PPM3 MK2

Production ISP Programmer

User Guide

EQUINOX
TECHNOLOGIES
EQUINOX TECHNOLOGIES UK LIMITED

The Embedded Solutions Company

PPM3 Programmer Module
High-end Programming Module

Contents

Copyright Information

Information in this document is subject to change without notice and does not represent a commitment on the part of the manufacturer. The software described in this document is furnished under license agreement or nondisclosure agreement and may be used or copied only in accordance with the terms of the agreement.

It is against the law to copy the software on any medium except as specifically allowed in the license or nondisclosure agreement.

The purchaser may make one copy of the software for backup purposes. No part of this manual may be reproduced or transmitted in any form or by any means, electronic, mechanical, including photocopying, recording, or information retrieval systems, for any purpose other than for the purchaser's personal use, without written permission.

Equinox guarantees that its products will be free from defects of material and workmanship under normal use and service, and these products will perform to current specifications in accordance with, and subject to, the Company's standard warranty which is detailed in Equinox's Purchase Order Acknowledgment.

Equinox Warranty Information

This product is guaranteed by Equinox Technologies UK Limited for a period of 12 months (1 year) after the date of purchase against defects due to faulty workmanship or materials. One guarantee covers both parts and labour. Service under the guarantee is only provided upon presentation of reasonable evidence that the date of the claim is within the guarantee period (e.g. completed registration/guarantee card or a purchase receipt).

The guarantee is not valid if the defect is due to accidental damage, misuse or neglect and in the case of alterations or repair carried out by unauthorised persons. A number of exceptions to the warranty are listed in the 'Exceptions to warranty' section below. Service (during and after guarantee period) is available in all countries where the product is distributed by Equinox Technologies UK Limited.

Exceptions to warranty

i.   **Over-voltage damage**
     This warranty does not cover damage to the programmer due to voltages beyond the specified voltage limits being applied to the 'DC Power Input' (CON1) or any of the ISP Headers. The user must ensure that sufficient care is taken to avoid over-voltage and static conditions on any of the 'ISP Header' I/O pins.

ii. **Over-current damage**
     This warranty does not cover damage to the programmer due to excessive current being drawn from the programmer power supply. The user must ensure that there is sufficient over-current protection within the test fixture to protect against short circuit loads.

iii. **Short-circuit damage**
     This warranty does not cover damage to the programmer due to short-circuit loads being placed across programmer I/O lines.

Warning!
Any damage caused to the programmer by Electrostatic Discharge (ESD) through inadequate earthing is not covered under the warranty of the product.

Disclaimer

Whilst every effort has been made to ensure that programming algorithms are correct at the time of their release, it is always possible that programming problems may be encountered, especially when new devices and their associated algorithms are initially released. It is Equinox's Company Policy to endeavour to rectify any programming issues as quickly as possible after a validated fault report is received.

It is recommended that high-volume users always validate that a sample of a devices has been programmed correctly, before programming a large batch. Equinox Technologies UK Ltd. can not be held responsible for any third party claims which arise out of the use of this programmer including 'consequential loss' and 'loss of profit'.

Equinox Technologies UK Ltd. cannot be held responsible for any programming problems which are 'out of our control'. This type of problem is usually listed in the 'Errata Sheet' for the particular device being programmed and is available from the silicon vendor.

Information contained in this manual is for guidance purposes only and is subject to change. E&OE.

Electromagnetic Compatibility (EMC) Compliance

The **'PPM3 MK2 Programmer'** is a CE Approved Product. It is designed for use in an ESD controlled environment i.e. development or production. This means, therefore, that the user must ensure that there is no possibility of damage from electrostatic discharge (ESD). Since the devices and equipment to which this product is likely to be connected may well themselves be susceptible to ESD, this should not pose any difficulty.

For example, if you are handling microcontrollers and EEPROMS etc. then you will already be used to appropriate precautions, such as the use of anti-static mats, wrist straps and so on. You should treat your **'PPM3 MK2 Programmer'** with the same care as you would these types of devices. Always ensure that you are not yourself carrying a static charge before handling the product. Wearing an earthed anti-static wrist strap is recommended.

Equinox have taken great care in designing this product to be compliant with the European EMC directive. When using the equipment be sure to follow the instructions provided. Although RF emissions are within prescribed limits, care should be taken if you are using the product near to sensitive apparatus. If you experience any difficulty please refer to Equinox Technical Support.

ESD Points to remember

- Work in a static-free environment.
- Wear an earthed wrist strap when handling either the programmer and/or any programmable device.
- Ensure that the PC, programmer and Target system are connected to the same EARTH (0V) potential.
- Do NOT plug the ISP cable of the programmer into a Target System when the Target power is ON.

Warning!
Any damage caused to the programmer by Electrostatic Discharge (ESD) through inadequate earthing is not covered under the warranty of the product.

Technical Support

It is often the case that users experience problems when installing or using a product for the first time.

If you have a technical support problem, please consult the following list for help:

► **Manual**

► **On-line help**

Press <F1> for help at any time when running EQTools or ISP-PRO.

The help system is context-sensitive. Simply press <F1> on any error message and the Possible causes of the error should be listed. This help system is updated on a regular basis. Please see software update details for information on keeping up-to-date with software revisions.

► **Internet Web Site**

The support page for all Equinox ISP Programmers can be found at:
http://www.equinox-tech.com/products/downloadsearch.asp

► **E-mail**

Please e-mail any technical support questions about this product to:
support@equinox-tech.com

► **Fax**

Please fax any technical support questions about this product to: +44 (0) 1942 844181

Equinox will try our best to answer your questions about this product as quickly as possible. However, we cannot promise an immediate reply. Please consult our web site for new software updates as the problem that you are enquiring about may have already been fixed in a new version.

Product Documentation

This manual provides an overview of the contents of the PPM3 Programming System plus associated hardware and software. References may be made to other hardware and software products which are not covered in detail in this manual. Please refer to the table below for a list of sources of documentation and/or browse to http://www.equinox-tech.com/products/downloadsearch.asp

Software:	
	EQTools Script Builder – Manual This software is used to create and upload 'Programming Projects' to the programmer. The following sources of documentation are available for this software: • Installation and Getting Started Guide (pdf manual) • Help file
	ASCII Text Communications Protocol – Application Note This protocol can be used to control the programmer from an external controller via RS-232. The following sources of documentation are available for this protocol: • Application Note – AN110 for a full description.
	ISP Pro – Manual This software is used to control the programmer in a production environment. It is not supplied as standard with this programmer. The following sources of documentation are available for this software: • Installation and User Manual • Help File
	Upload Wizard - Standalone Project Upload Utility This software utility is used to upload Programming Projects to any Equinox programmer. These projects can then be used in Standalone Mode, i.e. without a PC. • Please follow the on-screen instructions within the Upload Wizard utility itself.

CONFIGIT	**Configit – Firmware Upgrade Utility** This utility is used to upgrade the firmware of the programmer if the firmware version is < 3.00. A firmware update may be required to add support for new devices and to correct any firmware issues.
LabVIEW®	**Labview – Remote Application Control – Application Note** This upgrade allows a production facility to control a single programmer from a 'Labview for Windows' application. The Application note describes how to control the programmer using a custom Labview (from National Instruments) application. The following sources of documentation are available for this software: • Application Note - Remote Application Control of ISP-PRO
REMOTE APPLICATION Control	**Remote Application Control – Application Note** Describes how to control the programmer using a custom Remote Application written in e.g. Visual Basic, C++, C Builder, Delphi etc.
ConsoleEDS	**ConsoleEDS Pro – Application note** This software utility allows any Equinox programmer to be controlled via simple Command Line instructions from a Command Window within Windows. The following sources of documentation are available for this software: • Application Note 111 – ConsoleEDS Manual
JTAG ATmega ISP	**JTAG In-System (ISP) Upgrade – Application Note** This license upgrade enables the PPM3 Programmer to support high-speed In-System Programming (ISP) of the Atmel ATmega microcontroller family using the JTAG algorithm. Support is offered for both single and multiple JTAG devices in a JTAG Chain. Please refer to the following application note for further information: • Application Note 101
ERROR MESSAGES	**Error Message Descriptions** This document lists all the possible error messages which can be generated by the EQTools / ISP-PRO applications.

Downloading up-to-date documentation and software:

In line with our policy of continuous improvement, the software and associated documentation for this product are updated on a regular basis. For the most up-to-date software, firmware and documentation, please refer to the **<Downloads>** page on our website.

1.0 Programmer Overview

1.1 System Contents

The PPM3-MK2 Module is supplied as a Programmer Module along with two plug-in I/O Connector Modules, cables and software. Please see the full contents list detailed below.
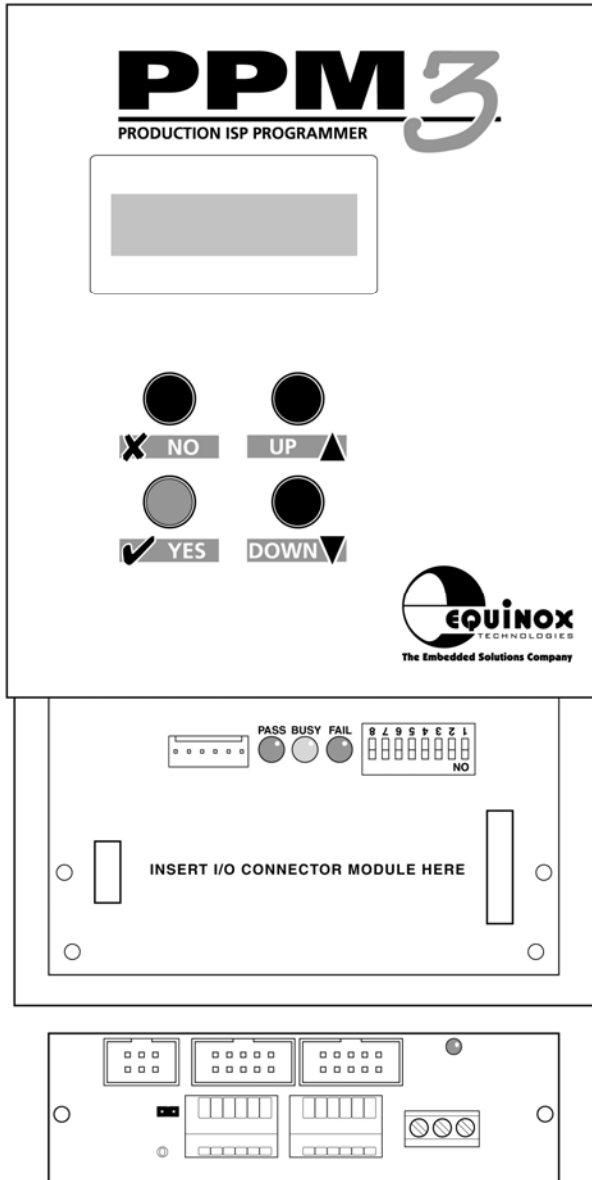
Hardware

- PPM3 MKII Programmer – including SFM V1.0 Module (fitted internally)
- EQ-IOCON-1 – I/O Connection Module
- EQ-IOCON-2 – I/O Connection Module
- External Power Supply Adaptor (9V DC regulated @ 1500 mA)

Cables

- DC Power Cable (jack to bare wires)
- 9-way Female to 9-way Male Serial Cable
- 25-way Female to 9-way Male Serial Adaptor
- 10 way ISP Cable
- 6 way ISP Cable

Miscellaneous

- 4 x Rubber Feet
- 1 x Jumper Link for re-flashing programmer

Software
(Supplied on an Equinox CD-ROM)

- EQTools
  (Project Management Utility for Equinox Production ISP Programmers)

Documentation

- PPM3 MK2 - User Guide

1.2 Hardware Overview (external layout)
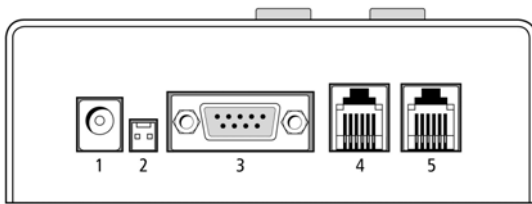
Rear Panel
1. DC Power Input - Jack Socket
2. DC Power Input - Molex Connector
3. RS-232 Communications Port
4. RS-485 Communications Port (IN)
5. RS-485 Communications Port (OUT)
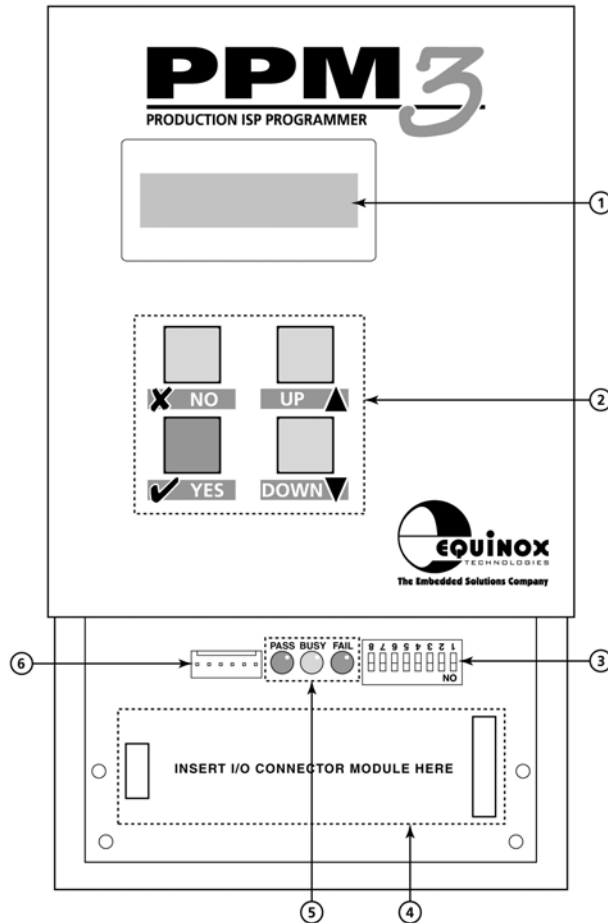
Front Panel
1. Liquid Crystal Display (LCD) – 16 x2 characters

2. Keypad (4 button)

3. Programmer Configuration DIP Switches

4. I/O Connector Module interface

5. Programmer Status LED's

6. 'Remote System Status' LED port (6-way Square Pin Molex Connector)

1.3 I/O Connector Modules

1.3.1 Overview

Equinox has developed a range of plug-in user-replaceable 'I/O Connector Modules' which are designed to interface the Equinox 'PPM3 Programming Module' to a variety of different Target Systems. This interchangeable approach allows the required connection system to be tailored to a particular Target System or Test Fixture without the requirement for unwiring the Test Fixture.

The required I/O Connector Modules simply plugs into the mating connectors at the end of the programmer as detailed in the figure below.

Fig. 1.3.1 PPM3 – Inserting the I/O Connector Module
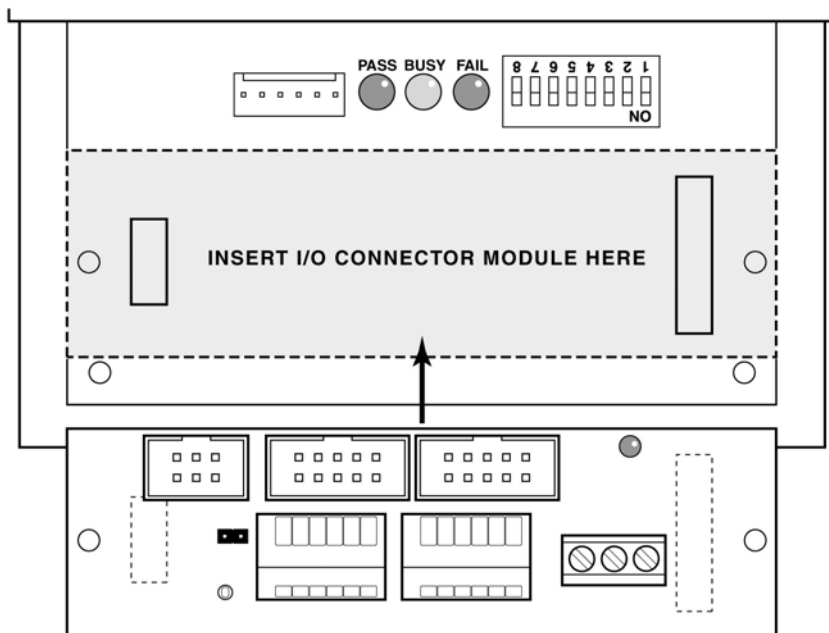
The currently available 'I/O Connector Modules' are as follows:
- EQ-IOCON-1 (supplied as standard)
- EQ-IOCON-2 (supplied as standard)
- EQ-IOCON-3 (must be purchased separately)
- EQ-CALCON (must be purchased separately)

Please refer to the separate manual 'PPM3 – I/O Connector Modules' for a full description of all the available modules.

PPM3 Programmer Module
High-end Programming Module

Features:

- Plug-in 'I/O Connector Modules' – simply plugs into the mating sockets on the PPM3 Module
- Different connector methods for interfacing to the Target System can be implemented by simply plugging in the relevant 'I/O Connector Module'.
- Programmer or Test Fixture can be easily swapped out without the requirement for unwiring the Test Fixture
- One programmer can easily be used with multiple Test Fixtures
- New connection systems can be implemented using a new 'I/O Connector Module' without requiring any redesign of the PPM3 Module.
- IO-CON-2 Module supports 'wire-wrap' connections
- IO-CON-1 / -2 support clip-in wire 'Fast Connect' wiring system
- Target Vcc LED - illuminates when the 'Target Vcc' is detected
- Auxiliary Power available via screw-terminal connections

1.3.2 Connector Systems currently supported

The following ISP connector systems are currently supported

- Atmel 10-way ISP Header (SPI)
- Atmel 6-way ISP Header (SPI)
- Atmel 10-way ISP Header (JTAG)
- Equinox 10-way ISP Header for Atmel T89C51Rx2 microcontrollers
- Equinox 10-way ISP Header for Philips P89C51Rx2 and P89C66x microcontrollers
- Wire-wrap (all devices)
- Fast-Connect - plug-in wired connectors (all devices)
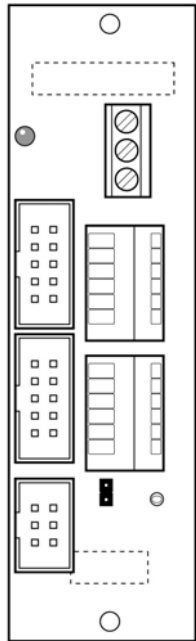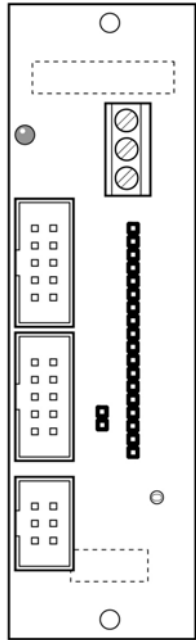
1.3.3 Programmers supported

The I/O Connector Modules are compatible with the following Equinox programmers:

- PPM3 MK1 – Production ISP Module Version 1
- PPM3 MK2 – Production ISP Module Version 2

1.3.4 Overview of I/O Connector Modules

There are currently four different I/O Connector Modules available as detailed in the table below:

Order Code	Description
EQ-IOCON-1 	**I/O Connector Module 1 (SPI) – Fast Connect Version** I/O connector module for In-System Programming (ISP) of Atmel microcontrollers using SPI protocol (Fast Connect version) This module plugs into the PPM3 and provides the relevant I/O connectors for SPI programming of Atmel devices. **Features:** • Plugs into suitable Equinox programmer eg. PPM3 Module • Atmel 10-way IDC ISP Header • Atmel 6-way IDC ISP Header • Equinox 10-way IDC ISP header • 'Fast Connect' clip-in connectors for direct wiring to Test Fixture i.e wires from bed-of-nails probe wires • Screw terminals for power connections • Target Vcc Status LED • Link to connect / isolate the programmer Vcc from the Target Vcc
EQ-IOCON-2 	**I/O Connector Module 2 (SPI) – Wire-wrap version** I/O connector module for In-System Programming (ISP) of Atmel microcontrollers using SPI protocol (wire-wrap version) **Features:** • Plugs into suitable Equinox programmer e.g. PPM3 Module • Atmel 10-way IDC ISP Header • Atmel 6-way IDC ISP Header • Equinox 10-way IDC ISP header • Single-in-line header with all programmer I/O brought out for wire-wrapping to bed-of-nails probe wires • Screw terminals for power connections • Target Vcc Status LED • Link to connect / isolate the programmer Vcc from the Target Vcc

PPM3 Programmer Module
High-end Programming Module

EQ-IOCON-3	**I/O Connector Module 3 (JTAG) – Fast Connect Version** I/O connector module for In-System Programming (ISP) of Atmel microcontrollers using JTAG protocol **Features:** • Plugs into suitable Equinox programmer e.g. PPM3 Module • Atmel 10-way JTAG IDC ISP connector (same as JTAG-ICE) • Atmel 6-way IDC ISP Header • Equinox 10-way IDC ISP header • Single-in-line header with all programmer I/O brought out for wire-wrapping to bed-of-nails probe wires • Screw terminals for power connections • Target Vcc Status LED • Link to connect / isolate the programmer Vcc from the Target Vcc **Please note** The 'Atmel AVR JTAG License' (Order code: PPM3A1-UPG7) is also required to enable the PPM3 to program Atmel AVR devices via JTAG.
EQ-CALCON	**EQ-CALCON Module** The EQ-CALCON Module is a special plug-in I/O Connector Module for the PPM3-MK2 programmer which is used to facilitate very accurate calibration of the Internal Oscillator of an Atmel ATmega or ATtiny AVR microcontroller. The module features an on-board high accuracy 32kHz Oscillator Module which can be connected to the target AVR microcontroller MOSI pin during the Oscillator Calibration Procedure as described in the Atmel Application Note 'AVR053: Calibration of the Internal Oscillator'. **Features:** • Plugs into suitable Equinox programmer e.g. PPM3 Module • Atmel 10-way JTAG IDC ISP connector (same as JTAG-ICE) • Atmel 6-way IDC ISP Header • Equinox 10-way IDC ISP header • Screw terminals for power connections • Target Vcc Status LED • Link to connect / isolate the programmer Vcc from the Target Vcc **Please note** The use of this module requires development of / purchase of suitable 'Calibration Firmware' for the chosen ATmega AVR microcontroller.
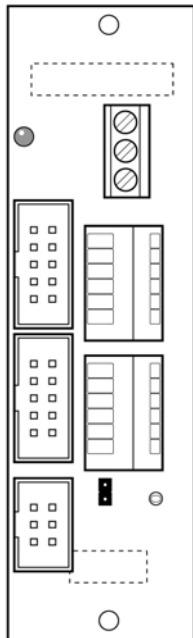
1.4 Special Function Modules (SFM)

1.4.1 SFM Module - Overview

The PPM3-MK2 programmer features an interchangeable *'Special Function Module'*. This module contains all the *'I/O Driver Circuitry'* which interfaces the programmer to the Target System. In the event of damage to the programmer driver circuitry, it is possible to remove the complete Special Function Module and replace it with a new one. This allows the programmer to be repaired in-situ without the need to send it back to Equinox thereby saving valuable lost production time.

