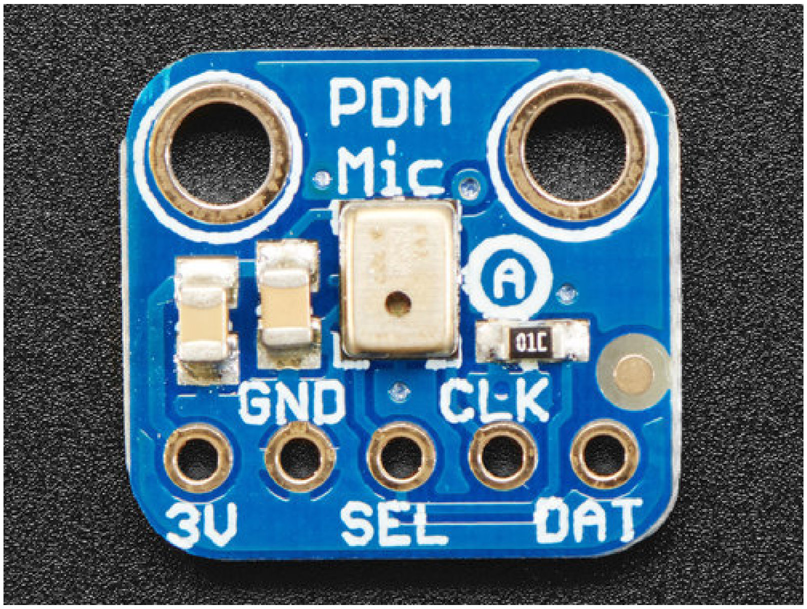
adafruit learning system

# Adafruit PDM Microphone Breakout

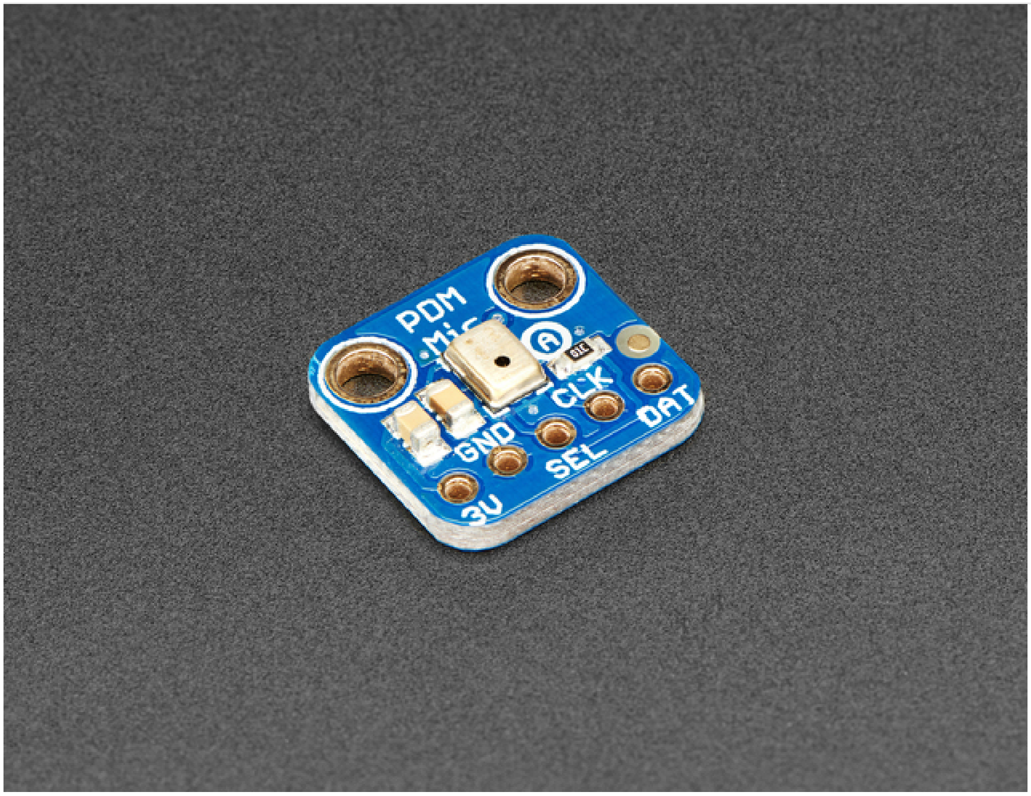
Created by lady ada



Last updated on 2018-01-10 10:25:53 PM UTC

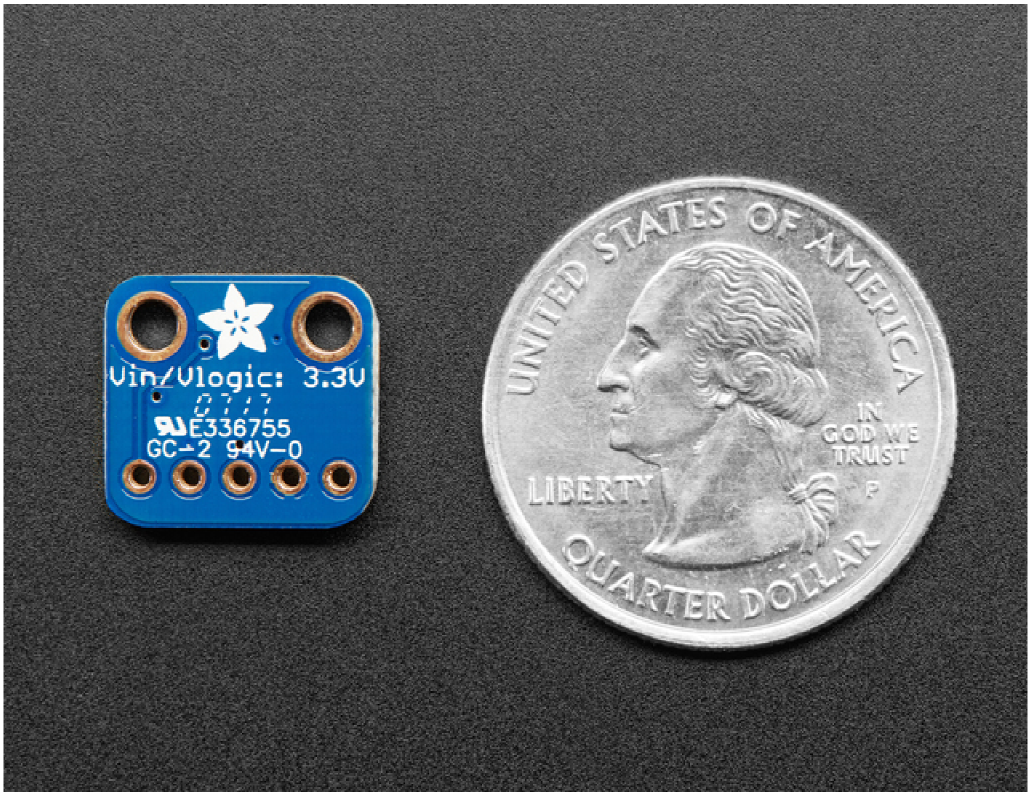
# Guide Contents

# Overview

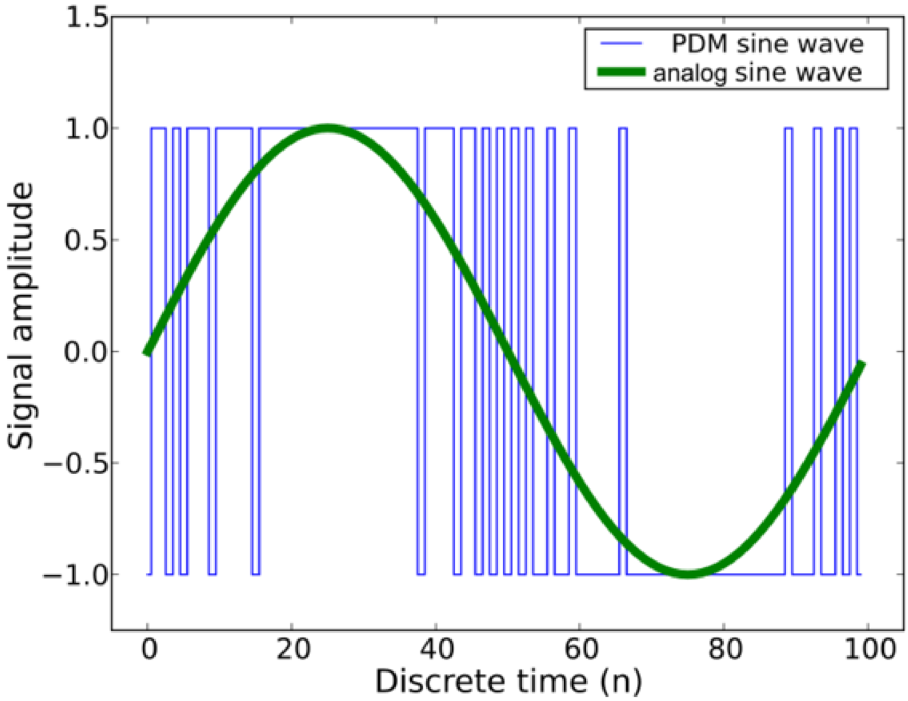


An exotic new microphone has arrived in the Adafruit shop, a **PDM MEMS Microphone**! PDM is the 'third' kind of microphone you can integrate with electronics, apart