APPLICATION NOTES

4D SYSTEMS

TURNING TECHNOLOGY INTO ART

Designer or ViSi Analog and Joystick Inputs

DOCUMENT DATE: 1st MAY 2020

DOCUMENT REVISION: 1.1

WWW.4DSYSTEMS.COM.AU

Description

This Application Note shows how to use the analog input of the Goldelox processor.

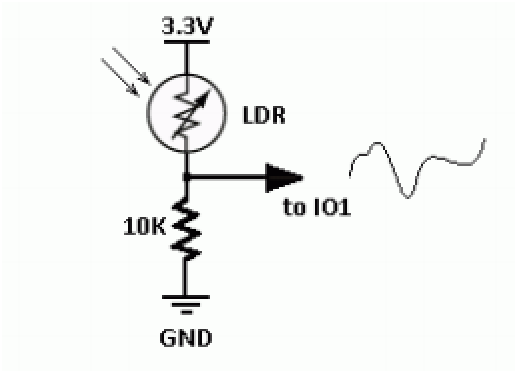
Before getting started, the following are required:

- Any of the following 4D Goldelox display modules:
  uOLED-96-G2
  uOLED-128-G2
  uOLED-160-G2
  uLCD-144-G2
  uTOLED-20-G2
  or any superseded module that supports the Designer environment
- 4D Programming Cable or µUSB-PA5
- Workshop 4 IDE (installed according to the installation document)
- micro-SD
- uCAM-II serial camera module
- A 10-kiloohm potentiometer
- Multi-switch joystick (see page 5 for the schematic diagram).
- When downloading an application note, a list of recommended application notes is shown. It is assumed that the user has read or has a working knowledge of the topics presented in these recommended application notes.
- This application note requires that the reader has a basic knowledge of any programming language such as C.

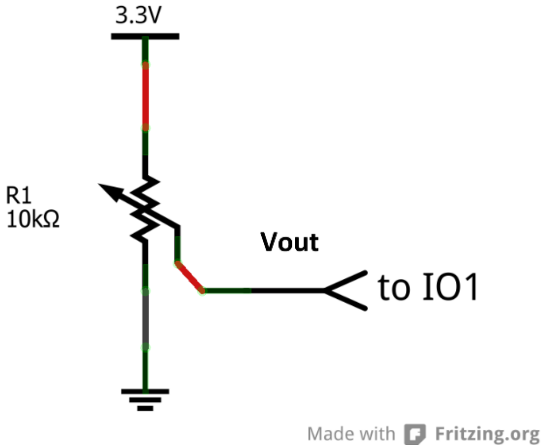
Content

Application Overview

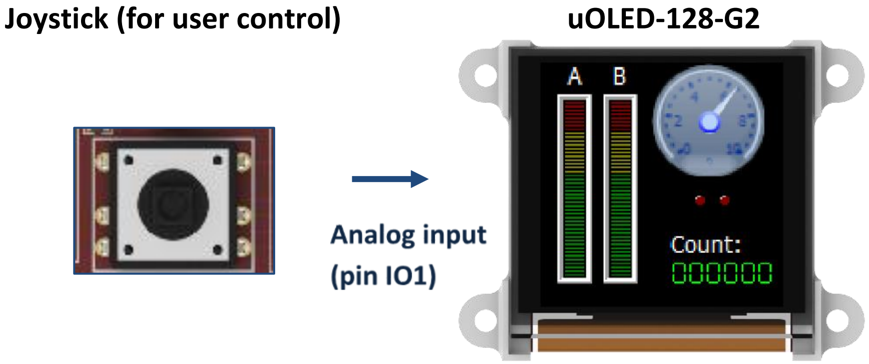
The IO1 pin of the Goldelox display can be programmed as an A/D input. Option is available to select 8 bit or 10 bit resolution. The voltage reference or the highest voltage that can be read by the IO1 pin is 3.3 volts. The diagram below is a circuit of a Light Dependent Resistor (LDR) connected to IO1 to measure and record changes in ambient light. This diagram illustrates one way to utilize the analog input of the Goldelox display.



The application described in this document makes use of a variable resistor as a voltage divider, the output of which is connected to the IO1 pin of a uOLED-128-G2.



Pin IO1 is then configured as an analog input and the readings are shown on the screen. The application also shows how the **joystick()** function is used, which is essentially an ADC process as well.