- The PPM3-MK2 is fitted with the 'EQ-SFM-1' module as standard.
- There are also other SFM modules available which offer faster SPI / JTAG speeds and better ESD / Over-voltage protection.

1.4.2 SFM Module – Selection Guide

The SFM Modules which are currently available are listed in the table below:

Order Code	Description
EQ-SFM-1	**Standard Special Function Module V1.0** • Fitted to all PPM3-MK2 programmers as standard • Supports JTAG speeds up to approx 900 kHz • Supports SPI speeds up to approx 500 kHz • Limited ESD and over-voltage protection • Not recommended for driving capacitive or inductive loads • Not recommended if the Target System power is not supplied / controlled from the programmer.
EQ-SFM-MAX-1.2	**High-speed High Current SFM Module V1.2** • Supports JTAG speeds up to approx 1.3 MHz • Supports SPI speeds up to approx 4 MHz • Enhanced ESD protection • Enhanced over-voltage / fault protection • Safe to use when Target System is independently powered • Better at driving capacitive or inductive loads • Requires programmer firmware 3.04 or above
EQ-SFM-MAX-1.3	**High-speed High Current SFM Module with 32kHz Oscillator V1.3** • Supports JTAG speeds up to approx 1.3 MHz • Supports SPI speeds up to approx 4 MHz • Integrated 32 kHz Oscillator for performing accurate calibration of Atmel AVR on-chip oscillators (replaces the CALCON module) • Enhanced ESD protection • Enhanced over-voltage / fault protection • Safe to use when Target System is independently powered • Better at driving capacitive or inductive loads • Requires programmer firmware 3.04 or above

1.4.3 SFM Module – how to fit a new module

To fit a new SFM Module to the programmer:
- Make sure the programmer is powered off
- Remove all cables from the programmer
- Remove the four fixing screws in the side of the programmer case
- Carefully slide the lid of the programmer upwards and then off
- You can find the existing SFM Module underneath the keypad PCB
- Carefully remove the existing SFM Module by applying pressure from underneath at both sides of the circuit board → the SFM module should pop out of the red connectors
- Fit the new SFM Module in the same position and orientation as the one you have just take out (make sure to line up the long red connector on the main PCB to the long red connector on the SFM module
- Push the new SFM Module down gently until it locks into place
- Replace the programmer lid
- Replace the programmer screws

1.4.4 SFM Module – declaring the module in a Programming Project

The EQ-SFM-1 and EQ-SFM-V1.2 / V1.3 modules actually use slightly different firmware in the programmer. It is therefore necessary to specify which SFM Module is fitted in each Programming Project.

To specify the correct SFM Module in a Programming Project:

In EDS (Development Mode):
- Open your EDS (*.eds) development project
- Select the *<Programmer and device>* tab
- Locate the *'Special Function Module fitted'* list box
- Select the relevant SFM Module which you have fitted to your programmer

In Project Builder:
- Open your Programming Project (*.ppm) project
- Select the *<Programmer and Project Type>* tab
- Locate the 'Special Function Module fitted' list box
- Select the relevant SFM Module which you have fitted to your programmer
- Re-compile your project to save the changes

Important notes:
- Failure to specify the correct SFM Module can result in erratic operation of the programmer.
- If you have existing projects which use the standard EQ-SFM module, these projects will need to be recompiled for the new EQ-SFM-MAX 1.2 or 1.3 Modules.

1.5 Programmer dimensions and mounting holes

The PPM3 programmer can be mounted to a Test Fixture by using the two mounting holes provided. The mounting holes are threaded to accept an M3 bolt. The bolt thread must not penetrate more than 6mm into the box otherwise it may short onto the circuit board.

Fig. 1.5 Programmer dimensions and mounting holes
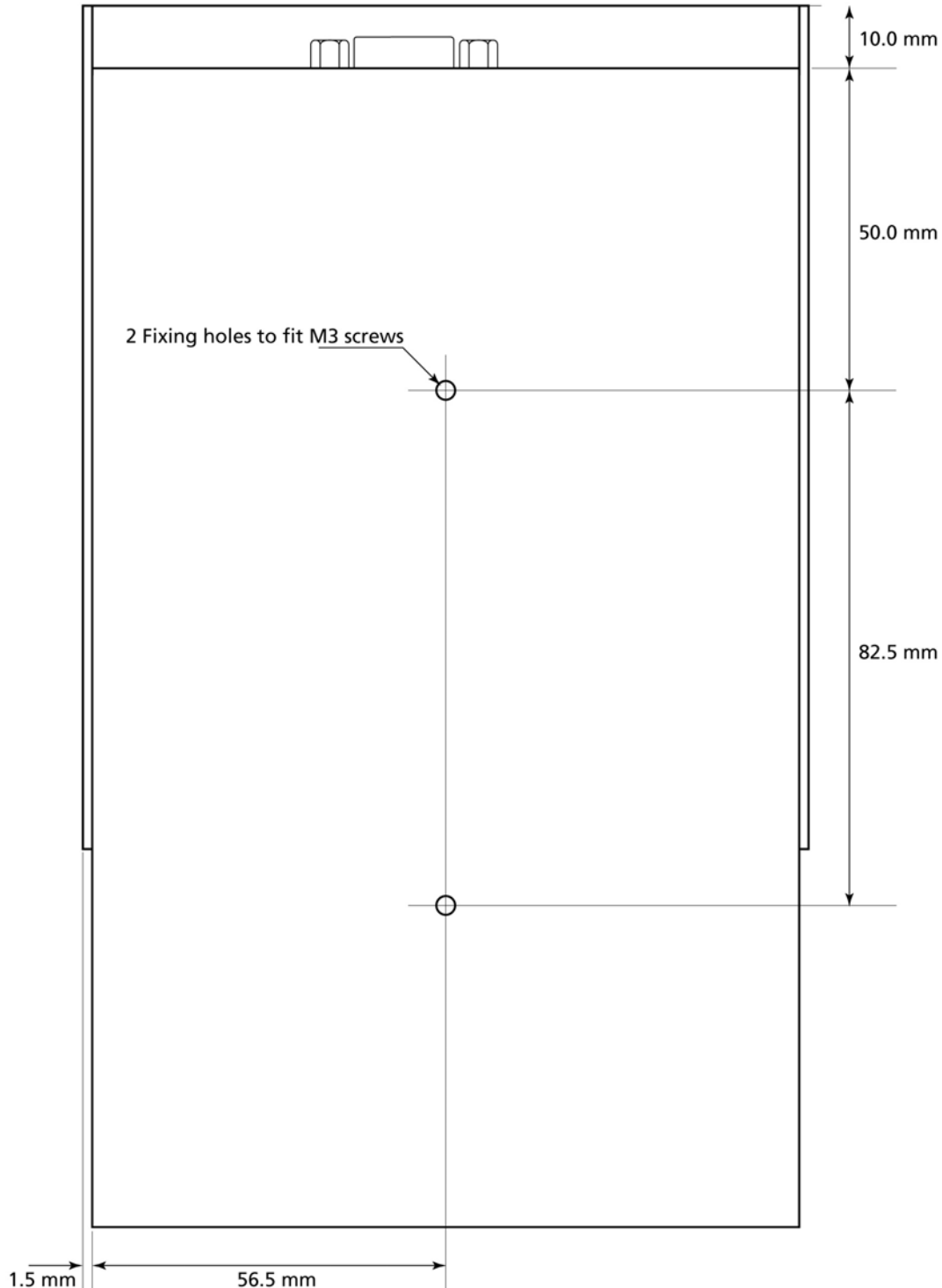
1.6 Programmer Specifications

The table below details the hardware specifications for the programmer. Please refer to the stated section for further information about any specific parameter.

#	Parameter	Description / comment	Refer to section
1	**Target Device Support**	See Device Support List	1.7.1
2	**Target Device Programming Interfaces**	The programmer supports the following programming interfaces: • Atmel Low Voltage SPI • Atmel High Voltage SPI (+12V Vpp) • Atmel AVR JTAG (chargeable update) • Atmel (formerly Temic) 8051 Boot loader ISP • Philips (NXP) 8051 - Boot Loader ISP • Generic I2C (SDA + SCL)	1.7.2
3	**Operating modes**	The programmer supports the following operating modes: • Development Mode (PC controlled) • Project Upload Mode (PC controlled) • Standalone Mode - Keypad + LCD operation • Standalone Mode – Automatic Target Connect / Disconnect Sensing • Standalone Mode - Remote Controlled via 'ASCII Text Communications' RS-232 Serial Protocol • Standalone Mode - Remote Controlled via 4-wire TTL Remote Control Port • PC controlled via ISP-PRO software (chargeable upgrade) • PC controlled via ConsoleEDS utility (chargeable upgrade)	2.4
4	**On-board FLASH Memory Store**	32 Mbits (4 MBytes) FLASH Memory	
5	**Project storage in Memory Store**	64 x Programming Projects	
6	**Keypad entry**	4 x Push Button - <YES>, <NO>, <UP>, <Down>	
7	**Status LED's**	i. 3 x Status LED's located on the programmer: <PASS>, <BUSY>, <FAIL> ii. 3 x remote Status LED's can also be connected off-board on a Test Fixture via the 'Remote System Status' port.	
8	**Programmer Display**	2 x 16 Back-lit LCD	
9	**Target Vcc Voltage Detection LED**	This LED is located on the I/O-CON Module: • Target Vcc (POWER) LED will light when Target Vcc is detected. • Threshold detection voltage: approx 3.1V	

10	PC Control Software	The programmer can be controlled using: • EQTools (as standard) • ASCII Text Communications Mode (as standard) • ISP-PRO (chargeable upgrade)	2.3
11	EQTools / ISP-PRO PC requirements	PC running Windows 95 / 98 / 2000 / ME / NT4 / XP	2.3
12	Power Supply Options	There are 3 possible modes for powering the Programmer / Target System: • Programmer controls power to the Target System • Programmer and Target System are independently powered • Target System powers the Programmer (+5V only)	4.0
13	Voltage range	• Device Programming voltage range: 3.0 – 5.0V • Programmer Controlled Power Supply range: 3.0 – 5.0V • Possible Target System voltage range: 3.0 – 5.0V	4.4
14	Vpp Voltage Generator	On Board +12.0V Vpp Voltage Generator (Applies +12.0V to RESET pin for Atmel ATtiny HV programming)	
15	DC Power Connector (jack socket)	DC Power Connector • 2.5 mm Jack Socket – Centre Positive • 9.0 – 12.0V DC regulated @ 500mA (worst case) • Mates with Power Supply Adaptor supplied with programmer • DC Power Lead also supplied with programmer if a customer power supply is to be used.	4.2
16	DC Power Connector (Molex)	DC Power Connector • 2-pin pluggable Molex Connector • 9.0 – 12.0V DC regulated @ 500mA (worst case)	
17	Max. current through programmer	This refers to the maximum current which the programmer can supply from the 'controlled' power supply. Max current = 300 mA.	
18	Target SPI Frequency	**Software SPI algorithm:** • SLOW SPI: 0 – 490.2 kHz (user selectable) • MEDIUM SPI: 0 – 490.2 kHz (user selectable) (SPI speeds are estimations only due to uneven mark/space ratio and non-continuous waveforms) **Hardware SPI algorithm (requires firmware 3.04 or greater):** • SLOW SPI: 115.2 kHz to 3.6864 MHz (user selectable) • FAST SPI: 115.2 kHz to 3.6864 MHz (user selectable) • SPI speeds > 500kHz require the fitting of the faster EQ-SFM-MAX V1.2 or V1.3 SFM Modules	

19	Target JTAG Frequency	**With standard 'SFM Module' fitted:** The programmer supports a user-defined JTAG frequency: • SLOW JTAG: 0 – 833 kHz **With 'SFM-MAX V1.2 or V1.3' Module fitted:** • SLOW JTAG: 0 – 833 kHz • FAST JTAG: approx 1.4 MHz	
20	JTAG Programming Modes	**The programmer supports the following JTAG modes:** • Single JTAG device connected to programmer • Multiple JTAG devices connected in a JTAG Chain • JTAG ID can be verified for both Atmel AVR and non-Atmel JTAG devices • JTAG chain validation is also supported	
21	I2C Frequency	The programmer supports a user-defined I2C frequency: • SLOW I2C: 0 – 400 kHz • Supports ISP programming of both 100 kHz and 400 kHz I2C Serial EEPROMs	
22	Target UART Speed	• This is the Communications Speed (BAUD rate) from the programmer to the Target Device when communicating via a Boot Loader. • The BAUD rate is configurable from 1,200 to 115 kBaud. • This settings is completely independent from the BAUD rate settings for the PC communicating with the programmer.	
23	ISP Headers	The programmer can support any ISP header by simply plugging in the relevant 'EQ-IOCON' I/O Connector Module. The currently supported ISP Headers are as follows: • Atmel 10-way (SPI) • Atmel 6-way (SPI) • Atmel 10-way (JTAG) • Equinox 10-way (SPI+UART) • Fast-connect wire connectors Please refer to the separate manual 'PPM3 I/O Connector Modules' for full details of each module and the connection mechanisms available.	
24	Frequency generator output (SCK2)	Programmer can output a configurable square wave frequency on the SCK2 pin. This is used to clock the Target Device in the absence of a Target Oscillator. • Frequency range: 167kHz – 1.8432 MHz	
25	AVR On-chip Oscillator Calibration	If the programmer is fitted with either the 'CALCON' or 'EQ-SFM-V1.3' module, then it can be used to calibrate the internal	

		oscillator of an Atmel AVR microcontroller. See 'Special Function Modules' application note for further details.	
25	**PC Connection**	**i. RS232 – Single Programmer Channel Control** • 9-way Female D Connector • Connects to spare PC COM port • All 9 connections must be made in serial cable. • Baud rate: 38,400 (fixed) **ii. RS-485 Multiple Programmer Channel Control** • Programmers are daisy chained via RS-485 leads • An RS-232 to RS-485 converter is required at the PC end of the chain.	3.0
26	**Temperature range**	0 to 50 deg. C	
27	**Dimensions**	Actual Programmer Module: 185mm long x 114mm wide x 45mm high	
28	**Shipped Weight**	2.0 kg (dead weight – including accessories)	

PPM3 Programmer Module
High-end Programming Module

EQUINOX
The Embedded Solutions Company

1.7 Device Support

1.7.1 Devices supported by the programmer

The programmer supports the following devices:

S – Device supported as standard
U – Chargeable license upgrade required

#	Silicon vendor	Family	Programming algorithm	Device support	
1	**Atmel**	AT89S	LV SPI	AT89(L)S 8252/53 AT89(L)S 51/52 (including 'L' versions) AT89S 2051 / 4051	S
2	**Atmel**	AT90S (AVR)	LV SPI	AT90(L)S 1200(A) / 2313 / 2323 / 2333 / 2343 / 4414 / 4433 / 4434 / 8515 / 8535 (including 'L' versions)	S
3	**Atmel**	AT90CAN	LV SPI	AT90CAN32 / AT90CAN64 / AT90CAN128	S
4	**Atmel**	AT90CAN	JTAG (AVR)	AT90CAN32 / AT90CAN64 / AT90CAN128	U
5	**Atmel**	AT90USB	LV SPI	AT90USB1286 / AT90USB1287 / AT90USB162 / AT90USB646 / AT90USB647 / AT90USB82	S
6	**Atmel**	AT90USB	JTAG (AVR)	AT90USB1286 / AT90USB1287 / AT90USB646 / AT90USB647	U
7	**Atmel**	AT90PWM	LV SPI	AT90PMW 2/3 AT90 PWM 3	S
8	**Atmel**	ATmega (AVR)	LV SPI	ATmega 8 / 16 / 32 / 48(V) / 64/ 88(V) / 103 / 128 / 1280(V) / 1281(V) /161 / 162 / 163 /165 / 168 / 169(V) / 169PV / 2560(V) / 2561(V) / 323 / 324PV / 325 / 3250 (V) / 328P(V) / 329 / 3290(V) / 640(V) / 644(V) / 645 / 6450(V) / 649(V) / 6490 (V) 8515 / 8535 (including 'L' versions)	S
9	**Atmel**	ATmega (AVR)	JTAG (AVR)	ATmega 16 / 32 / 64 / 128 / 1280 / 1281(V) / 162 / 169(V) / 2560(V) / 2561(V) / 323 / 324P(V) / 325(V) / 3250(V) / 329 / 3290(V) / 406 / 640(V) /644(V) / 645 / 6450(V) / 649(V) / 6490(V) (including 'L' versions)	U
10	**Atmel**	ATmega PICO (AVR)	LV SPI	ATmega 164P(V) / 168P(V) / 169P(V) / 324P / 329P(V) / 3290P(V) / 48P(V) / 644P(V) / 88P(V)	S
11	**Atmel**	ATmega PICO (AVR)	JTAG (AVR)	ATmega 164P(V) / 169P(V) / 324P / 329P(V) / 3290P(V) / 644P(V)	U
12	**Atmel**	ATtiny (AVR)	LV SPI	ATtiny 11 / 12(V) / 13 / 22 / 24(V) / 25(V) / 26 / 44(V) / 45(V) / 84(V) / 85(V) / 2313 (including 'L' versions)	S

13	**Atmel**	ATtiny (AVR)	HV SPI (+12V VPP)	ATtiny 11 / 12 / 15L (including 'L' versions – Vcc must be +5V with a +12V Vpp applied to the RESET pin) HV ISP support for the ATtiny13, 25, 45 etc is not currently available.	S
14	**Atmel (Temic)**	T89C51 Rx2 AC2 Ix2 CC01	UART Boot Loader	T89C51 RB2 / RC2 / RD2 T89C51 AC2 T89C51 IB2 / IC2 T89C51CC01 (Support for the AT89Cxxxx devices is not currently available)	S
15	**Atmel**	AT24Cxxx	I2C	AT24C 01A / 01B / 11 / 02A / 02B / 04(A/B) / 08(A/B) / 16(A/B) / 32(A/C/SC) / 64(A/B) / 128(B/SC) / 256B / 512(B/SC) / 1024(B/SC)	U
16	**Atmel**	AT24HCxxx	I2C	AT24HC 02B / 04B	U
17	**Atmel**	AT34Cxxx	I2C	AT34C 02C	U
18	**Philips**	P89C 51Rx2 66x	UART Boot Loader	P89C51 RB2 / RC2 / RD2 P89C 660 / 662 / 664 / 668	S
19	**Zensys**	ZW series	LV SPI	ZW0102 / ZW0201 / ZW0301	S

Please note:

1. JTAG algorithms
- The ATmega AVR JTAG programming algorithms require a chargeable 'JTAG Upgrade License' to be purchased.
- Firmware 3.07 or above should be used for JTAG programming.

