



ИССЛЕДОВАНИЕ МЕТОДОВ ПОСТРОЕНИЯ МОДУЛЬНОЙ СИСТЕМЫ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ СИСТЕМ ПЕРЕДАЧИ ДАННЫХ С ПОМЕХОУСТОЙЧИВЫМ КОДИРОВАНИЕМ

С. С. Владимиров¹, А. А. Самайданов¹

¹ СПбГУТ, Санкт-Петербург, 193232, Российская Федерация
Адрес для переписки: vladimirov.opds@gmail.com

Информация о статье

УДК 004.942

Язык статьи – русский.

Поступила в редакцию 18.12.15, принята к печати 20.01.16.

Ссылка для цитирования: Владимиров С. С., Самайданов А. А. Исследование методов построения модульной системы имитационного моделирования систем передачи данных с помехоустойчивым кодированием // Информационные технологии и телекоммуникации. 2016. Том 4. № 1. С. 9–16.

Аннотация

Предмет исследования. Статья посвящена исследованию методов построения модульной системы имитационного моделирования систем передачи данных с помехоустойчивым кодированием. **Метод.** В качестве метода исследования применяется компьютерное моделирование. В работе оценивается быстродействие программ, построенных по рассмотренным методам построения модульных систем имитационного моделирования. **Основные результаты.** В статье приведены результаты оценки быстродействия программ, построенных по рассмотренным методам, для одноядерных и многоядерных ЭВМ. **Практическая значимость.** Приведены рекомендации по выбору метода построения системы имитационного моделирования для использования на многоядерных ЭВМ под управлением POSIX-совместимых операционных систем.

Ключевые слова

система моделирования, имитационное моделирование, помехоустойчивое кодирование, сокет, именованный канал.

THE RESEARCH OF ECC DATA TRANSMISSION SYSTEMS MODULAR SIMULATION MODELLING PROGRAM DESIGN METHODS

S. Vladimirov¹, A. Samaydanov¹

¹ SPbSUT, St. Petersburg, 193232, Russian Federation
Corresponding author: vladimirov.opds@gmail.com

Article info

Article in Russian.

Received 18.12.15, accepted 20.01.16.

For citation: Vladimirov S., Samaydanov A.: The Research of ECC Data Transmission Systems Modular Simulation Modelling Program Design Methods // Telecom IT. 2016. Vol. 4. N 1. pp. 9–16 (in Russian).

Abstract

Research subject. The article concentrates on the research of ECC data transmission systems modular simulation modelling program design methods. **Method.** The research method is computer modelling. In the article we estimate the execution time of programs based on described design methods. **Core results.** The paper contains the results of calculated efficiency of examined methods, which has been estimated with execution time of programs based on described design methods for computers with single-core and multi-core architectures. **Practical relevance.** The paper presents recommendations of choosing the simulation modelling program design method for multi-core computers powered by POSIX operation.

Keywords

Modelling program, simulation modelling, error-correcting coding, socket, named pipe.

Введение

В современном мире для систем передачи, обработки и хранения информации всё шире применяются различные методы помехоустойчивого кодирования. Одним из основных методов, используемых современными инженерами и учёными при исследовании и разработке помехоустойчивых кодеров и декодеров, является имитационное моделирование. В частности, широко используются системы компьютерной алгебры, такие как Matlab и GNU/Octave, в которых реализованы модели многих элементов типичной системы передачи данных, таких как модуляторы и демодуляторы, кодеры и декодеры наиболее распространенных помехоустойчивых кодов, модели цифровых и аналоговых каналов передачи данных. Недостатком этих систем является их невысокое быстродействие, что зачастую не позволяет проводить исследование методом набора статистики. Особенно это заметно при исследовании новых моделей, написанных самим исследователем на внутреннем языке программирования этих систем компьютерной алгебры. Другим подходом является написание исследователем своих моделей на том или ином языке программирования. Основным недостатком такого подхода является частая необходимость заново реализовывать отдельные элементы систем передачи данных, таких как модели каналов или модуляторы/демодуляторы. Далеко не все языки программирования позволяют удобно под-



ключать существующие библиотеки с реализованными подпрограммами имитационных моделей. К тому же, усложняется совместная работа нескольких исследователей в том случае, когда они используют различные языки программирования.

На кафедре сетей связи и передачи данных СПбГУТ перед авторами была поставлена задача разработки программного комплекса для имитационного моделирования систем передачи данных. Главной задачей, которую решает разрабатываемая система моделирования является подбор оптимальных методов модуляции и помехоустойчивого кодирования для различных каналов передачи данных, которые реализуются в виде программных моделей.