Setup Procedure

For instructions on how to launch Workshop 4, how to open a **Designer** project, and how to change the target display, kindly refer to the section "**Setup Procedure**" of the application note

[Designer Getting Started - First Project]

For instructions on how to launch Workshop 4, how to open a **ViSi** project, and how to change the target display, kindly refer to the section "**Setup Procedure**" of the application note

[ViSi Getting Started - First Project for Goldelox]

Create a New Project

For instructions on how to create a new **Designer** project, please refer to the section "**Create a New Project**" of the application note

[Designer Getting Started - First Project]

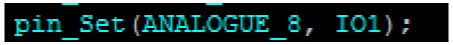
For instructions on how to create a new **ViSi** project, please refer to the section "**Create a New Project**" of the application note

[ViSi Getting Started - First Project for Goldelox]

Design the Project

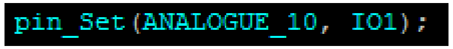
Configure the IO1 Pin for ADC

To configure the IO1 pin for ADC, user either one of the two following functions.



```
pin_Set(ANALOGUE_8, IO1);
```

Or

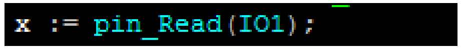


```
pin_Set(ANALOGUE_10, IO1);
```

The first line configures the IO1 pin for 8-bit resolution ADC, the second for 10-bit resolution ADC.

Read the Analog Value of Pin IO1

To read the analog value of pin IO1,

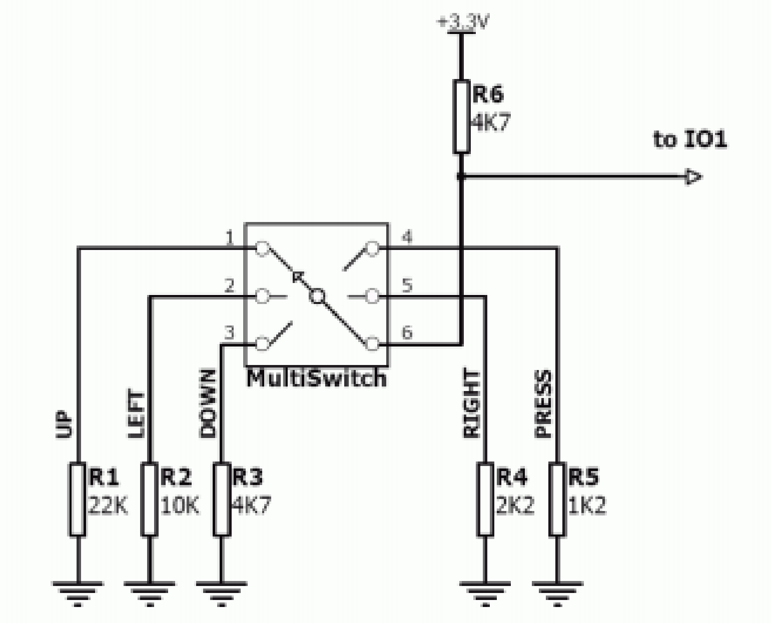


```
x := pin_Read(IO1);
```

For 8-bit resolution ADC, the pin_Read() function returns a value between 0 and 255. For 10-bit resolution, the pin_Read() function returns a value between 0 and 1023.

