

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

К защите допустить:

Заведующий кафедрой ПОИТ

_____ Н. В. Лапицкая

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к дипломному проекту

на тему

**ПРОГРАММНОЕ СРЕДСТВО
УПРАВЛЕНИЯ ДОКУМЕНТООБОРОТОМ МАЛОГО ПРЕДПРИЯТИЯ**

БГУИР ДП 1-40 01 01 03 110 ПЗ

Студент	В. М. Сидоренко
---------	-----------------

Руководитель	Н. П. Журавский
--------------	-----------------

Консультанты:	А. А. Ктотов
<i>от кафедры ПОИТ</i>	Г. А. Харитонов
<i>по экономической части</i>	Л. А. Корбут
<i>по охране труда</i>	П. Ю. Бранцевич

Нормоконтролер	И. П. Иванов
----------------	--------------

Рецензент	
-----------	--

Минск 2014

РЕФЕРАТ

Пояснительная записка 78 с., 24 рис., 5 табл., 29 источников
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, КОНСТРУКЦИЯ, ВИБРАЦИЯ,
СИГНАЛ, ИЗМЕРИТЕЛЬНО-ВЫЧИСЛИТЕЛЬНЫЙ КОМПЛЕКС

Объектом исследования является математическое и программное обеспечение систем для проведения испытания конструкций зданий и сооружений на динамические воздействия.

Цель работы - разработка программного средства для обработки результатов, полученных с помощью измерительно-вычислительного комплекса «Тембр».

Разработка данного программного средства обеспечит формирование базы данных по отдельным конструктивным элементам с хранением сведений о их конструктивных параметрах, результатах экспериментальных исследований, расчетных и экспериментально полученных значениях собственных частот.

Проведен анализ методов обработки вибрационных сигналов, получаемых при испытаниях конструкций, способов определения собственных частот конструкций по экспериментальным данным.

Рассмотрены математические модели определения собственных частот конструкций и расчета жесткости и прогибов конструкций на основе значений собственных частот.

Проведены экспериментальные исследования по определению собственных частот ряда конструкций при их возбуждении динамическим воздействием.

Разрабатываемое программное обеспечение должно стать элементом системы автоматизации процедуры принятия решений по оценке остаточной устойчивости и жесткости зданий и сооружений.

Министерство образования Республики Беларусь
Учреждение образования
**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ
И РАДИОЭЛЕКТРОНИКИ**

Факультет КС и С Кафедра ПОИТ
Специальность 1-40 01 01 Специализация 03

УТВЕРЖДАЮ

Н.В.Лапицкая

« » 20 г.

ЗАДАНИЕ
по дипломному проекту студента

Кротовича Александра Михайловича

(фамилия, имя, отчество)

1. Тема проекта: **Программное средство управления документооборотом
малого предприятия**

утверждена приказом по университету от « XX » xxxxxx 20 г. № XX-с

2. Срок сдачи студентом законченной работы 01 июня 2013 года

3. Исходные данные к проекту Тип операционной системы – ОС Windows XP;
Язык программирования – Visual C++; Перечень выполняемых функций:

Назначение разработки: автоматизация процессов подготовки отчетно-справочной доку-
ментации малого предприятия

4. Содержание пояснительной записки (перечень подлежащих разработке вопросов)

Введение

1 Анализ программных систем управления документооборотом

2 Моделирование предметной области и разработка функциональных требований

3 Проектирование программного средства

4 Создание программного средства

5 Тестирование программного средства

6 Руководство по установке и использованию

7 Техничко-экономическое обоснование

8 Охрана труда

Заключение

Список использованных источников

Приложение А Текст программы

Приложение Б Иллюстрации работы программы

5. Перечень графического материала (с точным указанием наименования) и обозначения вида и типа материала)

Организация системы документооборота. Плакат — формат А1, лист 1.

Диаграмма UML программного средства. Плакат - формат А1, лист 1.

Диаграмма классов программного средства. Плакат - формат А1, лист 1.

Программа обработки данных. Схема программы - формат А1, лист 1.

База данных интернет-магазина. Схема данных - формат А1, лист 1.

Модуль шифрования. Схема программы - формат А1, лист 1.

