Raspberry Pi

Avancé: épisode 2, ou comment devenir maître Jedi

Version du 15 juin 2019 - www.makilab.org - CC-BY-SA

MakiLab
WE ARE ALL MAKERS !

Source image: LEGO [www.lego.com]

Le bus I²C

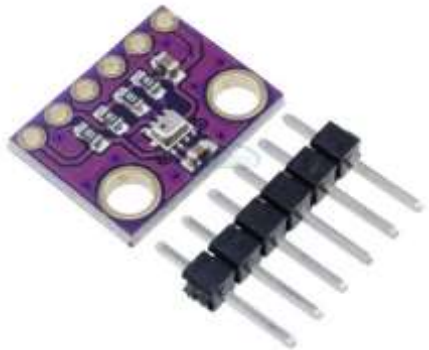
Source [https://www.raspberrypi.org/documentation/usage/gpio/README.md]

Le bus I²C

C’est un “bus” de donnée, qui permet de lire ou d’écrire des informations sur les composants connectés au bus.

Par exemple:

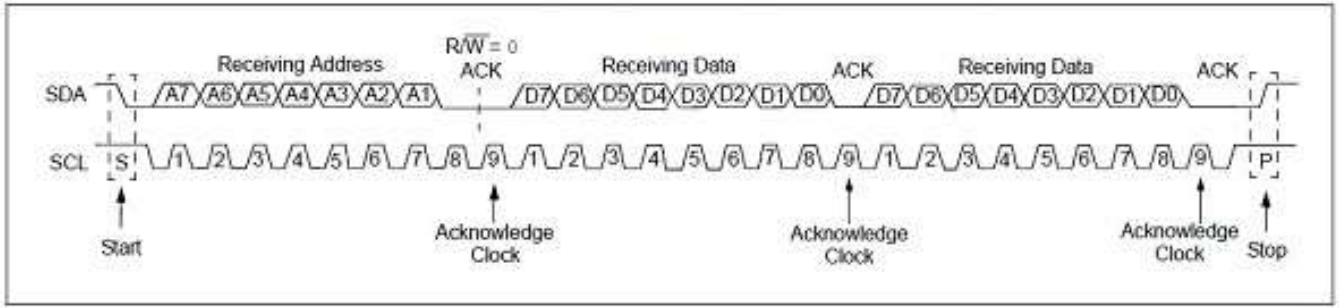
et



Source [https://www.raspberrypi.org/documentation/usage/gpio/README.md]

I²C

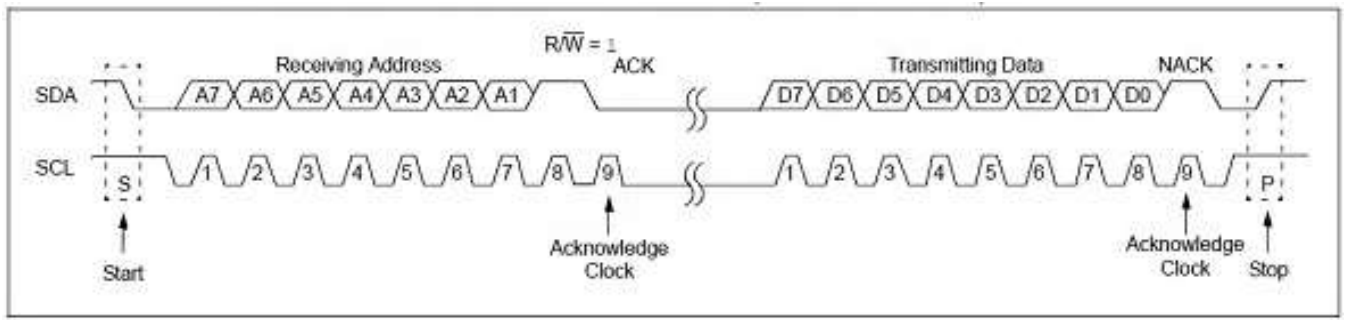
Ecriture



3 connections minimums:

- GND: masse
- SDL (SDA): Signal de donnée
- SLCK (SCK) (SCL): Signal d'horloge (clock)

Lecture



Source image: Developpez.com [https://f-leb.developpez.com/tutoriels/arduino/bus-i2c/]
How to Mechatronics [https://howtomechatronics.com/tutorials/arduino/how-i2c-communication-works-and-how-to-use-it-with-arduino/]

Le bus I²C

Sur le Raspberrry pi, on va devoir l'activer via `sudo raspi-config`

→ Interfacing options → I2C

Pour l'écran OLED, on va aussi devoir installer la bibliothèque python `Luma.`

```
$> sudo apt-get install python3-dev python3-pip libfreetype6-dev libjpeg-dev
build-essential

$> sudo -H pip3 install --upgrade luma.oled
```

Pour le capteur de température/pression BMP280, la bibliothèque `bmp280`

```
$> sudo -H pip3 install --upgrade bmp280
```


Source [https://www.raspberrypi.org/documentation/usage/gpio/README.md]

Branchement

L’écran OLED: affichage de texte

```
$> cd ~/Scripts
$> nano oled.py
```

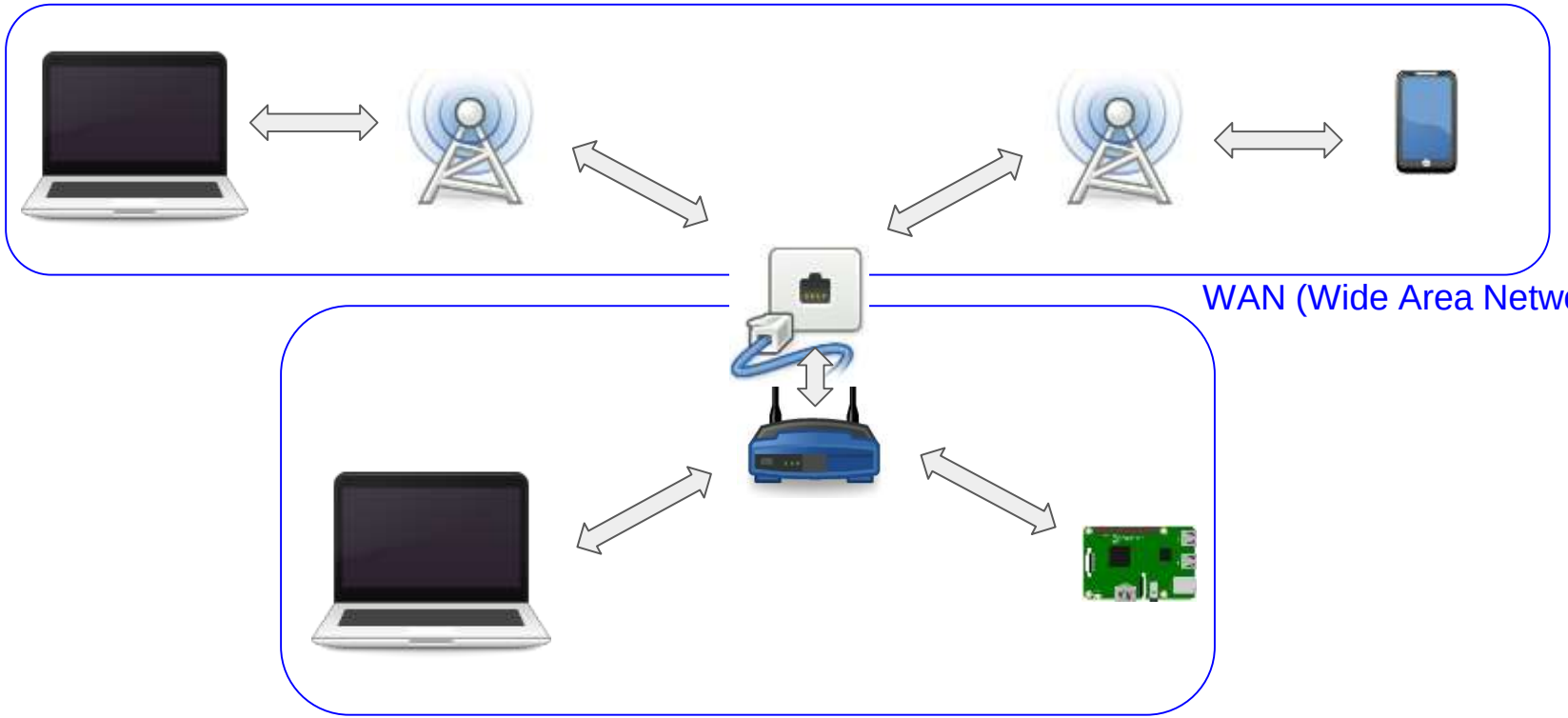
```
from luma.core.interface.serial import i2c
from luma.core.render import canvas
from luma.oled.device import ssd1306
from signal import pause
serial = i2c(port=1, address=0x3C)
device = ssd1306(serial)

with canvas(device) as draw:
    draw.rectangle(device.bounding_box, outline="white", fill="black")
    draw.text((10, 40), "Bonjour à tous!", fill="white")

pause()
```

```
$> python3 oled.py
```


Source [https://www.raspberrypi.org/documentation/usage/gpio/README.md]

