

[SHOP](#)[LEARN](#)[BLOG](#)[SERVICES](#)

MicroMod STM32 Processor Hookup Guide

CONTRIBUTORS:  *ALEX THE GIANT, ELL C*

♡ FAVORITE

0

Introduction

We've brought the power and precision of the STM32 Processor to the MicroMod ecosystem! Please welcome the MicroMod STM32 Processor Board! With the high-performance Arm® Cortex®-M4 32-bit RISC core, Flash memory up to 1 Mbyte, up to 192 Kbytes of SRAM, a memory protection unit (MPU), high-speed embedded memories, up to 4 Kbytes of backup SRAM, and an extensive range of enhanced I/Os and peripherals, this board is ready to rock your MicroMod world. Let's dive in!



SparkFun MicroMod STM32 Processor

● DEV-17713








\$16.50

Product Showcase: SparkFun STM32 Processor Boards



Required Materials

To follow along with this tutorial, you will need the following materials. 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.

STM32 MicroMod Processor Board Wish List <small>SparkFun Wish List</small>	
	SparkFun MicroMod STM32 Processor DEV-17713
	SparkFun Serial Basic Breakout - CH340G DEV-14050 The SparkFun Serial Basic Breakout is an easy-to-use USB-to-Serial adapter based on the CH340G IC from WCH....
	SparkFun MicroMod ATP Carrier Board DEV-16885
	Qwiic Cable - 100mm PRT-14427
	SparkFun High Precision Temperature Sensor - TMP117 (Qwiic) SEN-15805
	(2) USB 3.1 Cable A to C - 3 Foot CAB-14743
	Jumper Wires Standard 7" M/M - 30 AWG (30 Pack) PRT-11026 If you need to knock up a quick prototype there's nothing like having a pile of jumper wires to speed things up, and ...

Suggested Reading

The SparkFun MicroMod ecosystem offers a unique way to allow users to customize their project to their needs. Do you want to send your weather data via a wireless signal (e.g. Bluetooth or WiFi)? There's a MicroMod Processor Board for that. Looking to instead maximize efficiency and processing power? You guessed it, there's a MicroMod Processor Board for that. If you are not familiar with the MicroMod ecosystem, take a look here:

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

MicroMod

MicroMod Ecosystem

We also recommend reading through the following tutorials if you are not familiar with the concepts covered in them:



Serial Communication

Asynchronous serial communication concepts: packets, signal levels, baud rates, UARTs and more!

