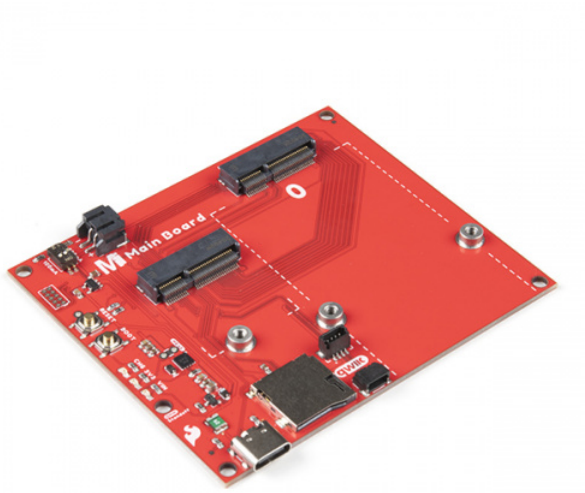
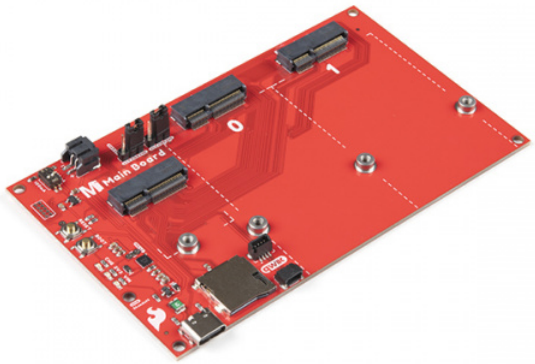
MicroMod Main Board Hookup Guide

Introduction

The MicroMod Main Board - Single and Double are specialized carrier boards that allow you to interface a Processor Board with a Function Board(s). The modular system allows you to add an additional feature(s) to a Processor Board with the help of a Function Board(s).



SparkFun MicroMod Main Board - Single
DEV-18575

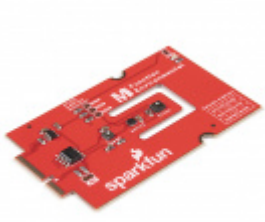


SparkFun MicroMod Main Board - Double

DEV-18576

Required Materials

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



SparkFun MicroMod Environmental Function Board

SEN-18632

SparkFun MicroMod Main Board - Single

DEV-18575

SparkFun MicroMod WiFi Function Board - ESP32

WRL-18430

Reversible USB A to C Cable - 2m

CAB-15424

SparkFun MicroMod Artemis Processor

DEV-16401

Pocket Screwdriver Set

TOL-12891

microSD Card - 1GB (Class 4)

COM-15107

MicroMod Main Board

To hold the processor and function boards, you will need one Main board. Depending on your application, you may choose to have either one or two function boards.

SparkFun MicroMod Main Board - Single
◉ DEV-18575

SparkFun MicroMod Main Board - Double
◉ DEV-18576

MicroMod Processor Board

There are a variety of MicroMod Processor Boards available to choose from. You will probably want to avoid having the same Processor and Function Board since there is an ESP32 on both types of boards.

SparkFun MicroMod Artemis Processor
◉ DEV-16401

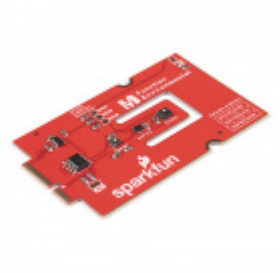
SparkFun MicroMod ESP32 Processor
○ WRL-16781

SparkFun MicroMod nRF52840 Processor
◉ WRL-16984

SparkFun MicroMod STM32 Processor
○ DEV-17713

MicroMod Function Board

To add additional functionality to your Processor Board, you'll want to include one or two function boards when connecting them to the Main Board. Make sure to check out the catalog for other function boards.



SparkFun MicroMod Environmental Function Board

SEN-18632

SparkFun MicroMod WiFi Function Board - ESP32

WRL-18430

Tools

You will need a screw driver to secure the Processor and Function boards. To set the charge rate, you will need a precision screw driver (or a tiny, rounded object). The pocket screwdriver set is an excellent option. For users using a microSD card and want to easily read the contents of the memory card, you will need a microSD card adapter or USB reader.

microSD USB Reader

COM-13004

Pocket Screwdriver Set

TOL-12891

SparkFun Mini Screwdriver

TOL-09146

Suggested Reading

If you aren't familiar with the MicroMod ecosystem, we recommend reading here for an overview. We recommend reading here for an overview if you decide to take advantage of the Qwiic connector.

MicroMod Ecosystem	*Qwiic Connect System*

If you aren’t familiar with the following concepts, we also recommend checking out a few of these tutorials before continuing. Make sure to check the respective hookup guides for your processor board and function board to ensure that you are installing the correct USB-to-serial converter. You may also need to follow additional instructions that are not outlined in this tutorial to install the appropriate software.

What is an Arduino?
What is this 'Arduino' thing anyway? This tutorials dives into what an Arduino is and along with Arduino projects and widgets.

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

How to Install CH340 Drivers
How to install CH340 drivers (if you need them) on Windows, Mac OS X, and Linux.

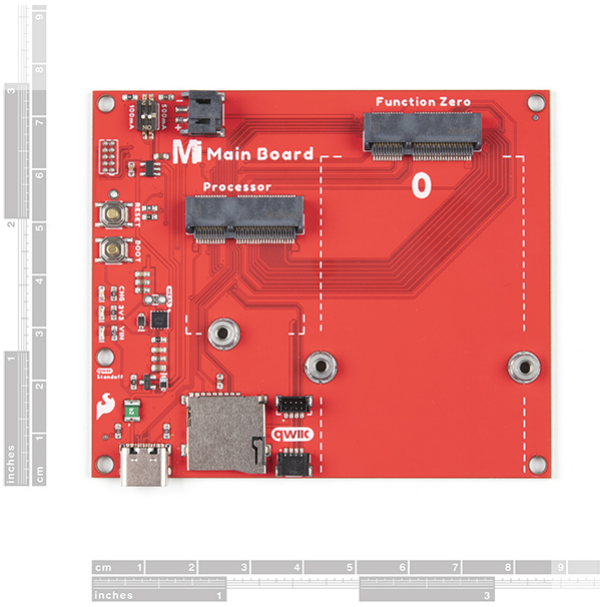
Getting Started with MicroMod
Dive into the world of MicroMod - a compact interface to connect a microcontroller to various peripherals via the M.2 Connector!

Hardware Overview

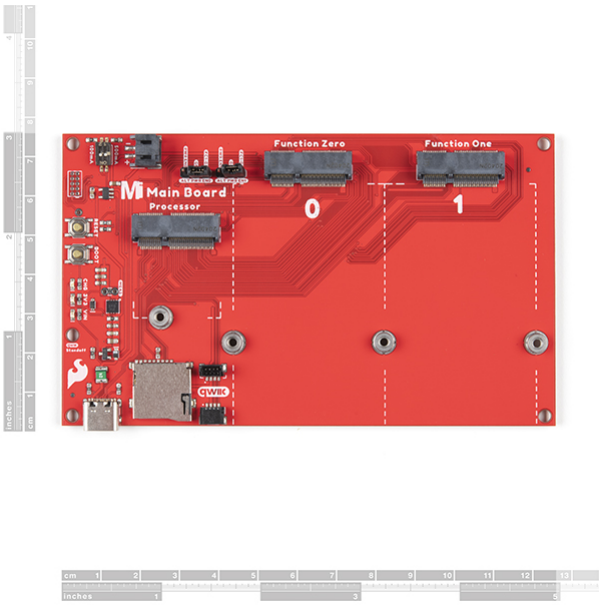
The overall functionality of the Single and Double Main Boards are the same. We'll use the Single Main Board more in this section to highlight the features since this is also included in the Double Main Board. We'll switch to the Double Main Board when necessary to highlight the features that are only included in the Double Main Board.

