|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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: «Специальное машиностроение»

Кафедра: СМ11 «Подводные аппараты и роботы»

**Лабораторная работа № 2**

по курсу «Электронные устройства мехатронных и робототехнических систем»

Вариант 8

Выполнил: Ионин Даниил

Группа: СМ11-61Б

Проверил(а):

Москва, 2024 г.

ОГЛАВЛЕНИЕ

[ОГЛАВЛЕНИЕ 2](#_Toc165834569)

[Введение 3](#_Toc165834570)

[Тема работы: 3](#_Toc165834571)

[Цель работы: 3](#_Toc165834572)

[Теоретическая часть 4](#_Toc165834573)

[Память 4](#_Toc165834574)

[Порты ввода/вывода 4](#_Toc165834575)

[Условные обозначения 6](#_Toc165834576)

[Блок схема ATMEGA16 7](#_Toc165834577)

[Работа со стендом EasyAVR5 8](#_Toc165834578)

[Практическая часть 10](#_Toc165834579)

[Постановка задачи: 10](#_Toc165834580)

Введение

Тема работы:

Рассмотрение принципов работы микроконтроллеров при помощи стенда EAZYAVR 5

Цель работы:

Изучить функциональные возможности учебно-отладочного стенда, внутреннюю структуру и систему команд МК ATmega16. Получить первичные навыки программирования МК ATmega16.

Теоретическая часть

Память

Память микроконтроллеров типа AVR распределяется на два ключевых сегмента: область хранения информации и пространство для программного кода. Данная модель предполагает наличие дополнительной памяти EEPROM, которая обеспечивает независимость от источника питания при сохранении данных.

Память микроконтроллера Atmega16 типа AVR представляет собой область, предназначенную для временного хранения переменных и данных, необходимых для выполнения операций, предусмотренных программой. Эта память является энергозависимой, что означает, что её содержимое теряется при отключении питания устройства. Она обладает высокой скоростью доступа, и организована как 8 кбайт х 16.

Порты ввода/вывода

В составе данного МК имеет четыре 8-разрядных двунаправленных порта.

Для обслуживания каждого порта отведено три регистра: регистр данных PORTx, регистр направления данных - DDRx и регистр выводов порта PINx (вместо х подставляется соответственно буквы A, B, C, D). Регистр выводов порта предназначен только для чтения, в то время как регистр данных и регистр направления данных - для чтения/записи.

Все выводы портов имеют отдельно подключаемые подтягивающие резисторы. Выходы порта могут поглощать ток до 20 мА и непосредственно управлять светодиодными индикаторами. Выводы используются как входы и замыкаются на землю, если включены внутренние подтягивающие резисторы, при этом выводы являются источниками тока (IIL).

Все 8 бит каждого из портов при использовании для ввода/вывода одинаковы.

Бит DDxn регистра DDRx выбирает направление передачи данных. Если бит установлен (1), вывод сконфигурирован как выход. Если бит сброшен (0) - вывод сконфигурирован как вход. Если PORTxn установлен и вывод сконфигурирован как вход, включается КМОП подтягивающий резистор. Для отключения резистора, PORTxn должен быть сброшен (0) или вывод должен быть сконфигурирован как выход (см. таблицу 1).

Таблица 1 – Влияние DDxn на выводы порта

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DDxn | PORTxn | Вх/Вых | Подт.резист | Комментарий |
| 0 | 0 | Вход | Нет | Третье состояние (Hi-Z) |
| 0 | 1 | Вход | Да | Pxn источник тока IIL, если извне соединен с землей |
| 1 | 0 | Выход | Нет | Выход установлен в 0 |
| 1 | 1 | Выход | Нет | Выход установлен в 1 |

n = 7,6...0 - номер вывода.

