**Лабораторная работа №1**

**Тема:** ---

**Задание:** ---

**Код программы:**

.include "m16def.inc"

.def temp=r16

.equ const1=25

.equ const2=0x15

.equ const3=0b01001100

.cseg

.org 0

jmp RESET ; Reset Handler

jmp EXT\_INT0 ; IRQ0 Handler

jmp EXT\_INT1 ; IRQ1 Handler

jmp TIM2\_COMP ; Timer2 Compare Handler

jmp TIM2\_OVF ; Timer2 Overflow Handler

jmp TIM1\_CAPT ; Timer1 Capture Handler

jmp TIM1\_COMPA ; Timer1 CompareA Handler

jmp TIM1\_COMPB ; Timer1 CompareB Handler

jmp TIM1\_OVF ; Timer1 Overflow Handler

jmp TIM0\_OVF ; Timer0 Overflow Handler

jmp SPI\_STC ; SPI Transfer Complete Handler

jmp USART\_RXC ; USART RX Complete Handler

jmp USART\_UDRE ; UDR Empty Handler

jmp USART\_TXC ; USART TX Complete Handler

jmp ADC\_C ; ADC Conversion Complete Handler

jmp EE\_RDY ; EEPROM Ready Handler

jmp ANA\_COMP ; Analog Comparator Handler

jmp TWSI ; Two-wire Serial Interface Handler

jmp EXT\_INT2 ; IRQ2 Handler

jmp TIM0\_COMP ; Timer0 Compare Handler

jmp SPM\_RDY ; Store Program Memory Ready Handler

;

;Инициализация

RESET:

ldi temp,high(RAMEND) ; Set Stack Pointer to top of RAM

out SPH,temp

ldi temp,low(RAMEND)

out SPL,temp

ldi temp,0x3F ; настройка порта D на выход

out DDRD,temp

rcall initlcd ; инициализация ЖКИ

;Основной цикл

Loop: ldi r24, 20 ;установка счетчика символов

shift: ldi r25,$18 ;сдвиг дисплея на один символ

rcall icom\_2313

rcall ddelay ;задержка ~200 мс при кварце 8 МГц

dec r24

brne shift

ldi r24,5 ;остановка после вывода каждой строки

stop: rcall ddelay

dec r24

brne stop

rjmp Loop ;повторить сначала

initlcd:

ldi r25,$20 ;4-битный интерфейс, 1-строка, шрифт-5х7 точек

rcall icom\_2313

rcall del

ldi r25,$0C ;включить дисплей, выключить курсор

rcall icom\_2313

rcall del

ldi r25,$20 ;4-битный интерфейс, 1-строка, шрифт-5х7 точек

rcall icom\_2313

rcall del

ldi r25,$1 ;очистить дисплей и установить курсор в

rcall icom\_2313 ; начальную позицию

rcall delay

;Вывод в DDRAM постоянного текста

ldi r25,$80 ;после этой команды данные записываются в DDRAM

rcall icom\_2313 ;начиная с адреса 0

rcall del

ldi r30,low(text0\*2) ;установить указатель Z на начало буфера text0

ldi r31,high(text0\*2)

ldi r24,80 ;установить счетчик программируемых ячеек DDRAM

ddram:

lpm ;извлечь из flash-памяти байт данных

mov r25,r0

rcall idat\_2313 ;записать его в DDRAM

rcall del

ld r1,z+ ;увеличить на 1 указатель Z адреса flash-памяти

dec r24 ;если еще не все ячейки запрограммированы

brne ddram ;вернуться назад

ret

;Задержка вывода для удобства восприятия человеком

ddelay:

ldi r19,100

dd: rcall delay

dec r19

brne dd

ret

; 1+(13828+3+1+2)\*99+(13828+3+1+1)\*1+4=1383405 тактов (173 мс при 8 МГц)

del:

ldi r18,100 ;задержка ~40 мкс при кварце 8 МГц; 1 такт

l4: dec r18 ;1 такт

brne l4 ;1/2 такт

ret ;4 такта

; 1+(1+2)\*99+(1+1)\*1+4=304 такта (при 8 МГц 38 мкс)

