

Отчёт по лабораторной работе 4

Архитектура компьютеров

Сухоруков Сергей Андреевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Программа Hello world!	6
2.2	Транслятор NASM	7
2.3	Компоновщик LD	8
2.4	Выполнение заданий для самостоятельной работы.	9
3	Выводы	11

Список иллюстраций

2.1	Создание каталога и файла	6
2.2	Программа hello.asm	7
2.3	Трансляция hello.asm	7
2.4	Трансляция hello.asm с дополнительными опциями	8
2.5	Линковка программы	8
2.6	Линковка программы	8
2.7	Запуск программ	9
2.8	Код программы в файле lab4.asm	9
2.9	Запуск программы lab4.asm	10

Список таблиц

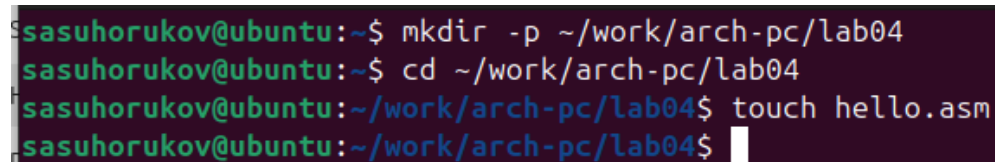
1 Цель работы

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

2 Выполнение лабораторной работы

2.1 Программа Hello world!

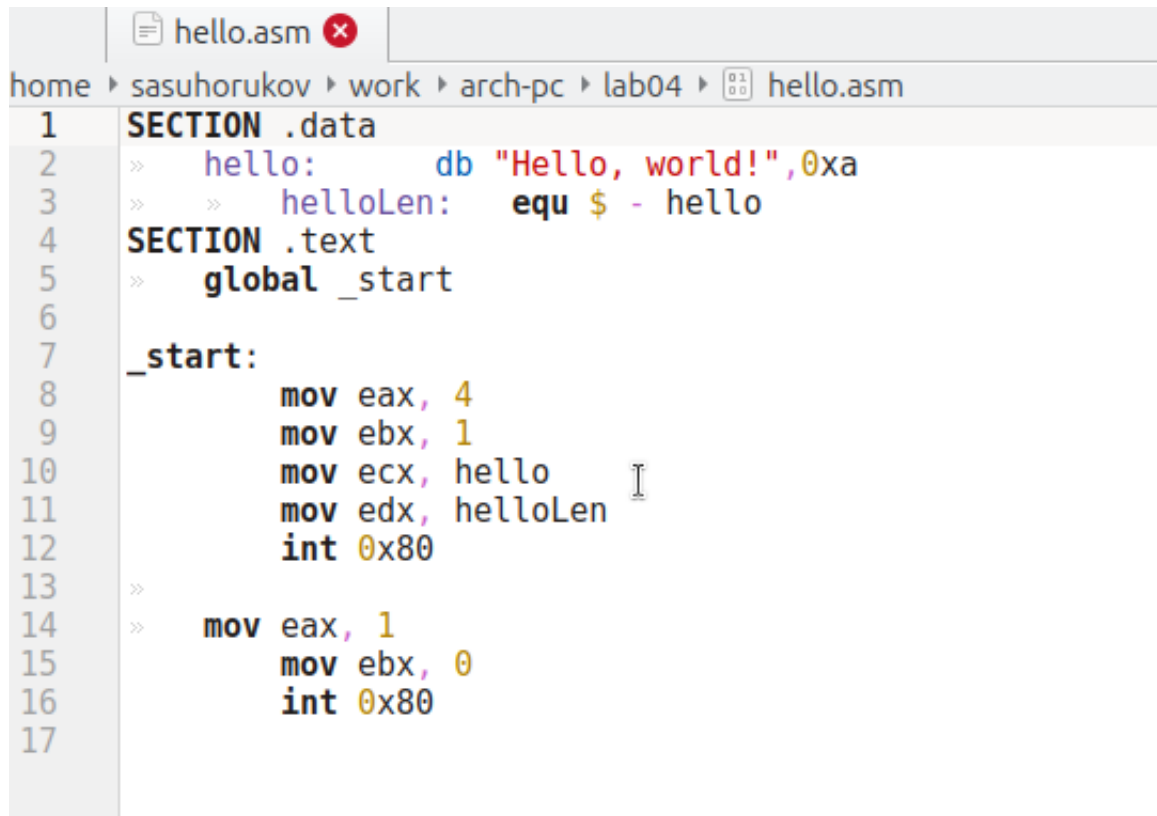
Создаю каталог lab04 командой mkdir, перехожу в него с помощью команды cd, создаю файл hello.asm. (рис. 2.1)

