

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
Кафедра «Прикладная математика и информатика»

01.03.02 ПРИКЛАДНАЯ МАТЕМАТИКА И ИНФОРМАТИКА

СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ И КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ

БАКАЛАВРСКАЯ РАБОТА

на тему Моделирование преобразователей информационно-измерительной
системы взвешивания железнодорожных вагонов

Студент _____ Д.С. Глеске _____

Руководитель _____ А.И. Туищев _____

Консультант по _____ Н.В. Яценко _____
аннотации

Допустить к защите
Заведующий кафедрой к.тех.н, доцент, А.В. Очеповский _____

« _____ » _____ 20__ г.

Тольятти 2017

АННОТАЦИЯ

Цель исследования – смоделировать и разработать информационно-измерительную систему взвешивания железнодорожных вагонов с применением высокочувствительных преобразователей.

Объектом исследования является точность процесса взвешивания железнодорожных вагонов с использованием информационно-измерительных систем.

Предметом исследования является компьютерная модель подсистемы обработки данных информационно-измерительной системы, предназначенной для повышения эффективности процесса поосного взвешивания вагонов.

Выпускная квалификационная работа состоит из введения, трех глав, заключения, списка литературы и приложения. Объем работы составляет 66 страниц, 39 рисунков, 8 формул и 1 таблицы. Использовано 22 источника литературы.

Введение к ВКР включает в себя описание цели и актуальности создания систем взвешивания на железнодорожном транспорте, а также краткую структура ВКР.

В первой главе проведен анализ существующих систем взвешивания железнодорожных вагонов. Определен подход к разработке. Сформулированы требования к моделируемой системе.

Во второй главе смоделирован преобразователь и новая подсистема ИИС взвешивания.

В третьей главе проведено тестирование смоделированной и разработанной системы и представлен пользовательский интерфейс.

В заключении представлены развёрнутые выводы по проделанной работе и её результатам.

ABSTRACT

The title of the given graduation work is “Modeling of Transducers of the Information-measuring System of Railway Wagons Weighing”.

The object of the graduation work is the precision of the railway wagon weighing using the information-measuring systems. The subject of the work is the computer model of data processing subsystem, which is used for efficiency increase of wagon weighing process. The aim of the work is to model and develop information-measuring system of railway wagon weighing using the high-sensitive transducers. The relevance of work is to automatize the process of information collecting.

The graduation work consists of an explanatory note on 66 pages, including 39 figures, 8 formulas, 1 table and the list of 22 references including 5 foreign sources and 1 appendix.

The first part of the work contains analysis of existing railway wagons weighing systems, and modeled system requirements.

The second part of the work is concentrated on modelling of transducer and new subsystem of the information-measuring system of weighing.

The third part of the work deals with testing of created system.

The result of the work is a working model of information-measuring system, which describes the processes of railway transport weighing with different levels of specification.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
ГЛАВА 1. АНАЛИЗ СУЩЕСТВУЮЩИХ ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНЫХ СИСТЕМ ВЗВЕШИВАНИЯ ЖЕЛЕЗНОДОРОЖНЫХ ВАГОНОВ	7
1.1. Информационно-измерительные системы взвешивания железнодорожных вагонов.	7
1.2. Анализ существующих систем взвешивания	9
1.2.1. Информационно-измерительная система «ВЕСМАШ»	10
1.2.2. Информационно-измерительная система «Mettler Toledo»	10
1.2.3. Информационно-измерительная система Корпорация «АСИ»	11
1.2.4. Выбор подхода к разработке	11
1.2.5. Типовые компоненты информационно–измерительных систем взвешивания	16
1.2.6. Построение модели информационно-измерительной системы «AS-IS»	21
1.3. Формализация требований к моделируемому программному продукту и выбор программных средств реализации	24
ГЛАВА 2. МОДЕЛИРОВАНИЕ И РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПРЕОБРАЗОВАТЕЛЕЙ ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНОЙ СИСТЕМЫ ВЗВЕШИВАНИЯ ЖЕЛЕЗНОДОРОЖНЫХ ВАГОНОВ	29
2.1. Моделирование информационно-измерительных систем	29
2.1.1. Устройства сбора, первичной обработки и передачи информации в информационно–измерительных системах	33

2.1.2. Методологические особенности поосного взвешивания железнодорожных вагонов	37
2.1.3. Построение модели информационно-измерительной системы «ТО-ВЕ»	43
ГЛАВА 3. ТЕСТИРОВАНИЕ И ВНЕДРЕНИЕ РАЗРАБОТАННЫХ ПРОГРАММНЫХ РЕШЕНИЙ	51
3.1. Тестирование программного продукта	51
ЗАКЛЮЧЕНИЕ	56
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ	59
ПРИЛОЖЕНИЕ А. Листинг программы	61

ВВЕДЕНИЕ

Применение информационно-измерительных систем, построенных на индуктивных датчиках, дает возможность взвешивать железнодорожные вагоны и автопоезда большой грузоподъемности. Такие системы могут применяться в железнодорожном транспорте во время погрузки, а также встраиваться в железнодорожную структуру. В частности, в вновь строящиеся железные дороги, что дает возможность взвешивать на терминалах и при движении нагрузку на каждую ось вагона, что предотвратит преждевременное разрушение дорог.

Опыт реализации подобного оборудования, показал, что весы поосного типа получили признание у организаций разного направления: сельхозпроизводителей, металлообрабатывающих организаций, строительных предприятий.

В результате установки электронных железнодорожных весов организация получает контроль над грузооборотом, что приводит к уменьшению производственных потерь, а применение программного обеспечения позволяет еще более эффективно контролировать все этапы взвешивания.

Как правило, точность этих весов определяет основные экономические показатели предприятия, в том числе и уровень, так называемых, «нетехнологических» потерь.

Существенным недостатком многих информационно-измерительных систем является то, что сотрудники постов весового контроля при определении допустимой осевой нагрузки пользуются специальными справочниками, где расписаны предельные нагрузки для всех типов транспорта. То есть они вынуждены анализировать и вручную обрабатывать данные, полученные в цифровой или аналоговой форме. Это значительно увеличивает время, затраченное на обработку, и приводит к росту количества неверно обработанных данных.

Актуальность обусловлена тем, что автоматизация этих процессов позволяет значительно снизить время, затраченное на обработку, и уменьшить количество ошибок при обработке данных.

Цель исследования – анализ информационно-измерительных систем взвешивания железнодорожных вагонов и моделирование информационно-измерительной системы взвешивания железнодорожных вагонов с применением высокочувствительных преобразователей.

Объектом исследования является процесс взвешивания железнодорожных вагонов с использованием информационно-измерительных систем.

Предметом исследования является модель подсистемы обработки данных информационно-измерительной системы, предназначенной для повышения эффективности процесса поочередного взвешивания вагонов.

В соответствии с объектом, предметом и целью исследования потребовалось решить следующие **задачи**:

1. Изучить особенности построения информационно-измерительных систем взвешивания железнодорожных вагонов;
2. Рассмотреть состав информационно-измерительных систем, используемых при взвешивании железнодорожных вагонов;
3. Проанализировать принципы работы датчиков давления и методы расчета их упругих элементов;
4. раскрыть сущность процесса взвешивания транспорта с использованием информационно-измерительных систем;
5. Смоделировать информационно-измерительную систему взвешивания железнодорожных вагонов.

Практическая значимость исследования состоит в том, что:

- построены модели, позволяющие детально рассмотреть все процессы, происходящие при взвешивании транспорта с использованием ИИС;

Структура ВКР обусловлена логикой и последовательностью решения задач исследования. ВКР состоит из введения, двух глав, заключения, списка использованных источников.

В первой главе проведен анализ существующих систем взвешивания железнодорожных вагонов. Определен подход к разработке. Сформулированы требования к моделируемой системе.

Во второй главе смоделирована разработана преобразователь и новая подсистема ИИС взвешивания.

В третьей главе проведено тестирование смоделированной и разработанной системы и представлен пользовательский интерфейс.

В заключении представлены развёрнутые выводы по проделанной работе и её результатам.

ГЛАВА 1. АНАЛИЗ СУЩЕСТВУЮЩИХ ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНЫХ СИСТЕМ ВЗВЕШИВАНИЯ ЖЕЛЕЗНОДОРОЖНЫХ ВАГОНОВ

1.1. Информационно-измерительные системы взвешивания железнодорожных вагонов

Железнодорожные вагонные весы предназначены для взвешивания подвижного состава (с расцепкой или без расцепки вагонов) в статическом или динамическом режиме на стационарных постах, а также имеют широкое распространение на предприятиях промышленности и транспорта.

Все железнодорожные вагонные весы классифицируются по нескольким принципам:

1) принцип действия:

а) механические – в настоящее время не стоят на производстве;

б) электронные - весы, в основе которых чаще всего используются тензометрические датчики.

2) назначение:

а) для предприятий - весы данного типа преимущественно используются для коммерческого измерения веса. В случае небольшого грузопотока используются статические весы. Если же грузопоток вагонов больше 15 вагонов в сутки, то использование стато-динамических весов, которые позволяют взвешивать как в статике, так и в движении, более целесообразно;

б) для железной дороги - весы устанавливаются на главных путях, станционных путях для контроля перегруза вагонов. Принципиальная особенность данных весов – неразрезной рель, позволяющий не понижать скорость движения через весы до 15 км/ч.

3) тип взвешивания:

а) статические - взвешивание производится в статике. Данный тип взвешивания является наиболее точным;

б) статодинамические - взвешивание производится как с остановкой, так и в движении. Взвешивание подвижного вагона производится на скорости не более 10-15 км/ч. По окончании прохождения состава выдается отчет о массе всех вагонов, прошедших через весы. Преимущества данных весов – экономия времени и затрат на маневровых работах;

в) динамические - взвешивание вагонов производится только в движении. Предназначено для технологического учета на крупных предприятиях и железной дороге.

4) тип конструкции:

а) фундаментные - классические вагонные весы, требуют заливки фундамента из железобетона;

б) весы на сборном железобетоне - устанавливаются на заранее изготовленных конструкциях, что позволяет сократить сроки монтажа вагонных весов до 3 -5 недель;

в) бесфундаментные вагонные весы - это новейшее поколение вагонных весов, может быть выполнено на шпалах или металлическом фундаменте. Поскольку весы данного типа устанавливаются на основание из щебня, обязательным требованием к данному типу вагонных весов является нечувствительность к просадкам грунта, что гарантирует стабильные метрологические характеристики. Основное преимущество – это короткие сроки монтажа, установка вагонных весов может быть выполнена за 2-3 рабочие смены.

5) тип взвешивания:

а) двуосные;

б) четырехосные;

в) шестиосные;

г) восьмиосные.

б) тип электроники:

а) аналоговая тензометрическая система - наиболее распространенная система, построенная на аналоговых тензометрических датчиках и балансировочной коробке. Основным недостатком является наличие всего одного канала на все датчики;

б) цифровая тензометрическая система - получила распространение в наши дни. Может быть построена как на цифровых тензодатчиках, так и аналоговых датчиках с цифровой распределительной коробкой. Основное преимущество заключается в том, что каждый датчик имеет отдельный канал, что позволяет производить простую диагностику и настройку. Благодаря разделению каналов стало возможным определение продольного и поперечного смещения центра тяжести взвешиваемого вагона.

1.2. Анализ существующих систем взвешивания

На сегодняшний день уже существуют готовые разнообразные информационно-измерительные системы взвешивания вагонов. Все эти системы имеют множество отличий и могут быть реализованы по-разному [9]. Большое количество систем измерения веса выполнены по клиент-серверной архитектуре. Такую структуру можно условно разделить на две основных составляющих: сервер и множество клиентов. С помощью линий связи осуществляется взаимодействие между сервером и многочисленными клиентами. Положительная сторона подобной архитектуры заключается в том, что большая часть логики и используемой информации, как правило, размещены на сервере, а клиенты получают доступ только к предоставляемой сервером информации. В качестве преимуществ данной архитектуры можно отметить возможность работы с системой сразу несколькими клиентами, а также возможность взаимодействия на большом расстоянии, то есть клиент может быть удален от сервера на большое расстояние, но, тем не менее, получить доступ к нему, например, через Интернет.

Программное обеспечение, выполненное по клиент-серверной архитектуре можно подразделить на два вида: работающие с помощью специализированных клиентов и работающие через браузер. Браузерные системы являются более удобными, так как не требуется устанавливать отдельное специализированное программное обеспечение на компьютерах-клиентах. Еще одним преимуществом систем, работающих через браузер, является то, что их можно разместить на хостинге в сети Интернет, тем самым предоставив доступ к этим приложениям извне.

Далее будут рассмотрены наиболее популярные системы взвешивания железнодорожных вагонов созданные компаниями «ВЕСМАШ», «Mettler Toledo» и Корпорацией «АСИ».

1.2.1. Информационно-измерительная система «ВЕСМАШ»

Вагонные весы для взвешивания в статике от завода-производителя «ВЕСМАШ» подходят:

- для производства взвешивания порожних и загруженных вагонов;
- для регистрации в базе данных веса вагона и нетто, и брутто, с последующим формированием и выдачей информации на печать протокола и/или экран дисплея;
- для хранения и передачи информации о массе груза ее технологическим пользователям (коммерческой службе или диспетчерам цеха) в соответствии со всеми пожеланиями заказчика;
- для осуществления связи с подсистемами АСУ предприятия;
- для выполнения тестирования и иных служебных функций.

Основой для установки всех моделей, выпускаемых заводом «ВЕСМАШ» вагонных весов может служить не только фундамент из железобетона в классическом исполнении, но также опорная рамная конструкция, являющаяся бесфундаментным вариантом установки.

Узлы встройки датчиков в железнодорожных весах "ВЕСМАШ" находятся под платформой. Такое расположение датчиков увеличивает

точность всей весовой системы, а также снижает воздействие на них внешней среды. Таким образом, это сводит к минимуму погрешность при взвешивании.

1.2.2. Информационно-измерительная система «Mettler Toledo»

Информационно-измерительная система взвешивания железнодорожных вагонов «Mettler Toledo» сопряжена с весами для статического взвешивания вагонов, а также допускает возможность подключения весов для производства взвешивания железнодорожных составов в движении. Технология взвешивания в движении без расцепки позволяет за короткое время измерить массу нескольких вагонов или целых поездов.

1.2.3. Информационно-измерительная система Корпорация «АСИ»

Весы для взвешивания железнодорожных вагонов и составов, разработанные Корпорацией "АСИ", предназначены для быстрого и надежного получения информации о грузопотоках. Информация о взвешивании может быть использована как для коммерческих расчетов, так и складского учета, контроля производства, накопления и анализа статистики, экономической безопасности предприятия, контроля перемещения груза, контроля передвижения материалов внутри предприятия.

1.2.4. Выбор подхода к разработке

При разработке системы используются два основополагающих подхода: структурный подход для анализа бизнес процессов и объектно-ориентированный подход для построения будущей системы.

Структурный подход в разработке автоматизированных систем характеризуется декомпозицией создаваемой системы на отдельные автоматизированные функции: выполняется разделение системы на фундаментальные подсистемы, которые делятся на подфункции, подразделяющиеся на задачи и так далее. При таком подходе проектируемая

система сохранит целостное представление, а составляющие компоненты будут взаимосвязаны.

В структурном анализе для иллюстрации функций, которые выполняет система и отношений между данными в основном используется две группы средств. Для каждой группы средств существуют определенные виды диаграмм. Наиболее распространенными среди них являются следующие:

- 1) SADT модели и функциональные диаграммы, соответствующие им;
- 2) диаграммы потоков данных DFD;
- 3) диаграммы сущность-связь ERD.

При проектировании системы модели уточняются, расширяются, дополняются диаграммами, которые отражают структуру программного обеспечения: схемы структуры программ, архитектуру программного обеспечения и диаграммы, отражающие экранные формы.

Объектно-ориентированный подход основывается на систематическом применении моделей для независимой от языка разработки программной системы. Основой в этом подходе является прагматика системы.

Прагматика определяется целями, которые должна решать разрабатываемая система. При описании цели принимают участие понятия и предметы реального мира, которые имеют отношение к проектируемой программной системе. В объектно-ориентированном подходе эти понятия и предметы замещаются их моделями, то есть специальными формальными конструкциями, которые их представляют в программной системе.

