

# AVRISP mkII – руководство пользователя

<b>Введение.....</b>	<b>1</b>
Введение.....	1
Начало работы.....	2
Установка программного обеспечения и USB-драйвера.....	3
Поддерживаемые устройства.....	5
<b>Интерфейсные программные средства.....</b>	<b>6</b>
Использование AVR Studio совместно с AVRISP mkII.....	6
Подключение.....	6
Автоматическое обновление.....	7
Program (Программирование).....	7
Fuses (Конфигурационные биты).....	8
LockBits (Биты блокировки).....	9
Advanced (Дополнительные возможности).....	10
Board (Параметры платы).....	11
Auto (Автоматический режим).....	13
Программное обеспечение для работы из командной строки.....	14
<b>Аппаратная часть.....</b>	<b>16</b>
Подключение AVRISP mkII.....	16
Описание аппаратной части.....	17
<b>Устранение неисправностей.....</b>	<b>20</b>
Устранение неисправностей.....	20
Обновление прошивки вручную.....	20

Более полную и новую информацию см. в Справке к программе AVR Studio 4.

## Введение

### Введение

AVRISP mkII совместно с программой AVR Studio может программировать все 8-разрядные микроконтроллеры с RISC-архитектурой серии AVR, обладающие возможностью внутрисхемного программирования (интерфейсом ISP).

Список поддерживаемых устройств вы можете найти на сайте производителя:

[http://www.atmel.com/dyn/products/tools\\_card.asp?tool\\_id=3808](http://www.atmel.com/dyn/products/tools_card.asp?tool_id=3808)

#### Характеристики AVRISP mkII

- совместим с AVR Studio (AVR Studio 4.12 и более поздними версиями);
- поддерживает все устройства серии AVR с возможностью внутрисхемного программирования (с интерфейсом ISP);
- поддержка программирования Flash и EEPROM;
- поддержка программирования битов конфигурации (*fuses*) и битов блокировки;
- возможность обновления прошивки для поддержки новых устройств;
- поддержка микроконтроллеров с напряжением питания от 1.8 В до 5.5 В;
- регулируемая скорость программирования (от 50 Гц до 8 МГц частоты SCK);
- совместим с USB 2.0 (Full-speed, 12 Мбит/с);
- питание от шины USB, не требует внешнего источника питания;
- защита интерфейсных линий программируемого устройства, в т.ч. защита от короткого замыкания.



AVRISP mkII поддерживается AVR Studio (см. требования AVR Studio). Последнюю версию AVR Studio можно найти по ссылке: <http://www.atmel.com/products/AVR/>.

- Пожалуйста, прочтите "Начало работы с AVRISP mkII".

### **Начало работы**

Пожалуйста, прочтите данный раздел, перед тем как подключить AVRISP mkII к компьютеру или программируемому устройству. Для начала работы с AVRISP mkII выполните следующие четыре шага:

- 1) Установите AVR Studio и драйвер USB.
- 2) Подключите AVRISP mkII к компьютеру, компьютер должен обнаружить и автоматически установить новое оборудование (AVRISP mkII).
- 3) Запустите AVR Studio и выберите AVRISP mkII Programming Dialog.
- 4) Подключите AVRISP mkII к программируемому устройству.

### **Установка USB**

Перед тем как использовать AVRISP mkII, необходимо установить AVR Studio и USB-драйвер. Пожалуйста, не подключайте AVRISP mkII к компьютеру до запуска программы установки USB-драйвера. Выполните процедуру установки так, как это описано в разделе "Установка программного обеспечения и USB-драйвера".

### **Комплектация AVRISP mkII**

Коробка с AVRISP mkII должна содержать следующие предметы:

- AVRISP mkII;
- USB-кабель;
- CD-диск "Atmel Technical Library" с техническими описаниями (*datasheets*), примерами применения (*application notes*) и программным обеспечением (*software*).



### Требования к системе

Ниже приведены минимальные аппаратные и программные требования к системе:

- процессор Pentium (рекомендуется Pentium II или выше);
- 64 Мбайт ОЗУ;
- 100 Мбайт свободного пространства на жёстком диске для установки AVR Studio 4.12 или более поздних версий;
- Windows 98, Windows ME, Windows 2000 или Windows XP;
- USB-порт, способный обеспечить 200 мА.

*Примечание:* Windows 95 и Windows NT не поддерживают USB, поэтому не могут быть использованы с AVRISP mkII.

### Требования к AVR Studio

Для использования совместно с AVRISP mkII необходима программа AVR Studio версии 4.12 (или более поздних версий). Последнюю версию AVR Studio можно найти по ссылке: <http://www.atmel.com/products/AVR/>.

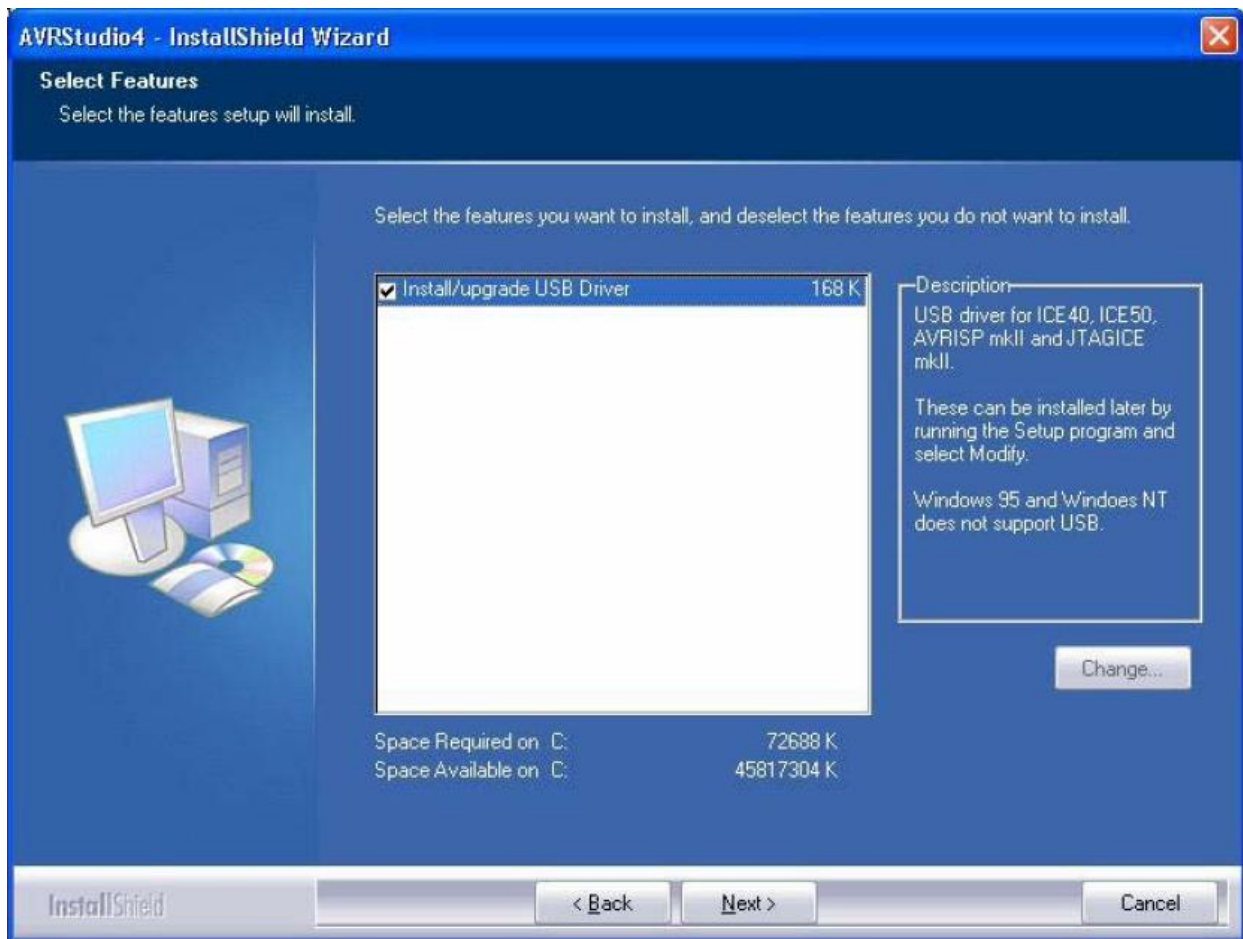
*Примечание:* AVR Studio и USB-драйвер необходимо установить до подключения AVRISP mkII к компьютеру.

