

A

A

B

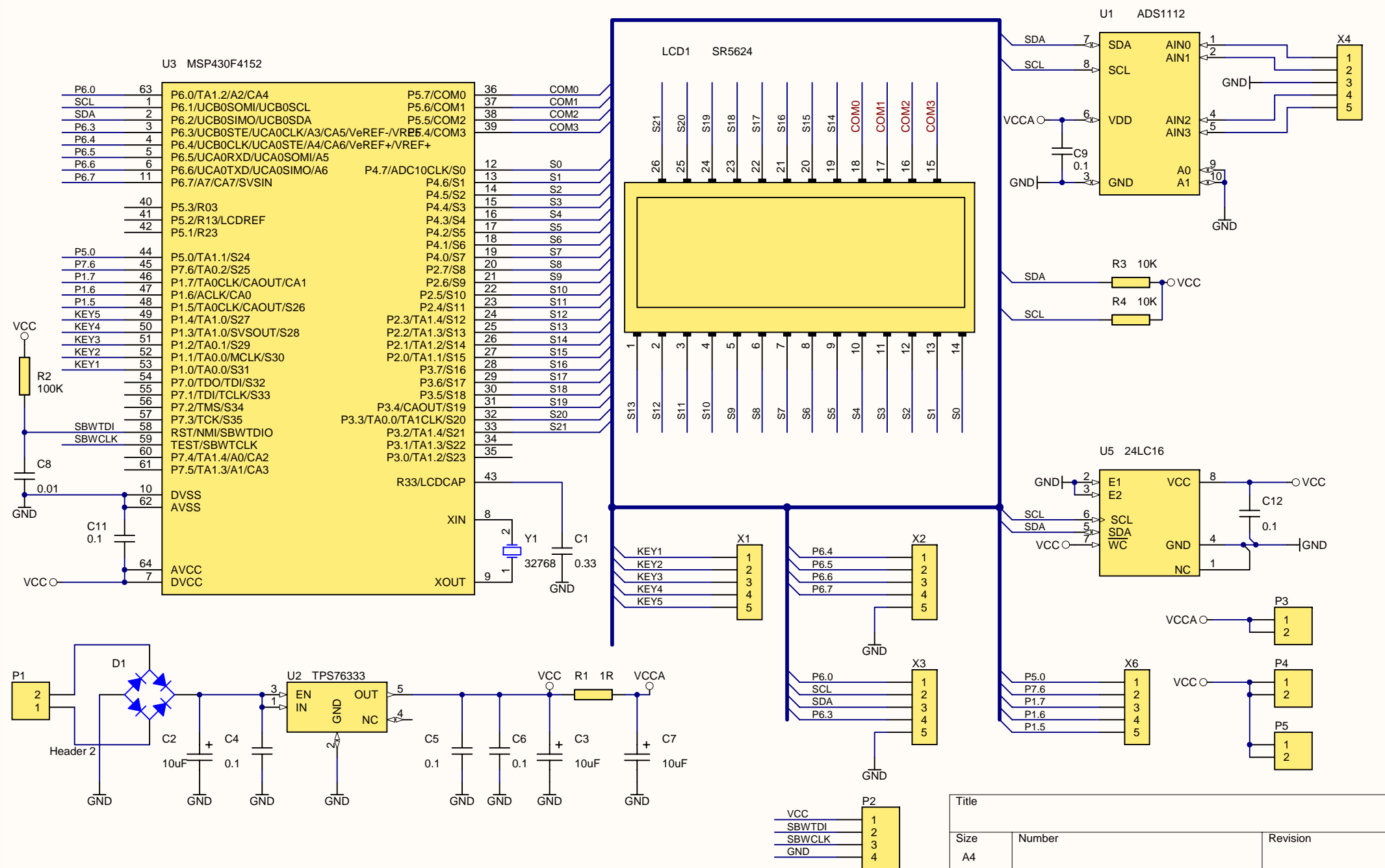
B

C

C

D

D



Title		
Size	Number	Revision
A4		
Date:	27.01.2010	Sheet of
File:	D:\my_work\...\mlcd.SchDoc	Drawn By:

