

# TWINPICKS In-System Programmer

## Содержание

|   |           |
|---|-----------|
| Введение .....  | 2         |
| Использование Программатора TwinPicks In-System Programmer.....     | 3         |
| Старт программатора .....   | 3         |
| Пользовательский Интерфейс Программатора TwinPicks: ALTERA .....    | 4         |
| Пользовательский Интерфейс Программатора TwinPicks: ATMEL .....     | 6         |
| Основные Установки .....  | 6         |
| "Program" Установки.....  | 6         |
| "Fuses" Установки.....  | 7         |
| "Lock Bits" Установки .....   | 8         |
| "Advanced" Установки .....  | 9         |
| "Auto" Установки .....  | 10        |
| "Log" Окно .....  | 11        |
| Главное Меню .....  | 12        |
| "File" Меню .....   | 12        |
| "Edit" Меню.....  | 12        |
| "Tools" Меню .....  | 13        |
| "Help" Меню .....   | 13        |
| «Program Options...» Редактор Настроек Программатора TwinPicks..... | 14        |
| "Application" .....   | 14        |
| "AVR Devices" .....   | 15        |
| "Timing" .....  | 17        |
| Особые Предостережения .....  | 19        |
| Fuse Программирование .....   | 19        |
| Использование RESET в качестве General IO port.....                 | 19        |
| AVR Устройства не имеющие ISP Режим Программирования.....           | 19        |
| Устройства без Подстраиваемого RC Генератора.....                   | 19        |
| Known Issues of Latest Release.....                                 | 20        |
| ATtiny2313 Programming Issues.....                                  | 20        |
| ATtiny13 Programming Issues.....                                    | 20        |
| Firmware Upgrade .....  | 20        |
| Обратная Совместимость .....  | 21        |
| <b>Заключение .....</b>   | <b>21</b> |

## Введение

Программатор TwinPicks In-System Programmer (“ПрограмматорTwinPicks”) разработан для конфигурирования Программируемых Логических Интегральных Схем (ПЛИС, PLD - programmable logic devices) фирмы Altera® через адаптер ByteBlasterMV™, подключаемый к параллельному LPT порту персонального компьютера. ПрограмматорTwinPicks поддерживает два режима программирования ПЛИС: Passive Serial (PS) и JTAG™ (Join Test Actions Group). Программатор TwinPicks принимает бинарный формат входных данных *raw binary file (.rbf)* и текстовый формат входных данных *Intel-hex file (.hexout)*. Через порт Passive Serial Программатором TwinPicks возможно программирование только одного устройства ПЛИС, через порт JTAG™ Программатор TwinPicks позволяет программировать как одиночные устройства ПЛИС, так и каскадно-соединенные устройства. Схема каскадного соединения устройств и режимы их программирования указываются в текстовом файле Chain Description File (.cdf), генерируемом программой Quartus® II (Altera).

В Программаторе TwinPicks реализована возможность программирования AVR™ микроконтроллеров фирмы Atmel®. В настоящее время реализован низковольтный режим программирования Low Voltage Serial Programming (LVSP) через последовательный интерфейс SPI. Возможность программирования AVR™ микроконтроллеров через последовательный порт JTAG находится в стадии разработки.

Программатор TwinPicks является результатом C++ реинженеринга программ: MBlaster™ и JRunner™, исходные коды которых доступны на Интернет странице компании Altera®: <http://www.altera.com>. Графический интерфейс пользователя Программатора TwinPicks выполнен в стиле программатора AVRISP™ фирмы Atmel®. Программатор TwinPicks был разработан и протестирован на операционных системах Windows 9x/NT.


Пожалуйста, проверяйте регулярно главную Интернет страницу программатора TwinPicks на предмет новых обновлений. Последняя версия Программатора TwinPicks доступна по адресу: [twinpicks.tripod.com](http://twinpicks.tripod.com).

Этот документ описывает основной принцип работы Программатора TwinPicks.

## Использование Программатора TwinPicks In-System Programmer

Здесь представлены основные сведения о Программаторе TwinPicks, дано описание графического интерфейса пользователя, а также основные сведения об опциях программатора, доступных его пользователям.

### Старт программатора

После инсталляции программатора запустите программу TwinPicks кликом мышки по пиктограмме , расположенной в главном меню Windows: Пуск → Программы → TwinPicks → TwinPicks. Общий вид программатора TwinPicks представлен на рисунке 1. Программатор TwinPicks спроектирован таким образом, чтобы выдавать как можно меньше сообщений в диалоговых окнах и не отвлекать пользователя на кликание мышкой «Да/Нет». Все сообщения пишутся в окно лога, текст которого можно сохранить в автоматическом режиме или принудительно.

