МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

**«Рязанский государственный радиотехнический университет»**

Факультет: ФВТ К защите

Направление: 11.03.03 Зав. кафедрой САПР ВС \_\_\_\_\_\_\_\_\_\_

« 15 » июня 2015 г.

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к бакалаврской выпускной квалификационной работе на тему**

«Разработка программных средств расчета надежности электронных средств»

Студент (  *Афанасьев М.Н.*  )

Руководитель бакалаврской

выпускной квалификационной работы(  *Федоров В.П.*  )

Консультант кафедры (  *Федоров В.П.*  )

« 15 » июня 2015 г.

Оглавление

[Введение 5](#_Toc422351860)

[1. Технико-экономическое обоснование темы проекта 7](#_Toc422351861)

[2. Теоретическая часть. Показатели надежности объектов и математические основы расчета систем на надежность 8](#_Toc422351862)

[2.1. Основные понятия и определения теории надежности 8](#_Toc422351863)

[2.1.1. Общие сведения 8](#_Toc422351864)

[2.1.2. Показатели безотказности 12](#_Toc422351865)

[2.1.3. Показатели безотказности и ремонтопригодности восстанавливаемых систем 17](#_Toc422351866)

[2.1.4. Законы распределения и проявления этих законов в теории надежности 20](#_Toc422351867)

[3. Обзор программных комплексов по расчету надежности ЭС 23](#_Toc422351868)

[3.1. АСОНИКА-К 23](#_Toc422351869)

[3.2. Автоматизированная система расчета надежности 25](#_Toc422351870)

[3.3. Программный комплекс АРБИТР 28](#_Toc422351871)

[3.4. Программа расчета надежности Nad.exe (программа, используемая в университете) 31](#_Toc422351872)

[4. Выбор концепции построения и описание архитектуры учебной системы расчета структурной и функциональной надежности 32](#_Toc422351873)

[4.1. Программные средства для решения поставленной задачи 33](#_Toc422351874)

[4.1.1. БД MySQL 5 33](#_Toc422351875)

[4.1.2. Язык гипертекстовой разметки HTML 34](#_Toc422351876)

[4.1.3. Язык описания сценариев PHP 39](#_Toc422351877)

[4.1.4. Web – сервер Apache 47](#_Toc422351878)

[4.1.5. Пакет «Денвер» 48](#_Toc422351879)

[4.2. Архитектура разрабатываемой системы расчета структурной надежности 49](#_Toc422351880)

[4.2.1. Разработка базы данных 51](#_Toc422351881)

[4.2.1.1. Концептуальное проектирование базы данных 51](#_Toc422351882)

[4.2.1.2. Логическое проектирование базы данных 53](#_Toc422351883)

[4.3. Разработка пользовательского интерфейса 55](#_Toc422351884)

[4.3.1. Разработка структуры пользовательского интерфейса 56](#_Toc422351885)

[4.3.2. Ввод данных 57](#_Toc422351886)

[4.3.3. Обработка введенных пользователем данных 57](#_Toc422351887)

[5. Анализ надежности и способов её повышения для типовых схем и конструкций электронных средств. 59](#_Toc422351888)

[5.1. Краткое описание типовых схем и конструкций электронных средств как объектов расчета надежности 59](#_Toc422351889)

[5.1.1. Схема блока индикации положения алюминиевого расплава в электромагнитном тигле 59](#_Toc422351890)

[5.2. Факторы, влияющие на надежность электронных средств, и учет этих факторов в моделях расчета 62](#_Toc422351891)

[5.2.1. Модель безотказной работы изделия (системы) 63](#_Toc422351892)

[5.2.2. Модели надежности при различном соединении элементов 66](#_Toc422351893)

[5.2.3. Модели надежности, учитывающие нагрузку элементов при внешних и внутренних воздействиях 69](#_Toc422351894)

[5.2.4. Методика расчета проектной надежности при основном (последовательном) соединении элементов 83](#_Toc422351895)

[5.3. Способы повышения надежности 86](#_Toc422351896)

[5.3.1. Классификация методов повышения надежности 86](#_Toc422351897)

[*5.3.2.* Способы уменьшения интенсивности отказов элементов 87](#_Toc422351898)

[5.3.3. Пути облегчения режимов работы элементов 88](#_Toc422351899)

[5.3.4. Защита от температурных воздействий 89](#_Toc422351900)

[5.3.5. Защита от механических воздействий 91](#_Toc422351901)

[5.3.6. Защита от воздействия внешних электромагнитных помех 93](#_Toc422351902)

[*5.3.7.* Рационализация схемы и конструкции 94](#_Toc422351903)

[*5.3.8.* Резервирование 96](#_Toc422351904)

[5.4. Расчет функциональной надежности 101](#_Toc422351905)

[6. Разработка электронного лабораторного практикума по расчету надежности ЭВС 104](#_Toc422351906)

[6.1. Разработка алгоритмов компьютерного расчета и моделирования надежности ЭВС 104](#_Toc422351907)

[6.1.1. Разработка скриптов 106](#_Toc422351908)

[6.2. Разработка графического интерфейса 109](#_Toc422351909)

[6.3. Проверка работы программ 113](#_Toc422351910)

[6.3.1. Объект испытаний 113](#_Toc422351911)

[6.3.2. Цель испытаний 113](#_Toc422351912)

[6.3.3. Требования к программе 113](#_Toc422351913)

[6.3.4. Технические требования 114](#_Toc422351914)

[6.3.5. Методы испытаний 114](#_Toc422351915)

[Заключение 117](#_Toc422351916)

[Библиографический список 118](#_Toc422351917)

# Введение

Повышение качества учебного процесса в высшем учебном заведении невозможно без развития и совершенствования технических средств обучения. Сюда относится, в частности, модернизация лабораторных стендов, макетов, внедрение новых программных средств, способствующих более глубокому усвоению учебного материала.

Целью данной работы является создание программных средств для обучения расчету надежности ЭВС. Необходимо разработать базу данных и программу расчета структурной надежности электронных средств при обеспечении интерфейса пользователя. База данных хранит всю необходимую информацию, а СУБД обеспечивает автоматизацию обработки этой информации.

В информационной системе должна быть отражена следующая информация:

базовое значение интенсивности отказов элемента;

На основе информации, полученной из базы данных разрабатываемая система должна рассчитывать для заданных условий эксплуатации и режимов работы элементов, в соответствии с их математическими моделями отказов, значения интенсивности отказов.

При создании программного обеспечения системы необходимо руководствоваться следующими принципами:

Система должна быть направлена на обучение

Система должна иметь удобный для пользователя интерфейс и тем самым обеспечивать простоту восприятия.

Система должна поддерживать возможность удаленного доступа

Пояснительная записка к дипломному проекту состоит из восьми частей.

Первая часть называется «Технико-экономическое обоснование темы проекта». В ней кратко охарактеризована проблематика и сущность вопроса, назначение и область применения.

Вторая часть – теоретическая, она содержит анализ задач расчета надежности.

Третья часть - обзор программных комплексов по расчету надежности ЭС. Здесь рассматриваются уже существующие аналоги моей системы расчета, их достоинства и недостатки.

Четвертая часть – «Выбор концепции построения и описание архитектуры учебной системы расчета структурной надежности". Рассматривается архитектура системы, основные алгоритмы расчета.

Пятая часть - конструкторская. Здесь приведено краткое описание схем, используемых в проекте, рассматриваются способы повышения надежности.

Шестая часть – "Разработка электронного лабораторного практикума по расчету надежности ЭВС". Производится разработка алгоритмов компьютерного расчета и моделирования надежности ЭВС, разработка графического интерфейса системы, отладка и тестирование программы.

# Технико-экономическое обоснование темы проекта

Изучение теории надежности давно и прочно вошло в учебные планы подготовки инженеров по широкому кругу специальностей. На кафедре САПР ВС для студентов, обучающихся по направлению 11.03.03 «Конструирование и технология электронных средств», читается курс "Взаимозаменяемость и надежность". По темам раздела «Надежность» в рабочей программе данной дисциплины предусмотрено проведение лабораторных работ. Одна из них – «Расчет надежности электронного блока с использованием автоматизированной справочно-информационной системы (АСРН)». Для обучения расчету надежности в рамках этой лабораторной работы необходима соответствующая программа. Определенный период времени в учебном процессе для указанных целей использовалась программа АСРН в уже устаревшей версии и нелицензированная.

Не останавливаясь на подробном объяснении причин такой ситуации, а также причин, по которым не удалась покупка более новой и лицензированной программы АСРН, следует только отметить немаловажность финансовой стороны этой проблемы.

Однако если бы даже средства на приобретение новой лицензированной версии АСРН были бы выделены, нельзя сказать, что эти затраты стали бы абсолютно оправданы. И АСРН и ее существующие продаваемые аналоги не совсем подходят для обучения, так как созданы с целью использования на предприятиях, где главным образом важен результат расчета. Нам же необходимо показать студенту сам процесс расчета, влияние каждого параметра на окончательный результат.

В связи с этим ставится задача создать необходимое приложение.

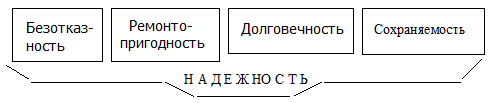
Немаловажным аспектом является и цена создаваемой информационной системы. Создание недорогой системы расчета надежности обеспечит большие возможности для её последующего внедрения. Цена разрабатываемого проекта зависит от выбора используемой СУБД. В нашем случае предпочтение будет отдаваться некоммерческим вариантам.

# Теоретическая часть. Показатели надежности объектов и математические основы расчета систем на надежность

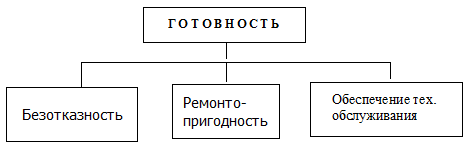
## Основные понятия и определения теории надежности

### Общие сведения

Согласно отечественному ГОСТ 27.002-89 **надежность** (Reliability) есть *способность изделия выполнять требуемые функции в заданных условиях в течение заданного периода времени*. В примечаниях к этому стандарту выделяют четыре составляющих Н свойства:



По стандарту МЭК (IEC) надежность (Dependability) – собирательный термин, применяемый для описания свойства готовности (системы, объекта) и влияющих на готовность свойств безотказности, ремонтопригодности и обеспечения тех. обслуживания/ремонта:



В этом толковании понятия Н большое внимание уделено восстановлению работоспособности.

Проблема надежности – одна из важнейших в технике. Надежность является одной из важнейших характеристик качества всякого объекта. При этом надежность нельзя смешивать с другими характеристиками качества и противопоставлять им (нельзя, например, говорить «изделие высокого качества, но низкой надежности»). Вся история техники в значительной степени может рассматриваться и как борьба за повышение надежности. Наверное, наилучшим образом это подтверждает совершенствование электронно-вычислительных и радиоэлектронных средств (ЭВС и РЭС).

В зависимости от **способа анализа объекта** разделяют два больших класса методов расчёта надёжности: **структурные и функциональные.**

Структурная надежность [1, 3] – это результирующая надежность при заданной структуре технической системы и известных значениях показателей надежности всех входящих в эту систему элементов. Для расчета структурной надежности должны быть известны схема и конструкция устройства (системы) и показатели надежности всех элементов.

Расчёт **функциональной надёжности** - это определение показателей надёжности выполнения объектом заданных функций. Поскольку такие показатели надёжности зависят от ряда действующих факторов (вида заданной функции, структурной надёжности, математического и программного обеспечения, работы операторов) то, как правило, расчёт функциональной надёжности более сложен, чем расчёт структурной надёжности.

В любом ТЗ на разработку устройства, системы всегда записываются требования по надежности. Надежность закладывается при проектировании, реализуется в производстве и поддерживается при эксплуатации изделий. То есть вопрос о будущей надежности решается на этапах жизненного цикла изделий, начиная с составления ТЗ.

Неотъемлемой частью системы обеспечения качества (в соответствии со стандартами ИСО 9001 – 9003) является система обеспечения надежности. Принципы функционирования такой системы устанавливаются стандартом ИСО 9000-4:93/ МЭК 300-1:93 “Руководство по управлению программой обеспечения общей надежности”. Центральными понятиями стандарта являются программа обеспечения общей надежности (ПОН) и план обеспечения общей надежности. Под “общей надежностью” понимается собирательный термин, используемый для определения характеристики эксплуатационной готовности и влияющих на нее факторов, а именно: характеристик надежности, ремонтопригодности и обеспечения технического обслуживания **/** ремонта. Термин используется для общих описаний без количественной оценки.

ПОН содержит перечень работ и мероприятий по обеспечению надежности, в частности:

* организационно-технические мероприятия;
* расчетно-теоретические работы;
* исследовательские, экспериментальные и испытательные работы;
* производственно-технологические работы;
* мероприятия по информационному обеспечению (сбору данных о всех процессах, связанных с созданием системы, и анализу этих данных).

Важной составной частью ПОН является анализ дефектов разрабатываемой системы.

На каждой стадии создания изделия ПОН содержит перечень работ, проводимых на следующей стадии, и предварительный перечень работ для более поздних стадий. Так ПОН на стадии “техническое предложение” содержит полный перечень работ по обеспечению Н, планируемых на стадии “Эскизный проект”.

Исследования в области надежности становятся на прочную основу только при наличии в распоряжении инженера *критериев и показателей*, позволяющих производить сравнительную количественную оценку, расчеты и испытания на надежность.

***Критерий*** – признак, мерило, на основании которого производится оценка, определение или сравнение.

***Показатель надежности*** – *количественная характеристика одного или нескольких единичных свойств, определяющих надежность объекта.* Соответственно различают единичные и комплексные показатели. Для оценки надежности среди показателей выбирают тот, который для конкретного объекта и конкретных условий его применения наилучшим образом выражает комплексное свойство надежности. Такому показателю придают функцию критерия надежности и именно его нормируют, как правило, в ТЗ и разрабатываемой на данный объект технической документации. Случаи, когда нормируется один показатель Н, а при сравнении вариантов используется другой, составляют исключения.

Необходимо отличать критерий надежности от критерия отказа и критерия предельного состояния. *Критерий отказа* – признак или совокупность признаков неработоспособного состояния объекта; эти признаки должны быть прописаны в НТД и (или) КД. Аналогично трактуется *критерий предельного состояния.*

Соответственно различным сторонам (составляющим) надежности введены показатели безотказности, долговечности, сохраняемости, достоверности. Выбор и обоснование номенклатуры показателей Н должен производиться с учетом специфики объекта, его назначения и условий эксплуатации.

Существенным фактором является *восстанавливаемость* или *невосстанавливаемость* объекта. Объект относят к группе восстанавливаемых (ВО), если его восстановление предусмотрено документацией и технически возможно на месте эксплуатации. Восстанавливаемые объекты кроме оценки надежности по единичным показателям характеризуются еще и комплексными показателями. Это, прежде всего, вероятность того, что восстанавливаемый объект окажется работоспособным в произвольный момент времени. Такой показатель учитывает свойства безотказности и восстанавливаемости.

Показатели надежности относятся к категории показателей, используемых для характеристики случайных величин и случайных событий. Уже из самого определения надежности следует, что количественно выразить свойство сохранения во времени некоторой совокупности параметров в каких-либо единицах (по примеру выражения физических свойств объекта) невозможно.

Далее рассматриваются математические и статистические определения показателей надежности, прежде всего ее важнейшей и наиболее существенной в выполняемой работе стороны - безотказности.

### Показатели безотказности

***Безотказность* –** свойство изделия сохранять работоспособность в течение некоторой наработки без вынужденных перерывов [1,3,5]. Соответственно фундаментальным в теории надежности является понятие ***отказа*** как события, заключающегося в нарушении работоспособного состояния.

В зависимости от условий и причин возникновения отказы классифицируются на постепенные и внезапные, независимые и зависимые, самоустраняющиеся и устраняемые вмешательством извне. По связи причины возникновения с этапом жизненного цикла изделия различают отказы конструкционные, производственные и эксплуатационные.

Далее показатели безотказности рассматриваются для двух групп объектов: сначала невосстанавливаемых, а затем – восстанавливаемых.

1. **Вероятность безотказной работы (ВБР)** *P(t) = P(T ≥ t),* (2.1)

где Т – время непрерывной работы объекта до первого отказа (фактически Т - время события отказа).

Соответственно **вероятность отказа** – вероятность того, что в пределах заданной наработки отказ не возникнет - определяется как

*Q(t) = 1 – P(t)*. (2.2)

Функция Q(t) – это функция распределения случайной величины Т.

Приведенные математические выражения *P(t)* и *Q(t)* отражают составленную математическую модель. Однако в реальности показатели Н чаще всего определяются из опытных данных. В таких случаях речь идет о *статистическом определении показателей Н*. Выражениям (2.1), (2.2) соответствуют статистические вероятности (частости):

 , (2.3)

где N(0) – число изделий, поставленных на испытания; n(t) – число изделий, отказавших за время t испытаний;

. (2.4)

1. **Плотность вероятности отказа** в момент времени t

. (2.5)

Соответствующая статистическая оценка получается по результатам наблюдений за работой N(0) изделий как отношение числа отказавших в единицу времени изделий к общему числу изделий при условии, что отказавшие изделия не восстанавливаются:

 . (2.6)

**Интенсивность отказов λ(t)**. Это условная плотность вероятности возникновения отказа [1,2] невосстанавливаемого объекта, определяемая для рассматриваемого момента времени при условии, что до этого момента отказ не возник. То есть λ(t) есть плотность распределения, отнесенная к вероятности безотказной работы:

. (2.7)

В статистической трактовке интенсивность отказов  есть отношение числа отказов в единицу времени на интервале Δt, примыкающему к моменту времени t. Можно также сказать, что статистическая интенсивность отказов  подсчитывается как отношение числа отказавших в единицу времени изделий к среднему числу Np работоспособных на рассматриваемом отрезке времени изделий:

 . (2.8)

При постоянстве функции плотности вероятности интенсивность отказов растет (ведь Np убывает).

Интенсивность отказов, плотность вероятности и вероятность безотказной работы связаны соотношением (2.9)

 . (2.9)

Отсюда после разделения переменных

 . (2.10)

И после интегрирования обеих частей уравнения (2.10):

 . (2.11)

где P0 – вероятность ненаступления отказа при t = 0; для изначально работоспособного устройства Р0 = 1.

Из (11) следует, что

, (2.12)

откуда

****. (2.13)

***Это соотношение является в теории Н одним из основных.***

В частном случае при λ = const

. (2.14)

К слову сказать, постоянство λ соответствует экспоненциальному убывающему характеру функции f(t).

Функцию зависимости интенсивностей отказов λ(t) от времени называют лямбда-характеристикой, кривой «жизни» элемента (рис. 2.1). Иногда ее график именуют “ваннообразной” кривой.

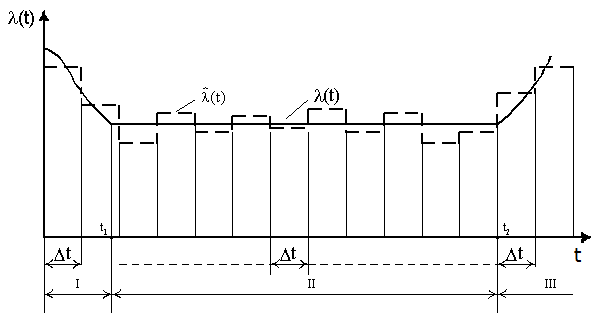


Рисунок 2.1 - Лямбда-характеристика:

- - - опытные данные; ----- линеаризованная усредненная кривая; I – интервал приработки; II – интервал нормальной эксплуатации; III – интервал старения.

Кривая имеет три явно выраженные области: I. II и III. В области I интенсивность отказов высока, что обусловлено выходом из строя изделий в первый период работы из-за наличия скрытых дефектов, не обнаруженных на этапе производства. Область II характеризуется незначительно изменяющейся (примерно постоянной) интенсивностью отказов и соответствует периоду нормальной эксплуатации. И, наконец, в области III интенсивность отказов резко возрастает из-за старения и износа и в связи с наступлением предельного состояния.

1. **Средняя наработка до отказа Т1**.

В математической трактовке этот показатель определяется как среднее взвешенное значение случайного времени наступления отказа, то есть – математическим ожиданием случайной величины с плотностью распределения f(t):

, (2.15)

а в статистической трактовке:

. (2.16)

Для невосстанавливаемых объектов статистику набирают на множестве одинаковых объектов в одинаковых условиях.

Интересно установить связь рассматриваемого показателя Т1 с интенсивностью отказов. Вернемся к математическому выражению (2.15) и сделаем некоторые преобразования:

 . (2.17)

Здесь учтено, что Q(∞) → 1, а Q(0) → 0.

Таким образом, . (2.18)

Отсюда следует, что в распространенных случаях применимости выражения (2.14)



 . (2.19)

Гамма-процентная наработка (γ-%-я наработка)

Прежде всего, зачем она введена. Дело в том, что средняя наработка Т1 до отказа соответствует весьма низкому значению ВБР, а именно

. (2.20)

С целью оценки наработки на приемлемом уровне безотказности и введено понятие γ-%-я наработки.

Гамма-процентная наработка до отказа Тγ – это наработка, в течение которой отказ объекта не возникает с вероятностью γ, выражаемой в процентах.

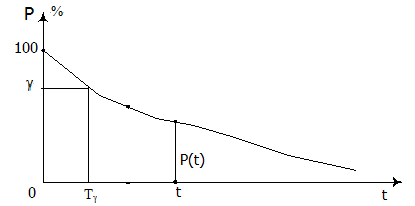


Рисунок 2.2 - К пояснению гамма-процентной наработки по кривой убыли вероятности безотказной работы Р(t)

При известном значении λ (найденном с помощью компьютерной системы расчета) при экспоненциальном законе распределения гамма-процентная наработка Тγ находится [1-3 , 5] по формуле

. (2.21)

### Показатели безотказности и ремонтопригодности восстанавливаемых систем

1. Средняя наработка на отказ – это отношение наработки восстанавливаемого объекта к математическому ожиданию числа его отказов в течение этой наработки (наработка-продолжительность или объем работы объекта).

 , (2.22)

где r(t) – число отказов в течение наработки t;

r(t2 – t1) – число отказов в течение наработки (t2 – t1).

Математическому выражению Т0 (2.22) соответствует следующая статистическая оценка показателя

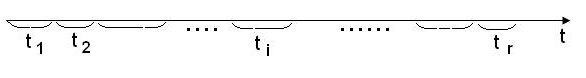
, (2.23)

где ri - число отказов (refuse по англ. – отказ) для i – го эксперимента, проводимого как с множеством восстанавливаемых объектов, так и с каждым из них.

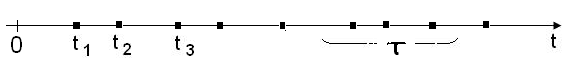
Для восстанавливаемых же объектов вводят параметр Тср “средняя наработка между отказами”. Это – математическое ожидание наработки объекта от окончания восстановления его работоспособности после отказа до возникновения следующего отказа. Статистическая оценка данного показателя:

, (2.24)

где r – число отказов за рассматриваемый период.



Простейший поток событий характеризуется стационарностью, ординарностью, отсутствием последействия.



