

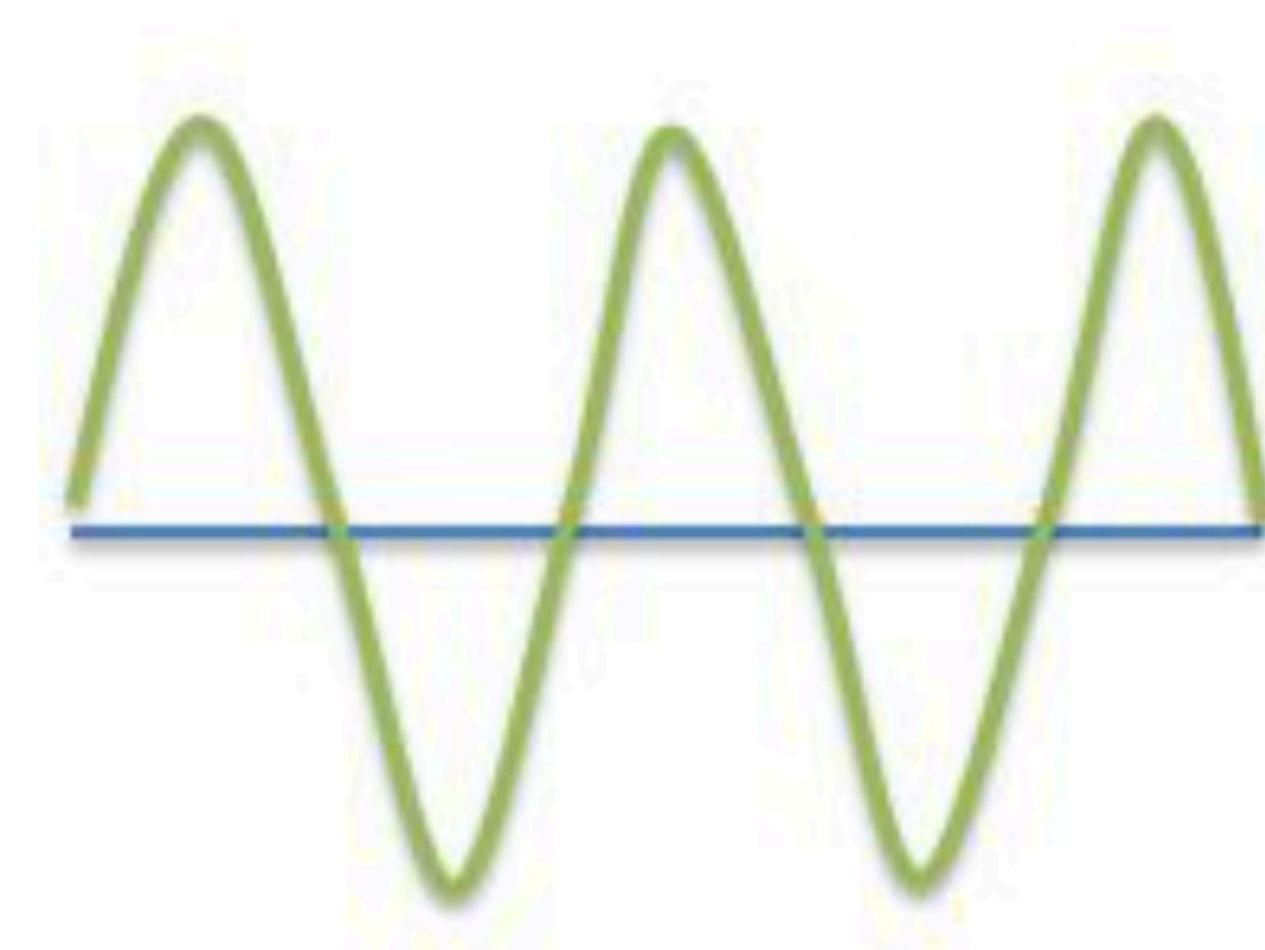
Lecture #2

Basic Electronics

Developer Platform

Android Things 2023

Signals



**Analog
Signal**

Vs

0100111101

**Digital
Signal**

Analog and Digital

- We can store only digital signal.
- We will convert everything that is analog to digital.
- When reading or writing digital signals we need to establish values for:
 - Bits per sample.
 - Sampling rate.

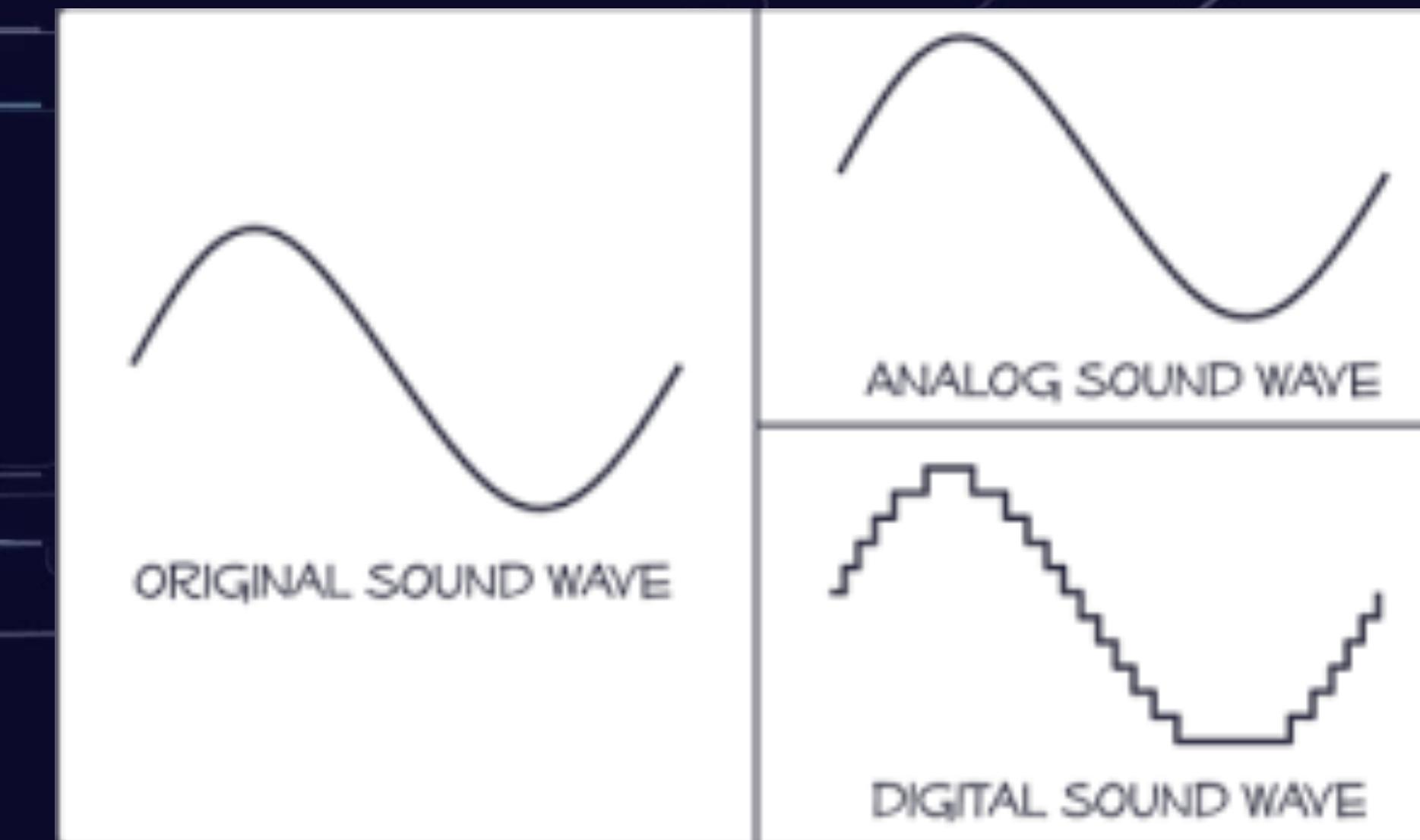
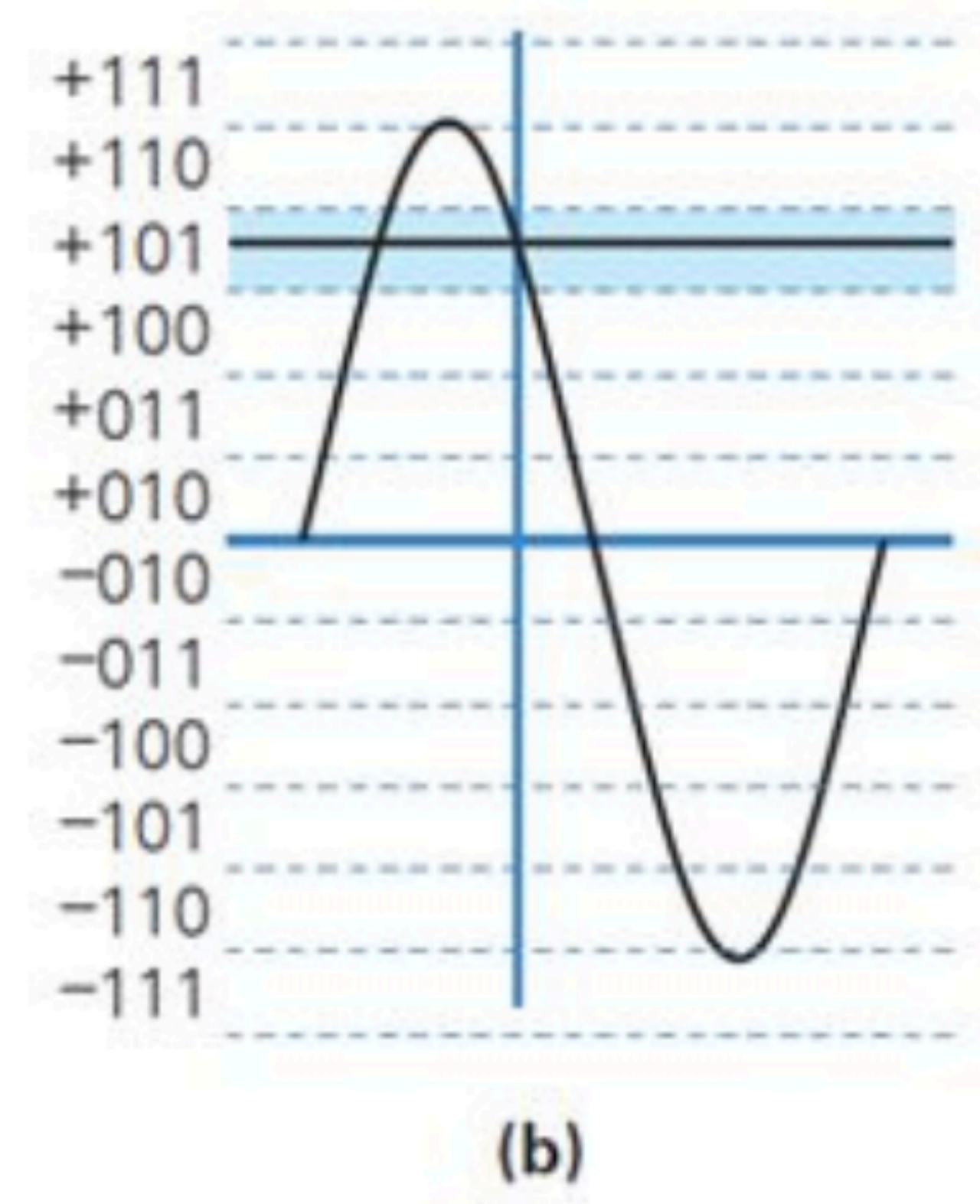
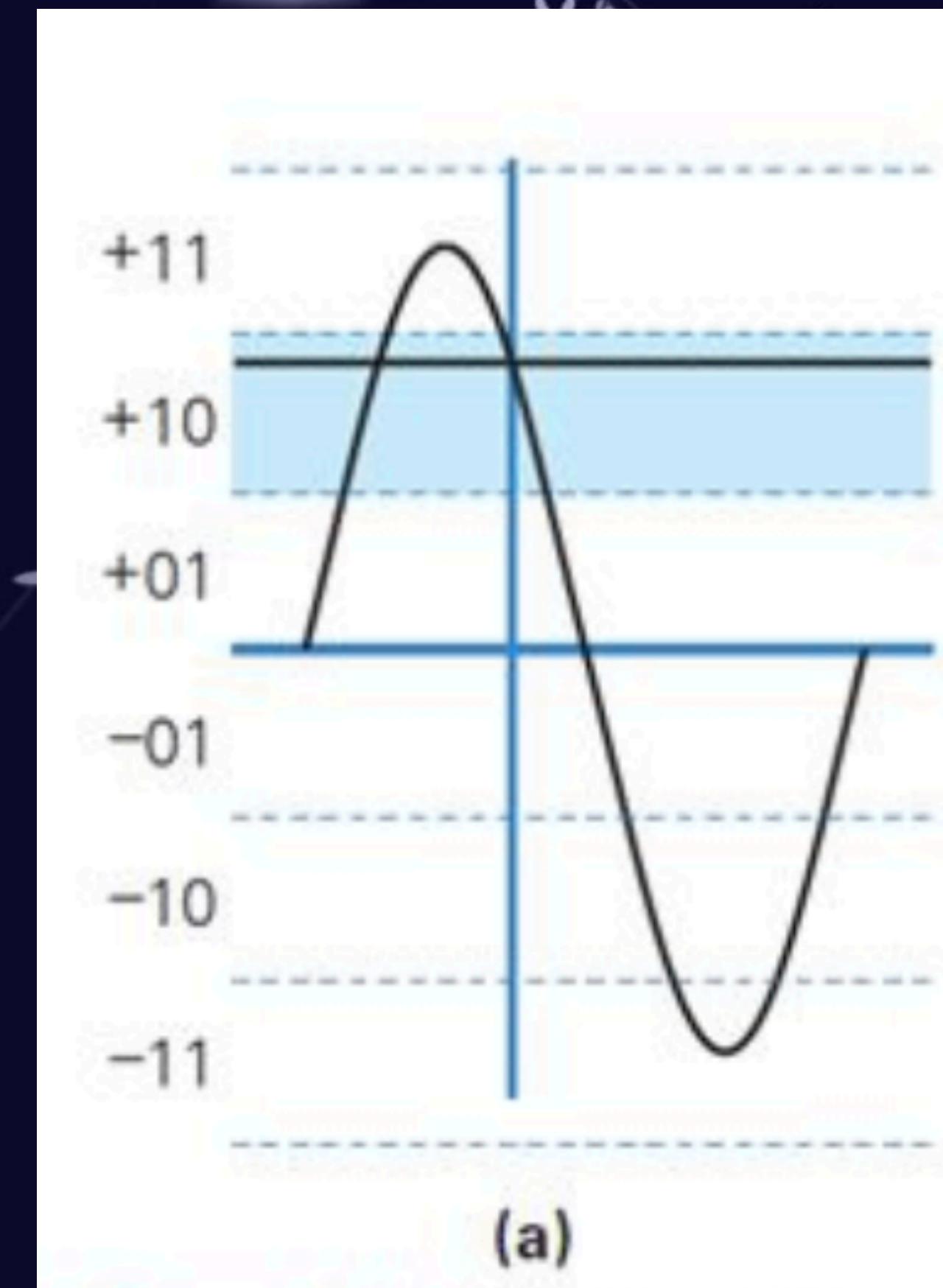


Image source: www.electronicproducts.com

Bits per sample

- How many bits are used to represent a value.



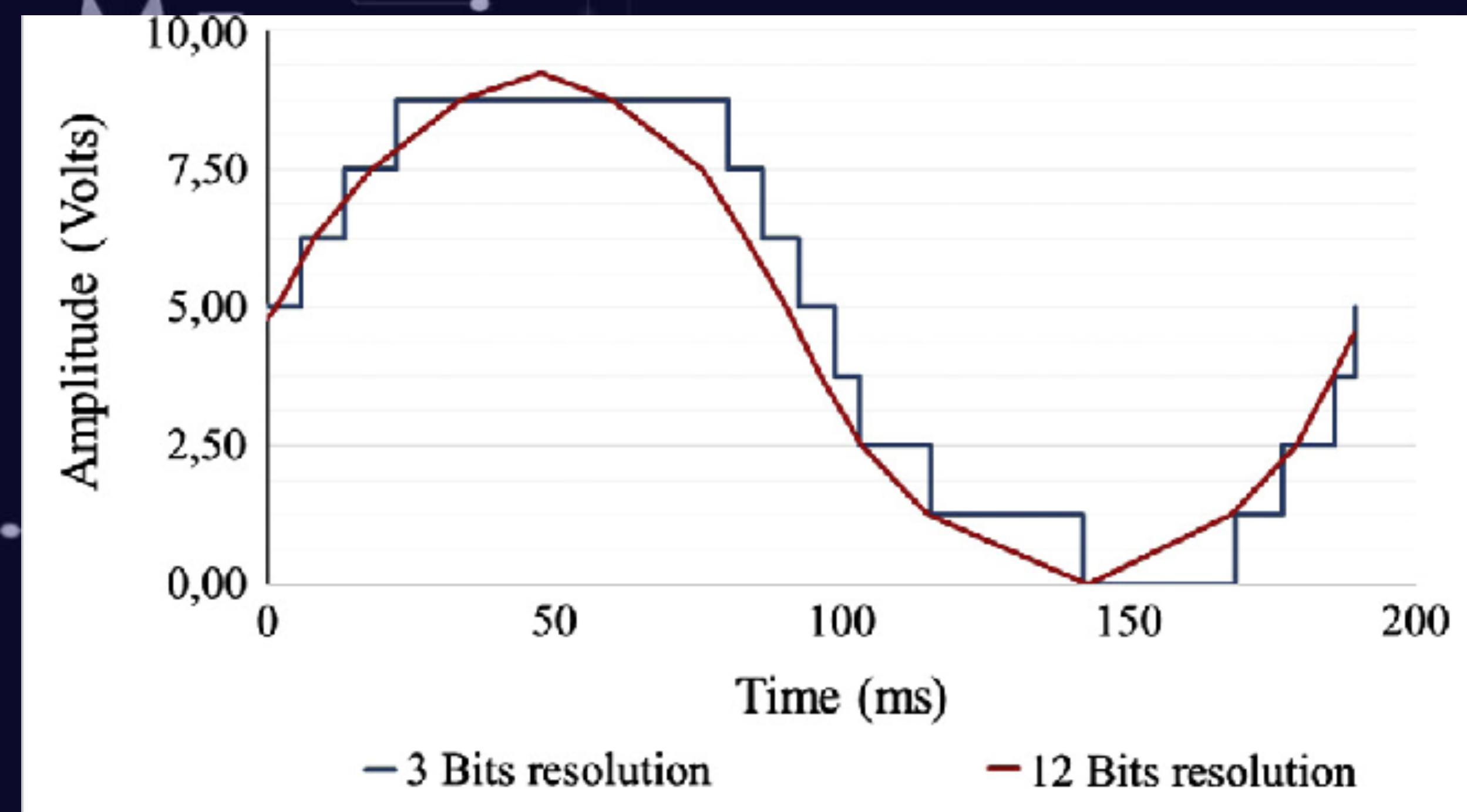
Bits per sample

- For n bits:
 - Values will range between 0 and $2^n - 1$.
 - Eg. for 3 bits we will have:
 - 000 - 0
 - 001 - 1
 - 010 - 2
 - 011 - 3
 - 100 - 4
 - 101 - 5
 - 110 - 6
 - 111 - 7



