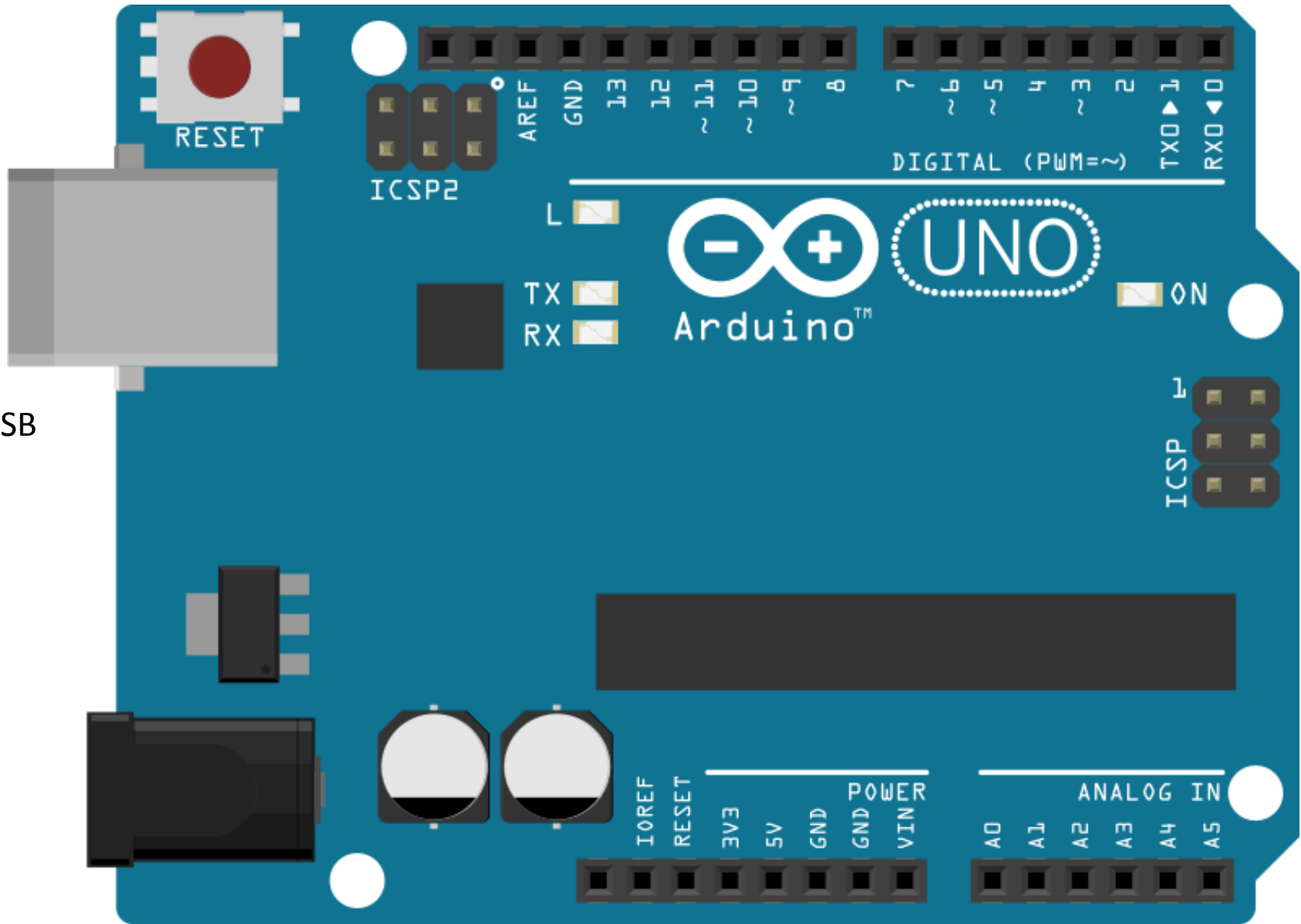




Initiation Arduino

Club astro QF – Pascal ANDRE fev 2016



14 Broches Digital dont 6 PWM
(pulse with modulation)
E/S « 0 ou 1= 0V +5V »
3 LED : Sortie 13, Power , Tx/Rx

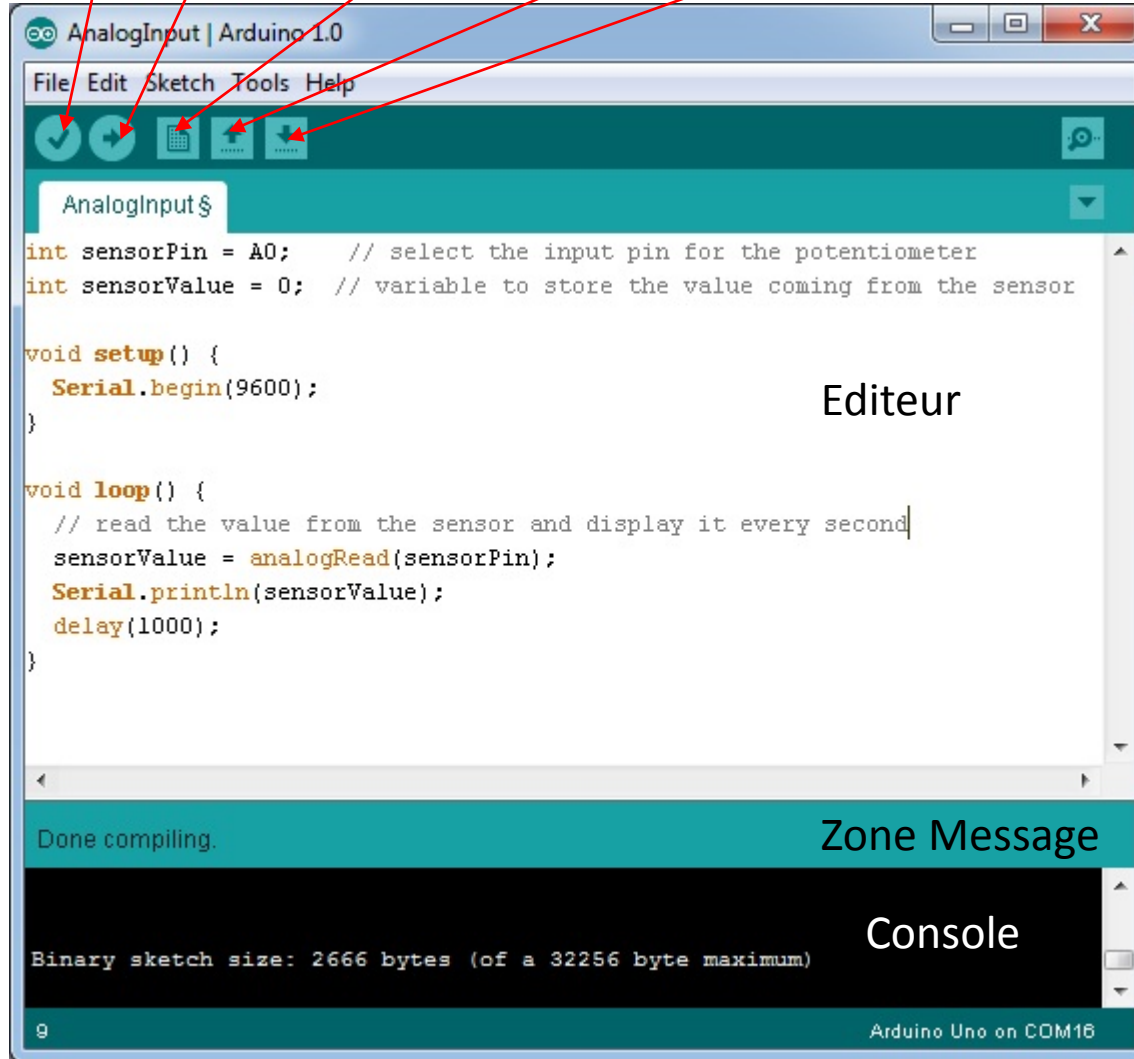
USB

Jack 7 à 12 V = VIN

Reset vers GND
Power

6 Broches analogiques (0-255)

Verifier Téléverser Nouveau Croquis Ouvrir Enregistrer



Au préalable
Identifier Carte
Identifier Port