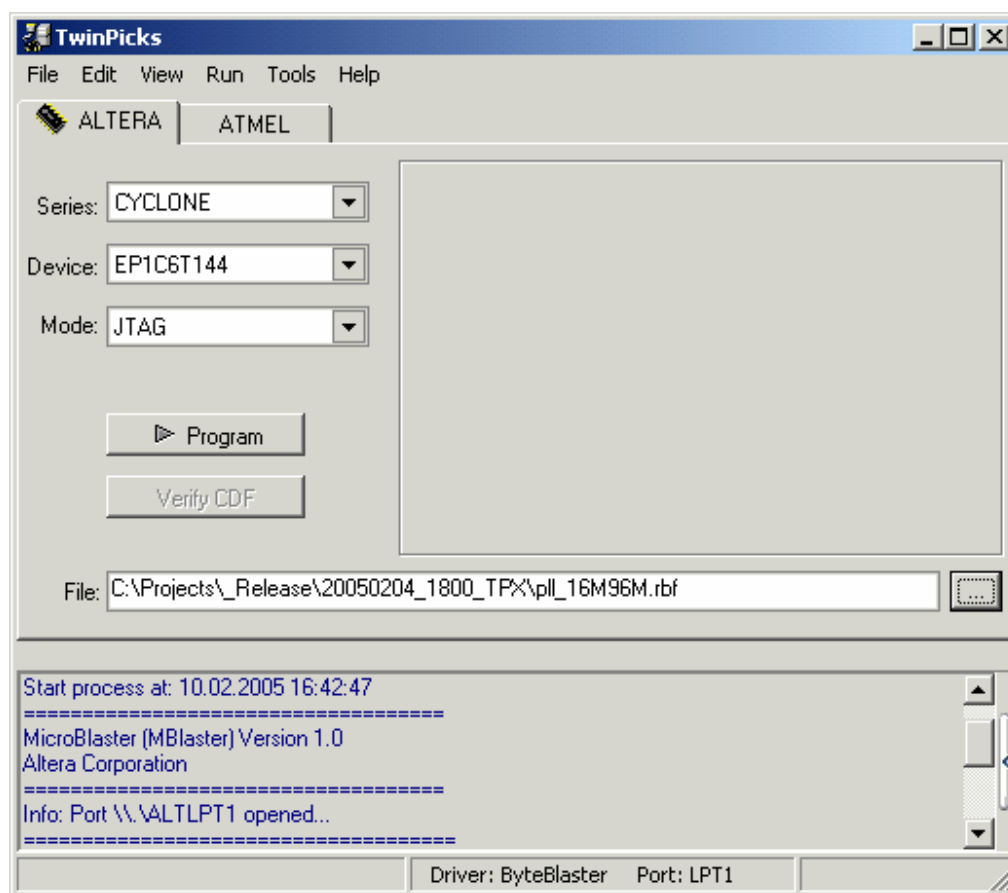


Рисунок 1а. Графический Интерфейс Пользователя: ALTERA TAB

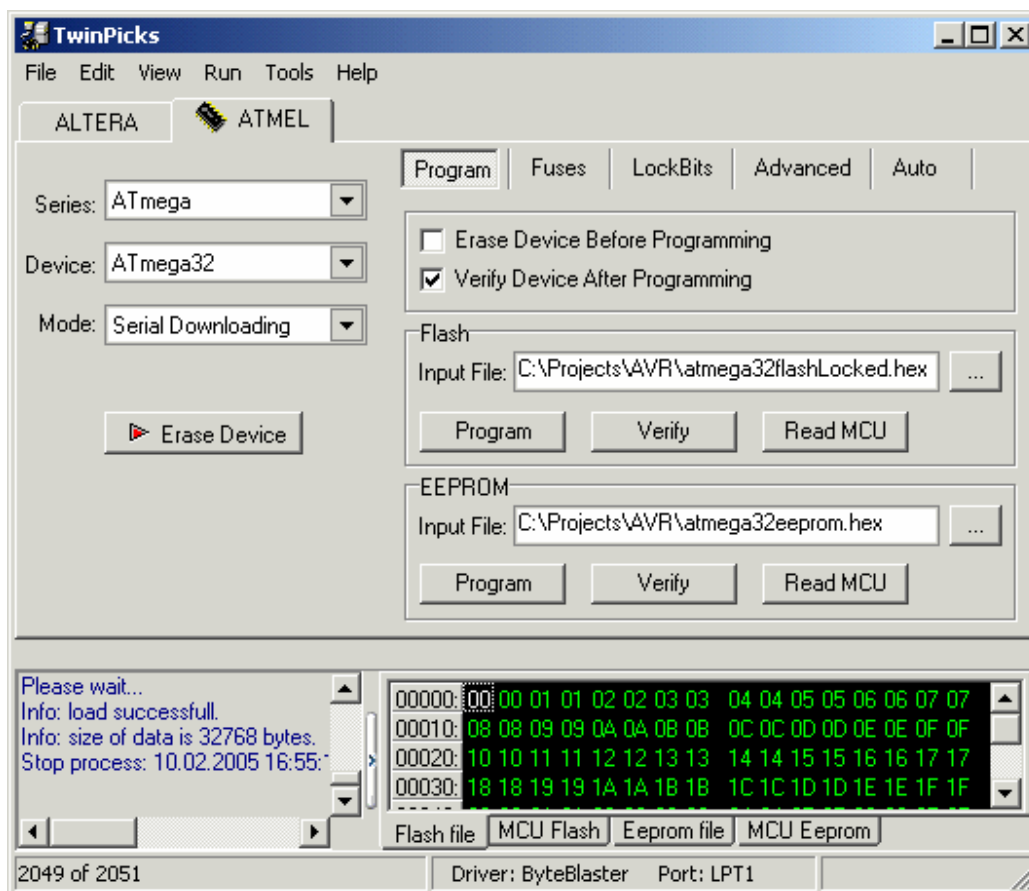


Рисунок 16. Графический Интерфейс Пользователя: ATMEL TAB

### Пользовательский Интерфейс Программатора TwinPicks: ALTERA

Пользовательский интерфейс закладки ALTERA показан на рисунке 1а и включает элементы выбора серии «Series», выбора номенклатурного наименования микросхемы «Device» и указания режима программирования кристалла ПЛИС «Mode». Здесь же представлена кнопка «Program», запускающая процесс конфигурации ПЛИС, и кнопка «Verify CDF», осуществляющая сверку конфигурируемого устройства с описанием схемы каскадного соединения микросхем, представленной файлом\*.cdf.

Редактируемое поле файла исходных данных «File» взаимосвязано с полями «Series», «Device» и «Mode». Если осуществляется программирование одной ПЛИС через порт Passive Serial или JTAG, то в поле «File» необходимо указать полный путь к файлу с расширением \*.rbf или \*.hexout. Пример показан на рисунке 1а.

Если осуществляется программирование каскадно-соединенных микросхем ПЛИС, то в поле «File» необходимо указать файл с расширением \*.cdf. Поскольку файл \*.cdf уже содержит описание типов микросхем и сведения о режимах программирования каждой, включая файлы исходных данных, то указатель «Mode» автоматически переводится в режим JTAG, а указатели «Series» и «Device» обнуляются. Список микросхем, указанных в файле \*.cdf, отображается в правом поле, отмеченном рельефной рамкой, и в дополнение активируется кнопка «Verify CDF». «Галочками» в списке микросхем помечены устройства ПЛИС, которые будут подвергнуты конфигурированию.

Непомеченные микросхемы будут переведены в режим «Bypass mode». Пример показан на рисунке 2.