6. Содержание задания по технико-экономическому обоснованию

Расчет экономической эффективности от внедрения программного средства

Задание выдал _____ / Т.Л. Синкевич /

7. Содержание задания по производственной и экологической безопасности

Расчет освещенности рабочего места программиста

Задание выдал: _____ / Л.А.Петров /

КАЛЕНДАРНЫЙ ПЛАН

Наименование этапов дипломного проекта (работы)	Объём этапа в %	Срок вы- полнения этапа	Примечание
Анализ предметной области, разработка технического задания	15-20	03.02–16.02	
Разработка функциональных требований, проектирование архитектуры программы	20-15	18.02–07.03	
Разработка схемы программы, алгоритмов, схемы данных	20-15	11.03–22.03	
Разработка программного средства	15-20	25.03–26.04	
Тестирование и отладка	10	29.04–10.05	
Оформление пояснительной записки и графического материала	20	13.05–31.05	

Дата выдачи задания XX xxxxxx 2014 г. Руководитель _____ /А.Я. Ковин/

Задание принял к исполнению _____ / А.М. Кротович /

СОДЕРЖАНИЕ

Введение	7
1 Анализ современного состояния сетевых технологий	8
1.1 Основные направления развития	8
1.2 Вычислительные сети как частный случай распределенных систем	
1.3 Понятие «открытая система» и проблемы стандартизации	
2 Разработка структуры программных средств.....	
2.1 Подраздел первый	
2.2 Подраздел второй	
2.3 Подраздел третий	
3 Разработка структуры данных.....	
3.1 Подраздел первый	
3.2 Подраздел второй	
3.3 Подраздел третий	
4 Экспериментальные исследования программы моделирования	
4.1 Подраздел первый	
4.2 Подраздел второй	
4.3 Подраздел третий	
5 Техничко-экономическое обоснование	
5.1 Подраздел первый	
5.2 Подраздел второй	
5.3 Подраздел третий	
6 Организация освещенности на рабочем месте программиста	
6.1 Подраздел первый	
6.2 Подраздел второй	
6.3 Подраздел третий	
Заключение	
Список использованных источников	
Приложение А Результаты моделирования алгоритма определения частоты сигнала	
Приложение Б Текст программного модуля обработки сообщений	

ОПРЕДЕЛЕНИЯ И СОКРАЩЕНИЯ

В настоящей пояснительной записке применяются следующие определения и сокращения.

Алгоритмический язык - искусственный язык, предназначенный для выражения алгоритмов.

Инициализация - приведение областей памяти в состояние, исходное для последующей обработки или размещения данных.

Программа - данные, предназначенные для управления конкретными компонентами системы обработки информации в целях реализации определенного алгоритма.

Программное обеспечение - совокупность программ системы обработки информации и программных документов, необходимых для эксплуатации этих программ.

Программирование - научная и практическая деятельность по созданию программ.

Прикладная программа - программа, предназначенная для оказания услуг общего характера пользователям и обслуживающему персоналу системы обработки информации.

Программный модуль - программа или функционально завершенный фрагмент программы, предназначенный для хранения, трансляции, объединения с другими программными модулями и загрузки в оперативную память.

Подпрограмма - программа, являющаяся частью другой программы и удовлетворяющая требованиям языка программирования к структуре программы.

Процесс обработки данных - система действий, реализующая определенную функцию в системе обработки информации и оформленная так, что управляющая программа данной системы может перераспределять ресурсы этой системы в целях обеспечения мультипрограммирования.

Программный документ - документ, содержащий в зависимости от назначения данные, необходимые для разработки, производства, эксплуатации сопровождения программы или программного средства.

Программно-аппаратные средства (firm-ware) - технические средства, содержащие компьютерную программу и данные, которые не могут изменяться средствами пользователя. Компьютерная программа и данные, входящие в программно-аппаратные средства, классифицируются как программное обеспечение; схемы, содержащие компьютерную программу и данные, классифицируются как технические средства.

Программное обеспечение (software) - программы, процедуры, правила и любая соответствующая документация, относящиеся к работе вычислительной системы.

Системная программа - программа, предназначенная для поддержания работоспособности системы обработки информации или повышения эффективности ее использования в процессе выполнения прикладных программ.

Система программирования - система, образуемая языком программирования компиляторами или интерпретаторами программ, представленных на этом языке, соответствующей документацией, а также вспомогательными средствами для подготовки программ к форме, пригодной для выполнения.

Спецификация программы - формализованное представление требований, предъявляемых к программе, которые должны быть удовлетворены при ее разработке, а также описание задачи, условия и эффекта действия без указания способа его достижения.

АСУ ТП - автоматизированная система управления технологическим процессом

АЦП - аналого-цифровой преобразователь

СКЗ - среднее квадратическое значение

СУБД - система управления базой данных

ЭВМ - электронная вычислительная машина

ARP- address resolution protocol (протокол разрешения адресов)

ATM - asynchronous transfer mode (асинхронный режим передачи))

ВВЕДЕНИЕ

Информационные и управляющие системы в настоящее время составляют основную область приложений компьютерных систем во многих видах производственной, финансовой, творческой деятельности, а также в образовании, фундаментальных и прикладных научных исследованиях.

