DATASHEET

# 4D SYSTEMS

*TURNING TECHNOLOGY INTO ART*

# GOLDELOX Processor

## Embedded Graphics Processor

**Document Date: 6th September 2012**
**Document Revision: 1.0**

# Contents

## 1. Description

The GOLDELOX is a custom embedded graphics controller designed to interface with many popular OLED and LCD display panels. Powerful graphics, text, image, animation and countless more features are built right inside the chip. It offers a simple plug-n-play interface to many 8bit 80-Series colour LCD and OLED displays.

The chip is designed to work with minimal design effort and all of the data and control signals are provided by the chip to interface directly to the display. Simply choose your display and interface it to the GOLDELOX on your application board. This offers enormous advantage to the designer in development time and cost saving and takes away all of the burden of low level design.

The GOLDELOX belongs to a family of processors powered by a highly optimised soft core virtual engine, EVE (Extensible Virtual Engine). EVE is a proprietary, high performance virtual processor 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.

The device offers modest but comprehensive I/O features and can interface to SPI, serial, analogue, digital, buttons, joystick and Dallas 1-wire devices. Provision is also made for creating complex sound effects for audible user feedback with an extended RTTTL tone generator.

All of the display built-in driver libraries implement and share the same high-level function interface. This allows your GUI application to be portable to different display controller types.

4D Systems software development IDE called Workshop 4 is FREE and there are no licensing requirements.

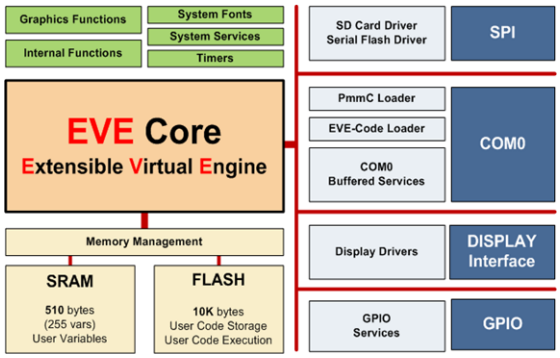
The GOLDELOX offers one of the most flexible embedded graphics solutions available.

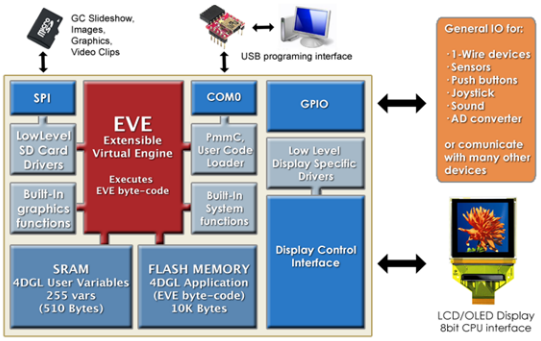
## 2. Features

- Low-cost OLED, LCD and TFT display graphics user interface solution.
- Ideal as a standalone embedded graphics processor or interface to any host controller as a graphics co-processor.
- Connect to any colour display that supports an 80-Series 8 bit wide CPU interface. All data and control signals are provided.
- Built in high performance virtual processor engine (EVE) with an extensive byte-code instruction set optimised for 4DGL, the high level 4D Graphics Language.
- 2 x GPIO ports supports:
  - Digital I/O
  - A/D converter with 8/10 bit resolution
  - Complex sound generation
  - Dedicated RTTTL tune engine
  - Multi-Switch Joystick
  - Dallas 1-Wire
- 10KB of Flash memory for user code storage and 510 bytes (255 x 16bit vars) of RAM for user variables.
- 1 x 32bit free running system timer with 1msec resolution.
- 4 x 16bit user timers with 1msec resolution
- Asynchronous hardware Serial port with auto-baud feature (300 to 256K baud).
- Hardware SPI port interface for micro-SD/micro-SDHC memory cards or Serial Flash memory chips for storing of icons, images, animations, etc.
- Comprehensive set of built in high level 4DGL graphics functions and algorithms that can draw lines, circles, text, and much more.
- Display full colour images, animations, icons and video clips.
- 8x8 built-in system font and support for unlimited user customisable fonts with fixed or proportional spacing with the aid of a freely provided Font-Tool.
- Single 3.3 Volt Supply @12mA typical.
- Available in a tiny 6mm x 6mm 28pin QFN.

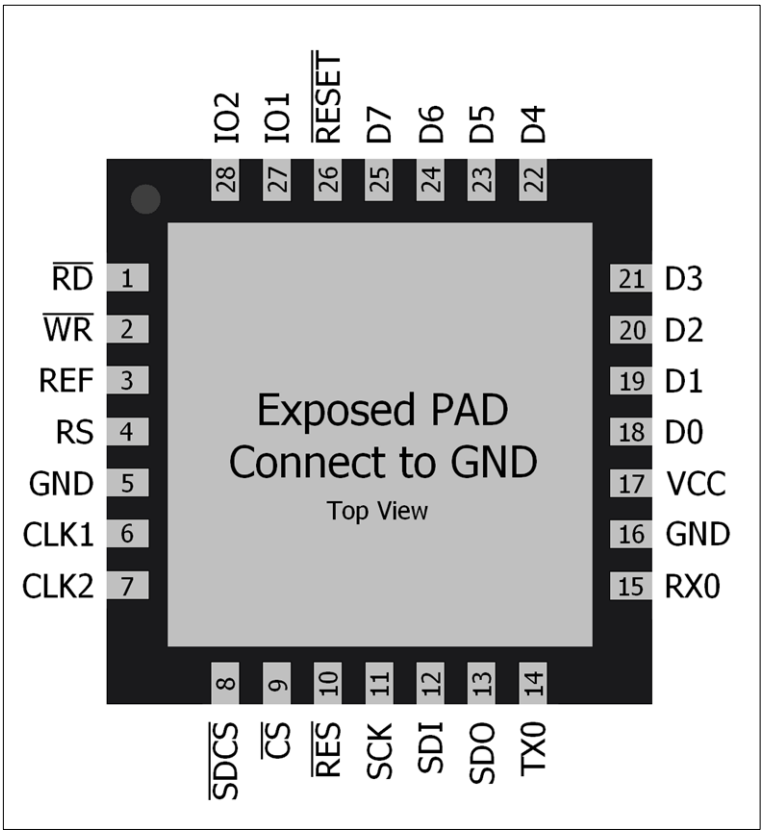
## 3. Applications

- Industrial (general).
- Test, measurement and general purpose instrumentation
- Elevator Control Systems.
- Point of Sale Terminals.
- Home Appliances (general).
- Security Systems.
- Access Control Systems.
- Air-conditioning Control Systems.
- Universal Remote Control.
- Automotive (general).
- Electronic Gauges and Meters.
- Portable ECG Systems.
- Portable Blood Pressure Monitors.
- Aviation (general).
- Gaming and Slot Machines.
- And much more..



## 4. Pin Configuration and Summary




| GOLDELOX Processor Pin Out | | | |
|---|---|---|---|
| **Pin** | **Symbol** | **I/O** | **Description** |
| 1 | RD | O | Display Read strobe signal. GOLDELOX asserts this signal LOW when reading data from the display. Connect this pin to the Read (RD) signal of the display. |
| 2 | WR | O | Display Write strobe signal. GOLDELOX asserts this signal LOW when writing data to the display. Connect this pin to the Write (WR) signal of the display. |
| 3 | REF | P | Internal voltage regulator filter capacitor. Connect a 4.7uF to 10uF capacitor from this pin to Ground. |
| 4 | RS | O | Display Register Select.<br>LOW: Display index or status register is selected.<br>HIGH: Display GRAM or register data is selected.<br>Connect this pin to the Register Select (RS or A0 or C/D or similar naming convention) signal of the display. |
| 5 | GND | P | Ground. |
| 6 | CLK1 | I | System Clock input 1 of a 12MHz crystal. |
| 7 | CLK2 | O | System Clock input 2 of a 12MHz crystal. |
| 8 | SDCS | O | SPI device Chip Select. Connect this pin to the Chip Enable (CE or CS) signal of the external SPI device (SD/SDHC memory card, Serial Flash chip, etc.). |
| 9 | CS | O | Display Chip Select. GOLDELOX asserts this signal LOW when accessing the display. Connect this pin to the Chip Select (CS) signal of the display. |
| 10 | RES | O | Display RESET. GOLDELOX initialises the display by strobing this pin LOW. Connect this pin to the Reset (RES) signal of the display. |

