

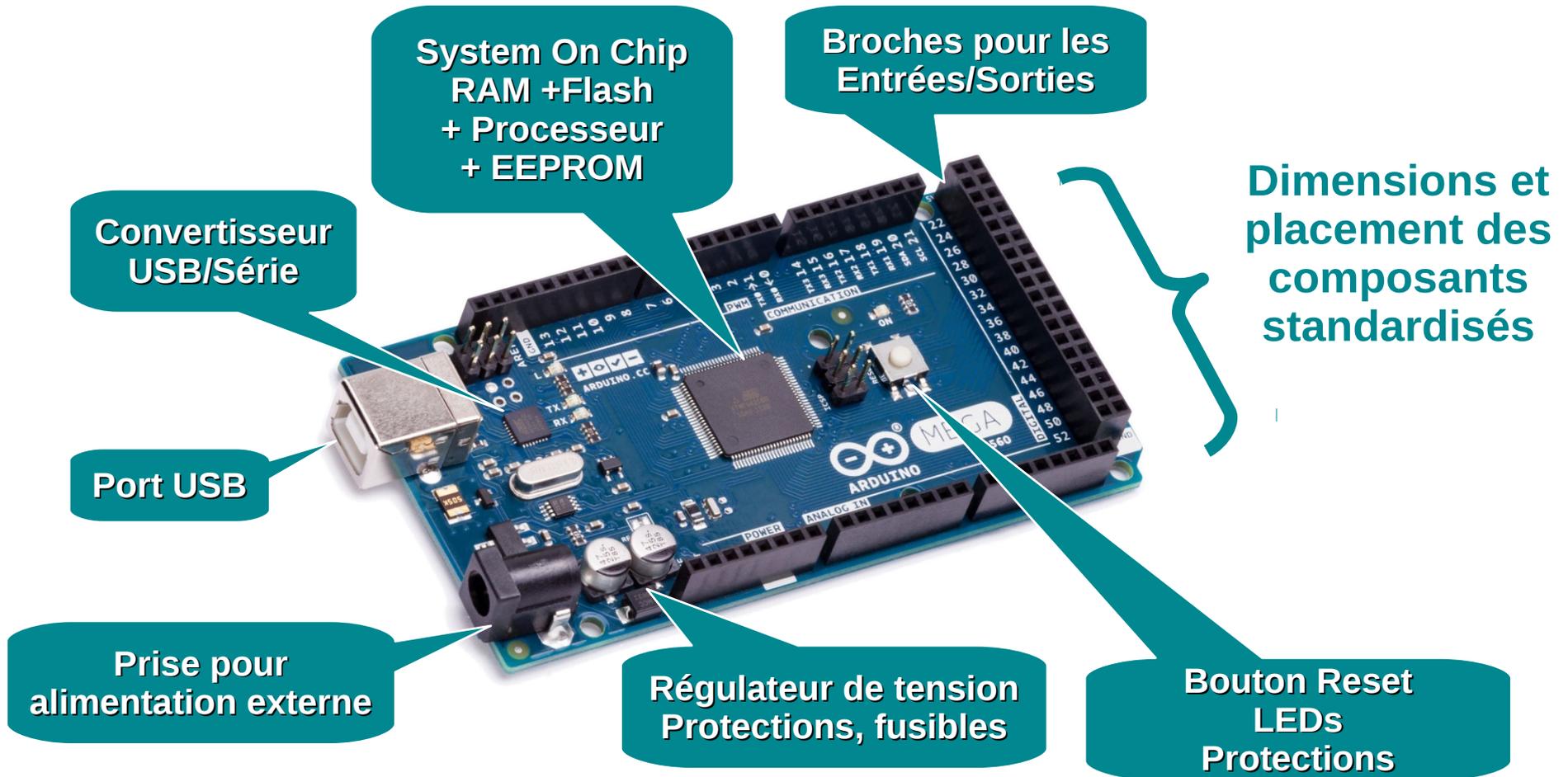
La carte électronique ARDUINO

Journée du Libre
Sarrebourg, le 25 novembre 2017
enunclic-cappel.fr - blog.lkiefer.org

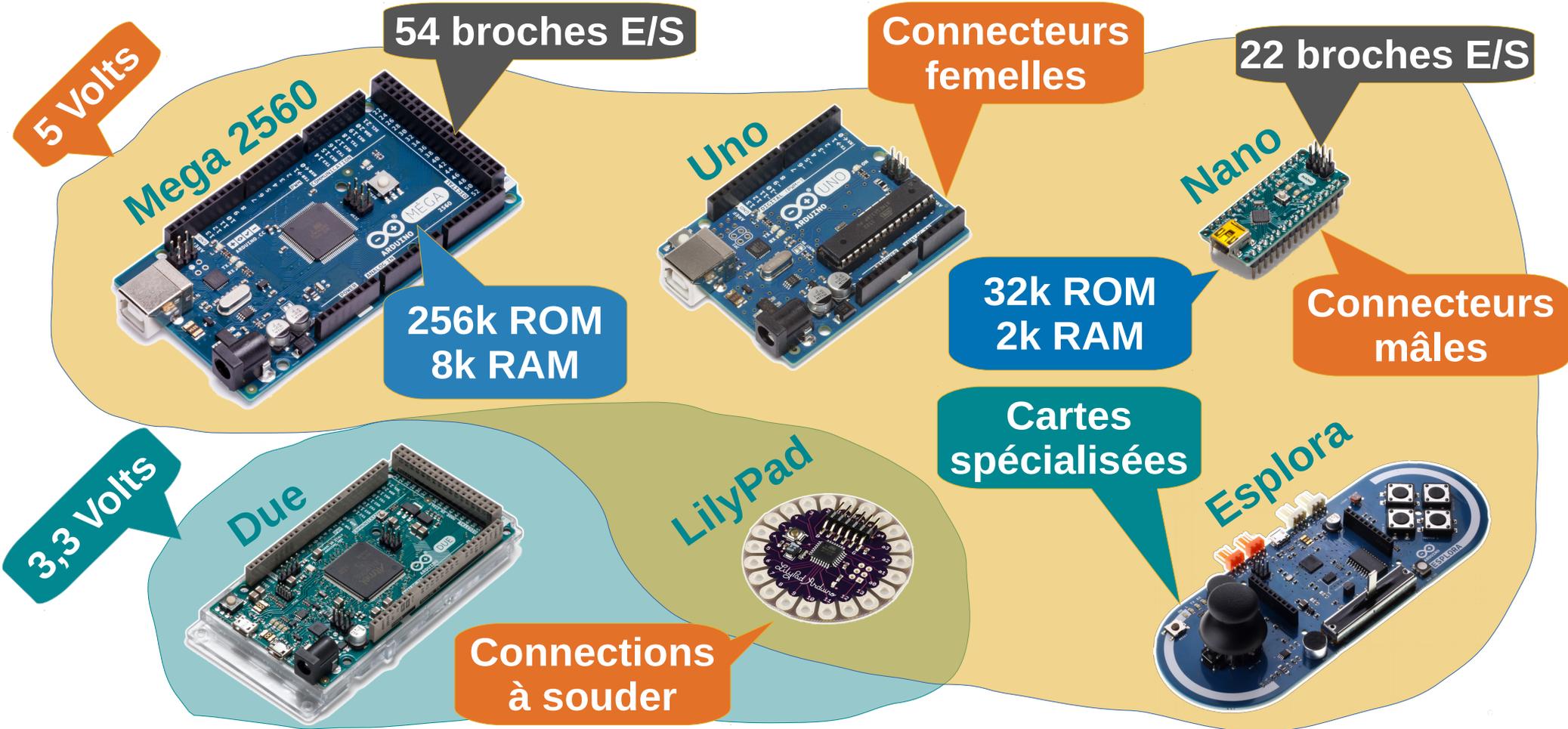
Arduino, c'est quoi?