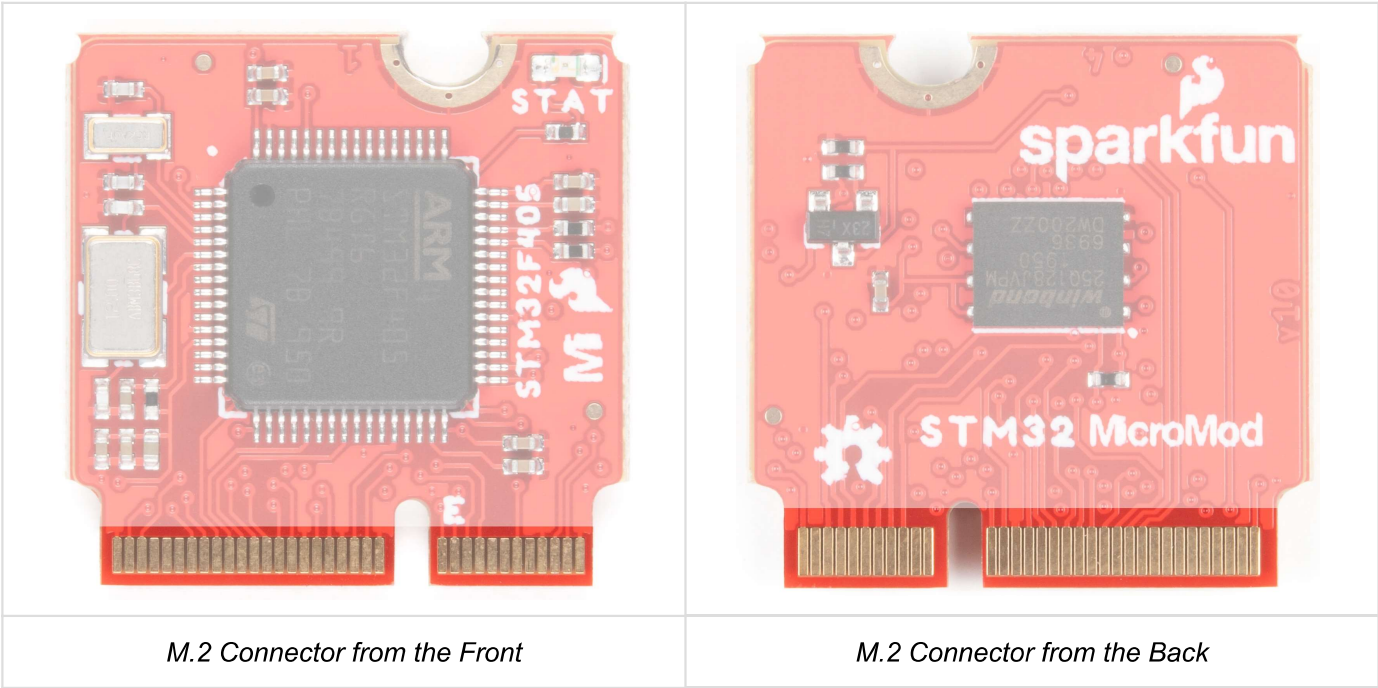
Getting Started with MicroMod

Dive into the world of MicroMod - a compact interface to connect a microcontroller to various peripherals via the M.2 Connector!

Hardware Overview

M.2 Connector

All of our MicroMod Processor boards come equipped with the **M.2 MicroMod Connector**, which leverages the M.2 standard and specification to allow you to install your MicroMod Processor board on your choice of carrier board.



STM32F405

There is so much packed into this chip! As stated in the introduction, STMicroelectronics' STM32F405RG family is based on the ARM Cortex M4 RISC core. At 168MHz, it provides very high performance, floating point single precision, a full set of DSP instructions, and a memory protection unit that enhances application security. For more information, refer to the Datasheet.



Power

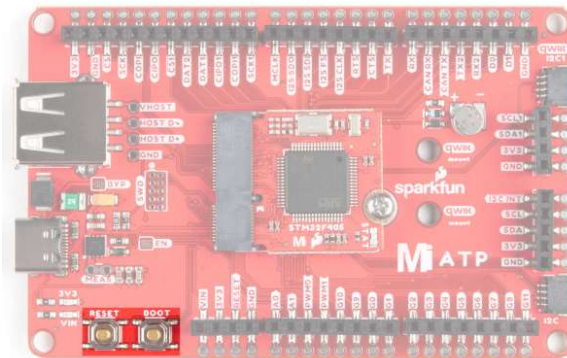
Power is supplied by the carrier board, but it should be noted that all pins are **3.3V**.

⚡ **Warning:** All pins are **3.3V**. DO NOT expose the pins to 5V.

Boot and Reset Buttons

In order to upload code to the STM32 MicroMod Processor Board, you'll need these two buttons to put the board into Boot mode. Hold the **Boot** button down, press the **Reset** button (while still holding the **Boot** button), and then release the **Boot** button.

These pins will be on the carrier board. For this tutorial, we will show you these pins on the MicroMod ATP Carrier board.

