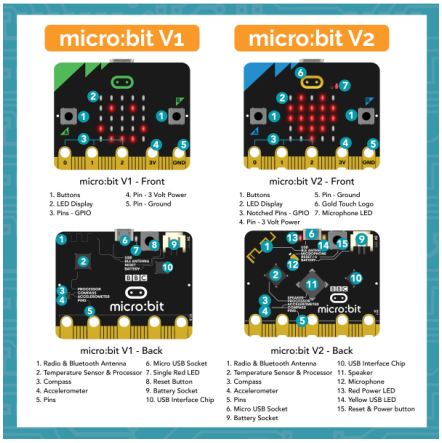
Micro:bit

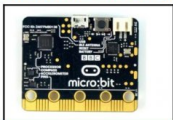
(page créée le 20 mai 2022, en cours de rédaction)

Il existe deux versions principales de micro:bit.

Caractéristiques principales



micro:bit V1.5 vs V2



Features/Specs	micro:bit v1.5	micro:bit v2
Release Date	NA	13th Oct 2020
MCU or Processor	Nordic Semiconductor nRF51822	Nordic Semiconductor **nRF52833**
MCU Core Architecture	ARM Cortex-M0 32-bit	ARM **Cortex-M4** 32-bit **(FPU)**
MCU Flash Size	256KB	**512KB**
RAM Size	16KB	**128KB**
MCU Clock	16MHz	**64MHz**
USB Interface Processor	NXP KL26Z, 16KB RAM	NXP **KL27Z, 32KB RAM**
Microphone, MIC	None	**MEMS Microphone, LED indicator**
Speaker	None	**Onboard Piezo Buzzer**
Touch Sensitive Logo	None	**Touch Sensitive Logo Pad**
Wireless	2.4GHz micro:bit radio/BLE Bluetooth 4.0	2.4GHz micro:bit radio/BLE **Bluetooth 5.1**
Power	5V via USB, 3V via edge connector or battery port	5V via USB, 3V via edge connector or battery port
Power indicator LED	NA	**Onboard Power Indicator LED**
Power Off Button	NA	**Onboard Power Button (Push and Hold*)**
Current for External	3V, 90mA	3V, **200mA**
Motion Sensor	ST LSM303	ST LSM303
Edge Connector	25-pin, 3 dedicated GPIO, PWM, I2C, SPI, Power, and etc	25-pin, **4 dedicated GPIO**, PWM, I2C, SPI, Power, and etc
Ring Connector	3 (GPIO) + 2 (Power) ring connectors	3 (GPIO) + 2 (Power) ring connectors, **notched edge**
I2C	Shared I2C Bus	**Dedicated I2C Bus** for peripherals
Software/IDE	C++, makecode, Python, Scratch	C++, makecode, Python, Scratch
Size	5cm (w) x 4cm (h)	5cm (w) x 4cm (h)

Brochage / Pinout

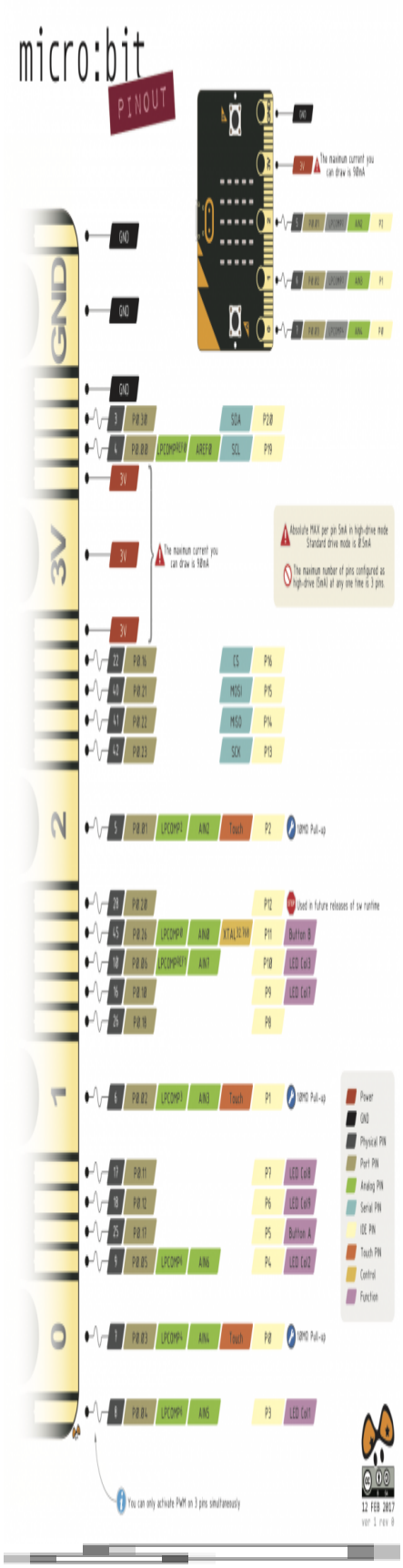


Schéma par Pighixx, CC BY-SA

Programmer la carte

En ligne, par bloc, en javascript ou micropython, avec l'éditeur makecode

- https://makecode.microbit.org/?lang=fr

En ligne / en local, par bloc avec Scratch

- https://scratch.mit.edu/microbit

En local, en code python, avec l'éditeur Mu

En local, en code C, avec l'IDE arduino

- https://learn.adafruit.com/use-micro-bit-with-arduino?view=all

Utilisation

Connexion série sur linux

A minima, on peut utiliser screen pour recevoir les données

```
ls /dev/ttyACM*               # chercher le port
screen /dev/ttyACM0 115200
# on arrête screen avec CTRL-A, puis K (pour Kill)...
# sur l'utilisation de screen, voir https://www.tecmint.com/screen-command-examples-to-manage-linux-terminals/
```

Des exemples pour une réception dans processing ici : https://github.com/emoc/microbit_utile

Extensions

micro:bit + shield grove : https://wiki.seeedstudio.com/Grove_Inventor_Kit_for_microbit/

micro:bit avec Scratch :

Ressources