

SparkFun QwiicBus Hookup Guide

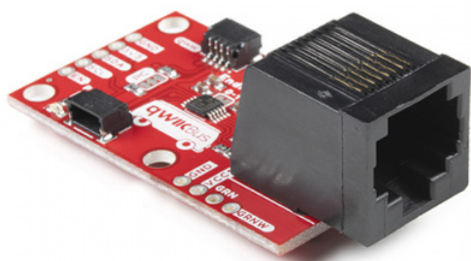
Introduction

Introducing the SparkFun QwiicBus system! The QwiicBus is a fast and easy way to extend the range of your I²C bus. The QwiicBus system features two boards: the SparkFun QwiicBus EndPoint and the SparkFun QwiicBus MidPoint. SparkFun also offers the QwiicBus Kit that includes two EndPoints, one MidPoint and two Ethernet cables to get you started with the QwiicBus.

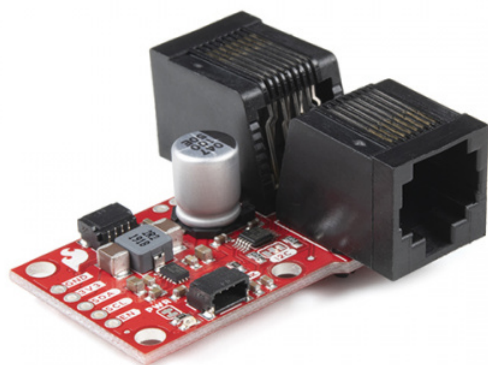


SparkFun QwiicBus Kit

* KIT-17250



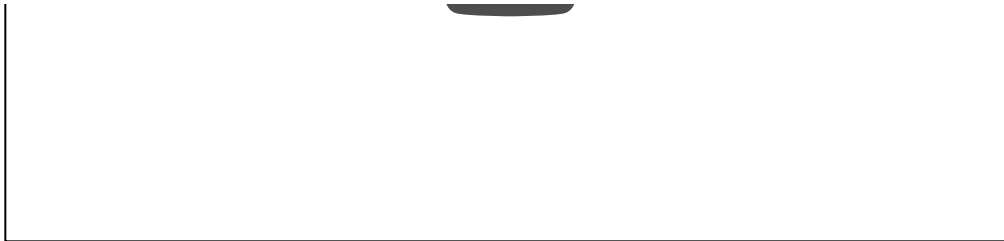
SparkFun QwiicBus - EndPoint
© COM-16988



SparkFun QwiicBus - MidPoint
© COM-18000

Product Showcase: SparkFun QwiicBus MidPoint & ...





Using NXP's PCA9615 differential I²C bus buffer IC, the QwiicBus converts the two default I²C signals into four differential signals (two for SCL and two for SDA). The differential signals are sent over an Ethernet cable, which attaches to the EndPoint or MidPoint through the on-board RJ-45 connectors. Differential signaling allows the I²C signals to reach distances of over 100 feet while still maintaining their signal integrity. In our testing using two EndPoints, four MidPoints, at least one Qwiic device on each node and over 200 feet of Ethernet cable, we were able to use all devices with nearly no signal integrity loss!

The EndPoint acts as the starting and ending points of the QwiicBus and the MidPoint allows you to add a drop-in I²C connection to your long-distance differential I²C chain wherever you would like.

These boards grew out of a collaboration with FarmHand Automation. While developing autonomous microtractors to help small farmers grow their business, Farmhand realized they needed a low cost, open source CAN/Modbus alternative. Read more about their story here. FarmHand founder, Alex Jones said, "We knew the sensors we wanted to use, but as soon as you need to communicate over long distances in noisy environments things get complicated. The Qwiic Midpoint is going to solve a lot of that headache." The QwiicBus MidPoint and QwiicBus EndPoint are ideal for applications that require long-range communication over Ethernet, such as agricultural technology or data collection/monitoring in rural/remote areas.

Whether you have a robot with multiple I²C devices throughout it like FarmHand, a sensor network project with multiple sensors over a large area or other I²C projects you can think of that require a wired signal transmission over long distances, the QwiicBus makes that communication a breeze!

Required Materials

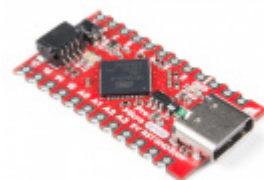
If you are using the QwiicBus Kit, you'll have the required boards and Ethernet cables to assemble your QwiicBus circuit. Otherwise, you'll want to pick up two EndPoints and however many MidPoints your project needs.

Along with the QwiicBus boards, you will need the following materials to follow along with this tutorial. You may not need everything though depending on what you have. Add it to your cart, read through the guide, and adjust the cart as necessary.

A microcontroller is needed to control any I²C devices attached to your QwiicBus. Below are a few options that come Qwiic-enabled out of the box:



SparkFun Thing Plus - ESP32 WROOM



SparkFun Qwiic Pro Micro - USB-C (ATmega32U4)

○ WRL-15663

● DEV-15795



SparkFun RedBoard Qwiic

● DEV-15123

SparkFun RedBoard Artemis

● DEV-15444

If you would prefer to use a single-board computer (SBC) like a Raspberry Pi or Jetson Nano as your controller, the products below will also work with the QwiicBus:



NVIDIA Jetson Nano Developer Kit (V3)

● DEV-16271

SparkFun DLI Kit for Jetson Nano

● KIT-16308



Raspberry Pi 4 Model B (2 GB)

○ DEV-15446

SparkFun Raspberry Pi 4 Desktop Kit - 4GB

○ KIT-16386

If your favorite microcontroller or single board computer is not already Qwiic-enabled, you can add that functionality with one or more of the following items:



SparkFun Qwiic Adapter

○ DEV-14495



SparkFun Qwiic SHIM for Raspberry Pi

● DEV-15794



SparkFun Qwiic Shield for Arduino

● DEV-14352



SparkFun Qwiic pHAT v2.0 for Raspberry Pi

● DEV-15945

You will also probably want a few Qwiic cables to connect your devices on your I²C bus:



Qwiic Cable - 100mm

● PRT-14427



Qwiic Cable - 500mm

● PRT-14429



Qwiic Cable - 50mm

● PRT-14426



Qwiic Cable - 200mm

● PRT-14428

Finally, if you are not using the QwiicBus Kit you'll need at least one straight through Ethernet cable.

Optional Extras for Alternate QwiicBus Power Configurations

The QwiicBus offers multiple power configurations intended for applications where many devices need to be powered over the QwiicBus. We cover these configurations in more detail in the [Hardware Overview](#) and [Hardware Assembly](#). To use these configurations you will need some extra hardware and tools. Click the button below to view some recommended products for alternate power configurations.

