МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

федеральное государственное бюджетное образовательное учреждение высшего образования

**«Вологодский государственный университет»**

**Институт математики, естественных и компьютерных наук**

**Информатика и вычислительная техника**

**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

Архитектура ЭВМ и система команд

Дисциплина: «ЭВМ и периферийные устройства»

Направление подготовки: 09.03.01. Информатика и вычислительная техника

|  |  |
| --- | --- |
| Руководитель | Павлов В.В. |
| Выполнили студенты | Пчелкина О.С. |
| Группа, курс | ВМ-31 |
| Дата сдачи | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Дата защиты | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  *(подпись преподавателя)* |

Вологда

2022 г.

Задание:

1. Ознакомиться с архитектурой ЭВМ

2. Записать в ОЗУ "программу", состоящую из пяти команд варианта 8. Команды разместить в последовательных ячейках памяти.

3. При необходимости установить начальное значение в устройство ввода IR.

4. Определить те программно-доступные объекты ЭВМ, которые будут изменяться при выполнении этих команд.

5. Выполнить в режиме Шаг введенную последовательность команд, фиксируя изменения значений объектов

6. Если в программе образуется цикл, необходимо просмотреть не более двух повторений каждой команды, входящей в тело цикла.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | IR | Команда 1 | Команда 2 | Команда 3 | Команда 4 | Команда 5 |
| 6 | X | RD #4 | WR 11 | RD @11 | ADD #330 | JS 000 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Последовательность | Значения | | | | |
| Команды | RD #4 | WR 11 | RD @11 | ADD #330 | JS 000 |
| Коды | 211004 | 220011 | 212011 | 231330 | 130000 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| PC | Acc | M(11) | PC | Acc | M(11) |
| 000 | 000000 | 000000 | 004 | 129670 |  |
| 001 | 000004 |  | 000 |  |  |
| 002 |  | 000004 | 001 | 000004 |  |
| 003 | 130000 |  | 002 |  |  |

Ответ на контрольный вопрос:

8. Как записать программу в машинных кодах в память модели ЭВМ?

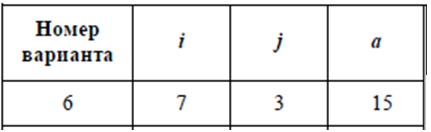
Для этого необходимо записать в текст программы коды, которые нам нужно использовать, нажать кнопку компиляции, либо через вспомогательное окно работа → компиляция. Введенные коды программа переведет в машинный код и запишет в память ЭВМ.

Можно вводить код программы сразу в ячейку памяти, он автоматически будет отображается в программе.

Задание:

Программирование разветвляющегося процесса

1. Разработать программу вычисления и вывода значения функции:





для вводимого из IR значения аргумента *x* . Функции и допустимые пре-

делы изменения аргумента приведены в табл. 10.6, варианты заданий —

в табл. 10.7.

2. Исходя из допустимых пределов изменения аргумента функций (табл. 10.6)

и значения параметра *a* для своего варианта задания (табл. 10.7) выделить

на числовой оси *Ox* области, в которых функция *y* вычисляется по пред-

ставленной в п. 1 формуле, и недопустимые значения аргумента. На недо-

пустимых значениях аргумента программа должна выдавать на OR мак-

симальное по модулю отрицательное число: 199 999.

3. Ввести текст программы в окно **Текст программы**, при этом возможен

набор и редактирование текста непосредственно в окне **Текст программы**

или загрузка текста из файла, подготовленного в другом редакторе.

4. Ассемблировать текст программы, при необходимости исправить синтак-

сические ошибки.

