|  |  |
| --- | --- |
| Gerb-BMSTU_01**в** | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное автономное 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 (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01 Информатика и вычислительная техника**

**Отчет**

**по лабораторной работе № 2**

**Название: Обработка внешних прерываний в микроконтроллерах AVR**

**Дисциплина: Микропроцессорные системы**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-63Б |  |  | В.К. Залыгин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Е.Ю. Гаврилова |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2025

**Цель работы:**

* изучение системы прерываний микроконтроллеров AVR;
* освоение системы команд микроконтроллеров AVR;
* ознакомление с работой стека при вызове подпрограмм и обработчиков прерываний.

**Практическая часть**

**Задание 1**

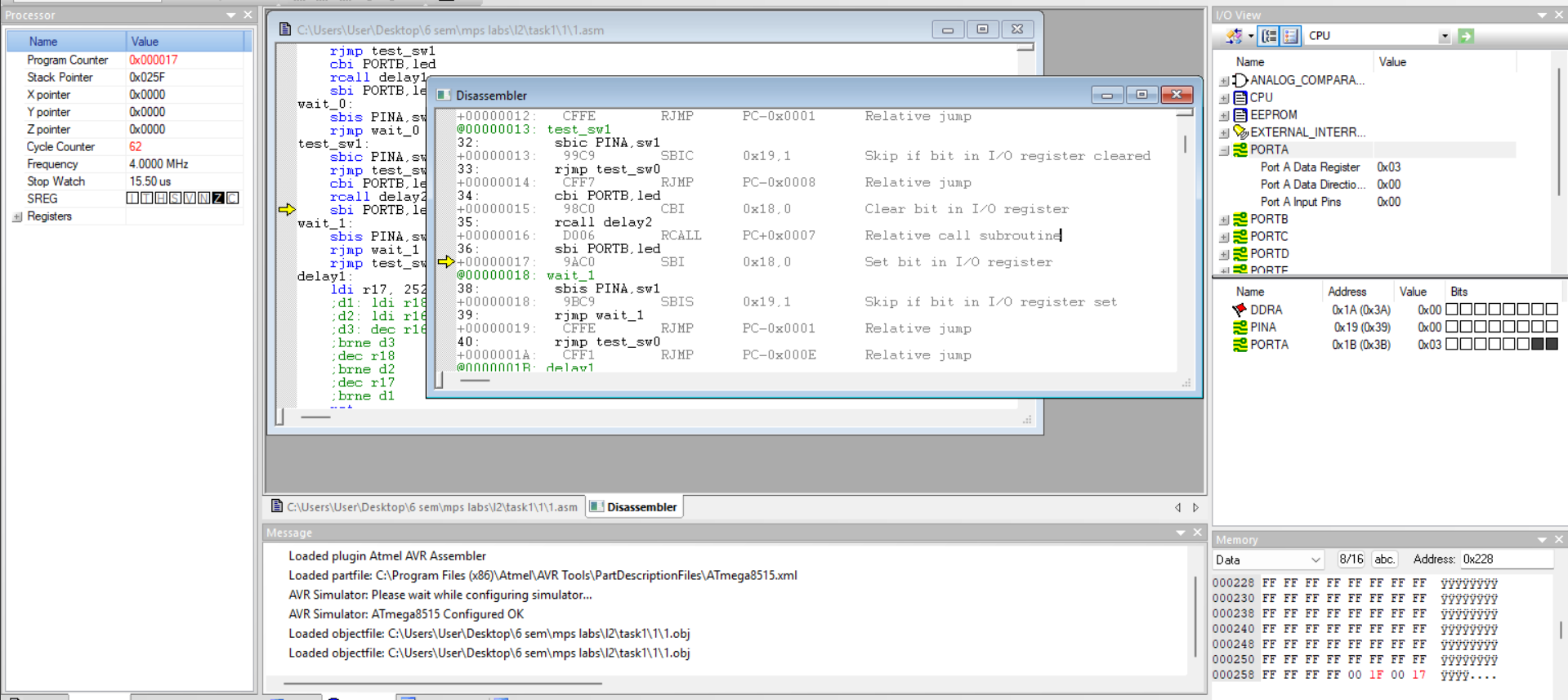
****

Рисунок 1 – указатель SP,PC (верхний левый угол), состояние стека (нижний правый угол), дизасемблер (по центру)

Как видно, в стеке находится 2 адреса возврата – следующей команды после rcall delay2 и адрес команды ret в delay2.

