Министерство образования и науки Украины
Национальный технический университет
«Харьковский политехнический институт»

# Методические указания

# к лабораторной работе №2

**«Изучение программирования микроконтроллера»**

**по курсу «*Интерфейсы компьютерных систем управления*»**

Харьков 2006

Министерство образования и науки Украины

# МЕТОДИЧЕСКИЕ УКАЗАНИЯ

# К ЛАБОРАТОРНОЙ РАБОТЕ № 2

# «ИЗУЧЕНИЕ ПРОГРАММИРОВАНИЯ МИКРОКОНТРОЛЛЕРА»

# ПО КУРСУ «ИНТЕРФЕЙСЫ КОМПЬЮТЕРНЫХ СИСТЕМ УПРАВЛЕНИЯ»

для студентов специальности 7.0910002

Утверждено
редакционно-издательским
советом университета,
протокол № 2 от 02.06.04

Харьков НТУ «ХПИ» 2006

Методические указания к лабораторной работе «Изучение программирования микроконтроллера» по курсу «Интерфейсы компьютерных систем управления» для студентов специальности 7.0910002 / Сост. Д.А. Лунин, М.В. Гунбин. – Харьков: НТУ «ХПИ» 2006. – 13 с.

Составители: Д.А. Лунин, М.В. Гунбин

Рецензент А.В. Ивашко

Кафедра автоматики и управления в технических системах

***Цель работы***: закрепление и углубление лекционного материала по структуре и принципам функционирования микроконтроллера ATmega16; изучение структуры и программного обеспечения программатора, приобретение практических навыков в программировании микроконтроллера ATmega16

## 1 ОПИСАНИЕ ЛАБОРАТОРНОЙ УСТАНОВКИ

Рабочее место в лаборатории состоит из персонального компьютера и специализированного лабораторного стенда.

Программно-отладочный стенд «AVR – микролаб» (далее «стенд») предназначен для моделирования и отладки микропроцессорных устройств на базе микроконтроллеров фирмы Atmel и может быть использованным как для промышленных разработок, так и в учебных целях.

В конструкции стенда реализован открытый модульный принцип, в частности, стенд представляет собой коллекцию наиболее часто используемых типов модулей периферийных устройств системы, коммутируемых в моделирующую структуру с помощью прилагаемых плоских однотипных 10-жильных кабельных перемычек. Каждый модуль имеет от 1 до 4 однотипных разъемов, на которые выведены все функциональные связи модуля.

Такая конструкция позволяет конфигурировать информационно-измерительную или управляющую структуру как из имеющихся в стенде модулей, так и путем включения в нее внешних модулей или реальных объектов - датчиков, исполнительных устройств, индикаторов. Более того, данный принцип позволяет моделировать и более сложные структуры – микропроцессорные системы различной конфигурации путем объединения нескольких стендов.

В состав функциональных устройств стенда входят процессорный модуль, 10 модулей периферийных устройств, интерфейсные кабели для подключения к COM и LPT портам компьютера, и 5 плоских кабельных перемычек для коммутации модулей в моделирующую структуру.

В состав стенда входят следующие модули периферийных устройств:

1. модуль микроконтроллера МС;
2. модуль статической светодиодной индикации;
3. модуль динамической светодиодной индикации;
4. модуль буквенно-цифрового дисплея LCD;
5. модуль функциональной клавиатуры;
6. модуль аналого-цифрового преобразователя;
7. модуль цифро-аналогового преобразователя;
8. модуль драйвера последовательного интерфейса RS 232;
9. модуль последовательного периферийного интерфейса SPI;
10. модуль драйверов управления шаговыми двигателями;
11. модуль блока выходных ключей.

Лабораторный стенд питается от внешнего маломощного нестабилизированного источника питания постоянного тока напряжением 9…12В ($I_{пит}$ < 400 mA).