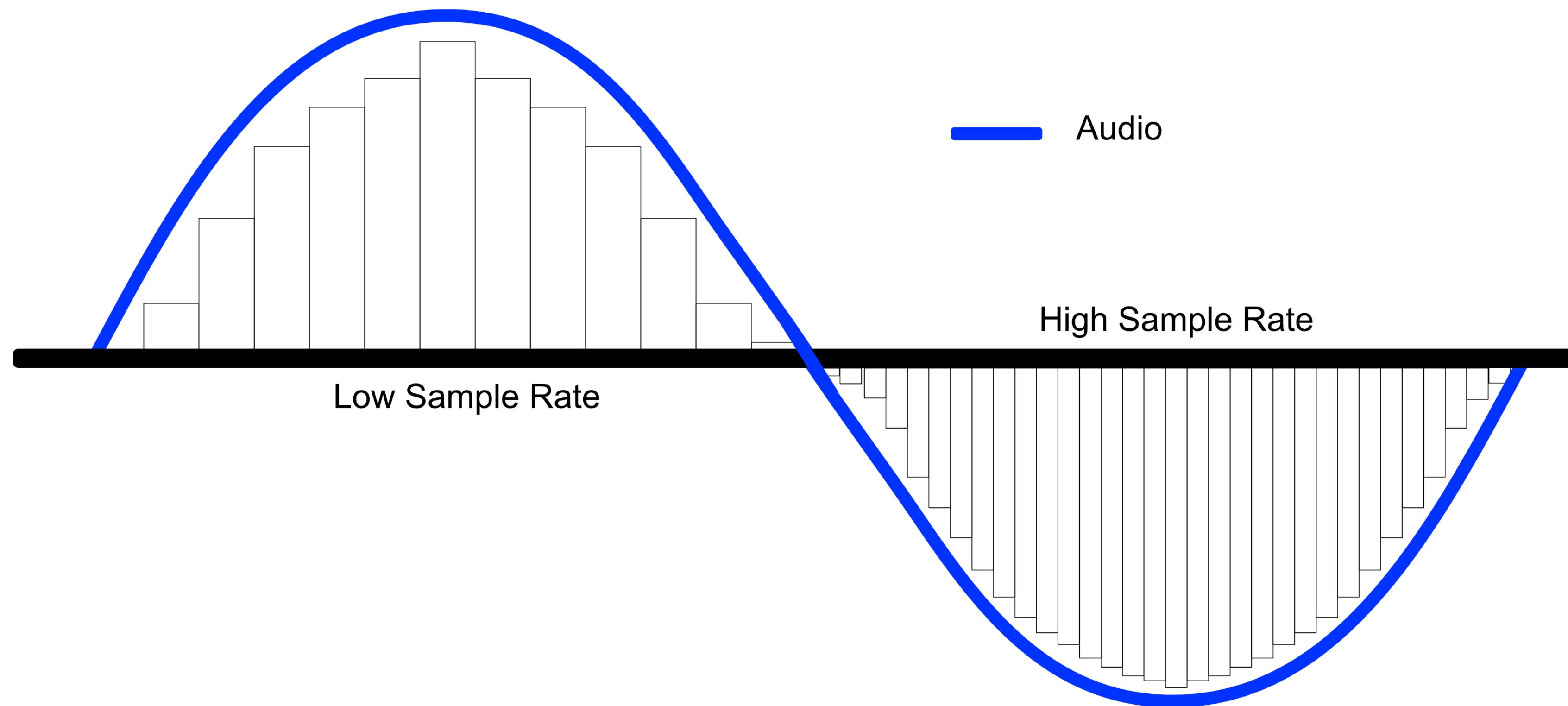
Source: etowns

Bits per sample

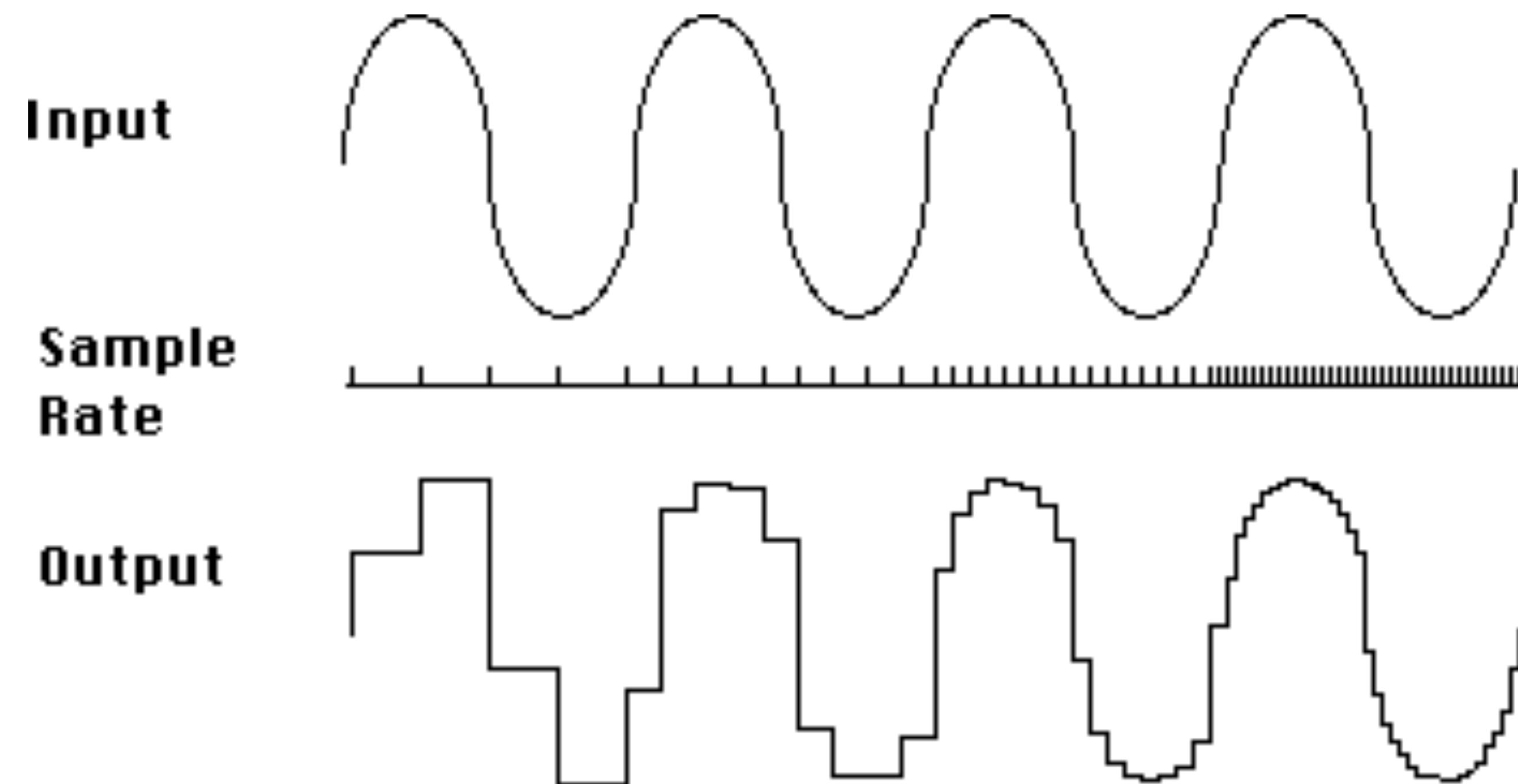


Sampling

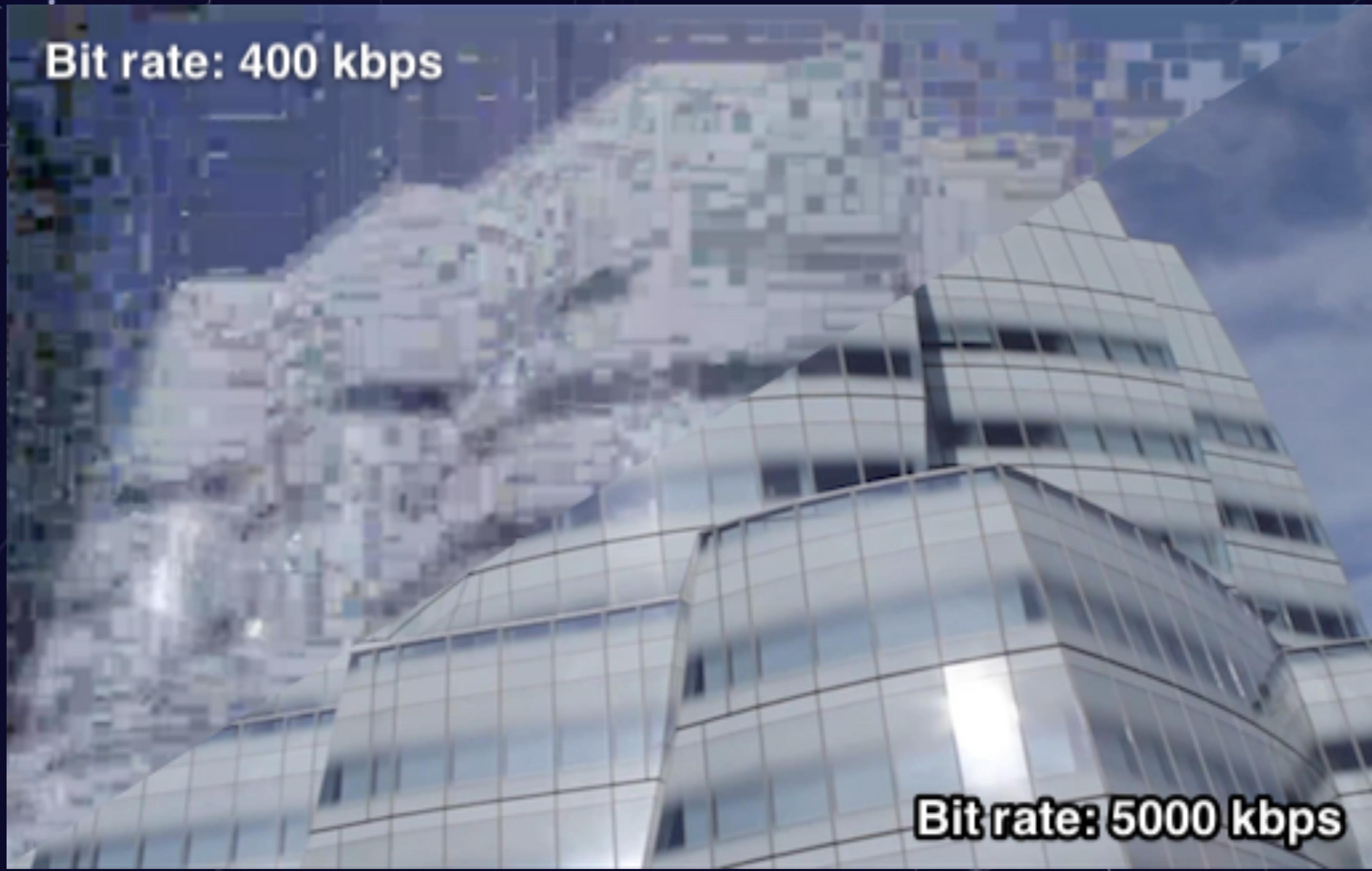
- Make measurements



Sampling



Sampling



Frequency

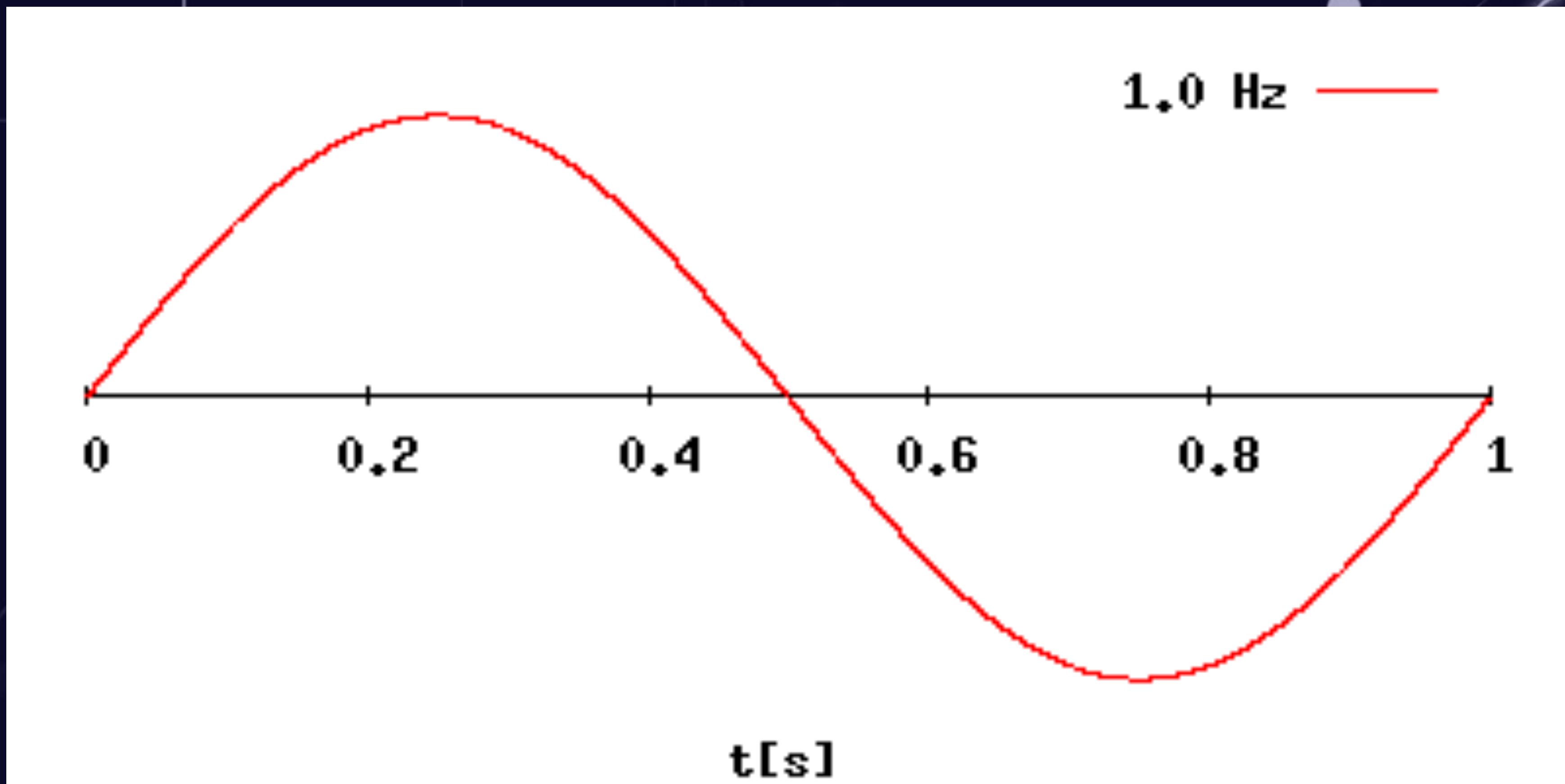
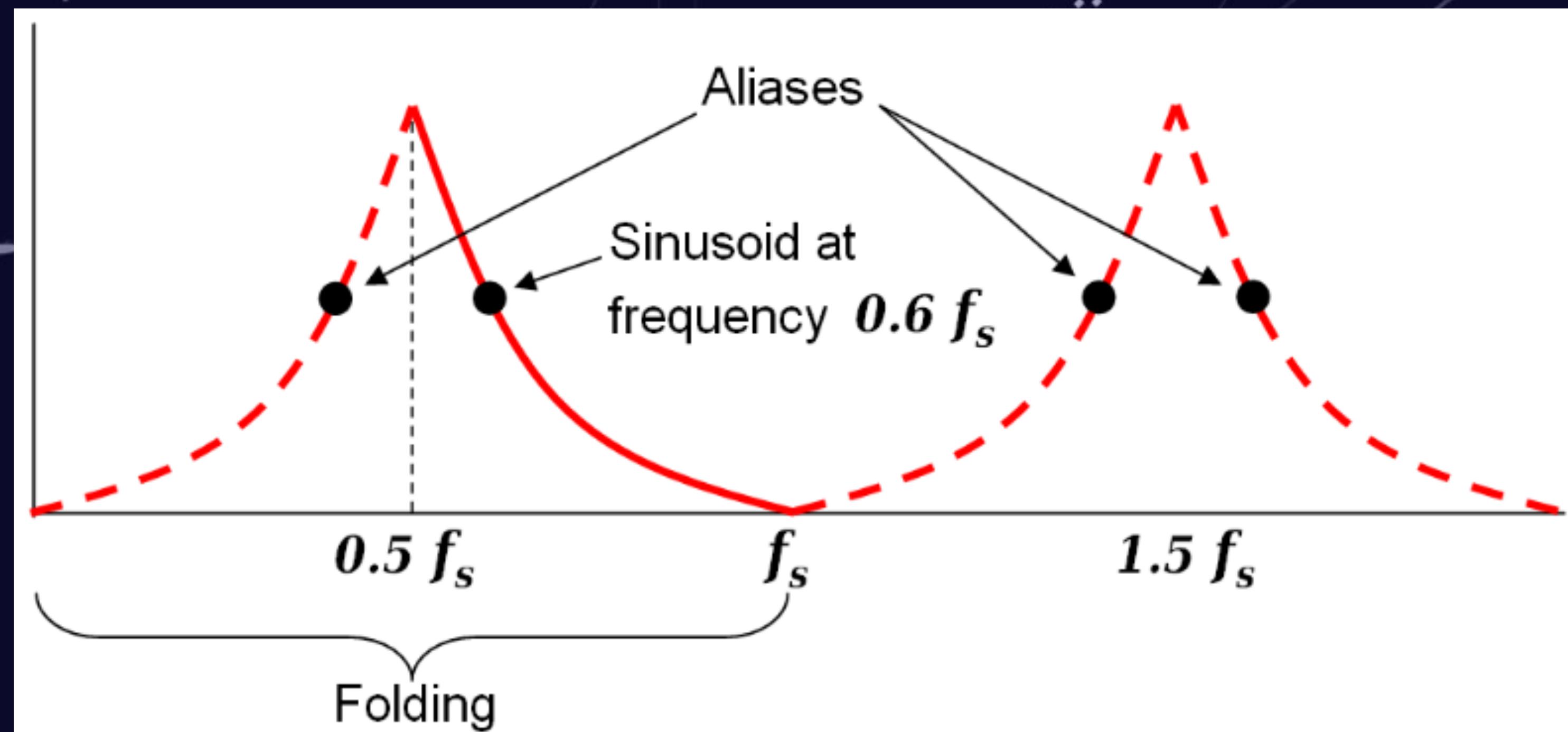