**Задание 2**

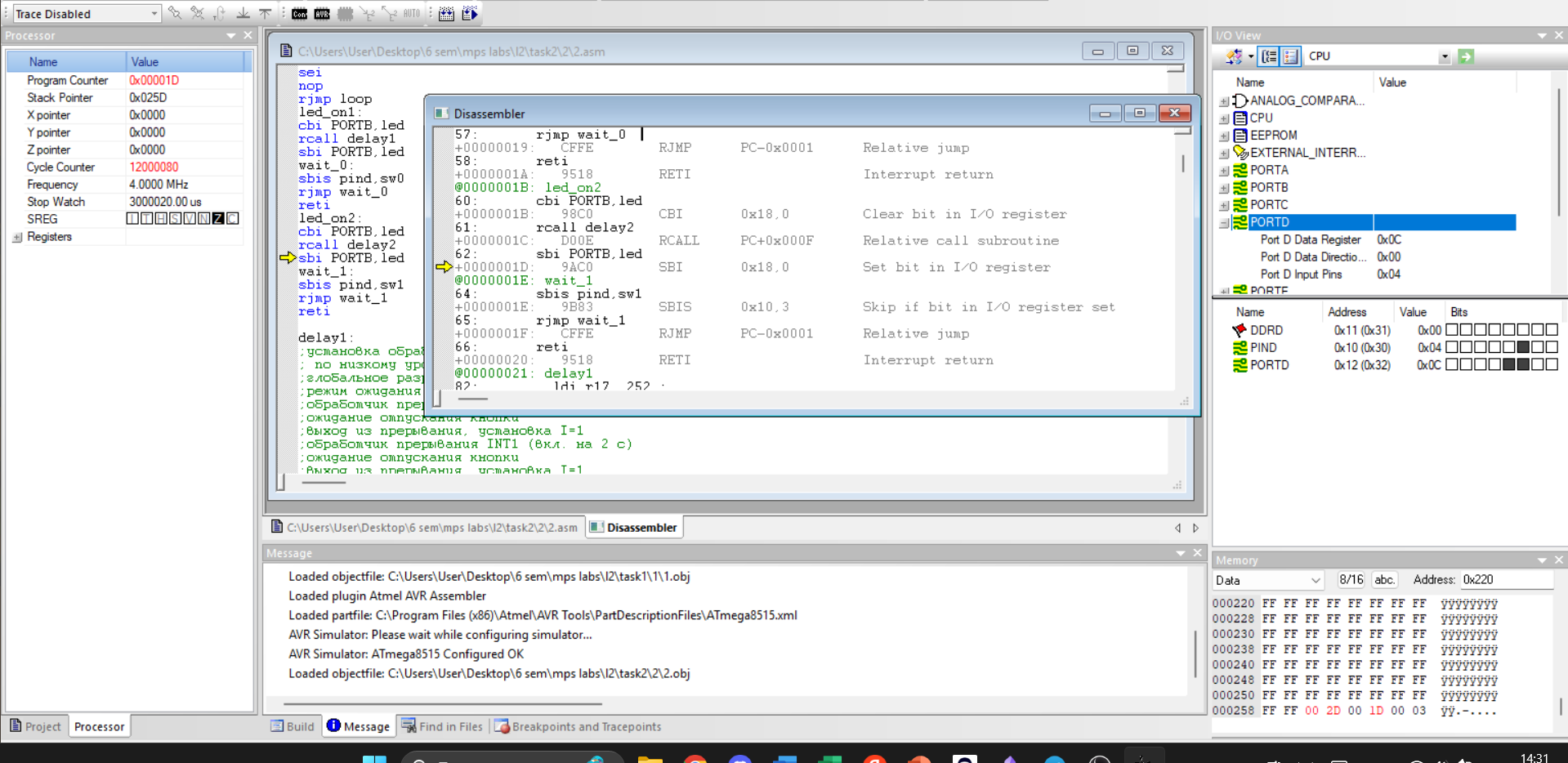


Рисунок 2 – указатель SP,PC (верхний левый угол), состояние стека (нижний правый угол), дизасемблер (по центру)

Как видно, в стеке находится 3 адреса возврата, для delay2, для ret внутри delay2 и для INIT.

**Задание 3**

Результат выполнения задания представлен ниже в листинге 1.