The only differences are that the Double Main Board includes:

- two jumper shunts
- ability to add a second MicroMod Function Board to the mix
- board's width



Main Board - Single Measurements



Main Board - Double Measurements (Zoomed Out)

Power

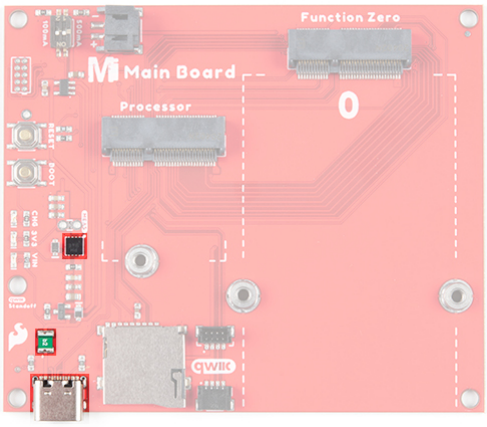
There are two ways to power the Main Boards, Processor Board, and Function Board(s).

- USB
- Single Cell LiPo Battery

It is fine to connect a power source to the USB connector and LiPo battery's JST connector at the same time. The MicroMod Main Board has power-control circuitry to automatically select the best power source.

Power USB

One option of powering the board is through the USB Type C connector. You will need a USB Type C cable to power the board with **5V**. Power connected to the board's USB C connector will go through a resettable PTC fuse (rated at 2A max) and then the AP7361C 3.3V voltage regulator (rated at 1A max). The little green component close to the USB connector is the resettable PTC fuse while the square IC is the voltage regulator. The voltage regulator accepts voltages between *~3.3V to 6.0V.*

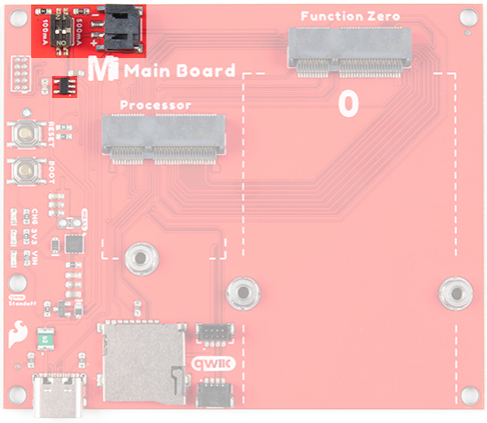


The USB Type C connector is also used to upload code to your Processor Board, send serial data to a terminal window, or charge the LiPo battery. Of course for portable power, you could connect a USB battery as an alternative to using a LiPo battery.

Power applied to the connector will light up the VIN and 3V3 LED. If you decide to bypass the PTC fuse, simply add a solder blob to the jumper labeled as PTC. There is also a jumper labeled as MEAS to measure the current consumption at the output of the 3.3V voltage regulator for your project.

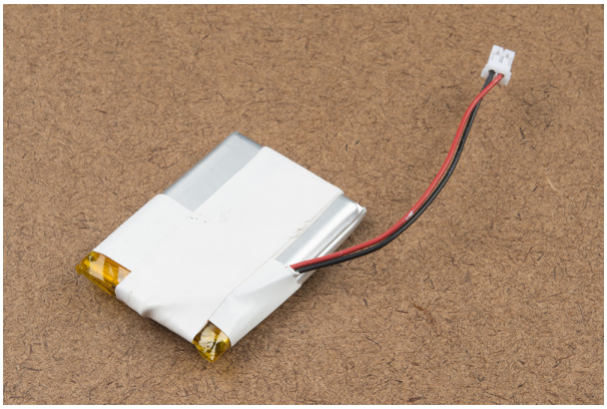
Power LiPo

The other option is to connect a single cell LiPo battery (i.e. nominal **3.7V**, **4.2V** fully charged) to the 2-pin JST connector as shown below. A MCP73831 charge IC is included on the boards to safely charge the LiPo batteries via USB Type C connector. A switch is included to set the charge rate. The charge rate is probably set to **~166mA** with both switches flipped to the ON position. This may vary depending on the position of the switch when it was pulled from the reel. Flip the switch to adjust the charge rate to either *100mA* or *500mA* using a precision flat head screw driver or tweezers.



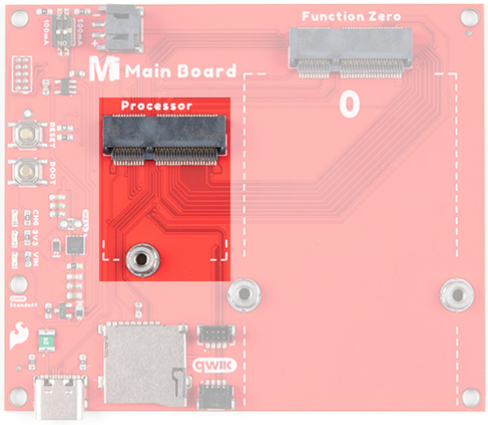
The voltage from the LiPo battery is regulated down to **3.3V** as it goes through the AP7361C 3.3V voltage regulator (rated at 1A max).

Note: For more information on proper handling of LiPo batteries, check out the LilyPad Basics: Powering Your Project - LiPo Battery Safety Care.



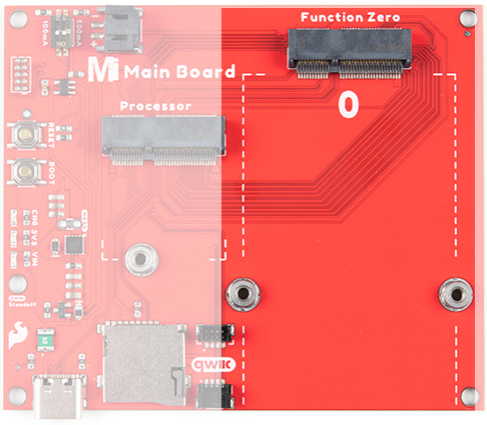
MicroMod Processor Board

The MicroMod ecosystem allows you to easily swap out processors depending on your application. The location of the M.2 connector labeled as Processor is where you would connect and secure a MicroMod Processor Board.

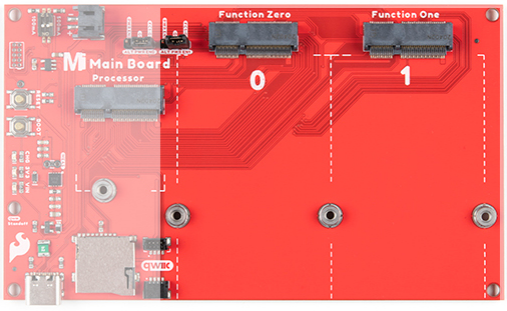


MicroMod Function Board

Beside the MicroMod Processor's socket is another M.2 connector for MicroMod Function Boards, which allow you to add additional functionality to your Processor Board. The Single Main Board includes one socket for a single Function Board while the Double Main Board includes two sockets for up two Function Boards.