- Des marques
- Des cartes électroniques 
- Un logiciel 
- Des utilisateurs => Blogs, forums, vidéos
- Un écosystème => Clones, modules, composants
- Des idées, des projets

Une carte Électronique ?

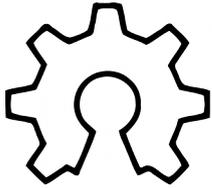


Une collection de cartes

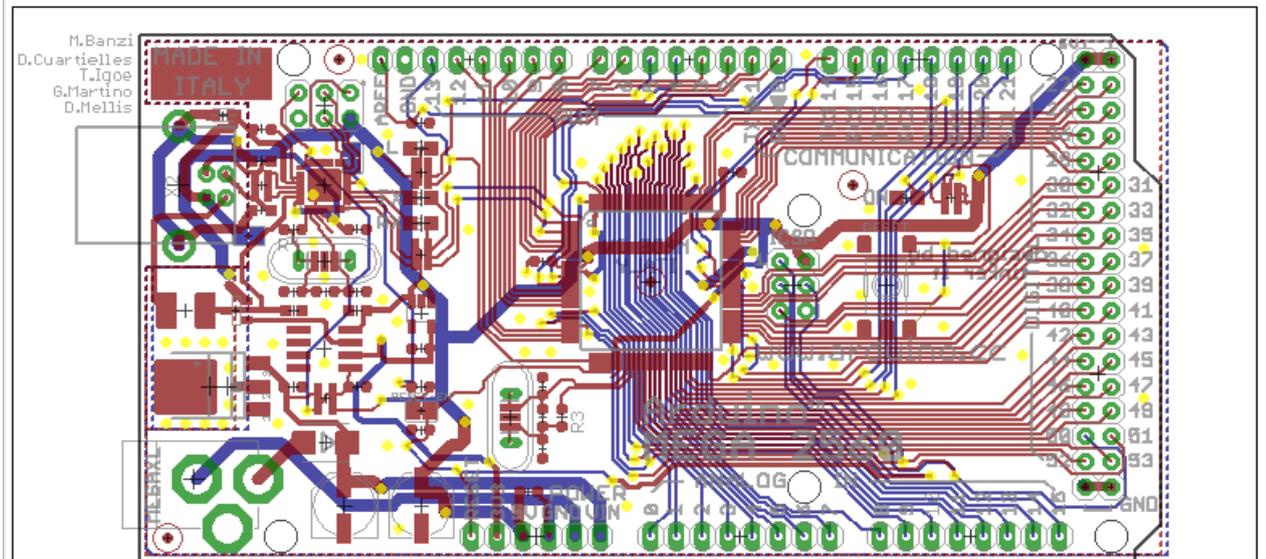
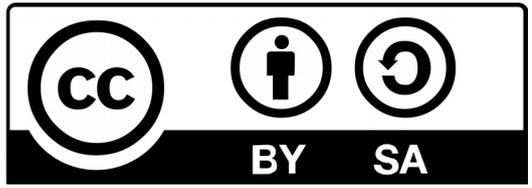


Une carte libre

Open Hardware
Designs sous licence
Creative Commons by-sa



open hardware



Arduino Mega 2560 Reference Design

Reference Designs ARE PROVIDED "AS IS" AND "WITH ALL FAULTS. Arduino DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PU

Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future char The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this info

Size: 4860 x 2827 mil

Une carte libre

Logiciel Libre Arduino IDE GNU General Public Licence



```
Fichier Édition Croquis Outils Aide
[Icons]
Blink
1 //*/
2   Blink
3   Turns on an LED on for one second, then off for
4
5   Most Arduinos have an on-board LED you can control
6   it is attached to digital pin 13, on MKR1000 on
7   the correct LED pin independent of which board it
8   If you want to know what pin the on-board LED is on
9   the Technical Specs of your board at https://www.arduino.cc/en/Tutorial/Blink
10
11   This example code is in the public domain.
12
```