Image source: https://commons.wikimedia.org/wiki/File:Wave_frequency.gif

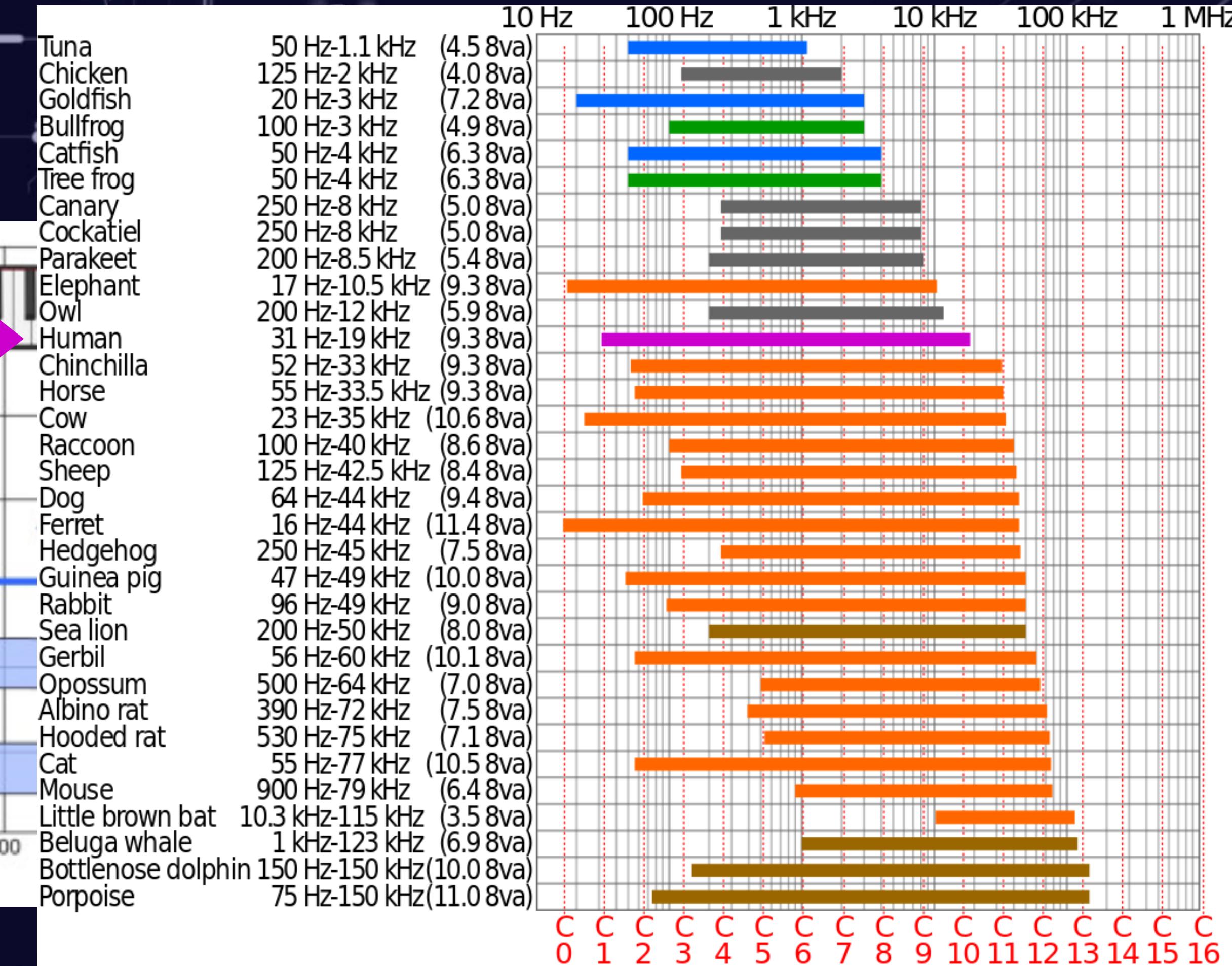
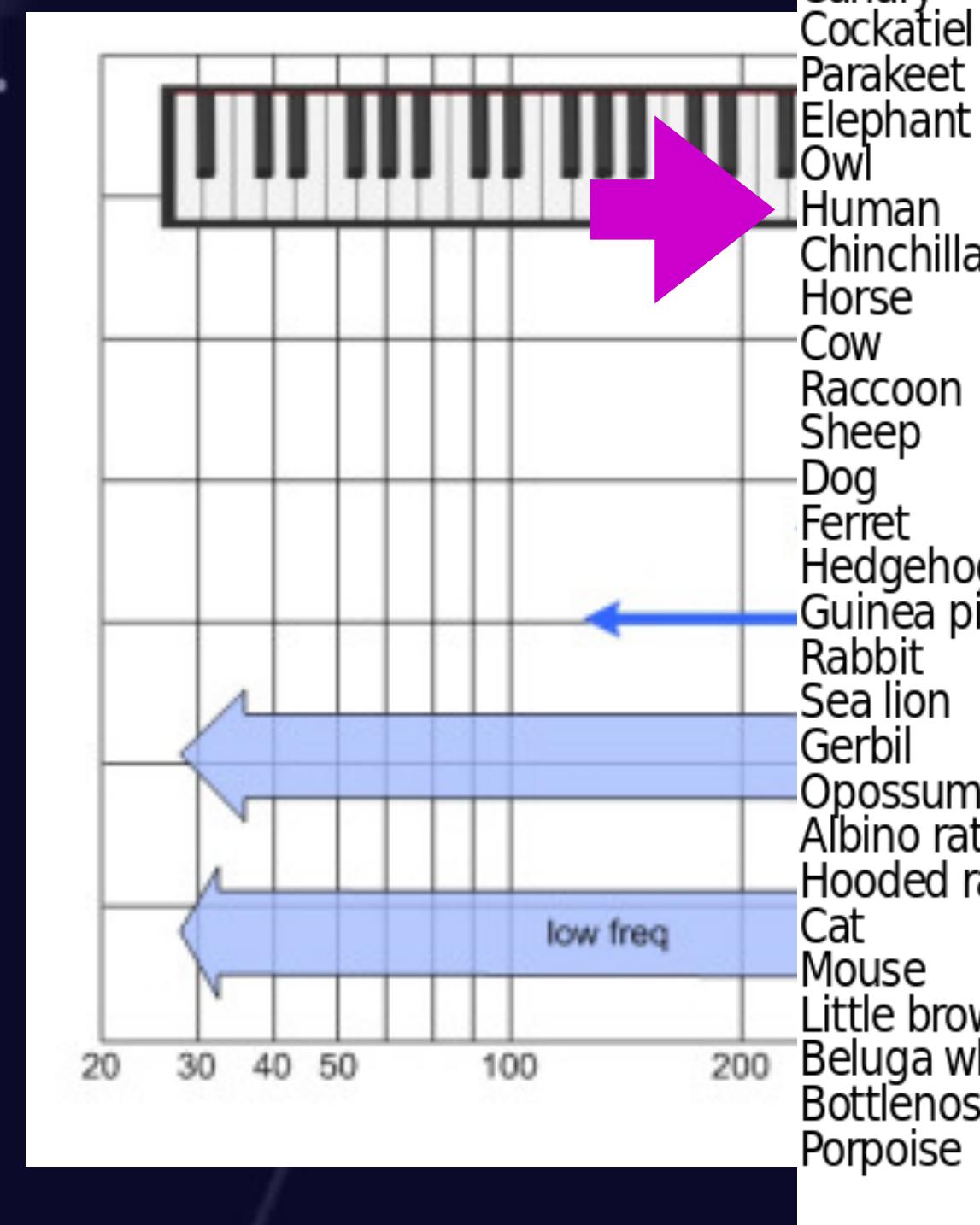
Sampling Rate

- Nyquist Theorem: Sampling rate $\geq 2 \times \text{max frequency}$



Voice Frequency

- Human audible range:
- 20 Hz to 20KHz.



Voice Frequency

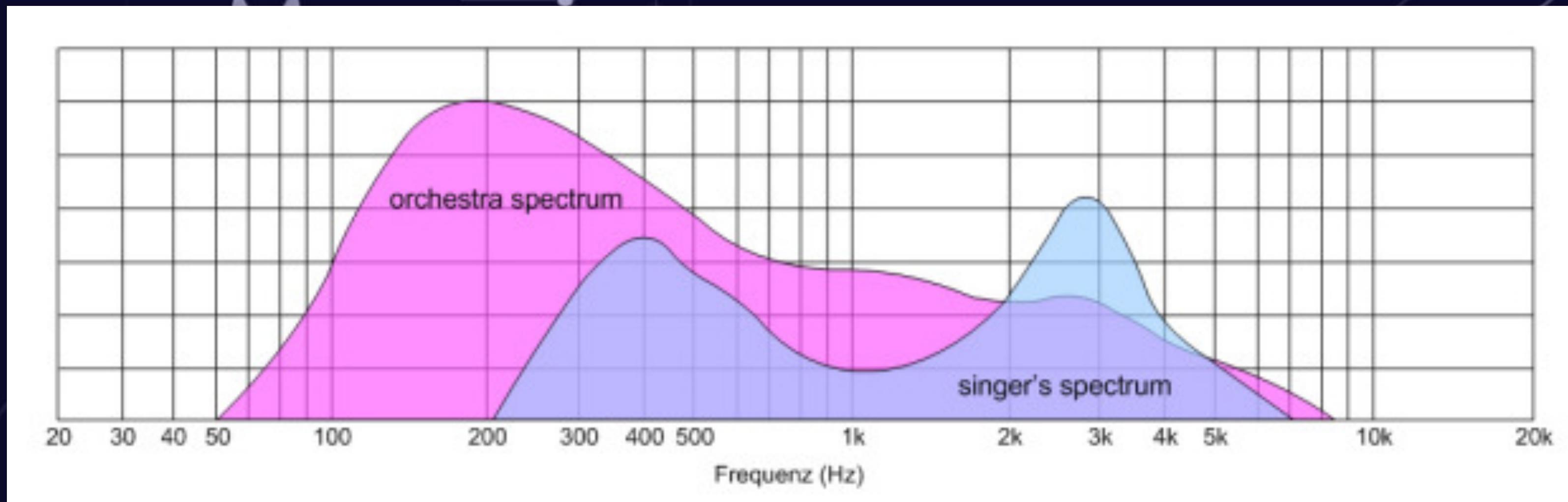


Image source: www.proav.de/audio/speech-level.html



This record has been made

Voice Frequency

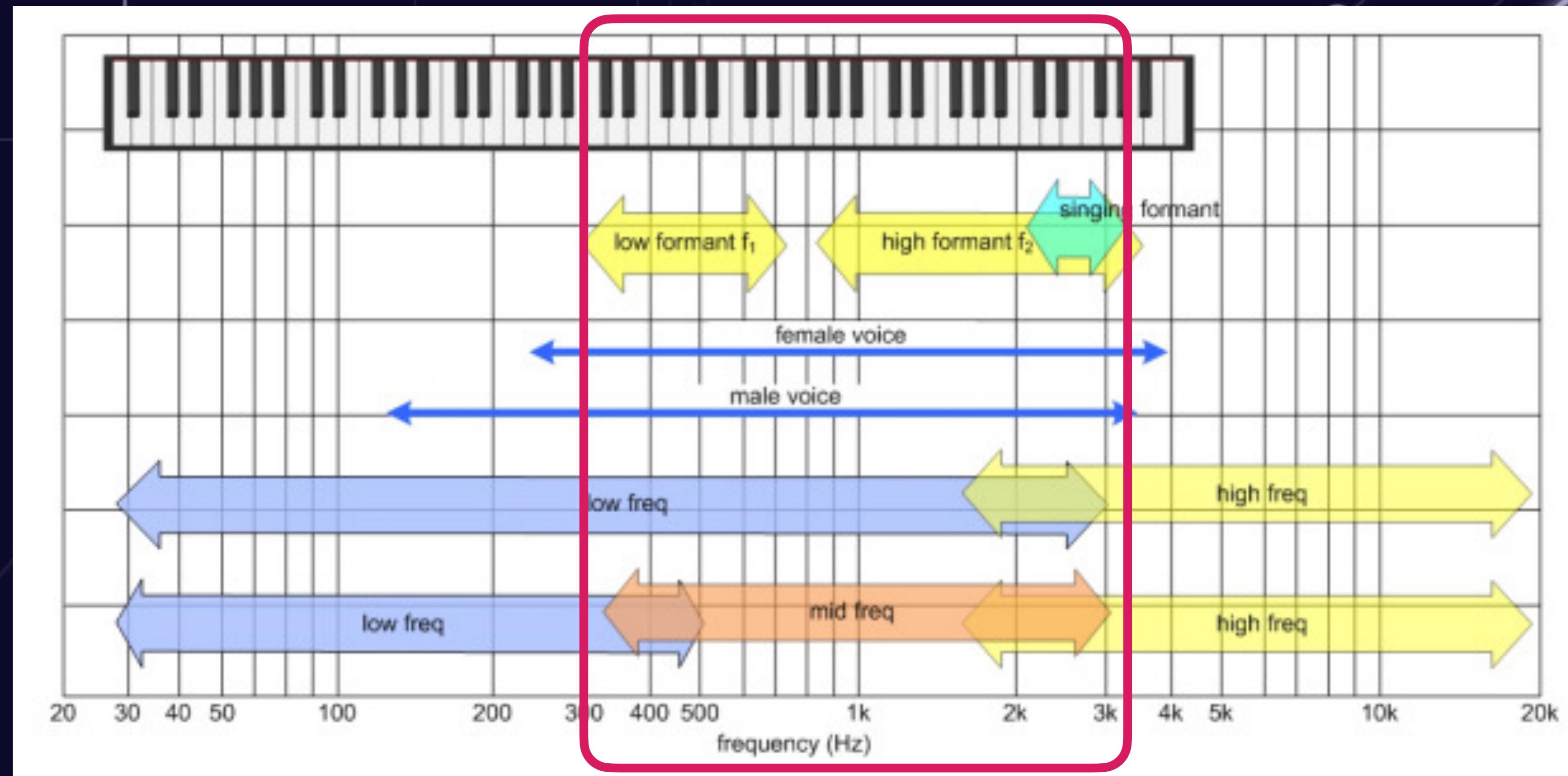


Image source: www.proav.de/audio/speech-level.html

Telephony, voice frequency range: 300Hz to 3400Hz.

Ohm Law

$$I = \frac{V}{R}$$

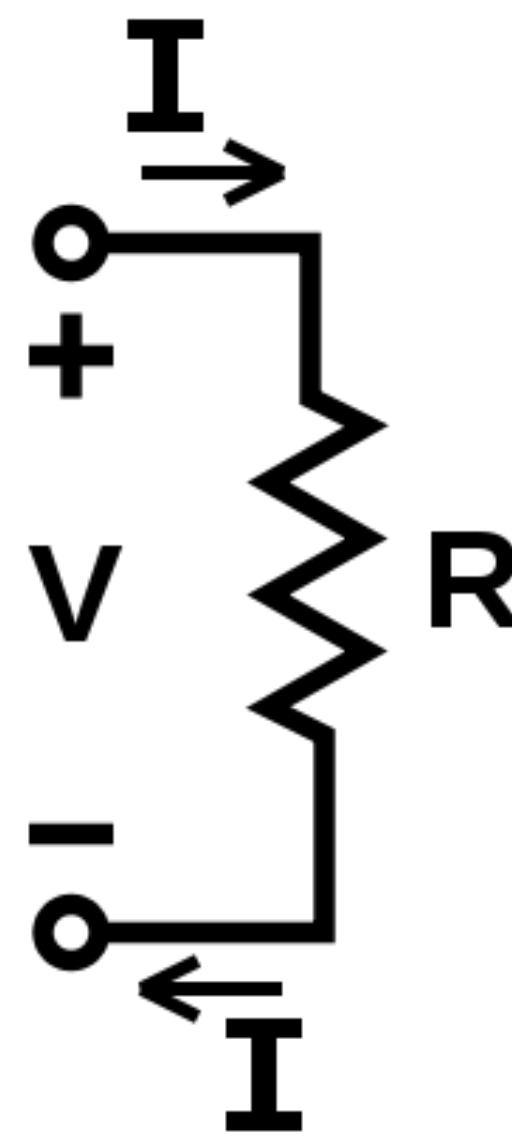


Image source: <http://stat.case.edu/~pillar/genealogy/ohm.gif>

Ohm Law

Ohm's Law Calculator

*Voltage (V) = Current (I) * Resistance (R)*
*Power (P) = Voltage (V) * Current (I)*

Enter any two known values and press "Calculate" to solve for the others.