Система имитационного моделирования должна обладать рядом свойств.

- Мультиязычность – данное требование подразумевает, что в системе для написания отдельных элементов модели возможно использовать несколько языков программирования. Из этого свойства вытекает следующее.

- Модульность – модель системы передачи, составляется как совокупность модулей, реализованных в виде отдельных программ – элементов системы передачи (источник, кодер, декодер и другие).

- Расширяемость – возможность добавления новых программ-модулей, которые не будут затрагивать уже имеющиеся модули.

- Настраиваемость – имеющиеся модули можно гибко конфигурировать (в зависимости от их реализации), а сами модули соединять с другими по единому интерфейсу, образуя полноценные модели систем передачи данных.

Эти свойства позволяют интегрировать в неё уже имеющиеся на кафедре разработки [1, 2] при внесении в них необходимых изменений. Основными операционными системами, для которых реализуется разрабатываемая система моделирования, являются используемые на кафедре POSIX-совместимые операционные системы GNU/Linux и BSD Unix.

Основные методы построения модульных программных комплексов для моделирования систем передачи данных

Заданное при разработке программы свойство модульности программного комплекса для моделирования подразумевает выбор способа взаимодействия между отдельными модулями-программами в программном комплексе. В POSIX-совместимых операционных системах существуют различные методы взаимодействия программ между собой, так называемые методы межпроцессного взаимодействия – Inter-process Communication (IPC). К основным методам можно отнести взаимодействие с использованием сокетов и взаимодействие через каналы или через именованные каналы (каналы FIFO) [3, 4]. Кратко рассмотрим основные особенности этих методов.

Под сокетом понимается программный интерфейс для организации обмена данными между различными процессами, которые, в зависимости от типа сокета, могут быть запущены на одной ЭВМ (UNIX-сокеты), либо разных, но связанных между собой сетью по протоколу TCP/IP (INET-сокеты). При взаимодействии двух программ через сокет различают серверный сокет, который ожидает соединения, и клиентский сокет, который инициализирует соединение с серверным сокетом и передает в него данные [3]. Общая схема взаимодействия двух программ через сокеты представлена на рис. 1.



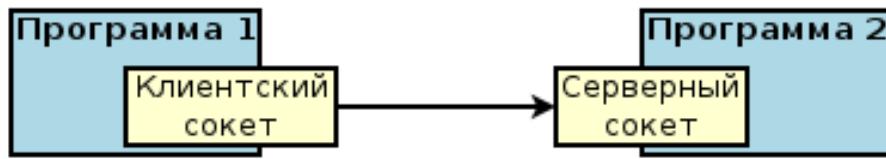


Рис. 1. Взаимодействие программ через сокеты

Для построения модульной системы моделирования, предназначенной для работы на одной ЭВМ, необходимо использовать UNIX-сокеты. Данные сокеты для адресации используют файловую систему как пространство имен, т. е. представляются в виде файлов с конкретным именем. Это позволяет двум различным процессам открыть один и тот же сокет и использовать его для взаимодействия между собой.

Ключевой особенностью при организации обмена данными между модулями-программами посредством сокетов является соблюдение последовательности запуска программ. Одна из программ должна работать как сервер, создавая серверный сокет и ожидая подключения, а вторая – как клиент, создавая клиентский сокет и подключаясь к серверному сокету [3]. В случае, если серверный сокет отсутствует, программа-клиент будет вынуждена либо завершить работу, либо ожидать запуска сервера, периодически проверяя наличие серверного сокета. Диаграмма последовательности работы программ при взаимодействии через сокеты показана на рис. 2 [3].