5. Отладить программу. Для этого:

а) записать в IR значение аргумента *x* *a* (в области допустимых значений);

б) записать в PC стартовый адрес программы;

в) проверить правильность выполнения программы (т. е. правильность ре-

зультата и адреса останова) в автоматическом режиме. В случае нали-

чия ошибки выполнить пп. 5, *г* и 5, *д*; иначе перейти к п. 5, *е*;

г) записать в PC стартовый адрес программы;

д) наблюдая выполнение программы в режиме **Шаг**, найти команду, яв-

ляющуюся причиной ошибки; исправить ее; выполнить пп. 5, *а* — 5, *в*;

е) записать в IR значение аргумента *x* *a* (в области допустимых значе-

ний); выполнить пп. 5, *б* и 5, *в*;

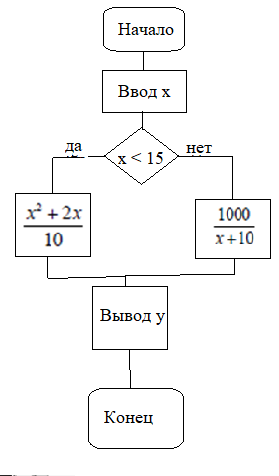
ж) записать в IR недопустимое значение аргумента *x* и выполнить пп. 5, *б*

и 5, *в*.

6. Для выбранного допустимого значения аргумента *x* наблюдать выполне-

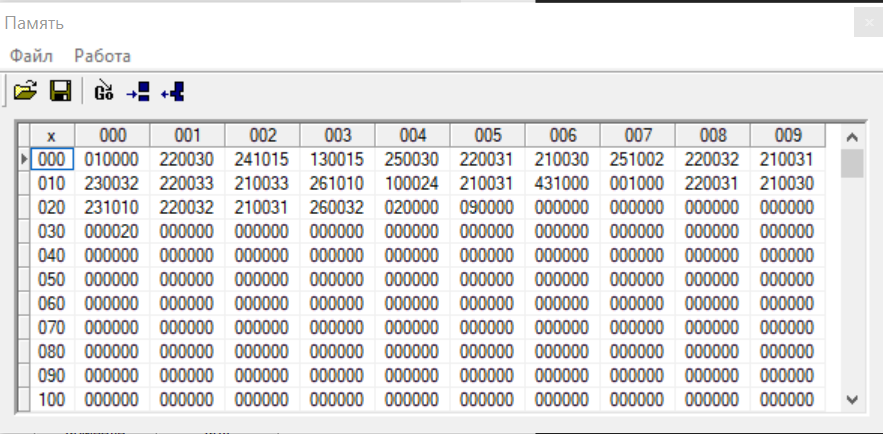
ние отлаженной программы в режиме **Шаг** и записать в форме табл. 10.3

содержимое регистров ЭВМ перед выполнением каждой команды.

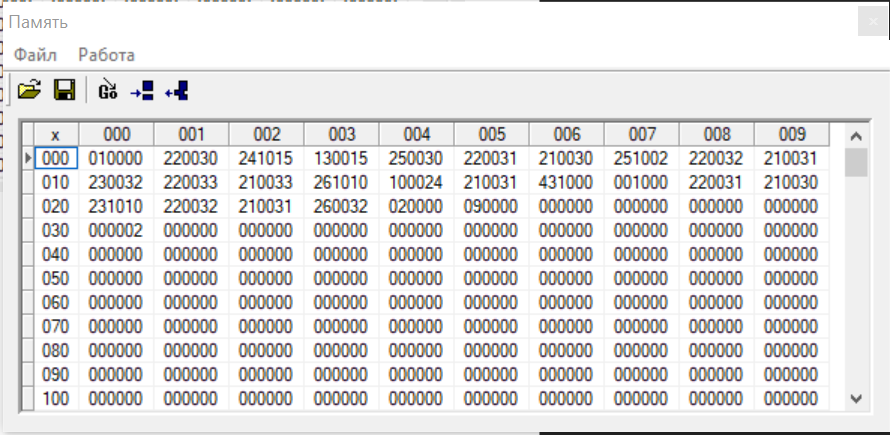


Размещение данных в ОЗУ

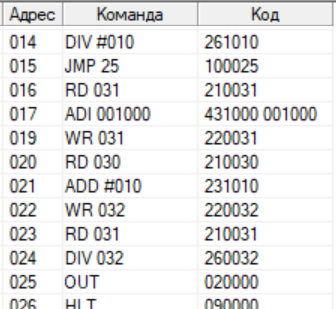
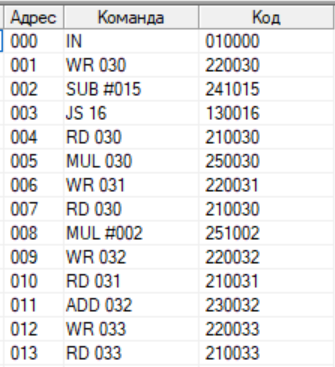
Если х>15



Если х<15



Команда и код, соответствующий ей.