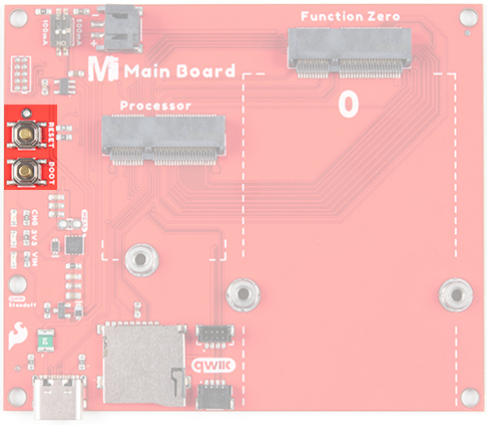
Main Board - Single Function Board Socket



Main Board - Double Function Board Sockets

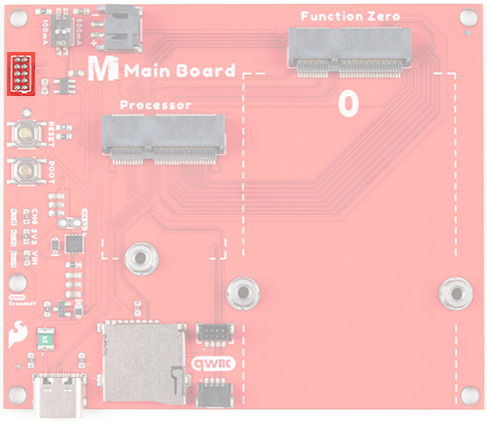
Reset and Boot Buttons

Each board includes a reset and boot button. There is an additional reset button PTH next to the reset button. Hitting the reset button to restart your Processor Board. Hitting the boot button will put the Processor Board into a special boot mode. Depending on the processor board, this boot pin may not be connected.



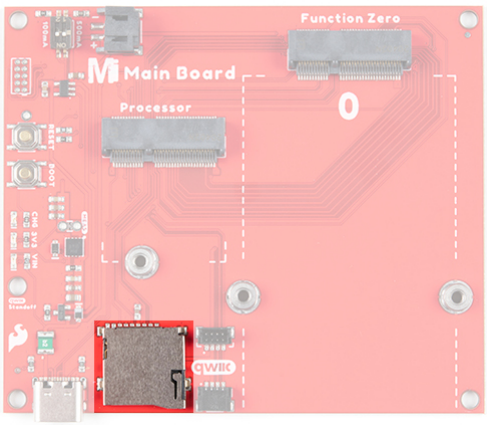
SWD Pins

For advanced users, we proke out the 2x5 SWD programming pins. Note that this is not populated so you will need a compatible header and compatible JTAG programmer to connect.



MicroSD Socket

The board includes a microSD socket if your application requires you to log and save data to a memory card. The primary SPI pins (SDO, SDI, SCK, CS0) from your Processor and Function Board are connected to the microSD Socket.



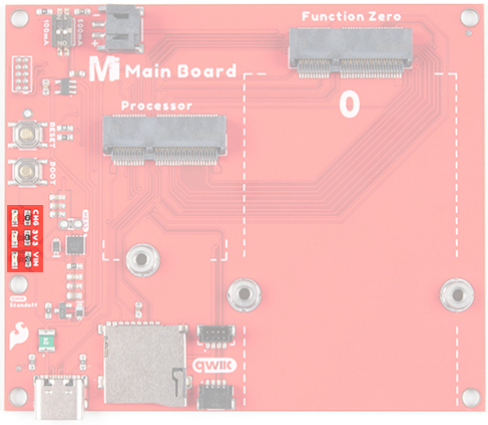
Note: Note that the CS pin on the MicroMod Main Board - Single is on **Processor Board's** `D1` while the MicroMod Main Board - Double is on the **Processor Board's** `G4` .

LEDs

There are three LEDs on the board:

- **VIN** - The VIN LED lights up to indicate when power available from the USB connector.
- **3V3** - The 3V3 LED lights up to indicate when there is a 3.3V available after power is regulated down from the USB connector or LiPo battery.
- **CHG** - The on-board yellow CHG LED can be used to get an indication of the **charge status** of your battery. Below is a table of other status indicators depending on the state of the charge IC.

Charge State	LED status
No Battery	Floating (should be OFF, but may flicker)
Shutdown	Floating (should be OFF, but may flicker)
Charging	ON
Charge Complete	OFF



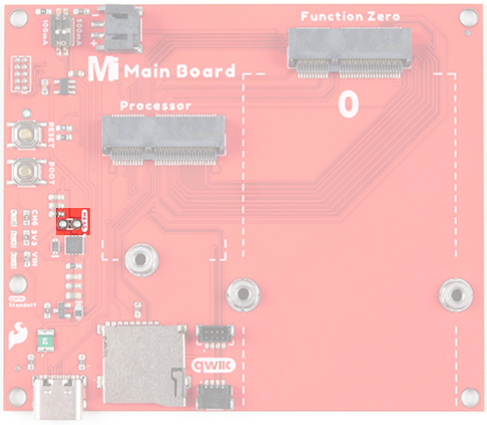
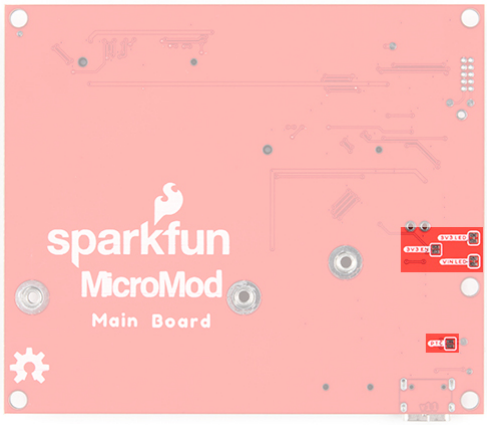
Jumpers

Note: If this is your first time working with jumpers, check out the How to Work with Jumper Pads and PCB Traces tutorial for more information.

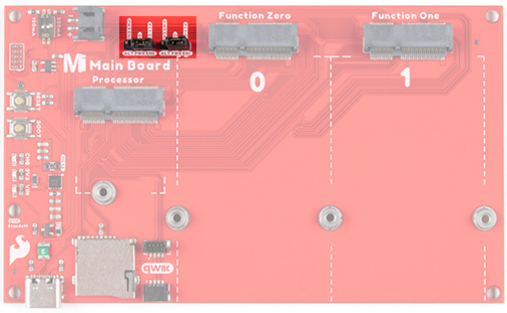
The following five jumpers are included on both the Single and Double Main Boards.

- **MEAS** - By default, the jumper is closed and located on the top side of the board. This jumper is used to measure your system's current consumption. You can cut this jumper's trace and connect the PTHs to a ammeter/multimeter to probe the output from the 3.3V voltage regulator. Check out our How to Use a Multimeter tutorial for more information on measuring current.

- **PTC** - By default, the jumper is open and located on the bottom of the board. For advanced users that know what you are doing, add a solder blob to the jumper to bypass the resettable PTC fuse to pull more than 2A from the USB source.
- **3.3V EN** - By default, this jumper is open and located on the bottom of the board. Closing this jumper enables processor control of the 3.3V bus.
- **VIN LED** - By default, this jumper is closed and located on the bottom of the board. Cut this trace to disable the LED that is connected to the input of the USB
- **3.3V LED** - By default, this jumper is closed and located on the bottom of the board. Cut this trace to disable the LED that is connected to the output of the 3.3V voltage regulator.