Для простейшего потока число событий, попадающих на любой фиксированный интервал времени, распределено по закону Пуассона, а именно: вероятность Рm появления m событий в заданном интервале t определяется формулой

 . (2.25 )

Здесь А - среднее число событий, происходящих с интенсивностью λ;

А = λ· t.

В качестве показателя надежности восстанавливаемых (ремонтируемых) систем часто используют ***параметр потока отказов******ω (t).***

Если в качестве функции отказов потока использовать функцию

 , ( 2.26)

где r(t) – число отказов до наработки t; N0– число поставленных на эксплуатацию (испытания) изделий,

то  . (2.27)

Величина *ω (t)* характеризует среднее количество отказов в единицу времени, взятое для рассматриваемого момента времени.

Статистическое определение:

 . (2.28)

*Для стационарных потоков* ***ω*** *совпадает с интенсивностью отказов* ***λ****.*

ВБР ***P ( t 1 , t 2 )***. Речь идет о вероятности безотказной работы в указанном интервале времени.

. (2.29)

При стационарном потоке ω = const

. (2.30)

Вероятность восстановления. Интервал времени от момента отказа до момента восстановления является случайной величиной θ. Интегральная функция распределения этой величины выражает вероятность того, что величина θ не превысит заданного времени:

PB (t) = P (θ < t) . (2.31)

Среднее время восстановления математически выражается на основе функции (27):

**. (2.32)**

**, (2.33)**

**где tBi – время обнаружения и устранения i-го отказа объекта.**

**Коэффициент готовности**

** (2.34)**

**где Тср0 - среднее время безотказной работы; Тв – среднее время восстановления.**

**Иногда удобнее пользоваться коэффициентом простоя**

** (2.35)**

### Законы распределения и проявления этих законов в теории надежности

Отказы – случайные события; время работы до отказа, число отказов на множестве изделий, время восстановления – случайные величины.

*Для исчисления вероятностей должна быть информация*. Ведь как можно исчислять то, о чем нет никаких сведений? Такую информацию дают законы распределения случайных величин: дискретные и непрерывные.

Как известно [1-5], при для нахождения показателей надежности используют ряд известных законов распределения:

* экспоненциальное (вытекающее, в частности, из распределения Пуассона)
* нормальное
* гамма-распределение
* распределение Вейбулла и др.

Для данной работы наиболее важным является экспоненциальный закон т.к. он описывает надежность работы изделия в период его нормальной эксплуатации.

Для пуассоновского потока вероятность не появления ни одного отказа на отрезке времени τ рассчитывается по формуле (2.7) при m = 0, откуда

. (2.8)

Если начало отсчета времени t совместить с началом отрезка τ, то , а функция распределения имеет вид:

.

Соответственно путем дифференцирования Q(t) находится плотность вероятности:

. (2.9)

Ранее для рассматриваемого экспоненциального распределения мы уже выводили формулу математического ожидания времени до отказа, то есть средней наработки до отказа:

m t (= T1) = 1 **/** λ . (2.10)

Аналогичным образом выводится формуоа для дисперсии

 . (2.11)

Отсюда следует, что среднее квадратическое отклонение **σ** времени до отказа так же, как и мат. ожидание, равно 1 **/** λ. Равенство **σ**  иT1 – характерный признак экспоненциального закона распределения.

В практике расчетов часто используется приближенная формула

P(t) = 1 - λ·t , (2.12)

так как λ·t << 1.

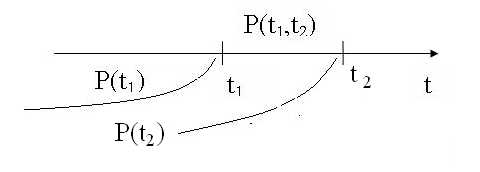
Смысл экспоненциального распределения виден из следующих рассуждений. Возьмем произвольный интервал времени (t1, t2) – рис.

Вероятность P(t1) можно интерпретировать как вероятность работы объекта до момента времени t1, а вероятность P(t1,t2) – вероятность работы на интервале времени (t1, t2). Событие безотказной работы объекта до момента времени t2 состоит в совместном появлении событий работы до момента t1 и на интервале (t1, t2). То есть

. (2.13)

Отсюда

 . (2.14)



Из (2.14) следует, что вероятность P(t1,t2) не зависит от времени t1, то есть экспоненциальный закон не учитывает предыстории.

В области старения и износа применяют нормальный закон распределения, которому соответствует монотонно нарастающая интенсивность отказов. Кривая плотности вероятности данного распределения имеет строго симметричную форму.

Промежуточными между нормальным и экспоненциальным распределениями являются гамма распределение и Вейбулла.

# Обзор программных комплексов по расчету надежности ЭС

Необходимость расчёта надёжности технических устройств и систем существовала с момента использования их человеком. Поэтому к настоящему времени существует определенное количество программ для реализации этих целей. Рассмотрим некоторые из них.

## АСОНИКА-К

При создании Базы Данных (БД) ПК АСОНИКА-К были созданы Справочная и Проектная части (СЧБД и ПЧБД), а в качестве Архивной части БД (АЧБД), в соответствии с рекомендациями стандартов в области CALS-технологий, предполагалось использовать возможности PDM (Product Data Management) системы, установленной на конкретном предприятии.

Для архивации проектов ПК АСОНИКА-К была разработана специализированная Система Архивации (АЧБД, Интерфейс пользователя и Интерфейс администратора Архива), которая позволяет реализовать следующие функции:

* управление документами и их изменениями, связанными с разработкой ЭС. Такими документами для ПК АСОНИКА-К являются проекты, хранящиеся в ПЧБД (формат "ask"), файлы отчета (формат "html") и выходные файлы ПК "ТРиАНА" - ПС моделирования тепловых процессов в конструкциях ЭС;
* контроль над занесением информации. Не контролируемое занесение в Архив данных приведет к его "замусориванию" и усложнит поиск информации в Архиве;
* контроль над извлечением информации. Содержащиеся в Архиве данные о надежности ЭС могут представлять собой коммерческую, а иногда и государственную тайну;
* разграничение прав доступа к Архиву. Доступ к информации определяется системными привилегиями пользователя;
* доступ к Архиву на сетевом уровне. Доступ к информации осуществляется напрямую с компьютера пользователя;
* резервирование Архива. Горячий резерв - использование резервного сервера, холодный резерв - копирование Архива на магнитно-оптические носители данных или другие внешние устройства.

Для практической реализации этих функций была выбрана СУБД Oracle9i. Еще одним аргументом в пользу этого выбора послужило и то, что СЧБД ПК АСОНИКА-К создана на платформе Oracle. Использование этой СУБД позволяет:

* обеспечить средствами СУБД передачу информации по глобальным и локальным сетям (Internet/Intranet) на основе Net 8;
* обеспечить средствами СУБД защиту информации при передаче по сетям (используя как внутреннюю шифровальную систему, так и подключая внешние);
* реализовать средствами СУБД систему разграничения прав пользователей;
* обеспечить средствами СУБД целостность данных;
* обеспечить средствами СУБД одновременную, полностью синхронизированную работу серверов Архива.

Для поиска информации в Архиве были созданы поисковые словари, для которых определены необходимые термины. Поиск информации в Архиве осуществляется путем выбора соответствующих критериев поиска (терминов). В разработанной СА поиск может проводиться для всех уровней разукрупнения СРН ЭС (изделия, компонентов 1-го, 2-го и 3-го уровней) по двум критериям: "Название" и "Децимальный номер".

Созданная к настоящему времени версия САР позволяет проводить следующие виды анализа:

* анализ результатов расчетов изделий (тип 1), СРН которых представляет собой произвольное соединение составных частей (СЧ) (древовидное, иерархическое, объединение СЧ в различные виды резервных групп и т.д.);
* анализ результатов расчета СЧ (тип 2), СРН которых представляет собой последовательное соединение ЭРИ и (или) СЧ.

Такое деление обусловлено тем, что, во-первых, пути повышения надежности для 1-го и 2-го типов существенно различаются:

* для 1-го типа - это изменение структуры СРН, в том числе, изменение видов и параметров резервирования, изменение ТЗ и др.
* для 2-го типа - это изменение типономинала ЭРИ (базовой интенсивности отказов), изменение схемы и конструкции ЭС (изменение электрических, тепловых, механических режимов работы ЭРИ) и др.

Система предоставляет возможность построения функции распределения времени наработки на отказ изделия, зависимости эксплуатационной интенсивности отказов компонента

С помощью ПК АСОНИКА-К можно рассчитать вероятность безотказной работы (ВБР), коэффициент оперативной готовности и коэффициент готовности, то есть те +показатели надежности, которые являются обязательными для восстанавливаемых ЭС.

Система АСОНИКА создавалась для использования на предприятиях-разработчиках электронных средств, поэтому упор был сделан на расчет параметров надежности, что не позволяет в полной мере изучить методы и особенности расчета надежности.

Из идеологии системы АСОНИКА-К в нашем случае можно использовать такие ее возможности как расчет ВБР, расчет при различных тепловых, механических режимах работы.

## Автоматизированная система расчета надежности

Данная система была разработана одним из институтов мин. обороны России и рекомендована для применения на предприятиях ОПК и на предприятиях народного хозяйства.

В настоящее время поставляются четыре варианта системы для расчета надежности:

1.АСРН-2000 - автоматизированная система расчета надежности ЭРИ и РЭА отечественного производства.

Система позволяет рассчитать эксплуатационную надежность, с учетом множества факторов (группы аппаратуры, приемки, коэффициента нагрузки, температуры, конструктивных особенностей и т.п.). Также в системе приведены справочные данные по каждому типу ЭРИ (номер ТУ, базовая интенсивность отказов по результатам испытаний и хранения, сроки хранения...).

Расчет можно производить как для одного конкретного типа ЭРИ из 37 классов, так и для сложного модуля, состоящего из многих типов.

2. АСРН-1 - автоматизированная система расчета надежности ИЭТ и РЭА народно-хозяйственного назначения.

Система позволяет рассчитать эксплуатационную надежность, с учетом множества факторов (группы аппаратуры, приемки, коэффициента нагрузки, температуры, конструктивных особенностей и т.п.). Также в системе приведены справочные данные по каждому типу ИЭТ (номер ТУ, базовая интенсивность отказов по результатам испытаний…).

Расчет можно производить как для одного конкретного типа ИЭТ из 24 классов, так и для сложного модуля, состоящего из многих типов.

3. АСН-F - автоматизированная система расчета надежности ЭРИ зарубежного производства, содержащая базу данных о надежности 15 классов ЭРИ (142 групп).

При формировании АСН-F были использованы материалы аналогичных зарубежных справочников по расчету надежности электронного оборудования.

Система позволяет рассчитать эксплуатационную надежность ЭРИ и РЭА, с учетом множества факторов (группы аппаратуры, приемки, коэффициента нагрузки, температуры, конструктивных особенностей и т.п.).

4. АСРН ЭРИ-2002. 14 редакция (справочник + автоматизированная система расчета надежности отечественных и зарубежных ЭРИ)

Автоматизированная система расчета надежности (АСРН) разработана на базе справочника "Надежность ЭРИ" и MIL-HDBK 217 F и позволяет рассчитывать надежность модулей 1-го и 2-го уровней без резервирования в режиме эксплуатации (только для отечественных ЭРИ) и хранения в составе подвижных и неподвижных объектов. Система снабжена генератором отчетов, конвертором результатов расчета в формат HTML, а также базой данных импортных ЭРИ, формируемой пользователями. В пакет поставки также входят:

* справочник "Надежность ЭРИ" последней редакции;
* программа чтения pdf - файлов Acrobat Reader 5.0;
* справочник по расчету надежности зарубежной элементной базы;
* шаблоны Word для представления результатов расчета по ЕСКД.

На рисунке 3.1 показан интерфейс программы.

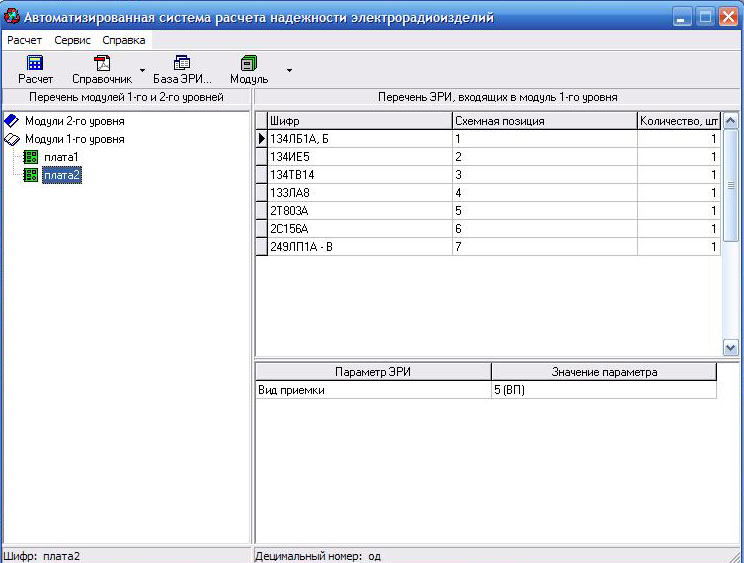


Рисунок 3.1 – Интерфейс пользователя программы АСРН.

В АСРН все расчеты проводятся автоматически, без привлечения пользователя, который, следовательно, может и не вникать в их суть*.* для нашего случая это является недостатком.

## Программный комплекс АРБИТР

АРБИТР предназначен для:

* автоматизированного моделирования и расчета показателей надежности структурно-сложных систем, включая объекты использования атомной энергии (ОИАЭ) и другие опасные производственные объекты (ОПО);
* автоматизированного моделирования и расчета вероятностей возникновения (невозникновения) аварийных ситуаций и аварий ОПО, включая ОИАЭ.

В настоящее время комплекс АРБИТР реализует следующие функции, прошедшие процедуру аттестации:

* представление в исходной СФЦ (в суперграфе СФЦ) до 400 элементов (вершин) и до 100 элементов в каждой декомпозированной вершине (подграфах СФЦ) основного графа исследуемой системы (т.е. возможность ввода до 40 000 вершин);
* автоматическое построение логических функций, представляющих кратчайшие пути успешного функционирования (КПУФ), минимальные сечения отказов (МСО) или их немонотонные комбинации (явные детерминированные модели исследуемых свойств системы);
* автоматическое построение вероятностных функций, обеспечивающих точный расчет показателей устойчивости, эффективности и риска исследуемых систем;
* расчет вероятности реализации заданных критериев, представляющих свойства устойчивости (надежности, стойкости, живучести) и безопасности (технического риска, вероятностей возникновения аварийных ситуаций и аварий) систем;
* расчет вероятности безотказной работы или отказа и средней наработки до отказа невосстанавливаемых систем;
* расчет коэффициента готовности, средней наработки на отказ, среднего времени восстановления и вероятности безотказной работы восстанавливаемых систем;
* расчет вероятности готовности смешанных систем, состоящих из восстанавливаемых и невосстанавливаемых элементов;
* расчет значимостей, положительных и отрицательных вкладов всех элементов исследуемой системы в вероятность реализации исследуемого свойства, используемые для выработки и обоснования управленческих решений по обеспечению устойчивости, живучести, безопасности эффективности и риска функционирования;
* вспомогательный режим приближенных расчетов, которые выполняются по двум методикам: для независимых отказов элементов (аналог методики, используемой в комплексах "Risk Spectrum" (Швеция) и "Saphire-7" (США)), и с учетом трех типов отказов элементов – "отказ на требование", "отказ в режиме работы" и "скрытый отказ в режиме ожидания" (методы разработаны специалистами ФГУП ОКБМ им И.И.Африкантова и впервые реализованы в аттестованном комплексе "CRISS 4.0");
* расчет вероятности реализации отдельных КПУФ или МСО системы;
* расчет значимости и суммарной значимости сечений отказов по Fussell-Vesely;
* расчет значимости, уменьшения и увеличения риска элементов по Fussell-Vesely;
* приближенный расчет вероятностных характеристик системы с учетом трех типов отказов элементов: отказ на требование, отказ в режиме работы и скрытый отказ в режиме ожидания (по методике, реализованной в ПК CRISS 4.0);
* структурный и автоматический учет отказов групп элементов по общей причине (модели альфа-фактора, бета-фактора и множественных греческих букв);
* учет различных видов зависимостей и множественных состояний элементов, представляемых c помощью групп несовместных событий;
* учет двухуровневой декомпозиции структурной схемы, дизъюнктивных и конъюнктивных кратностей сложных элементов (подсистем);
* учет неограниченного числа циклических (мостиковых) связей между элементами и подсистемами;
* учет различных комбинаторных отношений (К из N) между группами элементов.

Для моделирования и расчета показателей надежности и безопасности системы пользователю необходимо подготовить исходные данные:

* Изучить объект, разработать событийно-логическую структурную модель исследуемого свойства системы (например, в виде блок-схемы, графа связности, дерева отказов, дерева событий и др.), представит ее в форе СФЦ (одной или нескольких) и ввести в комплекс АРБИТР.
* Определить и ввести в АРБИТР вероятностные и другие параметры элементов исследуемой системы.

Недостатком этой системы является не совсем удобный интерфейс, рассчитанный по большей части на профессиональных разработчиков

## **Программа расчета надежности Nad.exe** (программа, используемая в университете)

Обучающая программа с возможностью вывода принципиальной схемы. Здесь электронный блок рассматривается как невосстанавливаемый в процессе работы объект. Программа предполагает ручную работу со справочной системой по расчёту надежности

В этой системе отсутствует возможность смены условий эксплуатации, также не учитывается интенсивность отказов соединений, разъемов и прочее.

# Выбор концепции построения и описание архитектуры учебной системы расчета структурной и функциональной надежности

Перед нами стоит задача создать такую систему, благодаря которой можно будет существенно упростить процесс обучения расчету надежности. Одной из особенностей этой системы является ее направленность на дистанционное обучение.

Вследствие вышеперечисленного, работа с системой будет проходить в браузере, где необходимая информация будет отображаться на web-страницах. Данные для расчета будут поступать в систему из базы данных, также в системе организован ввод данных пользователем.

Основные идеи современной информационной технологии базируются на концепциибаз данных (БД)[10]. База данных (БД) — структурированный организованный набор данных, описывающих характеристики каких-либо физических или виртуальных систем. Согласно данной концепции основой информационной технологии являютсяданные,организованные в БД, адекватно отражающие реалии действительности в той или иной предметной области и обеспечивающие пользователя актуальной информацией в соответствующей предметной области. Для создания и управления информационной системой необходима система управления базами данных (СУБД). СУБД - специализированная программа (чаще комплекс программ), предназначенная для организации и ведения базы данных.

Обычно современная СУБД содержит следующие компоненты:

* ядро, которое отвечает за управление данными во внешней и оперативной памяти и [журнализацию](http://ru.wikipedia.org/wiki/%D0%96%D1%83%D1%80%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D0%B8%D0%B7%D0%BC%D0%B5%D0%BD%D0%B5%D0%BD%D0%B8%D0%B9);
* процессор языка базы данных, обеспечивающий [оптимизацию запросов](http://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D1%82%D0%B8%D0%BC%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D0%B7%D0%B0%D0%BF%D1%80%D0%BE%D1%81%D0%BE%D0%B2_%D0%A1%D0%A3%D0%91%D0%94) на извлечение и изменение данных, и создание, как правило, машинно-независимого исполняемого внутреннего кода,
* подсистему поддержки времени исполнения, которая интерпретирует программы манипуляции данными, создающие пользовательский интерфейс с СУБД
* сервисные программы, обеспечивающие ряд дополнительных возможностей по обслуживанию информационной системы.

Хорошая система управления базами данных относительно справляется с ролью эффективного посредника между пользователем и данными, организовывает, каталогизирует, резервирует и выполняет целый ряд других операций, которые направлены на то, чтобы ускорить и облегчить работу с данными.

## Программные средства для решения поставленной задачи

### БД MySQL 5

Решающим критерием является выбор СУБД. Среди некоммерческих вариантов наиболее подходящим является MySQL, так как она считается самой совершенной СУБД, распространяемой на условиях открытых исходных текстов. В MySQL реализованы многие возможности, традиционно встречавшиеся только в масштабных коммерческих продуктах. По сравнению с другими MySQL предоставляет ряд преимуществ:

* это одна из самых популярных систем управления базами данных, используемых в настоящее время в web-среде;
* она свободно доступна для загрузки из Internet и установки на практически любой машине;
* ее легко установить на многих операционных системах (включая Windows и Unix);
* она проста в использовании и включает в себя несколько удобных инструментов администрирования;
* это быстрая, мощная клиент/серверная система, которая справляется с очень крупными, сложными базами данных и хорошо проявляет себя, когда дело касается крупных проектов.

MySQL — свободная система управления базами данных. MySQL является собственностью компании Sun Microsystems, осуществляющей разработку и поддержку приложения. MySQL является решением для малых и средних приложений. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы. [16]

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей.

InnoDB — одна из выбираемых подсистем низкого уровня в СУБД MySQL, входит во все стандартные сборки для различных операционных систем. Основным отличием InnoDB от других подсистем низкого уровня MySQL является наличие механизма транзакций. В отличие от таблиц MyISAM, где для каждой таблицы создается один файл данных, данные InnoDB в настройках по умолчанию хранятся в больших совместно используемых файлах (изменить это можно с помощью настроек опции innodb\_file\_per\_table), что позволяет использовать постраничный кэш страниц базы данных. Формат данных InnoDB обеспечивает надежное хранение данных за счет транзакционности.

### Язык гипертекстовой разметки HTML

Язык разметки гипертекстовых страниц (HTML – Hypertext Markup Language) представляет собой язык, разработанный специально для создания Web-документов.[11] Он определяет синтаксис и размещение специальных инструкций (тегов), которые не выводятся на экран, но указывают браузеру, как отображать содержимое документа. Он предназначен для создания форматированного текста, насыщенного изображениями, звуком, анимацией, видеоклипами и гипертекстовыми ссылками на другие документы, разбросанные как по всему Web-пространству, так и находящиеся на этом же сервере или являющиеся составной частью этого же Web-проекта.

На практике на стандарт HTML большое влияние оказывает наличие тегов, предложенных и поддерживаемых наиболее известными браузерами, такими как Microsoft Internet Explorer и Opera. Эти теги в данный момент могут как входить, так и не входить в состав действующей спецификации HTML.

#### Создание документов в формате HTML. Принципы гипертекстовой разметки. Структура документов

За основу модели разметки документов в HTML принята тэговая модель. Тэговая модель описывает документ как совокупность контейнеров, каждый из которых начинается и заканчивается тэгами. Т.е. документ НТМL представляет собой не что иное, как обычный АSСII-файл, с добавленными в него управляющими НТМL-кодами (тэгами).

Тэги НТМL-документов в большинстве своем просты для понимания и использования, ибо они образованы с помощью общеупотребительных слов английского языка, понятных сокращений и обозначений. НТМL-тэг состоит из имени, за которым может следовать необязательный список атрибутов тэга. Текст тэга заключается в угловые скобки (< и >). Простейший вариант тэга - имя, заключенное в угловые скобки, например <HEAD> или <i>. Для более сложных тэгов характерно различие атрибутов, которые могут иметь конкретные значения, определенные автором для видоизменения функции тэга.

