

# Présentation de l'Arduino

ADREF13  
2023

# Plan de l'exposé

- Un Arduino, c'est quoi ?
- On peut faire quoi avec ?
- Qu'est ce qu' on ne peut pas faire avec ?
- Les caractéristiques
- Les différents modèles
- Les ressources
- Le logiciel de développement intégré: IDE
- Le langage de programmation
- Traitons un exemple simple : Blink
- Les Entrées / Sorties
- Travaux Pratiques
  - BME 280
  - Servo
  - Internet of Things
- Les Périphériques

# Un Arduino c'est quoi ?

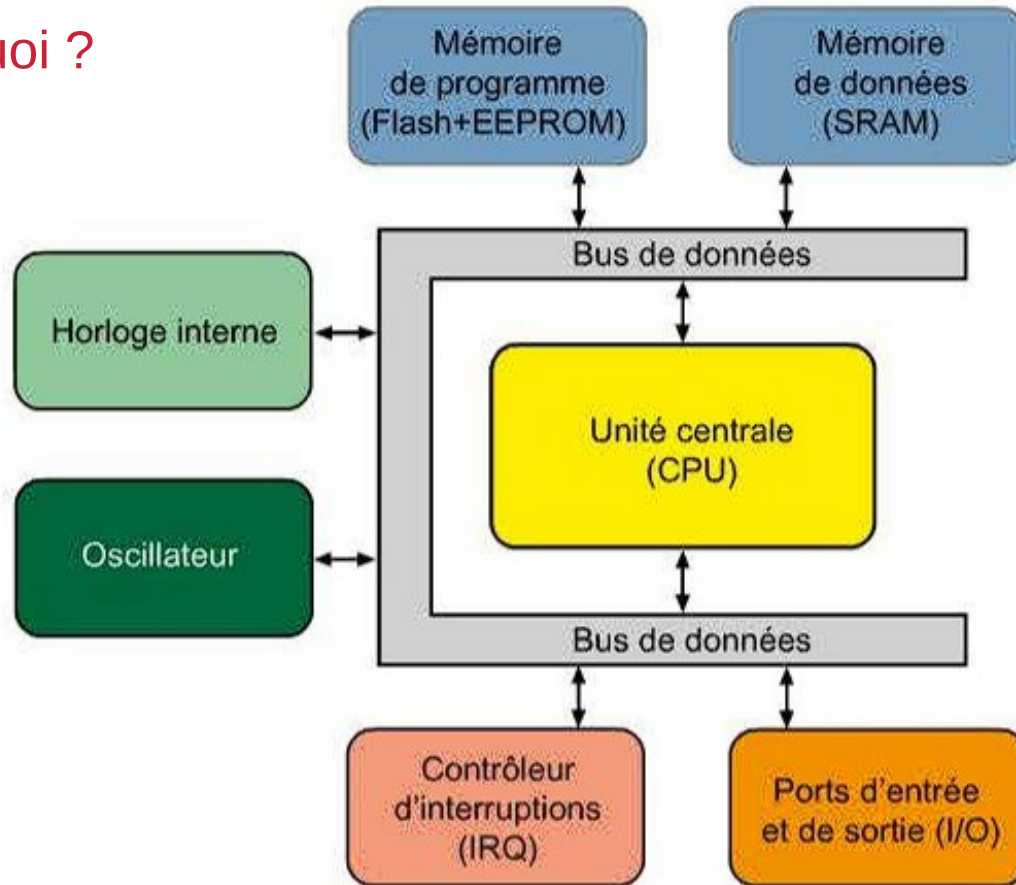
L'Arduino est une carte électronique basée sur un micro contrôleur ATMEL AtMega 328.

Il embarque :

- un système d'alimentation,
- un processeur,
- son horloge,
- un interface USB,
- de la mémoire programme, travail
- des ports d' Entrées Sorties,
- la connectique nécessaire.



## Un Arduino c'est quoi ?



## On peut faire quoi avec ?

- L'Arduino peut stocker et exécuter un programme,
- Il fait l'acquisition des ports d'entrées,
- effectue les traitements,
- écrit les sorties,
- Il peut travailler sur des signaux logiques ou analogiques,
- Il peut communiquer avec d'autres équipements par différents protocoles.

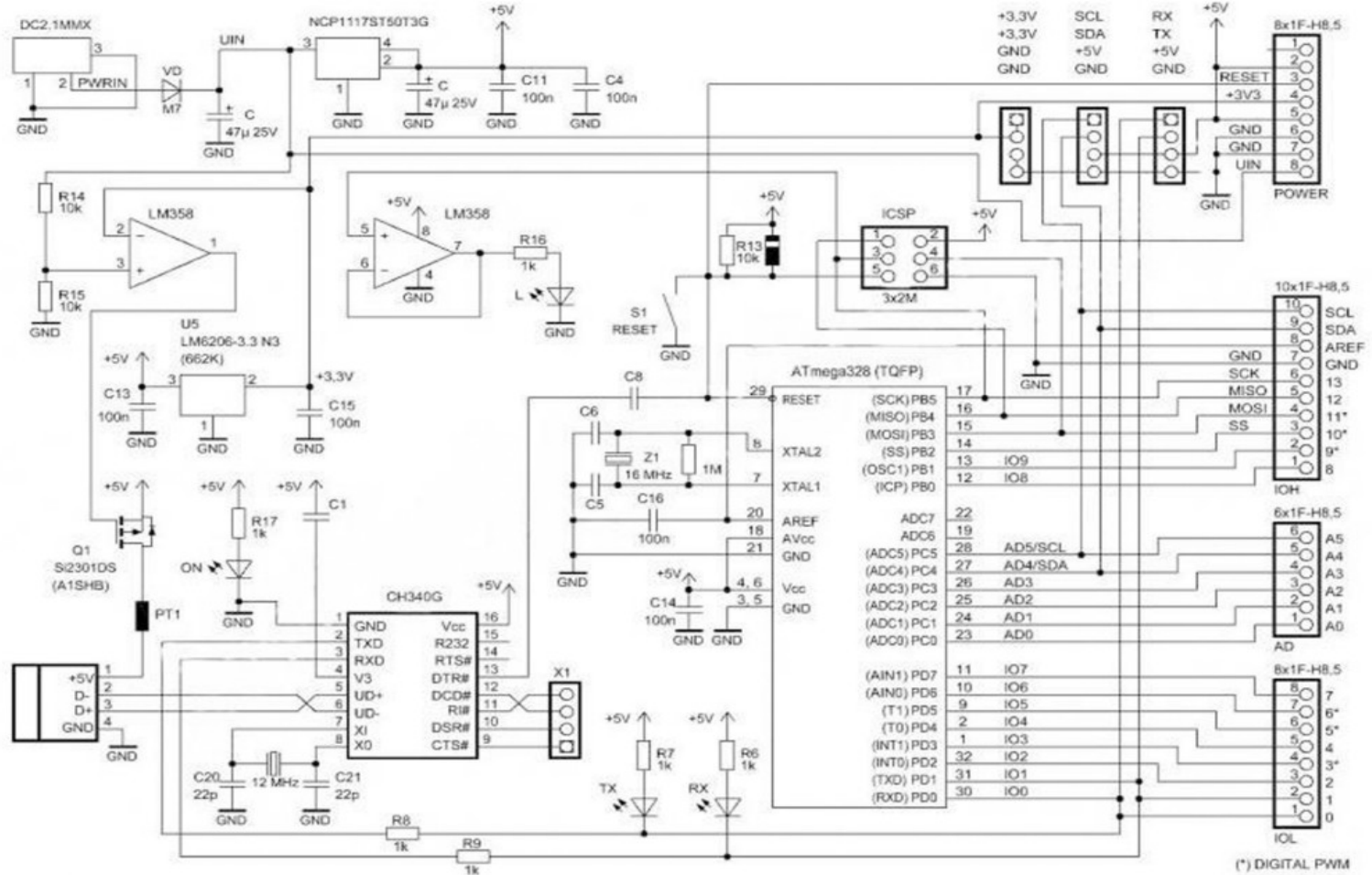
## Qu'est ce qu' on ne peut pas faire avec ?

Il ne peut pas :

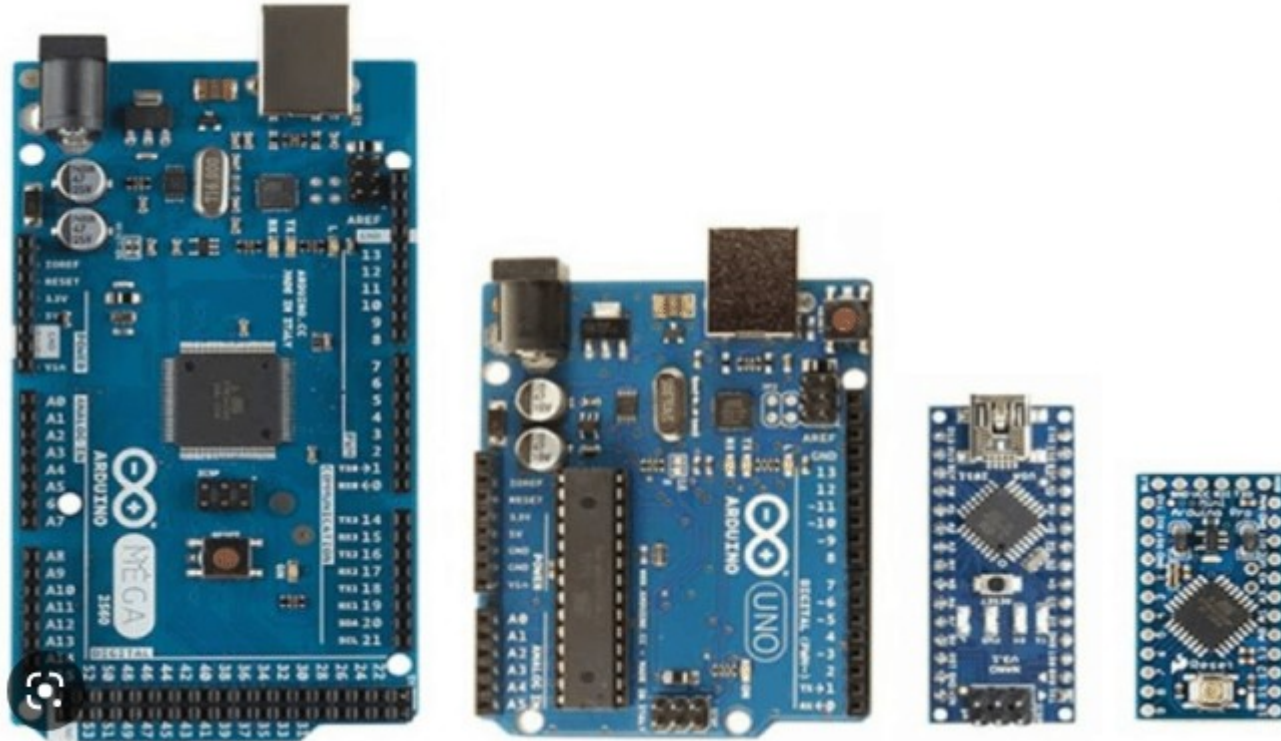
- recevoir un OS type Windows ou Linux,
- recevoir un Disque Dur,
- recevoir de mémoire supplémentaire
- recevoir d'extensions d'IO.

# Les caractéristiques

- Alimentation:
  - via port USB ou
  - 7 à 12 V connecteur alim 5,5x2,1 mm
- Microcontrôleur: ATmega328
- Mémoire flash: 32 kB
- Mémoire SRAM: 2 kB
- Mémoire EEPROM: 1 kB
- Interfaces:
  - 14 broches d'E/S dont 6 PWM
  - 6 entrées analogiques 10 bits
  - Bus série, I2C et SPI
- Intensité par E/S: 40 mA
- Cadencement: 16 MHz
- Gestion des interruptions
- Connecteur USB B



## Les différents modèles





## Les différents modèles

Arduino Uno




Arduino Mega 2560



Arduino Micro



			
Price Points	\$19.99-\$23.00	\$36.61 - \$39.00	\$19.80 - \$24.38
Dimension	2.7 in x 2.1 in	4 in x 2.1 in	0.7 in x 1.9 in
Processor	Atmega328P	ATmega2560	ATmega32U4
Clock Speed	16MHz	16MHz	16MHz
Flash Memory (kB)	32	256	32
EEPROM (kB)	1	4	1
SRAM (kB)	2	8	2.5
Voltage Level	5V	5V	5V
Digital I/O Pins	14	54	20
Digital I/O with PWM Pins	6	15	7
Analog Pins	6	16	12
USB Connectivity	Standard A/B USB	Standard A/B USB	Micro-USB
Shield Compatibility	Yes	Yes	No
Ethernet/Wi-Fi/Bluetooth	No (a Shield/module can enable it)	No (a Shield/module can enable it)	No

## Les différents modèles

Characteristics	Arduino UNO	Arduino Nano	Arduino Mega	Arduino Due
Microcontroller	ATmega328	ATmega328	ATmega1280	AT91SAM3X8E
Clock frequency	16 MHz	16 MHz	16 MHz	84 MHz
Analog inputs	8	8	16	12
Analog output	0	0	0	2
Digital input/output	22	22	54	54
Analog signals range	0 to 5 V	0 to 5 V	0 to 5 V	0 to 3.3 V
Analog signals resolution	10 bit/4.88 mV	10 bit/4.88 mV	10 bit/4.88 mV	12 bit/0.806 mV

## Les ressources

- L'essentiel des ressources est sur le site : <https://www.arduino.cc/>
- Il est conseillé d'y créer un compte (gratuit).
- On y trouve tout sur :
  - Le matériel (brochages, caractéristiques),
  - Le software (IDE),
  - Le langage (Rubrique Référence),
  - Des cours,
  - Des exemples de programmes appelés Sketches
  - Et beaucoup plus encore (cloud, forum, projets, librairies, ....)
- Et puis bien sur tout internet avec le meilleur et le pire.

<https://pecquery.wixsite.com/arduino-passion>

## Le logiciel de développement intégré: IDE

Ce logiciel permet de :

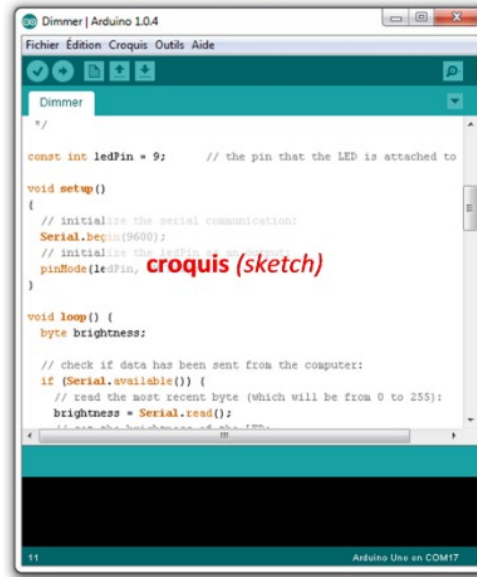
- Éditer les programmes (Sketches),
- Les Compiler,
- Les Téléverser dans l'Arduino cible.
  
- Il y a 2 possibilités :
  - Installer le logiciel IDE sur son PC
  - Ou travailler sur le Web et sauvegarder ses projets sur le Cloud Arduino
- Nous envisagerons le 1er cas.
- L'installation est décrite dans un document à part.

## La mise en œuvre de l'Arduino

L'installation et la mise en œuvre de IDE sur un programme simple est décrite dans le fichier

[Installation IDE.docx](#)

### le logiciel de programmation



```
Dimmer | Arduino 1.0.4
Fichier Édition Croquis Outils Aide

Dimmer
*/

const int ledPin = 9;    // the pin that the LED is attached to

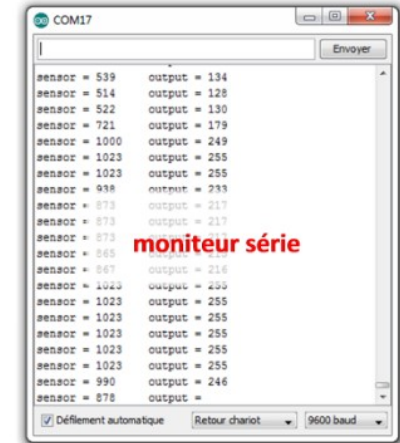
void setup()
{
  // initialize the serial communication:
  Serial.begin(9600);
  // initialize the LED pin as an output:
  pinMode(ledPin, croquis (sketch)
}

void loop() {
  byte brightness;

  // check if data has been sent from the computer:
  if (Serial.available()) {
    // read the most recent byte (which will be from 0 to 255):
    brightness = Serial.read();
  }
}
```

compilation et téléversement

### l'interface utilisateur



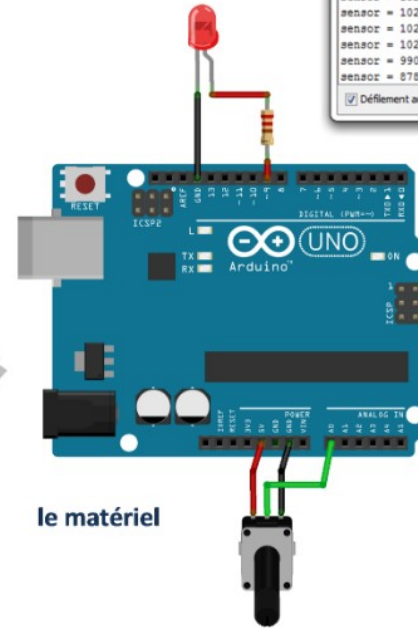
```
COM17
Envoyer

senaeor = 539   output = 134
senaeor = 514   output = 128
senaeor = 522   output = 130
senaeor = 721   output = 179
senaeor = 1000  output = 249
senaeor = 1023  output = 255
senaeor = 1023  output = 255
senaeor = 938   output = 233
senaeor = 873   output = 217
senaeor = 873   output = 217
senaeor = 873   output = 217
senaeor = 865   output = 216
senaeor = 867   output = 216
senaeor = 1023  output = 255
senaeor = 1023  output = 255
senaeor = 1023  output = 255
senaeor = 1023  output = 255
senaeor = 1023  output = 255
senaeor = 1023  output = 255
senaeor = 990   output = 246
senaeor = 878   output =

 Défilement automatique Retour chariot 9600 baud
```

