Programming with MicroBit

Overview on Hardware Features

Sensors and Actuators

Part II

Micro:Bit OLED Display

What is OLED?

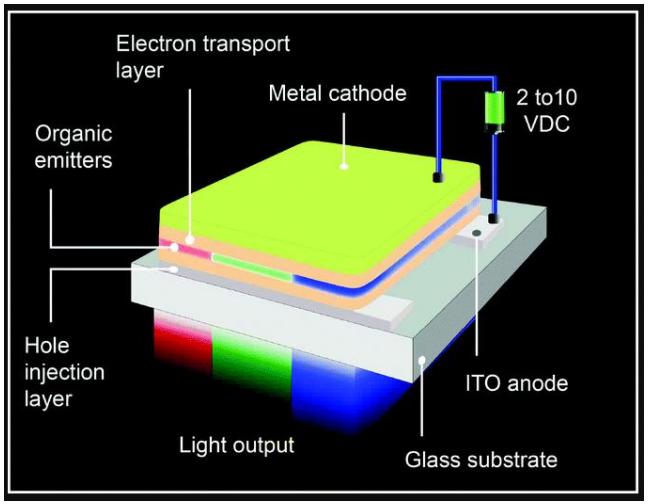
OLED is an abbreviation of Organic Light Emitting Diode

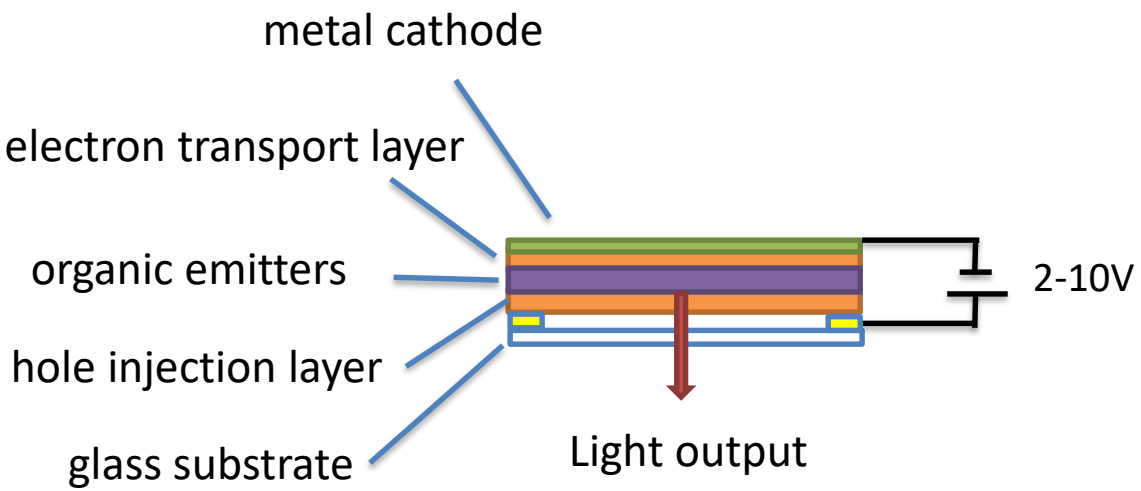
- A flat light emitting technology
- Use organic materials that emit light when electric current is applied

Introduction to OLED:
1) https://www.youtube.com/watch?time_continue=201&v=ZcTxI4QoGAo
2) https://www.oled-info.com/introduction

Basic Design of OLED

- Like a sandwich
- Place a series of organic thin films between two conductors.
- When electrical current is applied, a bright light is emitted.





DOI: 10.1039/C3CS60449G (Review Article) *Chem. Soc. Rev.*, 2014, **43**, 3259-3302

Advanced Applications

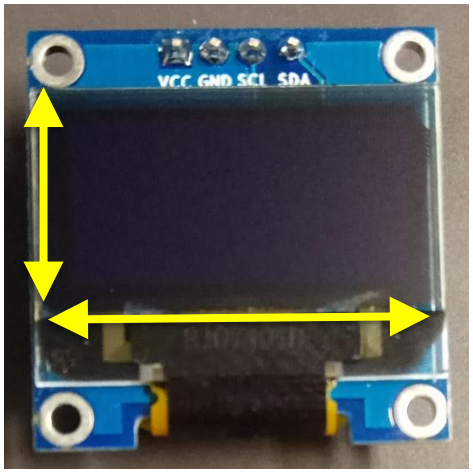
Flexible display

Transparent display

OLED Used in Project

- Passive OLED
- Key features
  - Dimension: 0.96 inch
  - Resolution: 128 x 64 (dots)
- Location specified by (x,y) co-ordinates
- Text and simple shape can be made by dots
- Control via I2C bus
- Optional display for mini-project

64 dots



128 dots

What is I^2C (I Square C)?

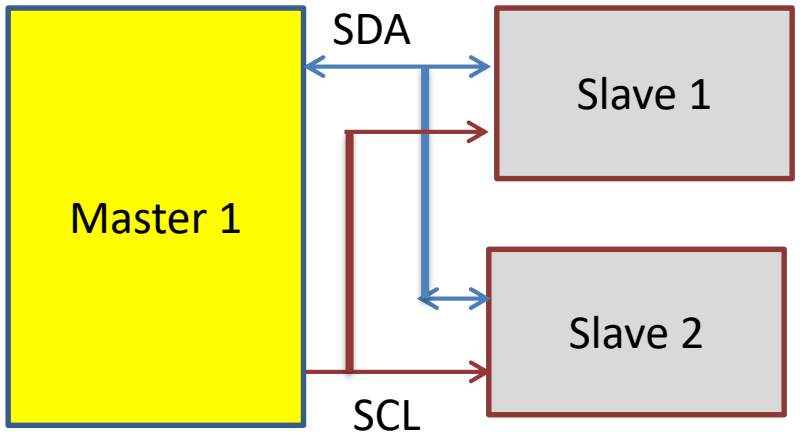
- I2C : Inter-Integrated Circuit
- A synchronous serial communication, multi-master, multi-slave, wired communications
- Only required two signal wires: SCL (Serial Clock) and SDA (Serial Data)
  - SCL is the clock signal, and SDA is the data signal.

Introduction of I2C : https://learn.sparkfun.com/tutorials/i2c

Master and Slave in I^2C

- The device that initiates a transaction on the I^2C bus is called the master. The master controls the clock signal. A device being addressed by the master is called a slave.
- Using just 2 lines (SDA and SCL), a master can connect to multiple slaves (up to 1008)
- We can also have multiple masters / multiple slaves.

