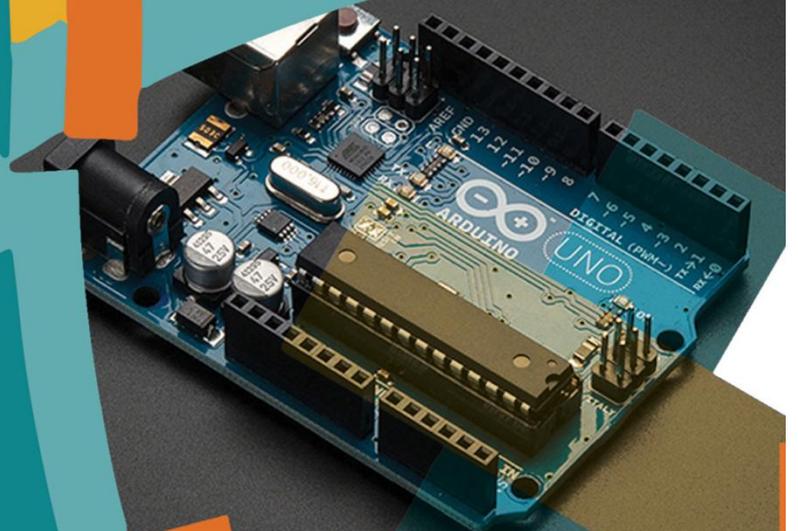


Initiation à Arduino (Ardublock)

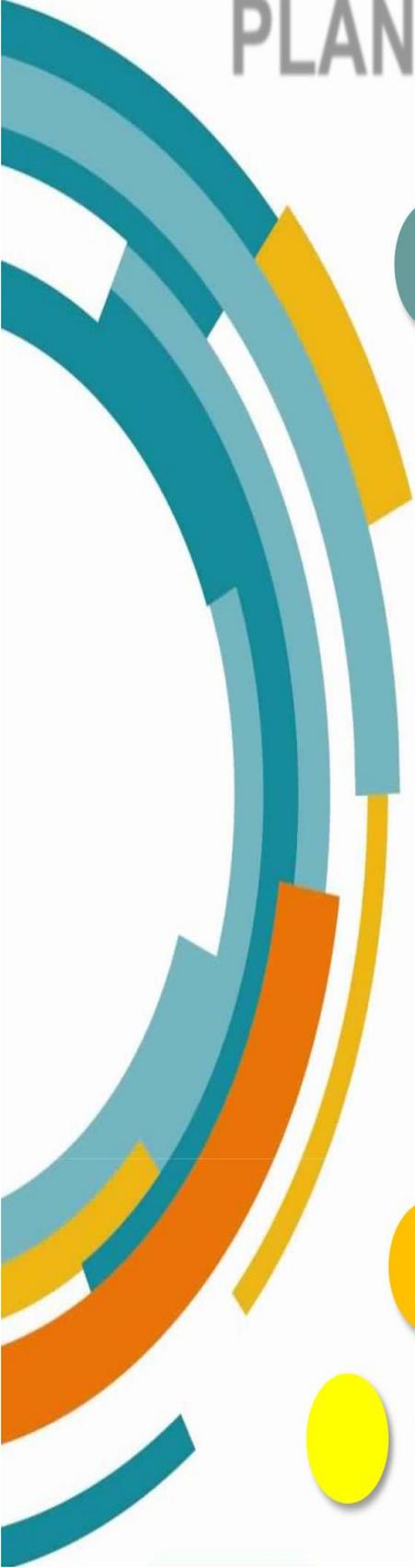
Réalisée par : - Mr. Khaled TATOUH

Encadrement : - Mr. Nasr BEN MOHAMED
- Mr. Moez MALLOUKI

Arduino



PLAN du WORKSHOP

- 
- **Prise en main Arduino – Ardublock**
 - **Gestion des entrées/sorties numériques (TOR)**
 - **Utilisation de la communication série**
 - **Gestion des entrées/sorties analogiques**
 - **Utilisation du module ultrason **HC-SR04****
 - **Commande de servomoteur et moteur à CC**

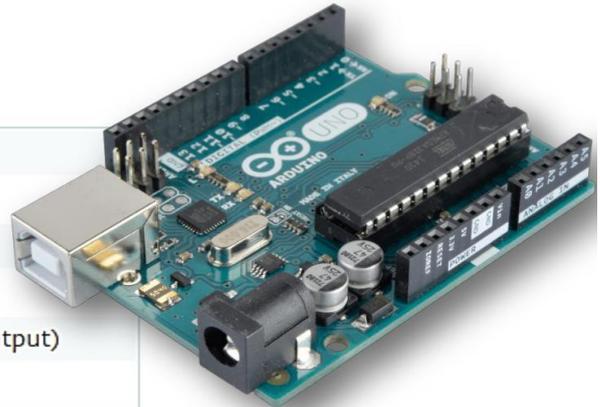


PRISE EN MAIN D'ARDUINO UNO

PRISE EN MAIN D'ARDUINO UNO

Technical specs

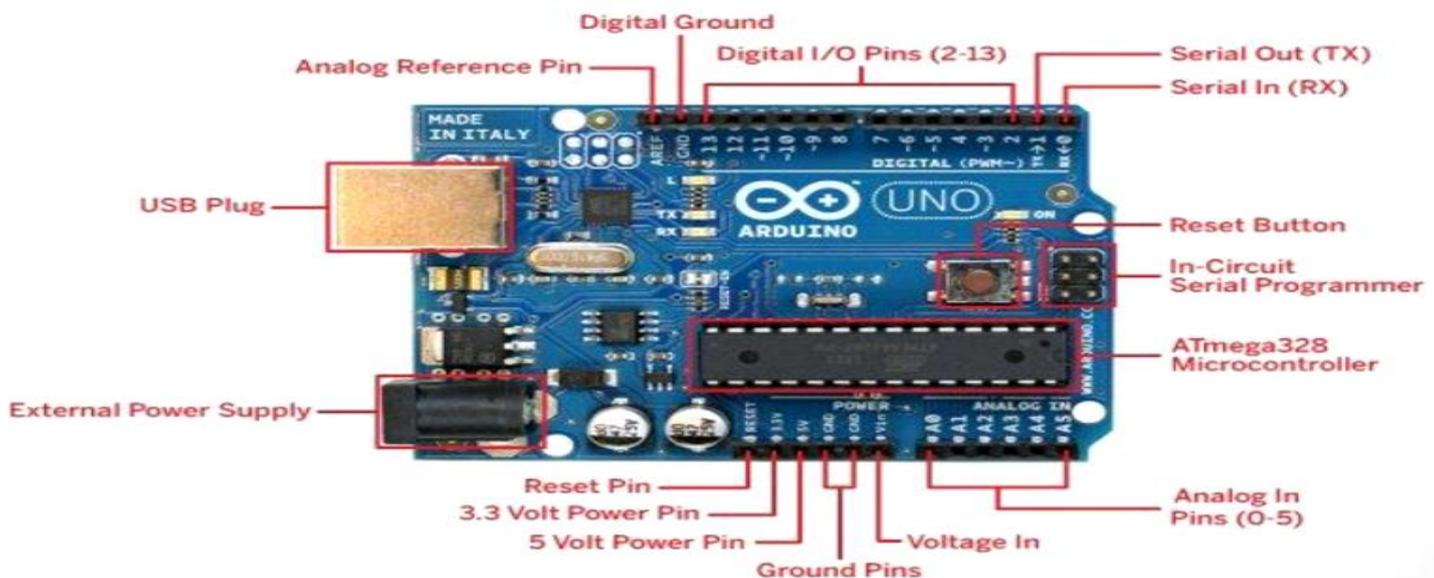
Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g



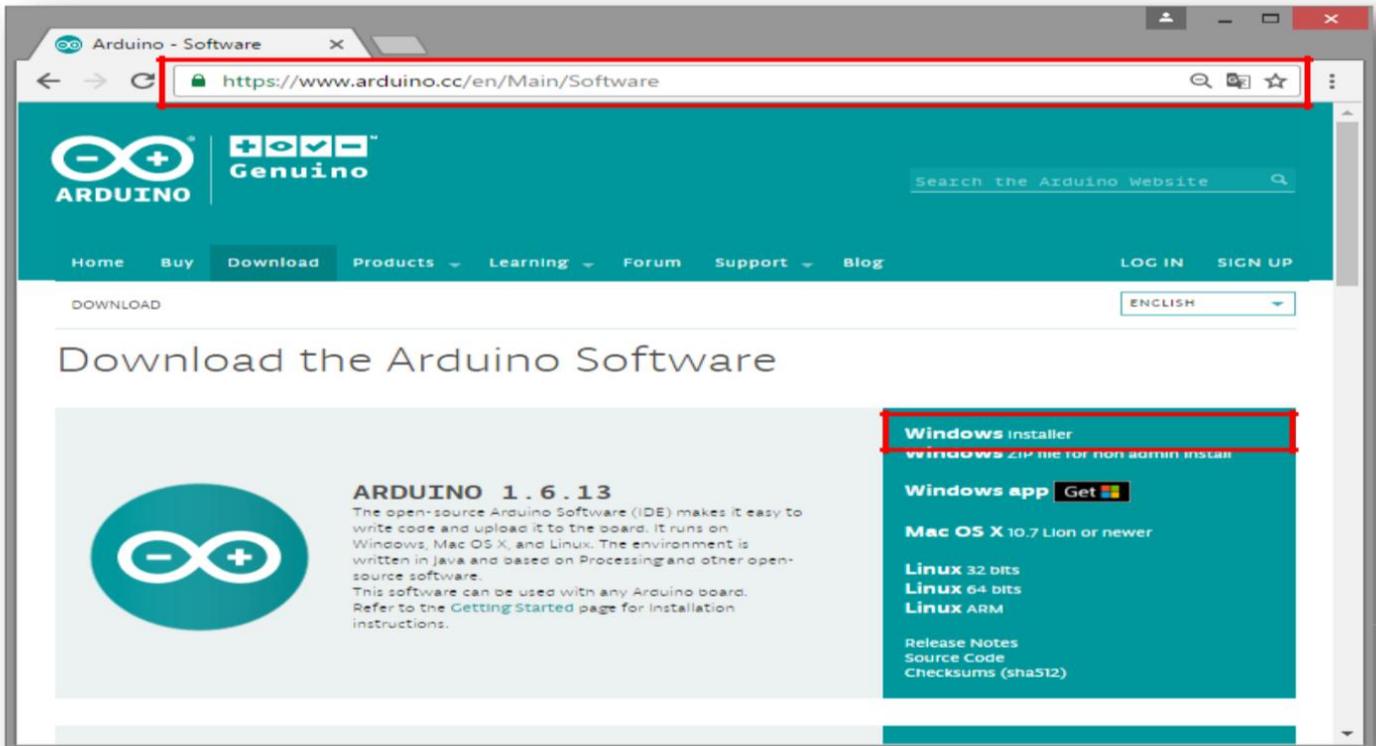
Overview



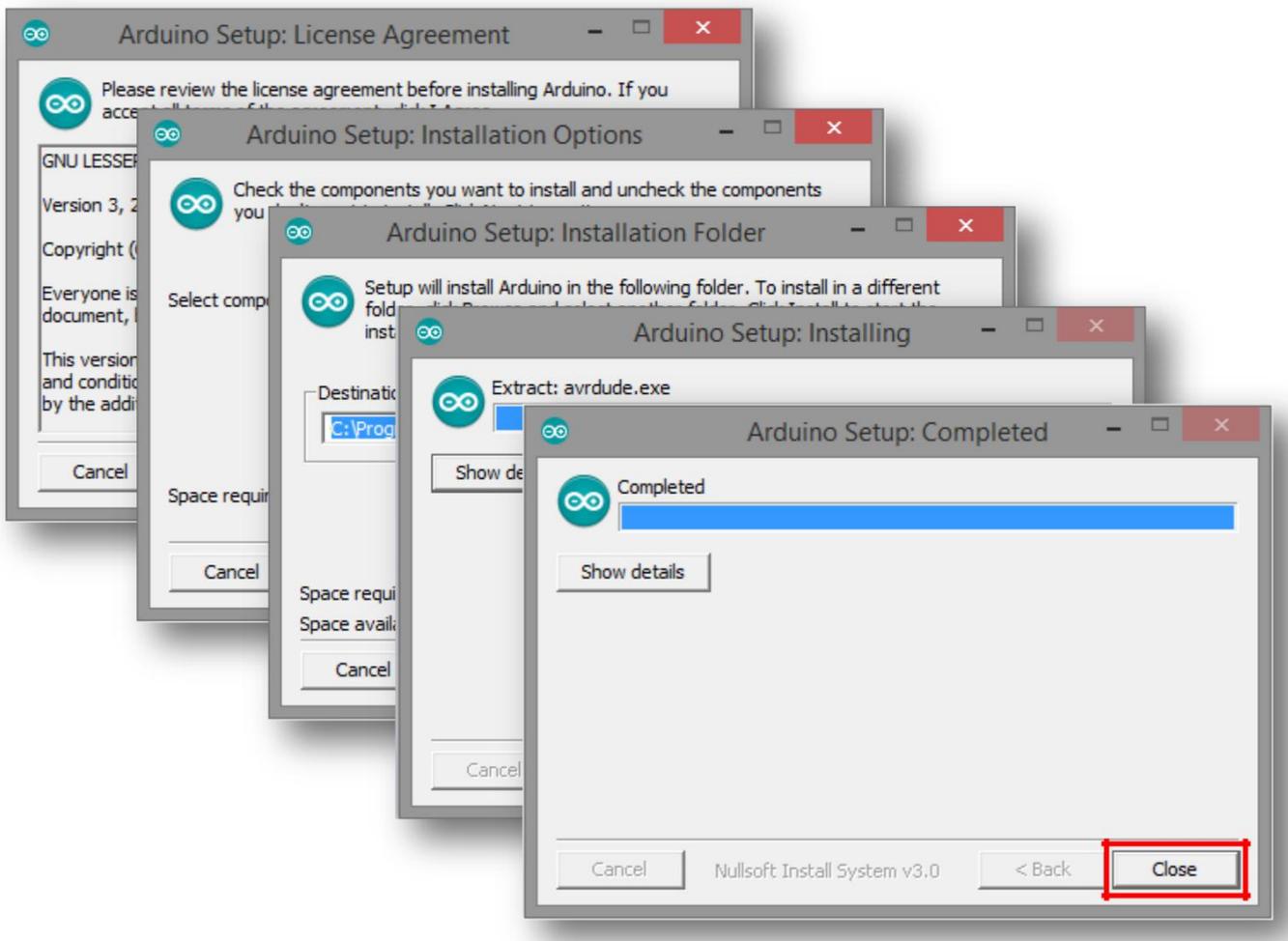
Les différentes broches de la carte ARDUINO UNO