В некоторых случаях конечные тэги в документе можно опускать. Большинство браузеров реализованы так, что при обработке текста документа начальный тэг воспринимается как конечный тэг предыдущего. Самый распространенный тэг такого типа - тэг абзаца <Р>. Поскольку он используется в документе очень часто, то его обычно ставят только в начале каждого абзаца. Когда один абзац заканчивается, следующий тэг <Р> сигнализирует браузеру о том, что нужно завершить данный абзац и начать следующий.

Атрибуты тэга следуют за именем и отделяются друг от друга одним или несколькими знаками табуляции, пробелами или символами возврата к началу строки. Порядок записи атрибутов в тэге значения не имеет.

#### Группы тэгов НТМL

Все тэги НТМL по их назначению и области действия можно разделить на следующие основные группы

* определяющие структуру документа;
* оформление блоков гипертекста (параграфы, списки, таблицы, картинки);
* гипертекстовые ссылки и закладки;
* формы для организации диалога;
* вызов программ.

Структура гипертекстовой сети задается гипертекстовыми ссылками. Гипертекстовая ссылка - это адрес другого HTML документа или информационного ресурса Internet, который тематически, логически или каким-либо другим способом связан с документом, в котором ссылка определена.

Этот текст содержит гипертекстовую ссылку

<A HREF="http://polyn.net.kiae.su/altai/index.html"></A>

В приведенном выше примере тэг "A", который в HTML называют якорем (anchor), использует атрибут "HREF", который обозначает гипертекстовую ссылку (Hypertext Reference), для записи этой ссылки в форме URL. Данная ссылка указывает на документ с именем "index.html" в директории "altai" на сервере "polyn.net.kiae.su", доступ к которому осуществляется по протоколу "http".

#### Контейнеры HTML-документа

Каждая из составных частей документа имеет свой набор контейнеров, которые можно внутри нее использовать. Контейнеры тела документа не используются в заголовке или в контейнере FRAMSET.

Заголовок документа не имеет атрибутов. Основное назначение тэгов заголовка - это описание общих для всего документа параметров отображения.

Тэг МЕТА предназначен для определения в заголовке документа конструкций, отсутствующих в спецификации НТМL.

META HTTP-EQUIV="Keywords"

CONTENT="Plasma, Nuclear Physics">

При таком использовании в заголовок НТТР-пакета будет включена строка: Keywords: Plasma, Nuclear Physics, что удобно при отправке почты, например.

#### Средства описания таблиц в HTML

Для описания таблиц служит тэг <ТАВLЕ>. Тэг <ТАВLЕ>, как и многие другие, автоматически переводит строку до и после таблицы.

Тэг <ТR> (сокращение от Таble Row - строка таблицы) создает строку таблицы.

Внутри строки таблицы обычно размещаются ячейки с данными. Каждая ячейка, содержащая текст или изображение, должна быть окружена тэгами <ТD></ТD>. Число тэгов <ТD></ТD> в строке определяет число ячеек.

При задании заголовков для столбцов и строк таблицы используются тэг заголовка <ТН></ТН> (Таblе Неаder, заголовок таблицы).

Тэг <CAPTION> позволяет создавать заголовки таблицы. По умолчанию заголовки центрируются и размещаются либо над (<САРТION АLIGN=ТОР>), либо под таблицей (<САРТION ALIGN=ВОТТОМ>). Заголовок может состоять из любого текста и изображений.

Обычно любой текст в таблице, не помещающийся в одну строку ячейки, переходит на следующую строку. При использовании атрибута NOWARP с тэгами <ТН> или <ТD> длина ячейки расширяется на столько, чтобы, заключенный в ней текст поместился в одну строку.

Тэги <ТD> и <ТН> модифицируются с помощью атрибута СОLSPAN= (Column Span, соединение столбцов).

Атрибут ROWSPAN=, используемый в тэгах <ТD> и <ТН>, аналогичен атрибуту СОLSPAN=, только он задает число строк, на которые растягивается ячейка.

Атрибут WIDТН= применяется в двух случаях. Можно поместить его в тэг <ТАВLЕ> для задания ширины всей таблицы, а можно использовать в тэгах <ТR> или <ТН> для задания ширины ячейки или группы ячеек. Ширину можно указывать в пикселях или в процентах.

Атрибут UNIT= тэга <ТАВLЕ> определяет единицы измерения, используемые при указании размеров, как всей таблицы, так и ее отдельных столбцов.

Атрибут СОLSРЕС=, используемый с атрибутом UNIТ=, определяет, сколько места занимает каждый столбец таблицы, и как в нем выравниваются данные. Применяется только в тэге<ТАВLЕ>.

Атрибут DР= (Decimal Point, десятичный знак) определяет символ, используемый для отделения целой части десятичной дроби.

Атрибут СЕLLPADDING= определяет ширину пустого пространства между содержимым ячейки и ее границами, то есть задает поля внутри ячейки.

Тэги <ТR>, <ТD> и <ТН> можно модифицировать с помощью атрибутов ALIGN= и VALIGN=. Атрибут АLIGN определяет выравнивание текста и графики по горизонтали, то есть по левому или правому краю, либо по центру,

Атрибут VALIGN= осуществляет выравнивание текста и графики внутри ячейки по вертикали. Вертикальное выравнивание может быть задано несколькими способами: VALIGN=ТОР. Выравнивает содержимое ячейки по ее верхней границе.

VALIGN=МIDDLE центрирует содержимое ячейки по вертикали.   
VALIGN=ВОТТОМ выравнивает содержимое ячейки по ее нижней границе.

Атрибут СЕLLSPACING= определяет в пикселях ширину промежутков между ячейками.

### Язык описания сценариев PHP

#### Общие сведения о PHP

PHP (Personal Home Page Tools) - это язык описания сценариев общего назначения, который создан специально для Web и который можно внедрять в HTML. Важным отличием от других языков, используемых для Web-программирования, является то, что вместо написания программы с большим количеством команд для вывода HTML, пишется HTML-скрипт с некоторым количеством встроенного кода для выполнения каких-либо действий. Код PHP заключён в специальные начальный и конечный тэги, что позволяет входить и выходить из "режима PHP".

PHP отличается от других подобных языков, типа клиентского JavaScript, тем, что код выполняется на сервере. То есть сам скрипт находится на сервере, а клиент получает только результат работы этого скрипта, не имея возможности определить, каков был исходный код. Наилучшим качеством PHP является то, что он довольно прост, но предлагает много разнообразных возможностей для программиста.

PHP в основном сориентирован на серверный скриптинг, поэтому может делать всё то, что делают CGI-программы: сбор данных форм, динамическую генерацию содержимого страницы или приём и отправку cookies. Но PHP может намного больше.

PHP может использоваться на всех крупных операционных системах (ОС), включая Linux, многие варианты Unix (HP-UX, Solaris и OpenBSD), Microsoft Windows, Mac OS X, RISC OS и, возможно, другие. PHP имеет поддержку для большинства существующих web-серверов. Это Apache, Microsoft Internet Information Server, Personal Web Server, Netscape и iPlanet-серверы, Oreilly Website Pro, Caudium, Xitami, OmniHTTPd и многие другие. Для большинства этих серверов PHP имеет модули. В других, поддерживающих стандарт CGI, PHP может работать как CGI-процессор.

Итак, PHP дает свободу выбора ОС и web-сервера. В PHP нет ограничений в выводе HTML. PHP может выводить изображения, PDF-файлы и даже клипы Flash, генерируемые на лету. PHP имеет предельно удобные возможности для работы с текстом, он может выводить любой текст, включая XHTML, и любой другой XML-файл. PHP автоматически генерирует эти файлы и сохраняет их в файловой системе, вместо их распечатки, формируя серверный кэш для динамического содержимого.

Одна из наиболее сильных и привлекательных черт PHP - поддержка им большого количества баз данных (БД). Имеется также абстрактное расширение DBX, позволяющее прозрачно использовать любую БД, поддерживаемую этим расширением. Также PHP поддерживает ODBC, стандарт Open Database Connection, и дает возможность подключаться к любой БД, поддерживающей этот всемирный стандарт.

#### Принцип работы PHP

Когда PHP разбирает файл, он просто передаёт текст файла, пока не обнаружит один из специальных тэгов, который говорит о необходимости начать интерпретацию текста как кода PHP.[15] Разборщик выполняет весь найденный код до закрывающего тэга PHP, который говорит разборщику, что нужно снова начать просто передавать текст. Этот механизм позволяет внедрять PHP-код в HTML: всё за пределами тэгов PHP остаётся без изменений, а внутри тэгов - разбирается как код.

Имеются четыре набора тэгов, которые используются для обозначения блоков кода PHP. Только два из них (<?php. . .?> и <script language="php">. . .</script>) всегда доступны; другие можно включать и отключать из файла конфигурации php.ini. Хотя сокращённые тэги и тэги в стиле ASP могут быть удобны, они не так переносимы, как их длинные версии.

#### Синтаксис языка PHP

PHP является настоящим языком программирования со всеми присущими ему атрибутами: переменными, операторами контроля процесса, командами ввода/вывода и др.

Для написания кода программ на PHP используется стандартный набор латинских символов. В строковых выражениях могут использоваться символы национальных алфавитов. В качестве комментария используется последовательность символов //.

Переменные – это контейнеры, в которых содержатся данные. Для создания переменной нужно дать ей имя и значение. Чтобы PHP мог отличать переменные от обычного текста или команды, их имя должно начинаться со знака «$». Имя переменной чувствительно к регистру символов. Также существуют определенные ограничения на имя переменной: оно не должно содержать пробелов, не должно содержать знаков апострофов и некоторых других символов. Как и везде, чтобы создать переменную, нужно написать ее имя, а затем, через знак присваивания значение:

$a = 125;

Таким образом, один раз объявив переменную в дальнейшем на нее можно просто ссылаться.

PHP поддерживает 8 примитивных типов.

4 скалярных типа:

[boolean](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/types.boolean.html)

[integer](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/types.integer.html)

[число с плавающей точкой (float)](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/types.float.html)

[string](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/types.string.html)

Два составных типа:

[array](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/types.array.html)

[object](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/types.object.html)

И, наконец, два специальных типа:

[resource](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/types.resource.html)

[NULL](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/types.null.html)

Тип переменной обычно программистом не устанавливается; напротив, он определяется РНР на этапе прогона, в зависимости от контекста, в котором эта переменная используется.[12] В случаях, когда необходимо проверить тип и значение конкретного [выражения](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/expressions.html), используется функция [*var\_dump()*](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/f/var-dump.html). Если нужно получить читабельное представление типа для отладки, используется [*gettype()*](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/f/gettype.html). Но следует заметить, что для проверки конкретного типа функция [*gettype()*](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/f/gettype.html) не используется, а используется функция *is\_type.*

Для того чтобы форсированно конвертировать переменную в определённый тип, можно либо [привести](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/types.type-juggling.html#language.types.typecasting) переменную, либо использовать с ней функцию [*settype*](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/f/settype.html)*.* Приведение типов в PHP работает так же, как и в C: имя нужного типа записывается в скобках перед переменной, которая к этому новому типу приводится.

Допускаются следующие приведения типов:

(int), (integer) - приведение к integer

(bool), (boolean) - приведение к boolean

(float), (double), (real) - приведение к float

(string) - приведение к string

(array) - приведение к array

(object) - приведение к object

Изменение значений переменных происходит при помощи использования оператора присваивания "=". Кроме оператора присваивания используются следующие основные логические операции:

«= =» - знак равенства;

«>», «<», «>=», «<=» - больше, меньше, больше или равно, меньше или равно;

«!=» или «<>» - знак неравенства;

а и b – условия;

а && b – действие будет выполнено, если выполняется И условие а, И условие b

а II b - действие будет выполнено, если выполняется ИЛИ условие а, ИЛИ условие b, ИЛИ оба условия одновременно.

Никакой язык программирования не может обойтись без математических операций. В PHP также встроены операторы, которые позволяют производить математические вычисления. Основные математические операции:

a + b – сложение;

a – b – вычитание;

a \* b – умножение;

a / b – деление;

a % b – остаток от деления а на b;

$переменная++; - запись эквивалентна следующей: $переменная =

$переменная+1; - значение переменной увеличивается на 1;

$переменная--; - значение переменной уменьшается на 1.

Массивы – это набор данных, объединенных под одним именем. Храниться он может в переменной. Одними из самых легких способов создания массива являются следующие:

$имя массива [номер элемента] = значение

$имя массива = array (значение);

Индексы и значения в массиве разделяются оператором =>. Пары index=>value разделяются запятыми, они определяют индекс и значение. Индекс может быть как числовым, так и строковым. В ассоциированных массивах индекс всегда ведет себя как строковой. В случае, если индекс не указан, будет подставляться автоинкремент (на 1 больше), начиная с 0. Если при создании массива были указаны два элемента с одинаковыми индексами, то последний элемент заменяет первый. Для просмотра элементов массива существуют две функции: var\_dump() и print\_r().

#### Управление потоком вычислений в PHP

PHP имеет следующие операторы, реализующие основные алгоритмы управления потоком вычислений (flow control): оператор ветвления, оператор цикла с конечным числом повторений, оператор цикла while.

Рассмотрим их синтаксис.

Оператор ветвления реализует выбор той или иной последовательности действий в зависимости от условия (условный оператор):

If (условие) {…команды, которые должны выполняться, если условие верно…;} еlse {… команды, которые должны выполняться, если условие неверно…;}

Выражение "условие" должно быть булевского типа. Если "условие" вычисляется в TRUE, PHP выполнит команды, а если вычисляется в FALSE - оператор игнорируется.

Также в PHP в операторе ветвления используется комбинация if и else - elseif. Подобно else, она расширяет оператор if для выполнения других операторов в том случае, если оригинальное выражение if вычисляется в FALSE. Однако, в отличие от else, elseif будет выполнять альтернативное выражение только тогда, когда условное выражение в elseif будет вычислено в TRUE.

Может быть несколько elseif внутри одного оператора if. Первое выражение elseif (если имеется), которое вычисляется в TRUE, будет выполняться. В PHP вы можете также записать 'else if' (двумя словами), и поведение будет идентично 'elseif' (в одно слово). Оператор elseif выполняется только в том случае, если предшествующее выражение if и любое предшествующее выражение elseif вычислены в FALSE, а текущее выражение elseif вычислено в TRUE.

PHP предлагает альтернативный синтаксис для некоторых структур управления: if, while, for, foreach и switch. В каждом случае базовая форма синтаксиса изменяется - открывающая фигурная скобка заменяется на двоеточие, а закрывающая - на endif;, endwhile;, endfor;, endforeach; или endswitch;, соответственно.

Оператор цикла с конечным числом повторений повторяет определенную последовательность действий заданное число раз.

Синтаксис цикла for таков:

For (выражение 1, выражение 2, выражение 3) {…команды…;}, где:

выражение 1 = начальное условие счетчика;

выражение 2 = условие продолжения цикла;

выражение 3 = изменение счетчика на каждом цикле

Первое выражение вычисляется (выполняется) один раз и безусловно в начале выполнения цикла.

При начале каждой итерации вычисляется выражение 2. Если оно вычисляется в TRUE, цикл продолжается и выполняется вложенный оператор. Если оно вычисляется в FALSE, выполнение цикла прекращается.

В конце каждой итерации вычисляется (выполняется) выражение 3.

Каждое из выражений может быть пустым. Пустое выражение 2 означает, что цикл должен выполняться бесконечно (PHP неявно предполагает, что это условие TRUE, как в C). Это может быть не так бессмысленно, как может показаться, поскольку часто бывает необходимо закончить цикл оператором [break](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/break.html) вместо использования проверки выражения-условия цикла for. Оператор break заканчивает выполнение текущей структуры (цикла) for, foreach, while, do..while или switch. Он принимает необязательный числовой аргумент, указывающий, сколько содержащих вкладывающих структур прерывают выполнение.

Циклы while это простейшие циклы PHP. Они ведут себя, как их аналоги из C. Вот базовая форма оператора while:

while (условие) {…команды…;}

Значение оператора while говорит PHP, что нужно неоднократно выполнять вложенный оператор, пока «условие» вычисляется в TRUE. Значение выражения проверяется каждый раз в начале цикла, поэтому, если это значение изменилось при выполнении вложенного оператора, выполнение не остановится до конца данной итерации (каждый раз, когда PHP выполняет все операторы цикла, называется одной итерацией цикла). Иногда, если «условие» вычисляется в FALSE в самом начале цикла, вложенный оператор может быть не выполнен ни разу.

Как и в операторе if, можно создавать блок из операторов внутри цикла while с помощью фигурных скобок {} или используя альтернативный синтаксис:

while (условие) {…команды, которые должны выполняться, если условие верно…;}... endwhile;

#### Основные команды, используемые в PHP

В PHP, как и в любом другом языке программирования есть команды, с помощью которых можно выполнять определенные действия:

**еcho()** - одна из наиболее часто используемых команд, выводящая на web-страницу какой-либо текст, заключенный в кавычки.

**include()** «имя файла» - команда, вставляющая содержимое одного файла в другой. Файл именно вставляется, то есть пути к картинкам уже нужно указывать от того файла, где вставляется команда include

**return() -** если вызывается из функции, оператор return(**)** немедленно заканчивает выполнение текущей функции и возвращает свой аргумент как значение вызова функции. Оператор return() также заканчивает выполнение оператора [eval()](mk:@MSITStore:C:\Documents%20and%20Settings\Александр\Рабочий%20стол\ДИПЛОМ\PHP\PHP42\php42.chm::/f/eval.html) или файла скрипта. Если вызывается из глобальной области видимости, выполнение файла текущего скрипта заканчивается.

**require() -** ператор require() подключает/include и вычисляет определённый файл.Операторыrequire() и include() идентичны почти во всём, за исключением того, как они обрабатывают неудачное выполнение. include() выдаёт Warning!, а require() выдаёт Fatal Error.

### Web – сервер Apache

Современная web-разработка во многих случаях требует использования динамических решений, написания скриптов на различных языках программирования, навыков работы с базами данных. В этом творческом процессе возникает необходимость отладки созданных программ на соответствующем тестовом сервере. Кроме самого сервера, нам еще понадобятся интерпретаторы необходимых языков программирования, а также СУБД.

На сегодняшний день самым популярным web-сервером является Apache. Этот продукт распространяется свободно, а благодаря открытому коду пользуется благосклонностью разработчиков дополнительных модулей. Поддержкой и координацией развития Apache занимается группа Apache Server Foundation, а сам сервер признается одним из самых крупных достижений разработчиков свободного ПО.

Совместно с Apache, как правило, используются языки программирования РНР и Perl, хотя могут быть и другие варианты - это зависит от поставленных задач.[13] Это также свободные продукты с большим числом поклонников, на деле показавшие свою пригодность для применения в web-проектах самого различного плана и уровня.

Что касается СУБД, то достаточно часто для решения подобных задач используют пакет MySQL. Основные его достоинства - это достаточная функциональность и быстродействие в совокупности с простотой разработки приложений с ее использованием. Существуют и намного более мощные системы, в том числе и свободные, однако MySQL непрерывно развивается и не собирается сдавать свои позиции. Если говорить про условия приобретения, то пакет MySQL бесплатен для некоммерческого применения.

Поскольку, как мы уже отметили, разработка может вестись и на домашней машине, это в подавляющем большинстве случаев означает, что иметь дело надо будет с операционными системами от Microsoft. Названные пакеты могут работать на многих платформах, среди которых Unix, Linux, BSD, Solaris и даже MacOS. Для нас важен тот факт, что есть версии всех необходимых нам пакетов, совместимые с ОС Windows.

Более того, есть решение, позволяющее избежать процедуры ручной установки и настройки их по отдельности. Это так называемые "комплексы", которые включают Windows-версии Apache, MySQL и реализации некоторых популярных языков программирования. Данные компоненты заранее отконфигурированы, а ведь именно на этом этапе делается наибольшее количество ошибок людьми, не работавшими ранее с подобным программным обеспечением.

Подобные пакеты можно, в принципе, использовать не только по прямому назначению, то есть для тестирования собственноручно написанных скриптов для своих web-страниц, но и в роли полноценного сервера в рамках, например, локальной сети. [9]

### Пакет «Денвер»

Этот пакет разработан российскими программистами. [14] Название его расшифровывается следующим образом: "ДжЕнтльменский Набор ВЕб-Разработчика". Он отличается завидной для своих возможностей компактностью (размер дистрибутива всего около двух мегабайт) и наличием поддержки как PHP, так и Perl. Конфигурация "Денвера" текущей версии такова:

* Apache 2.2.4 + SSL;
* PHP 5.2.12 с поддержкой GD, MySQL, SQLite;
* MySQL 5.0.51a;
* PhpMyAdmin 3.2.3;
* Ядро Perl без стандартных библиотек.

Кроме названных компонентов, "Денвер", как и положено такой системе, обладает единым инсталлятором, а также рядом оригинальных дополнений, например, основанной на шаблонах системой управления виртуальными хостами Apache. Совсем не лишним представляется и переведенные на русский язык комментарии в конфигурационных файлах.

Как говорится в сопроводительной документации, "Предлагаемый комплекс программ позволяет запустить полноценный веб-сервер Apache с поддержкой PHP, Perl и сервер MySQL на машине, работающей под управлением MS Windows 95/98/Me/NT/2000/XP. Как правило, при установке не требуется никакой настройки, комплекс может использоваться даже неподготовленными пользователями. Тем не менее, он может найти применение не только для обучения азам веб-программирования, но и для отладки интерактивных сайтов, программ, использующих базу данных MySQL, написанных на языках PHP и Perl” - трудно добавить к этому что-либо от себя, разработчики выразились более чем доступно.

Существует два варианта запуска "Денвера". Первый предусматривает создание виртуального диска при загрузке Windows. Старт серверов осуществляется вручную. Этот вариант считается предпочтительным, так как дает некоторые дополнительные возможности при работе с командной строкой и установленными программами без запуска самого Apache. Второй подразумевает создание виртуального диска непосредственно при ручном запуске серверов. В рамках данного дипломного проекта был выбран первый вариант запуска.

## Архитектура разрабатываемой системы расчета структурной надежности

В данной системе расчета структурной надежности имеется база данных, в которой находятся все необходимые компоненты для используемых в лабораторной работе схем. В БД также имеются базовые значения интенсивности отказов для каждого элемента.

Исходными данными для расчета надежности служат:

* схема устройства с перечнем входящих в нее элементов, интенсивности отказов которых известны;
* параметры внешних воздействий, условия эксплуатации;
* время работы и хранения устройства (блока, платы);
* электрическая нагруженность элементов, учитываемая коэффициентами нагрузки по напряжению и мощности.

Рассмотрим основополагающие факторы, влияющие на выбор архитектуры информационной системы.

Архитектура информационной системы будет зависеть от типа базы данных, то есть от выбора СУБД. Так как информационная система предназначена для использования в лабораторном практикуме, проводимом в течение одного семестра для студентов двух-трех групп, то количество её пользователей, вероятно, будет не велико.

