

Полное руководство пользователя



ELNEC s. r. o.
Москва, Россия – Presov, Slovakia
Июнь 2005



Данное руководство пользователя содержит информацию о следующих программаторах, выпускаемых в настоящее время компанией ELNEC:

JetProg

Универсальный программатор на основе аппаратного драйвера на 48 выводов с возможностью модернизации до 256 выводов.

BeeProg

Универсальный программатор на основе аппаратного драйвера на 48 выводов с интерфейсом USB/LPT и поддержкой режима ISP.

LabProg+

Универсальный программатор на основе аппаратного драйвера на 48 выводов.

SmartProg2

Универсальный программатор на основе аппаратного драйвера на 40 выводов с интерфейсом USB и поддержкой режима ISP.

MEMprog

Специализированный программатор для всех серий микросхем памяти.

MEMprogL

Специализированный программатор для всех серий микросхем памяти.

T51prog

Специализированный программатор для микроконтроллеров серий MCS51 и Atmel AVR с поддержкой режима ISP.

PIKprog+

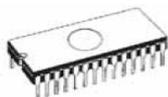
Специализированный программатор микроконтроллеров всех серий Microchip PICmicro с поддержкой режима ISP.

PIKprog

Специализированный программатор микроконтроллеров серий Microchip PICmicro.

SEEpog

Специализированный программатор микросхем памяти серий Serial EEPROM.



COPYRIGHT © 1997 – 2005
ELNEC s. r. o.

This document is copyrighted by ELNEC s. r. o., Presov, Slovakia. All rights reserved. This document or any part of it may not be copied, reproduced or translated in any form or in any way without the prior written permission of ELNEC s. r. o.

The control program is copyright ELNEC s.r.o., Presov, Slovakia. The control program or any part of it may not be analyzed, disassembled or modified in any form, on any medium, for any purpose.

Information provided in this manual is intended to be accurate at the moment of release, but we continuously improve all our products. Please consult manual on www.elnec.com.

ELNEC s.r.o. assumes no responsibility for misuse of this manual.

ELNEC s.r.o. reserves the right to make changes or improvements to the product described in this manual at any time without notice. This manual contains names of companies, software products, etc., which may be trademarks of their respective owners. ELNEC s.r.o. respects those trademarks.

ZLI-0017E

АВТОРСКОЕ ПРАВО © 1997 – 2005
ELNEC s. r. o.

Этот документ защищен авторским правом компании ELNEC s.r.o., Presov, Slovakia. Все права защищены. Этот документ или любая его часть не могут быть скопированы, воспроизведены или переведены в любой форме без письменного разрешения компании ELNEC s.r.o.

Управляющая программа защищена авторским правом компании ELNEC s.r.o., Presov, Slovakia. Управляющая программа или любая ее часть не могут быть анализированы, декомпилированы или модифицированы в любой форме и для любых целей.

Информация представленная в данном документе актуальна на текущий момент. Компания ELNEC s.r.o. постоянно принимает все меры к улучшению качества своей продукции. Для получения последних версий документа посетите сайт компании в Интернет: www.elnec.com.

Компания ELNEC s.r.o. не несет никакой ответственности за последствия неправильного применения данного документа.

Компания ELNEC s.r.o. оставляет за собой право изменения любой части данного документа, управляющей программы и оборудования, описанного в данном документе без какого-либо предварительного уведомления. Данный документ содержит названия компаний, программных продуктов и иных предметов, которые могут быть товарными знаками этих компаний.

ZLI-0017E

Как пользоваться руководством

Руководство пользователя содержит информацию о том как использовать приобретенный программатор и установить программное обеспечение на компьютер. Принято считать, что пользователь программатора имеет базовые навыки по подключению аппаратной части периферийных устройств к компьютеру и установке программного обеспечения.

Уважаемый покупатель!

Благодарим Вас за приобретение программатора компании ELNEC.

После установки программного обеспечения настоятельно рекомендуется в первую очередь ознакомиться со справочными материалами, находящимися в комплекте с программным обеспечением, так как руководство пользователя может содержать устаревшую на настоящий момент информацию.

Замечание: Руководство пользователя содержит информацию обо всех программаторах, выпускаемых в настоящее время компанией ELNEC. Для получения необходимых сведений по конкретной модели программатора обратитесь к соответствующему разделу руководства.

Руководство пользователя состоит из двух частей:

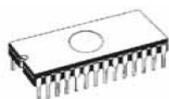
Краткое руководство

Эта часть руководства предназначена для опытных пользователей. В ней находится краткая информация об установке программного обеспечения и основных приемах работы с программатором. Для получения более полной информации следует обратиться к разделам '**Полное руководство**' или '**Устранение неисправностей**'.

Полное руководство

Эта часть руководства предназначена для менее опытных пользователей, а также пользователей, желающих получить дополнительные сведения о программаторе или программном обеспечении.

Последняя версия руководства пользователя размещена для свободного доступа в Интернет, на сайте компании ELNEC, по адресу: www.elnec.com

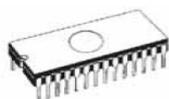


СОДЕРЖАНИЕ

КАК ПОЛЬЗОВАТЬСЯ РУКОВОДСТВОМ.....	4
<i>КРАТКИЙ ОБЗОР</i>	9
КОМПЛЕКТ ПОСТАВКИ	13
СИСТЕМНЫЕ ТРЕБОВАНИЯ.....	14
<i>КРАТКОЕ РУКОВОДСТВО</i>	15
<i>ПОЛНОЕ РУКОВОДСТВО</i>	17
<i>JETPROG</i>	18
ВВЕДЕНИЕ	19
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА.....	23
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ.....	24
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ.....	26
ВНУТРИСХЕМНОЕ ПРОГРАММИРОВАНИЕ (ISP)	27
САМОДИАГНОСТИКА И КАЛИБРОВКА	30
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	31
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ.....	33
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.....	36
<i>BEEPLOG</i>	39
ВВЕДЕНИЕ	40
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА.....	43
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ.....	45
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ.....	47
ВНУТРИСХЕМНОЕ ПРОГРАММИРОВАНИЕ (ISP)	48
МУЛЬТИПРОГРАММИРОВАНИЕ	50
САМОДИАГНОСТИКА И КАЛИБРОВКА	51
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	52
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ.....	54
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.....	57
<i>LABPROG+</i>	60
ВВЕДЕНИЕ	61
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА.....	64
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ.....	65
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ.....	67
САМОДИАГНОСТИКА И КАЛИБРОВКА	68
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	69
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ.....	69
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.....	72
<i>SMARTPROG2</i>	75
ВВЕДЕНИЕ	76
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА.....	78
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ.....	80
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ.....	81
ВНУТРИСХЕМНОЕ ПРОГРАММИРОВАНИЕ (ISP)	82
САМОДИАГНОСТИКА И КАЛИБРОВКА	84
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	85
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ.....	86
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.....	88



<i>MEMPROG</i>	91
ВВЕДЕНИЕ	92
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА	94
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ	95
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ	97
САМОДИАГНОСТИКА И КАЛИБРОВКА	98
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	99
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ	100
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	101
<i>MEMPROGL</i>	104
ВВЕДЕНИЕ	105
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА	107
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ	108
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ	110
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	111
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ	112
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	113
<i>T51PROG</i>	116
ВВЕДЕНИЕ	117
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА	119
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ	120
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ	122
ВНУТРИСХЕМНОЕ ПРОГРАММИРОВАНИЕ (ISP)	123
САМОДИАГНОСТИКА И КАЛИБРОВКА	125
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	126
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ	127
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	128
<i>PIKPROG+</i>	131
ВВЕДЕНИЕ	132
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА	135
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ	136
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ	138
ВНУТРИСХЕМНОЕ ПРОГРАММИРОВАНИЕ (ISP)	139
САМОДИАГНОСТИКА И КАЛИБРОВКА	141
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	142
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ	143
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	144
<i>PIKPROG</i>	147
ВВЕДЕНИЕ	148
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА	149
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ	150
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ	152
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	153
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ	153
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	154
<i>SEEPROG</i>	156
ВВЕДЕНИЕ	157
ОСНОВНЫЕ ЭЛЕМЕНТЫ ПРОГРАММАТОРА	158
ПОДКЛЮЧЕНИЕ ПРОГРАММАТОРА К КОМПЬЮТЕРУ	159
УСТАНОВКА ПРОГРАММИРУЕМОЙ МИКРОСХЕМЫ	161
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	162
ТИПЫ ПОДДЕРЖИВАЕМЫХ МИКРОСХЕМ	162
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	163



<i>ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ</i>	165
ВВЕДЕНИЕ	166
ГЛАВНОЕ ОКНО УПРАВЛЯЮЩЕЙ ПРОГРАММЫ	168
ЗАРЕЗЕРВИРОВАННЫЕ ФУНКЦИОНАЛЬНЫЕ КЛАВИШИ	169
ФАЙЛ (FILE).....	170
БУФЕР (BUFFER)	176
МИКРОСХЕМА (DEVICE).....	185
ПРОГРАММАТОР (PROGRAMMER).....	214
НАСТРОЙКИ (OPTIONS)	220
ПОМОЩЬ (HELP).....	226
<i>ПРИМЕЧАНИЯ</i>	228
РЕЖИМ КОМАНДНОЙ СТРОКИ УПРАВЛЯЮЩЕЙ ПРОГРАММЫ	229
АППАРАТНАЯ СОВМЕСТИМОСТЬ	231
ВНУТРИСХЕМНОЕ ПРОГРАММИРОВАНИЕ (ISP)	232
ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ.....	245
<i>УСТРАНЕНИЕ НЕИСПРАВНОСТЕЙ</i>	247
КАК ДЕЙСТВОВАТЬ	248
ЧТО ДЕЛАТЬ С НЕПОДДЕРЖИВАЕМОЙ МИКРОСХЕМОЙ	249
КОНТАКТНАЯ ИНФОРМАЦИЯ	249
ГАРАНТИЙНЫЕ ОБЯЗАТЕЛЬСТВА.....	250
ДЕФЕКТНЫЙ ЛИСТОК (РУССКАЯ ВЕРСИЯ).....	251
ДЕФЕКТНЫЙ ЛИСТОК (ENGLISH VERSION)	253
УСЛУГА KEEP-CURRENT	254
УСЛУГА ALGOR	256
ЛИСТОК ЗАПРОСА (РУССКАЯ ВЕРСИЯ).....	256
ЛИСТОК ЗАПРОСА (ENGLISH VERSION).....	257
РЕГИСТРАЦИОННАЯ КАРТОЧКА (РУССКАЯ ВЕРСИЯ).....	258
РЕГИСТРАЦИОННАЯ КАРТОЧКА (ENGLISH VERSION).....	259
СЕРТИФИКАТЫ.....	260

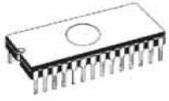
Обозначения принятые в тексте

Пункты меню универсальной управляющей программы отмечены выделенным текстом: **Файл (File)**, **Открыть (Load)**. Сочетания ‘быстрых’ клавиш отмечены выделенным текстом и заключены в угловые скобки: **<Ctrl+F1>**, **<F4>**.

Термины и сокращения принятые в тексте

В руководстве используется ряд технических терминов и сокращений:

Микросхема, ИС (Device)	любая поддерживаемая микросхема, которая может быть запрограммирована с помощью программатора.
Гнездо с нулевым усилением (ZIF socket)	основной разъем программатора для установки адаптеров или программируемых ИС.
Буфер (Buffer)	область памяти управляющей программы, предназначенная для временного хранения данных.
Порт принтера (Printer port)	один из типов интерфейсных разъемов компьютера.
HEX–формат данных (HEX data format)	один из форматов представления файлов данных, который может быть открыт текстовыми редакторами. Информация представлена с помощью символов шестнадцатеричной системы исчисления (5АН). Одна строка файла содержит адрес размещения данных в памяти и сами данные. Целостность файла может быть проверена путем подсчета контрольной суммы.



Краткий обзор

Руководство пользователя содержит информацию о следующих программаторах компании ELNEC: **JetProg**, **BeeProg**, **LabProg+**, **SmartProg2**, **MEMprog**, **MEMprogL**, **T51prog**, **PIKprog+**, **PIKprog** и **SEEpog**.

JetProg – универсальный программатор и тестер логических ИС, основанный на аппаратном драйвере на 48 выводов с возможностью модернизации до 256 выводов. Список ИС, поддерживаемых этим программатором, может быть легко расширен.

BeeProg – универсальный программатор и тестер логических ИС с интерфейсом USB/LPT, основанный на аппаратном драйвере на 48 выводов. Программатор имеет поддержку режима ISP. Список ИС, поддерживаемых этим программатором, может быть легко расширен. Этот программатор по настоящему универсален для своей низкой цены в подобном классе программаторов других компаний–производителей.

LabProg+ – универсальный программатор и тестер логических ИС, основанный на аппаратном драйвере на 48 выводов. Список ИС, поддерживаемых этим программатором, может быть легко расширен.

SmartProg2 – универсальный программатор с интерфейсом USB, основанный на аппаратном драйвере на 40 выводов. Программатор имеет поддержку режима ISP.

MEMprog – специализированный программатор всех типов микросхем памяти, включая EPROM, EEPROM, Flash EPROM, NVRAM, Serial EEPROM. Программатор позволяет тестировать микросхемы памяти Static RAM.

MEMprogL – специализированный программатор всех типов 8–разрядных микросхем памяти, включая EPROM, EEPROM, Flash EPROM, NVRAM, Serial EEPROM. Программатор позволяет тестировать микросхемы памяти Static RAM.

T51prog – специализированный программатор для микроконтроллеров серий MCS51 и Atmel AVR с поддержкой режима ISP. Программатор поддерживает алгоритмы работы с микросхемами памяти I²C (24Cxx), SPI (25Cxx) и Microware (93Cxx).

PIKprog+ – специализированный программатор микроконтроллеров всех серий Microchip PICmicro с поддержкой режима ISP. Программатор поддерживает алгоритмы работы с микросхемами памяти I²C (24Cxx), SPI (25Cxx) и Microware (93Cxx).



PIKprog – специализированный программатор микроконтроллеров серий Microchip PICmicro – младшая модель программатора PIKprog+. Программатор поддерживает алгоритмы работы с микросхемами памяти I²C (24Cxx), SPI (25Cxx) и Microware (93Cxx).

SEEplog – специализированный программатор микросхем памяти серий Serial EEPROM. Программатор поддерживает алгоритмы работы с микросхемами памяти I²C, SPI, Microware и алгоритмы работы цифровыми устройствами (цифровые термометры). Также имеется поддержка работы с микросхемами с пониженным напряжением питания (LV 3,3V).

Все программаторы легко подключаются к компьютерам аналогичным IBM PC без использования каких-либо дополнительных адаптеров или плат расширения – требуется свободный порт USB или LPT.

Универсальная управляющая программа, предназначенная для работы со всеми моделями программаторов, корректно работает в среде любой операционной системы MS Windows 95/98/Me/NT/2000/XP.

Жесткий контроль готовой продукции и использование оригинальных комплектующих в производстве программаторов позволили компании ELNEC предоставить гарантийное обслуживание своей продукции в течение трех лет (только для продукции приобретенной в компании ELNEC). Гарантийный срок работы гнезда программаторов ZIF socket определен изготовителем в 25'000 срабатываний.

Бесплатная поддержка пользователей включает в себя:

бесплатная техническая консультация (телефон/факс/электронная почта);
бесплатное обновление программного обеспечения через Интернет.

Бесплатное программное обеспечение всегда размещено для свободного доступа в Интернет, на сайте компании ELNEC, по адресу:

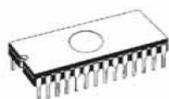
www.elnec.com



Компания ELNEC также предоставляет пользователям продукции новые дополнительные услуги:

- **Keep-Current** – платная услуга, позволяющая пользователям незамедлительно получать самые последние версии программного обеспечения и документацию, а также всегда быть в курсе всех новостей и событий компании ELNEC;
- **AlgOR** – услуга, позволяющая по запросу в компанию получить поддержку работы программного обеспечения с ИС, которой еще нет в списке поддерживаемых ИС.

***Замечание:** Компания ELNEC не рекомендует использовать программаторы LabProg+, PREPROM-02aLV, MEMprog, MEMprogL, 51&AVRprog, PIKprog и SEEprog в непредусмотренном для них режиме ISP.*



Комплект поставки

	JetProg	BeeProg	LabProg+	SmartProg2	MEMprog	MEMprogL	T51prog	PIKprog+	PIKprog	SEEprog
Программатор	•	•	•	•	•	•	•	•	•	•
Кабель LPT	•	•	•	–	•	•	•	•	•	•
Кабель USB	–	•	–	•	–	–	–	–	–	–
Сетевой адаптер	•	•	•	•	•	•	•	•	•	•
Диагностический модуль (POD)	•	•	•	•	•	–	•	•	–	–
Кабель ISP	–	•	–	•	–	–	•	•	–	–
Чехол для гнезда ZIF	•	•	•	•	•	–	•	•	–	–
Руководство пользователя	•	•	•	–	–	–	–	–	–	–
Краткое руководство	–	–	–	•	•	•	•	•	•	•
Регистрационная карточка	•	•	•	•	•	•	•	•	•	•

Если комплект поставки Вашего программатора отличается от приведенных выше данных, свяжитесь с Вашим торговым агентом.

Системные требования

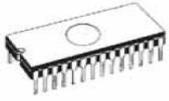
Минимальные системные требования

- Microsoft Windows® 95, NT или старше;
- PC Pentium II 300;
- 64 МВ оперативной памяти;
- 60 МВ свободного места на диске;
- LPT порт (для программаторов, подключаемых к порту LPT);
- USB порт версии 1.1 или старше (для программаторов, подключаемых к порту USB).

Рекомендуемые системные требования

- Microsoft Windows® XP;
- PC Pentium 4;
- 256 МВ оперативной памяти;
- 150 МВ свободного места на диске;
- LPT порт с поддержкой режима EPP/ЕСР (для программаторов, подключаемых к порту LPT);
- USB порт версии 1.1 или старше (для программаторов, подключаемых к порту USB).

***Замечание:** Если компьютер имеет всего один порт LPT, рекомендуется приобрести и установить дополнительную мультиплату с портом LPT для одновременного использования программатора и печатающего устройства.*



Краткое руководство

Подключение программатора

- отключите питание компьютера;
- подключите интерфейсный разъем программатора к порту LPT компьютера с помощью интерфейсного кабеля;
- включите питание компьютера;
- включите питание программатора.

Установка программного обеспечения

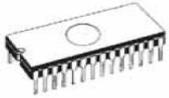
Откройте содержимое диска с программным обеспечением и запустите файл 'Setup.exe'. Следуйте предлагаемым инструкциям. Для получения последней версии программного обеспечения обратитесь на сайт компании ELNEC, по адресу: www.elnec.com.

Управляющая программа

Запустите файл 'PG4UW.exe'. Меню **Микросхема (Device)** содержит команды для работы с текущими ИС. Меню **Файл (File)** содержит команды для работы с файлами. Меню **Буфер (Buffer)** содержит команды для работы с буфером управляющей программы.

Программирование микросхемы

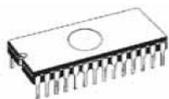
Используйте сочетание клавиш <Alt+F5> для доступа к меню выбора необходимой микросхемы по типу, производителю или по маске. Для чтения содержимого памяти микросхемы в буфер управляющей программы, поместите ИС в гнездо программатора ZIF и нажмите клавишу <F7>. Для программирования микросхемы данными из файла нажмите клавишу <F3> и выберите необходимый файл данных – он будет загружен в буфер управляющей программы. Далее установите микросхему в гнездо программатора ZIF. Для проверки очистки памяти микросхемы нажмите клавишу <F6>, а для программирования микросхемы нажмите клавишу <F9>. После окончания процесса программирования можно верификацию микросхемы – проверить качество программирования. Для этого нажмите клавишу <F8>.



Полное руководство

JetProg





Введение

JetProg – это представитель новой линейки универсальных программаторов, разработанных компанией ELNEC и использующих в основе своей работы различные алгоритмы программирования. Универсальная управляющая программа поддерживает корректную работу программатора в среде любой операционной системы MS Windows 95/98/Me/NT/2000/XP.

Программатор поддерживает практически все типы программируемых микросхем существующие на настоящий момент, что делает его наиболее конкурентоспособным при данном соотношении цены и качества в своем классе, одновременно являясь надежным и простым в повседневной эксплуатации инструментом.

Подключение программатора осуществляется к любым компьютерам семейства IBM PC 486 или более современным, оборудованным стандартным параллельным портом (LPT) без применения специальных адаптеров или дополнительных плат расширения.

Для получения максимальной производительности рекомендуется использовать параллельный порт компьютера в стандарте IEEE–1284 (режим ECP/EPP). Это позволяет значительно сократить время программирования микросхем. Например микросхема AT29C040A может быть запрограммирована за 28 секунд.

Основная конфигурация программатора JetProg включает в себя следующие части:

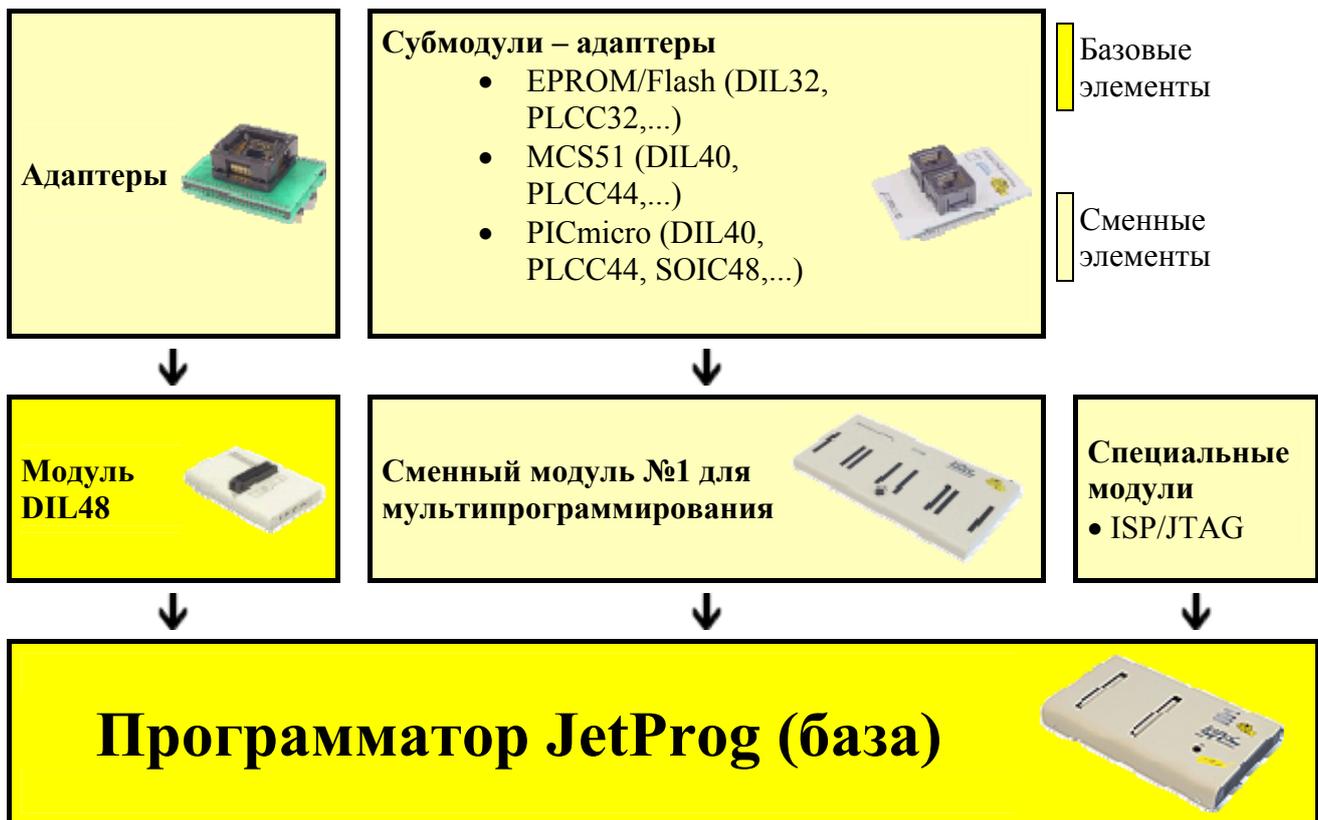
- база программатора JetProg;
- модуль DIL48.

Программатор основан на аппаратном драйвере на 48 выводов с возможностью модернизации до 256 выводов с помощью специального сменного модуля. Аппаратный драйвер сменного модуля исключает повреждение программируемых микросхем при установке и позволяет работать с микросхемами поддерживающими рабочее напряжение вплоть до 1,8V.

Модульная структура программатора позволяет приспособить его к конкретным потребностям – будь то многопоточное производство или исследовательская лаборатория.

Аппаратная часть программатора содержит мощные TTL–совместимые буферы, обеспечивающие выбор Н/Л логических уровней а также режимов PU/PD на любом контакте гнезда программатора.

Структурная схема программатора JetProg

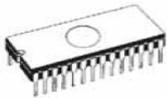


Современные быстродействующие буферы программатора обеспечивают надежную работу на самых высоких скоростях для всех поддерживаемых типов микросхем.

Эти особенности позволяют легко приспособить аппаратную часть программатора к работе с новыми микросхемами путем обновления управляющей программы.

JetProg является не только программатором микросхем. Он является еще и тестировщиком логических ИС TTL/CMOS и микросхем памяти. Алгоритмы тестирования могут быть изменены для конкретного семейства ИС.

Программатор представляет собой полностью законченную микропроцессорную систему, предназначенную для многократного программирования однотипных данных в серию микросхем.



Программатор практически независим от состояния компьютера и позволяет продолжить программирование даже если от компьютера не поступает информация (отказ оборудования компьютера).

Перед началом процесса программирования программатор автоматически проводит процедуру по проверке правильности установки программируемой микросхемы в гнездо и наличие надежного контакта со всеми выводами микросхемы. Также доступна функция чтения идентификатора ИС для определения параметров программирования. Эти функции позволяют избежать необратимого повреждения программируемой микросхемы из-за ошибок, допущенных пользователем при ее установке в программатор или выбора неправильного режима программирования.

Программатор имеет обширные инструменты для самодиагностики, позволяющие проверить работу интерфейсной части, формирование уровней напряжений и задержек сигналов.

Оптимальное проектирование конструкции программатора повышает надежность его эксплуатации, исключая потерю контактов в разъемах, наводки помех по цепям питания или провалы напряжений на выводах гнезда программатора. Все входные и выходные цепи программатора, гнезда разъемы имеют защиту от повреждения статическими разрядами (ESD protect).

Проверка качества записи выбранной микросхемы проводится при предельно допустимых минимальном и максимальном напряжениях питания, что позволяет в дальнейшем гарантировать надежную работу запрограммированной микросхемы в самых сложных условиях эксплуатации.

Адаптеры позволяют работать с микросхемами в различных корпусах: PLCC, SOIC, PSOP, SSOP, TSOP, TSSOP, TQFP, QFN (MLF), SDIP, BGA и других типах корпусов. Для работы с микросхемами имеющими более 48 выводов используются специальные субмодули-адаптеры.

Интерфейс управляющей программы включает в себя несколько панелей инструментов, на которых расположены кнопки-пиктограммы, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка). Для облегчения работы так же предусмотрена поддержка функциональных клавиш и их сочетаний. В большинстве случаев возможно использование встроенной контекстной подсказки.

Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

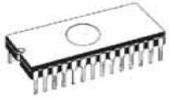
Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера. Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

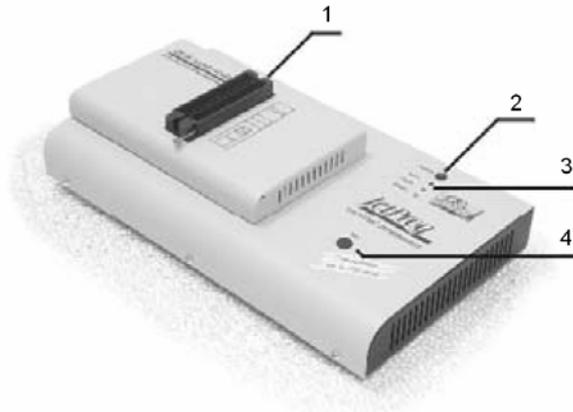
Поиск и выбор необходимой микросхемы производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных (по маске).

Работа с новыми ИС может быть уже начата после простого обновления текущей версии программного обеспечения – модернизация аппаратной части не требуется.

Жесткий контроль готовой продукции и использование оригинальных комплектующих в производстве программаторов позволили компании ELNEC предоставить гарантийное обслуживание своей продукции в течение трех лет (только для продукции приобретенной в компании ELNEC).



Основные элементы программатора

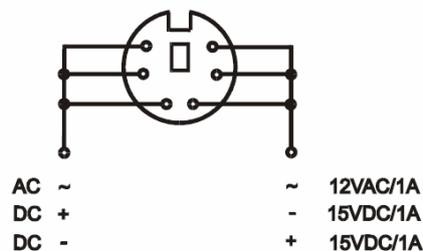


1. Модуль DIL48 с колодкой ZIF socket;
2. Индикатор питания;
3. Индикаторы состояния программатора;
4. Кнопка функции YES!



5. Интерфейсный разъем для подключения кабеля LPT;
6. Разъем питания для подключения сетевого адаптера;
7. Системный разъем.

Расположение контактов разъема питания программатора:



Замечание: Из-за низкого энергопотребления в неактивном состоянии (индикатор питания 2 не активен), программатор не имеет выключателя питания.

Подключение программатора к компьютеру

***Замечание:** При подключении программатора, строго следуйте данной инструкции. Это позволит избежать поломок и выхода из строя Вашего персонального компьютера и программатора. Нельзя использовать электронные или механические переключатели параллельного порта (LPT switches) для одновременной работы с программатором и печатающим устройством. Для подключения печатающего устройства установите в компьютер плату расширения с дополнительным параллельным портом (LPT). Плата расширения приобретается отдельно.*

Выключите питание персонального компьютера и отсоедините шнур питания от сети 220V. Сетевой адаптер программатора также должен быть отключен от питающей сети.

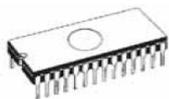
Подключите разъем интерфейсного кабеля, находящегося в комплекте с программатором к разъему программатора (5). Противоположный разъем интерфейсного кабеля подключите к свободному параллельному порту (LPT) компьютера. Обязательно закрепите предохранительные винты на разъемах интерфейсного кабеля – это позволит избежать выпадения кабеля из разъема и исключит возможные проблемы при работе с программатором.

Если компьютер оборудован единственным параллельным портом и к нему подключено печатающее или иное устройство, необходимо выключить печатающее устройство, отсоединить шнур питания от сети 220V, а затем отключить интерфейсный кабель этого устройства от компьютера.

Подключите разъем сетевого адаптера к гнезду питания программатора (6) и включите сетевой адаптер в питающую сеть. Наличие питания программатора можно проверить по активному состоянию индикатора питания (2). При этом индикаторы состояния программатора (3) будут активны в течение короткого промежутка времени. Программатор готов к работе после того как индикаторы состояния программатора (3) перейдут в неактивное состояние.

Подключите компьютер к сети 220V и включите его. После загрузки операционной системы компьютера, запустите универсальную управляющую программу.

***Замечание:** Для исключения случаев повреждения порта LPT компьютера, не рекомендуется подключение и отключение интерфейсного кабеля при включенном компьютере.*



Если по каким-то причинам требуется подключить программатор к работающему компьютеру, воспользуйтесь нижеприведенной рекомендацией:

Для подключения программатора к работающему компьютеру, **вначале** к параллельному порту (LPT) компьютера **подключается интерфейсный кабель** программатора, а **затем** к питающей сети **подключается сетевой адаптер** программатора.

Если по каким-то причинам требуется отключить программатор от работающего компьютера, воспользуйтесь нижеприведенной рекомендацией:

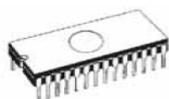
Для отключения программатора от работающего компьютера, **вначале** **отключается сетевой адаптер** программатора от питающей сети, а **затем** **отключается интерфейсный кабель** программатора от параллельного порта (LPT) компьютера.

Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме–ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.



Внутрисхемное программирование (ISP)

Для использования программатора в режиме внутрисхемного программирования необходимо заменить модуль DIL48 на специальный модуль ISP/JTAG.

Аппаратный драйвер специального модуля ISP/JTAG исключает повреждение программируемых микросхем при установке и позволяет работать с микросхемами поддерживающими рабочее напряжение вплоть до 1,8V.

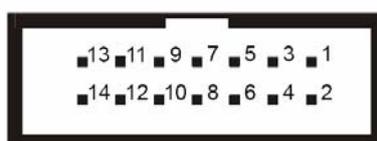
Проверка качества записи микросхемы проводится при предельно допустимых минимальном и максимальном напряжениях питания, что позволяет в дальнейшем гарантировать надежную работу запрограммированной микросхемы в самых сложных условиях эксплуатации.

Конструкция программатора обеспечивает корректное питание программируемой системы в режиме ISP.

Справочная система управляющей программы предоставляет исчерпывающую информацию о режиме ISP для выбранной микросхемы и отображает в реальном времени процесс программирования.

Дополнительная информация о программировании в режиме ISP доступна в разделе руководства '**Схемная реализация режима ISP**'.

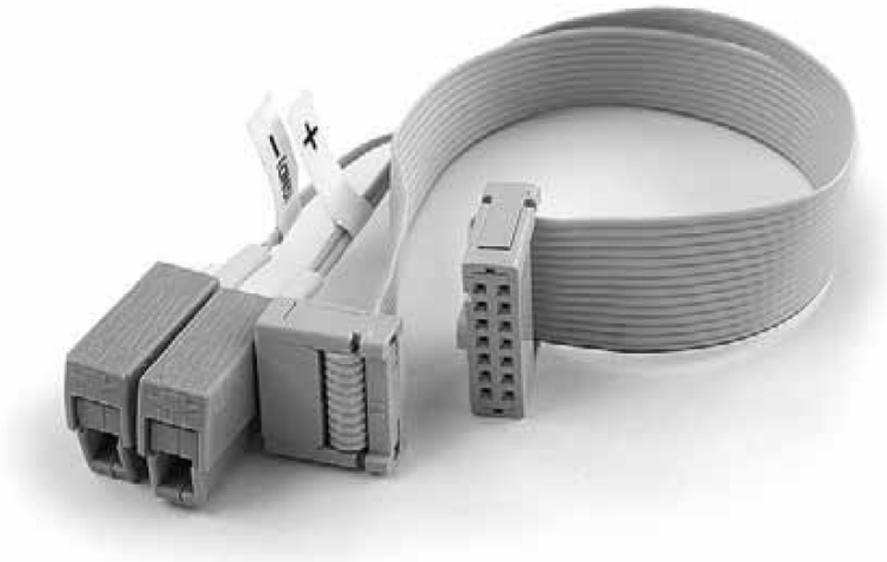
Нумерация контактов разъема ISP (вид спереди):



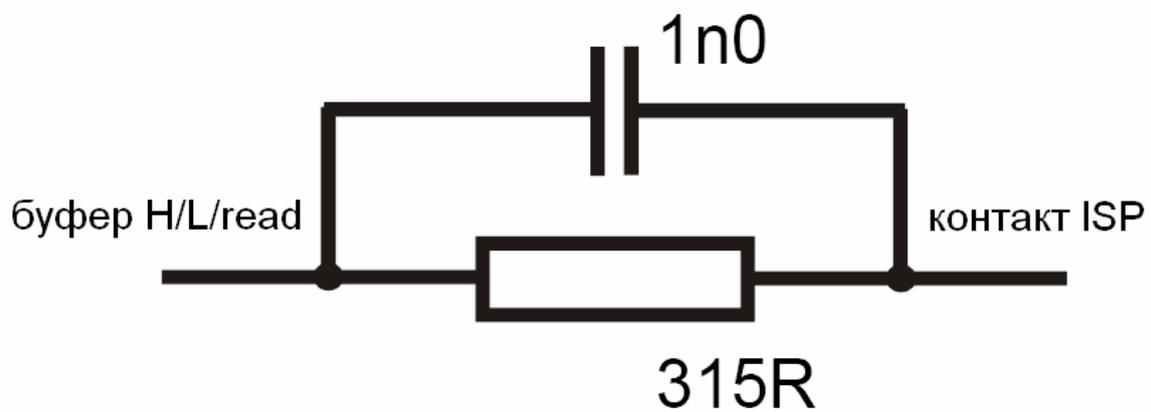
Назначение контактов разъема ISP зависит в каждом конкретном случае от выбранной для программирования микросхемы (режим ISP для данного типа микросхемы устанавливается при его выборе из списка поддерживаемых микросхем). Более подробная информация может быть получена в меню **Микросхема–Информация о ИС (Device–Device Info)** управляющей программы или с помощью сочетания клавиш <Ctrl+F1>.

***Замечание:** Контакт №1 на кабеле для внутрисхемного программирования помечен символом '▲'.*

Кабель ISP для программатора JetProg:

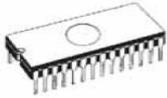


Принципиальная схема построения узла ISP для контактов №№ 3 и 10 (буферы H/L/read):



Принципиальная схема построения узла ISP для контактов №№ 2, 4, 6 и 8 (буферы H/L/read):

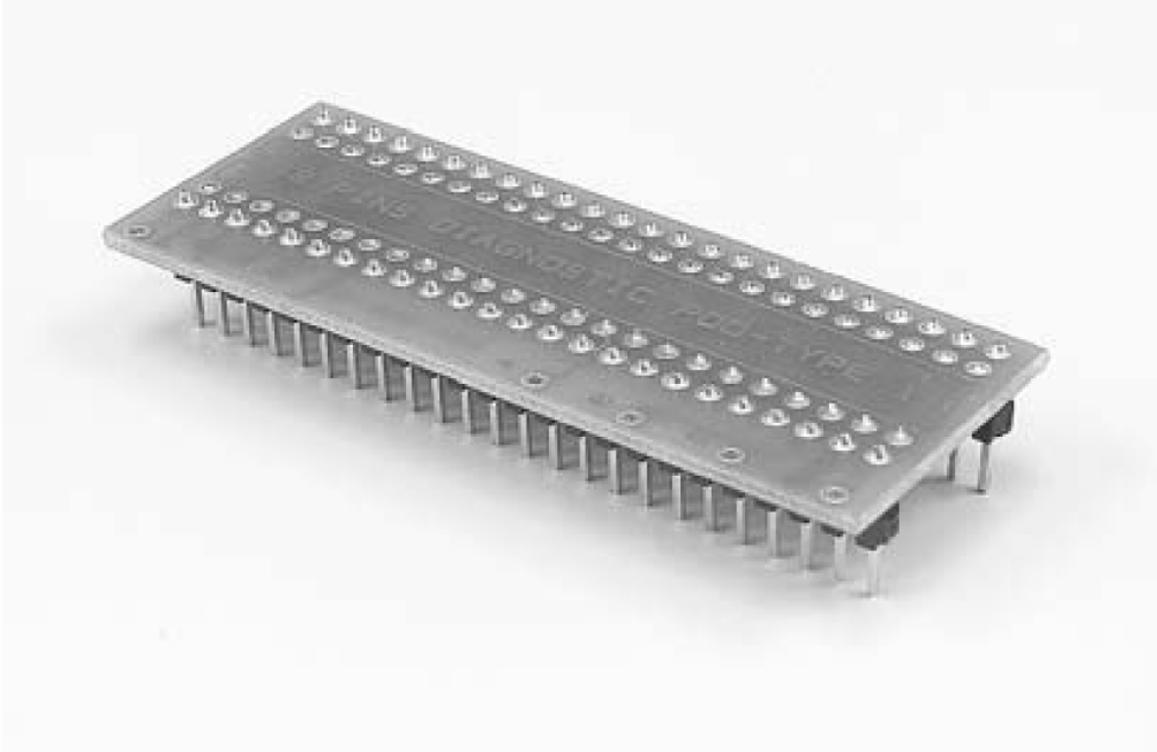




Для исключения повреждения и выхода из строя программируемой микросхемы при использовании режима ISP, выполняйте следующие правил:

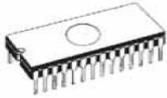
- используйте кабель ISP входящий в комплект программатора. Применение кабелей иной длины или кабелей, изготовленных из других материалов, может привести к нестабильному программированию или полной невозможности программирования микросхемы Вашим программатором;
- программатор поддерживает программирование микросхем (контакт № 1 разъема ISP) и ограниченных программируемых систем (контакты №№ 5, 13, 14 разъема ISP), однако программируемые системы не поддерживают программатор;
- программатор контролирует величину напряжения программирования для выбранной микросхемы. Процесс программирования не будет выполняться, если конструкция программируемой системы оказывает влияние на величину напряжения программирования.

Самодиагностика и калибровка



При возникновении подозрений в некорректной работе программатора, рекомендуется использовать диагностический модуль (POD), входящий в комплект поставки программатора, для проведения процедур тестирования программатора и его калибровки. Тестирование и калибровку рекомендуется проводить каждые 6 месяцев.

Различные способы и методы тестирования и калибровки программатора доступны в меню **Программатор (Programmer)** управляющей программы.



Технические характеристики

База программатора, DAC

- максимальная скорость передачи данных интерфейсной части программатора, совместимой со стандартом IEEE–1284, до 1 Mbit/s;
- самостоятельный 20 MHz – центральный процессор программатора;
- три цифро–аналоговых преобразователя (DAC) для управления напряжениями VCCP, VPP1 и VPP2;
- диапазон напряжения VCCP 0...8V/1A;
- диапазон напряжения VPP1, VPP2 0...26V/1A;
- автокалибровка;
- самотестирование;
- защита от повреждения статическими разрядами (ESD protect) на интерфейсном разъеме и разъеме питания;
- возможность подключения антистатического браслета для защиты от повреждения статическими разрядами (ESD protect).

Дополнительные субмодули–адаптеры

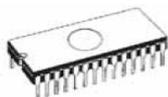
- аппаратный драйвер на 48 выводов с возможностью модернизации до 256 выводов;
- переназначение напряжений VCC, VPP1 и VPP2 на любой вывод субмодуля;
- переназначение нулевого потенциала на любой вывод субмодуля;
- мощные TTL–совместимые буферы, обеспечивающие выбор H/L/CLK логических уровней а также режимов PU/PD на любом выводе субмодуля;
- аналоговые буферы обеспечивающие уровень напряжения 1,8...26V на любом выводе субмодуля;
- защита от повреждения статическими разрядами (ESD protect) на любом выводе субмодуля (стандарт IEC1000–4–2: 15kV воздушный зазор, 8kV контакт на вывод);
- непрерывное тестирование перед каждой операцией программирования.

Модуль DIL48

- 48–контактное гнездо модуля с нулевым усилением (ZIF socket) форм–фактора DIL для установки любых микросхем с количеством выводов до 48 и расстоянием между рядами выводов в диапазоне 300–600 mil.

Модуль ISP/JTAG

- 14–контактный разъем, с ключом, предотвращающим неправильную установку кабеля ISP;
- 6 TTL–совместимых буферов обеспечивают выбор H/L/CLK логических уровней а также режимов PU/PD. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV);
- диапазон напряжения VCCP 2...7V/100mA;
- диапазон напряжения VPP1, VPP2 2...25V/50mA;
- возможность интеллектуального выбора уровня напряжения VCCP;
- поддержка напряжения питания программируемой системы в диапазоне 2...15V/300mA;
- защита от повреждения статическими разрядами (ESD protect) на любом выводе разъема (стандарт IEC1000–4–2: 15kV воздушный зазор, 8kV контакт на вывод).



Типы поддерживаемых микросхем

Модуль DIL48

- EPROM: NMOS/CMOS, 2708 *, 27xxx, 27Cxxx, с 8/16-битной шиной и поддержкой семейств LV;
- EEPROM: NMOS/CMOS, 28xxx, 28Cxxx, 27EExxx, с 8/16-битной шиной;
- Flash EPROM: 28Fxxx, 29Cxxx, 29Fxxx, 29BVxxx, 29LVxxx, 29Wxxx, 49Fxxx, с объемом памяти от 256Kbit до 32Mbit, с 8/16-битной шиной и поддержкой семейств LV;
- Serial E(E)PROM: 24Cxxx, 24Fxxx, 25Cxxx, 45Dxxx, 59Cxxx, 25Fxxx, 25Pxxx, 85xxx, 93Cxxx, NVM3060, MDAxxx, с полной поддержкой семейств LV;
- Configuration (EE)PROM: XCFxxx, XC17xxxx, XC18Vxxx, EPCxxx, AT17xxx, 37LVxx;
- 1-Wire E(E)PROM: DS1xxx, DS2xxx;
- PROM: AMD, Harris, National, Philips/Sigmetics, Tesla, TI;
- NV RAM: Dallas DSxxx, SGS/Inmos MKxxx, SIMTEK STKxxx, XICOR 2xxx, ZMD U63x;
- PLD Altera: MAX 3000A, MAX 7000A, MAX 7000B, MAX 7000S, MAX7000AE;
- PLD Lattice: ispGAL22V10x, ispLSI1xxx, ispLSI1xxxEA, ispLSI2xxx, ispLSI2xxxA, ispLSI2xxxE, ispLSI2xxxV, ispLSI2xxxVE, ispLSI2xxxVL, LC4xxxB/C/V/ZC, M4-xx/xx, M4A3-xx/xx, M4A5-xx/xx, M4LV-xx/xx;
- PLD Xilinx: XC9500, XC9500XL, XC9500XV, CoolRunner XPLA3, CoolRunner-II;
- PLD семейств SPLD/CPLD: AMI, Atmel, AMD-Vantis, Gould, Cypress, ICT, Lattice, NS, Philips, STM, VLSI, TI;
- Микроконтроллеры серии 48: 87x41, 87x42, 87x48, 87x49, 87x50;
- Микроконтроллеры серии 51: 87xx, 87Cxxx, 87LVxx, 89Cxxx, 89Sxxx, 89LVxxx, всех производителей, Philips 87C748...752, Philips серии LPC, Cygnal/Silicon Laborat. C8051;
- Микроконтроллеры Intel серии 196: 87C196 KB/KC/KD/KT/KR/...;
- Микроконтроллеры Atmel AVR: AT90Sxxxx, ATtiny, ATmega;
- Микроконтроллеры Cypress: CY8Cxxxxxx;
- Микроконтроллеры ELAN: EM78Pxxx;
- Микроконтроллеры Microchip PICmicro: PIC10xxx, PIC12xxx, PIC16xxx, PIC17Cxxx, PIC18xxx, dsPIC;
- Микроконтроллеры Motorola: 68HC05, 68HC08, 68HC11;

- Микроконтроллеры National: COP8xxx;
- Микроконтроллеры NEC: uPD78Pxxx;
- Микроконтроллеры Scenix (Uvicom): SXxxx;
- Микроконтроллеры SGS–Thomson: ST6xx, ST7xx, ST10xx;
- Микроконтроллеры TI: MSP430 и MSC121x;
- Микроконтроллеры ZILOG: Z86/Z89xxx и Z8xxx;
- Микроконтроллеры: EM Microelectronic, Fujitsu, Goal Semiconductor, Princeton, Macronix, Winbond, Hitachi, Holtek, Infineon (Siemens), NEC, Samsung, Toshiba.

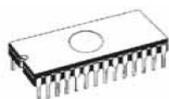
Тестирование микросхем

- TTL: 54, 74 S/LS/ALS/H/HC/HCT;
- CMOS: 4000, 4500;
- Static RAM: 6116...624000;
- Определяемые пользователем алгоритмы тестирования.

