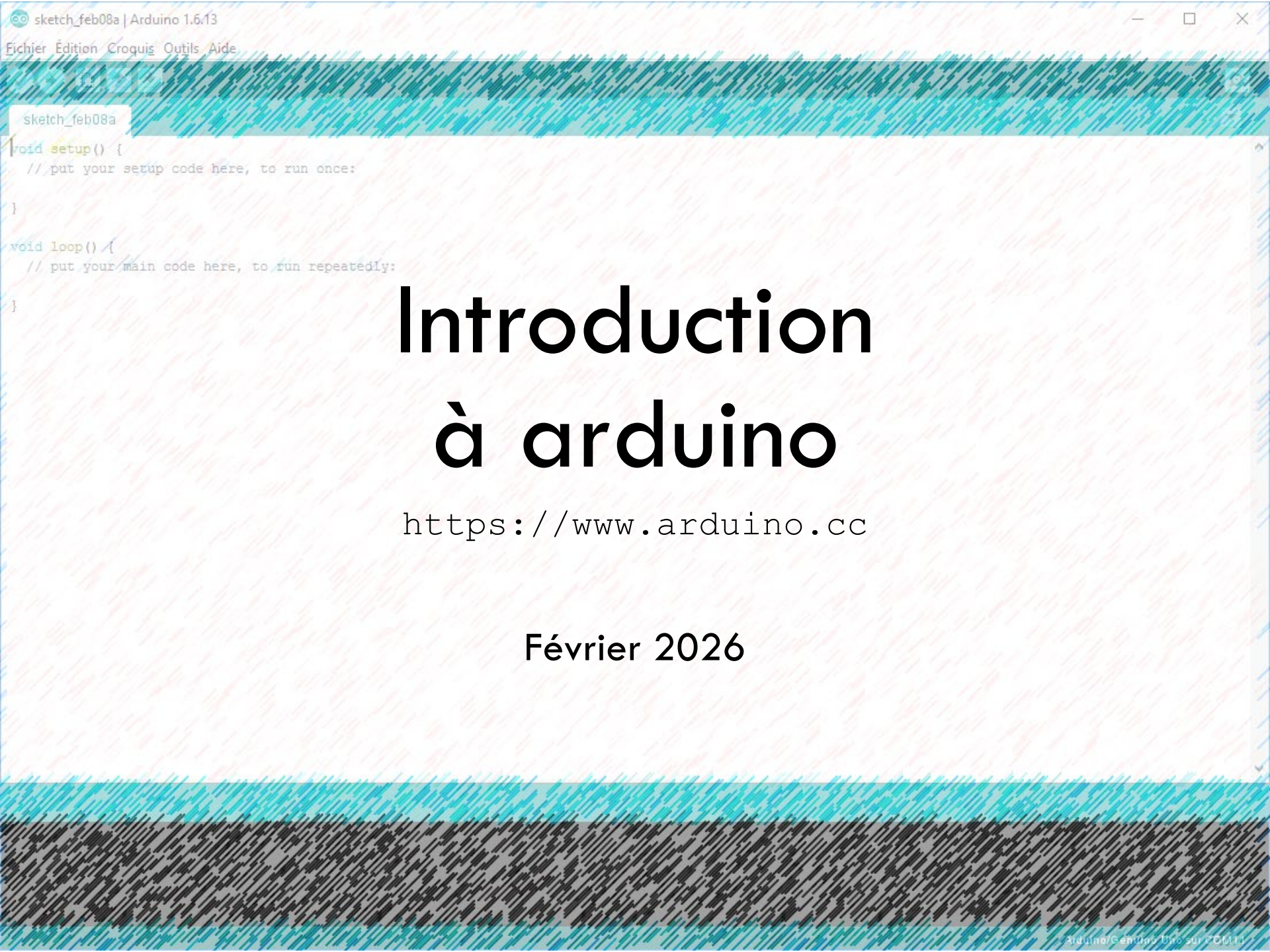


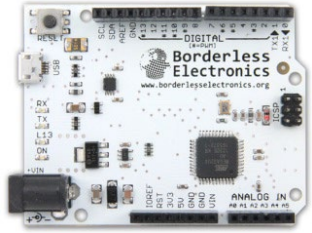
Introduction à arduino

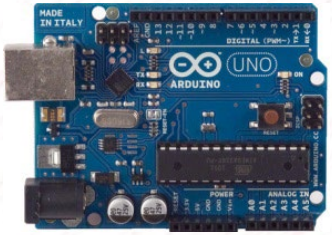
https://www.arduino.cc

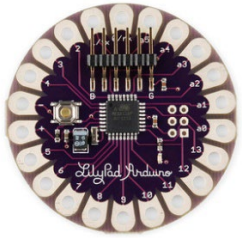
Février 2026

Qu'est ce qu'Arduino ?

C'est d'abord du matériel !





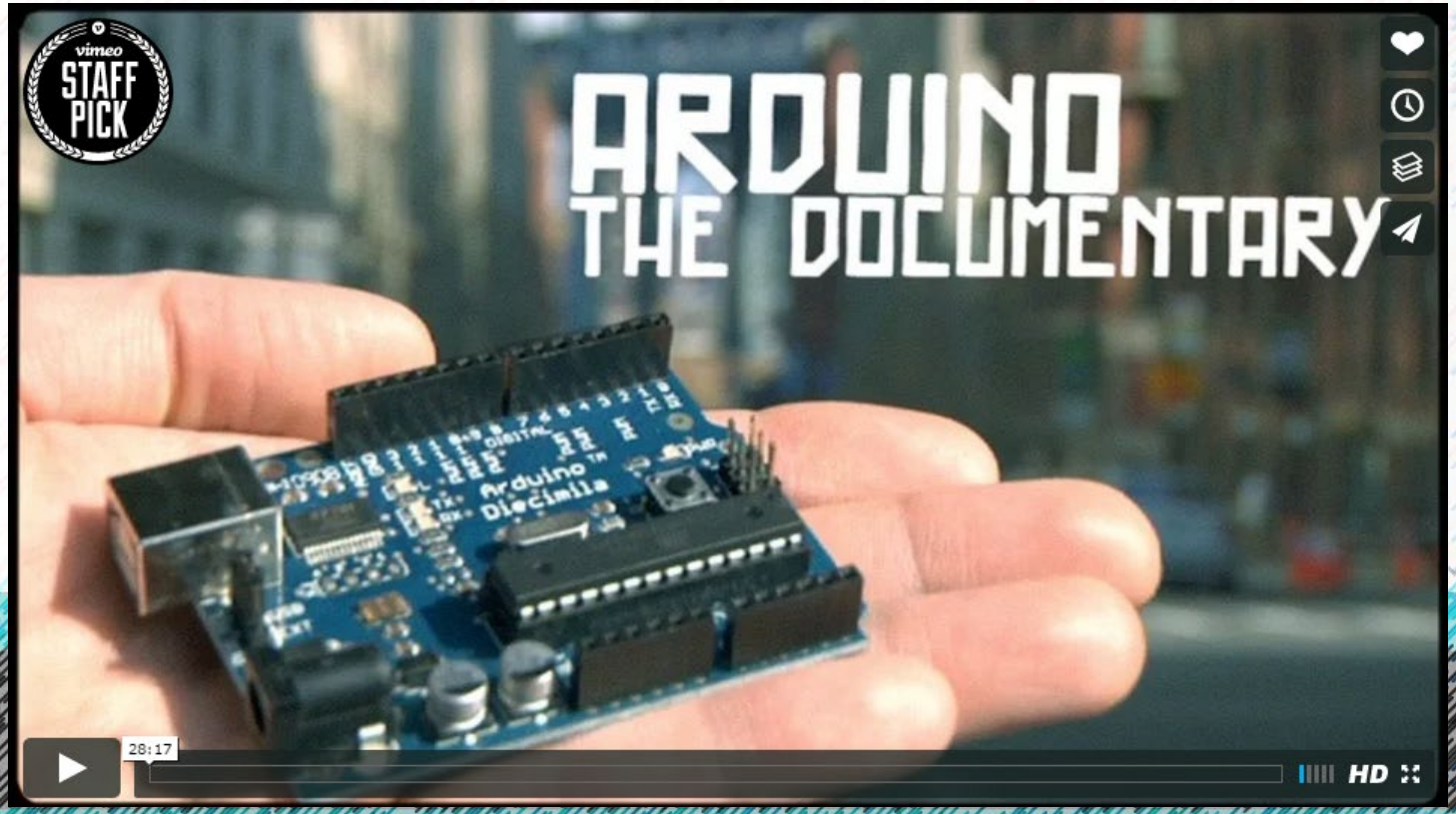


avec plein de versions différentes !

Une histoire d'Arduino ...

https://youtu.be/D4D1WhA_mi8

https://arduinohistory.github.io



Historique

Design by Numbers
http://dbn.media.mit.edu
Date : 1999-2001
Lieu : MIT Media Lab
John Maeda

Visible Language Workshop
http://museum.mit.edu/150/115
Date : 1975
Lieu : MIT
Muriel Cooper

Processing
https://www.processing.org
Date : Printemps 2001
Lieu : MIT Media Lab
Ben Fry / Casey Reas