При построении архитектуры информационной системы нужно учесть ее основную особенность, а именно web – направленность. Web-приложения представляют собой особый тип программ, построенных по архитектуре "клиент-сервер". Особенность их заключается в том, что само Web-приложение находится и выполняется на сервере - клиент при этом получает только результаты работы. Работа приложения основывается на получении запросов от пользователя (клиента), их обработке и выдачи результата. За счет наличия исполняемой части, Web-приложения способны выполнять практически те же операции, что и обычные Windows-приложения, с тем лишь ограничением, что код исполняется на сервере, в качестве интерфейса системы выступает браузер, а в качестве среды, посредством которой происходит обмен данными, - сеть (локальная или глобальная). Таким образом, информационная система состоит из 2 частей:

* Веб-сервер — это сервер, принимающий HTTP-запросы от клиентов, обычно веб-браузеров, и выдающий им HTTP-ответы, обычно вместе с HTML-страницей, изображением, файлом, медиа-потоком или другими данными.
* Веб-обозреватель или браузер — программное обеспечение для поиска, просмотра веб-страниц, то есть для запроса веб-страниц (преимущественно из Сети), для их обработки, вывода и перехода от одной страницы к другой.

Архитектура системы расчета надежности приведена на рис. 4.1

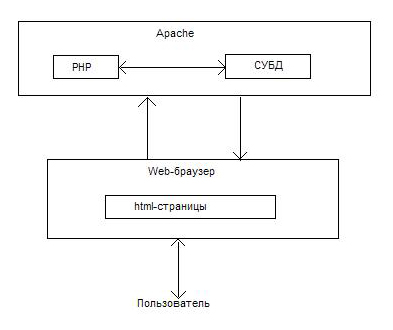


Рисунок 4.1- Архитектура системы расчета надежности.

Принцип работы PHP и СУБД. Имеется скрипт PHP (набор команд). Также есть сервер SQL, они существуют раздельно и друг с другом не связаны. PHP с помощью специальных команд устанавливает связь с сервером SQL. Как правило, PHP посылает запрос в базу данных. SQL обрабатывает эти команды и возвращает результат. Все это скрыто для пользователя.

Для пользователя отображаются только с результатами работы PHP и SQL в браузере в виде html-страницы.

### Разработка базы данных

### Концептуальное проектирование базы данных

Прежде чем приступать к разработке самой БД, определять сущности и связи, разработчик, прежде всего, должен уяснить, для кого разрабатывается база данных, и что она должна делать, какие функции выполнять.

На данном этапе собирается максимальный объем информации, относящейся к предметной области, для которой планируется разработать базу данных. Предметная область - часть реального мира, подлежащая изучению с целью организации управления и, в конечном счете, автоматизации. Каждый фрагмент предметной области характеризуется множеством объектов и процессов, использующих объекты, а также множеством пользователей, характеризуемых различными взглядами на предметную область. Информацию можно получить либо посредством изучения документов, характеризующих предметную область, либо при помощи интервью с заказчиком, заинтересованным в разработке.

Для того, чтобы построить модель будущей базы, необходимо произвести анализ, в результате которого будут определены объекты-сущности, информацию о которых необходимо будет хранить, а также определить как взаимодействуют объекты между собой.

Разрабатываемая информационная система предназначена для хранения данных о надежности элементов. С нашей БД могут работать студенты при выполнении лабораторной работы.

Вторым шагом в разработке БД является описание предметной области. Причём, чем точнее и глубже разработчик будет вести это описание, тем полнее и правильнее получиться сама БД, и только тогда она будет работать наиболее эффективно.

Основная таблица базы данных будет содержать следующие поля:

- название элемента;

- тип элемента. Схема объекта расчета надежности состоит из элементов разного типа (резисторы, диоды, интегральные микросхемы и тд). При расчете надежности с учетом различных условий эксплуатации выбирается соответствующий коэффициент, свой для каждого типа элемента;

- базовая интенсивность отказов элемента. Здесь находится базовое значение интенсивности отказов для каждого элемента;

- интенсивность отказов элемента в режиме хранения. Наша система позволяет исследовать элементы не только при непосредственной их работе, но также и учитывать изменения в их надежности при хранении. Поэтому в базу данных добавлен и этот параметр.

### Логическое проектирование базы данных

Разработку логической модели модно осуществлять различными методами. Наиболее формализованным и простым для понимания является метод «сущность-связь», или ER-метод.

Логическая структура базы данных определяет:

* + таблицы и их имена, также называемые *сущностями* (entities);
  + имена полей, также называемые *атрибутами* (attributes) каждой таблицы;
  + характеристики полей, например, уникальность их значения и допустимость значений NULL, а также тип данных, хранимых в поле;
  + первичный ключ каждой таблицы — поле (несколько полей) со значениями, уникально идентифицирующими каждую запись в таблице. В таблице также могут существовать другие уникальные поля, но только одно из них рассматривается как уникальный ключ доступа для поиска записей — первичный ключ. В таблице не обязательно должен существовать первичный ключ, однако рекомендуется оп­ределять его для каждой таблицы;
  + связи между таблицами. Записи в таблице могут зависеть от одной или несколь­ких записей другой таблицы. Такие отношения между таблицами называются *связями.* Связь определяется следующим образом: поле или несколько полей одной таблицы, называемое *внешним ключом,* ссылается на первичный ключ другой таблицы.

На основе вышеописанных рассуждений введем следующие сущности и их атрибуты.

* Справочник (Название, Тип элемента, Базовая интенсивность отказов).
* Механическое воздействие (Условия эксплуатации, Коэффициент при вибрации, Коэффициент при ударной нагрузке, Коэффициент, учитывающий суммарное воздействие)

База данных, составленная по этой информационной модели, обладает избыточностью, отношения не нормализованы.

Нормализация – это процесс улучшения качества отношений между таблицами и, как следствие, свойств базы данных в целом. Основополагающим принципом является наложение некоторых ограничений и правил на структуру таблиц в базе данных.

В теории реляционных баз данных выделяют следующие нормальные формы:

* Первая нормальная форма (1NF).
* Вторая нормальная форма (2NF).
* Третья нормальная форма (3NF).
* Нормальная форма Бойса – Кодда (BCNF).
* Четвертая нормальная форма (4NF).
* Пятая нормальная форма (5NF).

Мы рассмотрим только первые три. Вызвано это тем, что приведение базы данных к третьей нормальной форме включительно является достаточным условием для того, чтобы база данных соответствовала основным правилам построения качественной базы данных: доступность, расширяемость, непротиворечивость.

Таблица находится в 1НФ, если все ее атрибуты содержат только данные простых типов. Таблица не должна иметь повторяющихся записей, групп полей, необходимо наличие, по крайней мере, одного уникального индекса.

В рассматриваемой базе данных все отношения (таблицы) находятся в 1НФ, т.к. имеют ключевые поля, которые по определению являются уникальными индексами.

Таблица находится во 2НФ, если она находится в 1НФ и каждый ее неключевой атрибут (тот, который не входит в состав первичного ключа) полностью зависит от первичного ключа.

Понятие третьей нормальной формы основывается на понятии нетранзитивной зависимости. Транзитивная зависимость наблюдается в том случае, если один из двух описательных реквизитов зависит от ключа, а другой описательный реквизит зависит от первого описательного реквизита. Таблица находится в 3НФ, если она находится во 2НФ и каждый неключевой атрибут нетранзитивно зависит от первичного ключа.

Наша БД имеет не такую сложную структуру, поэтому ее необходимо привести лишь к 1НФ, введя ключевое поле id.

Тогда сущность Справочник будет иметь следующий вид:

Справочник (Id, Название элемента, Тип элемента, Базовая интенсивность отказов, Группа).

Для создания логической и физической структуры базы данных использовался пакет PhpMyAdmin 2.6.1.

PhpMyAdmin - это некоммерческое приложение, написанное на языке PHP, реализующее довольно удобный и функциональный Web-интерфейс к базе данных MySQL.

С помощью PhpMyAdmin можно создавать, удалять и редактировать таблицы, выполнять отдельные SQL-запросы, создавать текстовые дампы данных, просматривать список процессов, создавать и удалять пользователей, изменять их привилегии и многое другое.

## Разработка пользовательского интерфейса

Данная система расчета надежности реализована в виде web-приложения, что позволяет работать с ней не только непосредственно в университете, но и удаленно, что является преимуществом перед существующими аналогами. Также эта система удобна тем, что не требует для своей работы установки специального программного обеспечения.

В системе расчета надежности имеется возможность самостоятельного ввода параметров внешнего воздействия на схему (температура, механическое воздействие, условий эксплуатации), параметров нагруженности элементов, выбор корпуса, коэффициента приемки. Также имеется возможность ввода времени нахождения аппаратуры в полете и на земле, времени работы, хранения оборудования

При входе в систему пользователь попадает на главную страницу. Здесь предлагается создать уровень (блок) в котором будут располагаться подуровни (платы).

### Разработка структуры пользовательского интерфейса

На сегодняшний день, создание внутренних корпоративных веб-приложений уже, пожалуй, не просто дань моде, когда все, что так или иначе связано с интернетом считалось популярным и прогрессивным. Бизнес-приложения предприятий, перенесенные на новую технологическую "веб-оснастку" действительно значительно уменьшают издержки по поддержанию данных приложений в актуальном состояние в дальнейшем. Вот краткий перечень достоинств, которыми обладают корпоративные веб-приложения:

* не требуют инсталляции и обновления клиентского программного обеспечения;
* снижают затраты на обучение - в качестве клиентской части используется стандартный веб-браузер;
* пользователи могут работать на любой платформе;
* логика приложения сосредоточена на стороне сервера;
* возможность интеграции с ресурсами интернета;
* создание сколь угодно привлекательного веб-интерфейса.

Если мы будем рассматривать веб-приложения с точки зрения программиста, то представить их можно как некий особый класс систем клиент-сервер, в которых взаимодействие с пользователем осуществляется через стандартный веб-браузер. Если сравнивать структуру веб-приложения и классического приложения, основанного на технологии клиент-сервер, то можно охарактеризовать составляющие компоненты следующим образом:

* В роли "тонкого клиента", который отвечает за отображения данных и передачу их от пользователя серверу, выступает браузер. Пользовательский интерфейс всецело определяется HTML-документом, со всеми возможными дизайнерскими ухищрениями.
* Веб-сервер обеспечивает работу по протоколу HTTP, принимает запросы от клиента, взаимодействует непосредственно с веб-приложением, передает ответы клиенту. Веб-приложение - программа, которая, используя веб-сервер, обрабатывает запросы от клиента, производит необходимые манипуляции с данными, передает ответы клиенту.

Таким образом, разрабатывая интерфейс приложения нужно учитывать его web – ориентацию. В качестве наиболее подходящего было выбрано приложение, имеющее одну главную страницу со ссылочным меню.

### Ввод данных

В системе расчета надежности пользователю предлагается ввести условия эксплуатации для заданной схемы:

* температура. Пользователю предлагается выбрать температуру из интервала 25..60 °C.
* условия эксплуатации. Здесь предлагаются на выбор следующие условия: лабораторные, стационарные, автофургонные, железнодорожные, корабельные, самолетные. Также имеется возможность выбора дополнительных условий использования аппаратуры. Это эксплуатация при вибрации, при ударных нагрузках, а также суммарное воздействие.
* влажность. Предлагается выбрать значения: 60..70% и 90..98%;
* высота. Имеется возможность выбора высоты, на которой используется оборудование. Предлагается множество значений от 0 до 40 км;
* приемка. ВП (5) и ОС (9).

Далее после ввода необходимых данных следует нажать кнопку «Расчет», после чего необходимая информация передастся файлу-обработчику, с помощью которого будет произведен расчет необходимых параметров.

### Обработка введенных пользователем данных

В системе имеется база данных, в которую занесены элементы с их базовыми значениями интенсивности отказов. При выборе пользователем параметров эксплуатации базовые значения интенсивности отказов подставляются в соответствующие формулы и рассчитывается интенсивность отказов системы, а также такие параметры как вероятность безотказной работы, гамма-процентная наработка.

При вводе условий эксплуатации значение коэффициента выбирается исходя из двух условий (условий эксплуатации и условий механического воздействия). Поэтому в базе данных создана таблица, из которой в зависимости от действий пользователя выбирается коэффициент. (рис.4.2)



Рисунок 4.2 – Таблица коэффициентов различных условий механического воздействия.

Запрос к базе данных осуществляется функцией mysql\_query. Затем отобранные по заданным в запросе параметрам значения интенсивности отказов подставляются в формулу, и рассчитывается суммарная интенсивность отказов с соответствующими для каждого типа элементов коэффициентами. Полученный результат отображается на странице.

# Анализ надежности и способов её повышения для типовых схем и конструкций электронных средств.

## Краткое описание типовых схем и конструкций электронных средств как объектов расчета надежности

В системе расчета надежности имеется набор элементов для некоторых вариантов схем, параметры надежности которых и предстоит рассчитать в рамках лабораторной работы. Расчету подлежат:

* Блок индикации положения алюминиевого расплава в электромагнитном тигле;

Рассмотрим эту схему.

### Схема блока индикации положения алюминиевого расплава в электромагнитном тигле

Данное устройство состоит из четырех (по числу контролируемых тиглей) каналов индикации и блока питания.

Схема канала индикации включает:

* входную часть (измерительный мост и повторитель);
* фильтр низкой частоты (ФНЧ);
* масштабный усилитель;
* схему выделения переменной составляющей полезного сигнала;
* устройство световой индикации.

Входная часть, необходимая для выделения полезного сигнала, включает выпрямительные и фильтрующие цепи (VD1-VD4, R3, R4, C1, C2), балансный потенциометр R7, а также повторитель, собранный на микросхеме операционного усилителя DА3, необходимый для согласования измерительного моста с последующей частью схемы.

Фильтр НЧ выполнен на микросхемах DА6 и DА9. Включение фильтра в схему обусловлено необходимостью подавления напряжения помехи, накладывающейся на полезный сигнал. Параметры данного фильтра V-го порядка соответствуют частоте среза fср = 10 Гц. Это позволяет осуществить эффективное подавление помехи, в которой превалирует составляющая частотой f = 100 Гц.

Масштабный усилитель (микросхемы DA1, DA2) обеспечивает увеличение амплитуды полезного сигнала.

Оконечной частью представленной на рис. 1 схемы канала индикации является устройство, преобразующее полезный сигнал и его переменную составляющую в свечение индикаторных элементов (светодиодов VD7, VD10, VD13). Данное устройство включает три одинаковых компаратора, собранных на микросхемах DA4, DA7 и DA8, и логические элементы И-НЕ (микросхема DD1). На вход первого компаратора (микросхема DA4) полезный сигнал с выхода усилителя DA2 подается непосредственно, а на входы двух других – через схему выделения переменной составляющей (C3, R13, R16, DА5).

Зажигание светодиода VD13 происходит при загрузке образца Al в электромагнитный тигель (ЭМТ), а при выпадении образца расплава из ЭМТ этот светодиод гаснет. При возникновении колебаний образца на выходе усилителя DA2 появляется переменное напряжение с частотой, равной частоте указанных колебаний; амплитуды напряжения пропорциональны амплитудам механических колебаний образца. Если отклонения последнего от положения равновесия столь велики, что соответствующие им разнополярные напряжения на выходе усилителя DA5 превышают по абсолютным значениям пороговые напряжения, выставляемые потенциометрами R22 и R23, то работа схем индикации обеспечивает зажигание светодиодов VD11 и VD12 «превышение уровня».

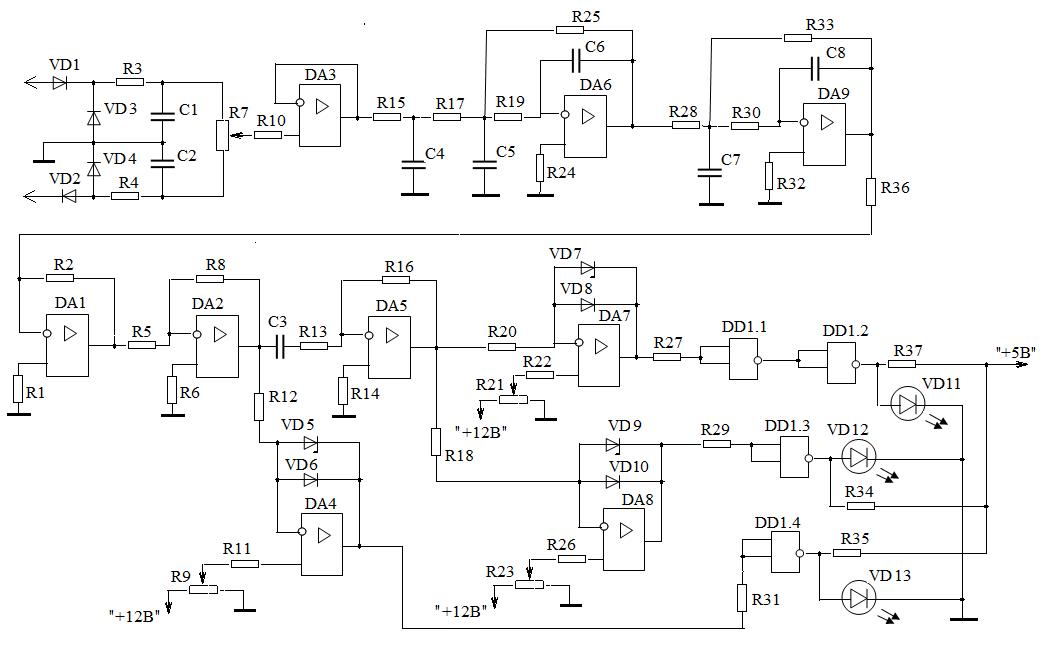


Рисунок 5.2 - Принципиальная схема канала индикации положения расплава.

**Блок питания** вырабатывает следующие стабилизированные напряжения, необходимые для работы прибора:

U1,2 = ± 12,6 В; U3 = + 5,6 В.

В состав блока входит трансформатор Тр1 и три источника указанных выше стабилизированных напряжений. Каждый из этих трех источников включает в себя: однофазный мостовой выпрямитель на диодах, транзисторный стабилизатор и емкостной фильтр. Все транзисторные стабилизаторы напряжений собраны по известной схеме с составным транзистором.

**Конструкция.** Несущими элементами конструкции блока являются основание и лицевая панель. С внутренней стороны к основанию прикреплена базовая плата. На базовой плате блока в разъемах установлены платы каналов индикации, являющиеся по конструктивной иерархии ЭВС модулями 1-го уровня, ТЭЗами. На каждой из этих плат размещены элементы схемы, кроме светодиодов, вынесенных на лицевую панель. Кроме плат каналов индикации на базовой плате размещен трансформатор и другие элементы блока питания.

Данная схема интересна возможностью разных подходов конструирования. Например, элементы этой схемы можно размещать как на одной плате, так и на нескольких, что позволяет заменить блок при выходе его из строя. Эта особенность обеспечивает ремонтопригодность блока.

## Факторы, влияющие на надежность электронных средств, и учет этих факторов в моделях расчета

При выполнении расчетов надежности оперируют с некоторым математическим объектом, отражающим наиболее существенные и влияющие на надежность свойства реального изделия. Такой идеализированный объект называют математической моделью надежности.

Наиболее простой является модель невосстанавливаемого изделия (элемента), так как она сводится к модели безотказности.

Модель надежности должна отражать:

* характер отказов электрорадиоизделий (ЭРИ); поскольку отказы – случайные события, речь идет о выборе подходящей функции распределения наработки до отказа;
* характеристики надежности, связанные с технологическими факторами, тепловыми, механическими и прочими воздействиями, процессами старения;
* электрические режимы работы ЭРИ;
* соединение элементов в схеме надежности (последовательное, параллельное или смешенное).

Расчет надежности – расчет, в результате которого получаются количественные значения показателей надежности исследуемого объекта. Целью расчета надежности является: сравнение вариантов при выборе технического решения; получение приближенных оценок показателей надежности.

Реальные объекты в большинстве случаев состоят из совокупностей взаимосвязанных элементов. При этом связь между элементами может быть не только физической или технической, но и в смысле надежности. При возможности расчленения сложной системы на отдельные элементы, для каждого из которых можно определить показатели надежности, для расчета надежности системы используются структурные схемы - модели надежности систем. Чаще всего структурная схема системы, построенная для решения задач надежности, не совпадает с функциональной схемой системы или конструктивной схемой соединения ее элементов. Модель надежности системы строится на основе анализа влияния определенного вида отказов элементов на надежность системы в целом.

При выполнении расчетов надежности оперируют не с самим техническим изделием, а с некоторым математическим объектом, который отражает наиболее существенные свойства реального изделия и называется математической моделью надежности. Наиболее простой является модель невосстанавливаемого объекта – модель безотказности. Построение модели безотказности состоит из:

* формирования признаков отказа;
* выбора и обоснования функции распределения наработки до отказа;
* определение численных значений параметров функции надежности по статистическим данным, полученным по испытаниям и в процессе наблюдения при эксплуатации;

На основании составленной модели возможен расчет любых показателей надежности.

### Модель безотказной работы изделия (системы)

При построении этой модели существенным представляется разделение всевозможных отказов на внезапные и постепенные.

Постепенные (износные) отказы возникают в результате постепенного протекания того или иного процесса повреждения, прогрессивно ухудшающего выходные параметры объекта. К постепенным отказам относятся отказы, связанные с процессами старения (изнашивания, коррозии, усталости и ползучести материалов).

Внезапные отказы возникают в результате сочетания неблагоприятных факторов и случайных внешних воздействий, превышающих возможности объекта к их восприятию (т.е. нарушение работоспособности вследствие внешних и внутренних факторов).

Считается доказанным, что, время возникновения отказов из-за уходов параметров за допустимые значения подчиняется нормальному закону распределения, а при внезапных отказах – экспоненциальному.

Основными количественными характеристиками надежности невосстанавливаемых объектов в процессе эксплуатации являются: вероятность исправной работы, средняя наработка до отказа, частота и интенсивность отказов.

Наиболее полно характеризовать надежность элементов можно частотой отказов. Это объясняется тем, что частота отказов является плотностью распределения времени возникновения отказов и поэтому полностью описывает случайную величину с вероятностной точки зрения. Однако плотность достаточно трудно определить экспериментально. Поэтому наиболее распространенными на практике являются вероятность безотказной работы и интенсивность отказов. Так как указанные характеристики однозначно связаны межу собой, то достаточно знать одну из них, чтобы вычислить другую.

Наиболее удобной характеристикой является интенсивность отказов λ, так как ее наиболее просто получить экспериментально и она, как правило, имеет простое аналитическое выражение.

Функция ***экспоненциального*** распределения времени до отказа имеет вид:

. (2.36)

Соответственно путем дифференцирования Q(t) находится плотность вероятности:

. (2.37)

Нормальный закон характеризуется плотностью вероятности вида

; (2.38)

данная функция полностью задается двумя параметрами: математическим ожиданием ***m*** и средним квадратическим отклонением ***σ***.

Если при экспоненциальном распределении интенсивность отказов постоянна (λ = const), то при нормальном распределении величина λ монотонно возрастает. Закону Гаусса подчиняются постепенные отказы, когда плотность их возникновения, низкая в начале эксплуатации, затем достигает максимума и далее снижается.

Нормальное распределение описывает наработку до отказа в области старения и износа.

Теперь надо подчеркнуть: и ***нормальное распределение*** (2.38) ***и*** ранее рассмотренное ***экспоненциальное*** (2.36) ***представляют своеобразные крайние положения:*** в первом случае функция f(t) строго симметрична при монотонном возрастании λ, во втором – наблюдается выраженный асимметричный характер f(t) при λ = const.

