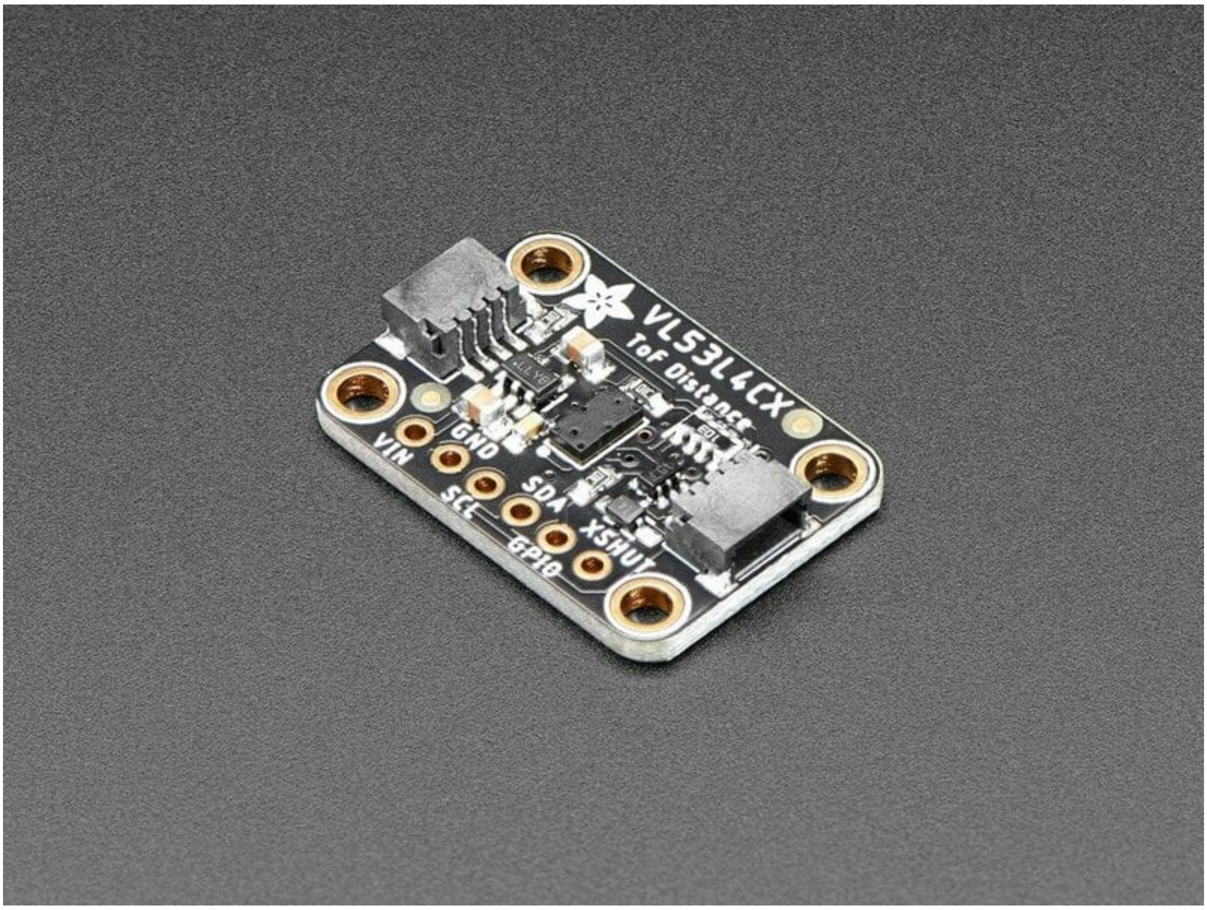
Adafruit VL53L4CX Time of Flight Distance Sensor

Created by Liz Clark



https://learn.adafruit.com/adafruit-vl53l4cx-time-of-flight-distance-sensor

Last updated on 2024-10-03 10:54:11 AM EDT

Table of Contents

Overview 3

- Removing the Protective Tape

Pinouts 6

- Power Pins
- I2C Logic Pins
- Other Pins
- Power LED and Jumper

Arduino 7

- Wiring
- Library Installation
- Load Example

Arduino Docs 10

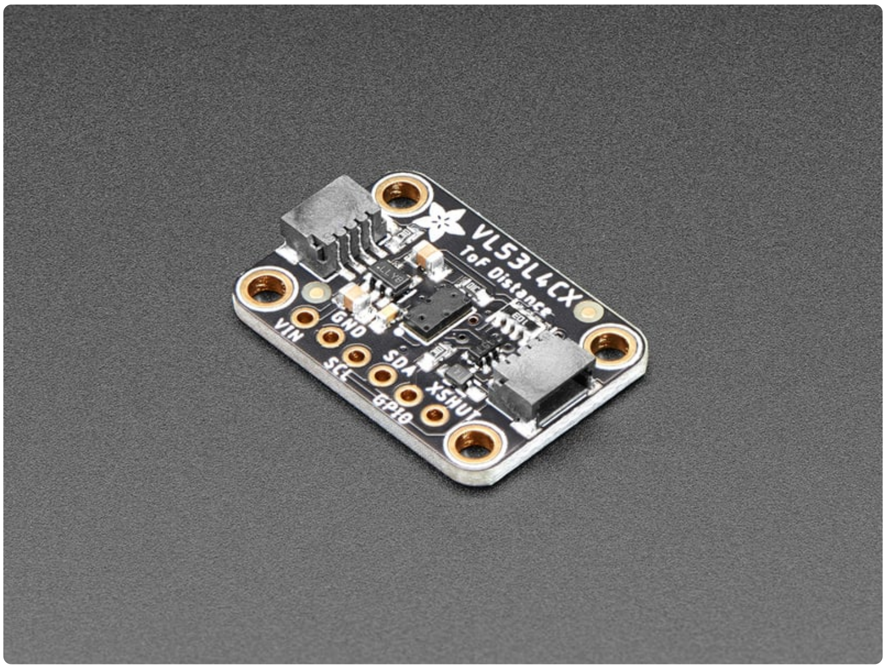
WipperSnapper 10

- What is WipperSnapper
- Wiring
- Usage

Downloads 16

- Files
- Schematic and Fab Print

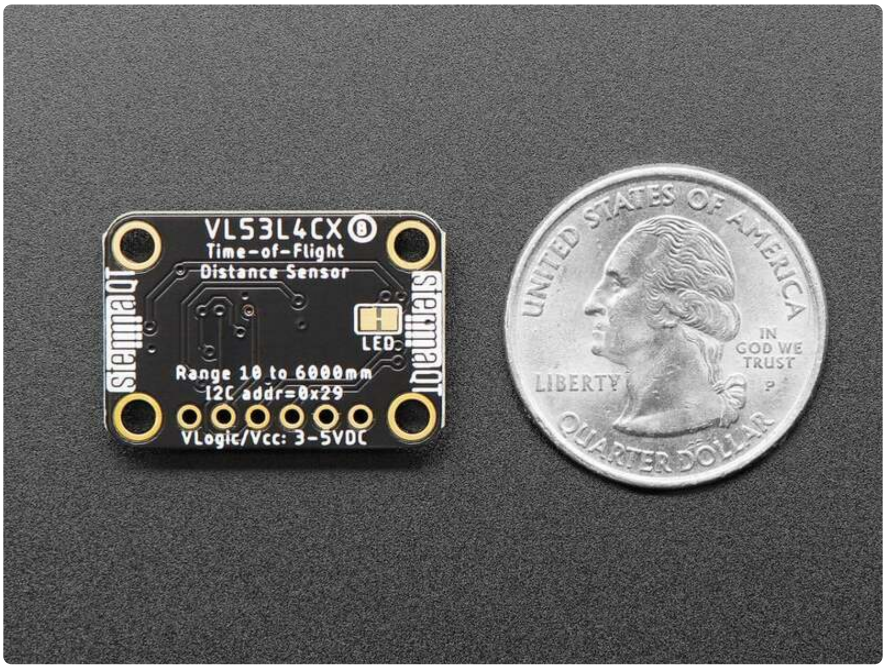
Overview



The **Adafruit VL53L4CX Time of Flight Sensor** is another great Time of Flight (https://adafru.it/ZZD) distance sensor from ST in the VL5 series of chips. This one is great for **long distances -** it goes up to 6 meters compared with 4 meter max of the VL53L1X.

