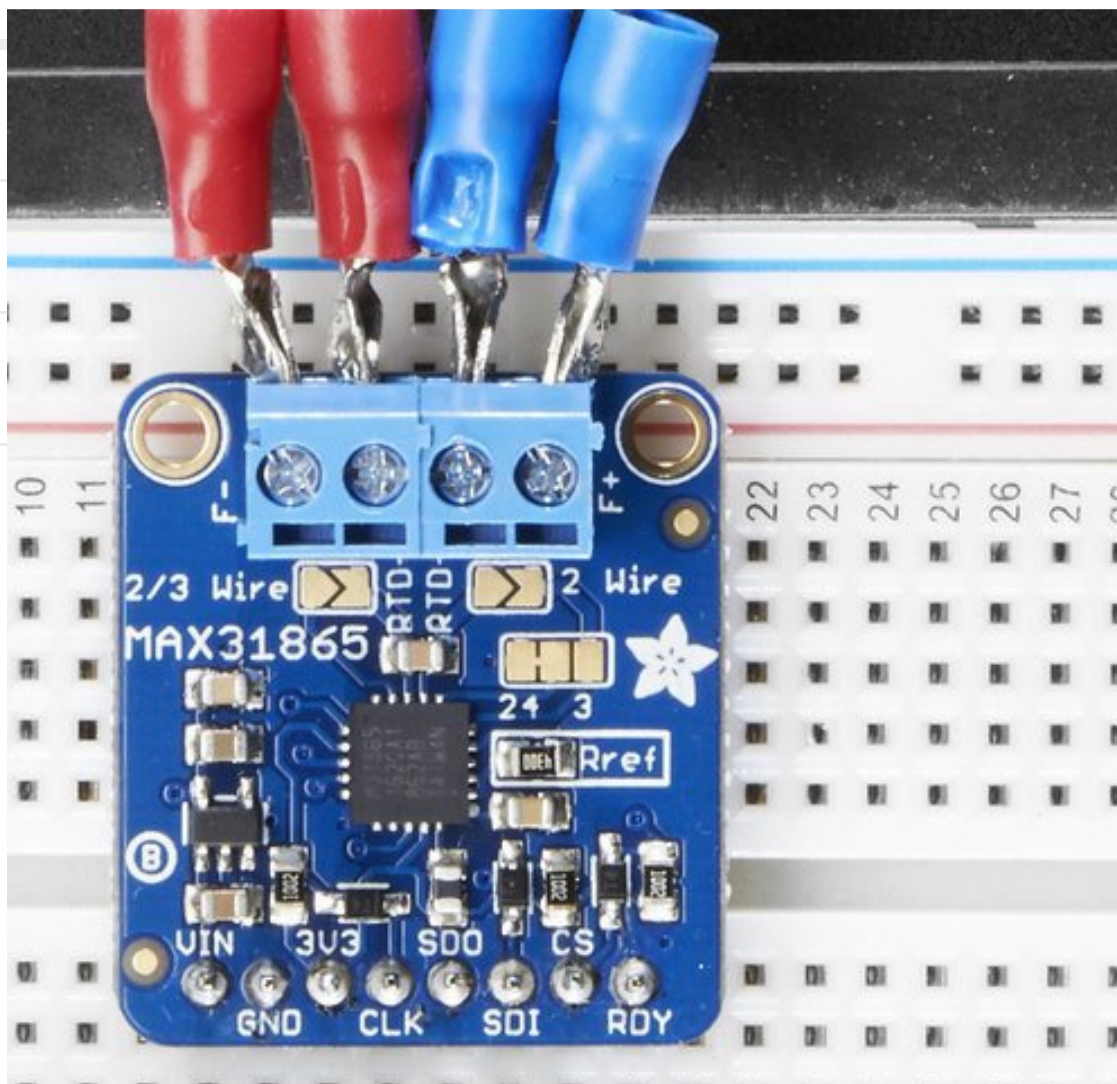




[865 RTD PT100 or PT1000 Amplifier](#) > [Python & CircuitPython](#)



MAX31865 RTD PT100 or PT1000

using'

[RTD Wiring & Config](#)

[Arduino Code](#)

▼ [Python & CircuitPython](#)

[Python Docs](#) 

[F.A.Q.](#)

[Downloads](#)

[Featured Products](#)

[Single page](#)

[Download PDF](#)

[Feedback? Corrections?](#)

Python & CircuitPython

 [Save](#)

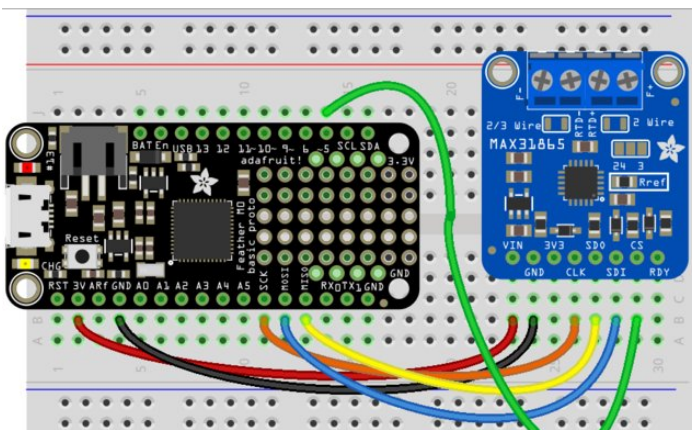
 [Subscribe](#)

It's easy to use the MAX31865 sensor with Python or CircuitPython, and the [Adafruit CircuitPython MAX31865](#) module. This module allows you to easily write Python code that reads the range from the sensor.

You can use this sensor with any CircuitPython microcontroller board or with a computer that has GPIO and Python [thanks to Adafruit Blinka, our CircuitPython-for-Python compatibility library](#).

CircuitPython Microcontroller Wiring

First wire up a MAX31865 to your board exactly as shown on the previous pages for Arduino. Here's an example of wiring a Feather M0 to the sensor with a SPI connection:

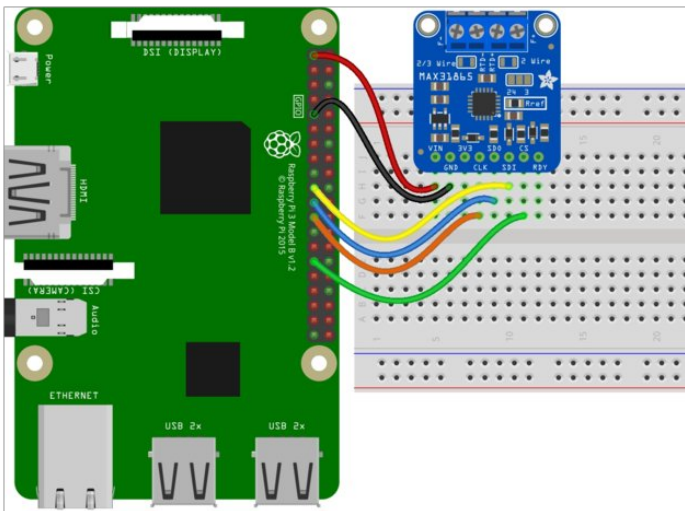


- **Board 3V** to sensor VIN
- **Board GND** to sensor GND
- **Board SCK** to sensor CLK
- **Board MOSI** to sensor SDI
- **Board MISO** to sensor SDO
- **Board D5** to sensor CS (or any other digital I/O pin)

Python Computer Wiring

Since there's *dozens* of Linux computers/boards you can use we will show wiring for Raspberry Pi. For other platforms, [please visit the guide for CircuitPython on Linux to see whether your platform is supported](#).

