Notice d'utilisation
« Application pour le Réglage du Lecteur MP3 »

Version 1

Le projet décrit la prise en main du logiciel pour paramétrer le Lecteur MP3 avec l'application dédiée développée avec le logiciel « Processing ».

L'application est compatible sous Windows, Linux, Mac, Raspberry.

Aucune connaissance en programmation n'est nécessaire pour l'installation et le paramétrage de l'application.

Ne pas utiliser le moniteur série du logiciel « Arduino » lors de l'utilisation de cette application.

La connexion avec la carte Arduino se fait via le port USB et donc une communication pour le port série (Port COM).

Les données sont enregistrées dans la mémoire Eeprom de la carte Arduino : les données sont conservées après une mise hors tension de la carte.

L'application permet de régler le Lecteur MP3 en mode analogique et/ou digital.

Pour faciliter les différents réglages, une application spécifique a été réalisée à cet effet.

L'application permet également de sauvegarder les paramètres dans un fichier PDF.

---

1. Prérequis ____ **2**

2. Compiler l'application ____ **2**

a. Télécharger « Processing » *2*
b. Installer une librairie *2*
c. Télécharger l'application de réglages *3*
d. Compiler l'application *4*

3. « Reglages_decodeur_sonore » ____ **5**

a. Connexion à la carte Arduino *5*
b. Principes généraux *6*
c. Type de centrale *6*
d. Mode de fonctionnement *6*
e. Adresse DCC de lecture des sons *7*
f. Volume sonore *7*
g. Adresses DCC des sons *7*
h. Gestion en mode aléatoire *7*
i. Jouer des sons *8*
j. Enregistrement de la configuration : PDF *8*

---

1. Prérequis

Il faut avoir au préalable téléverser la version N°2 du programme Arduino de commande du Lecteur MP3 dans une carte Arduino UNO ou NANO.

L'application « Processing » n'est pas obligatoire pour le réglage des servomoteurs. L'application sera nécessaire pour compiler le programme pour disposer d'une application auto exécutable.

« Processing » est un environnement de développement libre (sous licence GNU GPL).

Les programmes réalisés avec Processing en mode Java peuvent être exportés sous forme d'applications indépendantes pour Windows, Linux ou Mac OS X (en réalité n'importe quelle machine disposant d'une machine virtuelle Java).

2. Compiler l'application

Il est apparu difficile de mettre en téléchargement l'application : il semble que des fichiers « java » soient supprimés lors des téléchargements.

Pour cela, il est proposé de recompiler le programme et générer un fichier exécutable en fonction du système d'exploitation souhaité.

a. *Télécharger « Processing »*

En fonction du système d'exploitation utilisé, télécharger l'application « Processing » sur le site : https://processing.org/download/

L'application « Processing » est gratuite.

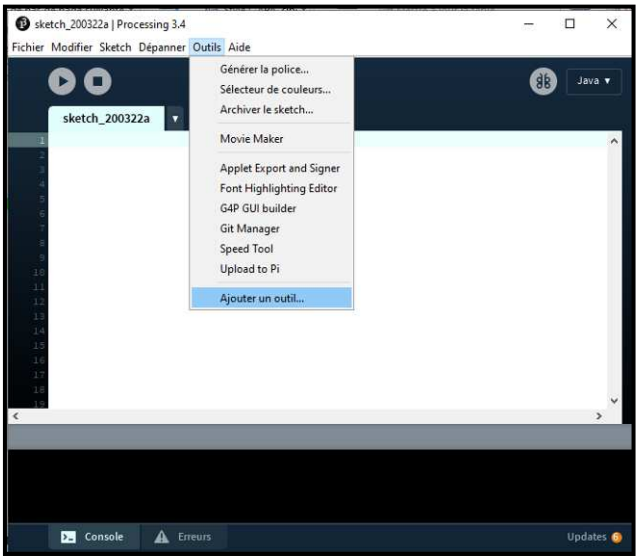
Installer l'application « **Processing 3.x** ».