I = Input, O = Output, P = Power, A = Analogue

| GOLDELOX Processor Pin Out (continued…) | | | |
|---|---|---|---|
| **Pin** | **Symbol** | **I/O** | **Description** |
| 11 | SCK | O | SPI Serial Clock output. Connect this pin to the SPI Serial Clock (SCK) signal of the external device. Nominally reserved for SD/SDHC memory card or serial flash memory chip.<br>See Section 8.3 for detailed timing diagram. |
| 12 | SDI | I | SPI Serial Data Input. Connect this pin to the SPI Serial Data Out (SDO) signal of the external device. Nominally reserved for SD/SDHC memory card or serial flash memory chip.<br>See Section 8.3 for detailed timing diagram. |
| 13 | SDO | O | SPI Serial Data Output. Connect this pin to the SPI Serial Data In (SDI) signal of the external device. Nominally reserved for SD/SDHC memory card or serial flash memory chip.<br>See Section 8.3 for detailed timing diagram. |
| 14 | TX0 | O | Asynchronous Serial Transmit pin. Output data is at TTL voltage levels. Connect this pin to external device Serial Receive (Rx) signal. This pin is tolerant up to 5.0V levels. |
| 15 | RX0 | I | Asynchronous Serial Receive pin. Connect this pin to external device Serial Transmit (Tx) signal. This pin is tolerant up to 5.0V levels. |
| 16 | GND | P | Ground. |
| 17 | VCC | P | Positive supply with respect to GND pin. |
| 18 | D0 | I/O | Display Data Bus bit 0. |
| 19 | D1 | I/O | Display Data Bus bit 1. |
| 20 | D2 | I/O | Display Data Bus bit 2. |
| 21 | D3 | I/O | Display Data Bus bit 3. |
| 22 | D4 | I/O | Display Data Bus bit 4. |
| 23 | D5 | I/O | Display Data Bus bit 5. |
| 24 | D6 | I/O | Display Data Bus bit 6. |
| 25 | D7 | I/O | Display Data Bus bit 7. |
| 26 | RESET | I | Master Reset signal. Connect a 4.7K resistor from this pin to VCC. |
| 27 | IO1 | I/O/A | General purpose IO1 pin. See Section 2.4 for more detail. |
| 28 | IO2 | I/O | General purpose IO2 pin. See Section 2.4 for more detail. |
| **PAD** | **GND** | **P** | **Exposed metal pad under the package, must connect to GND.** |

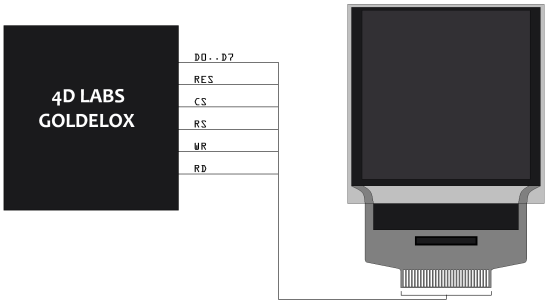
I = Input, O = Output, P = Power, A = Analogue

## 5. Hardware Interface - Pins

The GOLDELOX provides both a hardware and software interface. This section describes in detail the hardware interface.

### 5.1. Display Interface

The GOLDELOX supports LCD and OLED displays with an 80-Series 8 bit wide CPU data interface. The connectivity to the display is easy and straight forward. The chip generates all of the necessary timing to drive the display.




| CS | RS | RD | WR | Operation |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | Read Display Status Register |
| 0 | 0 | 1 | 0 | Write Display Index Register |
| 0 | 1 | 0 | 1 | Read Display GRAM Data |
| 0 | 1 | 1 | 0 | Write Register or GRAM Data |
| 1 | X | X | X | No Operation |

**Display Operation Table**

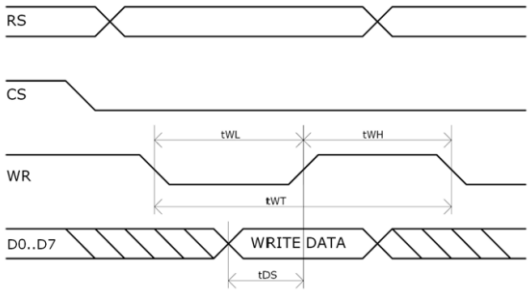
**D0-D7 pins (Display Data Bus):**
The Display Data Bus (D0-D7) is an 8 bit bidirectional port and all data writes and reads occur over this bus. Other control signals such as RW, RD CS, and RS synchronise the data transfer to and from the display.

**CS pin (Display Chip Select):**
The access to the display is only possible when the Display Chip Select (CS) is asserted LOW. Connect this pin to the Chip Select (CS) signal of the display.

**RS pin (Display Register Select):**
The RS signal determines whether a register command or data is sent to the display.
LOW: Display index or status register is selected.
HIGH: Display GRAM or register data is selected.
Connect this pin to the Register Select (RS) signal of the display. Different displays utilise various naming conventions such as RS, A0, C/D or similar. Be sure to check with your display manufacturer for the correct name and function.

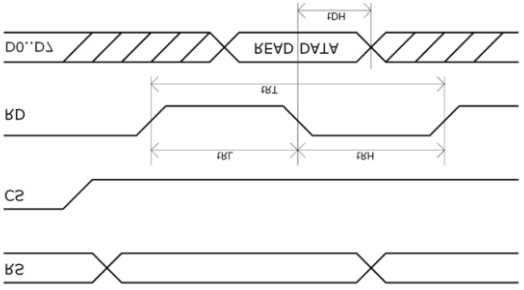
**RES pin (Display Reset):**
Display RESET. GOLDELOX initialises the display by strobing this pin LOW. Connect this pin to the Reset (RES) signal of the display. This signal can also be used to control the back-light of the LCD or as the DC/DC converter enable. Refer to the reference design in Section 7 in this document for an example.

**WR pin (Display Write):**
This is the display write strobe signal. The GOLDELOX asserts this signal LOW when writing data to the display in conjunction with the display data bus (D0-D7). Connect this pin to the Write (WR) signal of the display.




| Item | Sym | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Write Low Pulse | tWL | 170 | - | - | ns |
| Write High Pulse | tWH | 85 | - | - | ns |
| Write Bus Cycle Total | tWT | 255 | - | - | ns |
| Write Data Setup | tDS | 85 | - | - | ns |

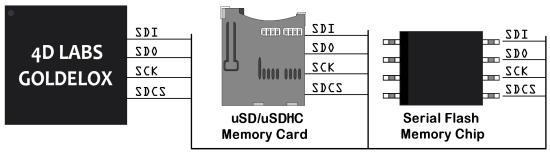
**RD pin (Display Read):**
This is the display read strobe signal. The GOLDELOX asserts this signal LOW when reading data from the display in conjunction with the display data bus (D0-D7). Connect this pin to the Read (RD) signal of the display.



| Item | Sym | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Read Low Pulse | tRL | 300 | - | - | ns |
| Read High Pulse | tRH | 300 | - | - | ns |
| Read Bus Cycle Total | tRT | 600 | - | - | ns |
| Read Data Hold | tDH | 150 | - | - | ns |

## 5.2. SPI Interface – Master Mode Only

The GOLDELOX supports micro-SD/micro-SDHC memory cards as well as Serial Flash memory chips via its hardware SPI interface. These storage devices are used for all multimedia file storage 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 micro-SD with up to 2GB capacity and for high capacity HC memory cards starting from 4GB and above. The GOLDELOX also supports any other general purpose SPI serial device.




**SDI pin (SPI Serial Data In):**
The SPI Serial Data Input (SDI). It connects to the Serial Data Out (SDO) pin of external SPI device.

**SDO pin (SPI Serial Data Out):**
The SPI Serial Data Output (SDO). This pin connects to the Serial Data In (SDI) signal of the external SPI device.

**SCK pin (SPI Serial Clock):**
The SPI Serial Clock output (SCK). This pin connects to the Serial Clock (SCK) signal of the external SPI device.

**SDCS pin (SPI Chip Select):**
SPI device Chip Select (SDCS). Connect this pin to the Chip Enable (CE or CS) signal of the external SPI device.
Also refer to "**Section 13.3 SPI Timing Diagram**"

## 5.3. Serial Port - UART

The GOLDELOX has a dedicated hardware UART that can communicate with external serial devices. This is referred to as the COM0 module. The primary features are:

- Full-Duplex 8 bit data transmission and reception through the TX and RX pins.
- Data format: 8 bits, No Parity, 1 Stop bit.
- Auto Baud feature.
- Baud rates from 300 baud up to 256K baud.
- Single byte transmits and receives or 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 the serial port. This frees up the application to service other tasks.

The Serial port is also the primary interface for downloading user application code (compiled 4DGL byte-code) into the GOLDELOX flash program memory. Once the download is complete the serial port is available for user application.

**Note: Low level PmmC chip programming and updates also take place via the serial port.**

Refer to "**Section 4. In-Circuit-Serial-Programming (ICSP)**" for further details.

**TX pin (Serial Transmit):**
Asynchronous Serial port Transmit pin, TX. Connect this pin to external serial device Serial Receive (Rx) signal.

**RX pin (Serial Receive):**
Asynchronous Serial port Receive pin, RX. Connect this pin to external serial device Serial Transmit (Tx) signal.

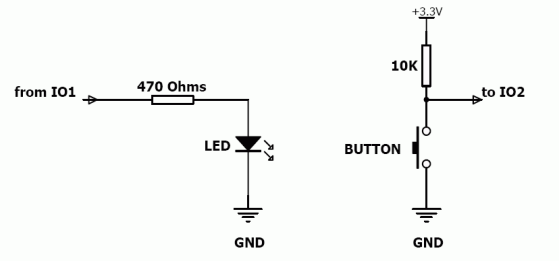
## 5.4. General Purpose I/O Interface

There are 2 GPIO pins available, **IO1** and **IO2**. Each GPIO has a multitude of high level functions associated with it and these can be selected within 4DGL user application code.
Refer to the separate document titled ***"GOLDELOX-Internal-Functions.pdf"*** for a complete set of built in 4DGL library functions.