from analog or I2S. These microphones are very commonly used in products, but are rarely seen in maker projects. Still, they have some benefits so we thought we'd offer a breakout for the shop.

The first thing to note is that this sensor does not provide an 'analog' output like many of our electret microphone assemblies. So it's great for chips that do not have analog inputs. Secondly, the digital interface is a very simplistic **p**ulse **d**ensity **m**odulation output. It's digital but its *not* PWM and it's *not* I2S. You will need to make sure your chip has a PDM interface - most 32-bit processors these days do!



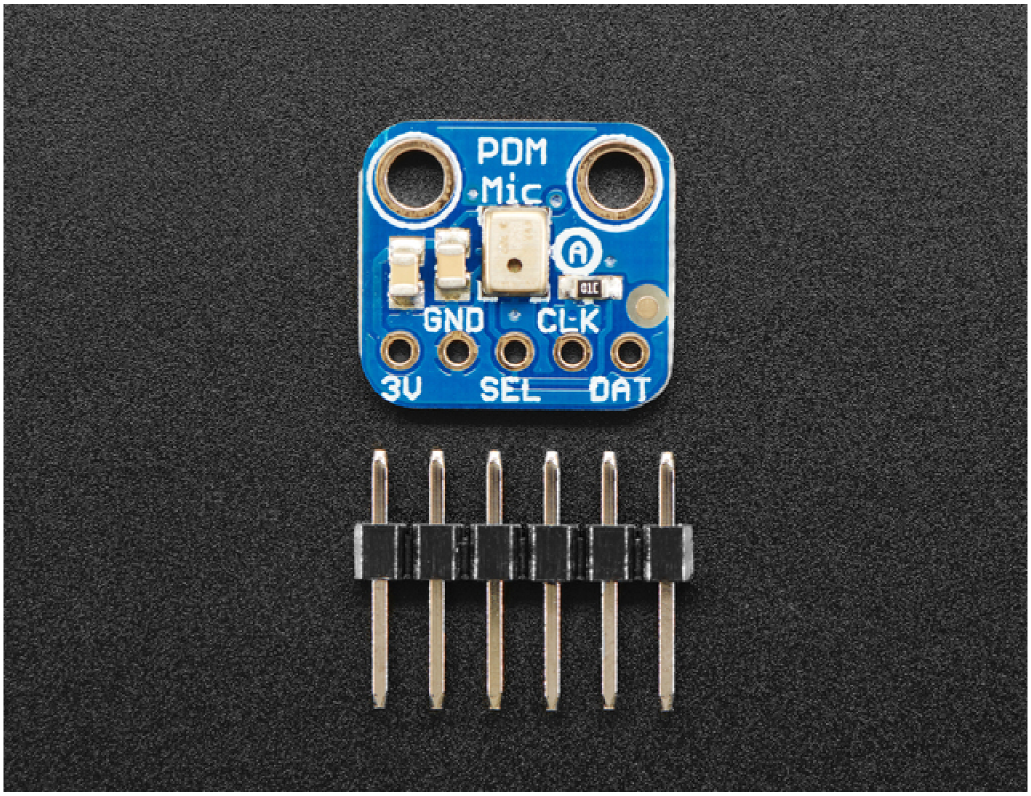
PDM is a little like 1-bit PWM. You clock the mic with a 1 MHz - 3 MHz clock rate, and on the data line you'll get a square wave out that syncs with the clock. The data line with be 0 or 1 logic output, with the square wave creating a *density* that when averaged will result in the analog value out.



