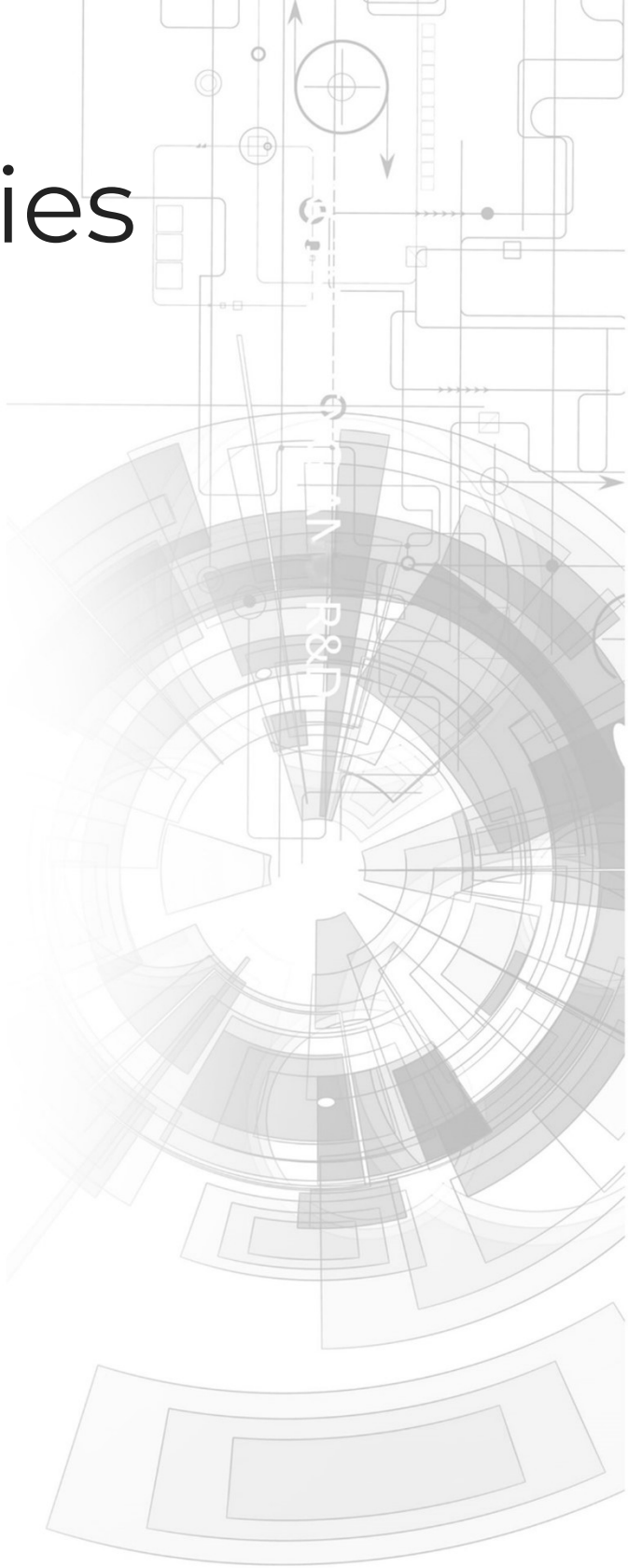
uLCD-43D Series



uLCD-43D (Non-touch)
uLCD-43DT (Resistive touch)
uLCD-43DCT (Capacitive touch)

Datasheet

Revision 1.10

Copyright © 2023 4D Systems

Content may change at any time. Please refer to the resource centre for latest documentation.

Contents

1. Description 4
2. Module Features 5
3. Hardware Overview 6
4. Hardware Interface - Pins 9
   4.1. Serial Ports - TTL Level Serial 9
   4.2. General Purpose I/O 12
   4.3. System Pins 14
   4.4. Alternate Pin Function Overview 15
   4.5. SPI 18
   4.6. I2C 19
   4.7. Pulse Out 20
   4.8. PWM Out 21
   4.9. Pin Counter 22
   4.10. Quadrature In 23
   4.11. Analog Inputs 24
5. PmmC/Firmware Programming 25
6. Module Features 27
   6.1. Diablo16 Processor 27
   6.2. Audio 27
   6.3. SD/SDHC Memory Cards 28
   6.4. FAT16 29
7. Display Precautions 30
8. Hardware Tools 31
   8.1. Programming Cable/Adaptor 31
9. Programming Language 32
10. Workshop4 IDE 33
11. Display Module Part Numbers 34
12. Starter Kit 35

13. Mechanical Details 36
14. Schematic Diagram REV 2.0 37
15. Schematic Diagram REV 2.2 38
16. Specifications 39
17. Revision History 43

1. Description

The uLCD-43D/DT/DCT is an upgrade to the favourite uLCD-43P/PT/PCT range of 4.3" Intelligent Display Modules, now powered by the 4D Labs Diablo16 Graphics Processor.

The uLCD-43D/DT/DCT intelligent display modules are designed for applications demanding a good sized intelligent display module, and where performance is important, along with a good amount of Flash and RAM.

Driving the display and peripherals is the Diablo16 processor, a very capable and powerful chip that enables stand-alone functionality, programmed using the 4D Systems Workshop4 IDE Software. The Workshop IDE enables graphic solutions to be constructed rapidly and with ease due to its design being solely for 4D graphics processors.

The Diablo16 Processor offers considerable FLASH and RAM upgrades over the PICASO processor, and also provides mappable functions such as I2C, SPI, Serial, PWM, Pulse Out, and Quadrature Input, to various GPIO, and also provide up to 4 Analog Input channels.

The display module has an array of features including PWM for Sound, Touch Detection, micro-SD memory storage, general-purpose I/O including Analog Inputs, multiple TTL Serial, I2C and SPI channels and multiple millisecond resolution timers, amongst many more features.

Anything that has been designed in the past to run on a PICASO Processor or uLCD-43P/PT/PCT intelligent display can theoretically run on this Diablo16 Module with minor changes. Please ensure you contact the 4D Systems' support team if unsure if upgrading from a PICASO product and wanting to design with or change over to this uLCD-43D/DT/DCT Module.

A table is included in this datasheet to outline the main differences between the uLCD-43P versions and the uLCD-43D version.

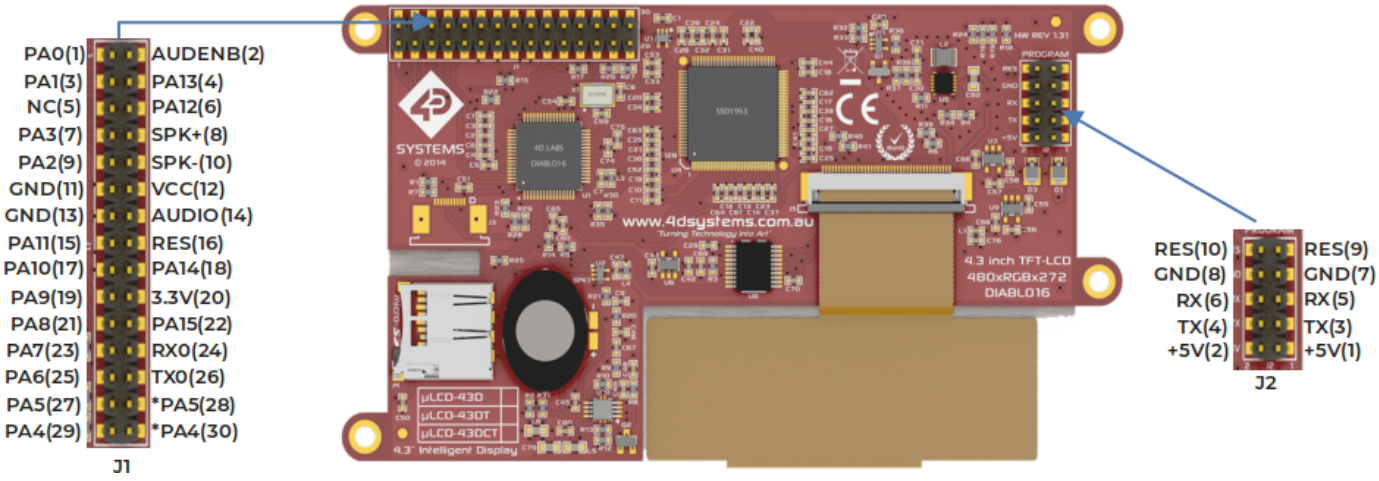
Note

Revision 2.2 hardware, some modules may be populated with SSD1961 Driver ICs and some with SSD1963, depending on supply availability at the time of manufacture. In 99% of cases, the functionality is identical, there are some PAGE-related functions (found in the Internal Function Documentation) which have slight differences only. If you have a specific requirement, please contact our Sales team.

