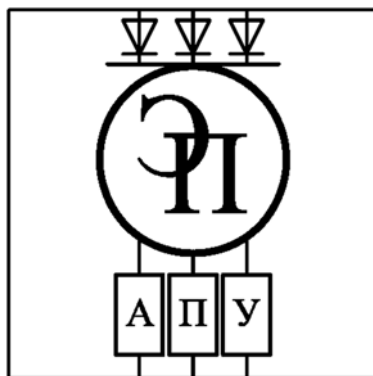


МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Электропривод и автоматизация промышленных установок»

КОНТРОЛЬ И ДИАГНОСТИКА СЛОЖНЫХ СИСТЕМ

*Методические рекомендации к лабораторным работам
для студентов направления подготовки
13.03.02 «Электроэнергетика и электротехника»
дневной формы обучения*



Могилев 2023

УДК 621.316
ББК 32.975
К65

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Электропривод и автоматизация промышленных установок» «2» мая 2023 г., протокол № 7

Составитель ст. преподаватель О. А. Капитонов

Рецензент С. В. Болотов

Методические рекомендации предназначены для студентов направления подготовки 13.03.02 «Электроэнергетика и электротехника» дневной формы обучения.

Учебное издание

КОНТРОЛЬ И ДИАГНОСТИКА СЛОЖНЫХ СИСТЕМ

Ответственный за выпуск	А. С. Коваль
Корректор	И. В. Голубцова
Компьютерная верстка	Н. П. Полевничая

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 16 экз. Заказ №

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».
Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.
Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский
университет, 2023

Содержание

Введение.....	4
1 Лабораторная работа № 1. Подготовка исходных данных и формирование структурных моделей.....	5
2 Лабораторная работа № 2. Подготовка исходных данных для построения функциональных моделей.....	7
3 Лабораторная работа № 3. Подготовка тестов и моделирование интегральных схем	10
4 Лабораторная работа № 4. Проверка правильности функционирования программ, описывающих работу интегральных схем	12
5 Лабораторная работа № 5. Моделирование логического анализатора.....	13
6 Лабораторная работа № 6. Построение схемы сигнатурного анализатора	16
7 Лабораторная работа № 7. Моделирование сигнатурного анализатора	19
8 Лабораторная работа № 8. Проектирование тестопригодных схем	22
9 Лабораторная работа № 9. Контроль и тестирование вычислительных систем	25
10 Лабораторная работа № 10. Контроль и тестирование оперативных запоминающих устройств.....	27
11 Лабораторная работа № 11. Контроль и тестирование дисковых запоминающих устройств.....	28
12 Лабораторная работа № 12. Контроль и тестирование периферийных устройств	31
13 Лабораторная работа № 13. Тестирование вычислительных сетей.....	34
Список литературы	37

Введение

Цель дисциплины заключается в изучении современных методов и алгоритмов контроля и технической диагностики – основные задачи автоматизации диагностирования; модели объектов диагностирования и неисправностей; алгоритмы диагностирования: тесты; моделирование объектов; аппаратура систем поэлементного диагностирования; сигнатурный анализ; системы автоматизации диагностирования РЭА.

Курс является необходимым для профиля подготовки «Электрооборудование автомобилей и тракторов», в научной и практической деятельности студента.

Студент, изучивший дисциплину, познает основные требования и критерии качества средств контроля и диагностики, основные алгоритмы и методы построения средств контроля и диагностики, научится проектировать средства контроля и диагностики, овладеет основными методами контроля и диагностики, методами построения тестопригодных систем.

1 Лабораторная работа № 1. Подготовка исходных данных и формирование структурных моделей

Цель работы: изучить принципы построения структурных моделей сложных электронных и электромеханических систем.

1.1 Порядок выполнения работы

1.1.1 Получить у преподавателя индивидуальное задание – словесное описание устройства либо системы.

1.1.2 Составить таблицу исходных данных для построения модели.

1.1.3 Составить перечень входных и выходных сигналов.

1.1.4 Вывести логическое уравнение, описывающее моделируемое устройство.

1.1.5 На основании исходных данных разработать структурную модель устройства.

1.1.6 Привести описание каждого элемента структурной модели.

1.1.7 Оформить отчет о выполнении лабораторной работы.

1.2 Основные теоретические положения

Структурная схема – это совокупность элементарных звеньев объекта и связей между ними, один из видов графической модели. Под элементарным звеном подразумевается часть объекта, системы управления и т. д., которая реализует элементарную функцию.

Элементарные звенья изображаются прямоугольниками, а связи между ними – сплошными линиями со стрелками, показывающими направление действия звена. Иногда в поле прямоугольника вписывают математическое выражение закона преобразования сигнала в звене, в этом случае схему называют алгоритмичной.

В схемотехнике вместе со структурной различают также принципиальную и функциональную схемы. Среди всех этих схем структурная наименее детализирована.

Она предназначена для отражения общей структуры устройства, т. е. его основных блоков, узлов, частей и главных связей между ними. Из структурной схемы должно быть понятно, зачем нужно данное устройство и что оно делает в основных режимах работы, как взаимодействуют его части. Обозначения структурной схемы могут быть достаточно свободными, хотя некоторые общепринятые правила всё же лучше выполнять.

Рассмотрим схему определения четности, описываемую следующим логическим уравнением:

$$\text{even} = a_0'a_1'a_2' + a_0a_1a_2' + a_0'a_1a_2' + a_0'a_1a_2 = (a_2 \text{ xor } a_1 \text{ xor } a_0)'.$$

Структурная схема устройства может иметь вид, представленный на рисунке 1.1.

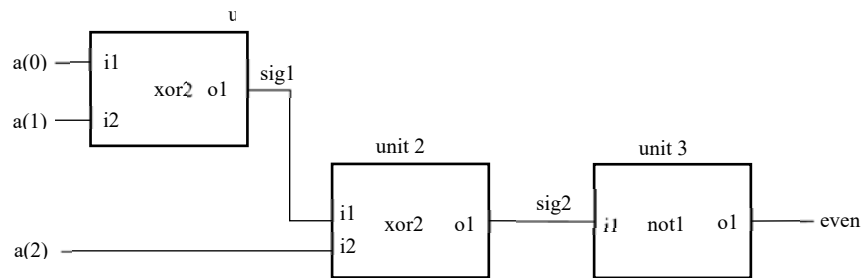


Рисунок 1.1 – Структурная схема устройства определения четности

Данная схема состоит из трех компонентов: двух компонентов, реализующих оператор xor, и одного инвертора. На VHDL это описывается с помощью оператора вставки компонента. Сначала в начальном разделе Architecture указываем, что в схеме используется дополнительный компонент, а затем после begin конкретизируем, как именно (т. е. какими сигналами) этот компонент связан с другими элементами схемы. В данном случае один компонент xor вставляется 2 раза, что означает, что создаются два одинаковых компонента, соединенных разными способами. Между ключевыми словами component и end component описывается интерфейс компонента, а с помощью оператора port map конкретизируется его связь с внешними сигналами.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 1.1.1–1.1.7.
- 4 Вывод.

Отчет оформляется на персональном компьютере в текстовом редакторе.

Контрольные вопросы

- 1 Что такое структурная схема?
- 2 Какие виды схем применяются в схемотехнике?
- 3 Поясните состав описания структурной модели на языке VHDL.
- 4 Какие исходные данные необходимы для построения структурных моделей?

2 Лабораторная работа № 2. Подготовка исходных данных для построения функциональных моделей

Цель работы: изучить принципы построения функциональных моделей сложных электронных и электромеханических систем.

2.1 Порядок выполнения работы

2.1.1 Получить у преподавателя индивидуальное задание – словесное описание устройства либо системы.

2.1.2 Составить таблицу исходных данных для построения модели.

2.1.3 Составить перечень входных и выходных сигналов.

2.1.4 Вывести логическое уравнение, описывающее моделируемое устройство.

2.1.5 На основании исходных данных разработать функциональную модель устройства.

2.1.6 Привести описание каждого элемента функциональной модели.

2.1.7 Оформить отчет о выполнении лабораторной работы.

2.2 Основные теоретические положения

Рассмотрим схему, работающую по таблице истинности (таблица 2.1) и со следующей возможной реализацией.

Таблица 2.1 – Таблица истинности устройства определения четности

a(2)	a(1)	a(0)	even
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Можно написать логическое уравнение, описывающее эту схему:

$$\text{even} = a_0'a_1'a_2' + a_0a_1a_2' + a_0'a_1a_2' + a_0'a_1a_2.$$

На рисунке 2.1 показана функциональная схема устройства определения четности.

