

Adafruit AMG8833 8x8 Thermal Camera Sensor

Created by Justin Cooper

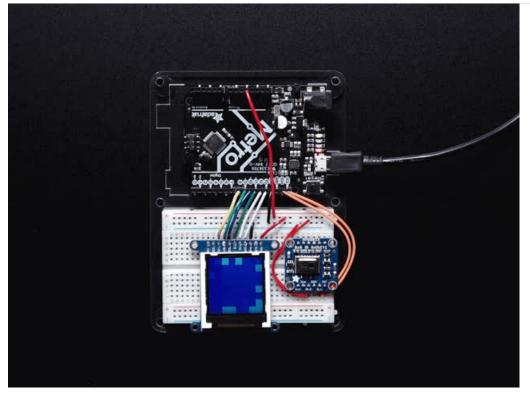


Last updated on 2018-01-17 05:57:08 PM UTC

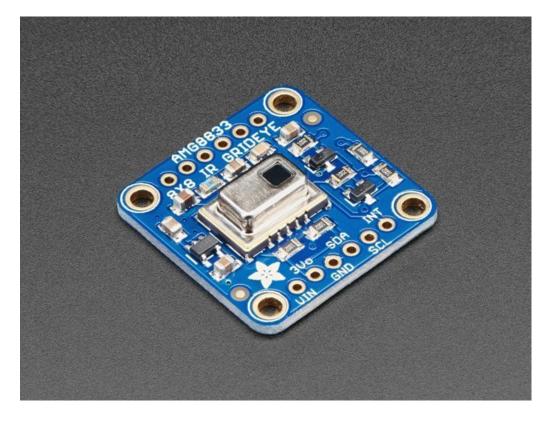
Guide Contents

Guide Contents	2
Overview	3
Pinouts	6
Power Pins:	6
Logic pins:	6
Assembly	8
Prepare the header strips:	8
Add the breakout board:	9
And Solder!	10
Arduino Wiring & Test	12
I2C Wiring	12
Download Adafruit_AMG88xx library	12
Load Thermistor Test	13
Pixel Array Output	14
Library Reference	15
Arduino Library Docs	16
Arduino Thermal Camera	17
Adafruit 1.44" Color TFT LCD Display with MicroSD Card breakout	17
CircuitPython Wiring & Test	19
I2C Wiring	19
Download Adafruit_CircuitPython_AMG88xx library	19
Raspberry Pi Thermal Camera	22
Raspberry Pi 3 - Model B - ARMv8 with 1G RAM	22
PiTFT Plus Assembled 320x240 2.8" TFT + Resistive Touchscreen	23
Assembled Pi T-Cobbler Plus - GPIO Breakout	23
Setup PiTFT	23
Install Python Software	23
Enable I2C	24
Wiring Up Sensor	25
Run example code	26
Downloads	28
Documents	28
Schematic	28
Dimensions	28

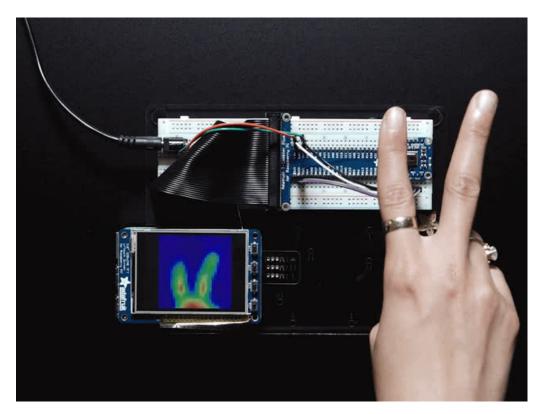
Overview

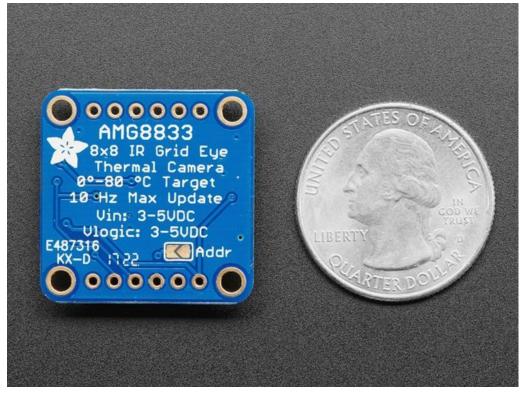


Add heat-vision to your project and with an Adafruit AMG8833 Grid-EYE Breakout! This sensor from Panasonic is an 8x8 array of IR thermal sensors. When connected to your microcontroller (or raspberry Pi) it will return an array of 64 individual infrared temperature readings over I2C. It's like those fancy thermal cameras, but compact and simple enough for easy integration.