Here's the Raspberry Pi wired with SPI:



- Pi 3V3 to sensor VIN
- Pi GND to sensor GND
- Pi MOSI to sensor SDI
- Pi MISO to sensor SDO
- Pi SCLK to sensor CLK
- Pi GPIO5 to sensor CS (or use any other free GPIO pin)

CircuitPython Installation of MAX31865 Library

Next you'll need to install the [Adafruit CircuitPython MAX31865](#) library on your CircuitPython board.

First make sure you are running the [latest version of Adafruit CircuitPython](#) for your board.

Next you'll need to install the necessary libraries to use the hardware--carefully follow the steps to find and install these libraries from [Adafruit's CircuitPython library bundle](#). For example the Circuit Playground Express guide has [a great page on how to install the library bundle](#) for both express and non-express boards.

Remember for non-express boards like the Trinket M0, Gemma M0, and Feather/Metro M0 basic you'll need to manually install the necessary libraries from the bundle:

- `adafruit_max31865.mpy`
- `adafruit_bus_device`

Before continuing make sure your board's lib folder or root filesystem has the `adafruit_max31865.mpy`, and `adafruit_bus_device` files and folders copied over.

Next [connect to the board's serial REPL](#) so you are at the CircuitPython >>> prompt.

Python Installation of MAX31865 Library

You'll need to install the Adafruit_Blinka library that provides the CircuitPython support in Python. This may also require enabling I2C on your platform and verifying you are running Python 3. [Since each platform is a little different, and Linux changes often, please visit the CircuitPython on Linux guide to get your computer ready!](#)

Once that's done, from your command line run the following command:

- `sudo pip3 install adafruit-circuitpython-max31865`

If your default Python is version 3 you may need to run 'pip' instead. Just make sure you aren't trying to use CircuitPython on Python 2.x, it isn't supported!

CircuitPython & Python Usage

To demonstrate the usage of the sensor we'll initialize it and read the range and more from the board's Python REPL.

Run the following code to import the necessary modules and initialize the SPI connection with the sensor:

 Download File

 Copy Code

```
import board
import digitalio
import adafruit_max31865
spi = board.SPI()
cs = digitalio.DigitalInOut(board.D5) # Chip select of the MAX31865 board.
sensor = adafruit_max31865.MAX31865(spi, cs)
```

Notice you need to explicitly define the chip select digital I/O pin--be sure to use the same pin as your wiring (D5 if following this example exactly).

By default the MAX31865 class assumes a **2 wire** sensor, however you can change this by setting the **wires** keyword argument in the initializer. Set this to the number of wires in your sensor (2, 3, or 4). For example to create a 3 wire sensor:

[Download File](#)[Copy Code](#)

```
import board
import digitalio
import adafruit_max31865
spi = board.SPI()
cs = digitalio.DigitalInOut(board.D5) # Chip select of the MAX31865 board.
sensor = adafruit_max31865.MAX31865(spi, cs, wires=3)
```

Be sure to set **wires** to the appropriate value for your sensor!

In addition you can specify the nominal resistance and reference resistance of the RTD with these optional keyword arguments of the initializer:

- **rtd_nominal** - This is the resistance value in Ohms of the RTD at a nominal value (typically 0 degrees Celsius). This defaults to 100 for a PT100 sensor. **For a PT1000 change it to 1000 Ohms.**
- **ref_resistor** - The reference resistor value in Ohms. The default is 430 Ohms which matches the PT100 version of the breakout. **For a PT1000 breakout change this to 4300 Ohms.**

For example here's how to create a 2-wire PT1000 sensor instance with 1000 Ohm nominal resistance and 4300 Ohm reference resistance:

[Download File](#)[Copy Code](#)

```
import board
import digitalio
import adafruit_max31865
spi = board.SPI()
cs = digitalio.DigitalInOut(board.D5) # Chip select of the MAX31865 board.
sensor = adafruit_max31865.MAX31865(spi, cs, rtd_nominal=1000.0, ref_resistor=4300.0)
```

Now you're ready to read values from the sensor using any of these properties:

- **temperature** - The temperature measured by the sensor in degrees Celsius.
- **resistance** - The resistance of the RTD in Ohms.

 Download File

 Copy Code

```
print('Temperature: {0:0.3f}C'.format(sensor.temperature))
print('Resistance: {0:0.3f} Ohms'.format(sensor.resistance))
```

```
>>> print('Temperature: {0:0.3f}C'.format(sensor.temperature))
Temperature: 23.402C
>>> print('Resistance: {0:0.3f} Ohms'.format(sensor.resistance))
Resistance: 109.127 Ohms
>>> █
```

See the [simpletest.py example](#) for a complete demo of printing the range every second. Save this as **code.py** on the board and examine the REPL output to see the temperature printed every second.

That's all there is to using the MAX31865 with CircuitPython!

Full Example Code

 Download Project Bundle

 Copy Code

```
# SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
# SPDX-License-Identifier: MIT

# Simple demo of the MAX31865 thermocouple amplifier.
# Will print the temperature every second.
import time
import board
import digitalio
import adafruit_max31865

# Create sensor object, communicating over the board's default SPI bus
spi = board.SPI()
cs = digitalio.DigitalInOut(board.D5) # Chip select of the MAX31865 board.
sensor = adafruit_max31865.MAX31865(spi, cs)
# Note you can optionally provide the thermocouple RTD nominal, the reference
```



```
# resistance, and the number of wires for the sensor (2 the default, 3, or 4)
# with keyword args:
# sensor = adafruit_max31865.MAX31865(spi, cs, rtd_nominal=100, ref_resistor=430.0,
wires=2)

# Main loop to print the temperature every second.
while True:
    # Read temperature.
    temp = sensor.temperature
    # Print the value.
    print("Temperature: {0:0.3f}C".format(temp))
    # Delay for a second.
    time.sleep(1.0)
```

[View on GitHub](#)

[← Arduino Code](#)

[F.A.Q. >](#)

This guide was first published on Nov 15, 2016. It was last updated on Nov 15, 2016.