Последовательность состояния регистров ЭВМ

Если х>15 Если х<15

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| PC | ACC | IR | OR | PC | ACC | IR | OR |
| 000 | 000000 | 000020 | 000000 | 000 | 000000 | 000002 | 000000 |
| 001 | 000020 | 000020 | 000000 | 001 | 000002 | 000002 | 000000 |
| 002 | 000020 | 000020 | 000000 | 002 | 000002 | 000002 | 000000 |
| 003 | 000005 | 000020 | 000000 | 003 | 100013 | 000002 | 000000 |
| 004 | 000005 | 000020 | 000000 | 016 | 100013 | 000002 | 000000 |
| 005 | 000020 | 000020 | 000000 | 017 | 000000 | 000002 | 000000 |
| 006 | 000400 | 000020 | 000000 | 018 | 001000 | 000002 | 000000 |
| 007 | 000400 | 000020 | 000000 | 019 | 001000 | 000002 | 000000 |
| 008 | 000020 | 000020 | 000000 | 020 | 000002 | 000002 | 000000 |
| 009 | 000040 | 000020 | 000000 | 021 | 000002 | 000002 | 000000 |
| 010 | 000040 | 000020 | 000000 | 022 | 000012 | 000002 | 000000 |
| 011 | 000400 | 000020 | 000000 | 023 | 000012 | 000002 | 00000 |
| 012 | 000440 | 000020 | 000000 | 024 | 001000 | 000002 | 00000 |
| 013 | 000440 | 000020 | 000000 | 025 | 000083 | 000002 | 00000 |
| 014 | 000440 | 000020 | 000000 | 026 | 000083 | 000002 | 000083 |
| 015 | 000044 | 000020 | 000000 |  |  |  |  |
| 025 | 000044 | 000020 | 000000 |  |  |  |  |
| 026 | 000044 | 000020 | 000044 |  |  |  |  |

Вводим число 2 в регистр IR для расчёта значения функции

Результат выполнения в регистре OR число 83.

Вводим число 20 в регистр IR для расчёта значения функции

Результат выполнения в регистре OR число 44.

Вводим число 55 в регистр IR для расчёта значения функции

Результат выполнения в регистре OR число 313.

Код программы

IN

WR 30

SUB #15

JS 16

RD 30

MUL 30

WR 31

RD 30

MUL #2

WR 32

RD 31

ADD 32

WR 33

RD 33

DIV #10

JMP 25

RD 31

ADI #1000

WR 31

RD 30

ADD #10

WR 32

RD 31

DIV 32

OUT

HLT

Ответ на контрольный вопрос

3. Как работают команды передачи управления?

Выполнение программы, записанной в памяти ЭВМ, осуществляется последовательно по командам в порядке возрастания адресов команд или в порядке, определяемом командами передачи управления.

Команды передачи управления: безусловный и шесть условных переходов, вызов подпрограммы, возврат из подпрограммы, цикл, программное прерывание, возврат из прерывания;

Команда передачи управления JMP переводит работу программы с текущего адреса в котором находится команда, на адрес указанный в команде.

Команда передачи управления JS переводит работу программы с текущего адреса в котором находится команда, на адрес указанный в команде, проверяя условие отрицательной разности входящего значения.

Задание.

1. Написать программу определения заданной характеристики после-

довательности чисел C1 , C2 , ..., Cn . Варианты заданий приведены

в табл. 10.9.

2. Записать программу в мнемокодах, введя ее в поле окна Текст программы.