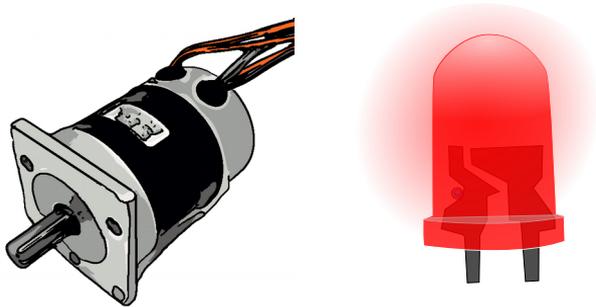
Arduino Nano, ATmega328 sur /dev/ttyUSB0

Pour quoi faire ? La technique :

Sorties Numériques



Sorties PWM



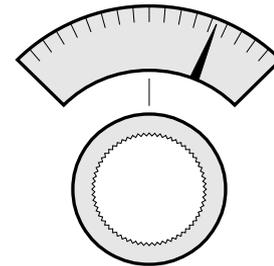
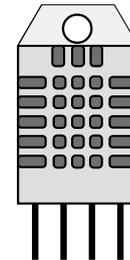
Protocoles de communication



Entrées Numériques



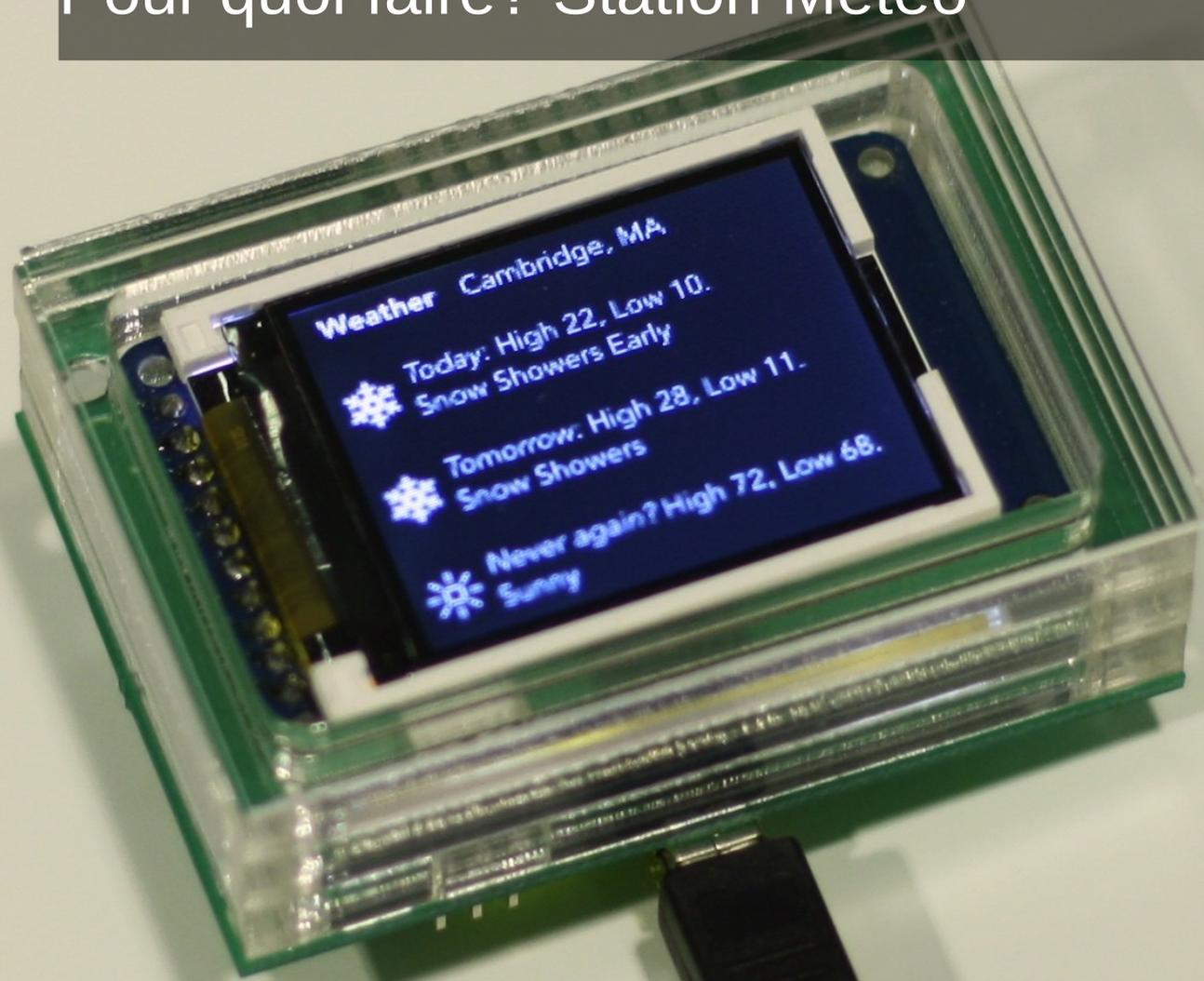
Entrées analogiques



Pour quoi faire? Table LED



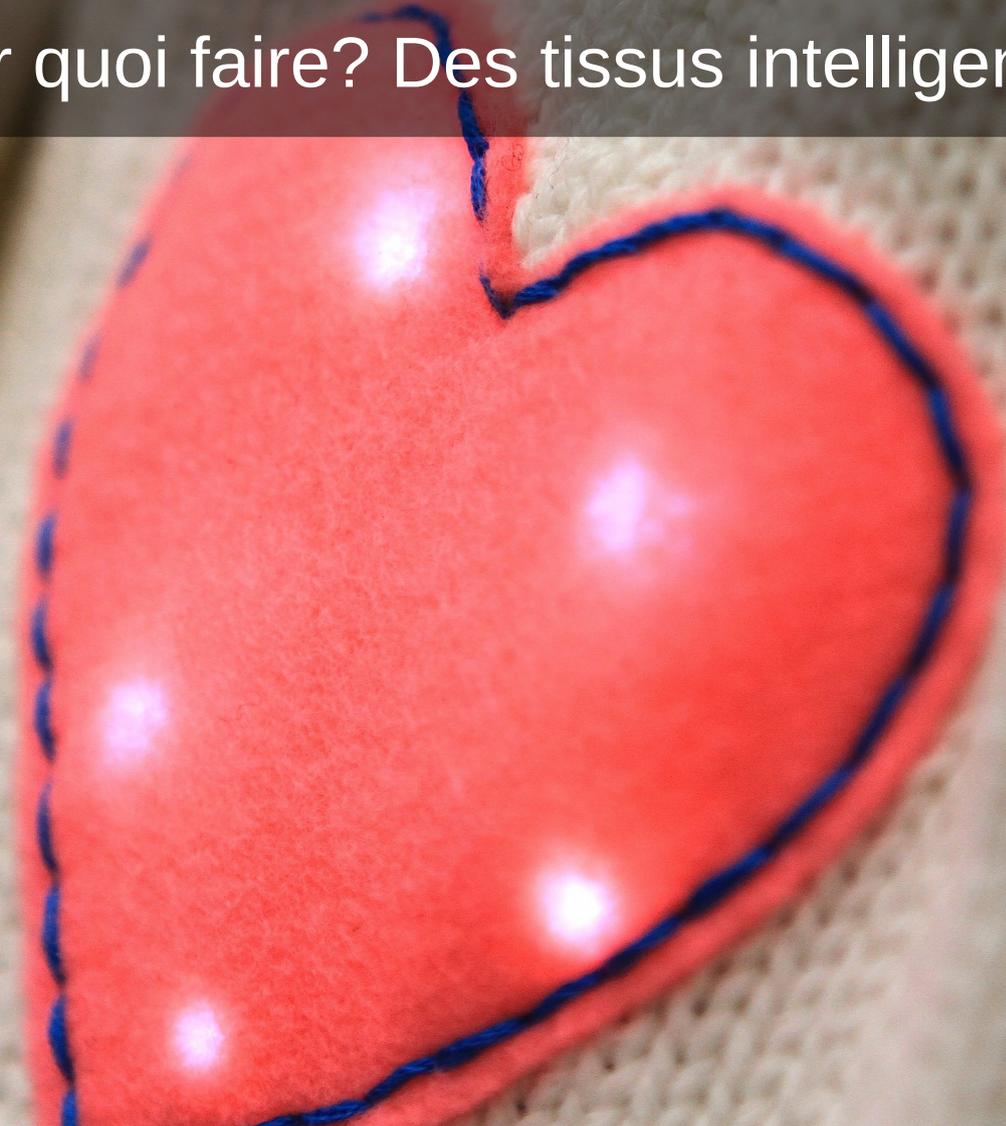
Pour quoi faire? Station Météo



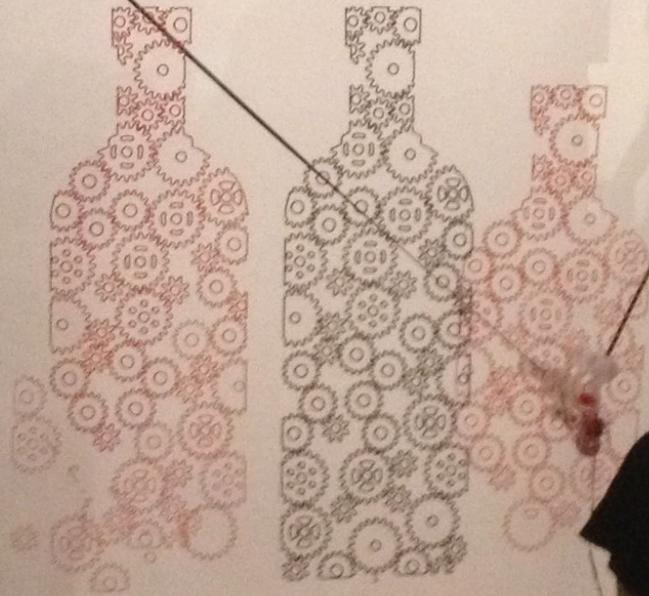
Pour quoi faire? De la musique



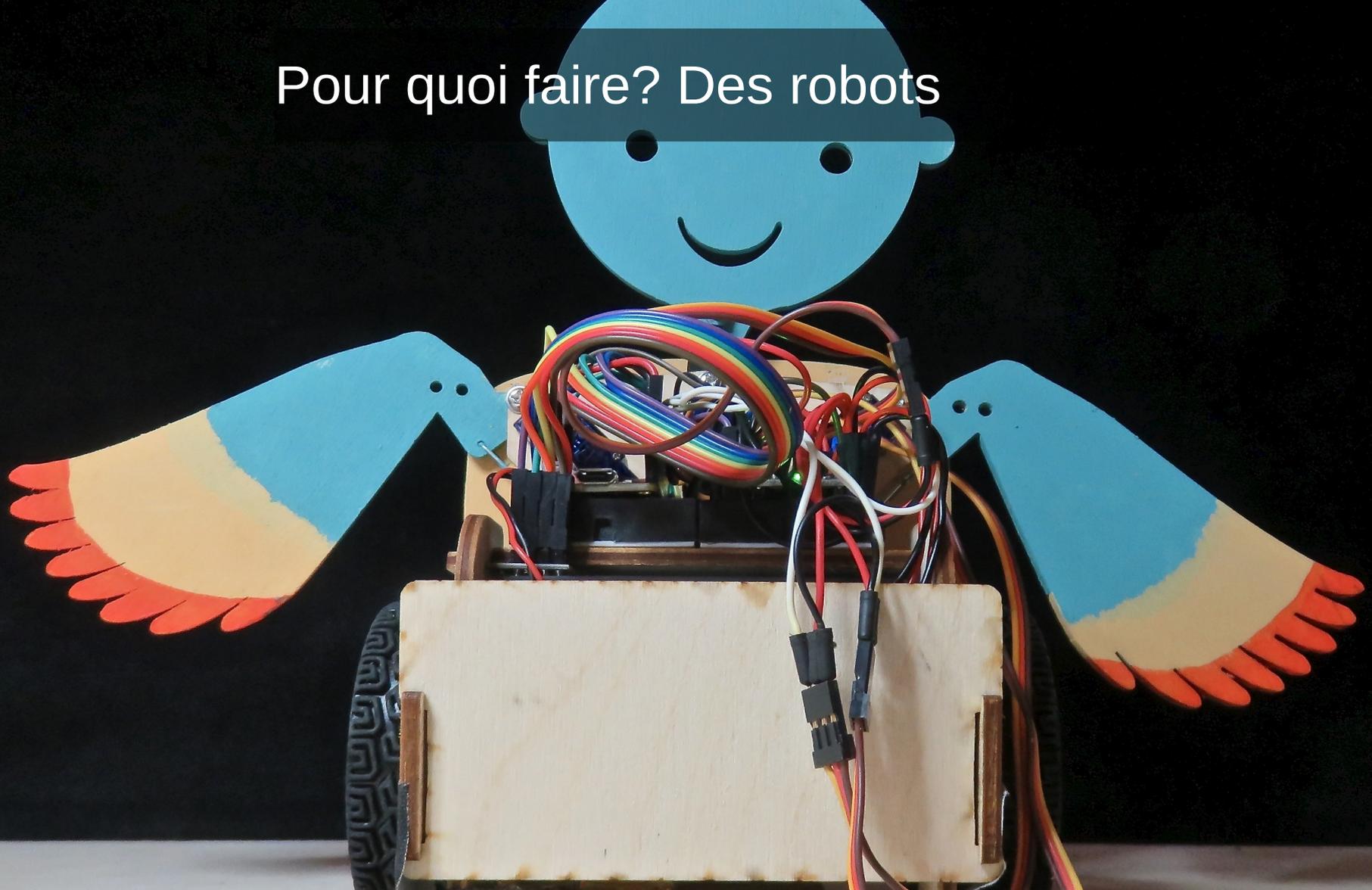
Pour quoi faire? Des tissus intelligents



