильтра. Каждый элемент в массиве представляет одно значение этой функции по частоте. Можно ввести одну или более точек в возрастающем порядке, чтобы описать поведение передаточной функции в каждой полосе частот. ВП DFD Remez Design связывает все точки, чтобы сформировать непрерывную идеальную передаточную функцию в полосе частот. Интервал по частоте между двумя последовательными полосами есть полоса перехода.

Определяется каждая точка частотой, амплитудой и весовым коэффициентом. Весовой коэффициент подчеркивает относительную важность приблизительного уровня пульсаций АЧХ.

Относительно более высокие значения весового коэффициента приводят к меньшим пульсациям АЧХ. ВП DFD Remez Design линейно интерполирует значения весовых коэффициентов по частоте между разными точками. Например, чтобы проектировать ФНЧ, чьи пульсации АЧХ в полосе пропускания составляют половину от уровня пульсаций в полосе задержания, устанавливают весовой коэффициент в полосе пропускания, равный 2, а в полосе задержания – 1.

На рис. 16.69 показаны параметры в полосе частот для ФНЧ. Полоса пропускания – в диапазоне от 0 до 0,2 и полоса задержания – в диапазоне частот от 0,3 до 0,5. Частота дискретизации – 1.

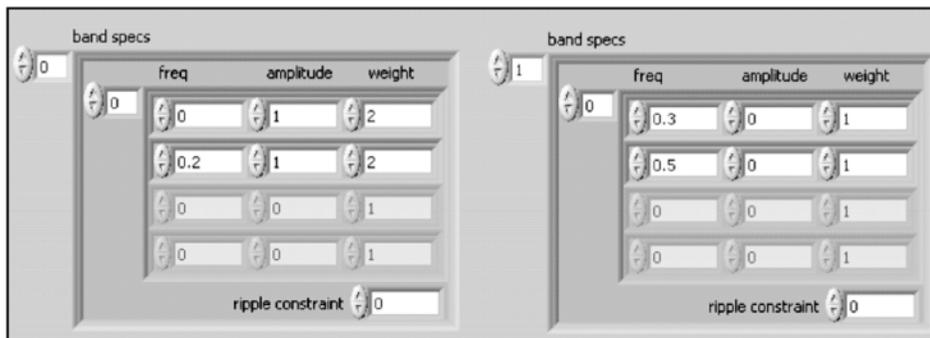


Рис. 16.69. Параметры полосы КИХ-фильтра ФНЧ

Необходимо заметить, что полоса частот от 0,2 до 0,3 не определена. Эту полосу называют полосой перехода. Необходимо также заметить, что значения весового коэффициента (**weight**) в полосе пропускания вдвое больше, чем в полосе задержания, что обеспечивает пульсации АЧХ в полосе пропускания в два раза меньше пульсаций в полосе задержания.

**Примечание.** Передаточная функция, которая описывается вводом параметров полосы (**band specs**), есть АЧХ со знаком и можно обеспечить отрицательные заданные значения величины (**amplitude**). Однако если устанавливается на вводе типа фильтра (**filter type**) – **Minimum Phase** или **Maximum Phase**, то частотная характеристика, описываемая при вводе параметров полосы (**band specs**), есть АЧХ, и все заданные значения **amplitude** должны быть положительными.

### 16.7.4. Определение порядка фильтра

Можно установить на вводе **minimum order** и **order**, чтобы определить порядок фильтра. Число коэффициентов фильтра равняется порядку КИХ-фильтра плюс один.

Если устанавливается **minimum order** при вводе **user defined**, то ВП DFD Remez Design определяет порядок фильтра, используя значение ввода **order**. Например, если устанавливается ввод **filter order** на 20 и устанавливается **minimum order** при вводе **user defined**, то окончательный фильтр будет иметь 21 коэффициент.

Если устанавливается при вводе **minimum order** на **minEven** или **minOdd**, то ВП DFD Remez Design игнорирует введенный **order** и определяет минимальный требуемый порядок фильтра при запуске ВП. Если устанавливается при вводе **minimum order** в **minEven** или **minOdd**, необходимо определить ограничение на пульсации АЧХ для всех полос. Для получения информации об установке ограничения на пульсации необходимо обратиться к пункту *Ограничение пульсаций АЧХ* этого подраздела.

### 16.7.5. Выбор типа фильтра

Можно определить тип фильтра, который необходимо проектировать, выбирая один из следующих вариантов ввода типа фильтра (**filter type**):

- симметричный (Symmetric);
- несимметричный (Antisymmetric);
- дифференциатор (Differentiator);
- Гильберта (Hilbert);
- минимально-фазовый (Minimum Phase);
- максимально-фазовый (Maximum Phase).

### 16.7.6. Симметричные и несимметричные типы фильтра

Симметричные и несимметричные варианты определяют симметрию импульсной характеристики фильтра и, следовательно, определяют симметрию его передаточной функции относительно частоты составляющей с нулевой фазой, указанной в параметрах полосы. Для получения информации об установке ввода **filter**

**type** для проектирования одного из четырех типов КИХ-фильтров с линейной ФЧХ можно обратиться к табл. 16.8.

### 16.7.7. Дифференциатор и типы фильтра Гильберта

Дифференциатор и фильтр Гильберта имеют несимметричные импульсные передаточные функции. Опция **Differentiator** использует встроенную весовую функцию, которая обратно пропорциональна к частоте, для достижения постоянного уровня пульсаций, выраженного в процентах относительно АЧХ. Опция **Hilbert** фактически та же самая, как и опция **Antisymmetric**. Для получения информации о дифференциаторах и преобразователях Гильберта можно обратиться к подразд. 16.5.

### 16.7.8. Минимально-фазовый и максимально-фазовый типы фильтра

Фильтры минимально-фазовые иногда называют фильтрами с минимальной задержкой энергии, или минимальной задержки. КИХ-фильтр минимальной фазы имеет все нули  $z$ -преобразования передаточной функции внутри или на круге с единичным радиусом на  $z$ -плоскости. КИХ-фильтр максимально-фазовый имеет обратную по времени импульсную передаточную функцию относительно характеристики минимально-фазового фильтра. При этом нули максимально-фазового фильтра находятся вне или на круге единичного радиуса на  $z$ -плоскости.

### 16.7.9. Аппроксимация АЧХ КИХ-фильтра

В подразд. 16.5 рассмотрено проектирование КИХ-фильтров с линейной ФЧХ, которые являются фактически фильтрами с аппроксимированными АЧХ, что обеспечивается следующим сложным критерием аппроксимации:

$$\min \left( \max \left( \sum_{i=0}^{L-1} (W(i) \cdot |H(\omega_i) - D(\omega_i)|) \right) \right), \quad (16.17)$$

где  $D(\omega_i)$  – идеальная передаточная функция;  $H(\omega_i)$  – передаточная функция разработанного фильтра;  $W(i)$  – положительный вес в  $i$ -й точке частоты.

Многие приложения требуют, чтобы была гарантирована линейная ФЧХ, при которой спектральные составляющие входного сигнала проходят через фильтр с одинаковой задержкой.

Если имеется приложение, которое не требует линейной ФЧХ, можно изменять АЧХ фильтра и позволять процессу фильтрации произвольно изменять относительную задержку между различными спектральными составляющими. Можно использовать следующее выражение для этого типа проблемы аппроксимации АЧХ:

$$\min \left( \max \left( \sum_{i=0}^{L-1} (W(i) \cdot \|H(\omega_i) - D(\omega_i)\|) \right) \right). \quad (16.18)$$

Поскольку ограничение на ФЧХ устранено, можно достигнуть той же самой величины ошибки аппроксимации с пониженным порядком фильтра или меньшей величиной ошибки аппроксимации с тем же самым порядком фильтра, который имеет меньшую стоимость.

Несколько наборов коэффициентов фильтра могут иметь ту же самую передаточную функцию. В  $z$ -области можно создать новый набор коэффициентов с той же самой передаточной функцией, если все нули не находятся на круге единичного радиуса. Обеспечивается такая операция через всепропускающее зеркальное отражение нулей к их сопряженным значениям по взаимному расположению относительно круга единичного радиуса. Можно определить понятия минимально-фазовый и максимально-фазовый фильтры, чтобы устранить двусмысленность, по которой используется набор коэффициентов КИХ-фильтра.

### 16.7.10. Проектирование минимально-фазовых и максимально-фазовых КИХ-фильтров

Все нули в передаточной функции минимально-фазового КИХ-фильтра находятся внутри или на круге единичного радиуса на плоскости  $z$ .

Минимально-фазового фильтры иногда называют фильтрами минимальной задержки энергии, потому что энергия импульсной передаточной функции максимально сконцентрирована к началу импульсной передаточной функции.

Все нули в передаточной функции максимально-фазового КИХ-фильтра расположены снаружи или на круге единичного модуля на  $z$ -плоскости.

Для максимально-фазового КИХ-фильтра энергия импульсной передаточной функции максимально сконцентрирована к концу этой функции. С учетом заданной АЧХ импульсные передаточные функции минимально-фазовых и максимально-фазовых КИХ-фильтров обратны во времени.

На рис. 16.70 показана АЧХ КИХ-фильтра 16-го порядка.

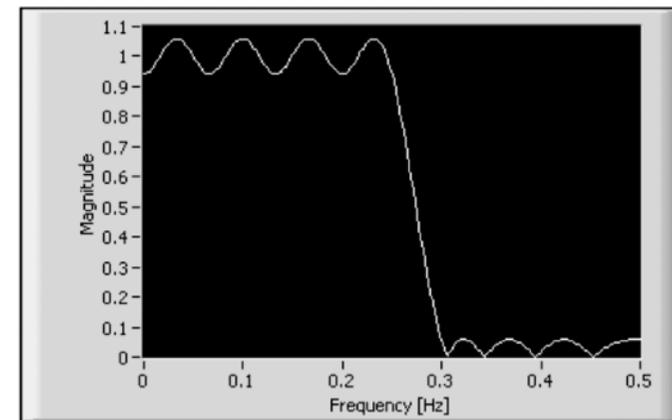


Рис. 16.70. АЧХ КИХ-фильтра 16-го порядка минимально/максимально-фазового

На рис. 16.71 и 16.72 показано, как два различных набора коэффициентов фильтра обеспечивают АЧХ КИХ-фильтра.

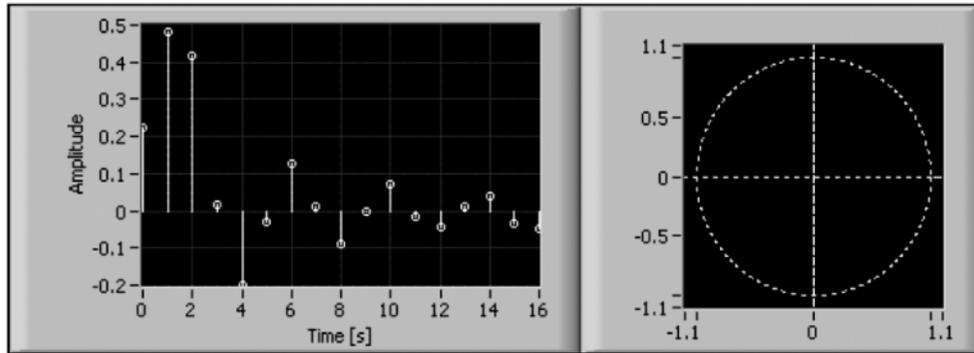


Рис. 16.71. Импульсная передаточная функция минимально-фазового фильтра и расположение нулей

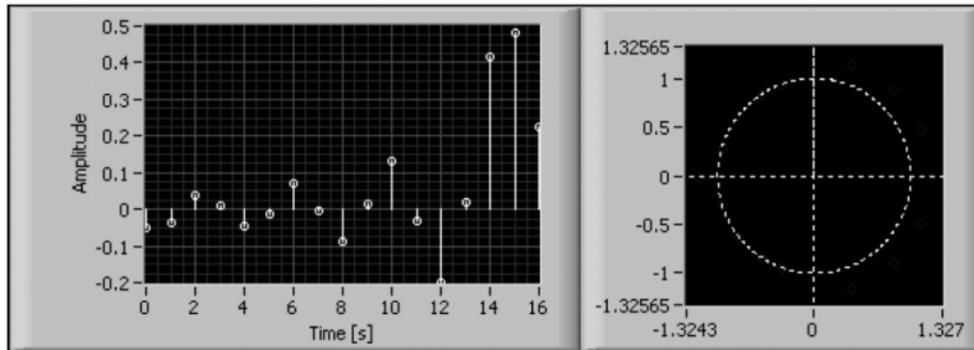


Рис. 16.72. Импульсная передаточная функция максимально-фазового фильтра и поле расположения нулей

На рис. 16.71 представлена импульсная передаточная функция и расположение нулей для минимально-фазового фильтра с показанной на рис. 16.70 АЧХ.

На рис. 16.72 представлена импульсная передаточная функция и расположение нулей для максимально-фазового фильтра, имеющего АЧХ, показанную на рис. 16.70.

Минимально-фазовые фильтры особенно полезны в приложениях управления. Процесс фильтрации в петле управления обычно требует отклика на входное воздействие, чтобы реакция была настолько быстрой, насколько это возможно. Большая задержка процесса фильтрации может привести к отрицательной обратной связи в петле управления и к нестабильности в системе управления.

КИХ-фильтры с линейной ФЧХ имеют один недостаток, заключающийся в том, что системная задержка, также называемая групповой задержкой, определяется половиной порядка фильтра и может быть неприемлемо длинна, когда фильтр имеет большой порядок. Можно использовать минимально-фазовый фильтр в ситуациях, когда уменьшение задержки является критическим, если система не требует фильтр с линейной ФЧХ или порядок такого фильтра является относительно малым.

### 16.7.11. Проектирование минимально-фазовых и максимально-фазовых фильтров в ВП DFD Remez Design

Можно проектировать минимально-фазовые или максимально-фазовые КИХ-фильтры, используя ВП DFD Remez Design. Для этого необходимо установить ввод **filter type** на **Minimum Phase** или **Maximum Phase** и оставшиеся параметры такими же, как при проектировании фильтра с линейной ФЧХ. Для получения информации о проектировании фильтров с линейной ФЧХ можно обратиться к подразд. 16.5. Для получения дополнительной информации о проектировании минимально-фазовых и максимально-фазовых фильтров с точной установкой коэффициента передачи или ограничениями пульсаций АЧХ можно обратиться к пунктам: *Точная установка коэффициента передачи* и *Ограничение пульсаций АЧХ* настоящего подраздела.

Можно создать минимально-фазовый фильтр, разрабатывая фильтр с линейной ФЧХ и преобразовывая его в минимально-фазовый фильтр, зеркально отражая нули, которые расположены вне круга единичного радиуса, к их сопряженной взаимной позиции в этом круге. Хотя новый минимально-фазовый фильтр обладает той же самой АЧХ, как и исходный фильтр с линейной ФЧХ, новый фильтр не оптимален относительно параметров АЧХ фильтра. Можно обеспечить более близкое соответствие параметрам АЧХ или более низкий порядок фильтра, создавая минимально-фазовый фильтр непосредственно.

**Примечание.** Минимально-фазовые фильтры могут обеспечить лучшие результаты в величине более низкой пульсации АЧХ или понизить порядок относительно заданных параметров АЧХ, чем эквивалентные фильтры с линейной ФЧХ. Поэтому выбор в пользу минимально-фазового фильтра может быть лучше, чем в линейной ФЧХ в приложениях, где ФЧХ не контролируется.

### 16.7.12. Полосовой фильтр единственной точки по частоте

Полосовой фильтр точечный по частоте есть фильтр, у которого полоса состоит из единственной точки.

Полосовые фильтры точечные по частоте полезны, когда проектируются фильтры-пробки или пиковые фильтры. Можно разработать полосовой фильтр точечный по частоте на ВП DFD Remez Design, вводя только одну точку в строке **band specs** при описании АЧХ.

Например, необходимо проектировать фильтр-пробку с частотами подавления 0,15 и 0,35 Гц. Пять полос в этом примере имеют диапазоны частот: [0, 0,12], [0,15], [0,18, 0,32], [0,35] и [0,38, 0,5].

Все полосы пропускания, полоса около единственной точки по частоте и полосы задержания имеют одинаковый весовой коэффициент.

На рис. 16.73 приведена АЧХ разработанного фильтра 30-го порядка.

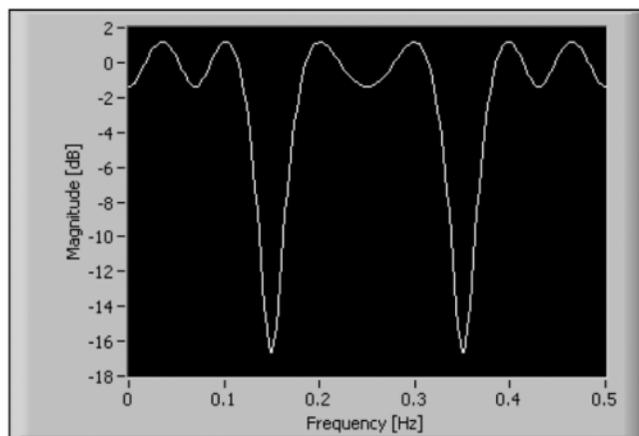


Рис. 16.73. АЧХ фильтра-пробки 30-го порядка на две частоты

Можно найти другие использования для параметров точечного по частоте фильтра. Например, добавляя полосу единственной точки в полосу перехода, можно вынудить АЧХ проходить близко к заданной точке. Альтернативно, когда приходится сталкиваться с аномалиями в полосе перехода при разработке многополосного фильтра, можно найти способы удаления аномалий, добавляя полосы в АЧХ точечного по частоте фильтра в пределах его полос перехода.

### 16.7.13. Точная установка значений передаточной функции

В этом методе вводятся значения передаточной функции и частоты в полосе в строку ввода **band specs** для ВП DFD Remez Design. При этом для создания проекта фильтра используются комплексные точные или приближенные величины. Можно использовать точную установку передаточной функции, чтобы обеспечить заданные значения АЧХ фильтра в определенных значениях по частоте.

Чтобы использовать точную установку передаточной функции в ВП DFD Remez Design, надо ввести в строку **freqs of exact gain [Hz]** список частот, которые связаны с идеальными значениями амплитуд. Можно также установить те же частоты в строке **band specs** в ВП DFD Remez Design, где вводятся соответствующие им амплитуды. Если же эти частоты не вводятся в строку **band specs** в ВП DFD Remez Design, то эти значения линейно интерполируются.

Рассмотрим АЧХ фильтра, представленную на рис. 16.73. Можно обеспечить более острые вырезки в АЧХ, применяя точную установку коэффициента передачи в частотах 0,15 и 0,35 Гц. Введение для этих точек частоты точечной полосы в строку **freqs of exact gain [Hz]** перепроектирует фильтр-пробку в фильтр, имеющий АЧХ, показанную на рис. 16.74.

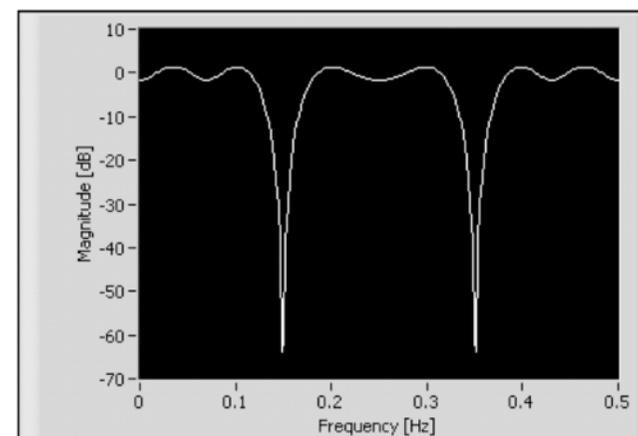


Рис. 16.74. АЧХ фильтра-пробки с точной установкой значений передаточной функции

Необходимо обратить внимание, что график имеет более острые вырезки в АЧХ по сравнению с вырезками, показанными на рис. 16.73.

На рис. 16.75 показана АЧХ ФНЧ 12-го порядка с одинаковым уровнем пульсаций АЧХ без задания точных значений передаточной функции по частоте. Диапазон полосы пропускания – [0, 0,25] и диапазон полосы задержания – [0,3, 0,5] имеют равный весовой коэффициент в обоих полосах. Необходимо обратить внимание, что коэффициент передачи для постоянной составляющей не равен единице.

Чтобы приравнять коэффициент передачи к единице для постоянной составляющей, надо ввести 0 в строку **freqs of exact gain [Hz]** и перепроектировать ФНЧ. Окончательная АЧХ фильтра приведена на рис. 16.76.

Также следует обратить внимание, что коэффициент передачи для постоянной составляющей становится точно равным единице без значительного увеличения уровня пульсаций АЧХ.

На рис. 16.77 представлена АЧХ ФВЧ 12-го порядка с полосой задержания в диапазоне частот [0, 0,2] и диапазоном частот полосы пропускания [0,3, 0,5].

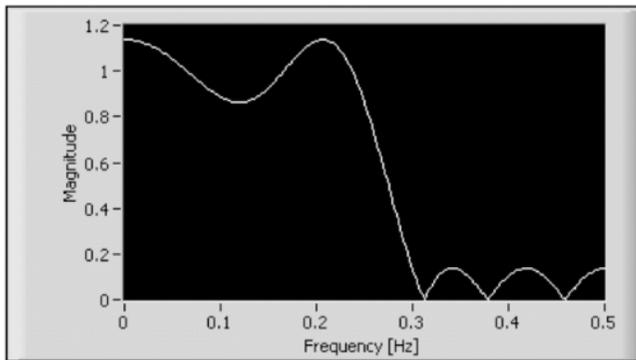


Рис. 16.75. АЧХ ФНЧ 12-го порядка  
(без точных значений передаточной функции)

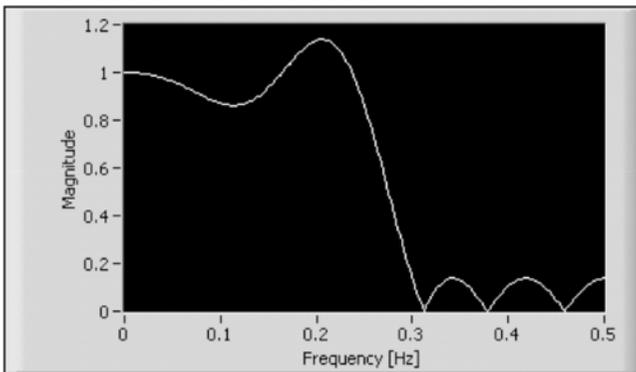


Рис. 16.76. АЧХ ФНЧ 12-го порядка  
с точными значениями передаточной функции

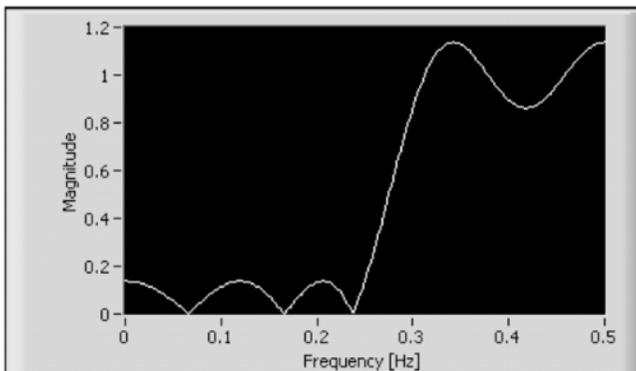


Рис. 16.77. АЧХ ФВЧ 12-ого порядка  
(нет точных значений коэффициента передачи)

Например, предположим, что сигнал относительно чист, если бы не помеховая составляющая с частотой 0,1 Гц и ее вторая гармоника с частотой 0,2 Гц. Чтобы уменьшить мощность помехи на этих частотах, можно ввести в строку **freqs of exact gain [Hz]** значения 0,1 и 0,2.

На рис. 16.78 показана АЧХ фильтра с точными значениями передаточной функции, указанными на частотах 0,1 и 0,2 Гц. Необходимо обратить внимание, что значения АЧХ на частотах 0,1 и 0,2 Гц точно равны нулю.

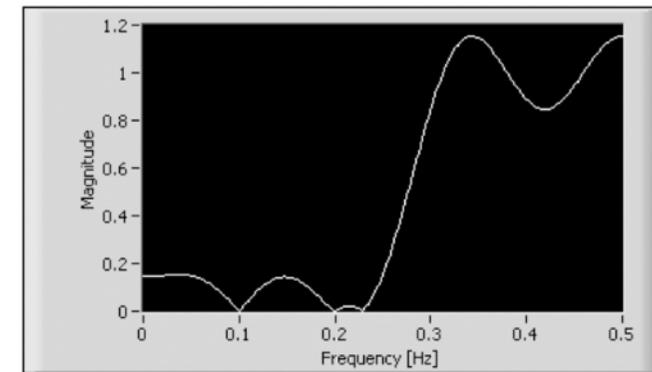


Рис. 16.78. АЧХ ФВЧ 12-го порядка  
с точными значениями передаточной функции

### 16.7.14. Ограничение пульсаций АЧХ

Ограничение пульсаций АЧХ гарантирует, что максимальная величина ошибки между АЧХ разработанного фильтра и заданной АЧХ равна или ниже заданной. Можно определить ограничение пульсаций для заданной полосы, вводя положительное значение при вводе **ripple constraint** для этой полосы. Когда определяется ограничение пульсаций, в ВП DFD Remez Design используется это значение как верхняя граница уровня пульсаций в пределах заданной полосы.

**Примечание.** Ограничение пульсаций АЧХ в ВП DFD Remez Design выполняется по-другому алгоритму, чем взвешивание пульсаций в ВП Parks-McClellan, который основан на классическом методе дизайна Ремеза. Классический метод дизайна Ремеза допускает уровни пульсаций АЧХ в различных полосах пропорционально. Кроме того, классический алгоритм гарантирует только то, что разработанный фильтр имеет те же самые относительные значения пульсаций между различными полосами, а не фактически введенные значения.

Можно проектировать фильтры с ограничениями пульсаций АЧХ в следующих ситуациях:

- определить порядок фильтра и ограничить уровень пульсаций в некоторых полосах;
- ограничить уровень пульсаций АЧХ во всех полосах, тогда можно применить ВП DFD Remez Design для определения минимального порядка фильтра, удовлетворяющего заданным требованиям;
- ограничить уровень пульсаций АЧХ во всех полосах и также определить порядок фильтра. ВП DFD Remez Design сообщает об ошибке, если ограничения не могут быть выполнены. В противном случае, ВП DFD Remez Design выполняет правильный проект с относительными пульсациями между различными полосами такими же, как ограничения на уровень относительных пульсаций.

Как пример для первой ситуации, предположим, что необходимо спроектировать ФНЧ 15-го порядка, и необходимо также, чтобы уровень пульсаций АЧХ в его полосе пропускания был не больше, чем 0,1. Можно установить диапазон частоты полосы пропускания в пределах  $[0, 0,2]$  и диапазон полосы задержания –  $[0,3, 0,5]$  и применить равный весовой множитель в обеих полосах. На рис. 16.79 приведена АЧХ разработанного фильтра.

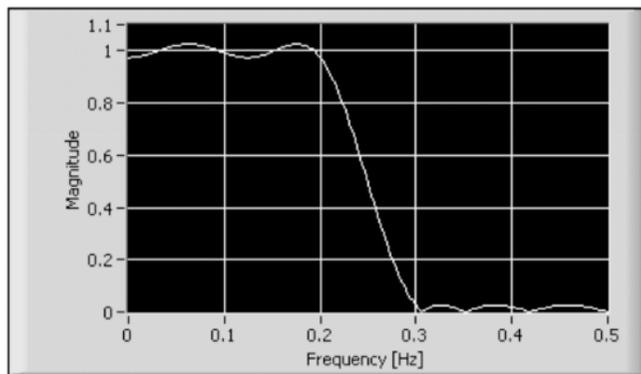


Рис. 16.79. АЧХ ФНЧ 15-го порядка с ограничением пульсаций в полосе задержания (весовой множитель в полосе пропускания равен 1)

Необходимо обратить внимание, что пульсации в полосе пропускания намного меньше, чем заданные. Чтобы сделать ограничения уровня пульсаций в полосе пропускания относительно верхнего значения, можно уменьшить начальный весовой множитель (**weight**) в полосе пропускания до значений  $[0,01, 0,01]$  и перепроектировать фильтр.

На рис. 16.80 показана АЧХ перепроектированного фильтра.

Меньшее значение весового множителя (**weight**) позволяет ВП DFD Remez Design проектировать фильтр с пульсациями в полосе пропускания, которые первоначально превышают ограничение, и затем автоматически скорректировать значение весового множителя итерационно, пока ограничение не будет выполнено.

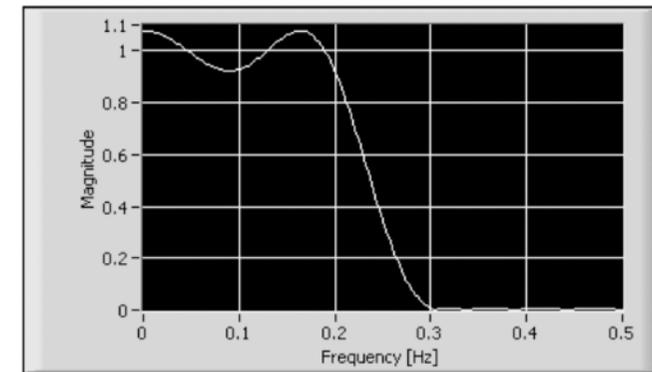


Рис. 16.80. АЧХ ФНЧ 15-го порядка с ограничением пульсаций в полосе задержания (весовой множитель в полосе пропускания равен 0,01)

Как пример для второй ситуации, предположим, что необходимо спроектировать ФНЧ с той же самой полосой пропускания и полосой задержания, как и в предыдущем примере. Можно определить ограничения пульсаций АЧХ для полосы пропускания и полосы задержания как 0,01 и 0,001 соответственно, и установить ввод **minimum order** в положение **minEven**. На рис. 16.81 показана АЧХ КИХ-фильтра 28-го порядка с равномерными пульсациями, разработанного в ВП DFD Remez Design.

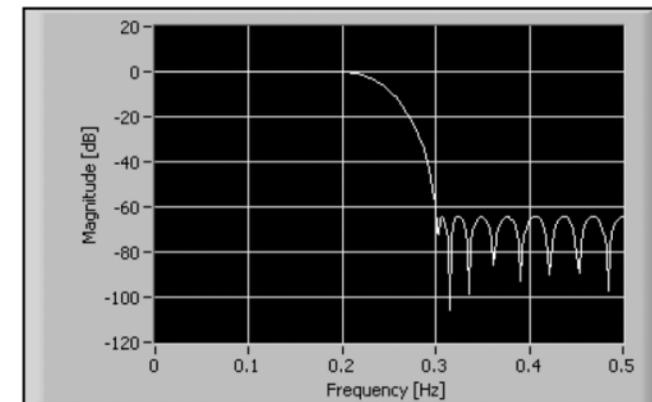


Рис. 16.81. АЧХ КИХ-фильтра минимального 28-ого порядка с ограничениями пульсаций во всех полосах

### 16.7.15. Метод Least Pth Norm Design

Можно использовать ВП DFD Least Pth Norm Design для проектирования следующих БИХ- и КИХ-фильтров:

- проект КИХ-фильтра с линейной ФЧХ;
- проект с аппроксимацией АЧХ и ФЧХ КИХ- и БИХ-фильтров;
- проект с аппроксимацией только АЧХ для КИХ- и БИХ-фильтров, хотя можно проектировать фильтры с линейной ФЧХ, используя ВП DFD Remez Design. Можно также проектировать КИХ- и БИХ-фильтры с произвольной АЧХ и ограничениями ФЧХ, используя ВП DFD Least Pth Norm Design.

Классические методы проектирования типа Баттерворта (Butterworth) и эллиптические методы определяют фильтр как оптимальный, когда фильтр обеспечивает выполнение заданных критериев. При помощи ВП DFD Least Pth Norm Design можно проектировать БИХ-фильтры, которые не основаны на классическом БИХ-оптимальном критерии, но вместо этого основаны на следующих критериях.

Передаточная характеристика фильтра БИХ с  $N$  нулями и  $M$  полюсами представлена уравнением:

$$H(\omega) = \frac{B(\omega)}{A(\omega)} = \frac{\sum_{n=0}^N b(n)e^{-j\omega n}}{1 + \sum_{n=1}^M a(n)e^{-j\omega n}}. \quad (16.19)$$

Когда  $M$  равняется нулю, БИХ-фильтр редуцируется до КИХ-фильтра.

Учитывая комплексный характер идеальной передаточной функции  $D(\omega)$ , ВП DFD Least Pth Norm Design проектирует оптимальный БИХ-фильтр, определенный в смысле наименьшей  $p$ -й нормы. ВП использует для создания проекта или комплексную аппроксимацию или аппроксимацию по амплитуде.

Комплексная аппроксимация представлена уравнением

$$\|E\|_p = \left( \sum_{i=0}^{L-1} (W(i)|H(\omega_i) - D(\omega_i)|)^p \right)^{\frac{1}{p}}. \quad (16.20)$$

Аппроксимация по амплитуде представлена уравнением

$$\|E\|_p = \left( \sum_{i=0}^{L-1} (W(i)||H(\omega_i)| - |D(\omega_i)||)^p \right)^{\frac{1}{p}}, \quad (16.21)$$

где  $W(i)$  – положительный вес в  $i$ -й точке частоты;  $H$  – передаточная функция разработанного фильтра;  $D$  – заданная передаточная функция;  $L$  – число точек частоты, в которых выполняют вычисление;  $p$  – норма.

Уравнения (16.20) и (16.21) определены в величинах коэффициентов фильтра  $a[]$  и  $b[]$ .

Если устанавливается ввод **filter type** на **Minimum Phase** или **Maximum Phase**, то ВП DFD Least Pth Norm Design выполняет аппроксимацию по амплитуде и иг-

норирует информацию об аргументе передаточной функции  $D(\omega)$ . Если устанавливается ввод **filter type** на **Symmetric**, **Antisymmetric**, **Differentiator** или **Hilbert**, то ВП DFD Least Pth Norm Design Нормы использует комплексную аппроксимацию.

ВП DFD Least Pth Norm Design предлагает две реализации:

- метод итерационного перевзвешенного наименьшего квадрата (IRLS);
- метод Ньютона (Newton).

Первый алгоритм обычно работает лучше, чем второй в любом специальном приложении. Можно проверить оба метода и выбрать метод, который обеспечивает лучшие результаты.

### 16.7.16. Использование ВП DFD Least Pth Norm Design

Чтобы проектировать цифровой фильтр на основе использования ВП DFD Least Pth Norm Design, необходимо ввести параметры для заданной передаточной функции, порядок фильтра, порядок нормы и ограничение радиуса (модуля) полюса. Ниже приведены специальные соображения, как ввести параметры фильтра. Для получения окончательной информации о ВП DFD Least Pth Norm Design можно обратиться к *LabVIEW Help*.

### 16.7.17. Определение заданной передаточной функции

Можно использовать строку **band specs**, чтобы определить заданную передаточную функцию фильтра. Каждый элемент в строке представляет параметры одной полосы частот. Можно ввести одну или более точек в возрастающем порядке, чтобы описать передаточную функцию в каждой полосе. ВП DFD Least Pth Norm Design подключает заданные точки, чтобы сформировать непрерывную идеальную передаточную функцию для полосы. Диапазон частот между двумя последовательными полосами есть полоса перехода.

ФЧХ фильтра определяется  $\theta_{overall}(\omega)$  **phase** в **band specs**  $\theta_{specified}(\omega)$  и **group delay**  $\tau_{gp}$ , как показано в следующем уравнении:

$$\theta_{overall}(\omega) = -\tau_{gp}\omega + \theta_{specified}(\omega). \quad (16.22)$$

### 16.7.18. Определение порядка фильтра

Можно определить порядок фильтра, устанавливая ввод **order**, включая цифры **num** порядка и порядка **den** знаменателя. Нужно установить **den** равным нулю при проектировании КИХ-фильтра.

### 16.7.19. Определение порядка нормы

Можно определить порядок нормы, устанавливая ввод **p**, который соответствует  $p$  в уравнениях (16.20) и (16.21). Значение  $p$  должно быть между 2 и 128. Если  $p$  равно 2, то ВП DFD Least Pth Norm Design выполняет решение по методу наи-

меньших квадратов. В случае, если  $p$  растет, решение асимптотически приближается к решению для равномерного уровня пульсаций АЧХ. Когда  $p = 128$ , то ВП DFD Least Pth Norm Design выполняет решение, неотличимое от решения для равномерной величины пульсаций АЧХ.

### 16.7.20. Определение ограничения радиуса полюса

Можно определить максимальный допустимый радиус (модуль) полюса, устанавливая ввод **pole radius constraint**. Маленький радиус полюса снижает возможность неустойчивости фильтра, вызванной влиянием конечной точности. Однако малое значение ограничения радиуса полюса может неблагоприятно изменить потенциальную резкость передаточной функции.

### 16.7.21. Проектирование КИХ-фильтров с минимальной величиной $p$ -й нормы и линейной ФЧХ

Можно проектировать КИХ-фильтры с линейной ФЧХ, используя ВП DFD Least Pth Norm Design и устанавливая следующие параметры. Установить порядок знаменателя равным 0, **filter type** к **Symmetric** или **Antisymmetric**, все фазы в параметрах полосы приравнять к 0 и групповую задержку – к половине порядка числителя.

Например, предположим, что необходимо спроектировать ФНЧ КИХ-фильтр с линейной ФЧХ с диапазоном частот полосы пропускания  $[0, 0,2]$  и диапазоном полосы задержания  $[0,3, 0,5]$ . Установить параметры, как показано на рис. 16.82.

На рис. 16.83 показана АЧХ разработанного фильтра. Поскольку  $p = 128$ , разработанный фильтр почти идентичен результату, полученному при использова-

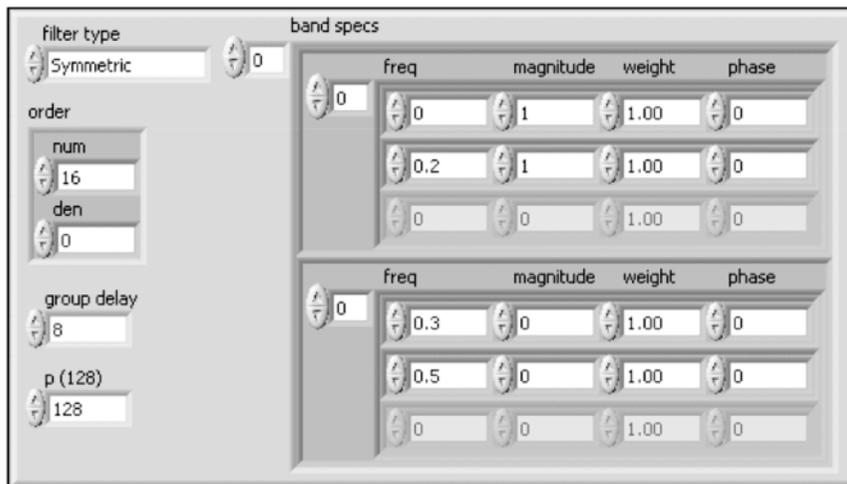


Рис. 16.82. Параметры КИХ-фильтра ФНЧ с линейной ФЧХ

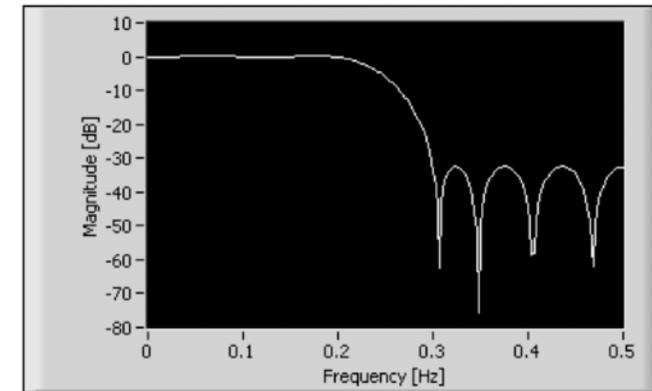


Рис. 16.83. АЧХ КИХ-фильтра ФНЧ с линейной ФЧХ

нии проекта равномерных пульсаций АЧХ по методу Ремеза с теми же самыми параметрами фильтра.

### 16.7.22. Проектирование БИХ-фильтров с приблизительно линейной ФЧХ

Можно проектировать БИХ-фильтры с приблизительно линейной ФЧХ, используя ВП DFD Least Pth Norm Design. Необходимо установить **filter type** в **Symmetric** или **Antisymmetric**.

Хотя невозможно проектировать каузальные цифровые БИХ-фильтры с теоретически точной линейной ФЧХ, можно проектировать БИХ-фильтры с приблизительно линейной ФЧХ. Например, предположим, что необходимо спроектировать ФНЧ БИХ-фильтр с приблизительно линейной ФЧХ с диапазоном частот полосы пропускания  $[0, 0,2]$  и полосой задержания с диапазоном частот  $[0,3, 0,5]$ . Можно установить параметры, как показано на рис. 16.84.

На рис. 16.85 представлена АЧХ разработанного фильтра.

На рис. 16.86 приведена ФЧХ разработанного фильтра.

Необходимо обратить внимание, что этот фильтр имеет большее ослабление в полосе задержания, чем КИХ-фильтр с линейной ФЧХ с представленной на рис. 16.83 АЧХ, этот фильтр сохраняет ФЧХ в полосе пропускания груболинейной.