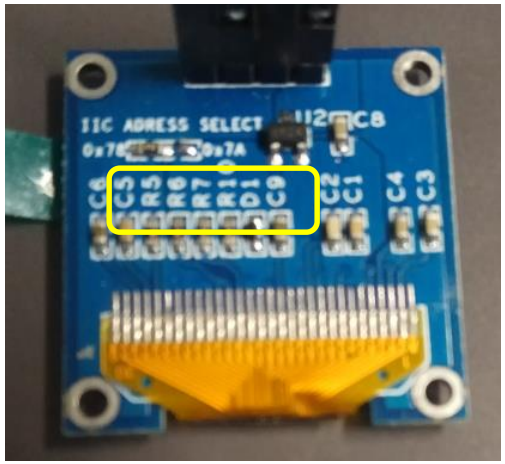
Example of 1 master and 2 slaves



Address of Slave Devices

- How to differentiate different devices if they connected to same master?
  - Each I^2C-compatible hardware slave device comes with a predefined device address. Eg. our OLED is 60_d
  - To allow multiple identical devices connecting to one I2C bus, user can alter some lower bits of the address.
  - For example, our OLED's address can be modified by re-soldering a resistor at the back.

    (Address will then be changed to 61_d)

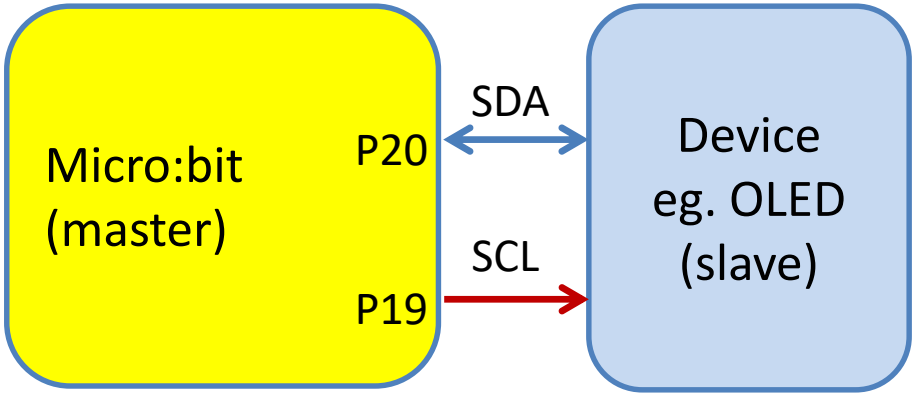


How I^2C Works?

- The master transmits the device address of the intended slave at the beginning of every transaction.
- Each slave is responsible for monitoring the bus, but only the slave with that address will respond.

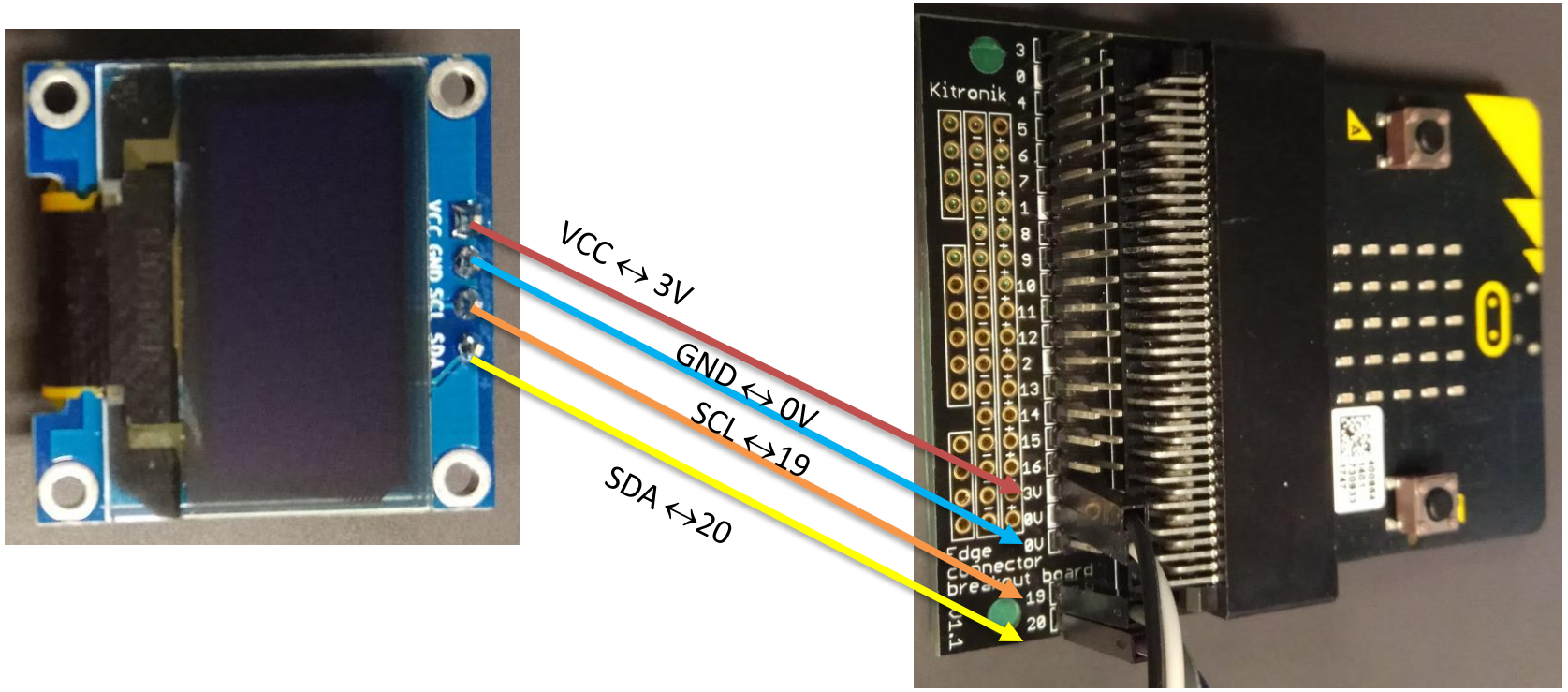
Connecting OLED to Micro:bit

- I^2C requires two wires
  - SCL: clock signal
  - SDA: data signal



- Micro:bit uses P19 and P20 for I2C
- Connections
  - 3V for supply
  - Pin 19 of micro:bit → Slave device's SCL
  - Pin 20 of micro:bit → Slave device's SDA
  - Common ground