Processing 4

p5.js

Wiring
http://wiring.org.co
Date : 2003
Lieu : IDII
Hernando Barragán

Arduino
https://www.arduino.cc
Date : 2005
Lieu : IDII
Massimo Banzi

Genuino

ARDUINO

ARDUINO

Qualcomm

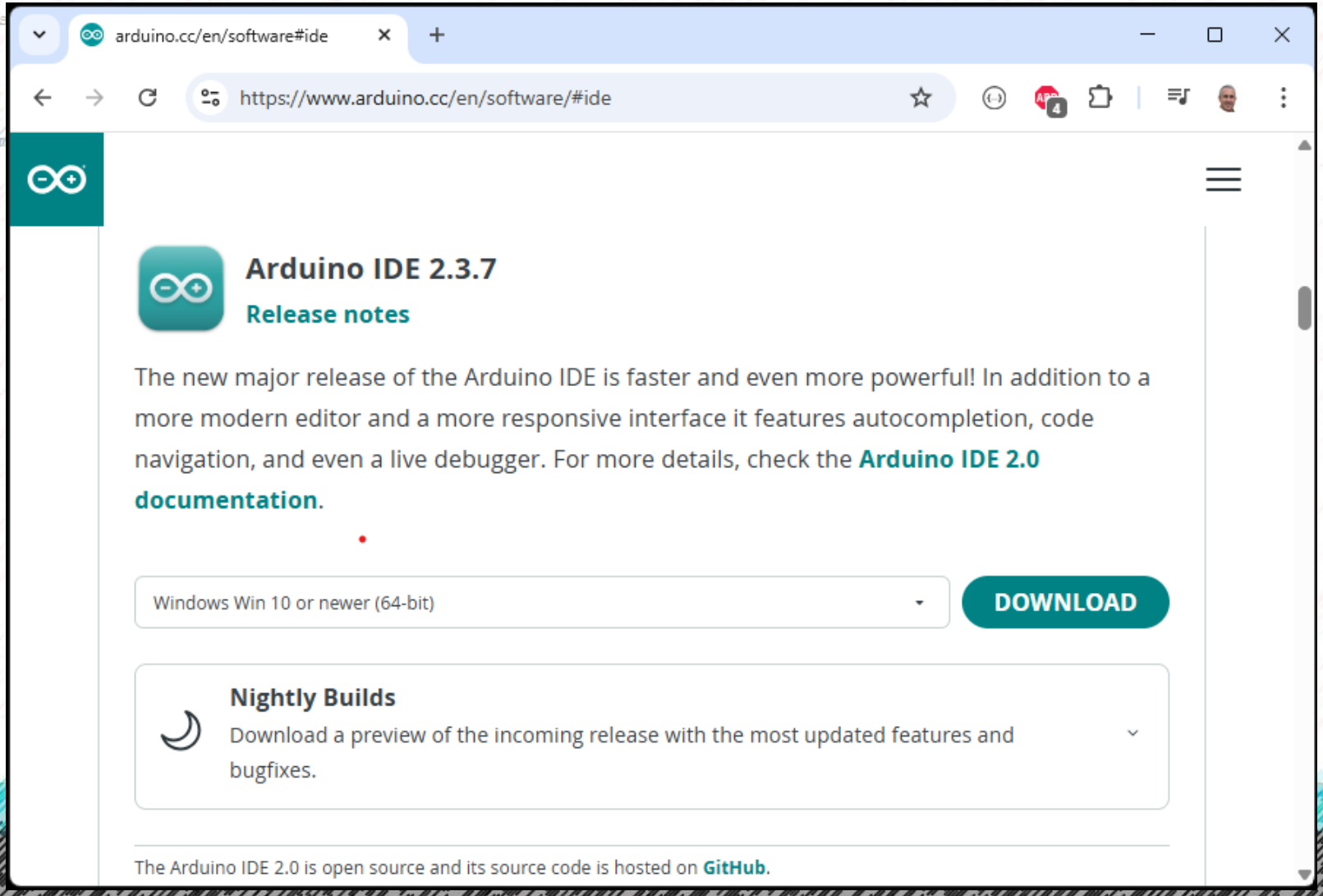
ARDUINO

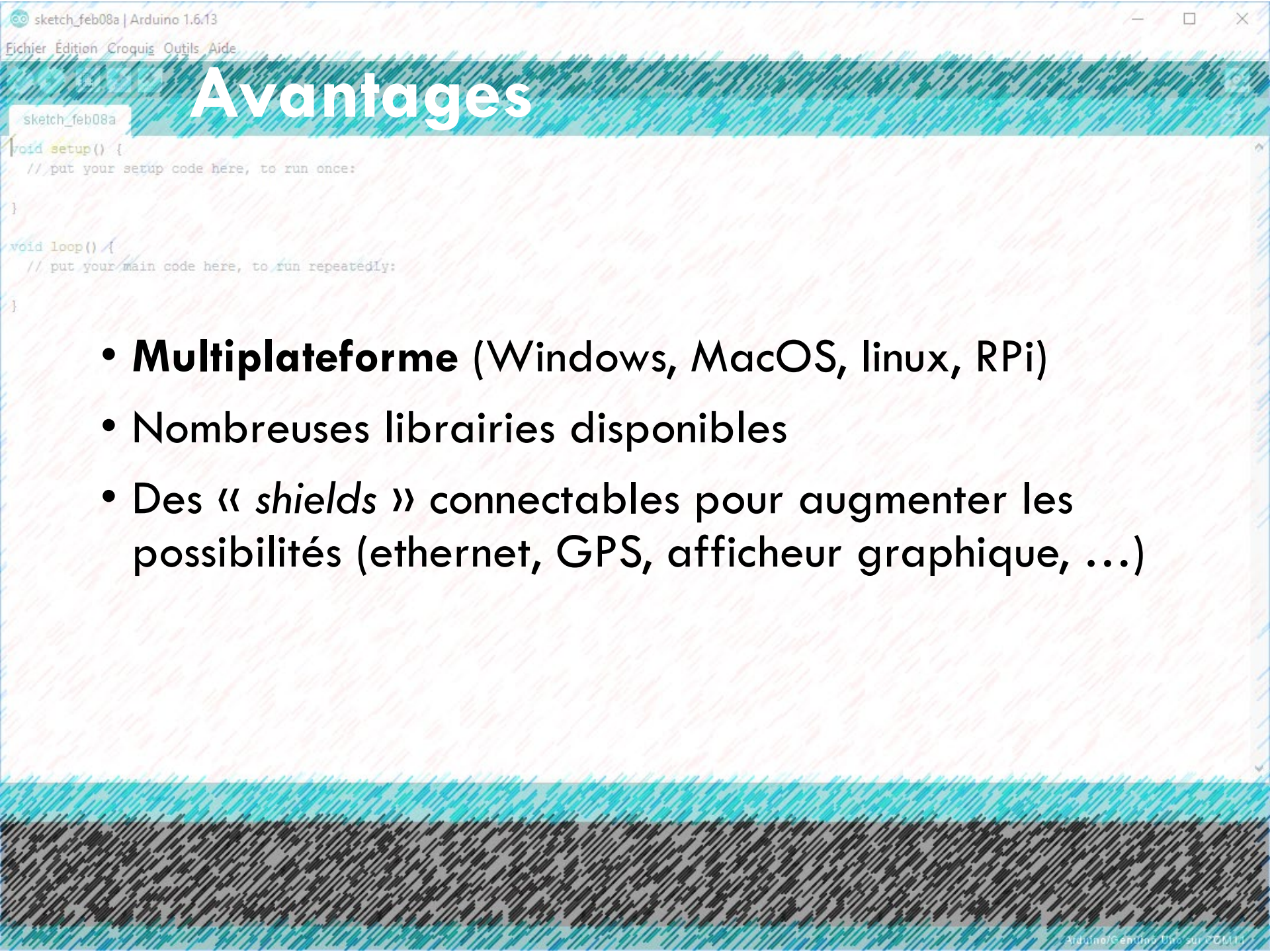
Arduino, c'est aussi du code …

- Une syntaxe simple
- du codage *bas niveau* (C/C++)

- Des possibilités d'extension via des librairies

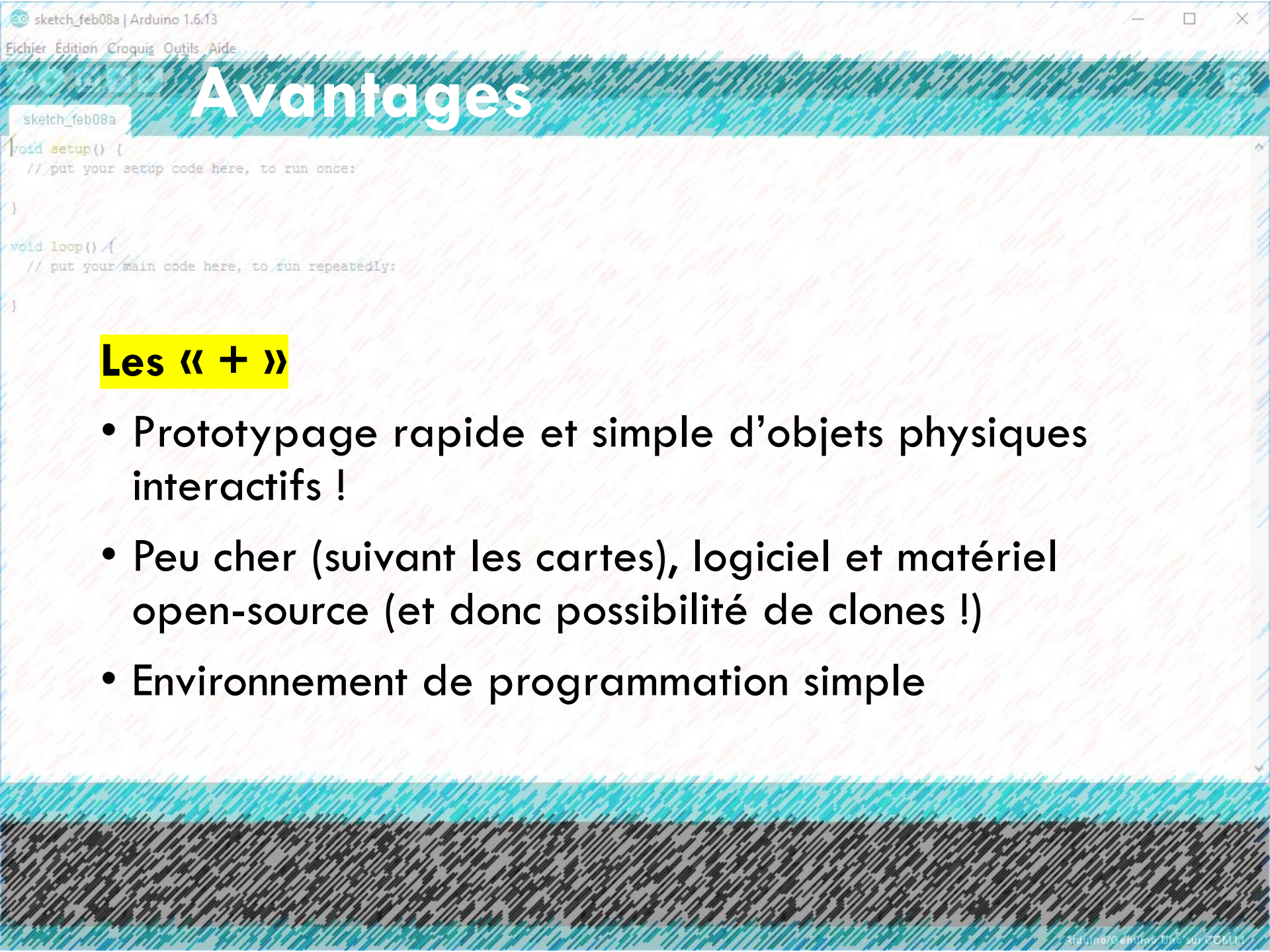
IDE – Environnement de dév.