Для питания внешних нагрузок ($I_{нагр}$ > 300 mA) необходимо использовать внешний источник питания соответствующей мощности.

Подключение периферийного модуля к одному из четырех портов модуля микроконтроллера осуществляется при помощи плоских десятиконтактных перемычек (8 линий порта и 2 провода питания).

Все неиспользуемые периферийные модули стенда обесточены, а используемые питаются от модуля микроконтроллера через перемычки (контакт 9 – «+5В», контакт 10 – «-5В»).

Использование для модулей гнезд разъемов «с ключом» исключает «неправильную» коммутацию перемычек.

Схемотехника модулей предусматривает цепи защиты микроконтроллера от «переполюсовки» и превышения уровней сигналов.

Каждый периферийный модуль может быть подключен к любому порту микроконтроллера за исключением тех модулей, которые функционально связаны с определенными выводами микроконтроллера (RS232 - PORTD, ADC – PORTA, SPI - PORTB). Для таких модулей, на их принципиальных схемах, в обозначении разъема указывается конкретный порт микроконтроллера.

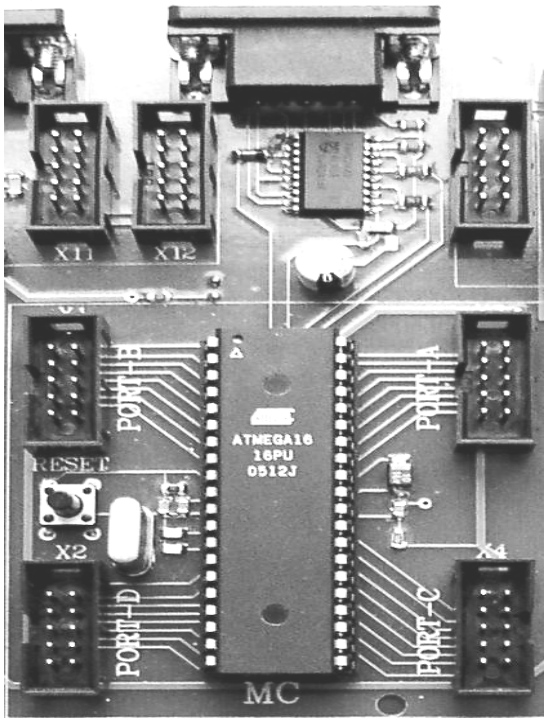


Рис. 1 – Внешний вид модуля микроконтроллера

Модуль микроконтроллера является основой лабораторного стенда. Он состоит из: 40-контактной разъемной панели, в которую может быть установлен любой AVR микроконтроллер из ряда: ATmega16, ATmega8535, ATmega163, ATmega323 ATmega32 без каких-либо коммутаций и установок, встроенного стабилизатора напряжения питания, элементов программатора. Для подключения к четырем портам ввода-вывода микроконтроллера периферийных устройств модуль содержит 4 унифицированных разъема. Кроме того, в состав модуля входят цепи синхронизации и сброса. Принципиальная схема модуля приведена на Рис.2.