### 16.7.23. Проектирование минимально-фазового и максимально-фазового БИХ-фильтра

Можно использовать опции **Minimum Phase** или **Maximum Phase** при вводе **filter type**, если надо иметь минимально-фазовый или максимально-фазовый фильтр или если ФЧХ не важна. В случае, если используются опции **Minimum Phase** или

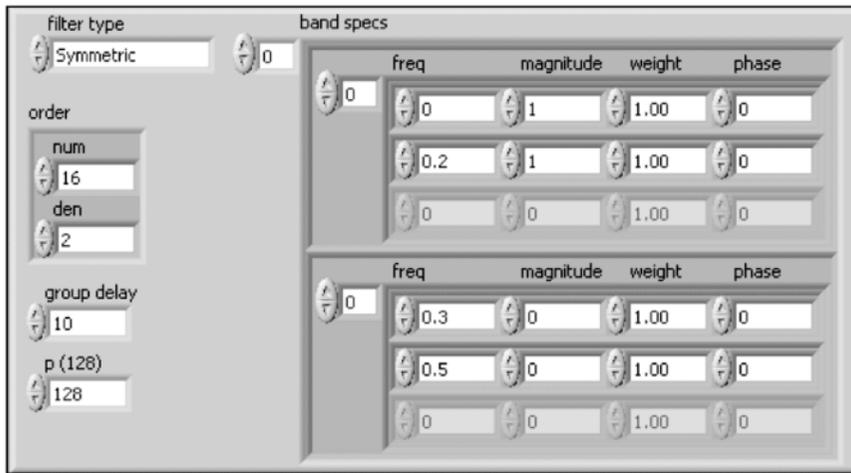


Рис. 16.84. Параметры ФНЧ КИХ-фильтра с приблизительно линейной ФЧХ

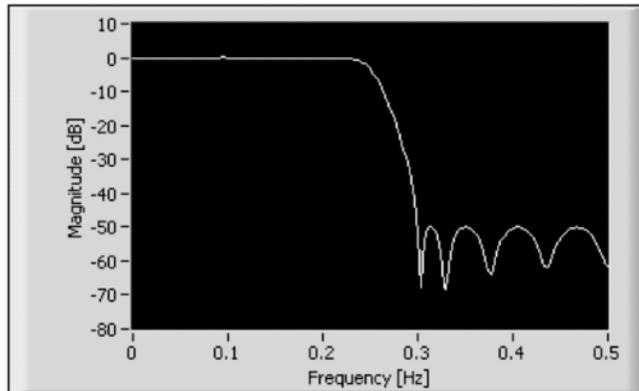


Рис. 16.85. АЧХ КИХ-фильтра ФНЧ с приблизительно линейной ФЧХ

**Maximum Phase**, то ВП DFD Least Pth Norm Design игнорирует **phase** (установку ФЧХ) и вводит **group delay** (групповая задержка).

Например, предположим, что необходимо проектировать минимально-фазовый ФНЧ БИХ-фильтр с диапазоном частот полосы пропускания  $[0, 0,2]$  и полосой задержания в диапазоне частот  $[0,3, 0,5]$ . Можно установить параметры, как показано на рис. 16.87.

На рис. 16.88 представлена АЧХ разработанного фильтра.

На рис. 16.89 приведена ФЧХ разработанного фильтра.

Необходимо обратить внимание, что разработанный фильтр имеет большее ослабление в полосе задержания, чем БИХ-фильтр с приблизительно линейной

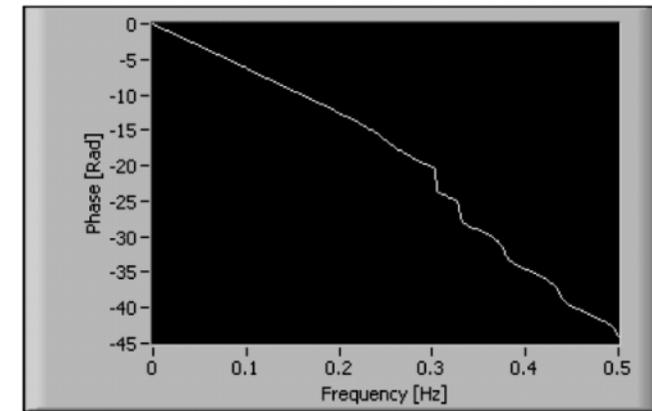


Рис. 16.86. ФЧХ КИХ-фильтра ФНЧ с приблизительно линейной ФЧХ

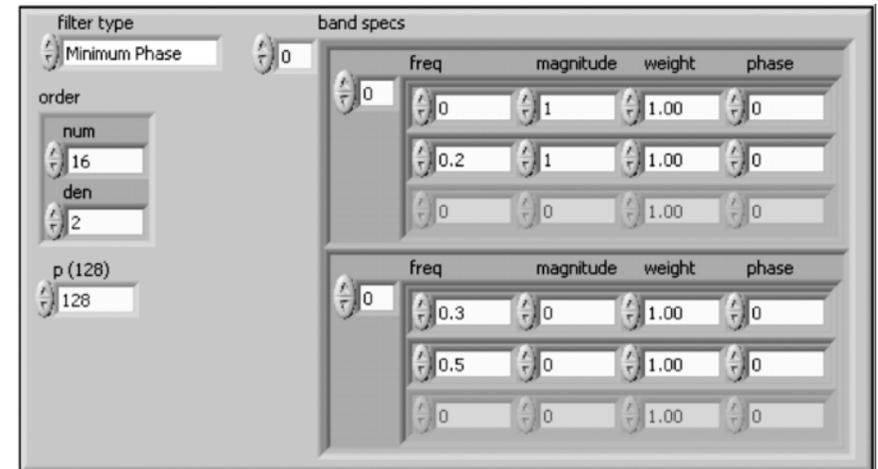


Рис. 16.87. Параметры минимально-фазового БИХ-фильтра ФНЧ

ФЧХ, представленный АЧХ на рис. 16.85, но ФЧХ в полосе пропускания теперь более нелинейная.

## Резюме

1. Можно использовать ВП DFD Remez Design, чтобы создать следующие фильтры:
  - проект КИХ-фильтра с линейной ФЧХ (типы I–IV);
  - проект дифференциатора;

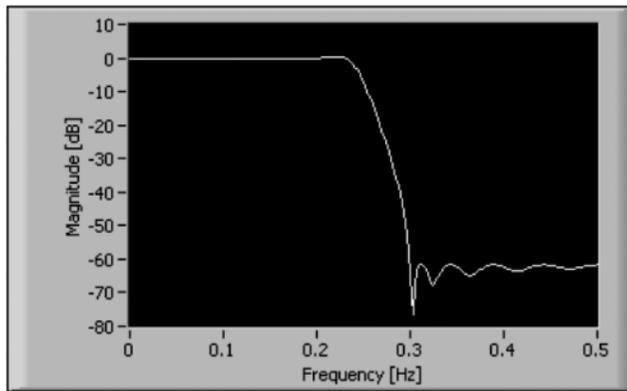


Рис. 16.88. АЧХ минимально-фазового БИХ-фильтра ФНЧ

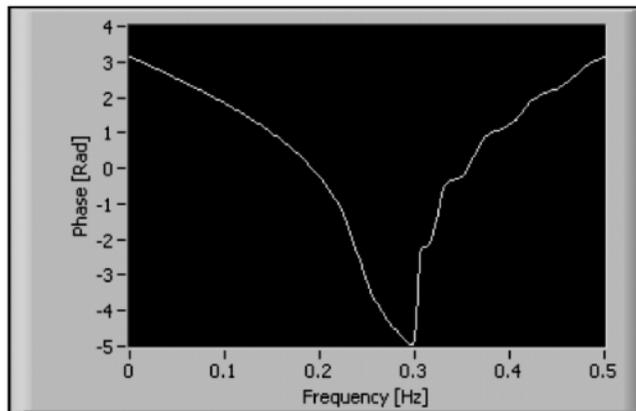


Рис. 16.89. ФЧХ минимально-фазового БИХ-фильтра ФНЧ

- проект преобразователя Гильберта;
  - проект КИХ-фильтра с произвольной формой АЧХ;
  - проект оптимального фильтра с аппроксимацией ФЧХ (минимально-фазовый или максимально-фазовый);
  - параметры точечного по полосе фильтра (пробка или пик);
  - точное управление коэффициента передачи;
  - параметры с ограничением пульсаций;
2. Использование ВП DFD Least Pth Norm Design для создания следующих фильтров:
- проект КИХ-фильтра с линейной ФЧХ;
  - проект КИХ- и БИХ-фильтров с аппроксимированными АЧХ и ФЧХ;
  - проект КИХ- и БИХ-фильтров с аппроксимированной только АЧХ.

## 16.8. Проектирование КИХ-фильтра

### 16.8.1. Описание задачи

**Исходные данные.** Проект КИХ-фильтра ФНЧ 23-го порядка с линейной ФЧХ и с АЧХ, близкой к идеальной, с полосой пропускания в 0,3Гц. Этот проект основан на следующих окнах: прямоугольное (Rectangular); треугольное (Triangular); Хэннинга (Hanning); Хэмминга (Hamming); Блэкмена (Blackman).

**Необходимо определить.** График импульсной характеристики, АЧХ, расположение нулей и полюсов этих фильтров. Произвести сравнение видов АЧХ.

Смотреть также: проект проф. Стана Ривес в Университете Аубурга [36].

**Решение.** При решении сформулированной задачи используются методы взвешивания при проектировании КИХ-фильтров. Можно загрузить программу ВП FIR Windowed Filter Design, записанную в LabVIEW 7.1, которая обеспечивает решение этой задачи. Необходимо открыть ВП и запустить его, чтобы увидеть импульсную характеристику, АЧХ и поле расположения нулей и полюсов. Изменив тип окна на любой другой из 11 возможных типов окон, можно рассмотреть связь этих изменений с характеристиками фильтров.

На рис. 16.90 и 16.91 показаны характеристики фильтров при выборе типа окна.

### 16.8.2. Реализация фильтров в LabVIEW

1. Открыть незаполненный ВП в LabVIEW. Нажать правой кнопкой мыши блок-диаграмму, чтобы открыть палитру функций (**Functions**). Нажать кнопку **All Functions** на этой палитре и затем определить положение палитры инструментария проектирования цифровых фильтров.

**Примечание.** Палитры LabVIEW, показанные на рис. 16.92 – 16.96, скомпонованы на дисплее в виде икон и текста. По умолчанию, дисплеи в LabVIEW показывают в палитрах только иконы. Можно изменить компоновку палитры на дисплее как в виде икон, так и в виде текста, выбрав Tools»Options в LabVIEW, затем выбрать Controls/Functions Palettes, нажимая наверху в меню **Options** диалогового окна и выбирая Иконы и Текст (**Icons and Text**) в списке **Формат (Format)**. Можно обратиться к Getting Started в руководстве к LabVIEW для более глубокого изучения и получить больше информации о работе в LabVIEW.

2. При необходимости проектирования КИХ-фильтра можно открыть палитру Filter Design и затем палитру Advanced FIR Filter Design, как показано на рис. 16.94–16.96.
3. Если необходимо, исследовать понятия, связанные с проектированием методом взвешивания (окна) КИХ-фильтра, выбрать ВП DFD Windowed FIR Design из палитры Advanced FIR Filter Design, как показано на рис. 16.96, и поместить его в блок-диаграмму.

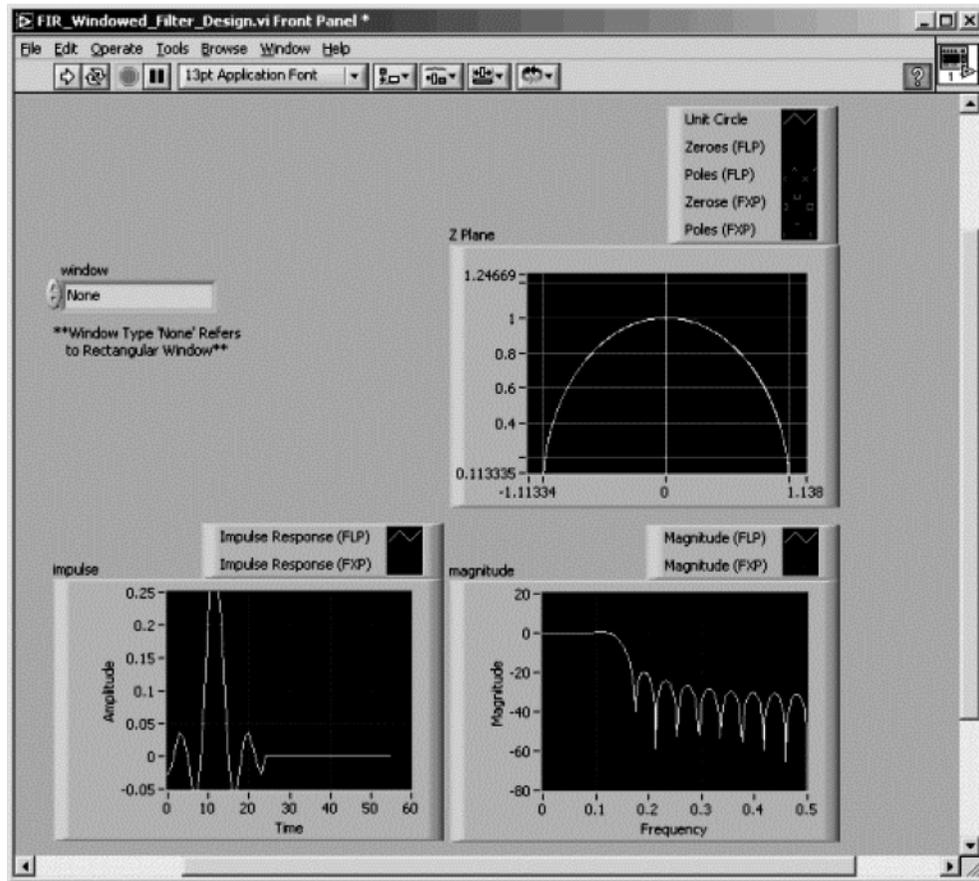


Рис. 16.90. Прямоугольное окно  
(практически не является окном при использовании в проекте фильтра)

4. Проверить ВП DFD Windowed FIR Design на панно соединений, как показано на рис. 16.97, для изучения входов и выходов ВП.
5. Установить органы управления по входам ВП DFD Windowed FIR Design. После выполнения этих установок блок-схема будет напоминать блок-диаграмму, показанную на рис. 16.98.

Нажать правой кнопкой мыши входы фильтра и выбрать Create»Constant из ярлыков меню, чтобы создать константы. Установить константу – ФНЧ. Если необходимо создать фильтр 23-го порядка, связать константу порядок с выходом и установить на ней число 23. Для простоты решения установить частоту дискретизации  $f_s$  равную, например, 1 Гц. Верхняя граничная частота  $f_h$  устанавливается только для полосовых (band-pass) фильтров, следовательно, ее можно игнорировать в этом примере.

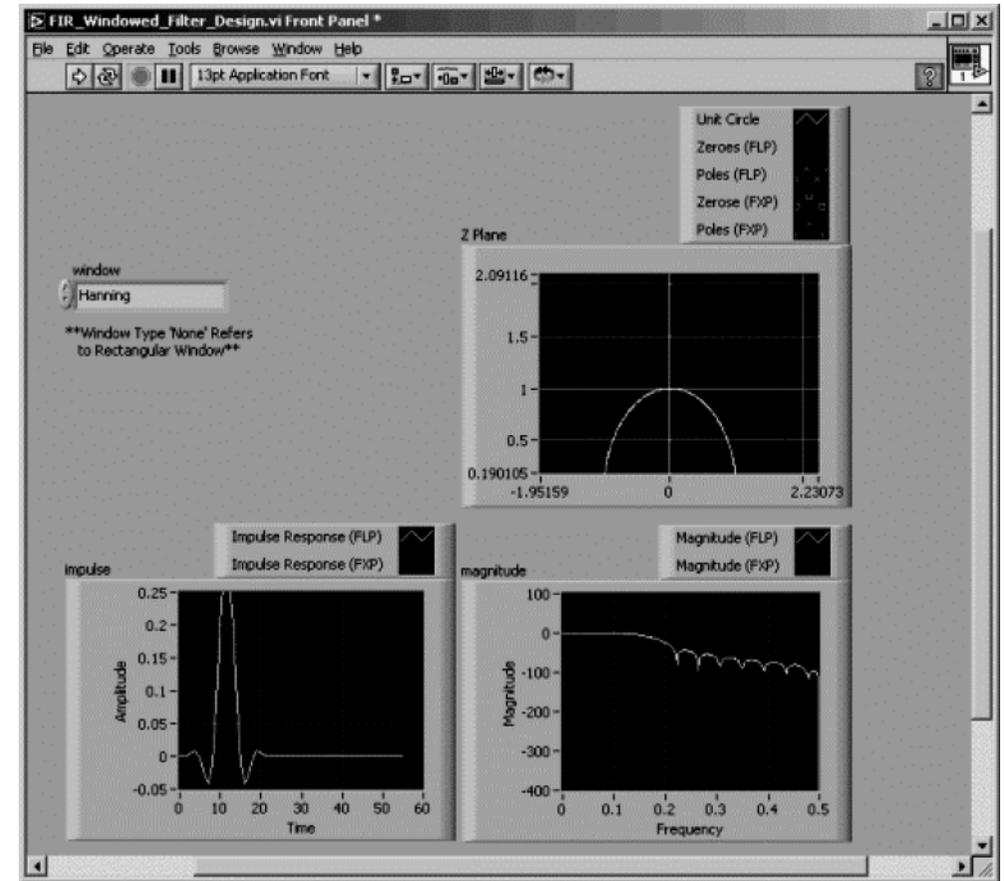


Рис. 16.91. Окно Хэннинга

Связать константу нижней частоты  $f_l$  среза с входом и установить число 0,15. Относительная частота среза будет равна  $0,3p$ .  $P$  определяется по формуле  $p = f_s / 2$ . Следовательно, частота среза есть  $0,3p = (0,3f_s)/2 = 0,15$ . Щелкнуть правой кнопкой мыши на входе Window и выбрать из меню ярлыков Create»Control. Можно проектировать различные взвешенные КИХ-фильтры в ВП DFD Windowed FIR Design. Вход Window определяет тип используемого окна. На рис. 16.99 приведен список окон, которые могут быть установлены в этом ВП. Можно найти эту информацию в LabVIEW Help, выбрав кнопку Help ВП, затем Function и & How-To Help для выбора в индексе DFD Windowed FIR Design.

6. Теперь, после того как фильтр спроектирован, необходимо найти ВП DFD Filter Analysis Express, расположенный в палитре Filter Analysis, как показано на рис. 16.100, и установить его на блок-схеме созданного фильтра. ВП Express является интерактивным ВП на диалоговом окне параметров

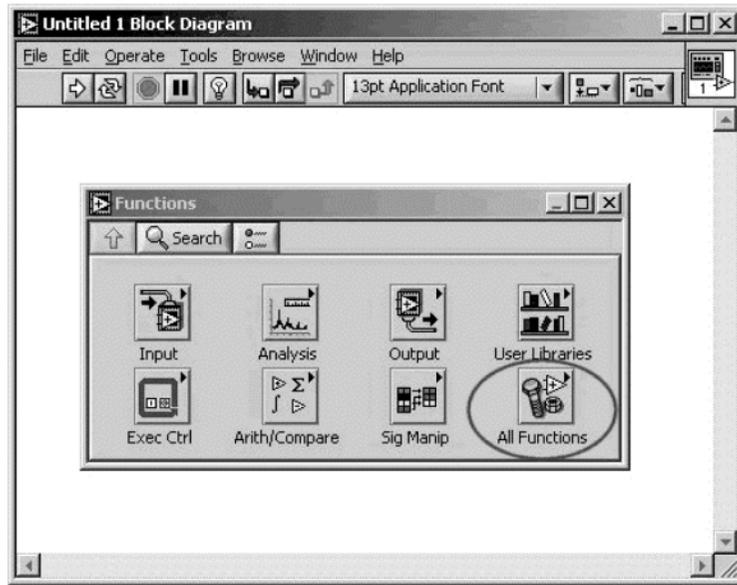


Рис. 16.92. Палитра Функции (Functions)



Рис. 16.93. Икона (Digital Filter Design Toolkit Palette)

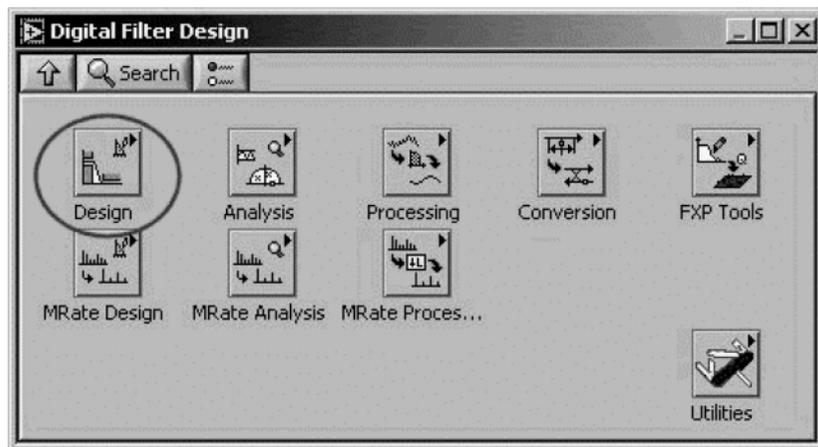


Рис. 16.94. Палитра Digital Filter Design

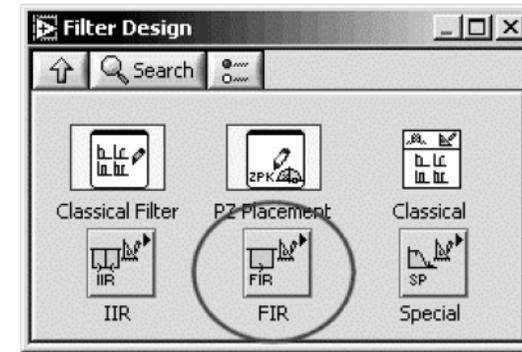


Рис. 16.95. Палитра Filter Design

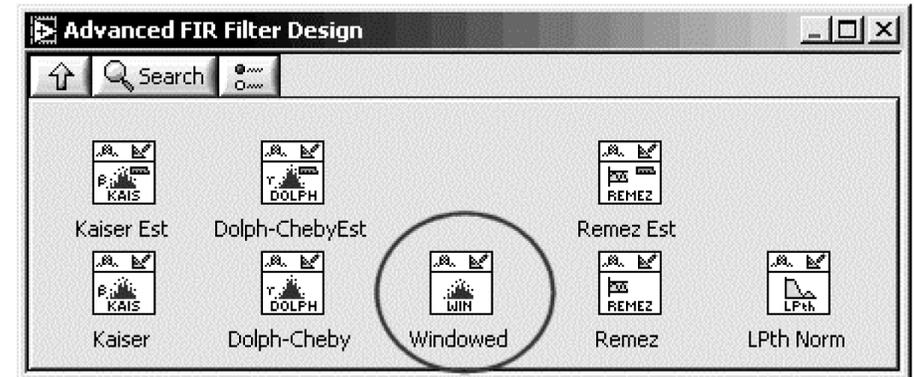


Рис. 16.96. Палитра Advanced FIR Filter Design

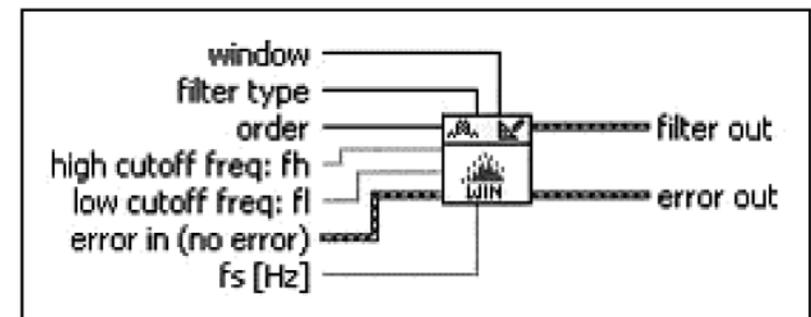


Рис. 16.97. Панно соединений ВП DFD Windowed FIR Design

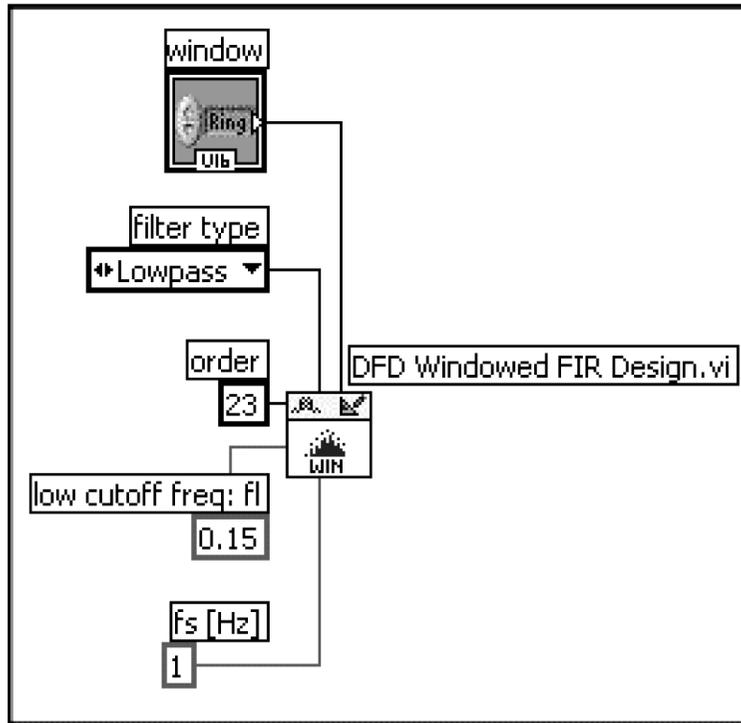


Рис. 16.98. Входы ВП Inputs to the DFD Windowed FIR Design в проекте фильтра

**U16** **window** specifies the time-domain window the VI uses.

0	None (default)
1	Hanning
2	Hamming
3	Blackman-Harris
4	Exact Blackman
5	Blackman
6	Flat Top
7	4 Term B-Harris
8	7 Term B-Harris
9	Low Sidelobe
30	Triangular

Рис. 16.99. Опции Window в ВП DFD Windowed FIR Design

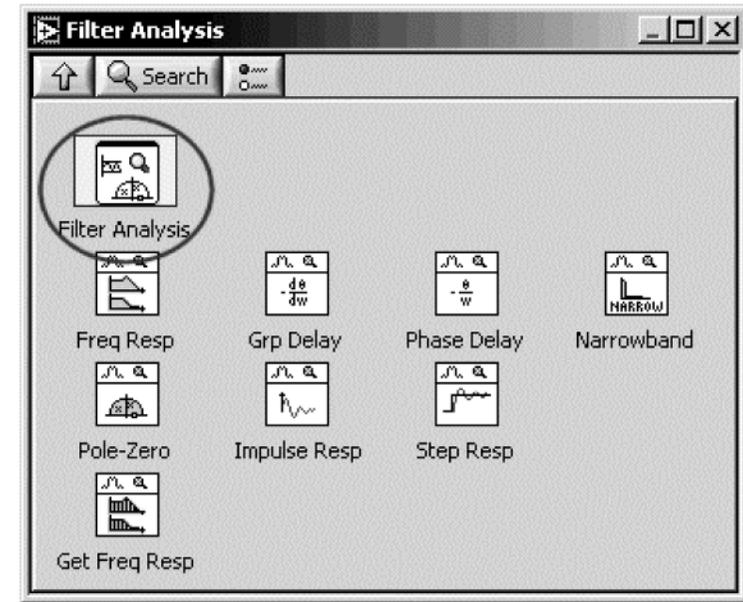


Рис. 16.100. Палитра Filter Analysis

для сравнения со стандартным ВП, состоящего из модулей ВП, которые соединены по входным установкам к ВП.

7. Когда на блок-диаграмме размещен ВП Express, появляется диалоговое окно параметров. Если необходимо анализировать только импульсную характеристику, АЧХ и положение нулей и полюсов, следует поместить метки при надписях: Show step response, Show phase response, and Show group delay checkboxes, как показано на рис. 16.101.
8. Наметить, какие опции необходимо установить на осях графика – линейный масштаб или в децибелах (dB). Выбрать dB для этой задачи, так как этот масштаб может лучше показать изменения величин в большом диапазоне амплитуд. Нажать кнопку ОК на диалоговом окне параметров.
9. Подсоединить выходы ВП DFD Windowed FIR Design к входам фильтра ВП DFD Filter Analysis Express.
10. Нажать правой кнопкой мыши на **magnitude**, **Z Plane** и **impulse** на выходах ВП DFD Filter Analysis Express и выбрать **Create»Graph Indicator** или **Create»Indicator** из ярлыков меню для каждого графика.

Окончательная блок-диаграмма показана для напоминания на рис. 16.102.

**Различия в представлении результатов решения.** Можно показать характеристики фильтра на многих окнах на отдельных графиках или на фронтальной панели для перекрестного сравнения, как показано на рис. 16.103–16.105. Можно обратиться к примеру, основанному на эксперименте с ВП, результаты которого приведены на перечисленных рисунках.

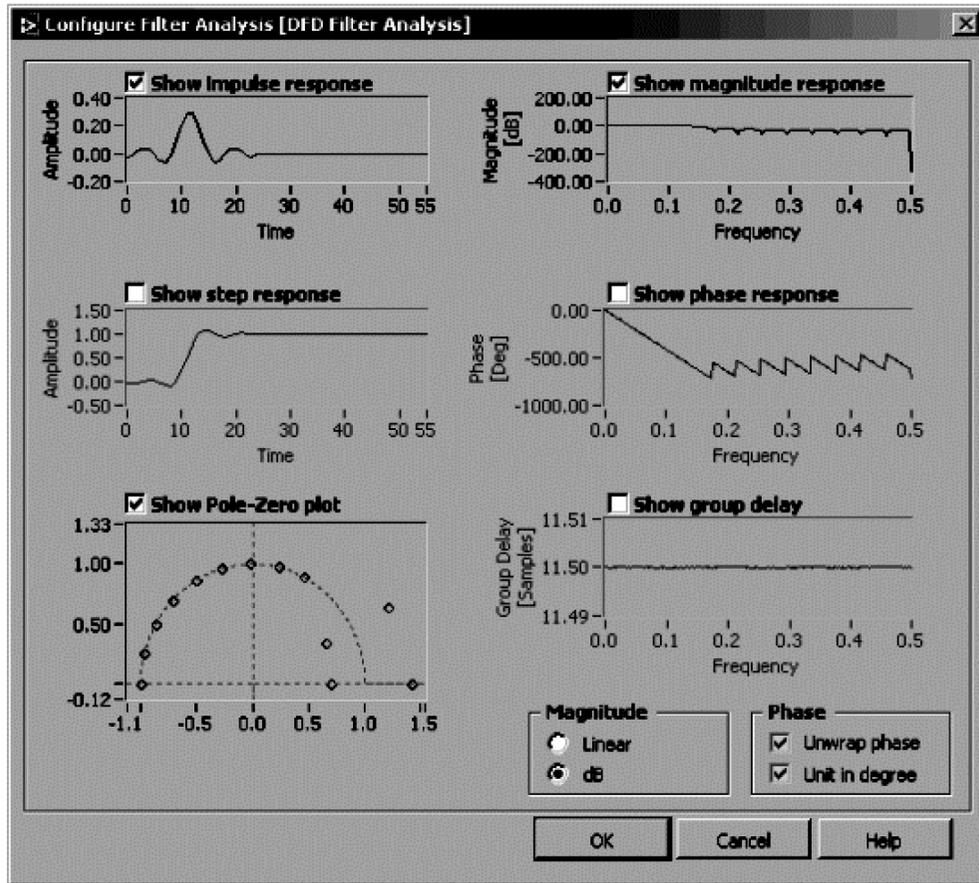


Рис. 16.101. Диалоговое окно Configuration для ВП DFD Filter Analysis Express

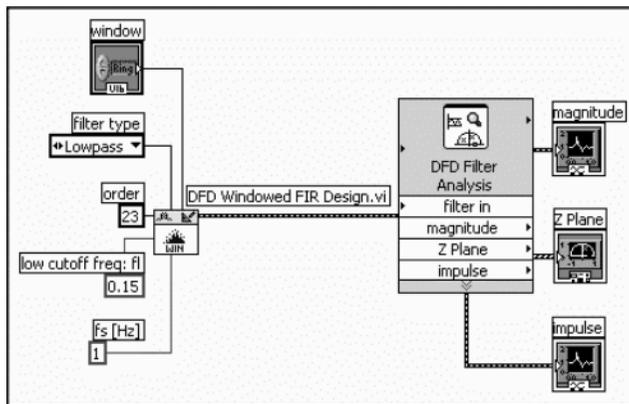


Рис. 16.102. Окончательная блок-диаграмма для проектирования и анализа фильтра

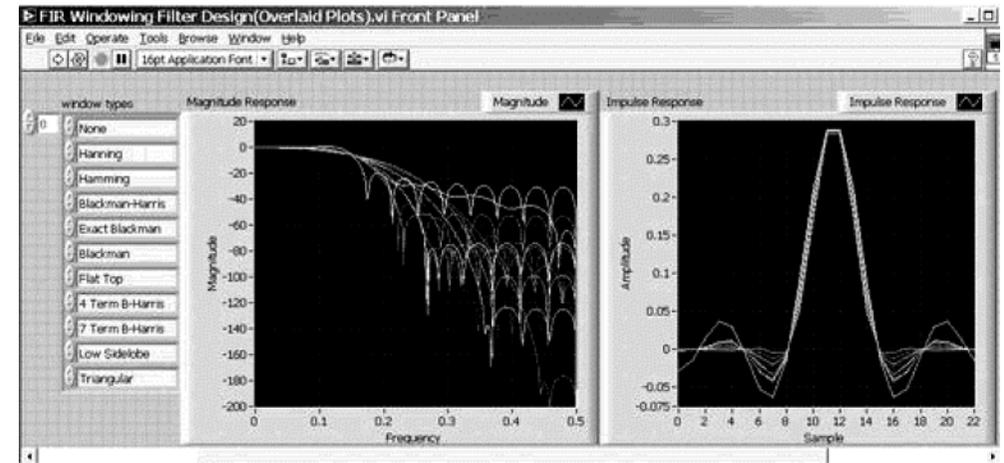


Рис. 16.103. Характеристики фильтра на лицевых графиках

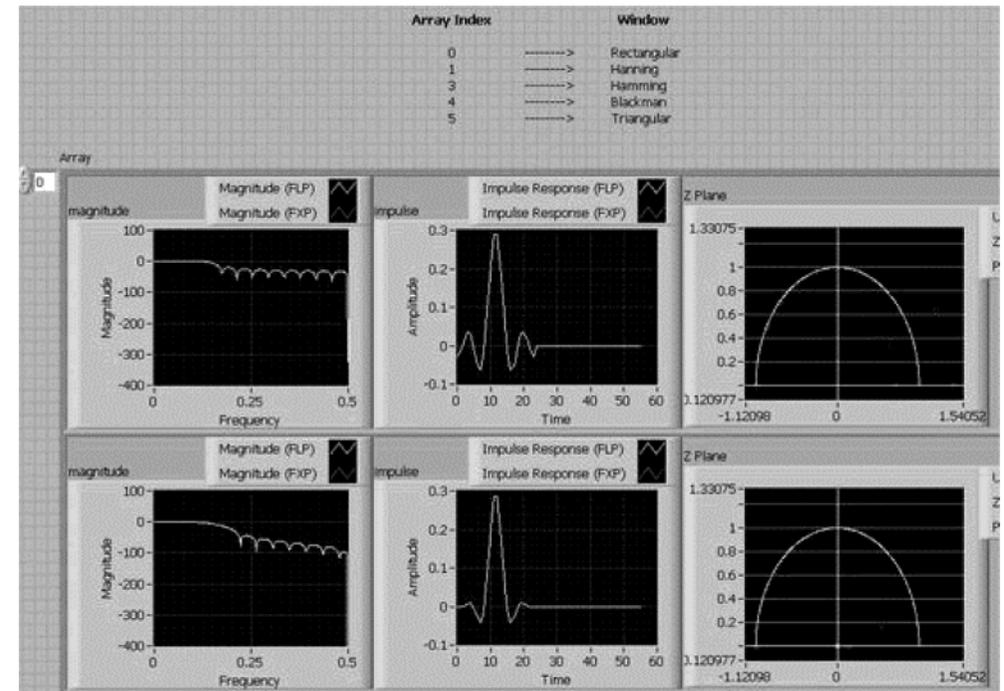


Рис. 16.104. Характеристики фильтра на графиках одиночных массивов

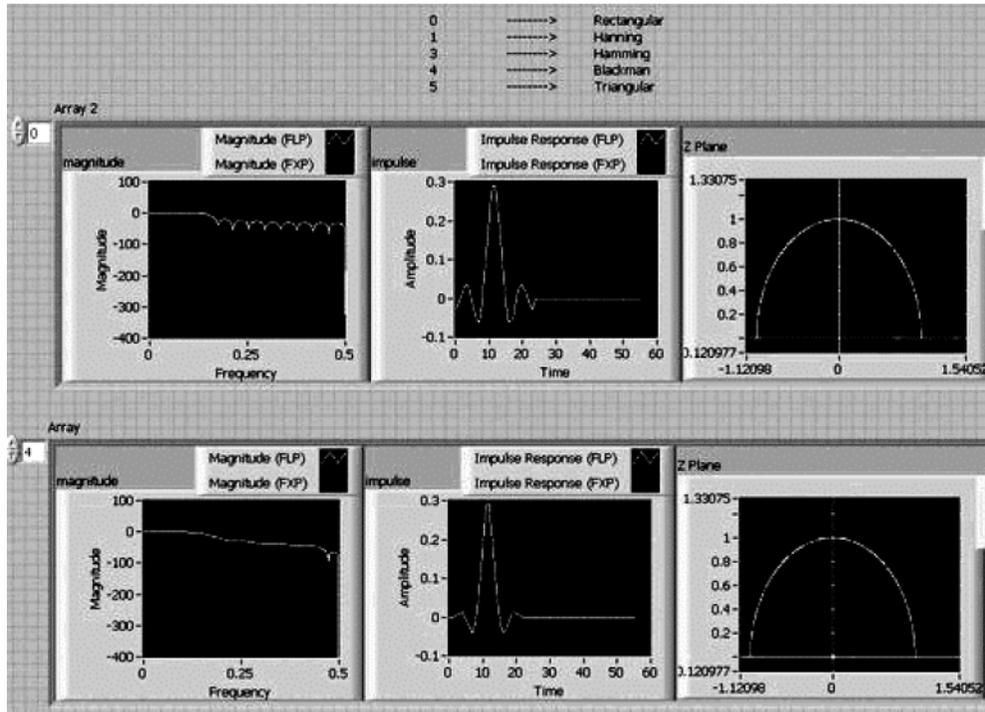
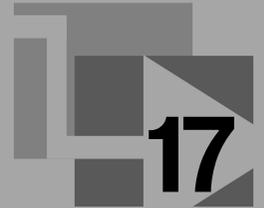


Рис. 16.105. Графики на дисплее для второго массива

Можно изменить размеры массива для показа характеристик фильтра в нескольких последовательных типах окон несколько раз.

Такой показ дает возможность произвольно сравнить характеристики фильтра для двух типов окон.

## Машинное зрение в LabVIEW



Модуль для разработки систем машинного зрения предназначен для инженеров и ученых, решающих задачи обработки как статических, так и динамических сигналов для различных приложений. В его состав входит программный продукт **NI Vision Assistant** – интерактивный помощник для быстрой разработки прототипа без необходимости программирования, а также библиотека **IMAQ Vision**, обладающая широким набором функций для обработки изображений. Оба продукта могут работать совместно для упрощения процесса проектирования системы. Приложение Vision Assistant может автоматически генерировать блок-диаграмму LabVIEW, которую можно использовать для обработки изображений как отдельно, так и в составе сложной тестирующей системы, включающей управление движением, сбор данных или другие задачи.

### 17.1. Интерактивный помощник NI Vision Assistant

Интерактивный помощник NI Vision Assistant предназначен для построения систем машинного зрения, решающих задачи измерений, коррекции, идентификации и детального анализа изображений. Поскольку работа с данной программой не требует навыков программирования, исследователь может сосредоточить внимание на разработке алгоритмов анализа изображений. После определения функциональности можно сгенерировать блок-диаграмму в LabVIEW или создать файл, включающий в себя определенный набор функций для LabWindows/CVI, Visual Basic, C и Visual C++. При использовании указанных возможностей время перехода от проекта к законченному рабочему решению многократно сокращается. Необходимо отметить также, что для разработки измерительной системы в NI Vision Assistant имеется более 200 предустановленных функций.

Сгенерированная программой блок-диаграмма включает в себя функции сбора данных, анализа и отображения в соответствии с определенной пользователем задачей. Полученный код является самодостаточным, т. е. его можно использовать как в отдельном приборе, так и в составе сложной тестовой системы. Кроме того, его можно изменять, добавляя функциональность.

Файл, созданный помощником для LabWindows/CVI, C и Visual Basic содержит детальное пошаговое описание процедуры получения изображения, его обработки, анализа и отображения. Текст можно извлечь для использования в составе проекта C или Visual Basic.

## 17.2. Использование библиотеки IMAQ Vision в задачах машинного зрения

Библиотека IMAQ Vision – это часть модуля для разработки систем машинного зрения (Vision Development Module), представляющая собой набор ВП LabVIEW. Она также включает в себя функции для LabWindows™/CVI™ и других сред разработки на языке C, и управляющие элементы ActiveX для Visual Basic. На рис. 17.1 представлено диалоговое окно пользователя интерактивного помощника при обработке изображения.