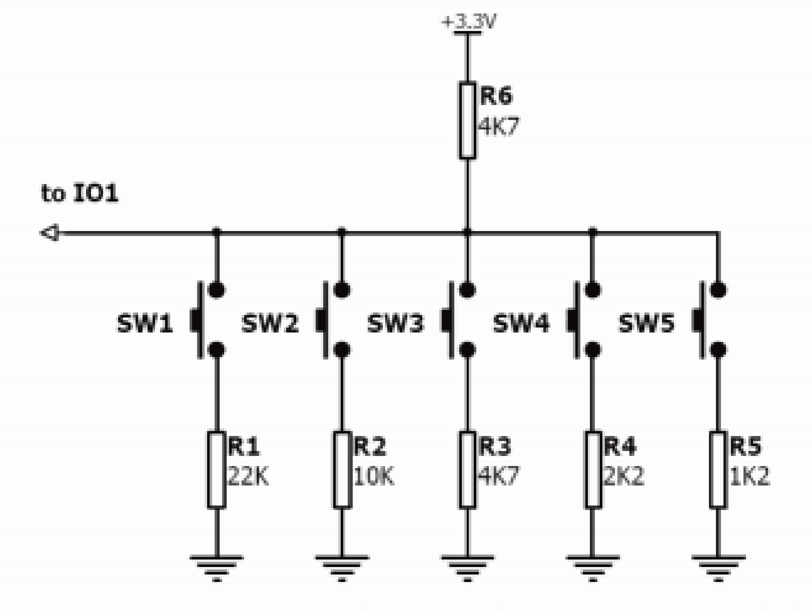
The Joystick

The Joystick is essentially a five-position multiswitch. Each position connects to a junction of a resistor ladder network that forms a voltage divider. The output of the joystick voltage divider connects directly to the IO1 pin of the Goldelox display module. Utilising the analog-to-digital-conversion feature of Goldelox displays, each individual switch position voltage value can be read and decoded. Below is the schematic diagram for the joystick.



ACTION	VOUT
UP	0.82*3.3V = 2.71V
LEFT	0.68*3.3V = 2.24V
DOWN	0.50*3.3V = 1.65V
RIGHT	0.32*3.3V = 1.05V
PRESS	0.20*3.3V = 0.66V
IDLE	= 3.30V

Note that the circuit can also be implemented using tactile switches and resistors.



The Joystick Function

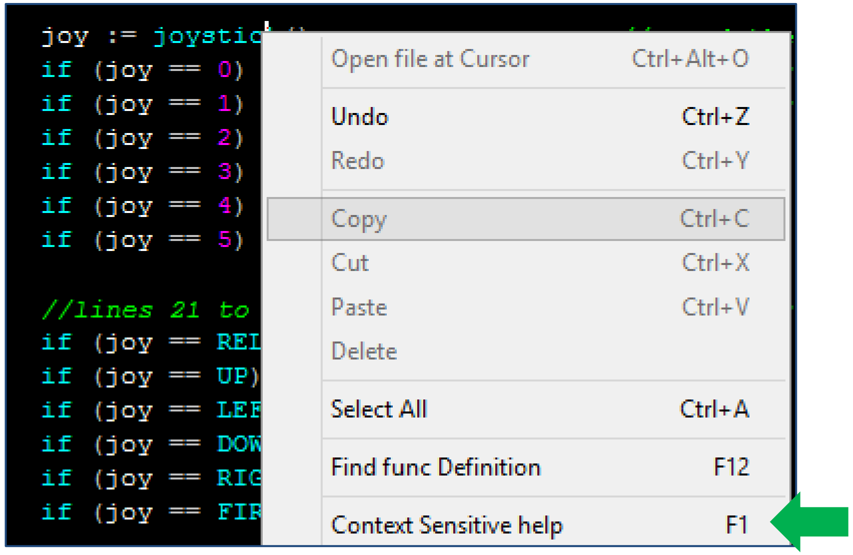
To use the joystick function, first declare a variable to hold an integer value.

```
var joy;
```

This variable will hold the return value of the function **joystick()**.

```
    joy := joystick();                  //
```

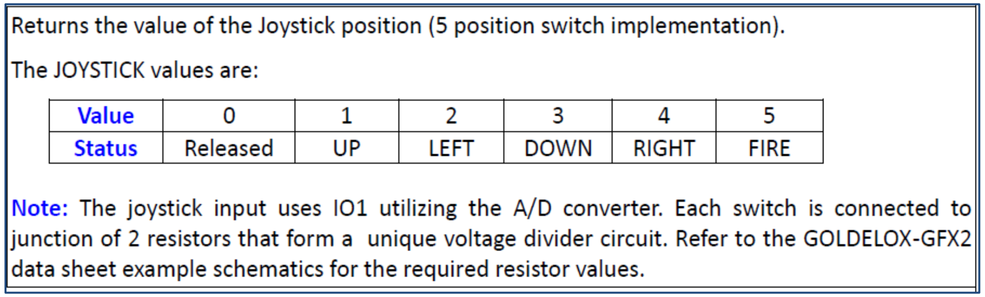
Put the cursor on the function name and either press F1 or right click on the mouse. For the latter, a pop-up menu will appear. Choose the last option, “Context Sensitive help”.



```
joy := joystic
if (joy == 0)
if (joy == 1)
if (joy == 2)
if (joy == 3)
if (joy == 4)
if (joy == 5)

//lines 21 to
if (joy == REL
if (joy == UP)
if (joy == LEF
if (joy == DOW
if (joy == RIG
if (joy == FIR
```

The Goldelox Internal Functions Reference Manual will open, if it has not been opened yet. Section 2.1.5 Joystick gives the following information.