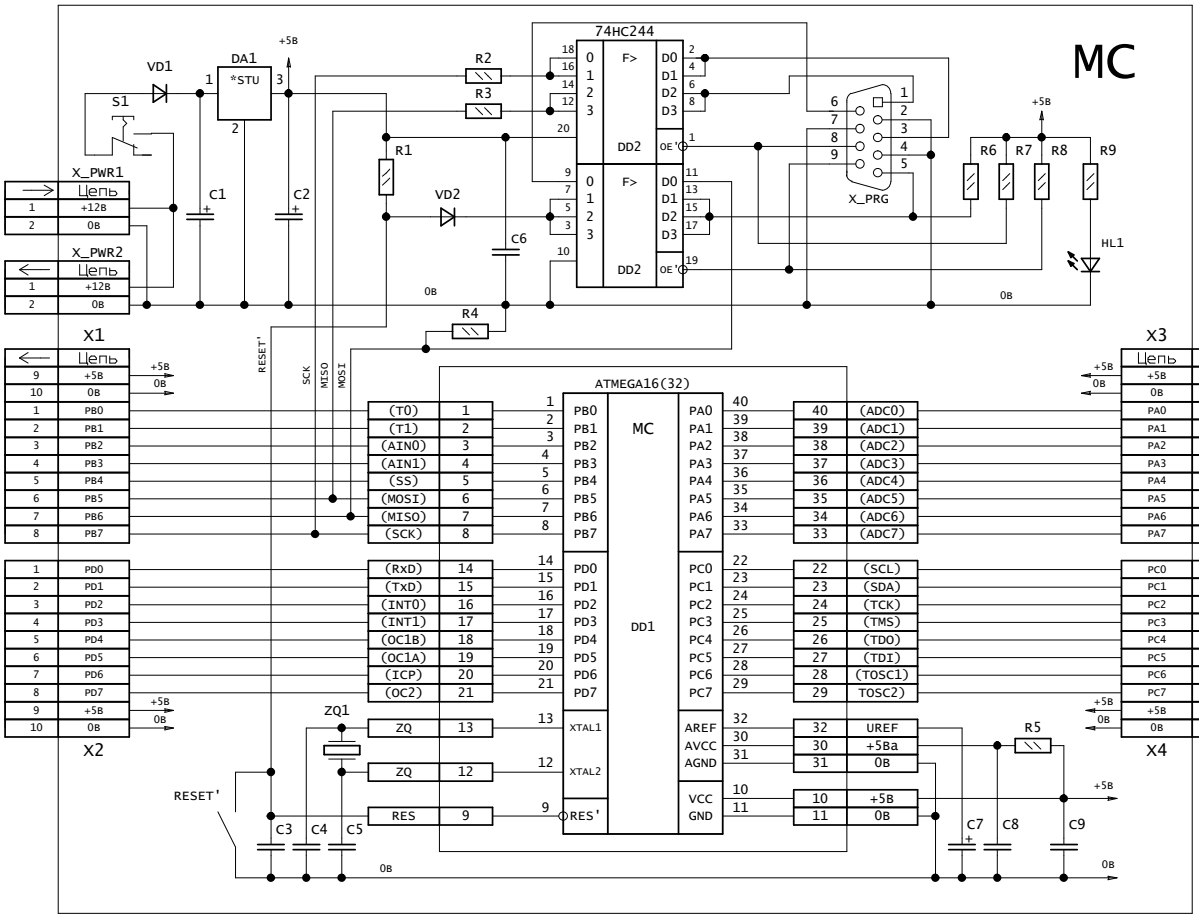


Рис.2 Схема электрическая принципиальная модуля микроконтроллера.

Основой модуля является микроконтроллер DD1 с цепями сброса R1,C3 и синхронизации ZQ1, C4, C5. Кварцевый резонатор ZQ1 определяет рабочую частоту микроконтроллера – 4(8)МГц. Кнопка «RESET» позволяет выполнить перезапуск программы микроконтроллера в случае необходимости. Аналоговая часть микроконтроллера запитана от источника питания +5В через фильтрующую RC цепь R5,C8. Конденсатор C7 предназначен для фильтрации напряжения $U_{ref}$ = 2,56В на выходе встроенного в микроконтроллер источника опорного напряжения. На контакты гнезд X1…X4 выведены линии четырех портов микроконтроллера. Причем, младшему разряду порта микроконтроллера соответствует младший номер контакта разъема. К старшим контактам разъемов портов подведено напряжение питания +5В. За основу встроенного в схему модуля микроконтроллера программатора взята рекомендованная фирмой ATMEL схема внутрисхемного программатора (ISP) STK200/300 построенного на микросхеме шинного формирователя DD2 и резисторах R2…R4, R6…R8. Резисторы R4, R6…R8 подтягивают «висящие» входы шинного формирователя DD2 к уровню напряжения питания, а резисторы R2, R3 предохраняют выходы микроконтроллера DD1 и шинного формирователя DD2 при одновременной работе программатора и SPI интерфейса. Диод VD2 препятствует закорачиванию выходов шинного формирователя DD2 на «землю» при нажатии кнопки «RESET». Питается модуль микроконтроллера