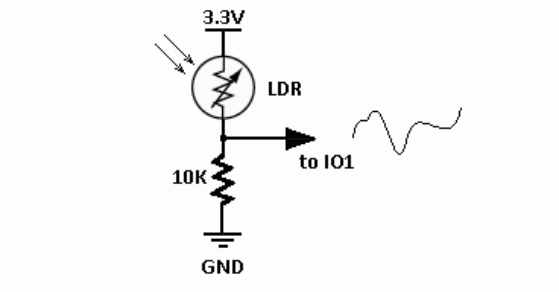
**IO1, IO2 pins (General Purpose Input Output):**
General purpose IO1, IO2 pins. The table below lists the available GPIO functions and features.

| GPIO Functions and Features | | |
|---|---|---|
| **Function** | **IO1** | **IO2** |
| Digital Input | √ | √ |
| Digital Output | √ | √ |
| A/D Converter 8/10 bits | √ | -- |
| Dallas 1-Wire support | √ | √ |
| Sound Generation, RTTTL Tunes | √ | √ |
| Joystick – 5 position multi-switch | √ | -- |

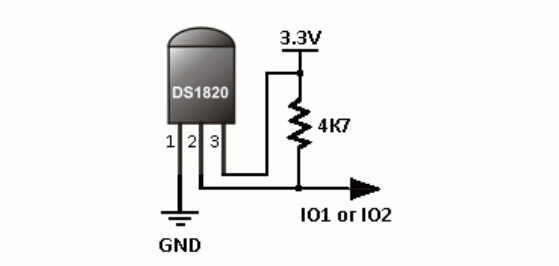
**Input/Output:**
Both IO1 and IO2 pins can be programmed to be Inputs or Outputs. Diagram below shows a LED connected to IO1 (programmed as an output) and a button connected to IO2 (programmed as an input).



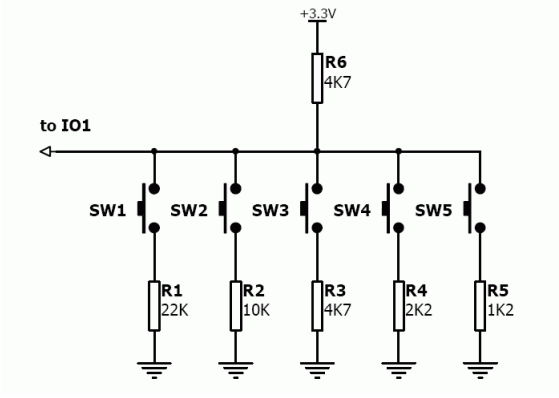

**Analogue to Digital Converter:**
The IO1 pin can be programmed as an A/D input. Option is available to select 8 bit or 10 bit resolution. Diagram below is a circuit of a Light Dependant Resistor (LDR) connected to IO1 to measure and record changes in ambient light.

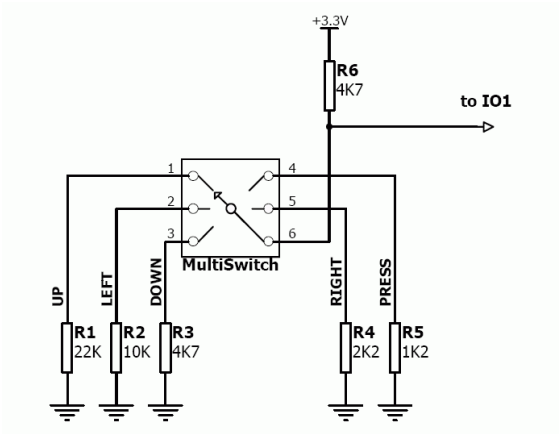



**Dallas 1-Wire:**
The Dallas 1-Wire protocol is a form of serial communications designed to operate over a single data line plus ground reference. Multiple 1-Wire devices can be attached to the same shared data line to network many devices. One wire device support is available on both the IO1 and the IO2 pins.
The diagram below depicts a typical 1-Wire temperature sensor interface.




**Joystick - Multi Switch:**
Multiple buttons or a multi-switch Joystick can be connected to the IO1 pin. Up to 5 buttons or a 5 position multi-switch joystick connects to a junction of a resistor ladder network that forms a voltage divider. The A/D converter of the IO1 pin internally reads the analogue value and decodes it accordingly. This feature is supported by dedicated 4DGL library functions. The following diagrams indicate how to connect up to 5 individual buttons or a multi-switch joystick to the IO1 pin.

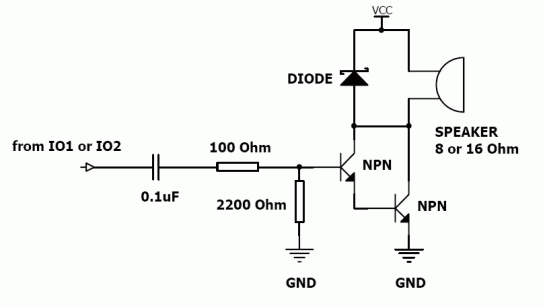



Unused buttons do not need resistors to be connected to the circuit. The table below lists the buttons and corresponding resistor values.

| Number of Buttons | Button Number | Resistor Value |
|---|---|---|
| 1 | SW1 | 22K |
| 2 | SW2 | 10K |
| 3 | SW3 | 4.7K |
| 4 | SW4 | 2.2K |
| 5 | SW5 | 1.2K |

**Sound Output:**




The GOLDELOX is capable of generating complex sounds and RTTTL tunes from its IO1 and IO2 pins. A simple speaker circuit as shown below can be utilised.

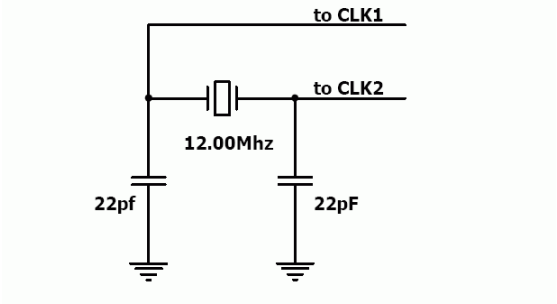
## 5.5. System Pins

**VCC pin (Device Supply Voltage):**
Device supply voltage pin. This pin must be connected to a regulated supply voltage in the range of 3.0 Volts to 3.6 Volts DC. Nominal operating voltage is 3.3 Volts.

**GND, PAD pins (Device Ground):**
Device ground pins. These pins must be connected to ground.

**RESET pin (Device Master Reset):**
Device Master Reset pin. An active low pulse of greater than 2 micro-seconds will reset the device. Connect a resistor (1K through to 10K, nominal 4.7K) from this pin to VCC. Only use open collector type circuits to reset the device if an external reset is required. This pin is not driven low by any internal conditions.

**CLK1, CLK2 pins (Device Oscillator Inputs):**
CLK1 and CLK2 are the device oscillator pins. Connect a 12MHz AT strip cut crystal with 22pF capacitors from each pin to GND as shown in the diagram below.



## 6. 4DGL - Software Language

The GOLDELOX graphics processor belongs to a family of processors powered by a highly optimised soft core 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.

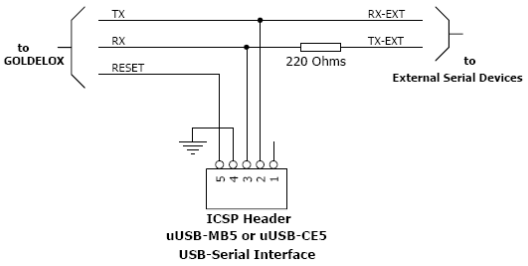
For detailed information pertaining to the 4DGL language, please refer to the following documents:
**"4DGL-Programmers-Reference-Manual.pdf"**
**"GOLDELOX-4DGL-Internal-Functions.pdf"**

To assist with the development of 4DGL applications, the Workshop 4 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.

## 7. In Circuit Serial Programming ICSP

The GOLDELOX processor can be re-programmed with the latest PmmC configuration for updates and future proofing. The chip-level configuration is available as a PmmC (Personality-module-micro-Code) file and the programming must be performed over the serial interface. The chip-resident internal 4DGL functions are part of the GOLDELOX PmmC configuration file so please check regularly for the latest updates and enhancements.

A PmmC file can only be programmed into the device via its serial port and an access to this must be provided for on the target application board. This is referred to as In Circuit Serial Programming (ICSP). Diagram below provides a typical implementation for the ICSP interface.




**ICSP Interface**

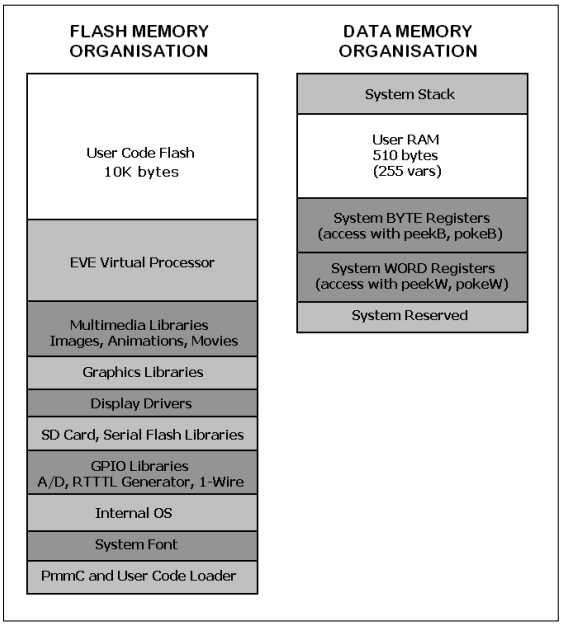
The PmmC file is programmed into the device with the aid of Workshop 4, the 4D Systems IDE software (See Section 11). To provide a link between the PC and the ICSP interface, a specific 4D Programming Cable is required and is available from 4D Systems.