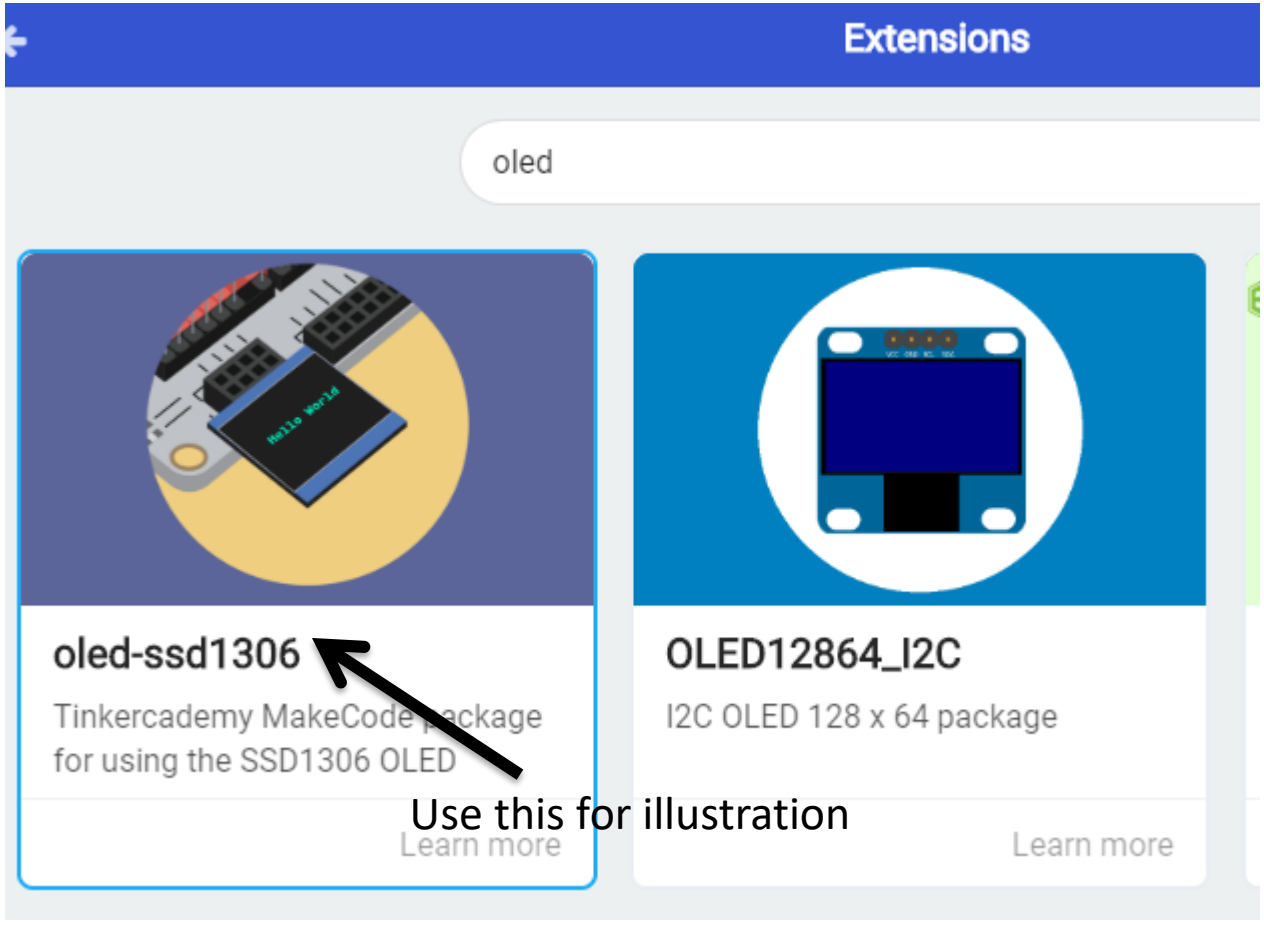
Connecting OLED to Micro:bit



You may need to solder the connector for pin 19 and Pin 20, if you can't find in your breakout board.

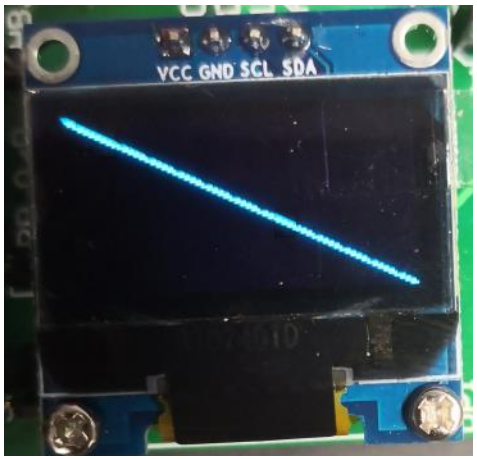
Programming of OLED

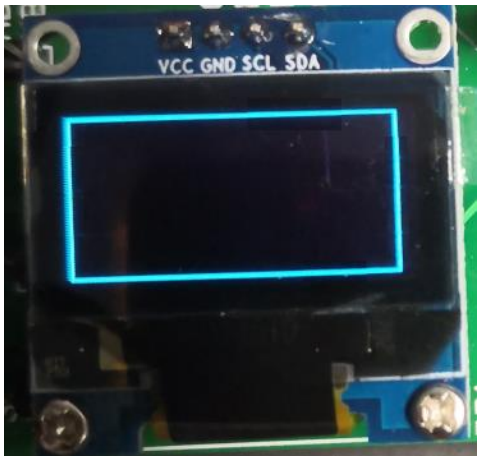
- You will need to add an extension to program the module. (search “OLED”)



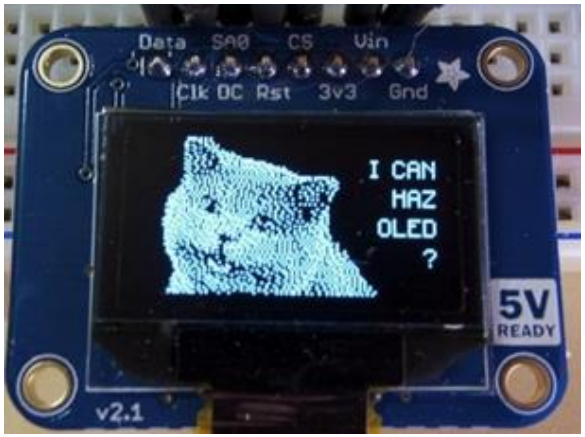
What can an OLED display?

