Bill Liu

Bill’s Log book

Table of Contents

[Mesh Network (23/3/2021) 2](#_Toc70633257)

[What is mesh network? 2](#_Toc70633258)

[What is the mesh network based on, Wi-Fi or Bluetooth? 2](#_Toc70633259)

[Which one is better? 2](#_Toc70633260)

[Wi-Fi: (not a mesh) 2](#_Toc70633261)

[Bluetooth: 2](#_Toc70633262)

[Zigbee: 2](#_Toc70633263)

[What is the available mesh network protocols? 2](#_Toc70633264)

[What mesh network best suits this project? 3](#_Toc70633265)

[MEMS Microphone research (12/04/2021) 3](#_Toc70633266)

[SPH0644LM4H-1 3](#_Toc70633267)

[What is MEMS microphone 3](#_Toc70633268)

[MEMS Microphone acoustic sensor 3](#_Toc70633269)

[Output Signal 4](#_Toc70633270)

[Programming with Arduino 4](#_Toc70633271)

[How to read Mic input with teensy (18/4/2021) 5](#_Toc70633272)

[Mic output 5](#_Toc70633273)

[Input converting 5](#_Toc70633274)

[How does the data acquisition works? 5](#_Toc70633275)

[Converting PDM to PCM 6](#_Toc70633276)

[Existing examples 7](#_Toc70633277)

[First example (code for PDM to PCM) 7](#_Toc70633278)

[Second example (acquires the PDM using DFSDM output as fft) 7](#_Toc70633279)

[Third example (Ultrasonic communication by STM32L4 and MEMS microphone) 8](#_Toc70633280)

# Mesh Network (23/3/2021)

What is mesh network? A mesh network is a local network topology in which the infrastructure nodes. Mesh networks can relay messages using either a flooding technique or a routing technique. With routing, the message is propagated along a path by hopping from node to node until it reaches its destination. To ensure that all its paths are available, the network must allow for continuous connections and must reconfigure itself around broken paths, using self-healing algorithms such as Shortest Path Bridging. A mesh network whose nodes are all connected to each other is a fully connected network. Fully connected wired networks have the advantages of security and reliability.

## What is the mesh network based on, Wi-Fi or Bluetooth?

## Which one is better?

## Wi-Fi: (not a mesh)

High bandwidth.

It only works well for short ranges. If your devices are too far apart you will have to use an extender, which will likely affect latency and connection speed.

It is not very power efficient. Wi-Fi-based devices usually last about 10 hours.

## Bluetooth:

More robust.

Low energy consumption.

Secure: state-of- the-art security.

Range problem: ***what is the antenna size and range?***

They are more expensive to implement.

Their topologies are more complicated and difficult to build and maintain.

They have a higher chance of redundant connections, which adds to the costs and potential for reduced efficiency.

## Zigbee:

Low energy consumption.

Short range.

Low transmission rate.

Not very secure.

## What is the available mesh network protocols?

Bluetooth:-95dB 1Mbps or -103dB 125kbps

IEEE 802.15.4-2006: 250kbps

2.4GHz transceiver: 2 Mbps or 1 Mbps

## What mesh network best suits this project?

Still to decide.

# MEMS Microphone research (12/04/2021)

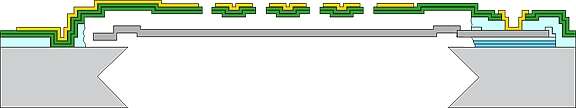
## SPH0644LM4H-1

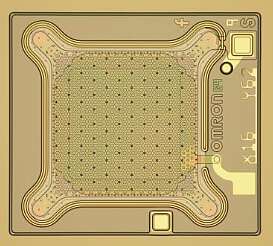
## What is MEMS microphone

The application of MEMS (microelectro-mechanical systems) technology to microphones has led to the development of small microphones with very high performance.  MEMS microphones offer high SNR, low power consumption, good sensitivity, and are available in very small packages that are fully compatible with surface mount assembly processes.  MEMS microphones exhibit almost no change in performance after reflow soldering and have excellent temperature characteristics.

## MEMS Microphone acoustic sensor

MEMS microphones use acoustic sensors that are fabricated on semiconductor production lines using silicon wafers and highly automated processes. Layers of different materials are deposited on top of a silicon wafer and then the unwanted material is then etched away, creating a moveable membrane and a fixed backplate over a cavity in the base wafer. The sensor backplate is a stiff perforated structure that allows air to move easily through it, while the membrane is a thin solid structure that flexes in response to the change in air pressure caused by sound waves.





Changes in air pressure created by sound waves cause the thin membrane to flex while the thicker backplate remains stationary as the air moves through its perforations.  The movement of the membrane creates a change in the amount of capacitance between the membrane and the backplate, which is translated into an electrical signal by the ASIC.

