

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

кафедра «Информационные системы»

Отчёт
по лабораторной работе №3
по дисциплине «Технические средства информационных систем»

Выполнил:
ст.гр. ИС/б-20-2-о

Филозоф А.Н.

Принял:
Минкин С.И.

Севастополь

2022 г.

ИССЛЕДОВАНИЕ МЕТОДОВ АДРЕСАЦИИ И ПРОГРАММИРОВАНИЯ АРИФМЕТИЧЕСКИХ И ЛОГИЧЕСКИХ ОПЕРАЦИЙ

3.1. Цель работы

Изучить основные директивы языка ассемблера, исследовать их воздействие на процесс ассемблирования и формирования листинга программы.

Исследовать особенности функционирования блоков 16-разрядного микропроцессора при выполнении арифметических и логических операций и при использовании различных способов адресации. Приобрести практические навыки программирования на языке ассемблера МП 8086 арифметических и логических операций с применением различных способов адресации.

3.2. Постановка задачи

Составит программу в com и exe форматах, осуществляющей вычисление выражения (1).

$$X = 3 \cdot (C - 2 \cdot A) + (B - C + 1) / 2 \quad (1)$$

Произвести отладку разработанных программ в пошаговом режиме и проследить за изменениями содержимого регистров.

3.3. Ход выполнения работы

В листинге 1 представлен код программы, вычисляющей значение функции по варианту.

Листинг 1 — Код программы

```

org 100h

CALL main
RET

.code
main PROC
    MOV AX, 257                ; + 4
    MOV BX, 0                  ; + 4
    MOV CX, -257               ; + 4

    CALL f                     ; + 19
    RET
main ENDP

f PROC
    MOV f_c, CX                ; + 10
    SAL AX, 1 ; AX = 2 * a      ; + 2
    SUB CX, AX ; CX = c - 2 * a ; + 4
    MOV AX, CX ; move result to AX ; + 2
    ; AX = AX * CX, where CX = 3
    MOV CX, 3                  ; + 4
    IMUL CX                     ; + (128~154)
    SUB BX, f_c ; b -= c        ; + 16
    INC BX ; ++b                ; + 2
    SAR BX, 1 ; b /= 2          ; + 2
    ADD AX, BX                  ; + 3
    MOV BX, 0                   ; + 4
    MOV CX, 0                   ; + 4

    RET
f ENDP

.data
    f_c dw ?

```

Было проведено тестирование составленной программы. Входные тестовые данные и результаты тестов показаны в таблице 1.

Таблица 1 — Тестовые данные и результаты тестов

| A | B | C | Ожидаемое зн. | Полученное зн. |
|-----|-----|------|---------------|--------------------------|
| 10 | 100 | 10 | $000 F_{16}$ | $000 F_{16}$ |
| -1 | -20 | -30 | -79 | $FFB1_{16} = -79_{10}$ |
| 257 | 0 | -257 | -2184_{10} | $F778_{16} = -2184_{10}$ |

Время выполнения программы при n тактах: 216 — 242. Если частота тактового генератора — 5МГц, то время выполнения программы: $216 \cdot \frac{1}{5 \cdot 10^9} = 43,2 \cdot 10^{-9}$ с

или $242 \cdot \frac{1}{5 \cdot 10^9} = 48,4 \cdot 10^{-9}$ с.

Вывод

При выполнении данной работы было получено понимание работы 16-рядного процессора Intel8086. Также были получены навыки комментирования и вычисления времени выполнения программы, написанной на языке ассемблера.

Приложение А

Ответы на контрольные вопросы

1. Различие между командой и директивой состоит в том, что команды порождают одну машинную команду, а директивы содержат управляющую информацию для ассемблера.

2. Для определения данных используются директивы DB (определение переменной размером 1 байт), DW (определение переменной размером 1 машинное слово, или 2 байта), DD (определение переменной размером 2 слова, или 4 байта). Для инициализации нескольких ячеек используется DUP.

3. Для оформления процедур используются директивы PROC (начало процедуры) и ENDP (окончание процедуры). После PROC допустимо указание NEAR (код процедуры находится в том же сегменте, что и точки вызова процедуры) или FAR (код процедуры находится в другом сегменте).

4. Используются следующие типы сегментов: DATA (сегмент с данными), CODE (сегмент с кодом), STACK (сегмент стека).

5. Используются следующие типы выравнивания: PARA (адрес сегмента XXX0, где X – любое шестнадцатеричное число), PAGE (XX00), WORD (XXHE) и BYTE (XXXX).

