uLCD-35DT

Datasheet

Revision 1.8

Copyright © 2023 4D Systems

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

Contents

1. Description

The uLCD-35DT is a high-resolution 480x320 pixel 3.5" Intelligent Display Module powered by the 4D Systems' DIABLO16 Graphics Processor.

The uLCD-35DT is designed with versatility in mind, allowing for either cable loom connection* or direct PCB connection to a Host, enabling a wider range of customers to take advantage of this display module.

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 Workshop 4 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 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-35DT Module.

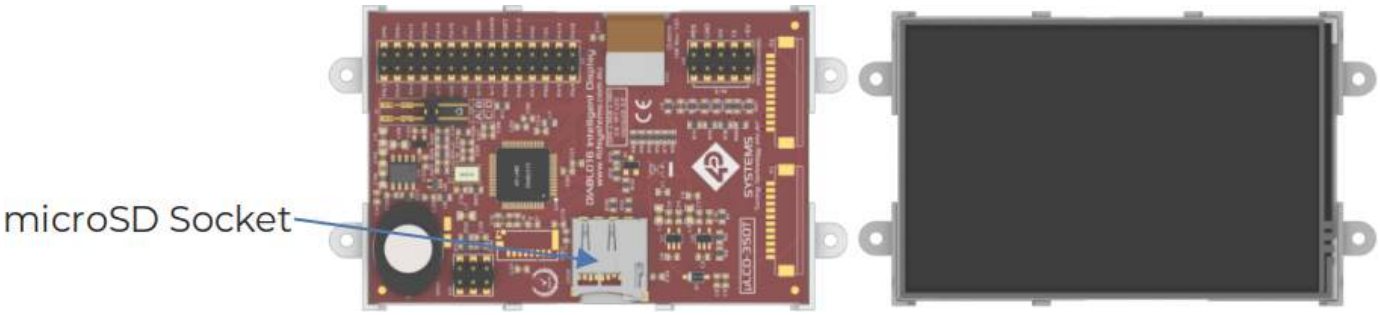
Note

Cable loom connection possible on special orders only, as the cable loom connectors are not populated by default on the uLCD-35DT and only with special orders subject to a MOQ. These are not stocked and are available on special order requests only. Please contact the 4D Systems Sales department if you require customised modules with these headers populated.

2. Features

- Powerful 3.5" Intelligent LCD-TFT display module powered by DIABLO16.
- 480x320 HVGA Resolution, RGB 65K true-to-life colours, TFT Screen with integrated 4-wire Resistive 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. HC memory card support is also available for cards larger than 4GB.
- An alternative micro-SD memory card connector is available on special orders.
- DOS-compatible file access (FAT16 format) as well as low-level access to card memory.
- A dedicated raw PWM Audio pin is driven by WAV files from a micro-SD card.
- 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: 56.9 x 97.6 x 15.8mm (including corner plates).
- Weighing ~ 50g.
- Display Viewing Area: 49.0 x 73.4mm
- 4x corner plates with 2.6mm holes for mechanical mounting.
- RoHS and CE Compliant.

3. Hardware Overview



Note

P1, P2 and uSD1 are not Populated on standard modules. On special order subject to MOQ, these can be produced. Please contact 4D Systems Sales for more information.

H2 Pinout (Programming Header)

Pin	Symbol	I/O	Description
1	+5V	P	Main Voltage Supply +ve input pin. Reverse polarity protected. The range is 4.0V to 5.5V, nominal 5.0V.
3	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. 5.0V Tolerant Pin. Connected to TX0 via Resistor.
5	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. 5.0V Tolerant Pin. Connected to RX0 via Resistor.
7	GND	P	Supply Ground
9	RESET	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.
2,4,6,8,10	N/C	-	Pins are Reserved and must be left disconnected at all times

H1 Pinout (Programming Header)