ADDITIONAL HARDWARE & TOOLS FOR ALTERNATE QWIICBUS POWER CONFIGURATIONS

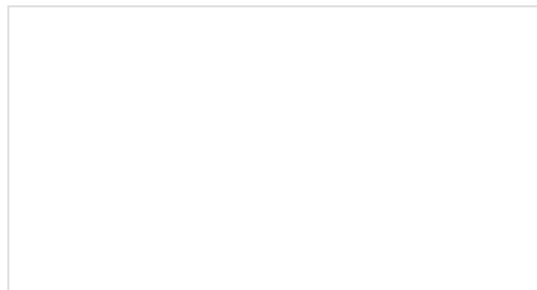
Suggested Reading

If you aren't familiar with the Qwiic system, we recommend reading here for an overview.



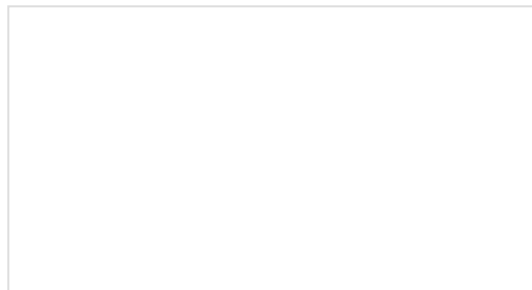
Qwiic Connect System

We would also recommend taking a look at the following tutorials if you aren't familiar with the topics they cover.



Logic Levels

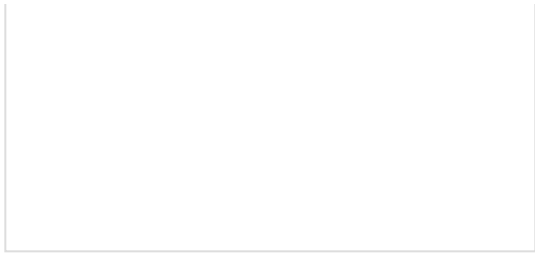
Learn the difference between 3.3V and 5V devices and logic levels.



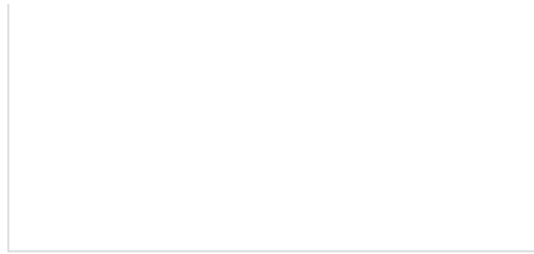
Electric Power

An overview of electric power, the rate of energy transfer. We'll talk definition of power, watts, equations, and power ratings. 1.21 gigawatts of tutorial fun!





I2C
 An introduction to I2C, one of the main embedded communications protocols in use today.



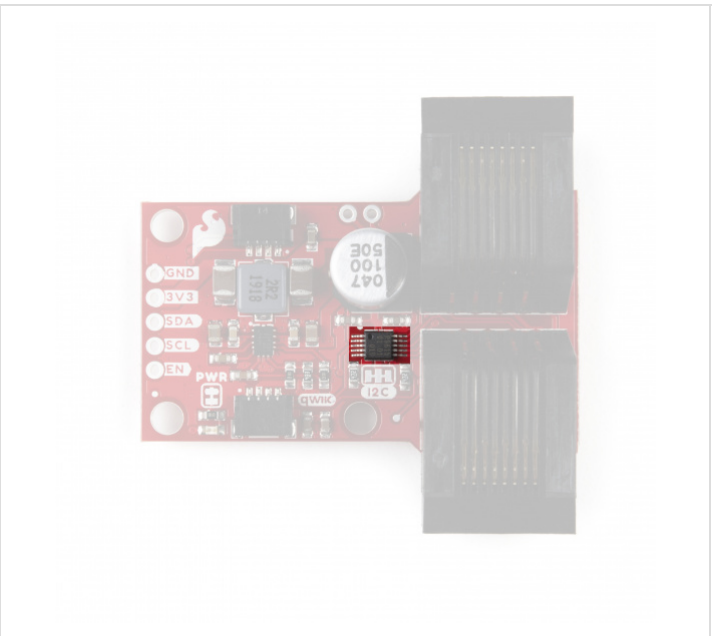
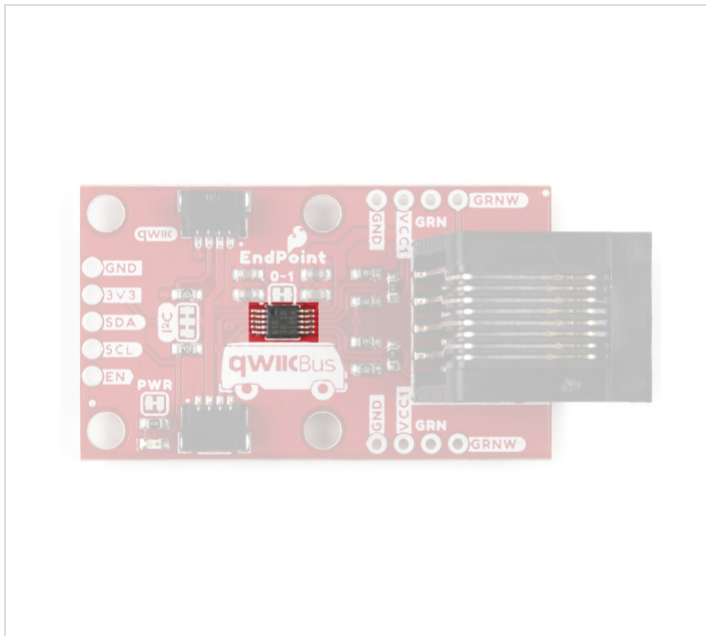
How to Work with Jumper Pads and PCB Traces
 Handling PCB jumper pads and traces is an essential skill. Learn how to cut a PCB trace, add a solder jumper between pads to reroute connections, and repair a trace with the green wire method if a trace is damaged.

Hardware Overview

The simplicity of the QwiicBus is one of its biggest appeals as all you really need to get started is the QwiicBus boards, Ethernet and Qwiic cables and a controller (either development board or SBC). Other I²C communication methods require packetizing I²C communication into another protocol, be it RS-485 or 1-Wire. However, the PCA9615 keeps the I²C protocol by utilizing a differential transceiver. In this section, we'll take a closer look at the QwiicBus boards and the hardware present on them to better understand how they work.

PCA9615 Bus Buffer IC

Let's take a quick look at the PCA9615 IC at the heart of the MidPoint and EndPoint. The PCA9615 acts as a bridge that translates a standard two-wire I²C bus to a four-wire differential I²C bus. Translating to a differential bus helps prevent signal disruption in noisy environments as well as extending the signal over long-distance transmissions. For detailed information about the full functionality and feature set of the PCA9615 review the datasheet.