Avantages

- **Multiplateforme** (Windows, MacOS, linux, RPi)
- Nombreuses librairies disponibles
- Des « *shields* » connectables pour augmenter les possibilités (ethernet, GPS, afficheur graphique, …)



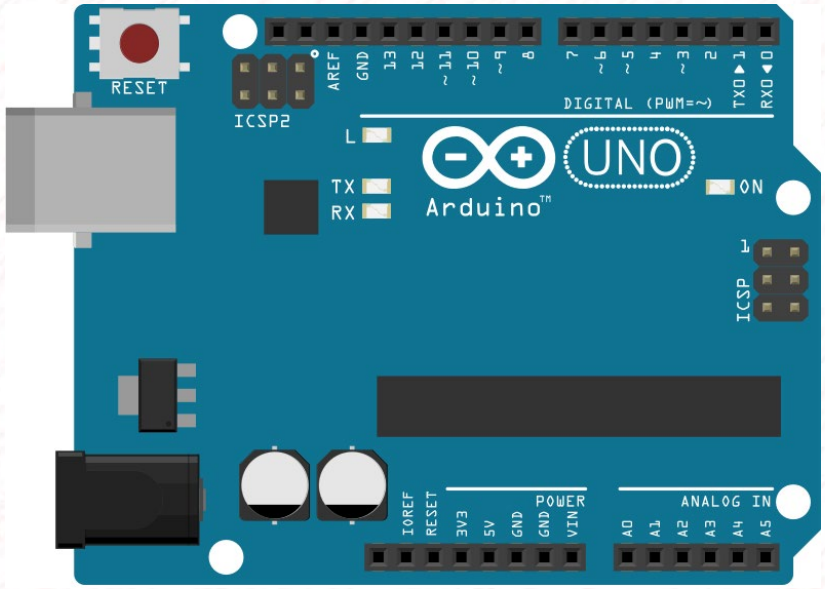
Avantages

Les « + »

- Prototypage rapide et simple d'objets physiques interactifs !
- Peu cher (suivant les cartes), logiciel et matériel open-source (et donc possibilité de clones !)
- Environnement de programmation simple

La carte arduino UNO …

- Des entrées/sorties numériques
- Des entrées analogiques (**A**)
- …



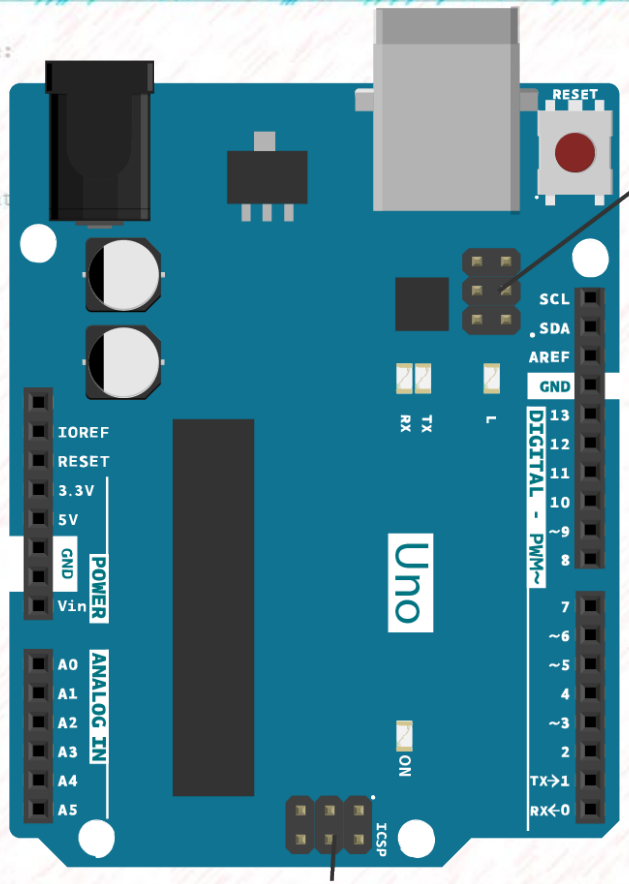
arduino UNO pinout

IORef: 5V

Vin: 7-12V DC max.

Serial: Serial is attached to pins 0 and 1, and to the USB-Serial micrcontroller on board.

The Uno has a second microcontroller on board to handle USB-to-serial communications.
This is the ICSP header for that microcontroller.



GPIO	ADC	Comm.	PWM	Interrupts	
GPIO18	ADC4	SDA			
GPIO19	ADC5	SCL			
	AREF				
Gnd					
GPIO13		SCK			LED
GPIO12		MISO			
GPIO11		MOSI	PWM11		
GPIO10		CS	PWM10		
GPIO9			PWM9		
GPIO8					
GPIO7					
GPIO6			PWM6		
GPIO5			PWM5		
GPIO4					
GPIO3			PWM3	INT1	
GPIO2				INT0	
GPIO1		TX			
GPIO0		RX			

IORef
Reset
+3.3V
+5V
Gnd
Gnd
Vin

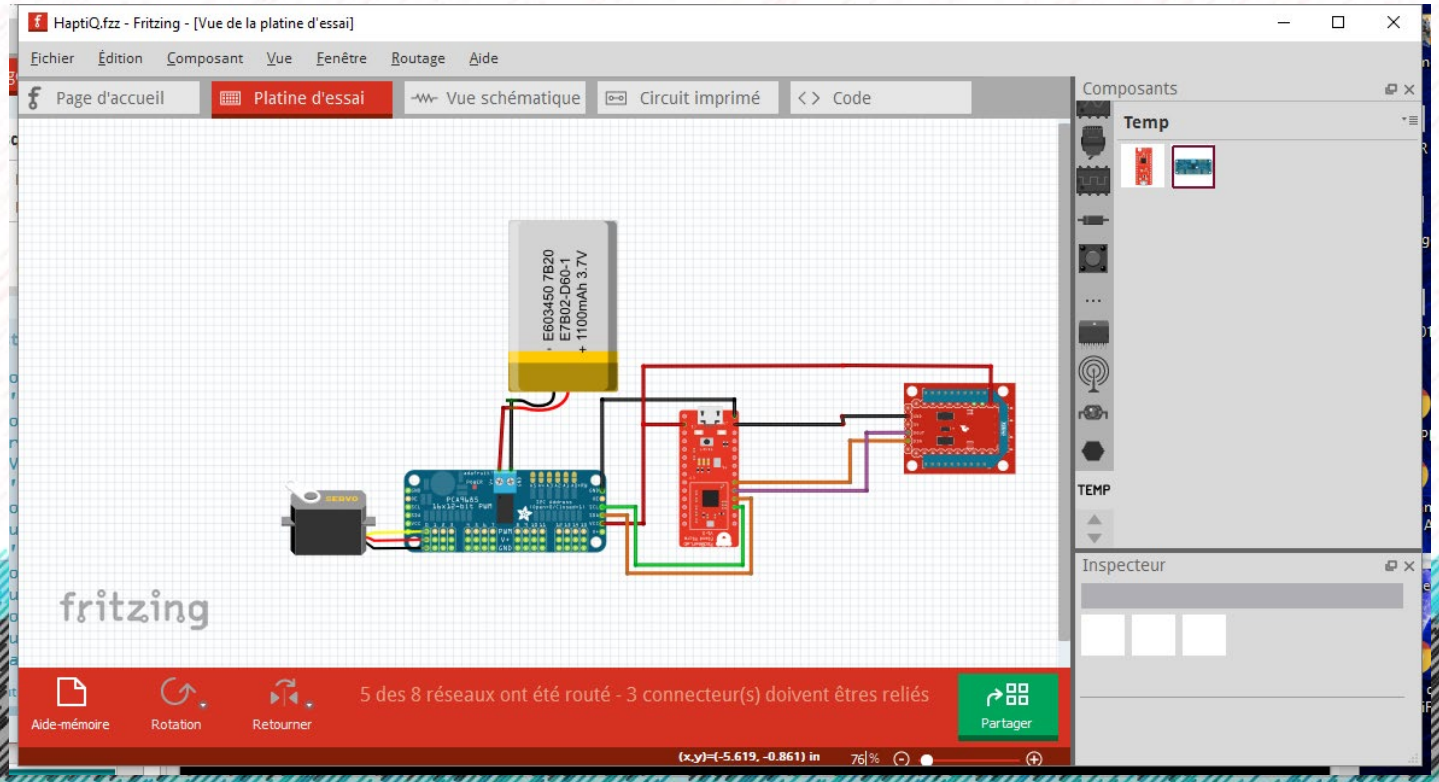
Comm.	ADC	GPIO
	ADC0	GPIO14
	ADC1	GPIO15
	ADC2	GPIO16
	ADC3	GPIO17
SDA	ADC4	GPIO18
SCL	ADC5	GPIO19

ICSP:

Reset	SCK	MISO
Gnd	MOSI	+5V

Un outil d'aide au montage

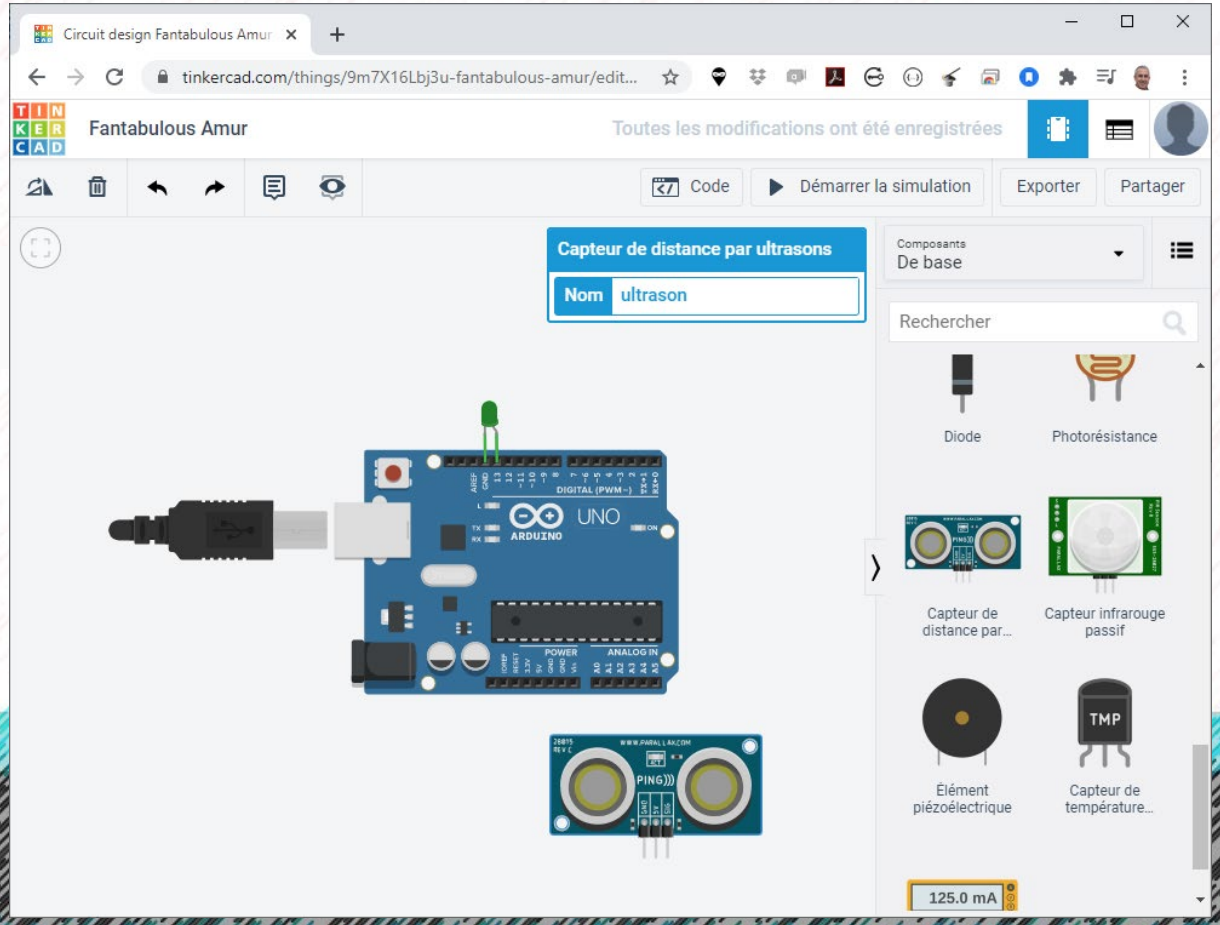
- **Fritzing** - https://fritzing.org (payant depuis 2019)
  https://www.softpedia.com/get/Science-CAD/Fritzing.shtml#download