Принцип работы информационных и управляющих систем основан на создании встроенных моделей предметной области, общих моделей контролируемых объектов пользователя в этой области и их частных экземпляров, отражающих конкретные свойства объектов.

Накопление моделей и ситуаций в виде некоторой информационной базы позволяет более глубоко оценивать складывающуюся обстановку и принимать решения о выработке требуемых воздействиях на объекты и ситуацию, то есть об управляющих воздействиях на окружение человека или объекты его производственной или иной деятельности. Эта связь между информационной и управляющей частью компьютерных систем естественна и неразрывна. Различия между ними в основном определяются уровнем автоматизации накопления информации и принятия решений в конкретных системах [1].

1 АНАЛИЗ СОВРЕМЕННОГО СОСТОЯНИЯ СЕТЕВЫХ ТЕХНОЛОГИЙ

1.1 Основные направления развития

Сегодня вычислительные сети продолжают развиваться, причем достаточно быстро. Разрыв между локальными и глобальными сетями постоянно сокращается во многом из-за появления высокоскоростных территориальных каналов связи, не уступающих по качеству кабельным системам локальных сетей. В глобальных сетях появляются службы доступа к ресурсам, такие же удобные и прозрачные, как и службы локальных сетей. Подобные примеры в большом количестве демонстрирует самая популярная глобальная сеть - Internet.

Изменяются и локальные сети. Вместо соединяющего компьютеры пассивного кабеля в них в большом количестве появилось разнообразное коммуникационное оборудование - коммутаторы, маршрутизаторы, шлюзы. Благодаря такому оборудованию появилась возможность построения больших корпоративных сетей, насчитывающих тысячи компьютеров и имеющих сложную структуру. Возродился интерес к крупным компьютерам - в основном из-за того, что после спада эйфории по поводу легкости работы с персональными компьютерами выяснилось, что системы, состоящие из сотен серверов, обслуживать сложнее, чем несколько больших компьютеров. Поэтому на новом витке эволюционной спирали мэйнфреймы стали возвращаться в корпоративные вычислительные системы, но уже как полноправные сетевые узлы, поддерживающие Ethernet или Token Ring, а также стек протоколов TCP/IP, ставший благодаря Internet сетевым стандартом де-факто.

Таблица 1.1 - Схема маршрутизации маршрутизатора

Номер сети назначения	Сетевой адрес следующего маршрутизатора	Сетевой адрес выходного порта	Расстояние до сети назначения
51	M1(2)	M4(1)	1
52	-	M4(1)	0 (подсоединена)
53	M1(2)	M4(1)	1
54	M2(1)	M4(1)	1

Проявилась еще одна очень важная тенденция, затрагивающая в равной степени как локальные, так и глобальные сети. В них стала обрабатываться несвойственная ранее вычислительным сетям информация - голос, видеоизображения, рисунки. Это потребовало внесения изменений в работу протоколов, сетевых операционных систем и коммуникационного оборудования. Сложность передачи такой мультиме-

дийной информации по сети связана с ее чувствительностью к задержкам при передаче пакетов данных - задержки обычно приводят к искажению такой информации в конечных узлах сети. Так как традиционные службы вычислительных сетей - такие как передача файлов или электронная почта - создают малочувствительный к задержкам трафик и все элементы сетей разрабатывались в расчете на него, то появление трафика реального времени привело к большим проблемам.