The PCA9615 has two supply voltage rails: VDDA and VDDB. VDDA acts primarily as the I²C-bus side power supply and VDDB primarily is used for the differential side power supply. While the two power supplies have slightly different operating ranges (VDDA supply voltage range: **2.3-5.5V**. VDDB supply voltage range: **3.0-5.5V**), both the EndPoint and MidPoint default to net both voltages together at **3.3V**.

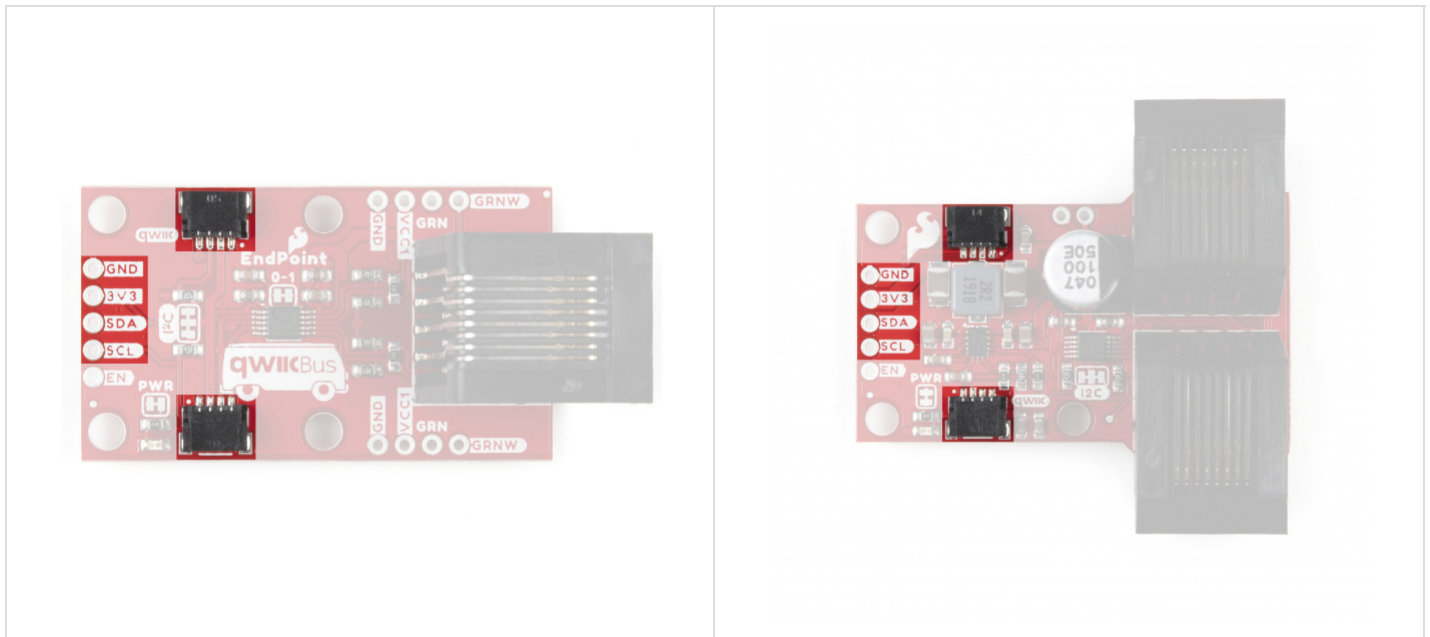
The PCA9615 supports I²C clock speeds up to 1MHz though the maximum cable length is inversely related to the clock speed. At max speed, the max rated cable length is listed at 3m but can be increased at lower clock speeds. For example, in our testing of the QwiicBus with over 200 feet of Ethernet cable we observed no significant signal loss while running at the standard clock speed for the Arduino Wire Library (100KHz).

We've designed the QwiicBus MidPoint and EndPoint to have multiple configuration options for powering your QwiicBus so there is a bit to take note of before powering everything up. We cover the different power configurations further down in the Solder Jumpers sub section as well as in the Hardware Assembly section so read on for more information.

The QwiicBus boards also break out the Enable (EN) pin to a PTH header if you would like to control the PCA9615 manually. By default, it is pulled to VDDA via an internal resistor. Refer to section 7.3 of the PCA9615 datasheet for more information on using this pin.

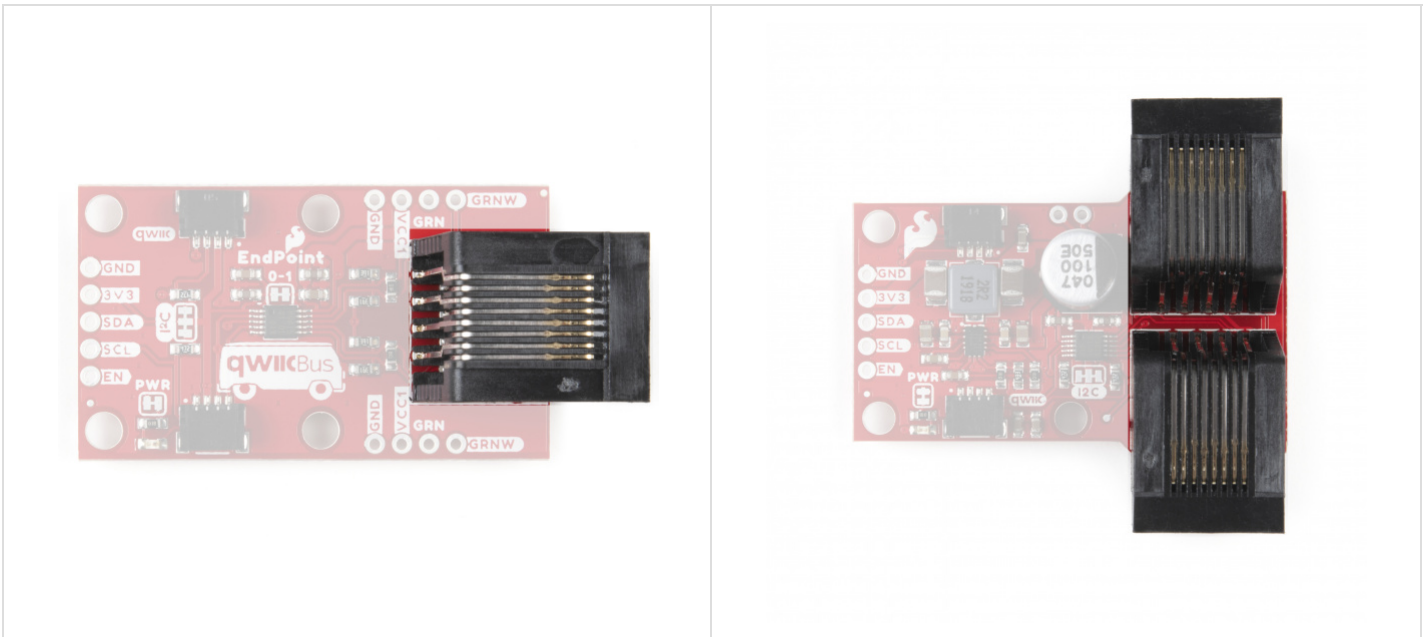
Qwiic and I²C Interface

As a large portion of you readers have come to expect, the I²C pins are broken out to two Qwiic connectors so you can easily connect your QwiicBus boards to other Qwiic devices. We've also broken out the four I²C pins to standard 0.1"-spaced PTH header pins for users who prefer the standard plated through-hole connection.



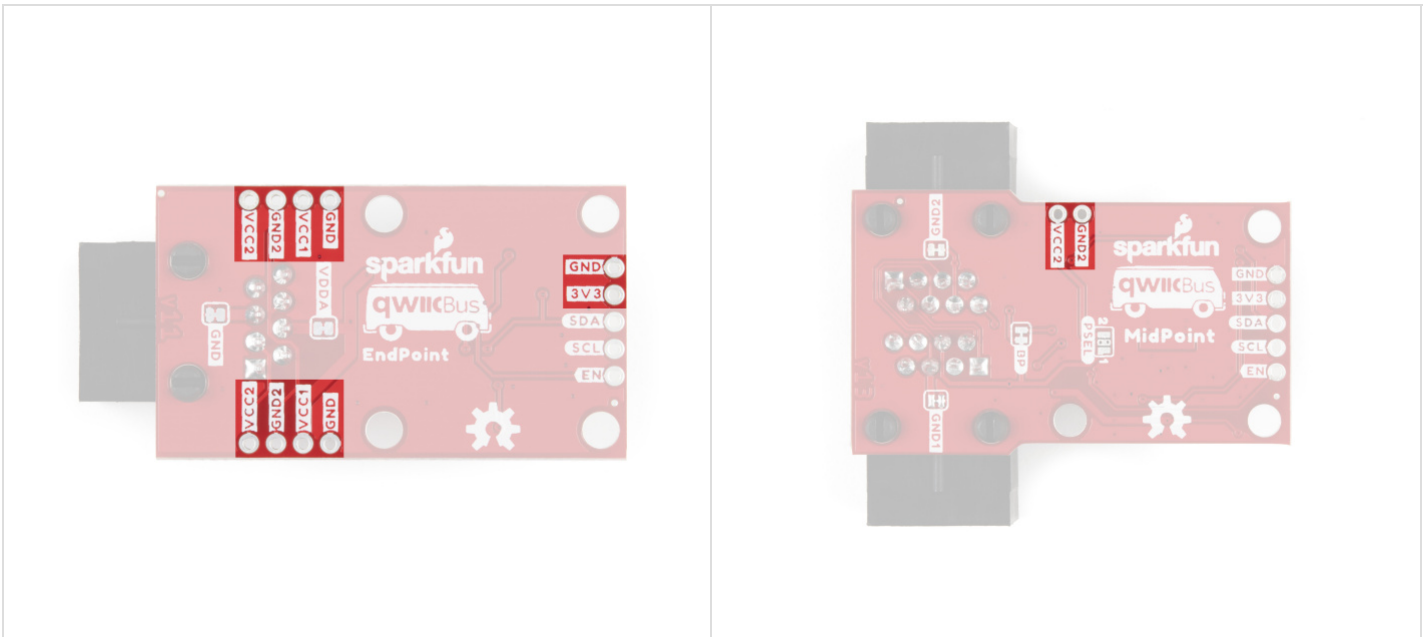
Differential Output and RJ-45 Connectors

The EndPoint has a single RJ-45 to send the differential signal out to another EndPoint or to your MidPoint nodes. The MidPoint comes with two RJ-45 connectors so you can easily integrate it into your existing bus.



Depending on your project's power needs, the unused Green and Blue twisted pairs in the Ethernet cable can be used to send **5V** over the Blue pair (Ethernet pins 4 and 5) or **3.8V to 36V** over the Green pair (Ethernet pins 3 and 6). On the EndPoint, the Blue pair is broken out to PTH pins labeled **VCC1** and **GND** and the Green pair is broken out to PTH pins labeled **VCC2** and **GND2**.

We've also broken out the **VCC2** and **GND2** lines to a pair of PTH pins on the MidPoint if users would like to use them for powering peripherals or running the voltage over the QwiicBus chain on separate wires.

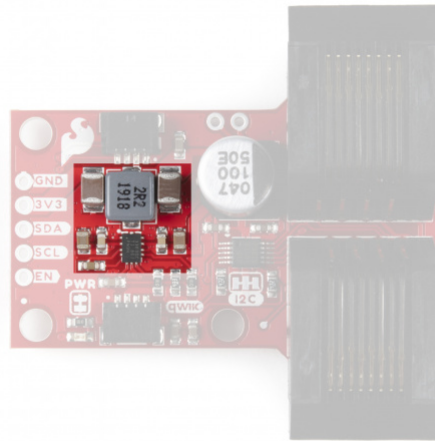


VCC1 is tied to both VDDA and VDDDB on the PCA9615 so long as the 0-1 jumper is **CLOSED** (more on that in the following sections). VCC2 and GND2 are connected to the Ethernet Green pair intended for powering the buck regulator on any attached MidPoint(s). If users are powering the QwiicBus through either of these voltage inputs, the Bypass jumper on all MidPoints must be **OPENED**. Using the Green or Blue pair to provide voltage to the QwiicBus requires the proper power configuration covered in the Hardware Assembly section.

Lastly, you may notice the VCC2 and GND2 PTH pins on the EndPoint are labeled **GRN** and **GRNW** when viewed from the top. The Green pair can be used as an isolated signal line by opening the VDDA and GND jumpers on the EndPoint. Note, when using the Green pair in this way, make sure the GND2 jumper(s) on any attached MidPoints are opened.

Buck Regulator - *MidPoint Only*

The MidPoint includes the LMR33630 Simple Switcher® variable buck regulator from Texas Instruments®. The regulator accepts an input voltage between **3.8V** to **36V** so you can send higher voltages over the Ethernet cable to render any voltage drop over long cables negligible and the **3A@3.3V** output provides a larger current source for any devices attached to the MidPoint. The LMR33630 has a variable voltage output that the MidPoint design sets to **3.3V**.



The buck regulator is not powered by default but the design features a dual jumper labeled PSEL users can adjust to enable the buck regulator to power the MidPoint(s) from a higher voltage (**5V** or **3.8V** to **36V**) via the Blue and Green pair pins broken out on the EndPoint and MidPoint. Read on to the Solder Jumpers sub-section below and the Hardware Assembly section for more information on how to use these alternate power configurations.

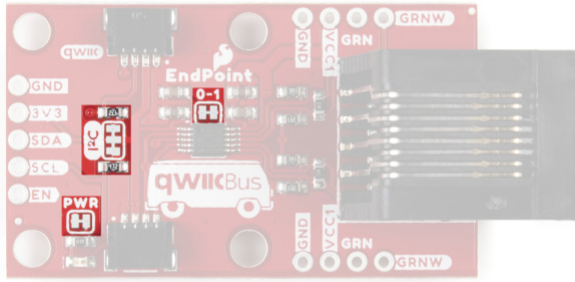
Note: While the buck regulator is rated for **3A**, sourcing max current through the regulator continuously will cause it to heat up and decrease its efficiency. Running max current through the regulator for an extended period of time can also damage the IC and is not recommended.

Solder Jumpers

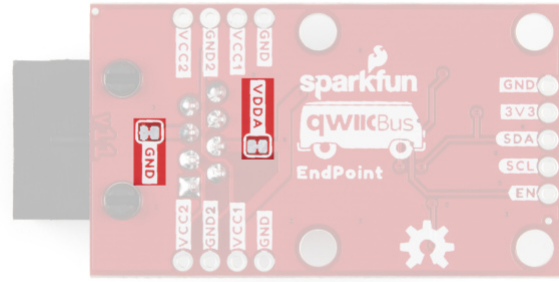
If you have never worked with solder jumpers and PCB traces before or would like a quick refresher, check out our [How to Work with Solder Jumpers and PCB Traces](#) tutorial for detailed instructions and tips.

In this section we'll cover the jumpers found on both QwiicBus boards and their functionality, default state and a few things to take note of prior to adjusting them.

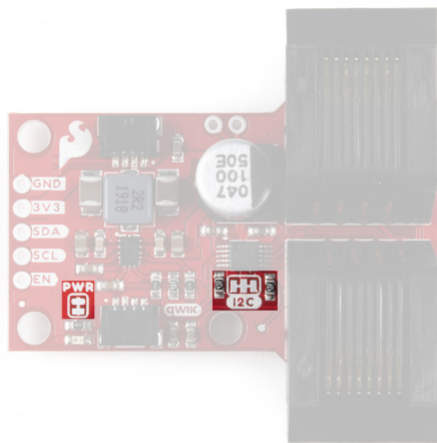
Since this guide covers both QwiicBus boards (EndPoint and MidPoint) we'll denote which jumpers are found on both boards as well as board-specific jumpers. The EndPoint's five jumpers are labeled: **I2C**, **PWR**, **VDDA**, **GND** and **0-1**. The six MidPoint jumpers are labeled: **I2C**, **PWR**, **BP**, **PSEL**, **GND1** and **GND2**.



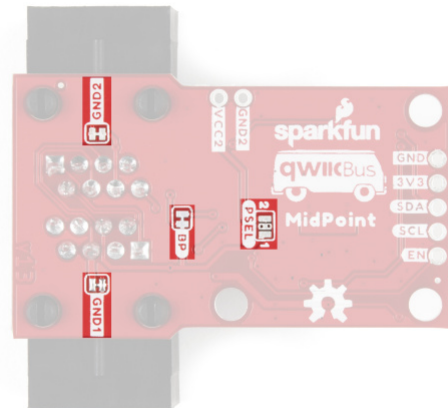
QwiicBus EndPoint Solder Jumpers - Top



QwiicBus EndPoint Solder Jumpers - Bottom



QwiicBus MidPoint Solder Jumpers - Top



QwiicBus MidPoint Solder Jumpers - Bottom

I²C Pull-Up Jumper - *EndPoint and MidPoint*

This jumper connects the SDA and SCL lines to VDDA of the PCA9615 (normally **3.3V**) via a pair of **4.7kΩ** resistors. The jumper's default state is **CLOSED**. To disable the pull-up resistors open the jumper by severing the trace between the three pads.

Note: Each MidPoint will need its own set of pull-up resistors on the I²C side of the PCA9615. We recommend disabling any pull-up resistors on devices connected to the MidPoint's Qwiic connectors or the SDA/SCL pins since if multiple devices have their pull-up resistors enabled, the parallel equivalent resistance can create too strong of a pull-up for the bus to operate correctly.

Power (PWR) LED Jumper - *EndPoint and MidPoint*

This jumper connects the Power LED's cathode to **3.3V** via a **1kΩ** resistor. The jumper is **CLOSED** by default. Open the jumper by severing the trace between the two pads to disable the power LED on either QwiicBus board.

VDDA and GND Jumpers - *EndPoint*

The VDDA and GND jumpers connect VDDA (if the 0-1 jumper is closed) and Ground to the Green twisted pair on the RJ-45 jacks and Ethernet cable. Their default state is **CLOSED**. If users do not need the Green pair for a power input, these jumpers can be opened so that pair is available for other data lines or redundant power connections. Most users will want to leave these jumpers alone. If using the Green pair as an independent signal, make sure the GND2 jumper on any MidPoints is **OPEN**.

0-1 Jumper - *EndPoint*

The 0-1 jumper nets VDDA and VDDDB on the PCA9615 together. By default this jumper is **CLOSED** to power both VDDA and VDDDB with the same supply voltage (**3.3V** in the default power configuration). Open this jumper if using two separate voltages for VDDA (VCC) and VDDDB (VCC1). If using separate voltages, make sure they fall within the voltage ranges for VDDA (**2.3V-5.5V**) and VDDDB (**3.0V-5.5V**).

Heads Up! When using any of the alternate power configurations, the 0-1 jumper on the *primary / first* EndPoint should be **OPEN** but the 0-1 jumper on the *terminating / last* should be **CLOSED** to avoid leaving VDDDB on the terminating EndPoint floating. Take note that with the 0-1 jumper **CLOSED**, **5V** is sent to the Qwiic connectors and **3.3V** PTH pin on the terminating EndPoint.

Bypass (BP) Jumper - *MidPoint*

The Bypass jumper (labeled BP on the board) nets VCC1 with the **3.3V** rail on the MidPoint. By default, this jumper is **CLOSED**. When the QwiicBus system is powered at **3.3V**, this jumper can remain closed but if the PSEL jumper is adjusted to use anything over **3.3V** to power the QwiicBus, this jumper *must* be **OPEN**.

Power Select (PSEL) Jumper - *MidPoint*

This dual jumper selects which voltage supplies the input voltage for the MidPoint buck regulator. The Power Select jumper defaults to **all open** and the buck regulator is *unpowered*. In this default setting, power for each EndPoint is **3.3V** and provided over the Blue pair of the Ethernet cable from the microcontroller/SBC or dedicated **3.3V** power supply.