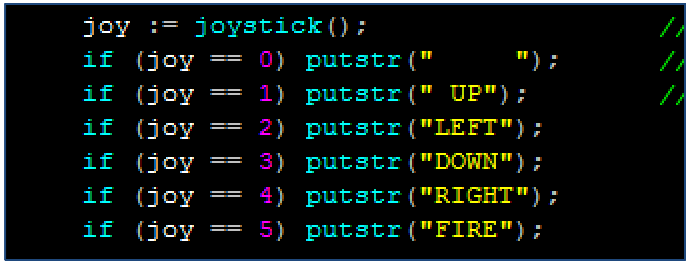
Returns the value of the Joystick position (5 position switch implementation).

The JOYSTICK values are:

Value	0	1	2	3	4	5
Status	Released	UP	LEFT	DOWN	RIGHT	FIRE

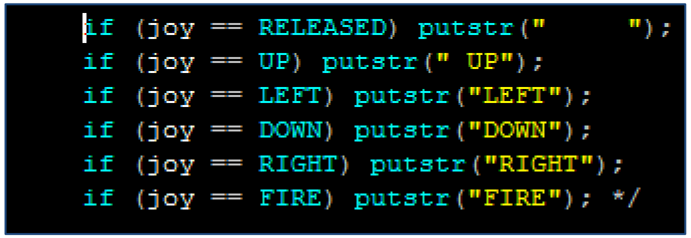
Note: The joystick input uses IO1 utilizing the A/D converter. Each switch is connected to junction of 2 resistors that form a unique voltage divider circuit. Refer to the GOLDELOX-GFX2 data sheet example schematics for the required resistor values.

IF statements can be used to evaluate the current status of the joystick.



```
    joy := joystick();                         //
    if (joy == 0) putstr("      ");            //
    if (joy == 1) putstr(" UP");               //
    if (joy == 2) putstr("LEFT");
    if (joy == 3) putstr("DOWN");
    if (joy == 4) putstr("RIGHT");
    if (joy == 5) putstr("FIRE");
```

The block can also be coded as:



```
    if (joy == RELEASED) putstr("      ");
    if (joy == UP) putstr(" UP");
    if (joy == LEFT) putstr("LEFT");
    if (joy == DOWN) putstr("DOWN");
    if (joy == RIGHT) putstr("RIGHT");
    if (joy == FIRE) putstr("FIRE"); */
```

Run the Program

For instructions on how to save a **Designer** project, how to connect the target display to the PC, how to select the program destination, and how to compile and download a program, please refer to the section “**Run the Program**” of the application note

Designer Getting Started - First Project

For instructions on how to save a **ViSi** project, how to connect the target display to the PC, how to select the program destination (this option is not available for Goldelox displays), and how to compile and download a program, please refer to the section “**Run the Program**” of the application note

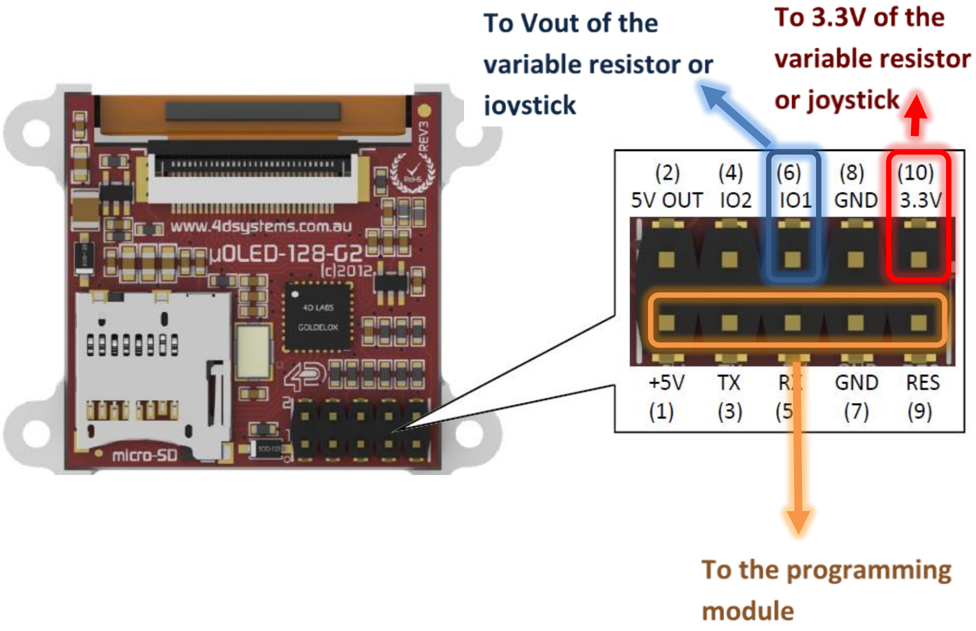
[**ViSi Getting Started - First Project for Goldelox**]