Voltage (V)

3.3

Volts (V)

Current (I)

16.5

milliamps (mA)

Resistance (R)

200

ohms (Ω)

Power (P)

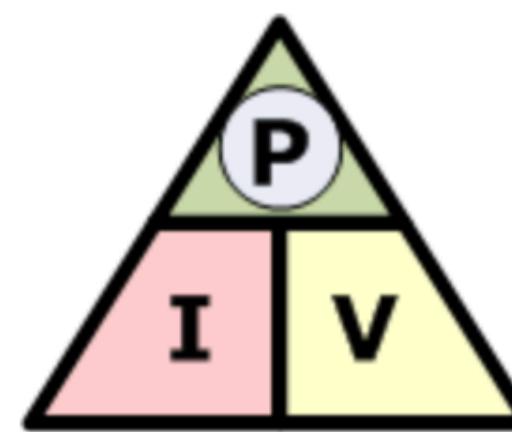
0.05445

Watts (W)

Calculate

Click "Calculate" to update the fields with orange borders.

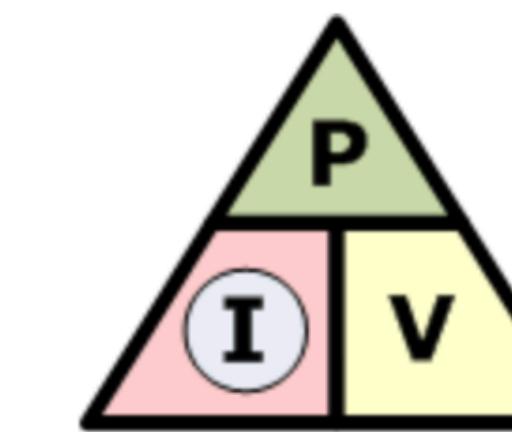
<http://>



$$\textcircled{P} = I \times V$$



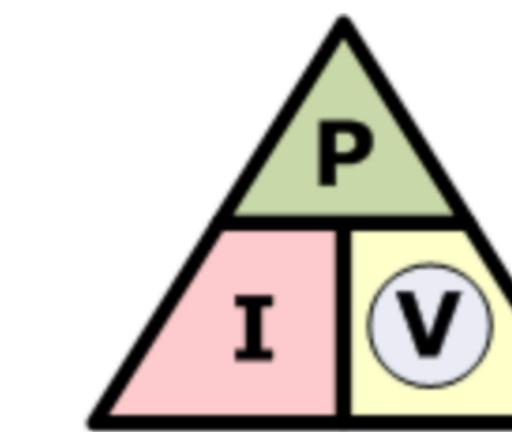
$$\textcircled{V} = I \times R$$



$$\textcircled{I} = \frac{P}{V}$$



$$\textcircled{I} = \frac{V}{R}$$



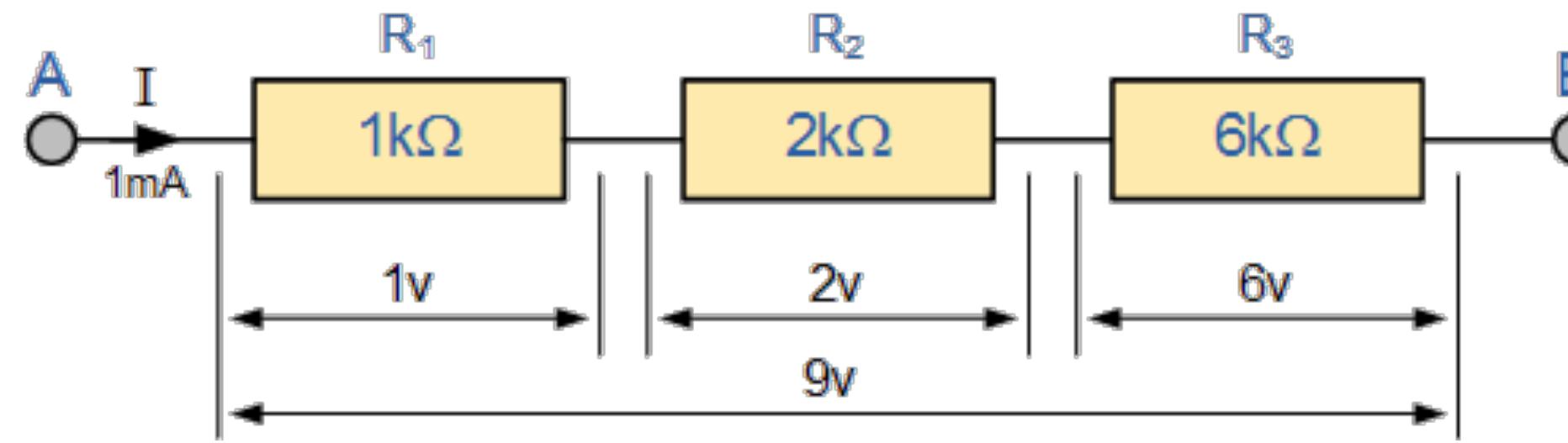
$$\textcircled{V} = \frac{P}{I}$$



$$\textcircled{R} = \frac{V}{I}$$

ulator

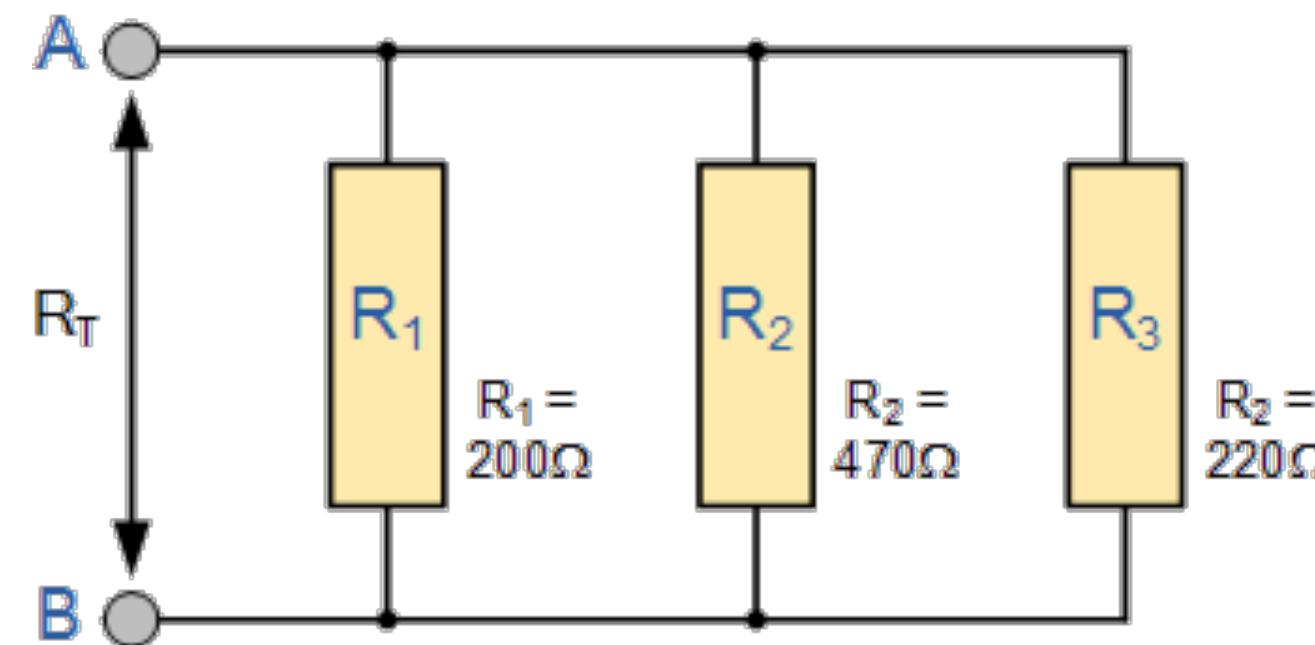
Resistors in Series



$$R_T = R_1 + R_2 + R_3$$

$$R_{EQ} = R_1 + R_2 + R_3 = 1\text{k}\Omega + 2\text{k}\Omega + 6\text{k}\Omega = 9\text{k}\Omega$$

Resistors in Parallel



$$\frac{1}{R_T} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} \dots + \frac{1}{R_n} \text{ etc}$$

$$\frac{1}{R_T} = \frac{1}{200} + \frac{1}{470} + \frac{1}{220}$$

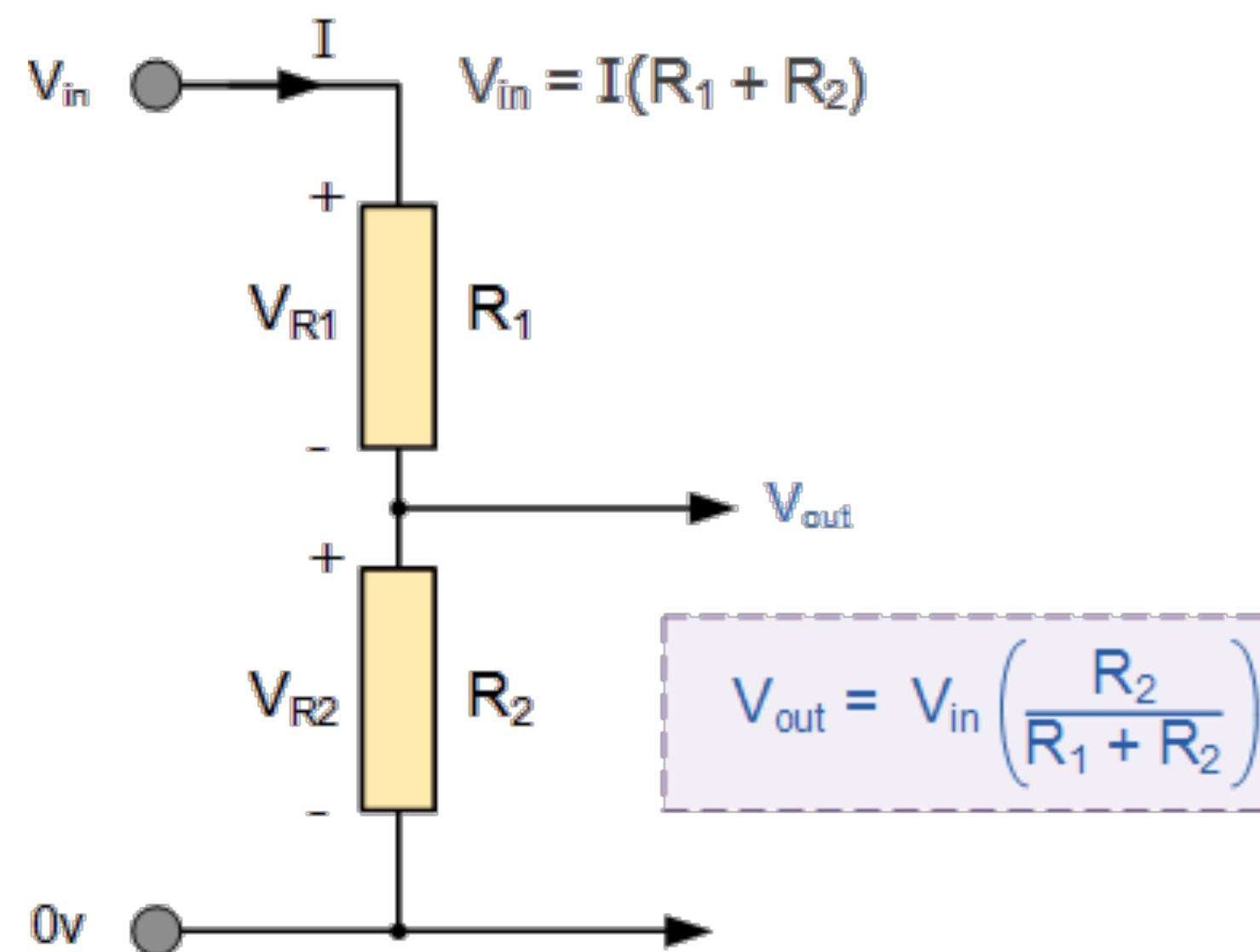
$$= \frac{1}{200} + \frac{1}{470} + \frac{1}{220} = 0.0117$$

$$\text{therefore: } R_T = \frac{1}{0.0117} = 85.67\Omega$$

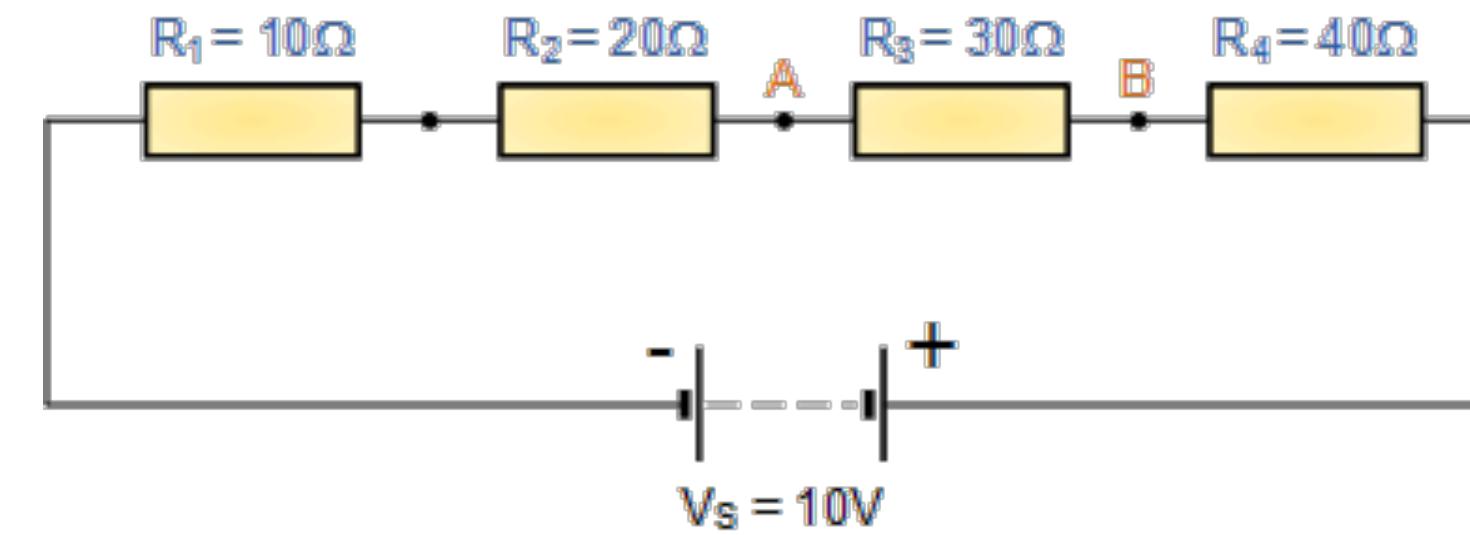
Voltage Divider Network

Kirchhoff's Voltage Law:

“the supply voltage in a closed circuit is equal to the sum of all the voltage drops ($I \cdot R$) around the circuit”



