МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**

(БГТУ им. В.Г.Шухова)

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Программирование микроконтроллеров

Лабораторная работа №2

**Обработка прерываний. Управление сегментным индикатором**

Выполнил:

ст. гр. ВТ-31

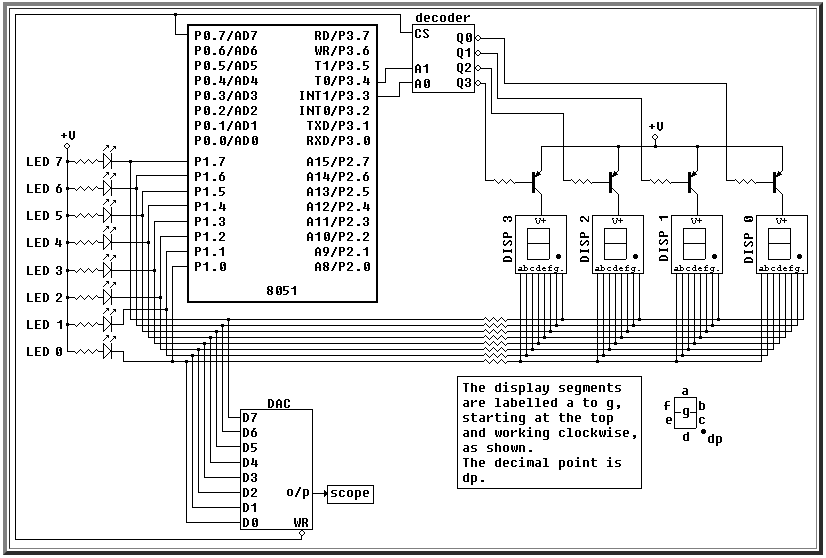
Новожен Н.В.

Белгород, 2020

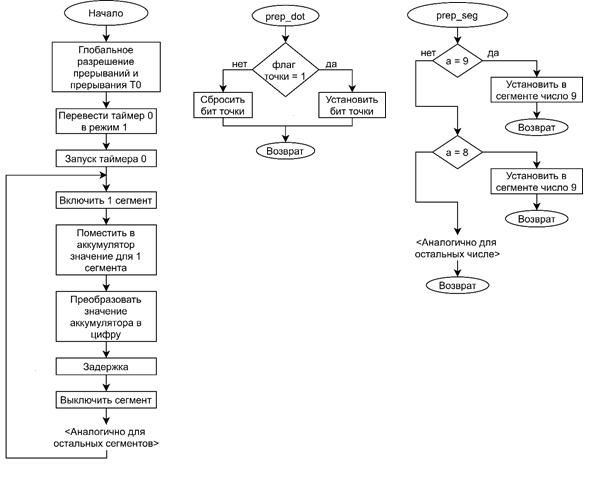
**Цель работы:** изучить устройство таймеров микроконтроллеров Intel 8051, а также принцип управления сегментным индикатором.

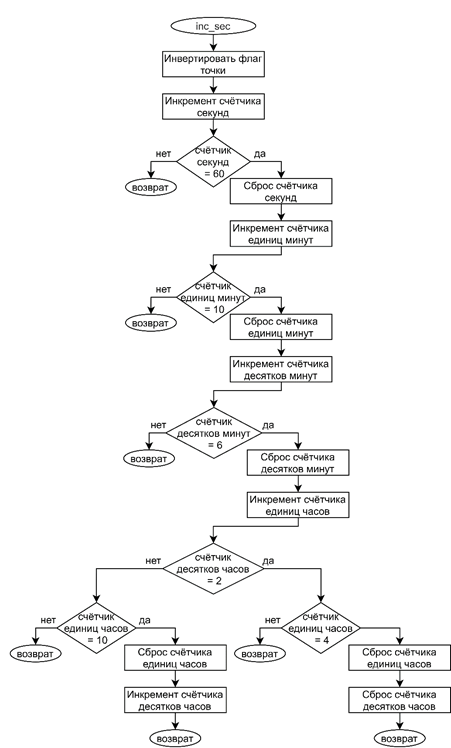
**Задание:** разработать программу, выводящую на сегментный индикатор часы, работающие на таймере.