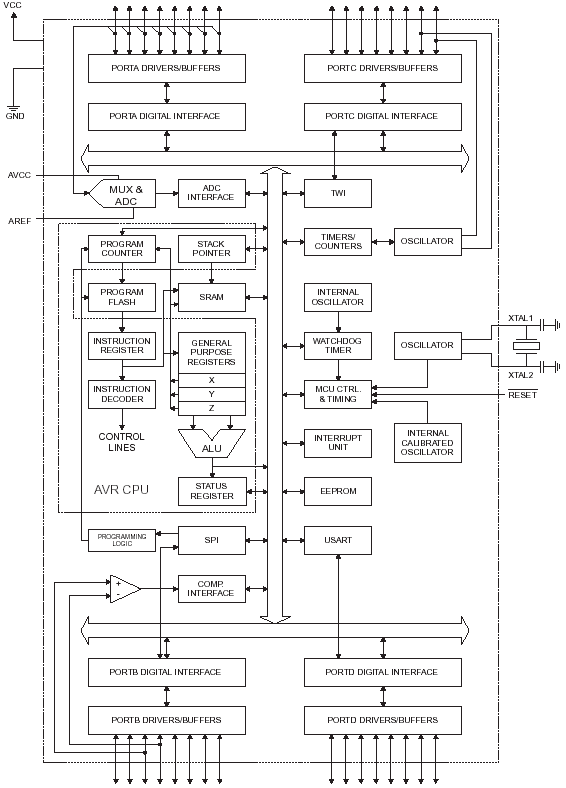
Стенд Easyavr5 представляет собой лабораторный макет с микроконтроллером. К каждому выводу микроконтроллера подключена кнопка и светодиод. На макете установлено четыре семисегментных индикатора. Также имеется возможность подключения к макету внешних устройств через специальные устройства. Питание макета осуществляется либо от внешнего источника напряжения 8–16В, либо через USB-порт. Кроме этого, на макете установлен внутрисхемный программатор.

В лабораторных работах будет использоваться микроконтроллер фирмы ATMEL – ATmega16.

