Лабораторная работа №4

Создание и процесс обработки программ на языке ассемблера NASM

Селиванов Вячеслав Алексеевич

Содержание

# 1 Цель работы

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

# 2 Задание

Создание программы Hello world!  
Работа с транслятором NASM  
Работа с расширенным синтаксисом командной строки NASM  
Работа с компоновщиком LD  
Запуск исполняемого файла  
Выполнение заданий для самостоятельной работы.

# 3 Теоретическое введение

Основными функциональными элементами любой электронно-вычислительной машины (ЭВМ) являются центральный процессор, память и периферийные устройства (рис. 4.1). Взаимодействие этих устройств осуществляется через общую шину, к которой они подклю- чены. Физически шина представляет собой большое количество проводников, соединяющих устройства друг с другом. В современных компьютерах проводники выполнены в виде элек- тропроводящих дорожек на материнской (системной) плате. Основной задачей процессора является обработка информации, а также организация координации всех узлов компьютера. В состав центрального процессора (ЦП) входят следующие устройства: • арифметико-логическое устройство (АЛУ) — выполняет логические и арифметиче- ские действия, необходимые для обработки информации, хранящейся в памяти; • устройство управления (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.

# 4 Выполнение лабораторной работы

## 4.1 Создание программы Hello world!

Создаю пустой текстовый файл hello.asm(рис. [1](#fig:001)).

Figure 1: Перемещение по директориям и создание файла

Figure 1: Перемещение по директориям и создание файла

Открываю файл в текстовом редакторе gedit(рис. [2](#fig:002)).

Figure 2: Файл hello,asm

Figure 2: Файл hello,asm

Заполняю файл, заполняю программу для вывода Hello world (рис. ??).