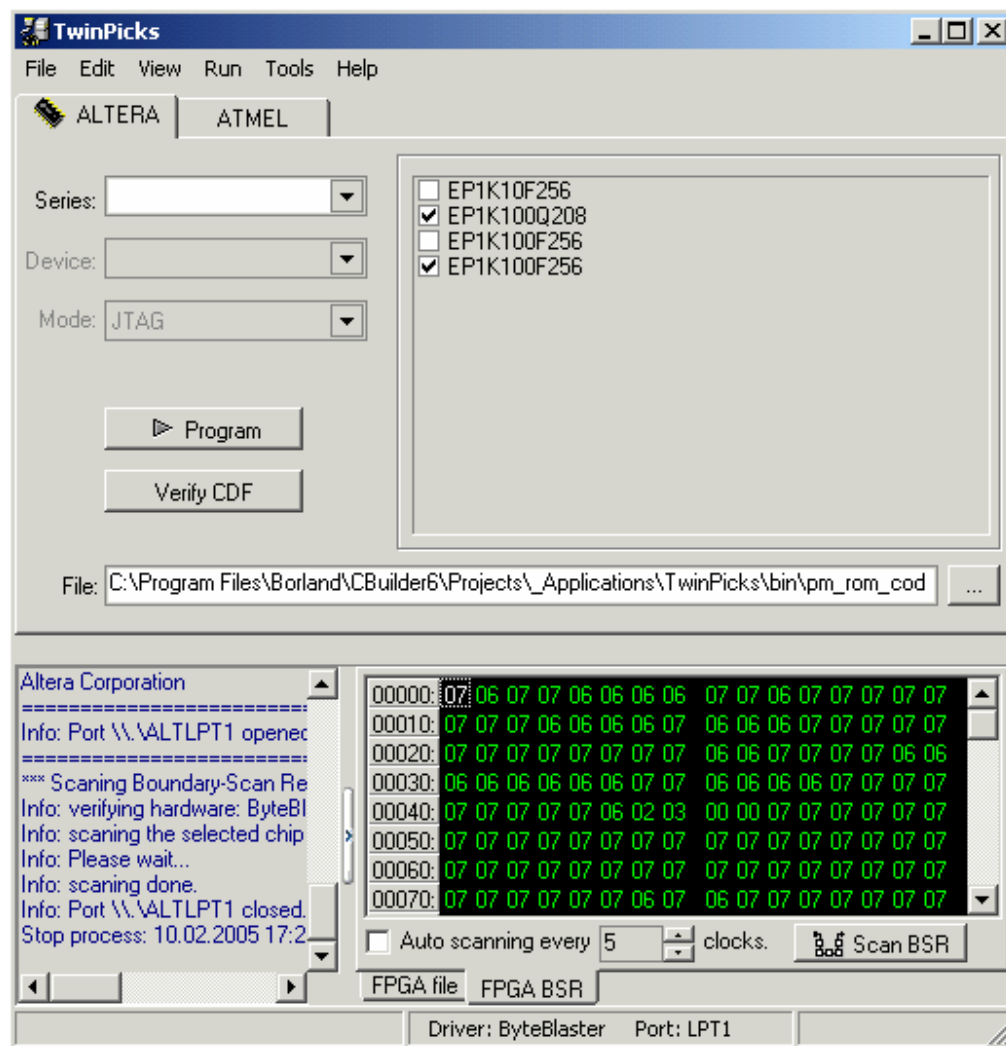


Рисунок 2. Каскадное программирование ПЛИС

На рисунке 2, показано также консольное окно просмотра содержимого Boundary-Scan Register (BSR). Доступ к консольному окну осуществляется по нажатию правой кнопки мышки на всплывающей шторке, расположенной в правой части окна лога. Все данные представлены в шестнадцатеричном формате. Каждая ячейка BSR отображает текущее состояние соответствующего вывода микросхемы или ее внутреннего регистра: направление передачи вход/выход, логическое состояние на входе и выходе ячейки. Для детального ознакомления с Boundary-Scan Registers следует анализировать \*.bsdl файлы, доступные на Интернет странице фирмы Altera, а также обращаться к соответствующей документации по программированию и анализу состояния ПЛИС через интерфейс JTAG. Обновление информации в окне «FPGA BSR» происходит по нажатию кнопки «Scan BSR» или в автоматическом режиме при маркировании «галочкой» поля «Auto scanning every NN clocks», где NN задает период обновления информации в секундах. Для каскадно-соединенных ПЛИС в списке микросхем следует пометить «галочкой» только одну микросхему, регистры которой должны быть просканированы. Если задан в поле

«File» файл данных с расширением \*.rbf или \*.hexout, то его содержимое можно посмотреть на закладке «FPGA file», выбрав команду «Refresh», всплывающего меню правой кнопки мышки.

### Пользовательский Интерфейс Программатора TwinPicks: ATMEL

Пользовательский интерфейс Программатора TwinPicks обладает рядом достаточно эффективных средств для программирования AVR микроконтроллеров в системе (In-System Programming). Профиль настроек микроконтроллера разделен на пять разделов, представленных соответствующими закладками (рисунок 1б и рисунок 3). Различные кристаллы имеют различные возможности по их программированию. Настройки микроконтроллеров, недоступные для программирования, подсвечены серым цветом.

#### Основные Установки

Выпадающий список «Series» предназначен для выбора соответствующего типа микроконтроллера (рисунок 1б). Список «Device» позволяет указать микроконтроллер. Выпадающий список «Mode» позволяет выбрать режим программирования микроконтроллера. Программатор TwinPicks поддерживает только режим низковольтного программирования (ISP low voltage mode). Для новых устройств, не включенных в список поддерживаемых на момент компиляции программы TwinPicks, доступен режим программирования «Try Serial Downloading». Режим программирования «Try Serial Downloading» не гарантирует, что микроконтроллер будет запрограммировано корректно. В зависимости от типа Flash памяти: байт ориентированная или страничная, в режиме «Try Serial Downloading» для микроконтроллера будет выбран алгоритм, аналогичный алгоритму программирования микроконтроллеров AT90S8515 или ATmega8515.

В этой же группе настроек представлена кнопка стирания «Erase Device», нажатие которой приводит к стиранию памяти программ FLASH и памяти данных EEPROM.

#### "Program" Установки

Установки программирования условно разделены на три подгруппы.

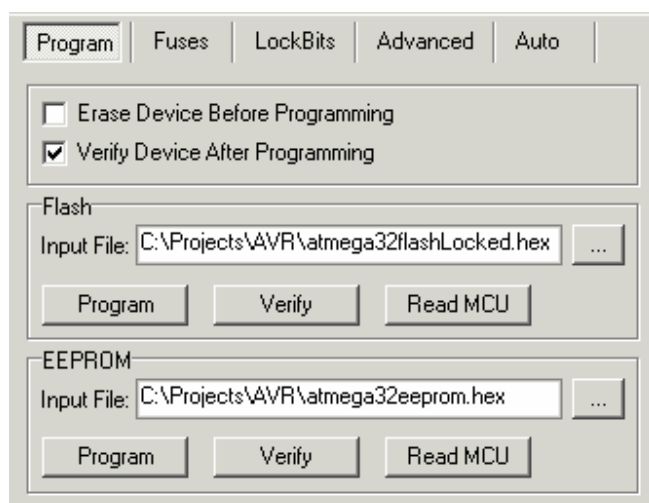



Рисунок 3. Установки раздела «Program»


## Режимы Программирования

Маркирование поля «Erase Device Before Programming» приводит к стиранию памяти микроконтроллера перед выполнением команды программирования памяти программ и памяти данных микроконтроллера. Маркирование поля «Verify Device After Programming» осуществляет автоматическую сверку памяти программ и памяти данных микроконтроллера с исходным файлом данных после выполнения команд программирования.

### Flash

Для установки корректного пути к файлу данных Flash памяти нажмите кнопку обзора  или впишите полный путь к файлу в текстовом поле. Файл исходных данных должен иметь бинарный формат «Raw Binary-rbf» или текстовый «Intel-hex», или формат «extended Intel-hex».








### EEPROM

Для установки корректного пути к файлу данных Eeprom памяти нажмите кнопку обзора  или впишите полный путь к файлу в текстовом поле. Файл исходных данных должен иметь бинарный формат «Raw Binary-rbf» или текстовый «Intel-hex», или формат «extended Intel-hex».

### "Fuses" Установки

На закладке «Fuses» представлен обзор режимов программирования Fuse битов. Некоторые Fuse биты доступны лишь в режиме параллельного или высоковольтного программирования (Parallel /High-Voltage programming). Такие Fuse биты отображаются как недоступные для программирования в режиме ISP. Нажмите кнопку «Read» для считывания текущего состояния Fuse битов и кнопку «Write» для программирования Fuse битов. Маркированное состояние поля Fuse разрешает выбранное состояние Fuse, что означает программирование логических нулей соответствующих Fuse бит. Помните, что на состояние Fuse битов не влияют команды стирания памяти микроконтроллера (т.е. нажатие кнопки «Erase Device» в разделе «Основных» Установок).

Описание битовых полей ( Check Box) и многопозиционных переключателей ( Radio Button)

| Иконка  | Описание   |
|---|--|
|  | Неактивное состояние fuse или lock бит.  |
|  | Активное состояние fuse или lock бит.  |
|  | Неизвестное состояние fuse или lock бита, но бит доступен для программирования. (Установлено активное состояние.)                                  |
|  | Неизвестное состояние fuse или lock бита, но бит доступен для программирования. (Установлено неактивное состояние.)                                |
|  | Чтение битов показывает, что текущее состояние бита неактивное, но lock или fuse бит не доступен для программирования в Serial Downloading режиме. |
|  | Чтение битов показывает, что текущее состояние бита активное, но lock или fuse бит не доступен для программирования в Serial Downloading режиме.   |
|  | Fuse или lock бит недоступен ни для чтения, ни для программирования.   |

Детальное описание доступных для программирования fuse битов в различных режимах программирования доступно в соответствующей документации на микроконтроллер. Маркировка поля "Auto Verify" приводит к автоматической сверке состояния fuse битов микроконтроллера после каждого их программирования.