Text





Geometric shapes

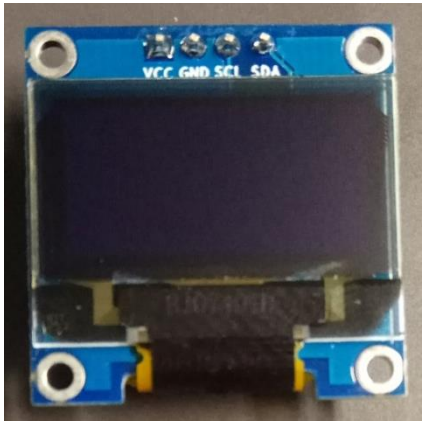


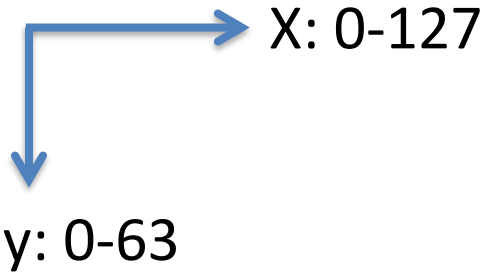
Graphics and Image

1) https://blog.adafruit.com/2019/12/13/a-simple-python-game-with-microbit-and-oled-display-oled-gaming-python-microbit-blogmywiki/
2) https://github.com/fizban99/microbit_ssd1306

Size of Display Area

- The size of display area is 64 x 128 (dots)





- Max: 8 rows and 21 characters on one screen; if more, it scrolls up

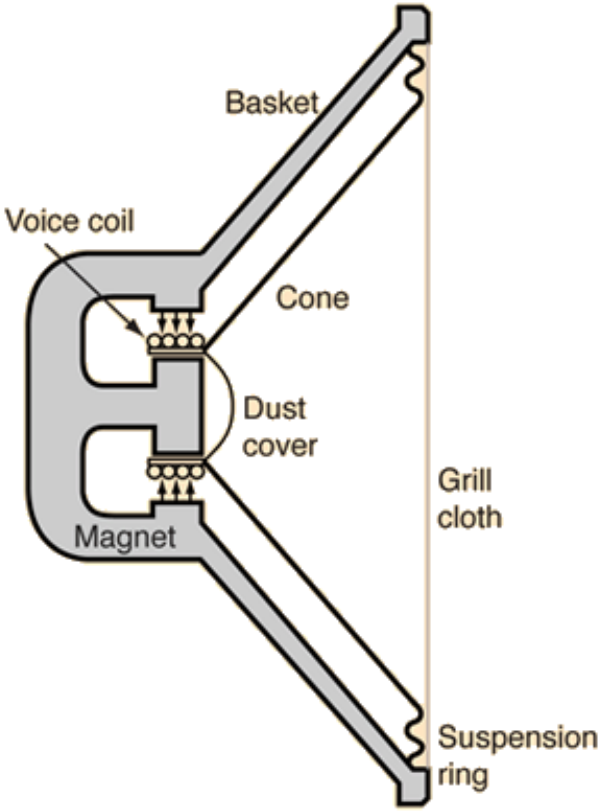
https://github.com/fizban99/microbit_ssd1306

OLED Display

Speakers

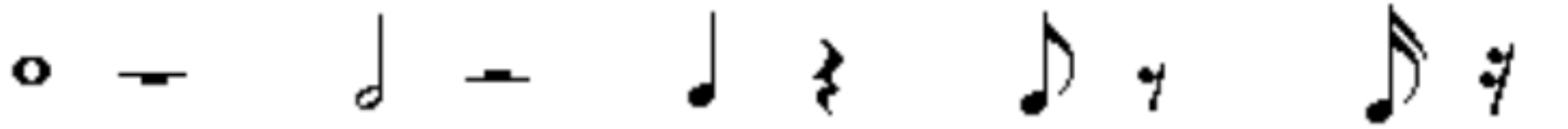
Loudspeaker

- Loudspeaker reproduces sound based on electromechanical process.
- When voltage is applied, the voice core generates magnetic field, and together with the fixed magnet, the cone will move accordingly (attraction and repulsion).
- As a result, it vibrates and generates sound according to the signal.



Music Notes

- Music Notes and Rest notes (no sound) for different durations



whole note	half note	quarter note	eighth note	sixteenth note
semibreve	minim	crotchet	quaver	semiquaver
4 beats	2 beats	1 beat	½ beat	¼ beat

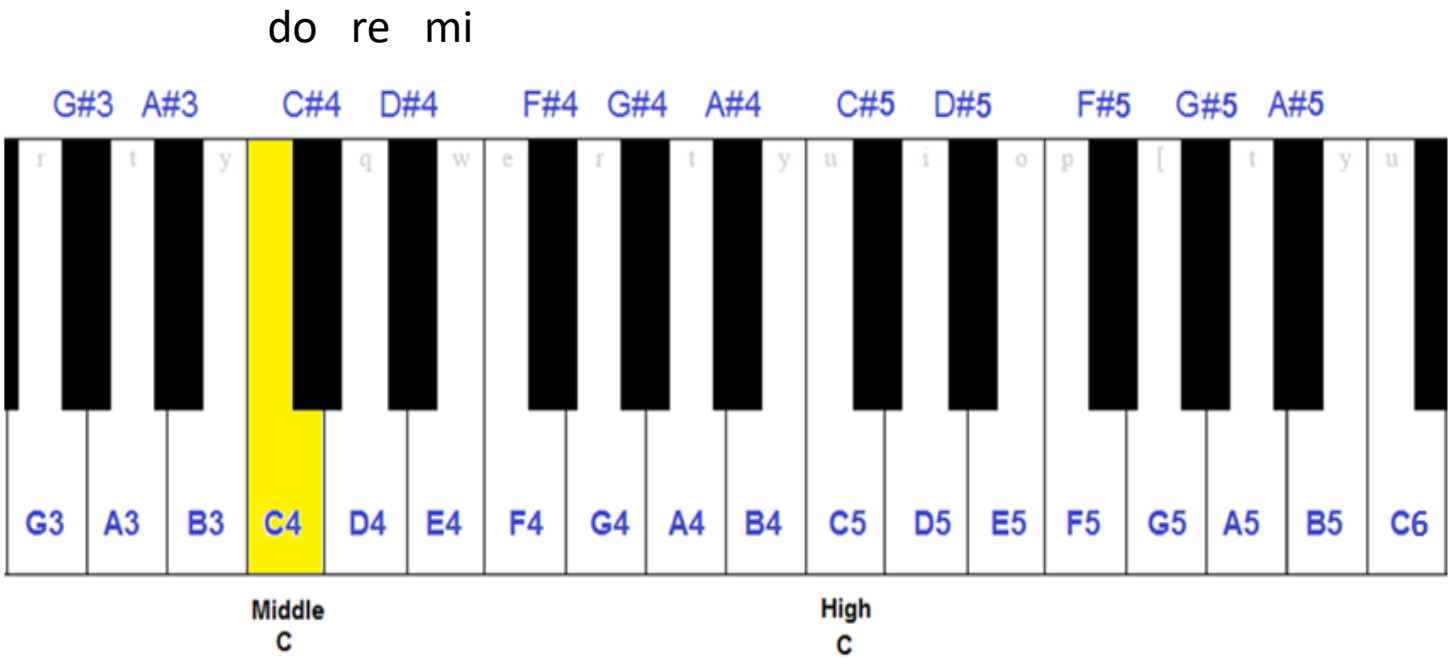
- The actual time duration of a note depends on the tempo of the song, which is in a unit of beats per minute (bpm).
- Eg. 1 beat in a tempo with 96bpm lasts for