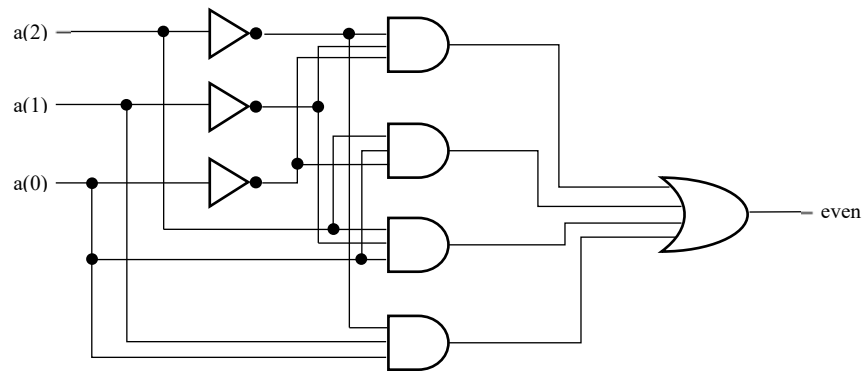


Рисунок 2.1 – Функциональная схема устройства определения четности

На VHDL данную схему можно описать следующим кодом:

```

library ieee;
use ieee.std_logic_1164.all;
entity even_detector is
    port(
        a: in std_logic_vector(2 downto 0);
        even: out std_logic);
end even_detector;

architecture sop_arch of even_detector is
    signal p1, p2, p3, p4 : std_logic;
begin
    even <= (p1 or p2) or (p3 or p4)
    p1 <= (not a(2)) and (not a(1)) and (not a(0))
    p2 <= (not a(2)) and a(1) and a(0)
    p3 <= a(2) and (not a(1)) and a(0)
    p4 <= a(2) and a(1) and (not a(0))
end sop_arch;

```

Вначале идет объявление библиотеки, затем – раздел Entity, далее – раздел Architecture. Раздел Entity описывает внешний интерфейс схемы, т. е. содержит список сигналов, с помощью которых схема связана с окружающим ее миром. Раздел Architecture описывает, как эта схема функционирует, т. е. описывает то, что у схемы внутри. Необходимо сразу сказать, что существуют как минимум два способа для описания, что происходит внутри схемы. Первый называется функциональным, т. е. описывается функциональность схемы или зависимость выходных сигналов от входных по какому-либо алгоритму. Второй способ называется структурным и предполагает, что описываемая схема состоит из более мелких компонентов, каждый из которых реализует ее часть. Тогда если функциональность этих составных компонентов уже описана, то для описания исходной схемы достаточно просто задать, как эти компоненты соединяются между собой и с внешними сигналами. Конечно, часто используется и

комбинированный подход, когда часть схемы описана функционально, а часть – структурно.

Рассматриваемый VHDL-код реализует функциональный подход к описанию схемы.

После ключевого слова `Architecture` и до слова `begin` идет подраздел объявления промежуточных (или внутренних) сигналов схемы. После слов `begin` идет место непосредственно задания связей между сигналами с помощью операторов языка. Знак «`<=`» представляет оператор параллельного назначения сигнала. Параллельность здесь выражается в том, что все эти операторы выполняются одновременно, вне зависимости от места нахождения в тексте кода. Главное то, что тут действует правило однократного назначения сигнала, т. е. нельзя дважды присвоить значение одному сигналу. Ключевые слова `and`, `or` и `not` представляют логические операторы, аналогичные операторам булевой алгебры.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 2.1.1–2.1.7.
- 4 Вывод.

Отчет оформляется на персональном компьютере в текстовом редакторе.

Контрольные вопросы

- 1 Поясните отличие функциональных моделей от структурных.
- 2 Для чего служит раздел `Entity` в описании функциональной модели на языке VHDL?
- 3 Для чего служит раздел `Architecture` в описании функциональной модели на языке VHDL?
- 4 Каким образом объявляются внутренние сигналы схемы?
- 5 Изложите методику построения функциональных моделей цифровых микросхем на языке VHDL.

3 Лабораторная работа № 3. Подготовка тестов и моделирование интегральных схем

Цель работы: изучить основные подходы к моделированию цифровых интегральных схем; освоить методы тестирования моделей.

3.1 Порядок выполнения работы

3.1.1 Получить индивидуальное задание у преподавателя, содержащее таблицу истинности и перечень входных и выходных сигналов моделируемого устройства.

3.1.2 Составить модель устройства по индивидуальному заданию на языке VHDL.

3.1.3 Подготовить тесты для разработанной модели устройства.

3.1.4 Провести моделирование и тестирование устройства.

3.2 Основные теоретические положения

Тесты необходимы на всех этапах жизненного цикла цифровой системы. Актуальность тестов обусловлена потребностью в них на этапе проектирования для анализа корректности проектов на всех этапах проектирования, т. к. методы формальной верификации не достигли уровня практического применения. На этапе производства тесты необходимы для контроля исправности готовых изделий, на этапе эксплуатации – для подтверждения исправности, работоспособности или правильного функционирования устройства. На каждом из этапов жизненного цикла изделия нужно контролировать объект с учетом моделей неисправностей, соответствующих физическим дефектам, возникающим в связи с рядом конструктивно-технологических факторов и дестабилизирующим воздействием внешней среды.

Основная цель работы – построение тестов контроля цифровых систем на верхних уровнях проектирования для своевременной оценки степени контролепригодности проекта, реализованного в разных технологических библиотеках, и сокращения сроков проектирования объекта.

Основные этапы подготовки тестов.

1 Выбор математической платформы для разработки программной модели объекта, представленного в виде описания поведения или RTL на языке VHDL.

2 Направленное построение теста контроля цифровой системы в целом в классе неисправностей, свойственных физическому объекту.

3 Построение тестов функциональных блоков на основе моделирования и символьных вычислений.

Решение данной задачи в указанной постановке позволяет, во-первых, уменьшить размерность сложной наукоемкой задачи, во-вторых, на ранних этапах в динамике анализировать качество проекта с точки зрения его контро-

лепригодности, в-третьих, оценить контролепригодность проекта применительно к разным технологическим библиотекам на ранних этапах проектирования.

Рассмотрим задачу построения теста в следующей постановке. Цифровая система представлена на RTL-уровне или уровне описания поведения. Структурная реализация системы отсутствует, но известны технологические библиотеки элементов, которые доступны на этапе его синтеза.

Необходимо построить тест контроля объекта, при этом функциональные модели рассматриваемых неисправностей должны соответствовать моделям неисправностей константного типа, свойственных реальному физическому объекту. Исходными данными для решения задачи является VHDL-описание объекта. По этому описанию необходимо построить DD потоков данных и DD потока управления, которые затем нужно перевести во множество предикатов изменения состояния системы, и затем с использованием предикатов будут определяться ограничения при активизации и распространении неисправностей.

Алгоритм построения тестов.

1 VHDL-описанию объекта строим диаграмму принятия решений потоков данных и диаграмму принятия решений потока управления (используется синтезируемое подмножество языка VHDL).

2 Выбираем очередной оператор программного кода.

3 Выбираем очередную функциональную неисправность оператора.

4 Извлекаем функциональные ограничения для очувствления данной неисправности.

5 Извлекаем функциональные ограничения для распространения эффекта неисправности к выходам системы.

6 Выполняем разрешение ограничений.

7 Если ограничения удовлетворены, то сравниваем реакции системы по выполнению программного VHDL-кода с исходным оператором и с оператором-мутантом. Если реакции различные, то фиксируем найденный тест контроля определенной группы неисправностей.

8 В противном случае переходим к пп. 3 или 2 или на выход при окончании процесса или по времени.

Иерархический подход к генерации тестов позволяет на самом высоком уровне анализировать способность системы к тестированию, уменьшает размерность задачи. При этом можно сравнивать контролепригодность объекта в зависимости от используемой технологической элементной базы. Можно генерировать тесты для разных возможных проектов реализаций (различные технологические библиотеки) и выбирать решение, которое является лучшим с точки зрения способности к тестированию.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

1 Титульный лист.

2 Цель работы.

3 Описание хода выполнения работы по пп. 3.1.1–3.1.4.

4 Вывод.

Отчет оформляется на персональном компьютере в текстовом редакторе.

Контрольные вопросы

1 На каких этапах проектирования аппаратуры необходимо применение тестов?

2 Перечислите основные этапы подготовки тестов.

3 Опишите алгоритм построения тестов по описанию системы на языке VHDL.

4 Опишите методику моделирования интегральных микросхем.

4 Лабораторная работа № 4. Проверка правильности функционирования программ, описывающих работу интегральных схем

Цель работы: изучить принципы проверки правильности функционирования программ, описывающих работу интегральных схем.

4.1 Порядок выполнения работы

4.1.1 Получить у преподавателя индивидуальное задание, включающее таблицу истинности моделируемого устройства.

4.1.2 Составить логическое уравнение, описывающее работу устройства, на основе таблицы истинности.

4.1.3 Составить программу, реализующую полученное логическое уравнение.

4.1.4 Составить модель устройства на языке VHDL.

4.1.5 Произвести моделирование устройства и сравнение результатов моделирования с результатами выполнения программы.

