

```
.device ATtiny2313
.include <tn2313def.inc>
.equ MCUclock=4000000    ;4MHz

;*** порты ЖКИ ***
.equ HD44780_port_out      =PORTB
.equ HD44780_port_in      =PINB
.equ HD44780_port_direction =DDRB

.equ HD44780_control_port  =PORTD

;*** Управляющие линии порта ЖКИ ***
.equ RS =6                ;RS = (1 - данные,                0 - команда)
.equ RW =5                ;RD = (1 - чтение данных из дисплея,    0 - запись данных в дисплей)
.equ EN =4                ;EN = (1 - установить тактовый сигнал,    0 - сбросить)
.equ BS =7                ;BS = (1 - дисплей занят,                0 - свободен)

;*** Описание строк ЖКИ ***
.equ str0=0x80
.equ str1=0xC0

;*** Описание позиций символов ЖКИ ***
.equ pos0=0
.equ pos1=1
.equ pos2=2
.equ pos3=3
.equ pos4=4
.equ pos5=5
.equ pos6=6
.equ pos7=7

;*** РОИ по умолчанию ***
;r0... r15
.def SREG_temp =r15
;r16... r31
.def temp0 =r16
.def temp1 =r17
.def temp2 =r18
```

```
.def temp3      =r19
.def temp4      =r20
.def temp5      =r21
.def temp6      =r22
.def temp7      =r23
.def PL         =r24    ;PL
.def PH         =r25    ;PH
;.def XL        =r26    ;XL
;.def XH        =r27    ;XH
;.def YL        =r28    ;YL
;.def YH        =r29    ;YH
;.def ZL        =r30    ;ZL
;.def ZH        =r31    ;ZH
```

```
.dseg
.org SRAM_START
;*** метка DATA стека ***
data_stack_label:
```

```
.cseg
.org 0x00
rjmp RESET      ;Reset Handler
rjmp INT0_I     ;External Interrupt0 Handler
rjmp INT1_I     ;External Interrupt1 Handler
rjmp TIM1_CAPT  ;Timer1 Capture Handler
rjmp TIM1_COMPA ;Timer1 CompareA Handler
rjmp TIM1_OVF   ;Timer1 Overflow Handler
rjmp TIM0_OVF   ;Timer0 Overflow Handler
rjmp USART0_RXC ;USART0 RX Complete Handler
rjmp USART0_DRE ;USART0,UDR Empty Handler
rjmp USART0_TXC ;USART0 TX Complete Handler
rjmp ANA_COMP   ;Analog Comparator Handler
rjmp PCINT      ;Pin Change Interrupt
rjmp TIMER1_COMPB ;Timer1 Compare B Handler
rjmp TIMER0_COMPA ;Timer0 Compare A Handler
rjmp TIMER0_COMPB ;Timer0 Compare B Handler
```

```

rjmp    USI_START           ;USI Start Handler
rjmp    USI_OVERFLOW        ;USI Overflow Handler
rjmp    EE_READY            ;EEPROM Ready Handler
rjmp    WDT_OVERFLOW        ;Watchdog Overflow Handler

```

Mes_lcd:

```

.db 18,"LCD WH2002A-YGH-CT " ;18 - количество символов в строке,"LCD WH2002A-YGH-CT " - сообщение внутри кавычек
;в конце сообщения желательно ставить пробел, если количество байт в строке + 1 байт число
;указатель на длину строки нечетные, таким образом чтобы общее число байт стало чётным.
;При этом пробел при выводе на дисплей не учитывается и на дисплей не выводится.