На рисунке 4 показано градиентное подсвечивание fuse режимов: тип fuse байта «Extended» имеет насыщенный бежевый фон, тип fuse байта «High» имеет светло-бежевый фон, «Low» байт имеет белый фон (на рисунке 4 не показан).

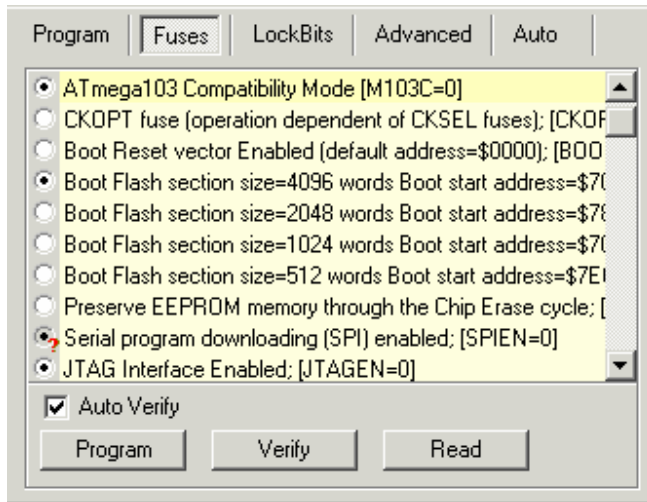


Рисунок 4. Установки Fuses

**ПРЕДУПРЕЖДЕНИЕ:** программирование fuse битов RSTDISBL или SPIEN делает недоступным дальнейшее программирования устройства в режиме ISP-programming.

#### "Lock Bits" Установки

Аналогично «Fuses» установкам, закладка «Lock» показывает доступные режимы защиты устройства от дальнейшего программирования и сверки. Все lock биты доступны в режиме ISP programming. Режим защиты устройства определяется состоянием нескольких lock битов. Программатор TwinPicks учитывает это, и корректная установка lock битов выставляется автоматически для выбранного режима защиты устройства. Если установлен более высокий режим защиты устройства, то программирование более низкого уровня защиты не возможно. Единственным способом установки более низкого уровня защиты устройства является полное стирание памяти устройства командой «Erase Device», что приводит к очистке памяти программ и памяти данных. Есть одно исключение из этого правила: если у программируемого устройства активирован fuse бит «EESAVE», то содержание памяти данных EEPROM не очистится при выполнении команды полного стирания устройства (Erase Device). Маркировка поля «Auto Verify» приводит к автоматической сверке состояния lock битов микроконтроллера после каждого их программирования.



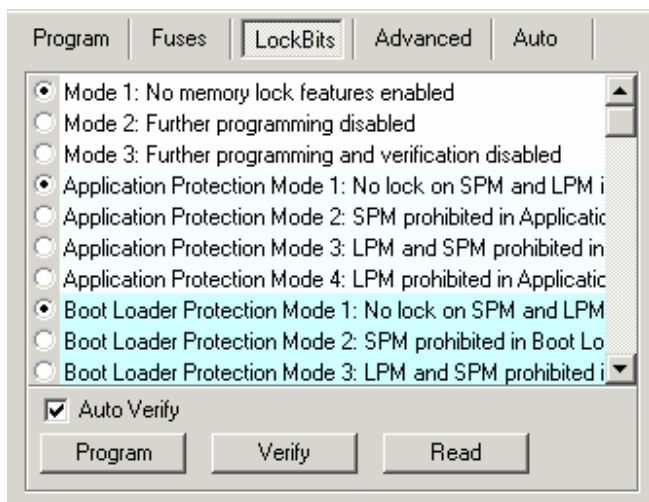


Рисунок 5. Установки Lock

#### "Advanced" Установки

Закладка Advanced представлена двумя подгруппами.

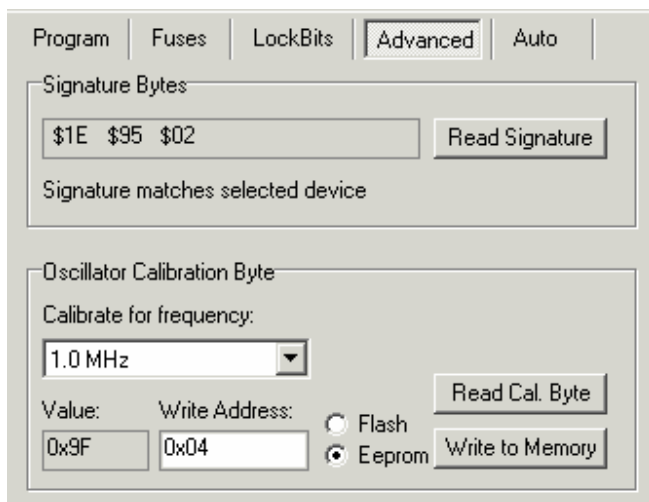


Рисунок 6. Установки Advanced

#### Байты Сигнатуры

Нажатие кнопки «Read Signature», осуществляет считывание батов сигнатуры. Байты сигнатуры играют роль идентификаторов устройства. За информации о байтах сигнатуры обращайтесь к документации программируемого устройства.

#### Калибровочные Байты Внутреннего Генератора

Для устройств, имеющих внутренний калиброванный RC генератор, доступны калибровочные байты-коэффициенты, записанные в устройство на этапе их производства, и которые в свою

очередь не могут быть стерты или изменены пользователем. Калибровочные байты используются для точной подстройки заданной частоты внутреннего RC генератора путем записи их значений в регистр OSCCAL.

#### Чтение Калибровочных Байтов Внутреннего Генератора

Нажатие кнопки «Read Cal. Byte» приводит к считыванию калибровочных байтов. Значение калибровочных байтов для заданной сетки частот отображается в текстовом поле «Value» в шестнадцатеричном формате. Следует отметить, что ряд устройств не предоставляют доступа к калибровочным байтам в режиме выполнения микроконтроллером программы, поэтому значения калибровочных байтов, если их планируется в дальнейшем использовать, должны быть предварительно сохранены в памяти микроконтроллера. Подсветка данного раздела серым цветом означает, что устройство не имеет внутреннего RC генератора. Устройство, имеющее автоподстройку RC генератора, не нуждается в управлении калибровочными байтами (для более детального ознакомления с данным вопросом обращайтесь к соответствующей документации).

#### Запись Калибровочных Байтов

Если калибровочные байты не доступны в режиме работы программы микроконтроллера и микроконтроллер не имеет автоподстройку RC генератора, то значение калибровочного байта должно быть предварительно сохранено в любое определенное Вами место памяти программ Flash или памяти данных EEPROM. Для этого укажите шестнадцатеричный адрес регистра памяти в текстовом поле «Write Address» и нажмите кнопку «Write to Memory». Значение калибровочного байта будет записано во «Flash» или «Eeprom» в зависимости от состояния двухпозиционного переключателя.