The uLCD-32PTU, uLCD-35DT, uOLED-96-G2, and/or uOLED-160-G2 display modules are commonly used as examples, but the procedure is the same for other displays.

Connection Diagrams

The programming module (4D USB programming cable or uUSB-PA5) can be used to power the display.

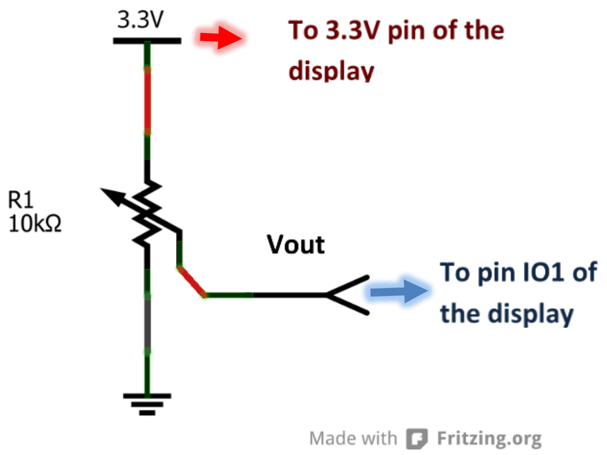
Pin Configuration of the Display



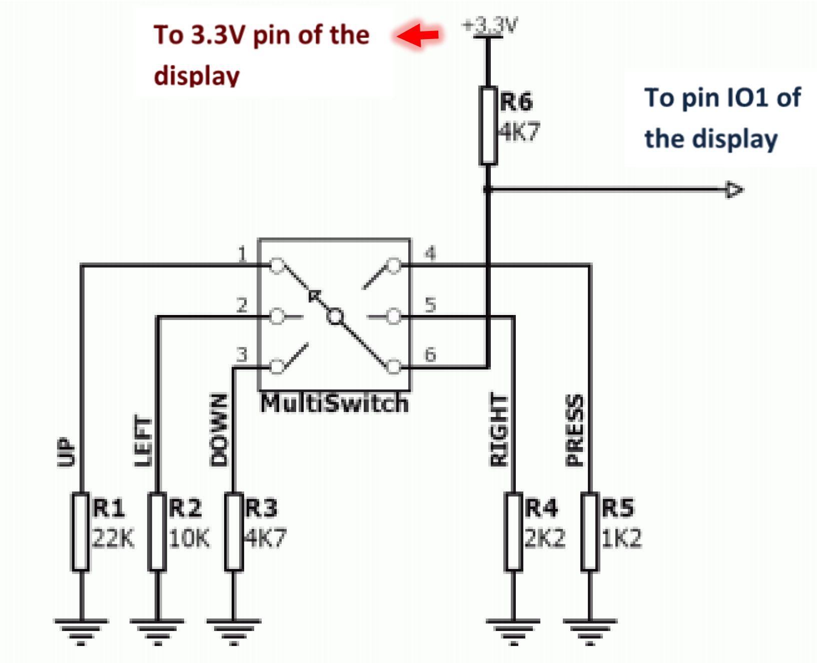
Don’t forget the common ground connections.

The Variable Resistor

The variable resistor is a 10-kiloohm potentiometer.