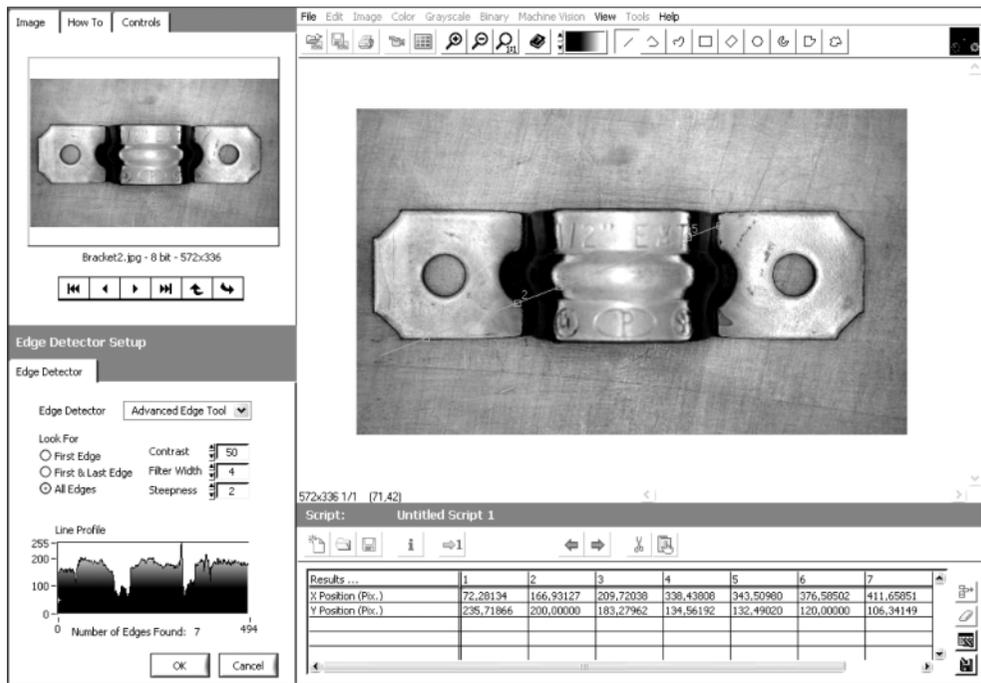


Рис. 17.1. Диалоговое окно пользователя интерактивного помощника

Рис. 17.2. Последовательность создания приложений IMAQ Vision

На рис. 17.2 и 17.3 показаны последовательности создания приложений с помощью IMAQ Vision в LabVIEW.

### 17.2.1. Последовательность создания приложений IMAQ Vision

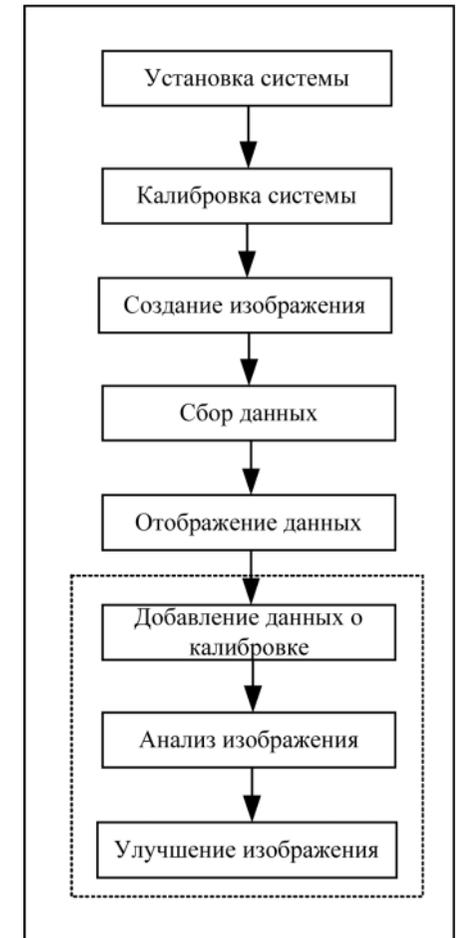
**Установка системы.** До начала сбора, анализа и обработки изображения необходимо установить необходимое оборудование и программное обеспечение. Выбор и настройка оборудования зависят от характеристик среды проведения измерений, а также от методов анализа. Система обработки изображений должна получать данные с максимально возможным качеством, что позволит извлечь необходимую информацию без потерь.

**Калибровка системы.** После установки системы можно провести ее калибровку, для того чтобы назначить реальные координаты каждой отображаемой точке и компенсировать ошибки, свойственные данной системе. Ошибки возникают, если ось камеры не перпендикулярна цели, а также из-за искажений в линзах камеры.

Указанные ошибки искажают получаемое изображение, но при этом информация в нем не обязательно теряется. ВП, необходимые для проведения калибровки, расположены в палитре Vision Utilities ⇒ Calibration (Сервисные функции ⇒ Калибровка).

**Создание изображения.** Для того чтобы создать изображение, необходимо использовать ВП IMAQ Create VI, расположенный в палитре Vision Utilities ⇒ Image Management (Сервисные функции ⇒ Управление изображением). После создания изображения необходимо выбрать тип данных:

- Grayscale (U8, default) – черно-белый, 8-битовый, беззнаковый;
- Grayscale (I16) – черно-белый, 16-битовый, знаковый;
- Grayscale (SGL) – черно-белый с плавающей точкой;
- Complex (CSG) – комплексный;
- RGB (U32) – 32-битовый, RGB;



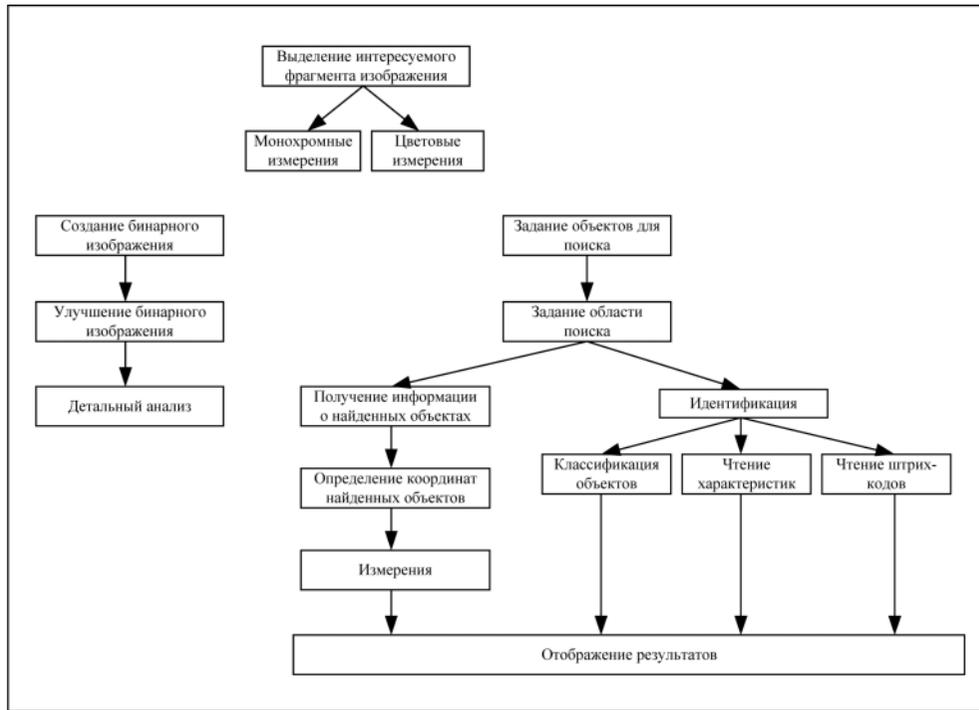


Рис. 17.3. Шаги по обработке изображений

- HSL (U32)—32-битовый, HSL;
- RGB (U64)—64-битовый, RGB.

Пользователь может создавать неограниченное число изображений в одном коде программы, но каждому из них должно быть присвоено собственное имя.

**Сбор данных.** После создания изображения сбор данных можно осуществлять тремя способами. Это может быть получение изображения с камеры, из файла или конвертация данных из двумерного массива. Для чтения данных указанными способами предназначены следующие ВП:

- Однократный сбор данных осуществляется с помощью ВП IMAQ Snap VI (захватить фрагмент). После вызова данной функции, инициализируется работа источника изображения и считывается следующий видеофрагмент. Данный ВП рекомендуется применять в задачах, требующих простоты программного кода.
- Для непрерывного сбора данных необходимо применять захват видеоизображения. В данном случае чтение выполняется в цикле, и данные пишутся в буфер. ВП IMAQ Grab Setup VI, Grab Acquire VI и IMAQ Stop VI (настройка, сбор, остановка) предназначены для установки параметров захвата, сбора данных и завершения выполнения кода соответственно. Использование данных функций позволяет выполнять чтение видеоизображения с высокой скоростью.

- Для чтения фиксированного числа фрагментов изображения предназначен ВП IMAQ Sequence VI (последовательность). В данном случае происходит сбор данных до тех пор, пока количество считанных фрагментов не станет равным числу, запрошенному оператором. При этом возможен сбор изображений с пропуском заданного числа фрагментов между ними.

После завершения сбора данных обязательно применять ВП IMAQ Close VI (закрыть сессию).

Для чтения и обработки изображений из файла применяются ВП, расположенные в палитре Vision Utilities ⇒ Files (Сервисные функции ⇒ Файлы), позволяющие обрабатывать такие форматы, как BMP, TIFF, JPEG, PNG и AIPD (ВП IMAQ ReadFile VI, IMAQ Read Image VI, IMAQ Vision Info VI, IMAQ GetFileInfo VI – Считать файл, Считать информацию о файле, Извлечь информацию о файле). С их помощью также можно считывать видеоизображение в формате AVI (IMAQ AVI Open, IMAQ AVI Read Frame, IMAQ AVI Close – Открыть, Считать и Закрыть AVI-файл).

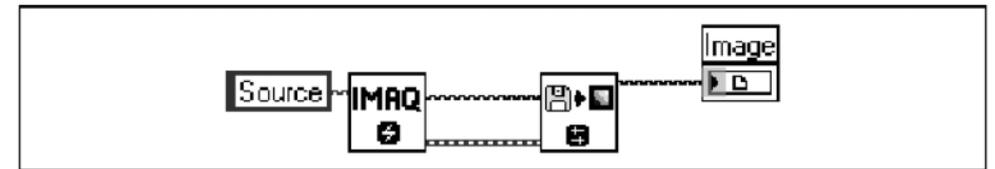


Рис. 17.4. Пример блок-диаграммы прибора, осуществляющего простейший вывод изображения на дисплей

Для конвертации данных из двумерного массива в изображение и наоборот предназначены ВП IMAQ ArrayToImage VI и IMAQ ImageToArray VI (Массив – изображение и Изображение – массив).

**Отображение данных.** Данные можно отображать, используя ВП, расположенные в палитре функций External Display (Внешний дисплей), или использовать элементы лицевой панели, расположенные в палитре Vision control (Управление отображением).

**Добавление данных о калибровке.** При желании пользователь может добавить информацию о калибровке и текущих установках в каждое собранное изображение. ВП IMAQ Set Calibration Info VI (Информация о калибровке) из палитры Vision Utilities ⇒ Calibration (Сервисные функции ⇒ Калибровка) принимает исходное изображение и информацию об исходных настройках. На выходе данного ВП изображение содержит в себе данные о калибровке.

**Анализ изображения.** После чтения и отображения данных, может возникнуть необходимость провести анализ изображения, целью которого является:

- определить, является ли качество изображения достаточным для выполнения поставленной задачи;
- получить значения параметров, необходимых при выполнении функций обработки данных.

Для определения качества изображения предназначены ВП Гистограмма, Гистограф (IMAQ Histogram, IMAQ Histogram VIs) и ВП Профиль линии (IMAQ LineProfile VI). Гистограмма позволяет получить информацию о насыщенности и контрастности изображения, а профиль линии – о распределении точек вдоль заданной линии. Кроме распределения вдоль линии можно задавать также определение распределения точек в заданном контуре (IMAQ ROIProfile VI). Все указанные ВП находятся в палитре Image Processing ⇒ Analysis (Обработка изображения ⇒ Анализ).

**Улучшение изображения.** Информация, полученная при анализе, может быть использована для улучшения качества изображения. В модуле для разработки систем машинного зрения для этого предустановлены функции поиска (палитра Image Processing ⇒ Processing), фильтрации (палитра Image Processing ⇒ Filters), полутоновой морфологии (палитра Image Processing ⇒ Morphology) и быстрого преобразования Фурье (палитра Image Processing ⇒ Frequency Domain).

## 17.2.2. Шаги по обработке изображений

**Выделение интересующего фрагмента изображения.** Инструменты для работы с фрагментами изображения (Region-of-interest – ROI) в IMAQ Vision позволяют выделять интересующие области как программно с помощью инструментов, расположенных в палитрах Machine Vision ⇒ Select Region of Interest (Машинное зрение ⇒ Задать область поиска) и Vision Utilities ⇒ Region of Interest ⇒ Region of Interest Conversion (Сервисные функции ⇒ Преобразование области поиска), так и в интерактивном режиме с помощью ВП IMAQ ConstructROI VI (Конструктор ROI) или дисплея на лицевой панели. Эта библиотека ВП может обрабатывать следующие формы: точка, линия, многоугольник, прямоугольник, вращаемый прямоугольник, круг, эллипс и кольцо.

**Монохромные измерения.** Для выполнения монохромных измерений предназначены функции измерения яркости и качественного анализа, расположенные в палитрах Machine Vision ⇒ Measure Intensities (Машинное зрение ⇒ Измерение интенсивности) и Image Processing ⇒ Analysis (Обработка изображения ⇒ Анализ) соответственно. Кроме того, с помощью ВП IMAQ Centroid VI (Центроид) возможно вычисление положения энергетического центра изображения.

**Цветовые измерения.** С помощью библиотеки IMAQ Vision можно анализировать и обрабатывать компоненты 8- и 16-битовых изображений. Используя ВП, расположенные в палитре Vision Utilities ⇒ Color Utilities (Сервисные функции ⇒ Сервисные функции цвета), исходное цветное изображение можно преобразовать в различные наборы первичных компонентов, таких как RGB, HSI, HSL или HSV. Применение ВП, расположенных в палитре Image Processing ⇒ Color Processing (Обработка изображения ⇒ Обработка цвета), позволяет сравнить цветовое содержание различных фрагментов снимка.

**Детальный анализ.** Данный вид обработки применяется для получения статистической информации, такой как размер, число, местоположение и наличие частиц. Указанная информация необходима при решении многих задач технического

зрения, например, обнаружение дефектов пайки на плате или неоднородностей на кремниевых подложках. Последовательность выполнения операций при детальном анализе следующая:

1. С помощью ВП IMAQ Threshold VI, IMAQ AutoBThreshold VI, IMAQ MultiThreshold VI (Установка пороговых значений) осуществляется настройка изображения таким образом, чтобы удалить фон.
2. С помощью функций палитры Image Processing ⇒ Morphology (Обработка изображения ⇒ Морфология) производится улучшение изображения и удаление ненужных деталей.

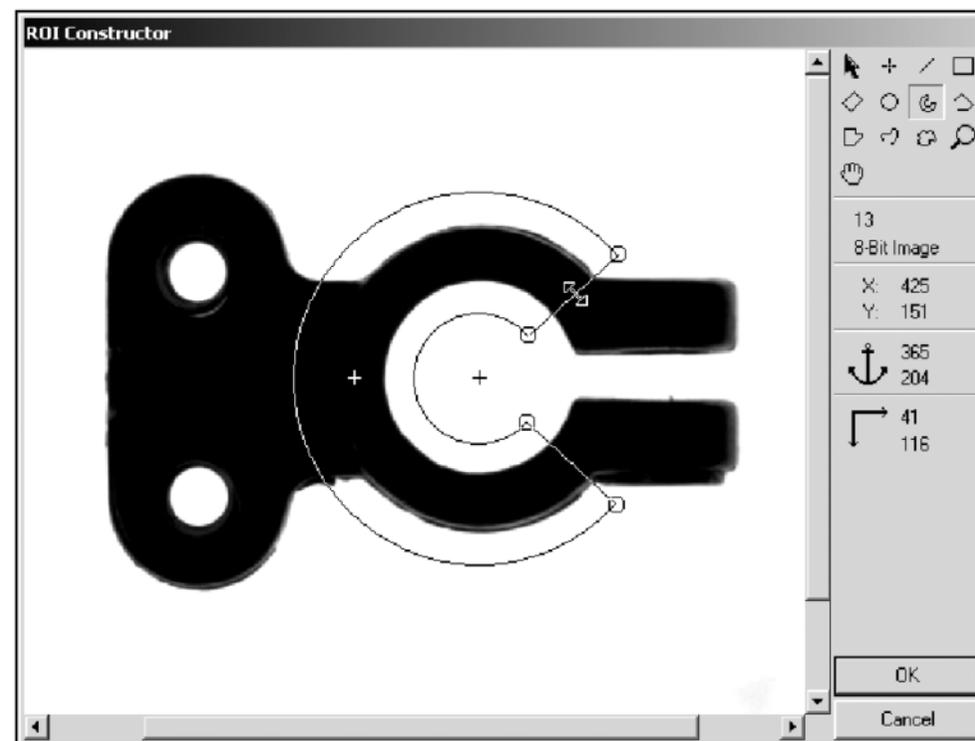


Рис. 17.5. Конструктор области поиска (ROI)

**Поиск, идентификация и классификация объектов.** Большинство задач по инспекции изображений представляют собой обнаружение наличия или отсутствия заданных фрагментов, определение набора параметров и выявление соответствия их заданным спецификациям. Данные измерения базируются на характерных особенностях объекта, представленного в изображении. Алгоритмы обработки традиционно классифицируют информацию, содержащуюся в изображениях, на типы: грани, поверхности и текстуры или образцы. Различные алгоритмы машинного зрения извлекают информацию одного или нескольких типов.

Определители граней (Edge detectors) с высокой точностью определяют границы объекта, что можно использовать при измерении величины элемента по методу фиксации (clamping) и классификации объекта по его конфигурации. Также возможно задавать сложные области инспекции (например, в форме круга) для определения точек пересечения с гранями объекта.

Алгоритмы сопоставления с образцом (Pattern matching) позволяют с высокой точностью определять положение объектов, идентичных по заданному набору характеристик опорного изображения. Найденные фрагменты измеряются посредством вычисления их длины, угловых размеров и других параметров.

Учитывая все изложенное выше, можно сделать вывод о том, что возможности библиотеки IMAQ Vision позволяют решать в LabVIEW большинство известных задач машинного зрения, однако необходимо учитывать то, что точность измерений в большой мере зависит от стабильности условий получения изображения. Разрешающая способность камеры, освещение, оптика, контроль вибраций и параметры внешней среды – ключевые компоненты системы сбора и анализа изображений.

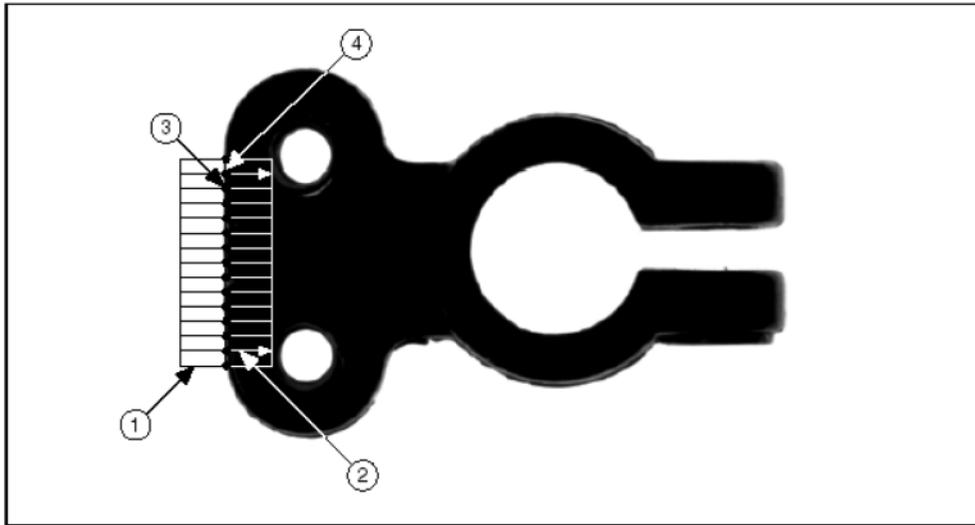


Рис. 17.6. Пример определения граней объекта в интерактивном режиме:  
1 – область поиска; 2 – линии поиска; 3 – обнаруженные точки пересечения;  
4 – линия, аппроксимирующая найденные точки

## Использование LabVIEW в телекоммуникации. Пакет NI Modulation Toolkit



Библиотека NI Modulation Toolkit (набор инструментов для работы с модуляцией) добавляет в LabVIEW функции генерации, анализа, визуализации и обработки стандартных и специальных видов модуляции. Использование данной библиотеки компонентов позволяет значительно ускорить процесс разработки приложений для задач исследования, тестирования систем связи и компонентов модулирующих и демодулирующих сигналов. Возможности библиотеки NI Modulation Toolkit довольно обширны и включают в себя работу с такими видами модуляции, как АМ, ЧМ, ФМ, амплитудная манипуляция, частотная манипуляция, манипуляция с минимальным частотным сдвигом, гауссова манипуляция, фазовая манипуляция, квадратурная фазовая модуляция, амплитудно-фазовая модуляция и квадратурная амплитудная модуляция (AM, FM, PM, ASK, FSK, MSK, GMSK, PSK, QPSK, PAM, QAM). Перечисленные виды являются основой многочисленных беспроводных технологий, таких как стандарты 802.11 a/b/g, ZigBee (802.15.4), WiMax (802.16a), RFID, а также спутниковой связи и коммерческого вещания.

Результатом интеграции пакета Modulation Toolkit и LabVIEW является возможность комбинирования специального анализа модуляционных характеристик и универсальных функций анализа, обработки, визуализации и контроля, предустановленных в LabVIEW. Системы на базе аналоговой или цифровой модуляции, построенные с помощью указанных функций, позволяют не только работать в режиме симуляции, но и проводить тестирование качества модуляции передатчика и демодуляции приемника.

При установке библиотеки для работы с модуляцией в LabVIEW появляются новые палитры ВП и более чем 100 примеров исходного кода. Новые компоненты можно использовать для аппаратно-независимой работы с данными из файла или сгенерированными моделью. Их также можно использовать для разработки приложений высоких частот. LabVIEW позволяет использовать функции данной библиотеки совместно с функциями, осуществляющими управление и обмен данными с реальными устройствами. Это могут быть как встраиваемые в ПК устрой-

ства, производимые компанией National Instruments, так и измерительные системы других производителей, подключаемые к ПК посредством любых коммуникационных протоколов, поддерживаемых LabVIEW. Пакет Modulation Toolkit позволяет работать с устройствами PXI-5660 (радиочастотный векторный анализатор сигналов), PXI-5670 и PXI-5671 (радиочастотные векторные генераторы сигналов) компании National Instruments. Для более низких частот также возможна работа с модуляцией при использовании встраиваемых цифровых осциллографов, генераторов сигналов произвольной формы и устройств цифрового ввода/вывода.

## 18.1. Использование функций библиотеки NI Modulation Toolkit при построении цифровых систем связи

Типичная цифровая система связи включает в себя передатчик, приемник и канал передачи данных (см. разд. 2).

Передатчик содержит блоки кодирования источника, кодирования канала, модуляции, моделирования реальных ухудшений сигнала и преобразования частоты. Лицевая панель передатчика сигналов с квадратурной фазовой модуляцией типа DQPSK представлена на рис. 18.1. Глазковые диаграммы ФМ-сигнала, связывающие время и состояния фазы – в фазе или в квадратуре, показаны на рис. 18.2. Приемник включает в себя блоки преобразования частоты, согласованной фильтрации, компенсации потерь, декодирования канала и источника сигнала. Результаты работы системы можно определить, используя инструменты для измерений и визуализации.

## 18.2. Кодирование

Источник данных генерирует информационный сигнал, посылаемый получателю. Этот сигнал может быть или аналоговым сигналом (например, речевым), или цифровым сигналом (например, битовой последовательностью). Информационный сигнал – это типичный немодулированный сигнал, представленный уровнем напряжения.

На рис. 18.3 представлена обобщенная модель цифровой системы связи.

Блок кодирования источника производит сжатие данных. Например, ATSC-стандарт для передачи цифрового видео (DVB), кодирует изображение, которое будет передано, используя алгоритм MPEGII. Примеры других типов алгоритмов сжатия, обычно используемых в исходном кодировании: А-закон, Му-закон, JPEG, А-87.6

Блок кодирования канала производит добавление избыточных битов к потоку данных, чтобы увеличить устойчивость приемника к шуму и помехам в канале. На выходе блока кодирования канала – последовательность нулей и единиц. Среди

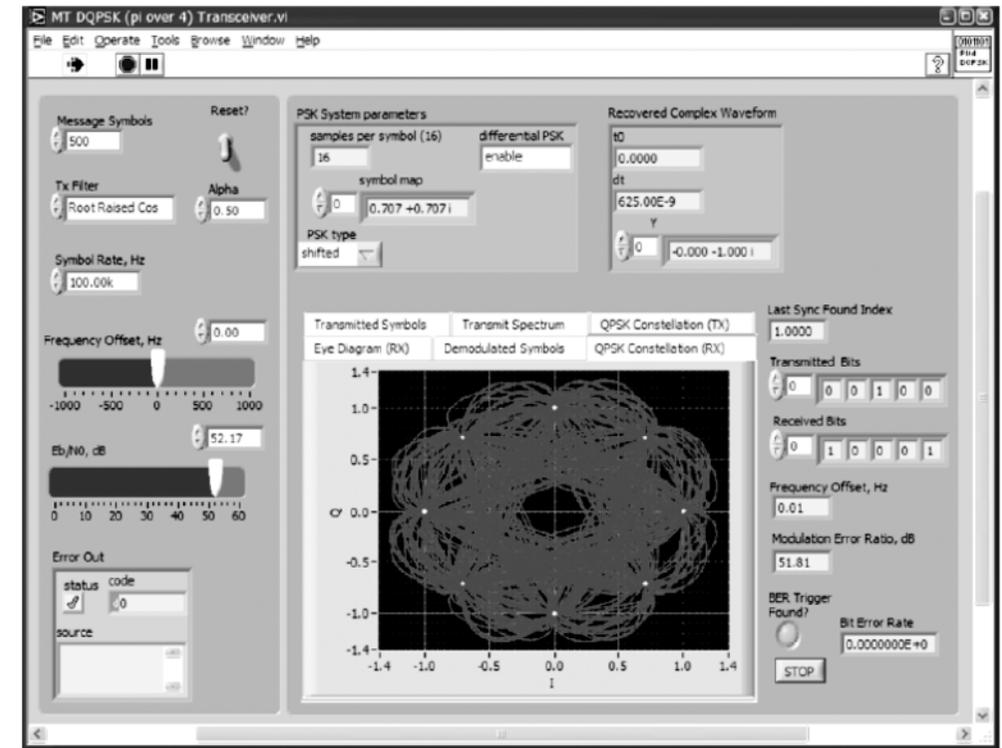


Рис. 18.1. Лицевая панель передатчика сигналов с модуляцией типа DQPSK (квадратурная фазовая модуляция)

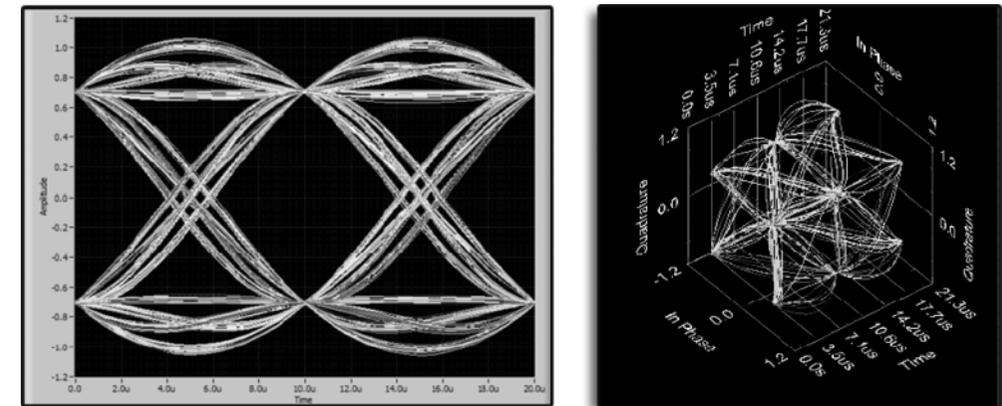


Рис. 18.2. Двухмерная и трехмерная глазковые диаграммы ФМ-сигнала

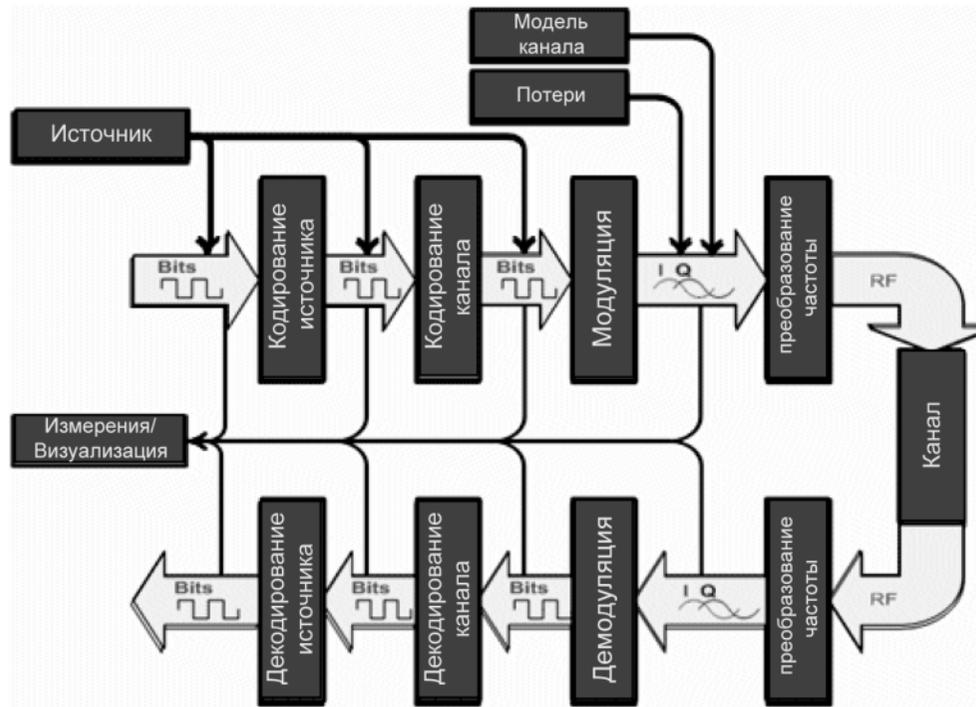


Рис. 18.3. Модель цифровой системы связи

самых популярных схем с исправлением ошибок являются блочное и сверточное кодирование.

В библиотеке NI Modulation Toolkit для кодирования канала предназначены следующие ВП:

- MT Hamming Encoder VI – код Хэмминга;
- MT Golay Encoder VI – код Голея;
- MT Reed Solomon Encoder VI – код Рида – Соломона;
- MT BCH Encoder VI – код BCH;
- MT Convolutional Encoder VI – сверточное кодирование;
- MT Spread Symbols VI – код, определенный пользователем.

### 18.3. Модуляция

Блок модуляции конвертирует информационный сигнал в синфазную (I) и квадратурную (Q) компоненты данных. Этот блок также уменьшает влияние межсимвольных помех. В зависимости от типа информационного сигнала и передачи данных, используются различные методы модуляции. Например, в амплитудной модуляции (AM) информация представлена изменениями амплитуды несущего сигнала.

Для осуществления аналоговой модуляции в NI Modulation Toolkit представлены ВП MT Modulate AM VI (AM), MT Modulate FM VI (ЧМ) и MT Modulate PM VI (ФМ).

Для цифровой модуляции предназначены ВП: MT Modulate ASK VI (Амплитудная манипуляция), MT Modulate FSK VI (Частотная манипуляция), MT Modulate MSK VI (Модуляция с минимальным частотным сдвигом), MT Modulate PAM VI (Амплитудно-фазовая модуляция), MT Modulate PSK VI (Фазовая манипуляция), MT Modulate QAM VI (Квадратурная амплитудная модуляция).

### 18.4. Преобразование (повышение) частоты

Спектр модулированного сигнала переносится на частоту передачи, соответствующую радиочастотному диапазону (ВП MT Upconvert Base Band VI – Перенос спектра сигнала на радиочастоту).

### 18.5. Ухудшения (потери)

Все системы передачи (включая беспроводные, волоконно-оптические и медные) вносят некоторые искажения в исходный сигнал. Чтобы математически представить такие реальные искажения были разработаны различные типы моделей канала. Инструменты NI Modulation Toolkit позволяют генерировать модулируемое сообщение и добавлять к нему шум, ухудшения, и выбранную модель канала. Для этого предназначены следующие ВП: MT Add AWGN VI (Добавление гауссова шума), MT Add Multi Tone VI (Добавление мультитона), MT Apply IQ Impairments VI (Внесение ухудшений в синфазную и квадратурную составляющие сигнала), MT Add Phase Noise VI (Добавление фазового шума) и т. д.

### 18.6. Преобразование (понижение) частоты

Первый шаг в процесс демодуляции – преобразование принятого сигнала в квадратурные составляющие с понижением частоты (ВП MT Downconvert Passband VI – перенос спектра сигнала на низкую частоту). Этот процесс включает в себя смешивание реального сигнала со сгенерированным несущим колебанием, а затем смешанный сигнал проходит фильтр нижних частот.

### 18.7. Демодуляция

После понижения частоты сигнал подвергается демодуляции. Эта процедура является противоположностью модуляции и подразумевает извлечение исходного информационного сигнала из моделированного. Процесс цифровой демодуляции включает в себя: согласованную фильтрацию (MT Matched Filter Vis – согласованные фильтры); измерение периода и синхронизацию символов, сопровождаемую исправлением смещения частоты (MT Align to Ideal Symbols VIs – переход к идеальным символам).

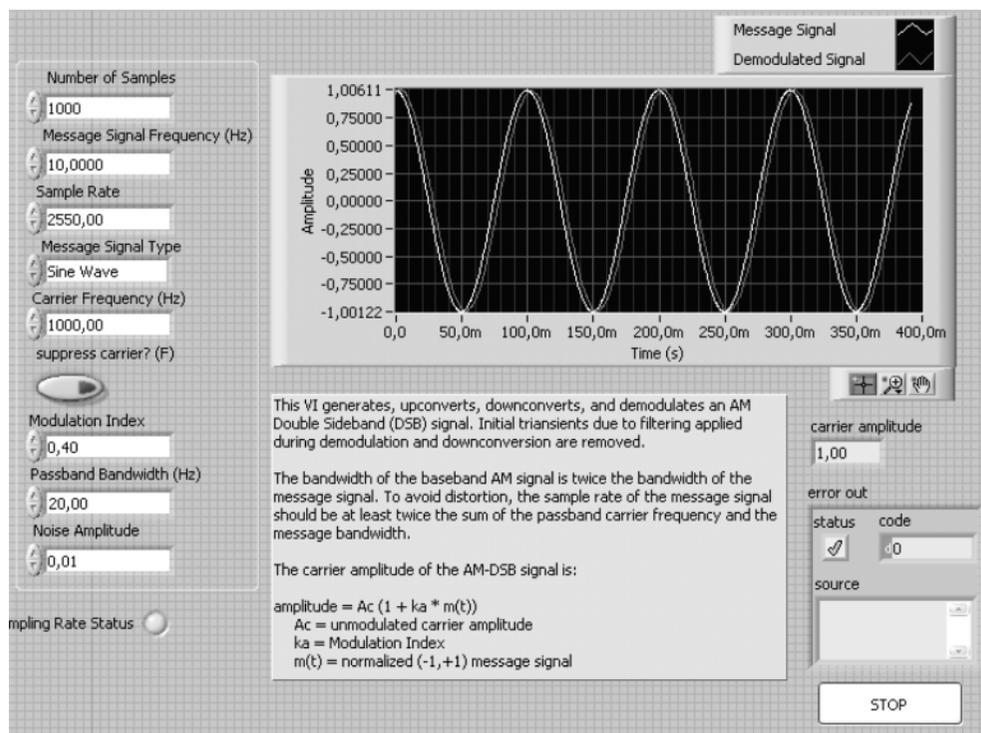


Рис. 18.4. Демодулятор сигнала с AM

Для демодуляции аналоговых сигналов предназначены ВП: MT Demodulate AM VI (демодуляция AM), MT Demodulate FM VI (демодуляция ЧМ), MT Demodulate PM VI (демодуляция ФМ).

Для демодуляции цифровых сигналов предназначены ВП: MT Demodulate ASK VI (амплитудная манипуляция), MT Demodulate MSK VI (модуляция с минимальным частотным сдвигом), MT Demodulate FSK VI (частотная манипуляция), MT Demodulate PAM VI (амплитудно-фазовая модуляция), MT Demodulate PSK VI (фазовая манипуляция) и MT Demodulate QAM VI (квадратурная амплитудная модуляция).

## 18.8. Компенсация потерь

Библиотека NI Modulation Toolkit использует адаптивный компенсатор с прямой связью, т. е. коэффициенты непрерывно изменяются, чтобы компенсировать потери в канале. Адаптивный компенсатор с прямой связью использует метод наименьших средних квадратов (least-mean-squared) с прямой связью (LMS). В начале процесса компенсации нужно установить биты обучения, чтобы обучить компенсатор. Как только обучение закончено, устройство переключается в режим обратной связи, в котором обучение происходит на основе принятых им решений.

В Modulation Toolkit предустановлены следующие виды цифровых компенсаторов: MT ASK Feedforward Equalizer VIs, MT PSK Feedforward Equalizer VIs, MT PAM Feedforward Equalizer VIs и MT QAM Feedforward Equalizer VIs – для сигналов с амплитудной и фазовой манипуляцией, а также амплитудно-фазовой и квадратурной амплитудной модуляцией соответственно.

## 18.9. Декодирование

Декодирование данных подразумевает удаление избыточных битов из последовательности и исправление любых ошибок, которые, возможно, возникли при передаче. Данная процедура обычно более сложна, чем кодирование и может быть довольно трудоемкой в вычислениях. Однако в настоящее время разработан ряд эффективных схем декодирования, например метод Витерби, который используется для данных, закодированных алгоритмом свертки.

Декодирование канала:

- MT Hamming Decoder VI – декодер Хэмминга;
- MT Golay Decoder VI – декодер Голея;
- MT RSDecoder VI – декодер Рида – Соломона;
- MT BCHDecoder VI – декодер Бозе – Хаудрари – Хокенгема (BCH)
- MT ConvolutionalDecoder VI – декодер свертки;
- MT DeSpreadSymbols VI – декодер пользовательского кода.

## 18.10. Измерение

Инженеры используют в системах связи различные инструментальные средства, чтобы оценить, насколько качественно (и количественно) прошел прием данных. Частота передачи ошибочных битов (BER), относительная ошибка модуляции (MER) – стандартные процедуры для данного вида анализа.

Эти и другие измерения можно производить, используя выход Measurements (Измерения) инструментов для цифровой демодуляции, ВП MT Measure Quadrature Impairments VIs (Измерение квадратурных искажений) и др.

## 18.11. Визуализация

Для визуализации данных предназначены графики и глазковые диаграммы:

- MT Display 3D Eye\_VI – трехмерная глазковая диаграмма;
- MT Format Eye Diagram\_VI – двухмерная глазковая диаграмма;
- MT Format Constellation\_VI – график совокупности;
- MT Format Trellis Diagram\_VI – решеточная диаграмма;
- MT IQ to XY Graph\_VI – график отображения синфазной и квадратурной составляющих сигнала.

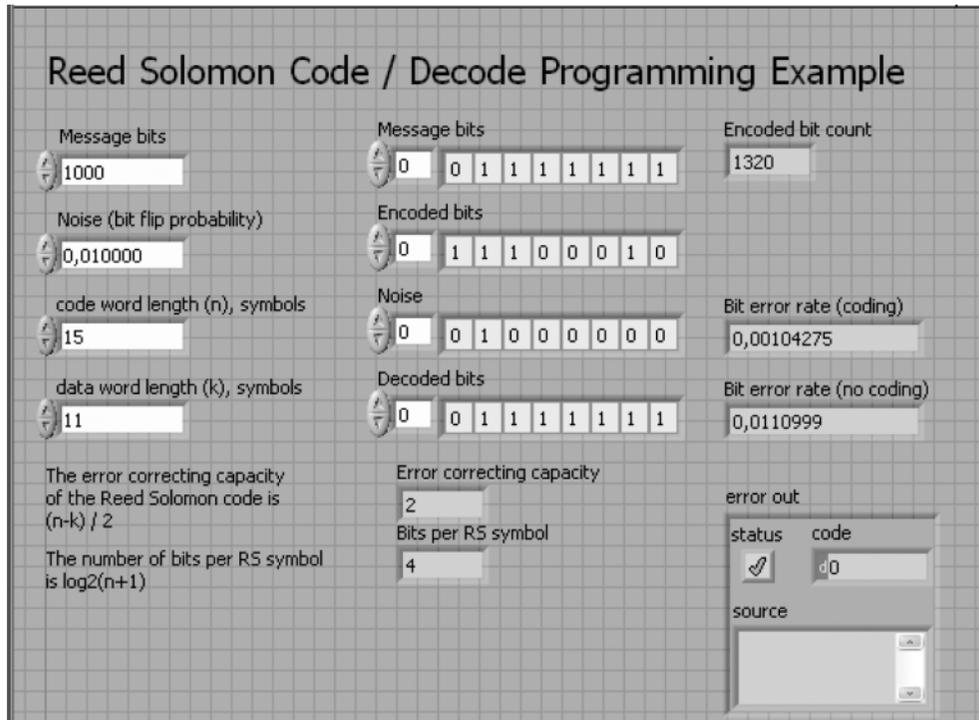


Рис. 18.5. Кодер/Декодер Рида – Соломона

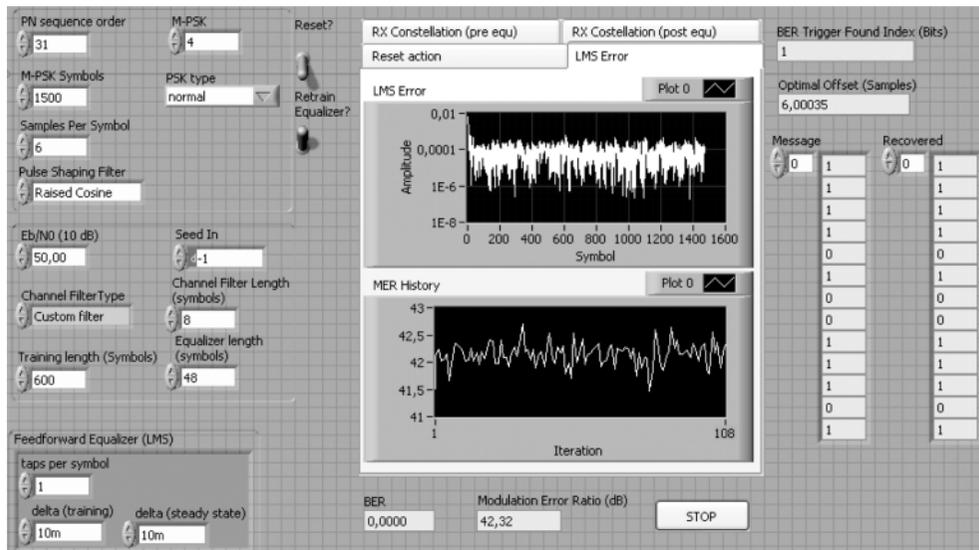
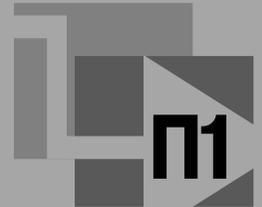


Рис. 18.6. Пример прибора, измеряющего относительную ошибку модуляции (MER) сигнала с фазовой манипуляцией

# Приложение Разработка алгоритмов на основе использования LabVIEW MathScript: Часть 1 – Ветви LabVIEW 8 MathScript



Это приложение представляет примеры знакомства с пакетом LabVIEW MathScript. Приведенный текст делает возможной попытку обучения разработке алгоритмов на основе LabVIEW MathScript на примерах, от шага к шагу. Далее в тексте приложения вводится интерактивный LabVIEW MathScript Window.