#### "Auto" Установки

Для программирования одностипных устройств закладка Auto предоставляет удобное средство автоматического исполнения серии команд. Порядок исполнения команд соответствует порядку их отображения в списке. Только помеченные маркером команды исполняются. Например, если помечена только команда «Program FLASH», то нажатие кнопки «Start» запишет в память программ FLASH данные из HEX (или RBF) файла, путь к которому указан в разделе «Program». Только поддерживаемые устройством команды могут быть исполнены. Для сохранения сообщений, отображаемых в окне лога, необходимо пометить поле «Log to file».

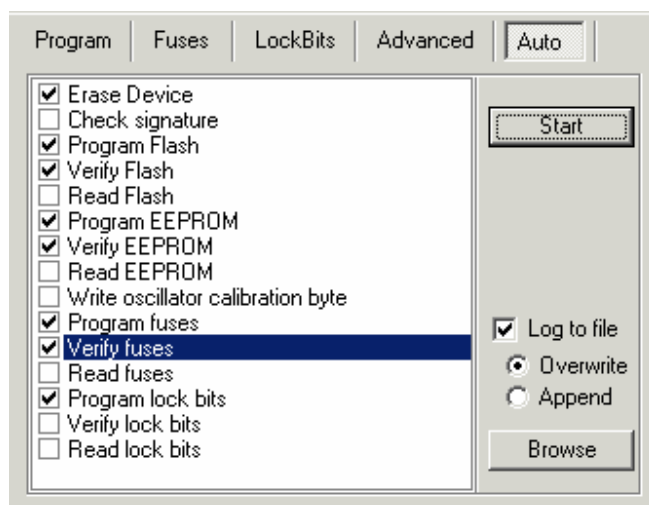


Рисунок 7. Установки Auto

## "Log" Окно

Окно Лога отображает сообщения, посылаемые программатором пользовательскому интерфейсу. Сохранение истории сообщений программы в файл производится автоматически по окончании работы с программатором TwinPicks если поле «Log to file» маркировано на закладке «Auto» или в соответствующем разделе настройки программы (описание настройки программы приведено ниже).

Аналогично интерфейсу пользователя раздела ALTERA справа от окна лога есть кнопка-шторка, открывающая консоль контроля исходных данных (рисунок 8).

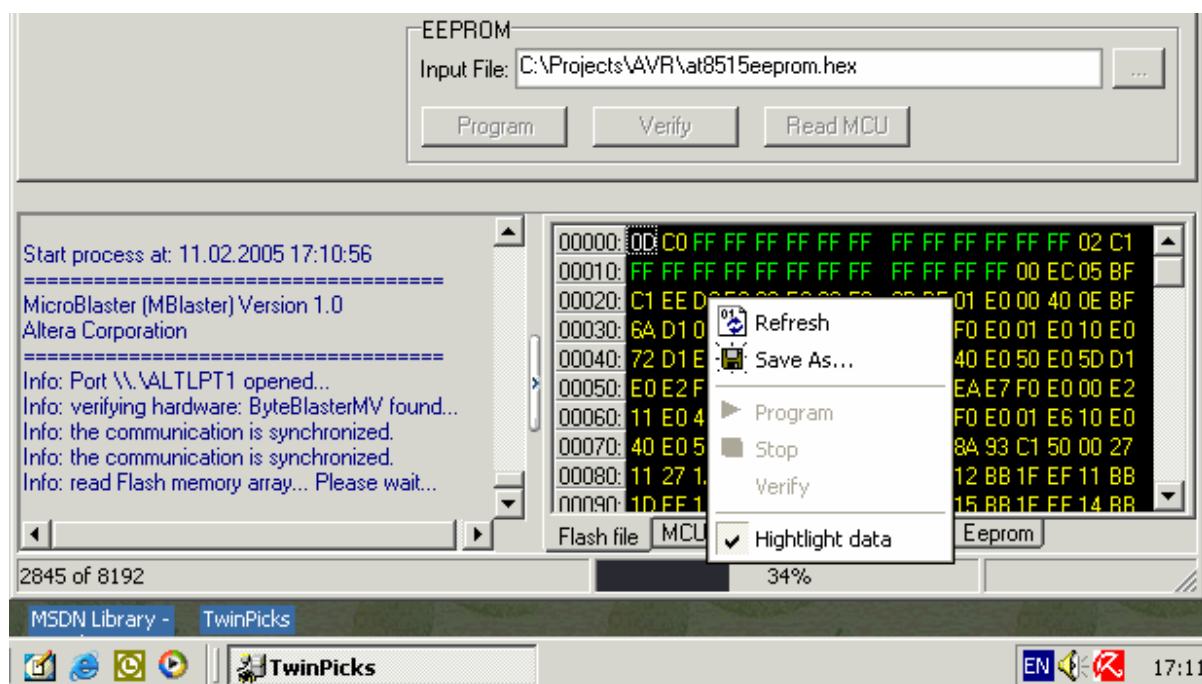


Рисунок 8. Консоль контроля исходных данных

Всплывающее pop-up меню закладки «Flash file» позволяет обновить отображаемые данные выбором команды «Refresh», сохранить данные в отдельный файл с расширением \*.hex или \*.rbf. Команда меню «Highlight data» включает цветовую маркировку различий между исходным файлом данных и данными, считанными из Flash памяти микроконтроллера в консоль «MCU Flash». Зеленый цвет свидетельствует об отсутствии различий в данных, желтый — о не соответствии данных Flash памяти микроконтроллера и исходного файла, красным цветом отображаются «лишние» данные, возникающие при разных размерах файла данных и Flash памяти микроконтроллера. Все выше сказанное справедливо и для консоли Eeprom.

## Главное Меню

Большинство разделов главного меню программы TwinPicks дублирует аналогичные разделы всплывающих меню правой кнопки мышки и кнопок пользовательского интерфейса, поэтому в данном параграфе будут описаны лишь те команды, которые не включены в другие разделы.

### "File" Меню

Раздел меню «File» помимо стандартной функции закрытия программы «Exit» и принудительного сохранения Лога программы «Save Log As...» содержит функции сохранения «Save Atmel Settings As...» и загрузки «Load Atmel Settings...» профиля установок микроконтроллера. Команды сохранения и загрузки профиля настроек микроконтроллера доступны только при активации закладки «ATMEL» и только после загрузки параметров микроконтроллеров из XML файлов директории «Partdescriptionfiles».

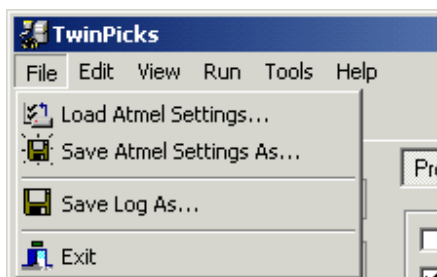


Рисунок 9. Меню «File»

### "Edit" Меню

Раздел «Edit» содержит ряд стандартных функций редактирования текста, а также важную команду вызова редактора настроек программы «Program Options...». Описание редактора настроек Программатора TwinPicks дано в следующем разделе этого руководства.