There's a few ways to manage these mics:

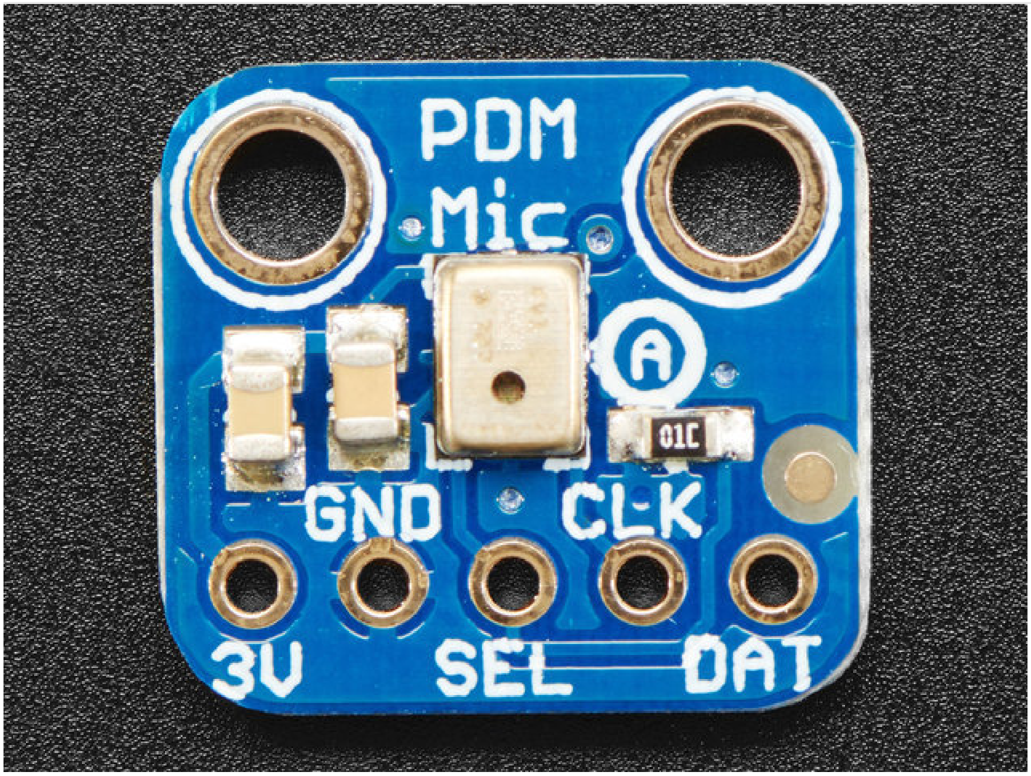
- Your chip comes with a hardware peripheral and library that does all the data managing at high speed, collects samples, applies a filter and gives you an analog value (Ideal!)
- Your chip comes with a hardware peripheral that gives you values, then it is up to you to perform the decimation/filtering. (We have some example code for this on the ATSAMD21 chipset)
- Your chip does not come with a hardware peripheral but you're pretty clever and come up with a way to make it work (See this example for the ATtiny85)
- You generate the high speed clock, then add an analog filter on the data line, and read the analog value (A hack, but works!)

Either way you decide to go, make sure you have a handle on what support you get with your platform, as these chips are a little tricky!



Each order comes with one fully assembled and tested microphone, and a little header to solder on for breadboard-compatibility

## Pinouts



These mics are very simple!

- **3V -** This is the power input pin, this powers the chip directly. Use a quiet power supply pin if available. (The chip supports 1.8-3.3V but we have not tested it at 1.8V)
- **GND** - Power and data ground reference
- **SEL** - Left/Right select. If this pin is high, the output is on the falling edge of CLK considered the 'Right' channel. If this pin is low, the output is on the rising edge, a.k.a 'Left' channel.
- **CLK** - PDM clock in, 1 - 3 MHz square wave required
- **DAT** - PDM data out.

## Wiring & Test

At this time we only have example code for the SAMD21 chipset using the I2S peripheral, you'll be limited to what pins you can use and the digital filtering must be done in software but it *does* work! We don't necessarily recommend this mic for SAMD21 - an analog microphone will work quite well with less hassle!

> The SAMD21 can be used with PDM but it's not ideal unless you're willing to put in the work to filter data and manage the peripheral - consider this code for test & experimentation!

