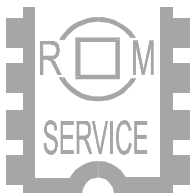


ROMSERVICE



Универсальный программатор

Octopus

Инструкция по эксплуатации

Москва 2016

Содержание.

1. Назначение прибора.....	3
2. Устройство прибора.....	3
3. Обновления.....	4
4. Основы работы с энергонезависимой памятью. Особенности микросхем различных семейств.....	5
4.1. Общие сведения.....	5
4.2. Mask ROM.....	7
4.3. Flash.....	7
4.4. EEPROM.....	8
4.5. NVRAM и FRAM.....	11
4.6. DataFlash.....	12
4.7. Микроконтроллеры MCS-51.....	13
4.8. PIC-контроллеры.....	15
4.9. AVR-контроллеры.....	18
5. Подготовка к работе.....	20
6. Порядок работы.....	21
7. Форматы файлов.....	23
8. Внутрисхемное программирование.....	24
9. Обновление ПО.....	26
10. Базовый комплект поставки.....	27
11. Комплект поставки Octopus+.....	27
12. Гарантийные обязательства.....	27
13. Возможные проблемы при работе с программатором и способы их решения.....	28

1. Назначение прибора.

Универсальный программатор Octopus предназначен для работы с широким спектром микросхем энергонезависимой памяти с последовательным интерфейсом программирования семейств EEPROM, FLASH, NVRAM, FRAM и др., а также с микроконтроллерами, содержащими такую память.

Функциональные возможности и технические характеристики программатора Octopus делают его пригодным для применения в условиях лабораторий и мастерских при разработке и ремонте электронной техники, а также при мелко- и среднесерийном промышленном производстве.

2. Устройство прибора.

На верхней (лицевой) стороне корпуса программатора расположены: контактная панель для установки микросхем и три светодиодных индикатора. На задней стенке корпуса размещен разъем mini-USB для подачи питания и подключения к персональному компьютеру. Рекомендуется использовать кабель mini-USB, входящий в комплект поставки программатора или другой высококачественный низкоомный кабель (суммарное сопротивление жил питания не должно превышать 3,5 Ом). Не допускается применение высокоомных кабелей, предназначенных только для передачи данных, а не для питания и зарядки приборов.

Универсальная 8-контактная тестовая панель класса UltraLIF (Ultra Low Insertion Force) позволяет устанавливать микросхемы в 8-выводном корпусе DIP. Панели UltraLIF, обладая такими же характеристиками по долговечности, легкости и скорости установки и извлечения микросхем, необходимыми при серийном производстве, как и ZIF-панели, обеспечивают дополнительно эффект легкого трения, значительно повышающий качество контакта с не новыми микросхемами при ремонте аппаратуры. Положение 1-го вывода микросхемы по умолчанию помечено на корпусе прибора треугольной меткой. Для отдельных микросхем, требующих противоположного положения 1-го вывода, в меню микросхем управляющей программы помещается соответствующий комментарий.

Микросхемы в корпусе DIP, имеющие более 8 выводов, а также выполненные в других видах корпусов (PLCC, SOIC, TSOP, QFP, PSOP, SSOP, BGA и др.) устанавливаются в панель через соответствующие адаптеры (переходники), которые могут быть приобретены отдельно или изготовлены самостоятельно. При этом, для подавляющего большинства 8-выводных микросхем требуется распайка (схема соединений) адаптеров «pin-to-pin» («один-в-один»).

Панель программатора используется также как разъем для внутрисхемного программирования. Расположение сигналов при этом различно для разных семейств микросхем и приводится в соответствующих разделах настоящей инструкции.

Светодиодный индикатор, помеченный надписью «ON», является индикатором питания и подключения к USB-порту компьютера.

Светодиодный индикатор, помеченный надписью «GO», светится в процессе выполнения операций над микросхемой, установленной в панели, когда на нее подано питание. В это время, установка/удаление микросхемы в панели не допускается.

Светодиодный индикатор, помеченный надписью «OL» (OverLoad - Перегрузка), светится красным цветом при срабатывании защиты по току, вызванном неисправностью программируемой микросхемы, ее неверным выбором в меню или неправильной установкой в панель.

3. Обновления.

Конструкция программатора Octopus предусматривает возможность обновления его внутреннего программного обеспечения (ПО) с целью расширения его функциональных возможностей. Обновление осуществляется под управлением управляющей компьютерной программы и предлагается автоматически при несоответствии версий управляющей программы и внутреннего ПО программатора. «Обратное» обновление (замена более новой версии на более старую) также допускается, но не рекомендуется, т.к. уменьшает функциональные возможности программатора. Новые версии управляющей

программы размещаются, по мере их выпуска, на интернет-сайте производителя, в разделе «Техподдержка».

4. Основы работы с энергонезависимой памятью. Особенности микросхем различных семейств.

4.1. Общие сведения.

Энергонезависимой называется память, способная хранить информацию в отсутствие электропитания (в отличие от оперативной памяти, информация в которой при выключении питания теряется). Довольно часто используемая аббревиатура «ПЗУ» – Постоянное Запоминающее Устройство (по англ. ROM – Read Only Memory – память только для чтения), строго говоря, не совсем корректна. Первые, наиболее старые представители энергонезависимой памяти, действительно использовались в аппаратуре только в режиме чтения, а их запись (программирование) осуществлялась, либо в процессе изготовления кристалла, либо перед установкой в аппаратуру с помощью программатора. В дальнейшем, по мере совершенствования технологии производства и упрощения методов и алгоритмов записи, их современные модификации все чаще стали использовать в приборах и устройствах в режимах записи, стирания и перезаписи. Например, в модулях фискальной памяти кассовых аппаратов, в них заносится итоговая информация о дневной выручке и количестве покупок. В телевизорах ПЗУ используют для хранения различных настроек, а в телефонных аппаратах – для хранения и быстрого набора часто используемых телефонных номеров («записная книжка»). Все эти применения противоречат самому смыслу понятия «память только для чтения». Попытки устранить это противоречие привели к обрастанию аббревиатуры «ПЗУ» уточняющими приставками: ППЗУ – программируемые ПЗУ, СППЗУ – стираемые ППЗУ, РПЗУ – репрограммируемые ПЗУ (PROM – Programmable ROM, EPROM – Erasable PROM, EEPROM – Electrically Erasable PROM) и т.д. Однако, наиболее точным обобщающим названием этого класса приборов является «энергонезависимая память». Этого понятия, применяя сокращение ЭП, мы и будем придерживаться в настоящей инструкции.