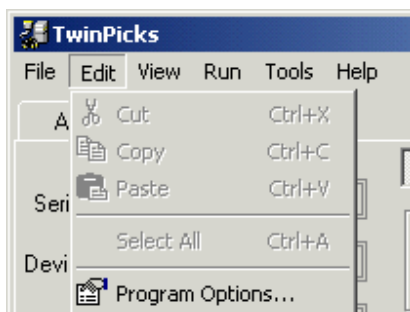


Рисунок 10. Меню «Edit»

### "Tools" Меню

Раздел меню «Tools» содержит команду «Load Atmel Files From...» и позволяет обновить список микроконтроллеров, указав в диалоге открытия файлов директорию, содержащую XML файлы устройств. Данная команда дублирует последнюю строчку «Load from disk» списка типов устройств «Series». Из выбранной директории загружаются все XML файлы, даже, если выбран только один файл.

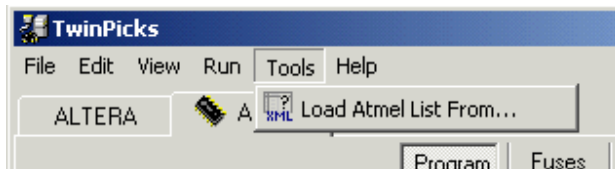


Рисунок 11. Меню «Tools»

### "Help" Меню

Меню «Help» в настоящее время содержит единственную команду «About...», вызов которой отрывает модальную форму, содержащую основные сведения о программе: наименование, номер версии, контактную информацию и список зарегистрированных товарных марок (рисунок 12).

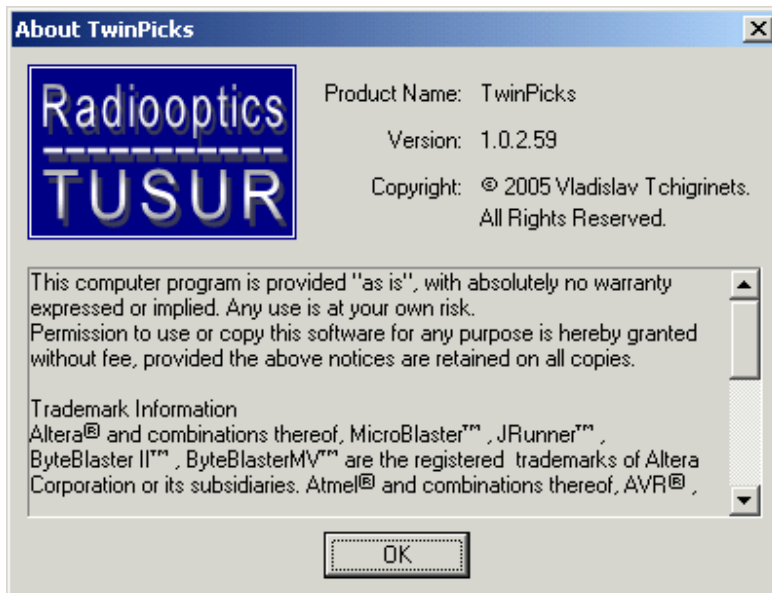
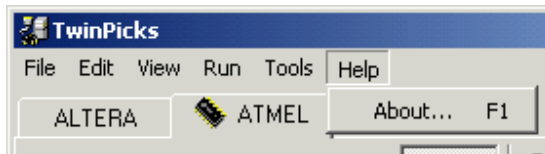


Рисунок 12. Меню «Help» и форма «About TwinPicks»

## «Program Options...» Редактор Настроек Программатора TwinPicks

Внимательно прочитайте этот раздел, чтобы избежать ошибок при работе с Программатором TwinPicks и повысить производительность его работы. Редактор настроек структурно разделен на три раздела, каждый из которых представлен своей закладкой. Каждый пользователь рабочей станции имеет собственные настройки программы, не влияющие на настройки программы других пользователей рабочей станции.

### "Application"

Закладка «Application» содержит общие настройки программатора, представленные в двух подразделах.

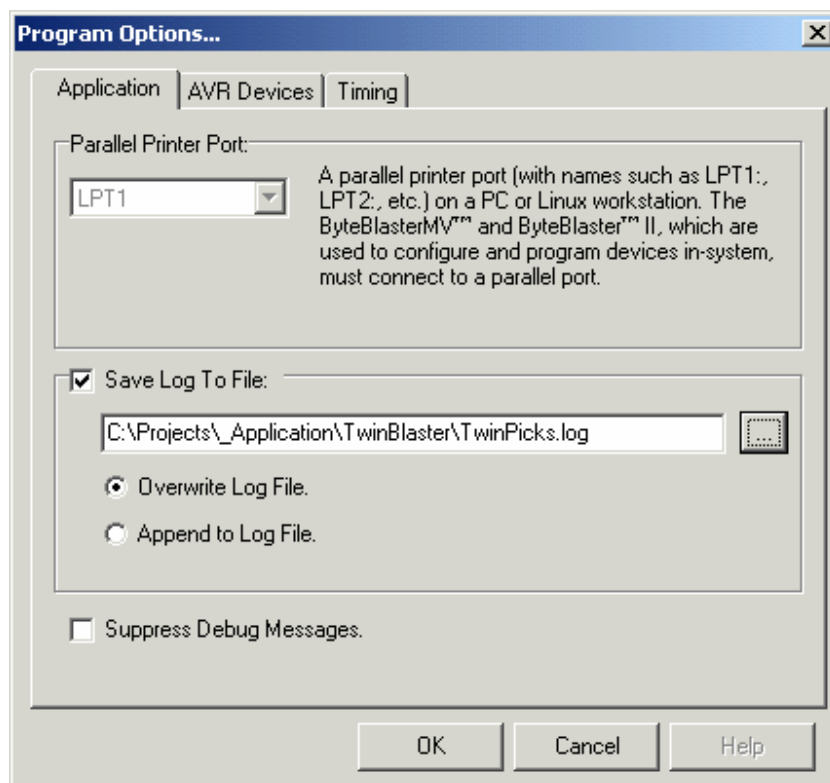


Рисунок 13. Редактор настроек Программатора TwinPicks. Общие Настройки Программы

### Parallel Print Port

На данный момент этот подраздел зарезервирован, но не реализован. В NT подобных операционных системах Программатор TwinPicks производит обмен данными с портом LPT1 через ByteBlaster драйвер Altera, устанавливаемый отдельно. При полной установке Программатора TwinPicks драйвер Altera ByteBlaster не устанавливается автоматически, а лишь копируется в папку «drivers» директории TwinPicks. Инструкцию по установке драйвера Altera ByteBlaster следует искать на главной странице <http://twinpicks.tripod.com> или на Интернет странице компании [Altera](#). Если на компьютере установлен Quartus II версии 4.2, то в установке драйвера Altera ByteBlaster нет необходимости — его функции берет на себя Sentinel System Driver компании Rainbow Technologies. Для операционных систем Windows 9x/Me обмен данными производится по базовому адресу LPT порта 0x378 напрямую и драйвер Altera ByteBlaster не нужен.

#### Save Log To File

Пометка поля «Save Log To File» делает доступными текстовое поле полного пути к Log файлу и выбор режимов сохранения файла: перезаписать Log файл новыми данными «Overwrite Log File» или добавить новые сообщения в конец Log файла «Append Log File». Сохранение Log файла производится в момент закрытия Программы TwinPicks или выбором соответствующей команды меню. Если пользователь рабочей станции не обладает правами администратора, то следует указать полный путь к файлу сохраняемого Log-а отличный от директории TwinPicks, либо пользователь должен иметь права на создание модификацию файлов директории TwinPicks. Автор настоятельно рекомендует активировать сохранение Log файлов, чтобы иметь возможность, в случае необходимости, проанализировать результаты работы Программатора TwinPicks.