4.2 Основные теоретические положения

Моделирование цифровых систем является важным шагом в маршруте их разработки. Возрастание сложности проектируемых устройств заставляет разработчиков тратить все больше времени на их моделирование. Целями моделирования могут быть как исследование алгоритмов работы проектируемого устройства, так и верификация характеристик, получаемых при его аппаратной реализации. В первом случае производится моделирование на верхних уровнях абстрагирования (т. е. преимущественно на поведенческом и, возможно, RTL), а моделирование на физическом уровне призвано проверить возможность работы созданного устройства в заданных условиях эксплуатации

(т. е. проверяется возможность работы на заданной тактовой частоте, с требуемыми длительностями сигналов, в заданном температурном диапазоне и т. д.).

При моделировании используется подход, основанный на «испытательном стенде» (testbench). Моделируемое устройство (в англоязычной литературе UUT, UnitUnderTest) представляется своим синтезируемым кодом, а для проверки его поведения в различных условиях создаются описания тестовых воздействий («моделируемый код»).

Задание тестовых входных воздействий может быть выполнено на VHDL с помощью модуля типа VHDLTestBench.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 4.1.1–4.1.5.
- 4 Вывод.

Контрольные вопросы

- 1 С какой целью производится моделирование цифровых микросхем?
- 2 В каком виде может быть представлено цифровое устройство для моделирования?
- 3 Как может быть выполнено задание входных воздействий при построении модели цифрового устройства на языке VHDL?
- 4 Какими принципами необходимо руководствоваться при проверке правильности функционирования программ, описывающих работу интегральных схем?

5 Лабораторная работа № 5. Моделирование логического анализатора

Цель работы: изучить принцип работы и назначение логического анализатора.

5.1 Порядок выполнения работы

5.1.1 Получить у преподавателя индивидуальное задание в виде таблицы истинности исследуемого устройства.

5.1.2 Разработать модель устройства в среде Multisim.

5.1.3 При помощи блока Logic Analyzer в среде Multisim получить диаграммы входных и выходных сигналов моделируемого устройства.

5.1.4 Сделать выводы о правильности функционирования исследуемого устройства по результатам работы с логическим анализатором.

5.2 Основные теоретические положения

Логический анализатор (англ. Logic Analyzer) – электронный прибор, который может записывать и отображать последовательности цифровых сигналов. Он используется для тестирования и отладки цифровых электронных схем, например, при проектировании компонентов компьютеров и управляющих электронных устройств. В отличие от осциллографов, логические анализаторы имеют значительно больше входов (обычно от 16 до нескольких сотен), но при этом часто способны показывать лишь два уровня сигнала («0») и («1»), к которым иногда добавлено состояние «Z».

Существуют приборы, являющиеся гибридом логических анализаторов и осциллографов, которые позволяют синхронно записывать цифровые и аналоговые сигналы.

Логический анализатор может запускать запись по какому-либо триггеру – специфичному набору некоторых входных линий. В некоторых моделях для анализа полученных записей можно использовать собственные программы.

Крупнейшие производители логических анализаторов: Agilent Technologies, Tektronix, LeCroy.

Выбирая между осциллографом и логическим анализатором, многие инженеры отдают предпочтение осциллографу. Однако в некоторых случаях польза от осциллографа весьма невелика. Существует множество задач, где логический анализатор может оказаться куда полезней.

Осциллограф необходим:

- когда нужно увидеть небольшие выбросы на сигнале;
- когда нужна высокая точность при определении временных интервалов.

Логический анализатор необходим:

- когда нужно увидеть много сигналов одновременно;
- когда нужно представить сигналы именно так, как видит их само оборудование;
- если нужно синхронизироваться от определенной комбинации сигналов на нескольких линиях и увидеть результат.

Когда сигнал в системе пересекает пороговое значение, логический анализатор реагирует на него точно так же, как и сама логическая схема. Он распознает лишь два состояния сигнала – «ноль» или «единица». Кроме того, он может синхронизироваться по определенному сочетанию нулей и единиц исследуемых сигналов.

В общем случае необходимо использовать логический анализатор тогда, когда нужно увидеть больше сигналов, чем может показать осциллограф. Логические анализаторы очень полезны для определения временных соотношений или для исследования данных, передаваемых по шине, например, адресов, данных или управляющих сигналов на шине микропроцессора. Они могут

декодировать информацию на шинах микропроцессоров и представлять ее в осмысленном виде.

Моделирование логического анализатора в среде Multisim.

Блок (Logic analyzer) – логический анализатор. Логический анализатор позволяет записывать одновременно шестнадцать цифровых сигналов, поданных на его входы, с последующим их просмотром. На передней панели логического анализатора расположены не все его настройки, а лишь несколько кнопок.

Кнопка позволяет остановить работу анализатора. Она служит для сброса логического анализатора. При нажатии на нее он заново строит графики сигналов.

Текущий логический уровень сигналов (0 или 1), действующих на входах анализатора, отображается в круглых индикаторах, расположенных по левой стороне окна прибора. В пункте «Clocks per division» можно выбрать, сколько выборок будет приходиться на одно деление. Число выборок на деление может быть: 1, 2, 4, 8, 16, 32, 64 и 128.

На экране прибора расположены два измерительных бегунка, которые позволяют измерять временные интервалы и (или) длительность импульсов. Результаты измерения отображаются в специальном поле на панели прибора.

При нажатии кнопки на панели «Clock» выйдет окно «Clock setup» («Опции синхронизации») настройки синхронизации прибора.

В нем можно настроить работу прибора, установить режим его работы от внутреннего или внешнего источника тактовых сигналов. Рассмотрим эти настройки подробнее.

«Clock edge» – режим срабатывания на тактовые импульсы.

Он может принимать значения: «Positive» – положительный импульс; «Negative» – отрицательный импульс.

«Clock mode» – режим синхронизации анализатора.

Он также может принимать два значения:

- 1) «Internal» – тактирование от внутреннего источника импульсов;
- 2) «External» – тактирование от внешнего источника импульсов.

«Clock qualifier» – управление синхронизацией логического анализатора по входу qualifier.

«Internal clock rate» – частота следования импульсов внутреннего источника синхросигналов.

«Logic analyzer» – настройка логического анализатора. На этой панели можно задать три настройки:

- 1) «Pre-trigger samples» – размер pre-буфера;
- 2) «Post-trigger samples» – размер post-буфера;
- 3) «Threshold voltage» – пороговое напряжение.

При нажатии на кнопку на панели «Trigger» появится окно «Trigger patterns» («Свойства триггера») настройки триггера анализатора, в котором можно настроить свойства триггера.

Триггер запускает анализатор при выполнении какого-либо условия. Не стоит рассматривать этот режим подробно, так как он не работает должным

образом. Кроме задания трех констант «А», «В» и «С» здесь можно настроить еще две опции:

- 1) Trigger combinations – режим работы триггера;
- 2) Trigger qualifier – управление триггером по входу qualifier.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 5.1.1–5.1.4.
- 4 Вывод.

Отчет оформляется на персональном компьютере в текстовом редакторе.

Контрольные вопросы

- 1 Для чего предназначен логический анализатор?
- 2 В каких случаях необходимо применять логический анализатор, а в каких – цифровой осциллограф?
- 3 Каким блоком логический анализатор моделируется в Multisim?
- 4 Перечислите основные настройки модели логического анализатора в среде Multisim.
- 5 Опишите методику применения логического анализатора для тестирования работы цифровых устройств.

6 Лабораторная работа № 6. Построение схемы сигнатурного анализатора

Цель работы: изучить принципы устройства сигнатурного анализатора.

6.1 Порядок выполнения работы

6.1.1 Получить у преподавателя индивидуальное задание – принципиальную схему устройства, для которого необходимо спроектировать сигнатурный анализатор.

6.1.2 Разработать структурную схему сигнатурного анализатора.

6.1.3 Построить список сигнатур для каждого узла тестируемого устройства.

6.2 Основные теоретические положения

Поиск неисправности в микропроцессорной системе (МПС) с помощью логического анализатора требует высокой квалификации персонала. Для каждой

конкретной неисправности необходимо заново разрабатывать процедуру испытаний, настраивать прибор на новый режим работы, который, по мнению оператора, позволит обнаружить и локализовать неисправность. Поэтому логический анализатор используется в основном на этапе разработки МПС. В то же время он малопригоден на этапах производства и эксплуатации системы. Здесь большее распространение получили принципы сигнатурного анализа, которые, не требуя высокой квалификации персонала, позволяют быстро и точно отыскивать вышедший из строя компонент МПС с помощью недорогого и компактного оборудования.

Сигнатурный анализ основан на преобразовании длинных последовательностей двоичных сигналов в двоичное число, называемое сигнатурой. Измеряемые двоичные последовательности возбуждаются в контрольных точках МПС под действием специальной тестовой программы. Сигнатуры контрольных точек определяются на заведомо работоспособной системе и указываются на принципиальной схеме МПС подобно тому, как на схемах аналоговых устройств указываются осциллограммы и некоторые параметры аналоговых сигналов. Сигнатура представляется, как правило, в виде нескольких шестнадцатеричных цифр.