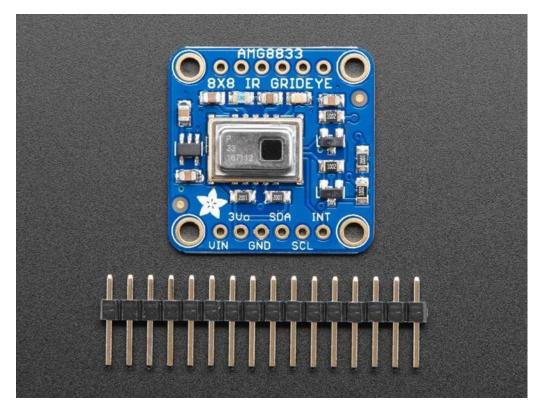
This part will measure temperatures ranging from 0°C to 80°C (32°F to 176°F) with an accuracy of +- 2.5°C (4.5°F). It can detect a human from a distance of up to 7 meters (23) feet. With a maximum frame rate of 10Hz, It's perfect for creating your own human detector or mini thermal camera. We have code for using this breakout on an Arduino or compatible (the sensor communicates over I2C) or on a Raspberry Pi with Python. On the Pi, with a bit of image processing help from the SciPy python library we were able to interpolate the 8x8 grid and get some pretty nice results!





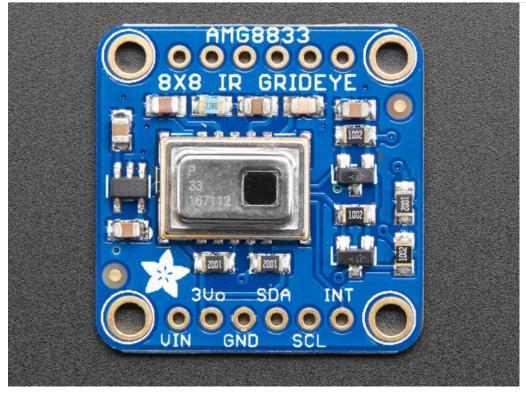
The AMG8833 is the next generation of 8x8 thermal IR sensors from Panasonic, and offers higher performance than it's predecessor the AMG8831. The sensor only supports I2C, and has a configurable interrupt pin that can fire when any individual pixel goes above or below a thresholds that you set.

To make it easy to use, we pick & placed it on a breakout board with a 3.3V regulator and level shifting. So you can use it with any 3V or 5V microcontroller or computer.



Even better - We've done all the hard work here, with example code and supporting software libraries to get you up in running in just a few lines of code!

Pinouts



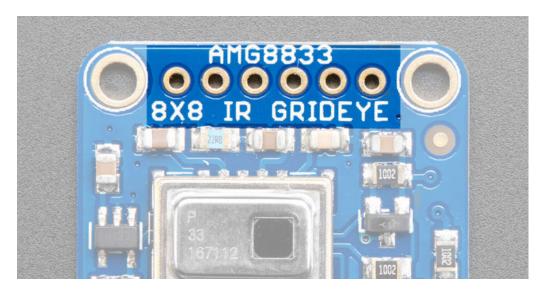
This camera has 4 mounting holes, and two header strips. Only the bottom strip is connected to the sensor. The top set of breakouts is there for mechanical stability only!

Power Pins:

- Vin this is the power pin. Since the sensor uses 3.3V, we have included an onboard voltage regulator that will take 3-5VDC and safely convert it down. To power the board, give it the same power as the logic level of your microcontroller e.g. for a 5V micro like Arduino, use 5V
- 3Vo this is the 3.3V output from the voltage regulator, you can grab up to 100mA from this if you like
- GND common ground for power and logic

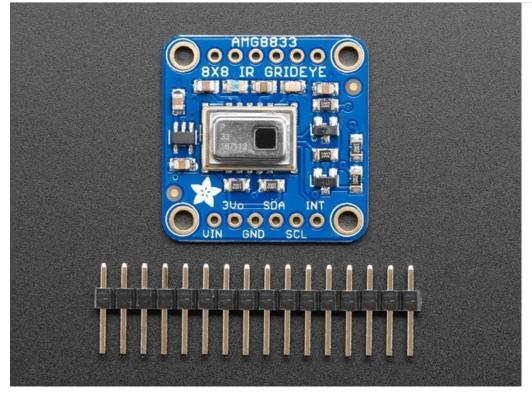
Logic pins:

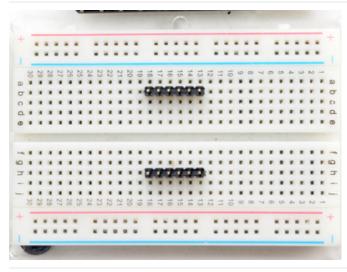
- SCL this is the I2C clock pin, connect to your microcontrollers I2C clock line. There is a 10K pullup on this pin and it is level shifted so you can use 3 5VDC.
- **SDA** this is the I2C data pin, connect to your microcontrollers I2C data line. There is a 10K pullup on this pin and it is level shifted so you can use 3 5VDC.
- INT this is the interrupt-output pin. It is 3V logic and you can use it to detect when something moves or changes in the sensor vision path.



The 6 holes at the top of the board are provided for stability and are not connected to anything. Use these if you want your sensor to sit nice and flat on a breadboard or Perma-Proto.

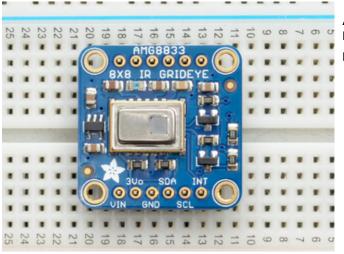
Assembly





Prepare the header strips:

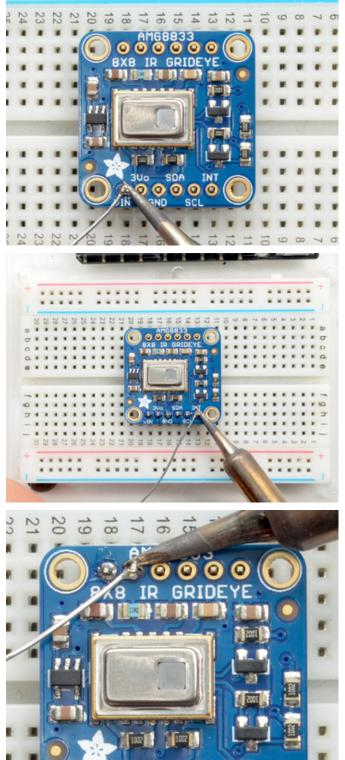
Cut the strips to length if necessary. It will be easier to solder if you insert it into a breadboard - **long pins down**



Add the breakout board:

Place the breakout board over the pins so that the short

pins poke through the breakout pads



And Solder!

Be sure to solder all pins for reliable electrical contact.

(For tips on soldering, be sure to check out our Guide to Excellent Soldering (https://adafru.it/aTk)).



You're done! Check your solder joints visually and continue onto the next steps

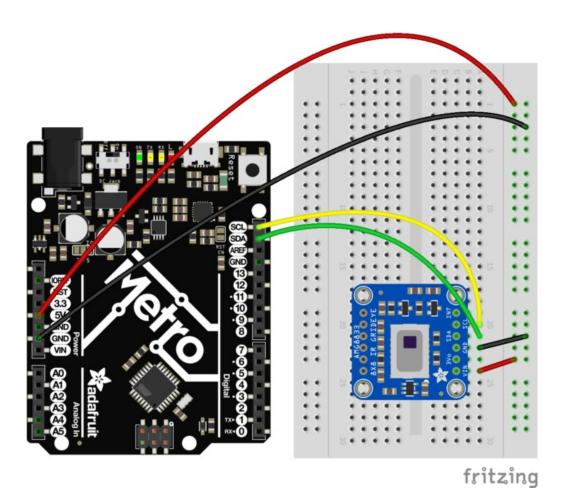
Arduino Wiring & Test

You can easily wire this breakout to any microcontroller, we'll be using an Arduino. You can use any other kind of microcontroller as well as long as it has I2C clock and I2C data lines.

I2C Wiring

- Connect Vin to the power supply, 3-5V is fine. Use the same voltage that the microcontroller logic is based off of. For most Arduinos, that is 5V
- Connect GND to common power/data ground
- Connect the SCL pin to the I2C clock SCL pin on your Arduino.
 On an UNO & '328 based Arduino, this is also known as A5, on a Mega it is also known as digital 21 and on a Leonardo/Micro, digital 3
- Connect the SDA pin to the I2C data SDA pin on your Arduino.
 On an UNO & '328 based Arduino, this is also known as A4, on a Mega it is also known as digital 20 and on a Leonardo/Micro, digital 2

By default, the I2C address is 0x69. If you solder the jumper on the back of the board labeled "Addr", the address will change to 0x68.



