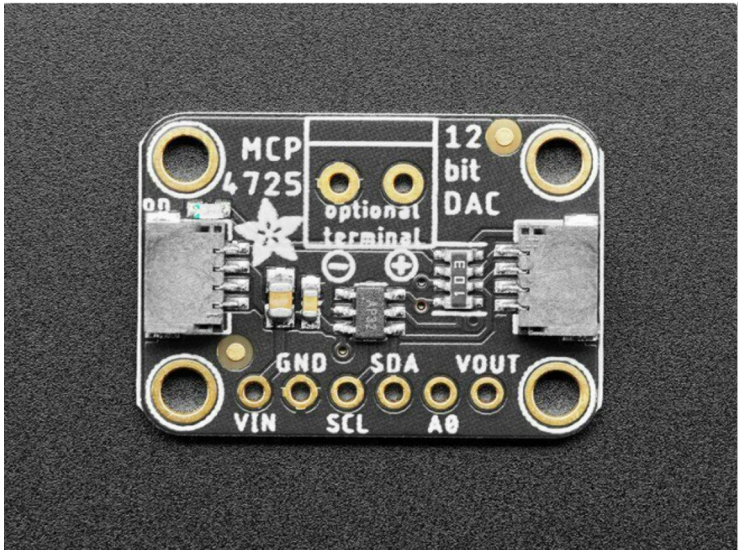
# MCP4725 DAC

# What is it?



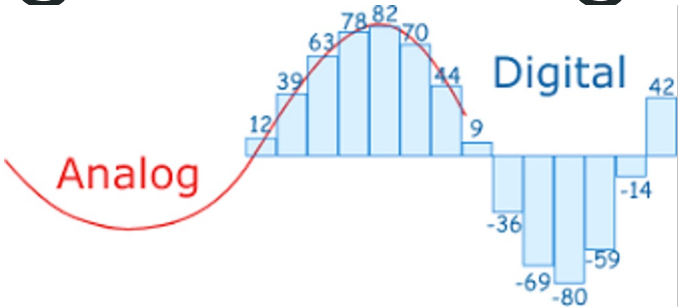

It is a Digital to Analog Converter

# Why would you need that?

Analog and Digital signals both carry information...

but they use different methods

# Analog Signals vs. Digital Signals




carries information through sin waves

- continuous electrical signal

carries information through square waves

- non-continuous electrical signal

# Why would you need that?

To convert between signals in your breadboard

# How does it work?

The MCP4725 has a 12 bit resolution, meaning it can specify 2^12 or 4096 different values.

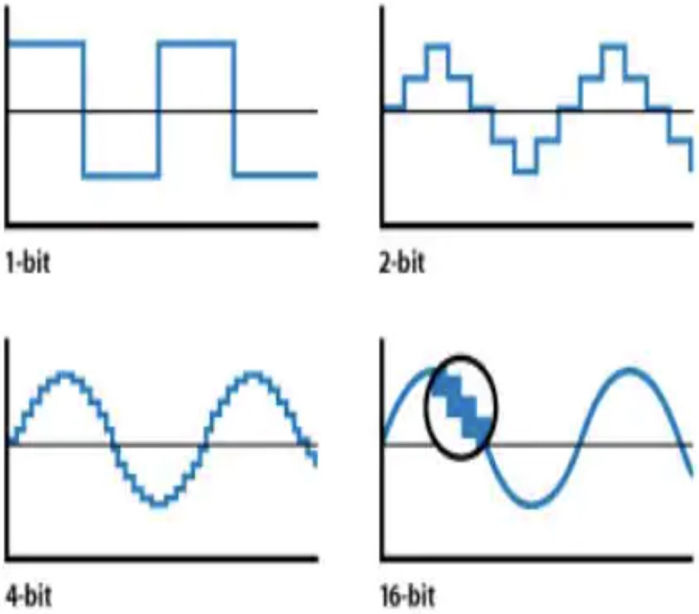
- The reference voltage (5V), is divided into equally spaced parts
- Based on the digital value received, we can create the analog voltage we want
- For example, if we wanted an analog output of 2.5V or half of the reference voltage, we would send the digital signal 2048, which is half of 4096
- This can be done in increments as small as 5V/4096 = 1.22mV
- If the reference voltage is 3V, we have even finer voltage output steps (.73mV)