и все подключаемые к нему модули от встроенного стабилизатора напряжения, выполненного на микросхеме DA1 и конденсаторах фильтра C1, C2. Диод VD1 предохраняет схему от случайной переполюсовки напряжения питания на контактах разъемов питания X_PWR1 или X_PWR2.

## 2 АРХИТЕКТУРА И ПРОСТЕЙШИЕ ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ МИКРОКОНТРОЛЛЕРА ATmega16

ATmega16 - 8-разрядные микроконтроллеры изготавливаются по малопотребляющей КМОП-технологии, которая в сочетании с усовершенствованной RISC-архитектурой позволяет достичь наилучшего соотношения стоимость/быстродействие/энергопотребление. Они имеют FLASH-память программ объемом 16 Кбайт, ОЗУ объемом 1 Кбайт и EEPROM-память данных объемом 512 байт. Максимальное число контактов ввода/вывода равно 32. Рабочие напряжения 4,5 - 5,5 В, рабочая частота 0 - 16 МГц, диапазон рабочих температур 0…+70 ºС (коммерческое исполнение).

Таблица 1 – Описание выводов модели ATmega16

| Обозначение | Номер вывода | | Тип вывода | Описание |
|---|---|---|---|---|
| | DIP | TQFP | | |
| XTAL1 | 13 | 8 | I | Вход тактового генератора |
| XTAL2 | 12 | 7 | O | Выход тактового генератора |
| RESET | 9 | 4 | I | Вход сброса. При удержании на входе НИЗКОГО уровня в течении 50 нс выполняется сброс устройства |
| Порт A. 8-разрядный двунаправленный порт ввода/вывода с внутренним подтягивающими резисторами | | | | |
| PA0 (ADC0) | 40 | 37 | I/O | A0 (Вход АЦП) |
| PA1 (ADC1) | 39 | 36 | I/O | A1 (Вход АЦП) |
| PA2 (ADC2) | 38 | 35 | I/O | A2 (Вход АЦП) |
| PA3 (ADC3) | 37 | 34 | I/O | A3 (Вход АЦП) |
| PA4 (ADC4) | 36 | 33 | I/O | A4 (Вход АЦП) |
| PA5 (ADC5) | 35 | 32 | I/O | A5 (Вход АЦП) |
| PA0 (ADC6) | 34 | 31 | I/O | A6 (Вход АЦП) |
| PA7 (ADC7) | 33 | 30 | I/O | A7 (Вход АЦП) |
| Порт A. 8-разрядный двунаправленный порт ввода/вывода с внутренним подтягивающими резисторами | | | | |
| PB0 (T0/XCK) | 1 | 40 | I/O | B0 (Вход внешнего тактового сигнала таймера/счетчика T0/вход/выход внешнего тактового сигнала USART) |
| PB1 (T1) | 2 | 41 | I/O | B1 (Вход внешнего тактового сигнала таймера/счетчика T1) |
| PB2 (AIN0/INT2) | 3 | 42 | I/O | B2 (Положительный вход компаратора/вход внешнего |