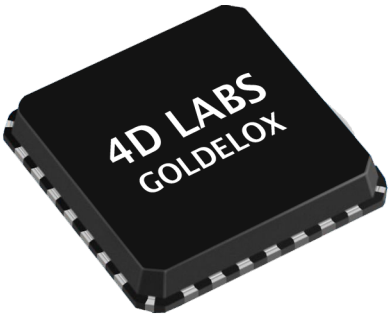
Using a non-4D programming interface could damage your display, and **void your Warranty.**

**Note: The GOLDELOX chip is shipped blank and it must be programmed with the PmmC configuration file.**

## 8. Memory Organisation

The figure below illustrates how the GOLDELOX internal memory is organised.



## 9. System Registers Memory Map

The following tables outline in detail the GOLDELOX system registers and flags.

**Table 5.1: System (BYTE Size) Registers Memory Map**

| LABEL | ADDRESS | | USAGE | SIZE | *NOTES |
|---|---|---|---|---|---|
| | DEC | HEX | | | |
| VX1 | 128 | 0x80 | display hardware GRAM x1 pos | BYTE | SYSTEM (R/O) |
| VY1 | 129 | 0x81 | display hardware GRAM y1 pos | BYTE | SYSTEM (R/O) |
| VX2 | 130 | 0x82 | display hardware GRAM x2 pos | BYTE | SYSTEM (R/O) |
| VY2 | 131 | 0x83 | display hardware GRAM y2 pos | BYTE | SYSTEM (R/O) |
| SYS_X_MAX | 132 | 0x84 | display hardware X res-1 | BYTE | SYSTEM (R/O) |
| SYS_Y_MAX | 133 | 0x85 | display hardware Y res-1 | BYTE | SYSTEM (R/O) |
| WRITE_GRAM_REG | 134 | 0x86 | display GRAM write address | BYTE | SYSTEM (R/O) |
| READ_GRAM_REG | 135 | 0x87 | display GRAM read address | BYTE | SYSTEM (R/O) |
| IMAGE_WIDTH | 136 | 0x88 | loaded image/animation width | BYTE | SYSTEM (R/O) |
| IMAGE_HEIGHT | 137 | 0x89 | loaded image/animation height | BYTE | SYSTEM (R/O) |
| IMAGE_DELAY | 138 | 0x8A | frame delay (if animation) | BYTE | USER |
| IMAGE_MODE | 139 | 0x8B | image/animation colour mode | BYTE | SYSTEM (R/O) |
| CLIP_LEFT_POS | 140 | 0x8C | left clipping point setting | BYTE | USER |
| CLIP_TOP_POS | 141 | 0x8D | top clipping point setting | BYTE | USER |
| CLIP_RIGHT_POS | 142 | 0x8E | right clipping point setting | BYTE | USER |
| CLIP_BOTTOM_POS | 143 | 0x8F | bottom clipping point setting | BYTE | USER |
| CLIP_LEFT | 144 | 0x90 | left clipping point active | BYTE | USER |
| CLIP_TOP | 145 | 0x91 | top clipping point active | BYTE | USER |
| CLIP_RIGHT | 146 | 0x92 | right clipping point active | BYTE | USER |
| CLIP_BOTTOM | 147 | 0x93 | bottom clipping point active | BYTE | USER |
| FONT_TYPE | 148 | 0x94 | 0 = fixed, 1 = proportional | BYTE | SYSTEM (R/O) |
| FONT_MAX | 149 | 0x95 | number of chars in font set | BYTE | SYSTEM (R/O) |
| FONT_OFFSET | 150 | 0x96 | ASCII offset (usually 0x20) | BYTE | SYSTEM (R/O) |
| FONT_WIDTH | 151 | 0x97 | width of font (pixel units) | BYTE | SYSTEM (R/O) |
| FONT_HEIGHT | 152 | 0x98 | height of font (pixel units) | BYTE | SYSTEM (R/O) |
| TEXT_XMAG | 153 | 0x99 | text width magnification | BYTE | USER |
| TEXT_YMAG | 154 | 0x9A | text height magnification | BYTE | USER |
| TEXT_MARGIN | 155 | 0x9B | text place holder for CR | BYTE | SYSTEM (R/O) |
| TEXT_DELAY | 156 | 0x9C | text delay effect (0-255msec) | BYTE | USER |
| TEXT_X_GAP | 157 | 0x9D | X pixel gap between chars | BYTE | USER |
| TEXT_Y_GAP | 158 | 0x9E | Y pixel gap between chars | BYTE | USER |
| GFX_XMAX | 159 | 0x9F | width of current orientation | BYTE | SYSTEM (R/O) |
| GFX_YMAX | 160 | 0xA0 | height of current orientation | BYTE | SYSTEM (R/O) |
| GFX_SCREENMODE | 161 | 0xA1 | Current screen mode (0-3) | BYTE | SYSTEM (R/O) |
| reserved | 162-165 | 0xA2-0xA5 | reserved | BYTE | SYSTEM (R/O) |

***NOTES:**

| | |
|---|---|
| **SYSTEM** | SYSTEM registers are maintained by internal system functions and should not be written to. They should only ever be read. **DO NOT WRITE to these registers.** |
| **USER** | USER registers are read/write (R/W) registers used to alter the system behaviour. Refer to the individual functions for information on the interaction with these registers. |

These registers are accessible with **peekB** and **pokeB** functions.

| Table 5.2: System (WORD size) Registers Memory Map | | | | | |
|---|---|---|---|---|---|
| LABEL | ADDRESS | | USAGE | SIZE | *NOTES |
| | DEC | HEX | | | |
| SYS_OVERFLOW | 83 | 0x53 | 16bit overflow register | WORD | USER |
| SYS_COLOUR | 84 | 0x54 | internal variable for colour | WORD | SYSTEM |
| SYS_RETVAL | 85 | 0x55 | return value of last function | WORD | SYSTEM |
| GFX_BACK_COLOUR | 86 | 0x56 | screen background colour | WORD | USER |
| GFX_OBJECT_COLOUR | 87 | 0x57 | graphics object colour | WORD | USER |
| GFX_TEXT_COLOUR | 88 | 0x58 | text foreground colour | WORD | USER |
| GFX_TEXT_BGCOLOUR | 89 | 0x59 | text background colour | WORD | USER |
| GFX_OUTLINE_COLOUR | 90 | 0x5A | circle/rectangle outline | WORD | USER |
| GFX_LINE_PATTERN | 91 | 0x5B | line draw tessellation | WORD | USER |
| IMG_PIXEL_COUNT | 92 | 0x5C | count of pixels in image | WORD | SYSTEM |
| IMG_FRAME_COUNT | 93 | 0x5D | count of frames in animation | WORD | SYSTEM |
| MEDIA_HEAD | 94 | 0x5E | media sector head position | WORD | SYSTEM |
| SYS_OUTSTREAM | 95 | 0x5F | Output stream handle | WORD | SYSTEM |
| GFX_LEFT | 96 | 0x60 | image left real point | WORD | SYSTEM |
| GFX_TOP | 97 | 0x61 | image top real point | WORD | SYSTEM |
| GFX_RIGHT | 98 | 0x62 | image right real point | WORD | SYSTEM |
| GFX_BOTTOM | 99 | 0x63 | image bottom real point | WORD | SYSTEM |
| GFX_X1 | 100 | 0x64 | image left clipped point | WORD | SYSTEM |
| GFX_Y1 | 101 | 0x65 | image top clipped point | WORD | SYSTEM |
| GFX_X2 | 102 | 0x66 | image right clipped point | WORD | SYSTEM |
| GFX_Y2 | 103 | 0x67 | image bottom clipped point | WORD | SYSTEM |
| GFX_X_ORG | 104 | 0x68 | current X origin | WORD | USER |
| GFX_Y_ORG | 105 | 0x69 | current Y origin | WORD | USER |
| RANDOM_LO | 106 | 0x6A | random generator LO word | WORD | SYSTEM |
| RANDOM_HI | 107 | 0x6B | random generator HI word | WORD | SYSTEM |
| MEDIA_ADDR_LO | 108 | 0x6C | media byte address LO | WORD | SYSTEM |
| MEDIA_ADDR_HI | 109 | 0x6D | media byte address HI | WORD | SYSTEM |
| SECTOR_ADDR_LO | 110 | 0x6E | media sector address LO | WORD | SYSTEM |
| SECTOR_ADDR_HI | 111 | 0x6F | media sector address HI | WORD | SYSTEM |
| SYSTEM_TIMER_LO | 112 | 0x70 | 1msec system timer LO word | WORD | USER |
| SYSTEM_TIMER_HI | 113 | 0x71 | 1msec system timer HI word | WORD | USER |
| TIMER0 | 114 | 0x72 | 1msec user timer 0 | WORD | USER |
| TIMER1 | 115 | 0x73 | 1msec user timer 1 | WORD | USER |
| TIMER2 | 116 | 0x74 | 1msec user timer 2 | WORD | USER |
| TIMER3 | 117 | 0x75 | 1msec user timer 3 | WORD | USER |
| INCVAL | 118 | 0x76 | predec/preinc/postdec/postinc addend | WORD | USER |
| TEMP_MEDIA_ADDRLO | 119 | 0x77 | temporary media address LO | WORD | SYSTEM |
| TEMP_MEDIA_ADDRHI | 120 | 0x78 | temporary media address HI | WORD | SYSTEM |
| GFX_TRANSPARENTCOLOUR | 121 | 0x79 | Image transparency colour | WORD | USER |
| GFX_STRINGMETRIX | 122 | 0x7A | Low byte = string width<br>High byte = string height | WORD | SYSTEM |
| GFX_TEMPSTORE1 | 123 | 0x7B | Low byte = last character printed<br>High byte = video frame timer over-ride | WORD | SYSTEM |
| reserved | 124 | 0x7C | reserved | WORD | SYSTEM |
| reserved | 125 | 0x7D | reserved | WORD | SYSTEM |
| SYS_FLAGS1 | 126 | 0x7E | system control flags word 0 | WORD | FLAGS |
| SYS_FLAGS2 | 127 | 0x7F | system control flags word 1 | WORD | FLAGS |
| USR_SP | 128 | 0x80 | User defined stack pointer | WORD | USERSTACK |
| USR_MEM | 129 | 0x81 | 255 user variables / array(s) | WORD | MEMORY |
| SYS_STACK | 384 | 0x180 | 128 level EVE machine stack | WORD | SYSTEMSTACK |