```

Mes_k:

```

.db 20,128,75,184,112,184,187,187,184,229,97,32,50,48,32,99,184,188,179,111,187,32

```

RESET:

```

;*** stack point init ***
ldi    temp0,low(RAMEND)    ;Загрузить метку RAMEND
out    SPL,temp0           ;SPL=RAMEND; 0xDF
;*** stack point init ***

;*** data stack init ***
ldi    YH,high (data_stack_label+32);загрузить data stack H
ldi    YL,low  (data_stack_label+32);загрузить data stack L
;*** end data stack init ***

;CLPR  DIV/1
ldi    temp0,low(0x80)
out    CLKPR,temp0        ;CLKPR=0x80
clr    temp0
out    CLKPR,temp0        ;CLKPR=0x00

;Input/Output Ports initialization

;Port A initialization
;Func2=In Func1=In Func0=In
;State2=T State1=T State0=T
out    PORTA,temp0        ;PORTA=0x00;
out    DDRA,temp0        ;DDRA=0x00;

```

```
;Port B initialization
;Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out
;State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0
ldi    temp0,low(0x00)
out    PORTB,temp0           ;PORTB=0x00;
ldi    temp0,low(0xFF)
out    DDRB,temp0           ;DDRB=0xFF;

;Port D initialization
;Func6=Out Func5=Out Func4=Out Func3=In Func2=In Func1=In Func0=In
;State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0
clr    temp0
out    PORTD,temp0         ;PORTD=0x00
ldi    temp0,low(0b01110000)
out    DDRD,temp0         ;DDRD=0x04;

;Timer/Counter 0 initialization
;Clock source: System Clock
;Clock value: Timer 0 Stopped
;Mode: Normal top=FFh
;OC0A output: Disconnected
;OC0B output: Disconnected
;TCCR0A=0x00;
clr    temp0
out    TCCR0A,temp0       ;TCCR0A=0x00
out    TCCR0B,temp0       ;TCCR0B=0x00
out    TCNT0,temp0        ;TCNT0=0x00
out    OCR0A,temp0        ;OCR0A=0x00
out    OCR0B,temp0        ;OCR0B=0x00

;Timer/Counter 1 initialization
;Clock source: System Clock
;Clock value: 0 Hz
;Mode: Normal top=FFFFh
;OC1A output: Discon.
;OC1B output: Discon.
;Noise Canceler: Off
;Input Capture on Falling Edge
```

```
;Timer 1 Overflow Interrupt: Off
;Input Capture Interrupt: Off
;Compare A Match Interrupt: Off
;Compare B Match Interrupt: Off
out    TCCR1A,temp0          ;TCCR1A=0x00
out    TCCR1B,temp0          ;TCCR1B=0x00
out    TCNT1H,temp0         ;TCNT1H=0x00
out    TCNT1L,temp0         ;TCNT1L=0x00
out    ICR1H,temp0          ;ICR1H=0x00
out    ICR1L,temp0          ;ICR1L=0x00
out    OCR1AH,temp0         ;OCR1AH=0x00
out    OCR1AL,temp0         ;OCR1AL=0x00
out    OCR1BH,temp0         ;OCR1BH=0x00
out    OCR1BL,temp0         ;OCR1BL=0x00

;External Interrupt(s) initialization
;INT0: Off
;INT1: Off
;Interrupt on any change on pins PCINT0-7: Off
out    GIMSK,temp0          ;GIMSK=0x00
out    MCUCR,temp0          ;MCUCR=0x00;

;Timer(s)/Counter(s) Interrupt(s) initialization
out    TIMSK,temp0          ;TIMSK=0x00

;UART initialization
;Mode: Disabled
;Clock source: Register & Counter=no clk.
;USI Counter Overflow Interrupt: Off
clr    temp0
out    USICR,temp0          ;USICR=0x00

;Analog Comparator initialization
;Analog Comparator: Off
;Analog Comparator Input Capture by Timer/Counter 1: Off
ldi    temp0,low(0x80)
out    ACSR,temp0          ;ACSR=0x80

;Watchdog Timer initialization
```

```
;Watchdog Timer Prescaler: OSC/1024k
;Watchdog Timer interrupt: Off
ldi    temp0,low(0x39)
out    WDTCR,temp0           ;WDTCR=0x39;
ldi    temp0,low(0x29)
out    WDTCR,temp0           ;WDTCR=0x29;

rcall  HD44780_init
```