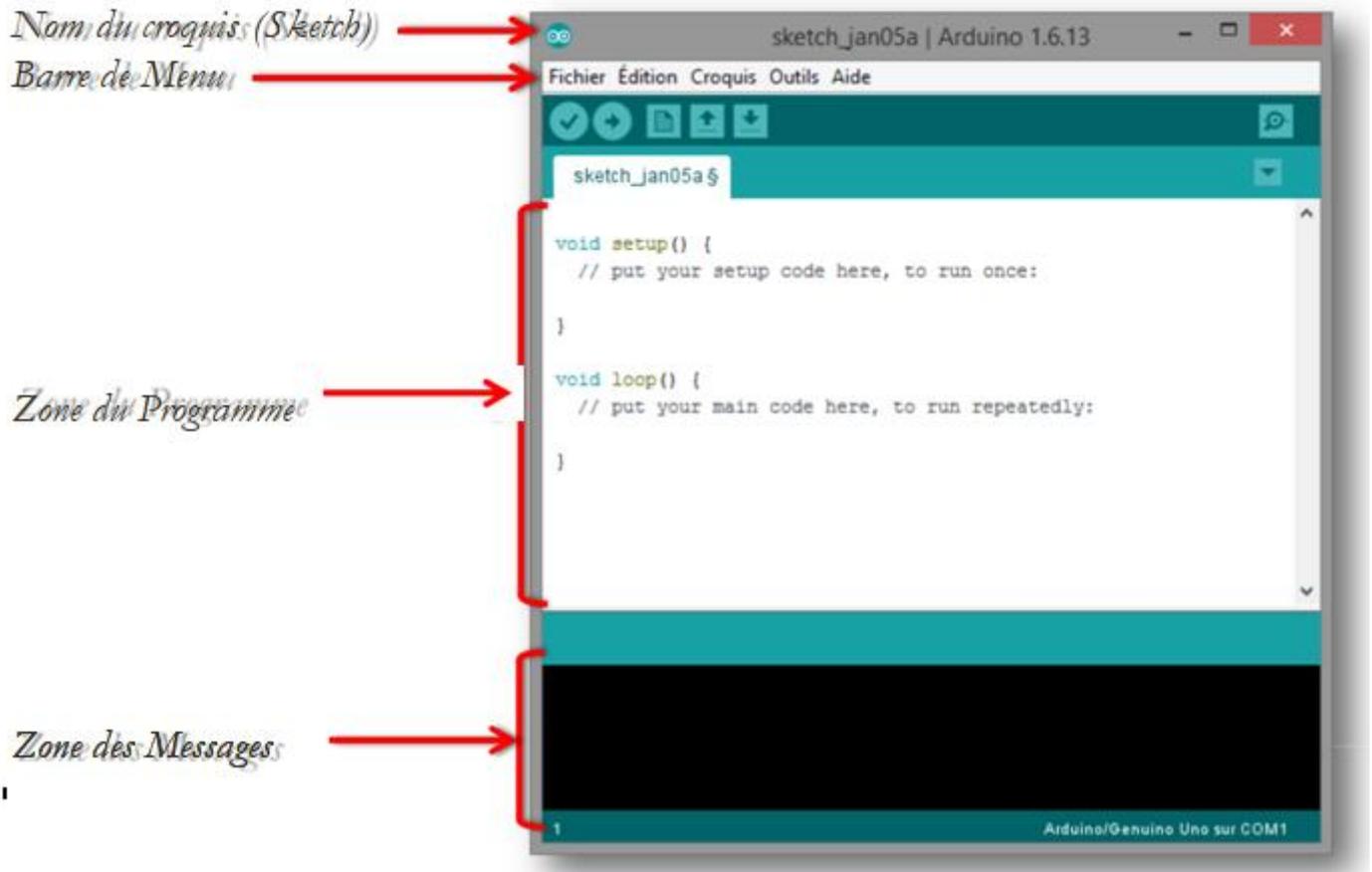
Téléchargement du Arduino IDE



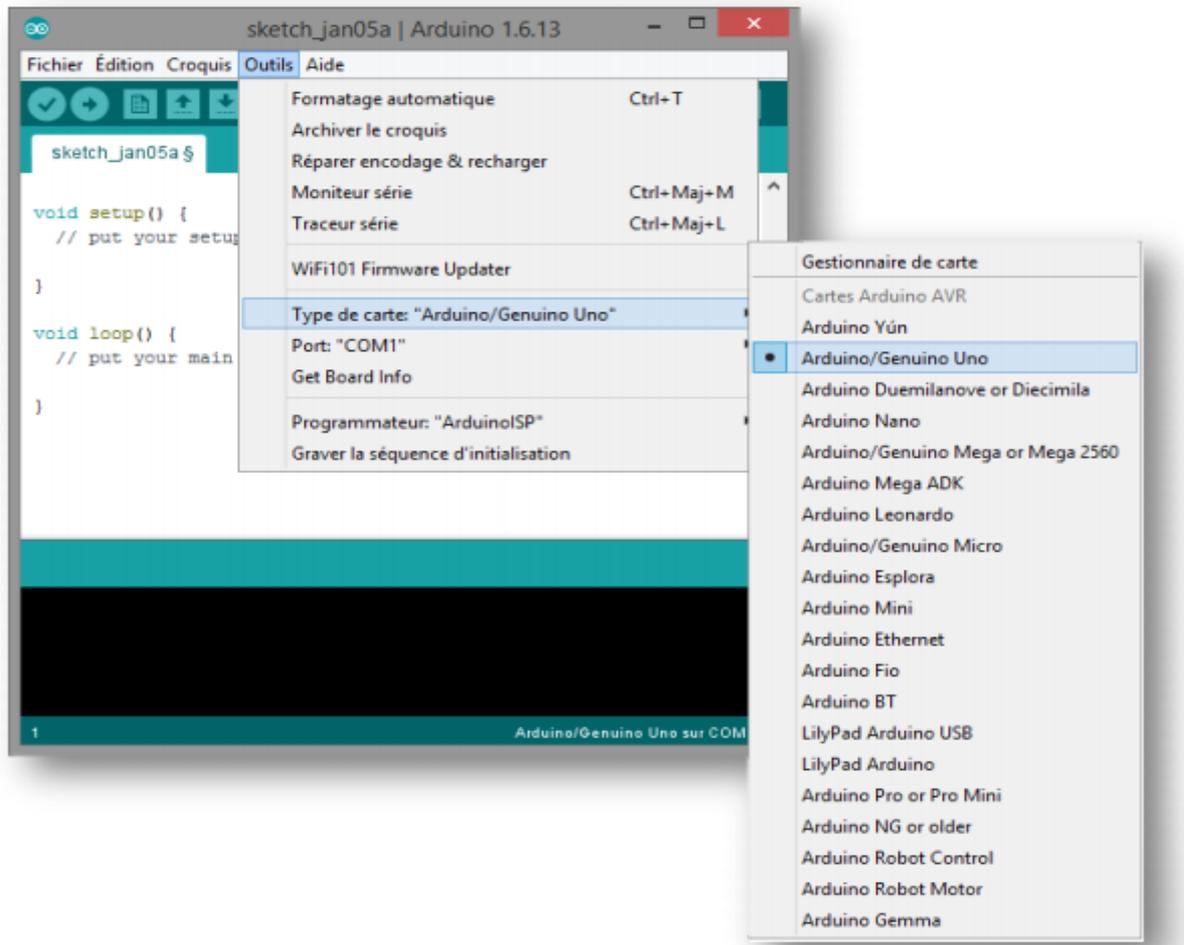
Installation du Arduino IDE



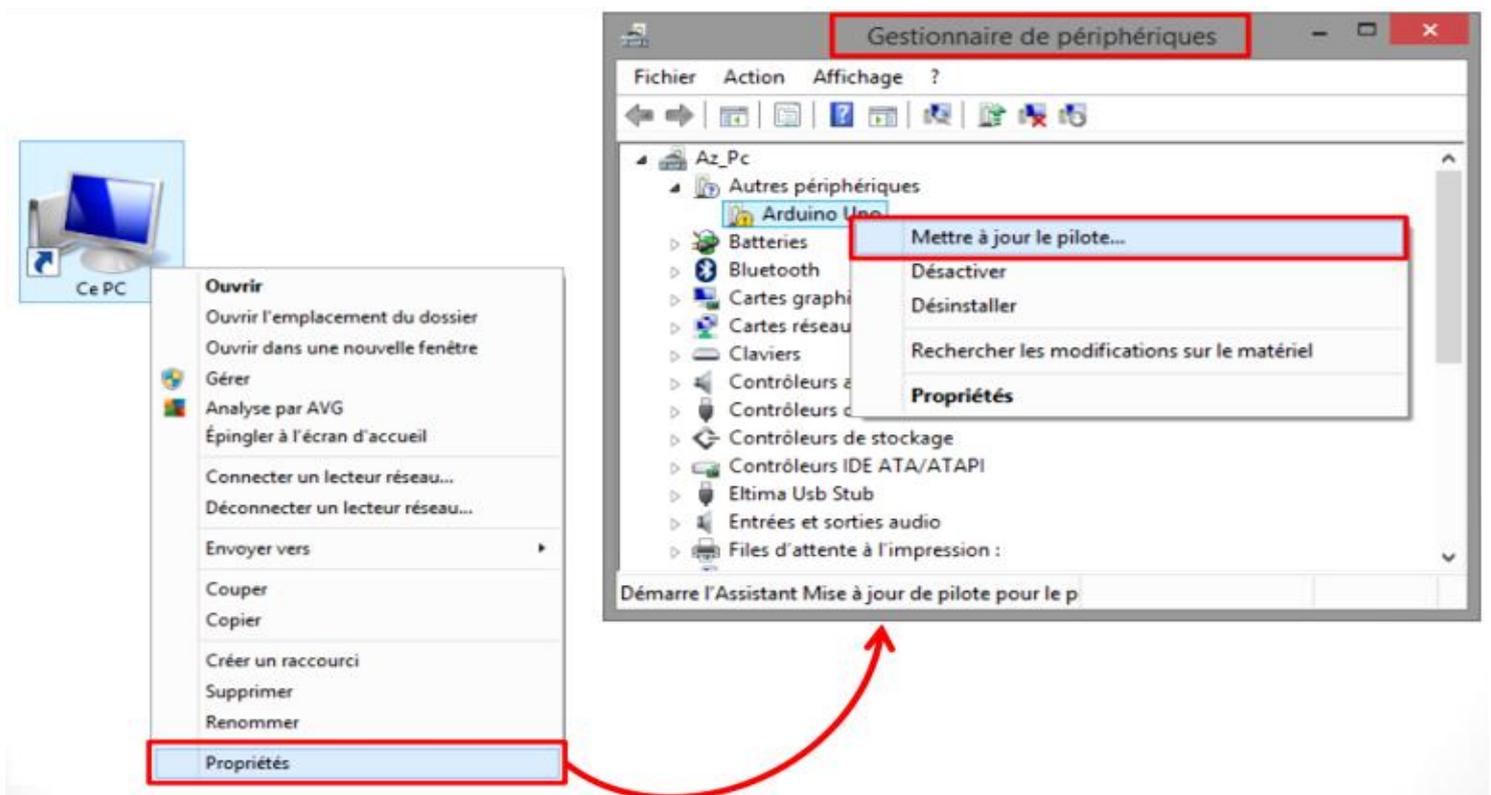
Interface de l'Arduino IDE



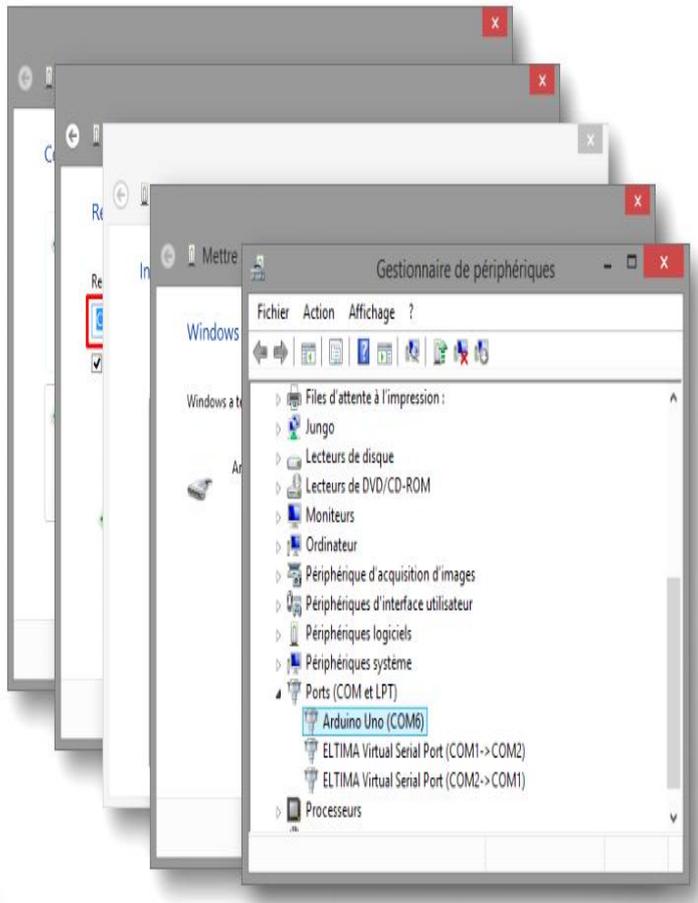
Sélection de la Carte Cible



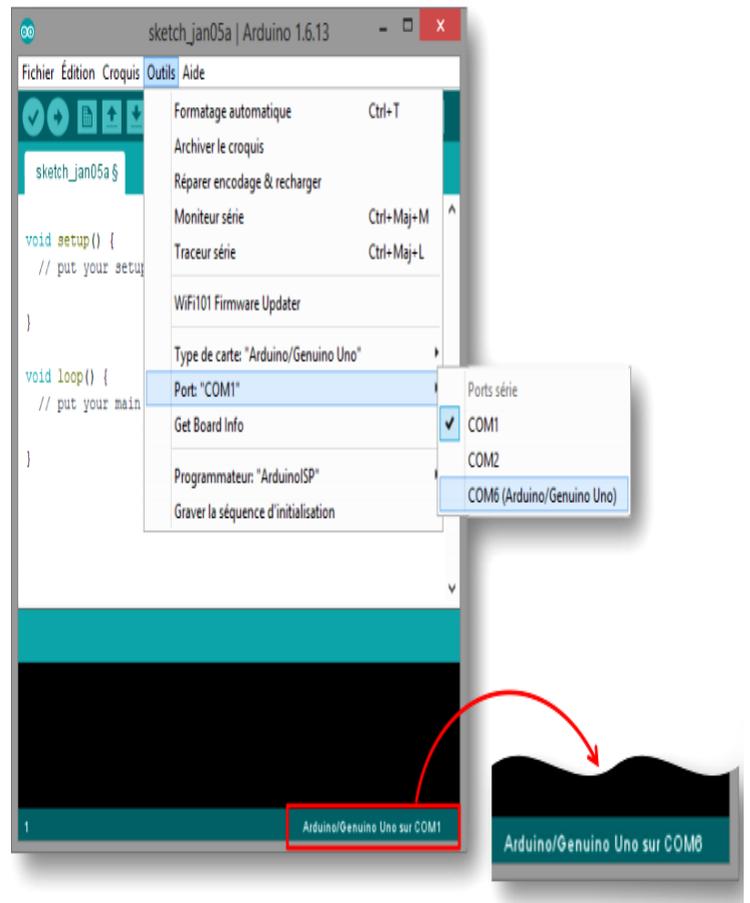
Installation du Pilote



Installation du Pilote



Sélection du Port Série



Structure du Programme

Zone des déclarations

Bibliothèques, Constantes, Variables, ...