В теории надежности большое распространение получили и некоторые другие распределения, представляющие промежуточные случаи изменения λ(t). Это, прежде всего, двухпараметрическое распределение Вейбулла [1 - 3], плотность вероятности которого описывается зависимостью

, (2.39)

где α – параметр формы кривой распределения; λ – параметр масштаба.

Экспоненциальное распределение является частным случаем (2.36) при α = 1. При α = 3,3 распределение Вейбулла становится близким к нормальному.

При α > 1 этот закон подходит для описания функции надежности быстро стареющих изделий.

### Модели надежности при различном соединении элементов

#### Последовательное соединение элементов

Последовательным (в смысле надежности) называют такое соединение элементов в системе, при котором отказ хотя бы одного элемента приводит к отказу всей системы. Последовательное соединение в указанном выше смысле совсем не следует понимать, как физически последовательное соединение элементов.

Расчетная схема представлена на рис.2.3

Рисунок 2.3 - Схема надежности при последовательном соединении:

p1(t)

p2(t)

Pn(t)

pi(t) – вероятность безотказной работы i – го элемента

По такой схеме надежности спроектировано большинство технических систем.

Отказы элементов предполагаются независимыми, то есть отказ любой группы элементов никак не повлияет на вероятностные характеристики остальных элементов. Элемент понимается в широком смысле слова - это один из самостоятельных участков последовательного соединения. Каждый элемент, включаемый в надежностную схему, характеризуется интенсивностью отказов *λi* и вероятностью безотказной работы *Pi(t)*; считают, что отказы отдельных элементов независимы между собой (хотя бы в первом приближении), поэтому вероятность безотказной работы изделия (всей системы) рассчитывается по формуле

 . (2.40)

Для произвольного закона распределения времени работы до отказа

. (2.41)

Очень часто в практике расчетов Н основываются на экспоненциальном законе распределения, когда интенсивности отказов элементов постоянны (λi  = const), то есть

. (2.42)

Тогда

, (2.43)

где λэ – интенсивность отказов всей системы;

, (2.44)

тогда среднее время наработки до отказа находится как

*T1= 1/σΣ .* (2.45)

Общий период эксплуатации любой бортовой (самолетной, ракетной, …) аппаратуры может быть разделен на два участка: tp – время в работе в условиях полета; tхр – время нахождения в выключенном состоянии. Интенсивности отказов, соответственно, равны λр и λхр . Еще надежность зависит от числа *N*ц циклов включения/ выключения и значения интенсивности отказов *λ*ц за один цикл.

При экспоненциальном законе распределения времени безотказной работы расчет вероятности безотказной работы следует производить по формуле

 . (2.46)

Если интенсивность отказов объекта за период времени t1 равна σ1, за последующий период t2 равна σ2 и т.д. Тогда интенсивность отказов элемента за период времени

*t = t1 + t2 + t3 + ….+ tk*

будет

*λэкв = *. (2.47)

При известном значении λ (найденном с помощью компьютерной системы расчета) при экспоненциальном законе распределения можно найти не только среднее время наработки до отказа Т1, но еще гамма-процентную наработку Тγ, которая находится по формуле (2.21).

#### Параллельное соединение элементов

При расчетах надежности параллельным называется такое соединение элементов [1-3], при котором отказ всего соединения в целом происходит при отказе всех элементов системы (элементы дублируют друг друга).

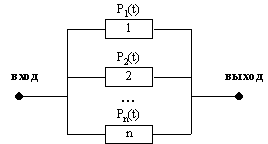


Рисунок 2.4 - Схема надежности при параллельном соединении: pi(t) – вероятность безотказной работы i – го элемента

Эта система – с резервированием элементов – работает, если работает хотя бы один из параллельно соединенных элементов.

Вероятность события отказа в течение времени t состоит в совместном появлении отказов всех *n* параллельно включенных в модель надежности элементов:

, (2.48)

где *qi(t) –* вероятность отказа одного элемента.

Вероятность безотказной работы системы с рассматриваемой схемой надежности определяется по формуле:

. (2.49)

#### Системы с последовательно-параллельным соединением элементов.

Схема надежности системы приведена на рисунке 2.5 для случаев поэлементного и общего дублирования (соответственно рис. 2.5,*а* и *б*)

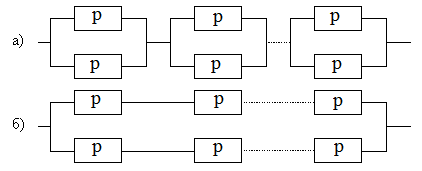


Рисунок 2.5 - Схемы надежности для параллельно-последовательного (а) и последовательно-параллельного (б) соединений элементов; эти схемы соответствуют случаям раздельного (поэлементного) и общего дублирования

Вероятности безотказной работы для приведенных схем определяются по формулам

 ; (2.50)

 . (2.51)

### Модели надежности, учитывающие нагрузку элементов при внешних и внутренних воздействиях

Значения эксплуатационной интенсивности отказов большинства электрорадиоизделий (ЭРИ) рассчитывается по математическим моделям, имеющим вид:

 (2.52)

или , (2.53)

где  - исходная (базовая) интенсивность отказов типа (группы ЭРИ), при номинальной электрической нагрузке и температуре окружающей среды tокр = 25 ° С;

*λб* – базовая интенсивность отказов типа (группы) ЭРИ для усредненных режимов применительно к стационарной аппаратуре при электрической нагрузке 0,4 от номинальной, tокр = 30 ° С; значения *λб (λб/)* приводятся для отдельных типов ЭРИ в справочной литературе.

*Кр* – коэффициент режима, учитывающий изменение  в зависимости от электрической нагрузки и (или) температуры окружающей среды;

*Кi* – коэффициенты, учитывающие изменение эксплуатационной интенсивности отказов *λЭ* в зависимости от различных факторов; *n* – число учитываемых факторов.

Коэффициенты *Кi* , входящие в формулу (2.53), можно разделить на две группы:

* первая группа (табл.2.1) объединяет коэффициенты, общие для ЭРИ различных типов (классов) и характеризующие режимы и условия эксплуатации, уровень качества производства и приемки ЭВС (РЭС);
* вторая группа коэффициентов включается в модели отдельных конкретных типов (классов) ЭРИ и отражает зависимость эксплуатационной интенсивности отказов *λЭ* от конструктивных, функциональных и технологических особенностей этих элементов.

Ввиду большого разнообразия коэффициентов моделей отдельных классов ЭРИ в табл.1 приведены только некоторые из них.

Таблица 2.1. Общие коэффициенты моделей

|  |  |
| --- | --- |
| Названия и условные обозначения коэффициентов | Факторы, учитываемые коэффициентами |
| Коэффициент режима КР(Т)  Коэффициент приемки КПР  Коэффициент эксплуатации КЭ  Коэффициент ионизирующих излучений КИЗЛ  Кусл | Величина электрической нагрузки и (или) температура окружающей среды (корпуса ЭРИ)  Степень жесткости требований при контроле качества и приемке изделий  Степень жесткости условий эксплуатации  Степень жесткости внешних ионизирующих излучений  Условия: эксплуатации или хранения |

Таблица 2.2. Коэффициенты моделей отдельных классов ЭРИ

|  |  |  |
| --- | --- | --- |
| Обозначения коэффициентов | | Факторы, учитываемые коэффициентами |
| Интегральные микросхемы (ИС) | | |
| КСТ  КV  Ккорп | Сложность ИС и температура окружающей среды  Величина напряжения питания для ИС КМОП  Тип корпуса | |
| Полупроводниковые приборы | | |
| Кф  КS1  КP(I) | Функциональное назначение  Отношение рабочего напряжения к макс. доп.  Нагрузка по мощности рассеяния (току) | |
| Резисторы | | |
| KR  Км  КS1  Кстаб | Величина омического сопротивления  Номинальная мощность  Отношение рабочего напряжения к максимально допустимому по ТУ  Допуск изготовления (точность резистора) | |

В литературе предлагаются также модели, отличающиеся от (2.52), (2.53). Так в [1] учет нескольких видов нагрузки – при условии независимости их воздействия – в значении интенсивности отказов элемента предлагается находить по формуле

, (2.54)

где v – число учитываемых видов нагрузки; h – поправочный коэффициент, учитывающий особенности элемента.

Основными из вышеприведенных коэффициентов [4] являются:

* коэффициент Кт, учитывающий изменение интенсивности отказов в зависимости от температуры окружающей среды;
* коэффициент приемки Кпр, отражающий два уровня качества изготовления изделий: в обозначениях [4] приемка ОВП («5») и ОС (приемка «9»);
* коэффициент условий Кусл , отражающий изменение интенсивности отказов в зависимости от нахождения аппаратуры в режиме ожидания (хранения) или нахождения ее в работе; в режиме ожидания интенсивность отказов рассчитывается по формуле

λх = λхсг·Кtx·Кпр·Кусл , (2.55)

где Кусл учитывает условия хранения (значения Кусл равны 1 и 1,2 для отапливаемых и неотапливаемых помещений соответственно); λхсг - базовая интенсивность отказов при хранении.

Наиболее сложной задачей при составлении расчетных моделей надежности ЭВС (РЭС) является отыскание численных значений коэффициентов, учитывающих нагрузки и режимы работы ЭРИ отдельных типов. Анализ справочных данных показывает, что значения интенсивностей отказов могут нелинейно зависеть от коэффициентов нагрузки. Это согласуется и с принятыми в [4] моделями расчета.

В имеющейся по вопросам расчета надежности литературе [3] предлагается в моделях (2.51), (2.52) учитывать некоторые внешние воздействующие факторы поправочными коэффициентами, зависящими *не от физических параметров воздействия, а от определяющих уровень этих воздействий условий применения изделия*. Так в табл. 3 приведены значения поправочного коэффициента, учитывающего механические воздействия (вибрации, удары) при различных условиях эксплуатации ЭВС (РЭС).

Таблица 2.3. Поправочный коэффициент Кλ1 при различных условиях эксплуатации

|  |  |  |  |
| --- | --- | --- | --- |
| Условия эксплуатации | Кλ1 | | |
| При вибрации | При ударах | Суммарном возд. |
| Лабораторные  Стационарные  Автофургонные  Железнодорожные  Корабельные  Самолетные | 1,00  1,04  1,35  1,40  1,30  1,46 | 1,00  1,03  1,08  1,10  1,05  1,13 | 1,00  1,07  1,46  1,54  1,37  1,65 |

По виду объекта установки РЭА можно разде­лить на три группы: стационарную, транспортируемую и портативную [8], техническое регламентирование которых приведено на рисунке 2.6.

***Стационарная РЭА*** - это аппаратура, эксплуатируемая в отапливае­мых и неотапливаемых помещениях, помещениях с по­вышенной влажностью, на открытом воздухе, в производственных цехах. Условия эксплуатации и транспортирования такой аппара­туры характеризуются весьма широким диапазоном рабочих (-50...+50 °С) и предельных (-50... +65 °С) температур, влажностью до 90...98 %, вибрацией до 120 Гц при 4...6 g, наличием многократных (до 5 g) и одиночных (до 75 g) ударов, воздействием дождя до 3 мм/мин и соляного тумана с дис­персностью капель до 10 мкм и содержанием воды до 3 г/м3.

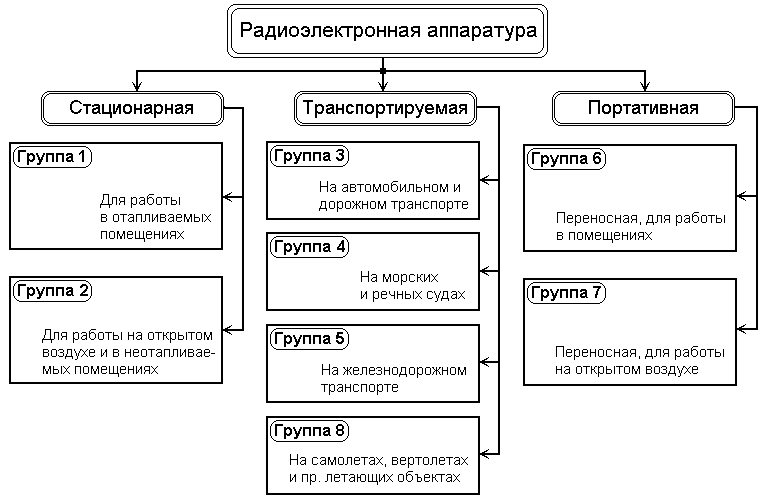


Рисунок 2.6 - Классификация РЭА по объектам установки и эксплуатации

***Транспортируемая РЭА*** - это аппаратура, устанавливаемая и экс­плуатируемая на автомобилях и автоприцепах, железнодорожном и гусе­ничном транспорте, на судах различных классов, на борту самолетов и вертолетов. Специфика работы этого вида аппарату­ры предопределяет повышенное воздействие механических факторов. Каж­дый вид транспорта имеет собственные вибрационные характеристики. Для предупреждения повреждения аппаратуры необхо­димо, чтобы вся она и отдельные ее части имели собственные частоты колебаний вне диапазона частот вибрации транспортного средства.

На РЭА, установленную на автомобильном транспорте, могут воздей­ствовать вибрация частотой до 200 Гц и удары, вызванные неровной доро­гой. При движении железнодорожного транспорта возможны внезапные толчки (при маневрировании - удары с ускорением до 40 g). Биение колес о стыки рельсов вы­зывают вибрацию с частотой до 400 Гц при ускорении до 2 g. Особо жест­ким воздействиям подвергается конструкция РЭА, эксплуатируемая на гусе­ничном транспорте. Здесь вследствие «стука» гусениц частота вибраций может доходить до 7000 Гц с амплитудой ±0,025 мм. Кроме того, постоянно воздей­ствие акустического шума.

Для РЭС(ЭВС) в морском исполнении, устанавливаемых на суда. характерными усло­виями работы является наличие вибраций, ударных нагрузок и агрессивной (морской) атмосферы. Вибрация на судне вызывается работой винтов, гребного вала, двигателей и гидродинамическими силами при движении судна по неспокойному морю. Диапазон частот вибраций на кораблях обычно не превышает 25 Гц с небольшой амплитудой вибраций.

На самолетах электронная аппаратура на­ходится, как правило, в фюзеляже. При этом на нее воздействуют вибраци­онные нагрузки частотой до 500 Гц с амплитудой до 10 мм и акустический шум, уровень которого достигает 150 дБ при частоте 50... 10000 Гц.

Различают и специальные виды РЭА, эксплуатируемые, например, в условиях химического производства. Для них характерны сверхбольшие значения одного - трех внешних факторов, на устойчивость к которым и проектируется конструк­ция такой РЭА.

Каждой из групп аппаратуры соответствует совокупность кли­матических и механических факторов, которой она должна соответствовать. В приводимой далее табл. 4, взятой из [8,] приводятся значения внешних воздействий для аппаратуры различных групп.

Таблица 2.4. Значения воздействующих факторов на группы РЭА

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Внешние факторы | | Группа РЭА | | | | | | | | | | | | | | | |
|  | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | |
| ***Климатические*** | |  | |  | |  | |  | |  | |  | |  | |  | |
| Пониженная температура, °С. | |  | |  | |  | |  | |  | |  | |  | |  | |
| *предельная* 1 степень | | -40 | | -40 | | -40 | | -40 | | -40 | | 5 | | -40 | | -60 | |
| 2 степень | | -50 | | -50 | | -50 | | -50 | | -50 | | 5 | | -50 | | -60 | |
| *рабочая* 1 степень | | 5 | | -10 | | -25 | | -10 | | -25 | | 5 | | -10 | | -40 | |
| 2 степень | | 5 | | -25 | | -40 | | -10 | | -40 | | 5 | | -25 | | -50 | |
| Повышенная температура, °С. | |  | |  | |  | |  | |  | |  | |  | |  | |
| Предельная | | 55 | | 60 | | 60 | | 60 | | 60 | | 50 | | 60 | | 75 | |
| Рабочая | | 40 | | 50 | | 50 | | 50 | | 50 | | 40 | | 50 | | 60 | |
| Относительная влажность, % при температуре, °С. | | 86 | | 93 | | 93 | | 93 | | 93 | | 80 | | 93 | | 98 | |
| 1 степень | | 25 | | 25 | | 25 | | 25 | | 25 | | 25 | | 25 | | 25 | |
| 2 степень | | 25 | | 40 | | 40 | | 40 | | 40 | | 40 | | 40 | | 25 | |
| время выдержки, ч. | | 48 | | 72 | | 72 | | 72 | | 72 | | 36 | | 72 | | 48 | |
| Интенсивность дождя, мм/мин | | — | | 3 | | 3 | | 3 | | 3 | | - | | 3 | | 3 | |
| время выдержки, ч. | | — | | 0,33 | | 0,33 | | 0,33 | | 0,33 | | - | | 0.33 | | 0,33 | |
| Пониженное давление, кПа | | 61 | | 61 | | 61 | | 61 | | 61 | | 61 | | 61 | | 61 | |
| время выдержки, ч. | | 2...6 | | 2...6 | | 2...6 | | 2...6 | | 2...6 | | 2…6 | | 2…6 | | 2...6 | |
| Поток пыли: | |  | |  | |  | |  | |  | |  | |  | |  | |
| скорость потока, м/с. | | — | | 10 | | 10 | | — | | 10 | | - | | 10 | | — | |
| время выдержки, ч. | | — | | 1 | | 1 | | — | | 1 | | - | | 1 | | — | |
| Морской туман: | |  | |  | |  | |  | |  | |  | |  | |  | |
| температура, °С. | | 27 | | 27 | | 27 | | 27 | | 27 | | - | | 27 | | 27 | |
| содержание воды, г/м3 | | 2...3 | | 2...3 | | 2...3 | | 2...3 | | 2...3 | | - | | 2…3 | | 2...3 | |
| время выдержки, ч. | | 24 | | 48 | | 48 | | 48 | | 48 | | - | | 48 | | 48 | |
| ***Механические*** | |  | |  | |  | |  | |  | |  | |  | |  | |
| Вибрация на одной частоте: | |  | |  | |  | |  | |  | |  | |  | |  | |
| частота, Гц | | 20 | | 20 | | 20 | | 20 | | 20 | | 20 | | 20 | | 20 | |
| ускорение, g | | 2 | | 2 | | 2 | | 2 | | 2 | | 2 | | 2 | | 2 | |
| время выдержки, ч. | | 0,5 | | 0,5 | | 0,5 | | 0,5 | | 0,5 | | 0,5 | | 0,5 | | 0,5 | |
| Вибрация в диапазоне частот: | |  | |  | |  | |  | |  | |  | |  | |  | |
| частота, Гц, от 1 до | | — | | — | | 200 | | 200 | | 300 | | — | | — | | 2500 | |
| ускорение, g | | — | | — | | 10 | | 5 | | 2 | | — | | — | | 1..13 | |
| время выдержки, ч. | | — | | — | | 12 | | 4 | | 12 | | — | | — | | 12 | |
| Одиночные удары: | |  | |  | |  | |  | |  | |  | |  | |  | |
| длительность, мс, от 15 до | | — | | — | | 75 | | 500 | | 40 | | — | | — | | 150 | |
| число ударов в 1 мин | | — | | — | | 15 | | 15 | | 15 | | — | | — | | 15 | |
| общее число ударов | | — | | — | | 60 | | 60 | | 60 | | — | | — | | 60 | |
| Удары многократные: | |  | |  | |  | |  | |  | |  | |  | |  | |
| длительность, мс | | — | | — | | 5-10 | | 5-15 | | 5-15 | | — | | — | | 2-10 | |
| число ударов в 1 мин | | — | | — | | 40-80 | | 40-80 | | 40-80 | | — | | — | | 40-80 | |
| ускорение, g | | — | | — | | 15 | | 15 | | 25 | | — | | — | | 5-15 | |
| общее число ударов | | — | | — | | 12000 | | 12000 | | 12000 | | — | | — | | 6000 | |

Рассмотрим подробнее построение математических моделей интенсивностей отказов типовых электрорадиоэлементов.

**Для интегральных микросхем** математическая модель определения коэффициента Кст дается [4] в следующем виде:

, (2.56)

где А и В – постоянные коэффициенты, t – температура окружающей среды; численные значения А и В приводятся в справочнике [4] для различных групп ИС (например, для аналоговых операционных усилителей А = 1,06·10-3, В = 23·10-3).

Значения λэ = Кст·λб приведены в таблице 5 в зависимости от температуры при λб = 3,7·10-8 1/ч

Таблица 2.5. Зависимость λэ от температуры

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| t, ° C | 25 | 30 | 35 | 40 | 50 | 60 |
| λэ·108, 1/ч | 3,72 | 4,17 | 4,68 | 5,25 | 6,61 | 8,31 |

**Для резисторов** математическая модель определения коэффициента режима выглядит следующим образом:

, (2.57)

где А, В, Nt , G, Ns , J, H – постоянные коэффициенты,

t - температура окружающей среды, °С; P, Pн - рабочая и номинальная мощности соответственно, Вт.

Для широко применяемых в ЭВС (РЭС) непроволочных постоянных резисторов (типов ВС, МЛТ, УЛМ, С1, С2, СЗ, С4 и других) значения постоянных модели согласно [4] таковы:

А = 0,26; В = 0,51; Nt = 343; G = 9,3; Ns = 0,88; J = 1; H = 0,87.

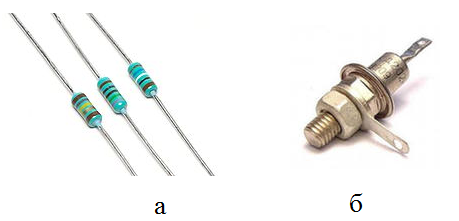


Рисунок 2.7 - Внешний вид непроволочных постоянных резисторов (а) и кремниевых полупроводниковых диодов (б)

**Для кремниевых полупроводниковых диодов** (кроме СВЧ диодов) математическая модель коэффициента режима выражается [4] в виде

; (2.58)

,

где А, NT, TM , L, Δt – постоянные данной модели;

t –температура окружающей среды; tпермакс – максимально допустимая температура перехода; tсниж – максимальная температура окружающей среды, для которой при 100% электрической нагрузке температура перехода достигает tпермакс; КЭЛ – отношение рабочей электрической нагрузки к максимально допустимой при температуре tсниж . Здесь коэффициент КЭЛ понимается как отношение рабочей величины среднего прямого тока через диод к максимально допустимому значению этого тока.

Для кремниевых диодов постоянные модели имеют следующие значения:

А = 44,1; NT = -2138; TM = 448; L = 17,7; Δt = 150;

tпермакс = 175 ° C ; tсниж = 25 ° C .