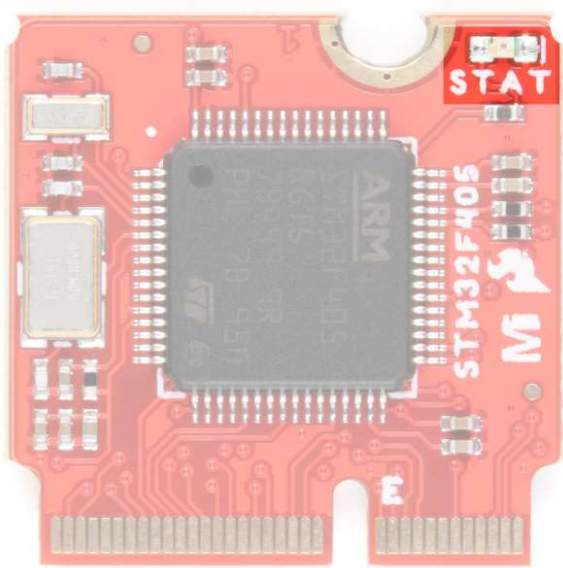


Flash

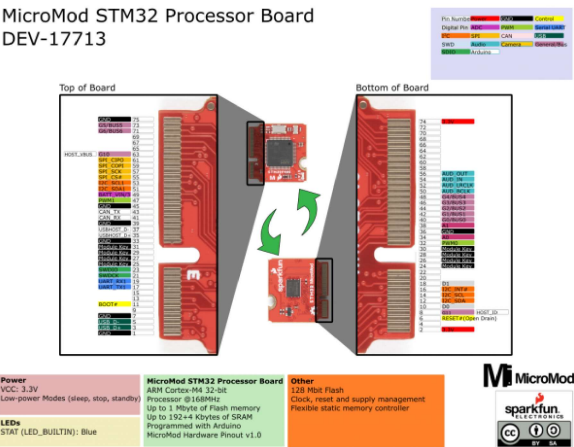
To complement the STM32F405 processor, we've added an additional 128Mb (16MB) serial flash memory chip.



Status LED



MicroMod STM32F405 Processor Pin Functionality



Click on image for a closer view of the graphical datasheet.

The complete pin map can be found in the table below or you can refer to the schematic.

STM32 PROCESSOR PINOUT TABLE

MICROMOD GENERAL PINOUT TABLE

MICROMOD GENERAL PIN DESCRIPTIONS

AUDIO	UART	GPIO/BUS	I ² C	SDIO	SPI	Dedicated
-------	------	----------	------------------	------	-----	-----------

STM32 Pin	Primary Function	Bottom Pin	Top Pin	Primary Function	STM32 Pin
-	(Not Connected)		75	GND	-
-	3.3V	74	73	G5 / BUS5	PC13
-	-	72	71	G6 / BUS6	PC2
-	-	70	69	-	-
-	-	68	67	-	-
-	-	66	65	-	-
-	-	64	63	G10/HOST_VBUS	PB13
-	-	62	61	SPI_CIPO (I)	PA6
-	-	60	59	SPI_COPI (O)	PA7
-	-	58	57	SPI_SCK (O)	PA5
PB4	AUD_OUT	56	55	SPI_CS#	PC4
PB5	AUD_IN	54	53	I2C_SCL1 (I/O)	PB6
PA4	AUD_LRCLK	52	51	I2C_SDA1 (I/O)	PB7
PB3	AUD_BCLK	50	49	BATT_VIN / 3 (I - ADC) (0 to 3.3V)	-
PC9	G4 / BUS4	48	47	PWM1	PC7
PC8	G3 / BUS3	46	45	GND	-
PA0	G2 / BUS2	44	43	CAN_TX	PB9
PA8	G1 / BUS1	42	41	CAN_RX	PB8
PD2	G0 / BUS0	40	39	GND	-
PB0	A1	38	37	USBHOST_D-	PB14
-	GND	36	35	USBHOST_D+	PB15
PC5	A0	34	33	GND	-
PC6	PWM0	32	31	Module Key	-

-	Module Key	30	29	Module Key	-
-	Module Key	28	27	Module Key	-
-	Module Key	26	25	Module Key	-
-	Module Key	24	23	SWDIO	PA13
-	-	22	21	SWCLK	PA14
-	-	20	19	UART_RX1 (I)	PA3
PC1	D1	18	17	UART_TX1 (O)	PA2
PB1	I2C_INT#	16	15	-	-
PB10	I2C_SCL (I/O)	14	13	-	-
PB11	I2C_SDA (I/O)	12	11	BOOT# (I - Open Drain)	BOOT0
PC0	D0	10	9	-	-
PB12	G11/HOST_ID	8	7	GND	-
-	RESET# (I - Open Drain)	6	5	USB_D-	PA11
-	-	4	3	USB_D+	PA12
-	3.3V	2	1	GND	-

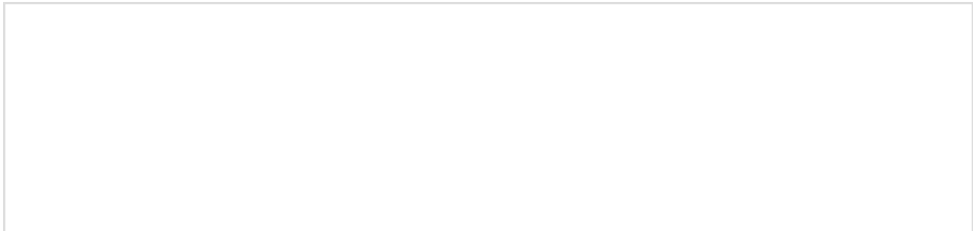
Board Outline

The board takes advantage of the standard MicroMod form factor.



