



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ Информатика и системы управления (ИУ)

КАФЕДРА «Информационная безопасность» (ИУ8)

МОДУЛЬ ВНЕШНИХ УСТРОЙСТВ ПРОГРАММНОГО СИМУЛЯТОРА PDP-11

**Программа и методика испытаний
листов 12**

Аннотация

В данном программном документе приведена программа и методика испытаний программного изделия «Программный симулятор PDP-11», предназначенного для изучения архитектуры ЭВМ PDP-11, языка ассемблера и работы внешних устройств.

Содержание

Аннотация.....	2
Основная часть.....	4
1 Введение.....	4
1.1 Наименование программы.....	4
1.2 Условное обозначение темы разработки.....	4
2 Назначение и область применения.....	4
2.1 Назначение программы.....	4
2.2 Область применения программы.....	4
3 Технические характеристики.....	4
3.1 Постановка задачи на разработку программы.....	4
3.2 Описание функционирования программы.....	5
3.2.1 Описание интерфейса внешнего устройства.....	5
3.2.2 Описание функционирования программы.....	6
3.2.3 Возможные взаимодействия программы с другими программами.....	6
3.2.4 Описание и обоснование метода организации входных данных.....	7
3.2.5 Описание и обоснование метода организации выходных данных.....	7
3.2.6 Описание и обоснование выбора состава технических средств.....	7
3.2.7 Описание и обоснование выбора состава программных средств.....	7
4 Ожидаемые технико-экономические показатели.....	7
5 Источники, используемые при разработке.....	8
Лист регистрации изменений.....	9

Основная часть

1 Объект испытаний

1.1 Наименование испытываемой программы

Наименование – «Программный симулятор PDP-11».

1.2 Область применения испытываемой программы

Программа предназначена к применению в учебных и исследовательских целях.

1.3 Обозначение испытываемой программы

Наименование темы разработки – «Программный симулятор PDP-11».

2 Цель испытаний

Цель проведения испытаний – проверка соответствия характеристик разработанной программы функциональным и иным, отдельным видам требований, изложенным в программном документе «Техническое задание».

3 Требования к программе

При проведении испытаний функциональные характеристики программы подлежат проверке на соответствие требованиям, изложенным в п. «Требования к функциональным характеристикам» Технического задания.

4 Требования к программной документации

Состав программной документации должен включать в себя:

1. Техническое задание;
2. Спецификация;
3. Текст программы;
4. Описание программы;
5. Программу и методику испытаний;
6. Пояснительную записку симулятора;
7. Пояснительную записку модуля внешних устройств;
8. Описание применения;
9. Руководство программиста;

10. Руководство оператора.

5 Состав и порядок испытаний

5.1 Технические средства, используемые во время испытаний

Состав используемых во время испытаний технических средств:

- процессор с тактовой частотой не менее 2 ГГц;
- оперативную память объемом не менее 1 ГБ;
- свободное место на диске не менее 200 МБ.

5.2 Программные средства, используемые во время испытаний

Поддерживаемые операционные системы и необходимые библиотеки представлены на следующей странице:

<https://github.com/dotnet/core/blob/main/release-notes/6.0/supported-os.md>

5.3 Порядок проведения испытаний

Испытания проводятся в два этапа:

1. Проверка комплектности программной документации;
2. Проверка программы на соответствие функциональным требованиям:
 - a) Программа позволяет открывать, сохранять, редактировать написанное ПО;
 - b) Программа позволяет ассемблировать написанное ПО;
 - c) Программа позволяет запускать написанное ПО в пошаговом и в автоматическом режимах;
 - d) Программа позволяет настраивать внешние устройства;
 - e) Программа предоставляет интерфейс внешних устройств;
 - f) Программа поддерживает прерывания внешних устройств;
 - g) Программа поддерживает аппаратные ловушки;
 - h) Программа отображает текущее состояние симулятора;
 - i) Программа предоставляет справку о симуляторе;
 - j) В случае ошибок программа выводит сообщения об ошибке;

6 Методы испытаний

6.1 Проверка комплектности программной документации

Проверка комплектности программной документации на программное изделие производится визуально представителем службы, ответственной за эксплуатацию. В ходе проверки сопоставляется состав и комплектность программной документации, представленной Разработчиком, с перечнем программной документации, приведенным в п. «Состав программной документации, предъявляемой на испытания» настоящего документа.

Проверка считается завершенной в случае соответствия состава и комплектности программной документации, представленной Разработчиком, перечню программной документации, приведенному в указанном выше пункте.

6.2 Проверка программы на соответствие функциональным требованиям

6.2.1 Программа позволяет редактировать ПО

Для проверки данного функционала необходимо:

1. Запустить программу;
2. В появившемся окне выбрать *Create*;
3. Ввести имя проекта *test*;
4. Выбрать директорию проекта *C:\test*;
5. Убедиться, что были открыты файлы *test.pdp11proj* и *main.asm*;
6. Убедиться, что эти файлы были созданы в директории *C:\test*, и что их содержимое совпадает с содержимым в окне программы;
7. Изменить файл *main.asm*;
8. Нажать кнопку *File>Save*;
9. Убедиться, что файл был изменен в соответствии с введенными данными.