## Output Signal

Most digital microphones use pulse density modulation (PDM), which produces a highly oversampled single-bit data stream. The density of the pulses on the output of a microphone using pulse density modulation is proportional to the instantaneous air pressure level. Pulse density modulation is similar to the pulse width modulation (PWM) used in class D amplifiers. The difference is that pulse width modulation uses a constant time between pulses and encodes the signal in the pulse width, while pulse density modulation uses a constant pulse width and encodes the signal in the time between pulses.

In addition to the output, ground, and VDD pins found on analog mics, most digital mics also have inputs for a clock and a L/R control.  The clock input is used to control the delta-sigma modulator that converts the analog signal from the sensor into a digital PDM signal.  Typical clock frequencies for digital microphones range from about 1 MHz to 3.5 MHz. The microphone’s output is driven to the proper level on the selected clock edge and then goes into a high impedance state for the other half of the clock cycle.  This allows two digital mic outputs to share a single data line.  The L/R input determines which clock edge the data is valid on.

**Although codecs are not required for digital MEMS microphones, in most cases the pulse density modulated output must be converted from single-bit PDM format into multibit pulse code modulation (PCM) format.  Many codecs and SoCs have PDM inputs with filters that convert the PDM data into PCM format.  Microcontrollers can also use a synchronous serial interface to capture the PDM data stream from a digital mic and convert it into PCM format using filters implemented in software.**

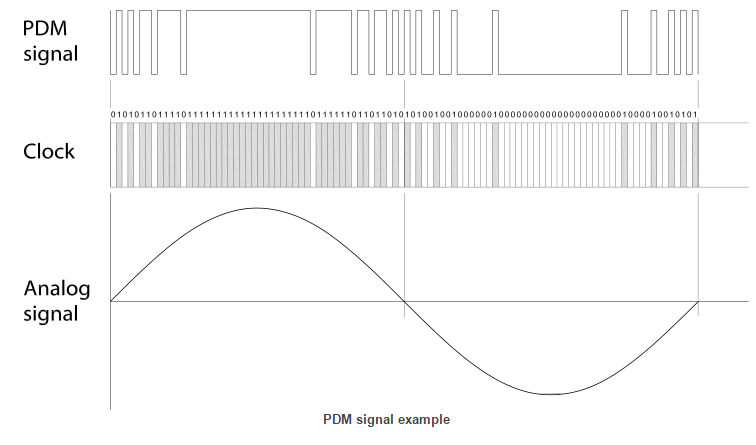
## Programming with Arduino

PDM library link: <https://www.arduino.cc/en/Reference/PDM>

# How to read Mic input with teensy (18/4/2021)

## Mic output

Digital PDM output from SPH0644LM4H-1.



The digital PDM would looks like the 010101101111011111111111111111111111011111011011010.

## Input converting

### How does the data acquisition works?

The digital MEMS microphone outputs a PDM signal, which is a high frequency (1 to 3.25

MHz) stream of 1-bit digital samples. The data is acquired by the DFSDM serial transceiver

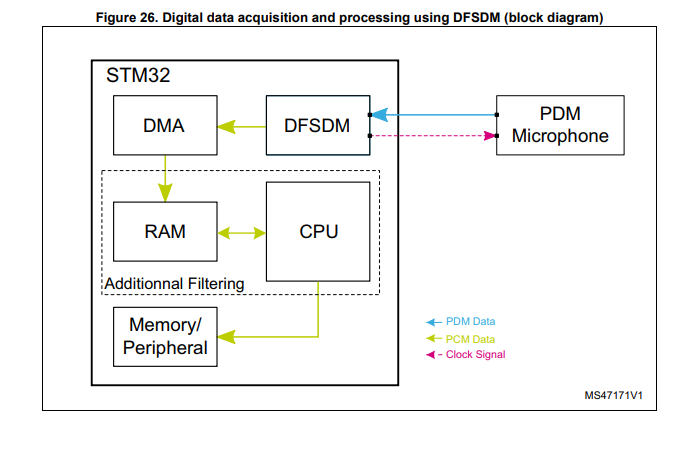
that provides connection to the external Sigma-Delta modulator of the digital microphone.

The digital filters perform CPU-free filtering that averages the 1-bit input data stream from

the SD modulator into a higher resolution and a lower sample rate. This data is transferred

through DMA (thus reducing the software overhead) to a system RAM buffer to be further filtered. After that, the PCM raw data can be handled depending on the application implementation (stored as wave/compressed data in a mass storage media, transferred to an external

audio codec DAC).