b. *Installer une librairie*

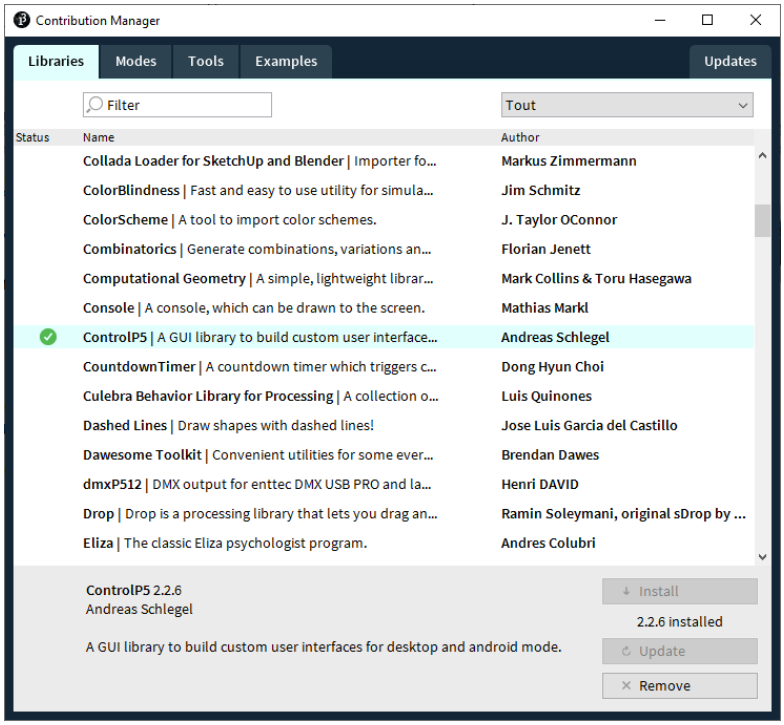
Pour utiliser l'application développée par l'UAICF-Nevers-Vauzelles, il faudra installer une librairie qui n'est pas installée par défaut.

Pour cela, lancer le programme « **processing.exe** ».

Pour installer une librairie, dans la barre de menus cliquer sur « **Outils** » puis sur « **Ajouter un outil…** »



Sélectionner « ControlP5 » et cliquer sur le bouton « **Installer** ».



Quand l'installation est terminée, fermer la fenêtre en cliquant sur la « **Croix** » en haut à droite.

c. ***Télécharger l'application de réglages***

Télécharger les fichiers « **Sources** » sur le site de l'UAICF Nevers-Vauzelles : http://modelisme58.free.fr

Après avoir téléchargé le fichier « **Reglages_Decodeur_Sonore-Sources.zip**», il faut le décompresser avec un logiciel gratuit dédié comme 7zip, IRZAC,…

Le répertoire où sont décompressés les fichiers doit être nommé « **Reglages_decodeur_sonore** ».

Le répertoire doit comprendre plusieurs fichiers « PDE » ainsi que 2 répertoires :

- « data » qui contient des fichiers images et polices de caractères ;
- « PDF » où seront enregistrés les réglages de la carte Arduino.

Depuis la barre de menus, cliquer sur « **Fichier** » puis sur « **Ouvrir…** ».

Rechercher le fichier « **Reglages_decodeur_sonore.pde** » dans le répertoire portant le même nom « **Reglages_decodeur_sonore** » puis cliquer sur « **Ouvrir** ».

Pour vérifier que tous les fichiers sont correctement installés, lancer l'application en cliquant sur le bouton « **Exécuter** » ou depuis la barre de menus sur « **Sketch** » puis « **Exécuter** ».

Vue si application fonctionnelle, sans carte Arduino connectée

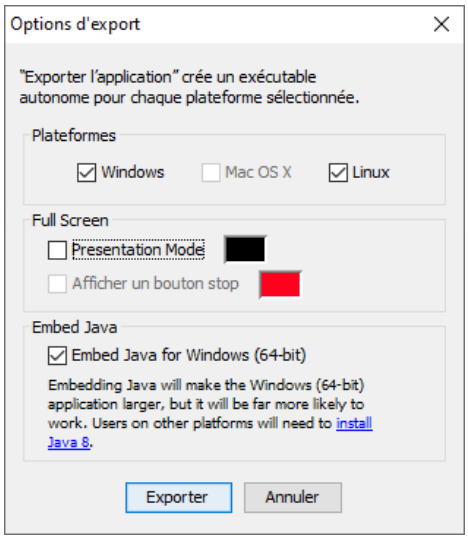
Pour quitter l'application, cliquer sur l'icône « **Carré** » ou sur le bouton « **Arrêter** » dans le logiciel « Processing ».

d. ***Compiler l'application***

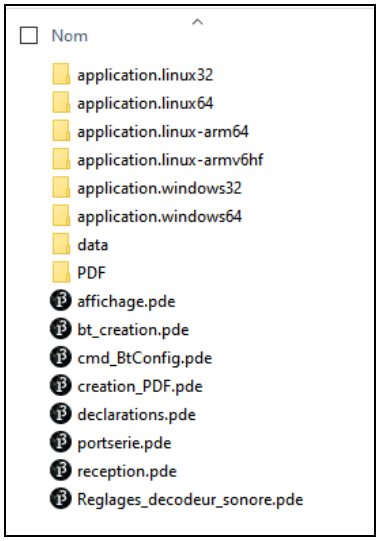
Pour disposer d'une application autonome sans avoir à lancer le logiciel « **Processing** », on réalise une compilation du programme.