	
Main Board - Single Top View Jumpers	*Main Board - Single Bottom View Jumpers*

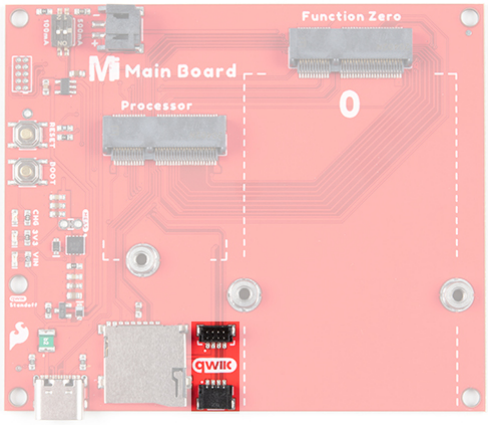
Included only on the Double Main Board are two 1x3 male headers with 2-pin jumper shunts to enable the 3.3V voltage regulator for any Function Board connected to Function Zero and Function One using alternative Processor GPIO pins. Since certain processors have limited GPIO and may not be broken out on certain locations, alternative pins have been provided on the board. The **ALT PWR EN0** jumper allows users to control the 3.3V voltage regulator on any Function Board that is connected to **Function Zero**. When the jumper shunt is on the left side toward the 2-pin JST connector, the jumper shunt connects the **PWR EN0** to the Processor Board's GPIO **G5** pin. Moving the jumper shunt to the other side connects the Processor Board's GPIO **G5** pin to Function Board One's GPIO **G3** pin.



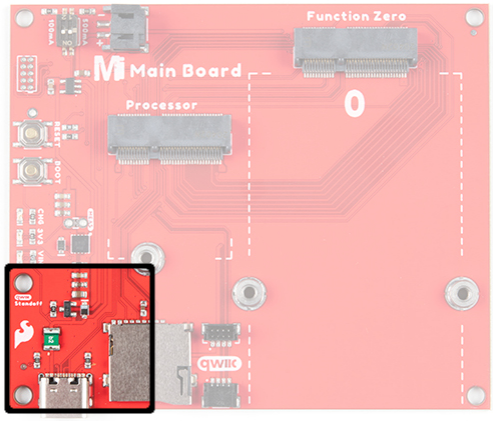
The **ALT PWR EN1** jumper allows users control power from the 3.3V voltage regulator for any Function Board that is connected to **Function One** when the jumper shunt is connecting **PWR EN1** and Processor Board's GPIO **G6** pin. Moving the jumper shunt to the other side connects the Processor Board's GPIO **G6** pin to Function Board One's GPIO **G4** pin.

Qwiic and I^2C

The board includes a vertical and horizontal Qwiic connector. These are connected to the primary I^{2}C bus and 3.3V power on both the Processor and Function Board connectors allowing you to easily add a Qwiic-enabled device to your application.



Note that there are two mounting holes for Qwiic-enabled boards that have a standard 1.0"x1.0" size board. The image below highlighted with a black square is where you would place the board.



MicroMod Pinout

Depending on your window size, you may need to use the horizontal scroll bar at the bottom of the table to view the additional pin functions. Note that the M.2 connector pins on opposing sides are offset from each other as indicated by the bottom pins where it says (Not Connected)*. There is no connection to pins that have a "-" under the primary function.

MICROMOD MAIN BOARD - SINGLE
MICROMOD MAIN BOARD - DOUBLE
MICROMOD GENERAL FUNCTION BOARD
MICROMOD GENERAL PROCESSOR BOARD
MICROMOD GENERAL PIN DESCRIPTIONS

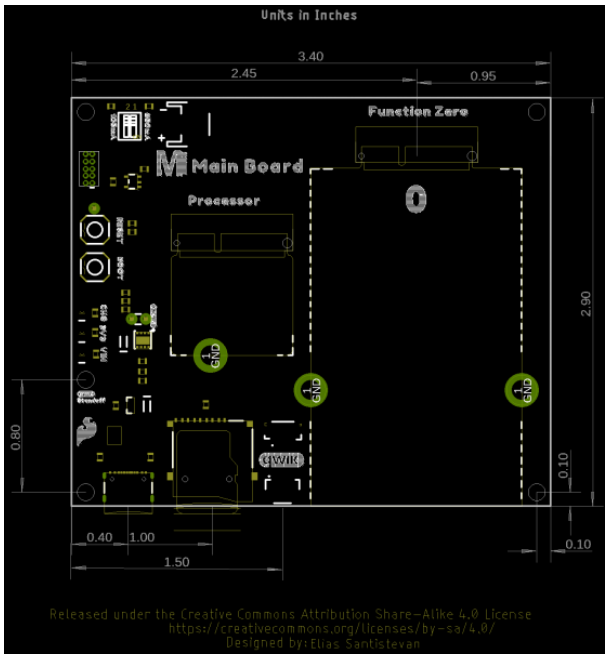
AUDIO	UART	GPIO/BUS	I^{2}C	SDIO	SPI	Dedicated

Processor Pins	Function Zero Pins	Main Board - Single Miscellaneous Pins
GND	GND	GND
3.3V	-	3.3V
USB_D+_Processor	-	USB_D+
3.3V_EN	-	3.3V_EN
USB_D-_Processor	-	USB_D-
RESET# (I - Open Drain)	-	RESET# (I - Open Drain)
GND	GND	GND
USB_VIN	-	USB_VIN
D0_Processor	F0	-
BOOT (I - Open Drain)	-	BOOT (I - Open Drain)
I2C_SDA_Processor	I2C_SDA	I2C_SDA (Qwiic)
UART_RTS_Processor	UART_RTS	-
I2C_SCL_Processor	I2C_SCL	I2C_SCL (Qwiic)
UART_CTS_Processor	UART_CTS	-
I2C_INT_Processor	I2C_INT	-
UART_TX_Processor	UART_RX	-
D1_Processor	-	SD Card CS (microSD Card)
UART_RX_Processor	UART_TX	-
SWDCK_Processor	-	SWDCK (2x5 Header)
SWDIO_Processor	-	SWDCK (2x5 Header)
PWM0_Processor	PWM0	-
GND	GND	GND
A0_Processor	A0	-
USBHOST_D+_Processor	USBHOST_D+	-

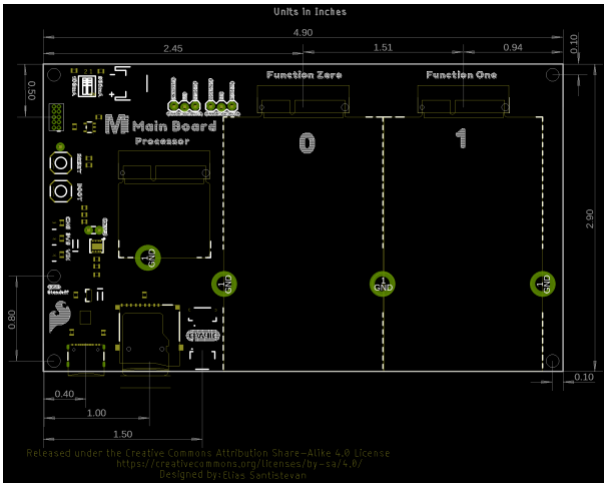
GND	GND	GND
USBHOST_D-_Processor	USBHOST_D-	-
A1_Processor	PWR_EN0	-
GND	GND	GND
G0_Processor	F3	-
CAN_RX_Processor	CAN_TX	-
G1_Processor	F4	-
CAN_TX_Processor	CAN_RX	-
G2_Processor	F5	-
GND	GND	GND
G3_Processor	F6	-
G4_Processor	F7	-
BATT_VIN / 3 (I - ADC) (0 to 3.3V)	-	BATT_VIN / 3 (I - ADC) (0 to 3.3V)
SPI_CS0_Processor	F1	-
SPI_SCK_Processor	SPI_SCK	SPI_SCK (microSD Card)
SPI_DI_Processor	SPI_DO	SPI_DO (microSD Card)
SPI_DO_Processor	SPI_DI	SPI_DI (microSD Card)
SDIO_DATA2_Processor	PWR_EN0	-