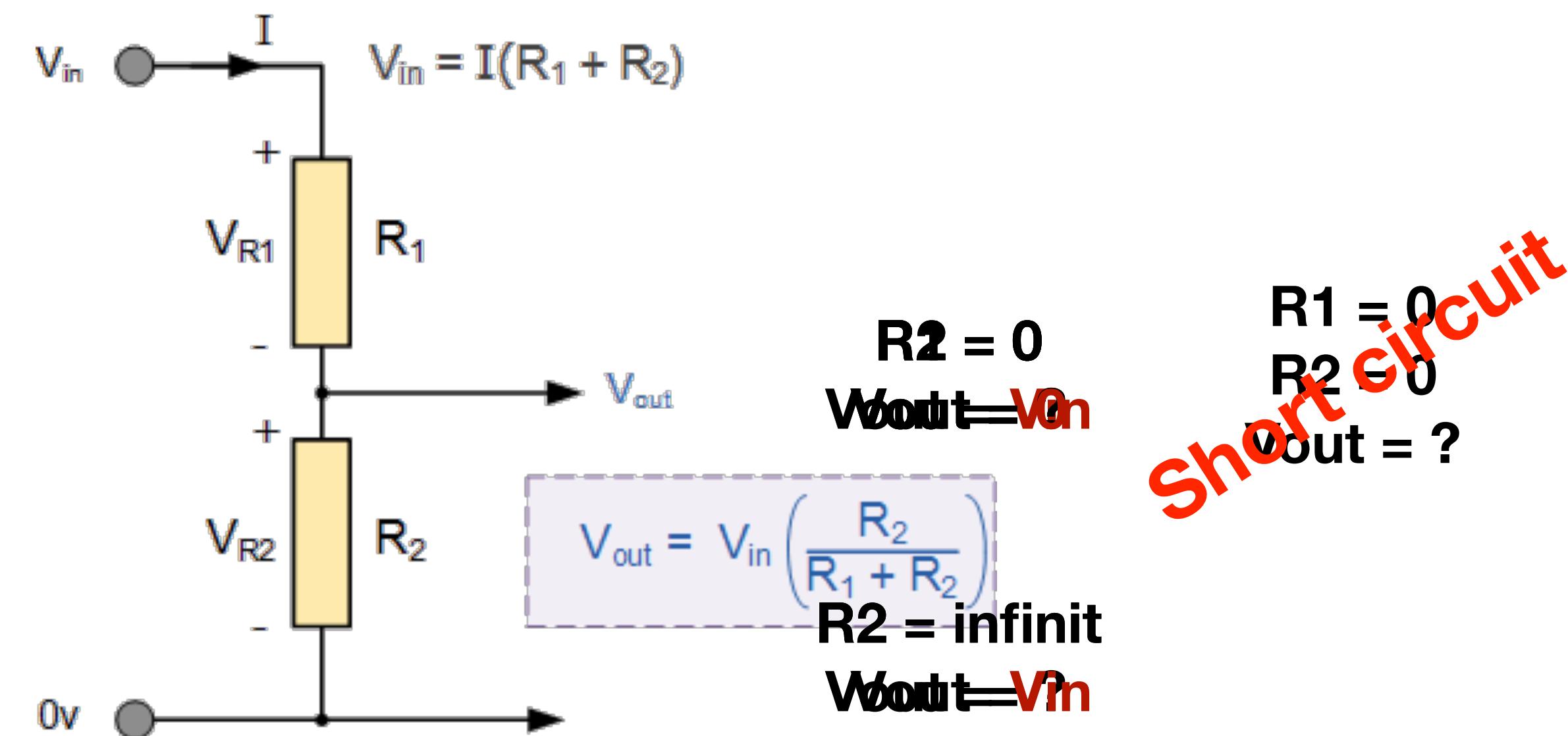
Voltage Divider Network



$$V_{AB} = V_{R3} = V_s \times \frac{R_3}{R_1 + R_2 + R_3 + R_4}$$

$$V_{AB} = 10 \times \frac{30}{10+20+30+40} = 10 \times 0.3 = 3V$$

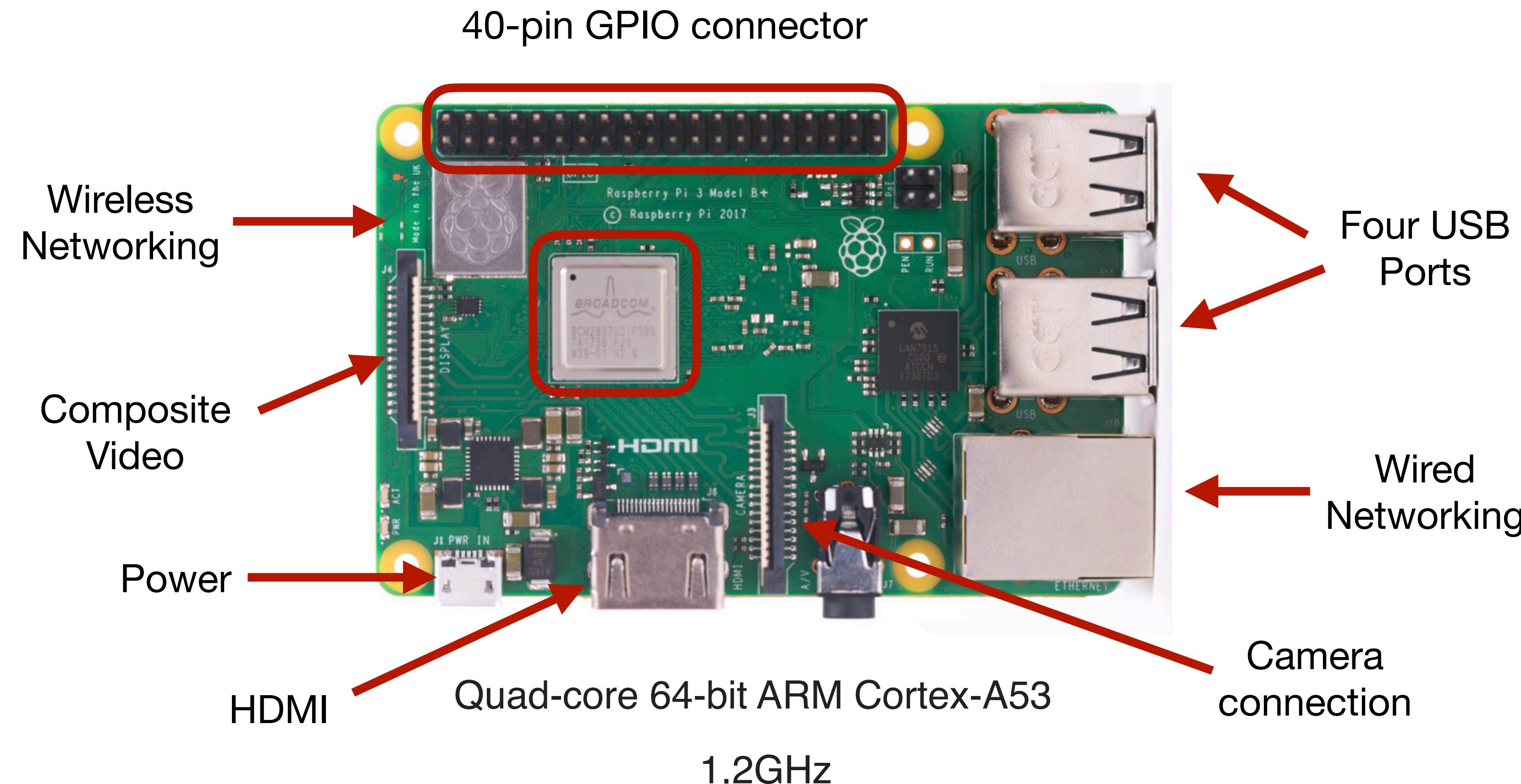
Voltage Divider Network



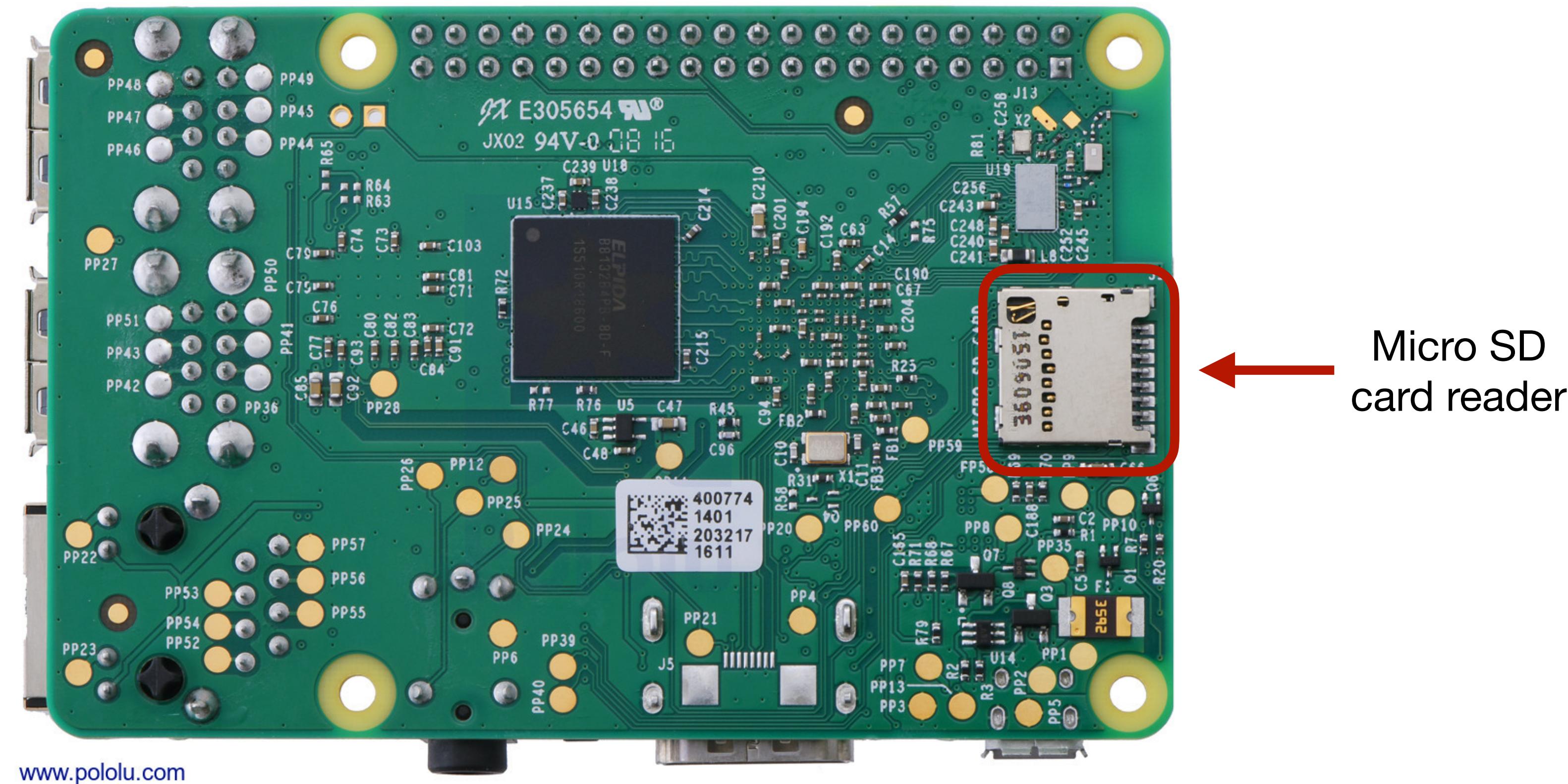
Peripheral I/O

- General Purpose Input/Output (GPIO)
- Pulse Width Modulation (PWM)
- Serial Communication

Raspberry Pi 3 Model B



Raspberry Pi 3 Model B



<https://developer.android.com/things/hardware/raspberrypi.html>

I/O Pinouts

GPIO Signal	Alternate Functions	
BCM2	I2C1 (SDA)	
BCM3	I2C1 (SCL)	
BCM7	SPI0 (SS1)	
BCM8	SPI0 (SS0)	
BCM9	SPI0 (MISO)	
BCM10	SPI0 (MOSI)	
BCM11	SPI0 (SCLK)	
BCM13	PWM1	
BCM14	UART0 (TXD)	MINUART (TXD)
BCM15	UART0 (RXD)	MINUART (RXD)
BCM18	I2S1 (BCLK)	PWM0
BCM19	I2S1 (LRCLK)	
BCM20	I2S1 (SDIN)	
BCM21	I2S1 (SDOUT)	

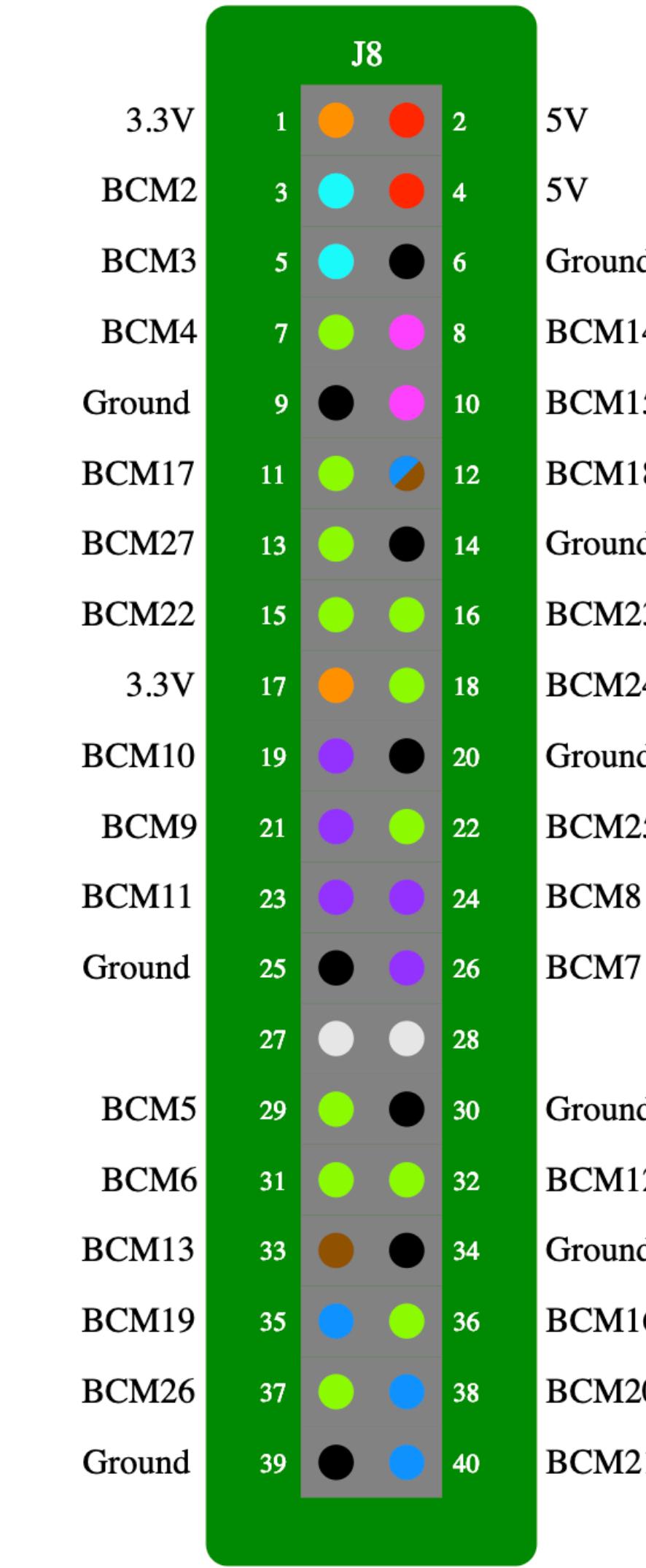
● = 5V
● = 3.3V

● = 1.8V
● = Ground

● = GPIO
● = PWM

● = I2C
● = I2S

● = SPI
● = UART

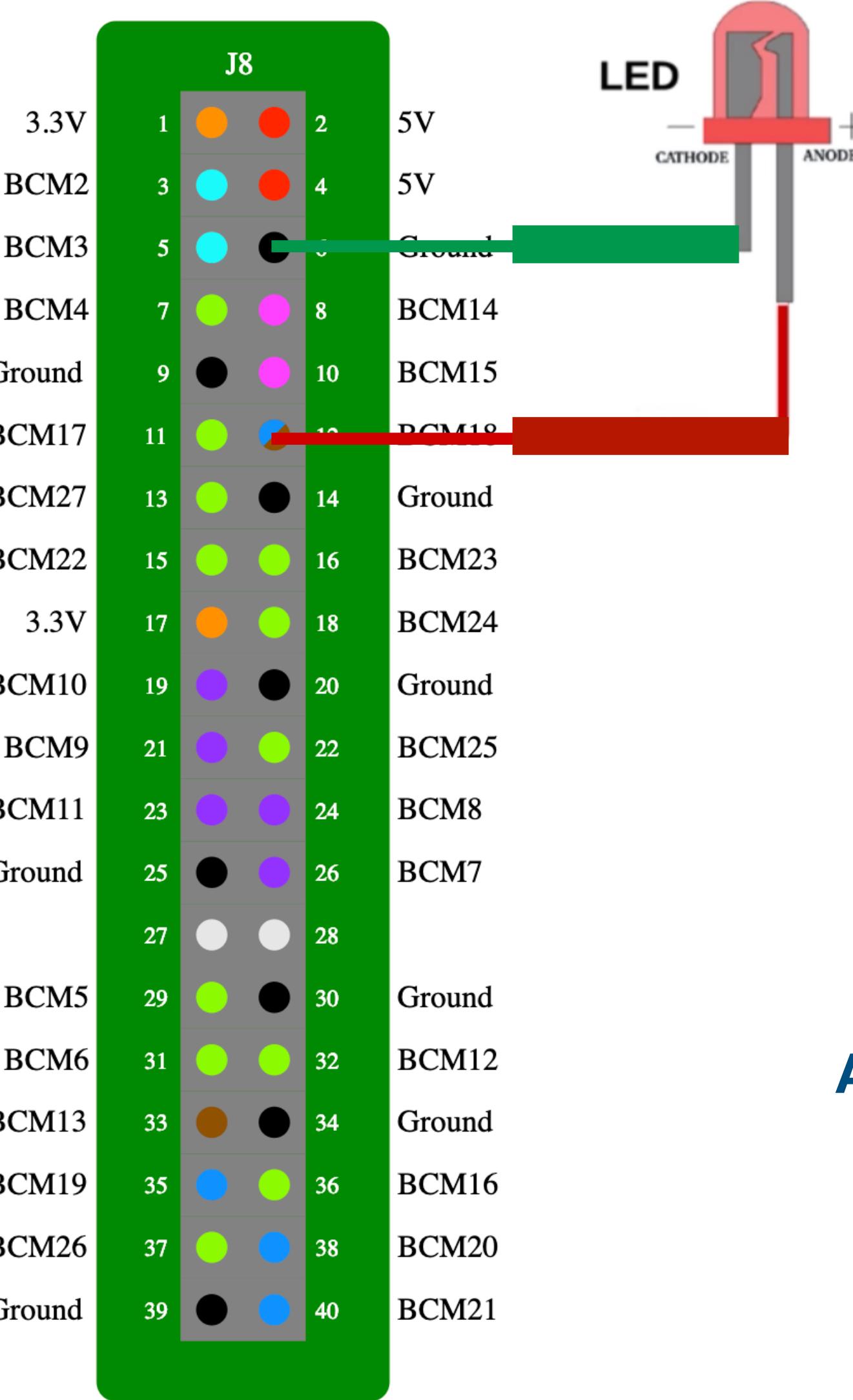


Output and Input

- Digital Pins
 - Two values:
 - Low (0)
 - High (1)
 - Acts either:
 - Battery (OUTPUT)
 - Voltage meter (INPUT)

J8				
3.3V	1	Orange	2	5V
BCM2	3	Cyan	4	5V
BCM3	5	Cyan	6	Ground
BCM4	7	Green	8	BCM14
Ground	9	Black	10	BCM15
BCM17	11	Green	12	BCM18
BCM27	13	Green	14	Ground
BCM22	15	Green	16	BCM23
3.3V	17	Orange	18	BCM24
BCM10	19	Purple	20	Ground
BCM9	21	Purple	22	BCM25
BCM11	23	Purple	24	BCM8
Ground	25	Black	26	BCM7
	27	White	28	
BCM5	29	Green	30	Ground
BCM6	31	Green	32	BCM12
BCM13	33	Brown	34	Ground
BCM19	35	Blue	36	BCM16
BCM26	37	Green	38	BCM20
Ground	39	Black	40	BCM21

Output



- Pin 12 can act like a battery (OUTPUT)

- LOW - 0V

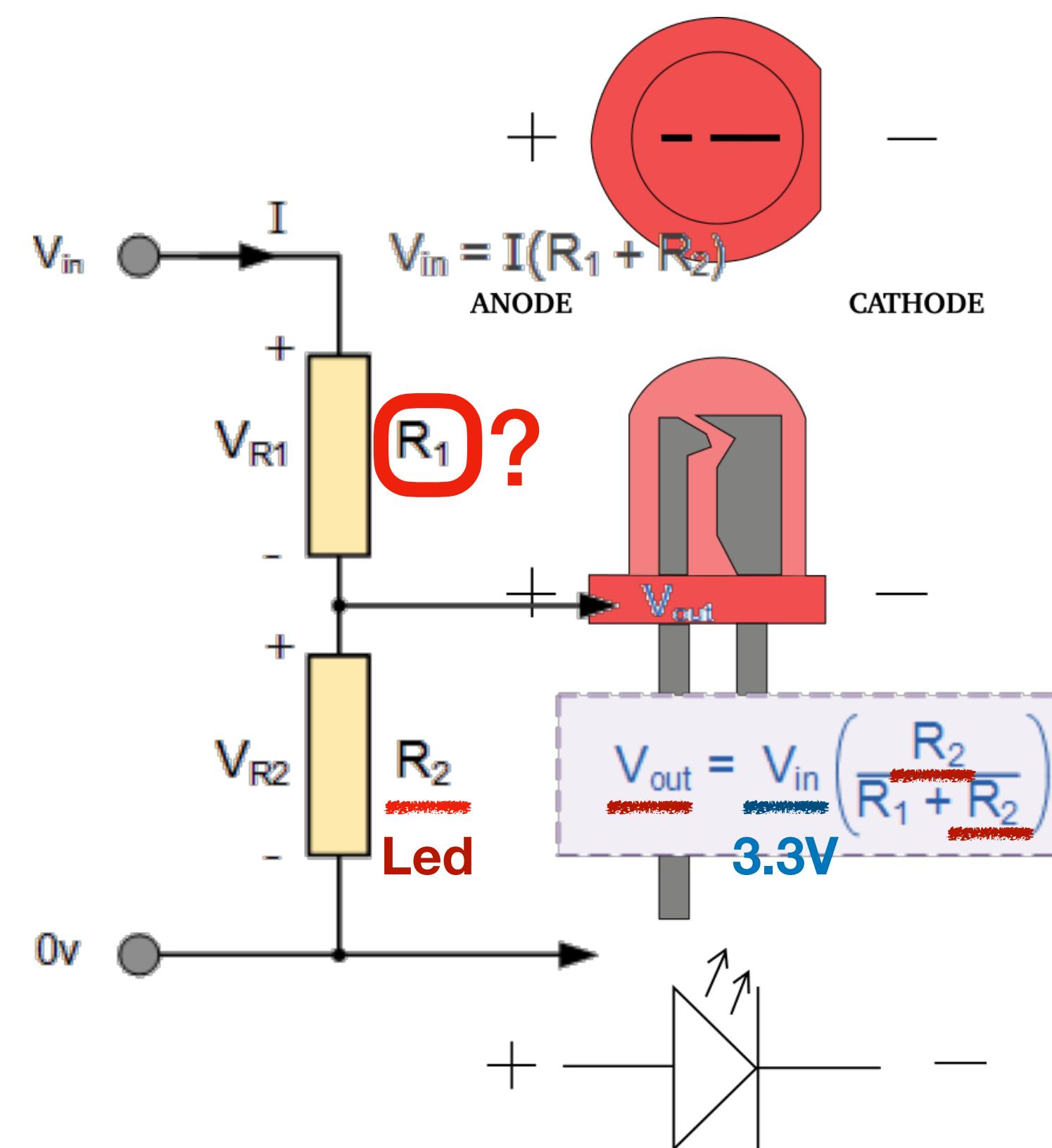
- HIGH - 5V, 3.3V or 1.8V

Arduino

Raspberry Pi

Others

What resistor to use?