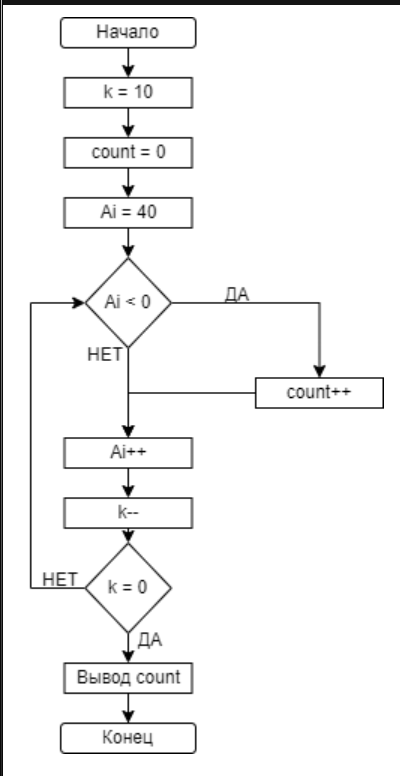
3. Сохранить набранную программу в виде текстового файла и произвести

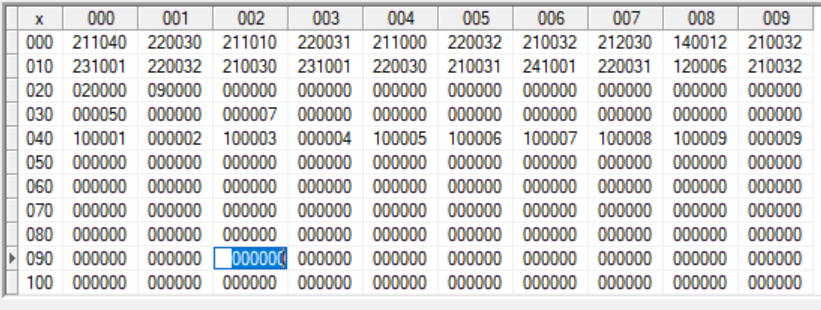
ассемблирование мнемокодов.

4. Загрузить в ОЗУ необходимые константы и исходные данные.

5. Отладить программу.







Код программы:

RD #40

WR 30

RD #10

WR 31

RD #0

WR 32

M1: RD 32

RD @30

JNS 12

RD 32

ADD #1

WR 32

RD 30

ADD #1

WR 30

RD 31

SUB #1

WR 31

JNZ M1

RD 32

OUT

HLT

Исходные данные: -1; 2; -3; 4; -5; -6; -7; -8; -9; 9.

Результат: 7.

Ответ на контрольный вопрос:

3. Как поведет себя программа, приведенная в табл. 10.8, если в ней будет

отсутствовать команда WR 31 по адресу 014?

Цикл будет бесконечным, так как никуда не запишется уменьшение параметра цикла, программа не сможет прерваться.

Задание.

Составить и отладить программу учебной ЭВМ для решения следующей за-

дачи. Три массива в памяти заданы начальными адресами и длинами. Вычис-

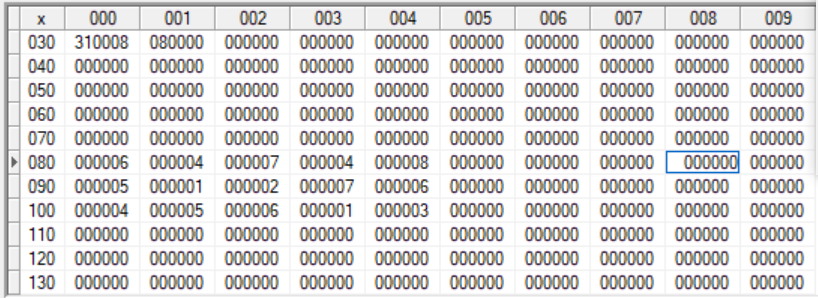
лить и вывести на устройство вывода среднее арифметическое параметров

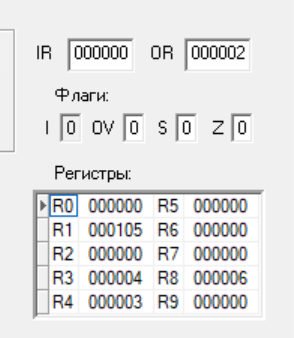
этих массивов. Параметры определяются заданием к предыдущей лаборатор-

ной работе (см. табл. 10.9), причем соответствие между номерами вариантов

заданий к лабораторным работам № 5 и 6 устанавливается по табл. 10.11.