| *NOTES: | |
|---|---|
| **SYSTEM** | SYSTEM registers are maintained by internal system functions and should not be written to. They should only ever be read.<br>**DO NOT WRITE to these registers.** |
| **USER** | USER registers are read/write (R/W) registers used to alter the system behaviour. Refer to the individual functions for information on the interaction with these registers. |
| **USERSTACK** | Used by the debugging and system extension utilities |
| **MEMORY** | 255 word size variables for users program |
| **STACK** | 128 word EVE system stack (STACK grows upwards) |
| **FLAGS** | FLAGS are a mixture of bits that are either maintained by internal system functions or set / cleared by various system functions. Refer to the FLAGS Register Bit Map table, and individual functions for further details. |
| These registers are accessible with **peekW** and **pokeW** functions. | |

**Table 5.3: FLAG Registers Bit Map**

| REGISTER | ADDRESS DEC | ADDRESS HEX | NAME | USAGE | *NOTES | VALUE |
|---|---|---|---|---|---|---|
| **SYS_FLAGS1** | **126** | **0x7E** | *** denotes auto reset** | | | |
| | Bit 0 | | _STREAMLOCK | Used internally | SYSTEM | 0x0001 |
| | Bit 1 | | _PENSIZE | Object, 0 = solid, 1 = outline | SYSTEM | 0x0002 |
| | Bit 2 | | _OPACITY | Text, 0 = transparent, 1 = opaque | SYSTEM | 0x0004 |
| | Bit 3 | | _OUTLINED | box/circle outline 0 = off, 1 = on | SYSTEM | 0x0008 |
| | Bit 4 | | _BOLD | * text, 0 = normal, 1 = bold | SYSTEM | 0x0010 |
| | Bit 5 | | _ITALIC | * Text, 0 = normal, 1 = italic | SYSTEM | 0x0020 |
| | Bit 6 | | _INVERSE | * Text, 0 = normal, 1 = inverse | SYSTEM | 0x0040 |
| | Bit 7 | | _UNDERLINED | * Text, 0 = normal, 1 = underlined | SYSTEM | 0x0080 |
| | Bit 8 | | _CLIPPING | 0 = clipping off, 1 = clipping on | SYSTEM | 0x0100 |
| | Bit 9 | | _STRMODE | Used internally | SYSTEM | 0x0200 |
| | Bit 10 | | _SERMODE | Used internally | SYSTEM | 0x0400 |
| | Bit 11 | | _TXTMODE | Used internally | SYSTEM | 0x0800 |
| | Bit 12 | | _MEDIAMODE | Used internally | SYSTEM | 0x1000 |
| | Bit 13 | | _PATTERNED | Used internally | SYSTEM | 0x2000 |
| | Bit 14 | | _COLOUR8 | Display mode, 0 = 16bit, 1 = 8bit | SYSTEM | 0x4000 |
| | Bit 15 | | _MEDIAFONT | 0 = internal font, 1 = media font | SYSTEM | 0x8000 |
| **SYS_FLAGS2** | **127** | **0x7F** | | | | |
| | Bit 0 | | _MEDIA_INSTALLED | SD or FLASH device is detected/active | SYSTEM | 0x0001 |
| | Bit 1 | | _MEDIA_TYPE | 0 = SD, 1 = FLASH chip | SYSTEM | 0x0002 |
| | Bit 2 | | _MEDIA_READ | 1 = MEDIA read in progress | SYSTEM | 0x0004 |
| | Bit 3 | | _MEDIA_WRITE | 1 = MEDIA write in progress | SYSTEM | 0x0008 |
| | Bit 4 | | _OW_PIN | 0 = IO1, 1 = IO2 (Dallas OW Pin) | SYSTEM | 0x0010 |
| | Bit 5 | | _PTR_TYPE | Used internally | SYSTEM | 0x0020 |
| | Bit 6 | | _TEMP1 | Used internally | SYSTEM | 0x0040 |
| | Bit 7 | | _TEMP2 | Used internally | SYSTEM | 0x0080 |
| | Bit 8 | | _RUNMODE | 1 = running pcode from media | SYSTEM | 0x0100 |
| | Bit 9 | | _SIGNED | 0 = number printed '-' prepend | SYSTEM | 0x0200 |
| | Bit 10 | | _RUNFLAG | 1 = EVE processor is running | SYSTEM | 0x0400 |
| | Bit 11 | | _SINGLESTEP | 1 = set breakpoint for debugger | SYSTEM | 0x0800 |
| | Bit 12 | | _COMMINT | 1 = buffered coms active | SYSTEM | 0x1000 |
| | Bit 13 | | _DUMMY16 | 1 = display needs 16bit dummy | SYSTEM | 0x2000 |
| | Bit 14 | | _DISP16 | 1 = display is 16bit interface | SYSTEM | 0x4000 |
| | Bit 15 | | _PROPFONT | 1 = current font is proportional | SYSTEM | 0x8000 |

# 10. Hardware Tools

The following hardware tools are required for full control of the GOLDELOX Processor.

## 10.1. 4D Programming Cable

The 4D Programming Cable is an essential hardware tool to program, customise and test the GOLDELOX Processor.

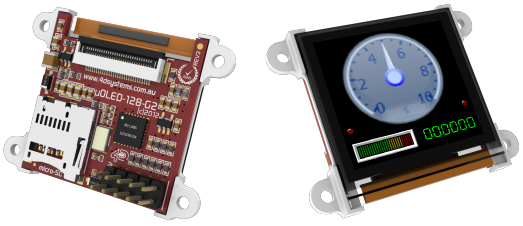
The 4D Programming Cable is used to program a new Firmware/PmmC and downloading compiled 4DGL code into the module. It even serves as an interface for communicating serial data to the PC.

The 4D Programming Cable is available from 4D Systems, www.4dsystems.com.au

*4D Programming Cable*

## 10.2. Evaluation Display Modules

The following modules, available from 4D Systems, can be used for evaluation purposes to discover what the GOLDELOX processor has to offer.



µOLED-128-G2 - 1.5" Intelligent GOLDELOX Display

Other modules, such as the 0.96" and 1.7" OLED, or 1.44" LCD versions are also available. Please contact 4D Systems for more information, or visit the 4D Systems website, www.4dsystems.com.au

# 11. 4D Systems - Workshop 4 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 Workshop 4 IDE supports multiple development environments for the user, to cater for different user requirements and skill level.

- The **Designer** environment enables the user to write 4DGL code in its natural form.
- 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.
- A **Serial** environment is also provided to transform the GOLDELOX Processor into a slave serial module, allowing the user to control the display from any host microcontroller or device with a serial port.

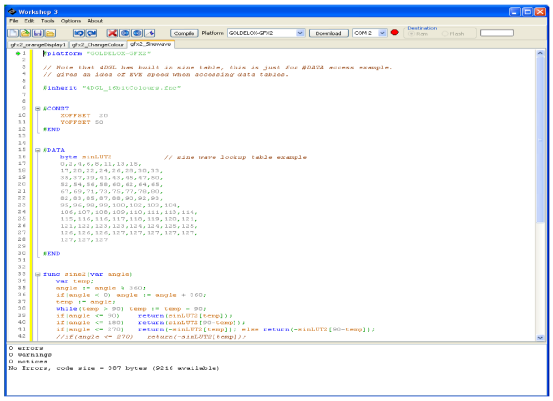
The Workshop 4 IDE is available from the 4D Systems website. www.4dsystems.com.au

For a comprehensive manual on the Workshop 4 IDE Software, refer to its documentation from the 4D Systems website.
**"Workshop-4-IDE-User-Manual.pdf"**

## 11.1. Workshop 4 – Designer Environment

Choose the Designer environment to write 4DGL code in its raw form.