The sensor contains a very tiny invisible laser source and a matching sensor. The VL53L4CX can detect the "time of flight", or how long the light has taken to bounce back to the sensor. Since it uses a very narrow light source, it is good for determining distance of only the surface directly in front of it.

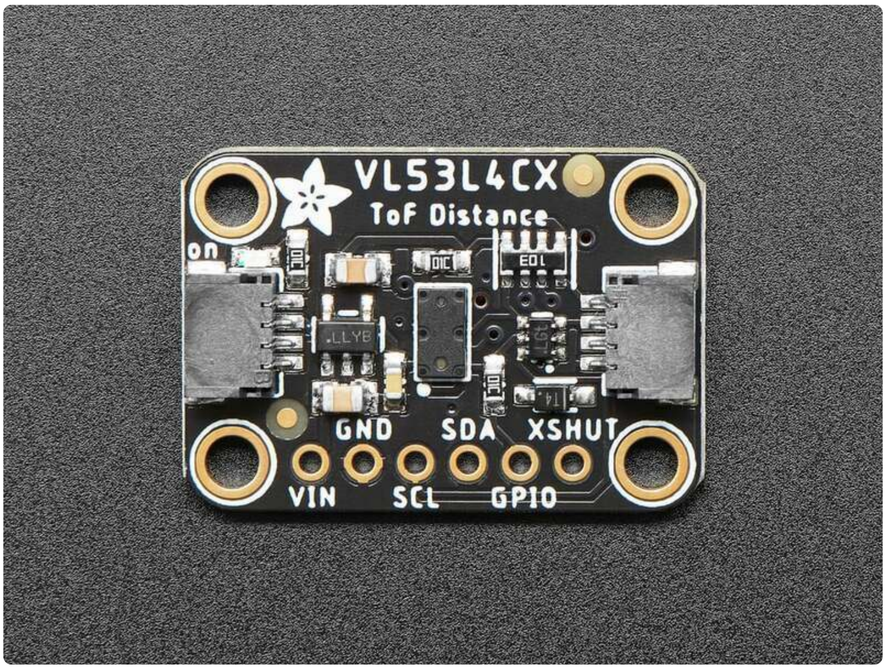
Unlike sonars that bounce ultrasonic waves, the 'cone' of sensing is very narrow. Unlike IR distance sensors that try to measure the amount of light bounced, the VL53 is much more precise and doesn't have linearity problems or 'double imaging' where you can't tell if an object is very far or very close.



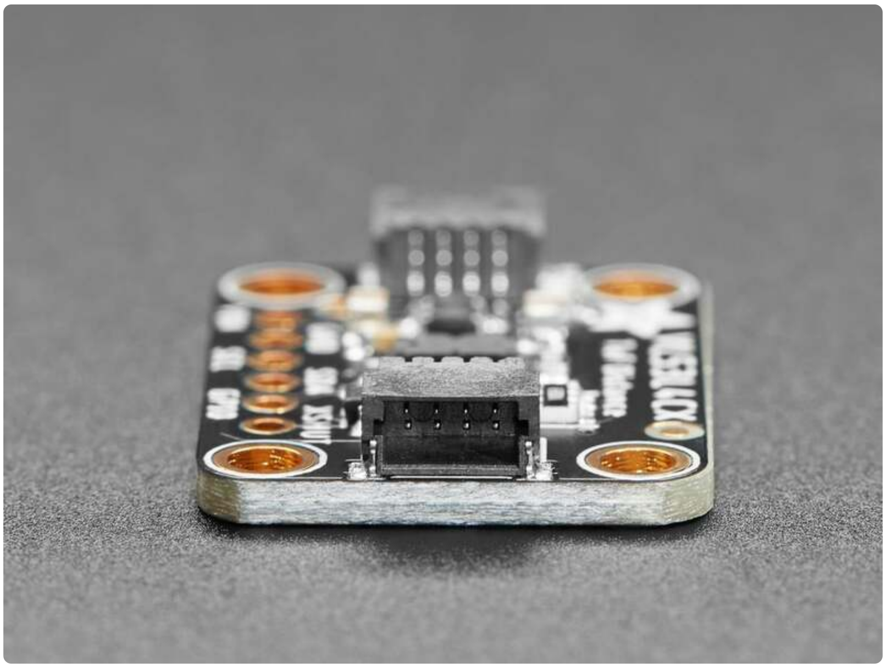
This breakout ships with a protector over the sensor. It must be removed before use! See details below.

This is a 'big sister' of the VL53L4CD ToF sensor (http://adafru.it/5396) and can handle about ~**1 to 6000mm** of range distance, it also has some ability to do 'multi object detection'. Basically it can identify when more than one object is in view and tell you the two distances.

Please note, the Arduino driver for this chip does not support 'small memory' boards like the ATmega328 - you'll need a SAMD21, SAMD51, ESP, etc chip with 50K of flash memory available!



The sensor is small and easy to use in any robotics or interactive project. Since it needs 2.8V power and logic we put the little fellow on a breakout board with a regulator and level shifting. You can use it with any 3-5V power or logic microcontroller with no worries. Works great with the **3.3V logic level of a Feather, or the 5v level of a Arduino Mega**, this breakout is ready to work with most common microcontrollers or SBCs. and since it speaks I2C, you can easily connect it up with two data wires plus power and ground.



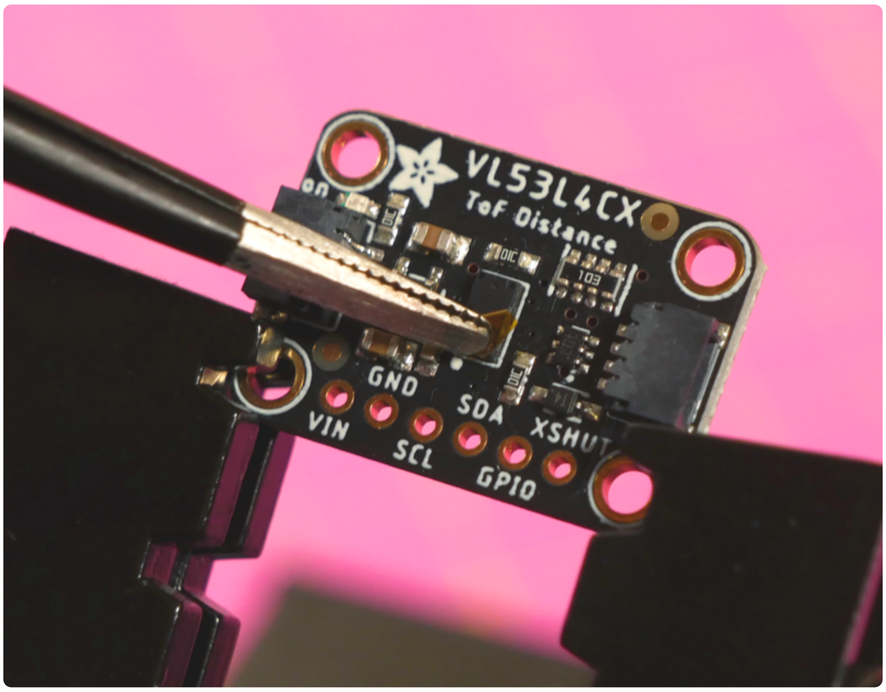
As if that weren't enough, we've also added SparkFun qwiic (https://adafru.it/Fpw) compatible STEMMA QT **(https://adafru.it/Ft4)** connectors for the I2C bus **so you don't even need to solder.** Just wire up to your favorite micro with a plug-and-play cable to get ToF data ASAP. For a no-solder experience, just wire up to your favorite micro, like the STM32F405 Feather (http://adafru.it/4382) using a STEMMA QT adapter cable. (https://adafru.it/JnB) The Stemma QT connectors also mean the VL53L4CX can be used with our various associated accessories. (https://adafru.it/Ft6) QT Cable is not included, but we have a variety in the shop (https://adafru.it/17VE)

Communicating to the sensor is done over I2C with an API written by ST, they have an Arduino library with an example for communication here (https://adafru.it/ZPd).

