|  |  |
| --- | --- |
|  | **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 (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01 Информатика и вычислительная техника**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 1 |

**Название:**

Программирование портов ввода-вывода микроконтроллеров AVR.

**Дисциплина:** Микропроцессорные системы.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент | ИУ6-62Б |  |  | И.С. Марчук | |
|  | (Группа) |  | (Подпись, дата) | | (И.О. Фамилия) |
|  |  |  |  | |  |
| Преподаватель |  |  |  | |  |
|  |  |  | (Подпись, дата) | | (И.О. Фамилия) |

Москва, 2022

**Вариант 18.**

**Цели работы:**

* изучение системы команд микроконтроллеров AVR и приемов программирования на языке AVR Aссемблер;
* получение навыков отладки программ в среде отладки AVR Studio 4 и VMLab;
* работа со стартовым набором (платой) STK500.

**Ход работы.**

**Задание 1.**

Проверить работу программы в шаговом режиме работы с помощью симулятора AVR Studio 4.

Проверим работу кода, заданного в условии, и после изменим его так, чтобы между переключением светодиодов происходила задержка в 0.5 секунд. Измененный код программы представлен ниже:

*;.include "8515def.inc" ;файл определений для AT90S8515*

*.include "m8515def.inc" ;файл определений для ATmega8515*

*.def temp = r16 ;временный регистр*

*.def reg\_led = r20 ;регистр состояния светодиодов*

*.equ START = 0 ;0-ой вывод порта*

*.equ STOP = 1 ;1-ый вывод порта*

*.org $000*

*rjmp init*

*;\*\*\*Инициализация\*\*\**

*INIT:*

*ldi reg\_led,0xFE ;сброс reg\_led.0 для включения LED0*

*sec ;C=1*

*set ;T=1 – флаг направления*

*ser temp ;инициализация выводов*

*out DDRB,temp ; порта PB на вывод*

*out PORTB,temp ;погасить СД*

*clr temp ;инициализация*

*out DDRD,temp ; порта PD на ввод*

*ldi temp,0x03 ;включение ‘подтягивающих’*

*out PORTD,temp ; резисторов порта PD (0-й, 1-й разряды)*

*WAITSTART: ;ожидание*

*sbic PIND,START ; нажатия*

*rjmp WAITSTART ; кнопки START*

*LOOP:*

*out PORTB,reg\_led ;вывод на индикаторы*

*;\*\*\*Задержка (три вложенных цикла)\*\*\**

*ldi r17, 200*

*d0: ldi r18, 160*

*d1: ldi r19, 20*

*d2: dec r19*

*brne d2*

*dec r18*

*brne d1*

*dec r17*

*brne d0*

*sbic PIND,STOP ;если нажата кнопка STOP,*

*rjmp MM ; то переход*

*rjmp WAITSTART ; для проверки кнопки START*

*MM: brts LEFT ;переход, если флаг T установлен*

*sbrs reg\_led,0 ;пропуск следующей команды,*

*; если 0-й разряд reg\_led установлен*

*set ;T=1 - переключение флага направления*

*ror reg\_led ;сдвиг reg\_led вправо на 1 разряд*

*rjmp LOOP ;переход на проверку нажатия STOP*

*LEFT:*

*sbrs reg\_led,7 ;пропуск следующей команды,*

*; если 7-й разряд reg\_led установлен*

*clt ;T=0 – переключение флага направления*

*rol reg\_led ;сдвиг reg\_led влево на 1 разряд*

*rjmp LOOP*

На рисунках 1 и 2 показано время «Stop Watch» до входа в цикл и по кончании выполнения цикла. Время выполнения циклов = 5.250 мс - 504155.25 мс = ~0.5 c.



Рисунок 1 - время до входа в цикл



Рисунок 2 – время после исполнения цикла

Схема алгоритма программы показана на рисунках 3 и 4.



Рисунок 3 – схема алгоритма



Рисунок 4 – схема алгоритма

**Задание 2.**

Проверить работу программы в среде VMLab (или Proteus).

