

**Ce Document présente la
procédure à suivre pour :**

- **Préparer
l'environnement de
développement**
- **Transférer
l'application vers
une autre machine**

TESTEUR ELECTRIQUE MULTICABLES

Guide d'installation

Imane Azzimani

Sommaire:

Introduction	3
PARTIE - I	3
I. IMPLEMENTATION DU PROGRAMME SUR ARDUINO :	3
II. PREPARATION DE L'ENVIRONNEMENT PYTHON :	4
1. Installation de Python (Anaconda distribution) :	4
2. Exécution de l'application :	5
III. GENERATION DE L'EXECUTABLE :	7
PARTIE - II	8
1. Installation du driver Arduino :	8
2. Importation de l'application :	10
3. Prototype	11
Liens utiles	13

INTRODUCTION :

Il s'agit d'un mini-testeur électrique qui permette de tester plusieurs fils électriques simultanément (continuité/inversion, court-circuit), ce qui va optimiser l'opération du test manuel en minimisant le temps consommé ainsi que la marge d'erreur.

Dans la première partie, on va expliquer comment préparer l'environnement du développement , implémenter le programme sur Arduino, et comment générer l'exécutable. (PC 1)

La deuxième partie est consacrée à l'exportation de l'application vers le PC destinataire (PC 2).

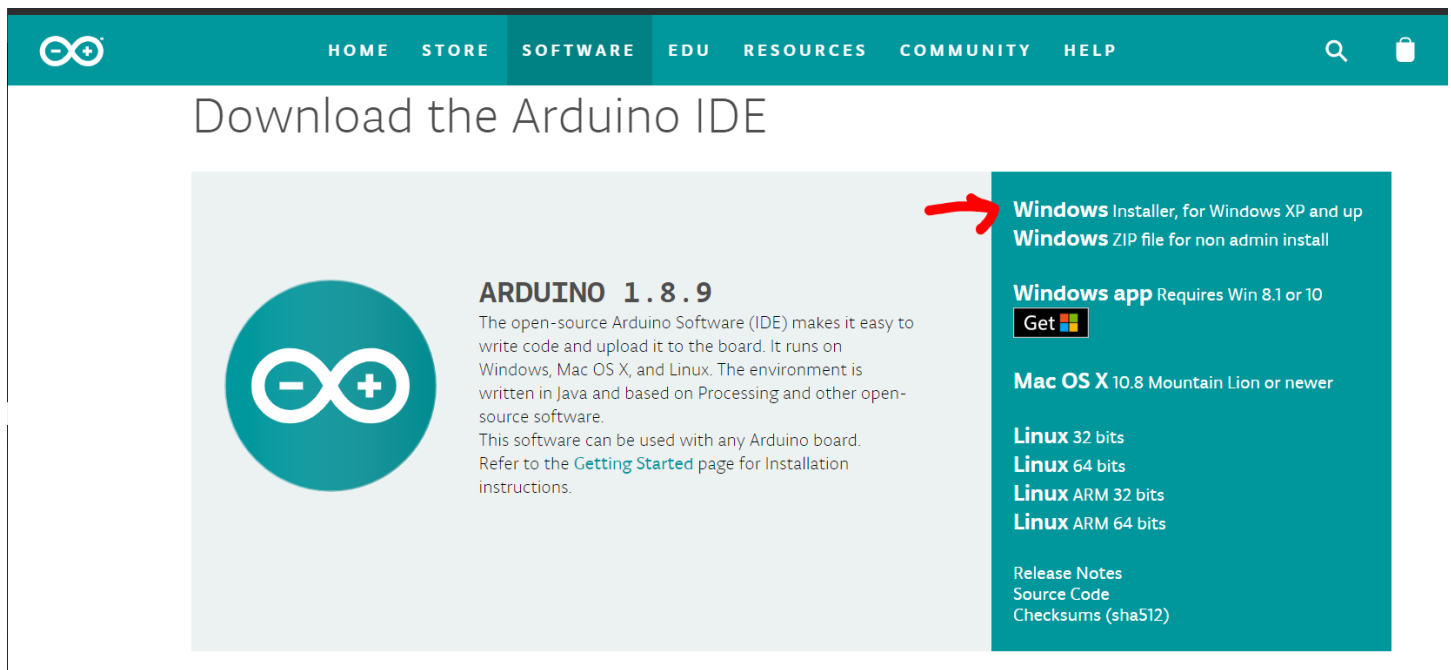
PARTIE -I


- *IMPLEMENTATION DU PROGRAMME SUR ARDUINO*
 - *INSTALLATION DE PYTHON*
 - *GENERATION DE L'EXECUTABLE*
-

Avant de commencer, on décompresse le dossier `TESTER_APP_GUIDE.rar` dans le disque D

I. IMPLEMENTATION DU PROGRAMME SUR ARDUINO :

- On télécharge l'environnement Arduino : <https://www.arduino.cc/en/Main/Software>