2. Module Features

- Powerful 4.3" Intelligent LCD-TFT display module powered by Diablo16.
- 480x320 HVGA Resolution, RGB 65K true-to-life colours, TFT Screen with integrated 4-wire Non-touch, Resistive or Capacitive Touch Panel.
- Easy 5-pin interface to any TTL Serial Host: VCC, TX, RX, GND, RESET
- 6 banks of 32750 bytes of Flash memory for User Application Code and Data.
- 32Kb of SRAM purely for the User.
- 16 General Purpose I/O pins for user interfacing, which include 4 configurable Analog Inputs.
- The GPIO is variously configurable for alternative functions such as:
    - 3x I2C channels available.
    - 1x SPI dedicated for SD Card and 3x configurable SPI channels available.
    - 1x dedicated and 3x configurable TTL Serial comm ports available.
    - Up to 6 GPIO can be used as Pin Counters.
    - Up to 6 GPIO for PWM (simple and Servo).
    - Up to 10 GPIO for Pulse Output.
    - Up to 14, GPIO can be configured for Quadrature Encoder Inputs (2 channels).
- On-board micro-SD memory card connector for multimedia storage and data logging purposes.
- Display full colour images, animations, icons and video clips on chosen 4D Systems display.
- Supports all available Windows fonts.
- 4.0V to 5.5V range operation (single supply).
- Module dimensions: 119.5 x 67.0 x 15.2mm (including tabs) - See mechanical drawing
- Weighing ~ 80g.
- Display Viewing Area: 95.04 x 53.85mm
- 4x mounting tabs with 3.0mm holes for mechanical mounting.
- RoHS and CE Compliant.

3. Hardware Overview



J2 Pinout (Programming Header)

Pin	Symbol	I/O	Description
1, 2	+5V	P	Main Voltage Supply +ve input pin. Reverse polarity protected. The range is 4.5V to 5.5V, nominal 5.0V.
3, 4	TX	O	Asynchronous Serial Transmit pin, TTL level. Connect this pin to the Receive (Rx) signal of other serial devices. Used in conjunction with the RX pin for programming this micro LCD. This pin is tolerant up to 5.0V levels.
5, 6	RX	I	Asynchronous Serial Receive pin, TTL level. Connect this pin to the Transmit (Tx) signal of other serial devices. Used in conjunction with the TX pin for programming this micro LCD. This pin is tolerant up to 5.0V levels.
7, 8	GND	P	Supply Ground
9, 10	RES	I	Master Reset signal. Internally pulled up to 3.3V via a 10K resistor. An active Low pulse greater than 2 micro-seconds will reset the module. If the module needs to be reset externally, only use open collector-type circuits. This pin is not driven low by any internal conditions. The host should control this pin via one of its port pins using an open collector/drain arrangement.

Note

I = Input, **O** = Output, **P** = Power

J1 Pinout (Programming Header)

Pin	Symbol	I/O	Description
1	PA0	I/O/A	General Purpose I/O pin with Analog Capability. This pin is 3.3V tolerant when used as a Digital, with a range of 0-3.3V when used as an Analog Input
2	AUDENB	O	Audio Amplifier Enable, to enable external amplifier
3	PA1	I/O/A	General Purpose I/O pin with Analog Capability. This pin is 3.3V tolerant when used as a Digital, with a range of 0-3.3V when used as an Analog Input
4	PA13	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant. Used for I2C for Resistive and Capacitive Touch on the uLCD-43DT and uLCD-43DCT models. Please refer to the [General Purpose I/O] section under the heading PA12-PA13 for more information.
5	NC	-	Not connected
6	PA12	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant. Used for I2C for Resistive and Capacitive Touch on the uLCD-43DT and uLCD-43DCT models. Please refer to the [General Purpose I/O] section under the heading PA12-PA13 for more information.
7	PA3	I/O/A	General Purpose I/O pin with Analog Capability. This pin is 3.3V tolerant when used as a Digital, with a range of 0-3.3V when used as an Analog Input, also used for Bus Read
8	SPK+	O	Speaker Output + Can connect to external speaker (disconnect on-board)
9	PA2	I/O/A	General Purpose I/O pin with Analog Capability. This pin is 3.3V tolerant when used as a Digital, with a range of 0-3.3V when used as an Analog Input, also used for Bus Write
10	SPK-	-	Speaker Output – Can connect to an external speaker (disconnect on-board)
11	GND	P	Supply Ground
12	+5V	P	Supply Input +ve, 4.0V to 5.5V, 5.0V Nominal
13	GND	P	Supply Ground
14	AUDIO_OUT	O	Audio Output, to feed into external amplifier
15	PA11	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant
16	RES	I	Master Reset, Active Low (GND) (Refer J2 Pinout)
17	PA10	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant

Pin	Symbol	I/O	Description
18	PA14	I/O	General Purpose Input/Output, 3.3V Level – only 3.3V Tolerant
19	PA9	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant
20	3.3V	I	Regulated 3.3 Volts output, maximum available current 250mA (HW REV 2.0, previously 150mA) to power external circuitry.
21	PA8	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant
22	PA15	-	General Purpose Input/Output, 3.3V Level – only 3.3V Tolerant
23	PA7	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant
24	RX0	I	Asynchronous serial port 0 receive pin. COM0 (same as the RX pin on the J2 Programming Header). 3.3V Level - 5V Tolerant
25	PA6	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant
26	TX0	O	Asynchronous serial port 0 transmit pin. COM0 (same as the TX pin on the J2 Programming Header). 3.3V Level Output.
27	PA5	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant
28	-	-	Connected to PA5, to allow this pin to be used as Serial TX1 like uLCD-43P
29	PA4	I/O	General Purpose Input/Output, 3.3V Level - 5V Tolerant
30	NC	-	Connected to PA4, to allow this pin to be used as Serial RX1 like uLCD-43P

Note

I = Input, **O** = Output, **P** = Power, **A** = Analog Input

4. Hardware Interface - Pins

The uLCD-43D/DT/DCT Module provides both a hardware and software interface. This section describes in detail the hardware interface pins of the device.

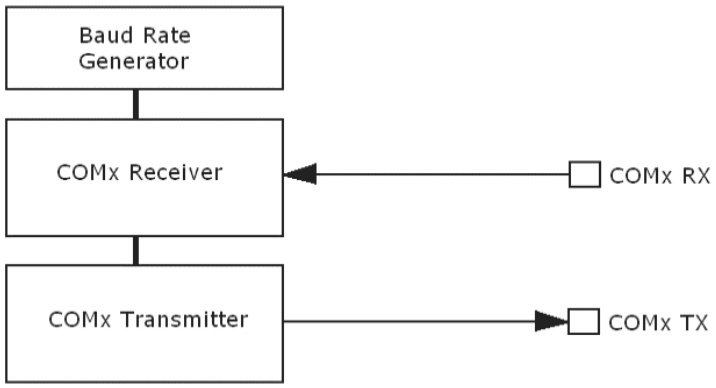
4.1. Serial Ports - TTL Level Serial

The Diablo16 Processor has three hardware asynchronous serial ports (COM1 - COM3) that can be configured on a variety of processor GPIO pins. TX/RX0 (COM0) is dedicated and its pins are fixed. All of the Diablo16's serial ports can be used to communicate with external serial devices.

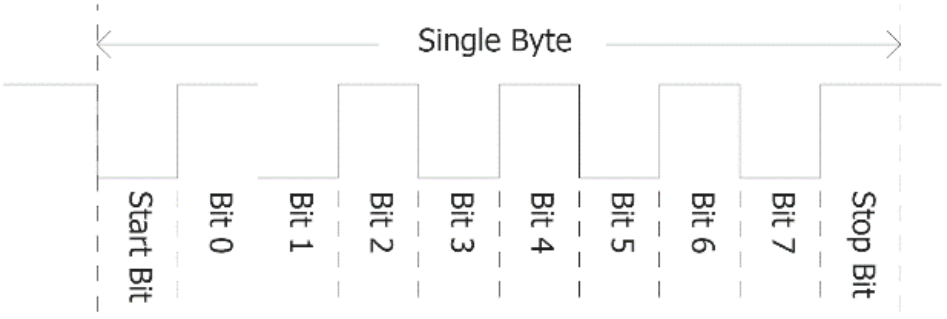
TX/RX0 are referred to as COM0 and is the only one used for programming the Diablo16 itself.

The primary features are:

- Full-Duplex 8-bit data transmission and reception.
- Data format: 8 bits, No Parity, 1 Stop bit.
- Independent Baud rates from 300 baud up to 600K baud.
- Single byte transmits and receives a fully buffered service. The buffered service feature runs in the background capturing and buffering serial data without the user application having to constantly poll any of the serial ports. This frees up the application to service other tasks.



A single-byte serial transmission consists of the start bit, 8 bits of data followed by the stop bit. The start bit is always 0, while a stop bit is always 1. The LSB (Least Significant Bit, Bit 0) is sent out first following the start bit. The figure below shows a single-byte transmission timing diagram.



COM0 is also the primary interface for 4DGL user program downloads and chip configuration PmmC programming. Once the compiled 4DGL application program (EVE byte-code) is downloaded and the user code starts executing, the serial port is then available to the user application. Refer to the PmmC/Firmware Programming section for more details on PmmC/Firmware programming.

TX0 pin (Serial Transmit COM0):

Dedicated Asynchronous Serial port COM0 transmit pin, TX0. Connect this pin to the external serial device receive (Rx) signal. This pin is 5.0V tolerant.

RX0 pin (Serial Receive COM0):

Dedicated Asynchronous Serial port COM0 receive pin, RX0. Connect this pin to an external serial device transmit (Tx) signal. This pin is 5.0V tolerant.

TX1 pin (Serial Transmit COM1):

Asynchronous Serial port COM1 transmit pin, TX1. Connect this pin to the external serial device receive (Rx) signal. This can be configured to one of the GPIO pins, see the table below.

RX1 pin (Serial Receive COM1):

Asynchronous Serial port COM1 receive pin, RX1. Connect this pin to an external serial device transmit (Tx) signal. This can be configured to one of the GPIO pins, see the table below.

TX2 pin (Serial Transmit COM2):

Asynchronous Serial port COM2 transmit pin, TX2. Connect this pin to the external serial device receive (Rx) signal. This can be configured to one of the GPIO pins, see the table below.