| | | | | |
|---|---|---|---|---|
| | | | | прерывания) |
| PB3 (AIN1/OC0) | 4 | 43 | I/O | B3 (Отрицательный вход компаратора/выход таймера/счетчика T0 (режимы Compare, PWM)) |
| PB4 (SS) | 5 | 44 | I/O | B4 (Выбор Slave-устройства на шине SPI) |
| PB5 (MOSI) | 6 | 45 | I/O | B5 (Выход (Master) или вход (Slave) данных модуля SPI) |
| PB6 (MISO) | 7 | 46 | I/O | B6 (Вход (Master) или выход (Slave) данных модуля SPI) |
| PB7 (SCK) | 8 | 47 | I/O | B7 (Выход (Master) или вход (Slave) тактового сигнала модуля SPI) |
| Порт A. 8-разрядный двунаправленный порт ввода/вывода с внутренним подтягивающими резисторами | | | | |
| PC0 (SCL) | 22 | 19 | I/O | C0 (Тактовый сигнал модуля TWI) |
| PC1 (SDA) | 23 | 20 | I/O | C1 (Линия данных модуля TWI) |
| PC2 (TCK) | 24 | 21 | I/O | C2 (Тактовый сигнал JTAG) |
| PC3 (TMS) | 25 | 22 | I/O | C3 (Выбор режима JTAG) |
| PC4 (TDO) | 26 | 23 | I/O | C4 (Выход данных JTAG) |
| PC5 (TDI) | 27 | 24 | I/O | C5 (Вход данных JTAG) |
| PC6 (TOSC1) | 28 | 25 | I/O | C6 (Выход для подключения резонатора к таймеру/счетчику T2) |
| PC7 (TOSC2) | 29 | 26 | I/O | C7 (Вход для подключения резонатора к таймеру/счетчику T2) |
| Порт A. 8-разрядный двунаправленный порт ввода/вывода с внутренним подтягивающими резисторами | | | | |
| PD0 (RXD) | 14 | 9 | I/O | D0 (Вход USART) |
| PD1 (TXD) | 15 | 10 | I/O | D1 (Выход USART) |
| PD2 (INT0) | 16 | 11 | I/O | D2 (Вход внешнего прерывания) |
| PD3 (INT1) | 17 | 12 | I/O | D3 (Вход внешнего прерывания) |
| PD4 (OC1B) | 18 | 13 | I/O | D4 (Выход B таймера/счетчика T1(режимы Compare, PWM)) |
| PD5 (OC1A) | 19 | 14 | I/O | D5 (Выход A таймера/счетчика T1(режимы Compare, PWM)) |
| PD6 (ICP) | 20 | 15 | I/O | D6 (Вход захвата таймера/счетчика T1 (режим Capture)) |
| PD7 (OC2) | 21 | 16 | I/O | D7 (Выход таймера/счетчика T2(режимы Compare, PWM)) |
| AREF | 32 | 29 | P | Вход опорного напряжения для АЦП |
| AGND | 31 | 28 | P | Общий вывод (аналоговый) |
| $AV_{CC}$ | 30 | 27 | P | Вывод источника питания АЦП |
| GND | 11 | 6, 18, 39 | P | Общий вывод |

| | | | | |
|---|---|---|---|---|
| $V_{CC}$ | 10 | 5, 17, 38 | P | Вывод источника питания |