Энергонезависимая память выпускается, как в виде самостоятельных устройств – собственно микросхем памяти, так и в составе других устройств, например, микроконтроллеров, где она используется для хранения управляющей микропрограммы, а иногда и данных.

По способу стирания (удаления старой информации перед записью новой), ЭП делится на нестираемую (программируемую в процессе изготовления кристалла либо однократно программируемую потребителем), стираемую ультрафиолетовым излучением (УФ), электрически стираемую (стирание осуществляется электрическими сигналами программатора), автостираемую (удаление старой информации выполняется автоматически, в процессе записи новой).

Наиболее простые представители микросхем ЭП содержат один массив ячеек памяти, информация в котором может быть считана, записана, а для электрически стираемых микросхем и стерта, при помощи программатора. Более сложные приборы содержат несколько (обычно – 2) массивов памяти, например, микроконтроллеры, содержащие память программ и память данных. Во многих микросхемах памяти и микроконтроллерах имеются также одна или несколько дополнительных ячеек, запись определенных данных в которые, управляет отдельными специфическими функциями микросхемы. К таким дополнительным ячейкам относятся, например, биты и регистры защиты, конфигурации, состояния, идентификации и др.

Краткие сведения о функциональном составе различных микросхем памяти и микроконтроллерах приведены ниже, отдельно по семействам микросхем. Более подробную информацию по конкретным микросхемам можно найти в специальной литературе и в фирменных описаниях микросхем (datasheets), распространяемых их производителями через интернет-сайты, а также на компакт-дисках и в виде печатных каталогов.

Для первичной идентификации информации в микросхеме памяти или в файле, часто используют «контрольную сумму» – число, упрощенно говоря, представляющее собой сумму всех данных в блоке памяти. Обычно контрольной суммой пользуются для быстрой проверки микросхем после записи (она должна совпасть с контрольной суммой микросхемы-образца или файла,

из которого производилась запись). Следует иметь в виду, что различные программаторы и программы используют при подсчете контрольной суммы разные алгоритмы (формулы), поэтому значения контрольной суммы одной и той же информации, вычисленные разными программаторами, могут различаться.

4.2. Mask ROM.

Микросхемы Mask ROM («масочные» ПЗУ) в наибольшей степени оправдывают аббревиатуру ROM – Read Only Memory – память только для чтения. Информация в них заносится на заводе в процессе изготовления кристалла по фотомаске. Ни стереть, ни дополнительно что-то записать в них нельзя. Такой тип памяти в течении многих лет был наиболее выгодным по цене при заказе большого количества микросхем с одинаковой прошивкой для крупносерийного производства. В настоящее время разница в цене Mask ROM и Flash значительно сократилась и масочная память применяется редко.

Пример микросхем этого семейства – серия MX23Lxx54 от производителя Macronix с интерфейсом SPI.

Такие микросхемы программатором можно только считать и, при необходимости замены вышедших из строя, скопировать в полностью совместимые по расположению выводов и протоколам чтения Flash.

4.3. Flash.

Стирание микросхем Flash-памяти осуществляется электрически, командами программатора. Многие микросхемы Flash, помимо общего стирания, дополнительно допускают стирание отдельных частей памяти, называемых блоками или секторами, причем разбивка на блоки может быть равномерной (блоки равного объема) или неравномерной. При выполнении команды стирания блока, программатором будет выполнено стирание того блока, в котором находится начальный адрес выбранной для работы части микросхемы. Наиболее популярными являются SPI Serial Flash серии 25, выпускаемые десятками различных фирм. Отдельные производители выпускают постранично автостираемые Flash, приближающиеся по своим

свойствам к EEPROM. Пример – серии M25PE и M45PE от ST-microelectronics, Numonix, Micron.

Почти все SPI Serial Flash имеют энергонезависимый регистр статуса – 8, 16 или 24 бит, управляющих режимами работы микросхемы и защитой отдельных блоков от случайного перепрограммирования. После установки защиты, информация в защищенных блоках не может быть изменена (стирание и запись этих блоков не будут выполнены, данные сохраняют старые значения). Иногда такая защита бывает однократно устанавливаемой, т.е. ее отключение невозможно. Однако, в большинстве микросхем, защита может быть отменена путем изменения соответствующих битов регистра статуса. Назначение битов регистра статуса у различных микросхем отличается и приводится в фирменных технических описания микросхем («даташитах»).

Отдельные SPI Serial Flash снабжены дополнительными служебными регистрами и/или управляющими битами, работа с которыми полностью поддерживается программатором.

Большинство SPI Serial Flash последних лет выпуска имеют один или несколько дополнительных блоков памяти небольшого объема. Эти блоки могут быть стираемыми, однократно программируемыми или однократно защищаемыми. В технике эти дополнительные блоки используются крайне редко из-за низкой совместимости протоколов обращения к ним у микросхем различных производителей, что приводит к ограничению возможности замены микросхем аналогами при производстве аппаратуры. Тем не менее программатор обеспечивает возможность выполнения всех операций с данными блоками, предусмотренных производителями микросхем.

4.4. EEPROM.

С точки зрения пользователя микросхемы EEPROM, в отличие от Flash, не требуют предварительного стирания старой информации перед записью новой, т.е. любую ячейку можно перезаписать «поверх старой информации», не затрагивая остальные. В действительности стирание требуется, но оно осуществляется микросхемой автоматически поячеечно. Некоторые наиболее старые EEPROM требуют подачи команды стирания каждой ячейки перед ее перезаписью. Программатор это

делает автоматически и для пользователя работа с такими старыми микросхемами не отличается от работы с современными.