Dans la barre de menus « **Fichier** », cliquer sur « **Exporter…** ».

Sélectionner le système d'exploitation et cliquer sur le bouton « **Exporter** ».

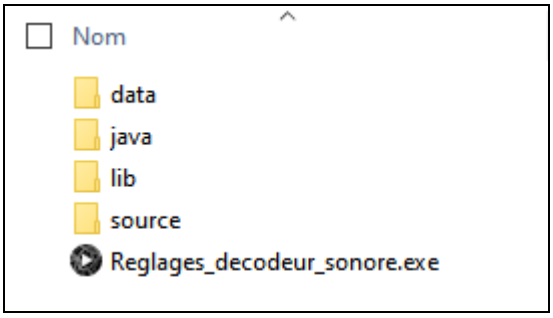


Export



Résultat de l'exportation

A la fin de la compilation, sont créés des répertoires par système d'exploitation.



Contenu d'un répertoire créé

Cliquer sur le fichier « **Reglages_decodeur_sonore** » pour lancer l'application.

3. « Reglages_decodeur_sonore »

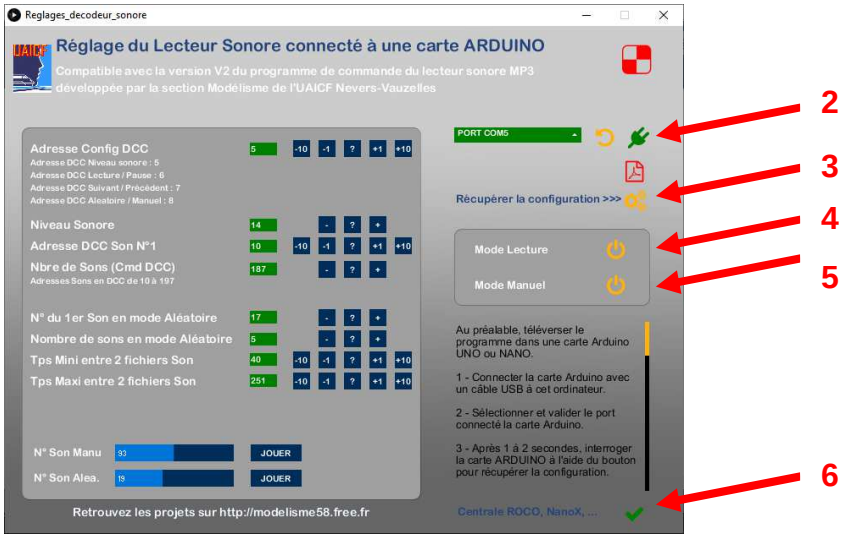
a. *Connexion à la carte Arduino*

1 – Lancer l'application « **Reglages_decodeur_sonore** ».

2 – **Relier une seule carte ARDUINO** avec le programme préalablement téléversé.

3 – Cliquer sur l'icône avec la flèche pour **lister les ports COM** disponibles.

Lister les ports COM disponibles



Affichage suite à la connexion de la carte Arduino

Si plusieurs ports COM sont disponibles, sélectionner le port COM où est reliée la carte Arduino.

L'icône avec la prise verte permet de connecter le port COM sélectionné.

4 – Cliquer sur l'icône orange (**cf. 3**) avec la roue dentée pour **récupérer la configuration** de la carte Arduino : les données sont enregistrées dans la mémoire Eeprom de la carte.

S'affiche alors les paramètres enregistrés dans la mémoire Eeprom de la carte Arduino.

b. Principes généraux

Pour utiliser les fonctions, il faut cliquer sur les icônes ou boutons.

Quand les chiffres sont en « **Vert** », les données sont enregistrées dans la mémoire de la carte Arduino.

Quand les chiffres sont en « **Rouge** », les données ont été modifiées mais pas encore envoyée à la carte Arduino.

Pour modifier une valeur, cliquer sur les boutons « **-** » ou « **+** ».

Le bouton « **?** » permet d'afficher le paramètre enregistré dans la mémoire Eeprom de la carte Arduino.

Pour enregistrer une valeur, il faut cliquer sur le bouton « **Envoyer les modifications** ».

c. Type de centrale

En mode analogique, cette fonction n'est pas nécessaire.

Comme tous les fabricants n'ont pas utilisé le même plan d'adressage des accessoires en DCC, il y a un décalage de 4 adresses entre les centrales DCC Lenz et Roco.

Le décodeur de servomoteurs a prévu cette modification pour éviter de se poser la question « est-ce que je dois ou pas décaler mes adresses de 4 unités ? »

Cliquer sur le symbole vert « ✓ » pour sélectionner la centrale DCC utilisée (**cf. 6**).

d. Mode de fonctionnement

L'icône orange (**cf. 4**) permet de sélectionner le mode de lecture des sons :

- Lecture des sons ;
- Pause.