Pin	Symbol	I/O	Description
1	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
2	SPK-	O	Used in conjunction with SPK+, this is used to connect an external speaker for applications that require a larger more powerful speaker than what is supplied onboard the uLCD-35DT. Disconnect the onboard speaker when using this.
3	PA2	I/O/A	Used in conjunction with SPK-, this is used to connect an external speaker for applications that require a larger more powerful speaker than what is supplied onboard the uLCD-35DT. Disconnect the onboard speaker when using this.
4	SPK+	O	Used in conjunction with SPK-, this is used to connect an external speaker for applications that require a larger more powerful speaker than what is supplied onboard the uLCD-35DT. Disconnect the onboard speaker when using this.
5	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
6	PA11	I/O	General Purpose I/O. This pin is 5.0V tolerant.
7	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.
8	PA10	I/O	General Purpose I/O. This pin is 5.0V tolerant.
9	N/C	-	Not Connected
10	PA14	I/O	General Purpose I/O. This pin is 3.3V tolerant only.
11	GND	P	Supply Ground
12	PA15	I/O	General Purpose I/O. This pin is 3.3V tolerant only.
13	GND	P	Supply Ground
14	+5V	P	Main Voltage Supply +ve input pin. Reverse polarity protected. The range is 4.0V to 5.5V, nominal 5.0V. (Not connected to H2 +5V, separated with diodes)
15	N/C	-	Not Connected
16	AUDIO	I/O	PWM Audio output from Module, or Audio Input to Amplifier, determined by J1
17	NC	-	Not Connected
18	AUDENB	I/O	External Amplifier enables the pin, to control an external amplifier from the module, or control the onboard Amplifier from an external source.
19	PA9	I/O	General Purpose I/O. This pin is 5.0V tolerant.
20	RESET	I	Master Reset, Active Low (GND), See H2 Pin 9 Description
21	PA8	I/O	General Purpose I/O. This pin is 5.0V tolerant.
22	+3.3V_OU	P	3.3V Output, limited to approximately 100mA, for external use.
23	PA7	I/O	General Purpose I/O. This pin is 5.0V tolerant.

Pin	Symbol	I/O	Description
24	RX0	I	Asynchronous serial port 0 receive pin. COM0. 3.3V Levels, 5V tolerant
25	PA6	I/O	General Purpose I/O. This pin is 5.0V tolerant.
26	TX0	O	Asynchronous serial port 0 transmit pin. COM0. 3.3V Level Output
27	PA5	I/O	General Purpose I/O. This pin is 5.0V tolerant.
28	PA13	I/O	General Purpose I/O. This pin is 5.0V tolerant.
29	PA4	I/O	General Purpose I/O. This pin is 5.0V tolerant.
30	PA12	I/O	General Purpose I/O. This pin is 5.0V tolerant.

P1 Pinout (NOT POPULATED)

Pin	Symbol	I/O	Description
1	+5V	P	Main Voltage Supply +ve input pin. Reverse polarity protected. The range is 4.0V to 5.5V, nominal 5.0V.
2	+3.3V_OUT	P	3.3V Output, limited to approximately 100mA, for external use.
3	GND	P	Supply Ground
4	PA12	I/O	General Purpose I/O. This pin is 5.0V tolerant.
5	PA13	I/O	General Purpose I/O. This pin is 5.0V tolerant.
6	PA14	I/O	General Purpose I/O. This pin is 3.3V tolerant only.
7	PA15	I/O	General Purpose I/O. This pin is 3.3V tolerant only.
8	AUDIO	I/O	PWM Audio output from Module, or Audio Input to Amplifier, determined by J1
9	AUDENB	I/O	External Amplifier enables the pin, to control an external amplifier from the module, or control the onboard Amplifier from an external source.
10	RESET	--	Master Reset, Active Low (GND), See P1 Pin 5 Description
11	RX0	--	Asynchronous serial port 0 receive pin. COM0. 3.3V Levels, 5V tolerant
12	--	--	Asynchronous serial port 0 transmit pin. COM0. 3.3V Level Output

P2 Pinout (NOT POPULATED)

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	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
3	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
4	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
5	PA4	I/O	General Purpose I/O. This pin is 5.0V tolerant.
6	PA5	I/O	General Purpose I/O. This pin is 5.0V tolerant.
7	PA6	I/O	General Purpose I/O. This pin is 5.0V tolerant.
8	PA7	I/O	General Purpose I/O. This pin is 5.0V tolerant.
9	PA8	I/O	General Purpose I/O. This pin is 5.0V tolerant.
10	PA9	I/O	General Purpose I/O. This pin is 5.0V tolerant.
11	PA10	I/O	General Purpose I/O. This pin is 5.0V tolerant.
12	PA11	I/O	General Purpose I/O. This pin is 5.0V tolerant.

Note

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

4. Hardware Interface - Pins

This section describes in detail the hardware interface pins of the device.

4.1. Unmounted Connector

Header P1 and P2 (which are not mounted by default) are a 12-way (1x12) 1.25mm pitch wire-to-board housing, part number **125C-SR-12T** from JAWS, enabling a wiring loom to be attached for remote mount applications. Mating connectors for P1 and P2 are **125C-12P** for the housing and **125C-T** for the terminals.

uSD connector uSD1 (which is not mounted by default) is a Molex **47309-2651** micro-SD Push-Pull card connector. This shares the same bus as uSD2 and therefore only one is mounted on the board at once.