If using the **5V** power configuration, this jumper should be closed to set it to the **"1"** side. When set to **"1"**, the input voltage is still provided over the Blue pair of the Ethernet cable but at **5V**.

If using the **3.8V-36V** power configuration, close the jumper to set it to the **"2"** side. When set to **"2"**, input voltage is provided over the Green pair of the Ethernet cable.

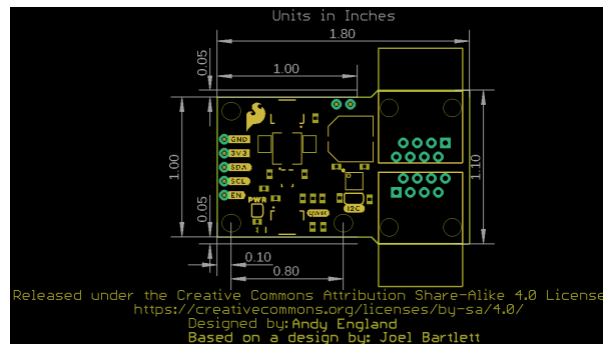
Important! If either alternate setting of the PSEL Jumper is used, the Bypass Jumper *must be OPEN*.

GND1 and GND2 Jumpers - *MidPoint*

The GND1 and GND2 jumpers net both input voltages to a common ground. Both jumpers are **CLOSED** by default. In advanced-use cases, users can open one or both of these jumpers to reduce noise and/or prevent ground loops on QwiicBus boards further down the chain. Most users will want to leave these jumpers alone.

Board Dimensions

The QwiicBus EndPoint PCB measures identically to the Differential I²C Breakout at 1.75in x 1.00in (44.45mm x 25.40mm). The MidPoint measures 1.80in x 1in (45.72mm x 25.40mm) and is flared to 1.10in (27.94mm) wide at the RJ-45 end. The RJ-45 connectors extend roughly 0.30in (7.62mm) from the edge of the PCB on both boards.



Hardware Assembly

In this section we'll go over configuring your QwiicBus EndPoint and MidPoint as how to assemble the QwiicBus circuit. Before we start connecting anything there are a couple of things to take note of.

Ethernet Cables

Make sure your Ethernet cables are straight-through (i.e. Pin 1 on one end of the cable is connected to Pin 1 on the other end) and *not* crossover. Most Ethernet cables sold are straight-through so you probably have nothing to worry about here. If you're not sure what type of Ethernet cable you have, you can test for pin continuity with a digital multimeter.

Also take the voltage drop over the Ethernet cables into account. For longer chains or for circuits with many devices on multiple MidPoints, consider using one of the alternate power configurations so the voltage drop is negligible. The formula below can help you calculate the voltage drop at different lengths of CAT-6/Ethernet cable and whether or not you will need to consider one of the alternate power options:

$$V_{drop} = IR$$

In this formula, I is the current through the object in Amperes and R is the resistance of the wires in Ohms. Most CAT-6/Ethernet cable will have 24AWG internal wires and depending on the quality of the cable will be either copper or aluminum. Refer to a table like this one from PowerStream to determine the resistance in $\Omega/1000\text{ft}$ or Ω/km to help calculate the approximate voltage drop.

I²C Pull-Up Resistors

Take time to note which devices you will have connected to your EndPoint and MidPoint's I²C bus side have their pull-up resistors enabled and what voltage they are tied to. Almost all Qwiic products have pull-up resistors enabled by default that pull the SDA/SCL lines connected directly to the I²C bus to **3.3V**.

The PCA9615 requires pull-up resistors on the two-wire I²C bus so **each** EndPoint and MidPoint must have at least one pair of pull-up resistors enabled on the two-wire side. The pull-up resistors do *not* translate through to the differential side of the PCA9615. To help make things a bit simpler, each QwiicBus EndPoint and MidPoint come with pull-up resistors on both SDA and SCL lines tied to VDDA. Normally, VDDA is **3.3V** but can be as high as **5V** on the EndPoint depending on the power configuration settings.

Note: If multiple devices are connected and you have a single device with pull-up resistors, make sure the voltage they are netted to is within the operating voltage range of all other I²C devices on the bus (normally **3.3V** for Qwiic devices). If needed, you can shift the voltage to safe levels with a level shifter or voltage divider. If you're not familiar with logic levels and how to shift them, take a read through this tutorial.

Power Configuration Settings

As we mentioned in the Hardware Overview section of this guide, both QwiicBus boards feature a plethora of jumpers as well as two pairs of dedicated power PTH pins to configure how to power your QwiicBus circuit. Let's take a closer look at the three options available for powering the QwiicBus.

⚠ Important! When using alternate power configurations make sure to double check the jumpers on all EndPoints and MidPoints on your QwiicBus are set properly per the instructions below. Any QwiicBus boards or other Qwiic devices on your bus can be damaged if the circuit is assembled improperly.

Default - Entire System at 3.3V

In this configuration, the entire system is powered at **3.3V** and the buck regulator(s) on the MidPoint(s) are unpowered. On the MidPoint(s), the Bypass jumper is **CLOSED** and both sides of the PSEL jumper are **OPEN**. **3.3V** is provided either from your development board or single-board computer through the Qwiic connector or the **3.3V** pin on the primary EndPoint.

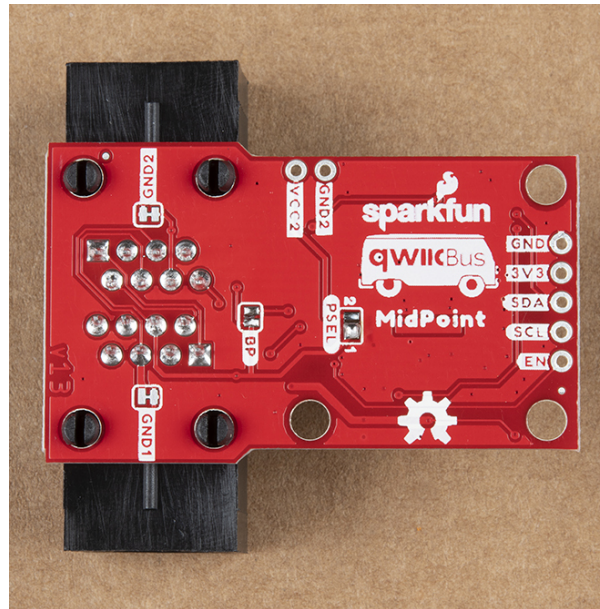
Take note, in this configuration where all power is provided over the CAT-6/Ethernet cables, your total current draw will for attached devices will be limited to **~550mA** with standard CAT-6/Ethernet cables due to the physical constraints of the wires inside. Anything beyond **550mA** runs the risk of damaging the wires and can create a fire hazard.