Un simulateur en ligne

- https://www.tinkercad.com/dashboard
- Choisir

Circuits



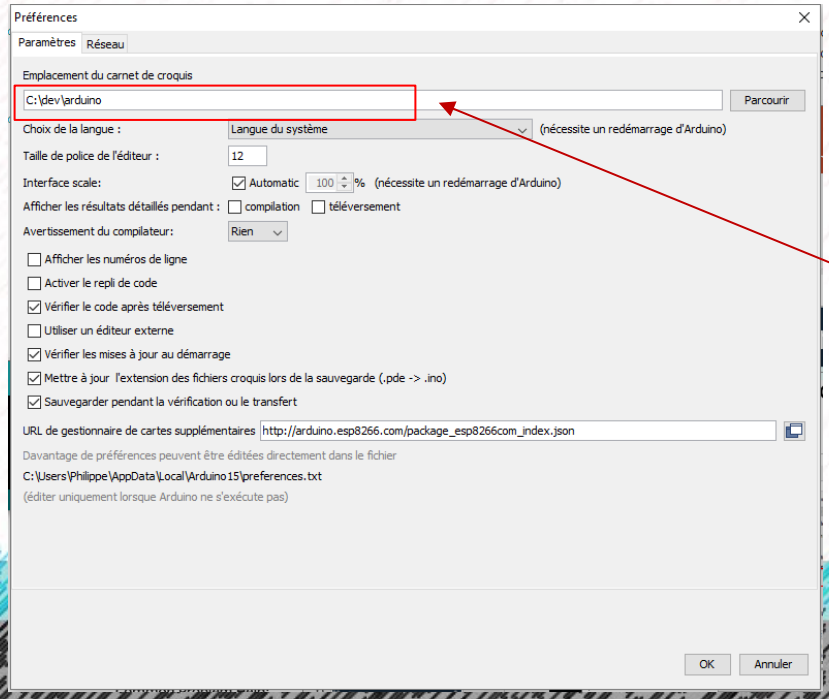
Programmation arduino

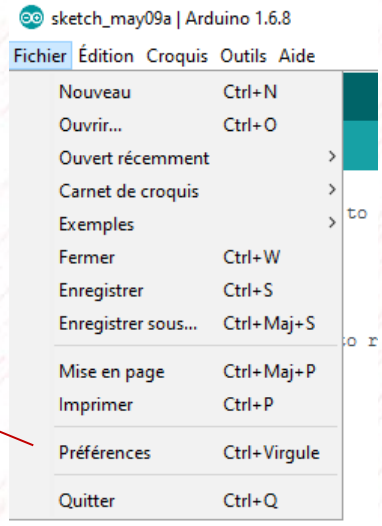
Arduino est ***un langage commun*** (syntaxe C++) indépendant des langages bas-niveau permettant de prototyper rapidement des applications physiques.

La base du programme arduino est le « *sketch* » (programme, prototype)
L'extension est le « **.ino** »

Structure

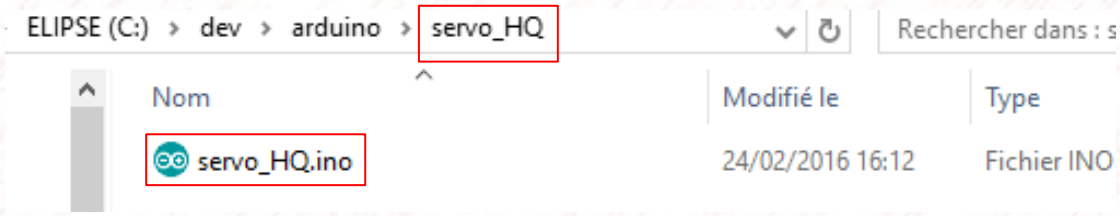
- Les « sketchs » (programmes) sont localisés dans le répertoire « préférences »





Structure

- un sketch est composé de :
  - Au moins un fichier **.ino** (cela peut être plus – un par classe objet).
    Le fichier principal doit **avoir le même nom** que le répertoire du sketch



Deux fonctions basiques

sketch_feb08a | Arduino 1.6.13

Fichier Édition Croquis Outils Aide

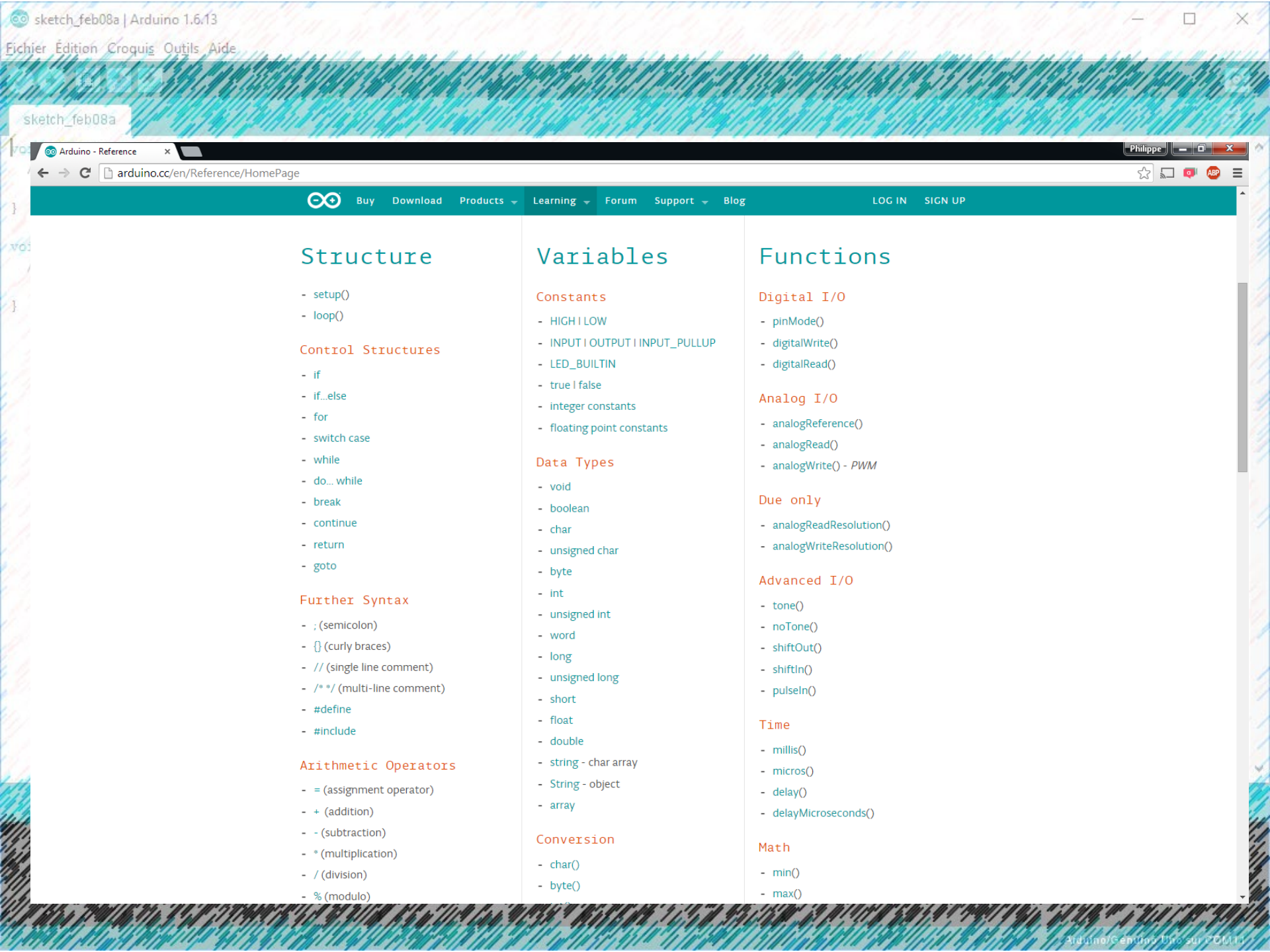
sketch_feb08a

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

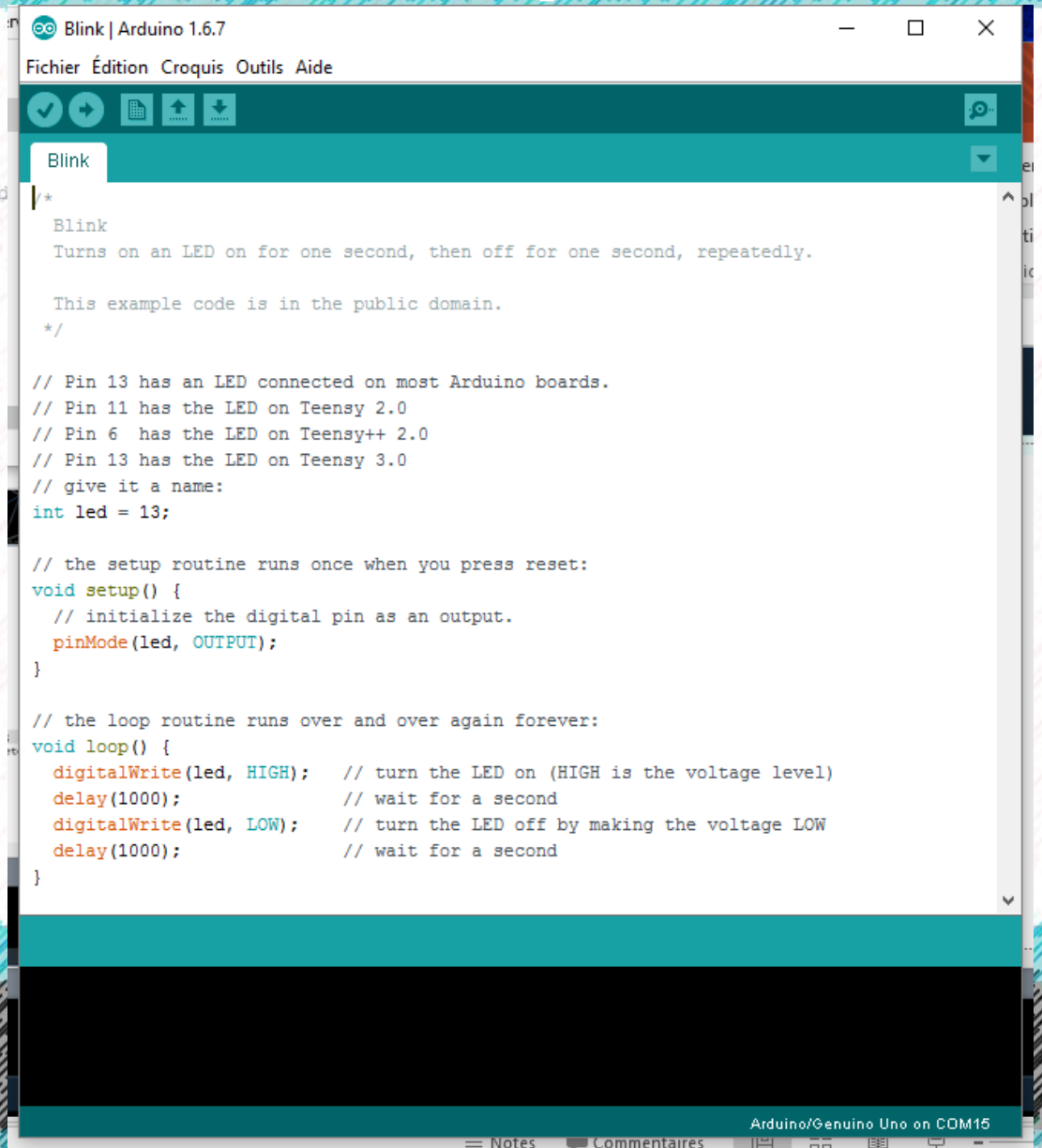
- **setup** : exécuté une seule fois au démarrage – permet d'initialiser les variables du programme