*Несложная численная оценка при t = 25 ° C, Кэл = 1 дает: F = 448; .*

**Излучатели полупроводниковые (**относятся к группе **оптоэлектронных приборов**) характеризуются коэффициентом режима, рассчитываемым по формуле

, (2.59)

где *IПР.СР.0 –* средний прямой ток излучателя в номинальном режиме;

*IПР.СР. –* средний прямой ток излучателя в рабочем режиме;

*t п0 –* температура p-n-перехода в номинальном режиме, °С;

*t п –* температура перехода в рабочем режиме, ° С;

*Еа –* энергия активации процесса деградации; *Еа* = 0,6 эВ;

*К*  - постоянная Больцмана; *К* = 8,625· 10-5 эВ/град;

*m –* показатель, зависящий от свойств полупроводникового кристалла и режима работы; для светодиодов, работающих в импульсном режиме, *m = 2;* для светодиодов, работающих в непрерывном режиме, *m = 1,5.*

Согласно [4] температура p-n-перехода полупроводниковых излучателей зависит от температуры окружающей среды *t* и определяется по формуле

; *tп0 = 45 ° С.* (2.60)

Проверочная численная оценка коэффициента Кр по приведенной выше формуле для импульсного режима работы оптоэлектронного излучателя (m=2) при нагрузке Iпр.ср / Iпр.ср.0 = 0,1 , температуре окружающей среды t = 30 ° C дает значение Кр = 0,004, совпадающее с приведенным в соответствующей таблице справочника [4].

Далее приведем взятую из [4] математическую модель коэффициента режима **конденсатора**:

, (2.61)

где А, В, NT, G, H, NS – постоянные модели; t – температура окружающей среды, °С ; U – рабочее напряжение, В; UН – номинальное напряжение, В.

Например, для керамических, тонкопленочных с неорганическим диэлектриком конденсаторов численные значения постоянных модели (2.61) таковы: А = 5,9·10-7 ; B = 14,3; NT = 398; G = 1,0; NS = 0,3; H = 3.

Далее приведем взятую также из [4] математическую модель для расчета коэффициента режима **трансформатора**:

 , (2.62)

где tm = tокр + tп ; tокр – температура окружающей среды; tn  - температура перегрева, рассчитываемая для частоты питания f = 50 Гц по формуле

tп = 0,25·tпТУ [3·(P/Pmax)2 + 1]. (2.63)

В этой формуле P и Pmax – соответственно рабочая и максимально допустимая мощности, Вт; tту – максимальная температура перегрева по ТУ; А, TM, G – постоянные, зависящие от класса изоляции и некоторых других характеристик [4].

Модель (2.62) иллюстрируется нижеследующей таблицей, где в зависимости от коэффициента нагрузки КН приведены интенсивности отказов λэ .

Таблица 2.6. Зависимость интенсивности отказов трансформатора от коэффициента нагрузки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| КН | 0,4 | 0,6 | 0,8 | 1,0 |
| λЭ ·109, 1/ч | 5,9 | 7,9 | 15,1 | 65,0 |

Как следует из приведенной зависимости, величина λ возрастает с увеличением коэффициента нагрузки трансформатора нелинейно; график такой зависимости, рассчитанный при tокр = 25 ° С, представлен на рис.2.7

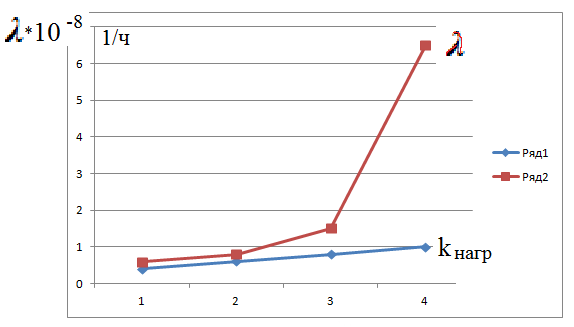
**

Рисунок 2.7. - Зависимость интенсивностей отказов λ трансформатора от коэффициента нагрузки kнагр

Значения интенсивности отказов **соединений** при эксплуатации рассчитывают по модели:

где λ б i – базовое значение интенсивности отказов i – го вида соединения,

 (2.64)

N i – количество соединений одного вида,

n – количество видов соединений в устройстве,

К э – коэффициент жесткости условий эксплуатации.

Модель распространяется на соединения, используемые во всех платах (узлах), кроме соединений (паек) в платах с металлизированными сквозными отверстиями. Все виды паек на платах с металлизированными сквозными отверстиями учитываются в модели расчета эксплуатационной интенсивности отказов этих плат.

Таблица 2.7. Значения базовой интенсивности отказов различных видов соединений

|  |  |
| --- | --- |
| Вид соединения |  |
| Ручная пайка ЭРИ без накрутки  Ручная пайка ЭРИ с накруткой  Пайка ЭРИ волной  Сварка  Обжимка (опрессовка)  Беспаяное соединение накруткой  Скрутка | 0,13  0,007  0,0069  0,0015  0,012  0,00068  0,026 |

Значения интенсивности отказов **многослойных плат** при эксплуатации рассчитывают по модели:

 (2.65)

где N 1 – количество сквозных отверстий, пропаянных волной;

N 2 – количество сквозных отверстий, пропаянных ручной пайкой;

К с – коэффициент, зависящий от количества слоев в плате.

Таблица 2.8. Значения базовой интенсивности отказов в зависимости от технологии межсоединений

|  |  |
| --- | --- |
| Технология межсоединений |  |
| Печатный монтаж  Навесной монтаж (дискретные проводники) | 0,0017  0,011 |

### Методика расчета проектной надежности при основном (последовательном) соединении элементов

В практике расчетов Н часто основываются на экспоненциальном законе распределения [1-3, 5], когда интенсивности отказов элементов постоянны (λi  = const), то есть

. (2.66)

Тогда

, (2.67)

где λэ – интенсивность отказов всей системы;

, (2.68)

Часто в системе есть определенное количество элементов одного и того же типа.

Пусть число элементов j-го типа равно mj, а всего типов элементов насчитывается k.

, (2.69)

где λj – интенсивность отказов элементов j-го типа (предполагается, что элементы одного типа имеют одинаковую Н);

mj· λj – суммарная интенсивность отказов элементов типа j.

Средняя наработка до отказа для рассматриваемого экспоненциального закона

*T1 = 1 / λэ* . (2.70)

При расчетах Н учитывают иерархическое построение аппаратуры (оборудования) технических систем [1,2,8]. А именно: конструктивно законченные единицы (модули) входят в сборочные единицы более высокого уровня иерархии, а сами включают элементы более низкого уровня (рис.2.8 ).

**Уровень 0.** Конструктивно неделимый элемент - интегральная микросхема с радиоэлементами ее обслуживания.

**Уровень I.** На уровне I неделимые элементы объеди­няются в схемные сочетания, имеющие более сложный функ­циональный признак, образуя ячейки, модули, типовые эле­менты замены (ТЭЗ). Эти конструктивные единицы не имеют ли­цевой панели и содержат единицы и десятки микросхем. К первому структурному уровню относят печатные платы и большие гибридные интегральные схемы (БГИС), полученные путем электрического и механического объединения безкорпусных микросхем и кристаллов полупроводниковых приборов на общей плате.

**Уровень II.** Этот уровень включает в себя конструктив­ные единицы - блоки, предназначенные для механического и электри­ческого объединения элементов уровня I.

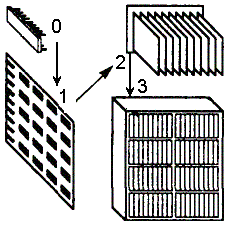


Рисунок 2.8 - Уровни конструктивной иерархии ЭВС (РЭС):

0 – уровень микросхемы; 1 – печатная плата (ТЭЗ); 2- блок; 3 – стойка или прибор

Начинают расчет с модулей низкого уровня:

, (2.71)

λ элj – интенсивность отказов элементов j-го типа; mj – число элементов данного типа; k – число типов элементов.

ЭРИ

Плата (ТЭЗ)

Блок

Стойка

1-й уровень

2-й уровень

3-й уровень

Рисунок 2.9 - Схема, показывающая иерархию конструктивных модулей

, (2.72)

где aj – число модулей (ТЭЗов) данного типа;

S – число типов различных ТЭЗов (печатных плат).

Предположим, что следующий этап – окончательный. Тогда интенсивность отказов всей системы находится как

, (2.73)

где N – число блоков в рассчитываемой электронной системе.

**Ориентировочная оценка Н** (на стадии эскизного проектирования)

Здесь учитывают:

* типы элементов и их количество;
* условия эксплуатации.

Условия эксплуатации учитывают поправочным коэффициентом K:

. (2.74)

**Уточненный расчет** проводится на этапе технического проекта [2, 8], когда принципиальные решения приняты, схема и конструкция разработаны, но еще есть возможности изменять режимы работы отдельных узлов.

Уточнение связано с введением поправочного коэффициента, учитывающего на элементном уровне режимы работы (применительно к электронным устройствам – действительную электрическую нагрузку по отношению к номинальной, а также факторы внешней среды):

, (2.75)

где λЭ0 – базовая интенсивность отказов элемента, приведенная к нормальным условиям окружающей среды\* при номинальной электрической нагрузке;

Кр – коэффициент режима, учитывающий изменение λ в зависимости от электрической нагрузки и температуры;

Кi – поправочный коэффициент, учитывающий воздействие единичного фактора внешней среды; n – число учитываемых факторов.

\**Нормальные условия окружающей среды: температура 25º±10ºС, влажность 45÷75%, атм. давление 860±1060 гектоПа.*

## Способы повышения надежности

### Классификация методов повышения надежности

Требуемый уровень надежности ЭВС закладывается на этапе проектирования, реализуется в производстве и поддерживается при эксплуатации изделий.

Все методы повышения надежности, применяемые при проектировании, конструировании и производстве ЭВС, с учетом программно-аппаратного состава этих средств, можно разделить на следующие группы:

* схемные и конструкторско-технологические;
* информационные (программно-алгоритмические);
* способы отдаления отказа на стадии эксплуатации.

Рассмотрим более подробно прежде всего первую из названных выше групп в связи с решаемой в дипломном проекте задачей расчета структурной надежности. Полноценное решение этой задачи не сводится только к нахождению численных оценок показателей надежности какой-то заданной схемы, а предполагает нахождение таких оценок для различных схемных и конструкторско-технологических решений с целью рационального достижения требуемого уровня надежности.

Известны следующие схемные и конструкторско-технологические пути повышения надежности:

* уменьшение интенсивности отказов элементов;
* рационализация схемы и конструкции в плане минимизации числа элементов устройства и повышения надежности межэлементных и межмодульных соединений;
* облегчение работы элементов при наличии неблагоприятных внешних и внутренних воздействий;
* уменьшение времени восстановления;
* резервирование, предполагающее введение избыточности (структурной, функциональной, информационной, временной).

### Способы уменьшения интенсивности отказов элементов

#### Производственно-технологические способы

Наиболее эффективными оказываются технологии, **базирующиеся на новых научных достижениях**. То есть магистральным направлением работ здесь является создание элементов, работающих на новых принципах. Именно на этом пути достигается кардинальное повышение надежности.

Интенсивности отказов элементов одинакового назначения, изготовленных по разным технологиям, могут отличаться на 2-3 десятичных порядка.

Снижение интенсивности отказов элементов, не связанное с революционными изменениями технологии, может обеспечиваться строгим соблюдением существующей технологии (или ее совершенствованием), отбраковкой худших элементов.

Важным мероприятием в направлении уменьшения интенсивностей отказов элементов являются отбраковочные испытания. В состав таких испытаний входит термоциклирование, воздействие повышенной влажности, тепловые удары и выжигание дефектов высокой температурой. Диапазон изменения температур в общем случае может составлять от 75 до 225°C, а относительной влажности — от 50 до 90%. Стандартной комбинацией температуры и влажности является 85°C и 85%. Она позволяет выявить большинство скрытых дефектов, проявляющихся на ранней стадии эксплуатации, а также спровоцировать некоторые отказы в полупроводниковых устройствах. Различные исследования показывают, что компоненты, выдержавшие такого рода испытания, работают намного надёжнее.

Это может вести к удорожанию производства (например, ввиду применения более дорогих материалов, включения в стоимость отобранных элементов стоимости забракованных элементов). То есть за надежность в любом случае надо платить.

В разрабатываемой системе выбор более качественных элементов одного и того же типа учитывается (как и в известной системе АСРН) коэффициентом приемки Кпр.

### Пути облегчения режимов работы элементов

Здесь заботятся о снижении коэффициента нагрузки *КН*, под которым понимают отношение рабочего значения нагрузочного параметра (тока, напряжения, мощности, момента силы и др.) к номинальному значению.

Например, при проектировании и конструировании систем электроавтоматики в схему «закладывается» такая нагрузка элементов, при которой *КН* ≤ 0,5 … 0,6.

Облегчение режима работы элементов достигается:

* выбором рационального варианта схемы и конструкции;
* грамотным расчетом параметров схемы и конструкции;
* принятием специальных мер по защите конструкции от неблагоприятных внешних воздействий (применением виброизоляторов, термостатов, защитных покрытий…).

Для некоторых конструкций (прежде всего силовых) важным является запас прочности. Для электродвигателей необходимо обеспечить запас величины пускового момента. Для многих систем (например, регулирования) есть понятие “запас устойчивости”.

### Защита от температурных воздействий

Высокая температура является одним из вредных факторов, воздействующих на все без исключения типы компонентов электронной схемы. Для предотвращения отказов компонентов из-за чрезмерного нагрева, тепловой анализ проекта должен быть выполнен так же тщательно, как и анализ электрических схем. Аналогично электрическим токам в цепях, в системе существуют тепловые потоки от переходов в окружающую среду [5]. Предельная температура перехода для полупроводниковых устройств общего назначения составляет около 150°C. Чем ниже рабочая температура перехода, там выше надёжность устройства. Снизить температуру переходов во время функционирования аппаратуры можно посредством специальных теплоотводов, охлаждающих вентиляторов или просто отверстий в корпусе для естественной вентиляции воздуха. Такие же меры применяются для охлаждения и других элементов схем: конденсаторов, трансформаторов, катушек, резисторов, реле и др. Дополнительными источниками тепла могут быть электрические перегрузки и ряд других факторов, например, процесс пайки или близость к нагревающимся элементам, поэтому разработчик должен полностью оценить тепловое поведение системы до начала производства.

На тепловую нагрузку электронных устройств большое влияние оказывает температура окружающей среды. Ситуация ухудшается в области высоких температур, а значит при моделировании необходимо оценивать весь диапазон температур в реальных условиях эксплуатации. Ни у кого не вызывает сомнения тот факт, что повышенная температура резко увеличивает вероятность выхода из строя полупроводниковых приборов. Это объясняется тем, что все реакции имеют физико-химическую природу и при повышенных температурах протекают быстрее. Таким образом подавляющее большинство механизмов отказов является зависимым от температуры. Косвенными причинами тепловых повреждений являются электрические перегрузки и электростатические разряды, вызывающие перегорание или плавление проводников, а также карбонизация пластиковых инкапсулирующих материалов. Для предотвращения таких повреждений необходимо эксплуатировать устройство в пределах его рабочей температуры и соответствующим образом защищать от воздействия статического электричества, электромагнитных помех и тепловых перегрузок. Последние могут стать причиной термической усталости материалов, тепловых уходов параметров, появления точек перегрева и некоторых других форм тепловых повреждений, в конечном счёте приводящих к полному или частичному отказу оборудования.

Все типы компонентов выделяют и рассеивают тепло. Некоторые из них делают это более интенсивно, например, проволочные резисторы, мощные регуляторы напряжения, транзисторы и диоды. В число способов передачи тепла входят отвод тепла через материал и проводники печатной платы, конвекция окружающего воздуха и тепловое излучение компонентов. Перегрев компонентов и участков платы во время пайки может вызвать существенные их повреждения. В общем случае, тепловые перегрузки вызывают следующие повреждения печатных плат: перегорание проводников вследствие протекания большого тока, потеря адгезии и расслаивание, обесцвечивание, деформирование и, в чрезвычайных случаях, обугливание платы. Компоненты могут противостоять умеренному тепловому воздействию в процессе монтажа и сборки, например, при кратковременном нагревании во время пайки. Тепловое же воздействие сверх установленной нормы вызовет сбои в работе устройства или его отказ.

Рекомендации по устранению проблем, связанных с перегревом:

* установка чувствительных к нагреванию элементов на теплоотводах (например, мощные диоды, транзисторы устанавливают на радиаторах);
* размещение чувствительных к нагреванию элементов (например, электролитических конденсаторов) отдельно от интенсивных источников тепла, таких как проволочные резисторы, мощные диоды, транзисторы и регуляторы напряжения;
* создание вокруг теплоотводов хороших условий для циркуляции воздуха;
* прорезывание в корпусе устройства вентиляционных отверстий (при необходимости).

Выбор теплоотводов и вентиляторов лучше произвести на начальном этапе проектирования, так как доработки, связанные с их присутствием, сокращают полезное пространство оборудования и ухудшают циркуляцию воздуха.

### Защита от механических воздействий

Все виды ЭВС подвергаются воздействию внешних механических нагрузок (вибрации, удары, ускорения, акустические шумы), которые передаются к каждой детали, входящей в конструкцию. Механические воздействия имеют место в работающей аппаратуре, если она установлена на подвижном объекте, или только при транспортировке ее в нерабочем состоянии, как в случае стационарной и некоторых видов возимой РЭА. Количество переданной энергии определяет уровень и характер изменения конструкции. Допусти­мые уровни механического изменения конструкции определяются ее проч­ностью и устойчивостью к механическим воздействиям.

Под *прочностью* конструкции понимается способность аппаратуры выполнять функции и сохранять параметры после приложения механиче­ских воздействий. *Устойчивость* конструкции - способность РЭА сохра­нять функции и параметры в процессе механических воздействий.

Откликом, или реакцией конструкции на механические воздействия называют трансформацию и преобразование энергии меха­нического возбуждения. К ним относятся механические напряжения в элементах конструкции, перемещения элементов конструкции и их соударения, деформации и разрушения конструктивных элементов, изменения свойств и параметров конструкции.

Механические воздействия могут приводить к взаимным перемещениям деталей и узлов, деформации крепежных, несущих и других элементов конструкций, их соударению. При незначительных механических воздейст­виях в элементах конструкций возникают упругие деформации, не сказывающиеся на работоспособности аппаратуры. Увеличение нагрузки приводит к появлению остаточной деформации и при определенных усло­виях разрушению конструкции. Разрушение может наступить и при нагрузках, много меньших предельных значений статической прочности мате­риалов, если конструкция окажется подверженной знакопеременным на­грузкам.

Отказы аппаратуры бывают *восстанавливаемыми* после снятия или ослабления механического воздействия (изменение параметров компонентов, возникновение электрических шумов) и *невосстанавливаемыми* (обрывы и замыкания электрических соединений, отслаивание проводников печатных плат, нару­шение элементов крепления и разрушение несущих конструкций).

На транспортируемую РЭА в процессе ее эксплуатации воздействует вибрации, ударные нагрузки и линейные ускорения. *Гармонические вибрации* характеризуются частотой, амплитудой, ускорением. *Ударные нагрузки* характеризуются числом одиночных ударов или их серией (обычно оговаривают максимальное число ударов), длительностью ударного импульса и его формой, мгновенной скоростью при ударе, перемещением соударяющихся тел. *Линейные ускорения* характеризуются ускорением, длительностью, знаком воздействия ускорения.

Возникающие при вибрациях, ударах и ускорениях перегрузки оценивают соответствующими коэффициентами. Для уменьшения воздействия вибраций и ударов аппаратуру устанавливают на амортизаторы или применяют демпфирующие материалы.

Воздействие линейных ускорений эквивалентно увеличению массы аппаратуры и при значительной длительности воздействия требует увеличения прочности конструкции. Амортизаторы от линейных перегрузок не защищают.

Как показывает опыт эксплуатации транспортируемой РЭА, наибольшее разрушающее воздействие на конструкцию оказывают вибрации. Как правило, конструкция аппарата, выдержавшая воздействие вибрационных нагрузок в определенном частотном диапазоне, выдерживает ударные нагрузки и линейные ускорения с большими значениями соответствующих параметров.

### Защита от воздействия внешних электромагнитных помех

Здесь главной мерой защиты является экранировка системы, позволяющая значительно снизить вредное воздействие внешних электромагнитных помех. Принцип работы экрана заключается в поглощении или отражении электромагнитных и электростатических полей.

Для защиты от низкочастотных электрических полей рекомендуется использовать экраны из немагнитных материалов, например, алюминия или меди, так как они лучше отражают нежелательные поля. Материалы с высокой магнитной проницаемостью, такие как железо, железоникелевые сплавы (в частности, пермаллой) используются для защиты от низкочастотных магнитных полей. Магнитная проницаемость этих материалов снижается с ростом частоты, поэтому экраны из них на высоких частотах не эффективны. Однако, в этом случае хорошо работают экраны из меди и алюминия, так как они отражают падающую волну из-за разницы импедансов среды и экрана. Материалы с высокой проводимостью, такие как медь или алюминий, являются полезными для экранировки от электрических полей, но для низкочастотных магнитных полей они неэффективны.

Важно обеспечить сплошной экран вокруг защищаемой системы. Для устранения утечек поля, все отверстия в экране должны иметь диаметр, не превышающий l/20, где l — минимальная длина волны сигналов. Другими словами, не должно быть отверстий, работающих как щелевые антенны. Неиспользуемые разъёмы также могут работать как антенны, поэтому их необходимо закрывать поглощающим материалом. Все изолирующие элементы должны иметь достаточную электрическую прочность диэлектрика, чтобы избежать пробоя под воздействием высокого электрического поля. Установка соединителей внутри углублений помогает избегать случайного контакта с объектами, заряженными статическим электричеством. Все кабели, используемые для введения сигналов внутрь экранированного корпуса, должны быть экранированными, а их оплетка должна иметь качественный круговой контакт с корпусом, что позволит избежать появления антенных эффектов.

### Рационализация схемы и конструкции

#### Минимизация числа элементов в схеме (конструкции)

Известно парадоксальное утверждение: “Идеальное техническое средство то, которого нет, а требуемая функция выполняется”. С одной стороны, есть выдающиеся примеры предельно простых и чрезвычайно надежных технических средств, подтверждающих тезис о том, что все в окружающем мире просто. С другой стороны, та же природа создала необычайно сложные системы, высокая надежность которых достигается избыточным количеством элементов. Соответственно сложны выполняемые такими системами функции.

Можно привести примеры из практики проектирования различных технических средств, когда принципиально более сложные схемы позволяют получить нужные качественные характеристики, каких нельзя достичь в более простых схемах.

Важна надежность не только самих элементов, но также их соединений. Например, соединения пайкой и сваркой имеют интенсивность отказов λ ≈ 10 - 10 1/ч, соединения в разъемах – примерно 10 - 8 1/ч. Хотя указанные интенсивности отказов невелики, число соединений обычно многократно превышает число самих электрорадиоэлементов.

#### Выбор варианта компоновки схемы

Повышению надежности разрабатываемого устройства способствует, наряду с уже указанными выше мерами, также упрощение схемы, применение более надежных соединительных элементов при минимизации их числа.

Вместе с тем при разработке ЭВС конструктор-схемотехник-технолог должен стремиться к повышению технологичности конструкции, сокращению номенклатуры деталей, максимальному использованию стандартных, унифицированных деталей и узлов (что дает большой экономический эффект за счет снижения затрат на разработку конструкции).

В настоящее время широкое применение получили такие принципы конструирования электрорадиоаппаратуры, как моносхемный, схемно-узловой, каскадно-узловой, функционально-узловой и модульный [8].

