

## Программатор AVR – аналог AVRISP



- Полный аналог AVRISP
- Работает под управлением AVR Studio
- Программное обеспечение от Atmel
- Может работать через конвертер USB-COM
- Внутрисхемное программирование
- Программирование в программаторе
- Напряжение питания ..... 8 – 12 в

Благодаря оптимальному соотношению цена/качество микроконтроллеры семейства AVR фирмы Atmel пользуются заслуженной популярностью как среди радиолюбителей, так и профессионалов. Возможность их программирования по последовательному интерфейсу SPI позволяет использовать для этих целей очень простые программаторы, например PonyProg [1].

Такой программатор подключается к LPT или COM порту компьютера и, по сути, содержит только буферные элементы для согласования уровней выводов порта с контроллером. Все необходимые алгоритмы реализуются управляющей программой, запускаемой на компьютере. Во всех деталях работа с программаторами такого типа описана в [2]. Казалось бы, это самый простой и в то же время наилучший вариант. Зачем усложнять задачу? К сожалению, все не так просто.

Основное назначение LPT порта – это работа с принтером, а COM порта – обеспечение связи с

внешними устройствами, например модемом по интерфейсу RS-232. Программное управление уровнями на отдельных линиях этих портов хотя и возможно, но это уже нестандартная задача, а значит за то, что она будет выполняться корректно, изготовители материнских плат ответственности не несут. Да и не далек тот день, когда эти порты вообще исчезнут из компьютера, а работать через конвертер USB-COM простейшие программаторы не могут.

Более того, стандартная аппаратно-программная конфигурация персонального компьютера даже при наличии быстродействующих процессора и ОЗУ не позволяет точно сформировать интервалы времени, меньше нескольких сотен миллисекунд. Для программирования же современных контроллеров необходимы импульсы длительностью порядка единиц микросекунд и менее. Поэтому невозможно гарантировать корректную работу простых программаторов на всех без исключения компьютерах.

