

Отчет по лабораторной работе №4

Дисциплина: Архитектура компьютера

Авдеенко Марьяна Дмитриевна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Задания для самостоятельной работы	11
6	Выводы	13
	Список литературы	14

Список иллюстраций

4.1	Подготовка к созданию программы Hello world!	8
4.2	Окно текстового редактора (введение программы Hello world!) . .	9
4.3	Преобразование текста в объектный код	9
4.4	Компиляция исходного кода	9
4.5	Создание исполняемого файла	10
4.6	Выполнение команды компановщиком	10
4.7	Запуск исполняемого файла	10
5.1	Копирование файла	11
5.2	Выведение текста файла с именем и фамилией	11
5.3	Выведение текста файла с именем и фамилией	12
5.4	Выведение текста файла с именем и фамилией	12

Список таблиц

1 Цель работы

Цель работы: освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM.

2 Задание

Здесь приводится описание задания в соответствии с рекомендациями методического пособия и выданным вариантом.

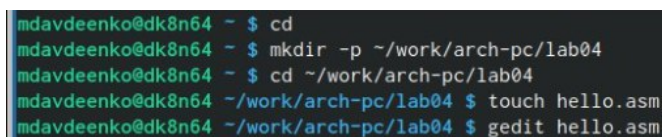
3 Теоретическое введение

Основными функциональными элементами любой электронно-вычислительной машины

(ЭВМ) являются центральный процессор, память и периферийные устройства (рис. 4.1). Взаимодействие этих устройств осуществляется через общую шину, к которой они подклю- чены. Физически шина представляет собой большое количество проводников, соединяющих устройства друг с другом. В современных компьютерах проводники выполнены в виде элек- тропроводящих дорожек на ма- теринской (системной) плате. Основной задачей процессора является обработка информации, а также организация координации всех узлов компьютера. В состав центрального процессора (ЦП) входят следующие устройства: • арифметико- логическое устройство (АЛУ) — выполняет логические и арифметиче- ские дей- ствия, необходимые для обработки информации, хранящейся в памяти; • устрой- ство управления (УУ) — обеспечивает управление и контроль всех устройств компьютера; • регистры — сверхбыстрая оперативная память небольшого объё- ма, входящая в со- став процессора, для временного хранения промежуточных результатов выполнения инструкций; регистры процессора делятся на два типа: регистры общего назначения и специальные регистры.

4 Выполнение лабораторной работы

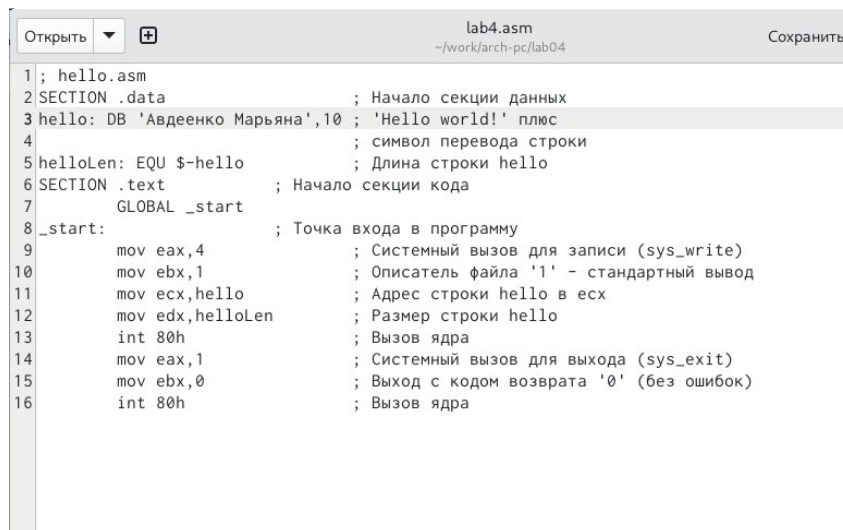
- 1) Создала каталог для работы с программами на языке ассемблера NASM (команда `mkdir -p ~/work/arch-pc/lab04`), перешла в эту директорию, в ней создала текстовый файл с именем `hello.asm` (команда `touch hello.asm`), открыла этот файл с помощью текстового редактора `gedit` (команда `gedit hello.asm`) (рис. 4.1).



```
mdavdeenko@dk8n64 ~ $ cd
mdavdeenko@dk8n64 ~ $ mkdir -p ~/work/arch-pc/lab04
mdavdeenko@dk8n64 ~ $ cd ~/work/arch-pc/lab04
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ touch hello.asm
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ gedit hello.asm
```

Рис. 4.1: Подготовка к созданию программы Hello world!