## Available I2S Pins

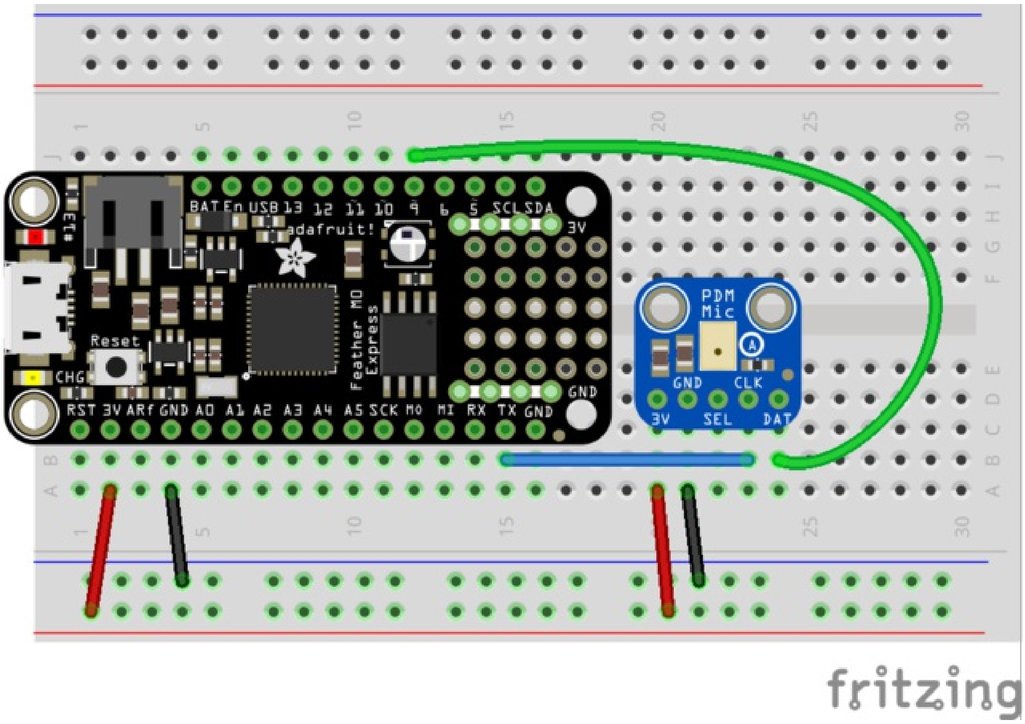
As we are using the I2S peripheral, not all pins can be used! For the Feather M0 / Metro M0 / Arduino Zero family, here's the available I2S pins:

Available Clock Pins:

- **PA10** a.k.a D1 or TX
- **PB11** a.k.a SCK
- **PA20** a.k.a. D6

Available Data Pins:

- **PA07** a.k.a D9
- **PA08** a.k.a D4
- **PA19** a.k.a. D12

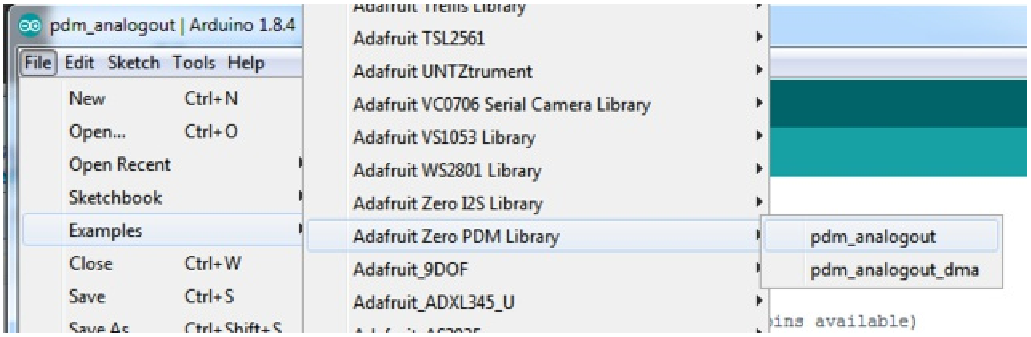

BAT En USB 13 12 11 10 9 6 5 SCL SDA
adafruit!
3V
Feather M0 Express
Reset
CHG
RST 3V ARf GND A0 A1 A2 A3 A4 A5 SCK MO MI RX TX GND
GND
PDM Mic
GND CLK
3V SEL DAT
fritzing

## Install Library

Download the latest version of the **ZeroPDM** library from github. Install as usual!

We have a two examples, one uses the DMA capability to grab data, which means we don't have to do as much work, but there's more setup involved and requires the ZeroDMA library.

We recommend starting with the basic demo, which will echo audio data to **A0** (the the analog output). Connect up headphones or an oscilloscope to A0 to hear/see the audio!