Листинг 1 – Код программы

|  |
| --- |
| .include "m8515def.inc"  .def job = r25  .def reg\_led = r24  .def temp = r19  .equ START\_BUTTON = 3  .equ STOP\_BUTTON = 0  .equ WORKING = 1  .equ WAITING = 0  .def lupa = r21  .equ A = 1  .equ LUPA1 = 1  .equ LUPA2 = 7  .org $000  rjmp INIT  nop  rjmp start  nop  nop  nop  nop  nop  nop  nop  nop  nop  nop  rjmp stop  INIT:  ldi job, WAITING  ldi temp,high(RAMEND)  out SPH,temp  ldi temp,low(RAMEND)  out SPL,temp  ser temp  out DDRC,temp  out PORTC,temp  ldi reg\_led,0b10000000  out PORTC,reg\_led  clr temp  out DDRD,temp  out DDRE, temp  ldi temp,0b00001000  out PORTD,temp  ldi temp,0b00000001  out PORTE,temp  ldi temp,((1<<INT1)|(1<<INT2))  out GICR,temp  ldi temp,0  out MCUCR,temp  ldi reg\_led, 1  sei  wait:  cpi job, WAITING  brne work  rjmp wait  work:  cpi job, WORKING  brne wait  ;\*\*\*От 7 до 1 один выключен, от 1 до 7 один включен\*\*\*  sbrs lupa,A  rjmp nex  sbrs reg\_led,LUPA1  rjmp left  clr lupa  ldi reg\_led,0b11111101  sec  rjmp next  left:  ror reg\_led  ror reg\_led  rjmp next  nex:  sbrc reg\_led,LUPA2  rjmp right  ser lupa  ldi reg\_led,0b10000000  clc  rjmp next  right:  rol reg\_led  rol reg\_led  next:  out PORTC,reg\_led ;вывод содержимого регистра в порт  rjmp work  start:  ldi job, WORKING  wait\_start:  sbis pind,START\_BUTTON  rjmp wait\_start  reti  stop:  ldi job, WAITING  wait\_stop:  sbis pine,STOP\_BUTTON  rjmp wait\_stop  reti  delay:  ldi r16,250  d0:  ldi r17,250  d1:  ldi r18,8  d2: dec r18  brne d2  dec r17  brne d1  dec r16  brne d0  ret |

**Задание 4**

Полученный код программы представлен ниже в листинге 2.

Листинг 2 – Код программы

|  |
| --- |
| .include "m8515def.inc"  .def temp = r16  .equ led = 0  .equ sw0 = 0  .equ sw1 = 1  .org $000  rjmp INIT  nop  nop  nop  nop  nop  nop  nop  nop  nop  nop  nop  nop  rjmp led\_on  INIT:  ldi temp,high(RAMEND)  out SPH,temp  ldi temp,low(RAMEND)  out SPL,temp  ser temp  out DDRB,temp  out PORTB,temp  clr temp  out DDRE,temp  out DDRA,temp  ldi temp,0b00000001  out PORTE,temp  ldi temp,0b00000011  out PORTA,temp  ldi temp,(1<<INT2)  out GICR,temp  ldi temp,0  out MCUCR,temp  sei  loop:  nop  rjmp loop  led\_on:  sei  sbis PINA,sw0  rjmp led\_on1  sbis PINA,sw1  rjmp led\_on2  rjmp led\_on  led\_on1:  cbi PORTB,led  rcall delay1  sbi PORTB,led  wait\_0:  sbis pina,sw0  rjmp wait\_0  reti    led\_on2:  cbi PORTB,led  rcall delay2  sbi PORTB,led  wait\_1:  sbis pina,sw1  rjmp wait\_1  reti  delay1:  ; start delay  ldi r16,255  d0:  ldi r17,255  d1:  ldi r18,20  d2: dec r18  brne d2  dec r17  brne d1  dec r16  brne d0  ; end delay  ret  delay2:  rcall delay1  rcall delay1  ret |

Собранная схема в Proteus приведена на рисунке 3.

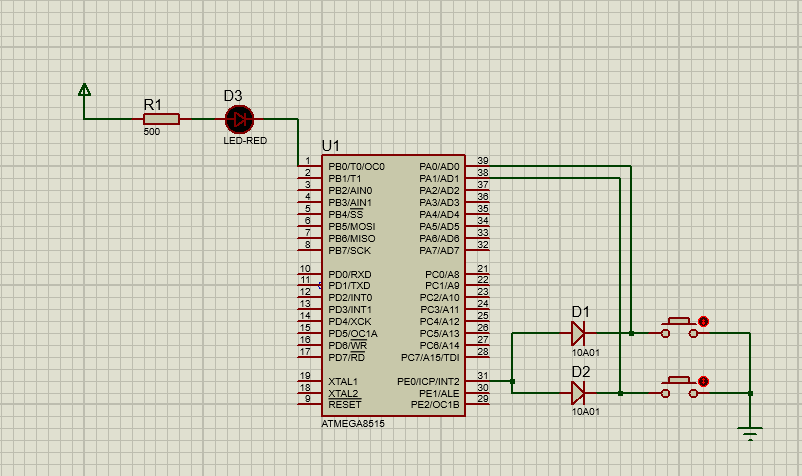


Рисунок 3 – Собранная схема в Proteus

**Выводы**

В ходе выполнения лабораторной работы были освоены навыки работы с прерываниями микроконтроллера Atmega8515. Была модифицирована программа из первой ЛР под прерывания. Получили практические навыки отладки программы (просмотр содержимого стека, дизасемблера, PC и SP), а также работы с подпрограммами и стеком.