 **[P34 Interfacing PDM digital microphones using STM32 MCUs and MPUs]**

The DFSDM peripheral is dedicated to interface the external Σ∆ modulators to

microcontroller and then to perform digital filtering of the received data streams (which

represent analog value on Σ∆ modulators inputs) **[ARM Cortex-M7 datasheet (STM32H753xI)]**

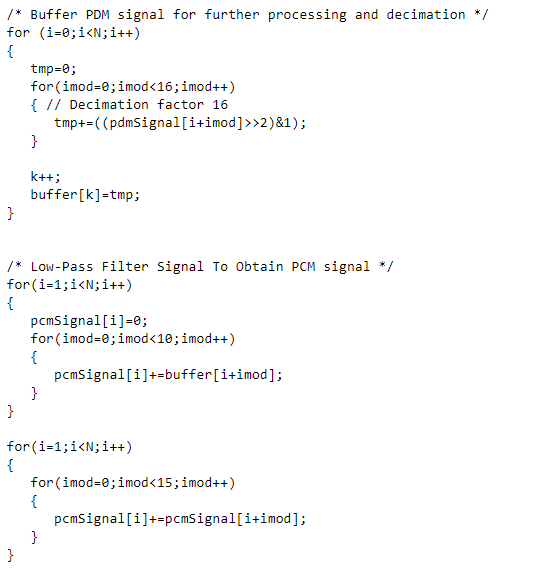
### Converting PDM to PCM

Converting PDM to analog is in principle very simple. The one-bit signal already contains the audio in the low part of the spectrum. All that is required to recover it is a low-pass filter. In practice, the fast switching edges in the signal require careful design of the analog filtering stages, but it is certainly possible to recover a very high-quality analog signal this way.

Converting PDM to PCM is more involved. The sample rate needs to be reduced by the oversampling factor. This is accomplished in a digital filtering operation called **decimation**. Decimation is the counterpart to interpolation: samples are removed from the signal to reduce the sampling rate. It is important that the noise above the audio band in the 1-bit representation not be allowed to alias into the audio band. The decimation filters are designed to filter out this noise, leaving the baseband audio signal intact. The output of the decimator is a PCM audio stream at the baseband (non-oversampled) rate. Typically, the wordlength increases from 1 bit to around 20 effective bits during the filtering. **[P8 Understanding PDM Digital Audio]**

## Existing examples

### First example (code for PDM to PCM)



[https://www.dsprelated.com/showthread/comp.dsp/288391-1.php]-**more information on website same Mic (not the best option).**

### Second example (acquires the PDM using DFSDM output as fft)

This project acquires the PDM (Pulse Density Modulation) microphone signal using DFSDM (Digital filter for Sigma-Delta modulators interface) function of STM32 MCU and outputs its frequency characteristics by using FFT.

Link: <https://github.com/y2kblog/NUCLEO-L476RG_DFSDM_PDM-Mic>

### Third example (Ultrasonic communication by STM32L4 and MEMS microphone)

**Link:** [**https://github.com/araobp/ultrasonic-communication**](https://github.com/araobp/ultrasonic-communication)

# Answering some problems of for the PDM to PCM conversion (29/4/2021 time:2h)

## What sampling rate best use for the MEMs microphone?

The device to which the microphone connects provides the master clock to the PDM microphone. The clock rate defines the sampling rate of the system, as well as the rate at which bits are transmitted on the data line. Although there is no defined standard, typically the oversampling ratio is 64. So to achieve a bandwidth of 24 kHz (comparable to a PCM system sampled at 48 kHz), a master clock frequency of 3.072 MHz is needed.

Manufacturers of such devices typically use 4th-order sigma delta modulators at a clock frequency of 3.072 MHz, which in the receiver is typically decimated by a factor of 64 to a baseband sampling rate of 48 kHz.

## Wordlength (Buffer) size of the PCM signal

PCM

Before we tackle PDM, let’s first review PCM, that is, conventional multi-bit digital audio. In PCM, the

audio signal is represented as a series of samples, each a fixed number of bits long. Two factors

determine the performance of the system:

• Sampling rate. This determines the bandwidth of the system.

Understanding PDM Digital Audio 4

• Wordlength. This determines the signal-to-noise ratio (SNR) of the system.

In particular, the bandwidth is fs/2, where fs is the sampling rate, and the SNR is given by

(6.02N + 1.76) dB, where N is the wordlength in bits.

