**3 ЛАБОРАТОРНАЯ РАБОТА №3**

**ПОДПРОГРАММЫ И СТЕК. КОМАНДНЫЙ ЦИКЛ ПРОЦЕССОРА**

**Цель:** Реализация подпрограмм. Исследование работы микрокоманд.

**3.1 Формулировка варианта задания**

Вариант 12. Даны три массива чисел. Требуется вычислить среднее арифметическое позиций первых положительных элементов.

**3.2 Схема алгоритма основной программы**

Ниже представлена схема работы основной программы на рисунке 3.1.

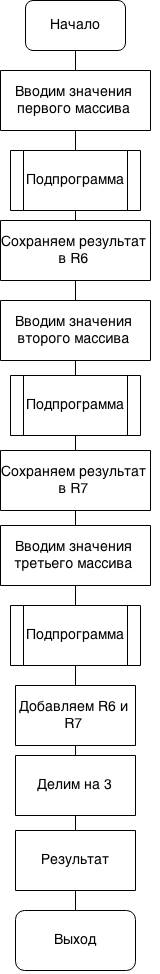


Рисунок 3.1 – Основная программа

**3.3 Схема алгоритма подпрограммы**

Ниже представлена схема нахождения позиции положительного элемента

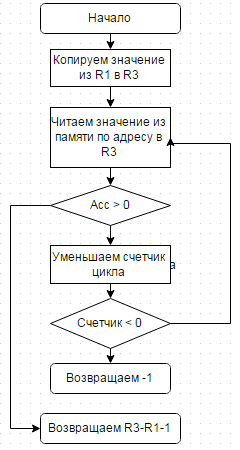


Рисунок 3.2 – Алгоритм подпрограммы

**3.4 Распределение памяти**

Первый массив элементов начинается с адреса 80 и имеет 7 элементов, второй – адрес 90 и 9 элементов, третий – адрес 100 и 8элементов. Регистры R6, R7 и R8 предназначены для хранения позиций первых положительных элементов. Подпрограмма получает параметры через регистры R1 и R2. Регистр R3 хранит адрес текущего элемента.

**3.5 Тексты программ и подпрограмм**

RD #80

WR R1

RD #7

WR R2

CALL FIND\_FIRST\_ POS

WR R6

RD #90

WR R1

RD #9

WR R2

CALL FIND\_FIRST\_ POS

WR R7

RD #100

WR R1

RD #8

WR R2

CALL FIND\_FIRST\_ POS

WR R8

RD #0

ADD R6

ADD R7

ADD R8

DIV #3

OUT

HLT

FIND\_FIRST\_NEG:

RD R1

WR R3

LOOP:

RD @R3+

JNS END\_LOOP

RD R2

SUB #1

WR R2

JNS LOOP

RDI 10001

RET

END\_LOOP:

RD R3

SUB R1

SUB #1

RET

**3.6 Результаты выполнения программы**

Первый массив : 100510, 100900, 100041, 000005, 100020, 100000.

Второй массив : 100562, 000174, 056548, 10045, 012345, 000000.

Третий массив : 100789, 105558, 188898, 045545, 007885.

Результат программы : 3.

**3.7 Таблица состояний процессора**

Вариант 12. Состояния процессора представлены в таблице 3.1

**3.8 Последовательности микрокоманд**

ADD R3: MAR:= PC, MRd, CR:=MDR, PC:=PC+1, RAR:=CR5, RRd, DR:=RDR, ALU<-COP, Strart ALU, END\_COMMAND

ADD @R3: MAR:= PC, MRd, CR:=MDR, PC:=PC+1, RAR:=CR5, RRd, RA:=RDR, MAR:=RA, MRd, DR:=MDR, ALU<-COP, Start ALU, END\_COMMAND

ADD @R3+: MAR:= PC, MRd, CR:=MDR, PC:=PC+1, RAR:=CR5, RRd, RA:=RDR, MAR:=RA, MRd, DR:=MDR, INC\_GR, ALU<-COP, Start ALU, END\_COMMAND

ADD -@R3: MAR:= PC, MRd, CR:=MDR, PC:=PC+1, RAR:=CR5, DEC\_GR, RRd, RA:=RDR, MAR:=RA, MRd, DR:=MDR, ALU<-COP, Start ALU, END\_COMMAND

MOV R4,R2: MAR:= PC, MRd, CR:=MDR, PC:=PC+1, RAR:=CR5, RRd, RAR:=CR4, RWr, END\_COMMAND

JMP M: MRd, CR:=MDR, PC:=PC+1, PC:=ADR, END\_COMMAND

PUSH R3: MAR:= PC, MRd, CR:=MDR, PC:=PC+1, RAR:=CR5, RRd, MDR:=RDR, SP:=SP-1, MAR:=SP, MWr, END\_COMMAND

POP R5: MAR:= PC, MRd, CR:=MDR, PC:=PC+1, MAR:=SP, MRd, RDR:=MDR, SP:=SP+1, RAR:=CR5, RWr, END\_COMMAND

CALL M: MAR:= PC, MRd, CR:=MDR, PC:=PC+1,MDR:=PC, SP:=SP-1, MAR:=SP, MWr, PC:=ADR, END\_COMMAND

**ВЫВОДЫ**

В данной лабораторной работе была создана программа, которая с помощью вызова подпрограмм вычисляет среднее значение позиций положительных чисел в массиве. Так же рассмотрели более тщательно выполнение команд, которые разбиваются на микрокоманды. структура всех команд оказалась похожей, сначала заполняются вспомогательные регистры и ячейки, а затем выполняется непосредственно необходимая операция.