Любая модель имеет не все свойства и признаки понятия или предмета, который она отражает, а только некоторые наиболее важные для проектируемой системы. Таким образом, модель проще оригинала, что делает ее более удобной для использования.

Объектно-ориентированный подход позволяет преодолеть такие сложные проблемы, как:

- возможность использования определенных компонентов повторно;
- возможность улучшать определенные компоненты, не изменяя другие;

- увеличение надежности системы;
- снижение сложности системы.

Следование объектно-ориентированному подходу способствует разработке надежных, улучшаемых, хорошо структурированных программных систем.

Объектно-ориентированный подход основан на нескольких важнейших принципах, которые придают объектам новые свойства. К этим принципам относят следующие принципы:

- 1) наследование;
- 2) полиморфизм;
- 3) инкапсуляция;
- 4) абстракция.

Принцип наследования можно охарактеризовать, как возможность объектов порождать объекты-потомки. Объект-потомок наследует автоматически от объекта-родителя все методы и поля, может быть дополнен новыми полями и методами, или же существующие поля и методы могут быть заменены на новые.

Данный принцип объектно-ориентированного программирования решает проблемы модификации свойств какого-либо объекта, а также придает объектно-ориентированному программированию в целом гибкость в очень высокой степени. Когда программист работает с объектами, он обычно выбирает объект, который по своим свойствам наиболее близок для решения поставленной задачи, а после этого создает необходимое количество потомков от выбранного объекта, которые будут реализовывать то, что не предусмотрено в родителе.

Последовательное применение на практике принципа объектно-ориентированного программирование наследование хорошо сочетается с поэтапной концепцией к разработке программных проектов и стимулирует такой подход во многом.

Полиморфизм – это свойство, которым обладают родственные объекты, то есть объекты, имеющие одного родителя, которое заключается в способности объектов решать однотипные по смыслу проблемы различными способами. Свойства, определяющие поведение объекта, в рамках объектно-ориентированного программирования определяются совокупностью методов, входящих в него. Изменив алгоритм какого-либо метода у объекта-потомка, можно придать этим объектам специфические свойства, отсутствующие у объекта-родителя. Для того чтобы изменить метод требуется перекрыть его в объекте-потомке, то есть объявить одноименный метод в объекте-потомке и описать в нем требуемые действия. Таким образом, в объекте потомке и объекте-родителе будут существовать одноименные методы, которые имеют разный программный код и по этой причине, придающие разные свойства объектам.

Инкапсуляция – это принцип, согласно которому данные и алгоритмы обработки этих данных объединяются в единое целое. В рамках объектно-ориентированного программирования данные называют полями, а алгоритмы обработки данных называют методами.

Принцип инкапсуляции позволяет в высшей степени выполнить изолирование какого-либо объекта от внешнего окружения. Инкапсуляция может существенно повысить надежность разрабатываемого программного обеспечения, так как помещенные внутрь объекта алгоритмы выполняют обмен данными с программой достаточно в небольших объемах, причем тип данных и количество чаще всего жестко контролируются. Таким образом, модификация или замена данных и алгоритмов, помещенных в объект, обычно не несет за собой последствий, которые сложно проследить.

Еще одним важным следствием, которое вытекает из инкапсуляции, является простота обмена различными объектами и их переноса между программами. Это влечет за собой создание библиотек классов, которые потом очень удобно использовать в других проектах.

Абстракция – это принцип объектно-ориентированного программирования, который состоит в возможности придания объекту определенных характеристик, которые совершенно точно определяют его границы, таким образом, отличая данный объект от каких-либо других объектов. Главная идея заключается в отделении способа использования объектов от механизмов их реализации в качестве более простых объектов. Эта идея подобна тому, как разделяет функциональная абстракция способ применения функции и особенностей ее реализации в виде более примитивных функций, в результате этого функцией высокого уровня обрабатываются данные посредством вызова функций более низкого уровня.

Принципиальное различие между структурным и объектно-ориентированным подходом заключается в способе декомпозиции системы рассмотренном на рисунке 1. Объектно-ориентированный подход использует объектную декомпозицию, при этом статическая структура системы описывается в терминах объектов и связей между ними, а поведение системы описывается в терминах обмена сообщений между объектами.



Рисунок 1 – Сравнительный анализ функционально-ориентированной и объектно-ориентированной декомпозиций

1.2.5. Типовые компоненты информационно–измерительных систем взвешивания

Информационно-измерительная система (ИИС) — комплекс технических средств (измерительных и вычислительных) и программного обеспечения для автоматического получения необходимой информации непосредственно от контролируемого объекта, визуализации, регистрации выходных данных и обработки этой информации на ЭВМ.

В состав технических средств входит оборудование для ввода, хранения, преобразования и вывода данных, в том числе вычислительные машины (ВМ) или компьютеры, устройства сопряжения ВМ с объектами, аппаратура передачи данных (коммуникационное оборудование) и линии связи.

Программное обеспечение (программные средства) – совокупность программ, реализующих возложенные на систему функции (различают системное и прикладное).

Состав и структура конкретной ИИС определяются общими техническими требованиями, установленными государственным стандартом, и частными требованиями, содержащимися в техническом задании на его создание.

Информационно-измерительная система должна:

- управлять измерительным экспериментом в соответствии с принятым алгоритмом функционирования;
- выполнять возложенные на нее функции в соответствии с назначением и целью;
- обладать требуемыми показателями и характеристиками точности, надежности и быстродействия;
- отвечать экономическим требованиям, предъявляемым к способам и форме представления информации, размещению технических средств и т.д.;
- быть приспособленным к функционированию с ИИС смежных уровней иерархии и другими ИИС и ИВК, т.е. обладать свойством

программной, технической, информационной и метрологической совместимости;

- допускать возможность дальнейшей модернизации и развития и др.

Схема взаимодействия основных компонентов ИИС представлена на рисунке 2.

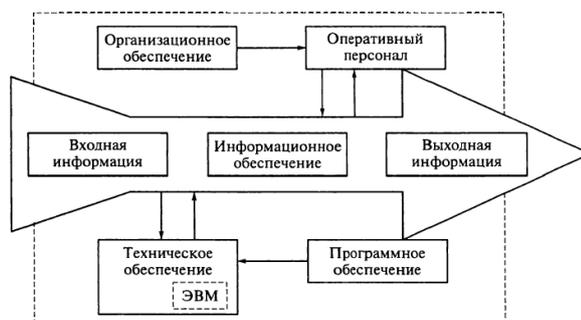


Рисунок 2 - Схема взаимодействия основных компонентов ИИС

Большинство информационно-измерительных систем имеет 2 уровня сбора и обработки информации:

- уровень передачи данных и диспетчерского контроля (верхний уровень);
- технологический уровень (нижний уровень).

На верхнем уровне системы используются следующие компоненты:

- автоматизированные рабочие места диспетчеров на базе персональных компьютеров (пункты управления - ПУ);
- рабочие станции, функционирующие в режиме коммуникационных серверов и поддерживающие диагностические функции (центральные концентраторы информации - ЦКИ);
- устройства сбора, обработки и хранения технологической информации группы объектов из состава нижнего уровня комплекса (концентраторы информации - КИ).

Обычно аппаратура ПУ располагается в диспетчерской, а ЦКИ – в узле связи. Связь между ПУ и ЦКИ осуществляется с помощью различных средств и систем связи: физическая линия, оптоволокно, радиосвязь, GSM-

связь, сеть Ethernet и т.д. КИ используется для интеграции нескольких объектов нижнего уровня системы.

На нижнем уровне системы используются контролируемые пункты телемеханики (КП), системы автоматического управления (САУ), устройства управления объектом (УУО) и другие устройства необходимые для корректного функционирования ИИК.

В качестве программного обеспечения, устанавливаемого на ПУ, ЦКИ, КИ и т.д. применяется комплекс программ, обеспечивающий сбор технологической информации в реальном времени, ее отображение и архивирование, трансляцию команд управления оборудованием и выполнение других необходимых функций.

Рассмотрим подробнее те устройства, которые осуществляют функции сбора, предварительной обработки, представления, передачи и обработки информации. Схема ИИК представлена на рисунке 3. На рисунке используются следующие сокращения:

- УВХ – устройство выборки и хранения;
- АЦП – аналого-цифровой преобразователь;
- CPU (central processing unit) – центральное процессорное устройство;
- DSP (Digital signal processor) – цифровой сигнальный процессор;
- DRAM (Dynamic random access memory) – динамическая память с произвольным доступом;
- HD (High Density) – формат высокой плотности записи информации;
- SVGA (Super Video Graphics Array) – видеоадаптер;
- ЦАП – цифро-аналоговый преобразователь.

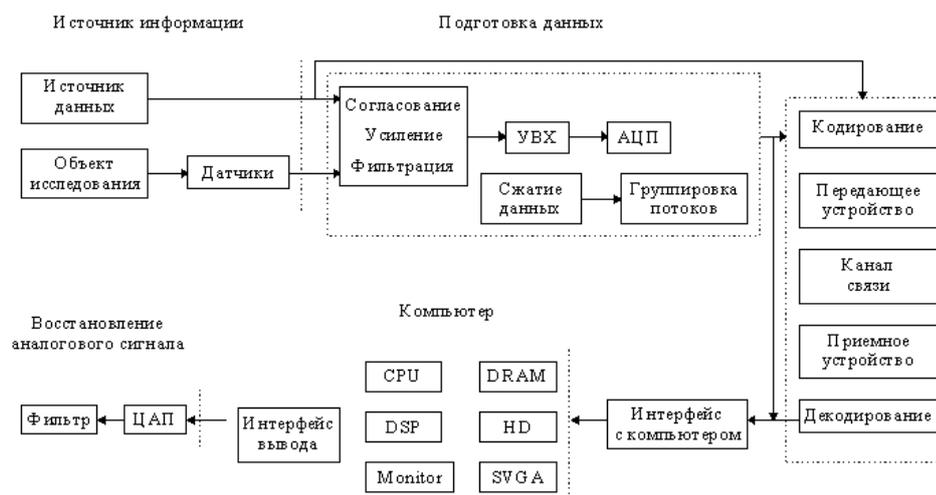


Рисунок 3 - Обобщенная схема информационно-измерительной системы

На вход поступает аналоговый сигнал, сформированный информационным устройством (или датчиком), являющимся источником данных. Цепь преобразования данных одного устройства (или датчика) в многоканальной системе образует измерительный канал.

Источник данных оснащается следующей аппаратурой:

- датчиковой (тензометрические, пьезоэлектрические, термоэлектрические, потенциометрические, индукционные и др.);
- усилительно-согласующей;
- регистрирующей;
- обрабатывающей.

При подготовке данных, к сигналу применяется предварительная аналоговая обработка – согласование, усиление (приведение амплитуды к динамическому диапазону УВХ), полосовая фильтрация (ограничение полосы частот сигналов для корректной оцифровки).

Каждый сигнал подвергается процедуре аналого-цифрового преобразования в модуле АЦП, так как в современных ИИС подсистема обработки является цифровой системой. Отсчеты от разных измерительных каналов объединяются в общий поток для процедуры ввода в компьютер или передачи по каналу связи. Иногда применяются устройства сжатия данных,

иногда сжатие осуществляется после ввода данных в компьютер (программные методы сжатия).

Приведенная схема не является единственной возможной последовательностью расположения устройств. Также, в различных ИИС состав устройств может отличаться, однако наличие данных устройств как типовых в системах различного назначения и технического воплощения является характерным.

Подсистема передачи данных состоит из кодирующего и декодирующего устройства, передающего и приемного устройства и самого канала связи. Кодирующее и декодирующее устройства выполняют помехоустойчивое зашифровку и дешифровку сигналов для обеспечения дополнительной защиты передаваемых сообщений от помех в канале связи и могут отсутствовать при наличии качественного канала.

Восстановление исходного сообщения аналогового типа по цифровым отсчетам производится на приемной стороне с допустимым уровнем погрешности. В современных системах в восстановлении сообщения нет необходимости, поскольку хранение, регистрация и обработка информации осуществляются в цифровом виде, но наличие возможности восстановления обычно предусматривается.

Основной задачей подсистемы цифровой обработки, которая выполняется с использованием ресурсов компьютера и специализированных процессоров, является сортировка информации и отбрасывание заведомо неверных результатов наблюдений. Отбраковка является частным случаем более общей задачи – фильтрации сигналов от помех.

Подсистема обработки также выполняет следующие задачи:

- предварительная обработка данных (сглаживание, удаление тренда);
- статистическая обработка сигналов с использованием алгоритмов в зависимости от назначения ИИС;
- спектральная обработка;
- формирование моделей процессов и явлений;

- представление результатов обработки или анализа;
- хранение данных.

Состав и структура конкретной ИИС определяются техническими требованиями, установленными государственным стандартом, и требованиями, содержащимися в техническом задании на его создание.

На вход комплекса поступает аналоговый сигнал, сформированный датчиком. В блоке подготовки сигнал подвергается предварительной аналоговой обработке – фильтрация, усиление, АЦП. Подсистема передачи данных включает кодирующее и декодирующее устройства, передающее и приемное устройства и сам канал связи.

Далее данные попадают в подсистему цифровой обработки, которая выполняется с использованием ресурсов компьютера и специализированных программ.

В современных комплексах чаще всего нет необходимости в восстановлении аналогового сообщения, но наличие такой возможности обычно предусматривается.

Состав и последовательность расположения устройств в различных ИИС может отличаться, но характерным является наличие данных устройств как типовых в системах различного назначения и технического воплощения.

1.2.6. Построение модели информационно-измерительной системы «AS-IS»

С помощью программы моделирования IDEF была составлена функциональная модель существующих информационно-измерительных систем взвешивания железнодорожных вагонов. Построение функциональной модели начинается с изучения происходящих в моделируемой области процессов. Для изучения этих процессов строится функциональная модель AS-IS (модель «как есть»), т.е. модель уже существующих процессов и функций [13].

Построение модели AS-IS позволяет четко зафиксировать, какие процессы осуществляются, какие информационные объекты используются при выполнении различных функций на всех уровнях детализации.

Анализ функциональной модели AS-IS позволяет понять наиболее слабые места, в чем будут состоять преимущества новых процессов, каким изменениям подвергнется существующая структура организации процесса и насколько глубокими будут эти изменения.

Согласно рисунку 3 на вход системы поступает аналоговый сигнал, сформированный датчиком. В блоке подготовки сигнал подвергается предварительной аналоговой обработке – фильтрация, усиление, АЦП. Подсистема передачи данных включает кодирующее и декодирующее устройства, передающее и приемное устройства и сам канал связи.

Далее данные попадают в подсистему цифровой обработки, которая выполняется с использованием ресурсов компьютера и специализированных программ.

В современных комплексах чаще всего нет необходимости в восстановлении аналогового сообщения, но наличие такой возможности обычно предусматривается.

Модель AS-IS процесса «Взвешивание транспорта с использованием ИИС» представлена на рисунке 4.

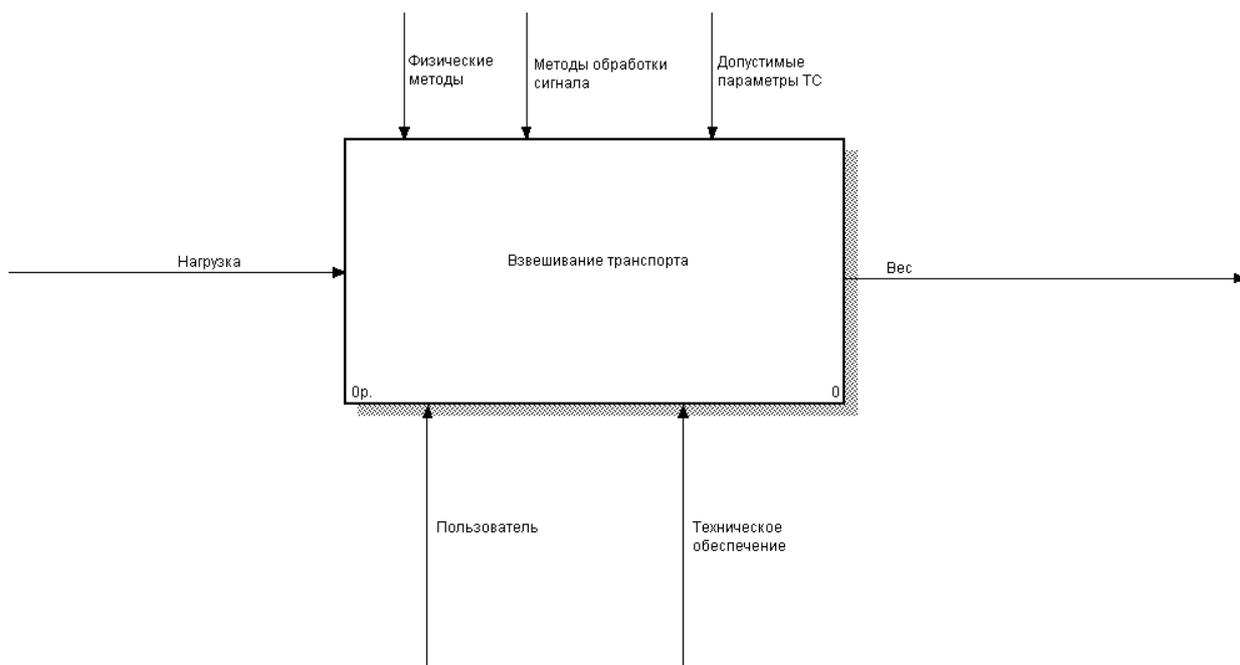


Рисунок 4 - Контекстная диаграмма процесса «Взвешивание транспорта с использованием ИИС» (AS-IS)

Для более детального изучения происходящих при взвешивании вагонов процессов необходимо выполнить декомпозицию блока «Взвешивание транспорта с использованием ИИС» представленной на рисунке 5.