Please contact the 4D Systems Sales team if these parts are desirable for your given applications. Please note that a minimum order quantity may be required, and these are not stocked in this configuration.

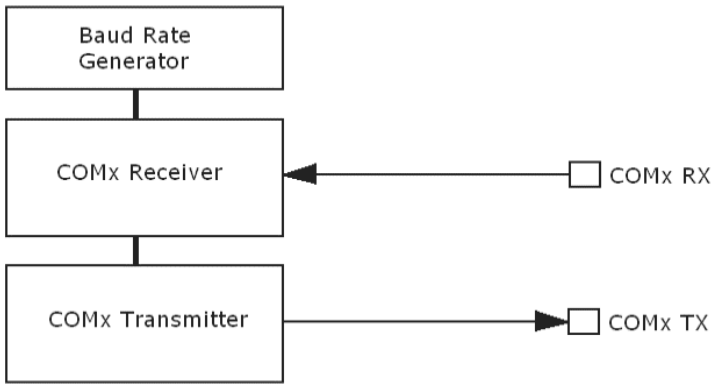
4.2. Serial Ports - TTL Level

The DIABLO16 Processor has three hardware asynchronous serial ports (COM1 - COM3) that can be configured on a variety of the 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.

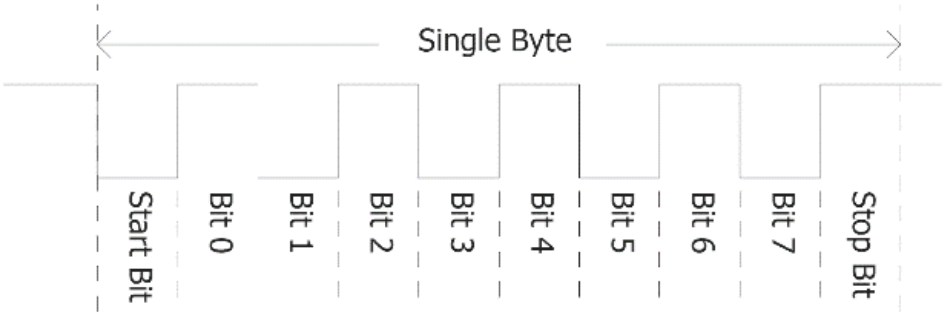
TX/RX0 is referred to as COM0 and is the only one used for programming the DIABLO16 itself.

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.

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 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	RX3
PA0		Yes		Yes		Yes
PA1	Yes	Yes	Yes	Yes	Yes	Yes
PA2		Yes		Yes		Yes
PA3	Yes	Yes	Yes	Yes	Yes	Yes
PA4	Yes	Yes	Yes	Yes	Yes	Yes
PA5	Yes	Yes	Yes	Yes	Yes	Yes
PA6	Yes	Yes	Yes	Yes	Yes	Yes
PA7	Yes	Yes	Yes	Yes	Yes	Yes
PA8	Yes	Yes	Yes	Yes	Yes	Yes
PA9	Yes	Yes	Yes	Yes	Yes	Yes
PA10		Yes		Yes		Yes
PA11		Yes		Yes		Yes
PA12	Yes	Yes	Yes	Yes	Yes	Yes
PA13	Yes	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.

4.3. 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. When set as Digital Inputs, the pins are 3.3V tolerant. Digital GPIO can source/sink 10mA. For more information see the Specifications section.

When set as Analog Inputs, the pins have a 0 to 3.3V range and have 12 bit resolution. For more information, see the Analog Inputs section.

PA4-PA13:

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.

PA14-PA15:

Same as PA4-PA13 however these pins are 3.3V tolerant only.

4.4. System Pins

+5V (Module Voltage Input):

H1 Pin 14, H2 Pin 1, P1 Pin 1:

Module supply voltage input pin. 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 that backlight brightness will be lower for voltages under 5.0V.

3.3V_OUT (3.3V Output):

H1 Pin 22, P1 Pin 2:

3.3V Output for the user limited to approximately 100mA. Used for powering small external devices or components.

GND (Module Ground):

H1 Pins 11 & 13, H2 Pin 7, P1 Pin 3:

Device ground pins. At least one of these pins must be connected to the ground.

RESET (Module Master Reset):

H1 Pin 20, H2 Pin 9, P1 Pin 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):

H1 Pin 18, P1 Pin 9:

Pin used to enable/disable the onboard amplifier, or to control an external amplifier. The pin can be controlled from the Diablo16 or driven directly. Active Low to power the onboard amplifier.