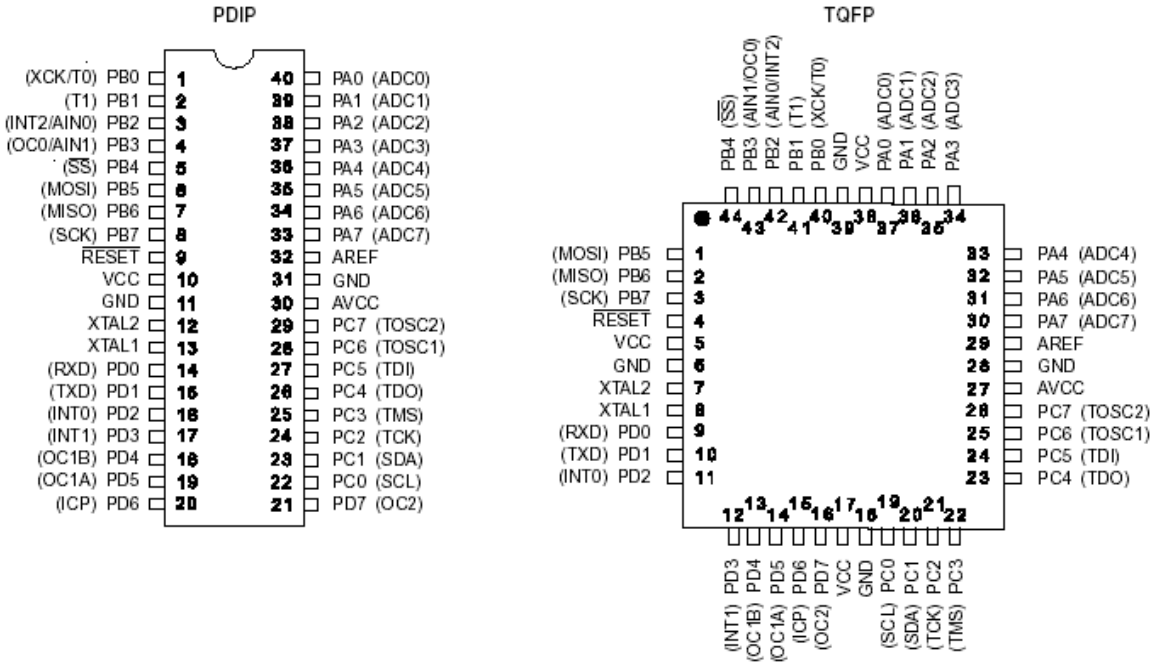


Рис. 3 - Расположение выводов:

## 3 ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРА ATmega16

Программирование FLASH памяти микроконтроллера стенда производится через LPT порт персонального компьютера при помощи специального кабеля, схема которого приведена на Рисунке 12.

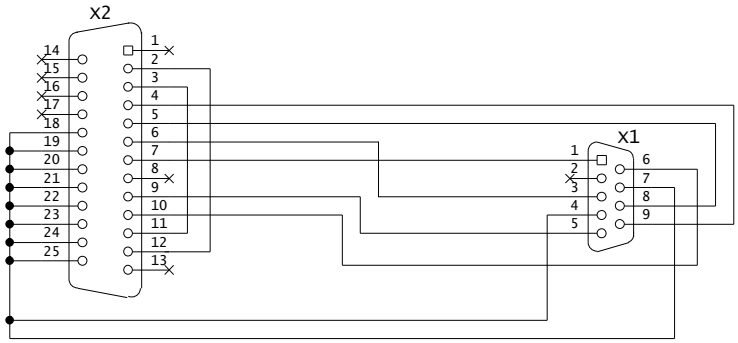


Рис. 4 Схема кабеля программатора.

Стенд работает в комплексе с персональным компьютером, на котором должна быть установлена операционная система WINDOWS, начиная с WIN98, с минимальными требованиями к производительности и быстродействию.

Программной средой для работы со стендом может быть ассемблер AvrAsm_32, интегрированный в среду разработки AVR Studio_4 и программатор AVR ISP 3.35, бесплатно распространяемые фирмой ATMEL.

Кроме того, существует не менее шести демонстрационных версий компиляторов языков высокого уровня (C++, Basic, Pascal) с ограниченным размером компилируемого кода, недостаточным для промышленных применений, но пригодных для учебных задач лабораторного практикума.

В общей сложности микроконтроллеры семейства Mega поддерживают следующие режимы программирования:

- режим последовательного программирования (по интерфейсу SPI);
- режим параллельного программирования при высоком наряжении;
- режим программирования через интерфейс JTAG.

Кроме того, микроконтроллеры семейства Mega имеют возможность самопрограммирования. Под этим термином понимается изменения содержимого памяти программ, управляемое самим микроконтроллером.