Un bout de code = un croquis
Exemple

Basic

Blink

Librairies : 2 emplacements

Mes documents/arduino

ProgFiles/arduino

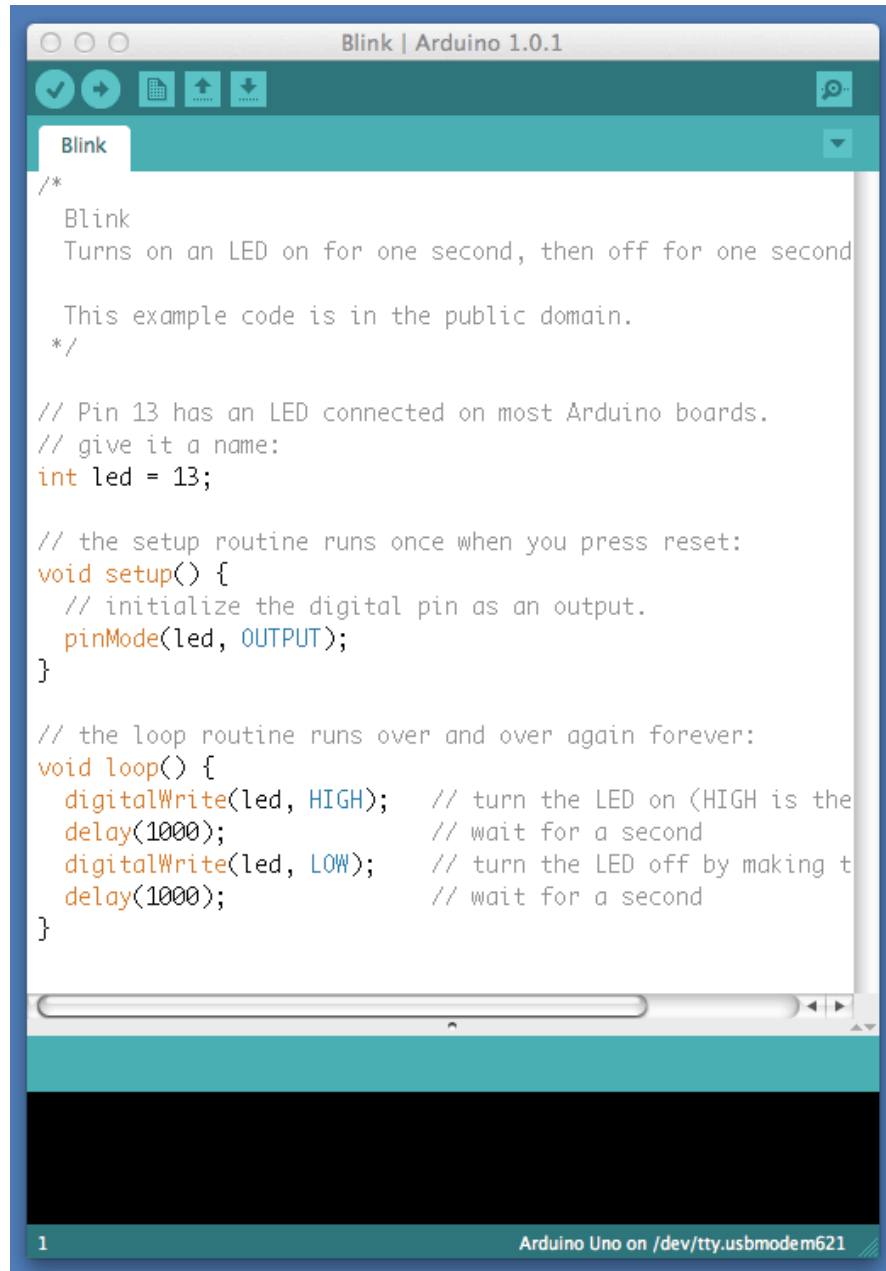
Attention

Aux conflits /emplacements

Aux versions

Au rangement > Ss repertoire

Du même nom *.cpp *.h

The image shows a screenshot of the Arduino IDE interface. The window title is "Blink | Arduino 1.0.1". The code editor displays the following C++ code for a Blink sketch:

```
/*
 * Blink
 * Turns on an LED on for one second, then off for one second
 *
 * This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the
  delay(1000); // wait for a second
  digitalWrite(led, LOW); // turn the LED off by making t
  delay(1000); // wait for a second
}
```

The IDE interface includes a toolbar with icons for check, run, upload, and download. A status bar at the bottom indicates "1" and "Arduino Uno on /dev/tty.usbmodem621".

