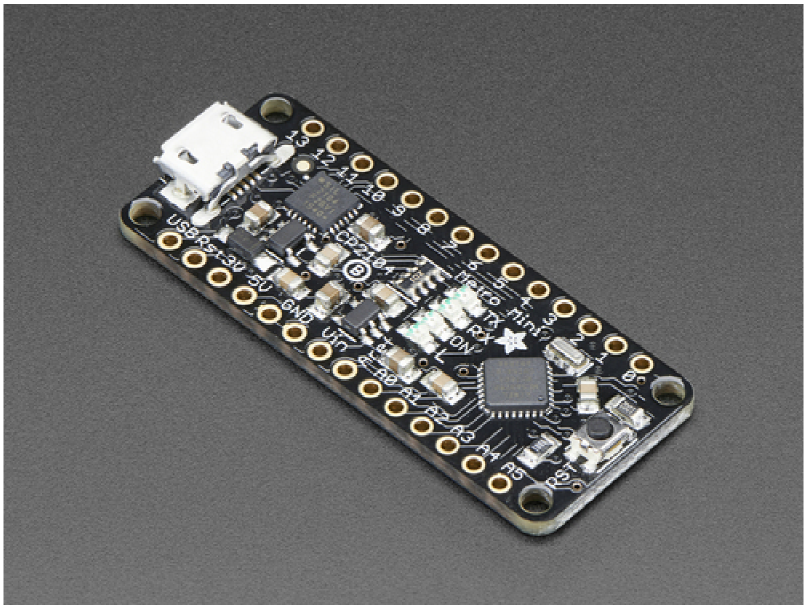
adafruit learning system

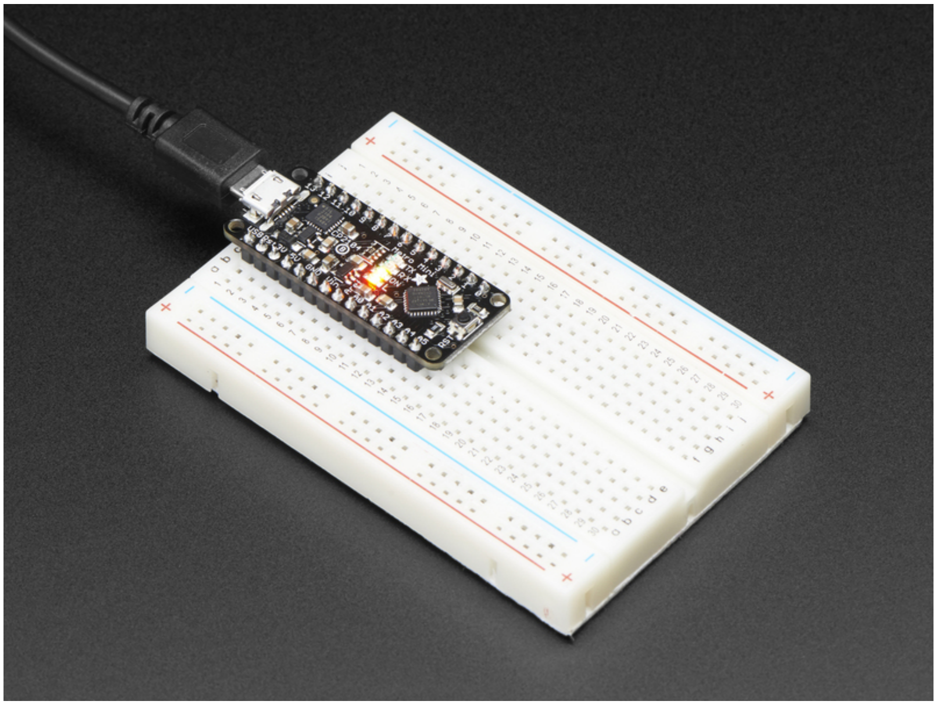
Adafruit Metro Mini

Created by lady ada

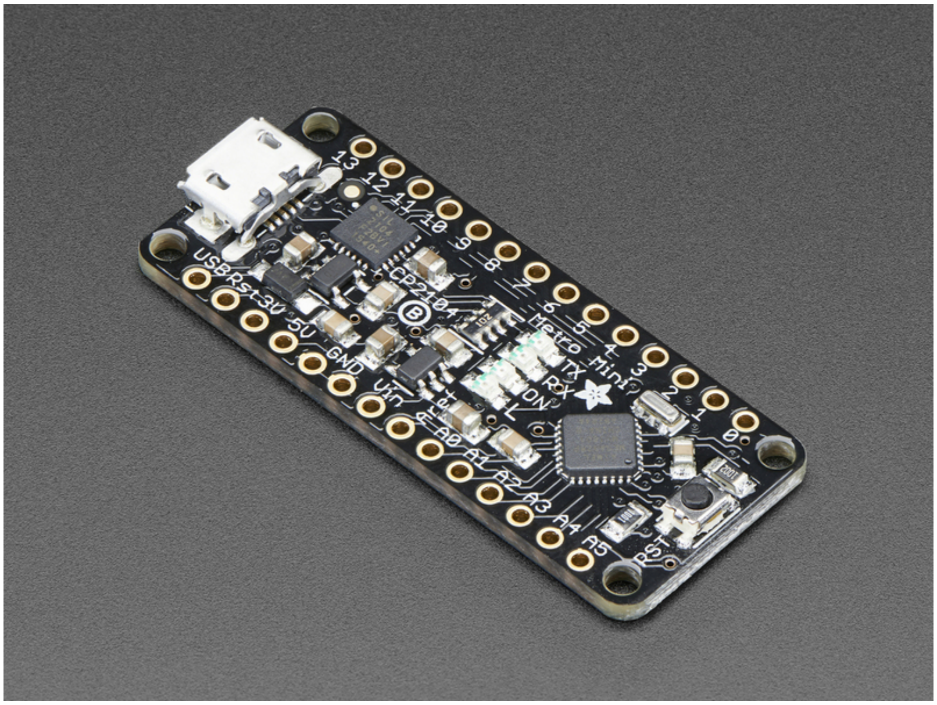


Last updated on 2019-03-29 04:29:04 AM UTC

Overview



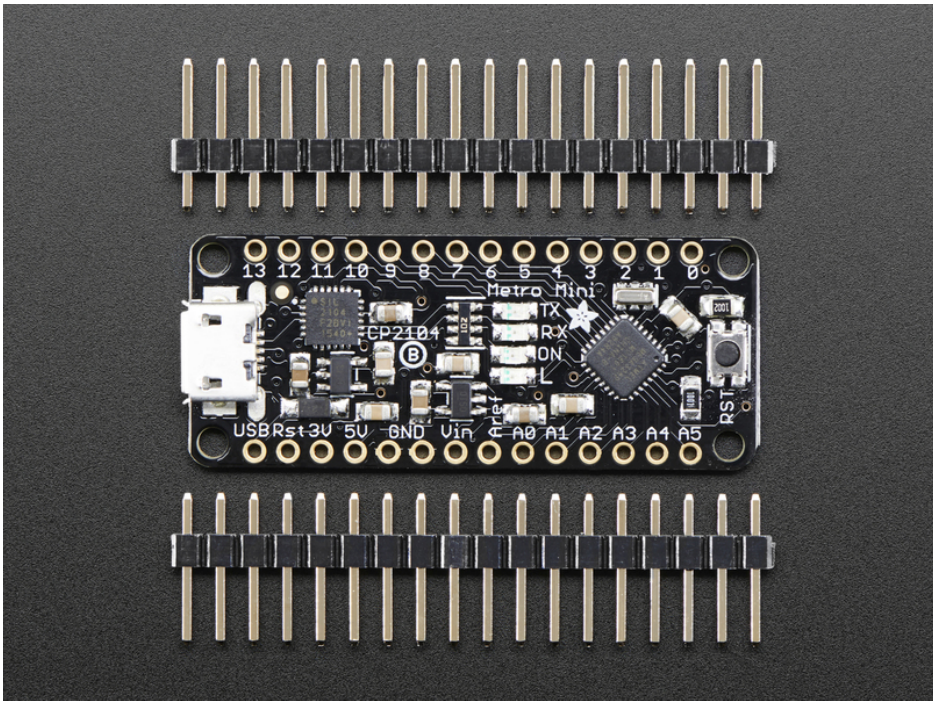
We sure love the ATmega328 here at Adafruit, and we use them *a lot* for our own projects. The processor has plenty of GPIO, Analog inputs, hardware UART SPI and I2C, timers and PWM galore - just enough for most simple projects. When we need to go small, we use a Pro Trinket 3V (http://adafru.it/2010) or 5V (https://adafru.it/dUf), but if you want to have USB-to-Serial built in, we reach for an **Adafruit METRO Mini**.



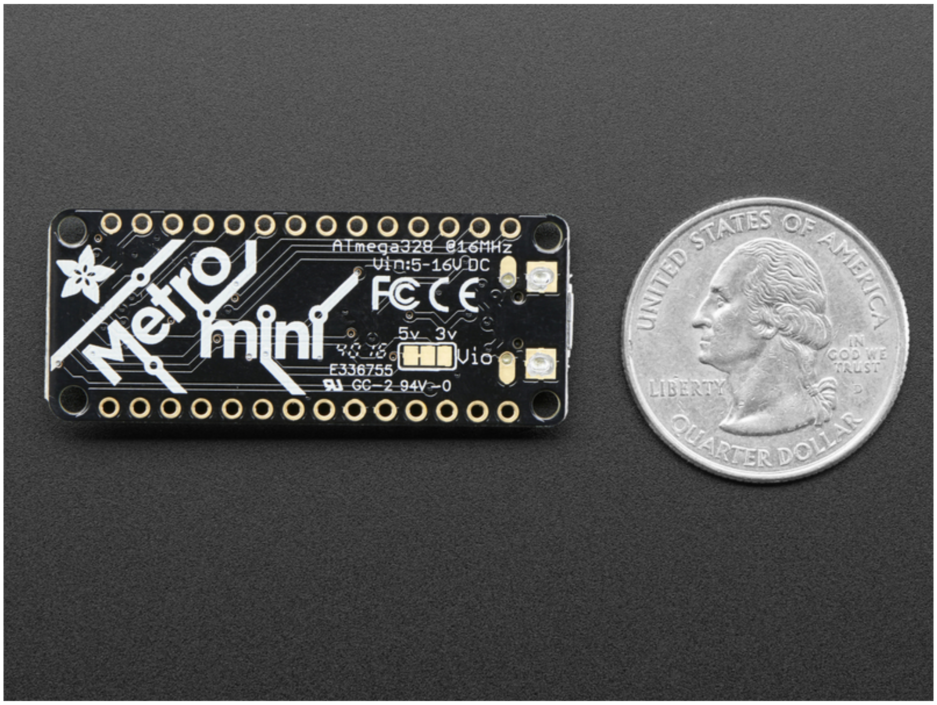
METRO Mini is the culmination of years of playing with AVRs: we wanted to make a tiny, breadboard-friendly development board that is easy to use and is hacker friendly. Metro Mini can be programmed with the Arduino IDE

(select 'UNO' in the boards dropdown)

- **ATmega328 brains** - This popular chip has 32KB of flash (1/2 K is reserved for the bootloader), 2KB of RAM, clocked at 16MHz
- **Power the METRO Mini** with 6-16V polarity protected on the Vin pin, or plug the micro USB connector to any 5V USB source.