2. 24Cxxx Serial EEPROM I2C algorithms
- The 24Cxxx Serial EEPROM algorithms require a chargeable '24Cxxx Upgrade License' to be purchased.
- For I2C programming of 24Cxxx Serial EEPROMs, please use firmware 3.07 or above.

1.7.2 Programming Interface to Target Device

The programmer supports the following programming interfaces to the Target Device:

#	Programming Interface	Family	Description / Comment
1	LV SPI	AT89S AT90S AT90USB AT90CAN AT90PWM ATmega ATmega PICO ATtiny Zensys	**Atmel Low Voltage Serial Programming Mode** - Atmel 3-wire SPI interface (SCK, MOSI, MISO) - Target Device RESET - Works at target voltage (no Vpp required)
2	HV SPI (+12V Vpp)	ATtiny	**Atmel High Voltage Serial Programming Mode** - Atmel 4-wire SPI Interface (SCK, MOSI, MISO, MOSI2) +12V Vpp is applied to the RESET pin to enter High Voltage Serial Programming Mode - Requires different connections to Target Device compared with LV SPI. - Requires all pins of target device to be accessible by the programmer.
3	UART Boot Loader	T89C51Rx2 T89C51AC2 T89C51Ix2 T89C51CC01 P89C51Rx2 P89C66xxx	**Atmel / Philips UART Boot Loader Interface** - Uses RXD, TXD, PSEN, and Target Device RESET pin - TTL Levels from programmer to Target System
4	JTAG (Atmel AVR)	ATmega AT90USB AT90CAN ATmega PICO	**Atmel ATmega AVR – JTAG ISP Interface** - Uses same connector as Atmel JTAG ICE - A chargeable license upgrade is required to enable the ATmega JTAG ISP libraries. - Supports JTAG Chain Mode programming
5	I2C	AT24Cxxx AT24HCxxx AT34Cxxx	**I2C 2-wire Interface** - There is no dedicated connector for I2C programming so it is necessary to re-use either an SPI or JTAG connector for I2C. - The programmers uses the following I/O pins for I2C: - SDA (SPI MOSI pin) - SCL (SPI SCK pin) - External pull-up resistors (eg. 4k7) are required on the SDA and SCL lines. These can be fitted on the IO-CON Module, Test Fixture or Target System.

Please refer to the separate manual *'PPM3 I/O Connector Modules'* for information on which ISP Header to use for the required programming interface.

2.0 Getting Started Guide

2.1 Overview

This section gives a 'Quick Start' Guide to using the programmer. Please refer to the table below for help installing the programmer hardware and software and also for the operating instructions for the programmer.

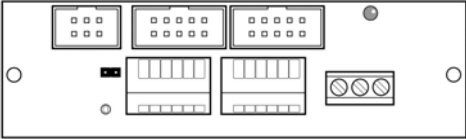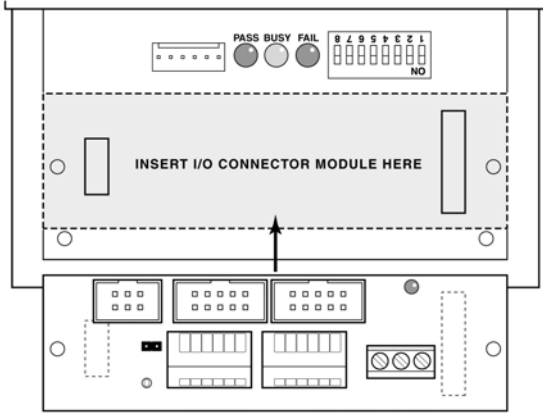| Description	Refer to section
Hardware Installation Procedure	2.2
Software Installation Overview	2.3
Programmer Operating Modes	2.4
Development Mode (EDS)	2.5
Standalone - Keypad Mode	2.6
Standalone – ASCII Text Communications Control	2.7
Standalone – 4-Wire TTL Control	2.8
Target System connect / Disconnect Detection	2.9
Script Mode	2.10
ConsoleEDS	2.11

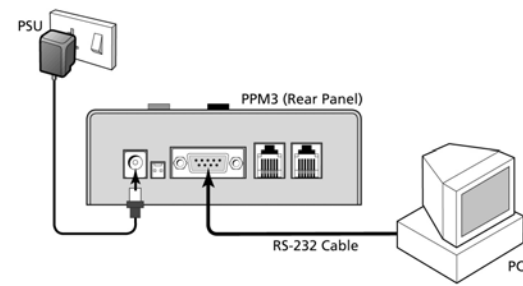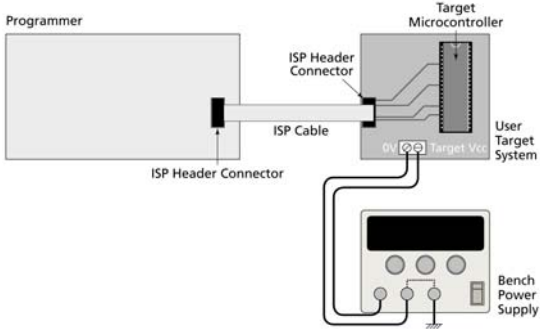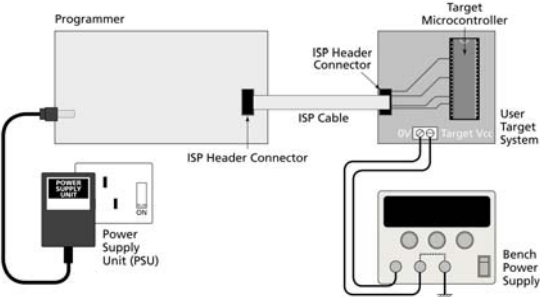Further help and instructions can be found in the following documents:

1. *'EQTools – Getting Started Guide'* (pdf manual) supplied on CD-ROM which came with the programmer (also available on-line)
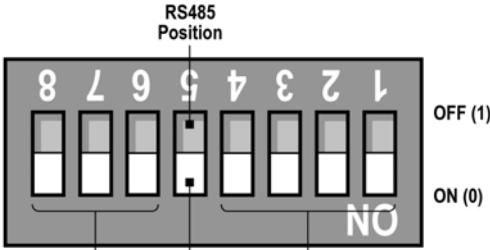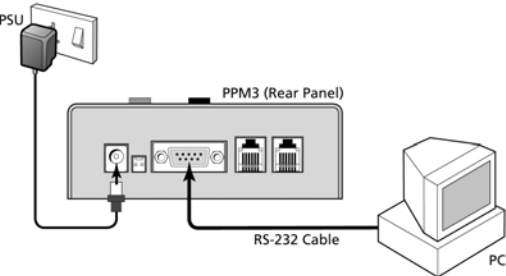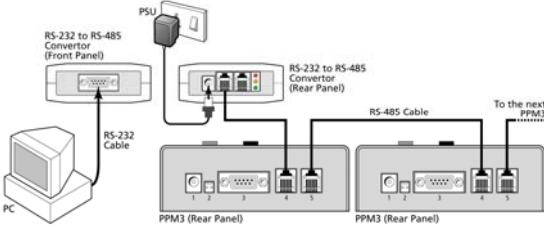2. *'EQTools – On-line Help File'* (Press <F1> within EQTools).

2.2 Hardware Installation Procedure
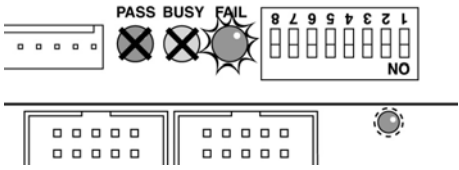
Please follow the instructions below to set up the programmer hardware:

#	Action	Observation
1a	**Ensure you are wearing an ESD strap and/or you are working in an ESD protected environment.**	The programmer I/O lines are ESD protected, but it always good practice to take proper ESD precautions. This can help to avoid ESD damage to either the programmer or Target System.
1b	**Remove programmer from all packaging and place on a flat surface.**	
1c	**If you are planning to use the programmer as a desktop programmer, affix the four stick feet to the base of the enclosure.**	The four rubber feet are supplied separately in a plastic bag.

2	**Select + fit I/O Connector Module**	
2a	**Select the required 'I/O Connection Module'** eg. EQ-IOCON-1 Module	For full details of which devices / algorithms each I/O Connection Module supports, please refer to the separate document 'PPM3 I/O Connectors Manual'. The PPM3 programmer is supplied with two I/O connection Modules as standard: • EQ-IOCON-1 • EQ-IOCON-2 A module for supporting JTAG programming can also be purchased as an optional extra: • EQ-IOCON-3
2b	**Fit the selected 'I/O Connection Module' to the programmer**	• Remove selected 'I/O Connection Module' from its packaging. • Position the module so that the long red connector on the back of the module matches up with the mating connector on the PPM Module. • Gently apply pressure to both sides of the module until it snaps securely into place.

3	Powering the programmer / Target System	
3a	**Select the required method of powering the programmer**	The PPM3 programmer supports the following power supply methods: • Programmer powered from external 9.0 – 12.0V DC regulated supply (eg. mains adaptor) – see (3b) • Programmer is powered from the Target System at +5.0V – see (3c) • Programmer and Target System are independently powered – see (3d).
3b	**Powering the programmer using an external power supply adaptor** 	• Plug power supply jack into mating jack socket on the rear of the programmer. • External power supply must be 9.0 to 12.0V DC regulated • Plug the mains adaptor into the mains supply but do not switch the power on.
3c	**Powering the programmer from the Target System at +5.0V** 	• The Target System supplies +5.0V to the programmer via either the ISP Cable / Header or the screw terminal power connectors. • The mains power supply adaptor must NOT be used if you are powering from the Target System.
3d	**Programmer and Target System are Independently powered** 	• The programmer is powered from +9.0 to 12.0V via the DC power input on the rear panel of the programmer. • The Target System is powered from an independent power supply. • The Vcc connection between the programmer and Target System must be removed. • The programmer 'I/O Level Convertor' must be powered at the same voltage as the Target System. This can be set up in the Programming Project,

4	Setting up the Communications Mode	
4a	**Select the Communications Mode (RS-232 or RS-485)** RS485 Position 8 7 6 5 4 3 2 1 OFF (1) ON (0) NO Spare reserved for future use RS232 Position Communications Address	The programmer supports the following communication modes: • RS-232 (single programmer channel – see 4b) • RS-485 (1 - 16 programmer channels - see 4c) The required Communications Mode can be selected using DIP Switch 5: • ON – RS-232 (default) • OFF – RS-485 **Please refer to section 3 for a full description of both modes.**
4b	**RS-232 Communication Mode** PSU PPM3 (Rear Panel) RS-232 Cable PC	• Single Channel RS-232 Operation • Connect Programmer to PC COM Port or other RS-232 Control System via RS-232 cable. • Set the 'RS-232/RS-485' DIP Switch to 'RS-232' • Set the Programmer Communications Node Address to '0' (factory default setting)
4c	**RS-485 Communication Mode** RS-232 to RS-485 Convertor (Front Panel) PSU RS-232 to RS-485 Convertor (Rear Panel) RS-485 Cable To the next PPM3 RS-232 Cable PC PPM3 (Rear Panel) PPM3 (Rear Panel)	• Multiple channel RS-485 Operation • Set the 'RS-232/RS-485' DIP Switch to 'RS-485' • Programmer now uses either of the RS-485 ports on the rear panel of the module. • Set the required Communications address using the DIP switches 1-4 as a binary weighted address in the range 0 – 15.

5	Testing the programmer for the first time	
5a	**Apply power to the programmer**	→ Programmer circuitry powers up to 5.0V. → FAIL LED flashes to indicate that the programmer has rebooted. → The programmer display now initialises to show the programmer type, firmware version etc. See 5(b), 5(c), 5(d).

5b	**No Programming Projects Loaded**
	
	 `NO / INVALID`
	 `PROJECT LOADED!`
		• If the programmer has never been used before or the on-board FLASH Memory Store has been erased, then the programmer will display the message: 'No / Invalid Project Loaded'.
		 • It is not possible to check the programmer information until a valid project is uploaded. Please upload a valid project to the programmer using EQTools and then try again.

5c	**Check the programmer firmware version**
	
	 `EQUINOX-TECH UK`
	 `PPM3 Ver:-2.37`
		• The programmer display now shows the programmer type and the firmware version.
		 e.g. 2.37.
		 • You can check that you have the very latest firmware version by viewing the 'Configit – Firmware Update Utility' on the Equinox website.

5d	**Check the programmer serial number, build revision and build date**
	
	 `SN:619 1.05`
	 `DATE 9-2-2003`
		After 2 seconds the display will update to show the following information:
		 • Serial Number (SN): eg. 619
		 • Hardware Build Version: eg. 1.05
		 • Build date: eg. 9 /2/2003

6	**Hardware installation is now complete.**	

PPM3 Programmer Module
High-end Programming Module

2.3 Software Overview and Installation

2.3.1 Software Overview

The PPM3 programmer is supplied with a comprehensive Integrated Development Environment (IDE) called **'EQTools Version 2.xx Software Suite'**. This software supports creation and testing of so-called Programming Projects which can then be uploaded to the programmer for use in 'Standalone Mode'. EQTools contains many separate utilities most of which are enabled as standard for this programmer. Certain options such as compiling and running of 'Programming Scripts' must be purchased from Equinox as a 'License upgrade.

The table below details the various EQTools components and whether they are enabled as standard.

Fig. 2.3.1 Software / Control Options for the PPM3 Programmer

EQTools - Project Builder	YES
EQTools - Project Management Utility	YES
EQTools - Project Upload/Download Utility (Upload Wizard)	YES
EQTools - Development Suite (EDS)	YES
EQTools - Script Builder	U
EQTools - Script Editor	U
Development Mode (PC Controlled)	YES
Standalone operation (Keypad – no PC)	YES
Remote System Control (4-wire)	YES
ASCII Text Mode Control	YES
ConsoleEDS – Evaluation version	YES
ConsoleEDS – PRO version	U
ISP-PRO Software	U
- Network Management Utility	U
- Interface Database (Diagnostics Log)	U
- Labview Control (PC)	U
- Remote Application Control (PC)	U

Key
YES – feature available as standard
U – a license upgrade must be purchased from Equinox to enable this feature.

2.3.2 Programmer Control Mechanisms

The programmer can be controlled from by many different methods including EQTools, EDS, from a Remote System via the Equinox ISP-PRO software or via the ConsoleEDS command-line software. An overview of the available control mechanisms is given below.

Standalone Operation(Keypad)
This indicates that the programmer can be operated without being connected to a PC or other control system. In this case an operator can control programming operations using a keypad on the programmer e.g. Epsilon5, FS200A, PPM3-MK2

Remote System Control (4-wire)
This indicates that the programmer can be controlled from a remote electrical system e.g. In Circuit Tester using a 4-wire TTL protocol (START, ACTIVE, PASS, FAIL). This method only supports execution of 'Standalone Programming Projects' which have been previously uploaded to the programmer.

ASCII Text Mode Control
This indicates that the programmer can be controlled using a simple 'ASCII Serial Communications Protocol'. This is ideal for applications where a Remote System wishes to control the programmer using a serial communications link e.g. RS-232. This method only supports execution of 'Standalone Programming Projects' which have been previously uploaded to the programmer.

ISP-PRO Software
This software suite is capable of controlling up to 16 x Equinox Production ISP programmers on an RS-485 network at the same time. All programming operations are controlled via 'Programming Scripts' and 'Programming Projects' which have been created using EQTools. The application minimises user intervention during programming and logs all programming operations to an Interface Database.

Labview Control (PC)
This allows an Equinox programmer to be controlled by a remote Labview Application. The Labview application can launch 'Programming Scripts' and read/write unique data such as serial numbers and calibration data.

Remote Application Control (PC)
This allows an Equinox programmer to be controlled by a Remote Application written in e.g. Visual Basic, C++ etc. The Remote Application can launch 'Programming Scripts' and read/write unique data such as serial numbers and calibration data.

ConsoleEDS
This is a powerful utility which allows any Equinox ISP Programmer to be controlled from a PC application by executing simple command-line instructions. This method of control is ideal for interfacing the programmer to any Windows application including Visual Basic, Labview etc.

2.3.3 Software Installation

The PPM3 programmer comes with the software suite called EQTools as standard. The latest version of this software which was available at the time of shipping is supplied on CD-ROM with the programmer. However, this software version is likely to have been superseded by a newer version. Please check the Equinox website for the latest version of this software.

Locate the latest version of the EQTools software by browsing to:
http://www.equinox-tech.com and clicking on the *<Downloads>* tab.

EQTools versions are quoted as e.g. *EQTools V2.1.0 Build 412*
The filename for this version would be '*eqtools210_412.exe*'

For further information, please locate the *'EQTools – Getting Started Guide'* PDF document either on the CD-ROM supplied with the programmer or from the Equinox Web Site. Refer to Section 2 of the guide for detailed EQTools installation instructions.

2.4 Programmer Operating Modes

The PPM3 programmer can be controlled in various different ways as detailed in the table below:

#	Programmer Mode	Control Software	U/S	Functional description	Refer to section
1	Development Mode	EQTools - EDS	S	In this mode, the programmer is controlled by EQTools - Equinox Development Suite (EDS). EDS allows you to read/write the target device FLASH / EEPROM / FUSE / LOCK bits interactively under PC control without uploading a 'Programming Project' to the programmer.	2.5
2	Standalone - Keypad Mode	None	S	In this mode, the programmer is not connected to a PC and can program a single 'Programming Project' into the Target Device when the <YES> key is pressed. (A valid project must have been previously uploaded to the programmer using EQTools – Upload Wizard.)	2.6
3	Standalone - ASCII Text Communications Control	Any Terminal Emulation Software	S	This protocol can be used to control an Equinox programmer from a Remote Application or Remote System using a simple set of ASCII serial commands. This protocol is ideal for controlling Equinox programmers from any Remote System which features an RS-232 serial port and which can operate at a fixed baud rate of 38,400.	2.7
4	Standalone - 4-Wire TTL Control	None	S	The PPM3 programmer can be controlled using a 4-wire TTL signalling protocol as shown in the diagram below. This control method is ideal for interfacing the programmer In-Circuit Testers (ICT's) or other production equipment which does not have an RS-232 port.	2.8
5	Target System auto-detect mode	None	S	The PPM3 programmer is capable of automatically detecting when a Target System is connected to / disconnected from the programmer. This can be used to trigger the execution of a Programming Project in Standalone Mode.	2.9

| 6	**Script Mode**	EQTools – Script Debugger or ISP-PRO	U	In this mode, the programmer is controlled from a 'Programming Script' running within either the EQTools or ISP-PRO applications on a PC.	2.10
7	**ConsoleEDS**	ConsoleEDS	U	In this mode, the programmer is controlled via the ConsoleEDS utility which allows control via simple Command Line commands and arguments	2.11 See App Note AN111

Key
S – feature available as standard
U – a license upgrade must be purchased from Equinox to enable this feature.