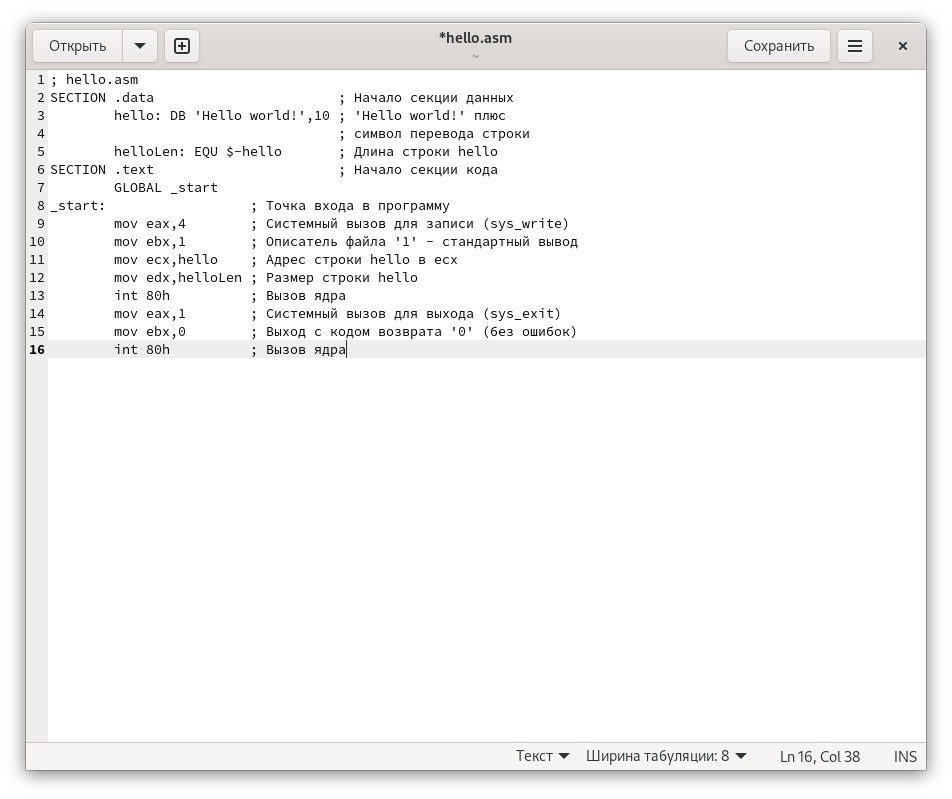


Figure 3: Программа

## 4.2 Работа с транслятором NASM.

Превращаю текст программы для вывода “Hello world!” в объектный код с помощью транслятора NASM, используя команду nasm -f elf hello.asm, ключ -f указывает транслятору nasm, что требуется создать бинарный файл в формате ELF и с помощью утилиты ls проверяю создан ли файл hello.o (рис. [4](#fig:004)).

Figure 4: Создание бинарного файла

Figure 4: Создание бинарного файла

C помощью утилиты ls проверяю создан ли файл hello.o (рис. [5](#fig:005)).

Figure 5: Создание бинарного файла

Figure 5: Создание бинарного файла

##Работа с расширенным синтаксисом командной строки NASM. Ввожу команду, которая скомпилирует файл hello.asm в файл obj.o, при этом в файл будут включены символы для отладки (ключ -g), также с помощью ключа -l будет создан файл листинга list.lst. (рис. [6](#fig:006)).

Figure 6: Создание файла листинга

Figure 6: Создание файла листинга

C помощью утилиты ls проверяю создан ли файл obj.o и list.lst (рис. [7](#fig:007)).

Figure 7: Создание бинарного файла

Figure 7: Создание бинарного файла

## 4.3 Работа с компоновщиком LD.

Передаю объектный файл hello.o на обработку компоновщику LD, чтобы получить исполняемый файл hello. Ключ -о задает имя создаваемого исполняемого файла. Далее проверяю с помощью утилиты ls правильность выполнения команды. (рис. [8](#fig:008)).

Figure 8: Получение исполняемого файла

Figure 8: Получение исполняемого файла

Выполняю следующую команду Исполняемый файл будет иметь имя main, т.к. после ключа -о было задано значение main. Объектный файл, из которого собран этот исполняемый файл, имеет имя obj.o. (рис. [9](#fig:009)).

Figure 9: Значение main

Figure 9: Значение main

## 4.4 Запуск исполняемого файла.

Запускаю созданный файл, чтобы проверить программу. (рис. [10](#fig:010)).

Figure 10: Проверка программы

Figure 10: Проверка программы

## 4.5 Выполнение заданий для самостоятельной работы.

Создаю копию файла, называю его lab04.asm. (рис. [11](#fig:011))

Figure 11: Копирование файла hello.asm

Figure 11: Копирование файла hello.asm

С помощью текстового редактора gedit открываю файл lab04.asm и меняю программу так, чтобы она выводила мои имя и фамилию.(рис. [12](#fig:012))

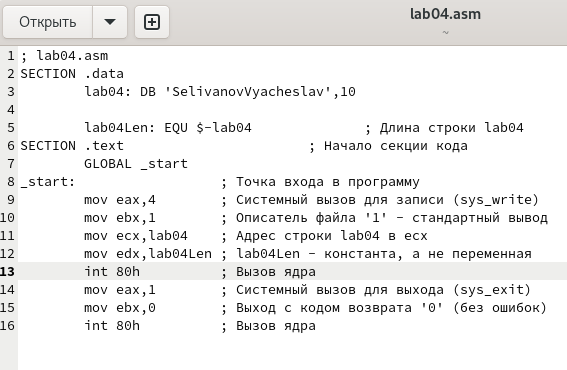


Figure 12: Изменение программы

Компилирую текст программы в объектный файл.Проверяю с помощью утилиты ls, что файл lab04.o создан.(рис. [11](#fig:011))

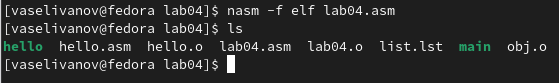


Figure 13: Создание объектного файла

Передаю объектный файл lab04.o на обработку компоновщику LD, чтобы получить исполняемый файл lab04.(рис. [14](#fig:014))