60sec/96 = 0.625sec = 625 ms

Pitch of a sound

- Pitch of sound is characterized by frequency

- The keyboard of a piano:



Frequency and pitch

Pitch	G3	G#3	A3	A#3	B3	C4	C#4	D4
Freq(Hz)	196	208	220	233	247	262	277	294

Pitch	D#4	E4	F4	F#4	G4	G#4	A4	A#4
Freq(Hz)	311	330	349	370	392	415	440	466

Pitch	B4	C5	C#5	D5	D#5	E5	F5
Freq(Hz)	494	523	554	587	622	659	698

Pitch	F#5	G5	G#5	A5	A#5	B5	C6
Freq(Hz)	740	784	831	880	932	988	1047

Generating a music

- Specify the tempo (x Beats per minute)
- Generating the wave with specific frequency (the pitch f) and last for certain number of beats (y beats) time (the duration or the number of beats).
- Number of cycles to generate $N = 60fy/x$

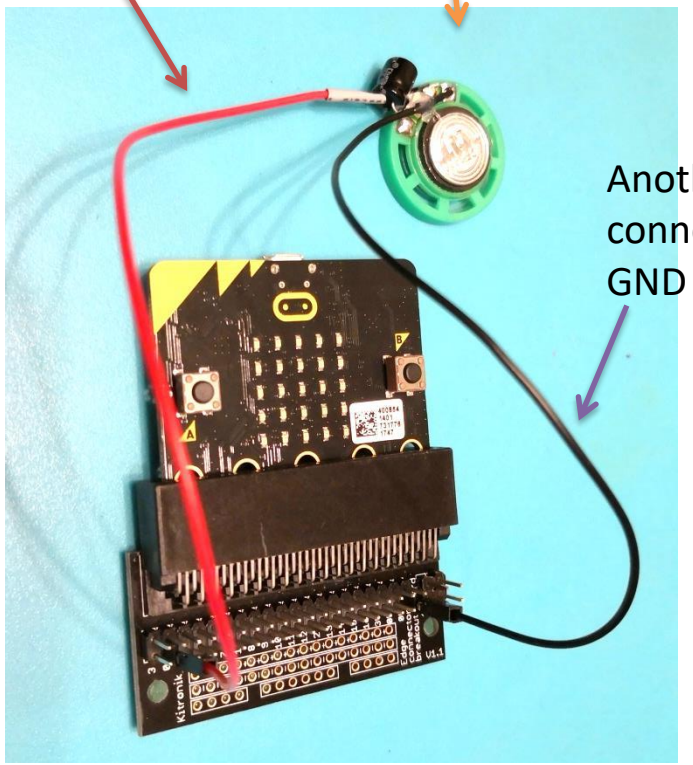
Example

```
from microbit import *
import music

tune = ["c4:4", "d4:4",
"e4:4", "c4:4", "c4:4",
"d4:4", "e4:4", "c4:4",
"e4:4", "f4:4", "g4:8",
"e4:4", "f4:4", "g4:8"]

music.play(tune)
```

For low power speaker, it can be connected directly to an I/O pin

The wire with a capacitor is + and it is connected to P0

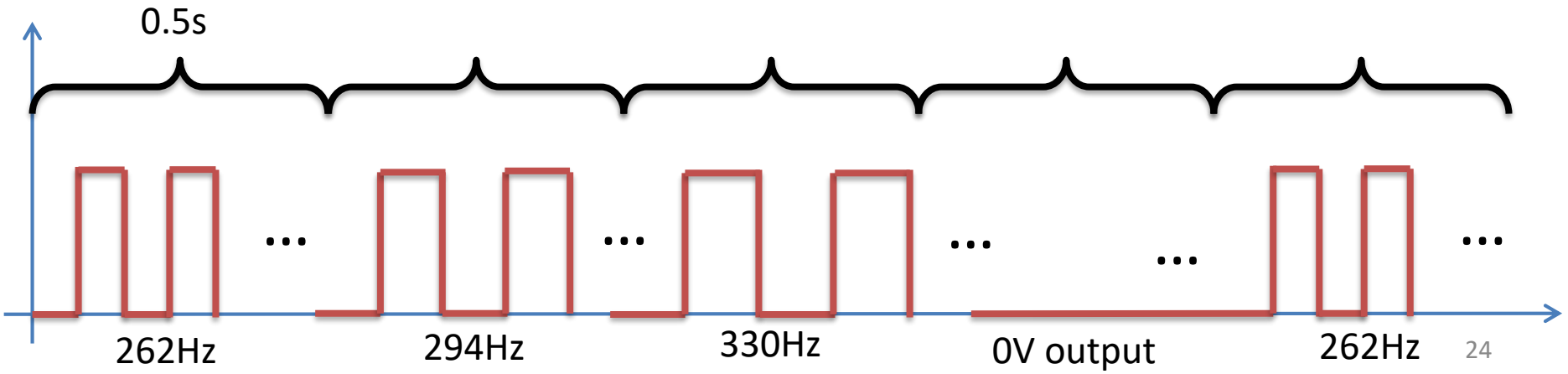
A capacitor (for ac coupling)



Another wire is connected to GND (0V).