Наиболее распространены микросхемы EEPROM с последовательным интерфейсом (Serial EEPROM или SEEPROM), работающие по шинам I²C (серии 24, 85), Microwire (серия 93), 4-Wire (серия 59), SPI (серии 25, 95), SCI (серия 64), MPS (серия 84), I²C/Bitstream (серия AT17), Mitsubishi 3-wire (серия M6M80), Rohm 3-wire (серия BR90), INTERMETAL (NVM3060) и др.

Большинство микросхем SEEPROM с шинами Microwire, 4-Wire, SCI, Rohm 3-wire имеют 16-разрядную внутреннюю организацию. Многие экземпляры имеют возможность переключения для работы в 8-разрядном режиме. При работе с ними программатор Octopus использует 16-разрядный режим, данные в файле размещаются младшим байтом вперед, что не всегда соответствует внутренней автоматическому перераспределению данных в этих микросхемах при переключении разрядности. Поэтому может возникнуть проблема совместимости формата файла, полученного от другого программатора, использующего при считывании микросхемы 8-разрядный режим. Функциональные возможности редактора управляющей программы программатора позволяют осуществить необходимое преобразование файла путем попарного обмена байт в нем. Некоторые фирмы выпускают микросхемы с шиной Microwire, работающие только в 8-разрядном режиме.

SEEPROM с шиной I²C допускают возможность внутрисхемного программирования. Такой режим доступен в программаторе Octopus для микросхем, у которых в качестве производителя указано «Standard SEEPROM». При включении этого режима сигналы на панели программатора располагаются в следующем порядке:



При установке в схему эти микросхемы получают схемный адрес (Slave address), определяемый комбинацией сигналов на ее адресных входах, что дает возможность подключать несколько

микросхем к одной общей шине I²C. В режиме внутрисхемного программирования программатор Octopus позволяет осуществлять выбор этого адреса для обращения к конкретной микросхеме. Питание микросхемы осуществляется устройством, в котором она установлена. **Значение напряжения питания микросхемы в настройках программатора, следует установить равным напряжению питания микросхемы в схеме, где осуществляется ее программирование. Это необходимо для согласования уровней вырабатываемых программатором сигналов с напряжением питания схемы.**

Многие SEEPR0M имеют ряд дополнительных ячеек, управляющих режимами работы микросхемы или защитой отдельных участков памяти от случайного перепрограммирования. Ниже приводятся несколько примеров таких дополнительных функций.

SEEPR0M с шиной Microwire, имеющие регистр защиты, обычно содержат в своей маркировке букву «S» (например, 93Sxxx, 93CSxxx, 93LCSxxx). В этот регистр может быть записан адрес, начиная с которого информация в ПЗУ защищена от перезаписи. Содержимое этого регистра может быть считано, очищено (защита снята), записано (защита установлена) и затем заблокировано (защита установлена навсегда).

SEEPR0M с шиной I²C (34C02, 34WC02, 24LCS52, 24CSxxx и некоторые другие) имеют возможность однократной установки защиты на часть памяти (обычно расположенную в конце массива). После установки защиты, информация в этом блоке не может быть изменена. Отключение (снятие) такой защиты невозможно.

В микросхемах 24xx65 фирмы Microchip также предусмотрена функция установки защиты информации от последующего изменения на любое количество расположенных подряд блоков размером 0.5 КБайт. Защита может быть установлена только 1 раз и не может быть снята. Кроме того, один из блоков (0.5 КБайт) имеет увеличенный в 10 раз ресурс по количеству циклов перезаписи (HEB – High Endurance Block), положение которого в адресном пространстве микросхемы может быть запрограммировано пользователем.

Выбор различных уровней защиты информации от случайной перезаписи в SPI SEEPR0M серий 25xxx, 25Cxxx, 95xxx и др.

осуществляется установкой отдельных битов имеющегося в микросхеме регистра статуса в соответствии с описанием на конкретную микросхему. В большинстве случаев, для отключения защиты и обеспечения возможности перезаписи всей памяти, в статус-регистр достаточно записать все нули.

EEPROM серии AT17Cxxx содержат конфигурационный байт выбора полярности сигнала RESET/OE. Для выбора низкого активного уровня сигнала RESET, в этот байт необходимо записать значение FFh, для выбора высокого активного уровня RESET – 00h. Запись в этот байт других значений не допускается. Содержимое этого байта может быть также считано программатором.

Микросхемы AT17C512A/010A имеют дополнительный конфигурационный бит блокировки сигнала DCLK. Контрольное считывание состояния этого бита в данных микросхемах не предусмотрено.

EEPROM с шинами Microwire, Rohm 3-wire и некоторые другие, выполненные в корпусах SOIC8, выпускается с двумя видами расположения сигналов на выводах – основным и альтернативным (кристалл внутри микросхемы повернут на 90°). Названия таких микросхем присутствуют в меню управляющей программы программатора с частичным включением буквенно-цифровых суффиксов, ориентируясь на которые, программатор автоматически устанавливает на контактах панели альтернативное расположение сигналов, что делает возможным применение для микросхем со стандартной и альтернативной распиновкой одних и тех же адаптеров DIP8>>SOIC8 с распайкой «pin-to-pin».

4.5. NVRAM и FRAM.

Понятие «Энергонезависимое ОЗУ» (Non-volatile RAM или NVRAM) включает в себя два подсемейства памяти. Оба они отличаются от других видов ЭП моментальной записью ячейки. Поэтому вместо термина «программирование» по отношению к этим микросхемам обычно применяют термин «запись».

