

**Министерство образования и науки Российской Федерации  
Государственное бюджетное образовательное учреждение высшего профессионального  
образования  
«Томский государственный университет систем управления и радиоэлектроники»**

**УТВЕРЖДАЮ**

Зав.кафедрой ЭС

\_\_\_\_\_  
" \_\_\_\_ " \_\_\_\_\_ Н.Е.Родионов  
2014 г.

Вводится в действие с " \_\_\_\_ " \_\_\_\_\_ 20 г.

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

**ПО ПРОВЕДЕНИЮ ПРАКТИЧЕСКИХ РАБОТ**

**по дисциплине**

**Основы автоматизированного проектирования**

Составлена кафедрой

Электронных систем

Для студентов, обучающихся  
по направлениям подготовки 221000.62 «Мехатроника и робототехника»  
220000.62 «Инноватика»

Форма обучения

очная

Составитель Ст. преподаватель кафедры  
Электронных систем

Нестеренко П.Г.

"01" мая 2014 г

Томск 2014 г.

## **Введение**

Дисциплина играет важную роль в формировании профессиональных знаний в области компьютерных технологий управления в робототехнике и мехатронике. И нацелена на подготовку студентов к освоению средств и методов повышения производительности инженерного труда и научной деятельности, сокращению сроков и улучшения качества разработок, исследованию в области проектирования и совершенствования аппаратных и программных средств мехатронных устройств, созданию и применению алгоритмического, аппаратного и программного обеспечения систем автоматизированного проектирования систем различной степени сложности.

Полученные знания и навыки могут быть использованы при разработке ПО микропроцессорных систем, в системах и комплексах автоматизированного управления.

Практические работы выполняются студентами очной формы обучения индивидуально под контролем со стороны преподавателя. Все консультации осуществляются преподавателем. Число студентов, одновременно присутствующих на занятии не должно превышать 12 человек. Если в списочном составе группы студентов больше 12, то группа должна быть разделена на подгруппы численностью от 6 до 12 человек в каждой.

Для выполнения практических заданий целесообразно в учебном расписании выделять 4 академических часа подряд, без больших перерывов. Расписание также должно предусматривать раздельное проведение занятий у подгрупп, если группа была разделена.

Перед началом занятий студенты должны изучить инструкцию по охране труда. Преподаватель должен убедиться в знании инструкции, задавая студенту вопросы по ее содержанию, после чего сделать соответствующую запись в журнале охраны труда.

Во время проведения практических занятий в аудитории (лаборатории) студентам запрещается передавать друг другу файлы и другие материалы, являющиеся результатом выполнения заданий.

Студент имеет право:

- Выходить из аудитории (лаборатории) не спрашивая разрешения у преподавателя.
- Самостоятельно распределять аудиторное время, определяя необходимость перерыва или непрерывной работы.
- Просить консультации у преподавателя, если он в текущий момент не распределяет задания, не принимает выполненные работы и не консультирует другого студента.

Преподаватель, давая консультацию студенту, указывает раздел технической документации или методической литературы, в которой имеется ответ на вопрос студента. Если необходимые сведения в документации и литературе отсутствуют, то преподаватель должен дать устные пояснения или продемонстрировать практические действия, приводящие к требуемому результату, с последующей отменой для повторения студентом.

Самостоятельная работа студентов над практическими занятиями осуществляется в той же аудитории (лаборатории), где проводятся практические занятия. Преподаватель должен согласовать со студентами расписание самостоятельной работы - не менее 2 астрономических часов в неделю. В указанное время по учебному расписанию студентов и в аудитории (лаборатории) не должны проводиться другие занятия. Преподаватель должен обеспечить доступ студентов в аудиторию (лабораторию) в указанные часы. Необходимость самостоятельной работы определяет студент.

Консультации, выдача практических заданий и прием результатов выполнения осуществляется только во время аудиторных занятий. Задания выполняются последовательно. Правильное выполнение некоторых заданий возможно только, если студент корректно выполнил предыдущие задания. Поэтому приступать к следующему заданию студент может, только сдав преподавателю результат выполнения предыдущего.

### **Техническое обеспечение практических работ**

Для выполнения лабораторных работ студенту предоставляется индивидуальное рабочее место, в состав которого входят:

- персональный компьютер с операционной системой WindowsXP;
- текстовый редактор Microsoft Office 2003;
- программа для чтения документов в формате .pdf;
- программное обеспечение для проектирования сложных программно-технических систем IAR VisualState;
- программное обеспечение для организации совместной работы над проектом OpenSVN;
- программное обеспечение для организации совместной работы над проектом OpenSVN;
- программное обеспечение для организации системы отслеживания ошибок и управления проектом BugTraking.