Модуль ISP/JTAG

- Serial E(E)PROM: I²C;
- Микроконтроллеры Atmel: AT89Sxxx, AT90Sxxxx, ATtiny, ATmega;
- Микроконтроллеры Cypress: CY8C2xxxx;
- Микроконтроллеры Elan: EM78Pxxx;
- Микроконтроллеры EM Microelectronic: с 4/8–битной шиной;
- Микроконтроллеры Microchip PICmicro: PIC10xxx, PIC12xxx, PIC16xxx, PIC17xxx, PIC18xxx, dsPIC;
- Микроконтроллеры Motorola/Freescale: HC08 GT, LJ, QY, QT;
- Микроконтроллеры Philips: серии LPC;
- Микроконтроллеры TI: MSP430;
- PLD Lattice: ispGAL22xV10x, ispLSI1xxxEA, ispLSI2xxxE, ispLSI2xxxV, ispLSI2xxxVE, ispLSI2xxxVL, M4–xx/xx, M4LV–xx/xx, M4A3–xx/xx, M4A5–xx/xx, LC4xxxB/C/V/ZC;
- Altera: MAX 3000A, MAX 7000A, MAX 7000B, MAX 7000S, MAX 9000, MAX II;
- Xilinx: XC9500, XC9500XL, XC9500XV, CoolRunner XPLA3, CoolRunner–II.

*Микросхемы отмеченные символом * являются устаревшими и программируются с помощью специальных адаптеров.*



Адаптеры

- разнообразные адаптеры для корпусов типов DIP, PLCC, SOIC, PSOP, SSOP, TSOP, TSSOP, TQFP, QFN (MLF), SDIP, BGA и других типоразмеров;
- поддержка всех микросхем в корпусах типоразмера DIP без применения дополнительных адаптеров;
- поддержка всех микросхем в корпусах отличных от типоразмера DIP с количеством выводов более 48 с помощью ряда универсальных адаптеров;
- частичная совместимость с адаптерами сторонних производителей.

Скорость программирования

Микросхема	Действие	Время
27C010	программирование и верификация	21 секунда
AT29C040A	программирование и верификация	31 секунда
AM29F040	программирование и верификация	35 секунд
PIC16C67	программирование и верификация	10 секунд
PIC18F452	программирование и верификация	4 секунды

Замечание: Измерение скорости программирования производилось на тестовом компьютере P4, 2,4 GHz, режим LPT порта ECP, Установленная операционная система MS Windows XP.

Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

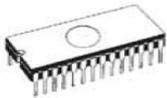
- выбор необходимой микросхемы по типу, производителю или любому фрагменту имеющейся информации;
- автоматическое определение семейства микросхем памяти EPROM/Flash EPROM по идентификатору ID;
- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- редактирование и установка конфигурационных битов;
- тестирование на ошибочные биты;
- подсчет контрольной суммы.

Операции контроля микросхем

- тест на правильную ориентацию микросхемы в гнезде программатора ZIF;
- тест на наличие надежного контакта по всем выводам микросхемы;
- проверка соответствия настроек для выбранной микросхемы по идентификатору.

Специальные операции с микросхемами

- непрерывный режим – начало программирования после смены микросхемы в гнезде программатора ZIF;
- автоматическая генерация идентификаторов (серийных номеров) при программировании группы однотипных микросхем;
- статистика программирования;
- режим счетчика.



Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение.

Поддерживаемые форматы файлов

- Unformatted binary (BIN (raw));
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX;
- Altera POF, JEDEC (версии 3.0.A), ABEL, CUPL, PALASM, TANGO PLD, OrCAD PLD, PLD Designer ISDATA;
- JAM (JEDEC STAPL), JBC (Jam STAPL Byte Code), STAPL (STAPL File) JEDEC стандарта JESD-71;
- VME (ispVME File VME2.0/VME3.0).

Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

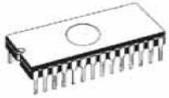
- напряжение питания 12–15V AC или 15–18V DC;
- потребляемый ток 1A;
- максимальная потребляемая мощность 12W в активном режиме;
- максимальная потребляемая мощность 2,5W в неактивном режиме;
- размеры 275x157x58 мм;
- масса без дополнительных адаптеров 1,8 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**

Комплект поставки

- база программатора JetProg;
- модуль DIL48;
- кабель LPT;
- диагностический модуль POD;
- чехол для гнезда ZIF;
- сетевой адаптер питания 100...240V AC/15V DC/1A;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep-Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.



BeeProg



Введение

VeeProg – это представитель новой линейки универсальных программаторов с интерфейсом USB, разработанных компанией ELNEC и использующих в основе своей работы различные алгоритмы программирования. Универсальная управляющая программа поддерживает корректную работу программатора в среде любой операционной системы MS Windows 95/98/Me/NT/2000/XP.

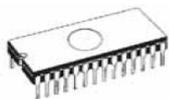
Программатор поддерживает практически все типы программируемых микросхем существующие на настоящий момент, что делает его наиболее конкурентоспособным при данном соотношении цены и качества в своем классе, одновременно являясь надежным и простым в повседневной эксплуатации инструментом.

VeeProg является не только программатором микросхем. Он является еще и тестировщиком логических ИС TTL/CMOS и микросхем памяти. Алгоритмы тестирования могут быть изменены для конкретного семейства ИС.

Подключение программатора осуществляется к любым компьютерам семейства IBM Pentium или более современным, оборудованным стандартным портом USB или стандартным портом LPT. Программатор поддерживает стандарт IEEE–1284 (режим ECP/EPP) для порта LPT. Наличие порта USB/LPT позволяет подключить программатор к любым компьютерам – самым современным или портативным, а также к более старым, без применения специальных адаптеров или дополнительных плат расширения.

Аппаратная часть программатора содержит 48 мощных TTL–совместимых буферов, обеспечивающих выбор Н/Л логических уровней а также режимов PU/PD на любом контакте гнезда программатора. Современные быстродействующие буферы программатора обеспечивают надежную работу на самых высоких скоростях для всех поддерживаемых типов микросхем и позволяют работать с микросхемами поддерживающими рабочее напряжение вплоть до 1,8V.

Перед началом процесса программирования программатор определяет правильность установки микросхемы в гнездо ZIF и наличие надежного контакта со всеми выводами микросхемы. Также доступна функция чтения идентификатора ИС для определения параметров программирования. Эти функции позволяют избежать необратимого повреждения программируемой микросхемы из–за ошибок, допущенных пользователем при ее установке в программатор или выбора неправильного режима программирования.



Все входные и выходные цепи программатора, гнезда разъемы имеют защиту от повреждения статическими разрядами (ESD protect) до 15kV.

Проверка качества записи выбранной микросхемы проводится при предельно допустимых минимальном и максимальном напряжениях питания, что позволяет в дальнейшем гарантировать надежную работу запрограммированной микросхемы в самых сложных условиях эксплуатации.

Дополнительные адаптеры позволяют работать с микросхемами в различных корпусах: PLCC, SOIC, PSOP, SSOP, TSOP, TSSOP, TQFP, QFN (MLF), SDIP, BGA и других типах корпусов.

Интерфейс управляющей программы включает в себя несколько панелей инструментов, на которых расположены кнопки–пиктограммы, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка). Для облегчения работы так же предусмотрена поддержка функциональных клавиш и их сочетаний. В большинстве случаев возможно использование встроенной контекстной подсказки.

Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера. Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

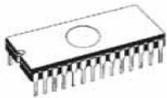
Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

Поиск и выбор необходимой микросхемы производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных (по маске).

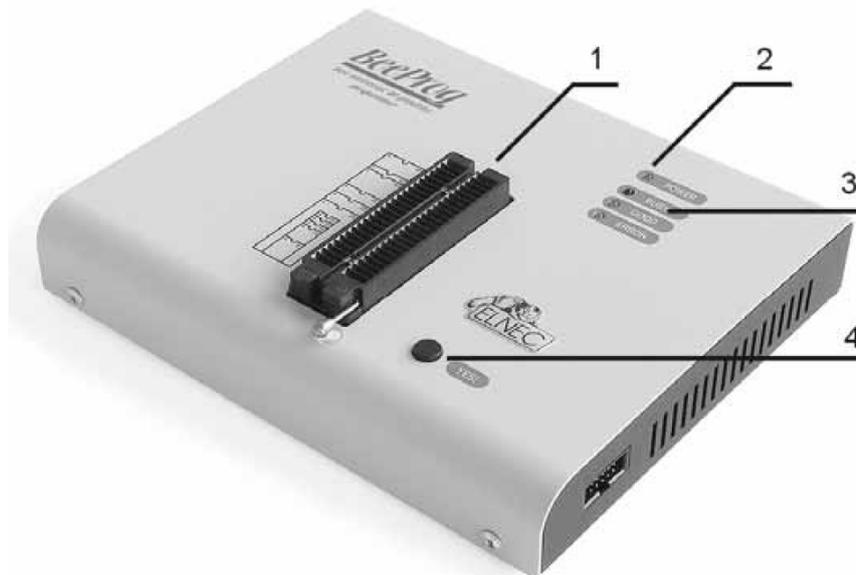


Работа с новыми ИС может быть уже начата после простого обновления текущей версии программного обеспечения – модернизация аппаратной части не требуется.

Жесткий контроль готовой продукции и использование оригинальных комплектующих в производстве программаторов позволили компании ELNEC предоставить гарантийное обслуживание своей продукции в течение трех лет (только для продукции приобретенной в компании ELNEC).



Основные элементы программатора



1. Гнездо ZIF socket на 48 контактов;
2. Индикатор питания;
3. Индикаторы состояния программатора;
4. Кнопка функции YES!

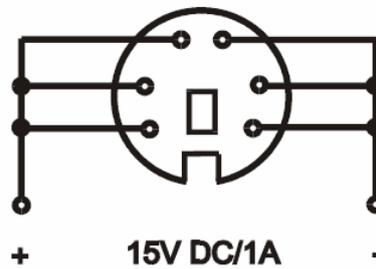


5. Интерфейсный разъем для подключения кабеля LPT;
6. Интерфейсный разъем для подключения кабеля USB;
7. Разъем питания для подключения сетевого адаптера.

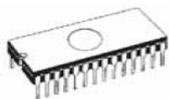


8. Разъем ISP.

Расположение контактов разъема питания программатора:



Замечание: Из-за низкого энергопотребления в неактивном состоянии (индикатор питания 2 не активен), программатор не имеет выключателя питания.



Подключение программатора к компьютеру

Подключение к порту LPT

***Замечание:** При подключении программатора, строго следуйте данной инструкции. Это позволит избежать поломок и выхода из строя Вашего персонального компьютера и программатора. Нельзя использовать электронные или механические переключатели параллельного порта (LPT switches) для одновременной работы с программатором и печатающим устройством. Для подключения печатающего устройства установите в компьютер плату расширения с дополнительным параллельным портом (LPT). Плата расширения приобретается отдельно.*

Выключите питание персонального компьютера и отсоедините шнур питания от сети 220V. Сетевой адаптер программатора также должен быть отключен от питающей сети.

Подключите разъем интерфейсного кабеля, находящегося в комплекте с программатором к разъему программатора (5). Противоположный разъем интерфейсного кабеля подключите к свободному параллельному порту (LPT) компьютера. Обязательно закрепите предохранительные винты на разъемах интерфейсного кабеля – это позволит избежать выпадения кабеля из разъема и исключит возможные проблемы при работе с программатором.

Если компьютер оборудован единственным параллельным портом и к нему подключено печатающее или иное устройство, необходимо выключить печатающее устройство, отсоединить шнур питания от сети 220V, а затем отключить интерфейсный кабель этого устройства от компьютера.

Подключите разъем сетевого адаптера к гнезду питания программатора (6) и включите сетевой адаптер в питающую сеть. Наличие питания программатора можно проверить по активному состоянию индикатора питания (2). При этом индикаторы состояния программатора (3) будут активны в течение короткого промежутка времени. Программатор готов к работе после того как индикаторы состояния программатора (3) перейдут в неактивное состояние.

Подключите компьютер к сети 220V и включите его. После загрузки операционной системы компьютера, запустите универсальную управляющую программу.



Если по каким-то причинам требуется подключить программатор к работающему компьютеру, воспользуйтесь нижеприведенной рекомендацией:

Для подключения программатора к работающему компьютеру, **вначале** к параллельному порту (LPT) компьютера **подключается интерфейсный кабель** программатора, а **затем** к питающей сети **подключается сетевой адаптер** программатора.

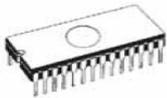
Если по каким-то причинам требуется отключить программатор от работающего компьютера, воспользуйтесь нижеприведенной рекомендацией:

Для отключения программатора от работающего компьютера, **вначале** **отключается сетевой адаптер** программатора от питающей сети, а **затем** **отключается интерфейсный кабель** программатора от параллельного порта (LPT) компьютера.

Подключение к порту USB

Подключение программатора к порту USB компьютера не отличается от обычной процедуры подключения любого устройства с интерфейсом USB.

Программатор, подключенный к порту USB, должен использовать отдельный сетевой адаптер, так как питание от порта USB программатор не получает.



Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме-ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

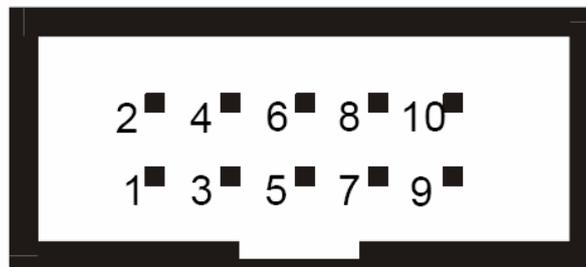
Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.

Внутрисхемное программирование (ISP)

Назначение контактов разъема ISP зависит в каждом конкретном случае от выбранной для программирования микросхемы (режим ISP для данного типа микросхемы устанавливается при его выборе из списка поддерживаемых микросхем). Более подробная информация может быть получена в меню **Микросхема–Информация о ИС (Device–Device Info)** управляющей программы или с помощью сочетания клавиш <Ctrl+F1>.

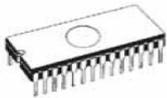
Нумерация контактов разъема ISP (вид спереди):



Кабель ISP для программатора BeeProg:



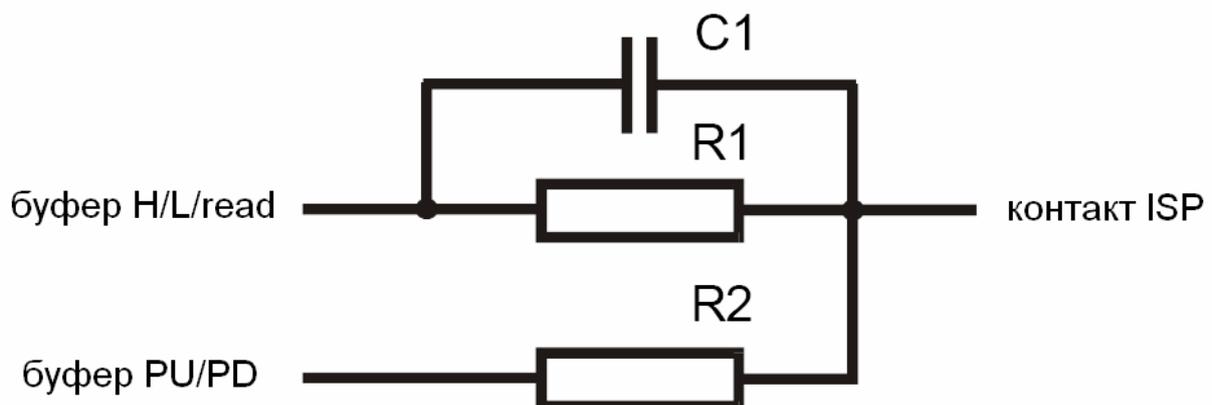
Замечание: Контакт №1 на кабеле для внутрисхемного программирования помечен символом ‘▲’.



Для исключения повреждения и выхода из строя программируемой микросхемы при использовании режима ISP, выполняйте следующие правил:

- при программировании микросхемы в режиме ISP не устанавливайте одновременно другую микросхему в гнездо ZIF socket;
- при программировании микросхемы в гнезде ZIF socket, не подключайте кабель ISP к программатору;
- используйте кабель ISP входящий в комплект программатора. Применение кабелей иной длины или кабелей, изготовленных из других материалов, может привести к нестабильному программированию или полной невозможности программирования микросхемы Вашим программатором;
- программатор поддерживает программирование микросхем (контакт № 1 разъема ISP) и ограниченных программируемых систем (контакт № 5 разъема ISP), однако программируемые системы не поддерживают программатор;
- программатор контролирует величину напряжения программирования для выбранной микросхемы. Процесс программирования не будет выполняться, если конструкция программируемой системы оказывает влияние на величину напряжения программирования.

Принципиальная схема построения узла ISP (буферы H/L/read):



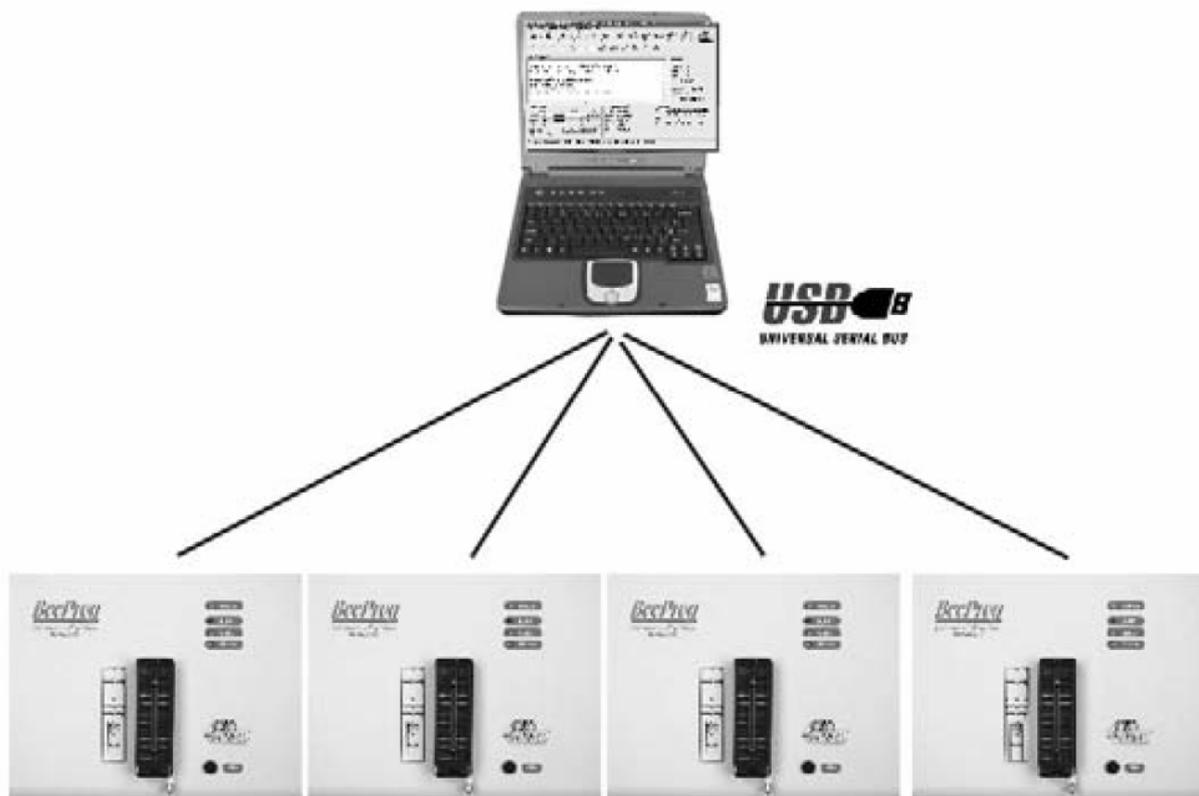
Мультипрограммирование

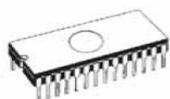
Наличие интерфейса USB программатора VeeProg открывает возможности по подключению к одному компьютеру одновременно до четырех программаторов.

При таком способе подключения отсутствует снижение скорости программирования. Также на каждом отдельном программаторе допускается программирование различных микросхем с различными параметрами программирования.

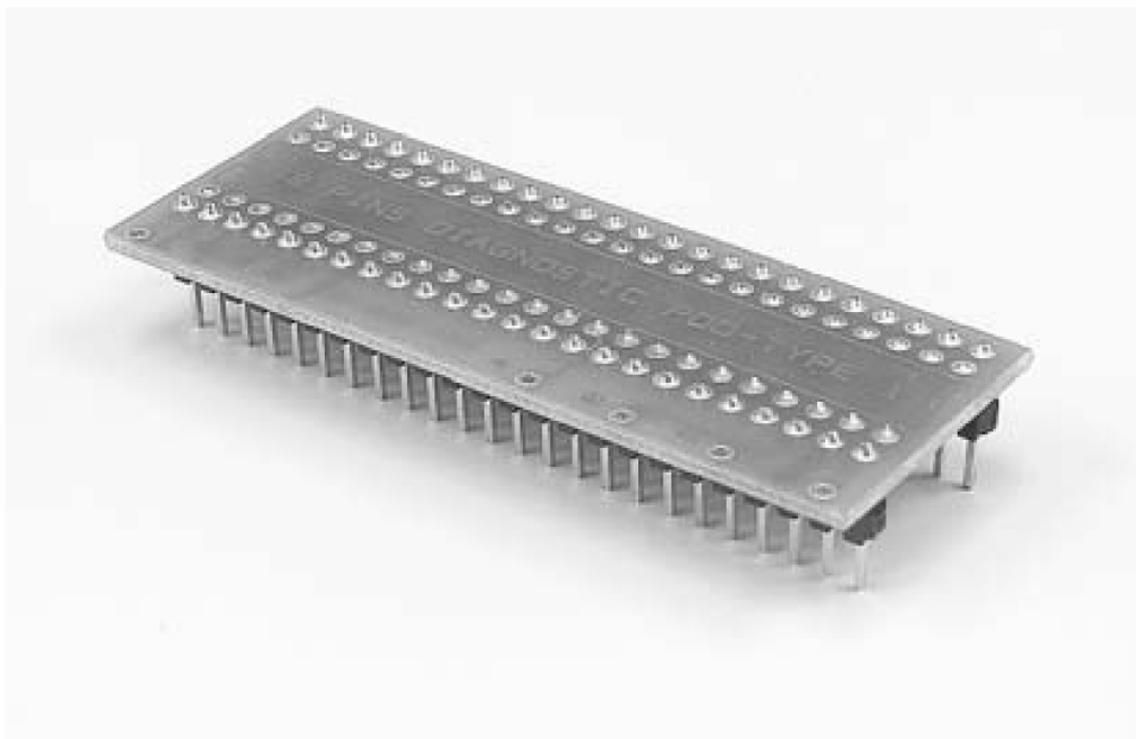
Для использования режима мультипрограммирования с программаторами VeeProg, при установке программного обеспечения требуется добавить режим мультипрограммирования, отметив специальный пункт в Мастере установки программного обеспечения.

Перед началом работы в режиме мультипрограммирования требуется запуск специальной программы **pg4uwmc.exe**. Эта программа позволяет работать с файлами данных и файлами проектов и установить необходимые параметры отдельно для каждого программатора.





Самодиагностика и калибровка



При возникновении подозрений в некорректной работе программатора, рекомендуется использовать диагностический модуль (POD), входящий в комплект поставки программатора, для проведения процедур тестирования программатора и его калибровки. Тестирование и калибровку рекомендуется проводить каждые 6 месяцев.

Различные способы и методы тестирования и калибровки программатора доступны в меню **Программатор (Programmer)** управляющей программы.

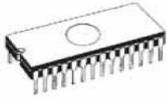
Технические характеристики

База программатора, DAC

- интерфейс USB (версия 2,0);
- максимальная скорость передачи данных интерфейсной части программатора, совместимой со стандартом IEEE-1284, до 1 Mbit/s;
- самостоятельный центральный процессор программатора;
- три цифро-аналоговых преобразователя (DAC) для управления напряжениями VCCP, VPP1 и VPP2;
- диапазон напряжения VCCP 0...8V/1A;
- диапазон напряжения VPP1, VPP2 0...26V/1A;
- автокалибровка;
- самотестирование;
- защита от повреждения статическими разрядами (ESD protect) на интерфейсном разъеме LPT и разъеме питания.

Разъем DIL ZIF socket

- 48-контактное гнездо модуля с нулевым усилением (ZIF socket) форм-фактора DIL для установки любых микросхем с количеством выводов до 48 и расстоянием между рядами выводов в диапазоне 300-600 mil;
- аппаратный драйвер на 48 выводов;
- переназначение напряжений VCC, VPP1 и VPP2 на любой вывод гнезда ZIF;
- переназначение нулевого потенциала на любой вывод гнезда ZIF;
- мощные TTL-совместимые буферы, обеспечивающие выбор H/L/CLK логических уровней а также режимов PU/PD на любом выводе гнезда ZIF;
- аналоговые буферы обеспечивающие уровень напряжения 1,8...26V на любом выводе гнезда ZIF;
- защита от повреждения статическими разрядами (ESD protect) на любом выводе гнезда ZIF (стандарт IEC1000-4-2: 15kV воздушный зазор, 8kV контакт на вывод);
- непрерывное тестирование перед каждой операцией программирования.



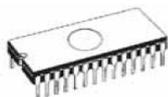
Разъем ISP

- 10–контактный разъем, с ключом, предотвращающим неправильную установку кабеля ISP;
- 5 TTL–совместимых буферов обеспечивающих выбор H/L/CLK логических уровней а также режимов PU/PD. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV);
- диапазон напряжения VCCP 2...7V/100mA (переназначается на выводы №№ 1 и 3);
- диапазон напряжения VPP 2...25V/50mA (переназначается на выводы №№ 2, 3, 4, 6, 8 и 10);
- возможность интеллектуального выбора уровня напряжения VCCP;
- поддержка напряжения питания программируемой системы в диапазоне 2...6V/250mA.

Типы поддерживаемых микросхем

Разъем DIL ZIF socket

- EPROM: NMOS/CMOS, 2708 *, 27xxx, 27Cxxx, с 8/16–битной шиной и поддержкой семейств LV;
- EEPROM: NMOS/CMOS, 28xxx, 28Cxxx, 27EExxx, с 8/16–битной шиной;
- Flash EPROM: 28Fxxx, 29Cxxx, 29Fxxx, 29BVxxx, 29LVxxx, 29Wxxx, 49Fxxx, с объемом памяти от 256Kbit до 32Mbit, с 8/16–битной шиной и поддержкой семейств LV;
- Serial E(E)PROM: 24Cxxx, 24Fxxx, 25Cxxx, 45Dxxx, 59Cxxx, 25Fxxx, 25Pxxx, 85xxx, 93Cxxx, NVM3060, MDAxxx, с полной поддержкой семейств LV;
- Configuration (EE)PROM: XCFxxx, XC17xxxx, XC18Vxxx, EPCxxx, AT17xxx, 37LVxx;
- 1–Wire E(E)PROM: DS1xxx, DS2xxx;
- PROM: AMD, Harris, National, Philips/Sigmetics, Tesla, TI;
- NV RAM: Dallas DSxxx, SGS/Inmos МКxxx, SIMTEK STKxxx, XICOR 2xxx, ZMD U63x;
- PLD Altera: MAX 3000A, MAX 7000A, MAX 7000B, MAX 7000S, MAX7000AE;
- PLD Lattice: ispGAL22V10x, ispLSI1xxx, ispLSI1xxxEA, ispLSI2xxx, ispLSI2xxxA, ispLSI2xxxE, ispLSI2xxxV, ispLSI2xxxVE, ispLSI2xxxVL, LC4xxxB/C/V/ZC, M4–xx/xx, M4A3–xx/xx, M4A5–xx/xx, M4LV–xx/xx;
- PLD Xilinx: XC9500, XC9500XL, XC9500XV, CoolRunner XPLA3, CoolRunner–II;
- PLD семейств SPLD/CPLD: AMI, Atmel, AMD–Vantis, Gould, Cypress, ICT, Lattice, NS, Philips, STM, VLSI, TI;
- Микроконтроллеры серии 48: 87x41, 87x42, 87x48, 87x49, 87x50;
- Микроконтроллеры серии 51: 87xx, 87Cxxx, 87LVxx, 89Cxxx, 89Sxxx, 89LVxxx, всех производителей, Philips LPC;
- Микроконтроллеры Intel серии 196: 87C196 KB/KC/KD/KT/KR/...;
- Микроконтроллеры Atmel AVR: AT90Sxxxx, ATtiny, ATmega;
- Микроконтроллеры Cypress: CY8Cxxxxx;
- Микроконтроллеры ELAN: EM78Pxxx;
- Микроконтроллеры Microchip PICmicro: PIC10xxx, PIC12xxx, PIC16xxx, PIC17Cxxx, PIC18xxx, dsPIC;
- Микроконтроллеры Motorola: 68HC05, 68HC08, 68HC11;
- Микроконтроллеры National: COP8xxx;



- Микроконтроллеры NEC: uPD78Pxxx;
- Микроконтроллеры Scenix (Uvicom): SXxxx;
- Микроконтроллеры SGS–Thomson: ST6xx, ST7xx, ST10xx;
- Микроконтроллеры TI: MSP430 и MSC121x;
- Микроконтроллеры ZILOG: Z86/Z89xxx и Z8xxx;
- Микроконтроллеры: EM Microelectronic, Fujitsu, Goal Semiconductor, Hitachi, Holtek, Princeton, Macronix, Winbond, Infineon (Siemens), NEC, Samsung, Toshiba.

Тестирование микросхем

- TTL: 54, 74 S/LS/ALS/H/HC/HCT;
- CMOS: 4000, 4500;
- Static RAM: 6116...624000;
- Определяемые пользователем алгоритмы тестирования.

Разъем ISP

- Serial E(E)PROM: серии I²C;
- Микроконтроллеры Atmel: AT89Sxxx, AT90Sxxxx, ATtiny, ATmega;
- Микроконтроллеры Cypress: CY8C2xxxx;
- Микроконтроллеры Elan: EM78Pxxx;
- Микроконтроллеры EM Microelectronic: с 4/8–битной шиной;
- Микроконтроллеры Microchip PICmicro: PIC10xxx, PIC12xxx, PIC16xxx, PIC17xxx, PIC18xxx, dsPIC;
- Микроконтроллеры Motorola/Freescale: HC08 GT, LJ, QY, QT;
- Микроконтроллеры Philips: серии LPC;
- Микроконтроллеры TI: MSP430;
- PLD Lattice: ispGAL22xV10x, ispLSI1xxxEA, ispLSI2xxxE, ispLSI2xxxV, ispLSI2xxxVE, ispLSI2xxxVL, M4–xx/xx, M4LV–xx/xx, M4A3–xx/xx, M4A5–xx/xx, LC4xxxB/C/V/ZC;
- Altera: MAX 3000A, MAX 7000A, MAX 7000B, MAX 7000S, MAX 9000, MAX II;
- Xilinx: XC9500, XC9500XL, XC9500XV, CoolRunner XPLA3, CoolRunner–II.

*Микросхемы отмеченные символом * являются устаревшими и программируются с помощью специальных адаптеров.*

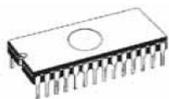
Адаптеры

- разнообразные адаптеры для корпусов типов DIP, PLCC, SOIC, PSOP, SSOP, TSOP, TSSOP, TQFP, QFN (MLF), SDIP, BGA и других типоразмеров;
- поддержка всех микросхем в корпусах типоразмера DIP без применения дополнительных адаптеров;
- поддержка всех микросхем в корпусах отличных от типоразмера DIP с количеством выводов более 48 с помощью ряда универсальных адаптеров;
- частичная совместимость с адаптерами сторонних производителей.

Скорость программирования

Микросхема	Действие	Время
AT29C040A	программирование и верификация	21
AM29DL323DB	программирование и верификация	38
AM29DL640	программирование и верификация	76
AT45D081	программирование и верификация	43
AT89C51RD2	программирование и верификация	15
PIC18F452	программирование и верификация	4

Замечание: Измерение скорости программирования производилось на тестовом компьютере P4, 2,4 GHz, режим LPT порта ECP, Установленная операционная система MS Windows XP.



Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

- выбор необходимой микросхемы по типу, производителю или любому фрагменту имеющейся информации;
- автоматическое определение семейства микросхем памяти EPROM/Flash EPROM по идентификатору ID;
- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- редактирование и установка конфигурационных битов;
- тестирование на ошибочные биты;
- подсчет контрольной суммы.

Операции контроля микросхем

- тест на правильную ориентацию микросхемы в гнезде программатора ZIF;
- тест на наличие надежного контакта по всем выводам микросхемы;
- проверка соответствия настроек для выбранной микросхемы по идентификатору.

Специальные операции с микросхемами

- непрерывный режим – начало программирования после смены микросхемы в гнезде программатора ZIF;
- автоматическая генерация идентификаторов (серийных номеров) при программировании группы однотипных микросхем;
- статистика программирования;
- режим счетчика.

Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение.

Поддерживаемые форматы данных

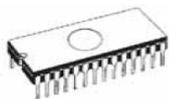
- Unformatted binary (BIN (raw));
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX;
- Altera POF, JEDEC (версии 3.0.A), ABEL, CUPL, PALASM, TANGO PLD, OrCAD PLD, PLD Designer ISDATA;
- JAM (JEDEC STAPL), JBC (Jam STAPL Byte Code), STAPL (STAPL File) JEDEC стандарта JESD-71;
- VME (ispVME File VME2.0/VME3.0).

Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

- напряжение питания 15–18V DC;
- потребляемый ток 1A;
- максимальная потребляемая мощность 12W в активном режиме;
- максимальная потребляемая мощность 2W в неактивном режиме;
- размеры 160x190x42 мм;
- масса без дополнительных адаптеров 0,9 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**



Комплект поставки

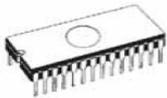
- программатор VeeProg;
- кабель LPT;
- кабель USB;
- кабель ISP;
- диагностический модуль POD;
- чехол для гнезда ZIF;
- сетевой адаптер питания 100...240V AC/15V DC/1A;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep-Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.

LabProg+





Введение

LabProg+ – это представитель линейки универсальных программаторов, поддерживающих различные режимы программирования большого количества микросхем. Аппаратная часть программатора содержит 48 мощных TTL–совместимых буферов, обеспечивающих выбор H/L/CLK логических уровней а также режимов PU/PD на любом контакте гнезда программатора. Современные быстродействующие буферы программатора обеспечивают надежную работу на самых высоких скоростях для всех поддерживаемых типов микросхем с типоразмером корпусов DIP без использования дополнительных адаптеров.

LabProg+ также является тестировщиком логических ИС TTL/CMOS и микросхем памяти. Алгоритмы тестирования могут быть изменены для конкретного семейства ИС.

Подключение программатора осуществляется к любым компьютерам семейства IBM Pentium или более современным, оборудованным стандартным портом LPT. Программатор поддерживает стандарт IEEE–1284 (режим ECP/EPP) для порта LPT. Наличие порта LPT позволяет подключить программатор к любым компьютерам – самым современным или портативным, а также к более старым, без применения специальных адаптеров или дополнительных плат расширения.

Разработанный на основе микропроцессорной системы, программатор **LabProg+** представляет собой интеллектуальный продукт, способный закончить операцию с микросхемой даже в случае нарушения работы компьютера или потери с ним связи. Эта особенность позволяет использовать программатор совместно с различными операционными системами, в том числе и мультизадачными. (MS Windows 95/98/Me/NT/2000/XP).

Перед началом процесса программирования программатор определяет правильность установки микросхемы в гнездо ZIF и наличие надежного контакта со всеми выводами микросхемы. Также доступна функция чтения идентификатора ИС для определения параметров программирования. Эти функции позволяют избежать необратимого повреждения программируемой микросхемы из–за ошибок, допущенных пользователем при ее установке в программатор или выбора неправильного режима программирования.

Программатор имеет обширные инструменты для самодиагностики, позволяющие проверить работу интерфейсной части, формирование уровней напряжений и задержек сигналов.

Оптимальное проектирование конструкции программатора повышает надежность его эксплуатации, исключая потерю контактов в разъемах, наводки помех по цепям питания или провалы напряжений на выводах гнезда программатора. Все входные и выходные цепи программатора, гнезда разъемы имеют защиту от повреждения статическими разрядами (ESD protect).

Проверка качества записи выбранной микросхемы проводится при предельно допустимых минимальном и максимальном напряжениях питания, что позволяет в дальнейшем гарантировать надежную работу запрограммированной микросхемы в самых сложных условиях эксплуатации.

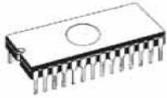
Адаптеры позволяют работать с микросхемами в различных корпусах: PLCC, SOIC, PSOP, SSOP, TSOP, TSSOP, TQFP, QFN (MLF), SDIP, BGA и других типах корпусов.

Интерфейс управляющей программы включает в себя несколько панелей инструментов, на которых расположены кнопки–пиктограммы, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка). Для облегчения работы так же предусмотрена поддержка функциональных клавиш и их сочетаний. В большинстве случаев возможно использование встроенной контекстной подсказки.

Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера. Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

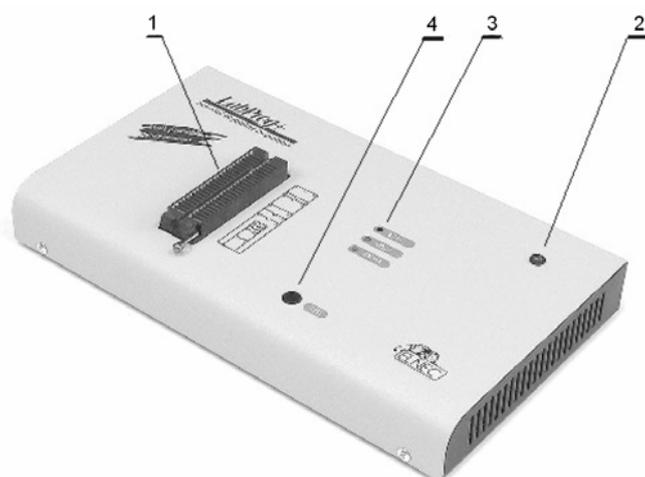


Поиск и выбор необходимой микросхемы производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных (по маске).

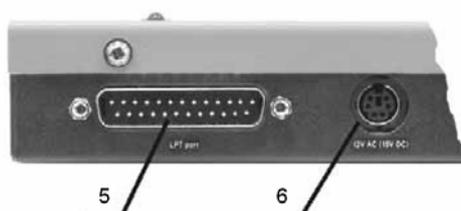
Работа с новыми ИС может быть уже начата после простого обновления текущей версии программного обеспечения – модернизация аппаратной части не требуется.

Жесткий контроль готовой продукции и использование оригинальных комплектующих в производстве программаторов позволили компании ELNEC предоставить гарантийное обслуживание своей продукции в течение трех лет (только для продукции приобретенной в компании ELNEC).

Основные элементы программатора

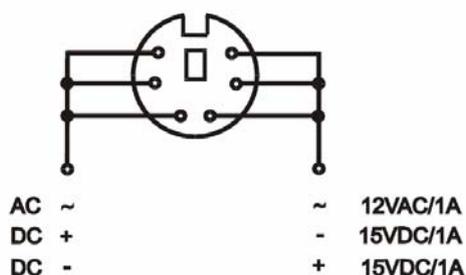


1. Гнездо ZIF socket на 48 контактов;
2. Индикатор питания;
3. Индикаторы состояния питания;
4. Кнопка функции YES!

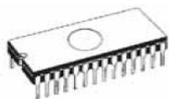


5. Интерфейсный разъем для подключения кабеля LPT;
6. Разъем питания для подключения сетевого адаптера.

Расположение контактов разъема питания программатора:



Замечание: Из-за низкого энергопотребления в неактивном состоянии (индикатор питания 2 не активен), программатор не имеет выключателя питания.



Подключение программатора к компьютеру

***Замечание:** При подключении программатора, строго следуйте данной инструкции. Это позволит избежать поломок и выхода из строя Вашего персонального компьютера и программатора. Нельзя использовать электронные или механические переключатели параллельного порта (LPT switches) для одновременной работы с программатором и печатающим устройством. Для подключения печатающего устройства установите в компьютер плату расширения с дополнительным параллельным портом (LPT). Плата расширения приобретается отдельно.*

Выключите питание персонального компьютера и отсоедините шнур питания от сети 220V. Сетевой адаптер программатора также должен быть отключен от питающей сети.

Подключите разъем интерфейсного кабеля, находящегося в комплекте с программатором к разъему программатора (5). Противоположный разъем интерфейсного кабеля подключите к свободному параллельному порту (LPT) компьютера. Обязательно закрепите предохранительные винты на разъемах интерфейсного кабеля – это позволит избежать выпадения кабеля из разъема и исключит возможные проблемы при работе с программатором.

Если компьютер оборудован единственным параллельным портом и к нему подключено печатающее или иное устройство, необходимо выключить печатающее устройство, отсоединить шнур питания от сети 220V, а затем отключить интерфейсный кабель этого устройства от компьютера.

Подключите разъем сетевого адаптера к гнезду питания программатора (6) и включите сетевой адаптер в питающую сеть. Наличие питания программатора можно проверить по активному состоянию индикатора питания (2). При этом индикаторы состояния программатора (3) будут активны в течение короткого промежутка времени. Программатор готов к работе после того как индикаторы состояния программатора (3) перейдут в неактивное состояние.

Подключите компьютер к сети 220V и включите его. После загрузки операционной системы компьютера, запустите универсальную управляющую программу.

***Замечание:** Для исключения случаев повреждения порта LPT компьютера, не рекомендуется подключение и отключение интерфейсного кабеля при включенном компьютере.*

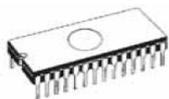


Если по каким-то причинам требуется подключить программатор к работающему компьютеру, воспользуйтесь нижеприведенной рекомендацией:

Для подключения программатора к работающему компьютеру, **вначале** к параллельному порту (LPT) компьютера **подключается интерфейсный кабель** программатора, а **затем** к питающей сети **подключается сетевой адаптер** программатора.

Если по каким-то причинам требуется отключить программатор от работающего компьютера, воспользуйтесь нижеприведенной рекомендацией:

Для отключения программатора от работающего компьютера, **вначале** **отключается сетевой адаптер** программатора от питающей сети, а **затем** **отключается интерфейсный кабель** программатора от параллельного порта (LPT) компьютера.



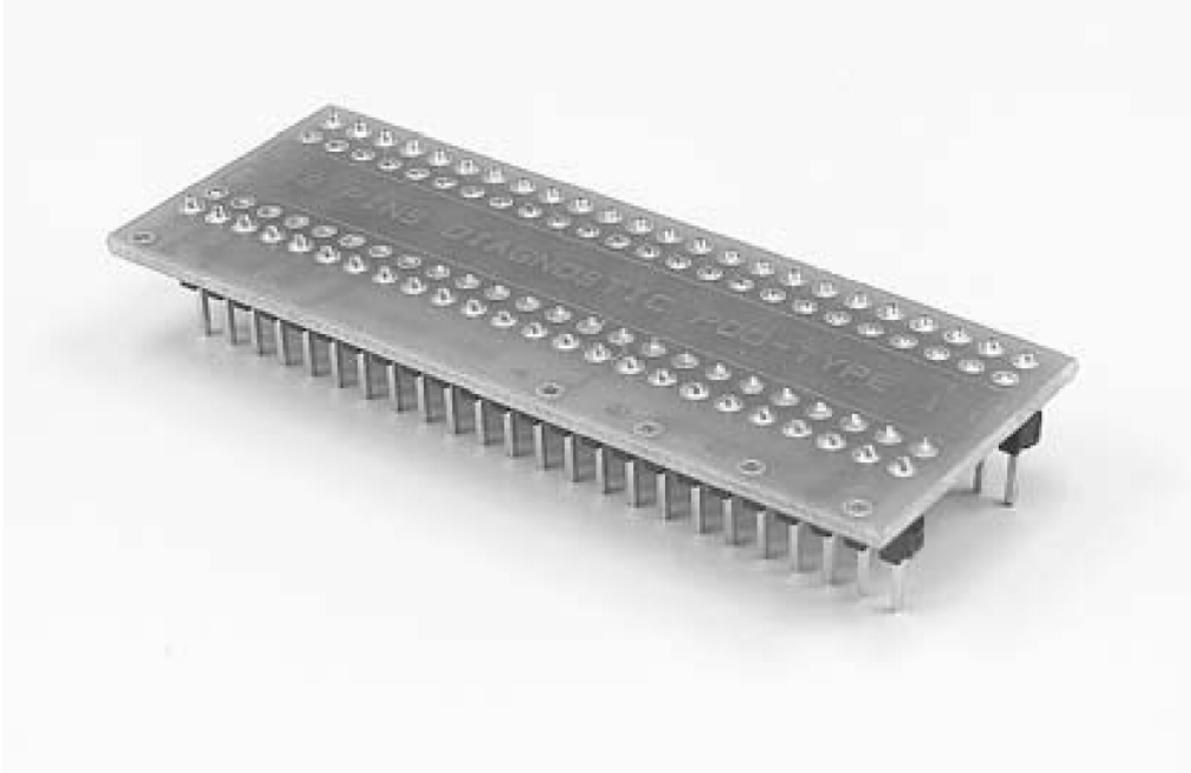
Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме-ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

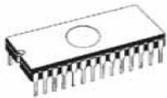
Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.

Самодиагностика и калибровка



При возникновении подозрений в некорректной работе программатора, рекомендуется использовать диагностический модуль (POD), входящий в комплект поставки программатора, для проведения процедур тестирования программатора и его калибровки. Тестирование и калибровку рекомендуется проводить каждые 6 месяцев.

Различные способы и методы тестирования и калибровки программатора доступны в меню **Программатор (Programmer)** управляющей программы.



Технические характеристики

База программатора, DAC

- 48-контактное гнездо модуля с нулевым усилением (ZIF socket) форм-фактора DIL для установки любых микросхем с количеством выводов до 48 и расстоянием между рядами выводов в диапазоне 300–600 mil;
- три цифро-аналоговых преобразователя (DAC) для управления напряжениями VCCP, VPP1 и VPP2;
- мощные TTL-совместимые буферы, обеспечивающие выбор H/L/CLK логических уровней а также режимов PU/PD на любом выводе гнезда ZIF;
- полная поддержка микросхем с пониженным напряжением питания LV (до 1,8V);
- автокалибровка.