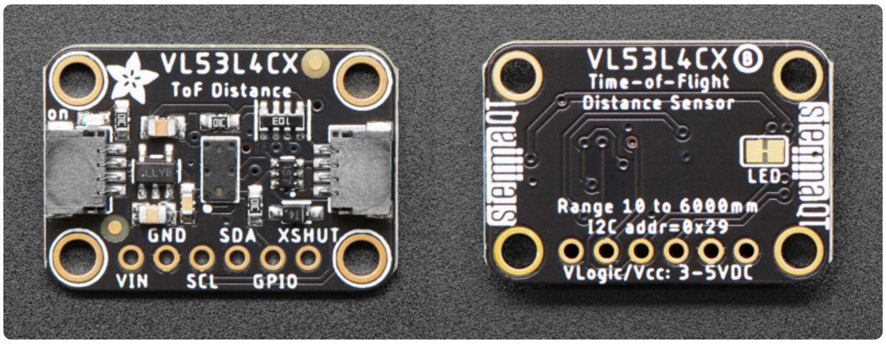
Removing the Protective Tape

Be careful when removing the tape! You don't want to damage the sensor.

Using tweezers (or some other appropriate tool), CAREFULLY remove the protective tape over the sensor, as seen in the image below. There is a small tab on the side of the tape that you can use to remove it.



Pinouts



This breakout ships with a protector over the sensor. It must be removed before use! See details at the bottom of the Overview page.

Power Pins

- **VIN** - This is the power pin. To power the board, give it the same power as the logic level of your microcontroller - e.g. for a 3V microcontroller like a Feather M4, use 3V, or for a 5V microcontroller like Arduino, use 5V.
- **GND** - This is common ground for power and logic.

I2C Logic Pins

The default I2C address for the VL53L4CX is **0x29**.

- **SCL** - I2C clock pin, connect to your microcontroller I2C clock line. There's a **10K pullup** on this pin.
- **SDA** - I2C data pin, connect to your microcontroller I2C data line. There's a **10K pullup** on this pin.
- **STEMMA QT** (https://adafru.it/Ft4) - These connectors allow you to connect to development boards with **STEMMA QT** connectors or to other things with various associated accessories (https://adafru.it/JRA).

Other Pins

- **GPIO** - This is the interrupt output pin, it is 2.8V logic level output - it can be read by 3.3V and most 5V logic microcontrollers.
- **XSHUT** - This is the shutdown pin. It is active low, and is logic-level shifted so you can use 3V or 5V logic.

Power LED and Jumper

- **Power LED** - This LED is a green LED located on the front of the board. It is labeled **on**.
- **LED jumper** - This jumper is located on the back of the board. Cut the trace on this jumper to cut power to the "on" LED.

Arduino

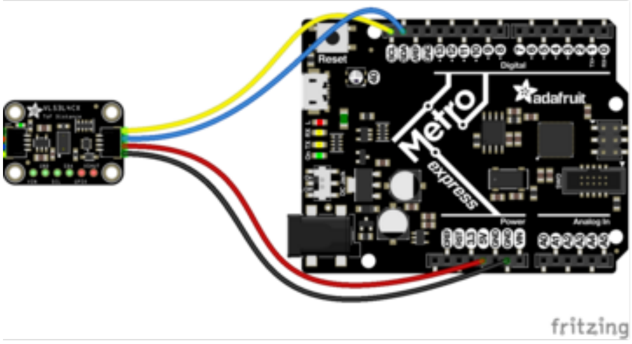
Using the VL53L4CX with Arduino involves wiring up the sensor to your Arduino-compatible microcontroller, installing the STM32duino VL53L4CX (https://adafru.it/ZPd) library and running the provided example code.

Please note, the Arduino driver for this chip does not support 'small memory' boards like the ATmega328 - you'll need a SAMD21, SAMD51, ESP, etc chip with 50K of flash memory available!

Wiring

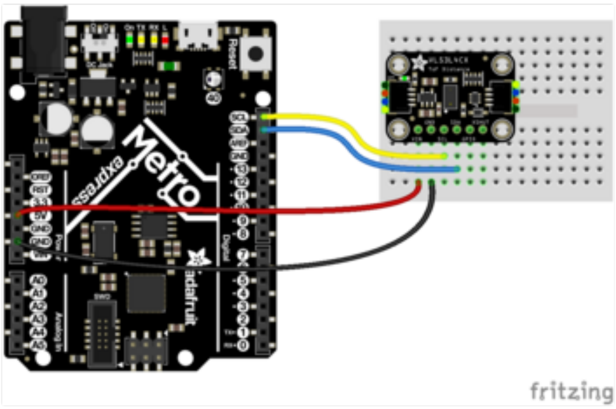
Wire as shown for a **5V** board. If you are using a **3V** board, like an Adafruit Feather, wire the board's **3V** pin to the VL53L4CX **VIN**.

Here is an Adafruit Metro Express wired up to the VL53L4CX using the STEMMA QT connector:



Board 5V to **sensor VIN (red wire)**
Board GND to **sensor GND (black wire)**
Board SCL to **sensor SCL (yellow wire)**
Board SDA to **sensor SDA (blue wire)**

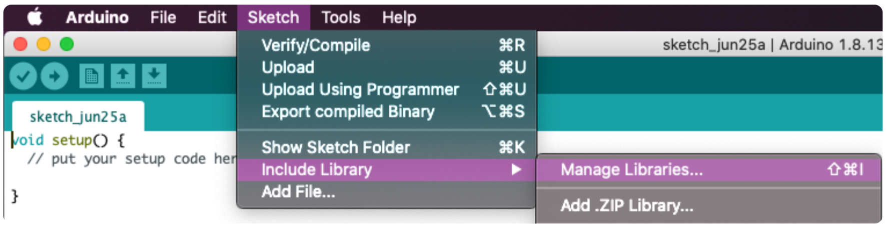
Here is an Adafruit Metro Express wired up using a solderless breadboard:



Board 5V to **sensor VIN (red wire)**
Board GND to **sensor GND (black wire)**
Board SCL to **sensor SCL (yellow wire)**
Board SDA to **sensor SDA (blue wire)**

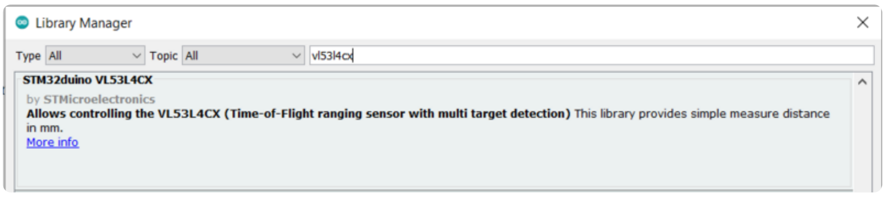
Library Installation

You can install the **VL53L4CX** library for Arduino using the Library Manager in the Arduino IDE.



```
void setup() {
  // put your setup code her
}
```