- **METRO has 20 GPIO pins**, 6 of which are Analog in as well, and 2 of which are reserved for the USB-serial converter. There's also 6 PWMs available on 3 timers (1 x 16-bit, 2 x 8-bit). There's a hardware SPI port, hardware I2C port and hardware UART to USB.
- **GPIO Logic level is 5V** but by cutting and soldering closed a jumper on the bottom, you can easily convert it to 3.3V logic
- **5V onboard regulator** with 150mA out, 3.3V 50mA available via FTDI chip
- **USB to Serial converter**, there's a genuine SiLabs CP2104 hardware USB to Serial converter that can be used by any computer to listen/send data to the METRO, and can also be used to launch and update code via the bootloader
- **Four indicator LEDs**, on the top of the PCB, for easy debugging. One green power LED, two RX/TX LEDs for the UART, and a red LED connected to pin PB5 / digital #13
- **Easy reprogramming**, comes pre-loaded with the Optiboot bootloader (https://adafru.it/wbs)
- **Beautiful styling** by PaintYourDragon and Bruce Yan, in Adafruit Black with gold plated pads.



The Metro Mini comes as a fully assembled and tested board, with bootloader burned in and also a stick of 0.1" header. Some light soldering is required if you'd like to plug it into a breadboard, or you can solder wires or header directly to the breakout pads. Once headers are installed they can be fitted into 0.6" wide sockets (https://adafru.it/ync)

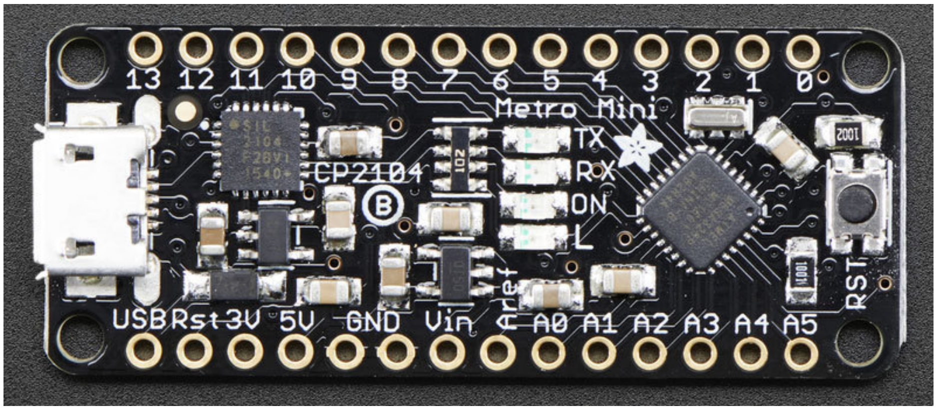


Detailed specifications:

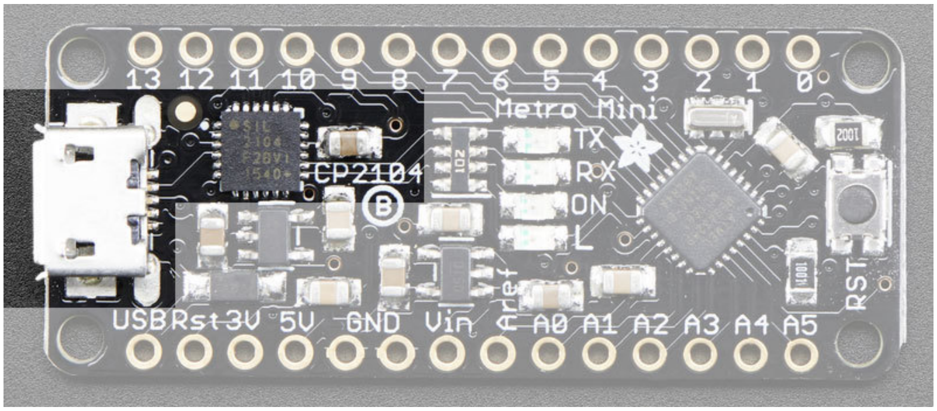
- ATmega328 microcontroller with Optiboot (UNO) Bootloader
- USB Programming and debugging via the well-supported genuine FTDI FT231X or the SiLabs CP2104
- Input voltage (Vin): 6-16V (a 9VDC power supply is recommended)
- 5V logic with 3.3V compatible inputs, can be converted to 3.3V logic operation
- 20 Digital I/O Pins: 6 are also PWM outputs and 6 are also Analog Inputs
- 32KB Flash Memory - 0.5K for bootloader, 31.5KB available after bootloading
- 16MHz Clock Speed
- Adafruit Black PCB with gold plate on pads
- 18mm x 44mm x 4mm / 0.7" x 1.7" x 0.2"
- Weight: 3g
- Derivative of "Arduino UNO R3 Reference design" (https://adafru.it/ynd)
- Open source hardware files on github! (https://adafru.it/yne)

Pinouts

We packed a full Metro's worth of goodies into this slim little PCB. There's everything you need to build a project in Arduino - lets check it out!

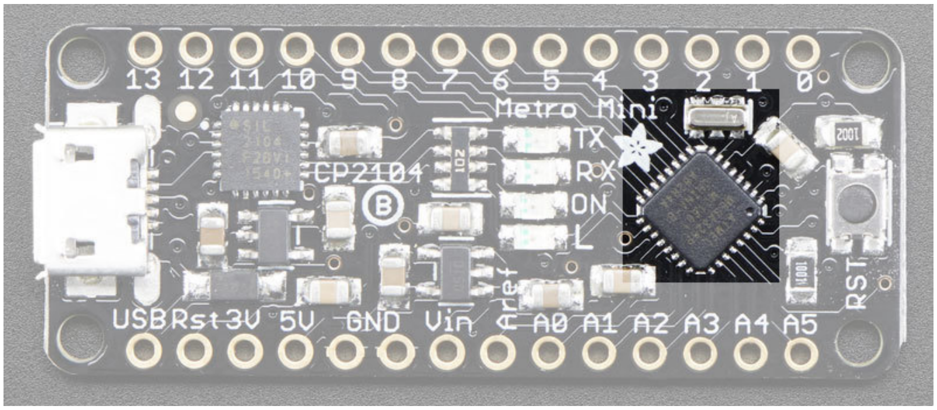


USB & Serial converter



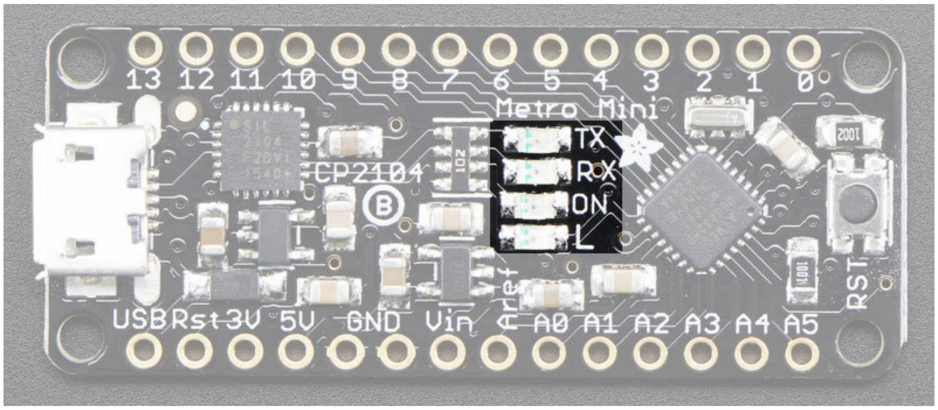
The USB plug is how you will need to program and debug your code. We use it to both power your project and also provide the USB serial interface. Check the **Arduino IDE Usage** page for how to install the drivers.

Microcontroller & Crystal



This is the brains of your Metro Mini. We use the ATmega328P at 16 MHz. This matches the same exact chip used in the Adafruit Metro and Arduino Uno, so you know that you'll have 100% compatibility.

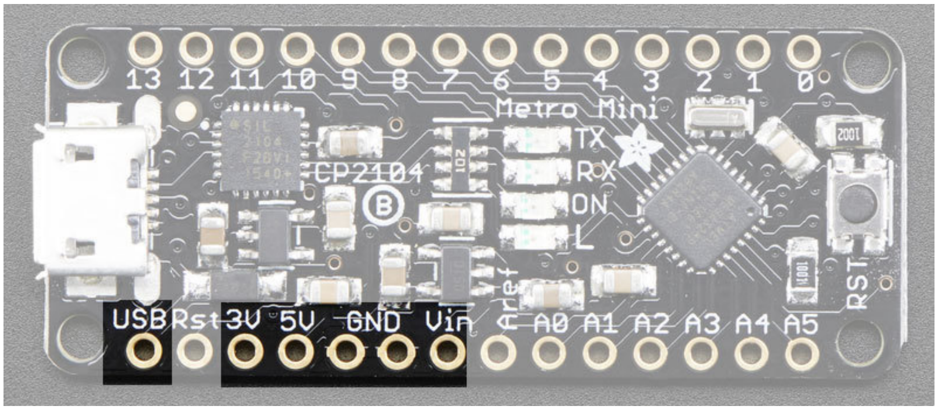
LEDs



There are 4 LEDs in the middle of the board. In no particular order:

- **ON** is a green LED that is always lit when the board is powered, it is connected directly to the 5V power supply through a 1K resistor
- **L** is the red 'pin 13' LED and is connected to (you guessed it) pin #13 through a 1K resistor
- **RX** and **TX** are controlled by the USB serial chip. They will let you know when data is sent from or to the computer. Both are yellow. They are driven by the USB serial chip only, you cannot control them directly.