Before uploading, be sure to change the instantiator to match your pinouts:

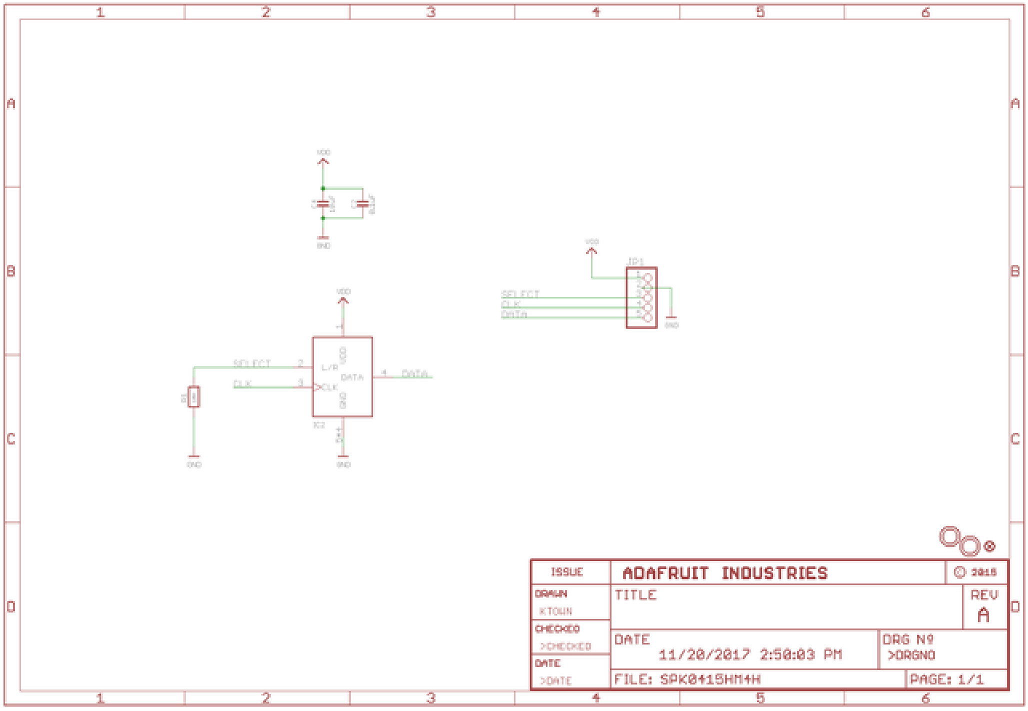
```
// Create PDM receiver object, with Clock and Data pins used (not all pins available)
Adafruit_ZeroPDM pdm = Adafruit_ZeroPDM(1, 4);  // Metro M0 or Arduino zero
//Adafruit_ZeroPDM pdm = Adafruit_ZeroPDM(34, 35);  // CPlay express
```

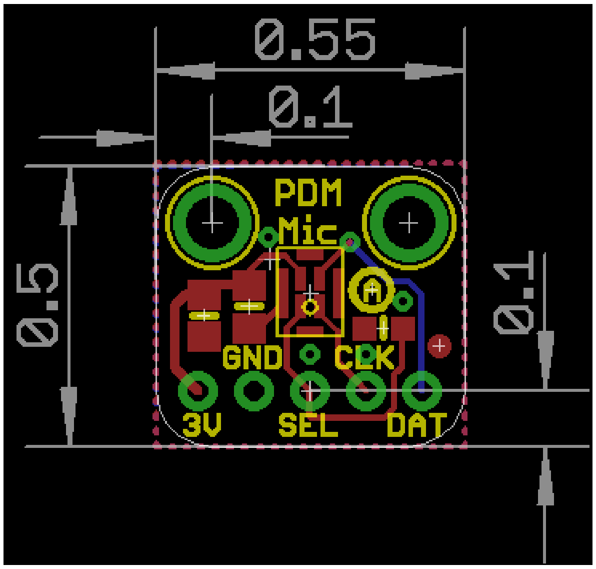
# Downloads

## Files

- MP34DT01-M datasheet

## Schematic & Fabrication Print


1
2
3
4
5
6
A
B
C
D
VDD
GND
VDD
JP1
1
2
3
4
5
SELECT
CLK
DATA
GND
VDD
1
SELECT
2
L/R
CLK
3
CLK
DATA
4
DATA
VDD
GND
R1
IC2
GND
GND
ISSUE
ADAFRUIT INDUSTRIES
© 2015
DRAWN
KTOWN
CHECKED
>CHECKED
DATE
>DATE
TITLE
REV
A
DATE
11/20/2017 2:50:03 PM
DRG Nº
>DRGNO
FILE: SPK0415HM4H
PAGE: 1/1


0.55
0.1
0.5
0.1
PDM
Mic
GND
CLK
3V
SEL
DAT