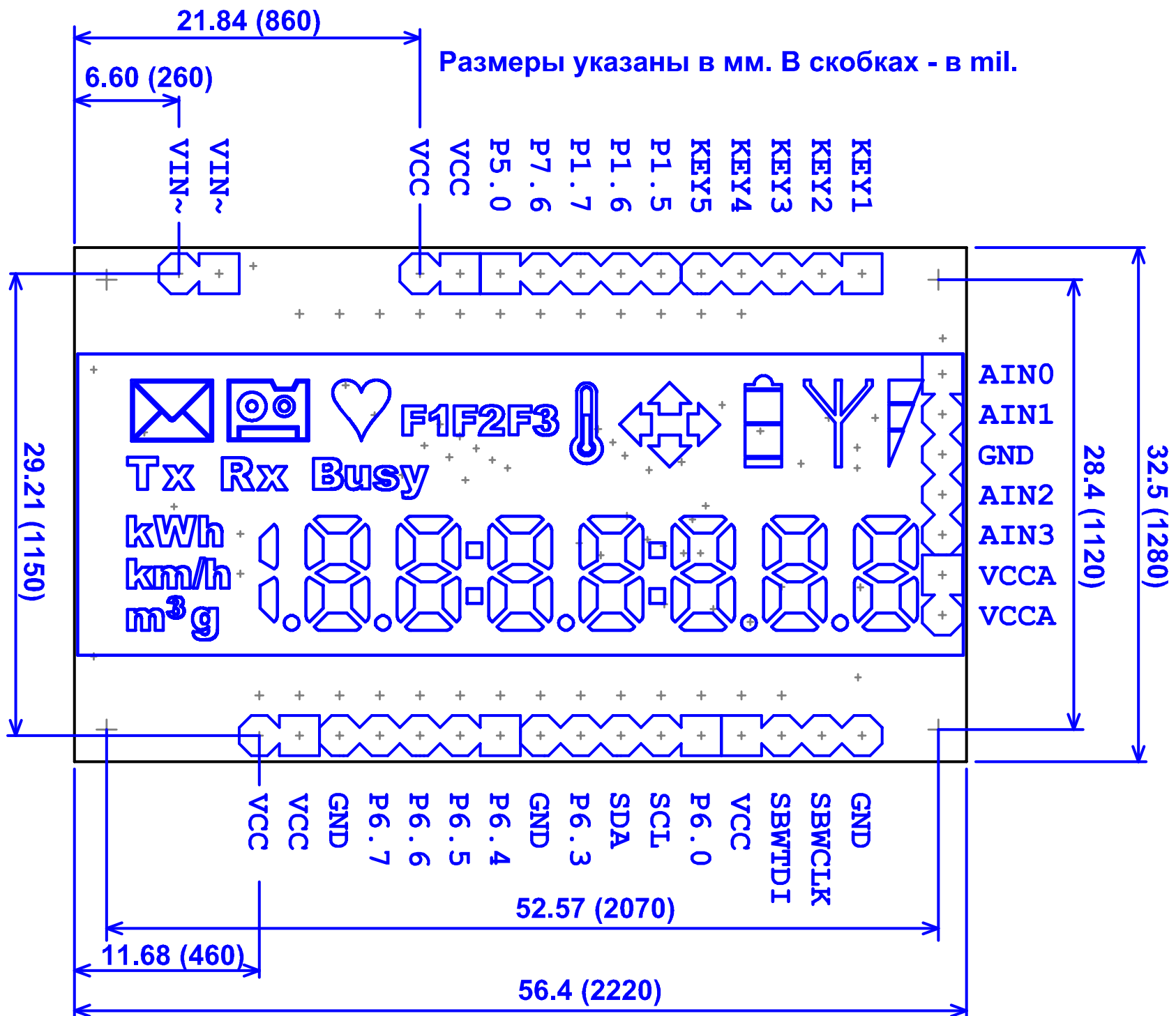



