Se connecter à distance… depuis n’importe où

Jusqu’à présent, on est resté sur le réseau local

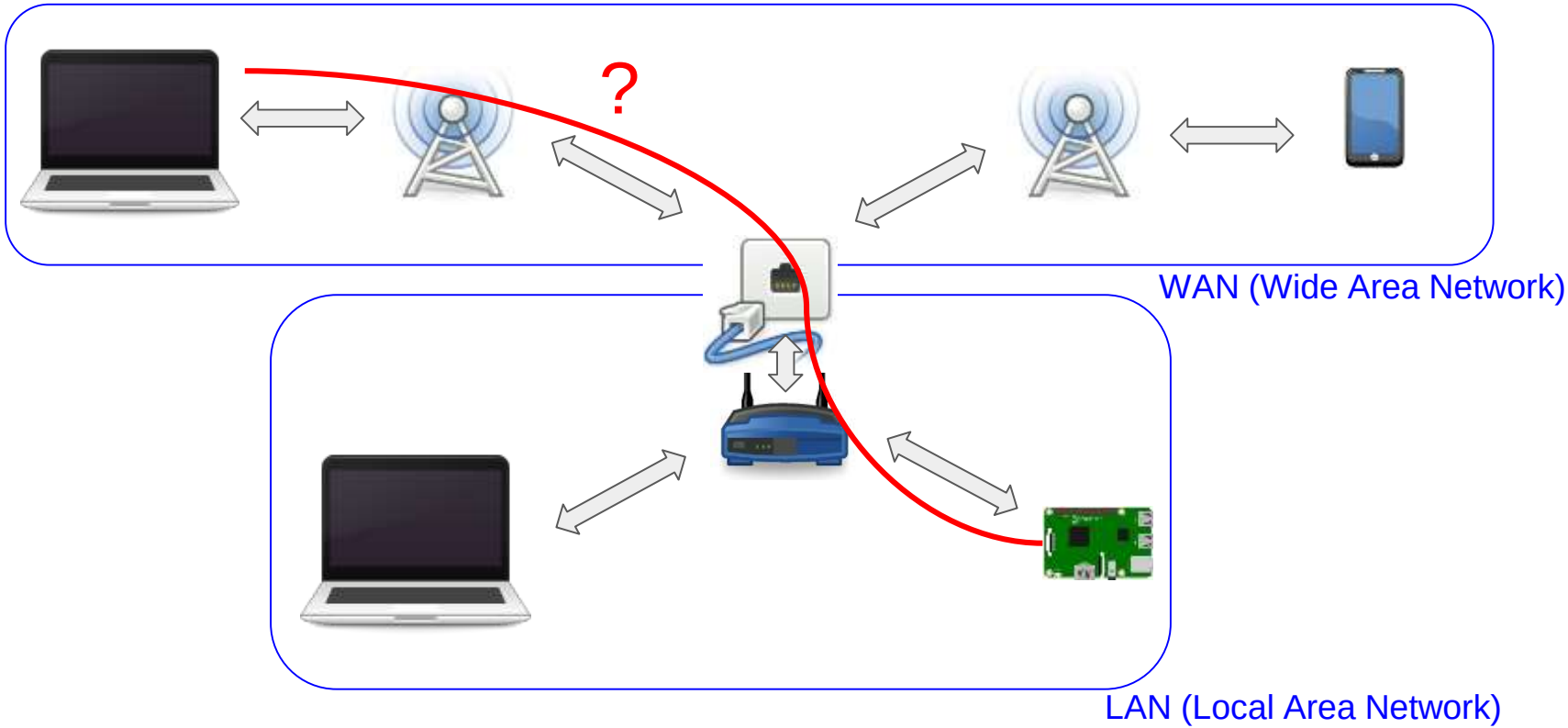


WAN (Wide Area Network)

LAN (Local Area Network)

Source image [By Rickterto - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=44607502]

Configurer l’accès extérieur?



Source image [By Rickterto - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=44607502]

Accès extérieur: 2 solutions

1

configurer votre routeur / “box” internet pour faire le routage correctement.

2

utiliser un service en ligne.

3

configurer un VPN sur votre réseau local et y accéder via la connection VPN.

Accès extérieur, solution 1

Redirection de port et DNS dynamique

L’idée: configurer votre routeur / “box” internet pour rediriger les ports

Dépend de votre modèle, mais en général cela s’appelle “port forwarding” ou “redirection de port”.

Chaque service sur votre Rpi utilise un ou plusieurs port(s) différent(s):

- **SSH**: port 22 par défaut
- **VNC**: ports de 5900 à 5906

Accès extérieur, solution 1

Le **principe**: le périphérique sur le WAN (par exemple un laptop) se connecte à votre routeur sur un port (par exemple 22), et le routeur le redirige vers votre RPi.

Autre **problème**: quelle est l'adresse IP de votre routeur côté WAN?
(Indice: elle n'est pas fixe, et peut changer souvent en fonction de votre fournisseur d'accès).

La **solution**: un service de DNS dynamique.

Votre box soit supporter la redirection de port et le DNS dynamique.
(Indice: la Bbox proximus devrait pouvoir le faire).

Pour ceux qui voudraient se lancer: https://raspbian-france.fr/mettre-en-ligne-serveur-web-raspbian-dydns-port-forwarding/

Accès extérieur, solution 2

Utiliser un service en ligne

L’idée: un service en ligne s’occupe de tout, via un logiciel spécifique installé sur votre RPi.

L’avantage: la simplicité

Le désavantage: vous n’avez pas la maîtrise, et ça peut devenir payant

Source image [https://www.raspberrypi.org/documentation/remote-access/access-over-Internet/README.md]

Accès extérieur, solution 2

Utiliser un service en ligne

Pour ceux qui voudraient se lancer: www.remote.it

Dès que les services sont “pulics” sur internet, il ne faudra pas longtemps pour qu’on essaie de pirater le RPi (botnets, …). Ne jamais le faire avec un Raspberry pi où d’utilisateur et le mot de passe par défaut est actif.
Plus d’infos / d’idée pour sécuriser (en anglais)
https://www.raspberrypi.org/documentation/configuration/security.md

Source image [https://www.raspberrypi.org/documentation/remote-access/access-over-Internet/README.md]

Sans écran ni clavier… Établissons une connection

SSH = Secure SHell

Dans un terminal (ou via “mRemoteNG” sous windows)

- `ssh pi@monnomdhote.local`

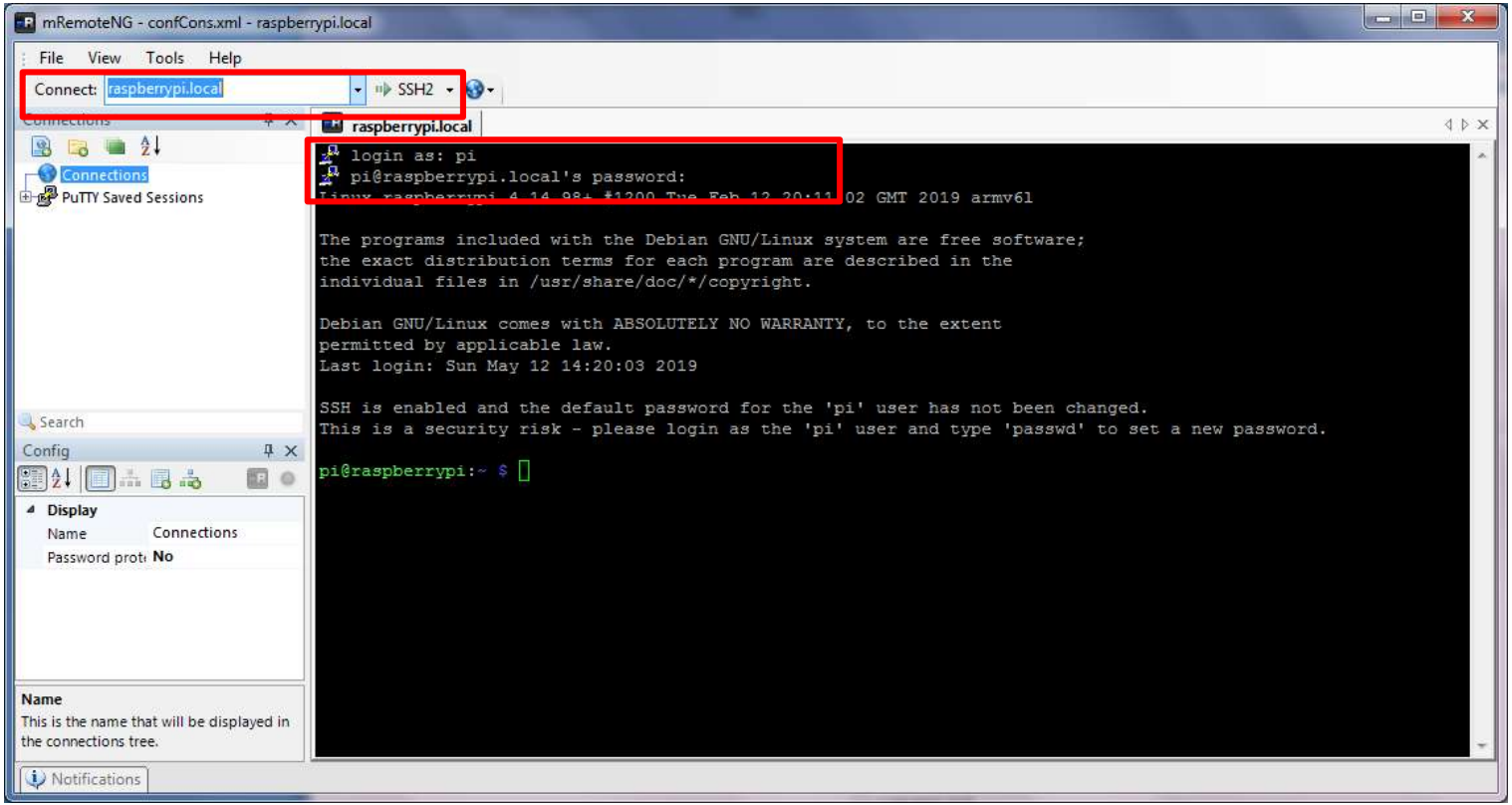
Par défaut le nom d’hote est *raspberrypi*

Par défaut l’utilisateur est *pi* et
le mot de passe *raspberry*

Note: les adresses “.local” ne fonctionnent sous Windows que si les Services d’Impression Bonjour (Apple) sont installés.