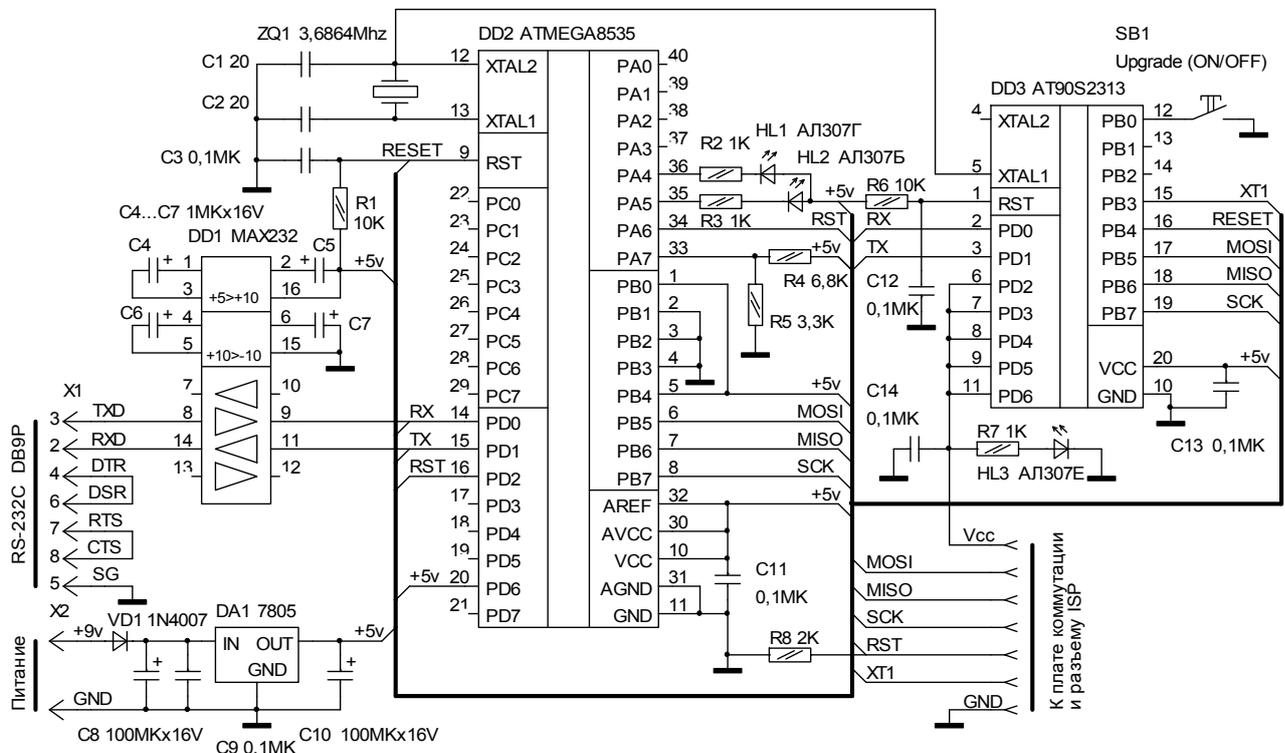


Рис.1 Принципиальная схема основной платы

Есть и еще одна важная сторона вопроса программирования. В каталог Atmel 1997г., посвященный AVR было включено всего 4 контроллера этого класса. В 2002г. их было уже 18, а в 2006 – 48! Энтузиасты – программисты share ware программаторов просто не в силах своевременно вводить корректную поддержку всех новых типов, ведь для этого как минимум надо иметь хотя бы один экземпляр каждого типа и достаточное количество свободного времени. Даже те радиолюбители, которых вполне устраивают старые типы контроллеров, могут оказаться в сложной ситуации. Например, класс AT90S полностью снят с производства. На смену ему пришли ATtiny и ATmega, которые имеют несколько иные алгоритмы программирования. В такой ситуации логично будет не изобретать велосипед, а воспользоваться разработками производителя AVR.

Предлагаемый программатор – это упрощенный и модернизированный аналог выпускаемого и поддерживаемого компанией Atmel внутрисхемного программатора AVRISP. Упрощение заключается в исключении из схемы элементов защиты от статического электричества – в серьезных случаях они все равно не помогут – и замене преобразователей уровня COM-TTL на дискретных элементах специализированной микросхемой MAX232.

В отличие от прототипа программатор позволяет программировать контроллеры не только внутрисхемно, но также и непосредственно в программаторе. Ведь иногда нужно просто запрограммировать контроллер, например, по просьбе друга. Программатор поддерживает все микроконтроллеры AVR, имеющие функцию внутрисхемного про-

граммирования. Управление осуществляется через COM-порт персонального компьютера. Возможно подключение и к USB через конвертер, например на FT232BM, описание которого можно найти в [3] или на моем сайте. Управляющая программа является составной частью фирменной интегрированной отладочной среды разработки AVR Studio от Atmel. Ее последняя версия всегда свободно доступна по адресу <http://www.atmel.com>.

Принципиальная схема основной платы программатора показана на рис. 1. Собственно программатор выполнен на DD2 типа ATmega8535. Светодиоды HL1 красного цвета и HL2 зеленого индицируют режим программирования и готовности соответственно. DD3 предназначен для записи новых версий прошивок в основной контроллер, а также для управления напряжением питания программируемого контроллера.

Прототип от Atmel не имеет собственного источника питания т.к. он предназначен только для внутрисхемного программирования, поэтому питание на него подается от пользовательского устройства. В данной конструкции питание на программируемый контроллер может быть подано от программатора, в связи с чем появляется необходимость его отключения на момент установки контроллера в панельку. В базовом программном обеспечении такая функция не предусмотрена, вмешиваться же в фирменную прошивку DD2 нецелесообразно, т.к. она постоянно обновляется. А вот прошивка загрузчика DD3 неизменна, поэтому она и была несколько доработана.

В рабочем режиме этот контроллер не выполняет никаких функций, поэтому его вполне можно

