Отчёта по лабораторной работе №5

Создание и процесс обработки программ на языке ассемблера NASM

Акопян Сатеник Манвеловна

Содержание

# 1 Цель работы

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

# 2 Теоретическое введение

Основными функциональными элементами любой электронно-вычислительной машины (ЭВМ) являются центральный процессор, память и периферийные устройства

Взаимодействие этих устройств осуществляется через общую шину, к которой они подключены. Физически шина представляет собой большое количество про- водников, соединяющих устройства друг с другом. В современных компьютерах проводники выполнены в виде электропроводящих дорожек на материнской (системной) плате.

Язык ассемблера (assembly language, сокращённо asm) — машинно- ориентированный язык низкого уровня. Можно считать, что он больше любых других языков приближен к архитектуре ЭВМ и её аппаратным возможностям, что позволяет получить к ним более полный доступ, нежели в языках высокого уровня, таких как C/C++, Perl, Python и пр. Заметим, что получить полный доступ к ресурсам компьютера в современных архитектурах нельзя, самым низким уровнем работы прикладной программы является обращение напрямую к ядру операционной системы. Именно на этом уровне и работают программы, напи- санные на ассемблере. Но в отличие от языков высокого уровня ассемблерная программа содержит только тот код, который ввёл программист. Таким образом язык ассемблера — это язык, с помощью которого понятным для человека образом пишутся команды для процессора.

# 3 Выполнение лабораторной работы

1. Создаём текстовый файл с именем hello.asm (рис. 1)

Рис. 1: рисунок 1

Рис. 1: рисунок 1

1. Открываем этот файл с помощью gedit (рис. 2)

Рис. 2: рисунок 2

Рис. 2: рисунок 2

1. Вводим в него текст, данный в лабораторной работе (рис. 3)



Рис. 3: рисунок 3

1. Компилируем текст программы “Hello world” (рис. 4)

Рис. 4: рисунок 4

Рис. 4: рисунок 4

1. С помощью команды ls проверяем, что объектный файл был создан (рис. 5). Объектный файл имеет имя hello.o

Рис. 5: рисунок 5

Рис. 5: рисунок 5

1. Скомпилируем исходный файл hello.asm в obj.o и с помощью команды ls проверяем, что файлы были созданы.(рис. 6)

Рис. 6: рисунок 6

Рис. 6: рисунок 6

1. Далее чтобы получить исполняемую программу, объектный файл необходимо передать на обработку компоновщику,с помощью команды ls проверяем что файл hello был создан. (рис. 7)

Рис. 7: рисунок 7

Рис. 7: рисунок 7

1. Выполняем следующую команду, которая создаёт файл main (рис. 8)

Рис. 8: рисунок 8

Рис. 8: рисунок 8

1. Запускаем на выполнение созданный исполняемый файл, находящийся в текущем каталоге (рис. 9)

Рис. 9: рисунок 9

Рис. 9: рисунок 9

# 4 Задание для самостоятельной работы

1. В каталоге ~/work/arch-pc/lab05 с помощью команды cp создаем копию файла hello.asm с именем lab5.asm (рис. 10)

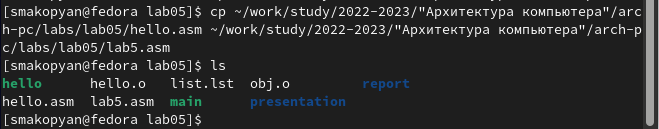


Рис. 10: рисунок 10

1. С помощью gedit вносим изменения в текст программы в файле lab5.asm так, чтобы вместо Hello world! на экран выводилась строка с моей фамилией и именем. (рис. 11)



Рис. 11: рисунок 11

1. Оттранслируем полученный текст программы lab5.asm в объектный файл. (рис. 12) (рис. 13)

Рис. 12: рисунок 12

Рис. 12: рисунок 12

Рис. 13: рисунок 13

Рис. 13: рисунок 13

1. Выполняем компоновку объектного файла и запустите получившийся исполняемый файл. (рис. 14) (рис. 15) (рис. 16)

Рис. 14: рисунок 14

Рис. 14: рисунок 14

Рис. 15: рисунок 15

Рис. 15: рисунок 15

Рис. 16: рисунок 16

Рис. 16: рисунок 16

1. Копируем файлы hello.asm и lab5.asm в локальный репозиторий в каталог ~/work/study 2022-2023/“Архитектура компьютера”/arch-pc/labs/lab05/. Загружаем файлы на github (рис. 17)

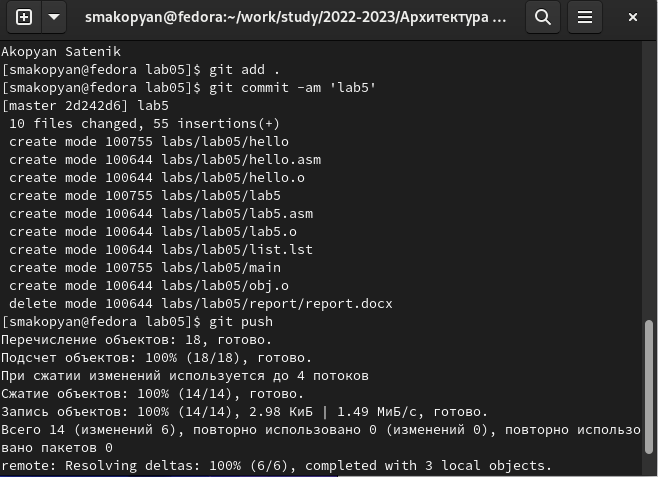


Рис. 17: рисунок 17

# 5 Выводы

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

# Список литературы