Типы поддерживаемых микросхем

Разъем DIL ZIF socket

- EPROM: NMOS/CMOS, 1702 *, 2708 *, 27xxx, 27Cxxx, с 8/16-битной шиной и поддержкой семейств LV;
- EEPROM: NMOS/CMOS, 28xxx, 28Cxxx, 27EExxx, с 8/16-битной шиной;
- Flash EPROM: 28Fxxx, 29Cxxx, 29Fxxx, 29BVxxx, 29LVxxx, 29Wxxx, 49Fxxx, с объемом памяти от 256Kbit до 32Mbit, с 8/16-битной шиной и поддержкой семейств LV;
- Serial E(E)PROM: 24Cxxx, 24Fxxx, 25Cxxx, 45Dxxx, 59Cxxx, 25Fxxx, 25Pxxx, 85xxx, 93Cxxx, NVM3060, MDAxxx, с полной поддержкой семейств LV;
- Configuration (EE)PROM: XCFxxx, XC17xxxx, XC18Vxxx, EPCxxx, AT17xxx, 37LVxx;
- PROM: AMD, Harris, National, Philips, Signetics, Tesla, TI;
- NV RAM: Dallas DSxxx, SGS/Inmos MKxxx, SIMTEK STKxxx, XICOR 2xxx, ZMD U63x;
- PLD Altera: MAX 3000A, MAX 7000A, MAX 7000B, MAX 7000S, MAX7000AE;

- PLD Lattice: ispGAL22V10x, ispLSI1xxx, ispLSI1xxxEA, ispLSI2xxx, ispLSI2xxxA, ispLSI2xxxE, ispLSI2xxxV, ispLSI2xxxVE, M4–xx/xx, M4A3–xx/xx, M4A5–xx/xx, M4LV–xx/xx;
- PLD Xilinx: XC9500, XC9500XL, XC9500XV, CoolRunner XPLA3, CoolRunner–II;
- PLD семейств SPLD/CPLD: AMI, Atmel, AMD–Vantis, Gould, Cypress, ICT, Lattice, NS, Philips, STM, VLSI, TI;
- Микроконтроллеры серии 48: 87x41, 87x42, 87x48, 87x49, 87x50;
- Микроконтроллеры серии 51: 87xx, 87Cxxx, 87LVxx, 89Cxxx, 89Sxxx, 89LVxxx, всех производителей, Philips 87C748...752, Philips LPC, Cygnal/Silicon Laborat. C8051;
- Микроконтроллеры Intel серии 196: 87C196 KB/KC/KD/KT/KR/...;
- Микроконтроллеры Atmel AVR: AT90Sxxxx, ATtiny;
- Микроконтроллеры ELAN: EM78Pxxx;
- Микроконтроллеры Microchip PICmicro: PIC10xxx, PIC12Cxxx, PIC16C5x, PIC16Cxxx, PIC17Cxxx, PIC18Cxxx, dsPIC;
- Микроконтроллеры Motorola: 68HC11 **;
- Микроконтроллеры National: COP8xxx;
- Микроконтроллеры NEC: uPD78Pxxx;
- Микроконтроллеры Scenix (Ubicom): SXxxx;
- Микроконтроллеры SGS–Thomson: ST6xx;
- Микроконтроллеры TI: MSP430;
- Микроконтроллеры ZILOG: Z86xxx;
- Микроконтроллеры: Cypress, EM Microelectronic, Fujitsu, Goal Semiconductor, Princeton, Macronix, Winbond, Hitachi, Holtek, Infineon (Siemens), NEC, Samsung, Toshiba.

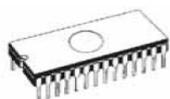
Тестирование микросхем

- TTL: 54, 74 S/LS/ALS/H/HC/HCT;
- CMOS: 4000, 4500;
- Static RAM: 6116...624000;
- Определяемые пользователем алгоритмы тестирования.

*Микросхемы отмеченные символом * являются устаревшими и программируются с помощью специальных адаптеров.*

*Микросхемы компании Motorola используют специальные режимы программирования. Программатор LabProg+ не поддерживает работу с такими микросхемами за исключением отмеченных символом **.*

Программатор LabProg+ не поддерживает режим ISP. Используйте программаторы поддерживающие этот режим!



Адаптеры

- разнообразные адаптеры для корпусов типов DIP, PLCC, SOIC, PSOP, SSOP, TSOP, TSSOP, TQFP, QFN (MLF), SDIP, BGA и других типоразмеров;
- поддержка всех микросхем в корпусах типоразмера DIP без применения дополнительных адаптеров;
- поддержка всех микросхем в корпусах отличных от типоразмера DIP с количеством выводов более 48 с помощью ряда универсальных адаптеров;
- частичная совместимость с адаптерами сторонних производителей.

Скорость программирования

Микросхема	Действие	Время
27C010	программирование и верификация	39
AT29C040A	программирование и верификация	75
AM29F040	программирование и верификация	165
PIC16C67	программирование и верификация	30

Замечание: Измерение скорости программирования производилось на тестовом компьютере P4, 2,4 GHz, режим LPT порта ECP, Установленная операционная система MS Windows XP.

Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

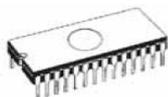
- выбор необходимой микросхемы по типу, производителю или любому фрагменту имеющейся информации;
- автоматическое определение семейства микросхем памяти EPROM/Flash EPROM по идентификатору ID;
- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- редактирование и установка конфигурационных битов;
- тестирование на ошибочные биты;
- подсчет контрольной суммы.

Операции контроля микросхем

- тест на правильную ориентацию микросхемы в гнезде программатора ZIF;
- тест на наличие надежного контакта по всем выводам микросхемы;
- проверка соответствия настроек для выбранной микросхемы по идентификатору.

Специальные операции с микросхемами

- непрерывный режим – начало программирования после смены микросхемы в гнезде программатора ZIF;
- автоматическая генерация идентификаторов (серийных номеров) при программировании группы однотипных микросхем;
- статистика программирования;
- режим счетчика.



Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение.

Поддерживаемые форматы данных

- Unformatted binary (BIN (raw));
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX;
- Altera POF, JEDEC (версии 3.0.A), ABEL, CUPL, PALASM, TANGO PLD, OrCAD PLD, PLD Designer ISDATA;
- JAM (JEDEC STAPL), JBC (Jam STAPL Byte Code), STAPL (STAPL File) JEDEC стандарта JESD-71.

Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

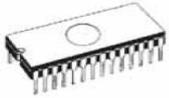
- напряжение питания 12–15V AC или 15–18V DC;
- потребляемый ток 1A;
- максимальная потребляемая мощность 12W в активном режиме;
- максимальная потребляемая мощность 1,5W в неактивном режиме;
- размеры 275x157x47 мм;
- масса без дополнительных адаптеров 1,5 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**

Комплект поставки

- программатор LabProg+;
- кабель LPT;
- диагностический модуль POD;
- чехол для гнезда ZIF;
- сетевой адаптер питания 100...240V AC/15V DC/1A;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep–Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.



SmartProg2



Введение

SmartProg2 – это представитель новой линейки универсальных программаторов, разработанных компанией ELNEC и использующих в основе своей работы различные алгоритмы программирования. Программатор будет полезен как исследовательским центрам так и сервисным и ремонтным предприятиям.

Программатор **SmartProg2** имеет в своей конструкции разъем для внутрисхемного программирования (ISP), что позволяет работать с программируемыми микросхемами без их извлечения из готовых устройств. Доступна функция тестирования микросхем Static RAM.

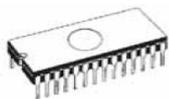
Высокая скорость программирования обеспечивается применением интерфейса USB 2,0 для связи с компьютером.

Подключение программатора осуществляется к любым компьютерам семейства IBM Pentium или более современным, оборудованным стандартным портом USB, без применения специальных адаптеров или дополнительных плат расширения.

Аппаратная часть программатора содержит 40 мощных TTL–совместимых буферов, обеспечивающих выбор Н/Л логических уровней а также режимов PU/PD на любом контакте гнезда программатора. Современные быстродействующие буферы программатора обеспечивают надежную работу на самых высоких скоростях для всех поддерживаемых типов микросхем и позволяют работать с микросхемами поддерживающими рабочее напряжение вплоть до 1,8V.

Перед началом процесса программирования программатор определяет правильность установки микросхемы в гнездо ZIF и наличие надежного контакта со всеми выводами микросхемы. Также доступна функция чтения идентификатора ИС для определения параметров программирования. Эти функции позволяют избежать необратимого повреждения программируемой микросхемы из–за ошибок, допущенных пользователем при ее установке в программатор или выбора неправильного режима программирования.

Программатор имеет обширные инструменты для самодиагностики, позволяющие проверить работу интерфейсной части, формирование уровней напряжений и задержек сигналов.



Проверка качества записи выбранной микросхемы проводится при предельно допустимых минимальном и максимальном напряжениях питания, что позволяет в дальнейшем гарантировать надежную работу запрограммированной микросхемы в самых сложных условиях эксплуатации.

Интерфейс управляющей программы включает в себя несколько панелей инструментов, на которых расположены кнопки–пиктограммы, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка). Для облегчения работы так же предусмотрена поддержка функциональных клавиш и их сочетаний. В большинстве случаев возможно использование встроенной контекстной подсказки.

Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера. Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

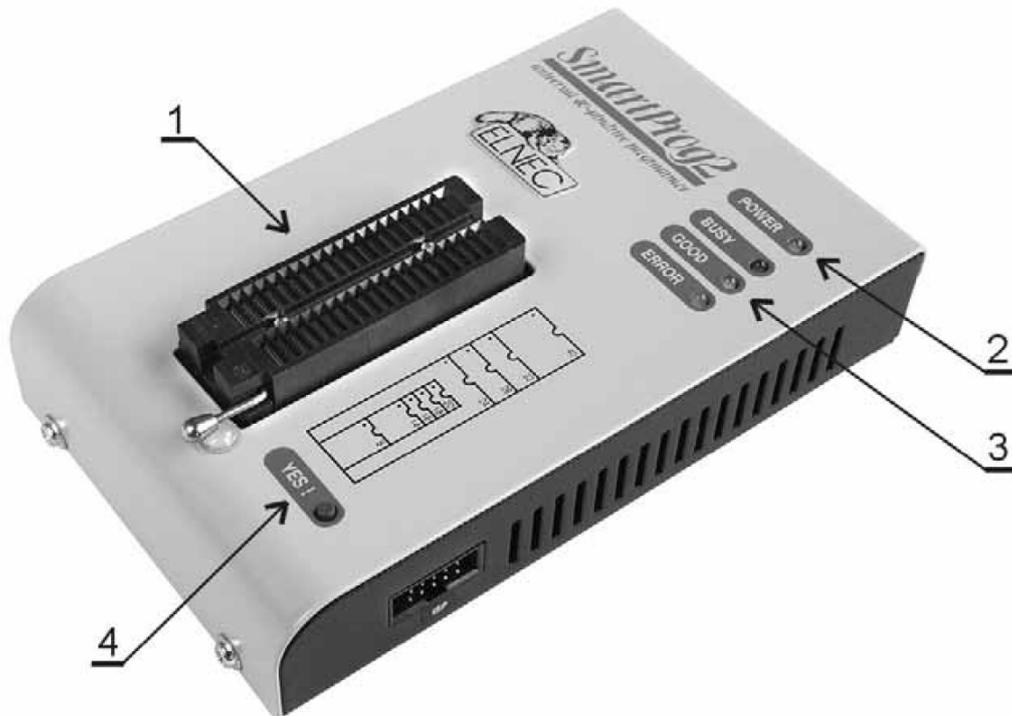
Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

Поиск и выбор необходимой микросхемы производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных (по маске).

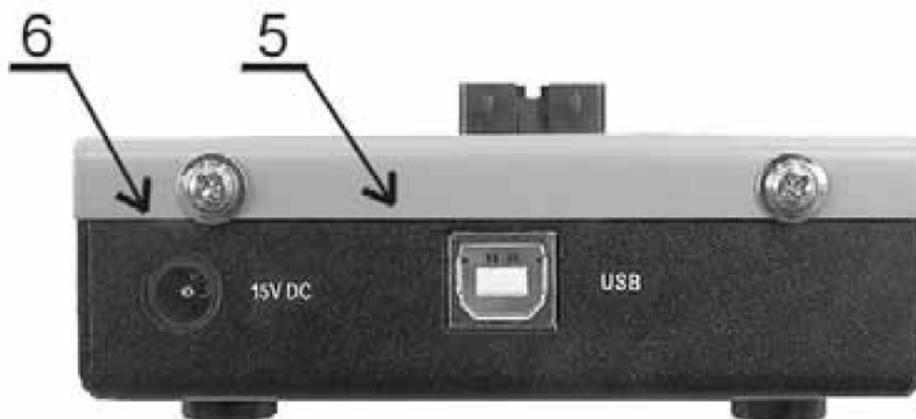
Адаптеры позволяют работать с микросхемами в различных корпусах: PLCC, SOIC, PSOP, SSOP, TSOP, TSSOP, TQFP, QFN (MLF), SDIP, BGA и других типах корпусов.

Жесткий контроль готовой продукции и использование оригинальных комплектующих в производстве программаторов позволили компании ELNEC предоставить гарантийное обслуживание своей продукции в течение трех лет (только для продукции приобретенной в компании ELNEC).

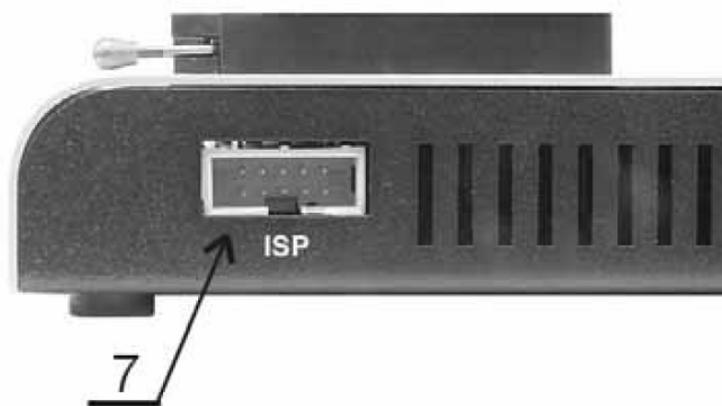
Основные элементы программатора



1. Гнездо ZIF socket на 40 контактов;
2. Индикатор питания;
3. Индикаторы состояния программатора;
4. Кнопка функции YES!



5. Интерфейсный разъем для подключения кабеля USB;
6. Разъем питания для подключения сетевого адаптера.



7. Разъем ISP.

Расположение контактов разъема питания программатора:

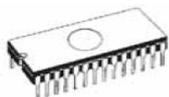


***Замечание:** Из-за низкого энергопотребления в неактивном состоянии (индикатор питания 2 не активен), программатор не имеет выключателя питания.*

Подключение программатора к компьютеру

Подключение программатора к порту USB компьютера не отличается от обычной процедуры подключения любого устройства с интерфейсом USB.

Программатор, подключенный к порту USB, должен использовать отдельный сетевой адаптер, так как питание от порта USB программатор не получает.



Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме-ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

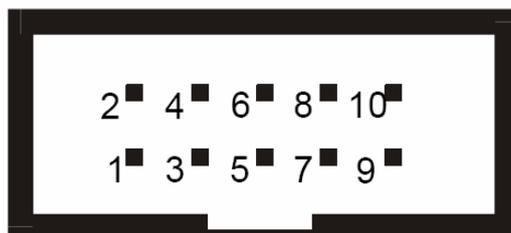
Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.

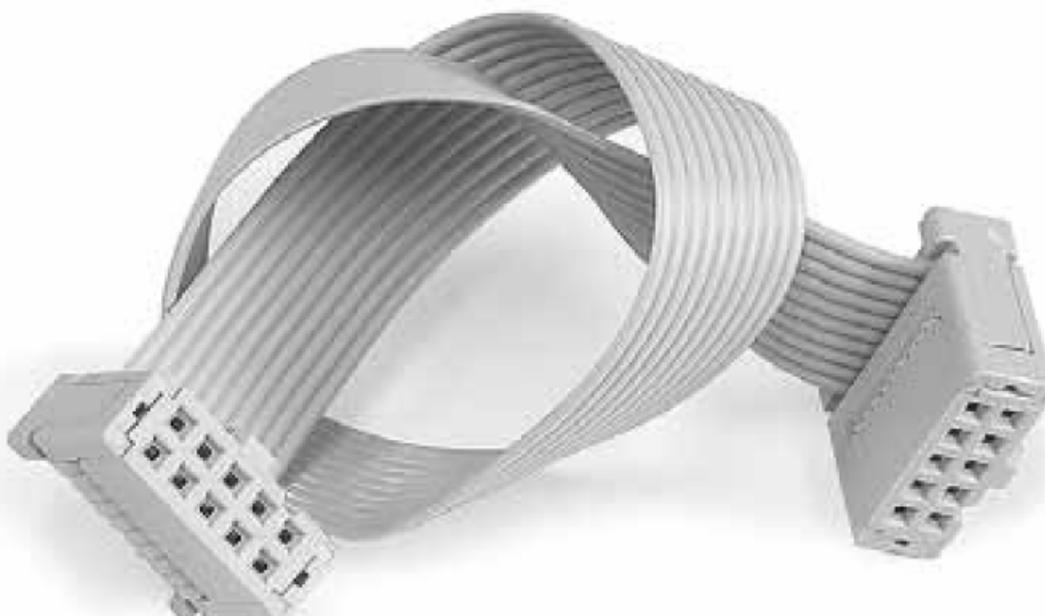
Внутрисхемное программирование (ISP)

Назначение контактов разъема ISP зависит в каждом конкретном случае от выбранной для программирования микросхемы (режим ISP для данного типа микросхемы устанавливается при его выборе из списка поддерживаемых микросхем). Более подробная информация может быть получена в меню **Микросхема–Информация о ИС (Device–Device Info)** управляющей программы или с помощью сочетания клавиш <Ctrl+F1>.

Нумерация контактов разъема ISP (вид спереди):



Кабель ISP для программатора SmartProg2:



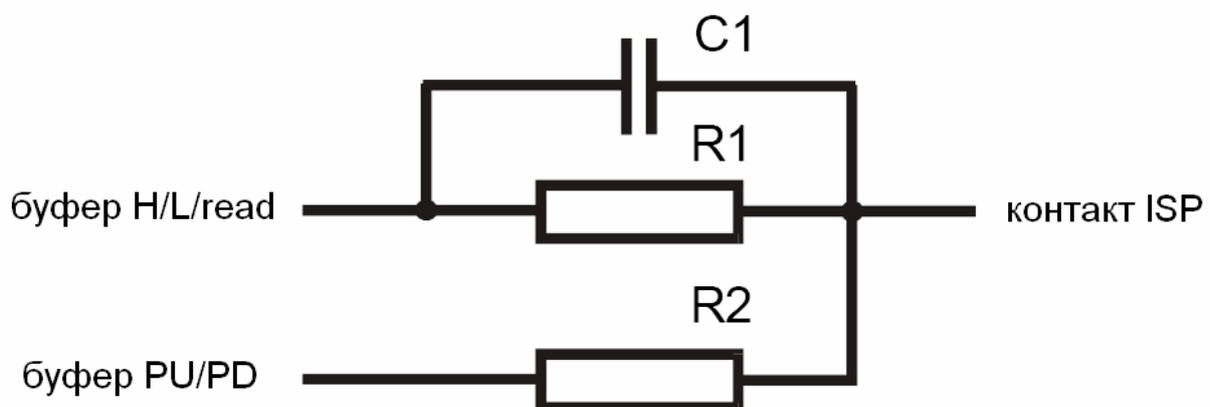
Замечание: Контакт №1 на кабеле для внутрисхемного программирования помечен символом '▲'.



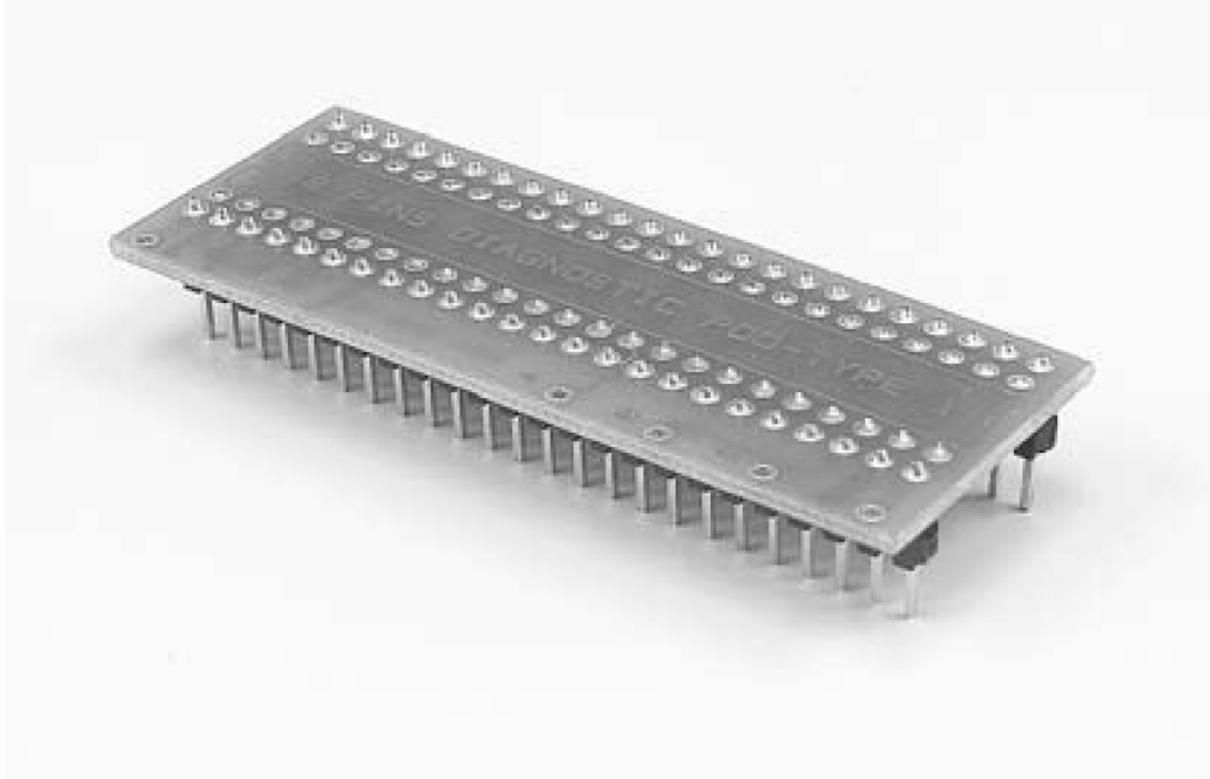
Для исключения повреждения и выхода из строя программируемой микросхемы при использовании режима ISP, выполняйте следующие правил:

- при программировании микросхемы в режиме ISP не устанавливайте одновременно другую микросхему в гнездо ZIF socket;
- при программировании микросхемы в гнезде ZIF socket, не подключайте кабель ISP к программатору;
- используйте кабель ISP входящий в комплект программатора. Применение кабелей иной длины или кабелей, изготовленных из других материалов, может привести к нестабильному программированию или полной невозможности программирования микросхемы Вашим программатором;
- программатор поддерживает программирование микросхем, однако программируемые системы не поддерживают программатор;
- программатор контролирует величину напряжения программирования для выбранной микросхемы. Процесс программирования не будет выполняться, если конструкция программируемой системы оказывает влияние на величину напряжения программирования.

Принципиальная схема построения узла ISP (буферы H/L/read):

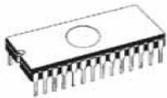


Самодиагностика и калибровка



При возникновении подозрений в некорректной работе программатора, рекомендуется использовать диагностический модуль (POD), входящий в комплект поставки программатора, для проведения процедур тестирования программатора и его калибровки. Тестирование и калибровку рекомендуется проводить каждые 6 месяцев.

Различные способы и методы тестирования и калибровки программатора доступны в меню **Программатор (Programmer)** управляющей программы.



Технические характеристики

База программатора, DAC

- два цифро–аналоговых преобразователя (DAC) для управления напряжениями VCCP и VPP;
- диапазон напряжения VCCP 2...7V/350mA;
- диапазон напряжения VPP 2...25V/200mA;
- интерфейс USB (версия 1,1/2,0);
- автокалибровка;
- самотестирование.

Разъем DIL ZIF socket

- 40–контактное гнездо модуля с нулевым усилением (ZIF socket) форм–фактора DIL для установки любых микросхем с количеством выводов до 40 и расстоянием между рядами выводов в диапазоне 300–600 mil;
- 40 мощных TTL–совместимых буферов, обеспечивающих выбор напряжений GND/VPP/VCC, логических уровней H/L/CLK, режимов PU/PD на любом выводе гнезда ZIF. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV);
- непрерывное тестирование перед каждой операцией программирования.

Разъем ISP

- 10–контактный разъем, с ключом, предотвращающим неправильную установку кабеля ISP;
- 5 TTL–совместимых буферов обеспечивающих выбор H/L/CLK логических уровней а также режимов PU/PD. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV);
- диапазон напряжения VCCP 2...7V/100mA;
- диапазон напряжения VPP 2...25V/50mA *;
- возможность интеллектуального выбора уровня напряжения VCCP.

* программатор не поддерживает питание программируемой системы. Если требуется поддержка – используйте программатор VeeProg.

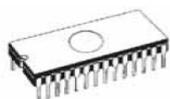
Типы поддерживаемых микросхем

Разъем DIL ZIF socket

- EPROM: NMOS/CMOS, 27xxx, 27Cxxx, с 8/16–битной шиной и поддержкой семейств LV ⁽¹⁾⁽²⁾;
- EEPROM: NMOS/CMOS, 28xxx, 28Cxxx, 27EExxx, с 8/16–битной шиной и поддержкой семейств LV ⁽¹⁾⁽²⁾;
- Flash EPROM: 28Fxxx, 29Cxxx, 29Fxxx, 29BVxxx, 29LVxxx, 29Wxxx, 49Fxxx, с 8/16–битной шиной и поддержкой семейств LV ⁽¹⁾⁽²⁾;
- Serial E(E)PROM: 24Cxxx, 24Fxxx, 25Cxxx, 45Dxxx, 59Cxxx, 25Fxxx, 25Pxxx, 85xxx, 93Cxxx, с полной поддержкой семейств LV ⁽¹⁾;
- Configuration (EE)PROM: XCFxxx, 37LVxx, XC17xxxx, EPCxxx, AT17xxx, с полной поддержкой семейств LV;
- NV RAM: Dallas DSxxx, SGS/Inmos МКxxx, SIMTEK STKxxx, XICOR 2xxx, ZMD U63x;
- PLD: Atmel, AMD–Vantis, Cypress, ICT, Lattice, NS ⁽¹⁾;
- Микроконтроллеры серии 51: 87Cxxx, 87LVxx, 89Cxxx, 89Sxxx, 89LVxxx, серии LPC для Atmel, Atmel W&M, Intel, Philips, SST, Winbond ⁽¹⁾⁽²⁾;
- Микроконтроллеры Atmel AVR: ATtiny, AT90Sxxx, ATmega ⁽¹⁾⁽²⁾;
- Микроконтроллеры Cypress: CY8Cxxxxxx;
- Микроконтроллеры ELAN: EM78Pxxx;
- Микроконтроллеры EM Microelectronic: с 4/8–битной шиной;
- Микроконтроллеры Microchip PICmicro: PIC10xxx, PIC12xxx, PIC16xxx, PIC17Cxxx, PIC18xxx, dsPIC ⁽¹⁾⁽²⁾;
- Микроконтроллеры Scenix (Ubicom): SXxxx.

⁽¹⁾ Имеются дополнительные адаптеры для микросхем с типоразмером корпуса, отличным от DIL.

⁽²⁾ Имеется всего несколько адаптеров с количеством выводов более 40. Если требуемые адаптеры отсутствуют – используйте программаторы LabProg+, BeeProg, JetProg.



Разъем ISP

- Serial E(E)PROM: серии I²C;
- Микроконтроллеры Atmel: AT89Sxxx, AT90Sxxxx, ATtiny, ATmega;
- Микроконтроллеры Cypress: CY8C2xxxx;
- Микроконтроллеры Elan: EM78Pxxx;
- Микроконтроллеры EM Microelectronic: с 4/8-битной шиной;
- Микроконтроллеры Microchip PICmicro: PIC10xxx, PIC12xxx, PIC16xxx, PIC17xxx, PIC18xxx, dsPIC;
- Микроконтроллеры Philips: серии LPC;

Тестирование микросхем

- Static RAM: 6116...624000.

Скорость программирования

Микросхема	Действие	Время
27C010	программирование и верификация	24
AT29C040A	программирование и верификация	32
AM29F040	программирование и верификация	69
PIC16C67	программирование и верификация	10
PIC18F452	программирование и верификация	7
AT89C52	программирование и верификация	17
PIC16F876A	программирование и верификация	5
PIC12C508	программирование и верификация	3

Замечание: Измерение скорости программирования производилось на тестовом компьютере P4, 2,4 GHz, порт USB версии 2,0, Установленная операционная система MS Windows XP.

Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

- выбор необходимой микросхемы по типу, производителю или любому фрагменту имеющейся информации;
- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- редактирование и установка конфигурационных битов;
- тестирование на ошибочные биты;
- подсчет контрольной суммы.

Операции контроля микросхем

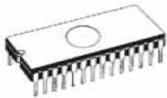
- тест на правильную ориентацию микросхемы в гнезде программатора ZIF;
- тест на наличие надежного контакта по всем выводам микросхемы;
- проверка соответствия настроек для выбранной микросхемы по идентификатору.

Специальные операции с микросхемами

- автоматическая генерация идентификаторов (серийных номеров) при программировании группы однотипных микросхем;
- статистика программирования;
- режим счетчика.

Операции с файлами

- автоматическое определение формата файлов данных.



Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение.

Поддерживаемые форматы данных

- Unformatted binary (BIN (raw));
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX;
- Altera POF, JEDEC (версии 3.0.A), ABEL, CUPL, PALASM, TANGO PLD, OrCAD PLD, PLD Designer ISDATA.

Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

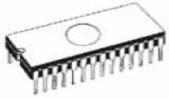
- напряжение питания 15–20V DC;
- потребляемый ток 500mA;
- максимальная потребляемая мощность 6W в активном режиме;
- максимальная потребляемая мощность 1,4W в неактивном режиме;
- размеры 160x95x35 мм;
- масса без дополнительных адаптеров 0,5 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**

Комплект поставки

- программатор SmartProg2;
- кабель USB;
- кабель ISP;
- диагностический модуль POD;
- чехол для гнезда ZIF;
- сетевой адаптер питания 15V DC/500mA, нестабилизированный;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep–Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.



MEMprog



Введение

MEMprog – специализированный программатор, поддерживающий работу со всеми типами 8–и и 16–битных микросхем памяти – EPROM, EEPROM, NVRAM, Flash EEPROM и Serial EEPROM. Имеется поддержка микросхем с пониженным напряжением питания. В качестве дополнительной функции программатор позволяет тестировать микросхемы Static RAM.

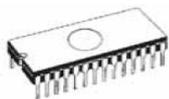
Разработанный на основе FPGA–микросхемы и использующий для связи с компьютером быстродействующий параллельный порт IEEE–1284, программатор **MEMprog** обеспечивает очень высокую скорость программирования среди разработок аналогичного класса других производителей.

Подключение программатора осуществляется к любым компьютерам семейства IBM PC 486 или более современным, оборудованным параллельным портом (LPT) без применения адаптеров или дополнительных плат расширения.

Аппаратная часть программатора содержит 40 мощных TTL–совместимых буферов, обеспечивающих выбор Н/Л логических уровней а также режимов PU/PD на любом контакте гнезда программатора ZIF socket. Современные быстродействующие буферы программатора обеспечивают надежную работу на самых высоких скоростях для всех поддерживаемых типов микросхем памяти, а также работу с микросхемами, использующими пониженное напряжение питания вплоть до 1,8 В.

Перед началом процесса программирования программатор автоматически проводит процедуру по проверке правильности установки программируемой микросхемы в гнездо ZIF socket и наличие надежного контакта со всеми выводами микросхемы. Эта функция позволяет избежать необратимого повреждения программируемой микросхемы из–за ошибок, допущенных пользователем при ее установке в программатор.

Проверка качества записи выбранной микросхемы проводится при предельно допустимых минимальном и максимальном напряжениях питания, в соответствии со спецификациями компаний–производителей микросхем, что позволяет в дальнейшем гарантировать надежную работу запрограммированной микросхемы в самых сложных условиях эксплуатации.



Интерфейс управляющей программы допускает работу с функциональными клавишами, сочетаниями ‘быстрых’ клавиш и пиктограммами панелей инструментов, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка). Обширная и подробная справочная система поможет найти ответ практически на любой вопрос. Поиск и выбор необходимой микросхемы в управляющей программе производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных.

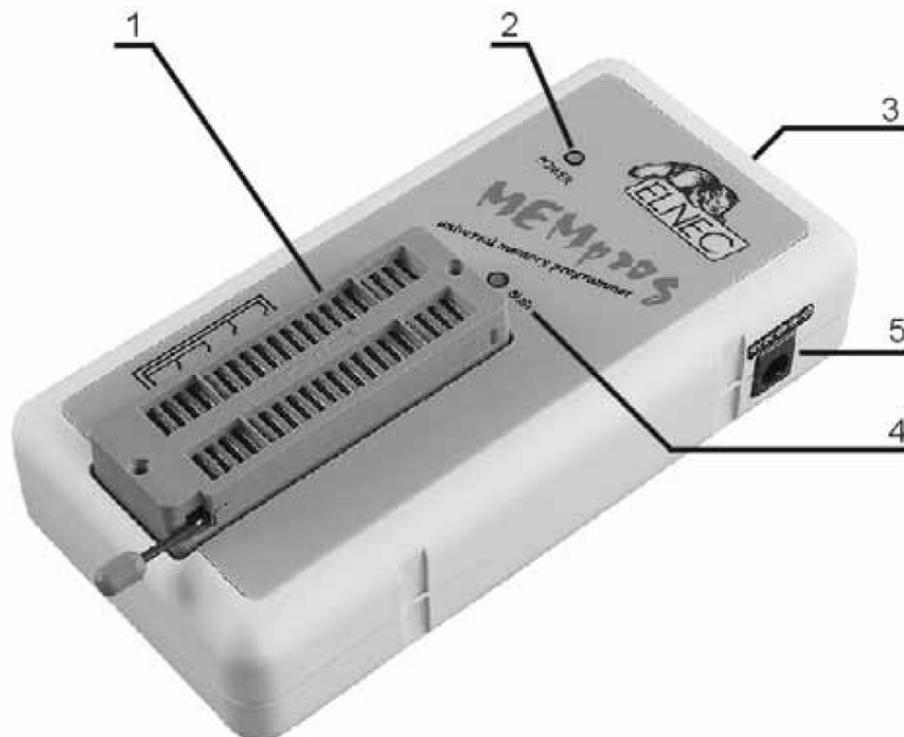
Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для партии однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера. Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

Для работы с программируемыми микросхемами в различных корпусах (PLCC, SOIC, SSOP, TSSOP, TQFP и некоторых других) разработаны специальные адаптеры, приобретаемые дополнительно.

Основные элементы программатора



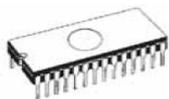
1. Гнездо ZIF socket на 40 контактов;
2. Индикатор питания;
3. Интерфейсный разъем для подключения кабеля LPT;
4. Индикатор активного состояния программатора;
5. Разъем питания для подключения сетевого адаптера.

Расположение контактов разъема питания программатора:



Для питания программатора используется сетевой адаптер, поставляемый в комплекте с программатором.

Замечание: Использование непредусмотренных конструкцией программатора сетевых адаптеров или блоков питания может привести к поломке программатора и выходу его из строя.



Подключение программатора к компьютеру

***Замечание:** При подключении программатора, строго следуйте данной инструкции. Это позволит избежать поломок и выхода из строя Вашего персонального компьютера и программатора. Нельзя использовать электронные или механические переключатели параллельного порта (LPT switches) для одновременной работы с программатором и печатающим устройством. Для подключения печатающего устройства установите в компьютер плату расширения с дополнительным параллельным портом (LPT). Плата расширения приобретается отдельно.*

Выключите питание персонального компьютера и отсоедините шнур питания от сети 220V. Сетевой адаптер программатора также должен быть отключен от питающей сети.

Подключите разъем интерфейсного кабеля, находящегося в комплекте с программатором к разъему программатора (3). Противоположный разъем интерфейсного кабеля подключите к свободному параллельному порту (LPT) компьютера. Обязательно закрепите предохранительные винты на разъемах интерфейсного кабеля – это позволит избежать выпадения кабеля из разъема и исключит возможные проблемы при работе с программатором.

Если компьютер оборудован единственным параллельным портом и к нему подключено печатающее или иное устройство, необходимо выключить печатающее устройство, отсоединить шнур питания от сети 220V, а затем отключить интерфейсный кабель этого устройства от компьютера.

Подключите разъем сетевого адаптера к гнезду питания программатора (5) и включите сетевой адаптер в питающую сеть. Наличие питания программатора можно проверить по активному состоянию индикатора питания (2).

Подключите компьютер к сети 220V и включите его. После загрузки операционной системы компьютера, запустите универсальную управляющую программу.

***Замечание:** Для исключения случаев повреждения порта LPT компьютера, не рекомендуется подключение и отключение интерфейсного кабеля при включенном компьютере.*

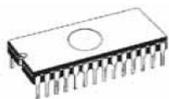


Если по каким-то причинам требуется подключить программатор к работающему компьютеру, воспользуйтесь нижеприведенной рекомендацией:

Для подключения программатора к работающему компьютеру, **вначале** к параллельному порту (LPT) компьютера **подключается интерфейсный кабель** программатора, а **затем** к питающей сети **подключается сетевой адаптер** программатора.

Если по каким-то причинам требуется отключить программатор от работающего компьютера, воспользуйтесь нижеприведенной рекомендацией:

Для отключения программатора от работающего компьютера, **вначале** **отключается сетевой адаптер** программатора от питающей сети, а **затем** **отключается интерфейсный кабель** программатора от параллельного порта (LPT) компьютера.



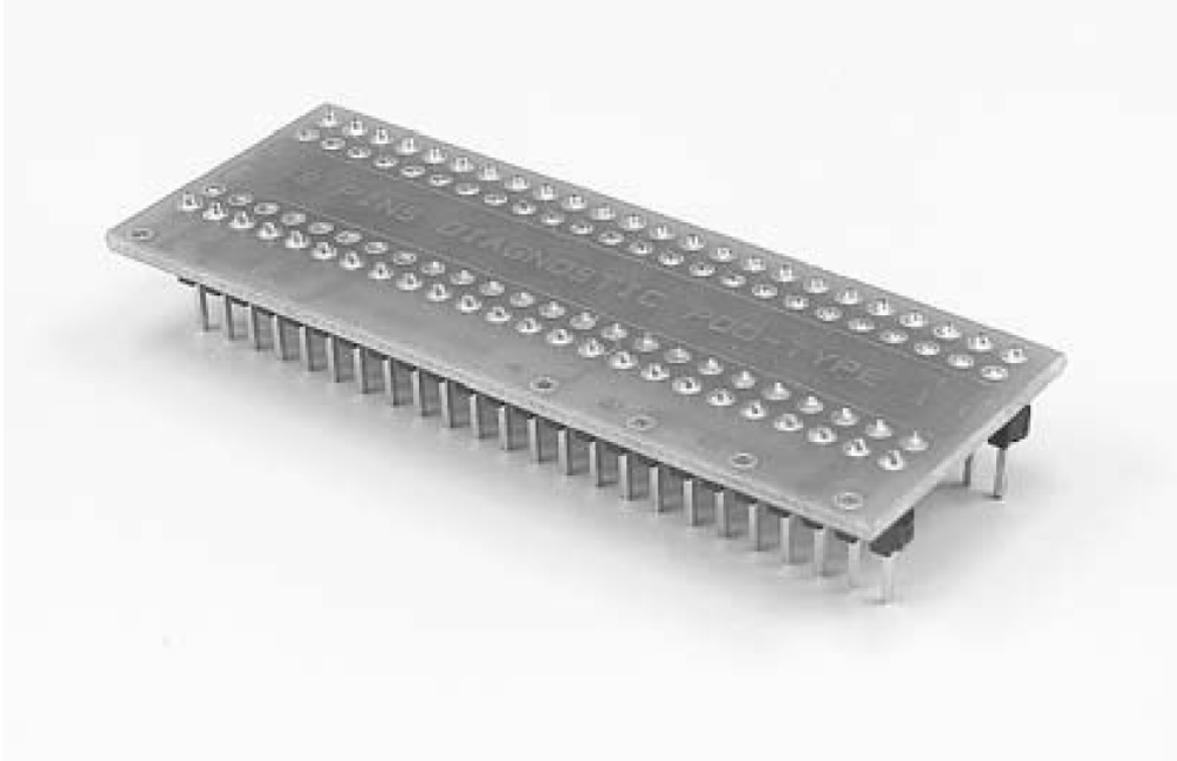
Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме-ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

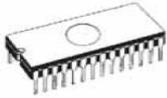
Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.

Самодиагностика и калибровка



При возникновении подозрений в некорректной работе программатора, рекомендуется использовать диагностический модуль (POD), входящий в комплект поставки программатора, для проведения процедур тестирования программатора и его калибровки. Тестирование и калибровку рекомендуется проводить каждые 6 месяцев.

Различные способы и методы тестирования и калибровки программатора доступны в меню **Программатор (Programmer)** управляющей программы.



Технические характеристики

База программатора, DAC

- два цифро–аналоговых преобразователя (DAC) для управления напряжениями VCCP и VPP;
- диапазон напряжения VCCP 0...7V/350mA;
- диапазон напряжения VPP 0...25V/200mA;
- интерфейс LPT, стандарта IEEE–1284, скорость передачи данных до 1 Мбит/сек;
- автокалибровка;
- самотестирование.

Разъем DIL ZIF socket

- 40–контактное гнездо модуля с нулевым усилением (ZIF socket) форм–фактора DIL для установки любых микросхем с количеством выводов до 40 и расстоянием между рядами выводов в диапазоне 300–600 mil;
- 40 мощных TTL–совместимых буферов, обеспечивающих выбор напряжений GND/VPP/VCC, логических уровней H/L/CLK, режимов PU/PD на любом выводе гнезда ZIF. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV);
- непрерывное тестирование перед каждой операцией программирования.

Типы поддерживаемых микросхем

Разъем DIL ZIF socket

- EPROM: NMOS/CMOS, 2708⁽³⁾, 27xxx, 27Cxxx, с 8/16–битной шиной и поддержкой семейств LV⁽¹⁾⁽²⁾;
- EEPROM: NMOS/CMOS, 28xxx, 28Cxxx, 27EExxx, с 8/16–битной шиной и поддержкой семейств LV⁽¹⁾⁽²⁾;
- Flash EPROM: 28Fxxx, 29Cxxx, 29Fxxx, 29BVxxx, 29LVxxx, 29Wxxx, 49Fxxx, с 8/16–битной шиной и поддержкой семейств LV⁽¹⁾⁽²⁾;
- Serial E(E)PROM: 24Cxxx, 24Fxxx, 25Cxxx, 45Dxxx, 59Cxxx, 25Fxxx, 25Pxxx, NVM3060, MDA206x⁽³⁾, 85xxx, 93Cxxx, с полной поддержкой семейств LV⁽¹⁾⁽²⁾;
- Configuration (EE)PROM: 37LVxx, XC17xxxx, AT17xxx, с полной поддержкой семейств LV⁽¹⁾;
- NV RAM: Dallas DSxxx, SGS/Inmos MKxxx, SIMTEK STKxxx, XICOR 2xxx, ZMD U63x.

Тестирование микросхем

- Static RAM: 6116...624000.

Скорость программирования

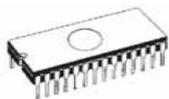
Микросхема	Действие	Время
27C010	программирование и верификация	23
AT29C040A	программирование и верификация	31
AM29F040	программирование и верификация	60

Замечание: Измерение скорости программирования производилось на тестовом компьютере P4, 2,4 GHz, режим LPT порта ECP, Установленная операционная система MS Windows XP.

⁽¹⁾ Имеются дополнительные адаптеры для микросхем с типоразмером корпуса, отличным от DIL.

⁽²⁾ Имеется всего несколько адаптеров с количеством выводов более 40. Если требуемые адаптеры отсутствуют – используйте программаторы LabProg+, VeeProg, JetProg.

⁽³⁾ Эти микросхемы являются устаревшими и программируются с помощью специальных адаптеров.



Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

- выбор необходимой микросхемы по типу, производителю или любому фрагменту имеющейся информации;
- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- редактирование и установка конфигурационных битов;
- тестирование на ошибочные биты;
- подсчет контрольной суммы.

Операции контроля микросхем

- тест на правильную ориентацию микросхемы в гнезде программатора ZIF;
- тест на наличие надежного контакта по всем выводам микросхемы;
- проверка соответствия настроек для выбранной микросхемы по идентификатору.

Специальные операции с микросхемами

- автоматическая генерация идентификаторов (серийных номеров) при программировании группы однотипных микросхем;
- статистика программирования;
- режим счетчика.

Операции с файлами

- автоматическое определение формата файлов данных.

Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение.

Поддерживаемые форматы данных

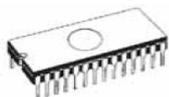
- Unformatted binary (BIN (raw));
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX.

Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

- напряжение питания 12–15V DC;
- потребляемый ток 500mA;
- максимальная потребляемая мощность 6W в активном режиме;
- максимальная потребляемая мощность 0,5W в неактивном режиме;
- размеры 137x65x40 мм;
- масса без дополнительных адаптеров 0,2 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**



Комплект поставки

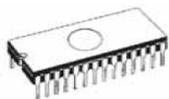
- программатор MEMprog;
- кабель LPT;
- диагностический модуль POD;
- чехол для гнезда ZIF;
- сетевой адаптер питания 12V DC/500mA, нестабилизированный;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep–Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.

MEMprogL





Введение

MEMprogL – специализированный программатор, поддерживающий работу с 8–битными микросхемами памяти EPROM, EEPROM, NVRAM, Flash EEPROM и Serial EEPROM. Имеется поддержка микросхем с пониженным напряжением питания. В качестве дополнительной функции программатор позволяет тестировать микросхемы Static RAM.

Разработанный на основе FPGA–микросхемы и использующий для связи с компьютером быстродействующий параллельный порт IEEE–1284, программатор **MEMprogL** обеспечивает очень высокую скорость программирования среди разработок аналогичного класса других производителей.

Подключение программатора осуществляется к любым компьютерам семейства IBM PC 486 или более современным, оборудованным параллельным портом (LPT) без применения адаптеров или дополнительных плат расширения.

Аппаратная часть программатора содержит 32 мощных TTL–совместимых буфера, обеспечивающих выбор Н/Л логических уровней а также режимов PU/PD на любом контакте гнезда программатора ZIF socket. Современные быстродействующие буферы программатора обеспечивают надежную работу на самых высоких скоростях для всех поддерживаемых типов микросхем памяти, а также работу с микросхемами, использующими пониженное напряжение питания вплоть до 1,8 В.

Перед началом процесса программирования программатор автоматически проводит процедуру по проверке правильности установки программируемой микросхемы в гнездо ZIF socket и наличие надежного контакта со всеми выводами микросхемы. Эта функция позволяет избежать необратимого повреждения программируемой микросхемы из–за ошибок, допущенных пользователем при ее установке в программатор.

Проверка качества записи выбранной микросхемы проводится при предельно допустимых минимальном и максимальном напряжениях питания, в соответствии со спецификациями компаний–производителей микросхем, что позволяет в дальнейшем гарантировать надежную работу запрограммированной микросхемы в самых сложных условиях эксплуатации.

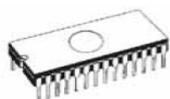
Интерфейс управляющей программы допускает работу с функциональными клавишами, сочетаниями 'быстрых' клавиш и пиктограммами панелей инструментов, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка). Обширная и подробная справочная система поможет найти ответ практически на любой вопрос. Поиск и выбор необходимой микросхемы в управляющей программе производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных.

Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для партии однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера. Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

Для работы с программируемыми микросхемами в различных корпусах (PLCC, SOIC, SSOP, TSSOP, TQFP и некоторых других) разработаны специальные адаптеры, приобретаемые дополнительно.



Основные элементы программатора



1. Гнездо ZIF socket на 32 контакта;
2. Индикатор питания;
3. Интерфейсный разъем для подключения кабеля LPT;
4. Индикатор активного состояния программатора;
5. Разъем питания для подключения сетевого адаптера.

Расположение контактов разъема питания программатора:



Для питания программатора используется сетевой адаптер, поставляемый в комплекте с программатором.

Замечание: Использование непредусмотренных конструкцией программатора сетевых адаптеров или блоков питания может привести к поломке программатора и выходу его из строя.

