



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Компьютерные системы и сети»

НАПРАВЛЕНИЕ ПОДГОТОВКИ «09.03.01 ПРИКЛАДНАЯ ИНФОРМАТИКА»

ОТЧЕТ по лабораторной работе № 2

Название: «Обработка внешних прерываний в микроконтроллерах AVR»

Дисциплина: «Микропроцессорные системы»

Вариант № 19

Студент ИУ6-62Б
(Группа)

(Подпись, дата)

А.Е.Медведев
(И. О. Фамилия)

Преподаватель

(Подпись, дата)

Б.И. Бычков
(И. О. Фамилия)

2022 г.

1 Цель работы:

- изучение системы прерываний микроконтроллеров AVR,
- освоение системы команд микроконтроллеров AVR,
- ознакомление с работой стека при вызове подпрограмм и обработчиков прерываний,
- программирование внешних прерываний.

2 Задания:

2.1 Задание 1

Запустив AVR Studio, проверить работу программы в шаговом режиме. С целью ускорения отладки сократить время задержек до минимума. Проконтролировать работу стека при вызове подпрограмм delay1, delay2.

Листинг 2.1 – Код реализации прерывания на плате ATmega8515

```
.include "m8515def.inc"          ;???? ????????????? ??? ATmega8515
.def temp = r16                  ;???????????? ????????
.equ led = 0                     ;0-? ??? ?????? PB
.equ sw0 = 0                     ;0-? ??? ?????? PA
.equ sw1 = 1                     ;1-? ??? ?????? PA

.org $000
    rjmp INIT                    ;???????????? ????????

INIT:    ldi temp,$5F             ;????????????
        out SPL,temp            ; ????????????? ??????
        ldi temp,$02            ; ?? ?????????????
        out SPH,temp            ; ???????? ???
        ser temp                ;???????????????????? ????????
        out DDRB,temp           ; ?????? PB ?? ??????
        out PORTB,temp          ;???????????? LED
        clr temp                ;????????????????????
        out DDRA,temp           ; ?????? PA ?? ?????

        ldi temp,0b00000011     ;???????????? ?????????????????
```

```

        out PORTA,temp        ; ?????????? ????? PA
test_sw0:
        sbic PINA,sw0        ; ?????????? ??????????
        rjmp test_sw1        ; ?????? sw0
        cbi PORTB, led
        rcall delay1
        sbi PORTB,led
wait_0:
        sbis PINA,sw0
        rjmp wait_0
test_sw1:
        sbic PINA,sw1        ; ?????????? ??????????
        rjmp test_sw0        ; ?????? sw1
        cbi PORTB,led
        rcall delay2
        sbi PORTB,led
wait_1:
        sbis PINA,sw1
        rjmp wait_1
        rjmp test_sw0
delay1:                                ; ?????????????? 1 ?
        ret
delay2:                                ; ?????????????? 2 ?
        rcall delay1
        rcall delay1
        ret

```

При вызове подпрограммы или функции в тек помещается адрес возврата. По условию задания начало стека установлена в конец секции данных. Работы стека продемонстрирована на рисунке 2.1

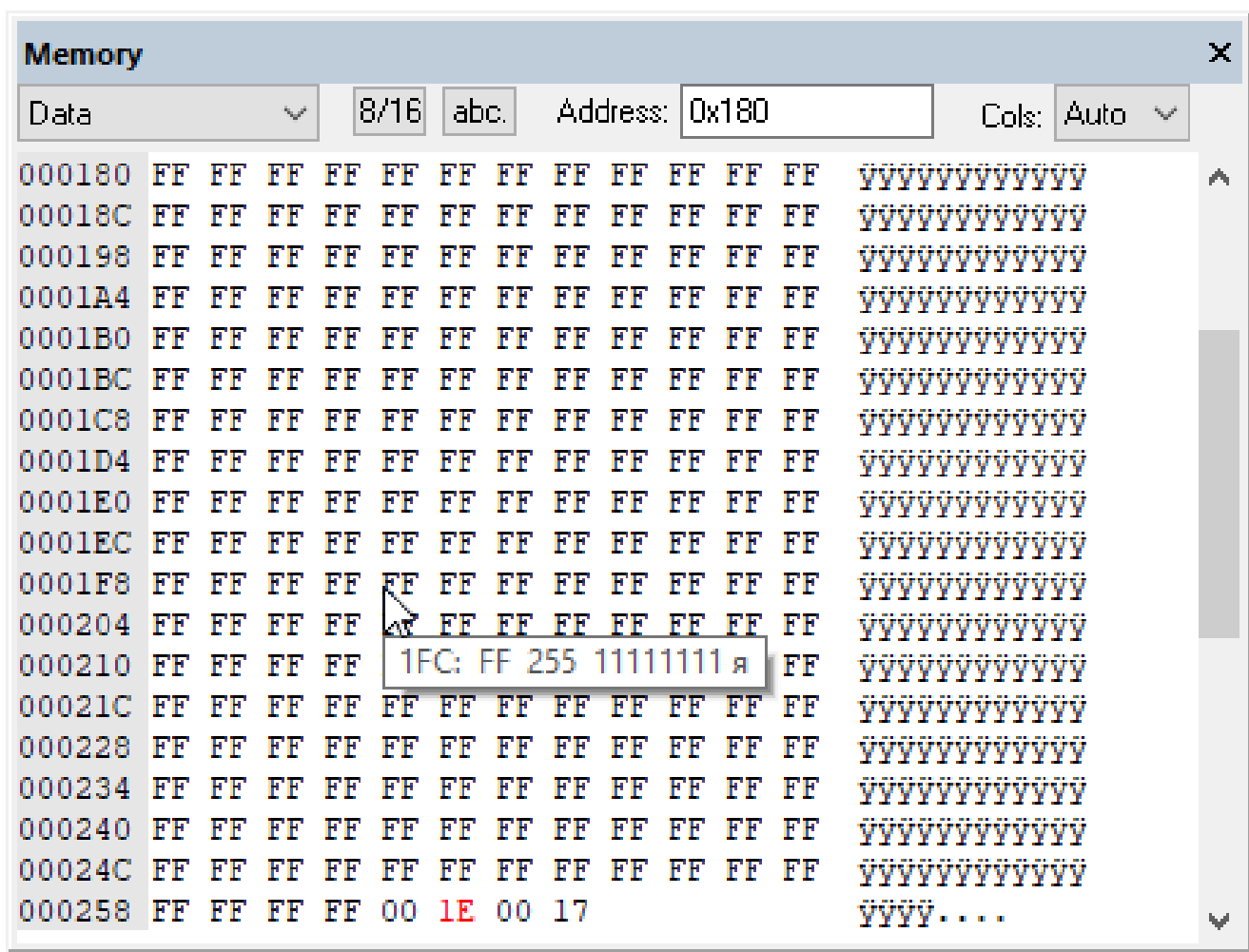


Рисунок 2.1 – Состояние стека при вызове подпрограммы или функции

2.2 Задание 2

Вносим изменения и дополнения в исходный текст программы 2.1, касающиеся обработки прерываний. На этапе инициализации указываются область стека для сохранения адресов возврата, при необходимости адреса векторов прерываний и сами векторы, маска прерываний, общее разрешение прерываний. Завершаем инициализацию переводом процессора в фоновый режим ожидания:

Листинг 2.2 – Код реализации прерывания на плате ATmega8515

```
.include "m8515def.inc"      ;???? ???????????? ??? ATmega8515
.def temp = r16              ;??????????? ????????
.equ led = 0                  ;0-? ??? ?????? PB
.equ sw0 = 0                  ;0-? ??? ?????? PA
.equ sw1 = 1                  ;1-? ??? ?????? PA
```

```

.org $000
    rjmp INIT                ;????????? ??????

INIT:    ldi temp,$5F        ;?????????
    out SPL,temp            ; ?????????? ??????
    ldi temp,$02            ; ?? ??????????
    out SPH,temp            ; ??????? ??
    ser temp                ;????????????????? ????????
    out DDRB,temp           ; ?????? PB ?? ??????
    out PORTB,temp          ;?????????? LED
    clr temp                ;?????????????????
    out DDRA,temp           ; ?????? PA ?? ?????

    ldi temp,0b00000011    ;????????????? ??????????????????
    out PORTA,temp          ; ?????????????? ?????? PA

test_sw0:
    sbic PINA,sw0           ;?????????? ???????????
    rjmp test_sw1           ; ??????? sw0
    cbi PORTB, led
    rcall delay1
    sbi PORTB,led

wait_0:
    sbis PINA,sw0
    rjmp wait_0

test_sw1:
    sbic PINA,sw1           ;?????????? ???????????
    rjmp test_sw0           ; ??????? sw1
    cbi PORTB,led
    rcall delay2
    sbi PORTB,led

wait_1:
    sbis PINA,sw1
    rjmp wait_1
    rjmp test_sw0

delay1:                ; ?????????????????? 1 ?
    ret

delay2:                ; ?????????????????? 2 ?
    rcall delay1

```

```
rcall delay1
ret
```

Состояние регистров и стека продемонстрировано на рисунках 2.2 — 2.3.

000228	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿ
000234	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿ
000240	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿ
00024C	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿ
000258	FF	FF	00	24	00	1D	00	14					ÿÿ.\$....

Рисунок 2.2 – Состояние стека при вызове подпрограммы или функции

































































































































































































	SPCR	0x30 (0x50)	0x00								
	SP	0x3D (0x5D)	0x0259								
	SPMCR	0x37 (0x57)	0x00								
	SREG	0x3F (0x5F)	0x02								
	EEPROM										
	EXTERNAL_I										
	GICR	0x3B (0x5B)	0xC0								
	GIFR	0x3A (0x5A)	0x00								
	PORTA										
	PORTB										
	DDRB	0x17 (0x37)	0xFF								
	PINB	0x16 (0x36)	0xFE								
	PORTB	0x18 (0x38)	0xFE								
	PORTC										
	PORTD										
	DDRD	0x11 (0x31)	0x00								
	PIND	0x10 (0x30)	0x04								
	PORTD	0x12 (0x32)	0x0C								
	PORTE										
	DDRE	0x06 (0x26)	0x00								
	PINE	0x05 (0x25)	0x00								
	PORTE	0x07 (0x27)	0x00								
	SPI										
	SPCR	0x0D (0x2D)	0x00								
	SPDR	0x0F (0x2F)	0x00								
	SPSR	0x0E (0x2E)	0x00								
	TIMER_COUNT										
	TIMER_COUNT										
	USART										
	UBRRH	0x20 (0x40)	0x00								
	UBRL	0x09 (0x29)	0x00								
	UCSRA	0x0B (0x2B)	0x20								

Рисунок 2.3 – Состояние регистров

2.3 Задание 3

Изменить программу для микроконтроллера ATmega8515, заменив прерывание INT0 на INT2.

Листинг 2.3 – Код реализации прерывания на плате ATmega8515

```
;?????????: SW0-PD0, SW2-PD2, LED-PB
;*****
;.include "8515def.inc"
.include "m8515def.inc"
.def temp = r16
.def reg_led = r20
.def key = r21
.equ START = 0
.org $000

rjmp INIT
.org $002
rjmp START_PRESSED ;????????? ?????????? ?????????? INT1(START)
.org $00D
rjmp STOP_PRESSED ; ?????????? ?????????? ?????????? INT2(STOP)

INIT:
    ldi temp,$5F
    out SPL,temp
    ldi temp,$02
    out SPH,temp

    ldi reg_led,0x3F
    sec ;C=1
    clt ;T=0
    clr key

    ser temp
    out DDRB,temp
    out PORTB,temp
    clr temp
    out DDRD,temp
```

```

    ldi temp,0x08
    out PORTD, temp
    clr temp
    out DDRE,temp
    ldi temp,0x1
    out PORTE,temp
    ldi temp,(1<<INT2)| (1<<INT1)
    out GICR,temp
    ldi temp,0x00
    out MCUCR,temp
    out MCUCSR,temp
    sei
WAIT:
    sbrs key,0
    rjmp WAIT
    out PORTB,reg_led
    rcall DELAY

    brts LEFT

; sbrs reg_led,1
    mov r27, reg_led
    andi r27, 0x01
    brne R_OFFSET
    set
    ldi reg_led,0x03
    clc
    rjmp WAIT
;end if

R_OFFSET:
    ror reg_led
    rjmp WAIT

LEFT:
    ;sbrc reg_led,6
    mov r27, reg_led
    andi r27, 0x3F
    brne L_OFFSET