Моносхемный принцип конструирования заключается в том, что полная принципиальная схема электронного аппарата располагается на одной печатной плате. Оперативная замена вышедшего из строя элемента здесь затруднена (как из-за сложности обнаружения неисправности, так и вследствие сложности самой замены).

При схемно-узловом принципе конструирования на каждой печатной плате располагается часть полной принципиальной схемы радиотехнического устройства, имеющая четко выраженные входные и выходные характеристики. По такому принципу сконструированы настольные и бортовые ЭВМ, где различные устройства ЭВМ выполняют на одной или нескольких (небольшом числе) платах, а объединение их между собой производят с помощью коммутационной платы и проводных жгутов.

Каскадно-узловой принцип заключается в том, что принципиаль­ную схему делят на каскады, которые не могут выполнять самостоятельных функций. Электронные средства с относительно сложной и большой структурой строятся по каскадно-узловому принципу, а ЭС с более простой структурой – по схемно-узловому принципу.

Функционально-узловой принцип конструирования позволяет резко сократить время на разработку и изготовление РЭА, т.к. разрабаты­ваемая конструкция расчленяется на функционально завершенные узлы, которые могут быть отдельно сконструированы и построены, испытаны до объединения их в готовую конструкцию. Базовым элементом конструкции здесь является типовой элемент замены (ТЭЗ). Имея необходимый набор ТЭЗ, можно построить целый ряд вычислительных машин с различными техническими характеристиками.

Модульный принцип конструирования предполагает компоновку устройства из модулей, представляющих собой функционально - структур­ные ее части, выполненные в виде завершенных самостоятельных конст­рукций.

Выбор наиболее подходящего варианта конструкции должен быть сделан с учетом требуемой для изделия **надёжности**. При этом надо принимать в расчет не только показатели безотказности, но и показатели **ремонтопригодности** (в частности, среднее время восстановления, вероятность восстановления за определенное время).

Различные варианты компоновки могут быть сопоставлены на базе расчета. С помощью разрабатываемой системы предполагается оценивать на этапе проектирования показатели безотказности для различных вариантов компоновки заданной схемы.

### Резервирование

Резервирование - метод обеспечения надежности, состоящий в применении дополнительных средств для сохранения работоспособности объекта при отказе одного или нескольких его элементов или нарушении связей между ними.

Различают следующие виды резервирования:

* структурное;
* функциональное;
* временное;
* информационное.

Общее во всех этих видах резервирования - введение избыточности.

Наиболее значимое в данной работе - структурное резервирование, характерный признак которого: возможность удаления всех резервных элементов при отсутствии риска отказа основных элементов.

В зависимости от способа соединения элементов различают ***общее, раздельное и групповое резервирование***, в зависимости от того, что резервируется (весь объект в целом, отдельные элементы или группа элементов). В последнем случае структурная схема (модель) надежности имеет следующий вид (рис. 5.9).

1

2

3

4

5

6

7

8

Рисунок 5.9 - Структурная схема надежности при групповом резервировании: группа последовательно соединенных элементов 2-3-4 резервируется группой элементов 6-7-8.

В случае системы с общим структурным резервированием модель надежности называют последовательно-параллельной (рис. 5.10), а для системы с раздельным резервированием – параллельно-последовательной (рис. 5.11).

1

2

3

4

5

7

8

9

6

10

Рисунок 5.10 - Структурная схема модели надежности при общем резервировании

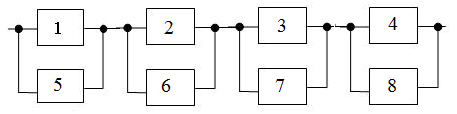


Рисунок 5.11 - Структурная схема модели надежности при раздельном резервировании

В зависимости от способа включения резерва в работу различают следующие виды резервирования.

* Постоянное, когда резервные элементы участвуют в функционировании объекта наравне с остальными (основными) элементами.
* Динамическое, связанное с перестроением структуры объекта (когда, например, при повреждении какого-либо узла в системе сообщения создается другой маршрут для транспортного потока).
* Резервирование замещением, при котором резервный элемент принимает на себя функции основного элемента только при отказе последнего. Это может происходить, например, с применением переключающего устройства К (рис. 5.12,а). В структурных схемах моделей надежности резервирование замещением условно показывается согласно рис. 5.12,б.

1

2

3

1

2

3

a)

б)

**К**

Рисунок 5.12 - Резервирование замещением: а) реализация в конкретной схеме; б) условное обозначение на структурной схеме.

* Скользящее резервирование, которое можно применять, если все основные (резервируемые) элементы системы одинаковые. Суть скользящего резервирования заключается в том, что резервные элементы не закрепляются за определенными основными элементами, а могут заменить любой из них.

Скользящему резервированию соответствует следующая структурная схема (рис. 5.13).

2

n-1

1

2

n

р1

рm

Рисунок 5.13 - Скользящее резервирование: 1 ÷ n – основные элементы системы; р1 ÷ рm – резервные элементы.

Основным параметром структурного резервирования является его кратность, определяемая как отношение числа резервных элементов к числу основных элементов.

Еще вводят понятие «резервированная подсистема» (см. групповое резервирование), то есть «связка» элементов, соединенных в схеме надежности последовательно. Тогда кратность резервирования определяется как отношение числа резервных подсистем к числу основанных подсистем.

К структурному резервированию относят также **мажоритарное резервирование** (- от слова «большинство»). Избыточность здесь вводится не в виде дублирующих структур, а в виде мажоритарных структур. В простейшем случае мажоритарное резервирование реализуется по принципу «два из трех». Например, результат в системе с тремя вычислительными машинами, решающими одну и ту же задачу, считается верным, если он совпадает с результатами двух из трех задействованных машин.

Повышение надежности в соответствии с данным принципом может достигаться и в два этапа: сначала в группе, а затем – в системе из трех групп. При этом результаты должны совпадать не менее чем в двух группах из трех, а в каждой из этих групп, содержащих по три элемента, результат берется также по принципу два из трех.

Чем больше число элементов, результаты которых должны совпадать, тем выше достоверность.

## Расчет функциональной надежности

Отказом устройства при расчете функциональной надежности считается выход значения критериального (определяющего) параметра за допустимые пределы. То есть изделие может быть в состоянии отказа, даже если ни один его элемент в структурном смысле не отказал.

Влияние изменения первичных параметров схемы на критериальный (выходной) параметр определяется выражением

.

Величина Ψвых является случайной. Её вероятностные характеристики – математическое ожидание и дисперсия – находятся в зависимости от соответствующих характеристик первичных параметров *x1, x2,…* по известным формулам:

; (\*)

. (\*\*)

Расчет функциональной надежности позволяет получить более подробную информацию о надежности схемы, но предполагает бόльшую детализацию исследования функционирования схемы и более громоздкие вычисления.

Для выполнения изделием требуемых функций необходимо, во-первых, чтобы не было отказа элементов, включенных в структурную схему расчета надежности, и, во-вторых, чтобы определяющий выполнение функции параметр не ушел за пределы установленного поля допуска.

Если вероятность невыхода определяющего параметра за пределы установленного поля допуска обозначить Рф(t), а вероятность безотказной работы, рассчитанную по структурной схеме надежности, обозначить Р(t), то вероятность выполнения изделием заданных функций рассчитывается как

Ррез(t) = Р(t)∙ Рф(t). (\*^\*)

Вероятность Рф(t) подсчитывается исходя из допустимых пределов изменения величины Ψвых критериального параметра и разброса первичных параметров входящих в изделие элементов. Эта информация позволяет найти дисперсию D[Ψвых] выходного параметра по формуле (\*\*).

Если предположить, что величина Ψвых распределена по нормальному закону, то при известной дисперсии DΨ и соответственно среднем квадратическом отклонении σΨ вероятность попадания значений критериального параметра в поле допуска (Ψн, Ψв) определяется с использованием функции Лапласа [Л] по формуле

Р(Ψн ≤ Ψ≤ Ψв) = Ф(β) - Ф(α),

где α = Ψн / σ ; β = Ψв / σ .

Например, для фильтра верхних частот (рис.) в качестве выходного критериального параметра может выступать резонансная частота ω0, связанная с параметрами схемы соотношением

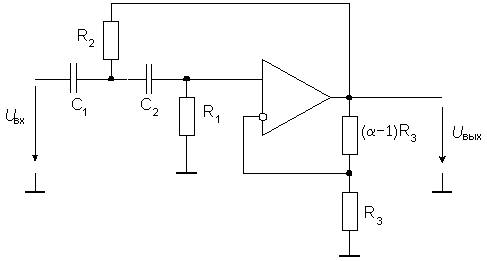


Рисунок.5.14 - Активный фильтр верхних частот второго порядка

.

.



# Разработка электронного лабораторного практикума по расчету надежности ЭВС

## Разработка алгоритмов компьютерного расчета и моделирования надежности ЭВС

Для того, чтобы рассчитать надежность системы, необходимо придерживаться следующего алгоритма. (рис. 6.1).

Данные для расчета надежности

Обращение к БД

БД содержит: (коэффициенты моделей, базовые интенсивности *λб,* таблицы для добавления новых элементов)

Создание уровня(блока), в котором будут располагаться подуровни(платы)

Запуск программы Денвер

Добавление элементов в созданные подуровни(платы)

SQL

Расчет коэффициентов режима

Расчет интенсивности отказов для каждого типа элементов

Расчет интенсивности отказов платы

Расчет ВБР

Расчет гамма-процентной наработки

Вывод результатов

Ввод условий( температура, условия эксплуатации, ввод коэффициента нагрузки, время эксплуатации, материал корпуса, рабочее напряжение, рабочая величина среднего прямого тока через диод, вид приемки)

Рисунок 6.1 - Алгоритм расчета параметров надежности.

### Разработка скриптов

Скрипт - это программа на языке PHP. Вместо термина скрипт иногда используется термин сценарий. Скрипт - это практически то же самое, что командный файл в Dos (.bat-файлы) и VMS (.com-файлы). Отличие в том, что, поскольку скрипты являются обычными программами, то вызываются они просто по имени (а не специальной командой типа "call" или "@"), могут иметь произвольное имя (со специальным расширением .php), и нет ограничений на глубину вызовов скриптов из других скриптов.

Простейший скрипт - это несколько обычных команд, которые должны выполняться последовательно.

При обычной работе в командной строке зачастую приходится писать довольно длинные команды, постоянно набирать которые заново в конце концов попросту надоедает (да и вероятность допустить опечатку тем выше, чем длиннее команда). Некоторые последовательности команд приходится повторять также все снова и снова. Вот такие команды и их последовательности и являются хорошими кандидатами для помещения в скрипты, чтобы потом их можно было вызвать по короткому имени.

Термин "скрипт" обычно подразумевает сценарий на языке PHP, являющимся интерпретатором, т.е. программа является текстовым файлом, который исполняется сразу же, не требуя предварительной компиляции в бинарный файл.

Разработанные в процессе дипломного проектирования скрипты приведены в таблице 6.1

Таблица 6.1. Скрипты и их назначение.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/g | Скрипт | Назначение | Входные данные | Выходные данные |
| 1 | Add.php | Форма ввода данных при добавлении элемента в БД | - | $nazvanie, $tip, $bazov\_int, $int\_chran, $kolichestvo, $variant |
| 2 | Add\_db.php | Добавляет элемент в БД | $nazvanie,$tip, $bazov\_int, $int\_chran, $kolichestvo, $variant | - |
| 3 | Check\_pass.php | Производит проверку введенного пароля | $password | - |
| 4 | Db.php | Соединяется с базой данных | - | - |
| 5 | Index.php | Заглавная страница | - | - |
| 6 | Login.php | Форма ввода пароля | - | $password |
| 7 | add\_circuit2.php | Добавляет плату в блок | $circuit\_name, $level\_id | - |
| 8 | add\_element2.php | Добавляет элемент на плату | $name, $level, $circuit, $element\_id, $category\_id, $position, $amount, $korpus | - |
| 9 | add\_level.php | Добавляет блок | $level | - |
| 10 | calculate.php | Форма ввода данных для расчета надежности | $level, $ circuit | $mech\_vozd, $temperature, $vysota, $priemka, $t\_expl, $pri, $level, $circuit |
| 11 | calculate2.php | Производит расчет надежности | $mech\_vozd, $temperature, $vysota, $priemka, $t\_expl, $pri, $level, $circuit | $intensity\_is, $intensity\_rezist, $intensity\_kondensator, $intensity\_diod, $intensity\_paika, $intensity\_transformator, $intensity |
| 12 | del.php | Удаляет элементы, платы и блоки | $level\_id,  $circuit\_id,  $element\_id |  |
| 13 | functions.php | Содержит функции, используемые в других скриптах | - | - |
| 14 | list\_of\_elements.php | Отображает список элементов | $type,  $group\_id | - |
| 15 | list\_of\_levels.php | Отображает список категорий | - | - |
| 16 | tree.php | Отображает дерево элементов на главной странице | $id | - |

## Разработка графического интерфейса

Разрабатываемая система расчета надежности должна обладать понятным и несложным интерфейсом. Также должна быть возможность ввода студентом различных данных, влияющих на работу схем. Немаловажным фактором является возможность добавления новых элементов в базу. С учетом всех этих требований был разработан графический интерфейс, который представлен ниже.

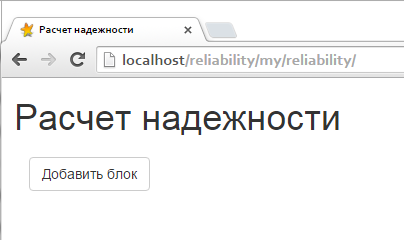


Рисунок 6.1 - Стартовая страница системы расчета надежности.

Здесь предлагается создать уровень (блок) в котором будут располагаться подуровни (платы).

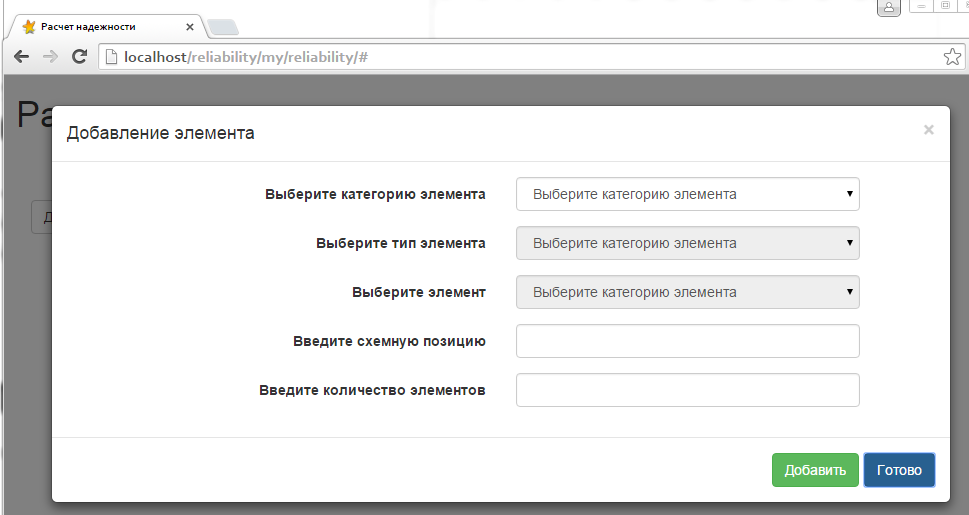


Рисунок 6.2 – Страница добавления элемента.

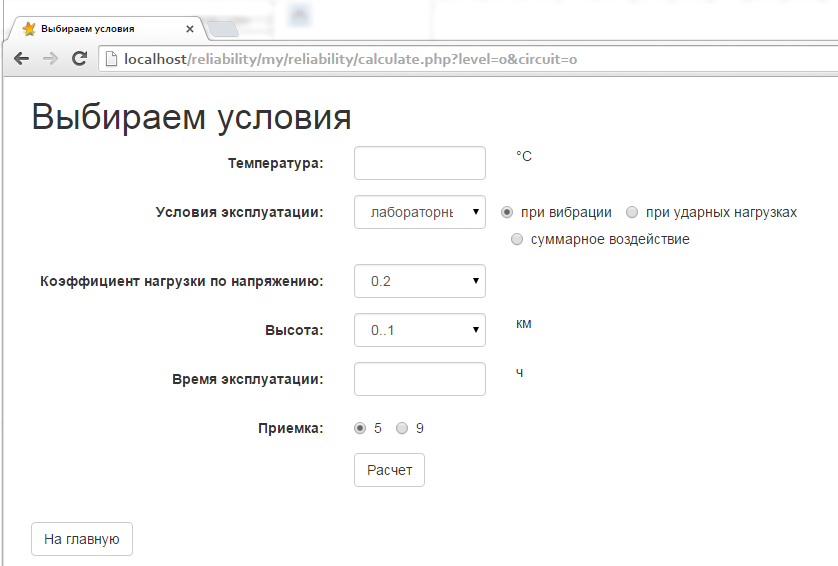


Рисунок 6.3 - Страница выбора условий.

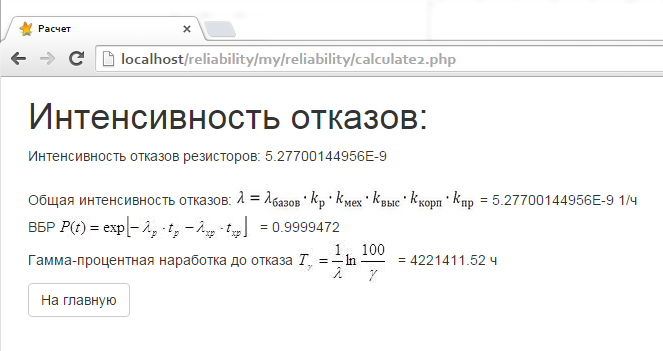


Рисунок 6.4 - Результат расчета программы.

После нажатия кнопки «Расчет» производится расчет параметров надежности системы. После этого полученные данные отображаются на странице. Для удобства понимания материала при отображении результата выводятся также формулы, по которым производится расчет.

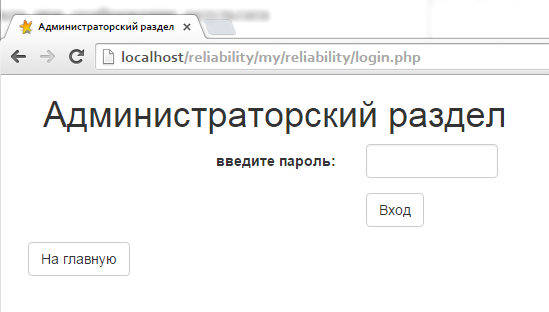


Рисунок 6.5 - Страница ввода пароля.

По адресу /login.php происходит переход на страницу ввода пароля. Пароль необходим для того, чтобы избежать «случайного» ввода элементов в базу. Если пароль неверный или поле ввода пароля пустое, то добавление элементов невозможно.

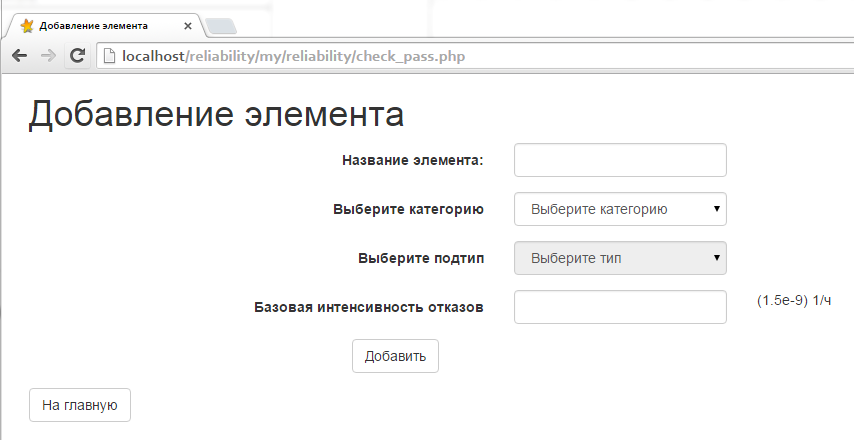


Рисунок 6.6 - Страница добавления элемента в базу.

Заполняются соответствующие поля ввода, после чего элемент добавляется в БД.

## Проверка работы программ

### Объект испытаний

Объектом испытаний является система расчета надежности ЭВС. Разработка во многом повышает усвоение студентами материала по теме надежности, является эффективным средством обучения.

### Цель испытаний

Целью испытаний является выявление возможных функциональных ошибок и сбоев, возникающих при работе данного программного продукта, обеспечение более устойчивого его функционирования в процессе эксплуатации, соответствие предъявленным требованиям, а также исследование результатов работы программы.

### Требования к программе

При тестировании программы необходимо проверить:

* правильность установки программы;
* правильность функционирования программы;
* адекватность действий всех управляемых интерфейсных элементов.

Программа должна функционировать:

* при отсутствии или неверном формате входных данных;
* при заведомо неправильных действиях пользователя.

Возможные ошибки:

* + Неверно работающие ссылки – гиперссылки, указывающие на несуществующею страницу или на страницу с неверным адресом.
  + Неверное отображение страницы программы – открывая программу в различных браузерах, можно обнаружить, что страница выглядит по-другому (сдвинуты картинки, неверные шрифты и т.д.).
  + Отсутствие каких-либо элементов (могут не отображаться картинки, текст и т. д.).
  + Неверно работающие скрипты. Могут неверно пересылаться данные, введенные пользователем.

### Технические требования

Для отладки программы в локальном режиме необходимо наличие специального ПО, работающего, как локальный web-сервер (например, Apache), и набора служб, функционирование которых необходимо для правильной работы программы (Perl, PHP, Sеndmail).

После того, как набор этого ПО будет установлен, можно приступить непосредственно к отладке. В данном случае для отладки программы использовался набор Web-разработчика Denwer - набор дистрибутивов (Apache+SSL, PHP5 в виде модуля, MySQL5, phpMyAdmin и т.д.) и браузеры I.E. 7.0, Opera, Mozilla.

### Методы испытаний

В качестве методики тестирования изберем комбинированный метод «черного» и «белого» ящика.

Тестирование методом «черного ящика» предполагает обработку системы как “непрозрачного объекта”, таким образом, знание внутренней структуры в ином виде не используется. Тестирование этим методом обычно подразумевает проверку функциональных возможностей. Синонимами понятия метода “черного ящика” являются: поведенческое тестирование, функциональное тестирование, метод непрозрачного ящика, метод закрытого ящика. При тестировании программного обеспечения методом “черного ящика” тестировщик знает только набор вводимых параметров и ожидаемые на выходе результаты, каким образом программа достигает этих результатов ему не известно.

Тестирование методом “белого ящика” предполагает обработку системы как “прозрачного объекта” и позволяет заглянуть внутрь, фокусируя внимание на использовании знаний о конкретном программном обеспечении для правильного подбора тестовых данных. Синонимами понятия метода “белого ящика” являются: структурное тестирование, метод прозрачного ящика, метод стеклянного ящика.

В отличие от метода “черного ящика” данный метод основан на использовании определенных знаний программного кода, необходимых для контроля корректности данных на выходе. Тест является правильным только в том случае, когда тестировщик знает, что конкретно должна делать программа. Таким образом, тестировщик может контролировать ожидаемый результат. Тестирование методом “белого ящика” не обрабатывает случайные ошибки, но наряду с этим весь видимый код должен быть удобочитаемым.

