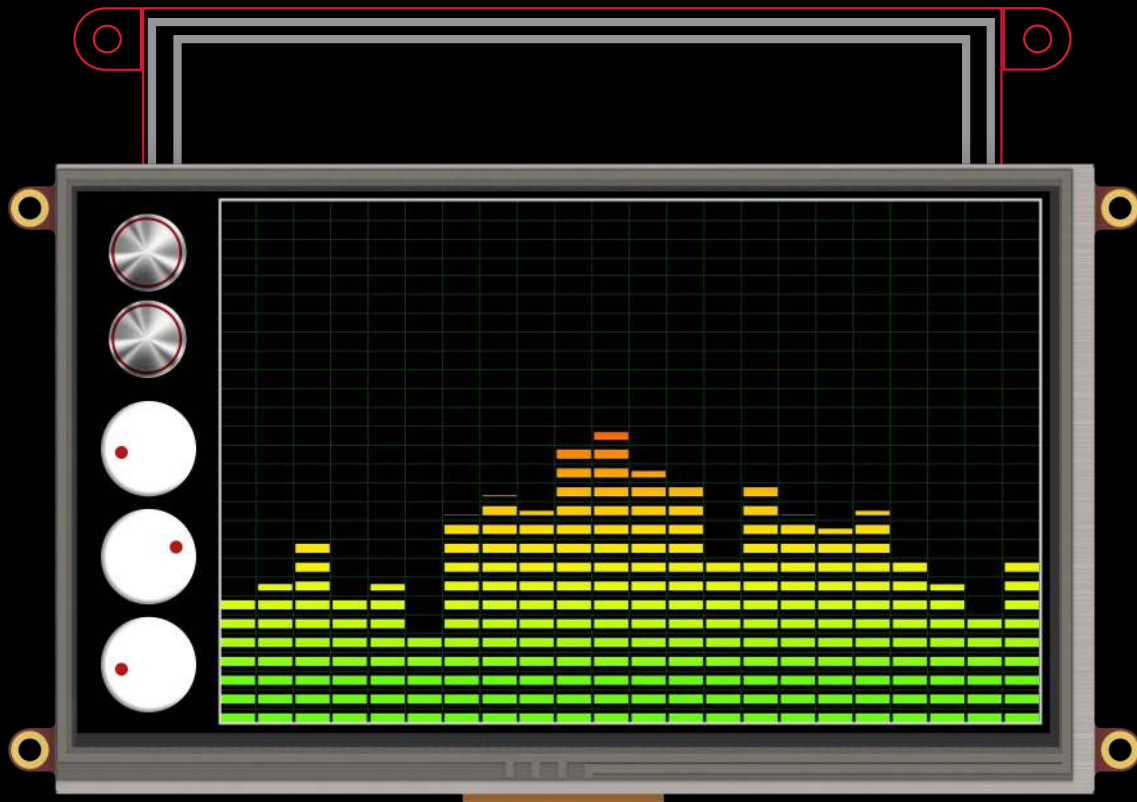


uLCD SERIES

uLCD-70DT

7.0" Intelligent Display Module

USER GUIDE



4D SYSTEMS

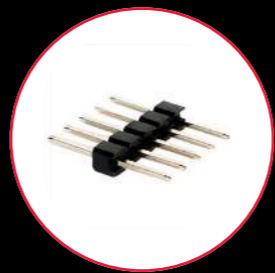
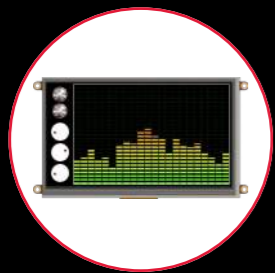
MAKING HUMAN INTELLIGENCE SMARTER

Contents

Introduction	4
System Requirements	5
1 Hardware	5
2 Software	6
Connecting The Display Module To The Pc	7
Connection Options	8
Option A - Using the 4D Programming Cable	8
Option B - Using the uUSB-PA5-II	8
Option C - Using the gen4-PA	8
Option D - Using the 4D-UPA	9
Getting Started With A Simple Project	10
Application Notes	15
Reference Documents	16
GLOSSARY	19
Hardware	19
Software	20

This user guide will help you started using the uLCD-70DT module along with the WorkShop4 IDE. It also includes a list of essential project examples and application notes.