В циклическом избыточном контроле входной двоичный набор подается в линейную последовательностную схему, которая осуществляет деление двоичного потока на некоторый характеристический полином, и в регистре сдвига образуется остаток от деления. Обычно остаток добавляется к передаваемому двоичному потоку в качестве кода, обнаруживающего ошибки. Если же вместо добавления остатка к двоичному потоку вывести его на индикацию, это значение будет уникальным для входного двоичного набора. Имея запоминающие элементы, схема учитывает все прошлые и текущие события и может обрабатывать очень длинные потоки данных. Уникальный остаток для конкретного входа служит как бы «отпечатками пальцев» этого набора и может использоваться для его идентификации. Зависимость остатка от входного двоичного потока привела к термину «сигнатура» (т. е. подпись). Опираясь на принцип «временного окна» и используя импульсы пуска и останова и сигналы синхронизации от проверяемой системы, в узел логической схемы можно подать периодический набор. Этот набор подается на вход линейной последовательностной схемы, и при восприятии сигнала останова в регистре сдвига окажется «сигнатура» данного узла и конкретного тест-набора. Фактическое значение сигнатуры несущественно, но оно должно быть одним и тем же для данного узла, он стимулируется одним и тем же тест-набором и когда применяется те же самые сигналы пуска, останова и синхронизации.

Для получения наилучших результатов при применении способа регистра сдвига требуется последовательность максимальной длины, что приводит к широкому классу схем, называемых генераторами псевдослучайной последовательности. В 16-разрядном регистре сдвига имеется 2048 способов реализации отводов обратной связи, удовлетворяющих данному критерию. В полиноме ЦИК-16 применяется четное число входов, что приводит к группированию ошибок, а при тестировании узлов предпочтителен метод, который максимально

распределяет ошибки.

Входной двоичный поток от проверяемого узла с помощью сигналов пуска и останова подается в регистр, который синхронизируется сигналом от проверяемой системы. После прекращения двоичного потока остаток, находящийся в регистре, выводится на индикаторы в виде четырех 16-ричных символов, которые и представляют собой «сигнатуру» проверяемого узла.

Структурная схема простого сигнатурного анализатора приведена на рисунке 6.1.

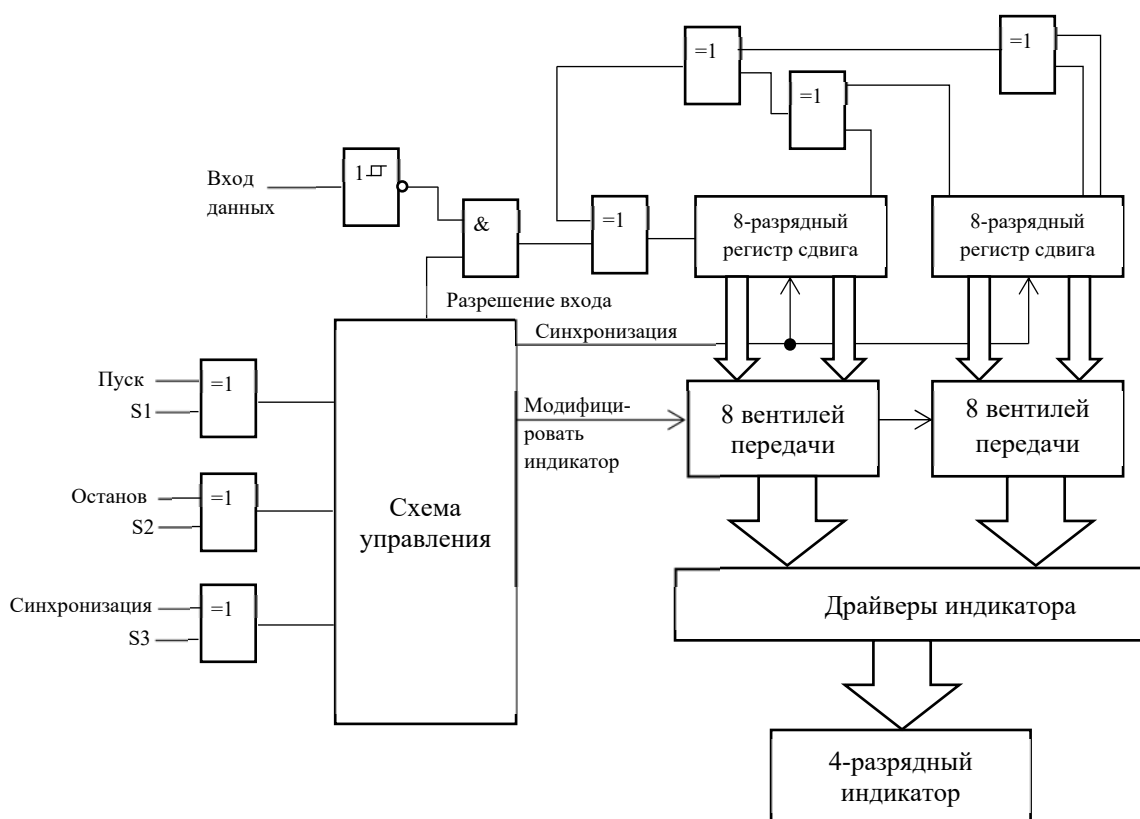


Рисунок 6.1 – Структурная схема сигнатурного анализатора

Схема иллюстрирует несколько интересных особенностей. 16-разрядный регистр сдвига реализован на двух микросхемах КР1533ИР8, а цепь обратной связи построена на двухвходовых элементах исключающего ИЛИ 1533ЛП5. Входной поток данных для улучшения формы сигналов подается на триггер Шмитта.

В приборе широкого назначения необходимо иметь возможность устанавливать для сигналов пуска, останова и синхронизации любой активный фронт – нарастающий или спадающий. Например, в одном тесте нужен сигнал пуска с активным нарастающим фронтом, а в другом тесте он должен запускать операции спадающим фронтом. Возможность задания активного фронта сигнала обеспечивается входными элементами исключающего ИЛИ, через которые сигналы проходят в схему управления. В случае сигнала пуска переключатель S1 подсоединяется к земле или к источнику питания Vcc. Когда ключ замкнут на землю, на выходе элемента исключающего ИЛИ повторяется входной сигнал

пуска. Если же S1 подключен к Vcc, сигнал на выходе представляет собой инверсию входного сигнала. Предположим, что для инициирования схемы управления всегда требуется нарастающий фронт сигнала. Тогда для удовлетворения этого требования с помощью S1 можно выбрать либо нарастающий фронт сигнала (S1 подключен к земле), либо спадающий фронт сигнала (S1 подключен к Vcc).

Остаток, сформированный в регистре сдвига, при восприятии сигнала останова индицируется как «сигнатура» проверяемого узла.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 6.1.1–6.1.3.
- 4 Вывод.

Контрольные вопросы

- 1 Для чего предназначен сигнатурный анализатор?
- 2 Перечислите основные принципы построения сигнатурных анализаторов.
- 3 Опишите методику получения сигнатур элементов электронных устройств.
- 4 Опишите методику построения тест-наборов.
- 5 Поясните структурную схему логического анализатора.

7 Лабораторная работа № 7. Моделирование сигнатурного анализатора

Цель работы: освоить приемы работы с сигнатурным анализатором.

7.1 Порядок выполнения работы

7.1.1 В соответствии с индивидуальным заданием, полученным для лабораторной работы № 6, создать модель тестируемого устройства в среде Multisim.

7.1.2 По структурной схеме сигнатурного анализатора, разработанной в лабораторной работе № 6, создать модель сигнатурного анализатора в среде Multisim.

7.1.3 Произвести моделирование работы тестируемого устройства с сигнатурным анализатором.

7.1.4 Сравнить полученные в результате моделирования сигнатуры узлов тестируемого устройства с составленными в ходе выполнения лабораторной работы № 6.

7.1.5 Сделать выводы о корректности работы тестируемого устройства.

7.2 Основные теоретические положения

Поиск неисправности с применением сигнатурного анализа заключается в определении возможной области с неисправностью и проверки сигнатур до обнаружения неправильной сигнатуры. Пользуясь принципиальной схемой и таблицами сигнатур, неисправность прослеживают «назад» до получения правильной сигнатуры. Неисправность существует в той части схемы, которая находится между последней неправильной и первой правильной сигнатурами.

Тестирование постоянного запоминающего устройства (ПЗУ) в режиме свободного счета.

Хотя главное назначение свободного счета заключается в проверке системного ядра, он применим и для контроля ПЗУ. В режиме свободного счета на шине адреса периодически формируются все двоичные наборы. Подаваемая в центральный процессор (ЦП) холостая команда заставляет его выполнять операции считывания по каждому адресу. ПЗУ содержит только фиксированные команды, которые в режиме свободного счета последовательно выдаются на шину данных. Используя линию разрешения кристалла ПЗУ для сигналов пуска и останова, а управляющую линию READ для сигнала синхронизации анализатора, можно проверить содержимое любого системного ПЗУ.