AUDIO (Audio Input/Output):

H1 Pin 16, P1 Pin 8:

This pin outputs Audio from the DIABLO16 Processor to either the onboard amplifier or to an external amplifier or can be used to receive external audio to be played on the onboard amplifier. Use Jumper J1 to control its use. Used in conjunction with the AUDENB signal automatically by the Diablo Processor. Refer to the Schematic diagram to see the 3 viable positions of the J1 Jumper.

4.5. 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, I2C, Serial or a range of other functions.

Note

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

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 a pin is allocated to an alternate function, another pin cannot also be allocated to the same alternate function.
- Quadrature In requires 2 Pins

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 following table illustrates which GPIO can be used for the three different SPI channels available.

DIABLO16 Alternate Pin Configurations SPI Communications

	SPI1 SDO	SPI1 SDI	SPI1 SCK	SPI2 SDO	SPI2 SDI	SPI2 SCK	SPI3 SDO	SPI3 SDI	SPI3 SCK
PA0		Yes			Yes			Yes	
PA1	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PA2		Yes			Yes			Yes	
PA3	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PA4	Yes	Yes	Yes	Yes		Yes	Yes	Yes	Yes
PA5	Yes	Yes	Yes	Yes		Yes	Yes	Yes	Yes
PA6	Yes	Yes	Yes	Yes		Yes	Yes	Yes	Yes
PA7	Yes	Yes	Yes	Yes		Yes	Yes	Yes	Yes
PA8	Yes	Yes	Yes	Yes		Yes	Yes	Yes	Yes
PA9	Yes	Yes	Yes	Yes		Yes	Yes	Yes	Yes
PA10		Yes			Yes			Yes	
PA11		Yes			Yes			Yes	
PA12	Yes	Yes	Yes	Yes		Yes	Yes	Yes	Yes
PA13	Yes	Yes	Yes	Yes		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	I^2C3 SCL
PA0	Yes	Yes	Yes	Yes	Yes	Yes
PA1	Yes	Yes	Yes	Yes	Yes	Yes
PA2	Yes	Yes	Yes	Yes	Yes	Yes
PA3	Yes	Yes	Yes	Yes	Yes	Yes
PA4	Yes	Yes	Yes	Yes	Yes	Yes
PA5	Yes	Yes	Yes	Yes	Yes	Yes
PA6	Yes	Yes	Yes	Yes	Yes	Yes
PA7	Yes	Yes	Yes	Yes	Yes	Yes
PA8	Yes	Yes	Yes	Yes	Yes	Yes
PA9	Yes	Yes	Yes	Yes	Yes	Yes
PA10	Yes	Yes	Yes	Yes	Yes	Yes
PA11	Yes	Yes	Yes	Yes	Yes	Yes
PA12	Yes	Yes	Yes	Yes	Yes	Yes
PA13	Yes	Yes	Yes	Yes	Yes	Yes
PA14		SPECIAL		SPECIAL		SPECIAL
PA15	SPECIAL		SPECIAL		SPECIAL	

Note

SPECIAL - please see I2C section.

4.6. 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 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.

4.7. I2C

There are 3 user-configurable I2C channels available for mapping to GPIO, for use by the user for the target application. All 3 I2C 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.

4.8. 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.

4.9. 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.

4.10. 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 the 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.

4.11. 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.

4.12. 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.

> **Note**
>
> The various analog modes can interfere with the operation of the touch screen if their functions are called too frequently. It is recommended to limit the calls of the analog functions to a maximum of once every millisecond. Please refer to the Diablo16 Internal Functions Manual for further information on this topic.

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 0-3.3V 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.

