4D Arduino Adaptor Shield

Datasheet

Contents

1. Description

The 4D Arduino* Adaptor Shield (4Display-Adaptor-Shield) is a simple Arduino Shield designed to provide a convenient interface to attach 4D Systems display modules to the Arduino platform, without having to use jumper wires on the Arduino headers.

The Arduino Adaptor Shield is included in various Arduino Module Packs but can be purchased separately if a 4D Display Module is not required or is owned already.

Communication to the 4D Systems Display Modules is performed via the Arduino serial port and is provided to the user in a simple 5-pin interface, where a 5-way cable (included in the Arduino Module Packs) can be attached between the Shield and the Display Module.

A single digital on the Arduino is used for an external reset for the display and can be disconnected if required. Jumpers are provided to enable selection between different I/O to offer more flexibility over the REV 1 design. Please refer to the Configuration Options section for more information.

Power for the display is supplied from the Arduino's 5V bus, or via an external source such as a 4D Programming Cable or 5V power supply.

The 4D Arduino Adaptor Shield is compatible with the following 4D Systems display modules:

- uOLED-xx-G1/G2 modules
- uLCD-xx-G1/G2 modules
- uLCD-xx-PT modules
- uLCD-43 (All versions)
- uLCD-xx-PTU modules
- uVGA-II/III modules
- uLCD-35DT
- uLCD-70DT
- + Future models

While older modules from 4D Systems may be pin compatible, the 4D Arduino Libraries may not specifically support them.

This Adaptor Shield is also compatible with other platforms which feature Arduino-style headers and use Arduino Shields, such as the Intel Galileo, Arduino Due etc.
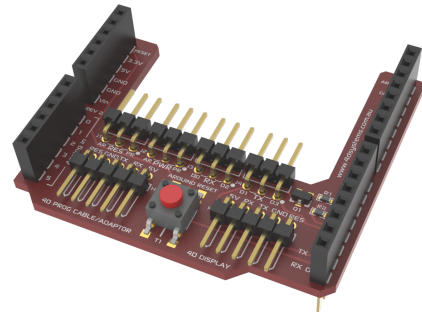
> ✏️ **Note**
>
> This Datasheet is for the Hardware Revision 2.xx of the 4D Arduino Adaptor Shield, not for the Hardware Revision 1 4D Arduino Adaptor Shield. Please refer to the 4D Arduino Adaptor Shield product page on the 4D Systems website for the datasheet on the previous model if required.

The 4D Arduino Adaptor is compatible with most Arduino and clones. Works with 3.3V and 5.0V I/O Arduino modules, provided a 5V supply exists from the Arduino itself, or if no 5V supply exists, an external 5v supply must be connected via the H2 Header, and the J2 jumper moved accordingly.

Arduino Libraries are available to download from the 4D Systems website.
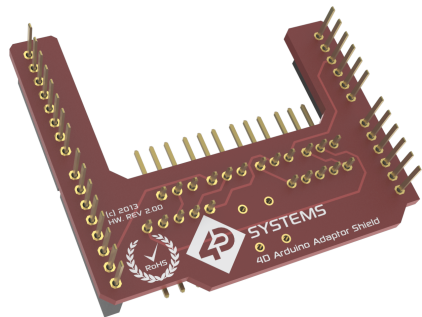
4D Arduino Adaptor Shield Top
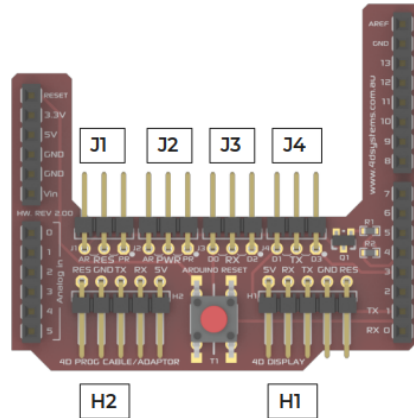

4D Arduino Adaptor Shield Bottom

> ✏️ **Note**
>
> Arduino remains the property of the Arduino Team. All references to the word Arduino and Arduino Hardware are licensed under the Creative Commons Attribution Share-Alike license.