This page (Python & CircuitPython) was last updated on Sep 12, 2021.

Text editor powered by [tinymce](#).

...

Difficulty: Intermediate



Guide Type: Product



Contributors: [lady ada](#), [Tony DiCola](#), [Kattni Rembor](#), [Carter Nelson](#), [Danny Nosonowitz](#)

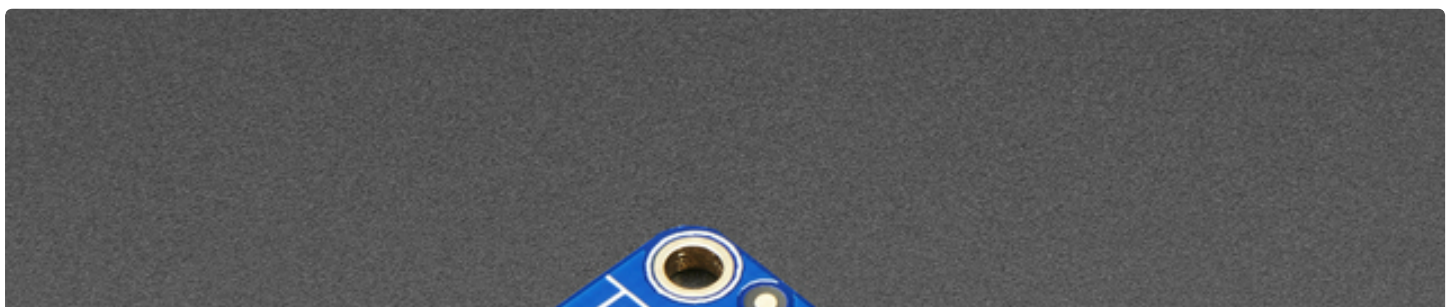


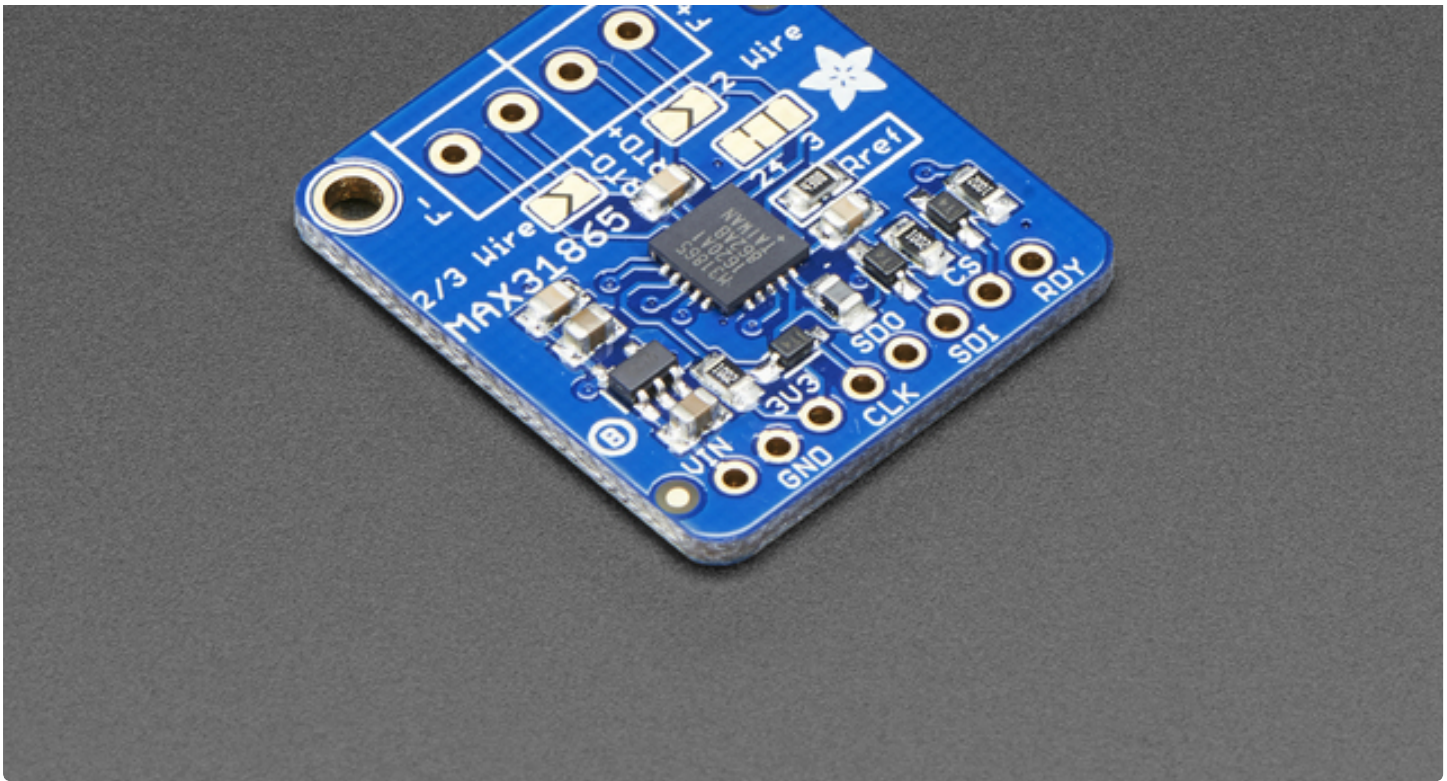
Categories: [Adafruit Products](#)
[Sensors/Temperature & Humidity](#)
[Breakout Boards/Amplifiers/Sound](#)