Board Dimensions

The board dimension of the MicroMod Main Board - Single is 2.90" x 3.40" while the MicroMod Main Board - Double is 2.90" x 4.90". Both boards include 5x mounting holes. Four are located on the edge of each board. The fifth mounting hole is located 0.80" away from another mounting hole to mount Qwiic-enabled boards that have the standard 1.0"x1.0" size board.



MicroMod Main Board - Single



MicroMod Main Board - Double

Note: You'll notice that the Main Boards have the USB C connector, microSD card socket, and Qwiic connector on one side of the board. This is part of the design for users deciding to place the MicroMod Main Board in an enclosure. The JST connector for the LiPo battery is facing in toward the board is also part of the design. Users can insert a LiPo battery and have its wires neatly tucked into their enclosure.

Hardware Hookup

If you have not already, make sure to check out the Getting Started with MicroMod: Hardware Hookup for information on inserting your Processor and Function Boards to the Main 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!

USB

To program and power the Main Board, you will need to insert the USB-C cable into the USB connector. We will leave the other end disconnected when connecting a Processor or Function board to the Main Board.

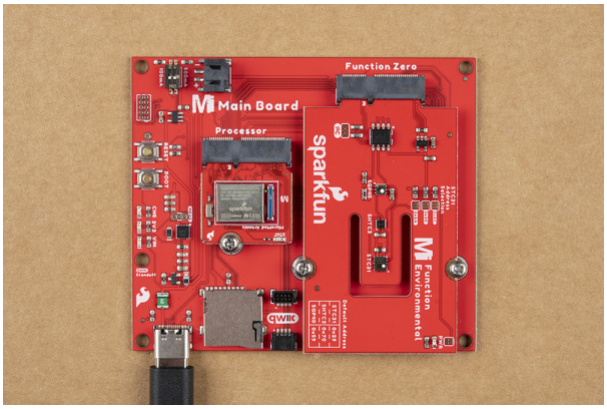
When the boards are secure, insert the other end to a computer. When you are finished programming the processor, you can use a USB battery via the USB connector or LiPo battery via the JST connector to power the board.

Processor Board

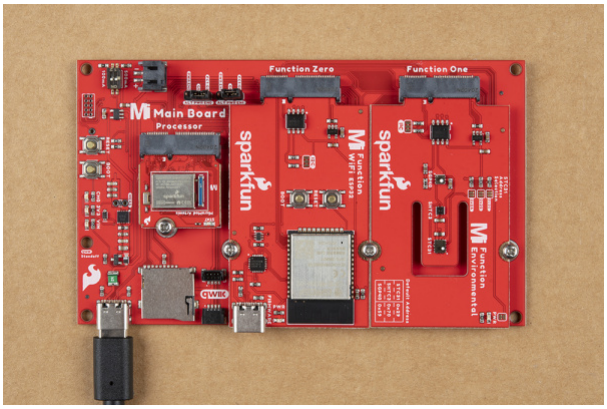
Align the Processor Board's key into its M.2 connector's socket. Insert the board at an angle (~25°), push down, and tighten the screw. In this case, we had the MicroMod Artemis Processor Board secured in the M.2 connector socket. Depending on your application, you may have a different Processor Board.

Function Boards

Align the Function Board's key into its M.2 connector's socket. Insert the board at an angle (~25°), push down, and tighten one of the screw to hold the board down. Attach the second screw on the other side of the board. Once the board is aligned, tighten both screws fully to secure the board. In this case, we had the Environmental Function Board secured in the M.2 connector socket. Depending on your application, you may have a different Function Board.



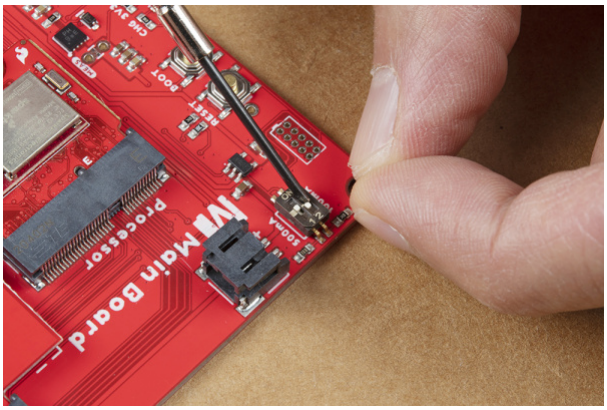
If you decide to have two function boards attached to the Main Board - Double, we recommend tightening the screw between the two Function Boards first to hold them down before attaching the remaining screws on either side of the Function Boards. In this case, we had the WiFi Function Board and the Environmental Function Board secured in the M.2 connector socket. Depending on your application, you may have different function boards.



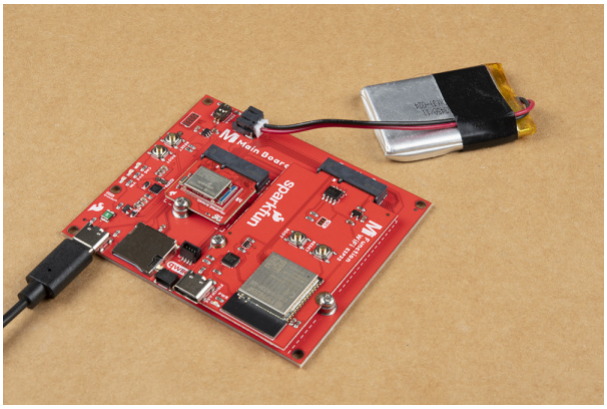
Single Cell LiPo Battery

Note: You may notice a film that is covering over the switch. This is used by the pick and place machine to pick the component up from the reel and place it on the PCB. You will need to peel the film back to access the switches and adjust the charge rate.

A precision flat head (or a tiny, rounded object) is needed to set the charge rate for your single cell LiPo battery. When the switch is flipped to the ON position on the 500mA side, the charge rate will be **500mA**. When the switch is flipped to the ON position on the 100mA side, the charge rate will be **100mA**. When the switch is flipped to the ON position on both the 100mA and 500mA sides, the charge rate will be **166mA**.

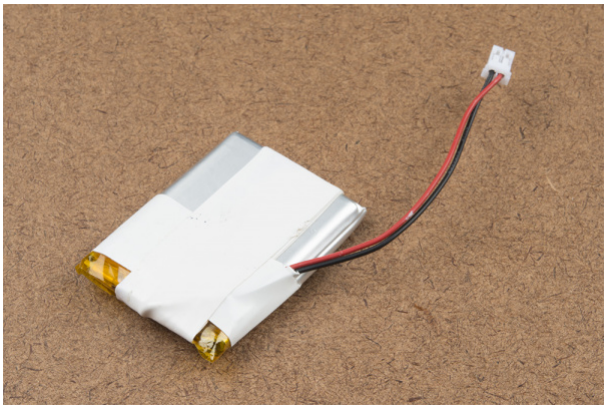


For mobile applications, attach a single cell LiPo battery to the 2-pin JST connector. Attach a USB cable to a USB port or charger when the battery is low to begin charging.