2.5 Development Mode (EDS)

2.5.1 Overview

In 'Development Mode', the programmer is controlled from the EQTools – EDS (Equinox Development Suite) running on a PC. The Development utility is called Equinox Development Suite or EDS for short. In this mode it is NOT necessary to upload the 'Programming Project' to the programmer as the EDS utility co-ordinates all programming operations.
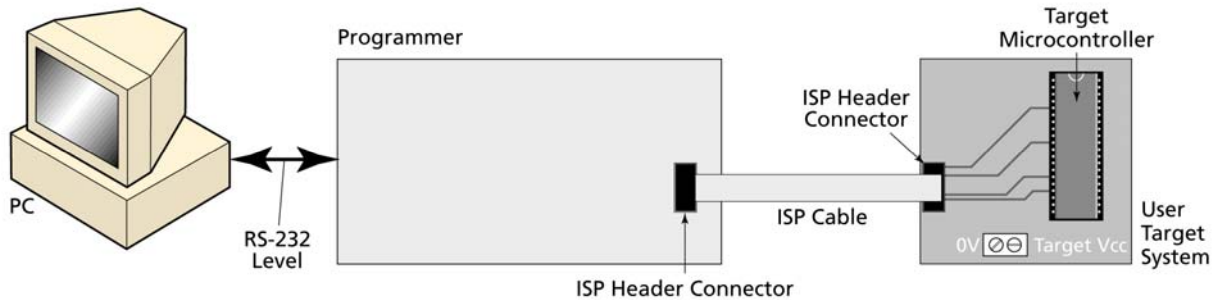
In EDS Mode, you can perform the following actions under PC control:
- Write data directly to a Target Device from the PC e.g. a HEX file
- Read data from a Target Device to a file on the PC
- Write / Read the Configuration Fuses of a Target Device
- Write / Read the Security Fuses of a Target Device
- Read the Target Vcc Voltage

2.5.2 Creating a new EDS (Development Mode) Project

To test a Target Chip in EDS (Development Mode), follow the instructions detailed below:
- Launch EQTools
- From the Welcome Screen, select the **'Create a new Equinox Development Project (EDS)'** option.

  OR
- From the left-hand pane, select the **<Development>** tab and then select **'New Development Project'**.

→ The EDS Wizard will now launch

- Follow the EDS Wizard by filling in each tab and then clicking the <Next> button:
  - Select Programmer – click <Get Info> button to detect your attached programmer
  - Select Project Type – leave as the default 'Standalone Keypad Operation'
  - Select Device – This is your Target Chip.
  - Select Target Oscillator – If unsure of settings leave all settings as the defaults and click <Next>.
  - Select Target Voltage
  - Select FLASH Data File (optional – you can enter the file name at a later stage in EDS)
  - Select EEPROM Data File (optional – you can enter the file name at a later stage in EDS)

- At the end of the EDS Wizard:
  Click the <Test> button
  Save the EDS file with an appropriate name eg. ATmega163.eds
  → The EDS Window will now launch – see section 2.5.4

2.5.3 Testing an existing Programming Project in a Project Collection in EDS (Development Mode)

If you have already created a Programming Project and want to test it in EDS (Development Mode), please follow the steps below:

- From the left-hand pane, select **<Project Manager>**
- Select **<Open Collection>**
- Browse to and select a Project Collection (*.ppc) file
- → Project Collection File launches in **Project Manager** view
- Select the Programming Project you wish to test in EDS
- Click the **<Test Project in EDS>** button at the bottom right of the **Project Manager** window
- → The EDS window now launches with all the settings of the select Programming Project – see section 2.5.4

2.5.4 Overview of EDS – Development Mode

The EDS Utility consists of a series of tabs. Each tab controls a different area of functionality including FLASH Area Programming, EEPROM Area programming, Configuration Fuse Programming and Security Fuse programming. It is also possible to set up the SPI / JTAG / UART Target Speeds and the Power Supply Settings from EDS.

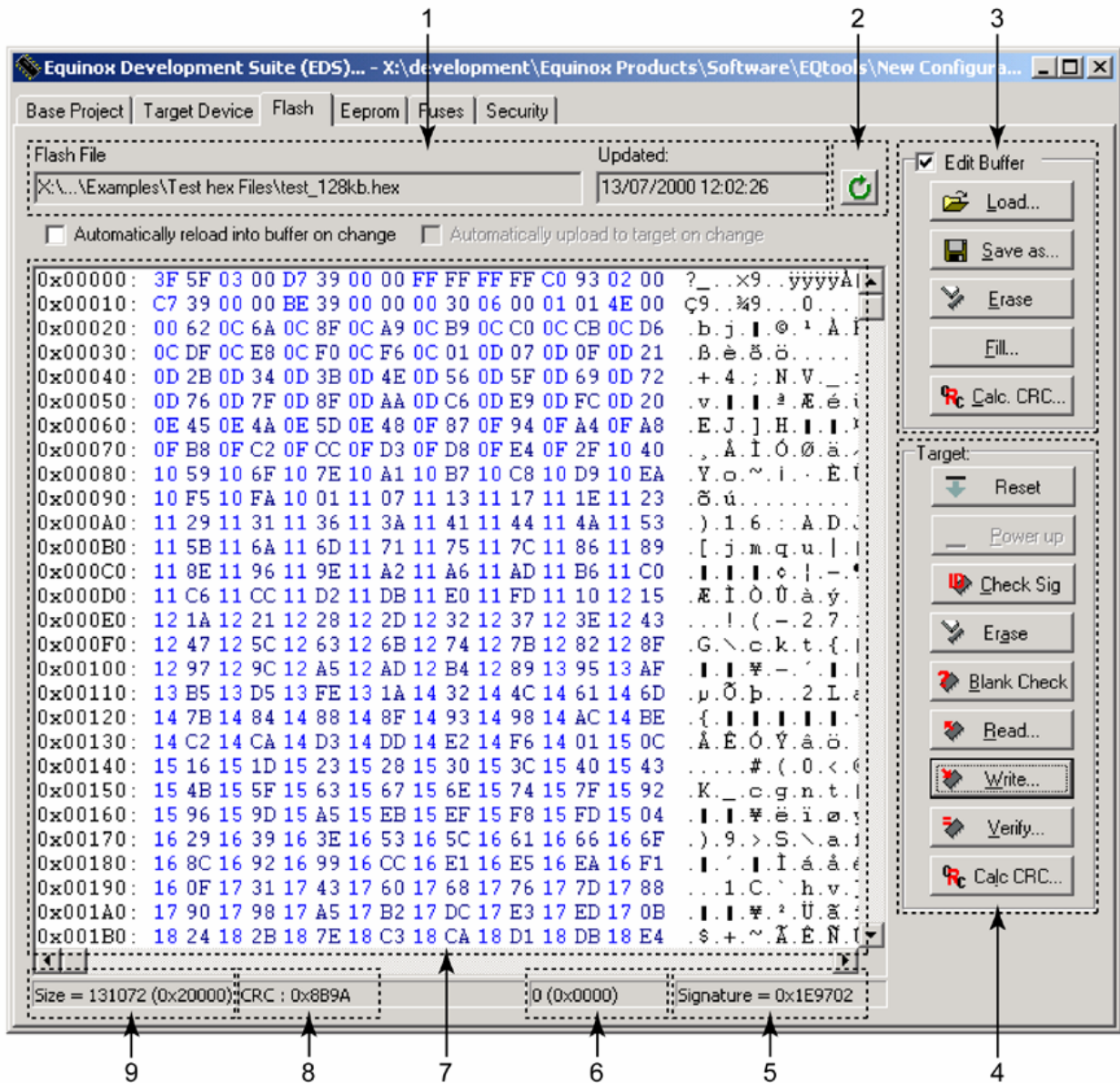*Fig. 2.5.4.1 EDS – FLASH Tab Overview*

Fig. 2.5.4.2 FLASH / EEPROM tab – functional description

#		
1	**File name and properties**	This specifies the 'File Name / path' and 'Last updated' date of the file to be loaded into the Buffer Window. • By default, this will point to the file specified in the associated Programming Project (*.ppm). • To load a different file, click the <Load> button and browse to the required file.
2	**Re-load file to buffer**	Clicking the [icon] icon re-loads the specified Flash / EEPROM file into the Buffer Window. This will overwrite any information already in the Buffer Window.
3	**Buffer Control Buttons**	This group of buttons control operations on the EDS – Buffer Window. • To use these buttons or to manually edit the Buffer Window, it is necessary to check the 'Edit Buffer' check box. • This extra step helps to avoid accidental modification of data in the buffer.
4	**Target Control Buttons**	This group of buttons control operations on the actual Target Device. • A programmer and suitable Target System must be connected when using these buttons.
5	**Signature of Target Device**	This is the signature (Device ID) which is expected for the Target Device.
6	**Buffer Address**	This is the address of the currently selected location in the Buffer Window.
7	**Buffer Window**	The Buffer Window displays a hexadecimal and alphanumeric representation of the data which has either been loaded from file or read back from a Target Device. • The bytes are grouped into rows of 16 bytes with the start address of each row displayed in the left-hand column. • The Hexadecimal representation of the 16 bytes is displayed in the middle column • The ASCII representation of the 16 bytes is displayed in the right-hand column.
8	**CRC**	This is a CRC Checksum of the entire Buffer (ie. from address 0x00000 to the end address specified in the buffer.
9	**Size**	This is the physical address range in bytes of the entire Buffer.

2.5.5 Buffer Window - Control Buttons

The Buffer Window within EDS is used as a virtual data store which allows data to be transferred from file → Buffer Window → Target Device and vice-versa.

The 'Buffer – Control Buttons' support operations on the data in the Buffer Window including loading a file into Buffer or saving the contents of the Buffer to file. These buttons do NOT initiate any actual programming operation of the Target Device.

#	Button	Description / Purpose
1	**Re-load File**	• Re-loads the specified HEX / Binary file into the Buffer Window.
2	**Edit Buffer**	• Tick the box to enable manual editing of the Buffer Window. • This will also allow the <Load>, <Erase> and <Fill> buttons to be selected.
3	**<Load>**	• This button allows you to manually browse to select a file to be load into the Buffer Window. • The path will default to the path set up in the Programming Project (if a file is specified).
4	**<Save as>**	• This button allows you to save the contents of the Buffer Window to a file. It is possible to specify the following properties for the file: • File type (HEX / Binary) • File Start and End Address
5	**<Erase>**	• This button erases the entire contents of the Buffer Window. • All locations are set to the value 0xFF.
6	**<Fill>**	• This button allows you to fill a specified area of the Buffer Window with a specified value.
7	**Calc CRC**	• This button calculates the CRC Checksum of the specified range of the Buffer Window.

2.5.6 Example of using EDS

Here is an example of how to use EDS:

- Select the **<FLASH>** tab
- Click the **<Load>** button to load a file
- Click the **<Check Sig>** button to check the Device Signature of the Target Chip
- Click the **<Write>** button to write the data in the Buffer Window to the Target Device
- Click the **<EEPROM>** tab to select operations on the EEPROM area of the Target Chip/
- Click the **<Fuses>** tab to read / write the Target Chip 'Configuration Fuses'
- Click the **<Security>** tab to read / write the Target Chip 'Security Options

For further instructions about how to use the EDS utility, please refer to the **'EQTools – Getting Started Guide'** which can be found on the CD-ROM supplied with the programmer.

2.6 Standalone Keypad Mode
2.6.1 Overview

In 'Standalone Mode', the programmer is not connected to the PC and is controlled via the push buttons on the front panel instead. The Programmer LCD and the LED Status Indicators are used to display the current status of the programmer.

LCD Display	Buttons	Status LED's
PRESS YES TO PROGRAM TARGET	✖ NO UP ▲ ✔ YES DOWN ▼	PASS BUSY FAIL ⊗ ⊗ ⊗
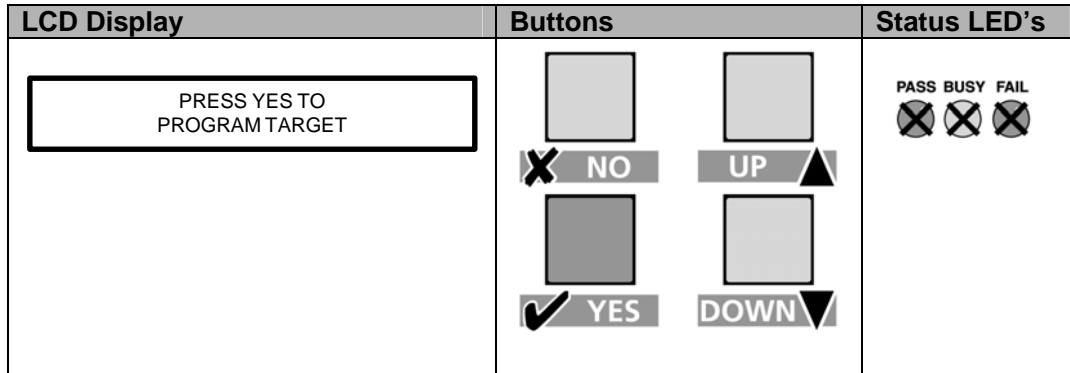
This mode is ideal for both field and production ISP requirements as the programming data is held locally in FLASH memory on the programmer so a PC is not required.
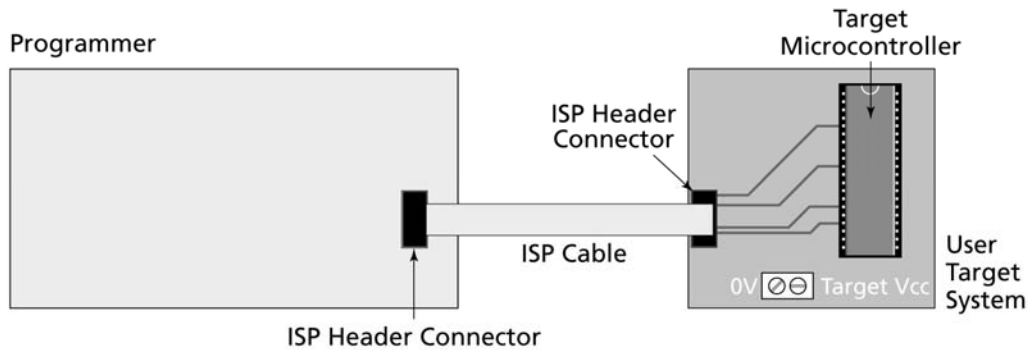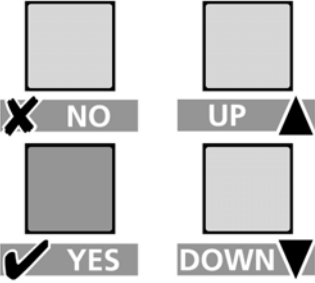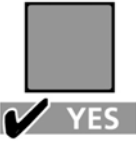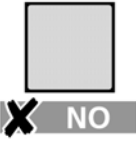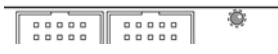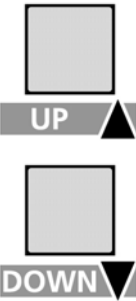

A valid Project Collection containing at least one Programming Project must be uploaded to the programmer before attempting to use Standalone Mode.

To execute a Programming Project in Standalone Mode:

#	Action	Observation / comment
1	Connect the programmer to a spare PC COM port	
2	Apply power to the Programmer / Target System	→ Programmer Target Vcc LED illuminates
3	Upload your 'Programming Project(s)' to the programmer	See EQTools Manual – Project Upload Wizard for full instructions.
4	Disconnect power from the Programmer / Target System	
5	Disconnect the serial lead from the programmer	→ Programmer is no longer controlled from PC
6	Follow the instructions to program the Target Device in Standalone Mode	

2.6.2 Push Button Functions

	The PPM3 programmer features a four button keypad which is used to control the programmer when it is operating in 'Standalone Mode'. The function of these buttons is described below.

Button	Function of button
✔ YES	The **<YES>** button has the following functions: i. To execute a programming project which has been pre-loaded into the programmer 'FLASH Memory Store'. ii. To return to the 'Standby' state
✘ NO	The **<No>** button has the following functions: i. To cancel a programming action ii. To return to the 'Standby' state
UP ▲ DOWN ▼	• The **<Up>** and **<Down>** buttons are used to scroll up and down through the list of available 'Programming Projects' stored in the programmer 'FLASH Memory Store'. • If there are not 'Programming Projects' currently in memory, then pressing thee buttons will have no effect.

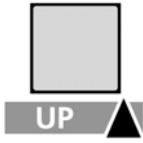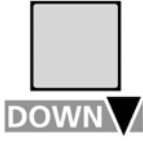## 2.6.3 Standalone Keypad Operation – step-by-step guide

The instructions below detail how to operate the programmer in Standalone Mode using the programmer keypad and LCD. A PC is not required while operating in Standalone Mode.

1. Ensure that you have already uploaded a valid 'Project Collection' to the programmer
- Use the EQTools – Upload Wizard to upload your 'Project Collection'.
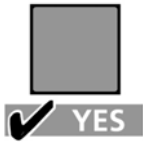
2. Select the 'Programming Project' which you wish to execute:
- Use the *<UP>* and *<Down>* keys to move up and down the list of projects currently resident in the programmer 'FLASH Memory Store'.
- If there is one 'Programming Project' resident in the programmer, then this project will be displayed permanently on the LCD.
- If Project Version Control has been enabled for the project(s), then the project version information will be displayed on the bottom line of the LCD.

LCD Display	Buttons	Status LED Display
Select Project Project1 Select Project Project2 Select Project Project3	UP ▲ DOWN ▼	PASS BUSY FAIL ⊗ ⊗ ⊗

3. Press the RED <YES> button to select the Programming Project

LCD Display	Buttons	Status LED Display
Select Project Project1	✔ YES	PASS BUSY FAIL ⊗ ⊗ ⊗

4. Wait for the programmer to CRC check the Programming Project

- The programmer performs a full CRC validation check of the entire Programming Project stored in the programmer 'FLASH Memory Store'.
- If the check is OK, the programmer goes to step (5).
- If the check FAILS, the programmer will display 'Internal Image is corrupted'. Please try re-uploading your Project Collection and then repeat the Standalone Mode process from the beginning.

LCD Display	Buttons	Status LED Display
CHECKING IMAGE PAGE n	✔ YES	PASS BUSY FAIL

Please note:

- It is possible to switch off the *'Project CRC Image Check'* function by disabling the 'CRC check' in the Programming Project.
- This feature is available in EQTools build 888 or above and firmware 3.07 or above.

5. Press the RED <YES> button again to execute the Programming Project

- The Programming Project will now start to execute.
- The *<BUSY>* LED will illuminate to denote that the programmer is active.
- This may take a few seconds or a few minutes depending on the nature of the Programming Project.
- The Programming Project will either PASS or FAIL

LCD Display	Buttons	Status LED Display
Current programming operation is displayed	To abort the programming sequence at any time, press ✖ NO	PASS BUSY FAIL

6. Programming Project – PASS

If the Programming Project programs the target device without any errors:

- The *<PASS>* LED will FLASH
- The LCD will display: 'Operation Complete OK!'
- Press *<YES>* key to program another device

LCD Display	Buttons	Status LED Display
OPERATION COMPLETE OK	Press	PASS BUSY FAIL

	✔ YES To program another device.	

7. Programming Project – FAIL

If the Programming Project fails for some reason:

- The *<FAIL>* LED will FLASH
- The programmer will automatically remove power from the Target System (if the programmer is controlling power)
- The LCD will display the relevant Error Message.
- The List of possible Error Message can be found in the 'Error Messages' section of this manual.
- Press the *<NO>* button to clear the Error Message
- Press *<YES>* key to program another device. The programmer then goes back to step 3.