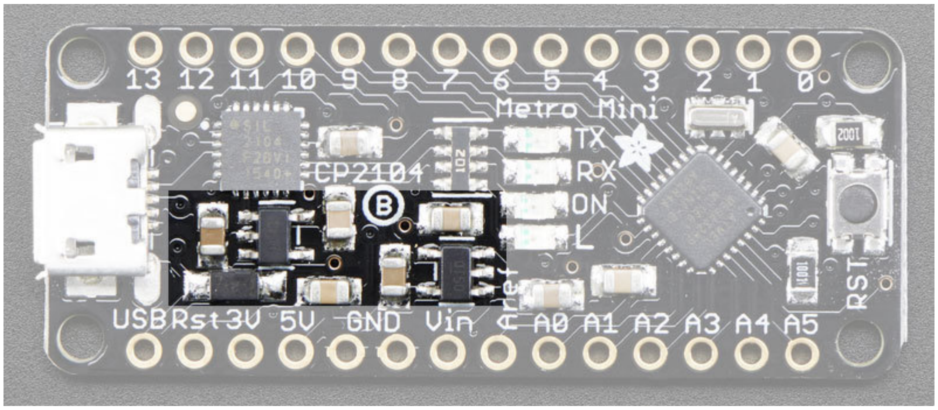
Power Pins & Regulators



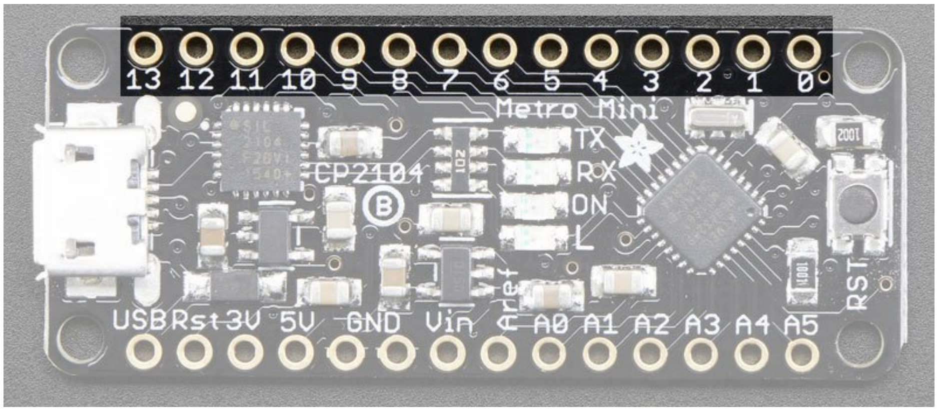
The bottom row has 6 power pins that you will find handy!

- **USB** is connected directly to the micro USB connector, and is great when you need a lot of 5V power for NeoPixels or servos, etc. If USB is not plugged in, this will not be powered
- **3V** this is the 3V regulator ouput. On the FTDI-using Metro Mini, this will be 50mA output. On the CP2104-using Metro Mini it will be 500mA max output. Note you can only get 500mA out when its plugged into USB
- **5V** is the 5V output. If powered from USB, it is powered from USB power through a Schottkey diode. If you are only powering from Vin, it will give you up to 150mA of current out
- **Vin** is the battery input. You can provide 5V to 16V input. Note that we linearly regulate this down, so using a higher voltage will cause your board to heat up *more*. 5-8V is a good place to stay.

You can connect both USB and Vin power, and the metro will automatically switch between the two to maintain the highest voltage output on 5V out.



Digital I/O Pins



On the right hand side are all the digital I/O pins you expect. These match the same names/usage of the Adafruit Metro and Arduino UNO. They are all 5V logic output/input unless you've set the Mini to be 3V logic (see at the end)

The two pins labeled **0** and **1** are the two Serial pins that are used to send data to and from the Arduino to the USB-Serial translator chip (https://adafru.it/oBC).

Don't connect anything to Digital 0 or 1 unless you are super sure because it will affect your Metro's ability to communicate!

- **Digital 2** through **Digital 12** are normal every day digital pins. Pins # 3, 5, 6, 9, 10, 11 can do PWM output
- **Digital 13** is a little special because it is also connected to the **L LED**. (https://adafru.it/oBD) You can use this pin without affecting the Arduino just be aware that the **L** LED will also blink at the same time.

SPI Port

The ATmega328p has a hardware SPI port, but you can only use the following pins:

- **SPI Clock** on D13
- **SPI MISO** on D12
- **SPI MOSI** on D11

The default chip select pin is D10. If you aren't using that pin for CS, it has to be a digital output. So we recommend using that pin for your first SPI-device CS

I2C Port

The ATmega328p has a hardware I2C port, but you can only use the following pins:

- **I2C Data** on A4
- **I2C Clock** on A5

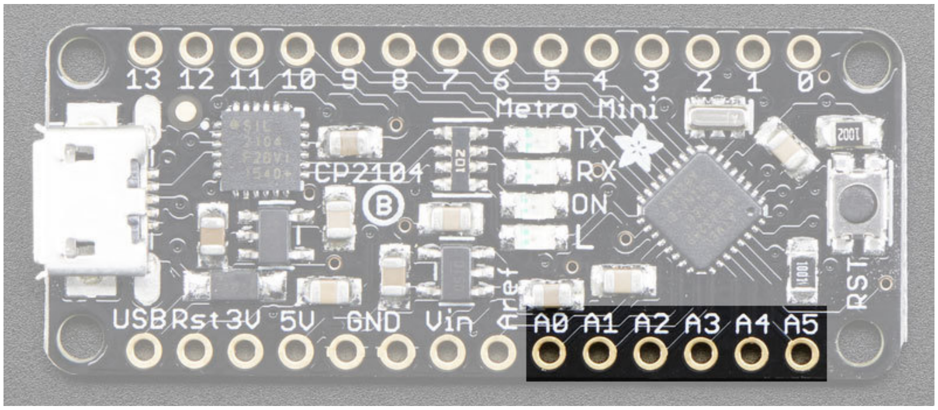
UART Port

The ATmega328p has a hardware I2C port, **but it is used for the Serial connection and programming the chip bootloader so it isn't recommended!**

- **UART TX** on D0
- **UART RX** on D1

For connecting to Serial-needing devices we recommend SoftwareSerial which can use any two pins

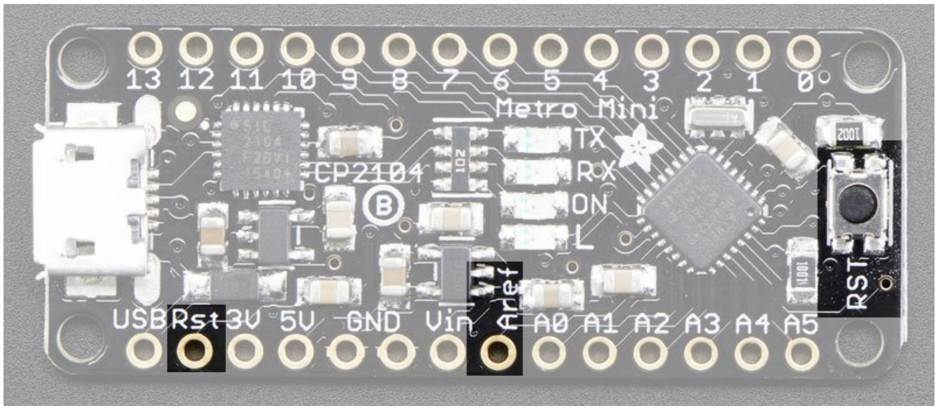
Analog In + Digital I/O Pins



The analog input pins are special pins that can read sensors. They can also be used as digital input/output pins, they really are the most versatile pins!