The **3.3V** supply voltage is then transferred from the initial EndPoint over the CAT-6/Ethernet cables in your QwiicBus system. Note that over long lengths of wire, you will see voltage drop and devices further down your QwiicBus chain may misbehave if the voltage drops below their operating range. Refer to the formula and links in the Ethernet Cables subsection above for calculating the voltage drop in your circuit.

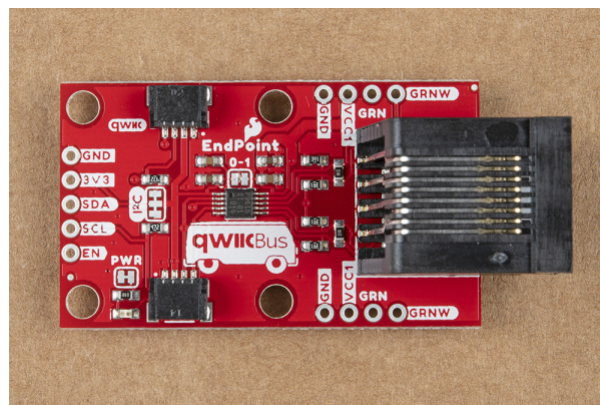
Alternate 1 - VCC1 at 5V

While the default **3.3V** configuration should work just fine for shorter QwiicBus chains, we found the PCA9615 functions better when powered with **5V**. Powering VCC1 with **5V** also allows us to also use the buck regulator on the MidPoint to source up to **3A@3.3V** to devices connected to the MidPoint. Using this configuration requires adjusting both the EndPoint and MidPoint as well as multiple power supplies.

First and most importantly, the Bypass jumper on each MidPoint *must be OPEN*. On each MidPoint, adjust the PSEL jumper to the **"1"** side (see image below).



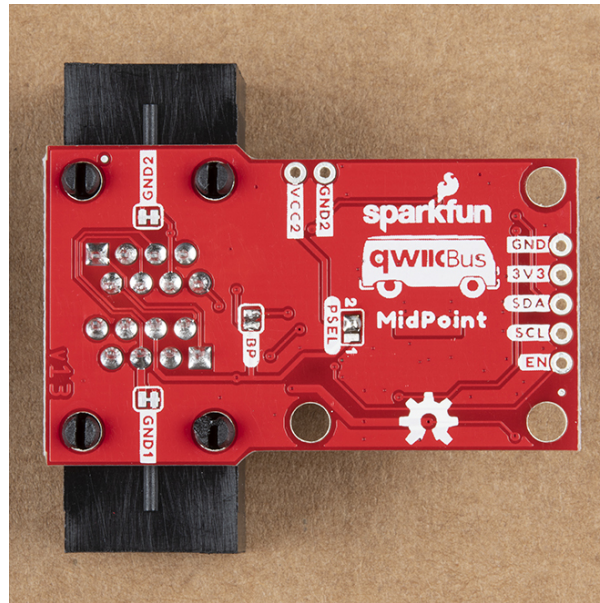
It is also strongly recommended to **OPEN** the 0-1 jumper on the primary / first EndPoint to isolate VCC (VDDA) from VCC1 (Vddb) but leave the 0-1 jumper on the terminating EndPoint **CLOSED**. Leaving the 0-1 jumper closed will send **5V** to any Qwiic devices attached to your terminating / last EndPoint.



With the jumpers adjusted and your QwiicBus circuit assembled including any peripheral devices attached to your MidPoint(s), connect your **5V** source to the VCC1 and GND PTH pins on the primary EndPoint. If the 0-1 jumper is opened, **3.3V** should be provided over the Qwiic connectors or through the **3.3V** PTH pin to power VCC (VDDA) on the primary EndPoint.

Alternate 2 - VCC2 at 3.8V to 36V

For this configuration, we use two separate voltages to power the EndPoints and MidPoint(s). First and most importantly, the Bypass jumper on each MidPoint *must be* **OPEN**. Adjust the PSEL jumper to the **"2"** position on each MidPoint.



If you plan to power VCC1 with **5V** it is recommended to **OPEN** the 0-1 jumper on the primary EndPoint to isolate VCC from VCC1. As covered above, we recommend leaving the 0-1 jumper **CLOSED** on the terminating EndPoint.

After adjusting the jumpers and assembling your QwiicBus circuit including any peripheral devices attached to your MidPoint(s), connect **3.8V** to **36V** to the VCC2 and GND2 PTH pins on the primary EndPoint. **3.3V** for the first EndPoint should be provided from the microcontroller/SBC via the Qwiic connectors or a dedicated source through the **3.3V** PTH pin.

Alternate 3 - VCC1 at 5V and VCC2 at 6V to 36V

This advanced configuration uses multiple power supply to allow for optimal performance of the QwiicBus over very long distances with multiple MidPoints. As mentioned above, we have found the QwiicBus operates best over long distances at **5V** (particularly VDDDB @**5V**). In this configuration, VCC1 is powered with **5V** and VCC2 is powered with **6V-36V**. Before connecting power supplies, a few jumpers must be adjusted.

First, the Bypass jumper on each MidPoint *must be OPEN*. Set the PSEL jumper on all MidPoints to the **"2"** position. The 0-1 jumper on the primary EndPoint is **OPEN** and the 0-1 jumper on the terminating EndPoint is **CLOSED**. This isolates VDDA and VDDDB on the first EndPoint so the Qwiic connectors on the first EndPoint are at **3.3V**. As we mentioned before, the Qwiic connectors and **3.3V** PTH pin on the terminating EndPoint will be at **5V** so be careful what is connected to that EndPoint.

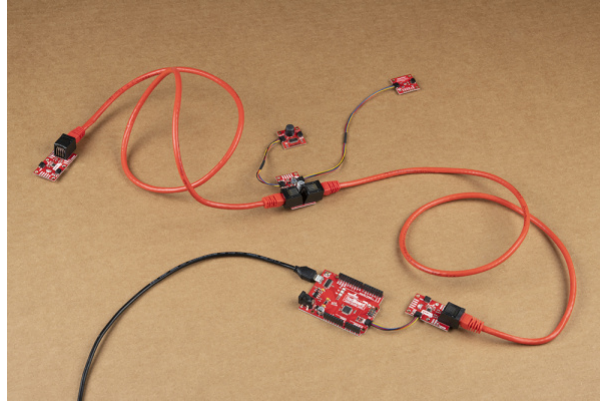
After adjusting jumpers on all the QwiicBus boards and assembling the rest of your circuit (including connecting any Qwiic devices to MidPoints/EndPoints), connect the **5V** power supply to the VCC1 and GND PTH pins and the power supply sourcing **6V** to **36V** to the VCC2 and GND2 PTH pins to the primary EndPoint.