19 Saves

Featured Products

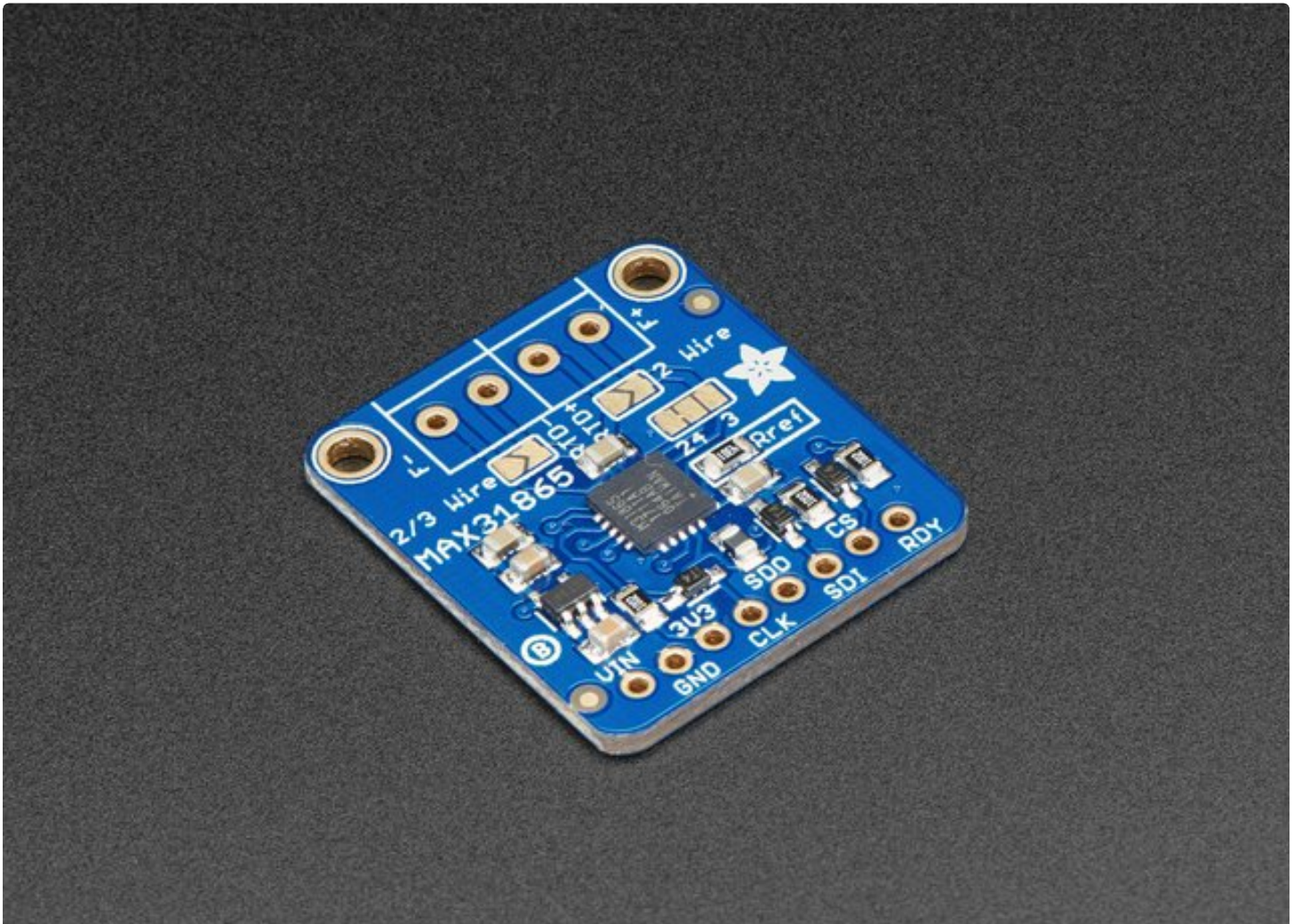




Adafruit PT100 RTD Temperature Sensor Amplifier - MAX31865

\$14.95

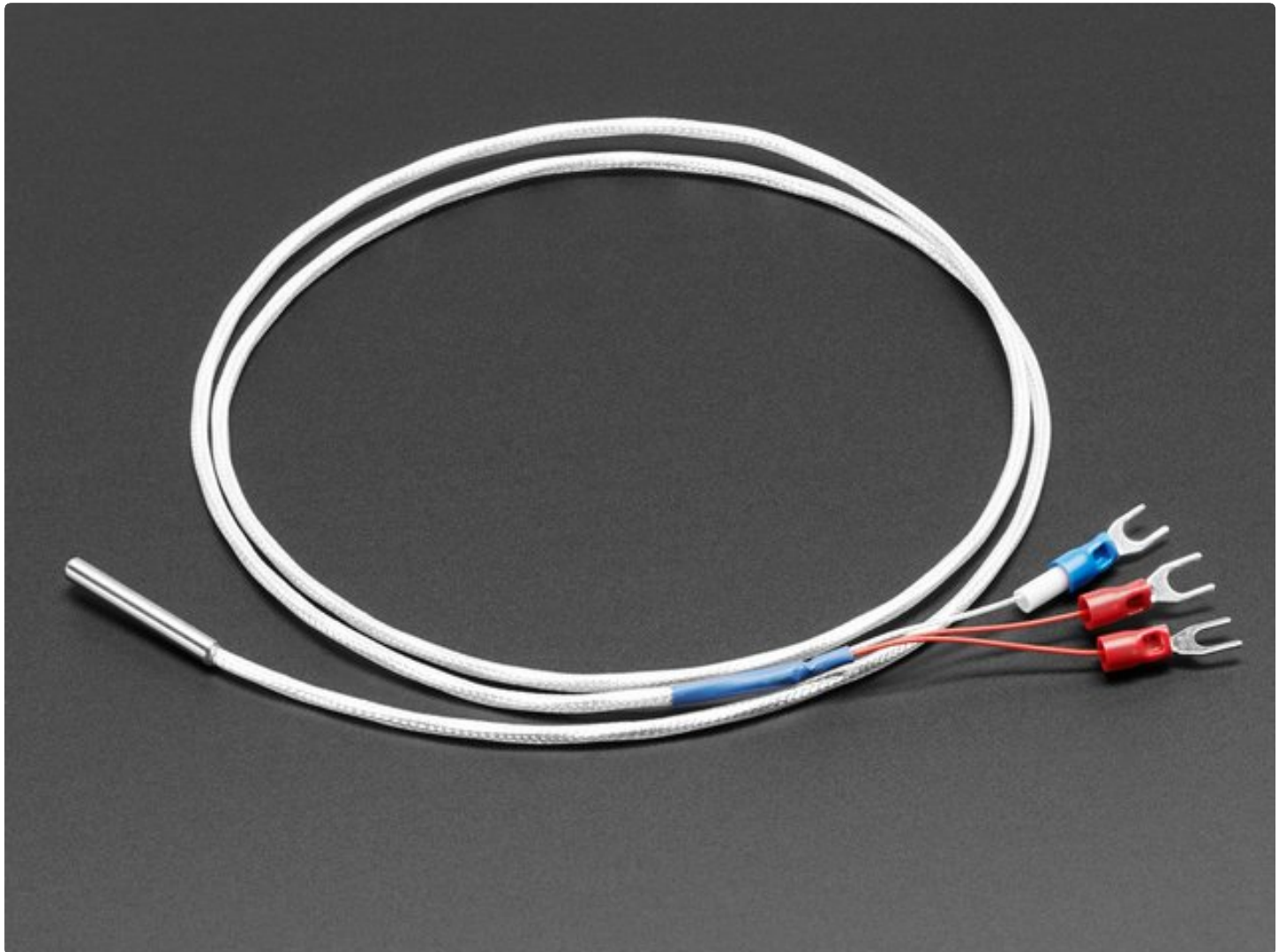
Add to Cart