What's In The Box





This User Guide is an introduction to the uLCD-70DT and the WorkShop4 software IDE associated with it. This manual should be treated only as a useful starting point and not as a comprehensive reference document. Refer to [Application Notes](#) for a list of all the detailed reference documents.

In this User Guide, we will briefly focus on the following topics:

- Hardware and Software Requirements
- Connecting the Display Module to your PC
- Getting Started with Simple Projects
- Application Notes
- Reference Documents

The **uLCD-70DT** is designed and manufactured by 4D Systems. The module features a 7.0" colour TFT LCD display, with resistive touch. It is powered by the feature-rich 4D Systems Diablo16 graphics processor, which offers an array of functionality and options for the designer/integrator/user.

Intelligent display modules are embedded solutions used in various applications in manufacturing automotive, medical, home automation, consumer electronics, and other various industries. In fact, there are very few embedded designs on the market today that do not have a display. Even many consumer white goods and kitchen appliances incorporate some form of display. Buttons, rotary selectors, switches and other input devices are being replaced by more colourful and easier-to-use touch screen displays in industrial





2 Software

2.1 Workshop4 IDE

Workshop4 is a comprehensive software IDE for Microsoft Windows that provides an integrated software development platform for all of the 4D family of processors and modules. The IDE combines the Editor, Compiler, Linker and Downloader to develop complete 4DGL application code. All user application code is developed within the Workshop4 IDE.

Workshop4 includes four development environments, for the user to choose based on application requirements or even user skill level- Designer, ViSi-Genie, and ViSi.

Workshop4 Environments

Designer

This environment enables the user to write 4DGL code in its natural form to program the display module.

ViSi - Genie

An advanced environment that 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 (similar to ViSi), set the events to drive them and the code is written for you automatically. ViSi-Genie provides the latest rapid development experience from 4D Systems.

ViSi

A visual programming experience that enables drag-and-drop type placement of objects to assist with 4DGL code generation and allows the user to visualize how the display will look while being developed.

Serial

This environment is also provided to transform the display module into a slave serial display module, allowing the user to control the display from any host microcontroller or device with a serial port.

2.2 Install Workshop4

Download links for the WS4 installer and installation guide can be found on the [Workshop4 product page](#).

Connecting The Display Module To The Pc





Connection Options

Option A – Using the 4D Programming Cable

Please refer to the image below for the proper connection of the display using the 4D Programming Cable:

Option B – Using the uUSB-PA5-II

Please refer to the image below for the proper connection of the display using the uUSB-PA5-II:

Option C – Using the gen4-PA

1. Connect one end of the 5-way Female-Female jumper cable to the gen4-PA with the micro-USB cable inserted.
2. Connect the other end of the 5-way Female-Female jumper cable to the display.

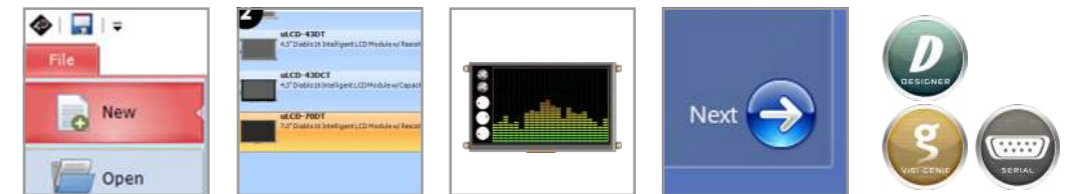
Option D – Using the 4D-UPA

1. Connect one end of the 5-way Female-Female jumper cable to the 4D- UPA with the micro-USB cable inserted.
2. Connect the other end of the 5-way Female-Female jumper cable to the display.