#### Suppress Debug Messages

Активация поля «Suppress Debug Messages» подавляет сообщения, имеющие тип «Debug: ...». Это позволяет значительно сократить размер Log файла, однако затрудняет анализ работы программатора. Рекомендуется активировать данное поле только после длительного тестирования Программатора TwinPicks с выбранными типами устройств.

#### "AVR Devices"

Закладка «AVR Devices» определяет поведение Интерфейса Пользователя раздела «ATMEL» Программатора TwinPicks.

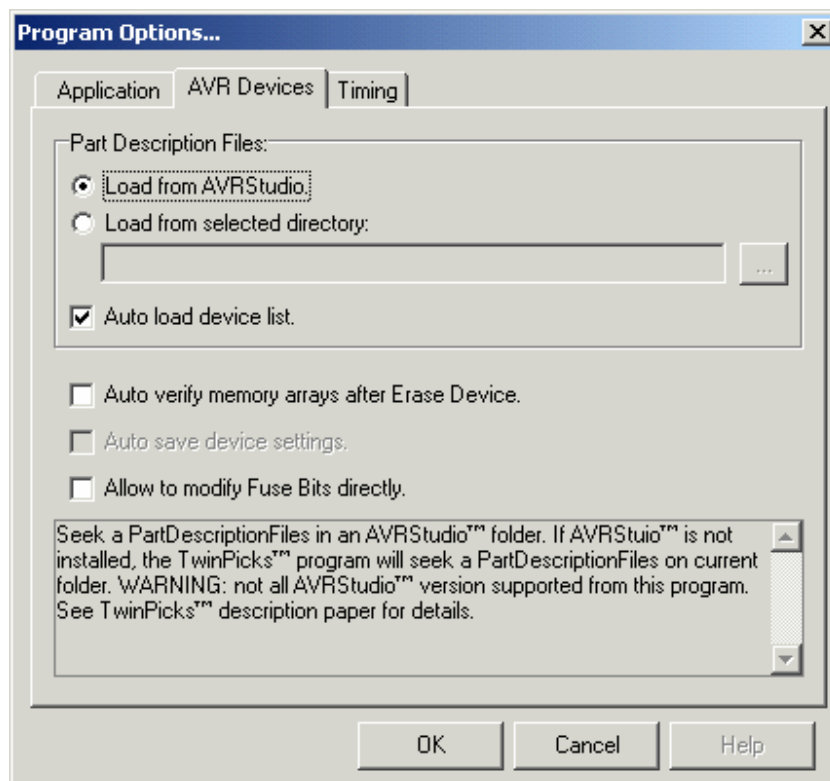


Рисунок 14. Установки раздела ATMEL Программатора TwinPicks

## Part Description Files

Подраздел «Part Description Files» указывает директорию-источник, откуда загружаются XML файлы, содержащие параметры устройств. Если на рабочей станции установлена программа AVRStudio версии 4.11, то при положении двухпозиционного переключателя «Load from AVRStudio» Программатор TwinPicks будет читать XML файлы из AVRStudio. Более ранние версии AVRStudio имеют устаревший формат данных XML, что является причиной неполной загрузки параметров микроконтроллеров, например: отсутствие Fuse и Lock битов. Положение двухпозиционного переключателя «Load from selected directory:» позволяет явно указать путь к XML файлам. Рекомендуется самостоятельно создать отдельную директорию, в которую скопировать только те XML файлы, с которыми в дальнейшем предполагается работать. Это позволит значительно ускорить загрузку списка поддерживаемых устройств. Путь к созданной директории следует прописать в текстовом поле «Load from selected directory:». Если пометить галочкой поле «Auto load device list», то при первом обращении к разделу «ATMEL» основного окна программы TwinPicks программатор произведет автоматическую загрузку списка поддерживаемых устройств из указанной директории.

## Auto verify memory arrays after Erase Device

Активация поля «Auto verify memory arrays after Erase Device» приводит к автоматической проверке памяти программ «Flash» и памяти данных «Eeprom» после выполнения команды «Erase Device». Если поле «Auto verify memory arrays after Erase Device» не маркировано галочкой, то проверку кристалла на чистоту можно осуществить командой «Blank Check», расположенной в меню: Run -> MCU Program -> Blank Check.

## Auto save device settings

Поле «Auto save device settings» зарезервировано и не используется.

## Allow to modify Fuse Bits directly

По возможности не активизируйте поле «Allow to modify Fuse Bits directly». При маркировании галочкой данного поля в окне комментария появляется предупреждение:

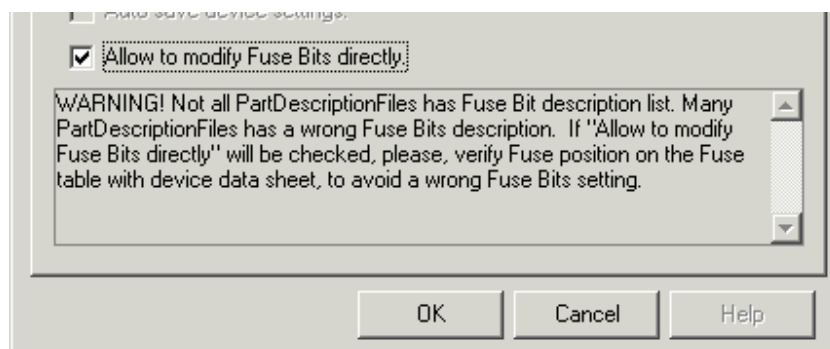


Рисунок 15. Прямой доступ к Fuse Bits

По умолчанию окно «Fuse» основной программы отображается список режимов программирования Fuse битов представленный в виде многопозиционного переключателя (☑ Radio Buttons). Как правило, за каждый из режимов отвечает несколько Fuse бит. При маркировании галочкой поля «Allow to modify Fuse Bits directly» становятся доступными сами Fuse биты, которые отображаются в виде битовых полей (☑ Check Box). Изменение многопозиционного переключателя отражается



на битовых полях, также как и сброс и установка битовых полей изменяет положение многопозиционного переключателя. Проблема заключается в том, что, во-первых, не все возможные установки флагов разрешены для использования, и, во-вторых, ряд XML файлов содержат неполное описание Fuse битов (например, в ATmega32.xml отсутствует описание битов SKOPT, SUT0 и SUT1), а в ATtiny2313 некоторые Fuse биты перепутаны местами, что может быть причиной неправильной установки Fuses. При программировании Fuses Программатор TwinPicks считывает только состояние многопозиционных переключателей. Дополнительно, текущие значения Fuse Bytes можно проконтролировать в окне Log-а основной программы. По причине выше сказанного, старайтесь избегать прямого редактирования Fuse битов путем установки и снятия битовых полей. Неверное описание некоторых полей не является ошибкой программы TwinPicks, скорее — это недоработка программистов ATMEL, т.к. Программатор TwinPicks отображает только те данные, которые содержатся в XML файлах.

#### "Timing"

Раздел «Timing» позволяет задать временные параметры для программирования Atmel устройств.