Adafruit PT1000 RTD Temperature Sensor Amplifier - MAX31865

\$14.95

Add to Cart

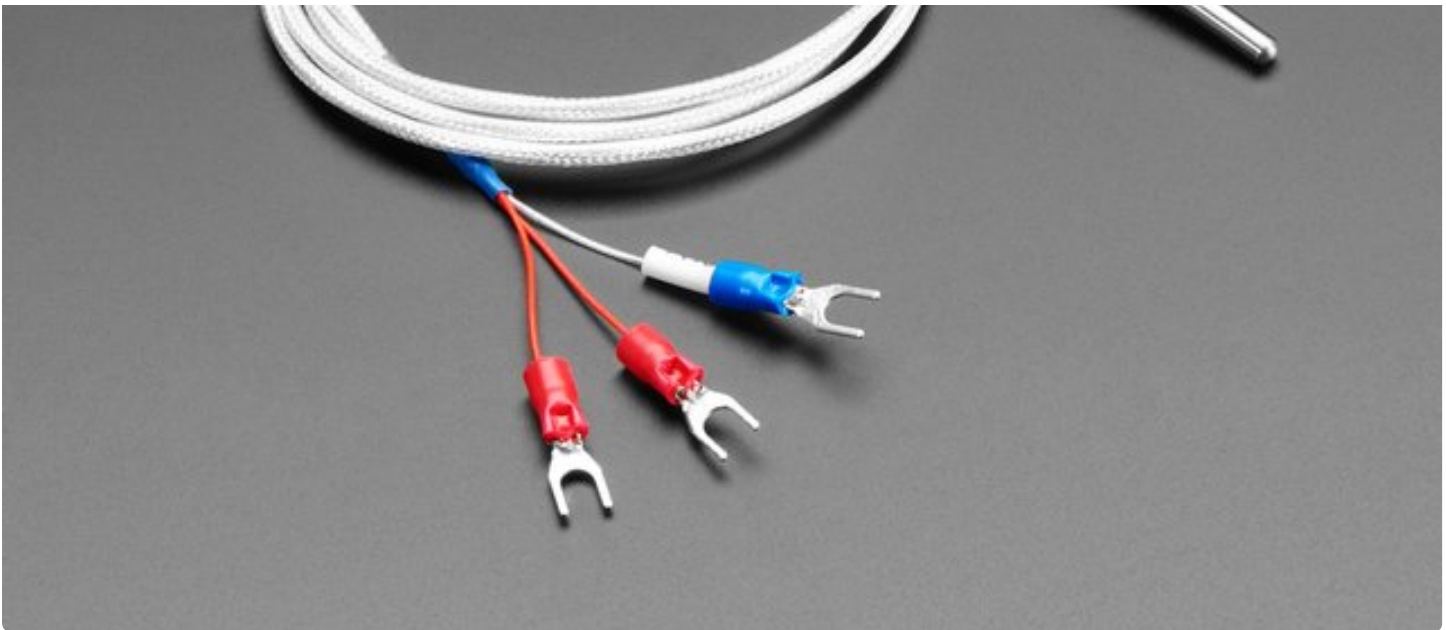


Platinum RTD Sensor - PT100 - 3 Wire 1 meter long

\$11.95

Add to Cart



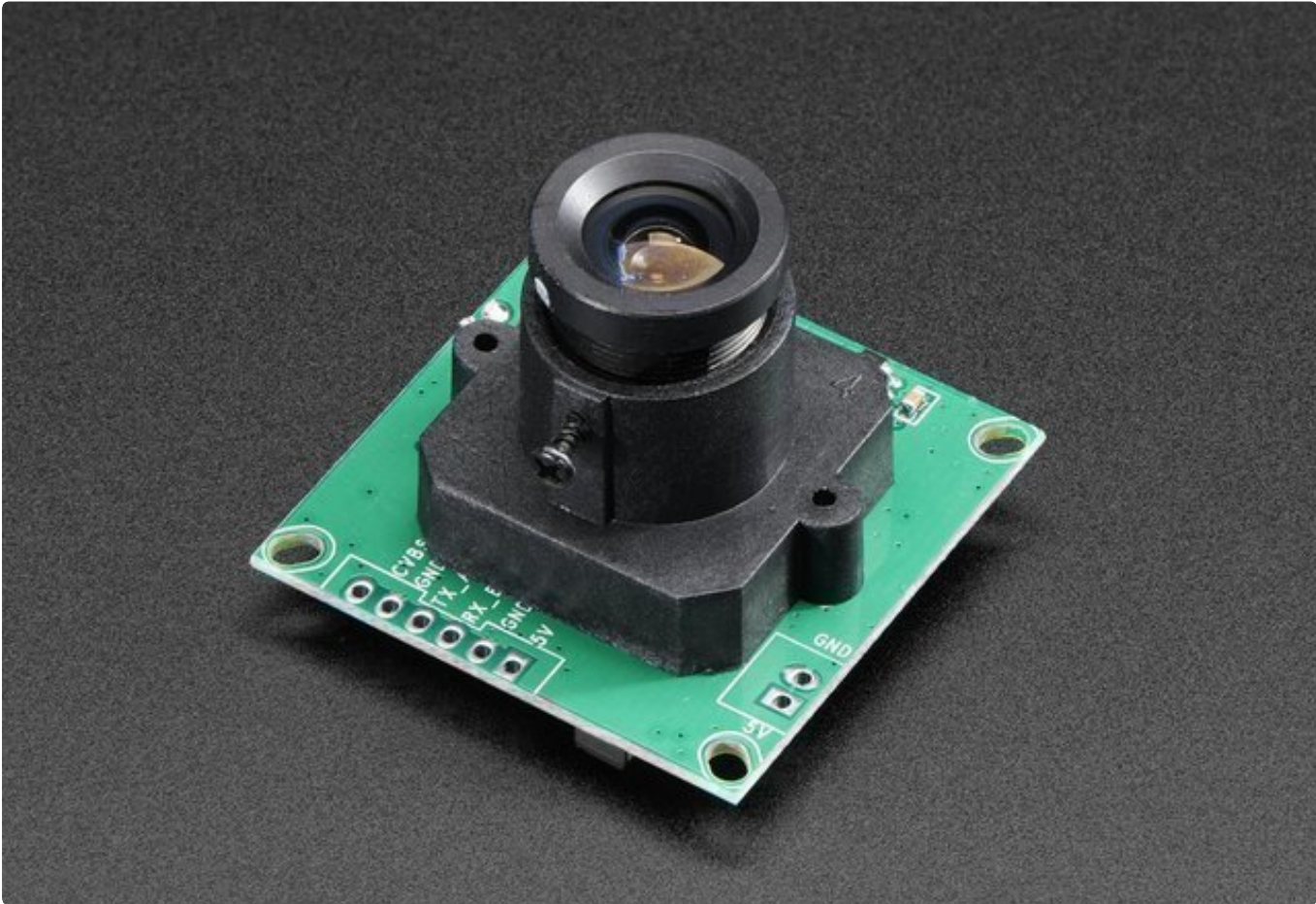