Click the **Manage Libraries ...** menu item, search for **VL53L4CX**, and select the **STM32duino VL53L4CX** library:



vl53l4cx

STM32duino VL53L4CX

by STMicroelectronics

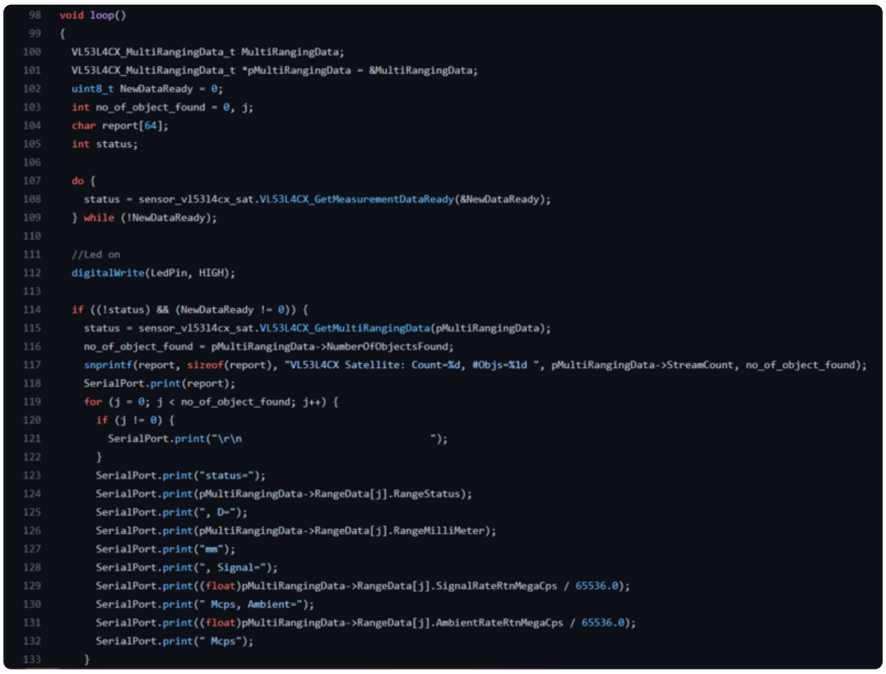
Allows controlling the VL53L4CX (Time-of-Flight ranging sensor with multi target detection) This library provides simple measure distance in mm.

More info

Load Example

Open up **File -> Examples -> STM32duino VL53L4CX -> VL53L4CX_Sat_HelloWorld** and upload to your Arduino wired to the sensor.

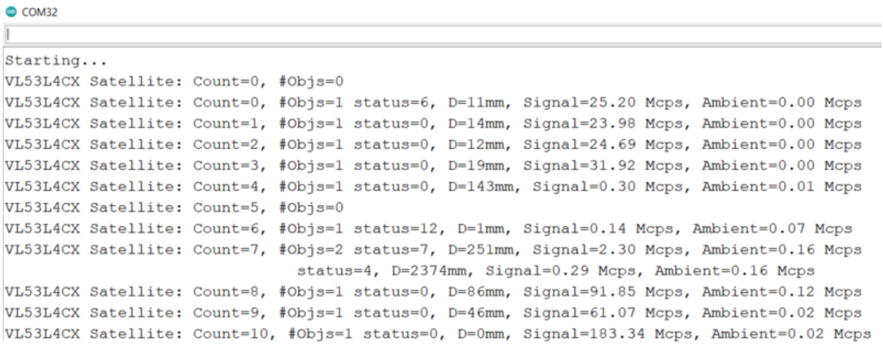
Note: this is only the loop in the screenshot.



```
void loop()
{
  VL53L4CX_MultiRangingData_t MultiRangingData;
  VL53L4CX_MultiRangingData_t *pMultiRangingData = &MultiRangingData;
  uint8_t NewDataReady = 0;
  int no_of_object_found = 0, j;
  char report[64];
  int status;

  do {
    status = sensor_vl53l4cx_sat.VL53L4CX_GetMeasurementDataReady(&NewDataReady);
  } while (!NewDataReady);

  //Led on
  digitalWrite(LedPin, HIGH);

  if ((!status) && (NewDataReady != 0)) {
    status = sensor_vl53l4cx_sat.VL53L4CX_GetMultiRangingData(pMultiRangingData);
    no_of_object_found = pMultiRangingData->NumberOfObjectsFound;
    snprintf(report, sizeof(report), "VL53L4CX Satellite: Count=%d, #Objs=%1d ", pMultiRangingData->StreamCount, no_of_object_found);
    SerialPort.print(report);
    for (j = 0; j < no_of_object_found; j++) {
      if (j != 0) {
        SerialPort.print("\r\n                               ");
      }
      SerialPort.print("status=");
      SerialPort.print(pMultiRangingData->RangeData[j].RangeStatus);
      SerialPort.print(", D=");
      SerialPort.print(pMultiRangingData->RangeData[j].RangeMilliMeter);
      SerialPort.print("mm");
      SerialPort.print(", Signal=");
      SerialPort.print((float)pMultiRangingData->RangeData[j].SignalRateRtnMegaCps / 65536.0);
      SerialPort.print(" Mcps, Ambient=");
      SerialPort.print((float)pMultiRangingData->RangeData[j].AmbientRateRtnMegaCps / 65536.0);
      SerialPort.print(" Mcps");
    }
```

Upload the sketch to your board and open up the Serial Monitor (**Tools -> Serial Monitor**) at 115200 baud. You should see the values from the sensor being printed out. The values read in the example are:

- `Count` - number of readings sent to the Serial Monitor
- `Objs` - Number of objects in front of the sensor
- `status` - Range status of the sensor
- `D` - Measurement of distance of an object from the sensor in millimeters
- `Signal` - Rate of the signal from the sensor
- `Ambient` - Ambient noise around the sensor

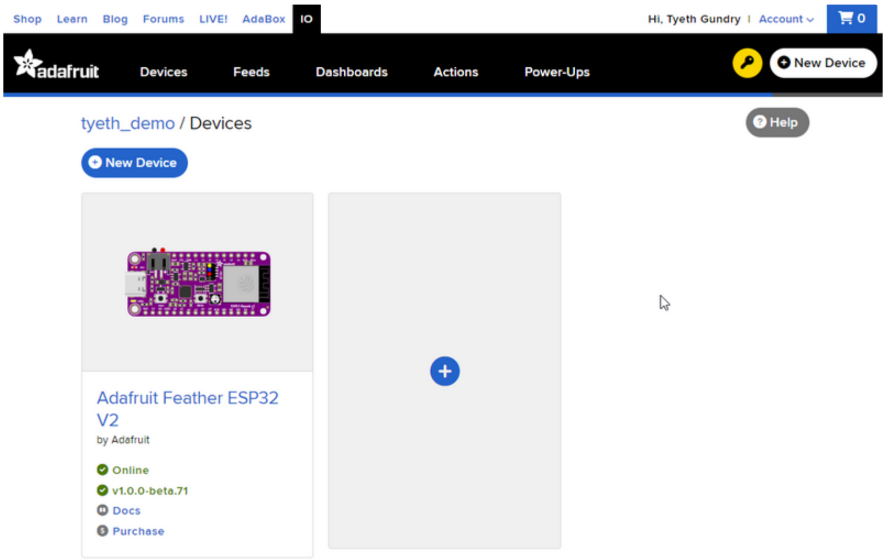


```
Starting...
VL53L4CX Satellite: Count=0, #Objs=0
VL53L4CX Satellite: Count=0, #Objs=1 status=6, D=11mm, Signal=25.20 Mcps, Ambient=0.00 Mcps
VL53L4CX Satellite: Count=1, #Objs=1 status=0, D=14mm, Signal=23.98 Mcps, Ambient=0.00 Mcps
VL53L4CX Satellite: Count=2, #Objs=1 status=0, D=12mm, Signal=24.69 Mcps, Ambient=0.00 Mcps
VL53L4CX Satellite: Count=3, #Objs=1 status=0, D=19mm, Signal=31.92 Mcps, Ambient=0.00 Mcps
VL53L4CX Satellite: Count=4, #Objs=1 status=0, D=143mm, Signal=0.30 Mcps, Ambient=0.01 Mcps
VL53L4CX Satellite: Count=5, #Objs=0
VL53L4CX Satellite: Count=6, #Objs=1 status=12, D=1mm, Signal=0.14 Mcps, Ambient=0.07 Mcps
VL53L4CX Satellite: Count=7, #Objs=2 status=7, D=251mm, Signal=2.30 Mcps, Ambient=0.16 Mcps
                               status=4, D=2374mm, Signal=0.29 Mcps, Ambient=0.16 Mcps
VL53L4CX Satellite: Count=8, #Objs=1 status=0, D=86mm, Signal=91.85 Mcps, Ambient=0.12 Mcps
VL53L4CX Satellite: Count=9, #Objs=1 status=0, D=46mm, Signal=61.07 Mcps, Ambient=0.02 Mcps
VL53L4CX Satellite: Count=10, #Objs=1 status=0, D=0mm, Signal=183.34 Mcps, Ambient=0.02 Mcps
```

Arduino Docs

Arduino Docs (https://adafru.it/ZPd)

WipperSnapper



What is WipperSnapper

WipperSnapper is a firmware designed to turn any WiFi-capable board into an Internet-of-Things device without programming a single line of code. WipperSnapper connects to Adafruit IO (https://adafru.it/fsU), a web platform designed (by Adafruit! (https://adafru.it/Bo5)) to display, respond, and interact with your project's data.