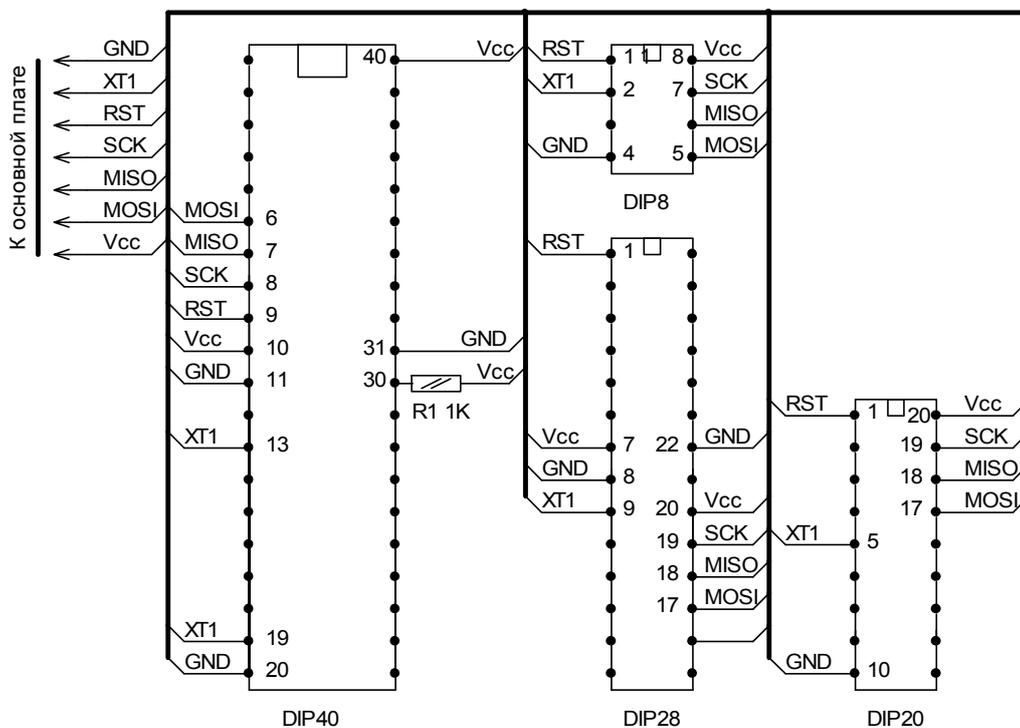


Рис.2 Принципиальная схема платы коммутации

использовать для целей коммутации Vcc. Для повышения нагрузочной способности выводы PD2...PD6 соединены параллельно. При нажатии на кнопку SB1 на этих выводах появляется напряжение 5 В, о чем сигнализирует свечение HL3 желтого цвета. Но это еще не все. При программировании контроллера на него должна подаваться тактовая частота. В готовом устройстве для этой цели чаще всего используется кварцевый резонатор. В данном случае тактовая частота 1,8 МГц формируется на выводе PB3 DD3, с которого и подается на программируемый контроллер. При повторном нажатии SB1 выводы PD2-PD6 и PB3 переходят в высокоимпедансное состояние, т.е. напряжение питания и тактовая частота с программируемого контроллера снимаются.

Таким образом, перед установкой контроллера в панельку следует нажать SB1, отключая питание, после установки нужно вновь нажать SB1. Свечение HL3 будет индцировать подачу питания и готовность к работе. После успешного программирования перед извлечением контроллера из панельки питание с него вновь снимают, нажимая SB1. При внутрисхемном программировании эта кнопка не используется.

Благодаря наличию резистора R8 на линии сброса RST в любом случае поддерживается уровень, близкий к нулю, что исключает неконтролируемый запуск программы. На работу в режиме программирования этот резистор влияния не оказывает.

Плата коммутации (рис. 2) предназначена для программирования контроллеров в DIP корпусах. Контроллеры в других корпусах могут программироваться внутрисхемно или с помощью переходников, подключаемых к выведенным также и на отдельный разъем линиям связи основной платы и платы коммутации. В качестве этого разъема удобно использовать СШ-7, широко применявшийся в бытовой аппаратуре конца прошлого века.

Размер основной платы (рис. 3) 76x76 мм, а платы коммутации (рис. 4) 95x80 мм. Они изготовлены из одностороннего фольгированного стеклотекстолита и оптимизированы для метода «утюжной» технологии. Все резисторы и неполярные конденсаторы в SMD корпусах, микросхемы – в DIP. В качестве DD3 можно использовать как устаревший AT90S2313, так и новый ATtiny2313. Причем первый даже более предпочтителен по причинам, которые будут рассмотрены ниже. В качестве DD2, очевидно, можно использовать не только ATmega8535, но и AT90S8535, правда такой вариант не тестировался.

Программатор выполнен в корпусе, спаянном из фольгированного гетинакса. Плата коммутации является верхней крышкой, в ней сделаны по месту не показанные на чертеже отверстия для SB1 и HL1-HL3. Разъем СШ-7 для внутрисхемного программирования размещается на боковой стенке. Кабель связи с компьютером может иметь длину 1...1,5 м. Питание осуществляется от любого сетевого адаптера с напряжением 9...15 В и током до 100 МА.