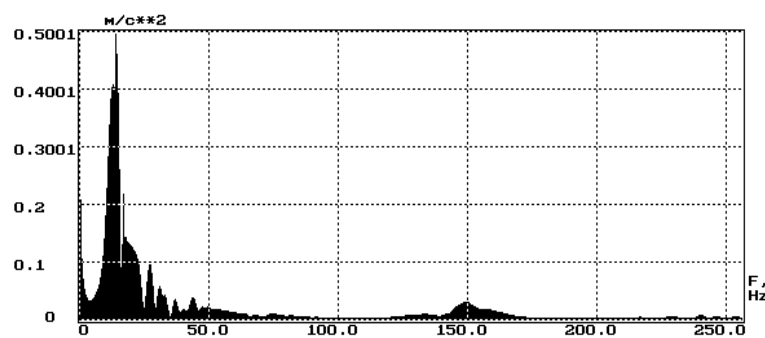
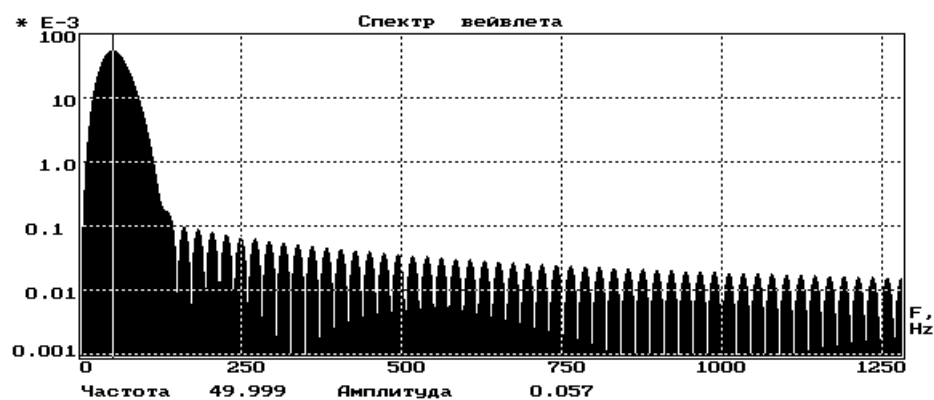


Рисунок 1.1 – Спектр виброударного импульса

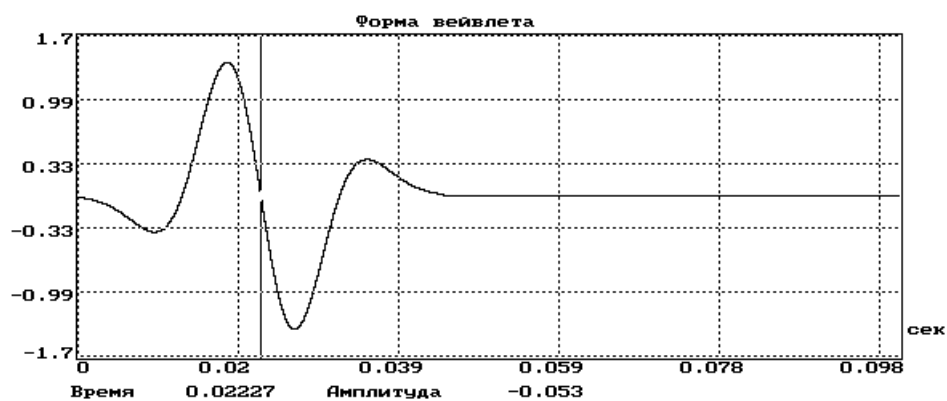
Сегодня эти проблемы решаются различными способами, в том числе и с помощью специально рассчитанной на передачу различных типов трафика технологии АТМ (Asynchronous Transfer Mode (асинхронный режим передачи)). Однако, несмотря на значительные усилия, предпринимаемые в этом направлении, до приемлемого решения проблемы пока далеко, и в этой области предстоит еще много сделать, чтобы достичь заветной цели - слияния технологий не только локальных и глобальных сетей, но и технологий любых информационных сетей - вычислительных, телефонных, телевизионных и т. п. Хотя сегодня эта идея многим кажется утопией, серьезные специалисты считают, что предпосылки для такого синтеза уже существуют, и их мнения расходятся только в оценке примерных сроков такого объединения - называются сроки от 10 до 25 лет. Причем считается, что основой для объединения послужит технология коммутации пакетов, применяемая сегодня в вычислительных сетях, а не технология коммутации каналов, используемая в телефонии, что, наверно, должно повысить интерес к сетям этого типа [2].

При поиске параметров модели ставится целью минимизация данного критерия. Таким образом, требуется решить оптимизационную задачу. Однако параметры модели связаны между собой неформализованными зависимостями, что существенно усложняет процесс поиска решения.

В качестве параметра, характеризующего скорость затухания возбужденных в конструкции колебаний, используют логарифмический декремент затухания колебаний.



а) частотная характеристика вейвлета



б) форма вейвлета

Рисунок 1.2 - Частотная характеристика и форма гауссова вейвлета 3-го порядка с центральной частотой 50 Гц

По форме результат испытания на динамическое воздействие - затухающее колебание, представляющее собой комбинацию затухающих колебаний нескольких частотных составляющих. Для последующей оценки технических характеристик испытуемой конструкции требуется определить значения собственных частот и логарифмические декременты затухания колебаний на этих частотах.

Применение дискретного преобразования Фурье для определения амплитудного спектра такого сигнала на конечном временном интервале позволяет получить только приблизительные значения собственных частот (частоты, выраженных по

амплитуде, спектральных линий).