В процессе программирования могут выполняться следующие операции:

- стирание кристалла;
- чтение/запись FLASH-памяти программ;
- чтение/запись EEPROM-памяти данных;
- чтение/запись конфигурационных ячеек;
- чтение/запись ячеек защиты;
- чтение ячеек идентификатора;
- чтение калибровочного байта.

Содержимое FLASH-памяти программ, а также содержимое EEPROM-памяти данных может быть защищено от записи и/или чтения посредством программирования ячеек (Lock Bits) LB1 и LB2. Возможные режимы защиты, соответствующие различным состояниям этих ячеек, приведены в Табл.2.

Таблица 2. Режимы защиты

| № режима | Ячейки защиты | | Описание |
|---|---|---|---|
| | LB1 | LB2 | |
| 1 | 1 | 1 | Защита кода данных отключена |
| 2 | 0 | 1 | Последующая запись FLASH и EEPROM запрещена |
| 3 | 0 | 0 | Запрещены запись и чтение FLASH и EEPROM |

В режимах 2 и 3 запрещается также изменение конфигурационных ячеек. Поэтому включение защиты следует выполнять в самую последнюю очередь, после программирования остальных областей памяти микроконтроллера.

Микроконтроллеры семейства Mega имеют четыре дополнительные ячейки защиты BLB02, BLB01, BLB12 и BLB11. Возможные режимы

защиты, соответствующие различным состояниям этих ячеек, приведены в табл.3 и 4 соответственно.

Таблица 3. Режимы защиты секции прикладной пограммы

| № режима | Ячейки защиты | | Описание |
|---|---|---|---|
| | BLB02 | BLB01 | |
| 1 | 1 | 1 | Нет никаких ограничений по доступу к коду, расположенному в секции прикладной программы |
| 2 | 1 | 0 | Команда SPM не может осуществлять запись по адресам, находящимся в пределах секции прикладной программы |
| 3 | 0 | 0 | Команда SPM не может осуществлять запись по адресам, находящимся в пределах секции прикладной программы, и команда LPM (ELPM), вызываемая из секции загрузчика, не может осуществлять чтение из секции прикладной программы |
| 4 | 0 | 1 | Команда LPM (ELPM), вызываемая из секции загрузчика, не может осуществлять чтение из секции прикладной программы |

Таблица 4. Режимы защиты секции загрузчика

| № режима | Ячейки защиты | | Описание |
|---|---|---|---|
| | BLB12 | BLB11 | |
| 1 | 1 | 1 | Нет никаких ограничений по доступу к коду, расположенному в секции загрузчика |
| 2 | 1 | 0 | Команда SPM не может осуществлять запись по адресам, находящимся в пределах секции загрузчика |
| 3 | 0 | 0 | Команда SPM не может осуществлять запись по адресам, находящимся в пределах секции загрузчика, и команда LPM (ELPM), вызываемая из секции прикладной программы, не может осуществлять чтение из секции загрузчика |
| 4 | 0 | 1 | Команда LPM (ELPM), вызываемая из секции прикладной программы, не может осуществлять чтение из секции |

| | | | |
|---|---|---|---|
| | | | загрузчика |

В исходном (незапрограммированом) состоянии во всех ячейках защиты содержится 1, после программирования – 0. Стирание ячеек (запись в них лог. 1) может быть произведено только при выполнении команды «Стирание кристалла», уничтожающей также содержимое FLASH и EEPROM-памяти.

Конфигурационные ячейки (Fuse Bits) определяют различные параметры конфигурации микроконтроллера. Эти ячейки расположены в отдельном адресном пространстве, доступном только при программировании.

Для изменения содержимого конфигурационных ячеек используются специальные команды программирования. Команда «Стирания кристалла» на состояние этих ячеек не влияет. При программировании ячеек защиты LB1 конфигурационные ячейки блокируются. Назначение конфигурационных ячеек приведено в табл 5, где в столбцах, отмеченных «звездочкой», указаны состояния конфигурационных ячеек по умолчанию.