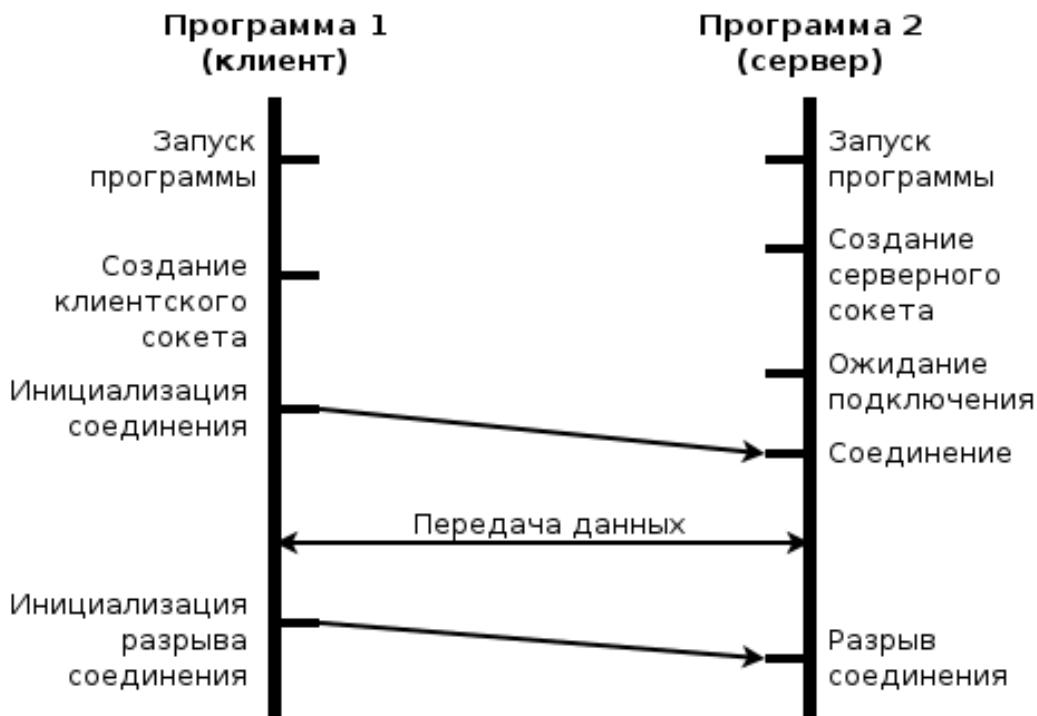


Рис. 2. Диаграмма последовательности работы программ при взаимодействии через сокеты

Другим механизмом организации межпроцессного взаимодействия являются именованные каналы или, как их часто называют, каналы FIFO (*First In First Out*).



Канал FIFO является односторонним средством передачи данных, причем данныечитываются в том же порядке, в котором были записаны [3]. Именованный канал представляет из себя специальный файл, который может быть открыт для чтения или записи [3, 4].

Программы, использующие файл именованного канала, условно называются производителем и потребителем. Работа с именованными каналами осуществляется с помощью простого алгоритма.

1. Создание файла именованного канала, еще до его непосредственного использования какой-либо программой.

2. Потребитель открывает файл для чтения, а производитель открывает файл для записи. При этом, если первым к каналу подключился потребитель, то он ожидает подключения производителя и начала записи данных в канал [3].

3. Когда обе программы открыли файл – одна для чтения, другая для записи – производитель может писать в файл данные, а потребитель считывать их.

Диаграмма последовательности работы программ при взаимодействии через каналы FIFO показана на рис. 3.



Рис. 3. Диаграмма последовательности работы программ при взаимодействии через именованный канал

К преимуществам построения модульной системы моделирования на основе именованных каналов можно отнести простой механизм работы – как с файлами; создание файлов-каналов вне использующих их программ, что позволяет просто организовывать работу модели, предварительно создавая все необходимые связи между отдельными модулями-программами; отсутствие клиента и сервера – режим работы файлов-каналов является асинхронным, что позволяет запускать программы-модули в любом порядке.

При этом необходимо отметить, что при построении системы моделирования, состоящей из отдельных модулей-программ, вне зависимости от используемого метода межпроцессного взаимодействия, необходимо соблюдать единый формат сообщений, которыми обмениваются модули-программы, связываемые между собой.



Моделирование исследуемых методов

Для исследования рассматриваемых методов построения модульной системы моделирования и их оценки по быстродействию была написана программная модель на языке Си с использованием компилятора GCC. Блок-схема программной модели представлена на рис. 4.