Размещение и освещенность рабочих мест в учебной аудитории (лаборатории) должно удовлетворять действующим требованиям СанПиН.

### **Прием результатов выполнения практических заданий**

Результаты выполнения практических заданий демонстрируются преподавателю. Во время приема выполненной работы преподаватель вправе:

- Требовать у студента демонстрации выполненного задания в виде таблиц, мнемосхем, рисунком или диаграмм.
- Самостоятельно производить манипуляции с программным обеспечением, не изменяя его конфигурацию.
- Требовать у студента пояснений, относящихся к способам реализации функций задания.

Задание считается выполненным и принимается преподавателем только в том случае, если реализован весь функционал, предусмотренный заданием. Если какие то функции, предусмотренные заданием, не работают, или работают неверно, то результат выполнения подлежит доработке. Студент должен работать внимательно и аккуратно. Подлежат исправлению замеченные преподавателем недочеты:

- грамматические ошибки в надписях, наименованиях элементов;
- небольшие неточности в описаниях, структурах, схемах;

Результаты выполнения заданий сохраняются преподавателем в электронном виде и хранятся в течение двух лет.

До конца семестра студент должен сдать результаты выполнения всех практических заданий, предусмотренным настоящими указаниями. В противном случае студенты к сдаче экзамена (зачета) не допускаются.

### **Задания для практических занятий**

1. Проектирование. Типовая логическая схема проектирования – 4 часа. Задание в приложении А.
2. Современные средства САПР создания программно-аппаратных комплексов – 4 часа. Задание в приложении Б;
3. Метод разработки программных средств на основе графа состояний. Системы прототипирования ПО на основе графа состояний. Система Visual STATE. Основные понятие и особенности. Создание проекта. Отладка. Тестирование Унифицированный процесс разработки программного обеспечения. – 12 часов. Задание в приложении В;
4. Системы контроля версий. Методы управления проектированием программных средств– 8 часов. Задание в приложении Г;

## **5. Библиографический список**

6. Технология разработки программного обеспечения : Учебное пособие / В. Т. Калайда, В. В. Романенко ; Федеральное агентство по образованию, Томский государственный университет систем управления и радиоэлектроники. - Томск : ТУСУР, 2007. - 237[1] с.
7. САПР систем логического управления : научное издание / В. А. Горбатов, А. В. Крылов, Н. В. Федоров ; ред. В. А. Горбатов. - М. : Энергоатомиздат, 1988. - 232 с. : ил. - Б. ц. (19 экз.)
8. Теоретические основы САПР : Учебник для вузов / В. П. Корячко, В. М. Курейчик, И. П. Норенков ; рец. Е. Л. Глориозов. - М. : Энергоатомиздат, 1987. - 398, [2] с. : ил. - Библиогр.: с. 392. - Алф. указ.: с. 393-397. - 01.20 р.
9. Микропроцессорные системы : Учебное пособие для вузов / Е. К. Александров [и др.]; ред. Д. В. Пузанков. - СПб. : Политехника, 2002. - 934, [2] с.

## **Приложение А**

### **Проектирование. Типовая логическая схема проектирования.**

#### **Цель работы:**

Получить представление о процессе проектирования, основных особенностях, жизненные циклы проектирования.

#### **Задание считается выполненным, если студент:**

1. Может дать объяснение следующим понятиям:
  - a. Проектирование
  - b. Алгоритм проектирования
  - c. Результат проектирования
  - d. Проектное решение
  - e. Типовое проектное решение
  - f. Цель процесса проектирования
  - g. Проектная процедура
  - h. проектной операцией
2. Оперировать понятиями - Системы проектирования.
3. Может графически представить:
  - a. Структуре системы неавтоматизированного проектирования
  - b. Структуре системы автоматизированного проектирования
4. Может объяснить Стадии и этапы проектирования.

#### **Вопросы для самоконтроля:**

1. Проектирование
2. Алгоритм проектирования
3. Результат проектирования
4. Проектное решение
5. Типовое проектное решение
6. Цель процесса проектирования
7. Проектная процедура
8. проектной операцией
9. Предпроектное исследование
10. Разработка технического задания (ТЗ).
11. Эскизное проектирование
12. Техническое проектирование
13. Подходы к конструированию на основе компьютерных технологий

## **Приложение Б**

### **Системы автоматизированного проектирования. Структура и разновидности САПР**

#### **Цель работы:**

Получить представление о САПР. САПР-это комплекс программно-технических средств автоматизированного проектирования, взаимосвязанных с необходимыми подразделениями проектной организации или коллективом специалистов, выполняющих проектирование.