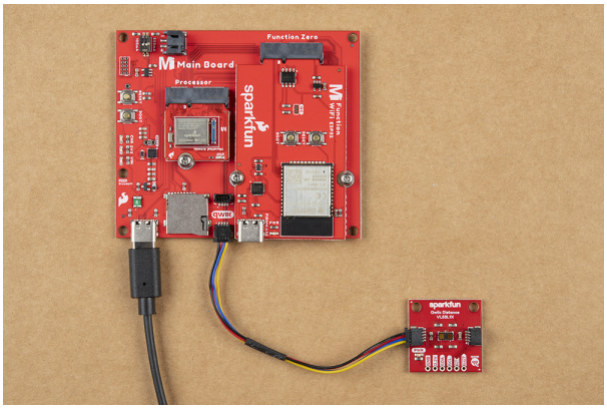
To remove the LiPo battery, you can pull the white JST connector away from the socket while also wiggling the connector side to side using your thumb and index finger.

Note: For more information on proper handling of LiPo batteries, check out the LilyPad Basics: Powering Your Project - LiPo Battery Safety Care.

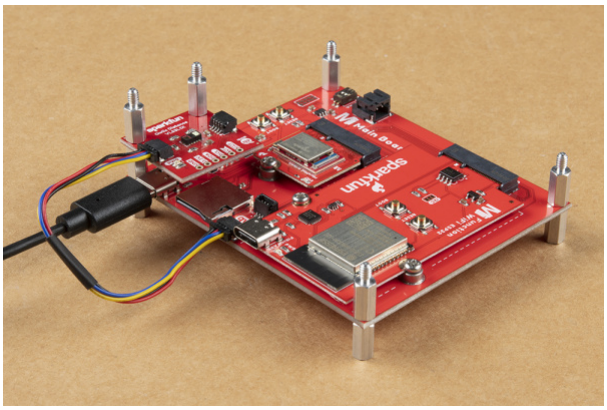


Qwiic-Enabled Devices

To Qwiic-ly connect I^{2}C devices, simply insert a Qwiic cable between one of the MicroMod Main Board's Qwiic ports and your Qwiic device.

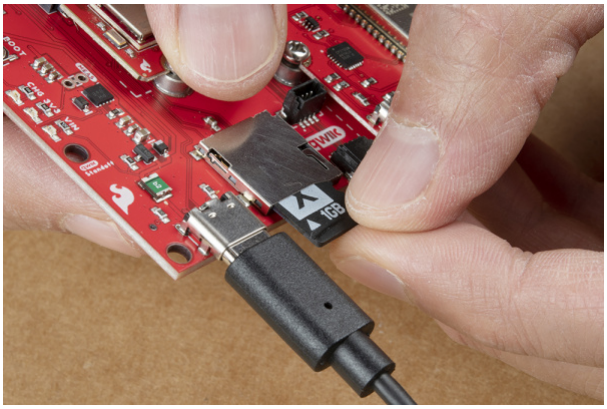


If you need to mount a Qwiic-enabled board to the MicroMod Main Board, you can grab some standoffs and mount a standard Qwiic 1.0"x1.0" sized board using the two mounting holes near the USB Type C connector. Place the standoff between the boards and tighten the screws to mount. The image below used standoffs with built-in threads.



MicroSD Card

With power removed from the board, insert a microSD card (with the pins facing toward the board) into the socket. You'll hear a nice click indicating that the microSD card is locked in place.



To remove, make sure power is off and press the microSD card into the socket to eject. You'll hear a nice click indicating that the card is ready to be removed from the socket.

Software Installation

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 the following tutorials.

- Installing the Arduino IDE
- Installing Board Definitions in the Arduino IDE
- Installing an Arduino Library

Arduino Board Definitions and Driver

We'll assume that you installed the necessary board files and drivers for your Processor Board. In this case, we used the MicroMod Artemis Processor Board which uses the CH340 USB-to-serial converter. If you are using a Processor Board, make sure to check out its hookup guide for your Processor Board.

MicroMod Artemis Processor Board Hookup Guide

OCTOBER 21, 2020

Get started with the Artemis MicroMod Processor Board in this tutorial!

How to Install CH340 Drivers

AUGUST 6, 2019

How to install CH340 drivers (if you need them) on Windows, Mac OS X, and Linux.

Arduino Examples

Let's go over a few basic examples to get started with the MicroMod Main Board Single and Double. We'll toggle the 3.3V voltage regulator and log data to a microSD card on both boards.

Example 1: MicroMod Main Board - Single Enable Function Board

If you have not already, select your Board (in this case the **MicroMod Artemis**), and associated COM port. Copy and paste the code below in your Arduino IDE. Hit the upload button and set the serial monitor to **115200** baud.

```
/*************************************************************************

  WRITTEN BY: Ho Yun "Bobby" Chan
  @ SparkFun Electronics
  DATE: 10/19/2021
  GITHUB REPO: https://github.com/sparkfun/SparkFun_MicroMod_Main_Board_Single
  DEVELOPMENT ENVIRONMENT SPECIFICS:
    Firmware developed using Arduino IDE v1.8.12

  ========== DESCRIPTION==========
  This example code toggles the Function Board's AP2112 3.3V voltage
  regulator's enable pin. The Function Boards built-in power LED should blink.
  This example is based on Arduino's built-in Blink Example:

      https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink

  Note that this example code uses the MicroMod Main Board - Single. The MicroMod
  Main Board - Double routes the PWR_EN# pins slightly different for the
  two function boards.

  ========== HARDWARE CONNECTIONS ==========
  MicroMod Artemis Processor Board => MicroMod Main Board - Single => Function Board

  Feel like supporting open source hardware?
  Buy a board from SparkFun!
       MicroMod Artemis Processor Board - https://www.sparkfun.com/products/16401
       MicroMod Main Board - Single - https://www.sparkfun.com/products/18575
       MicroMod Environmental Function Board - https://www.sparkfun.com/products/18632

  LICENSE: This code is released under the MIT License (http://opensource.org/licenses/MIT)

*************************************************************************/

/*Define the power enable pins for the processor board with either SDIO_DATA2 or A1.
  Depending on the processor board, the Arduino pin may be different.

  Note: Certain Processor Boards like the Artemis have SDIO_Data2 and A1 available
        to control the Function Board's voltage regulator. SAMD51, ESP32, and STM32
        Processor Board pins do not have SDIO Data 2, so users will need to reference
        the Processor Pin A1. Below are a few examples. */


//ARTEMIS
#define PWR_EN0 4   //Function Board 0's "PWR_EN0" pin <= MicroMod SDIO_DATA2 => Artemis Process
or Board (D4)
//Alternative option that does the same thing. Make sure to just choose one for PWR_EN0
//#define PWR_EN0 35   //Function Board 0's "PWR_EN0" pin <= MicroMod A1 => Artemis Processor Bo
ard (A35)

//TEENSY
//#define PWR_EN0 15   //Function Board 0's "PWR_EN0" pin <= MicroMod A1 => Teensy Processor Boa
rd (A1)
```

```
//SAMD51
//#define PWR_EN0 18   //Function Board 0's "PWR_EN0" pin <= MicroMod A1 => SAMD51 Processor Boa
rd (18)



void setup() {
  // initialize the digital pin as an output.
    pinMode(PWR_EN0, OUTPUT);
}



void loop() {
  digitalWrite(PWR_EN0, HIGH); // turn the 3.3V regulator on (HIGH is the voltage level)
  delay(5000);                 // wait for a few seconds to do something with the function board
s

  digitalWrite(PWR_EN0, LOW);  // turn the 3.3V regulator off by making the voltage LOW
  delay(5000);                 // wait for a few seconds before turning function boards back on

}
```

After uploading, take a look at your Function Board's PWR LED. The LED will be on for about 5 seconds and then turn off for another 5 seconds. It will continue to repeat until power is removed from the MicroMod Main Board - Single.