RX2 pin (Serial Receive COM2):

Asynchronous Serial port COM2 receive pin, RX2. Connect this pin to an external serial device transmit (Tx) signal. This can be configured to one of the GPIO pins, see the table below.

TX3 pin (Serial Transmit COM3):

Asynchronous Serial port COM3 transmit pin, TX3. Connect this pin to the external serial device receive (Rx) signal. This can be configured to one of the GPIO pins, see the table below.

RX3 pin (Serial Receive COM3):

Asynchronous Serial port COM3 receive pin, RX3. Connect this pin to an external serial device transmit (Tx) signal. This can be configured to one of the GPIO pins, see the table below.

Diablo16 Serial TTL Comm Port Configuration Options

	TX1	RX1	TX2	RX2	TX3
PA0		Yes		Yes	
PA1	Yes	Yes	Yes	Yes	Yes
PA2		Yes		Yes	
PA3	Yes	Yes	Yes	Yes	Yes
PA4	Yes	Yes	Yes	Yes	Yes
PA5	Yes	Yes	Yes	Yes	Yes
PA6	Yes	Yes	Yes	Yes	Yes
PA7	Yes	Yes	Yes	Yes	Yes
PA8	Yes	Yes	Yes	Yes	Yes
PA9	Yes	Yes	Yes	Yes	Yes
PA10		Yes		Yes	
PA11		Yes		Yes	
PA12	Yes	Yes	Yes	Yes	Yes
PA13	Yes	Yes	Yes	Yes	Yes
PA14					
PA15					

As per the table, not all GPIO can be configured to be every serial port, or RX/TX.

Note

Pins **PA4-PA13** are 5.0V tolerant, while pins **PA0-PA3, PA14, PA15** are 3.3V tolerant only. All pins output at 3.3V levels.

Please refer to the Diablo16 Internal Functions Manual for information on how to set the Diablo16 pin mappings.

4.2. General Purpose I/O

There are 16 general-purpose Input/Output (GPIO) pins available to the user. These provide flexibility for individual bit operations along with serving collectively for byte-wise operations using the BUS functions.

Diablo16 Alternate Pin Configurations General Purpose I/O

	Digital Input	Digital Output	Bus Read	Bus Write	Analog Read
PA0	Yes	Yes	Yes	Yes	Yes
PA1	Yes	Yes	Yes	Yes	Yes
PA2	Yes	Yes	Yes	Yes	Yes
PA3	Yes	Yes	Yes	Yes	Yes
PA4	Yes	Yes	Yes	Yes	
PA5	Yes	Yes	Yes	Yes	
PA6	Yes	Yes	Yes	Yes	
PA7	Yes	Yes	Yes	Yes	
PA8	Yes	Yes	Yes	Yes	
PA9	Yes	Yes	Yes	Yes	
PA10	Yes	Yes	Yes	Yes	
PA11	Yes	Yes	Yes	Yes	
PA12	Yes	Yes	Yes	Yes	
PA13	Yes	Yes	Yes	Yes	
PA14	Yes	Yes	Yes		
PA15	Yes	Yes	Yes		

Please refer to the Diablo16 Internal Functions Manual for information on how to set the Diablo16 pin mappings.

PA0-PA3:

General purpose I/O pins, or can serve as Analog Input pins. Each pin can be individually set for INPUT or OUTPUT or ANALOG. Power-Up Reset default is all INPUTS. Digital GPIO can source/sink 10mA. For more information, the Specifications section. These pins have a **0** to **3.3V** range and have a 12-bit resolution.

For more information, refer to the Analog Inputs Section.

PA4-PA11:

General purpose I/O pins. Each pin can be individually set for INPUT or OUTPUT. Power-Up Reset default is all INPUTS. When set as Digital Inputs, the pins are 5V tolerant. Digital GPIO can source/sink 10mA. For more information, see the Specifications section.

PA12-PA13:

PA12 and PA13 are used for the resistive touch controller or capacitive touch controller, on the uLCD-43DT or uLCD-43DCT models. These pins should ideally not be used for other I2C devices on these models, or else it may impact Touch performance. These signals feature 4.7K pull-up resistors on PA12 and PA13. These signals are present on the J1 Header and are not used for touch on the uLCD-43D so are available as GPIO on this model. These pins are 5V tolerant.

PA14-PA15:

General purpose I/O pins. Each pin can be individually set for INPUT or OUTPUT. Power-Up Reset default is all INPUTS. When set as Digital Inputs, the pins are 3.3V tolerant. These pins can be used for high-speed I2C, please refer to the I2C section for more information.

4.3. System Pins

+5V IN (Module Voltage Input): J1 Pin 12, J2 Pin 1

Module supply voltage input pins. At least one of these pins must be connected to a stable supply voltage in the range of 4.0 Volts to 5.5 Volts DC. The nominal operating voltage is 5.0 Volts. Note backlight brightness will be lower for voltages under 5.0V. 500mA of current is the minimum recommended for module stability.

3.3V_OUT (3.3V Output): J1 Pin 20

3.3V Output for the user, 250mA (HW REV 2.0, previously 150mA). Used for powering small external devices or components.

GND (Module Ground): J1 Pin 11/13, J2 Pin 7/8

Device ground pins. Any of these pins can be used to GND the module, at least one must be connected to the ground.

RESET (Module Master Reset): J1 Pin 16, J2 Pin 9/10

Module Master Reset pin. An active low pulse of greater than 2µs will reset the module. Internally pulled up to 3.3V via a 10K resistor. Only use open collector-type circuits to reset the device if an external reset is required.

AUDENB (Audio Enable Output): J1 Pin 2

Output dedicated to enabling or disabling an external amplifier, where required. Active Low.

AUDIO (PWM Audio Input/Output): J1 Pin 14

Universal Input or Output specifically for Audio. This pin connects directly to the onboard amplifier, but also with the filtered PWM output from the Diablo16 Processor. This pin can be used to input a signal into the onboard amplifier to play audio from an external device using the onboard amplifier, or as a line-level output to drive an external amplifier with Diablo16-generated Audio. Note that AUDENB must be enabled (low) in Diablo16 for any input signal to be heard.

4.4. Alternate Pin Function Overview

Most of the GPIO pins have an alternate function other than being for General Purpose I/O.

GPIO pins can be configured to be SPI, I^2C, Serial or a range of other functions.

Note

Not all pins can be configured to be any of the alternate pin functions. Please refer to the tables.

Refer to the tables below that illustrate the GPIO pins you can associate with alternative functions.

Diablo16 Alternate Pin Configurations I/O Support Functions

	Pulse Out	PWM Out	Pin Counter	Quadrature In
PA0	Yes			Yes
PA1	Yes			Yes
PA2	Yes			Yes
PA3	Yes			Yes
PA4	Yes	Yes	Yes	Yes
PA5	Yes	Yes	Yes	Yes
PA6	Yes	Yes	Yes	Yes
PA7	Yes	Yes	Yes	Yes
PA8	Yes	Yes	Yes	Yes
PA9	Yes	Yes	Yes	Yes
PA10				Yes
PA11				Yes
PA12				Yes
PA13				Yes
PA14				
PA15				

Note

- Once you allocate a pin to an alternate function, you can't allocate another pin to the same alternate function.
- Quadrature In requires 2 Pins

The table above illustrates the GPIO pins you can use for the 4 different I/O Support Functions.

Please refer to the Diablo16 Internal Functions Manual for more information on how to set the alternate pin configurations.

The Alternate pin functions have been broken up into a few tables for simplification. There are communication-based functions and I/O support-based functions.

Further information is available in the next sections for each of the alternative pin functions.

The table below illustrates the GPIO pins you can use for the three different SPI channels available.

Diablo16 Alternate Pin Configurations SPI Communications

	SPI1 SDO	SPI1 SDI	SPI1 SCK	SPI2 SDO	SPI2 SDI
PA0		Yes			Yes
PA1	Yes	Yes	Yes	Yes	Yes
PA2		Yes			Yes
PA3	Yes	Yes	Yes	Yes	Yes
PA4	Yes	Yes	Yes	Yes	
PA5	Yes	Yes	Yes	Yes	
PA6	Yes	Yes	Yes	Yes	
PA7	Yes	Yes	Yes	Yes	
PA8	Yes	Yes	Yes	Yes	
PA9	Yes	Yes	Yes	Yes	
PA10		Yes			Yes
PA11		Yes			Yes
PA12	Yes	Yes	Yes	Yes	
PA13	Yes	Yes	Yes	Yes	
PA14					
PA15					

The table below illustrates the GPIO pins you can use for the three different I^2C channels available.

Diablo16 Alternate Pin Configurations I2C Communication

	I^2C1 SDA	I^2C1 SCL	I^2C2 SDA	I^2C2 SCL	I^2C3 SDA
PA0	Yes	Yes	Yes	Yes	Yes
PA1	Yes	Yes	Yes	Yes	Yes
PA2	Yes	Yes	Yes	Yes	Yes
PA3	Yes	Yes	Yes	Yes	Yes
PA4	Yes	Yes	Yes	Yes	Yes
PA5	Yes	Yes	Yes	Yes	Yes
PA6	Yes	Yes	Yes	Yes	Yes
PA7	Yes	Yes	Yes	Yes	Yes
PA8	Yes	Yes	Yes	Yes	Yes
PA9	Yes	Yes	Yes	Yes	Yes
PA10	Yes	Yes	Yes	Yes	Yes
PA11	Yes	Yes	Yes	Yes	Yes
PA12	Yes	Yes	Yes	Yes	Yes
PA13	Yes	Yes	Yes	Yes	Yes
PA14		SPECIAL		SPECIAL	
PA15	SPECIAL		SPECIAL		SPECIAL

Note

SPECIAL - please see I2C section.