Platinum RTD Sensor - PT1000 - 3 Wire 1 meter long

\$14.95

Add to Cart

Related Guides

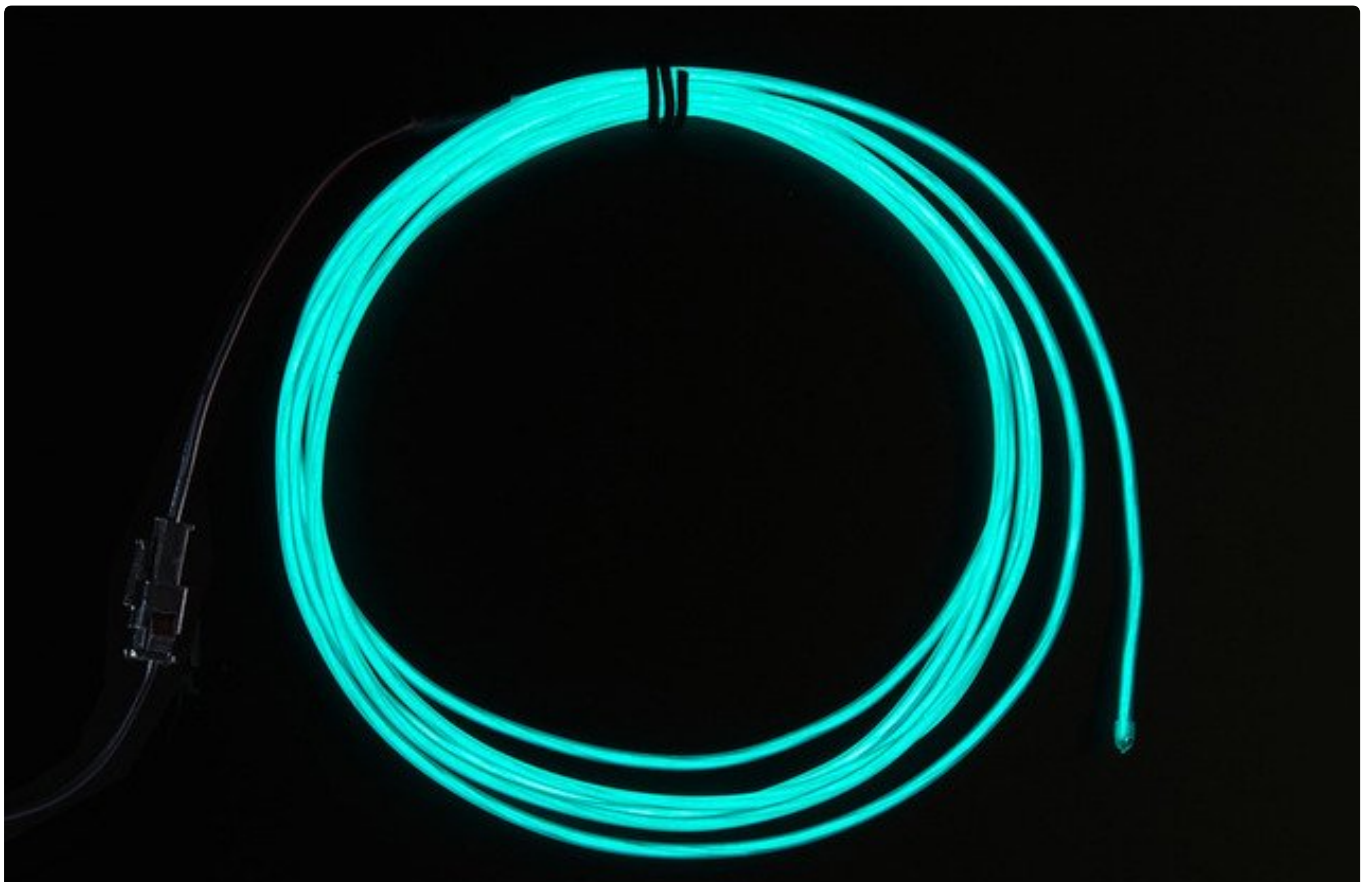


TTL Serial Camera

By [lady ada](#)

🔖 94

Beginner



EL Wire

By [lady ada](#)