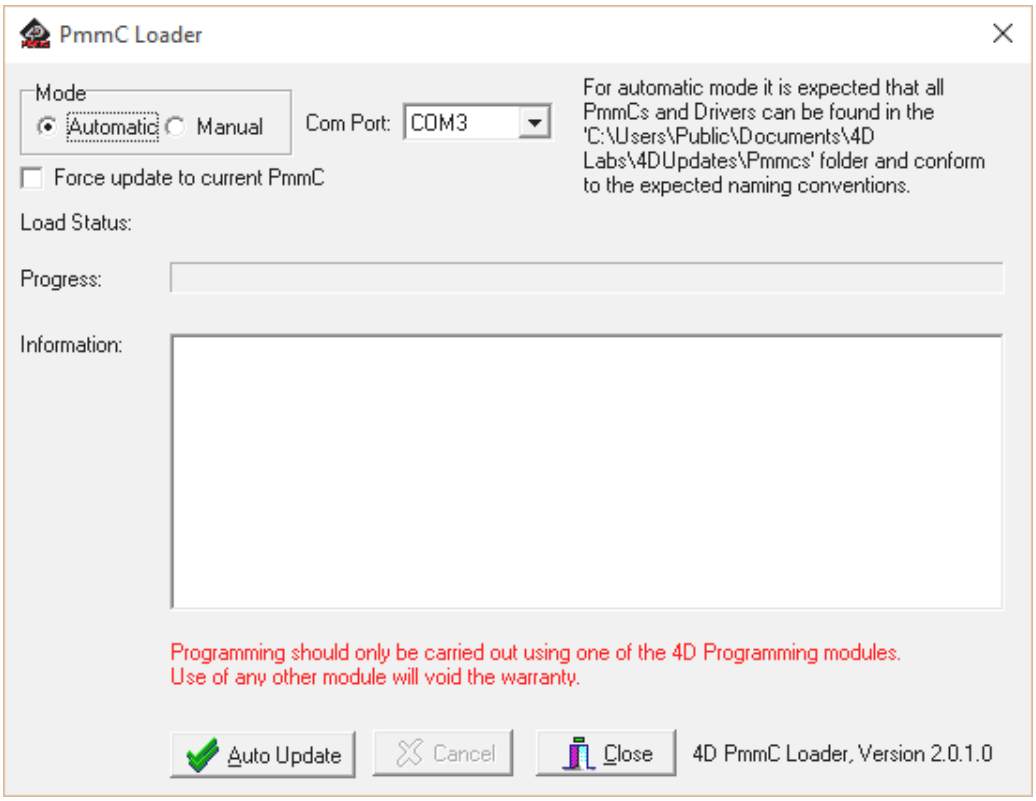
A PmmC file contains all of the low level micro-code information (analogy of that of a soft silicon) which defines the characteristics and functionality of the device. The ability to program the device with a PmmC file provides an extremely flexible method of customising as well as upgrading it with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

The Display Driver contains the initialisation and parameters associated with the particular display that is to be connected to the DIABLO16 processor.

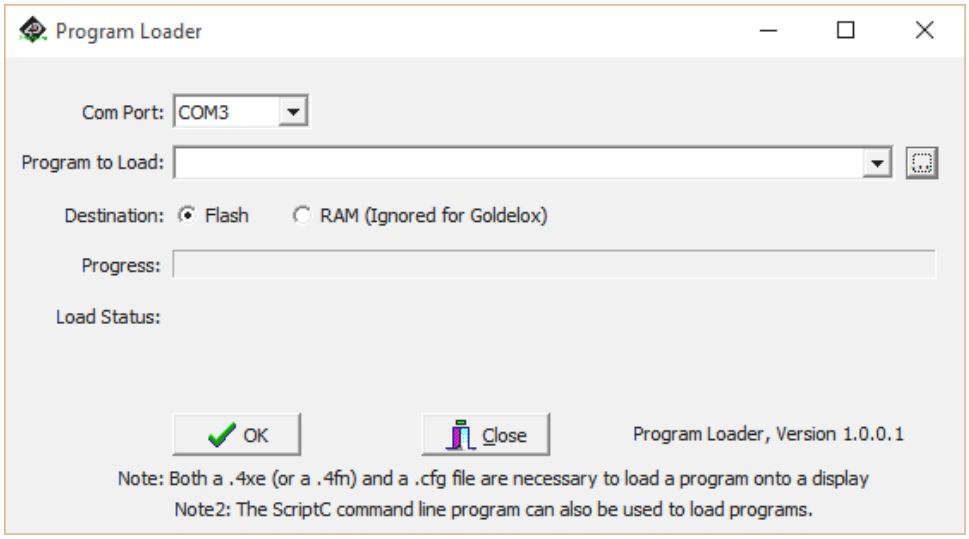
The PmmC file and Display Driver file can only be programmed into the device via the COM0 serial port with the aid of Workshop 4, the 4D Systems IDE software.

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

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.



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.



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-35DT module is designed to accommodate most applications. Some of the main features of the module are listed below.

6.1. Display - 3.5" TFT Touch Screen

The uLCD-35DT module is equipped with a 3.5" TFT display. Details of the display are listed below:

- Screen Size: 3.5" diagonal, 320x480 resolution, 65K colours
- Integrated 4-Wire Resistive Touch Screen
- Screen Dimensions: 54.5 x 83.0 x 3.35mm
- Viewing Area: 49.76 x 77.04mm
- Pixel Pitch: 0.153(H) x 0.153(V)mm
- Brightness: 220cd/m2
- Contrast Ratio: 500:1
- Viewing Angle Above Centre: 60 degrees
- Viewing Angle Below Centre: 60 degrees
- Viewing Angle Left of Centre: 70 degrees
- Viewing Angle Right of Centre: 70 degrees
- Viewing Direction: 6 O'clock
- 7 LEDs for Backlighting

Note

The Displays used are the highest-rated 'Grade A' Displays, which allow for 0-4 defective pixels. A defective pixel could be solid Black (Dead), White, Red, Green or Blue.