Первая разновидность NVRAM представляет собой обычные статические ОЗУ с очень низким потреблением энергии в режиме хранения, встроенным литиевым элементом питания (батарежкой) и усиленной защитой от искажения информации в момент включения и выключения внешнего питания. Важным преимуществом этих

микросхем является неограниченное количество циклов перезаписи (для EPROM, Flash и EEPROM оно обычно составляет от одной тысячи до ста миллионов). Мировым лидером в выпуске такой памяти долгое время являлась компания Dallas Semiconductor, позже продавшая свое производство компании Maxim.

Другой разновидностью NVRAM являются микросхемы, содержащие на одном кристалле энергозависимое ОЗУ (RAM) и резервную EEPROM-память, с возможностью сохранения (копирования) содержимого RAM в EEPROM и обратного восстановления данных из EEPROM в RAM. Многие из этих микросхем имеют функцию автоматического восстановления данных из EEPROM в RAM при включении питания. Пример – микросхемы 24x30/44/45 от Catalyst, Seiko и Xicor.

Микросхемы FRAM изготавливаются на базе материалов – сегнетоэлектриков (ферроэлектриков). Моментальная запись ячейки и почти неограниченное (порядка 10 миллиардов) количество циклов перезаписи приближают их по своим свойствам к NVRAM, однако пока уступают им по стоимости и размеру кристалла. FRAM часто выпускаются в виде микросхем, полностью совместимых по расположению выводов, алгоритмам и протоколам, и даже совпадающими по названию с популярными EEPROM с шинами I²C и SPI (серий 24 и 25). Наибольший вклад в развитие этого типа памяти внесли компании Ramtron, Cypress, Fujitsu, LAPIS.

4.6. DataFlash.

Микросхемы этого семейства представлены сериями AT45D и AT45DB фирм Atmel и Adesto, работающими по последовательной шине SPI. Как и EEPROM, они не требуют стирания старой информации перед записью. Они имеют страничную организацию с размером страниц 256+8, 512+16 или 1024+32 байт. Помимо основной памяти (256, 512 или 1024 байт), каждая страница содержит дополнительную память (8, 16 или 32 байта), что делает эти микросхемы удобными для применения при изготовлении FLASH-карт и других устройств, действующих по принципу электронного диска. В этом случае, в дополнительной памяти размещается служебная или контрольная информация.

При работе с сериями AT45D и AT45DB программатор Octopus осуществляет отдельную работу с основной и дополнительной памятью. При этом информация из всех страниц выбранного типа памяти размещается в файле слитно. Такой способ работы предоставляет больше удобств для просмотра и редактирования данных, считанных из микросхем Serial DataFlash. Некоторые другие программаторы осуществляют считывание данных из этих микросхем страницу за страницей, без разделения на их основные и дополнительные части, что может вызвать проблему совместимости форматов файлов. Для разделения слитного файла прошивки на два отдельных – для основной и дополнительной памяти можно воспользоваться специальной программой-конвертером, которая предоставляется компанией ROMSERVICE по запросу.

Новое поколение Serial DataFlash серии AT45DBxxxD имеет ряд дополнительных функций – дополнительные страницы памяти для размещения идентификационной информации, регистры, управляющие снимаемой и неснимаемой (постоянной) защитой отдельных блоков памяти от случайного перепрограммирования, а также режим работы «Power of 2», в котором дополнительная память во всех страницах отключается навсегда и размер страниц становится двоично-кратным для обеспечения большей совместимости с обычными микросхемами SPI Serial Flash.

4.7. Микроконтроллеры MCS-51.

Программатор Octopus позволяет осуществлять работу с микроконтроллерами семейства MCS-51, имеющими последовательный режим программирования.

К таким контроллерам относятся, в частности, микросхемы серий AT89S/AT89LS фирмы Atmel. Они имеют два режима программирования: параллельный, который требует 20 или более сигналов и не может быть реализован в программаторе Octopus, и последовательный, программирование в котором осуществляется по шести сигналам. У части этих микросхем последовательный режим может отключаться специальным конфигурационным битом. Работа с этим битом возможна только в параллельном режиме.

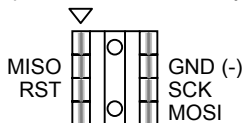
Программатор Octopus работает с этими микросхемами в последовательном режиме, причем для повышения скорости

работы программирование в панели программатора осуществляется при высокой частоте тактирования и соответствующей частоте шины SPI (150/375/750 кГц, в зависимости от типа микросхемы).

Некоторые микросхемы этих серий, помимо основной памяти, используемой для хранения выполняемой микроконтроллером программы, содержат дополнительную энергонезависимую память для данных.

Все контроллеры семейства MCS-51 имеют возможность защиты информации от считывания и копирования. Установку защиты следует производить после записи и контрольного считывания записанных данных. После установки защиты, считывание информации из контроллера становится невозможным. Защита устанавливается путем программирования т.н. «битов защиты», количество которых может быть от 1 до 3. В некоторых микроконтроллерах предусмотрена функция считывания состояния битов защиты, в других она отсутствует. Биты защиты стираются одновременно с остальной информацией в памяти микросхемы.

Серия AT89S/AT89LS допускает внутрисхемное программирование. При выборе внутрисхемного режима частота шины SPI снижается до 20 кГц, что позволяет программировать микросхемы в устройствах с низкой частотой тактирования, а сигналы на панели программатора располагаются в следующем порядке:



Питание и тактирование микросхемы осуществляется устройством, в котором она установлена. Выводы микросхемы, используемые для программирования, должны быть отключены от других цепей. **Значение напряжения питания микросхемы в настройках программатора, следует установить равным напряжению питания микросхемы в схеме, где осуществляется ее программирование. Это необходимо для согласования уровней вырабатываемых программатором сигналов с напряжением питания схемы.**

Микроконтроллеры семейства LPC900 фирм Philips/NXP (серия P89LPC9xx) имеют один последовательный режим программирования.