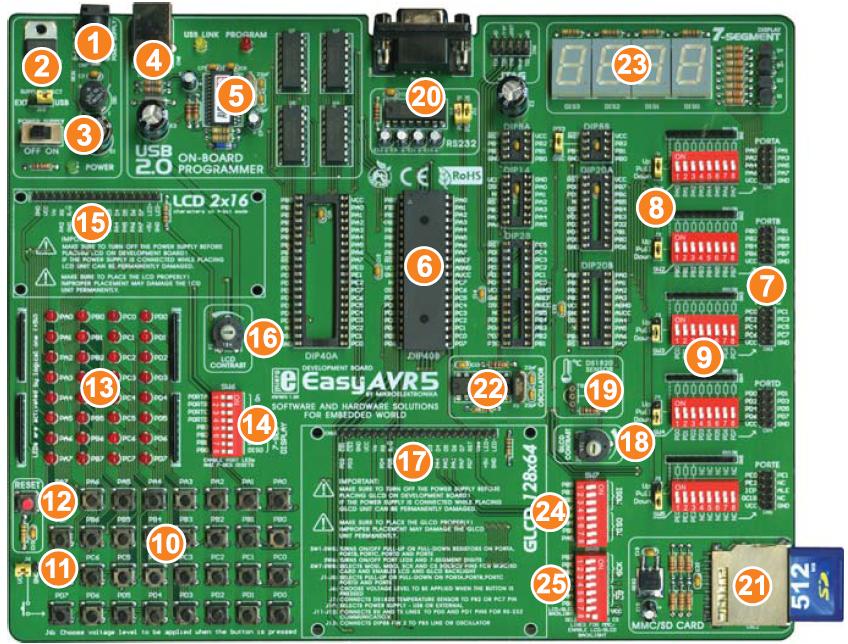
Условные обозначения

* Flash ROM - объем энергонезависимой памяти программ (в килобайтах);
* EEPROM - объем энергонезависимой памяти данных (в байтах);
* RAM - объем статической памяти данных (в байтах);
* External RAM - возможность подключения к микроконтроллеру дополнительной микросхемы внешней статической памяти данных (в килобайтах);
* ISP - возможность программирования микроконтроллера в системе (на целевой плате) при основном напряжении питания;
* SPM - функция самопрограммирования Flash ROM памяти микроконтроллера в системе без участия внешнего программатора;
* JTAG - встроенный JTAG - интерфейс;
* I/O (pins) - максимальное количество доступных линий ввода / вывода;
* Timer(s) 8/16 bit - количество и разрядность таймеров/счетчиков;
* USI - универсальный коммуникационный интерфейс;
* AC - аналоговый компаратор;
* ADC (channels) - количество каналов аналого-цифрового преобразования;
* Internal RC - наличие внутренней RC-цепочки для автономной работы микроконтроллера (без внешнего источника опорной частоты);
* WDT - сторожевой таймер;
* BDC - аппаратный программируемый блок защиты от сбоев при внезапном (в том числе и кратковременном) пропадании напряжения питания микроконтроллера;
* UART - асинхронный последовательный приемопередатчик;
* SPI - синхронный трехпроводной последовательный интерфейс;
* I2C - двухпроводной последовательный интерфейс;
* RTC - система реального времени;
* PWM (channels) - количество независимых каналов широтно - импульсной модуляции;
* Command Set - количество различных инструкций в системе команд микроконтроллера;
* Vcc - диапазон рабочих напряжений питания (в Вольтах);
* Clock - диапазон рабочих частот (в мегагерцах);

Блок схема ATMEGA16

 Рисунок 1 – Блок схема ATmega16

Работа со стендом EasyAVR5

 Рисунок 2 – Внешний вид стенда EasyAVR5

1 – разъем для подключения внешнего питания 8–16В;

2 – выбор между внешним питанием и питанием от USB;

3 – выключатель;

4 – разъем USB;

5 – встроенный внутрисхемный программатор;

6 – панельки для установки различных типов МК;

7 – разъемы для доступа к портам МК;

8 – джамперы для выбора типа подтягивающих резисторов (к 1 или к 0);

9 – группы переключателей для подключения подтягивающих резисторов к выводам МК;

10 – 32 кнопки, подключенные к выводам МК;

11 – джампер выбора высокого или низкого состояния при нажатии кнопки:

12 – кнопка сброса;

13 – светодиоды, каждый подключенный к выводу МК;

14 – группа переключателей, подключающая или отключающая светодиоды к МК;

15 – разъем для подключения символьной ЖК-матрицы;

16 – потенциометр для регулировки яркости символьной ЖК-матрицы;

17 – разъем для подключения графической ЖК-матрицы;

18 – потенциометр для регулировки яркости графической ЖК-матрицы;

19 – разъем для подключения датчика температуры DS1820;

20 – разъем порта RS232;

21 – слот для подключения MMC/SD карт памяти;

22 – встроенный генератор частоты;

23 – 4 семисегментных индикатора;

24 – группа переключателей для разрешения внутрисхемного программирования;

25 – группа переключателей для разрешения работы с MMC/SD картами памяти.

Практическая часть

Постановка задачи:

1. Выполните проверку работы программы в симуляторе AVR Studio. Измерьте в симуляторе время, в течение которого поочередно включен каждый семисегментный индикатор;
2. Выполните: программирование макетной платы. Проверьте работу платы в 3-х режимах:

- нажать кнопку PD1-«готовность» после загорания CИД(PB7) нажать PD2 "реакция"

- нажать кнопку PD1-«готовность», до загорания CИД(PB7) нажать PD2 "реакция"

- нажать кнопку PD1-«готовность» после загорания CИД(PB7) не нажимать PD2 "реакция".

1. Найдите и исправьте ошибку в программе;
2. Доработайте программу, увеличив счетчик дисплея Disp\_Count до 2-х байт с максимальным значением 0xFFFF, введя дополнительный байт Disp\_CountH. Запрограммируйте макетную плату;
3. Объясните работу схемы. Экспериментально подберите константу Val\_dispCount, определяющую время включения индикатора, когда глаз не воспринимает мигание индикаторов. Измерьте в симуляторе время, в течение которого поочередно включен каждый семисегментный индикатор.