Применительно к данному проекту анализ внутренней структуры скриптов слишком трудоемкое занятие, поэтому имеет смысл тестировать передачу управления и данных между разделами. Это и будет элементом тестирования методом “белого ящика”. В то же время критерием соответствия программы, поставленным для нее требованиям будет корректное отображение на страницах информации, объявленной по ссылке, по которой данная информация отображается. Таким образом, мы получаем ожидаемые выходные данные и сверяем их с действительными. В этом принципе заключается составляющая метода “черного ящика” в методике тестирования.

Для проверки правильности работы программы проведем сопоставительный расчет схемы при одних и тех же условиях эксплуатации в разработанной мной системе и с помощью системы расчета надежности АСРН. Результаты показаны на рисунках 6.8 и 6.9 соответственно.

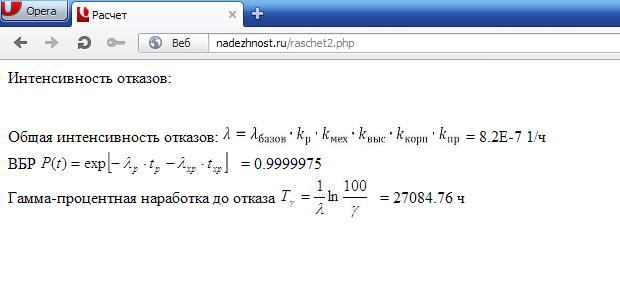


Рисунок 6.8 – Результат расчета надежности в разработанной системе.

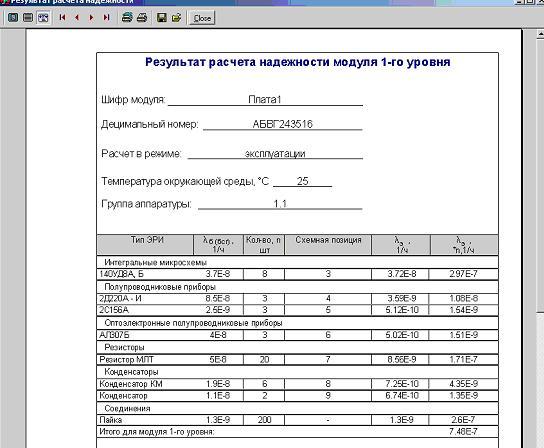


Рисунок 6.9 - Результат расчета надежности в АСРН.

Как видно из вышеприведенных рисунков, моя программа работает корректно. Допускается небольшая погрешность результата в силу нетождественности задаваемых воздействий (в АСРН они заданы комплексно для группы аппаратуры, а в нашей системе непосредственно первичными механическими параметрами).

# Заключение

В результате выполнения данного дипломного проекта, была разработана система расчета структурной надежности ЭВС для использования в лабораторном практикуме по курсу «Взаимозаменяемость и надежность».

Разработанная система состоит из двух частей - базы данных под управлением СУБД MySQL 5, и приложения, с которым работает пользователь. Программа имеет удобный для пользователя интерфейс и тем самым обеспечивает простоту восприятия и понимания, упрощая процесс проведения лабораторной работы. Внесение каких-либо изменений в базу данных может осуществляться исключительно под контролем преподавателя, так как эта функция защищена паролем.

На данный момент в приложение возможно добавлять неограниченное количество элементов для исследований, что обеспечивает более подробное изучение расчета надежности ЭВС.

Еще одной положительной чертой данного проекта являются его web-направленность. Это позволит в будущем модифицировать существующую систему расчета надежности для дальнейшего её развития и широкого применения, как в рамках корпоративной сети, так и в сети Интернет.

В ходе дипломного проектирования были освоены язык описания сценариев PHP, язык гипертекстовой разметки HTML и СУБД MySQL.

# Библиографический список

1. Острейковский В.А. Теория надежности. – М.: Высш. шк., 2008. – 463 с.
2. Чернышев А.А. Основы конструирования и надежности ЭВС. – М.: Радио и связь, 1988. – 448 с.
3. Черкесов Г.Н. Надежность аппаратно-программных комплексов. – СПб.: Питер, 2005. – 478 с.
4. Справочник «Надежность электрорадиоизделий». – МО, 2004. – 620 с.
5. Ямпурин Н.П., Баранова А.В. Основы надежности электронных средств. – М.: Изд. Центр «Академия», 2010. – 240 с.
6. Хрусталев Д. Об особенностях применения импортных компонентов в военной и специальной технике // Компоненты и технологии. 2001. №7. С. 4-5.
7. Жаднов В.В., Жаднов И.В., Измайлов А.С., Сотников В.В., Марченков К.В. Подсистема АСОНИКА-К – расчет надежности аппаратуры и ЭРИ // EDA Express. Научно-технический журнал. 2002. №5. С. 17-20.
8. Ивченко В.Г. Конструирование и технология ЭВМ. Конспект лекций. - /Таганрог: ТГРУ, Кафедра конструирования электронных средств.–2001.
9. Крушневич С.П. Микроконтроллерный сигнализатор протечки воды. – Радио, 2011, №5.
10. К. Дж. Дейт Введение в системы баз данных = Introduction to Database Systems. – 8-е изд. – М.: «Вильямс», 2006. – С. 1328.
11. Комолова Н. HTML. Самоучитель – Спб.: Питер, 2008, - С. 272.
12. Мерсер Дейв У. PHP 5.: Пер. с англ. – М.: «Вильямс», 2006. – С. 848.
13. М. Кузнецов, И. Симдянов Практика создания Web – сайтов – СПб.: «БХВ - Петербург», 2008 – С. 1264.
14. Официальный сайт «Денвер» - <http://www.denwer.ru/>
15. Документация по PHP - <http://php.net/>
16. Справочное руководство по MySQL - <http://www.mysql.ru/>
17. Документация по JavaScript - <http://javascript.ru/>
18. Документация по jQuery - <http://api.jquery.com/>
19. Документация по Bootstrap - <http://getbootstrap.com/>
20. Марше Ж. Операционные усилители и их применение. Л.: Энергия, 1974. 216 с.

Приложение 1

Схема резервирования источников питания

Принцип работы данной схемы состоит в следующем.

Когда источник питания включают в сеть, срабатывает реле К1, которое одновременно является датчиком разряда аккумулятора GB1. Через резистор R2 протекает зарядный ток 5... 10 мА. При отключении сетевого напряжения устройство получает питание от аккумулятора GB1, однако, если напряжение на аккумуляторе упадет ниже 6,5 В, реле отключится. Контакты реле разомкнут цепь питания и защитят таким образом аккумулятор от дальнейшего разряда.

Аккумуляторная батарея состоит из шести элементов Д-0,55. Ее ресурса хватает для автономной работы телефона в течение часа.  
В схеме использовано реле РЭС-64А РС4.569.724.  
Налаживают устройство подбором резистора R1, которым устанавливают напряжение отпускания реле К1. Подбором R2 устанавливают величину зарядного тока.

Режим работы устройства индицируется свечением светодиода: зеленый цвет -- работа в штатном режиме; красный — в аварийном (на батареях).  
Особенностью индикатора является то, что при работе от батареи ее разряд через подключенный основной блок питания исключен за счет использования диода в цепи затвора полевого транзистора.  
Для того чтобы при работе устройства от блока питания не происходила подпитка нагрузки от батареи, выходное напряжение блока питания должно на 0, 7... 0, 8 В превышать напряжение батареи.

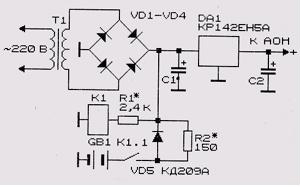


Рисунок 1 - Принципиальная резервирования источников питания.

Схема надежности резервированного источника питания (рис.5.3) включена в лабораторную работу для исследования на данном примере эффективности применения резервирования.

Вероятность отказа Qc всего резервированного источника согласно данной схеме определяется по формуле



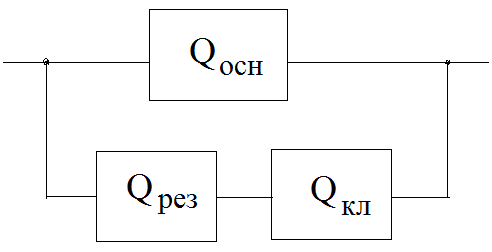


Рисунок 2 - Схема надежности резервированного источника питания:

Q0 - вероятность отказа основной ветви; Qрез , Qкл  - вероятности отказа резервного источника и переключателя соответственно.

Весьма интересным экспериментом с использованием разрабатываемой системы может быть нахождение коэффициента повышения надежности в зависимости от соотношений интенсивностей отказов основной и резервной ветвей и переключателя. Для проведения такого анализа надо предусмотреть возможность ввода интенсивностей отказов виртуальных элементов.

Приложение 2

Пример расчета функциональной надежности фильтра

Надо определить вероятность попадания параметра ***ω0*** в интервал надежной идентификации, если ***допустимые отклонения первичных параметров составляют ±2% от номинальных значений***.

Тогда отношения ΔRi / Ri , ΔCi / Ci составят 0,04.

Область допустимого изменения частоты примем 0,85 ≤ *ω0* ≤ 0,86 МГц.

Таким образом, задаем интервал [Ψн; Ψв] = [0,85; 0,86 МГц].

Для решения поставленной задачи попадания *ω0* = Ψ в указанный интервал потребуется нахождение значения дисперсии DΨ в соответствии с формулой (8):

DΨ  ≈ 4·(4·10-2)2 ·(0,855)2/ 4 = 11,7·10-4 (МГц)2.

Среднее квадратическое отклонение σΨ равно

σΨ = 3,4·10-2 МГц.

Математическое ожидание (Ψm = m) вычисляется при номинальных значениях первичных параметров.

Теперь по формулам (3), (4) остается рассчитать вероятность Рф .

z1 = (0,850 – 0,855) / 0,034 = ˗ 0,147;

z2 = (0,860 – 0,855) / 0,034 = 0,147.

Рф = Ф(z2) - Ф(z1) = 0,06 – (-0,06) = 0,12.

Приложение 3

Листинг основных скриптов

**functions.php**

<?php

function add\_level($name)

{

if (empty($\_SESSION['levels']))

{

$\_SESSION['levels'] = array();

$\_SESSION['levels'][] = $name;

}

else

{

$\_SESSION['levels'][] = $name;

}

echo "Добавлен уровень " . $name . "<br />";

}

function add\_circuit($level, $name)

{

if (empty($\_SESSION['circuits']))

{

$\_SESSION['circuits'] = array();

$\_SESSION['circuits'][$level][] = $name;

}

else

{

$\_SESSION['circuits'][$level][] = $name;

}

echo "Добавлена схема " . $name . "<br />";

}

function add\_element($level, $circuit, $id, $category\_id, $name, $position, $amount, $load\_coefficient\_diode, $load\_coefficient\_capacitor, $load\_coefficient\_resistor, $korpus)

{

if (empty($\_SESSION['positions'])) {

$\_SESSION['positions'] = array();

}

if (!in\_array($position, $\_SESSION['positions'])) {

if (empty($\_SESSION['elements'][$level][$circuit])) {

$\_SESSION['elements'][$level][$circuit] = array();

$\_SESSION['elements'][$level][$circuit][$id] = array('name' => $name, 'category\_id' => $category\_id, 'position' => $position, 'amount' => $amount, 'load\_coefficient\_diode' => $load\_coefficient\_diode, 'load\_coefficient\_capacitor' => $load\_coefficient\_capacitor, 'load\_coefficient\_resistor' => $load\_coefficient\_resistor, 'korpus' => $korpus);

$\_SESSION['positions'][] = $position;

echo "Добавлен элемент " . $name . ".";

} else {

if (!array\_key\_exists($id, $\_SESSION['elements'][$level][$circuit])) {

$\_SESSION['elements'][$level][$circuit][$id] = array('name' => $name, 'category\_id' => $category\_id, 'position' => $position, 'amount' => $amount, 'load\_coefficient\_diode' => $load\_coefficient\_diode, 'load\_coefficient\_capacitor' => $load\_coefficient\_capacitor, 'load\_coefficient\_resistor' => $load\_coefficient\_resistor, 'korpus' => $korpus);

end($\_SESSION['elements'][$level][$circuit]);

$\_SESSION['positions'][] = $position;

echo "Добавлен элемент " . $name . ".";

} else {

header('HTTP/1.1 500 Internal Server Error');

echo "Элемент " . $name . " уже присутствует на плате.";

}

}

} else {

header('HTTP/1.1 500 Error: duplicate position.');

echo "Элемент со схемной позицией " . $position . " уже присутствует на плате.";

}

}

function del($level\_id, $circuit\_id, $element\_id) //todo: replace Internal Server Error with meaningful message

{

if ( ($circuit\_id == "") && ($element\_id == "") ) //deletion of level

{

if(!isset($\_SESSION['circuits'][$level\_id]))

{

if (!isset($\_SESSION['levels'][$level\_id]))

{

header('HTTP/1.1 500 Level "' . $\_SESSION['levels'][$level\_id] . '" does not exists.');

echo "Уровень \"" . $\_SESSION['levels'][$level\_id] . "\" не существует.<br />";

}

else

{

echo "Уровень \"" . $\_SESSION['levels'][$level\_id] . "\" успешно удален.<br />";

unset($\_SESSION['levels'][$level\_id]);

}

}

else

{

header('HTTP/1.1 500 Level "' . $\_SESSION['levels'][$level\_id] . '" contain elements.');

echo "Уровень \"" . $\_SESSION['levels'][$level\_id] . "\" содержит элементы.<br />";

exit();

}

}

else if ( $element\_id == "" ) //deletion of circuit

{

if(!isset($\_SESSION['elements'][$level\_id][$circuit\_id]))

{

if (!isset($\_SESSION['circuits'][$level\_id][$circuit\_id]))

{

header('HTTP/1.1 500 Circuit "' . $\_SESSION['circuits'][$level\_id][$circuit\_id] . '" does not exists.');

echo "Схема \"" . $\_SESSION['circuits'][$level\_id][$circuit\_id] . "\" не существует.<br />";

}

else

{

echo "Схема \"" . $\_SESSION['circuits'][$level\_id][$circuit\_id] . "\" успешно удалена.<br />";

unset($\_SESSION['circuits'][$level\_id][$circuit\_id]);

if (empty( $\_SESSION['circuits'][$level\_id]))

{

unset($\_SESSION['circuits'][$level\_id]);

}

}

}

else

{

header('HTTP/1.1 500 Circuit "' . $\_SESSION['circuits'][$level\_id][$circuit\_id] . '" contain elements.');

echo "Схема \"" . $\_SESSION['circuits'][$level\_id][$circuit\_id] . "\" содержит элементы.<br />";

}

}

else //deletion of element

{

if (!isset($\_SESSION['elements'][$level\_id][$circuit\_id][$element\_id]))

{

header('HTTP/1.1 500 Element "' . $\_SESSION['elements'][$level\_id][$circuit\_id][$element\_id] . '" does not exists.');

echo "Элемент \"" . $\_SESSION['elements'][$level\_id][$circuit\_id][$element\_id] . "\" не существует<br />";

}

else

{

echo "Элемент \"" . $\_SESSION['elements'][$level\_id][$circuit\_id][$element\_id]['name'] . "\" успешно удален<br />";

$key = array\_search($\_SESSION['elements'][$level\_id][$circuit\_id][$element\_id]['position'], $\_SESSION['positions']);

unset( $\_SESSION['positions'][ $key ] );

unset( $\_SESSION['elements'][$level\_id][$circuit\_id][$element\_id] );

if (empty( $\_SESSION['elements'][$level\_id][$circuit\_id]))

{

unset($\_SESSION['elements'][$level\_id]);

}

}

}

}

**calculate2.php**

<?php session\_start(); ?>

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Расчет</title>

<link rel="stylesheet" href="css/bootstrap.css">

</head>

<body>

<script src="js/jquery-2.1.4.js"></script>

<script src="js/bootstrap.js"></script>

<script src="js/bootbox.js"></script>

<div class="container">

<h1>Интенсивность отказов:</h1>

<?php

include ("db.php");

$levels = $\_SESSION['levels'];

$circuits = $\_SESSION['circuits'];

$elements = $\_SESSION['elements'];

$intensity = (double) 0;

$mech\_vozd = $\_POST['environment'];

$temperature = (float) $\_POST['temperature'];

$vysota = (float) $\_POST['vysota'];

$priemka = (float) $\_POST['priemka'];

$t\_expl = (float) $\_POST['t\_expl'];

$pri = $\_POST['pri'];

$level = $\_POST['level'];

$circuit = $\_POST['circuit'];

$intensity\_is = 0;

$intensity\_rezist = 0;

$intensity\_kondensator = 0;

$intensity\_diod = 0;

$intensity\_paika = 0;

$intensity\_transformator = 0;

$sql = "SELECT " . $pri . " FROM `mechanicheskoe\_vozdeistvie` WHERE usloviya = '" . $mech\_vozd . "'";

$result = $mysqli->query( $sql );

$rez\_vozd = $result->fetch\_row();

$result->close();

$mech\_factory = (float) $rez\_vozd[0];

foreach ( $elements[ $level ][ $circuit ] as $element\_id => $element ) {

$sql = "SELECT intensivnost, elementtype, group\_id FROM spravochnik WHERE id = " . $element\_id;

$result2 = $mysqli->query( $sql );

$row2 = $result2->fetch\_assoc();

switch ( $row2['elementtype'] ) {

case 1:

$sql = "SELECT a, b FROM k\_r\_is\_coefficients WHERE id = " . $row2['group\_id'];

$result = $mysqli->query( $sql );

$row = $result->fetch\_assoc();

$k\_rezh\_is = $row['a'] \* exp( $row['b'] \* ( $temperature + 273 ) );

$result->close();

$intensity\_is += $row2['intensivnost'] \* $element['amount'] \* $k\_rezh\_is \* $element['korpus'] \* $mech\_factory \* $vysota \* $priemka;

break;

case 2:

$sql = "SELECT a, b, n\_t, g, n\_s, j, h FROM k\_r\_rezist\_coefficients WHERE id = " . $row2['group\_id'];

$result = $mysqli->query( $sql );

$row = $result->fetch\_assoc();

$k\_rezh\_resist = $row['a'] \* exp( $row['b'] \* pow( ( $temperature + 273 ) / $row['n\_t'], $row['g'] ) ) \* exp( pow( $element['load\_coefficient\_resistor'] / $row['n\_s'] \* pow( ( $temperature + 273 ) / 273, $row['j'] ), $row['h'] ) );

$result->close();

$intensity\_rezist += $row2['intensivnost'] \* $element['amount'] \* $k\_rezh\_resist \* $mech\_factory \* $vysota \* $priemka;

break;

case 5:

$sql = "SELECT a, b, n\_t, g, n\_s, h FROM k\_r\_kondensator\_coefficients WHERE id = " . $row2['group\_id'];

$result = $mysqli->query( $sql );

$row = $result->fetch\_assoc();

$k\_rezh\_kondensator = $row['a'] \* ( pow( ( $element['load\_coefficient\_capacitor'] ) / $row['n\_s'], $row['h'] ) + 1 ) \* exp( $row['b'] \* pow( ( $temperature + 273 ) / $row['n\_t'], $row['g'] ) );

$result->close();

$intensity\_kondensator += $row2['intensivnost'] \* $element['amount'] \* $k\_rezh\_kondensator \* $mech\_factory \* $vysota \* $priemka;

break;

case 4:

$sql = "SELECT a, n\_t, t\_m, l, delta\_t FROM k\_r\_diod\_coefficients WHERE id = " . $row2['group\_id'];

$result = $mysqli->query( $sql );

$row = $result->fetch\_assoc();

$f = 273 + $temperature + ( 175 - 175 ) + $row['delta\_t'] \* $element['load\_coefficient\_diode'] \* ( ( 175 - 25 ) / 150 );

$k\_rezh\_diod = $row['a'] \* exp( $row['n\_t'] / $f ) \* exp( pow( $f / $row['t\_m'], $row['l'] ) );

$result->close();

$intensity\_diod += $row2['intensivnost'] \* $element['amount'] \* $k\_rezh\_diod \* $mech\_factory \* $vysota \* $priemka;

break;

case 3:

case 6:

case 7:

case 8:

case 9:

case 10:

$intensity += $row2['intensivnost'] \* $element['amount'] \* pow( 1.12,$temperature ) \* $mech\_factory \* $vysota \* $priemka;

break;

case 11:

$intensity\_paika += $row2['intensivnost'] \* $element['amount'] \* $mech\_factory \* $vysota \* $priemka;

break;

case 12:

$sql = "SELECT a, g, t\_m FROM k\_r\_transformator\_coefficients WHERE id = " . $row2['group\_id'];

$result = $mysqli->query( $sql );

$row = $result->fetch\_assoc();

$t\_m = $temperature + 35.85;

$k\_rezh\_transformator = $row['a'] \* exp( pow( ( $t\_m + 273 ) / $row['t\_m'], $row['g'] ) );

$result->close();

$intensity\_transformator += $row2['intensivnost'] \* $element['amount'] \* $k\_rezh\_transformator \* $mech\_factory \* $vysota \* $priemka;

break;

default:

exit( 'element type is not defined' );

}

$result2->close();

}

$mysqli->close();

if( $intensity\_is != 0 ) {

echo "Интенсивность отказов ИМС: " . $intensity\_is . "<br>";

}

if( $intensity\_rezist != 0 ) {

echo "Интенсивность отказов резисторов: " . $intensity\_rezist . "<br>";

}

if( $intensity\_kondensator != 0 ) {

echo "Интенсивность отказов конденсаторов: " . $intensity\_kondensator . "<br>";

}

if( $intensity\_diod != 0 ) {

echo "Интенсивность отказов диодов: " . $intensity\_diod . "<br>";

}

if( $intensity\_paika != 0 ) {

echo "Интенсивность отказов пайки: " . $intensity\_paika . "<br>";

}

if( $intensity\_transformator != 0 ) {

echo "Интенсивность отказов тансформаторов: " . $intensity\_transformator . "<br>";

}

if( $intensity != 0 ) {

echo "Интенсивность отказов прочих элементов: " . $intensity . "<br>";

}

$intensity += $intensity\_is + $intensity\_rezist + $intensity\_kondensator + $intensity\_diod + $intensity\_paika + $intensity\_transformator;

$lyambda\_pict = '<img src="img/lyambda\_pict.JPG" width=240 height=30 align = middle>';

echo '<br>Общая интенсивность отказов: ', $lyambda\_pict, ' = ', $intensity, ' 1/ч';

$VBR = exp( -$intensity \* $t\_expl );

$VBR\_pict = '<img src="img/VBR\_pict.JPG" align = middle>';

echo '<br>ВБР ', $VBR\_pict, ' = ', round( $VBR, 7 );

$T\_gamma = ( 1 / $intensity ) \* log10( 100 / 95 );

$T\_gamma\_pict = '<img src="img/t\_gamma\_pict.JPG" align = middle>';

echo '<br>Гамма-процентная наработка до отказа ', $T\_gamma\_pict, ' = ', round( $T\_gamma, 2 ), ' ч <br>';

echo '<a href="index.php"><button class="btn btn-default">На главную</button></a>';

?>

</div>

</body>

</html>