Биты защиты в этих контроллерах защищают отдельные выбранные блоки памяти от последующей записи и стирания, а для защиты от копирования эти микросхемы вообще лишены возможности считывания информации из памяти. Поэтому для проверки после записи вместо контрольного считывания и сравнения с файлом используется сравнение контрольных сумм блоков данных в файле, подсчитанных управляющей программой программатора, с контрольными суммами блоков данных в памяти микросхемы, подсчитанными встроенным в нее автоматом.

Семейство LPC900 также допускает внутрисхемное программирование. При выборе внутрисхемного режима сигналы на панели программатора располагаются в следующем порядке:



Питание микросхемы осуществляется устройством, в котором она установлена. Выводы микросхемы, используемые для программирования, должны быть отключены от других цепей. **Значение напряжения питания микросхемы в настройках программатора, следует установить равным напряжению питания микросхемы в схеме, где осуществляется ее программирование. Это необходимо для согласования уровней вырабатываемых программатором сигналов с напряжением питания схемы.**

4.8. PIC-контроллеры.

Микросхемы PIC представляют собой высокопроизводительные однокристальные микроконтроллеры на базе RISC-процессора фирмы Microchip. В состав семейства входят однократно программируемые, УФ-стираемые и электрически стираемые микросхемы.

Для работы с памятью этих контроллеров требуется подача на них высокого напряжения (т.н. «напряжения программирования»). Требуемая величина Этого напряжения устанавливается программатором автоматически, в зависимости от выбранного типа микросхемы, но может быть, при необходимости, изменена пользователем. Следует иметь ввиду, что завышенное напряжение программирования в большинстве случаев приводит к выходу микросхемы из строя.

Энергонезависимая память PIC-контроллеров имеет 12-, 14- или 16-разрядную организацию и содержит память программ, ID-область и одно или несколько слов конфигурации. Каждое слово, соответствующее ячейке памяти микросхемы, хранится в файле в виде 2-х байтов, сначала младшие 8 бит, затем старшие.

Многие электрически стираемые PIC-контроллеры дополнительно имеют память данных, в которой значащими являются только 8 младших бит (младший байт) слова, старшие биты содержат '0'.

В соответствии с рекомендациями фирмы Microchip, информация в файле располагается следующим образом:

- для контроллеров с 12-разрядной памятью – сначала программа, затем ID-область, слово конфигурации по адресу 1FFEh-1FFFh;
- для контроллеров с 14-разрядной памятью – сначала программа, ID-область по адресам 4000h-4007h, слово конфигурации по адресу 400Eh-400Fh;
- память данных размещается, начиная с адреса 4200h;
- для контроллеров серии PIC18 с 16-разрядной памятью – программа, начиная с адреса 000000h, ID-область с адреса 200000h, слова конфигурации с адреса 300000h, память данных с адреса F00000h.

Распределение адресов для блоков информации, предназначенных для записи в различные области памяти в HEX-файлах, получаемых при компиляции программ для микроконтроллеров PIC18, делает неудобным прямое преобразование таких файлов в простой двоичный формат – размер файла получается слишком большим. Загрузку HEX-файла при работе с этими микросхемами следует осуществлять при помощи команды «Загрузить PIC18 HEX-файл», а не стандартной команды

«Импорт» меню «Файл». Данная команда открывает окно проводника для выбора HEX-файла, создает в окне редактора файлов новый файл и помещает в него информацию из загружаемого HEX-файла. При выполнении команды выводится информационное окно, содержащее сведения о наличии и объеме данных, предназначенных для тех или иных областей памяти микроконтроллера, и адреса размещения этих данных в созданном двоичном файле. Размер создаваемого двоичного файла и адреса размещения информации в нем зависят от типа микроконтроллера, выбранного в меню микросхем. Поэтому загрузку HEX-файла следует выполнять после выбора типа микросхемы в меню.

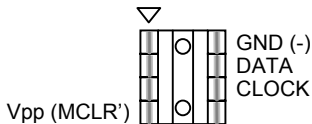
Последние 64 слова (128 байт) программной памяти микросхемы PIC14000 отведены для хранения калибровочной информации, предварительно записанной фирмой-изготовителем. Поскольку она не должна изменяться при программировании, следует выбирать для работы усеченный объем памяти контроллера. При стирании микросхемы (стираемые экземпляры имеют в своей маркировке буквы «JW»), содержимое калибровочной области должно быть предварительно считано, сохранено в файле с программой по соответствующим адресам и в последствии восстановлено при записи путем выбора полного объема памяти контроллера.

Память программ ряда других PIC-контроллеров также содержит в конце калибровочную область размером 1 слово (2 байта) и имеет те же особенности в работе, что и PIC14000. Для электрически стираемых контроллеров при выборе усеченного объема памяти считывание и последующее восстановление калибровочных данных осуществляется программатором автоматически, выполнение вышеописанной процедуры требуется лишь для УФ-стираемых контроллеров.

PIC-контроллеры имеют функцию защиты информации от несанкционированного доступа, которая включается путём записи определенных битов слова конфигурации. Ряд PIC-контроллеров предоставляют возможность отдельной установки защиты на память данных, память программ и на отдельные части памяти программ.

Интерфейс программирования PIC-контроллеров – последовательный, что делает возможным их программирование в составе различных устройств. Для использования этой возможности,

программатор предусматривает для этих контроллеров режим внутрисхемного программирования (ISP – In-System Programming). При выборе этого режима сигналы на панели программатора располагаются в следующем порядке:



Питание микросхемы осуществляется устройством, в котором она установлена. Выводы микросхемы, используемые для программирования, должны быть отключены от других цепей. **Значение напряжения питания микросхемы в настройках программатора, следует установить равным напряжению питания микросхемы в схеме, где осуществляется ее программирование. Это необходимо для согласования уровней вырабатываемых программатором сигналов с напряжением питания схемы.**

4.9. AVR-контроллеры

Микросхемы AVR представляют собой высокопроизводительные однокристальные микроконтроллеры на базе RISC-процессора фирмы Atmel – серий AT90, ATmega, ATtiny, ATxmega.