L'icône orange (**cf. 5**) permet de sélectionner le mode de lecture des sons :

- Lecture des sons en mode Aléatoire ;
- Lecture en mode Manuel : commande par les boutons ou en DCC.

e. Adresse DCC de lecture des sons

On définit une adresse DCC au lecteur MP3. Il sera automatiquement attribué 3 autres adresses pour commander le lecteur.

- Adresse « N » : pour augmenter ou diminuer le niveau du volume sonore ;
- Adresse « N + 1 » : mode « Lecture » ou « Pause » ;
- Adresse « N + 2 » : lecture du fichier « Suivant » ou « Précédent » ;
- Adresse « N + 3 » : lecture des fichiers en mode « Manuel » ou « Aléatoire ».

En appuyant sur les touches « Gauche » ou « Droite » de la commande des accessoires en DCC, on commande les fonctions du lecteur Mp3.

f. Volume sonore

Pour modifier le volume sonore, cliquer sur les boutons « **-** » ou « **+** ».

La valeur est comprise entre 0 et 30 ;

Le bouton « **?** » permet d'afficher la valeur actuellement enregistrée dans la mémoire Eeprom de la carte Arduino.

g. Adresses DCC des sons

En DCC, on attribue l'adresse du premier fichier son commandé.

On indique ensuite le nombre de fichier sons commandés.

L'interface indique automatiquement les plages d'adresses attribuées pour jouer les sons.

Si on attribue l'adresse DCC « 10 », en commandant l'accessoire ayant l'adresse « 10 », on jouera le son « 001.mp3 » présent sur la carte SD.

Avec l'adresse DCC « 11 », on jouera le son « 002.mp3 »
Avec l'adresse DCC « 12 », on jouera le son « 003.mp3 »
…

h. Gestion en mode aléatoire

Le mode aléatoire permet de :

- Jouer des sons aléatoire dans une plage définie par l'utilisateur ;
- Mettre une temporisation aléatoire entre la lecture de 2 fichiers sons qui est définie par l'utilisateur.

L'interface détermine le numéro du premier fichier lu en mode aléatoire.

Si on attribue la valeur « 17 », le premier son en mode aléatoire sera le fichier « 017.mp3 »

On indique le nombre de fichiers en mode aléatoire et il est calculé automatiquement la liste des fichiers pouvant être lu en mode aléatoire.

On paramètre ensuite le temps minimal et maximal en secondes pour la lecture entre 2 fichiers sons en mode aléatoire.

i. Jouer des sons

Les 2 boutons « Jouer » permettent de jour des sons en fonction des paramètres définis préalablement.

j. Enregistrement de la configuration : PDF

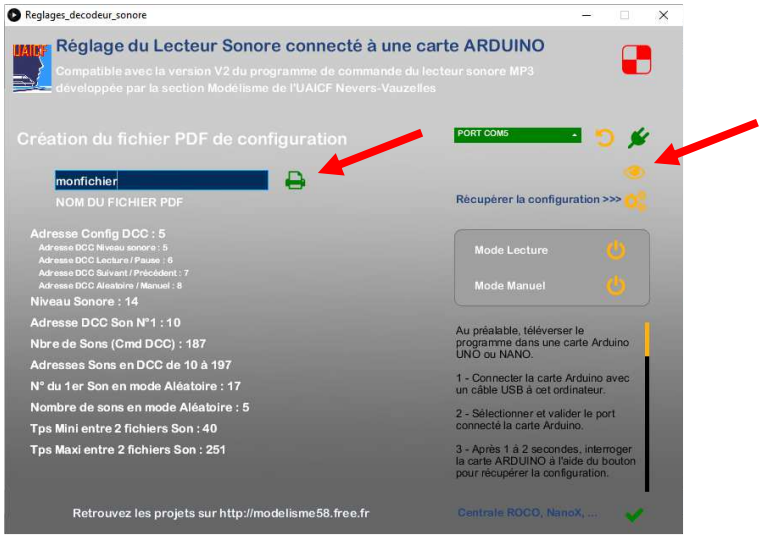
Il est possible de générer un fichier « PDF » où sera enregistrée la configuration avec les valeurs renseignées dans la mémoire de l'Eeprom de la carte Arduino.

1 – **Cliquer sur l'icône « PDF ».**

2 – **Renseigner** un **nom.**

3 – Pour générer le fichier, **cliquer** sur l'icône « **Imprimante** ».
Un fichier « PDF » est généré automatiquement avec la date, l'heure et le nom indiqué.
Le fichier est disponible dans le répertoire « **PDF** » de l'application utilisée.

4 – **Quitter l'interface PDF** en cliquant sur l'icône « **Œil** ».



Enregistrement des réglages dans un fichier « PDF »