LCD Display	Buttons	Status LED Display
ERROR MESSAGE	Press ✗ NO to clear the error message.	PASS BUSY FAIL ⊗ ⊗ ☀

2.6.4 Standalone Keypad Operation – Flowchart

The flowchart below details how a 'Programming Project' is selected in 'Standalone Mode'.
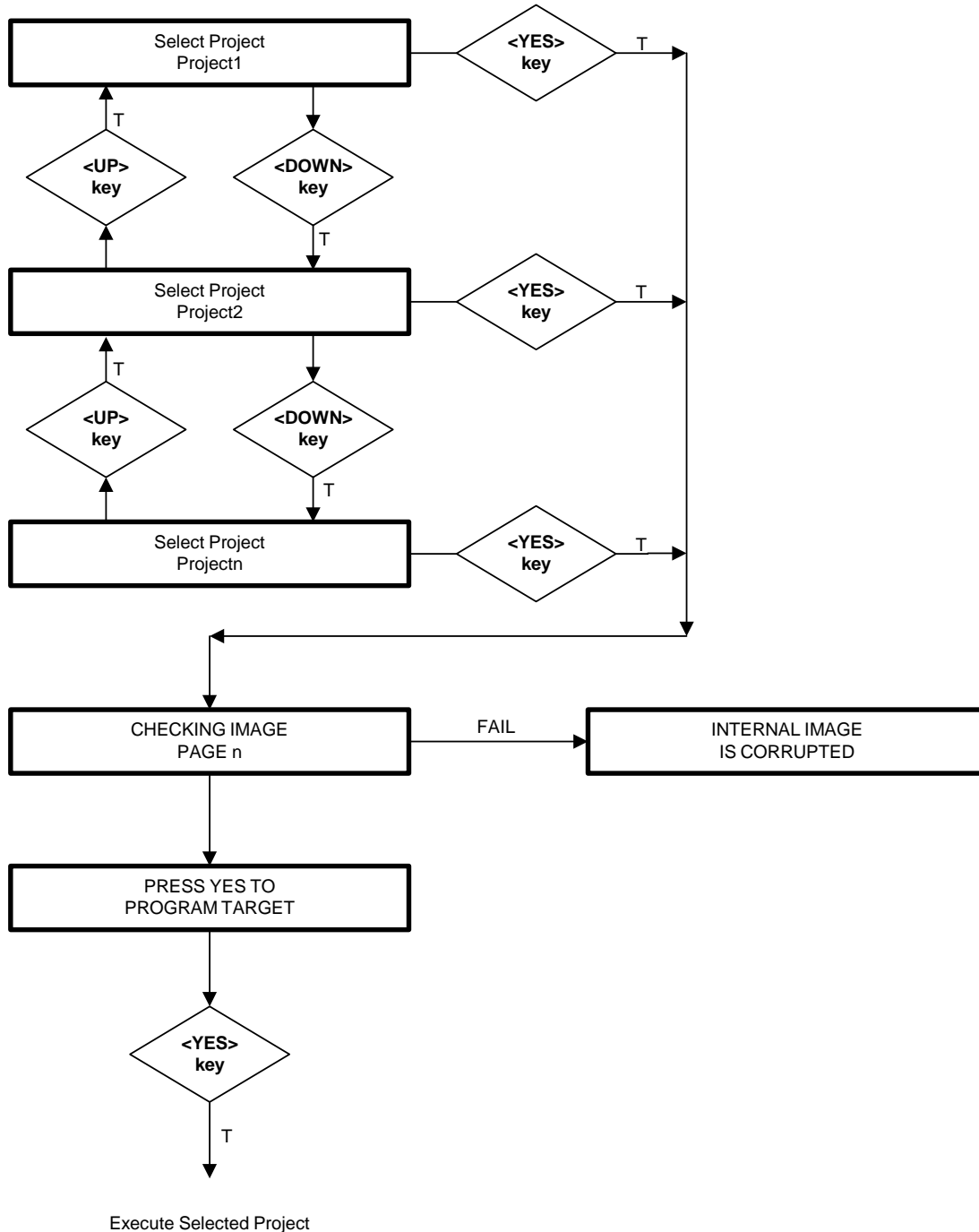
2.6.5 Programmer Status LED's

The current status of the programmer is displayed on the programmer Status LED's as detailed in the table below.

Fig. 2.6.5 Programmer Status LED's – state descriptions

Status LED Display	State	State Description
PASS BUSY FAIL	**WAITING**	• Programmer is now waiting to start an 'Autoprogram' operation.
PASS BUSY FAIL	**BUSY**	• Programmer is 'BUSY' performing a programming operation. • If the programmer is controlled from EDS, the BUSY LED will remain on after a programming operation until a 'RESET programmer' command is executed.
PASS BUSY FAIL	**FAIL**	• Programming operation has FAILED. • See programmer LCD for diagnostics • Programmer will automatically switch off target power if it is controlling power.
PASS BUSY FAIL	**PASS**	• Programming operation was successful.

Status LED key:

ON FLASHING OFF
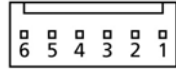
Please note:
There are also two special modes called *'Waiting Target Connection'* or *'Waiting Target Disconnection'* where the yellow BUSY LED will FLASH. These modes are enabled when 'Target Connection Sense' is enabled in the Programming Project.

2.6.6 Remote Status LED's

It is possible to connect remote Status LED's to the PPM3 programmer. This can be useful if the programmer is mounted underneath a Test Fixture thus obscuring the normal Status LED's. The Remote Status LED's simply connect to the 'Remote System Status' port (J13 Molex Connector) as detailed in the table below.

Fig. 2.6.6 Remote System Status Connector and pin-out

Connector	Connector pin-out
6 5 4 3 2 1	See table below

Pin	Signal name	Description / Function
1	FAIL+	Connect to ANODE of FAIL LED
2	FAIL-	Connect to CATHODE of FAIL LED
3	BUSY+	Connect to ANODE of BUSY LED
4	BUSY-	Connect to CATHODE of BUSY LED
5	OK+	Connect to ANODE of OK LED
6	OK-	Connect to CATHODE of OK LED

The ANODE of each LED should be connected to the relevant '+' connector and the CATHODE of each LED should be wired to the relevant '-' connector. It is not necessary to insert a series resistor with each LED as these are already included on the PPM3 circuit board.

2.7 ASCII Text Communications Mode

2.7.1 Overview

The 'ASCII Text Control' communications protocol or 'ATC' is designed as a simple way for a Remote Test System to control the basic programming operations of an Equinox programmer via an RS-232 serial link. It is an 'ASCII' protocol with simple commands to initiate programming of pre-loaded 'Programming Projects' which are stored in the on-board 'FLASH Memory Store' of the programmer. The protocol supports up to 64 independent Programming Projects per programmer, but is limited to one programmer.

Implementation of this protocol is straightforward due to the limited number of commands and restricted functionality. As this is an ASCII protocol, it can be quickly evaluated using any Terminal Emulation Software Utility such as Hyperterminal.

2.7.2 Programmers supported

The Equinox programmers which are capable of supporting 'ASCII Text Mode' control are listed in the table below.

Programmer	ASCII Text Mode Control	Order Code
EPSILON5	License upgrade required	EPSILON5A1-UPG6
FS2000A	License upgrade required	FS2000A-UPG7
PPM3 MK1	Enabled as standard	N/A
PPM3 MK2	Enabled as standard	N/A

Please note:
1. The PPM3 range of Production ISP Programmers are enabled for 'ASCII Text Mode' control as standard. The 'Communications Node Address' must be set to address '0'.
2. A license upgrade is required for all other programmers to enable them for 'ASCII Text Mode'.

2.7.3 Limitations of this protocol

This protocol has been designed as a simple method to allow a Remote Application such as an In-Circuit Tester to control the basic operations of an Equinox programmer. The programmer only supports the programming of entire 'Programming Projects' which are already uploaded to the programmer on-board FLASH memory. This protocol does not support writing / reading of individual bytes or blocks of data to / from a Target Device.

The limitations of this protocol are as follows:
1. The protocol only supports programming of Programming Projects which are pre-loaded into the Programmer on-board FLASH Memory Store.
2. The protocol does not support writing / reading of individual bytes or blocks of data to / from a Target Device.
3. The protocol does not support uploading of 'Programming Projects' to the programmer. This must be performed using the EQTools – Project Upload Wizard.
4. Only very limited diagnostics are available using this protocol ie. FAIL + Error Number.

2.7.4 Further Information

A full description of this protocol can be found in the Application Note 'AN110 - ASCII Text Control (ATC) Protocol for Remote Control of Equinox Programmers'.

2.8 Standalone Remote I/O Programmer Control

2.8.1 Overview

The PPM3 programmer can be controlled using a simple 4-wire Remote I/O (TTL) signalling protocol as shown in the diagram below. This control method is ideal for interfacing the programmer to In-Circuit Testers (ICT's) or other production equipment which does not have an RS-232 port.

Fig. 2.8.1 Remote System Control of Programmer
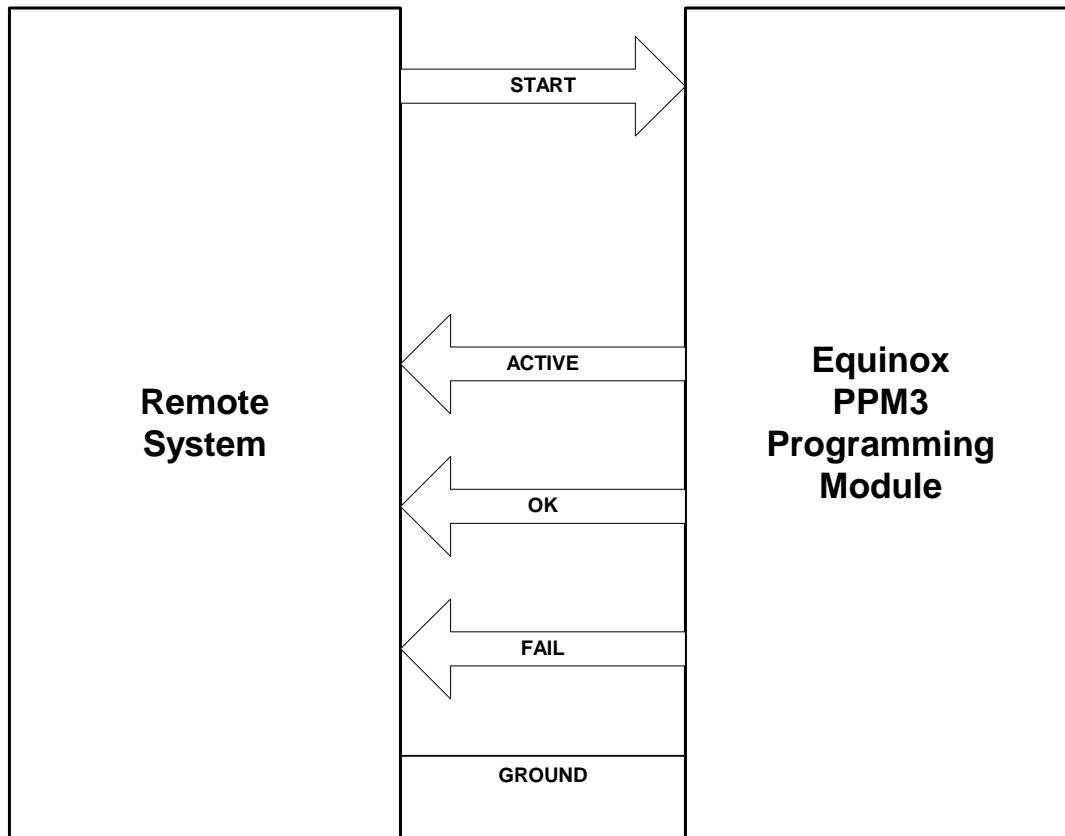
The Remote System simply asserts the programmer *<START>* signal to initiate the execution of a 'Programming Project'. The programmer will then assert the *<ACTIVE>* signal to indicate that it has commenced programming. At the end of the Programming operation the programmer asserts either the *<OK>* or *<FAIL>* signal depending on the outcome. The Remote System must then de-assert the *<Start>* signal to allow the programmer to reset ready to program the next device.

Please refer to section 2.8.3 for an example of the Remote I/O control signal waveforms for a typical programming iteration.

2.8.2 TTL 4-Wire Control – Signal Connections

The Remote System connects to the programmer via the 'Remote System Control Port' which is located on the relevant I/O Connection Module as shown in the table below. The signal names and descriptions are detailed in fig. 2.2.2b.

Fig. 2.8.2a Remote System Control Port – signal connections

I/O Connection Modules 1 and 3	I/O Connection Module 2
The 'Remote System Control Port' is the 6-way 'Fast Connect' connector marked as (5) on the above diagram. Unfortunately, the 'ACTIVE' signal is missing from this connector so this must be manually soldered to one of the red connectors.	The 'Remote System Control Port' is located on the wire-wrap port marked as (4) on the above diagram.

Fig. 2.8.2b Remote System Control Port – signal names and descriptions

Pin No.	Title	I/O	Description
1	START(+)	I	Remote <Start> Input Signal (3V –12 V DC)
2	ANA1	I	Spare Input Line (referenced to Target Vcc)
3	FAIL	O	FAIL (output)
4	OK	O	PASS (output)
5	TARGET_VCC	P	Target Vcc Voltage
6	TARGET_GND	P	Target GROUND

2.8.3 Control diagram for Remote I/O 4-wire Control

The control sequence for the '4-wire Control' method is detailed below.

Fig. 2.8.2 Control diagram for 4-wire Control
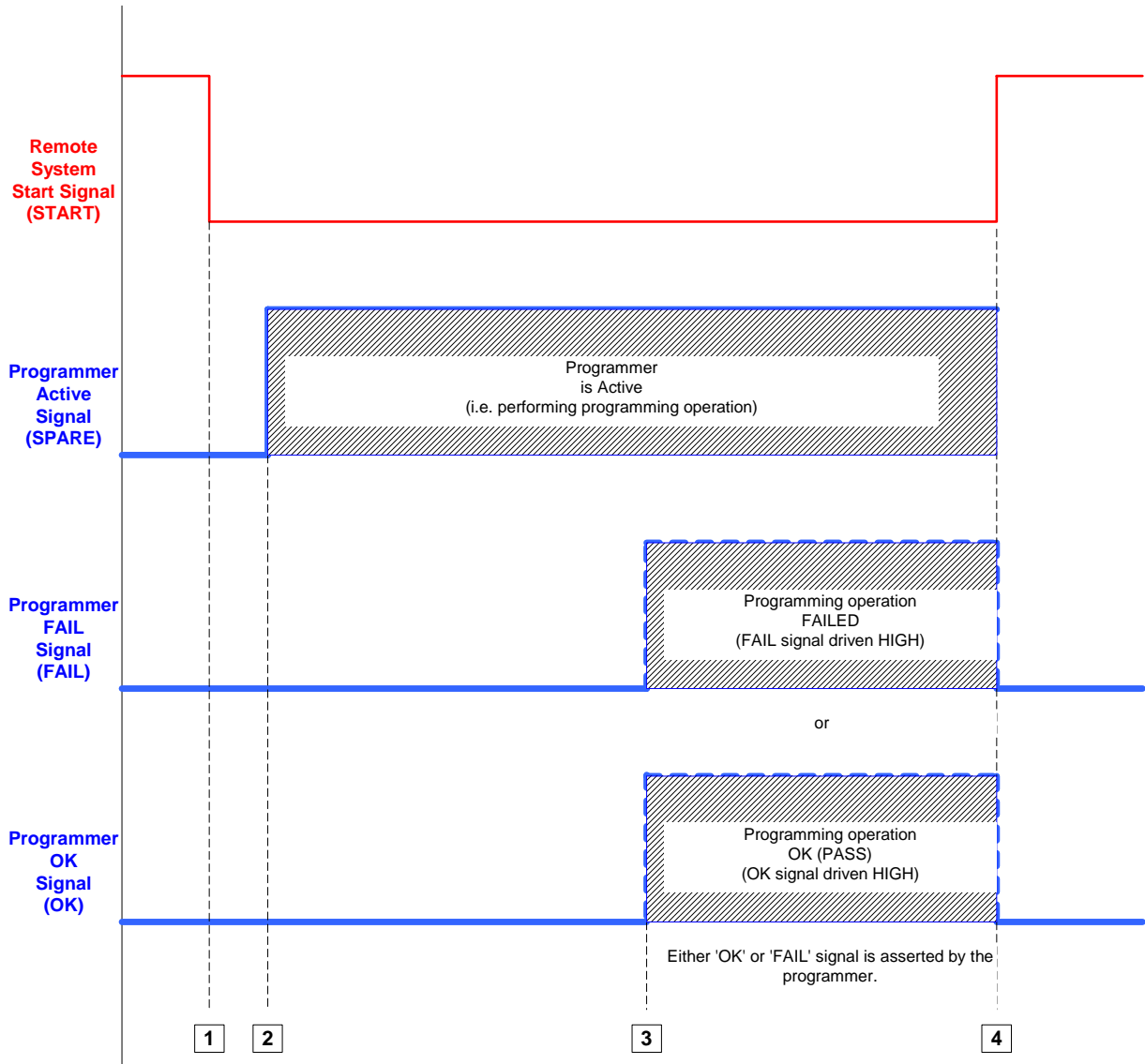
Key	Programmer / Remote System Action
1	The Remote System asserts the programmer <START> signal to initiate the execution of a 'Programming Project'.
2	The programmer will then assert the <ACTIVE> signal to indicate that it has commenced programming.
3	At the end of the Programming operation the programmer asserts either the <OK> or <FAIL> signal depending on the outcome.
4	The Remote System must then de-assert the <Start> signal to allow the programmer to reset ready to program the next device.

2.8.4 Setting up a Programming Project for TTL Control Mode

When using 'TTL Control Mode', a single 'Standalone Programming Project' or set of 'Chained Projects' must have already been uploaded to the programmer. This project must be have the 'Project Type' set to 'Standalone – Remote I/O Control'.

Please note:

It is not possible to control multiple Programming Projects using TTL mode as there is no method by which to select the project to be executed. TTL mode is therefore limited to single project.

To create a TTL Mode control project:

- Select **<Project Builder> <Create new project>**
- On the **<Programmer and Project type>** screen, select:
    - o Programmer: **PPM3-MK2**
    - o Project type: **Standalone – Remote I/O Control**
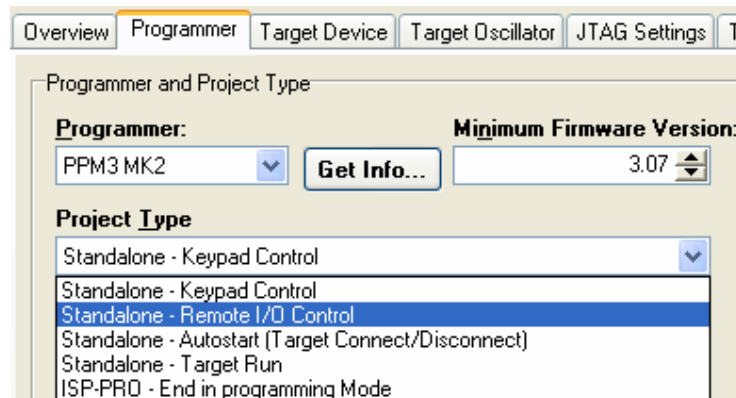
- Set up the FLASH / EEPROM / Fuses / Security Fuses as normal
- Compile the project
- Upload to the programmer using the Upload Wizard

2.9 Target System Connect / Disconnect detection

The PPM3-MK2 programmer has a special mode where it can detect when a Target System (UUT) is connected to or disconnected from the programmer. A load sensing technique is used to detect the presence of the UUT. This method of UUT detection can be used to trigger a 'Standalone Programming Project' or even a sequence of chained projects to start.

The typical sequence of events is as follows:

- Programmer waits for UUT to be connected (flashing BUSY LED)
- Operator connects UUT to programmer
- Programmer detects connection of UUT
- Programmer automatically executes a sequence of Standalone Programming Projects
- At the end of the programming sequence, the programmer will FLASH either the PASS or FAIL LED
- Programmer then goes into 'Target Disconnection Sensing' mode (flashing BUSY LED)
- Operator removes the UUT
- Programmer resets itself ready for the next UUT

Please contact Equinox for further instructions on how to set up your programmer to operate in this mode.

2.10 Script Mode (ISP-PRO)

'Script Mode' is designed for production users who require logging of all programmer operations to a database. This mode utilises the Equinox ISP-PRO software which allows execution of Programming Scripts. It is possible to write scripts to program unique data such as serial numbers, calibration data etc. It is also possible to control the programmer from a Remote Application written in eg. Labview, Visual Basic, C++ etc.

In order to use 'Script Mode' with your programmer, it is necessary to purchase the relevant license upgrade for ISP-PRO. Please see the Equinox Website for further details.

2.11 ConsoleEDS – Command Line control

ConsoleEDS is a powerful software utility which allows any Equinox ISP Programmer to be controlled from a custom Remote Application running under Windows.