Download Adafruit_AMG88xx library

To begin reading sensor data, you will need to install the Adafruit_AMG88xx library.

Start up the IDE and open the Library Manager:

dama	Verify/Compile Upload	Ctrl+R Ctrl+U	~	
demo		Cuito	Manage Libraries	
	Upload Using Programmer Export compiled Binary	Ctrl+Shift+U Ctrl+Alt+S	Add .ZIP Library	rn
Hinclud	Show Sketch Folder Include Library	Ctrl+K	Arduino libraries ArduinoHttpClient	
#includ	Add File one pixel at a time,		ArduinoSound AudioZero	

Type in AMG88xx until you see the Adafruit Library pop up. Click Install!

Library Manag	er		
ype All		▼] amg88	
Arduino librar		it Version 1.0.0 INSTALLED rmocouple Sensors. Arduino library for AMG88xx (especially AMG8833) 8x8 Grid-e	ye thermal

We also have a great tutorial on Arduino library installation at: http://learn.adafruit.com/adafruit-all-about-arduino-libraries-install-use

Load Thermistor Test

Open up File->Examples->Adafruit_AMG88xx->amg88xx_test and upload to your Arduino wired up to the sensor. This example just connects to the sensor and reads the internal thermistor to test your connections.

e Edit Sketch New Open Open Recent	Ctrl+N Ctrl+O		
Sketchbook	I	**********************	*******
Examples			
Close	Ctrl+W	Adafruit_ADXL345_U Adafruit_AMG88xx	amg88xx_interrupt
Save	Ctrl+S	Adafruit_AS3935	amg88xx_test
Save As	Ctrl+Shift+S	Adafruit_BMP085	thermal_cam
Page Setup	Ctrl+Shift+P	Adafruit_BMP183	► Contraction
Print	Ctrl+P	Adafruit_BMP183_Unified	Þ
Preferences	Ctrl+Comma	Adafruit_CAP1188	▶ss is Ox69
0	Ctrl+0	Adafruit_CharacterOLED	► ce code,
Quit	UTT Additute	Adafruit_CompositeVideo	g products
from Adafru	it!	Adafruit_DRV2605	•
Written by	Dean Miller f	Adafruit_EPD	•
-	, all text ab	Adafruit_FastFloraPixel	bution
******	*****	Adafruit_FeatherOLED	*****************/

Once uploaded to your Arduino, open up the serial console at 9600 baud speed to see the internal thermistor reading. If you get a reading of $^{\sim}26^{\circ}$ degrees (room temperature) then everything is wired and working correctly!

💿 СОМЗ	-	- 🗆	×
			Send
AMG88xx test Thermistor Test			
Thermistor Temperature = 26.19 *C			
Thermistor Temperature = 26.19 *C			
Thermistor Temperature = 26.19 *C			
Thermistor Temperature = 26.19 *C			
Thermistor Temperature = 26.19 *C			
Thermistor Temperature = 26.19 *C			
Autoscroll	No line ending 🔍	9600 ba	aud 🗸

Pixel Array Output

OK now that we know the sensor is working, let's read actual thermal data. Load up File -> Examples -> Adafruit_AMG88 -> pixels_test

Upload the code, and open the serial console at 9600 baud rate. You should see a printout of the array of readings every second. Each number is the detected temperature in Celsius, and in the 8x8 grid order that comes from the sensor

The numbers should increase if you put your hand or face above the sensor. They'll decrease if you hold up something cold in front of the sensor eye

💿 COM21			_	
				Send
AMG88xx pixels Pixels Test				
25.75, 25.25, 26.00, 25.75, 29.00, 28.75, 27.00, 26.00, 29.50, 30.50, 31.25, 31.00,	25.50, 25.50, 27.00, 29.00, 26.50, 25.75, 25.00, 25.00, 25.75, 24.75, 27.25, 24.50,	, 25.25, 24.75 26.50, 26.00, 29.50, 28.00, 25.25, 25.00, 24.75, 24.50, 25.00, 25.00, 25.50, 25.00, 25.50, 25.75,	27.00, 26.00, 25.00, 25.00, 25.00, 25.25,	25.00, 25.25, 25.25, 25.50, 25.25, 26.75,
Autoscroll		Newline	~ 9	9600 baud 🗸 🗸

Library Reference

To create the object, use

Adafruit AMG88xx amg;

