МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ В.Ф. УТКИНА»

КАФЕДРА «ЭВМ»

Отчет по лабораторной работе №1

по дисциплине

«Машинно-зависимые языки программирования»

Выполнили:

Студенты группы 945

Комлева Екатерина

Соболева Елизавета

Горбачева Дарья

Проверили:

доц., к.т.н. Муратов Е.Р.

асс. каф. ЭВМ Тарасов А.С.

Рязань 2021

**Тема лабораторной работы:** Основы программирование на ассемблере

**Цель работы:** получить сведения об особенностях языка программирования, ознакомиться с инструментами разработки и их настройке.

**ХОД РАБОТЫ**

1. **MASM**

Создадим программу prog11.asm, которая ничего не делает.

.586P

.model flat, stdcall

\_data segment

\_data ends

\_text segment

start:

ret

\_text ends

end start

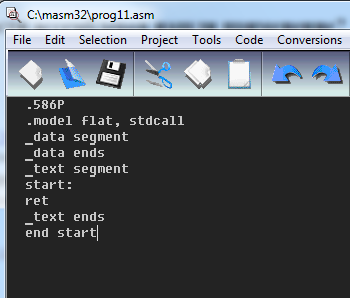


Рисунок 1 – Создание проекта prog11 и написание программы.

Произведем ассемблирование (трансляцию) файла prog11.asm, используя установленный ассемблер:

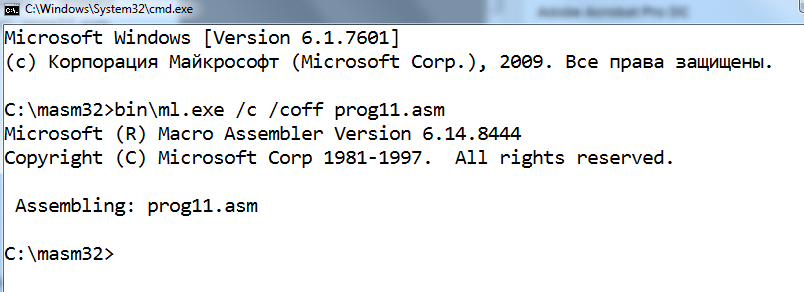


Рисунок 2 – Реализация заданной программы

1. **Работа с MASM в Visual studio**

Для создания нового консольного приложения запускаем Microsoft Visual Studio 2010 Express и переходим в меню Файл->Создать->Проект. В появившемся окне выбираем Консольное приложение Win32 и задаем имя проекта и нажимаем кнопку OK. После того, как в Visual Studio появилось окно проекта (в левой части появившегося окна отображается Обозреватель решений), для добавления нового файла программы в проект выбираем по правой кнопке мыши на папке Файлы исходного кода меню Добавить->Создать элемент. В появившемся окне выбираем Файл C++ (.cpp), задаем имя файла и вручную добавляем к нему расширение asm. Нажимаем кнопку Добавить. В появившемся окне набираем текст программы. В качестве примера можно использовать следующий текст:

.686P

.MODEL FLAT, STDCALL

.STACK 4096

.DATA

MB\_OK EQU 0

STR1 DB “Моя первая программа”,0

STR2 DB “Привет всем!”,0

HW DD ?

EXTERN MessageBoxA@16:NEAR

.CODE

START:

PUSH MB\_OK

PUSH OFFSET STR1

PUSH OFFSET STR2

PUSH HW

CALL MessageBoxA@16

RET

END START

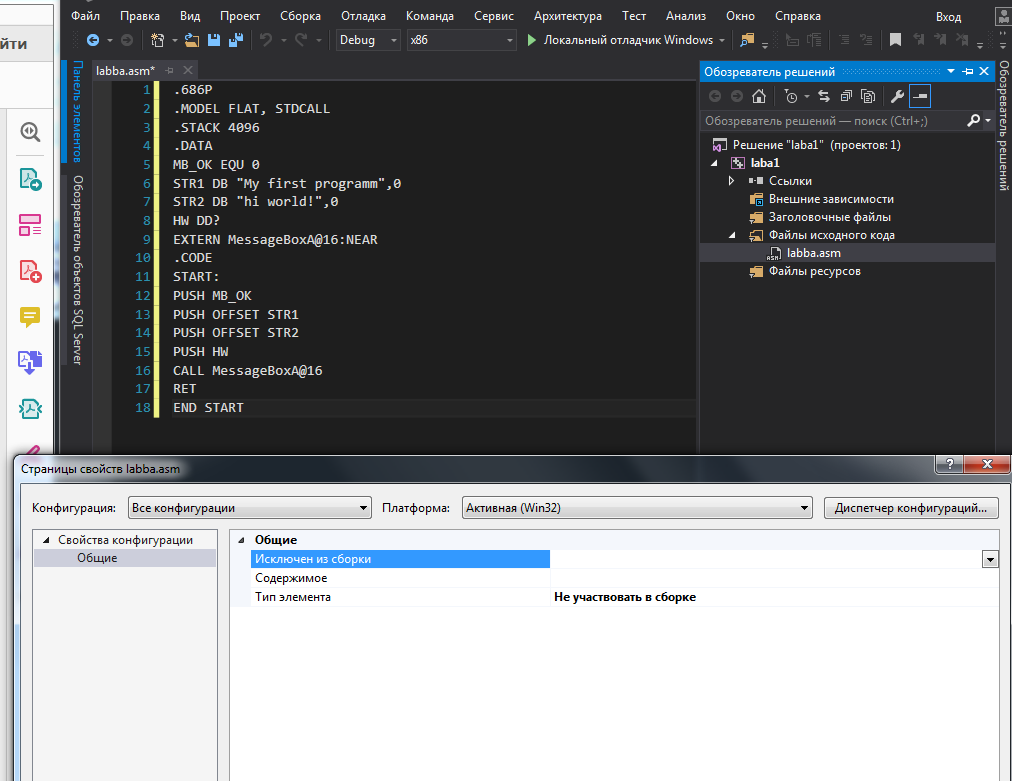


Рисунок 3 – Создание нового проекта и написание заданной программы

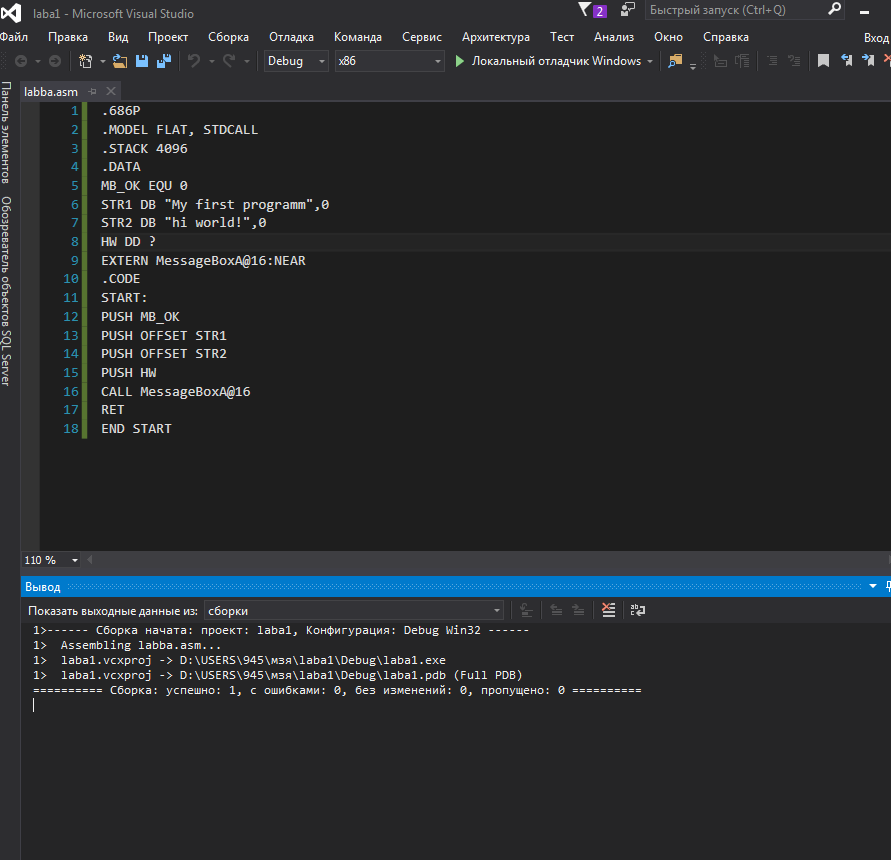


Рисунок 4 – Проверка соответствующих инструментов сборки

В появившемся окне ставим галочку для masm (Microsoft Macro Assembler) и нажимаем OK. Теперь нужно проверить, что для файла на языке ассемблера установился соответствующий инструмент сборки. По правой кнопке мыши для файла с расширением .asm выбираем опцию Свойства. В появившемся окне для выбранного файла отмечаем инструмент сборки Microsoft Macro Assembler. Для построения проекта выбираем меню Отладка->Построить решение. Результат выполнения программы:

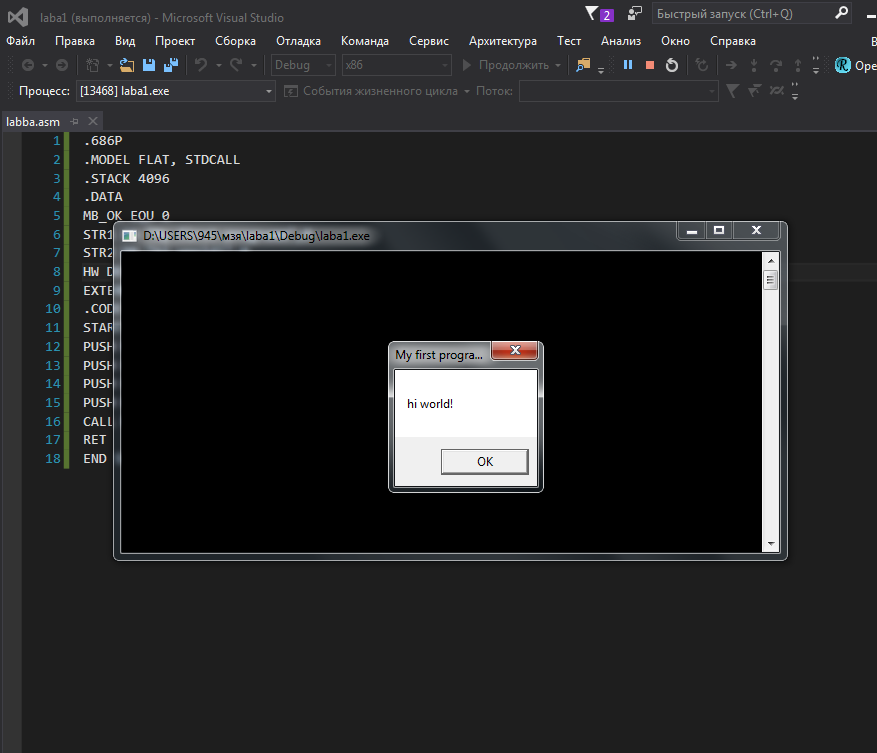


Рисунок 5 - Результат выполнения программы

1. **Изменить тип приложения с консольного на оконное**

Чтобы убрать консоль (поменять тип приложения с консольного на оконное, или наоборот) необходимо обратиться к меню Свойства проекта, вызванного по правой кнопке мыши. В появившемся окне выбрать раздел Компоновщик->Система, и в разделе Подсистема поменять тип с Консоль на Windows (или наоборот). Повторная сборка и запуск программы на выполнения выдадут следующий результат (консоли нет):