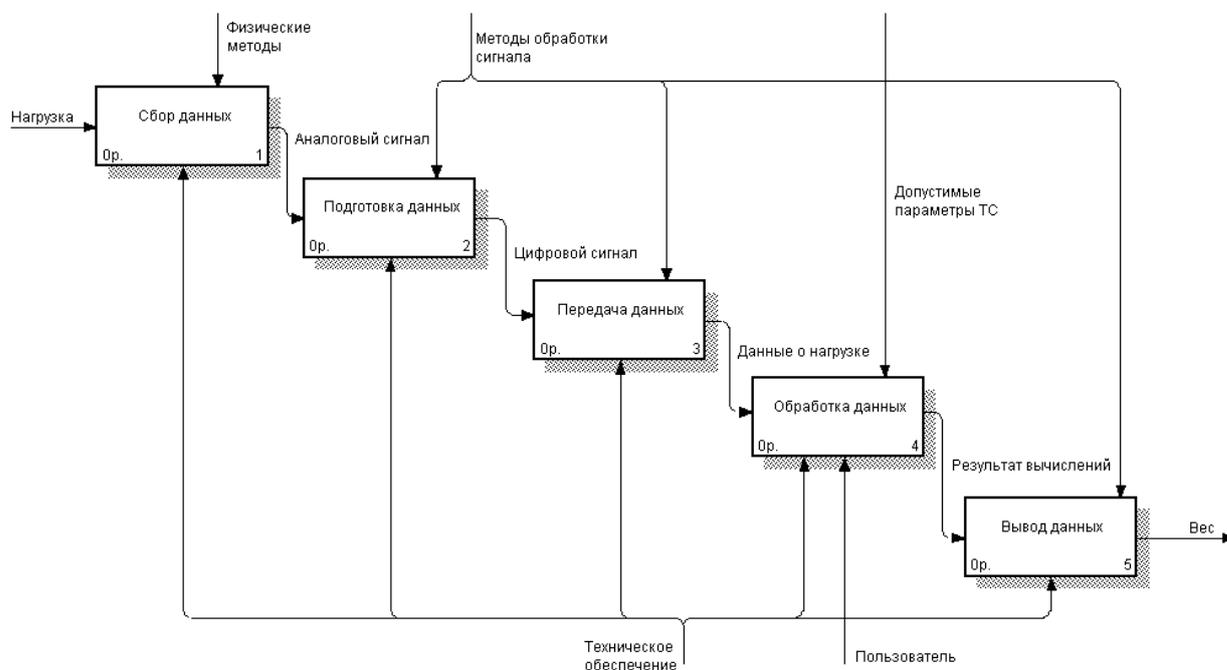


Рисунок 5 - Декомпозиция процесса «Взвешивание транспорта с использованием ИИС» (AS-IS)

Из рисунка 5 видно, что процесс «Взвешивание транспорта с использованием ИИС» состоит из следующих подпроцессов:

- сбор данных, где происходит получение данных от измеряемого объекта и их преобразование в аналоговый сигнал;
- подготовка данных, где рассмотрены такие процессы как: фильтрация аналогового сигнала, усиление сигнала и аналого-цифровое преобразование;
- передача данных, включающая в себя процессы кодирования, передачи, приема и декодирования данных;
- обработка данных, где вычисляются осевые нагрузки и общий вес транспортного средства;
- вывод данных, предполагающий вывод информации о весе транспортного средства в цифровом (если необходимо еще и в аналоговом) виде.

Более детальная декомпозиция процессов нецелесообразна, так как в различных ИИС рассматриваемые процессы и их состав могут значительно отличаться.

Существенным недостатком многих ИИС является то, что сотрудники постов весового контроля при определении допустимой осевой нагрузки пользуются специальными справочниками, где расписаны предельные нагрузки для всех типов транспорта. То есть они вынуждены анализировать и вручную обрабатывать данные, полученные в цифровой или аналоговой форме.

1.3. Формализация требований к моделируемому программному продукту и выбор программных средств реализации

Исходя из недостатков, выявленных с помощью модели AS-IS, можно сделать вывод о необходимости разработки автоматизированной подсистемы обработки данных, которая будет отображена с помощью модели TO-BE.

На вход любой информационно-измерительной системы взвешивания поступает аналоговый сигнал, сформированный датчиком (преобразователем), являющимся источником данных. В ИИС используются

различные типы преобразователей, но в работе рассматриваются индуктивные преобразователи, которые преобразуют значение измеряемой величины в значение индуктивности.

Большинство используемых в настоящее время методов расчета упругих элементов основано на ряде заранее принятых допущений, позволяющих упростить методики расчета, но приводящих к большому различию между расчетными характеристиками и полученными в результате эксперимента.

Сложность расчета уравнений, основанных на теории тонких оболочек, объясняется большим числом параметров модели (толщина, форма, радиус изгиба, наличие краевых эффектов и т.д.). Использование компьютерных процедур для решения моделей позволяет с высокой точностью получать параметры и характеристики деформационного состояния тонкостенных оболочек.

Наибольшее внимание в работе будет уделено процессам «Сбор данных» и «Обработка данных». Использование для сбора данных об исследуемом объекте датчиков, чувствительные элементы которых спроектированы на основе уравнений, приведенных во второй главе данной работы, позволяет повысить точность измерений. А использование разрабатываемой автоматизированной подсистемы обработки данных позволяет значительно снизить время, затраченное на обработку, и уменьшить количество ошибок при обработке данных.

При выборе программного средства для реализации подсистемы обработки данных были проанализированы различные программные системы (Microsoft Visual Basic, C++, Delphi).

Microsoft Visual Basic — язык программирования и среда разработки, объединенные в средство разработки, созданное корпорацией Microsoft. Язык Visual Basic является прямым наследником языка Бейсик, у которого есть немало диалектов. Visual Basic имеет элементы как компонентно-

ориентированного, так и объектно-ориентированного языков программирования. Среда разработки VB обладает инструментами для визуального конструирования пользовательского интерфейса.

Visual Basic считается достаточно хорошим средством быстрой разработки рабочих прототипов программа, а также для разработки приложений управления базами данных и для конструирования программ, работающих под управлением операционных систем семейства Microsoft Windows.

C++ — компилируемый статически типизированный язык программирования общего назначения.

Язык использует различные парадигмы программирования, а также имеет свойства языков как низкого, так и высокого уровня. В C++ основное внимание сконцентрировано на объектно-ориентированном и обобщенном программировании.

На сегодняшний день, C++ является одним из популярнейших языков программирования, и широко используется в различных областях разработки программного обеспечения, таких как создание прикладных программ, операционных и встраиваемых систем, высокопроизводительных серверов, а также в разработке развлекательных приложений.

Delphi — среда программирования и язык, разработанные компанией Borland на основе языка Object Pascal. Изначально среда разработки была предназначена исключительно для разработки приложений Microsoft Windows, затем был реализован также для платформ Linux.

Сравнение возможностей различных средств разработки представлено в таблице 1.

Таблица 1 - Сравнительная таблица средств разработки

Возможности	Программные системы		
	Visual basic	C++	Delphi
Императивный	+	+	+
Объектно-ориентированный	+	+	+

Продолжение таблицы 1

Возможности	Программные системы		
	Visual basic	C++	Delphi
Динамическая типизация	+	+/-	+/-
Блок else (исключения)	+	-	+
Динамические массивы	+	+	+
Интерфейсы	+	+	+
Локальные функции	+/-	+	+
Работа с файлами данных	+	+	+
Запуск внешних программ	+	+	+

Для реализации подсистемы была выбрана среда *Visual basic*. Это интегрированная среда разработки приложений, которая дает возможность визуализировать процесс создания графического интерфейса разрабатываемого приложения, т.е. позволяет создавать объекты и задавать значения их свойств с помощью диалоговых окон системы программирования.

Достоинства Visual basic:

- Visual basic выгодно отличается от других языков программирования своей простотой и наглядностью;
- Visual basic включен в такие программы как Word, Excel и др. с его помощью можно управлять этими программами из других программ;
- VB поддерживает все операторы структурного программирования (if, while, for, и т.д.);
- являясь средством быстрой разработки приложений, VB максимально упрощает создание интерфейса будущих приложений.

К программному обеспечению, используемому для разработки подсистемы обработки данных ИИС, предъявлялись следующие требования:

- возможность создания графического пользовательского интерфейса;

- наличие динамических массивов;
- поддержка математических функций;
- возможность импортирования данных из файлов;
- наличие функций, для запуска внешних программ;
- использование динамической типизации.

Как и любой другой язык программирования Visual Basic имеет свои преимущества и свои недостатки. Данный язык был выбран для реализации проекта, потому что он отвечает всем предъявленным требованиям, кроме того, VB отличается от других языков программирования своей простотой и наглядностью.

Visual Basic первоначально разрабатывался для начинающих программистов, поэтому написание программ на этом языке максимально приближено к естественному языку (английскому), а обучиться этому языку гораздо проще, чем другим.

ГЛАВА 2. МОДЕЛИРОВАНИЕ И РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПРЕОБРАЗОВАТЕЛЕЙ ИНФОРМАЦИОННО- ИЗМЕРИТЕЛЬНОЙ СИСТЕМЫ ВЗВЕШИВАНИЯ ЖЕЛЕЗНОДОРОЖНЫХ ВАГОНОВ

2.1. Моделирование информационно-измерительных систем

Моделирование можно разделить на два вида - аналитическое и имитационное моделирование. При аналитическом моделировании изучаются математические модели реального объекта в виде различных уравнений, предусматривающих осуществление однозначной вычислительной процедуры, приводящей к их точному решению. При имитационном моделировании математические модели исследуются в виде алгоритма, воспроизводящего функционирование исследуемой системы путем последовательного выполнения большого количества элементарных операций [1].

Различные сферы применения компьютерных моделей предъявляют разные требования к надежности получаемых с их помощью результатов. Для моделирования зданий и деталей самолетов требуется высокая точность и степень достоверности, тогда как модели эволюции городов и социально-экономических систем используются для получения приближенных или качественных результатов.

Формализованные математические модели, пройдя несколько стадий исторических изменений, приобрели дополнительные компоненты своего построения, такие как программный, алгоритмический и аппаратный способы своей реализации. Современный уровень развития технологий позволяет говорить о компьютерных моделях и компьютерном моделировании как о методе с большей степенью обобщения, в рамках которого можно зафиксировать образующие факторы и современные тенденции [4].

Процесс создания компьютерной модели включает в себя несколько этапов. К основным этапам компьютерного моделирования относятся:

- постановка задачи;
- разработка модели;
- компьютерный эксперимент;
- анализ результатов моделирования.

Первым этапом компьютерного моделирования является постановка задачи [2]. Под задачей моделирования в самом общем смысле понимается какая-либо проблема, которую нужно решить. На этапе постановки задачи необходимо отразить три основных момента:

- описание задачи;
- определение целей моделирования;
- анализ объекта или процесса.

По характеру постановки все задачи можно разделить на два типа:

- как изменятся характеристики исследуемого объекта или процесса, при некотором воздействии на него;
- какое воздействие нужно оказать, чтобы произвести изменение характеристик объекта или процесса до определенных значений.

Цель моделирования различается в каждом конкретном случае. Целью может являться познание окружающего мира, создание объектов с определенными свойствами, изучение последствий воздействия на объект, эффективность управления объектом или процессом и так далее [11].

После определения цели моделирования необходимо выполнить анализ изучаемого объекта. Результат анализа объекта появляется в процессе выявления составляющих его частей (элементарных объектов), связей между ними и различных видов взаимодействия между частями объекта.

Следующим после постановки задачи является этап разработки модели, который можно разделить на несколько основных стадий:

- построение описательной информационной модели;
- построение формализованной (знаковой) модели;
- построение компьютерной модели.

Построение информационной (описательной) модели является отправным пунктом разработки модели. На этой стадии все входные параметры объектов, выделенные на этапе анализа, располагают в порядке убывания значимости и упрощают модель в соответствии с целью моделирования. Все элементарные объекты, выделенные при анализе, должны быть показаны во взаимосвязи друг с другом. В информационной модели отображаются только бесспорные связи и очевидные действия [5].

Такая модель выделяет существенные, с точки зрения целей проводимого исследования, параметры объекта, а несущественными параметрами пренебрегает. Информационная модель дает первичную идею, определяющую дальнейший ход моделирования.

На второй стадии создается формализованная (знаковая) модель, то есть описательная информационная модель записывается с помощью какого-либо формального языка [17].

В такой модели взаимосвязи элементарных объектов и соотношения между начальными и конечными значениями свойств объектов описываются с помощью формул, уравнений, неравенств и т.д. Также на этой стадии накладываются ограничения на допустимые взаимосвязи между объектами и значения их свойств.

Не всегда удается найти формулы, точно описывающие моделируемый объект или процесс, а также явно выражающие искомые величины через исходные данные. В таких случаях используют приближенные математические методы, позволяющие получать результаты с некоторой точностью [12].

На третьей стадии формализованную информационную модель преобразовывают в компьютерную модель, т.е. выражают ее с помощью языка понятного для компьютера. Существуют различные пути построения компьютерных моделей, например:

- создание компьютерной модели в виде проекта на одном из языков программирования;

- построение компьютерной модели с применением электронных таблиц, систем компьютерного черчения, систем управления базами данных, геоинформационных систем и т.д.

Важным элементом процесса создания компьютерной модели является разработка графического интерфейса. Удобный графический интерфейс позволит визуализировать формальную модель и даст возможность реализовать интерактивный диалог человека с компьютером во время исследования модели [14].

Третьим этапом компьютерного моделирования является компьютерный (вычислительный) эксперимент [19]. В помощь экспериментальным образцам и испытательным стендам, а иногда и для их замены используют компьютерные исследования моделей. Этап проведения компьютерного эксперимента включает две стадии: составление плана моделирования и технологию моделирования.

Составленный исследователем план моделирования должен четко отражать последовательность работы с моделью. Обычно план моделирования включает в себя тестирование модели.

Тестирование - проверка правильности функционирования модели. Обычно для теста используется набор исходных данных, для которых заранее известен результат. После тестирования, когда есть уверенность в том, что модель функционирует корректно, можно переходить к технологии моделирования.

Технология моделирования - совокупность действий пользователя над компьютерной моделью для достижения необходимого результата [3].

Если компьютерная модель выполнена в виде проекта на одном из языков программирования, ее можно запустить на выполнение, ввести исходные данные и получить результаты или наблюдать изменение объекта и характеризующих его величин [15].

Если компьютерная модель исследуется с помощью сторонних приложений, например, в электронных таблицах, то можно построить

диаграмму или график, провести сортировку и поиск данных или использовать другие специализированные методы обработки данных [10].

2.1.1. Устройства сбора, первичной обработки и передачи информации в информационно–измерительных системах

Усовершенствования функционирования моделируемой информационно-измерительной системы взвешивания будет осуществляться посредством использования более чувствительных преобразователей. На вход любой информационно-измерительной системы взвешивания поступает аналоговый сигнал, сформированный информационным устройством (датчиком), являющимся источником данных. Датчик малых перемещений большой чувствительности, является первичным преобразователем информационно-измерительной системы взвешивания.