Pour quoi faire? Dessiner sur du verre



Pour quoi faire? Des robots



Pour quoi faire? Des imprimantes 3D



Qui est derrière Arduino?

Arduino - Genuino
arduino.cc
Créé en Europe, Italie
Entreprise internationale aux USA

Cartes Originales:
Marque Arduino ou Genuino
Logo Arduino
Couleur bleue / Bleu canard

Clones:
Freduino, SainSmart, AVR-Duino...
~~Logo Arduino~~
~~Couleur bleue / Bleu canard~~

Attention aux contrefaçons

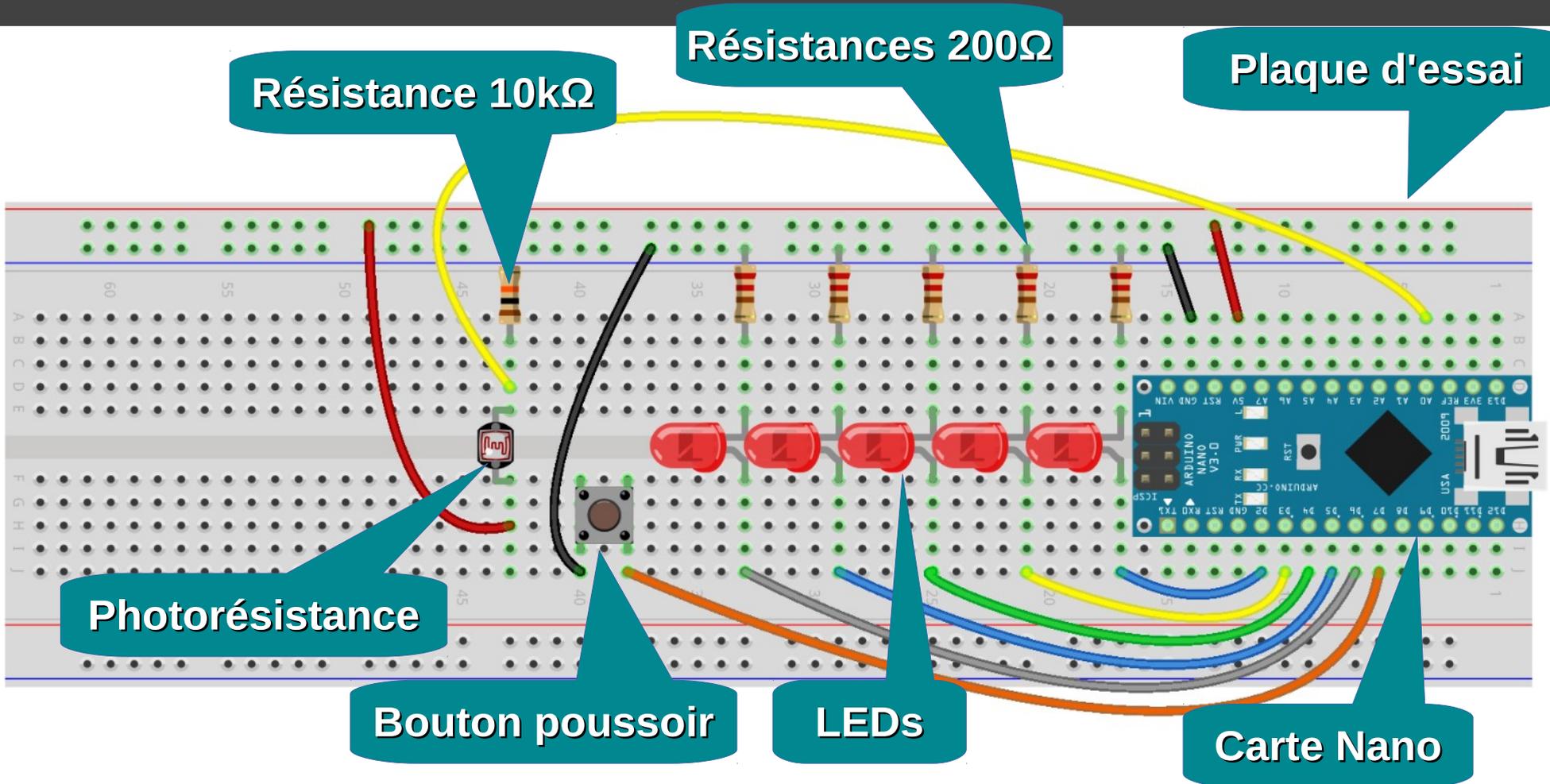
Integrated Development Environment
Environnement de développement intégré

- **Écrire des programmes / C++ / graphique**
- **Gérer les bibliothèques**
- **Compiler les programmes**
- **Envoyer les programmes sur les cartes**
- **Communiquer avec les cartes**

Dans la pratique, Utiliser Arduino c'est :

- Rédiger un programme
- Compiler et envoyer le programme sur la carte
- Assembler la carte et les composants sur une «Breadboard»
- Relier les composants avec des câbles
- Alimenter le montage
- Montage final: souder les composants sur une «Perfboard»

OK, Allons-y!