Simply load the WipperSnapper firmware onto your board, add credentials, and plug it into power. Your board will automatically register itself with your Adafruit IO account.

From there, you can add components to your board such as buttons, switches, potentiometers, sensors, and more! Components are dynamically added to hardware, so you can immediately start interacting, logging, and streaming the data your projects produce without writing code.

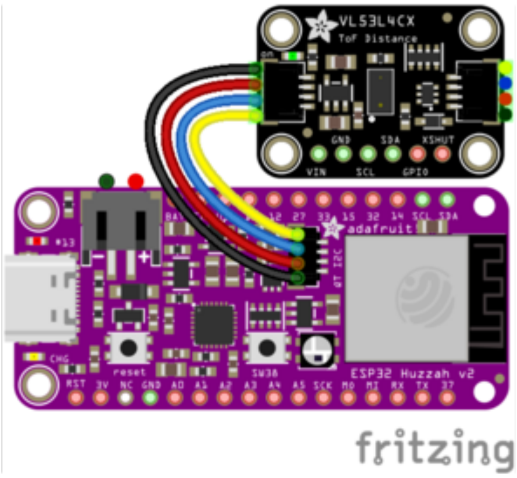
If you've never used WipperSnapper, click below to read through the quick start guide before continuing.

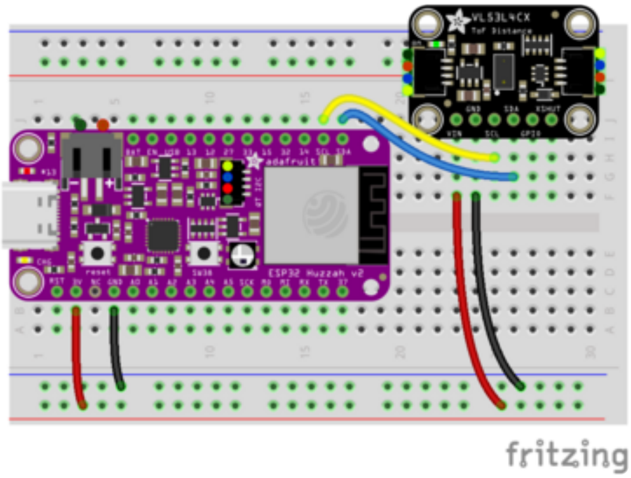
Quickstart: Adafruit IO WipperSnapper

https://adafru.it/Vfd

Wiring

First, wire up a VL53L4CX to your board exactly as follows. Here is an example of the VL53L4CX wired to an Adafruit ESP32 Feather V2 (http://adafru.it/5400) using I2C with a STEMMA QT cable (no soldering required) (http://adafru.it/4210)



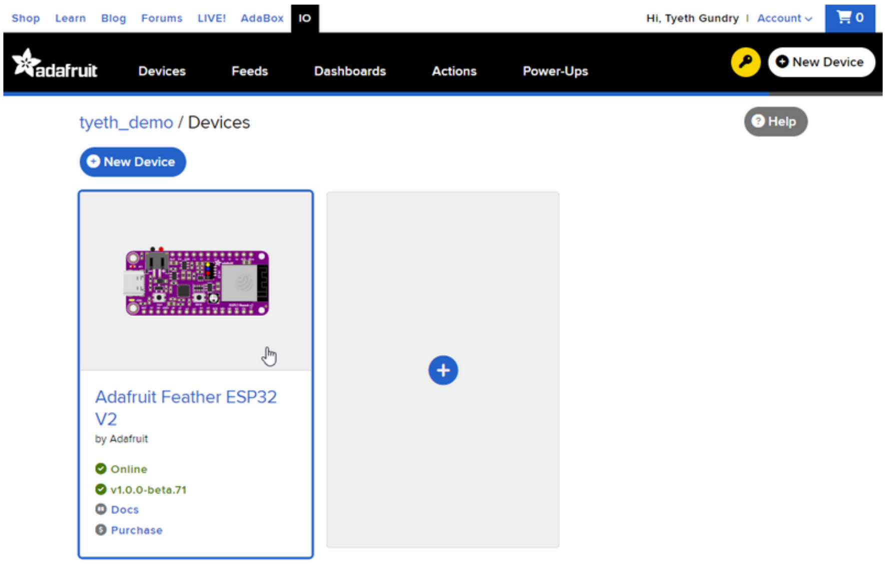


Board 3V to **sensor VIN (red wire on STEMMA QT)**
Board GND to **sensor GND (black wire on STEMMA QT)**
Board SCL to **sensor SCL (yellow wire on STEMMA QT)**
Board SDA to **sensor SDA (blue wire on STEMMA QT)**

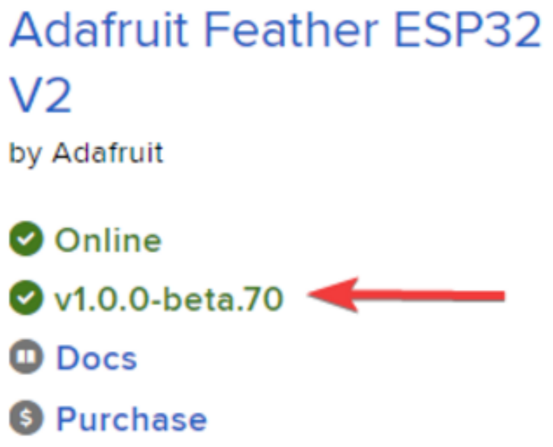
Usage

Connect your board to Adafruit IO Wippersnapper and navigate to the WipperSnapper board list (https://adafru.it/TAu).

On this page, **select the WipperSnapper board you're using** to be brought to the board's interface page.



If you do not see your board listed here - you need [to connect your board to Adafruit IO (https://adafru.it/Vfd)](https://adafru.it/Vfd) first.