Hardware Assembly

If you have not already, make sure to check out the Getting Started with MicroMod: Hardware Hookup for information on inserting your Processor Board into your Carrier Board.

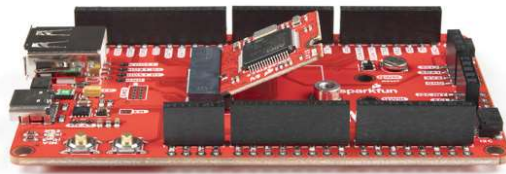


Getting Started with MicroMod

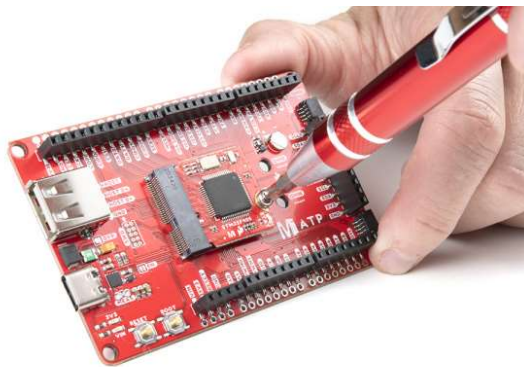
OCTOBER 21, 2020

Dive into the world of MicroMod - a compact interface to connect a microcontroller to various peripherals via the M.2 Connector!

After inserting the MicroMod STM32 processor board into a carrier board, your setup may look like the following:



Go ahead and secure the Processor Board by gently pressing it down and tightening the screw (not too much though).



Software Setup and Programming

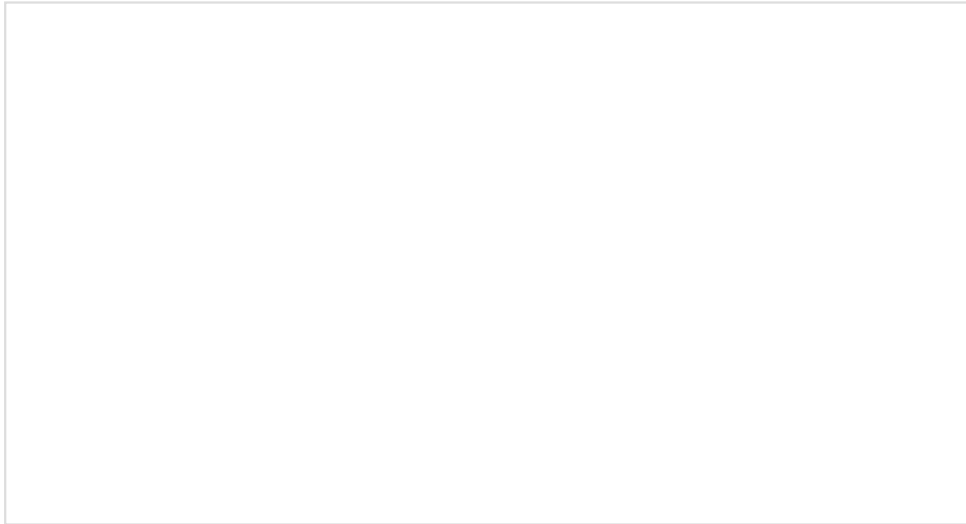
Note: This example assumes you are using the latest version of the Arduino IDE on your desktop. If this is your first time using Arduino, please review our tutorial on installing the Arduino IDE.

Arduino Board Definition

Installation for the STM32 MicroMod Processor is relatively straight-forward. You will want to install the board definitions via the Arduino Boards manager. Search for **SparkFun STM32** and you should see the option for the STM32 MicroMod Processor show up.



For more information on installing boards via the Arduino Board Manager, check out the add-ons section of our [Installing Arduino IDE tutorial](#).



Installing Arduino IDE

MARCH 26, 2013

A step-by-step guide to installing and testing the Arduino software on Windows, Mac, and Linux.

Install STM32Cube Programmer Software

In order to work with the STM32 MicroMod Processor, you'll need to install the STM32Cube Programmer. This is an all-in-one multi-OS software tool for programming STM32 products. It primarily provides the driver we need, but you can also program your board using this GUI.

