

Лабораторная работа №4

Архитектура Компьютера

Овчинников Антон Григорьевич

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выполнение заданий для самостоятельной работы	10
6	Листинги	12
7	Выводы	14

Список иллюстраций

4.1	Создание файла	8
4.2	Введенный текст	8
4.3	Созданные файлы	9
4.4	Выполнение нужных для компоновки команд	9
5.1	Компоновка объектного файла и его запуск	10
5.2	Перемещение файлов	10
5.3	Проверка файлов	11

Список таблиц

1 Цель работы

Освоить процедуры компиляции и сборки программ, написанных на ассемблере NASM.

2 Задание

Сделать отчет.

3 Теоретическое введение

Взаимодействие этих устройств осуществляется через общую шину, к которой они подклю- чены. Физически шина представляет собой большое количество про- водников, соединяющих устройства друг с другом. В современных компьютерах проводники выполнены в виде элек- тропроводящих дорожек на материнской (системной) плате. Основной задачей процессора является обработка информа- ции, а также организация координации всех узлов компьютера. В состав централь- ного процессора (ЦП) входят следующие устройства: • арифметико-логическое устройство (АЛУ) — выполняет логические и арифметиче- ские действия, необхо- димые для обработки информации, хранящейся в памяти; • устройство управ- ления (УУ) — обеспечивает управление и контроль всех устройств компьютера; • регистры — сверхбыстрая оперативная память небольшого объёма, входящая в со- став процессора, для временного хранения промежуточных результатов выполнения инструкций; регистры процессора делятся на два типа: регистры общего назначения и специальные регистры. Для того, чтобы писать программы на ассемблере, необходимо знать, какие регистры процессора существуют и как их можно использовать. Большинство команд в программах написанных на ас- семблере используют регистры в качестве операндов. Практически все команды представляют собой преобразование данных хранящихся в регистрах процессора, это например пересылка данных между регистрами или между регистрами и памятью, пре- образование (арифметические или логические операции) данных хранящихся в регистрах.

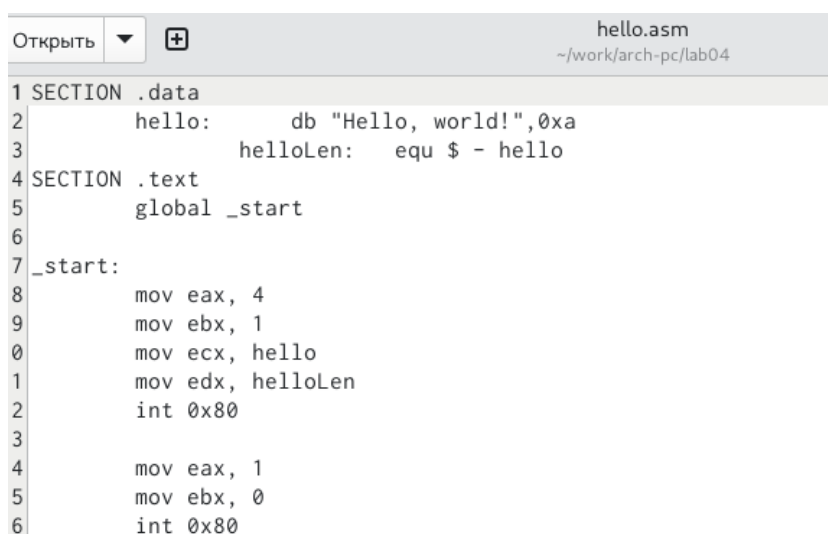
4 Выполнение лабораторной работы

1. Создаю и открываю текстовый файл для дальнейшей работы. (рис. 4.1)

```
agovchinnikov@dk8n81 ~ $ mkdir -p ~/work/arch-pc/lab04
agovchinnikov@dk8n81 ~ $ cd ~/work/arch-pc/lab04
agovchinnikov@dk8n81 ~/work/arch-pc/lab04 $ gedit hello.asm
```

Рис. 4.1: Создание файла

2. Ввожу в файл следующий текст. (рис. 4.2)



```
Открыть ▼ + hello.asm
~/work/arch-pc/lab04
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
0     mov ecx, hello
1     mov edx, helloLen
2     int 0x80
3
4     mov eax, 1
5     mov ebx, 0
6     int 0x80
```