### **Установка программного обеспечения и USB-драйвера**

Для использования AVRISP mkII необходимо установить USB-драйвер. Пожалуйста, не подключайте AVRISP mkII к компьютеру до запуска программы установки USB-драйвера. Установка USB-драйвера выполняется во время установки AVR Studio.

*Примечание:* для AVRISP mkII необходима AVR Studio версии 4.12 или более поздних версий. Последнюю версию AVR Studio можно найти по ссылке: <http://www.atmel.com/products/AVR/>.

Запустите программу установки AVR Studio. В процессе установки появится диалоговое окно, показанное на рис. ниже.



Чтобы установить USB-драйвер, установите флажок "Install/Upgrade USB Driver", в результате USB-драйвер будет установлен автоматически.

#### Установка нового оборудования

После окончания установки AVR Studio и USB-драйвера подключите USB-кабель к компьютеру и AVRISP mkII (AVRISP mkII питается от USB). Если AVRISP mkII подключается к компьютеру впервые, отобразится следующее диалоговое окно:



В Windows XP вам понадобится несколько раз нажать кнопку "Next". Пожалуйста, дождитесь завершения процесса установки. Он может занять от нескольких секунд до нескольких минут в зависимости от компьютера и операционной системы.

Если USB-драйвер установлен корректно, и AVRISP mkII подключен к компьютеру, должен загореться зелёный светодиод, расположенный внутри корпуса AVRISP mkII рядом с USB-разъёмом.

Если после установки USB-драйвера AVR Studio по каким-то причинам не определяет AVRISP mkII, попробуйте перезагрузить компьютер.

### Установка USB-драйвера после установки AVR Studio

USB-драйвер можно установить и после установки AVR Studio. Для этого выполните следующие действия:

- 1) Откройте "Панель управления" на вашем ПК (Windows 95 и Windows NT не поддерживают USB);
- 2) Выберите "Установка и удаление программ";
- 3) Выберите "AVRStudio4" в списке программ;
- 4) Нажмите кнопку "Изменить";
- 5) Выберите "Modify";
- 6) Выберите "Install/upgrade USB Driver".

После чего USB-драйвер будет установлен.

*Примечание:* для AVRISP mkII необходим USB-порт, способный обеспечить 200 мА.

- Пожалуйста, прочтите об использовании AVR Studio с AVRISP mkII.

### **Поддерживаемые устройства**

AVRISP mkII поддерживает все 8-разрядные микроконтроллеры с RISC-архитектурой серии AVR, обладающие возможностью внутрисхемного программирования (интерфейсом ISP). Поддержка новых устройств будет добавлена в новых версиях AVR Studio. Последняя версия AVR Studio всегда доступна по ссылке: <http://www.atmel.com/products/AVR/>.

Если доступна новая прошивка, AVR Studio предложит обновить прошивку AVRISP mkII.

Поддерживаются следующие устройства:

Tiny	Mega	Другие
ATtiny12	ATmega48	AT86RF401
ATtiny13	ATmega8	AT89S51
ATtiny15	ATmega88	AT89S52
ATtiny22	ATmega8515	AT90PWM2
ATtiny2313	ATmega8535	AT90PWM3
ATtiny24	ATmega16	AT90CAN32
ATtiny26	ATmega162	AT90CAN128
ATtiny45	ATmega165	
ATtiny861	ATmega168	
	ATmega169	
	ATmega32	
	ATmega325	
	ATmega3250	
	ATmega329	
	ATmega3290	
	ATmega64	
	ATmega640	
	ATmega644	
	ATmega645	
	ATmega6450	
	ATmega649	
	ATmega6490	
	ATmega128	
	ATmega1280	
	ATmega1281	
	ATmega2560	
	ATmega2561	

AVRISP mkII поддерживает все различные по напряжению и скорости версии указанных в таблице устройств.



## Интерфейсные программные средства

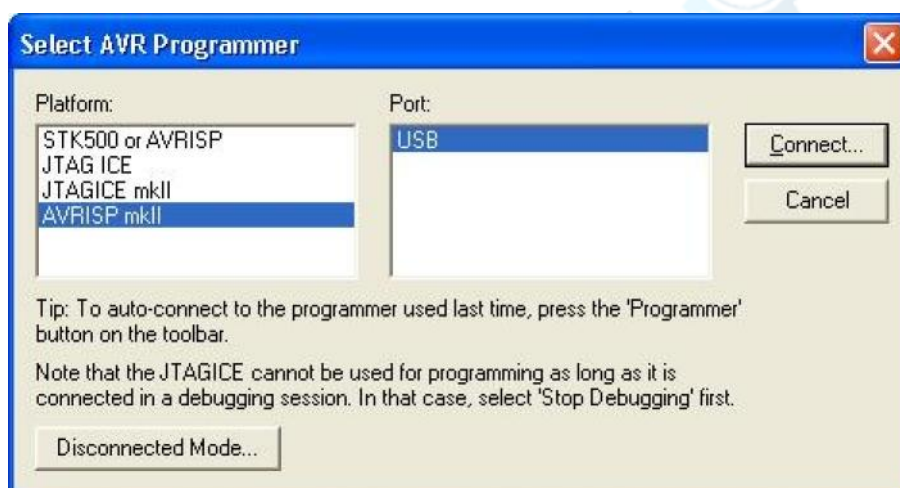
### Использование AVR Studio совместно с AVRISP mkII


AVR Studio представляет собой пользовательский интерфейс для AVRISP mkII. Последнюю версию AVR Studio можно найти по ссылке: <http://www.atmel.com/products/AVR/>.

*Примечание:* до подключения AVRISP mkII к компьютеру необходимо установить USB-драйвер. Более подробно вы можете прочитать об этом в разделе "**Установка программного обеспечения и USB-драйвера**".

### Подключение

Запустите диалог программирования, нажав одну из двух кнопок подключения  на панели инструментов в AVR Studio. Левая кнопка () позволяет выбрать, к какому программатору и коммуникационному порту следует подключиться:



При нажатии на правую кнопку () выполняется "быстрое" подключение к последнему выбранному инструменту и коммуникационному порту. Если устройство с такими параметрами найдено не будет, программа выполнит сканирование всех остальных коммуникационных портов в поисках подключенного к ним какого-либо инструментального средства AVR.

Если программа обнаружит, что к компьютеру подключено несколько программаторов AVRISP mkII, она предложит пользователю выбрать один из них, указав необходимый серийный номер (его можно посмотреть на нижней стороне программатора).

*Примечание:* каждая копия программы AVR Studio может открыть только одно инструментальное средство, подключенное через USB. Если используется несколько AVRISP mkII и/или другие инструменты AVR, подключаемые через USB, для каждого из них следует запускать отдельную копию программы AVR Studio.

Более подробно о диалоговом окне программирования для AVRISP mkII вы можете прочитать в следующих разделах:

- Program (Программирование);
- Fuses (Конфигурационные биты);
- LockBits (Биты блокировки);
- Advanced (Дополнительные возможности);

- Board (Параметры платы);
- Auto (Автоматический режим).