Анализатор регистрирует только данные, относящиеся к проверяемому ПЗУ, хотя процессор сканирует все адресное пространство.

Аналогичный тест для микросхем оперативного запоминающего устройства (ОЗУ) применять нельзя, т. к. их содержимое не фиксировано и для проверки работы ОЗУ разработаны другие тесты. Поскольку ЦП выполняет только операции считывания из памяти, невозможно проверить каналы ВВ, особенно в том случае, если ВВ отображен на адресное пространство памяти, в режиме свободного счета проверить можно, т. к. ЦП считает обращение к ним операциями считывания из памяти. При этом необходимо управлять входами в порты, для чего обычно применяется тестовый прибор с возможностью задания известных двоичных наборов. Для проверки выходных портов потребуется операция записи, которой в режиме свободного счета нет.

Для проверки тех частей системы, которые недоступны режиму свободного счета, необходимо написать и выполнить специальные программы. Каждая из них предназначена для проверки одной части системы, например входного порта, и обычно состоит всего из нескольких строк ассемблерного кода. Такие тест-программы сигнатурного анализа обычно помещаются в ПЗУ, которое находится в системе, но при нормальной работе не используется. Тест-ПЗУ приводится в действие либо переключением линии OE от первого системного ПЗУ с последующим сбросом системы, либо вводом в процессор команды RESTART и размещением тест-ПЗУ по адресу рестарта. Набор тест-программ обычно организован как цикл, который периодически выполняется при включении тест-ПЗУ в работу. В начале полного тест-цикла предусматривается формирование импульса на линии сигнала пуска сигнатурного анализатора, а в конце тест-цикла формируется импульс на линии сигнала останова анализатора. Часто для обоих сигналов пуска и останова используется одна и та

же линия, что устраняет необходимость формирования отдельного сигнала останова.

Принцип теста заключается в сдвиге состояния логической «1» по всем восьми линиям ВВ выходного порта. Таким образом, до перехода к следующему тесту будет произведено восемь операций записи в выходной порт. При использовании линии A15 для сигналов пуска и останова, а линии WRITE для входа синхронизации берутся сигнатуры с каждой выходной линии и сравниваются с задокументированными значениями. Аналогичный тест можно написать для проверки каждой входной линии входного порта, но вначале необходимо задать их известные состояния. Обычно для этого к входным линиям подключается тест-прибор, позволяющий устанавливать состояния входных линий.

Ограничения сигнатурного анализатора.

Сигнатурный анализатор регистрирует только те события, которые синхронны с сигналом синхронизации, используемым для получения сигнатур от узлов системы. Микропроцессор является синхронным конечным автоматом и при сигнатурном анализе проверяется большинство его частей. Однако некоторые события происходят асинхронно с системной синхронизацией и их нельзя проконтролировать с помощью сигнатурного анализа. Последовательная линия связи в стандарте RS232C обычно работает асинхронно с системной синхронизацией, что препятствует применению сигнатурного анализа. В этом случае на вход синхронизации анализатора можно подать сигнал синхронизации универсального асинхронного передатчика и получить сигнатуры последовательной линии связи. Однако компьютер необходимо запрограммировать на выдачу в линию периодического сигнала, чтобы в тесте участвовала известная периодическая двоичная последовательность.

Компьютер в шинной структуре можно считать замкнутой системой с обратной связью, в которой по шине адреса осуществляется запрос данных, а реакция выводится с шины данных. Проблема в такой структуре заключается в том, что отказ распространяется по всей петле, вызывая получение кажущихся плохих сигнатур в тех местах, где отказ отсутствует. Сигнатурный анализ не показывает, в каком месте временного окна впервые возникает отказ, что можно было бы использовать для локализации неисправного компонента. Плохая сигнатура по истечении временного окна не показывает «историю» прослеживания отказа. Поэтому, когда много микросхем осуществляют вывод на общую шину и появляется неправильная сигнатура, сигнатурный анализ не может указать, в какой именно микросхеме возник отказ.

Усовершенствованной формой сигнатурного анализа явился трассовый анализ, который локализует отказ в окне пуска-останова. Сигнатурный трассовый анализатор программируется (обычно вручную) на последовательность промежуточных сигнатур и их моменты появления во временном окне. При работе прибора первое несовпадение измеренных данных с хранимой информацией заставляет сигнатурный трассовый анализатор остановиться и индцировать момент появления ошибки. После этого отказ можно подозревать в той микросхеме, которая работала в момент его появления. Правильные данные для сигнатурного трассового анализа, как и сигнатуры, необходимо получить от

заведомо исправной системы и задокументировать их. При этом сигнатуры разделяются на короткие секции с соответствующим увеличением объема документируемой информации. Однако дополнительная документация потребуется только для тех частей системы, которые невозможно адекватно проверить, пользуясь обычным сигнатурным анализом.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 7.1.1–7.1.5.
- 4 Вывод.

Контрольные вопросы

- 1 Опишите методику поиска неисправностей с помощью сигнатурного анализатора.
- 2 Опишите методику тестирования ПЗУ в режиме свободного счета при помощи сигнатурного анализатора.
- 3 Поясните методику тестирования ОЗУ при помощи сигнатурного анализатора.
- 4 Перечислите ограничения сигнатурного анализатора.

8 Лабораторная работа № 8. Проектирование тестопригодных схем

Цель работы: освоить принципы проектирования тестопригодных схем.

8.1 Порядок выполнения работы

8.1.1 Получить у преподавателя индивидуальное задание – словесное описание устройства.

8.1.2 Разработать по словесному описанию структурную и функциональную схемы устройства. При разработке следует руководствоваться принципами проектирования ремонтпригодных, контролепригодных и тестопригодных электронных средств.

8.1.3 Провести анализ тестопригодности разработанной схемы.

8.2 Основные теоретические положения

Анализ тестопригодности цифровой электронной схемы позволяет решать следующие задачи:

- выявлять потенциально трудные для тестирования узлы схемы (узлы с

низкой управляемостью и наблюдаемостью) и модернизировать их таким образом, чтобы повысить их тестопригодность с сохранением прежних логических функций;

- оптимизировать выбор испытательных точек при использовании внутри-схемного контроля (с перемещаемыми зондами);

- принимать решение о модернизации всего узла (блока) с целью улучшения его тестопригодности.

Улучшить показатели тестопригодности той или иной схемы можно несколькими способами.

Самым простым из них является вынесение на выходной разъём электронной платы выходов тех узлов, которые имеют самые низкие показатели управляемости и наблюдаемости.

Другим способом является изменение логической структуры наиболее труднотестируемых узлов с сохранением той же логической функции.

Но самым эффективным способом является применение элементной базы более высокого уровня интеграции, что позволяет плохо тестируемые узлы погружать внутрь интегральной схемы более высокой степени интеграции. Особенно удобны в этом отношении программируемые логические интегральные схемы (ПЛИСы).

В настоящее время некоторые САПР электронных средств комплектуют подсистемой тестопригодного проектирования – DFT (Design for Test). Эта подсистема позволяет вычислять оценки управляемости, наблюдаемости и тестопригодности для разрабатываемых схем. Для проведения такого анализа в подсистему вводится полное описание проверяемой схемы на языке данной САПР.

Анализ проводится в три этапа: сначала выполняется анализ управляемости каждого логического узла схемы, затем – анализ наблюдаемости каждого узла и всей схемы в целом и, наконец, определяются показатели тестопригодности для каждого логического узла и всей схемы в целом. Поскольку логические узлы могут быть и комбинационными, и последовательными (с памятью), то на каждом этапе определяются два вида показателей: комбинационные и последовательные. Определение комбинационных показателей было рассмотрено ранее. Показатели же последовательного вида определяют, какое число последовательных состояний должен пройти данный логический узел (или вся электронная схема), чтобы в данной точке (на определённом логическом выходе) можно было бы получить требуемое состояние (показатель управляемости) или чтобы можно было бы на данном выходе наблюдать требуемое состояние (показатели наблюдаемости).

Чтобы к потребителю поступала продукция надлежащего качества, необходимо обеспечить в том числе и ее ремонтпригодность, контролепригодность и тестопригодность.

Кратко выделим принципы проектирования ремонтпригодных, контролепригодных и тестопригодных электронных средств:

- использование современных схмотехнических и других инженерных решений;

- широкое применение (по возможности) типовых схем повышения уровня унификации и стандартизации разработки;
- включение в конструкцию изделия встроенных систем контроля, в том числе автоматических;
- дублирование жизненно важных органов изделия (особенно для изделий, поломка которых критична);
- повышение технического уровня производства, комплексная механизация и автоматизация;
- применение современных систем автоматизированного проектирования (САПР), систем автоматизированного контроля;
- применение современной элементной базы;
- вынесение на выходной разъём электронной платы выходов тех узлов, которые имеют самые низкие показатели управляемости и наблюдаемости;
- изменение логической структуры наиболее трудно тестируемых узлов с сохранением той же логической функции;
- применение элементной базы более высокого уровня интеграции, что позволяет плохо тестируемые узлы погружать внутрь интегральной схемы более высокой степени интеграции. Особенно удобны в этом отношении программируемые логические интегральные схемы (ПЛИСы).