The main features of ConsoleEDS are:

- Allows an Equinox ISP Programmer to be remote controlled via simple Command Line commands.
- Suitable for interfacing to any application which executes under Windows
- Simple Command Line interface makes even complex programming operations simple to implement
- Supports writing of a block of data from a file to the FLASH or EEPROM of a Target Chip
- Supports reading of a block of data from the FLASH or EEPROM of the Target Chip to a file on the PC hard disk
- Supports uploading of pre-compiled standalone 'Programming Projects' to a target programmer without requiring EQTools or the Project Upload Wizard.
- Supports running of a specified 'Programming Project' which has already been uploaded into the programmer 'FLASH Memory Store'
- Supports programming of the 'Configuration Fuses' of the Target Chip
- Supports programming of the 'Security Fuses' of the Target Chip
- Supports automatic generation of and programming of unique information such as serial numbers and calibration data
- Supports reading / writing of Atmel AVR on-chip Oscillator Calibration Byte

Please note:
Many of the above features are only accessible when using 'ConsoleEDS PRO'.

Typical ConsoleEDS example:
ConsoleEDS BaseProject.prj /AUTOPROGRAM=MainFirmware
/EEPROMWRITE=CalData.hex,0,50

For further information about ConsoleEDS, please refer to Application Note 111.

3.0 Serial Communications Ports

3.1 Overview

The PPM3 Programming Module can be controlled from any external device which supports either the RS-232 or RS-485 hardware communications protocols. The Module features an RS-232 port which can connect to a COM port of any suitable Control System such as a PC running the Equinox EQTools software. This mode of operation is restricted to a single programmer channel.

In many medium to high volume production applications there is a requirement to run more than one PPM3 programmer from the same PC COM port. This can be achieved by daisy-chaining up to 16 modules (using the RS-485 In/Out ports on each programmer) on an RS-485 bus and using an RS-485 converter to convert the PC RS-232 COM port to the required RS-485 levels. The Equinox ISP-PRO Software Suite is required to run the programmers in this mode.

The table below details the applications of both the RS-232 and RS-485 communications modes.

Fig. 3.1 PPM3 – Comparison of RS-232 / RS-485 Serial Communication Interfaces

Mode	Maximum Programming channels	Programmer Node Address	Serial port	Description
RS-232	1	0 (can be altered)	RS-232 port (3)	• Single Channel RS-232 Operation • Set 'Communications Mode Switch' to 'RS-232' • Programmer connects to PC COM port or other RS-232 Control System via RS-232 cable. • Programmer Communications Node Address is set to '0' • This is the factory default setting.
RS-485	16	0-15	RS-485 port (4 & 5)	• Multi-channel RS-485 operation • Set 'Communications Mode Switch' to 'RS-485' • Allows up to 16 x PPM3 Modules to be daisychained from one PC COM port. • Requires an external RS-485 Converter • If more than one module is to be connected to the same PC, then the Equinox ISP-PRO Software Suite is also required. • Programmer Communications Node Address on each module must be set to a unique address between 0 and 15.

3.2 Communications Connectors (Rear panel)

The diagram below shows the rear panel of the PPM3 Module. Please refer to the table below for a description of each port.

Fig. 3.2.1 PPM3 Module – Rear Panel Connectors
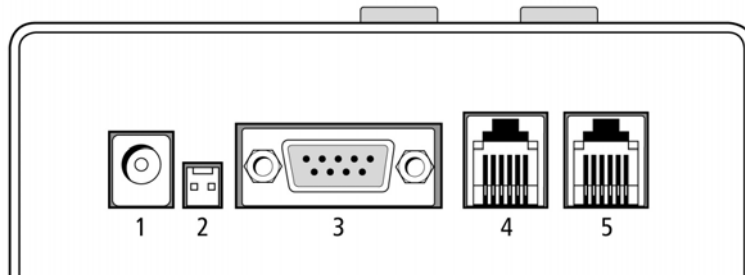
Key No	Function	Further description
1	DC Power Input	Jack socket
2	DC Power Input	2-way Molex connector
3	RS-232 Communications Port	9-way female D-connector
4	RS-485 Input (& output)	Input/output for RJ11 RS-485 cable
5	RS-485 Input (& output)	Input/output for RJ11 RS-485 cable

3.3 Communications - DIP Switch Settings

The Communications Mode (RS-232 / RS-485) and also the Communications Node Address are configured by setting the relevant combination of DIP Switches as detailed in the diagram below.
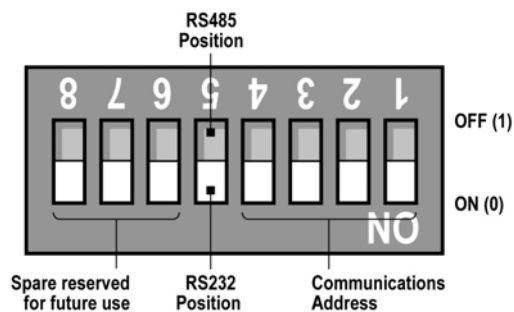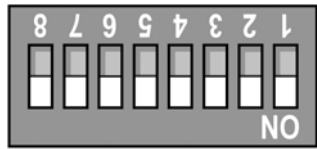
For RS-232 operation (default setting):
- Set the RS-232 / RS-485 DIP Switch (5) to the ON (RS-232) position and set the Node Address to '0'.

For RS-485 operation:
- Set the RS-232 / RS-485 DIP Switch (5) to the OFF position and set the required Node Address in the range 0 – 15. The address is a simple binary address with 0=ON position and 1=OFF position.

Fig. 3.3.2 Communications – DIP Switch Settings

Mode	Node Address	DIP Switch Setting	Comment
RS-232	0		• **RS-232 Mode Enabled (factory default)** • Set DIP Switch 5 to the 'RS-232' position • Programmer now uses the 9-way COM port on the rear panel of the module. • Communications address is set by DIP switches 1-4 as a binary weighted address in the range 0 – 15. • Factory Default address is 0. (i.e. DIP Switches 1-4 in the ON or '0' position).
RS-485	0		• **RS-485 Mode Enabled** • Set DIP Switch 5 to the 'RS-232' position • Programmer now uses either of the RS-485 ports on the rear panel of the module. • Communications address is set by DIP switches 1-4 as a binary weighted address in the range 0 – 15.
RS-485	1		
RS-485	2		
RS-485	3		

The layout of the DIP switch is shown in the diagram below.

3.4 RS-232 Operation

The PPM3 programmer is shipped from the factory enabled for 'RS-232' communications at communications address '0'. This is the normal mode of operation when only one programmer is to be connected to a PC or other RS-232 enabled Control System. A suitable 9-way to 9-way serial cable is supplied with the programmer which connects from the programmer 9-way RS-232 port to a spare COM port on your PC.

Fig. 3.4.1 Set up for RS-232 Communication (Single channel)
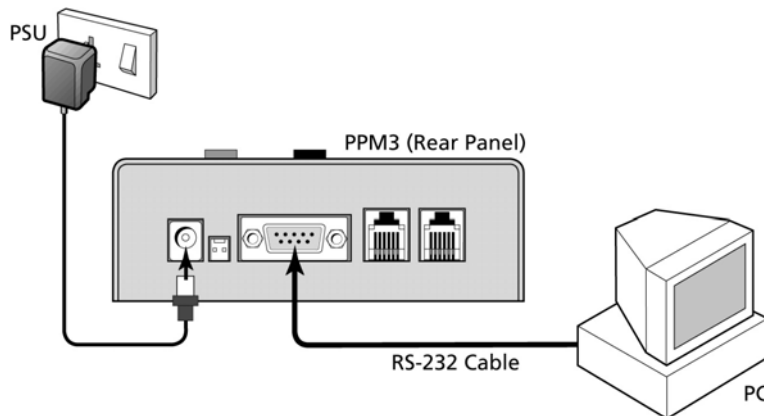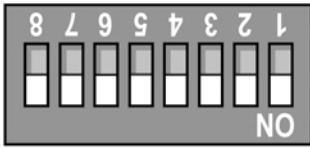
Fig. 3.4.2 DIP Switch Settings for RS-232 Communications

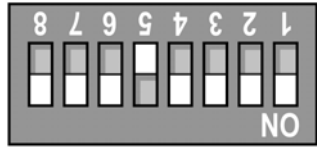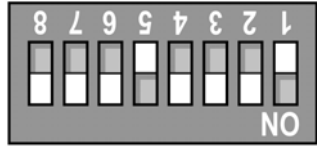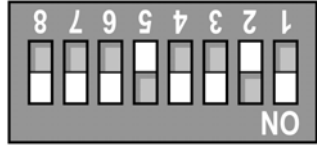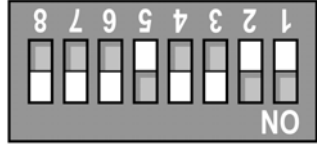| Mode	Node Address	DIP Switch Setting	Comment
RS-232	0		• **RS-232 Mode Enabled (factory default)** • Set DIP Switch 5 to the ON position • Programmer now uses the 9-way COM port on the rear panel of the module. • Communications address is set by DIP switches 1-4 as a binary weighted address in the range 0 – 15. • Factory Default address is 0 (i.e. DIP Switches 1-4 in the ON or '0' position).

For RS-232 operation (default setting):
- Set the RS-232 / RS-485 DIP Switch (5) to the ON (RS-232) position
- Set the Node Address to '0' (default address).
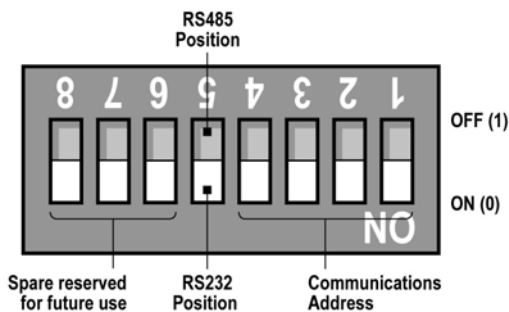- Connect the 9-way serial cable from the 9-way COM port on the PPM3 programmer to a spare COM port on your PC or other RS-232 Control System

3.5 RS-485 Operation

It is possible to network up to 16 x PPM3 programmers on an RS-485 network and control them from a single 'Supervisor PC' running a single session of the Equinox EQTools or ISP-PRO software. This scenario requires that each PPM3 programmer is set to 'RS-485' operation and that a unique 'Node Address' is set for each programmer. An RS-232 to RS-485 converter is also required to convert the RS-232 signal from the PC COM port to an optically isolated RS-485 signal suitable for interfacing with the PPM3's on the network.

The typical setup is shown in the figure below.

Fig. 3.5 Setup for RS-485 Communications (Multi channel)
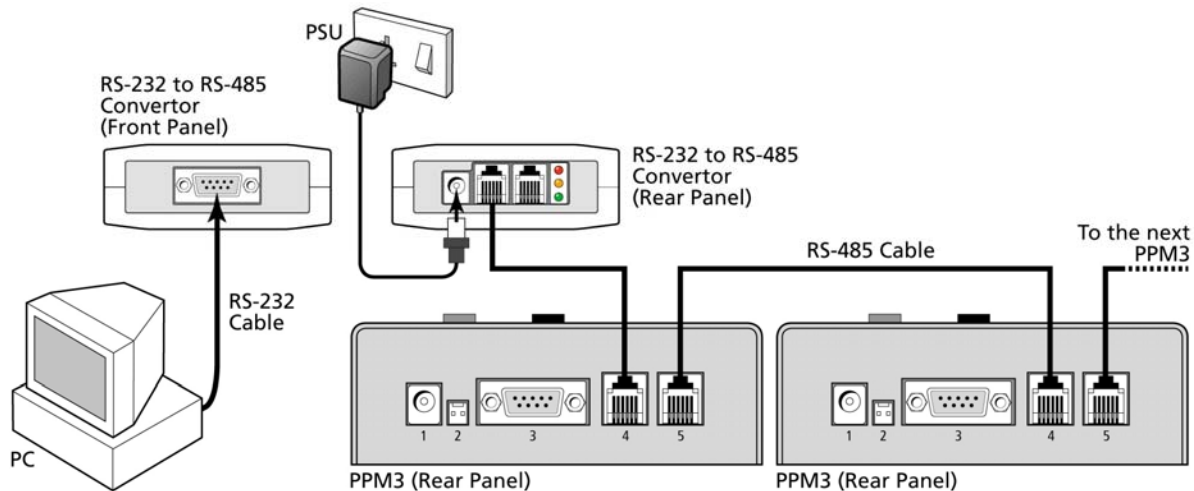
For RS-485 operation:
- Place the PPM3 programmers side-by-side on a flat surface with the rear panels all facing in the same direction as detailed in fig. 3.5
- For each PPM3 programmer, set the RS-232 / RS-485 DIP Switch (5) to the RS-485 position
- For each PPM3 programmer set the required Node Address in the range 0 – 15. (Each programmer must have unique address. It is customary to start have the first programmer set to address 0, the next programmer in the chain at address 1 and so on. See section 3.3 for further details.)
- Plug the 9-way RS-232 cable into the port on the RS-232 to RS-485 converter and connect the other end of the cable to a spare PC COM port.
- Plug an RS-485 network cable from one of the RS-485 outputs on the RS-485 converter to the 'RS-485 Input' connector (left-hand connector) on the rear panel of the first PPM3 Module.
- Plug another RS-485 network cable from the 'RS-485 Output' connector (right-hand connector) on the rear panel of the first PPM3 to the next PPM3 in the chain.
- A return RS-485 cable from the last PPM3 in the chain to the RS-485 converter is NOT required.
- Power up the PC.
- Plug in and switch on the power supply adaptor supplied with the RS-232/RS-485 convertor into a spare mains socket.
- Power up all PPM3 Programmers.
- Launch the ISP-PRO software on the PC and set up the required number of PPM's within the software.
- Test that all PPM3's are communicating with the PC at their correct 'Node Address'. (The Node Address may be changed while the programmer is powered up).

3.6 Serial Cables and adaptors

The programmer is supplied with the following PC Serial Cables / Adaptors:

1. **9-way D (female) to 9-way D (Male) Serial Cable**
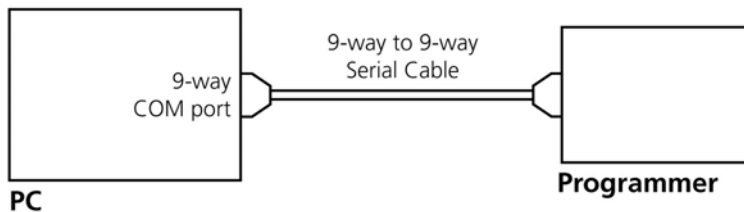
Programmer to 9-way PC Serial Port Connection

This cable should be connected between the 9-way D connector on the programmer and a spare 9-way RS-232 COM port on your PC.

2. **9-way D (female) to 25-way D (Female) adaptor**
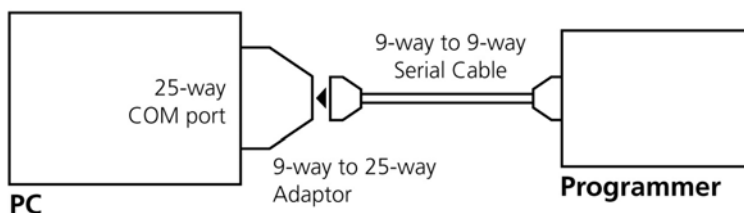
Programmer to 25-way PC Serial Port Connection

If your PC has a 25-way D connector, you will need to plug the 9-25way adaptor onto the end of the 9-9 way Serial Cable as shown in the diagram below.

3.7 RS-232 Serial Cable Specification

The serial cable supplied with the programmer is approximately 2m in length. If you wish to make your own longer cable, please ensure that you connect all 9 pins at each end in a straight-through pin-to-pin fashion. See fig. 3.7 for the pin-out specifications. The programmer features an ESD protected full RS-232 specification driver chipset and so should operate reliably when driving serial cables up to 10m in length. However, it is recommended to keep the serial cable as short as possible to enhance EMI immunity and reduce EMI emissions.

Important note:
The programmer uses ALL of the NINE pins of the COM port for different purposes. It is therefore imperative that the cable used to connect between the programmer and the PC COM port has all NINE cables connected. Failure to do this will result in the programmer not operating correctly.

3.8 RS-232 Serial Cable Pin-outs

Fig. 3.8.1 Pin-out RS-232 Communications Port

Pin No.	RS232 pin	Description
1	DCD	Controls the RESET pin of the programmer ATmega128 controller during the Firmware Update process. Jumper LK1 must be fitted to link DCD to the RESET pin.
2	RXD	Receive
3	TXD	Transmit
4	DTR	Same function as DCD.
5	0 Volt	0V
6	DSR	Not Connected
7	RTS	Used as Serial Clock signal during the Firmware Update process.
8	CTS	Connected to RXD
9	RING	Not Connected

Fig. 3.8.2 Connections for 9-way to 9-way Serial Cable (Straight through cable)

9-way (Male)	9-way (Female)
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

Fig. 3.8.3 Connections for 9-way to 25-way Adaptor

25-way (Female)	9-way (Male)
8	1
3	2
2	3
20	4
7	5
6	6
4	7
5	8
22	9

4.0 Programmer / Target System Power Supply Scenarios

4.1 Overview

The PPM3 Programmer Module can be powered in a number of different ways depending on the application. Please refer to the table below to choose the required 'Power Supply Scenario':

Fig. 4.1 Power supply scenarios

#	Power Supply Scenario	Refer to section
1	Programmer supplies Controlled Voltage to the Target System (@ user specified voltage and maximum current)	4.7
2	Target System powers the Programmer (@ 5.0V only)	4.8
3	Programmer and Target System are independently powered	4.9

Please note:

The favoured method of powering is for the programmer to power the Target System as using this method means that the sequencing of power is controlled from the Programming Project / Script. This allows the PPM3 Programmer to be a completely self-contained Programming System with no requirement for an external Target Power supply and the associated power sequencing system.

This section also details the actual power connectors, voltage range and earthing requirements of the programmer. Please see the table below for the relevant section reference.

Fig. 4.1 Power supply scenarios

#	Power Supply Scenario	Refer to section
1	Power Connectors – Rear Panel	4.2
2	DC Power Cable Specification	4.3
3	Voltage Range	4.4
4	Programmer / Target System Power Supply Schematic Overview	4.5
5	Earthing requirements	4.6

Electrostatic Discharge (ESD) Warning!
There are a number of electrostatic issues which must be taken into account to avoid damage to the programmer and/or the Target System during In-System Programming. Please see section 4.3.

4.2 Power Connectors (rear panel)

The PPM3 rear panel power connectors are shown in the figure below. The Jack Socket and 2-way Molex connectors are wired in parallel internally. The input voltage for both connectors is 9.0 to 12.0V DC regulated from an external power supply.

Fig. 3.2.1 PPM3 Module – Rear Panel Connectors
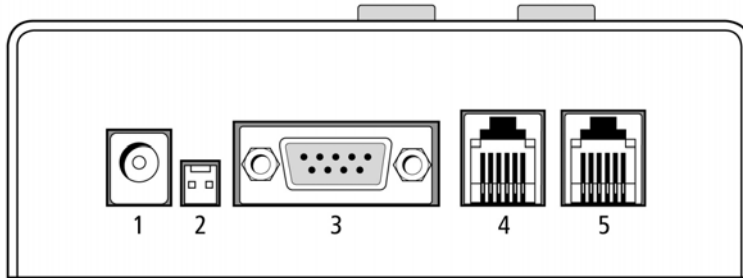
Key No	Function	Further description
1	DC Power Input (1)	• 2.5mm Jack socket (centre positive) • Input voltage: 9.0 – 12.0V DC regulated
2	DC Power Input (2)	• 2-way Molex connector • Input voltage: 9.0 – 12.0V DC regulated • This connector is wired internally in parallel with the jack socket input (1).

4.3 DC Power Cable Specification

The programmer is supplied with a DC power cable which plugs into the DC power connector jack socket on the rear panel of the programmer. The other end of this cable is supplied with bare wire ends for connecting to an external power supply. e.g. bench PSU.

Cable Specification:
- One end features a female 2.5mm DC jack plug
- The other end of the cable is supplied with bare wire ends for connecting to an external power supply. e.g. bench PSU.
- The black lead with the white stripe is the +VE cable.
- The all black lead is the GROUND cable.
- The external voltage applied to the Jack Socket connector must be between 9.0 and 12.0V DC regulated !