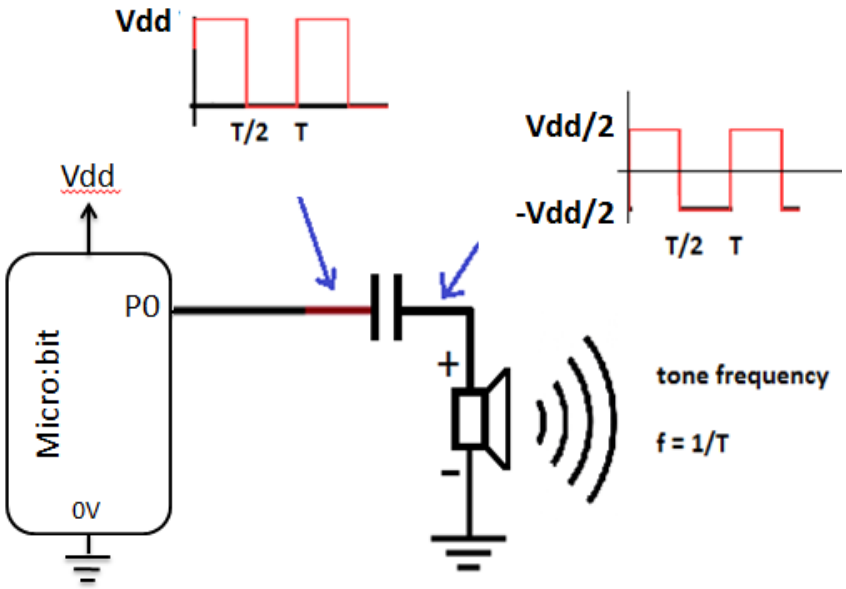
Timing

- 120bpm: 1 beat lasts 0.5sec
- Frequency
  - Middle C: 262Hz
  - Middle D: 294Hz
  - Middle E: 330Hz



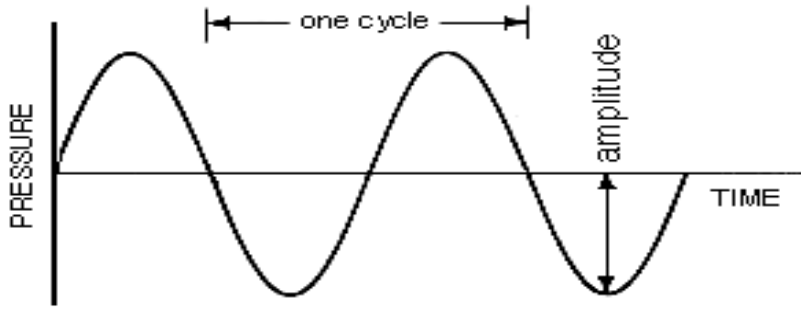
Remark 1: DC Problem

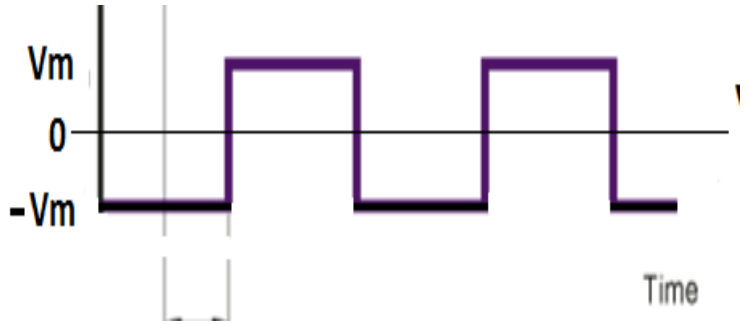
- Output of an I/O pin: 0V – LOW; Vdd (3V) – HIGH
- The generated square wave has a DC shift which may damage a loudspeaker
- Solution:
  - Add ac-coupling capacitor in series as shown
  - The capacitor will block the dc components, but let the ac components go through



Limitation

- Loudness depends on the amplitude of the wave V_m

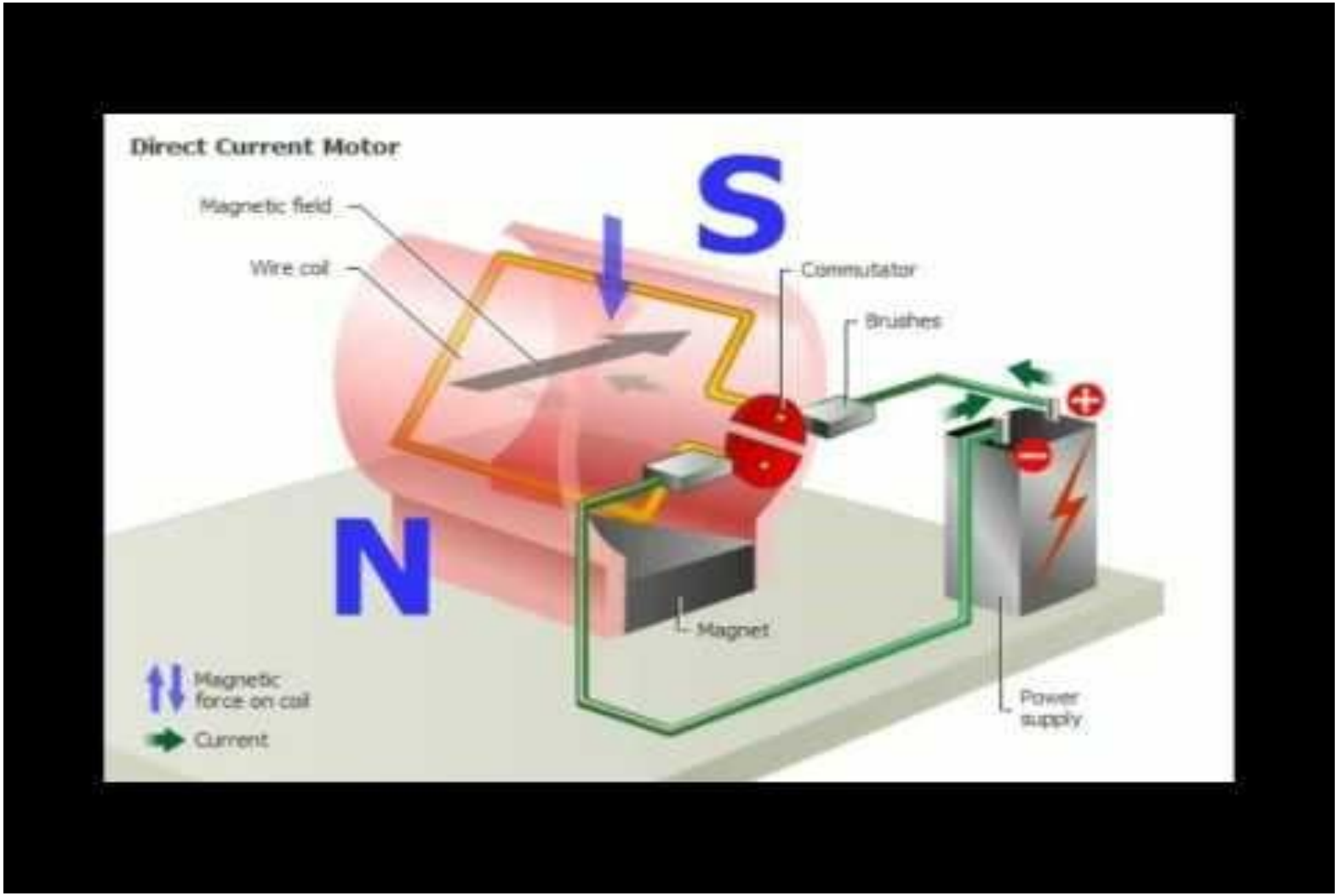




- Voltage level of I/O pin cannot be modified to different level (Note: even PWM only modify the duty cycle, not the peak of amplitude of the square wave)