OK, Allons-y!

- Sur votre poste,
lancez le logiciel
Arduino IDE
- Un programme par
défaut apparaît

```
void setup() {  
    // put your setup code here, to run once:  
    // code à exécuter une seule fois  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    // code principal, répété en boucle indéfiniment  
}
```

OK, Allons-y!

- Sur votre poste,
lancez le logiciel
Arduino IDE
- Un programme par
défaut apparaît

```
void setup() {  
    // put your setup code here, to run once:  
    // code à exécuter une seule fois  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    // code principal, répété en boucle indéfiniment  
}
```

OK, Allons-y!

- Sur votre poste,
lancez le logiciel
Arduino IDE
- Un programme par
défaut apparaît

```
void setup() {  
  // put your setup code here, to run once:  
  // code à exécuter une seule fois  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  // code principal, répété en boucle indéfiniment  
}
```

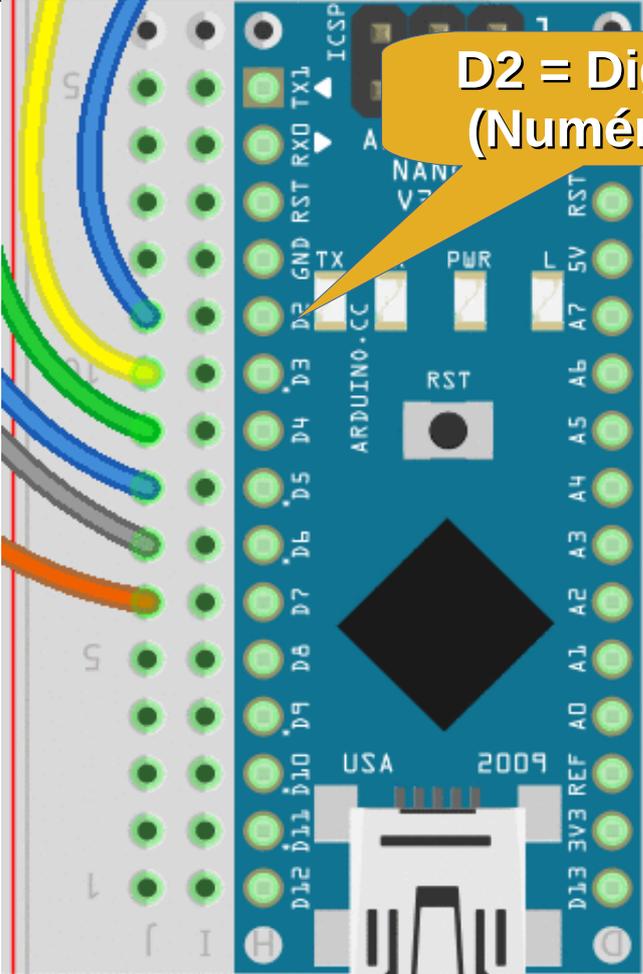
OK, Allons-y!

- Allez dans le menu Fichier > Exemples > 01.Basics > Blink
- Une nouvelle fenêtre apparaît, avec un programme d'exemple
- Brancher le montage sur l'ordinateur (USB)
- Configurer le logiciel Arduino:
 - Outils > Port > ttyUSB0
 - Outils > Type de carte > Arduino Nano



Téléverser !

OK, Allons-y!

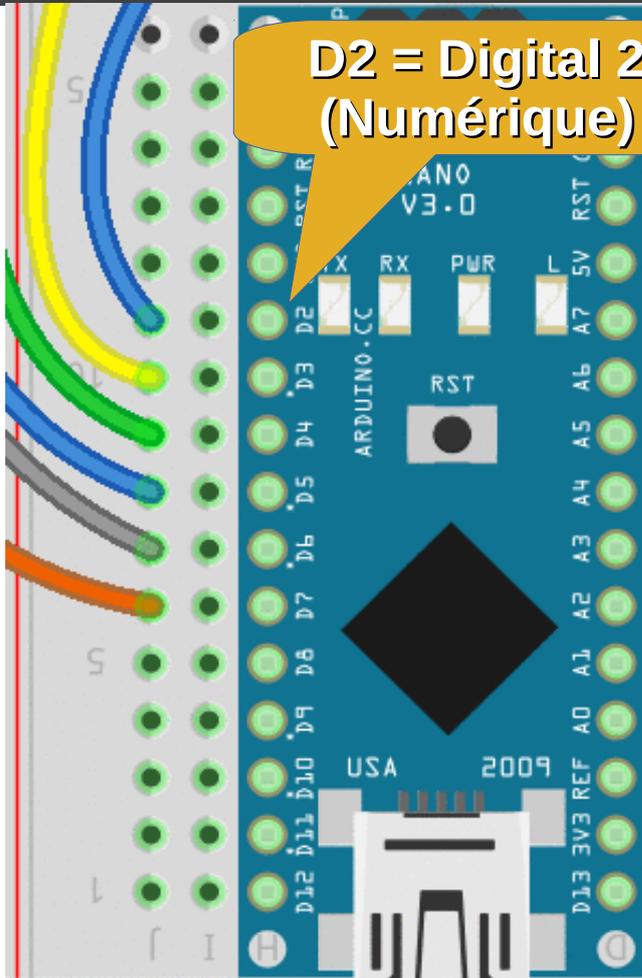


D2 = Digital 2
(Numérique)

```
void setup() {  
  // Configurer la broche LED_BUILTIN en sortie.  
  pinMode(LED_BUILTIN, OUTPUT);  
}
```

```
void loop() {  
  // On envoie une valeur haute  
  digitalWrite(LED_BUILTIN, HIGH);  
  // On attend 1 seconde  
  delay(1000);  
  // On envoie une valeur basse  
  digitalWrite(LED_BUILTIN, LOW);  
  // On attend une seconde  
  delay(1000);  
}
```

OK, Allons-y!