GET THE STM32CUBE PROGRAMMER HERE

DFU Bootloader

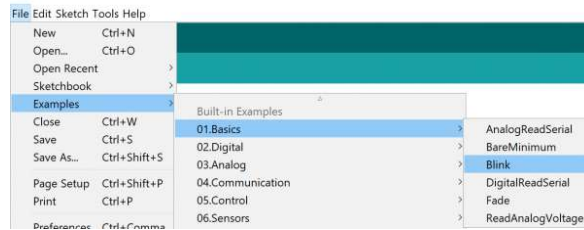
As of this writing, SparkFun is using the DFU bootloader to upload code to the STM32 MicroMod Processor. In order to do so, you need to do the following:

- Press and hold down the Boot button
- Press and release the Reset button while continuing to press the Boot button
- Keep pressing the Boot button until the code is uploaded

Examples

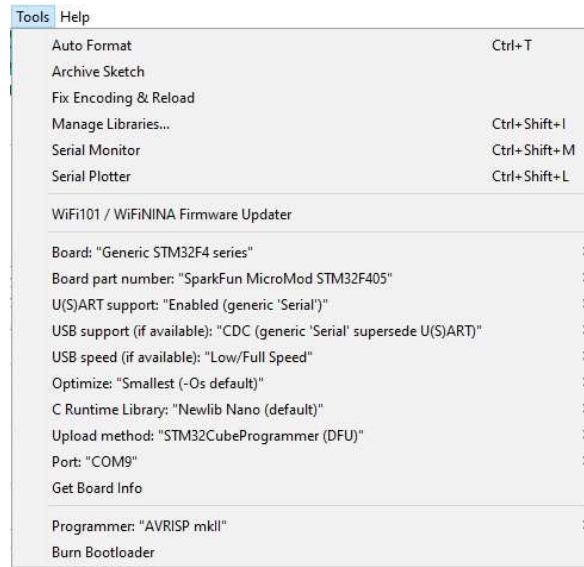
Blink

Let's start with something basic - let's blink an LED. Go to **File->Examples->01.Basics->Blink**.



Having a hard time seeing? Click the image for a closer look.

Once you've plugged your MicroMod Carrier Board with your MicroMod STM32 Processor into your computer, you'll need to go to your Tools menu and set up your options to look like the following:



With everything setup correctly, you'll need to put the Carrier Board into *Boot Mode* in order to upload the code.

- Press and hold down the Boot button
- Press and release the Reset button while continuing to press the Boot button
- Release the Boot button and press the *Upload* button in your Arduino IDE

Once the code finishes transferring, you should see the STAT LED on the STM32 Processor Board begin to blink!



If the blue LED remains dimly lit, it's probably still sitting in the bootloader. After uploading a sketch, you may need to **tap the RST button** to get your STM32 MicroMod Processor Board to begin running the sketch.

I²C Scanner

The Qwiic Connect Ecosystem makes attaching sensors a breeze. That said, sometimes it's nice to be able to scan your I²C connections to find out the address of your sensor. That's what we'll do here!

Grab your MicroMod STM32 Processor Board and your Carrier Board, and attach a Qwiic Sensor to the Qwiic port on the Carrier like so:



Copy and paste the code below into a new Arduino sketch.

```
// -----  
// I2C Scanner example using Wire1  
//  
//  
// This sketch tests the standard 7-bit addresses  
// Devices with higher bit address might not be seen properly.  
//  
  
#include <Wire.h>  
  
TwoWire Wire1(SDA1,SCL1); //Intialize Wire1 class  
  
void setup()  
{  
  Wire1.begin();  
  
  Serial.begin(115200);  
  while (!Serial);           // Leonardo: wait for serial monitor  
  Serial.println("\nI2C Scanner");  
}  
  
void loop()  
{  
  byte error, address;  
  int nDevices;  
  
  Serial.println("Scanning...");  
  
  nDevices = 0;  
  for(address = 1; address < 127; address++ )  
  {  
    // The i2c_scanner uses the return value of  
    // the Write.endTransmission to see if  
    // a device did acknowledge to the address.  
    Wire1.beginTransmission(address);  
    error = Wire1.endTransmission();  
  
    if (error == 0)  
    {  
      Serial.print("I2C device found at address 0x");  
      if (address<16)  
        Serial.print("0");  
      Serial.print(address,HEX);  
      Serial.println(" !");  
  
      nDevices++;  
    }  
    else if (error==4)  
    {  
      Serial.print("Unknown error at address 0x");  
      if (address<16)  
        Serial.print("0");  
      Serial.println(address,HEX);  
    }  
  }  
}  
if (nDevices == 0)
```