В ИИС используются различные типы преобразователей: емкостные, индуктивные, индукционные, термоэлектрические, реостатные, тензопреобразователи, магнитоупругие, пьезоэлектрические и т.д. [16] Рассмотрим подробнее индуктивные преобразователи.

Устройства, преобразующие значение измеряемой (механической) величины в значение индуктивности, называют индуктивными преобразователями [8]. На рисунке 6 изображен самый распространенный индуктивный преобразователь с малым воздушным зазором δ , изменяющимся под действием измеряемой величины P (сосредоточенная сила, давление, линейное перемещение).

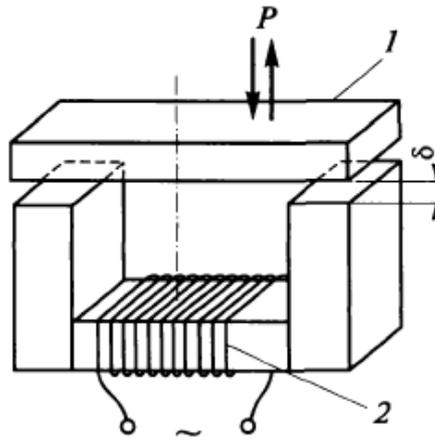


Рисунок 6 - Индуктивный преобразователь

После изменения зазора меняется магнитное сопротивление магнитной цепи, а, следовательно, и индуктивность обмотки 2, надетой на сердечник и включенной в цепь переменного тока. Индуктивность этой обмотки:

$$L = \frac{w^2}{R_M} = \frac{w^2}{R_{M.СТ} + R_\delta} = \frac{w^2}{R_{M.СТ} + \frac{2\delta}{\mu_0 S}} \quad (1)$$

где w – число витков катушки;

R_M – полное сопротивление магнитной цепи;

$R_{M.СТ}$ – магнитное сопротивление участков из стали;

R_δ – магнитное сопротивление воздушных зазоров;

S – площадь воздушного зазора;

μ_0 – магнитная постоянная, $\mu_0 = 1,26 \cdot 10^{-6}$ Гн/м.

Таким образом, у индуктивного преобразователя входной величиной является перемещение подвижной части сердечника l , а изменение индуктивности обмотки является выходной величиной. Между измеряемой механической величиной P и электрическим сопротивлением Z преобразователя возникает функциональная зависимость ($z=f(P)$ и $\Delta Z=f(\Delta P)$) из-за того, что изменение индуктивного сопротивления катушки ведет к изменению ее полного сопротивления Z [6].

Индуктивный датчик изображенный на рисунке 7 может применяться для измерения давления. Измеряемое давление воздействует на тонкую мембрану $б$ (в манометрах давление воздействует через трубку 1),

припаянную к корпусу 2 преобразователя. Магнитный поток, создаваемый катушками 5, замыкается через сердечник 3, стаканы 4 и мембрану 6.

При воздействии измеряемого давления на мембрану с одной стороны мембрана прогибается, и магнитное сопротивление для потока одной катушки уменьшается, а для потока другой катушки увеличивается. Катушки преобразователя включаются в соседние плечи моста, благодаря чему устраняются погрешности, обусловленные влиянием изменения окружающей температуры [18].

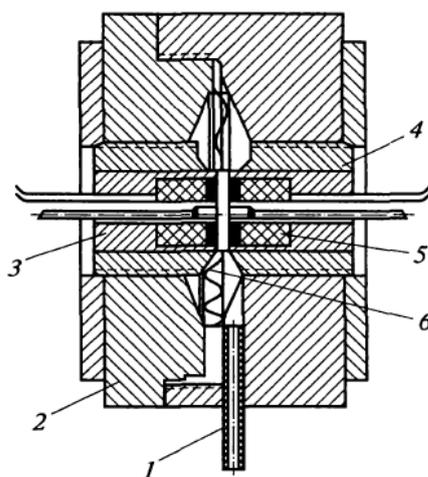


Рисунок 7 -Индуктивный датчик манометра.

1 – трубка; 2 – корпус; 3 – сердечник; 4 – стакан; 5 – катушка; 6 - мембрана

Мембраны датчиков относят к тонким пластинам, расчет которых ведется на основе методов проектирования тонкостенных оболочек. В настоящее время разработаны различные инженерные методы расчета упругих элементов, но большинство этих методов основано на ряде заранее принятых допущений, позволяющих упростить методики расчета. Однако часто это является причиной большого различия между расчетными и опытными характеристиками, полученными в результате эксперимента [7].

По сравнению с плоскими мембранами, гофрированные допускают максимальный прогиб при меньших рабочих напряжениях в материале и имеют более линейную зависимость прогиба $W=F(P)$, где P – усилие, прикладываемое к центру мембраны изображено на рисунке 8.

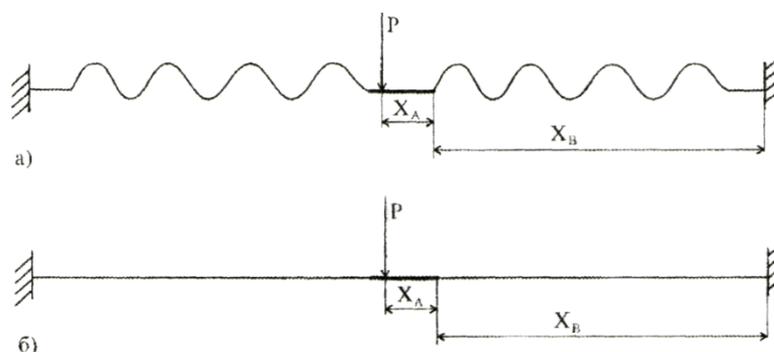


Рисунок 8 - Схемы гофрированной (а) и плоской (б) мембраны

Данные экспериментов и расчеты демонстрируют зависимость между усилием P и прогибом центра мембраны W в виде:

$$P = E A \frac{W}{h} + B \frac{W^2}{h} + C \frac{W^3}{h} \quad (2)$$

где E – модуль упругости материала мембраны, H/m^2 ;

A, B, C – коэффициенты, зависящие от толщины h , радиуса мембраны X_B , типа и глубины гофров, количества гофр и вида заделки мембраны.

Выражение (2) при параметрах мембраны: $l/h \geq 16$ и $H/h \leq 2,5$ дает точность расчета в пределах 15%.

Сложность расчета уравнений, основанных на теории тонких оболочек, объясняется большим числом параметров модели (толщина, форма, радиус изгиба, наличием краевых эффектов и т.д.). Существующие методы расчета и конструирования тонкостенных упругих элементов не позволяют с определенной точностью предсказать их параметры и характеристики, а также требуют дорогостоящих экспериментальных работ, подтверждающих достоверность расчетов [21].

Использование компьютерных процедур для решения моделей позволяет с высокой точностью получать параметры и характеристики деформационного состояния тонкостенных оболочек.

На вход любой информационно-измерительной системы взвешивания поступает аналоговый сигнал, сформированный датчиком (преобразователем), являющимся источником данных. В ИИС используются различные типы преобразователей, но в работе рассматриваются

индуктивные преобразователи, которые преобразуют значение измеряемой величины в значение индуктивности.

Большинство используемых в настоящее время методов расчета упругих элементов основано на ряде заранее принятых допущений, позволяющих упростить методики расчета, но приводящих к большому различию между расчетными характеристиками и полученными в результате эксперимента.

Сложность расчета уравнений, основанных на теории тонких оболочек, объясняется большим числом параметров модели (толщина, форма, радиус изгиба, наличие краевых эффектов и т.д.). Использование компьютерных процедур для решения моделей позволяет с высокой точностью получать параметры и характеристики деформационного состояния тонкостенных оболочек.

2.1.2. Методологические особенности поосного взвешивания железнодорожных вагонов

Проблему учета грузовых потоков предприятия можно успешно решить, используя весоизмерительную систему. Если основным способом доставки грузов является железнодорожный транспорт, то основой весоизмерительной системы являются железнодорожные весы. Несмотря на достаточно высокую стоимость, при большом грузопотоке весы окупятся быстро.

Чтобы использование весов было выгодным для владельца, необходимо учесть следующие особенности производства и оценить следующие факторы:

- сезон максимального товарооборота;
- площадь и расположение имеющейся свободной территории;
- количество используемого транспорта;
- необходимую точность учета товарных потоков.

В последнее время механические весы практически полностью сменили электронные устройства. Усилие от взвешиваемого груза поступает на датчики, после этого преобразуется в электрический сигнал, который передается на электронный индикатор. Электронный индикатор выдает информацию о массе груза на дисплей, предварительно преобразовав электрический сигнал в цифровой код.

Весы по функциональным возможностям подразделяют на динамические (изготавливаются по ГОСТ 30414–96 «Весы для взвешивания транспортных средств в движении. Общие технические требования») и статические (по ГОСТ 29329–92 «Весы для статического взвешивания. Общие технические требования»). Весы всех типов должны соответствовать требованиям пылевлагозащищенности, ударопрочности и выдерживать рабочие температуры широкого диапазона.

При статическом измерении веса вагон обычно находится на платформе весов целиком, хотя можно взвешивать его и по частям с дальнейшим суммированием показаний. Статическое взвешивание характерно для предприятий:

- требующих более точного взвешивания (погрешность примерно 0,02%);
- с достаточно большой свободной территорией;
- использующих системы дозирования грузов (зерно, гравий, песок, жидкости и т. п.).

Взвешивание в динамике обычно применяется на предприятиях, не нуждающихся в очень точном измерении, т.е. для некоммерческого взвешивания (погрешность около 1%).

Применение динамических железнодорожных весов существенно повышает эффективность использования транспортных средств, снижает простои. Скорость движения состава при проведении измерений должна составлять от 2 до 5 км/ч.

В последнее время широкое распространение получили мобильные весы. Такие весы легко перемещаются, не требуют больших площадок и существенно дешевле полнопрофильных стационарных весов.

Мобильные весы разделяются на два основных типа:

- для поколесного взвешивания;
- для поосного (совмещение 2-х, 3-х осных мостов) взвешивания.

Взвешивание транспорта с совмещенными осями на поколесных весах приводит к большим ошибкам из-за влияния подвески вагона на результат взвешивания, поэтому очевидно преимущество поосных весов перед поколесными [22].

Взвешивание на поосных весах происходит следующим образом: производится последовательный наезд всех мостов вагона на установленные весоизмерительные платформы с последующим суммированием результата взвешивания осей. При наезде очередного моста на весоизмерительные платформы происходит перераспределение составляющих вектора центра тяжести, что может вносить дополнительные погрешности во взвешивании.

При взвешивании на мобильных весах жидких грузов, может происходить смещение груза и соответственно центра тяжести вагона. При взвешивании вагона с грузом, имеющим высокий центр тяжести (объемные конструкции, автокраны и т.д.) возникает перекося, что также приводит к перераспределению составляющих вектора центра тяжести. При взвешивании прицепов на взвешивание оказывает влияние пружина, находящаяся в системе крепления прицепа. Оси, не стоящие на весах во время взвешивания, находятся либо на пандусах, либо на неровной площадке, из-за чего появляется еще одна сила, которая оказывает влияние на взвешивание. Возникающие ошибки практически невозможно учесть [20].

Исходя из перечисленных особенностей поосных весов можно сделать вывод, что применять поосный метод для коммерческого взвешивания грузов неэффективно. При поосном взвешивании даже самых дешевых коммерческих грузов, таких как песок и щебень, у потребителей регулярно

возникают вопросы относительно правильности их учета – квартальные отчеты вскрывают большие «недостачи».

Нагрузка на ось (осевая нагрузка) – это нагрузка от массы вагона, передаваемая на поверхность колесами одной оси. Масса вагона (снаряженная или полная) и нагрузки на его оси связаны соотношением:

$$M_o = N_{o1} + N_{o2} + \dots + N_{oi}, \quad (3)$$

где M_o – общая масса вагона, а N_{oi} – сила давления колес i -ой оси на рельсы (осевая нагрузка).

Рассмотрим подробнее процессы поосного одновременного (ОВ) и последовательного взвешивания (ПВ) весами, состоящими из двух или одной пары грузоприемных платформ.

Полная масса вагона при одновременном измерении осевых нагрузок двумя парами грузоприемных платформ (рисунок 9) равна сумме измеренных осевых нагрузок:

$$M_o = N_{o1} + N_{o2} \quad (4)$$

где M_o – общая масса вагона, а N_{oi} – сила давления колес i -ой оси на рельсы (осевая нагрузка).

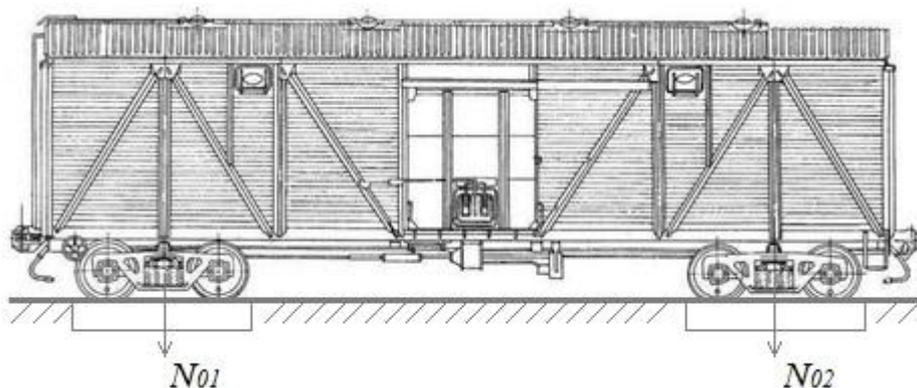


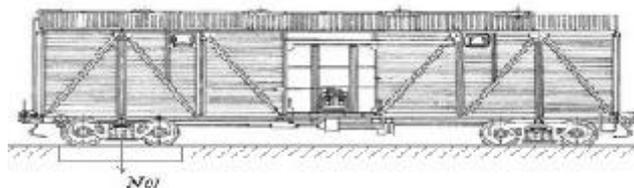
Рисунок 9 - Одновременное измерение осевых нагрузок двумя парами грузоприемных платформ

Величина M_o с точностью до величины погрешностей весов будет равна полной массе вагона, определенной на обычных полноразмерных весах статического взвешивания.

При последовательном взвешивании вагон сначала наезжает передней осью на пару весовых платформ и производится статическое измерение

нагрузки на рельсы, создаваемой первой осью – $N_{П1}$ (рисунок 10, А). На втором шаге вагон заезжает на пару весов второй осью и определяется нагрузка $N_{П2}$ (рисунок 10, Б).

А)



Б)

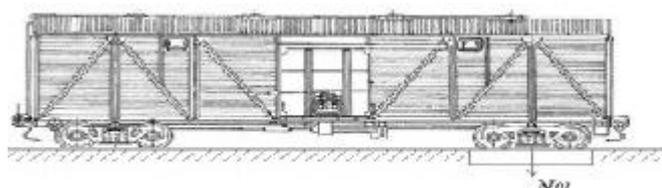


Рисунок 10 - Взвешивание вагона путем последовательного измерения осевых нагрузок

В результате перестановки вагона дважды менялось распределение нагрузок между осями, поэтому их сумма, в общем случае, не будет равна полной массе вагона:

$$M_{П} = N_{П1} + N_{П2} \neq M_{O}, \quad (5)$$

Также не будут равны и сами величины осевых нагрузок:

$$N_{O1} \neq N_{П1}, N_{O2} \neq N_{П2}. \quad (6)$$

Эти неравенства возникают не из-за недостатков весов или процессов взвешивания, а из-за особенностей элементов подвески вагона.

С целью разгрузки рессор от действия горизонтальных сил, возникающих в процессе разгона, торможения или маневрирования вагона, к ним шарнирно крепятся реактивные тяги или рычаги. Возникающие в них реакции, а также силы трения покоя определяют случайное перераспределение нагрузок между осями одной тележки, обнаруживаемое при неоднократном взвешивании вагона.