4.5. SPI

There are 3 user-configurable SPI channels available for mapping to GPIO, for use by the user for the target application. All 3 SPI channels are Master only, and cannot be configured to be slaves at this time.

The SPI Bus speed is configurable using the **SPIx_Init()** Function in 4DGL and allows various speeds from 78.125Khz to 17.5Mhz.

Please refer to the table for details on which GPIO can be configured for SPI.

> **Note**
>
> The additional SPI channel (SPI0) is dedicated to the memory cards and cannot be reconfigured for alternate uses.

To map an SPI channel to a set of GPIO pins, the following 4DGL functions are used:

- **SPIx_SCK_pin(pin);** // Map the SCK pin
- **SPIx_SDI_pin(pin);** // Map the SDI pin
- **SPIx_SDO_pin(pin);** // Map the SDO pin

where:

'SPIx' is substituted with SPI1, SPI2 or SPI3 accordingly, and

'pin' is the target GPIO pin compatible with that particular pin function.

Chip Select for use with SPI can be any other unused GPIO pin, configured as a Digital Output. The lowering and raising of the selected CS (GPIO) pin is done manually by the user in the 4DGL application.

Please refer to the Diablo16 Internal Functions Manual for more information on how to use the SPI functions, along with the Diablo16 Processor Datasheet.

4.6. I2C

There are 3 user-configurable I2C channels available for mapping to GPIO, for use by the user for the target application. All 3 I^2C channels are Master only, and cannot be configured to be slaves at this time.

Please refer to the table for details on which GPIO can be configured for I^2C.

To map an I^2C Channel to a set of GPIO pins, the following 4DGL function is used:

- **I2Cx_Open(Speed, SCLpin, SDApin);**

where:

'I2Cx' is substituted with I2C1, I2C2 or I2C3 accordingly,

'Speed' is the desired I2C Bus speed, and

'SCLpin' and 'SDApin' are the target GPIO pins compatible with that particular pin function.

Note

The normal I^2C pins are PA0 to PA13, however, the use of these pins has a few limitations.

1. There is no slew rate control at I2C_MED.
2. I2C_FAST is not truly 1MHz.

If either of these restrictions needs to be addressed, a special case of SCLpin = PA14 and SDApin = PA15 exists ONLY for speeds I2C_MED (which uses slew rate control) and I2C_FAST (which is truly 1MHz)

Please refer to the Diablo16 Internal Functions Manual for more information on how to use the I^2C functions, along with the Diablo16 Processor Datasheet.

4.7. Pulse Out

Pulse Out is used to create a single pulse of set duration on the selected pin of choice, which is inverted in polarity to the current state of the pin.

This '*inversion of polarity*' means if a Pin is currently held HI, and Pulse Out is executed on that Pin, the pin will pulse LO and then return to HI. Same with vice versa, if currently LO and Pulse Out are executed on that Pin, it will pulse HI and then return to LO.
This is available in both blocking and non-blocking versions.

Please refer to the table for details on which GPIO can be configured to Pulse Out.

> **Note**
>
> Each Pulse Out request needs at least a 1ms lead time due to the scheduling of the event with the internal 1ms timer.

To enable the Pulse Out function on a GPIO pin, the following 4DGL functions are used:

- **pin_Pulseout(pin, value);** //Non-Blocking
- **pin_PulseoutB(pin, value);** //Blocking

where:

'pin' is the target GPIO pin compatible with that particular pin function, and

'value' is the length of the pulse in milliseconds.

Please refer to the Diablo16 Internal Functions Manual for more information on how to use the Pulse Out functions, along with the Diablo16 Processor Datasheet.

4.8. PWM Out

There are 6 PWM channels available to be configured by the user, with 4-time bases available for selection. The PWM can be configured to be used in Servo Mode, or Simple Mode.

Please refer to the table for details on which GPIO can be configured to PWM.

Servo Mode allows a millisecond input value with 0.01ms resolution, which runs at a frequency of approximately 50Hz or 50pps (20ms). The position of the servo is determined by the width of the pulse. Generally, 1.5ms is 90 degrees, 1ms is 0 degrees and 2ms is 180 degrees. Servos however vary, and the Diablo16 PWM control can be adjusted to suit most applications.

Simple Mode allows a percentage input value with a resolution of 0.1%, which runs at a frequency of approximately 70KHz.

To enable the PWM output on a GPIO pin, the following 4DGL function is used:

- **PWM_Init(pin, mode, value);**

where:

'pin' is the GPIO compatible with the particular pin function,

'mode' is the type of PWM to generate, and

'value' is the parameter that defined the PWM pulse itself.

Please refer to the Diablo16 Internal Functions Manual for more information on how to use the PWM functions, along with the Diablo16 Processor Datasheet.

4.9. Pin Counter

There are 6 Pin Counter channels available to be configured by the user, used to count incoming pulses with the ability to call a user function on overflow. The Pin Counter function is available for use in a variety of modes.

The counters can be read and written at any time.

Please refer to the table for details on which GPIO can be configured to Pin Counter.

To enable the Pin Counter function on a GPIO pin, the following 4DGL function is used:

- **pin_Counter(pin, mode, OVFfunction);**

where:

'pin' is the GPIO pin compatible with this particular function,

'mode' is the type of trigger used to count on such as Rising/Falling/Edge, and

'OVFfunction' is the user function to call when the counter overflows if desired.

Please refer to the Diablo16 Internal Functions Manual for more information on how to use the Pin Counter functions, along with the Diablo16 Processor Datasheet.

4.10. Quadrature In

There are two Quadrature Input channels available on the Diablo16 processor, which requires 2 GPIO pins each.

Please refer to the table for details on which GPIO can be configured for Quadrature Input.

Quadrature Input allows a quadrature encoder to be connected, and the position counter and delta counter can be read at any time.

To enable the Quadrature Input function on a set of GPIO pins (2 pins required), the following 4DGL function is used:

- **Qencoderx(PHApin, PHBpin, mode);**

where:

'Qencoderx' is substituted for Quencoder1 or Quencoder2 accordingly,

'PHApin' is the pin connected to the A Phase of the Encoder,

'PHBpin' is the pin connected to the B Phase of the Encoder, and

'mode' is not currently used so is to be set to zero (0).

Please refer to the Diablo16 Internal Functions Manual for more information on how to use the Quadrature Input functions, along with the Diablo16 Processor Datasheet.

4.11. Analog Inputs

Please refer to the table under the General Purpose I/O section for details about pins you can configure to analog inputs.

The analog inputs on the Diablo16 have a range of 0V to 3.3V, each with a max resolution of 12 bits. You can read the analog inputs using either the standard mode, averaged mode or high-speed mode.

- **Standard Mode** results are immediately read in a sample. Standard Mode can read over 40000 values per second. Operates at 12-bit.
- **Averaged Mode** results are immediately read in 16 samples and their average is returned. Averaged Mode can read ~20000 values per second. Operates at 12-bit.
- **Highspeed Mode** collects a user-specified number of samples at a user-specified rate/frequency and can execute a user function when complete. The updated value updates ~250000 times across 1-4 channels. Operates at 10-bit.

To enable a GPIO as an Analog Input for Standard or Averaged modes, you must use the following 4DGL function to set the pin :

- **pin_Set(mode, pin);**

where:

'mode' is the desired mode defined above, either Standard or Averaged, and

'pin' is the GPIO compatible with this function which is to become an Analog Input.

For highspeed mode, you must the following 4DGL function to set the pin and define the parameters:

- **ana_HS(rate, samples, 1buf, 2buf, 3buf, 4buf, func);**

where:

'rate' is the number of samples per second,
'samples' is the number of samples to collect per channel,
'1buf' to '4buf' are the buffer addresses for the 4 channels, and
'func' is the user function to call when the number of samples specified has been collected.

Please refer to the Diablo16 Internal Functions Manual for more information on how to use the Analog Input functions, along with the Diablo16 Processor Datasheet.

Note

Analog Inputs are 0V-3.3V tolerant only. Do not apply voltages outside of this range as you will damage Diablo16.

5. PmmC/Firmware Programming

The Diablo16 processor is a custom graphics processor. All functionality including the high-level commands is built into the chip. This chip level configuration is available as a PmmC (Personality-module-micro-Code) file, which can be likened to traditional Firmware. There is also a Display Driver file, which separates specific display settings from the PmmC, unlike on the PICASO processor where everything is combined.

A PmmC file contains all of the low-level micro-code information (analogy of that of a soft silicon) which defines the characteristics and functionality of the device. The ability to program the device with a PmmC file provides an extremely flexible method of customising as well as upgrading it with future enhancements.

The Display Driver contains the initialisation and parameters associated with the particular display that is to be connected to the Diablo16 processor, along with product-specific settings and parameters which are required over and above what is found in the PmmC.