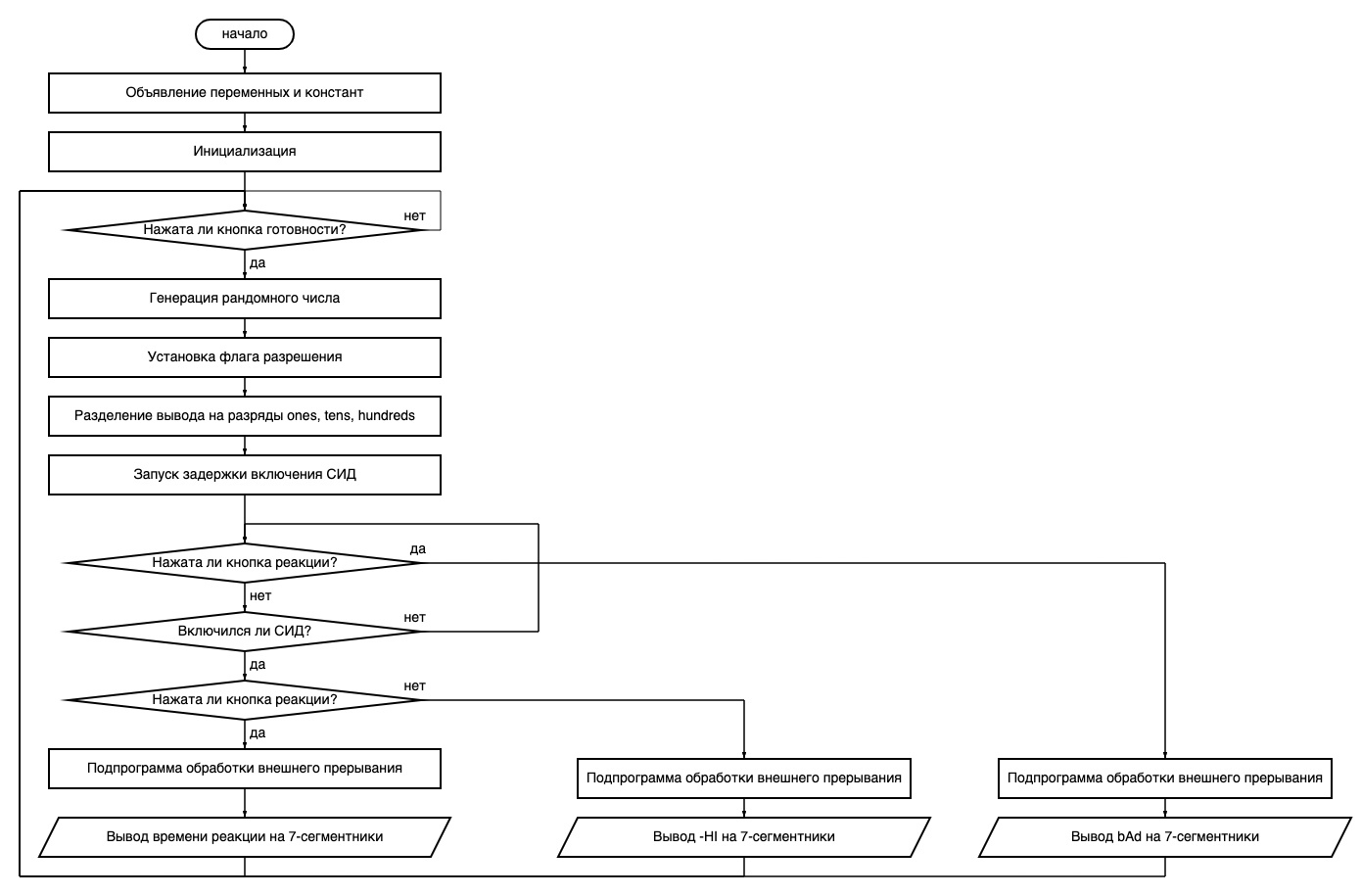


Рисунок 2 – блок схема программного кода.

Схема электрическая принципиальная и перечень элементов в приложении 1.