Подключение программатора к компьютеру

***Замечание:** При подключении программатора, строго следуйте данной инструкции. Это позволит избежать поломок и выхода из строя Вашего персонального компьютера и программатора. Нельзя использовать электронные или механические переключатели параллельного порта (LPT switches) для одновременной работы с программатором и печатающим устройством. Для подключения печатающего устройства установите в компьютер плату расширения с дополнительным параллельным портом (LPT). Плата расширения приобретается отдельно.*

Выключите питание персонального компьютера и отсоедините шнур питания от сети 220V. Сетевой адаптер программатора также должен быть отключен от питающей сети.

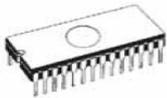
Подключите разъем интерфейсного кабеля, находящегося в комплекте с программатором к разъему программатора (3). Противоположный разъем интерфейсного кабеля подключите к свободному параллельному порту (LPT) компьютера. Обязательно закрепите предохранительные винты на разъемах интерфейсного кабеля – это позволит избежать выпадения кабеля из разъема и исключит возможные проблемы при работе с программатором.

Если компьютер оборудован единственным параллельным портом и к нему подключено печатающее или иное устройство, необходимо выключить печатающее устройство, отсоединить шнур питания от сети 220V, а затем отключить интерфейсный кабель этого устройства от компьютера.

Подключите разъем сетевого адаптера к гнезду питания программатора (5) и включите сетевой адаптер в питающую сеть. Наличие питания программатора можно проверить по активному состоянию индикатора питания (2).

Подключите компьютер к сети 220V и включите его. После загрузки операционной системы компьютера, запустите универсальную управляющую программу.

***Замечание:** Для исключения случаев повреждения порта LPT компьютера, не рекомендуется подключение и отключение интерфейсного кабеля при включенном компьютере.*



Если по каким-то причинам требуется подключить программатор к работающему компьютеру, воспользуйтесь нижеприведенной рекомендацией:

Для подключения программатора к работающему компьютеру, **вначале** к параллельному порту (LPT) компьютера **подключается интерфейсный кабель** программатора, а **затем** к питающей сети **подключается сетевой адаптер** программатора.

Если по каким-то причинам требуется отключить программатор от работающего компьютера, воспользуйтесь нижеприведенной рекомендацией:

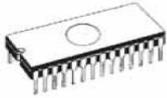
Для отключения программатора от работающего компьютера, **вначале** **отключается сетевой адаптер** программатора от питающей сети, а **затем** **отключается интерфейсный кабель** программатора от параллельного порта (LPT) компьютера.

Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме–ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.



Технические характеристики

База программатора, DAC

- два цифро–аналоговых преобразователя (DAC) для управления напряжениями VCCP и VPP;
- диапазон напряжения VCCP 0...7V/350mA;
- диапазон напряжения VPP 0...25V/200mA.

Разъем DIL ZIF socket

- 32–контактное гнездо модуля с нулевым усилением (ZIF socket) форм–фактора DIL для установки любых микросхем с количеством выводов до 32 и расстоянием между рядами выводов в диапазоне 300–600 mil;
- 32 мощных TTL–совместимых буфера, обеспечивающих выбор напряжений GND/VPP/VCC, логических уровней H/L/CLK, режимов PU/PD на любом выводе гнезда ZIF. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV).

Типы поддерживаемых микросхем

Разъем DIL ZIF socket

- EPROM: NMOS/CMOS, 27xxx, 27Cxxx, с 8–битной шиной и поддержкой семейств LV ⁽¹⁾ ⁽²⁾;
- EEPROM: NMOS/CMOS, 28xxx, 28Cxxx, 27EExxx, с 8–битной шиной и поддержкой семейств LV ⁽¹⁾ ⁽²⁾;
- Flash EPROM: 28Fxxx, 29Cxxx, 29Fxxx, 29BVxxx, 29LVxxx, 29Wxxx, 49Fxxx, с 8–битной шиной и поддержкой семейств LV ⁽¹⁾ ⁽²⁾;
- Serial E(E)PROM: 24Cxxx, 24Fxxx, 25Cxxx, 59Cxxx, 85xxx, 93Cxxx, с полной поддержкой семейств LV ⁽¹⁾;
- NV RAM: Dallas DSxxx, SGS/Inmos МКxxx, SIMTEK STKxxx, XICOR 2xxx, ZMD U63x.

Тестирование микросхем

- Static RAM: 6116...624000.

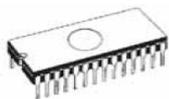
Скорость программирования

Микросхема	Действие	Время
27C010	программирование и верификация	37
AT29C040A	программирование и верификация	90
AM29F040	программирование и верификация	110

Замечание: Измерение скорости программирования производилось на тестовом компьютере P4, 2,4 GHz, режим LPT порта ECP, Установленная операционная система MS Windows XP.

⁽¹⁾ Имеются дополнительные адаптеры для микросхем с типоразмером корпуса, отличным от DIL.

⁽²⁾ Имеется всего несколько адаптеров с количеством выводов более 32. Если требуемые адаптеры отсутствуют – используйте программаторы SmartProg, SmartProg2, LabProg+, BeeProg, JetProg.



Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

- выбор необходимой микросхемы по типу, производителю или любому фрагменту имеющейся информации;
- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- редактирование и установка конфигурационных битов;
- тестирование на ошибочные биты;
- подсчет контрольной суммы.

Операции контроля микросхем

- проверка соответствия настроек для выбранной микросхемы по идентификатору.

Специальные операции с микросхемами

- автоматическая генерация идентификаторов (серийных номеров) при программировании группы однотипных микросхем;
- статистика программирования;
- режим счетчика.

Операции с файлами

- автоматическое определение формата файлов данных.

Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение.

Поддерживаемые форматы данных

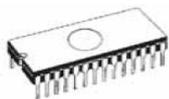
- Unformatted binary (BIN (raw));
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX.

Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

- напряжение питания 12–15V DC;
- потребляемый ток 500mA;
- максимальная потребляемая мощность 6W в активном режиме;
- максимальная потребляемая мощность 0,5W в неактивном режиме;
- размеры 137x65x40 мм;
- масса без дополнительных адаптеров 0,2 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**



Комплект поставки

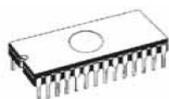
- программатор MEMprogL;
- кабель LPT;
- сетевой адаптер питания 12V DC/500mA, нестабилизированный;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep–Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.

T51prog





Введение

T51prog – это представитель новой линейки специализированных программаторов, разработанный компанией ELNEC в тесном сотрудничестве со специалистами компании Atmel W&M, благодаря чему аппаратные средства программатора корректно работают со всеми существующими на настоящий момент семействами микроконтроллеров этой компании.

T51prog – компактный и быстрый программатор, поддерживающий работу с микроконтроллерами семейств MCS51 и Atmel AVR, а также с микросхемами памяти с использованием алгоритмов I²C, SPI и Microwire, в корпусах DIP8–DIP40 без применения дополнительных адаптеров. Алгоритм последовательного программирования позволяет работать с микроконтроллерами семейств MCS51 и Atmel AVR, поддерживающими режим внутрисхемного программирования (ISP).

Разработанный на основе FPGA–микросхемы и использующий для связи с компьютером быстродействующий параллельный порт IEEE–1284, программатор **T51prog** обеспечивает очень высокую скорость программирования среди разработок аналогичного класса других производителей.

Подключение программатора осуществляется к любым компьютерам семейства IBM PC 486 или более современным, оборудованным параллельным портом (LPT) без применения специальных адаптеров или дополнительных плат расширения.

Аппаратная часть программатора содержит 40 мощных TTL–совместимых буферов, обеспечивающих выбор Н/Л логических уровней а также режимов PU/PD на любом контакте гнезда программатора ZIF socket. Современные быстродействующие буферы программатора обеспечивают надежную работу на самых высоких скоростях для всех поддерживаемых типов микросхем, а также работу с микросхемами, использующими пониженное напряжение питания вплоть до 1,8 В.

Перед началом процесса программирования программатор автоматически проводит процедуру по проверке правильности установки программируемой микросхемы в гнездо ZIF socket и наличие надежного контакта со всеми выводами микросхемы. Эта функция позволяет избежать необратимого повреждения программируемой микросхемы из–за ошибок, допущенных пользователем при ее установке в программатор.

Расширенная система самодиагностики программатора позволяет проверить параметры уровней напряжения на любом выводе гнезда ZIF socket программатора, а также работоспособность интерфейса связи программатора и компьютера. Проверка качества записи выбранной микросхемы проводится при предельно допустимых минимальном и максимальном напряжениях питания, в соответствии со спецификациями компаний–производителей микросхем, что позволяет в дальнейшем гарантировать надежную работу запрограммированной микросхемы в самых сложных условиях эксплуатации.

Поиск и выбор необходимой микросхемы производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных.

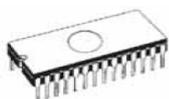
Интерфейс управляющей программы включает в себя несколько панелей инструментов, на которых расположены кнопки–пиктограммы, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка). Для облегчения работы так же предусмотрена поддержка функциональных клавиш и их сочетаний. В большинстве случаев возможно использование встроенной контекстной подсказки.

Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

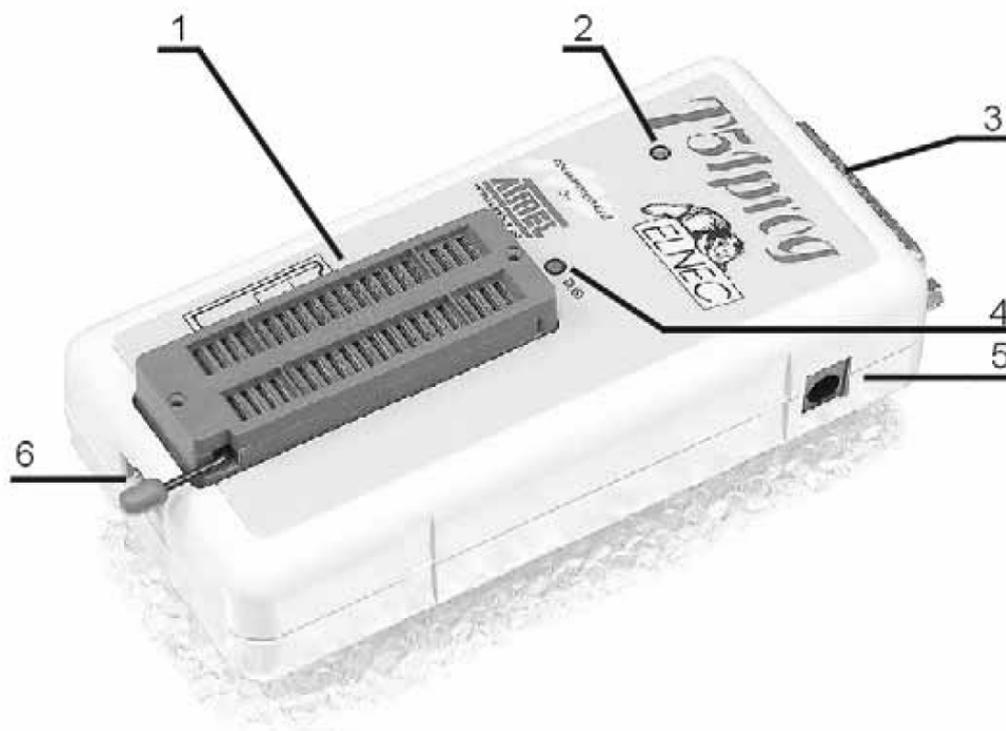
Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера. Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

Для работы с программируемыми микросхемами в различных корпусах (PLCC, SOIC, SSOP, TSSOP, TQFP, QFN (MLF) и некоторых других) разработаны специальные адаптеры, приобретаемые дополнительно.



Основные элементы программатора



1. Гнездо ZIF socket на 40 контактов;
2. Индикатор питания;
3. Интерфейсный разъем для подключения кабеля LPT;
4. Индикатор активного состояния программатора;
5. Разъем питания для подключения сетевого адаптера.

Расположение контактов разъема питания программатора:



Для питания программатора используется сетевой адаптер, поставляемый в комплекте с программатором.

Замечание: Использование непредусмотренных конструкцией программатора сетевых адаптеров или блоков питания может привести к поломке программатора и выходу его из строя.

Подключение программатора к компьютеру

***Замечание:** При подключении программатора, строго следуйте данной инструкции. Это позволит избежать поломок и выхода из строя Вашего персонального компьютера и программатора. Нельзя использовать электронные или механические переключатели параллельного порта (LPT switches) для одновременной работы с программатором и печатающим устройством. Для подключения печатающего устройства установите в компьютер плату расширения с дополнительным параллельным портом (LPT). Плата расширения приобретается отдельно.*

Выключите питание персонального компьютера и отсоедините шнур питания от сети 220V. Сетевой адаптер программатора также должен быть отключен от питающей сети.

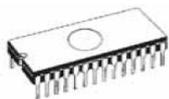
Подключите разъем интерфейсного кабеля, находящегося в комплекте с программатором к разъему программатора (3). Противоположный разъем интерфейсного кабеля подключите к свободному параллельному порту (LPT) компьютера. Обязательно закрепите предохранительные винты на разъемах интерфейсного кабеля – это позволит избежать выпадения кабеля из разъема и исключит возможные проблемы при работе с программатором.

Если компьютер оборудован единственным параллельным портом и к нему подключено печатающее или иное устройство, необходимо выключить печатающее устройство, отсоединить шнур питания от сети 220V, а затем отключить интерфейсный кабель этого устройства от компьютера.

Подключите разъем сетевого адаптера к гнезду питания программатора (5) и включите сетевой адаптер в питающую сеть. Наличие питания программатора можно проверить по активному состоянию индикатора питания (2).

Подключите компьютер к сети 220V и включите его. После загрузки операционной системы компьютера, запустите универсальную управляющую программу.

***Замечание:** Для исключения случаев повреждения порта LPT компьютера, не рекомендуется подключение и отключение интерфейсного кабеля при включенном компьютере.*



Если по каким-то причинам требуется подключить программатор к работающему компьютеру, воспользуйтесь нижеприведенной рекомендацией:

Для подключения программатора к работающему компьютеру, **вначале** к параллельному порту (LPT) компьютера **подключается интерфейсный кабель** программатора, а **затем** к питающей сети **подключается сетевой адаптер** программатора.

Если по каким-то причинам требуется отключить программатор от работающего компьютера, воспользуйтесь нижеприведенной рекомендацией:

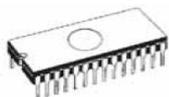
Для отключения программатора от работающего компьютера, **вначале** **отключается сетевой адаптер** программатора от питающей сети, а **затем** **отключается интерфейсный кабель** программатора от параллельного порта (LPT) компьютера.

Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме–ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

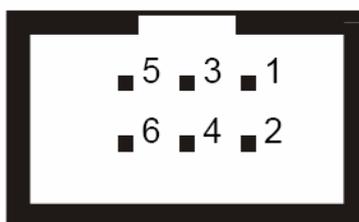
Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.



Внутрисхемное программирование (ISP)

Назначение контактов разъема ISP зависит в каждом конкретном случае от выбранной для программирования микросхемы (режим ISP для данного типа микросхемы устанавливается при его выборе из списка поддерживаемых микросхем). Более подробная информация может быть получена в меню **Микросхема–Информация о ИС (Device–Device Info)** управляющей программы или с помощью сочетания клавиш <Ctrl+F1>.

Нумерация контактов разъема ISP (вид спереди):



Кабель ISP для программатора T51prog:

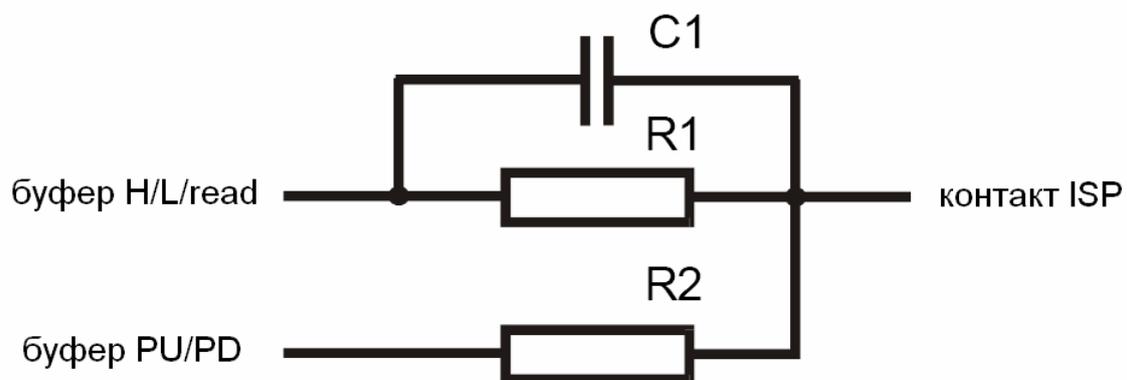


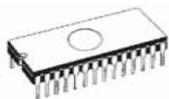
Замечание: Контакт №1 на кабеле для внутрисхемного программирования помечен символом ‘▲’.

Для исключения повреждения и выхода из строя программируемой микросхемы при использовании режима ISP, выполняйте следующие правил:

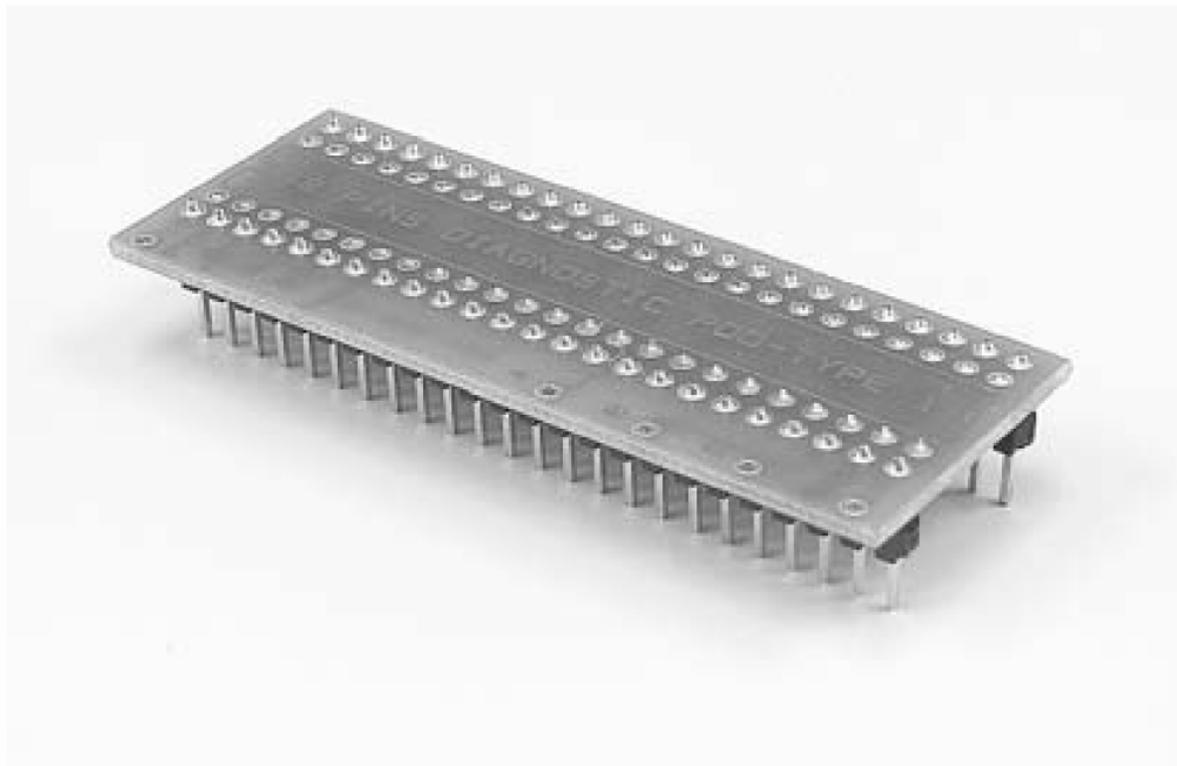
- при программировании микросхемы в режиме ISP не устанавливайте одновременно другую микросхему в гнездо ZIF socket;
- при программировании микросхемы в гнезде ZIF socket, не подключайте кабель ISP к программатору;
- используйте кабель ISP входящий в комплект программатора. Применение кабелей иной длины или кабелей, изготовленных из других материалов, может привести к нестабильному программированию или полной невозможности программирования микросхемы Вашим программатором;
- программатор поддерживает программирование микросхем, однако программируемые системы не поддерживают программатор;
- программатор контролирует величину напряжения программирования для выбранной микросхемы. Процесс программирования не будет выполняться, если конструкция программируемой системы оказывает влияние на величину напряжения программирования.

Принципиальная схема построения узла ISP (буферы H/L/read):





Самодиагностика и калибровка



При возникновении подозрений в некорректной работе программатора, рекомендуется использовать диагностический модуль (POD), входящий в комплект поставки программатора, для проведения процедур тестирования программатора и его калибровки. Тестирование и калибровку рекомендуется проводить каждые 6 месяцев.

Различные способы и методы тестирования и калибровки программатора доступны в меню **Программатор (Programmer)** управляющей программы.

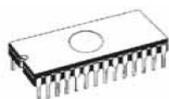
Технические характеристики

Разъем DIL ZIF socket

- интерфейс LPT, стандарта IEEE–1284, скорость передачи данных до 1 Мбит/сек;
- 40–контактное гнездо модуля с нулевым усилением (ZIF socket) форм–фактора DIL для установки любых микросхем с количеством выводов до 40 и расстоянием между рядами выводов в диапазоне 300–600 mil;
- два цифро–аналоговых преобразователя (DAC) для управления напряжениями VCCP и VPP;
- диапазон напряжения VCCP 0...8V/250mA;
- диапазон напряжения VPP 0...16V/150mA;
- 40 мощных TTL–совместимых буферов, обеспечивающих выбор напряжений GND/VPP/VCC, логических уровней H/L/CLK, режимов PU/PD на любом выводе гнезда ZIF. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV).

Разъем ISP

- 6–контактный разъем, с ключом, предотвращающим неправильную установку кабеля ISP;
- 4 TTL–совместимых буфера обеспечивающих выбор H/L/CLK логических уровней а также режимов PU/PD. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV);
- Расположение выводов соответствует спецификации компании Atmel AN AVR910.



Типы поддерживаемых микросхем

Разъем DIL ZIF socket

- микроконтроллеры MCS51: 87Cxxx, 87LVxxx, 89Cxxx, 89Sxxx, 89LVxxx, LPC Atmel, Atmel W&M, Intel, Philips, SST, Winbond ⁽¹⁾;
- микроконтроллеры Atmel AVR: ATtiny, AT90Sxxx, ATmega ⁽¹⁾;
- Serial E(E)PROM: 24Cxxx, 24Fxxx, 25Cxxx, 59Cxxx, 85xxx, 93Cxxx.

Разъем ISP

- микроконтроллеры Atmel: AT89Sxxx, AT90Sxxx, ATtiny, ATmega;
- микроконтроллеры Philips: поддержка серий LPC.

Скорость программирования

Микросхема	Действие	Время
AT89C52	программирование и верификация	15
T87C5111	программирование и верификация	14

Замечание: Измерение скорости программирования производилось на тестовом компьютере P4, 2,4 GHz, порт USB версии 2,0, Установленная операционная система MS Windows XP.

⁽¹⁾ Имеются дополнительные адаптеры для микросхем с типоразмером корпуса, отличным от DIL.

Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

- выбор необходимой микросхемы по типу, производителю или любому фрагменту имеющейся информации;
- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- редактирование и установка конфигурационных битов;
- тестирование на ошибочные биты;
- подсчет контрольной суммы.

Операции контроля микросхем

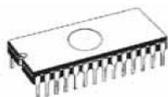
- тест на правильную ориентацию микросхемы в гнезде программатора ZIF;
- тест на наличие надежного контакта по всем выводам микросхемы;
- проверка соответствия настроек для выбранной микросхемы по идентификатору.

Специальные операции с микросхемами

- автоматическая генерация идентификаторов (серийных номеров) при программировании группы однотипных микросхем;
- статистика программирования;
- режим счетчика.

Операции с файлами

- автоматическое определение формата файлов данных.



Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение.

Поддерживаемые форматы данных

- Unformatted binary (BIN (raw));
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX.

Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

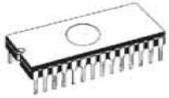
- напряжение питания 12–15V DC;
- потребляемый ток 500mA;
- максимальная потребляемая мощность 5W в активном режиме;
- размеры 137x65x40 мм;
- масса без дополнительных адаптеров 0,2 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**

Комплект поставки

- программатор T51prog;
- кабель LPT;
- кабель ISP;
- диагностический модуль POD;
- чехол для гнезда ZIF;
- сетевой адаптер питания 12V DC/500mA, нестабилизированный;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep-Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.



PIKprog+



Введение

PIKprog+ – это представитель новой линейки специализированных программаторов, разработанных компанией ELNEC и использующих в основе своей работы последовательный и параллельный алгоритмы программирования. Универсальная управляющая программа поддерживает корректную работу программатора в среде любой операционной системы Windows 95/98/ME/NT/2000/XP.

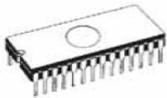
Несмотря на то, что программатор **PIKprog+** не является рекомендуемой средой разработки для продукции компании Microchip™, его создание проводилось в тесном сотрудничестве со специалистами компании Microchip™, благодаря чему аппаратные средства программатора корректно работают со всеми существующими на настоящий момент семействами микроконтроллеров PICmicro® с любыми типами корпусов и любым расположением выводов, а в будущем смогут поддерживать работу с новыми семействами микроконтроллеров PICmicro® путем обновления управляющей программы.

PIKprog+ – компактный и быстрый программатор, поддерживающий работу со всеми микроконтроллерами PICmicro® и микросхемами памяти с использованием алгоритмов I²C, SPI и Microwire, в корпусах DIP8–DIP40 без применения дополнительных адаптеров. Алгоритм последовательного программирования позволяет работать с микроконтроллерами семейства PICmicro®, поддерживающими режим внутрисхемного программирования (ISP).

Программатор **PIKprog+** представляет собой наилучшее соотношение цены и качества в своем классе, являясь надежным и простым в повседневной эксплуатации инструментом.

Разработанный на основе FPGA–микросхемы и использующий для связи с компьютером быстродействующий параллельный порт IEEE–1284, программатор **PIKprog+** обеспечивает очень высокую скорость программирования среди разработок аналогичного класса других производителей.

Подключение программатора осуществляется к любым компьютерам семейства IBM PC 486 или более современным, оборудованным параллельным портом (LPT) без применения специальных адаптеров или дополнительных плат расширения.



Аппаратная часть программатора содержит 40 мощных TTL–совместимых буферов, обеспечивающих выбор Н/Л логических уровней а также режимов PU/PD на любом контакте гнезда программатора ZIF socket. Современные быстродействующие буферы программатора обеспечивают надежную работу на самых высоких скоростях для всех поддерживаемых типов микросхем, а также работу с микросхемами, использующими пониженное напряжение питания вплоть до 1,8 В.

Перед началом процесса программирования программатор автоматически проводит процедуру по проверке правильности установки программируемой микросхемы в гнездо ZIF socket и наличие надежного контакта со всеми выводами микросхемы. Эта функция позволяет избежать необратимого повреждения программируемой микросхемы из–за ошибок, допущенных пользователем при ее установке в программатор.

Проверка качества записи выбранной микросхемы проводится при предельно допустимых минимальном и максимальном напряжениях питания, в соответствии со спецификацией компании Microchip™, что позволяет в дальнейшем гарантировать надежную работу запрограммированной микросхемы в самых сложных условиях эксплуатации.

Поиск и выбор необходимой микросхемы производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных.

Интерфейс управляющей программы включает в себя несколько панелей инструментов, на которых расположены кнопки–пиктограммы, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка). Для облегчения работы так же предусмотрена поддержка функциональных клавиш и их сочетаний. В большинстве случаев возможно использование встроенной контекстной подсказки.

Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

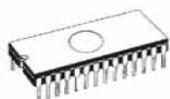
Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера.



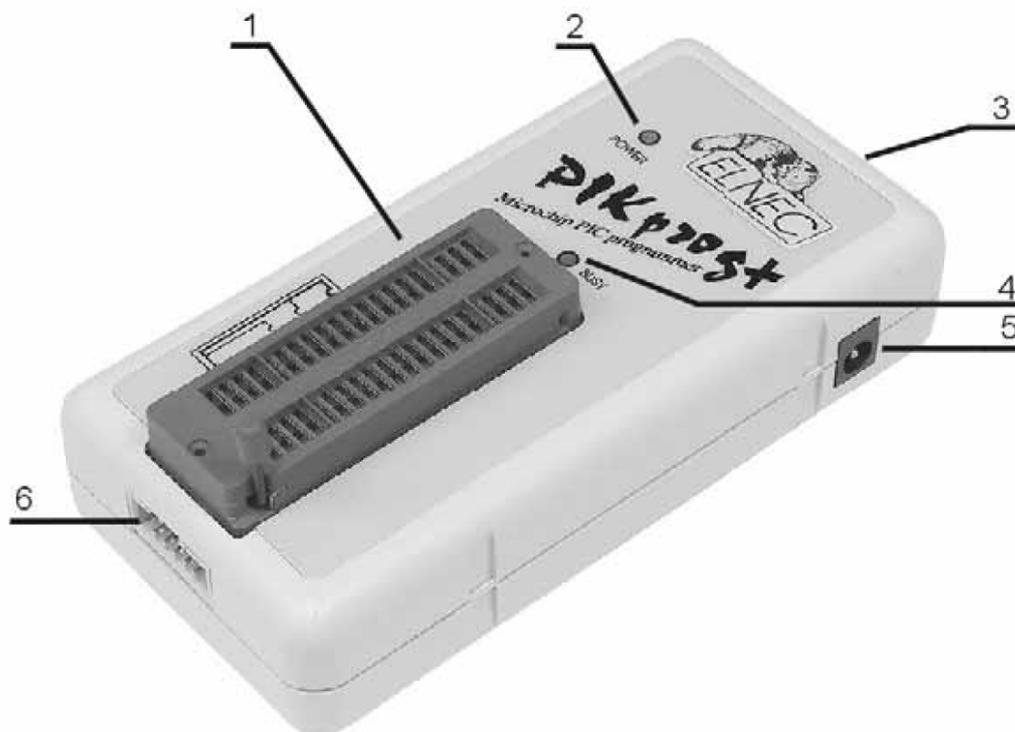
Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

Для работы с программируемыми микросхемами в различных корпусах (PLCC, SOIC, SSOP, TSSOP, TQFP и некоторых других) разработаны специальные адаптеры, приобретаемые дополнительно.



Основные элементы программатора



1. Гнездо ZIF socket на 40 контактов;
2. Индикатор питания;
3. Интерфейсный разъем для подключения кабеля LPT;
4. Индикатор активного состояния программатора;
5. Разъем питания для подключения сетевого адаптера.

Расположение контактов разъема питания программатора:



Для питания программатора используется сетевой адаптер, поставляемый в комплекте с программатором.

Замечание: Использование непредусмотренных конструкцией программатора сетевых адаптеров или блоков питания может привести к поломке программатора и выходу его из строя.

Подключение программатора к компьютеру

***Замечание:** При подключении программатора, строго следуйте данной инструкции. Это позволит избежать поломок и выхода из строя Вашего персонального компьютера и программатора. Нельзя использовать электронные или механические переключатели параллельного порта (LPT switches) для одновременной работы с программатором и печатающим устройством. Для подключения печатающего устройства установите в компьютер плату расширения с дополнительным параллельным портом (LPT). Плата расширения приобретается отдельно.*

Выключите питание персонального компьютера и отсоедините шнур питания от сети 220V. Сетевой адаптер программатора также должен быть отключен от питающей сети.

Подключите разъем интерфейсного кабеля, находящегося в комплекте с программатором к разъему программатора (3). Противоположный разъем интерфейсного кабеля подключите к свободному параллельному порту (LPT) компьютера. Обязательно закрепите предохранительные винты на разъемах интерфейсного кабеля – это позволит избежать выпадения кабеля из разъема и исключит возможные проблемы при работе с программатором.

Если компьютер оборудован единственным параллельным портом и к нему подключено печатающее или иное устройство, необходимо выключить печатающее устройство, отсоединить шнур питания от сети 220V, а затем отключить интерфейсный кабель этого устройства от компьютера.

Подключите разъем сетевого адаптера к гнезду питания программатора (5) и включите сетевой адаптер в питающую сеть. Наличие питания программатора можно проверить по активному состоянию индикатора питания (2).

Подключите компьютер к сети 220V и включите его. После загрузки операционной системы компьютера, запустите универсальную управляющую программу.

***Замечание:** Для исключения случаев повреждения порта LPT компьютера, не рекомендуется подключение и отключение интерфейсного кабеля при включенном компьютере.*



Если по каким-то причинам требуется подключить программатор к работающему компьютеру, воспользуйтесь нижеприведенной рекомендацией:

Для подключения программатора к работающему компьютеру, **вначале** к параллельному порту (LPT) компьютера **подключается интерфейсный кабель** программатора, а **затем** к питающей сети **подключается сетевой адаптер** программатора.

Если по каким-то причинам требуется отключить программатор от работающего компьютера, воспользуйтесь нижеприведенной рекомендацией:

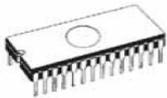
Для отключения программатора от работающего компьютера, **вначале** **отключается сетевой адаптер** программатора от питающей сети, а **затем** **отключается интерфейсный кабель** программатора от параллельного порта (LPT) компьютера.

Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме–ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

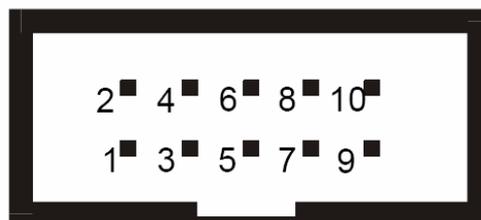
Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.



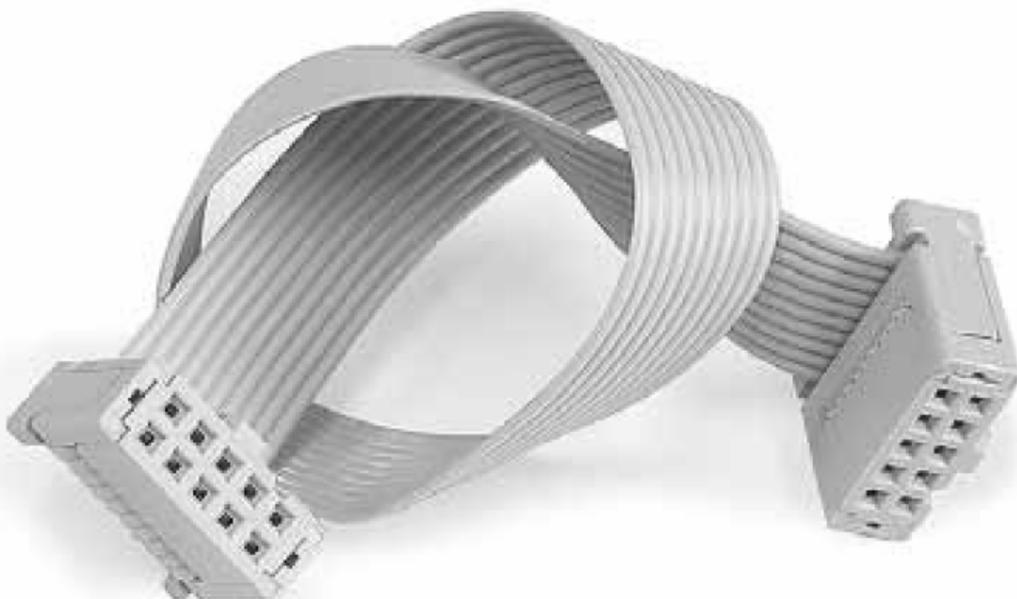
Внутрисхемное программирование (ISP)

Назначение контактов разъема ISP зависит в каждом конкретном случае от выбранной для программирования микросхемы (режим ISP для данного типа микросхемы устанавливается при его выборе из списка поддерживаемых микросхем). Более подробная информация может быть получена в меню **Микросхема–Информация о ИС (Device–Device Info)** управляющей программы или с помощью сочетания клавиш <Ctrl+F1>.

Нумерация контактов разъема ISP (вид спереди):



Кабель ISP для программатора PIKprog+:

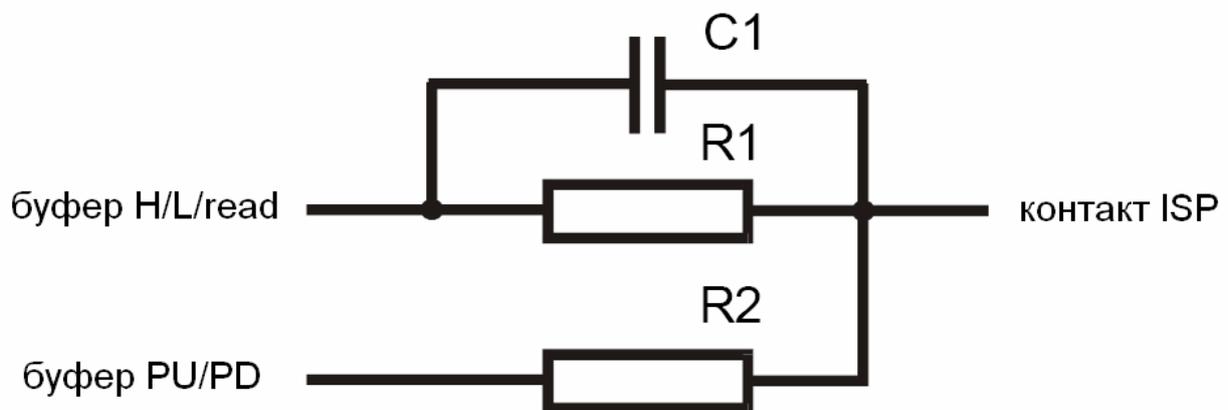


Замечание: Контакт №1 на кабеле для внутрисхемного программирования помечен символом '▲'.

Для исключения повреждения и выхода из строя программируемой микросхемы при использовании режима ISP, выполняйте следующие правил:

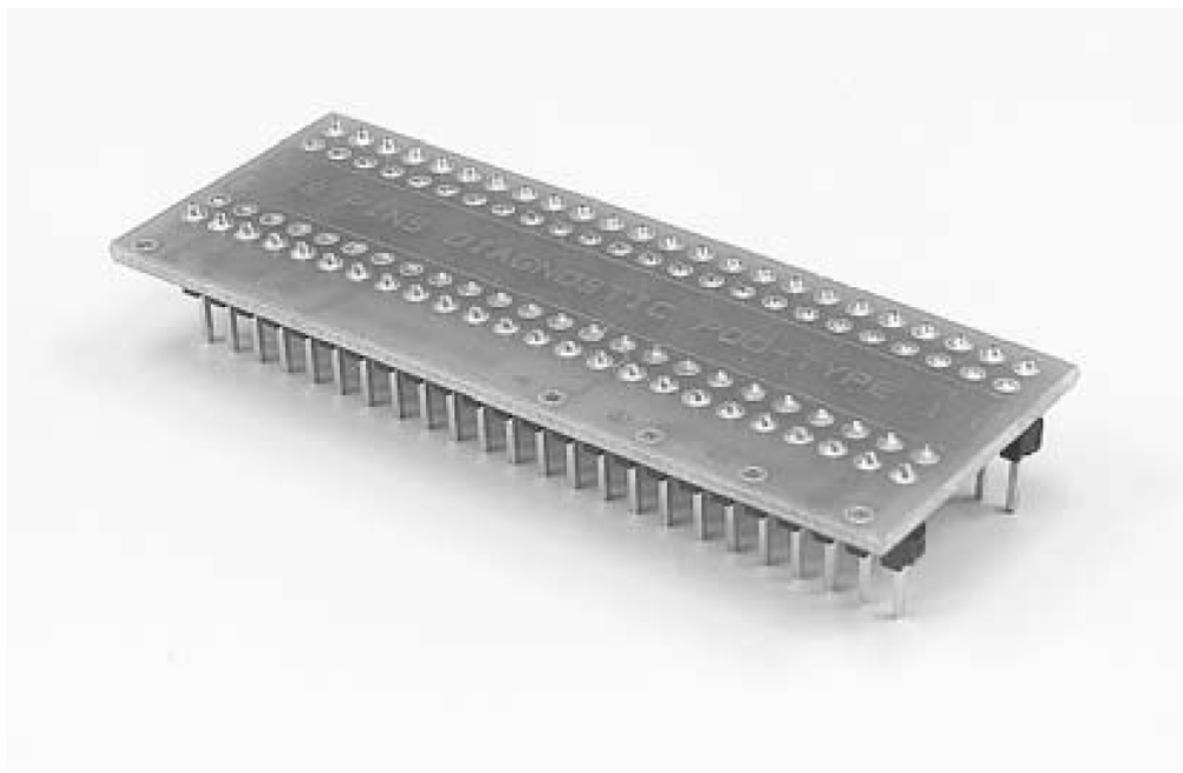
- при программировании микросхемы в режиме ISP не устанавливайте одновременно другую микросхему в гнездо ZIF socket;
- при программировании микросхемы в гнезде ZIF socket, не подключайте кабель ISP к программатору;
- используйте кабель ISP входящий в комплект программатора. Применение кабелей иной длины или кабелей, изготовленных из других материалов, может привести к нестабильному программированию или полной невозможности программирования микросхемы Вашим программатором;
- программатор поддерживает программирование микросхем, однако программируемые системы не поддерживают программатор;
- программатор контролирует величину напряжения программирования для выбранной микросхемы. Процесс программирования не будет выполняться, если конструкция программируемой системы оказывает влияние на величину напряжения программирования.

Принципиальная схема построения узла ISP (буферы H/L/read):





Самодиагностика и калибровка



При возникновении подозрений в некорректной работе программатора, рекомендуется использовать диагностический модуль (POD), входящий в комплект поставки программатора, для проведения процедур тестирования программатора и его калибровки. Тестирование и калибровку рекомендуется проводить каждые 6 месяцев.

Различные способы и методы тестирования и калибровки программатора доступны в меню **Программатор (Programmer)** управляющей программы.

Технические характеристики

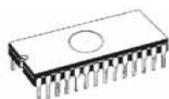
Разъем DIL ZIF socket

- интерфейс LPT, стандарта IEEE–1284, скорость передачи данных до 1 Мбит/сек;
- 40–контактное гнездо модуля с нулевым усилением (ZIF socket) форм–фактора DIL для установки любых микросхем с количеством выводов до 40 и расстоянием между рядами выводов в диапазоне 300–600 mil;
- два цифро–аналоговых преобразователя (DAC) для управления напряжениями VCCP и VPP;
- диапазон напряжения VCCP 0...7V/250mA;
- диапазон напряжения VPP 0...15V/150mA;
- 40 мощных TTL–совместимых буферов, обеспечивающих выбор напряжений GND/VPP/VCC, логических уровней H/L/CLK, режимов PU/PD на любом выводе гнезда ZIF. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV);
- непрерывное тестирование перед каждой операцией программирования;
- автокалибровка;
- самотестирование.

Разъем ISP

- 10–контактный разъем, с ключом, предотвращающим неправильную установку кабеля ISP;
- 5 TTL–совместимых буферов обеспечивающих выбор H/L/CLK логических уровней а также режимов PU/PD. Уровень H изменяется в диапазоне 1,8...5V (поддержка микросхем с низковольтным питанием LV);
- диапазон напряжения VCCP 2...7V/100mA *;
- диапазон напряжения VPP 2...25V/50mA;
- возможность интеллектуального выбора уровня напряжения VCCP.

* программатор не поддерживает питание программируемой системы. Если требуется поддержка – используйте программатор VeeProg.



Типы поддерживаемых микросхем

Разъем DIL ZIF socket

- микроконтроллеры PICmicro®: PIC10xxx, PIC12xxx, PIC14xxx, PIC16xxx, PIC17xxx, PIC18xxx, dsPIC⁽¹⁾;
- Serial E(E)PROM: 17Cxxx, 24Cxxx, 24Fxxx, 25Cxxx, 59Cxxx, 85xxx, 93Cxxx.

Разъем ISP

- микроконтроллеры PICmicro®: PIC10xxx, PIC12xxx, PIC16xxx, PIC17xxx, PIC18xxx, dsPIC.

Скорость программирования

Микросхема	Действие	Интерфейс	Время
PIC16C67	программирование и верификация	ZIF socket	12
PIC18F452	программирование и верификация	ZIF socket	8
PIC16F876A	программирование и верификация	ISP	6
PIC12C508	программирование и верификация	ISP	4

Замечание: Измерение скорости программирования производилось на тестовом компьютере P4, 2,4 GHz, порт USB версии 2,0, Установленная операционная система MS Windows XP.

⁽¹⁾ Имеются дополнительные адаптеры для микросхем с типоразмером корпуса, отличным от DIL.

Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

- выбор необходимой микросхемы по типу, производителю или любому фрагменту имеющейся информации;
- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- редактирование и установка конфигурационных битов;
- тестирование на ошибочные биты;
- подсчет контрольной суммы.

Операции контроля микросхем

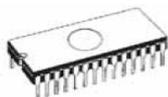
- тест на правильную ориентацию микросхемы в гнезде программатора ZIF;
- тест на наличие надежного контакта по всем выводам микросхемы;
- проверка соответствия настроек для выбранной микросхемы по идентификатору.

Специальные операции с микросхемами

- автоматическая генерация идентификаторов (серийных номеров) при программировании группы однотипных микросхем;
- статистика программирования;
- режим счетчика.

Операции с файлами

- автоматическое определение формата файлов данных.



Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение.

Поддерживаемые форматы данных

- Unformatted binary (BIN (raw));
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX.

Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

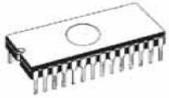
- напряжение питания 12–15V DC;
- потребляемый ток 500mA;
- максимальная потребляемая мощность 5W в активном режиме;
- размеры 137x65x40 мм;
- масса без дополнительных адаптеров 0,2 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**

Комплект поставки

- программатор PIKprog+;
- кабель LPT;
- кабель ISP;
- диагностический модуль POD;
- чехол для гнезда ZIF;
- сетевой адаптер питания 12V DC/500mA, нестабилизированный;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep-Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.



PIKprog



Введение

ПКprog – компактный и быстрый программатор, поддерживающий работу с микроконтроллерами семейств PICmicro® в корпусах DIP8–DIP40 без применения дополнительных адаптеров. В виде дополнительной функции, реализована поддержка программирования микросхем памяти с использованием алгоритмов I²C, SPI и Microwire.

Подключение программатора осуществляется к любым компьютерам семейства IBM PC 486 или более современным, оборудованным параллельным портом (LPT) без применения адаптеров или дополнительных плат расширения.

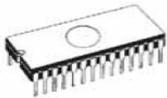
Интерфейс управляющей программы допускает работу с функциональными клавишами, сочетаниями ‘быстрых’ клавиш и пиктограммами панелей инструментов, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка).. Обширная и подробная справочная система поможет найти ответ практически на любой вопрос. Поиск и выбор необходимой микросхемы в управляющей программе производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных.

Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

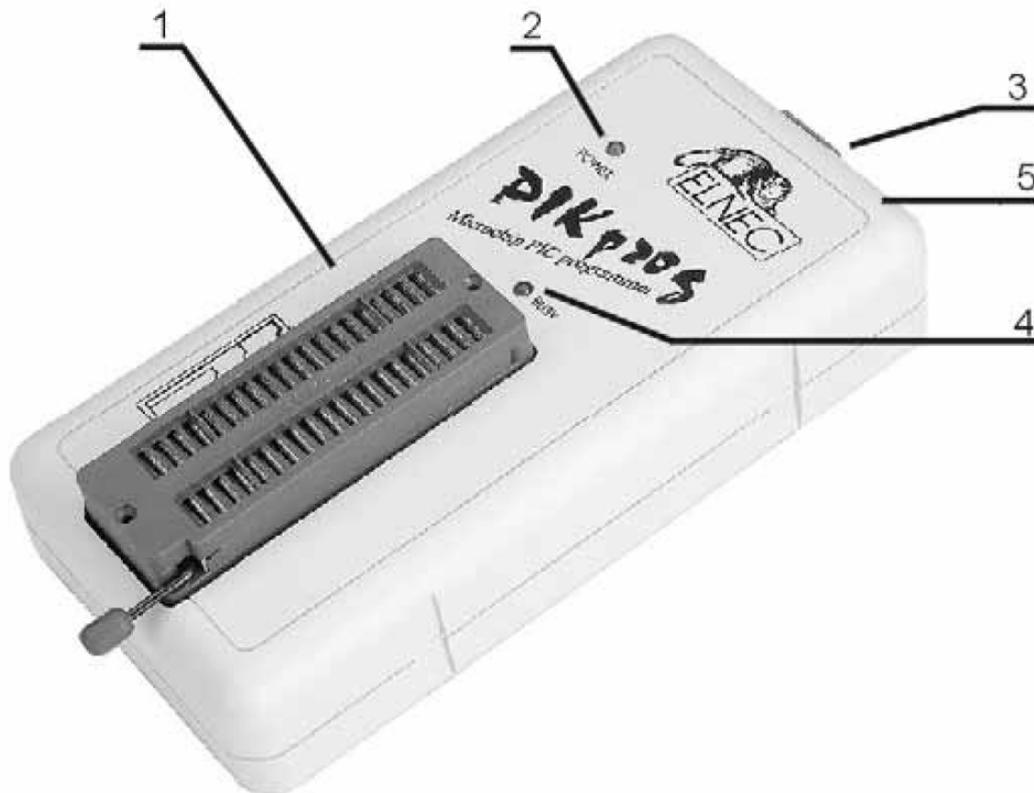
Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для партии однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера. Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

Для работы с программируемыми микросхемами в корпусах SOIC разработаны специальные адаптеры, приобретаемые дополнительно.



Основные элементы программатора



1. Гнездо ZIF socket на 40 контактов;
2. Индикатор питания;
3. Интерфейсный разъем для подключения кабеля LPT;
4. Индикатор активного состояния программатора;
5. Разъем питания для подключения сетевого адаптера.

Расположение контактов разъема питания программатора:



Для питания программатора используется сетевой адаптер, поставляемый в комплекте с программатором.

Замечание: Использование непредусмотренных конструкцией программатора сетевых адаптеров или блоков питания может привести к поломке программатора и выходу его из строя.

Подключение программатора к компьютеру

***Замечание:** При подключении программатора, строго следуйте данной инструкции. Это позволит избежать поломок и выхода из строя Вашего персонального компьютера и программатора. Нельзя использовать электронные или механические переключатели параллельного порта (LPT switches) для одновременной работы с программатором и печатающим устройством. Для подключения печатающего устройства установите в компьютер плату расширения с дополнительным параллельным портом (LPT). Плата расширения приобретается отдельно.*

Выключите питание персонального компьютера и отсоедините шнур питания от сети 220V. Сетевой адаптер программатора также должен быть отключен от питающей сети.

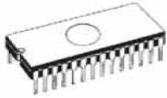
Подключите разъем интерфейсного кабеля, находящегося в комплекте с программатором к разъему программатора (3). Противоположный разъем интерфейсного кабеля подключите к свободному параллельному порту (LPT) компьютера. Обязательно закрепите предохранительные винты на разъемах интерфейсного кабеля – это позволит избежать выпадения кабеля из разъема и исключит возможные проблемы при работе с программатором.

Если компьютер оборудован единственным параллельным портом и к нему подключено печатающее или иное устройство, необходимо выключить печатающее устройство, отсоединить шнур питания от сети 220V, а затем отключить интерфейсный кабель этого устройства от компьютера.

Подключите разъем сетевого адаптера к гнезду питания программатора (5) и включите сетевой адаптер в питающую сеть. Наличие питания программатора можно проверить по активному состоянию индикатора питания (2).

Подключите компьютер к сети 220V и включите его. После загрузки операционной системы компьютера, запустите универсальную управляющую программу.

***Замечание:** Для исключения случаев повреждения порта LPT компьютера, не рекомендуется подключение и отключение интерфейсного кабеля при включенном компьютере.*



Если по каким-то причинам требуется подключить программатор к работающему компьютеру, воспользуйтесь нижеприведенной рекомендацией:

Для подключения программатора к работающему компьютеру, **вначале** к параллельному порту (LPT) компьютера **подключается интерфейсный кабель** программатора, а **затем** к питающей сети **подключается сетевой адаптер** программатора.

Если по каким-то причинам требуется отключить программатор от работающего компьютера, воспользуйтесь нижеприведенной рекомендацией:

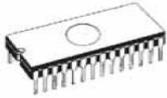
Для отключения программатора от работающего компьютера, **вначале** **отключается сетевой адаптер** программатора от питающей сети, а **затем** **отключается интерфейсный кабель** программатора от параллельного порта (LPT) компьютера.

Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме–ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.



Технические характеристики

Разъем DIL ZIF socket

- 40-контактное гнездо модуля с нулевым усилением (ZIF socket) форм-фактора DIL для установки любых микросхем с количеством выводов до 40 и расстоянием между рядами выводов в диапазоне 300–600 mil;
- TTL-совместимые буферы, обеспечивающие выбор логических уровней H/L/read на любом выводе гнезда ZIF.

Типы поддерживаемых микросхем

Разъем DIL ZIF socket

- микроконтроллеры PICmicro®: PIC12xxx, PIC14xxx, PIC16xxx, PIC18xxx (1).
- Serial E(E)PROM: 24Cxxx, 24Fxxx, 25Cxxx, 59Cxxx, 85xxx, 93Cxxx (2).

Замечание: Программатор не поддерживает режим ISP. Если требуется поддержка режима ISP, используйте соответствующие программаторы (BeeProg, SmartProg...).

(1) *Неполная поддержка микроконтроллеров семейства PIC18xxx. Если требуется поддержка специальных микроконтроллеров, отправьте образец разработчику программатора.*

(2) *Скорость программирования микросхем памяти ограничена – программирование микросхем памяти вспомогательная функция программатора.*

Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- редактирование и установка конфигурационных битов.

Специальные операции с микросхемами

- автоматическая генерация идентификаторов (серийных номеров) при программировании группы однотипных микросхем.

Операции с файлами

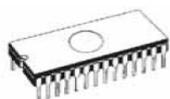
- автоматическое определение формата файлов данных.

Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение;
- подсчет контрольной суммы.

Поддерживаемые форматы данных

- Unformatted binary (BIN (raw));
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX.



Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

- напряжение питания 15–20V DC;
- потребляемый ток 500mA;
- максимальная потребляемая мощность 3W в активном режиме;
- размеры 137x65x40 мм;
- масса без дополнительных адаптеров 0,2 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**

Комплект поставки

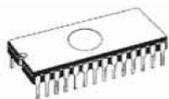
- программатор PIKprog;
- кабель LPT;
- сетевой адаптер питания 15V DC/500mA, нестабилизированный;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep–Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.

SEEprog





Введение

SEEprog – специализированный программатор, поддерживающий работу с микросхемами памяти EEPROM в корпусах DIP8, без применения дополнительных адаптеров. Программатор использует алгоритмы I²C, SPI, Microwire а также алгоритм для работы с цифровыми термометрами. Предусмотрен режим работы с микросхемами, использующими пониженное напряжение питания (LV) вплоть до 3,3 В.

Подключение программатора осуществляется к любым компьютерам семейства IBM PC 486 или более современным, оборудованным параллельным портом (LPT) без применения адаптеров или дополнительных плат расширения.

Интерфейс управляющей программы допускает работу с функциональными клавишами, сочетаниями ‘быстрых’ клавиш и пиктограммами панелей инструментов, которые вызывают наиболее часто используемые стандартные операции по работе с выбранными для программирования микросхемами (чтение, программирование, очистка, проверка). Обширная и подробная справочная система поможет найти ответ практически на любой вопрос. Поиск и выбор необходимой микросхемы в управляющей программе производится по ее названию, типу, маркировке, названию компании–производителя или по любому фрагменту этих данных.

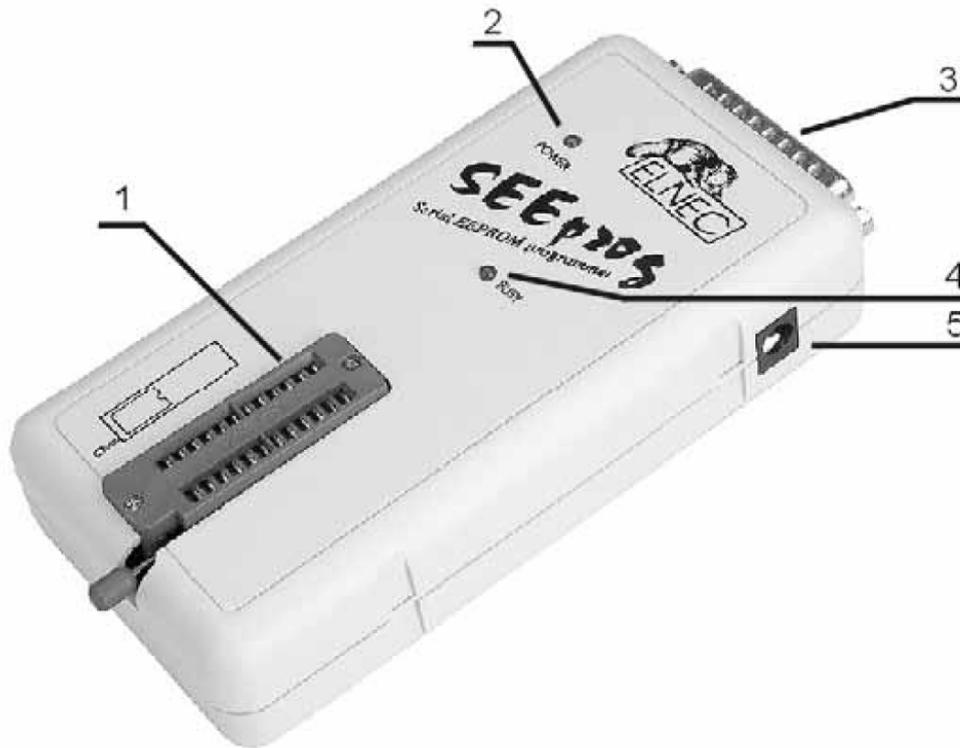
Управляющая программа поддерживает все известные форматы файлов данных. Имеется функция автоматического определения формата данных и преобразования загружаемого файла с данными.

Среди специальных возможностей управляющей программы, присутствует функция автоматической генерации идентификаторов или серийных номеров для партии однотипных программируемых микросхем, основанная на принципе увеличения идентификатора или серийного номера. Дополнительно предусмотрена возможность чтения таблиц идентификаторов или серийных номеров из специального файла данных.

Управляющая программа предоставляет исчерпывающую информацию о всех доступных режимах программирования выбранной микросхемы. Кроме того имеется встроенная библиотека с подробной расшифровкой маркировки, нанесенной на корпуса программируемых микросхем.

Для работы с микросхемами в корпусах SOIC разработаны специальные адаптеры, приобретаемые дополнительно.

Основные элементы программатора



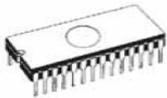
1. Гнездо ZIF socket на 24 контакта;
2. Индикатор питания;
3. Интерфейсный разъем для подключения кабеля LPT;
4. Индикатор активного состояния программатора;
5. Разъем питания для подключения сетевого адаптера.

Расположение контактов разъема питания программатора:



Для питания программатора используется сетевой адаптер, поставляемый в комплекте с программатором.

Замечание: Использование непредусмотренных конструкцией программатора сетевых адаптеров или блоков питания может привести к поломке программатора и выходу его из строя.



Подключение программатора к компьютеру

***Замечание:** При подключении программатора, строго следуйте данной инструкции. Это позволит избежать поломок и выхода из строя Вашего персонального компьютера и программатора. Нельзя использовать электронные или механические переключатели параллельного порта (LPT switches) для одновременной работы с программатором и печатающим устройством. Для подключения печатающего устройства установите в компьютер плату расширения с дополнительным параллельным портом (LPT). Плата расширения приобретается отдельно.*

Выключите питание персонального компьютера и отсоедините шнур питания от сети 220V. Сетевой адаптер программатора также должен быть отключен от питающей сети.

Подключите разъем интерфейсного кабеля, находящегося в комплекте с программатором к разъему программатора (3). Противоположный разъем интерфейсного кабеля подключите к свободному параллельному порту (LPT) компьютера. Обязательно закрепите предохранительные винты на разъемах интерфейсного кабеля – это позволит избежать выпадения кабеля из разъема и исключит возможные проблемы при работе с программатором.

Если компьютер оборудован единственным параллельным портом и к нему подключено печатающее или иное устройство, необходимо выключить печатающее устройство, отсоединить шнур питания от сети 220V, а затем отключить интерфейсный кабель этого устройства от компьютера.

Подключите разъем сетевого адаптера к гнезду питания программатора (5) и включите сетевой адаптер в питающую сеть. Наличие питания программатора можно проверить по активному состоянию индикатора питания (2).

Подключите компьютер к сети 220V и включите его. После загрузки операционной системы компьютера, запустите универсальную управляющую программу.

***Замечание:** Для исключения случаев повреждения порта LPT компьютера, не рекомендуется подключение и отключение интерфейсного кабеля при включенном компьютере.*

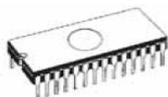


Если по каким-то причинам требуется подключить программатор к работающему компьютеру, воспользуйтесь нижеприведенной рекомендацией:

Для подключения программатора к работающему компьютеру, **вначале** к параллельному порту (LPT) компьютера **подключается интерфейсный кабель** программатора, а **затем** к питающей сети **подключается сетевой адаптер** программатора.

Если по каким-то причинам требуется отключить программатор от работающего компьютера, воспользуйтесь нижеприведенной рекомендацией:

Для отключения программатора от работающего компьютера, **вначале** **отключается сетевой адаптер** программатора от питающей сети, а **затем** **отключается интерфейсный кабель** программатора от параллельного порта (LPT) компьютера.



Установка программируемой микросхемы

После выбора в управляющей программе необходимой для программирования микросхемы, ее можно установить в гнездо ZIF socket (1) программатора. Для этого необходимо блокировочный рычаг гнезда перевести в вертикальное положение и установить микросхему в гнездо. Ориентация корпуса микросхемы производится по схеме-ключу, расположенному на верхней панели программатора возле гнезда. После установки микросхемы необходимо блокировочный рычаг перевести в горизонтальное положение, для закрепления в гнезде программатора выводов микросхемы.

Установка и извлечение программируемой микросхемы допускается только в неактивном состоянии программатора (индикатор активного состояния программатора (BUSY) погашен).

Замечание: Программатор не имеет защитных механизмов, предохраняющих от повреждения содержимое программируемой микросхемы в ряде критических ситуаций в режиме программирования (пропадание напряжения питающей сети). Кроме того, содержимое программируемой микросхемы будет повреждено в случае извлечения программируемой микросхемы из гнезда программатора, находящегося в активном состоянии. Несмотря на то, что перед началом программирования, программатор проверяет правильность установки программируемой микросхемы, всегда имеется риск ее повреждения из-за неправильной ориентации в гнезде программатора.

Технические характеристики

Разъем DIL ZIF socket

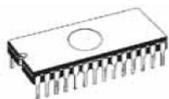
- 24-контактное гнездо модуля с нулевым усилением (ZIF socket) форм-фактора DIL для установки любых микросхем с количеством выводов до 24 и расстоянием между рядами выводов 300 mil;
- TTL-совместимые буферы обеспечивают корректные параметры логических уровней L и режима PU. Любой вывод может быть переопределен в состояние read;
- работа с микросхемами с низким напряжением питания (LV).

Типы поддерживаемых микросхем

Разъем DIL ZIF socket

- EEPROM I²C: 24Cxxx;
- EEPROM Microwire: 93Cxxx;
- EEPROM SPI: 25Cxxx, 25LVxxx, 25Fxxx, 25Pxxx;
- микросхемы памяти семейств LV;
- поддержка алгоритмов работы со специальными микросхемами (цифровые термодатчики).

Замечание: Программатор не поддерживает режим ISP. Если требуется поддержка режима ISP, используйте соответствующие программаторы (BeeProg, SmartProg...).



Программное обеспечение

использование программатором только тех алгоритмов, которые рекомендованы производителями программируемых микросхем;
регулярное обновление (приблизительно каждые 2 недели) программного обеспечения программатора.

Стандартные операции с микросхемами

- проверка очистки, чтение и верификация микросхемы;
- программирование микросхемы;
- очистка микросхемы;
- подсчет контрольной суммы.

Операции с буфером управляющей программы

- просмотр/редактирование, поиск/замена;
- заполнение, копирование, перемещение и обращение байтов;
- деление слов и двойных слов данных, подсчет контрольной суммы;
- пересылка содержимого буфера во внешнее приложение.

Поддерживаемые форматы данных

- Binary (BIN):
- HEX: Intel, Intel EXT, Motorola S-record, MOS, Exormax, Tektronix, ASCII-SPACE-HEX.

Системные требования

Более подробная информация находится в соответствующем разделе руководства.

Дополнительные характеристики

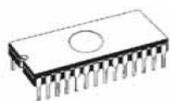
- напряжение питания 8–15V DC;
- потребляемый ток 50mA;
- максимальная потребляемая мощность 3W в активном режиме;
- размеры 137x65x40 мм;
- масса без дополнительных адаптеров 0,15 кг;
- рабочий диапазон температур 5...40 °C;
- влажность воздуха 20...80 %. **Обязательно отсутствие конденсата!**

Комплект поставки

- программатор SEEprog;
- кабель LPT;
- сетевой адаптер питания 12V DC/500mA, нестабилизированный;
- руководство пользователя;
- программное обеспечение;
- регистрационная карточка;
- упаковочная коробка.

Сервисное обслуживание

- услуга Keep–Current;
- услуга AlgOR;
- бесплатная техническая консультация (телефон/факс/электронная почта);
- бесплатное обновление программного обеспечения через Интернет.



Программное обеспечение

Введение

В комплект поставки программаторов ELNEC входит CD с различной информацией, включающей в себя управляющую программу Pg4uw, руководство пользователя, дополнительные утилиты и справочные материалы. Последняя версия управляющей программы и руководства пользователя размещена для свободного доступа в Интернет, на сайте компании ELNEC, по адресу: www.elnec.com.

Установка управляющей программы

Для установки управляющей программы необходимо CD, входящий в комплект с программатором поместить в привод оптических дисков компьютера. В зависимости от настроек компьютера, установка управляющей программы может начаться автоматически. Если этого не произошло, необходимо открыть содержимое CD и запустить файл 'setup.exe'. Далее необходимо следовать инструкциям, появляющимся в окнах 'Мастера установки программного обеспечения'.

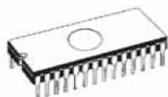
Программа Pg4uw является универсальной управляющей программой для всех типов программаторов, выпускаемых компанией ELNEC. Компания гарантирует безошибочную работу управляющей программы в среде операционных систем MS Windows 95/98/Me/NT/2000/XP.

Новые версии управляющей программы

Всегда рекомендуется использовать самую последнюю версию управляющей программы – это позволит оптимизировать имеющиеся функции программатора и добавить ряд новых возможностей.

Обновление управляющей программы

Скопируйте во временную папку файл 'PG4UWARC.exe' и запустите его. Спустя некоторое время во временной папке можно будет обнаружить список всех доступных файлов. Для установки управляющей программы запустите файл 'setup.exe'. После того как установка закончится, файлы из временной папки можно удалить.



Использование управляющей программы

Компания ELNEC гарантирует, что все файлы, расположенные на CD, входящем в комплект поставки программатора свободны от каких-либо разновидностей вирусных программ.

Запуск управляющей программы

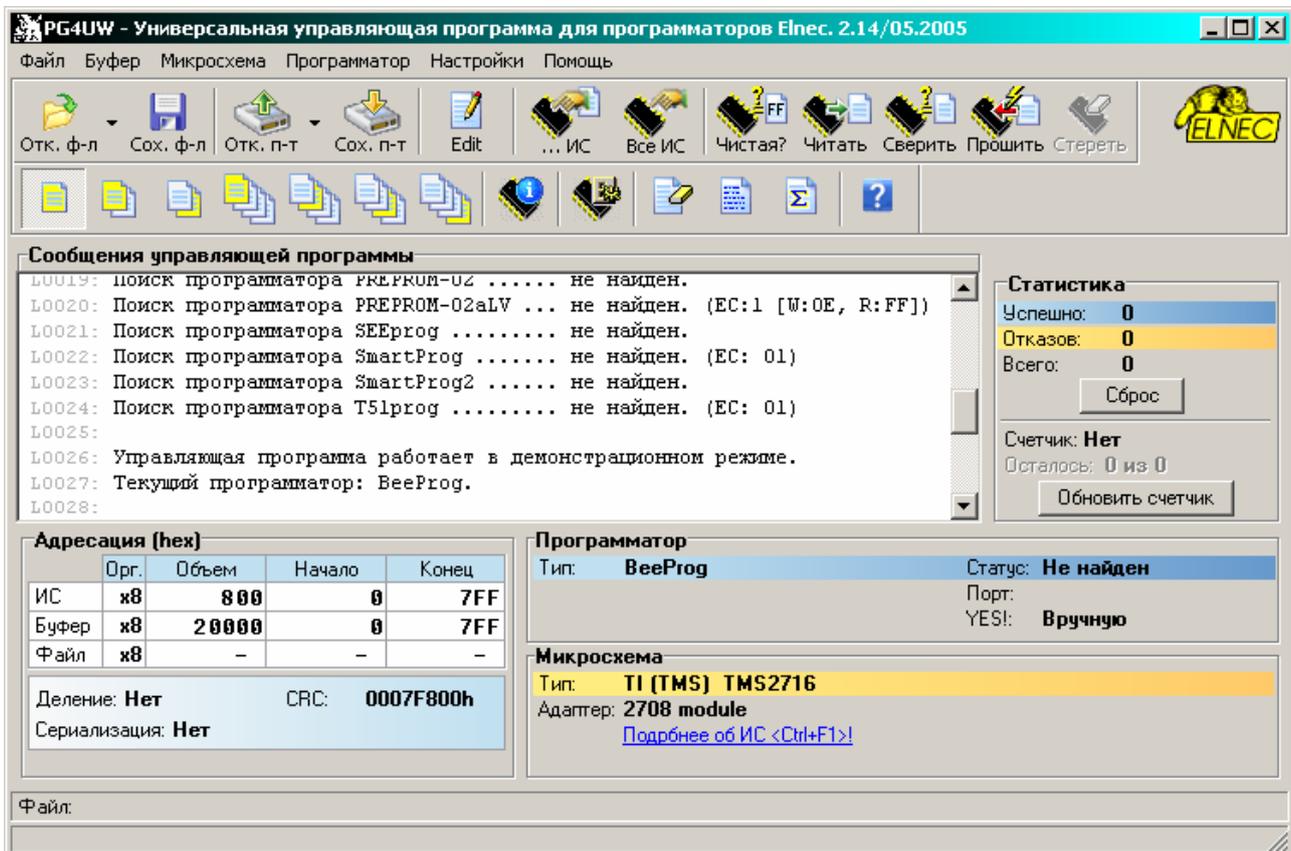
Для запуска управляющей программы необходимо одним из доступных способов запустить файл 'Pg4uw.exe'.

После запуска, управляющая программа проводит сканирование интерфейсных портов компьютера, с целью поиска доступных программаторов.

Если управляющая программа не может установить соединение с программатором, на экран будет выведено соответствующее сообщение с указанием номера ошибки. Кроме того, перед каждой операцией с микросхемой, проводится дополнительная проверка соединения компьютера и программатора.

***Замечание:** Управляющая программа циклически проводит проверку на целостность своих компонентов. Если целостность нарушена, будет выведено соответствующее сообщение. Проверка на целостность позволяет повысить надежность работы с управляющей программой.*

Главное окно управляющей программы



PG4UW - Универсальная управляющая программа для программаторов Elnes. 2.14/05.2005

Файл Буфер Микросхема Программатор Настройки Помощь

Отк. ф-л Сох. ф-л Отк. п-т Сох. п-т Edit ... ИС Все ИС Чистая? Читать Сверить Прошить Стереть

Сообщения управляющей программы

L0019: поиск программатора FKEFKUJ-UZ не найден.
L0020: Поиск программатора PREPROM-02aLV ... не найден. (EC:1 [W:0E, R:FF])
L0021: Поиск программатора SEEprog не найден.
L0022: Поиск программатора SmartProg не найден. (EC: 01)
L0023: Поиск программатора SmartProg2 не найден.
L0024: Поиск программатора T51prog не найден. (EC: 01)
L0025:
L0026: Управляющая программа работает в демонстрационном режиме.
L0027: Текущий программатор: VeeProg.
L0028:

Статистика

Успешно: 0
Отказов: 0
Всего: 0

Счетчик: Нет
Осталось: 0 из 0

Адресация (hex)

	Орг.	Объем	Начало	Конец
ИС	x8	800	0	7FF
Буфер	x8	20000	0	7FF
Файл	x8	-	-	-

Деление: Нет CRC: 0007F800h
Сериализация: Нет

Программатор

Тип: VeeProg Статус: Не найден
Порт:
YES!: Вручную

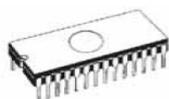
Микросхема

Тип: TI (TMS) TMS2716
Адаптер: 2708 module
[Подробнее об ИС <Ctrl+F1>](#)

Файл:

1. Заголовок окна (Header bar) – верхняя область окна управляющей программы, содержащая информацию о названии и номере версии.
2. Меню (Menu bar) – область под заголовком, содержащая основные функции.
3. Файл (Filename) – область в нижней части окна, содержащая название текущего файла, загруженного в буфер.
4. Программатор (Programmer) – область, отображающая информацию о программаторе.
5. Адресация (Addresses) – область, отображающая информацию об организации памяти выбранной микросхемы.
6. Микросхема (Device) – область, отображающая информацию о текущей микросхеме.
7. Строка помощи (Help bar) – область в нижней части окна, содержащая контекстную подсказку.

Работа с пунктами меню осуществляется обычным способом – с помощью мыши. Для быстрого вызова некоторых функций доступны зарезервированные сочетания функциональных клавиш.



Зарезервированные функциональные клавиши

<F1>	Помощь (Help)	Вызов справки
<F2>	Сохранить файл (Save)	Сохранить файл
<F3>	Открыть файл (Load)	Загрузить файл в буфер
<F4>	Буфер (Edit)	Редактировать буфер
<F5>	История ИС (Select/default)	Выбрать ИС из 10 последних
<Alt+F5>	Выбор ИС (Select/manual)	Выбрать ИС из всего списка
<F6>	Чистая? (Blank)	Проверка чистоты ИС
<F7>	Читать (Read)	Чтение ИС в буфер
<F8>	Сверить (Verify)	Верификация ИС
<F9>	Прошить (Program)	Программирование ИС
<Alt+Q>	Выход без сохранения (Exit without save)	Выход из программы без сохранения текущих настроек
<Alt+X>	Выход (Exit and save)	Выход из программы и сохранение текущих настроек
<Ctrl+F1>		Дополнительная информация о текущей ИС
<Ctrl+F2>	Стереть (Erase)	Очистка ИС
<Ctrl+Shift+F2>		Заполнение буфера случайными данными

Файл (FILE)

Меню **‘Файл (File)’** содержит команды для работы с папками и файлами. С помощью команд меню можно открывать и сохранять файлы проектов и файлы данных, изменять текущие папки для работы с файлами.

При работе с файлами данных можно самостоятельно изменять адреса блоков буфера для открытия и сохранения файлов данных. Файлы данных можно сохранять в различных форматах: **‘BIN’ (binary)**, **‘MOTOROLA’**, **‘MOS Technology’**, **‘Intel’ (extended) ‘HEX’**, **‘Tektronix’**, **‘ASCII space’**, **‘JEDEC’** и **‘POF’**.

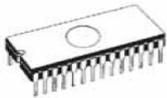
Файл / Открыть (FILE / LOAD)

Эта команда анализирует формат файла данных и загружает его в буфер управляющей программы. Имеется возможность самостоятельно выбрать формат открываемого файла: **‘BIN’ (binary)**, **‘MOTOROLA’**, **‘MOS Technology’**, **‘Tektronix’**, **‘Intel’ (extended) ‘HEX’**, **‘ASCII space’**, **‘JEDEC’** и **‘POF’**. Управляющая программа запоминает формат последнего удачно открытого файла данных.

Параметр **‘Автоопределение формата файла данных (Automatic file format recognition)’** позволяет выбрать режим автоматического определения формата файла данных. Если формат файла данных не определен, по умолчанию принимается бинарный формат файла (BIN (binary)).

Если параметр **‘Автоопределение формата файла данных (Automatic file format recognition)’** не отмечен, имеется возможность самостоятельного выбора подходящего формата файла данных из списка **‘Выбор формата файла данных (Selected file format)’**. Изменение настроек по умолчанию доступно в диалоговом окне **‘Настройки / Основные настройки (Options / General options)’**, в разделе **‘Ассоциация файлов (File options) / Определять формат файла данных (Load file format)’**.

Параметр **‘Смещение адреса буфера (Buffer offset for loading)’** позволяет загрузить файл данных форматов HEX и binary (BIN) в произвольную область буфера управляющей программы. Изменение настроек по умолчанию доступно в диалоговом окне **‘Настройки / Основные настройки (Options / General options)’**, в разделе **‘HEX-файл (Hex file options) / Установить отрицательное смещение адреса буфера для загрузки (Negative offset for loading)’**.



Параметр **‘Очистить буфер (Erase buffer before loading)’** позволяет очистить буфер управляющей программы перед загрузкой в него файла данных. Переменная для заполнения (очистки) буфера определяется в параметре **‘Значение (Value)’**. Изменение настроек по умолчанию доступно в диалоговом окне **‘Настройки / Основные настройки (Options / General options)’**, в разделе **‘HEX-файл (Hex file options)’**.

Если активен параметр **‘Обратить байты (Swap bytes)’**, дополнительно можно обратить пары байтов в сохраняемом файле. Данная функция полезна при открытии файлов в форматах **‘Motorola’**. Стандартный способ открытия файла **‘старший–первый, а младший–последний (Little endian)’**.

***Замечание:** Порядок распределения полей данных, в зависимости от требуемой цели, может быть организован двумя способами:*

*Первый способ состоит в том, что старшие значащие байты какой-либо структуры располагаются в памяти по младшему адресу, а младшие байты располагаются по старшему адресу, то есть старший–первый, а младший–последний. Этот способ также называется **‘Little endian’**.*

*Второй способ определяет другой порядок полей, то есть когда младшие значащие байты идут первыми, а старшие–последними. Этот способ называется **‘Big endian’**.*

Например, имеется двухбайтовое слово данных 4F52H, которое должно быть расположено по адресу 1000H. В зависимости от применяемого распределения полей, данные располагаются следующим образом:

Адреса	<i>Big endian</i>	<i>Little endian</i>
	Порядок	порядок
1000H	4FH	52H
1001H	52H	4FH

Файл / Сохранить (FILE / SAVE)

Эта команда позволяет сохранить в файл на диске данные из буфера управляющей программы. Формат сохраняемого файла может быть выбран самостоятельно. Отдельно можно указать границы блока буфера управляющей программы, подлежащего сохранению в параметрах **‘Начало (Buffer start)’** и **‘Конец (Buffer end)’**.

Файл может быть сохранен в одном из форматов: **‘BIN’ (binary)**, **‘MOTOROLA’**, **‘MOS Technology’**, **‘Tektronix’**, **‘Intel’ (extended) ‘HEX’**, **‘ASCII space’**, **‘JEDEC’** и **‘POF’**.

Если активен параметр **‘Обратить байты (Swap bytes)’** дополнительно можно обратить пары байтов в сохраняемом файле. Данная функция полезна при сохранении файлов в форматах **‘Motorola’**. Стандартный способ сохранения файла **‘старший–первый, а младший–последний (Little endian)’**.

Файл / Открыть проект (FILE / LOAD PROJECT)

Эта команда предназначена для открытия ранее сохраненного файла проекта, содержащего конфигурацию микросхемы и данные из буфера управляющей программы.

Диалоговое окно **‘Открыть проект (Load project)’** содержит область с описанием открываемого проекта. Описание включает в себя информацию о дате/времени создания проекта, используемой микросхеме а также дополнительную информацию о пользователе.

Описание проекта включает в себя:

- название микросхемы и информация о производителе.
- дата и время создания проекта.
- описание проекта.

Замечание: Для проектов использующих сериализацию:

*Настройки сериализации всегда будут приняты для открываемого проекта. Дополнительный файл с данными для сериализации будет использован, если он найден и данные из этого файла успешно прочитаны. Дополнительный файл с данными для сериализации будет связан с файлом проекта. Когда настройки сериализации из дополнительного файла загружены, текущие настройки сериализации для проекта отменены. Имя дополнительного файла с данными для сериализации совпадает с именем файла проекта, но имеет расширение **‘.sn’**. Дополнительный файл с данными для сериализации всегда расположен в папке **‘serialization\’**, находящейся в папке установки управляющей программы.*

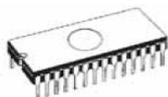
Пример:

Имя файла проекта: my_work.prj

Папка управляющей программы: c:\Program Files\Programmer

Местоположение дополнительного файла с данными для сериализации

c:\Program Files\Programmer\serialization\my_work.prj.sn



Дополнительный файл с данными для сериализации создается после первого успешного процесса программирования микросхемы. Обязательным условием является активированная функция сериализации для текущего проекта. Команда **‘Файл / Сохранить проект (File / Save project)’** удаляет дополнительный файл с данными для сериализации, если он существует и связан с текущим проектом.

Файл / Сохранить проект (FILE / SAVE PROJECT)

Эта команда предназначена для сохранения текущих настроек проекта в файл на диске. Сохраняемые настройки включают в себя конфигурацию микросхемы и данные из буфера управляющей программы. Сохраненный проект может быть прочитан в полном объеме командой меню **‘Файл / Открыть проект (File / Load project)’**.

Диалоговое окно **‘Сохранить проект (Save project)’** включает в себя область для внесения описания сохраняемого проекта и область, отображающую уже существующее описание проекта.

На панели диалогового окна имеется иконка–ключ. Это дополнительная команда, позволяющая установить пароль на файл проекта. Смотрите также: **‘Настройки / Защищенный режим (Options / Protected mode)’**.

Описание проекта включает в себя:

- название микросхемы и информация о производителе.
- дата и время создания проекта.
- описание проекта.

Первая область описания проекта содержит ранее сохраненную информацию о проекте (при первом создании файла проекта она пуста).

Вторая область описания проекта содержит информацию о микросхеме текущего проекта и дате/времени его создания.

Третья область описания предназначена для ввода записей пользователем и может содержать имя автора и другую полезную информацию.

Файл / Журнал файлов (FILE / RELOAD FILE)

Эта команда позволяет повторно открыть (обновить) текущий файл данных в буфере управляющей программы или открыть недавно использовавшийся файл данных.

Когда пользователь открывает файл данных, имя этого файла и путь к нему автоматически добавляются в Журнал файлов. Файлы данных отсортированы по мере их использования.

Для открытия файла данных из Журнала файлов необходимо:

- В меню ‘**Файл (File)**’ открыть пункт ‘**Журнал файлов (Reload file)**’.
- В списке недавно использовавшихся файлов данных выбрать требуемый файл и открыть его.

***Замечание:** Повторное открытие (обновление) файла данных приведет к его загрузке в буфер управляющей программы в том формате, который определен для этого типа файлов. Иначе будет произведено автоматическое определение формата файла данных.*

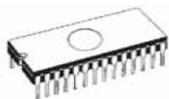
Файл / Журнал проектов (FILE / RELOAD PROJECT)

Эта команда позволяет повторно открыть (обновить) текущий проект из файла проекта или открыть недавно использовавшийся файл проекта.

Когда пользователь открывает файл проекта, имя этого файла и путь к нему автоматически добавляются в Журнал проектов. Файлы проектов отсортированы по мере их использования.

Для открытия файла проекта из Журнала проектов необходимо:

- В меню ‘**Файл (File)**’ открыть пункт ‘**Журнал проектов (Reload project)**’.
- В списке недавно использовавшихся файлов проектов выбрать требуемый файл и открыть его.



Файл / Свойства проекта (FILE / PROJECT OPTIONS)

Эта команда позволяет просмотреть и отредактировать текущие настройки открытого проекта. Свойства проекта содержат следующую информацию:

- название микросхемы и информация о производителе.
- дата и время создания проекта.
- описание проекта.

В свойствах проекта пользователь может изменить только описание проекта и обновить дату/время создания проекта. Информацию о микросхеме изменить нельзя.

Файл / Открыть Encryption table (FILE / LOAD ENCRYPTION TABLE)

Эта команда позволяет загрузить в специальную область буфера управляющей программы данные для крипто таблицы микросхемы из файла. Данные должны быть представлены в формате 'BIN' (binary).

Файл / Сохранить Encryption table (FILE / SAVE ENCRYPTION TABLE)

Эта команда позволяет сохранить часть специальной области буфера управляющей программы (крипто таблицу) в файл на диске. Файл сохраняется в формате 'BIN' (binary).

Файл / Выход без сохранения (FILE / EXIT WITHOUT SAVE)

Эта команда позволяет выйти из управляющей системы в оболочку операционной системы. Буфер управляющей программы и текущие настройки интерфейса не сохраняются.

Файл / Выход (FILE / EXIT AND SAVE)

Эта команда позволяет выйти из управляющей программы в оболочку операционной системы с сохранением текущих настроек интерфейса программы. В INI-файл сохраняется список 10 последних использовавшихся микросхем. Содержимое буфера управляющей программы не сохраняется.

Буфер (BUFFER)

Меню ‘**Буфер (Buffer)**’ содержит команды для работы с буфером управляющей программы и блоками буфера. С помощью команд меню можно заполнить часть буфера данными, очистить буфер, подсчитать контрольную сумму блоков буфера, а также просмотреть и отредактировать содержимое буфера (скопировать, переместить и обратить содержимое буфера).

Буфер / Просмотр/Редактирование (BUFFER / VIEW/EDIT)

Эта команда позволяет работать с буфером данных управляющей программы в режиме ‘**Просмотр (View)**’ или ‘**Редактирование (Edit)**’. Клавиши управления курсором можно использовать для выбора области, подлежащей редактированию. Для удобства использования, область редактирования подсвечивается другим цветом.

Для доступа к этой функции зарезервирована клавиша <F4>.

Работа с буфером (VIEW / EDIT BUFFER)

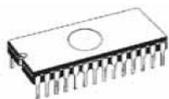
<F1> – показать текущее окно помощи.

<F2> – заполнить буфер или блок буфера выбранной самостоятельно последовательностью переменных. Начальный и конечный адреса блока буфера можно указать самостоятельно. Последовательность переменных может быть представлена в виде строки в формате ASCII или HEX.

<Ctrl+F2> – очистить всю область буфера управляющей программы путем записи по всем адресам буфера некоторой переменной. Значение этой переменной можно выбрать самостоятельно.

<Ctrl+Shift+F2> – заполнить всю область буфера последовательностью случайных переменных.

<F3> – скопировать выбранный блок данных в текущем буфере управляющей программы по новому адресу. Адрес, по которому будут размещены копируемые данные (новый адрес) не может находиться за пределами адресов текущего буфера управляющей программы.



<F4> – переместить блок буфера управляющей программы, с указанными самостоятельно начальным и конечным адресами, в новый адрес, указанный также самостоятельно. Значение нового адреса не может находиться за пределами текущего буфера управляющей программы. Блок буфера, в котором ранее находились перемещенные данные автоматически заполняется переменной, выбранной по умолчанию для очистки буфера управляющей программы.

<F5> – поменять местами младший и старший байты в соседней паре байтов. Начальный и конечный адреса блока буфера управляющей программы, подлежащие обращению можно указать самостоятельно. Количество байтов блока должно быть четным. Если это условие не выполняется, первым байтом с которого начнется операция обращения будет байт, расположенный по адресу ‘Начало блока’ + 1.

<F6> – переслать данные.

<F7> – искать строку переменных (максимальная длина строки 16 символов ‘ASCII’).

<F8> – искать строку переменных и заменить (максимальная длина строки 16 символов ‘ASCII’).

<F9> – изменить текущий адрес буфера.

<F10> – изменить режим ‘Просмотр/Редактирование (View/Edit).

<F11> – изменить режим отображения буфера между 8–битным и 16–битным. При изменении режима отображения также изменяется способ отображения текущих данных (8–битное или 16–битное отображение).

<F12> – подсчет контрольной суммы выделенного блока данных в буфере управляющей программы.

<Arrow keys> – клавиши управления курсором. Позволяют перемещаться по области данных вверх, вниз, влево и вправо.

<Home/End> – быстрый переход в начало или конец текущей строки данных.

<PgUp/PgDn> – быстрый переход на предыдущую или следующую страницу данных.



<Ctrl+PgUp/PgDn> – быстрый переход в начало или конец текущей страницы данных.

<Ctrl+Home/End> – быстрый переход в начало или конец буфера данных ИС.

<Shift+Home/End> – быстрый переход в начало или конец текущего буфера данных.

<Backspace> – клавиша ‘забой’, позволяет переместить курсор на одну позицию влево (назад).

Замечание: Символы 20H – FFH (режим ‘Редактировать’ (Edit), область ASCII), символы 0...9 и A...F (режим ‘Редактировать’ (Edit), область HEX) немедленно изменяют содержимое буфера управляющей программы.

Переслать данные (PRINT BUFFER)

Эта команда позволяет передать выделенный блок буфера управляющей программы во внешнее приложение, для дальнейшего сохранения в текстовый файл или для вывода на печатающее устройство. По умолчанию передача осуществляется в текстовый редактор ‘NOTEPAD.EXE’, так как этим редактором укомплектована любая версия операционной системы.

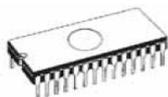
В диалоговом окне доступны следующие параметры:

‘**Начало блока** (Block start)’ – начальный адрес выделяемого блока данных буфера управляющей программы.

‘**Конец блока** (Block end)’ – конечный адрес выделяемого блока данных буфера управляющей программы.

‘**Внешний редактор** (External editor)’ – приложение, в которое будет передан выделенный блок данных. Допускается самостоятельно изменить приложение (текстовый редактор) на имеющееся в наличии, например редактор ‘WORDPAD.EXE’ или иной, удобный для пользователя.

Выбранный путь к исполняемому файлу внешнего редактора автоматически сохраняется в INI-файле управляющей программы.



Искать текст (FIND DIALOG BOX)

Область **‘Искать (Text to find)’** предназначена для ввода необходимого текста. Поиск начинается при клике на кнопку **‘Искать (Find)’**. Клик на кнопку **‘Отмена (Cancel)’** прекращает операцию поиска.

Текст может быть представлен в виде строки **‘ASCII’** или в виде последовательности **‘HEX’**.

‘Направление поиска (Direction)’ может быть определено **‘Вверх (Backward)’** или **‘Вниз (Forward)’** от текущего положения курсора.

‘Область поиска (Origin)’ может быть выбрана **‘От курсора (From cursor)’** – поиск идет от текущей позиции курсора или **‘Везде (Entire scope)’** – поиск идет по всей доступной области буфера.

Заменить текст (FIND & REPLACE DIALOG BOX)

Область **‘Искать (Text to find)’** предназначена для ввода необходимого текста.

Область **‘Заменить на (Replace with)’** предназначена для ввода текста на который будет произведена замена.

Текст может быть представлен в виде строки **‘ASCII’** или в виде последовательности **‘HEX’**.

‘Направление поиска (Direction)’ может быть определено **‘Вверх (Backward)’** или **‘Вниз (Forward)’** от текущего положения курсора.

‘Область поиска (Origin)’ может быть выбрана **‘От курсора (From cursor)’** – поиск идет от текущей позиции курсора или **‘Везде (Entire scope)’** – поиск идет по всей доступной области буфера.

Клик на кнопку **‘Отмена (Cancel)’** прекращает операцию поиска.

При клике на кнопку **‘Заменить (Replace)’** будет появляться диалоговое окно с запросом о подтверждении действия при обнаружении совпадения поиска.

Возможные варианты ответа:

- ‘Да (Yes)’ заменить совпадение и искать дальше.
- ‘Нет (No)’ не заменять совпадение и искать дальше.
- ‘**Заменить все (Replace All)**’ заменить все найденные совпадения автоматически.
- ‘**Прервать поиск (Abort search)**’ прервать операцию.

Работа с буфером микросхем ‘PLD’ (VIEW EDIT BUFFER FOR PLD)

<Ctrl+F2> – очистить всю область буфера управляющей программы путем записи по всем адресам буфера некоторой переменной. Значение этой переменной можно выбрать самостоятельно.

<Ctrl+Shift+F2> – заполнить всю область буфера последовательностью случайных переменных.

<F9> – изменить текущий адрес буфера.

<F10> – изменить режим работы с буфером ‘Просмотр/Редактирование’ (View/Edit).

<F11> – изменить режим отображения буфера между 1–битным и 8–битным. При изменении режима отображения также изменяется способ отображения текущих данных (‘битовое’ или ‘шестнадцатеричное’ отображение).

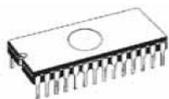
<Arrow keys> – клавиши управления курсором. Позволяют перемещаться по области данных вверх, вниз, влево и вправо.

<Home/End> – быстрый переход в начало или конец текущей строки данных.

<PgUp/PgDn> – быстрый переход на предыдущую или следующую страницу данных.

<Ctrl+PgUp/PgDn> – быстрый переход в начало или конец текущей страницы данных.

<Ctrl+Home/End> – быстрый переход в начало или конец редактируемого буфера данных.



<Backspace> – клавиша ‘забой’ переместить курсор на одну позицию влево (назад).

Замечание: Символы 20H – FFH (режим ‘Редактировать’ (Edit), область ASCII), символы 0...9 и A...F (режим ‘Редактировать’ (Edit), область HEX), символы 0 и 1 (режим ‘Редактировать’ (Edit), область битового отображения (1 bit)) немедленно изменяют содержимое буфера управляющей программы.

Буфер / Заполнить (BUFFER / FILL BLOCK)

Эта команда позволяет заполнить буфер или блок буфера заранее определенной последовательностью переменных. Начальный и конечный адреса блока буфера можно указать самостоятельно. Последовательность переменных синхронно представлена в виде строк в форматах ‘ASCII’ и ‘HEX’.

Буфер / Копировать (BUFFER / COPY BLOCK)

Эта команда позволяет скопировать выбранный блок данных в текущем буфере управляющей программы по новому адресу. Адрес, по которому будут размещены копируемые данные (новый адрес) не может находиться за пределами адресов текущего буфера управляющей программы.

Буфер / Переместить (BUFFER / MOVE BLOCK)

Эта команда позволяет переместить блок буфера управляющей программы, с указанными самостоятельно начальным и конечным адресами, в новый адрес, указанный также самостоятельно. Значение нового адреса не может находиться за пределами адресов текущего буфера управляющей программы. Блок буфера, в котором ранее находились перемещенные данные автоматически заполняется переменной, выбранной по умолчанию для очистки буфера управляющей программы.