Source [https://raspbian-france.fr/controlez-raspberry-pi-ssh-ordinateur/]

SSH avec mRemoteNG



Makilab
WE ARE ALL MAKERS !

Et pour l'arrêter?

Pas de bouton OFF…

- En ligne de commande (si on est connecté au RPi):

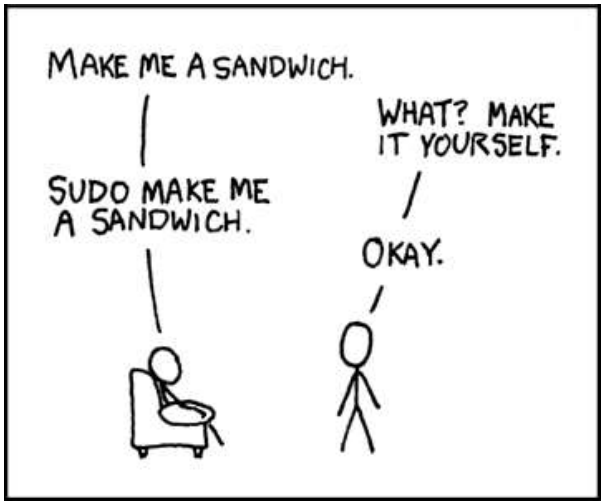
  `sudo poweroff`
- Si pas de connection au RPi: pas d'autre solution que le débrancher…

En débranchant "sauvagement" l'alimentation d'un RPi, on risque de corrompre la carte SD. Au minimum, éviter de le faire lorsque la LED verte (indiquée "act") clignote.

- Pour le redémarrer:

  `sudo reboot`

C’est quoi ce “sudo”?



Il y a deux types d’utilisateurs sur un système Linux:

- le super-utilisateur (*root*)
- tous les autres.

La commande *sudo* permet d'exécuter une autre commande en tant que super-utilisateur

L’utilisateur “root” peut littéralement tout faire sur le système, donc aussi le casser, le corrompre, … Réfléchissez-y à deux fois avant de taper une commande avec “sudo”, surtout si elle est copié/collée d’internet...

Source image [https://www.xkcd.com/149/]

OK, quelques règles de bonne pratique

Modifier la configuration standard (le plus simple: via *`sudo raspi-config`* en ligne de commande):

- Changer le **mot de passe** par défaut de l'utilisateur par défaut
  - → Change user password
- Changer le **nom d'hôte** si ce n'est pas déjà fait
  - → Network options → Hostname
- Changer la **langue**, le **fuseau horaire**
  - → Localisation options → Change Locale → ajouter "fr_BE.UTF-8" mais garder "en_US.UTF-8"
  - → Localisation options → Change Timezone → Europe/Brussels

Se connecter à distance en mode graphique

Il va falloir installer et configurer le nécessaire pour utiliser **VNC** (Vitual Network Computing):

Côté “serveur” (le RPi): via `sudo raspi-config`:

- → Interfacing options → VNC → Yes
- → Advanced option → Resolution → 1280x720 (ou autre)

Côté “client” (votre ordinateur):

- Un client VNC: realVNC viewer

1. Seule les versions *“Raspbian Stretch with desktop”* démarrent en mode graphique par défaut.
2. La version du protocole VNC du serveur utilisé par défaut sur RPi (RealVNC Server) n’est en pratique compatible qu’avec le client RealVNC Client :-(

Source [https://raspbian-france.fr/vnc-raspberry-pi/]

Se connecter à distance au système de fichier

Il va falloir installer et configurer le nécessaire pour utiliser **SFTP** (**S**SH **F**ile **T**ransfer **P**rotocol):

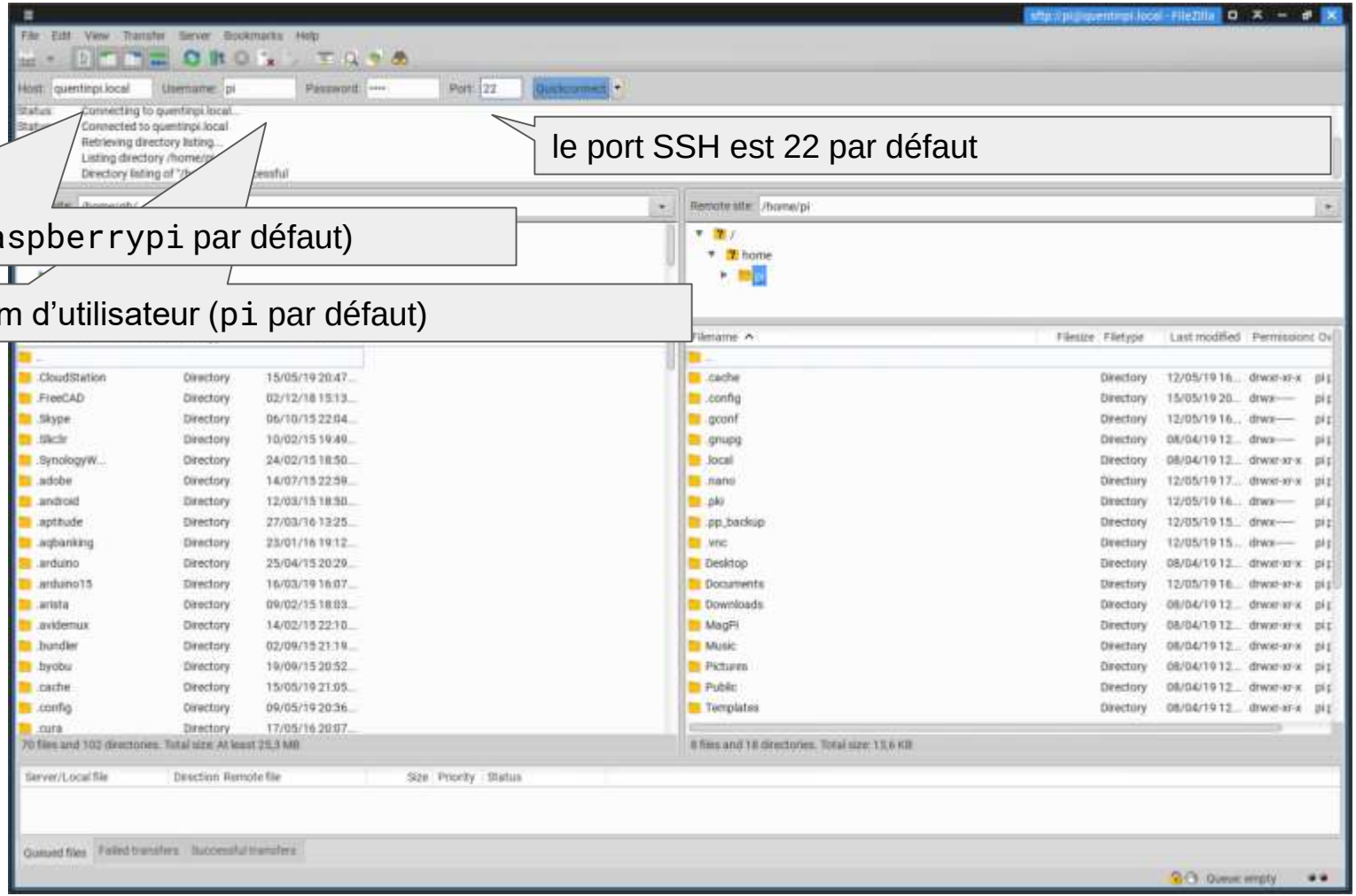
Côté **serveur** (le RPi):

rien à faire de plus, cela “passe” par une connection SSH

Côté **client** (votre ordinateur):

un client SFTP, par exemple FileZilla (fonctionne sous Linux/Windows/Mac OS X)

FileZilla



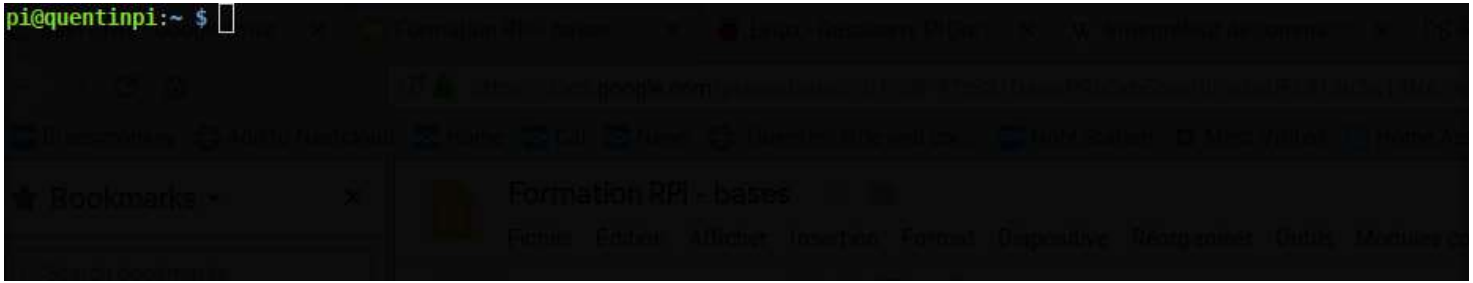
MakiLab
WE ARE ALL MAKERS !

Survivre en ligne de commande

Survivre sous Linux en ligne de commande

“L’interpréteur de commande” (CLI en anglais - Common Line Interface) est l’interface de commande privilégiée sous Linux. On l’appelle aussi parfois “Terminal”.

