

Отчёт по лабораторной работе 4

Архитектура компьютеров

Хзиба Хаким НПИбд-02-24

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

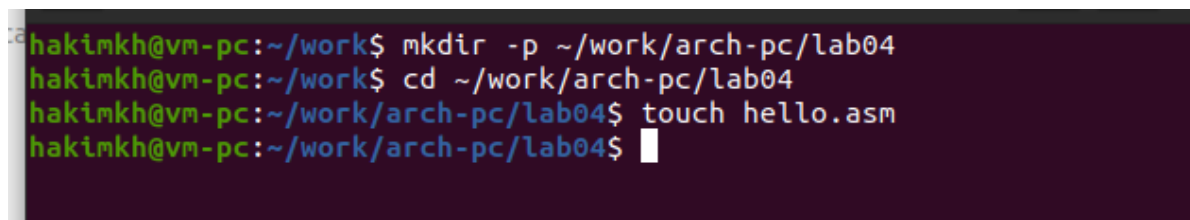
1 Цель работы

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` с помощью команды `mkdir`, перехожу в него с помощью `cd`, и создаю файл `hello.asm`. (рис. 2.1)



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

Рис. 2.1: Создание каталога и файла

Открываю файл и пишу 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

```

Рис. 2.2: Программа hello.asm

2.2 Транслятор NASM

Транслирую файл командой `nasm`, что позволяет получить объектный файл `hello.o`. (рис. 2.3)

```

hakimkh@vm-pc:~/work/arch-pc/lab04$
hakimkh@vm-pc:~/work/arch-pc/lab04$ nasm -f elf hello.asm
hakimkh@vm-pc:~/work/arch-pc/lab04$ ls
hello.asm  hello.o
hakimkh@vm-pc:~/work/arch-pc/lab04$

```

Рис. 2.3: Трансляция hello.asm

Использую команду `nasm` с дополнительными опциями для создания файла листинга `list.lst`, объектного файла `obj.o`, и добавляю отладочную

информацию в программу. (рис. 2.4)

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

Рис. 2.4: Трансляция hello.asm с дополнительными опциями

2.3 Компоновщик LD

Выполняю линковку с помощью команды `ld` и получаю исполняемый файл. (рис. 2.5)

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

Рис. 2.5: Линковка программы

Повторяю линковку для объектного файла `obj.o` и получаю исполняемый файл `main`. (рис. 2.6)

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

Рис. 2.6: Линковка программы

Запускаю полученные исполняемые файлы. (рис. 2.7)


```

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

```

Рис. 2.7: Запуск программ

2.4 Выполнение заданий для самостоятельной работы.

Копирую программу в новый файл.

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

	lab4.asm	hello.asm
1	SECTION .data	
2	» hello: db "Hakim",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		

Рис. 2.8: Код программы в файле lab4.asm

```
hakimkh@vm-pc:~/work/arch-pc/lab04$  
hakimkh@vm-pc:~/work/arch-pc/lab04$ nasm -f elf lab4.asm  
hakimkh@vm-pc:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o lab4  
hakimkh@vm-pc:~/work/arch-pc/lab04$ ./lab4  
Hakim  
hakimkh@vm-pc:~/work/arch-pc/lab04$
```

Рис. 2.9: Запуск программы lab4.asm

3 Выводы

В ходе выполнения лабораторной работы я освоил процесс компиляции и сборки программ на ассемблере NASM. Полученные навыки включают создание объектных файлов, использование транслятора и компоновщика, а также работу с отладочной информацией и выполнение программ.