Для проверки работы программы в Proteus была создана схема, представленная на рисунке 5.

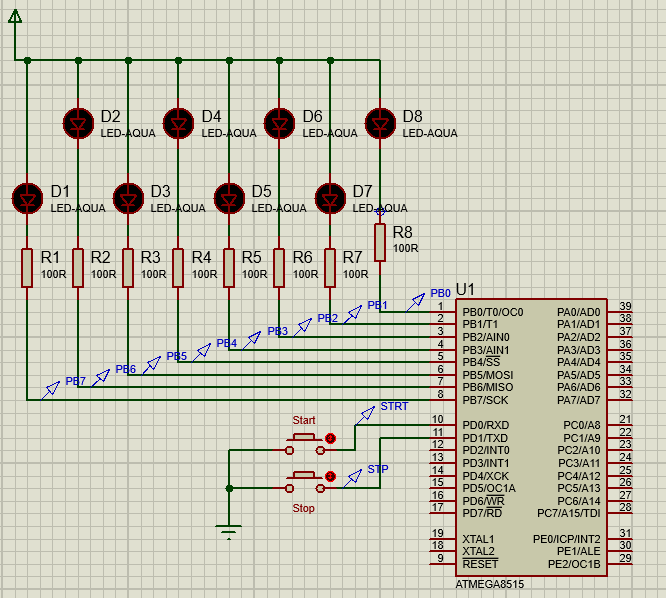


Рисунок 5 – схема в Proteus

С помощью логического анализатора была получена временная диаграмм, представленная на рисунке 6. Как видно, время задержки равно примерно 0.5 c.

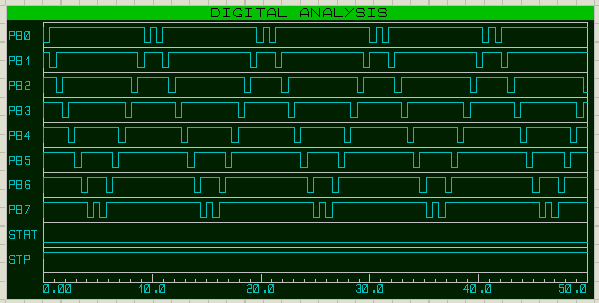


Рисунок 6 – временная диаграмма сигналов

Диаграмма демонстрирует, что программа работает корректно.

**Задание 3.**

По заданию преподавателя изменить программу для переключения светодиодов в заданной последовательности.

Задание для 18 варианта: непрерывно, перемещая один ВЫКЛ светодиод, начиная с 0 разряда влево до 7, затем один ВКЛ светодиод снова с 0 до 7 и т.д. Порт индикации PC, задержка 250мс.

Код, реализующий программу представлен ниже:

*.include "m8515def.inc";файл определений для AT90S8515*

*.def temp = r16 ;временный регистр*

*.def reg\_led = r20 ;регистр состояния светодиодов*

*.equ START = 0 ;0-ой вывод порта*

*.equ STOP = 1 ;1-ый вывод порта*

*.org $000*

*rjmp INIT*

*;\*\*\*Инициализация\*\*\**

*INIT: ldi reg\_led, 0xFE ;сброс reg\_led.0 для включения LED0*

*sec ;C=1*

*set ;T=1 – флаг направления*

*ser temp ;инициализация выводов (ser R1 - единицы в разряды регистра)*

*out DDRC, temp ; порт B весь на вывод*

*out PORTC, temp ; вывести единицы с порта B, что погасит светодиоды*

*clr temp ; вывод нулей в регистр*

*out DDRD,temp ; порта PD на ввод*

*ldi temp,0x03 ;включение ‘подтягивающих’*

*out PORTD,temp ; резисторов порта PD (0-й, 1-й разряды)*

*rjmp WAITSTART*

*WAITSTART: ;ожидание*

*sbic PIND, START; нажатия (sbic - смотрим бит номер START в порту PIND, если 0, пропускаем)*