Процесс поосного статического взвешивания двухосных вагонов лишен упомянутых выше недостатков. Для него всегда (в пределах погрешностей весов) будут выполняться равенства:

$$M_O = N_{O1} + N_{O2} = M_{\Pi}; \quad (7)$$

$$N_{O1} = N_{\Pi1}, N_{O2} = N_{\Pi2}. \quad (8)$$

Проблема учета грузовых потоков предприятия успешно решается путем использования весоизмерительных систем. В последнее время произошел почти полный переход с механических весов на электронные устройства.

В таких устройствах усилие от взвешиваемого груза поступает на датчики, после чего преобразуется в электрический сигнал. Далее электрический сигнал преобразуется в цифровой код и информация о массе груза выводится на дисплей.

Таким образом, в первой главе был сформирован теоретико-методологический аппарат построения компьютерной модели информационно-измерительной системы.

Проанализированы этапы процесса создания компьютерной модели, изучены особенности выполнения каждого этапа и его результатов.

Рассмотрено типовое строение ИИС, его функционирование и различные преобразования информации в ИИС с момента поступления на вход через датчик до вывода на дисплей.

Изучено строение индуктивного датчика, используемого во многих ИИС. Выполнен анализ различных методов расчета упругих элементов и сформированы уравнения в линейной и нелинейной формах для расчета мембраны индуктивного преобразователя. Описаны сложности расчета уравнений, основанных на теории тонких оболочек, и причины возникновения различий между расчетными характеристиками и полученными в результате эксперимента.

Рассмотрены методологические особенности поосного взвешивания железнодорожного транспорта. Определены области эффективного и неэффективного использования такого метода взвешивания. Получены данные необходимые для построения компьютерной модели

информационно-измерительной системы взвешивания железнодорожного транспорта.

2.1.3. Построение модели информационно-измерительной системы «ТО-ВЕ»

На основании выявленных недостатков и сделанных выводов построена концептуальная модель бизнес-процессов ТО-ВЕ. На рисунке 11 представлена контекстная диаграмма процесса «Взвешивание транспорта с использованием ИИС»

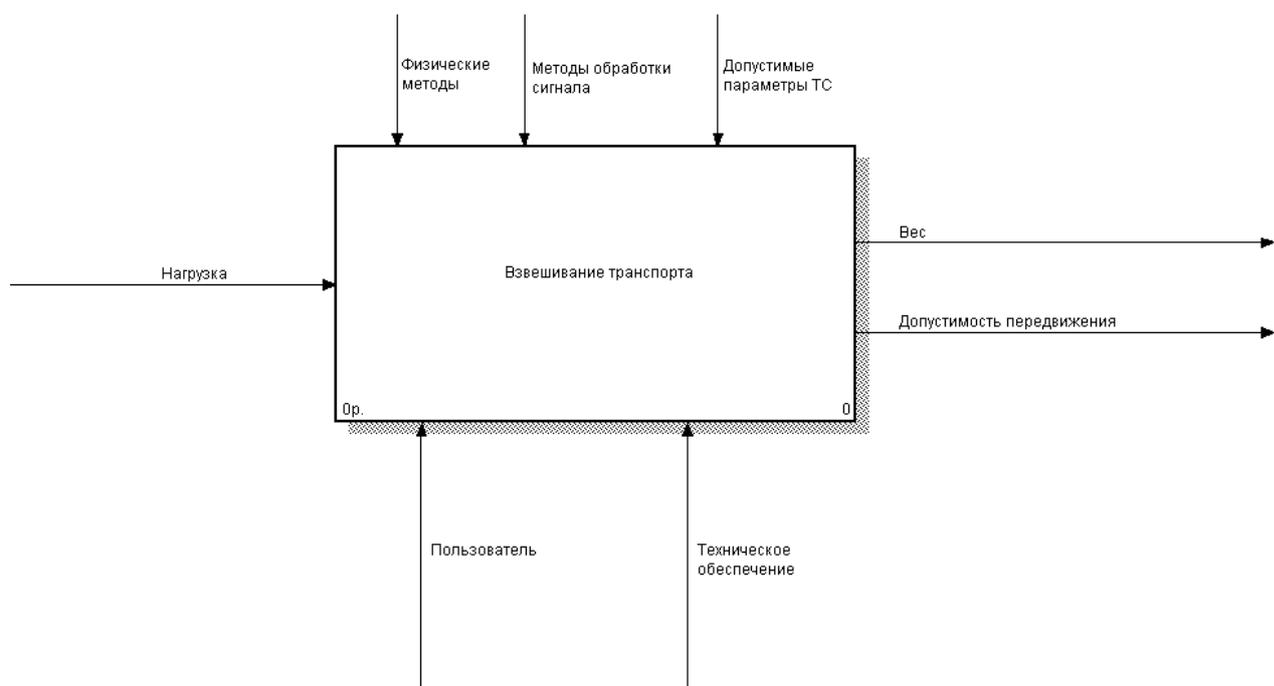


Рисунок 11 - Контекстная диаграмма процесса «Взвешивание транспорта с использованием ИИС» (ТО-ВЕ)

В отличие от контекстной диаграммы, представленной на рисунке 4, новая диаграмма включает в себя вывод информации о допустимости передвижения транспорта по дорогам выбранной категории.

После декомпозиции (рисунок 12) видно, что процесс «Взвешивание транспорта с использованием ИИС» состоит из следующих подпроцессов:

- сбор данных;
- подготовка данных;
- передача данных;

- обработка данных;
- вывод данных.

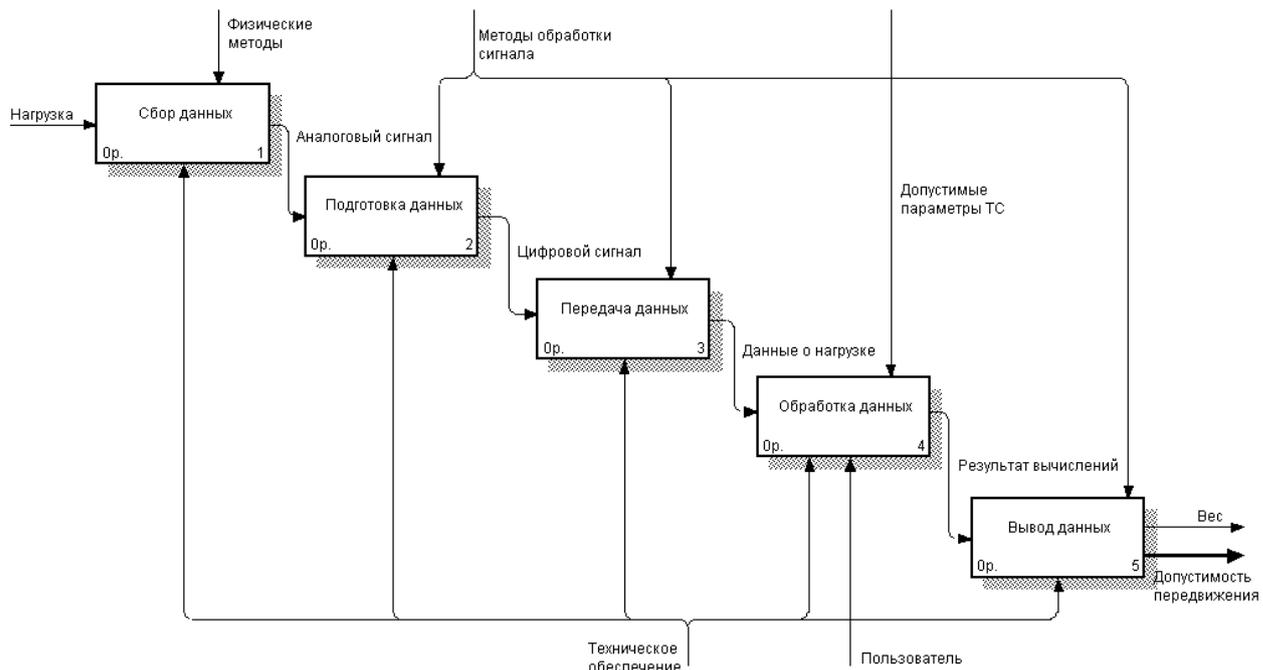


Рисунок 12 - Декомпозиция процесса «Взвешивание транспорта с использованием ИИС» (ТО-ВЕ)

Детальные декомпозиции подпроцессов взвешивания транспорта с использованием ИИС представлены на рисунках 13-17.

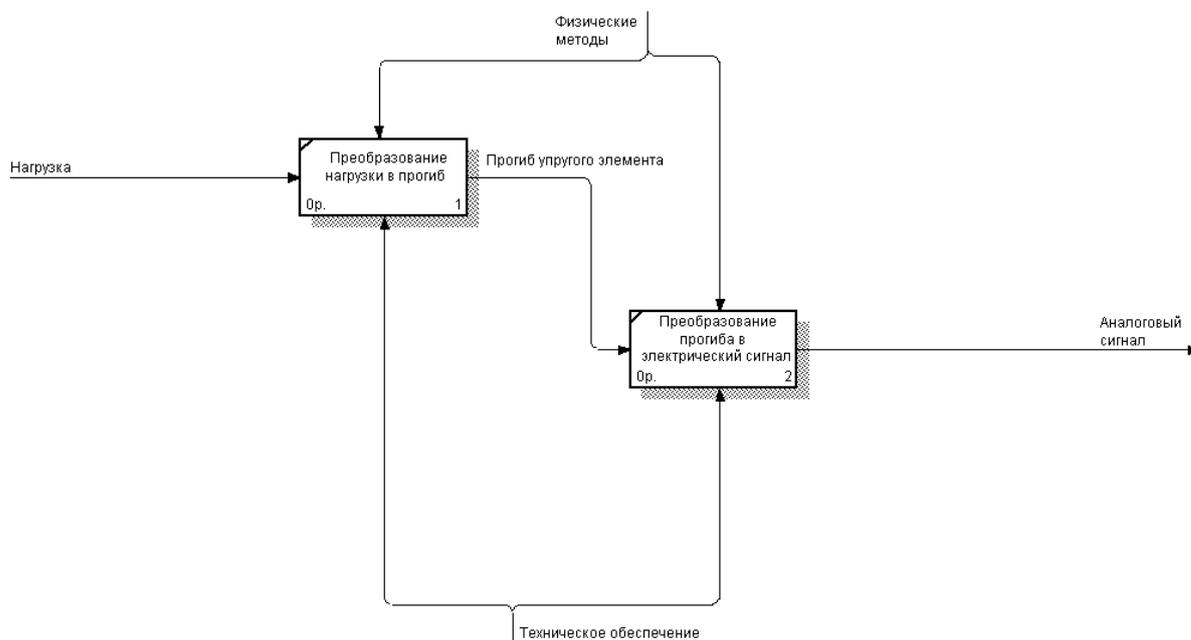


Рисунок 13 - Декомпозиция процесса «Сбор данных» (ТО-ВЕ)

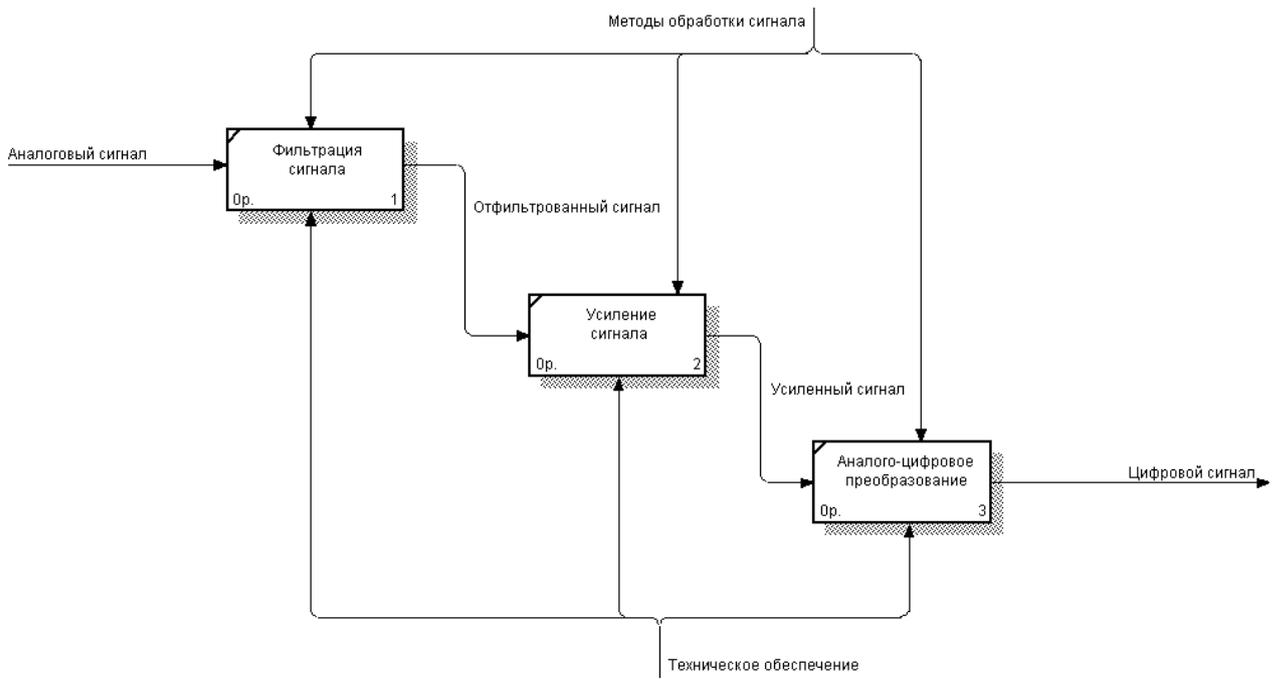


Рисунок 14 - Декомпозиция процесса «Подготовка данных» (ТО-ВЕ)

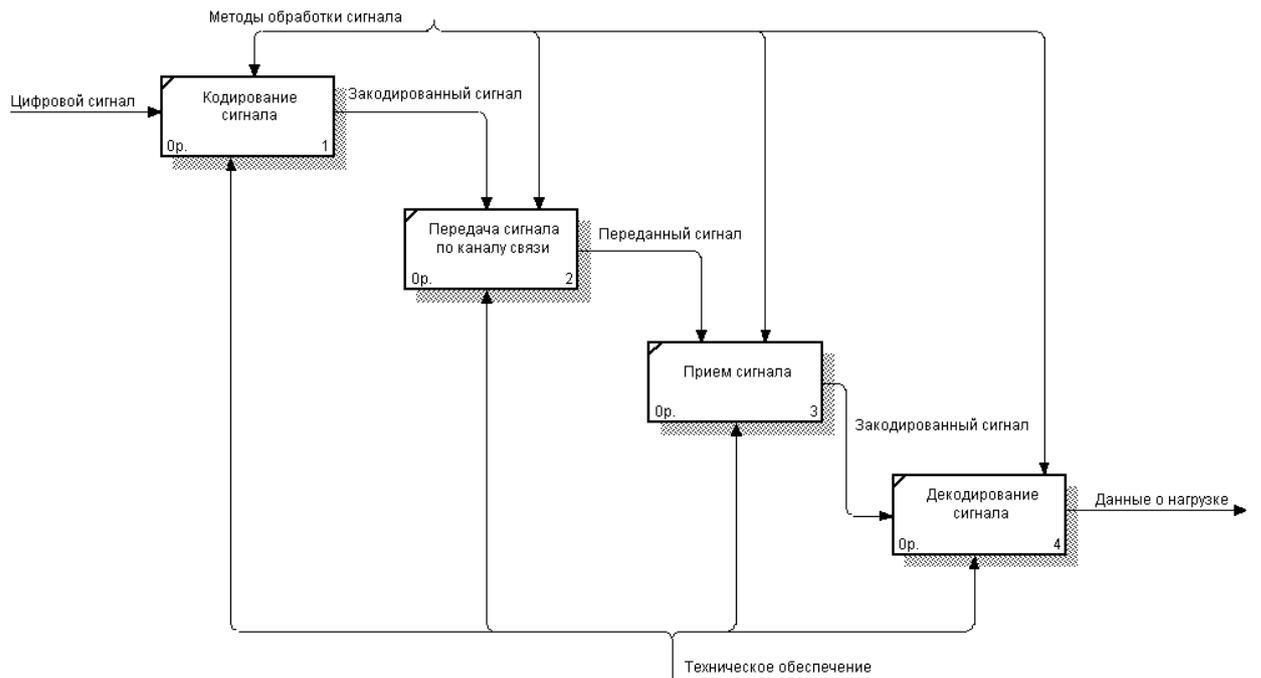


Рисунок 15 - Декомпозиция процесса «Передача данных» (ТО-ВЕ)