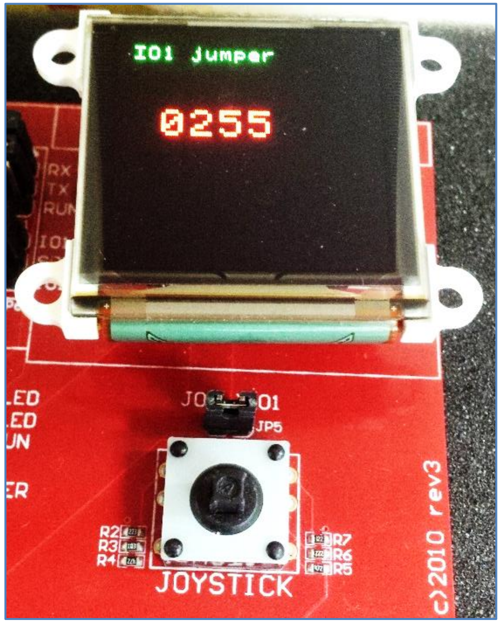


The Joystick

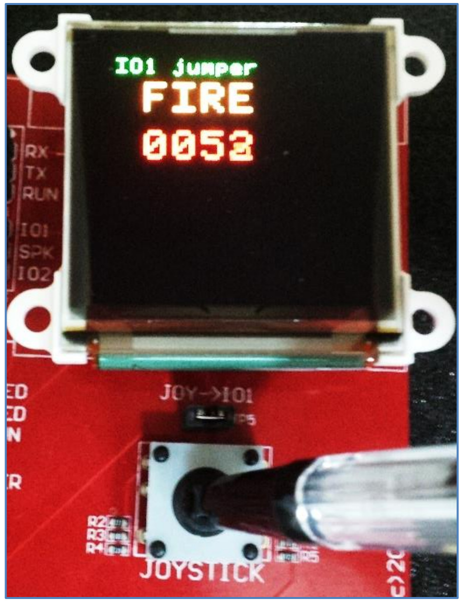


Photos of the Project

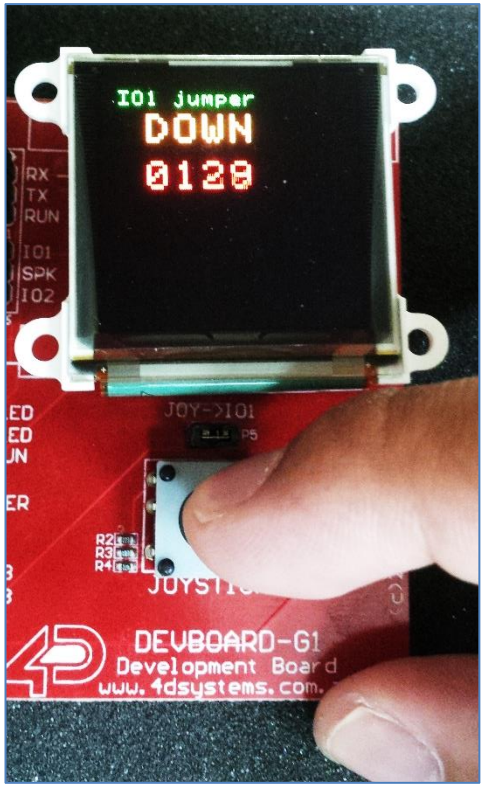
Here the display is placed on a development board which has a joystick. Note that aside from using the joystick function, the attached project also displays the analog readings of the IO1 pin. In the image below, the joystick is idle or in the "RELEASED" state. The analog reading is 255 (refer to the schematic for the joystick).



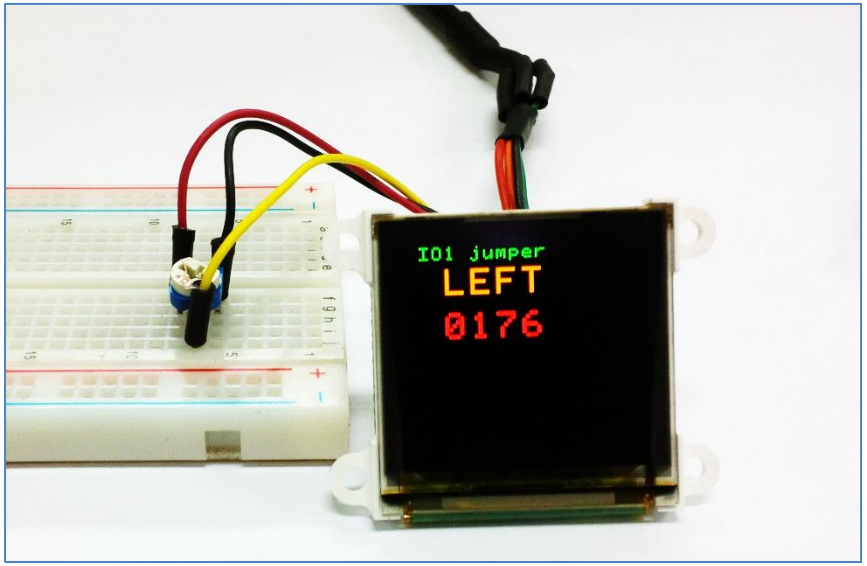
When the joystick is pressed, the state and analog reading are printed.