Programme de Configuration

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

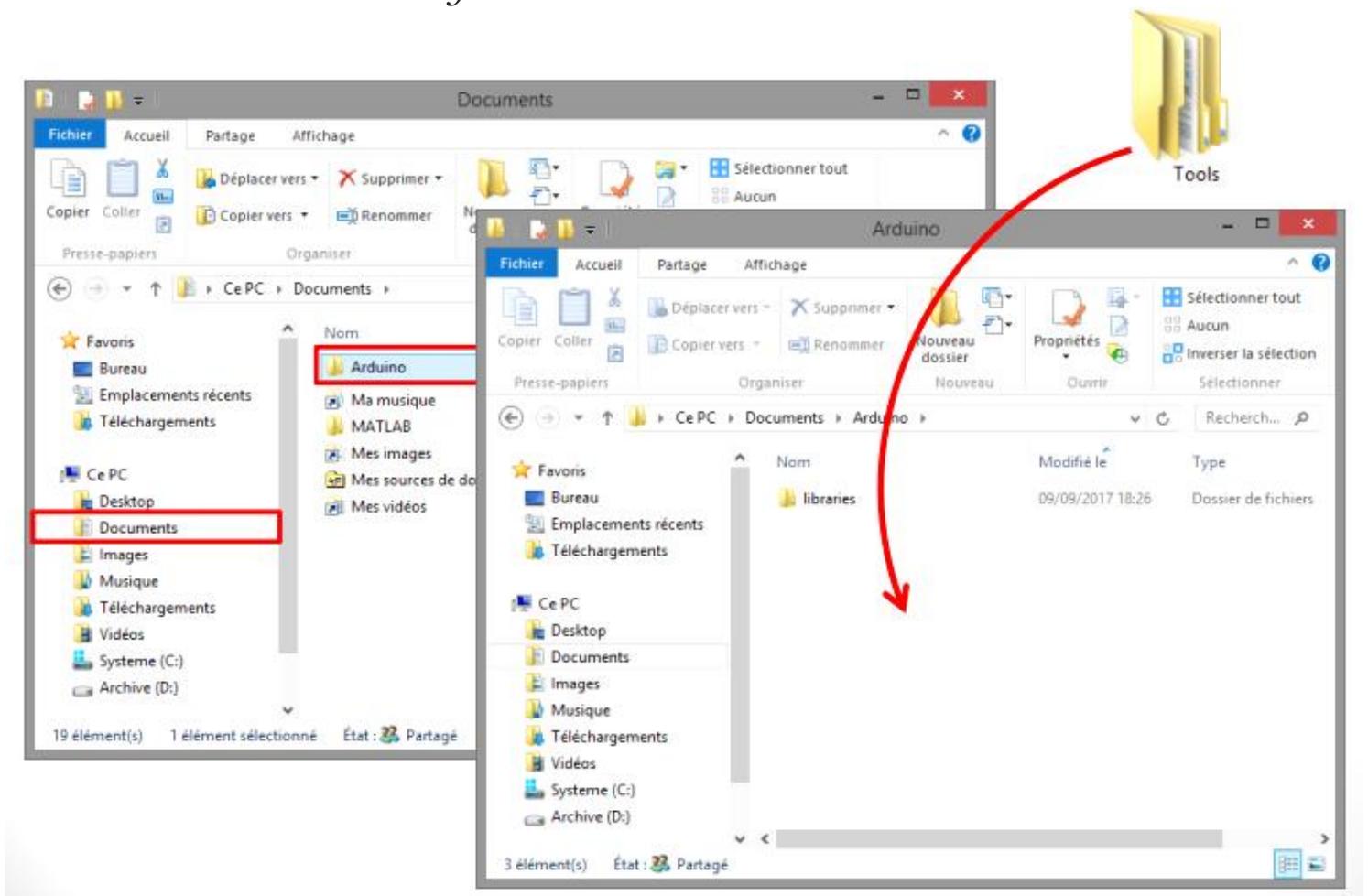
S'exécute une seule fois

Programme Principale

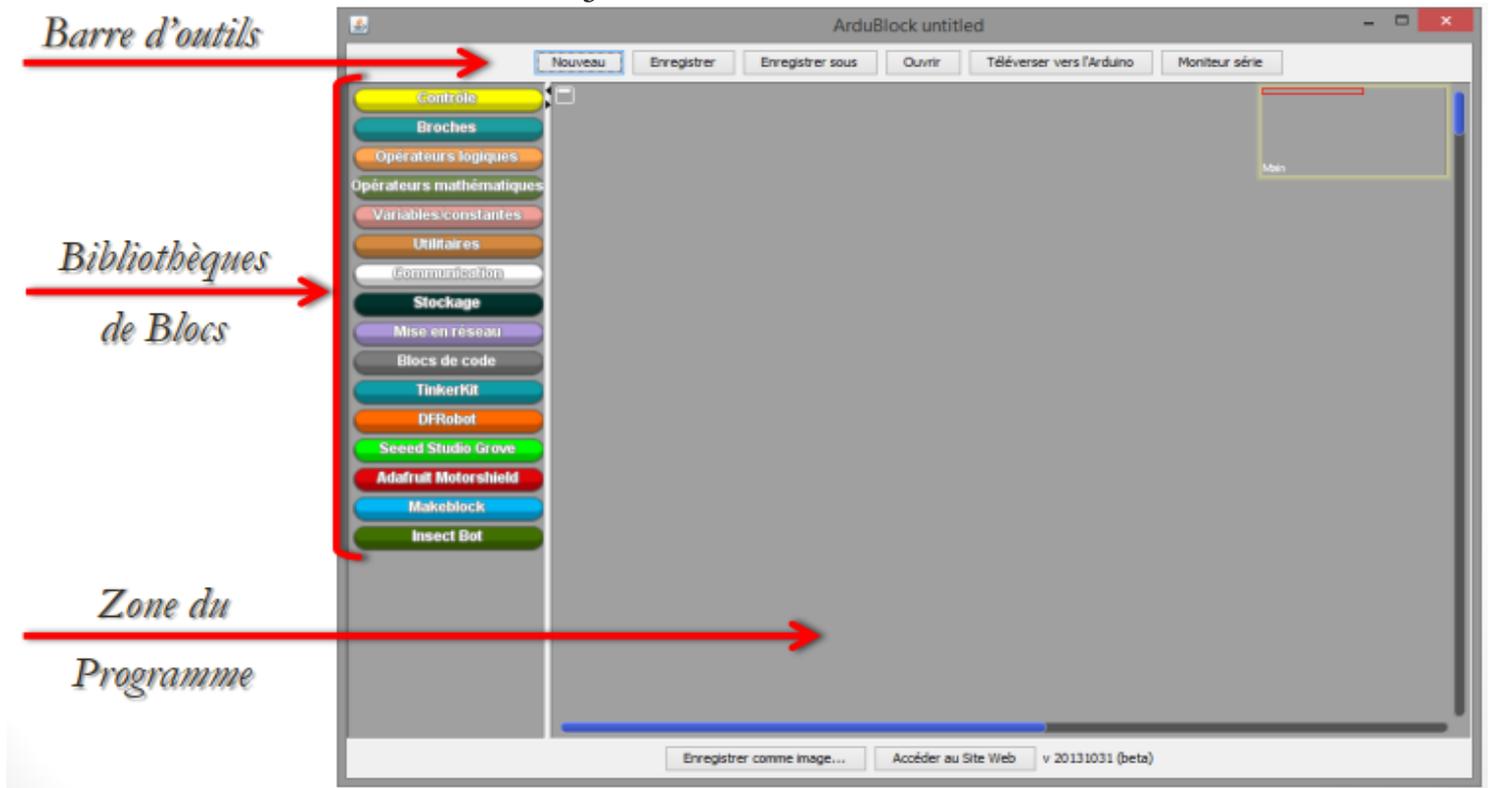
```
void loop() {
  // put your main code here, to run repeatedly:
}
```

S'exécute en boucle infinie

Ajout de l'outil Ardublock



Interface du Ardublock



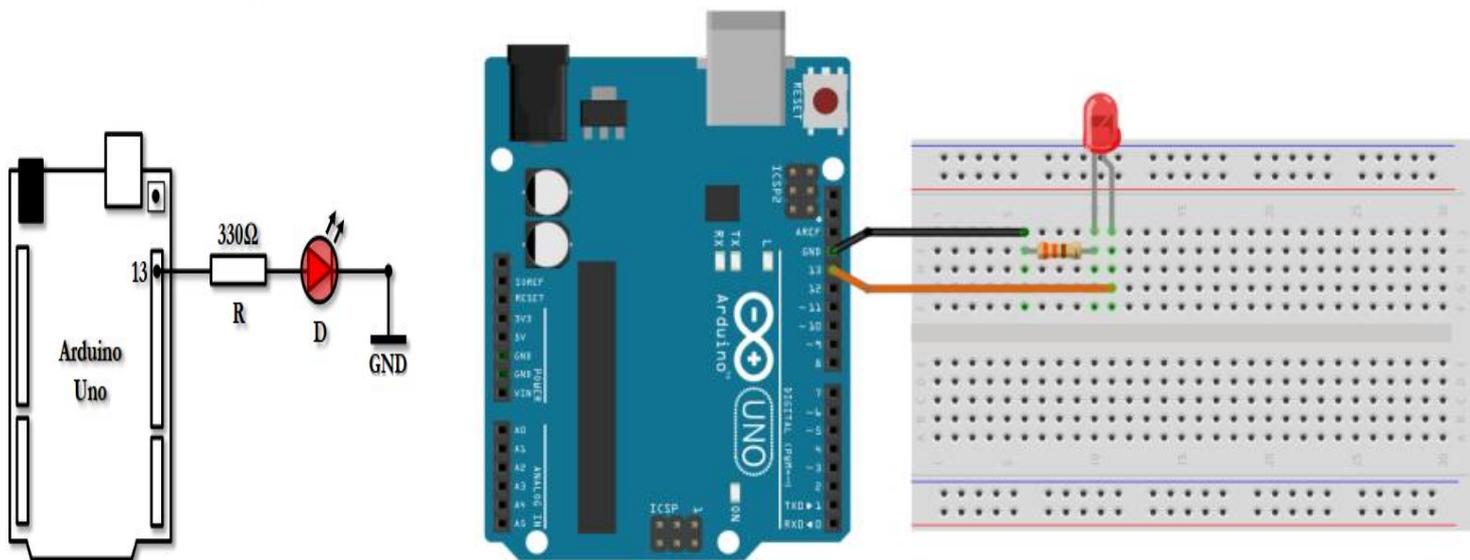
ATELIER N° 1 : GESTION DES SORTIES NUMÉRIQUES (TOR)

Activité 1 : Commande d'une seule diode LED

Activité 1 : Commande d'une seule diode LED

Montage du Circuit

Schéma Electrique



► Travail demandé :

1) **Programmez** la carte **Arduino** afin d'**allumer** la diode **LED** reliée à la **broche 13**.



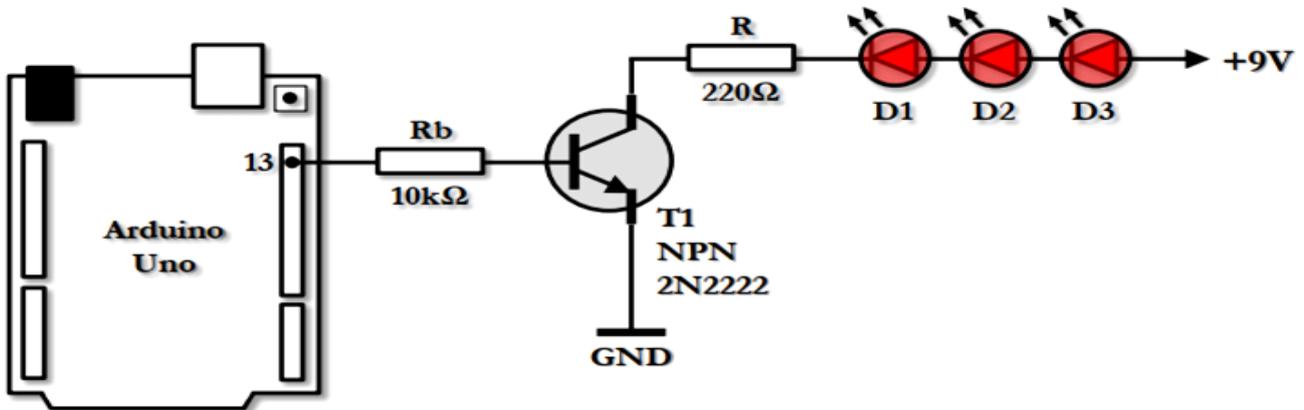
Vérifiez le même fonctionnement avec une autre broche de la carte Arduino.

2) **Reprogrammez** la carte **Arduino** afin de **clignoter** la diode **LED** reliée à la **broche 13** avec une **fréquence** de **1Hz**.

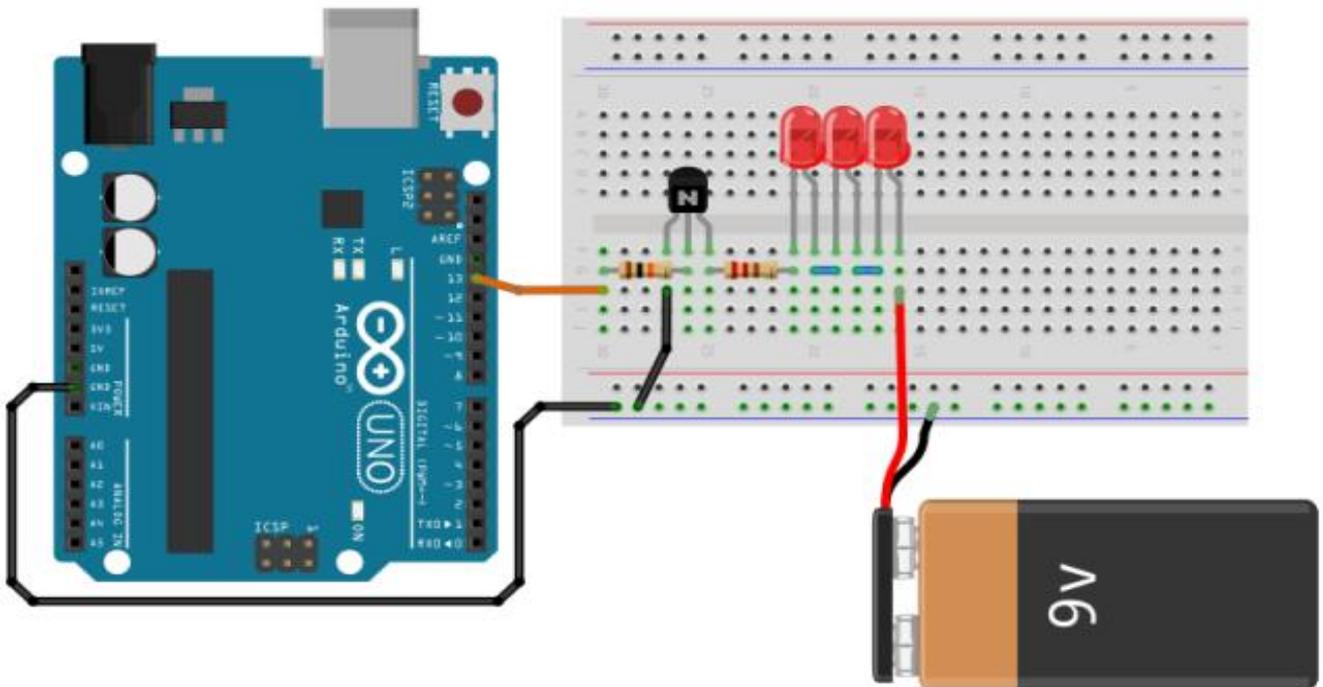


Vérifiez le même fonctionnement avec d'autres fréquences.

Activité 2 : Commande de plusieurs diodes LEDs Schéma Electrique



Montage du Circuit



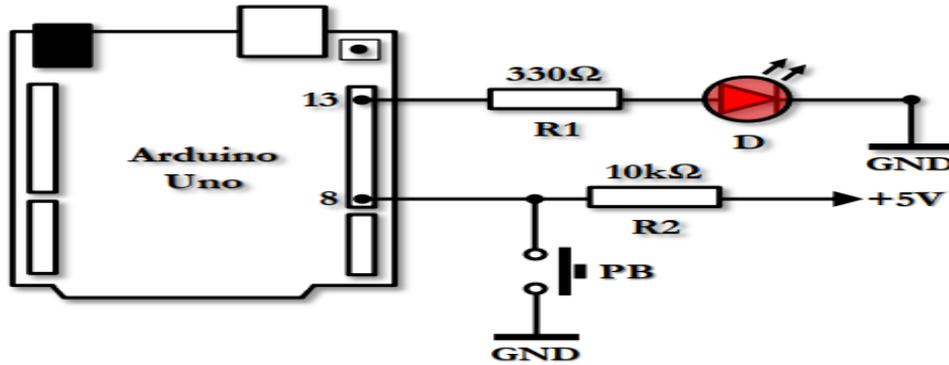
Programmez la carte Arduino et vérifiez que les trois LEDs clignotent bien avec une fréquence de 1Hz.



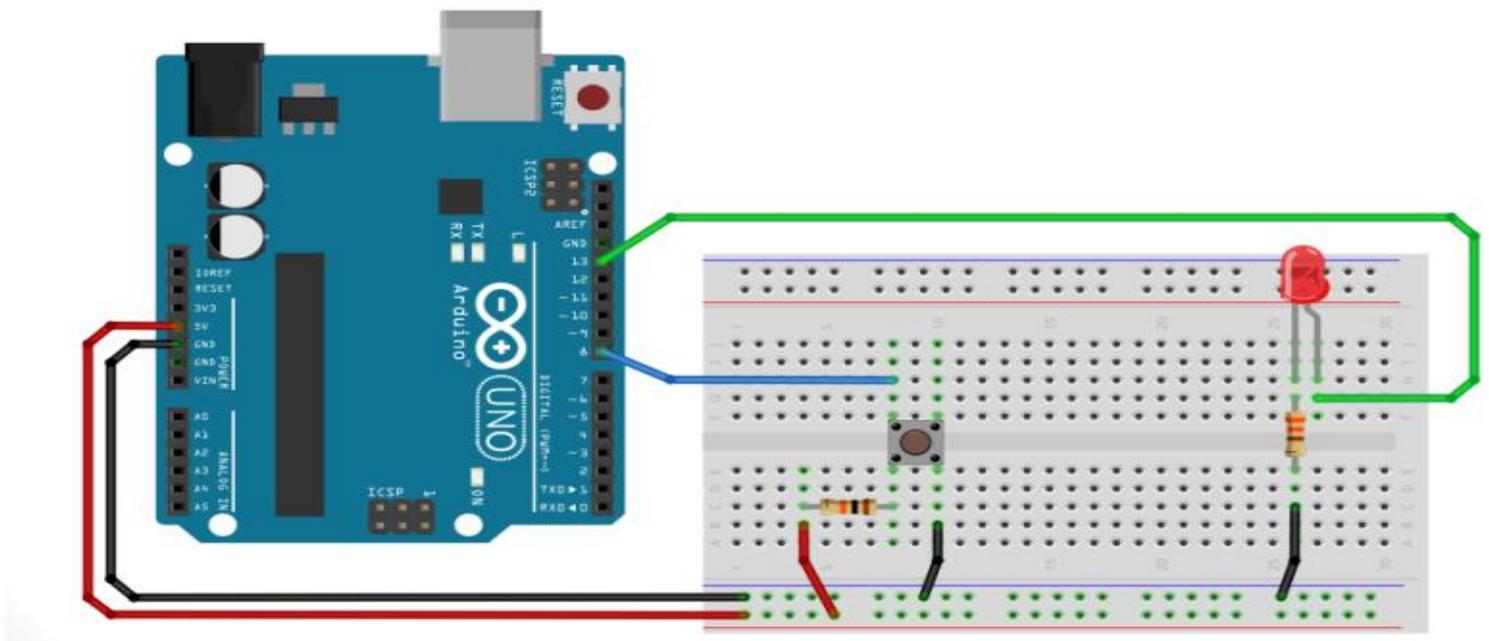
ATELIER N° 2 : LECTURE DES ENTRÉES NUMÉRIQUES (TOR)