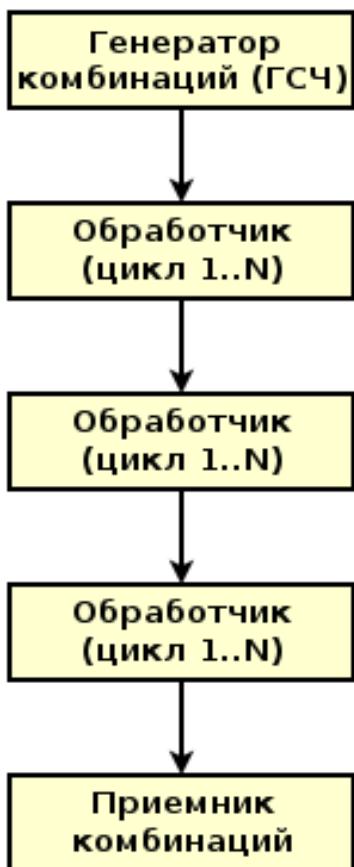


Рис. 4. Блок-схема программной модели оценки быстродействия методов построения модульной системы моделирования систем передачи данных с помехоустойчивым кодированием

Модель состоит из источника сообщений, реализованного по принципу генератора случайных чисел; трех модулей-обработчиков, осуществляющих обработку комбинации в цикле из N шагов; приемника-заглушки, получающего обработанные комбинации.

Исследование быстродействия проведено для метода построения системы моделирования на основе сокетов и для метода на основе именованных каналов. Для сравнения оценено быстродействие аналогичной по функционалу модели, построенной в виде одной программы.

Результаты исследования показаны на рис. 5.

Как следует из графиков, при работе модели на ЭВМ с одноядерным процессором быстродействие модели, реализованной в виде одной программы, превышает быстродействие моделей, построенных с использованием сокетов и именованных каналов. Это связано с наличием дополнительных операций по обеспечению взаимодействия между отдельными модулями-программами в модели, которые выполняются на одном ядре наравне с процедурами расчета в циклах. То есть, модель выполняется последовательно, независимо от используемой архитектуры.

Из графиков для двух- и четырехядерных ЭВМ видно, что в этом случае быстродействие моделей, построенных на основе сокетов и именованных каналов примерно на 40 % превышает быстродействие модели, построенной в виде одной программы. Это обуславливается тем, что отдельные модули-программы, составляющие модель на основе сокетов или именованных каналов, выполняются параллельно на разных ядрах процессора. При этом выигрыш из-за распараллеливания вычислений превышает накладные временные расходы на передачу данных между модулями-программами. При этом быстродействие систем, построенных на основе сокетов и именованных каналов, отличается незначительно.

Подводя итог, можно сделать вывод, что для построения модульной системы имитационного моделирования систем передачи данных с помехоустойчивым кодированием, предназначенных для работы на многоядерных ЭВМ под управле-



нием POSIX-совместимых операционных систем лучше использовать метод межпроцессного взаимодействия на основе именованных каналов, поскольку при этом обеспечивается высокая производительность наряду с простотой разработки моделей.