Let WS4 Identify the Display Module

After following the appropriate set of instructions in the previous section, you now need to configure and setup Workshop4 to make sure that it identifies and connects to the correct display module.

1. Open Workshop4 IDE and create a new project.
2. Select the display module you're using from the list.
3. Select your desired orientation for your project.
4. Click next.
5. Choose a WS4 Programming Environment. Only the compatible programming environment for the display module will be enabled.





Getting Started With A Simple Project

After successfully connecting the display module to the PC using your programming module, you can now start creating a basic application. This section shows how to design a simple user interface using the ViSi-Genie environment and utilizing the slider and gauge widgets.

The resulting project consists of a slider (an input widget) controlling a gauge (an output widget). The widgets can also be configured to send event messages to an external host device through the serial port.

Create a New ViSi-Genie Project

You can create a ViSi-Genie project by opening Workshop and by choosing the display type and the environment that you want to work with. This project will be using the ViSi-Genie environment.

1. Open Workshop4 by double-clicking the icon.
2. Create New Project with the New Tab.
3. Choose your display type.
4. Click Next.
5. Choose ViSi-Genie Environment.



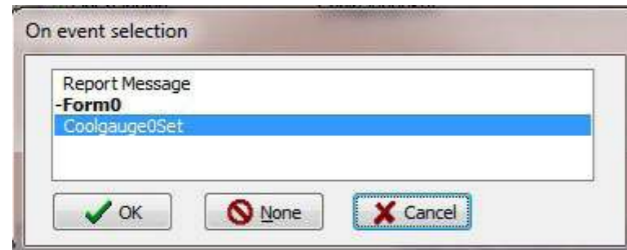
Add a Slider Widget

To add a slider widget, simply click on the Home tab and choose the Inputs Widgets. From the list, you may choose the type of widget that you want to use. In this case, the slider widget is selected.



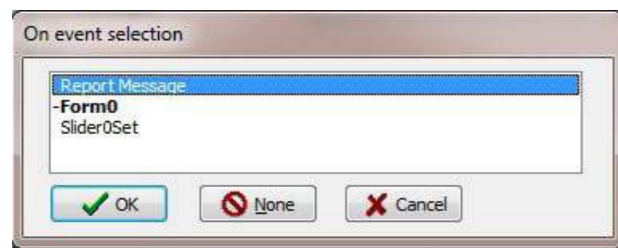


The on-event selection window appears. Select coolgauge0Set, then click OK.



Configure the Input Widget to Send Messages to a Host

An external host, connected to the display module through the serial port, can be made aware of the status of a widget. This can be achieved by configuring the widget to send event messages to the serial port. To do this, set the *OnChanged* event handler of the slider widget to *Report Message*.



Build and Compile the Project

To Build/Upload the project, click the (Build) Copy/Load icon.



Copy the Required Files to the microSD Card

Graphics data for the widgets will actually be stored to a microSD card, which will be accessed by the graphics processor of the display module during runtime. The graphics processor will then render the widgets on the display.

At this point, WS4 generates the required graphics files and will prompt you for the drive to which the microSD card is mounted. Make sure that the microSD card is properly mounted to the PC, then select the correct drive in the Copy Confirmation window, as shown in the image below.





Workshop4 Pro

Standard Workshop4 has its capabilities to make a UI design, but a richer and more interactive design can be achieved with Workshop4 Pro. The Workshop4 PRO License unlocks Genie Magic for the ViSi- Genie environment, along with the Smart Widget Editor for both ViSi and ViSi-Genie Environment. New features and additions continue to be developed, and updates are made available for free once the license has been purchased.

Genie Magic

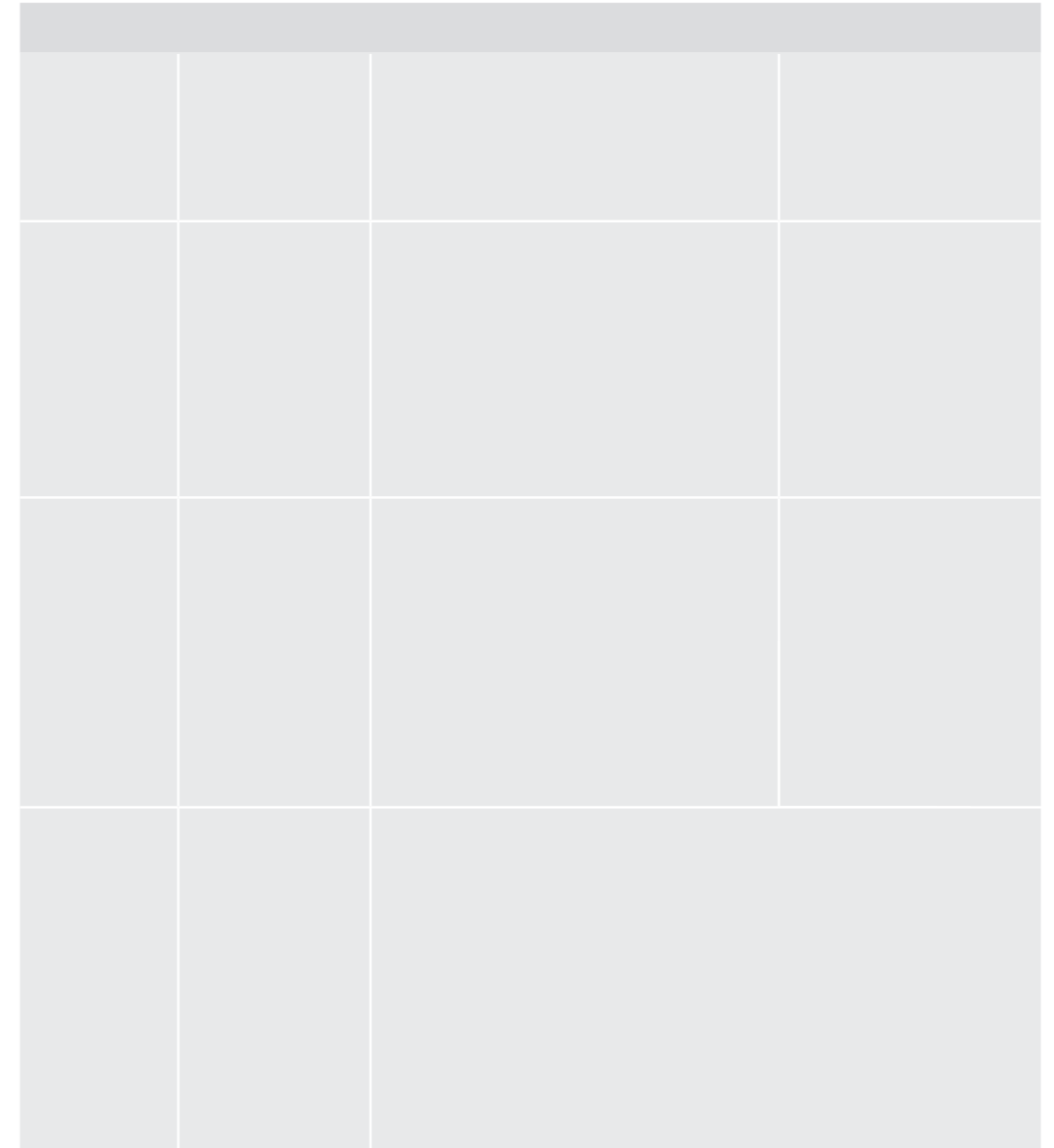
Genie Magic is an upgraded version of the standard ViSi-Genie environment. With this advanced environment, the user can overcome the limitation of the standard ViSi-Genie environment which is its strictly drag-and-drop and event-adding type of programming; while the code can be added using another microcontroller (i.e. Arduino) interfaced to a 4D Systems display. Genie Magic brings the ability to add standard 4DGL code to various points within the ViSi-Genie environment. This enables you to have all the advantages of ViSi-Genie but with the new ability to add extra 4DGL where you want it.