## Содержание приложения П1:

- Что такое MathScript?
- Ветвь LabVIEW – программа MathScript.
- Введение в LabVIEW и графическое программирование (Graphical Programming).
- Использование ветви MathScript для разработки алгоритмов.

## П1.1. Что такое MathScript?

В LabVIEW 8 можно выбрать наиболее эффективный синтаксис для технических расчетов, в котором создаются алгоритмы, исследуются понятия обработки сигналов или анализ результатов. Можно объединить в LabVIEW парадигму графического программирования потока данных с LabVIEW MathScript – математически ориентированный текстуальный язык программирования, который в основном совместим с синтаксисом *m*-файлов.

## П1.2. Ветвь LabVIEW – программа MathScript

LabVIEW MathScript позволяет работать на основе двух интерфейсов: LabVIEW MathScript Window или ветвь MathScript. Можно обратиться к П1.3 для получения более полной информации о LabVIEW MathScript Window.

С помощью ветви LabVIEW MathScript можно выполнить математическое описание графических программ (виртуальные приборы – ВП (VIs)) из LabVIEW. Можно также использовать ветвь MathScript для вставки текстуальных алгоритмов в ВП, и затем использовать графическое программирование LabVIEW в среде прибора для описания добавочных роликов, слайдов, кнопок, графиков и других средств управления и индикации. В добавление к этому можно ввести групповое описание в текст редактора окна (рис. П1.1).

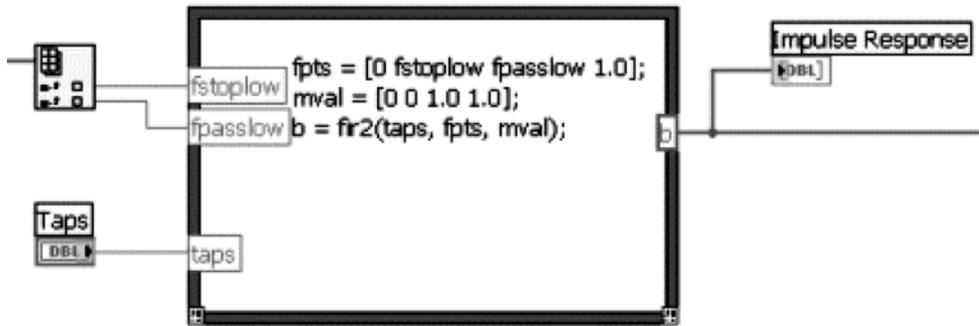


Рис. П1.1

## П1.3. Введение в LabVIEW и Графическое программирование (Graphical Programming)

LabVIEW есть графически развитая платформа для проектирования, управления и контроля. В LabVIEW встраивается пользовательский интерфейс, фронтальная панель с элементами управления и индикаторами. Элементы управления являются роликами, кнопками, циферблатами и другими механизмами ввода. Индикаторы являются графиками, светодиодами и другими указателями. После того как построен пользовательский интерфейс, можно добавить код, использующий ВП и структуры контроля объектов на фронтальной панели. Блок-диаграмма содержит этот код (рис. П1.2).

Для построения фронтальной панели и блок-диаграммы ВП необходимо выполнить следующие действия:

1. Запустить LabVIEW.
2. Из окна **Getting Started** выбрать **File** ⇒ **New VI** или нажать **Blank VI**, связанный с созданием новых ВП (рис. П1.3).

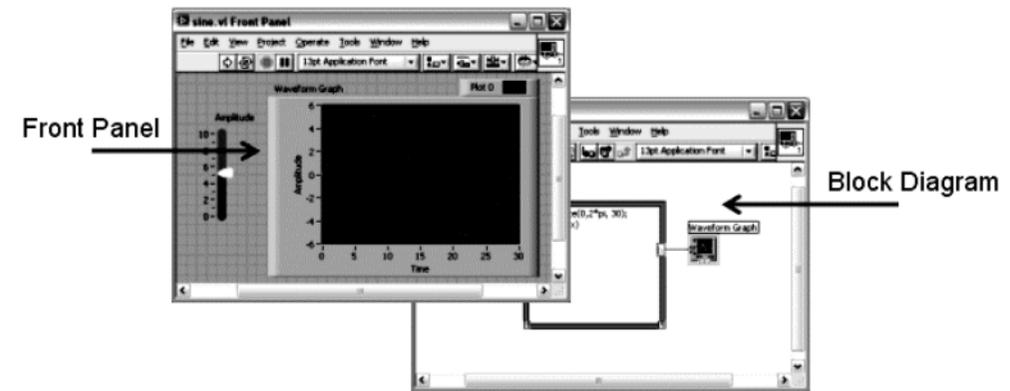


Рис. П1.2

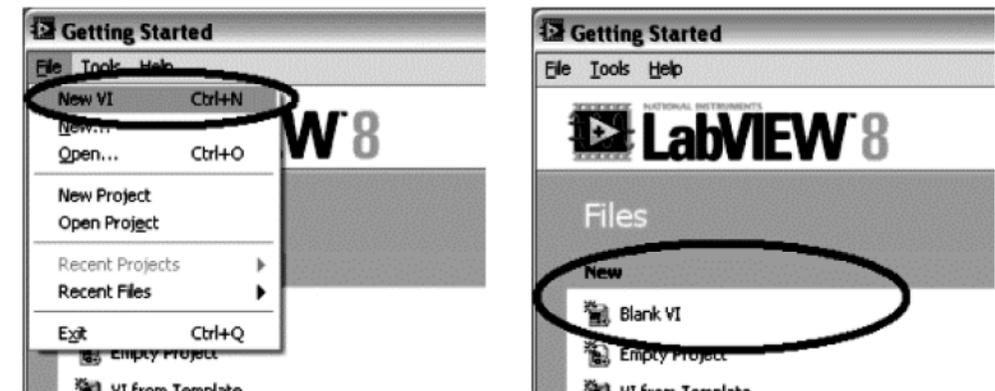


Рис. П1.3

3. На фронтальной панели ВП выбрать **View** ⇒ **Controls Palette** на дисплее палитры **Controls**.
4. Далее курсор переместить на икону палитры **Modern**, расположенной на палитре **Numeric**. Затем перевести курсор на иконы палитры **Controls** с именем подпалитры, управления, при этом появляется индикатор в окантованной полоске ниже иконы.
5. Нажать кнопку на иконе **Numeric** дисплея палитры **Numeric**.
6. Переместить курсор на поверхность иконы на палитре **Numeric**, расположенной на **Vertical Pointer Slide**.
7. Нажать кнопку мыши на иконе **Vertical Pointer Slide** и перетащить ее на фронтальную панель (рис. П1.4).
8. Произвести двойной щелчок мыши на метке **Slide** в **Vertical Pointer Slide**. В тексте блока появится надпись типа **Amplitude** (рис. П1.5).

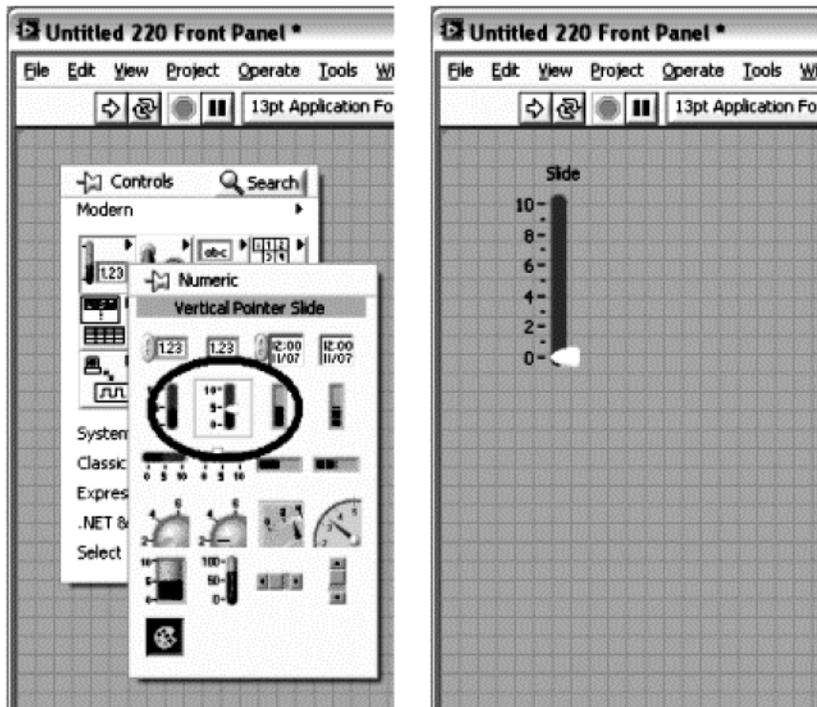


Рис. П1.4

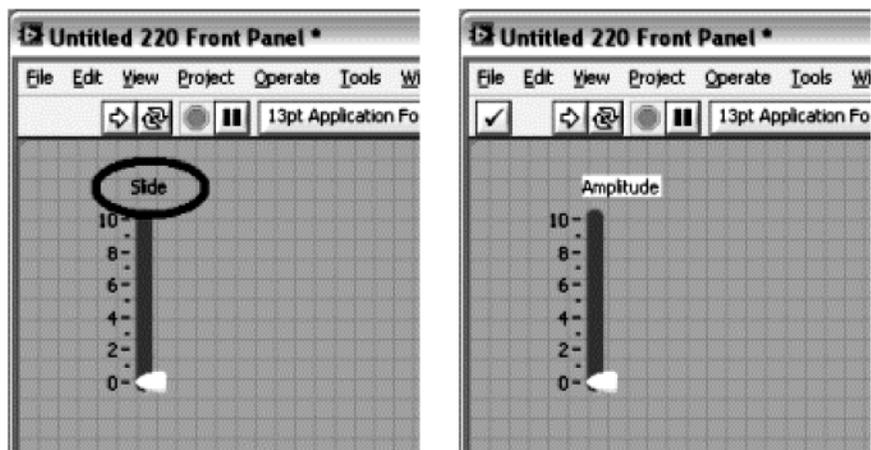


Рис. П1.5

9. На палитре **Controls** произвести щелчок на палитре **Modern**, связанной с возвратом к верхнему уровню палитры **Modern**. Переместить курсор на икону палитры **Modern**, размещенной на палитре **Graph**.
10. Произвести щелчок мыши на иконке **Graph** дисплея палитры **Graph**.
11. Переместить курсор на икону палитры **Graph**, расположенной на Waveform Graph. Щелкнуть левой кнопкой мыши на иконке **Waveform Graph** и переместить ее на фронтальную панель. Можно видеть на дисплее результаты появления используемых индикаторов, диаграмм и других графических диаграмм на фронтальной панели ВП (рис. П1.6).

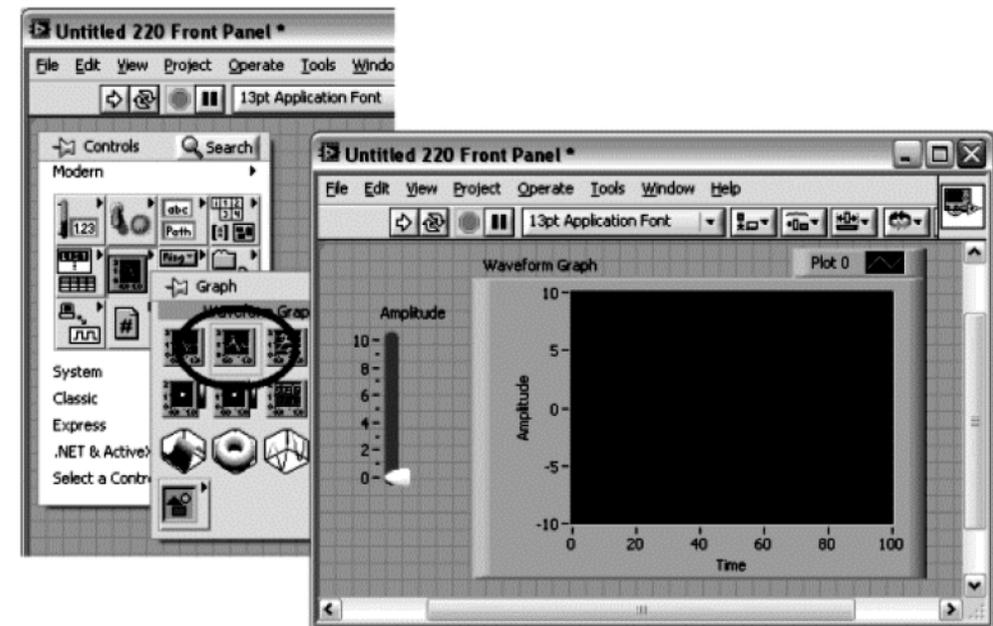


Рис. П1.6

12. Выбрать **Window** ⇒ **Show Block Diagram** на показанной блок-диаграмме ВП. Построение блок-диаграммы подобно построению фронтальной панели. Когда объекты размещаются на фронтальной панели, LabVIEW создает терминалы для этих объектов на блок-диаграмме. Можно соединить проводниками данных внутри или снаружи терминалов блок-диаграммы. Затем можно передать данные обратно и прямо между блок-диаграммой и фронтальной панелью.
13. Найти терминалы **Amplitude** и **Waveform Graph** на блок-диаграмме (рис. П1.7).
14. Выбрать **File** ⇒ **Save As** и записать ВП.

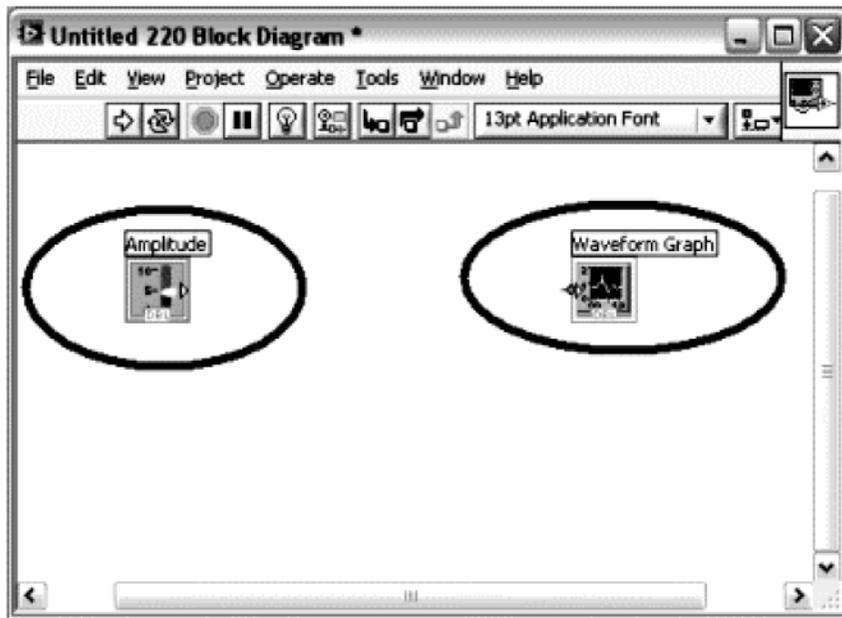


Рис. П1.7

## П1.4. Использование ветви MathScript для разработки алгоритмов

Можно использовать ветвь MathScript для разработки алгоритмов внутри LabVIEW. Необходимо выполнить следующие шаги для создания алгоритмов в ветви MathScript и запустить это описание в ВП, описанном в П1.3:

1. На блок-диаграмме выбрать палитру **View** ⇒ **Functions Palette** на палитре **Functions** дисплея.
2. Переместить курсор на икону палитры **Programming**, расположенной на палитре **Structures**.
3. Щелкнуть на иконе **Structures** палитры **Structures** дисплея.
4. Переместить курсор на икону **Structures** палитры, расположенной на ветви MathScript.
5. Щелкнуть мышью на иконе **MathScript Node**.
6. Щелкнуть на блок-диаграмме и переместить мышью в прямоугольную форму икону **MathScript Node** (рис. П1.8).
7. Щелкнуть внутри MathScript Node и выполнить следующие команды:

```
x = linspace(0, 2*pi, 30);
b = A*sin(x).
```

Можно заметить, что команда  $x = \text{linspace}(0, 2\pi, 30)$  создает новую переменную  $x$  и обычную переменную в 30 единичных отсчетов, распределенных между 0 и  $2\pi$ .

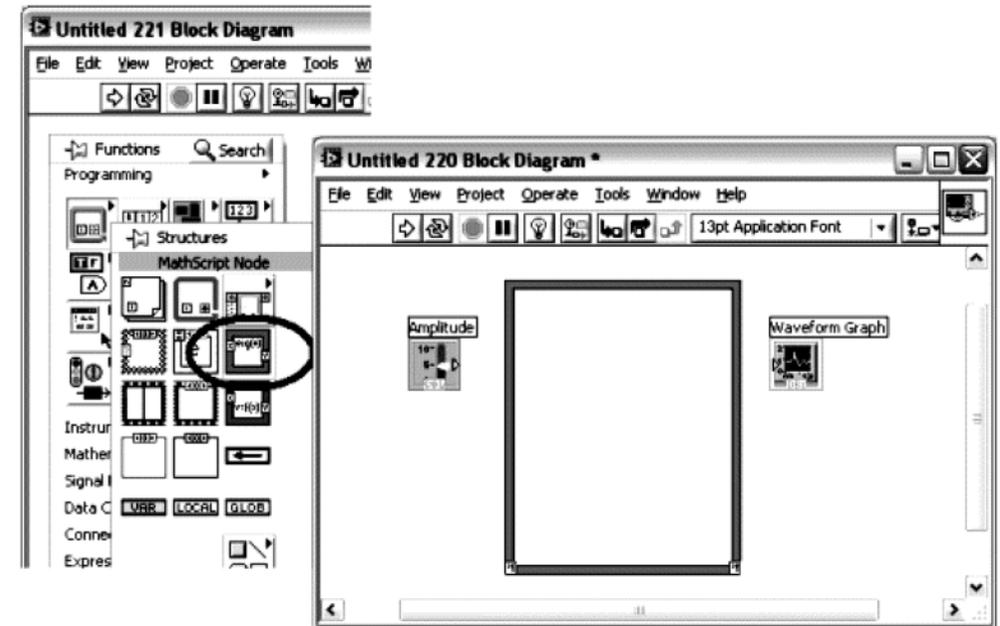


Рис. П1.8

**Дополнительно.** Можно щелкнуть правой кнопкой мыши на MathScript Node, затем также щелкнуть на ярлыке **Import** в меню, чтобы переместить созданное описание в  $m$ -файл (рис. П1.9).

8. Щелкнуть правой кнопкой мыши на рамке MathScript Node и выбрать **Add Input** из ярлыков меню.
9. На входном терминале добавить Тип A на входе для переменной A в ее описании (рис. П1.10).
10. Щелкнуть правой кнопкой мыши на рамке MathScript Node и выбрать **Add Output** из ярлыков меню.
11. На входном терминале добавить Тип b к выходу в описании переменной b.
12. Щелкнуть правой кнопкой мыши на **b** на выходном терминале и выбрать **Choose Data Type** ⇒ **1D-Array** ⇒ **DBL 1D** из ярлыков меню для определения типа данных выходной переменной b (рис. П1.11).
13. Переместить курсор на строку в терминале **Amplitude**. Курсор станет катушкой с проводом или инструментом Wiring. Использовать инструмент Wiring для соединения объектов вместе на блок-схеме.
14. Когда появится инструмент Wiring, щелкнуть на строке **Amplitude** терминала и затем щелкнуть на строке **A** входа MathScript Node для соединения двух объектов вместе. Данные перемещаются вдоль этого провода из терминала **Amplitude** в MathScript Node.
15. Аналогично соединяется выход **b** терминала **Waveform Graph**. Теперь можно получить соответствующие данные из описания и визуально анализиро-

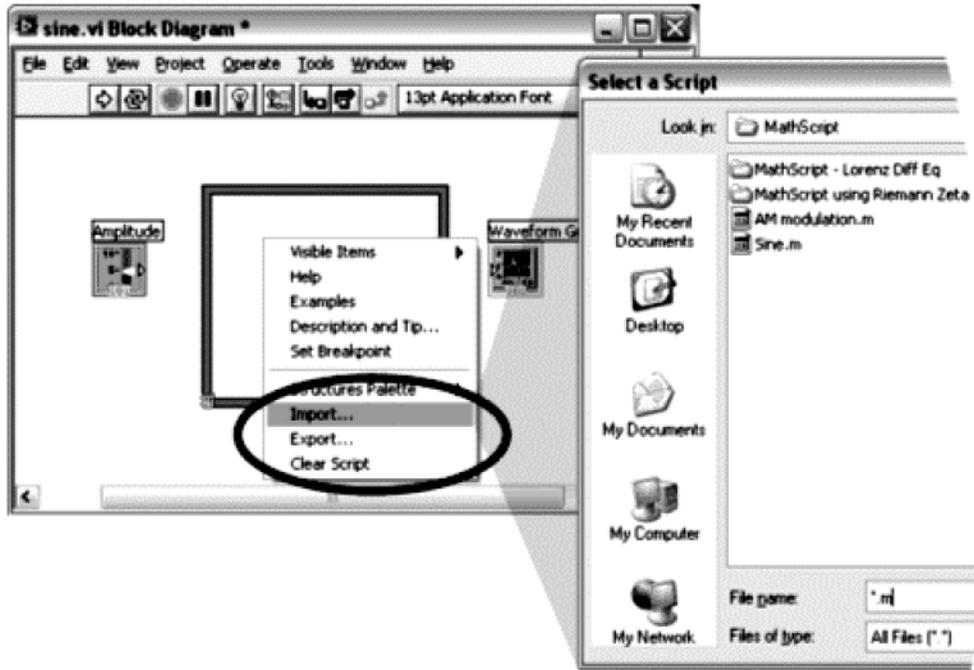


Рис. П1.9

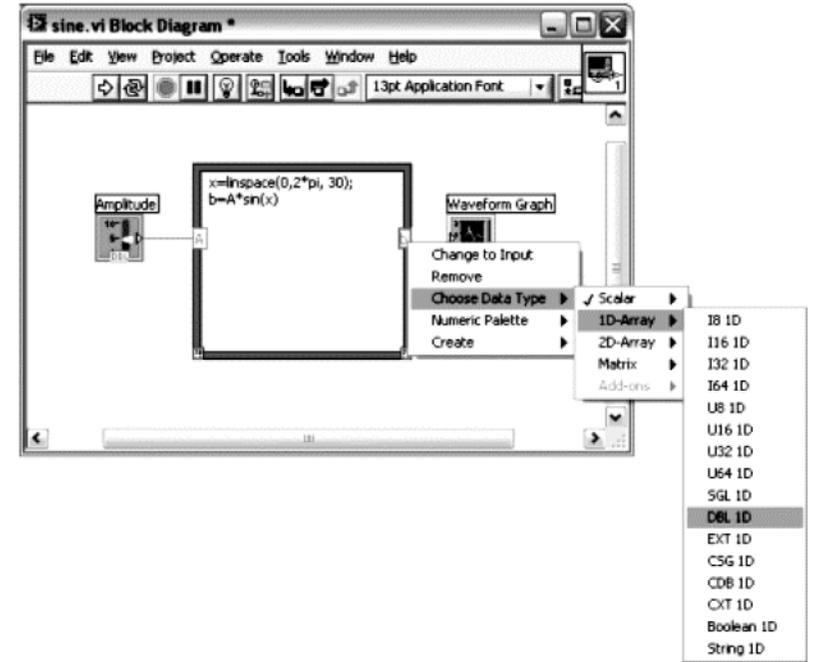


Рис. П1.11

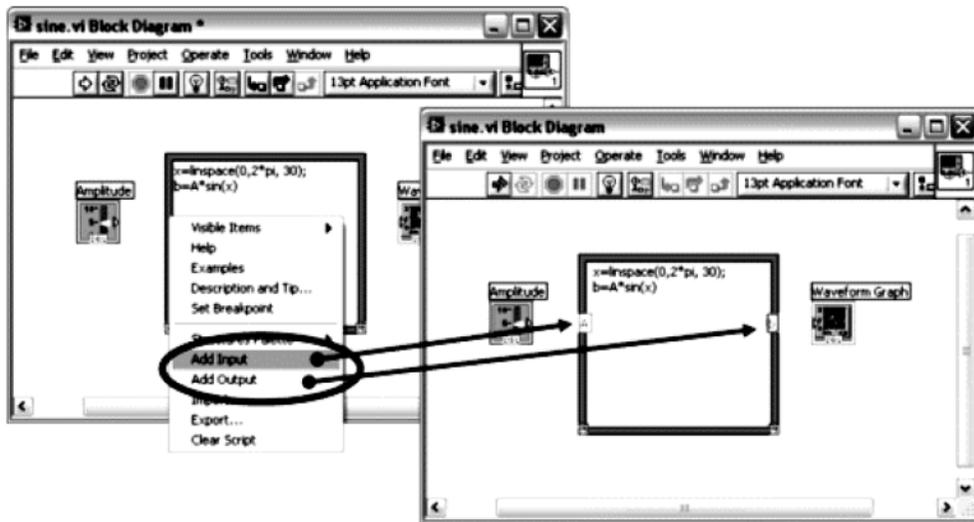


Рис. П1.10

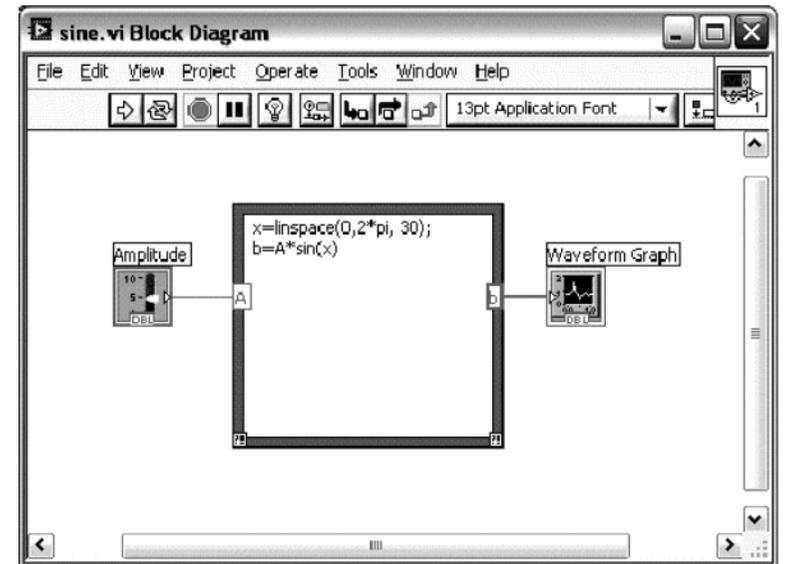


Рис. П1.12

вать результаты в пользовательском терминале с богатыми возможностями LabVIEW. Блок-диаграмма будет выглядеть, как показано на рис. П1.12.

16. Выбрать **Window** ⇒ **Show Front Panel** на дисплее фронтальной панели ВП.
17. Переместить движок управления **Amplitude** на место приблизительно 3.
18. Щелкнуть левой кнопкой мыши на кнопке **Run** или на белой строке в верхней части фронтальной панели. Заметим, что график сигнала заменяется графиком синусоидального колебания (рис. П1.13).

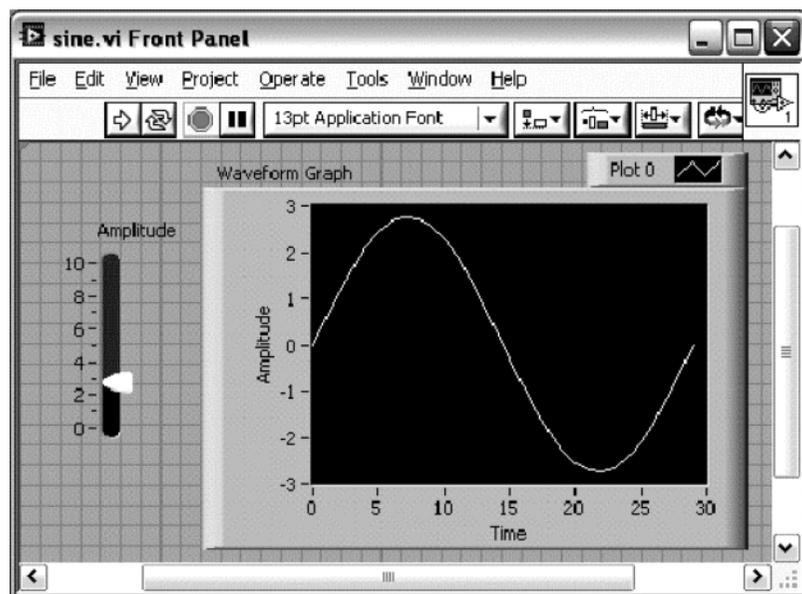


Рис. П1.13

19. Переместить движок управления **Amplitude** на различные значения и запустить ВП снова. Можно заметить, что синусоида обновляется соответственно.
20. (Дополнительно) Щелкнуть левой кнопкой мыши на кнопке **Run Continuously** или на круглой стрелке в верхней части фронтальной панели. Заметим, что график обновляется каждый раз, когда изменяется значение установки **Amplitude**.
21. Выбрать **File** ⇒ **Save** и записать ВП.

Можно обратиться к инструментам LabVIEW и функциям интерактивной палитры, диалогового окна или меню. LabVIEW также обеспечивает сотни ВП, которые можно использовать внутри созданных ВП, позволяет перемещать и включать эти ВП на блок-диаграмме для определения применений описания. Такие точечные щелчки позволяют существенно уменьшить время, необходимое для получения из первоначальных установок окончательного решения.

## П1.5. Окно LabVIEW MathScript

Можно работать с LabVIEW MathScript в двух интерфейсах: LabVIEW MathScript Window или MathScript Node.

Ниже показано, как можно использовать интерактивный MathScript Window, вводя команды одну за другой. Можно также вводить пакетное описание в виде простого текста в редакторе window, загружать из текстового файла или импортировать из отдельных текстовых редакторов. MathScript window обеспечивает немедленную обратную связь в различные формы, такие как графики и текст (рис. П1.14).

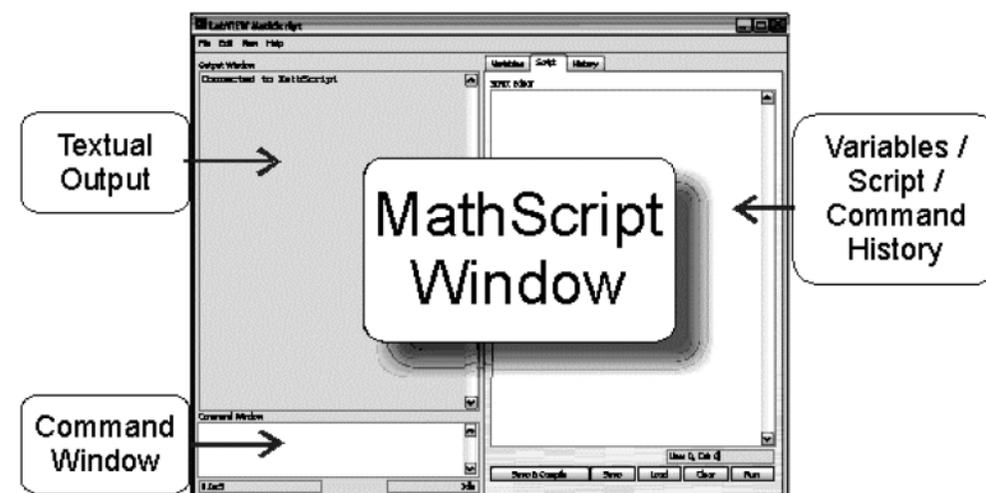


Рис. П1.14

На рис. П1.14 использованы следующие обозначения: **MathScript Window** – окно MathScript; **Textual Output** – вывод текста; **Command Window** – окно команд; **Variables** – переменные; **Script** – описание; **command history** – предыдущие команды.

### Использование MathScript Window для создания алгоритмов:

1. В окне **Getting Started** выбрать **Tools** ⇒ **MathScript Window**, чтобы активировать **MathScript Window**.
2. В текстовом поле **Command Window** ввести  $4+6+1$ , затем нажать клавишу **<Enter>**. Результат выполнения команды появится в **Output Window**. Можно ввести команды по одной для непосредственной обратной связи.
3. Щелкнуть на поле таблицы с обозначением **Variables**. LabVIEW обновит переменную, появится ответ под именем **Local** в дереве **Partition/Variable**, где содержатся последние выполненные команды.
4. Щелкнуть на поле таблицы с обозначением **Script** и ввести следующие команды:

```
x=linspace(0,2*pi,30);
b=sin(x)./(cos(x));
plot(x,b); axis([0 2*pi -20 20]).
```

- Можно заметить, что команда `x=linspace(0,2*pi,30);` создала новую переменную `x` и заполнила этой переменной 30 отсчетов, распределенных между 0 и  $2\pi$ .
- Щелкнуть кнопкой **Run** на странице **Script**. В окне **Plot 1** появляется и отображается на графике  $XY$   $b$  в зависимости от  $x$ . Можно закрыть окно **Plot 1**, щелкая  $x$  под правым углом окна.
- Щелкнуть на поле таблицы с обозначением **Variables** для отображения созданных переменных, таких как  $x$  и  $b$ .
- Выбрать  $b$  на дереве **Partition/Variable**. Таблица представляет число появляющихся величин  $b$ . Запустить переключатель **Graphical First?** для изменения числа отображаемых графиков на дисплее. Появляются результаты в виде графика. Когда запускается переключатель **Graphical First?**, можно увидеть графические отображения первых переменных величин быстрее, чем их цифровые представления (рис. П1.15).

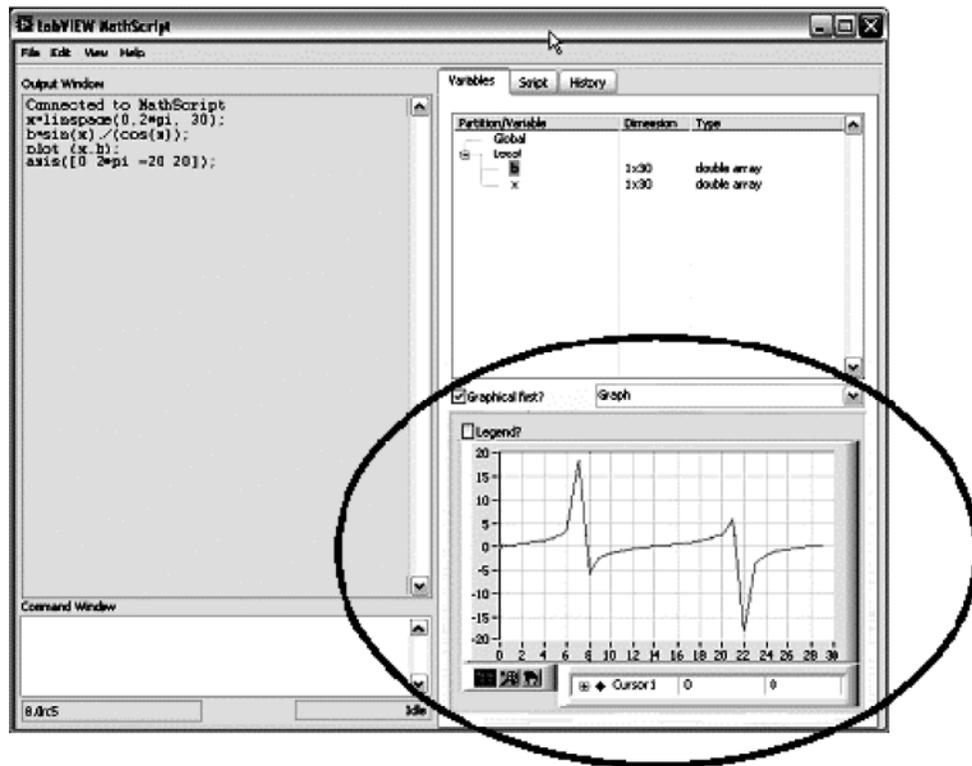


Рис. П1.15

- Щелкнуть на позиции **Script** и щелкнуть на кнопке **Load**. Выбрать `Mitra P2_1.m` (Mitra, Sanjit and Kaiser, James H., *Handbook for Digital Signal Processing* [New York: John Wiley and Sons, 1993], 21). Это основное описание тестового сигнала, а также применения фильтров текущего усреднения для тестирования сигналов.
- Щелкнуть кнопкой **Run** и запустить описание (script). Появляется диалоговое окно **Prompt User for Input**. Ввести положительные значения в **Desired length of the filter** = текстовое окно и щелкнуть кнопкой **OK**. Описание MathScript включает интерактивное диалоговое окно для ввода запросов пользователя.
- Описание `Mitra P2_1.m`, показанное ниже, использует подзаголовки команд для определения от одного до четырех подзаголовков на плане окна. После определения подзаголовка, последующие команды затрагивают этот подзаголовок. Например, команда заголовка, после команды подзаголовка, содержит подзаголовок, указанный в команде подзаголовка, с заголовком, указанным в команде заголовка. (For example, a plot command, following a subplot command, will populate the sub-plot specified in the subplot command with the plot specified in the plot command.)

```
% Program P2_1
% Simulation of an M-point Moving Average Filter
% Generate the input signal
n = 0:100;
s1 = cos(2*pi*0.05*n); % A low-frequency sinusoid
s2 = cos(2*pi*0.47*n); % A high frequency sinusoid
x = s1+s2;
% Implementation of the moving average filter
M = input('Desired length of the filter = ');
num = ones(1,M);
y = filter(num,1,x)/M;
% Display the input and output signals
clf;
subplot(2,2,1);
plot(n, s1);
axis([0, 100, -2, 2]);
xlabel('Time index n'); ylabel('Amplitude');
title('Signal #1');
subplot(2,2,2);
plot(n, s2);
axis([0, 100, -2, 2]);
xlabel('Time index n'); ylabel('Amplitude');
title('Signal #2');
subplot(2,2,3);
plot(n, x);
```



Mitra P2\_1.m

```
axis([0, 100, -2, 2]);
xlabel('Time index n'); ylabel('Amplitude');
title('Input Signal');
subplot(2,2,4);
plot(n, y);
axis([0, 100, -2, 2]);
xlabel('Time index n'); ylabel('Amplitude');
title('Output Signal');
axis;
```

Можно также использовать текстовое окно **Command Window**, чтобы найти больше информации о командах. Например, в окне **Output Window** появляются справки в текстовом окне **Command Window** и описание команд наряду с информацией о синтаксисе, входах, выходах и примеры.

## Приложение Справочник виртуальных приборов ВП [10] в телекоммуникациях



На рис. П2.1 представлены иконы виртуальных приборов для приложений в телекоммуникациях.