🔖 83

Beginner

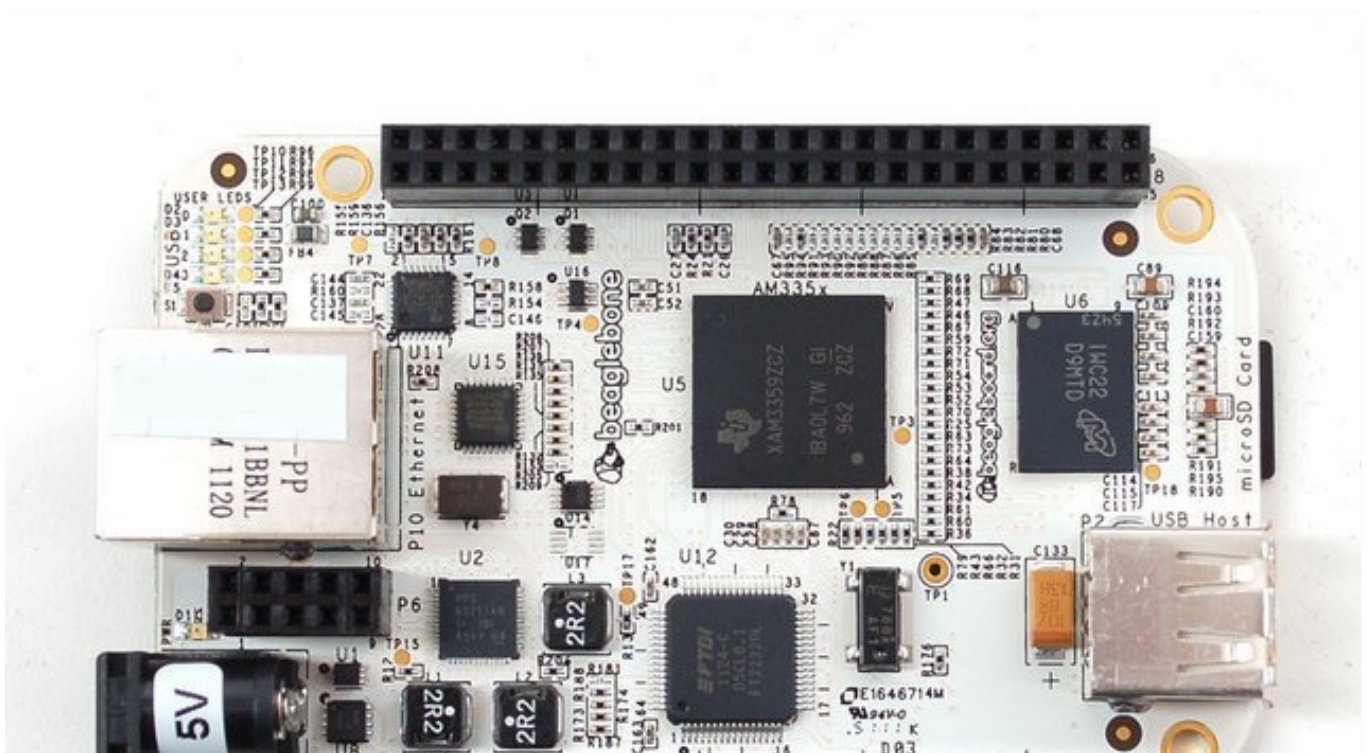


Hacking the Kinect

By lady ada

93

Intermediate



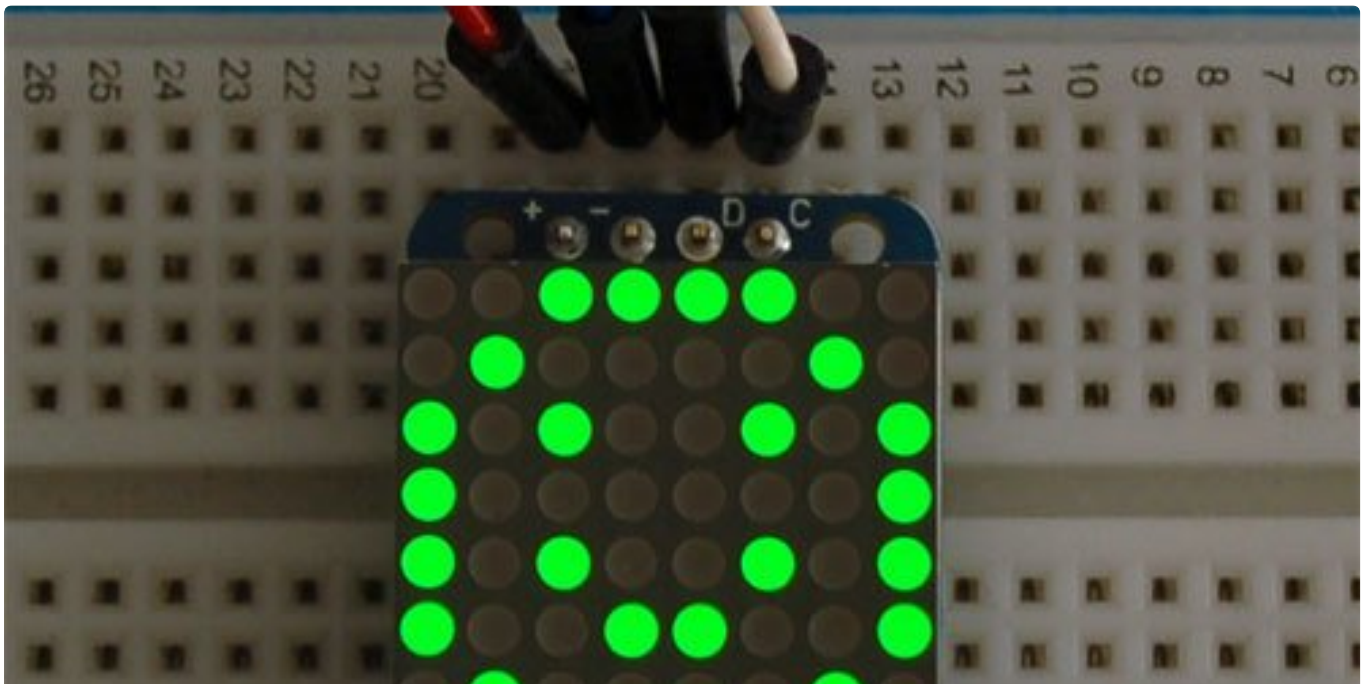


BeagleBone

By [lady ada](#)