A terminal window with a dark purple background and light green text. It shows four lines of commands and their execution. The first line creates a directory named 'lab04' under the path '~/work/arch-pc/'. The second line changes the current directory to 'lab04'. The third line creates a file named 'hello.asm' in the current directory. The fourth line shows the prompt after the file creation.

```
sasuhorukov@ubuntu:~$ mkdir -p ~/work/arch-pc/lab04
sasuhorukov@ubuntu:~$ cd ~/work/arch-pc/lab04
sasuhorukov@ubuntu:~/work/arch-pc/lab04$ touch hello.asm
sasuhorukov@ubuntu:~/work/arch-pc/lab04$
```

Рисунок 2.1: Создание каталога и файла

Открыл файл и написал код программы по заданию.(рис. 2.2)

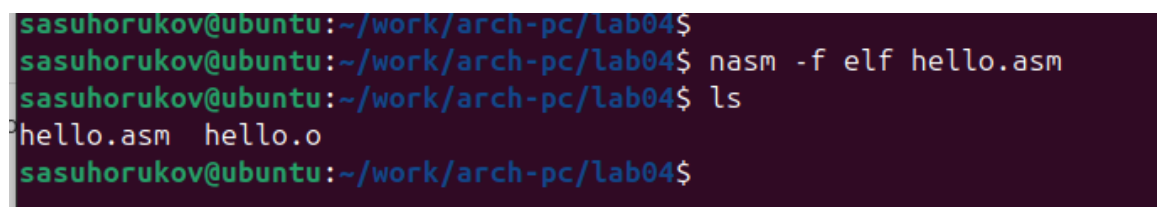


```
1 SECTION .data
2 >> hello: db "Hello, world!",0xa
3 >> helloLen: equ $ - hello
4 SECTION .text
5 >> global _start
6
7 _start:
8     mov eax, 4
9     mov ebx, 1
10    mov ecx, hello
11    mov edx, helloLen
12    int 0x80
13
14    >> mov eax, 1
15    mov ebx, 0
16    int 0x80
17
```

Рисунок 2.2: Программа hello.asm

2.2 Транслятор NASM

Транслирую файл командой `nasm`. Получился объектный файл `hello.o` (рис. 2.3)

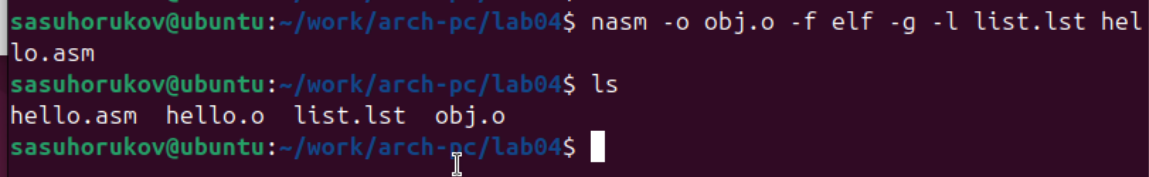


```
sasuhorukov@ubuntu:~/work/arch-pc/lab04$
sasuhorukov@ubuntu:~/work/arch-pc/lab04$ nasm -f elf hello.asm
sasuhorukov@ubuntu:~/work/arch-pc/lab04$ ls
hello.asm  hello.o
sasuhorukov@ubuntu:~/work/arch-pc/lab04$
```

Рисунок 2.3: Трансляция hello.asm

Транслирую файл командой `nasm` с дополнительными опциями. (рис. 2.4)
Получился файл листинга `list.lst`, объектный файл `obj.o`, в программу добави-

лась отладочная информация.