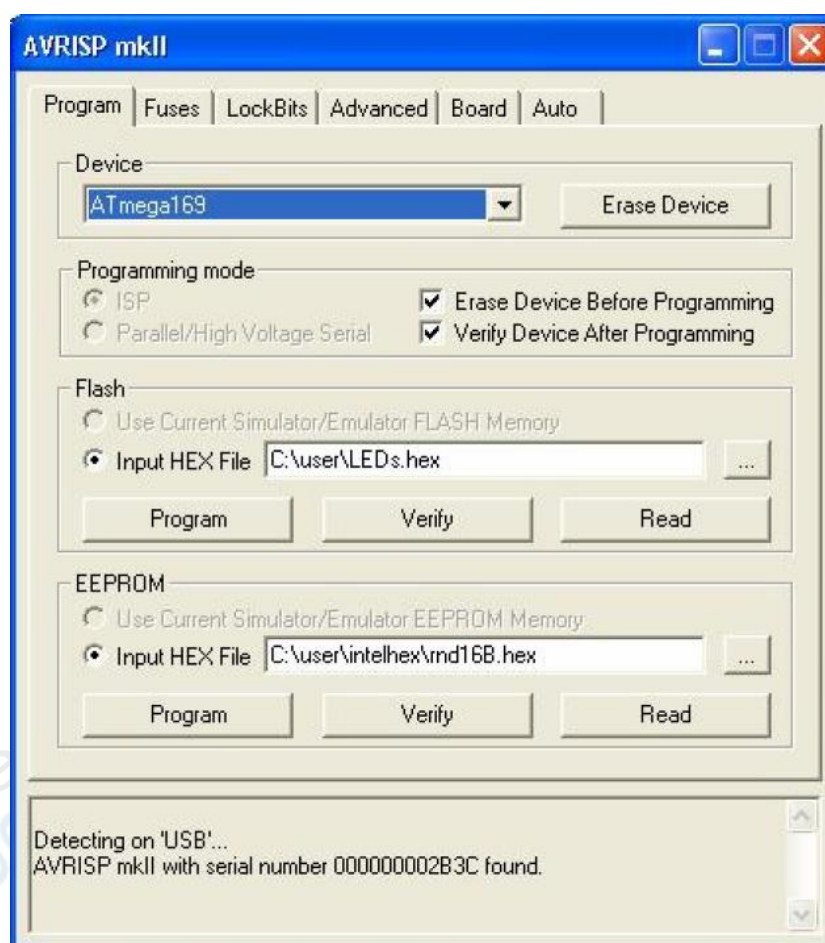
### Автоматическое обновление

AVR Studio содержит последнюю версию прошивки для AVRISP mkII. Если доступна более новая версия прошивки, AVR Studio автоматически определит это и запустит программу обновления.

О том, как подключиться к программируемому устройству, см. в разделе "**Подключение AVRISP mkII**".

### Program (Программирование)

Настройки программирования разделены на четыре группы.



#### Device (Устройство)

Из выпадающего списка выбирается нужный микроконтроллер. В этой группе также располагается кнопка "Erase Device" ("Стереть устройство"), при нажатии на которую выполняется полное стирание выбранного устройства. При этом стираются Flash и EEPROM, а также биты блокировки.

#### Programming mode (Режим программирования)

В этой группе выбирается режим программирования. AVRISP mkII поддерживает только режим ISP. Если установить флажок "Erase Device Before Programming" ("Стереть устройство перед программированием"), AVRISP mkII будет выполнять полное стирание микроконтроллера перед его программированием. Если установить флажок "Verify Device After Programming" ("Проверить

устройство после программирования"), AVRISP mkII после программирования будет выполнять проверку памяти.

## Flash

Если интерфейс пользователя AVRISP mkII открыт при отсутствии загруженного в AVR Studio проекта, опция "*Use Current Simulator/Emulator FLASH Memory*" ("*Использовать текущую Flash-память симулятора/эмулятора*") будет недоступна. Если же проект открыт, эта опция позволяет программирование Flash-памяти содержимым, присутствующим в данный момент в окне Flash-памяти AVR Studio. (Более подробно об окнах памяти AVR Studio см. в системе помощи программы AVR Studio).

Если же открытых проектов нет, либо исходный код хранится в отдельном HEX-файле, выберите опцию "*Input HEX File*" ("*Ввод HEX-файла*"). Укажите нужный файл, нажав на кнопку "...", либо введите полный путь к файлу и его имя в текстовое поле. Выбранный файл должен быть в формате "Intel-hex" или "extended Intel-hex".

## EEPROM

Если интерфейс пользователя AVRISP mkII открыт при отсутствии загруженного в AVR Studio проекта, опция "*Use Current Simulator/Emulator EEPROM Memory*" ("*Использовать текущую EEPROM-память симулятора/эмулятора*") будет недоступна. Если же проект открыт, эта опция позволяет программирование EEPROM-памяти содержимым, присутствующим в данный момент в окне EEPROM-памяти AVR Studio. (Более подробно об окнах памяти AVR Studio см. в системе помощи программы AVR Studio).

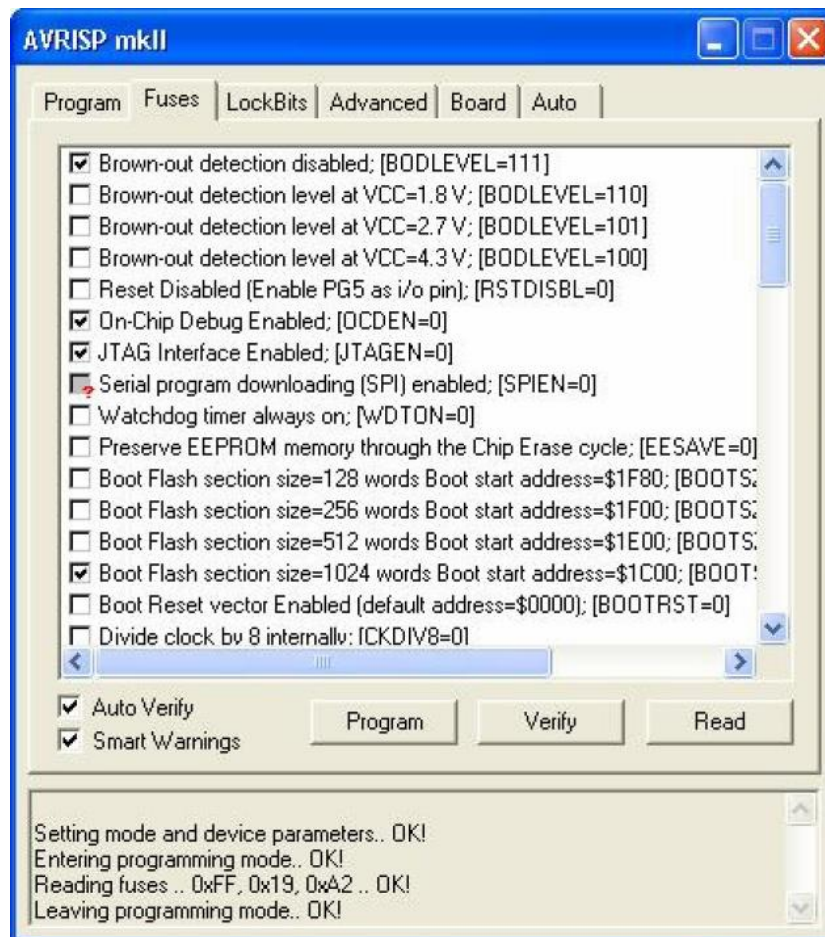
Если же открытых проектов нет, либо исходный код хранится в отдельном HEX-файле, выберите опцию "*Input HEX File*" ("*Ввод HEX-файла*"). Укажите нужный файл, нажав на кнопку "...", либо введите полный путь к файлу и его имя в текстовое поле. Выбранный файл должен быть в формате "Intel-hex" или "extended Intel-hex".