1.2 Вычислительные сети как частный случай распределенных систем

1.2.1 Связь с распределенными системами

Компьютерные сети относятся к распределенным (или децентрализованным) вычислительным системам. Поскольку основным признаком распределенной вычислительной системы является наличие нескольких центров обработки данных, то наряду с компьютерными сетями к распределенным системам относят также мультипроцессорные компьютеры и многомашинные вычислительные комплексы

1.2.2 Мультипроцессорные компьютеры

В мультипроцессорных компьютерах имеется несколько процессоров, каждый из которых может относительно независимо от остальных выполнять свою программу. В мультипроцессоре существует общая для всех процессоров операционная система, которая оперативно распределяет вычислительную нагрузку между процессорами. Взаимодействие между отдельными процессорами организуется наиболее простым способом - через общую оперативную память.

1.3 Понятие «открытая система» и проблемы стандартизации

Универсальный тезис о пользе стандартизации, справедливый для всех отраслей, в компьютерных сетях приобретает особое значение. Сети - это соединение разного оборудования, а значит, проблема совместимости является одной из наиболее острых. Без принятия всеми производителями общепринятых правил построения оборудования прогресс в распространении сетей был бы невозможен. Поэтому все развитие компьютерной отрасли в конечном счете отражено в стандартах - любая новая технология только тогда приобретает «законный» статус, когда ее содержание закрепляется в соответствующем стандарте.

В компьютерных сетях идеологической основой стандартизации является многоуровневый подход к разработке средств сетевого взаимодействия. Именно на основе этого подхода была разработана стандартная семиуровневая модель взаимодействия открытых систем, ставшая своего рода универсальным языком сетевых специалистов.

2 МАТЕМАТИЧЕСКИЕ МОДЕЛИ РАСЧЕТА ПАРАМЕТРОВ

2.1 Подраздел первый

Для устранения такого эффекта можно применить сглаживание функций. Одним из наиболее простых способов такого сглаживания является арифметическое усреднение. При его применении каждое i -ое значение дискретной функции вычисляется в соответствии с выражением

$$x_i = \frac{1}{K} \sum_{j=i-\frac{K}{2}}^{j=i+\frac{K}{2}} x_j, \quad (2.1)$$

где K - количество точек для арифметического усреднения;
 x_j - j -ое значение функции до обработки.

Результат действия арифметического усреднения представлен на рисунке 5.

Можно применить и другие, достаточно эффективные способы сглаживания. На рисунке 6 показаны графики, полученные после применения 15-точечного сглаживания Спенсера, описываемого выражением:

$$x_i = \frac{1}{320} (-3x_{i-7} - 6x_{i-6} - 5x_{i-5} + 3x_{i-4} + 21x_{i-3} + 46x_{i-2} + 67x_{i-1} + 74x_i + \\ - 3x_{i+7} - 6x_{i+6} - 5x_{i+5} + 3x_{i+4} + 216x_{i+3} + 46x_{i+2} + 67x_{i+1}), \quad (2.2)$$

и сглаживания параболой четвертой степени по одиннадцати точкам, определяемого формулой:

$$x_i = \frac{1}{429} (18x_{i-5} - 45x_{i-4} - 10x_{i-3} + 60x_{i-2} + 120x_{i-1} + 143x_i + \\ 18x_{i+5} - 45x_{i+4} - 10x_{i+3} + 60x_{i+2} + 120x_{i+1}). \quad (2.3)$$

В сущности, формулы сглаживания описывают некоторые фильтры высоких частот и может быть определена их амплитудно-частотная характеристика. Однако смысловое понятие частоты при обработке временных трендов отличается от аналогичного понятия при обработке сигналов. Это объясняется тем, что при исследовании временных трендов интерес представляет не их частотный состав, а вид изменения (увеличение, уменьшение, постоянство, цикличность и т.д.).