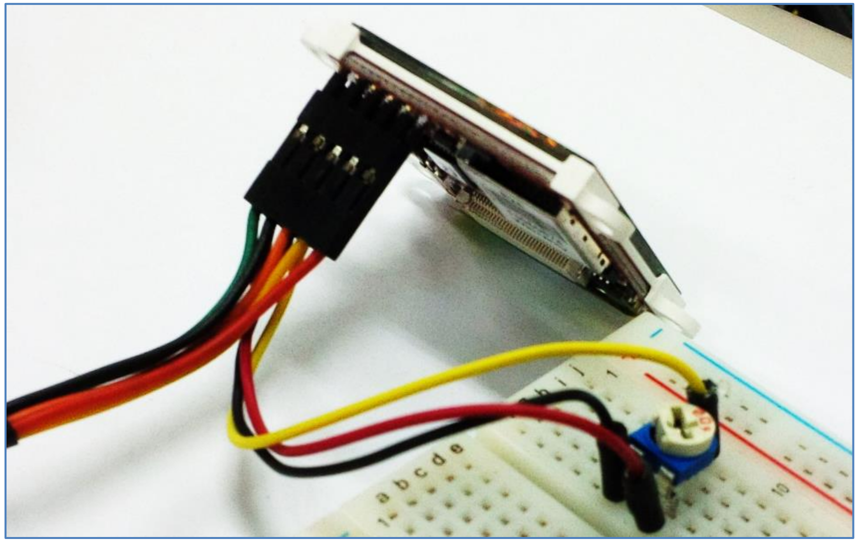


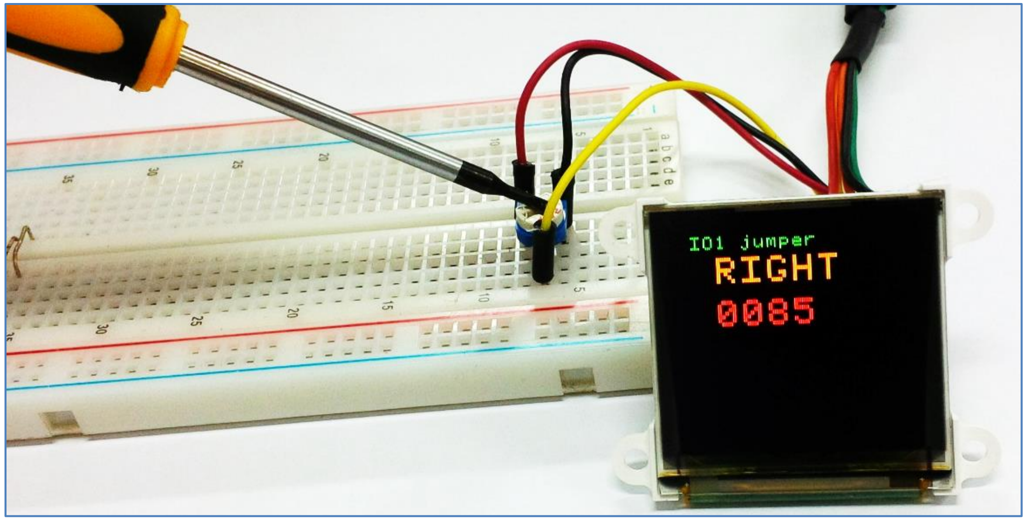
Below is the image when the state is"DOWN".



Below are images of a setup using a 10-kiloohm potentiometer.






IO1 jumper
RIGHT
0085

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.

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

Disclaimer of Warranties & Limitation of Liability

4D Systems makes no warranty, either expresses 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.

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.