МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Курский государственный университет»

кафедра программного обеспечения и администрирования

информационных систем

Отчёт

по лабораторной работе №2.2

«Программирование систем контроля времени»

по дисциплине

###### «Архитектура вычислительных систем»

|  |  |
| --- | --- |
| Выполнил: | студент группы 313  Козявин Максим Сергеевич |
| Проверил: | к.т.н., доцент  кафедры ПОиАИС  Жмакин А.П. |

Курск

2023

**Вариант задания:**



Рисунок 1 – Вариант задания

1. Система: 2 секундомера
2. Счётчик: Таймер\_5
3. Способ формирования временного интервала: Сброс по совпадению
4. Способ обнаружения переполнения/сравнения счётчика времени TCNT: По прерыванию
5. Устройство ввода: Матричная клавиатура
6. Устройство вывода: 7-сегментный индикатор

**Граф-схема алгоритма основной программы и подпрограмм:**

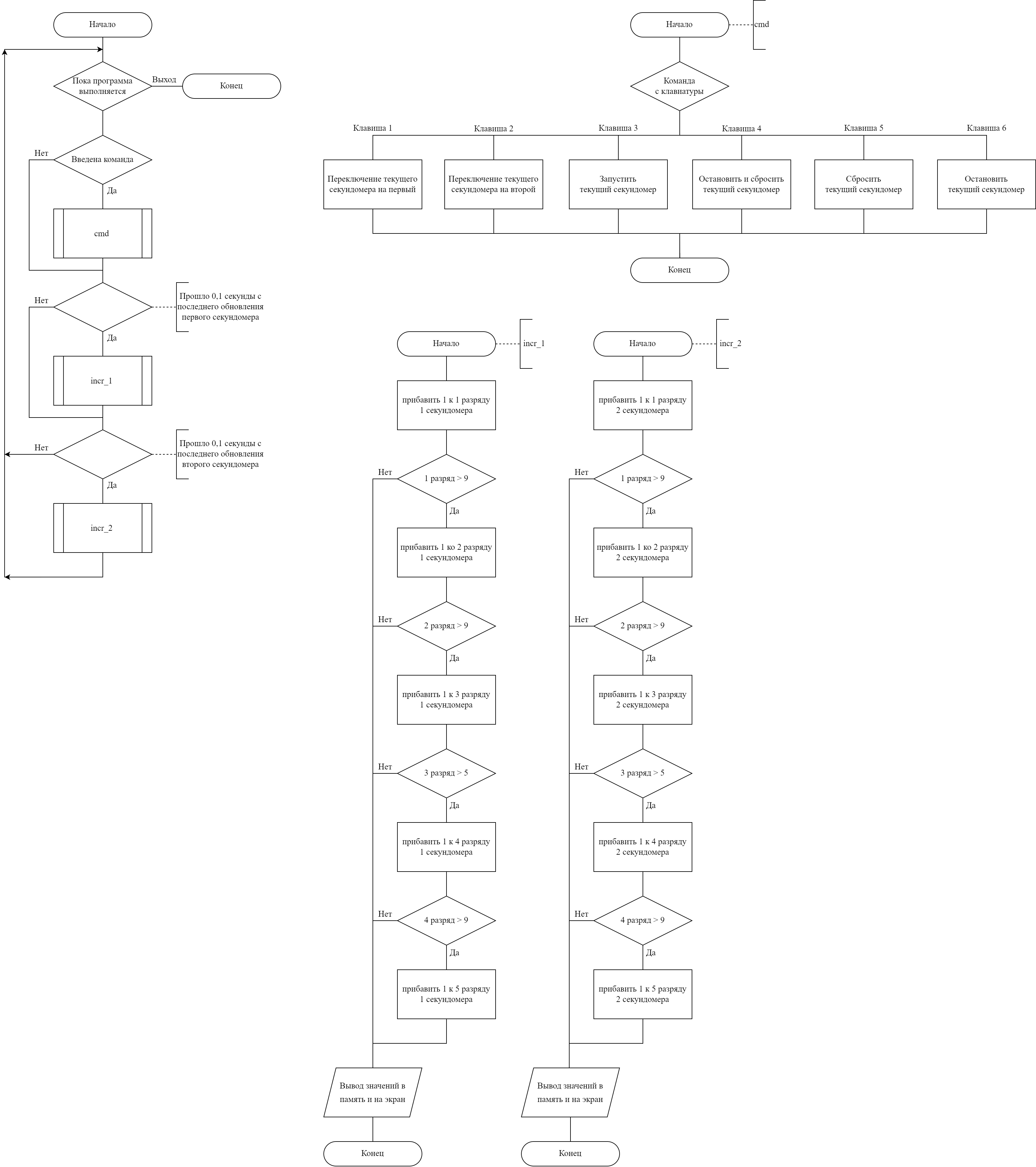


Рисунок 2 – граф-схема алгоритма основной программы

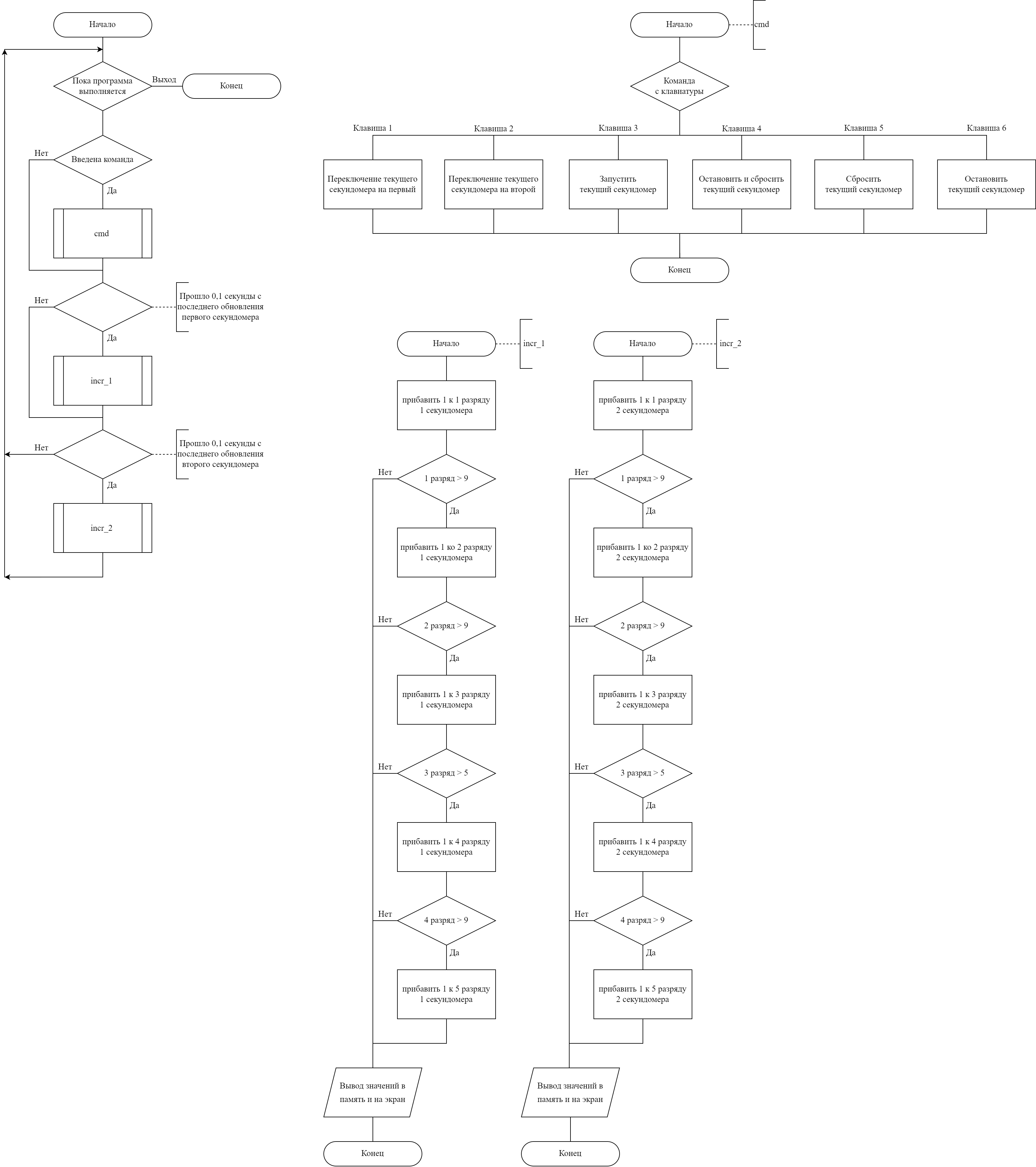


Рисунок 3 – граф-схема подпрограммы обработчика клавиатуры

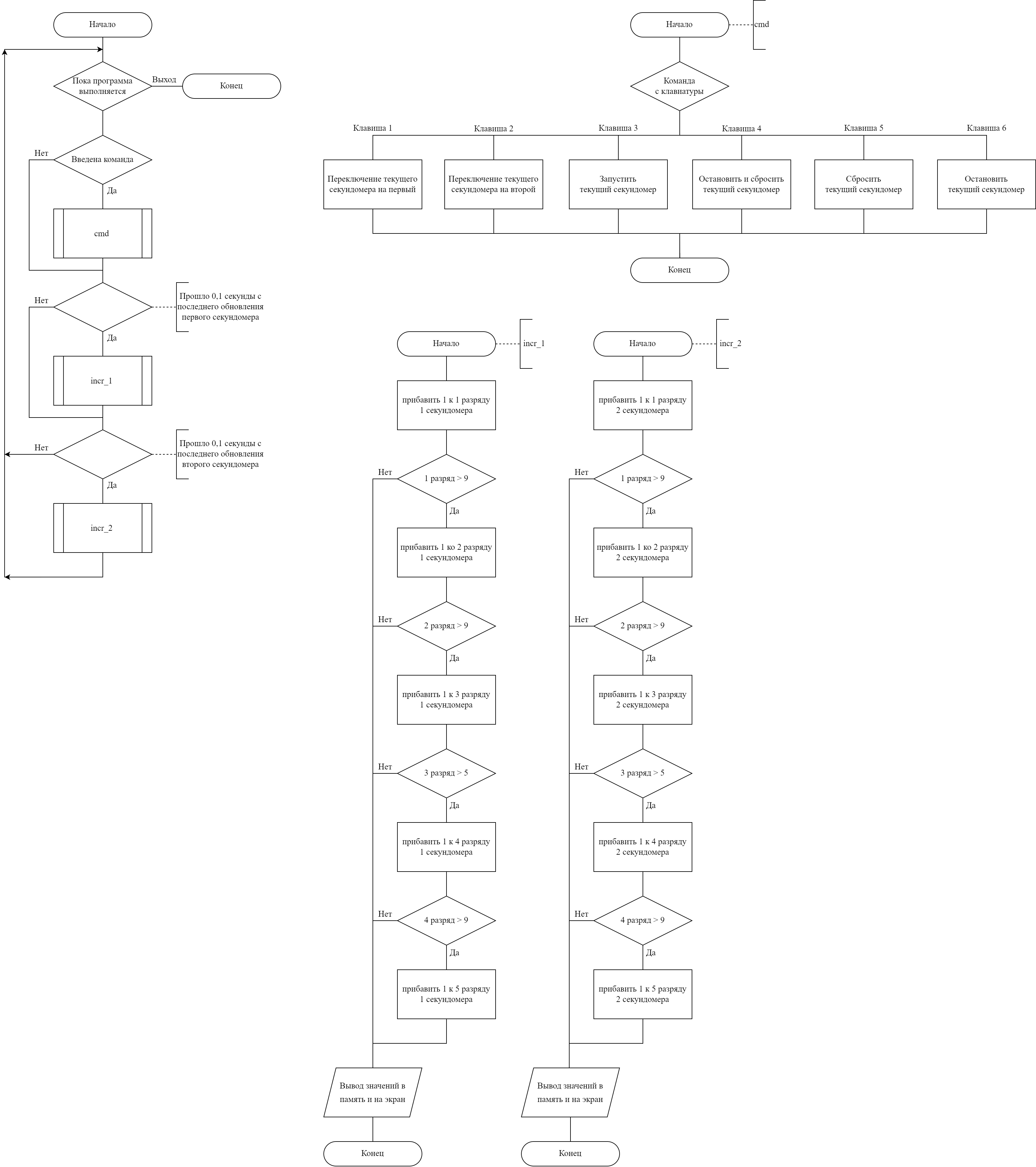


Рисунок 4 – граф-схема подпрограмм инкрементирования

**Текст программы на языке Ассемблера fN8 с комментариями:**

; Таймер 5 базовый адрес 1 Вектор прерывания 1

; Таймер 5 базовый адрес 2 Вектор прерывания 2

; Клав. и индикация базовый адрес 4 Вектор прерывания 4

; Клавиша 1 - Переключить на 1 секундомер

; Клавиша 2 - Переключить на 2 секундомер

; Клавиша 3 - Запустить текущий секундомер

; Клавиша 4 - Остановить и сбросить текущий секундомер

; Клавиша 5 - Сбросить текущий секундомер

; Клавиша 6 - Остановить текущий секундомер

rd #1

wr r7 ; r7 отвечает за текущий секундомер

call fill0

rd #0

wr r7

call fill0 ; заполнение нулями

rd #0b111 ; настройка 7-сег.