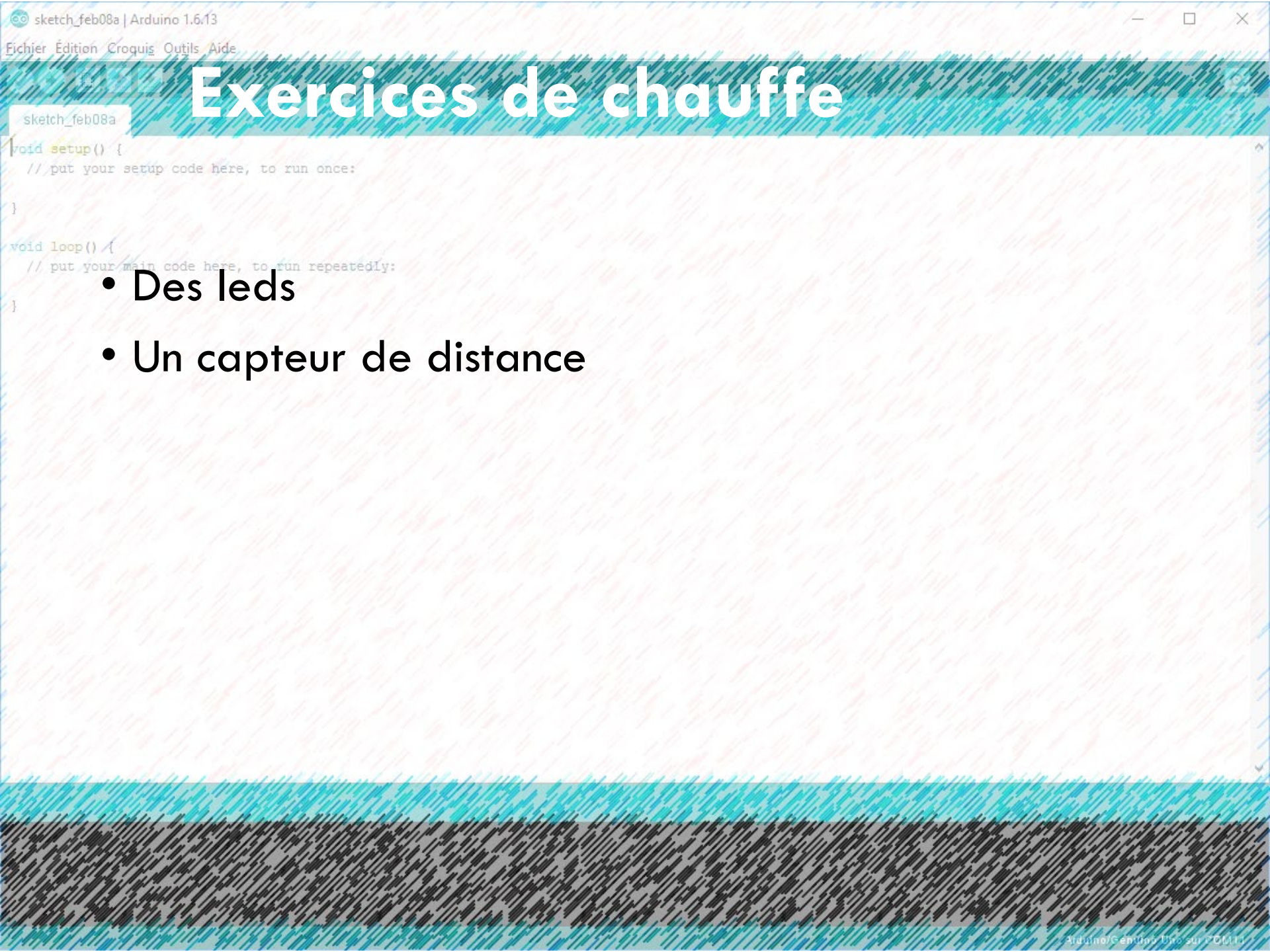
```
void setup() {
  Serial.begin(9600);
  Serial.println("16 channel Servo test!");

  pwm.begin();
  pwm.setPWMFreq(60);  // Analog servos run at ~60 Hz updates
  yield();
}
```

- **loop** : c'est la boucle de traitement des capteurs exécutée « *à l'infini* » (*mainloop*)


sketch_feb08a | Arduino 1.6.13
Fichier Édition Croquis Outils Aide
sketch_feb08a
Arduino - Reference
Philippe
arduino.cc/en/Reference/HomePage
Buy Download Products Learning Forum Support Blog LOG IN SIGN UP
Structure
setup()
loop()
Control Structures
if
if...else
for
switch case
while
do... while
break
continue
return
goto
Further Syntax
; (semicolon)
{} (curly braces)
// (single line comment)
/* */ (multi-line comment)
#define
#include
Arithmetic Operators
= (assignment operator)
+ (addition)
- (subtraction)
* (multiplication)
/ (division)
% (modulo)
Variables
Constants
HIGH | LOW
INPUT | OUTPUT | INPUT_PULLUP
LED_BUILTIN
true | false
integer constants
floating point constants
Data Types
void
boolean
char
unsigned char
byte
int
unsigned int
word
long
unsigned long
short
float
double
string - char array
String - object
array
Conversion
char()
byte()
Functions
Digital I/O
pinMode()
digitalWrite()
digitalRead()
Analog I/O
analogReference()
analogRead()
analogWrite() - PWM
Due only
analogReadResolution()
analogWriteResolution()
Advanced I/O
tone()
noTone()
shiftOut()
shiftIn()
pulseIn()
Time
millis()
micros()
delay()
delayMicroseconds()
Math
min()
max()