2. Configuration Options

On the 4D Arduino Adaptor, four jumpers (J1, J2, J3 and J4) have been added to provide flexibility to the user Arduino project, to enable I/O to be changed or disabled as required, including the power source for the display.

2.1. Header 1 (H1)

Header 1 (H1) is used to connect the 4D Arduino Adaptor Shield to the 4D Systems display module of choice, using the 5-way cable supplied with the shield. This cable provides 5V, GND, TX, RX and Reset signals to/from the display and Arduino, allowing them to communicate.

2.2. Header 2 (H2)

A 4D Programming Cable/Adaptor connector (**H2**) has been added which can be used to power the display module in the case the display being used requires more power than can be provided by the Arduino (for example if the Arduino is powered off a laptop USB rather than a 5V Jack). If the 4D Programming Cable/Adaptor isn't used and an external power supply is, then the RES pin can be used to reset the display externally, if required. See the J1 section below. The power supply source is determined by the **J2** jumper, please refer to the J2 section below.

> ✏️ **Note**
>
> • The RX and TX Signals from the 4D Programming Cable/Adaptor are **NOT** passed from this header, it is used to provide additional power only.
> • This header **CANNOT** be used to program the display via this Adaptor, while connected to the Arduino.

2.3. Arduino Reset Button (T1)

T1 is a reset button for the Arduino, not for the display. This allows access to the Arduino's reset button which may be covered up when a shield is placed on top of the Arduino for a number of the Arduino models.

2.4. Jumper 1 (J1)

J1 enables the Display Reset to be changed from being triggered from the Arduino on GPIO D4 when in the AR position, or from the H2 connector using an external source, when in the PR position. If the jumper is connected between the AR pin and the central RES pin, then the display can be reset by setting D4 high in the Arduino logic, and then low again. If the jumper is connected between the PR pin and the central RES pin, then the display reset is controlled via the **H2** Connector. This could be switched low by an external circuit, providing an external method of resetting the display.

Removing the jumper from **J1** entirely will disconnect the displays Reset from the Arduino completely, which may be valid for some projects.

> ✏️ **Note**
>
> D4 is always pulled low to GND via a 4.7K Resistor due to the nature of the reset circuit. So even if the **J1** is disconnected, the Arduino D4 input is always going to have a 4.7K pull-down on it. In most cases, this should not affect the operation of D4 if it is to be used for something else in a project when **J1** is disconnected, however, it should be noted that this is the case when using this shield.

2.5. Jumper 2 (J2)

J2 enables the Power Source for the display to be changed from being powered via the Arduino's 5V Supply, or via the **H2** connector. If the Arduino is powered from the User's computer's USB, some computers cannot provide enough current for both the Arduino and the Display to be powered off the same USB port, so auxiliary power can be required. If **J2** is connected between the AR pin and the central PWR pin, then the display is powered off the Arduino's 5V supply. If **J2** is connected between the PR pin and the central PWR pin, then the display is powered off the **H2** connector, via something like the 4D Programming Cable. 5V could also come from an external power source if desired. Removing the jumper from **J2** entirely will disconnect power from the display.

2.6. Jumper 3 (J3)

J3 enables the Serial signals from the display to be received either onto D0 or D2. D0 on most Arduinos is the Hardware Serial Port 0 RX Pin. D2 is often used for Software serials. This allows flexibility for different setups. If the jumper is connected between the D0 pin and the central RX pin, then the Arduino's hardware serial port 0 will be connected to the display. In this configuration, programming of the Arduino is not possible without disconnecting the display, as the USB programming signals for the Arduino will also be received by the display, which will result in corrupt programming signals being received if the Display transmits anything, resulting in a failed programming attempt. Disconnecting this jumper will disconnect the Arduino from the Display, allowing the Arduino to be programmed. If the jumper is connected between the D2 pin and the central RX pin, this will allow Software Serial (or similar) to be used to communicate to the display. The Arduino should be able to be programmed successfully when the jumper is in this position without having to remove the display from **H1** due to the display and Arduino not sharing the same serial port as the Arduino's USB programming.