```



```

        clt
        ldi reg_led,0x3F
        sec
        rjmp WAIT
        ;end if
L_OFFSET:
        rol reg_led
        rjmp WAIT

STOP_PRESSED:
        ldi key,0
        reti
START_PRESSED:
        ldi key,1
        reti

DELAY: ldi r17,11
d1: ldi r18,255
d2: ldi r19,236
d3: dec r19
        brne d3
        dec r18
        brne d2
        dec r17
        brne d1
        ret

```

На рисунке 2.4 продемонстрировано состояние программы в процессе отладки. На данном рисунке можно определить работу стека, изменения состояния регистров и флагов.

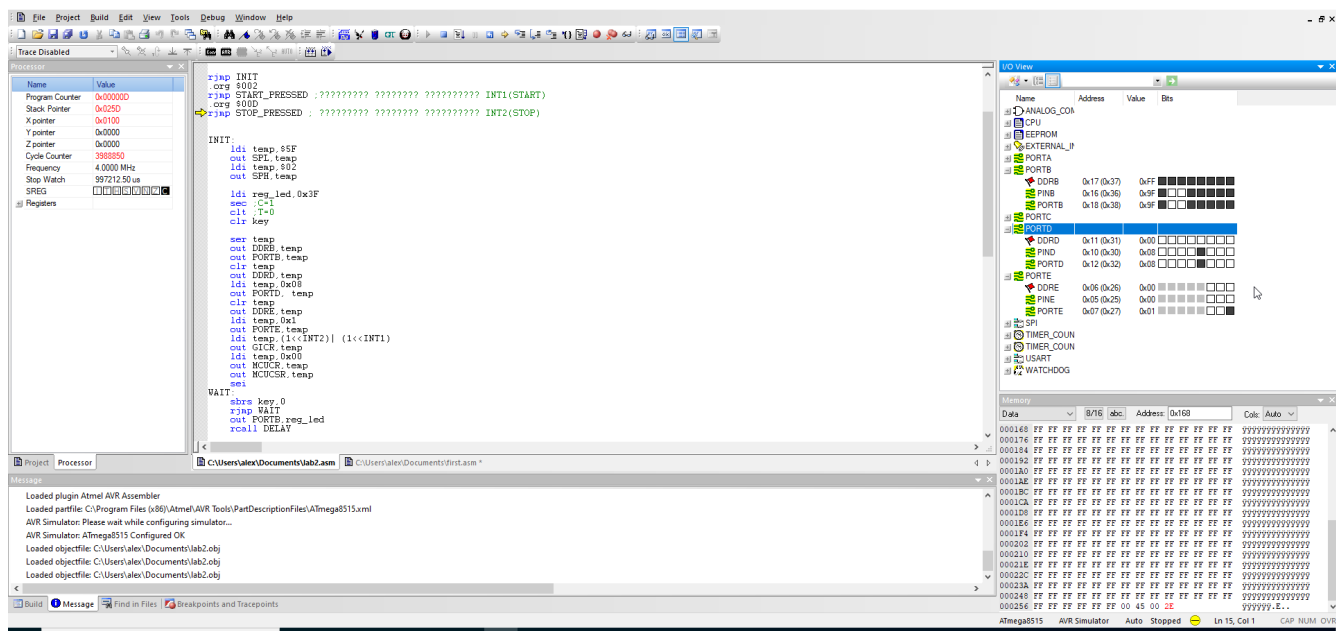


Рисунок 2.4 – Состояние программы в процессе отладки

2.4 Задание 4

Запустив программу Proteus ISIS, собрать проект, включающий микроконтроллер, 2 кнопки и 8 светодиодов.