Код программы:

RD #80

WR R1

RD #5

WR R2

CALL M

RD #90

WR R1

RD #5

WR R2

CALL M

RD #100

WR R1

RD #5

WR R2

CALL M

DIV #3

OUT

HLT

M: RD @R1

WR R3

L2: RD @R1+

WR R4

RD R3

SUB R4

JS L1

JZ L1

RD R8

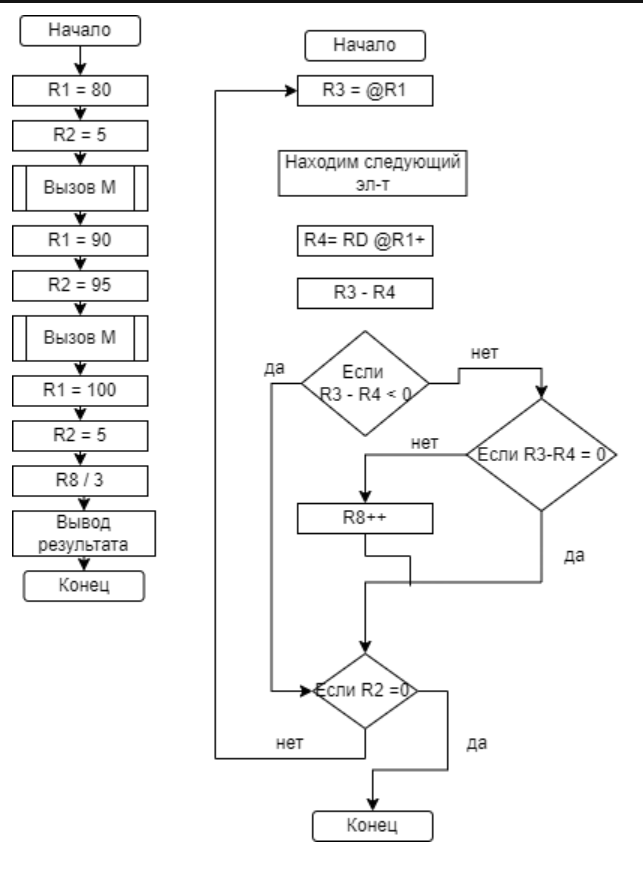
ADD #1

WR R8

L1: JRNZ R2,L2

RD R8

RET



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | IR | Команда 1 | Команда 2 | Команда 3 | Команда 4 | Команда 5 |
| 6 | X | RD #4 | WR 11 | RD @11 | ADD #330 | JS 000 |