If desired, a jumper wire can be placed from the central pin of J3 to a completely different pin on the Arduino, such as D19 on an Arduino Mega, allowing Hardware Serial1 to be used. Jumper wires are not included.

2.7. Jumper 4 (J4)

J4 enables the Serial signal to the display to be transmitted from either the Arduino's D1 or D3 pins. D1 on most Arduinos is the Hardware Serial Port 0 TX Pin. D3 is often used for Software serials. Please refer to the description for J3 above, the same applies to **J4**. If the jumper is connected between the D1 pin and the central TX pin, then the Arduino's hardware serial port 0 will be connected to the display. If the jumper is connected between the D3 pin and the central TX pin, will allow Software Serial to be used to communicate to the display. Removing the jumper will disconnect the Arduino from the Display.

If desired, a jumper wire can be placed from the central pin of J4 to a completely different pin on the Arduino, such as D18 on an Arduino Mega, allowing Hardware Serial1 to be used. Jumper wires are not included.

3. Schematic Diagram HW REV 2.01
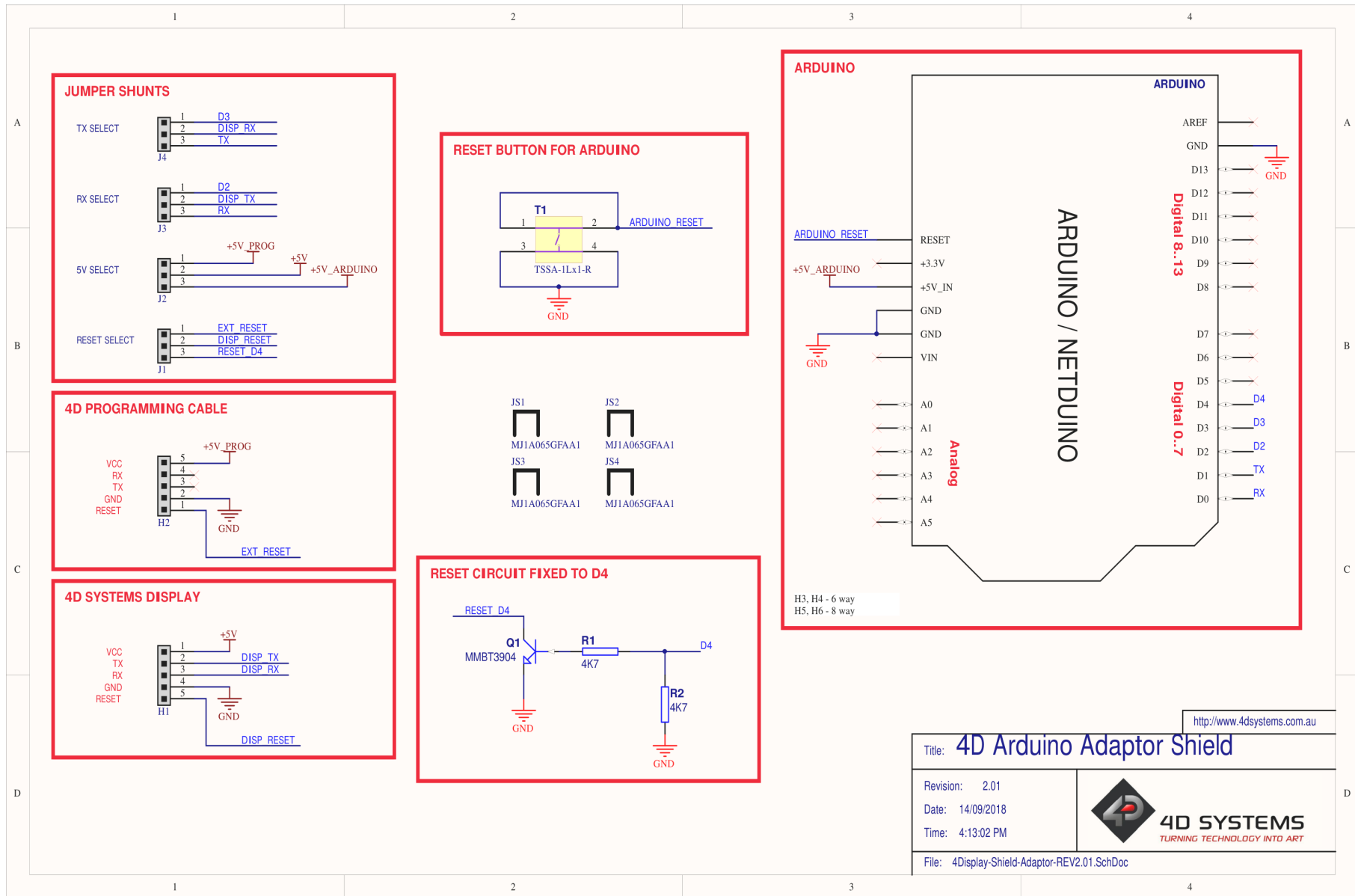
4. Hardware Connection Overview

The following pictures illustrate how to connect the parts of the AR kit. Note, the display module illustrated below is the gen4-uLCD-24DT. The kit you have will determine what display you have. This is just an example.
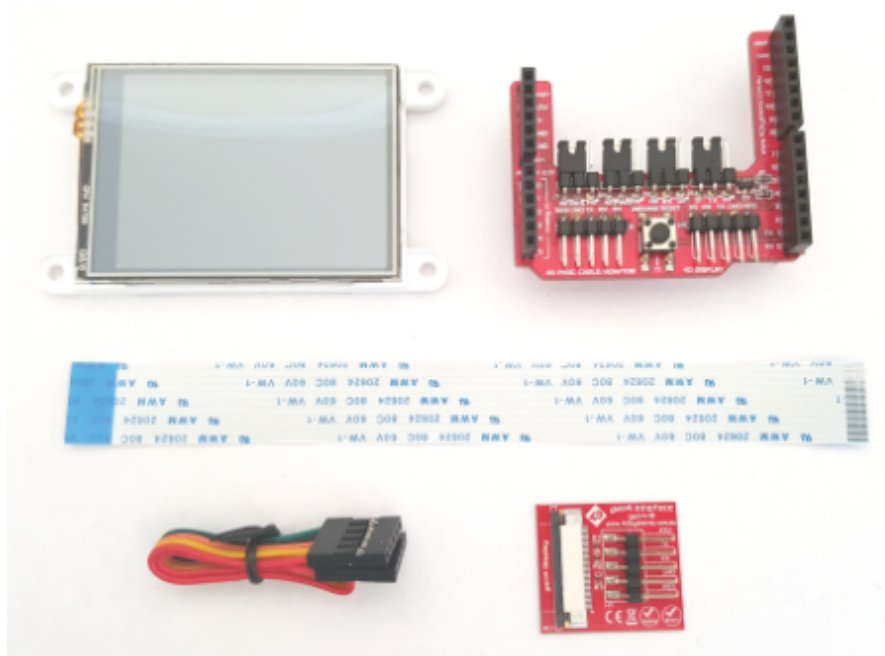
Components in a typical gen4 AR Kit
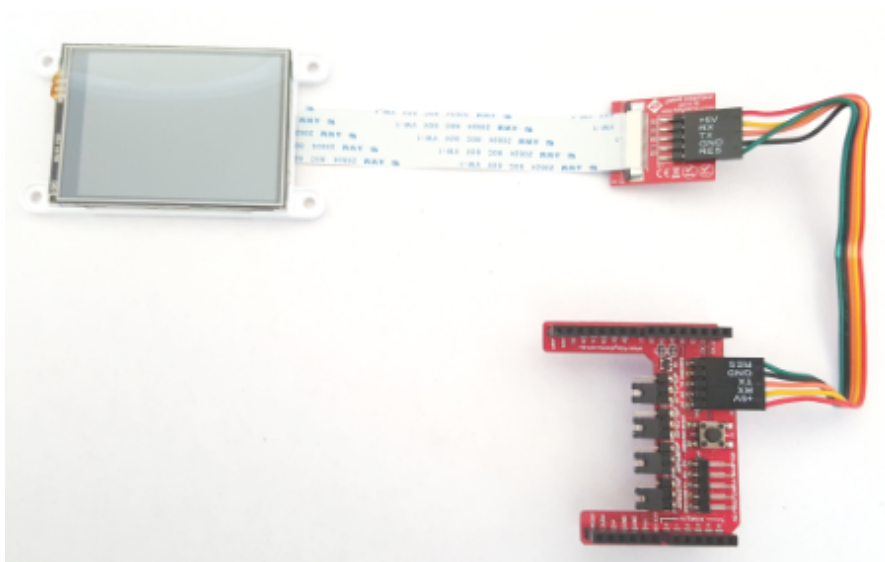
Components in the AR Kit are connected. Arduino Adaptor Shield has shown then connects on top of the customer's Arduino board (not included)