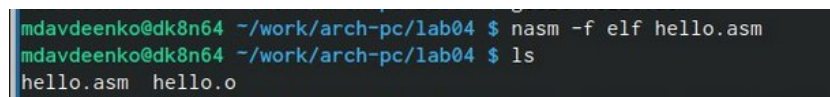
- 2) В открывшемся окне текстового редактора ввела текст указанный в задании лабораторной работы (рис. 4.2).



```
1; hello.asm
2SECTION .data          ; Начало секции данных
3hello: DB 'Авдеенко Марьяна',10 ; 'Hello world!' плюс
4                        ; символ перевода строки
5helloLen: EQU $-hello   ; Длина строки hello
6SECTION .text          ; Начало секции кода
7GLOBAL _start
8_start:                ; Точка входа в программу
9    mov eax,4           ; Системный вызов для записи (sys_write)
10   mov ebx,1           ; Описатель файла '1' - стандартный вывод
11   mov ecx,hello       ; Адрес строки hello в есх
12   mov edx,helloLen    ; Размер строки hello
13   int 80h             ; Вызов ядра
14   mov eax,1           ; Системный вызов для выхода (sys_exit)
15   mov ebx,0           ; Выход с кодом возврата '0' (без ошибок)
16   int 80h             ; Вызов ядра
```

Рис. 4.2: Окно текстового редактора (введение программы Hello world!)

- 3) С помощью NASM превратила текст программы в объектный код (команда `nasm -f elf hello.asm`), затем проверила выполнение команды (команда `ls`) (рис. 4.3).



```
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ls
hello.asm hello.o
```

Рис. 4.3: Преобразование текста в объектный код

- 4) Скомпилировала исходный файл `hello.asm` в `obj.o` (команда `nasm -o obj.o -f elf -g -l list.lst hello.asm`), затем проверила выполнение команды (команда `ls`) (рис. 4.4).



```
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ls
hello.asm hello.o list.lst obj.o
```

Рис. 4.4: Копиляция исходного кода

- 5) передать на обработку компоновщику объектный файл (команда `ld -m`

elf_i386 hello.o -o hello), затем проверила выполнение команды (команда ls) (рис. 4.5).

```
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ld -m elf_i386 hello.o -o hello
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  list.lst  obj.o
```

Рис. 4.5: Создание исполняемого файла

- 6) Выполнила следующую команду (команда `ld -m elf_i386 obj.o -o main`), (рис. 4.6).

```
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ld --help
```

Рис. 4.6: Выполнение команды компоновщиком

- 7) Запустила на выполнение созданный исполняемый файл, находящийся в текущем каталоге (команда `./hello`) (рис. 4.7).

```
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ./hello
Hello world!
```