PPM3 Programmer Module

High-end Programming Module
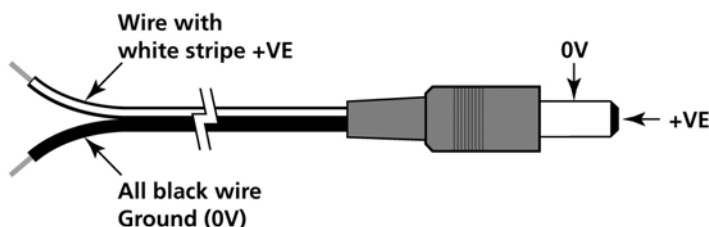
4.4 Voltage range

4.4.1 Programmer Output / Supply Voltage range

The PPM3 Module has been designed so that it can be powered from an external 9.0 to 12.0V DC regulated power supply via the jack socket (J6) or directly from a Target System running at +5.0V DC. The programmer is capable of generating a controlled supply voltage to the Target System of between 3.0V and 5.0V at 300mA, but this is only possible if the programmer is powered from an external 9.0 to 12.0V power supply,

Fig. 1.5.2 Programmer voltage supply range for reliable operation

#	Programmer mode	Min Vcc	Max Vcc	Units
1	**Programmer powering the Target System** • Programmer must be powered by external 9V DC supply via DC power connector (J6) or Molex power connector (J16) • Programmer is capable of generating a controlled supply voltage to the Target System of between 3.0V and 5.0V at 300mA • Target Sense functionality is only available in this mode.	3.00	5.00	V
2	**Target System powering the programmer** • Programmer circuitry must be powered at +5V • Target System must supply +5.0V via one of the ISP Headers to the programmer.	5.00	5.00	V

58   *PPM3 MKII Programmer - User Guide - V1.04 - 10th May 2008*

Downloaded from Elcodis.com electronic components distributor

4.4.2 Device Programming Voltage Range

The PPM3 programmer can program devices with a target voltage between the min. and max. values detailed in the table below. The programmer cannot currently support programming of devices below 3.V. Please contact Equinox directly for further assistance if your Target Voltage is < 3.0V.

#	Programmer mode	Min Vcc	Max Vcc	Units
1	Allowed voltage range for actual programming of devices	3.00	5.00	V

Important note:

To verify that a particular device can be programmed at a certain Vcc voltage:
- Check the supply / programming voltage parameters for the actual device (IC) in the manufacturer's datasheet.
- Many devices have two different voltage range versions eg. 2.7 – 6.0V and 4.0 – 6.0V. Please ensure that you are programming the device within the stated voltage range.
- Ensure that you have selected the required Target Vcc Voltage in your EQTools – Programming Project (*.ppm).
  e.g. if you are programming at +3.3V, you must set up this voltage in your Programming Project. This can be found in the EQTools - Project Builder <Target Vcc> tab.
- For most Atmel AVR and AT89S devices, the FLASH and EEPROM Write Times increase as the Target Voltage decreases. The timings are automatically set for you in your project as long as you have declared the correct Target Vcc voltage in your project.

4.5 Programmer / Target System Power Supply Schematic

The schematic shown in fig. 4.5 details the possible power supply scenarios for powering both the programmer and the Target System.

The key points to note are as follows:

- The programmer can be powered from an external power supply (9.0 – 12.0V DC regulated) via either the 'DC Jack Socket (J6)' or the 'DC Molex Connector (J15)'
- The programmer can power the Target System with a user-specified voltage from 3.0 – 5.0V via the 'Programmer Controlled Power Supply'.
- The programmer 'Line Driver Circuitry' must always be powered at the same voltage as the Target System. This ensures that the logic levels generated from the programmer are compatible with the logic levels on the Target System.
- The external link 'LK1' is a virtual link. It simply represents that the Vcc connection between the programmer and Target Vcc is broken.

fig. 3.2.1. Explanation of all Power Supply Circuit Schematic labels

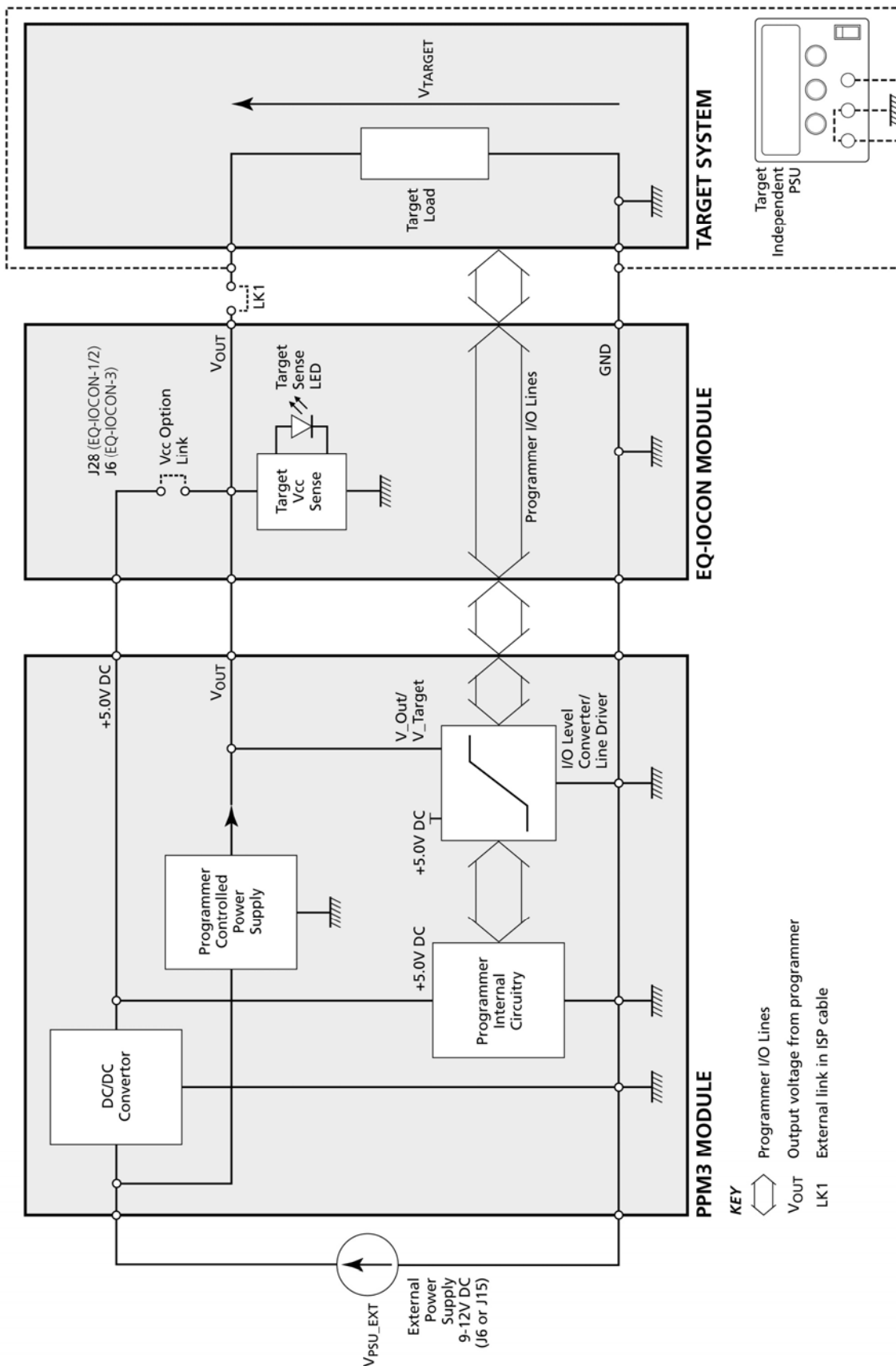| #	Item	Description
1	V_PSU_EXT	• This is the voltage applied from an external power supply (9.0 – 12.0V DC regulated) via either the 'DC Jack Socket (J6)' or the 'DC Molex Connector (J15)'
2	GND	• This is the common ground of the programmer which is connected to J6/J15 GROUND and also to the TARGET GROUND via one of the ISP Headers.
3	V_Target	• This is the voltage measured between the Vcc and Ground of the Target System.
4	LK1	• This is a virtual link (does not really exist) signifying the cable between the Programmer TVCC and the Target System Vcc. • If the programmer and Target System are independently powered, then LK1 must be not fitted i.e. the Target System Vcc wire should not be connected to the programmer TVCC connection.
4	I/O Level Converter / Driver Circuit	• This is the circuitry which translates the 5.0V I/O signals from the programmer 'Supervisor Microcontroller' to the Target System voltage levels. • The supply to the 'I/O Level Converters' must be at the same voltage as the Target System. I.e. if the Target System is powered at 3.3V, then the 'I/O Level Converters' must also be powered at 3.3V.
5	Target Load	• This is the load presented to the programmer across the PROG_VCC and GROUND terminals of any ISP Header.
6	Bench Power Supply	• This simply indicates a stable DC power supply with a current limit facility.

Fig 4.5 – Programmer / Target System Power Supply Overview Schematic

4.6 Earthing requirements

4.6.1 Overview

When setting up the programmer to In-System Program (ISP) a device on a Target System, extreme care must be taken to ensure that the 0V of the PC, programmer, Target System and any external devices are at the same 'earth potential'. If you are using the programmer with a laptop computer, please read section 3.2.2 carefully.

ESD Points to remember

Failure to follow correct ESD procedures when using this product could lead to damage to the programmer and/or the Target System.

- Wear an earthed wrist strap when handling either the programmer and/or any programmable device.
- Ensure that the PC, programmer, Target System and any external devices are all connected to the same GROUND potential.

4.6.2 Laptop earthing issues

The Problem:
Most laptops use an external double-insulated mains power supply which is not earthed to mains earth. This means that the laptop earth is likely to be floating at some voltage other than 0V. If the programmer is attached to the PC COM port with the PC powered ON and then the ISP lead is plugged into an earthed Target System, then the PC GROUND will discharge through the COM port GROUND, through the programmer into the Target System. This ESD may damage both the programmer and the Target System !

The Solution
- Ensure that the laptop is powered OFF when connecting the ISP cable from the programmer to an earthed Target system
- Use the programmer in Standalone Mode if possible (ie. not connected to a PC)

4.6.3 Desktop PC earthing issues

The Problem:
The chassis on desktop PC must be connected to mains earth to comply with Health and Safety legislation. If the chassis is not earthed for some reason, the PC power supply GROUND will be floating and it is then possible to discharge a high level of energy stored in the power supply mains filter through the programmer into the Target System. This ESD may damage both the programmer and the Target System !

The solution:
- Ensure that the PC chassis is properly earthed to mains earth.
- Connect an earthing bond wire from the PC chassis to the GROUND connection of either the programmer or the Target system / Test Fixture.

4.6.4 Earthing recommendations

To avoid catastrophic damage to PC, programmer or target system:
- Ensure that both your target system and PC are connected to a common earth point
- Make sure that all interconnections are made before applying power to PC and target system

1 Earth connection to PC
2 Target system earth
3 Earth from external device

Common earthing point
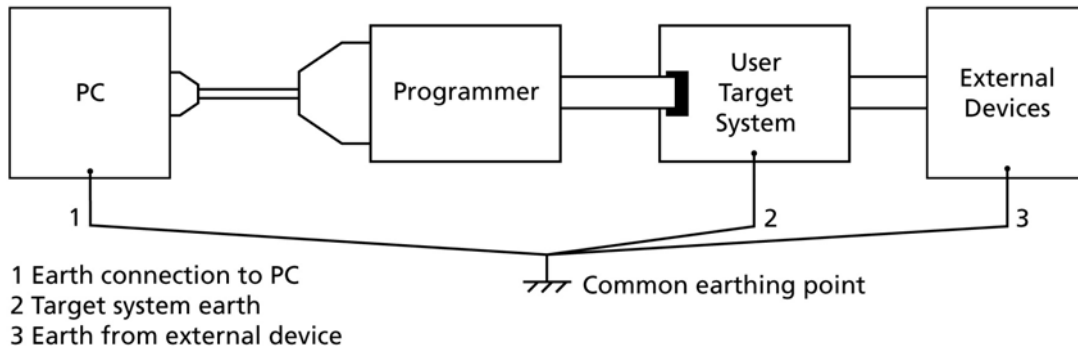
- If you are using a laptop or PC which is not connected to mains earth, it is recommended that you make a hard-wired connection from the COM port D-connector shell and the target system to a common earth point
- Avoid plugging and unplugging ISP connector while either the PC or Target System is powered up
- Please ensure that any devices connected to the user target system are also properly grounded to the same common earth point

Warning!
Any damage caused to the programmer by Electrostatic Discharge (ESD) through inadequate earthing is not covered under the warranty of the product.

4.7 Programmer Powers the Target System

4.7.1 Overview

The PPM3 programmer is capable of supplying a controlled voltage from 3.0 to 5.0V to power the Target System at a maximum current of 300mA. This allows the programmer to power the Target System via the ISP Cable and switch power on / off under Programming Project / Script control. The voltage and current settings must be set up in the Programming Project prior to powering up the Target System. In this mode it is also possible for the programmer to automatically sense whether a Target System is connected to the programmer – see Target Sensing section.

Fig. 4.7.1 Programmer powering the Target System
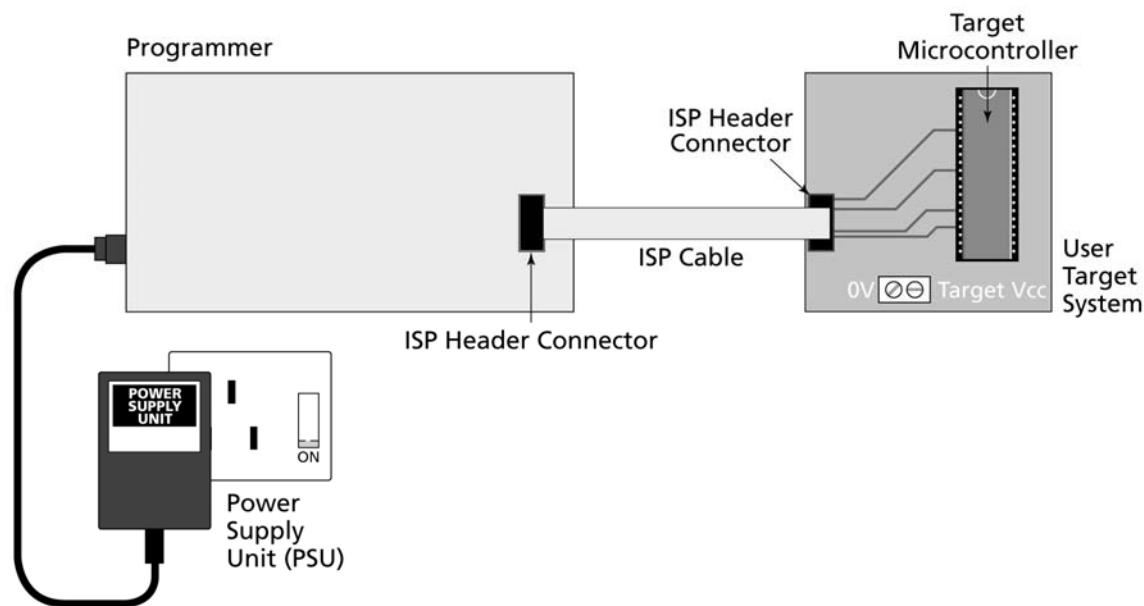
fig. 4.7.2 Settings for the programmer powering the Target System

#	Parameter	Description
1	Programmer Input voltage source	• External 9.0 – 12.0 V DC regulated supply required • Connect power supply to either DC Jack Socket or 2-way Molex Power Connector on the rear panel of the programmer.
2	V_Target (Output voltage to Target System)	• Programmer outputs a Controlled Voltage to the Target System: 3.0 – 5.0V DC @ 300mA max • Voltage / current setting must be set up in the Programming Project
3	Vcc Option Link	• Not Fitted
4	LK1 Link	• Fitted • Programmer Vcc signal should be connected to the Target Vcc
5	Programmer I/O Level Converters	• Programmer I/O Level Converter is powered at the Controlled Voltage which is the same as the Target System voltage.

4.7.2 Instructions

The instructions below detail how to set up the programmer to supply a controlled voltage to the Target System.

1. Create a Programming Project which powers the Target System

- In EQTools – Project Builder – Select the *<Target Power Supply Settings>* tab
- Set the required *'Target Voltage'* and *'Tolerance'* eg. 3.3V +/- 500mV.
- The Voltage *'Stabilise Time'* should be set to a long enough time to allow the voltage to reach its final voltage (default: 200ms).
- Under the *'Programmer Controlled Power Supply'* section, select: *<Programmer controlled Target Power Supply: ON>*

Programmer Controlled Power Supply

Programmer controlled Target Power Supply: ON
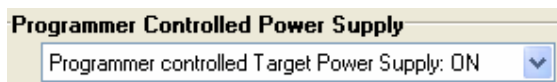
→ This enables the programmer to supply power to the Target System.
- Enter the maximum current which your Target System could take from the programmer supply (worst case i.e. current inrush condition)
- Leave all other settings as the default values
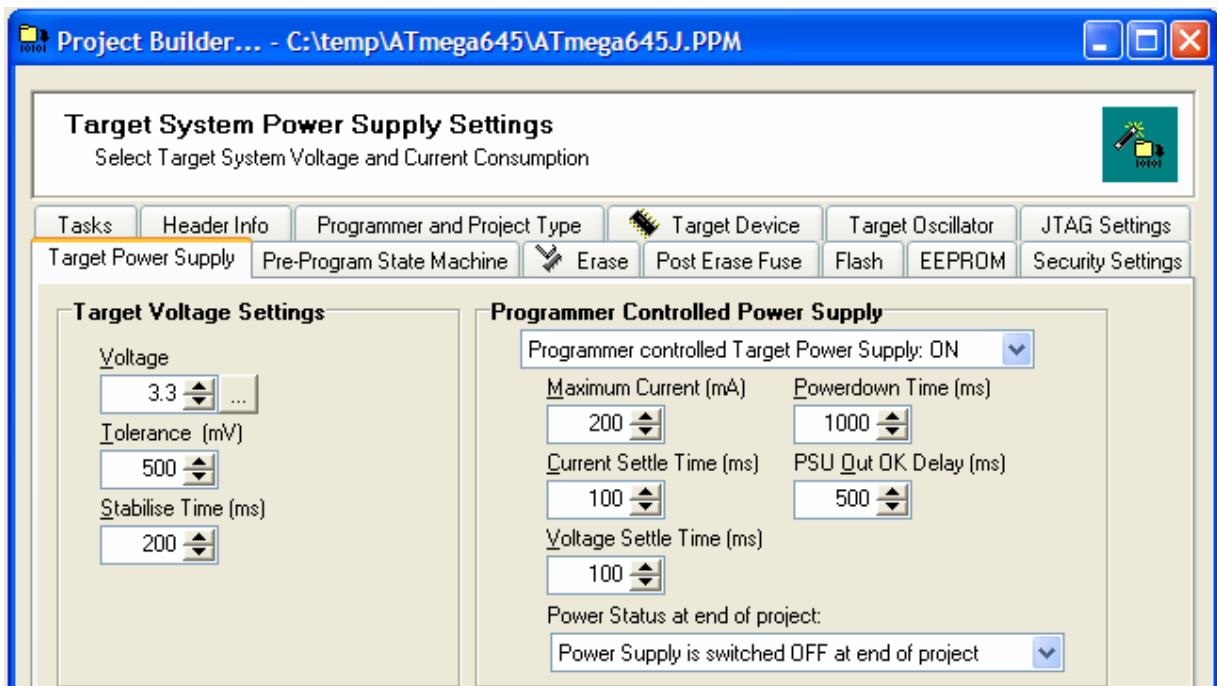
2. Test the Project in Equinox Development Suite (Target System disconnected)

- Launch the Project in EDS (Development Mode)
- Select the **<Target Power Supply>** tab
  → the Power Supply settings are now displayed