Adafruit Feather ESP32 V2

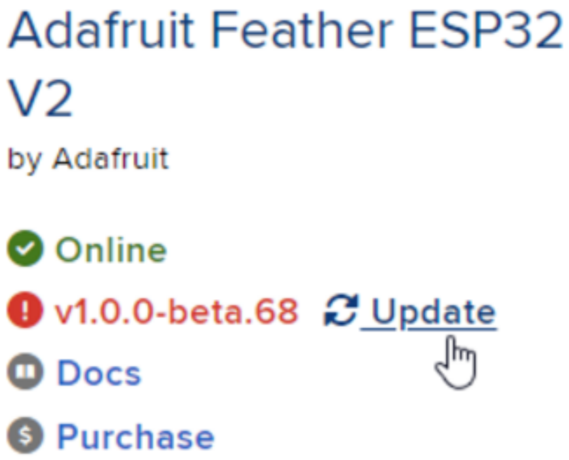
by Adafruit

✓ Online

✓ v1.0.0-beta.70

Docs

Purchase



Adafruit Feather ESP32 V2

by Adafruit

✓ Online

! v1.0.0-beta.68 Update

Docs

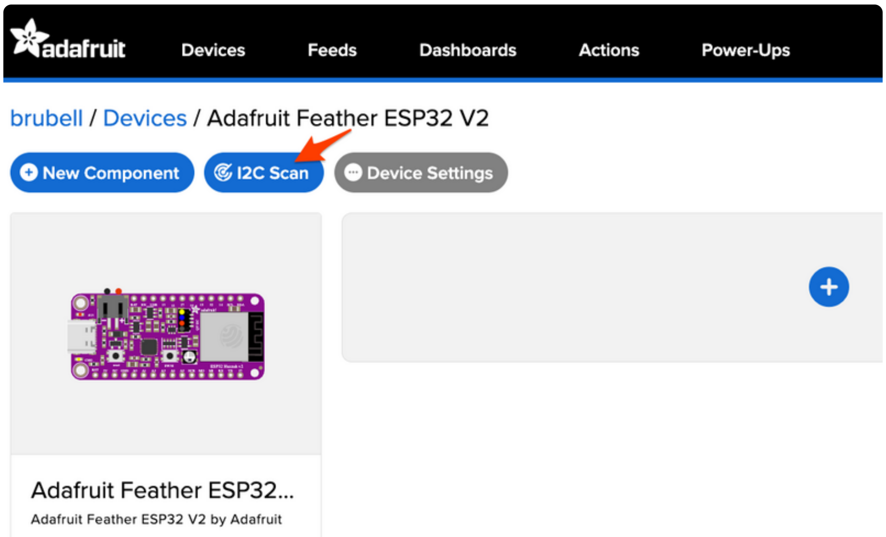
Purchase

On the device page, quickly **check that you're running the latest version of the WipperSnapper firmware**.

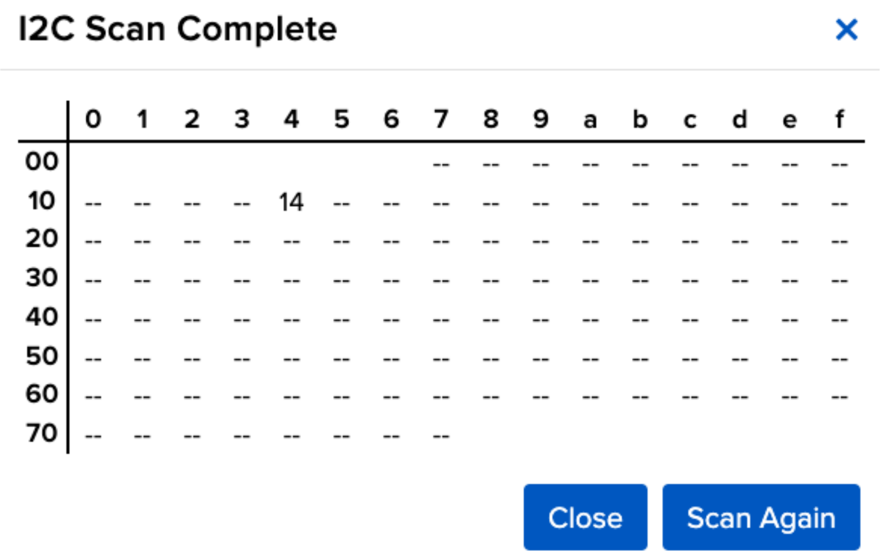
The device tile on the left indicates the version number of the firmware running on the connected board.

If the firmware version is green with a checkmark - continue with this guide.
If the firmware version is red with an "X" - [update to the latest WipperSnapper firmware (https://adafru.it/Vfd)](https://adafru.it/Vfd) on your board before continuing.

Next, make sure the sensor is plugged into your board and click the **I2C Scan** button.


adafruit
Devices
Feeds
Dashboards
Actions
Power-Ups
brubell / Devices / Adafruit Feather ESP32 V2
New Component
I2C Scan
Device Settings
+
Adafruit Feather ESP32...
Adafruit Feather ESP32 V2 by Adafruit

You should see the VL53L4CX's default I2C address of `0x14` pop-up in the I2C scan list.


I2C Scan Complete

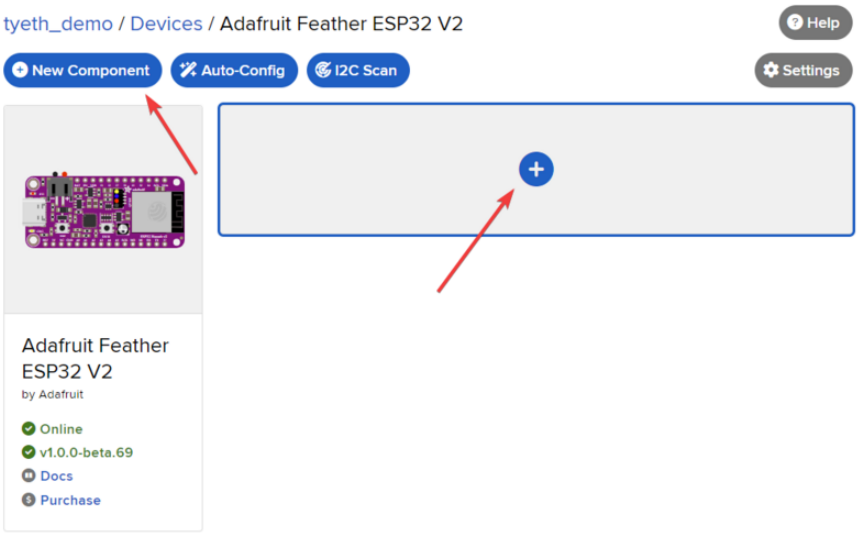
	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00								--	--	--	--	--	--	--	--	--
10	--	--	--	--	14	--	--	--	--	--	--	--	--	--	--	--
20	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
30	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
40	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
50	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
60	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
70	--	--	--	--	--	--	--	--								

I don't see the sensor's I2C address listed!

> First, double-check the connection and/or wiring between the sensor and the board.
>
> Then, reset the board and let it re-connect to Adafruit IO WipperSnapper.

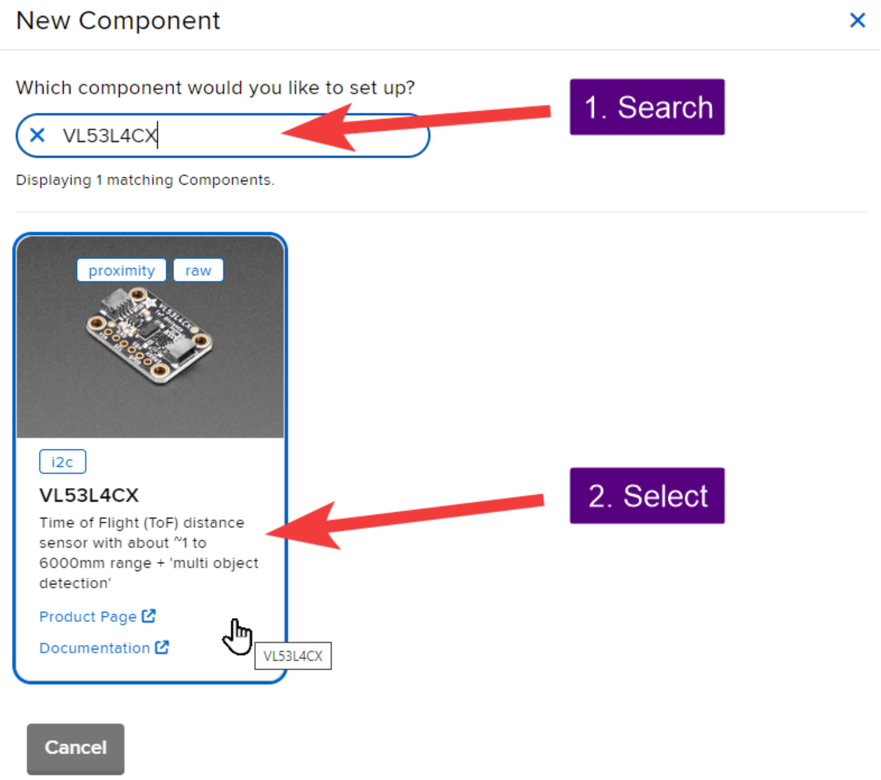
With the sensor detected in an I2C scan, you're ready to add the sensor to your board.

