

Отчёт по лабораторной работе 4

Архитектура компьютера

Ушаков Данила Алексеевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Программа Hello world!	6
2.2	Транслятор NASM	7
2.3	Расширенный синтаксис командной строки NASM	8
2.4	Компоновщик LD	8
2.5	Запуск исполняемого файла	9
2.6	Задание для самостоятельной работы	9
3	Выводы	11

Список иллюстраций

2.1	Создан каталог для работы и файл для программы	6
2.2	Программа в файле hello.asm	7
2.3	Трансляция программы	8
2.4	Трансляция программы с дополнительными опциями	8
2.5	Компоновка программы	9
2.6	Компоновка программы	9
2.7	Запуск программы	9
2.8	Скопировал файл	9
2.9	Программа в файле lab4.asm	10
2.10	Проверка программы lab4.asm	10

Список таблиц

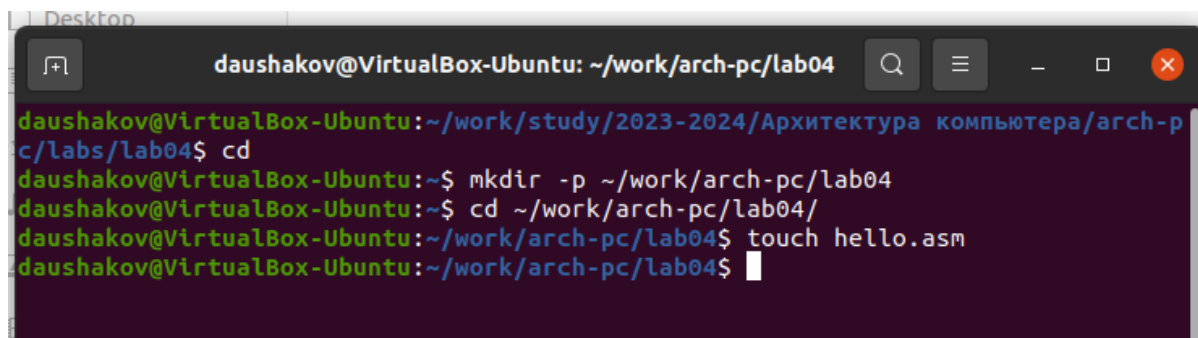
1 Цель работы

Целью работы является освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM.

2 Выполнение лабораторной работы

2.1 Программа Hello world!

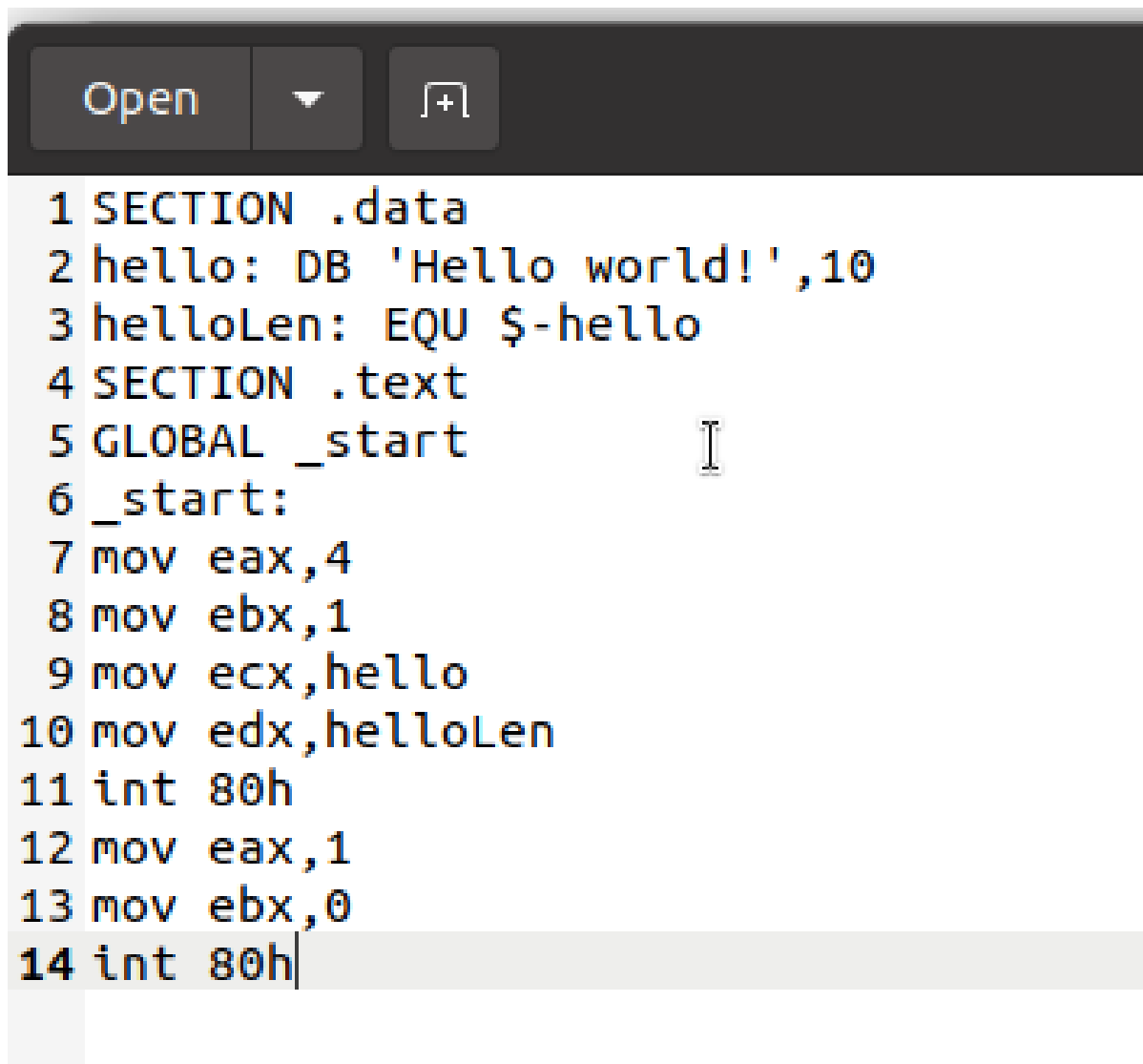
Создал каталог lab04 командой `mkdir`, перешел в него с помощью команды `cd` и создал файл `hello.asm`, в который напишу программу. Убеждаюсь с помощью команды `ls`, что создал файл.