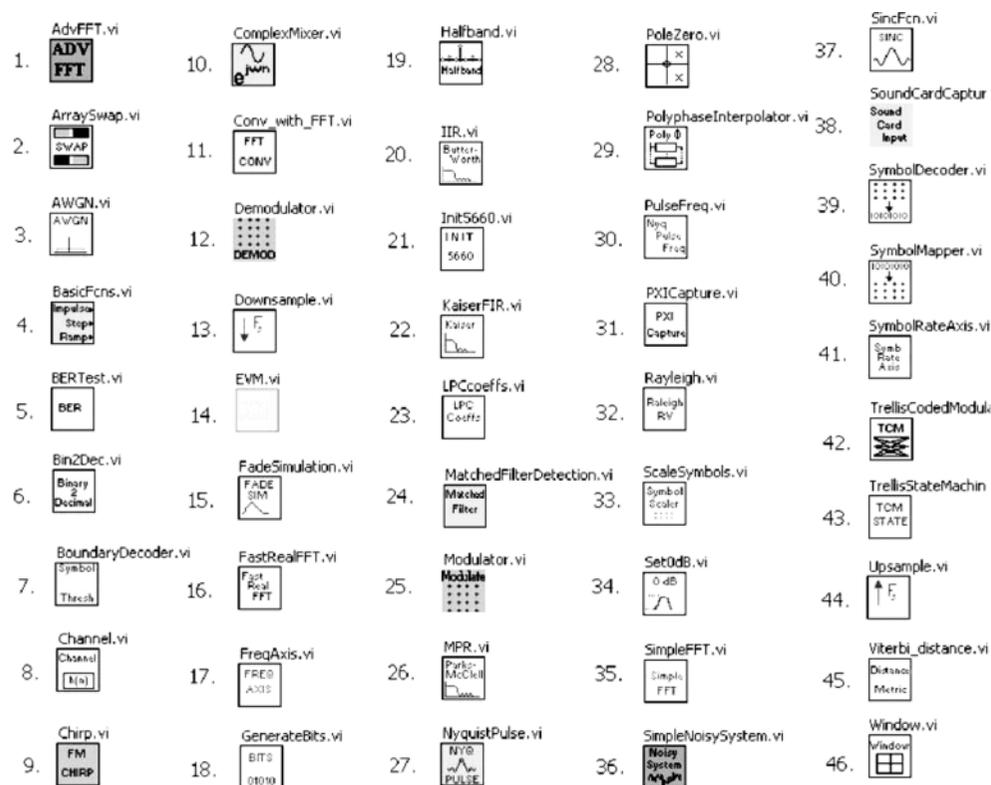


Рис. П2.1

1. **AdvFFT.vi** (рис. П2.2). Расширенная утилита БПФ. Этот ВП вызывается для использования алгоритма БПФ, дополняя длину входной выборки к числу по основанию 2. Ввод может быть реальным или комплексным и вывод БПФ сдвинут наполовину частоты дискретизации, а также выводится график от частоты.
2. **ArraySwap.vi** (рис. П2.3). Простая утилита для переключения положительных и отрицательных половин вывода БПФ (или любой массив в этом отношении).
3. **AWGN.vi** (рис. П2.4). Генерирует белый гауссов шум с заданной шумовой мощностью. Сгенерированный шум может быть реальным (рис. П2.4, а) или комплексным (рис. П2.4, б).
4. **BasicFens.vi** (рис. П2.5). Формирует  $\delta$ -импульс, импульс пилообразной формы и прямоугольный импульс с указанным числом выборок и задержкой. Амплитуда неизменна и равна 1.
5. **BERTest.vi** (рис. П2.6). Этот ВП вычисляет число ошибок в битах так же, как частоты передачи ошибочных битов формы колебания с 16-QAM (квадратурная амплитудная модуляция). Известная разрядная последовательность выводится из модулятора и сравнивается с восстановленной разрядной последовательностью после того, как форма колебания разрушена в ВП Channel.vi.
6. **Bin2Dec.vi** (рис. П2.7). Преобразовывает двойной входной массив к десятичным числам. Самый высокий индекс – наибольший значащий бит, и индекс 0 – наименьший значащий бит.
7. **BoundaryDecoder.vi** (рис. П2.8). Этот ВП выбирает самое близкое соответствие символа совокупности для восстановленных символов, основанных на граничных решениях. Для этой совокупности с 16-QAM области от 0 до  $\pm 1$  и  $\geq \pm 2$ .
8. **Channel.vi** (рис. П2.9). Этот ВП моделирует канал с замираниями, формируя огибающую с Рэлеевскими замираниями и применяя конфигурацию замираний к модулируемой форме колебаний.
9. **Chirp.vi** (рис. П2.10). Вычисляет линейную последовательность ЛЧМ-сигнала, где мгновенная частота колебания линейно увеличивается с коэффициентом наклона  $k$ .
10. **ComplexMixer.vi** (рис. П2.11). Формирует комплексный синусоидальный сигнал, смещенный по несущей частоте заданных длины и частоты для установленной частоты дискретизации.
11. **Conv\_with\_FFT.vi** (рис. П2.12). Выполняет линейную свертку с БПФ. Входной сигнал и коэффициенты фильтра дополнены нулями до длины, равной числу по основанию 2. БПФ продленных последовательностей умножены, и рассчитано обратное БПФ. Выход затем урезан для удаления дополнительных нулей.
12. **Demodulator.vi** (рис. П2.13). Демодулирует форму колебания 16-QAM. Дополнительные точки импульса, формирующего свертку, урезаны, форма колебания субдискретизирована и восстановленные символы масштабированы.

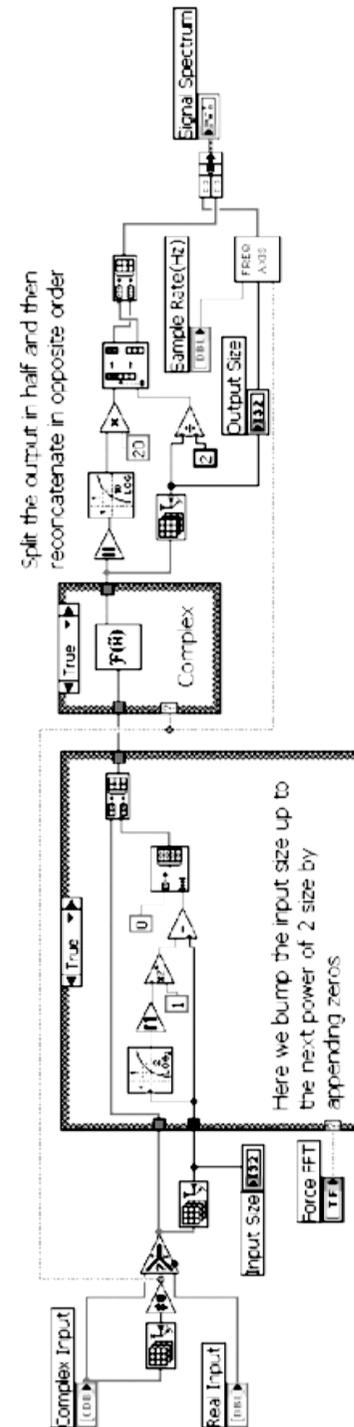


Рис. П2.2

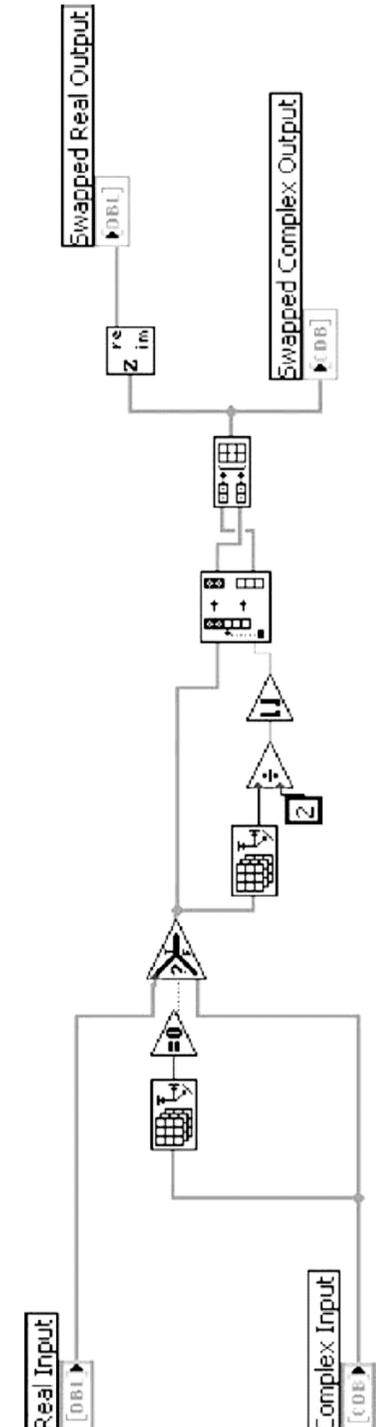


Рис. П2.3

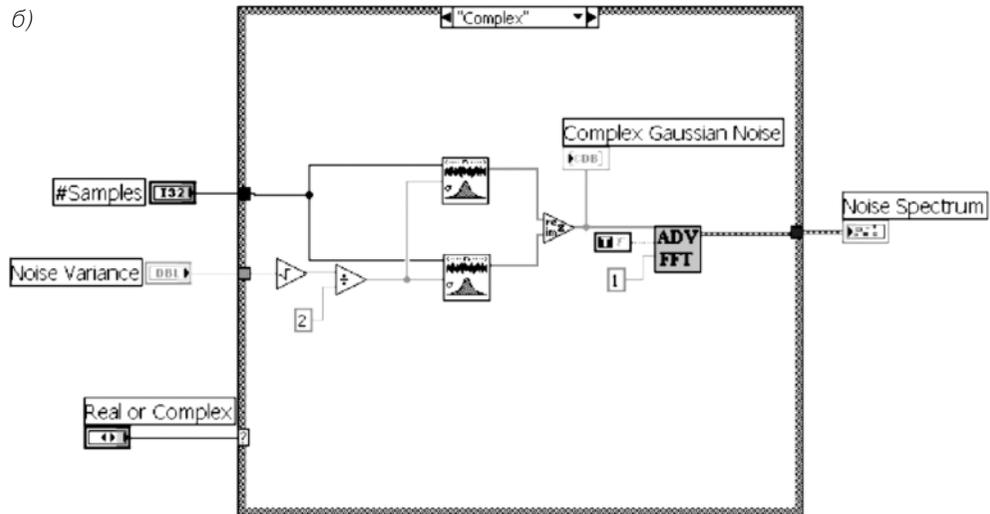
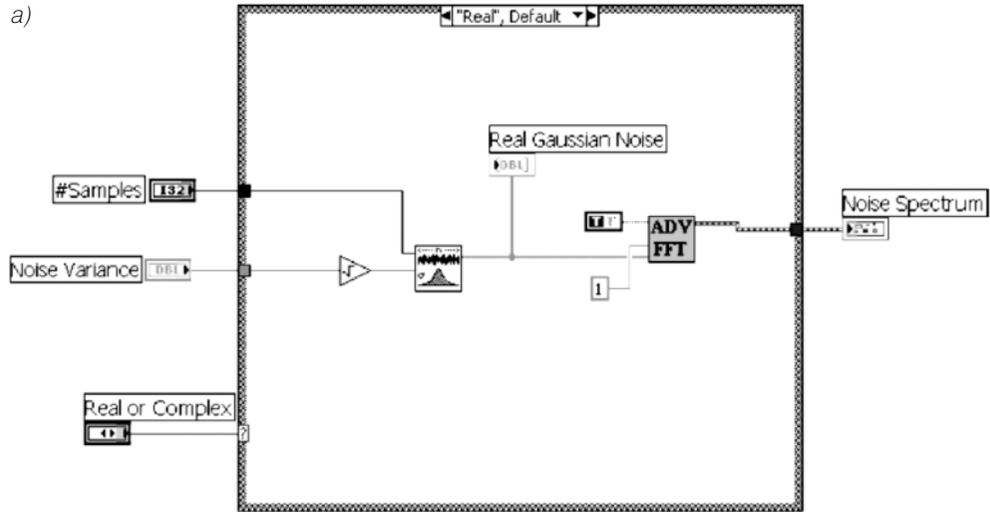


Рис. П2.4

13. **Downsample.vi** (рис. П2.14). Понижает частоту дискретизации дискретной формы колебания во времени, сохраняя только один из каждого блока выборки размером  $M$ . Фильтрация может быть необходима перед этой операцией для сохранения расширенной полосы прореженной последовательности внутри предела  $f_s / 2$ .
14. **EVM.vi** (рис. П2.15). Этот ВП вычисляет амплитуду вектора ошибки восстановленной последовательности символов, сравнивая ее с опорной пос-

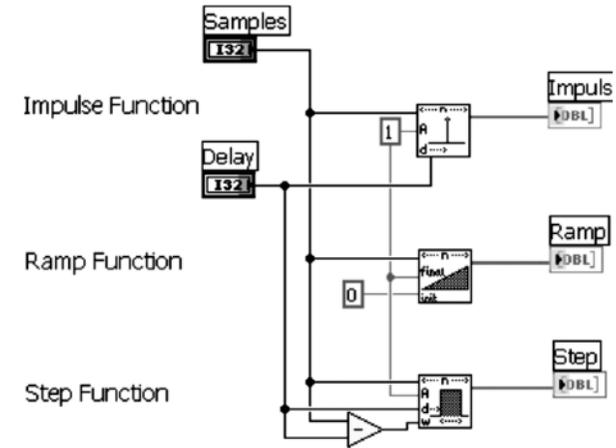


Рис. П2.5

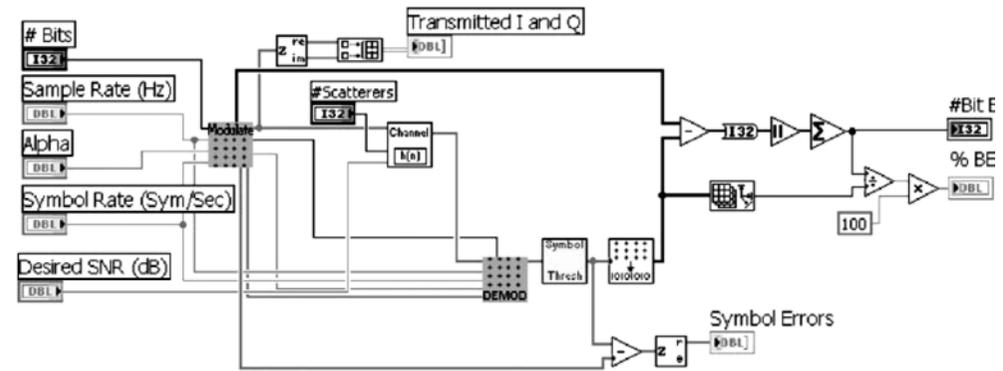


Рис. П2.6

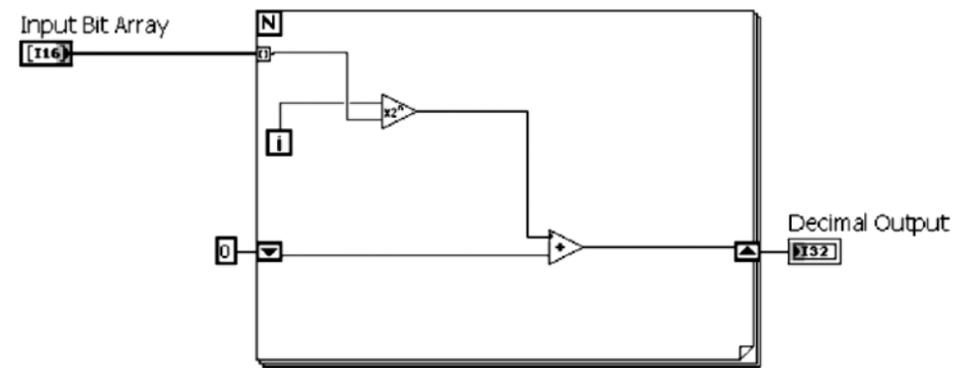


Рис. П2.7

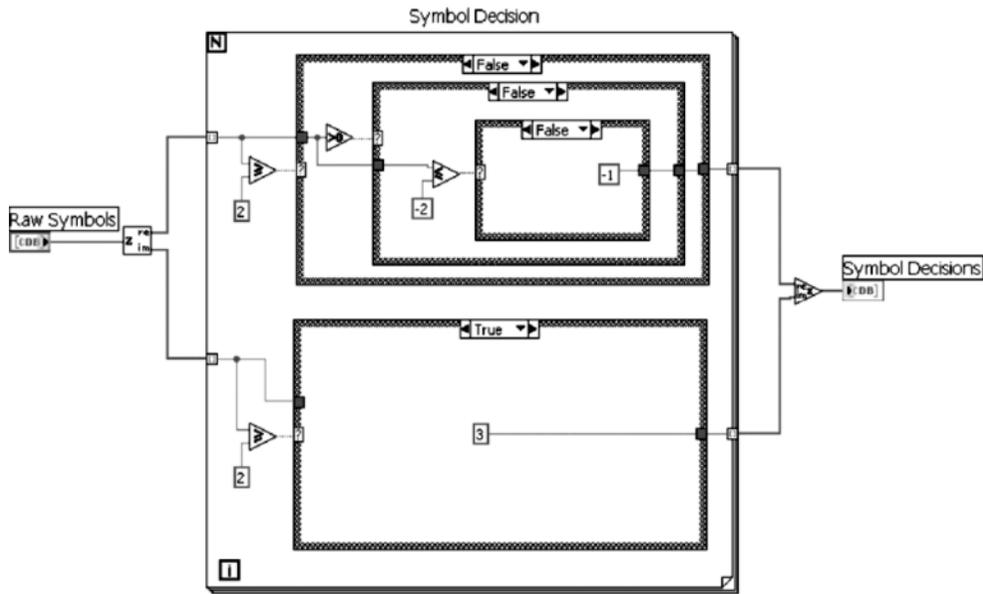


Рис. П2.8

ледовательностью. Амплитуда ошибок нормализована к амплитуде совокупности угловых символов.

15. **FadeSimulation.vi** (рис. П2.16). Моделирует среду с замираниями, генерируя форму колебания 16-QAM, применяя ВП Channel.vi для замираний, и демодулирует сигнал с замираниями.
16. **FastRealFFT.vi** (рис. П2.17). Вычисляет БПФ реальной последовательности длиной  $2N$ , повторно собирая ее в последовательность  $N$ -точек комплексов и используя комплексную БПФ подпрограмму. Оперирует со спектром  $N$ -точек комплексов, чтобы произвести  $2N$ -точки желательного БПФ.
17. **FreqAxis.vi** (рис. П2.18). Генерирует значения оси частот, чтобы составить график двустороннего или одностороннего спектра.
18. **GenerateBits.vi** (рис. П2.19). Формирует последовательность битов, используя генератор случайного числа в цикле и округляя результат до самого близкого целого числа.
19. **Halfband.vi** (рис. П2.20). Вычисляет коэффициенты фильтра для ФНЧ (использует алгоритм Parks-McClellan) с граничной частотой, равной  $1/4$  частоты дискретизации. Фильтр полуполосы имеет свойство, при котором каждый второй коэффициента равен нулю.
20. **HR.vi** (рис. П2.21). Генерирует ФНЧ с коэффициентами БИХ фильтра, используя проект ВП Butterworth LabVIEW. Порядок фильтра вычисляется из формул проекта.
21. **Init5660.vi** (рис. П2.22). Инициализирует пример PXI-5660 аппаратных средств. Решение и просмотр передает для чтения и закрывает функции.

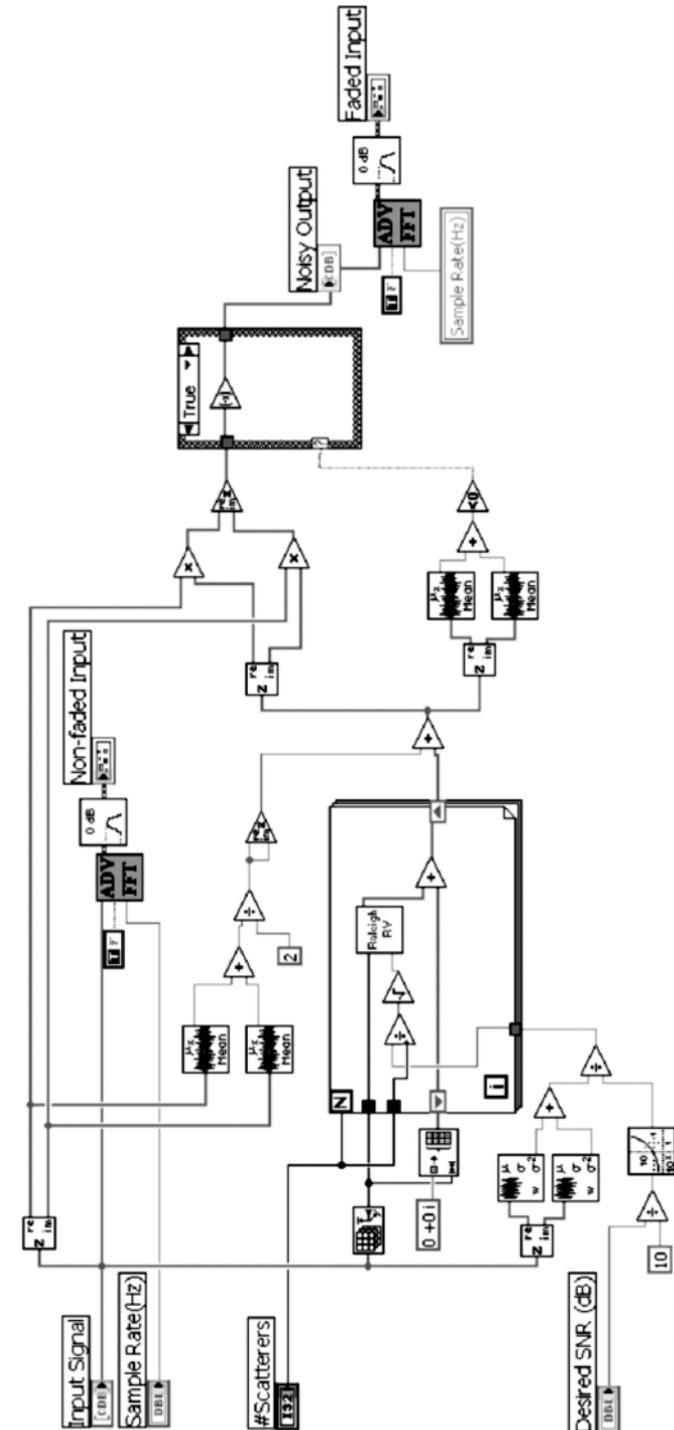


Рис. П2.9

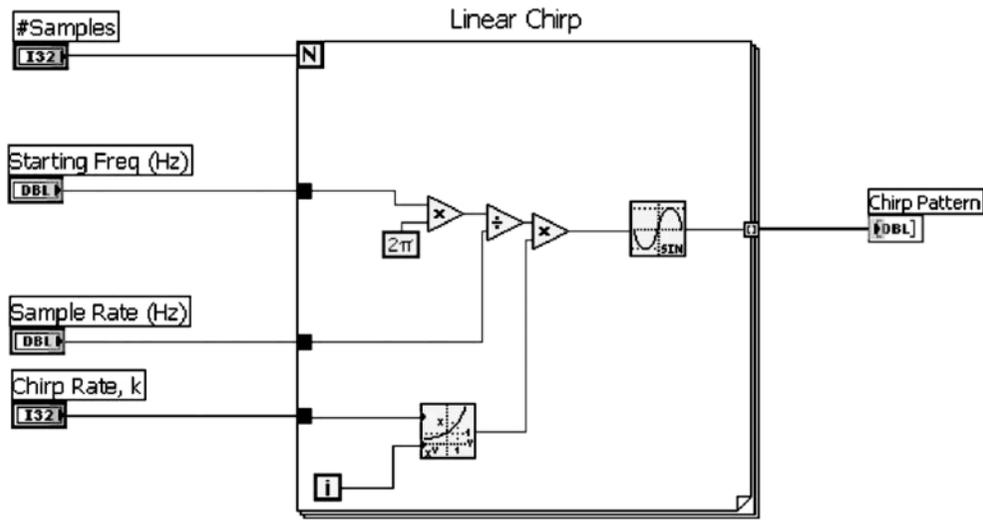


Рис. П2.10

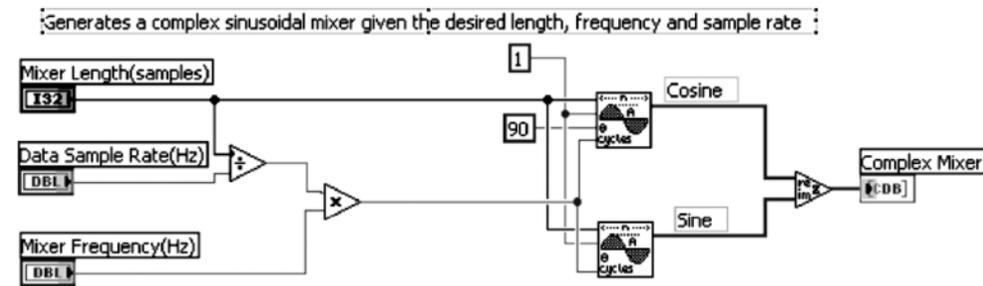


Рис. П2.11

22. **KaiserFIR.vi** (рис. П2.23). Этот ВП генерирует цифровые коэффициенты фильтра, основанные на методе окна Кайзера.
23. **LPCcoeffs.vi** (рис. П2.24). Вычисляет набор коэффициентов фильтра для линейной прогнозирующей модели речевого фрагмента. Эти коэффициенты могут использоваться для синтеза фрагмента речи как модель всеполюсного фильтра.
24. **MatchedFilterDetection.vi** (рис. П2.25). Демонстрация пропускания зашумленного входного сигнала через согласованный фильтр, чтобы определить, какой из двух сигналов присутствовал в смеси с шумом.
25. **Modulator.vi** (рис. П2.26). Этот ВП модулирует последовательность битов, отображая их в виде совокупности 16-QAM, увеличивая частоту дискретизации символов колебания и, наконец, формируя импульсы для нулевых ISI (межсимвольных помех).

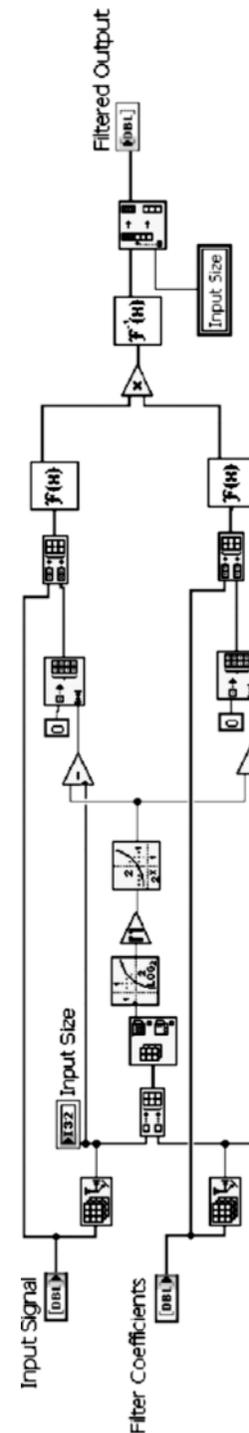


Рис. П2.12

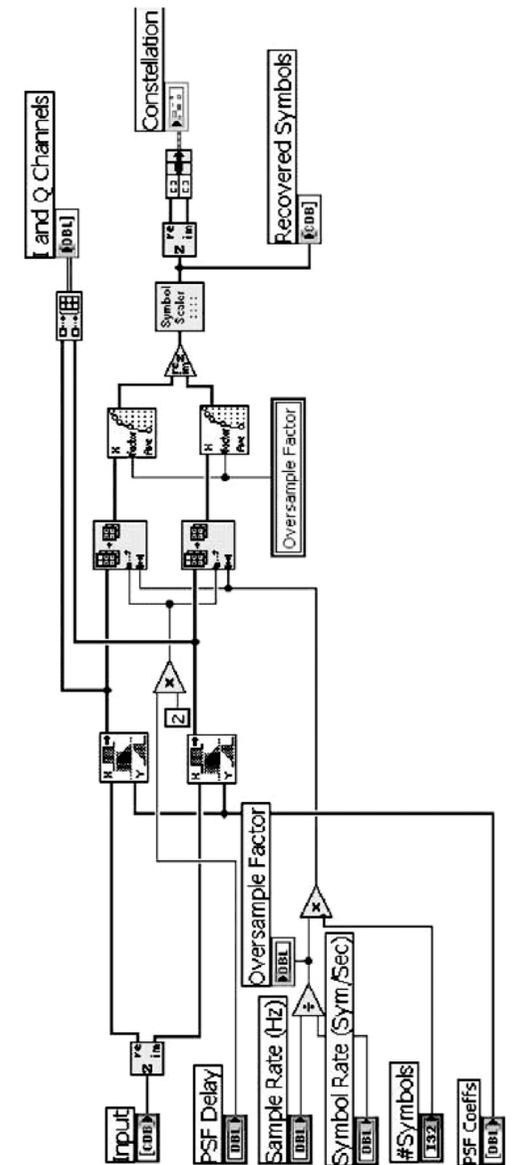
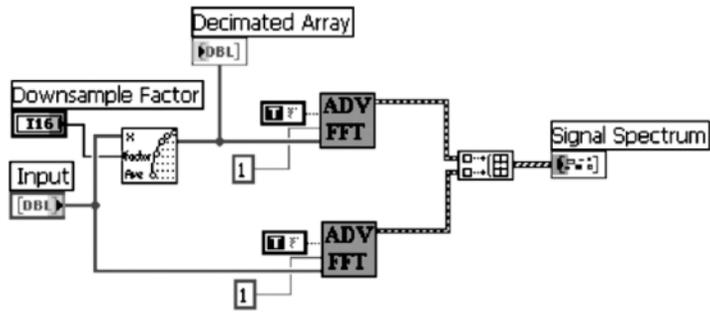


Рис. П2.13



Downsample.vi keeps only 1 out of M samples of input.

Рис. П2.14

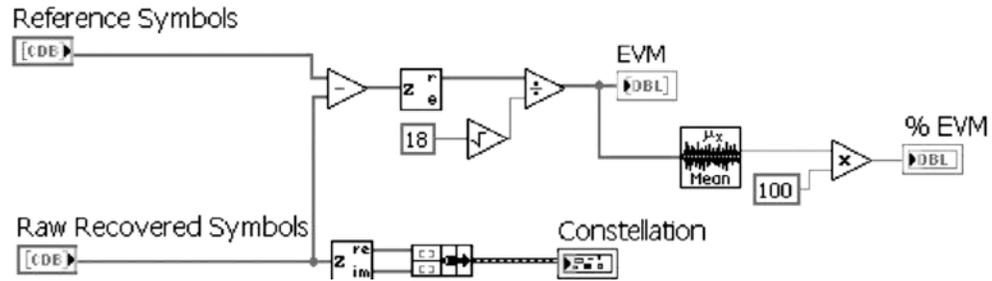


Рис. П2.15

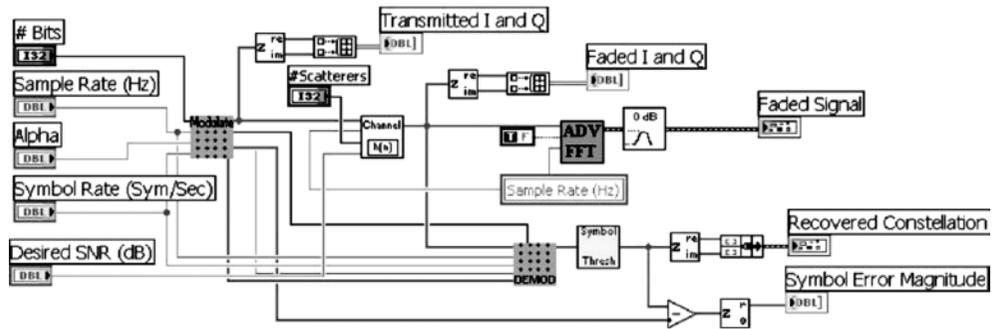


Рис. П2.16

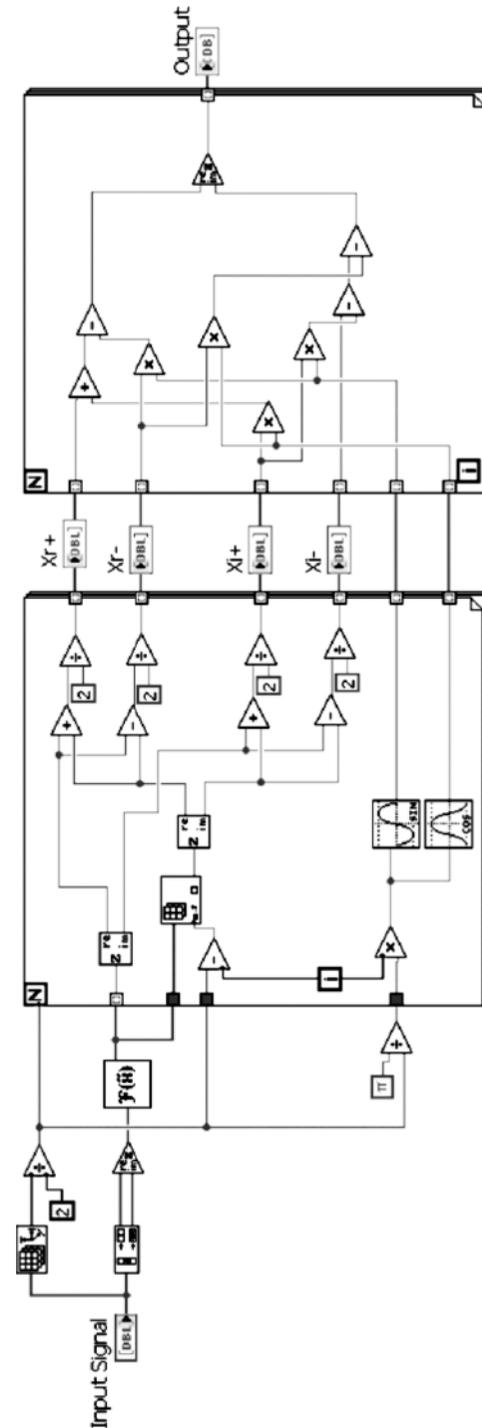


Рис. П2.17

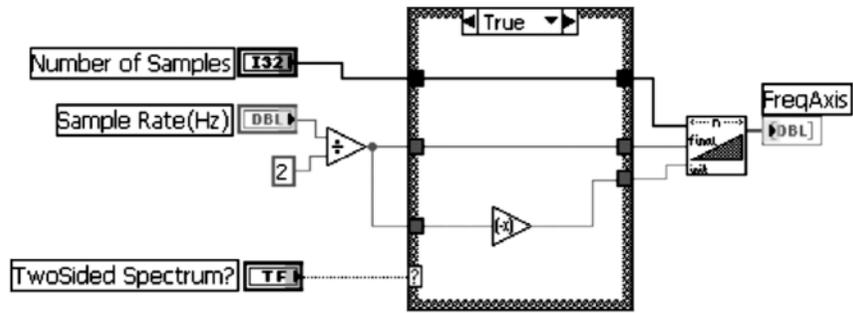


Рис. П2.18

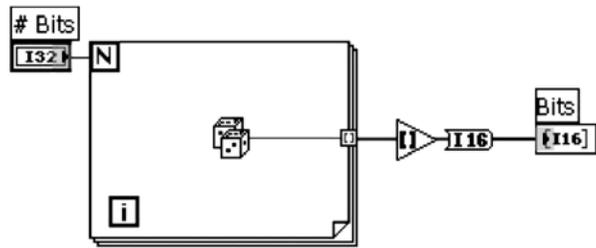


Рис. П2.19

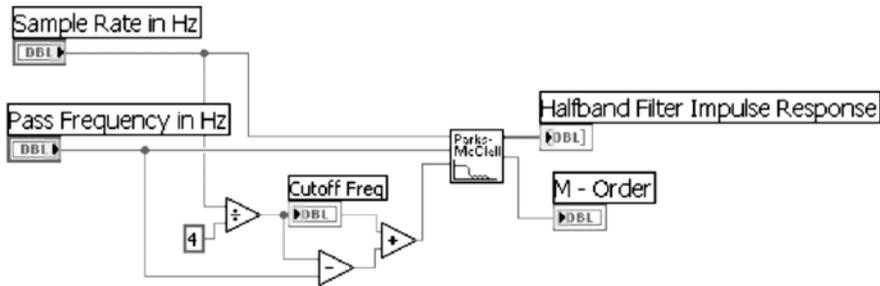


Рис. П2.20

26. **MPR.vi** (рис. П2.27). Генерирует коэффициенты ФНЧ, используя оптимальный Parks-McClellan алгоритм. Этот ВП выберет сам соответствующий порядок фильтра или пользователь может определить желательный порядок фильтра.
27. **NyquistPulse.vi** (рис. П2.28). Генерирует или поднятый косинус, или корень из поднятого косинуса формирующего фильтр. Фильтр поднятого косинуса из класса формирования импульсов по Найквисту, который обеспечивает нулевые межсимвольные помехи.

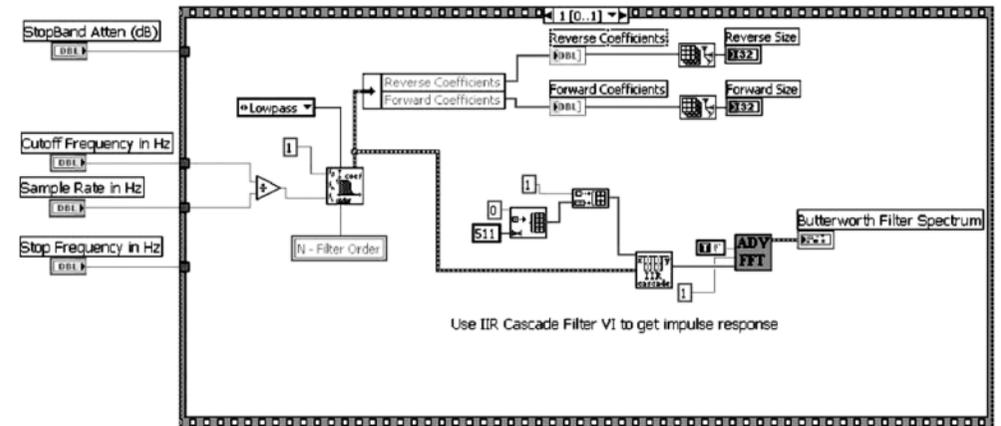
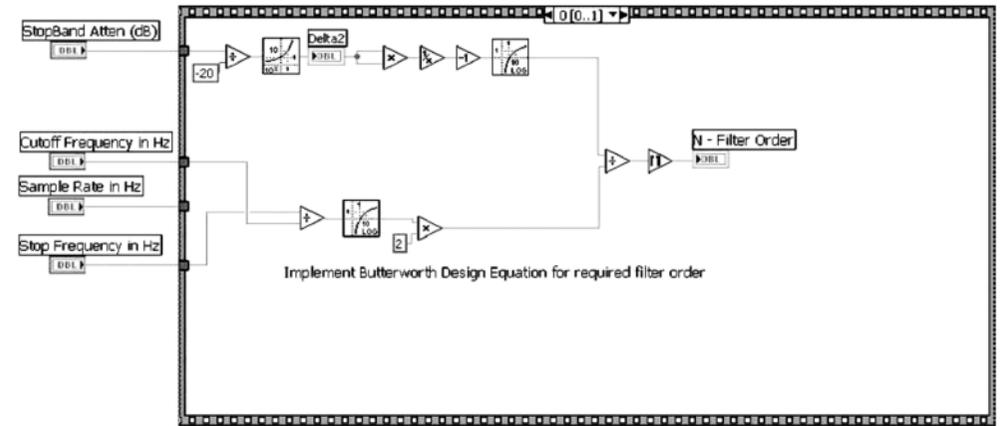


Рис. П2.21

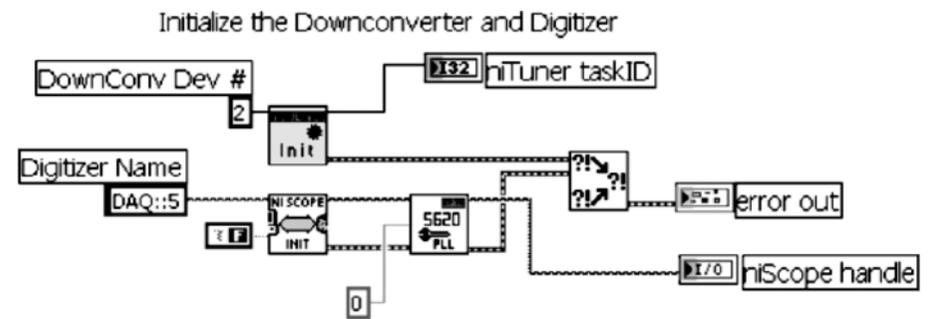


Рис. П2.22

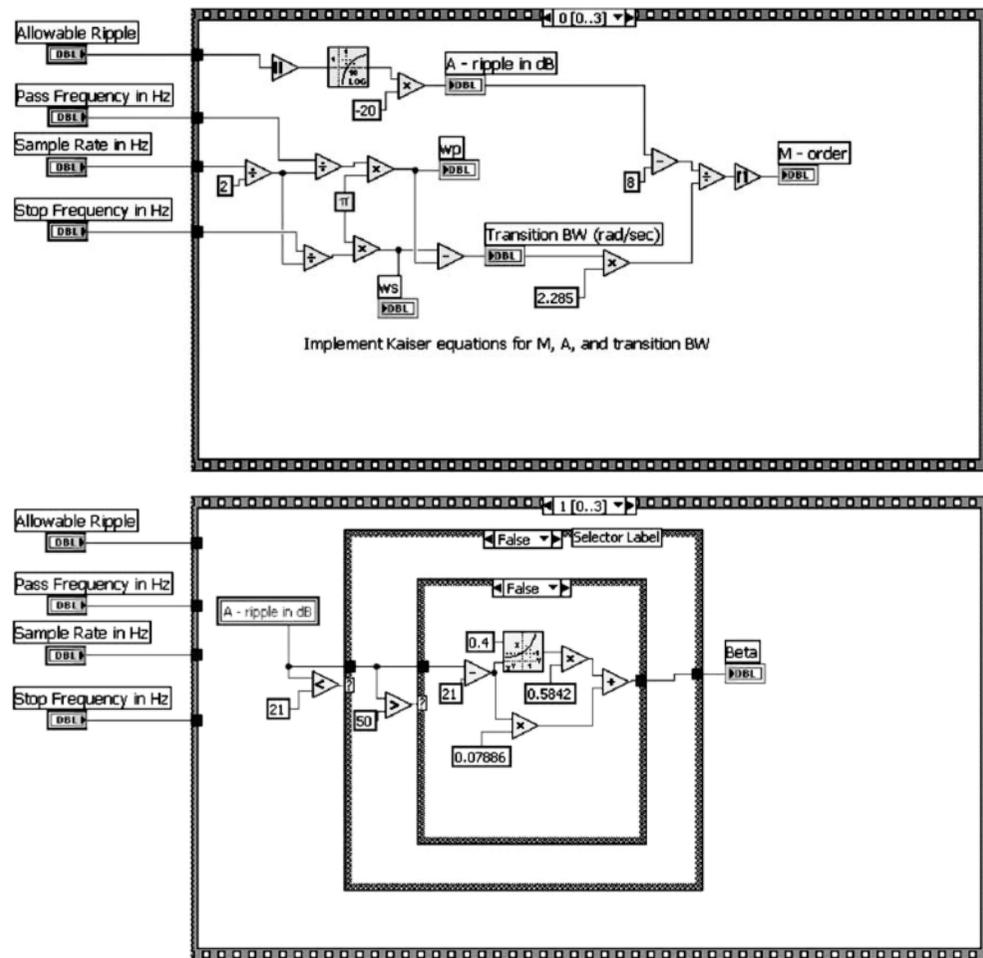


Рис. П2.23 (начало)

28. **PoleZero.vi** (рис. П2.29). Решает линейное дифференциальное уравнение с постоянными коэффициентами и составляет график полюсов и нулей относительно круга единичного радиуса.
29. **PolyphaseInterpolator.vi** (рис. П2.30). Повышает частоту дискретизации фильтра, используя эффективный полифазный подход. Фильтр разбит на отдельные блоки и повышает частоту дискретизации и фильтрацию в объединенной операции.
30. **PulseFreq.vi** (рис. П2.31). Вычисляет значения частоты для ВП импульса Найквиста.
31. **PXICapture.vi** (рис. П2.32). Использует PXI-5660-преобразователь с понижением частоты и АЦП, чтобы обеспечить непрерывный ввод и запись за указанной интервал времени.