Initialize the sensor using

```
status = amg.begin();
    if (!status) {
        Serial.println("Could not find a valid AMG88xx sensor, check wiring!");
        while (1);
    }
```

to read the pixels you will need an array to place the readings into. Once you have one, you can call readPixels. Make sure the array you create is big enough by using the pre-defined AMG88xx_PIXEL_ARRAY_SIZE macro.

```
float pixels[AMG88xx_PIXEL_ARRAY_SIZE];
amg.readPixels(pixels);
```

Arduino Library Docs

Arduino Library Docs (https://adafru.it/Au6)

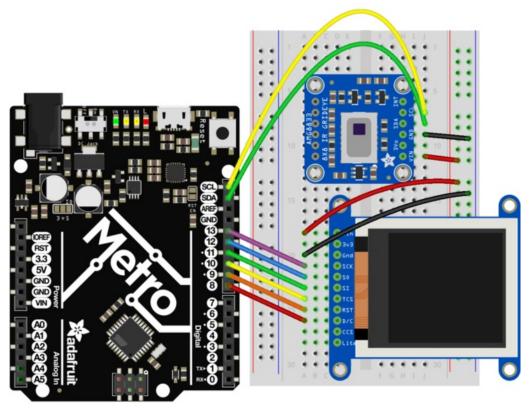
Arduino Thermal Camera

To make your Arduino into a cool thermal camera, we can add a small display.

In this example we use an Adafruit 1.44" Color TFT. With some code changes, you can use other size displays but a color display is best of course.



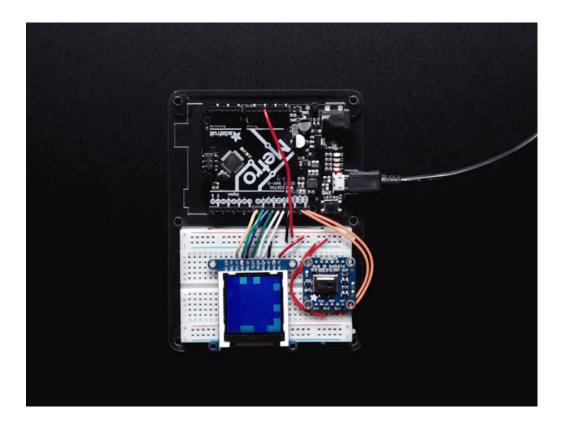
Keep your AMG8833 breakout wired as you already have it from the Wiring & Test section above, and add your TFT like this



fritzing

Once everything is all wired up, load up File->Examples->Adafruit_AMG88xx->thermal_cam

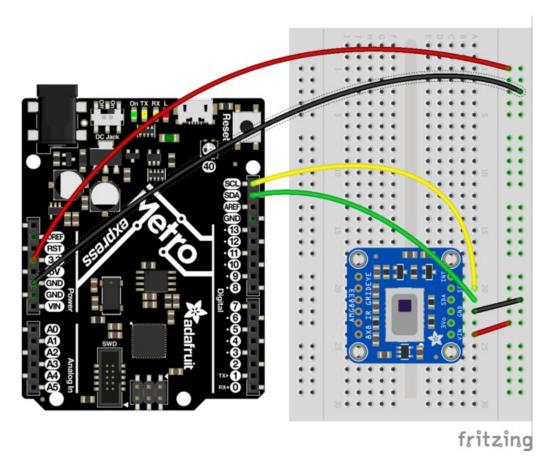
Hit upload and you should have a simple thermal camera!



CircuitPython Wiring & Test I2C Wiring

- Connect Vin to the power supply, 3-5V is fine.
- Connect **GND** to common power/data ground
- Connect the SCL pin to the I2C clock SCL pin on your Feather or Metro MO.
 On a Gemma MO this would be Pad #2/ A1
- Connect the SDA pin to the I2C data SDA pin on your Feather or Metro M0. On an Gemma M0 this would be Pad #0/A2

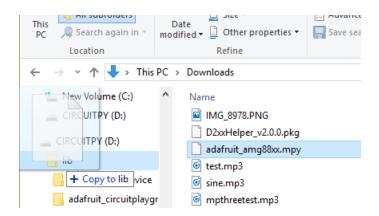
By default, the I2C address is 0x69. If you solder the jumper on the back of the board labeled "Addr", the address will change to 0x68.



Download Adafruit_CircuitPython_AMG88xx library

To begin reading sensor data, you will need to download Adafruit_CircuitPython_AMG88xx from our github repository. You can do that by downloading the most recent .mpy release from here: https://github.com/adafruit/Adafruit_CircuitPython_AMG88xx/releases

The plug in your CircuitPython device via the USB cable and drag the downloaded **adafruit_amg88xx.mpy** file to the /lib folder on the CIRCUITPY drive.