Elle sera souvent votre seul moyen de dialoguer avec le RPi, surtout si il n’a pas de clavier ou d’écran connecté.



Source [https://www.domo-blog.fr/le-raspberry-pi-et-la-ligne-de-commande/] et [http://www.semageek.com/les-42-commandes-les-plus-utiles-sur-raspberry-pi/]

Se déplacer...

- **`pwd`** affiche le répertoire actuel
- **`ls`** Liste les fichiers dans le répertoire actuel:
  - **`-l`** : affiche des infos additionnelles (permissions, date, taille)
  - **`-a`** : affiche tous les fichiers (y compris les fichiers “cachés” commençant par un . (point)
  - **`-h`**: affiche les tailles de fichiers de manière humaine (Mb, Gb, …)
- **`cd`** change le répertoire actuel

Besoin d’aide? **`man`** affiche la page de manuel d’une commande

- L’interpréteur de commande (`bash`) permet l’autocomplétion avec la touche TAB, ne pas s’en priver!
- 2 répertoires spéciaux: “**~**” = répertoire home de l’utilisateur, et “**-**” le précédent répertoire

L’arborescence classique sous Linux

```
pi@quentinpi:~ $ ls -l /
total 71
drwxr-xr-x   2 root root  4096 Apr  8 12:01 bin
drwxr-xr-x   3 root root  2560 Jan  1  1970 boot
drwxr-xr-x  14 root root  3320 May 12 23:15 dev
drwxr-xr-x 112 root root  4096 May 12 17:34 etc
drwxr-xr-x   3 root root  4096 Apr  8 11:48 home
drwxr-xr-x  16 root root  4096 Apr  8 11:58 lib
drwx------   2 root root 16384 Apr  8 12:40 lost+found
drwxr-xr-x   2 root root  4096 Apr  8 11:37 media
drwxr-xr-x   2 root root  4096 Apr  8 11:37 mnt
drwxr-xr-x   7 root root  4096 Apr  8 12:20 opt
dr-xr-xr-x 149 root root     0 Jan  1  1970 proc
drwx------   5 root root  4096 May 12 17:32 root
drwxr-xr-x  27 root root   840 May 15 21:22 run
drwxr-xr-x   2 root root  4096 Apr  8 12:01 sbin
drwxr-xr-x   2 root root  4096 Apr  8 11:37 srv
dr-xr-xr-x  12 root root     0 May 15 21:45 sys
drwxrwxrwt  14 root root  4096 May 15 21:17 tmp
drwxr-xr-x  11 root root  4096 Apr  8 12:09 usr
drwxr-xr-x  11 root root  4096 Apr  8 12:43 var
pi@quentinpi:~ $
```

Sous Linux, “everything is file”

/bin => Exécutables essentiels au système, utilisables par tous les utilisateurs (ls pwd cp)

/boot => fichiers permettant à Linux de démarrer

/dev => Point d'entrée de tous les périphériques (disque dur, écran, partition, consoles TTY)

/etc => contient les commandes et fichiers nécessaires à l'administrateur système (*.conf...)

/home => Répertoire personnel des utilisateurs

/lib => contient les bibliothèques partagées essentielles au système lors du démarrage

/media => Contient les points de montages pour les médias amovibles

/mnt => Point de montage pour monter temporairement un système de fichiers

/opt => Emplacement pour des applications installées hors gestionnaire de paquets (logiciels optionnels)

/proc => Répertoire virtuel pour les informations système (états du noyau et des processus système)

/root => Répertoire personnel de l'administrateur (le seul qui n’est pas dans /home)

/sbin => Contient les programmes système essentiels utilisables par l'admin uniquement.

/srv => Données pour les services du système (ex: documents de comptes FTP)

/tmp => Répertoire fichier temporaires, effacé au redémarrage

/usr => Contient des programmes installés (/usr/bin) avec leur librairies (/usr/lib ou /usr/lib64) tels que firefox, libreoffice, …)

/var => contient les données variables (fichiers de log) mais parfois les bases de données (/var/lib/mysql) et les pages de site web (/var/www/html)

Makilab
WE ARE ALL MAKERS !

Source [https://doc.ubuntu-fr.org/arborescence] et [https://www.linuxtricks.fr/wiki/arborescence-du-systeme-linux]

Manipuler les fichiers...

- **`cp`** copie un fichier
- **`mv`** déplace un fichier
- **`rm`** efface un fichier
- **`mkdir`** crée un répertoire
- **`rmdir`** supprime un répertoire si il est vide (ne contient pas de fichier)

Toutes ces commandes peuvent être utilisées avec les “expansions”

- **`*`** remplace tous les caractères
- **`?`** remplace un caractères
- **`[...]`** remplace les caractères entre []: exemple [0-9]

Editer les fichiers...

- **`less`** affiche le contenu d'un fichier texte (mais ne permet pas de l'éditer)
  (utiliser q pour quitter)
- **nano** est l'éditeur de texte par défaut: simple et efficace.

Mise en pratique: créer un répertoire "Scripts" et y mettre un fichier "lisezmoi.txt"

```
$> cd ~
$> mkdir Scripts
$> nano lisezmoi.txt

Ajouter le texte "Ce répertoire contiendra nos scripts python" et sauver

$> less lisezmoi.txt
```

Installer et mettre à jour des programmes

Raspbian utilise le gestionnaire de paquet “APT”:

- **`apt search`** pour rechercher un paquet
- **`apt show`** pour les informations détaillées d’un paquet
- **`sudo apt install`** pour installer un paquet
- **`sudo apt remove`** pour désinstaller un paquet
- **`sudo apt update`** pour mettre à jour la liste des paquets (programmes)
- **`sudo apt upgrade`** pour mettre à jour les paquets installés à la dernière version

Un exemple pratique: installons les pages de manuels (man pages) en français

```
$> apt search manpages french
...
$> sudo apt install manpages-fr-extra
```

S’initier à Python

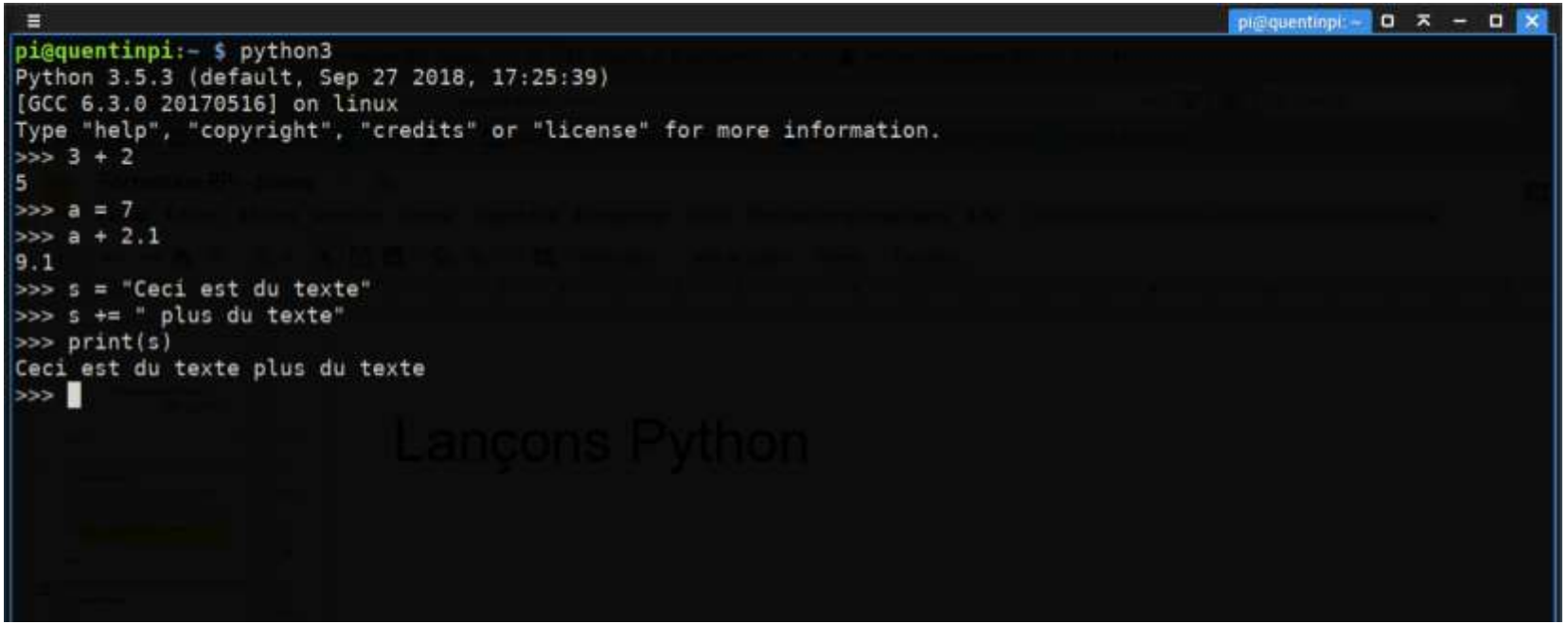
Source image [https://docs.python.org/fr/3/]

Pourquoi Python?

- C’est le standard “de facto” pour la programmation sur Raspberry Pi.
- C’est un langage de programmation qui reste simple et lisible.
- Il permet de facilement programmer les GPIO’s de la carte
- L’écosystème (les bibliothèques ou modules additionnels) est énorme

Il y a deux versions de Python: Python 2.x et Python 3.x (sorti en 2008).
Python 2 n’est plus développé depuis 2.7 (sorti en 2010).
→ Python 3 est recommandé!

Lançons Python



MakiLab
WE ARE ALL MAKERS !

