



# 4D SYSTEMS

*TURNING TECHNOLOGY INTO ART*

## **gen4 Raspberry Pi Display Module Kits** **All gen4 sizes (Picaso and Diablo16)**

**Compatible with:**

**2.4", 2.8", 3.2", 3.5", 4.3", 5.0" and 7.0" Picaso and  
Diablo16 gen4 Display Modules**

**DATASHEET**

**Document Date: 27<sup>th</sup> January 2016**

**Document Revision: 1.0**

# Contents

---

1. Description .....	3
2. Hardware Connection Overview .....	4
3. Advanced Hardware Options .....	5
4. Hardware Revision History .....	6
5. Legal Notice .....	7
6. Contact Information .....	7

## 1. Description

This datasheet covers all -PI kits (Raspberry Pi Kits) for the gen4 range of Intelligent Display modules, for all sizes. It is an overview only, and specific information about the gen4 Display Module or the parts in the kit, can be found in their individual datasheets.

A Raspberry Pi kit is a normal gen4 display module, such as a gen4-uLCD-32DT, which has been kitted with Serial Pi Adaptor Shield and a few other accessories (listed below), to make connection to a Raspberry Pi board simple.

In a typical -PI kit, you will have the following parts:

- Gen4 Intelligent Display Module
- Serial Pi Adaptor Shield
- 30-way FFC cable
- Gen4-IB (Interface Board)
- 5-way Interface Cable

Starter Kits are also available, which include a microSD card and Programming Adaptor, which extend the functionality of the kit significantly.

The Serial Pi Adaptor is a simple shield which fits directly on top of your favourite Raspberry Pi board, and provides a simple 5-way interface to the 4D display module.

The 30-way FFC cable is used to connect the gen4 Display Module to the gen4-IB.

The 5-way Interface Cable is used to connect the gen4-IB to the Serial Pi Adaptor.

The gen4 display module can be configured in a number of ways using the 4D Systems Workshop4 IDE. There are 4 different environments available in the Workshop4 IDE, allowing you to have the gen4 Display Module as a slave which takes commands from your Raspberry Pi, through to programming the display module directly and designing advanced graphics which interface to the Raspberry Pi.

Communication to the Display Module is performed via the Raspberry Pi's serial UART port.

Power for the display is supplied from the Raspberry Pi's 5V bus. Please note that some of the larger gen4 display modules can draw over 500mA of power, some up to 800mA depending on the model.

As shown on the next page, the gen4-IB board is a simple interface which converts the 30-way FFC cable into the 5 signals used for the Raspberry Pi interface. The remaining signals from the gen4 display module are not accessible through this interface board.

A gen4-PA (Programming Adaptor) is included with the Starter Kits (SK). This is a USB programmer for the gen4 range of display modules that additionally breaks out all of the signals from the display.

If you have existing 4D Products, and have a uUSB-PA5 or uUSB-PA5-II, or even a 4D Programming Cable, then you can use these programming devices via the gen4-IB board supplied with this kit, and program the gen4 display module directly, without the need of a gen4-PA board.

For a detailed listing of the capabilities of the display module in this Raspberry Pi Pack, please refer to the datasheet for the display itself, available from the 4D Systems website, [www.4dsystems.com.au](http://www.4dsystems.com.au)

## 2. Hardware Connection Overview

The following pictures illustrate how to connect the parts of the PI kit together. Note, the display module illustrated below is the gen4-uLCD-24DT. The display module can vary depending on the specific kit you are using.



Figure 1. Components in a typical gen4 PI Kit.



Figure 2. Components in the PI Kit connected together. Serial Pi Adaptor Shield shown then connects on top of your Raspberry Pi board (not included).

### 3. Advanced Hardware Options

The display module (gen-uLCD-xx) is a very capable and powerful piece of hardware, which can be reconfigured beyond the boundaries of the Raspberry Pi.

With the use of the 4D Systems Workshop4 IDE Software, the display module can be configured and programmed independently of the Raspberry Pi.

If you wish to explore the capabilities of the gen4 display module and reconfigure the display module, please refer to the Datasheet for your specific display module. Datasheets are available from the 4D systems website, [www.4dsystems.com.au](http://www.4dsystems.com.au). The display module specific datasheets include a lot more information than what is discussed in this datasheet.

You can quickly and easily reset your module to default factory firmware/settings by using the Workshop4 IDE Software.

To take advantage of the 4D Systems Workshop4 Software, a 4D Programming Adaptor is required. A Programming Adaptor is available on the 4D Systems website, or from your local distributor. Programming Adaptors are also included in all Starter Kits.

The Workshop4 IDE enables the display module to be configured using 4 different environments.