6. Директива ORG позволяет переопределить счётчик команд ассемблера. Используется при написании программ EXE, где требуется 100_{16} отступ на от начала исполняемого файла.

7. Директива END для процедур сообщает о завершении кода процедуры; для сегмента — приостановить инкремент счётчика адресов относительно сегмента.

8. ...

9. Арифметические операции: сложение ADD, сложение с переносом ADC, инкремент данных INC, вычитание SUB, вычитание с заёмом SBB, декремент данных DEC, сравнение CMP и умножение MUL/IMUL.

10. ...

11. Благодаря директиве можно указать какому сегментному регистру сопоставлять какой сегмент.

12. В EXE-программах обычно определяется сегмент данных, а регистр DS инициализируется адресом этого сегмента; сегмент стека следует задавать самостоятельно. В COM-программах все данные должны быть определены в сегменте кода; стек генерируется автоматически.

13. ...

14. Назначение вывода— указание шине управления, осуществлять передачу данных между процессором и портом ввода (при 0) или между процессором и памятью (при 1). Соединён непосредственно с контроллером шины.

15. Заголовок EXE-файла имеет размер 512 байт. Заголовок хранится на диске в начале exe-файла. В заголовке содержится информация о размере выполняемого модуля, области загрузки в памяти, адрес стека и относительное смещение. В нем также указывается число байтов в последнем блоке exe-файла, число настраиваемых параметров, количество параграфов в заголовке и некоторые другие данные.

16. При запуске COM-программы DOS заносит в сегментные регистры адрес префикса программного сегмента (256 байт=100H), который резервируется DOS непосредственно перед COM- или EXE –программой в памяти. Так как адресация начинается со смещения 100H от начала префикса, то в программе после директивы SEGMENT следует вносить директиву ORG 100H.

17. Методы адресации: регистровая, непосредственная, прямая, косвенная регистровая, базовая, прямая с индексированием, базово-индексная.

18. В языке ассемблера строкой называется набор однотипных элементов (байтов, слов). Для обработки строк используются команды:

19. Для задания направления перемещения по строкам применяется флаг D, который устанавливается командой STD и сбрасывается командой CLD. При нулевом значении флага при каждом повторении происходит увеличение адресов источника и приемника, а при единичном – уменьшение.

20. Отличие команд и состоит в том, что позволяет сравнить содержимое двух областей памяти размером 1 байт или одно слово, а — содержимое регистра АХ и области памяти.

21. ...

22. Суть защищённого режима работы процессора состоит в защите кода текущей программы или пространства других программ от непреднамеренного чтения/записи. Защита осуществляется следующим образом: в сегментный регистр заносится не реальное значение сегмента, а селектор.

23. Дескриптор сегмента — структура данных, характеризующая размещение и длину используемого сегмента. Располагается в памяти ПЭВМ. Состоит из 4-х слов. Первое зарезервировано, третье содержит младшие разряды базы, четвёртое — размер сегмента. Второе слова содержит следующую информацию: биты «присутствие в основной памяти» Р, «обращение к сегменту» А и «системный сегмент» S; уровень привилегий дескриптора DPL; тип сегмента; старшие разряды базы.

24. Виртуальная память – это способ организации основной памяти большой емкости с помощью внешней памяти. Она позволяет при составлении программы распоряжаться всем пространством адресов, называемых виртуальными. Для поддержки виртуальной памяти используется дескриптор сегмента следующим образом: бит Р указывает при $P=1$, что данная страница находится в памяти и УУП осуществляет преобразование виртуального адреса в физический, иначе УУП передаёт ЦП сигнал отсутствия сегмента; в «Тип» указывается, разрешена ли запись в сегмент, тип сегмента.

25. Процессор второго поколения состоит из четырех блоков: адресного АУ, шинного ВУ, исполнительного ЕУ и командного ІУ, причем все блоки могут работать параллельно. Шинный блок осуществляет считывание памяти и портов ввода/вывода. Адресный блок вычисляет все адреса и формирует физические адреса. За счёт параллельной работы блоков возможно повышение производитель-

ности в 2-3 раза. Также процессор может работать в реальном и защищённом режимах.

26. ...

27. Сегмент состояния задачи состоит из 22-х слов. В нём хранятся состояния всех регистров, начальные значения указателей стеков уровней привилегий 0, 1 и 2, селектор локальной таблицы дескрипторов данной задачи. Совокупность ячеек ПС, ФР и РОН называют блоком управления задачей.