

Отчёт по лабораторной работе 4

Архитектура компьютеров

Дмитраков Михаил Алексеевич

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	10
2.9	Запуск программы lab4.asm	10

Список таблиц

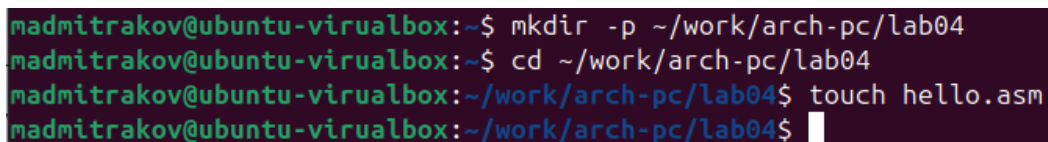
1 Цель работы

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

2 Выполнение лабораторной работы

2.1 Программа Hello world!

На первом этапе я создаю рабочий каталог lab04 с помощью команды `mkdir`, затем перехожу в него командой `cd` и формирую файл `hello.asm`. Данные действия подготавливают среду для дальнейшей разработки программы (рис. 2.1).



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

Рисунок 2.1: Создание каталога и файла

После этого открываю созданный файл и ввожу исходный код программы в соответствии с заданием лабораторной работы (рис. 2.2).

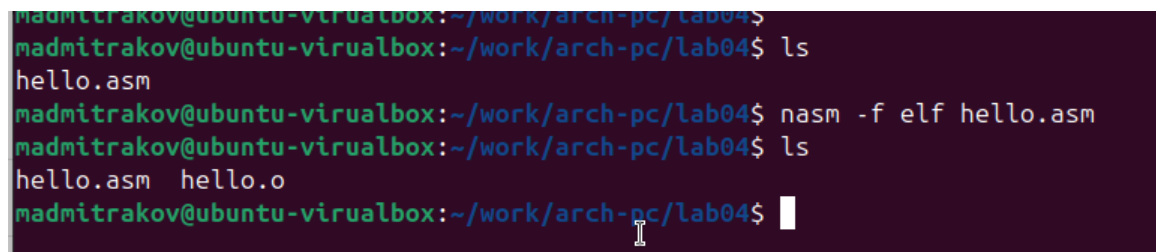


```
1 SECTION .data
2     hello:      db "Hello, world!",0xa
3     helloLen:   equ $ - hello
4 SECTION .text
5     global _start
6
7 _start:
8     mov eax, 4
9     mov ebx, 1
10    mov ecx, hello
11    mov edx, helloLen
12    int 0x80
13
14    mov eax, 1
15    mov ebx, 0
16    int 0x80
17
```

Рисунок 2.2: Программа hello.asm

2.2 Транслятор NASM

Далее выполняю трансляцию исходного файла с помощью транслятора NASM. В результате выполнения команды формируется объектный файл hello.o (рис. 2.3).



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

Рисунок 2.3: Трансляция hello.asm

Затем повторяю трансляцию, используя дополнительные параметры NASM. В ходе этого шага создаются файл листинга `list.lst` и объектный файл `obj.o`, а также в программу добавляется отладочная информация (рис. 2.4).

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

Рисунок 2.4: Трансляция `hello.asm` с дополнительными опциями

2.3 Компоновщик LD

После получения объектных файлов выполняю компоновку с помощью команды `ld`, в результате чего формируется исполняемый файл (рис. 2.5).

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

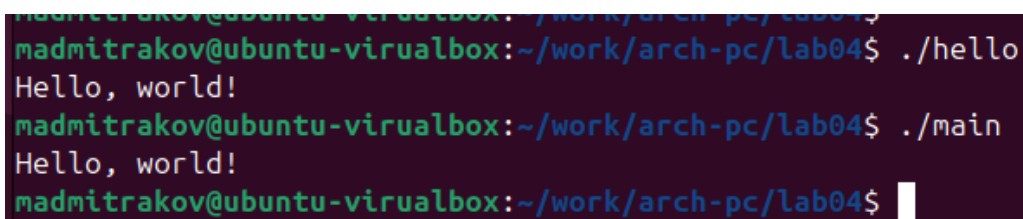
Рисунок 2.5: Линковка программы

Аналогичную операцию провожу для объектного файла `obj.o`, после чего получаю исполняемый файл `main` (рис. 2.6).

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

Рисунок 2.6: Линковка программы

Далее запускаю созданные исполняемые файлы и проверяю корректность их работы (рис. 2.7).

A terminal window with a dark background and green text. The prompt is 'madmitrakov@ubuntu-virtualbox:~/work/arch-pc/lab04\$'. The first command is './hello', followed by the output 'Hello, world!'. The second command is './main', followed by the output 'Hello, world!'. The prompt is now ready for a new command.

