|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ | | | | | |
| **ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ** | | | | | |
| **УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ** | | | | | |
| **«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»** | | | | | |
|  | | | | | |
| Институт информационных технологий и управления в технических системах | | | | | |
| (полное название института) | | | | | |
|  | | | | | |
| кафедра «Информационные системы» | | | | | |
| (полное название кафедры) | | | | | |
|  | | | | | |
|  | | | | | |
| **Лабораторные работы** | | | | | |
| по дисциплине “ Встроенные микропроцессорные системы” | | | | | |
| студента группы ИС/б-17-2-о  **Волобуева Юрия Сергеевича** | | | | | |
| № лр. | Выполнение | Работу принял | | | |
| Дата | Дата | Оценка | Ф.И.О. | Подпись |
| 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
| 5 |  |  |  |  |  |
| 6 |  |  |  |  |  |
| 7 |  |  |  |  |  |
| 8 |  |  |  |  |  |
| 9 |  |  |  |  |  |
| 10 |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| 20 | 20 | г. |

Лабораторная работа №3

3.1 ЦЕЛЬ

Целью работы является изучение среды моделирования электронных схем PROTEUS VSM и приобретение практических навыков составления, исследования и отладки микропроцессорных систем и их программного обеспечения.

3.2 ПОСТАНОВКА ЗАДАЧИ

1. Ознакомиться со схемами подключения переключателей (клавиш) и элементов индикации к микроконтроллеру.

2. Начертить схему микропроцессорного устройства согласно заданному варианту.

3. Составить программу опроса клавиш и управления индикацией.

4. Отладить программу в среде Atmel Studio и убедиться в правильности функционирования устройства.

5. Начертить разработанную схему микропроцессорного устройства в рабочем окне симулятора Proteus VSM и загрузить исполняемый программный модуль.

6. Запустить режим процесса симуляции и убедиться в корректной работе устройства.

7. Проверить функционирование разработанного устройства в динамическом режиме.

8. Подключить виртуальный осциллограф к элементам индикации и исследовать вид сигналов на соответствующих электродах.

Задача: подключите к микроконтроллеру типа ATmega16 один кнопочный переключатель и 8 светодиодов. При нажатии кнопки диоды должны поочередно зажигаться (бегущие огни). Светодиоды при отпущенной кнопке не должны светиться. Предусмотреть схему сброса микроконтроллера от отдельной кнопки.

3.3 ХОД РАБОТЫ

Для реализации поставленной задачи необходимо разработать программы на языке программирования Assembler. Эмуляция работы осуществляется на основе 8-разрядного микроконтроллера ATmega16. Код программы представлен ниже:

.include "m16def.inc"

.def temp = r16

.def loop1 = r17

.def loop2 = r18

.def loop3 = r19

.cseg

.org 0

ldi YH, high(RAMEND)

ldi YL, low(RAMEND)

out SPH, YH

out SPL, YL

ldi temp, 0

out DDRD, temp

ldi temp, 0b00000001

out PORTD, temp

main:

in temp, PIND

sbrc temp, 0

rjmp m1

ldi temp, 0b00000001

out DDRC, temp

sbi PORTC, 0

rcall delay

cbi PORTC, 0

rcall delay

ldi temp, 0b00000010

out DDRC, temp

sbi PORTC, 1

rcall delay

cbi PORTC, 1

rcall delay

ldi temp, 0b00000100

out DDRC, temp

sbi PORTC, 2

rcall delay

cbi PORTC, 2

rcall delay

ldi temp, 0b00001000

out DDRC, temp

sbi PORTC, 3

rcall delay

cbi PORTC, 3

rcall delay

ldi temp, 0b00010000

out DDRC, temp

sbi PORTC, 4

rcall delay

cbi PORTC, 4

rcall delay

ldi temp, 0b00100000

out DDRC, temp

sbi PORTC, 5

rcall delay

cbi PORTC, 5

rcall delay

ldi temp, 0b01000000

out DDRC, temp

sbi PORTC, 6

rcall delay

cbi PORTC, 6

rcall delay

ldi temp, 0b10000000

out DDRC, temp

sbi PORTC, 7

rcall delay

cbi PORTC, 7

rcall delay

rjmp main

m1:

ldi temp, 0b00000000

out DDRC, temp

out PORTC, temp

rjmp main

delay:

push loop1

push loop2

push loop3

ldi loop3, 5

z1:

dec loop3

breq z4

ldi loop2, 100

z2:

dec loop2

breq z1

ldi loop1, 100

z3:

dec loop1

brne z3

rjmp z2

z4:

pop loop3

pop loop2

pop loop1

ret

Далее программа разбирается пошагово:

*ldi YH, high(RAMEND)* – Инициализация вершины стека. Запись в старший байт Y-регистра старшего байта адреса памяти ОЗУ

*ldi YL, low(RAMEND)* – Запись в младший байт регистра Y младшего байта адреса памяти ОЗУ // Ram End означает $045Fвершину стека

*out SPH, YH* – Запись старшего байта адреса памяти в старший байт регистра стека

*out SPL, YL* – Запись младшего байта адреса памяти в младший байт регистра стека

Схематично процесс представлен рисунками 1 и 2.

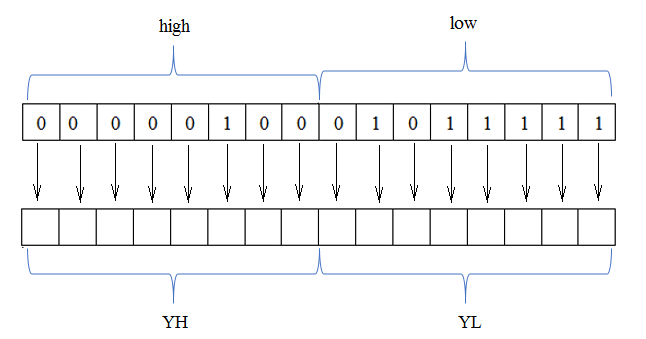


Рисунок 1 – Запись в регистр Y старшего и младшего байта адреса памяти ОЗУ

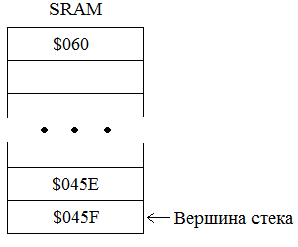


Рисунок 2 – SRAM и указатель вершины стека

*ldi temp, 0* – Запись во все разряды регистра r16 0

*out DDRD, temp* – Подключение порта D на вход

*ldi temp, 0b00000001* – Запись в младший разряд регистра r16 1

*out PORTD, temp* – Подключение младшему пину порта D подтягивающего резистора. При отпущенной кнопке будет единица установлена, при нажатой кнопке 0.

*main:*

*in temp, PIND* – Читаем содержимое порта D записываем в temp

*sbrc temp, 0* – Пропускаем следующую команду, если младший разряд порта D на низком логическом уровне (т.е. равен 0)

*rjmp m1*

Схематично процесс представлен рисунком 3.

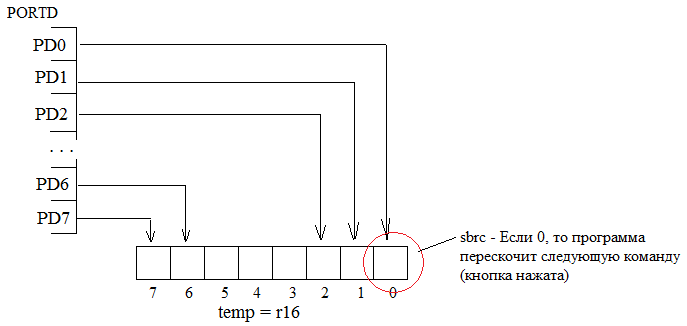


Рисунок 3 – Чтение содержимого порта D в регистр r16 и проверка на 0 младшего разряда