out 0x42

rd #5

out 0x14 ; настройка таймеров

out 0x24

rd #0b00101001

out 0x16

rd #0b00111001

out 0x26

rd #0b01000

out 0x17

out 0x27

ei

loop:

nop

jmp loop

incr\_1: ; инкремент значений с переносом при переполнении

rd 0x0

add #1

cmp #10

wr 0x0

jnz skip1

rd #0

wr 0x0

rd 0x1

add #1

cmp #10

wr 0x1

jnz skip1

rd #0

wr 0x1

rd 0x2

add #1

cmp #6

wr 0x2

jnz skip1

rd #0

wr 0x2

rd 0x3

add #1

cmp #10

wr 0x3

jnz skip1

rd #0

wr 0x3

rd 0x4

add #1

cmp #6

wr 0x4

jnz skip1

rd #0

wr 0x4

skip1:

rd r7

cmp #0

jnz rt1

call display\_write ; вывод новых значений

rt1:

iret

cmd: ; обработка команд с матричной клавиатуры

in 0x41

cmp #1 ; установка 1 секундомера текущим

jnz next

rd #0

wr r7

jmp brk

next:

cmp #2 ; установка 2 секундомера текущим

jnz next2

rd #1

wr r7

jmp brk

next2:

cmp #3 ; запуск

jnz next3

rd r7

cmp #0

jnz scnd

rd #0b00101001

out 0x16

jmp brk

scnd:

rd #0b00111001

out 0x26

jmp brk

next3:

cmp #4 ; стоп

jnz next4

rd r7

cmp #0

jnz scnd2

rd #0b00100001

out 0x16

call fill0

call display\_write

jmp brk

scnd2:

rd #0b00110001

out 0x26

call fill0

call display\_write

jmp brk

next4:

cmp #5 ; сброс

jnz next5

call fill0

jmp brk

next5:

cmp #6 ; пауза

jnz brk

rd r7

cmp #0

jnz scnd3

rd #0b00100001

out 0x16

jmp brk

scnd3:

rd #0b00110001

out 0x26

brk:

iret

incr\_2: ; инкремент для 2 секундомера

rd 0x10

add #1

cmp #10

wr 0x10

jnz skip2

rd #0

wr 0x10

rd 0x11

add #1

cmp #10

wr 0x11

jnz skip2

rd #0

wr 0x11

rd 0x12

add #1

cmp #6

wr 0x12

jnz skip2

rd #0

wr 0x12

rd 0x13

add #1

cmp #10

wr 0x13

jnz skip2

rd #0

wr 0x13

rd 0x14

add #1

cmp #6

wr 0x14

jnz skip2

rd #0

wr 0x14

skip2:

rd r7

cmp #1

jnz rt2

call display\_write

rt2:

iret

fill0: ; заполнение нулями 1 или 2 секундомера в зависимости от значения r7

rd #5

wr r6

rd r7

cmp #0

jz frst

rd r6

rd #0x15

wr r6

frst:

rd #0

wr -@r6

rd r6

cmp #0

jnz frst

ret

display\_write: ; вывод значений секундомера зависимо от r7

rd #4

wr r6

rd #0

out 0x44 ; 0

rd r7

cmp #0

jz first

rd #0x14

wr r6

rd 0x42

out 0x44 ; 2

jmp print

first:

rd 0x41

out 0x44 ; 1

print:

rd #0

out 0x44

rd @r6-

add #0x40

wr r0

rd @r0

out 0x44

rd @r6-

add #0x40

wr r0

rd @r0

add #1 ; .

out 0x44

rd @r6-

add #0x40

wr r0

rd @r0

out 0x44

rd @r6-

add #0x40

wr r0

rd @r0

add #1

out 0x44

rd @r6-

add #0x40

wr r0

rd @r0

out 0x44

ret

.org 4

.dw incr\_2

.org 2

.dw incr\_1

.org 8

.dw cmd

.c 2

.org 0x40

.db 0b01111110, 0b00001100, 0b10110110, 0b10011110, 0b11001100, 0b11011010, 0b11111010, 0b00001110, 0b11111110, 0b11011110