Led

LED - Basic Red 5mm

COM-09590 ROHS ✓ J5 F 3D

\$0.35

Volume sales pricing

- 1 +

ADD TO CART

Quantity discounts available

DESCRIPTION

FEATURES

DOCUMENTS

The diagram consists of three identical triangles. Each triangle has a yellow top section containing a circled 'V'. The bottom-left section is pink and contains a circled 'I' with a horizontal bar underneath it, representing current. The bottom-right section is purple and contains a circled 'R' with a horizontal bar underneath it, representing resistance.

$$\mathbf{V} = \mathbf{I} \times \mathbf{R}$$
$$\mathbf{I} = \frac{\mathbf{V}}{\mathbf{R}}$$
$$\mathbf{R} = \frac{\mathbf{V}}{\mathbf{I}}$$

LEDs - those blinky things. A must have for power indication, pin status, opto-electronic sensors, and fun blinky displays.

This is a very basic 5mm LED with a red lens. It has a typical forward voltage of 2.0V and a rated forward current of 20mA.

Compute R1

LED - Basic Red 5mm
 COM-09590 ROHS✓ J5 3D

\$0.35
 Volume sales pricing

- 1 +
 Quantity discounts available

DESCRIPTION FEATURES DOCUMENTS 0v **V_{out}**

ADD TO CART

$R_2 = \frac{V_{in} - V_{out}}{I}$

$R_1 = \frac{V_{out}}{I}$

$R_1 = \frac{V_{out}}{I} = \frac{V_{out}}{0.020A} = 65\Omega$



$$R = \frac{V}{I}$$

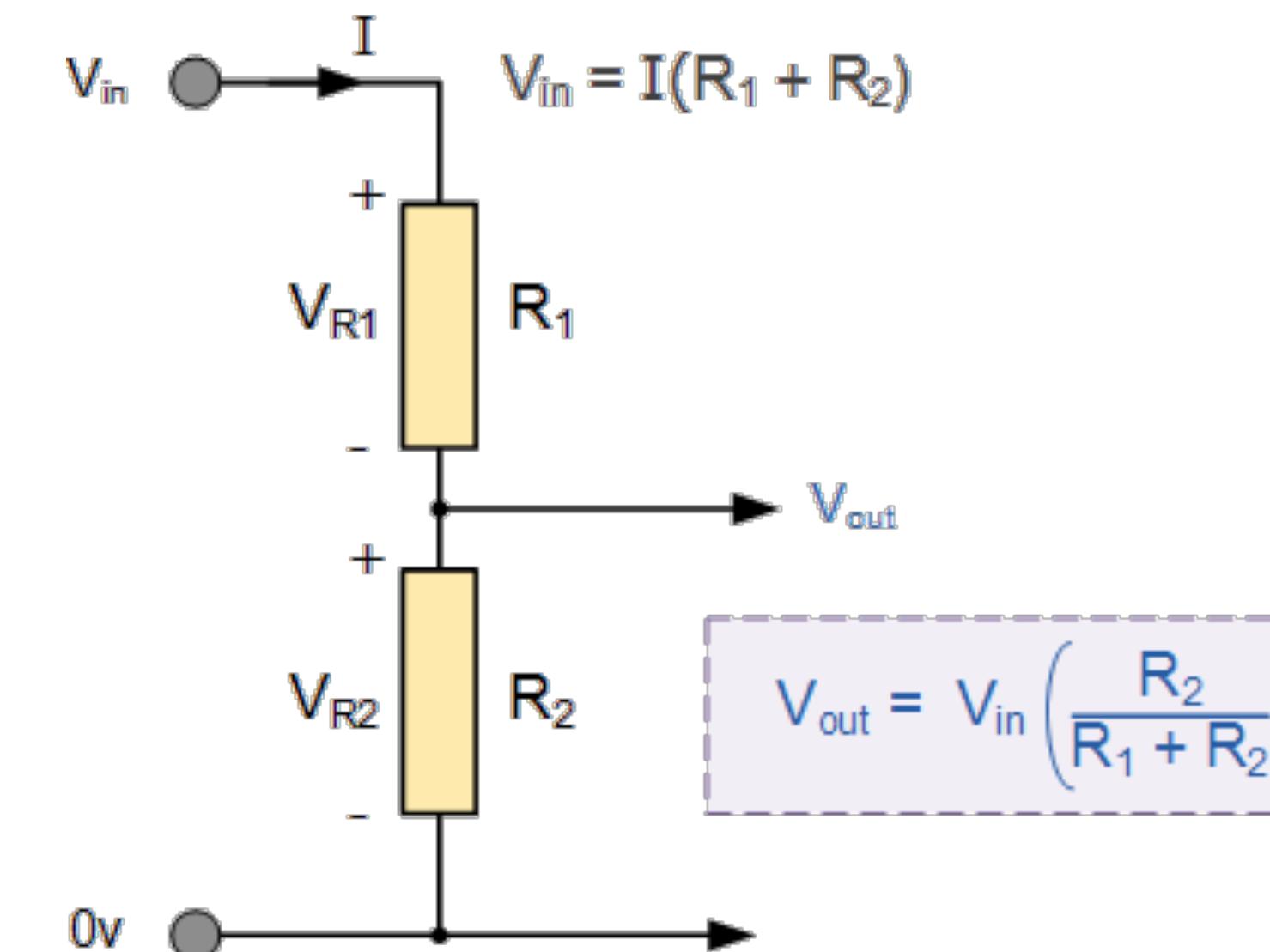
LEDs - those blinky things. A must have for power indication, pin status, opto-electronic sensors, and fun blinky displays.

This is a very basic 5mm LED with a red lens. It has a typical forward voltage of 2.0V and a rated forward current of 20mA.

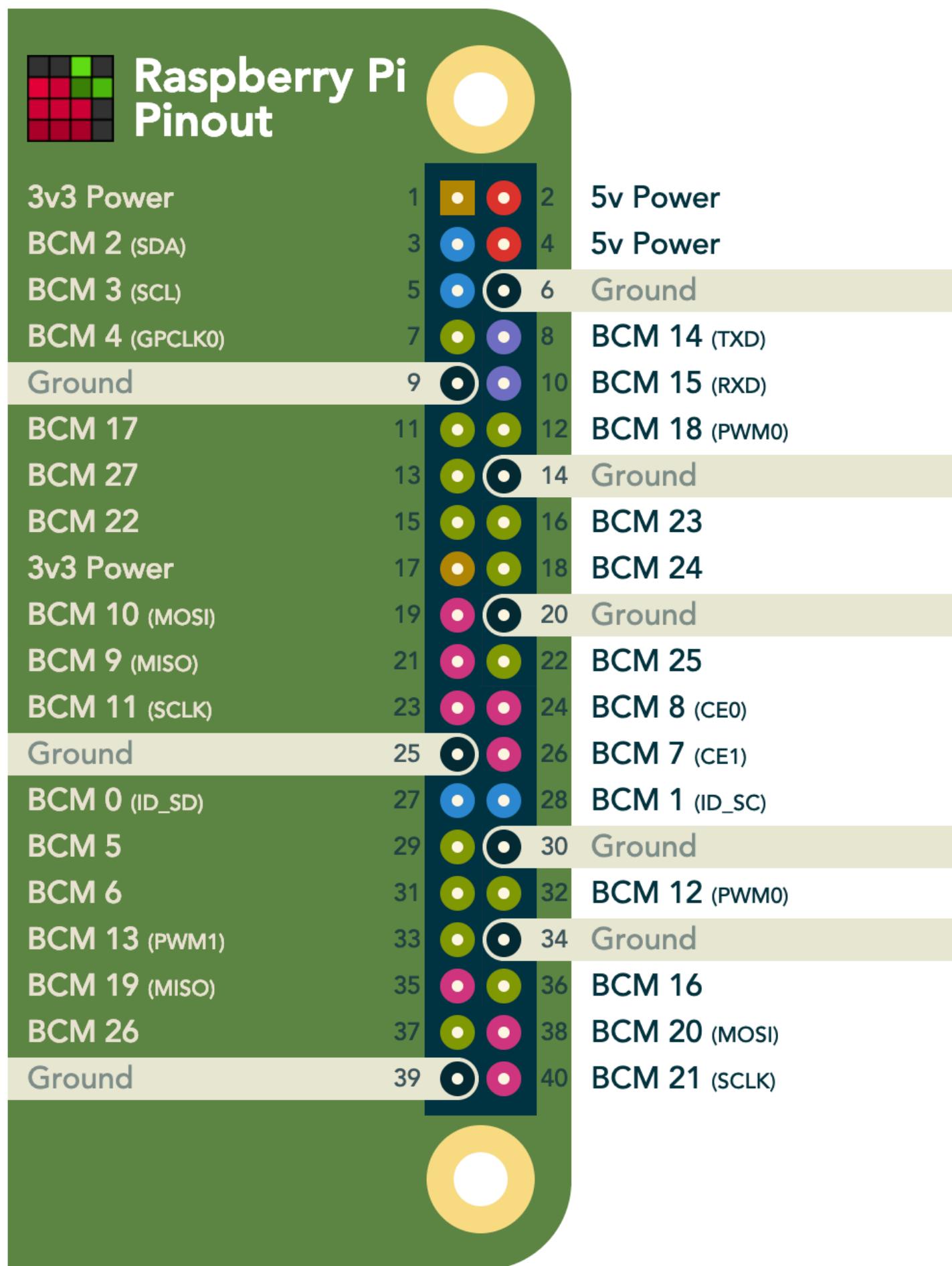
Input

J8			
3.3V	1	2	5V
BCM2	3	4	5V
BCM3	5	6	Ground
BCM4	7	8	BCM14
Ground	9	10	BCM15
BCM17	11	12	BCM18
BCM27	13	14	Ground
BCM22	15	16	BCM23
3.3V	17	18	BCM24
BCM10	19	20	Ground
BCM9	21	22	BCM25
BCM11	23	24	BCM8
Ground	25	26	BCM7
	27	28	
BCM5	29	30	Ground
BCM6	31	32	BCM12
BCM13	33	34	Ground
BCM19	35	36	BCM16
BCM26	37	38	BCM20
Ground	39	40	BCM21

- In this case we are measuring V_{out}
- Represented on one bit.



I/O Pinouts



Ground	DPI	GPCLK	JTAG	1-WIRE	PCM	SDIO	I2C	SPI	UART	WiringPi
Browse more HATs, pHATs and add-ons »										
Arcade Bonnet Connect joystick, buttons and speakers to your Pi	MotoZero Control 4 motors from your Raspberry Pi	XBee Shield Use XBee modules with the Raspberry Pi	Score:Zero A super-simple and stylish soldering kit - makes an NES-style games controller when assembled.							

Ground

The Ground pins on the Raspberry Pi are all electrically connected, so it doesn't matter which one you use if you're wiring up a voltage supply.

Generally the one that's most convenient or closest to the rest of your connections is tidier and easier, or alternatively the one closest to the supply pin that you use.

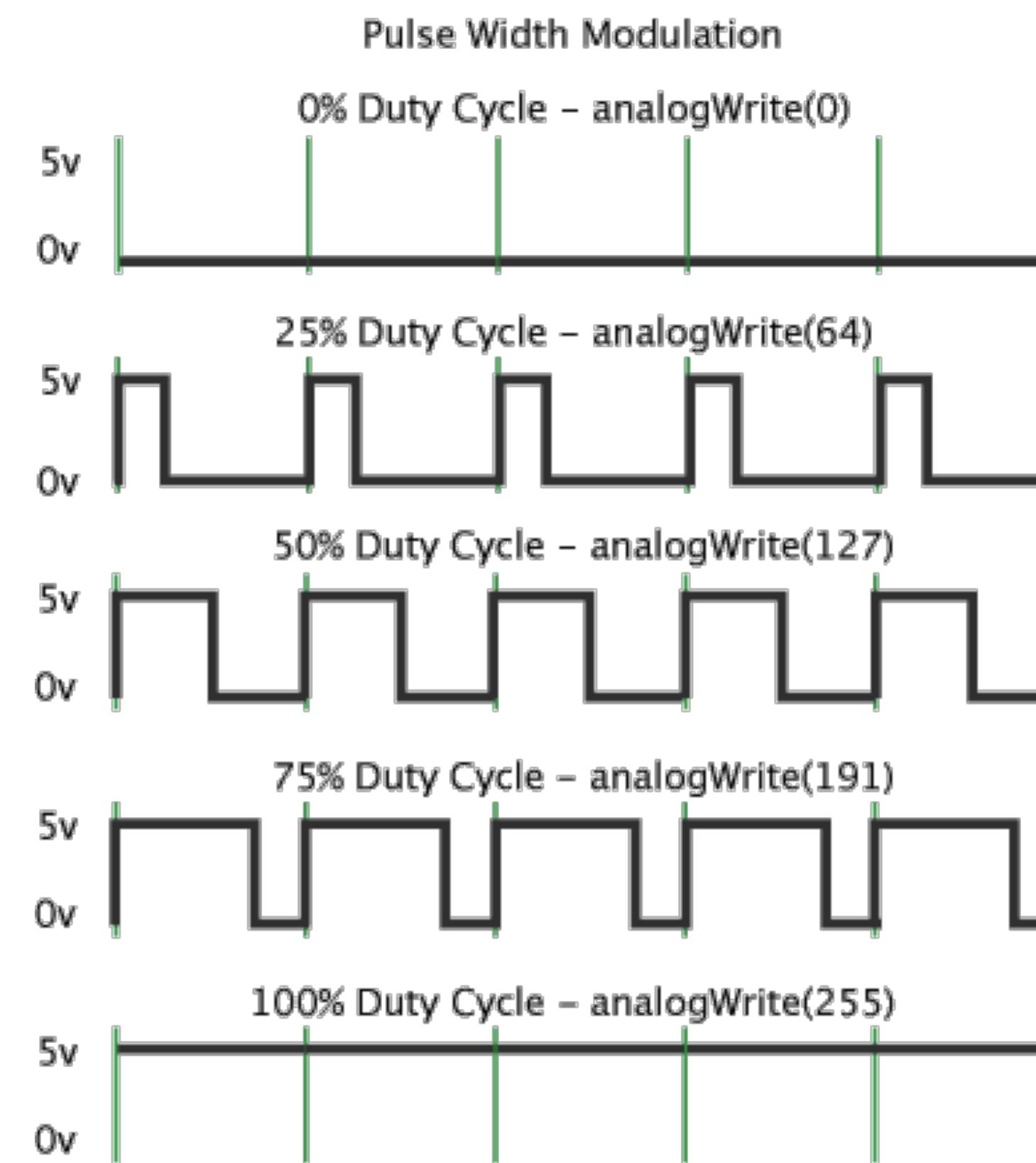
For example, it's a good idea to use Physical Pin 17 for 3v3 and Physical Pin 25 for ground when using the SPI connections, as these are right next to the important pins for SPI0.