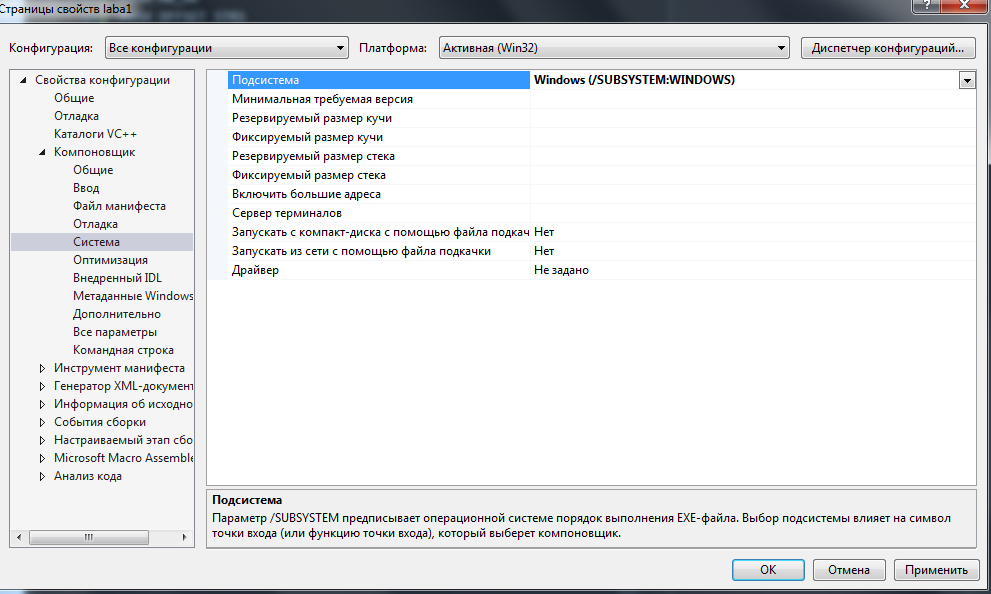


Рисунок 6 – Смена типа с Консоль на Windows

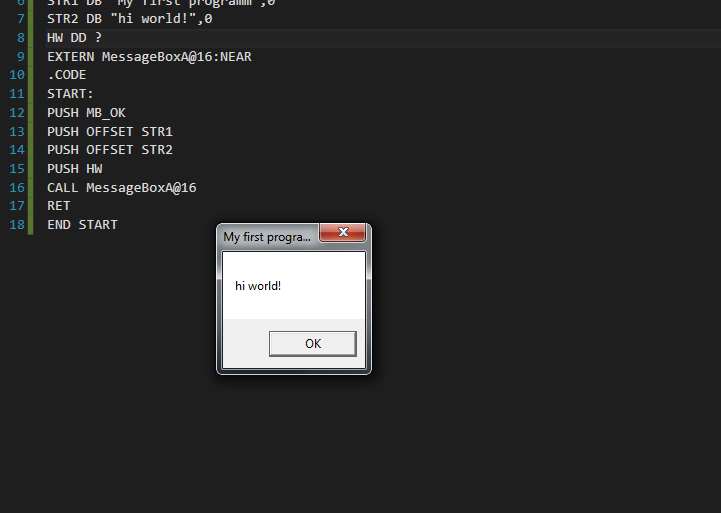


Рисунок 7 – Результат выполнения программы

1. **Подсветка синтаксиса языка ассемблера**

Для того, чтобы включить подсветку синтаксиса языка ассемблера в Microsoft Visual Studio Express 2010 необходимо загрузить файл usertype и распаковать его в папку C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE Для подключения подсветки синтаксиса выбираем меню Сервис->Параметры. В появившемся окне выбрать Текстовый редактор->Расширение файла и вручную добавляем расширение asm. Нажимаем кнопку Добавить, затем — OK. После перезапуска Microsoft Visual Studio Express 2010 подсветка синтаксиса языка ассемблера будет активна.

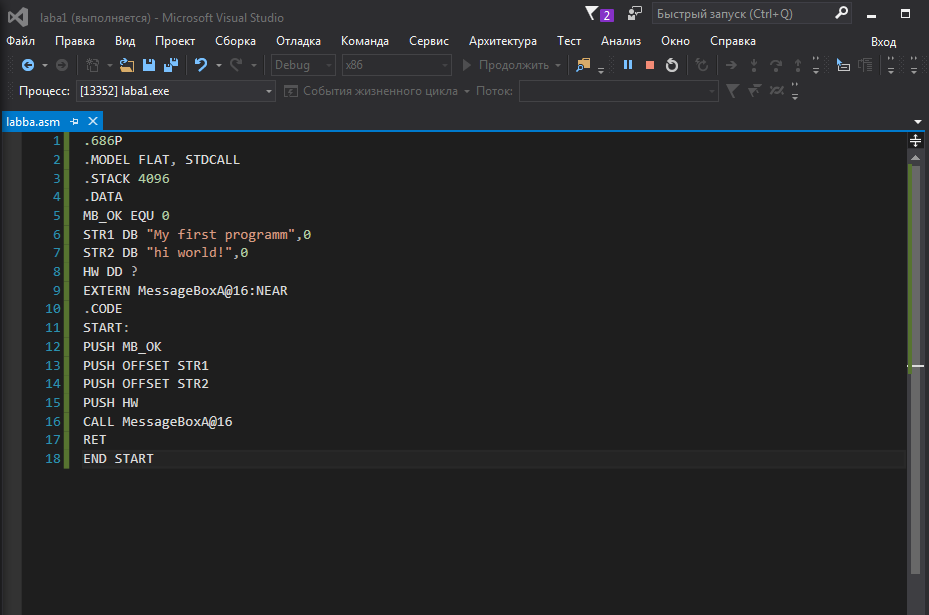


Рисунок 8 – Результат выполнения задания

**Вывод:** В данной лабораторной работе мы получили сведения об особенностях языка программирования, ознакомились с инструментами разработки и их настройке.