delay:

ldi r17,18 ;задержка ~1,8 ms при кварце 8 МГц

l: ldi r18,255

l1: dec r18

brne l1

dec r17

brne l

ret

; Т0=1+(1+2)\*254+(1+1)\*1=765

; 1+(Т0+1+2)\*17+(Т0+1+1)\*1+4=13828 тактов (1,73 мкс при 8 МГц)

icom\_2313:

swap r25 ;подготовить старшую тетраду для передачи

mov r26,r25

andi r26,$ef ;очистка бита 4 порта D (сигнал RS=0)

ori r26,$e0 ;установка бита 5 порта D ( сигнал E=1)

out portd,r26

cbi portd, 5 ;стробирующий сброс бита 5 порта D ( сигнал E=0).

swap r25 ;подготовить младшую тетраду для передачи

mov r26,r25

andi r26,$ef ;очистка бита 4 порта D (сигнал RS=0)

ori r26,$e0 ;установка бита 5 порта D ( сигнал E=1)

out portd,r26

cbi portd, 5 ;стробирующий сброс бита 5 порта D ( сигнал E=0)

ret

idat\_2313:

swap r25 ;подготовить старшую тетраду для передачи

mov r26,r25

ori r26,$f0 ;установка бита 4 порта D (сигнал RS=1) и

out portd,r26 ;бита 5 порта D ( сигнал E=1)

cbi portd,5 ;стробирующий сброс бита 5 порта D ( сигнал E=0)

swap r25 ;подготовить младшую тетраду для передачи

mov r26,r25

ori r26,$f0 ;установка бита 4 порта D (сигнал RS=1) и

out portd,r26 ;бита 5 порта D ( сигнал E=1)

cbi portd,5 ; стробирующий сброс бита 5 порта D ( сигнал E=0)

ret

EXT\_INT0: ; IRQ0 Handler

EXT\_INT1: ; IRQ1 Handler

TIM2\_COMP: ; Timer2 Compare Handler

TIM2\_OVF: ; Timer2 Overflow Handler

TIM1\_CAPT: ; Timer1 Capture Handler

TIM1\_COMPA: ; Timer1 CompareA Handler

TIM1\_COMPB: ; Timer1 CompareB Handler

TIM1\_OVF: ; Timer1 Overflow Handler

SPI\_STC: ; SPI Transfer Complete Handler

USART\_RXC: ; USART RX Complete Handler

USART\_UDRE: ; UDR Empty Handler

USART\_TXC: ; USART TX Complete Handler

ADC\_C: ; ADC Conversion Complete Handler

EE\_RDY: ; EEPROM Ready Handler

ANA\_COMP: ; Analog Comparator Handler

TWSI: ; Two-wire Serial Interface Handler

EXT\_INT2: ; IRQ2 Handler

TIM0\_COMP: ; Timer0 Compare Handler

SPM\_RDY: ; Store Program Memory Ready Handler

TIM0\_OVF: ; Timer0 Overflow Handler

reti

;Во flash - памяти после программы размещаем постоянные данные

text0:

.DB ' ' , ' ' , ' ' , ' ' , ' ' , ' ' , ' ' , ' ' , ' ' , ' ' ;Пустая строка

.DB ' ' , ' ' , ' ' , ' ' , ' ' , ' ' , ' ' , ' ' , ' ' , ' '

text1:

.DB ' ',' ',' ','I',' ','l','o','v','e',' ';I love my AVR

.DB 'm','y',' ','A','V','R',' ',' ',' ',' '

text2:

.DB ' ',' ',' ','I',' ','l','o','v','e',' ';I love my LCD

.DB 'm','y',' ','L','C','D',' ',' ',' ',' '

text3:

.DB ' ',' ',' ',' ','I',' ','l','o','v','e';I love НУК

.DB ' ',' ','H','Y','K',' ',' ',' ',' ',' '

**Вывод:** в данной лабораторной работе мы научились писать программу для осуществления паралельного вывода информации, разработали блок-схему и електрическую схему для задания.