Un script (fichier .py)

```
$> cd ~/Scripts
$> nano helloworld.py
```

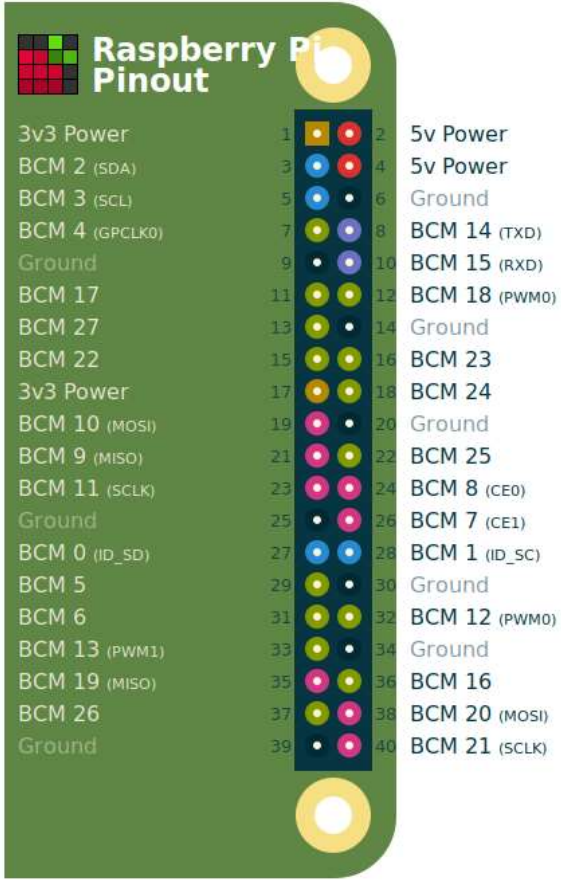
```
import datetime

# Affiche un texte d'accueil, puis la date et l'heure courante
print("Bonjour tout le monde")
aujourdhui = datetime.date.today()
print("Nous sommes le", aujourdhui)
# weekday() renvoie le jour de la semaine, 0 pour lundi, 6 pour
dimanche
if aujourdhui.weekday() == 5 or aujourdhui.weekday() == 6:
          print("Chouette c'est le weekend")
else:
          print("C'est la semaine")
```

```
$> python3 helloworld.py
```

Les GPIO’s et la programmation “physique”

Source [https://www.raspberrypi.org/documentation/usage/gpio/README.md]

GPIO’s




```
$> pinout
```

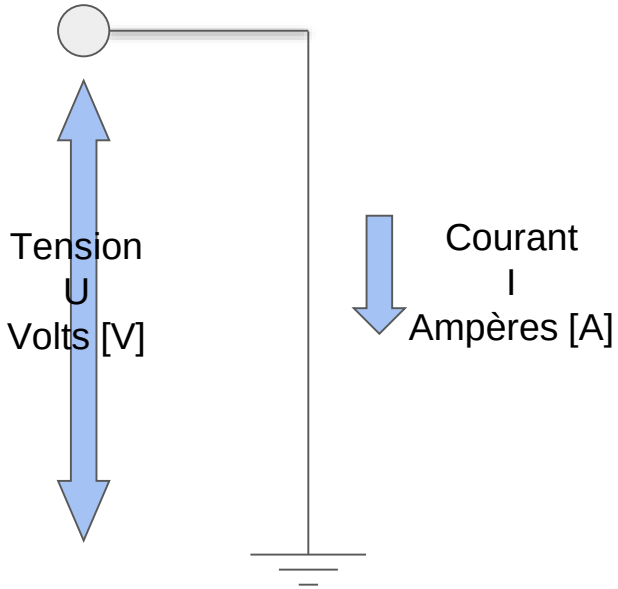
Makilab
WE ARE ALL MAKERS !

Source image [https://pinout.xyz]

Power on!

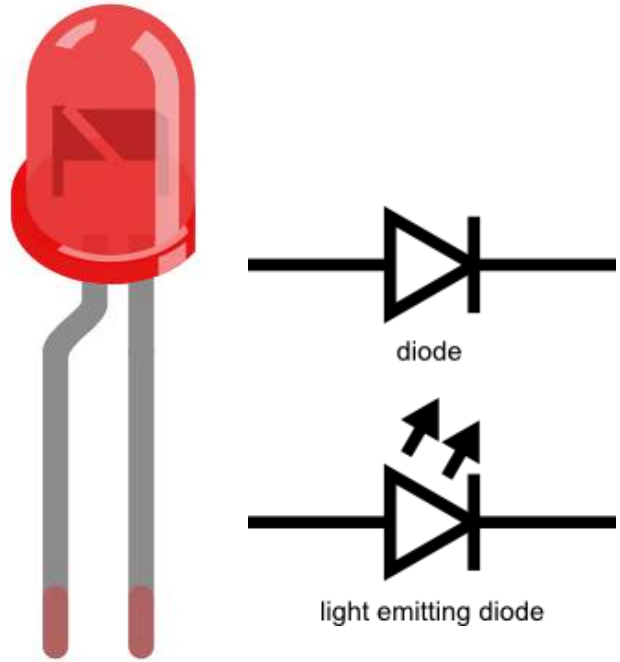
Quelques notions d’électricité:

- C’est une forme d’énergie (comme la chaleur, la lumière, …)
- Elle se déplace dans des conducteurs (comme les fils)
- On peut la convertir de et vers d’autres formes d’énergie
- AC/DC



LED the fun begin

Allumons une LED



Anode **+** **-** Cathode

Source image: Fritzing [www.fritzing.org] & SparkFun [https://learn.sparkfun.com/tutorials/light-emitting-diodes-leds]

LED the fun begin

Trop de courant tue une LED,
en règle générale

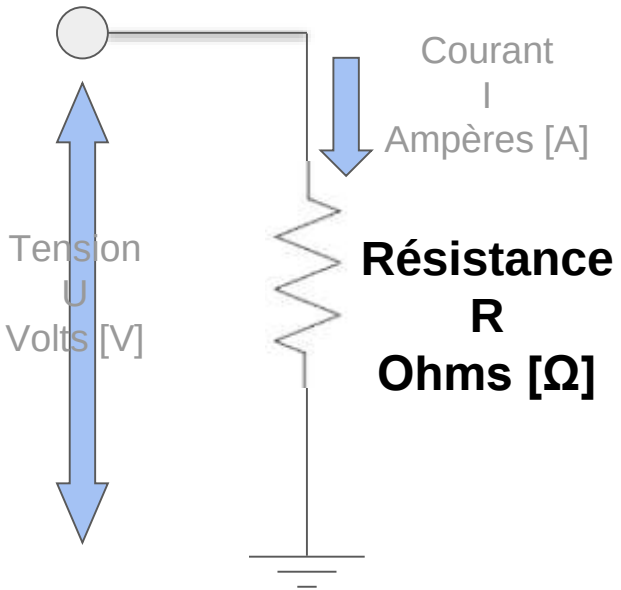
max **20mA**

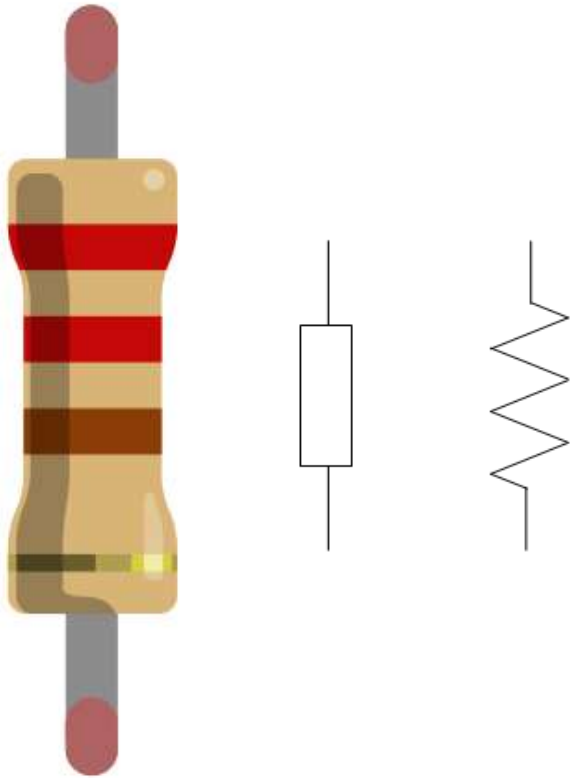
et le Raspberry Pi peut supporter jusqu'à

50mA par pin I/O

Source image: SparkFun [https://learn.sparkfun.com/tutorials/light-emitting-diodes-leds]

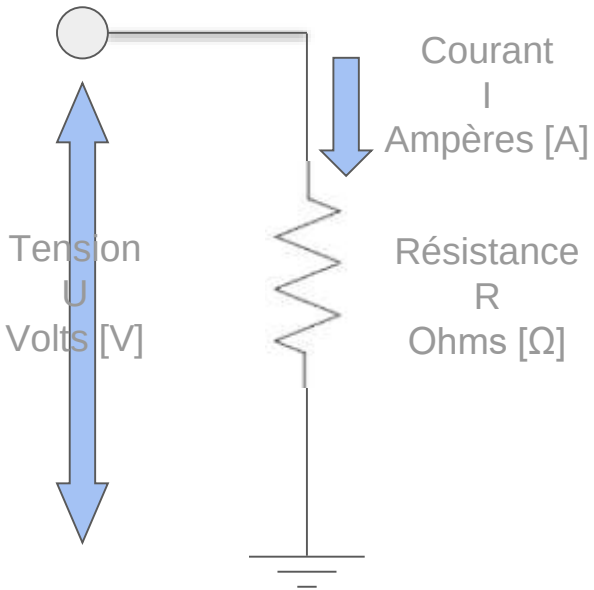
Ohm my god





Source image: Fritzing [www.fritzing.org] & Quora [https://www.quora.com/What-is-the-symbol-of-a-fixed-resistor]

Ohm my god



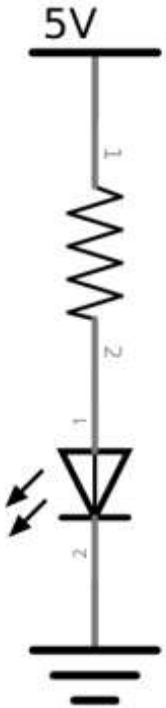
$$U = R * I$$

$$I = U / R$$

$$R = U / I$$

MakiLab
WE ARE ALL MAKERS !