## **Fuses (Конфигурационные биты)**

На вкладке "*Fuses*" располагается список доступных конфигурационных битов. Некоторые пункты могут подразумевать комбинации из нескольких конфигурационных битов; для заданного конфигурационного режима нужные биты устанавливаются автоматически.

Обратите внимание, что полное стирание микроконтроллера ("*Chip Erase*") не влияет на конфигурационные биты.





При нажатии на кнопку "Read" ("Считать") из микроконтроллера будут считаны текущие значения конфигурационных битов. При нажатии на кнопку "Write" ("Записать") текущие установки будут записаны в микроконтроллер. Установка любого из указанных флажков означает, что этот конфигурационный бит (биты) необходимо включить/запрограммировать, т.е. записать "0" в соответствующую ячейку микроконтроллера.

Более подробную информацию о конфигурационных битах и их функциональном назначении вы найдёте в техническом описании на конкретный микроконтроллер.

Текст в квадратных скобках, следующий за описанием каждого конфигурационного бита, показывает название группы битов и их фактическое значение. Обратите внимание, что, как указано в техническом описании на AVR, **конфигурационный бит является запрограммированным, если в него записан "0", и незапрограммированным, если в него записана "1"**.

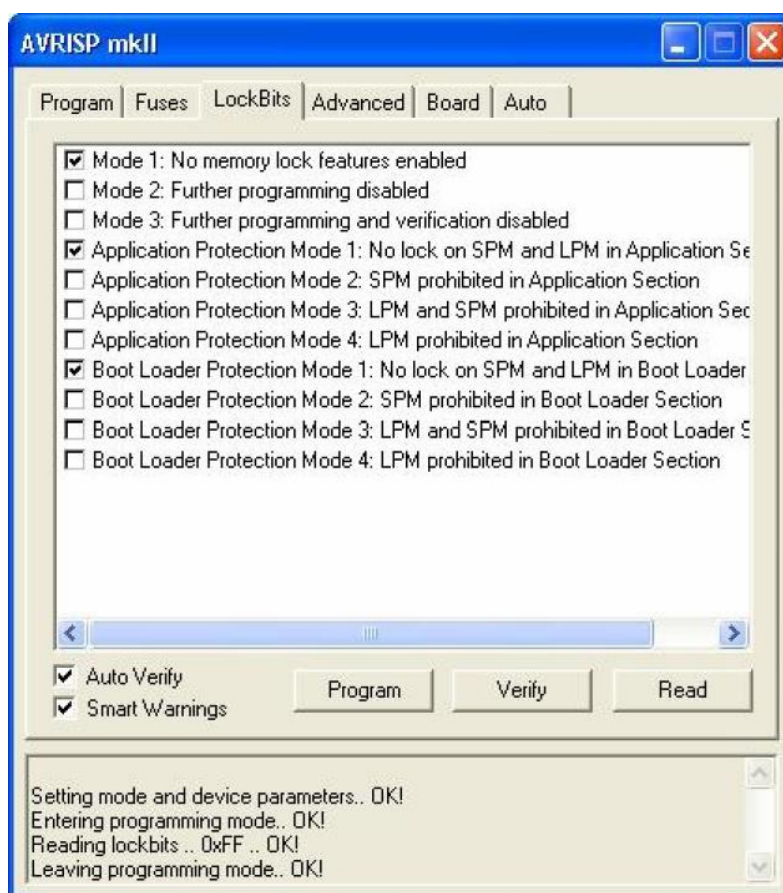
В расположенном внизу окне состояния значения конфигурационных битов приводятся в шестнадцатеричном формате, при этом младшие байты располагаются слева.

### LockBits (Биты блокировки)

На вкладке "LockBits" приведены режимы блокировки, применимые к выбранному микроконтроллеру. Режим блокировки может состоять из комбинации нескольких битов блокировки; для выбранного режима блокировки нужные биты устанавливаются автоматически.

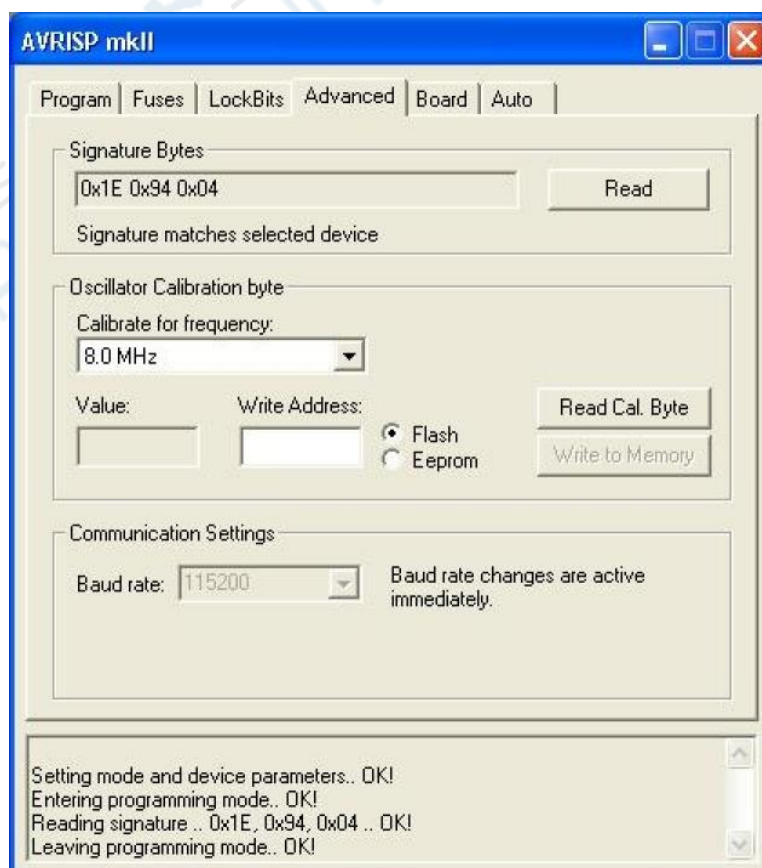
Как только включен уровень защиты "Режим блокировки", понизить его выбором "более низкой" степени защиты с помощью выбора другого "Режима блокировки" невозможно. Единственный способ удалить установленные биты блокировки – это выполнить полное стирание устройства.

*Примечание:* Если у программируемого микроконтроллера установлен конфигурационный бит "EESAVE", содержимое памяти EEPROM будет сохранено даже при полном стирании устройства.



### Advanced (Дополнительные возможности)

Настройки на вкладке "Advanced" разделены на три группы.



### Signature Bytes (Байты сигнатуры)

При нажатии на кнопку "*Read Signature*" ("*Считать сигнатуру*") из микроконтроллера считываются байты сигнатуры, выполняющие функцию идентификатора микроконтроллера. После чтения программа проверяет, является ли сигнатура корректной согласно выбранному устройству. Более подробную информацию о байтах сигнатуры см. в технических описаниях на AVR.

### Oscillator Calibration Byte (Байт калибровки генератора)

Байт калибровки содержит подстроечное значение, которое необходимо записать в регистр OSCCAL для настройки частоты внутреннего RC-генератора. Байт калибровки генератора записывается в микроконтроллер в процессе производства и не может быть стёрт или изменён пользователем.

### Чтение байтов калибровки генератора

При нажатии на кнопку "*Read Cal. Byte*" из микроконтроллера считываются калибровочные значения и отображаются в текстовом поле "*Value*". Обратите внимание, что для некоторых микроконтроллеров калибровочные значения не доступны напрямую во время выполнения программы, и их необходимо записывать в ячейку памяти во время программирования, если планируется их использование в программе. Если эта опция не доступна, значит, выбранное устройство не имеет настраиваемого внутреннего RC-генератора.