*ldi temp, 0b00000001* – Запись единицы в младший разряд регистра r16

*out DDRC, temp* – Подключение младшего пина порта С на выход (из РОН в РВВ)

*sbi PORTC, 0* – Установка единицы в младший пин порта C (зажигание светодиода)

*rcall delay* – Вызов подпрограммы задержки

*cbi PORTC, 0* – Установка ноля в младший пин порта С (гаснет светодиод)

*rcall delay* - Вызов подпрограммы задержки

Для каждого пина порта С аналогичный код

*m1:* - Выполняется, если отпущена кнопка (младший пин порта D = 1)

*ldi temp, 0b00000000* – Установка нолей во все разряды регистра temp

*out DDRC, temp* – Подключение всех пинов порта С на вход

*out PORTC, temp* – все пины порта С установлены в 0 (Светодиоды все не горят)

*rjmp main* – Безусловный переход на метку главной программы

Схематично процесс представлен рисунком 4.

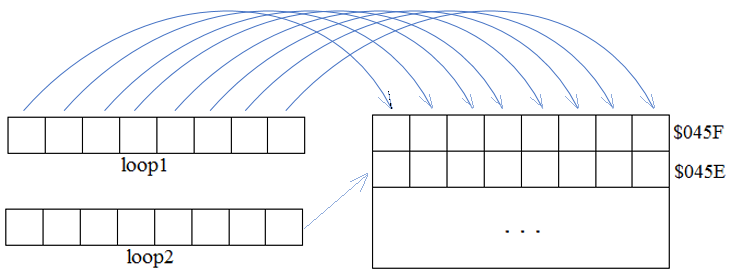


Рисунок 4 – Помещение в стеке регистров loop1, loop2, loop3 при организации задержки

Далее проводится полная отладка программы по всем направлениям. Рисунком 5 представлены начальные параметры выполнения программы.

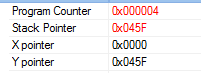


Рисунок 5 – Запись в регистр Y старшего и младшего байта адреса памяти ОЗУ и запись старшего и младшего байта адреса памяти в регистр стека

Последовательность зажигания светодиодов, подключённых к пинам порта С, регулируется установкой 1 (на выход) и 0 (на вход) последовательно в разряды порта С (рисунок 6):

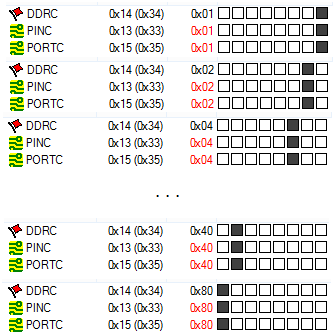


Рисунок 6 – Последовательное включение на выход и на вход пинов порта С

После тестирования программы создается симуляция в среде Proteus, схема которой представлена рисунком 7. Рисунком 8 представлена сама симуляция. Из рисунков видно, что диоды зажигаются по заданному алгоритму.