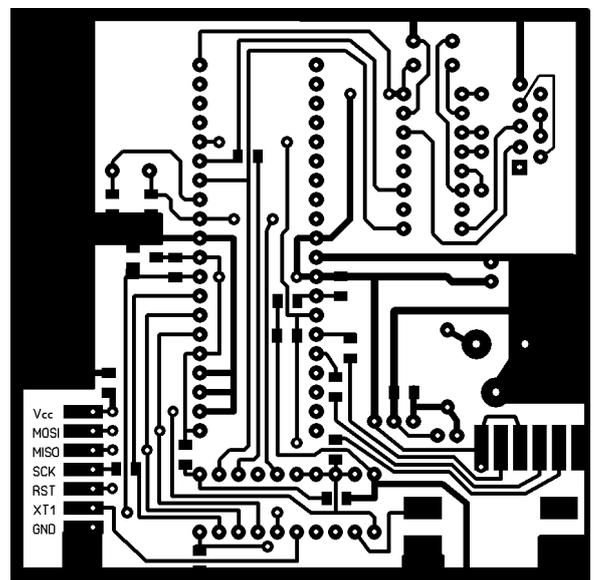
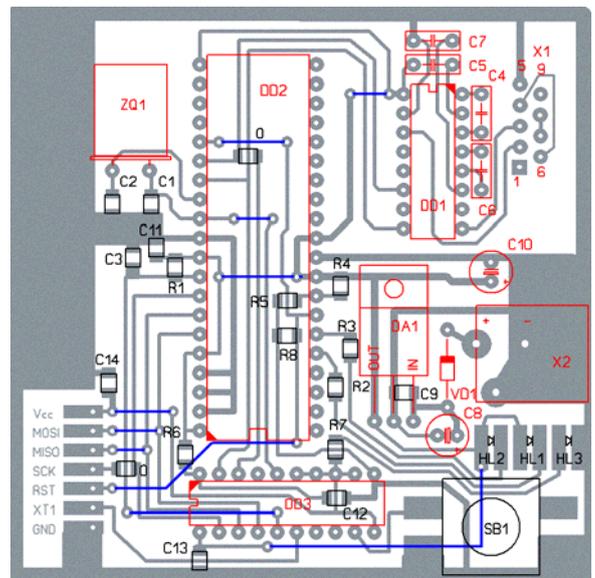


Рис. 3 Основная плата программатора

После сборки следует внимательно проверить правильность монтажа, отсутствие короткозамкнутых дорожек и непропаянных соединений. После этого нужно проверить наличие напряжения питания 5 В на выходе DA1. Если все в порядке, можно приступить к «оживлению» устройства. Это достаточно серьезная процедура и спешить тут не следует. Контроллер DD3 перед установкой в схему необходимо запрограммировать. Его прошивка разработана на основе Appnote AVR910: In-System Programming [4]. Первая версия этого документа датирована 1997 г. Первоначально использовался контроллер типа AT90S1200, в дальнейшем, не без активного участием радиолюбителей программа была адаптирована для AT90S2313, а ее возможности расширены [5].

В предлагаемом устройстве для записи в контроллер DD3 следует использовать файл прошивки **isp\_2313.hex**, который можно загрузить с моего сайта.