Details

- 1 pin header
- Uses 8 GPIO pins

Pulse Width Modulation

PWM

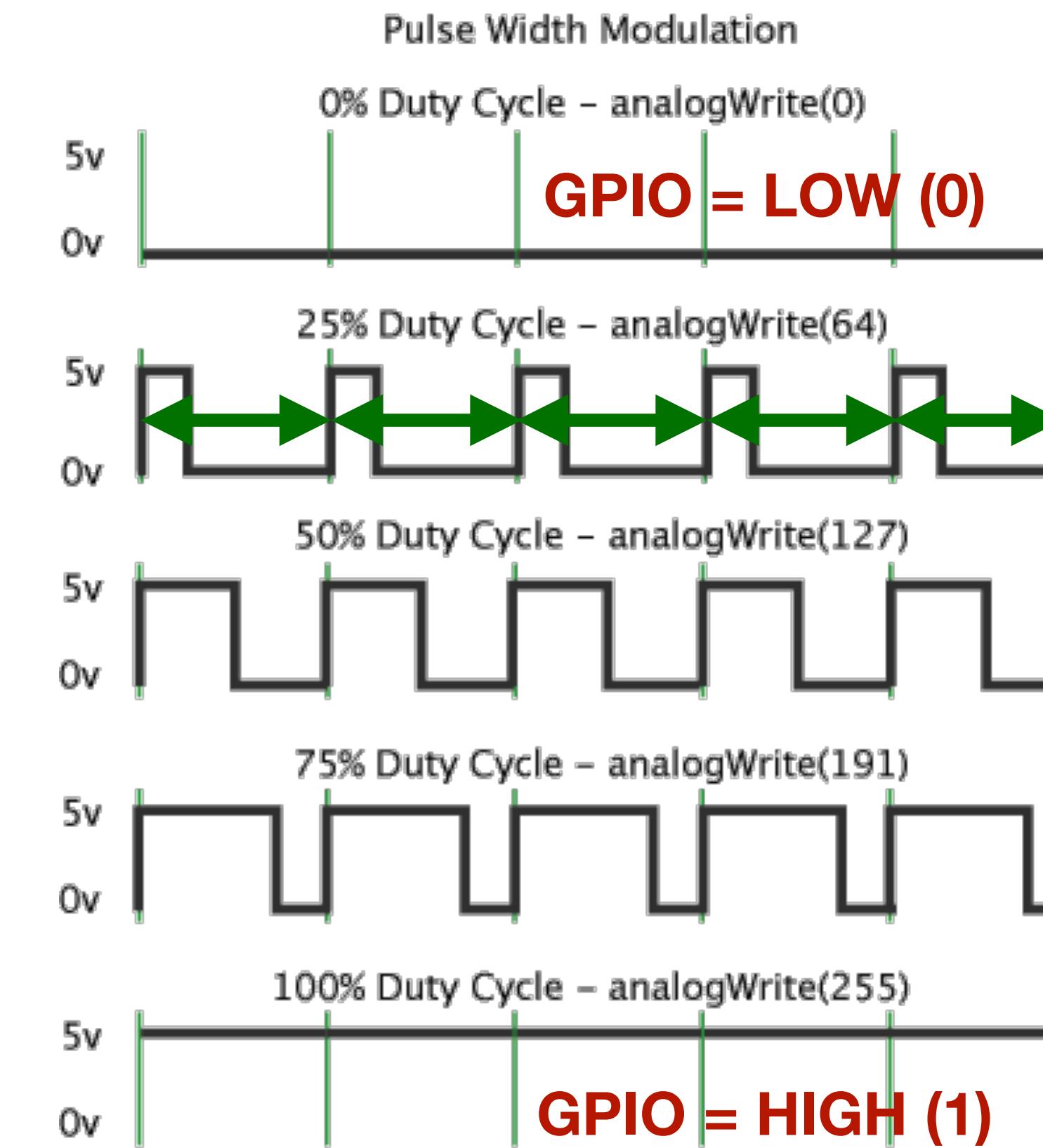


Pulse Width Modulation

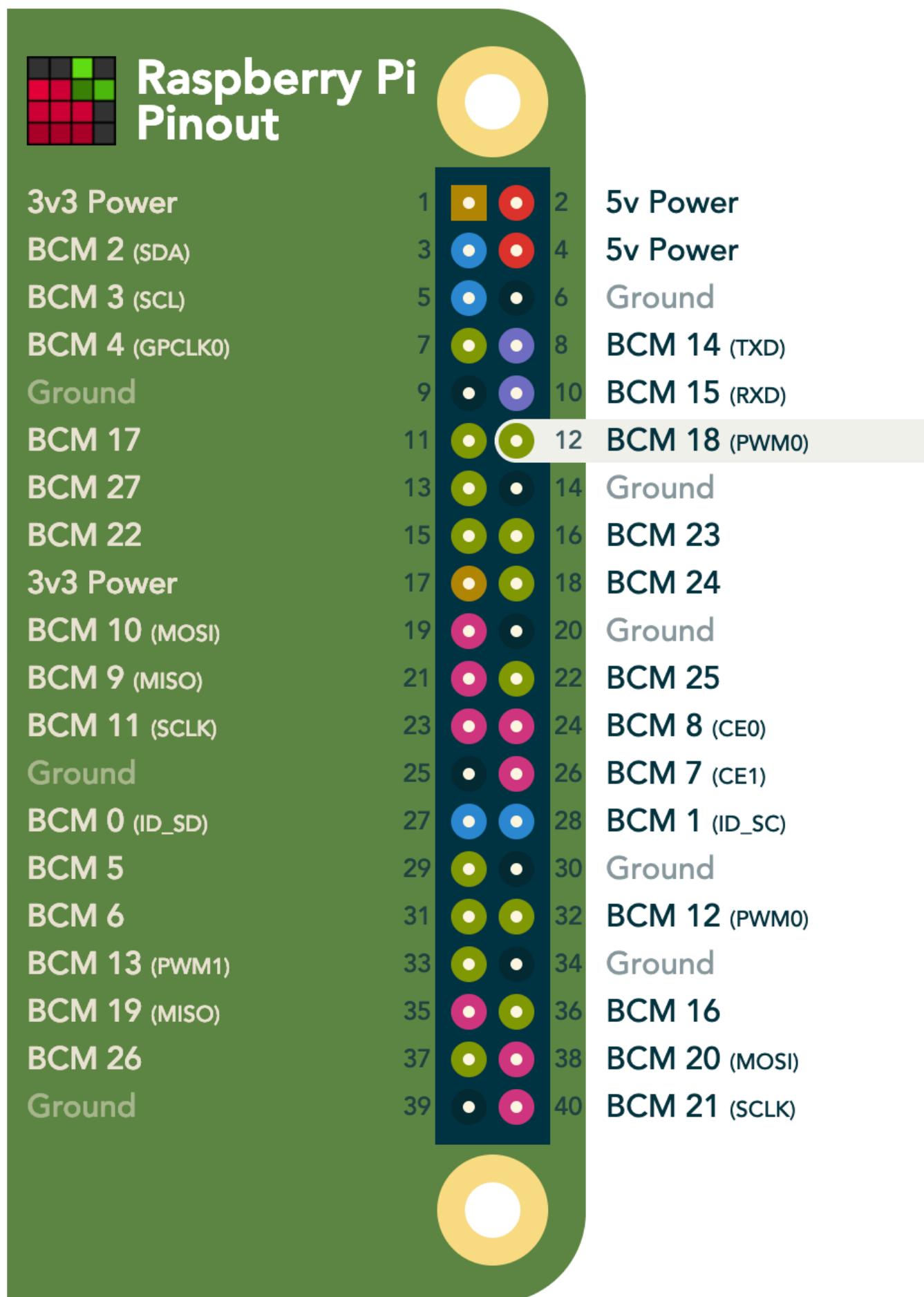
PWM

- PWM Pins
 - Capable to generate square wave signals.
 - We control the % of the duty cycle:
 - 0 - 0%
 - 255 - 100% (check hardware specs)
 - Possible usage:
 - Led dimming.
 - Control a servo motor.

↔ Between 1.1Hz and 19MHz



Analog to Digital Convertors



Ground DPI GPCLK JTAG 1-WIRE PCM SDIO I2C SPI UART WiringPi

Browse more HATs, pHATs and add-ons »

Arcade Bonnet
Connect joystick, buttons and speakers to your Pi

MotoZero
Control 4 motors from your Raspberry Pi

XBee Shield
Use XBee modules with the Raspberry Pi

Score:Zero
A super-simple and stylish soldering kit - makes an NES-style games controller when assembled.

BCM 18 (PWM0)

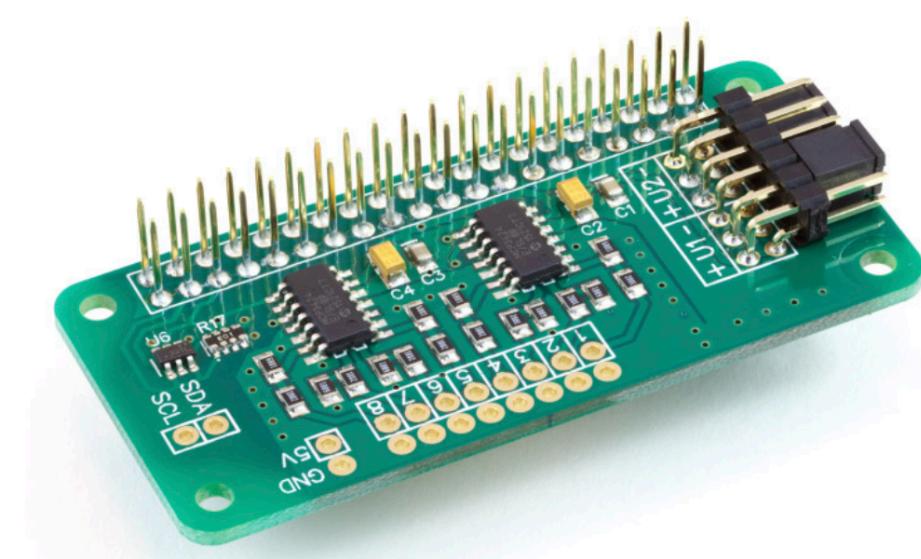
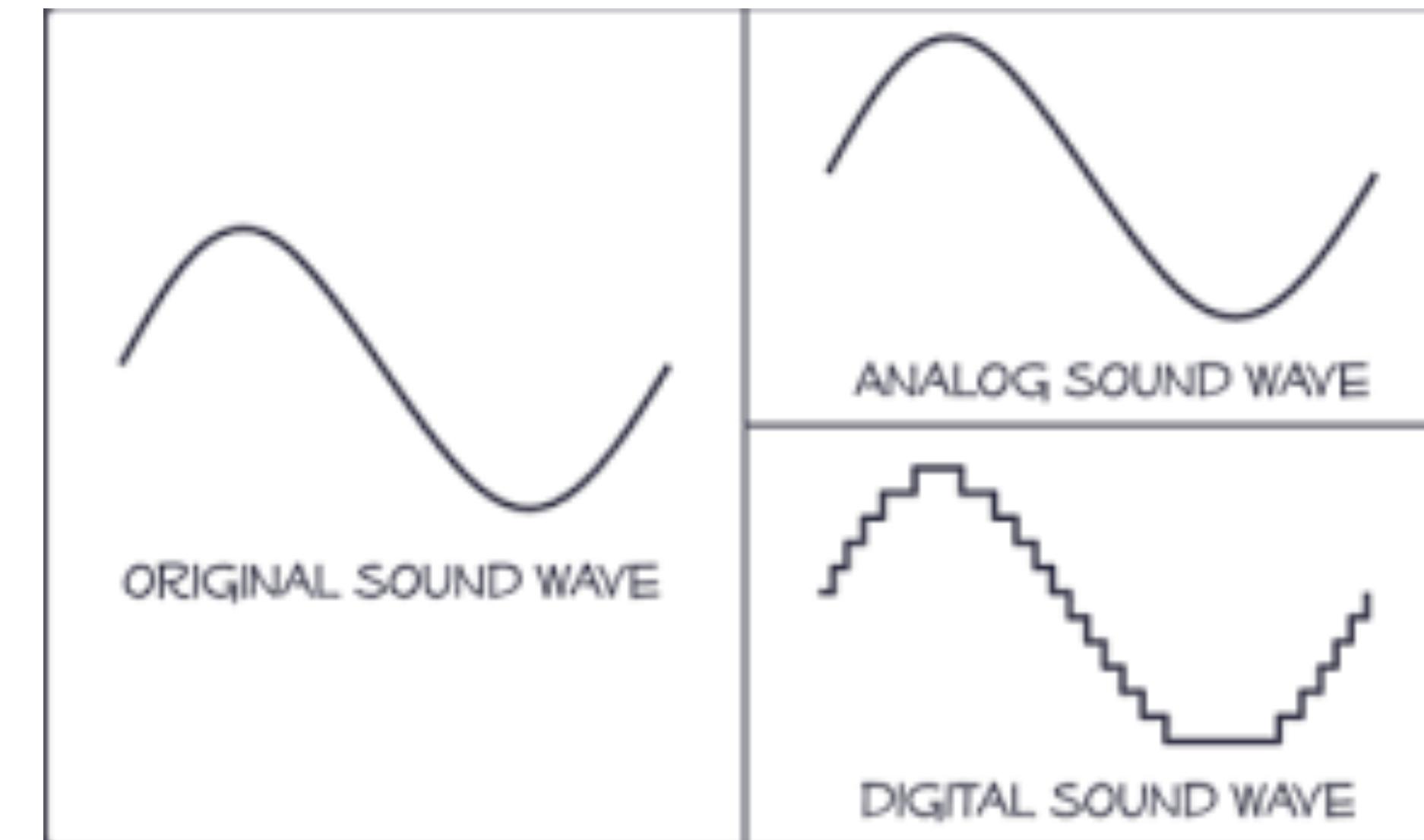
Alt0	Alt1	Alt2	Alt3	Alt4	Alt5
PCM CLK	SMI SD10	DPI D14	I2CSL SDA / MOSI	SPI1 CE0	PWM0

- Physical pin 12
- BCM pin 18
- Wiring Pi pin 1

The PWM0 output of BCM 18 is particularly useful, in combination with some fast, direct memory access trickery, for driving tricky devices with very specific timings. The WS2812 LEDs on the [Unicorn HAT](#) are a good example of this in action.

Analog to Digital Convertors

- When reading or writing digital signals we need to establish values for:
 - Bits per sample.
 - 1 bit - GPIO Input.
 - n bits - $0..2^n - 1$
 - Sampling rate.
ADC Pi

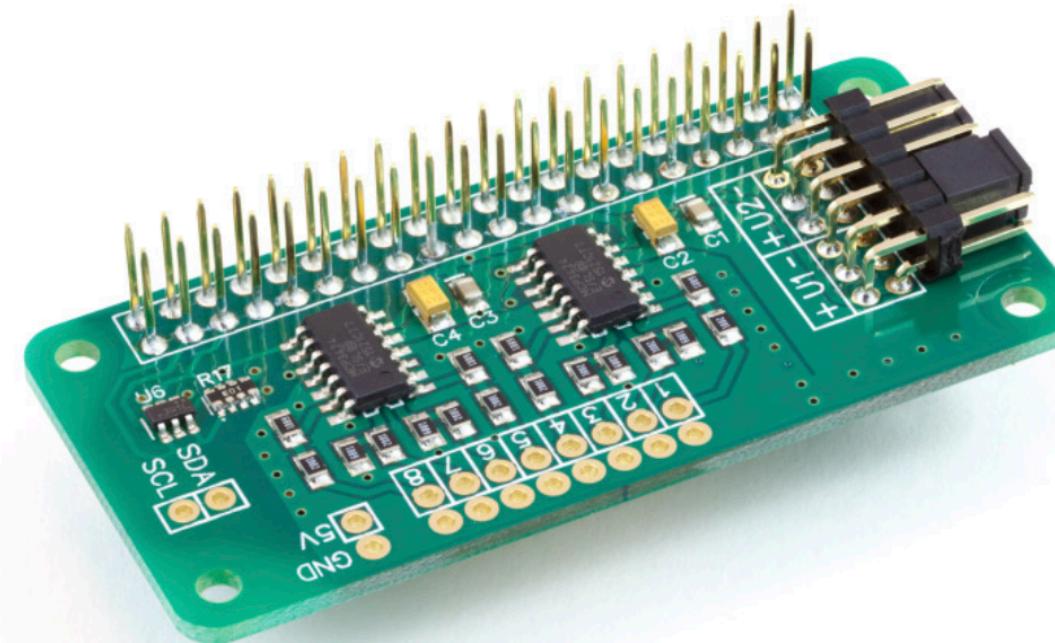


Analog to Digital Convertors

Features

- 8 x 17-bit 0 to 5V Single Ended Inputs
 - Control via the Raspberry Pi I2C port
 - Stack up to 4 ADC Pi boards on a single Raspberry Pi
 - ...