Motor

Basic Principle



Various types of motors

Nowadays Motor’s Applications

Flight devices

Automation

Robotics

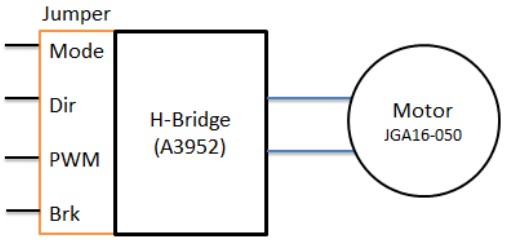
Smart Appliance

Security devices

Transportation

Driving Circuit

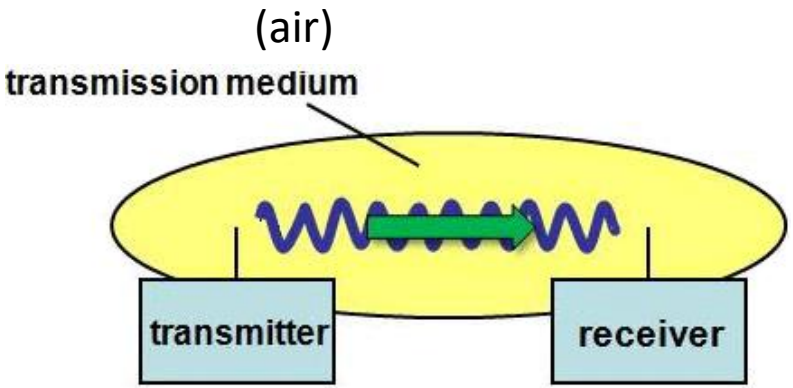
- Some motors need high voltage and high current
  - E.g. Motor (JGA16-050) needs 1A current, too large for micro:bit;
  - It also needs 6V but micro:bit can only output 3V (Vdd).
- Solution: Driving circuit is needed
  - e.g. to drive JGA16-050, a H-Bridge Motor Driver is needed



Wired and Wireless Communications

WIRED and WIRELESS

- Communications between units via a medium
- Wired Communication: Transfer of information between two or more points that are connected by an electrical conductor (wire)
- Wireless Communication: not via any physical wire



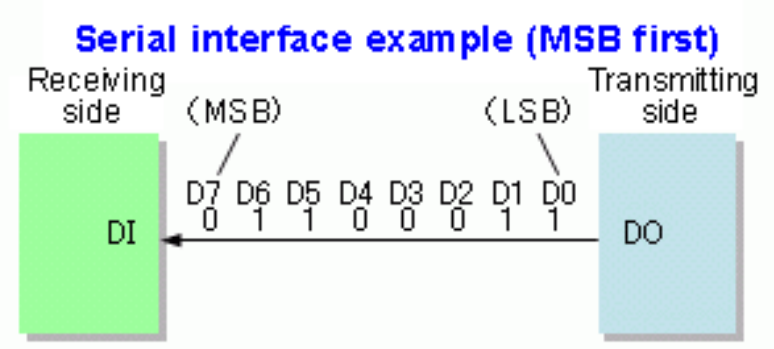
Sending message

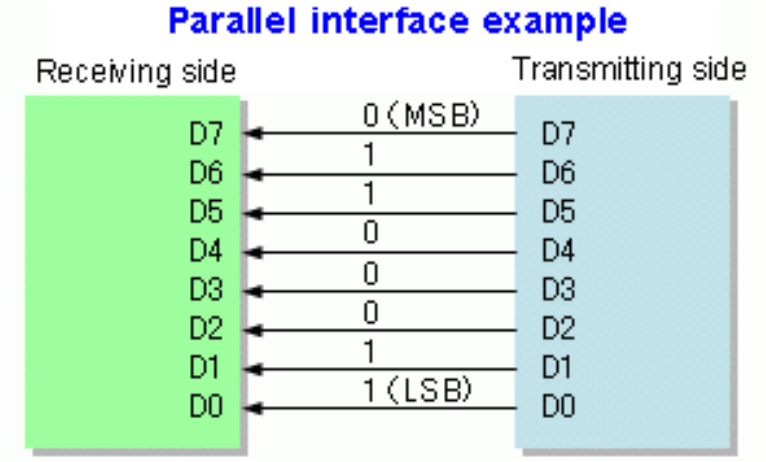
Receiving message

WIRED COMMUNICATION

Serial and Parallel Communication

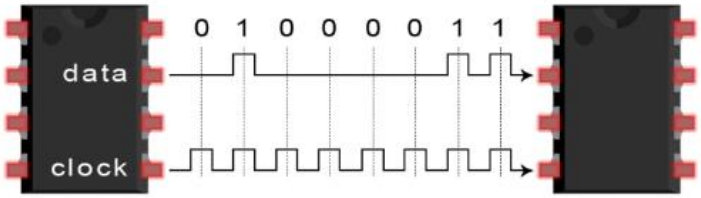
- Data in bytes (1 byte = 8 bits)
- Serial communication
  - Disadv: Bits are sent one by one (eg. 8 bits need 8 cycles), hence slow
  - Adv: Require very few pins (wires) for communications
- Parallel communication
  - Adv: Several bits (eg. a word of 8 bits) are sent in one cycle, hence fast
  - Disadv: Require many pins/wires



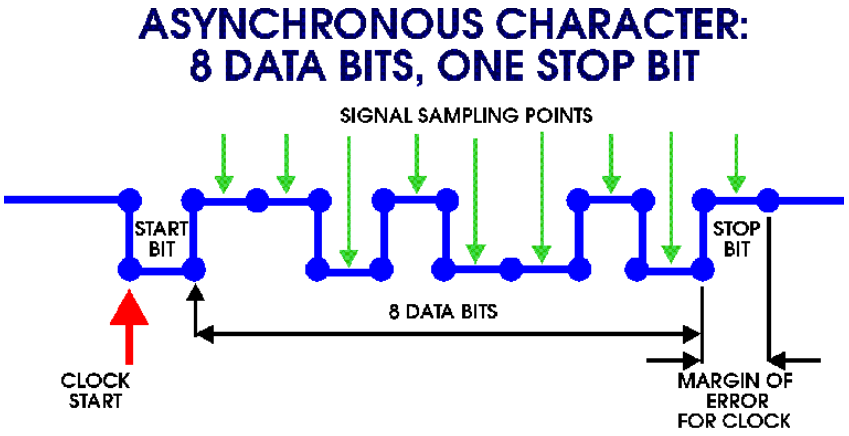


Synchronous and Asynchronous

- Serial port interface
  - Synchronous:
    - A clock to synchronize between transmitter and receiver so that the receiver can receive the data correctly.
  - Asynchronous:
    - No synchronous clock
    - Start and stop bits in the data packet being transferred to define the beginning and end of the data packet. With such, the receiver can know when to start reading the bits.