# Analog and Digital Signals

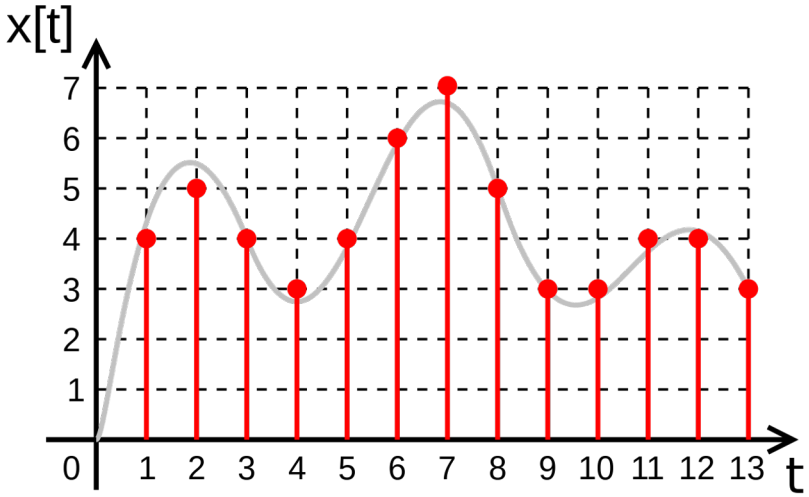
- Microcontrollers detect binary signals (digital signals)
  - Press of a button
  - 0V or 5V
- What if we want any other value that is not 0V or 5V?
  - We need analog signals!

# Digital-to-Analog Converter (DAC)

- DACs is a data converter which generates an analog output from a digital input
- There is a source of error in the analog output: digital values are only approximations of real-world analog signals
- The more digital bits represented by the DAC, the more accurate the analog output signal



# Digital-to-Analog Converter (DAC)



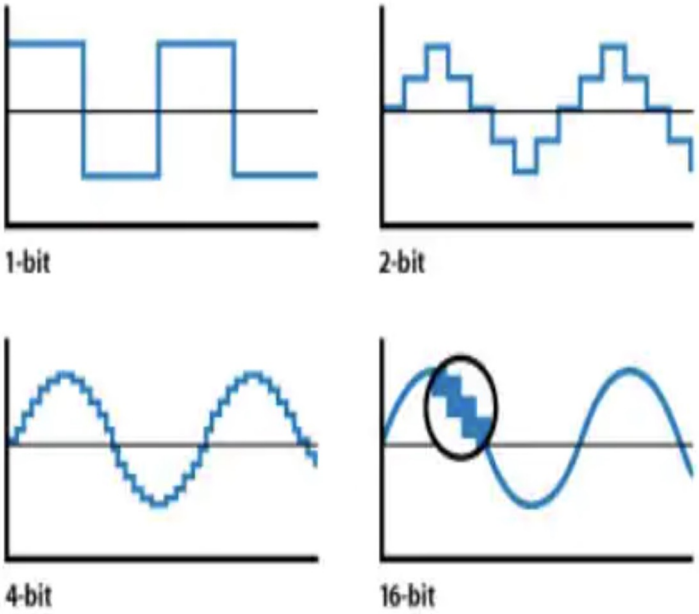

- The conversion produces an analog waveform that is discrete, not continuous
- Discrete output is then integrated in the DAC
- Resultant analog waveform is filtered on a low-pass filter to be smoothed out

# Analog-to-Digital Converter (ADC)

- ADCs convert analog voltage on a pin to a digital number
- Not every pin on a microcontroller has the ability to do analog conversions

# Analog-to-Digital Converter (ADC)

- The world contains information in continuous values, like temperature and motion
- ADCs convert continuous values to a sequence of discrete values
- The more bits in digital values, the closer it resembles an analog signal!