Commentaires `/* blabla */ ;//`

Instructions;

Déclaration variables

Fonction Setup `{}`

Fonction Loop `{ }` ouvre une boucle infinie)

Déclare variable led (type int - 32678 à +32768)

Broche (Pin) N° 13 en Sortie

On ouvre une boucle infinie

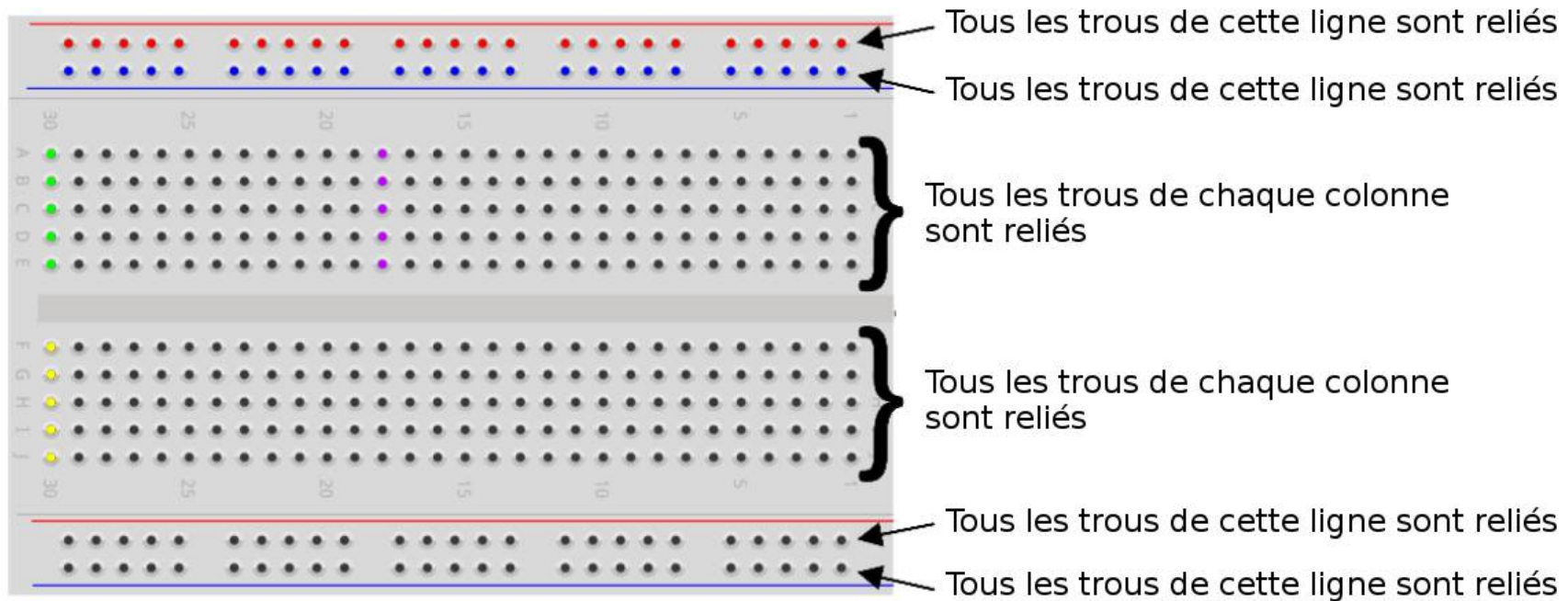
On allume la led (pin 13 mis en mode high)