A screenshot of a terminal window titled "daushakov@VirtualBox-Ubuntu: ~/work/arch-pc/lab04". The terminal shows a series of commands and their outputs:

```
daushakov@VirtualBox-Ubuntu:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04$ cd
daushakov@VirtualBox-Ubuntu:~$ mkdir -p ~/work/arch-pc/lab04
daushakov@VirtualBox-Ubuntu:~$ cd ~/work/arch-pc/lab04/
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ touch hello.asm
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
```

Рис. 2.1: Создан каталог для работы и файл для программы

Написал программу по заданию на языке ассемблера.



```
1 SECTION .data
2 hello: DB 'Hello world!',10
3 helloLen: EQU $-hello
4 SECTION .text
5 GLOBAL _start
6 _start:
7 mov eax,4
8 mov ebx,1
9 mov ecx,hello
10 mov edx,helloLen
11 int 80h
12 mov eax,1
13 mov ebx,0
14 int 80h
```

Рис. 2.2: Программа в файле hello.asm

2.2 Транслятор NASM

NASM превращает текст программы в объектный код. Если текст программы набран без ошибок, то транслятор преобразует текст программы из файла hello.asm в объектный код, который запишется в файл hello.o.

Транслировал файл командой `nasm`. Получился объектный файл hello.o.

```
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ls
hello.asm
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ nasm -f elf hello.asm
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ls
hello.asm  hello.o
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
```

Рис. 2.3: Трансляция программы

2.3 Расширенный синтаксис командной строки NASM

Полный вариант командной строки nasm выглядит следующим образом:

nasm [-@ косвенный_файл_настроек] [-o объектный_файл] [-f формат_объектного_файла] [-l листинг] [параметры...] [--] исходный_файл

Транслировал файл командой nasm с дополнительными опциями. С опцией -l Получил файл листинга list.lst, с опцией -f объектный файл obj.o, с опцией -g в программу добавилась отладочная информация.

```
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ls
hello.asm  hello.o
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.asm
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ls
hello.asm  hello.o  list.lst  obj.o
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
```

Рис. 2.4: Трансляция программы с дополнительными опциями

2.4 Компоновщик LD

Чтобы получить исполняемую программу, объектный файл необходимо передать на обработку компоновщику.

Выполнил команду ld и получил исполняемый файл hello из объектного файла hello.o.


```
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst obj.o
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
```

Рис. 2.5: Компоновка программы

Еще раз выполнил команду `ld` для объектного файла `obj.o` и получил исполняемый файл `main`.

```
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst obj.o
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst main obj.o
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
```

Рис. 2.6: Компоновка программы

2.5 Запуск исполняемого файла

Запустил исполняемые файлы.

```
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ./hello
Hello world!
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ./main
Hello world!
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
```