2.2 Подраздел второй

2.3 Подраздел третий

3 ПРОЕКТИРОВАНИЕ АРХИТЕКТУРЫ ПРОГРАММНОГО СРЕДСТВА

3.1 Подраздел первый

6 ИНСТРУКЦИЯ ПО УСТАНОВКЕ И ИСПОЛЬЗОВАНИЮ

ЗАКЛЮЧЕНИЕ

В дипломном проекте проведен анализ современного состояния в области сетевых технологий, рассмотрены: модель файловой системы, модели взаимодействия открытых систем, протоколы сетевого взаимодействия. Эти компоненты являются базой для объединения отдельных подсистем АСУ ТП в единую систему.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Гук, М. Процессоры Pentium II, Pentium Pro и просто Pentium / М. Гук. – СПб. : Питер Ком, 1999. – 288 с.
- [2] Кузелин, М. О. Современные семейства ПЛИС фирмы Xilinx : справ. пособие / М. О. Кузелин, Д. А. Кнышев, В. Ю. Зотов. – М. : Горячая линия-Телеком, 2004. – 440 с.
- [3] Технические средства диагностирования : справочник / В.В. Ключев [и др.]. – М. : Машиностроение, 1989. – 672 с.
- [4] Embedded Microcontrollers : Databook / Intel Corporation. – Santa Clara, Ca, 1994.
- [5] Проектирование самотестируемых СБИС : монография. В 2 т. / В. Н. Ярмолик и др. – Минск : БГУИР, 2001.
- [6] Микропроцессоры и микропроцессорные комплекты интегральных микросхем : справочник. В 2 т. / под ред. В. А. Шахнова. – М. : Радио и связь, 1988. – Т. 1. – 368 с.
- [7] Берски, Д. Набор ЭСЛ-микросхем для быстродействующего RISC-процессора / Д. Берски // Электроника. – 1989. – №12. – С. 21 – 25.
- [8] Аксенов, О. Ю. Методика формирования обучающих выборок для распознающей системы / О. Ю. Аксенов // VI Всероссийская науч.-техн. конференция «Нейроинформатика–2004» : сб. науч. тр. В 2 ч. / отв. ред. О. А. Мишулина. – М. : МИФИ, 2004. – С. 215 – 222. – (Научная сессия МИФИ–2004).
- [9] Xilinx [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://www.plis.ru/>.
- [10] Mobile Intel® Pentium® Processor-M [Электронный ресурс] : Datasheet / Intel Corporation. – Электронные данные. – Режим доступа: 25068604.pdf.
- [11] Nokia+Компьютер [Электронный ресурс] : инструкции, программы, драйверы, игры, мелодии, картинки для Nokia. – М., 2004. – 1 компакт-диск (CD-R).
- [12] Клепиков, В. Использование QNX Neutrino в системах автоматического управления для ответственных применений / В. Клепиков, Д. Подхвятилин, Г. Шарпов // Средства и системы промышленной автоматизации [Электронный ресурс]. – 2008. – Режим доступа: <http://www.swd.ru/index.php3?pid=794> Дата доступа: 15.04.08
- [13] LabVIEW Real-Time LabVIEW реального времени // Средства и системы промышленной автоматизации [Электронный ресурс]. – 2008. – Режим доступа: <http://www.asutp.ru/?p=600045&PHPSESSID=7ca23f6181d2e70b9b64b3263bd28d46> Дата доступа: 15.04.08
- [14] Wale, K. VME помогает строить DSP-системы реального времени / Wale, K. // Real-Time Magazine - 2007. - №1. - С. 67-70.
- [15] Малинецкий, Г.Г. Современные проблемы нелинейной динамики / Г.Г. Малинецкий, А.Б. Потапов – Москва: Эдиториал УРСС, 2000. – 336 с.
- [16] DDK – Windows Driver Development Kit / Microsoft Corporation [Electronic resource]. – 2008 – Mode of access: <http://www.microsoft.com/whdc/devtools/ddk/default.mspx> – Date of access: 17.03.2008.
- [17] Жданов, А. Замечания о выборе операционных систем при построении

систем реального времени / А. Жданов, А. Латыев // PCWeek. – 2001. – №(271)1. – С. 22–29.

[18] Бранцевич, П.Ю. Модель системы диагностики технических объектов по вибрационным параметрам / П.Ю. Бранцевич // Наука и технологии на рубеже XXI века Материалы международной научно-технической конференции / Редкол. И.П. Филонова [и др.]. – Минск.: УП Технопринт, 2000. – С. 259–264.

[19] Бранцевич, П.Ю. Измерительно-вычислительная система распределенного сбора и централизованной обработки виброметрических данных / П.Ю. Бранцевич // Датчики и преобразователи информации систем измерения, контроля и управления. Сборник материалов 12-ой научно-технической конференции с участием зарубежных специалистов / Редкол. В.Н. Азарова. [и др.]. Москва: МГИЭМ, 2000. – С. 170–171.

[20] Kramer, M.A. A rule-based approach to fault diagnosis using the signed directed graph / M.A. Kramer, B.L. Palowitch // AIChE Journal. – 2004. – Vol. 33, № 7, P. 1067–1078.