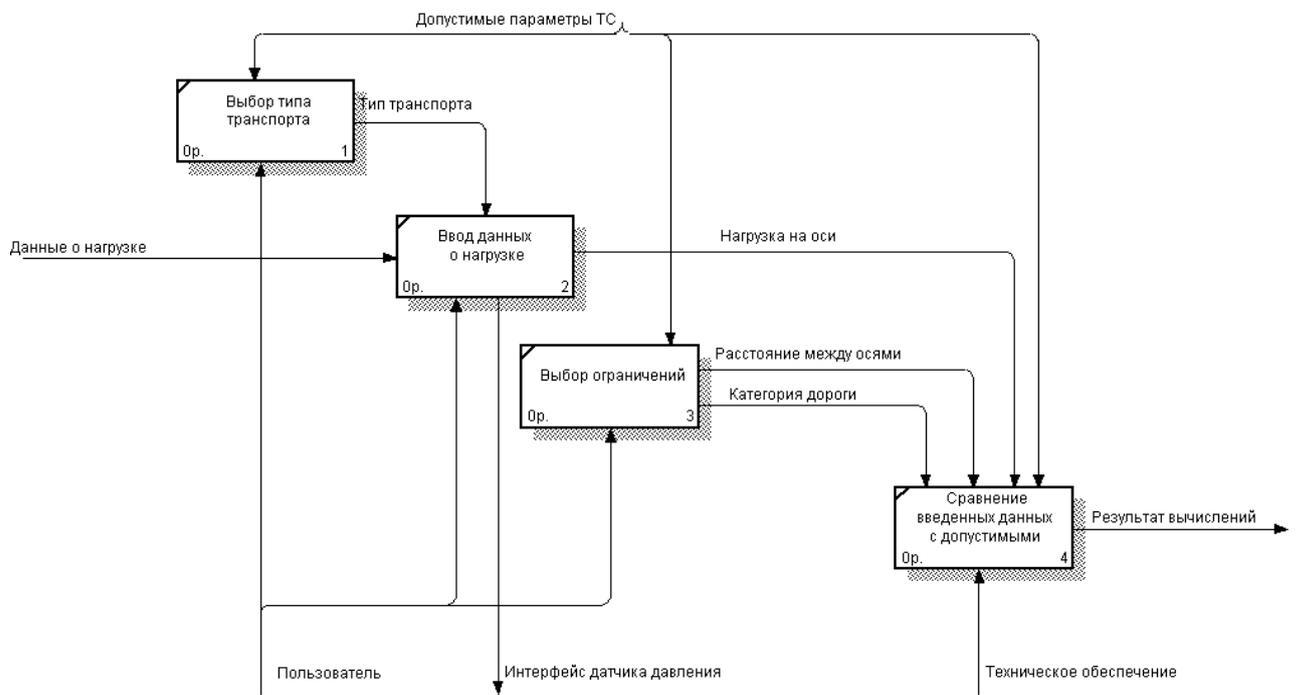


Рисунок 16 - Декомпозиция процесса «Обработка данных» (ТО-ВЕ)

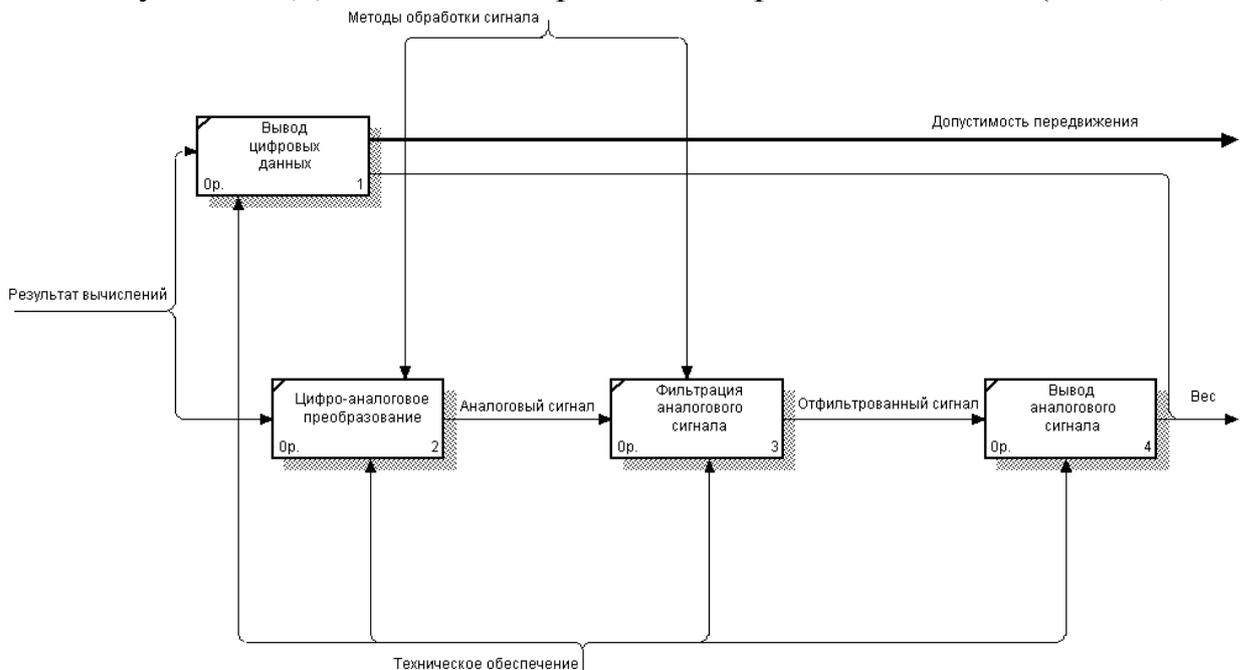


Рисунок 17 - Декомпозиция процесса «Вывод данных» (ТО-ВЕ)

Построенная модель AS-IS позволяет четко зафиксировать, какие процессы осуществляются во время взвешивания транспорта, какие информационные объекты используются при выполнении различных функций на всех уровнях детализации.

На основе недостатков, обнаруженных в модели AS-IS, построена модель TO-BE, позволяющая описать, как должны происходить различные процессы взвешивания с использованием ИИС.

Наибольшее внимание в работе уделено процессам «Сбор данных» и «Обработка данных». Использование для сбора данных об исследуемом объекте датчиков, чувствительные элементы которых спроектированы на основе уравнений, приведенных в первой главе данной работы, позволяет повысить точность измерений. А использование разрабатываемой автоматизированной подсистемы обработки данных позволяет значительно снизить время, затраченное на обработку, и уменьшить количество ошибок при обработке данных.

На основе построенных диаграмм (рисунок 11-17) разработаны алгоритмы функций, выполняемых автоматизированной системой обработки данных ИИС.

На рисунке 18 представлена декомпозиция процесса «Обработка данных», описывающая все подпроцессы, выполняемые во время обработки. Из рисунка видно, что обработка данных включает в себя следующие подпроцессы:

- выбор типа транспорта;
- ввод данных о нагрузке;
- выбор ограничений;
- сравнение введенных данных с допустимыми.

Процесс «Выбор типа транспорта» заключается в выборе одного из возможных типов транспорта (двухосный или трехосный) для последующего ввода данных о нагрузке на каждую ось и выбора допустимых характеристик, соответствующих выбранному типу. Алгоритм этого процесса представлен на рисунке 18.



Рисунок 18 - Алгоритм процесса «Выбор типа транспорта»

После выбора одного из доступных типов транспорта появляется окно для ввода данных о нагрузке на каждую ось и начинает выполняться следующий процесс «Ввод данных о нагрузке».

Данные о нагрузке на каждую из осей, полученные с железнодорожных весов вводятся в соответствующие окна для сравнения их с допустимыми значениями и определения возможности передвижения транспорта. Алгоритм процесса «Ввод данных о нагрузке» представлен на рисунке 19.

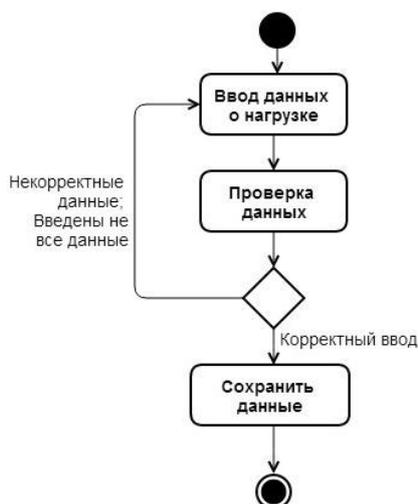


Рисунок 19 - Алгоритм процесса «Ввод данных о нагрузке»

В разрабатываемой подсистеме обработки данных предусмотрена возможность получения данных о нагрузке со спроектированного интерфейса датчика давления. Процесс получения данных о взвешиваемом объекте спроектированным датчиком отличается от автоматического

получения данных с железнодорожных весов. Алгоритм сбора данных разработанным датчиком представлен на рисунке 20.

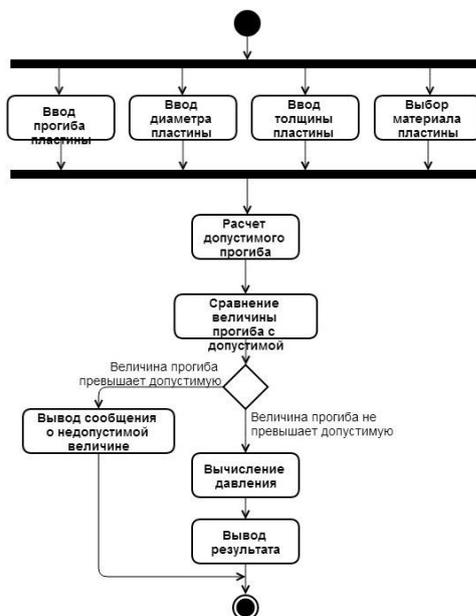


Рисунок 20 - Алгоритм получения данных разработанной моделью датчика

Если данные введены корректно, начинается процесс «Сравнение введенных данных с допустимыми». Алгоритм этого процесса изображен на рисунке 21.

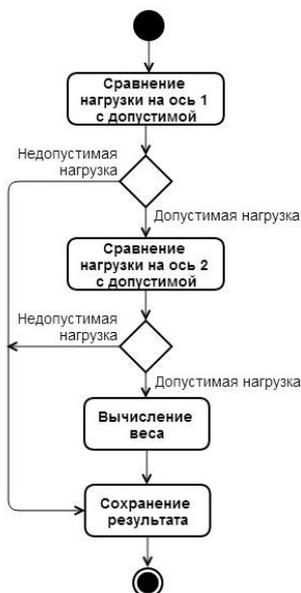


Рисунок 21 - Алгоритм процесса «Сравнение введенных данных с допустимыми»

После сравнения исходных данных о нагрузке с ограничениями, выводится сообщение о допустимости передвижения, а также выводятся на экран вес транспортного средства, нагрузки на каждую из его осей и

допустимые нагрузки на оси. Процесс вывода данных осуществляется по алгоритму, представленному на рисунке 22.

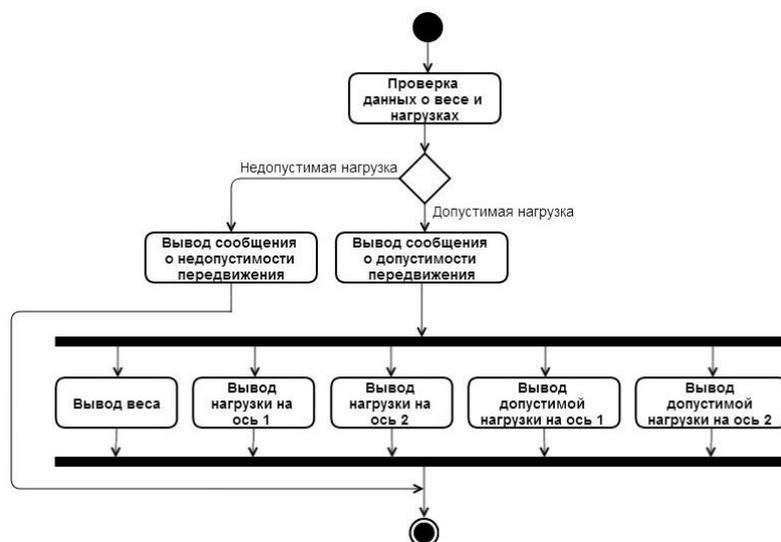


Рисунок 22 - Алгоритм процесса «Вывод цифровых данных»

С помощью построенных моделей и диаграмм можно детально рассмотреть процессы, происходящие во время взвешивания железнодорожного транспорта с использованием ИИС, увидеть недостатки типовых ИИС и альтернативные схемы осуществления процесса взвешивания железнодорожного транспорта с использованием ИИС.

Также, с их помощью, на языке Visual Basic была разработана автоматизированная подсистема обработки данных ИИС, сконцентрированная на процессах сбора и обработки данных (Приложение А). Ее использование позволяет значительно снизить время, затраченное на обработку, и уменьшить количество ошибок при обработке данных, а также, дает возможность детально рассмотреть выполнение каждого процесса, происходящего во время обработки данных при взвешивании железнодорожных вагонов с использованием ИИС.

ГЛАВА 3. ТЕСТИРОВАНИЕ И ВНЕДРЕНИЕ РАЗРАБОТАННЫХ ПРОГРАММНЫХ РЕШЕНИЙ

3.1. Тестирование программного продукта

Разрабатываемый в рамках исследования программный продукт предназначен для автоматизированной обработки данных ИИС и моделирования работы используемого в нем датчика давления.

Пользовательский интерфейс, разрабатываемой модели, представлен на рисунке 23:



Рисунок 23 - Стартовое окно подсистемы обработки данных

Чтобы начать обработку данных нужно выбрать тип транспорта, ввести значения нагрузок на оси, выбрать категорию дороги и расстояние между осями (рисунок 24).

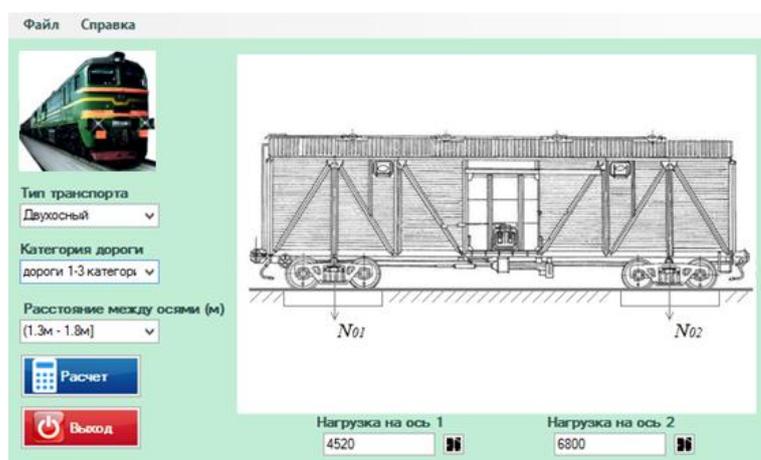


Рисунок 24 - Ввод начальных данных

Если все введенные данные корректны, то появится сообщение о допустимости (рисунок 25) или недопустимости (рисунок 26) нагрузки.

Далее появится окно, в котором отображены вес транспортного средства, нагрузка на каждую из его осей и допустимые нагрузки на оси (рисунок 27).