🔖 15

Beginner



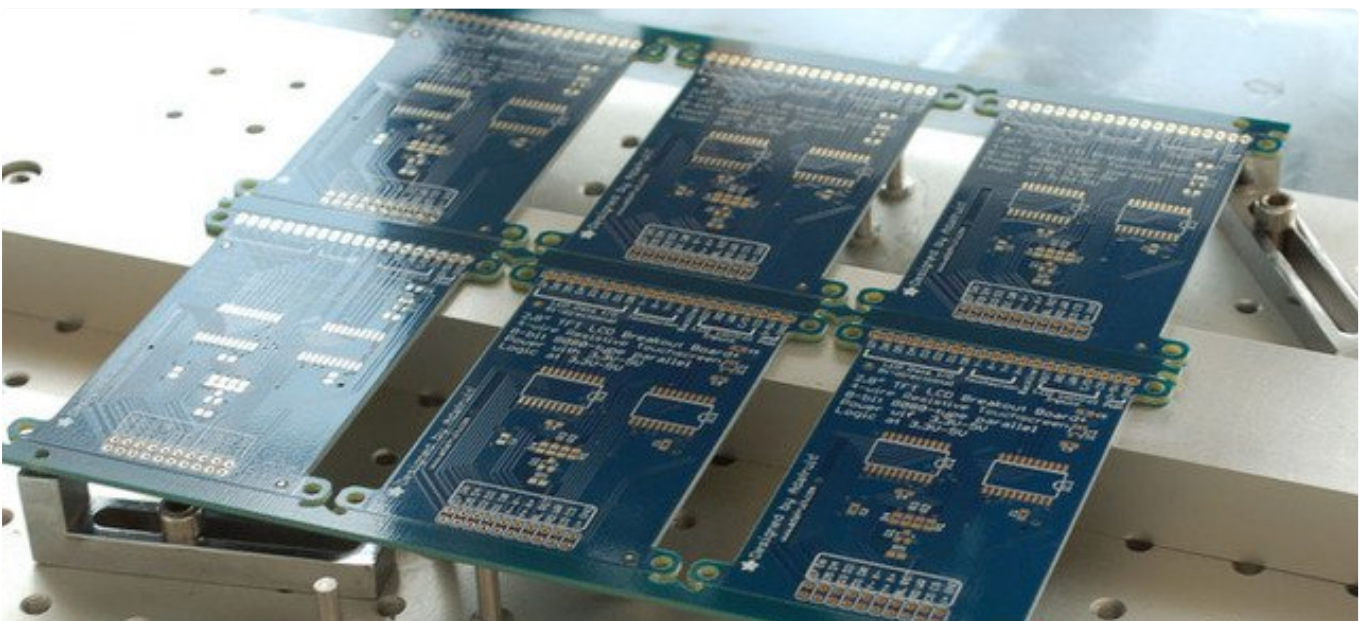


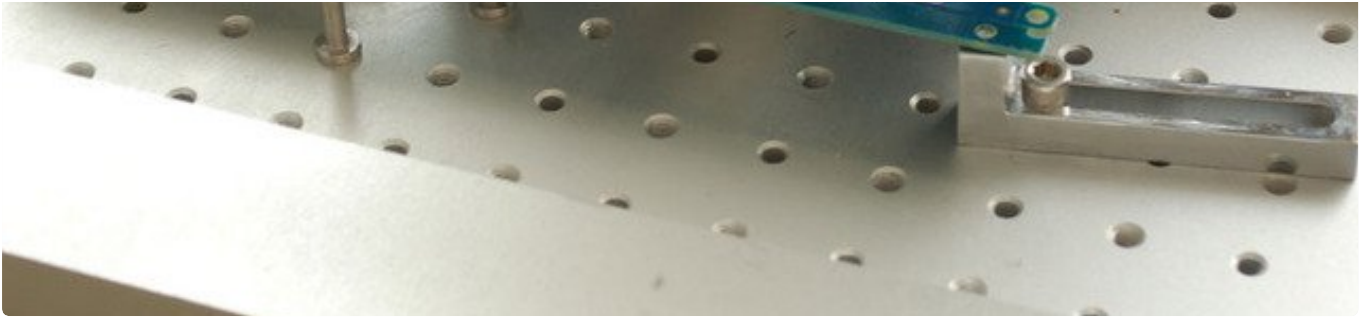
Adafruit LED Backpacks

By lady ada

132

Intermediate



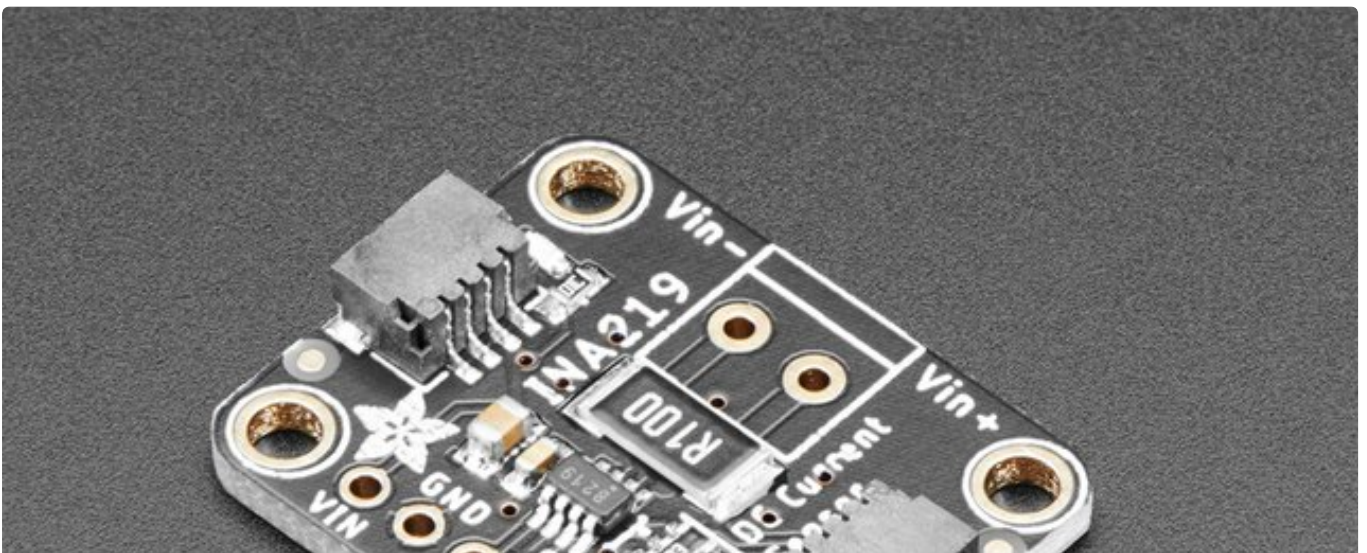


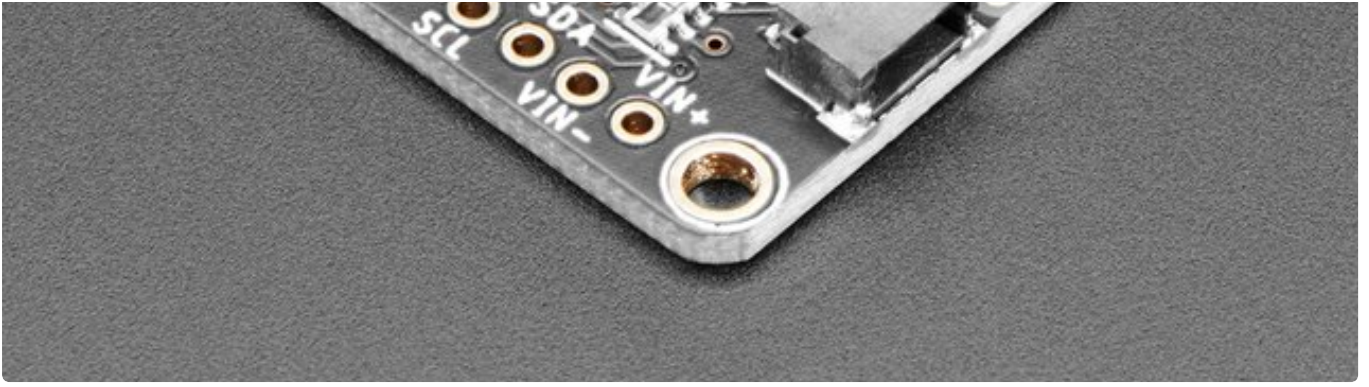
SMT Manufacturing

By lady ada

🔖 26

Intermediate





Adafruit INA219 Current Sensor Breakout

By [lady ada](#)

30

Beginner





MAX31855 Thermocouple

By [lady ada](#)

🔖 27

Intermediate

Contact Us

Tech Support Forums

FAQs

Shipping & Returns

Terms of Service

Privacy & Legal

About Us

Press

Educators

Distributors

Jobs

Gift Cards



A Minority and Woman-owned Business Enterprise (M/WBE)