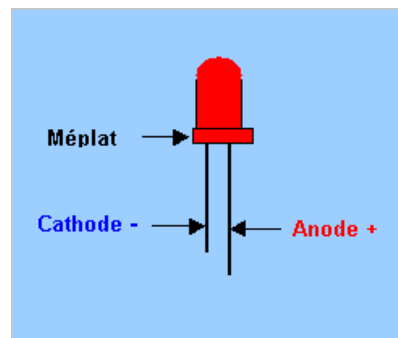
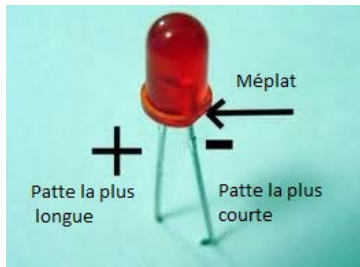
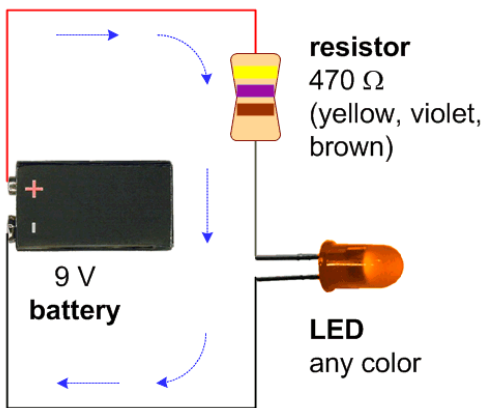
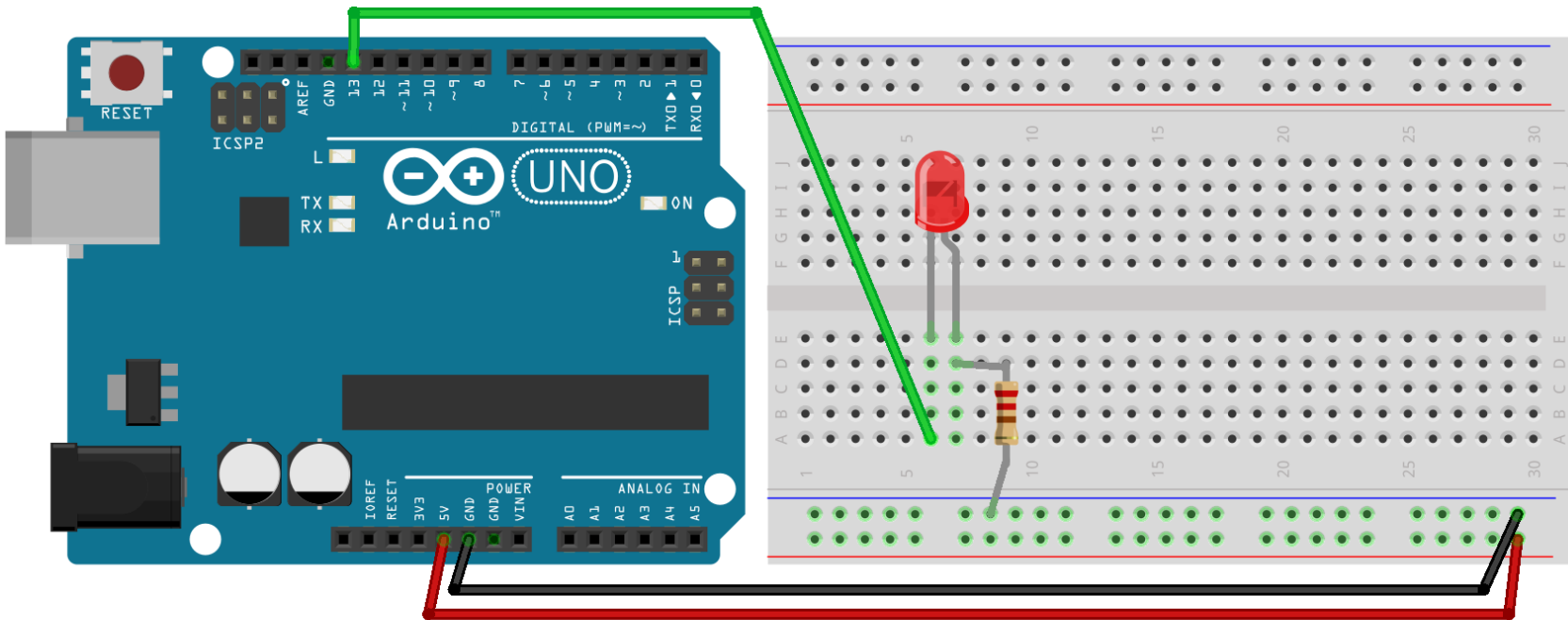
On attend 1000 ms (fonction `delay()`)

On éteint la led (pin 13 en mode low)