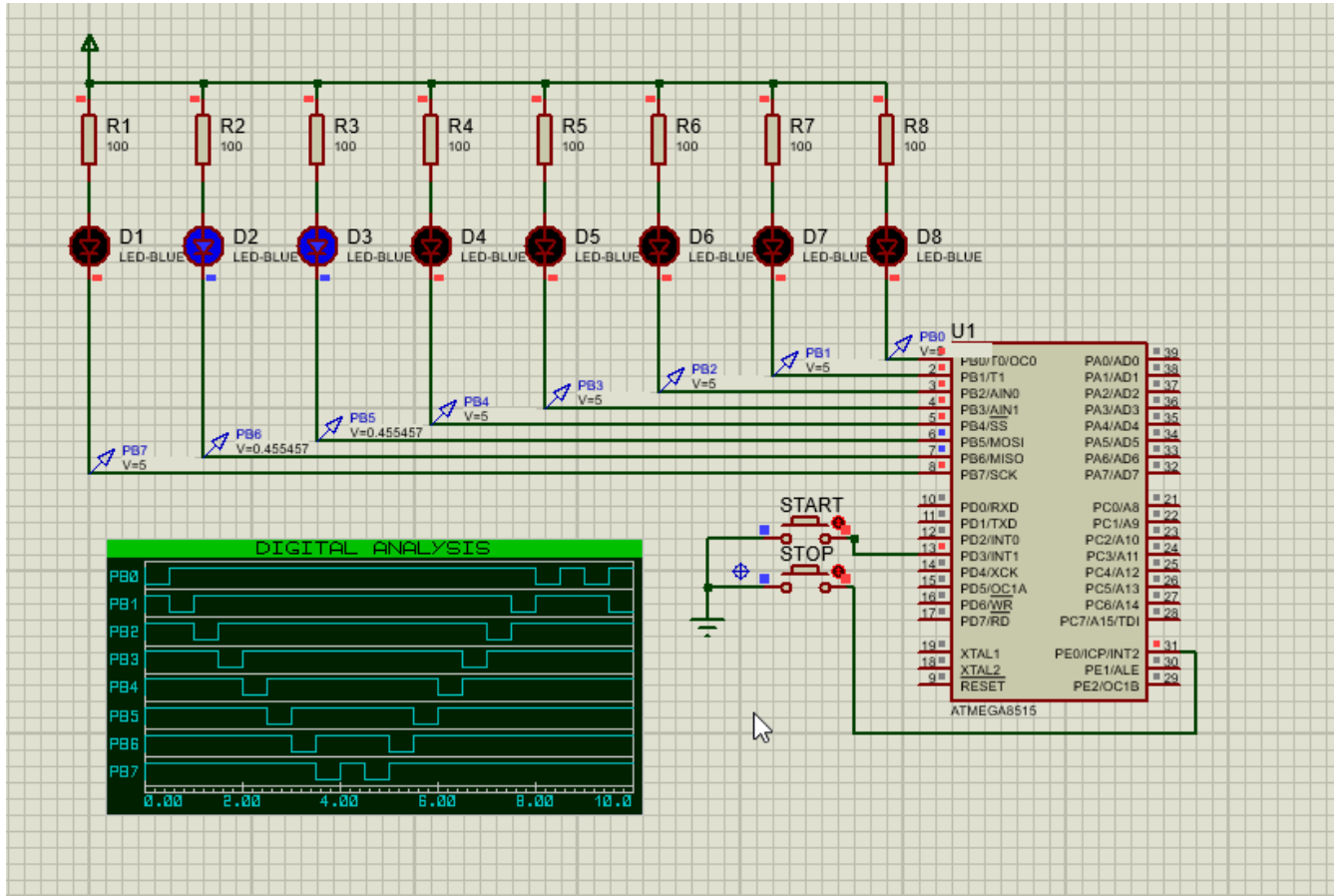


Рисунок 2.5 – Схема прерываний в протеусе

3 Вывод:

В ходе лабораторной работы были изучены прерывание микроконтроллера семейства ATx8515.