main:

```
;*** Указание позиции вывода на дисплей ***
ldi    temp0,low(str0+pos0)
st     -Y,temp0
rcall  HD44780_write_position
;*** Конец указания вывода позиции ***

;*** Вывод сообщения из FLASH памяти ***
ldi    ZH,high (Mes_lcd*2)
ldi    ZL,low  (Mes_lcd*2)
rcall  HD44780_write_str_flash
;*** Конец вывода сообщения из FLASH памяти ***

;*** Указание позиции вывода на дисплей ***
ldi    temp0,low(str1+pos0)
st     -Y,temp0
rcall  HD44780_write_position
;*** Конец указания вывода позиции ***

;*** Вывод сообщения из FLASH памяти ***
ldi    ZH,high (Mes_k*2)
ldi    ZL,low  (Mes_k*2)
rcall  HD44780_write_str_flash
;*** Конец вывода сообщения из FLASH памяти ***
```

```
*** Вывод байта ***
ldi     temp0,low('0')
st      -Y,temp0
rcall   HD44780_write_symbol
*** Конец вывода байта
```

```
*** Очистка дисплея***
rcall   HD44780_clear
*** Конец очистки дисплея***
```

```
wdr           ;сброс сторожевого
rjmp        main ;возврат в начало
```

```
*** Функции работы с HD44780 ***
*** Инициализация ЖКИ ***
```

HD44780_init:

```
*** ввод последовательности 0x3-0x3-0x3-0x2 ***
ldi     temp0,low(0x03) ;загрузить 0x03
st      -Y,temp0       ;передать в стек
rcall   HD44780_write_init_nibble ;вызвать передачу тетрады
```

```
sbiw   Y,low(0x01) ;иммитировать передачу в стек
rcall   HD44780_write_init_nibble ;вызвать передачу тетрады
```

```
sbiw   Y,low(0x01) ;иммитировать передачу в стек
rcall   HD44780_write_init_nibble ;вызвать передачу тетрады
```

```
ldi     temp0,low(0x02) ;загрузить 0x02
st      -Y,temp0       ;передать в стек
rcall   HD44780_write_init_nibble ;вызвать передачу тетрады
```

```
*** ввод последовательности 0x28-0x08-0x01-0x06-0x0C ***
ldi     temp0,low(0x28)
st      -Y,temp0
rcall   HD44780_write_command
```

```
ldi    temp0,low(0x08)
st     -Y,temp0
rcall  HD44780_write_command

ldi    temp0,low(0x01)
st     -Y,temp0
rcall  HD44780_write_command

ldi    temp0,low(0x06)
st     -Y,temp0
rcall  HD44780_write_command

ldi    temp0,low(0x0C)
st     -Y,temp0
rcall  HD44780_write_command

ret

;*** Конец инициализации ЖКИ ***

;*** Вывод строки из FLASH памяти
HD44780_write_str_flash:
push   temp0
push   temp1
lpm    temp0,Z+
HD44780_write_str_flash_loop:
lpm    temp1,Z+
st     -Y,temp1
rcall  HD44780_write_symbol
dec    temp0
brne   HD44780_write_str_flash_loop

pop    temp1
pop    temp0

ret

;*** Конец вывода строки из FLASH памяти
```



```
    ;*** вывод на дисплей символа ***  
HD44780_write_symbol:  
    rcall    HD44780_wait_busy  
    sbi      HD44780_control_port,RS ;RS=1    (1 - данные, 0 - команды)  
    rcall    HD44780_write_byte  
    ret  
    ;*** конец вывода символа ***
```

```
    ;*** указание позиции ***  
HD44780_write_position:  
    rcall    HD44780_write_command  
    ret  
    ;*** конец записи в дисплей команды ***
```

```
    ;*** запись в дисплей команды ***  
HD44780_clear:  
    push    temp0  
    ldi     temp0,low(0x01)  
    st      -Y,temp0  
    rcall   HD44780_write_command  
    pop     temp0  
    ret  
    ;*** конец записи в дисплей команды ***
```

```
    ;*** запись в дисплей команды ***  
HD44780_write_command:  
    rcall    HD44780_wait_busy  
    cbi      HD44780_control_port,RS ;RS=0    (1 - данные, 0 - команда)  
    rcall    HD44780_write_byte
```

```

ret
;*** конец записи в дисплей команды ***

;*** запись младшей тетрады байта в дисплей ***
HD44780_write_init_nibble:
rcall  HD44780_delay_init
;Сохранение регистров в стеке
;Предустановка сигналов
cbi    HD44780_control_port,RS    ;RS = 1 (1 - данные, 0 - команда)
cbi    HD44780_control_port,RW    ;RD = 0 (1 - чтение данных из модуля, 0 - запись данных в модуль)
sbi    HD44780_control_port,EN    ;EN = 1 (1 - установить тактовый сигнал, 0 - сбросить)
;Приём байта из стека данных
ld     temp0,Y+                   ;пагрузка байта из стека данных в temp0
swap  temp0                       ;поменять местами полубайты
andi  temp0,0xF0                  ;обнулить младший полубайт в temp0
in    temp1,HD44780_port_in       ;загрузить предустановленные сигналы в порту вывода дисплея
andi  temp1,0x0F
or    temp0,temp1                 ;Совместить предустановленные сигналы (в младшем полубайте порта вывода дисплея) и старший
полубайт данных регистра temp0
;Передаём старший полубайт
out    HD44780_port_out,temp0     ;lcd_port = high_nibble
rcall  HD44780_clk                ;тактировать
;Восстановление регистров из стека
ret
;*** Конец записи тетрады в дисплей ***

;*** Запись байта в дисплей ***
HD44780_write_byte:
push  temp0
push  temp1
push  temp2
push  temp3
;предустановка сигналов
cbi    HD44780_control_port,RW    ;RD = 0 (1 - чтение данных из дисплея, 0 - запись данных в дисплей)
;выгрузить байт из стека данных