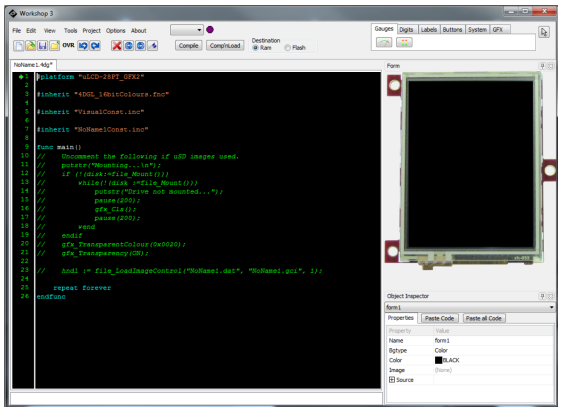
The Designer environment provides the user with a simple yet effective programming environment where pure 4DGL code can be written, compiled and downloaded to the GOLDELOX.



## 11.2. Workshop 4 – ViSi Environment

ViSi was designed to make the creation of graphical displays a more visual experience.

ViSi is a great software tool that allows the user to see the instant results of their desired graphical layout. Additionally, there is a selection of inbuilt dials, gauges and meters that can simply be placed onto the simulated module display. From here each object can have its properties edited, and at the click of a button all relevant 4DGL code associated with that object is produced in the user program. The user can then write 4DGL code around these objects to utilise them in the way they choose.



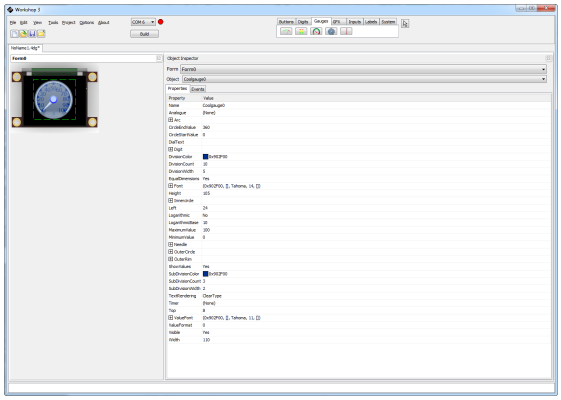
## 11.3. Workshop 4 – Serial Environment

The Serial environment in the Workshop 4 IDE provides the user the ability to transform the GOLDELOX Processor into a slave serial graphics controller.

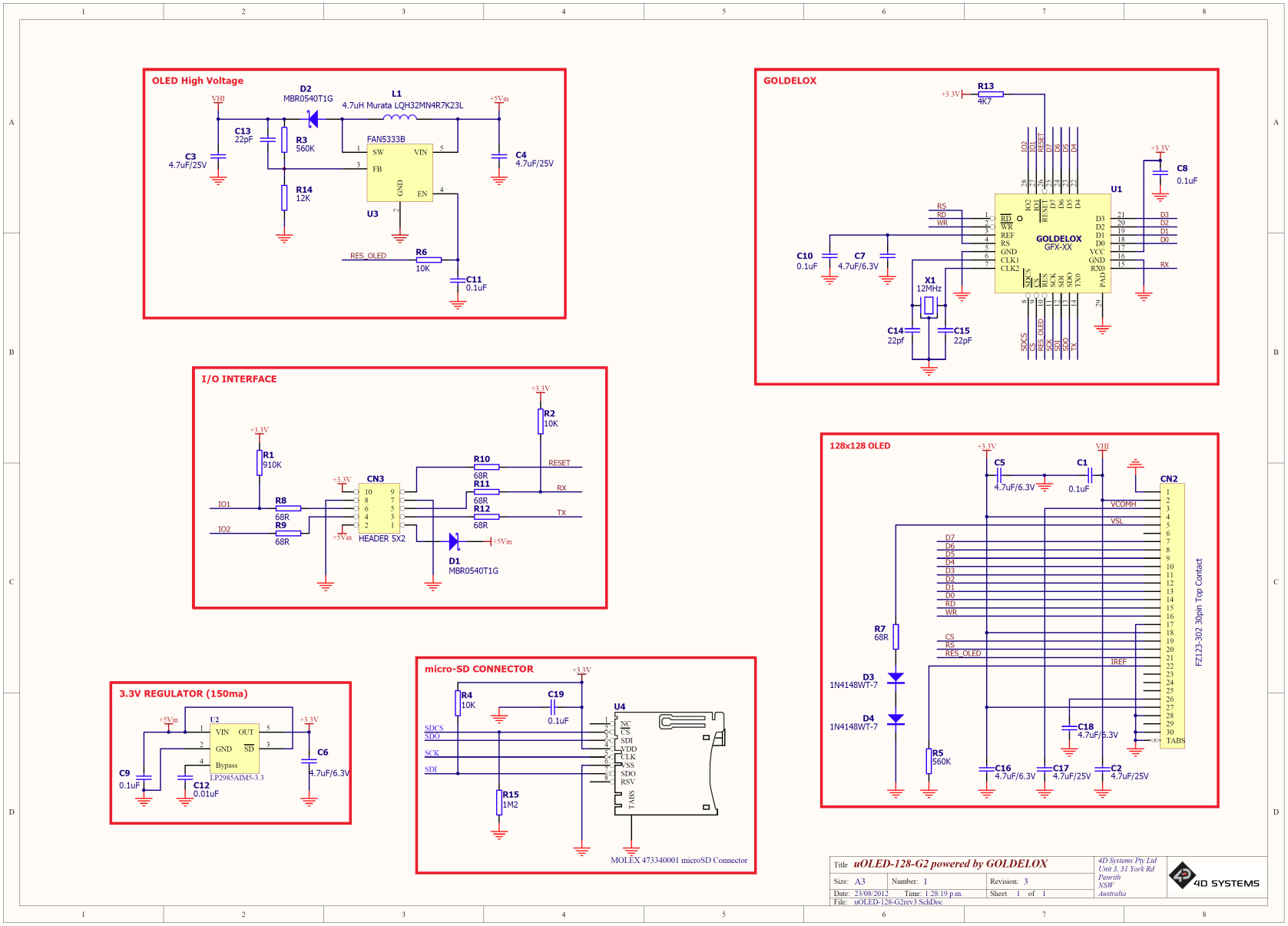
This enables the user to use their favourite microcontroller or serial device as the Host, without having to learn 4DGL or program in a separate IDE. Once the module is configured and downloaded to from the Serial Environment, simple graphic commands can be sent from the users host microcontroller to display primitives, images, sound or even video.

Refer to the Serial Environment section in the Workshop 4 user manual, for a complete listing of all the supported serial commands
**"Workshop-4-IDE-User-Manual.pdf"**

By default, each module shipped from the 4D Systems factory will come pre-programmed ready for use in the Serial mode.