En pratique



?

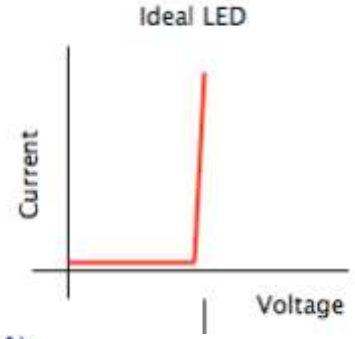
Quel est R pour que I soit <= 20mA

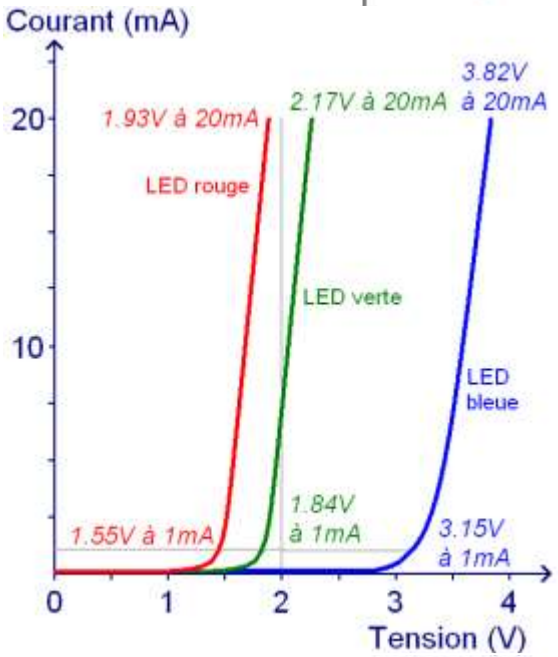
Retour sur la LED

En 1ère approximation:

$$R_{LED} \simeq 0\ \Omega \text{ et } U_{LED} \simeq \text{entre } 1.8 \text{ V et } 3.2 \text{ V}$$

DIALIGHT P/N	EMITTED COLOR	MATERIAL	LENS COLOR	LUMINOUS INTENSITY (mcd)			DOMINANT WAVELENGTH (nm)			FORWARD VOLTAGE (V)			VIEWING ANGLE
				If = 20 ma			If = 20 ma			If = 20 ma			* DEGREES
				MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
598-8010-107F	RED	AlInGaP	Water Clear	30	40	80	630	635	642	1.7	2.2	2.4	140
598-8020-107F	RED-ORANGE	AlInGaP	Water Clear	120	150	200	620	625	630	1.7	2	2.4	140
598-8030-107F	ORANGE	AlInGaP	Water Clear	70	-	150	600	-	610	1.7	2	2.4	140
598-8040-107F	YELLOW	AlInGaP	Water Clear	100	130	160	590	-	595	1.7	2	2.2	140
598-8050-107F	YELLOW	AlInGaP	Water Clear	100	130	160	583	-	590	1.7	2	2.4	140
598-8060-107F	YELLOW-GREEN	AlInGaP	Water Clear	20	40	60	570	-	575	1.8	2	2.4	140
598-8070-107F	GREEN	GaP	Water Clear	10	20	40	562	-	570	1.8	2	2.4	140
598-8081-107F	GREEN	InGaN	Water Clear	220	300	400	520	523	525	3	3.2	3.5	140
598-8091-107F	BLUE	InGaN	Water Clear	90	140	160	470	473	475	2.8	3.2	3.5	140

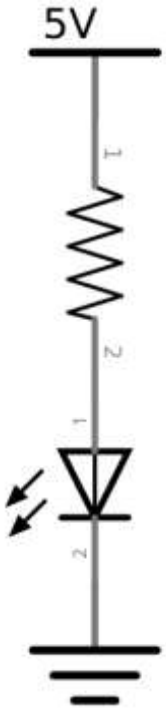




Source image : [http://www.gbtbox.myzen.co.uk/Tutorial/LEDs.html] & [http://dangerousprototypes.com/docs/Basic_Light_Emitting_Diode_guide] & [https://www.astuces-pratiques.fr/electronique/led-et-calcul-de-la-resistance-serie]

Makilab
WE ARE ALL MAKERS !

En pratique



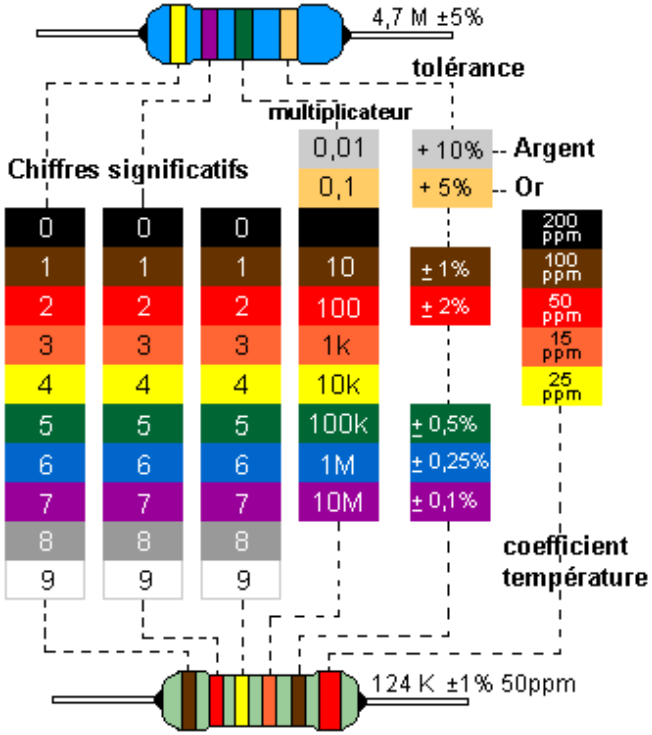
On peut la calculer précisément …

ou retenir que typiquement:

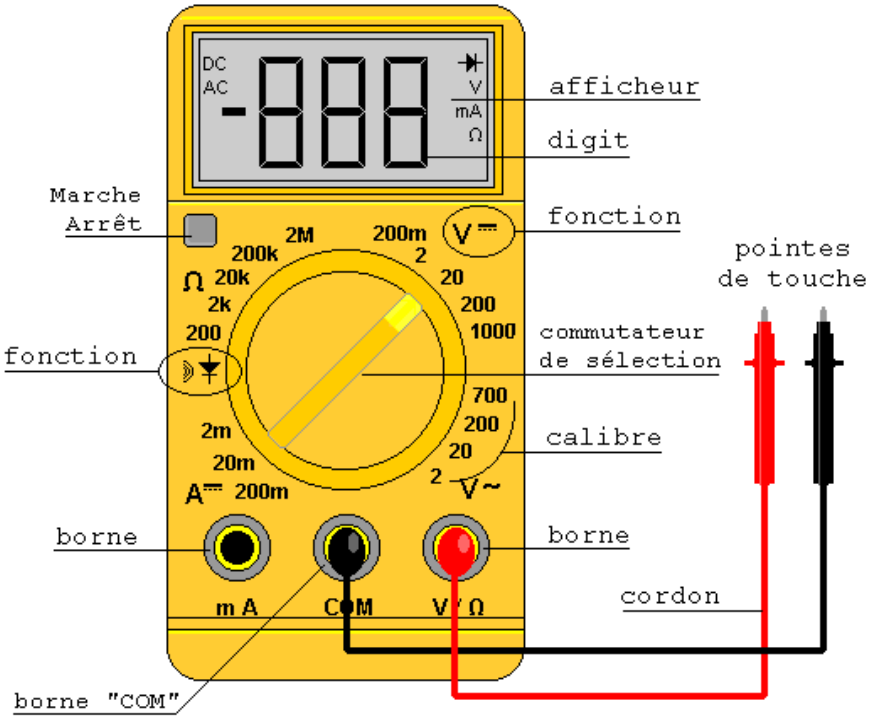
220 Ω

MakiLab
WE ARE ALL MAKERS !

Ma résistance a quelle valeur?