Рис. 2.7: Запуск программы

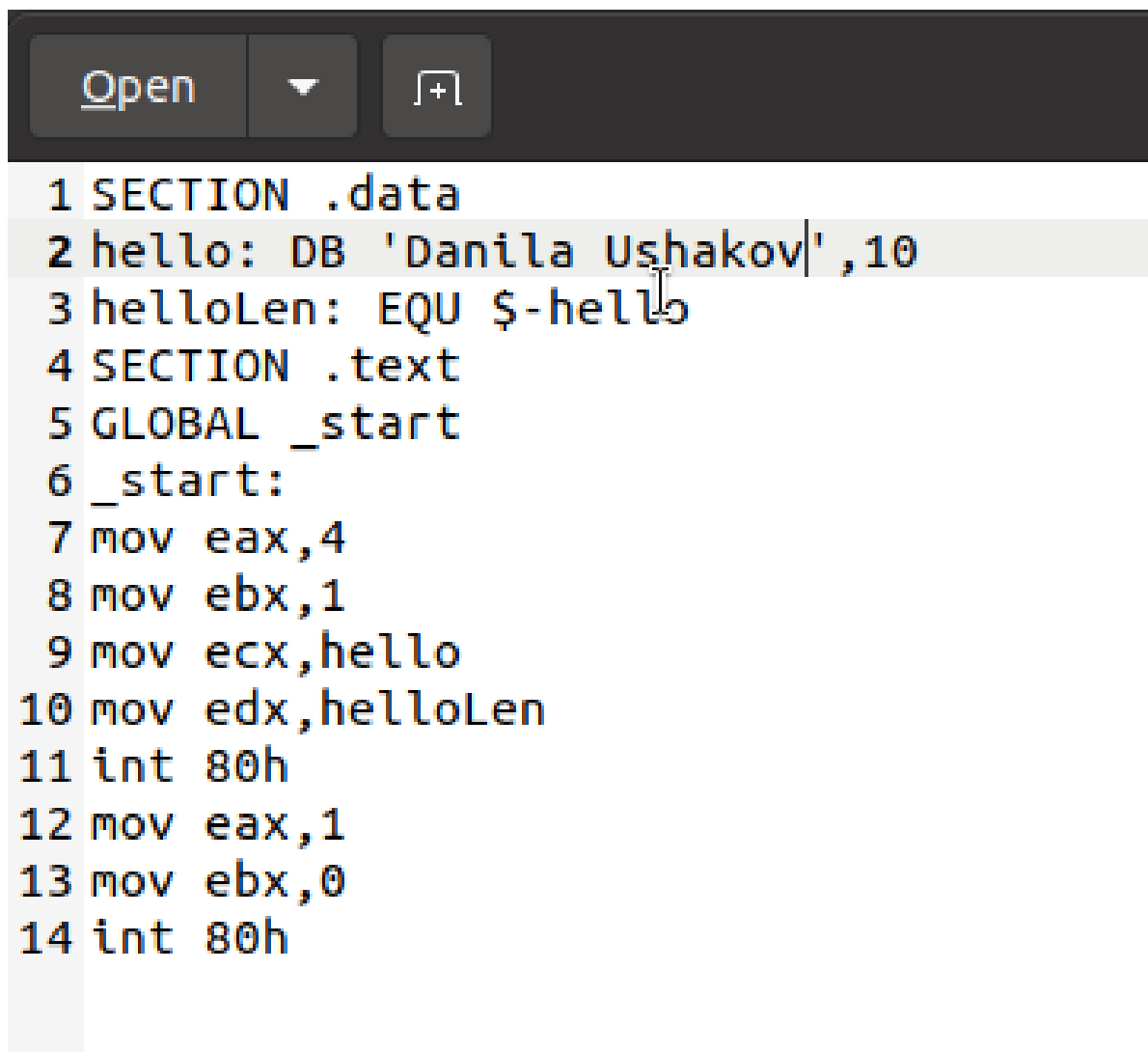
2.6 Задание для самостоятельной работы

Скопировал файл `hello.asm` в файл `lab4.asm`.

```
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o lab4.asm list.lst main obj.o
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
```

Рис. 2.8: Скопировал файл

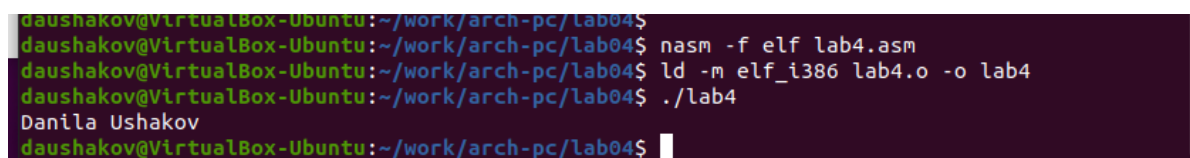
Изменил сообщение Hello world на свое имя.



```
1 SECTION .data
2 hello: DB 'Danila Ushakov',10
3 helloLen: EQU $-hello
4 SECTION .text
5 GLOBAL _start
6 _start:
7 mov eax,4
8 mov ebx,1
9 mov ecx,hello
10 mov edx,helloLen
11 int 80h
12 mov eax,1
13 mov ebx,0
14 int 80h
```

Рис. 2.9: Программа в файле lab4.asm

Запустил программу и проверил.



```
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o lab4
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$ ./lab4
Danila Ushakov
daushakov@VirtualBox-Ubuntu:~/work/arch-pc/lab04$
```

Рис. 2.10: Проверка программы lab4.asm

3 Выводы

Освоил процесс компиляции и сборки программ, написанных на ассемблере `nasm`.