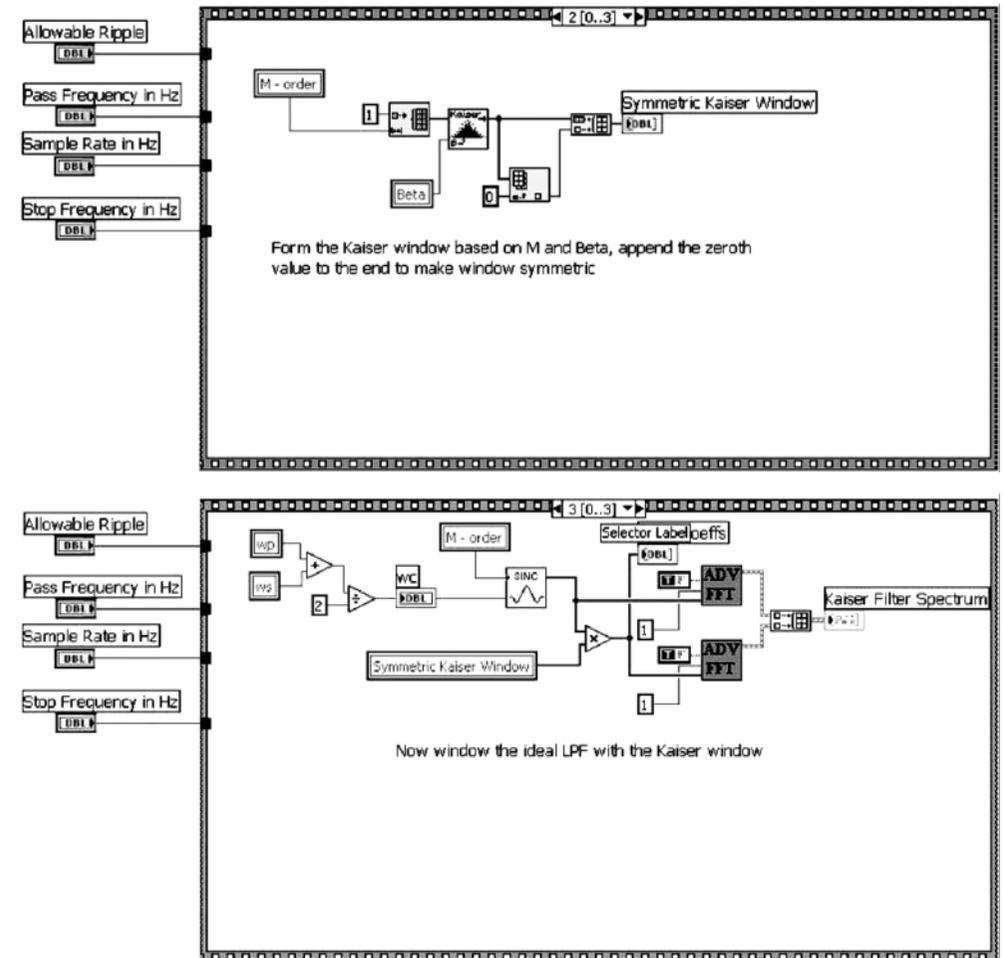


Рис. П2.23 (окончание)

32. **Rayleigh.vi** (рис. П2.33). Генерирует рэлеевскую случайную переменную, используя гауссовы реальные и мнимые компоненты и равномерно распределенную фазу.
33. **ScaleSymbols.vi** (рис. П2.34). Масштабирует восстановленные символы так, чтобы максимальная величина была КОРЕНЬ ИЗ 18 для совокупности 16-QAM.
34. **Set0dB.vi** (рис. П2.35). Устанавливает пиковое значение регистрируемого графика величины к уровню в 0 дБ.
35. **SimpleFFT.vi** (рис. П2.36). Вычисляет БПФ со встроенной функцией LabVIEW, преобразовывает величину к децибелу, затем сдвигает вывод, чтобы центрировать относительно уровня в 0 Гц.

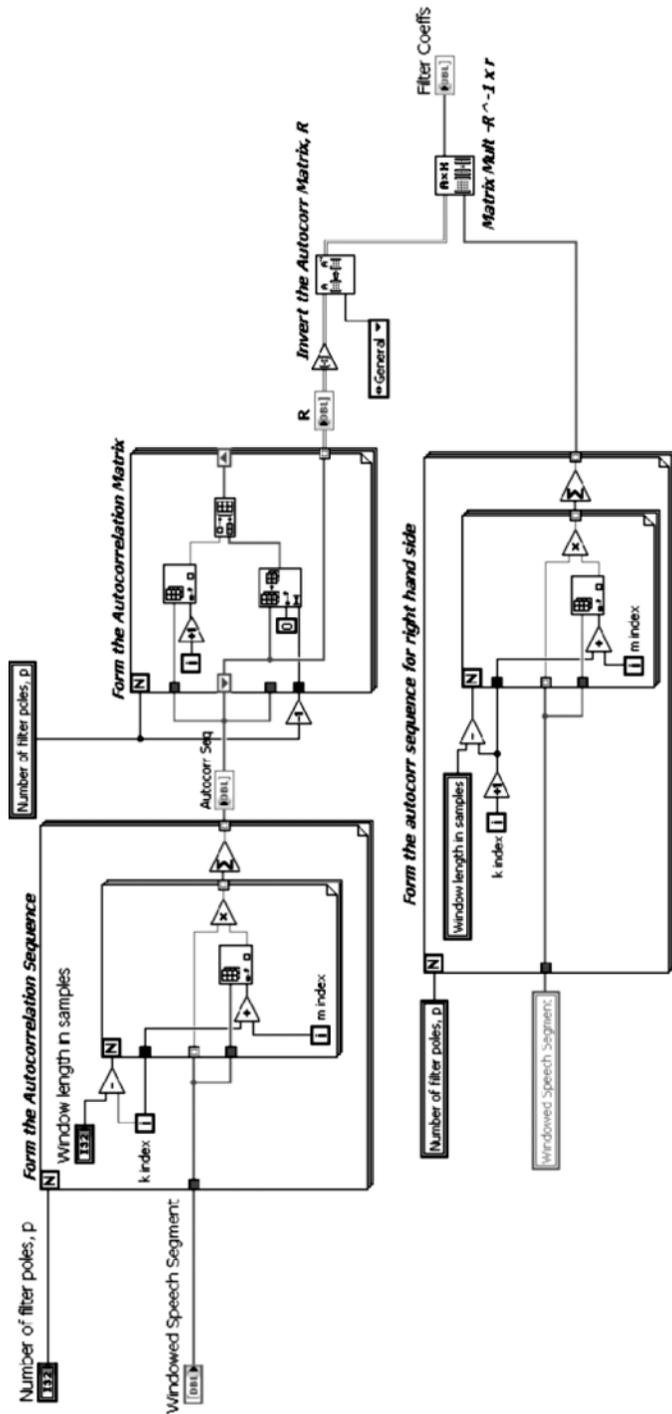


Рис. П2.24

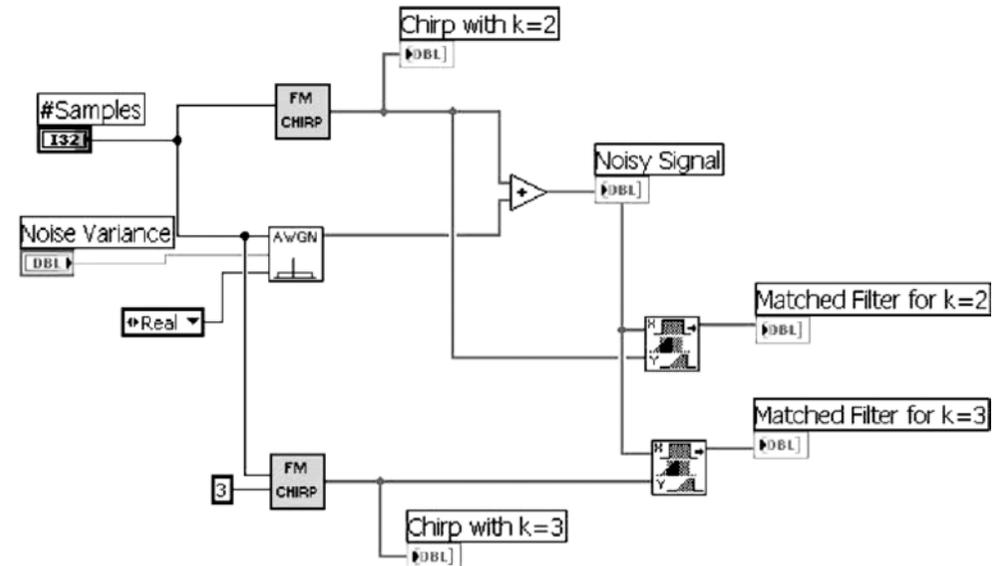


Рис. П2.25

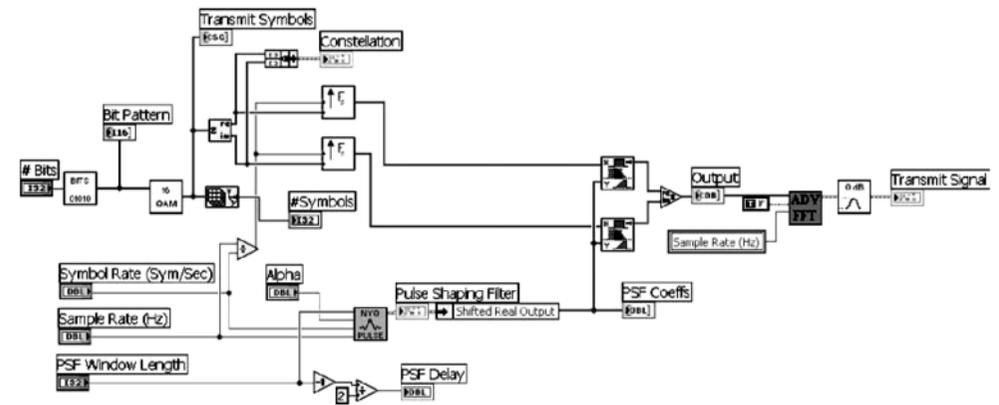


Рис. П2.26

36. **SimpleNoisySystem.vi** (рис. П2.37). Демонстрационная система связи, которая вызывает модулятор, добавляет AWGN-шум и затем детектирует. Восстановленная совокупность символов показывает некоторый разброс в результате воздействия шума.
37. **SincFcn.vi** (рис. П2.38). Вычисляет дискретную sinc-форму колебания с заданной длиной и частотой среза. Срез используется потому, что функция sinc формирует пару Фурье с идеальным ФНЧ.

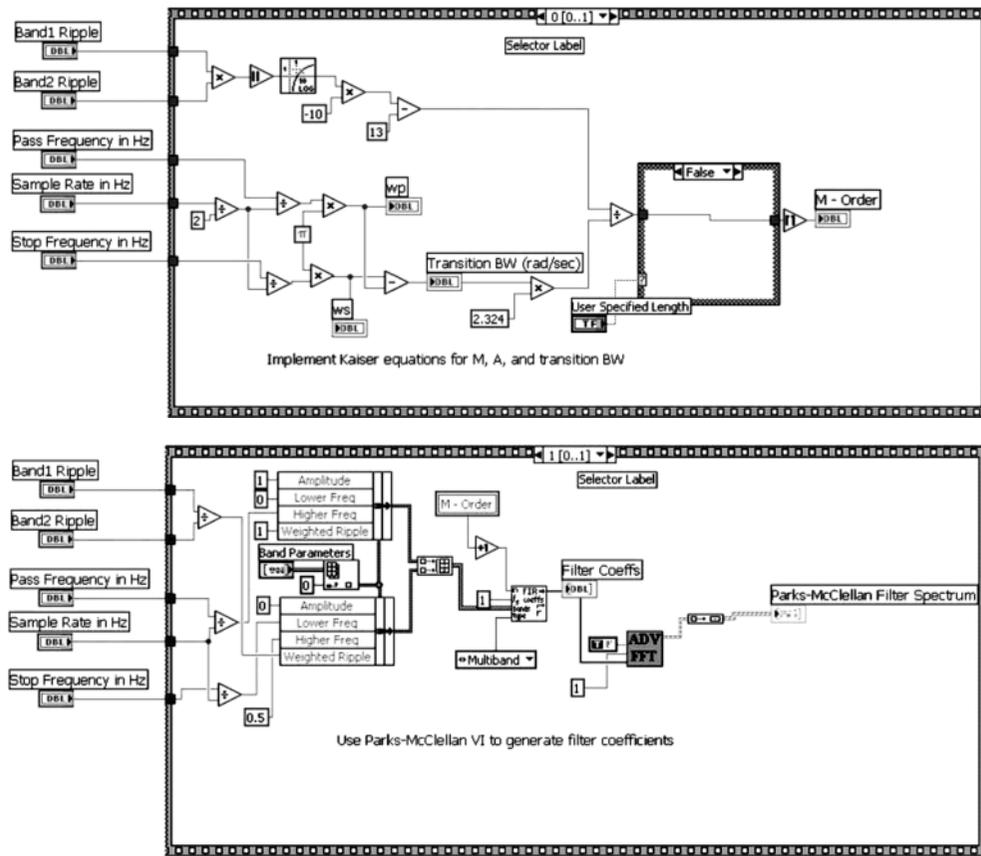


Рис. П2.27

38. **SoundCardCapture.vi** (рис. П2.39). Использует звуковую карту PC, чтобы оцифровать аналоговый ввод. Выборки могут быть 8 или 16 битов, моно или стерео.
39. **SymbolDecoder.vi** (рис. П2.40). Простой ВП, чтобы не отображать восстановленные символы назад в последовательность битов.
40. **SymbolMapper.vi** (рис. П2.41). Исполняет отображение малой последовательности к точкам на совокупности 16-QAM.
41. **SymbolRateAxis.vi** (рис. П2.42). Формирует ось частоты для показа фильтра формы импульса. Значения частоты нормализованы к частоте дискретизации.
42. **TrellisCodedModulator.vi** (рис. П2.43). Исполняет кодово-матричную модуляцию на указанном числе битов. ВП TrellisStateMachine.vi вызывается для вычисления текущего индекса символа выхода так же, как и для следующего состояния.

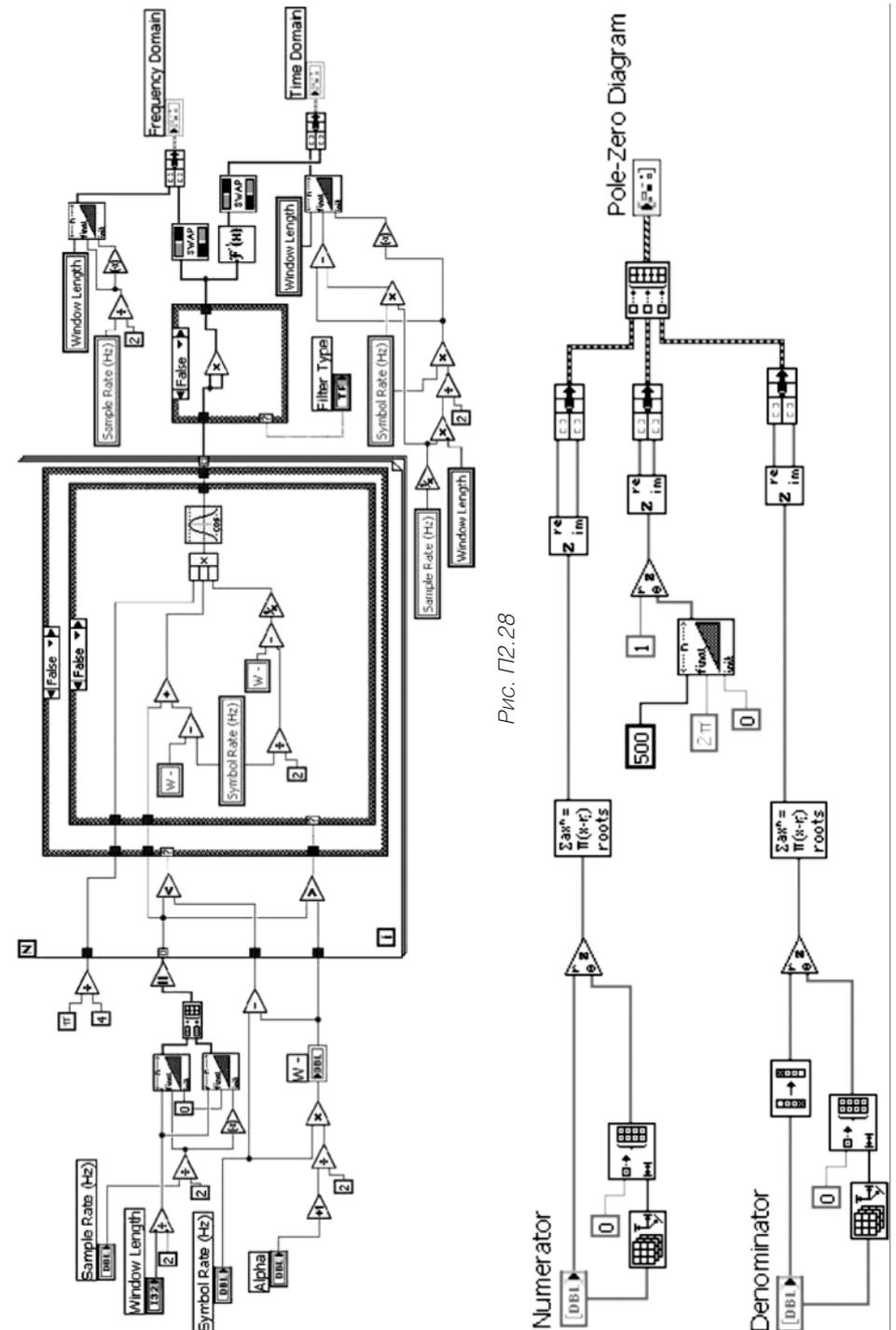


Рис. П2.28

Рис. П2.29

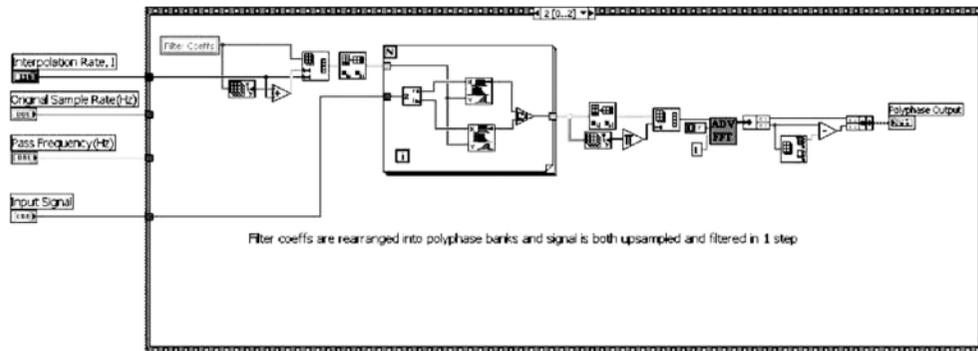
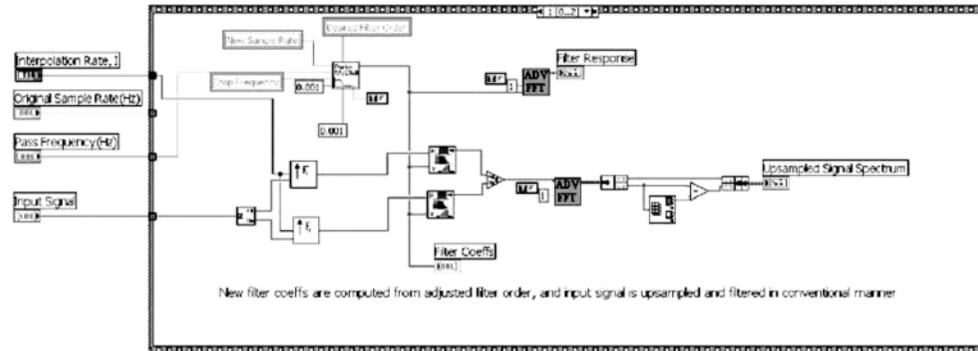
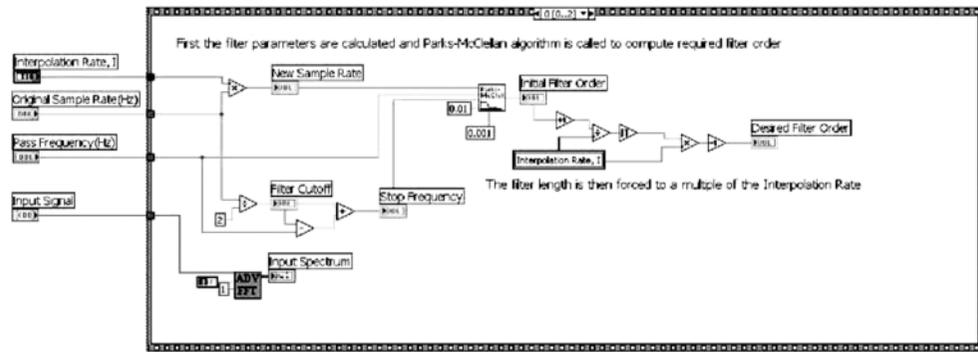


Рис. П2.30

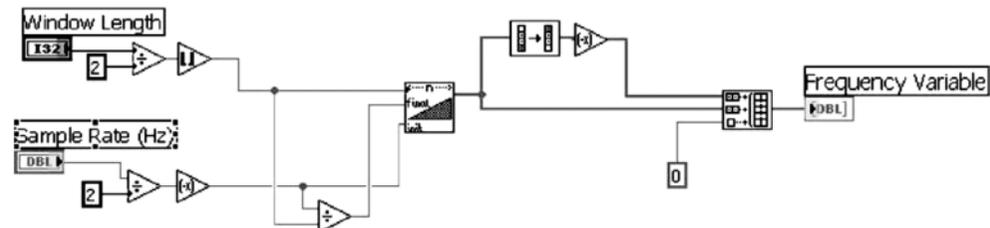


Рис. П2.31

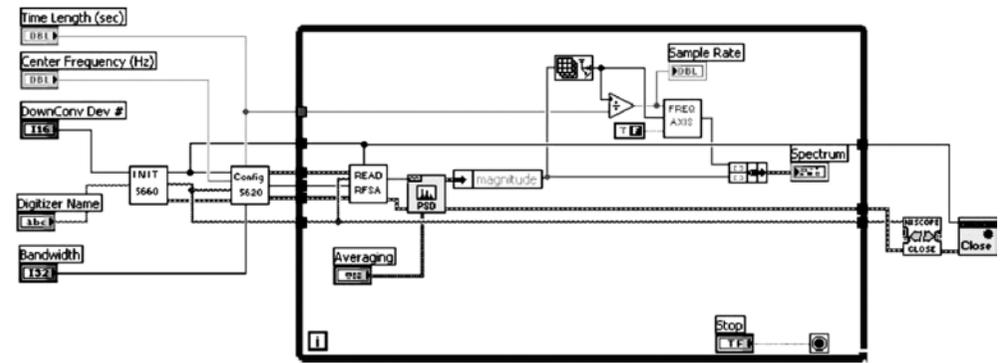
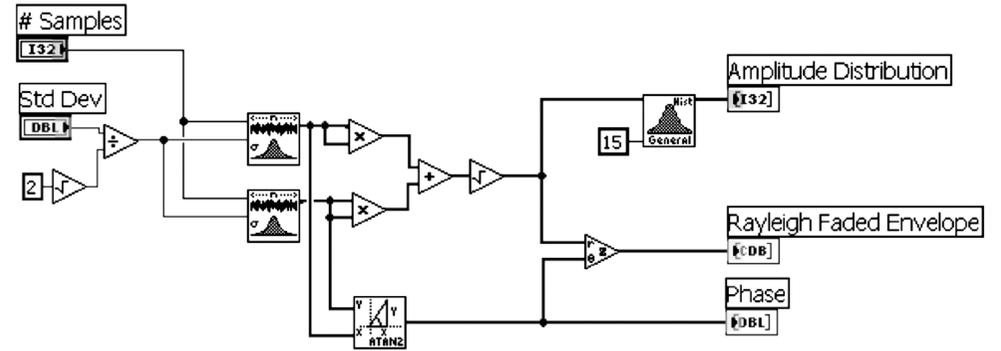


Рис. П2.32



Phase is uniformly distributed from  $-\pi$  to  $\pi$

Рис. П2.33

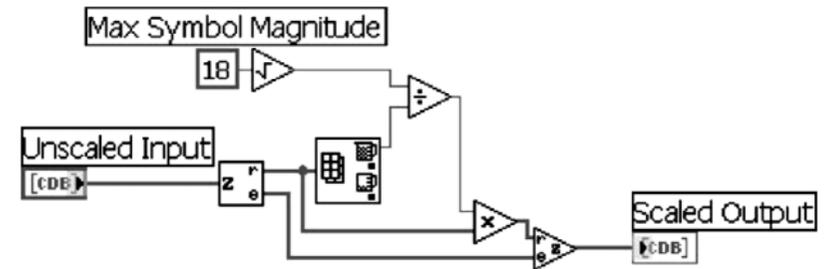
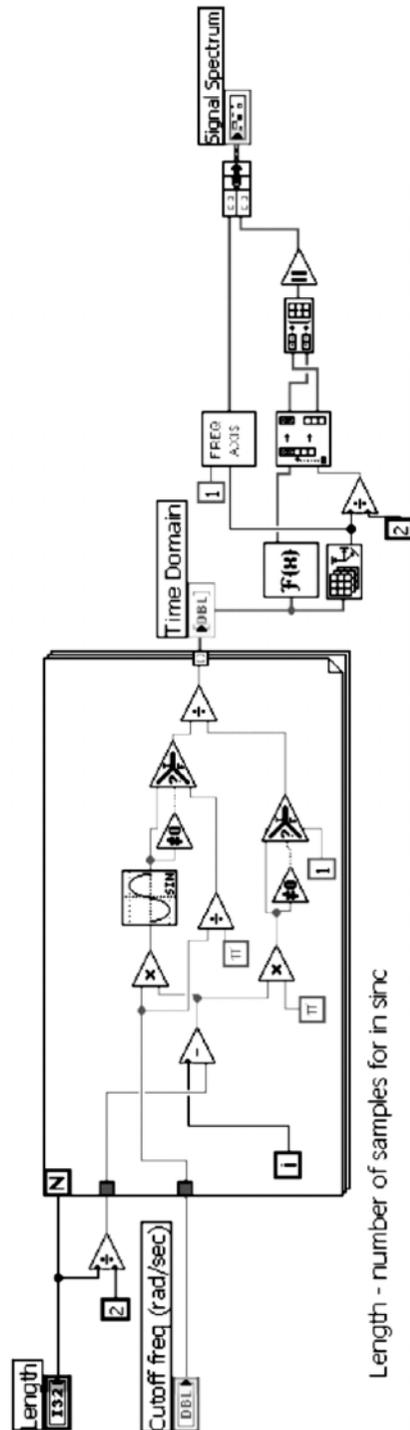


Рис. П2.34





Length - number of samples for in sinc

$$\text{Cutoff freq (rad/sec)} - \text{cutoff} = (\text{pass freq} + \text{stop freq}) / 2$$

Рис. П2.38

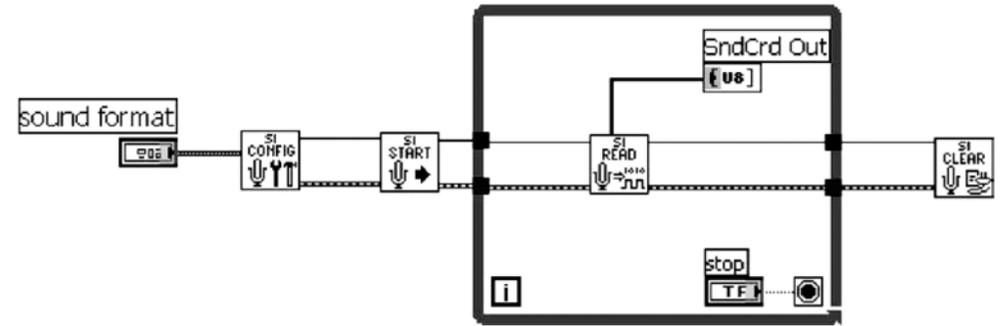


Рис. П2.39

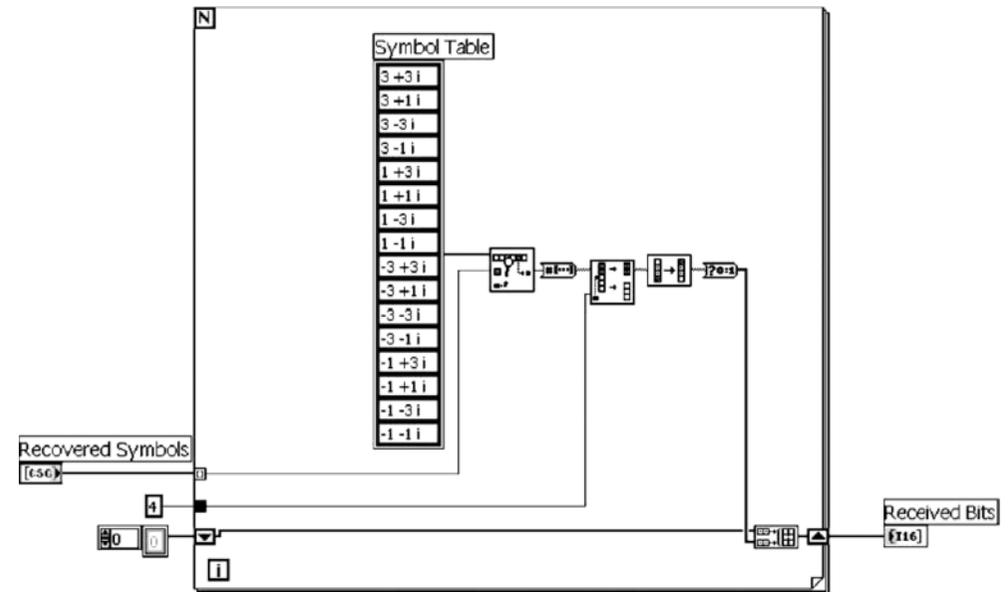


Рис. П2.40

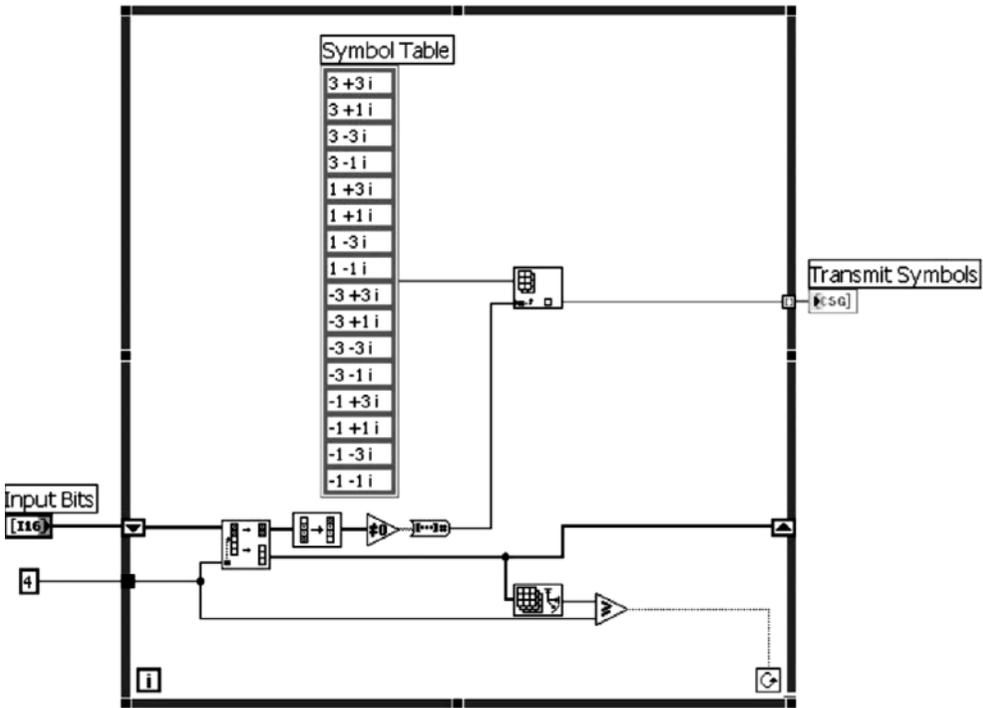


Рис. П2.41

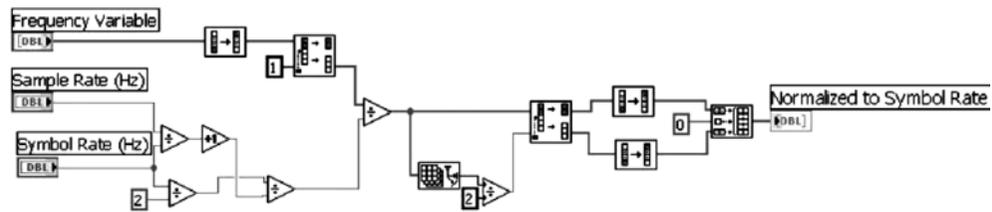


Рис. П2.42

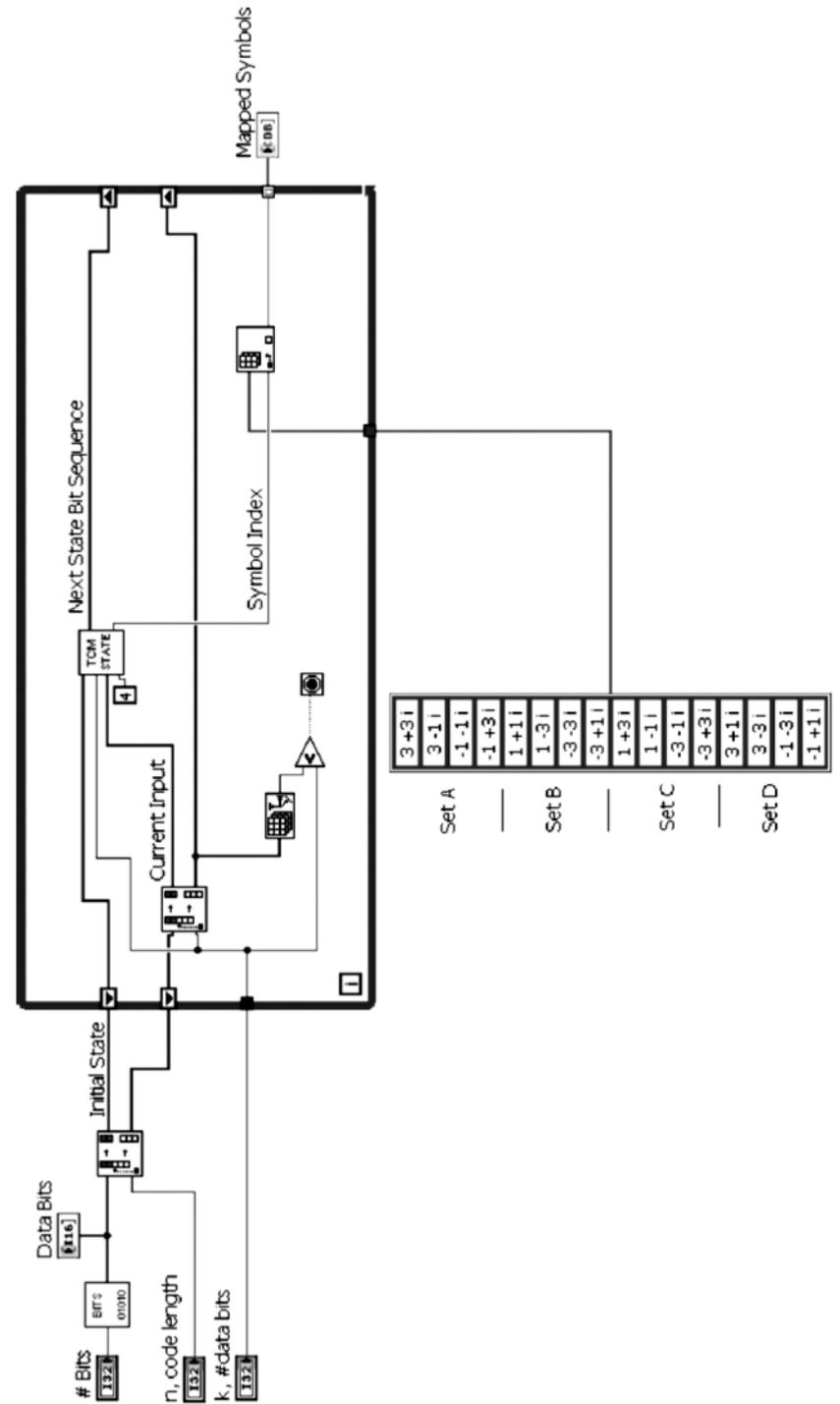


Рис. П2.43

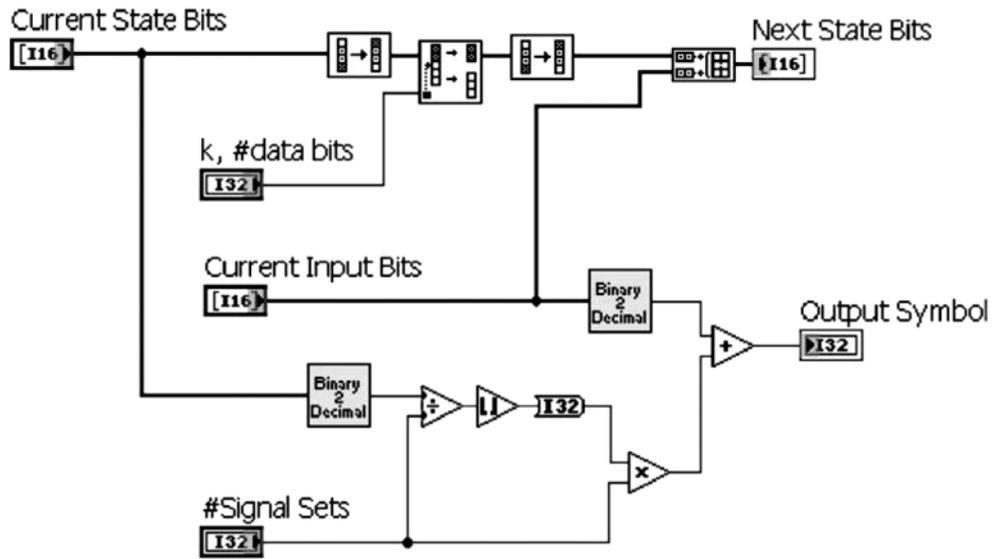
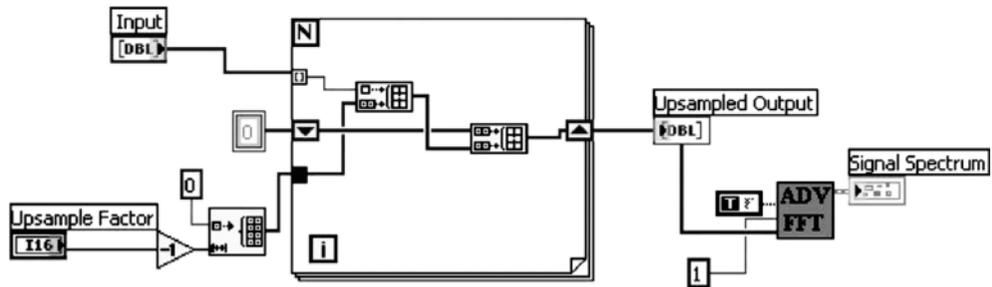


Рис. П2.44



Upsample.vi stuffs zeros between each sample of the input signal.