- With your Target System **DISCONNECTED**, place a Volt Meter across the Target Vcc (TVCC) and GROUND terminals of the programmer (on the I/O Connector Module)
- Click the **<Power Up>** button in EDS
  → the programmer **<BUSY>** LED should now illuminate
- The programmer should now output the specified **'Target Voltage'** on the **'TVCC'** pin of the programmer IO-CON module.
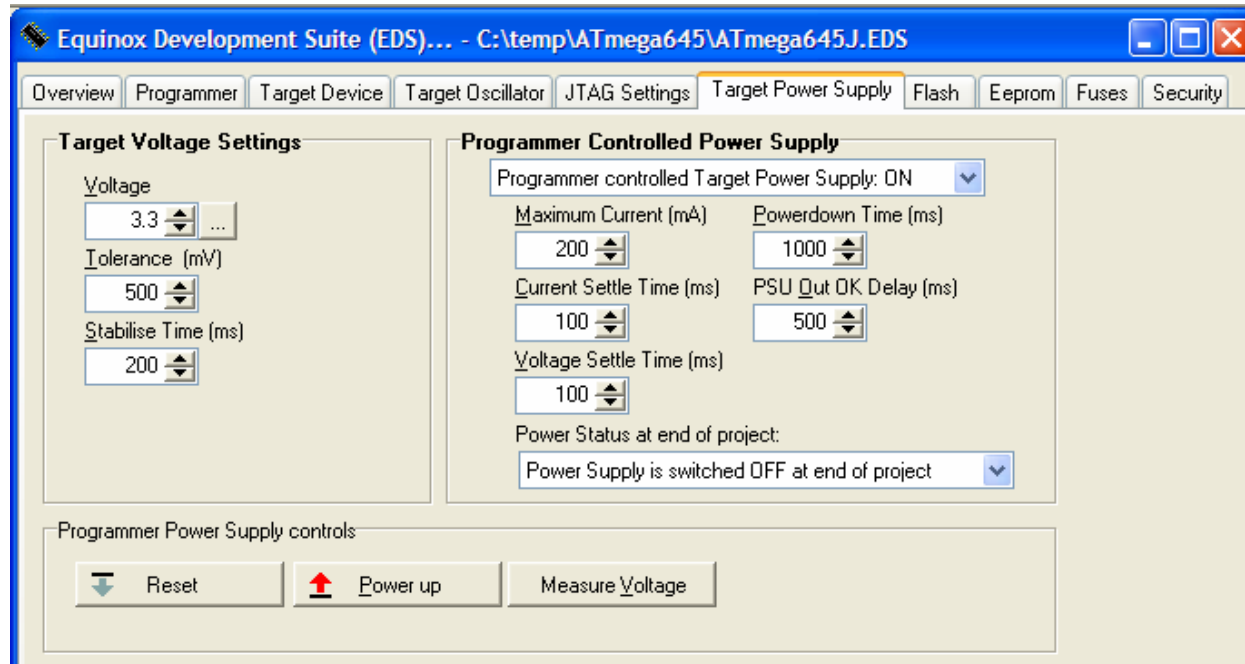 → the **'Target Vcc'** LED on the I/O Connector Module should now illuminate
- Measure the voltage across the Target Vcc **'TVCC'** and GROUND terminals of the programmer.
- If the voltage is within the limits specified in the project, click **<Power Down>** to remove power to the Target System and go to step (3).
- If the voltage is outside the limits specified in the project, adjust the tolerance settings and then repeat step (2).

3. Connect the Target System to the programmer

- Connect the Target System to the programmer via an ISP Cable, Fast Connect connectors or wire-wrap connectors.
- Remove the **'Vcc Option Link'** on the I/O Connector Module (labelled J6 or J28).
- Check that you have a zero ohm connection between **the Target Vcc** (Vout) of the programmer and the Vcc line of the Target System.
- Using a DVM, check that there are no shorts between the **Target_Vcc** and Target Ground

4. **Test the Project in Equinox Development Suite (Target System connected)**
   - Launch the Project in Equinox Development Suite
   - With your Target System **<u>CONNECTED</u>**, place a Volt Meter across the *Target Vcc* and GROUND terminals of the programmer (on the I/O Connector Module)
   - Click the ***<Power Up>*** button in EDS → the programmer ***<BUSY>*** LED should now illuminate
   - If the programmer reports an error on the LCD and the RED LED is flashing,
   - Measure the voltage across the *Target Vcc* and GROUND terminals of the programmer.
   - If required, measure the voltage across the *Target_Vcc* and GROUND on the actual Target System

4.8 Target System powers the Programmer (5.0V only)

4.8.1 Overview

In this scenario, the programmer is powered from the Target System via the ISP cable. The Target Voltage must be +5.0V for the programmer to operate correctly. This mode is not recommended for general use as the programmer is powered on/off every time the Target System is powered on/off. This can lead to over-voltage and ESD problems. It also makes the programming times longer as the programmer must run its internal self-check algorithm every time it is powered up.

Fig. 4.8.1 Target System powers the programmer at 5.0V
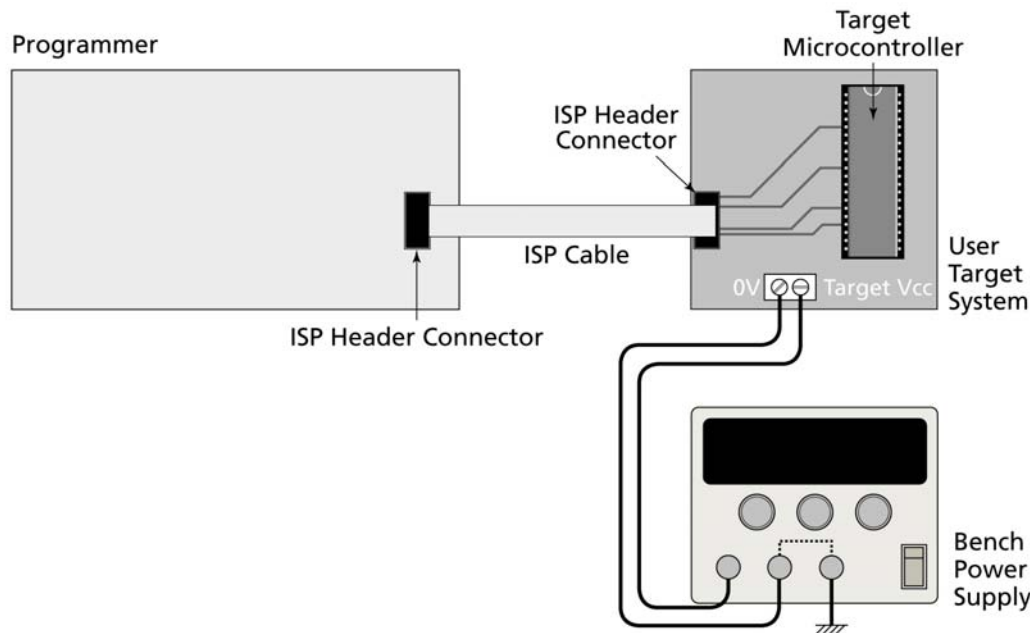
Please note
- The external voltage applied to the programmer must be between 5.0V!
- The Programming Project must specify a Target Voltage of +5.0V.
- The Target System must be able to supply the worst case current required during a programming operation.
- A 6V2 5W zener diode is internally connected across the programmer supply to protect against over-voltage.

fig. 4.8.2 Settings for Target System powering the programmer

#	Parameter	Description
1	Programmer Input voltage source	• None
2	V_Target (Output voltage to Target System)	• None • The Target System must supply +5.0V to the programmer. • The Programming Project must specify a Target Voltage of +5.0V.
3	Vcc Option Link	• Not Fitted
4	LK1 Link	• Fitted • Programmer Vcc signal should be connected to the Target Vcc
5	Programmer I/O Level Converters	• Programmer 'I/O Level Converter' is powered by the Target System voltage. • E.g. if the Target Voltage is 3.3V, then the Level Converter Voltage is also 3.3V.

Explanation of operation:
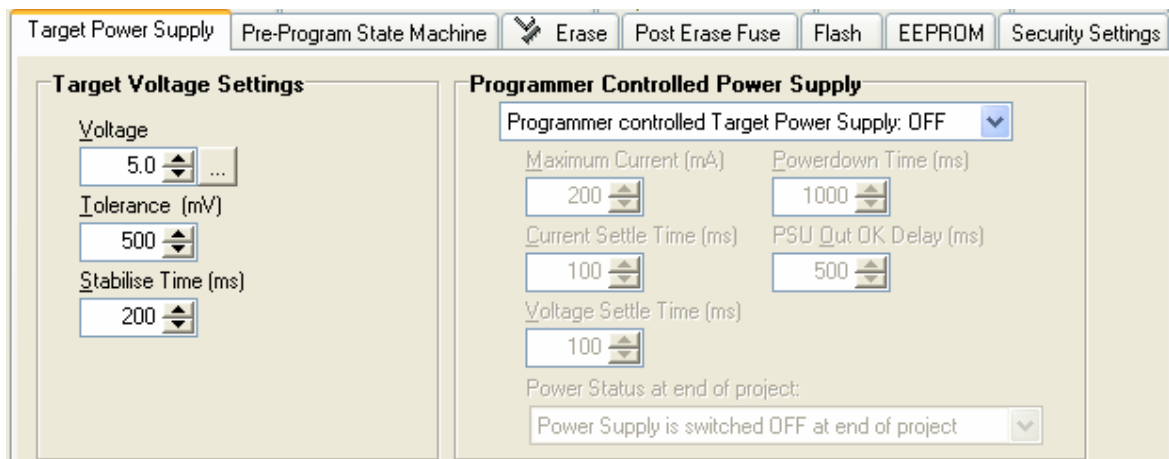- When the ISP Cable is connected from the programmer to a Target System and the Target system power supply is switched ON, the programmer will power up.
- When the ISP Cable is disconnected from the Target System or the Target System is powered OFF, the Programmer will power OFF.
- The programmer should only be connected/disconnect from the Target system when the Target power supply is switched OFF

4.8.2 Instructions

The instructions below detail how to set up the programmer to be powered at +5.0V from the Target System.

1. **Create a Programming Project which does NOT power the Target System**
   - In EQTools – Project Builder – Select the <Target Power Supply Settings> tab
   - Set the 'Target Voltage' to +5.0V with a tolerance of +/- 500mV.
   - Set the 'Programmer Controlled Power Supply' to *<Programmer controlled Target Power Supply: OFF>*
   - Leave all other settings as the default values.

| Target Power Supply | Pre-Program State Machine | Erase | Post Erase Fuse | Flash | EEPROM | Security Settings |

Target Voltage Settings

Voltage
5.0 ...

Tolerance (mV)
500

Stabilise Time (ms)
200

Programmer Controlled Power Supply

Programmer controlled Target Power Supply: OFF

Maximum Current (mA)
200

Powerdown Time (ms)
1000

Current Settle Time (ms)
100

PSU Out OK Delay (ms)
500

Voltage Settle Time (ms)
100

Power Status at end of project:
Power Supply is switched OFF at end of project

2. **Connect the programmer to the Target System**
   - Connect the Target System to the programmer via an ISP Cable, Fast Connect connectors or wire-wrap connectors.
   - Remove the *'Vcc Option Link'* on the I/O Connector Module (labelled J6 or J28).
   - Check that you have a zero ohm connection between the **Target Vcc** (Vout) of the programmer and the Vcc line of the Target System.
   - Using a DVM, check that there are no shorts between the **Target_Vcc** and Target Ground

3. **Typical programming iteration**
   - Power up the Target System → the programmer powers up
   - Using the keypad / display, select the required Programming Project
   - Press *<Yes>* key to execute Programming Project
   - Power down the programmer / Target System

4.9 Programmer and Target System are independently powered

4.9.1 Overview

In this scenario, the programmer is powered from a completely independent supply to the Target System. The programmer must be powered via one of the *'DC Power Connectors'* on the rear panel at 9.0 – 12.0V. The Target System is powered from an external power supply in the range of 3.0 to 5.0V. The *Programmer_Vcc* output must **NOT** be connected to the *Target_Vcc* line in this scenario.

Fig. 4.9.1 Programmer and Target System are independently powered
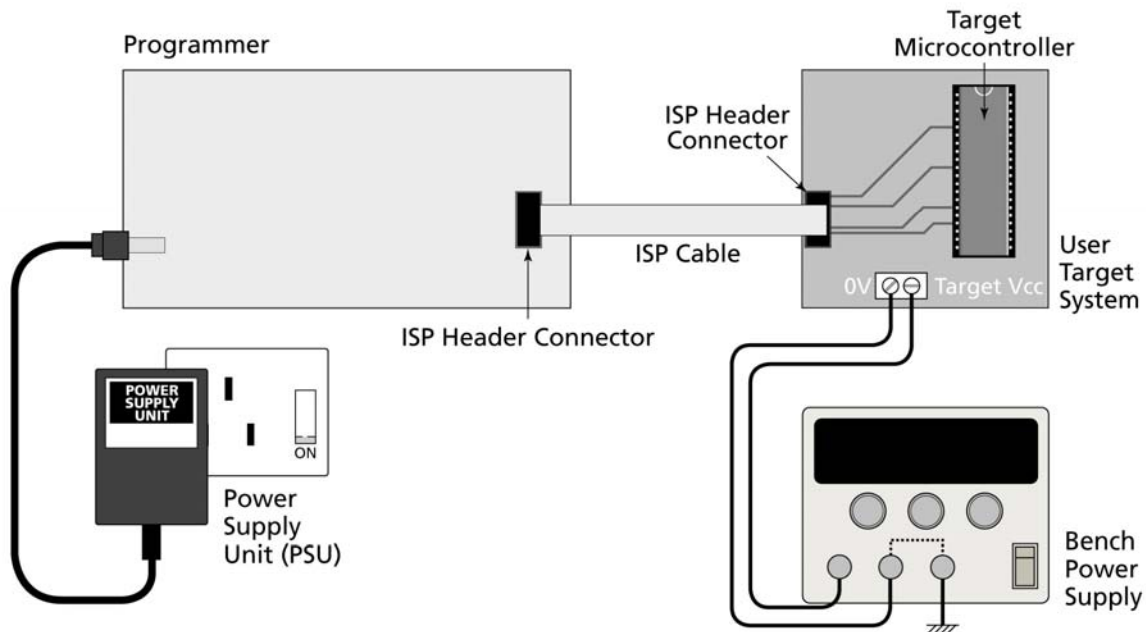
Explanation of operation:

- The programmer is always powered ON
- The *'Vcc Option Link'* is removed
- *LK1 connection* is removed (ie. Programmer and Target Vcc are disconnected)
- The programmer must supply the same voltage as the *Target_Vcc* to the Programmer *'I/O Level Converter Circuitry'*.
- The Target Power Supply should be switched OFF before the ISP cable is plugged into it
- The programmer does not draw any current from the Target System during programming
- When programming is complete, the Target System should be powered OFF and then the ISP cable removed

Warning!
- The voltage applied to the programmer via the ISP cable must be between 3.0 and 5.0V as this voltage is used to drive the programmer I/O Stage. A 6V2 5W zener diode is internally connected across the programmer supply to protect against over-voltage.
- The Programmer and Target Vcc must be physically disconnected by cutting the Vcc signal in the ISP cable if necessary.

Fig. 4.9.1 Settings - Programmer and Target System are independently powered

#	Paramater	Description
1	Programmer Input voltage source	• External 9.0 – 12.0 V DC regulated supply via external power supply. • Connect power supply to either DC Jack Socket or 2-way Molex Power Connector on the rear panel of the programmer.
2	V_Target (Output voltage to Target System)	• Programmer must be set up to output the same voltage to the programmer 'Line Driver Circuitry' as the Target Voltage. • This voltage is ONLY applied to the programmer 'Internal Line Converter Circuitry', not to the Target System ! • This ensures that the programmer I/O drivers are interfacing to the same voltage levels as on the Target System. • The Voltage / current setting must be set up in the Programming Project
3	Vcc Option Link	• Not Fitted
4	LK1 Link	• Not Fitted • Programmer Vcc signal **should NOT be connected to the Target Vcc** • This may involve physically cutting the Vcc signal line in the ISP cable.
5	Programmer I/O Level Converters	• Programmer I/O Level Converter is powered at the Controlled Voltage which is the same as the Target System voltage. • Eg. if the Target Voltage is 3.3V, then the Level Converter Voltage is also 3.3V.

4.9.2 Instructions

The instructions below detail how to set up the programmer when the Target System is independently powered from the Target System.

Example:
- The Target System is powered at 3.3V from an *'Independent Power Supply'*.
- The programmer 'Line Driver Circuitry' must also be powered at 3.3V using **the 'Programmer controlled Target Power Supply'**
- The Programmer TVCC and Target System Vcc are not physically connected

Here is how to set up the programmer:

1. **Create a Programming Project which powers the programmer 'I/O Level Converters'**
   - In EQTools – Project Builder – Select the *<Target Power Supply>* tab
   - Set the required *'Target Voltage'* and tolerance eg. 3.3V +/- 500mV.
   - The Voltage *'Stabilise Time'* should be set to a long enough time to allow the voltage to reach its final voltage (default: 200ms).
   - Set the *'Programmer controlled Power Supply'* to *'ON'*
     → This enables the programmer to supply power to the programmer 'I/O Level Converters' only.
   - The Target System is **NOT** powered by this voltage as the *Programmer_Vcc* and *Target Vcc* are physically disconnected.
   - Leave all other settings as the default values.
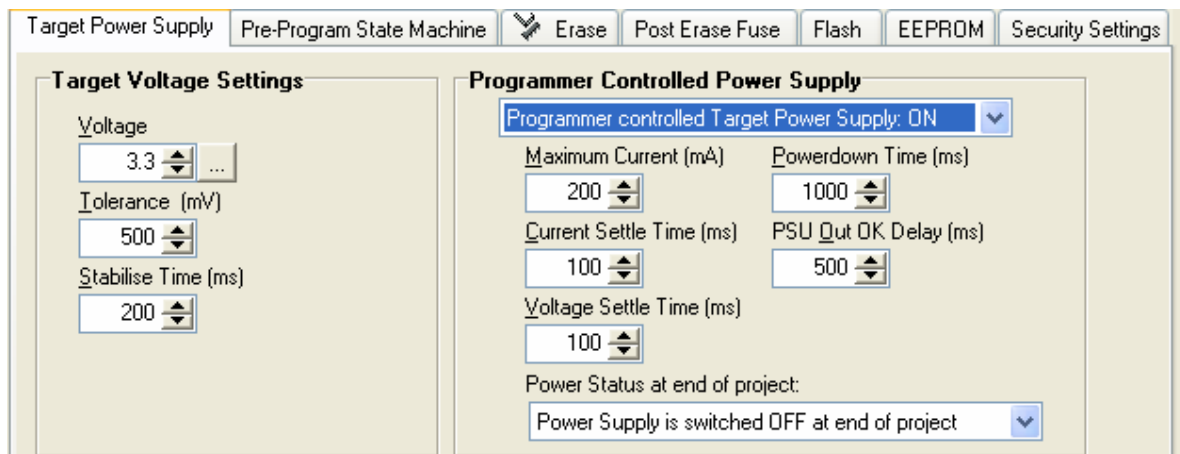
2. **Connect the Target System to the programmer**
   - Connect the Target System to the programmer via an ISP Cable, Fast Connect connectors or wire-wrap connectors.
   - Make sure that **the Programmer_Vcc is NOT connected to the Target_Vcc**. This may involve physically cutting the Vcc signal in the ISP cable if you are using one of the ISP headers to connect to the Target System. This is shown as a virtual link called *'LK1'* on the programmer schematic – see section 4.4.
   - Remove the *'Vcc Option Link'* on the I/O Connector Module (labelled J6 or J28).
   - Using a DVM, check that there are no shorts between the *Target_Vcc* and *Target Ground*

3. **Test the Project in Equinox Development Suite (Target System connected)**
   - Launch the Project in Equinox Development Suite
   - With your Target System **CONNECTED**, place a Volt Meter across the Target Vcc and GROUND terminals of the programmer (on the I/O Connector Module)
   - Click the *<Power Up>* button in EDS
     → the programmer *<BUSY>* LED should now illuminate

   - Measure the voltage across the *Target_Vcc* and *GROUND* terminals of the programmer.
   - Power up the Target System
   - Measure the voltage across the *Target_Vcc* and *GROUND* on the actual Target System
   - Compare the voltage measured across the programmer output and the Target_Vcc. These voltages should be approximately the same.
   - Try a programming operation such as *<Check Signature>*