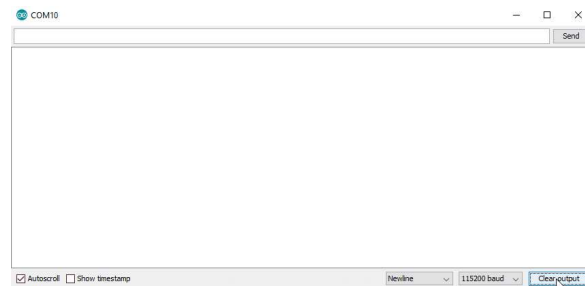
```
Serial.println("No I2C devices found\n");
else
    Serial.println("done\n");

delay(5000);           // wait 5 seconds for next scan
}
```

Make sure your options are all set up correctly in the *Tools* menu, and make sure you put the Carrier Board into *Boot Mode* in order to upload the code.

- Press and hold down the Boot button
- Press and release the Reset button while continuing to press the Boot button
- Release the Boot button and press the *Upload* button in your Arduino IDE

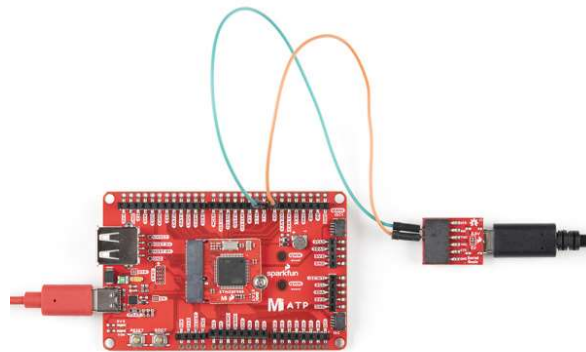
After uploading, open the serial monitor and set the baud to **115200**. You should see a similar printout to the one shown below.



UART Example

Let's have a quick look at an example using UART. If you're unfamiliar with Serial Output, go ahead and have a look at our [Serial Basic Tutorial](#).

Grab your MicroMod STM32 Processor Board and your Carrier Board, and attach the Serial Basic Rx and Tx pins like so:



Click on the image for a closer view

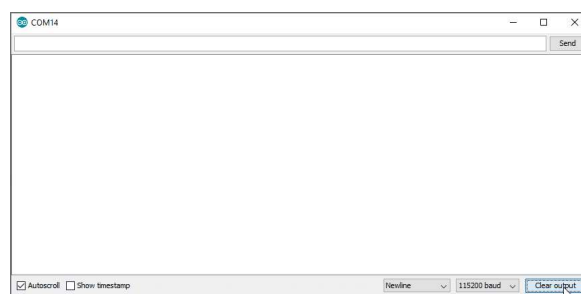
Copy and paste the code below into a new Arduino sketch.

```
// -----  
// UART example using Serial1  
//  
//  
// This sketch prints "Hello World!" every second  
// using the secondary UART pins RX1 and TX1.  
//  
  
HardwareSerial Serial1(RX1, TX1); //Attach Serial1 to RX1 and TX1  
  
void setup() {  
  Serial1.begin(115200);  
  while (!Serial1) {  
    ; // wait for serial port to connect. Needed for Native USB only  
  }  
  Serial1.println("Goodnight moon!");  
}  
  
void loop() {  
  Serial1.println("Hello World!");  
  delay(1000);  
}
```

Make sure your options are all set up correctly in the *Tools* menu, and make sure you put the Carrier Board into *Boot Mode* in order to upload the code.

- Press and hold down the Boot button
- Press and release the Reset button while continuing to press the Boot button
- Release the Boot button and press the *Upload* button in your Arduino IDE

Once your code is uploaded, open up the Serial Monitor attached to your Serial Basic with the baud rate set to **115200** to see your output!



Troubleshooting

❗ Not working as expected and need help?

If you need technical assistance and more information on a product that is not working as you expected, we recommend heading on over to the SparkFun Technical Assistance page for some initial troubleshooting.

SPARKFUN TECHNICAL ASSISTANCE PAGE

If you don't find what you need there, the SparkFun Forums are a great place to find and ask for help. If this

is your first visit, you'll need to create a Forum Account to search product forums and post questions.

