МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ АВТОМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

КАФЕДРА РАДИОЭЛЕКТРОННЫХ СРЕДСТВ

Отчет по дисциплине

«Цифровые устройства и микропроцессоры»

**«СИСТЕМА КОМАНД МИКРОПРОЦЕССОРА X86»**

Вариант 8

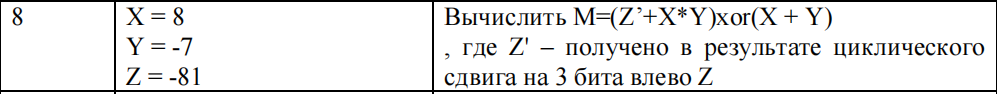
Выполнил: студент группы ИКТб – 3301\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Д. И. Касимов

Проверил: к.т.н. доцент кафедры РЭС \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_М. А. Земцов

Киров 2023

Цель работы: изучение системы команд и способов адресациимикропроцессоров с архитектурой x86.

Исходные данные:



Ход работы:

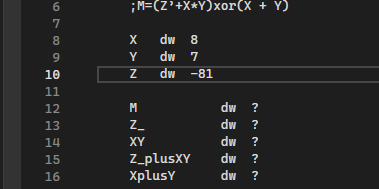


Рисунок 1 - Создание переменных типа dw (16 бит) без последующей инициализации

Примечание: переменной Y присовено значение 7 (по заданию -7) не ошибочно. В блоке выполнения переменной Y будет присовено значение 0-7.

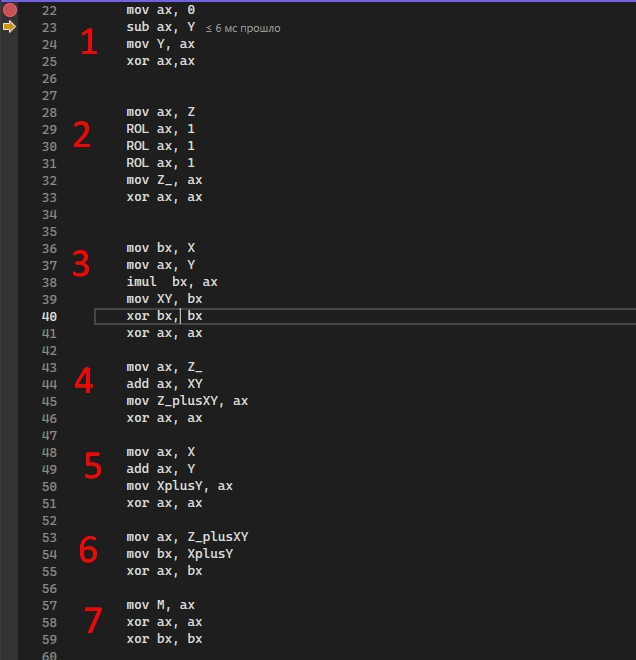


Рисунок 2 - Логика выполнения работы программы

Описание блоков кода:

1. В регистр ax помещаем 0. Из регистра ax отнимаем Y помещаем результат в регистр ax. Чистим регистр.
2. В регистр ax помещаем Z. Последущие 3 строки отвечают за циклический сдвиг на 1 бит влево (3 строки 3 битового сдивга влево). ПОмещаем результат из регистра ax в переменную Z\_. Чистим регистр ax.
3. В регистр bx помещаем X. В регистр ax помещаем Y. Используется умножение со знаком, перемножаются данные в регистрах ax и bx с учетом знака, результат помещается в регистр bx. Помещаем данные в переменную XY из регистра bx. Чистим регистры bx и ax.
4. В регистр ax помещаем значение переменной Z\_. К данным в регистре ax прибавляем XY результат запишется в регистр ax. Данные из регистра ax помещаем в Z\_plusXY. Чистим регистр ax.
5. В регистр ax помещаем значение из переменной X. К данным в регистре ax прибавляем данные из переменной Y. Данные из регистра ax помещаются в XplusY. Чистим регистр ax.
6. В регистр ax помещаем данные из переменной Z\_plusXY. Помещаем в регистр bx данные из XplusY. Затем данные из регистров ax и bx складываем по модулю 2 (копирование данных в регистр необходимо, потому как напрямую нельзя оперировать данными в переменных), результат запишется в регистр ax.
7. Помещаем в переменную M данные из регистра ax. Чистим регистры ax и bx.

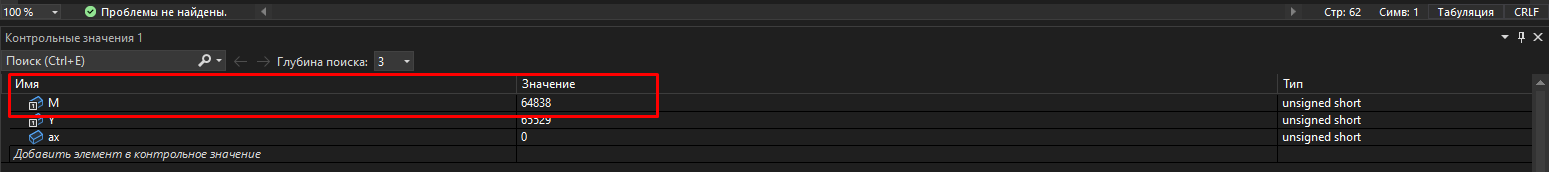


Рисунок 3 - Результат выполнения работы программы.

Полученный результат можно объяснить тем, что процесссор, которому мы задаем команды, «не знает» об отрицательных числах. В двоичной форме такому числу в старший разряд приписывается единица (для обозначения минуса), затем проводятся операции с дополнительным кодом и в итоге данные интерпретируется в окне контрольных значений в непонятной для пользователя форме.

**Вывод:** были изучены системы команд и способы адресации микропроцессоров с архитектурой x86.