Each analog pin can read a voltage between 0 and 5V. When used in digital mode, they are all 5V logic output/input. Unless you've set the Mini to be 3V logic (see at the end) in which case the analog read-range and logic level is 3.3V

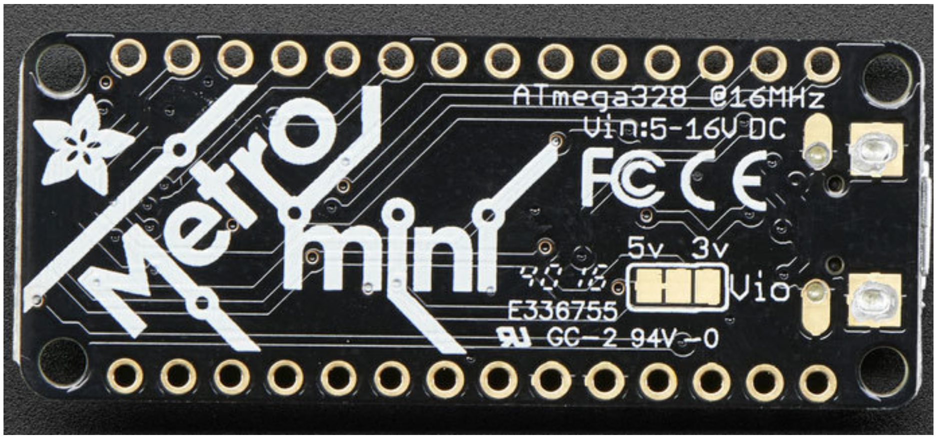
Other Pins



There's a **Reset** pin and button, they are connected together. When **RST** is connected to ground, it will reset your Metro Mini. Note it will not reset the serial connection so you can keep that open when resetting. The button will do the same as connecting **RST** to ground, but its fun to press.

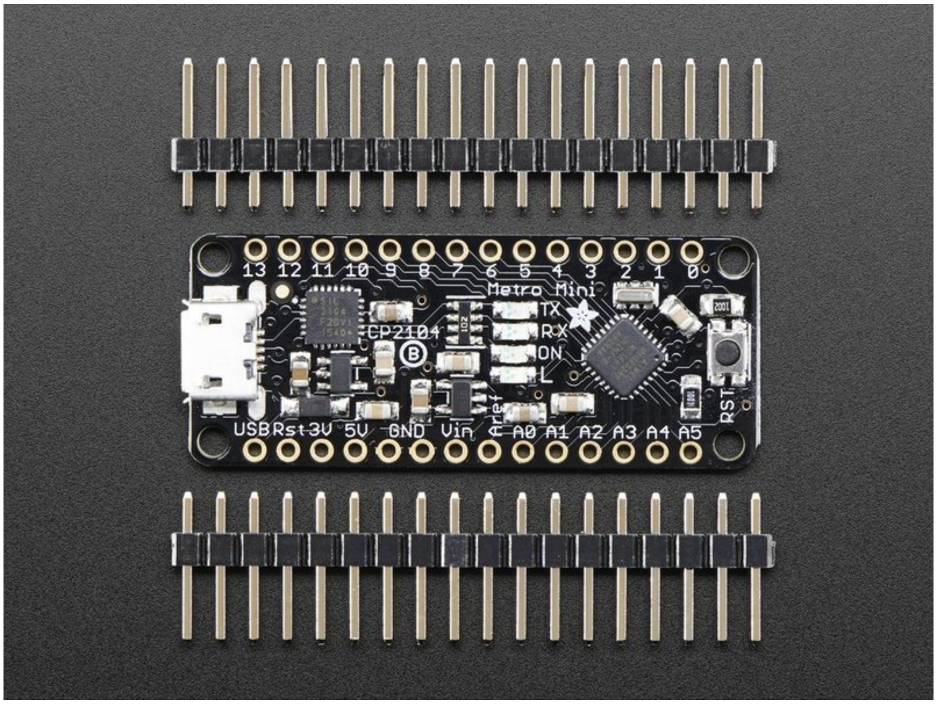
Aref is the analog reference, which can be connected to other voltages when you want to change the range or quality of your analog readings.

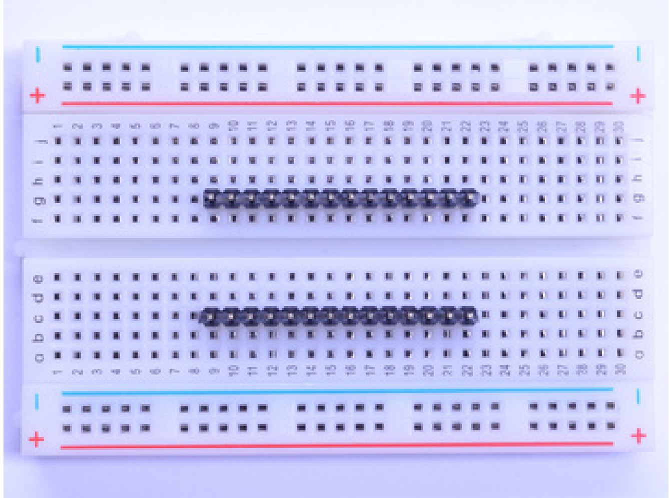
Voltage Select Jumper



On the bottom of the board, there is a **Vio** jumper, which you can see ties the middle pad to **5V**. If you want to use your Metro Mini in 3V logic mode - which can really simplify connecting to many chips, sensors, and displays - cut the little trace to the left, and solder the middle pad and right pad together. Use a multimeter to check that you really cut that trace, as shorting 3V and 5V together is very bad and may damage your Mini!

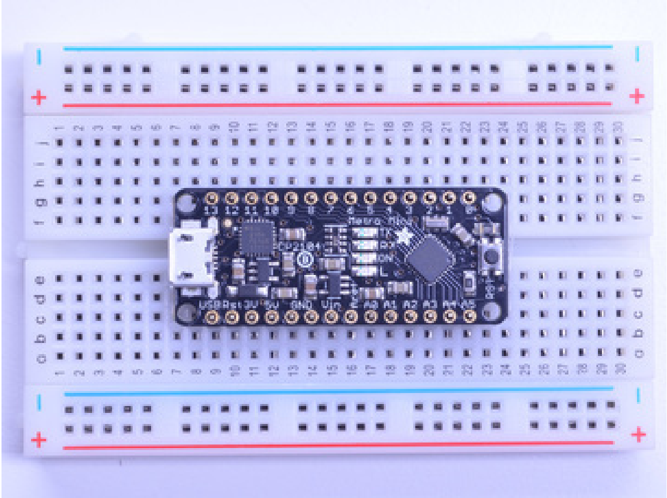
Assembly





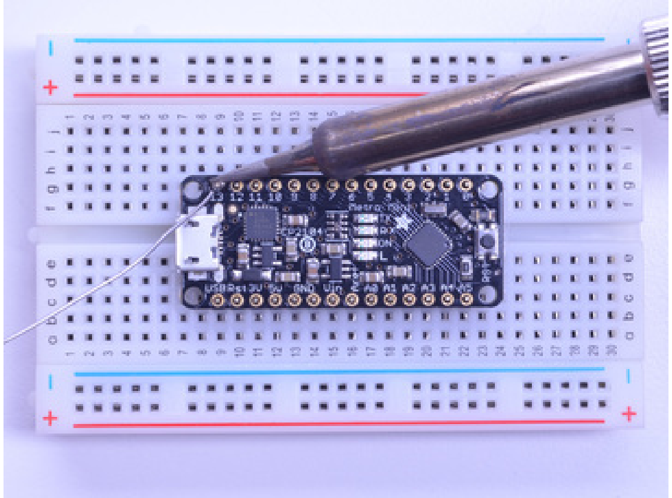
Prepare the header strip:

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



Add the board:

Place the board over the pins so that the short pins poke through the pads

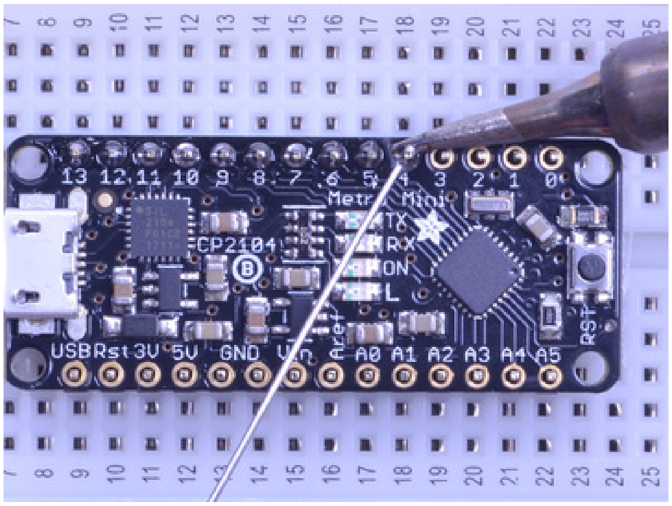


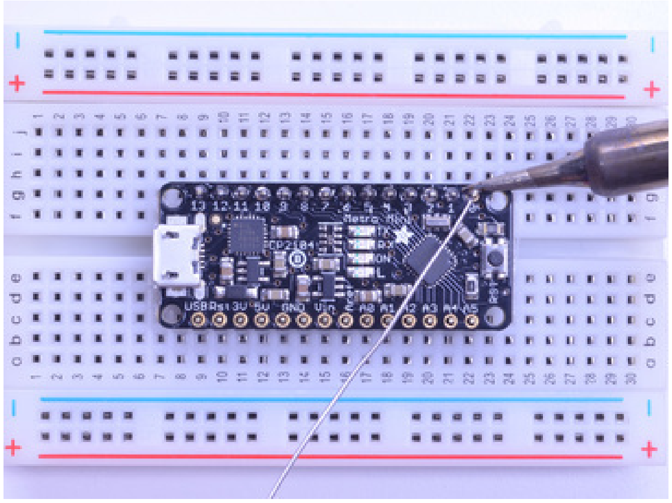
And Solder!

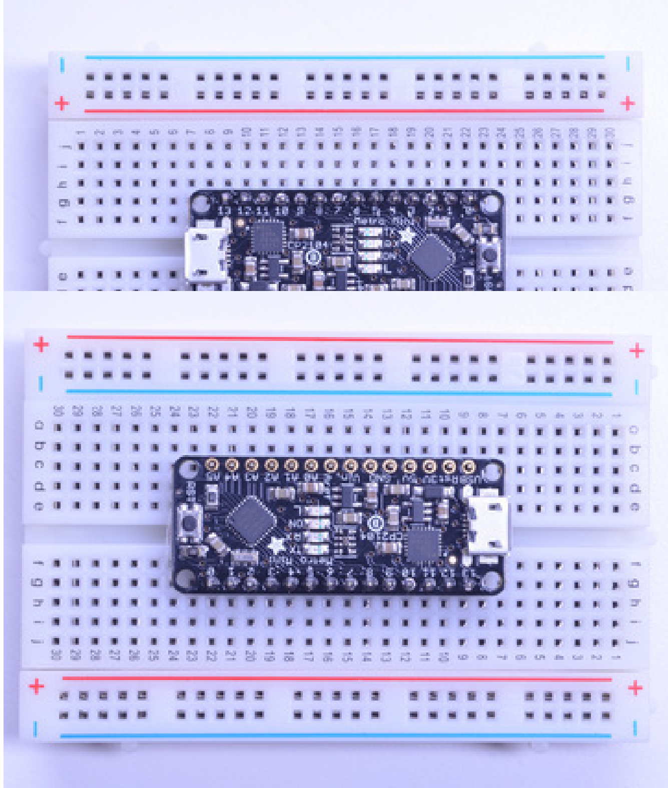
Be sure to solder all pins for reliable electrical contact.

Solder one side of the board at first

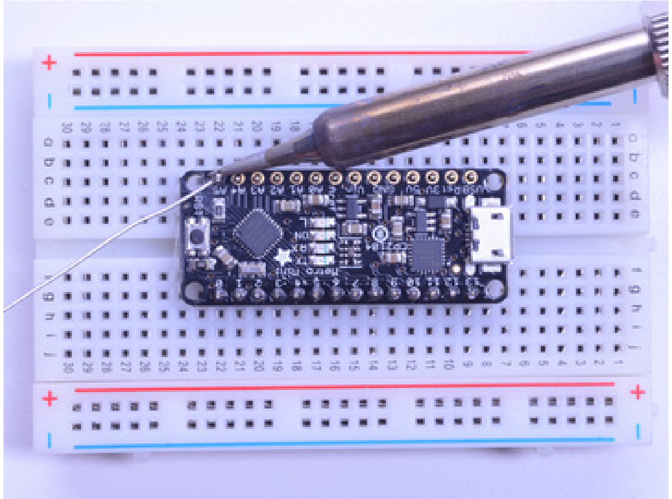
(For tips on soldering, be sure to check out our [Guide to Excellent Soldering](https://adafru.it/aTk) (https://adafru.it/aTk)).

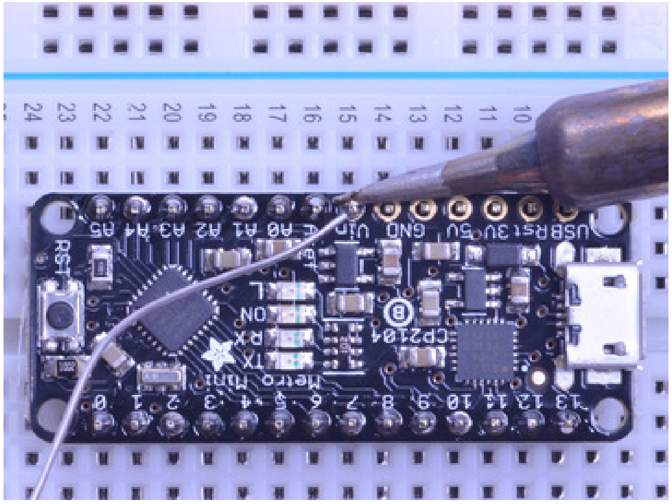


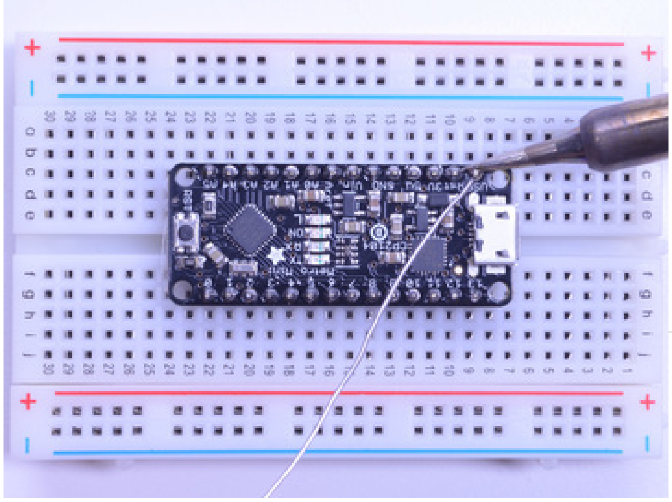


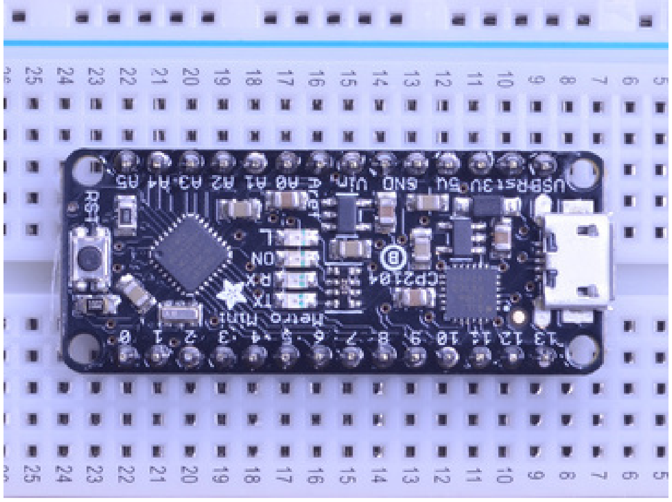


Flip the breadboard around to solder the other strip




A5 A4 A3 A2 A1 A0 Vin GND 5V 3V Rst USB
RST
L
ON
RX
TX
CP2104
Metro Mini
0 1 2 3 4 5 6 7 8 9 10 11 12 13