Un premier exemple



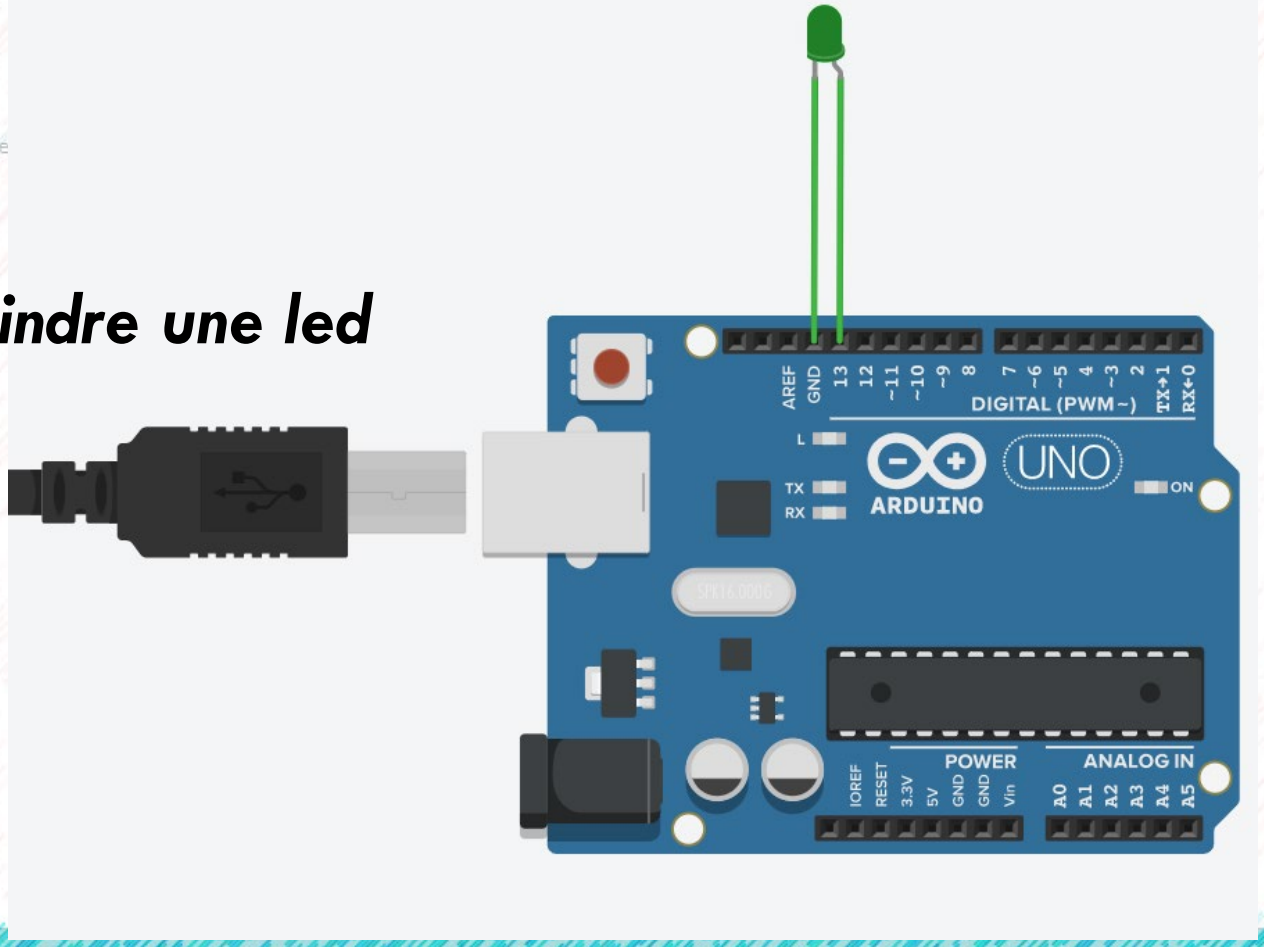


Exercices de chauffe

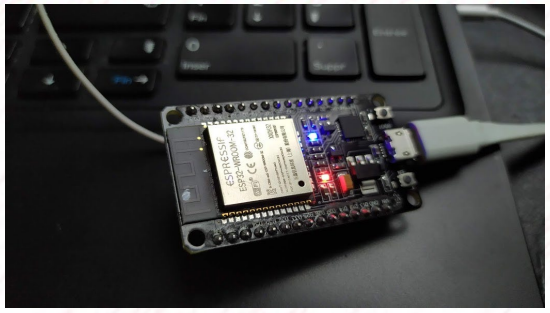
- Des leds
- Un capteur de distance

Exercices de démarrage

- ***Allumer/Eteindre une led***



Exercices de démarrage



- ***Allumer/Eteindre une led***

Ouvrir **Fichier | Exemples | 01.Basics | Blink**

LED_BUILTIN → Pin 13 sur l'Arduino UNO

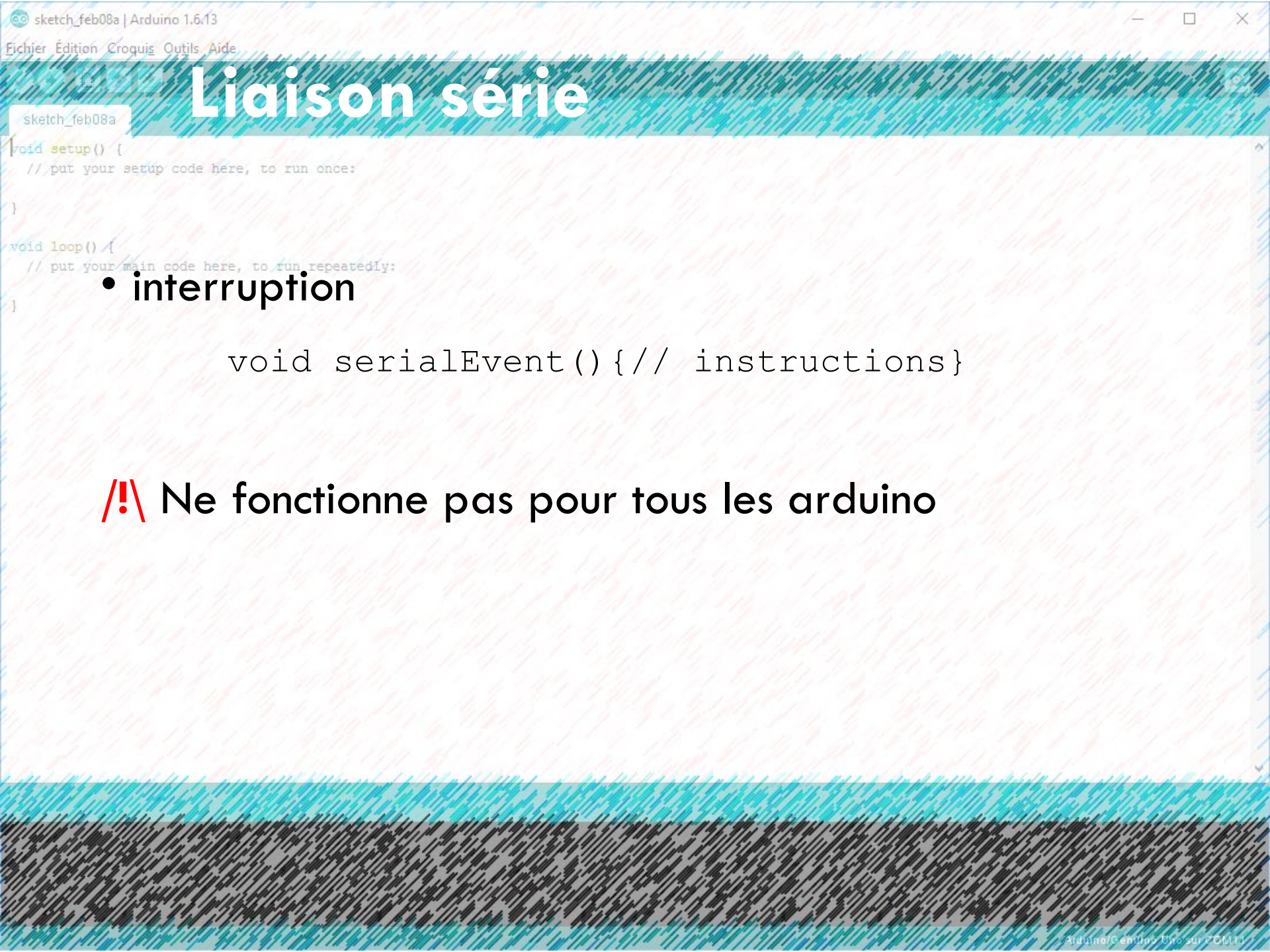
→ Modifier la durée du clignotement

Exercices de démarrage

→ Modifier le programme et envoyer l'état de la LED sur la liaison série

Serial.begin(rapidite_modulation)
Serial.println()

→ Modifier le programme pour piloter l'état de la LED depuis le PC



Liaison série

- interruption

```
void serialEvent(){// instructions}
```

/!\ Ne fonctionne pas pour tous les arduino

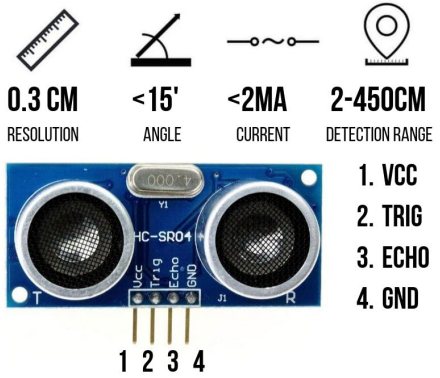
Exercices de démarrage

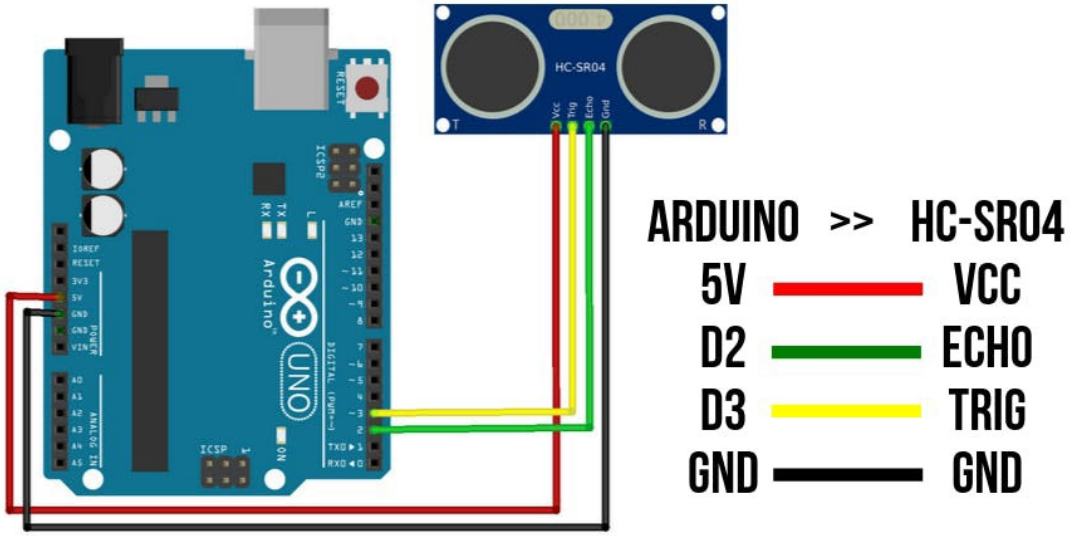
- Un capteur de distance
- Utiliser une librairie externe → capteur ultrason **HC-SR04**
  - https://bitbucket.org/teckel12/arduino-new-ping/downloads/NewPing_v1.9.7.zip (ou via la bibliothèque)

Pour être utilisable sur ESP32, il faut le modèle **HC-SR04P** ou modifier le capteur (https://www.instructables.com/Modify-Ultrasonic-Sensors-for-3-Volts-Logic-prepar/)

Exercices de démarrage

- Un capteur de distance

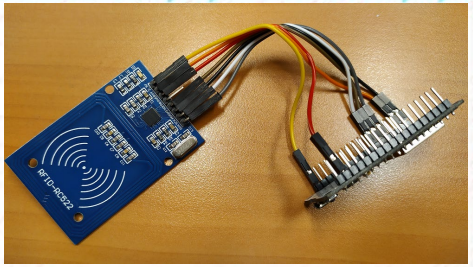




Exercices de démarrage

→ Écrire un programme qui envoie sur le port série la distance perçue par l'arduino avec le plus proche objet et allume la led **LED_BUILTIN** si la distance est inférieure à 20 cm

Exercices de démarrage



- **RFID-522** – Un lecteur NFC

MFRC522

by **GithubCommunity** Version **1.4.7** **INSTALLED**

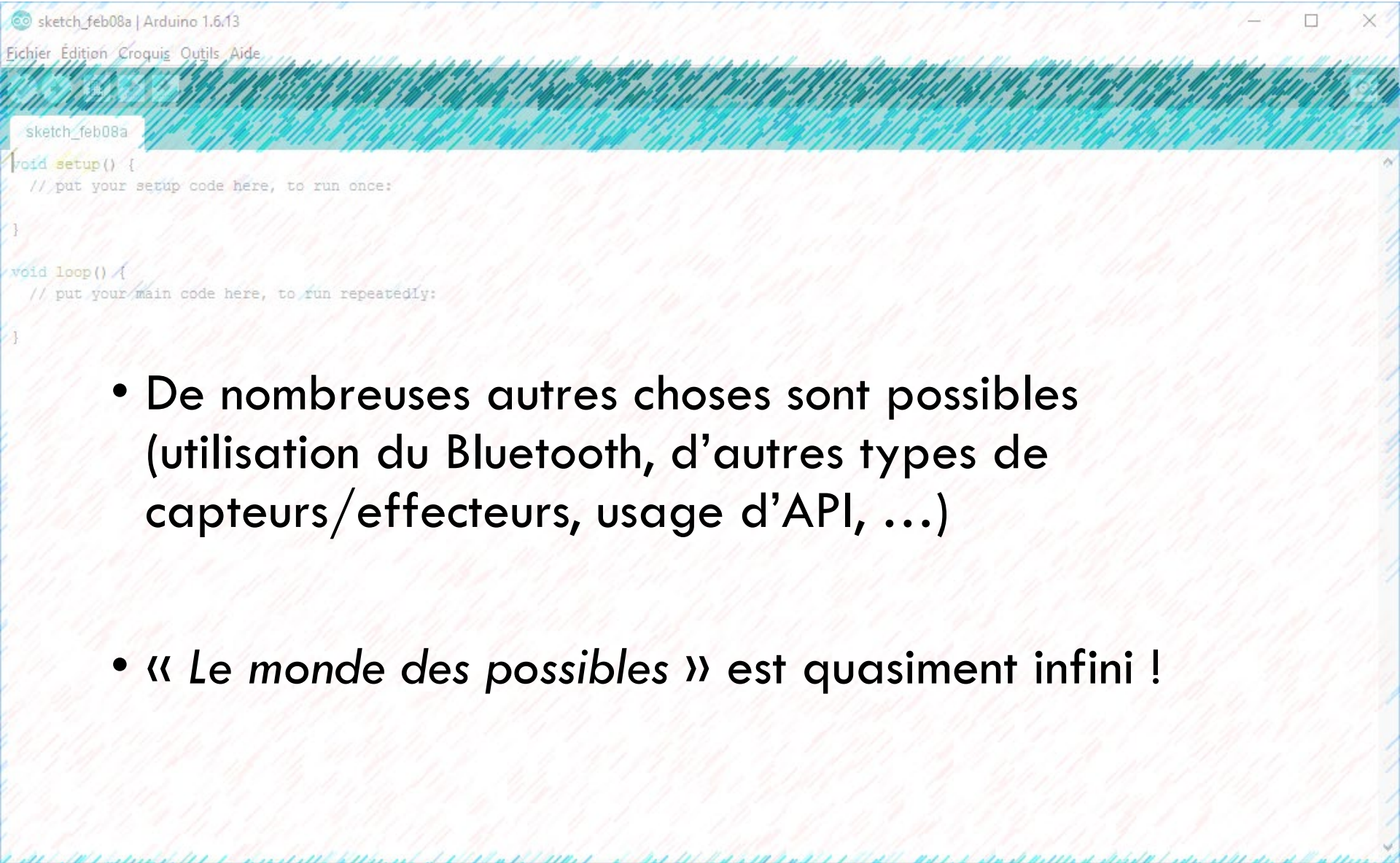
Arduino RFID Library for MFRC522 (SPI) Read/Write a RFID Card or Tag using the ISO/IEC 14443A/MIFARE interface.