Рис. П2.45

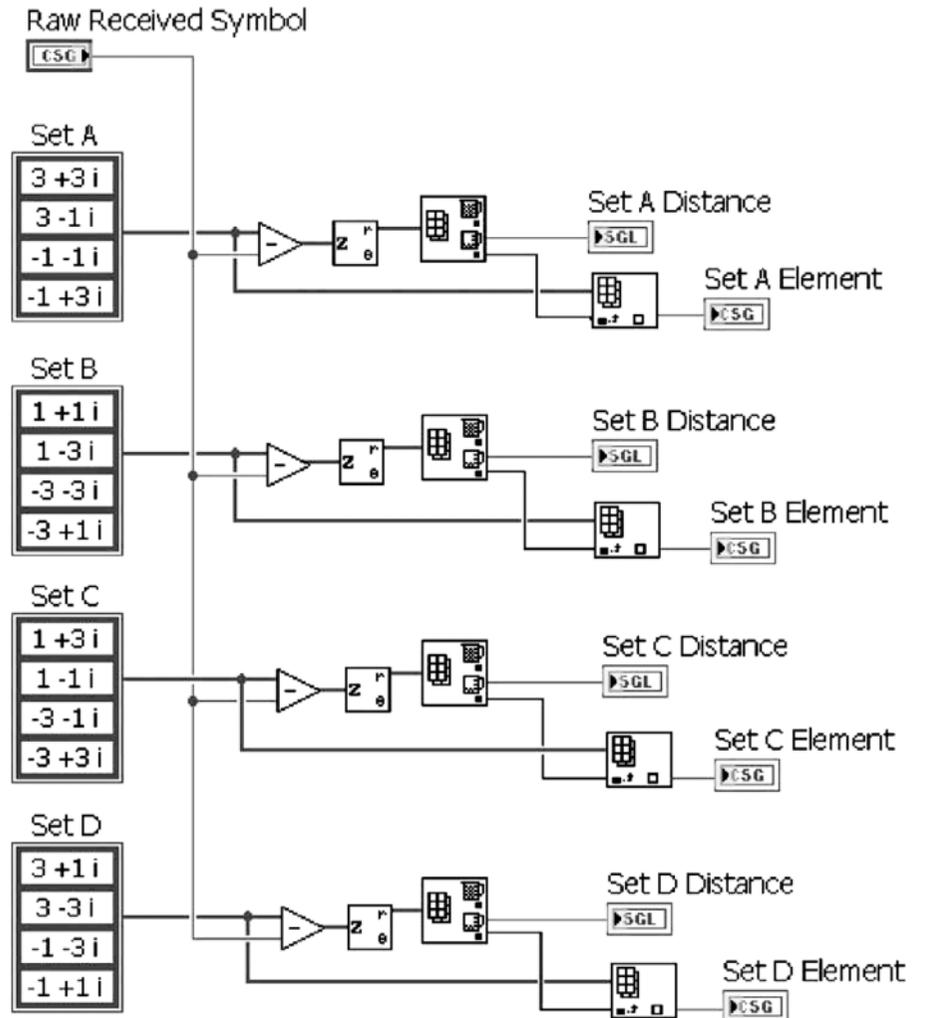


Рис. П2.46

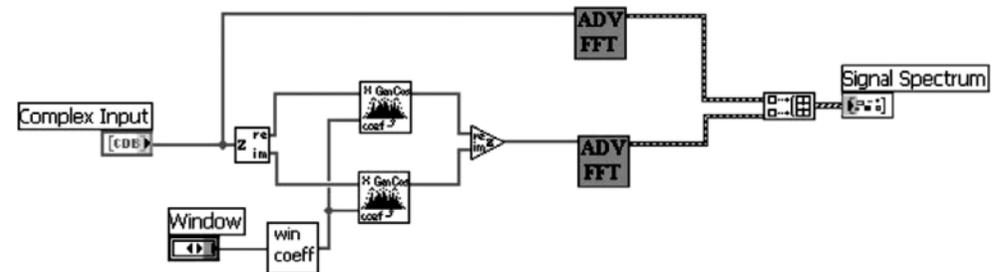


Рис. П2.47

# Приложение Лабораторная работа Прохождение случайных колебаний через линейные цепи и исследование явления нормализации



## Цель работы

Изучение методов анализа линейных цепей при случайных воздействиях. Определение изменений реализаций, спектральных плотностей, корреляционных функций и плотностей вероятностей стационарных случайных колебаний после прохождения их через линейные цепи. Исследование эффекта нормализации случайных колебаний узкополосными цепями.

## Теоретическая часть

При анализе прохождения стационарного случайного процесса (СП) через линейные электрические цепи (ЛЭЦ) (рис. ПЗ.1) будем полагать, что режим цепи установившийся, т. е. после подачи на вход цепи сигнала все переходные процессы, связанные с включением, закончились. Тогда и выходной СП будет стационарным. Рассматриваемая задача будет состоять в том, чтобы по заданной корреляционной функции входного сигнала или его спектральной плотности мощности определить  $B(\tau)$  или  $W(\omega)$  выходного сигнала.

Сначала рассмотрим решение этой задачи в частотной области. Входной СП задан своей спектральной плотностью мощности  $W_x(\omega)$ . Выходная спектральная плотность мощности  $W_y(\omega)$  определяется по формуле

$$W_y(\omega) = W_x(\omega) K^2(\omega), \quad (\text{ПЗ.1})$$

где  $K^2(\omega)$  – квадрат модуля комплексной передаточной функции цепи. Возведение в квадрат модуля основано на том, что искомая характеристика является



Рис. ПЗ.1

ся действительной функцией частоты и энергетической характеристикой выходного процесса.

Для определения связи между корреляционными функциями необходимо применить к обеим частям равенства (ПЗ.1) обратного преобразования Фурье:

$B_x(\tau) = F^{-1}[W_x(\omega)]; F^{-1}[K^2(\omega)] = B_h(\tau)$  – корреляционная функция импульсной характеристики исследуемой цепи. Таким образом, корреляционная функция выходного СП есть

$$B_y(\tau) = B_x(\tau) * B_h(\tau) = \int_{-\infty}^{\infty} B_x(t)B_h(t - \tau)dt. \quad (\text{ПЗ.2})$$

**Пример** прохождения стационарного случайного широкополосного сигнала через  $RC$ -цепь (фильтр нижних частот), представленную принципиальной электрической схемой, приведен на рис. ПЗ.2. Широкополосность понимается так, что энергетическая ширина спектра входного СП намного больше полосы пропускания цепи (рис. ПЗ.3). При таком соотношении между формой  $K^2(\omega)$  и  $W_x(\omega)$  можно не рассматривать ход характеристики  $W_x(\omega)$  в области верхних частот.

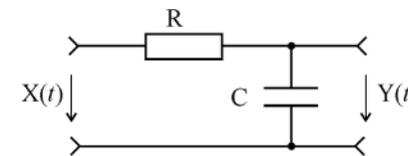


Рис. ПЗ.2

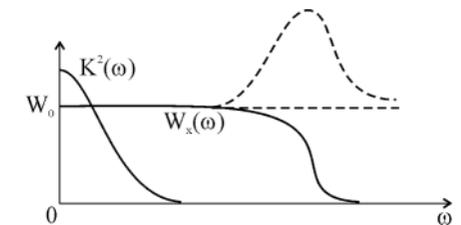


Рис. ПЗ.3

Учитывая, что в полосе частот, где  $K^2(\omega)$  существенно отличается от нуля, спектральная плотность мощности входного сигнала равномерна, можно без существенной погрешности входной сигнал аппроксимировать белым шумом, т. е. положить  $W_x(\omega) = W_0 = \text{const}$ . Такое предположение существенно упрощает анализ. Тогда  $W_y(\omega) = W_0 K^2(\omega)$ . Для заданной цепи  $K^2(\omega) = 1/[1 + (\omega RC)^2]$ , тогда  $W_y(\omega) = W_0/[1 + (\omega RC)^2]$ .

Определим энергетическую ширину спектра выходного сигнала: Мощность выходного СП  $P_y = s_y^2 = (2\pi)^{-1} \int_{-\infty}^{\infty} W_y(\omega)d\omega = W_0/(2RC)$ , тогда  $\Delta\omega_s = (W_0)^{-1} \int_0^{\infty} W_y(\omega)d\omega = \pi/(2RC)$ . На рис. ПЗ.4 показаны корреляционная функция выходного СП и его спектральная плотность мощности.

*Энергетической (шумовой) полосой пропускания* электрической цепи называется полоса частот, численно совпадающая с энергетической шириной спектральной плотности мощности сигнала на выходе цепи при воздействии на вход цепи белого шума. В данном случае  $\Delta\omega_s = \pi/(2RC)$ . Сравним ее с полосой пропуска-

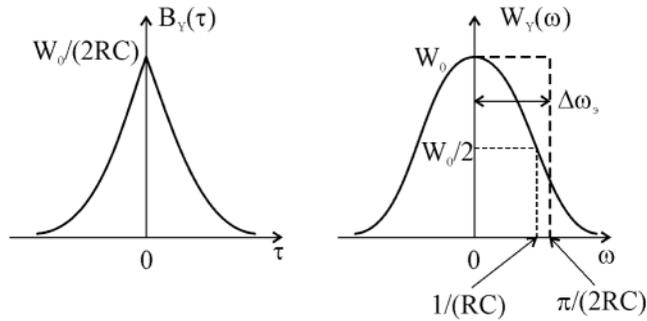


Рис. ПЗ.4

ния  $\omega_{rp}$  этой же цепи на уровне 0,707. Так как  $\omega_{rp} = 1/(RC)$ , то  $\Delta\omega_3 = \pi/2\omega_{rp}$ , т. е.  $\Delta\omega_3$  в  $\pi/2$  раз больше  $\omega_{rp}$ .

Определим корреляционную функцию сигнала на выходе RC-цепи при воздействии на ее вход белого шума.

Так как выходная спектральная плотность мощности уже определена, то можно вычислить искомую функцию обратным преобразованием Фурье. Но в рассматриваемом случае проще анализ выполнить во временной области, т. е.  $B_y(\tau) = B_x(\tau) * B_h(\tau)$ , но так как  $B_x(\tau) = W_0\delta(\tau)$ , то  $B_y(\tau) = W_0B_h(\tau)$  (учитывая фильтрующее свойство дельта-функции).

Таким образом, при воздействии на вход цепи белого шума, корреляционная функция выходного сигнала совпадает с точностью до постоянного множителя с корреляционной функцией импульсной характеристики рассматриваемой цепи.

Так как  $h(t) = 1/(RC) \exp[-t/(RC)]$ ,  $t \geq 0$ , то  $B_h(\tau) = \int_{-\infty}^{\infty} h(t)h(t - \tau)dt = 1/(2RC) \exp[-|\tau|/(RC)]$ ,  $-\infty < \tau < \infty$ .

На рис. ПЗ.5 представлены корреляционные функции (рис. ПЗ.5, а) и спектральные плотности мощности (рис. ПЗ.5, б) для двух значений постоянной времени заданной цепи  $(RC)_1 < (RC)_2$ . Дисперсия выходного СП  $\sigma_y^2 = B_y(0) = W_0/(2RC)$ .

Площадь под кривой  $B_y(\tau)$  равна значению спектральной плотности мощности при  $\omega = 0$ , т. е.  $W_0$ . Из сравнения графиков на рис. ПЗ.5 следует, что с уменьшением полосы пропускания цепи начальное (максимальное) значение корреляционной функции  $B_y(0)$  уменьшается, что связано с уменьшением мощности выходного сигнала (дисперсии), и корреляционная функция изменяется медленнее с увеличением RC заданной цепи.

**Пример** анализа прохождения белого шума через колебательный контур приведен на рис. ПЗ.6. Чтобы придать этой задаче физический смысл, сводим задачу, как и предыдущую, к аппроксимации входного сигнала белым шумом. Чтобы использовать такой прием, входной сигнал должен иметь спектральную плотность мощности, неизменную в пределах практически значимых значений ординат АЧХ цепи. Тогда  $W_x(\omega)$  можно считать равной  $W_0$ , а входной СП – белым шумом (рис. ПЗ.7).

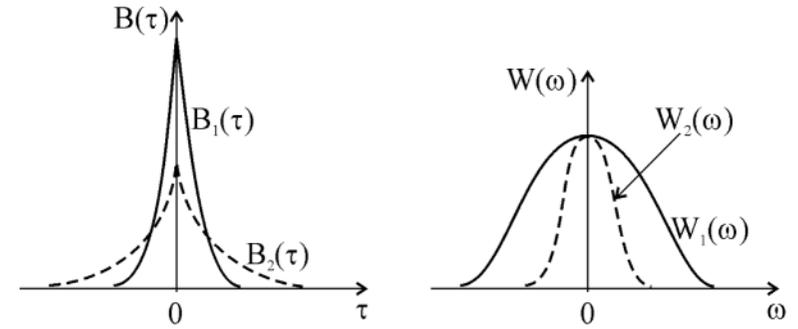


Рис. ПЗ.5

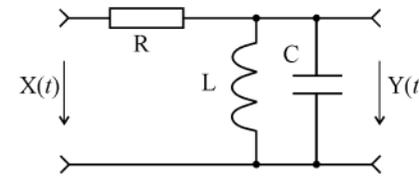


Рис. ПЗ.6

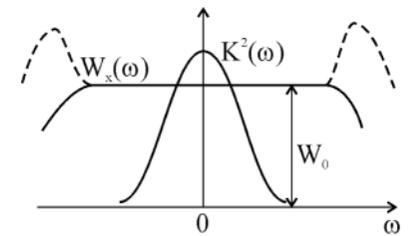


Рис. ПЗ.7

Передаточная функция такой цепи  $K(j\omega) = K_p/[1 + j2(\omega - \omega_p)/(\omega_p Q_3)]$ ;  $0 < \omega < \infty$ , где  $K_p$  – коэффициент передачи цепи при резонансной частоте  $\omega_p$ , т. е.  $K_p = R_{zp}/(R_{zp} + R)$ ;  $Q_3 = Q/(1 + R_{zp}/R)$  – добротность шунтированного нагрузкой R колебательного контура, его постоянная времени  $\tau_k = 2Q_3/\omega_p = 2/(\Delta\omega)_{0,7}$ , т. е. обратная половине полосы пропускания контура на уровне 0,707. Квадрат модуля передаточной функции  $K^2(\omega) = K_p^2/[1 + (\omega - \omega_p)^2\tau_k^2]$ . Найдем дисперсию процесса на выходе цепи  $\sigma_y^2 = W_0 K_p^2 / (\pi\tau_k) (\pi/2 + \arctg 2Q_3) \gg W_0 K_p^2 / (2\tau_k)$  при  $Q_3 \gg 1$ .

Оценим энергетическую полосу пропускания колебательного контура (рис. ПЗ.8)  $\Delta\omega_3 = (W_0 K_p^2)^{-1} \int_0^{\infty} W_y(\omega) d\omega \approx \pi/\tau_k$ . Сравним с полосой пропускания по уровню 0,707

(–3 дБ). Так как  $\tau_k = 2Q_3/\omega_p$ , то  $\Delta\omega_3 = \pi/2 (\Delta\omega)_{0,7}$ . Вычислим корреляционную функцию выходного процесса:  $B_y(\tau) = W_0 K_p^2 / (2\tau_k) \exp(-|\tau|/\tau_k) \cos \omega_p \tau$ ;  $-\infty < \tau < \infty$  (рис. ПЗ.9). Если рассматривать анализ контуров с разными добротностями, то можно увидеть различия в реализациях выходных процессов: рис. ПЗ.10 при добротности  $Q_1$  и рисунок ПЗ.11 при добротности  $Q_2$ .

Сравнительный анализ реализаций показывает, что увеличение добротности приводит к снижению полосы пропускания контура, а значит – к снижению средней скорости изменения огибающей во времени (можно сравнить с влиянием на огибающую АМК снижения частоты модулирующего колебания).

**Расчет плотности вероятности стационарного случайного сигнала на выходе линейной цепи.** Если для многих электрических цепей в установившемся

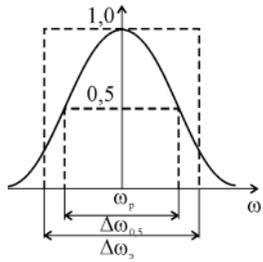


Рис. ПЗ.8

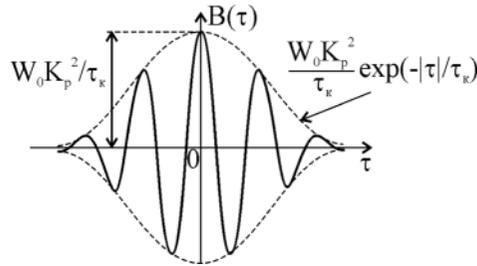


Рис. ПЗ.9



Рис. ПЗ.10



Рис. ПЗ.11

режиме просто рассчитать энергетический спектр и корреляционную функцию, то задача расчета плотности вероятности в общем случае не имеет общего решения. Расчет плотности вероятности на выходе такой цепи является сложной задачей, не имеющей аналитического решения. Трудности анализа обусловлены тем, что мгновенные значения сигнала на выходе линейной цепи зависят не только от мгновенных значений входного сигнала в данный момент времени, но и от значений сигнала в предыдущие моменты (поскольку цепь обладает инерционностью, вызванной наличием катушек индуктивности и конденсаторов в цепи). Однако имеет место единственный случай, когда законы плотности вероятности на входе и выходе цепи совпадают. Это случай, когда входной сигнал имеет нормальный закон распределения. Основным свойством нормального закона является то, что при прохождении сигнала с нормальной плотностью вероятности сам вид закона не изменяется, а меняются лишь его параметры, т. е. математическое ожидание, дисперсия, корреляционная функция. Здесь можно выявить аналогию с гармоническим колебанием в линейной цепи.

**Прохождение нормального стационарного СП через линейную электрическую цепь.** Задан входной СП  $X(t)$ , у которого плотность вероятности в каж-

дый момент времени  $f(x) = (\sigma_x^2 \sqrt{2\pi})^{-1} \exp[-x^2/(2\sigma_x^2)]$ ,  $-\infty < x < \infty$ , математическое ожидание считаем равным нулю. Поэтому средний квадрат такого СП равен дисперсии. Корреляционная функция  $B_x(\tau)$  и, следовательно,  $W_x(\omega)$  известны. Вычислить плотность вероятности сигнала на выходе цепи, заданной или импульсной характеристикой  $h(t)$ , или передаточной функцией  $K(j\omega)$ . Учитывая, что при прохождении случайного сигнала через линейную цепь нормальный закон распределения не изменяется, можно записать:  $f(y) = (\sigma_y^2 \sqrt{2\pi})^{-1} \exp[-y^2/(2\sigma_y^2)]$ , т. е. сама форма закона известна. Необходимо определить дисперсию  $\sigma_y^2$ , а она связана с мощностью процесса:  $B_y(0) = \sigma_y^2 = (2\pi)^{-1} \int_{-\infty}^{\infty} W_y(\omega) d\omega$ . Чтобы найти либо  $B_y(\tau)$ , либо

$W_y(\omega)$ , требуется знать  $B_x(\tau)$  или  $W_x(\omega)$ .

В частотной области  $W_y(\omega) = W_x(\omega)K^2(\omega)$ , тогда  $\sigma_y^2 = (2\pi)^{-1} \int_{-\infty}^{\infty} W_x(\omega) K^2(\omega) d\omega$ .

Во временной области  $B_y(\tau) = B_x(\tau) * B_h(\tau)$ . Полагая  $\tau = 0$  и учитывая, что  $B_h(-\tau) = B_h(\tau)$ , можно записать  $\sigma_y^2 = \int_{-\infty}^{\infty} B_x(\tau) B_h(\tau) d\tau$ .

**Нормализация случайных сигналов в узкополосных электрических цепях.** Допустим, есть устройство, имеющее структурную схему, показанную на рис. ПЗ.12.

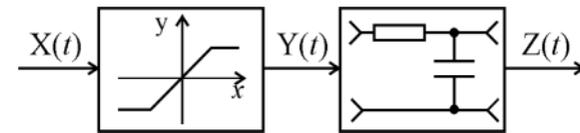


Рис. ПЗ.12

Если СП с нормальным законом подвергнуть какому-либо нелинейному преобразованию (например, двухстороннему ограничению), то его закон распределения изменится (рис. ПЗ.13).

На выходе узкополосной цепи опять получим сигнал с нормальным распределением и этот закон тем ближе будет к нормальному, чем уже полоса пропускания

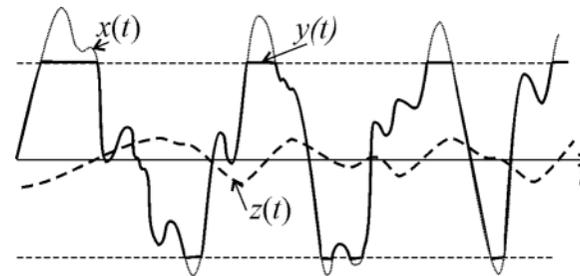


Рис. ПЗ.13

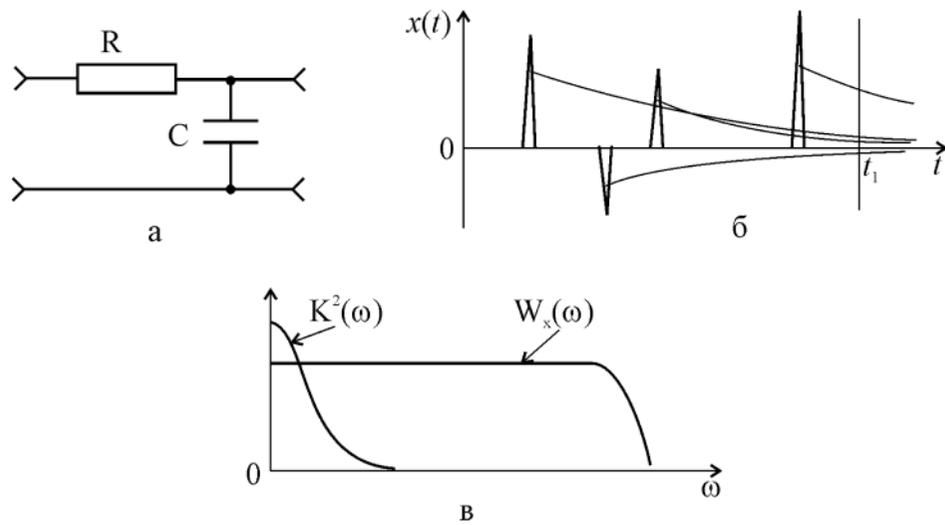


Рис. ПЗ.14

используемой линейной цепи относительно полосы входного процесса. Преобразование случайного процесса в линейной цепи поясняется с помощью рис. ПЗ.14, где представлены: линейная цепь (рис. ПЗ.14, а); реализация входного СП в виде коротких импульсов со случайными пиковыми значениями и временем появления, а также отклики заданной цепи на каждый импульс (рис. ПЗ.14, б); квадрат модуля АЧХ цепи и спектр входного процесса (рис. ПЗ.14, в).

Входной СП  $X(t)$  значительно отличается от нормального. Выходной СП  $Y(t)$  в момент времени  $t_1$  представляет собой сумму откликов на каждый из предшествующих во времени импульсов. Так как эта последовательность случайна, то мгновенные значения откликов в момент времени  $t_1$  от каждого импульса будут случайной величиной. Число слагаемых равно числу таких откликов. Если рассматривать установившийся режим, то число слагаемых теоретически бесконечно, при этом каждая случайная величина может иметь произвольный закон распределения. В то же время закон распределения суммы этих случайных величин определяется центральной предельной теоремой теории вероятности. В соответствии с этой теоремой плотность вероятности суммы независимых СВ с увеличением числа слагаемых стремится к нормальному закону, независимо от того, какой закон распределения имеет каждая из слагаемых величин (но при этом вклад в сумму всех СВ должен быть одинаков).

## Литература для углубленного изучения

1. Гоноровский И.С. Радиотехнические цепи и сигналы. – М.: Радио и связь, 1986. – 512 с.
2. Баскаков С.И. Радиотехнические цепи и сигналы. – М.: Высшая школа, 1988. – 535 с.

3. Гоноровский И.С., Демин М.П. Радиотехнические цепи и сигналы. – М.: Радио и связь, 1994. – 480 с.

4. Федосов В.П. Радиотехнические цепи и сигналы: для самостоятельного изучения. Учебное пособие. – Таганрог: Изд-во ТРТУ, 2004. – 208 с.

5. Смирнов Н.Н., Федосов В.П., Цветков Ф.А. Измерение характеристик случайных процессов: Конспект лекций. – М.: Сайнспресс, 2004. – 76 с.

6. Радиотехнические цепи и сигналы. Примеры и задачи: Учеб. пособие для вузов / Г.Г. Галустов, И.С. Гоноровский и др.; Под ред. И.С. Гоноровского – М.: Радио и связь, 1989.

## Вопросы для самоконтроля

1. На основе каких свойств преобразования Фурье определяется связь преобразований спектральных и корреляционных характеристик СП во временном и спектральном методах?
2. Запишите основные формулы во временном и спектральном методах.
3. Почему в спектральном методе используется квадрат модуля передаточной функции цепи?
4. Как объяснить явление нормализации СП узкополосными электрическими цепями?
5. Как по спектральной плотности узкополосного СП определить спектральную плотность его огибающей?
6. Каковы соотношения между корреляционными функциями узкополосного СП и его огибающей?
7. Объяснить влияние параметров цепи на вид реализации выходного случайного колебания.
8. В чем отличие выходных случайных процессов для различных параметров линейной цепи (для ФНЧ и колебательного контура)?
9. Если верхняя граничная частота ФНЧ равна половине полосы пропускания колебательного контура, то в чем заключается общность реализаций на их выходах?
10. При каком условии процесс можно назвать узкополосным?
11. Чем принципиально отличаются корреляционные функции узкополосного и широкополосного случайных процессов (при одинаковой энергетической полосе)?

## Описание средств для проведения исследования

Лабораторная работа выполняется с помощью программы, написанной в среде *LabVIEW* 8, блок-диаграмма которой представлена на рис. ПЗ.15.

Лицевые панели виртуальной измерительной системы сформированы отдельно для входного случайного процесса, для исследования процесса на выходе линейной цепи и для исследования нормализации и представлены, соответственно, на рис. ПЗ.16, рис. ПЗ.17 и рис. ПЗ.18.

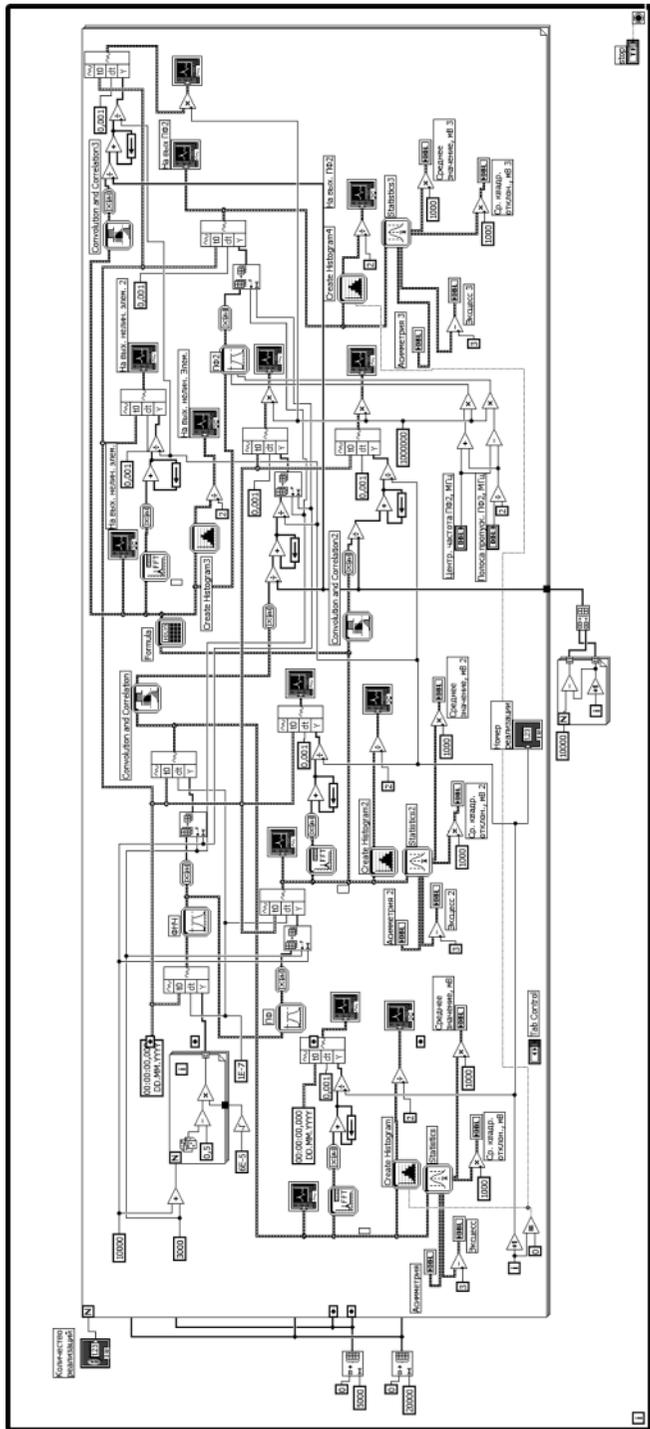


Рис. ПЗ.15

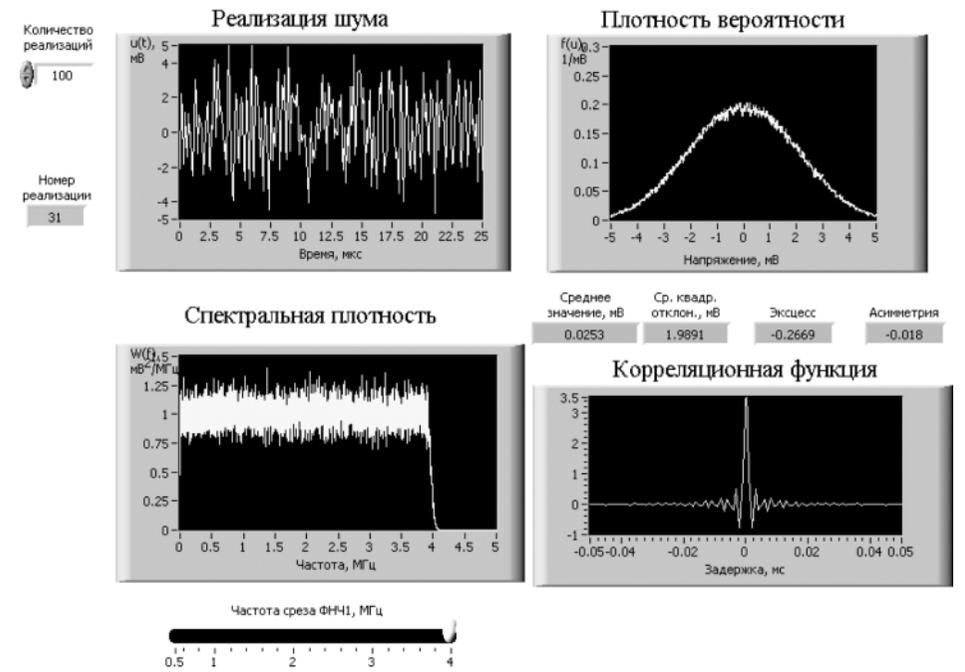


Рис. ПЗ.16

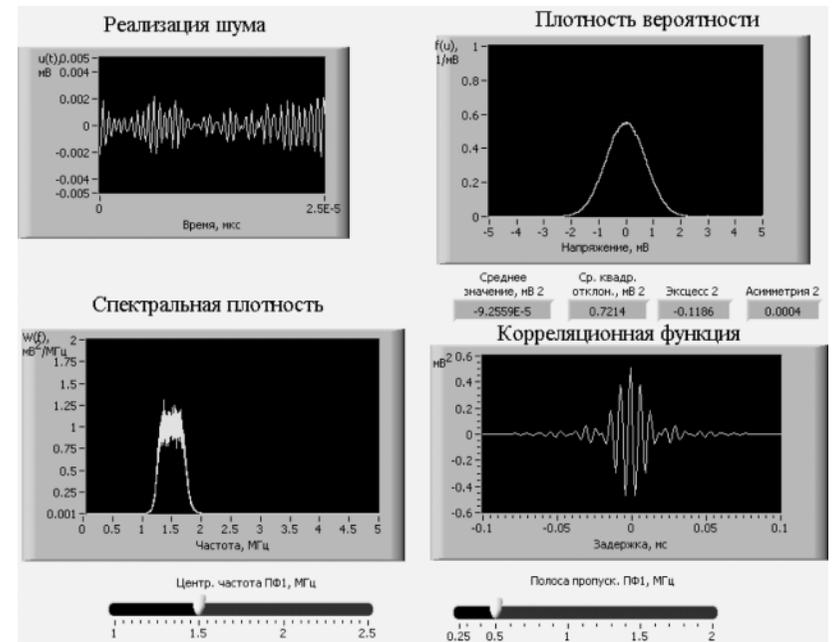


Рис. ПЗ.17

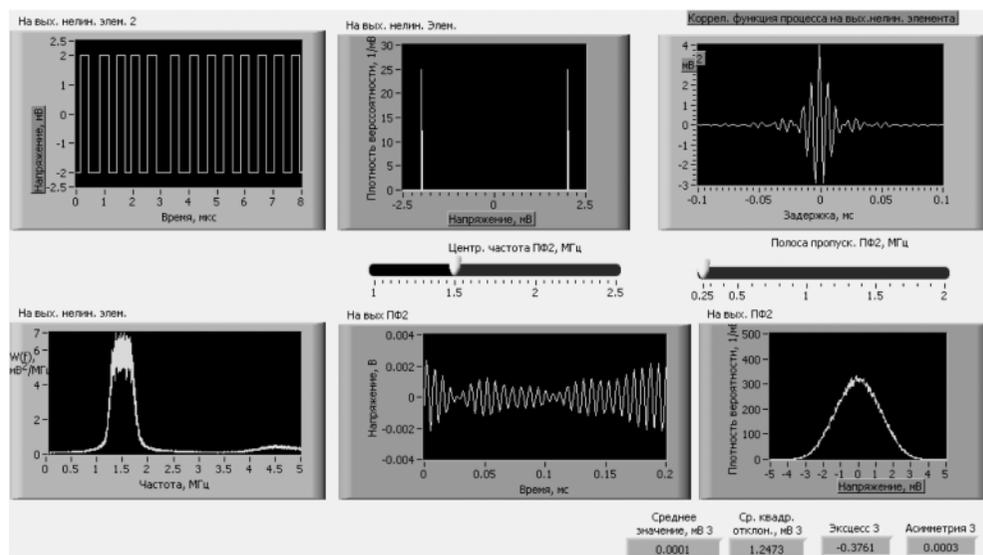


Рис. ПЗ.18

На лицевые панели выведены следующие элементы управления и индикаторы:

- установка количества реализаций для определения основных характеристик и параметров входного и выходного СП с заданной методической случайной погрешностью, обусловленной конечным временем усреднения;
- управление величиной частоты среза формирующего ФНЧ, МГц (ФНЧ1);
- индикатор номера реализации для контроля длительности эксперимента;
- индикатор измеренного среднеквадратического отклонения (СКО) СП в мВ;
- индикатор измеренного среднего значения в мВ;
- индикатор измеренного значения коэффициента эксцесса;
- индикатор измеренного значения коэффициента асимметрии;
- элемент установки центральной частоты полосового нормализующего полосового фильтра в МГц;
- элемент установки полосы нормализующего полосового фильтра в МГц.

На три лицевые панели виртуального прибора выводится графическая информация о реализациях СП, спектральных плотностях мощности, корреляционных функциях, плотностях вероятности, измеренных среднем значении, СКО, коэффициентах асимметрии и эксцесса.

## Задание на предварительное внеаудиторное исследование

1. Освоить временной и спектральный методы анализа линейных электрических цепей при пропускании через них случайных сигналов.
2. Рассчитать спектральную плотность мощности и корреляционную функцию для заданной преподавателем линейной цепи при воздействии на ее

вход белого шума. Определить среднеквадратическое значение процесса на выходе цепи при заданном значении спектральной плотности мощности белого шума.

3. Определить принципиальные различия в корреляционных функциях широкополосного и узкополосного СП.
4. Выяснить различия в реализациях широкополосного и узкополосного СП.
5. Выяснить, как по спектральной плотности СП рассчитать его дисперсию.
6. Определить связь между площадью, ограниченной корреляционной функцией, и значением спектральной плотности мощности при нулевом значении частоты.

## Задание на лабораторное исследование

1. Установить заданное преподавателем значение частоты среза формирующего ФНЧ1.
2. Исходя из заданной погрешности оценки спектральной плотности мощности и допуская ее прямоугольную форму, установить требуемое число реализаций входного случайного процесса.
3. Установить среднюю частоту и полосу пропускания узкополосного фильтра ПФ1 так, чтобы спектральная плотность выходного СП находилась в полосе пропускания ФНЧ1 (форму АЧХ ПФ1 считать прямоугольной).
4. Установить среднюю частоту и полосу пропускания нормализующего фильтра ПФ2 так, чтобы его АЧХ (прямоугольная) соответствовала середине АЧХ ПФ1.
5. Запустить виртуальную установку.
6. По достижении заданного числа реализаций остановить анализ для проверки работоспособности установки.
7. Перенести в отчет, создаваемый в Word, все три лицевые панели, используя буферную память ПК.
8. Изменяя полосу пропускания нормализующего фильтра ПФ2, выполнить анализ соответствия плотности вероятности СП на выходе ПФ2 нормальному закону по значениям коэффициента эксцесса этого процесса.
9. Сравнить все реализации, корреляционные функции, плотности вероятности на выходах виртуальных узлов установки по форме и численным значениям, сделать выводы из сравнения.

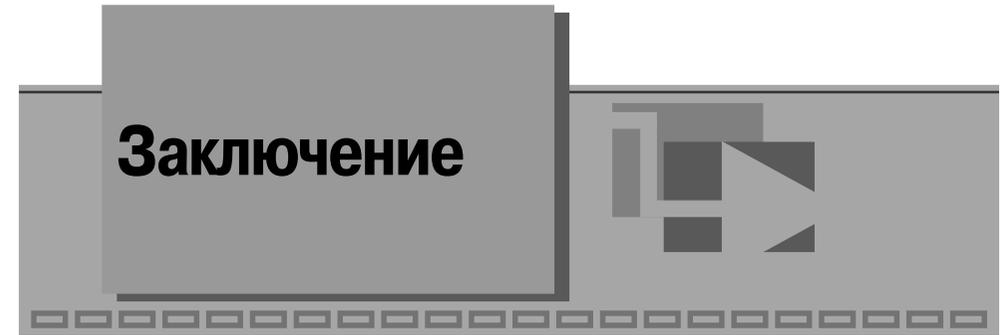
## Содержание отчета о проведении исследования

1. Результаты расчетов, возможный вид реализаций СП, спектральных плотностей мощности, корреляционных функций, плотностей вероятности, подготовленных при выполнении предварительного внеаудиторного исследования.
2. Реализации, спектральные плотности мощности, корреляционные функции, плотности вероятности, параметры плотностей вероятностей, зафиксированных при выполнении всех пунктов лабораторного исследования и перенесенных в отчет с помощью буферной памяти ПК.

3. Провести анализ полученных результатов и сделать выводы по всем пунктам лабораторного исследования.

## Контрольные вопросы

1. Какие преобразования процессов называют линейными?
2. В чем состоит принцип суперпозиции? В каких системах он выполняется?
3. Дайте определение непрерывности случайной функции. В чем состоят условия непрерывности (в среднеквадратическом смысле)?
4. Что понимают под производной случайного процесса? При выполнении каких условий процесс является дифференцируемым (в среднеквадратическом смысле)?
5. Что следует понимать под интегральным преобразованием случайной функции?
6. Каковы закономерности преобразования моментных функций при линейных преобразованиях случайных процессов?
7. Какими методами производится анализ прохождения случайных процессов через линейные цепи?
8. В каких случаях плотность вероятности не меняет вида при линейных преобразованиях?
9. Почему при использовании спектрального метода требуется знать квадрат коэффициента передачи цепи?
10. При использовании каких методов расчета учитывается процесс установления стационарного режима?
11. В чем состоят особенности анализа линейных цепей при воздействии белого шума? В каких случаях реальный процесс можно заменить моделью белого шума?
12. Как определяется шумовая (эффективная) полоса пропускания цепи?
13. Как влияет на спектральную плотность мощности и корреляционную функцию выходного процесса изменение постоянной времени интегрирующей цепи?
14. Чем отличается идеальное интегрирование случайного процесса от его прохождения через интегрирующую цепь?
15. Каковы особенности прохождения широкополосного шума через колебательные цепи? Как влияет на величину дисперсии и на вид корреляционной функции и спектральной плотности мощности изменение добротности колебательного контура?
16. В чем состоит эффект нормализации случайных процессов в линейных цепях? При каких условиях он происходит? Когда возможен противоположный эффект?
17. Как с помощью центральной предельной теоремы теории вероятности объяснить степень близости плотности вероятности выходного стационарного процесса к нормальной при возрастании инерционности линейной цепи?
18. В чем сходство и различие корреляционных функций и реализаций узкополосного и широкополосного случайных колебаний (при одинаковых энергетической полосе процессов и форме спектральной плотности мощности)?



Представленное учебное пособие показало возможность объединения программиста и инженера в одном лице. Однако для освоения программного пакета LabVIEW в рамках бакалавриата, студенту и магистранту по направлениям «Радиотехника» и «Телекоммуникации» необходимы знания базовых дисциплин учебного плана: основы теории цепей, радиотехнические цепи и сигналы, теория электросвязи, метрология и радиоизмерения, цифровая обработка сигнала, радиотехнические системы, в частности системы связи. Представленные виртуальные приборы могут служить основой для проведения моделирования и экспериментальных исследований на основе творческого подхода при усвоении дисциплин. Кроме того, пособие может послужить средством для организации курсового проектирования, а также окажет помощь при дипломном проектировании.

Авторы надеются, что материал, представленный в учебном пособии, послужит основой для проведения экспериментальных исследований по темам выпускных и диссертационных работ магистрантами и аспирантами.

# Список литературы

1. Akos D.M., M. Stockmaster, J.B.Y. Tsui and J. Caschera. Direct Bandpass Sampling of Multiple Distinct RF Signals // IEEE Transactions on Communications, Vol. 47. July 1999. P. 983–988.
2. Liu J., X. Zhou and Y. Peng. Spectral Arrangement and Other Topics in First-Order Bandpass Sampling Theory // IEEE Transactions on Signal Processing, Vol. 49. June 2001. P. 1260–1263.
3. Vaughan R.G., N.L. Scott and D.R. White. The Theory of Bandpass Sampling // IEEE Transactions on Signal Processing, Vol. 39. September 1991. P. 1973–1984.
4. Oppenheim A.V., R.W. Schaffer and J. R. Buck Discrete-Time Signal Processing. 2d ed. Prentice Hall, Upper Saddle River, NJ, 1998.
5. Brannon B. Basics of Designing a Digital Radio Receiver: Analog Devices, 1999. Available at [www.analog.com](http://www.analog.com).
6. Proakis J.G. Digital Communications. 4th ed. McGraw-Hill, New York, 2001.
7. Sklar B. Digital Communications. 2d ed. Prentice Hall, Upper Saddle River, NJ, 2001.
8. Harris F. On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform // Proceedings of the IEEE, Vol. 66, no. 1. January 1978.
9. Harris F. Multirate Digital Signal Processing, Prentice-Hall, Upper Saddle River, NJ, 2004.
10. Cory L. Clark. LabVIEW Digital Signal Processing and Digital Communication. McGraw-Hill, 2005. 225 p.
11. Lyons R.G. Understanding Digital Signal Processing. Prentice-Hall, Upper Saddle River, NJ, 2001.
12. Stuber G.L. Principles of Mobile Communication. 2d ed. Kluwer Academic Publishers, Boston, MA, 2001.
13. Prabhu G.S. and P.M. Shankar. Simulation of Flat Fading Using MATLAB for Classroom Instruction // IEEE Trans. on Education, Vol. 45, no 2. February 2002. P. 19–25.
14. Ingle V.K. and J.G. Proakis Digital Signal Processing using Matlab, Brooks/Cole, Pacific Grove, CA, 2000.
15. Rabiner L.R. and R.W. Schaffer Digital Processing of Speech Signals. Prentice-Hall, Upper Saddle River, NJ, 1978.
16. Рабинер Л., Гоулд Б. Теория и применение цифровой обработки сигналов. – М.: Мир, 1978. – 848 с.
17. Основы цифровой обработки сигналов: курс лекций / А.И. Солонина, Д.А. Улахович, С.М. Арбузов, Е.Б. Соловьева. 2-е изд. испр. и перераб. – СПб.: БВХ-Петербург, 2005. – 768 с.