Activité 1 : Commande d'une LED en mode Interrupteur

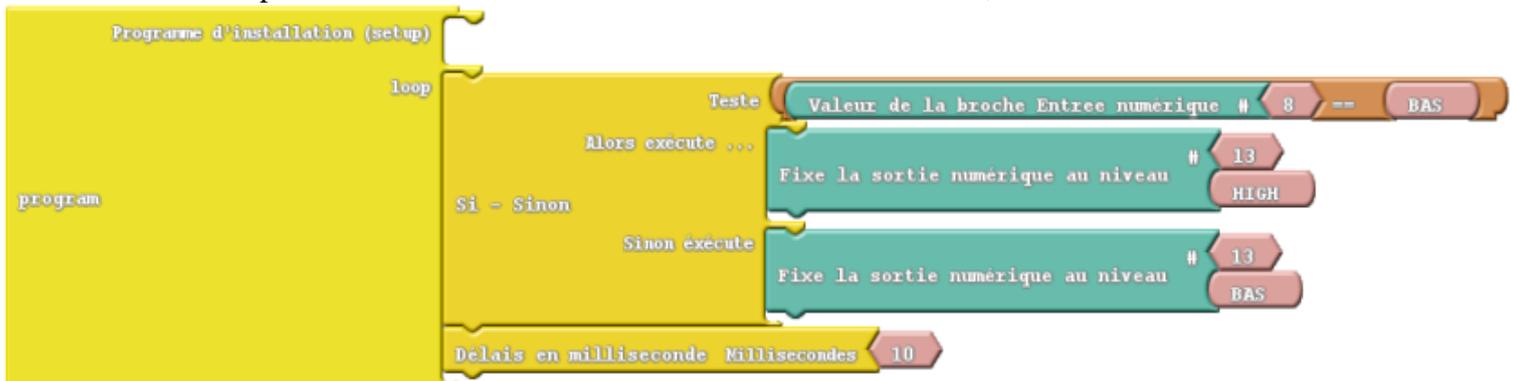
Schéma Electrique



Montage du Circuit



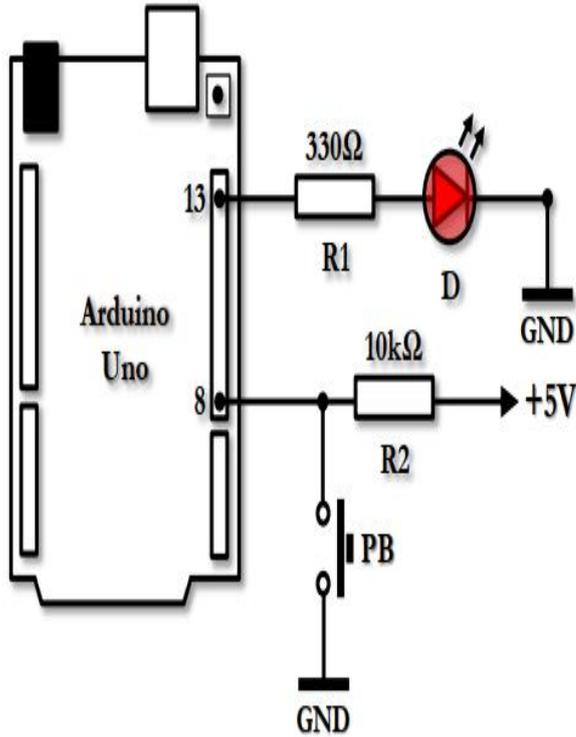
- 1) **Programmez** la carte **Arduino** afin d'**allumer** ou **éteindre** la diode **LED** reliée à la **broche 13** selon l'**état** du **bouton** poussoir relié à la **broche 8** : Enfoncé → LED Allumée ; Relâché → LED Eteinte



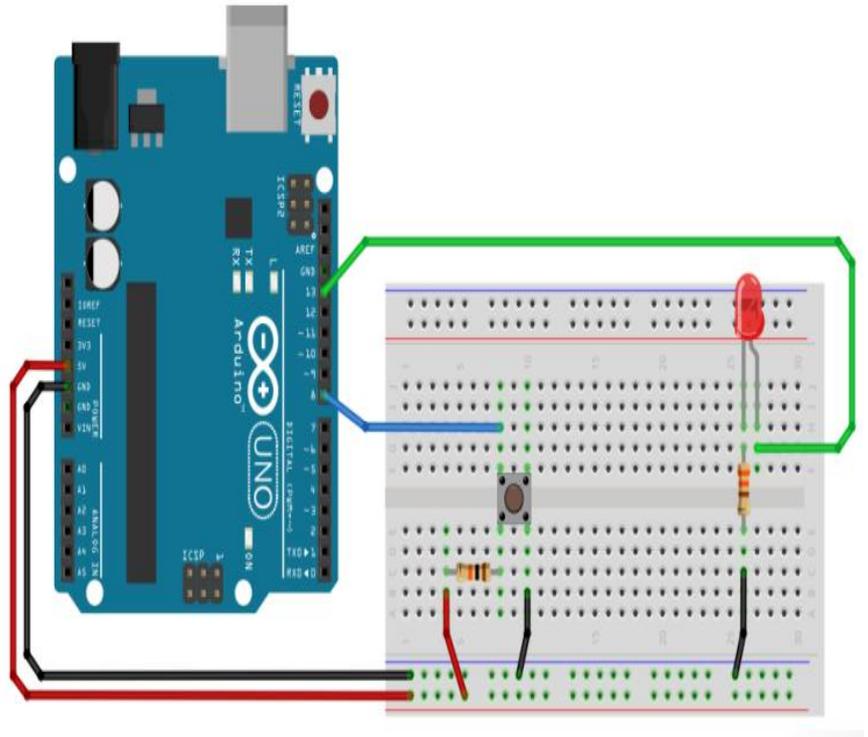
- 2) **Programmez** la carte **Arduino** afin d'**inverser** l'**état** de la **LED** reliée à la **broche 13** avec **chaque appui** sur le **bouton** poussoir relié à la **broche 8**.

Activité 2 : Commande d'une LED en mode Télerrupteur

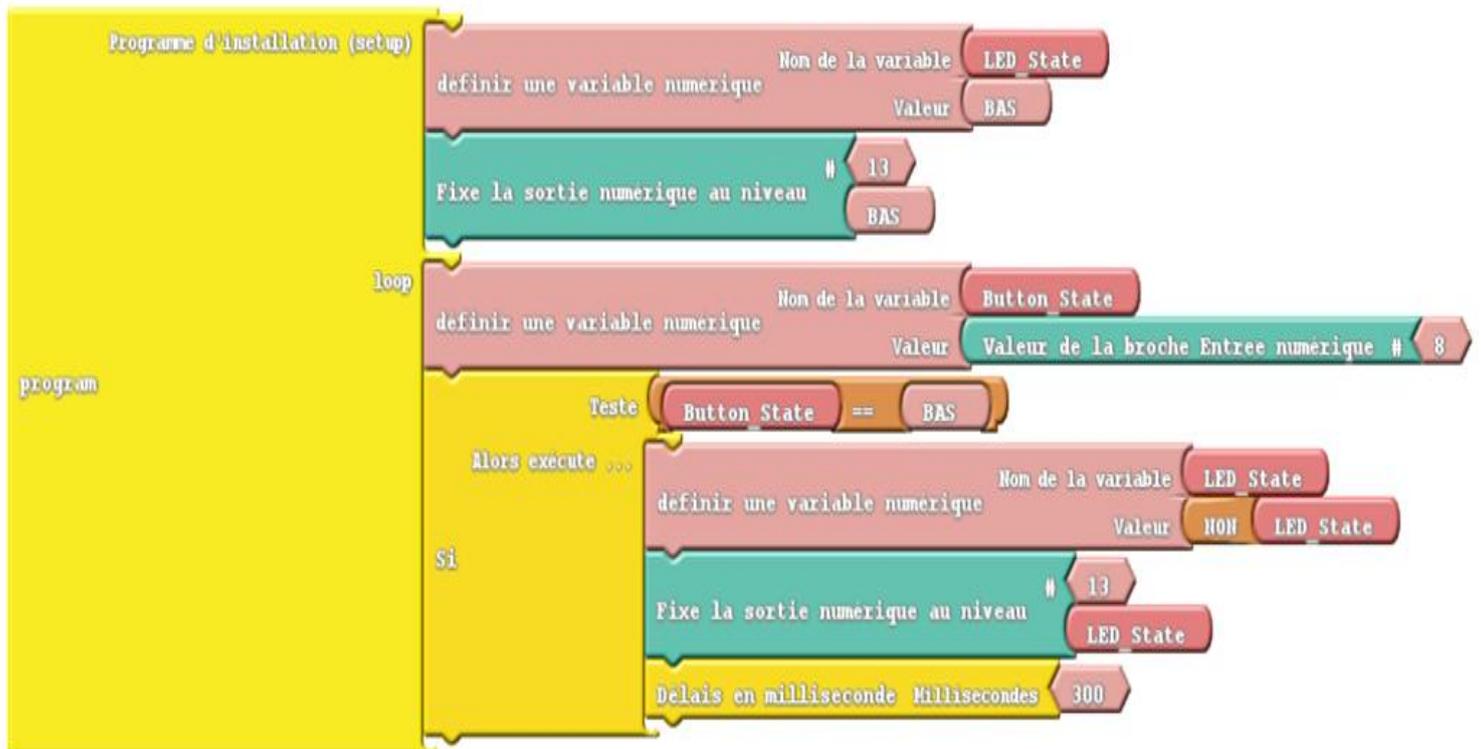
Schéma Electrique



Montage du Circuit

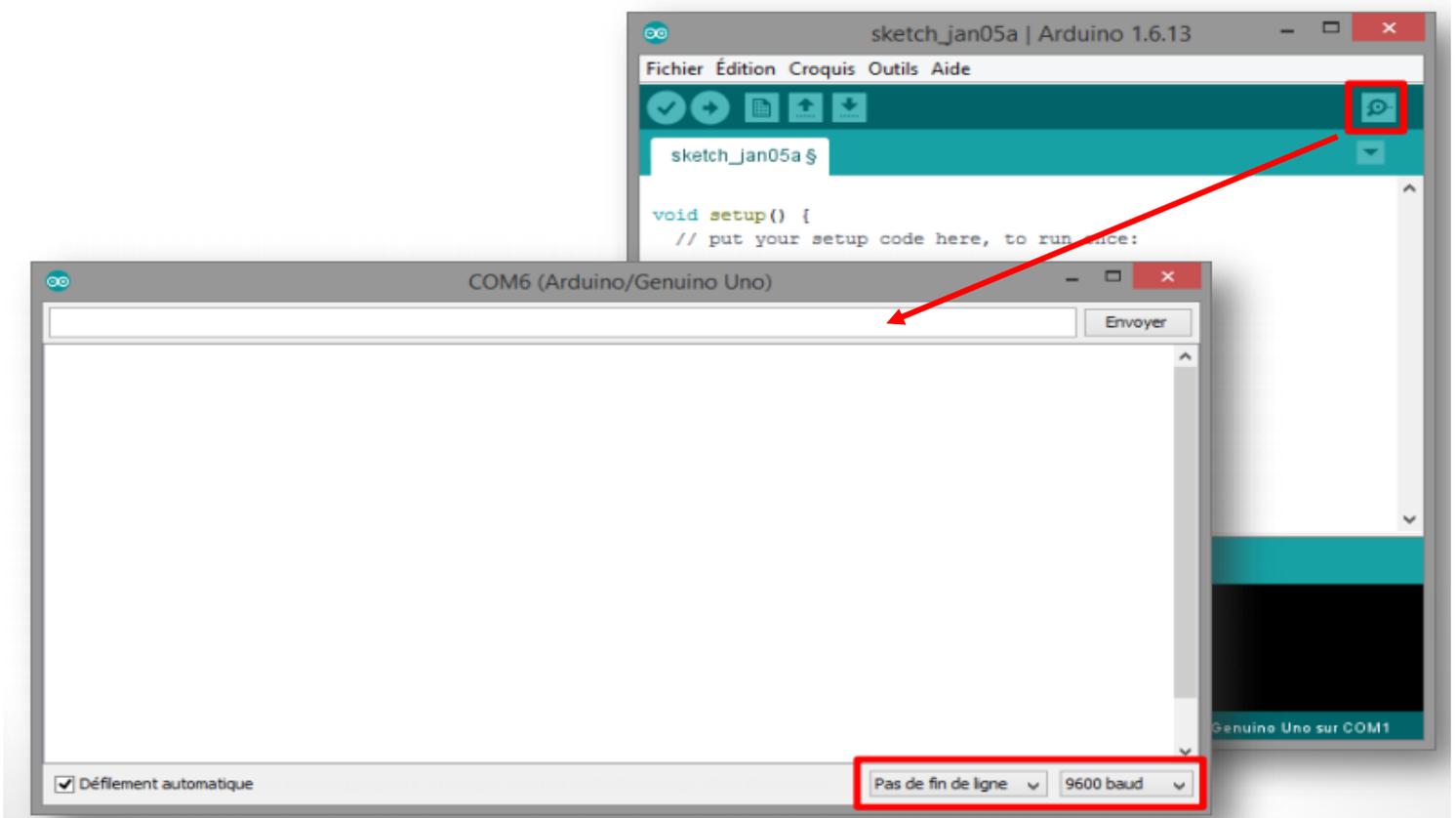


- 1) **Inverser** l'état de la **LED** avec chaque **appui** sur le **bouton**.
- 2) **Retrouver** le **même fonctionnement** en **stockant** l'état du **bouton** dans une **variable numérique** appelé « **Boutton_State** ».



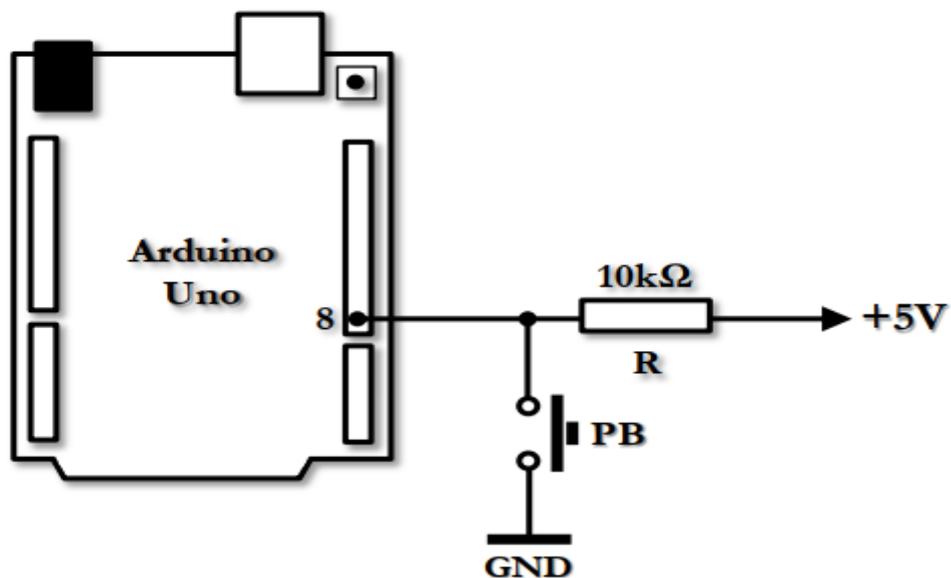
ATELIER N° 3 : UTILISATION DE LA COMMUNICATION SÉRIE

Moniteur Série

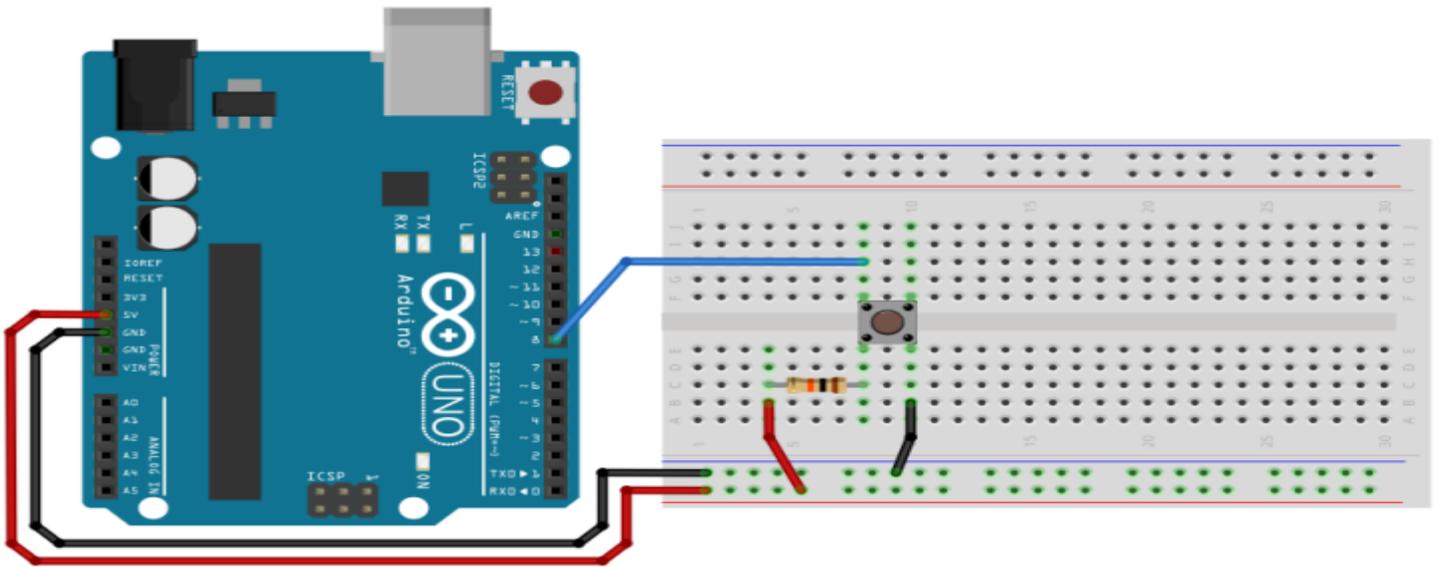


Activité 1 : Emission de données sur le port série

Schéma Electrique



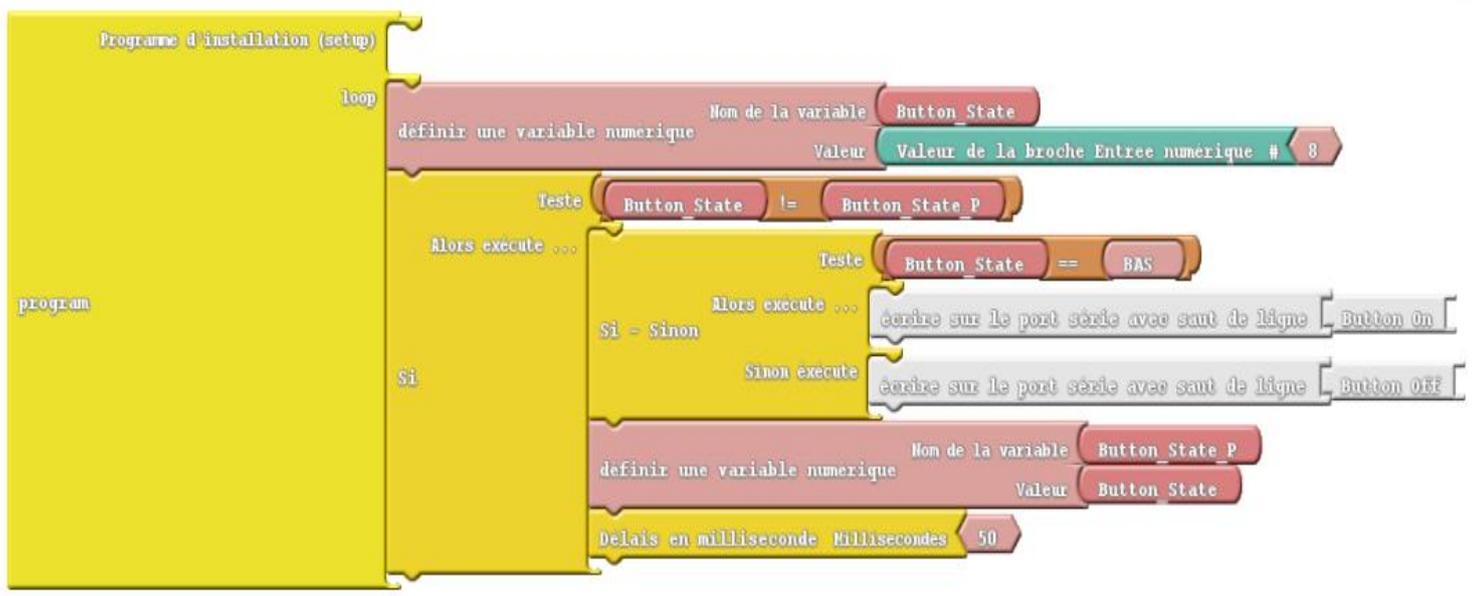
Montage du Circuit



1) **Programmez** la carte **Arduino** afin d'**envoyer** chaque **seconde**, sur le port **série**, « **Button On** » ou « **Button Off** » selon l'**état** du **bouton** relié à la **broche 8**.



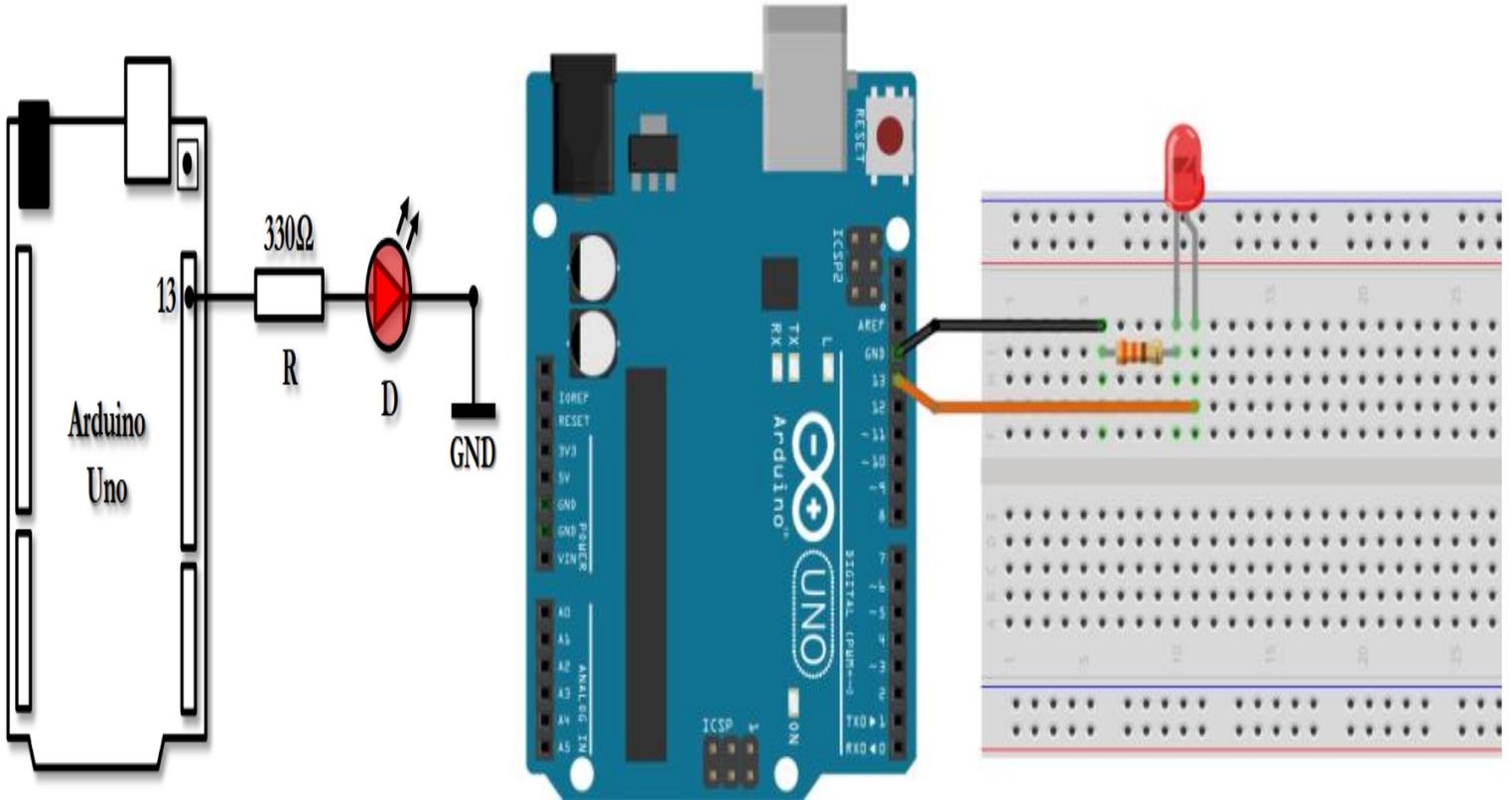
2) **Reprogrammez** la carte **Arduino** afin d'**envoyer**, sur le port **série**, l'**état** du **bouton** une **seule fois** après chaque **changement d'état**.



Activité 2 : Lecture des données reçues sur le port série

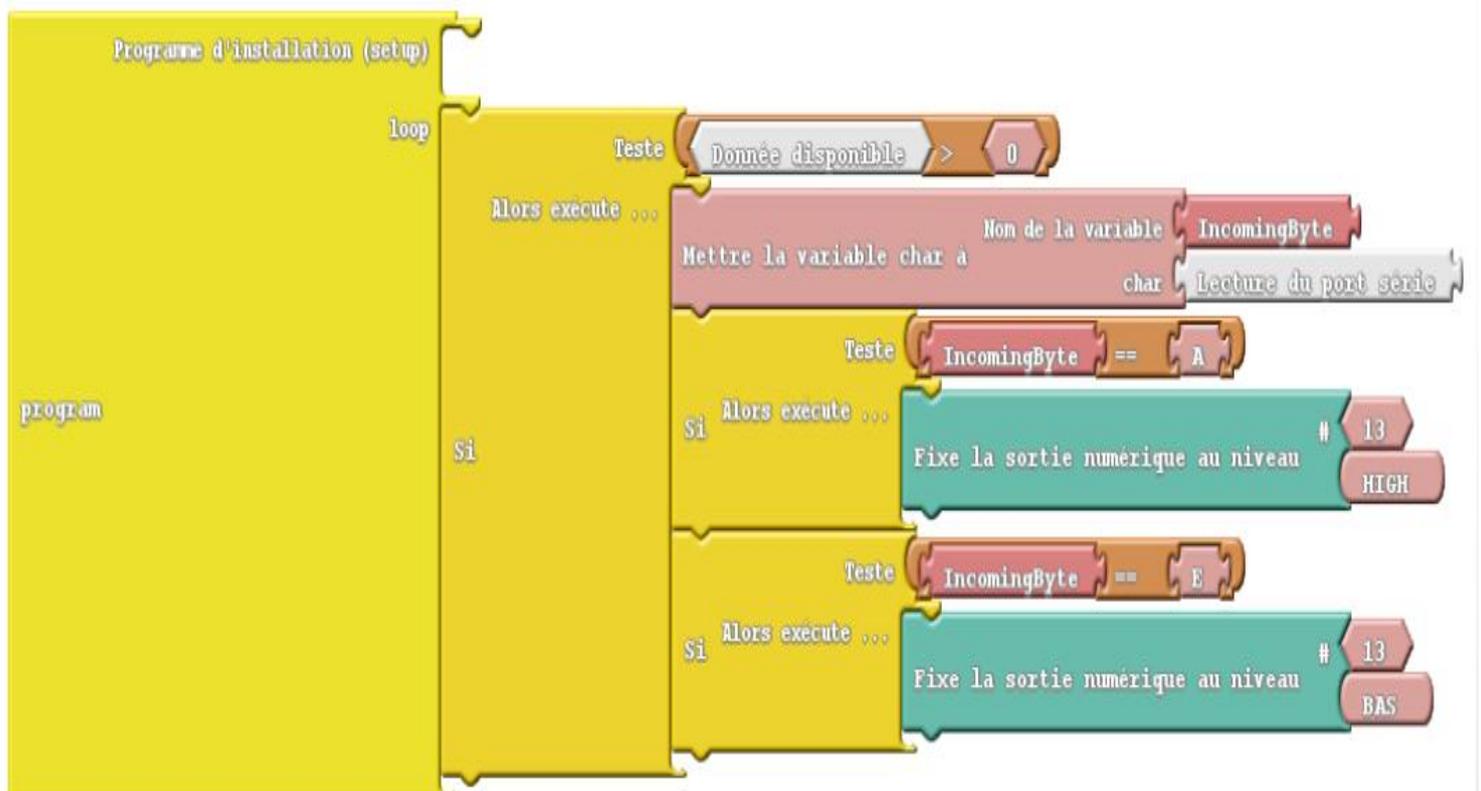
Schéma Electrique

Montage du Circuit



Programmez la carte **Arduino** afin d'**allumer** ou **éteindre** la LED reliée à la **broche 13** selon le **caractère reçu** sur le port série :

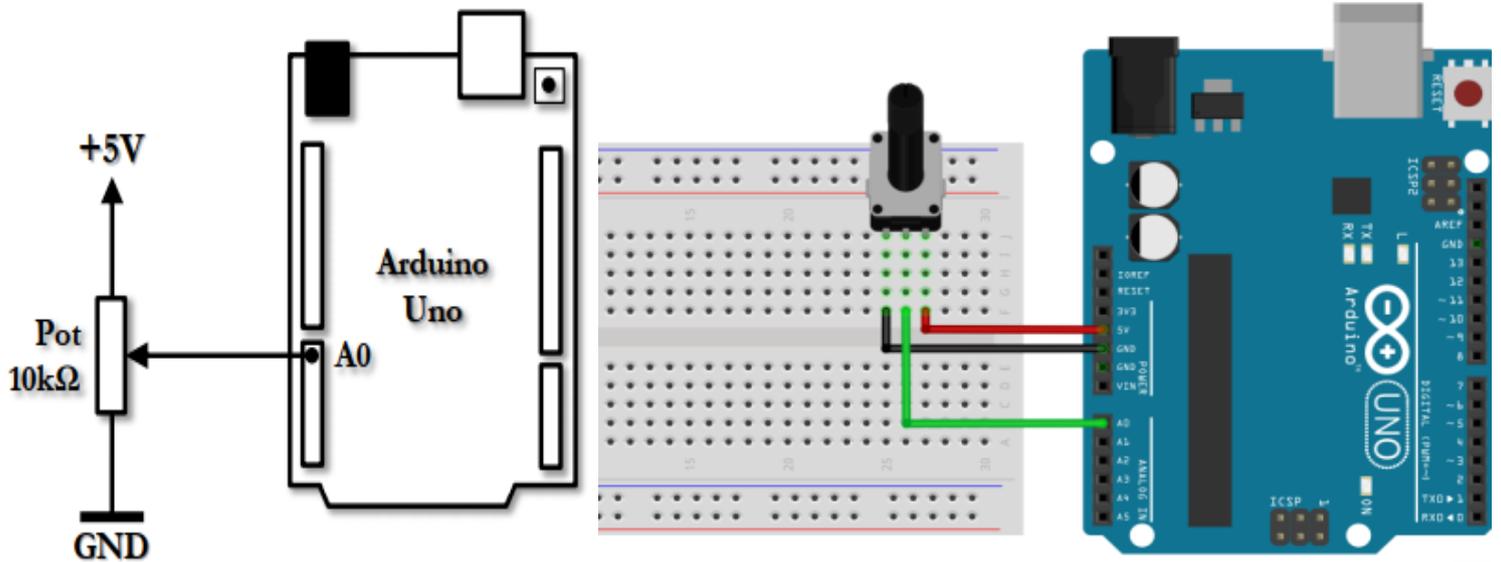
« **A** » → LED Allumée ; « **E** » → LED Eteinte



ATELIER N° 4 : LECTURE DES ENTRÉES ANALOGIQUES

Activité 1 : Mesure de tension analogique

Schéma Electrique Montage du Circuit



1) Programmez la carte Arduino afin d'afficher chaque seconde, sur le port série, la valeur numérique lue (entre 0 et 1023) sur le canal A0.



2) Modifiez le programme afin de convertir la valeur numérique lue sur le canal A0 en une tension et de l'afficher sur le port série.



3) Pour avoir un **affichage en virgule** de la tension convertie, **introduisez** les deux **changements suivants** dans le **code** :

a) **Changez** le **type entier** de la variable « Tension » à **réel** :

```
float_ABVAR_2_Tension = 0 ;
```

b) L'opération de calcul de la tension doit être une **opération réelle** en forçant l'**une** de ses **opérandes de type réel** :

```
_ABVAR_2_Tension = ( ( _ABVAR_1_Mesure * 5.0 ) / 1023 ) ;
```

The image shows two windows from the Arduino IDE. The left window is the code editor for a file named 'TP4_1_Tension_analogique'. The code is as follows:

```

int_ABVAR_1_Mesure = 0 ;
float_ABVAR_2_Tension = 0 ;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  _ABVAR_1_Mesure = analogRead(0) ;
  _ABVAR_2_Tension = ( ( _ABVAR_1_Mesure * 5.0 ) / 1023 ) ;
  Serial.print("Mesure :");
  Serial.print(_ABVAR_1_Mesure);
  Serial.println();
  Serial.print("Tension :");
  Serial.print(_ABVAR_2_Tension);
  Serial.println();
  Serial.print("-----");
  Serial.println();
  delay( 1000 );
}

```

The right window is the serial monitor for 'COM4 (Arduino/Genuino Uno)'. It displays the output of the code, showing the measured value and the converted voltage:

```

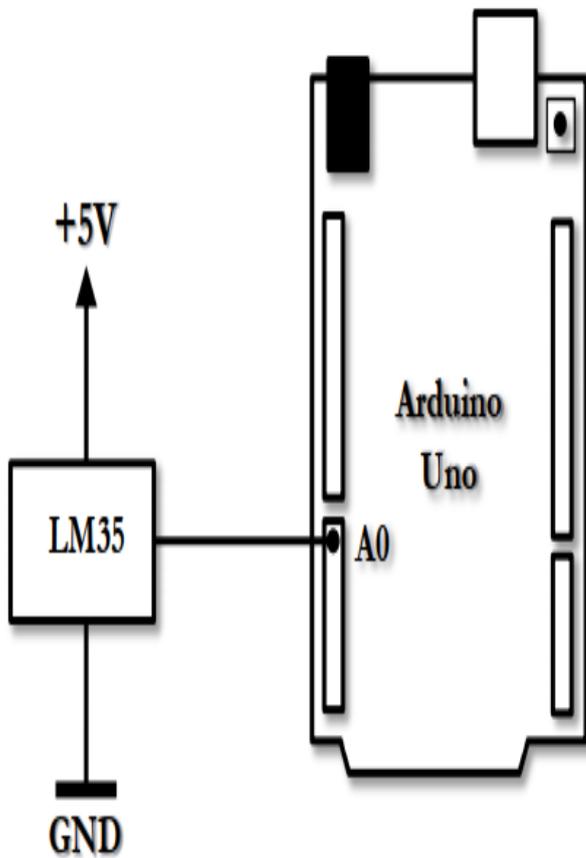
Tension :0.00
-----
Mesure :36
Tension :0.18
-----
Mesure :178
Tension :0.87
-----
Mesure :316
Tension :1.54
-----
Mesure :439
Tension :2.15
-----
Mesure :489
Tension :2.39
-----

```

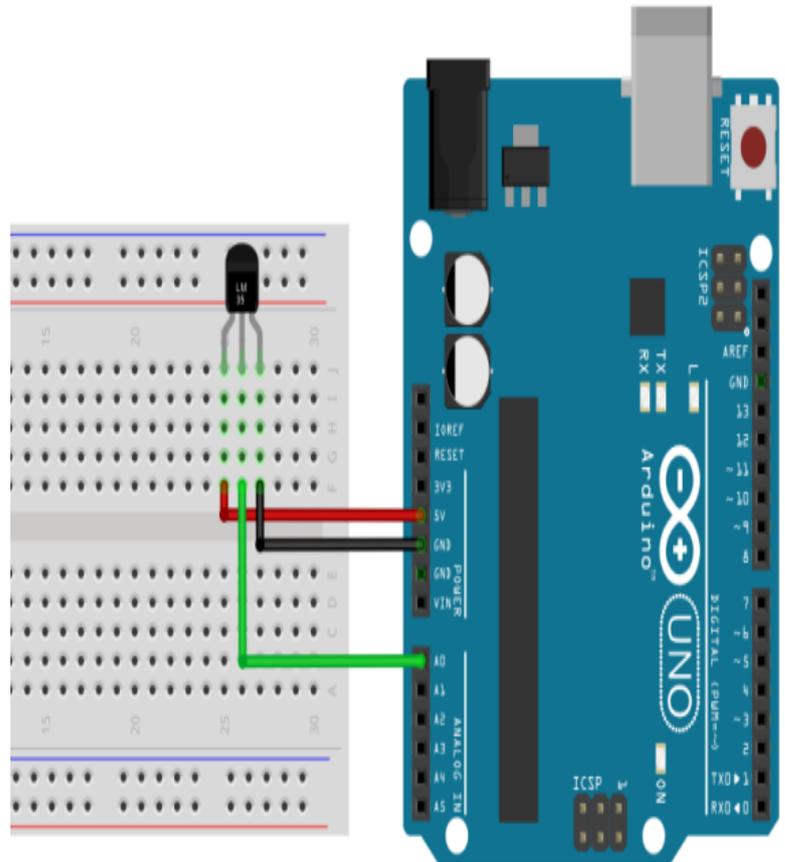
The serial monitor also shows 'Défilement automatique' checked and '9600 baud' selected.