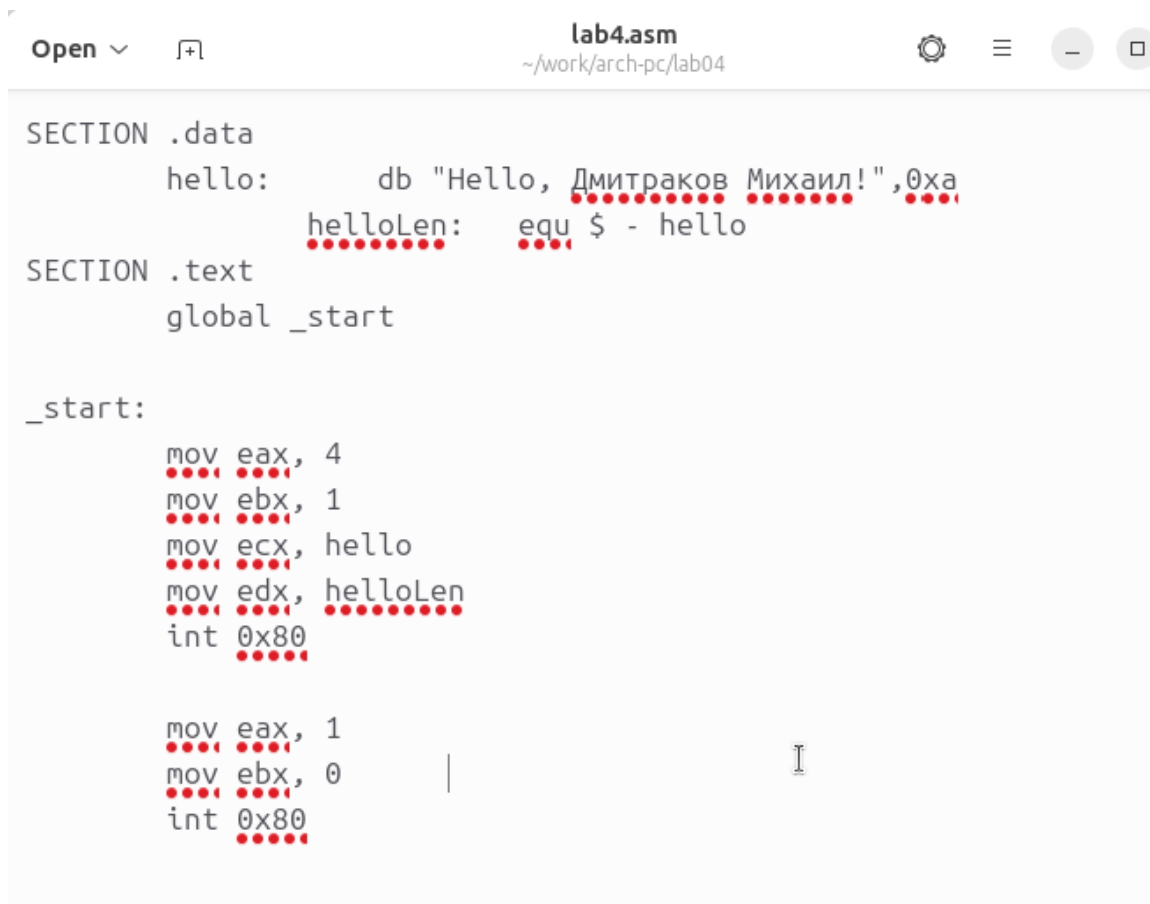
```
madmitrakov@ubuntu-virtualbox:~/work/arch-pc/lab04$ ./hello
Hello, world!
madmitrakov@ubuntu-virtualbox:~/work/arch-pc/lab04$ ./main
Hello, world!
madmitrakov@ubuntu-virtualbox:~/work/arch-pc/lab04$
```

Рисунок 2.7: Запуск программ

2.4 Выполнение заданий для самостоятельной работы

В рамках самостоятельной части лабораторной работы я копирую исходную программу в новый файл.

Затем изменяю выводимое сообщение, заменяя строку `Hello world` на своё имя, после чего компилирую и запускаю обновлённую программу. Результаты редактирования и выполнения представлены на рисунках (рис. 2.8, 2.9).



```
Open  ▾  [icon]  lab4.asm  ~/work/arch-pc/lab04  [gear]  [menu]  [minus]  [close]

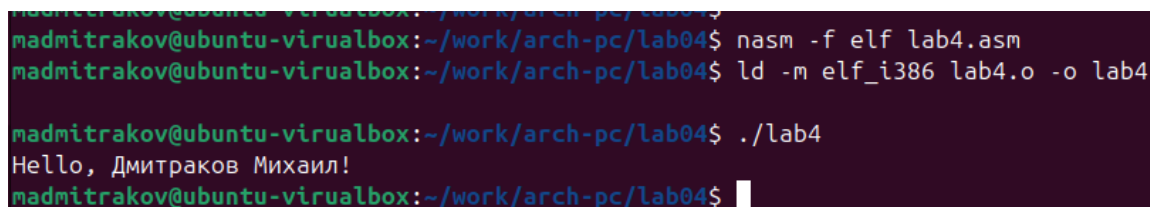
SECTION .data
    hello:      db "Hello, Дмитраков Михаил!",0x0a
                helloLen: equ $ - hello

SECTION .text
    global _start

_start:
    mov eax, 4
    mov ebx, 1
    mov ecx, hello
    mov edx, helloLen
    int 0x80

    mov eax, 1
    mov ebx, 0
    int 0x80
```

Рисунок 2.8: Код программы в файле lab4.asm



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

madmitrakov@ubuntu-virtualbox:~/work/arch-pc/lab04$ ./lab4
Hello, Дмитраков Михаил!
madmitrakov@ubuntu-virtualbox:~/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`.