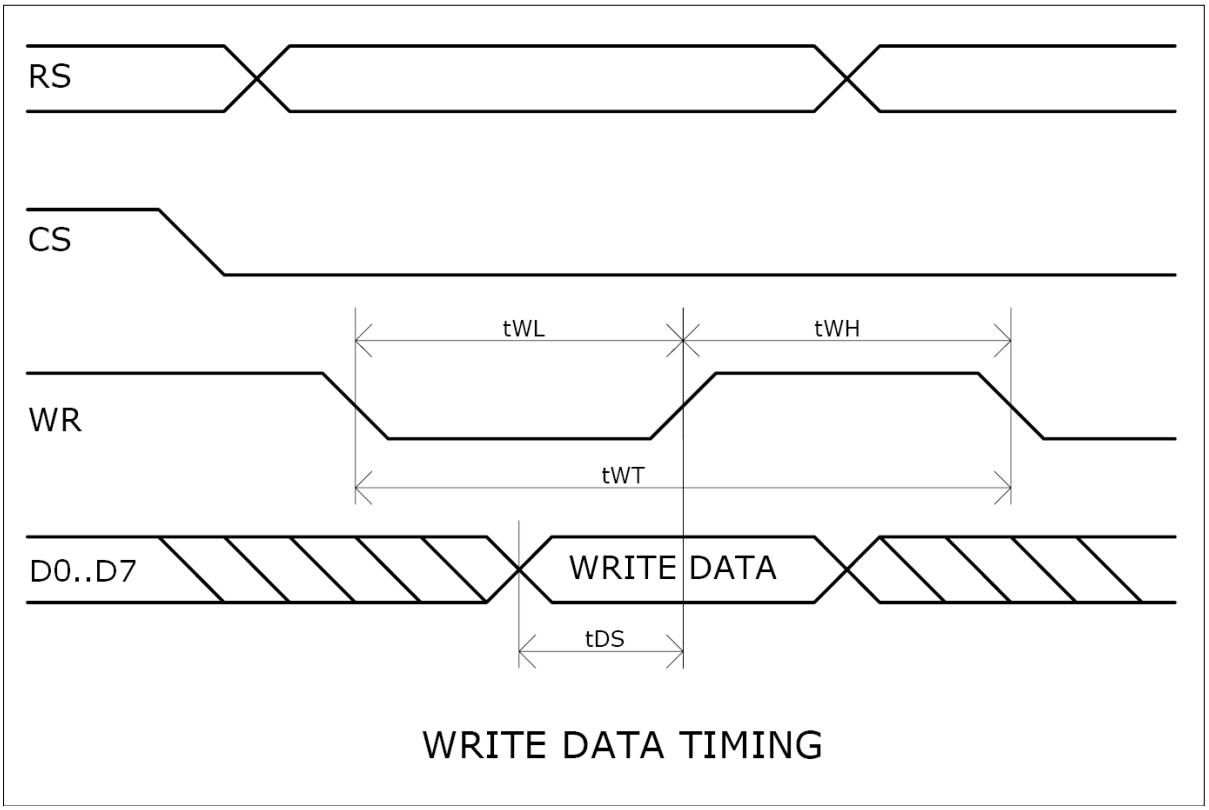


## 12. Reference Design



# 13. Timing Diagrams

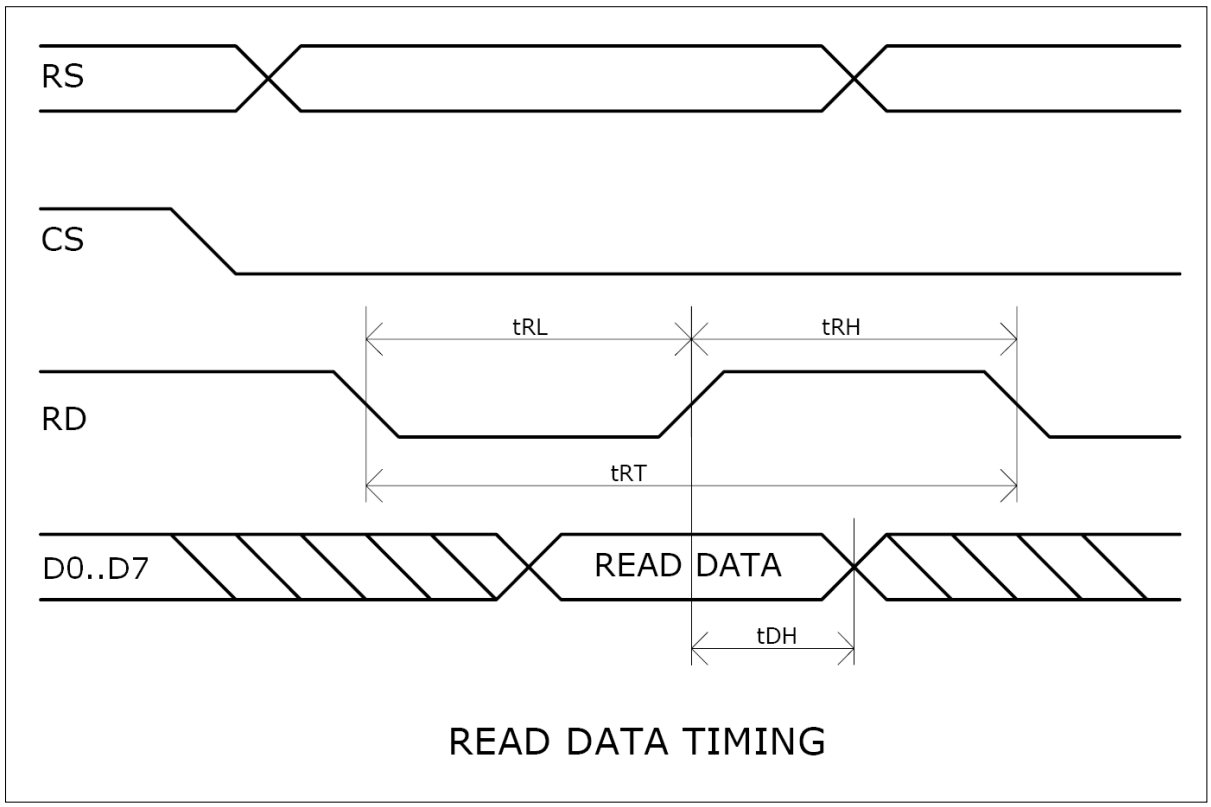
## 13.1. Display Write Data Timing




WRITE DATA TIMING

| Item | Symbol | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| Write Low level pulse width | tWL | 170 | - | - | ns |
| Write High level pulse width | tWH | 85 | - | - | ns |
| Write Bus Cycle Total | tWT | 255 | - | - | ns |
| Write Data Setup | tDS | 85 | - | - | ns |

## 13.2. Display Read Data Timing




READ DATA TIMING

| Item | Symbol | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| Read Low level pulse width | tRL | 300 | - | - | ns |
| Read High level pulse width | tRH | 300 | - | - | ns |
| Read Bus Cycle Total | tRT | 600 | - | - | ns |
| Read Data Hold | tDH | 150 | - | - | ns |

## 13.3. SPI Timing Diagram

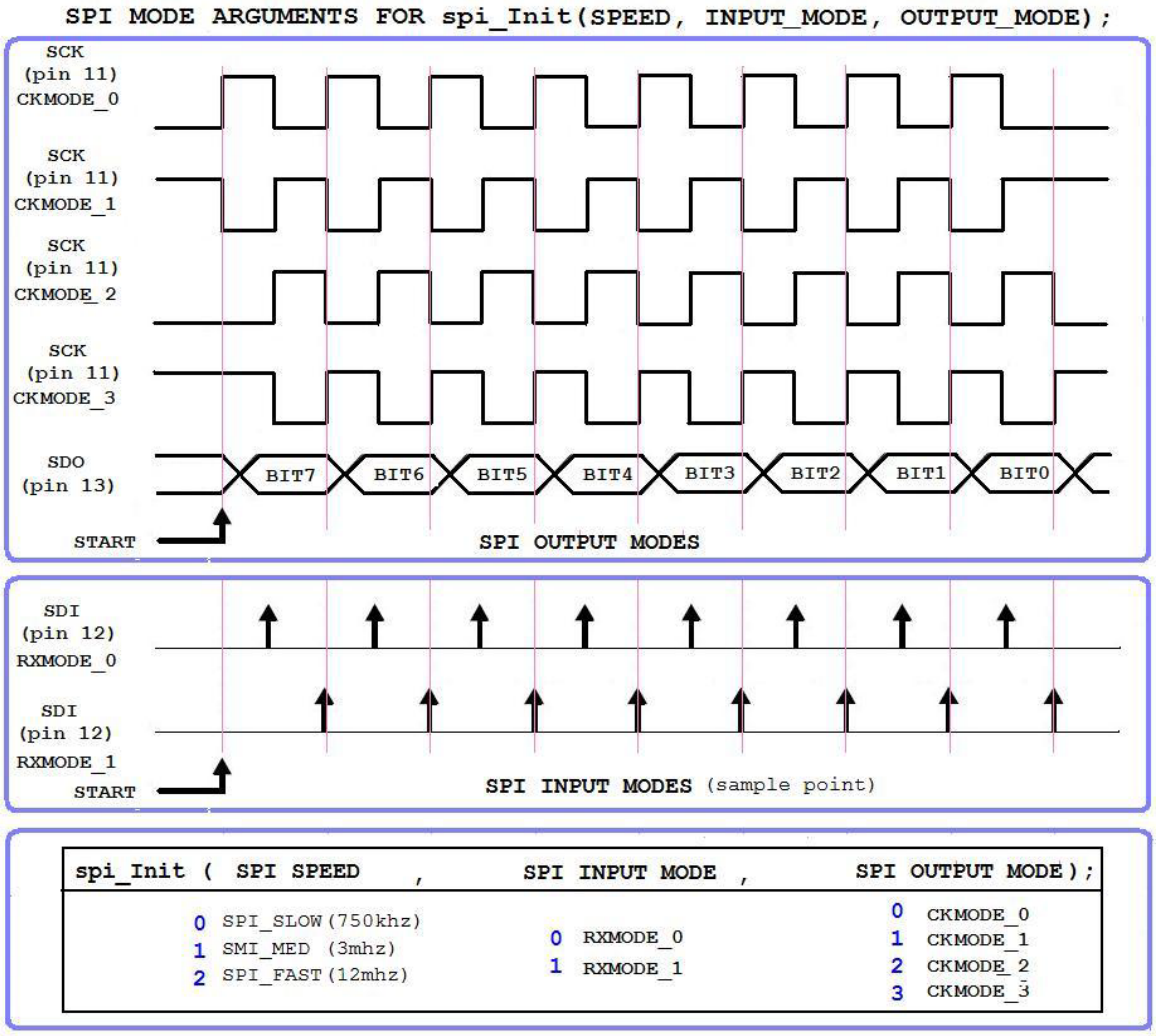
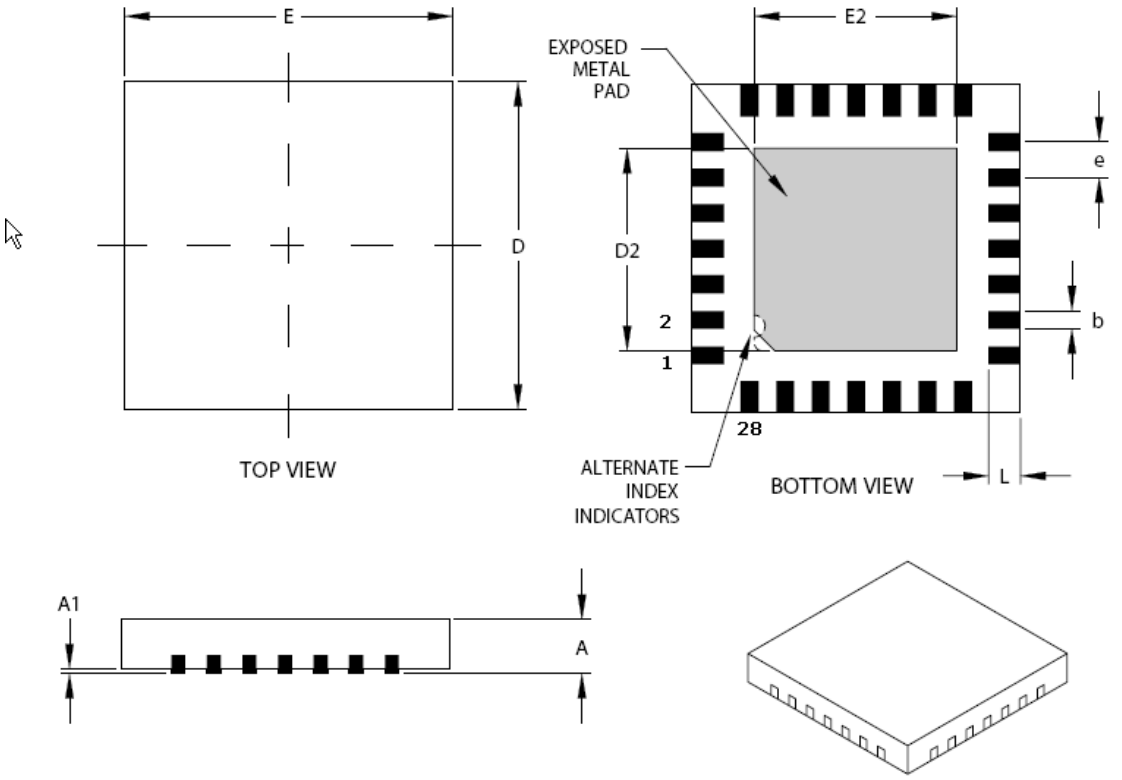