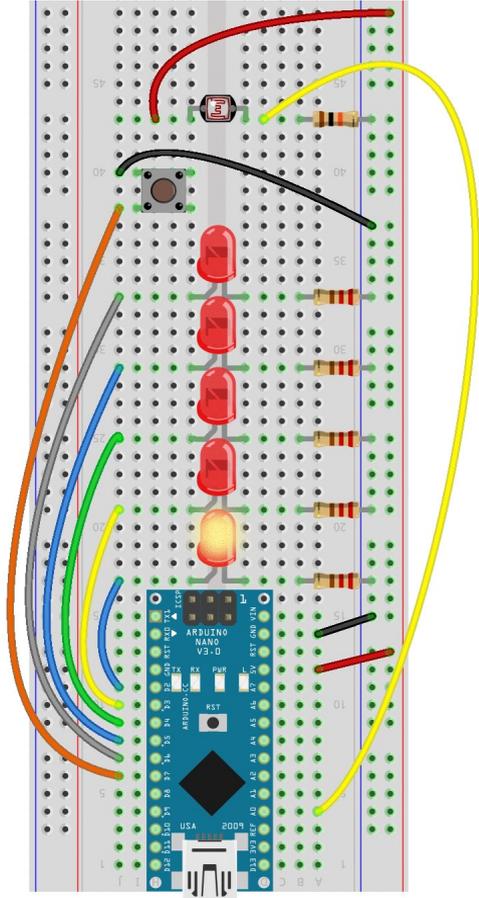
D2 = Digital 2
(Numérique)

```
void setup() {  
  // Configurer la broche LED_BUILTIN en sortie.  
  pinMode(2, OUTPUT);  
}  
void loop() {  
  // On envoie une valeur haute  
  digitalWrite(2, HIGH);  
  // On attend 1 seconde  
  delay(1000);  
  // On envoie une valeur basse  
  digitalWrite(2, LOW);  
  // On attend une seconde  
  delay(1000);  
}
```



Ensuite, Téléverser !

OK, Allons-y!



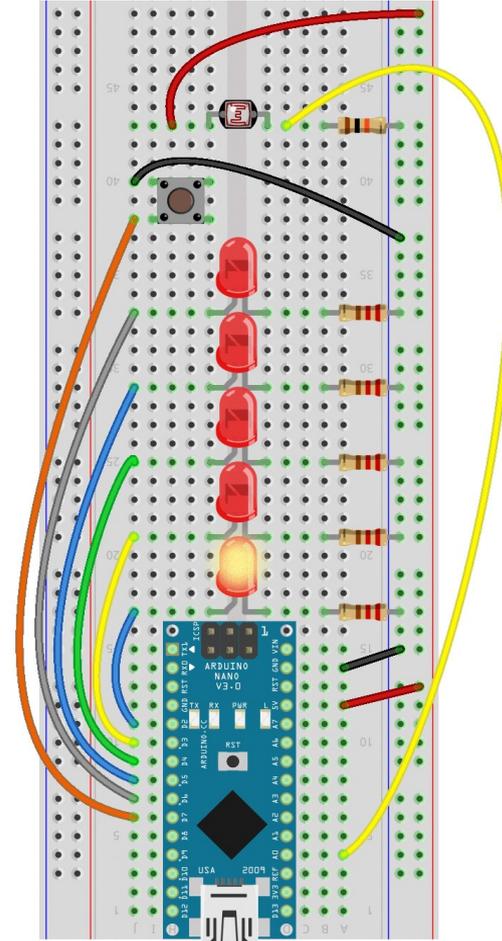
```
void setup() {  
  // Configurer la broche LED_BUILTIN en sortie.  
  pinMode(2, OUTPUT);  
}  
void loop() {  
  // On envoie une valeur haute  
  digitalWrite(2, HIGH);  
  // On attend 1 seconde  
  delay(1000);  
  // On envoie une valeur basse  
  digitalWrite(2, LOW);  
  // On attend une seconde  
  delay(1000);  
}
```



Ensuite, Téléverser !

OK, Allons-y!

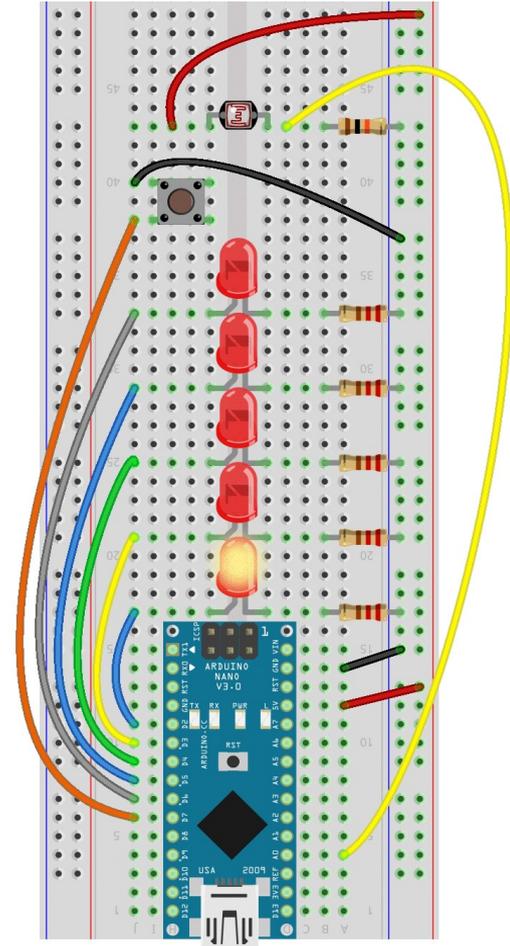
On souhaite maintenant faire clignoter les leds les unes après les autres.
Savez-vous comment faire?