Click the New Component button or the + button to bring up the component picker.



tyeth_demo / Devices / Adafruit Feather ESP32 V2

Adafruit IO supports a large amount of components. To quickly find your sensor, type `VL53L4CX` into the search bar, then select the **VL53L4CX** component.



New Component

Which component would you like to set up?

VL53L4CX

Displaying 1 matching Components.

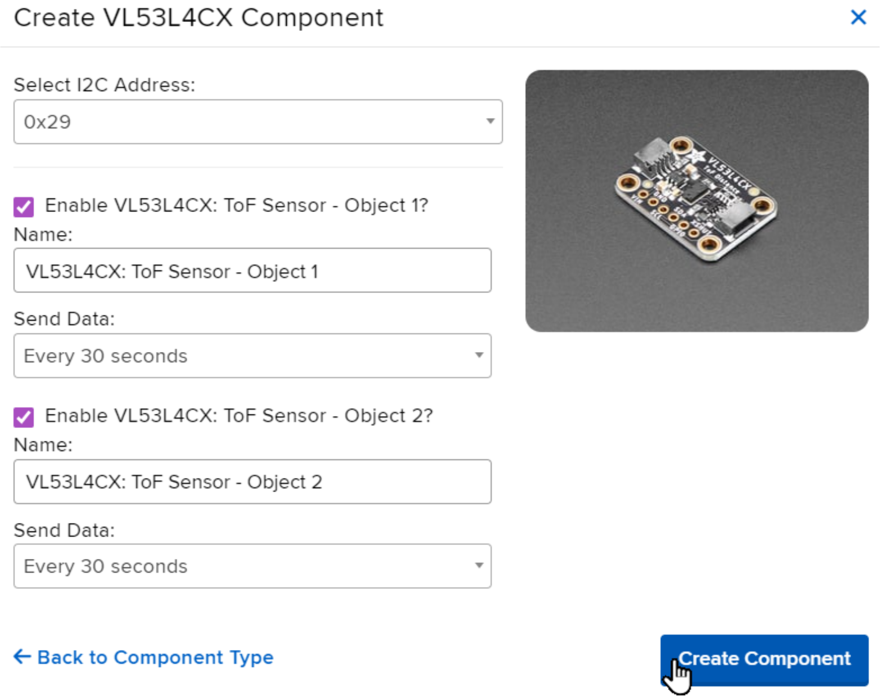
VL53L4CX

Time of Flight (ToF) distance sensor with about ~1 to 6000mm range + 'multi object detection'

On the component configuration page, the VL53L4CX's sensor address should be listed along with the sensor's settings.

The **Send Every** option is specific to each sensor's measurements. This option will tell the Feather how often it should read from the VL53L4CX sensor and send the data to Adafruit IO. Measurements can range from every 30 seconds to every 24 hours.

For this example, set the **Send Every** interval to every 30 seconds.



Create VL53L4CX Component

Select I2C Address: 0x29

☑ Enable VL53L4CX: ToF Sensor - Object 1?

Name: VL53L4CX: ToF Sensor - Object 1

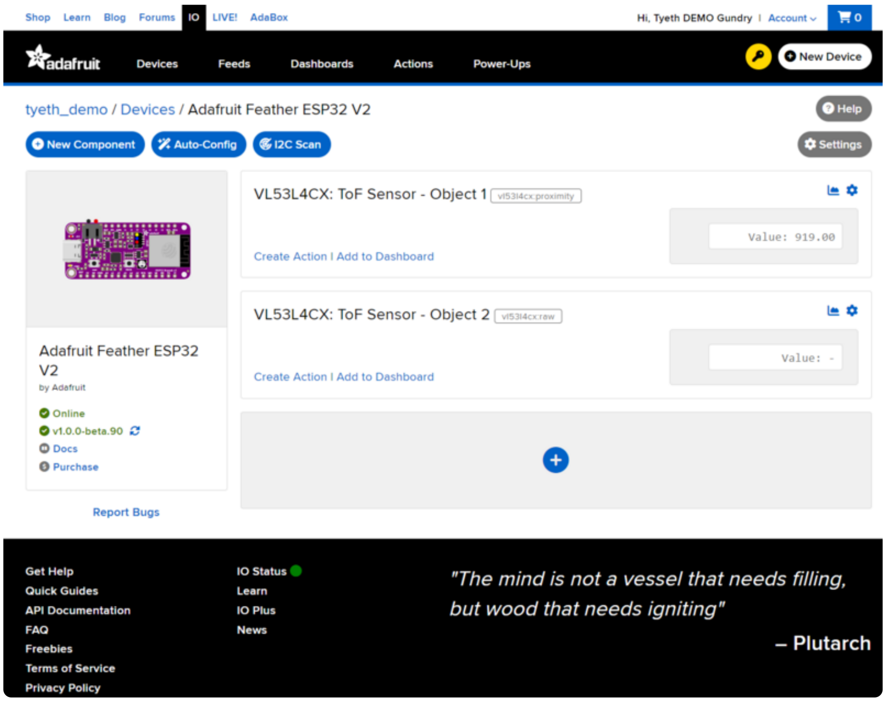
Send Data: Every 30 seconds

☑ Enable VL53L4CX: ToF Sensor - Object 2?

Name: VL53L4CX: ToF Sensor - Object 2

Send Data: Every 30 seconds

Your device interface should now show the sensor components you created. After the interval you configured elapses, WipperSnapper will automatically read values from the sensor(s) and send them to Adafruit IO.



tyeth_demo / Devices / Adafruit Feather ESP32 V2

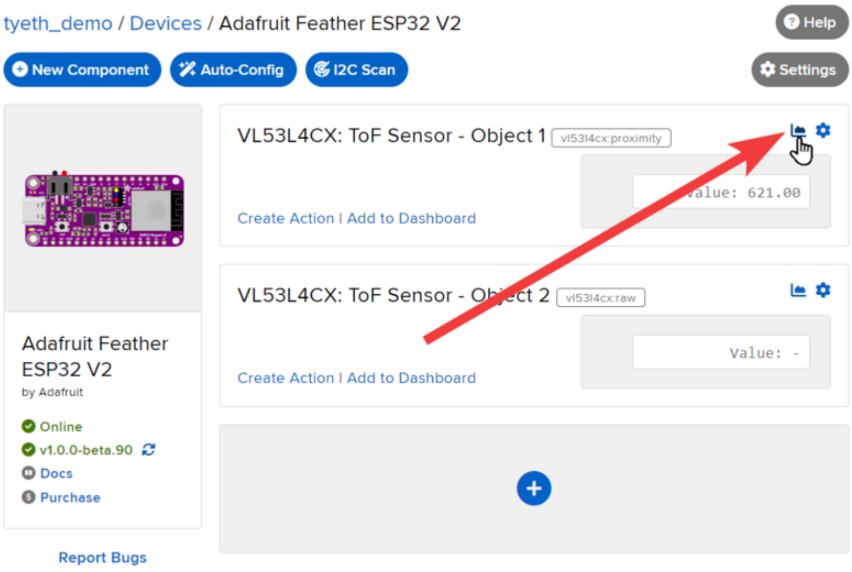
VL53L4CX: ToF Sensor - Object 1 vl53l4cx:proximity — Value: 919.00