6.2.2 Программа позволяет ассемблировать ПО

Для проверки данного функционала необходимо:

1. Запустить программу;
2. В появившемся окне выбрать *Create*;
3. Ввести имя проекта *test*;
4. Выбрать директорию проекта *C:\test*;
5. Ввести в файл *main.asm* текст в соответствии с листингом 1;
6. Нажать кнопку *Build*;
7. Удостоверится, что появилось сообщение с сообщением об успехе;
8. Удостовериться, что в директории *C:\test* был создан файл *test.pdp11bin* с содержимым, идентичном листингу 2.

Листинг 1. Входные данные для испытания ассемблирования

```
START:
    sub #6, D
    bmi LESS
    beq LESS
    mov A, R0
    bis B, R0
    mov R0, F
    jmp GLOBAL_END

LESS:
    mov A, R0
    bic B, R0
    mov R0, F

GLOBAL_END:
    halt

D: .word 1
A: .word 6
B: .word 3
F: .blkw 1
```

Листинг 2. Ожидаемые выходные данные для испытания ассемблирования

```
START:
    sub #6, D
    bmi LESS
    beq LESS
    mov A, R0
    bis B, R0
    mov R0, F
    jmp GLOBAL_END

LESS:
```

```

    mov A, R0
    bic B, R0
    mov R0, F

GLOBAL_END:
    halt

D: .word 1
A: .word 6
B: .word 3
F: .blkw 1

```

6.2.3 Программа поддерживает пошаговое и автоматическое исполнение ПО

Для проверки данного функционала необходимо:

1. Выполнить пункты из п. 6.2.3;
2. Нажать кнопку *Run*;
3. Убедиться, что открылось окно исполнителя;
4. Убедиться, что код программы совпадает с листингом 2;
5. Нажать кнопку *Step*;
6. Убедиться, что была выполнена одна команда;
7. Убедиться, что выделена следующая команда;
8. Нажать кнопку *Run*;
9. Убедиться, что программа завершила свою работу на команде *HALT*.

6.2.4 Программа позволяет настраивать внешние устройства

Для проверки данного функционала необходимо:

1. Запустить программу;
2. В появившемся окне выбрать *Create*;
3. Ввести имя проекта *test*;
4. Выбрать директорию проекта *C:\test*;
5. Нажать кнопку *Settings*;
6. Убедиться, что открылось окно настроек;
7. Нажать в области списка устройств правую кнопку мыши;
8. Убедиться, что открылось контекстное меню;
9. Нажать в нем кнопку *Add*;

10. В открывшемся окне выбрать файл внешнего устройства;
11. Убедиться, что внешнее устройство добавилось в список;
12. Выделить внешнее устройство в списке;
13. В контекстном меню выбрать Delete;
14. Убедиться, что внешнее устройство пропало из списка.

6.2.5 Программа предоставляет интерфейс внешних устройств

Для проверки данного функционала необходимо запустить проект https://github.com/mrypdm/asvt_sem7_kursach/tree/master/demo/device в автоматическом режиме.

После исполнения удостовериться, что в директории с проектом появился файл *memory.bin* с содержимым 0xA1, 0xFD по адресам 0xE, 0xF соответственно.

6.2.6 Программа поддерживает прерывания внешних устройств

Для проверки данного функционала необходимо запустить программу из п. 6.2.5 в пошаговом режиме и убедиться, что в ходе программы был переход на метку *HANDLER*.

6.2.7 Программа поддерживает аппаратные ловушки

Для проверки данного функционала необходимо запустить проект https://github.com/mrypdm/asvt_sem7_kursach/tree/master/demo/hardtrap в пошаговом режиме и убедиться, что в ходе исполнения программы был переход на метку *HANDLER*.

6.2.8 Программа отображает текущее состояние симулятора

Для проверки данного функционала запустить одну из приведенных выше программ в пошаговом режиме и удостовериться, что блок состояния окна исполнителя меняется в соответствии с состоянием симулятора.

6.2.9 Программа предоставляет справку о симуляторе

Для проверки данного функционала:

1. Удостовериться, что кнопка Help>Architecture главного окна открывает изображение схемы ЭВМ;

2. Удостовериться, что кнопка Help>Tutorial главного окна открывает справку о методах адресации и доступных командах;

6.2.10 Программа выводит сообщения об ошибках

Для проверки данного функционала:

1. Разработать некорректную программу;
2. Удостовериться, что при ассемблировании возникло сообщение об ошибке;
3. Удостовериться, что файл `rdp11bin` не был создан;
4. Удостовериться, что программа штатно продолжила работу,

Либо:

1. Нажать кнопку закрытия файла проекта;
2. Удостовериться, что возникло сообщение об ошибке.

Либо

1. Создать проект в соответствии с п. 6.2.7;
2. Удалить из файла *main.asm* строку 13;
3. Выполнить программу;
4. Удостовериться, что возникло сообщение о двойной ошибке шины.

Лист регистрации изменений

[illegible]