### Запись байта калибровки генератора

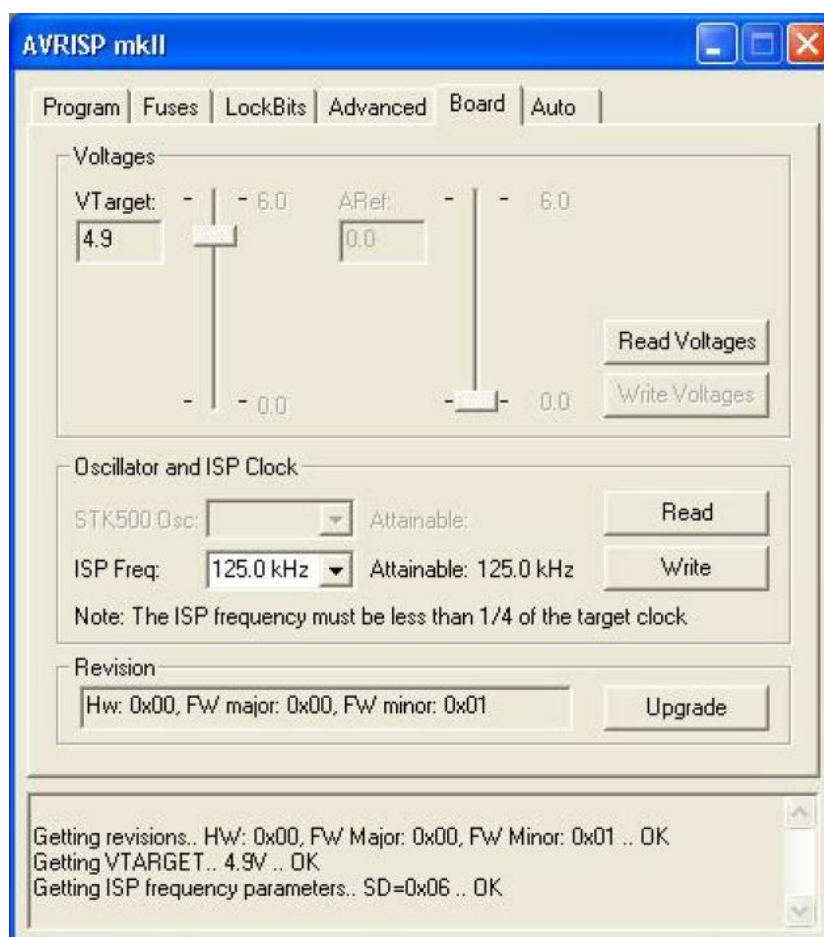
Поскольку для некоторых микроконтроллеров байт калибровки не доступен в процессе выполнения программы, пользователь должен записать калибровочный байт в известную ячейку памяти Flash или EEPROM. Для этого в текстовом поле "*Write Address*" ("*Запись по адресу*") необходимо указать желаемый адрес памяти и затем нажать кнопку "*Write to Memory*" ("*Записать в память*"). После чего калибровочный байт запишется в память, указанную переключателями "*Flash*" и "*EEPROM*".

### Communication Settings (Настройки коммуникационного порта)

Настройки коммуникационного порта недоступны для AVRISP mkII, поскольку он использует USB.

### **Board (Параметры платы)**

Настройки на вкладке "*Board*" делятся на три группы.



### Voltages (Напряжения)

**VTarget** представляет  $V_{cc}$  программируемой системы.

**ARef** не применимо к AVRISP mkII, поскольку он не поддерживает чтение ARef.

### Oscillator and ISP Clock (Тактовая частота ISP и генератора)

**STK500 Osc** не применимо к AVRISP mkII, поскольку он не может обеспечить тактирование программируемой системы.

**ISP Freq** задаёт тактовую частоту ISP (сигнал SCK интерфейса ISP).

Для большинства AVR в техническом описании приводится следующая частота ISP (SCK):

*Минимальные периоды высокого и низкого уровня для входа последовательной синхронизации (SCK) определяются следующим образом:*

*Низкий уровень: > 2 тактовых цикла ЦПУ для  $f_{clk} < 12$  МГц, 3 тактовых цикла ЦПУ для  $f_{clk} 12$  МГц*

*Высокий уровень: > 2 тактовых цикла ЦПУ для  $f_{clk} < 12$  МГц, 3 тактовых цикла ЦПУ для  $f_{clk} 12$  МГц*

Это означает, что для тактовой частоты программируемого микроконтроллера менее 12 МГц, частота ISP должна быть равной или меньшей  $1/4$  этой частоты.

Для тактовой частоты микроконтроллера выше 12 МГц, частота ISP должна быть равной или меньшей  $1/6$  этой частоты.

Обратите внимание, что для микроконтроллеров с конфигурационным битом CKDIV (*Clock Divide – деление частоты*) и/или с CLKPR (*Clock Prescaler Register – регистр предделителя тактовой частоты*) частоту ISP необходимо вычислять из частоты, получаемой после делителя.

*Примечание:* При использовании внутреннего RC-генератора убедитесь, что он откалиброван для использования с максимальной частотой ISP.

*Совет:* Если у вас возникли проблемы с подключением к программируемому микроконтроллеру, попробуйте понизить частоту ISP.

Пожалуйста, обратитесь к техническому описанию на устройство AVR.

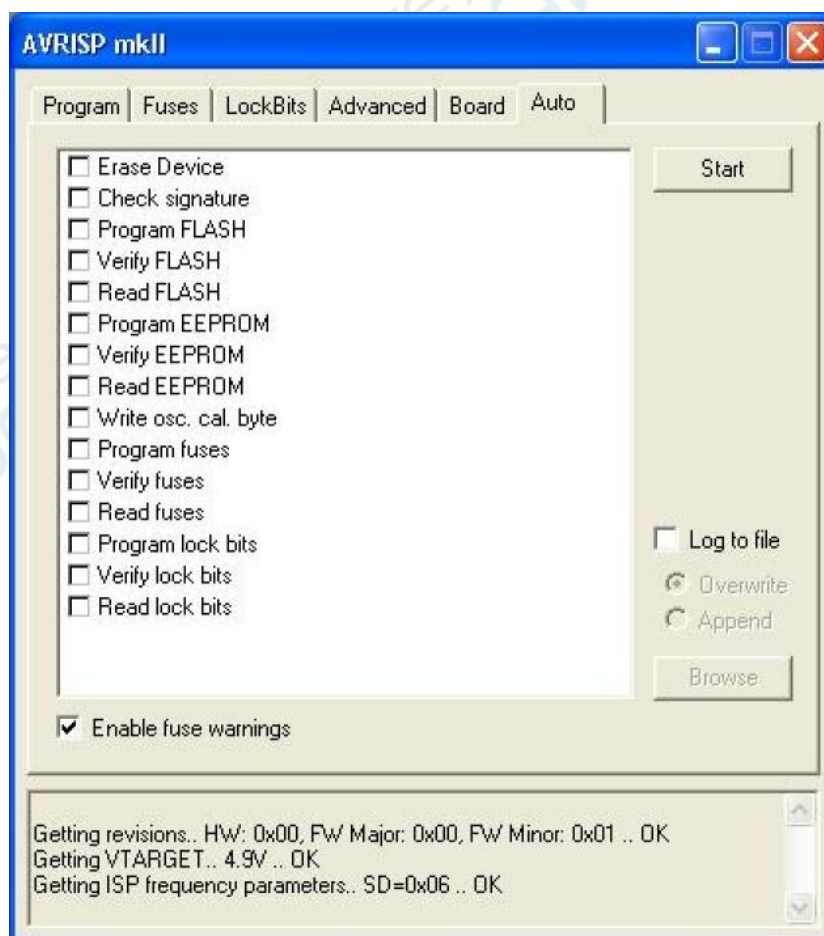
*Примечание:* AVRISP mkII не поддерживает программирование Flash или EEPROM на частоте ниже 2 кГц. Однако полное стирание устройства будет работать вплоть до частоты 51 Гц. Если из-за значения в регистре CLKPR частота микроконтроллера настолько низка, что он не программируется, то перед его программированием выполните полное стирание устройства.