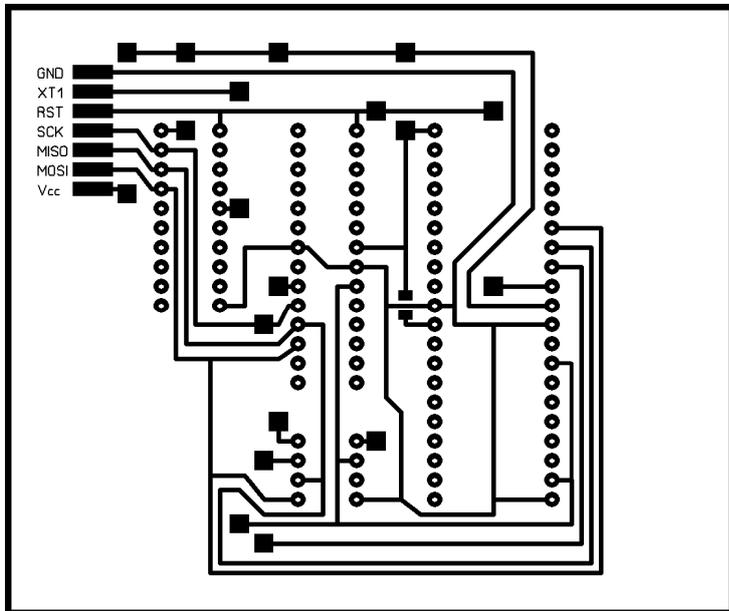
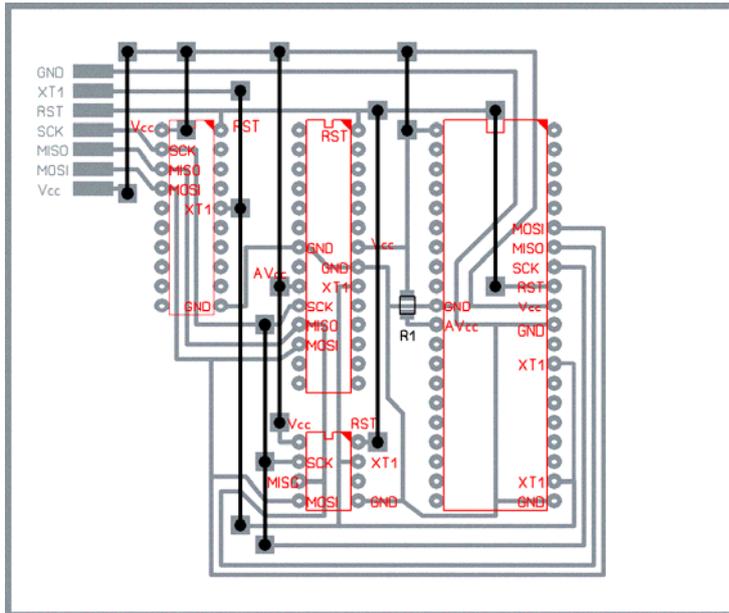


Рис. 4 Плата коммутации

Получается, что для того, чтобы сделать этот программатор, необходим другой программатор. Но он нужен только один раз, и для этой цели можно воспользоваться упоминавшимся ранее PonyProg или обратиться за помощью к друзьям. Как уже говорилось, с одинаковым успехом можно использовать как AT90S2313, так и ATtiny 2313. Прошивка будет работать с любым из них. Однако, если на Fuse биты первого можно не обращать внимания, используя их значения по умолчанию, то для ATtiny2313 Fuse биты следует установить следующим образом:  
**SPMEN = 1, DWEN = 1, EESAVE = 1, SPIEN = 0, WDTON = 1, BODLEVEL2...0 = 111, RSTDISBL = 1, CKDIV8=1, CKOUT=1, SUT1...0=11, CKSEL3...0=1101.**

Не забывайте, что 0 – значит запрограммирован, а 1 – нет. К сожалению, фирма Atmel не предусмотрела возможность включать состояние этих бит в файл прошивки, а их неправильная установка может привести к полной неработоспособности устройства. Поэтому, в случае возникновения сомнений, следует руководствоваться фирменной документацией на контроллер ATtiny2313, а тем, у кого нет опыта в работе с программаторами, предпочтительно поискать старый AT90S2313.

Установка Fuse бит ATmega8535 по умолчанию также не подходит, и, если есть возможность, их желательно предварительно запрограммировать следующим образом:

**S8515C = 1, WDTON = 1, SPIEN = 0, CKOPT = 0, EESAVE = 1, BOOTSZ1...0 = 00, BOOTRST = 1, BODLEVEL = 1, BODEN = 1, SUT1...0 = 11, CKSEL3...0 = 1111.**

Запрограммировать их можно будет и непосредственно в описываемом программаторе, но лучше использовать контроллер с предварительно установленными Fuse битами. Дело в том, что по умолчанию в качестве тактового установлен внутренний RC генератор на 1 МГц. Соответственно, на выводе XTAL2 никакого сигнала не будет, значит и на DD3 тактовая частота подаваться не будет. А это приведет к полной неработоспособности устройства.