18. Сергиенко А.Б. Цифровая обработка сигналов: учебник для вузов. 2-е изд. – СПб.: Питер, 2006. – 751 с.
19. Гоноровский И.С., Демин М.П. Радиотехнические цепи и сигналы: учебное пособие. – М.: Радио и связь, 1994. – 480 с.
20. [www.ni.com](http://www.ni.com)
21. FPGA Interface User Guide, National Instruments, 2004.
22. Chugani, Mahesh L.; Abhay R. Samant; and Michael Cerna. LabVIEW Signal Processing. Upper Saddle River, NJ: Prentice Hall, 1998.
23. Diniz, Paulo S.R.; Eduardo A.B. da Silva; and Sergio L Netto. Digital Signal Processing: System Analysis and Design. New York: Cambridge University Press, 2002.
24. Hogenauer E.B. An economical class of digital filters for decimation and interpolation // IEEE Transactions on Acoustics, Speech, and Signal Processing. ASSP-29 (2). 1981. P. 155–162.
25. Ifeachor E.C. and B. W. Jervis. Digital Signal Processing: A Practical Approach. 2d ed. Publishing House of Electronics Industry, 2003.
26. Jayasimha S. and P. V. R. N. Rao. An iteration scheme for the design of equiripple Mth-band FIR filters // IEEE Transactions on Signal Processing. Vol. 43 (8). Aug. 1995. P. 1998–2002.
27. Mintzer F. On half-band, third-band, and nth-band FIR filters and their design // IEEE Transactions on Acoustics, Speech, and Signal Processing. ASSP-30 (5). October 1982. P. 734–738.
28. Neuvo Y; C-Y Dong and S.K. Mitra. Interpolated finite impulse response filters // IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-32. June 1984. P. 563–570.
29. Oppenheim A.V. and R.W. Schaffer. Discrete-Time Signal Processing. Englewood Cliffs, NJ: Prentice Hall, 1989.
30. Orfanidis S.J. Introduction to Signal Processing. Upper Saddle River, NJ: Prentice Hall, 1998.
31. Parks T.W. and C.S. Burrus. Digital Filter Design. New York: John Wiley & Sons, Inc., 1987.
32. Rabiner L.R. Approximate design relationships for low-pass FIR digital filters // IEEE Transactions on Audio and Electroacoustics. Vol. 21 (5). Oct. 1973. P. 456–460.
33. Selesnick I.W. and C. S. Burrus. Generalized digital Butterworth filter design // IEEE Transactions on Signal Processing. Vol. 46 (6). June 1998. P. 1688–1694.
34. Vaidyanathan P.P. Multirate Systems and Filter Banks. Englewood Cliffs, NJ: Prentice Hall, 1993.
35. Vaidyanathan P.P. and T.Q. Nguyen. A 'trick' for the design of FIR half-band filters // IEEE Transactions on Circuits and Systems. Vol. 34 (3). March 1987. P. 297–300.
36. <http://zone.ni.com/devzone/conceptd.nsf/webmain/C37021EF11C2CB758625702200039D67>.
37. <http://zone.ni.com/devzone/devzone.nsf/webcategories/1284E6919C8B33CE862567AF0075696B>.
38. <<http://zone.ni.com/devzone/devzoneweb.nsf/OpenDoc?openagent&568C773718B080F286257096005BD314>>.

# Предметный указатель

## A

Acqiris, 34, 46, 47, 49, 50  
 A/D, 21, 23, 32, 212  
 AdvFFT, 133, 134, 135, 136, 139, 141, 150, 200, 414  
 Aliasing, 338, 347  
 allpass filter, 289, 290, 291, 306, 307  
 Amplitude Spectral Density, 118  
 Amplitude Spectrum,  
 Analyzer FFT, 96  
 analog, 21, 244  
 Analog to Digital Converter  
 Anti-aliasing Filter, 338  
   ideal, 26  
   practical, 26  
 APCO, 29, 31, 218  
 ARMA, 63, 79, 265, 289, 290, 306, 307, 309  
 Array, 22, 72, 145, 240, 387, 405, 414  
 ASIC, 256  
 Averaging, 125, 126  
 AWGN, 194, 195, 206, 213, 219, 395, 414, 429

## B

Bandpass filter, 66  
 Bandstop filter, 66  
 bins, 106  
 Blackman, окно, 140, 169, 373  
 Blackman – Harris, 138, 140, 141, 169  
 Bluetooth, 29, 31

## C

Cauer filter, 83  
 CDMA, 29, 31  
 Chebyshev, фильтр, 272, 281, 292  
 Coherence, 121  
 Coherent Gain, 172  
 Communication protocol, 31  
 CompactRIO, 238, 257, 261  
 Convolution encoder, 394,  
 Convolutional decoder, 397  
 Cross Power Spectrum, 121

## D

Data acquisition, 96  
 DAQ, 96, 99, 102, 245, 248, 259  
 Downsample, 173, 176, 199, 416  
 Decimation, 338, 343  
 Design, 342, 343, 344, 348, 349, 351, 352, 352, 353, 354, 355, 359, 360, 361, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379  
 DFT, 95, 99  
 Digital Signal Processor, 185  
 Direct form, 289, 290, 292, 304, 306, 307, 309  
 Discrete Fourier Transform, 95  
 DLL, 145, 146, 147, 148, 150, 237  
 DQPSK, 31, 392, 393  
 DSK, 239, 240, 241, 242  
 DSP, 242, 243, 244, 265, 266, 267, 268, 286, 287, 303, 310

DSP Test Toolkit, 239

## E

EDGE, 29, 31, 271, 390  
 ENBW, 172  
 Equiripple, 71, 72, 73, 75, 76  
 Estimate Power, 117  
 Estimate Frequency, 116  
 Exponential, 167, 171

## F

Fast Fourier Transform, 95  
 FFT, 42, 72, 95, 96, 108, 113, 120, 145, 147, 148, 222, 225, 414  
   based, 95, 105, 107  
 filter halfband, 179, 180, 349, 351, 418  
 Filter Coefficient, 76, 78, 82, 267, 297, 300  
 filter Order, 282, 289, 290, 292, 304, 306, 307, 309, 312, 339, 355, 365, 367  
 filter Specification, 93  
 FIR filter, 63, 78, 283, 284  
   design, 282, 286, 311, 373, 374, 375, 377, 378, 379  
   direct form, 290, 292, 304, 306, 307, 309  
 Flat top Window, 158, 159, 162, 166, 167, 169, 171, 172  
 FPGA, 256, 257, 258, 259, 260, 261, 262, 265, 266, 286, 287, 303, 305, 306, 308, 309, 310  
 Frequency Domain, 99, 154, 388  
 FSK, 219, 391, 395, 396

## G

GMSK, 31, 391  
 GPIB, 40, 57  
 GPS, 31  
 Group delay, 292, 328, 329, 331, 336, 367, 370, 379  
 GSM, 29, 31, 38, 218

## H

Hamming, 138, 140, 164, 168, 373, 394, 397  
 Hann, 138, 139, 140  
 Hanning, 158, 159, 163, 168, 171, 173  
 Hartley, 130  
 help, 220, 264, 274, 275, 277, 290, 311, 339, 341, 348, 349, 351, 352, 354, 367, 375  
 heterodyne, 242, 243, 244  
 Hilbert, 127, 129, 130, 314, 335, 355, 356, 367

## I

IF, 32, 37, 38, 41, 45  
 IFIR, 76, 78, 331  
 IIR filter, 90, 277, 282, 286, 311  
 IMAQ Vision, 383, 384, 385, 387, 388, 390  
 Impulse response, 63  
 Interface, 232, 233, 234, 237  
 Intermediate Frequency, 32  
 Intersymbol Interference, 212  
 Inverse Chebyshev filter, 272, 281, 292  
 ISI, 200, 212

## K

Kaiser, 140, 174, 274, 280, 292, 411, 420  
 Kaiser – Bessel, 165, 169, 171

## L

LabWindows, 383, 384  
 Lattice ARMA, 265, 289, 306, 307, 309  
 Lattice MA, 265, 289, 290, 306, 307  
 Least Pth Norm Design, 264, 326, 328, 335, 353, 365, 366, 367, 368, 369, 370, 372  
 Local oscillator, 32  
 LO, 32, 33, 38, 40  
 Lowpass filter, 66, 283, 284, 285, 287, 297, 300, 301, 308, 309

LPC, вокодер, 228, 232  
LPCcoeffs, 231, 232, 420

**M**

Magnitude, 70, 205, 279, 280, 284, 379  
Mapping, 199  
MATLAB, 19, 30, 218  
Median Filter, 91, 92, 93  
Modulation Toolkit, 391, 392, 394, 395, 396, 397,  
Moving average, 265, 411  
MSK, 219, 391, 395, 396

**N**

Narrowband coefficients, 76, 78  
Nearest, 294, 295  
NF, 48, 50  
NI Toolset, 218, 219  
NI Vision Assistant, 383  
Noise, 172, 251, 322, 395  
Normalized, 306, 307, 309  
Nyquist, 25, 35, 69, 77, 152, 271, 349  
Design, 338, 339, 341, 349, 350  
Pulse, 200, 424  
Shift, 111, 112

**O**

OFDM, 29, 31  
Order, 289, 290, 292, 304, 306, 307, 309, 312, 339, 341, 355, 365, 367  
Overflow, 294, 299, 300, 304

**P**

Passband, 66, 67, 93, 271, 280, 395  
Frequency, 283  
range, 334, 347, 349  
Ripple, 278, 279, 283, 334  
PC, 41, 58, 221, 223, 261, 430  
PCI, 46, 58, 238, 257  
Phase, 93, 289, 306, 307, 309, 328, 341, 345, 346, 355, 359, 366, 367, 369, 370, 379, 395

Response, 70  
Spectrum, 120  
pole, 368  
poleZero, 228, 229, 230, 275, 284, 426  
polyphase, 182, 184, 426  
Power Spectral Density, 118  
power spectrum, 111, 112, 113, 121, 142  
Pulse Parameters, 91, 92  
PXI, 33, 41, 42, 43, 44, 45, 47, 60, 149, 150, 218, 220, 238, 257, 261, 305, 392, 418, 426

**Q**

QAM, 50, 52, 53, 54, 56, 194, 195, 196, 198, 200, 213, 214, 215, 219, 291, 397, 414, 418, 420, 427, 430  
modulate, 395  
demodulate, 396  
QPSK, 31, 194, 391  
Quantizers, 296, 300

**R**

Radio frequency, 32  
Signal analyzer, 42  
Reed Solomon, код, 394  
Remez Exchange, 73, 75  
Remez Design, 264, 267, 312, 314, 315, 316, 318, 325, 326, 334, 335, 353, 354, 355, 359, 360, 361, 363, 364, 365, 366, 371  
Rectangular, 171, 373  
Resample, 219, 337  
RF, 32, 40, 41, 149, 192, 193  
RFSA, 42, 45, 150  
RIO, 256, 257, 305

**S**

Samples, 145  
Sampling, 25, 151  
Frequency, 283  
Scaling Factor, 172  
scalloping, 150

Scope, 42, 150  
Shannon Sampling Theorem, 25, 151  
Simulink, 233, 234, 235, 237, 238  
sinc, 156, 187, 188, 190, 327, 328, 429  
SMT, 42, 43, 45, 149, 150  
SNR, 118, 253  
Sound and Vibration Toolkit, 245, 246, 247, 248, 249, 253  
Spectral Measurement Toolset, 42  
Square, 396  
Stopband Attenuator, 283  
Structure, 287, 290, 291, 310, 404

**T**

target structure, 291  
TETRA, 31  
Time domain, 99, 154  
Triangle, 166, 168  
Transition Band, 26, 93, 271  
Transition Regions, 67  
Truncation, 294, 295, 304  
Tuner, 42, 150

**U**

UMTS, 31  
Underflows, 299, 300  
Uniform, 158, 159, 162, 171, 172  
Upsample, 173, 174, 199, 434  
USB, 29, 58

**W**

Waterfall, 253  
Windoved, 292, 373, 374, 375, 377, 378,  
Windows, 30, 58, 141, 151, 261, 274, 280, 292, 305, 375, 378, 399, 400, 403, 408, 409, 412, 434  
Windowcoefs, 141  
Windowing, 74, 75, 150

**Z**

Zero, 341, 345, 346

**A**

алгоритм, 20, 21, 23, 28, 65, 69, 73, 75, 76, 99, 107, 130, 140, 145, 152, 182, 221, 228, 232, 237, 256, 264, 267, 282, 289, 312, 344, 363, 367, 383, 389, 390, 392, 399, 404, 409, 414, 418, 424  
свертки, 397  
Ремеза, 274, 353  
амплитуда, 27, 28, 48, 68, 72, 92, 95, 96, 99, 102, 114, 119, 124, 127, 156, 166, 170, 171, 175, 186, 205, 248, 414, 418  
амплитудный спектр, 116, 119, 120  
анализ спектра, 57, 253  
октавный, 248, 252  
анализ искажений, 247, 253  
аналитический сигнал, 129  
аналого-цифровой  
преобразователь, 22, 32  
АЦП, 21, 22, 23, 26, 27, 32, 33, 34, 40, 41, 45, 46, 48, 54, 55, 58, 96, 142, 143, 338, 426  
АЧХ, 26, 46, 67, 70, 71, 72, 73, 74, 76, 77, 78, 83, 84, 85, 87, 88, 89, 118, 123, 183, 200, 264, 272, 282, 284, 286, 297, 298, 311, 312, 313, 314, 315, 316, 316, 317, 318, 319, 320, 321, 322, 323, 325, 324, 325, 326, 327, 328, 329, 331, 332, 333, 334, 335, 336, 338, 343, 344, 349, 351, 353, 355, 356, 357, 358, 360, 361, 361, 362, 363, 364, 365, 366, 368, 369, 370, 371, 372, 373, 379, 444, 448, 453  
Максимально плоская, 190, 265, 271, 274  
пульсации, 75, 93, 271, 276, 278, 279, 280, 318, 319, 354, 359, 363, 364, 365, 368, 369  
склон, 338, 341, 349, 350, 351

**Б**

белый шум, 48, 171, 299

БИХ-фильтр, 62, 63, 64, 65, 69, 79, 80, 81, 82, 83, 87, 88, 89, 90, 91, 265, 269, 270, 271, 272, 275, 276, 281, 286, 287, 289, 290, 291, 292, 296, 297, 320, 326, 328, 329, 331, 335, 337, 365, 366, 369, 370, 371, 372, 418

Блэкмана – Хэрриса, 138, 140, 141, 142

БПФ, 72, 95, 105, 106, 107, 108, 110, 113, 115, 117, 118, 119, 120, 124, 125, 126, 130, 131, 132, 132, 133, 134, 135, 142, 145, 150, 152, 154, 159, 162, 172, 174, 222, 223, 224, 414, 418, 427, 434

анализатор, 96

базовое, 95, 105

быстродействие, 106

двухстороннее, 110, 111, 120

дискретное, 97

комплексное, 108, 125, 126, 130, 132, 133, 224, 418

обратное, 74, 123, 224

реальное, 108, 130, 224

стандартное, 132

улучшенное, 134

быстрое преобразование Фурье, 72

**Г**

гармонический анализ, 247, 253

генерация, 185, 192, 193, 194, 196, 198, 251, 288, 289, 302, 308, 310

гетеродин, 32, 33, 38, 40, 244

гетеродинирование, 32, 40

глазковая диаграмма, 397

групповая задержка, 292, 311, 329, 370

**Д**

декодер, 215

декодирование, 199, 397

демодулятор, 202, 203, 205, 208, 396

демодуляция, 395, 396,

дещимация, 336, 337, 338, 339, 343

динамический диапазон, 247, 255, 295

дискретизация, 20, 34, 38, 134, 137, 152, 153, 154, 336, 337

повторная, 337

дискретное преобразование Фурье, 95, 99, 100, 101, 103, 131, 133

дисперсия, 28, 311, 444, 446

дифференциатор, 313, 317, 318, 335, 353, 355, 356, 371

типа IV, 318, 319

типа III, 318, 319, 320

дополнение нулями, 107

ДПФ, 95, 97, 99, 100, 101, 102, 103, 105, 106, 107, 113, 115, 116, 133, 134, 135, 137, 138, 139, 141, 150, 152, 154, 162, 168, 224

**З**

задержка, 90, 92, 186, 187, 199, 200, 304, 311, 343, 346, 358, 359

запятая, 270

затухание, 167, 271, 283, 334, 338, 341

звук, 58, 60, 232, 242, 245, 245, 248, 251, 252

звуковой анализ, 248

звуковой сигнал, 62, 246, 247

звуковая карта, 58, 59, 232, 430

ЗФ, 66, 67, 72, 76, 77, 82, 90, 271, 272, 274, 276

**И**

импульс, 63, 65, 72, 74, 79, 91, 92, 93, 123, 185, 186, 187, 200, 203, 213, 228, 299, 414, 420, 424, 426, 430, 448

импульсный сигнал, 72

импульсная характеристика, 62, 63, 65, 69, 72, 74, 75, 77, 78, 79, 91, 105, 122, 123, 179, 181, 200, 203, 215, 259, 270, 316, 355, 373, 379, 447

конечная, 76

бесконечная, 79

корреляционная характеристика, 443, 444

импульсная передаточная функция, 266, 270, 274, 344, 356, 357, 358

интегрирование, 247, 250, 251, 454

интервал дискретизации, 21, 110

интерполяция, 337, 339, 340

искажения, 23, 87, 117, 206, 207, 209, 211, 212, 220, 245, 287, 296, 311, 312, 336, 346, 395

**К**

каскадный график, 253

квантование, 287, 291

КИХ-фильтр, 62, 63, 64, 65, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 88, 89, 90, 91, 151, 169, 269, 270, 171, 274, 175, 276, 280, 281, 282, 286, 289, 290, 291, 292, 312, 313, 314, 317, 325, 326, 327, 331, 334, 335, 336, 337, 338, 339, 341, 341, 344, 345, 350, 353, 354, 355, 356, 357, 358, 359, 365, 366, 367, 368, 369, 370, 371, 372, 373, 375

код, 21, 29, 38, 130, 142, 146, 145, 221, 237, 239, 240, 241, 242, 244, 259, 260, 261, 265, 266, 267, 287, 303, 304, 305, 306, 307, 308, 310

FPGA, 261, 262, 264, 266

STCL, 305, 308, 309

генерация, 288, 289, 294, 302

Грея, 195, 196, 200, 206

разрядность, 290

С, 130, 145, 147, 265, 289, 302, 303, 304

кодирование, 30, 206, 221, 228, 232

STCL, 305, 309

речи, 228

кодовый узел интерфейса, 130

компенсаторы групповой задержки, 336

комплексная аппроксимация, 355

огibaющая, 192

передаточная функция, 63, 121, 123, 275, 282, 312, 314, 317

последовательность, 104

коэффициенты, 37, 69, 74, 76, 80, 91, 141, 179, 181, 190, 200, 228, 268, 270, 289, 290, 291, 293, 295, 299, 303, 304, 332, 334, 337, 338, 341, 349, 353, 396, 426

асимметрии, 452

Баттерворта, 265

весовые, 106, 126, 328, 354, 360, 361

гармоник, 247, 253

дещимации, 342, 343, 346, 348

дифференциального уравнения, 228

изменения частоты дискретизации, 179

интерполяции, 183, 342

когерентности, 124

комплексные, 80

корректирующие, 172

лестничной структуры, 290

масштабирования, 172

набор, 278, 296, 357

нелинейных искажения, 96

окна, 140

отражения, 232

передачи, 26, 66, 67, 68, 271, 272, 275, 278, 279, 280, 359, 361, 362, 372, 445, 454

канала, 193

преобразования частоты, 343, 348

ряда Фурье, 108

с фиксированной запятой, 303

уменьшения частоты дискретизации, 175, 176

фильтра, 65, 70, 75, 76, 151, 168, 169, 170, 175, 181, 182, 183, 185, 200, 203, 222, 224, 232, 265, 269, 275, 286, 287, 289, 293, 297, 302, 303, 312, 319, 320, 329, 344, 351, 355, 357, 366, 414, 418, 420, 424

- Найквиста, 350  
 обратные, 79, 82  
 прямые, 79, 82  
 цифровые, 269  
 эксцесса, 452, 453
- Л**  
 лепестки спектра, 262
- М**  
 массив, 22, 109, 111, 115, 116, 119, 120, 258, 354, 381, 382, 414  
 входной, 111, 171, 223, 414  
 двумерный, 386, 387  
 данных, 240  
 отсчетов БПФ, 113  
 спектра, 159  
 мощности, 113  
 двухстороннего, 115  
 частотной характеристики, 123  
 масштабирование, 248  
 входного сигнала, 205  
 выходного сигнала, 205  
 окон сглаживания, 171  
 мгновенная частота, 191, 414  
 модулятор, 199, 200, 201, 220, 414, 429, 434  
 модуляция, 52, 218, 319, 394, 395, 396  
 амплитудно-фазовая, 391, 395, 396  
 квадратурная амплитудная, 194, 395, 396, 414  
 квадратурная фазовая, 391, 393  
 комплексными экспонентами, 108  
 амплитудно-импульсная, 194
- О**  
 огибающая, 203, 208  
 комплексная, 192  
 рэлеевская, 192, 206  
 окно, 124, 137, 138, 142, 160, 162, 168, 171, 172, 410  
 LabVIEW, 243, 409, 411
- MathScript, 409  
 Simulink, 235  
 Блэкмена, 169  
 точное, 169  
 Блэкмена – Харриса, 141, 142, 169  
 диалоговое, 278, 279, 283, 284, 293, 297, 379, 380, 411  
 пользователя, 384  
 дискретизации выборки, 137, 139  
 Дольфа – Чебышева, 280, 292  
 Кайзера, 280, 292  
 Кайзера – Бесселя, 165, 166, 169  
 команд, 409  
 масштабируемое, 168  
 косинуса, 169  
 менеджера подключений, 236  
 модели, 234  
 палитры, 146  
 пользователя, 234, 235  
 прямоугольное, 141, 142, 156, 158, 163, 166, 170, 374  
 симметричное, 169, 170  
 сглаживания, 74, 156, 157, 160, 162, 165, 168, 170, 171  
 с плоской вершиной, 158, 159, 166, 167, 169  
 статуса, 244  
 текстовое, 411, 412  
 треугольное, 166, 168  
 Ханна, 156  
 Хэмминга, 156, 164, 168  
 Хэннинга, 158, 159, 163, 164, 168, 169, 170, 375  
 экспоненциальное, 167, 168  
 округление, 295, 304  
 ОСШ, 32, 37, 48, 53, 55, 206, 208  
 в символе, 52, 54  
 ухудшение (потери), 37, 38, 51, 54, 60  
 отводы, 70  
 отношение сигнал/шум, 32, 206  
 отсчеты, 90, 99, 156, 337, 338, 341, 343  
 входные, 70, 80, 135, 222, 269  
 выборки, 109, 154

- выходные, 43, 70, 80, 82, 181, 183, 224, 269, 270  
 комплексной огибающей, 206  
 мнимой части, 108  
 реальные, 108  
 сигнала, 21, 33, 65, 108, 159  
 спектральные, 99, 106, 138, 139  
 цифровые, 41
- П**  
 передискретизация, 219  
 переполнение, 287, 295, 296, 297, 299, 300  
 период дискретизации, 32  
 ПЛИС, 18, 156, 258, 159, 260, 261, 262, 265, 266, 286, 287, 290, 305, 306, 308, 310  
 повышение частоты  
 дискретизации, 173, 183, 199, 200  
 поле, 244, 358  
 комплексного переменного, 266  
 нулей и полюсов, 228, 275, 276, 282, 284, 373  
 таблицы, 409, 410  
 текстовое, 409  
 полифазный фильтр, 178, 183  
 полоса, 75, 80, 83, 127, 354, 359, 360, 361, 363, 364, 365  
 канала, 31  
 Найквиста, 337  
 перехода, 26, 67, 69, 74, 93, 271, 272, 281, 331, 342, 347, 354, 367  
 пропускания (прозрачности), 57, 65, 66, 67, 69, 76, 78, 200, 283, 343, 344, 354, 447, 454  
 звуковой карты, 58  
 избыточная, 202  
 эквивалентная, 172  
 эффективная, 172  
 задержания, 66, 67, 69, 76, 283, 334, 343, 354  
 односторонняя, 200  
 сигнала, 45, 53, 175, 176
- фильтра, 347  
 частот, 45, 46, 57, 354, 443  
 полосовой фильтр, 267, 271, 359  
 единственной точки  
 по частоте, 359, 360  
 эллиптический, 329  
 понижение частоты  
 дискретизации, 173  
 порядок, 90, 281, 374  
 нормы, 367  
 фильтра, 80, 200, 269, 275, 281, 282, 307, 309, 312, 313, 315, 316, 326, 328, 331, 332, 335, 336, 339, 341, 342, 344, 353, 355, 359, 364, 367, 418, 424  
 пробки, 320  
 числителя передаточной функции, 270, 307  
 знаменателя передаточной функции, 270, 307  
 постоянная составляющая, 106, 110, 112, 114  
 преобразование Гильберта, 127  
 преобразование Фурье, 97, 110, 132, 153, 154  
 быстрое, 105, 244, 253  
 дискретное, 99, 100, 101, 103, 131  
 комплексное, 131  
 непрерывное, 134  
 обратное, 97  
 реальное, 131  
 проектирование цифровых фильтров, 90, 265  
 прореживание, 60  
 выборки, 40, 45, 47, 48, 51, 52, 53, 55  
 нулями, 174  
 пульсации АЧХ, 93, 318, 355, 359  
 в полосе задержания, 93  
 в полосе пропускания, 67, 73, 93, 271, 276, 354  
 ПФ, 66, 67, 72, 76, 77, 78, 82, 88, 89, 90, 273, 453

**P**

отношения Парсеваля, 97  
теорема Парсеваля, 97, 98  
распределение Рэлея, 192, 193  
растекание спектра, 151, 152, 153,  
154, 155

Котельникова  
теорема, 25, 34, 151  
фильтр, 341

**C**

свертка, 70, 156, 183  
дискретная, 269  
круговая, 138  
линейная, 223, 224  
сигнал, 21, 22, 23, 26, 31, 32, 33, 34,  
39, 40, 43, 45, 46, 55, 91, 92, 96,  
116, 124, 157  
GSM-ячеистый, 38  
аналоговый, 20, 21, 22, 23, 24, 25,  
26, 27, 29, 35, 37, 59, 62, 135, 396  
аудио (звуковой), 27, 62, 246  
взвешенный, 142, 251  
видео, 41, 65, 323  
восстановленный, 38, 48  
входной, 26, 34, 64, 65, 66, 102, 103,  
116, 122, 123, 124, 129, 130, 133,  
134, 135, 137, 138, 141, 143, 149,  
154, 156, 162, 175, 181, 182, 185,  
205, 224, 267, 268, 269, 271, 276,  
284, 287, 291, 311, 342, 346, 414,  
420, 442, 443, 444  
высокочастотный, 41, 42  
выходной, 64, 65, 91, 123, 124, 134,  
147, 200, 203, 205, 212, 214, 215,  
228, 270, 276, 284, 287, 291, 311,  
442, 443, 444  
гармонический  
(синусоидальный), 24, 121,  
122, 158, 162, 244, 245, 247,  
251, 253, 320  
голосовой, 228  
взвешенный, 228

демодулируемый, 51  
дискретизированный, 35, 37  
дискретный, 21, 23, 24, 25, 26, 36,  
45, 48, 57, 156, 175, 177, 178  
импульсный, 72, 92, 185, 203  
информационный, 392, 394, 395  
комплексный (аналитический), 97,  
108, 127, 129, 132, 192, 253  
модулированный, 59, 193, 199, 206,  
245, 311, 395  
невзвешенный, 142  
обработка, 30  
периодический, 95, 124, 152  
радио, 27, 32, 33, 38, 43, 58, 220, 267  
реальный, 34, 35, 55, 58, 102, 108,  
113, 114, 131, 142, 224, 395  
ступенчатый, 299  
узкополосный, 108, 127  
цифровой, 20, 31, 39, 60, 108, 150,  
152, 173, 174, 199, 233, 257, 338,  
340, 396  
широкополосный, 443  
шумовой (случайный), 118, 126,  
171, 192, 452  
эхосигнал, 127, 128, 129  
сигнал/шум, 32, 54, 118, 206  
символ, 51, 52, 53, 54, 194, 196, 199,  
200, 203, 205, 206, 208, 215, 217,  
232, 414, 434  
синхронизация, 215, 217  
по частоте, 217  
по времени, 215  
системы, 29, 33, 54, 60  
аудио, 336  
дискретные, 70  
машинного зрения, 261  
операционные, 34, 221, 261  
параметрические, 268  
радиотехнические, 455  
связи, 30, 31, 37, 130, 134, 143, 199,  
200, 202, 204, 206, 208, 210, 212,  
214, 216, 218, 220, 455  
с фиксированной запятой, 286

цифровые, 29, 38, 40, 52, 206, 220,  
392, 394  
СКО, 28, 192, 452  
спектр, 34, 37, 46, 50, 51, 55, 58, 104,  
106, 111, 112, 115, 117, 124, 127,  
130, 132, 133, 135, 153, 158, 175,  
183, 186, 212, 244, 267, 325, 418  
амплитудный, 116, 118, 119, 120,  
121, 253  
взаимный (перекрестный), 105,  
120, 121, 122, 125, 126, 253  
взвешенного сигнала, 142  
гармонический, 249, 267  
двухсторонний, 110, 113, 114, 115,  
119, 120, 138  
диаграмма, 102, 117  
дискретного сигнала, 36, 52, 175,  
177, 178  
дублирования, 35  
комплексный, 119, 253, 255  
математический, 104  
мгновенный, 120  
модулированного сигнала, 395  
мощности, 83, 95, 105, 113, 114,  
115, 116, 117, 118, 119, 120, 122,  
123, 125, 130, 158, 159, 247, 253,  
254  
перекрестный, 122, 123  
наложения, 36, 37  
односторонний, 110, 115, 119, 120,  
152, 418  
окна, 138, 139, 156, 158, 159, 161,  
162, 169, 170  
октавный, 252  
непрерывный, 161  
периодического сигнала, 175  
полнооктавный, 252  
растекание, 151, 152, 152, 153, 154,  
155, 156  
реального сигнала, 51, 56, 114, 224  
сигнала электрокардиограммы, 321  
совмещение, 177, 342, 347  
средний, 123, 149

утечка, 124, 137, 138, 139, 141, 150,  
154, 155, 156, 159, 162, 163, 165, 170  
фазовый, 105, 118, 119, 120,  
121, 253  
цифрового сигнала, 338, 340  
шумовой, 185  
энергетический, 446  
спектральная плотность, 97, 142, 156,  
442, 449  
мощности, 46, 50, 53, 116, 117, 118,  
126, 143, 144, 172, 252, 253, 442,  
443, 444, 452, 453, 454  
односторонний, 116  
спектральное дублирование, 175  
спектральное разрешение, 135, 170  
спектральные преобразования, 130,  
142  
высокого уровня, 142, 143  
спектральные функции, 130  
спектральный анализ, 57, 85, 95, 96,  
97, 98, 102, 104, 105, 106, 112, 114,  
116, 118, 119, 120, 122, 124, 125,  
126, 128, 130, 132, 134, 136, 138,  
140, 142, 143, 151, 166, 168, 169,  
171, 172, 241, 247, 254  
в LabVIEW, 144, 146, 150  
среднее квадратическое  
отклонение, 192, 452  
среднеквадратическое  
значение, 92, 114, 115, 120, 122  
среднеквадратическое  
усреднение, 124, 125  
субдискретизация, 34, 38  
сходимость, 69

**T**

теорема  
Парсеваля, 97  
Шеннона, 152  
Фурье, 95

**У**

уровень вибрации, 252

уровень звука, 252  
 усечение, 74, 294  
 усреднение, 125, 143, 252  
   среднеквадратическое, 125  
   векторное, 125, 126  
   линейное, 247  
   пиковое, 125, 126  
   частотных компонент, 105, 126  
   экспоненциальное, 247, 252  
 устойчивость, 392  
 ухудшение (потери), 21, 22, 37, 51, 60, 150, 237, 287, 395  
   значимости, 295, 296, 297, 299, 300, 304

**Ф**

фаза, 102, 118, 192, 247, 248  
   начальная, 72, 102  
   системы, 121  
 фазовый, спектр, 119, 121  
 ФВЧ, 66, 67, 72, 76, 78, 81, 90, 271, 272, 273, 276, 313, 344, 361, 362, 363  
 фиксированная запятая, 270  
 фильтр, 26, 33, 63, 72, 75, 87, 90, 92, 94, 138, 176, 178, 200  
   авторегрессионного скользящего среднего, 63, 79  
   аналоговый, 62, 63  
   Баттерворта, 83, 84, 85, 86, 87, 272, 274, 281  
   Бесселя, 83, 87  
   верхних частот, 66  
   всепропускающий, 291, 344  
   второго порядка, 81, 82  
   входной, 32  
   Гильберта, 183, 356, 372  
   гребенчатый, 266, 323, 324, 325, 335, 343  
   децимации, 338, 342, 343, 347, 348, 352  
   дискретный, 63  
   заграждающий,  
   идеальный, 64, 69  
   инвариантный во времени, 268  
   интеграторы, 343, 344  
   интерполирующий, 334, 339, 340  
   каскадные, 343  
   Кауэра, 83  
   линейный, 268  
   медианный, 91, 92, 93  
   максимально-фазовый, 291, 353, 355, 356, 357, 369, 372, 291, 356, 358  
   минимально-фазовый, 291, 311, 328, 353, 355, 356, 357, 359, 369, 372, 291, 356, 359  
   многокаскадный, 353  
   многоскоростной, 336, 337, 342, 344, 346, 348, 349, 352  
   многоступенчатый, 342, 343, 344  
   модель, 228  
   Найквиста, 266, 341, 349, 350, 353  
   нелинейные, 63, 91  
   нижних частот, 22, 26  
   обратного отображения отсчетов, 340  
   оптимальный, 280  
   передискретизации, 177, 179, 181, 183  
   пиковый, 320, 324, 335, 353, 360  
   «поднятый косинус», 266, 351  
   полифазный, 178, 181, 183  
   полосовой, 359, 360, 452  
   полуполосы, 179, 183, 351, 418  
   порядок, 355, 367, 418, 424  
   практический, 69  
   причинный (каузальный), 268  
   пробка, 267, 320, 321, 324, 335, 353, 360, 361  
   рекурсивный, 79  
   с бесконечной импульсной характеристикой, 63, 79, 87, 91, 269, 270, 291  
   сглаживания, 338  
   согласованный, 212, 213, 214, 215, 217

с конечной импульсной характеристикой, 62, 63, 69, 70, 74, 76, 269, 270, 291, 373  
 скользящего среднего, 63, 69  
 с лестничной структурой, 291  
 с линейной ФЧХ, 312, 313, 325, 326, 334, 359, 366, 368, 369, 372  
 с нулевой ФЧХ, 344, 345, 346  
 с плавающей запятой, 264, 268, 277, 278, 281, 286, 287, 291, 296, 297, 298, 300  
 с произвольной АЧХ, 326, 328, 329, 335, 353  
 с произвольными склонами АЧХ, 46  
 с равномерными пульсациями АЧХ, 284, 353  
 с фиксированной запятой, 264, 267, 268, 286, 287, 291, 293, 296, 297, 298, 301, 304, 305  
 узкополосный, 127, 331, 333, 334  
 устойчивость, 296  
 формирующий, 215  
 цифровой, 39, 62, 63, 90, 263, 264, 267, 268, 269, 271, 272, 274, 276, 283  
 Чебышева, 83, 84, 86, 272, 274, 281  
 Чебышева II, 83, 84, 85, 272  
 инверсный, 272, 274, 281  
 четвертого порядка, 82  
 широкополосный, 79  
 эллиптический, 83, 86, 87, 272, 281  
 ФНЧ, 22, 23, 25, 26, 27, 33, 66, 67, 72, 76, 78, 79, 81, 90, 91, 92, 156, 271, 272, 276, 281, 283, 284, 310, 313, 318, 326, 327, 329, 333, 334, 337, 338, 341, 343, 354, 361, 362, 364, 365, 368, 369, 372, 374, 418, 449, 452, 453  
 Баттерворта, 83  
 Бесселя, 87, 88  
 идеальный, 26, 55, 74, 190, 429  
 полоса перехода, 342  
 порядок, 342

реальный, 26  
 с линейной ФЧХ, 369  
 с плавающей запятой, 283  
 Чебышева, 84, 86  
 Чебышева II, 85  
 эллиптический, 86  
 ФЧХ, 70, 71, 87, 88, 89, 91, 123, 263, 264, 270, 271, 274, 282, 286, 290, 312, 328, 331, 336, 337, 357, 359, 366, 367, 369, 370, 371, 372  
 линейная, 70, 87, 91, 93, 311, 312, 313, 314, 317, 325, 326, 334, 335, 344, 345, 353, 356, 359, 366, 368, 369, 370, 371, 372, 373  
 нелинейная, 89, 91, 311, 329, 331  
 нулевая, 341, 342, 344, 345, 346

**Ц**

ЦАП, 22, 23, 327  
 цифро-аналоговый преобразователь, 22

**Ч**

частота, 23, 24, 25, 26, 31, 32, 34, 47, 62, 63, 64, 69, 73, 74, 75, 83, 87, 95, 99, 104, 106, 110, 114, 118, 124, 134, 137, 139, 151, 154, 158, 159, 171, 212, 267, 272, 311, 316, 317, 318, 323, 326, 334, 335, 337, 338, 347, 363  
 промежуточная (IF), 37, 55  
 аналогового сигнала, 23  
 битов, 53  
 гармонической составляющей, 109, 170, 174, 176, 271, 335  
 граничная (среза), 69, 76, 83, 177, 178, 179, 181, 343, 351, 374, 375, 449  
 дискретизации, 21, 23, 24, 31, 35, 37, 43, 45, 46, 47, 48, 54, 55, 57, 69, 99, 100, 109, 110, 135, 163, 175, 176, 178, 182, 200, 271, 283, 334, 336, 337, 343, 354

дискретная, 138  
 кратная, 323  
 ложной составляющей, 23  
 ЛЧМ-сигнала, 190  
 мгновенная, 191, 414  
 Найквиста, 25, 103, 154  
 несущая, 33, 38, 46, 55, 212  
 нормализованная, 312  
 передачи ошибочных кодов, 397  
 подавления, 360, 361  
 половинной мощности, 83  
 отрицательная, 35, 104, 105, 110, 114, 119  
 отсчета БПФ, 159  
 резонансная, 127  
 радио, 32, 40, 211  
 символа, 52, 53  
 тактовая, 259, 307, 308  
 центральная, 127  
 частотная характеристика, 76, 78, 79, 83, 84, 85, 120, 121, 122, 123, 125, 126, 160, 268, 282, 355  
 чистый тон, 255

**Ш**

шум, 20, 48, 51, 92, 176, 192, 200, 206, 212, 213, 217, 267, 322, 334, 395, 414  
 AWGN, 219, 429  
 белый, 171, 192, 193, 194, 206, 299, 300, 443, 444, 453, 454  
 гауссовский, 185, 193, 194, 206, 212, 414

дополнительный, 55  
 в системе связи, 48  
 в полосе приема, 48  
 высокочастотный, 92, 338  
 источника электропитания, 269  
 источники, 192  
 канала, 203, 255  
 квантования, 48, 50, 54, 55  
 ложный, 48, 51  
 наложения, 37  
 розовый, 252  
 тепловой, 48, 192, 193  
 уровень, 62, 126  
 фазовый, 395  
 чрезмерный, 123  
 широкополосный, 454  
 шумовой порог, 48, 50, 51, 54, 125, 126

**Э**

энергетический спектр, 446  
 энергия, 154, 357  
 в символе, 53  
 в одном бите, 53  
 полная, 97  
 сигнала, 139  
 составляющей, 137, 152

**Я**

явление Гиббса, 74

Книги издательства «ДМК Пресс» можно заказать в торгово-издательском холдинге «АЛЬЯНС-КНИГА» наложенным платежом, выслав открытку или письмо по почтовому адресу: **123242, Москва, а/я 20** или по электронному адресу: **post@abook.ru**.

При оформлении заказа следует указать адрес (полностью), по которому должны быть высланы книги; фамилию, имя и отчество получателя. Желательно также указать свой телефон и электронный адрес.

Эти книги вы можете заказать и в Internet-магазине: **www.abook.ru**.

Оптовые закупки: тел. **(095) 258-91-94, 258-91-95**; электронный адрес **abook@abook.ru**.

Федосов Валентин Петрович,  
 Нестеренко Андрей Константинович

**Цифровая обработка сигналов в LabVIEW**

Главный редактор *Мовчан Д. А.*  
 dm@dmk-press.ru  
 Корректор *Маньч Э. И., Проценко И. А.*  
 Верстка *Чаннова А. А.*  
 Дизайн обложки *Мовчан А. Г.*

Подписано в печать \_\_.10.2004. Формат 70×100 <sup>1</sup>/<sub>16</sub>.  
 Гарнитура «Петербург». Печать офсетная.  
 Усл. печ. л. \_\_. Тираж \_\_\_\_ экз.  
 №