Рис. 4.7: Запуск исполняемого файла

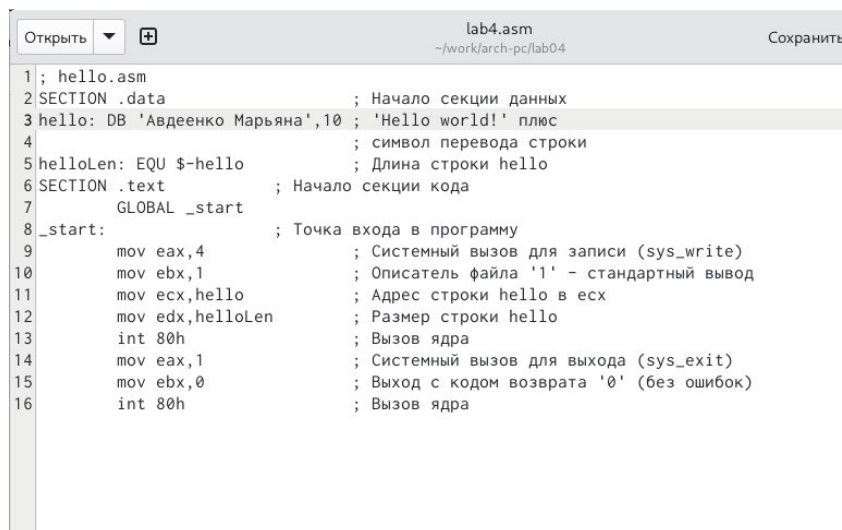
5 Задания для самостоятельной работы

- 1) В каталоге `~/work/arch-pc/lab04` с помощью команды `cp` создала копию файла `hello.asm` с именем `lab4.asm` (рис. 5.1).

```
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ cp hello.asm lab4.asm
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  lab4.asm  list.lst  main  obj.o
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $
```

Рис. 5.1: Копирование файла

- 2) С помощью текстового редактора `gedit` внесла изменения в текст программы в файле `lab4.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с моими фамилией и именем (рис. 5.2).



```
lab4.asm
~/work/arch-pc/lab04

1; hello.asm
2SECTION .data          ; Начало секции данных
3hello: DB 'Авдеенко Марьяна',10 ; 'Hello world!' плюс
4                        ; символ перевода строки
5helloLen: EQU $-hello   ; Длина строки hello
6SECTION .text          ; Начало секции кода
7    GLOBAL _start
8_start:                ; Точка входа в программу
9        mov eax,4        ; Системный вызов для записи (sys_write)
10       mov ebx,1        ; Описатель файла '1' - стандартный вывод
11       mov ecx,hello     ; Адрес строки hello в есх
12       mov edx,helloLen  ; Размер строки hello
13       int 80h          ; Вызов ядра
14       mov eax,1        ; Системный вызов для выхода (sys_exit)
15       mov ebx,0        ; Выход с кодом возврата '0' (без ошибок)
16       int 80h          ; Вызов ядра
```

Рис. 5.2: Выведение текста файла с именем и фамилией

- 3) Оттранслировала полученный текст программы lab4.asm в объектный файл, затем выполнила компоновку объектного файла и запустила получившийся исполняемый файл (рис. 5.3).

```
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ gedit lab4.asm
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ nasm -f elf lab4.asm
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ls
hello.o  hello.asm  hello.o  lab4.asm  lab4.o  list.lst  main  obj.o
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ nasm -o obj1.o -f elf -g -l list1.lst lab4.asm
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ls
hello.o  hello.asm  hello.o  lab4.asm  lab4.o  list1.lst  list.lst  main  obj1.o  obj.o
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ d -m elf_i386 lab4.o -o lab4
bash: d: команда не найдена
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ld -m elf_i386 lab4.o -o lab4
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj1.o -o main1
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ ./lab4
Авдеенко Марияна
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $
```

Рис. 5.3: Выведение текста файла с именем и фамилией

- 4) Скопировала файлы hello.asm и lab4.asm в локальный репозиторий в каталог ~/work/study/2023-2024/Архитектура/arch-pc/labs/lab04/ (рис. 5.4).

```
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ cp hello.asm ~/work/study/2023-2024/Архитектура/arch-pc/labs/lab04
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ cd ~/work/study/2023-2024/Архитектура/arch-pc/labs/lab04
mdavdeenko@dk8n64 ~/work/study/2023-2024/Архитектура/arch-pc/labs/lab04 $ ls
hello.asm  presentation  report
mdavdeenko@dk8n64 ~/work/study/2023-2024/Архитектура/arch-pc/labs/lab04 $ cd ~/work/arch-pc/lab04
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ cp lab4.asm ~/work/study/2023-2024/Архитектура/arch-pc/labs/lab04
mdavdeenko@dk8n64 ~/work/arch-pc/lab04 $ cd ~/work/study/2023-2024/Архитектура/arch-pc/labs/lab04
mdavdeenko@dk8n64 ~/work/study/2023-2024/Архитектура/arch-pc/labs/lab04 $ ls
hello.asm  lab4.asm  presentation  report
mdavdeenko@dk8n64 ~/work/study/2023-2024/Архитектура/arch-pc/labs/lab04 $
```

Рис. 5.4: Выведение текста файла с именем и фамилией

6 Выводы

В ходе выполнения данной лабораторной работы мной были освоены процедуры компиляции и сборки программ, написанных на ассемблере NASM.

Список литературы