**Распределение памяти и тестирование:**

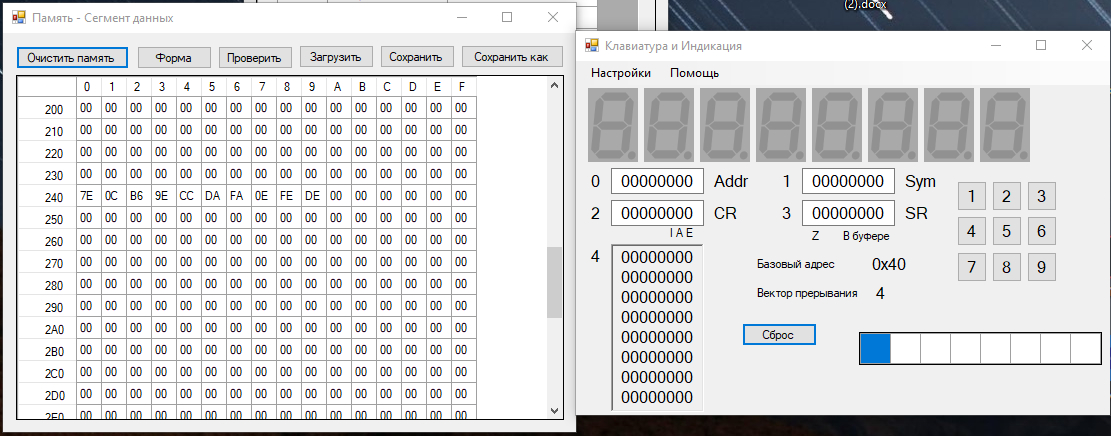
****

Рисунок 5 – Распределение памяти до запуска

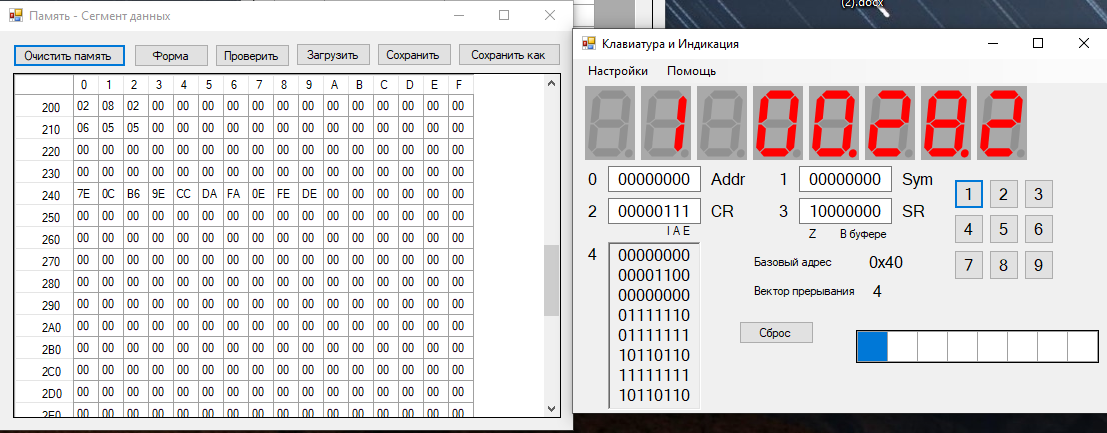


Рисунок 6 – Распределение памяти и вывод для 1 секундомера

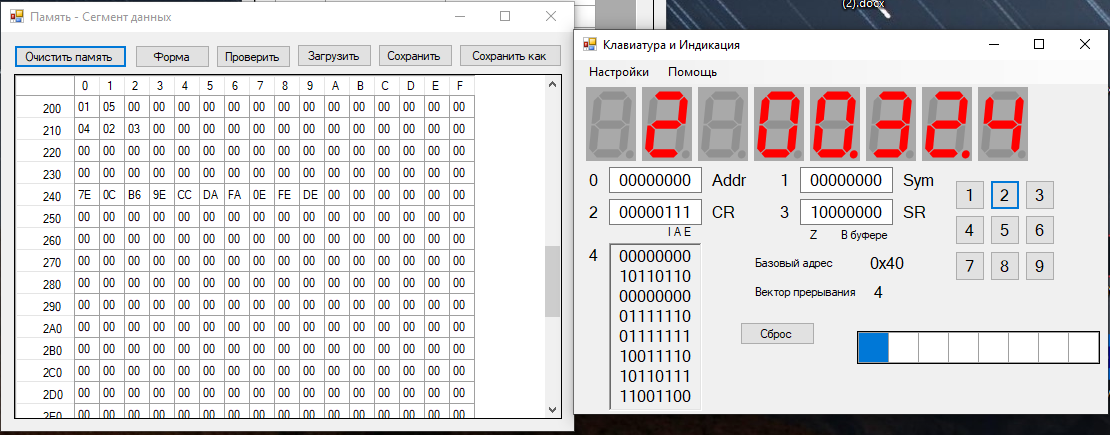


Рисунок 7 – Распределение памяти и вывод для 2 секундомера