RD #4

WR 11

RD @11

ADD #330

JS 000

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Адрес (PC) | Мнемокод | Микрокоманда | ОЗУ | | CR | | | АУ | | Ячейки |
| MAR | MDR | COP | TA | ADR | Acc | DR | 011 |
| 000 | RD #4 | MAR := PC | 000 | 000000 | 00 | 0 | 000 | 000000 | 000000 | 000000 |
|  |  | MRd | 000 |  |  |  |  |  |  |  |
|  |  | CR := MDR |  | 211004 |  |  |  |  |  |  |
|  |  | PC := PC+1 |  |  | 21 | 1 | 004 |  |  |  |
| 001 |  | Acc := ADR |  |  |  |  |  |  |  |  |
|  | WR 11 | MAR := PC |  |  |  |  |  | 000004 |  |  |
|  |  | MRd | 001 |  |  |  |  |  |  |  |
|  |  | CR := MDR |  | 220011 |  |  |  |  |  |  |
|  |  | PC := PC+1 |  |  | 22 | 0 | 011 |  |  |  |
| 002 |  | MAR := ADR |  |  |  |  |  |  |  |  |
|  |  | MDR := Acc | 011 |  |  |  |  |  |  |  |
|  |  | MWr |  | 000004 |  |  |  |  |  |  |
|  | RD @11 | MAR := PC |  |  |  |  |  |  |  | 000004 |
|  |  | MRd | 002 |  |  |  |  |  |  |  |
|  |  | CR := MDR |  | 212011 |  |  |  |  |  |  |
|  |  | PC := PC+1 |  |  | 21 | 2 | 011 |  |  |  |
| 003 |  | MAR := ADR |  |  |  |  |  |  |  |  |
|  |  | MRd | 011 |  |  |  |  |  |  |  |
|  |  | RA := MDR |  | 000004 |  |  |  |  |  |  |
|  |  | MAR := RA |  |  |  |  |  |  |  |  |
|  |  | MRd | 004 |  |  |  |  |  |  |  |
|  |  | Acc := MDR |  | 130000 |  |  |  |  |  |  |
|  | ADD #330 | MAR := PC |  |  |  |  |  | 130000 |  |  |
|  |  | MRd | 003 |  |  |  |  |  |  |  |
|  |  | CR := MDR |  | 231330 |  |  |  |  |  |  |
|  |  | PC := PC+1 |  |  | 23 | 1 | 330 |  |  |  |
| 004 |  | DR := ADR |  |  |  |  |  |  |  |  |
|  |  | ALU <-- COP |  |  |  |  |  |  | 000330 |  |
|  |  | Start ALU |  |  |  |  |  |  |  |  |
|  | JS 000 | MAR := PC |  |  |  |  |  | 129670 |  |  |
|  |  | MRd | 004 |  |  |  |  |  |  |  |
|  |  | CR := MDR |  | 130000 |  |  |  |  |  |  |
|  |  | PC := PC+1 |  |  | 13 | 0 | 000 |  |  |  |
| 005 |  | JS |  |  |  |  |  |  |  |  |