6.2. DIABLO16 Processor

The DIABLO16 is a smart Controller and the interface to the TFT-LCD 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.

6.3. Audio

Audio playback support in the DIABLO16 Processor enables this module to play audio WAV files stored in the micro-SD memory card. Filtered PWM audio is generated and made available on the AUDIO pin of the 30-way ZIF connector, ready to interface with an audio amplifier on your application board.

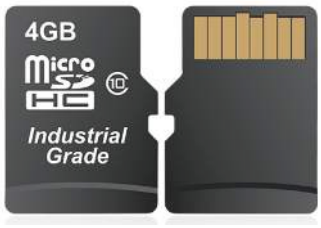
A simple instruction enables the user to play/pause/stop audio files while continuing the execution of the user application code, such as display updates, touch recognition, communications, etc. The audio system also allows real-time pitch change of audio samples.

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

- DIABLO16 4DGL Internal Functions

6.4. SD/SDHC Memory Cards

The module supports micro-SD memory cards via the onboard latch-type micro-SD connector. The memory card is used for all multimedia file retrieval such as images, animations and movie clips. The memory card can also be used as general-purpose storage for data logging applications. Support is available for off-the-shelf micro-SD (< 4GB) and high-capacity HC memory cards (4GB and above). Memory cards up to 32GB in size can be used, however, it must be noted that only a portion of this can be used by the FAT16 file system. See the section below for more details about FAT16.



Note

A microSD card capable of SPI is a requirement for all 4D Systems' display modules powered by Goldelox, Picaso or Diablo16 Processors. If a non-SPI compatible card is used, it will simply fail to mount, or may cause intermittent issues resulting in lockups and crashing of the application. Please refer to the 4D Systems website for microSD cards offered by 4D Systems.

6.5. FAT16

The uLCD-35DT Module uses off-the-shelf standard SDHC/SD/micro-SD 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 with the FAT16 option. 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.

7. Display Precautions

- Avoid having to display the same image/object on the screen for lengthy periods. This will 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 will cause the electrodes to corrode. Wipe off any moisture gently or let the display dry before usage.
- 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's corner plates to mount the display.

8. Hardware Tools

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

8.1. Programming Cable/Adaptor

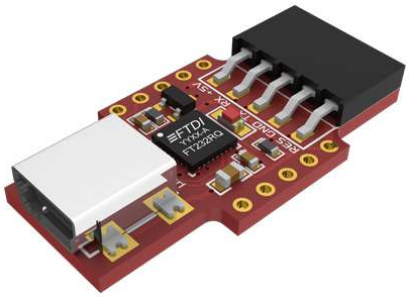
The 4D Programming Cable and uUSB-PA5/uUSB-PA5-II Programming Adaptors are essential hardware tools to program, customise and test the DIABLO16 Processor.

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.

The 4D Programming Cable and uUSB-PA5 Programming Adaptor are available from 4D Systems website.

4D Programming Cable



uUSB-PA5/uUSB-PA5-II Programming Adaptor

> **Note**
>
> In addition to these modules, the gen4-PA and uUSB-PA5/uUSB PA5-II can also be used.

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

9. Programming Language

The uLCD-35DT 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 which 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 downloader 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. The other two environments, Serial and ViSi-Genie do not directly use 4DGL by the User (Except in Workshop4 Pro, for ViSi-Genie), however, it is present in the background. Serial is an application that runs, and that is written in 4DGL. ViSi-Genie automatically generates 4DGL itself based on what is configured in the GUI. More information about each follows.

10. Workshop4 IDE

Workshop 4 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 Downloader to develop complete 4DGL application code. All user application code is developed within the Workshop 4 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. 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.