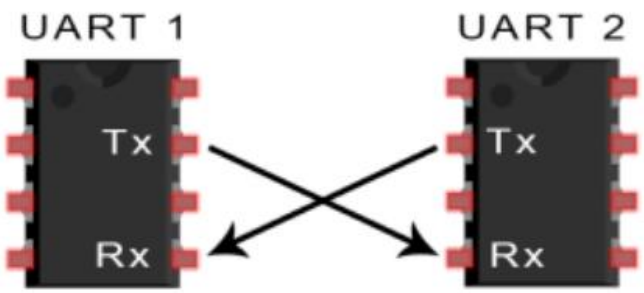


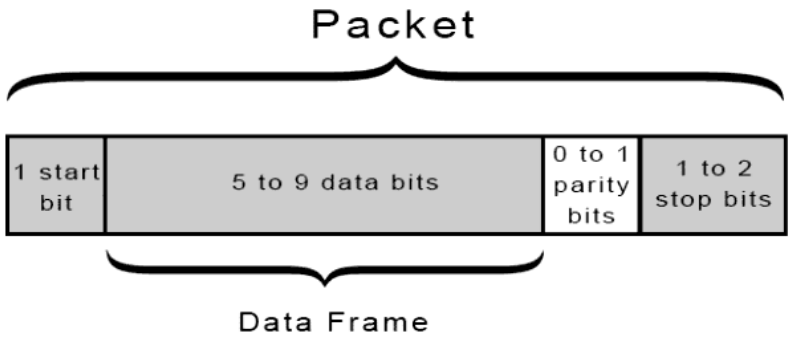
Synchronous



UART

- UART: Universal Asynchronous Receiver/Transmitter
- A kind of serial communications (Very commonly used)
- General flow:
  - 2 UARTs: one acts as transmitter and one is receiver
  - Transmitter: Data (in bytes) is firstly converted into a serial form (bits), then transmitted one by one at Tx pin.
  - Receiver: After received the serial data one-by-one from Rx pin, the receiver converts the serial data back as byte.





http://www.circuitbasics.com/basics-uart-communication/

Serial Communication

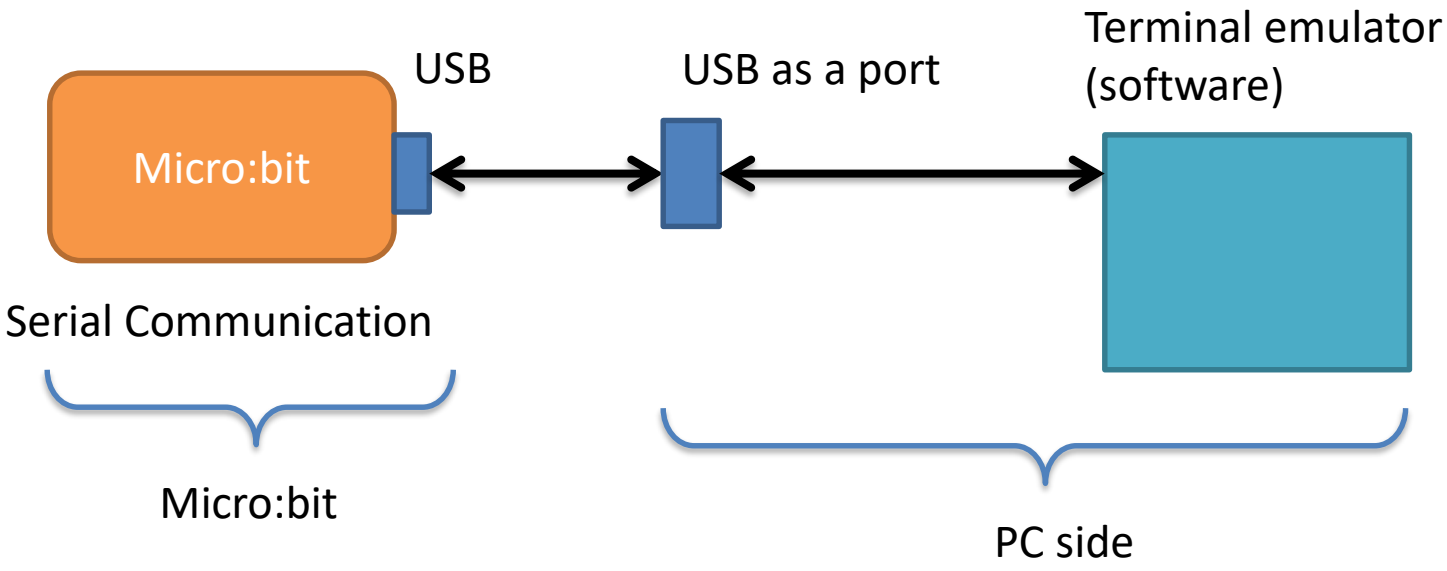
- Useful for communicating with PC
- Connecting micro:bit to your PC using an USB cable, communications can be established (Note: not only doing programming).
- Based on Serial Communications (wired)

What is Needed?

- A terminal emulator is needed at PC side to communicate via the port
  - Putty, See https://www.putty.org/
  - CoolTerm, download from http://freeware.the-meiers.org/ (we demonstrate this one).

Data Flow Diagram

- Data flow:
  - Terminal -> USB port -> Micro:bit or
    Micro:bit -> USB port -> Terminal
- What are needed?
  - Program the micro:bit
  - Configure terminal emulator and the USB port



WIRELESS COMMUNICATION
(BUILT-IN: RADIO COMMUNICATION)

Example of Wireless Communication

Wireless Comm. Unit

Fan with Bluetooth

Communicate over air

Wireless Comm. Unit

Mobile Phone

Devices with Wireless Communication Capability
(Devices can be different, but the wireless communication specifications should be the same.