ADD R3

ADD @R3

ADD @R3+

ADD -@R3

JRNZ R3,M

MOV R4,R2

JMP M

CALL M

RET: PUSH R3

POP R5

M: RD #4

WR 41

RD @41

ADD #330

JS 000

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Адрес (PC) | Мнемокод | Микрокоманда | ОЗУ | | CR | | | АУ | | Ячейки | |
| MAR | MDR | COP | TA | ADR | Acc | DR | 041 | R3 |
| 00 | ADD R3 | MAR:=PC | 000 | 000000 | 00 | 0 | 000 | 000000 | 000000 | 000 | 000 |
|  |  | MRd | 000 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 330003 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 33 | 0 | 003 |  |  |  |  |
| 001 |  | RAR:=CR5 |  |  |  |  |  |  |  |  |  |
|  |  | RRd |  |  |  |  |  |  |  |  |  |
|  |  | DR:=RDR |  |  |  |  |  |  |  |  |  |
|  |  | ALU <-- COP |  |  |  |  |  |  |  |  |  |
|  |  | Start ALU |  |  |  |  |  |  |  |  |  |
|  | ADD @R3 | MAR:=PC |  |  |  |  |  |  |  |  |  |
|  |  | MRd | 001 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 334003 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 33 | 4 | 003 |  |  |  |  |
| 002 |  | RAR:=CR5 |  |  |  |  |  |  |  |  |  |
|  |  | RRd |  |  |  |  |  |  |  |  |  |
|  |  | RA:=RDR |  |  |  |  |  |  |  |  |  |
|  |  | MAR:=RA |  |  |  |  |  |  |  |  |  |
|  |  | MRd | 000 |  |  |  |  |  |  |  |  |
|  |  | DR:=MDR |  | 330003 |  |  |  |  |  |  |  |
|  |  | ALU🡨COP |  |  |  |  |  |  | 330003 |  |  |
|  |  | Start ALU |  |  |  |  |  |  |  |  |  |
|  | ADD @R3+ | MAR:=PC |  |  |  |  |  | 330003 |  |  |  |
|  |  | MRd | 002 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 335003 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 33 | 5 | 003 |  |  |  |  |
| 003 |  | RAR:=CR5 |  |  |  |  |  |  |  |  |  |
|  |  | RRd |  |  |  |  |  |  |  |  |  |
|  |  | RA:=RDR |  |  |  |  |  |  |  |  |  |
|  |  | MAR:=RA |  |  |  |  |  |  |  |  |  |
|  |  | MRd | 000 |  |  |  |  |  |  |  |  |
|  |  | DR:=MDR |  | 330003 |  |  |  |  |  |  |  |
|  |  | INC\_GR |  |  |  |  |  |  |  |  |  |
|  |  | ALU🡨COP |  |  |  |  |  |  |  |  | 001 |
|  |  | Start AlU |  |  |  |  |  |  |  |  |  |
|  | ADD -@R3 | MAR:=PC |  |  |  |  |  |  | 560006 |  |  |
|  |  | MRd | 003 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 336003 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 33 | 6 | 003 |  |  |  |  |
| 004 |  | RAR:=CR5 |  |  |  |  |  |  |  |  |  |
|  |  | RRd |  |  |  |  |  |  |  |  | 000 |
|  |  | RA:=RDR |  |  |  |  |  |  |  |  |  |
|  |  | MAR:=RA |  |  |  |  |  |  |  |  |  |
|  |  | MRd | 000 |  |  |  |  |  |  |  |  |
|  |  | DR:=MDR |  | 330003 |  |  |  |  |  |  |  |
|  |  | ALU🡨COP |  |  |  |  |  |  |  |  |  |
|  |  | Start AlU |  |  |  |  |  |  |  |  |  |
|  | JRNZ R3, 10 | MAR:=PC |  |  |  |  |  | 790009 |  |  |  |
|  |  | MRd | 004 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 173010 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 17 | 3 | 010 |  |  |  |  |
| 005 |  | RAR:=CR2 |  |  |  |  |  |  |  |  |  |
|  |  | DEC\_GR |  |  |  |  |  |  |  |  |  |
|  |  | JRNZ |  |  |  |  |  |  |  |  | 100001 |
| 010 | RD #4 | MAR:=PC |  |  |  |  |  |  |  |  |  |
|  |  | MRd | 010 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 211004 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 21 | 1 | 004 |  |  |  |  |
| 011 |  | Acc:=ADR |  |  |  |  |  |  |  |  |  |
|  | WR 41 | MAR:=PC |  |  |  |  |  | 000004 |  |  |  |
|  |  | MRd | 011 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 220041 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 22 | 0 | 041 |  |  |  |  |
| 012 |  | MAR:=ADR |  |  |  |  |  |  |  |  |  |
|  |  | MDR:=Acc | 041 |  |  |  |  |  |  |  |  |
|  |  | MWr |  | 000004 |  |  |  |  |  |  |  |
|  | RD @41 | MAR:=PC |  |  |  |  |  |  |  | 004 |  |
|  |  | MRd | 012 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 212041 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 21 | 2 | 041 |  |  |  |  |
| 013 |  | MAR:=PC |  |  |  |  |  |  | 330003 |  |  |
|  |  | MRd | 041 |  |  |  |  |  |  |  |  |
|  |  | RA:=MDR |  | 000004 |  |  |  |  |  |  |  |
|  |  | MAR:=RA |  |  |  |  |  |  |  |  |  |
|  |  | MRd | 004 |  |  |  |  |  |  |  |  |
|  |  | Acc:=MDR |  | 173010 |  |  |  |  |  |  |  |
|  | ADD #330 | MAR:=PC |  |  |  |  |  | 173010 |  |  |  |
|  |  | MRd | 013 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 231330 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 23 | 1 | 330 |  |  |  |  |
| 014 |  | DR:=ADR |  |  |  |  |  |  |  |  |  |
|  |  | ALU🡨COP |  |  |  |  |  |  | 000330 |  |  |
|  |  | Start ALU |  |  |  |  |  |  |  |  |  |
|  | JS 000 | MAR:=PC |  |  |  |  |  | 172680 |  |  |  |
|  |  | MRd | 014 |  |  |  |  |  |  |  |  |
|  |  | CR:=MDR |  | 130000 |  |  |  |  |  |  |  |
|  |  | PC:=PC+1 |  |  | 13 | 0 | 000 |  |  |  |  |
| 015 |  | JS |  |  |  |  |  |  |  |  |  |