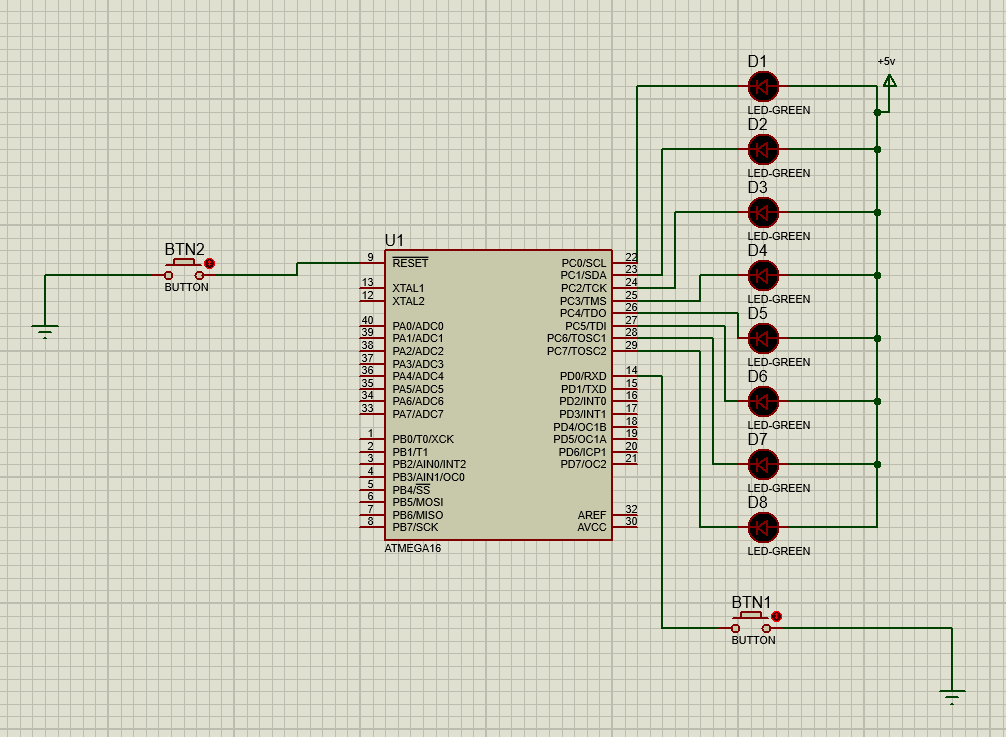


Рисунок 7 – Схема микропроцессорного устройства в Proteus

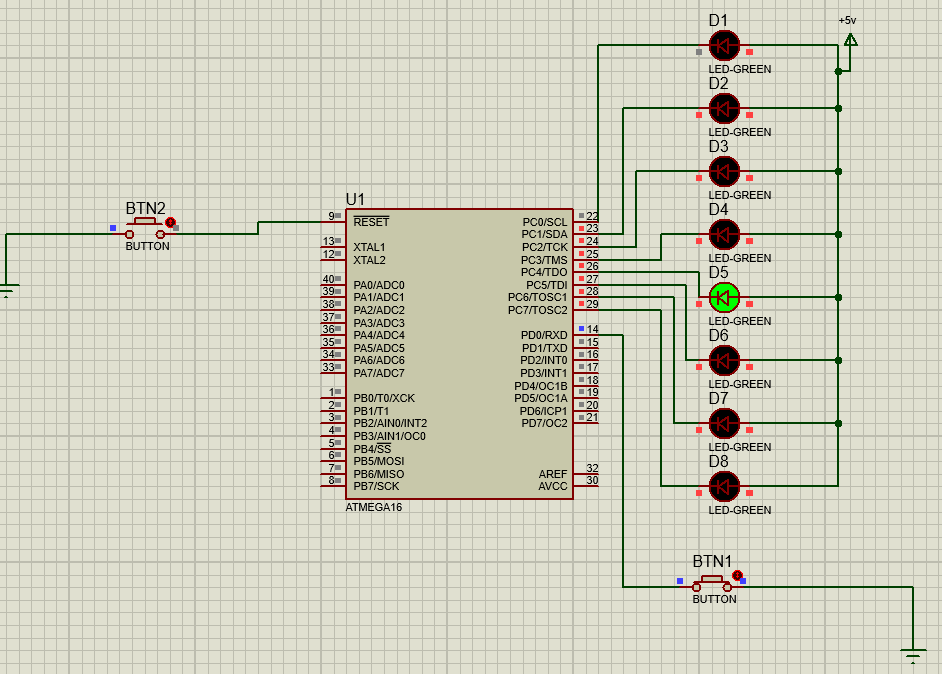


Рисунок 9 – Эмуляция в Proteus

ВЫВОДЫ

В процессе выполнения данной лабораторной работы была изучена среда моделирования электронных схем PROTEUS VSM и были приобретены практические навыки составления, исследования и отладки микропроцессорных систем и их программного обеспечения.