Буфер / Обратить блок (BUFFER / SWAP BLOCK)

Эта команда позволяет поменять местами младший и старший байты в соседней паре байтов. Начальный и конечный адреса блока буфера управляющей программы, подлежащие обращению можно указать самостоятельно.

Буфер / Очистить (BUFFER / ERASE)

Эта команда позволяет очистить всю область буфера управляющей программы путем записи по всем адресам буфера некоторой переменной. Переменная может быть выбрана пользователем самостоятельно.

Для быстрого вызова этой команды из любого места управляющей программы зарезервирована комбинация клавиш <Ctrl+F2>.

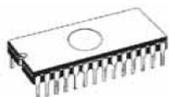
Буфер / Заполнить случайными данными (BUFFER / FILL RANDOM DATA)

Эта команда позволяет заполнить всю область буфера последовательностью случайных переменных.

Для быстрого вызова этой команды из любого места управляющей программы зарезервирована комбинация клавиш <Shift+Ctrl+F2>.

Буфер / Дублировать страницы памяти (BUFFER / DUPLICATE BUFFER CONTENT)

Эта команда позволяет дублировать содержимое памяти микросхемы–источника EPROM с малым объемом памяти во все страницы памяти микросхемы–получателя EPROM с заведомо большим объемом. Данная функция полезна в том случае если, например, используется имеющаяся в данный момент в наличии микросхема памяти EPROM 27C512 вместо требуемой микросхемы памяти EPROM 27C256. При этом устройство в котором используется имеющаяся в наличии микросхема не может корректно переключать страницы памяти (адресная линия A15) и чтение может произойти как из старшей страницы памяти так и из младшей.



Замечание: Данная функция всегда использует начальный адрес равный 00000h.

Буфер / Контрольная сумма (BUFFER / CHECKSUM)

Диалоговое окно '**Контрольная сумма (Checksum)**' позволяет подсчитать контрольную сумму выделенного блока данных в буфере управляющей программы.

Способы подсчета контрольной суммы:

BYTE –	суммирование байтов в "слово". Флаг CY игнорируется.
WORD –	суммирование слов в "слово". Флаг CY игнорируется.
BYTE (CY) –	суммирование байтов в "слово". Флаг CY суммируется с результатом.
WORD (CY) –	суммирование слов в "слово". Флаг CY суммируется с результатом.
CRC-CCITT –	суммирование байтов в "слово" по формуле $RESULT = PREVIOUS + (x16 + x12 + x5 + 1)$
CRC-XMODEM –	суммирование байтов в "слово" по формуле $RESULT = PREVIOUS + (x16 + x15 + x2 + 1)$

Позиция отмеченная как **NEG.** отображает результат преобразования контрольной суммы по формуле **SUM + NEG. = FFFFH.**

Позиция отмеченная как **SUPPL.** отображает результат преобразования контрольной суммы по формуле **SUM + SUPPL. = 0 (+ CY).**

Пункты диалогового окна:

'Начало блока (From address)': Начальный адрес блока данных в буфере управляющей программы, выбранного для подсчета контрольной суммы.

'Конец блока (To address)': Конечный адрес блока данных в буфере управляющей программы, выбранного для подсчета контрольной суммы.

'Тип результата (Insert checksum)': Специальный пункт, с помощью которого можно выбрать способ подсчета контрольной суммы и тип отображаемых данных.

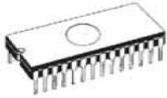
‘**Адрес вставки (Insert address)**’: Специальный пункт, с помощью которого определяется адрес буфера управляющей программы, по которому будет помещен результат подсчета контрольной суммы. Этот адрес не может находиться внутри диапазона определенных пунктами ‘**Начальный адрес (From address)**’ и ‘**Конечный адрес (To address)**’.

‘**Размер (Size)**’: С помощью этого пункта определяется разрядность результата подсчета контрольной суммы, помещаемого в буфер управляющей программы. Результат может иметь разрядность 8 (байт) или 16 (слово) бит. Если выбран 16–битный режим, в буфер будет помещен полный результат подсчета контрольной суммы. При выборе 8–битного режима в буфер будет записан только младший байт результата подсчета контрольной суммы.

‘**Считать (Calculate)**’: Клик на эту кнопку запускает процедуру подсчета контрольной суммы выбранного блока данных в буфере управляющей программы. После подсчета содержимое буфера не изменяется.

‘**Считать/Вставить (Calculate & insert)**’: Клик на эту кнопку запускает процедуру подсчета контрольной суммы выбранного блока данных в буфере управляющей программы. После подсчета контрольной суммы, результат будет помещен в буфер по адресу, определенному в пункте ‘**Адрес вставки (Insert address)**’.

***Замечание:** При выборе 16–битного режима, младший байт результата подсчета контрольной суммы будет помещен по адресу указанному в пункте ‘Адрес вставки’ (Insert address), а старший байт результата подсчета контрольной суммы будет помещен по адресу ‘Адрес вставки’ (Insert address) + 1.*



Микросхема (DEVICE)

Меню ‘Микросхема (Device)’ содержит команды для работы с микросхемами – выбор микросхем, чтение, проверка очистки, программирование и верификация.

Микросхема / Журнал ИС (DEVICE / SELECT FROM DEFAULT DEVICES)

Эта команда представляет доступ к циклическому буферу–истории, в котором содержится информация о последних десяти микросхемах, с которыми работал пользователь.

Для получения дополнительной информации о микросхеме используется сочетание зарезервированных клавиш <Ctrl+F1>. Информация отображает размер и организацию памяти микросхемы, алгоритм программирования, список программаторов, поддерживающих работу с этой микросхемой и некоторые специальные сведения. Типоразмер корпуса и расшифровка маркировки микросхемы также доступны для изучения.

Клавиша позволяет удалить из списка выбранную для просмотра микросхему. Полностью очистить список таким способом нельзя. Список истории микросхем всегда содержит информацию хотя-бы об одной микросхеме.

Микросхема / Выбор микросхем (DEVICE / SELECT DEVICE)

В этом диалоговом окне отображен список всех микросхем, поддерживаемых текущим программатором. Имеется возможность выбора нужной микросхемы ‘по имени (by name)’, ‘по типу (by type)’ или ‘по производителю (by manufacturer)’.

Выбранная микросхема автоматически сохраняется в списке ‘Микросхема / Журнал ИС (Device / Select from default devices)’. Этот список содержит последние десять микросхем с которыми работал пользователь.

Для получения более подробной информации о микросхеме зарезервировано сочетание клавиш <Ctrl+F1>. Информация отображает размер и организацию памяти микросхемы, алгоритм программирования, список программаторов, поддерживающих работу с этой микросхемой и некоторые специальные сведения. Типоразмер корпуса и расшифровка маркировки микросхемы также доступны для изучения.

Микросхема / Выбор EEPROM/FLASH по ID (DEVICE / SELECT EPROM/FLASH BY ID)

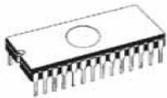
Эта команда используется для автоматического выбора типа микросхемы EPROM или FLASH путем чтения ее идентификатора ID. Программатор корректно определяет микросхемы только в том случае, если производитель разместил в памяти микросхемы идентификатор.

Автоматический выбор микросхемы предназначен только для микросхем памяти EPROM или FLASH, поддерживающих эту функцию. Если микросхема не поддерживает идентификацию по ID или производитель не занес данные идентификации в память, микросхема опознана не будет (неопределенное устройство).

Если для прочитанного идентификатора найдено несколько подходящих микросхем, будет показан весь список. Из этого списка можно самостоятельно выбрать наиболее подходящий тип микросхемы. Нажатие клавиши <ENTER> (или клик мышкой на кнопке 'OK') подтвердит выбор микросхемы. При нажатии клавиши <ESC> или клик мышкой на кнопке 'Отмена (Cancel)' прервет выбор микросхемы из списка. При этом никакая из микросхем выбрана для работы не будет.

***Замечание:** Программа управления поддерживает автоматическое определение микросхем EPROM и FLASH только с количеством выводов 28 и 32.*

Программатор использует повышенное напряжение на выводах микросхем. Это необходимо для получения идентификатора микросхемы. Не применяйте функцию автоматического выбора микросхемы по идентификатору к микросхемам, отличным от типов EPROM или FLASH. Это может повредить микросхему. Не рекомендуется применять эту функцию к микросхемам EPROM типов 2764 и 27128. Большинство этих микросхем не поддерживают идентификацию.



Микросхема / Дополнительные настройки (DEVICE / DEVICE OPTIONS)

Команды этого меню предназначены для более тонкой настройки процесса программирования микросхем, сериализации и других процессов.

Микросхема / Дополнительные настройки / Действия с ИС (DEVICE / DEVICE OPTIONS / OPERATION OPTIONS)

Настройки этого диалогового окна используются для установки специальных параметров программирования микросхем.

Перед изменением специальных настроек обязательно обратитесь к даташитам производителей микросхем, чтобы выяснить назначение всех интересующих параметров программирования.

Группа **‘Адреса (ADDRESS)’**:

Параметр **‘Начало в ИС (Device start address)’** по умолчанию равен 0.

Параметр **‘Конец в ИС (Device end address)’** по умолчанию равен объему памяти микросхемы – 1.

Параметр **‘Начало в буфере (Buffer start address)’** по умолчанию равен 0

Параметр **‘Деление (Split)’** по умолчанию нет. Этот параметр используется для установки специального режима буфера при программировании или чтении микросхемы. Использование этого параметра требуется в тех случаях, когда микросхемы с 8–битной шиной используются в составе систем с 16–битной или 32–битной шиной.

Приведенные таблицы определяют распределение данных при использовании различных режимов деления:

Деление	Микросхема	Адрес буфера
Нет	ИС [ADDR]	Буфер [ADDR]
Четное	ИС [ADDR]	Буфер [2*ADDR]
Нечетное	ИС [ADDR]	Буфер [1+ (2*ADDR)]
1./4	ИС [ADDR]	Буфер [4*ADDR]
2./4	ИС [ADDR]	Буфер [1+ (4*ADDR)]
3./4	ИС [ADDR]	Буфер [2+ (4*ADDR)]
4./4	ИС [ADDR]	Буфер [3+ (4*ADDR)]

Реальная адресация данных (все данные представлены в формате HEX):

Деление	Микросхема	Адрес буфера
Нет	Адрес ИС	00 01 02 03 04 05 06 07 08 09 ...
	Адрес буфера	00 01 02 03 04 05 06 07 08 09 ...
Четное	Адрес ИС	00 01 02 03 04 05 06 07 08 09 ...
	Адрес буфера	00 02 04 06 08 0A 0C 0E 10 12 ...
Нечетное	Адрес ИС	00 01 02 03 04 05 06 07 08 09 ...
	Адрес буфера	01 03 05 07 09 0B 0D 0F 11 13 ...
1./4	Адрес ИС	00 01 02 03 04 05 06 07 08 09 ...
	Адрес буфера	00 04 08 0C 10 14 18 1C 20 24 ...
2./4	Адрес ИС	00 01 02 03 04 05 06 07 08 09 ...
	Адрес буфера	01 05 09 0D 11 15 19 1D 21 25 ...
3./4	Адрес ИС	00 01 02 03 04 05 06 07 08 09 ...
	Адрес буфера	02 06 0A 0E 12 16 1A 1E 22 26 ...
4./4	Адрес ИС	00 01 02 03 04 05 06 07 08 09 ...
	Адрес буфера	03 07 0B 0F 13 17 1B 1F 23 27 ...



Сокращения, используемые в таблицах:

Адрес микросхемы представлен как ИС[ADDR].

Адрес буфера представлен как Буфер[ADDR].

Переменная ADDR может принимать значение от нуля до значения равного объему микросхемы (в байтах).

Все адреса – байт–ориентированные.

Группа ‘Тесты (INSERTION TEST)’:

Параметр ‘Тест установки (Insertion test)’ по умолчанию разрешено. Этот параметр позволяет программатору проверить качество контактов по всем выводам установленной микросхемы и соответствующим контактам гнезда ZIF. В ряде случаев программатор корректно отличает отсутствие контакта от неправильной установки микросхемы.

Параметр ‘Чтение ID из ИС (Device ID check)’ по умолчанию разрешено. Этот параметр позволяет прочитать из установленной в гнездо ZIF микросхемы идентификатор ID.

Замечания: Микросхемы ранних выпусков могут не содержать идентификатор ID. Если микросхема имеет установленный бит защиты от чтения кода программы, то идентификатор такой микросхемы прочитан не будет.

Группа ‘Действия (COMMAND EXECUTION)’:

Параметр ‘Проверить очистку перед записью (Blank check before programming)’ по умолчанию неактивен. Параметр позволяет проверить чистоту микросхемы перед началом программирования.

Параметр ‘Стереть перед записью (Erase before programming)’ по умолчанию неактивен. Параметр позволяет стереть (очистить) микросхему перед началом программирования.

Параметр ‘Проверить после чтения (Verify after reading)’ по умолчанию активен. Параметр позволяет повторно прочитать микросхему после операции чтения.

Параметр **‘Проверить (Verify)’** имеет два значения – **‘Один раз (Once)’**, **‘Два раза (Twice)’**. Параметр определяет количество циклов верификации микросхемы.

Параметр **‘Режим проверки (Verify options)’** имеет несколько значений ($VCC \pm 5\%$, $VCC \pm 10\%$, VCC_{min} , VCC_{max}). Параметр определяет при каких уровнях питающего напряжения будет проверяться запрограммированная микросхема.

Группа ‘Параметры ISP (ISP TARGET SUPPLY PARAMETERS)’:

Параметр **‘Запитать программируемую систему (Enable target system power supply)’** позволяет обеспечить электропитание программируемой системы за счет программатора. Напряжение питания подается перед процессом программирования и снимается после окончания программирования. Если установлен параметр **‘Контроль ISP после операции с ИС (Keep ISP signals at defined level after operation)’**, то программатор снимет напряжение питания после того как резисторы режима PU/PD будут деактивированы.

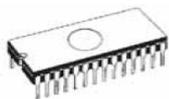
Параметр **‘Напряжение (Voltage)’** позволяет выбрать напряжение питания для программируемой системы. Поддерживается диапазон напряжений 2...6V.

Параметр **‘Максимальный ток (Max. current)’** позволяет определить максимальный потребляемый ток для программируемой системы. Поддерживается диапазон 0...300mA.

Параметр **‘Время нарастания напряжения (Voltage rise time)’** определяет время нарастания напряжения питания до рабочей величины для программируемой системы.

Параметр **‘Время удержания системы (Target supply settle time)’** определяет время в течение которого программируемая система стабилизирует свое состояние и будет готова к дальнейшим операциям с программируемой микросхемой.

Параметр **‘Время спада напряжения (Voltage fall time)’** определяет время спада напряжения питания после окончания работы с программируемой системой.



Параметр **‘Время отключения напряжения (Power down time)’** определяет время, в течение которого в программируемой системе может присутствовать остаточное напряжение питания. По истечении этого времени программатор может быть безопасно отключен от программируемой системы.

Группа ‘Параметры программируемой системы (TARGET SYSTEM PARAMETERS)’:

Параметр **‘Частота (Oscillator frequency)’** определяет частоту генератора микросхемы в программируемой системе. Параметр необходим для корректного определения скорости передачи данных в режиме программирования.

Параметр **‘Напряжение (Supply voltage)’** проверяет напряжение питания в программируемой системе. Перед каждой операцией с микросхемой управляющая программа проверяет или корректирует напряжение питания в программируемой системе.

Параметр **‘Не тестировать уровень напряжения (Disable test supply voltage)’** позволяет не проверять напряжение питания в программируемой системе. Значение напряжения питания перед операцией с микросхемой устанавливается в параметре **‘Напряжение (Supply voltage)’**.

Параметр **‘Задержка после сигнала (Reset Delay after reset active)’** определяет время ожидания до начала работы с программируемой системой после окончания действия сигнала сброса. Значение времени задержки зависит от конструктивных особенностей программируемой системы.

Параметр **‘Уровень на выводах разъема ISP (Inactive level of ISP signals)’** определяет состояние сигналов на выводах разъема ISP программатора после окончания работы с программируемой системой. Выводы разъема могут быть переведены в состояние PU или PD с помощью резисторов с номиналами 22 кОм.

Параметр **‘Удерживать уровни на выводах разъема ISP (Keep ISP signals at defined level after operation)’** позволяет сохранить установленные уровни на выводах разъема ISP после окончания работы с программируемой системой. При выходе из управляющей программы появляется предупреждение о том, что резисторы PU/PD активированы. После полного выхода из управляющей программы, резисторы деактивируются.

Микросхема / Дополнительные настройки / Сериализация (DEVICE / DEVICE OPTIONS / SERIALIZATION)

‘Сериализация (Serialization)’ это специальный режим управляющей программы. Когда режим активирован, специальные переменные автоматически помещаются в область буфера управляющей программы, предназначенную для хранения уникального серийного номера программируемой микросхемы. При программировании большого количества однотипных микросхем, управляющая программа самостоятельно изменяет значение серийного номера таким образом, что все запрограммированные микросхемы имеют уникальные серийные номера.

Доступны два способа сериализации:

- ‘Режим приращения (Incremental mode)’
- ‘Из файла (From file mode)’

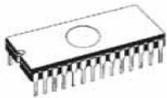
При выборе для работы новой микросхемы, режим сериализации по умолчанию деактивируется.

При выборе способа сериализации ‘Режим приращения’, в конфигурационный файл сохраняются следующие параметры: ‘Адрес (Address)’, ‘Объем (Size)’, ‘Размер S/N (Serial value)’, ‘Шаг приращения (Incremental step)’, ‘Тип (ASCII/BIN)’, ‘Вид (DEC/HEX)’ и параметр ‘Сохранить в буфер (Первый LS/MS байт (LS byte / MS Byte first))’.

При выборе способа сериализации ‘Из файла’ в конфигурационный файл сохраняются следующие параметры: имя файла, содержащего данные для режима сериализации, и метки указывающие на текущий серийный номер в файле.

При работе с мультипрограмматорами доступны дополнительные параметры в разделе ‘Незаписанные S/N из-за ошибок (Action on not programmed serial values due to error)’. В этом разделе расположены настройки:

- ‘Игнорировать незаписанные S/N (Ignore not programmed serial values)’.
- ‘Добавить незаписанные S/N в файл (Add not programmed serial values to file)’.



Первый параметр позволяет полностью игнорировать серийные номера не участвовавшие в процессе программирования и сериализации. Эти серийные номера не сохраняются.

Второй параметр позволяет все серийные номера которые не участвовали в процессе программирования, например из-за ошибок, сохранить в файл. Имя файла и папка могут быть указаны дополнительно.

Если процесс программирования остановлен пользователем или из-за ошибки программирования, управляющая программа не будет изменять серийный номер для следующей микросхемы. Добавление в файл незаписанных серийных номеров будет использовано только в том случае, если хотя-бы одна микросхема в мультипрограмматоре была успешно записана и проверена.

Режим сериализации работает только с буфером управляющей программы. Это означает, что режим сериализации использует только ту часть буфера, которая содержит области для программ и данных. Специальные области памяти микросхем, недоступные для буфера управляющей программы в процессе сериализации не участвуют.

Микросхема/ Дополнительные настройки / Сериализация/ Режим приращения (DEVICE / DEVICE OPTIONS / SERIALIZATION / INCREMENTAL MODE)

При использовании этого способа сериализации, уникальные серийные номера для микросхем будут вычислены в управляющей программе.

Для настройки процесса вычисления доступны следующие параметры:

Параметр '**Размер S/N (S/N size)**' определяет длину серийного номера в байтах.

Параметр '**Адрес (Address)**' определяет адрес буфера, по которому будет расположено значение серийного номера. Этот адрес должен находиться в границах памяти микросхемы.

Параметр '**Начать с (Start value)**' определяет начальное значение для вычисления серийного номера. Это значение должно быть в пределах 0...\$1FFFFFFF.

Параметр '**Шаг (Step)**' определяет шаг приращения следующего серийного номера.

Параметр '**Тип (S/N mode)**' определяет формат данных, помещаемых в буфер. Параметр может принимать два значения:

- '**ASCII**' – серийный номер помещается в буфер в виде строки '**ASCII**'. Например значение \$0528CD ('**ASCII**') будет помещено в буфер в виде последовательности (30h 35h 32h 38h 43h 44h ('0' '5' '2' '8' 'C' 'D')), то есть всего шесть байтов.
- '**BIN**' – серийный номер помещается в буфер в непосредственном виде, без изменений. Если серийный номер имеет длину более одного байта, имеется возможность выбора последовательности помещения байтов в буфер ('**Сохранить в буфер (Save to buffer)**').

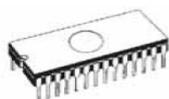
Параметр '**Вид (Style)**' определяет способ представления серийных номеров. Параметр может иметь два значения:

- '**DEC**' серийные номера отображаются в виде символов 0...9.
- '**HEX**' серийные номера отображаются в виде символов 0...9 и A...F.

В случае представления серийных номеров в виде '**DEC**' область для ввода '**Начать с (Start value)**' будет переключена в режим '**BCD**'. В этом режиме серийные номера вводятся в формате '**HEX**', но для ввода недоступны символы A...F.

Параметр '**Сохранить в буфер (Save to buffer)**' определяет последовательность расположения байтов серийного номера в буфере. Параметр имеет значение только для режима '**BIN**'. Доступны два способа для помещения данных:

- '**Первый LS (LSByte first)**' (используется для форматов Intel) помещает младший байт серийного номера в младший адрес буфера.
- '**Первый MS (MSByte first)**' (используется для форматов Motorola) помещает старший байт серийного номера в младший адрес буфера.



Параметр ‘**Делить S/N каждые xxx байт (Split serial number at every N byte(s))**’ позволяет делить серийный номер микросхемы на отдельные байты и помещать их в каждый N–ый адрес буфера. Этот параметр имеет практическое значение при специальном режиме (‘**Режим сериализации SQTP (SQTP serialization mode)**’) для микроконтроллеров Microchip PIC, когда серийный номер микросхемы может быть представлен частью памяти программ как группа инструкций RETLW. Пример использования этого способа сериализации находится ниже (пример 2).

Примеры:

1. Требуется запись серийных номеров для микросхем AT29C040 по адресу 7FFFАН, размер серийных номеров 4 байта, первый серийный номер начинается со значения 16000000Н, приращение серийных номеров с шагом 1, тип серийного номера BIN, младший байт располагается в младшем адресе буфера.

Сериализация: Режим приращения

Размер S/N: 4 байта

Тип: BIN

Вид: HEX

Сохранить в буфер: Первый LS

Адрес: 7FFFСН

Начать с: 16000000Н

Шаг: 1

Данные для записи будут распределены следующим образом:

Для 1–й микросхемы

Адрес	Данные
007FFF0	xx 00 00 00 16

Для 2–й микросхемы

Адрес	Данные
007FFF0	xx 01 00 00 16

Для 3–й микросхемы

Адрес	Данные
007FFF0	xx 02 00 00 16

Ячейки xx предназначены для хранения других данных.

Серийные номера будут записаны в микросхемы по адресам 7FFFСН.....7FFFFН, так как каждый серийный номер занимает 4 байта.

2. Пример использования специального режима (**‘Режим сериализации SQTP (SQTP serialization mode)’**), когда серийный номер разделен в группе инструкций RETLW для микроконтроллера Microchip PIC16F628.

Микросхема PIC16F628 использует 14–битные инструкции. Инструкция RETLW также имеет длину кода 14 бит:

Описание	СТЧ	14–битное слово	МЛЧ
RETLW Return with literal in W	11	01xx kkkk	kkkk

Ячейка xx может быть изменена на 00 или может содержать байт серийного номера. СТЧ – старшая часть слова. МЛЧ – младшая часть слова.

Код инструкции RETLW в десятично–шестнадцатеричном представлении имеет вид 34ККН, где КК может являться байтом серийного номера.

Предположим, требуется поместить серийный номер 1234ABCDH в область инструкций RETLW микросхемы. Старший байт серийного номера более значимый байт. Серийный номер будет записан в область памяти программ по адресу 40H. Использование специального режима в этой ситуации более предпочтительно. Сериализация без деления серийного номера приведет к следующему результату:

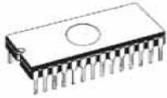
Адрес	Данные
0000080	CD AB 34 12 xx

Адрес имеет значение 80H потому что буфер имеет 8–битную организацию и микросхема также имеет 8–битную организацию, что эквивалентно адресу памяти программ 40H. Когда буфер имеет 16–битную организацию, адрес принимает значение 40H и серийный номер 1234ABCDH будет расположен в буфере следующим образом:

Адрес	Данные
0000040	ABCD 1234 xxxx xxxx xxxx xxxx xxxx xxxx

При использовании команды RETLW буфер примет вид:

Адрес	Данные
0000040	34CD 34AB 3434 3412 xxxx xxxx xxxx xxxx



Для получения такого эффекта сделайте следующие шаги:

1. Поместите четыре команды `RETLW` по адресу буфера `40H` (это можно сделать прямым редактированием буфера). Представление инструкции `RETLW` в виде 8-битовой последовательности сейчас не важно, так как процесс сериализации изменит данную область памяти и инструкция примет правильный вид.

Перед началом программирования микросхемы, буфер будет иметь вид:

Адрес	Данные
0000040	3400 3400 3400 3400 xxxx xxxx xxxx xxxx

каждые 8 бит инструкции `RETLW` имеют нулевое значение, но они могут быть абсолютно любыми.

2. Установите настройки сериализации:

Размер S/N: 4 байта
Адрес: `40H`
Начать с: `1234ABCDH`
Шаг: 1
Тип: `VIN`
Вид: `HEX`
Сохранить в буфер: Первый `LS`
Делить S/N каждые xxx байт: 2

Корректировка серийного номера проводится перед программированием микросхемы

После программирования микросхемы, буфер примет вид:

Адрес	Данные
0000040	34CD 34AB 3434 3412 xxxx xxxx xxxx xxxx



3. Этот пример использует настройки предыдущего, но параметр ‘Делить S/N каждые xxx байт’ будет принимать значение 3 и 4 соответственно.

При выборе параметра равным 3, содержимое буфера выглядит так:

8–битная организация буфера:

Адрес	Данные
0000080	CD xx xx AB xx xx 34 xx xx 12 xx xx xx xx xx xx

16–битная организация буфера:

Адрес	Данные
0000040	xxCD ABxx xxxx xx34 12xx xxxx xxxx xxxx

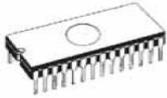
При выборе параметра равным 4, содержимое буфера выглядит так:

8–битная организация буфера:

Адрес	Данные
0000080	CD xx xx xx AB xx xx xx 34 xx xx xx 12

16–битная организация буфера:

Адрес	Данные
0000040	xxCD xxxx xxAB xxxx xx34 xxxx xx12 xxxx



Совет:

Если неизвестно какие результаты будут записаны в микросхему при сериализации, имеется возможность провести тестовую процедуру сериализации:

- Установите требуемые параметры сериализации в диалоговом окне ‘**Микросхема / Дополнительные настройки / Сериализация (Device / Device options/ Serialization)**’.
- В диалоговом окне ‘**Микросхема / Дополнительные настройки / Действия с ИС (Device / Device options/ Device operation options)**’ в свойствах пункта ‘**Чтение ID из ИС (Device ID check)**’ установите значение ‘Disabled’.
- Убедитесь что в гнезде программатора ZIF нет никаких микросхем.
- Начните процесс программирования (нажмите <F9>).
- После окончания программирования (обязательно появится сообщение об ошибке – нет микросхемы в гнезде программатора ZIF) нажмите клавишу <F4> и проверьте содержимое буфера по адресу, с которого должны начинаться данные с серийными номерами.

***Замечание:** Адреса для сериализации всегда должны соответствовать текущей организации памяти микросхемы. Если используется микросхема с 8-битной организацией, адреса должны быть также 8-битными. А если используется микросхема с 16-битной организацией, адреса для сериализации соответственно должны быть 16-битными.*

Микросхема/ Дополнительные настройки / Сериализация/ Из файла (DEVICE / DEVICE OPTIONS / SERIALIZATION / FROM FILE MODE)

При использовании этого режима сериализации серийные номера микросхем будут прочитаны из файла специального формата и помещены в буфер управляющей программы по определенным в файле адресам.

В этом режиме доступны следующие настройки:

Параметр ‘**Имя файла (File name)**’ определяет имя файла специального, содержащего данные для сериализации.

Параметр ‘**Метка строки (Start label)**’ определяет начальную строку в файле. Чтение данных для сериализации из файла будет начато со строки указанной в этом параметре.

Формат файла сериализации:

Файл сериализации включает в себя последовательности байтов и адреса, определяющие области буфера управляющей программы для размещения серийных номеров. Данные в файле представлены в текстовом формате. Структура файла выглядит следующим образом:

```
[метка1] адрес байт0 байт1.....байтn
.....
[меткан] адрес байт0 байт1.....байтm,   адрес байт0 байт1.....байтk
      \_____/          \_____/
      |                  |
      базовая часть     дополнительная часть

; Комментарии
```

Пояснения:

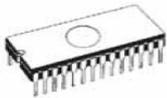
Базовая часть определяет адрес буфера и последовательность байтов, которые будут помещены в буфер. Базовая часть должна быть всегда размещена после метки строки.

Дополнительная часть определяет вторичные последовательности байтов и адрес буфера, в который будут помещены эти последовательности. Одна дополнительная часть должна быть всегда размещена после базовой части. Дополнительная часть должна быть отделена от базовой части с помощью символа–разделителя ‘,’ (запятая).

метка1, меткан – метки. Определяют каждую строку в файле сериализации и используются для адресации к строкам. Метки должны иметь уникальные названия.

адрес – определяет адрес буфера по которому будут записаны данные, расположенные за этим адресом.

байт0.....байтn, байт0.....байтm, байт0.....байтk – последовательность байтов, определяющая данные, предназначенные для размещения в буфер управляющей программы. Максимальная длина последовательности для одной строки не может превышать 64 байт. Данные последовательности помещаются в буфер последовательно, по мере их получения, начиная с начального адреса.



Пример размещения байтов данных:

байт0 в адрес
байт1 в адрес + 1
байт2 в адрес + 2
.....
байт n в адрес + n

Специальные символы, используемые в файле сериализации:

[] – квадратные скобки. Имена меток всегда помещаются в квадратные скобки.

, – запятая. Используется для разделения базовой и дополнительных частей.

; – точка с запятой. В файле сериализации допускается использование комментариев. Текст, помещенный за этим символом будет воспринят как комментарий и обрабатываться не будет.

Замечания: При определении имен меток допускается использование больших и маленьких символов. Маленькие символы всегда конвертируются в большие управляющей программой автоматически.

Все адреса и байты должны быть представлены в шестнадцатеричном виде.

Адрес может иметь длину от 1 до 4 байтов.

Длина данных в одной последовательности не может быть более 64 байтов. Если в одной строке используется базовая и дополнительная части, общая длина строки не должна превышать 80 байтов.

Всегда требуется проверка определяемых адресов. Если адрес последовательности не попадает в область физической памяти микросхемы, он считается адресом за границами диапазона – работа с таким адресом не проводится и процедура сериализации прекращается.

Адреса для сериализации всегда должны соответствовать текущей организации памяти микросхемы. Если используется микросхема с 8-битной организацией, адреса должны быть также 8-битными. А если используется микросхема с 16-битной организацией, адреса для сериализации соответственно должны быть 16-битными.

Пример:

[nav1] A7890 78 89 56 02 AB CD ; комментарий 1
[nav2] A7890 02 02 04 06 08 0A
[nav3] A7890 08 09 0A 0B A0 C0 ; комментарий2
[nav4] A7890 68 87 50 02 0B 8D
[nav5] A7890 A8 88 59 02 AB 7D

; следующая строка содержит дополнительную часть

[nav6] A7890 18 29 36 42 5B 6D , FFFF6 44 11 22 33 99 88 77 66 55 16

; это последняя строка файла

Микросхема / Дополнительные настройки / Параметры статистики (DEVICE / DEVICE OPTIONS / STATISTICS)

Это меню отображает суммарные результаты работы по конкретным типам микросхем. Если микросхема одного типа соответствует одной операции с ней, то количество типов микросхем будет соответствовать количеству всех операций.

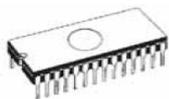
Параметр ‘Счетчик (Count down)’ позволяет подсчитывать количество микросхем в обратном порядке. Эта функция особенно полезна при массовой работе с микросхемами одного типа. Когда операция с микросхемой будет успешно выполнена, значение счетчика уменьшится на единицу. Когда значение счетчика станет равно нулю, это будет означать, что выполнена работа с заранее определенным количеством однотипных микросхем.

Диалоговое окно позволяет настроить следующие параметры:

Параметры ‘Запрограммировано (Program)’, ‘Проверено (Verify)’, ‘Проверка очистки (Blank)’, ‘Стерто (Erase)’ и ‘Считано (Read)’ позволяют отметить необходимые действия с микросхемой, подлежащие подсчету, после которых значение статистики будет увеличено.

Параметр ‘Счетчик (Count down)’ позволяет активировать функцию обратного отсчета и установить количество подсчитываемых микросхем.

Доступ к этому диалоговому окну можно получить также по правому клику мышкой на области счетчика главного окна управляющей программы.



Текущие показания счетчика отображаются в панели **‘Статистика (Statistics)’** главного окна управляющей программы. Панель также содержит информацию о результатах работы – **‘Успешно (Success)’**, **‘Отказов (Failure)’**, **‘Всего (Total)’**, **‘Счетчик (Count down)’** и **‘Осталось (Remains)’**.

Назначение показаний:

- **‘Успешно (Success)’** – количество успешно завершенных операций.
- **‘Отказов (Failure)’** – количество операций, завершенных отказами.
- **‘Всего (Total)’** – общее количество операций.
- **‘Счетчик (Count down)’** – текущее состояние счетчика.
- **‘Осталось (Remains)’** – количество микросхем, оставшихся до остановки счетчика и окончания процесса работы.

Успешной операцией считается работа с микросхемой, когда не было ошибок ни по одному из пунктов, отмеченных для учета статистики с микросхемой:

- **‘Запрограммировано (Program)’**
- **‘Проверено (Verify)’**
- **‘Проверка очистки (Blank)’**
- **‘Стерто (Erase)’**
- **‘Считано (Read)’**

Если возникла ошибка в процессе выполнения одного из пунктов, такая операция успешной не считается.

При установке нового типа микросхемы, настройки статистики обнуляются и счетчик деактивируется.

Кнопка **‘Сброс (Reset)’** на панели статистики сбрасывает в ноль текущие значения статистики.

Кнопка **‘Обновить счетчик (Reload Count down)’** на панели статистики позволяет обновить текущие показания счетчика.

Микросхема / Дополнительные настройки / Ассоциация (DEVICE / DEVICE OPTIONS / ASSOCIATED FILE)

Эта команда используется для ассоциации файла данных определенного формата с микросхемой определенного типа. Такая возможность позволяет автоматически открывать файл данных определенного типа и загружать его в буфер управляющей программы при выборе определенного типа микросхемы из списка или при запуске управляющей программы.

Самостоятельно можно выбрать файл и путь к этому файлу для ассоциации. Управляющая программа проверяет наличие этого файла на диске. Также имеется возможность выбора автоматической загрузки этого файла или загрузки в ручном режиме.

Микросхема / Дополнительные настройки / Специальный режим (DEVICE / DEVICE OPTIONS / SPECIAL OPTIONS)

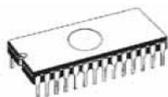
Эта команда предоставляет доступ к специальным настройкам режима программирования выбранной микросхемы. Для получения более подробных сведений о работе с пунктами этой команды обратитесь к даташитами производителя на микросхему.

Если название этого меню начинается с команды ‘**View/Edit...**’, это значит, что конфигурация микросхемы может быть изменена пользователем. Если этот пункт неактивен, микросхема не имеет специальных режимов настройки.

Микросхема / Чистая? (DEVICE / BLANK CHECK)

Эта команда позволяет проверить чистоту всех доступных областей памяти или только ее части для выбранной микросхемы. Результаты проверки отображаются в информационном окне.

В меню ‘**Микросхема / Дополнительные настройки / Действия с ИС (Device / Device options / Operation options)**’ можно выбрать начальный адрес области памяти для проверки чистоты (параметр ‘**Начало в буфере (Buffer start)**’).



Микросхема / Читать (DEVICE / READ)

Эта команда меню позволяет прочитать содержимое памяти всей микросхемы или ее части в буфер управляющей программы. В отдельных случаях может быть прочитана конфигурация микросхемы. Специальные настройки конфигурации микросхемы можно просмотреть и изменить в пунктах меню **‘Работа с буфером (View / Edit buffer)’** и **‘Микросхема / Дополнительные настройки / Специальный режим (Device / Device options / Special options) <Alt+S>’**.

Процесс чтения микросхемы отображается в информационном окне.

Команда меню **‘Микросхема / Дополнительные настройки / Действия с ИС (Device / Device options / Operation options)’** позволяет установить другую рабочую область памяти микросхемы по умолчанию. Установка режима **‘Проверить после чтения (Verify after reading)’** позволяет читать микросхемы с более высокой степенью надежности.

Микросхема / Сверить (DEVICE / VERIFY)

Эта команда позволяет сравнить данные в памяти микросхемы или ее части с данными в буфере управляющей программы. Процесс верификации отображается в информационном окне.

Команда меню **‘Микросхема / Дополнительные настройки / Действия с ИС (Device / Device options / Operation options)’** позволяет установить другую рабочую область памяти микросхемы по умолчанию.

Пункт меню **‘Настройки / Основные настройки / Ошибки (Options / General options/ Display errors)’** позволяет выводить сообщения об ошибках на экран или в файл **‘Verify.err’**. При выводе сообщений об ошибках на экран, будут показаны только первые 45 различий.

Микросхема / Прошить (DEVICE / PROGRAM)

Эта команда позволяет записать в микросхему содержимое буфера управляющей программы или его части. Процесс программирования отображается в информационном окне. При неудачном программировании микросхемы, выводится сообщение об ошибке.

В меню ‘**Микросхема / Дополнительные настройки / Действия с ИС (Device / Device options / Operation options)**’ находятся команды, позволяющие более тонко настроить параметры программирования для данной микросхемы.

Микросхема / Стереть (DEVICE / ERASE)

Эта команда позволяет очистить (стереть) программируемую область памяти микросхемы. По окончании процесса очистки (стирания) управляющая программа предоставляет отчет о результатах, содержащий информацию об ошибках, предупреждениях или об их отсутствии.

Микросхема / Тест (DEVICE / TEST)

Эта команда выполняет тестирование микросхем типа ‘Static RAM’ на программаторах, поддерживающих этот режим тестирования.

Микросхема / Тест ИС (DEVICE / IC TEST)

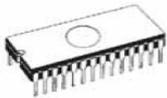
Эта команда позволяет проводить комплексное тестирование микросхем. Алгоритмы тестирования находятся в специальных библиотеках и разделены по типам микросхем.

Для тестирования необходимо выбрать соответствующую библиотеку, тип микросхемы и способ тестирования (циклический (**LOOP**) или пошаговый (**SINGLE STEP**)). Имеется возможность определения алгоритма тестирования самостоятельно. Подробная информация по составлению алгоритма тестирования находится в файле ‘example_e.lib’ в папке установки управляющей программы.

Замечание: При тестировании микросхемы имеется риск ее повреждения из-за неправильного определения алгоритма или из-за неправильно выбранного типа микросхемы.

Введение в JAM (INTRODUCTION TO JAM)

‘**Jam STAPL**’ был создан инженерами компании Altera® и используется группой компаний–производителей микросхем ‘**PLD**’.



‘**Standard Test and Programming Language (STAPL)**’, ‘**JESD-71**’ это стандарты форматов файлов для режима ISP. ‘**Jam STAPL**’ является свободно распространяемым стандартом (не требуется приобретение лицензий), позволяющим программировать и конфигурировать микросхемы с использованием стандарта IEEE1149.1 для интерфейса ‘**JTAG**’. Однако микросхема не может быть прочитана или верифицирована, потому что стандарт ‘**Jam STAPL**’ не поддерживает эти функции.

Программный комплекс ‘**Jam STAPL**’ состоит из двух частей: ‘**Jam Composer**’ и ‘**Jam Player**’.

Инструмент ‘**Jam Composer**’ является программой, которая генерирует файлы ‘**JAM**’, содержащие пользовательские данные и алгоритмы программирования данных в микросхему для конкретного проекта.

Инструмент ‘**Jam Player**’ представляет собой программу, предназначенную для открытия файлов ‘**JAM**’ и получения векторов для программирования и тестирования микросхем с интерфейсом ‘**JTAG**’.

Микросхемы могут быть запрограммированы с помощью установки в гнездо ZIF программатора или с помощью разъема ISP. Такие микросхемы содержат в своем названии строку ‘**[PLCC44](JAM)**’, ‘**(ISP-JAM)**’ или ‘**JTAG chain (ISP-JAM)**’.

Дополнительная информация доступна в сети Интернет:

http://www.altera.com/support/devices/programming/jam/dev-isp_jam.html

Работа с ISP:

<http://www.altera.com/literature/an/an100.pdf>

Использование Jam STAPL для ISP и ICR Embedded Processor:

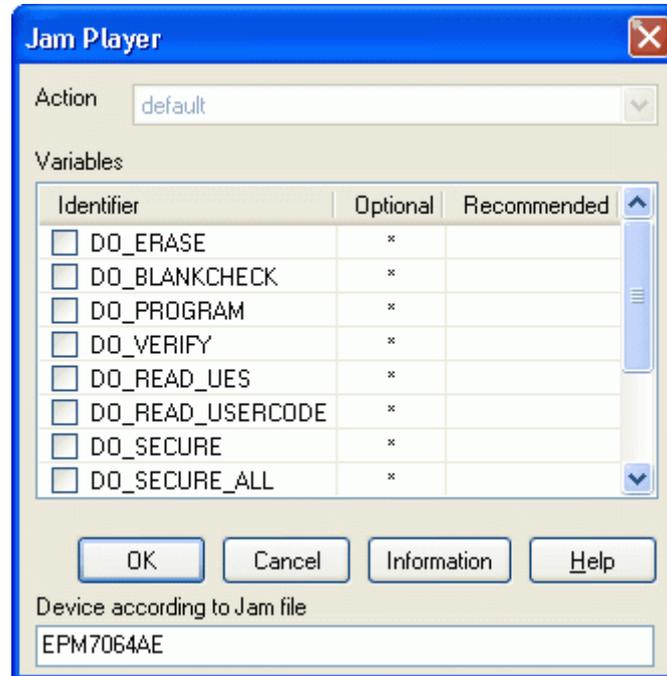
<http://www.altera.com/literature/an/an122.pdf>

Программное обеспечение:

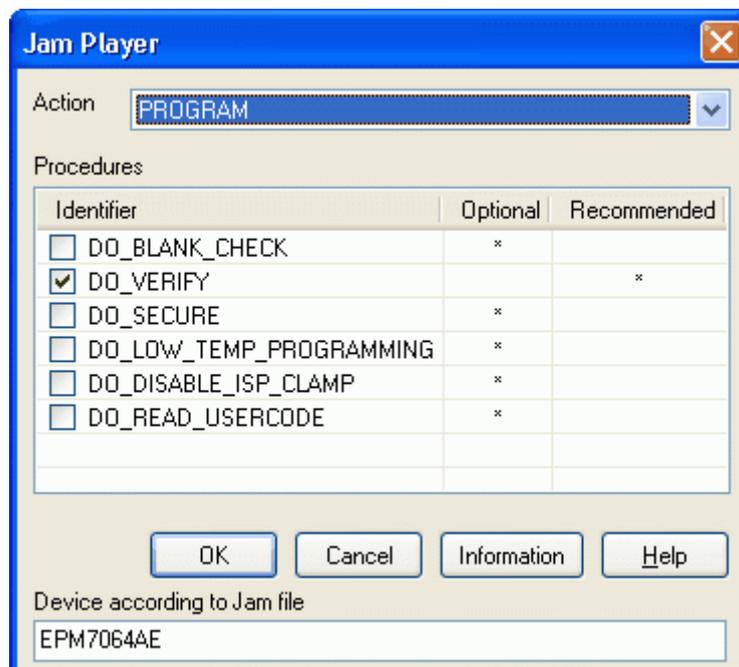
Altera: ‘**MAX+plus II**’, ‘**Quartus II**’, ‘**SVF2Jam**’, ‘**LAT2Jam**’.

Xilinx: ‘**Xilinx ISE Webpack**’.

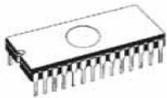
Диалоговое окно 'JAM player' (INFORMATION ABOUT THE JAM PLAYER DIALOG)



Jam Player версия 1 (настройки Action и Variable)



Jam Player версия 2 (настройки Action и Procedures)



Параметр **‘Действие (Action)’** позволяет выбрать необходимое действие. Файл **‘JAM’** версии 2 состоит из действий. Действия состоят из вызовов процедур. Файл **‘JAM’** версии 1 не содержит параметров **‘Действие (Action)’** и **‘Процедуры (Procedures)’** поэтому выбор действий недоступен. Процесс выполнения программирования управляет процедурами с приставкой **‘DO_’**. Если требуются специальные процедуры с приставкой **‘DO_’** необходимо обратиться к разработчиками программатора.

Параметр **‘Процедуры (Procedures)’**. Процесс выполнения программирования выполняет последовательности утверждений из каждой процедуры. Процедуры могут быть дополнительными и рекомендованными. Рекомендованные процедуры отмечены неявно. Самостоятельно можно разрешить или запретить некоторые процедуры в зависимости от конкретных требований. Jam Player выполняет только отмеченные процедуры. Остальные процедуры игнорируются. Число процедур различно и зависит от файла **‘JAM’**.

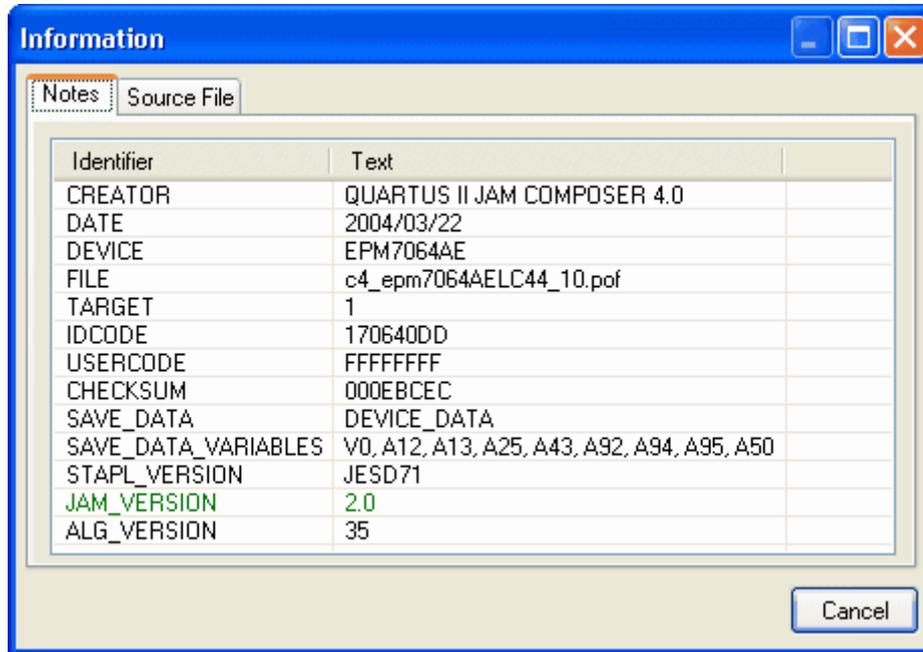
Параметр **‘Значения (Variables)’**. Файл **‘JAM’** версии 1 не содержит параметров **‘Действие (Action)’** и **‘Процедуры (Procedures)’** поэтому выбор действий недоступен. Процесс выполнения программирования управляет процедурами с приставкой **‘DO_’**. Jam Player выполняет все отмеченные процедуры с приставкой **‘DO_’**. Число процедур постоянно и не зависит от файла **‘JAM’**. Если требуются специальные процедуры с приставкой **‘DO_’** необходимо обратиться к разработчиками программатора.