Activité 2 : Mesure de température par le LM35

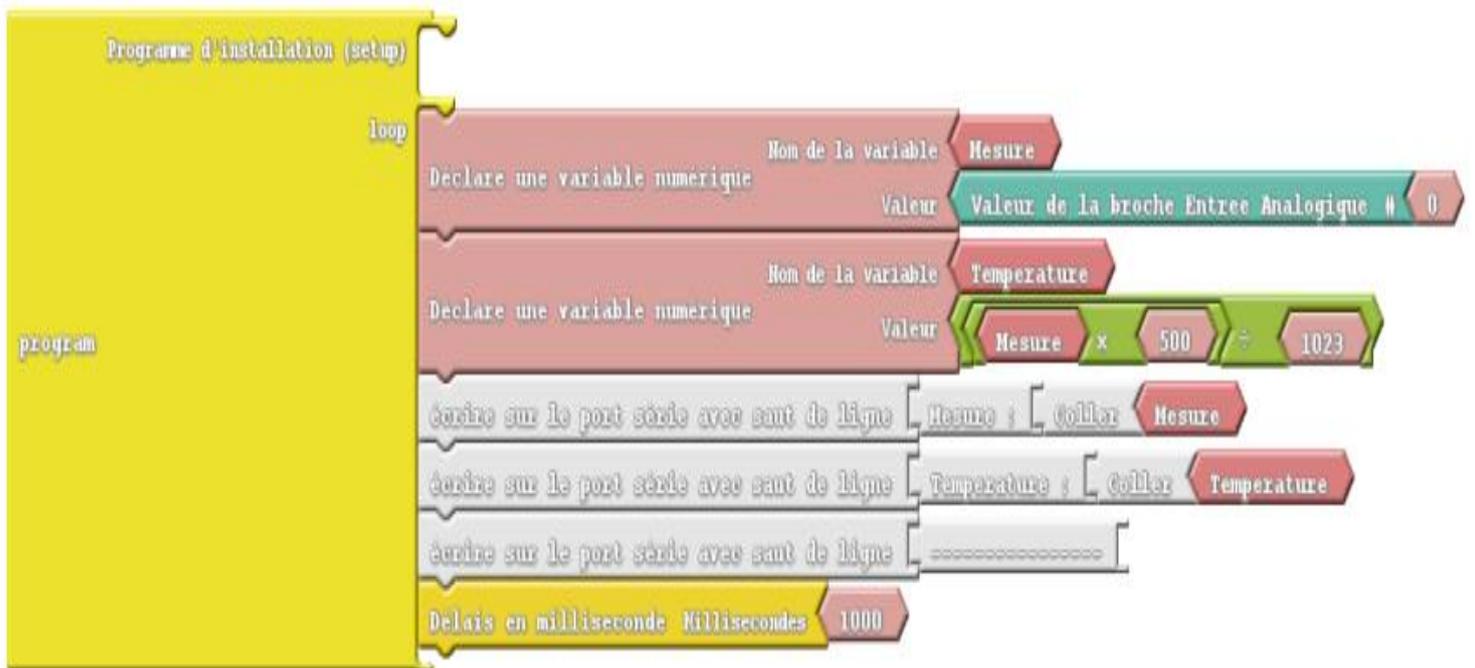
Schéma Electrique



Montage du Circuit



- 1) **Programmez** la carte **Arduino** afin d'**afficher** chaque **seconde**, sur le port **série**, la **valeur numérique** de la température lue sur le **canal A0**.

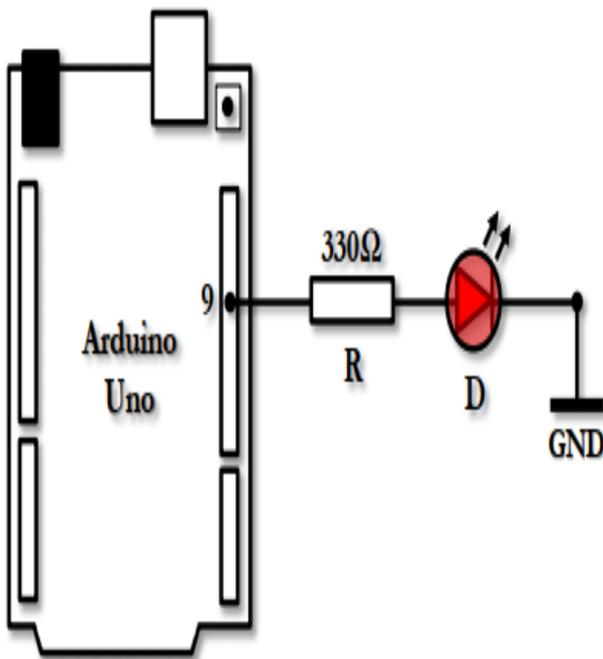


- 2) **Changez** le **type** de la variable « **Température** » en **réel** en **introduisant** les **mêmes changements** de l'activité précédente

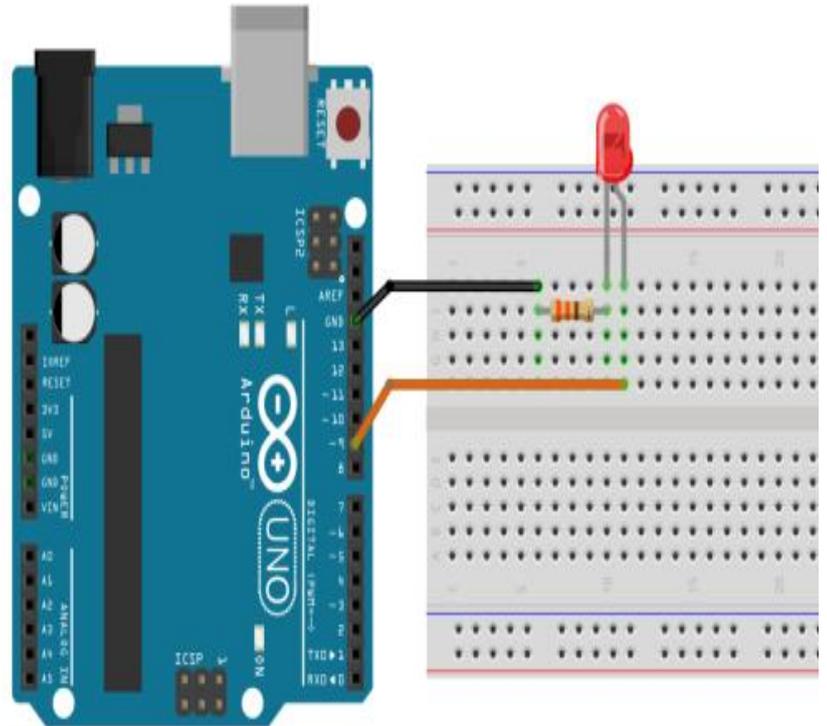
ATELIER N° 5 : GÉNÉRATION D'UN SIGNAL PWM

Activité 1 : Génération d'un signal PWM

Schéma Electrique



Montage du Circuit

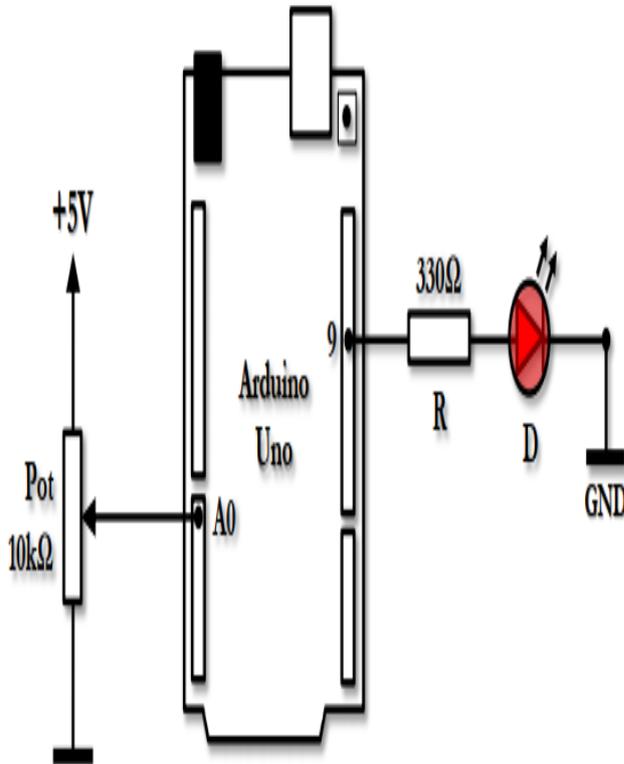


Programmez la carte **Arduino** afin d'allumer la **LED** reliée à la **broche 9** avec un **signal PWM** de **rapport cyclique 100%** puis **50%** et **25%**. Prévoyez un **délai d'une seconde** entre **chaque valeur**.

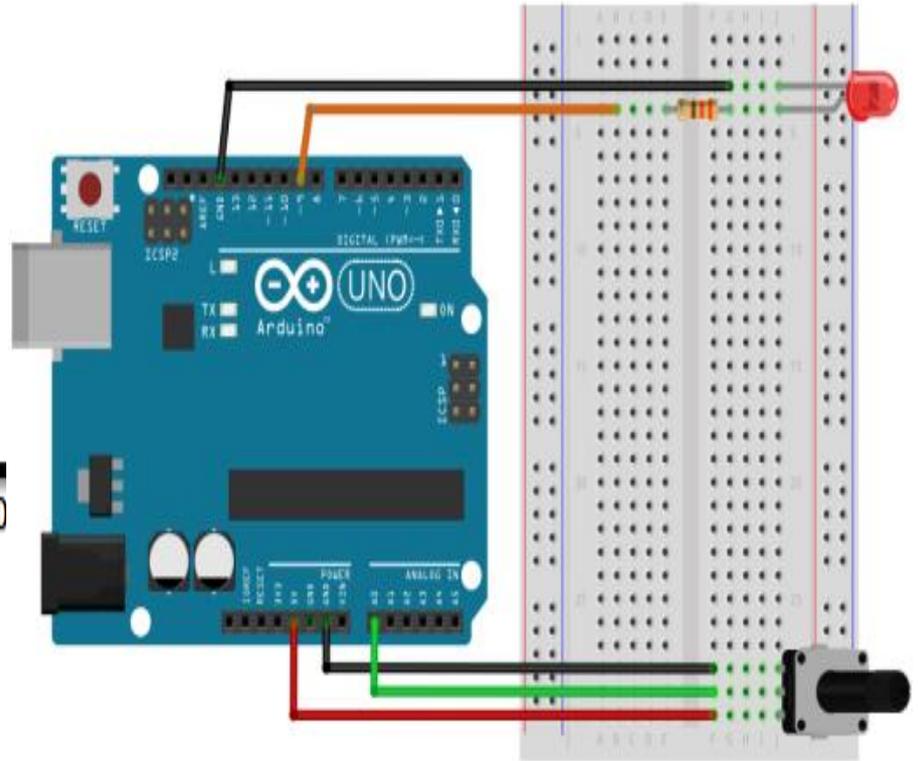


Activité 2 : Réglage de luminosité d'une LED via un potentiomètre

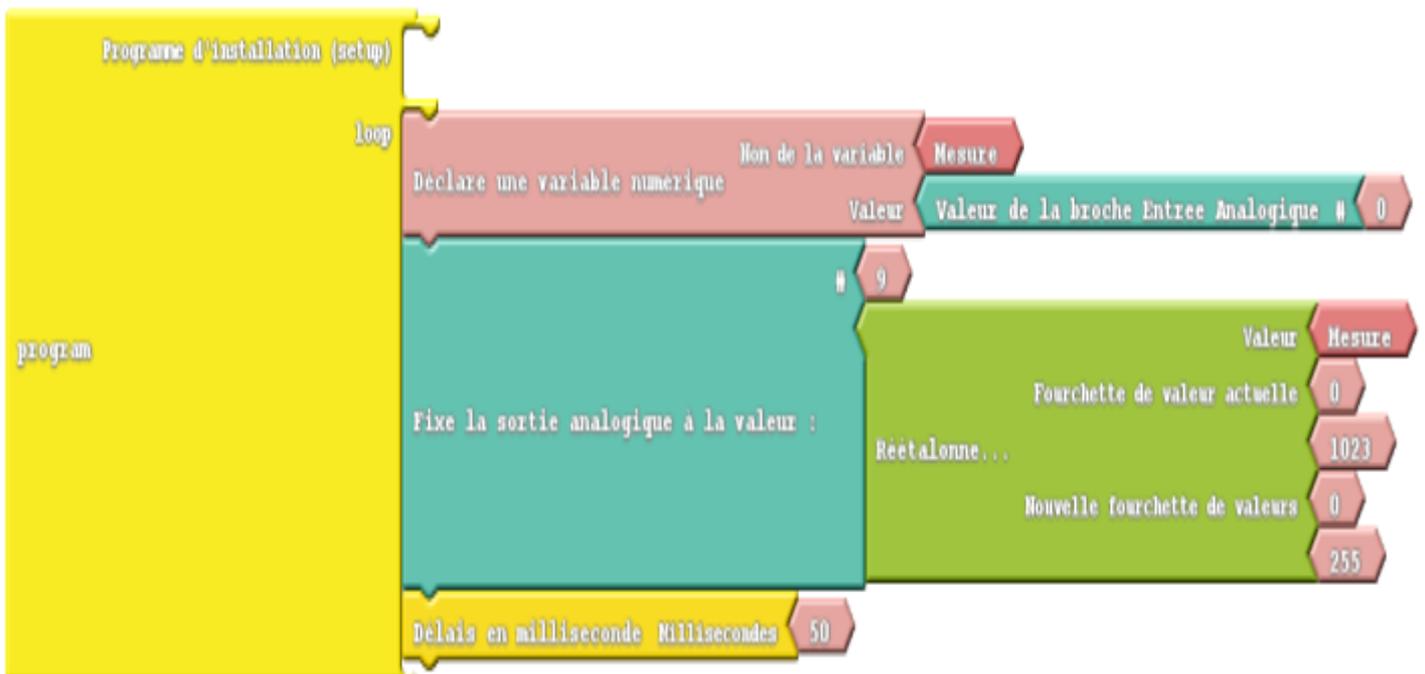
Schéma Electrique



Montage du Circuit



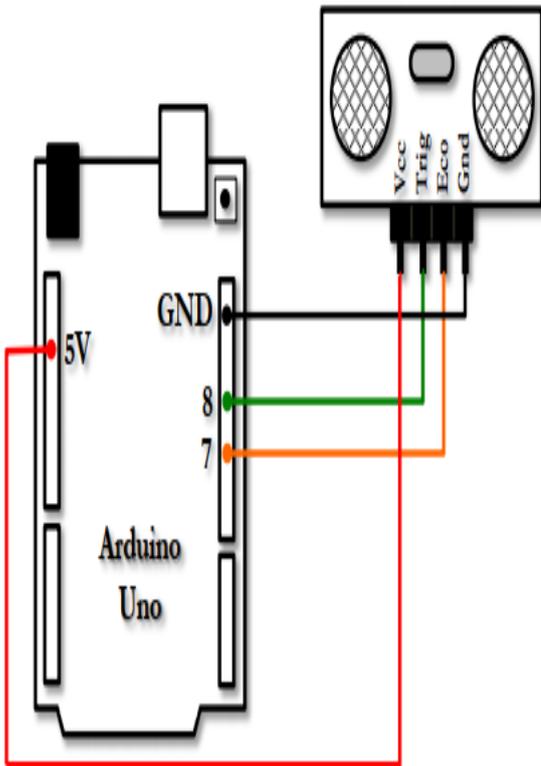
Programmez la carte **Arduino** afin d'**ajuster** la **luminosité** de la **LED** reliée à la **broche 9** selon la **position** du **potentiomètre**.