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 8.1.1–8.1.3.
- 4 Вывод.

Контрольные вопросы

- 1 Для чего необходим анализ тестопригодности цифровых электронных схем?
- 2 Перечислите задачи, которые решает анализ тестопригодности электронных схем.
- 3 Какими способами можно улучшить показатели тестопригодности?
- 4 Перечислите основные этапы анализа тестопригодности цифровых электронных схем.

9 Лабораторная работа № 9. Контроль и тестирование вычислительных систем

Цель работы: изучить основные принципы контроля и тестирования вычислительных систем.

9.1 Порядок выполнения работы

9.1.1 Запустить программу тестирования ПК NovaBench.

9.1.2 Запустить нагрузочное тестирование, нажав кнопку Start Benchmark Tests.

9.1.3 По завершении тестирования NovaBench отображает сравнительный показатель NovaBench – чем он выше, тем лучше. Программа также показывает результаты каждого отдельного теста.

9.1.4 Сделать выводы о стабильности и производительности тестируемой вычислительной системы.

9.2 Основные теоретические положения

Тестирование производительности в инженерии программного обеспечения – тестирование, которое проводится с целью определения, как быстро работает вычислительная система или её часть под определённой нагрузкой. Также может служить для проверки и подтверждения других атрибутов качества системы, таких как масштабируемость, надёжность и потребление ресурсов.

Тестирование производительности – это одна из сфер деятельности развивающейся в области информатики инженерии производительности, которая стремится учитывать производительность на стадии моделирования и проектирования системы, перед началом основной стадии кодирования.

В тестировании производительности различают следующие направления:

- нагрузочное;
- стресс;
- тестирование стабильности;
- конфигурационное.

Возможны два подхода к тестированию производительности программного обеспечения:

1) в терминах рабочей нагрузки: программное обеспечение подвергается тестированию в ситуациях, соответствующих различным сценариям использования;

2) в рамках бета-тестирования, когда система испытывается реальными конечными пользователями.

Нагрузочное тестирование.

Нагрузочное тестирование – простейшая форма тестирования производительности. Оно обычно проводится для того, чтобы оценить поведение приложения под заданной ожидаемой нагрузкой. Этой нагрузкой может быть,

например, ожидаемое количество одновременно работающих пользователей приложения, совершающих заданное число транзакций за интервал времени. Такой тип тестирования обычно позволяет получить время отклика всех самых важных бизнес-транзакций. В случае наблюдения за базой данных, сервером приложений, сетью и т. д. этот тип тестирования может также идентифицировать некоторые узкие места приложения.

Стресс-тестирование.

Стресс-тестирование обычно используется для понимания пределов пропускной способности приложения. Этот тип тестирования проводится для определения надёжности системы во время экстремальных или диспропорциональных нагрузок и отвечает на вопросы о достаточной производительности системы в случае, если текущая нагрузка сильно превысит ожидаемый максимум.

Тестирование стабильности.

Тестирование стабильности проводится с целью убедиться в том, что приложение выдерживает ожидаемую нагрузку в течение длительного времени. При проведении этого вида тестирования осуществляется наблюдение за потреблением приложения памяти, чтобы выявить потенциальные утечки. Кроме того, такое тестирование выявляет деградацию производительности, выражающуюся в снижении скорости обработки информации и/или увеличении времени ответа приложения после продолжительной работы по сравнению с началом теста.

Конфигурационное тестирование.

Конфигурационное тестирование – ещё один из видов традиционного тестирования производительности. В этом случае вместо того, чтобы тестировать производительность системы с точки зрения подаваемой нагрузки, тестируется эффект влияния на производительность изменений в конфигурации. Хорошим примером такого тестирования могут быть эксперименты с различными методами балансировки нагрузки. Конфигурационное тестирование также может быть совмещено с нагрузочным, стресс-тестированием или тестированием стабильности.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 9.1.1–9.1.4.
- 4 Вывод.

Контрольные вопросы

- 1 С какой целью проводится тестирование производительности вычислительных систем?
- 2 В чем заключается нагрузочное тестирование?
- 3 В чем заключается стресс-тестирование?

4 В чем заключается тестирование стабильности?

5 В чем заключается конфигурационное тестирование?

10 Лабораторная работа № 10. Контроль и тестирование оперативных запоминающих устройств

Цель работы: изучить основные методы контроля и тестирования оперативных запоминающих устройств.

10.1 Порядок выполнения работы

10.1.1 Запустить средство диагностики памяти Windows.

10.1.2 Задать параметры проверки. Тип проверки – широкий, использование кеша – включить, количество проходов – три.

10.1.3 Записать результаты тестирования.

10.1.4 Сделать выводы по результатам тестирования об исправности оперативной памяти.

10.2 Основные теоретические положения

Проверка работоспособности оперативной памяти может потребоваться в тех случаях, когда есть подозрения на то, что синие экраны смерти Windows, странности в работе компьютера и Windows вызваны именно проблемами с RAM.

Симптомы ошибок оперативной памяти.

Существует значительное число индикаторов сбоев RAM. Среди наиболее часто встречающихся признаков можно выделить следующие:

- частое появление BSOD – синего экрана смерти Windows. Не всегда связано с оперативной памятью (чаще – с работой драйверов устройств), но ее ошибки могут быть одной из причин;

- вылеты во время интенсивного использования RAM – в играх, 3D-приложениях, видеомонтаже и работе с графикой, архивации и распаковке архивов (например, ошибка unarc.dll часто бывает из-за проблемной памяти);

- искаженное изображение на мониторе – чаще является признаком проблем видеокарты, но в некоторых случаях вызвано ошибками RAM;

- компьютер не загружается и бесконечно пищит. Можно найти таблицы звуковых сигналов для данной материнской платы и узнать, соответствует ли слышимый писк сбоям памяти.

Средство диагностики памяти Windows 10, 8 и Windows 7.

Средство проверки (диагностики) памяти – встроенная утилита Windows, позволяющая проверить RAM на ошибки. Для её запуска необходимо нажать клавиши Win+R на клавиатуре, ввести mdsched и нажать Enter (либо использовать поиск Windows 10 и 8, начав вводить слово «проверка»).

После запуска утилиты будет предложено перезагрузить компьютер для выполнения проверки памяти на ошибки.

В процессе сканирования можно нажать клавишу F1 для изменения параметров проверки, в частности, можно изменить следующие настройки:

- тип проверки – базовый, обычный или широкий;
- использование кэша (вкл., выкл.);
- количество проходов теста.

По завершении процесса проверки компьютер перезагрузится, а после входа в систему отобразит результаты проверки.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 10.1.1–10.1.4.
- 4 Вывод.

Контрольные вопросы

- 1 Когда возникает необходимость в тестировании ОЗУ?
- 2 Перечислите наиболее распространенные программные утилиты для тестирования ОЗУ.
- 3 Опишите порядок работы со средством диагностики памяти Windows.
- 4 Перечислите основные признаки сбоев в работе ОЗУ.

11 Лабораторная работа № 11. Контроль и тестирование дисковых запоминающих устройств

Цель работы: изучить основные методы контроля и тестирования дисковых запоминающих устройств.

11.1 Порядок выполнения работы

11.1.1 Открыть папку «Мой компьютер» (для Windows 8: нажать комбинацию клавиш Windows + E).

11.1.2 Щёлкнуть правой кнопкой мыши по диску, который необходимо проверить, и выбрать пункт «Свойства».

11.1.3 Открыть вкладку «Сервис» и нажать кнопку «Выполнить проверку».

11.1.4 Отметить галочками напротив оба параметра и нажать кнопку «Запуск».

11.1.5 Будет предложено выполнить проверку в ходе следующей загрузки компьютера. Нажать кнопку «Расписание проверки диска».

11.1.6 Перезагрузить компьютер и дождаться завершения проверки.

11.1.7 По результатам проверки сделать вывод о состоянии жесткого диска.

11.2 Основные теоретические положения

Жёсткий диск (винчестер, HDD) – один из самых ненадёжных узлов как компьютера, так и ноутбука. Связано это с тем, что жесткие диски имеют как механические подвижные узлы, так и электронные платы управления. Любой механический узел вносит большую ненадёжность, связанную с тем, что механические узлы сильно страдают от вибраций, трясок, ударов.

Винчестер состоит из гермоблока и платы питания и управления.

Гермоблок (HDA) герметичен, чтобы пыль не попала внутрь, однако вакуума внутри нет. Гермоблок заполняется воздухом с давлением в одну атмосферу для создания «воздушной подушки» между дисками и головками.