- The **Designer** environment enables the user to write 4DGL code in its natural form to program the display module.
- A visual programming experience, suitably called **ViSi**, enables click-and-place type placement of objects to assist with 4DGL code generation and allows the user to visualise how the display will look while being developed.
- An advanced environment called **ViSi-Genie** doesn't require any 4DGL coding at all, it is all done automatically for you. Simply lay the display out with the objects you want, set the events to drive them and the code is written for you automatically. ViSi-Genie is a rapid development platform, which can be extended further with the purchase of a Workshop4 Pro license, available from the 4D Systems website.
- A **Serial** environment is also provided to transform the display module into a slave serial module, allowing the user to control the display from any host microcontroller or device with a serial port. This is the default configuration when the display module is shipped.

**Designer** and **ViSi** allow you to program and write your own protocol to communicate between the Raspberry Pi and the Display Module, and write 4DGL code directly into the display module to create advanced GUI's and graphics.

**ViSi-Genie** has a set protocol as outlined in the documentation associated with it (Available from the Workshop4 Product page of the website), and 4D Systems has developed a Linux Library specifically for the ViSi-Genie environment, to enable a rapid development platform. The library is available for download from the gen4 display modules PI product page on the 4D Systems Website, [www.4dsystems.com.au](http://www.4dsystems.com.au).

The **Serial** environment loads an application (SPE) onto the display module which enables commands from the Raspberry Pi to communicate directly with the display module and get it to perform specific actions. 4D Systems has developed a library specifically for this environment also, which is available for download from the gen4 display modules PI product page on the 4D Systems Website, [www.4dsystems.com.au](http://www.4dsystems.com.au).



## 5. Legal Notice

### Proprietary Information

The information contained in this document is the property of 4D Systems Pty. Ltd. and may be the subject of patents pending or granted, and must not be copied or disclosed without prior written permission.

4D Systems endeavours to ensure that the information in this document is correct and fairly stated but does not accept liability for any error or omission. The development of 4D Systems products and services is continuous and published information may not be up to date. It is important to check the current position with 4D Systems. 4D Systems reserves the right to modify, update or makes changes to Specifications or written material without prior notice at any time.

All trademarks belong to their respective owners and are recognised and acknowledged.

### Disclaimer of Warranties & Limitation of Liability

4D Systems makes no warranty, either expressed or implied with respect to any product, and specifically disclaims all other warranties, including, without limitation, warranties for merchantability, non-infringement and fitness for any particular purpose.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications.

Images and graphics used throughout this document are for illustrative purposes only. All images and graphics used are possible to be displayed on the 4D Systems range of products, however the quality may vary.

In no event shall 4D Systems be liable to the buyer or to any third party for any indirect, incidental, special, consequential, punitive or exemplary damages (including without limitation lost profits, lost savings, or loss of business opportunity) arising out of or relating to any product or service provided or to be provided by 4D Systems, or the use or inability to use the same, even if 4D Systems has been advised of the possibility of such damages.

4D Systems products are not fault tolerant nor designed, manufactured or intended for use or resale as on line control equipment in hazardous environments requiring fail – safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of the product could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). 4D Systems and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

Use of 4D Systems' products and devices in 'High Risk Activities' and in any other application is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless 4D Systems from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any 4D Systems intellectual property rights.

## 6. Contact Information

For Technical Support: [www.4dsystems.com.au/support](http://www.4dsystems.com.au/support)

For Sales Support: [sales@4dsystems.com.au](mailto:sales@4dsystems.com.au)

Website: [www.4dsystems.com.au](http://www.4dsystems.com.au)

Copyright 4D Systems Pty. Ltd. 2000-2016.



Компания «ЭлектроПласт» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Оперативные поставки широкого спектра электронных компонентов отечественного и импортного производства напрямую от производителей и с крупнейших мировых складов;
- Поставка более 17-ти миллионов наименований электронных компонентов;
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- Лицензия ФСБ на осуществление работ с использованием сведений, составляющих государственную тайну;
- Поставка специализированных компонентов (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Aeroflex, Peregrine, Syfer, Eurofarad, Texas Instrument, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Помимо этого, одним из направлений компании «ЭлектроПласт» является направление «Источники питания». Мы предлагаем Вам помощь Конструкторского отдела:

- Подбор оптимального решения, техническое обоснование при выборе компонента;
- Подбор аналогов;
- Консультации по применению компонента;
- Поставка образцов и прототипов;
- Техническая поддержка проекта;
- Защита от снятия компонента с производства.



#### Как с нами связаться

**Телефон:** 8 (812) 309 58 32 (многоканальный)

**Факс:** 8 (812) 320-02-42

**Электронная почта:** [org@eplast1.ru](mailto:org@eplast1.ru)

**Адрес:** 198099, г. Санкт-Петербург, ул. Калинина, дом 2, корпус 4, литера А.