*rjmp WAITSTART ; кнопки START*

*rjmp LOOP*

*LOOP:*

*brts NOINV; перейти, если флаг T установлен*

*mov r21, reg\_led*

*com r21 ; инверсия*

*out PORTC, r21 ;вывод на индикаторы*

*rjmp LOOP\_DELAY*

*NOINV:out PORTC, reg\_led;вывод на индикаторы*

*LOOP\_DELAY:;\*\*\*Задержка (три вложенных цикла)\*\*\**

*ldi r17, 215 ;(LDI Rd, K Загрузка константы в ргистр)*

*d0: ldi r18, 144*

*d1: ldi r19, 10*

*d2: dec r19*

*brne d2*

*dec r18*

*brne d1*

*dec r17 ; (Декремент регистра ; Признаки: Z,N,V)*

*brne d0 ; (Перейти, если не равно)*

*sbic PIND, STOP ;если нажата кнопка STOP, (sbic - смотрим бит номер STOP в порту PIND, если 0, пропускаем)*

*rjmp CONTINUE ; если не нажата идем дальше по циклу*

*rjmp WAITSTART ; если нажата, идем к проверке кнопки START*

*CONTINUE:*

*SBRC reg\_led, 7 ; пропуск следующей команды, если 7-й разряд в reg\_led = 0*

*rjmp SHIFT ; пропуск инверсии флага и очистки регистра*

*; инверсия флага T*

*brts INV\_1 ; переход, если флаг T установлен*

*set ;T=1*

*rjmp INV\_END*

*INV\_1: clt ; T=0*

*INV\_END:*

*;сброс reg\_led.0 для включения LED0*

*ldi reg\_led, 0xFE*

*rjmp LOOP*

*SHIFT: rol reg\_led ;сдвиг reg\_led влево на 1 разряд*

*rjmp LOOP*

На рисунках 7 и 8 показано время «Stop Watch» до входа в цикл и по кончании выполнения цикла. Время выполнения циклов = 5.250 - 255586.5 = ~0.25 c, что соответствует индивидуальному заданию.



Рисунок 7 - время до входа в цикл



Рисунок 8 – время после исполнения цикла

**Задание 4.**

Проверить работу подготовленной программы в VMLab (или Proteus). Запротоколировать работу программы в виде «скриншота» осциллограммы.

С помощью логического анализатора была получена временная диаграмм, представленная на рисунке 9.

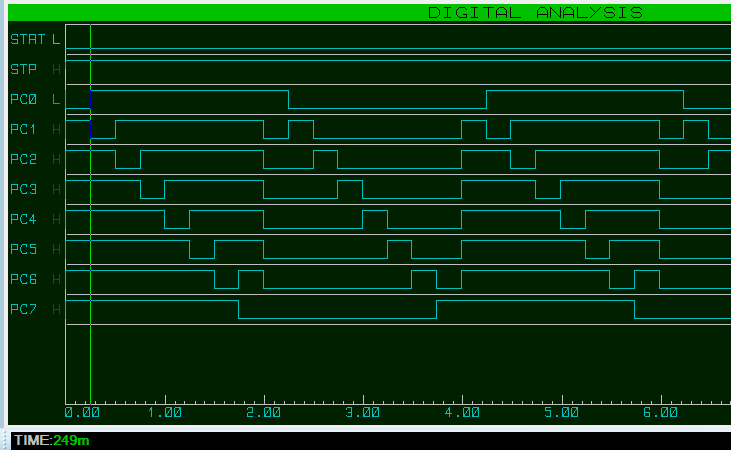


Рисунок 9 – временная диаграмма сигналов

Диаграмма демонстрирует, что программа работает корректно.



**Вывод:** в ходе выполнения лабораторной работы была изучена система команд микроконтроллеров AVR, получены навыки отладки программ в среде AVR Studio 4 с помощью точек остановки и просмотра состояния регистров и отладки в среде Proteus с помощью моделирования схемы и использования логического анализатора. Также был изучен способ формирования задержки с помощью вложенных циклов в программах для микроконтроллеров AVR.