**Цель:** изучить правила записи команд на языке Ассемблер, правила трансляции и редактирования программ, получить навыки работы в отладчике.

**Краткие теоретические сведения:**

Программа на языке Ассемблера называется исходной программой (исходным модулем). Ассемблерный транслятор (tasm) преобразует исходный текст в машинный код (объектную программу). И, наконец, программа–компоновщик (tlink) определяет все адресные ссылки для, возможно, нескольких объектных модулей, генерируя загрузочный модуль.

Основной формат кодирования команд (операторов) Ассемблера имеет следующий вид:

[метка] команда [операнды] [;комментарий]

Метка, команда и операнды разделяются по крайней мере одним пробелом или символом табуляции. Максимальная длина строки – 132 символа.

Метка в языке Ассемблера может содержать буквы от "A" до "Z" и от "a" до "z" и цифры от "0" до "9" , а также специальные символы "@", "\_", "$". Первым символом должна быть буква или специальный символ. Максимальная длина метки – 31 символ.

Мнемоническая команда указывает Ассемблеру, какое действие должен выполнять данный оператор.

Операнд может определять либо начальное значение данных, либо элементы, над которыми выполняются действия по соответствующей команде.

**Задание на лабораторную работу:**

1) Просмотреть файл **example1.asm**, содержащий пример программы на языке Ассемблера. Изучить назначение команд программы.

2) Оттранслировать программу и получить исполняемую программу в формате СОМ.

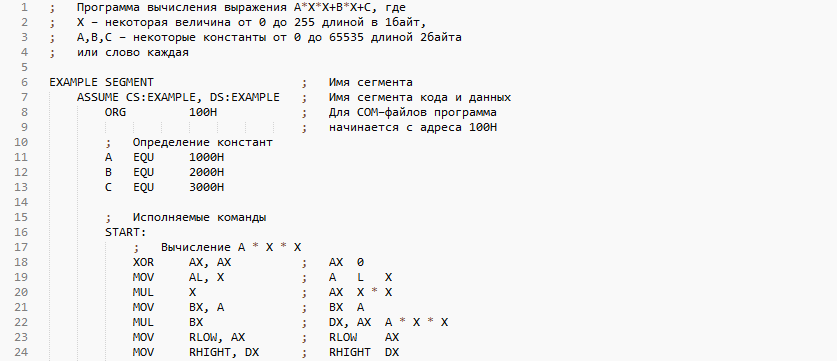
3) Выполнить программу в пошаговом режиме с помощью отладчика td.

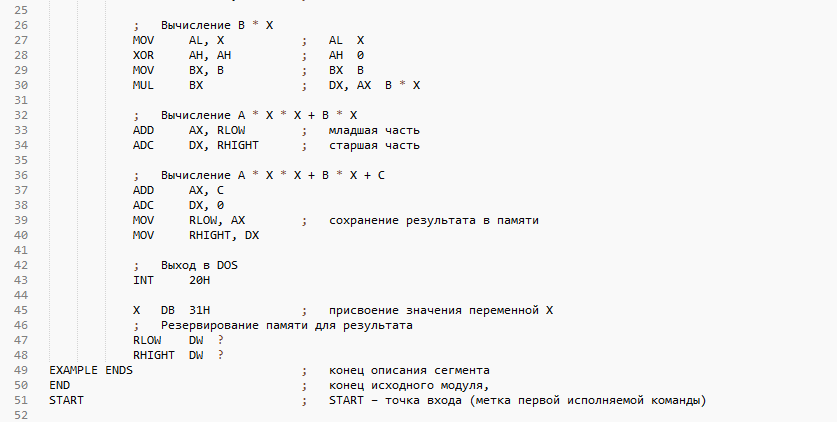
4) Проанализировать состояния регистров и памяти после выполнения каждой команды.

**Ход работы:**

Необходимо изучить тестовый пример программы на Ассемблере, создать исполняемый файл и выполнить с помощью отладчика в пошаговом режиме. Также необходимо изучить назначение команд в тестовом примере.

Текст программы:





Назначение команд:

SEGMENT – директива для описания сегментов (кода, данных, стека)

ASSUME – директива для определения назначения каждого сегмента

ORG – директива относительной позиции в сегменте

END – директива выхода из программы

EQU – присвоить

XOR – исключающее ИЛИ

MOV – переместить

MUL – умножить

ADD – сложить

ADC –

INT –

DW – директива определения переменной длиной в слово

START – директива точки входа

Отладка:

Окно отладчика представлено на рисунке 1.

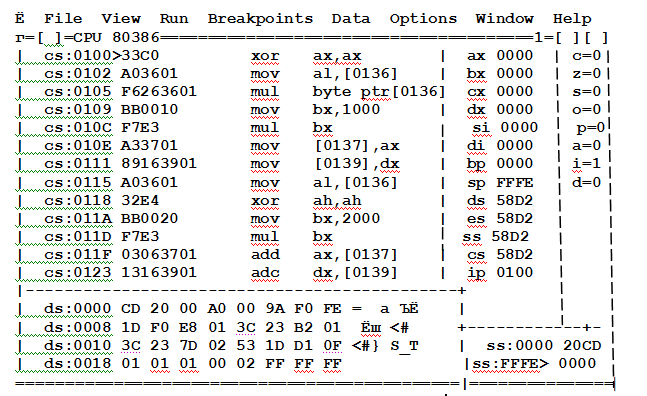
****

Рисунок 1 – Окно отладчика