Выходов из этой ситуации может быть несколько. Первый, как уже указывалось, использовать контроллер с предварительно запрограммированными Fuse битами. Если это невозможно, придется запаять временную перемычку между выводом XTAL1(13) DD2 и XTAL2(4) DD3. Теперь кварц окажется подключенным и к DD3. Все будет работать, но... только до того момента, пока Fuse биты не переконфигурируют тактовый генератор DD2 на работу с кварцем. После этого временную перемычку нужно удалить, разумеется, предварительно отключив питание. Если при установке бит была допущена ошибка, повторно этот способ может и не сработать. Но не стоит отчаиваться. В качестве крайней меры можно рекомендовать временно разорвать соединение между выводами XTAL2(12) DD2 и XTAL1(5) DD3 и подключить к DD3 второй кварц и 2 конденсатора аналогично ZQ1 и C1, C2. Или подать на вывод XTAL1 DD3 сигнал от внешнего источника необходимой частоты и амплитуды.

Теперь наступило время подключить программатор к COM порту и подать на него питание при **НАЖАТОЙ** кнопке SB1. Ни один из светодиодов не должен светиться. После этого нужно запустить AVR Studio. Желательно использовать версию не ниже 4.09. Все дальнейшие пояснения будут относиться к версии 4.12. После запуска в меню Tools надо выбрать AVR Prog.

Если все сделано правильно, через несколько секунд на экране появится окно, показанное рис. 5. Нажатие на кнопку «Advanced» (в правом нижнем углу) приведет к переходу на следующее окно. Если Fuse биты были заранее запрограммированы, то выглядеть оно будет так, как показано на рис. 6.