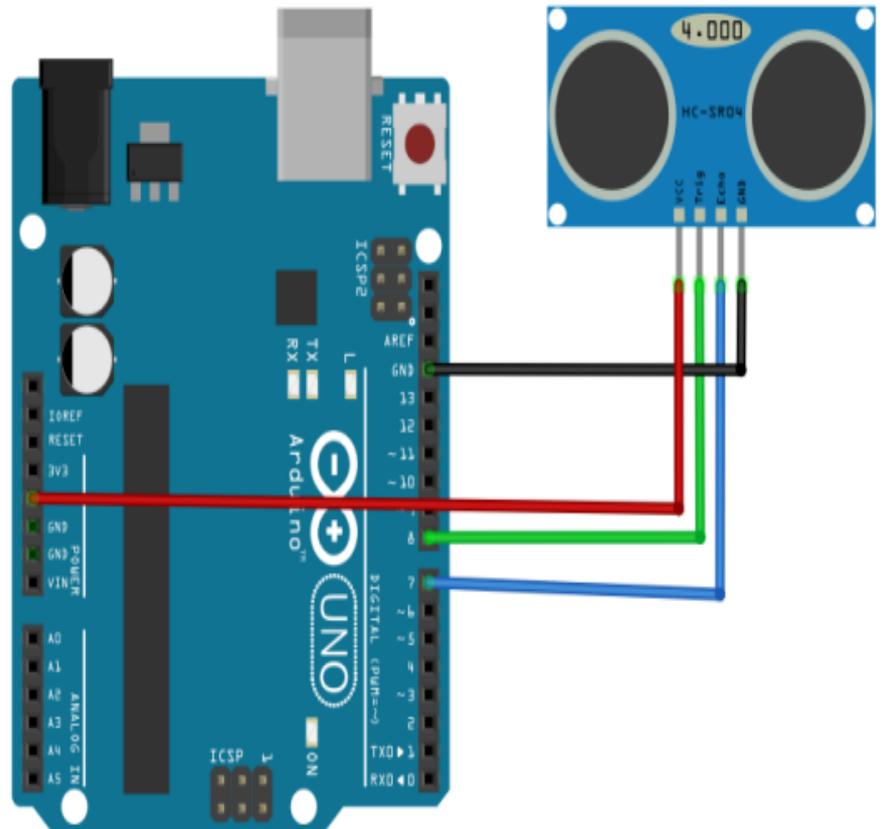
ATELIER N° 6 : UTILISATION DU MODULE ULTRASON HC-SR04

Activité 1 : Mesure de distance

Schéma Electrique



Montage du Circuit

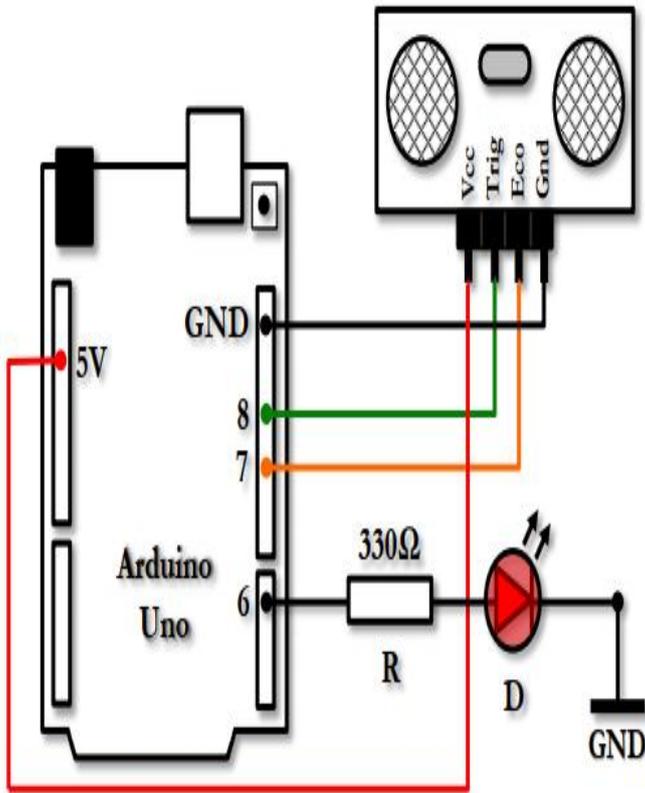


Programmez la carte **Arduino** afin de **mesurer** et **afficher** chaque **seconde** la **distance** en **centimètre** entre le **module ultrason** et n'importe quel **obstacle distant** de **moins de 1 mètre**.

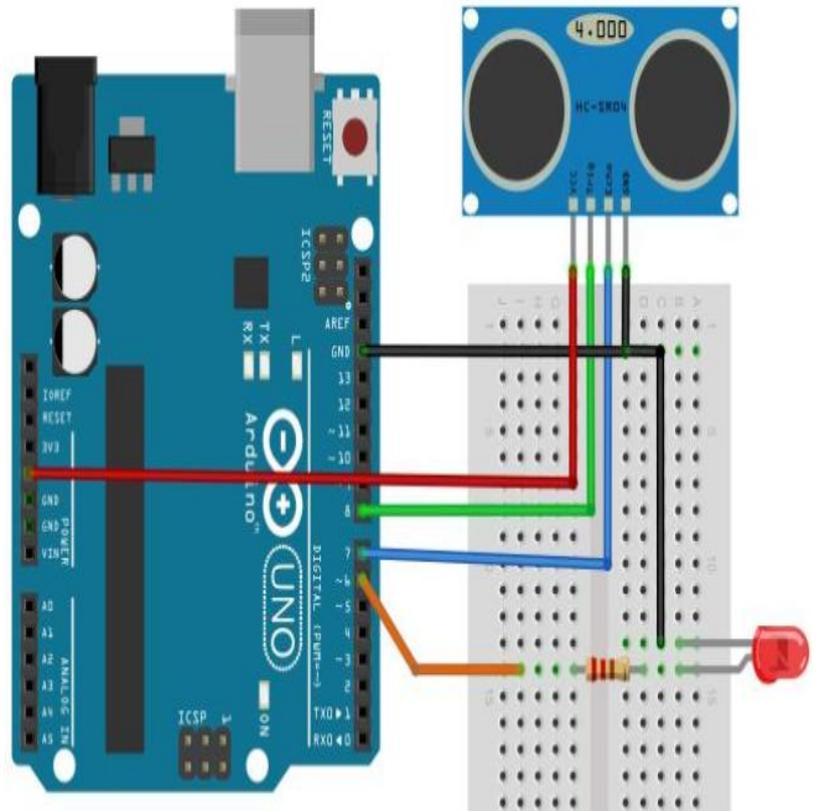


Activité 2 : Détection d'obstacles

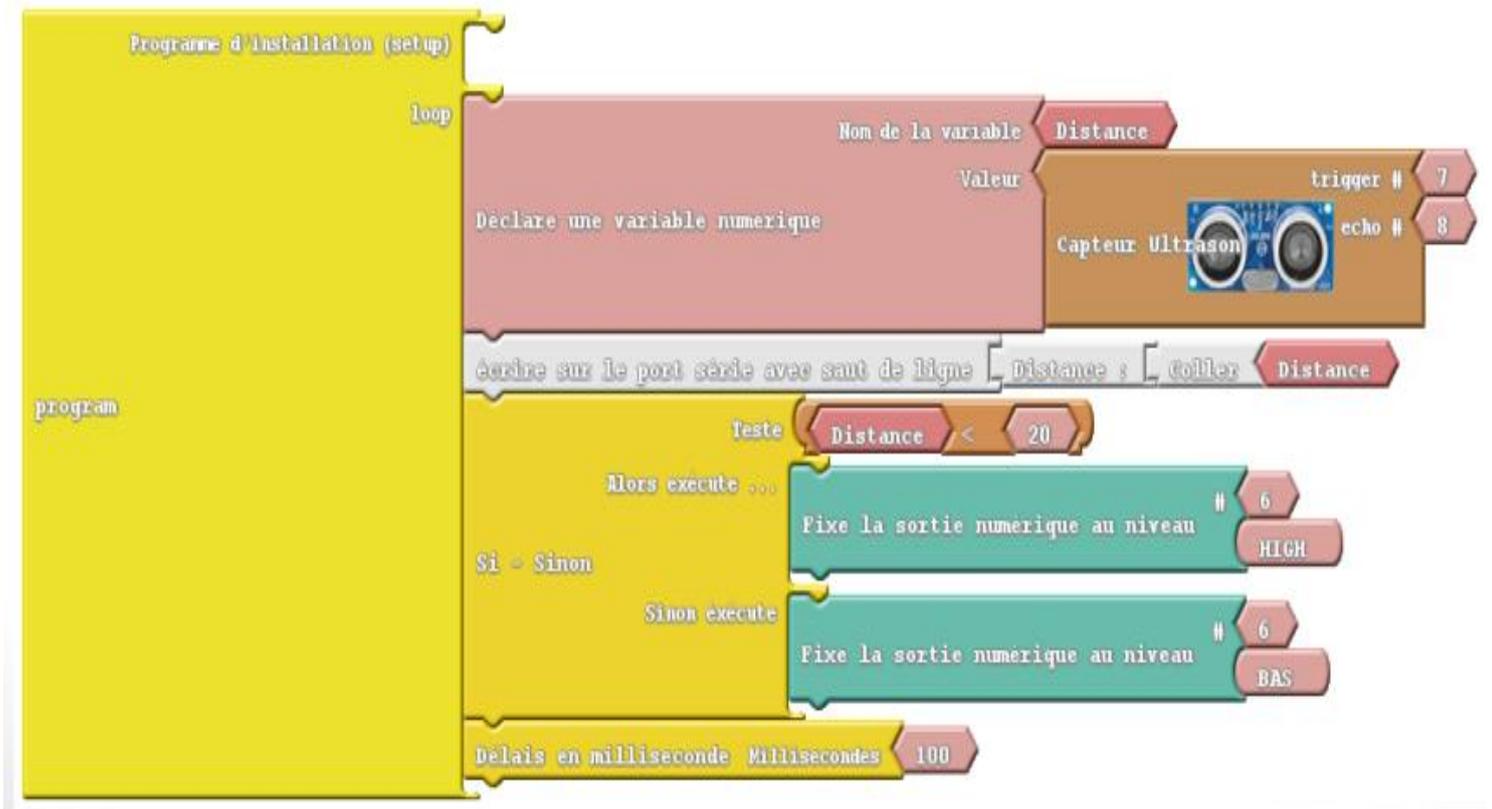
Schéma Electrique



Montage du Circuit



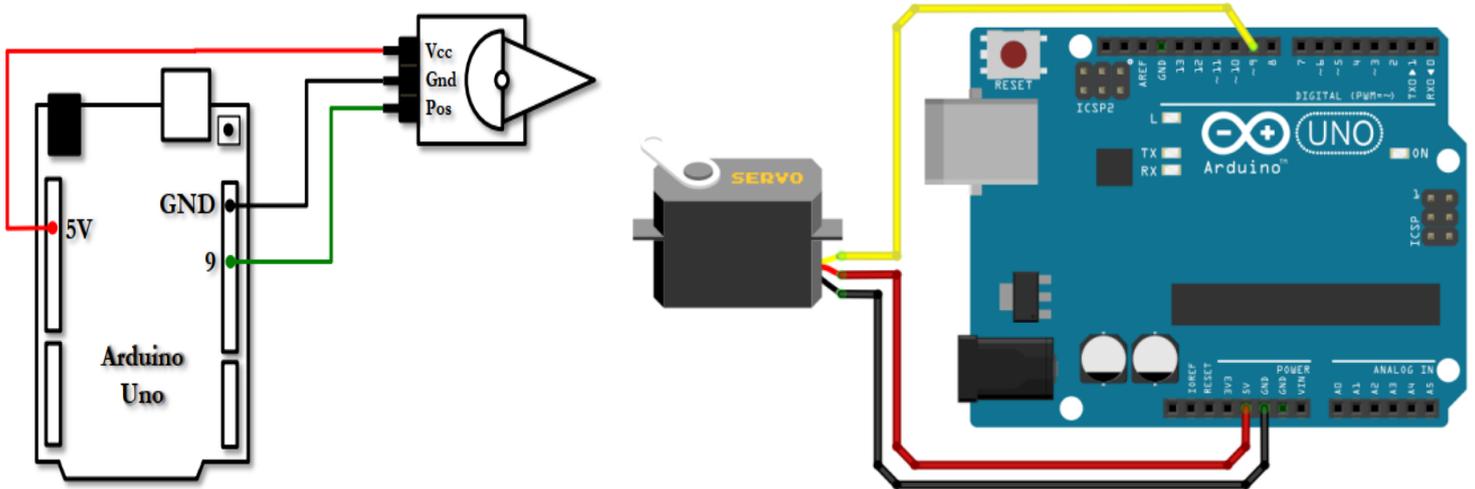
Programmez la carte **Arduino** afin d'**allumer** la **LED** reliée à la **broche 6** s'il y aurait un **obstacle** de **distance inférieure à 20 cm**.



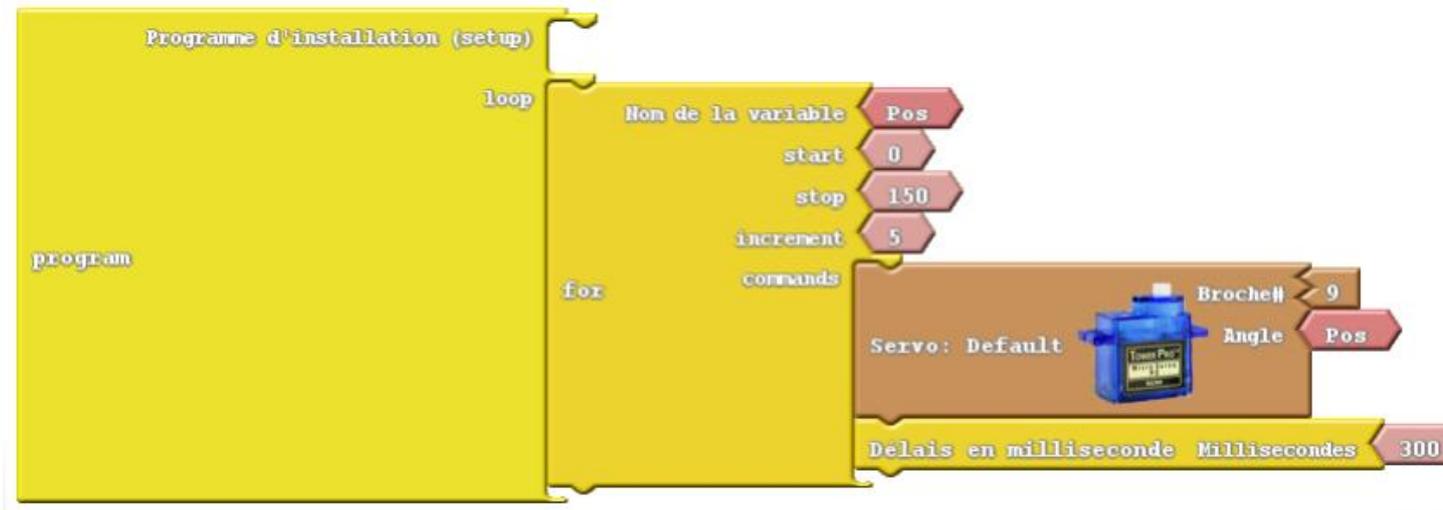
ATELIER N° 7 : COMMANDE DE SERVOMOTEUR

Activité 1 : Atteinte d'une position angulaire

Schéma Electrique Montage du Circuit



Programmez la carte **Arduino** afin de **commander** le servomoteur, relié à la **broche 9**, à atteindre un **angle choisi**.



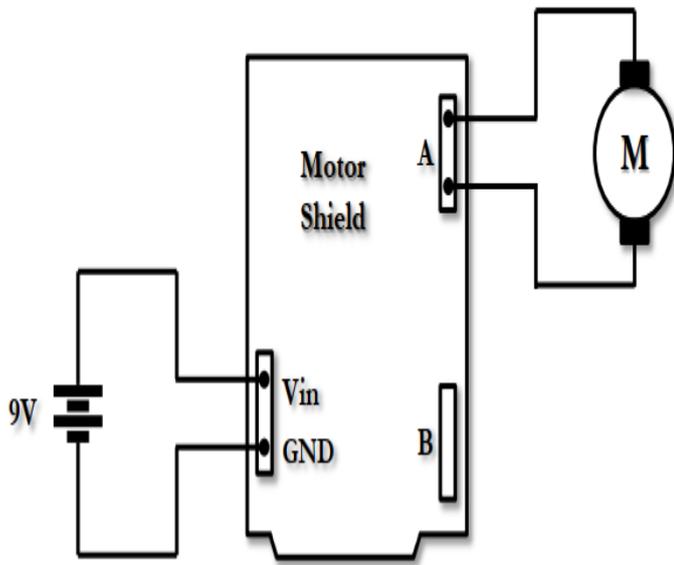
la carte **Arduino** afin de **commander** le servomoteur, relié à la **broche 9**, en **mode balayage de 180 degrés** par un **pas de 5°** chaque **300 ms**.