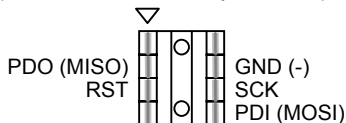
Память AVR-контроллеров состоит из 16-разрядной Flash-памяти программ (1...128 Кбайт), 8-разрядной EEPROM-памяти данных (64...4096 байт), 2...6 битов защиты информации и 8...24 битов конфигурации (Fuse bits). Биты конфигурации сгруппированы по 8 в байты – младший (Low byte, биты с 0 по 7), старший (High byte, биты с 8 по 15) и расширенный (Extended byte, биты с 16 по 23). ATxmega содержат 7 байт с битами конфигурации и 1 байт с битами защиты. Некоторые микросхемы, например ATtiny11, не имеют EEPROM-памяти данных.

Назначение битов защиты и конфигурации в разных микросхемах различно и подробно описано в документации на них. Некоторые микросхемы AVR, имеющие возможность тактирования от внутреннего RC-генератора, содержат до четырех служебных ячеек, в которых хранятся записанные производителем числа – калибровочные байты. Значения этих байтов может быть считано и

использовано записанной в микроконтроллер программой для коррекции частоты RC-генератора.

AVR-контроллеры (кроме серии ATxmega) имеют два режима программирования – высоковольтный и низковольтный (по последовательной шине SPI). Высоковольтный режим использует 7...8 сигналов у микросхем с количеством выводов менее 20 и 20...25 сигналов у микросхем с количеством выводов от 20 и более. Низковольтный режим использует 4 сигнала, не считая общего провода и питания. Поскольку программатор Octopus способен формировать только 8 сигналов, программирование в панели программатора AVR-контроллеров с малым количеством выводов осуществляется в высоковольтном режиме, а контроллеров с большим количеством выводов – в низковольтном при высокой частоте тактирования и соответствующей частоте шины SPI (375/750 кГц, в зависимости от типа микросхемы, для повышения скорости работы). Низковольтный режим может отключаться одним из битов конфигурации. Изменение состояния этого бита возможно только в высоковольтном режиме. Для некоторых самых старых микросхем этого семейства в низковольтном режиме недоступны отдельные функции (программирование и считывание битов конфигурации, считывание битов защиты).

Внутрисхемное программирование AVR-контроллеров (кроме ATxmega) всегда осуществляется в низковольтном последовательном режиме при низкой частоте шины SPI (20кГц), что позволяет программировать микросхемы в устройствах с низкой частотой тактирования. При этом сигналы на панели программатора располагаются в следующем порядке:

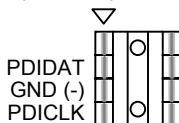


Питание и тактирование микросхемы осуществляются устройством, в котором она установлена. Выводы микросхемы, используемые для программирования, должны быть отключены от других цепей. **Значение напряжения питания микросхемы в настройках программатора, следует установить равным**

напряжению питания микросхемы в схеме, где осуществляется ее программирование. Это необходимо для согласования уровней вырабатываемых программатором сигналов с напряжением питания схемы.

Контроллеры серии ATxmega имеют один низковольтный последовательный режим программирования по специальной шине PDI, используемый как при работе с микросхемой в панели программатора, так и при внутрисхемной работе.

Сигналы для внутрисхемного программирования располагаются на панели программатора в следующем порядке:



Значение напряжения питания микросхемы в настройках программатора, следует установить равным напряжению питания микросхемы в схеме, где осуществляется ее программирование. Это необходимо для согласования уровней вырабатываемых программатором сигналов с напряжением питания схемы.

5. Подготовка к работе.

Перед началом работы внимательно изучите настоящую инструкцию.

Подключение программатора к персональному компьютеру (ПК) осуществляется через последовательный интерфейс USB. С В целях обеспечения достаточной мощности для питания программатора следует избегать подключения через USB-концентраторы, не снабженные собственным источником питания. Светодиод «ON» является индикатором включения питания и подключения к компьютеру.

Для работы программатора необходимо выполнить установку в компьютер программного обеспечения (ПО) – USB-драйвера и управляющей программы, которые размещены на официальном интернет-сайте ROMSERVICE (www.romservice.ru), в разделе

«Техподдержка». ПО предназначено для работы в операционной системе Windows.

При первом подключении программатора к порту USB компьютера Windows выводит сообщение об обнаружении нового устройства, распознает его как программатор Octopus и предлагает установить драйвер. Следует отказаться от автоматической установки (из библиотеки драйверов в Windows) и от поиска драйвера в интернете, а выбрать ручную установку и указать на папку, в которую помещен скачанный с сайта ROMSERVICE и разархивированный драйвер (inf-файл). В случае отказа от установки драйвера при первом подключении, в дальнейшем его можно установить из диспетчера устройств.

Драйвер программатора является не подписанным. Старые версии Windows (XP, 7) запрашивают подтверждение на установку таких драйверов. В новых версиях Windows (8, 10) для установки не подписанных драйверов необходимо выполнить перезагрузку компьютера в специальный режим. Порядок такой загрузки можно найти в справке по Windows или в интернете по запросу «установка неподписанных драйверов».

Для установки управляющей программы необходимо запустить установочный файл (setup.exe), далее следовать указаниям инсталлятора.

При необходимости удаления программы оно осуществляется из раздела «Установка/Удаление программ» в «Панели управления» Windows.

6. Порядок работы.

Для запуска управляющей программы используйте ярлык в разделе «Все программы» главного меню Windows, вызываемого кнопкой «Пуск». При необходимости этот ярлык можно скопировать на «Рабочий стол» или на панель быстрого запуска, перетащив его правой кнопкой мыши с последующем выбором операции «Копировать».

Для нормальной работы версия используемого ПО должна соответствовать версии внутреннего ПО программатора. При обнаружении несоответствия управляющая программа предложит осуществить замену (перезапись) внутреннего ПО – микро-

программы контроллера программатора. Эта операция используется для обновления программатора. Не рекомендуется замена версии микропрограммы на более старую. Такая замена сузит возможности программатора. В этом случае рекомендуется отказаться от замены и использовать последнюю версию управляющей программы.