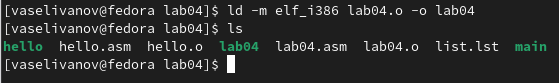


Figure 14: Получение исполняемого файла

Запускаю исполняемый файл lab04, на экран действительно выводятся мои имя и фамилия.(рис. [15](#fig:015))

Figure 15: Проверка программы

Figure 15: Проверка программы

Скидываю файлы hello.asm и lab04.asm в каталог 4 лабораторный работы.(рис. [16](#fig:016))

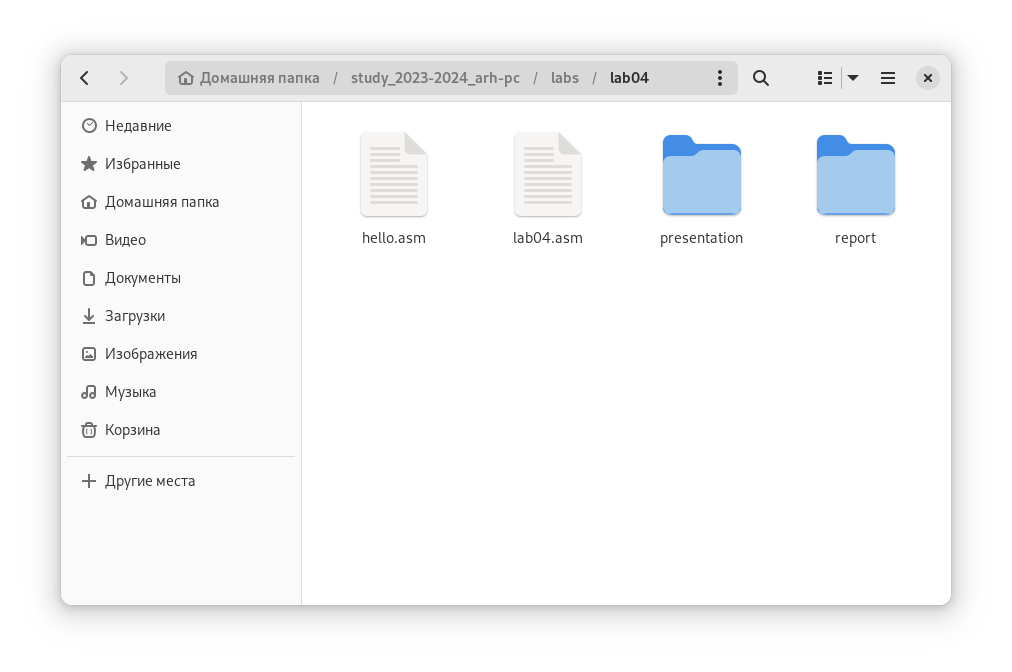


Figure 16: Добавление файлов в лабораторную работу

Отправление всех изменений на github. (рис. [17](#fig:017))

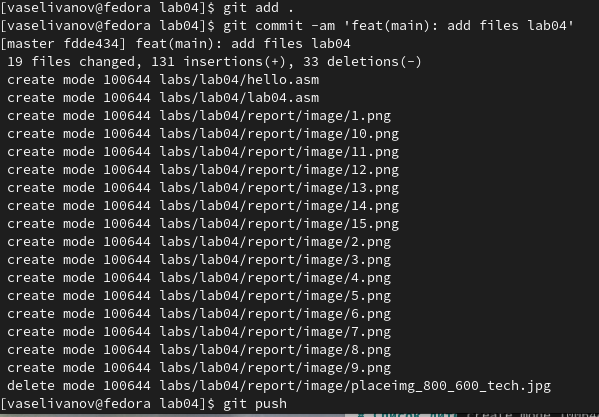


Figure 17: Отправление работы на github

# 5 Выводы

При выполнении данной лабораторной работы я освоила процедуры компиляции и сборки программ, написанных на ассемблере NASM.

# Список литературы

(1Архитектура ЭВМ)