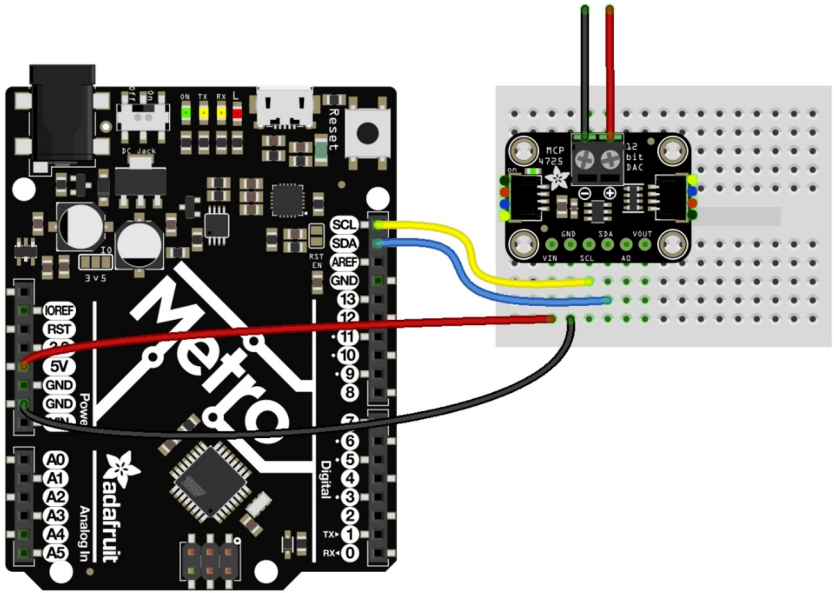


# Connections

The MPC4725 12-bit Digital to Analog Converter is connected via an I2C connection

- An I2C connection is an inter-integrated circuit. This means that it uses two bi-directional lines for data communication - the SDA and SCL.
- The SCL stands for serial clock pin and SDA for Serial Data Pin
- The lines are both pulled high, which means they are both at the operating voltage (as opposed to ground) of either 3.3 or 5V.

# Schematics




- The I2C connection means the wiring involved in setting up the sensor is incredibly simple
- One pin goes from the VIN to the 5V or 3V slot, depending on the system.
- Another runs from GND to GND
- Finally, the last two are connected via the SCL and SDA channels
- One can also use the A0 pin to connect a second MCP4725 on the same bus
- Additionally there is a VOUT which is the output for the I2C signal

# Setup for the MCP4725

The Adafruit_MCP4725 library must be downloaded first and is available through the Arduino Library manager