The Magic tab automatically appears once the Workshop4 Pro License has been purchased.

Smart Widgets Editor

Smart Widgets Editor brings a comprehensive tool which allows you to create and animate Gauges, Sliders, Knobs and more, using the tools provided. This allows you to easily add custom graphics and build customised widgets, exactly as you want them.

Smart Widgets are custom widgets designed using the Smart Widgets Editor and Workshop4 Pro comes with 30+ different examples of smart widgets.





Reference Documents



ViSi-Genie is the environment recommended for beginners. This environment doesn't necessarily involve coding, which makes it the most user-friendly platform among the four environments. However, ViSi-Genie has its limitations. For users wanting more control and flexibility during application design and development, the Designer, ViSi, or Serial environments are recommended. ViSi and Designer allow users to write the code for their applications. The programming language used with 4D Systems graphics processors is called "4DGL". The Serial environment, on the other hand, transforms the display module into a slave serial device, allowing the user to control the display using any external host with a serial port. Essential reference documents that can be utilized for further study of the different environments are listed below.

ViSi-Genie Reference Manual

ViSi-Genie does all the background coding, no 4DGL to learn, it does it all for you. This document covers the ViSi-Genie functions available for the PICASO and the DIABLO16 Processors and the communications protocol used known as the Genie Standard Protocol.

4DGL Programmer Reference Manual

4DGL is a graphics oriented language allowing rapid application development. An extensive library of graphics, text and file system functions and the ease of use of a language that combines the best elements and syntax structure of languages such as C, Basic, Pascal, etc. This document covers the language style, the syntax and flow control.

Internal Functions Manual



GLOSSARY



Hardware

1. **4D Programming Cable** – The 4D Programming Cable is a USB to Serial-TTL UART converter cable. The cable provides a fast and simple way to connect all of the 4D devices that require TTL level serial interface to USB.
2. **4D-UPA** – A universal programmer designed to work with multiple 4D Systems display modules.
3. **Embedded System** – A programmed controlling and operating system with a dedicated function within a larger mechanical or electrical system, often with real-time computing constraints. It is embedded as part of a complete device often including hardware and mechanical parts.
4. **Female Header** – A connector attached to a wire, cable, or piece of hardware, having one or more recessed holes with electrical terminals inside.
5. **Micro USB cable** – A type of cable used to connect the display to a computer.
6. **microSD Card** – A type of removable flash memory card used for storing information.
7. **Processor** – A small chip that drives the 4D Systems displays. Its basic job is to receive input and provide the appropriate output.
8. **Programming Adaptor** – Used for programming gen4 display modules, interfacing to a breadboard for prototyping, interfacing to Arduino and Raspberry Pi interfaces.
- 9.



1. **Comm Port** – A serial communication port or channel used to connect devices such as your display.
2. **Device Driver** – A particular form of software application that is designed to enable interaction with hardware devices. Without the required device driver, the corresponding hardware device fails to work.
3. **Firmware** – A specific class of computer software that provides the low-level control for the device's specific hardware.
4. **GTX Tool** – Genie Test Executor debugger. A tool used to check the data sent and received by the display.
5. **GUI** – A form of user interface that allows users to interact with electronic devices through graphical icons and visual indicators such as secondary notation, instead of text-based user interfaces, typed command labels or text navigation.
6. **Image Files** – Are graphics files generated upon program compilation that should be saved into the microSD Card.
7. **Object Inspector** – A section in Workshop4 where the user can change the properties of a certain widget. This is where the widgets customization and Events configuration happen.
8. **PmmC** – The PmmC contains the low level micro-code information (analogy