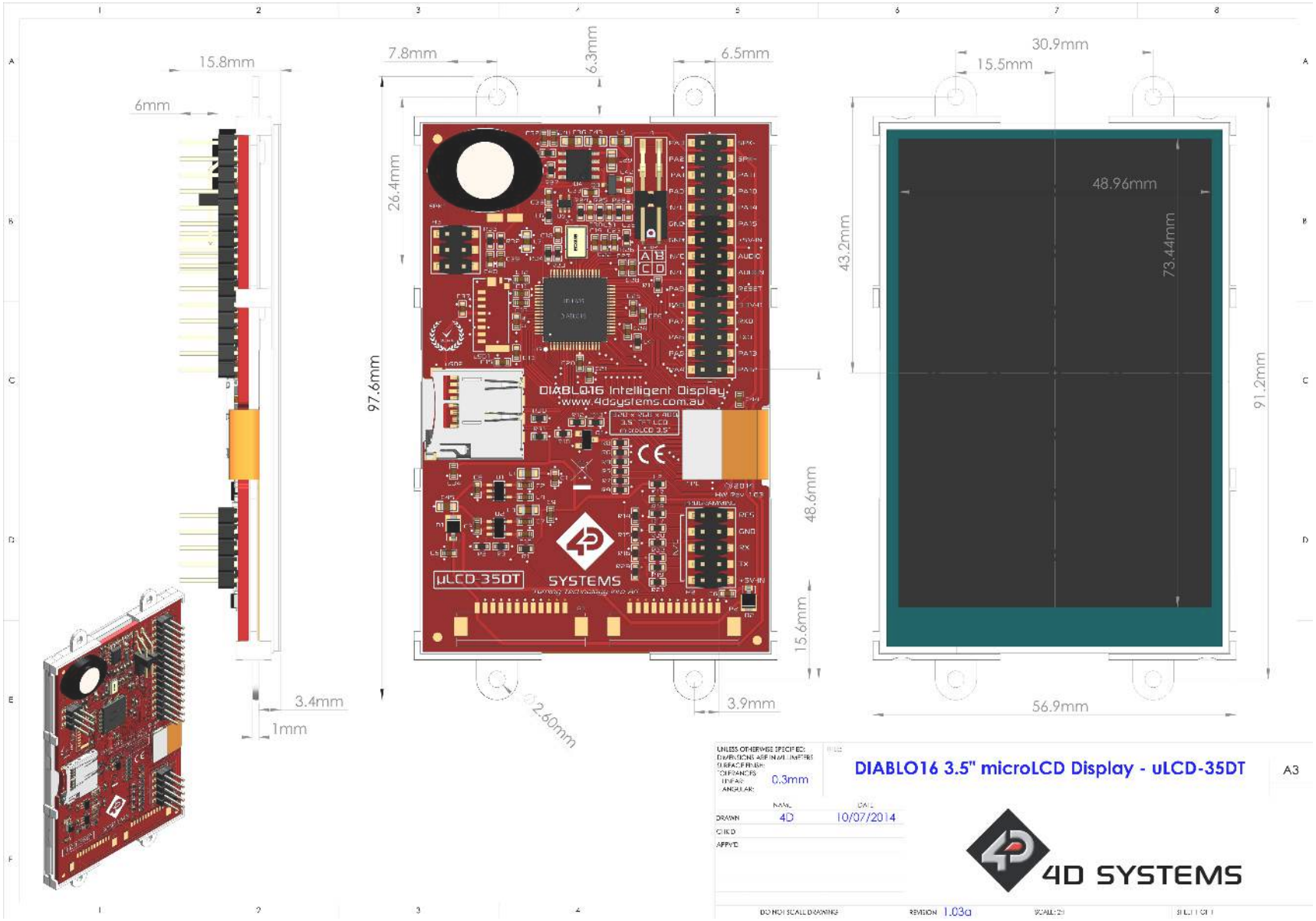
The **Serial** environment does not require either a uSD or Programming cable to be used, however, can use both depending on the user's requirements. The uSD card can be used for such things as storage of multimedia files and data logging, and the Programming cable for PmmC/Firmware updates, or changing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

The Starter Kit includes:

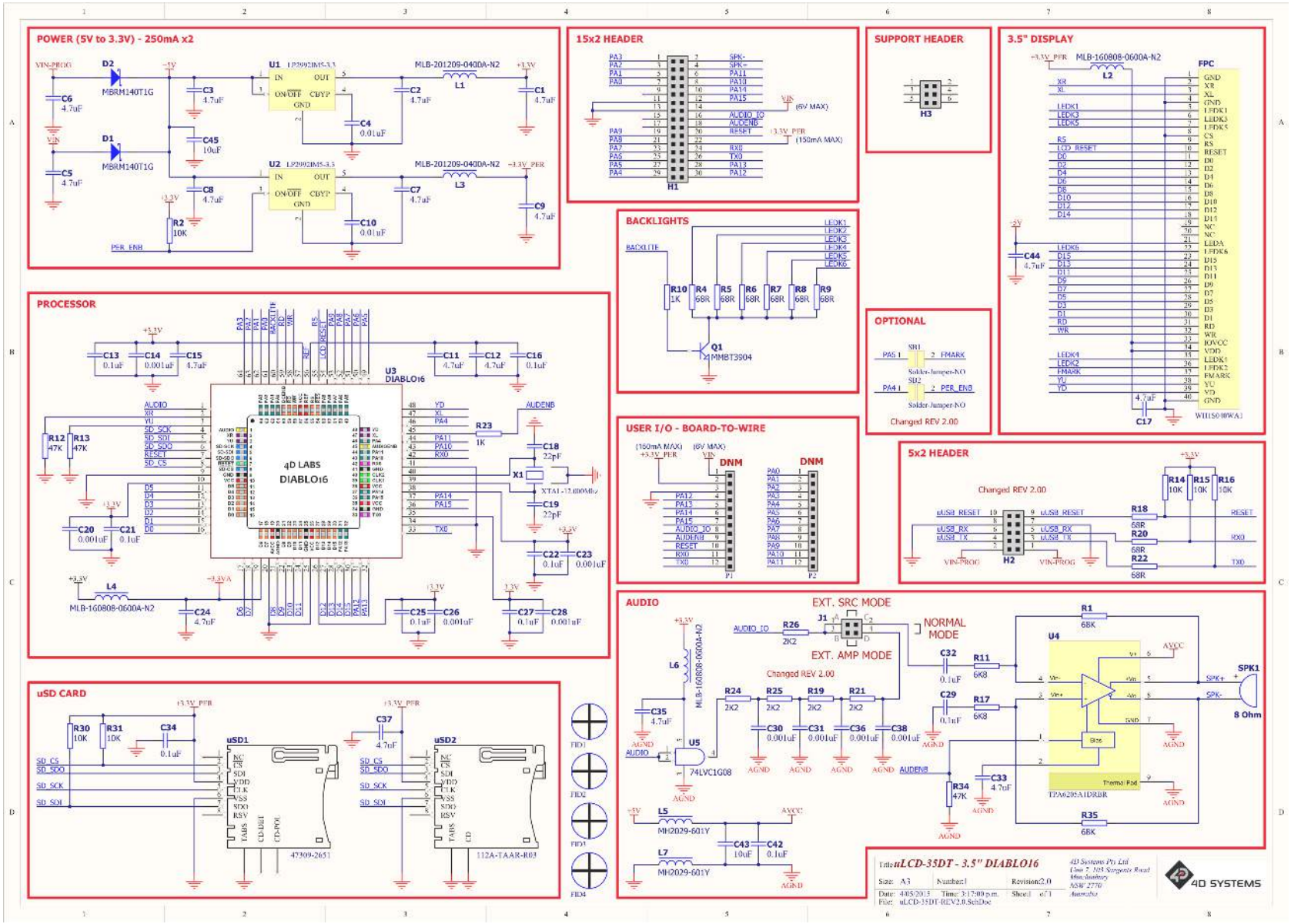
- uLCD-35DT Display Module
- 4GB MicroSD Card
- uUSB-PA5-II Programming Adaptor
- 150mm 5-way Female-Female jumper cable, for quick connection to another device or breadboard
- 5-way Male-Male adaptor (for converting the Female-Female cable to be Male-Female)

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

12. Mechanical Details



13. Schematic Diagram



14. 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	Units
Supply Voltage (VCC)	Stable external supply required	4.0	5.0	5.5	V
Processor voltage (VP)		--	3.3	--	V
Operating Temperature		-10	--	+60	°C
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	--	3.3	V
Input High Voltage (VIH)	PA4-PA13, RX0 and TX0 pins	0.8VP	--	5.5	V
Reset Pulse	External Open Collector	2.0	--	--	µs
Operational Delay	Power-Up or External Reset	500	--	3000	ms

Global Characteristics Based on Operating Conditions

Parameter	Conditions	Min	Typ	Max	Units
Supply Current (ICC)	3.3V, heavily depends on screen usage conditions, sleep mode, audio, SD card	--	250	--	mA
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	--	--	20	pF
Flash Memory Endurance	PICASO PmmC Programming	--	1000	--	E/W
Display Endurance	Hours of operation, measured to when the display is 50% original brightness	--	20000	--	H

ORDER INFORMATION

Order Code: uLCD-35DT

Packaging: Module sealed in an antistatic foam padded 4D Systems box

15. Revision History

Hardware Revision

Revision Number	Date	Description
1.03	25/11/13	Initial Public Release

Datasheet Revision

Revision Number	Date	Description
1.5	17/10/17	Initial Public Release
1.6	4/3/19	Cosmetic Changes to micro LCD Datasheet range
1.7	11/11/2019	Updated details regarding Serial Port voltage tolerance
1.8	12/01/2023	Modified datasheet for web-based documentation.