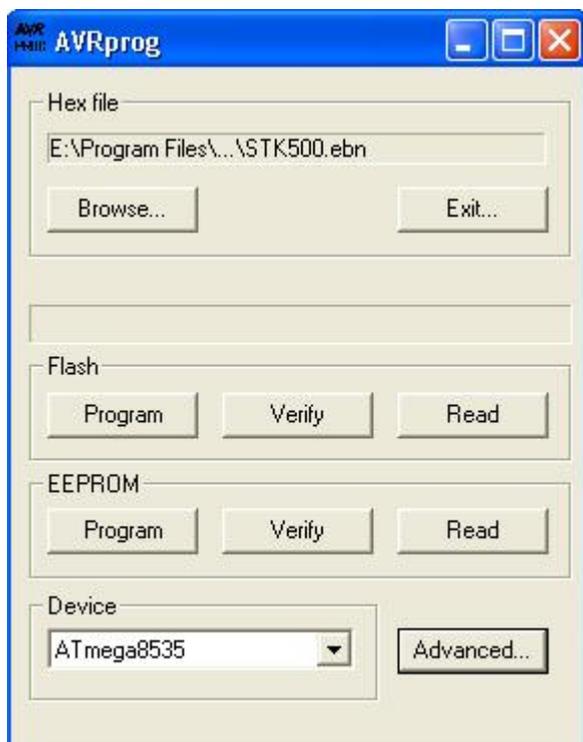


Рис. 5

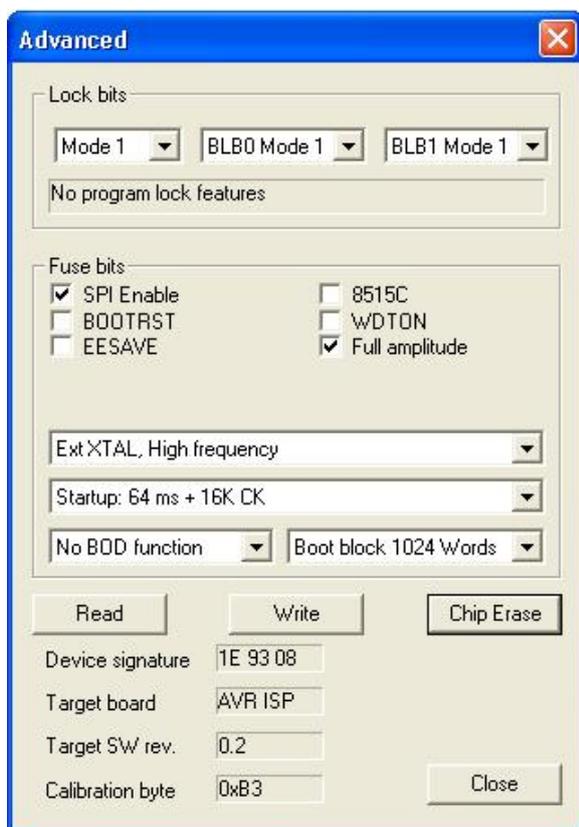


Рис. 6

Если используется чистый контроллер, нужно установить все биты в соответствии с этим рисунком. Если в поле «Device signature» только знаки вопроса, значит нет связи с программатором. Если были сделаны какие-то изменения, нужно нажать на кнопку «Write». Теперь следует восстановить первоначальную схему, если она была изменена для программирования Fuse бит как было описано выше и закрыть это окно, возвращаясь к предыдущему. Не помешает еще раз нажать на кнопку «Advanced» и убедиться в отсутствии ошибок.

Теперь следующий шаг – нажатие кнопки «Browser» и указание пути к файлу прошивки основного контроллера. Тип файлов следует выбрать «All files». Если AVR Studio была установлена в каталог по умолчанию, путь будет такой: C:\Program files\Atmel\AVR Tools\STK500\STK500.EBN

И наконец, нажатием кнопки «Flash Program» завершается «оживление» программатора. Нажимая «Exit» завершаем работу с AVR Prog и закрываем окно. Через несколько секунд на программаторе загорятся светодиоды HL2 и HL3 сигнализируя о готовности к работе.

Подобные действия следует производить каждый раз после установки новой версии AVR Studio для обновления прошивки ATmega8535. Только никаких проблем с Fuse битами уже не будет, они больше не будут изменяться. Меняется только версия файла STK500.EBN.

Теперь программатор можно запускать из меню Tools → Program AVR → Connect. При этом в панельку должен быть установлен программируемый контроллер. Если этого предварительно не сделать или неправильно указать его тип, может появиться сообщение о невозможности входа в режим программирования. Ничего страшного, нужно просто установить контроллер в панельку, предварительно отключив Vcc нажатием кнопки SB1. После установки нужно вновь включить Vcc и указать тип контроллера.

При первом запуске надо на вкладке «Board» установить тактовую частоту шины SPI = 230,4 kHz. Это значение выбирается исходя из того, что частота SPI не должна превышать  $F_{такт}/4$ .  $F_{такт}$  на XT1 = 1,8 МГц. Для сохранения этого выбора в энергонезависимой памяти нужно нажать кнопку «Write».

Работа с AVRISP интуитивно понятна и подробно описана в Help → AVR Tools User Guide. Переписывать это описание здесь не имеет смысла. Работа с предлагаемым программатором полностью идентична работе с фирменным AVRISP. Только не следует соглашаться на автоматический upgrade, а делать это всегда в ручном режиме, как описано выше.

И еще одно ограничение. Напряжение питания пользовательского устройства должно быть 5 В.

В заключение информация о том, по какой цене можно приобрести наиболее дефицитные комплектующие для этого устройства в интернет-магазине <http://www.chipinfo.ru>. Пассивные SMD компоненты можно извлечь из отработавших свой срок компьютерных плат. Все цены актуальны на начало

2006г. Для сравнения приведена стоимость серийно выпускаемого AVRISP.

Наименование	Цена
ATmega8535	150 руб.
ATtiny2313	40 руб.
MAX232	16 руб.
<b>Итого</b>	<b>206 руб.</b>
<b>Фирменный AVRISP</b>	<b>1510 руб.</b>

### Литература:

- 1. PonyProg - <http://www.lancos.com/prog.html>
- 2. А. Долгий. Разработка и отладка устройств на МК – Радио, 2001, №5-12.
- 3. И. Хуртин. Преобразователь интерфейса USB-RS232 на микросхеме FT232BM – Радио, 2005, №10, с.27-29.

- 4. Appnote AVR910: In-System Programming - <http://www.atmel.ru/Disks/AVR%20Technical%20Library/index.html>
- 5. AVR910 - <http://www.klaus-leidinger.de/mp/Mikrocontroller/AVR-Prog/AVR-Programmer.html>

Прошивку контроллера, чертеж печатной платы в формате Sprint Layout и другие дополнительные материалы к этой конструкции можно загрузить с сайта автора по адресам:

<http://ra4nal.qrz.ru>  
<http://ra4nal.lanstek.ru>  
<http://ra4nalr.tut.ru>

Разработка 2006 г.

Коммерческое использование с согласия автора.  
Перепечатка со ссылкой на первоисточник.