Требования к чистоте воздуха чрезвычайно велики – не должно быть пылинок размером более 0,3 мкм. В связи с этим винчестеры ни в коем случае нельзя разбирать в обычных условиях.

Современные накопители на жестких дисках имеют функцию автоматической парковки магнитных головок, т. е. при отключении питания головки устанавливаются на определенный, чаще всего последний, цилиндр. Эта парковочная секция называется Landing Zone, или сокращенно L-Zone.

При работе электрический импульс, поступая на катушку, вызывает ее смещение относительно жестко закрепленного постоянного магнита, перемещая при этом блок головок на заданный угол. Предусилитель-коммутатор обеспечивает усиление сигнала, подаваемого с магнитных головок при чтении на плату управления.

Проблемы с чтением данных могут возникнуть как из-за повреждения самих головок, так и выгорания микросхемы предусилителя или же повреждения самих магнитных дисков (царапины или запылы на ней).

Типичные неисправности жестких дисков.

Все неисправности винчестеров можно подразделить на программные и аппаратные. Аппаратные требуют ремонта диска для того, чтобы вытащить данные с него. Чаще всего жесткие диски выходят из строя из-за падений, ударов и других механических повреждений. При этом часто повреждаются или сами головки, или царапается поверхность дисков, или и то и другое.

В этом случае необходимо заменить блок головок точно с такого же донора. Иногда выходит из строя одна из головок и диск частично доступен. Тогда возможно вычитывание данных по одной или нескольким головам без замены блока магнитных головок (БМГ).

Предусилитель-коммутатор повреждается при бросках напряжения или при повреждении платы управления. В таком случае она может выйти из строя из-за нестабильности питания винчестера, плохих блоков питания компьютера. Плата управления ремонтируется или чаще заменяется с донора с перепрошивкой диска.

Служебная область восстанавливается только в технологическом режиме на

специальных стендах. Особенно сложны случаи повреждения транслятора – таблиц адресации секторов и дефектов диска. Клин шпинделя встречается не так часто, но характерен для некоторых HDD Seagate и ноутбучных Toshiba. В этом случае производится ремонт двигателя, после чего с диска копируются данные. Заклинивание двигателя можно устранить и пересадкой блинов на исправный накопитель.

Встречаются случаи залипания голов БМГ на поверхности дисков. Тогда производится вскрытие гермозоны и отвод голов в зону парковки. После этого делается копия на исправный винчестер на комплексе Data Extractor. Часто при залипании царапается поверхность дисков, так что не нужно пытаться включать диск после его выхода из строя, т.к. в таком случае не гарантируется 100-процентное восстановление данных HDD. Программные неисправности чаще всего связаны с повреждением секторов дисков (появление bad-секторов), что приводит, в свою очередь, к нарушению файловой системы. Данные пользователя оказываются недоступными. Процент успешного восстановления в каждом случае индивидуален и зависит от количества поврежденных блоков.

Диагностика состояния жесткого диска.

Если диск не работоспособен, не определяется в BIOS, издает нехарактерные звуки, то во всех этих случаях необходимо отдать его специалистам, не пытаться восстановить его самостоятельно: каждое включение (особенно при проблемах с головами), а тем более вскрытие усугубит ситуацию.

Если же винчестер корректно определяется, но загрузить операционную систему не получается или же данные недоступны, то в первую очередь необходимо проверить состояние жесткого диска с помощью диагностики SMART-атрибутов.

Технология SMART (Self Monitoring Analysing and Reporting Technology) – это технология оценки состояния жёсткого диска с помощью процедуры встроенной самодиагностики диска. SMART производит наблюдение за основными характеристиками накопителя. Каждой характеристике на заводе присваивается изначальное значение – 100 или 254. Также для каждого атрибута задано пороговое значение – минимальное, по достижении которого диск считается поврежденным.

По мере работы накопителя и его износа эти значения (current value) все время уменьшаются, пока не достигнут порогового значения (threshold value). Задача – проверить критические параметры на предмет их достижения порога.

Критическими параметрами здоровья диска считаются: Raw Read Error Rate, Spin-Up Time, Reallocated Sectors Count, Seek Error Rate, Spin-Up Retry Count, Reallocation Event Count, Current Pending Sector Count, Uncorrectable Sector Count. Ухудшение атрибутов:

- 1) Raw Read Error Rate и Seek Error Rate говорит об износе магнитной поверхности или головок БМГ;
- 2) Spin-Up Time и Spin-Up Retry Count – о проблемах в механике диска;
- 3) Reallocated Sectors Count, Reallocation Event Count и Current Pending Sector Count, Uncorrectable Sector Count – о наличии bad-секторов – уже перемещенных bad-секторов и готовых к перемещению.

Проверить SMART-атрибуты винчестера можно с помощью специальных утилит – как платных, так и бесплатных. Рекомендуются программы HDDScan, HDD Health, Victoria, MHDD, HD Tune.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 11.1.1–11.1.7.
- 4 Вывод.

Контрольные вопросы

- 1 Поясните внутреннее устройство жесткого диска.
- 2 Перечислите типичные неисправности жестких дисков.
- 3 Перечислите основные методы диагностики состояния жестких дисков.
- 4 Поясните назначение технологии SMART.
- 5 Перечислите основные параметры здоровья жесткого диска, представленные в системе SMART.

12 Лабораторная работа № 12. Контроль и тестирование периферийных устройств

Цель работы: изучить основные методы контроля и тестирования периферийных устройств.

12.1 Порядок выполнения работы

- 12.1.1 Выполнить тестирование максимальной скорости печати принтера.
- 12.1.2 Выполнить тестирование скорости печати многостраничного документа с иллюстрациями.
- 12.1.3 Выполнить тестирование качества воспроизведения шрифтов.
- 12.1.4 Выполнить печать универсальной тестовой полосы.
- 12.1.5 Выполнить печать тестовой страницы.
- 12.1.6 По результатам тестирования сделать вывод об исправности и качестве работы принтера.

12.2 Основные теоретические положения

Тестирование принтера осуществляется по следующим критериям.

Максимальная скорость печати.

Данный тест показывает максимальную скорость печатающего механизма. На принтер посылается несколько копий одностраничного PDF-файла со стандартным 5-процентным заполнением листа. Количество запущенных на печать копий соответствует заявленному производителем числу копий в минуту +1 копия. Замер времени начинается сразу после выдачи принтером первой копии. Таким образом, исключается зависимость результата от мощности компьютера, типа файла, драйверов, интерфейса принтера и скорости обработки задания самим принтером.

Тест запускается на всех типах принтеров. Результат – количество стр./мин.

Печать многостраничного документа с иллюстрациями.

В отличие от предыдущего теста, моделирующего тиражирование одностраничного документа, данный тест показывает время печати несложного 20-страничного документа в формате PDF, содержащего небольшие растровые иллюстрации в формате RGB и текст, набранный 10 кеглем. Печать производится из программы Adobe Reader 6.0; время замеряется от момента запуска документа в печать. Поскольку объем и тип документа нестандартны, а также замеряется время обработки задания драйвером, то результат отражает относительную производительность конкретной связки «приложение – компьютер – принтер». Конфигурация тестового стенда приведена в приложении.

Тест запускается на всех типах принтеров. Результат – время в минутах, секундах.

Оценка качества воспроизведения шрифтов.

Данный тест позволяет оценить качество печати текстового материала, набранного шрифтом с засечками и шрифтом без засечек (Times New Roman и Arial соответственно). Кегли шрифтов уменьшаются от 72 до 2 pt.

В отсутствие поддержки принтером языка PostScript на печать подается PDF-файл из программы Adobe Reader 6.0. Однако если производителем заявлена поддержка стандарта PostScript, тестовая страница выводится на принтер непосредственно из пакета верстки Adobe InDesign CS.

Тест запускается на всех типах принтеров. Результат – минимальный читаемый кегль шрифтов (размер в пунктах), качество печати криволинейных и наклонных элементов литер. Под читаемостью шрифта подразумевается сохранение целостности основных элементов литер (отсутствие «заплывших» элементов или разрывов литер).

Печать универсальной тестовой полосы.

Специальная тестовая страница сформирована таким образом, чтобы показать в явном виде возможности принтера. Для оценки доступны следующие элементы:

- шкала нейтральных плотностей. Диапазон 0 %...10 % с шагом 1 %, 10 %...90 % с шагом 10 % и 90 %...100 % с шагом 1 %;
- цветовые растяжки CMYKRGB 100 %...0 %;

– шкалы, показывающие минимальную и максимальную воспроизводимые плотности красок СМΥК. Минимальные плотности – это цифры от 1 до 8, окрашенные в соответствующие им оттенки, на белом фоне (0 %). Максимальные плотности – цифры от 92 до 99 с соответствующей окраской на фоне 100 %. Каждой краске соответствует своя пара шкал;

- шкала, состоящая из 100-процентных плашек цветов RBGYMCK;
- цветовая растяжка RYGBM, созданная средствами Adobe Photoshop CS;
- текст, набранный кеглями от 4 до 8 pt (5...9 pt для декоративного шрифта)

в начертаниях: декоративном, с засечками, без засечек. Приведены варианты: нормальный (черный текст на белом фоне), выворотка (белый текст на черном фоне) и цветной (текст разных цветов на насыщенном зеленом фоне);

– шкалы, позволяющие замерить максимальную воспроизводимую линейную. Представляют собой наборы концентрических окружностей. Доступны значения в линиях на дюйм: 60...80 с шагом в 5 линий на дюйм, 80...170 с шагом в 10 линий на дюйм;

– набор шкал, позволяющих оценить совмещение красок и точность калибровки принтера.