Кнопка **‘OK’** – применить все действия с выбранными процедурами

Параметр **‘Информация (Information)’** – отображает информацию о файле **‘JAM’**.

Параметр **‘ИС соответствует файлу JAM (Device according to Jam file)’**. Файл создается для конкретного типа микросхем. Имя микросхемы расположено в файле **‘JAM’** – область **‘NOTE’**, описатель **‘DEVICE’**. Название микросхемы должно быть идентично описателю микросхемы. Если будет обнаружено несоответствие, управляющая программа укажет на это.

Диалоговое окно 'A JAM file information' (INFORMATION ABOUT A JAM FILE INFORMATION DIALOG)



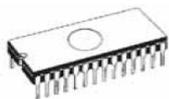
Вкладка 'Notes' используется для отображения информации о файле 'JAM'. Расположенная на этой вкладке информация может содержать любой тип данных, специфичный для файлов 'JAM'.

Вкладка 'Source file' содержит программу на языке 'JAM'. Программа 'JAM' состоит из последовательности утверждений. Каждое утверждение содержит метку, инструкцию и аргументы. Утверждение заканчивается символом ; (точка с запятой). Аргументами утверждения могут быть константы, переменные или выражения (Булевы или целочисленные). Каждое утверждение обычно занимает одну строку программы, но это необязательное требование – утверждение может быть разбито на несколько строк. Символ ' (апостроф) используется для выделения текста комментария. Язык 'JAM' не имеет ограничений на длину строки, длину утверждения или размер программы.

Дополнительная информация доступна в сети Интернет:

http://www.altera.com/support/devices/programming/jam/dev-isp_jam.html.

Файлы 'JAM' с расширением '.jbc' – это файлы формата 'Jam STAPL Byte code'. Такие файлы не отображаются.



Конвертирование файлов 'JED' в файлы 'JAM STAPL' для микросхем XILINX (INFORMATION ABOUT CONVERTING JED FILE TO JAM STAPL FILE FOR XILINX DEVICES)

Ниже представлена последовательность действий для конвертирования файла 'JED' в файл 'JAM STAPL':

1. Инсталлировать на компьютер среду 'Xilinx Integrated Software Environment (ISE) 6.3i'. Программное обеспечение доступно для свободного получения в сети Интернет. Требуются файлы: 'WebPACK_63_fcfull_i.exe', '6_3_02i_pc.exe'. Объем около 315Мб.
2. В установленной среде выбрать 'Xilinx ISE 6/Accessories/iMPACT':
 - в диалоговом окне 'Operation Mod Selection: What do you want to do first?' выбрать: 'Prepare Configuration Files'.
 - в диалоговом окне 'Prepare Configuration Files: I want create a:' выбрать: 'Boundary-Scan File'.
 - в диалоговом окне 'Prepare Boundary-Scan File: I want create a:' выбрать: 'STAPL File'.
 - в диалоговом окне 'Create a New STAPL File' записать имя файла 'JAM' с расширением '.stapl'.
 - в диалоговом окне 'Add Device' выбрать файл 'JED' с расширением '.jed'.
 - выбрать микросхему (например) XC2C32A и установить последовательность действий (например стереть, прошить, проверить).
 - в меню выбрать пункт 'Output/Stapl file/Stop writing to Stapl file'.
3. Запустить управляющую программу PG4UW, выбрать микросхему (например): Xilinx XC2x32A [QFG32](Jam), открыть файл 'JAM'.
4. Нажать клавиши <Alt+O> и кликнуть кнопку 'Jam configuration'. На предупреждение 'Выбранная микросхема и файл возможно различны! Продолжить? (Selected device from menu 'Select Devices' and Jam file is probably different! Continue?)' ответить 'Да (Yes)'. В диалоговом окне 'Jam player' выбрать необходимые действия с микросхемой и закрыть это диалоговое окно. Нажать кнопку 'Play Jam' на основной панели инструментов главного окна управляющей программы и после окончания работы просмотреть сообщения в окне сообщений управляющей программы.

Введение в VME (INTRODUCTION TO VME)

‘**ispVM Virtual Machine**’ – это виртуальная машина, которая была оптимизирована специальным образом для возможности программирования микросхем поддерживающих стандарт IEEE1149.1 для ‘**Boundary Scan**’. Программное обеспечение ‘**ispVM EMBEDDED**’ поддерживает язык ‘**Lattice ispVM Virtual Machine™**’ стандарта ‘**Serial Vector Format (SVF)**’ для ‘**Boundary Scan**’.

Программное обеспечение ‘**ispVM System**’ создает файлы стандарта ‘**VME**’ для форматов ‘**ispJTAG**’ и ‘**non-Lattice JTAG**’, совместимых со стандартом IEEE1149.1, а также файлы стандарта ‘**SVF**’ или файлы стандарта IEEE1532. Файлы ‘**VME**’ – это файлы шестнадцатеричного кодирования, содержащие информацию из ‘**ispVM System**’.

Микросхемы могут быть запрограммированы с помощью установки в гнездо ZIF программатора или с помощью разъема ISP. Такие микросхемы содержат в своем названии строку ‘**[PLCC44](VME)**’, ‘**(ISP-VME)**’ или ‘**JTAG chain (ISP-VME)**’.

Дополнительная информация доступна в сети Интернет:

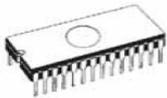
<http://www.latticesemi.com/products/devtools/software/ispvmembed/index.cfm>

Работа с ISP:

http://www.latticesemi.com/products/technology/isp_usage.cfm

Программное обеспечение:

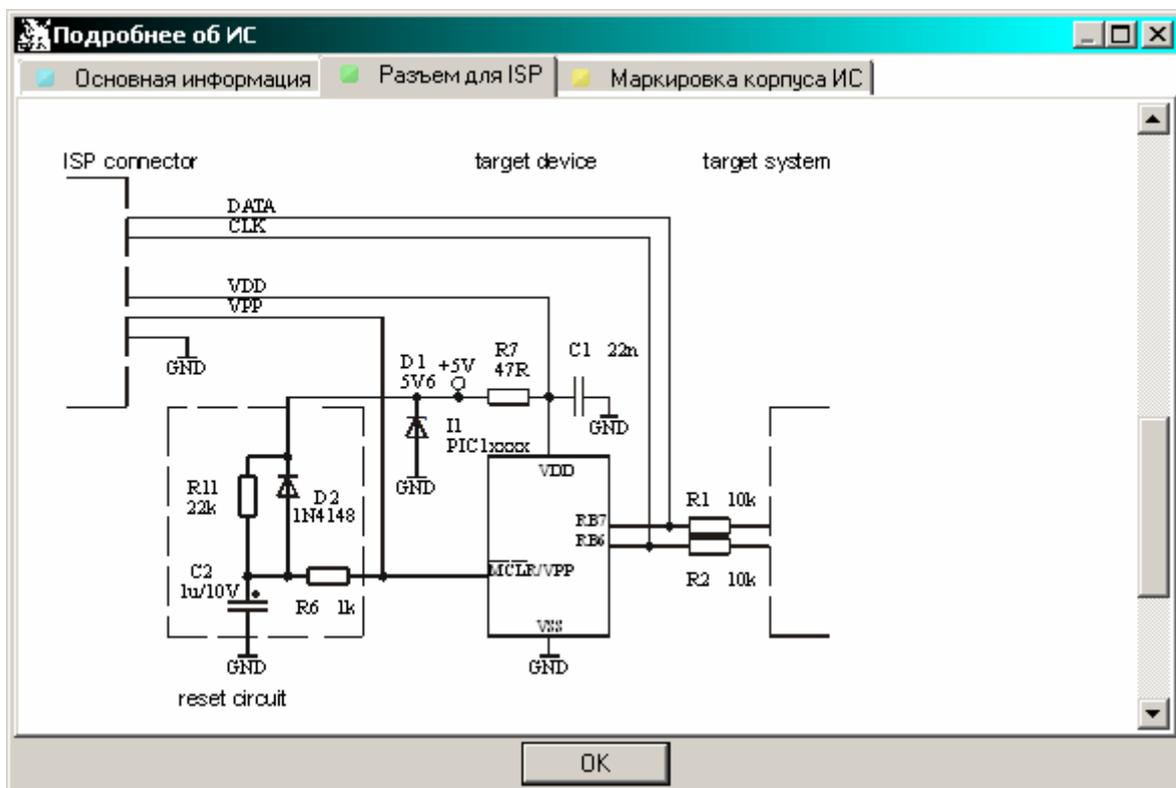
Lattice: ‘ispLEVER’, ‘ispVM System’, ‘PAC-Designer’, утилита ‘**svf2vme**’ для конвертирования в файлы ‘**VME**’.



Микросхема / Информация об ИС (DEVICE / DEVICE INFO)

Эта команда предоставляет расширенную информацию о текущей микросхеме – объем и организацию памяти микросхемы, алгоритм программирования, список программаторов, поддерживающих работу с микросхемой. Также доступна некоторая дополнительная информация по работе с микросхемой, размеры и тип корпуса, расшифровка обозначений на корпусе микросхемы.

Для доступа к этой функции из любого места управляющей программы зарезервировано сочетание клавиш <Ctrl+F1>.



Программатор (PROGRAMMER)

Меню **‘Программатор (Programmer)’** содержит команды для работы с текущим программатором.

Программатор / Поиск программатора (PROGRAMMER / FIND PROGRAMMER)

Эта команда позволяет выбрать новый тип программатора и переопределить параметры связи программатора и компьютера. Диалоговое окно содержит следующие параметры:

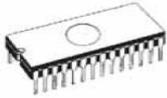
‘Программатор (Programmer)’ – параметр позволяет выбрать новый тип программатора для поиска. Если выбрано значение **‘Искать все (Search all)’**, то управляющая программа произведет поиск всех моделей программаторов по всем портам.

‘Скорость интерфейса (Establish communication)’ – параметр устанавливающий режим определения скорости обмена данными для нового типа программатора. Доступны значения **‘Вручную (Manual)’** или **‘Авто (Automatic)’**.

‘Скорость (Speed)’ – параметр, устанавливающий скорость обмена данными программатора и компьютера. Для получения максимальной производительности рекомендуется выбирать наибольшие значения этого параметра. Если при работе с программатором возникают ошибки, можно установить более меньшие значения.

‘Порт (Port)’ – параметр, определяющий к какому порту компьютера подключен программатор. Если выбрано значение **‘Все порты (All port)’**, то управляющая программа произведет поиск всех моделей программаторов по всем портам.

‘Адрес порта ‘Special’ (Address for special port)’ – параметр, определяющий адрес порта LPT, когда выбрано значение **‘Special’**.



Программатор / Повторный поиск программатора (PROGRAMMER / REFINDE PROGRAMMER)

Команды этого меню позволяют переустановить параметры скорости обмена данными текущего программатора и компьютера.

Для выбора другого типа программатора, установки параметров скорости обмена данными нового типа программатора и компьютера необходимо использовать меню: '**Программатор / Поиск программатора (Options / Find programmer)**'.

Программатор / Автомат (PROGRAMMER / HANDLER)

Эта команда предназначена для настройки параметров автоматического модуля '**Handler**'. '**Handler**' – это специальный внешний модуль, который производится сторонними компаниями и предназначен для использования в массовом производстве для автоматизированного контроля операций программирования.

В диалоговом окне доступны для настройки следующие параметры:

- '**Выбор автомата (Selected Handler)**' – параметр позволяет выбрать тип модуля '**Handler**'.
- '**Выбор порта (Search at port)**' – параметр позволяет выбрать порт COM, к которому подключен автомат '**Handler**'.

Нажатие клавиши **<Enter>** или клик мышкой на кнопке 'ОК' запускает процедуру поиска автомата '**Handler**'.

Замечание: Модуль в свободной продаже отсутствует.

Программатор / Настройка модулей (PROGRAMMER / MODULE OPTIONS)

Эта команда предназначена для настройки мультисокетных программаторов.

Настройки позволяют выбрать '**Гнездо MASTER (MASTER socket)**' (основное гнездо) программатора и определить дополнительные гнезда. Основное гнездо используется предпочтительно для операций чтения микросхем.

Параметры **‘Enable socket’** и **‘Disable socket’** позволяют настроить персональный доступ к каждому гнезду. Деактивированные гнезда (**Disable socket**) не участвуют ни в каких операциях с микросхемами.

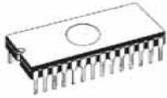
Программатор / Функция YES! (PROGRAMMER / AUTOMATIC YES!)

Это меню предназначено для настройки параметров программатора при работе функции YES! В таком режиме работы программатора после установки микросхемы в гнездо ZIF программатора с ней будет повторена последняя операция в автоматическом режиме без дополнительных манипуляций с управляющей программой. Установка микросхемы в гнездо ZIF отображается в главном окне управляющей программы. Для прекращения работы с этим режимом, необходимо нажать клавишу **<ESC>**, когда в гнездо ZIF еще не установлена новая микросхема.

Результаты текущей операции с микросхемой будут отображены светодиодами статуса **‘OK’** или **‘ERROR’**, расположенными на панели программатора. Активное состояние светодиода **‘BUSY’** указывает на то, что в текущий момент времени с микросхемой проводятся какие-то действия. После того, как микросхема удалена из гнезда ZIF, светодиоды статуса перейдут в неактивное состояние, а светодиод **‘BUSY’** будет мигать, указывая на ожидание программатора новой микросхемы. После установки новой микросхемы, светодиод **‘BUSY’** перейдет в активное состояние. Программа ожидает установку новой микросхемы в течение некоторого, заданного промежутка времени, по истечении которого (если микросхема не установлена или установлена неправильно) светодиод **‘ERROR’** перейдет в активное состояние. Когда новая микросхема будет установлена корректно светодиод **‘ERROR’** перейдет в неактивное состояние.

Если был произведен поиск нового программатора в пункте **‘Программатор / Поиск программатора (Options / Find programmer)’**, то текущее активное состояние функции YES! будет отменено.

Параметр **‘Время ответа (Response time)’** определяет интервал времени, в течение которого проверяется гнездо ZIF на предмет установки новой микросхемы. По умолчанию используется значение **‘Стандартно (Standard)’**. Если используются адаптеры, рекомендуется установить значение **‘Увеличено (Elongated)’**.



Параметр **‘Выводы с конденсаторами (Pins with capacitors)’** позволяет отметить выводы микросхемы или адаптера, к которым подключены конденсаторы (например блокировочный конденсатор адаптера на выводах VCC и GND), так как это может вызвать проблемы с определением микросхем.

Пример: 4,6,17

Параметр **‘Ожидание новой ИС (Device removal hold off time)’** определяет промежуток времени между извлечением предыдущей микросхемы и установкой новой в гнездо ZIF.

Параметр **‘Ожидание установки (Device insertion complete time)’** определяет время в течение которого производится закрепление микросхемы в гнезде ZIF, после того как программатор обнаружил начало установки новой микросхемы. По истечении этого времени будет проведена проверка на правильность установки.

Все параметры этого диалогового окна отменяются при выборе новой микросхемы в меню **‘Микросхема / Журнал ИС (Device / Select from default devices)’** или **‘Микросхема / Выбор микросхем (Device / Select device)’**.

Программатор / Самотестирование (PROGRAMMER / SELFTEST)

Эта команда выполняет процесс самотестирования программатора без использования диагностического модуля POD. Рекомендуется также проводить расширенную процедуру самотестирования: **‘Программатор / Самотестирование + (Programmer / Selftest plus)’**.

Программатор / Самотестирование + (PROGRAMMER / SELFTEST PLUS)

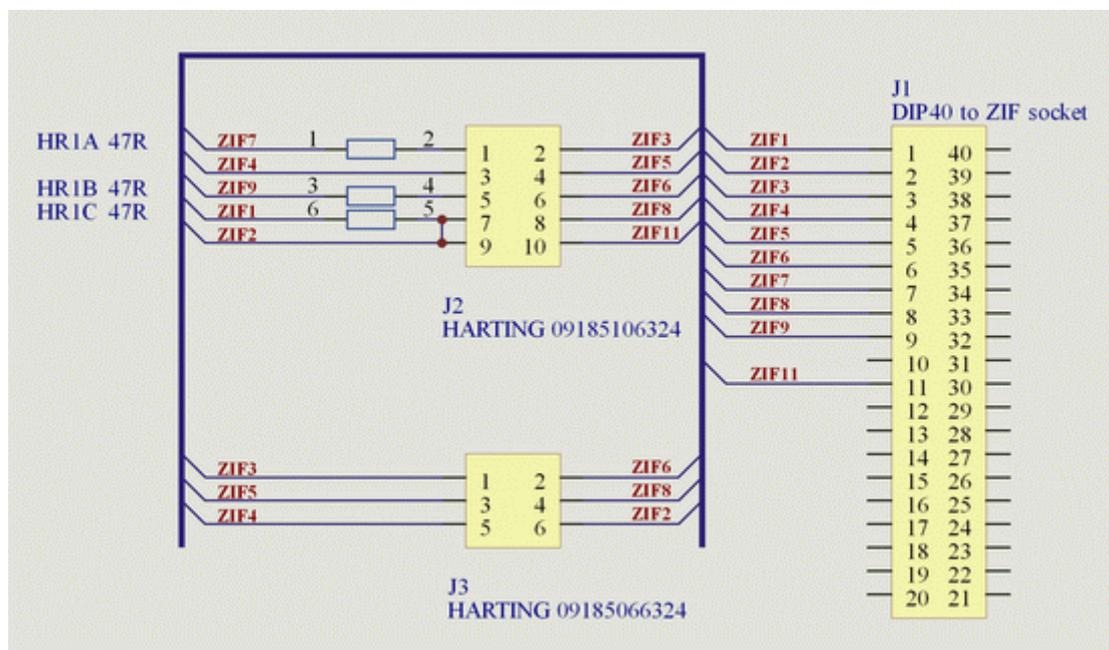
Эта команда позволяет выполнить самотестирование текущего программатора с использованием диагностического модуля POD, входящего в стандартную поставку программатора.

Тестирование программатора необходимо выполнять один раз в месяц, но для получения оптимальных результатов, рекомендуется эту процедуру проводить хотя-бы один раз в шесть месяцев.

Программатор / Само тестирование разъема ISP (PROGRAMMER / SELFTEST ISP CONNECTOR)

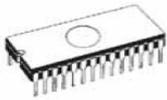
Эта команда позволяет провести само тестирование разъема ISP текущего программатора с использованием диагностического модуля POD для разъема ISP. Диагностический модуль используется для тестирования 6-и и 10-контактных разъемов ISP. Диагностический модуль может не входить в комплект поставки программатора. Для заказа модуля необходимо использовать его специальный номер: 70 – 0208.

Принципиальная схема диагностического модуля POD для разъема ISP:



Тестирование 6-контактного разъема ISP:

- Установить диагностический модуль POD для разъема ISP в гнездо ZIF программатора. Модуль устанавливается как стандартная микросхема в 40-выводном корпусе.
- Подключить 6-контактный разъем диагностического модуля к разъему ISP программатора с помощью кабеля ISP, находящегося в комплекте с программатором. Проверить правильность подключения (выводы должны совпадать: 1-1, 2-2,, 6-6).
- Запустить режим само тестирования разъема ISP в управляющей программе PG4UW.



Тестирование 10–контактного разъема ISP:

- Установить диагностический модуль POD для разъема ISP в гнездо ZIF программатора. Модуль устанавливается как стандартная микросхема в 40–выводном корпусе.
- Подключить 10–контактный разъем диагностического модуля к разъему ISP программатора с помощью кабеля ISP, находящегося в комплекте с программатором. Проверить правильность подключения (выводы должны совпадать: 1–1, 2–2,, 10–10).
- Запустить режим самотестирования разъема ISP в управляющей программе PG4UW.

Процедуру самотестирования рекомендуется проводить каждые шесть месяцев.

Программатор / Калибровка (PROGRAMMER / CALIBRATION TEST)

Эта команда выполняет тест калибровки программатора.

Программатор / Создать отчет о тестировании (PROGRAMMER / CREATE DIAGNOSTIC REPORT)

Эта команда создает диагностический отчет и помещает его в окно сообщений управляющей программы. Из окна сообщений информацию можно скопировать в другое приложение.

Эта функция полезна в том случае, когда пользователь не может найти причину отказа в работе программатора.

Список диагностических сообщений можно приложить к описанию проблемы и отправить разработчику программатора. Это позволит решить проблему в более короткие сроки.

Настройки (OPTIONS)

Меню **‘Настройки (Options)’** содержит команды для изменения текущей конфигурации управляющей программы, используемой по умолчанию.

Настройки / Основные настройки (OPTIONS / GENERAL OPTIONS)

В этом меню доступны для изменения основные настройки управляющей программы.

‘Ассоциация файлов (File options)’

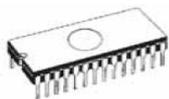
В этом разделе можно настроить ассоциацию форматов файлов их расширениям и параметры открытия/сохранения файлов по умолчанию.

Параметр **‘Маски форматов файлов (File format masks)’** используются для привязки расширения имени файла к его формату. Эти параметры используются при фильтрации имен файлов в диалоговых окнах **‘Файл / Сохранить (File / Save)’** и **‘Файл / Открыть (File / Load)’**. Маски файлов могут содержать специальные символы подмены (*, ?).

Параметр **‘Расширение имени файла проекта по умолчанию (Project file default extension)’** используется для привязки расширения имени файла проекта к его формату. Этот параметр используется при фильтрации имен файлов в диалоговых окнах **‘Файл / Открыть проект (File / Load project)’** и **‘Файл / Сохранить проект (File / Save project)’**.

Параметр **‘Замечено изменение открытого файла другим процессом (When current file is modified by another process)’** используется для настройки реакции управляющей программы на изменение открытого файла другим процессом. Возможные варианты:

- **‘Предупредить (Prompt)’** – будет выдано предупреждающее сообщение об изменении файла.
- **‘Обновить (Auto-reload)’** – сообщения не выводятся, обновление файла происходит автоматически.
- **‘Игнорировать (Ignore)’** – изменение файла игнорируется.



Проверка изменения файла проводится в случаях:

- возврат в окно управляющей программы из другого приложения.
- выбор операции с микросхемой – ‘Сравнить (Verify)’ или ‘Прошить (Program)’.
- при повторе последней операции с микросхемой в пакетном режиме.

Параметр ‘**Определять формат файла данных (Load file format)**’ используется для выбора режима определения формата файла данных при его открытии. Если выбран режим определения формата файла данных вручную, пользователь может сам выбрать формат. В этом случае тестирование файла данных не проводится и если формат выбран неверно, данные в буфере управляющей программы будут искажены.

‘HEX-файл (Hex file options)’

В этом разделе можно изменить настройки для работы с файлами в формате ‘HEX’.

Параметр ‘**Очистить буфер (Erase buffer)**’ используется для автоматической очистки буфера управляющей программы перед открытием файла данных. Переменная для заполнения буфера будет выбрана из параметра ‘**Значение (Value)**’.

Параметр ‘**Установить отрицательное смещение адреса буфера для загрузки (Negative offset for loading)**’ используется для определения адреса смещения в буфере управляющей программы для открываемого файла данных. Автоматическое смещение адреса рекомендуется только в специальных случаях. Этот параметр использует эвристический анализ и может неправильно определить данные в открываемом файле. Особенно критичны к этому параметру фрагментированные файлы данных, объем которых может превышать объем памяти используемой микросхемы – в этом случае некоторые блоки данным могут быть проигнорированы.

Пример:

Файл содержит данные в формате ‘Motorola S’. Стартовый адрес блока данных равен FFFF0H. Данный формат типа ‘S2’ имеет размер массива адресов равный 3 байтам. Для всех данных можно установить смещение равным FFFF0H. Это означает что смещение будет вычтено из всех адресов таким образом, что данные в буфере управляющей программы будут расположены с адреса равного 0.

Замечание: Значение отрицательного смещения вычитается из текущего адреса данных. При этом может получиться отрицательное число. Следите за корректностью ввода данных для смещения.

‘Язык интерфейса (Language)’

Этот раздел предназначен для выбора необходимых файлов–модулей интерфейса управляющей программы и файла помощи. Для вступления изменений в силу требуется перезапуск управляющей программы.

‘Звуки (Sound)’

Этот раздел помогает настроить звуковые сигналы управляющей программы и программатора.

‘Настройки управляющей программы (Sound settings)’ используются для звукового оповещения важных событий управляющей программы (программирование, верификация, чтение...).

‘Настройки программатора (Programmer internal speaker sound settings)’ используются для звукового оповещения событий с помощью внутреннего динамика программатора.

‘Лог–файл (Log file)’

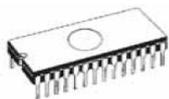
Этот раздел позволяет настроить дополнительный вывод сообщений окна управляющей программы **‘Сообщения управляющей программы (Log window)’**. Все сообщения могут также выводиться и в текстовый файл с именем по умолчанию **‘Report.rep’**. Самостоятельно можно изменить папку и имя лог–файла для записи сообщений.

‘Ошибки (Display errors)’

Этот раздел позволяет настроить вывод сообщений об ошибках на экран или в файл с именем по умолчанию **‘Verify.err’**. Самостоятельно можно изменить папку и имя файла для записи сообщений об ошибках.

‘Удаленный доступ (Remote control)’

Этот раздел предназначен для настройки сетевых параметров управляющей программы.



Для удаленного контроля за управляющей программой с помощью другой управляющей программы используется протокол TSP.

Настройки протокола TSP по умолчанию :

Порт: **telnet** Адрес: **127.0.0.1** или **localhost**

Для получения подробной информации по работе с режимом удаленного доступа следует обратиться к документации 'remotemanual.pdf', расположенной в папке 'RemoteCtrl\' управляющей программы

***Замечание:** Если операционная система использует в своей работе фаерволл или межсетевой экран, будет получено предупреждающее сообщение, на основании которого необходимо создать разрешающее правило для корректной работы удаленного доступа для управляющей программы.*

‘Сохранение (Save options)’

Этот раздел позволяет установить настройки для сохранения текущих параметров при выходе из управляющей программы. Доступные варианты:

- **‘Никогда не сохранять (Don't save options)’** – настройки управляющей программы не сохраняются при выходе.
- **‘Сохранять автоматически (Auto save options)’** – настройки управляющей программы сохраняются автоматически при выходе.
- **‘Спрашивать (Prompt for save options)’** – при выходе из управляющей программы будет показано диалоговое окно с предложением сохранить настройки управляющей программы.

‘Дополнительно (Other)’

Этот раздел позволяет настроить остальные необязательные параметры управляющей программы.

Параметр **‘Приоритет управляющей программы (Application priority)’** позволяет установить приоритет выполнения управляющей программы на фоне остальных приложений операционной системы. Чем выше приоритет, тем быстрее выполняются операции с микросхемами.

Параметр **‘Кнопки панелей инструментов (Tool buttons)’** позволяет настроить визуальное отображение кнопок на панелях инструментов.

Параметр ‘Текущая папка (Start-up directory)’ позволяет выбрать папку по умолчанию для работы с файлами при запуске управляющей программы.

Настройка / View (OPTIONS / VIEW)

Команды этого меню позволяют настроить ряд параметров по отображению панелей инструментов.

Настройка / View / Основная панель инструментов (OPTIONS / VIEW / MAIN TOOLBAR)

Этот пункт позволяет показать/скрыть основную панель инструментов.

Настройка / View / Дополнительная панель инструментов (OPTIONS / VIEW / ADDITIONAL TOOLBAR)

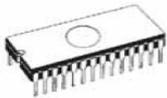
Этот пункт позволяет показать/скрыть дополнительную панель инструментов.

Настройка / View / Спрашивать настройки ИС (OPTIONS / VIEW / DEVICE OPTIONS BEFORE DEVICE OPERATION)

Этот пункт позволяет показать/скрыть дополнительный диалог с настройками для текущей микросхемы перед началом работы с этой микросхемой.

Настройка / Защищенный режим (OPTIONS / PROTECTED MODE)

‘Защищенный режим (Protected mode)’ – это специальный режим, в котором ограничены многие функции управляющей программы в целях безопасности. В защищенном режиме недоступны функции по редактированию буфера, настроек ИС. В защищенном режиме допускается программирование только той микросхемы, с которой проводилась работа до включения защищенного режима.



Переключить программу в защищенный режим можно двумя способами:

- Использовать команду меню '**Настройки / Защищенный режим (Options / Protected mode)**'. Диалоговое окно позволяет ввести пароль для защищенного режима и повторить ввод пароля для проверки правильности ввода. Этот же пароль потребуется для отключения защищенного режима.
- Открыть файл проекта, который был защищен паролем при его предыдущем сохранении в файл на диск.

Смотрите также: '**Файл / Сохранить проект (File / Save project)**'.

Для переключения управляющей программы из защищенного режима в нормальный режим, можно использовать команду меню '**Настройки / Обычный режим (Options/Normal mode)**'. В диалоговом окне потребуется ввести пароль, который был введен при включении защищенного режима.

Дополнительным способом выхода из защищенного режима управляющей программы является обычный выход из программы и ее повторный запуск. При этом управляющая программа будет запущена с обычными настройками, но файл проекта и его настройки, которые были установлены до выхода из программы, будут недоступны (потребуется ввод пароля).

Настройки / Сохранить настройки (OPTIONS / SAVE OPTIONS)

Эта команда сохраняет все доступные настройки и конфигурацию управляющей программы. Для сохранения доступны: основные настройки управляющей программы, список последних 10 использовавшихся микросхем, история файлов данных и файлов проектов, размеры и позиция главного окна управляющей программы.

Помощь (HELP)

Меню ‘Помощь (Help)’ содержит команды с помощью которых можно выяснить списки программаторов и микросхем, поддерживаемых управляющей программой, а также проверить текущую версию управляющей программы.

Помощь / Поддерживаемые ИС (HELP / SUPPORTED DEVICES)

Эта команда отображает список всех ИС поддерживаемых всеми моделями программаторов. Поиск требуемой ИС может производиться по названию, по типу, по производителю или по маске.

Префикс ‘g_’ перед названием микросхемы указывает на то, что поддерживается работа мультисокетным программатором (например JetProg).

Помощь / Поддерживаемые программаторы (HELP / SUPPORTED PROGRAMMERS)

Эта команда отображает краткий список всех программаторов, поддерживаемых текущей версией управляющей программы.

Помощь / Список ИС (текущий программатор) (HELP / DEVICE LIST (CURRENT PROGRAMMER))

Эта команда создает список всех микросхем, поддерживаемых текущим программатором и сохраняет этот список в текстовый файл примерного вида ‘?????dev.txt’ и HTML файл примерного вида ‘?????dev.htm’. Файлы создаются в папке установки управляющей программы. При создании файлов маска ????? подменяется сокращенным названием текущего программатора, для которого этот список создается.



Помощь / Список ИС (все программаторы) (HELP / DEVICE LIST (ALL PROGRAMMERS))

Эта команда создает список всех микросхем, поддерживаемых всеми программаторами и сохраняет этот список в текстовый файл вида '?????dev.txt' и HTML файл вида '?????dev.htm'. Файлы создаются в папке установки управляющей программы. При создании файлов маска ?????? подменяется сокращенным названием программатора, для которого этот список создается.

Замечание: Применение этой команды устанавливает все настройки управляющей программы по умолчанию (первый запуск после установки управляющей программы) и уничтожает все несохраненные настройки и данные в управляющей программе.

Помощь / Список ИС (полный список) (HELP / DEVICE LIST (CROSS REFERENCE))

Эта команда создает список всех микросхем, поддерживаемых всеми программаторами для текущей версии управляющей программы. Результаты создания списка помещаются в несколько файлов:

- главный файл HTML 'top_dev.htm' содержит список производителей микросхем.
- отдельные группы файлов HTML содержащие списки всех поддерживаемых микросхем для данного производителя.

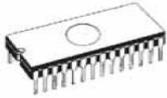
Главный файл HTML будет создан в папке установки управляющей программы. Группы файлов HTML будут созданы в папке 'dev_html\' расположенной в папке установки управляющей программы.

Помощь / О программе (HELP / ABOUT)

Это меню поможет выяснить текущую версию управляющей программы, авторские права и некоторую дополнительную информацию.



Примечания



Режим командной строки управляющей программы

Кроме обычного режима запуска управляющей программы – графического интерфейса, доступен режим запуска с помощью командной строки.

Основные правила по использованию параметров командной строки:

- Параметры командной строки нечувствительны к регистру вводимых символов;
- Параметры командной строки допустимо использовать при запуске управляющей программы или в том случае, когда управляющая программа уже запущена;
- Если управляющая программа уже запущена, параметры командной строки будут применены, когда управляющая программа закончит текущую операцию. Параметры командной строки не будут применены, если в управляющей программе открыты какие-либо дополнительные меню или диалоговые окна;
- Параметры командной строки рекомендуется применять в следующей последовательности:
 1. Открыть файл (/Loadfile:...).
 2. Открыть файл проекта (/Prj:...).
 3. Автоматическое определение типа микросхемы (/Eprom_Flash_Autoselect:...).
 4. Программирование микросхемы (/Program:...).
 5. Выход из управляющей программы (/Close использовать в сочетании с параметром /Program).

Параметры командной строки:

/Axxx проверка подключения программатора к порту LPT с адресом **xxx**.

Пример:

/A3bc

/SPP программатор подключен в однонаправленном режиме.

Дополнительные параметры командной строки:

/Prj:<file_name> загрузка файла проекта в управляющую программу при ее запуске или в том случае, когда управляющая программа уже запущена. Параметр **<file_name>** может содержать путь, указывающий местоположение файла проекта.

/Loadfile:<file_name> загрузка файла в управляющую программу при ее запуске или в том случае, когда управляющая программа уже запущена. Параметр **<file_name>** может содержать путь, указывающий местоположение файла. Формат загружаемого файла определяется автоматически.

/Program:xxx начало программирования микросхемы при запуске управляющей программы или в том случае, когда управляющая программа уже запущена. Параметр **xxx** может принимать значения:

- **Noquest** начало программирования микросхемы без дополнительных сообщений перед началом процесса программирования;
- **Noanyquest** начало программирования микросхемы без дополнительных сообщений перед началом процесса программирования. После окончания процесса программирования сообщение о повторе операции программирования не выводится.

Пример:

/Program

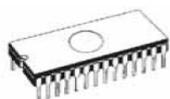
/Program:noquest

/Program:noanyquest

/Close автоматический выход из управляющей программы после окончания программирования микросхемы, независимо от результатов программирования. Используется только в сочетании с параметром **/Program**.

/Eprom_Flash_Autoselect:xxx автоматическое определение типа микросхемы EPROM или FLASH по идентификатору при запуске управляющей программы или в том случае, когда управляющая программа уже запущена.

Параметр **xxx** задает количество выводов микросхемы, установленной в гнездо ZIF программатора (допустимо указывать количество выводов 28 или 32). Функция автоматического определения работоспособна только на старших моделях программаторов. Для остальных моделей программаторов параметр игнорируется.



Аппаратная совместимость

В настоящее время существует огромное количество персональных компьютеров, оборудованных параллельными портами (LPT), построенными на основе микросхем различных производителей, зачастую не соответствующих предъявляемым стандартам в полной мере.

По указанной причине, не исключено возникновение ситуации, в которой программатор не будет функционировать требуемым образом ввиду несовместимости программатора с оборудованием пользователя. Внешнее проявление этой проблемы проявляется в том, что управляющая программа не может опознать программатор, подключенный к параллельному порту персонального компьютера.

В данной ситуации компания ELNEC рекомендует произвести процедуры по подключению программатора и установке управляющей программы на другом компьютере, желательно иного производителя, для того, чтобы пользователь мог удостовериться в работоспособности программатора.

В том случае, если программатор, подключенный к другому компьютеру работает корректно, компания ELNEC не сможет нести ответственность за несовместимость с оборудованием пользователя, но попытается смоделировать возникшую ситуацию и найти решение возникшей проблемы после того как получит дефектный листок с точным описанием проблемы.

Внутрисхемное программирование (ISP)

Режим внутрисхемного программирования – это такой режим, в котором реализована программно–аппаратная возможность программирования и перепрограммирования содержимого микросхемы с помощью последовательного алгоритма без ее демонтажа из законченного изделия.

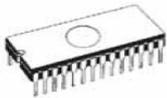
Такой подход позволяет экономить значительную часть материальных средств и времени при разработке оборудования и его сервисном обслуживании.

- **программируемая микросхема (Target device)** – микросхема (микроконтроллер), в составе некоторой системы, подлежащая программированию;
- **программируемая система (Target system)** – некоторая система, содержащая в своем составе микросхему (микроконтроллер), подлежащую программированию;
- **программатор ISP** – программатор, программно–аппаратная часть которого позволяет использовать алгоритм последовательного программирования для микроконтроллеров, в соответствии с требованиями производителя микросхем.

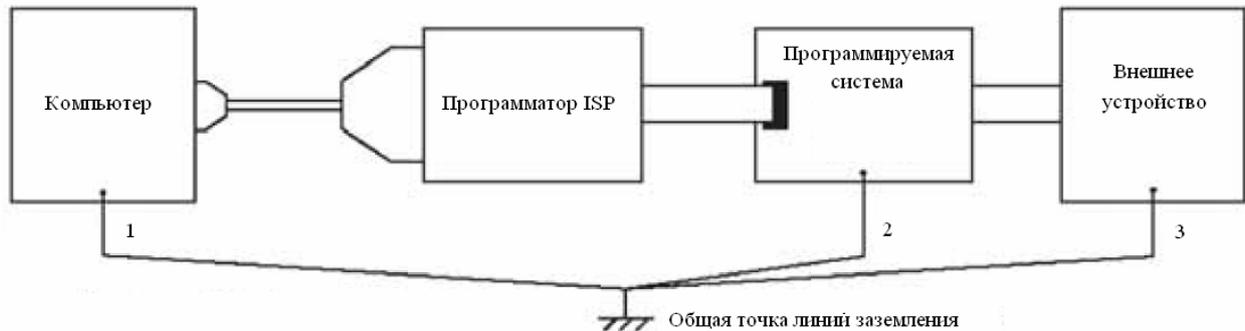
Требования совместимости и безопасности

Общие правила для программирования в режиме ISP помогут избежать множества проблем с программированием в этом режиме и исключат вероятность повреждением программируемых микросхем и программатора:

- в первую очередь обеспечьте общую линию заземления для компьютера и программируемой системы;
- на тех компьютерах, где невозможно реализовать общую линию заземления по причине отсутствия стандартного контакта заземления, используйте один из контактов GND порта LPT или COM компьютера, для объединения линии заземления;
- любые внешние устройства, подключенные к программируемой системе должны иметь общую линию заземления с компьютером и программируемой системой.



На рисунке приведены принципы реализации общей линии заземления для режима ISP:



Для успешного и безопасного для оборудования программирования в режиме ISP следует строго придерживаться перечисленных ниже правил:

- выключить питание программатора и устройства, подлежащего программированию;
- соединить контакты линий заземления компьютера, программируемой системы и внешних устройств в единую линию заземления;
- подключить кабель ISP к разъему ISP программатора и включить питание программатора;
- запустить управляющую программу и выбрать соответствующий режим программирования для микросхемы;
- установить режим программирования на программируемой системе;
- после настройки управляющей системы, подключить свободный разъем кабеля ISP к программируемой системе и начать процесс программирования;
- после окончания программирования отключить кабель ISP от программируемой системы.

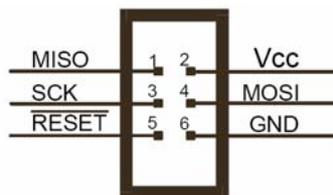
Замечание: Несоблюдение последовательности выполнения этих правил приведет к повреждению и выходу из строя программатора и программируемой системы.

Схемотехническое обеспечение режима ISP

Для обеспечения надежного и безопасного программирования, схемотехническое решение программируемой системы должно отвечать основному требованию – при программировании в режиме ISP, способ подключения микроконтроллера к программируемой системе должен обеспечивать изоляцию микроконтроллера от цепей этой системы или сводить к минимуму влияние цепей системы на процесс программирования.

Режим ISP для микроконтроллеров Atmel AVR

В соответствии с рекомендациями AVR910 компании Atmel по стандартизации интерфейса ISP, был разработан интерфейс разъема ISP для программатора T51prog со следующим назначением выводов:



Вывод	Название	Описание
SCK	Serial Clock (Синхроимпульс)	Синхроимпульс для режима ISP. Генерируется программатором.
MOSI	Master Out – Slave In (Ведущий выход – Ведомый вход)	Линия связи. Программатор (ведущий) – микроконтроллер (ведомый).
MISO	Master In – Slave Out (Ведущий вход – Ведомый выход)	Линия связи. Микроконтроллер (ведомый) – Программатор (ведущий).
GND	Ground (Общий)	Общая линия, соединяющая два устройства.
RESET	Reset (Сброс микроконтроллера)	Для режима ISP сигнал RESET должен быть активным. Генерируется программатором.
Vcc	Power (Питание микроконтроллера)	Линия питания. Для поддержки устройств с разными напряжениями питания, генерируется программатором.

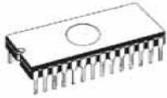


Схема реализации режима ISP для Atmel AVR:

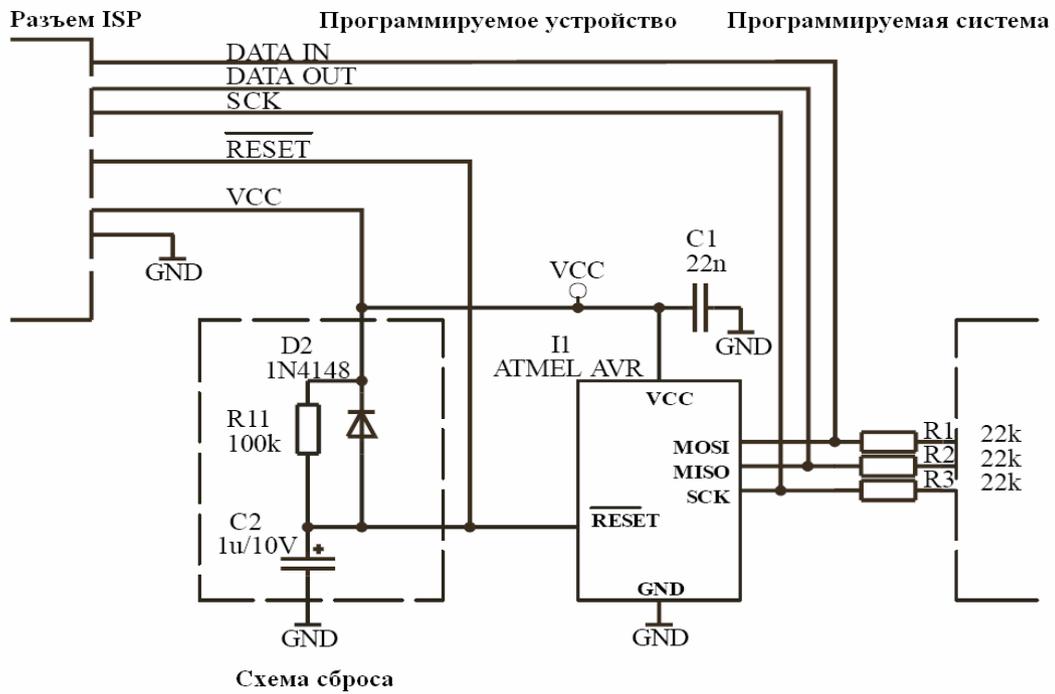
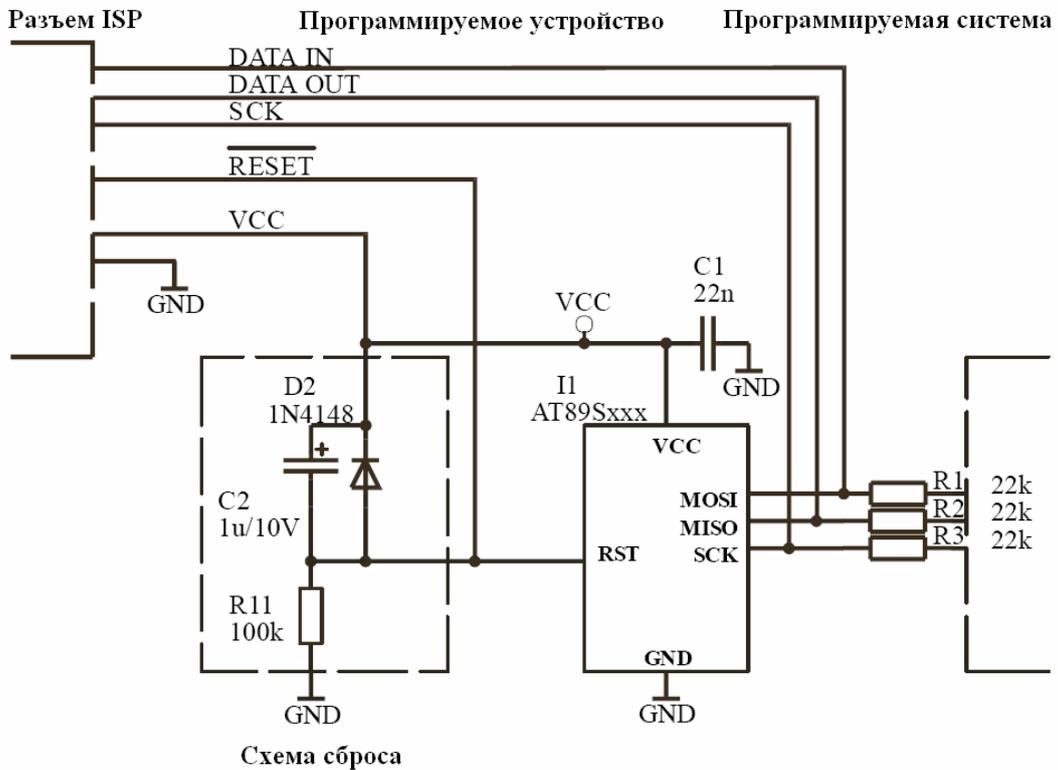


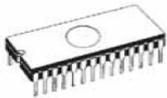
Схема реализации ISP для AT89Sxxx:



- для уменьшения влияния программируемой системы на процесс программирования рекомендуется установка резисторов R1, R2 и R3.
- в том случае, когда резисторы оказывают влияние на работу программируемого устройства в обычном режиме, допускается установка джамперов–перемычек или миниатюрных DIP–переключателей, с помощью которых микроконтроллер в режиме программирования отключается от программируемой системы.