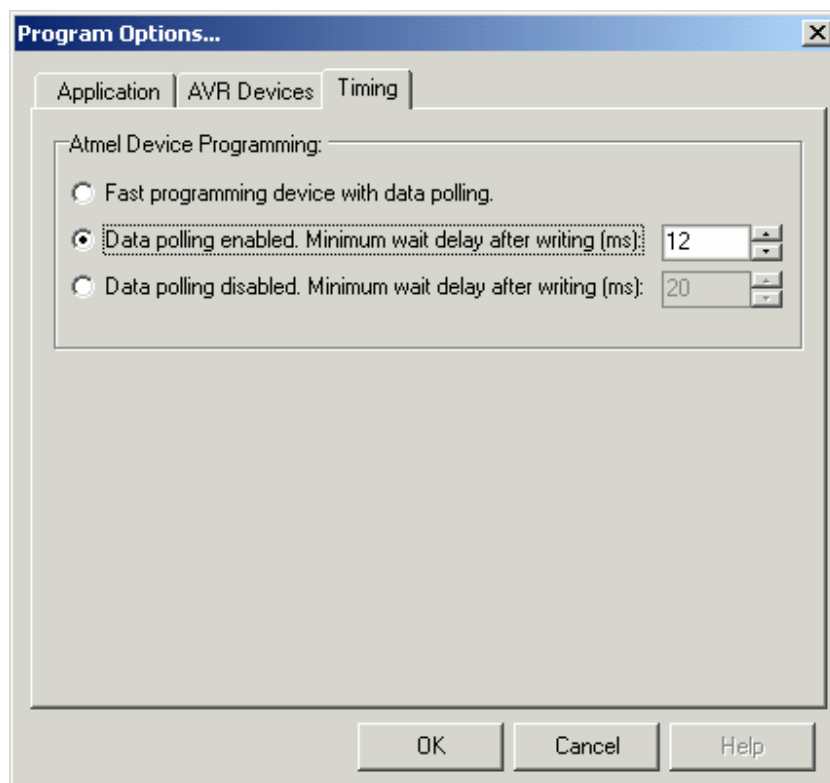


Рисунок 16. Прямой доступ к Fuse Bits

По умолчанию многопозиционный переключатель подраздела «Atmel Device Programming» находится в состоянии «Fast programming device with data polling». Это означает то, что запись следующего байта или страницы памяти программ (Flash) будет производится только после правильного считывания данных, записанных в предыдущий регистр или предыдущую страницу памяти. Это позволяет с точностью до миллисекунды рассчитать время записи  $t_{wd\_prog}$ . Для таких устройств, как ATtiny2313 контроль правильности записи (data polling) осуществляется специальной командой «Poll RDY/BSY», которая Программатором TwinPicks не поддерживается. Если контроль записи данных необходим, то необходимо установить многопозиционный

переключатель в положение «Data polling enabled. Minimum wait delay after writing (ms)» и задать минимальное время  $t_{wd\_prog}$  ожидания в миллисекундах после операции программирования. Если контроль процесса записи не обязателен, то его можно отключить, активировав поле «Data polling disabled. Minimum wait delay after writing (ms)» и также задать минимальное время  $t_{wd\_prog}$  ожидания в миллисекундах после операции программирования. Проконтролировать запись данных можно командой «Verify».

## **Особые Предостережения**

Существует ряд предостережений по использованию Программатора TwinPicks Software в качестве In-System программатора AVR устройств, которые должны быть отмечены в данном руководстве.

### **Fuse Программирование**

Некоторые устройства имеют fuse биты, недоступные для программирования в режиме ISP. Для их программирования необходимо использовать программатор, поддерживающий параллельный режим программирования. Некоторые AVR устройства предоставляют доступ к SPIEN и RSTDISBL fuses. Перепрограммирование этих fuses ведет к отключению ISP programming.

### **Использование RESET в качестве General IO port.**

Если вывод RESET используется в качестве general purpose IO, режим In-System Programming становится недоступным. По этой причине вывод RESET должен быть подтянут к +12V для активации высоковольтного последовательного программирования (High Voltage Serial) или параллельного режима программирования: Parallel Programming mode (HVSP or HVP). HVSP или HVP необходимо использовать для изменения состояния RSTDISBL fuse.

### **AVR Устройства не имеющие ISP Режим Программирования**

Некоторые устройства не имеют режима программирования ISP programming (к ним относится ATtiny28). Для программирования таких устройств необходим параллельный программатор, например, STK500 Starter Kit. Только устройства, поддерживающие низковольтное программирование через последовательный интерфейс (low voltage ISP programming), могут быть сконфигурированы Программатором TwinPicks Software.

### **Устройства без Подстраиваемого RC Генератора**

Не все устройства, имеющие с внутренний RC генератор, позволяют подстраивать тактовую частоту RC генератора. Для таких устройств внутренний RC генератор будет работать на частоте, определенной для него по умолчанию. Значение тактовой частоты определено в соответствующей документации.

## Known Issues of Latest Release

### ATtiny2313 Programming Issues

Please note that the Rev. A of ATtiny2313 has an error in the parallel programming interface thus this is not supported.

CAUTION: Избегайте установки fuse бита, запрещающего ISP интерфейс, т.к. это может привести в блокированию микроконтроллера и непригодности для его дальнейшего программирования.

---

### ATtiny13 Programming Issues

Please note that the ATtiny13 E.S. (Engineering Sample) has an errata concerning special combinations of fuse bits causing the device to lock for further programming. The following combinations of settings/fuse bits will cause this effect:

| Oscillator Setting                   | Start-up Time Setting      | Debugwire and Reset Setting                                      |
|--------------------------------------|----------------------------|--|
| 128KHz internal<br>CKSEL[1..0] = 11  | Shortest<br>SUT[1..0] = 00 | Debugwire = Enabled ( DWEN=0) OR Reset = Disabled (RSTDISBL = 0) |
| 9.6 MHz internal<br>CKSEL[1..0] = 10 | Shortest<br>SUT[1..0] = 00 | Debugwire = Enabled ( DWEN=0) OR Reset = Disabled (RSTDISBL = 0) |
| 4.8 MHz internal<br>CKSEL[1..0] = 01 | Shortest<br>SUT[1..0] = 00 | Debugwire = Enabled ( DWEN=0) OR Reset = Disabled (RSTDISBL = 0) |

### Problem fix/ Workaround

**Избегайте** выше приведенных fuse комбинаций. Выбор более длительного start-up time позволит избежать проблемы.

---

## Firmware Upgrade

In order to achieve a successful firmware upgrade, make sure that the following conditions are met:

- The target voltage must be stable and at least 2.7V.
  - No target hardware must drive the MISO, MOSI or SCK pins on the ISP connection.
-

## Обратная Совместимость

- Программатор TwinPicks Software совместим с версией AVR Studio 4.11. Если на компьютере пользователя установлена более ранняя версия AVR Studio, то для корректного программирования AVR устройств в настройках Программатора TwinPicks необходимо указать путь к XML файлам в директории "Partdescriptionfiles", устанавливаемым вместе с Программатором TwinPicks Software.

## Заключение

Программатор TwinPicks Software является простым в обращении и недорогим в аппаратной реализации устройством, позволяющим конфигурировать устройства программируемой логики Altera PLDs в режимах Serial Programming and JTAG, а также программировать AVR микроконтроллеры фирмы Atmel через последовательный интерфейс SPI Serial Downloading.

~~~~~

Чигринец Владислав  
Лаб. «Радиооптики» НИЧ Томского Университета Систем Управления и Радиоэлектроники,  
Томск, Россия, 634050.