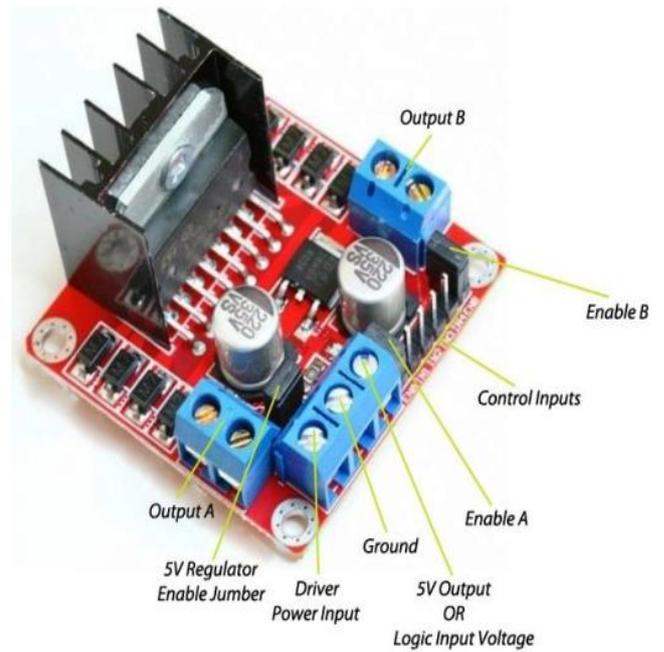
ATELIER N° 8 : COMMANDE DE MOTEUR À COURANT CONTINU

Activité 1 : Variation de sens et vitesse

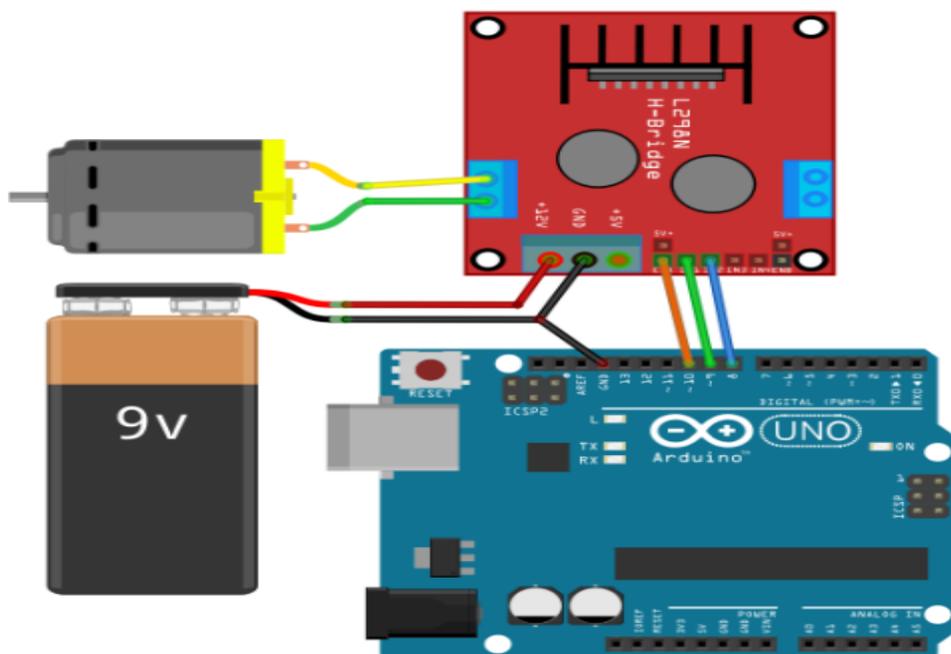
Schéma Electrique



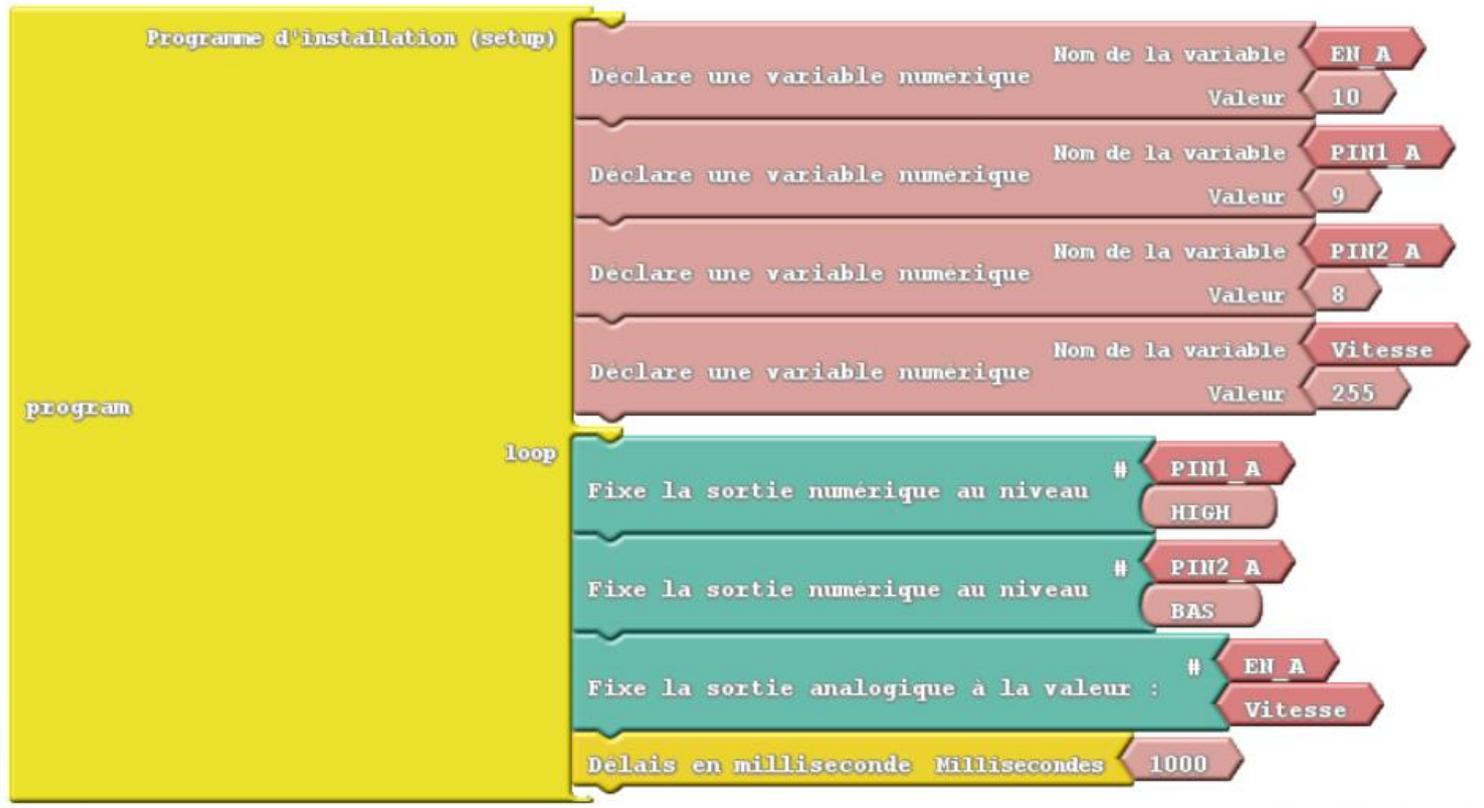
Module L298 Motor driver



Montage du Circuit



1) **Programmez** la carte **Arduino** afin de **commander** le moteur à **tourner** dans le **sens direct** à **pleine vitesse**.

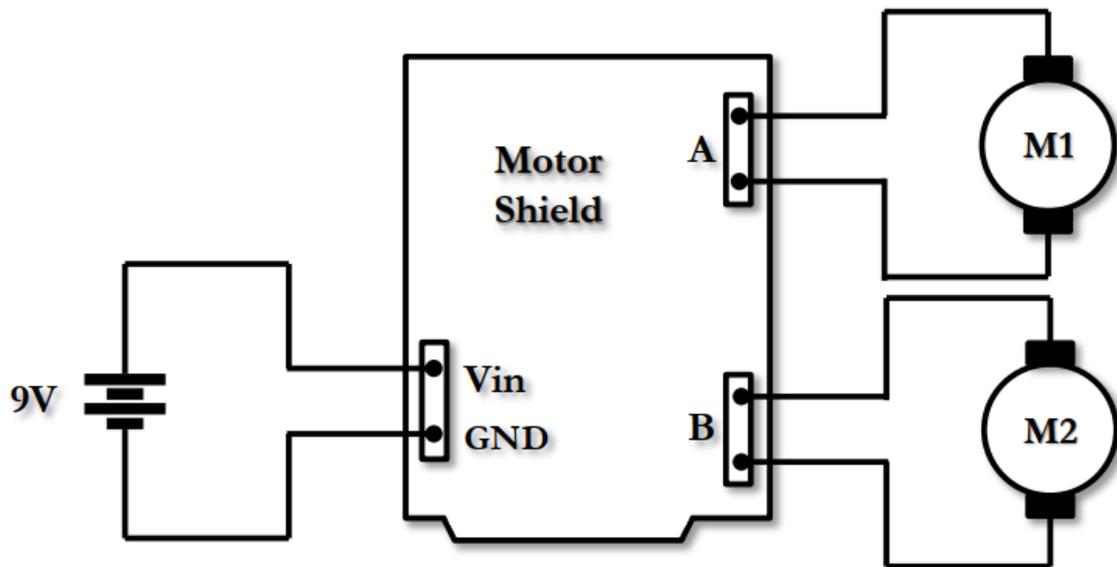


2) **Modifiez** le programme afin de **commander** le moteur à **tourner** dans le **sens indirect** à **demi vitesse**.

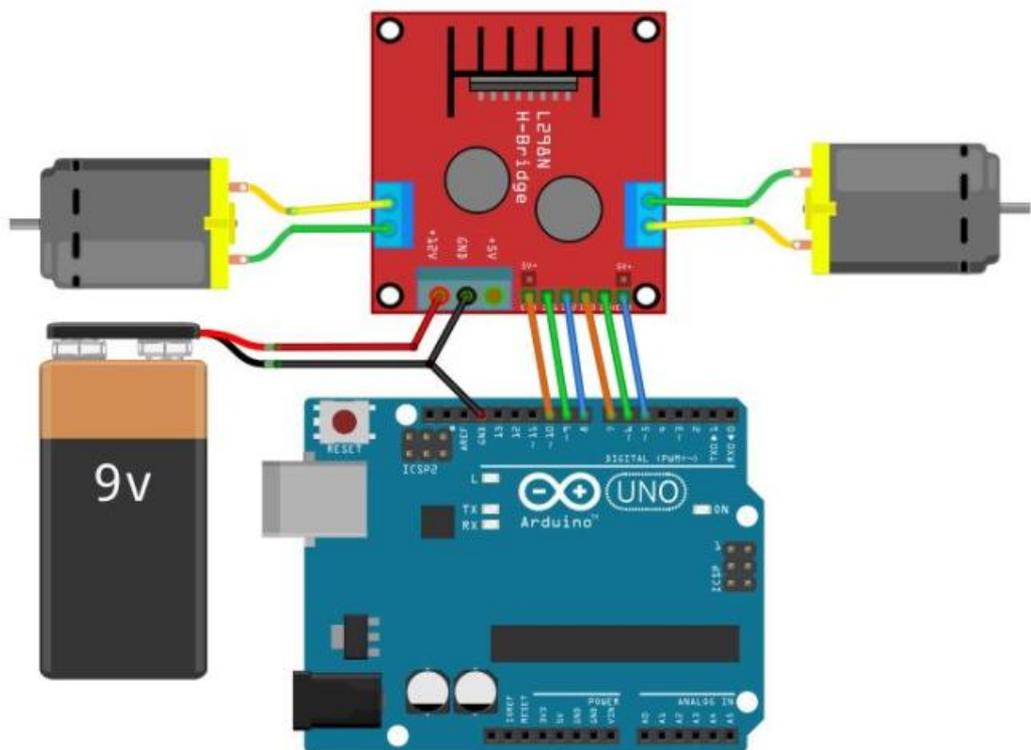


Activité 2 : Commande de Mouvements du robot

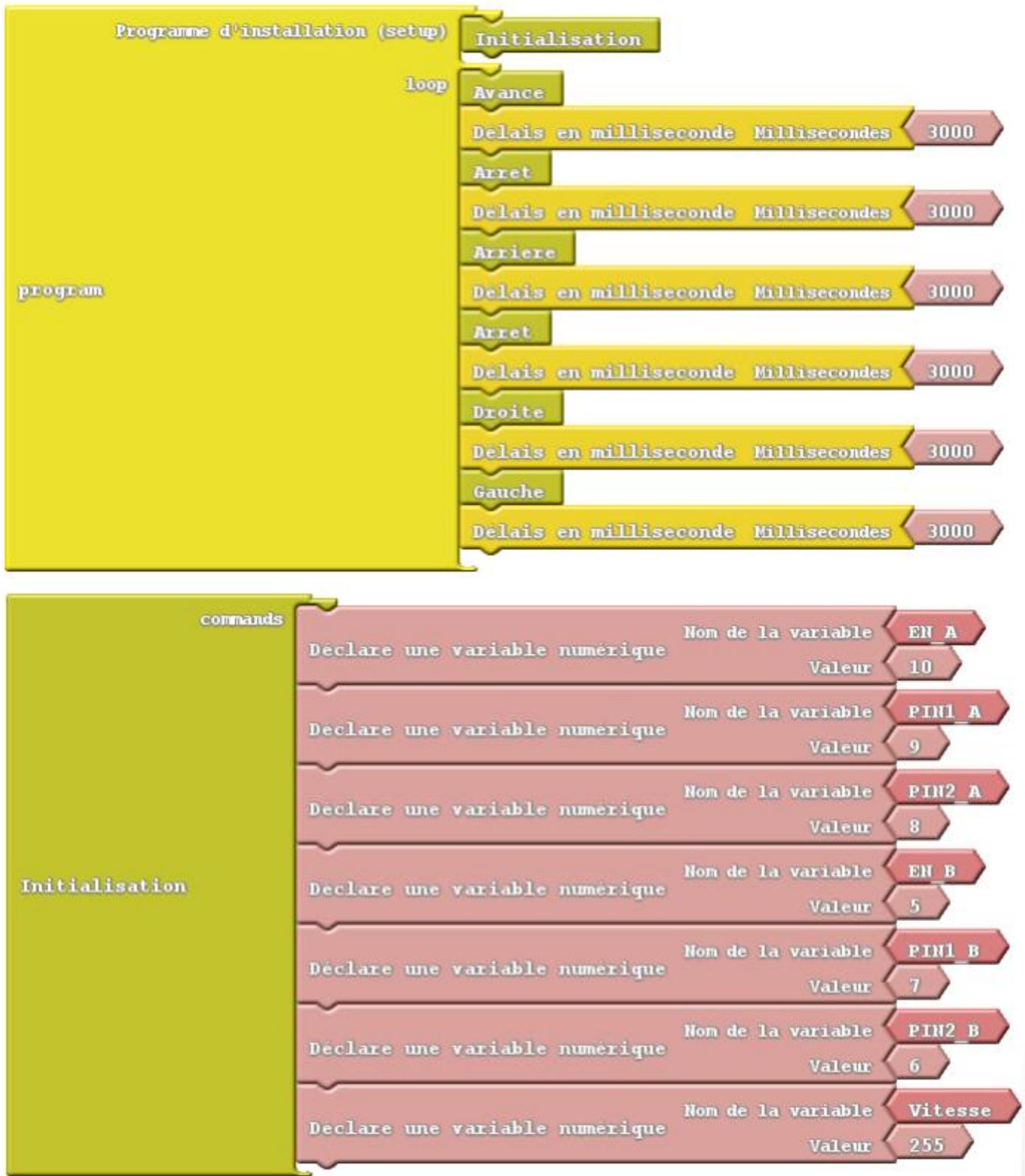
Schéma Electrique



Montage du Circuit



Programmez la carte **Arduino** afin de **commander** le robot à **effectuer** les **mouvements** suivants chacun pendant **3 secondes** : Avance – Arrêt – Recule – Arrêt – Droite – Gauche







Initiation à Arduino (Ardublock)

Réalisée par : - Mr. Khaled TATOUH

Encadrement : - Mr. Nasr BEN MOHAMED
- Mr. Moez MALLOUKI

Formation réalisée et animée par Mr. Khaled TATOUH et encadrée par Mr. Nasr BEN MOHAMED et Mr. Moez MALLOUKI dans le cadre de la formation continue dans le laboratoire génie électrique au sein du lycée Ibn Sina Kébili.

(Le 05, 06 et 07/12/2019)