**Задание считается выполненным, если студент владеет следующими знаниями:**

1. Определение САПР.
2. Разновидности САПР
  - a. По приложениям
  - b. По целевому назначению
  - c. По масштабам
  - d. По характеру базовой подсистемы
3. САПР как сложная система
4. Функциональные подсистемы
5. Виды обеспечения САПР - виды, назначение
6. Общая модель объекта проектирования

#### **Вопросы для самоконтроля:**

1. Определение САПР.
2. САПР как сложная система
3. Виды обеспечения САПР:
  - a. математическое;
  - b. программное;
  - c. информационное;
  - d. лингвистическое;
  - e. техническое;
  - f. инструктивно-методическое;
  - g. организационно-технологическое;

## Приложение В

### Метод разработки программных средств на основе графа состояний.

#### Цель работы:

Получения навыков разработки ПО в соответствии с моделью графа состояний.

#### Задание считается выполненным, если студент:

1. Графически продемонстрировал граф переходов на основании словесной формулировки алгоритмов управления.
2. Графическое представление графа переходов соответствует приведенному на рисунке 1.

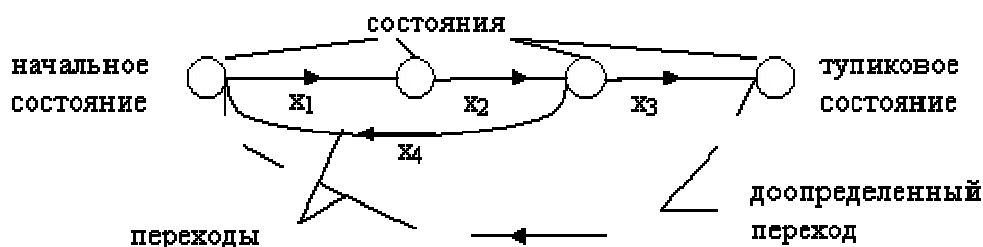


Рисунок 1.

3. В приведенном графе выделено следующее
  - Определенно начальное состояние объекта, из которого начинается процесс управления, и для него вводится начальное состояние на графе переходов, обозначаемое кружком "0".
  - Определяется очередность состояний объекта и необходимое для этого изменение состояний ИМ. Каждый переход из начального состояния в последующие изображается направленной линией, связывающей изображение этих состояний. Стрелка, указывающая направление перехода, обычно изображается в средней части линии. Над стрелкой указывается логическое условие, при выполнении которого осуществляется данный переход. Если из данного состояния возможны несколько переходов, то все они изображаются направленными переходами с соответствующими условиями переходов, при этом все условия должны быть взаимоисключающими, т.е. не должно выполняться более одного условия в данный момент времени.
  - Граф не содержит избыточного числа состояний и переходов, каждое новое состояние вводится только тогда, когда аналогичного состояния на графе не вводилось.
  - Построение графа переходов продолжается до тех пор, пока все последовательности состояний не образуют замкнутые циклы или подграфы.



- В графе нет тупиковых состояний, из которых нет переходов в другие состояния, свидетельствует, как правило, либо об ошибках построения графа переходов, либо о не полноте или ошибочности исходных данных, приведенных в словесной формулировке алгоритма (в этом случае необходимо доопределить и замкнуть граф соответствующим переходом).

**Вопросы для самоконтроля:**

1. Сформулируйте правила разметки графа состояний для автоматов Мура и Мили.
2. Сформулируйте правила построения графов переходов для автоматов Мура и Мили.
3. Сформулируйте правила построения таблиц переходов-выходов для автоматов Мура и Мили.
4. Сформулируйте правила построения таблицы переходов-выходов автоматов Мура и Мили.

## **10. Приложение Г Системы контроля версий.**

### **Цель работы:**

Получение и закрепления навыков работы с системами хранения и контроля версий.

**Задание считается выполненным, если студент может выполнить следующее:**

1. Показать файлы из CVS с определённой меткой.
2. Показать файлы из CVS на определённую дату.
3. Поместить новый проект в CVS.
4. Посмотреть локальные изменения файла по сравнению с последней версией файла из хранилища CVS.
5. Продемонстрировать конфликт и его разрешение в CVS.
6. Установить метку в CVS.
7. Создать ветку в CVS.

### **Вопросы для самоконтроля:**

5. Как идентифицируется хранилище в CVS? В Subversion?
6. Какой информации достаточно, чтобы взять конкретный файл в CVS?
7. Рассказать об операциях checkout, update, commit
8. Что такое метки и ветки?
9. Обзор клиентов cvs?
10. Как поместить новый проект в CVS?
11. Как поместить новый проект в Subversion?
12. Что такое "конфликт"?