OpenIO Labs - Client IOIs

Using the GP2 IOI

UG302 (v2017.1) April 3, 2017

Revision History

Date	Version	Revisions
3/4/2017	2017.1	*Initial release.*

Table of Contents

Chapter 1: Introduction

This User Guide describes the features and use of the OpenIO Labs GP2 IOI. The GP2 is an edge device within the OpenIO Labs system. The OpenIO Labs system comprises external sensor and control devices connected to an edge device such as the GP2 and a cloud server to which the GP2 is attached. The GP2 allows the user to create a range of control scripts that they can use to control the devices attached to the GP2, collect data from these devices, process the data and then push the data to the cloud server for presentation and storage. The connection between the GP2 and the cloud server uses the LWM2M protocol Ref [2] which is designed as a low latency, scalable protocol for Machine-to-Machine communications. A more detailed description of the OpenIO Labs system architecture can be found in Ref [1].

The OpenIO Labs GP2 is a very flexible edge processing device. With the GP2, its various interfaces and the use of ScriptML the user is able to connect a large number of devices and instruments to the GP2 for control, data collection and processing.

With the LWM2M interface to the OpenIO Labs cloud server, the user can create scripts via a web browser, download these scripts to the GP2 where they can control devices and sensors attached to the GP2 as well as collect data from these devices. This data can be further processed within the GP2 before pushing the data to the OpenIO Labs cloud server for further processing, presentation and storage.

This User Guide will go through the features of the GP2, provide information on the types of devices and instruments that can be attached to the GP2 and also provide additional information on how the user can program the GP2 for data collection, control and processing. First, we start with a review of the basic GP2 specifications.

GP2 Specifications

The specifications for the OpenIO Labs GP2 IOI are presented in the table below. The GP2 consists of a high specification embedded processor with a range of peripherals such as USB, I2C bus, GPIO and UART. The user has full access to these peripherals via the OpenIO Labs ScriptML programming system, in which the users can create applications using languages of their choice such as in C or in Python.

Feature	Description
CPU	1.2 GHz Quad core ARM Cortex-A53 processor
RAM	1 GByte SDRAM
USB	4 Port USB 2.0 supporting maximum transfer rates of 480 Mbit/s
Ethernet	10/100 Mbit/s via standard RJ-45 connector
WiFi	IEEE 802.11n support
I2C	Support for I2C bus with a number of devices allowed on the same bus
SPI	Support for the Serial Peripheral Interface bus
GPIO	16 General Purpose IO connections
UART	Support for RS232-like serial interface protocol
LWM2M	Support for the LWM2M protocol from the GP2 to the cloud server
ScriptML	Support for the OpenIO Labs ScriptML protocol for user defined applications

Table 1: Summary of GP2 Features

GP2 Interfaces and Connections

The GP2 front panel is illustrated in the figure below. The front panel includes:

- Micro-USB power connection and LED power indication
- 10/100 Mbps RJ45 Ethernet interfaces
- 4 x USB 2.0 type A connectors
- 40 Pin IDC IO interface connector.

The power connector is a standard micro-USB interface connector providing power to the OpenIO Labs GP2 IOI. The power LED is illuminated when power is applied. The RJ-45 Ethernet provides a wired access for the GP2 to external IP networks. The quad USB interface allows for the connection of a range of USB devices such as external test equipment like oscilloscopes and spectrum analysers (discussed in more detail in a later section).

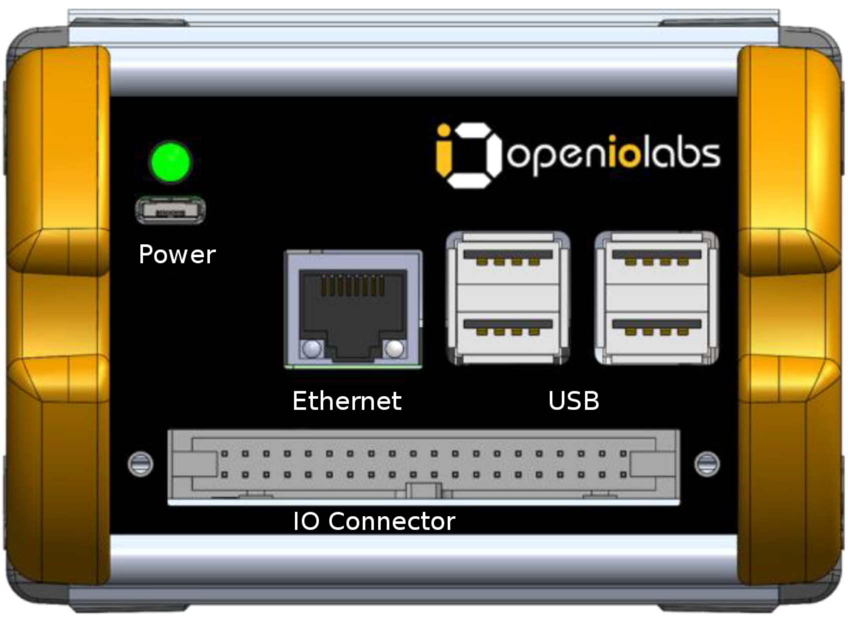


Figure 1: GP2 Front Panel Layout

GP2 IO Connector

The pin layout for the GP2 IO connector is presented in the figure below, and in more detail in the table below that. The connector is a standard 40 pin IDC connector, with pin 1 at the lower left corner and pin 40 at the upper right corner. The bottom row are the odd numbered pins in sequence, and the top row the even numbered pins in sequence.

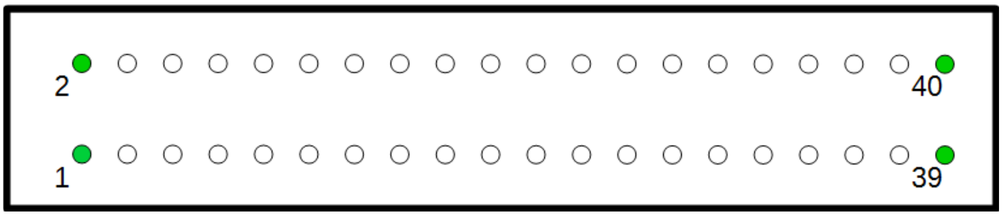


Figure 2: GP2 IO Connector Interface

Pin	Use
1	3V3 - Power
2	5V - Power
3	I2C - SDA
4	5V - Power
5	I2C - SCL
6	Ground
7	GPIO4
8	UART - TXD
9	Ground
10	UART - RXD
11	GPIO17
12	GPIO18
13	GPIO27
14	Ground
15	GPIO22
16	GPIO23

17	3V3 - Power
18	GPIO24
19	SPI - MOSI
20	Ground
21	SPI - MISO
22	GPIO25
23	SPI - SCLK
24	SPI - CE0
25	|Ground
26	SPI - CE1
27	Reserved
28	Reserved
29	GPIO5
30	Ground
31	GPIO6
32	GPIO12
33	GPIO13
34	Ground
35	GPIO19
36	GPIO16
37	GPIO26
38	GPIO20
39	WiFi - Config1
40	WiFi - Config2

Table 1: Pin Connections for the IO Connector

The GP2 IO connector supports a range of features including variable output power (+3.3V and +5V), a selection of GPIO interface pins, each individually controllable, I2C bus functions, SPI bus functions, UART serial interface connections and two pins that are used to configure the WiFi interface from the GP2 to a WiFi enabled router.

Initial WiFi Configuration

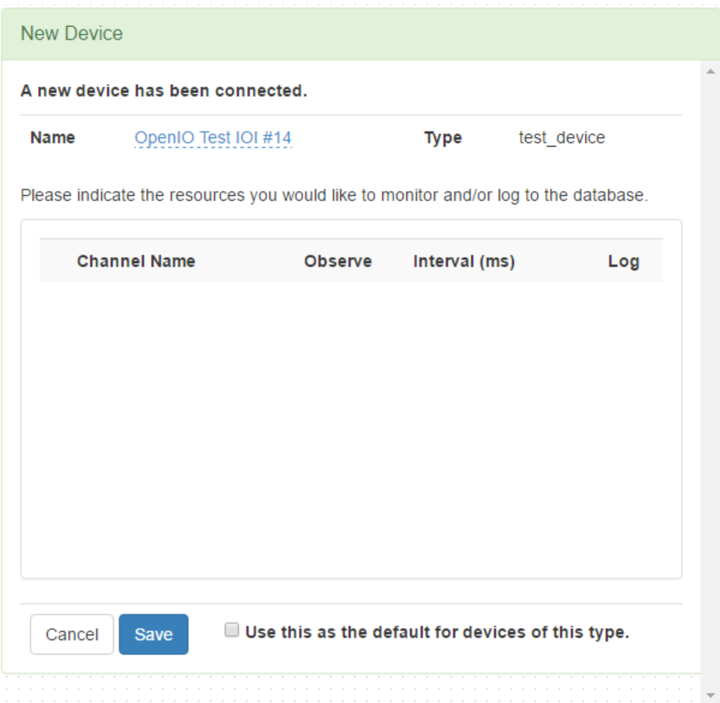
On first installation, or when the GP2 has been moved to a new location, the WiFi connection will need to be configured. To do this, follow the following procedure:

1. With the device powered-off, insert the link provided with the GP2 between pins 39 and 40 of the IO connector.
2. Power on the GP2
3. With a smart phone, tablet or PC, search for a WiFi network called: "**TBBBD**"
4. Connect to this WiFi network
5. On the web page that is displayed, enter the SSID of the network that you would like the GP2 to connect to in the box called **SSID**
6. If a password is required for the WiFi network, enter the password in the box called **Password**.
7. Power-off the GP2 by removing the USB power chord
8. Remove the link between pins 39 and 40
9. Power on the GP2 and it will now search and connect to the selected WiFi network.

Finding the GP2 on the Cloud Server

Next log in to the OpenIO Labs Cloud server using the e-mail and password provided by OpenIO Labs. For further details on logging in and access the OpenIO Labs system, refer to the UG102 Ref [3].

When you login you will see a pop-up window similar to the one shown in the figure below.



This indicates that the device has been registered with the OpenIO Labs network and is now ready to be used.

Using the GP2

Now that you have the GP2 powered on and connected to the OpenIO Labs network, there are a number of things that you can do. In the following sections we will consider some of the ways that you can use the OpenIO Labs network to create and control the experimental systems using the GP2 and the OpenIO Labs system.

We will consider three main topics in this and the following sections:

1. Connecting sensor and control devices via the I2C bus
2. Connecting simple edge devices via the GPIO pins
3. Connection of USB Test and Measurement equipment to the GP2

With these three sets of features, the GP2 allows the user to create a diverse set of experimental configurations. The range and types of sensors and controller that

I2C Devices and I2C Bus Interface

The I2C bus is a simple serial interface that is used by many different sensor and control devices. Further details of the I2C bus can be found in the link in Ref [8]. The types of devices range from simple temperature and pressure sensors, to multi-axis accelerometers and magnetometers. The GP2 provides access to the I2C bus using The IO connector on the front panel. From Table 1 above, we can see that the I2C data line is on Pin-3 and the I2C clock is on Pin-5 of the IO Connector. In addition, we need to supply power for the devices on the bus, so either a 3.3V or 5V pin can be selected (depending on the device) and the Ground connection also.

If we consider an example device such as the MAX518 two channel DAC (Digital to Analogue Converter) from Maxim, in the data-sheet, we will see pin connections for the device similar to those shown in the figure below.

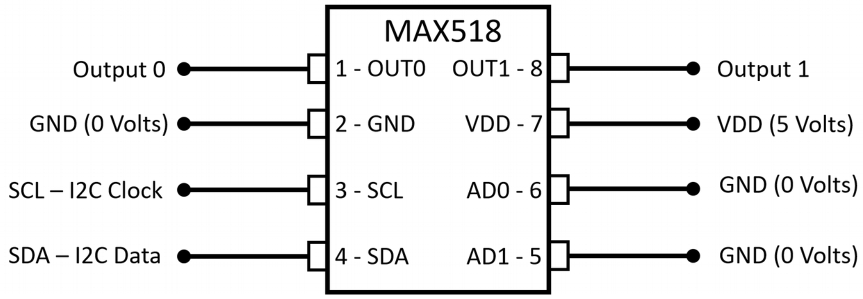


Figure 3: MAX518 Dual Channel DAC - Wiring Connections

The device has 8 pins that are a combination of address pins, outputs, power and I2C signals. The connections for this device to the GP2 IOI are presented in the table below.

MAX518 Pin	IO Connector	Description
1	NA	Analogue output 0 pin
2	Pin-6	Ground connection
3	Pin-5	I2C serial clock
4	Pin-3	I2C data signal
5	NA	MAX518 I2C address Pin 1 – set to 0 V
6	NA	MAX518 I2C address Pin 0 – set to 0 V
7	Pin-2	VDD power supply, set to 5V
8	NA	Analogue output 1 pin

Table 2: MAX518 to GP2 IO Connector Pin Connections

With the device connected to the GP2 IO Connector, we are now ready to start to program and control the outputs of the DAC.

Programming the MAX518 with ScriptML

To program the MAX518 we will use ScriptML. ScriptML is a programming system that has been developed by OpenIO Labs to allow users to create applications that run on the edge device (GP2) using a language of their choice (C, Python and Java as some examples).

An introduction to ScriptML is presented in Ref [4], some example code for programming the MAX518 is presented in Ref [5] and a general introduction to writing I2C drivers using ScriptML is presented in Ref [6].