Подробное описание работы с управляющей программой содержится в меню «Справка». Разделы «Оперции над микросхемами» и «Окно специальных функций» этого меню являются контекстно-зависимыми, т.е. их содержимое меняется в зависимости от типа микросхемы, выбранной в меню для работы.

Выбор типа микросхемы для работы осуществляется при помощи меню-списка, отображаемого в специальном окне управляющей программы. Названия микросхем в меню соответствуют их обозначениям в фирменной технической документации производителей и могут отличаться от маркировки на корпусе микросхемы, которая может быть сокращенной (особенно для миниатюрных корпусов) или, напротив, содержать в конце маркировки дополнительные символы – буквы и/или цифры, обозначающие тип корпуса, быстродействие, допустимый температурный диапазон, экологическое исполнение и прочую дополнительную информацию. У некоторых микросхем дополнительные символы располагаются не в конце маркировки, а в начале или внутри названия. В этом случае, они заменяются в меню символом «х». Микросхемы в меню расположены в алфавитном порядке без учета буквенно-цифрового префикса – нескольких (обычно 1...3) символов, в большинстве случаев обозначающих фирму-производителя. Префикс может вообще отсутствовать.

Для облегчения и ускорения поиска типа микросхемы в списке, предусмотрена возможность использования шаблона – ввода нескольких начальных символов названия микросхемы. При правке шаблона, для ввода предлагаются лишь те символы, которые действительно присутствуют в данной позиции хотя бы у одной микросхемы, удовлетворяющей уже составленной части шаблона.

Меню микросхем также содержит раздел «Последние 20», для быстрого выбора микросхем, работа с которыми уже недавно осуществлялась.

Включение и выключение программатора (подключение к компьютеру и отключение от него) следует производить при отсутствии микросхемы в панели программатора. Также следует избегать установки и извлечения микросхемы в панели во время выполнения операций, которые сопровождаются свечением светодиода «GO». В паузах между операциями светодиод «GO» не светится, на всех контактах панели поддерживается низкий потенциал, что обеспечивает безопасную установку и извлечение микросхемы.

Срабатывание защиты по току, вызванное неисправностью программируемой микросхемы, ее неверным выбором в меню или неправильной установкой в панель индицируется красным свечением светодиода «OL» (OverLoad - Перегрузка). Выполнение операции прекращается, светодиод «GO» гаснет, программа выводит на монитор компьютера соответствующее сообщение.

7. Форматы файлов.

Команды «Открыть» и «Сохранить» меню «Файл» управляющей программы работают с файлами прошивок в простом двоичном формате (BIN), т.е. образами микросхем, в которых каждой ячейке микросхемы соответствует один байт файла (для 12-, 14- и 16-разрядных микросхем – два байта). Для работы с файлами в текстовых 16-ричных форматах (Intel HEX, Motorola S19, S28, S37) следует пользоваться командами «Импорт» и «Экспорт» в меню «Файл». Описание опций импорта и экспорта HEX-файлов содержится в разделе «Редактор» меню «Справка».

Следует обратить внимание, что расширение имени файла далеко не всегда указывает на его формат. Расширения bin и hex в большинстве случаев действительно указывают на формат (простой двоичный и Intel HEX, соответственно). Другие расширения могут указывать на происхождение или назначение файла. rom и romr являются сокращениями от ROM map, т.е. карта прошивки. Расширения awd и ami у файлов с прошивками BIOSов компьютеров указывают на авторство BIOSа – компании Award и AMI, соответственно. Мастера по ремонту телевизоров часто обозначают расширением бренды производителей телевизоров, например son, pan, jvc (Sony, Panasonic, JVC), для облегчения поиска нужного файла в большой коллекции прошивок. Прошивки

для микроконтроллеров, созданные компиляторами, иногда имеют расширения `obj` или `tsk`.

Распознать формат файла проще всего, открыв его в любом текстовом редакторе или просмотрщике. HEX-файлы при этом будут отображаться как связный текст – последовательность строк, состоящих из 16-ричных цифр (0...9, A...F), в то время как BIN-файлы будут выглядеть как случайная последовательность прописных и строчных букв различных алфавитов, цифр и знаков препинания. На формат файла также косвенно указывает его размер. BIN-файлы имеют размер, равный объему микросхемы или задействованной части микросхемы. HEX-файлы обычно превышают объем микросхемы примерно в 3,5 раза, если в микросхеме нет незадействованных областей.

Расширения `eep` и `e2p` могут указывать как на назначение прошивки (для EEPROM-памяти данных в микроконтроллере), так и на формат EEP. Этот формат используется некоторыми простыми любительскими программаторами, предназначенными для работы только с популярными микросхемами Serial EEPROM (откуда и образовано название формата). Он представляет собой файл образа ПЗУ в простом двоичном формате, в начало которого добавлена служебная информация от программатора, которым был создан файл. В этой информации можно визуально проследить текстовую часть, содержащую, в частности, название микросхемы, из которой осуществлялось считывание. Для записи микросхемы из такого файла программатором Octopus достаточно открыть файл как простой двоичный и (после выбора типа микросхемы в меню) переместить движок «положение в файле» в окне выбора части вправо до упора.

8. Внутрисхемное программирование.

Под внутрисхемным программированием подразумевают работу с микросхемой без ее удаления из устройства, в котором она установлена. Обычно такой способ работы применяют для впаянных микросхем, чтобы избежать операции выпаивания, или для микросхем, установленных в панели, если требуется очень частое их перепрограммирование, например при отладке программы, хранящейся в этой микросхеме.