Assembling the QwiicBus Circuit

⚡ Important! Make sure your QwiicBus circuit is **not powered** when adding or removing devices to your MidPoint(s) and EndPoint. Connecting and disconnecting devices while your QwiicBus is powered can cause voltage spikes and damage the PCA9615.

After taking into consideration how you intend to configure your QwiicBus circuit it's time to assemble it. Adjust the appropriate jumpers (if necessary), connect any Qwiic/I²C devices to the EndPoints and MidPoint(s) and connect the QwiicBus boards to each other with Ethernet cable plugged into the RJ-45 connectors on each board. Once

the devices are all connected together, power up the circuit. The photo below shows the QwiicBus assembled in the default configuration (all boards powered with **3.3V**) so your circuit may differ.



*A completed QwiicBus Kit circuit using a SparkFun RedBoard Qwiic and operating at **3.3V***

With everything adjusted and connected properly, that's all you need to assemble your QwiicBus circuit. Go forth and build ridiculously long, wired I²C circuits to your heart's content!

Troubleshooting

Here a few troubleshooting tips for the SparkFun QwiicBus.

I²C Communication

The PCA9615 supports I²C clock speeds up to 1MHz at short distances but as the cable length increases, high clock speeds can become more unreliable. If you find you are losing data or having unreliable communication on your QwiicBus, try reducing the clock speed as a quick software fix. Switching to one of the alternate power configurations can also help boost the reliability over long distances at high clock speeds.

Voltage Drop

Over long distances of cable the voltage may drop below the operating voltage range of the PCA9615 or attached devices. The formula below can help you calculate the voltage drop at different lengths of CAT-5/Ethernet cable and whether or not you will need to use one of the alternate power options:

$$V_{drop} = IR$$

I is the current through the object in Amperes and R is the resistance of the wires in Ohms. Most CAT-5/Ethernet cable will have 24AWG internal wires and depending on the quality of the cable will be either copper or aluminum. Refer to a table like this one from PowerStream to determine the resistance in $\Omega/1000\text{ft}$ or Ω/km to help calculate the approximate voltage drop over your QwiicBus circuit.

Recommended Power-Up/Power-Down Procedure

Remember to power down the QwiicBus circuit *before* connecting or disconnecting any devices to the chain. Connecting or disconnecting devices to the QwiicBus can damage the PCA9615 on the QwiicBus boards if it is powered on.

Floating VDDDB on the Terminating EndPoint

Reminder, when using any of the alternate power configurations, the 0-1 jumper on the *primary / first* EndPoint should be **OPEN** but the 0-1 jumper on the *terminating / last* should be **CLOSED** to avoid leaving VDDDB on the terminating EndPoint floating. Take note that with the 0-1 jumper **CLOSED**, **5V** is sent to the Qwiic connectors and

3.3V PTH pin on the terminating EndPoint.

General Troubleshooting

Need help?

If your product is not working as you expected or you need technical assistance or information, head on over to the SparkFun Technical Assistance page for some initial troubleshooting.

If you don't find what you need there, the SparkFun Forums are a great place to find and ask for help from our Tech Support team and community. If this is your first visit, you'll need to create a Forum Account to search product forums and post questions.

Resources and Going Further

Now that you have your QwiicBus set up and configured to your liking, it's time to start creating some I²C device networks! For more information, take a look at the following links:

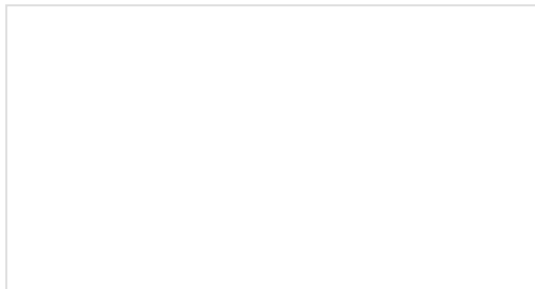
QwiicBus EndPoint Links

- Schematic (PDF)
- Eagle Files (ZIP)
- Dimensional Drawing (PNG)
- PCA9615 Datasheet (PDF)
- Hardware GitHub Repository

QwiicBus MidPoint Links

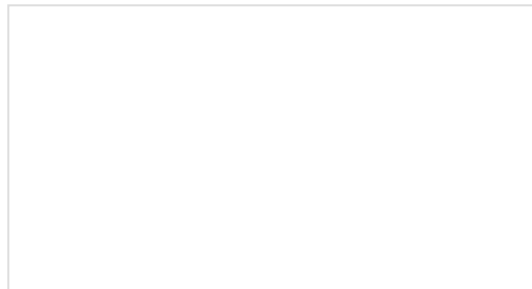
- Schematic (PDF)
- Eagle Files (ZIP)
- Dimensional Drawing (PNG)
- LMR33630 Datasheet (PDF)
- Hardware GitHub Repository

For inspiration for your I²C project using the QwiicBus, check out the following tutorials:



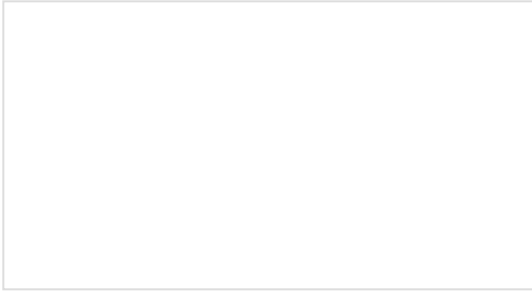
AS726X NIR/VIS Spectral Sensor Hookup Guide

It's now easier than ever to measure and characterize how different materials absorb and reflect different wavelengths of light. The AS726X spectral sensors allow you to detect wavelengths in the visible range (VIS) and near infrared range (NIR)!

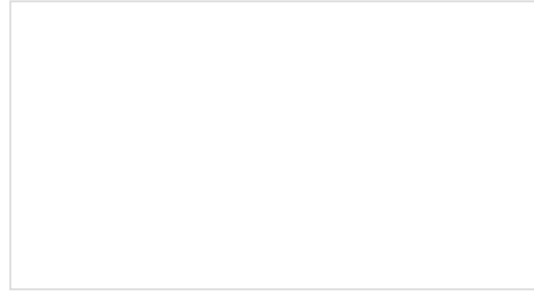


Qwiic MP3 Trigger Hookup Guide

Playing MP3s has never been easier.



Qwiic pHAT for Raspberry Pi Hookup Guide
Get started interfacing your Qwiic enabled boards with your Raspberry Pi. The Qwiic pHAT connects the I2C bus (GND, 3.3V, SDA, and SCL) on your Raspberry Pi to an array of Qwiic connectors.



SparkFun Qwiic MicroPressure Hookup Guide
Get started using your Qwiic MicroPressure breakout board with this hookup guide.