[21] Wenzel, H. Ambient Vibration Monitoring / H. Wenzel, D. Pichler. – John Wiley & Sons, 2005. – 295 p.

[22] Бранцевич, П.Ю. Система контроля и анализа технического состояния агрегатов и механизмов по вибрационным параметрам / П.Ю. Бранцевич, Э.И. Кульков, С.Ф. Костюк // Проблемы вибрации, виброналадки, вибромониторинга и диагностики оборудования электрических станций / Сборник докладов. Редкол.: Салимона А.В. [и др.]. – Москва, 2001. – с. 60–64.

[23] Mehr, A.F. A new approach to probabilistic risk analysis in concurrent and distributed design of aerospace systems. / A.F. Mehr, I.Y. Tumer. // ASME 2005 International Design Engineering Technical Conferences, sep. 24–28, Long Beach, California, USA. / 2005. – p. 167–189.

ПРИЛОЖЕНИЕ А

(обязательное)

Текст программного модуля управления таблицами

```
using System;
using System.Text;
using System.Windows.Forms;
using System.Runtime.InteropServices;
using System.Diagnostics;

namespace Timbre
{
    // Объявление класса делегатов на основе которых определены события для возврата
    // порции собранных данных и, соответственно, событие окончания сбора
    public delegate void DataReadetHandler(ushort NumberOfChannel, float[]
DataPortion);
    public delegate void DataCaptureFinishedHandler();
    /// <summary>
    /// Класс предназначен для работы с определенным типом платы фирмы LCard
    /// В данном случае версия класса предназначена для работы с картой типа E440
    /// Для работы класса по пути \bin\Debug\ необходимо наличие библиотеки
    lcomp.dll (она при установке драйвера
    /// копируется в system32 директорию и может быть там)
    /// необходимо наличие библиотеки wlcomp.dll
    /// необходимо наличие файла загрузки биоса платы E440.bio
    /// также для работы класса необходимо наличие в проекте класса-библиотеки
    csDriverLibrary
    /// </summary>
    class csLCardDevice
    {
        private const int IRQ_STEP = 1024; // шаг прерываний (по сути частота в
        // отсчетах с которой будет изменять свой шаг указатель положения слота данных)
        public const int PORTION_SIZE = 4096; // размер порции для циклического
        // сбора данных - минимальный размер порции от шага прерываний и должен быть в 4-е раза
        // больше
        // связано это с принципом сбора данных
        // если уменьшать размер порции - то может не хватить времени на отрисовку
        // данных в реальном времени, что приведет к ошибке
        // Это приведет к потере порции и порче всех данных
        // максимум этого параметра 131 072
        private const float KWANT_ACP = 1.25f; // квант ацп в милливольтх для E-440
        private uint slotNum = 0; // номер слота в котором установлена плата
        // для ноутбука 0, для NILVD_MAIN 1
        private uint Err = 0; // переменная которая используется для хранения номера
        // ошибки
        private uint hDll; // адрес начала библиотеки (используется при
        // инициализации устройства и соответственно его освобождения)
        private uint hIfc; // указатель на устройство
        private csDriverLibrary.WDAQ_PAR DAQparameters; // объявляем экземпляр
        // структуры в которой хранятся параметры сбора
        public static IntPtr dataStartPointer; // переменная, в которой будет
        // возвращен адрес начала большого буфера;
```

ПРИЛОЖЕНИЕ Б

(обязательное)