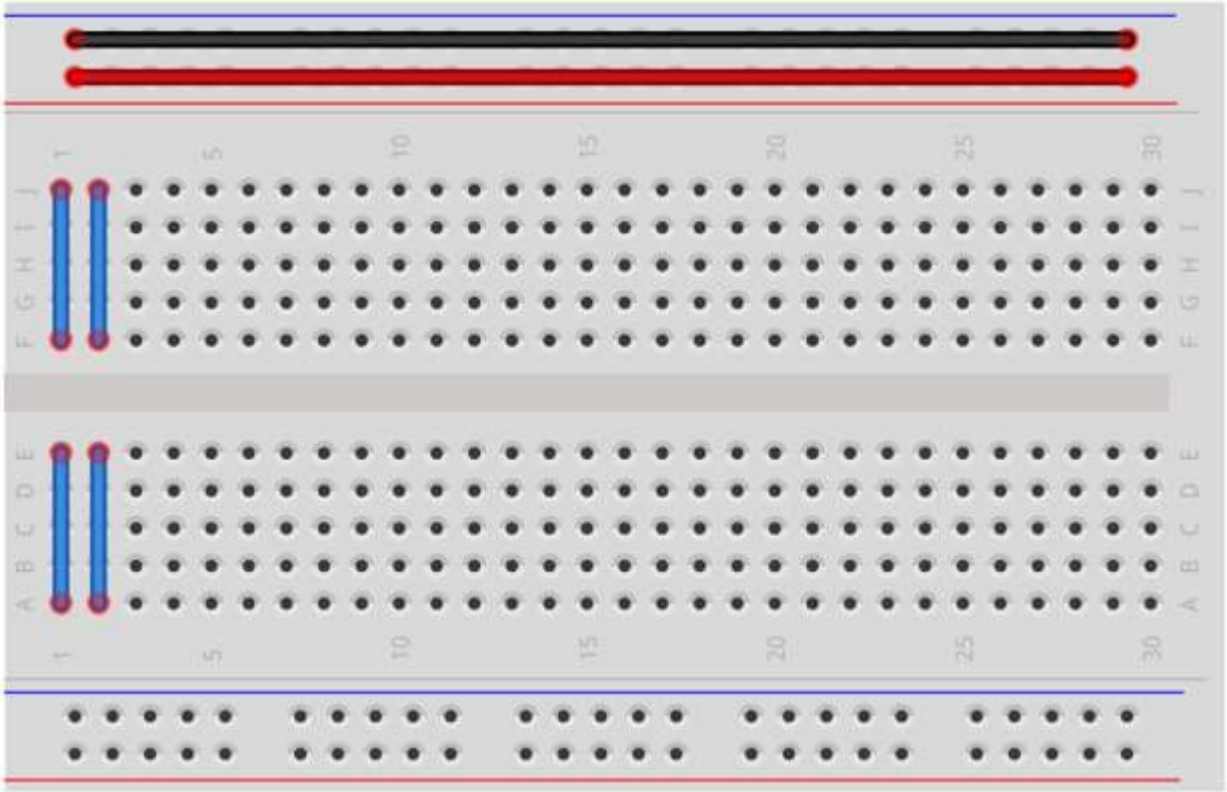


Ou...



Source image: Positron LIbre [https://www.positron-libre.com/cours/electronique/resistances/code-couleurs-resistances.php]
& ePieces [http://blog.epieces.fr/2017/01/18/guide-dutilisation-du-multimetre/]

“Breadboard”

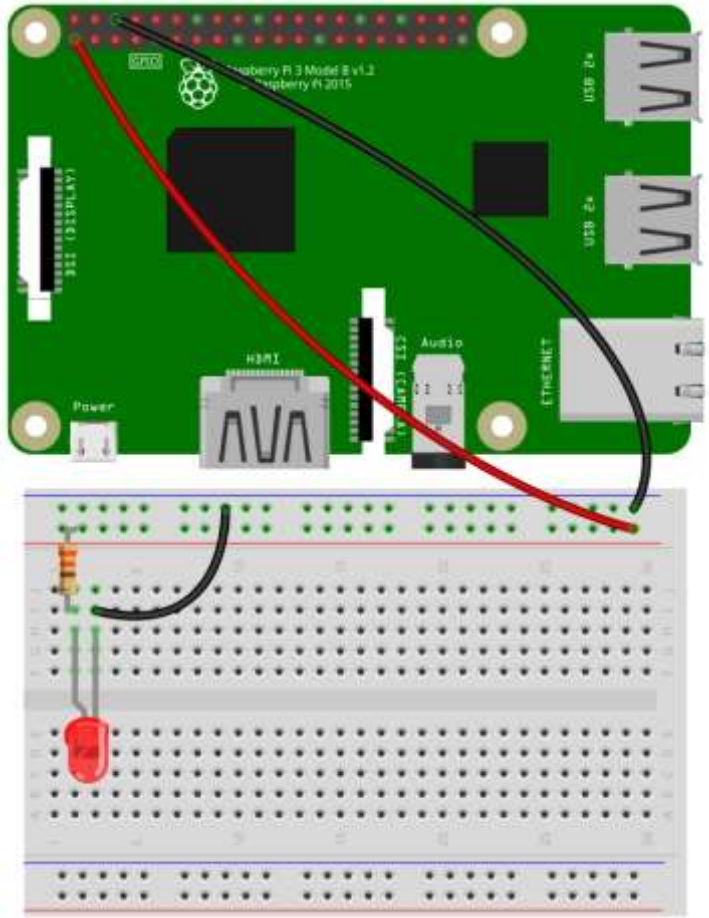


MakiLab
WE ARE ALL MAKERS !

Circuit LED

- Un circuit doit être “fermé” pour fonctionner.
- Toute l’énergie d’un circuit est utilisée/convertie, donc la tension chute toujours de la tension d’entrée à zéro.
- Le courant électrique cherche toujours le circuit de moindre résistance…

Court-circuit



MakiLab
WE ARE ALL MAKERS !

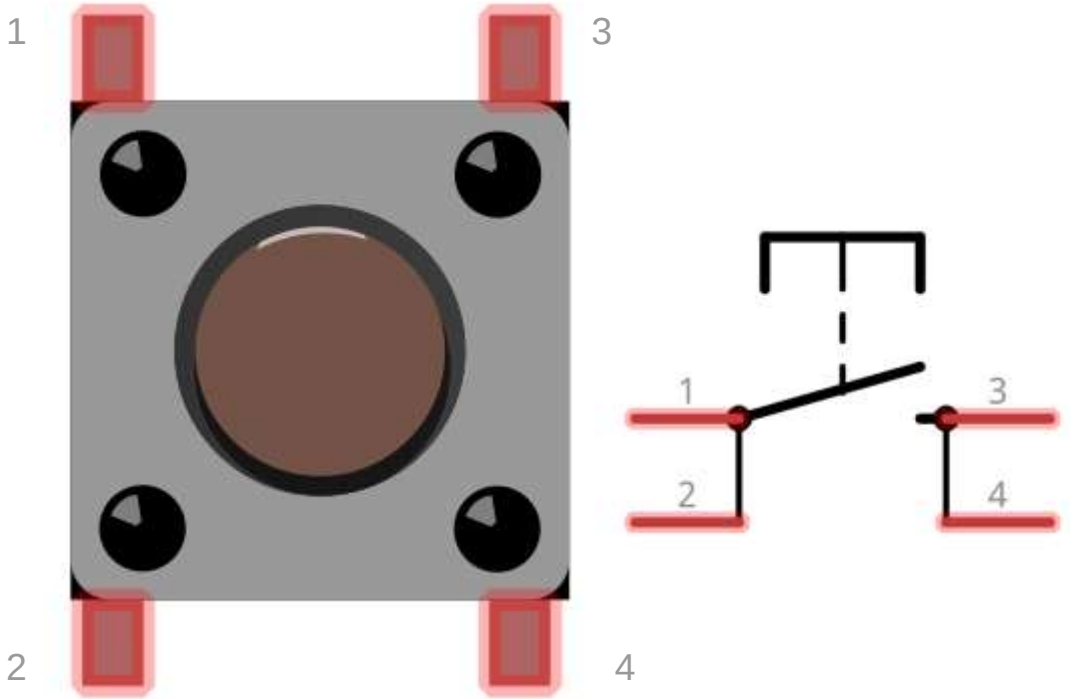
Sécurité ?

Faible courants et tensions
Pas de composants capacitifs utilisés aujourd'hui

Pas de risque pour vous… mais un peu pour le matériel

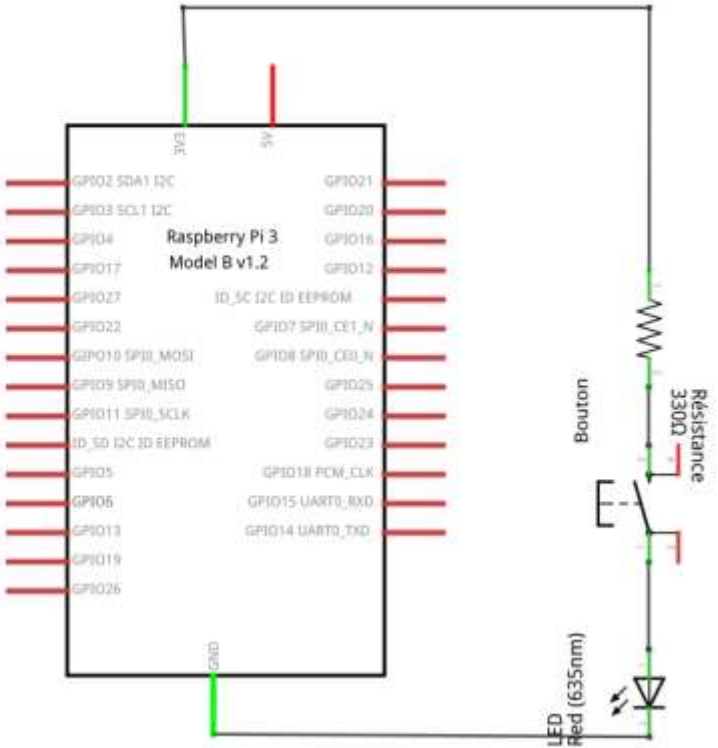
Conseillé de travailler hors tension
& de vérifier le montage avant mise sous tension