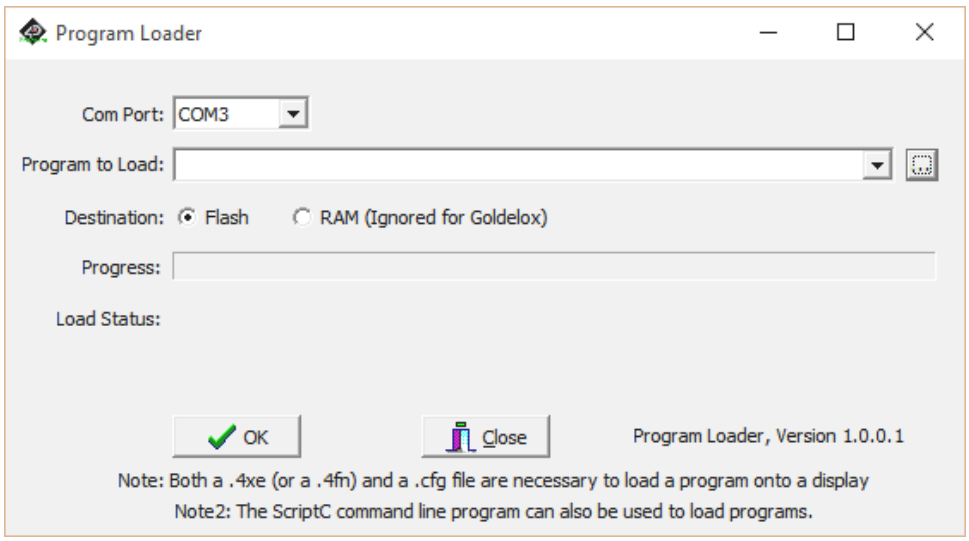
The PmmC file and Display Driver file can only be programmed into the device via the COM0 serial port with the aid of Workshop4, the 4D Systems IDE software.

Solutions, which remove the need to use Workshop4 to program the display modules, are available for commercial customers requiring batch programming or production line programming. These solutions are practical for production staff and minimize the chance of unwanted or unauthorised modifications on the production line.

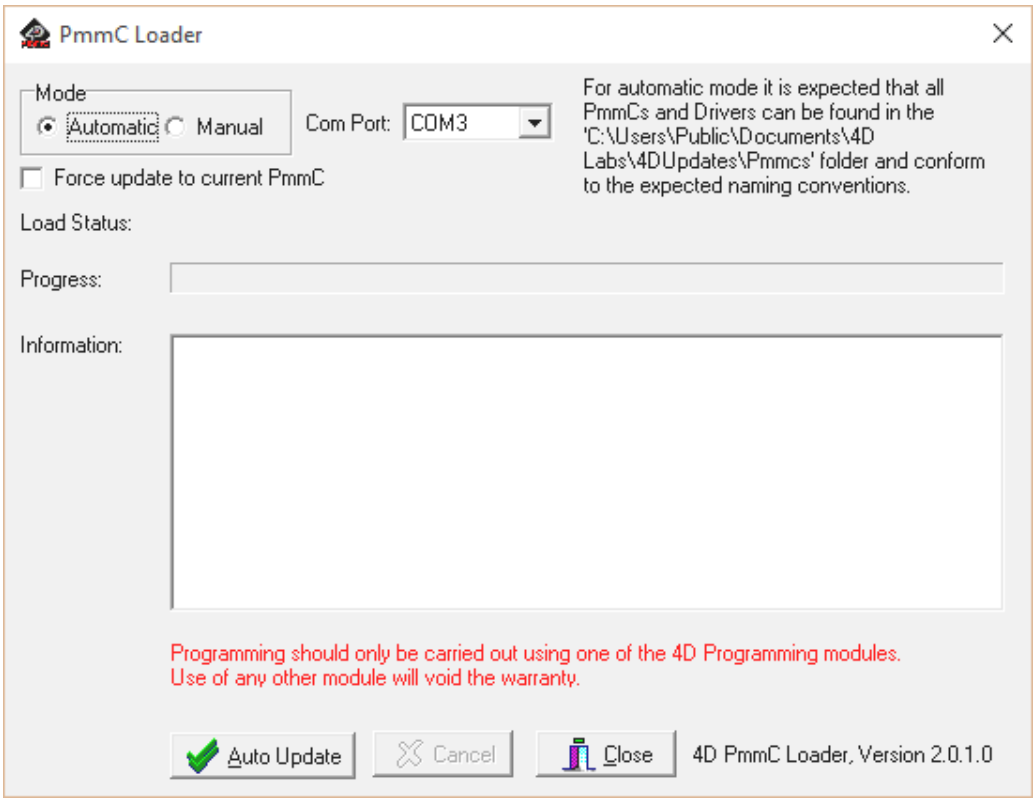
Three solutions come with the Workshop4 IDE, which can be separated for production line computers if required, **ScriptC**, **PmmC Loader** and **Program Loader**.

ScriptC is a command line interface that can be controlled from a Batch Script or similar, useful for repetitive loading or when controlled from an external application. Example scripts are provided, showing how to load PmmC, Display Drivers and Applications using a batch script.

Program Loader is a GUI interface designed to download Applications to either Flash or RAM, useful for testing or production loading, without the need for Workshop4 itself.



PmmC Loader is a GUI interface designed to download PmmC's and Drivers to 4D Systems Processors. It can automatically update existing PmmC's and Drivers present on a module, or manually change or force download to a blank processor or module, overwriting anything written in previously.



Please contact our Support Team for more information on what we can provide. Some solutions also exist for external processor based loading, if there is a requirement for that. This solution is available under NDA.

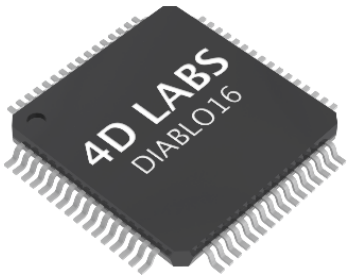
Using a non-4D programming interface could damage your module, and **void your warranty**.

6. Module Features

The uLCD-43D/DT/DCT module is designed to accommodate most applications. Some of the main features of the module are listed below.

6.1. Diablo16 Processor

The module is designed around the Diablo16 Graphics Controller from 4D-Labs.



The Diablo16 is a smart Controller and the interface to the TFT-LCD displays is almost plug-n-play.

All of the data and control signals are provided by the chip to interface directly with the display.

Powerful graphics, text, images, animation and countless more features are built right inside the chip.

You can refer to the Diablo16 Graphics Processor Datasheet for more information.

6.2. Audio

Audio playback support in the Diablo16 Processor enables the uLCD-43D/DT/DCT module to play audio WAV files stored in the micro-SD memory card. PWM and an onboard audio amplifier with 8Ω speaker ensure ample audio output capability. A simple instruction enables the user to play/pause/stop audio files while continuing the execution of the user application code, such as display updates, touch recognition, communications, etc. The audio system also allows real time pitch change of audio samples.

For a complete list of audio commands please refer to the separate document titled:

- Diablo16 4DGL Internal Functions

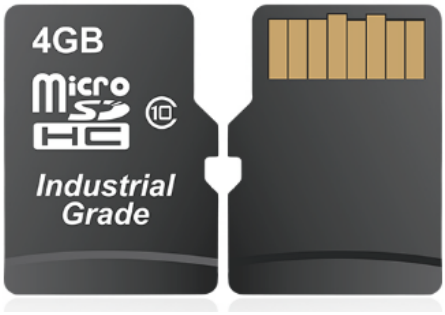
Note

The onboard speaker is a small device designed to project into an audio cavity. It is not very loud by itself. To enable louder audio, it is recommended to use a larger external 8Ω speaker. See the System Pins section for more information.

6.3. SD/SDHC Memory Cards

The Diablo16 processor uses off-the-shelf standard SDHC/SD/microSD memory cards with up to 2GB capacity usable with FAT16 formatting. For any FAT file-related operations, before the memory card can be used it must first be formatted to FAT16. The formatting of the card can be done on any PC system with a card reader. Select the appropriate drive and choose the FAT16 (or just FAT in some systems) option when formatting. The card is now ready to be used in the Diablo16-based application.

The Diablo16 processor also supports high-capacity HC memory cards (4GB and above). The available capacity of SD-HC cards varies according to the way the card is partitioned and the commands used to access it.



The FAT partition is always first (if it exists) and can be up to the maximum size permitted by FAT16. Windows 7 will format FAT16 up to 4GB. Windows XP will format FAT16 up to 2GB and the Windows XP command prompt will format FAT16 up to 4GB.

RMPET, a 4D Systems Tool found in the Workshop4 IDE, is capable of repartitioning and formatting microSD cards to be the appropriate type and format for 4D Systems processors. This tool should be used for all cards.

Note

- An SPI Compatible SDHC/SD/microSD card MUST be used. Diablo16 along with other 4D Systems Processors requires SPI mode to communicate with the SD card. If a non-SPI compatible SD card is used, then the processor will not be able to mount the card.
- Read disturbance is a well-known issue with flash memory devices, such as microSD cards, where reading data from a flash cell can cause the nearby cells in the same memory block to change over time. This issue can be prevented by using industrial-grade microSD cards with read disturb protection. Industrial-grade microSD cards have firmware that actively monitors the read operation and refreshes areas of memory that have high traffic and even move data around to prevent read disturb error from occurring. Furthermore, manufacturers may choose to implement read disturb protection on a specific part of the flash memory only, such that the beginning part of the memory might not be protected. The RMPET utility in Workshop4 is designed to create the first partition at an offset from the start of the microSD card to account for this situation. It is therefore recommended to always partition and format an industrial microSD card using the RMPET utility before using it with 4D Systems processors. Many commercial grade cards designed for Cameras etc, do not handle read disturb well at all, and therefore it is always recommended to use an Industrial grade microSD card with 4D modules. 4D offers one that is tried and tested, on our website.