### Revision (Номер ревизии)

Показывает номер ревизии (модификации) аппаратной части (*HW - Hardware*) и прошивки (*FW - Firmware*) AVRISP mkII.

### **Auto (Автоматический режим)**

Для программирования нескольких микроконтроллеров одной и той же прошивкой, вкладка "Auto" предлагает инструмент для автоматизации выполнения заданной пользователем последовательности команд. Команды приведены в списке в порядке выполнения. Чтобы разрешить команду, установите соответствующий флажок. Например, если включен только флажок "Program FLASH" ("Программирование FLASH"), то при нажатии на кнопку "Start" Flash-память микроконтроллера будет запрограммирована содержимым HEX-файла, указанного в настройках на вкладке "Program". Команды используют установки с других вкладок диалогового окна настроек AVRISP mkII.



При необходимости в текстовый файл можно записать протокол выполнения команд, включив флажок "Log to file" ("Запись протокола в файл").

### Настройка системы для автоматического программирования

Установите флажки для команд, которые необходимо выполнить. Типовая последовательность со стиранием и последующим программированием устройства показана на рис. выше. Сначала выполняется полное стирание устройства, затем оба типа памяти программируются и проверяются и, наконец, программируются конфигурационные биты и биты блокировки.

Однажды настроенная, одна и та же последовательность программирования будет выполняться по каждому нажатию на кнопку "Start". Это избавляет от необходимости выполнять лишнюю работу, а также уменьшает возможные ошибки, могущие возникнуть из-за ошибок пользователя.

### Запись протокола автоматического программирования в файл

После установки флажка "Log to file" весь вывод от команд будет записываться в текстовый файл. Файл выбирается/создается нажатием кнопки "Browse" ("Обзор") и выбором места, где он размещается или должен быть создан. Вывод будет сохраняться в этот файл и в последствии его можно будет посмотреть и отредактировать в текстовом редакторе.

### **Программное обеспечение для работы из командной строки**

Командная строка интерфейса программирования такая же, как и для STK500. Он называется "stk500.exe" и позволяет создать простой пакетный файл для автоматического программирования.

Ниже будет приведен пример создания простого пакетного файла для автоматизации программирования устройства.

#### **Ключи:**

**[-d device name] [-m s|p] [-if infile] [-ie infile] [-of outfile]  
 [-oe outfile] [-s] [-O address] [-Sf addr] [-Se addr] [-e] [-p f|e|b]  
 [-r f|e|b] [-v f|e|b] [-I value] [-L value] [-y] [-f value] [-E value]  
 [-F value] [-G value] [-q] [-Y] [-Z address] [-c port] [-ut value]  
 [-Wt value] [-ua value] [-wt] [-wa] [-b h|s] [-! freq] [-t] [-I freq]  
 [-J] [-h] [-?]**

#### Примеры:

```
STK500.EXE -cUSB -dATmega128 -e -ifFlash.hex -pf -vf
STK500.EXE -cUSB -dATmega128 -fF73A -FF73A -EFF -GFF
```

Обратите внимание, что между ключами и их параметрами отсутствуют пробелы, и что шестнадцатеричные значения указываются без префикса "0x".

#### Параметры:

<b>d</b>	Имя устройства. Чтобы посмотреть список поддерживаемых устройств, воспользуйтесь командой <b>STK500.EXE -?</b>
<b>m</b>	Выбор режима программирования: последовательное ( <b>s</b> ) или параллельное/с высоким напряжением ( <b>p</b> ). По умолчанию, если этот параметр не указывается, используется последовательный режим ( <b>s</b> ).
<b>if</b>	Имя входного файла FLASH-памяти. Требуется для программирования или проверки FLASH-памяти. Формат файла – "Intel Extended HEX".
<b>ie</b>	Имя входного файла EEPROM-памяти. Требуется для программирования или проверки EEPROM-памяти.

	Формат файла – "Intel Extended HEX".
<b>of</b>	Имя выходного файла FLASH-памяти. Требуется для считывания FLASH-памяти. Формат файла – "Intel Extended HEX".
<b>oe</b>	Имя выходного файла EEPROM-памяти. Требуется для считывания EEPROM-памяти. Формат файла – "Intel Extended HEX".
<b>s</b>	Считать байты сигнатуры.
<b>O</b>	Считать байт калибровки генератора. ' <b>address</b> ' – адрес байта калибровки, указанный в техническом описании на устройство.
<b>Sf</b>	Записать байт калибровки генератора во FLASH-память. ' <b>addr</b> ' – адрес байта.
<b>Se</b>	Записать байт калибровки генератора в EEPROM-память. ' <b>addr</b> ' – адрес байта.
<b>e</b>	Выполнить полное стирание устройства. Если указывается с другим параметром программирования, то сначала устройство будет стёрто, а затем выполнятся остальные команды программирования.
<b>p</b>	Запрограммировать устройство: FLASH ( <b>f</b> ), EEPROM ( <b>e</b> ) или обе памяти ( <b>b</b> ). При этом необходимо указать соответствующие входные файлы.
<b>r</b>	Считать устройство: FLASH ( <b>f</b> ), EEPROM ( <b>e</b> ) или обе памяти ( <b>b</b> ). При этом необходимо указать соответствующие выходные файлы.
<b>v</b>	Проверить устройство: FLASH ( <b>f</b> ), EEPROM ( <b>e</b> ) или обе памяти ( <b>b</b> ). Можно использовать с параметром <b>-p</b> или как самостоятельный параметр. При этом необходимо указать соответствующие входные файлы.
<b>l</b>	Установить байт блокировки. ' <b>value</b> ' – 8-битное значение в шестнадцатеричном формате.
<b>L</b>	Проверить байт блокировки. ' <b>value</b> ' – 8-битное значение в шестнадцатеричном формате, с которым следует сравнить байт блокировки.
<b>y</b>	Считать байт блокировки.
<b>f</b>	Установить конфигурационные байты. ' <b>value</b> ' – 16-битное значение в шестнадцатеричном формате, описывающее установки для старшей и младшей части конфигурационного слова.
<b>E</b>	Установить дополнительный конфигурационный байт. ' <b>value</b> ' – 8-битное значение в шестнадцатеричном формате, описывающее установки дополнительного конфигурационного байта.
<b>F</b>	Проверить конфигурационные байты. ' <b>value</b> ' – 16-битное значение в шестнадцатеричном формате, с которым следует провести сравнение.
<b>G</b>	Проверить дополнительный конфигурационный байт. ' <b>value</b> ' – 8-битное значение в шестнадцатеричном формате, с которым следует провести сравнение.
<b>q</b>	Считать конфигурационные биты.
<b>af</b>	Диапазон адресов FLASH. Задаёт рабочий диапазон адресов для Flash-памяти. По умолчанию используется вся FLASH. Адреса байт-ориентированные.
<b>ae</b>	Диапазон адресов EEPROM. Задаёт рабочий диапазон адресов для EEPROM-памяти. По умолчанию используется вся EEPROM. Адреса байт-ориентированные.
<b>Y</b>	Выполнить последовательность калибровки генератора. Более подробную информацию см. в примере применения AVR053.
<b>Z</b>	Загружает значение из ячейки EEPROM перед выполнением стирания устройства. Это значение затем может быть запрограммировано во FLASH или EEPROM с помощью опции <b>S</b> . Адреса представляют собой шестнадцатеричные значения. Более подробную информацию см. в примере применения AVR053.
<b>c</b>	Выбор коммуникационного порта: от ' <b>com1</b> ' до ' <b>com8</b> '. Если этот параметр не указан, программа выполнит сканирование коммуникационных портов в поисках STK500.
<b>ut</b>	Установить значение напряжения программируемого устройства VTARGET в вольтах. ' <b>value</b> ' – значение с плавающей точкой от 0.0 до 6.0, описывающее новое значение напряжения.
<b>ua</b>	Установить значение регулируемого напряжения AREF в вольтах. ' <b>value</b> ' – значение с плавающей точкой от 0.0 до 6.0, описывающее новое значение напряжения.
<b>wt</b>	Считать текущее значение напряжения программируемого устройства VTARGET.
<b>wa</b>	Считать текущее значение регулируемого напряжения AREF.
<b>Wt</b>	Проверить, что VTARGET соответствует заданному значению с точностью 5%. ' <b>value</b> ' – значение с плавающей точкой от 0.0 до 6.0.
<b>b</b>	Считать номер ревизии: аппаратной части ( <b>h</b> ) и программной части ( <b>s</b> ).
<b>!</b>	Установить частоту генератора. ' <b>freq</b> ' можно указывать в Гц, кГц или МГц.
<b>t</b>	Считать частоту генератора.
<b>I</b>	Установить частоту ISP. ' <b>freq</b> ' можно указывать в Гц, кГц или МГц.
<b>J</b>	Считать частоту ISP.
<b>g</b>	Выполнение в фоновом режиме.
<b>z</b>	Не используется. Поддерживается для обратной совместимости.
<b>h</b>	Справка (перекрывает все другие опции).