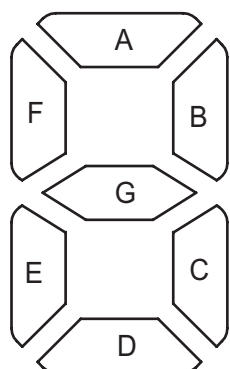
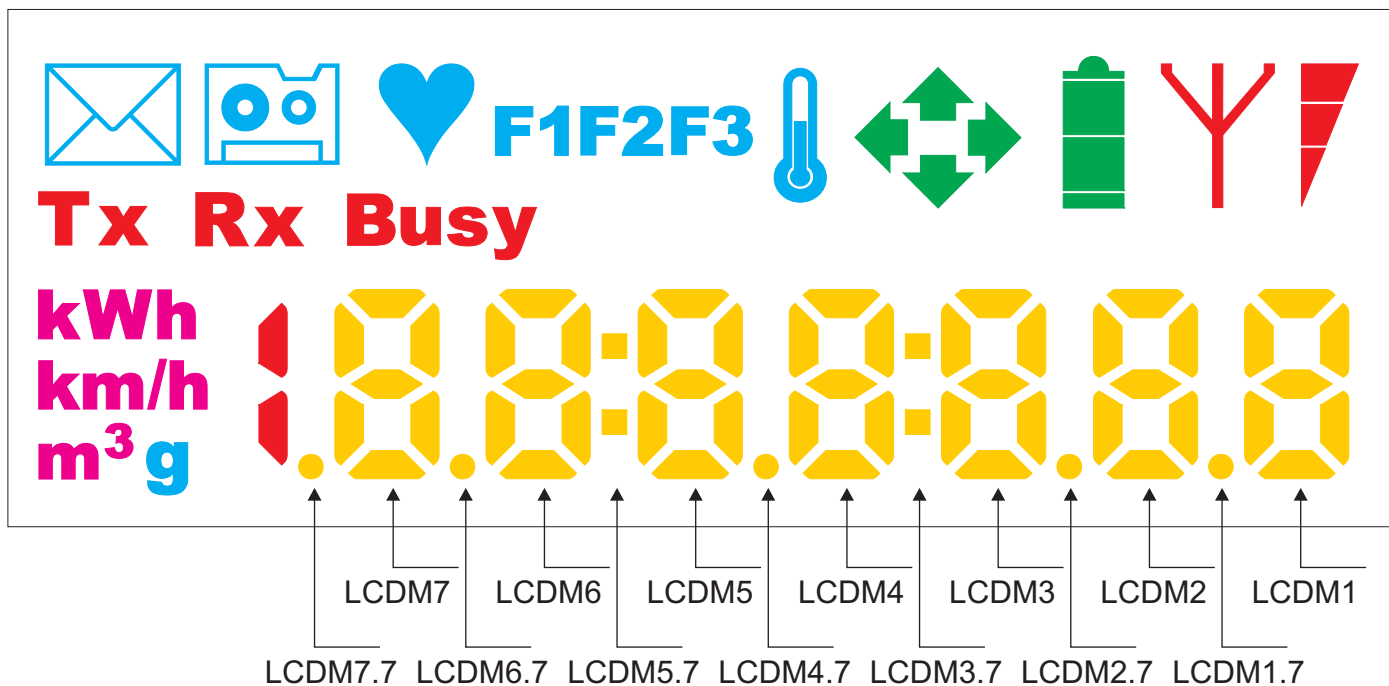


