

Содержание

1 Введение	4
1.1 Задание	4
1.2 Дополнительные требования к заданию	4
1.3 Требования к составным частям имитационной модели	5
1.4 Требования к сопроводительной документации	5
1.5 Требования к оформлению кода	6
2 Варианты заданий	7

Текущая версия документа: 0.1.151 от 13 октября 2014 г.

1 Введение

Данная лабораторная работа рассчитана на четыре занятия. Задание выполняется бригадой численностью от 5 до 8 человек (как правило, из учебной группы формируется две бригады). В каждой бригаде по итогам голосования назначается руководитель проекта, который отвечает за распределение обязанностей между студентами в бригаде. Руководителя проекта можно переизбирать в процессе выполнения лабораторных работ, для чего нужно в письменной форме уведомить преподавателя.

Цель лабораторной работы — закрепить навыки программирования на языке Python, освоить систему GitHub для контроля версий ПО с открытым программным кодом и поучить навыки коллективной разработки программного обеспечения на примере разработки программы имитационного моделирования радиосистемы передачи данных.

1.1 Задание

Составить при помощи ЭВМ имитационную модель цифровой радиосистемы системы передачи информации, которая должна включать в себя следующие составные части:

- а) Источник информации.
- б) Модуль сжатия исходной информации.
- в) Модуль помехоустойчивого кодирования.
- г) Модуль модуляции.
- д) Модуль наложения шума и помех.
- е) Модуль демодуляции.
- ж) Модуль декодирования.
- з) Модуль восстановления исходной информации.
- и) Модуль статистики.
- к) Графический интерфейс.
- л) Сопроводительная документация.

1.2 Дополнительные требования к заданию

Каждый модуль программы должен быть реализован в виде отдельной процедуры или функции к которому прилагаются тестовые примеры для проверки работоспособности. В графическом интерфейсе программы должна быть заложена возможность визуального контроля сигналов или информационной последовательности на каждом этапе выполнения программы (построения графиков по средствам меню или кнопок на форме).

В качестве языка программирования разрешено использовать только Python и C++. Для контроля версий ПО обязательно использовать систему GitHub. Для составления сопроводительной документации обязательно использовать текстовый редактор LaTeX.

1.3 Требования к составным частям имитационной модели

- а) Источник информации должен генерировать случайную последовательность, состоящую из 512 двоичных символов.
- б) Модуль сжатия информации должен устранять избыточность из исходной последовательности и вычислять коэффициент сжатия.
- в) В модуле помехоустойчивого кодирования требуется реализовать несколько алгоритмов в соответствии с вариантом задания. Выбор алгоритма помехоустойчивого кодирования должен осуществляться пользователем по средствам графического интерфейса.
- г) Методы модуляции и их параметры выбираются в соответствии с вариантом задания. Все параметры должны отображаться при помощи графического интерфейса, кроме того, у пользователя должна быть возможность оперативно изменять параметры модуляции. При использовании нескольких методов модуляции требуется выбор конкретной модуляции по средствам меню или графического интерфейса.
- д) Требования к шумам и помехам указаны в варианте задания. Необходимо предусмотреть возможность оперативного изменения параметров шума или помехи по средствам графического интерфейса.
- е) Модуль демодуляции должен соответствовать используемой модуляции.
- ж) Процедура декодирования должна соответствовать процедуре кодирования.
- з) Процедура восстановления исходной информации должна соответствовать процедуре сжатия.
- и) Модуль статистики должен вычислять вероятность ошибки в символе в зависимости от уровня шума и наличия помех на основе n выборок исходной информационной последовательности, количество которых должно задаваться при помощи графического интерфейса.

1.4 Требования к сопроводительной документации

Сопроводительная документация должна быть набрана в системе компьютерной верстки \LaTeX и поставляться в виде исходных `.tex` файлов и готового файла формата `pdf`. В документации должно быть:

- а) Описание программы
- б) Описание отдельных модулей, классов и функций, входящих в ее состав
 - 1) Назначение класса, модуля, функции
 - 2) Входные и выходные данные
 - 3) Параметры конструктора
 - 4) Пример использования в коде

- в) Описание установки необходимых для сборки файлов и библиотек
- г) Процедура сборки и запуска программы
- д) Скриншоты работы программы
- е) Любая другая, необходимая по вашему мнению, информация

Документация должна располагаться в папке `doc` в корне папки вашего проекта.

1.5 Требования к оформлению кода

Весь код должен быть набран в едином стиле, с едиными размерами отступов, имена переменных должны быть понятны и осмысленны. Неочевидные участки кода и все интерфейсы закомментированы.

Для оформления кода используйте общепринятые стандарты кодирования. PEP8 - для Python или *Google C++ Style Guide*, *Linux kernel coding style*, *Mozilla C++ Portability Guide*, *WebKit Coding Style Guidelines*. в случае C++ кода. Допускается в определенных случаях не строгое следования стандарту, однако в пределах бригады все участники должны придерживаться одного набора правил.

2 Варианты заданий

Во всех вариантах использовать дискретный бинарный источник информации (равновероятное возникновение нулей и единиц). Длина последовательности 512 символов. Модуль сжатия исходной информации используется на усмотрение студентов (использование не обязательно). В канале связи присутствует только аддитивный белый гауссовский шум. Мощность сигнала и шума выбирать таким образом, чтобы отношение сигнал/шум. Остальные требования указаны в таблице.

Номер варианта	Методы кодирования	Кратность исправляемой ошибки	Виды модуляции
1	код Рида-Саломона и БЧХ-код	5	QAM-128 и PM-16
2	код Хемминга и БЧХ-код	7	QAM-256 и QPSK