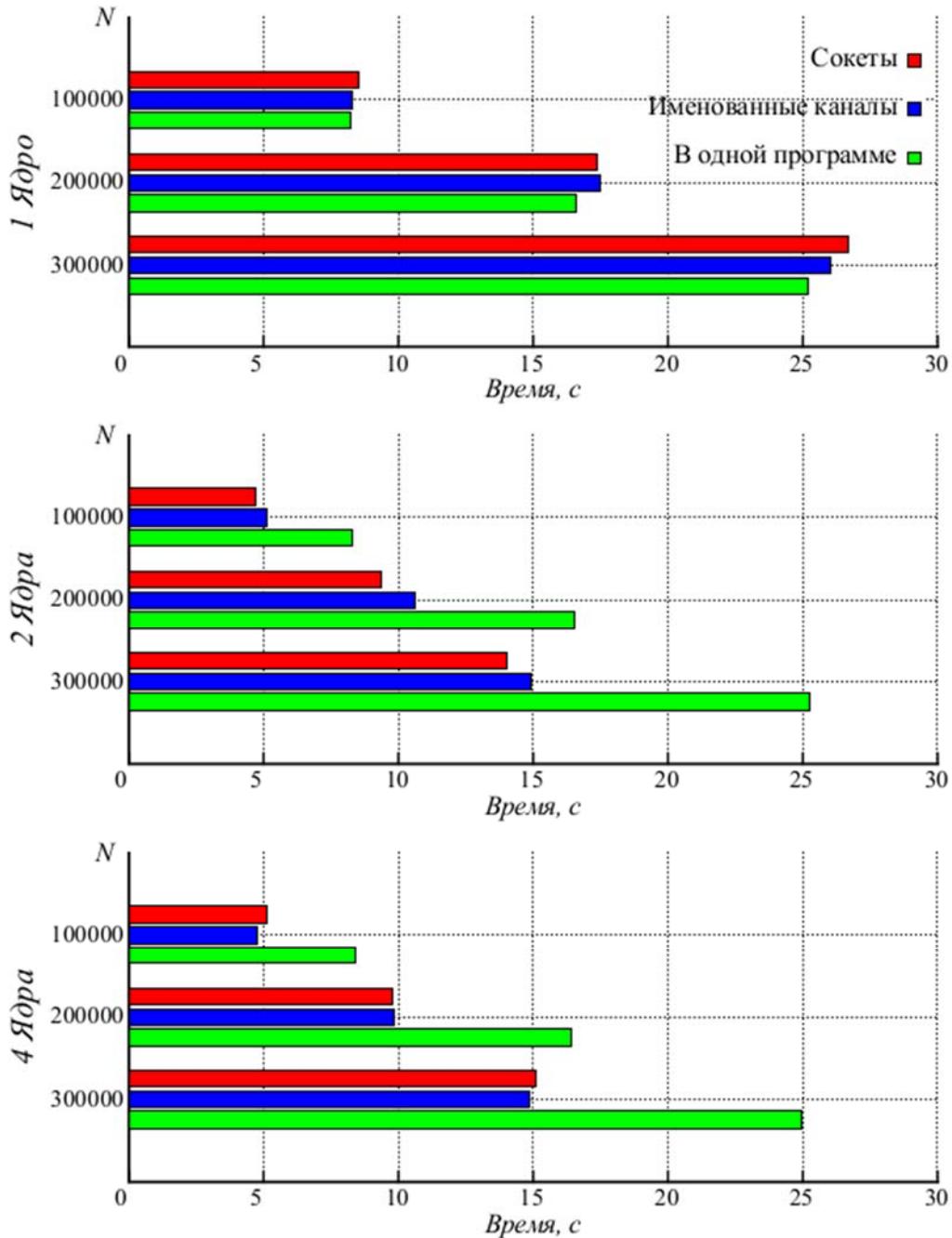


Рис. 5. Результаты исследования быстродействия методов построения модульной системы моделирования систем передачи данных с помехоустойчивым кодированием

Литература

- Когновицкий О. С. Алгоритмы декодирования циклических кодов в частотной области на основе преобразований Фурье-Мэттсона-Соломона и на основе двойственного базиса // Системы управления и информационные технологии. 2015. Т. 59. № 1.1. С. 148–153.



2. Владимиров С. С. Реализация четырехпараметрической модели канала Гилберта-Эллиotta в среде Octave/Matlab // Информационные технологии и телекоммуникации. 2014. № 4 (8). С. 88–96.
3. Робачевский А. М., Немнюгин С. А., Стесик О. Л. Операционная система UNIX. СПб. : БХВ-Петербург, 2005. 656 с. ISBN 978-5-94157-538-1.
4. Stevens W. Richard UNIX Network Programming: Interprocess communications, Volume 2, Second Edition. Prentice Hall PTR, 1999. 558 p.

References

1. Kognovickiy O.S. Algorithms for Decoding Cyclic Codes in the Frequency Domain Based on the Fourier-Solomon-Mattson Transformations and Based on the Dual Basis // Sistemy upravleniya I informacionnye tekhnologii. 2015. vol. 59. no. 1.1. pp. 148–153.
2. Vladimirov S.S. The Implementation of Four-Parameter Gilbert-Elliott Channel Model with Octave/Matlab // Telecom IT. 2014. no. 4 (8). pp. 88–96.
3. Robachevskij A.M., Nemnyugin S.A., Stesik O.L. Operation System UNIX (Operacionnaya sistema UNIX). SPb: BHV-Peterburg, 2005. 656 p. ISBN 978-5-94157-538-1.
4. Stevens W. Richard UNIX Network Programming: Interprocess Communications, Volume 2, Second Edition. Prentice Hall PTR, 1999. 558 p.

Владимиров Сергей Сергеевич

– кандидат технических наук, доцент, СПбГУТ, Санкт-Петербург, 193232, Российская Федерация, vladimirov.opds@gmail.com

Самайданов

Александр Алексеевич

– студент, СПбГУТ, Санкт-Петербург, 193232, Российская Федерация, samaydanov@gmail.com

Vladimirov Sergey

– Ph.D., assistant professor, SPbSUT, St. Petersburg, 193232, Russian Federation, vladimirov.opds@gmail.com

Samaydanov

Alexander Alekseevich

– student, SPbSUT, St. Petersburg, 193232, Russian Federation, samaydanov@gmail.com