Example 2: MicroMod Main Board - Double Enable Function Boards

If you have not already, select your Board (in this case the **MicroMod Artemis**), and associated COM port. Copy and paste the code below in your Arduino IDE. Hit the upload button and set the serial monitor to **115200** baud.

```
/*************************************************************************

  WRITTEN BY: Ho Yun "Bobby" Chan
  @ SparkFun Electronics
  DATE: 10/19/2021
  GITHUB REPO: https://github.com/sparkfun/SparkFun_MicroMod_Main_Board_Double
  DEVELOPMENT ENVIRONMENT SPECIFICS:
    Firmware developed using Arduino IDE v1.8.12

  ========== DESCRIPTION==========
  This example code toggles the Function Board's AP2112 3.3V voltage
  regulator's enable pin. The Function Boards built-in power LED should blink.
   This example is based on Arduino's built-in Blink Example:

      https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink

  Note that this example code uses the MicroMod Main Board - Double. The MicroMod
  Main Board - Single routes the PWR_EN0 pin slightly different for the
  function board.

  ========== HARDWARE CONNECTIONS ==========
  MicroMod Artemis Processor Board => MicroMod Main Board - Double => Function Boards

  Feel like supporting open source hardware?
  Buy a board from SparkFun!
       MicroMod Artemis Processor Board - https://www.sparkfun.com/products/16401
       MicroMod Main Board - Double - https://www.sparkfun.com/products/18576
       MicroMod Environmental Function Board - https://www.sparkfun.com/products/18632

  LICENSE: This code is released under the MIT License (http://opensource.org/licenses/MIT)

*************************************************************************/

/*Define the power enable pins for the processor board with SDIO_DATA2.
  Depending on the processor board, the Arduino pin may be different.

  Note: Certain Processor Boards like the Artemis have more than one pin available
        to control the Function Board's voltage regulator (e.g. SDIO_DATA2 and G5).
        SAMD51, ESP32, and STM32 Processor Board pins do not have SDIO Data 2, so
        users will need to reference the Processor Pin G5. Below are a few examples. */

//ARTEMIS
#define PWR_EN0 4    //Function Board 0's "PWR_EN0" pin <= MicroMod SDIO_DATA2 => Artemis Proces
sor Board (D4)
#define PWR_EN1 26   //Function Board 1's "PWR_EN1" pin <= MicroMod SDIO_DATA1 => Artemis Proces
sor Board (D26)

//Alternative option that does the same thing. Make sure to just choose one for PWR_EN0 and PWR_
EN1
//#define PWR_EN0 29   //Function Board 0's "PWR_EN0" pin <= MicroMod G5 => Artemis Processor Bo
ard (A29)
//#define PWR_EN1 14   //Function Board 1's "PWR_EN0" pin <= MicroMod G6 => Artemis Processor Bo
ard (D14)
```

```
//TEENSY
//#define PWR_EN0 39   //Function Board 0's "PWR_EN0" pin <= MicroMod SDIO_DATA2 => Teensy Proce
ssor Board (D39)
//#define PWR_EN1 34   //Function Board 1's "PWR_EN1" pin <= MicroMod SDIO_DATA1 => Teensy Proce
ssor Board (D34)

//Alternative option that does the same thing. Make sure to just choose one for PWR_EN0 and PWR_
EN1
//#define PWR_EN0 45   //Function Board 0's "PWR_EN0" pin <= MicroMod G5 => Teensy Processor Boa
rd (45)
//#define PWR_EN1 6   //Function Board 1's "PWR_EN1" pin <= MicroMod G6 => Teensy Processor Boar
d (6)

//Note: The SAMD51, ESP32, and STM32 Processor Board Pins do not have SDIO Data 2 and SDIO Data
 1.

//SAMD51
//#define PWR_EN0 7   //Function Board 0's "PWR_EN0" pin <= MicroMod G5 => SAMD51 Processor Boar
d (D7)
//#define PWR_EN1 8   //Function Board 1's"PWR_EN1" pin <= MicroMod G6 => SAMD51 Processor Board
(D8)

//ESP32
//#define PWR_EN0  32   //Function Board 0's "PWR_EN0" pin <= MicroMod G5 => ESP32 Processor Boa
rd (32)
//#define PWR_EN1  33   //Function Board 1's"PWR_EN1" pin <= MicroMod G6 => ESP32 Processor Boar
d (33)



void setup() {
  // initialize the digital pins as an output.
  pinMode(PWR_EN0, OUTPUT);
  pinMode(PWR_EN1, OUTPUT);

}



void loop() {

  digitalWrite(PWR_EN0, HIGH); // turn the 3.3V regulator on (HIGH is the voltage level)
  digitalWrite(PWR_EN1, HIGH); // turn the 3.3V regulator on (HIGH is the voltage level)
  delay(5000);                 // wait for a few seconds to do something with the function board
s

  digitalWrite(PWR_EN0, LOW);  // turn the 3.3V regulator off by making the voltage LOW
  digitalWrite(PWR_EN1, LOW);  // turn the 3.3V regulator off by making the voltage LOW
  delay(5000);                 // wait for a few seconds before turning function boards back on

}
```

After uploading, take a look at the PWR LEDs on the Function Boards. The LEDs will be on for about 5 seconds and then turn off for another 5 seconds. It will continue to repeat until power is removed from the MicroMod Main Board - Double.

Example 3: Reading and Writing to a MicroSD Card

Note: For the MicroMod Teensy, make sure to check out the SdFat library included with the Teensyduino add-on. You'll need to modify the macro's value for the CS pin for the microSD card from:

```
#define SD_CS_PIN SS
```

To one of the following values.

```
//TEENSY
#define SD_CS_PIN 5 // The microSD Card CS pin is D1 for the MicroMod Main Board - Single a
nd Teensy Processor (5). Adjust for your processor if necessary.
//#define SD_CS_PIN 44 // The microSD Card's CS pin is G4 for the MicroMod Main Board - Dou
ble and Teensy Processor (44). Adjust for your processor if necessary.
```

The example below uses the built-in SD Arduino library. The only difference in the code is the CS pin for the microSD card, which you will need to adjust for your processor board (in this case we use the Artemis Processor Board). Make sure to reference your Processor Board's CS pin for either the Main Board - Single (D1) or Main Board - Double (G4).

Note: The SPI pins will be defined as macros with your processor's board definitions so we won't need to define COPI, CIPO, or CLK pins. Sweet!