**Ideally we want no noise in the mic signal and, therefore, a signal-to-noise ratio (SNR) or infinity. A low self-noise rating is essential when capturing quiet sound sources (15 dBA or less) and so a “good” SNR would be 79 dB or more.**

A raw 16-bit system has a theoretical SNR of around 98 dB. In practice, dither is used to linearize the

system and eliminate noise modulation; this reduces the SNR by about 4 dB. Using the above formula,

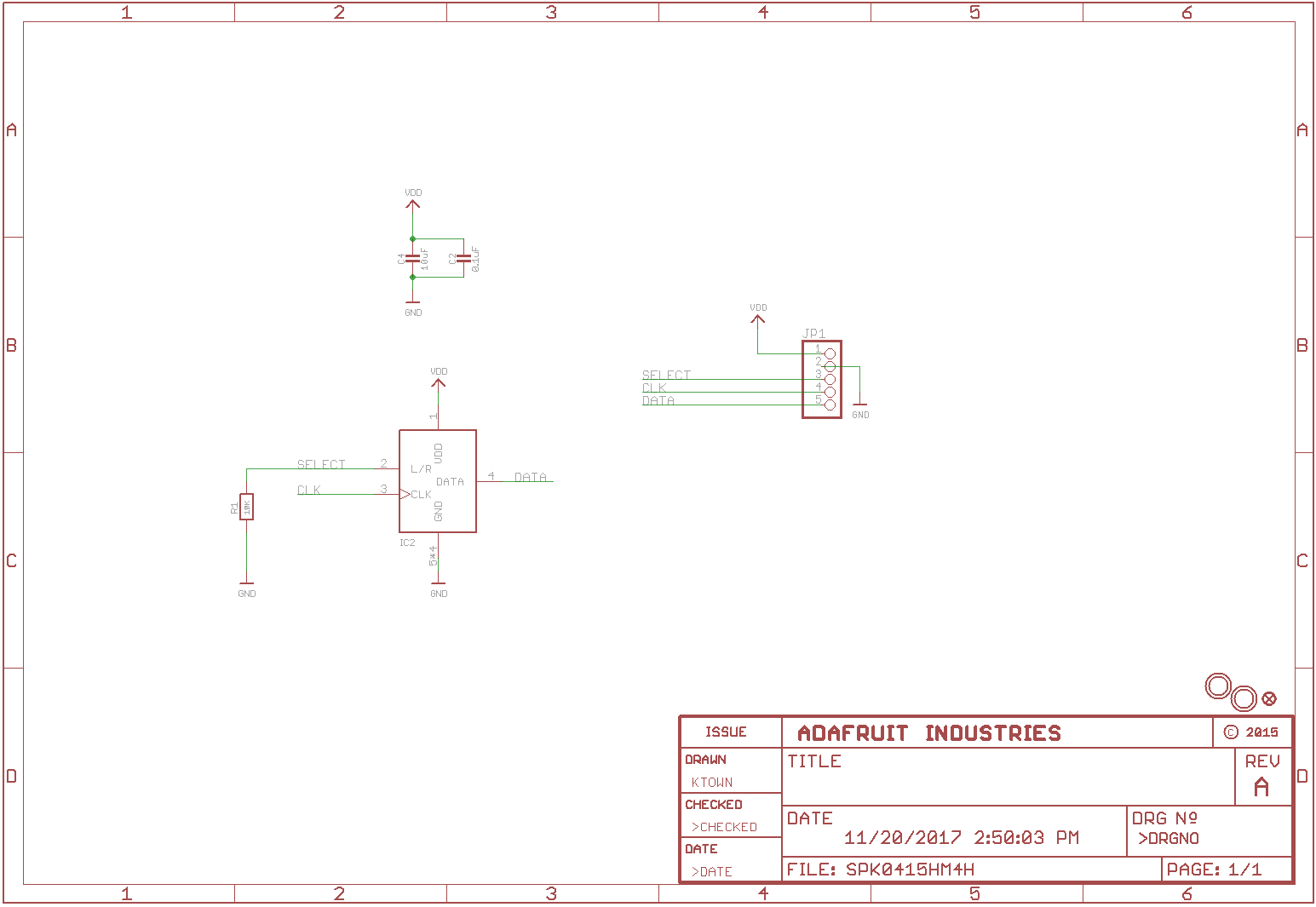
an undithered 1-bit system has an SNR of about 8 dB, which is of course unacceptable for any real audio work. Furthermore, optimal dither needs 2 LSBs to work; since a 1-bit system only has 1 LSB total, and that is used for the audio, hence there is no room for dither.

Since the system cannot be properly dithered, a 1-bit representation would at first blush appear to be a non-starter. The solution lies in an understanding of noise shaping and oversampling.

# MEMs microphone breakout board design and test codes(4/5/2021 time:4h)

## Examples of existing PDM MEMs

<https://learn.adafruit.com/adafruit-pdm-microphone-breakout/circuitpython>



# Some unsolved problems:

## What microcontroller choices do we have?

# Testing

With measurements of the oscilloscope, the mic’s sampling frequency is Table

Description automatically generated

The 110101010110.

Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidence