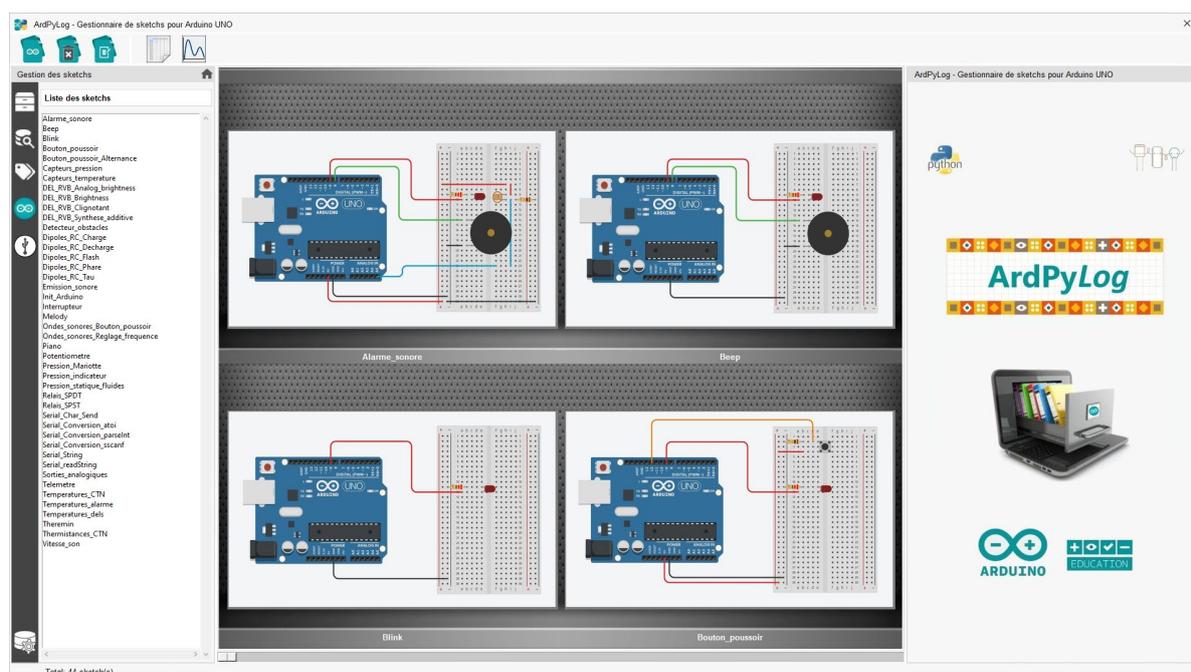


1. ArdPyLog, Qu'est-ce que c'est ?



Fenêtre d'accueil d'ArdPyLog

ArdPyLog est un logiciel programmé en Python 3 afin de faciliter la gestion des sketches pour les cartes Arduino UNO, dans les lycées d'enseignement général et technologique.

Les sketches enregistrés dans la base de données d'**ArdPyLog** peuvent être téléversés dans la mémoire de l'Arduino Uno.

Il est possible d'associer, à chaque sketch, une image représentant le circuit électronique correspondant ainsi qu'un document descriptif au format pdf.

Les sketches peuvent être classés par catégories et une fonction de recherche est disponible pour retrouver rapidement vos codes...

Les données envoyées par l'Arduino, via la liaison série, comme les valeurs d'un capteur, sont enregistrables et exploitables.

Enfin, **ArdPylog** dispose d'un mode « **Acquisition** » qui transforme l'Arduino Uno en centrale d'acquisition grâce à l'utilisation de ses entrées analogiques. Les tensions mesurées sur les entrées analogiques sélectionnées sont acquises pendant la durée et le taux d'échantillonnage définis par l'utilisateur.

2. Installation d'ArdPyLog

ArdPylog est disponible sous 2 versions :

- code source (nécessite un interpréteur Python)
- version portable (archive zip contenant l'exécutable et les modules dépendants)

2.1 Installation d'ArdPylog avec le code source

Python étant un langage de programmation interprété, pour faire fonctionner **ArdPylog**, il faut qu'un interpréteur **Python** soit installé.

ArdPylog nécessite au minimum la version 3.7 de Python et l'installation des bibliothèques suivantes :

- pyserial (pour la réception des données de l'Arduino Uno)
- matplotlib (pour tracer et visualiser des données sous formes de graphiques)
- numpy (pour le calcul scientifique)
- Pillow (pour le traitement des images)
- scipy (pour le traitement des données)
- PyMuPDF (pour l'affichage des documents PDF)
- keyboard (pour l'utilisation du clavier)

Une fois l'environnement de travail configuré et après avoir téléchargé puis décompressé le fichier "**ArdPyLog.zip**" qui contient tous les fichiers et dossiers nécessaires à son fonctionnement, **ArdPyLog** est démarré à l'aide du fichier "**Main.py**" situé dans le dossier principal "**ArdPyLog**" du programme.

Attention :

L'emplacement du dossier décompressé "**ArdPyLog**" n'a pas d'importance, mais tous les fichiers et dossiers contenus dans ce dossier ne doivent en aucun cas être modifiés ou déplacés.

Remarques :

Le dossier principal "**ArdPyLog**" du programme contient un dossier nommé "**venv**" dans lequel se situe un environnement virtuel de programmation avec les bibliothèques indispensables citées ci-dessus.

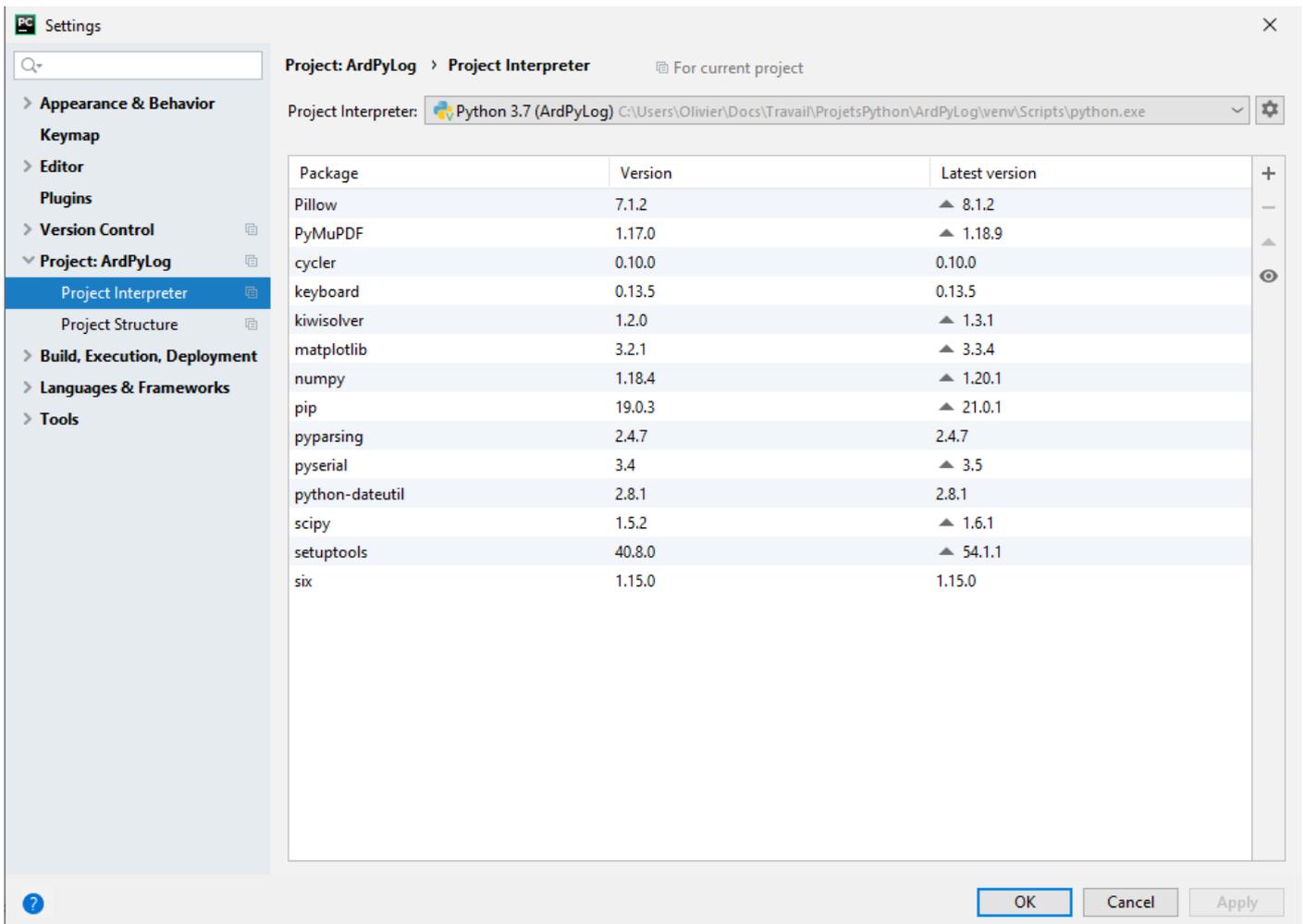
Nom	Modifié le	Type	Taille
.idea	11/03/2021 16:39	Dossier de fichiers	
__pycache__	11/03/2021 14:54	Dossier de fichiers	
Media	01/02/2021 14:55	Dossier de fichiers	
Sketchs	29/06/2020 11:02	Dossier de fichiers	
venv	01/07/2020 08:17	Dossier de fichiers	
AffichData.py	11/03/2021 14:25	Python File	11 Ko
AppliDef.py	10/03/2021 15:14	Python File	13 Ko
ArdPyLog.sql3	23/11/2020 11:19	Fichier SQ3	20 Ko
DescriptSketch.py	17/02/2021 11:18	Python File	11 Ko
EtagCircuits.py	11/03/2021 14:54	Python File	9 Ko
FindData.py	14/11/2020 16:50	Python File	3 Ko
FrmDef.py	16/12/2020 15:22	Python File	5 Ko
Graph.py	10/03/2021 17:44	Python File	124 Ko
Main.py	11/03/2021 14:49	Python File	72 Ko
ModeAcquisition.py	10/03/2021 16:04	Python File	42 Ko
ModifCircuit.py	15/11/2020 12:21	Python File	15 Ko
ModifData.py	11/03/2021 14:40	Python File	24 Ko
Parametres.py	28/01/2021 16:21	Python File	7 Ko
resolconfig	11/03/2021 10:29	Fichier	1 Ko
SerialMonitor.py	28/01/2021 16:18	Python File	11 Ko
SketchIDE.py	10/02/2021 15:03	Python File	18 Ko
Tableur.py	11/03/2021 10:35	Python File	57 Ko
Zoom.py	10/03/2021 17:21	Python File	16 Ko

Cet environnement de programmation peut être utilisé comme interpréteur du programme.

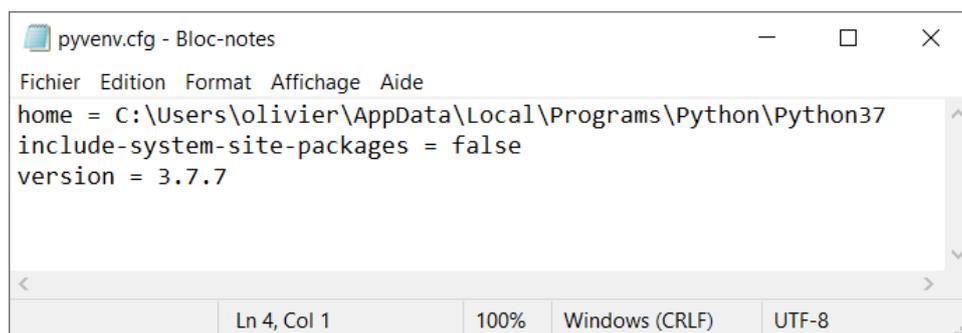
Dans ce cas, même si l'installation d'une distribution **Python** (3.7 au minimum) est indispensable, les bibliothèques dont **ArdPyLog** dépendent n'auront pas à être ajoutées à la distribution originale installée.

L'utilisation de l'environnement de programmation virtuel pour le fonctionnement d'**ArdPyLog** se configure par l'intermédiaire d'un environnement de développement Python (IDE), par exemple, **PyCharm**.

Ainsi dans **PyCharm**, après avoir ouvert le dossier d'**ARDUINO LAB**, il suffit d'indiquer dans les réglages que l'environnement virtuel est l'interpréteur du projet :



Il faudra cependant au préalable modifier le fichier **"pyvenv.cfg"** situé dans le dossier **"ArduinoLab/venv/"** pour indiquer le chemin d'installation de la distribution Python :



2.2 Utilisation de la version portable d'ArdPylog

L'utilisation de la version portable ne nécessite aucune installation. En effet, l'archive zip **"ArdPyLog_vp.zip"** contient l'exécutable et tous les fichiers indispensables à son fonctionnement.

Il suffit de décompresser l'archive zip sur le support de votre choix (disque dur interne ou externe, clé usb...).

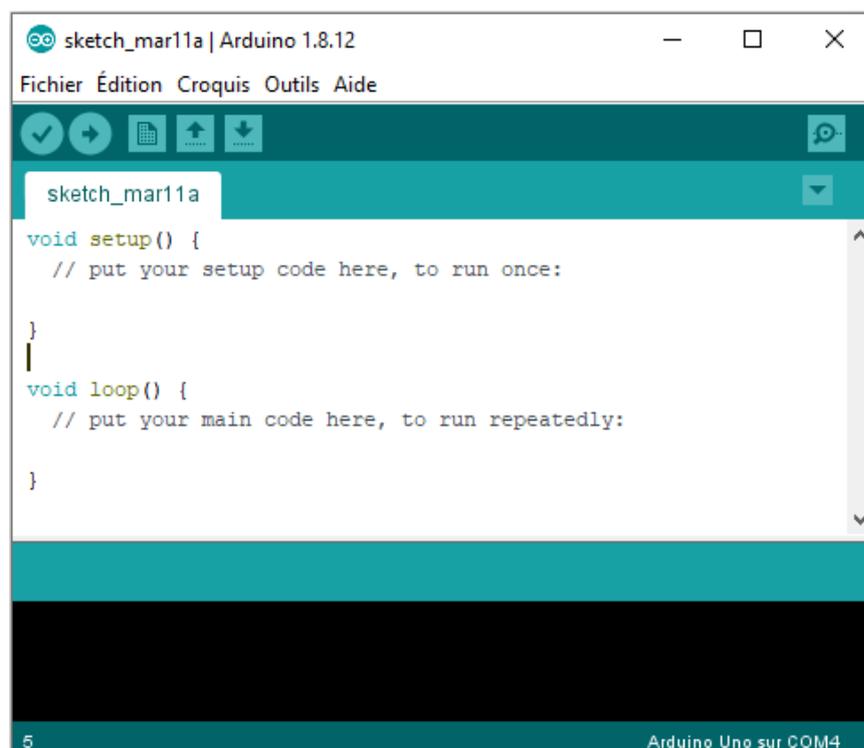
ArdPyLog est alors démarré à l'aide du fichier "**ArdPyLog.exe**" situé dans le dossier principal décompressé.

Nom	Modifié le	Type	Taille
lib	11/03/2021 09:48	Dossier de fichiers	
Media	11/03/2021 09:48	Dossier de fichiers	
Sketchs	11/03/2021 09:48	Dossier de fichiers	
ArdPyLog.exe	09/03/2021 13:04	Application	17 Ko
ArdPyLog.sq3	23/11/2020 11:19	Fichier SQ3	20 Ko
python3.dll	10/03/2020 09:42	Extension de l'application	58 Ko
python37.dll	10/03/2020 09:41	Extension de l'application	3 664 Ko
resolconfig	10/03/2021 17:22	Fichier	1 Ko
vcruntime140.dll	10/03/2020 08:39	Extension de l'application	88 Ko

Remarque : Pré-requis au fonctionnement d'ArdPyLog

Pour utiliser toutes les fonctionnalités d'**ArdPyLog**, notamment le téléversement des codes écrits en langage "Arduino", il est nécessaire d'installer, sur l'ordinateur utilisé pour contrôler l'Arduino, le logiciel "**IDE ARDUINO**" qui est disponible à l'adresse suivante :

<https://www.arduino.cc/en/Main/Software>

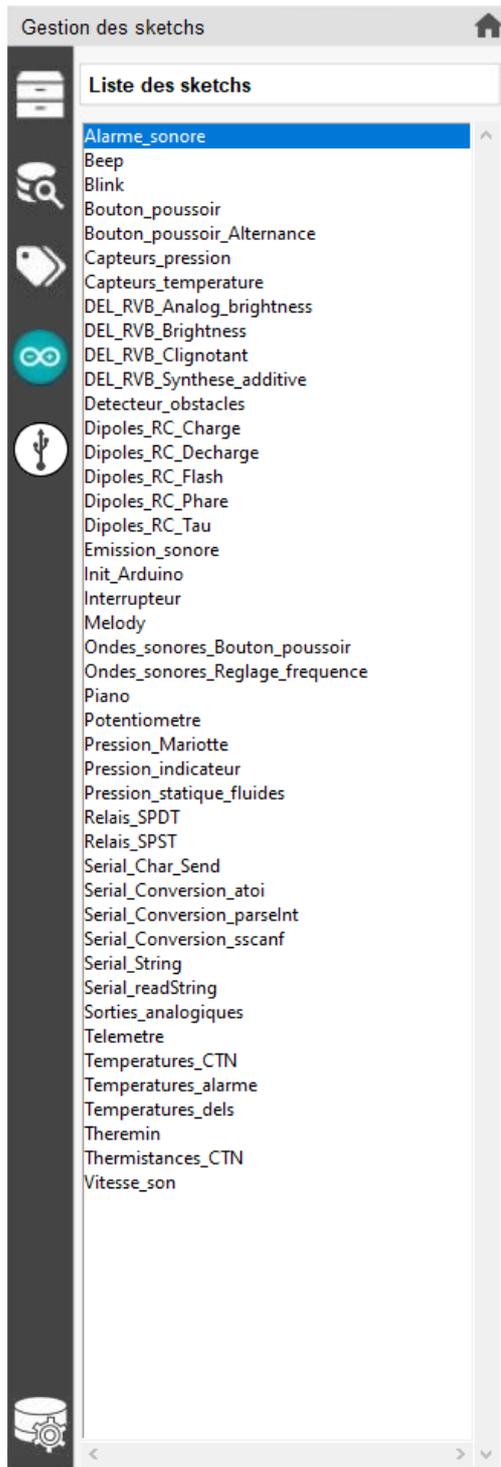


3. L'interface graphique - Les menus

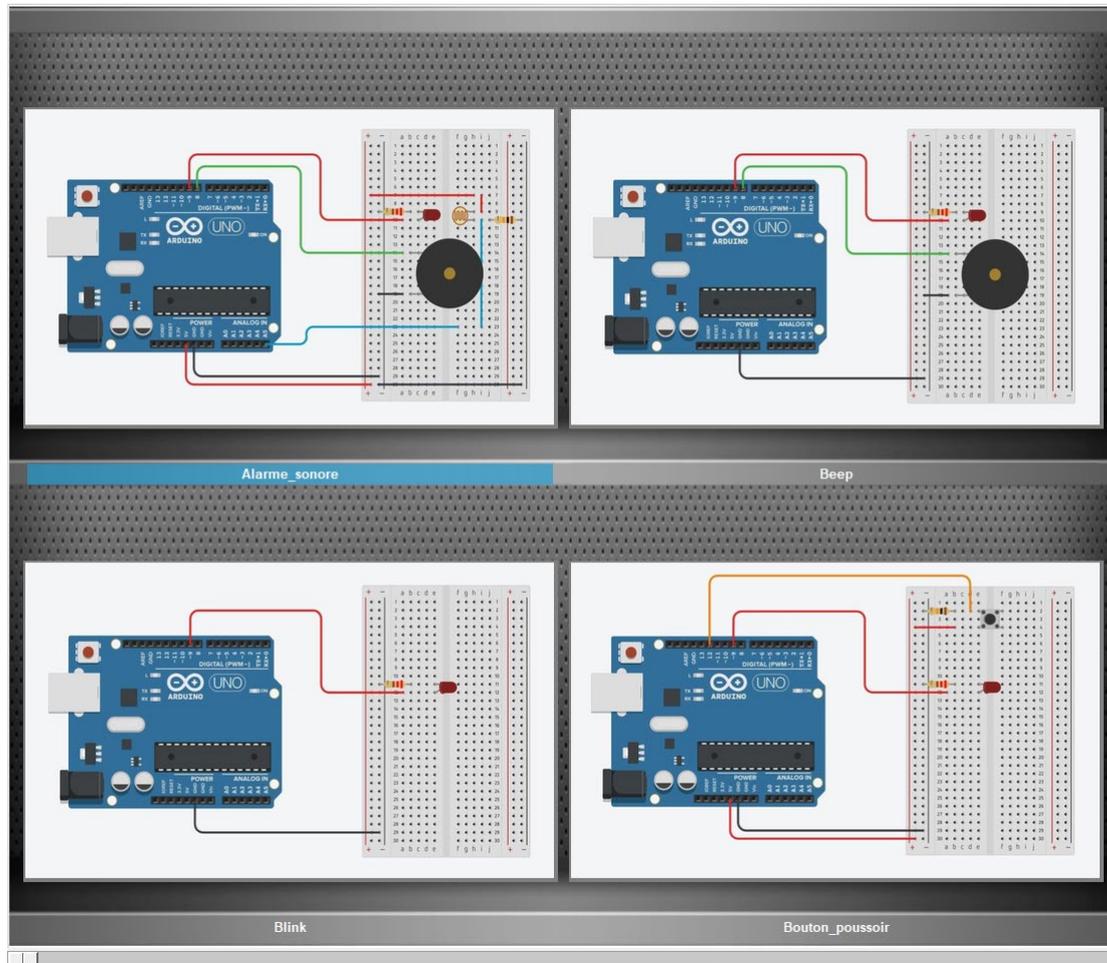
3.1 La fenêtre principale

La fenêtre principale d'**ArdPyLog** est composée de 3 panneaux :

- Le panneau de la gestion des sketchs avec la liste des codes enregistrés dans la base de données :



- Le panneau des circuits électroniques associés au sketches de la base de données :



- le panneau des détails du sketch (circuit, descriptif, code et documentation) :

Détails du sketch

Alarme sonore

Circuit Descriptif Code Documentation

Détails du sketch

Alarme sonore

Liste des composants:

- 1 DEL rouge
- 1 résistance de 220 Ω (résistance de protection de la DEL)
- 1 photorésistance
- 1 résistance de 10 k Ω (résistance de la photorésistance)
- 1 buzzer (ou piezo)
- 1 plaque d'essai
- Fils de connexion

Objectifs:

Dans cette activité, le programme "beep" est utilisé comme alarme de détection de passage.

Pour cela, on ajoute au montage du sketch "beep", une photorésistance qui sera éclairée par la DEL rouge. La sortie de la photorésistance est connectée à l'entrée analogique A5 de l'Arduino.

La valeur de la broche A5 est alors proportionnelle à l'intensité lumineuse reçue par la photorésistance. En présence d'un obstacle entre la DEL et la photorésistance, la tension mesurée au niveau de la broche A5 diminue et quand celle-ci est inférieure à un seuil (la sensibilité du capteur définie initialement), l'alarme sonore est déclenchée.

Circuit Descriptif Code Documentation

3.2 La barre de menu principale



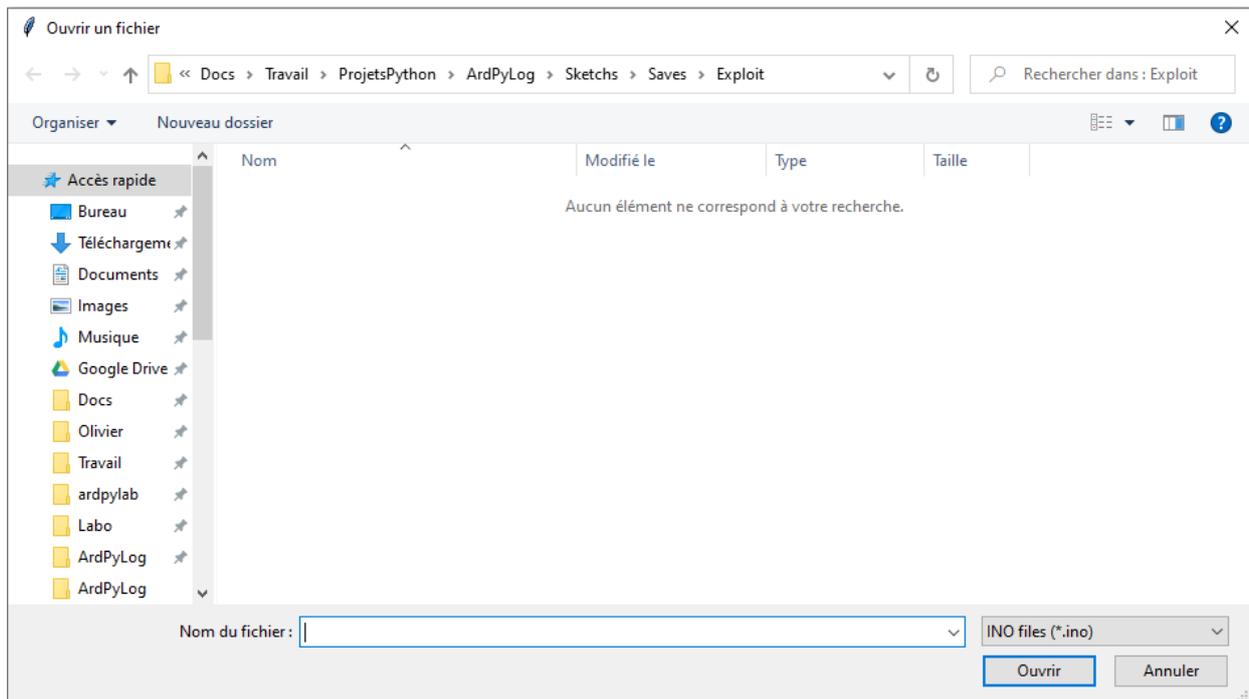
La barre de menu principale permet d'ajouter, supprimer ou modifier des sketches. Elle permet également d'accéder au mode "Tableur" ou au mode "Acquisition".

3.2.1 Ajouter un sketch

Pour ajouter un sketch à la base de données d'ArdPylog, il faut cliquer sur :

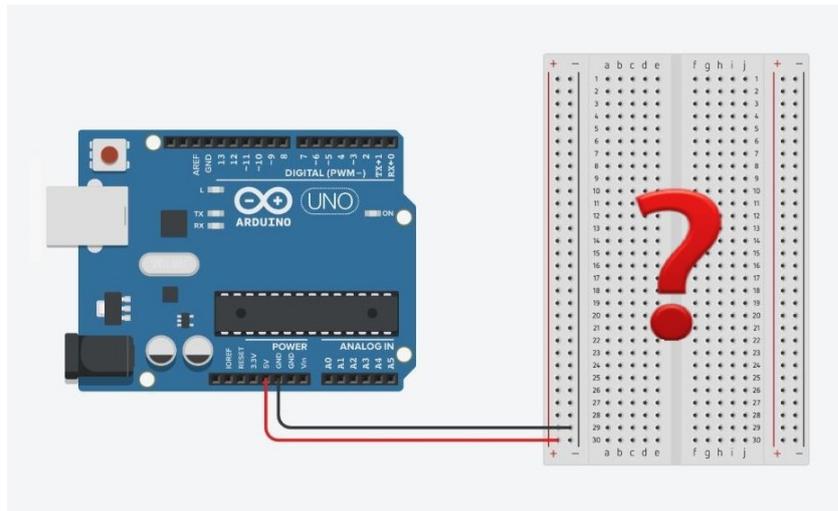


Il suffit alors d'indiquer le chemin du code (fichier avec une extension **.ino**) à ajouter à la base de données :



Le fichier est ensuite copié dans le dossier « Sketchs/Codes » du dossier principal d'ArdPylog.

Dans le panneau des circuits électroniques, le sketch ajouté est associé dans un premier temps à une image par défaut :



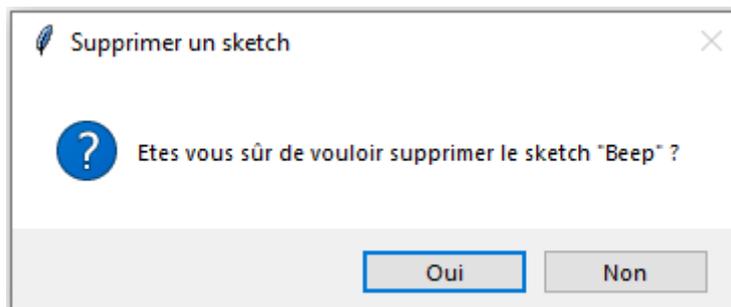
qu'il sera possible de modifier par la suite (voir : **3.2.3 Modifier les données du sketch** – page 10).

3.2.2 Supprimer un sketch

Pour supprimer un sketch de la base de données d'ArdPylog, il faut cliquer sur :



Une boîte de dialogue demande la confirmation de la suppression :



En cliquant sur « **oui** », le sketch est définitivement supprimé de la base de données d'ArdPylog.

Pour supprimer un sketch, il faut qu'il soit au préalable sélectionné en cliquant sur son image dans le panneau des circuits électroniques ou en cliquant sur son nom dans le panneau de la gestion des sketches.

3.2.3 Modifier les données d'un sketch

Dans ArdPylog, les données du sketch sont :

- le circuit électronique associé,
- la liste des composants,
- Le descriptif du sketch,
- un document pdf.

Les données du sketch sont définies ou modifiées en cliquant sur :



La fenêtre de modification des données s'ouvre alors :

Circuit:

Titre:
Beep

Liste des composants:

- 1 DEL rouge
- 1 résistance de 220 Ω (résistance de protection de la DEL)
- 1 buzzer
- 1 plaque d'essai
- Fils de connexion

Objectifs:

Dans cette activité, le programme des activités d'apprentissage qui permet de faire clignoter une DEL est modifié pour commander la production d'un signal sonore ("un beep"), émis par un buzzer ou un petit haut-parleur, synchrone avec le clignotement de la diode.

Code:

```
// Déclaration des constantes et variables
const int PinLED = 9;
const int PinTone = 8;
const int FreqTone = 440;
const int TimeSleep1 = 500;
const int TimeSleep2 = 500;

// Initialisation des entrées et sorties
void setup()
{
  pinMode(PinLED, OUTPUT);
}

// Fonction principale en boucle
void loop()
{
```

Documentation:

Beep
(Faire clignoter une DEL et produire un "beep" synchrone)

Liste des composants :

- 1 DEL rouge
- 1 résistance de 220 Ω (résistance de protection de la DEL)
- 1 buzzer
- 1 plaque d'essai
- Fils de connexion

- Objectif

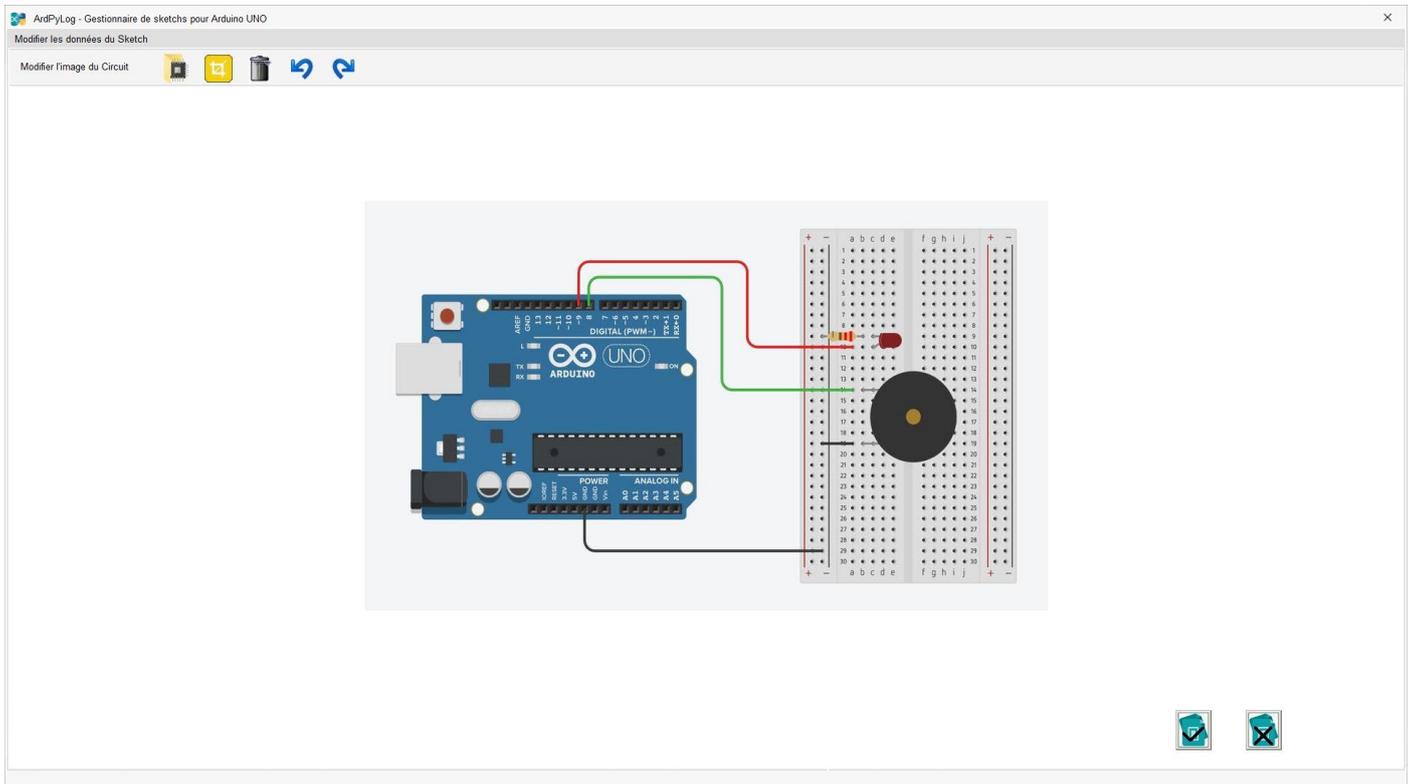
Dans cette activité, le programme des activités d'apprentissage qui permet de faire clignoter une DEL est modifié pour commander la production d'un signal sonore ("un beep"), émis par un buzzer ou un petit haut-parleur, synchrone avec le clignotement de la diode.

Dans cette fenêtre, il est possible de modifier le titre, de définir ou de modifier la catégorie du sketch ainsi que l'image du circuit électronique associé et les autres données.

3.2.3.1 modification de l'image du circuit

Pour définir ou modifier l'image du circuit électronique associé au sketch sélectionné, il faut effectuer un double clic sur l'image.

Une nouvelle fenêtre s'ouvre :



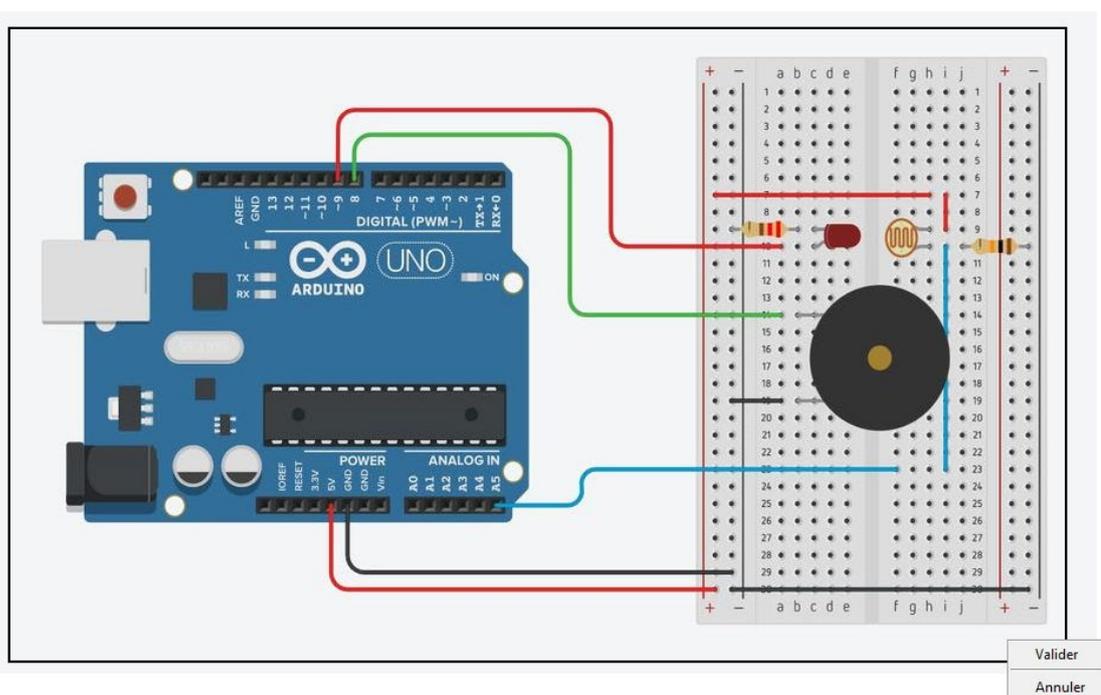
Actions des boutons de la barre de menu :



- Ouvrir et charger une image au format jpg,



- Recadrer l'image chargée par déplacement de la souris sur l'image en maintenant le bouton gauche de la souris appuyé et en validant à l'aide du bouton droit,





- Effacer l'image affichée,



- Annuler la dernière modification,



- Rétablir la dernière annulation.

Les modifications de l'image du circuit électronique associé au sketch sélectionné sont validées ou annulées à l'aide des boutons suivants :



3.2.3.2 modification du descriptif et/ou du code

Les zones de texte de la fenêtre de modification des données permettent de définir ou de modifier la liste des composants du circuit électronique associé au sketch, l'objectif et le code du sketch.

Il est également possible d'ouvrir des fichiers « .txt » contenant le texte à charger dans les zones de texte en cliquant sur :



Ou d'effacer le contenu des zones de texte en cliquant sur :



3.2.3.3 Définition ou modification du document PDF

La définition ou la modification du document PDF associé au sketch sélectionné est réalisée à l'aide du bouton :

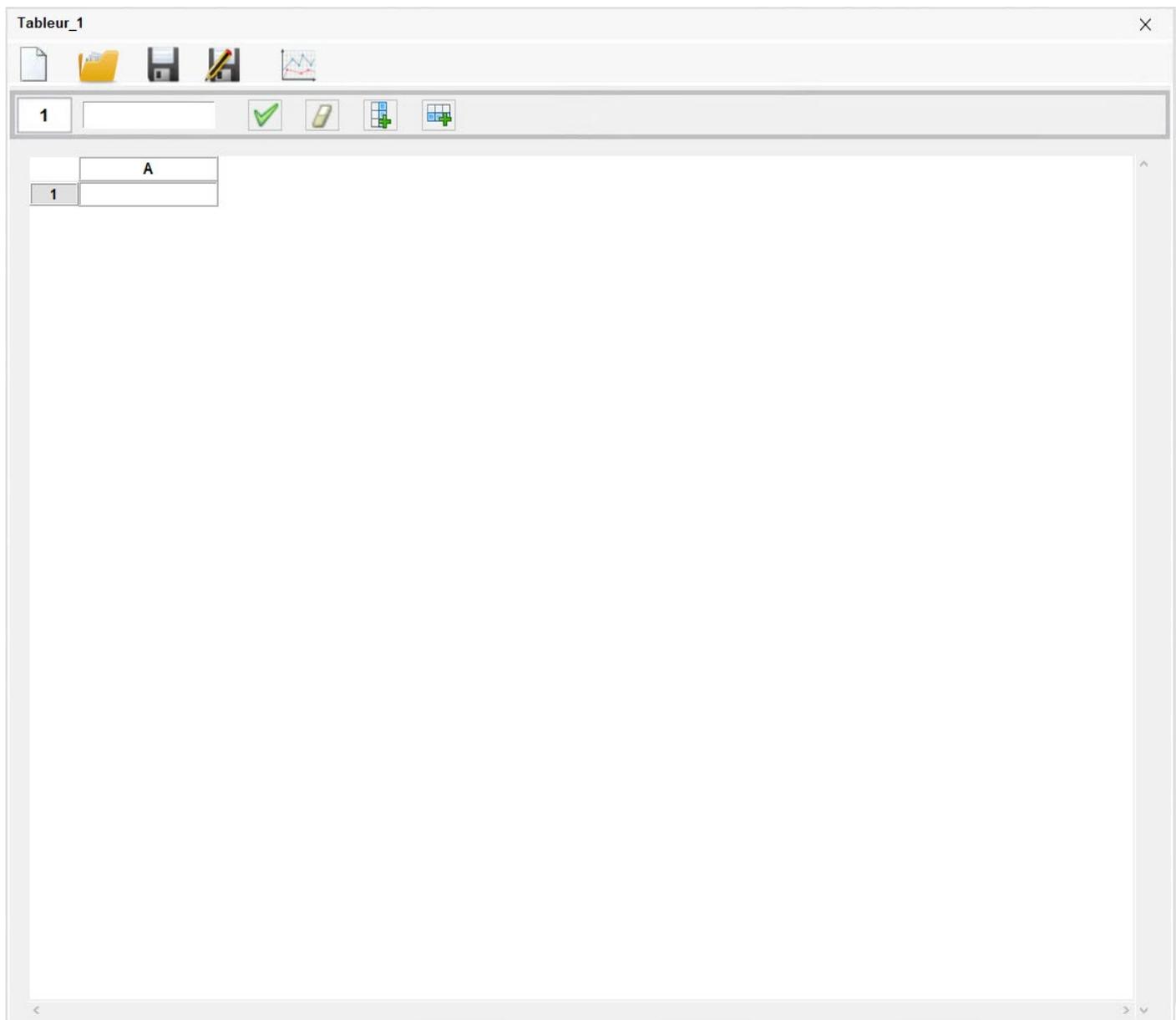


La suppression d'un document PDF associé au sketch sélectionné est effectuée en cliquant sur :



3.2.4 Mode « Tableur »

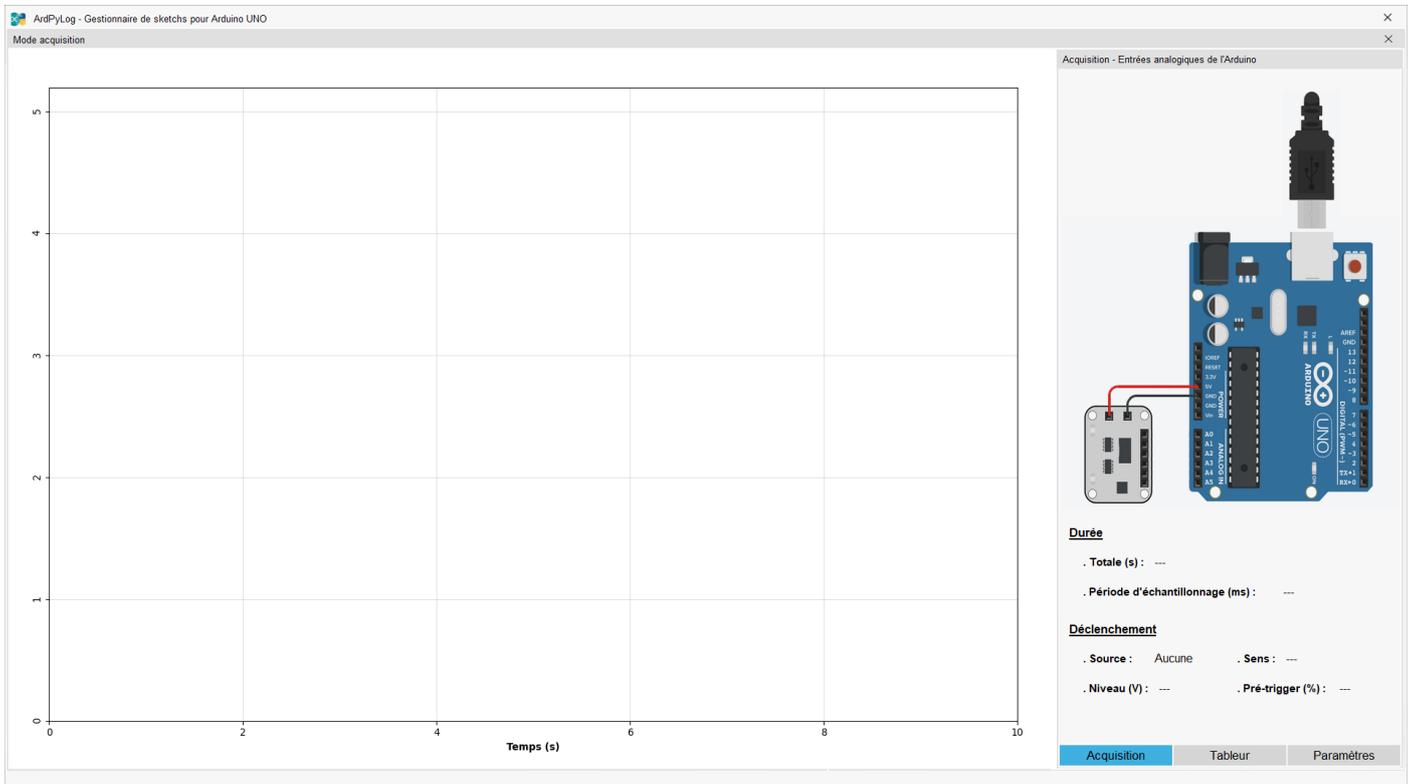
Le tableur permettant l'exploitation des données envoyées par l'Arduino est accessible en cliquant sur :



La description du mode « Tableur » fait l'objet d'une section particulière à la page 30.

3.2.5 Mode « Acquisition »

Le mode « Acquisition » d'ArdPylog transforme l'Arduino Uno en centrale d'acquisition grâce à l'utilisation de ses entrées analogiques. Les tensions mesurées sur les entrées analogiques sélectionnées sont acquises pendant la durée et le taux d'échantillonnage définis par l'utilisateur.



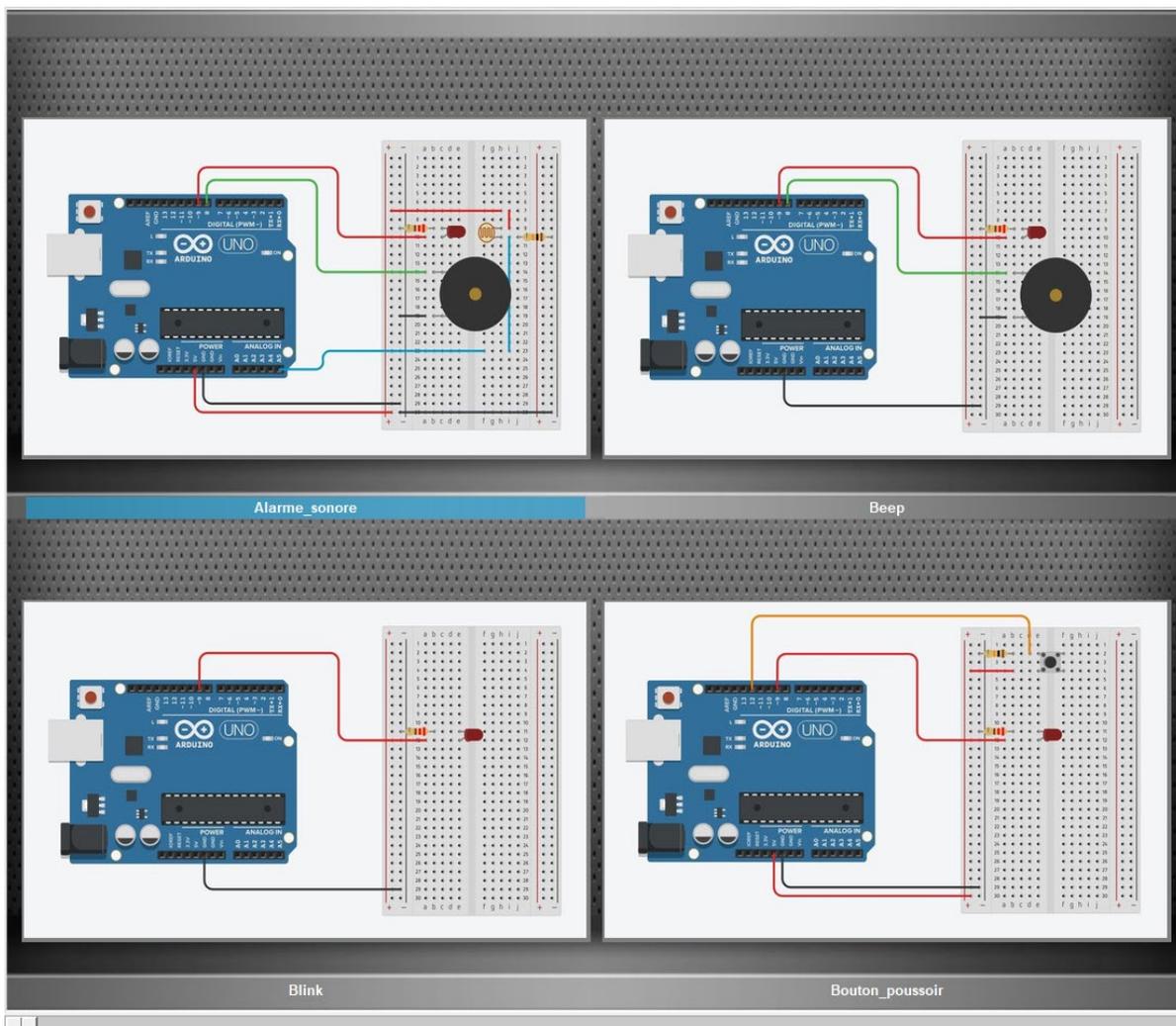
La description du mode « Acquisition » fait également l'objet d'une section particulière à la page **48**.

3.3 La barre de gestion des sketches

Les boutons de la barre de gestion des sketches permettent d'effectuer une recherche de sketches par nom ou par catégorie, de visualiser soit le code en langage Arduino d'un sketch sélectionné, soit le circuit électronique associé, d'accéder au moniteur série pour enregistrer les données envoyées par l'Arduino et enfin de paramétrer le port série utilisé par l'Arduino Uno.

3.3.1 Affichage des circuits électroniques ou du code

Par défaut, au démarrage d'ArdPyLog, le panneau central montre les circuits électroniques associés au sketches de la base de données. En cliquant sur un des circuits du panneau central ou sur un des éléments de la liste des sketches référencés, on sélectionne un sketch.



Pour visualiser le code du sketch sélectionné, il faut cliquer sur :



Le panneau central affiche alors la fenêtre de gestion des codes en langage Arduino dont le fonctionnement est similaire à celui du programme **Arduino IDE** :

```
Alarme_sonore

// Déclaration des constantes et variables

const int PinLED = 9;
const int PinTone = 8;
const int PinPhotoR = 5;
const int CapteurSensib = 500;
const int TimeSleep = 200;

int ValCapteur = 0;

// Initialisation des entrées et sorties

void setup() {
  pinMode (PinLED, OUTPUT);
  digitalWrite(PinLED, HIGH);
}

// Fonction principale en boucle

void loop() {
  ValCapteur = analogRead(PinPhotoR);
  if (ValCapteur < CapteurSensib) {
    tone(PinTone, 440);
    delay(TimeSleep);
    noTone(PinTone);
    delay(TimeSleep);
  }
  else {
    noTone(PinTone);
  }
}
```

La description du fonctionnement la fenêtre de gestion des codes en langage Arduino fait également l'objet d'une section particulière à la page 26.

Le retour à la fenêtre des étagères de circuits électroniques se fait en cliquant sur :



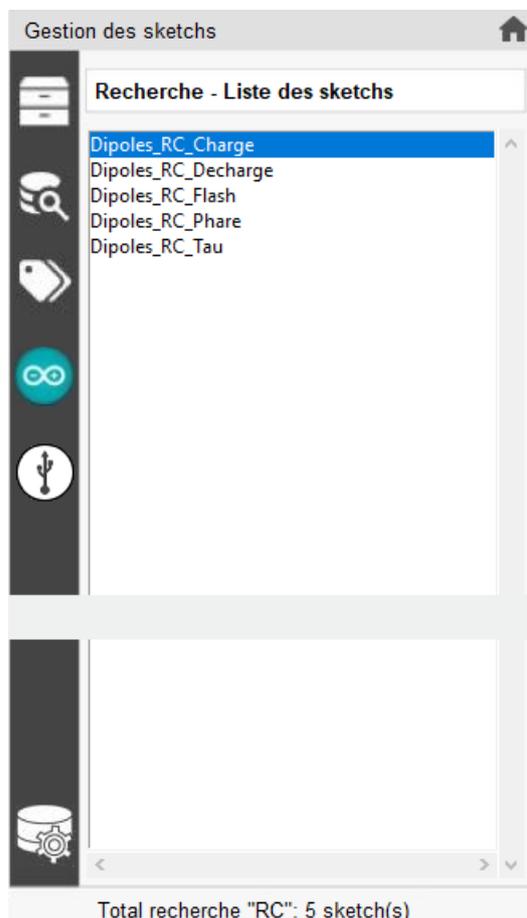
3.3.2 Recherche de sketches par nom ou par catégorie

Dans ArdPyLog, les sketches peuvent être classés par catégorie. Il est alors possible d'effectuer une recherche d'un sketch par son nom ou par sa catégorie en cliquant sur :



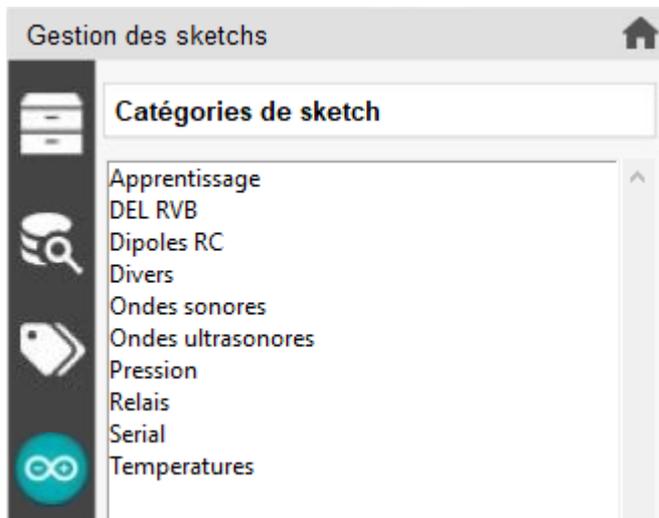
Une zone de texte dans laquelle il faut saisir le texte à rechercher dans les noms ou les catégories de sketches référencés (la recherche étant validée en cliquant sur la loupe ou en appuyant sur la touche « Entrée ») est alors affichée :

Le résultat de la recherche apparaît ensuite dans le panneau de gestion des sketches :

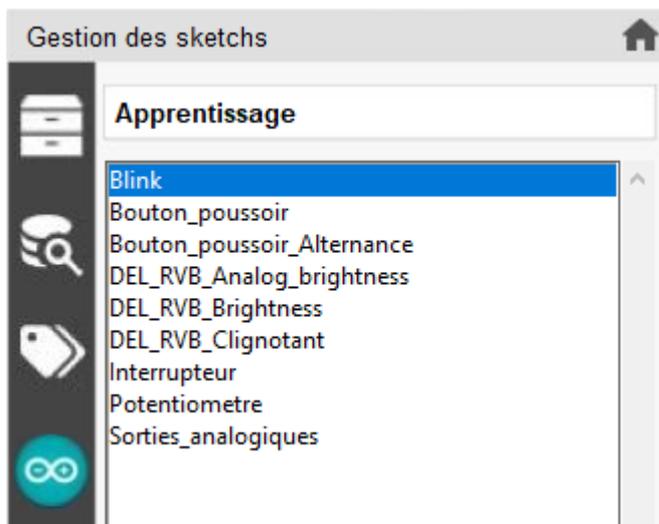


La liste des catégories de sketches référencés est consultable en cliquant sur :





Un clic sur le nom d'une catégorie affiche la liste des sketches référencés dans la catégorie sélectionnée :



3.3.3 Affichage du moniteur série

Dans le cas où un code a déjà été téléversé dans la mémoire de l'Arduino, il est possible d'accéder directement au moniteur série afin de visualiser les données envoyées par le microcontrôleur ou à l'inverse lui envoyer des données, à l'aide du bouton :



Le moniteur série est affiché au niveau du panneau central :



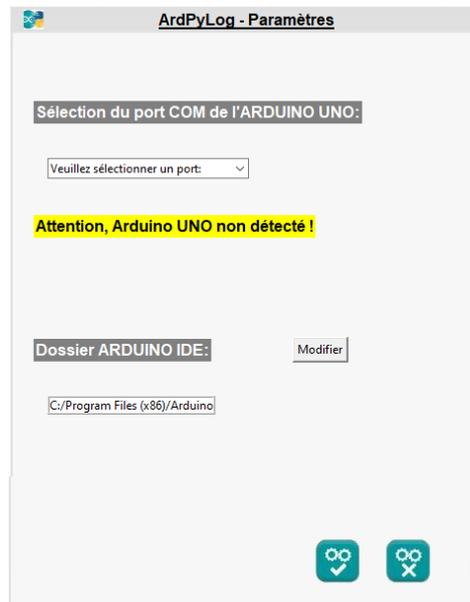
La description du fonctionnement du moniteur série fait également l'objet d'une section particulière à la page 28.

3.3.4 Modifications des paramètres d'ArdPyLog

Pour modifier les paramètres, il faut cliquer sur :



La fenêtre permettant de modifier les paramètres est affichée dans le panneau de droite :



Les paramètres d'ArdPylog sont :

- le port COM sur lequel l'Arduino Uno est connecté,
- le chemin du dossier d'installation du programme Arduino IDE.

Remarques :

. Si un Arduino Uno est déjà connecté au lancement d'ArdPylog, la sélection du port COM est automatique. La sélection du port COM n'est nécessaire que si l'Arduino Uno est connecté après le lancement d'ArdPylog ou si plusieurs microcontrôleurs sont connectés.

. Par défaut, le chemin du dossier du programme Arduino IDE est celui que le logiciel propose lors de son installation.

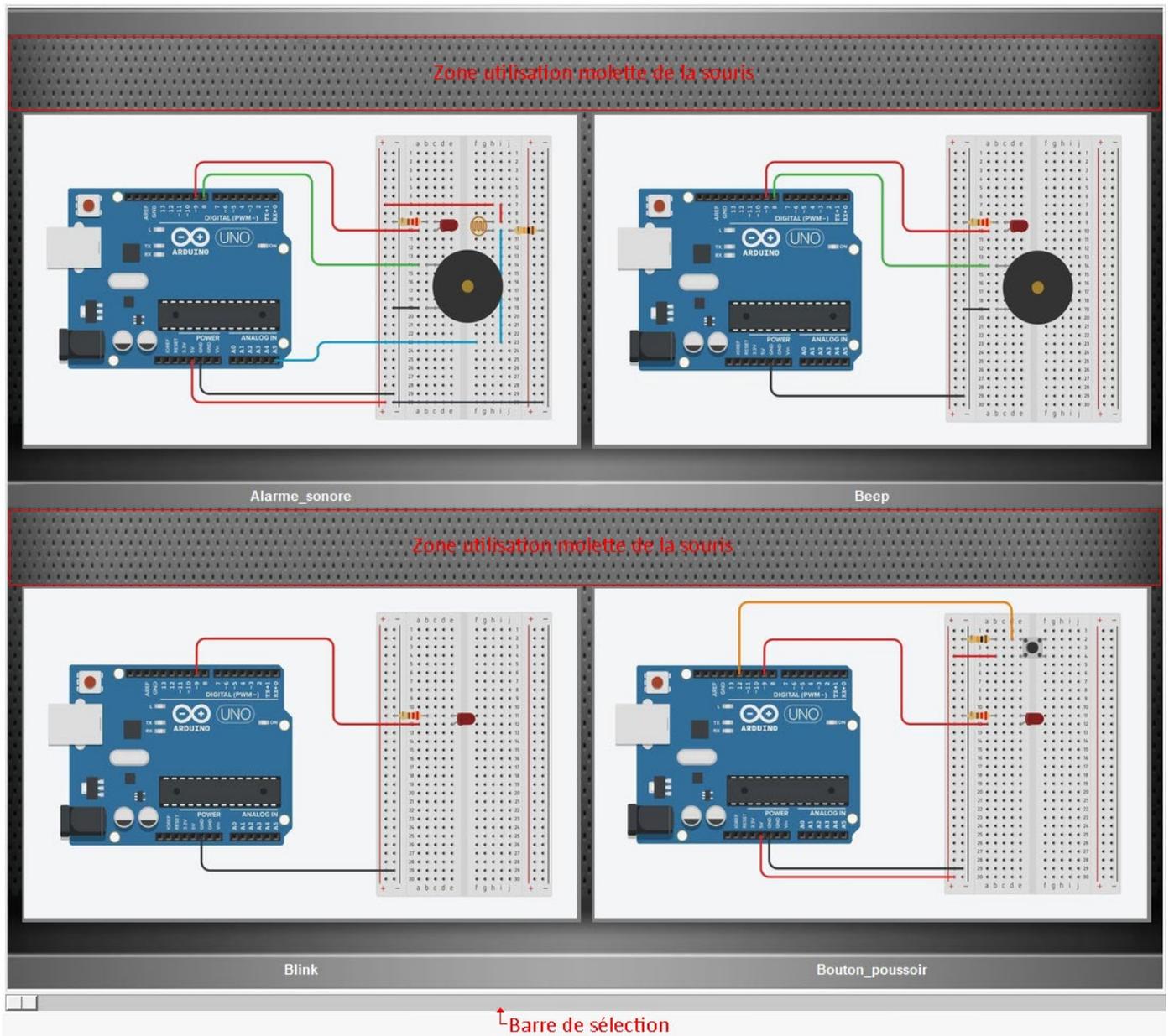
3.3.5 Retour à la fenêtre d'accueil d'ArdPyLog

Le bouton  permet à tout moment de revenir à la page d'accueil d'ArdPyLog et donc d'annuler toutes les actions en cours.

3.4 Le panneau des circuits électroniques

Quand le panneau central montre les circuits électroniques associés au sketchs de la base de données, en cliquant sur un de ces circuits, le sketch correspondant est sélectionné.

La navigation dans les étagères de circuits se fait avec la barre de sélection sous le panneau ou à l'aide de la molette de la souris quand celle-ci le survole)

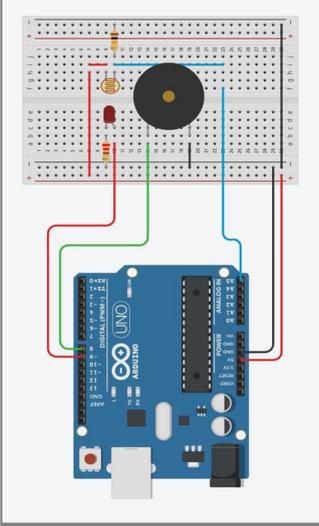


Un double clic sur les circuits affiche une nouvelle fenêtre permettant de visualiser tous les détails du sketch sélectionné :

ArdPyLog - Gestionnaire de sketches pour Arduino UNO

Données du Sketch

Circuit:



Titre:
Alarme_sonore

Liste des composants:

- 1 DEL rouge
- 1 résistance de 220 Ω (résistance de protection de la DEL)
- 1 photorésistance
- 1 résistance de 10 kΩ (résistance de la photorésistance)
- 1 buzzer (ou piezo)
- 1 plaque d'essai
- Fils de connexion

Objectifs:

Dans cette activité, le programme "beep" est utilisé comme alarme de détection de passage.

Pour cela, on ajoute au montage du sketch "beep", une photorésistance qui sera éclairée par la DEL rouge. La sortie de la photorésistance est connectée à l'entrée analogique A5 de l'Arduino.

La valeur de la broche A5 est alors proportionnelle à l'intensité lumineuse reçue par la photorésistance. En présence d'un obstacle entre la DEL et la photorésistance, la tension mesurée au niveau de la broche A5 diminue et quand celle-ci est inférieure à un seuil (la sensibilité du capteur définie initialement), l'alarme sonore est déclenchée.

Code:

```
// Déclaration des constantes et variables
const int PinLED = 9;
const int PinTone = 8;
const int PinPhotoR = 5;
const int CapteurSensibil = 500;
const int TimeSleep = 200;

int ValCapteur = 0;

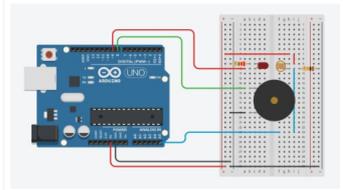
// Initialisation des entrées et sorties
void setup() {
  pinMode(PinLED, OUTPUT);
  digitalWrite(PinLED, HIGH);
}

// Fonction principale en boucle
```

Catégorie:
Ondes sonores

Documentation:

Alarme sonore (par détection de passage)



Liste des composants :

- 1 DEL rouge
- 1 résistance de 220 Ω (résistance de protection de la DEL)
- 1 photorésistance
- 1 résistance de 10 kΩ (résistance de la photorésistance)
- 1 buzzer
- 1 plaque d'essai
- Fils de connexion

Objectif

Dans cette activité, le programme "beep" est utilisé comme alarme de détection de passage.

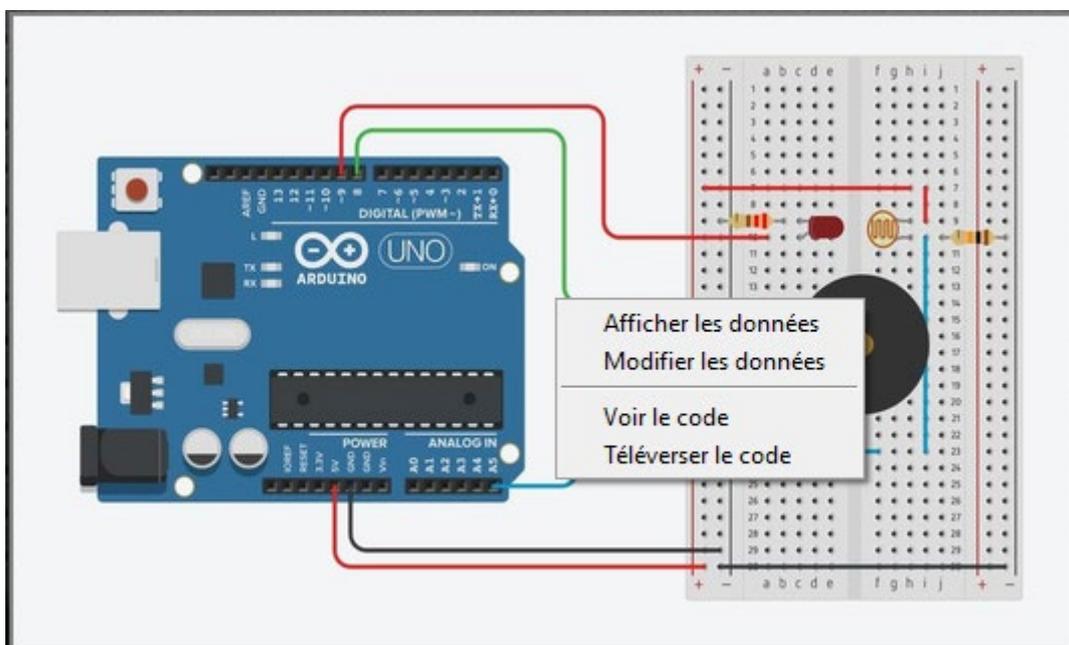
Pour cela, on ajoute au montage du sketch "beep", une photorésistance qui sera éclairée par la DEL rouge. La sortie de la photorésistance est connectée à l'entrée analogique A5 de l'Arduino.

La valeur de la broche A5 est alors proportionnelle à l'intensité lumineuse reçue par la photorésistance. En présence d'un obstacle entre la DEL et la photorésistance, la tension mesurée au niveau de la broche A5 diminue et quand celle-ci est inférieure à un seuil (la sensibilité du capteur définie initialement), l'alarme sonore est déclenchée.

. Le bouton  permet d'afficher la fenêtre de modification des détails du sketch (voir : 3.2.3 Modifier les données d'un sketch - page 10)

. Le bouton  permet de retourner au panneau des circuits électroniques.

Un clic droit sur un circuit affiche un nouveau menu permettant d'accéder rapidement à 4 actions :



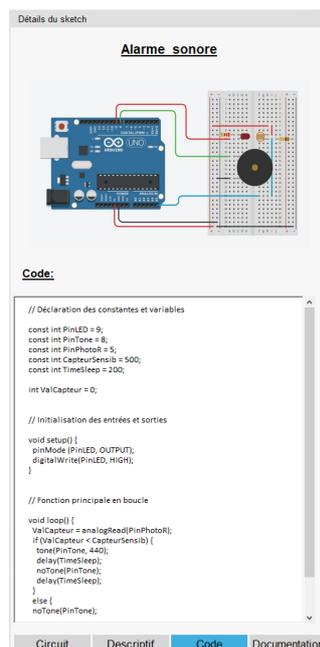
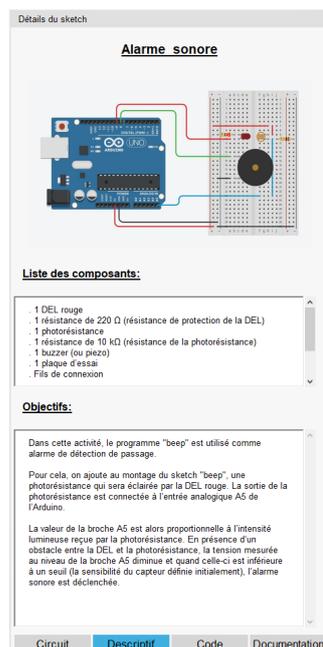
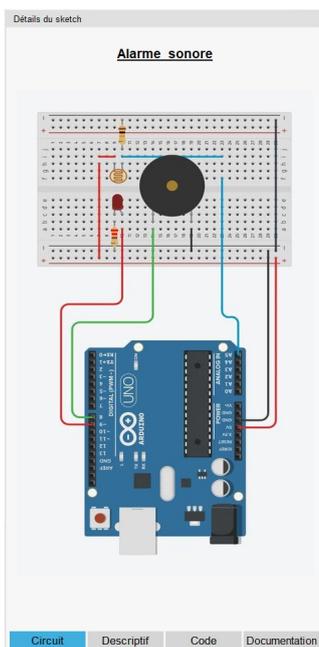
- **Afficher les données** : Affiche une nouvelle fenêtre permettant de visualiser tous les détails du sketch sélectionné,
- **Modifier les données** : Affiche la fenêtre de modification des détails du sketch (voir : *2.2.3 Modifier les données d'un sketch* - page 10),
- **Voir le code** : Affiche la fenêtre de gestion du code en langage Arduino (voir page 16),
- **Téléverser le code** : Téléverse le code du sketch sélectionné dans la mémoire de l'Arduino Uno.

3.5 le panneau des détails du sketch

Quand un sketch est sélectionné, le panneau de droite affiche les détails du sketch sur 4 volets :

- Le circuit électronique associé,
- La liste des composants et l'objectif du sketch,
- Le code en langage Arduino,
- Le document PDF de synthèse.

La navigation entre les différents volets du panneau des détails du sketch se fait avec la barre de menu suivante :



Un double clic sur le circuit, la liste des composants, l'objectif, le code ou le document PDF effectue un zoom sur l'élément sélectionné qui est alors affiché au niveau du panneau central.

ArdPyLog - Gestionnaire de sketches pour Arduino UNO

Gestion des sketches

Liste des sketches

- Alarme_sonore
- Beep
- Blink
- Bouton_poussoir
- Bouton_poussoir_Alternance
- Capteurs_pression
- Capteurs_temperature
- DEL_RVB_Analog_brightness
- DEL_RVB_Brightness
- DEL_RVB_Cignotant
- DEL_RVB_Synthese_additive
- Detecteur_obstacles
- Dipotes_RC_Charge
- Dipotes_RC_Decharge
- Dipotes_RC_Flash
- Dipotes_RC_Piase
- Dipotes_RC_Tau
- Emission_sonore
- Int_Arduino
- Interrupteur
- Melody
- Ondes_sonores_Bouton_poussoir
- Ondes_sonores_Reglage_frequence
- Piano
- Potentiometre
- Pression_Marotte
- Pression_indicateur
- Pression_statique_fluides
- Relais_SPDT
- Relais_SPST
- Serial_Char_Send
- Serial_Conversion_atoi
- Serial_Conversion_parseint
- Serial_Conversion_sscanf
- Serial_String
- Serial_readString
- Sorties_analogiques
- Telemetrie
- Temperatures_CTN
- Temperatures_alarme
- Temperatures_delta
- Thermoin
- Thermistances_CTN
- Vitesse_son

Alarme_sonore - Circuit

Détails du sketch

Alarme sonore

Liste des composants:

- 1 DEL rouge
- 1 résistance de 220 Ω (résistance de protection de la DEL)
- 1 photorésistance
- 1 résistance de 10 k Ω (résistance de la photorésistance)
- 1 buzzer (ou piezo)
- 1 plaque d'essai
- Fils de connexion

Objectifs:

Dans cette activité, le programme "beep" est utilisé comme alarme de détection de passage.

Pour cela, on ajoute au montage du sketch "beep", une photorésistance qui sera éclairée par la DEL rouge. La sortie de la photorésistance est connectée à l'entrée analogique A5 de l'Arduino.

La valeur de la broche A5 est alors proportionnelle à l'intensité lumineuse reçue par la photorésistance. En présence d'un obstacle entre la DEL et la photorésistance, la tension mesurée au niveau de la broche A5 diminue et quand celle-ci est inférieure à un seuil (la sensibilité du capteur définie initialement), l'alarme sonore est déclenchée.

Circuit Descriptif Code Documentation

Total: 44 sketch(s)

Remarque :

Si un lecteur de documents PDF est installé sur l'ordinateur hôte, un double clic avec la roulette de la souris sur le document PDF, affiché dans ArdPyLog, lance le lecteur et l'ouverture de ce document pour une lecture plus facile.

Alarme_sonore - Documentation

Alarme sonore (par détection de passage)

Liste des composants :

- . 1 DEL rouge
- . 1 résistance de 220 Ω (résistance de protection de la DEL)
- . 1 photorésistance
- . 1 résistance de 10 k Ω (résistance de la photorésistance)
- . 1 buzzer
- . 1 plaque d'essai
- . Fils de connexion

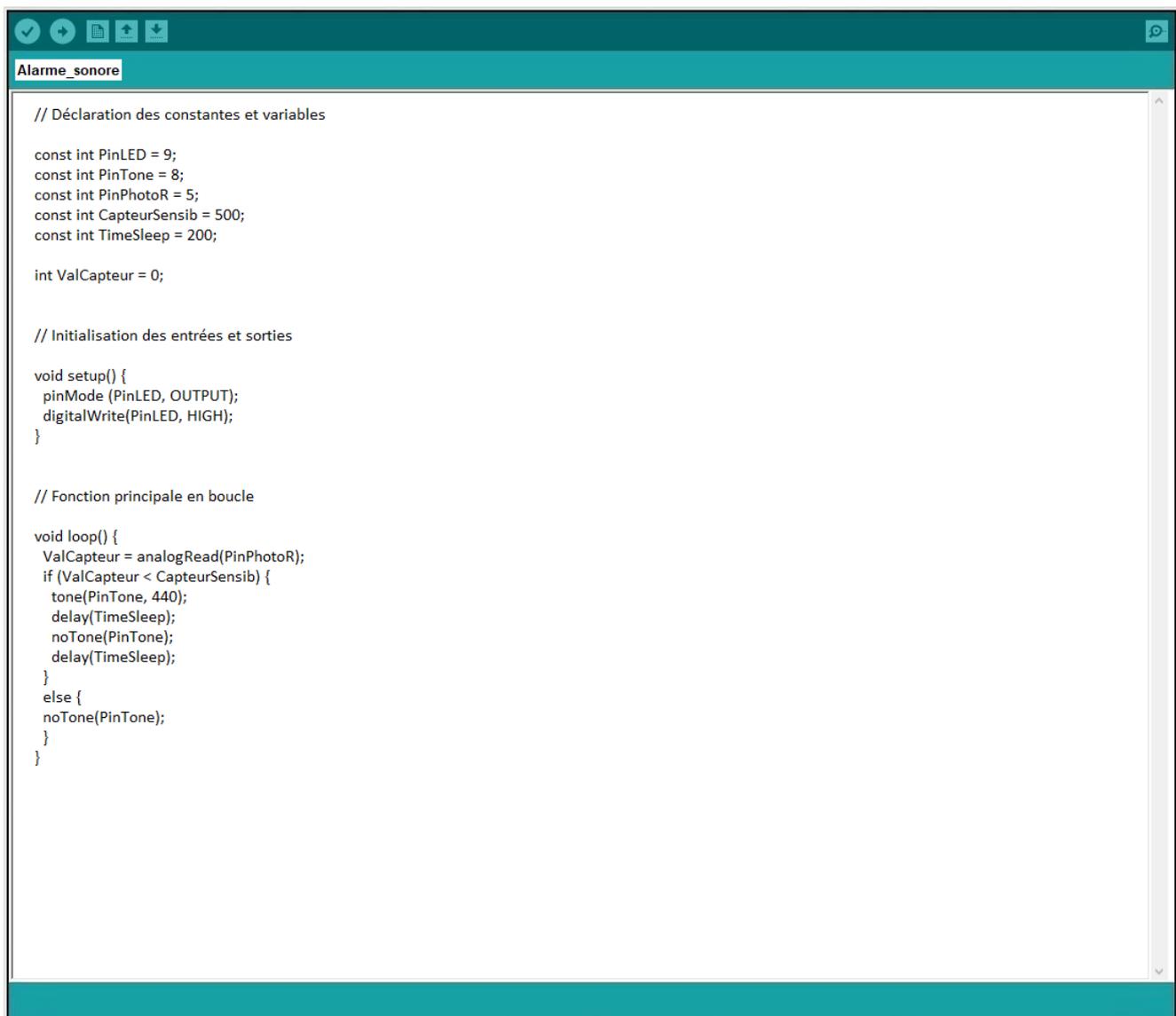
4. Gestion des codes en langage « Arduino »

4.1 La fenêtre principale

Pour visualiser le code d'un sketch sélectionné, il faut cliquer sur :



Le panneau central affiche alors la fenêtre de gestion des codes en langage Arduino dont le fonctionnement est similaire à celui du programme Arduino IDE :

A screenshot of the Arduino IDE code editor window. The window title is "Alarme_sonore". The code is written in C++ and is as follows:

```
// Déclaration des constantes et variables

const int PinLED = 9;
const int PinTone = 8;
const int PinPhotoR = 5;
const int CapteurSensib = 500;
const int TimeSleep = 200;

int ValCapteur = 0;

// Initialisation des entrées et sorties

void setup() {
  pinMode (PinLED, OUTPUT);
  digitalWrite(PinLED, HIGH);
}

// Fonction principale en boucle

void loop() {
  ValCapteur = analogRead(PinPhotoR);
  if (ValCapteur < CapteurSensib) {
    tone(PinTone, 440);
    delay(TimeSleep);
    noTone(PinTone);
    delay(TimeSleep);
  }
  else {
    noTone(PinTone);
  }
}
```

Si aucun sketch n'est sélectionné quand un clic est effectué sur le bouton d'affichage des codes, la fenêtre affiche une page vierge sur laquelle un code pourra être écrit :



Il est possible de sauvegarder le code modifié, d'éditer un nouveau code et d'ouvrir un code préalablement enregistré.

Le code original ou modifié après sauvegarde peut être vérifié et téléversé directement depuis **ArdPyLog** dans la mémoire de l'Arduino Uno.

Les fonctionnalités décrites ci-dessus sont accessibles en cliquant sur les boutons :

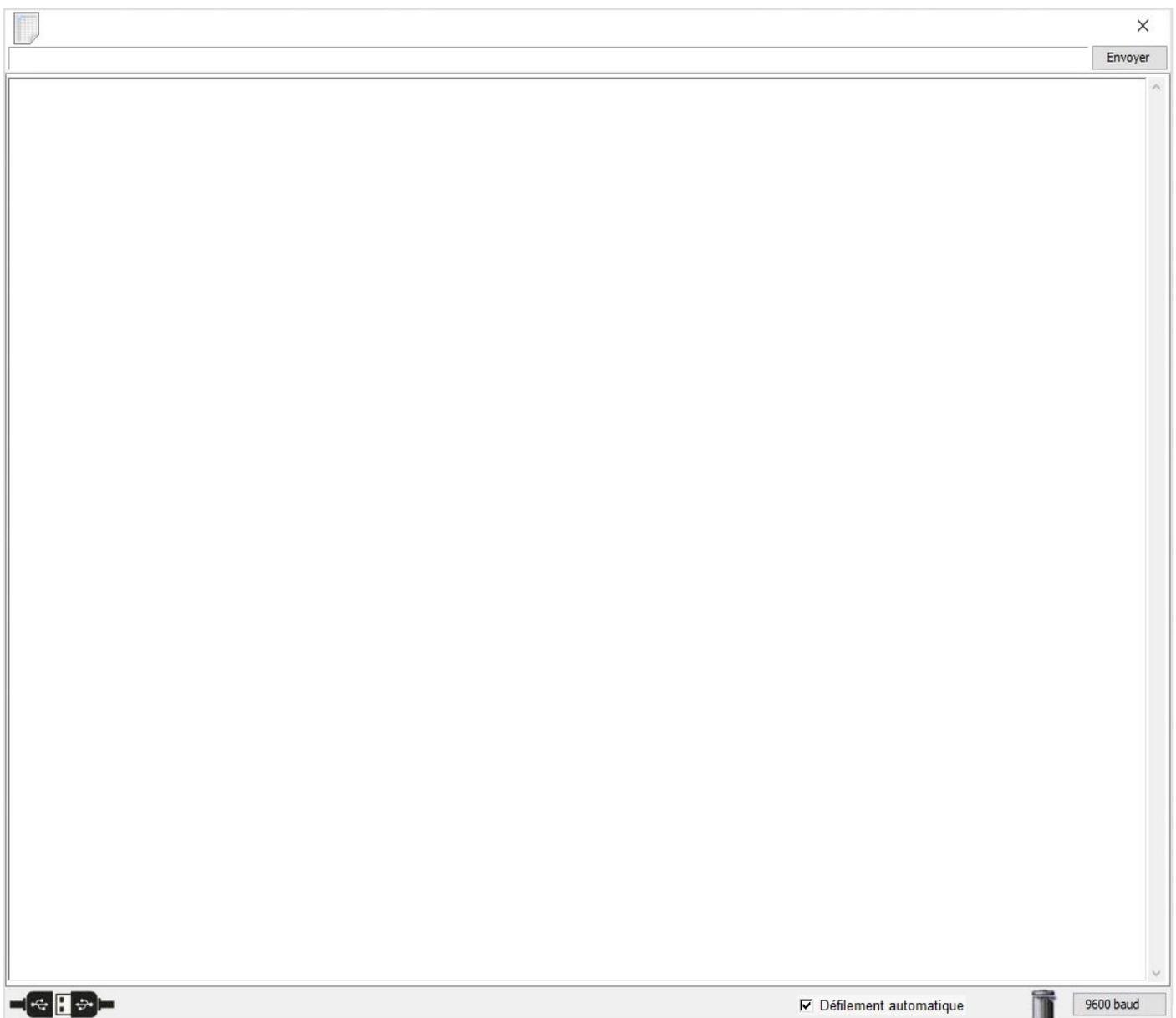
- Vérifier un code : 
- Téléverser un code : 
- Editer un nouveau code : 
- Ouvrir un code : 
- Sauvegarder un code : 

4.2 Le moniteur série

Comme Le logiciel Arduino IDE, **ArdPylog** dispose d'un moniteur série, qui permet de recevoir et d'envoyer des informations via la liaison série. Il est particulièrement intéressant pour l'acquisition des valeurs des capteurs connectés à l'Arduino Uno et permettre leur exploitation.

Le moniteur série est accessible en cliquant sur la loupe en haut à droite de la fenêtre de gestion des codes en langage Arduino: 

Toujours dans le panneau central, une nouvelle fenêtre s'ouvre alors :



La fenêtre est composée de deux zones blanches d'exploitation et de plusieurs commandes :

- la grande zone au centre est la zone d'affichage des données reçues par le moniteur et donc envoyées par l'Arduino,

- la petite zone rectangulaire, en haut est une zone de saisie des données à envoyer à l'Arduino à l'aide du bouton "Envoyer" qui se trouve à sa droite. C'est le programme du microcontrôleur qui génère la demande d'envoi de données par l'utilisateur et qui s'occupe de traiter les informations reçues,

- la case cochée "Défilement automatique" permet d'arrêter le défilement des données retournées par l'Arduino si on la décoche,

- le bouton  efface la zone des données reçues,

- le bouton  établit la connexion avec l'Arduino Uno et initialise le programme stocké dans sa mémoire,

- enfin, le bouton  affiche le tableur pour l'exploitation des données envoyées par l'Arduino Uno (Voir **5. Mode « tableur »** - page 30).

Remarques :

- Le débit de transmission des données par le port série est fixé à 9600 bauds.

- Tant que la connexion avec l'Arduino n'est pas établie par l'intermédiaire du bouton



, le programme stocké dans la mémoire de l'Arduino est en cours de fonctionnement, mais les données envoyées sur le port série ne sont pas lues. Le programme est réinitialisé après un clic sur le bouton et les données transmises par l'Arduino sont alors lues.

5. Le mode « Tableur »

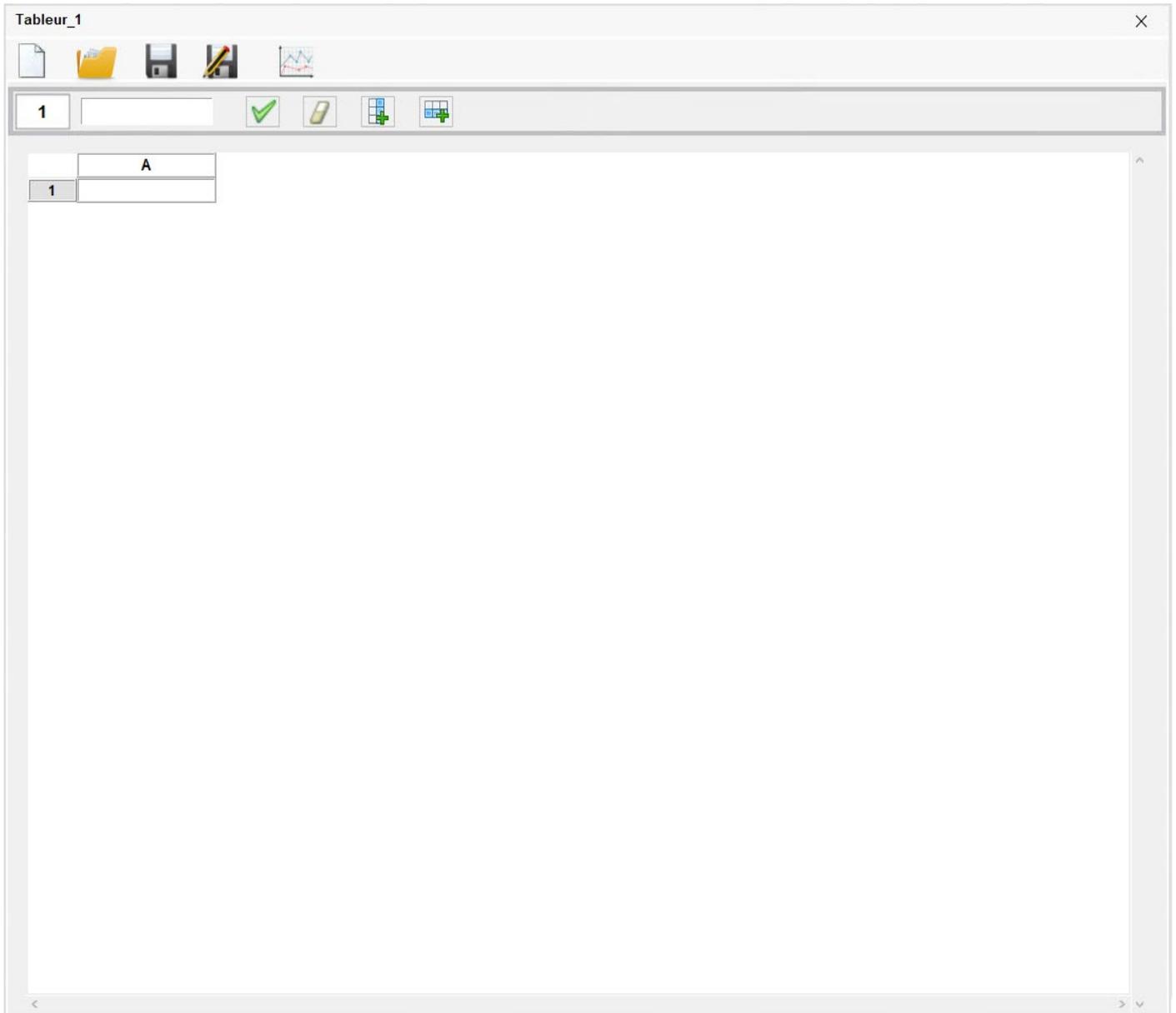
ArdPyLog dispose d'un tableur pour l'exploitation des données envoyées par l'Arduino Uno. Celui-ci est accessible en cliquant sur :



Ce bouton est situé dans la barre de menu principale :



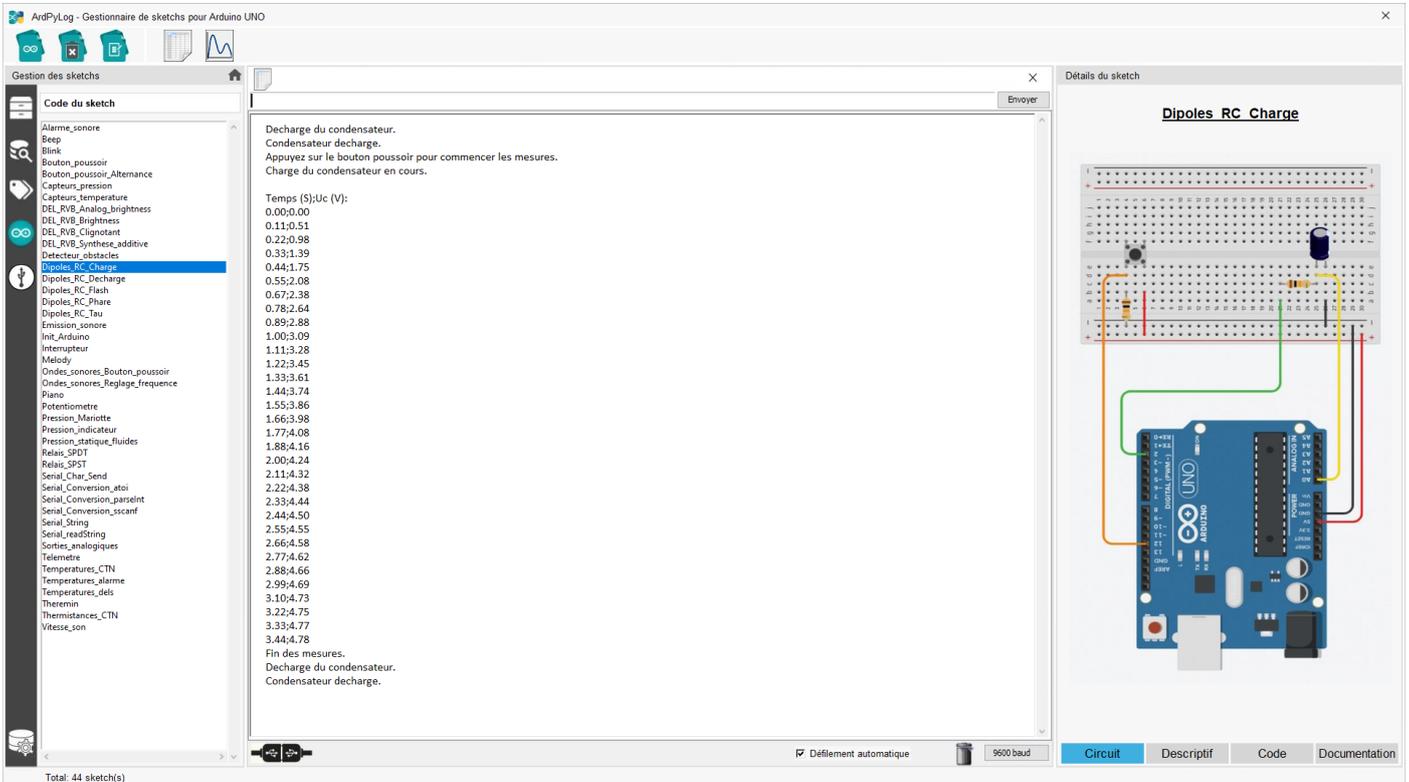
ou dans la fenêtre du moniteur série (voir page 28).



Tableur d'ArdPylog

5.1 Enregistrement des données envoyées par l'Arduino Uno

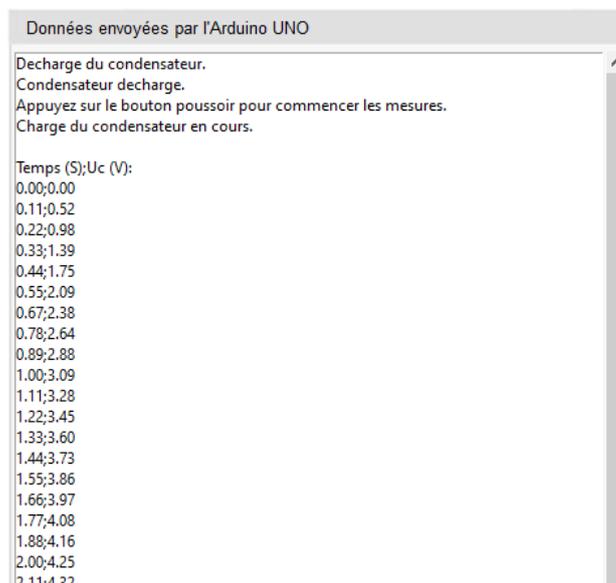
Dans un premier temps pour faire l'acquisition des données envoyées par l'Arduino Uno, on utilise le moniteur série, comme dans l'exemple ci-dessous où la charge du condensateur d'un dipôle RC est étudiée, l'Arduino mesurant la tension du condensateur en fonction du temps :



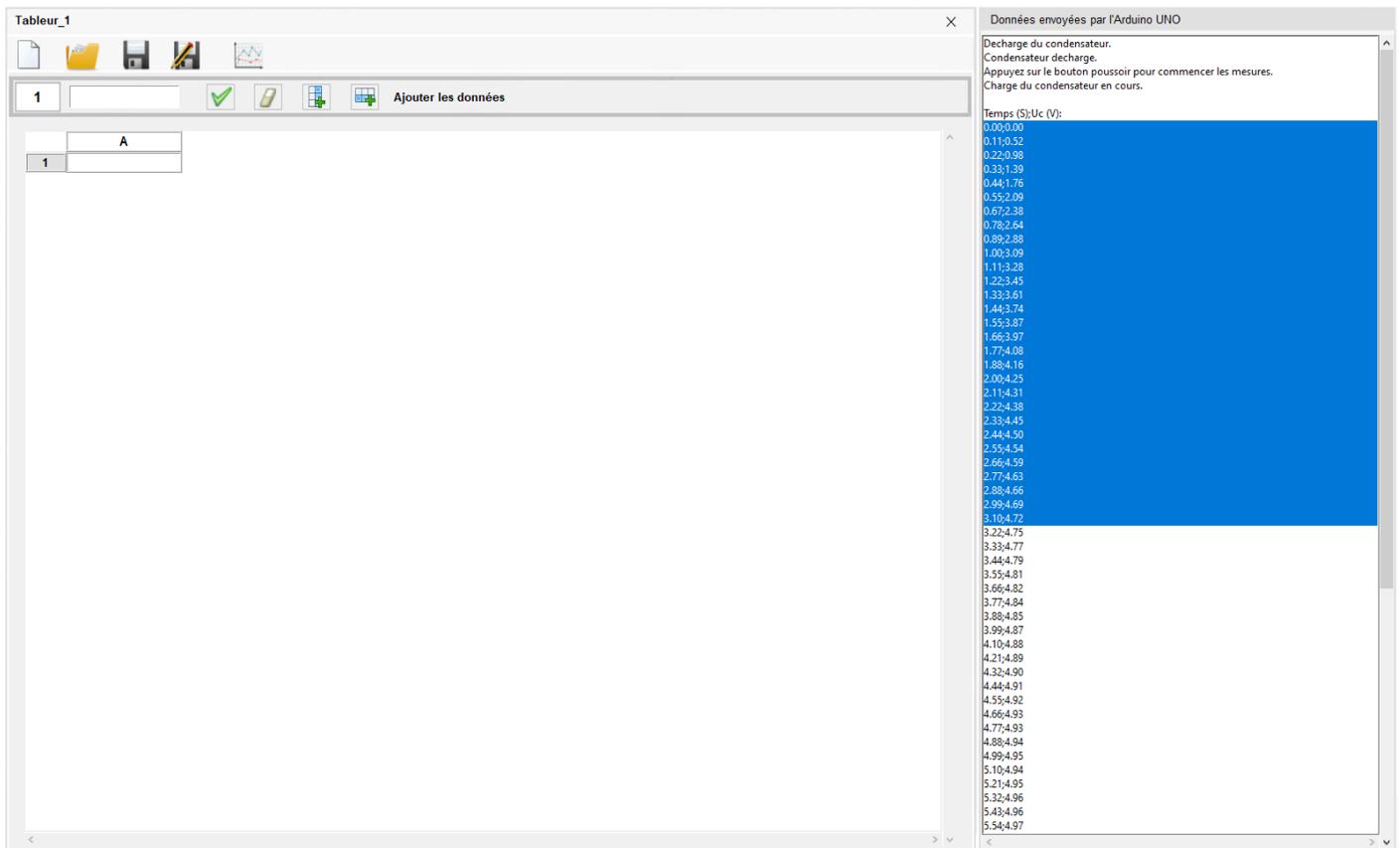
L'enregistrement des données affichées dans le moniteur série se fait en passant en mode tableur en cliquant sur le bouton en haut du panneau central :



Le tableur est affiché dans le panneau central et les données reçues par le moniteur série sont visibles dans le panneau de droite :

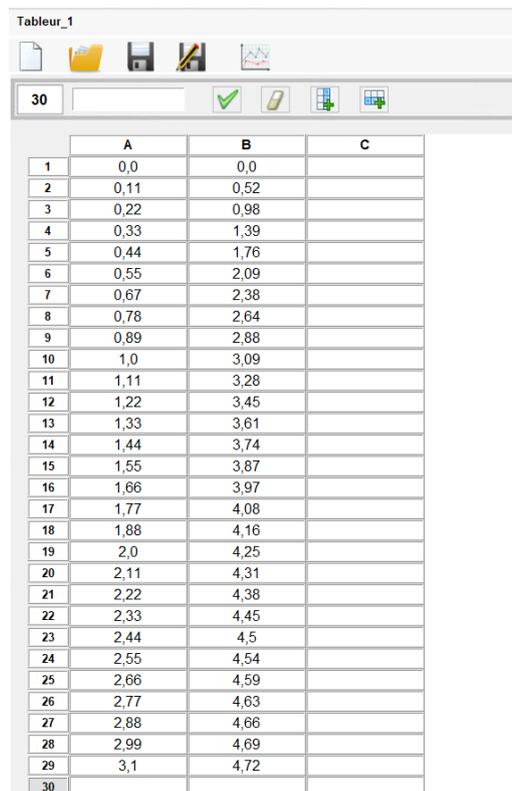


Pour transférer les données reçues dans le moniteur série, il faut sélectionner les données que l'on souhaite conserver et exploiter, puis cliquer sur le bouton .



The screenshot shows a software interface with two main windows. The left window, titled 'Tableur_1', contains a spreadsheet with a single cell 'A' in row 1. The right window, titled 'Données envoyées par l'Arduino UNO', displays a list of data points in a blue background. The data points are pairs of values: a time value followed by a voltage value (e.g., '0:00:0,00', '0:11:0,52', etc.). A green checkmark icon is visible in the top right corner of the interface.

Les données sélectionnées sont alors transférées dans le tableur :



The screenshot shows the spreadsheet 'Tableur_1' with 30 rows and 3 columns (A, B, C). The data is as follows:

	A	B	C
1	0,0	0,0	
2	0,11	0,52	
3	0,22	0,98	
4	0,33	1,39	
5	0,44	1,76	
6	0,55	2,09	
7	0,67	2,38	
8	0,78	2,64	
9	0,89	2,88	
10	1,0	3,09	
11	1,11	3,28	
12	1,22	3,45	
13	1,33	3,61	
14	1,44	3,74	
15	1,55	3,87	
16	1,66	3,97	
17	1,77	4,08	
18	1,88	4,16	
19	2,0	4,25	
20	2,11	4,31	
21	2,22	4,38	
22	2,33	4,45	
23	2,44	4,5	
24	2,55	4,54	
25	2,66	4,59	
26	2,77	4,63	
27	2,88	4,66	
28	2,99	4,69	
29	3,1	4,72	
30			

Il est alors conseillé d'enregistrer les données du tableur dans un fichier CSV.

5.2 Utilisation du tableur

Le mode d'utilisation du tableur d'ArdPylog est similaire à celui des autres tableurs qui servent généralement pour l'exploitation des expériences de sciences physiques.

5.2.1 La barre de menu de gestion des fichiers CSV

La barre de menu pour la gestion des fichiers CSV est constituée de 5 boutons :



-  : Crée un nouveau fichier CSV pour une utilisation en mode manuel,
-  : Ouvre un fichier CSV,
-  : Enregistre le fichier CSV sous le nom déjà défini. Dans le cas d'un nouveau tableur, le nom et le chemin du fichier est demandé,
-  : Enregistre le tableur sous un nom et un chemin spécifique,
-  : Affiche la fenêtre d'exploitation graphique du fichier CSV (voir page 38).

5.2.2 La barre de gestion des variables et des données du tableur

En dessous de la barre de menu pour la gestion des fichiers CSV, se trouve la barre de gestion des variables et des données du tableur :



Celle-ci est constituée de 4 boutons, d'une zone de texte et d'un numéro.

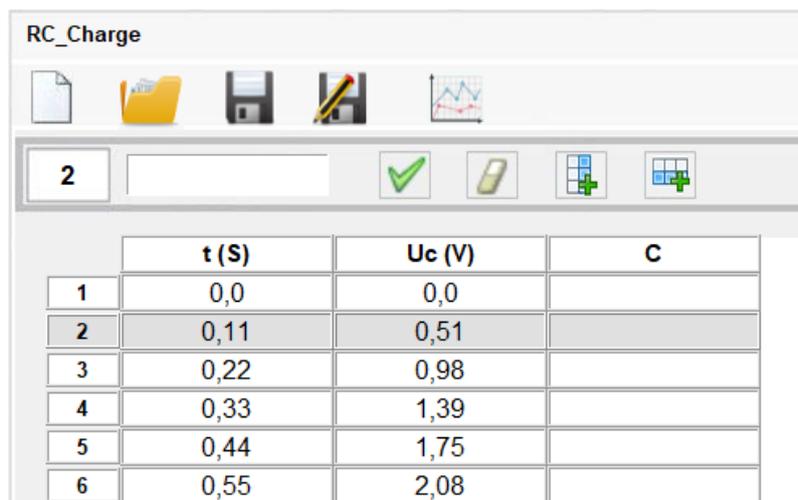
- Le numéro correspond au numéro de la ligne du tableur sélectionnée,
- La zone de texte sert à entrer manuellement des valeurs dans une case du tableur sélectionnée,

-  : sert à transférer les données du moniteur série vers le tableur ou à valider les valeurs saisies manuellement dans la zone de texte,
-  : Efface la ligne du tableur sélectionnée,
-  : Ajoute une colonne au tableur (à utiliser pour ajouter une variable),
-  : Ajoute une ligne au tableur pour une saisie manuelle des valeurs.

5.2.3 Manipulation des données et des variables

- Sélection d'une ligne du tableur :

Pour sélectionner une ligne, il faut cliquer sur le numéro de la ligne correspondant.



	t (S)	Uc (V)	C
1	0,0	0,0	
2	0,11	0,51	
3	0,22	0,98	
4	0,33	1,39	
5	0,44	1,75	
6	0,55	2,08	

Cette fonction est utile pour effacer la ligne de données sélectionnée en cliquant ensuite sur .

- Sélection d'une case du tableur :

Pour sélectionner une case du tableur, il faut effectuer un double clic dessus. Si la case contient une valeur, celle-ci est affichée dans la zone de texte :

	t (S)	Uc (V)	C
1	0,0	0,0	
2	0,11	0,51	
3	0,22	0,98	

La valeur peut être modifiée, soit à partir de la zone de texte en validant avec  ou avec la touche « **Entrée** », soit directement à partir de la case sélectionnée en validant avec la touche « **Entrée** ».

Et si initialement la case est vide, une valeur peut être saisie de la même façon.

- Sélection d'une colonne du tableur :

La sélection d'une colonne du tableur et donc d'une variable du tableur se fait par un double clic sur l'entête de la colonne.

	t (S)	Uc (V)	C
1	0,0	0,0	
2	0,11	0,51	
3	0,22	0,98	
4	0,33	1,39	
5	0,44	1,75	
6	0,55	2,08	

Une fenêtre permettant de modifier la variable sélectionnée est affichée dans le panneau de droite :

Tableur - Paramètres

Modifier la variable

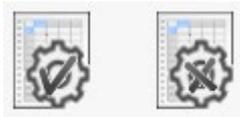
. Nom :

. Unité :

. Expression mathématique :

Le nom de la variable, l'unité et une éventuelle expression mathématique peuvent ainsi être modifiés.

Les modifications sont validées ou annulées en cliquant respectivement sur les boutons de la fenêtre « **Tableur – Paramètres** » suivant:

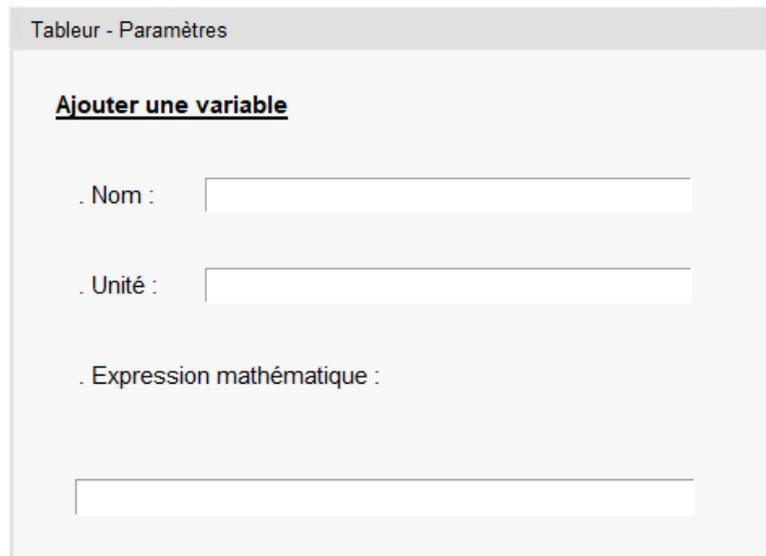


- **Ajout d'une variable** :

Pour ajouter une variable et donc une colonne au tableur, il existe 2 possibilités :

- . Cliquer sur le bouton ,
- . Double clic sur l'entête de la dernière colonne.

Une fenêtre permettant d'ajouter une variable est alors affichée dans le panneau de droite :



L'ajout de la nouvelle variable est validé ou annulé de la même façon que pour la modification d'une variable

Remarques :

Dans les expressions mathématiques appliquées aux variables, certaines règles sont à respecter :

- . Utilisation du point pour la virgule des nombres décimaux,
- . Respect de la casse des noms des variables intervenant dans les expressions mathématiques,
- . Utilisation des opérateurs mathématiques suivants :

- Addition : +
- Soustraction : -
- Multiplication : *
- Division : /

. Utilisation des fonctions mathématiques suivantes :

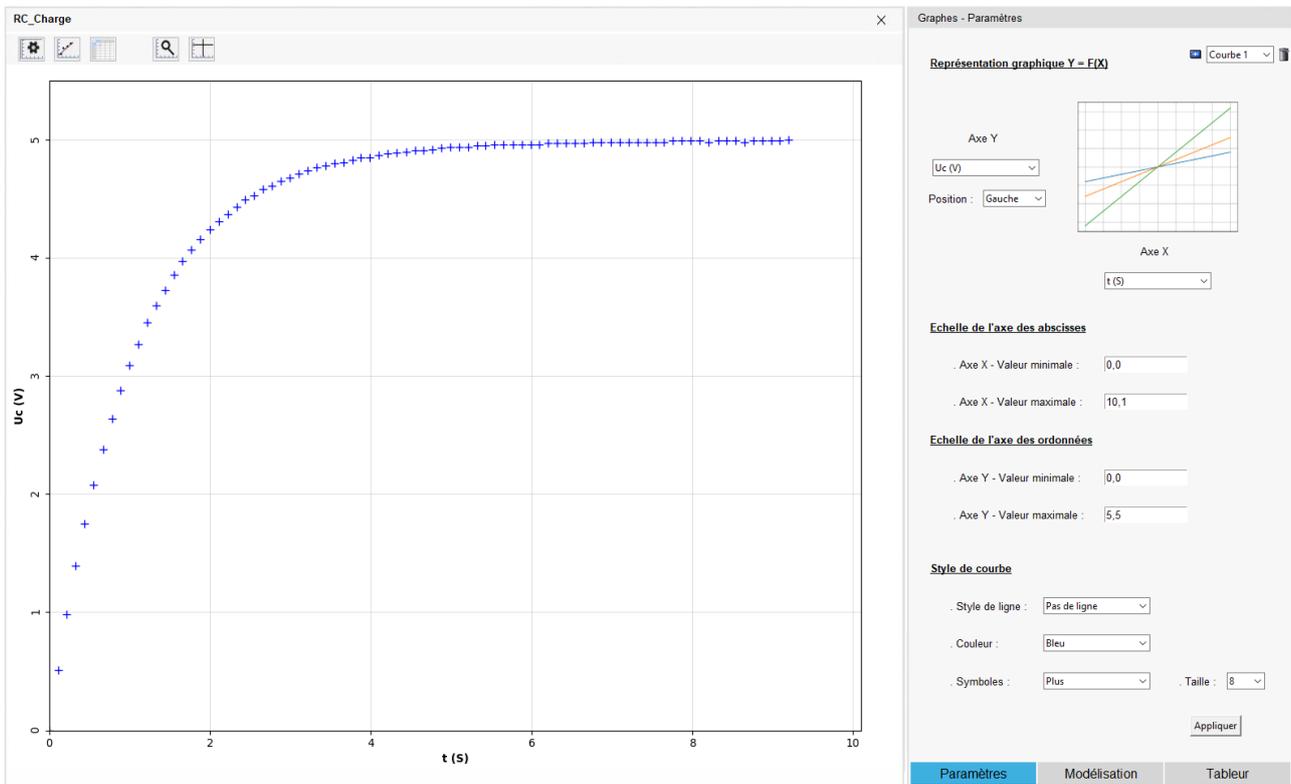
- Fonction puissance : **
- Fonction racine carrée : **sqrt()**
- Fonction valeur absolue : **abs()**
- Fonction Logarithmique népérien : **log()**
- Fonction exponentielle : **exp()**
- Fonction cosinus : **cos()**
- Fonction sinus : **sin()**
- Fonction tangente : **tan()**
- Fonction arc cosinus : **arccos()**
- Fonction arc sinus : **arcsin()**
- Fonction arc tangente : **arctan()**

. Utilisation du nombre π : **pi**

5.3 Exploitation graphique des données du tableur

L'exploitation graphique des données du tableur est réalisable en cliquant sur : 

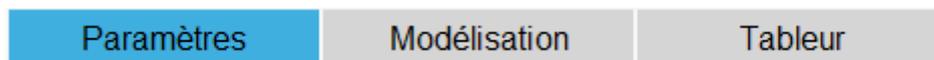
Par défaut, Le graphe représentant la variable de la 2^{ème} colonne en fonction de la variable de la 1^{ère} colonne du tableur est affiché au niveau du panneau central :



Exemple de graphe : Charge du condensateur d'un dipôle RC - $U_c = f(t)$

Dans le panneau de droite, on retrouve 3 onglets permettant de modifier les paramètres du graphe, d'effectuer une modélisation ou de visualiser les valeurs des points expérimentaux tracés et éventuellement modélisés dans un tableur.

La navigation entre les 3 onglets se fait à l'aide de la barre de menu :



Ou des boutons de la barre de menu du graphe :

-  : Affiche l'onglet « Paramètres »

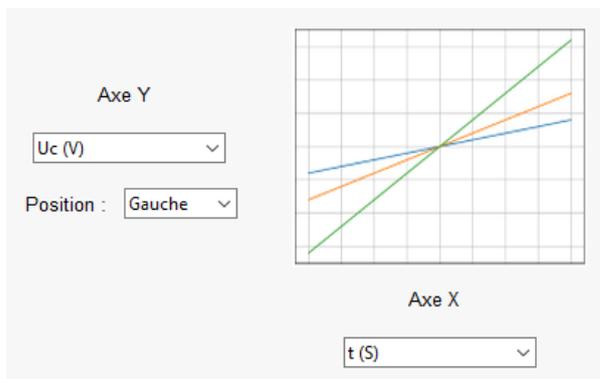
-  : Affiche l'onglet « Modélisation »

-  : Affiche l'onglet « Tableur »

5.3.1 Modification des paramètres du graphe

A partir de l'onglet « **Paramètres** », il est possible de modifier à l'aide de listes déroulantes et de zones de texte :

- La variable de l'axe des abscisses **X** de la représentation graphique de la fonction **Y = F(X)**,
- La variable de l'axe des ordonnées **Y** de la représentation graphique de la fonction **Y = F(X)**,
- La position de l'axe des ordonnées (gauche ou droite du graphe) :



- L'échelle des axes des abscisses et des ordonnées :

Echelle de l'axe des abscisses

. Axe X - Valeur minimale :

. Axe X - Valeur maximale :

Echelle de l'axe des ordonnées

. Axe Y - Valeur minimale :

. Axe Y - Valeur maximale :

- le style de la courbe :

Style de courbe

. Style de ligne :

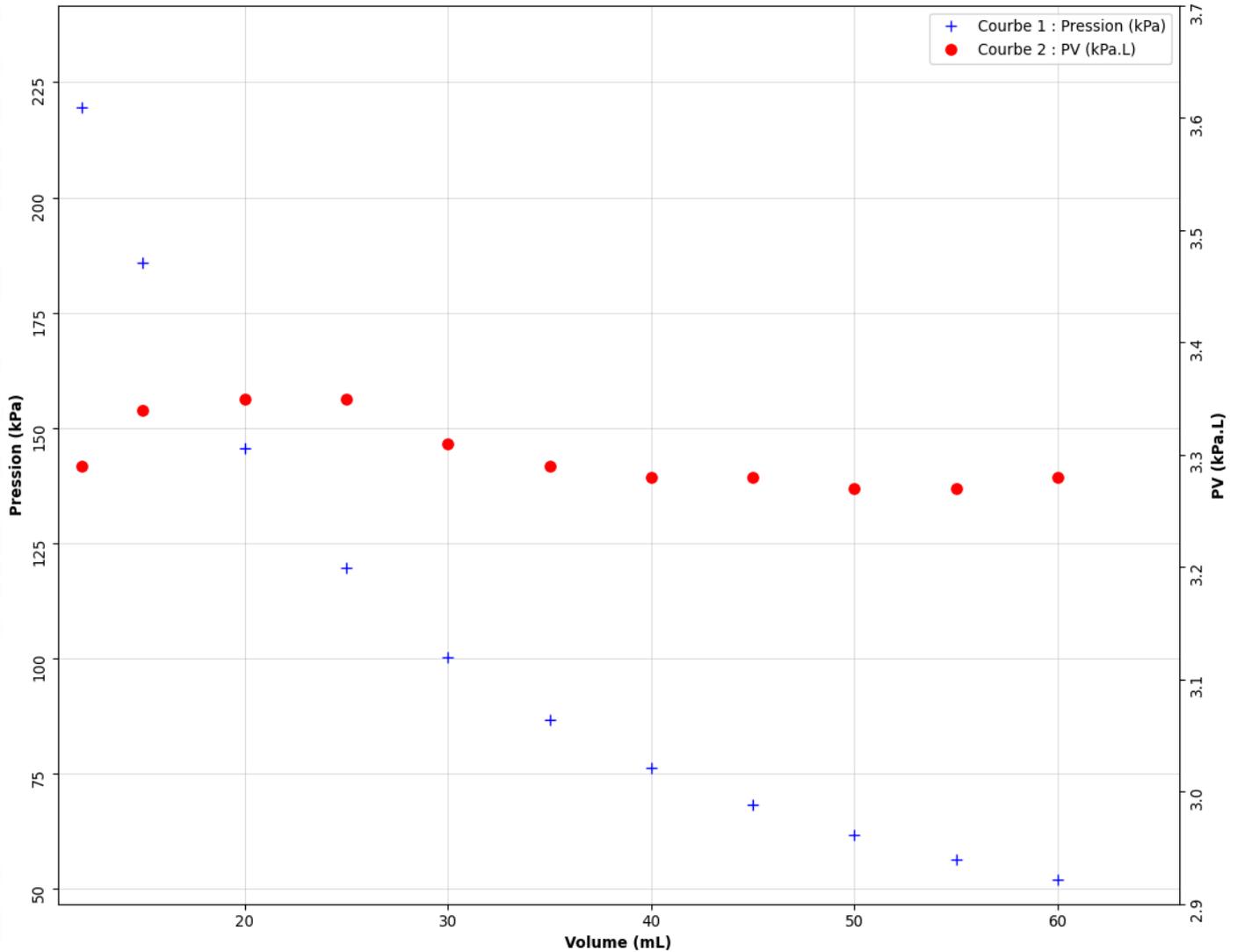
. Couleur :

. Symboles : . Taille :

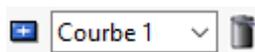
Les modifications sont validées en cliquant sur le bouton « **Appliquer** ».

Toujours à partir de l'onglet « Paramètres », il est également possible d'ajouter des courbes en cliquant sur , si le nombre de variables du tableur le permet (>2).

L'axe des abscisses de la nouvelle courbe sera alors le même que celui de la courbe précédente. L'axe des ordonnées correspondra à une variable du tableur non utilisée.



Pour modifier les paramètres des différentes courbes, il faut au préalable sélectionner une courbe à l'aide la liste déroulante :



La courbe ainsi sélectionnée peut être supprimée en cliquant sur : 

5.3.2 Modélisation des fonctions $Y=F(X)$

A partir de l'onglet « **Modélisation** », il est possible de modéliser la fonction $Y=F(X)$ sélectionnée à l'aide de la liste déroulante et représentée graphiquement.

Graphes - Modélisation

Courbe 1

Modélisation $Y = F(X)$

Linéaire
 $y = a \cdot x$

Affine
 $y = a \cdot x + b$

Polynomiale
 $y = a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \dots + a \cdot x + b$
Ordre: 2

Exponentielle
 $y = a \cdot e^{b \cdot x}$

Logarithmique
 $y = a \cdot \ln x + b$

Puissance

Appliquer

Paramètres Modélisation Tableur

Les modèles de fonction disponibles sont :

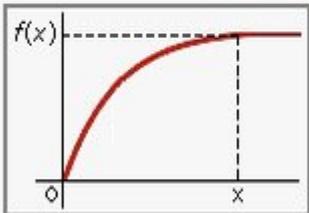
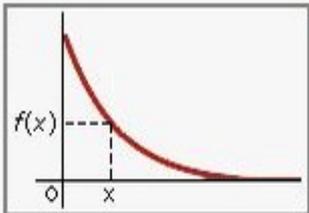
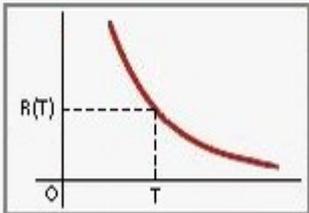
- Fonction linéaire,
- Fonction affine,
- Fonction polynomiale (jusqu'à l'ordre 4),
- Fonction exponentielle,
- Fonction logarithmique,
- Fonction puissance,

La sélection du modèle est réalisée en cliquant sur la case à cocher correspondante et en cliquant sur le bouton « **Appliquer** ».

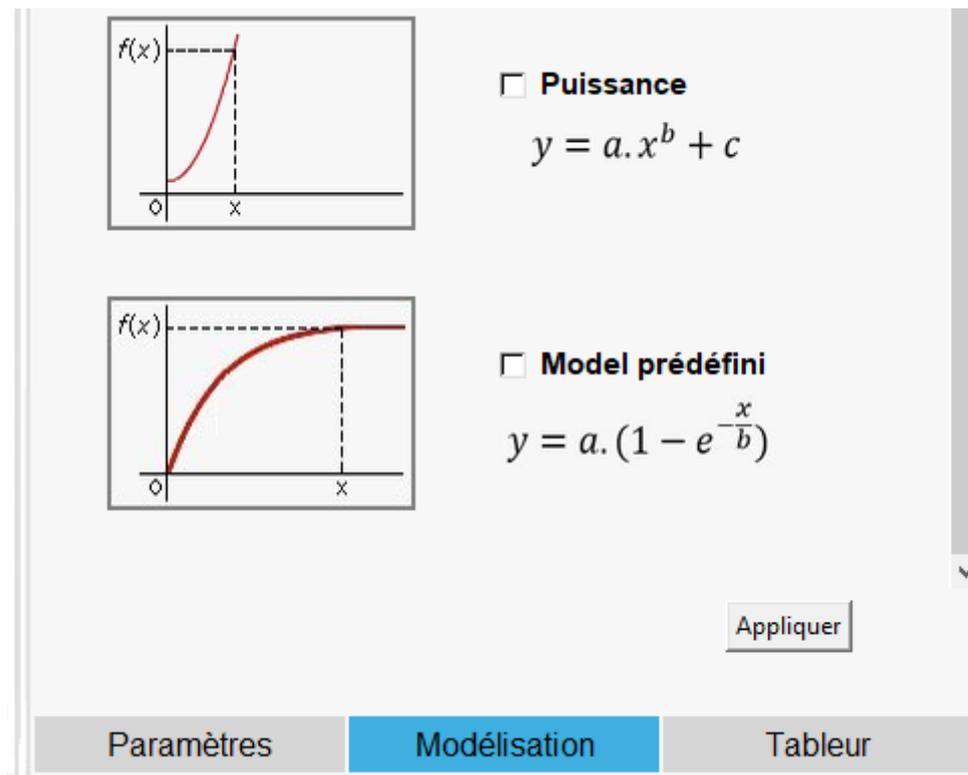
Sont aussi disponibles, des modèles de fonction prédéfinis utiles pour certaines expériences :

Graphes - Modélisation prédéfinies

Modélisation $Y = F(X)$

	Dipole RC - Charge C $y = a. (1 - e^{-\frac{x}{b}})$
	Dipole RC - Décharge C $y = a. e^{-\frac{x}{b}}$
	Relation simplifiée de Steinhart-Hart $R = R_0 . e^{\beta(\frac{1}{T} - \frac{1}{T_0})}$ Ro (Ohms) = <input type="text"/> To (Kelvins) = <input type="text"/>

Pour sélectionner un modèle prédéfini, il faut cliquer sur l'image correspondant au modèle souhaité après avoir cliqué sur l'image « **Modèle prédéfini** » :



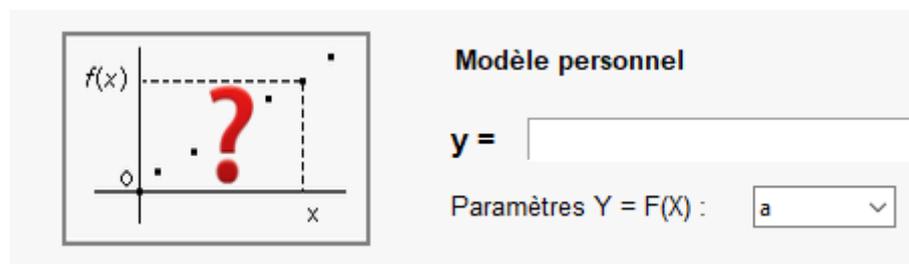
Puissance
 $y = a.x^b + c$

Modèle prédéfini
 $y = a.(1 - e^{-\frac{x}{b}})$

Appliquer

Paramètres Modélisation Tableur

Il est également possible de créer son propre modèle à partir de l'onglet des modèles prédéfinis :



Modèle personnel

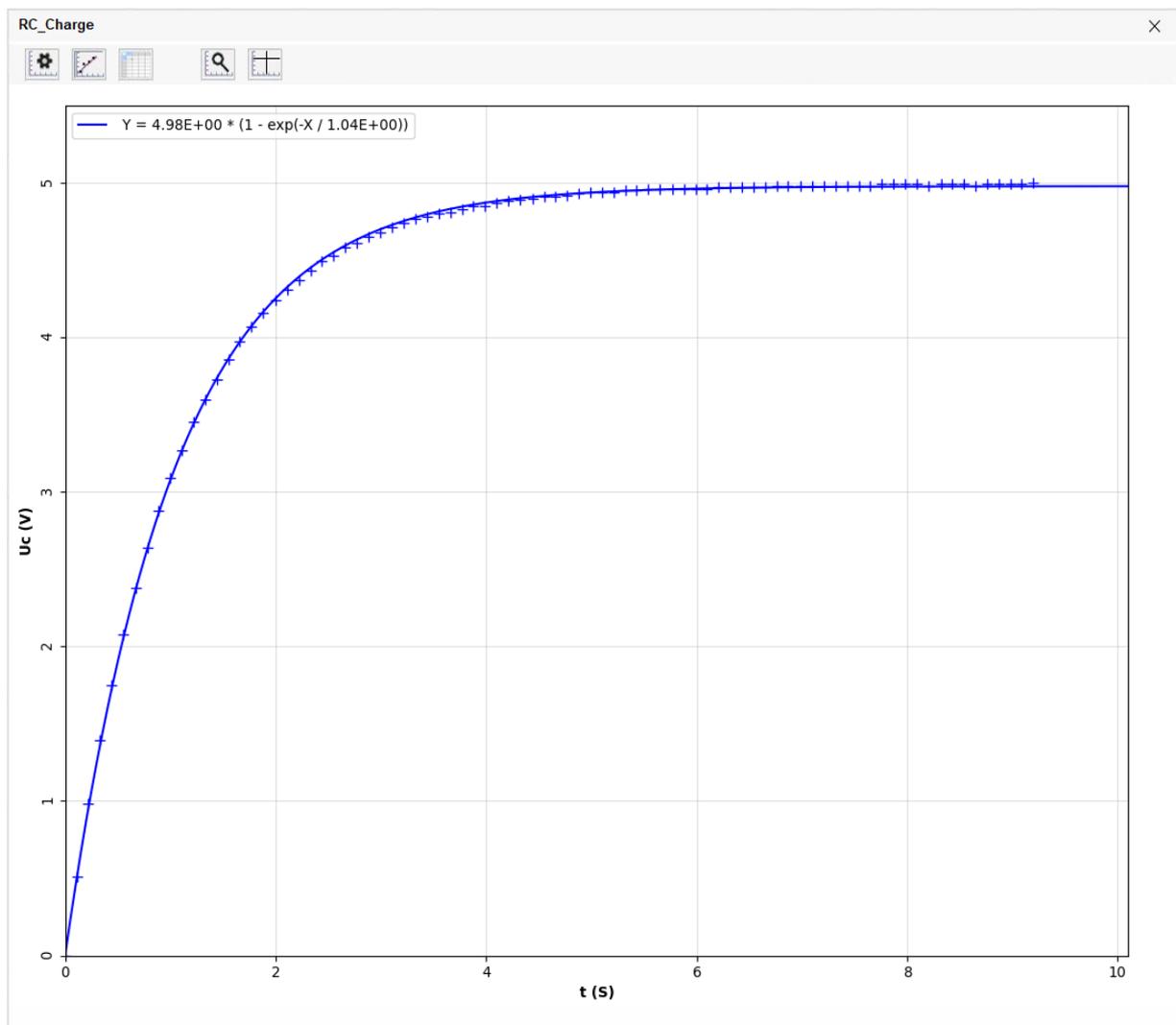
$y =$

Paramètres $Y = F(X)$:

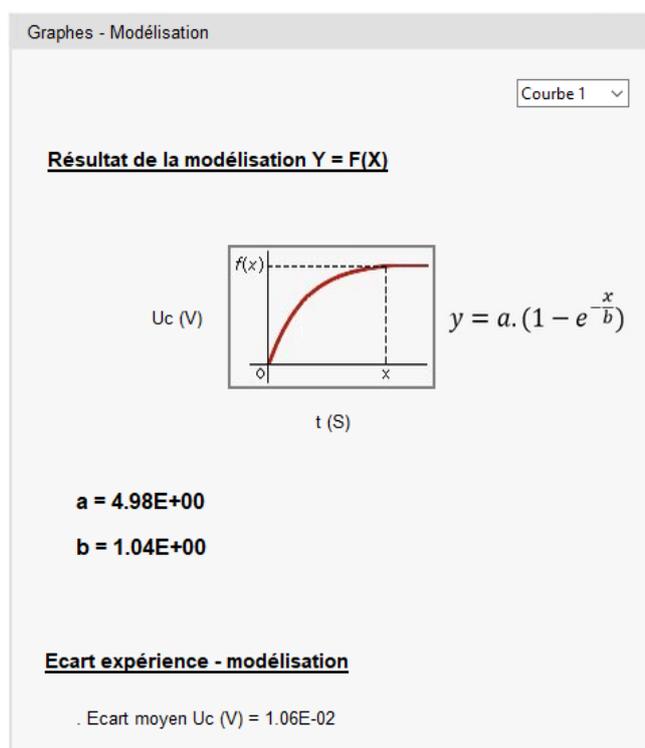
Après avoir sélectionné le nombre de paramètres du modèle (**a** ou **a,b** ou **a,b,c** ou **a,b,c,d**), la fonction mathématique doit être saisie suivant les mêmes règles que les expressions mathématiques des variables du tableur (voir page 34) et validée à l'aide de la touche « Entrée ».

Une fois le modèle prédéfini choisi ou son propre modèle créé, il faut cliquer sur le bouton « **Appliquer** » pour tracer sa représentation graphique.

Exemple : Charge du condensateur d'un dipôle RC - $U_c = f(t)$



Le résultat de la modélisation est affiché dans le panneau de droite :

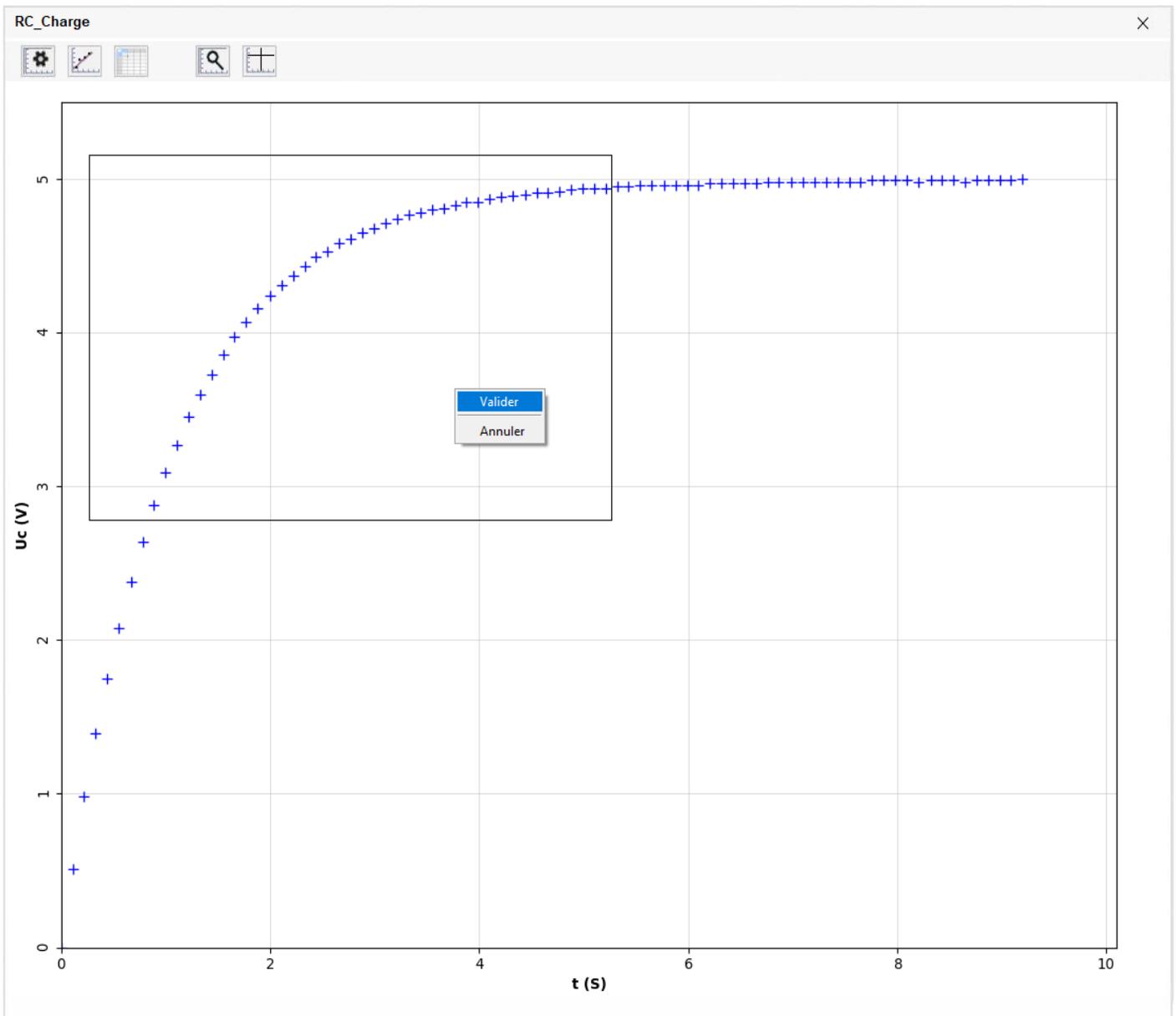


5.3.3 Les outils graphiques

La barre de menu du graphe dispose également de 2 boutons correspondants à des outils graphiques :

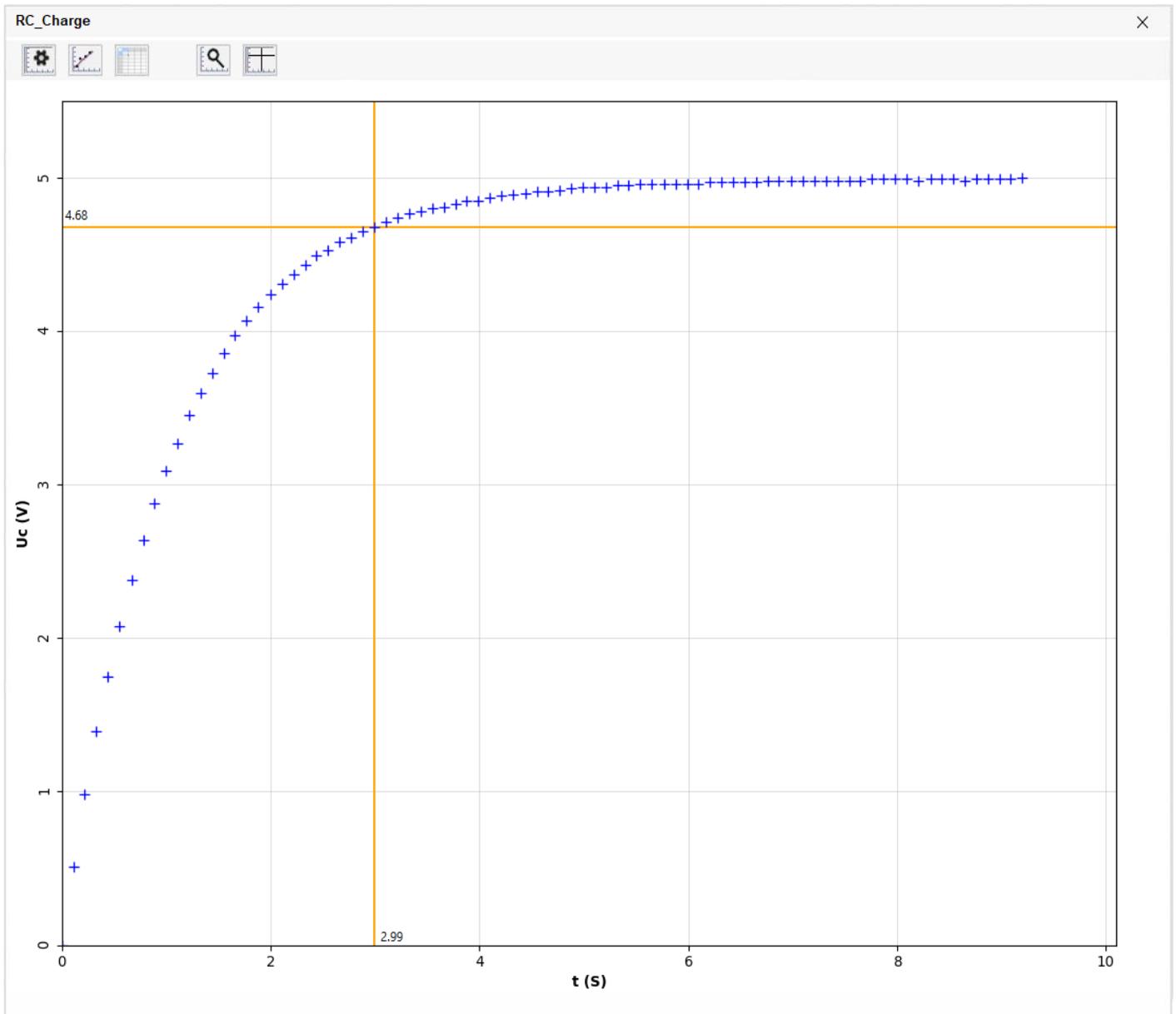
-  : effectue un zoom sur une partie du graphe

Après avoir cliqué sur ce bouton, la sélection de la zone du zoom est réalisée avec la souris en maintenant le bouton gauche appuyé et validée ou annulée en cliquant sur son bouton droit dans la zone sélectionnée.



-  : Affiche les curseurs

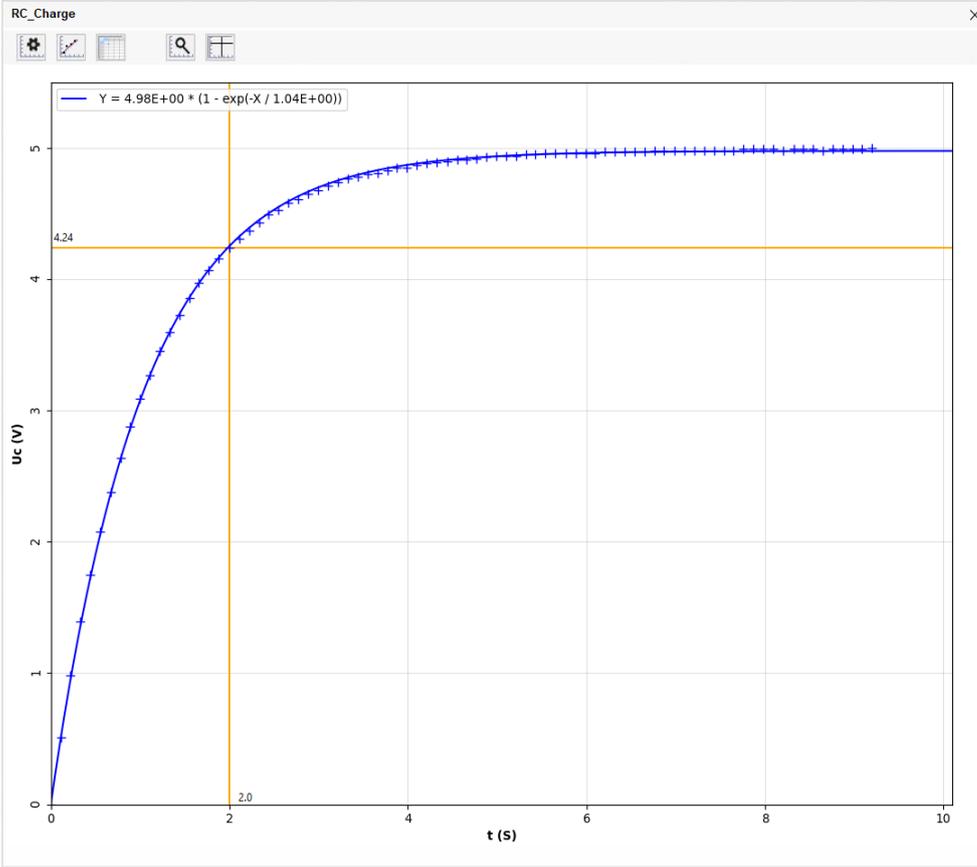
Après avoir cliqué sur ce bouton, les curseurs sont déplacés en survolant le graphe avec la souris.



5.3.4 L'onglet « Tableur »

L'onglet « Tableur » permet de visualiser les valeurs des points expérimentaux tracés et éventuellement modélisés.

La sélection d'une ligne du tableur affiche les curseurs du graphe sur le point de mesure correspondant.



Graphes - Tableau

Courbe 1

	t (S)	Uc (V)	Uc (V) modelisé(e)
1	0,0	0,0	0,0
2	0,11	0,51	0,5
3	0,22	0,98	0,95
4	0,33	1,39	1,36
5	0,44	1,75	1,72
6	0,55	2,08	2,05
7	0,67	2,38	2,37
8	0,78	2,64	2,63
9	0,89	2,88	2,87
10	1,0	3,09	3,08
11	1,11	3,27	3,27
12	1,22	3,45	3,44
13	1,33	3,6	3,59
14	1,44	3,73	3,73
15	1,55	3,86	3,86
16	1,66	3,97	3,97
17	1,77	4,07	4,07
18	1,88	4,16	4,16
19	2,0	4,24	4,25
20	2,11	4,31	4,33
21	2,22	4,37	4,39
22	2,33	4,43	4,45
23	2,44	4,49	4,5
24	2,55	4,53	4,55
25	2,66	4,58	4,59
26	2,77	4,61	4,63
27	2,88	4,65	4,67
28	2,99	4,68	4,7
29	3,1	4,71	4,73
30	3,22	4,74	4,75
31	3,33	4,77	4,78
32	3,44	4,78	4,8
33	3,55	4,8	4,81
34	3,66	4,81	4,83

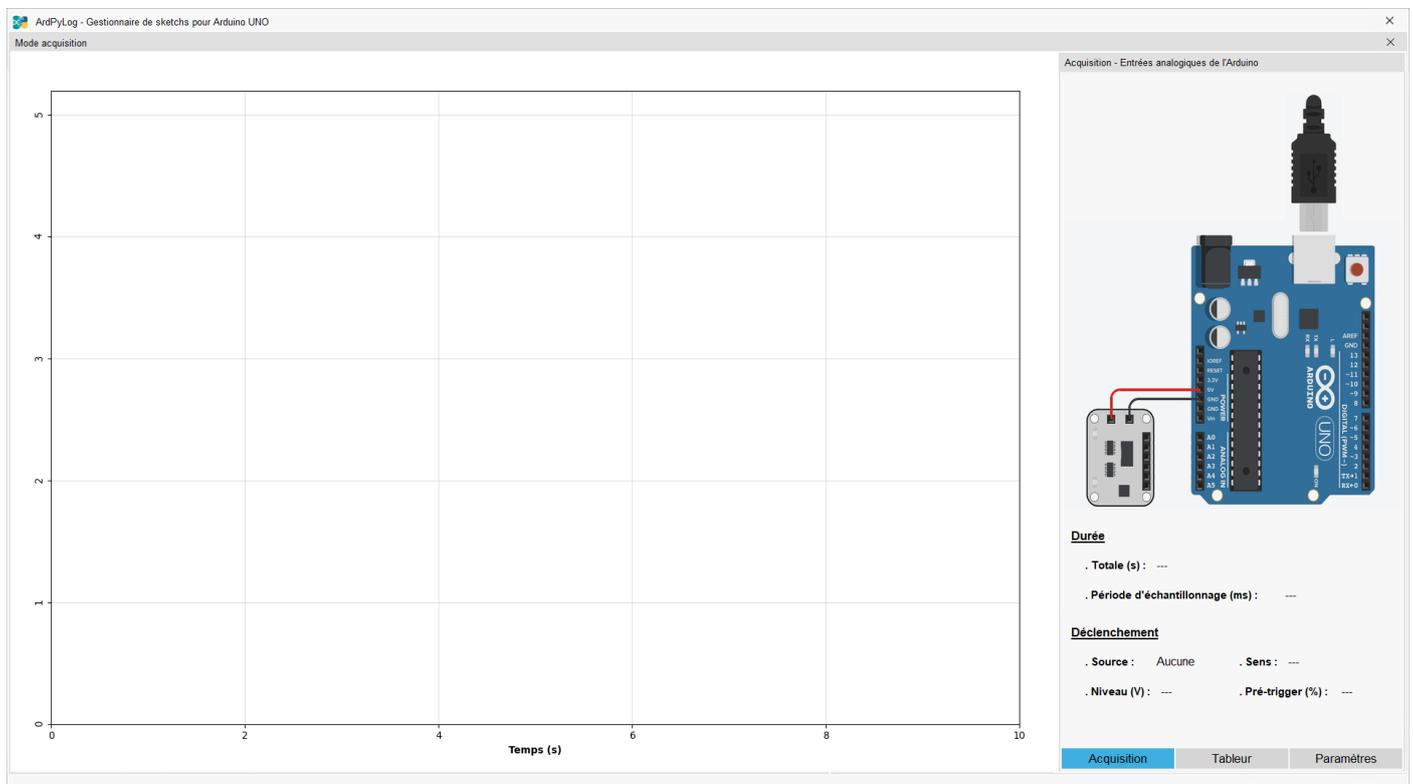
Paramètres Modélisation **Tableur**

6. Le mode « Acquisition »

Le mode « Acquisition » d'ArdPyLog transforme l'Arduino Uno en centrale d'acquisition grâce à l'utilisation de ses entrées analogiques.

Les tensions mesurées sur les entrées analogiques sélectionnées sont acquises pendant la durée et le taux d'échantillonnage définis par l'utilisateur.

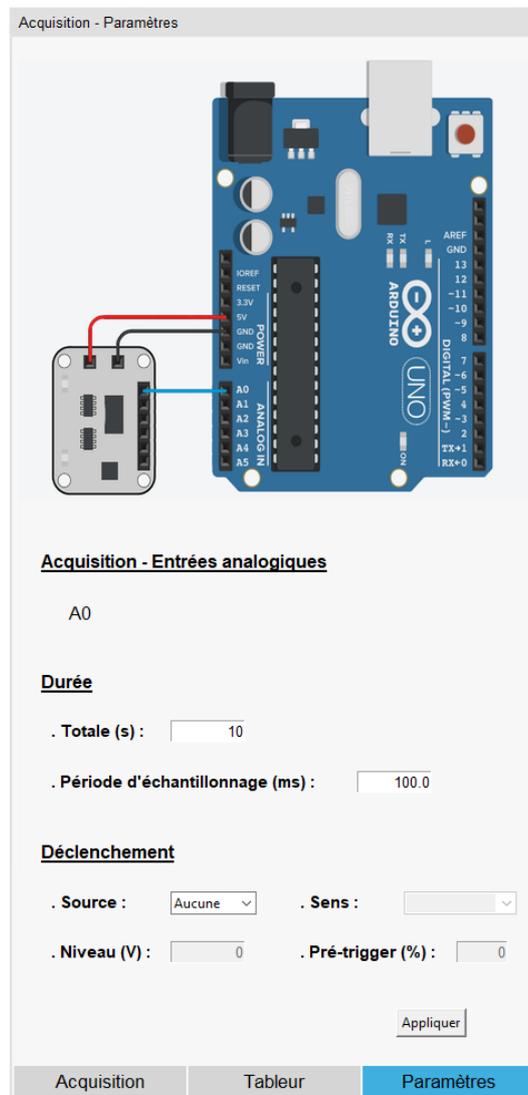
Le mode « Acquisition » est accessible en cliquant sur :
dans la barre de menu principale :



Fenêtre Mode « Acquisition »

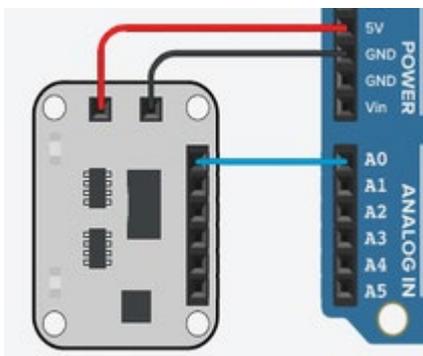
6.1 Définition ou modification des paramètres de l'acquisition

Avant d'effectuer une acquisition, il faut définir les paramètres de l'acquisition à l'aide de l'onglet « Paramètres » du panneau de droite.



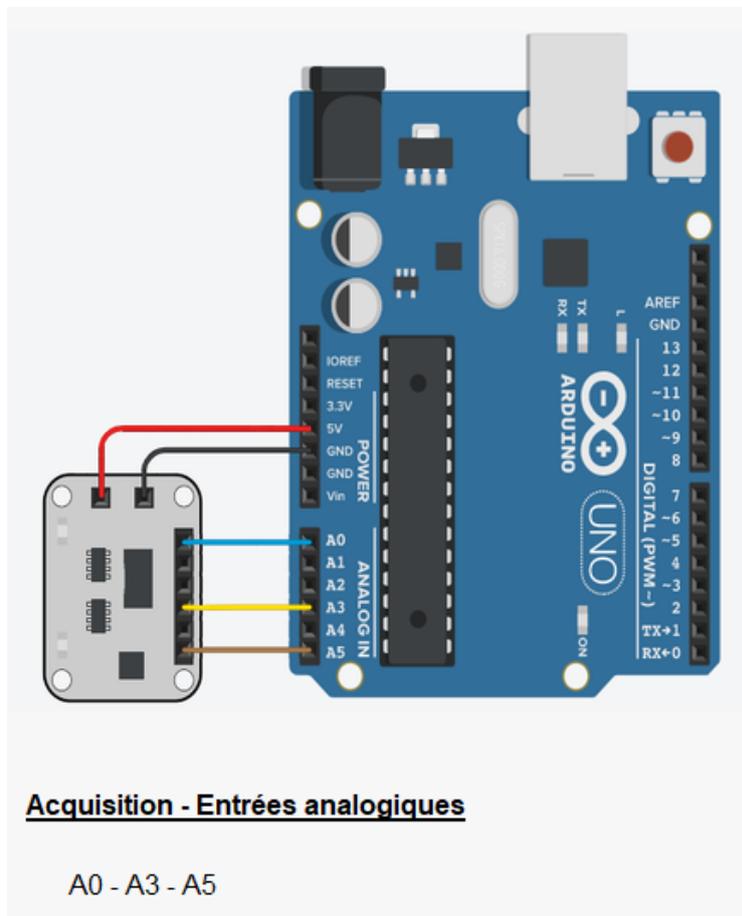
Dans un premier temps, il faut sélectionner les entrées analogiques de l'Arduino Uno pour lesquelles on souhaite suivre l'évolution temporelle des tensions qui leur sont appliquées.

Pour sélectionner une entrée analogique, il faut cliquer sur la broche correspondante (A0, A1, ..., A5) de l'image de l'Arduino Uno ou du capteur représenté symboliquement :



Une entrée analogique sélectionnée peut-être désélectionnée en cliquant à nouveau dessus.

La liste des entrées analogiques sélectionnées est affichée dans la partie « **Acquisition – Entrées analogiques** » :



Ensuite, la durée totale en seconde et la période d'échantillonnage en ms doivent être indiquées.

Par défaut, la durée totale est définie sur 10 s et la période d'échantillonnage sur 100 ms.

Durée

. Totale (s) :

. Période d'échantillonnage (ms) :

Remarques :

Avec un Arduino Uno, pour obtenir des mesures suffisamment précises, la période d'échantillonnage ne pourra pas être inférieure à 100 ms.

Par conséquent, la durée totale ne pourra donc pas être inférieure à 0,1 s.

Enfin, il est possible de définir un seuil de déclenchement.

L'acquisition ne débutera que si la tension mesurée est supérieure ou inférieure (en fonction du sens choisi, « **Montant** » ou « **Descendant** ») à une valeur seuil définie (**Niveau**) sur une des entrées analogiques sélectionnées (**Source**).

Déclenchement

. Source :	<input type="text" value="A0"/>	. Sens :	<input type="text" value="Montant"/>
. Niveau (V) :	<input type="text" value="0.05"/>	. Pré-trigger (%) :	<input type="text" value="1"/>

Remarque :

Le pré-trigger correspondant à la durée en pourcentage de la durée totale d'acquisition pendant laquelle les tensions sont enregistrées et représentées graphiquement avant que le déclenchement (trigger) ait lieu.

Les paramètres ainsi définis sont validés et enregistrés en cliquant sur le bouton « **Appliquer** ».

Attention :

Avant de valider les paramètres, il est indispensable que l'Arduino Uno soit connecté à l'ordinateur, le programme permettant de mesurer les tensions sur les entrées analogiques sélectionnées étant téléversé à ce moment.

Une fois le programme d'acquisition téléversé, l'onglet « **Acquisition** » du panneau de droite est affiché, afin de pouvoir procéder à l'acquisition.

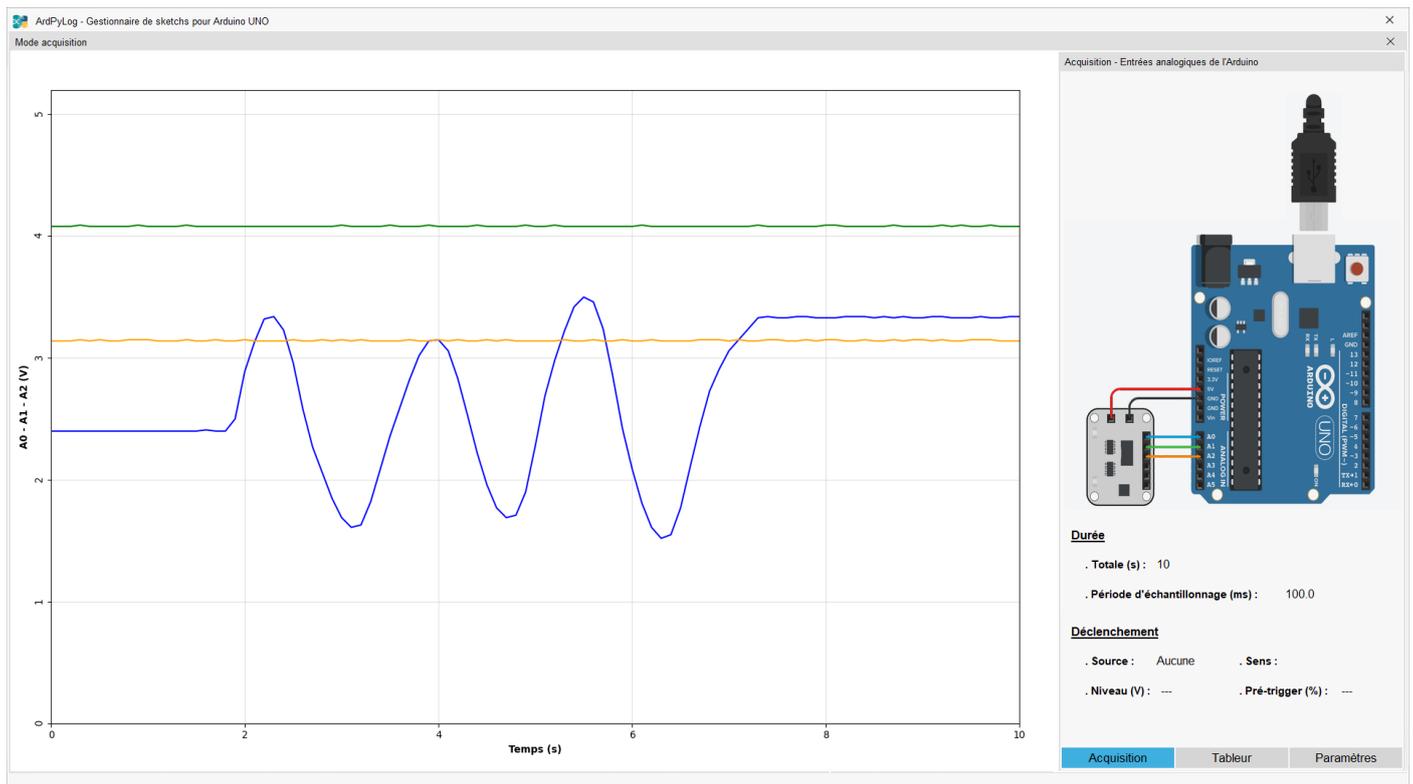
6.2 Acquisition des données

Une fois les paramètres définis et téléversés, l'acquisition est démarrée en cliquant sur la prise USB :



L'acquisition en cours peut être interrompu à tout moment en appuyant sur la touche « **Echap** ou **ESC** ».

Au cours de l'acquisition, les tensions mesurées sur les entrées analogiques de l'Arduino Uno sélectionnées sont représentées graphiquement en fonction du temps :



Une autre acquisition avec les mêmes paramètres peut être réalisée en re-cliquant sur la prise USB et cela autant de fois que souhaitées.

Pour modifier les paramètres de l'acquisition, il faudra sélectionner l'onglet « **Paramètres** » et suivre les instructions du paragraphe « **6.1 Définition ou modification des paramètres de l'acquisition** »

6.3 Enregistrement et exploitation des mesures

Après l'acquisition, les tensions mesurées sur les entrées analogiques de l'Arduino Uno sélectionnées peuvent être visualisées dans l'onglet « **Tableur** » du panneau de droite.

Acquisition - Tableur



	t (s)	A0 (V)	A1 (V)	A2 (V)
1	0	2.04	4.08	3.15
2	0.1	2.05	4.08	3.14
3	0.2	2.05	4.08	3.15
4	0.3	2.04	4.08	3.14
5	0.4	2.04	4.08	3.14
6	0.5	2.04	4.09	3.14
7	0.6	2.05	4.09	3.14
8	0.7	2.04	4.08	3.14
9	0.8	2.03	4.09	3.14
10	0.9	1.95	4.08	3.14
11	1.0	1.71	4.08	3.14
12	1.1	1.39	4.08	3.15
13	1.2	1.14	4.08	3.15
14	1.3	0.94	4.08	3.14
15	1.4	0.82	4.09	3.14
16	1.5	0.82	4.09	3.14
17	1.6	1.05	4.08	3.14
18	1.7	1.38	4.08	3.14
19	1.8	1.8	4.08	3.14
20	1.9	2.12	4.08	3.15
21	2.0	2.48	4.08	3.14
22	2.1	2.82	4.09	3.15
23	2.2	3.1	4.08	3.14
24	2.3	3.17	4.08	3.14

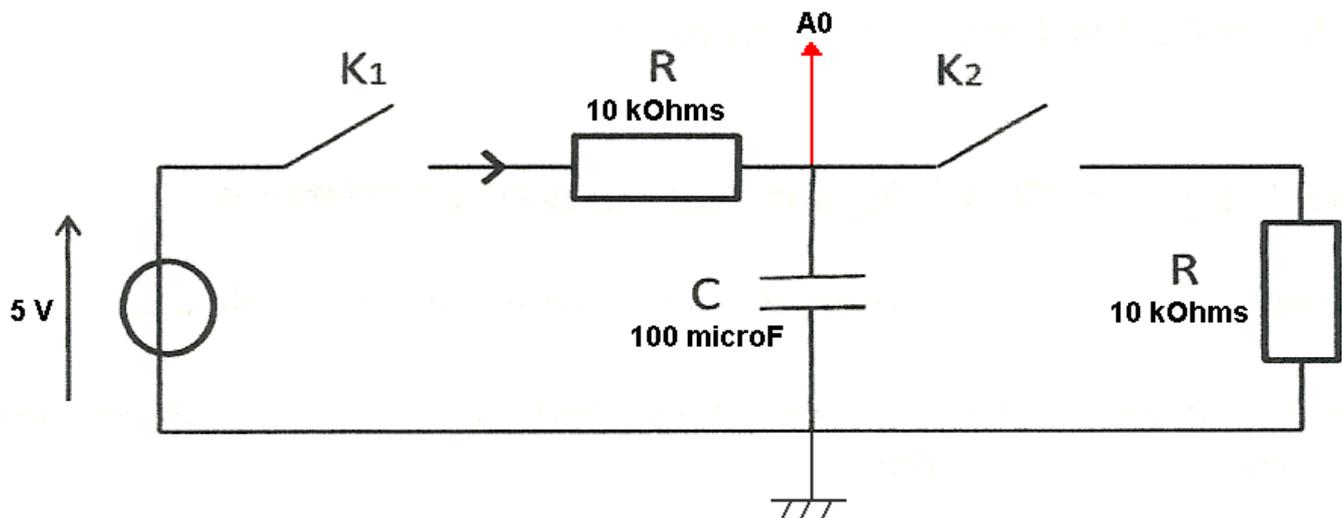
Ces données peuvent être soit exploitées à l'aide du mode « Tableur » d'ArdPylog (voir page 30), soit directement enregistrées au format **.CSV** pour une exploitation ultérieure.

La barre de gestion des données dispose de 2 boutons :

-  : Transfère les données acquises dans le tableur d'ArdPylog où il sera possible de les enregistrer et d'effectuer une modélisation,
-  : Enregistre les données acquises dans un fichier au format **.CSV** qu'il sera possible d'ouvrir ultérieurement avec le tableur d'ArdPylog.

6.4 Exemples d'acquisition (charge et décharge du condensateur d'un dipôle RC)

Dans cet exemple d'acquisition, nous allons suivre l'évolution temporelle de la tension aux bornes d'un condensateur d'un dipôle RC à l'aide d'un Arduino Uno dont l'entrée analogique A0 est connecté à une borne du condensateur selon le montage suivant :



Avec ce montage, quand K2 est ouvert et K1 fermé, le condensateur se charge à travers la résistance R et inversement, quand K1 est ouvert et K2 fermé, le condensateur se décharge dans la résistance R.

6.4.1 Charge du condensateur

Afin de suivre l'évolution temporelle de la tension aux bornes du condensateur pendant sa charge, on s'assure dans un premier temps qu'il est complètement déchargé en ouvrant K1 et en fermant K2.

Il faut ensuite définir les paramètres de l'acquisition :

- Entrées analogiques : **A0**
- Durée totale d'acquisition : **10 s**

En effet, la valeur théorique de τ (temps pour lequel le condensateur est chargé à 63 %) du dipôle RC est :

$$\tau = RC = 10 \cdot 10^3 \times 100 \cdot 10^{-6} = 1 \text{ S}$$

Avec une durée totale d'acquisition de 10 s, la charge sera complète.

- Période d'échantillonnage : **100 ms**

- Déclenchement :

Afin d'acquérir uniquement la tension aux bornes du condensateur au cours de sa charge, l'acquisition sera déclenchée quand la tension mesurée sur A0 sera supérieure à 0,05 V avec un pré-trigger de 1% (1% de 10 s, soit 100 ms) afin d'observer le début de la charge.

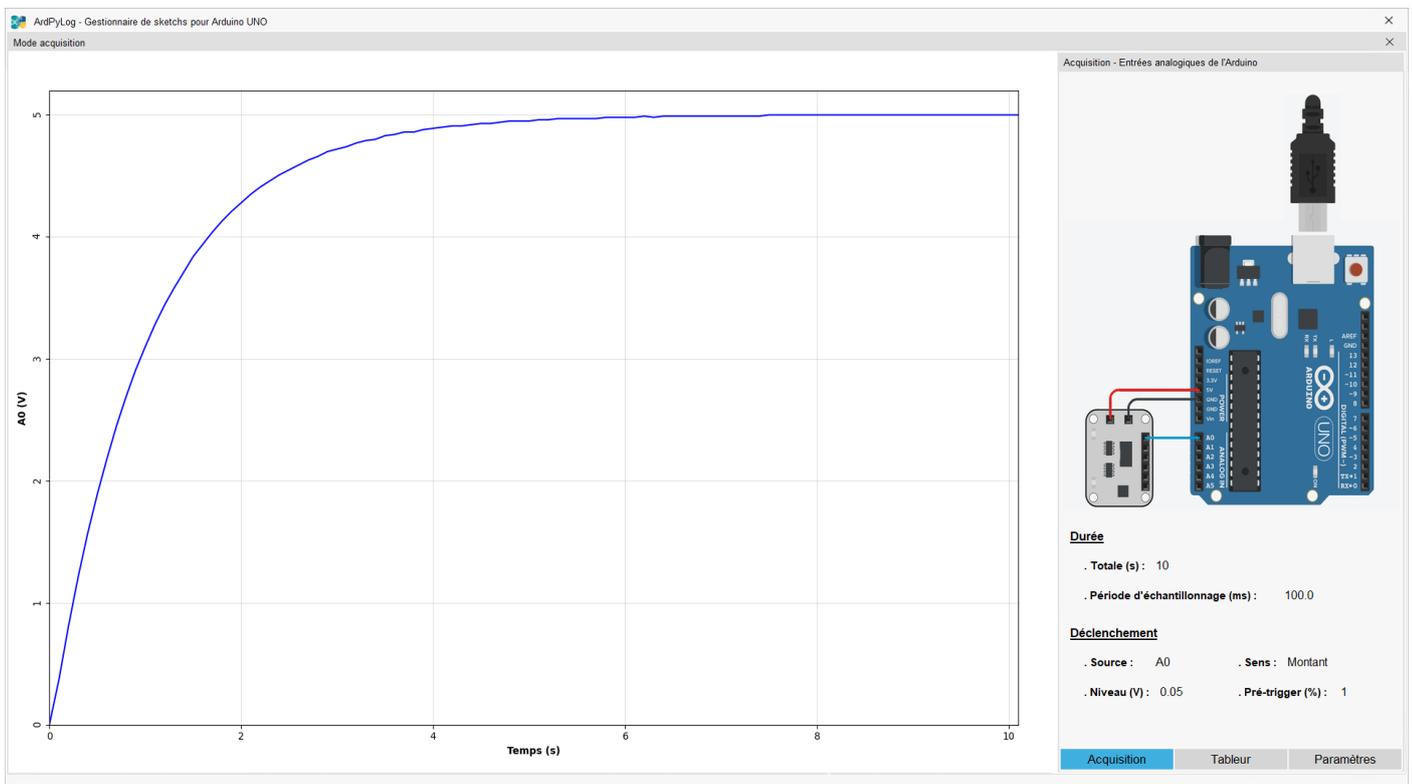
Les paramètres de déclenchement seront donc :

Déclenchement

. Source :	A0	. Sens :	Montant
. Niveau (V) :	0.05	. Pré-trigger (%) :	1

Après avoir validé les paramètres (bouton « **Appliquer** »), l'acquisition est démarrée (clic sur la prise USB).

K2 est alors ouvert et K1 fermé. Quand la tension U_c , aux bornes du condensateur, est supérieur à 0,05 V, l'évolution temporelle de U_c est tracée.

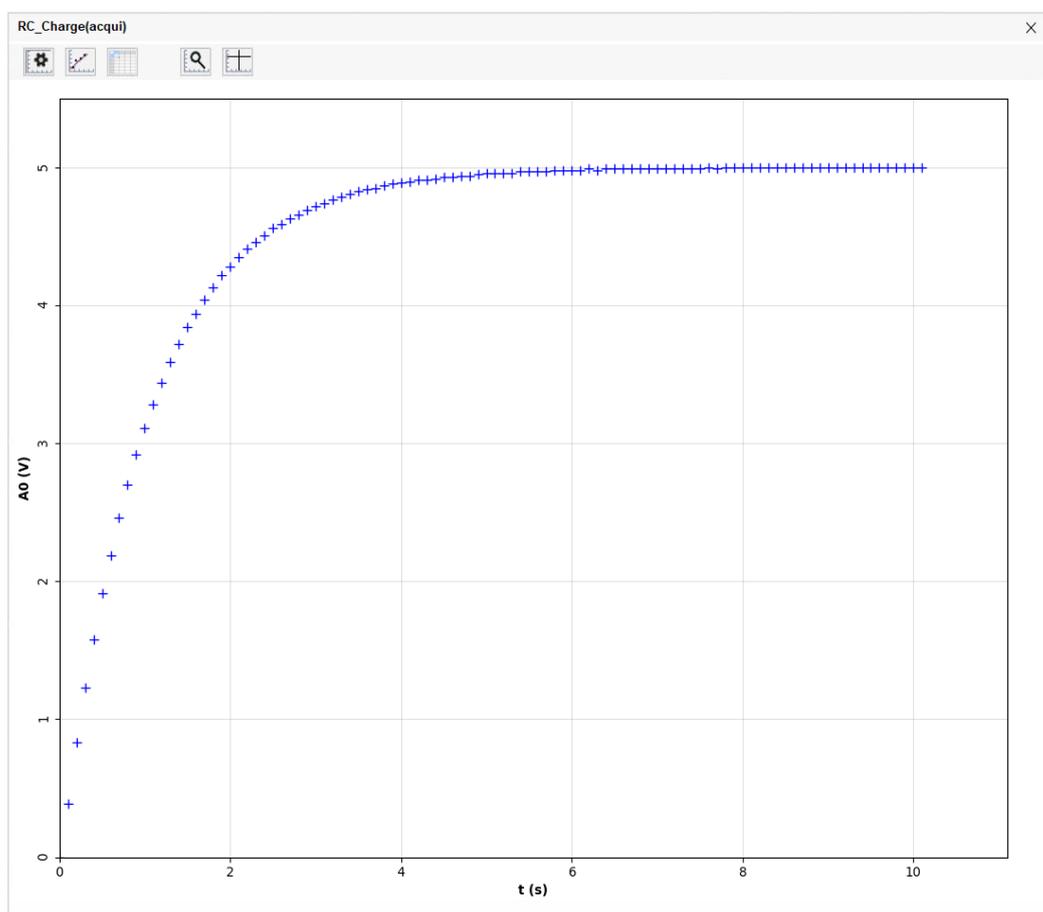


A partir de l'onglet « Tableur », les données acquises peuvent être enregistrées dans un fichier .CSV ou exploitées à l'aide du tableur d'ArdPyLog :

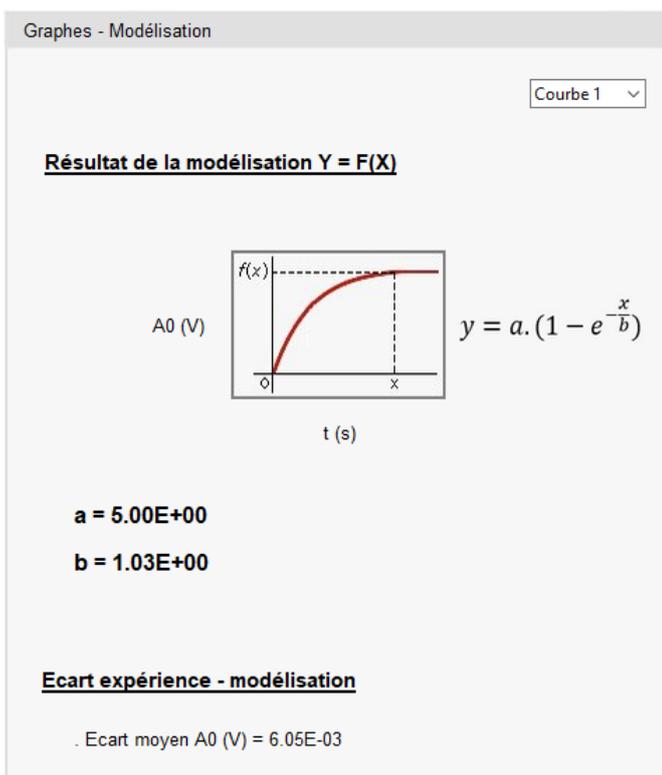
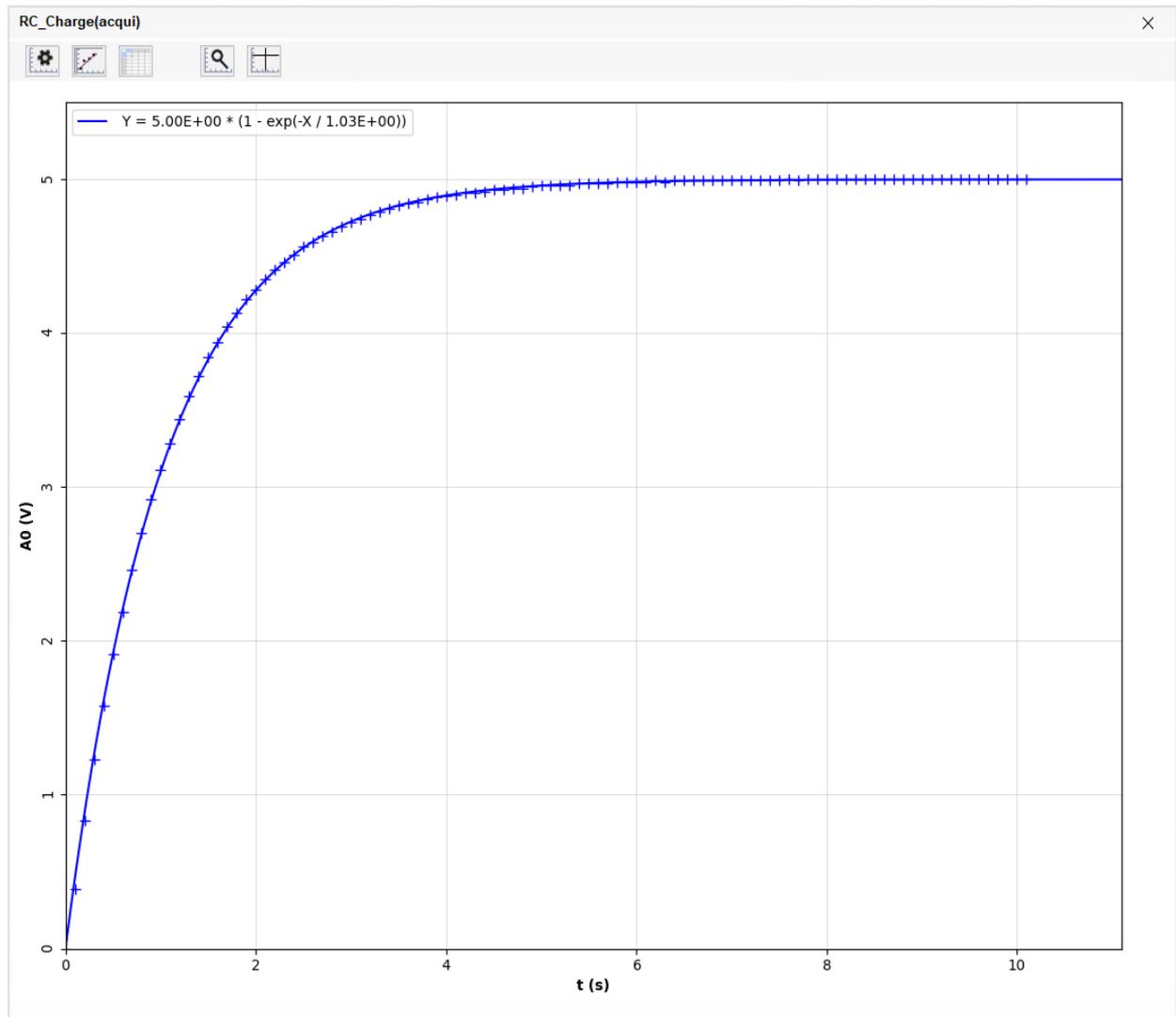
RC_Charge(acqui)

103

	t (s)	A0 (V)	C
1	0,0	0,0	
2	0,1	0,39	
3	0,2	0,83	
4	0,3	1,23	
5	0,4	1,58	
6	0,5	1,91	
7	0,6	2,19	
8	0,7	2,46	
9	0,8	2,7	
10	0,9	2,92	
11	1,0	3,11	
12	1,1	3,28	
13	1,2	3,44	
14	1,3	3,59	
15	1,4	3,72	
16	1,5	3,84	
17	1,6	3,94	
18	1,7	4,04	
19	1,8	4,13	
20	1,9	4,22	
21	2,0	4,28	
22	2,1	4,35	
23	2,2	4,41	
24	2,3	4,46	
25	2,4	4,51	
26	2,5	4,56	
27	2,6	4,59	
28	2,7	4,63	
29	2,8	4,66	
30	2,9	4,69	
31	3,0	4,72	
32	3,1	4,74	
33	3,2	4,77	



La modélisation des mesures selon la fonction théorique de charge du condensateur d'un dipôle RC, $U_c=f(t)$, donne :



Par la modélisation, la détermination de τ ($\tau=1,03$ s) donne une valeur très proche de la valeur théorique ($\tau = 1,0$ s).

6.4.2 Décharge du condensateur

Afin de suivre l'évolution temporelle de la tension aux bornes du condensateur pendant sa décharge, on s'assure dans un premier temps qu'il est complètement chargé en fermant K1 et en ouvrant K2.

Il faut ensuite définir les paramètres de l'acquisition :

- Entrées analogiques : **A0**

- Durée totale d'acquisition : **10 s**

En effet, la valeur théorique de τ (temps pour lequel le condensateur est déchargé à 37 %) du dipôle RC est :

$$\tau = RC = 10 \cdot 10^3 \times 100 \cdot 10^{-6} = 1 \text{ S}$$

Avec une durée totale d'acquisition de 10 s, la décharge sera complète.

- Période d'échantillonnage : **100 ms**

- Déclenchement :

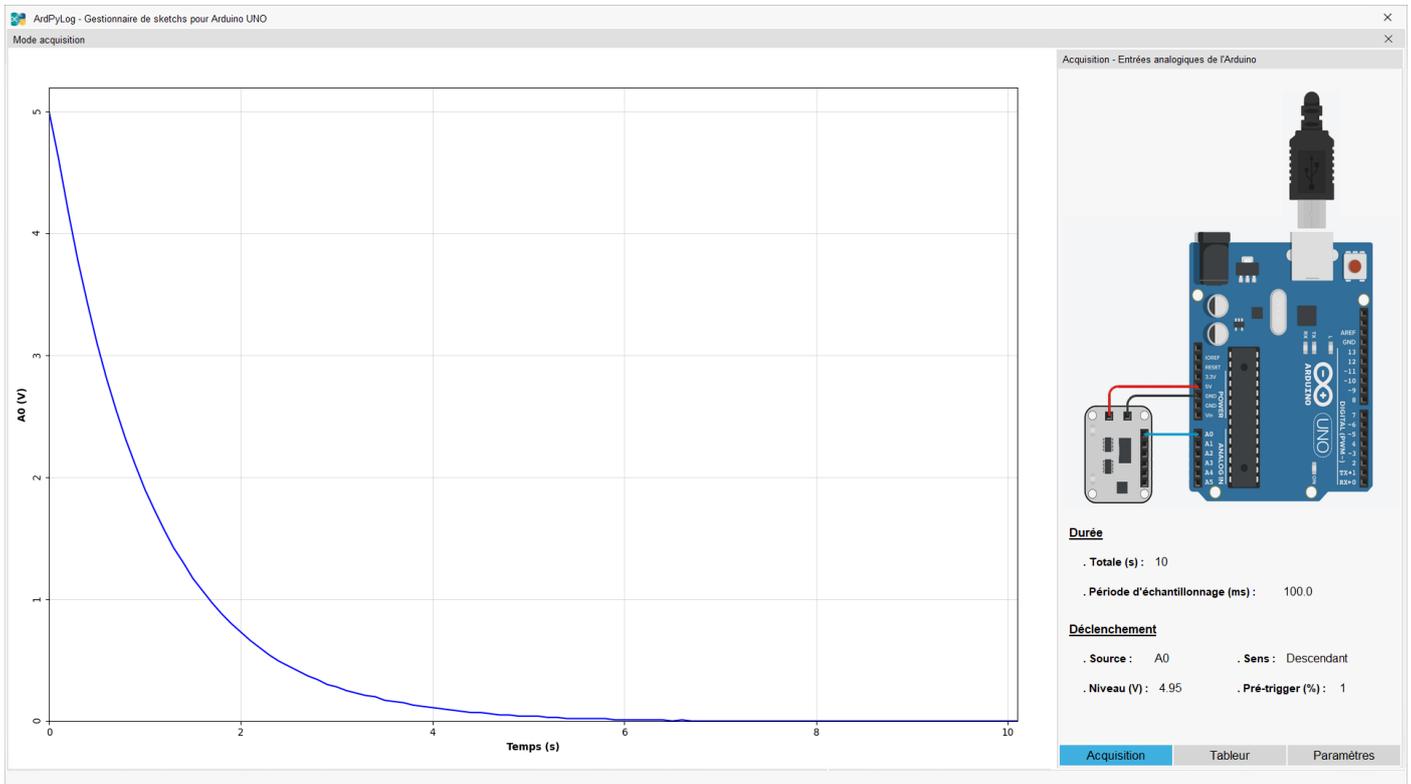
Afin d'acquérir uniquement la tension aux bornes du condensateur au cours de sa décharge, l'acquisition sera déclenchée quand la tension mesurée sur A0 sera inférieure à 4,95 V avec un pré-trigger de 1% (1% de 10 s, soit 100 ms) afin d'observer le début de la décharge.

Les paramètres de déclenchement seront donc :

<u>Déclenchement</u>			
. Source :	A0	. Sens :	Descendant
. Niveau (V) :	4.95	. Pré-trigger (%) :	1

Après avoir validé les paramètres (bouton « **Appliquer** »), l'acquisition est démarrée (clic sur la prise USB).

K1 est alors ouvert et K2 fermé. Quand la tension U_c , aux bornes du condensateur, est inférieur à 4,95 V, l'évolution temporelle de U_c est tracée.

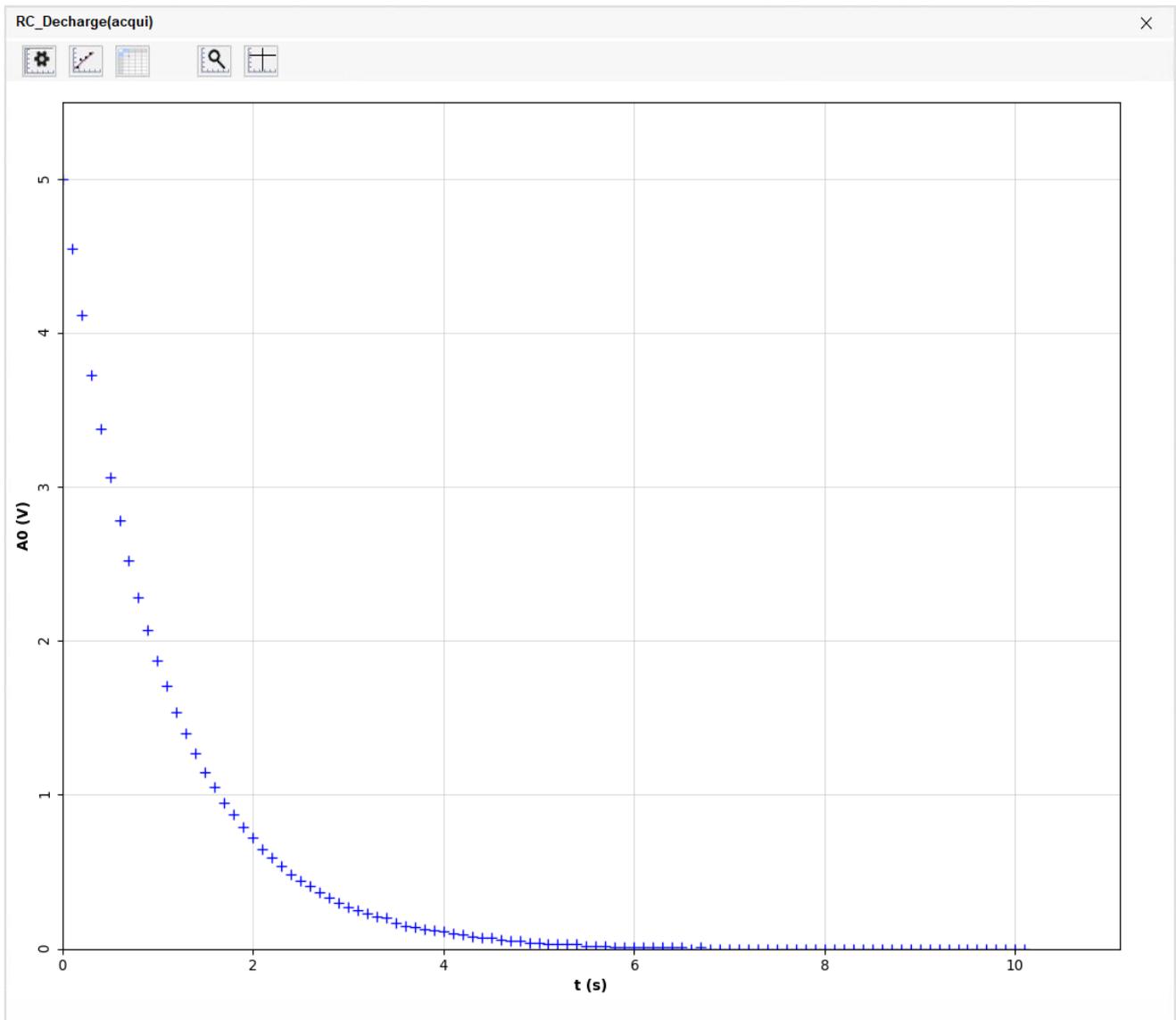


A partir de l'onglet « Tableur », les données acquises peuvent être enregistrées dans un fichier .CSV ou exploitées à l'aide du tableur d'ArdPyLog :

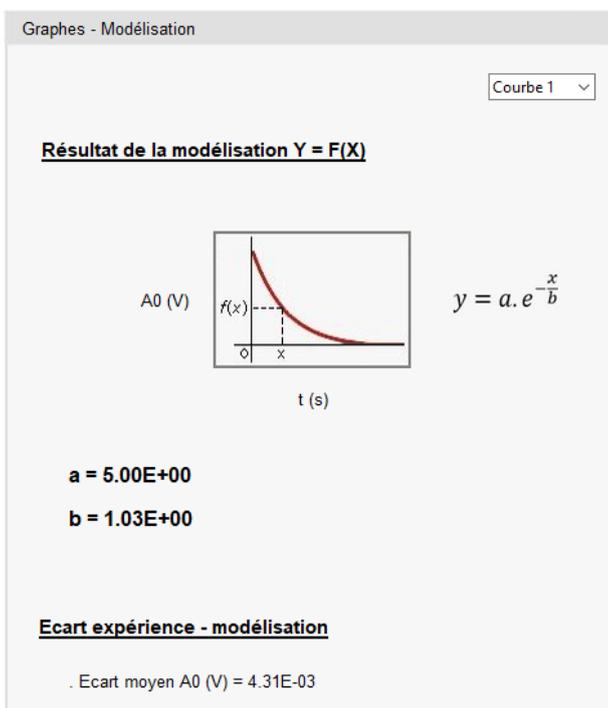
RC_Decharge(acquil)

103

	t (s)	A0 (V)	C
1	0,0	5,0	
2	0,1	4,55	
3	0,2	4,12	
4	0,3	3,73	
5	0,4	3,38	
6	0,5	3,06	
7	0,6	2,78	
8	0,7	2,52	
9	0,8	2,28	
10	0,9	2,07	
11	1,0	1,87	
12	1,1	1,71	
13	1,2	1,54	
14	1,3	1,4	
15	1,4	1,27	
16	1,5	1,15	
17	1,6	1,05	
18	1,7	0,95	
19	1,8	0,87	
20	1,9	0,79	
21	2,0	0,72	
22	2,1	0,65	
23	2,2	0,59	
24	2,3	0,54	
25	2,4	0,48	
26	2,5	0,44	
27	2,6	0,41	
28	2,7	0,37	
29	2,8	0,33	
30	2,9	0,3	
31	3,0	0,27	
32	3,1	0,25	
33	3,2	0,23	



La modélisation des mesures selon la fonction théorique de décharge du condensateur d'un dipôle RC, $U_c=f(t)$, donne :



Par la modélisation, la détermination de τ ($\tau=1,03$ s) donne une valeur très proche de la valeur théorique ($\tau = 1,0$ s).