Таблица соответствия символов

Символ	Регистр
	LCDM8.5
	LCDM8.6
	LCDM8.7
<b>F1</b>	LCDM8.0
<b>F2</b>	LCDM8.1
<b>F3</b>	LCDM8.2
	LCDM8.3
	LCDM9.3
	LCDM9.1
	LCDM9.0
	LCDM9.2
	LCDM10.0
	 LCDM9.4  LCDM9.5  LCDM9.6  LCDM9.7
	 LCDM10.1  LCDM10.2  LCDM10.3
<b>Tx</b>	LCDM10.6
<b>Rx</b>	LCDM10.5
<b>Busy</b>	LCDM10.4
<b>kW</b>	LCDM11.3
<b>km/h</b>	LCDM11.1
<b>m</b>	LCDM11.0
<b>3</b>	LCDM11.7
<b>g</b>	LCDM8.4
	LCDM10.7
<b>h</b>	LCDM11.2



A equ 001h; Бит LCDMx.0  
 B equ 002h; Бит LCDMx.1  
 C equ 004h; Бит LCDMx.2  
 D equ 008h; Бит LCDMx.3  
 E equ 040h; Бит LCDMx.6  
 F equ 010h; Бит LCDMx.4  
 G equ 020h; Бит LCDMx.5

a+b+c+d+e+f  
 b+c  
 a+b+d+e+g  
 a+b+c+d+g  
 b+c+f+g

a+c+d+f+g  
 a+c+d+e+f+g  
 a+b+c  
 a+b+c+d+e+f+g  
 a+b+c+d+f+g

```

;-----
; Пример программы часов для микроконтроллера MSP430F4152 и ЖКИ индикатора SR5624
; www.evodbg.com
;-----

; Определение переменных.
secs      equ    200h
mins      equ    201h
hrs        equ    202h

;*****
#include <msp430x41x2.h>
;-----
;-----
; RSEG    CSTACK                                ; Define stack segment
;-----
; RSEG    CODE                                  ; Assemble to Flash memory
;-----
RESET      mov.w    #300h,SP                    ; Инициализация указателя стека
Init_Sys   mov.w    #WDTPW+WDTHOLD,&WDTCTL      ; Остановить WDT

SetupP5     bis.b    #BIT4+BIT5+BIT6+BIT7, &P5DIR; Разрешить линии COM
            bis.b    #BIT4+BIT5+BIT6+BIT7, &P5SEL; Разрешить линии COM

SetupLCD    mov.b    #LCDON + LCD4MUX + LCDFREQ_64, &LCDACTL
            ; 4mux LCD, ACLK/64
            mov.b    #LCDS0+LCDS4+LCDS8+LCDS12+LCDS16+LCDS20+LCDS24, &LCDAPCTL0
            ; Разрешить сегменты 0-13
            mov.b    #LCDCPEN, &LCDAVCTL0       ; Включить charge pump
            mov.b    #VLCD_3_02, &LCDAVCTL1     ; Напряжение 3.02V

SetupBT     mov.b    #BTDIV+BTIP2+BTIP1,&BTCTL   ; Настройка Basic Timer
            bis.b    #BTIE,&IE2                ; Разрешить прерывание от Basic Timer

            clr.b    &hrs                      ; Очистка переменной часов
            clr.b    &mins                    ; Очистка переменной минут
            clr.b    &secs                    ; Очистка переменной секунд

ClearLCD    mov      #20,R15                    ; Цикл - очистка индикатора
Clear1      mov.b    #0,LCDM1(R15)
            dec      R15
            jc       Clear1

Mainloop    bis.w    #CPUOFF+GIE,SR             ; Режим LPM0, разрешить прерывания
            nop
            jmp      Mainloop

;-----
BT_ISR;     Прерывание каждую секунду - считаем время
;-----
            clrc
            dadd.b    #001h,&secs              ; Увеличиваем секунды
            cmp.b     #60h,&secs              ; Проверка если 60
            jnz       done
            mov.b     #0,&secs
            dadd.b    #0,&mins                ; Увеличить минуты
            cmp.b     #60h,&mins              ; Проверка если 60
            jnz       done
            mov.b     #0,&mins
            dadd.b    #0,&hrs                 ; Увеличить часы
            cmp.b     #13h,&hrs               ; Проверка если 13
            jnz       done
            mov.b     #01,&hrs                ; часы = 01
done
; Отображение на индикаторе

            mov.b     &hrs, R5                ; Копируем часы в регистр
            mov.b     R5,R6                  ; Повторно копируем

            rra.b     r6                     ; Сдвигаем вправо 4 раза
            rra.b     r6                     ; чтобы выделить десятки часов
            rra.b     r6
            rra.b     r6
            and.b     #0Fh, R6
            mov.b     LCD_Tab(R6), LCDM7      ; Перекодируем через таблицу и выводим на
ЖКИ

            mov.b     r5,r6                  ;
            and.b     #0Fh, R6               ; Выделяем единицы часов
            mov.b     LCD_Tab(R6), LCDM6      ; Перекодируем и выводим на ЖКИ

            mov.b     &mins, r5              ; Копируем минуты в регистр