More info

Sélectionner une version | Installer

SDA	GPIO21
SCK	GPIO18
MOSI	GPIO23
MISO	GPIO19
IRQ	NOT USED
GND	GND
RST	GPIO22
3v3	3v3

→ Modifier le code fourni qui permet d'allumer/éteindre une LED quand on présente une carte NFC spécifique



- De nombreuses autres choses sont possibles (utilisation du Bluetooth, d'autres types de capteurs/effecteurs, usage d'API, …)

- « *Le monde des possibles* » est quasiment infini !

Méthode de programmation

Pour programmer, nous aborderons la programmation sous la forme de Machine à Etats

- Un Etat → du code à exécuter
- Des événements (changement de valeur de capteurs, réception de données, …) nous ferons changer d'état

La boucle **loop** ne contiendra qu'une instruction switch/case

Machine à états

```
typedef enum {ETEINT=LOW, ALLUME=HIGH} MAE;
// l'énumération est définie sous le type MAE
MAE mae;
...

void loop(){
  switch(mae) {
    case ALLUME: ...
      break;
      ...
```

Machine à états

```
typedef enum {ETEINT=LOW, ALLUME=HIGH} MAE; //
l'énumération est définie sous le type MAE

MAE mae;


…
void loop(){
  switch(mae) {
    case ALLUME: ...
      break;
      ...
```

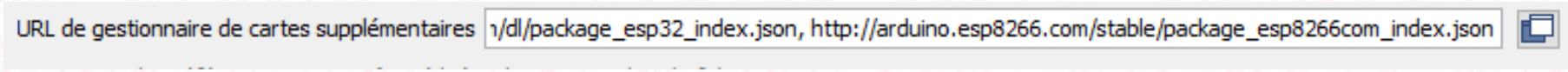
Les ESP

- Les ESP (8266 et 32) sont une série de micro-contrôleurs intégrant la gestion du wifi et du bluetooth (jusqu'à BLE)

- Ils sont peu onéreux et très appréciés dans le domaine de l'IoT !

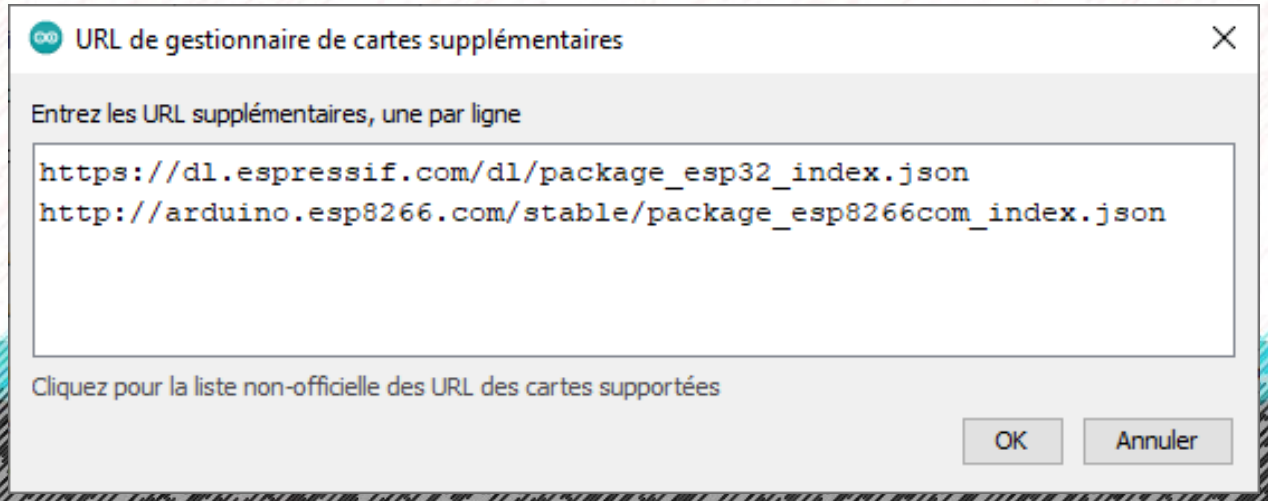
Installer ESP8266 et ESP32

- Dans `Fichier|Préférences`



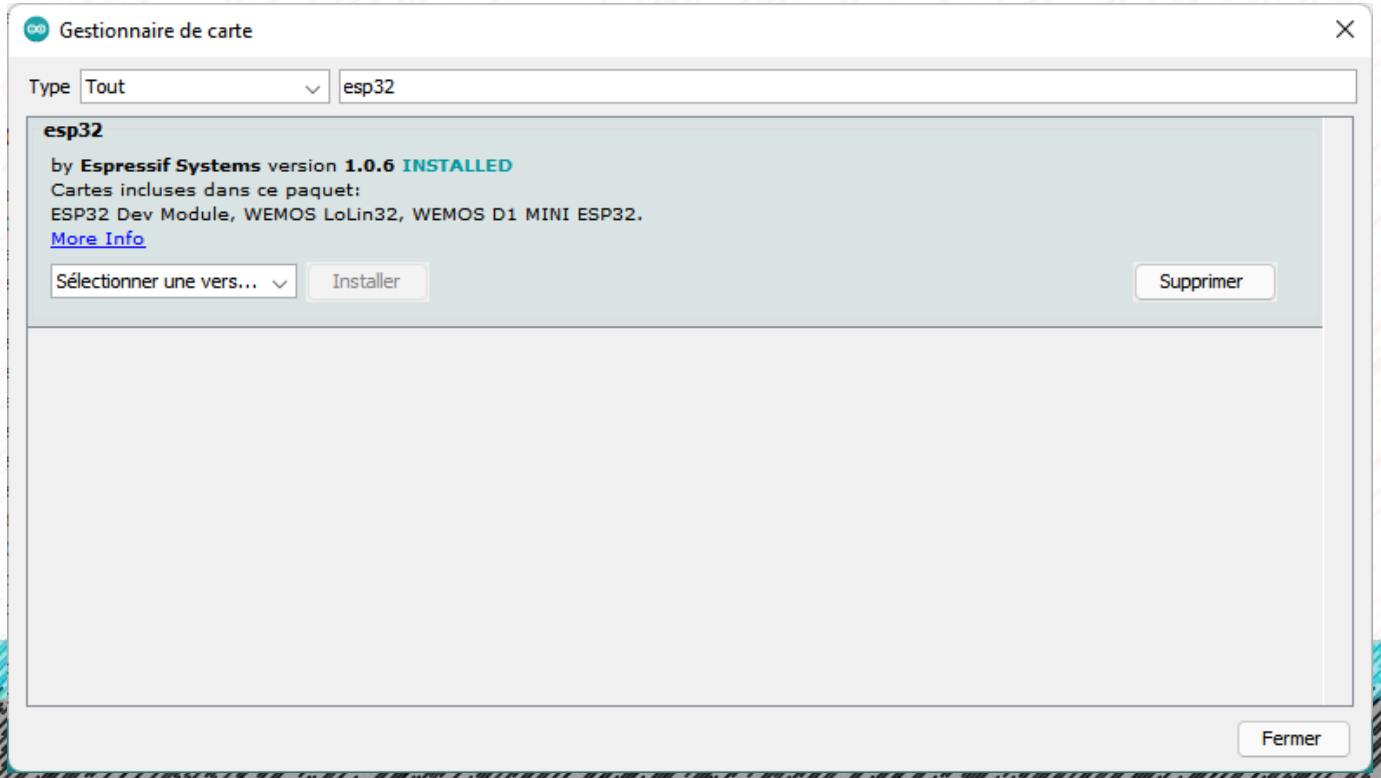
`https://dl.espressif.com/dl/package_esp32_index.json`