**Figure 8.1: SPI Timing Diagram and spi_Init() usage**

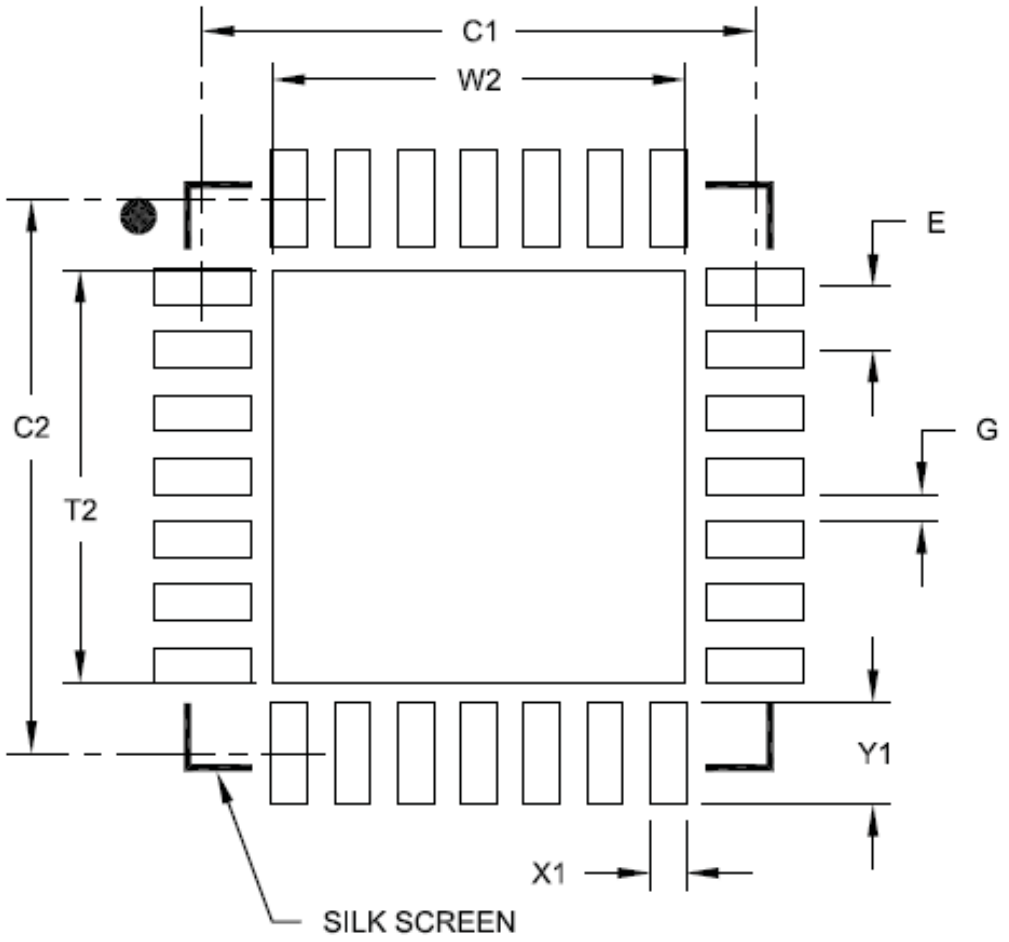
## 14. Package Details

**28 Pin QFN28 JEDEC MO-220**




| Units | | INCHES | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Pitch | e | | .026 BSC | | | 0.65 BSC | |
| Overall Height | A | .031 | .035 | .039 | 0.80 | 0.90 | 1.00 |
| Standoff | A1 | .000 | .001 | .002 | 0.00 | 0.02 | 0.05 |
| Contact Thickness | A3 | | .008 REF | | | 0.20 REF | |
| Overall Width | E | .232 | .236 | .240 | 5.90 | 6.00 | 6.10 |
| Exposed Pad Width | E2 | .140 | .146 | .152 | 3.55 | 3.70 | 3.85 |
| Overall Length | D | .232 | .236 | .240 | 5.90 | 6.00 | 6.10 |
| Exposed Pad Length | D2 | .140 | .146 | .152 | 3.55 | 3.70 | 3.85 |
| Contact Width | b | .009 | .011 | .013 | 0.23 | 0.28 | 0.33 |
| Contact Length | L | .018 | .022 | .024 | 0.45 | 0.55 | 0.65 |

## 15. PCB Land Pattern




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | 0.65 BSC | | |
| Optional Center Pad Width | W2 | | | 4.25 |
| Optional Center Pad Length | T2 | | | 4.25 |
| Contact Pad Spacing | C1 | | 5.70 | |
| Contact Pad Spacing | C2 | | 5.70 | |
| Contact Pad Width (X28) | X1 | | | 0.37 |
| Contact Pad Length (X28) | Y1 | | | 1.00 |
| Distance Between Pads | G | 0.20 | | |

## 16. Specifications and Ratings

| ABSOLUTE MAXIMUM RATINGS | |
|---|---|
| Operating ambient temperature | -40°C to +80°C |
| Storage temperature | -65°C +150°C |
| Voltage on any digital input pin with respect to GND | -0.3V to 6.0V |
| Voltage on SWITCH pin with respect to GND | -0.3V to 6.0V |
| Voltage on VCC with respect to GND | -0.3V to 4.0V |
| Maximum current out of GND pin | 300mA |
| Maximum current into VCC pin | 250mA |
| Maximum output current sunk/sourced by any pin | 4.0mA |
| Total power dissipation | 1.0W |

**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) | | 3.0 | 3.3 | 3.6 | V |
| Operating Temperature | | -40 | -- | +80 | °C |
| External Crystal (Xtal) | | -- | 12.00 | -- | MHz |
| Input Low Voltage (VIL) | VCC = 3.3V | VGND | -- | 0.8 | V |
| Input High Voltage (VIH) | VCC = 3.3V | 2.0 | -- | VCC | V |

| GLOBAL CHARACTERISTICS BASED ON OPERATING CONDITIONS | | | | | |
|---|---|---|---|---|---|
| **Parameter** | **Conditions** | **Min** | **Typ** | **Max** | **Units** |
| Supply Current (ICC) | VCC = 3.3V | -- | 12 | 26 | mA |
| Low Power Current(ICC) | VCC = 3.3V, Sleep Mode | 75 | 100 | -- | uA |
| Internal Operating Frequency | Xtal = 12.00MHz | -- | 48.00 | -- | MHz |
| Output Low Voltage (VOL) | VCC = 3.3V, IOL = 3.4mA | -- | -- | 0.4 | V |
| Output High Voltage (VOH) | VCC = 3.3V, IOL = -2.0mA | 2.4 | -- | -- | V |
| A/D Converter Resolution | IO1 pin | -- | 8 | -- | bits |
| Capacitive Loading | CLK1, CLK2 pins | -- | -- | 15 | pF |
| Capacitive Loading | All other pins | -- | -- | 50 | pF |
| Flash Memory Endurance | PmmC/4DGL Programming | -- | 1000 | -- | E/W |

| ORDERING INFORMATION |
|---|
| **Order Code:** GOLDELOX<br>**Package:** QFN28, 6mm x 6mm<br>**Packaging:** Tubes of 61 pieces |

## 17. Legal Notice

**Proprietary Information**

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

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

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

**Disclaimer of Warranties & Limitation of Liability**

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

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

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

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

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

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

## 18. Contact Information

For Technical Support: support@4dsystems.com.au

For Sales Support: sales@4dsystems.com.au

Website: www.4dsystems.com.au