Поддерживаемые устройства:

чтобы отобразить поддерживаемые устройства, наберите "STK500.EXE -?".

### Пример 1

```
stk500 -cUSB -dATMEGA8515 -e -pf -ifm8515test.hex
```

Расшифровка:

<b>stk500</b>	Запустить файл <b>stk500.exe</b> .
<b>-cUSB</b>	В качестве коммуникационного порта выбрать <b>USB</b> .
<b>-dAT90S8515</b>	Выбрать устройство <b>AT90S8515</b> .
<b>-e</b>	Выполнить полное стирание устройства.
<b>-if8515test.hex</b>	Выбрать файл " <b>8515test.hex</b> " для программирования во flash.
<b>-pf</b>	Запрограммировать <b>flash</b> (выбранным выше файлом).

### Пример 2

```
stk500 -cUSB:00001A52 -ut4.5 -ua4.5 -!3686000 -I460800 -dATMEGA8515 -IFF -fD9E4 -ms -ifC:\input.hex -pf -vf -e
```

Расшифровка:

<b>stk500</b>	Запустить файл <b>stk500.exe</b> .
<b>-cUSB:00001A52</b>	Выбрать <b>USB</b> -порт, к которому подключен <b>AVRISP mkII</b> с заданным <b>серийным номером*</b> .
<b>-ut4.5</b>	Задать значение напряжения программируемого устройства ( <b>4.5 В</b> ).
<b>-ua4.5</b>	Задать значение аналогового опорного напряжения ( <b>4.5 В</b> ).
<b>-!3686000</b>	Установить тактовую частоту STK500 в <b>3.68 МГц</b> .
<b>-I460800</b>	Установить частоту ISP в <b>460.8 кГц</b> .
<b>-dATMEGA8515</b>	Выбрать устройство <b>ATmega8515</b> .
<b>-e</b>	Выполнить полное стирание устройства.
<b>-IFF</b>	Установить биты блокировки в <b>0xFF</b> .
<b>-fD9E4</b>	Установить конфигурационные биты в <b>0xD9E4</b> (младший байт – в <b>0xD9</b> , старший байт – в <b>0xE4</b> ).
<b>-ms</b>	Выбрать <b>последовательный</b> режим программирования.
<b>-ifC:\input.hex</b>	Выбрать файл " <b>input.hex</b> " для программирования во flash.
<b>-pf</b>	Запрограммировать <b>flash</b> (указанным выше файлом).
<b>-vf</b>	Проверить <b>flash</b> (по указанному выше файлу).

\* Обратите внимание, что, в отличие от COM-портов, USB-порт не имеет номера. Если к одному ПК подключено несколько AVRISP mkII или другие инструменты программирования ATMEL AVR, для обращения к нужному AVRISP mkII укажите серийный номер. Серийный номер можно найти на нижней стороне AVRISP mkII.

## Аппаратная часть

### Подключение AVRISP mkII

В этом разделе описывается, как правильно подключить AVRISP mkII к хосту ПК и к программируемому устройству. Обратите внимание, что предварительно необходимо установить AVR

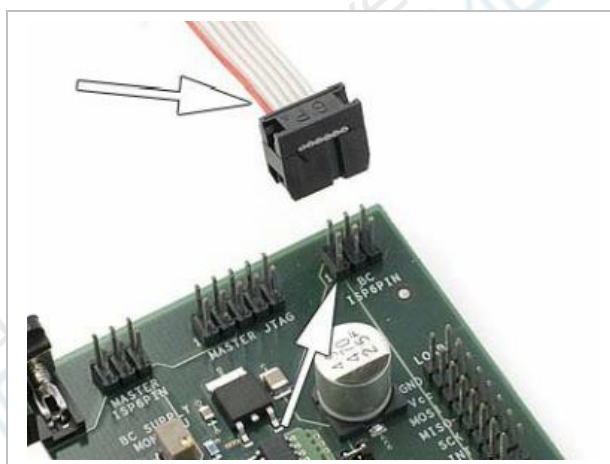


Studio и USB-драйвер. Подключать AVRISP mkII к компьютеру следует до подключения к программируемому устройству.



Если USB-драйвер установлен, то при подключении AVRISP mkII к компьютеру должен загореться зелёный светодиод, расположенный внутри корпуса рядом с USB-разъёмом. Главный светодиод состояния останется красным до подключения к программируемому устройству.

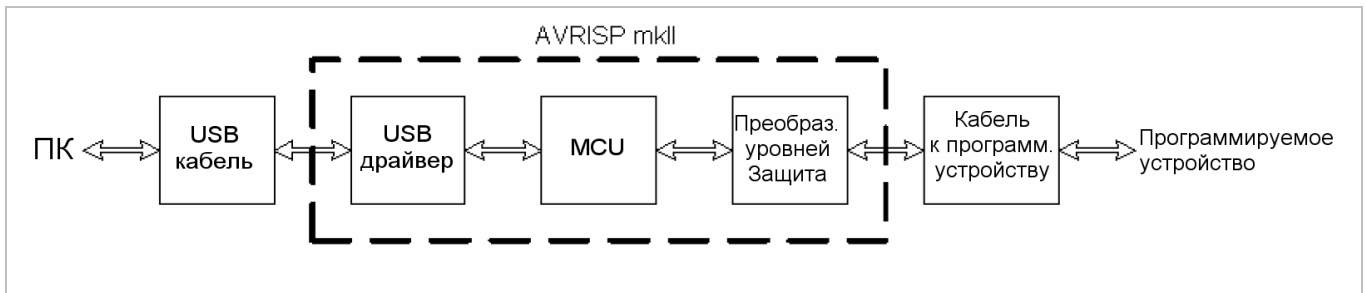
После подключения AVRISP mkII к ПК можно подключить его к микроконтроллеру. Красная полоска на кабеле ISP обозначает вывод 1, его следует присоединять к выводу 1 на разъёме ISP на плате с программируемым устройством.



Если AVRISP mkII подключен и к ПК, и к плате с программируемым микроконтроллером, главный светодиод состояния должен загореться зелёным светом, указывая, что обнаружено питание программируемого устройства. Теперь AVRISP mkII готов к работе с AVR Studio или программным обеспечением для работы из командной строки.

### Описание аппаратной части

Ниже приведена блок-схема аппаратной части AVRISP mkII:



## USB

Интерфейс USB – это USB 1.1 (USB 2.0 Full Speed) 12 Мбит/с.

Поскольку AVRISP mkII получает питание от USB-порта, необходимо, чтобы порт мог выдавать до 200 мА. Порты ПК, а также USB-концентраторы (хабы) с отдельным питанием, как правило, соответствуют этим требованиям.