Представляют собой наборы продольных, поперечных, наклонных, волнистых и пересекающихся линий разных цветов.

Все тестовые элементы расположены на сплошном сером фоне, не имеющем полей, что позволяет определить реальное поле печати принтера.

Для принтеров, не имеющих поддержки PostScript, на печать подается документ в формате PDF из Adobe Reader 6.0. Для PostScript-принтеров печать выполняется непосредственно из пакета верстки Adobe InDesign CS. Отпечатанная полоса сканируется, после чего оцениваются все параметры.

Тест запускается на всех типах принтеров. Результат – фактические значения всех параметров.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 12.1.1–12.1.6.
- 4 Вывод.

Контрольные вопросы

- 1 По каким основным критериям осуществляется тестирование печатающих устройств?
- 2 Поясните методику определения максимальной скорости печати.
- 3 Поясните методику оценки качества шрифтов.
- 4 Поясните методику тестирования с использованием универсальной тестовой полосы.

13 Лабораторная работа № 13. Тестирование вычислительных сетей

Цель работы: изучить основные методы контроля и тестирования вычислительных сетей.

13.1 Порядок выполнения работы

13.1.1 Определить топологию локальной вычислительной сети в компьютерном классе.

13.1.2 Составить список ip-адресов компьютеров в вычислительной сети, отдельно выделив ip-адрес сервера.

13.1.3 Проверить доступность сервера с каждого рабочего места при помощи утилиты ping.

13.1.4 Проверить доступность компьютеров в локальной сети друг для друга при помощи утилиты ping.

13.1.5 Определить скорость локальной вычислительной сети при помощи утилиты LAN Speed Test.

13.2 Основные теоретические положения

Процесс монтажа кабельной системы локальной сети с учетом разного рода особенностей изначально не может гарантировать 100-процентную работоспособность всех сегментов сети.

Связано это с использованием достаточно большого количества механических операций, автоматизировать которые невозможно по ряду причин. Именно поэтому монтаж локальной сети всегда сопровождается постоянным процессом тестирования. Когда же монтаж локальной сети полностью завершен, осуществляется полная проверка работоспособности сети с подготовкой соответствующей технической документации.

Подобная процедура – стандартный подход в случае, когда проектированием и монтажом локальной сети (или, как ее называют в этом случае, СКС (структурированная кабельная система)) занимается фирма-подрядчик.

Существуют определенные методы проверки работоспособности сети, которые позволяют устранить возникшую неисправность как на этапе монтажа локальной сети, так и после его завершения.

Использование тестеров.

Наиболее объективным и простым способом тестирования всех особенностей локальной сети является использование разного рода тестеров. Они позволяют максимально автоматизировать и упростить процесс тестирования, поэтому, если есть такая возможность, желательно применять именно этот способ.

Существуют разные варианты тестеров, отличающихся методами тестирования, количеством разнообразных тестов, а также способом выдачи результатов.

От этих функций напрямую зависит стоимость тестирующего оборудования. На рынке существует достаточно много тестирующего оборудования от разных производителей, стоимость которого колеблется в широком диапазоне: от 50 до 20000 долл. По понятным причинам использовать дорогостоящее оборудование может себе позволить лишь серьезная фирма, предоставляющая профессиональные услуги по монтажу СКС. На практике при тестировании большей части создаваемых локальных сетей с 30...50 компьютерами применяются простейшие тестеры, которые дают возможность только проверять состояние кабельного сегмента, чего в 90 % случаев вполне достаточно.

Различают два основных вида тестеров: для тестирования физических линий и сетевые анализаторы.

Тестеры для тестирования физических линий получили наибольшее распространение благодаря своей цене. Такой тестер способен определять неисправность кабельного сегмента на физическом уровне, вплоть до определения места обрыва проводников. Кроме того, он может, например, протестировать волновое сопротивление линии или измерить скорость передачи данных, что позволяет определить используемый сетевой стандарт или соответствие определенному стандарту. Покупку такого тестера может позволить себе даже небольшая фирма, что даст возможность быстро определять и устранять неисправность в процессе эксплуатации локальной сети.

Сетевые анализаторы – дорогостоящее оборудование, приобретение которого могут себе позволить только сетевые интеграторы. С помощью такого сетевого анализатора можно не только исследовать характеристики кабельной структуры, но и получить полную информацию о процессе, происходящем при прохождении сигнала от любого узла к любому узлу, с определением проблемных сегментов и «узких мест». Кроме того, можно даже прогнозировать состояние сети в ближайшем будущем и пути решения или предотвращения будущих проблем.

Хороший тестер позволяет оценить максимальное количество параметров кабеля, для чего в комплекте с тестером часто идут разного рода переходники и вспомогательные инструменты. Например, используя соответствующие переходники, можно производить тестирование как коаксиальных сегментов, так и сегментов кабеля «витая пара». Что касается оптоволоконных линий, то оборудование для их тестирования имеет более сложную конструкцию и часто ориентировано только на тестирование оптоволокна.

Тестирование кабельного сегмента происходит разными способами, которые зависят от наличия доступа к кабелю. Один из способов заключается в следующем: конец обжатого кабеля подключается к разъему на тестере, а на второй конец устанавливается специальная заглушка. В результате тестер может проверить сопротивление каждого проводника, а также соответствие их подключению одному из стандартов. Использование данных о сопротивлении позволяет определить технические характеристики кабеля, а также выяснить расстояние до точки обрыва.

Использование программного способа.

Когда возможности приобретения тестера нет, что часто происходит при монтаже офисной или «домашней» сети, целостность и качество кабельного сегмента можно проверить и программным путем, используя, например, системную утилиту ping.

Принцип работы этого метода крайне прост и сводится к тому, чтобы попытаться передать через кабель любые данные.

Например, чтобы проверить сегмент коаксиального пути, необходимо соединить им два компьютера и установить на них терминаторы. Далее нужно настроить IP-адресацию каждого компьютера, присвоив одному, например, IP-адрес 192.168.2.1, а второму – 192.168.2.2 с маской подсети 255.255.255.0. Затем на компьютере с адресом 192.168.2.1 следует запустить командную строку, в которой ввести следующую команду: ping 192.168.2.2.

Если в результате выполнения этой команды последует ответ «Ответ от 192.168.2.2: число байт=32 время < 1мс TTL=64», значит, кабельный сегмент физически цел.

Если же в результате выполнения команды на экране появится надпись «Превышен интервал ожидания для запроса», это будет свидетельствовать о том, что кабель имеет обрыв или коннекторы обжаты неправильно.

Подобным образом можно производить тестирование любого кабеля, в том числе и кабеля «витая пара». В случае с кабелем «витая пара» подобного рода подключение возможно только для варианта кроссовер. Если же необходимо протестировать работоспособность кабеля типа патч-корд, его нужно подключать к центральному узлу, например коммутатору, а в паре с ним использовать заведомо рабочий кабель, который подключен ко второму компьютеру.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

- 1 Титульный лист.
- 2 Цель работы.
- 3 Описание хода выполнения работы по пп. 13.1.1–13.1.5.
- 4 Вывод.

Контрольные вопросы

- 1 Перечислите основные виды топологий локальных сетей.
- 2 Поясните методику проверки локальной сети с использованием тестеров.
- 3 Какие виды тестеров для проверки локальных сетей существуют?
- 4 Поясните методику проверки локальной сети программным способом.
- 5 Перечислите основные утилиты Windows, необходимые для настройки параметров и тестирования локальных вычислительных сетей.

Список литературы

- 1 **Набоких, В. А.** Датчики автомобильных электронных систем управления и диагностического оборудования: учебное пособие / В. А. Набоких. – Москва: ИНФРА-М, 2022. – 239 с.
- 2 **Овсянников, Е. М.** Электрический привод : учебник / Е. М. Овсянников. – Москва: ФОРУМ, 2019. – 224 с.
- 3 **Коваленко, Н. А.** Научные исследования и решение инженерных задач в сфере автомобильного транспорта : учебное пособие / Н. А. Коваленко. – Москва: ИНФРА-М, 2023. – 271 с.
- 4 **Долгин, В. П.** Надежность технических систем: учебное пособие / В. П. Долгин, А. О. Харченко. – Москва : Вузовский учебник ; ИНФРА-М, 2020. – 167 с.