6.4. FAT16

All 4D Systems display modules featuring 4D Labs processors use off-the-shelf standard SDHC/SD/micro-SD memory cards (SPI Compatible Only) with up to 4GB capacity usable with FAT16 formatting.

For any FAT file-related operations, before the memory card can be used it must first be formatted correctly. Built into Workshop4 is a tool created by 4D, called RMPET (please refer to the Tools menu, in any Environment, inside the Workshop4 IDE). RMPET allows the User to easily partition and format microSD cards, to make their file system ready to be used with 4D Systems modules. The formatting of the card can be done on any PC system with a card reader.

The Diablo16 Processor also supports high-capacity HC memory cards (4GB and above). The available capacity of SD-HC cards varies according to the way the card is partitioned and the commands used to access it. Refer to the 4D Systems RMPET tool in the Workshop4 IDE.

A Max of 4GB can be utilised by the FAT16 file system. The FAT partition is always first (if it exists). Any space larger than 4GB will be RAW, and can still be utilised by your 4D Systems module, using different functions. Please refer to the Application Notes.

7. Display Precautions

- Avoid having to display the same image/object on the screen for lengthy periods. This can cause a burn-in which is a common problem with all types of display technologies. Blank the screen after a while or dim it very low by adjusting the contrast. Better still; implement a screen saver feature.
- Moisture and water can damage the display. Moisture on the surface of a powered display should not cause any problems, however, if water is to enter the display either from the front or from the rear, or come in contact with the PCB, it will damage. Wipe off any moisture gently or let the display dry before usage. If using this display module in an environment where it can get wet, ensure an appropriate enclosure is used.
- Dirt from fingerprint oil and fat can easily stain the surface of the display. Gently wipe off any stains with a soft lint-free cloth.
- The performance of the display will degrade under high temperatures and humidity. Avoid such conditions when storing.
- Do not tamper with the display flex cable that is connected to the control board. This may affect the connection between the display and the driving circuitry and cause failure.
- Displays are susceptible to mechanical shock and any force exerted on the module may result in deformed zebra stripes, a cracked display cell and a broken backlight.
- Always use the mounting holes on the module to mount the display where possible.
- Display modules have a finite life, which is typically dictated by the display itself, more specifically the backlight. The backlight contains LEDs, which fade over time. In the Specifications section is a figure for the typical life of the display, and the criteria are listed.
- The resistive touch model features a touch-sensitive film over the display which is sensitive to pressure. Take note when mounting the display module in an enclosure that pressure is not applied to the surface of the display by the enclosure, false touches will occur, or the touch will simply not function at all.

8. Hardware Tools

The following hardware tools are required for full control of the gen4 Integrated Display Modules.

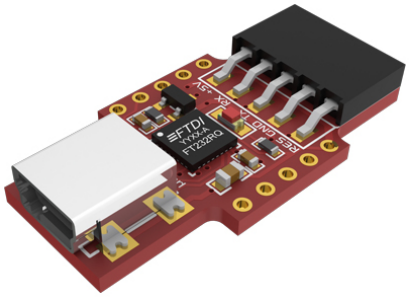
8.1. Programming Cable/Adaptor

The 4D Programming Cable and uUSB-PA5/uUSB-PA5-II Programming Adaptors are essential hardware tools to program, customise and test the Diablo16 Processor.

Either the 4D Programming Cable or the uUSB-PA5 Programming Adaptor can be used.

The 4D programming interfaces are used to program a new Firmware/PmmC, Display Driver and for downloading compiled 4DGL code into the processor. They even serve as an interface for communicating serial data to the PC.

4D Programming Cable



uUSB-PA5/uUSB-PA5-II Programming Adaptor

Note

In addition to these modules, the gen4-PA and 4D-UPA can also be used, however, are not plug-and-play like the Programming Cable or uUSB-PA5.

Using a non-4D programming interface could damage your processor, and **void your Warranty**. These programming interfaces are available from 4D Systems website.

9. Programming Language

The uLCD-43D/DT/DCT Module uses the Diablo16 processor, which belongs to a family of processors powered by a highly optimised softcore virtual engine, EVE (Extensible Virtual Engine).

EVE is a proprietary, high-performance virtual machine with an extensive byte-code instruction set optimised to execute compiled 4DGL programs. 4DGL (4D Graphics Language) was specifically developed from ground up for the EVE engine core. It is a high-level language that is easy to learn and simple to understand yet powerful enough to tackle many embedded graphics applications.

4DGL is a graphics-oriented language allowing rapid application development, and the syntax structure was designed using elements of popular languages such as C, Basic, Pascal and others.

Programmers familiar with these languages will feel right at home with 4DGL. It includes many familiar instructions such as `IF..ELSE..ENDIF, WHILE..WEND, REPEAT..UNTIL, GOSUB..ENDSUB, GOTO, PRINT` as well as some specialised instructions `SERIN, SEROUT, GFX_LINE, GFX_CIRCLE` and many more.

For detailed information about the 4DGL language, please refer to the following documents:

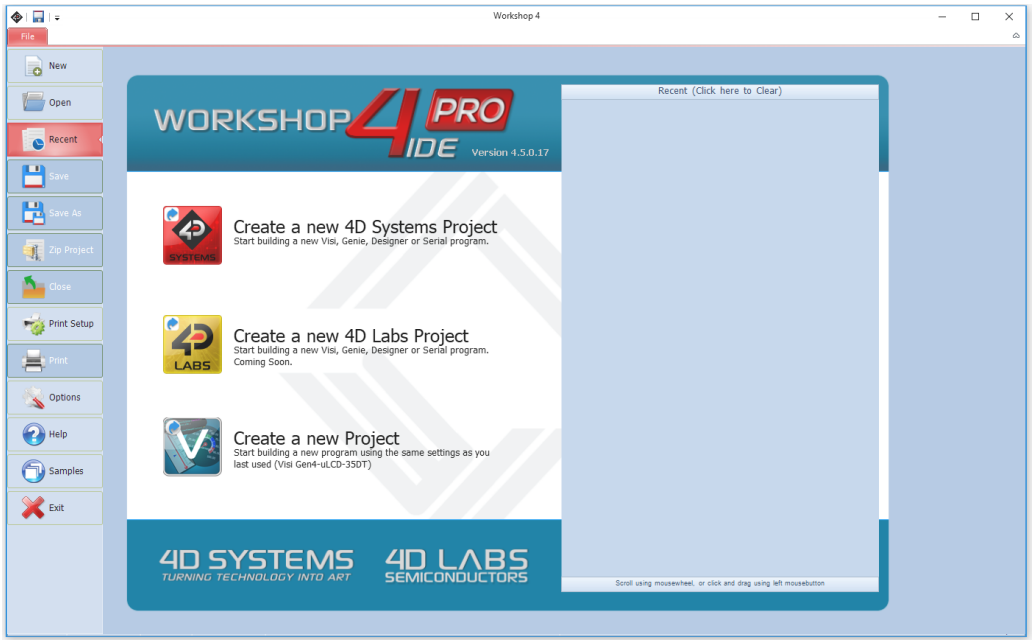
- 4DGL Programmers Reference Manual
- Diablo16 Internal Functions Manual

To assist with the development of 4DGL applications, the Workshop4 IDE combines a full-featured editor, a compiler, a linker and a down-loader into a single PC-based application. It's all you need to code, test and run your applications.

4DGL is available to be written in two of the four environments offered by the Workshop4 IDE, Designer and ViSi.

10. Workshop4 IDE

Workshop4 is a comprehensive software IDE 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 Down-Loader to develop complete 4DGL application code. All user application code is developed within the Workshop4 IDE.



The Workshop4 IDE supports multiple development environments for the user, to cater to different user requirements and skill levels.

- The **Designer** environment enables the user to write 4DGL code in its natural form to program the range of 4D System's intelligent displays.
- A visual programming experience, suitably called **ViSi**, enables drag-and-drop 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. This can be extended with additional features when a Workshop4 PRO license is purchased from the 4D Systems website. Extended Advanced features for Visi-Genie are available in the PRO version of WS4. Further details are explained in the Visi Genie section of the Workshop4 documentation.
- 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.

For more information regarding these environments, refer to the Workshop4 manuals.

The Workshop 4 IDE is available from the 4D Systems website.

11. Display Module Part Numbers

The following is a breakdown of the part numbers and what they mean.

Example:

- uLCD-43DCT

  uLCD - microLCD Display Family

  43 - Display size (4.3")

  D - Diablo16 Processor

  T - Resistive Touch.

  CT - Capacitive Touch

> **Note**
>
> For part numbers that do not include T or CT, are non-touch variants.

12. Starter Kit

4D Systems highly recommends all first-time buyers of 4D Systems' displays, to purchase the Starter Kit when purchasing their first 4D Systems display solution.

The Starter Kit provides all the hardware that is required to get the user up and running.

Not all development environments and features will be needed by every user, however, purchasing the display solution in a Starter Kit, ensures that if you want to take full advantage of the 4D Systems display solution and try out each of the 4D Workshop4 Environments, upgrade PmmC/firmware, you can.

The **Designer** environment can use every feature of the display, however, depending on the user requirements, a micro-SD (uSD) card may not be required. The uSD card is used when displaying images/video/sound, along with data logging to uSD, and a programming cable is required for downloading compiled code and PmmC/Firmware updates.

The **ViSi** environment is the same as Designer in terms of feature utilisation, but is image based so requires a uSD card, along with a programming cable.

The **ViSi-Genie** environment is also image-based, and therefore requires a uSD card and programming cable also.