Рис. 4.2: Введенный текст

3. Компилирую исходный файл hello.asm в obj.o. (рис. 4.3)

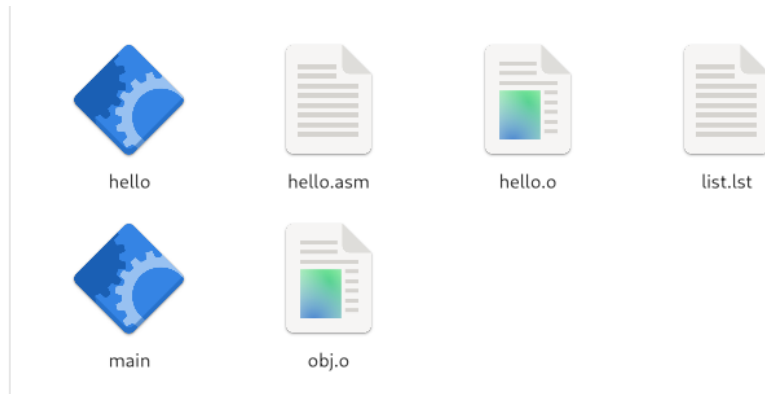


Рис. 4.3: Созданные файлы

4. Запускаю на выполнение созданный исполняемый файл. (рис. 4.4)

```
agovchinnikov@dk8n81 ~ $ cd ~/work/arch-pc/lab04
agovchinnikov@dk8n81 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
agovchinnikov@dk8n81 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
agovchinnikov@dk8n81 ~/work/arch-pc/lab04 $ ld -m elf_i386 hello.o -o hello
agovchinnikov@dk8n81 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main
agovchinnikov@dk8n81 ~/work/arch-pc/lab04 $ ./hello
Hello, world!
agovchinnikov@dk8n81 ~/work/arch-pc/lab04 $
```

Рис. 4.4: Выполнение нужных для компоновки команд

5 Выполнение заданий для самостоятельной работы

1. Создаю копию файла с другим именем и вывожу на экран свою имя и фамилию, после выполняю компоновку файла. (рис. 5.1)

```
agovchinnikov@dk1n22 ~ $ cd ~/work/arch-pc/lab04
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $ cp hello.asm lab4.asm
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $ gedit lab4.asm
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $ nasm -f elf lab4.asm
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst lab4.asm
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $ ld -m elf_i386 lab4.o -o hello
ld: не распознан режим эмуляции: elf_i386
Поддерживаемые эмуляции: elf_x86_64 elf32_x86_64 elf_i386 elf_iamcu
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $ ld -m elf_i386 lab4.o -o hello
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $ ./hello
Антон Овчинников
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $
```

Рис. 5.1: Компоновка объектного файла и его запуск

2. Копирую файлы hello.asm и lab4.asm в мой локальный репозиторий. (рис. 5.2)

```
agovchinnikov@dk1n22 ~ $ cd work/arch-pc
agovchinnikov@dk1n22 ~/work/arch-pc $ cd lab04
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $ mv hello.asm lab4.asm ~/work/study/"2023-
2024-Архитектура компьютера"/arch-pc/labs/lab04
agovchinnikov@dk1n22 ~/work/arch-pc/lab04 $
```

Рис. 5.2: Перемещение файлов

3. Загружаю файлы на github. (рис. 5.3)

Name	Last commit message	Last commit date
..		
bib	feat(main): make course structure	3 weeks ago
image	feat(main): make course structure	3 weeks ago
pandoc	feat(main): make course structure	3 weeks ago
Makefile	feat(main): make course structure	3 weeks ago
hello.asm	Add files via upload	now
lab4.asm	Add files via upload	now
report.md	feat(main): make course structure	3 weeks ago

Рис. 5.3: Проверка файлов

6 Листинги

```
SECTION .data
    hello:      db "Hello, world!",0xa
    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

SECTION .data
    hello:      db "Антон Овчинников",0xa
    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
```

7 Выводы

Я освоил процедуры компиляции и сборки программ, написанных на ассемблере NASM.