ADC Pi



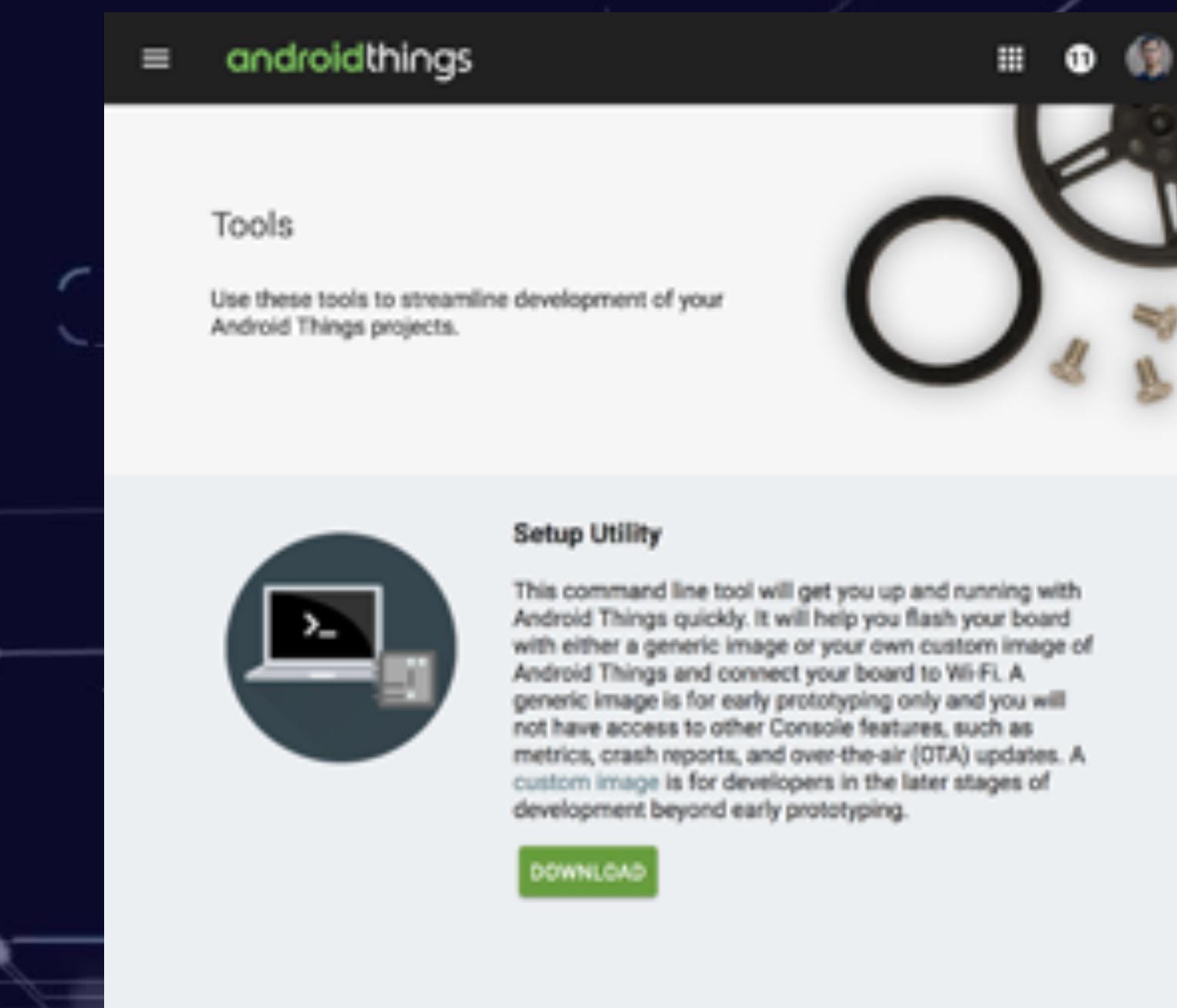
Setup



Image source: <https://rehoff.me>

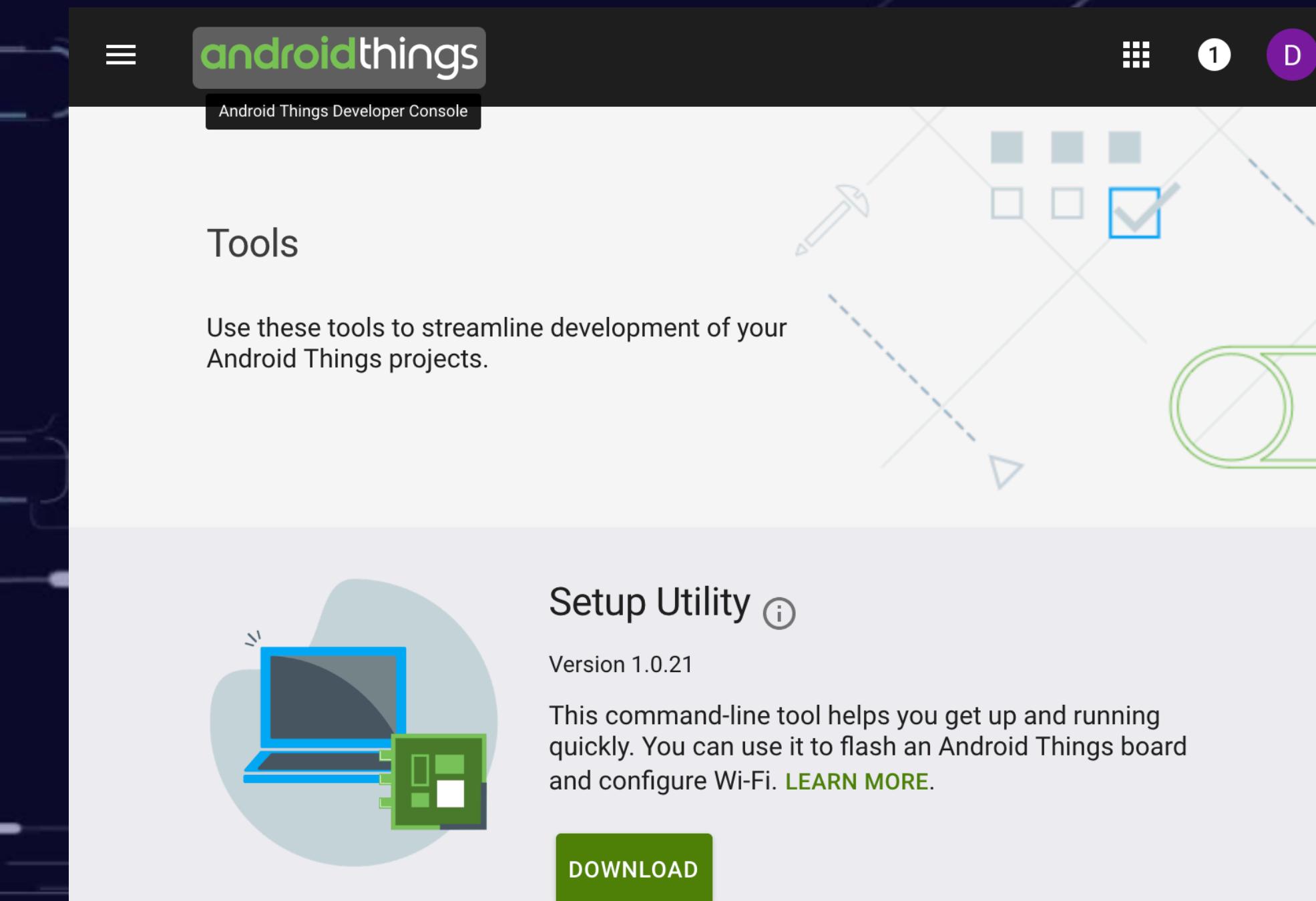
Flashing the Image

- Prerequisite:
 - Micro-USB cable.
 - Ethernet cable.
 - MicroSD card reader.
 - 8GB or larger microSD card.
- Optional items:
 - HDMI cable.
 - HDMI-enabled display.



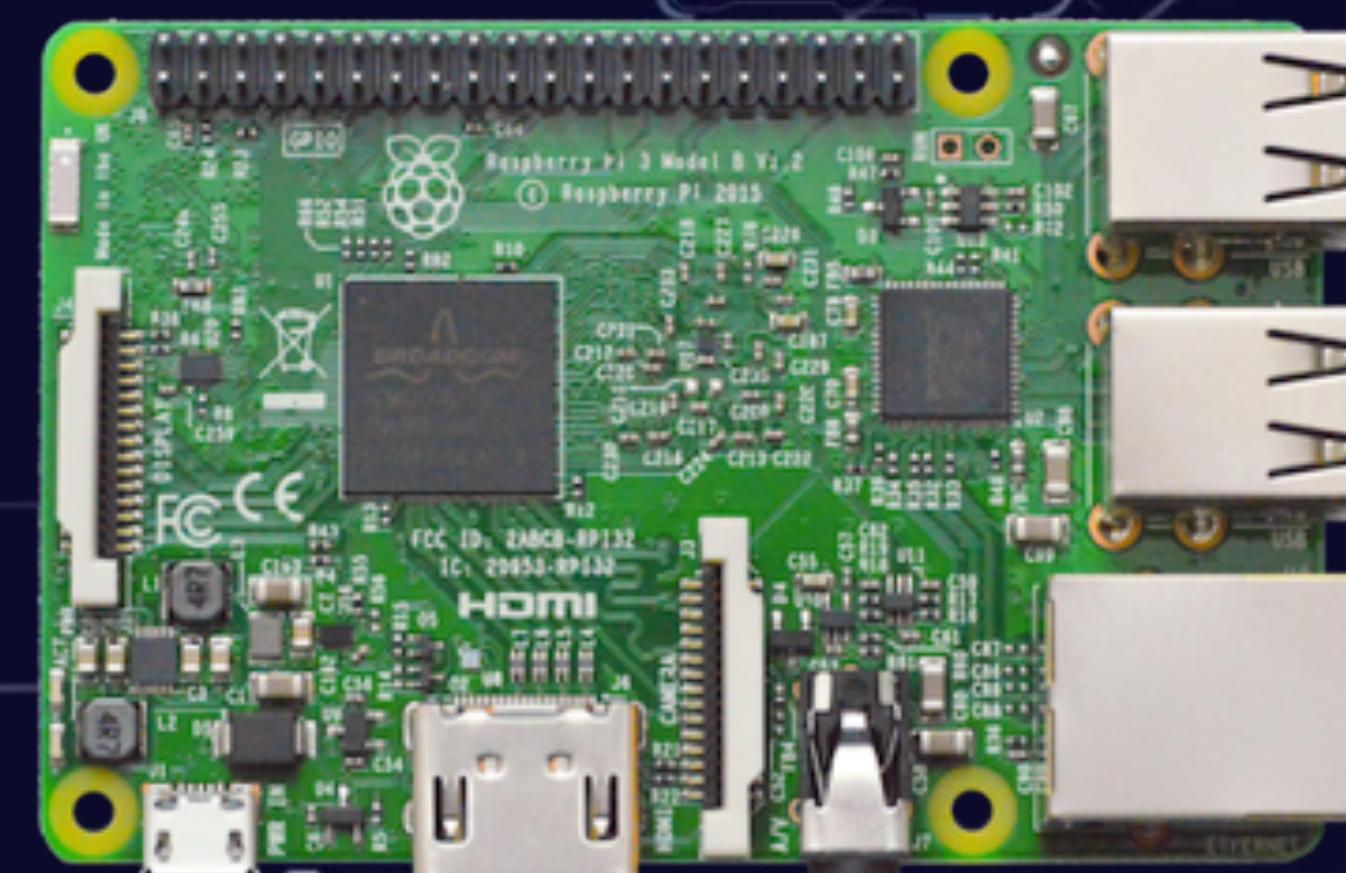
Step 1

- Download the Android Things Setup Utility.
 - <https://partner.android.com/things/console/#/tools>
- Using the setup utility:
 - Select the option to install Android Things and optionally set up Wi-Fi.
 - Insert the microSD card into the microSD slot on the underside of the Raspberry Pi.



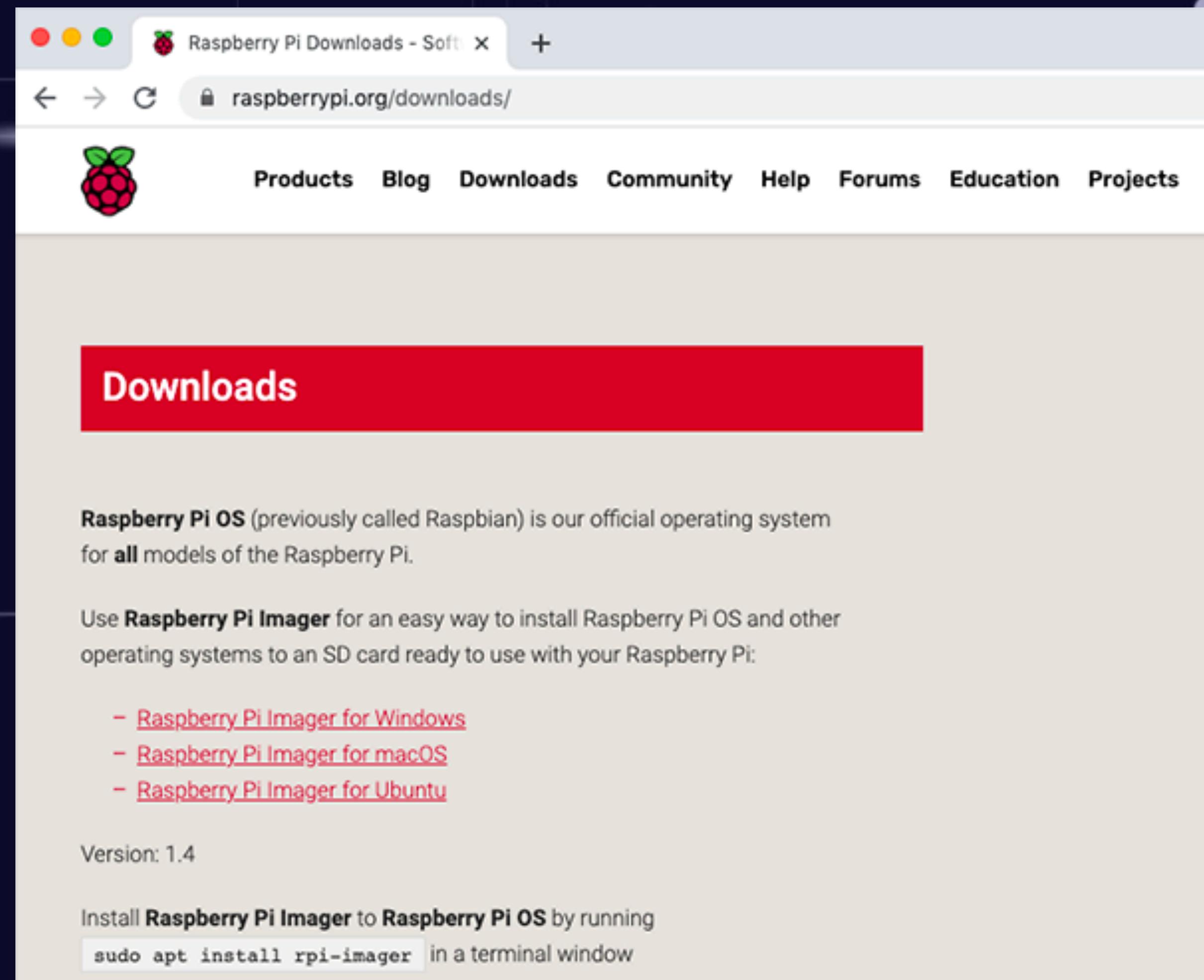
Step 2

- Connect a USB cable to J1 for power.
- Connect an Ethernet cable to your local network.
- (Optional) Connect an HDMI cable to an external display.
- Verify that Android is running on the device.
 - Connect to the IP address using the adb tool:
 - `adb connect <ip-address>`



connected to <ip-address>:5555

Raspberry Pi OS



The screenshot shows a web browser window displaying the official Raspberry Pi Downloads page at raspberrypi.org/downloads/. The page has a dark blue header with the Raspberry Pi logo and navigation links for Products, Blog, Downloads, Community, Help, Forums, Education, and Projects. A prominent red banner at the top says "Downloads". Below it, a section介绍 Raspberry Pi OS (previously Raspbian) as the official operating system for all Raspberry Pi models. It suggests using Raspberry Pi Imager for installation and provides download links for Windows, macOS, and Ubuntu. The page also notes Version 1.4 and instructions for installing Raspberry Pi Imager.

Raspberry Pi Downloads - Soft

raspberrypi.org/downloads/

Products Blog Downloads Community Help Forums Education Projects

Downloads

Raspberry Pi OS (previously called Raspbian) is our official operating system for **all** models of the Raspberry Pi.

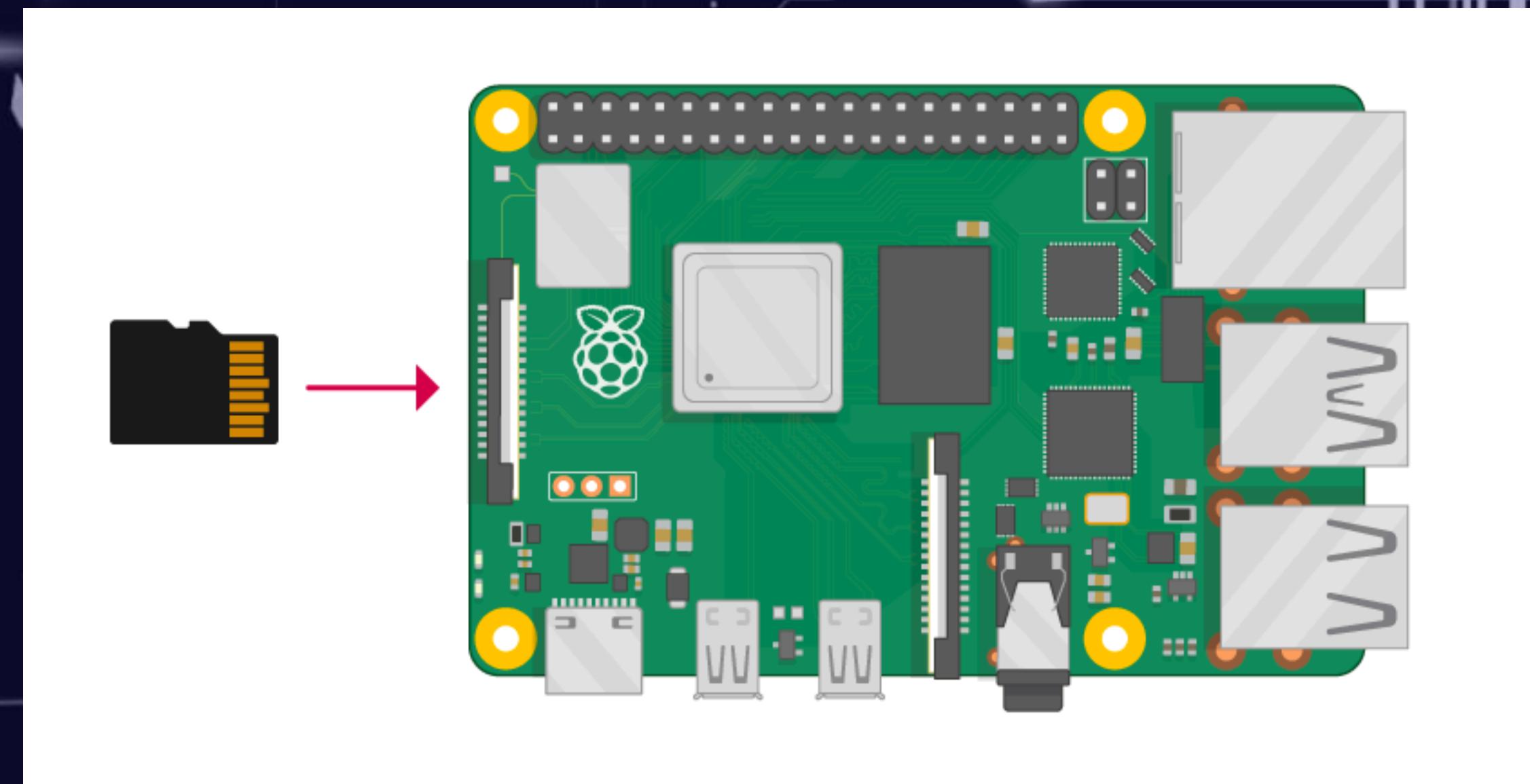
Use **Raspberry Pi Imager** for an easy way to install Raspberry Pi OS and other operating systems to an SD card ready to use with your Raspberry Pi:

- [Raspberry Pi Imager for Windows](#)
- [Raspberry Pi Imager for macOS](#)
- [Raspberry Pi Imager for Ubuntu](#)

Version: 1.4

Install **Raspberry Pi Imager** to **Raspberry Pi OS** by running
`sudo apt install rpi-imager` in a terminal window

Raspberry Pi OS



<https://projects.raspberrypi.org/en/projects/raspberry-pi-setting-up>

Lecture outcomes

- Understand how to establish the needed components.
- How to establish the hardware characteristics for the needed components.
- How to use the available pins for input and output.
- Setup the development platform.