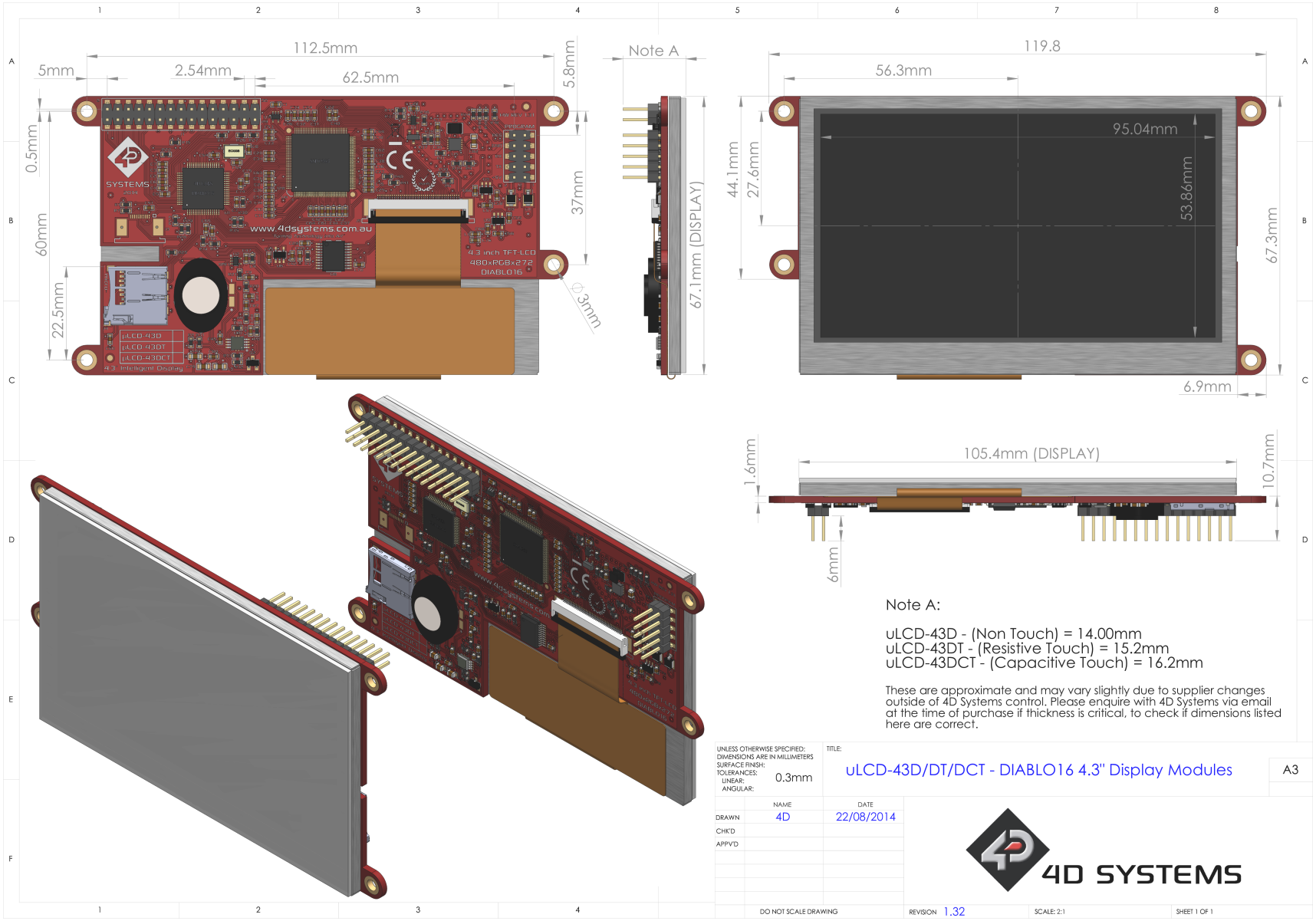
The **Serial** environment does not require either a uSD or Programming cable to be used, however, can use both depending on the user's requirements. The uSD card can be used for such things as storage of multimedia files and data logging, and the Programming cable for PmmC/Firmware updates, or changing to one of the other three programming environments.

The Starter Kit includes:

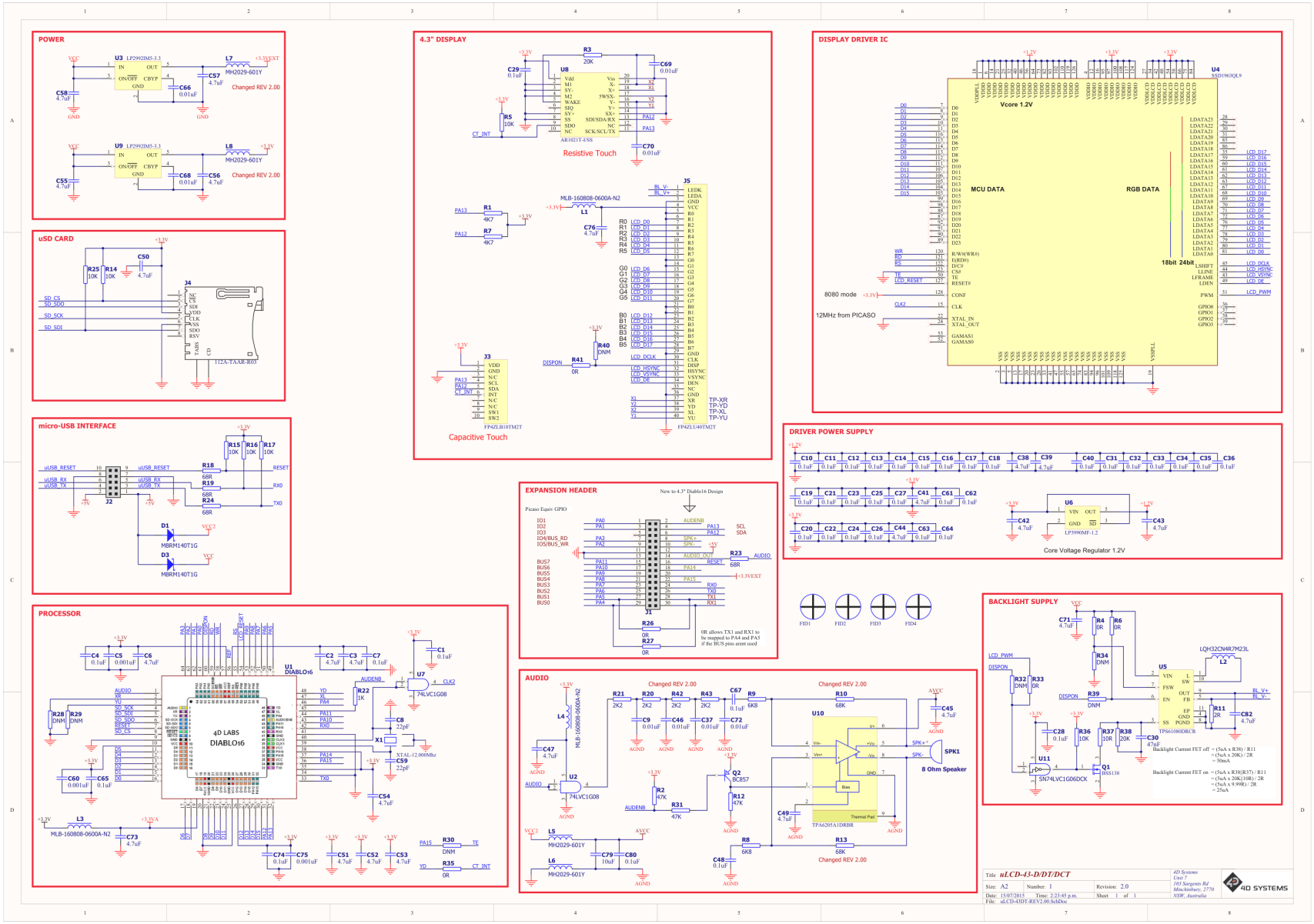
- uLCD-43D/DT/DCT Display Module
- 4GB Industrial microSD Card
- uUSB-PA5-II Programming Adaptor
- 150mm 5-way Female-Female jumper cable, for quick connection to another device or breadboard
- 5-way Male-Male adaptor (for converting the Female-Female cable to be Male-Female)

Simply select the Starter Kit option when purchasing the chosen display module on the 4D Systems shopping cart, or from your local distributor.