Текст программного модуля обработки сообщения

```
public unsafe string SetWorkParameters()
{
    csDriverLibrary.WADC_PAR_0 InitParameters = new
csDriverLibrary.WADC_PAR_0();
    InitParameters.s_Type = csDriverLibrary.L_ADC_PARAM; // тип структуры
(должен быть L_ADC_PARAM);
    InitParameters.AutoInit = 1;    // флаг указывающий на тип сбора данных 0
- однократный 1 - циклический;
    InitParameters.dRate = frameSamplingFrequency;    // частота опроса
каналов в кадре (кГц);
    InitParameters.dKadr = 0;    // интервал между кадрами (мс);
    InitParameters.dScale = 0;    // масштаб работы (таймера для 1250 или
делителя для 1221)
    InitParameters.SynchroType = 0; // тип синхронизации; (если больше 2 -
то нет синхронизации)
    InitParameters.SynchroSensitivity = 0; // вид синхронизации
    InitParameters.SynchroMode = 0; // режим синхронизации
    InitParameters.AdChannel = 0; // канал по которому выполняется
синхронизация
    InitParameters.AdPorog = 0;    // уровень синхронизации
    InitParameters.NCh = numberOfChannels; // количество опрашиваемых
каналов;
    // массив с номерами каналов и усилением на них; описывает так же
порядок опроса каналов;
    for (int i = 0; i < 32; i++)
        InitParameters.Chn[i] = channelsAmplifying[i];
    InitParameters.FIFO = 2048;    // размер половины аппаратного буфера
FIFO на плате согласно документации максимум 4096;
    InitParameters.IrqStep = IRQ_STEP; // шаг генерации прерываний;
    InitParameters.Pages = (ushort)(numberOfChannels * (PORTION_SIZE /
IRQ_STEP)); // размер кольцевого буфера в шагах прерываний
    // произведение этих двух параметров IRQ_STEP*Pages задает количество
отсчетов,
    // которое соберет плата при однократном сборе, но не больше чем 128K
отсчетов.
    // При циклическом сборе они игнорируются - буфер всегда 128K.
    InitParameters.IrqEna = 1;    // разрешение генерации прерывания от
платы (1/0),
    // при этом mask - это младшие 16 бит в слове разрешающем прерывания от
платы (блок АЦП);
    InitParameters.AdcEna = 0;    // разрешение работы АЦП (1/0);
    csDriverLibrary.FillDAQparameters(ref hIfc, ref InitParameters,
(uint)2); // сброс параметров в ацп
    string ParameterSettingLog = "";
    uint tm = (uint)(numberOfChannels * PORTION_SIZE); // выделение памяти
под буфер (значение этого параметра не совсем понятно)
    Err = csDriverLibrary.RequestBufferStream(ref hIfc, ref tm,
csDriverLibrary.L_STREAM_ADC); // функция служит для выделения памяти под большой
кольцевой буфер
    ParameterSettingLog += "\nCalled RequestBufferStream. Allocated memory
size(word): " + tm.ToString() + ". Error result = " + Err.ToString() + '\n';
    // функция настраивает плату ацп/цап под заданные параметры ввода вывода
    Err = csDriverLibrary.SetParametersStream(ref hIfc, ref InitParameters,
(uint)2, ref tm, ref dataStartPointer, ref syncVarPointer,
csDriverLibrary.L_STREAM_ADC);
    ParameterSettingLog += "Настройка АЦП/ЦАП заданными параметрами ввода
вывода... \n" + ErrorMessage(Err) + "\n";
```

```

        ParameterSettingLog += "Установленные параметры : " + "\n";
        ParameterSettingLog += "Размер буфера (word) : " + tm.ToString() + "\n";
        ParameterSettingLog += "Страницы отсчетов : " +
InitParameters.Pages.ToString() + "\n";
        ParameterSettingLog += "Шаг генерации прерываний : " +
InitParameters.IrqStep.ToString() + "\n";
        ParameterSettingLog += "Размер половины аппаратного буфера FIFO на плате
: " + InitParameters.FIFO.ToString() + "\n";
        ParameterSettingLog += "частота опроса каналов : " +
InitParameters.dRate.ToString() + "\n";
        DAQparameters.t3 = InitParameters; // записываем параметры инициализации
в поле структуры
        Err = csDriverLibrary.EnableCorrection(ref hIfc,1); // Функция
включает/выключает режим коррекции. Сама загружает коэффициенты в плату
        ParameterSettingLog += "Включение режима коррекции... \n" +
ErrorMessage(Err) + "\n";
        Err = csDriverLibrary.InitStartLDevice(ref hIfc); // Функция
инициализирует внутренние переменные драйвера перед началом сбора
        ParameterSettingLog += "Инициализация внутренних переменных драйвера...
\n" + ErrorMessage(Err) + "\n";
        return ParameterSettingLog;
    }

```

Обозначение					Наименование		Дополнительные сведения		
					<u>Текстовые документы</u>				
БГУИР ДП 1–40 01 01 03 110 ПЗ					Пояснительная записка		78 с.		
					Отзыв руководителя				
					Рецензия				
					Акт о внедрении				
					<u>Графические документы</u>				
ГУИР.951002-01 СП					Программа обработки текстов.		Формат А1		
					Схема программы				
ГУИР.951002-01 СА					Реализация сортировки строк.		Формат А1		
					Схема алгоритма				
ГУИР.951002-01 СД					База данных радиоэлементов.		Формат А1		
					Схема данных				
ГУИР.951002-01 ПЛ					Диаграмма классов.		Формат А1		
					Плакат				
ГУИР.951002-02 ПЛ					Экранные формы программы.		Формат А1		
					Плакат				
ГУИР.951002-03 ПЛ					Результаты проверки работоспособности.		Формат А1		
					Плакат				