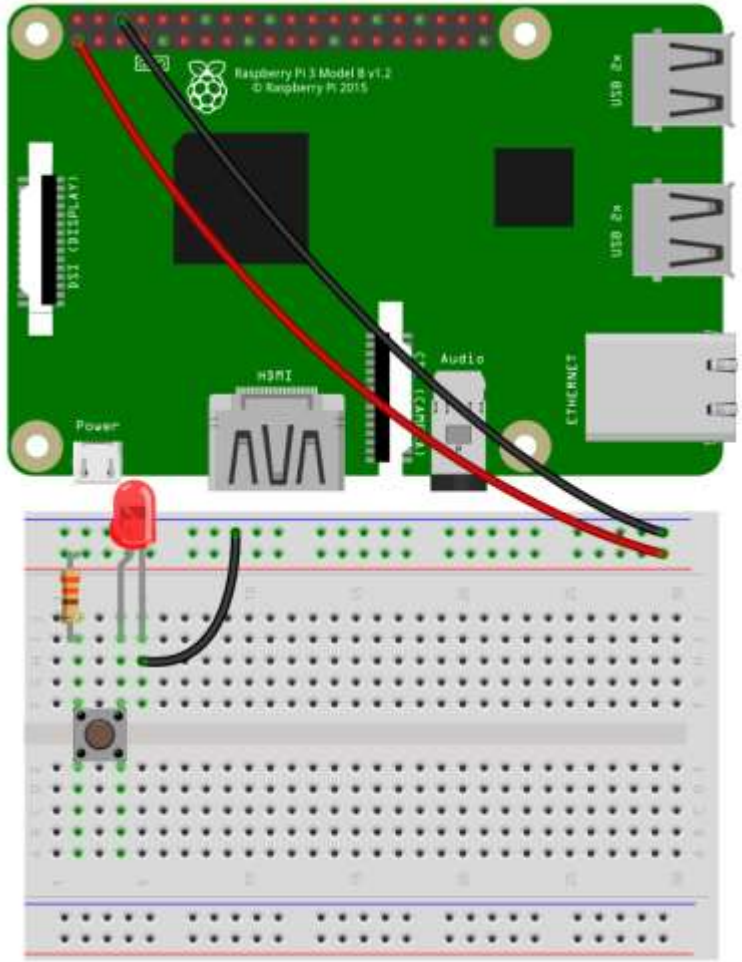
On / Off



Source image: Fritzing [www.fritzing.org]

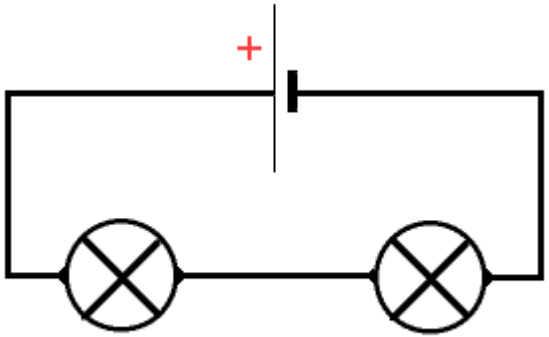
Circuit On/Off

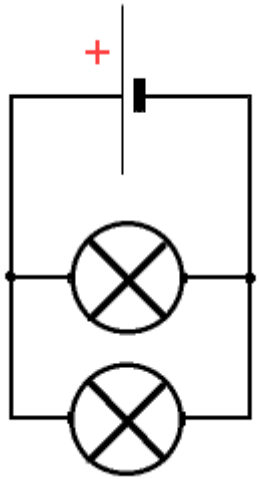




MakiLab
WE ARE ALL MAKERS !

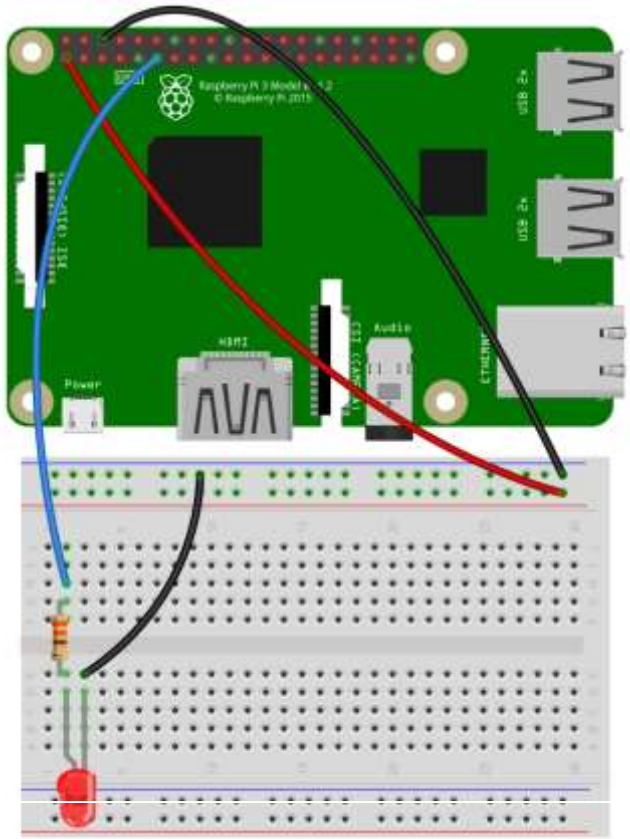
Série vs Parallèle





Source image: [http://users.skynet.be/fa346009/doc_animations/serie_para.htm]

~~Allumons~~ Programmons l'allumage d'une LED

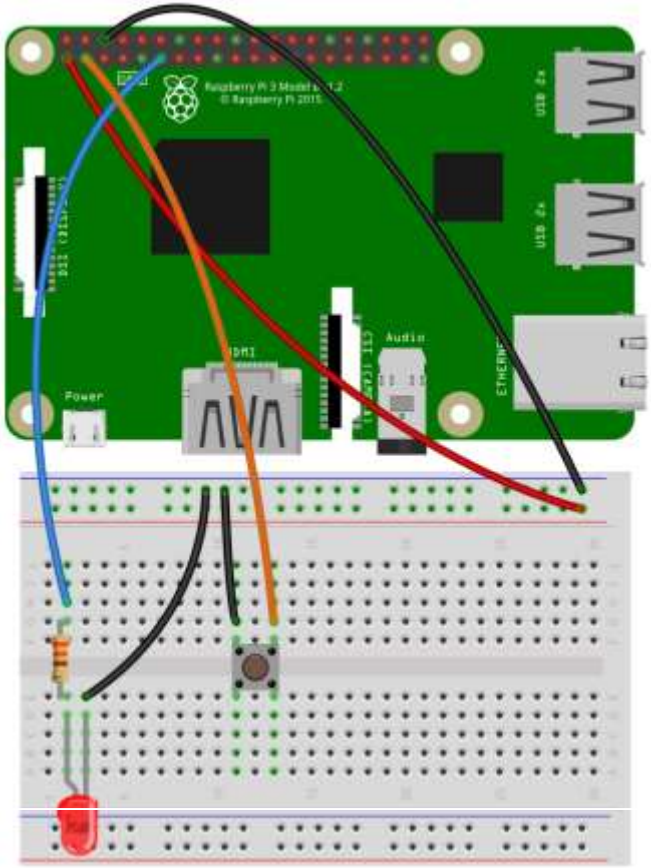


```
from gpiozero import LED
from time import sleep

led = LED(17)

while True:
    led.on()
    sleep(1)
    led.off()
    sleep(1)
```

Attention: les GPIO's du Raspberry Pi fonctionnent en 3.3v
(au lieu de 5V sur l'Arduino)

Source: [https://www.raspberrypi.org/blog/gpio-zero-a-friendly-python-api-for-physical-computing/]

Ajoutons un bouton

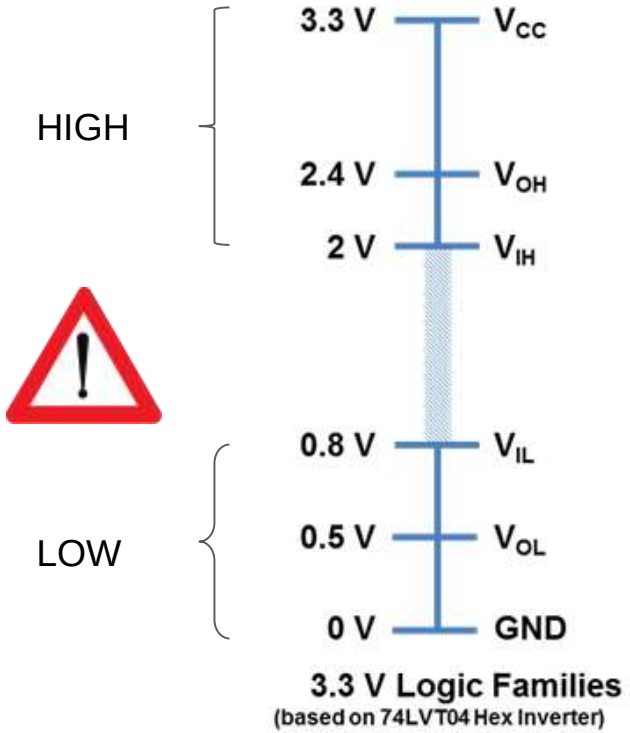



```
from gpiozero import LED, Button
from signal import pause

led = LED(17)
button = Button(2)

button.when_pressed = led.on
button.when_released = led.off

pause()
```

Source: [https://www.raspberrypi.org/blog/gpio-zero-a-friendly-python-api-for-physical-computing/]

Encore un petit mot sur les entrées digitales

Attention à la zone grise entre les niveaux HIGH et LOW



Source image: Arduino / Wikipedia [https://learn.sparkfun.com/tutorials/logic-levels/all]

That’s all folks

Formation faite au Makilab (www.makilab.org - info@makilab.org - Quentin Berten)

Diffusé sous licence (http://creativecommons.org/licenses/by-sa/4.0/)

Makilab
WE ARE ALL MAKERS !