```

```

mov.b    R5,R6                ; Повторно копируем
rra.b    r6                   ; Сдвигаем вправо
rra.b    r6                   ; чтобы выделить десятки минут
rra.b    r6
and.b    #0Fh, R6
mov.b    LCD_Tab(R6), LCDM5    ; Перекодируем и выводим на ЖКИ
bis.b    #BIT7, LCDM5         ; Отображаем "двоеточие" - разделитель
часов и минут
mov.b    r5,r6                ; Восстанавливаем минуты
and.b    #0fh, r6             ; Выделяем единицы минут
mov.b    LCD_Tab(R6), LCDM4    ; Перекодируем и выводим на ЖКИ

mov.b    &secs, r5            ; Копируем секунды
mov.b    R5,R6                ; Снова копируем
rra.b    r6                   ; Сдвигаем вправо 4 раза,
rra.b    r6                   ; чтобы выделить десятки секунд
rra.b    r6
and.b    #0Fh, R6
mov.b    LCD_Tab(R6), LCDM3    ; Перекодируем и выводим на ЖКИ

bis.b    #BIT7, LCDM3         ; Отображаем "двоеточие" - разделитель
mov.b    r5,r6                ; Восстанавливаем секунды
and.b    #0fh, r6
mov.b    LCD_Tab(R6), LCDM2    ; Выводим единицы секунд

mov.w    #GIE,0(SP)           ; Выходим из LPMX, прерывания разрешены
reti                                           ; Выходим из прерывания

```

```

;-----
;
;      Определения для ЖКИ
;-----
;Определение сегментов
a      equ      001h
b      equ      002h
c      equ      004h
d      equ      008h
e      equ      040h
f      equ      010h
g      equ      020h
h      equ      080h
Blank  equ      000h
LCD_Tab
EVEN
db      a+b+c+d+e+f          ; Выравнивание
db      b+c                  ; Отображение "0"
db      a+b+d+e+g            ; Отображение "1"
db      a+b+c+d+g            ; Отображение "2"
db      b+c+f+g              ; Отображение "3"
db      a+c+d+f+g            ; Отображение "4"
db      a+c+d+e+f+g          ; Отображение "5"
db      a+b+c                ; Отображение "6"
db      a+b+c+d+e+f+g        ; Отображение "7"
db      a+b+c+d+f+g          ; Отображение "8"
db      a+b+c+e+f+g          ; Отображение "9"
db      Blank                ; Отображение "A"
db      a+d+e+f              ; Отображение "Blank"
db      a+b+f+g              ; Отображение "C"
db      a+d+e+f+g            ; Отображение "degrees" o
db      a+e+f+g              ; Отображение "E"
db      a+e+f+g              ; Отображение "F"

```

```

;-----
;      COMMON  INTVEC          ; Вектора прерываний
;-----
ORG      BASICTIMER_VECTOR    ; Вектор прерывания от Basic Timer
DW      BT_ISR;
ORG      RESET_VECTOR         ; Вектор сброса
DW      RESET
END

```