moniteur série

communication

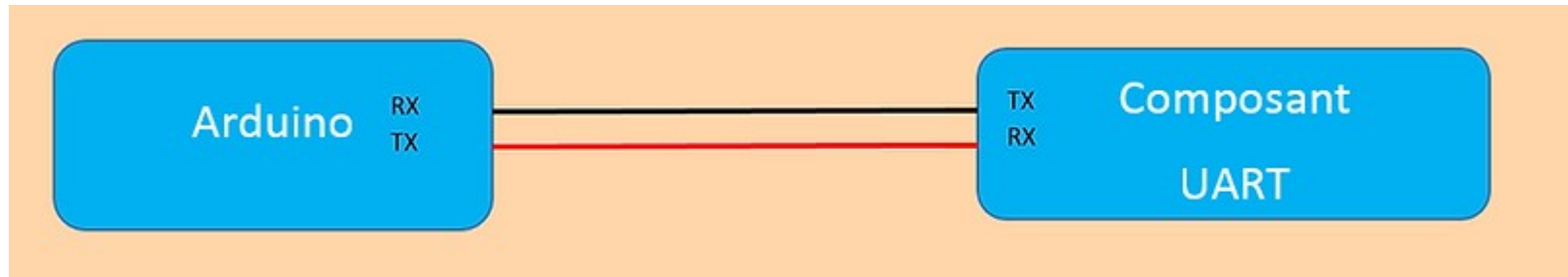
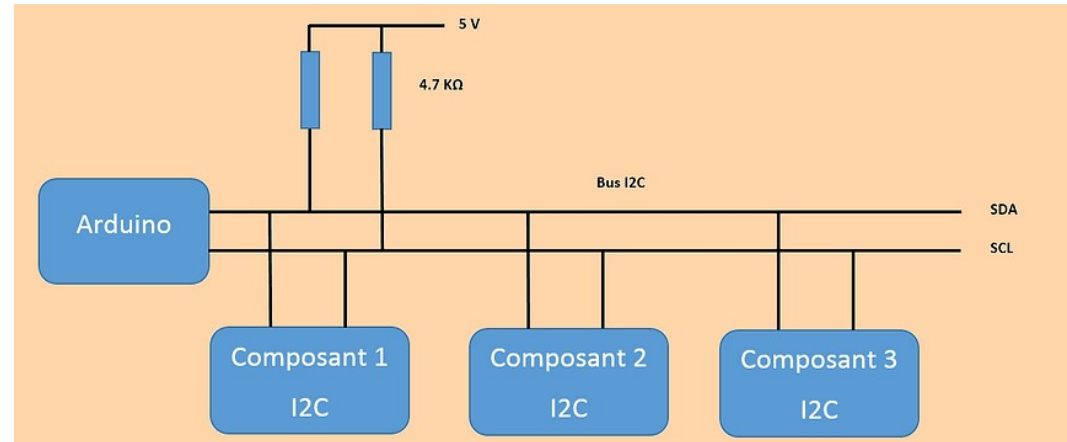
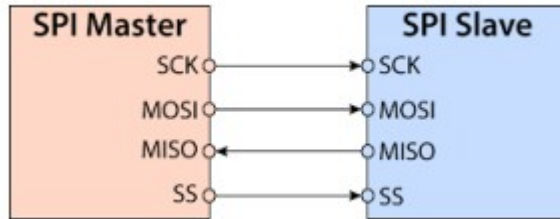


## Le langage de programmation

```
sketch_feb25a.ino
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }
10
```

On peut aller voir sur : <https://www.arduino.cc/reference/en/>

## Les Périphériques : Communication



## Travaux pratiques

- 1 : Gestion d'un servomoteur,
- 2 : Gestion d'un capteur BME280 (P, T, H) avec affichage sur écran OLED.
- 3 : Gestion d'un capteur DHT12 (T, H) avec transmission des valeurs sur Internet (IoT)