`http://arduino.esp8266.com/stable/package_esp8266com_index.json`



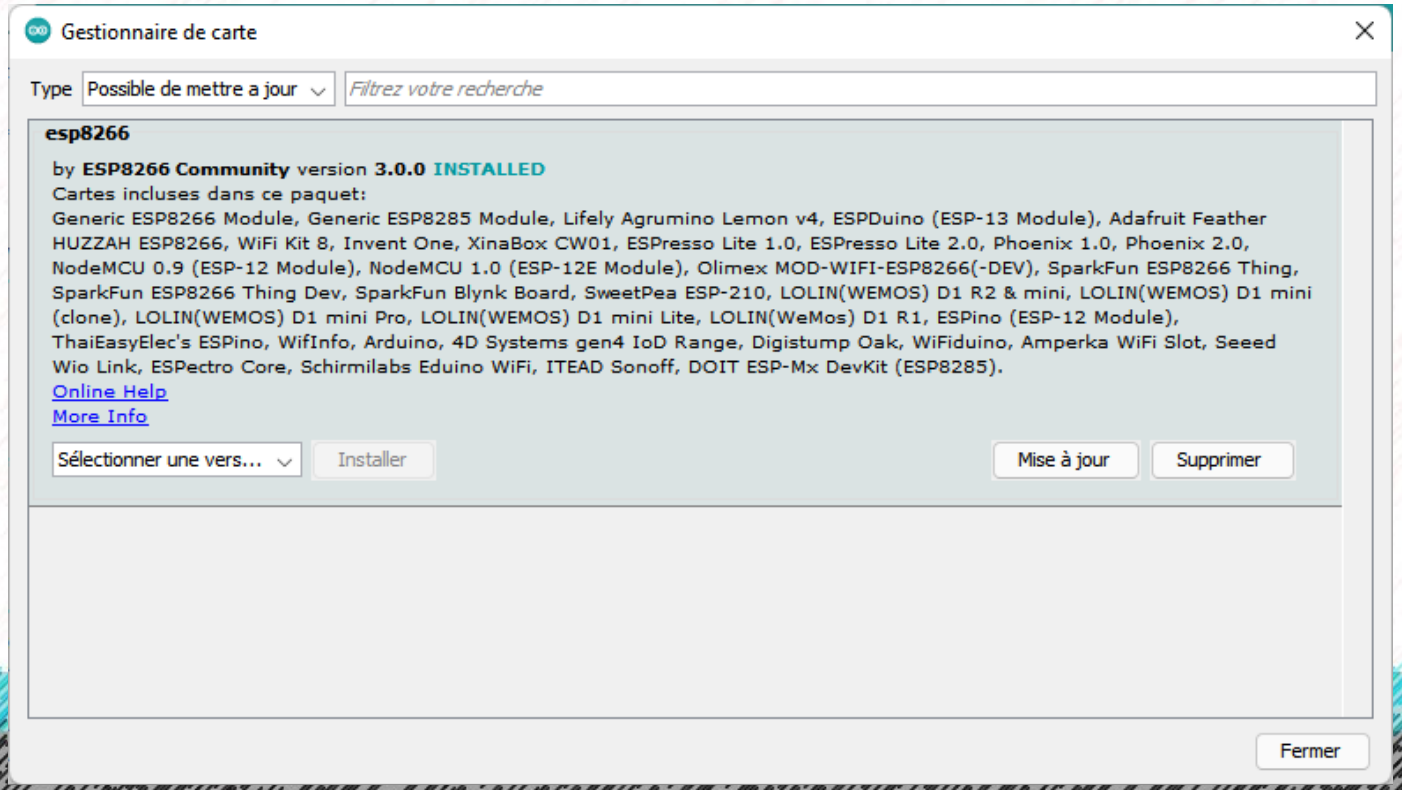
Installer ESP8266 et ESP32

- Dans `Outils |Type de carte |Gestionnaire de carte`

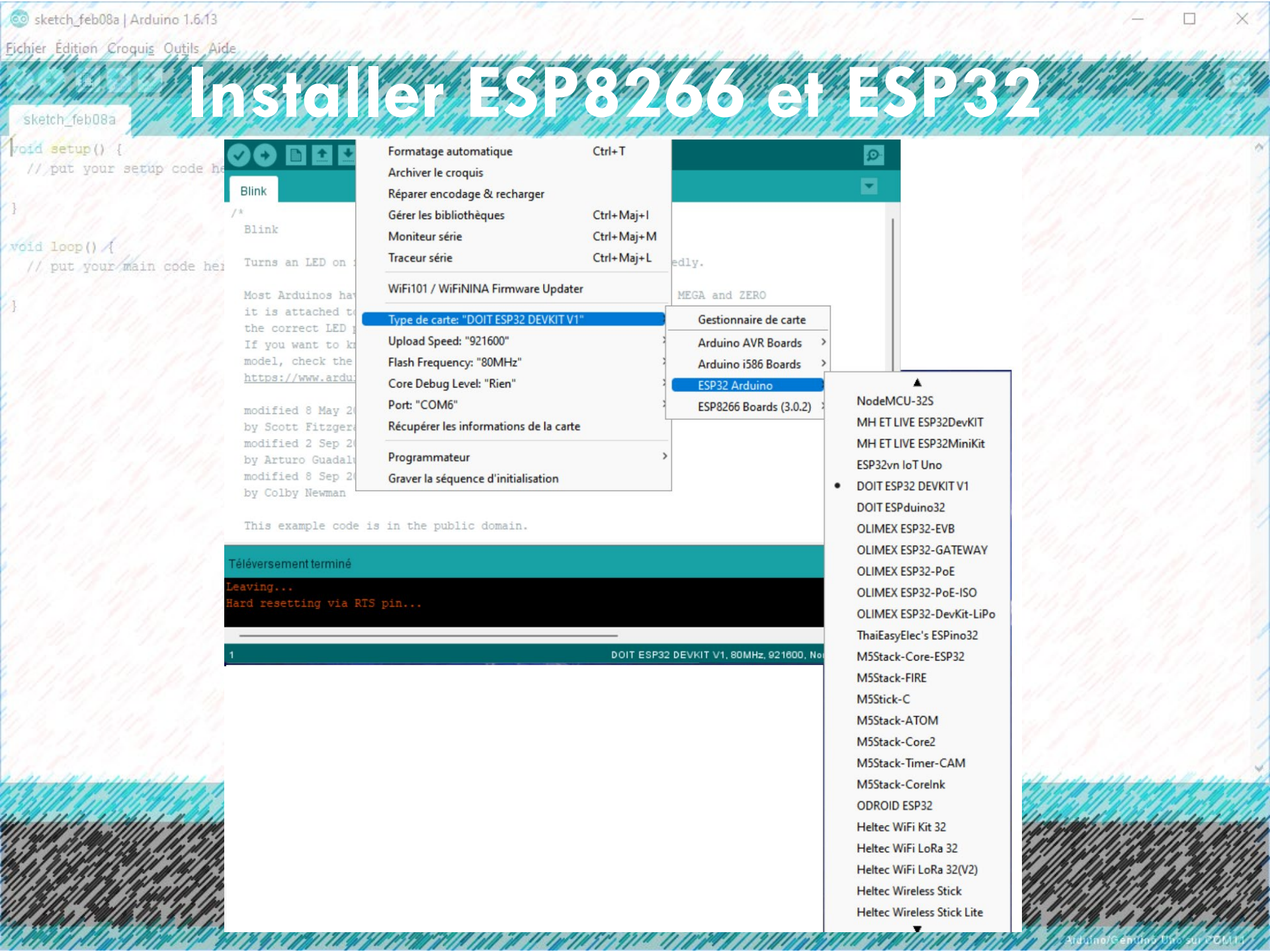


Installer ESP8266 et ESP32

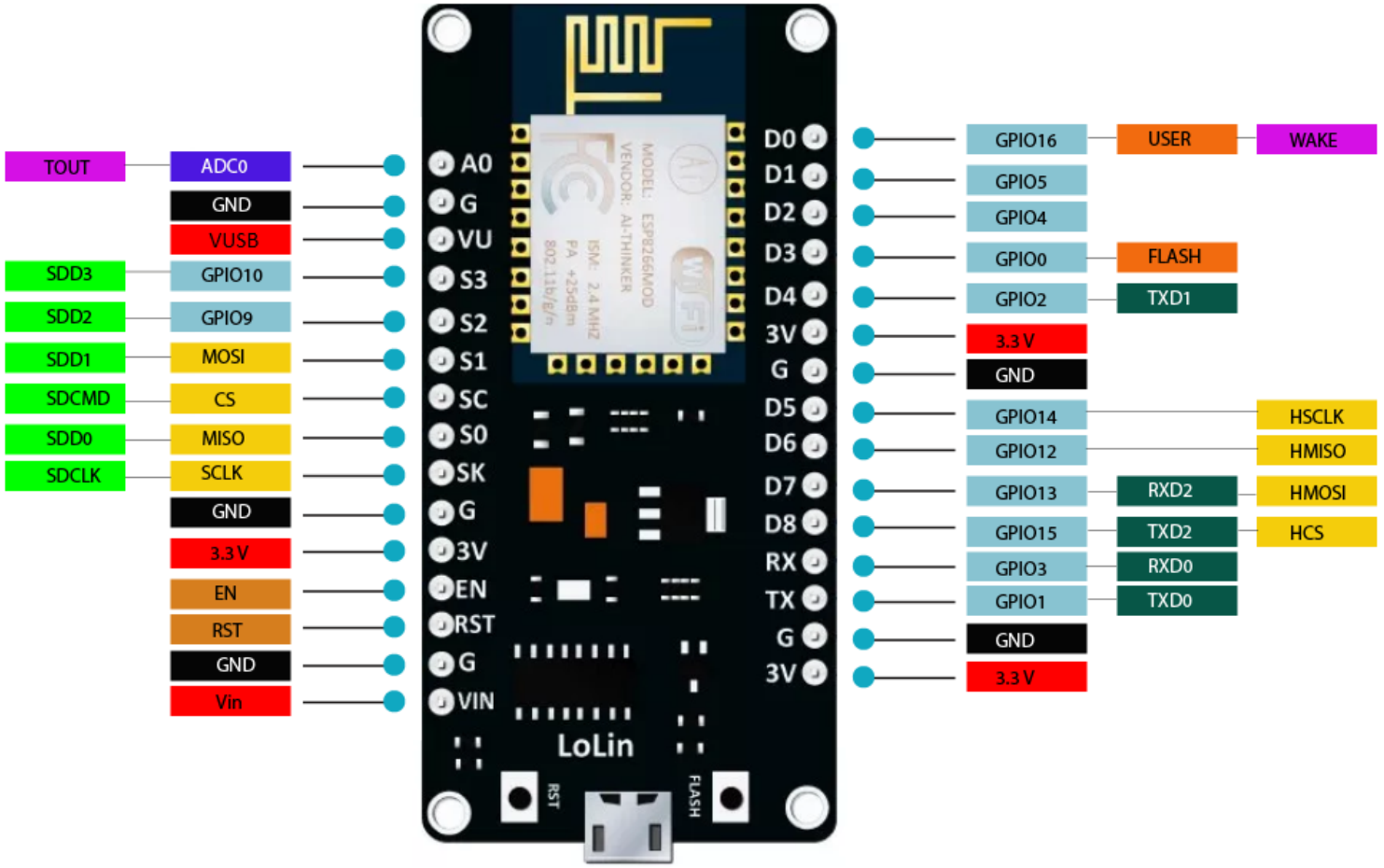
- Dans `Outils |Type de carte |Gestionnaire de carte`



Installer ESP8266 et ESP32



ESP8266 pinout



NodeMCU V3 Pinout

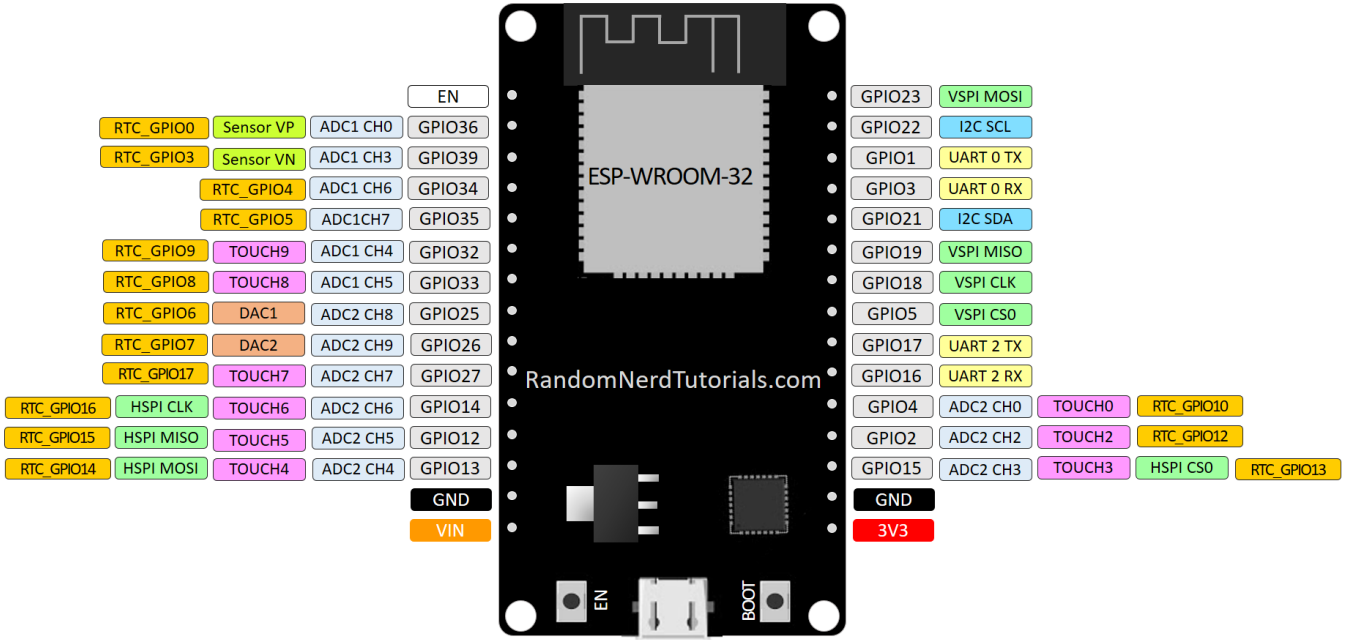
www.TheEngineeringProjects.com

D0	GPIO16
D1	GPIO05
D2	GPIO04
D3	GPIO00
D4	GPIO02
D5	GPIO14
D6	GPIO12
D7	GPIO13
D8	GPIO15
D9	GPIO03
D10	GPIO01

ESP32 pinout

ESP32 DEVKIT V1 – DOIT

version with 30 GPIOs



ESP32 pinout