## MCU

Управляющий микроконтроллер (MCU) управляет всей связью между программируемым микроконтроллером и клиентской программой. AVRISP mkII полностью управляется программно из AVR Studio, ручной настройки AVRISP mkII не требуется.

## Светодиод состояния

Состояние AVRISP mkII индицируется трёхцветным светодиодом. В случае ошибок решение проблемы см. в руководстве по устранению неисправностей.



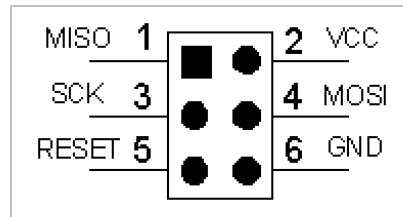
Цвет светодиода	Описание
Красный	Состояние простоя – Питание программируемого микроконтроллера не обнаружено
Зелёный	Состояние простоя – Питание программируемого микроконтроллера обнаружено
Оранжевый	Занят – Идёт программирование
Оранжевый мигающий	Кабель подключен к программируемому устройству неправильно.
Красный мигающий	Короткое замыкание на программируемом устройстве
Красный/оранжевый мигающий	Режим обновления прошивки

Кроме того, имеется дополнительный зелёный светодиод, расположенный внутри корпуса AVRISP mkII рядом с USB-разъёмом. Он индицирует обмен данными по USB.

## Интерфейс с программируемым устройством

Подключение к программируемому микроконтроллеру реализовано с применением преобразователя уровней и защиты от короткого замыкания.

Вывод 1 разъёма обозначен красной полосой на кабеле. Расположение сигнальных линий на кабеле для подключения к программируемому устройству показано на рис. ниже.



### Преобразователь уровней

AVRISP mkII поддерживает напряжения программируемого устройства от 1.8 В до 5.5 В.

### Защита от короткого замыкания

Защита от короткого замыкания автоматически отключит выводы программируемого устройства от AVRISP mkII, если ток через выводы превысит приблизительно 25 мА.

### Электростатическая защита (ESD)

Выводы программируемого устройства со стороны AVRISP mkII защищены подавителем помех (супрессором) и могут выдержать прямой разряд в 8 кВ.

### Подтягивающие резисторы

Номинал подтягивающих резисторов на MISO/MOSI/SCK, подключаемых к VCC/GND не должен превышать 820 Ом.

### Линия сброса

В качестве одного из этапов проверки правильности подключения кабеля ISP после подачи  $V_{target}$  будет выполняться проверка, является ли напряжение на линии сброса корректным и возможно ли притянуть эту линию к низкому уровню. Если на этой линии нет подтягивающего резистора, т.е. AVRISP mkII обнаруживает на ней 0В, светодиод состояния будет мигать оранжевым светом.

Если подтягивающий резистор на линии сброса имеет слишком большой номинал, то при попытке AVRISP mkII притянуть её к низкому уровню сработает защита от короткого замыкания.

Номинал подтягивающего резистора на линии сброса не должен превышать 4.7 кОм. Номинал конденсатора не должен превышать 10 мкФ.

*Примечание:* обратите внимание, что микроконтроллеры AT89, поддерживаемые AVRISP mkII, имеют противоположную полярность сброса. Если AVRISP mkII использовался с AVR, а затем его подключили к AT89, светодиод состояния может начать мигать оранжевым светом, показывая ошибку на линии сброса. Если в диалоговом окне программирования выбрать AT89 и, например, выполнить команду чтения сигнатуры "Read Signature", светодиод замигает оранжевым светом. Подобная ситуация возникает и в противоположном случае, когда AVR используется после работы с AT89.

## Устранение неисправностей

### Устранение неисправностей

№ п.п.	Проблема	Причина	Решение
1	Невозможно подключиться к AVRISP mkII со стороны ПК.	USB-хаб не может обеспечить достаточное питание.	Для AVRISP mkII необходим активный USB-хаб или USB-хаб с автономным питанием, способный выдать 500 мА.
2	Невозможно подключиться к AVRISP mkII со стороны ПК, при этом светодиод состояния не горит, а по пункту 1 всё правильно.	USB-драйвер установлен некорректно.	Установите USB-драйвер, следуя инструкциям в разделе " <b>Установка USB-драйвера</b> ".
3	Светодиод состояния AVRISP mkII мигает оранжевым светом.	Кабель ISP подключен некорректно.	Проверьте, что красная полоска на кабеле подсоединена к выводу 1 разъёма ISP. Более подробно см. в разделе " <b>Интерфейс с программируемым устройством</b> ".
4	Светодиод состояния AVRISP mkII мигает оранжевым светом, несмотря на правильное подключение кабеля ISP.	Проблема на линии сброса.	Проверьте, имеется ли на линии сброса подтягивающий резистор правильного номинала. Более подробно см. в разделе " <b>Линия сброса</b> ".
5	AVRISP mkII сообщает о коротком замыкании на программируемом МК.	Кабель ISP подключен некорректно, либо некоторые выводы программируемого МК закорочены на землю или VCC, либо они слишком перегружены.	Выполните проверку по пункту 3 и проверку на короткое замыкание. Также проверьте, что номинал подтягивающего резистора на линии сброса не слишком велик. Более подробно см. в разделе " <b>Интерфейс с программируемым устройством</b> ".
6	Не обнаруживается программируемый МК.	Интерфейс SPI на программируемом устройстве отключен из-за того, что не установлен конфигурационный бит SPI, и/или установлен конфигурационный бит RSRDSBL или DWEN.	Если интерфейс ISP отключен установкой конфигурационных битов, то для сброса этих битов понадобится другой интерфейс. Посмотрите об установках конфигурационных битов и интерфейсе программирования в техническом описании на МК. Для высоковольтного параллельного программирования можно использовать STK500, а для JTAG-программирования – JTAGICE mkII.
7	AVRISP mkII обнаруживает программируемый МК, но не может войти в режим программирования, либо при программировании возникает сбой.	Слишком высокая частота ISP.	Понижьте частоту ISP. Частота ISP зависит от тактовой частоты программируемого МК. Более подробно см. в разделе " <b>Board (Параметры платы)</b> ".

### Обновление прошивки вручную

Если прошивка, поставляемая с AVR Studio, более новая, чем та, что загружена в AVRISP mkII, то процесс обновления прошивки обычно проходит автоматически под управлением самой AVR Studio.

Однако если прошивка в AVRISP mkII повреждена, он может перестать реагировать на обновление прошивки. Повреждение прошивки может произойти из-за обрыва связи между ПК и AVRISP mkII во время обновления прошивки, либо если прошивка в AVRISP mkII перепрограммирована неправильным файлом.

Для обновления прошивки AVRISP mkII вручную следуйте приведённой ниже инструкции.

Эта процедура должна сработать во всех случаях:

- 1) Отключите AVRISP mkII от микроконтроллера.
- 2) Отключите кабель USB.

3) Откройте AVRISP mkII, нажав на четыре пластиковые защёлки, соединяющие верхнюю и нижнюю части корпуса.

4) Закоротите контакты 1 и 3 перемычкой рядом с надписью "AVRISP mkII" на печатной плате (см. рисунок ниже).

5) Подключите USB-кабель. Светодиод состояния AVRISP mkII должен замигать красным и оранжевым светом.

6) Запустите приложение "AVRISP mkII Upgrade...", находящееся в меню "Tools" ("Инструменты") в AVR Studio.

7) Нажмите кнопку "Start Upgrading" ("Запустить обновление").

8) Когда обновление завершится, уберите перемычку между выводами 1 и 3 AVRISP mkII.

9) Выключите/включите питания AVRISP mkII путём отключения и последующего включения USB-кабеля.

10) Нажмите кнопку "Done" ("Выполнено").

Теперь прошивка AVRISP mkII обновлена, и он готов к использованию.