Теоретически работать внутрисхемно можно с любой микросхемой, если отключить все ее выводы от схемы для обеспечения отсутствия конфликтов сигналов (например путем перерезания дорожек на плате) и подключить их к панели программатора проводочками. Очевидно, что практического смысла это не имеет, проще микросхему выпаять. Поэтому внутрисхемное программирование применяют лишь для тех микросхем, у которых такой режим специально предусмотрен производителем, максимально облегчен конструкцией микросхемы и явно описан в документации. Обычно этот режим требует подключения микросхемы к программатору всего 2...5 проводами, включая общий провод («землю»). Набор сигналов, необходимых для внутрисхемного программирования зависит от типа микросхемы (определен фирмой-производителем). Их расположение на панели программатора приведено выше в описаниях семейств микросхем.

Общие правила внутрисхемного программирования таковы:

1. Питание микросхемы осуществляется от схемы, где она установлена, а не от программатора. К программатору подключается общий провод («земля») и 1...4 сигнала, по которым осуществляется программирование.
2. Если микросхеме (обычно микроконтроллеру) требуется тактирование, то оно также осуществляется от схемы, где она установлена.
3. Необходимо обеспечить отсутствие конфликтов сигналов на выводах программируемой микросхемы, подключаемых к программатору, одним из следующих способов:
 - неиспользование этих выводов для других целей при разработке схемы и неподключение их к другим компонентам схемы;
 - продуманное использование этих выводов для других целей при разработке схемы, подключение их только к другим компонентам схемы, работающим на вход, а не на выход;
 - использование этих выводов в слаботочных цепях, подключение их к другим компонентам схемы через высокоомные резисторы, устраняющие конфликт с сигналами программатора;
 - механическое отключение путем перерезания дорожек на плате;

- механическое отключение путем сбрасывания заранее предусмотренных в схеме джамперов;
 - электрическое отключение предусмотренными в схеме электронными ключами.
4. Для микросхем с широким диапазоном напряжения питания необходимо обеспечить согласование логических уровней сигналов. Например, если микросхема допускает работу при питании от 2,5 до 5,5 Вольт и в схеме питается от 3 Вольт, нужно установить в настройках программатора такое же напряжение - 3 Вольта. Если та же микросхема стоит в 5-вольтовой схеме, то и в настройках программатора нужно установить 5 Вольт.

9. Обновление ПО.

Обновление ПО программатора позволяет расширять его функциональные возможности, в т.ч. и в первую очередь – пополнять список поддерживаемых микросхем.

Процедура обновления состоит из двух этапов – обновление версии управляющей программы в компьютере и обновление внутреннего ПО программатора.

Новые версии управляющей программы по мере их выпуска производителем размещаются на официальном интернет-сайте ROMSERVICE (www.romservice.ru), в разделе «Техподдержка».

Установку новых версий программы лучше выполнять без предварительного удаления старой версии. В этом случае старая версия будет заменена новой с сохранением ряда настроек программы, сделанных пользователем в процессе эксплуатации предыдущей версии.

Перед установкой новой версии программы, ее следует закрыть, т.к. Windows не позволит осуществить замену компонентов работающей программы.

После запуска новой версии программы, на этапе установки связи с программатором программа обнаружит несоответствие версий и предложит осуществить замену (перепрошивку) внутреннего ПО – микропрограммы контроллера программатора, после выполнения которой Windows может вывести сообщение об обнаружении неизвестного устройства, которое должно исчезнуть само примерно через 1 секунду.

10. Базовый комплект поставки.

- 1) Программатор Octopus.
- 2) Кабель интерфейсный mini-USB.
- 3) Коробка упаковочная.

11. Комплект поставки Octopus+.

- 1) Программатор Octopus.
- 2) Кабель интерфейсный mini-USB 1,8 м.
- 3) Адаптер DIP8 ZIF для серийной работы.
- 4) Адаптер универсальный DP8/DP14-40 (DIP8>>DIP14...DIP40).
- 5) Бокс пластиковый с отделениями для адаптеров.
- 6) Коробка упаковочная.

12. Гарантийные обязательства.

Изготовитель гарантирует нормальную работу прибора в течение 24 месяцев с момента продажи потребителю, а также бесплатный ремонт в течение всего гарантийного срока при условии соблюдения правил пользования, установленных настоящей инструкцией.

13. Возможные проблемы при работе с программатором и способы их решения.

1. Вид окна управляющей программы отличается от описанного в Help. Отсутствует меню микросхем, команда установки связи с программатором недоступна.

Неправильно осуществлен запуск управляющей программы. Для правильного запуска используйте ярлык в разделе «Все программы» главного меню Windows, вызываемого кнопкой «Пуск».

2. Неустойчивое считывание информации из микросхемы.

Если повторные подсчеты контрольной суммы и считывание данных дают различные результаты при заведомо исправной микросхеме, то причиной может являться плохой контакт микросхемы с панелью программатора вследствие окисления или загрязнения выводов.

Следует очистить выводы микросхемы с использованием лезвия ножа, наждачной бумаги или ластика, в зависимости от степени загрязнения.

Не рекомендуется применение для очистки выводов микросхем в корпусах SMD («планарных»: TSOP, SOIC, TQFP и пр.) спирта и других растворителей, включая специальные жидкости для очистки контактов. В состав пластика, из которого изготовлены корпуса микросхем, и маркировочной краски могут входить компоненты, которые, частично растворяясь в растворителе, образуют лак низкой концентрации, покрывающий выводы тончайшим невидимым слоем, что только вредит хорошему контакту.

3. Не удается считать данные из микроконтроллера.

Если данные из микроконтроллера считываются устойчиво, но не похожи на реальные – все ячейки содержат одно и то же число (00, FF или другое), числа с нулями в старших разрядах (например 0005, 000E) или представляют собой последовательные числа (00,01,02,03...), микроконтроллер, скорее всего, защищен от чтения и копирования. Большинство микроконтроллеров позволяют считать состояние защиты и убедиться, что она установлена.

4. Микросхема считывается верно, но не стирается и не записывается. В микросхеме остаются прежние данные.

Микросхема, скорее всего, защищена от перезаписи. Следует снять защиту в соответствии с документацией на микросхему. Порядок управления защитой отличается у разных микросхем. После перезаписи защиту можно вернуть в прежнее состояние.