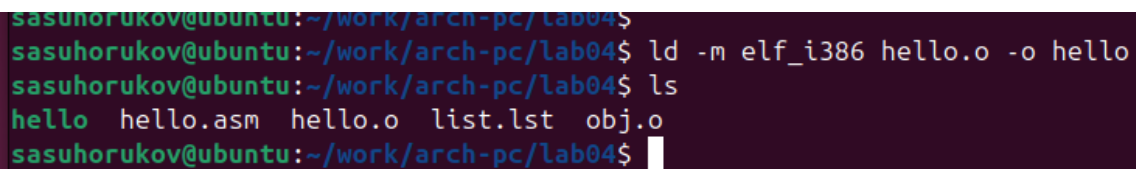


```
sasuhorukov@ubuntu:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.asm
sasuhorukov@ubuntu:~/work/arch-pc/lab04$ ls
hello.asm  hello.o  list.lst  obj.o
sasuhorukov@ubuntu:~/work/arch-pc/lab04$
```

Рисунок 2.4: Трансляция hello.asm с дополнительными опциями

2.3 Компоновщик LD

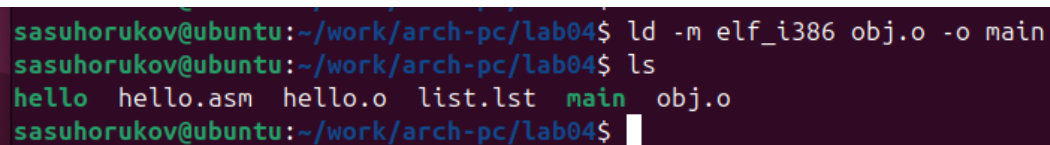
Выполняю линковку командой ld и получил исполняемый файл. (рис. 2.5)



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

Рисунок 2.5: Линковка программы

Еще раз выполняю линковку для объектного файла obj.o и получаю исполняемый файл main.(рис. 2.6)



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

Рисунок 2.6: Линковка программы

Запускаю исполняемые файлы.(рис. 2.7)


```

sasuhorukov@ubuntu:~/work/arch-pc/lab04$ ./hello
Hello, world!
sasuhorukov@ubuntu:~/work/arch-pc/lab04$ ./main
Hello, world!
sasuhorukov@ubuntu:~/work/arch-pc/lab04$

```

Рисунок 2.7: Запуск программ

2.4 Выполнение заданий для самостоятельной работы.

Копирую программу в новый файл.

Изменяю сообщение Hello world на свое имя (рис. 2.8) и запускаю новую программу. (рис. 2.9)

```

home ▸ sasuhorukov ▸ work ▸ arch-pc ▸ lab04 ▸ lab4.asm
1  SECTION .data
2  >>  hello:      db "Сухоруков Сергей",0xa
3  >>      helloLen: equ $ - hello
4  SECTION .text
5  >>  global _start
6
7  _start:
8      mov eax, 4
9      mov ebx, 1
10     mov ecx, hello
11     mov edx, helloLen
12     int 0x80
13
14     >>  mov eax, 1
15         mov ebx, 0
16         int 0x80
17

```

Рисунок 2.8: Код программы в файле lab4.asm

```
sasuhorukov@ubuntu:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
sasuhorukov@ubuntu:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o lab4
sasuhorukov@ubuntu:~/work/arch-pc/lab04$ ./lab4
Сухоруков Сергей
sasuhorukov@ubuntu:~/work/arch-pc/lab04$
```

Рисунок 2.9: Запуск программы lab4.asm

3 Выводы

При выполнении данной лабораторной работы я освоил процесс компиляции и сборки программ, написанных на ассемблере `nasm`.