[CREATE NEW FORUM ACCOUNT](#)[LOG INTO SPARKFUN FORUMS](#)[SPARKFUN FORUMS: MICROMOD](#)

Resources and Going Further

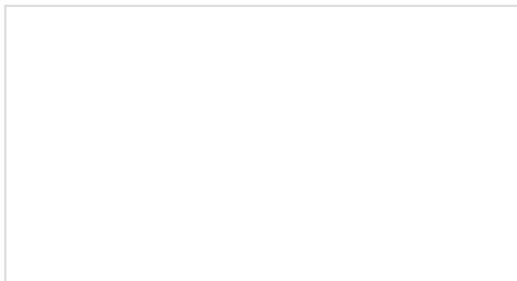
- **MicroMod STM32 Processor Documentation**

- Schematic (PDF)
- Eagle Files (ZIP)
- Board Dimensions (PNG)
- STM32F405xx Datasheet (PDF)
- Graphical Datasheet (PDF)
- GitHub Repo

- **MicroMod Documentation**

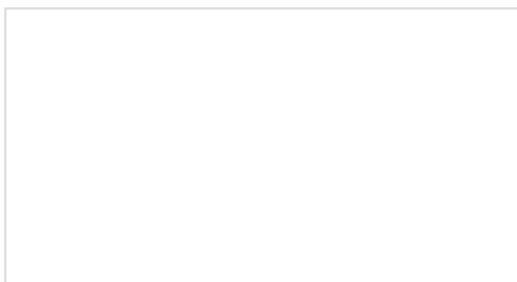
- SparkFun MicroMod Interface v1.0 - Pinout (PDF)
- SparkFun MicroMod Interface v1.0 - Pin Descriptions (PDF)
- Getting Started with MicroMod
- Designing with MicroMod
- MicroMod Info Page
- MicroMod Forums
- SparkFun Eagle Libraries contains example footprints for the M.2 connector and SMD standoff
- M.2 MicroMod Connector Datasheet
- MicroMod Reflowable Standoff Datasheet

Need some inspiration for your next project? Check out some of these related tutorials using STM32:

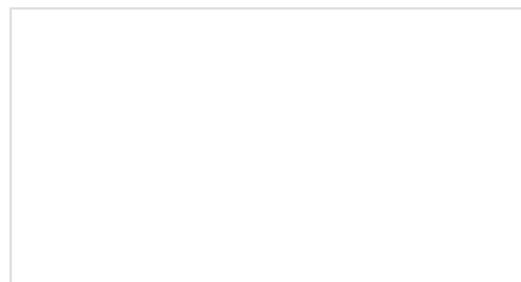


STM32 Thing Plus Hookup Guide
Get started with the STM32 Thing Plus!

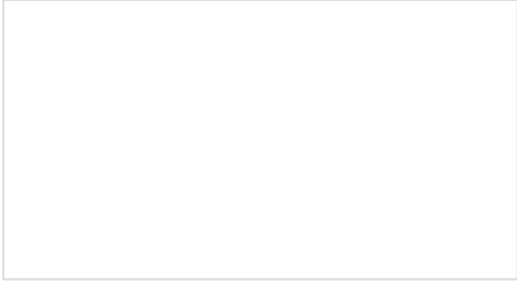
Or check out other tutorials with MicroMod:



MicroMod ESP32 Processor Board Hookup Guide
A short hookup guide to get started with the SparkFun MicroMod ESP32 Processor Board.

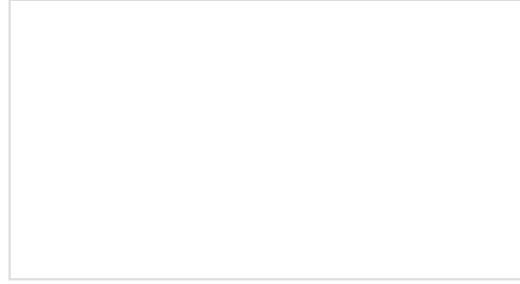


SparkFun MicroMod Input and Display Carrier Board Hookup Guide
A short Hookup Guide to get started with the SparkFun MicroMod Input and Display Carrier Board



1W LoRa MicroMod Function Board Hookup Guide

Everything you need to get started with the 1W LoRa MicroMod function board; a MicroMod function board that provides LoRa capabilities for your MicroMod project. Must be used in conjunction with a MicroMod main board and processor.



MicroMod Alorium Sno M2 Processor Board Hookup Guide

Get started with the MicroMod Alorium Sno M2 Processor Board!