Then make a file named **code.py** in the root folder of your CIRCUITPY drive (if it is not already made) and copy and paste this code:

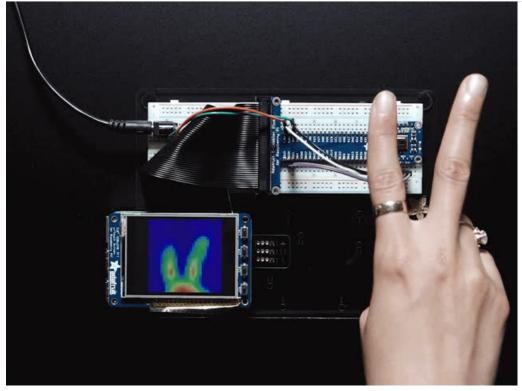
```
import time
import busio
import board
import adafruit_amg88xx
i2c_bus = busio.I2C(board.SCL, board.SDA)
amg = adafruit_amg88xx.AMG88XX(i2c_bus)
while True:
    for row in amg.pixels:
        #pad to 1 decimal place
        print(['{0:.1f}'.format(temp) for temp in row])
        print("\n")
        time.sleep(1)
```

save the file and open up your serial terminal. You should see the device print out an array of pixels every second.

💆 COM63 - Tera Term VT	_	×
<u>File Edit Setup Control Window H</u> elp		
['22.0', '22.5', '21.7', '22.0', '21.5', '21.5', '21.3', '22.0']		^
['22.0', '23.0', '21.7', '21.0', '21.0', '20.8', '21.0', '21.3']		
['23.0', '28.0', '27.5', '21.5', '21.3', '21.0', '21.0', '21.5']		
['24.2', '30.3', '29.5', '22.0', '20.8', '21.0', '21.0', '20.5']		
['26.5', '25.8', '25.3', '24.8', '24.5', '24.2', '25.0', '24.8']		
['24.0', '24.0', '23.8', '23.5', '24.2', '23.8', '23.5', '23.5']		
['22.8', '23.0', '22.5', '22.5', '22.0', '22.8', '22.5', '22.8']		
['27.3', '27.5', '27.3', '27.0', '27.5', '26.7', '26.7', '28.0']		
['22.5', '22.3', '21.7', '21.7', '21.5', '21.5', '21.5', '22.0']		
['22.0', '23.5', '23.0', '21.3', '21.0', '21.0', '20.8', '21.3']		
['23.3', '28.8', '28.0', '22.0', '21.3', '21.0', '21.3', '21.5']		
['24.8', '30.3', '30.0', '22.3', '21.0', '20.5', '20.8', '21.3']		

- -

Raspberry Pi Thermal Camera



The Raspberry Pi also has an i2c interface, and even better has processing capability to interpolate and filter the sensor output. By adding processing power, you can 'turn' the 8x8 output into what appears to be a higher-resolution display.

We're using a PiTFT 2.8" and a Pi Cobbler but the code can be adapted to output to the HDMI display - we're using pygame to draw to the framebuffer.

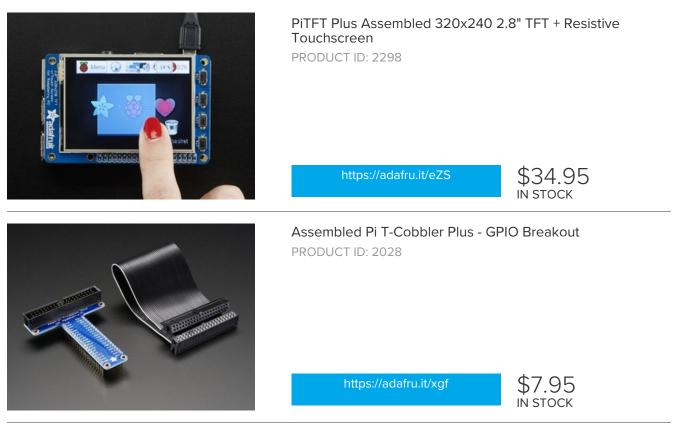
You can use any Raspberry Pi computer, from Pi A+ to Pi 3 or even a Pi Zero, but we happen to have a Pi 3 on our desk set up already so we're using that.



Raspberry Pi 3 - Model B - ARMv8 with 1G RAM PRODUCT ID: 3055

https://adafru.it/sc





Setup PiTFT

If you have not done so already, the first thing you will need to do is setup your PiTFT. Instructions on how to do so can be found in this guide.

Install Python Software

Once your PiTFT is all set up, and you have Internet access set up, lets install some software we will need. First make sure your Pi package manager is up to date.

sudo apt-get update

Next, we will install the Raspberry Pi library and Adafruit_GPIO which is our hardware interfacing layer

```
sudo apt-get install -y build-essential python-pip python-dev python-smbus git
git clone https://github.com/adafruit/Adafruit_Python_GPI0.git
cd Adafruit_Python_GPI0
sudo python setup.py install
```

Finally, install both pygame and scipy. Pygame lets us draw easily to a screen using python, we'll use that to make the display work. Scipy is a powerful scientific/data processing library that we can use to magically turn the 8x8 = 64 pixel array into something that looks more like a 32x32 = 1024 pixel array. Wow, isn't digital signal processing cool?

sudo apt-get install -y python-scipy python-pygame
sudo pip install colour Adafruit AMG88xx

Enable I2C

We need to enable the I2C bus so we can communicate with the sensor.

sudo raspi-config

Raspberry 1	Pi Software Configuration Tool (raspi-config)	
<pre>1 Expand Filesystem 2 Change Dser Password 3 Boot Options 4 Wait for Network at B 5 Internationalisation 6 Enable Camera 7 Add to Rastrack 8 Overclock</pre>	Choose whether to boot into a desktop Boot Choose whether to wait for network con	
9 Advanced Options 0 About raspi-config	Configure advanced settings Information about this configuration t	
o whome respi-contro	information about this configuration t	
(34	elect> <finish></finish>	
Raspberry P	Pi Software Configuration Tool (raspi-config)	_
Al Overscan	You may need to configure overscan if	
Al Nostname		1
A3 Nemory Split	Change the amount of memory made avail	
A4 SSR A5 VMC	Enable/Disable remote command line acc Enable/Disable graphical remote access	-
A6 SPI	Enable/Disable automatic loading of SP	Ξ.
A7 120 A8 Serial	Enable/Disable automatic loading of I. Enable/Disable shell and kernel messag	Ξ
A9 Audio	Force audio out through HDMI or 3.5mm	Ξ.
AA 1-Wire	Enable/Disable one-wire interface	1
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire		
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Inable/Disable one-wire interface</td><td></td></se<>	Inable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Enable/Disable one-wire interface</td><td></td></se<>	Enable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Inable/Disable one-wire interface</td><td></td></se<>	Inable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Inable/Disable one-wire interface</td><td></td></se<>	Inable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Inable/Disable one-wire interface</td><td></td></se<>	Inable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Inable/Disable one-wire interface</td><td></td></se<>	Inable/Disable one-wire interface	
AA 1-Wire <se< td=""><td>Inable/Disable one-wire interface</td><td></td></se<>	Inable/Disable one-wire interface	

select Advanced options, enable I2C, and then finish.

cot Options Choose whether to boot into a desktop ait for Network at Doot Choose whether to wait for network con nternationalisation Options Set up language and regional settings nable Camera Enable this Fi to work with the Raphery Fi dot to Rastrack Add this Fi to the colline Raphery Fi verelook Configure everclooking for your Fi	Expand Filesystem	Ensures that all of the SD card storag
ait for Network at Boot Choose whether to wait for network connermationalisation Options Set up language and regional settings nable Camera Enable this Fi to work with the Raspbe dd to Rastrack Configure overclocking for your Fi dwanced Options Configure advanced settings Dout raspi-config Information about this configuration t	Change User Password	
nternationalisation Options Set up language and regional settings nable Camera Enable this Fi to work with the Raspberry Fi det to Rastrack Add this Fi to the online Raspberry Fi verelook Configure overclocking for your Fi dwanced Options Configure advanced settings bout raspi-config Information about this configuration t		
nable Camera Enable this Pi to work with the Rasphe dit to Rastrack Add this Pi to the online Raspherry Pi werclock Configure overclocking for your Pi dwamced Options Configure advanced settings hour raspi-config Information about this configuration t		
dd to Rastrack Add this Pi to the online Raspberry Pi verclock Configure overclocking for your Pi dvanced Options Configure advanced settings bout raspi-config Information about this configuration t		
verclock Configure overclocking for your Fi dvanced Options Configure advanced settings hour rampi-config Information about this configuration t		
dvanced Options Configure advanced settings bout raspi-config Information about this configuration t	Overclock	
bout raspi-config Information about this configuration t	Advanced Options	
(Select) STITION	About raspi-config	
	(Select)	Summerican
	<pre><3elecc></pre>	and the second s

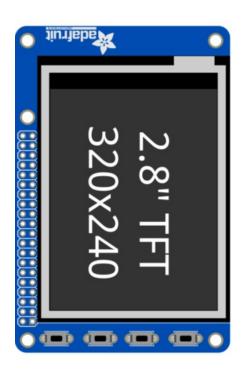
Once I2C is enabled, run sudo shutdown -h now to turn off the Pi and prepare for wiring

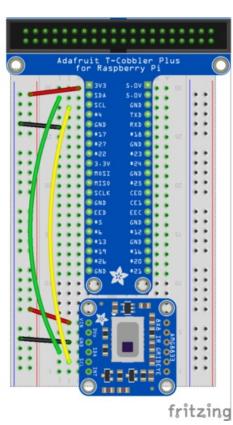
Wiring Up Sensor

With the Pi powered off, we can wire up the sensor to the Pi Cobbler like this:

- Connect Vin to the 3V or 5V power supply (either is fine)
- Connect **GND** to the ground pin on the Cobbler
- Connect SDA to SDA on the Cobbler
- Connect SCL to SCL on the Cobbler

You can also use direct wires, we happen to have a Cobbler ready. remember you can plug the cobbler into the bottom of the PiTFT to get access to all the pins!





Now you should be able to verify that the sensor is wired up correctly by asking the Pi to detect what addresses it can see on the I2C bus:

sudo i2cdetect -y 1

i@(sude											
		1	2	3	4	5	6	7	8	9	а	b	С	d	e	f
00:																
10:																
30:																
40:																
50:																
60:										69						
70:																
pi@	lear	ispi	i:~	Ş												

It should show up under it's default address (0x69). If you don't see 69, check your wiring, did you install I2C support, etc?

Run example code

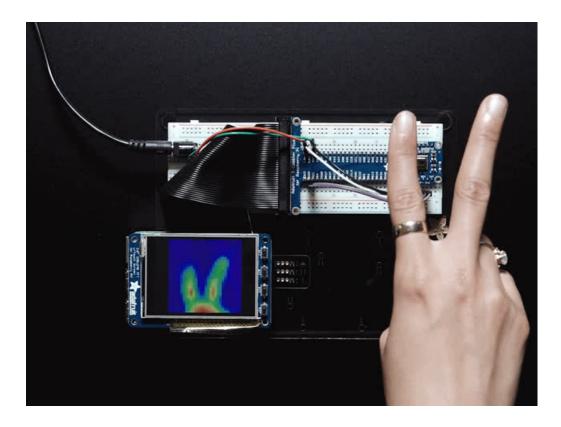
At long last, we are finally ready to run our example code

```
cd ~/
git clone https://github.com/adafruit/Adafruit_AMG88xx_python.git
cd Adafruit_AMG88xx_python/examples
sudo python thermal_cam.py
```

If you have everything installed and wired up correctly, you should see a nice thermal camera image. Cool tones (blue and purple) are cooler temperatures, and warmer tones (yellow, red) are warmer temperatures.

If your image seems to be flipped on the screen, try changing the orientation of the AMG8833 breakout on the breadboard.

If you're interested int he details, and want to know more about how we made 64 pixels look like many more, it's called bicubic interpolation (hat tip to OSHpark for the idea!)



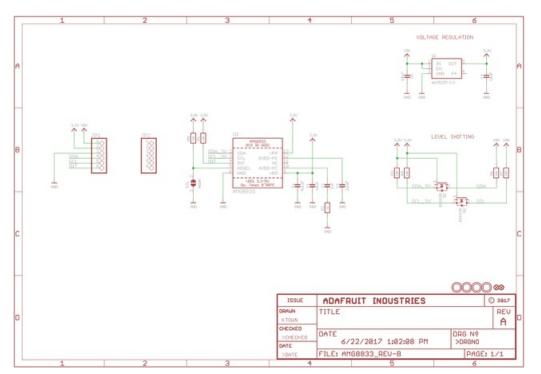
Downloads

Documents

- AMG8833 datasheet
- AMG8833 Arduino Driver
- Fritzing object in the Adafruit Fritzing library
- AMG8833 python driver
- AMG8833 CircuitPython Driver
- AMG8833 breakout PCB files (EAGLE format)

Schematic

click to enlarge



Dimensions

in inches. Click to enlarge