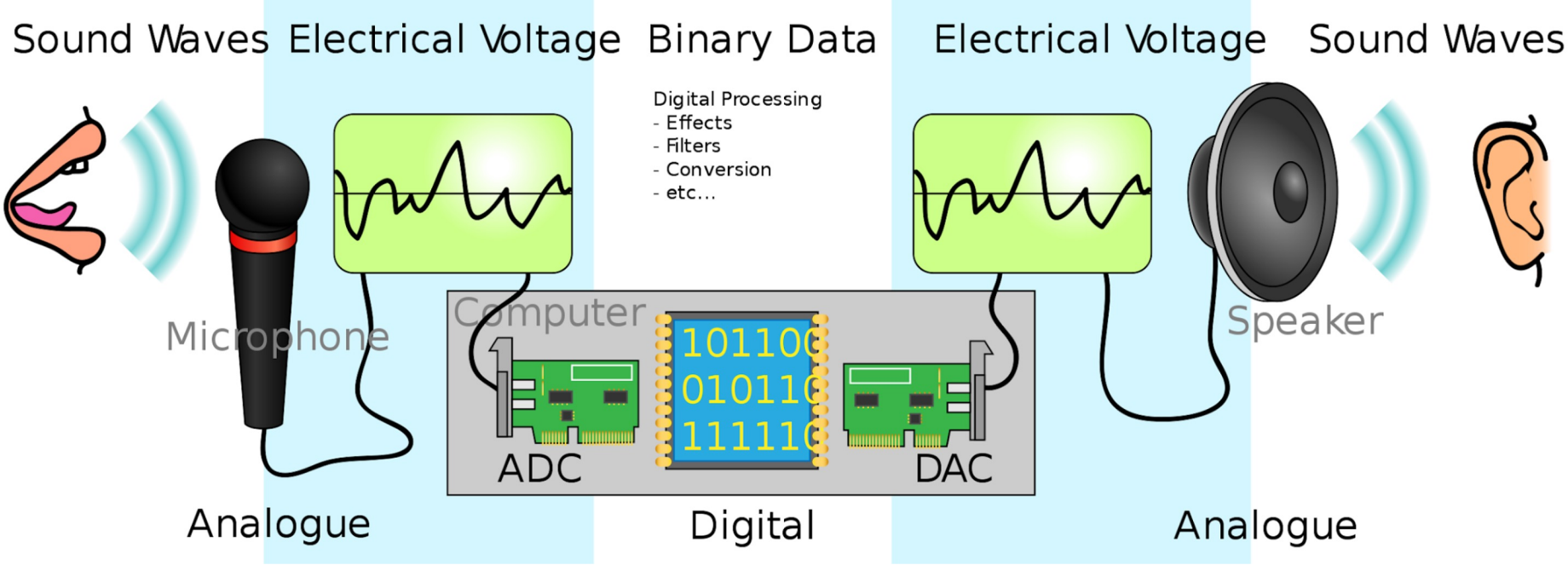
Considerations:

- The transfer speed is increased from 100 MHz to 400 MHz in the MCP4725 library versus the Arduino Wire Library
- The resolution is 12 bit, meaning you can specify values ranging from 1-4095 to control the output of the DAC in high resolution

# Applications

The Adafruit_MCP4725 can be used whenever a digital signal needs to be output as an analog signal

- Digital to analog converters are commonly used in music players to convert the digital music signal an analog signal that can be played by a speaker
- They are used on the receiving end of telephone calls to convert the digital signal to something we can hear
- Finally, they are sometimes used to create analog video from digital signals




- Data acquired by sensors generate an analog signal
- This signal is converted to a digital one by the ADC
- Computers/Arduino store and manipulate data digitally
- Signal must be reconverted to analog for the output to be continuous



## Why do we need an analog output?

- Physical quantities are continuous (at least on a macroscopic scale)
- Sound waves produced by speakers are analog - so its input must also be
- Controlling Servo Motors: position, speed, and acceleration are continuous quantities, so an analog, smooth signal will offer a higher precision in its control

# Sources


https://www.adafruit.com/product/935

https://byjus.com/physics/difference-between-analog-and-digital/#:~:text=Analog%20and%20digital%20signals%20are,have%20non%2Dcontinuous%20electrical%20signals.

https://www.google.com/imgres?imgurl=https%3A%2F%2Fwww.mathsisfun.com%2Fdata%2Fimages%2Fanalog-digital.svg&imgrefurl=https%3A%2F%2Fwww.mathsisfun.com%2Fdata%2Fanalog-digital.html&tbnid=KNvL5TIXZZ21tM&vet=12ahUKEwjcupysyfj8AhWZzskDHdPOD6wQMygGegUIARDQAQ..i&docid=vwi6hyqGhGGqaM&w=360&h=161&q=digital%20vs%20analog&ved=2ahUKEwjcupysyfj8AhWZzskDHdPOD6wQMygGegUIARDQAQ

https://cdn-learn.adafruit.com/downloads/pdf/mcp4725-12-bit-dac-tutorial.pdf

https://learn.adafruit.com//assets/108834

https://components101.com/ics/mcp4725-pinout-datasheet-equivalent-circuit-specs