Following the examples presented in Ref [5] the user can create an application that controls the outputs from the MAX518 DAC. The script that is created is loaded into the GP2 edge device using a browser that is connected to the OpenIO Labs cloud server.

In addition to the DAC, Ref [5] also presents the mechanisms to connect an ADC to the GP2 using the I2C bus. In the example that is presented, the user samples the outputs of the DAC using the ADC. The ScriptML application then collects these samples from the ADC and pushes them to the server, where they are plotted and saved onto the cloud server database.

Following this simple example, the user can then build on these simple applications to create complex experimental systems that allow the user to control edge devices and collect various types of data from them, and then push the results to the OpenIO Labs cloud server.

GPIO Interface

In addition to the I2C bus, the user can also access the GPIO pins that are available on the IO connector of the GP2 IOI. The GPIO pins are General Purpose Input Output pins, that can be used to set the pin to digital logic high and low voltage levels.

The use of the GPIO pins from the GP2 IO Connector is similar to the use of the I2C pins. The user will first select a GPIO pin that they want to use from available IO pins. So, as an example, the user wants to connect an LED to GPIO4 that is on Pin-7 of the GP2 IO connector. An example schematic for this is shown in the figure below, were the resistor value should be in the range of 200 Ohms to 400 Ohms (depending on the brightness required).

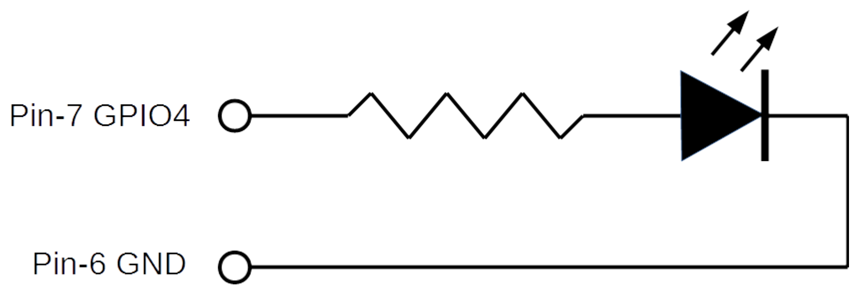


Figure 4: Pin Configuration for an LED attached to GPIO4

With the LED connected to the GPIO4 pin via a resistor, whenever GPIO4 is set to a logical one, the LED will illuminate.

Programming the GPIO Pins with ScriptML

The methods to create applications using ScriptML to drive the GPIO pins is presented in Ref [9]. In that User Guide the user is shown the various ways that device drivers can be created and used to control the GPIO pins that are available from the GP2 IO connector.

Interfacing to External Test Equipment

In addition to configuring and controlling devices via the I2C interface and using the GPIO pins, the OpenIO Labs GP2 can also interface directly with external test equipment that supports a USB programming interface and the IEEE 488.2 software programming model, and in particular the SCPI command sets.

The IEEE 488.2 specification is an enhancement to the IEEE 488.1 interface protocol that is generically referred to as GPIB and originally HPIB. The details for IEEE 488.2 can be found in Ref [10]. IEEE 488.2 extended the original IEEE 488 specification to also include software specifications for the control of various instrument types. This specification was extended in 1990 by the addition of the Standard Commands for Programmable Instruments (SCPI) which defined a consistent set of commands that can be used for programming a range of lab test equipment such as oscilloscopes, spectrum analysers and digital multi-meters.

The support for SCPI within the OpenIO Labs system is provided by the USBTMC library. This library is integrated into the GP2 IOI and allows control of compliant instruments via ScriptML. More details of the USBTMC measurement class can be found in Ref [12].

An example of controlling a test instrument such as an oscilloscope is presented in the User Guide UG405 – Ref [7].

Additional Resources

References

The following documents and references are cited within this guide:

1. UG101 – OpenIO Labs System Architecture
2. LWM2M Technical Specification – OMA-TS-LightweightM2M-V1_0-20141126-C
3. UG102 – Accessing the OpenIO Labs System
4. UG401 – ScriptML Overview
5. UG402 – ScriptML Examples in C and Python
6. UG403 – Writing I2C Device Drivers in C and Python
7. UG405 – Controlling USB Test Equipment with ScriptML
8. https://en.wikipedia.org/wiki/I2C
9. UG406 – Writing GPIO Device Drivers in C and Python
10. https://en.wikipedia.org/wiki/IEEE-488
11. https://en.wikipedia.org/wiki/Standard_Commands_for_Programmable_Instruments
12. http://www.usb.org/developers/docs/devclass_docs/USBTMC_1_006a.zip