Таблица 5 – Назначение конфигурационных ячееек

| Название | * | Назначение |
|---|---|---|
| CKSEL:<br><br>CKSEL3<br>CKSEL2<br>CKSEL1<br>CKSEL0 | <br><br>0<br>0<br>0<br>1 | Определяет режим работы тактового генератора, а также длительность задержки сброса $t_{TOUT}$ |
| BODLEVEL | 1 | Определяет порог срабатывания схемы BOR |
| BODEN | 1 | Разрешает/запрещает функционирование схемы BOR (0 - разрешена, 1 – запрещена) |
| SPIEN | 0 | Разрешает/запрещает прогриммирование по интерфейсу SPI (0 - разрешена, 1 – запрещена) |
| SUT:<br><br>SUT1<br>SUT0 | <br><br>1<br>0 | Определяет длительность задержки сброса $t_{TOUT}$ |
| CKOPT | 1 | Определяет функционирование тактового генератора, действие зависит от установок ячеек CKSEL |
| EESAVE | 1 | Определяет влияние команды «Стирание кристалла» на EEPROM-память (0 – не стирает, 1 – стирает) |
| BOOTSZ:<br><br>BOOTSZ1 | <br><br>0 | Определяет размер секции загрузчика |

| | | |
|---|---|---|
| BOOTSZ0 | 0 | |
| BOOTRST | 1 | Определяет положение вектора сброса |
| OCDEN | 1 | Разрешает/запрещает внутрисхемную отладку (0 - разрешена, 1 – запрещена) |
| JTAGEN | 0 | Разрешает/запрещает использование интерфейса JTAG (0 - разрешена, 1 – запрещена) |

## 4 ЗАДАНИЕ И ПОРЯДОК ЕГО ВЫПОЛНЕНИЯ

Осуществить программирование микроконтроллера ATmega16:

1. Войти в С-кросс компилятор CodeVisionAVR.
2. Создать новый проект File -> New -> Project.
3. Указать предварительные настройки для микроконтроллера ATmega16.
4. Сохранить проект File -> Generate, Save and Exit.
5. В созданном коде предварительных настроек вписать свой код:

```
while(1)        // Eternal loop
{
PORTB   = 0xFF; // Установка на выходе порта B сигнала высокого уровня
delay_ms(250); // Задержка 250 мс
PORTB   = 0x00; // Установка на выходе порта B сигнала низкого уровня
delay_ms(250); // Задержка 250 мс
}
```

6. Сохранить набранную программу, используя опции **file** и **save**.
7. Запустить компилятор Project -> Compile (F9).
8. После выполнения процедуры компиляции, среда выдает сообщения об ошибках (если такие есть) с указанием строки в программе, где эти ошибки обнаружены. Необходимо исправить ошибки в программе и повторить п.5. Повторять п.5 до тех пор пока компилятор не выдаст сообщения, что "ошибок 0", "замечаний 0".
9. После выполнения процедуры компиляции без ошибок и замечаний автоматически формируется (обновляется) hex-файл.
10. Далее необходимо выполнить программирование контроллера Tools -> Chip Programmer (Shift+F4). Выставив в соответствии с табл. 2-4 значения ячеек защиты и значения конфигурационных ячеек

осуществить запись кода во Flash и EEPROM-память (Program -> FLASH и Program -> EEPROM).

11. Выполнить запуск программы МК. Убедиться в правильности функционирования МК согласно написанной программе.

## КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Объясните назначение узлов лабораторного стенда.
2. Объясните назначение выводов микроконтроллера ATmega16.
3. Изложите методику компилирования программ.
4. Изложите методику программирования микроконтроллера ATmega16.
5. Объясните структуру и принципы построения программ для функционирования цифровых схем с использованием микроконтроллера ATmega16.
6. Объясните аппаратурную часть, которая использовалась при выполнении настоящей лабораторной работы.
7. Изложите задание и порядок выполнения работы.