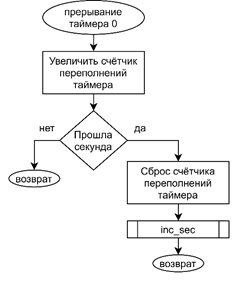
**Схема устройства**



**Блок-схема алгоритма**







**Назначения регистров**

* r0 - количество переполнений таймера
* r1 - счетчик секунд (0-59)
* r2 - счетчик единиц минут (0-9)
* r3 - счетчик десятков минут (0-5)
* r4 - счетчик единиц часов (0-9)
* r5 - счетчик десятков часов (0-2)
* r6 - флаг точки секунд

**Пример работы программы**

Начальный момент времени:



4 часа, чётное количество секунд:



5 часов, нечётной количество секунд:

+

**Код программы**

org 00h

jmp start

org 0Bh

inc r0

;cjne r0, #15, j\_reti

cjne r0, #1, j\_reti

mov r0, #0h

jmp inc\_sec

j\_reti:

reti

org 30h

start:

mov r0, #0h

mov r1, #0h

mov r2, #0h

mov r3, #0h

mov r4, #0h

mov r5, #0h

mov r6, #0h

;глобальное разрешение прерываний и прерывание таймера 0

mov ie, #10000010b

;переключение таймера 0 в 1 режим

mov tmod, #1h

;запуск таймера 0

setb tr0

main\_loop:

;сегмент 1

clr p3.3

clr p3.4

mov a,r2

call prep\_seg

call delay

mov p1,#0FFh

;сегмент 2

setb p3.3

clr p3.4

mov a,r3

call prep\_seg

call delay

mov p1,#0FFh

;сегмент 3

clr p3.3

setb p3.4

mov a,r4

call prep\_seg

call prep\_dot

call delay

mov p1,#0FFh

;сегмент 4

setb p3.3

setb p3.4

mov a,r5

call prep\_seg

call delay

mov p1,#0FFh

jmp main\_loop

prep\_dot:

cjne r6,#0,reset\_dot

setb p1.7

ret

reset\_dot:

clr p1.7

ret

prep\_seg:

cjne a,#9, num8

mov p1,#10010000b

ret

num8:

cjne a,#8, num7

mov p1,#10000000b

ret

num7:

cjne a,#7, num6

mov p1,#11111000b

ret

num6:

cjne a,#6, num5

mov p1,#10000010b

ret

num5:

cjne a,#5, num4

mov p1,#10010010b

ret

num4:

cjne a,#4, num3

mov p1,#10011001b

ret

num3:

cjne a,#3, num2

mov p1,#10110000b

ret

num2:

cjne a,#2, num1

mov p1,#10100100b

ret

num1:

cjne a,#1, num0

mov p1,#11111001b

ret

num0:

mov p1,#11000000b

ret

inc\_sec:

mov a, r6

cpl a

mov r6, a ; инверсия флага точки

inc r1 ; секунды

;cjne r1, #60, end\_inc\_sec

cjne r1, #1, end\_inc\_sec

mov r1, #0h

inc r2 ; единицы минут

;cjne r2, #10, end\_inc\_sec

cjne r2, #1, end\_inc\_sec

mov r2, #0h

inc r3 ; десятки минут

;cjne r3, #6, end\_inc\_sec

cjne r3, #1, end\_inc\_sec

mov r3, #0h

inc r4 ; единицы часов

cjne r5, #2, hour\_full ; если десятки часов не равны 2

cjne r4, #4, end\_inc\_sec

mov r4, #0h

mov r5, #0h

reti

hour\_full:

cjne r4, #10, end\_inc\_sec

mov r4, #0h

inc r5

end\_inc\_sec:

reti

delay:

mov b,#200

djnz b, $

ret