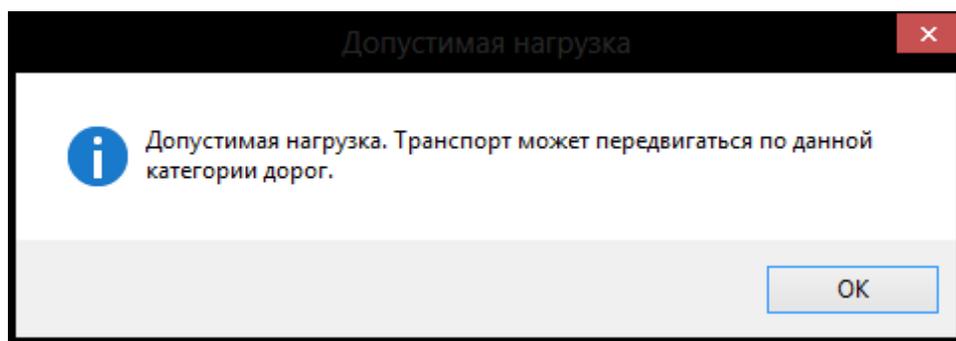


Рисунок 25 - Сообщение о допустимой нагрузке

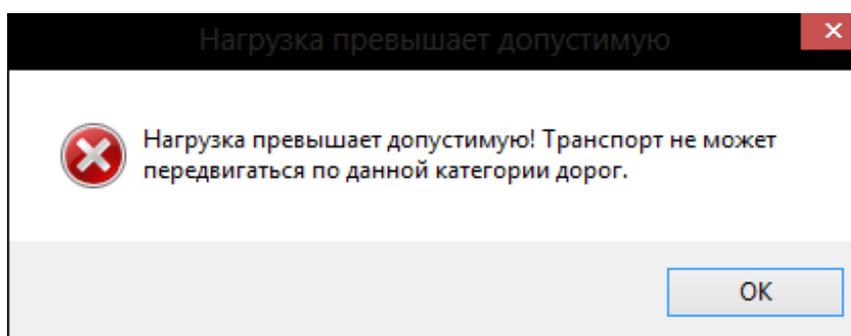


Рисунок 26 - Сообщение о недопустимой нагрузке

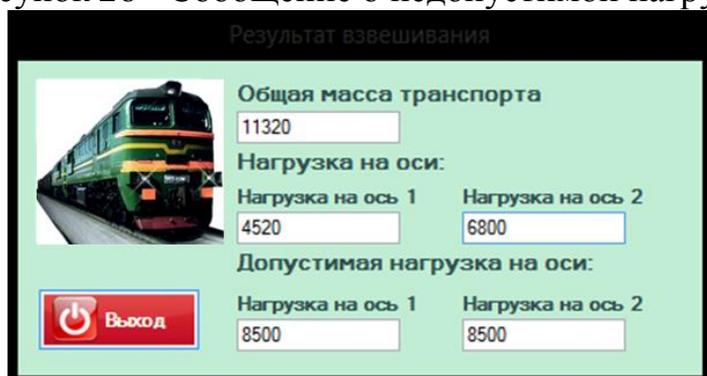


Рисунок 27 - Вывод результата

Если введены некорректные данные или введены не все данные, программы выдаст сообщение об ошибке (рисунки 28-29).

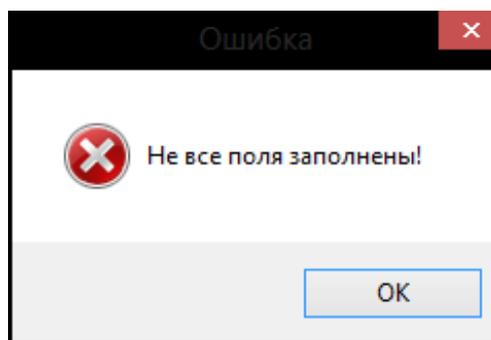


Рисунок 28 - Ошибка №1

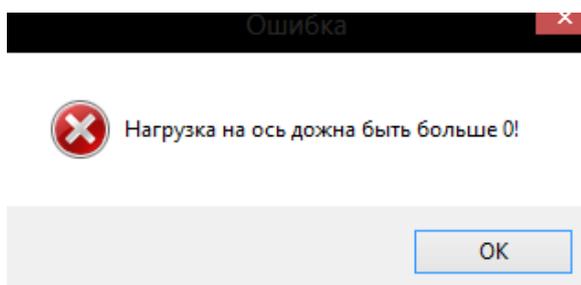


Рисунок 29 - Ошибка №2

Очистить все поля для ввода можно с помощью пункта меню "Очистить поля" (рисунок 30).

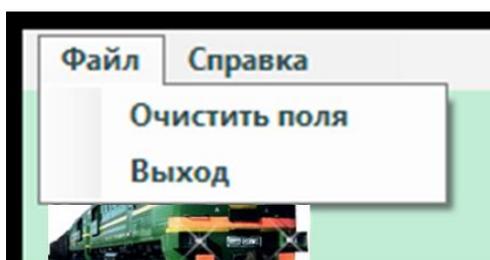


Рисунок 30 - Пункт "Очистить поля"

Также для помощи пользователям в работе с программой в разделе меню «Справка» предусмотрены пункты «Инструкция» и «О программе» (рисунок 31).

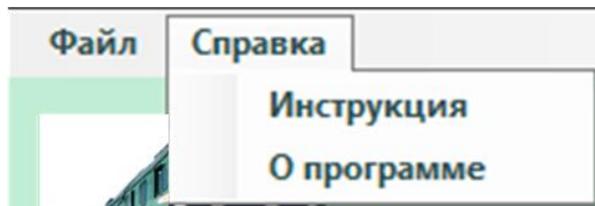


Рисунок 31 - Раздел меню "Справка"

При вводе данных о нагрузке на ось с помощью специальных кнопок (рисунок 32) можно запустить интерфейс датчика давления (рисунок 33).

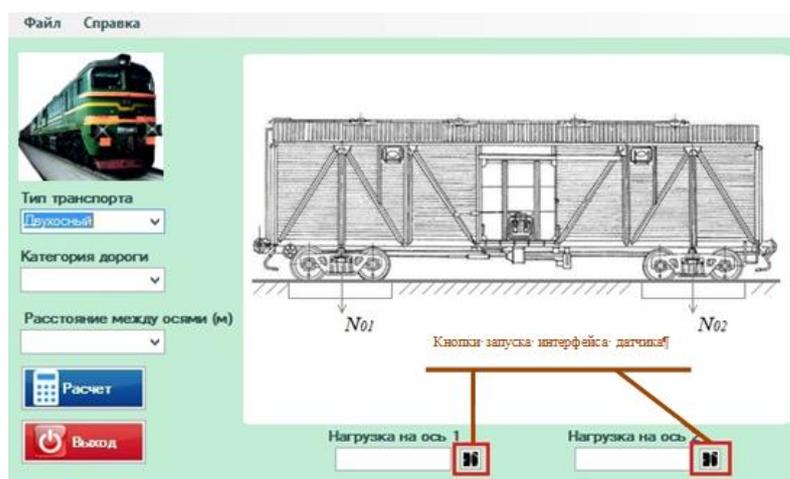


Рисунок 32 - Запуск интерфейса датчика давления

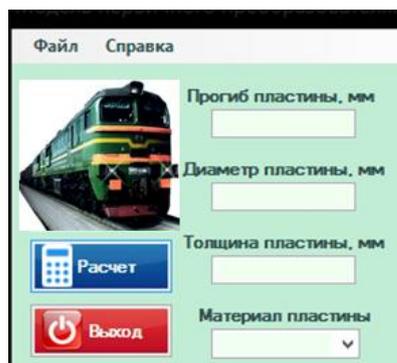


Рисунок 33 - Интерфейса датчика давления

Данная программа позволяет увидеть зависимость прогиба чувствительного элемента датчика (пластины) от усилия, приложенного к пластине и ее характеристик (материал, диаметр, толщина).

Чтобы начать моделирование нужно ввести характеристики пластины (упругого элемента): прогиб пластины, диаметр пластины, толщина пластины, материал пластины, используя соответствующие поля (рисунок 34).

Если введены некорректные данные или заполнены не все поля, появится соответствующее сообщение об ошибке (рисунки 35-36).

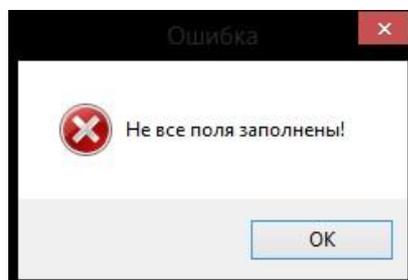


Рисунок 34 - Сообщение об ошибке №1

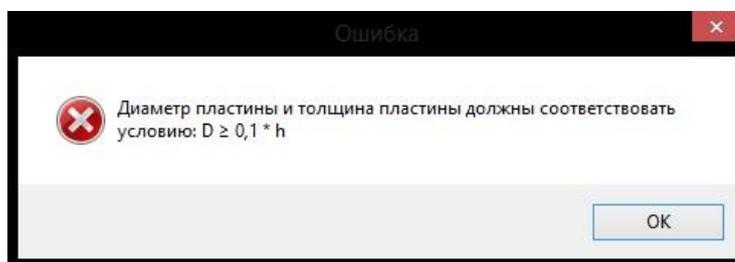


Рисунок 35 - Сообщение об ошибке №2

Если характеристики пластины введены корректно, будет вычислен допустимый для нее прогиб. Если введенный прогиб превышает допустимый, то на экране появится соответствующее сообщение (рисунок 36).

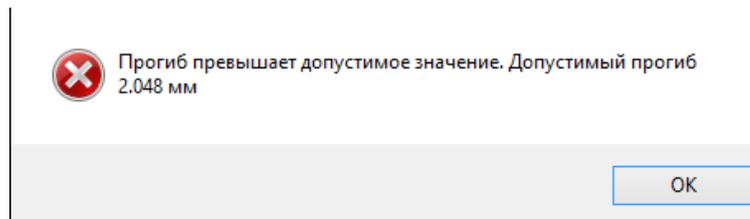


Рисунок 36 - Сообщение об ошибке №3

Если все данные введены корректно, то модель рассчитает давление, оказываемое на датчик, и выведет нагрузку на ось автомобиля в паскалях и килограммах в новом окне (рисунок 37).



Рисунок 37 - Вывод результата

Если необходимого материала нет в списке, его можно добавить с помощью модуля "Добавить материал" (рисунок 38).

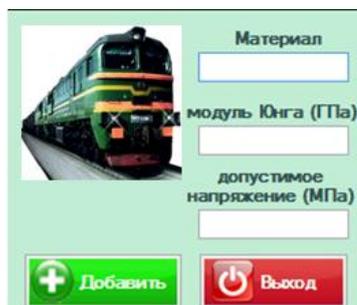


Рисунок 38 - Модуль "Добавить материал"

В этом модуле необходимо ввести название материала, модуль Юнга и допустимое напряжение для материала. Нажать кнопку "Добавить". После нажатия кнопки появится сообщение о том, что материал добавлен (рисунок 39).

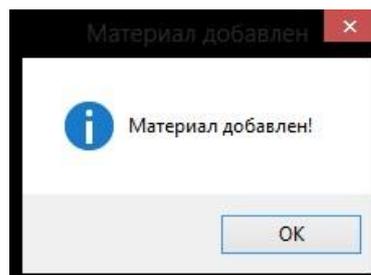


Рисунок 39 - Сообщение "Материал добавлен"

ЗАКЛЮЧЕНИЕ

Выполненный в работе анализ отечественных и зарубежных работ по компьютерному моделированию, проектированию информационных систем, расчету упругих чувствительных элементов и методологическим особенностям взвешивания железнодорожного транспорта с использованием информационно-измерительных систем позволяет сделать выводы о том, что проблема взвешивания транспорта с использованием ИИС является актуальной и включает в себя различные вопросы: методологические основы взвешивания транспорта, место ИИС в процессе взвешивания, сложность расчета упругих чувствительных элементов датчиков давления и т.д.

Особое внимание в работе уделено изучению процессов сбора и обработки данных во время взвешивания железнодорожного транспорта с использованием информационно-измерительных систем.

Использование для сбора данных об исследуемом объекте датчиков позволяет повысить точность измерений.

В процессе анализа работ, посвященных методологическим особенностям взвешивания железнодорожного транспорта, было обнаружено, что сотрудники постов весового контроля при определении допустимой осевой нагрузки пользуются специальными справочниками, где расписаны предельные нагрузки для всех типов железнодорожного транспорта. То есть они вынуждены анализировать и вручную обрабатывать данные, полученные в цифровой или аналоговой форме, что замедляет обработку данных и приводит к увеличению количества неверно обработанных данных.

Исходя из этого, был сделан вывод о необходимости всесторонне исследовать автоматизацию процессов обработки данных, полученных при взвешивании, и необходимости разработки модели подсистемы обработки данных, которая позволит решить описанные выше проблемы.

В результате исследования, были сделаны следующие выводы:

1. Компьютерное моделирование является эффективным методом изучения сложных систем. Благодаря возможности проводить вычислительные эксперименты, компьютерные модели проще и удобнее исследовать в тех случаях, когда проведение реальных экспериментов затруднено из-за финансовых, физических и других препятствий. Наряду с эффективностью компьютерное моделирование является сложным процессом. А создание компьютерной модели включает в себя несколько этапов, к которым относятся: постановка задачи, разработка модели, компьютерный эксперимент, анализ результатов моделирования.

2. Рассмотрено типовое строение ИИС, его функционирование и различные преобразования информации в ИИС с момента поступления на вход через датчик до вывода на дисплей. Получены данные, свидетельствующие о том, что ИИС, построенные на индуктивных датчиках, могут встраиваться в железнодорожную структуру. Что дает возможность определять нагрузку на ось вагона на терминалах и при движении на железной дороге.

3. Описаны сложности расчета уравнений, основанных на теории тонких оболочек, и причины возникновения различий между характеристиками, полученными в результате расчетов и экспериментов. Предложена методика расчета упругих элементов, повышающая точность измерений, а также сформированы уравнения в линейной и нелинейной формах для расчета мембраны индуктивного преобразователя.

4. Определены области эффективного и неэффективного использования поосного метода взвешивания. Рассмотрены методологические особенности поосного взвешивания железнодорожного транспорта, в результате чего получены данные, необходимые для построения компьютерной модели информационно-измерительного комплекса взвешивания железнодорожного транспорта.

5. Построены модели информационно-измерительной системы, позволяющие рассмотреть процессы, происходящие во время взвешивания железнодорожного транспорта с различным уровнем детализации.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

Научная и методическая литература

1. Антонец, И.В. Методы расчета и моделирования упругих элементов: учебное пособие / И.В. Антонец, А.П. Терешенок. – Ульяновск: УлГТУ, 2013. – 121 с.
2. Буренок, В.М. Математические методы и модели в теории информационно-измерительных систем. / В.М. Буренок, В.Г. Найденов, В.И. Поляков – М.: Машиностроение, 2011. – 336 с.
3. Волкова В.Н. Моделирование систем и процессов. / В.Н. Волкова, В.Н. Козлов. – М.: Юрайт, 2016. – 450 с.
4. Голубева, Н.В. Математическое моделирование систем и процессов. / Н.В. Голубева. – СПб.: Лань, 2016. – 192 с.
5. Горлач, Б.А. Математическое моделирование. Построение моделей и численная реализация. / Б.А. Горлач, В.Г. Шахов. – СПб.: Лань, 2016. – 292 с.
6. Датчики, электронные преобразователи, приборы экомониторинга / Казан. гос. техн. ун-т им. А. Н. Туполева ; [авт.-сост. А. Н. Глебов и др. ; под ред. А. Н. Глебова]. - 2-е изд., перераб. и доп.; ВУЗ/изд. - Казань: Экоцентр, 2006. — 131 с.
7. Клаассен, К.Б. Основы измерений. Датчики и электронные приборы: [учеб. пособие] / К.Б. Клаассен; пер. с англ. Е.В. Воронова, А.Л. Ларина. — 3-е изд. - Долгопрудный: Интеллект, 2008. — 350 с.
8. Котюк, А.Ф. Датчики в современных измерениях / А.Ф. Котюк. — М.: Радио и связь: Горячая линия - Телеком, 2006. — 96 с.
9. Макаров, В.В. Информационно-измерительные системы. / В.В. Макаров. – М.: МИЭМ НИУ ВШЭ, 2013. – 161 с.
10. Петров А.В. Моделирование процессов и систем. / А.В. Петров – СПб.: Лань, 2015. – 288 с.
11. Советов, Б.Я. Моделирование систем. / Б.Я. Советов, С.А. Яковлев. – М.: Юрайт, 2017. – 343 с.