OK, Allons-y!

Voici comment
fonctionne le
programme actuel.
Basez-vous sur ce
code pour la suite

Allumer la LED 2
Attendre 1 seconde
Éteindre la LED 2
Attendre 1 seconde



OK, Allons-y!

Il faudra coder cela
dans la boucle loop()

N'oubliez pas de
configurer les autres

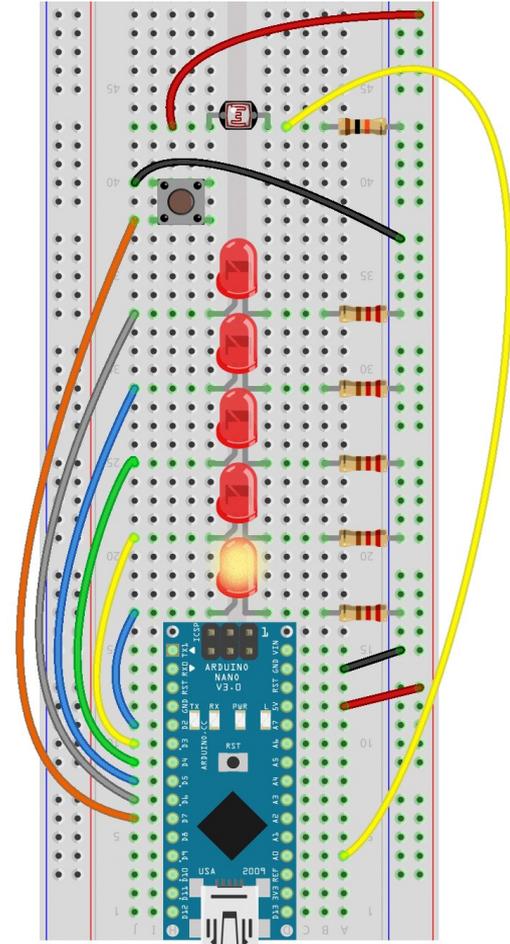
LEDs en sortie dans
la partie setup()

Allumer la LED 2
Attendre 1 seconde
Éteindre la LED 2

Allumer la LED 3
Attendre 1 seconde
Éteindre la LED3

Allumer la LED 4
Attendre 1 seconde
Éteindre la LED4

...



OK, Allons-y!

Vous devriez avoir
quelque chose
comme ça:

```
void setup() {  
    pinMode(2, OUTPUT);  
    pinMode(3, OUTPUT);  
    pinMode(4, OUTPUT);  
    pinMode(5, OUTPUT);  
    pinMode(6, OUTPUT);  
}  
void loop() {  
    digitalWrite(2, HIGH);  
    delay(1000);  
    digitalWrite(2, LOW);  
  
    digitalWrite(3, HIGH);  
    delay(1000);  
    digitalWrite(3, LOW);  
  
    digitalWrite(4, HIGH);  
    delay(1000);  
    digitalWrite(4, LOW);  
  
    // Et pareil pour LED 5 et LED 6
```

OK, Allons-y!

- Allez dans le menu Fichier > Carnet de croquis > Atelier
- Téléversez le programme sur la carte
- Le programme utilise:
 - les sorties numériques (LEDs), Entrées numériques (Bouton), Entrée analogique (Photorésistance)
 - les boucles for(), des conditions if(), des variables (i, luminosite), du port série.

Description du programme:

Lors de l'appui sur le bouton, une certaine quantité de LEDs s'allument en fonction de la luminosité mesurée par la photorésistance.

Les valeurs mesurées sont envoyées à l'ordinateur par le port série.

Projets personnels

- Glockenspiel: instrument de musique automatisé
- Lampe d'ambiance animée
- Contrôleur de lecteur audio
- Imprimante 3D

Pour apprendre

- <http://zestedesavoir.com> / <http://eskimon.fr>
- <http://playground.arduino.cc/French/Reference>
- <http://forum.arduino.cc/>
- Kit Arduino : carte + cables + breadboard + LEDs + Résistances + boutons + afficheurs + servos ... < 20€
- Amazon - Aliexpress - Gearbest

Licences des médias

Table LED
cc by Windell Oskay
<https://www.flickr.com/photos/oskay/121705244/in/photostream/>

Auto Squiggler landscape
cc by-sa Patrick Dinnen
<https://www.flickr.com/photos/pdinnen/14361133625/>

Weather Widget
cc by David Mellis
<https://www.flickr.com/photos/mellis/16317551360/>

Rube-Goldberg Arduino-based xylophone-- Version -1
cc by fdecomite
<https://www.flickr.com/photos/fdecomite/8443261903/>

fabric.jpg - cc by-sa CRUSTINA!
Best bud sweater
<https://www.flickr.com/photos/crustina/8760362694/>

robot.jpg
cc by-sa Fabrice Florin
Robot World Prototypes
<https://www.flickr.com/photos/fabola/36507004581/in/photostream/>

cc by dvdgmz
RepRap
<https://www.flickr.com/photos/dvdgmz/8200736050/in/photostream/>

Images cliparts
Domaine public: <http://openclipart.org>

Dessin du montage pour l'atelier:
Réalisé avec Fritzing par Ludovic Kiefer
CC by-sa

Photos des Cartes Arduino
CC by-sa © Arduino

Arduino et Genuino sont des marques déposées
Le logo Arduino est une marque déposée
Ce document est rédigé par un passionné qui n'est aucunement lié à
Arduino, et sans leur aide, en respectant l'utilisation de leur marque.

Réalisé grâce à:
LibreOffice, GIMP, Imagemagick

Auteur de la présentation: Ludovic Kiefer
Vous pouvez réutiliser cette présentation selon les termes de la
licence Creative Commons by-sa