```

```

ld      temp1,Y+                ;загрузить содержимое стека данных в регистр temp0
mov     temp2,temp1             ;копировать temp0 в temp1
andi   temp1,low(0xF0)         ;обнулить младшие байты в temp0
swap   temp2                   ;поменять тетрады в temp1
andi   temp2,low(0xF0)         ;обнулить младшие байты в temp1
in     temp3,HD44780_port_in    ;загрузить содержимое порта дисплея
andi   temp3,low(0x0F)         ;обнулить старшие байты в temp2
or     temp1,temp3             ;совместить содержимое temp0 и temp2
or     temp2,temp3             ;совместить содержимое temp1 и temp2
;выгрузить в порт дисплея сперва старший, после младший полубайты
out    HD44780_port_out,temp0   ;lcd_port = high_nibble
rcall  HD44780_clk              ;тактировать
out    HD44780_port_out,temp1   ;lcd_port = low_nibble
rcall  HD44780_clk              ;тактировать
;конец выгрузки в порт дисплея
pop    temp3
pop    temp2
pop    temp1
pop    temp0
ret

;*** Конец записи байта в дисплей ***

;*** Проверка и ожидание занятости дисплея ***
HD44780_wait_busy:
;*** переключить старшую тетраду порта дисплея на ввод данных
in     temp0,HD44780_port_direction ;считать содержимое порта направления дисплея
andi   temp0,low(0x0F)             ;обнулить 4 старших разряда
out    HD44780_port_direction,temp0 ;вернуть в порт очищенные разряды
;*** убраться на старшей тетраде порта дисплея все подтяжки
in     temp0,HD44780_port_out       ;загрузить содержимое порта дисплея
andi   temp0,low(0x0F)             ;обнулить 4 старших разряда
out    HD44780_port_out,temp0       ;вернуть в порт очищенные разряды
;*** принудительно сбросить бит BS
cbi    HD44780_control_port,RS      ;RS = 0 (1 - данные,           0 - команда)
sbi    HD44780_control_port,RW      ;RD = 1 (1 - чтение данных из дисплея, 0 - запись данных в дисплей)

HD44780_wait_busy_loop:

```

```

rcall    HD44780_rise_clk
in       temp1,HD44780_port_in           ;загрузить в temp0 содержимое порта HD44780_port_in
rcall    HD44780_fall_clk                ;установить тактовый сигнал + подождать
rcall    HD44780_clk                     ;тактировать
sbrc     temp1,BS                        ;проверить бит занятости
rjmp     HD44780_wait_busy_loop
;*** переключить старшую тетраду порта дисплея на вывод данных
in       temp0,HD44780_port_direction
ori      temp0,low(0xF0)
out      HD44780_port_direction,temp0
;*** ВЫХОД
ret
;*** конец проверки и ожидания ***

;*** такт CLK ***
HD44780_clk:
rcall    HD44780_rise_clk
rcall    HD44780_fall_clk
ret
;*** конец такта CLK ***
;*** нарастающий импульс ***
HD44780_rise_clk:
sbi      HD44780_control_port,EN ;EN = 1 (1 - установить тактовый сигнал, 0 - сбросить)
rcall    HD44780_clk_delay          ;задержка CLK
ret
;*** конец нарастающего импульса ***
;*** ниспадающий импульс ***
HD44780_fall_clk:
cbi      HD44780_control_port,EN ;EN = 1 (1 - установить тактовый сигнал, 0 - сбросить)
rcall    HD44780_clk_delay          ;задержка CLK ;
ret
;*** конец ниспадающего импульса ***
;*** задержка CLK ***
HD44780_clk_delay:
ldi      temp0,low(((MCUclock*8)/3000000)-1)
HD44780_clk_delay_loop:
dec      temp0                       ;temp0 = temp0 - 1

```

```
brne    HD44780_clk_delay_loop    ;
ret
;***  koniec  задержки  CLK  ***

;***  задержка  wait  init  ***
HD44780_delay_init:
ldi     PL,low  (((MCUclock*40)/6000)-1)
ldi     PH,high (((MCUclock*40)/6000)-1)
ldi     XL,byte3(((MCUclock*40)/6000)-1)
HD44780_delay_init_loop:
wdr
sbiw    PL,low  (0x01)
sbci    XL,low  (0x00)          ;
brne    HD44780_delay_init_loop   ;

ret
;***  koniec  задержки  wait  init  ***

;***  koniec  Функций  HD44780  ***
;***  Конеч  всех  подпрограмм  ***
```

```
INT0_I:           ;External Interrupt0 Handler
INT1_I:           ;External Interrupt1 Handler
TIM1_CAPT:        ;Timer1 Capture Handler
TIM1_COMPA:       ;Timer1 CompareA Handler
TIM1_OVF:         ;Timer1 Overflow Handler
TIM0_OVF:         ;Timer0 Overflow Handler
USART0_RXC:       ;USART0 RX Complete Handler
USART0_DRE:       ;USART0,UDR Empty Handler
USART0_TXC:       ;USART0 TX Complete Handler
```

```
ANA_COMP:                ;Analog Comparator Handler
PCINT:                   ;Pin Change Interrupt
TIMER1_COMPB:            ;Timer1 Compare B Handler
TIMER0_COMPA:            ;Timer0 Compare A Handler
TIMER0_COMPB:            ;Timer0 Compare B Handler
USI_START:               ;USI Start Handler
USI_OVERFLOW:            ;USI Overflow Handler
EE_READY:                ;EEPROM Ready Handler
WDT_OVERFLOW:
    reti
```