If you have not already, select your Board (in this case the **MicroMod Artemis**), and associated COM port. Copy and paste the code below in your Arduino IDE. Hit the upload button and set the serial monitor to **9600** baud.

```
/*
  SD Card Read/Write

  This example shows how to read and write data to and from an SD card file
  The circuit:

  SD card attached to SPI bus as follows:

   SD Card  - MicroMod Artemis Processor Board
   ----------------------------------
   COPI/SDO - pin 38
   CIPO/SDI - pin 43
   CLK      - pin 42
   CS       - pin 1 (Main Board - Single) or 28 (Main Board - Double)

  created   Nov 2010
  by David A. Mellis
  modified 9 Apr 2012
  by Tom Igoe

  This example code is in the public domain.

*/

#include <SPI.h>
#include <SD.h>

//ARTEMIS
//const int SD_CS_PIN = 1; // The microSD Card CS pin is D1 for the MicroMod Main Board - Single and Artemis Processor (D1). Adjust for your processor if necessary.
const int SD_CS_PIN = 28; // The microSD Card's CS pin is G4 for the MicroMod Main Board - Double and Artemis Processor (D28). Adjust for your processor if necessary.

File myFile;

void setup() {
  // Open serial communications and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native USB port only
  }


  Serial.print("Initializing SD card...");

  if (!SD.begin(SD_CS_PIN)) {
    Serial.println("initialization failed!");
    while (1);
  }
  Serial.println("initialization done.");

  // open the file. note that only one file can be open at a time,
  // so you have to close this one before opening another.
```

```
  myFile = SD.open("test.txt", FILE_WRITE);

  // if the file opened okay, write to it:
  if (myFile) {
    Serial.print("Writing to test.txt...");
    myFile.println("testing 1, 2, 3.");
    // close the file:
    myFile.close();
    Serial.println("done.");
  } else {
    // if the file didn't open, print an error:
    Serial.println("error opening test.txt");
  }

  // re-open the file for reading:
  myFile = SD.open("test.txt");
  if (myFile) {
    Serial.println("test.txt:");

    // read from the file until there's nothing else in it:
    while (myFile.available()) {
      Serial.write(myFile.read());
    }
    // close the file:
    myFile.close();
  } else {
    // if the file didn't open, print an error:
    Serial.println("error opening test.txt");
  }
}

void loop() {
  // nothing happens after setup
}
```

If all goes well, you should see the following output if this is the first time writing to the card!

```
Initializing SD card...initialization done.
Writing to test.txt...done.
test.txt:
testing 1, 2, 3.
```

If you are looking to go the extra mile to see if data was saved, close the Serial Monitor and remove power from the MicroMod Main Board. Eject your microSD card from the socket and insert into a microSD card adapter. Then insert the microSD card into your computer's card reader or USB port. Open the **test.txt** file in a text editor. You should see an output similar to what you saw in the Serial Monitor after the file was opened as shown below.

```
testing 1, 2, 3.
```

Besides verifying the data in the file, this step is also useful if you adjust the code to continuously log data in a CSV format. After logging data, you could the open text document in a spreadsheet and graph the values!

More Examples!

Sweet! Now that we know that we can read and write to the microSD card, try exploring the other examples in either the SD or SdFat Arduino libraries. Or check out the following MicroMod tutorials that have a built-in microSD card socket for ideas on data logging. Better yet, try adding a sensor and write some code to log some data!

MicroMod Weather Carrier Board Hookup Guide

A quick guide to help to create your own MicroMod weather station using the MicroMod Weather Carrier Board and Processor of your choice.

MicroMod Data Logging Carrier Board Hookup Guide

Get started with some customizable MicroMod data logging with the Data Logging Carrier Board.

MicroMod Asset Tracker Carrier Board Hookup Guide

Get started with the SparkFun MicroMod Asset Tracker Carrier Board following this Hookup Guide. The Asset Tracker uses the u-blox SARA-R510M8S LTE-M / NB-IoT module to provide a host of data communication options.

Looking for more examples with a Function Board? Below are a few Function Board examples from our tutorials that are tagged with MicroMod.

1W LoRa MicroMod Function Board

Everything you need to get started with the 1W LoRa MicroMod function board; a MicroMod function board that provides LoRa capabilities for your MicroMod

MicroMod WiFi Function Board - ESP32 Hookup Guide

The MicroMod ESP32 Function Board adds additional wireless options to MicroMod Processor Boards that do

project. Must be used in conjunction with a MicroMod main board and processor.

not have that capability. This special function board acts as a coprocessor that takes advantage of Espressif's ESP32 WROOM to add WiFi and Bluetooth® to your applications.

MicroMod Environmental Function Board Hookup Guide

The SparkFun MicroMod Environmental Function Board adds additional sensing options to the MicroMod Processor Boards. This function board includes three sensors to monitor air quality (SGP40), humidity & temperature (SHTC3), and CO2 concentrations (STC31) in your indoor environment. To make it even easier to use, all communication is over the MicroMod's I2C bus! In this tutorial, we will go over how to connect the board and read the sensors.

MicroMod WiFi Function Board - DA16200 Hookup Guide

Add IoT functionality to any MicroMod project with the MicroMod WiFi function Board - DA16200!

Better yet, try connecting a Qwiic-enabled device to the Main Board's Qwiic connector. Below are a few examples from our tutorials that are tagged with Qwiic.

Qwiic HAT for Raspberry Pi Hookup Guide

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

SparkFun SAMD21 Pro RF Hookup Guide

Using the super blazing, nay blinding, fast SAMD21 whipping clock cycles at 48MHz and the RFM96 module to connect to the Things Network (and other Radio woodles).

Qwiic UV Sensor (VEML6075) Hookup Guide

Learn how to connect your VEML6075 UV Sensor and figure out just when you should put some sunscreen on.

Qwiic SHIM Kit for Raspberry Pi Hookup Guide

Get started with the Serial LCD with RGB backlight and 9DoF IMU (ICM-20948) via I2C using the Qwiic system and Python on a Raspberry Pi! Take sensor readings and display them in the serial terminal or SerLCD.

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: MicroMod 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.

SPARKFUN FORUMS: MICROMOD

Resources and Going Further

Now that you've successfully got your MicroMod Main Board with a Processor Board, it's time to incorporate it into your own project! For more information, check out the resources below:

- MicroMod Main Board - Single
  - Schematic (PDF)
  - Eagle Files (ZIP)
  - Board Dimensions (PNG)
  - GitHub Hardware Repo
- MicroMod Main Board - Double
  - Schematic (PDF)
  - Eagle Files (ZIP)
  - Board Dimensions (PNG)
  - GitHub Hardware Repo
- SFE Product Showcase

Looking for more inspiration? Check out these other tutorials related to MicroMod.

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!

MicroMod STM32 Processor Hookup Guide

Get started with the MicroMod Ecosystem and the STM32 Processor Board!

MicroMod WiFi Function Board - ESP32 Hookup Guide

The MicroMod ESP32 Function Board adds additional wireless options to MicroMod Processor Boards that do not have that capability. This special function board acts as a coprocessor that takes advantage of Espressif's ESP32 WROOM to add WiFi and Bluetooth® to your applications.

MicroMod Environmental Function Board Hookup Guide

The SparkFun MicroMod Environmental Function Board adds additional sensing options to the MicroMod Processor Boards. This function board includes three sensors to monitor air quality (SGP40), humidity & temperature (SHTC3), and CO2 concentrations (STC31) in your indoor environment. To make it even easier to use, all communication is over the MicroMod's I2C bus! In this tutorial, we will go over how to connect the board and read the sensors.