VL53L4CX: ToF Sensor - Object 2 vl53l4cx:raw — Value: -

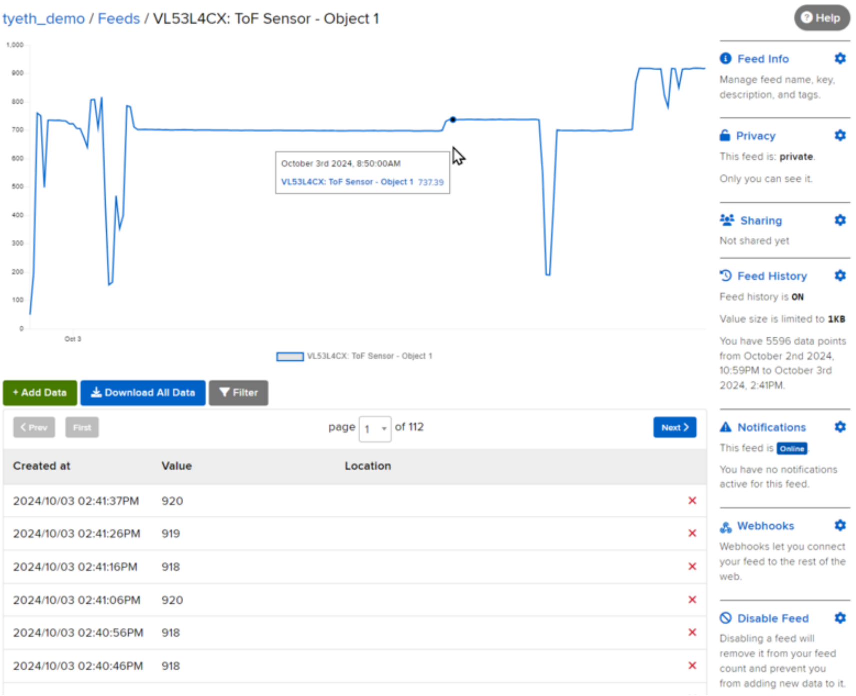
Adafruit Feather ESP32 V2
by Adafruit
Online
v1.0.0-beta.90

"The mind is not a vessel that needs filling, but wood that needs igniting"
– Plutarch

To view the data that has been logged from the sensor, click on the graph next to the sensor name.


tyeth_demo / Devices / Adafruit Feather ESP32 V2
Help
New Component
Auto-Config
I2C Scan
Settings
VL53L4CX: ToF Sensor - Object 1
vl53l4cx:proximity
Value: 621.00
Create Action | Add to Dashboard
VL53L4CX: ToF Sensor - Object 2
vl53l4cx:raw
Value: -
Create Action | Add to Dashboard
Adafruit Feather ESP32 V2
by Adafruit
Online
v1.0.0-beta.90
Docs
Purchase
Report Bugs

Here you can see the feed history and edit things about the feed such as the name, privacy, webhooks associated with the feed and more. If you want to learn more about how feeds work, check out this page (https://adafru.it/10aZ).


tyeth_demo / Feeds / VL53L4CX: ToF Sensor - Object 1
Help
1,000
900
800
700
600
500
400
300
200
100
0
Oct 3
October 3rd 2024, 8:50:00AM
VL53L4CX: ToF Sensor - Object 1 737.39
VL53L4CX: ToF Sensor - Object 1
+ Add Data
Download All Data
Filter
< Prev
First
page 1 of 112
Next >

Created at	Value	Location	
2024/10/03 02:41:37PM	920		×
2024/10/03 02:41:26PM	919		×
2024/10/03 02:41:16PM	918		×
2024/10/03 02:41:06PM	920		×
2024/10/03 02:40:56PM	918		×
2024/10/03 02:40:46PM	918		×

Feed Info

Manage feed name, key, description, and tags.

Privacy

This feed is: **private**.

Only you can see it.

Sharing

Not shared yet

Feed History

Feed history is **ON**

Value size is limited to **1KB**

You have 5596 data points from October 2nd 2024, 10:59PM to October 3rd 2024, 2:41PM.

Notifications

This feed is Online.

You have no notifications active for this feed.

Webhooks

Webhooks let you connect your feed to the rest of the web.

Disable Feed

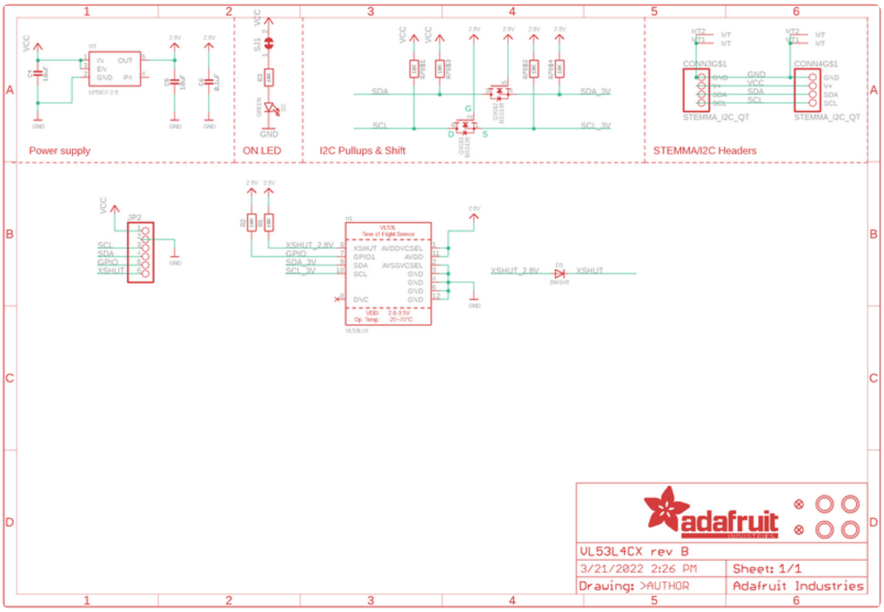
Disabling a feed will remove it from your feed count and prevent you from adding new data to it.

Downloads

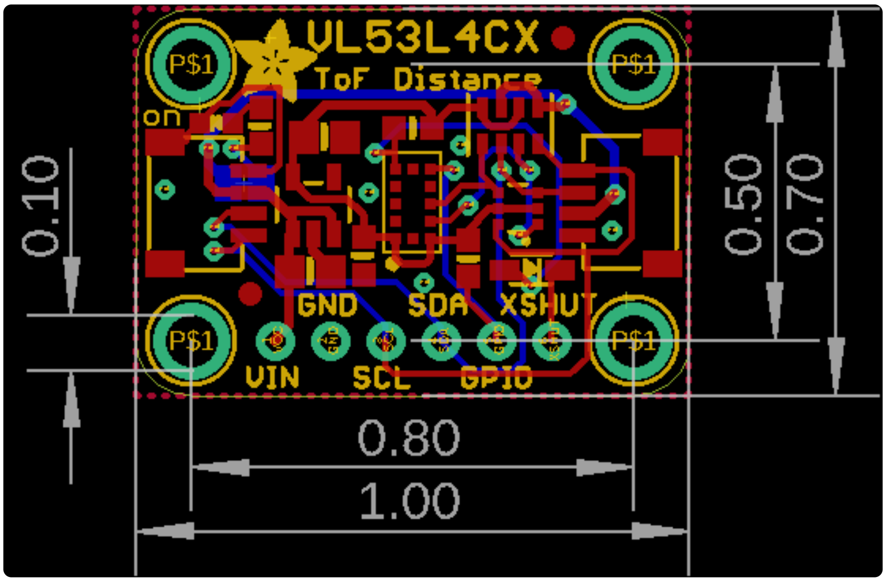
Files

- VL53L4CX Datasheet (https://adafru.it/ZZa)
- EagleCAD PCB files on GitHub (https://adafru.it/ZZb)
- Fritzing object in the Adafruit Fritzing Library (https://adafru.it/-1f)

Schematic and Fab Print



Power supply
ON LED
I2C Pullups & Shift
STEMMA/I2C Headers
VCC
SJ1
GND
SDA
SCL
SDA_3V
SCL_3V
CONN3GS1
CONN4GS1
STEMMA_I2C_QT
JP2
SCL
SDA
GPIO
XSHUT
XSHUT_2.8V
GPIO
SDA_3V
SCL_3V
XSHUT
XSHUT_2.8V
VL53L4CX rev B
3/21/2022 2:26 PM
Sheet: 1/1
Drawing: >AUTHOR
Adafruit Industries
adafruit



VL53L4CX
ToF Distance
on
P$1
P$1
P$1
P$1
GND
SDA
XSHUT
VIN
SCL
GPIO
0.10
0.50
0.70
0.80
1.00