You're done! Check your solder joints visually and enjoy!

Arduino IDE Usage

Installing USB Serial Drivers

The USB plug is how you will need to program and debug your code. We use it to both power your project and also provide the USB serial interface. On older Metro Mini's we used a FT231X chip. On newer Metro Mini's we now have the CP2104 chip. Both act the same but the CP2104 is a little less expensive so we were able to lower the price. For either one, you'll need to install drivers if you are using Mac or Windows. Linux comes with both drivers already.

Windows users can try plugging the board in, Windows update will probably install drivers for you. If the device is not recognized, simply download and install the Adafruit Windows Driver package. When you run it, make sure to click that you want to install the SiLabs CP210x and FTDI chip drivers too!

https://adafru.it/AB0

https://adafru.it/AB0

If you are running **Mac OS X , please use the SiLabs drivers** (https://adafru.it/yfA) here

> If you are using Mac OS 10.12.6 (Sierra) and you cannot upload with the latest Mac OS VCP driver, please try the legacy v4 driver below. Note you will need to uninstall the v5 driver using uninstall.sh (in the driver package)

https://adafru.it/ymF

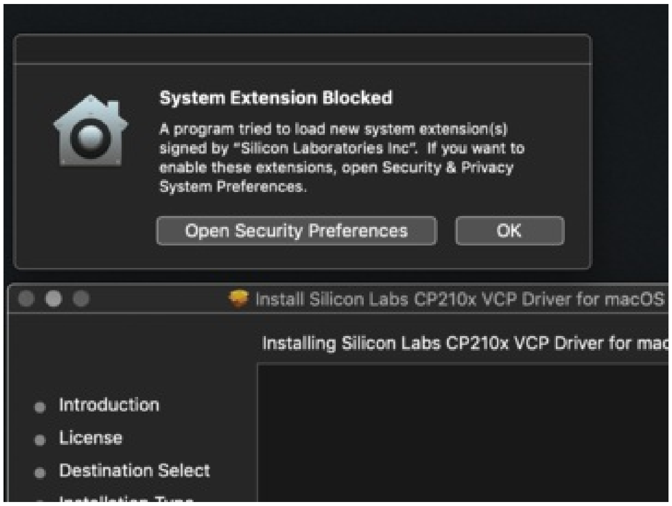
https://adafru.it/ymF

Once drivers are installed and you've rebooted, you will have a Serial COM port.

- How to find your COM Port on Windows (https://adafru.it/ynf)
- How to find your Serial device on Mac OS X (https://adafru.it/ynA)
- How to find your Serial device on Linux (https://adafru.it/ynB)

Driver Installation on Mac OS X

The latest versions of Mac OS X may complain when you install the driver that it doesn't have permission.



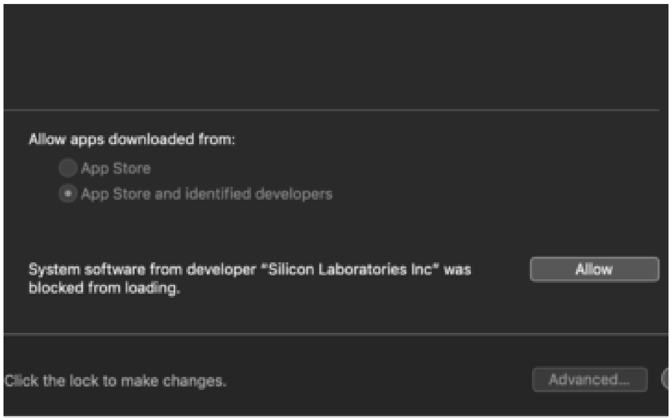
System Extension Blocked

A program tried to load new system extension(s) signed by "Silicon Laboratories Inc". If you want to enable these extensions, open Security & Privacy System Preferences.

Open Security Preferences
OK
Install Silicon Labs CP210x VCP Driver for macOS
Installing Silicon Labs CP210x VCP Driver for mac
Introduction
License
Destination Select

The latest versions of Mac OS X may complain when you install the driver that the system extension was blocked

Click **Open Security Prefs**



Allow apps downloaded from:

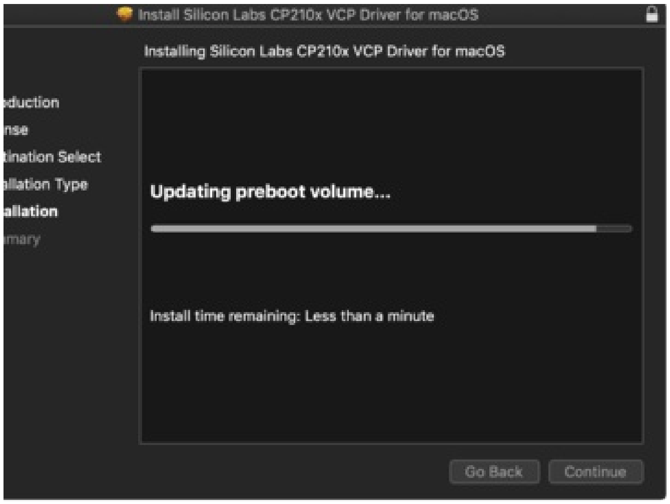
○ App Store

● App Store and identified developers

System software from developer "Silicon Laboratories Inc" was blocked from loading.

Click the lock to make changes.

Click **Allow** next to System Software from "Silicon Laboratories Inc"



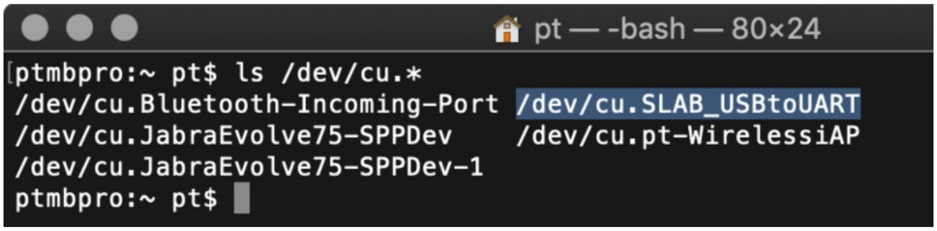
Install Silicon Labs CP210x VCP Driver for macOS

Installing Silicon Labs CP210x VCP Driver for macOS

Updating preboot volume...