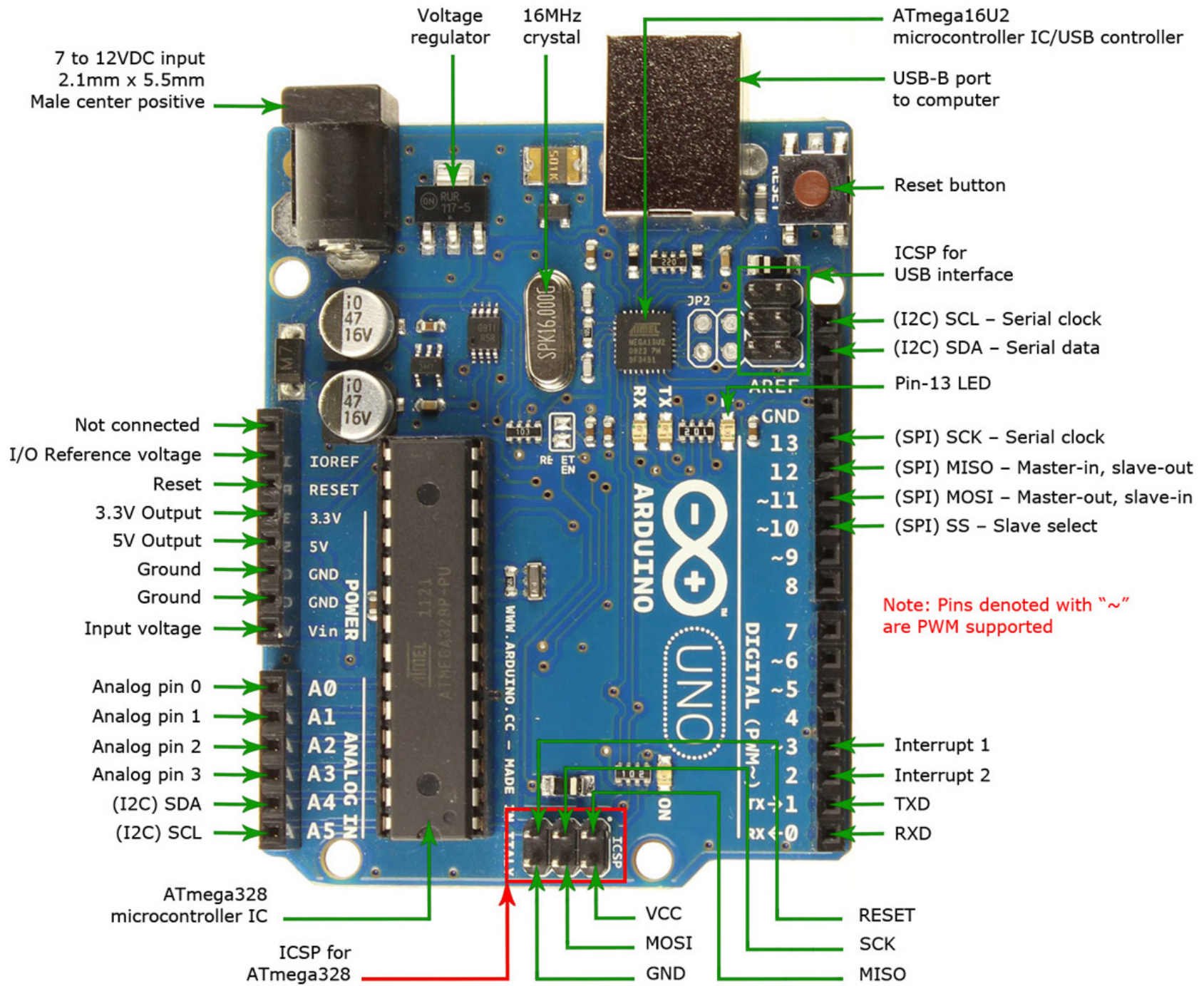
On attend 1000 ms

Et on reboucle





20 – 30 mA 2V
 $R = (V_{lim} - V_{bornes}) / i = (5 - 2) / 0,03 = 100 \text{ Ohms}$



Micro contrôleur	ATmega328
Tension de fonctionnement	5V
Tension d'alimentation recommandée	7 à 12V
Tension d'alimentation limite	6 à 20 V
Entrées/sorties numériques	14 dont 6 disposent d'une sortie PWM
Entrées analogiques	6 (ATTENTION : le niveau max en entrée doit être de 5 volts)
Courant max par broches E/S	40 mA (ATTENTION : 200 mA cumulé pour l'ensemble des broches E/S)
Courant max sur sortie 3,3 V générée par le régulateur interne	50 mA
Mémoire Flash	32 KB dont 0.5 KB utilisée par le bootloader
Mémoire SRAM	2 KB
Mémoire EEPROM	1 KB
Fréquence horloge	16 MHz

L'arduino UNO sait gérer des communications de type série RS232 ou I²C (Inter Integrated Circuit), le PWM (Pulse Width modulation), le SPI (Serial Peripheral Interface) et possède deux entrées d'interruption externes.