12. Тарасевич, Ю.Ю. Математическое и компьютерное моделирование. / Ю.Ю. Тарасевич – М.: URSS, 2013. – 152 с.
 13. Тарасик, В.П. Математическое моделирование технических систем. / В.П. Тарасик – М.: Инфра-М, 2016. – 592 с.
 14. Трусов, П.В. Введение в математическое моделирование. / П.В. Трусов. – М.: Логос, 2016. – 440 с.
 15. Федоткин, И.М. Математическое моделирование технологических процессов. / И.М. Федоткин – М.: URSS, 2015. – 416 с.
 16. Фрайден, Д. Современные датчики: справочник / Д. Фрайден. — М.: Техносфера, 2006. — 592 с.
 17. Шапкин, А.С. Математические методы и модели исследования операций. / А.С. Шапкин, В.А. Шапкин. – М.: Дашков и К, 2017. – 398 с.
- Литература на иностранном языке*
18. Kalantar-zadeh K. Sensors: An Introductory Course. / Kourosh Kalantar-zadeh. – New-York:Springer, 2013. – 196 с.
 19. Meerschaert M. Mathematical Modeling. Fourth Edition. / Mark. M. Meerschaert. – London: Academic Press, 2013. – 384 с. – 4th Edition
 20. Radoičić G.Experience with an On-board Weighing System Solution for Heavy Vehicles/ G. Radoičić, M. Jovanović, M. Arsić // ETRI Journal, Vol. 38. No. 4. P. 787-797. doi:10.4218/etrij.16.0115.0183
 21. Webster J.G. Measurement, Instrumentation, and Sensors Handbook, Second Edition: Spatial, Mechanical, Thermal, and Radiation Measurement/ J.G. Webster, H. Eren // CRC Press. – 2014. – 2nd Edition
 22. Žnidarič A. Railway bridge Weigh-in-Motion system/ A. Žnidarič, J. Kalin, M. Kreslin, P. Favai, P. Kolakowski // Sciencedirect, 2016. Vol. 14. P. 4010-4019. doi:10.1016/j.trpro.2016.05.498

ПРИЛОЖЕНИЕ А

Листинг программы

```
Public Class Form1
    Dim type As String
    Dim road As String
    Dim length(4) As String
    Dim filename, foldername1, foldername2 As String
    Dim limit1(4), limit2(4) As String
    Dim mass1, mass2 As Double
    Private Sub ComboBox1_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ComboBox1.SelectedIndexChanged
        If ComboBox1.SelectedIndex = 0 Then
            PictureBox2.Visible = True
            PictureBox2.Image = System.Drawing.Image.FromFile(My.Application.Info.DirectoryPath &
"image\biaxial.jpg")
            Label4.Location = New Point(266, 352)
            TextBox4.Location = New Point(275, 368)
            Label4.Visible = True
            TextBox4.Visible = True
            Label5.Location = New Point(466, 352)
            TextBox5.Location = New Point(475, 368)
            Label5.Text = "Нагрузка на ось 2"
            Label5.Visible = True
            TextBox5.Visible = True
            Button3.Visible = True
            Button3.Location = New Point(378, 368)
            Button4.Visible = True
            Button4.Location = New Point(578, 368)
            foldername1 = "\files\biaxial\"
            foldername2 = "\files\biaxial\"
        ElseIf ComboBox1.SelectedIndex = 1 Then
            PictureBox2.Visible = True
            PictureBox2.Image = System.Drawing.Image.FromFile(My.Application.Info.DirectoryPath &
"image\triaxial.png")
            Label4.Location = New Point(246, 352)
            TextBox4.Location = New Point(255, 368)
            Label4.Visible = True
            TextBox4.Visible = True
            Label5.Location = New Point(435, 352)
            TextBox5.Location = New Point(455, 368)
            Label5.Text = "Нагрузка на оси 2 и 3"
            Label5.Visible = True
            TextBox5.Visible = True
            Button3.Visible = True
            Button3.Location = New Point(358, 368)
            Button4.Visible = True
            Button4.Location = New Point(558, 368)
            foldername1 = "\files\biaxial\"
            foldername2 = "\files\triaxial\"
        End If
    End Sub
    Private Sub ComboBox2_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ComboBox2.SelectedIndexChanged
        If ComboBox2.SelectedIndex = 0 Then
            filename = "groupA.txt"
        ElseIf ComboBox2.SelectedIndex = 1 Then
            filename = "groupB.txt"
        End If
    End Sub
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
```

```

If TextBox4.Text = "" Or TextBox5.Text = "" Or ComboBox1.SelectedIndex = -1 Or
ComboBox2.SelectedIndex = -1 Or ComboBox3.SelectedIndex = -1 Then
    MsgBox("Не все поля заполнены!", vbCritical, "Ошибка")
    Exit Sub
ElseIf Val(TextBox4.Text) <= 0 Or Val(TextBox5.Text) <= 0 Then
    MsgBox("Нагрузка на ось должна быть больше 0!", vbCritical, "Ошибка")
    Exit Sub
End If
Dim MyFile As Object
MyFile = FreeFile() 'свободный номер
Dim i As Short 'счетчик
FileOpen(MyFile, My.Application.Info.DirectoryPath & foldername1 & filename, OpenMode.Input)
'открываем файл и считаем из него информацию
For i = 1 To 4
    length(i) = LineInput(MyFile) 'строка записывается в элемент массива материалов
    limit1(i) = LineInput(MyFile) 'строка записывается в элемент массива модулей Юнга
Next i
FileClose(MyFile) 'закрываем файл
FileOpen(MyFile, My.Application.Info.DirectoryPath & foldername2 & filename, OpenMode.Input)
'открываем файл и считаем из него информацию
For i = 1 To 4
    length(i) = LineInput(MyFile) 'строка записывается в элемент массива материалов
    limit2(i) = LineInput(MyFile) 'строка записывается в элемент массива модулей Юнга
Next i
FileClose(MyFile) 'закрываем файл
For i = 1 To 4
    If ComboBox3.SelectedItem = length(i) Then
        mass1 = limit1(i)
        mass2 = limit2(i)
    End If
Next i
Form2.TextBox4.Text = mass1
Form2.TextBox5.Text = mass2
If Val(TextBox4.Text) > mass1 Then
    Form2.Label2.ForeColor = System.Drawing.Color.Red
End If
If Val(TextBox5.Text) > mass2 Then
    Form2.Label3.ForeColor = System.Drawing.Color.Red
End If
If Val(TextBox4.Text) <= mass1 And Val(TextBox5.Text) <= mass2 Then
    MsgBox("Допустимая нагрузка. Транспорт может передвигаться по данной категории дорог.",
vbInformation, "Допустимая нагрузка")
    Else : MsgBox("Нагрузка превышает допустимую! Транспорт не может передвигаться по данной
категории дорог.", vbCritical, "Нагрузка превышает допустимую")
    End If
    Form2.TextBox1.Text = Val(TextBox4.Text) + Val(TextBox5.Text)
    Me.Hide()
    Form2.Show()
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
    End
End Sub

Private Sub ClearToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles ClearToolStripMenuItem.Click
    TextBox4.Text = ""
    TextBox5.Text = ""
    TextBox4.Visible = False
    TextBox5.Visible = False
    Label4.Visible = False

```

```

Label5.Visible = False
PictureBox2.Visible = False
Button3.Visible = False
Button4.Visible = False
ComboBox1.SelectedIndex = -1
ComboBox2.SelectedIndex = -1
ComboBox3.SelectedIndex = -1
End Sub

Private Sub ExitToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles ExitToolStripMenuItem.Click
End
End Sub
Private Sub InstructionToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles InstructionToolStripMenuItem.Click
Form3.Show()
End Sub

Private Sub AboutToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles AboutToolStripMenuItem.Click
Form4.Show()
End Sub

Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button3.Click
Shell(My.Application.Info.DirectoryPath & "\sensor\pressure_sensor.exe", AppWinStyle.NormalFocus, True,
1000000)
TextBox4.Text = Clipboard.GetText
End Sub

Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button4.Click
Shell(My.Application.Info.DirectoryPath & "\sensor\pressure_sensor.exe", AppWinStyle.NormalFocus, True,
1000000)
TextBox5.Text = Clipboard.GetText
End Sub
End Class

Public Class Form2
Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
Label2.Text = Form1.Label4.Text
Label3.Text = Form1.Label5.Text
TextBox2.Text = Form1.TextBox4.Text
TextBox3.Text = Form1.TextBox5.Text
Label4.Text = Form1.Label4.Text
Label5.Text = Form1.Label5.Text
End Sub
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click, Button1.Click
Me.Close()
Form1.Show()
End Sub
End Class

Public Class Form1
Dim i As Integer
Dim dn, mU, R, h, w, wd As Double
Dim Num As Short 'кол-во материалов
Dim Materials() As String 'Массив материалов
Dim Ung() As String 'Массив модулей Юнга

```

Интерфейс датчика давления

```

Dim DopNaprr() As String 'Массив допустимых напряжений

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
    Dim MyFile As Object
    MyFile = FreeFile() 'свободный номер
    FileOpen(MyFile, My.Application.Info.DirectoryPath & "\materials.txt", OpenMode.Input) 'открываем файл
для чтения
    Num = 0 'количество строк 0
    Dim Temp As String 'временная переменная
    Dim i As Short 'счетчик
    Do While Not EOF(MyFile) 'считаем количество строк
        Temp = LineInput(MyFile) 'строка из файла считывается в переменную Temp
        Num = Num + 1 ' количество строк увеличивается на 1
    Loop
    FileClose(MyFile) 'Num = количество строк в файле
    Num = Num / 3 'делим Num на 3, получаем количество материалов в файле
    FileOpen(MyFile, My.Application.Info.DirectoryPath & "\materials.txt", OpenMode.Input) 'открываем файл
и считаем из него информацию
    ReDim Materials(Num) 'массив материалов
    ReDim Ung(Num) 'массив модулей Юнга
    ReDim DopNaprr(Num) 'массив допустимых напряжений
    For i = 1 To Num
        Materials(i) = LineInput(MyFile) 'строка записывается в элемент массива материалов
        Ung(i) = LineInput(MyFile) 'строка записывается в элемент массива модулей Юнга
        DopNaprr(i) = LineInput(MyFile) 'строка записывается в элемент массива допустимых напряжений
    Next i
    FileClose(MyFile) 'закрываем файл
    For i = 1 To Num
        ComboBox1.Items.Add(Materials(i)) 'элемент массива материалов записывается в список
    Next i

End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
    For i = 1 To Num
        If ComboBox1.SelectedItem = Materials(i) Then 'если выбранный материал совпадает с материалом из
файла
            Label5.Text = Ung(i) 'то сохраняется соответствующее материалу значение модуля Юнга
            Label6.Text = DopNaprr(i)
        End If
    Next
    R = Val(TextBox2.Text) * 0.001 'переменной присваивается значение диаметра пластины
    w = Val(TextBox1.Text) * 0.001 'переменной присваивается значение прогиба пластины
    h = Val(TextBox3.Text) * 0.001 'переменной присваивается значение толщины пластины
    mU = Val(Label5.Text) * 1000000000 'переменной присваивается значение модуля Юнга
    dn = Val(Label6.Text) * 1000000 'переменной присваивается значение допустимого напряжения
    wd = 0.256 * dn * Math.Pow(R, 2) / (mU * h)
    If Val(TextBox2.Text) < Val(TextBox3.Text) * 10 Then
        MsgBox("Диаметр пластины и толщина пластины должны соответствовать условию:  $D \geq 0,1 * h$ ",
vbCritical, "Ошибка")
    ElseIf w > wd Then
        MsgBox("Прогиб превышает допустимое значение. Допустимый прогиб " + Str(wd * 1000) + " мм",
vbCritical, "Ошибка")
    ElseIf TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Or ComboBox1.SelectedIndex = -1
Then
        MsgBox("Не все поля заполнены!", vbCritical, "Ошибка")
    Else
        Me.Hide()
        Form2.Show()
    End If
End Sub

```

```

End Sub
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
    End
End Sub
Private Sub ClearTextBoxToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ClearTextBoxToolStripMenuItem.Click
    TextBox1.Text = ""
    TextBox2.Text = ""
    TextBox3.Text = ""
    ComboBox1.SelectedIndex = -1
End Sub
Private Sub ExitToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles ExitToolStripMenuItem.Click
    End
End Sub

Private Sub AddMaterialToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles AddMaterialToolStripMenuItem.Click
    AddMaterial.Show()
    Me.Hide()
End Sub

Private Sub AboutToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles AboutToolStripMenuItem.Click
    About.Show()
End Sub

Private Sub InstructionToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles InstructionToolStripMenuItem.Click
    Instruction.Show()
End Sub
End Class

Public Class Form2
    Dim Q As Double 'Давление
    Dim M As Double 'Масса
    Dim b As Double 'прогиб пластины
    Dim L As Double 'Диаметр пластины
    Dim mU As Double 'Модуль Юнга
    Dim h As Double 'Толщина пластины

    Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
        TextBox1.Text = Form1.TextBox1.Text 'переносится значение прогиба пластины
        TextBox2.Text = Form1.TextBox2.Text 'переносится значение диаметра пластины
        TextBox3.Text = Form1.TextBox3.Text 'переносится значение толщины пластины
        TextBox4.Text = Form1.ComboBox1.SelectedItem 'переносится название выбранного материала
        L = Val(TextBox2.Text) * 0.001 'переменной присваивается значение диаметра пластины
        b = Val(TextBox1.Text) * 0.001 'переменной присваивается значение прогиба пластины
        h = Val(TextBox3.Text) * 0.001 'переменной присваивается значение толщины пластины
        mU = Val(Form1.Label5.Text) * 1000000000 'переменной присваивается значение модуля Юнга
        Q = b * mU * Math.Pow(h, 3) / Math.Pow(L, 4) 'вычисление механического давления
        M = Q / 9.80665 'вычисление массы
        TextBox5.Text = Format(Q, "0.000")
        TextBox6.Text = Format(M, "0.000")
    End Sub
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
        Clipboard.Clear()
        Clipboard.SetText(TextBox6.Text)
        Me.Close()
    End Sub
End Class

```

```

    Form1.Show()
End Sub
End Class

Public Class AddMaterial

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
        Dim MyFile1 As Object
        MyFile1 = FreeFile() 'свободный номер
        FileOpen(MyFile1, My.Application.Info.DirectoryPath & "\materials.txt", OpenMode.Append) 'открываем
файл для добавления информации
        Print(MyFile1, vbCrLf + TextBox1.Text) 'на новую строку добавляем название материала из textbox1
        Print(MyFile1, vbCrLf + TextBox2.Text) 'на новую строку добавляем модуль Юнга для материала из
textbox2
        Print(MyFile1, vbCrLf + TextBox3.Text) 'на новую строку добавляем допустимое напряжение для
материала из textbox2
        FileClose(MyFile1) 'закрываем файл
        MsgBox("Материал добавлен!", vbInformation, "Материал добавлен") 'выводим сообщение о том, что
материал добавлен
    End Sub

    Private Sub Exit3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Exit3.Click
        Dim Num As Short 'кол-во материалов
        Dim Materials() As String 'Массив материалов
        Dim Ung() As String 'Массив модулей Юнга
        Dim DopNapr() As String 'Массив допустимых напряжений
        Dim MyFile As Object
        MyFile = FreeFile() 'свободный номер
        FileOpen(MyFile, My.Application.Info.DirectoryPath & "\materials.txt", OpenMode.Input) 'открываем файл
для чтения
        Num = 0 'количество строк 0
        Dim Temp As String 'временная переменная
        Dim i As Short 'счетчик
        Do While Not EOF(MyFile) 'считаем количество строк
            Temp = LineInput(MyFile) 'строка из файла считывается в переменную Temp
            Num = Num + 1 ' количество строк увеличивается на 1
        Loop
        FileClose(MyFile) 'Num = количество строк в файле
        Num = Num / 3 'делим Num на 3, получаем количество материалов в файле
        FileOpen(MyFile, My.Application.Info.DirectoryPath & "\materials.txt", OpenMode.Input) 'открываем файл
и считаем из него информацию
        ReDim Materials(Num) 'массив материалов
        ReDim Ung(Num) 'массив модулей Юнга
        ReDim DopNapr(Num) 'массив допустимых напряжений
        For i = 1 To Num
            Materials(i) = LineInput(MyFile) 'строка записывается в элемент массива материалов
            Ung(i) = LineInput(MyFile) 'строка записывается в элемент массива модулей Юнга
            DopNapr(i) = LineInput(MyFile) 'строка записывается в элемент массива допустимых напряжений
        Next i
        FileClose(MyFile) 'закрываем файл
        Form1.ComboBox1.Items.Clear()
        For i = 1 To Num
            Form1.ComboBox1.Items.Add(Materials(i)) 'элемент массива материалов записывается в список
        Next i
        Form1.Show()
        Me.Close() 'закрываем форму
    End Sub
End Class

```