Install time remaining: Less than a minute

When complete, check for /dev/cu.SLAB_USBtoUART by typing ls /dev/cu.* into Terminal

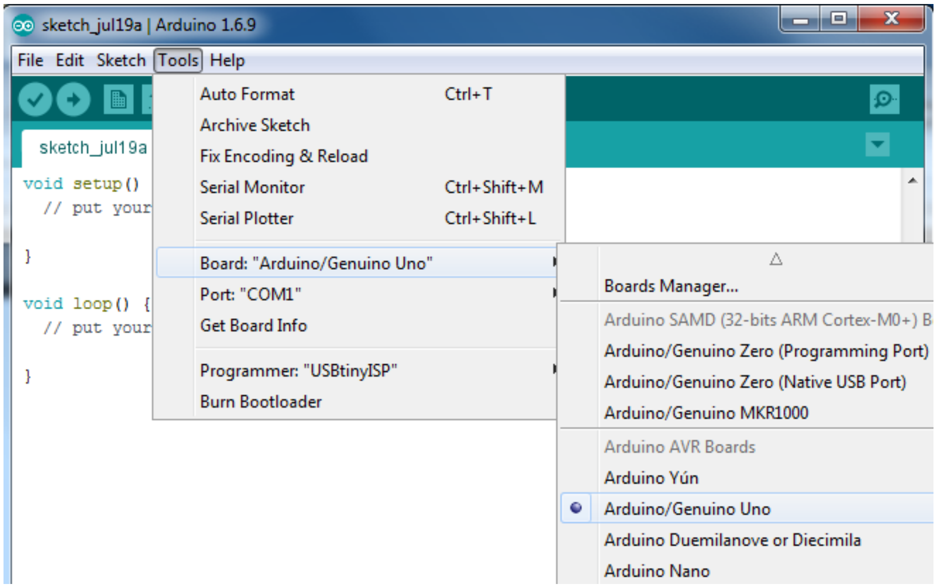


```
ptmbpro:~ pt$ ls /dev/cu.*
/dev/cu.Bluetooth-Incoming-Port /dev/cu.SLAB_USBtoUART
/dev/cu.JabraEvolve75-SPPDev    /dev/cu.pt-WirelessiAP
/dev/cu.JabraEvolve75-SPPDev-1
ptmbpro:~ pt$
```

Using the Arduino IDE

The Adafruit Metro Mini is an 'Arduino Compatible' - that means that when using the Arduino IDE or other Arduino-friendly development environments, you can simply treat the 'Mini like an Arduino UNO.

When programming, simply select **Arduino UNO** in the dropdown menu!



sketch_jul19a | Arduino 1.6.9
File
Edit
Sketch
Tools
Help
sketch_jul19a
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
Get Board Info
Burn Bootloader
Boards Manager...
Arduino SAMD (32-bits ARM Cortex-M0+) B
Arduino/Genuino Zero (Programming Port)
Arduino/Genuino Zero (Native USB Port)
Arduino/Genuino MKR1000
Arduino AVR Boards
Arduino Yún
Arduino Duemilanove or Diecimila
Arduino Nano

```
void setup()
  // put your

}

void loop() {
  // put your

}
```

Board: "Arduino/Genuino Uno"

Port: "COM1"

Programmer: "USBtinyISP"

● Arduino/Genuino Uno

https://adafru.it/rcc

https://adafru.it/rcc

https://adafru.it/ja3

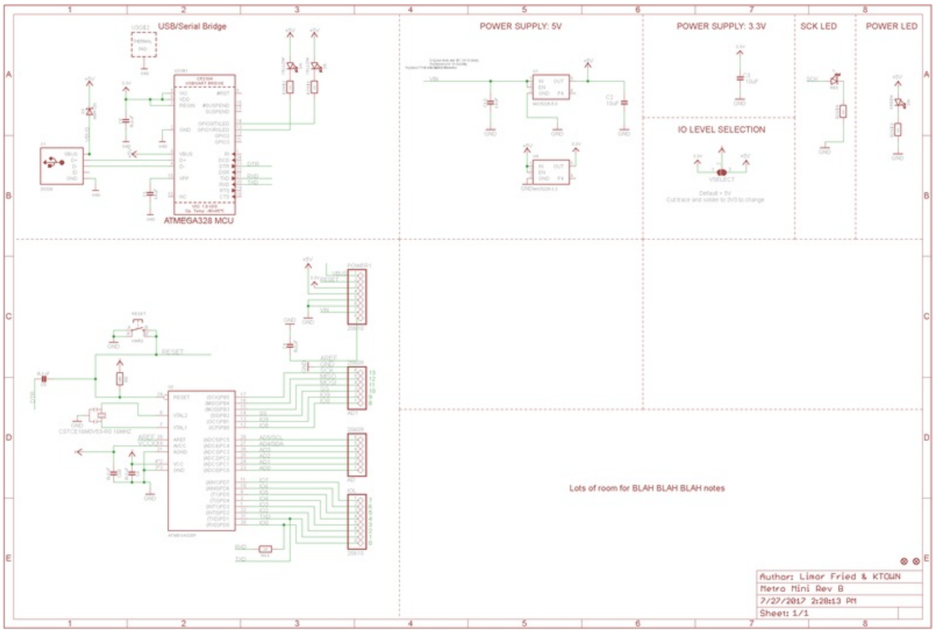
https://adafru.it/ja3

Downloads

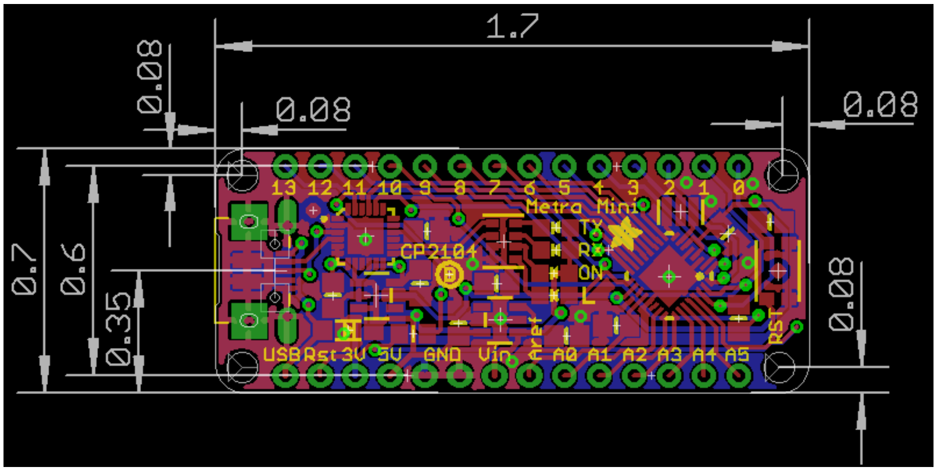
Files

- Derivative of "Arduino UNO R3 Reference design" (https://adafru.it/ynd)
- Open source hardware files on github! (https://adafru.it/yne)
- Fritzing object in the Adafruit Fritzing Library (https://adafru.it/aP3)
- ATmega328P product page (https://adafru.it/ynD)

Schematic



Fabrication Print


1.7
0.08
0.08
0.08
0.7
0.6
0.35
0.08
13 12 11 10 9 8 7 6 5 4 3 2 1 0
Metro Mini
CP2104
TX
RX
ON
L
USB Rst 3V 5V GND Vin Aref A0 A1 A2 A3 A4 A5
RST