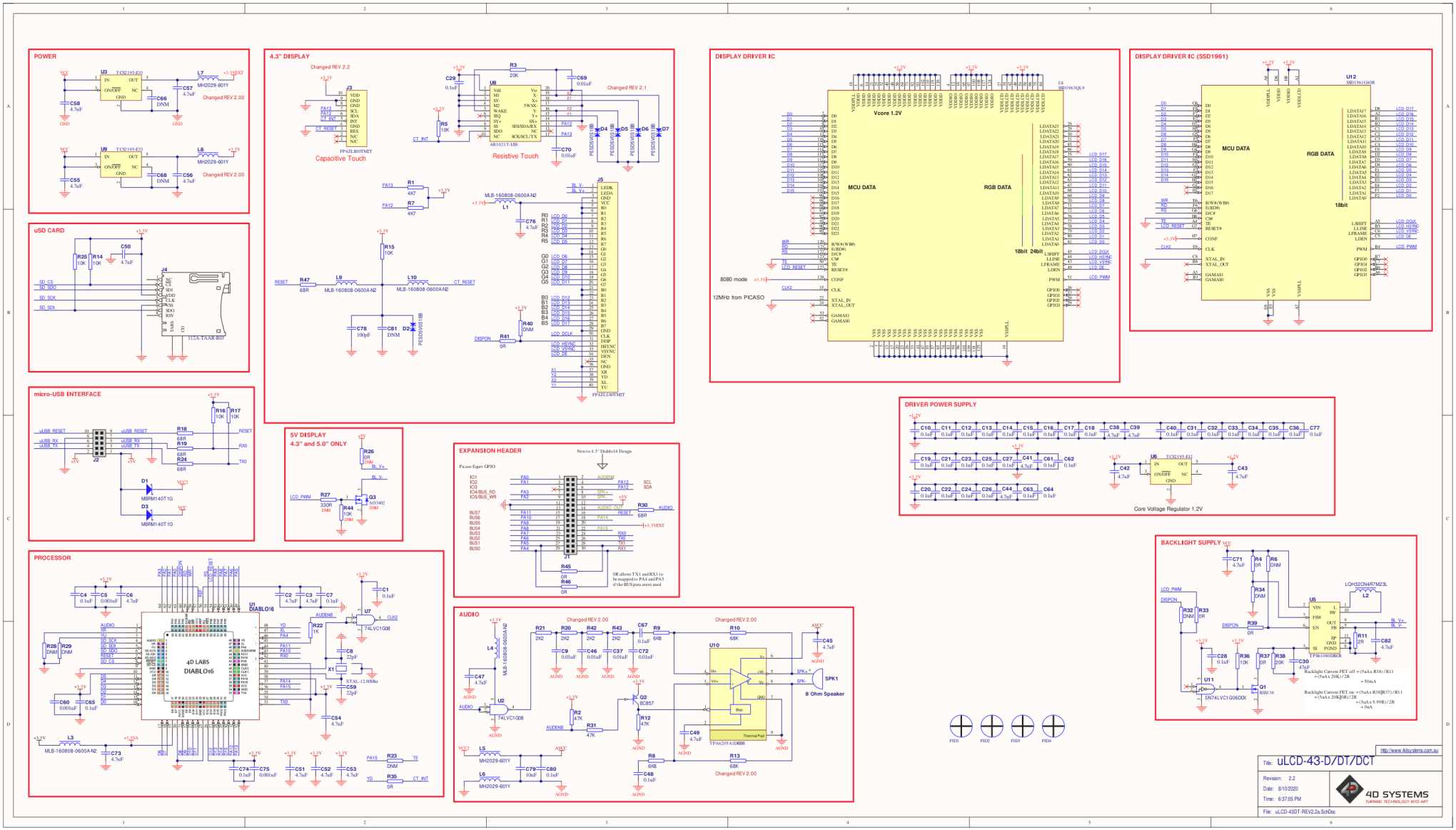
13. Mechanical Details



14. Schematic Diagram REV 2.0



15. Schematic Diagram REV 2.2



16. Specifications

Absolute Maximum Ratings

Operating ambient temperature	-20°C to +70°C
Storage temperature	-30°C to +80°C
Voltage on any digital input pin with respect to GND	-0.3V to 6.0V
Voltage on VCC with respect to GND	-0.3V to 6.0V
Maximum current sunk/sourced by any pin	10.0mA
Maximum current sunk/sourced by all ports	200.0mA

Note

Stresses above those listed here may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the recommended operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Recommended Operating Conditions

Parameter	Conditions	Min	Typ	Max	Un
Supply Voltage (VCC)	Stable external supply required	4.0	5.0	5.5	V
Processor voltage (VP)		--	3.3	--	V
Operating Temperature		-10	--	+60	°(
Input Low Voltage (VIL)	all pins	0	--	0.2VP	V
Input High Voltage (VIH)	non 5V tolerant pins	0.8VP	--	3.3	V
Input High Voltage (VIH)	5V Tolerant Pins, RX pin	0.8VP	--	5.5	V
Input High Voltage (VIH)	PA4-PA13, RX0 and TX0 pins	0.8VP	--	5.5	V
Reset Pulse	External Open Collector	2.0	--	--	µ
Operational Delay	Power-Up or External Reset	500	--	3000	m

Global Characteristics Based on Operating Conditions

Parameter	Conditions	Min	Typ	Max	Un
Supply Current (ICC)	5.0V, heavily depends on screen usage conditions, sleep mode, audio, SD card	--	250	--	m
Output Low Voltage (VOL)	3.3V, IOL = 3.4mA	--	--	0.4	V
Output High Voltage (VOH)	3.3V, IOL = -2.0mA	2.4	--	--	V
Capacitive Loading	All pins	--	--	50	p
Flash Memory Endurance	DIABLO PmmC Programming	--	10000	--	E/
LCD Backlight Operating Hours Endurance	Measured until the display is 50% original brightness, 25 degrees C.	30000	--	--	Ho
Touch Screen Endurance (Resistive Touch)	Number of touches/hits with a 12.5mm tip at a rate of 2x per second with 250gf force	--	1M	--	Touc
	Slide stylus on screen, 100gf force, 60mm/s speed with a 0.8mm polyacetal tip stylus pen	--	100K	--	Slic
Touch Screen Transparency	Resistive Touch	82	--	--	%
	Capacitive Touch	90	--	--	%
Touch Screen Operational Force (Resistive Touch)	Only use Finger or Stylus, do not use anything sharp or metal	20	--	100	C

LCD DISPLAY INFORMATION (TN DISPLAY)

Parameter	Conditions	Specification
Display Type	All displays produced before the IPS release date in the IPS table below, or any modules not specified are therefore TN type displays.	TN - TFT Transmissive LCD
Display Size		4.3" Diagonal
Display Resolution		480 x 272 (Landscape/Wide Viewing)
Display Brightness	uLCD-43D (Contrast = 15)	500 cd/m2
	uLCD-43DT (Contrast = 15)	400 cd/m2
	uLCD-43DCT (Contrast = 15)	475 cd/m2
Display Contrast Ratio	Typical	500:1
Display Viewing Angles	Above Centre	70 Degrees
	Below Centre	60 Degrees
	Left of Centre	70 Degrees
	Right of Centre	70 Degrees
Display Viewing Direction		12 O'clock Display (Optimal viewing is from above when in Landscape/Wide mode)
Display Backlighting		2x5 Parallel LED's
Pixel Pitch		0.198 x 0.198mm (Square pixels)
Pixel Density	Number of pixels in 1 row in 25.4mm	128 DPI/PPI

LCD DISPLAY INFORMATION (IPS DISPLAY)

Parameter	Conditions	Specification
Display Type	uLCD-43D – Modules produced March 2021 onwards. uLCD-43DT – Modules produced March 2021 onwards. uLCD-43DCT – Modules produced May 2020 onwards.	IPS - TFT Transmissive LCD
Display Size		4.3" Diagonal
Display Resolution		480 x 272 (Landscape/Wide Viewing)
Display Brightness	uLCD-43D (Contrast = 15)	600 cd/m2
	uLCD-43DT (Contrast = 15)	510 cd/m2
	uLCD-43DCT (Contrast = 15)	540 cd/m2
Display Contrast Ratio	Typical	800:1
Display Viewing Angles	Above Centre	80 Degrees
	Below Centre	80 Degrees
	Left of Centre	80 Degrees
	Right of Centre	80 Degrees
Display Viewing Direction		ALL (wide viewing IPS Display)
Display Backlighting		2x5 Parallel LED's
Pixel Pitch		0.198 x 0.198mm (Square pixels)
Pixel Density	Number of pixels in 1 row in 25.4mm	128 DPI/PPI

ORDER INFORMATION

Order Code:

uLCD-43D - Non Touch 4.3" Model

uLCD-43DT - Resistive Touch 4.3" Model

uLCD-43DCT - Capacitive Touch 4.3" Model

Packaging:

Module sealed in an antistatic foam padded 4D Systems box

17. Revision History

Hardware Revision

Revision Number	Date	Description
1.32	17/10/2014	Initial Public Release
2.0	22/06/2015	- Audio Amplifier circuitry has been improved to remove background noise - Additional filtering added on 3.3V supply. - Power circuit has been changed: U3 & U9 are different components now and L7 and L8 have been added. - 30Pin Header 3.3V output can deliver 250mA instead of 150mA.
2.2	27/09/2019	- Passive components changed from 0603 to 0402 due to production requirements changed, no functional change. - Ability to mount either SSD1961 or SSD1963 Driver IC's, changed due to supply challenges. Some boards use SSD1961, some use SSD1963, and virtually all features for this module are identical. If you have a specific requirement for one or the other, please contact our Sales Team. - Suppliers of some parts may have changed, but no functional changes. - Addition of CTP reset ability and some ESD hardening. - Movement of the display connector down the board, to support the same displays used by the gen4-uLCD modules as the flex size is different – however, the LCD is otherwise identical. No impact on Customer.

Datasheet Revision

Revision Number	Date	Description
1.5	17/10/2017	Initial Public Release
1.6	04/03/2019	Cosmetic Changes to micro LCD Datasheet range
1.7	11/11/2019	Updated details regarding Serial Port voltage tolerance
1.8	08/10/2020	Minor updates
1.9	01/07/2021	General updates/information. Addition of IPS display information/specs (see Specifications). IPS displays started being used in production as TN stocks were used up. The change-over date varies for each touch type. - Non-Touch (D) models changed over to IPS in March 2021. - Resistive-Touch (DT) models changed over to IPS in January 2021. - Capacitive-Touch (DCT) models changed over to IPS in May 2020.
1.10	06/01/2023	Modified datasheet for web-based documentation.