5. Advanced Hardware Options

4D display modules are very capable and powerful pieces of hardware, which can be reconfigured beyond the boundaries of the Arduino. With the use of the 4D Systems Workshop4 IDE Software, the display module can be configured and programmed independently of the Arduino environment.

If you wish to investigate the capabilities of the gen4 display module and reconfigure, please refer display module datasheet, available from the 4D Systems Resource Centre.

You can easily bring the module back to its default factory firmware/configuration by using the Workshop4 IDE Software.

To take advantage of the 4D Systems Workshop4 Software, a 4D Programming Adaptor is required, which is available from the 4D Systems website, or a 4D Systems distributor. These are also available in various 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 protocol to communicate between the Arduino and the Display Module and write 4DGL code directly into the display module to create advanced GUIs 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 an Arduino Library specifically for the ViSi-Genie environment, to enable a rapid development platform. The library is available for download from the gen4 display modules AR product page on the 4D Systems Website.

The **Serial** environment loads an application (SPE) onto the display module which enables commands from the Arduino 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 AR product page on the 4D Systems Website.

6. Programming the Display or Arduino

When programming the Arduino, unless Jumpers J3 and J4 have been moved and alternate GPIO is used for the Serial UART, the display must be disconnected from the adaptor shield in most situations. This is because the Arduino Adaptor Shield connects the display to the Arduino's RX/TX pins, which on most Arduino modules is also used to program the Arduino. There are exceptions, but in the case of the Arduino Uno, you must disconnect the display to program the Arduino.

Likewise, when programming the display, you must disconnect the display from the Arduino adaptor and program it directly with a 4D programming cable or adaptor. When the display has been programmed, connect it back to the adaptor shield to allow them to communicate.

You cannot program the display via this Adaptor. This is not an option. H2 is for external power only, you cannot connect a 4D Programmer to this port to program the display via Workshop4.

7. Specifications

RECOMMENDED OPERATING CONDITIONS

Parameter	Conditions	Min	Typ	Max	Units
Supply Voltage (VCC)	5V supply from either the Arduino itself or from the H2 Header from an External source.	4.5	5.0	5.5	V
Operating Temperature		-10	--	+70	°C

ORDERING INFORMATION

Order Code: 4Display-Adaptor-Shield

Packaging: Module sealed in antistatic foam padded 4D Systems Box

8. Revision History

Hardware Revision

Revision Number	Date	Description
2.01	14/06/2013	Initial Public Release Version

Datasheet Revision

Revision Number	Date	Description
1.3	14/06/2013	Initial Public Release Version
1.4	04/03/2019	Cosmetic changes, removed clutter, added 4th and 5th section
1.5	28/04/2021	Added some more detail around Arduino connection/programming and updated some formatting.
1.6	06/01/2023	Modified datasheet for web-based documentation