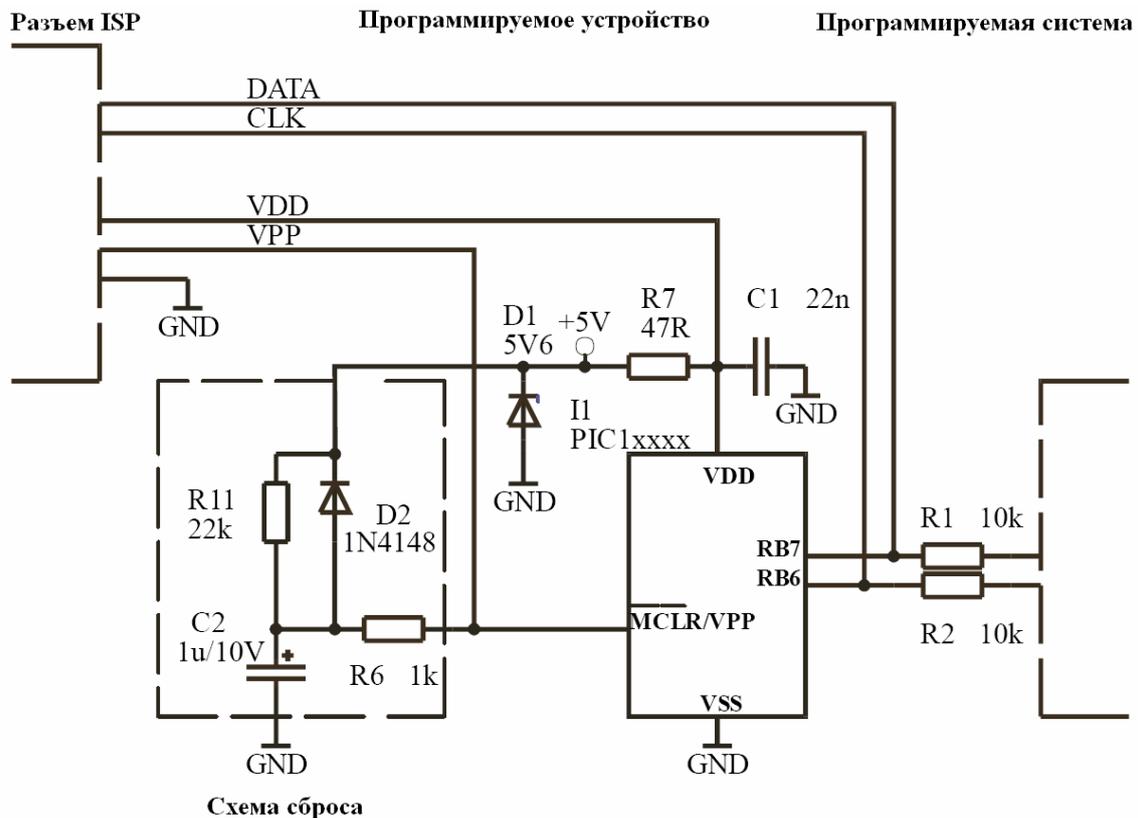
Выводы MOSI, MISO и SCK предназначены для передачи данных и их синхронизации в режиме программирования. По этой причине программируемая система не должна оказывать никакого влияния на эти выводы во время всего процесса программирования.

Схема сброса необходима лишь в том случае, если время стабилизации напряжения питания на требуемом уровне слишком большое (медленное нарастание напряжения питания).



Режим ISP для микроконтроллеров PICmicro®

Схема реализации ISP для PICmicro:



- стабилитрон D1 обеспечивает защиту цепей программируемой системы от повышенного напряжения питания в режиме программирования микроконтроллера.
- в том случае, когда напряжение питания отличается от типичного, необходимо подобрать стабилитрон соответствующего номинала.
- для уменьшения влияния программируемой системы на процесс программирования рекомендуется установка резисторов R1 и R2.
- в том случае, когда резисторы оказывают влияние на работу программируемого устройства в обычном режиме, допускается установка джамперов–перемычек или миниатюрных DIP–переключателей, с помощью которых микроконтроллер в режиме программирования отключается от программируемой системы.

Приведенная выше схема построения программируемой системы для режима ISP разработана на основании рекомендаций TB013, TB016, TB017 компании Microchip™.

Назначение выводов разъема ISP:

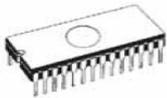
MCLR\ /VPP	сброс, переход в режим программирования микроконтроллера.
RB6 (GP1)	синхроимпульс.
RB7 (GP0)	данные.
VDD	напряжение питания.
GND	общий.

После окончания программирования микроконтроллера, напряжение на выводе MCLR\ принимает значение близкое к +12V. По этой причине следует обезопасить цепи программируемой системы от повреждения с помощью стабилитрона D1.

Выводы RB6 и RB7 предназначены для передачи данных и их синхронизации в режиме программирования. По этой причине программируемая система не должна оказывать никакого влияния на эти выводы во время всего процесса программирования.

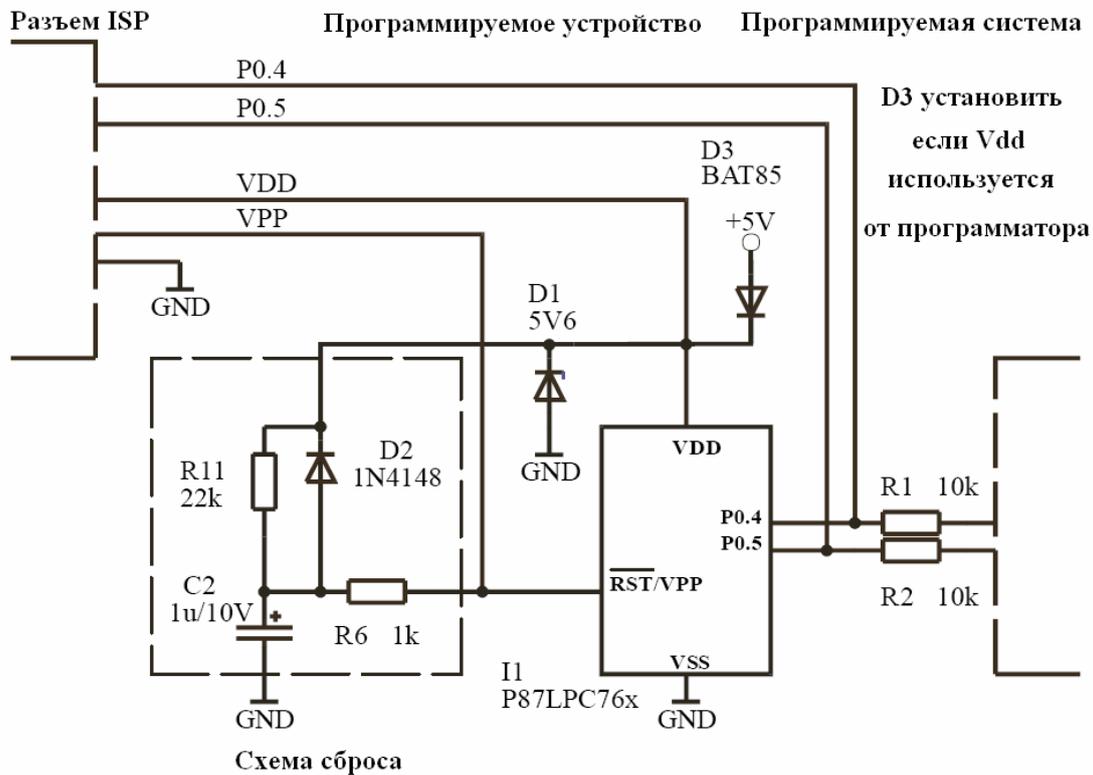
После окончания программирования, проводится проверка качества записи при минимальном и максимальном напряжении питания на выводе VDD. По этой причине программируемая система не должна оказывать никакого заметного влияния на вывод VDD.

Схема сброса необходима лишь в том случае, если время стабилизации напряжения питания на требуемом уровне слишком большое (медленное нарастание напряжения питания).



Режим ISP для микроконтроллеров Philips

Схема реализации ISP для P87LPC76x:



Приведенная выше схема построения программируемой системы для режима ISP разработана на основании рекомендаций AN466 компании Philips.

- стабилитрон D1 обеспечивает защиту цепей программируемой системы от повышенного напряжения питания в режиме программирования микроконтроллера.
- в том случае, когда напряжение питания отличается от типичного, необходимо подобрать стабилитрон соответствующего номинала.
- для уменьшения влияния программируемой системы на процесс программирования рекомендуется установка резисторов R1 и R2.
- в том случае, когда резисторы оказывают влияние на работу программируемого устройства в обычном режиме, допускается установка джамперов–перемычек или миниатюрных DIP–переключателей, с помощью которых микроконтроллер в режиме программирования отключается от программируемой системы.

Назначение выводов разъема ISP:

RST\ /VPP	сброс, переход в режим программирования микроконтроллера.
P0.5	синхроимпульс.
P0.4	данные.
VDD	напряжение питания.
GND	общий.

После окончания программирования микроконтроллера, напряжение на выводе RST\ принимает значение близкое к +10.75V. По этой причине следует обезопасить цепи программируемой системы от повреждения с помощью стабилитрона D1.

Выводы P0.4 и P0.5 предназначены для передачи данных и их синхронизации в режиме программирования. По этой причине программируемая система не должна оказывать никакого влияния на эти выводы во время всего процесса программирования.

Схема сброса необходима лишь в том случае, если время стабилизации напряжения питания на требуемом уровне слишком большое (медленное нарастание напряжения питания).

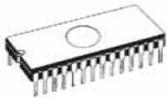
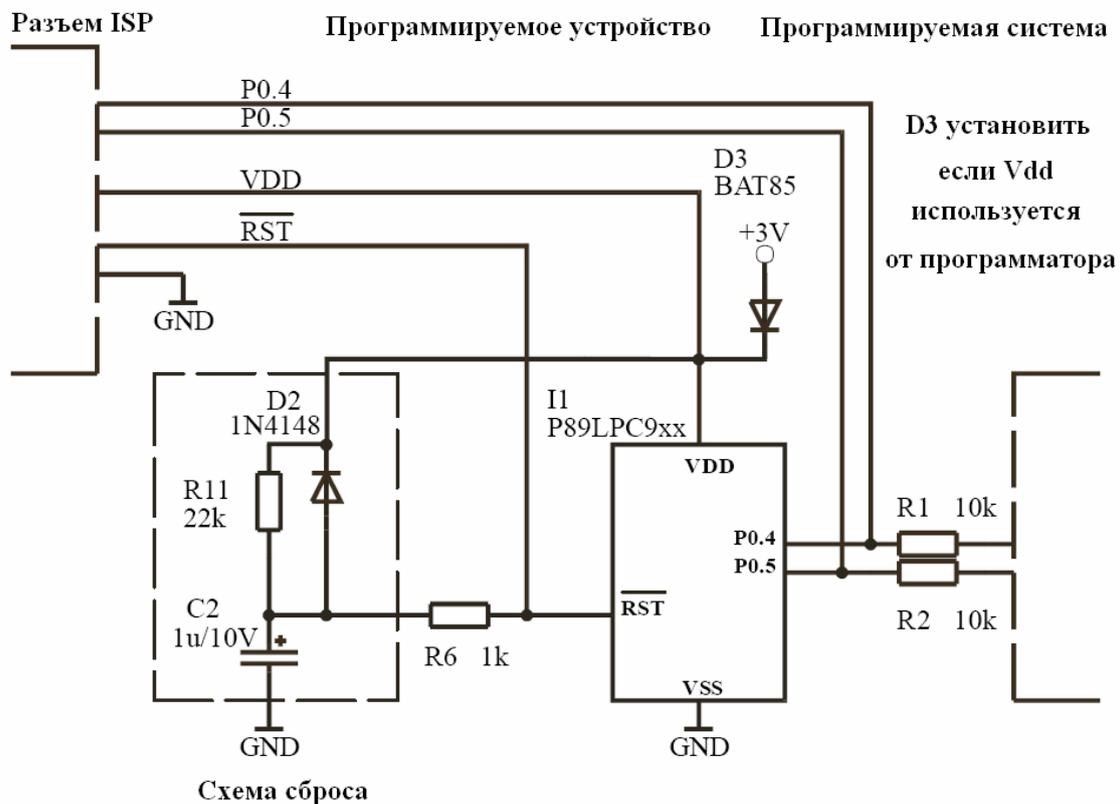


Схема реализации ISP для P89LPC9xx:



Приведенная выше схема построения программируемой системы для режима ISP разработана на основании рекомендаций AN466 компании Philips.

- для уменьшения влияния программируемой системы на процесс программирования рекомендуется установка резисторов R1 и R2.
- в том случае, когда резисторы оказывают влияние на работу программируемого устройства в обычном режиме, допускается установка джамперов–перемычек или миниатюрных DIP–переключателей, с помощью которых микроконтроллер в режиме программирования отключается от программируемой системы.

Назначение выводов разъема ISP:

RST\ /VPP	сброс, переход в режим программирования микроконтроллера.
P0.5	синхроимпульс.
P0.4	данные.
VDD	напряжение питания.
GND	общий.

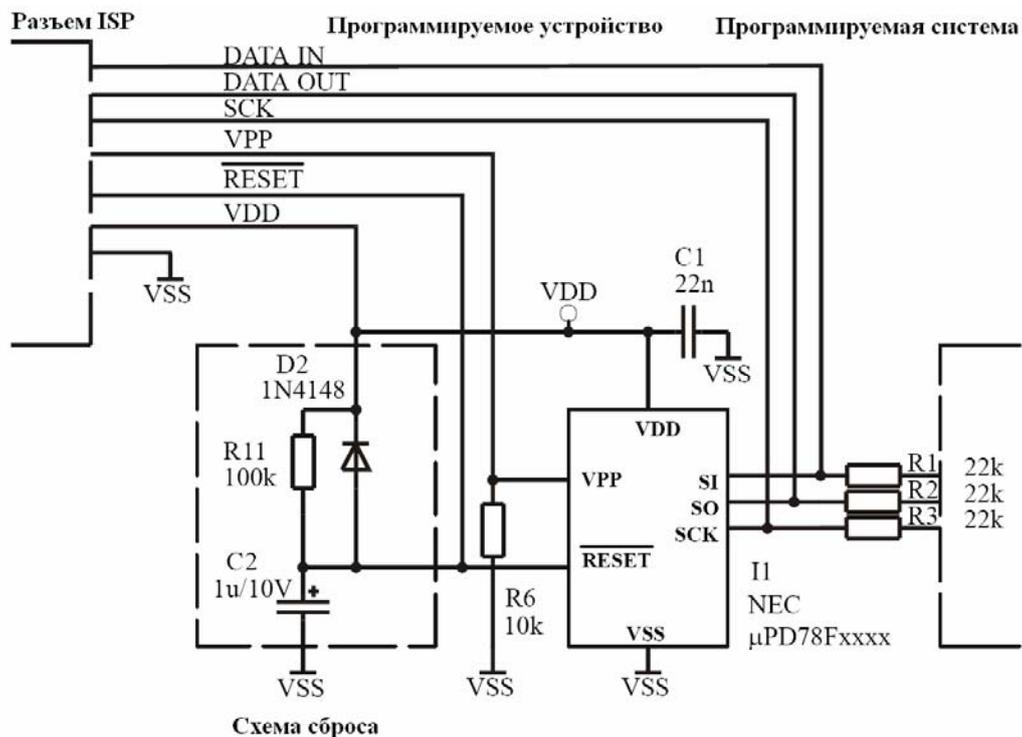


Выводы P0.4 и P0.5 предназначены для передачи данных и их синхронизации в режиме программирования. По этой причине программируемая система не должна оказывать никакого влияния на эти выводы во время всего процесса программирования.

Схема сброса необходима лишь в том случае, если время стабилизации напряжения питания на требуемом уровне слишком большое (медленное нарастание напряжения питания).

Режим ISP для микроконтроллеров NEC

Схема реализации ISP для μ PD78Fxxxx:

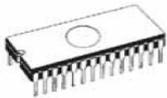


Назначение выводов разъема ISP:

RESET\	сброс.
VPP	переход в режим программирования.
SI	данные прием.
SO	данные передача.
SCK	синхроимпульс.
VDD	напряжение питания.
VSS	общий.

После окончания программирования микроконтроллера, напряжение на выводе VPP принимает значение близкое к +10V.

Выводы SI, SO, SCK предназначены для передачи данных и их синхронизации в режиме программирования. По этой причине программируемая система не должна оказывать никакого влияния на эти выводы во время всего процесса программирования. Вывод RESET также должен быть защищен от программируемой системы в процессе программирования.



Дополнительная информация

Программатор для своей работы использует прямой доступ к параллельному порту компьютера с помощью устанавливаемого драйвера прямого доступа к портам. При работе программатора не пытайтесь получить доступ к порту, занятому программатором – это может нарушить работу программатора и привести к порче программируемой микросхемы.

Во время работы программатора (активен светодиод **‘BUSY’**) не передвигайте информационные окна управляющей программы. В это время идет обмен информацией между компьютером и программатором. Перемещение информационных окон может привести к ошибкам программирования.

Драйвер порта LPT (LPT PORT DRIVER)

Для программаторов, подключаемых к порту LPT компьютера требуется установка в операционную систему драйвера порта LPT. Обычно драйвер устанавливается корректно при инсталляции управляющей программы, но в ряде случаев может возникнуть проблема с установкой драйвера (несколько портов LPT или полное отсутствие портов при установке управляющей программы).

Драйвер порта LPT при установке требует обязательного наличия порта LPT1 в операционной системе.

Проверка наличия портов в операционной системе:

- Кликнуть правой кнопкой мыши на значке **‘Мой компьютер (My computer)’** и выбрать пункт **‘Свойства (Properties)’**.
- В диалоговом окне пункте **‘Свойства системы (System properties)’** выбрать закладку **‘Оборудование (Hardware)’** и кликнуть на кнопке **‘Диспетчер устройств (Device manager)’**.
- В диалоговом окне **‘Диспетчер устройств (Device manager)’** выбрать пункт **‘Порты (COM и LPT) (Ports (COM & LPT))’**. После двойного клика на этом пункте можно просмотреть информацию о портах COM и LPT, присутствующих в системе.

В данном пункте обязательно должен присутствовать хотя–бы один порт LPT.

Если порт LPT присутствует или присутствуют несколько портов, но нет порта LPT1, проделайте следующие шаги:

- Дважды кликните на выбранном порте LPT для доступа к его настройкам.
- В диалоговом окне ‘Свойства (LPT port properties)’ выберите закладку ‘Параметры порта (Port settings)’.
- Измените текущий номер порта ‘LPT’ на номер ‘LPT1’ в пункте ‘Номер порта LPT (LPT Port Number)’.
- Кликните кнопку ‘ОК’.
- **Перезагрузите компьютер** для вступления изменений в силу, **даже если запрос на перезагрузку не появлялся.**

После этих действий управляющая программа должна корректно работать с программатором, подключенным ко порту LPT.

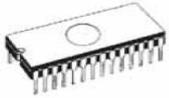
Если будет использоваться программатор подключенный к порту USB, установка драйвера для порта LPT не требуется.

Драйвер порта USB (USB PORT DRIVER)

Для программаторов, подключаемых к порту USB компьютера требуется корректная установка в операционную систему драйвера порта USB. Драйвер устанавливается при инсталляции управляющей программы.

Рекомендуется в первую очередь устанавливать управляющую программу, и только затем подключать программатор к порту USB. Операционная система автоматически обнаружит подключенный программатор.

Если программатор подключен до установки управляющей программы, операционная система обнаружив новое устройство запросит установку драйвера программатора. Для корректной установки драйвера необходимо в привод компакт-дисков компьютера поместить диск с программным обеспечением для программатора и выполнить действия по установке драйвера программатора в систему, отмечая необходимые пункты в ‘Мастере установки оборудования’.



Устранение неисправностей

Как действовать

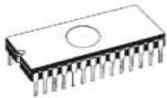
В случае возникновения отклонений в работе программатора, пожалуйста попробуйте решить проблему следующим образом:

- на сайте компании получите последнюю версию управляющей программы;
- удалите предыдущую версию управляющей программы и установите новую версию;
- перезагрузите операционную систему компьютера и проверьте работоспособность программатора;
- выключите питание компьютера и программатора, проверьте надежность и качество всех соединений;
- повторно включите компьютер и программатор;
- попробуйте подключить программатор к иному компьютеру или к другому параллельному порту (LPT) данного компьютера (если имеется);
- внимательно изучите имеющуюся документацию, в том числе, расположенную на сайте компании. Возможно Ваша проблема уже решена.

Если предложенные шаги положительного результата не принесли, обратитесь к торговому агенту или в авторизованный сервисный центр для обслуживания программатора.

Обратите внимание на то, что необходимо при себе иметь документы, подтверждающие дату покупки программатора и дефектный листок, с подробным описанием проблемы.

Замечание: Не пытайтесь самостоятельно отремонтировать программатор. Без достаточных знаний и навыков Вы можете нанести своим вмешательством значительный урон программатору. Кроме того, при самостоятельном вмешательстве в аппаратную часть программатора, Вы автоматически лишаетесь гарантийного обслуживания.



Что делать с неподдерживаемой микросхемой

При использовании программатора может возникнуть ситуация, когда предстоит работа с микросхемой, которая не поддерживается программатором.

В ряде случаев возможно решение возникшей проблемы без значительных материальных затрат:

- Проверьте список микросхем, поддерживаемых программатором, расположенный для свободного доступа в Интернет, на сайте компании ELNEC, по адресу: www.elnec.com. Возможно требуемая микросхема уже включена в список. Для начала работы с микросхемой достаточно загрузить на компьютер и установить последнюю версию управляющей программы.
- Если микросхемы в списке нет, заполните **‘Дефектный листок’**, находящийся в данном **‘Приложении’** или карточку **‘AlgOR’** и отправьте в компанию ELNEC. Возможно потребуется предоставить экземпляр неподдерживаемой микросхемы.

Контактная информация

Почтовый адрес: Elnec spol. s r. o.
Post 5
P.O. Box 22
SK-080 05 Presov
Slovak Republic

Телефон: +421-51-7734328, +421-51-7731007

Факс: +421-51-7732797

Электронная почта: elnec@elnec.com

Адрес в Интернет: www.elnec.com

Гарантийные обязательства

Жесткий контроль качества производимой продукции на всех этапах, использование качественных и надежных комплектующих, постоянное совершенствование аппаратной и программной части программаторов, позволило компании ELNEC предоставить пользователям гарантийное обслуживание своей продукции в течение трех лет с момента приобретения. Сроки гарантийного обслуживания могут быть изменены по усмотрению торгового агента.

Гарантийный срок работы гнезда программатора (DIL ZIF socket) оценен его производителями в 25'000 срабатываний, а гарантийный срок работы гнезд (ZIF socket) адаптеров для программатора оценен его производителями в 10'000 срабатываний.

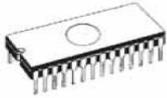
В случае выхода программатора или его частей из строя, в течение гарантийного периода, по вине компании или из-за использования дефектных комплектующих, ремонт будет произведен компанией ELNEC или авторизованными сервисными центрами бесплатно.

При обращении в компанию ELNEC или авторизованный сервисный центр для гарантийного обслуживания, необходимо при себе иметь документы, подтверждающие дату приобретения программатора и заполненный дефектный листок с указанием дефекта.

Гарантийное обслуживание не распространяется на продукцию в следующих случаях:

- наличие механических повреждений или естественный износ механизмов и частей программатора.
- следы ремонта лицами, не уполномоченными компанией ELNEC.
- следы изменения или модернизации конструкции программатора.
- неправильное использование или неправильная установка и эксплуатация программатора, включая повреждение статическим электричеством, а также его использование не по назначению.

В случае отказа в гарантийном обслуживании, специалисты компании ELNEC могут провести платный ремонт программатора с уведомлением о стоимости ремонта и сроках его проведения. Решение об отказе в гарантийном обслуживании может быть обжаловано, в порядке, предусмотренном действующим законодательством.

**Дефектный листок (Русская версия)**

Пожалуйста посетите сайт компании по адресу: www.elnec.com и заполните дефектный листок. Если Вы не имеете доступа к сети Интернет, распечатайте дефектный листок и отправьте в нашу компанию одним из способов, удобных для Вас – с помощью факса, с помощью почтовой службы или передайте торговому агенту.

Проблема: _____ Дата: _____

Клиент

Имя: _____ Торговый агент: _____
 Адрес: _____ Дата покупки: _____
 Контактная информация: _____ Дата отправки рег. карточки: _____

Информация о продукте

Программатор (тип/название): _____ Напряжение шины: _____ В (51&AVRprog, PIKprog)
 Серийный номер: _____ Версия управляющей программы: _____
 Конфигурация (адаптеры): _____
 Сетевой адаптер питания: Поставляемый в комплекте Иной (напряжение, ток): _____

Информация о компьютере

Производитель/тип/платформа: _____ Настольный Ноутбук
 Процессор/частота: _____ Порт LPT: Встроенный Плата ISA Плата PCI
 Операционная система: _____ Порт LPT: Стандартный ECP/EPP 1284
 Оперативная память: _____ Порт LPT: SPP BIDIР EPP ECP

Информация о проблемной микросхеме

Полное наименование (название, индексы): _____
 Материал корпуса: Пластик Керамика Другой материал корпуса: _____
 Производитель/логотип: _____
 Надписи на корпусе: _____
 Тип корпуса (DIP40, PLCC...): _____

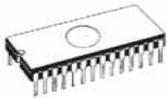
Периодичность проблемы: каждые _____ дней _____ недель _____ месяцев
 Использование программатора: часто редко один раз
 Количество запрограммированных микросхем на программаторе: _____
 К дефектному листку приложен образец микросхемы: Да Нет

Дополнительные вопросы

У Вас установлена последняя версия управляющей программы? Да Нет
 Вы знаете все особенности работы с программатором и микросхемой? Да Нет
 Гнездо ZIF socket программатора и гнезда адаптеров защищены от пыли? Да Нет
 Ваш программатор новый? Да Нет
 Проблема существует со всеми микросхемами? Да Нет У меня всего один чип
 Проблема существует с разными файлами данных? Да Нет Других файлов нет
 Когда возникает проблема? Проверка установки Чтение идентификатора
 Проблема существует при работе с другими типами микросхем? Да Нет
 Проблема существует всегда или ее появление хаотично? Всегда Хаотично
 Проблема существует при работе на другом компьютере? Да Нет
 Результаты самотестирования хорошие или плохие? Хорошие Плохие

Пожалуйста дополнительно перечислите последовательность своих действий, вызывающих проблему. Опишите все действия настолько подробно – это поможет быстрее найти решение проблемы. Опишите действия, вызывающие проблему или действия, наоборот предотвращающие возникновение проблемы. Скопируйте полностью всю информацию из окна сообщений управляющей программы, в том числе все сообщения при самотестировании программатора. Дополните информацию своими размышлениями по поводу возникновения проблемы. Наилучшим вариантом является пересылка в адрес компании экземпляра микросхемы, с которым возникает проблема. Если Ваша информация достаточно объемна, используйте для записей несколько листов бумаги.

***Замечание:** Обязательно установите последнюю версию управляющей программы – наша компания не ведет архив предыдущих версий управляющей программы. По этой причине моделирование нами Ваших действий может не найти решения с новой версией управляющей программы. Возможно, имеется вероятность того, что Ваша проблема уже решена в новой версии управляющей программы.*

**Дефектный листок (English version)**

Subject(title of problem): _____ Date: _____

Customer

Customer, name: _____ Distributor, name: _____
 Address: _____ Date of purchasing: _____
 Contact person and e-mail: _____ Date of sending registration card: _____

Information about product.

Programmer (type/modification): _____ Mains supply voltage: _____ V (in case 51&AVRprog and PKprog)
 Serial number: _____ Version of control program PG4UW: _____
 Configuration (modules, converters): _____
 Power supply unit: From delivery Other (output V and A): _____

Information about PC, to which is the programmer is attached.

Manufacturer/Type: _____ Desktop Notebook
 Processor, speed: _____ LPT port location: motherboard ISA card PCI card
 Operating system and version: _____ LPT port type: standard ECP/EPP 1284
 Memory/free memory: _____ LPT port setting: SPP BIDIR EPP ECP

Information about device with which you have the problem.

Device type (full name, prefix/suffix including): _____ Package type: plastic ceramic ceramic/windowed
 Vendor/logo: _____ All designation on the top _____
 Package (DIL40, PLCC44, SOIC20, ...): _____ and on the bottom side of device _____

Precedence rating: in __ days in __ weeks in __ months
 How often you work with this devices: still Y/N sometimes Y/N one-shot Y/N
 Number of programmed device: approx. ___ pcs per year.
 Samples are available? Yes (I'm sending it/attached) Yes No

Further questions.

- Did you have installed latest version of control program? Yes No
- Did you know thoroughly the features and correct behavior of programmer and programmed device? Yes No
- Is the socket of programmer or adapter free from dust and isn't out of life? Yes No
- Is the device with problem new or used? New Used
- Is the error reported for all of the tested devices? Yes No I have only one device
- Is the error reported for devices with other date code? Yes No I have only one batch
- During which procedure is an error reported? Read Program ID_check Insertion test
- Is the programmer successful in case of other types of devices? Yes No
- Does the error occur always or randomly? Always Randomly
- Does programmer work well with other PCs? Yes No Not tested
- What is the results of programmer selftest (if available)? OK Error

Please list the step-by-step description of all activities that invokes a problem. Please make your problem description as specific as possible - you can increase speed and chance to resolve a problem. Please mention any step that is known to cause the problem or any step that may prevent the problem. Please copy all error reports too - full content of LOG window is preferable. We recommended use command Diagnostics/ Create diagnostic report. Your comments and descriptions of expectations are welcomed. It's best, if you can send us the actual device with which the problem occurs. Use a separate sheet if necessary.

Услуга Keep–Current

Keep–Current – это платная услуга, позволяющая пользователям незамедлительно получать самые последние версии программного обеспечения и документацию, а также всегда быть в курсе всех новостей и событий компании ELNEC;

Программное обеспечение и документация могут быть отправлены двумя способами:

- электронная почта;
- обычная почта.

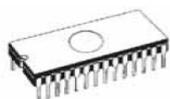
Конечный пользователь, в зависимости от своих потребностей, может выбрать один из трех пакетов услуги:

- **QUICK** – информация о выходе новых версий программного обеспечения и документации будет доставлена незамедлительно на адрес электронной почты.
- **STANDART** – информация о выходе новых версий программного обеспечения и документации будет доставлена незамедлительно на адрес электронной почты. Дополнительно раз в квартал по обычной почте будет отправлен CD, содержащий обновленные версии документации и программного обеспечения.
- **1/Y** – Один раз в год по обычной почте будет отправлен CD, содержащий обновленные версии документации и программного обеспечения.

Пакет услуги (файл KC_ARC.exe) включает в себя:

- последняя версия управляющей программы (PG4UWARC.exe);
- последняя версия списка поддерживаемых микросхем;
- информация о добавленных/удаленных из списка микросхемах и добавленных/удаленных алгоритмах программирования;
- информация о добавленных/удаленных функциях управляющей программы;
- последняя информация о продуктах компании ELNEC.

Стоимость подписки на услугу Keep–Current необходимо выяснить у торгового агента или в компании ELNEC.



Почему необходимо обновлять программное обеспечение

Производители микросхем и электронных компонентов постоянно пополняют списки выпускаемой продукции, для того чтобы удовлетворить потребности промышленности в применении электронных компонентов в самых различных областях применения. Идя в ногу со временем, компания ELNEC ежегодно пополняет список поддерживаемых микросхем более чем на 500 единиц.

В течение срока выпуска какого-либо типа микросхем, они подвергаются изменениям, направленным на улучшение характеристик. Изменения могут затрагивать алгоритмы программирования. Отказ от обновления программного обеспечения может привести к невозможности работы с микросхемами или к значительному снижению производительности работы с микросхемами.

Как и в любом программном продукте, в управляющей программе могут быть обнаружены ошибки недочеты – компания ELNEC старается их исправить.

Одним из обязательств компании является как можно более скорейшее применение в программном обеспечении поддержки новых микросхем – это особенно удобно для конечного пользователя, так как работа с только что выпущенной производителем микросхемой может быть начата путем обновления программного обеспечения.



Услуга AlgOR

AlgOR – это бесплатная услуга по поддержке пользователей. Она включает в себя доработку и обновление программного обеспечения с целью добавления новых алгоритмов программирования и поддержки новых типов микросхем, еще не включенных в текущую версию управляющей программы.

Процесс получения услуги прост – пользователь посылает в компанию запрос на включение микросхемы в список поддерживаемых устройств. Возможно, по запросу, потребуется предоставить образец микросхемы. После исследования компания предоставит пользователю обновленную версию управляющей программы и возвратит образец микросхемы. Если алгоритм поддержки микросхемы включить в управляющую программу не представляется возможным (черезмерно сложный или коммерческий алгоритм программирования), специалисты компании незамедлительно сообщат об этом и постараются предложить другой, наиболее подходящий способ решения проблемы.

Листок запроса (Русская версия)

Проблема: _____
Дата: _____

Информация о клиенте:

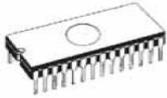
Имя: _____
Адрес: _____
Торговый агент: _____
Дата покупки: _____
Дата отправки рег. карточки: _____

Программатор (тип/название): _____
Серийный номер: _____
Версия управляющей программы: _____

Информация о микросхеме, требующей поддержки:

Полное наименование: _____
Производитель/логотип: _____
Тип корпуса (DIL40, PLCC...): _____

Использование программатора: _____ в день _____ в месяц _____ в год
Количество запрограммированных микросхем на программаторе: _____
К листку запроса приложен образец микросхемы: Да Нет

**Листок запроса (English version)**

Subject (title of problem): _____

Date: _____

Customer, name: _____

Address: _____

Contact person and E-mail: _____

Distributor, name: _____

Date of purchasing: _____

Date of sending registration card: _____

Programmer (type/modification): _____

Serial number: _____

Control program and version: _____

Information about device, you want to be supported

Device type (full name): _____

Vendor/logo: _____

Package (DIL40, PLCC44,...): _____

Precedence rating:

in __ days in __ weeks in __ months

Device to be programmed:

still Y/N sometimes Y/N one-shot Y/N

Number of programmed device:

approx. __ pcs per year.

Samples are available?

Yes Yes (I'm sending it/attached) No

Notes to request. Description of requested change in control program.

Enter please feature you want to the program will have.

**Регистрационная карточка (Русская версия)**

Программатор: _____
Серийный номер: _____
Имя пользователя: _____
Название компании: _____
Направление деятельности: _____
Адрес: _____
Город: _____
Страна: _____
Телефон/Факс/e-mail: _____
Дата покупки: _____
Способ покупки: _____

Ответьте на следующие вопросы:

Какие компьютеры используются в работе?

486 Pentium I - II Pentium III - IV

Какая операционная система установлена?

MS Windows 95/98/Me MS Windows NT/2000/XP

Откуда Вы узнали о компании ELNEC?

Реклама Торговый агент Другое: _____

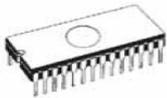
Почему Вы выбрали этот продукт?

Цена Качество Рекомендации

Особенности модели: _____

Ваши комментарии: _____

Замечание: Настоятельно рекомендуется заполнить и отправить одним из доступных для Вас способом регистрационную карточку. Это даст некоторые преимущества при гарантийном обслуживании и предоставлении услуг компанией. Отправляемые данные используются только компанией ELNEC и предназначены для улучшения качества продукции

**Регистрационная карточка (English version)**

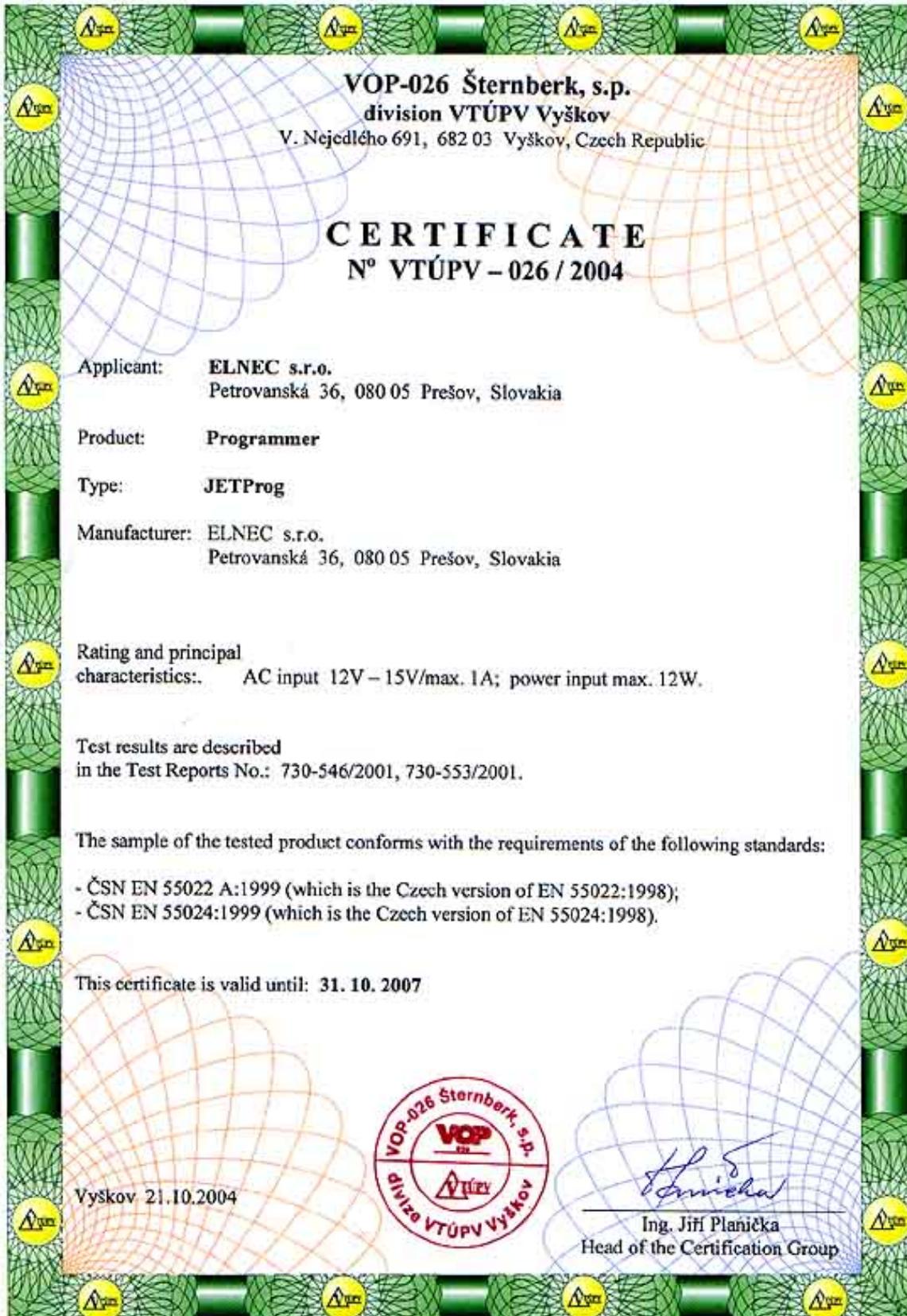
Programmer: _____
Serial number: _____
Name: _____
Company: _____
Department: _____
Address: _____
Post Code, City: _____
Country: _____
Phone/Fax/e-mail: _____
Date of purchase: _____
Purchase from: _____

Complete, please:

- what type of computer are you using:
486 Pentium I-II Pentium III-4
- operating system:
Windows 95/98/Me Windows NT/2000/XP
- how did you first hear about ELNEC ?
advert. dealer other _____
- why did you choose this product?
price quality recommendation
features (please specify) _____
- comments: _____

***Замечание:** Настоятельно рекомендуется заполнить и отправить одним из доступных для Вас способом регистрационную карточку. Это даст некоторые преимущества при гарантийном обслуживании и предоставлении услуг компанией. Отправляемые данные используются только компанией ELNEC и предназначены для улучшения качества продукции*

Сертификаты



VOP-026 Šternberk, s.p.
division VTÚPV Vyškov
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
N° VTÚPV – 026 / 2004

Applicant: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Product: Programmer

Type: JETProg

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: AC input 12V – 15V/max. 1A; power input max. 12W.

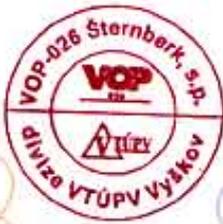
Test results are described
in the Test Reports No.: 730-546/2001, 730-553/2001.

The sample of the tested product conforms with the requirements of the following standards:

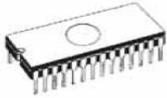
- ČSN EN 55022 A:1999 (which is the Czech version of EN 55022:1998);
- ČSN EN 55024:1999 (which is the Czech version of EN 55024:1998).

This certificate is valid until: **31. 10. 2007**

Vyškov 21.10.2004



J. Planička
Ing. Jiří Planička
Head of the Certification Group



VOP-026 Šternberk, s.p.
division VTÚPV Vyškov
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
N° VTÚPV - 038 / 2003

Applicant: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Product: Programmer

Type: BeeProg

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 15 – 18 VDC / max. 1 A; power input 12 W

Test results are described
in the Test reports No.: 6440-646/2003, 6440-653/2003.

The sample of the tested product conforms with the following standards:

- ČSN EN 55022:1999+A1 (which is the Czech version of the EN 55022:1998+A1);
- ČSN EN 55024:1999 (which is the Czech version of the 55024:1998).

This certificate is valid until: **31. 12. 2006**

Vyškov 04.12.2003




Ing. Jiří Planička
Head of the Certification Group

EC Attestation of Conformity**No.: E8 99 07 36878 002**

ELNEC, spol. s.r.o.
Petrovanska 36, Posta 5, P.O.Box 22

SK - 080 05 PRESOV

Item Identification: Information technology equipment (EMC directive)
Universal 48-pindrive programmer

LabProg-48LV

Item Description: 12VAC
16W

The equipment complies with the principal protection requirements of the EMC Directive (Directive 89/336/EEC relating to electromagnetic compatibility) based on a voluntary test.

This attestation applies only to the particular sample of the product and its technical documentation provided for testing and certification. The detailed test results and all standards used as well as the operation mode are listed in

Test report no. E 015L KO

Test standards: EN 55022/1994, EN 55022/A1:1995, EN 55022/A2:1997, EN 55024/1998

This attestation is released with the above mentioned attestation number by the Certification Body of TÜV PRODUCT SERVICE. It does not permit the use of a TÜV PRODUCT SERVICE certification mark.

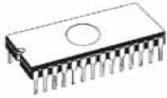
After preparation of the necessary technical documentation as well as the conformity declaration the CE marking as shown below can be affixed on the equipment as stipulated in Article 10.1 of the Directive. Other relevant Directives have to be observed.



Department:
Date:

EMC / STK / KO
July 22, 1999

TÜV PRODUCT SERVICE GMBH - Zertifizierstelle - Ridlerstrasse 31 - D-80339 München



VOP-026 Šternberk, s.p.
division VTÚPV Vyškov
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
N° VTÚPV – 005 / 2005

Applicant: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Product: **Programmer**

Type: **SmartProg2**

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 15V/0,5A.

Test results are described
in the Test Reports No.: 6440-144/2005, 6440-171/2005.

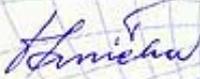
The sample of the tested product conforms with the requirements of the following standards for electromagnetic compatibility:

- EN 55022:1998;
- EN 55024:1998.

This certificate is valid until: **30. 04. 2008**

Vyškov 01.04.2005




Ing. Jiří Planička
Certification Head

VOP-026 Šternberk, s.p.
division VTÚPV Vyškov
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
N° VTÚPV - 039 / 2003

Applicant: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Product: Programmer

Type: MEMprog

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 15 – 20 VDC / max. 0,5 A; power input 5 W

Test results are described
in the Test reports No.: 6440-647/2003, 6440-654/2003.

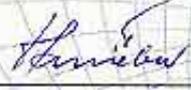
The sample of the tested product conforms with the following standards:

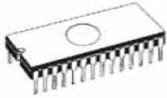
- ČSN EN 55022:1999+A1 (which is the Czech version of the EN 55022:1998+A1);
- ČSN EN 55024:1999 (which is the Czech version of the 55024:1998).

This certificate is valid until: **31. 12. 2006**

Vyškov 04.12.2003




Ing. Jiří Planička
Head of the Certification Group



VOP-026 Šternberk, s.p.
division VTÚPV Vyškov
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
N° VTÚPV – 007 / 2005

Applicant: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Product: Programmer

Type: MEMprogL

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 12V/0,5A.

Test results are described
in the Test Reports No.: 6440-247/2005, 6440-268/2005.

The sample of the tested product conforms with the requirements of the following standards for electromagnetic compatibility:

- EN 55022:1998;
- EN 55024:1998.

This certificate is valid until: **30. 04. 2008**

Vyškov 28.04.2005




Ing. Jiří Planička
Certification Head

VOP-026 Šternberk, s.p.
division VTÚPV Vyškov
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
N° VTÚPV – 027 / 2004

Applicant: **ELNEC s.r.o.**
Petrovanská 36, 080 05 Prešov, Slovakia

Product: **Programmer**

Type: **TS1Prog**

Manufacturer: **ELNEC s.r.o.**
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 15V – 20V/max. 500mA; power input max. 5W.

Test results are described in the Test Reports No.: 730-545/2001, 730-552/2001.

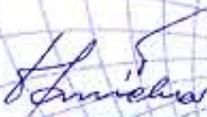
The sample of the tested product conforms with the requirements of the following standards:

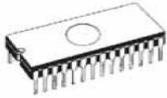
- ČSN EN 55022 A:1999 (which is the Czech version of EN 55022:1998);
- ČSN EN 55024:1999 (which is the Czech version of EN 55024:1998).

This certificate is valid until: **31. 10. 2007**

Vyškov 21.10.2004




Ing. Jiří Planička
Head of the Certification Group



Military Technical Institute of the Ground Forces
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
N° VTÚPV - 016 / 2002

Applicant: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Product: Programmer

Type: PIKprog +

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 12V / 500 mA; power input max. 5W

Test results are described
in the Test reports No.: 730-385/2002, 730-388/2002.

The sample of the tested product conforms with the following standards:

- ČSN EN 55022:1999+A1 (which is identical with EN 55022:1998+A1);
- ČSN EN 55024:1999 (which is identical with EN 55024:1998).

This certificate is valid until: 31. 07. 2005

Vyškov - 18.07.2002




Ing. Jiří Planička
Head of the Certification Group

Military Technical Institute of Ground Forces
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
No. VTÚPV – 021 / 2003

Applicant: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Product: Programmer

Type: PIKprog

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 12 V/0,5 A; power input 3 W.

Test results are described
in the Test Report No.: 730-399/2000, 730-404/2000.

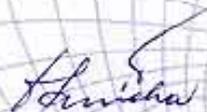
The sample of the tested product conforms with the requirements of the following standards:

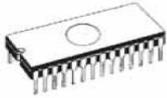
- ČSN EN 55022:1996 (which is identical with EN 55022:1994);
- ČSN EN 61000-4-2:1997 (which is identical with EN 61000-4-2:1995);
- ČSN EN 61000-4-3:1997 (which is identical with EN 61000-4-3:1996).

This certificate is valid until: **31. 05. 2006**

Vyškov 20.05.2003




Ing. Jiří Planička
Head of the Certification Group



Military Technical Institute of Ground Forces
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
No. VTÚPV – 022 / 2003

Applicant: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Product: Programmer

Type: SEEprog

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 12 V/0,5 A; power input 3 W.

Test results are described
in the Test Report No.: 730-400/2000, 730-405/2000.

The sample of the tested product conforms with the requirements of the following standards:

- ČSN EN 55022:1996 (which is identical with EN 55022:1994);
- ČSN EN 61000-4-2:1997 (which is identical with EN 61000-4-2:1995);
- ČSN EN 61000-4-3:1997 (which is identical with EN 61000-4-3:1996).

This certificate is valid until: **31. 05. 2006**

Vyškov 20.05.2003


Ing. Jiří Planička
Head of the Certification Group