- On raccorde la carte au PC en utilisant un câble USB, puis, on installe les pilotes du périphérique Série-USB (Voir PARTIE – II)
- On lance le logiciel.
- Si tout va bien, On ouvre le dossier **testerApp.ino**, (double clic sur le fichier)
- On transfère le programme vers la carte en cliquant sur 

II. PREPARATION DE L'ENVIRONNEMENT PYTHON :

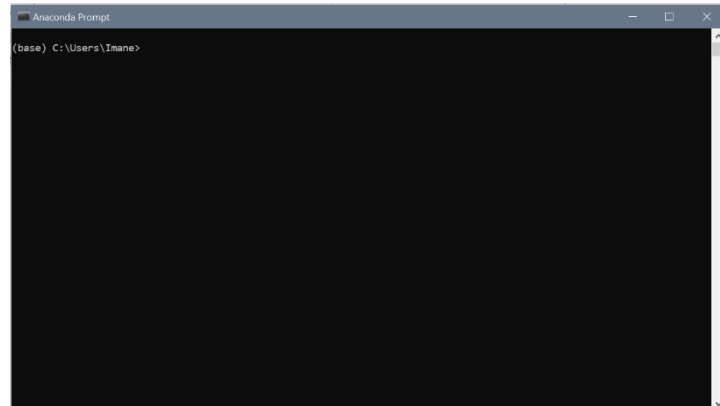
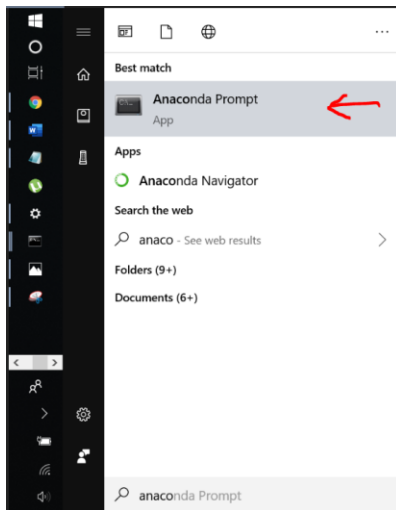
Il s'agit d'une interface graphique depuis laquelle on lance le programme Arduino.

Comme on utilise le langage python3.7 pour développer cette interface, on doit alors commencer par la préparation de l'environnement python.

Deux manières d'installation sont possibles : l'installation classique et l'installation de la distribution Anaconda.. Cette dernière est la plus facile car elle facilite la gestion des packages.

1. INSTALLATION DE PYTHON (ANACONDA DISTRIBUTION) :

- Installer Anaconda 3 :
Voir tutoriel <https://problemsolvingwithpython.com/01-Orientation/01.03-Installing-Anaconda-on-Windows/>
- Ouvrir anaconda prompt :



- Créer un environnement virtuel en exécutant la commande :
`conda create --name envname python=3`
- Activer l'environnement crée:
`activate envname`
- Installer le package pySerial:
`conda install pyserial`

2. EXECUTION DE L'APPLICATION :

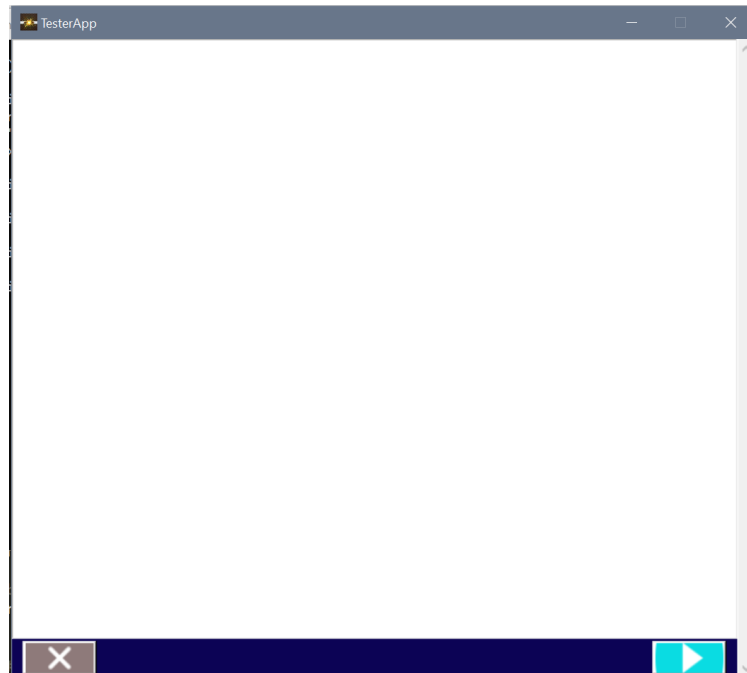
- Après avoir réussi la préparation de l'environnement python, on ouvre le fichier **appGUI.py** avec n'importe quel IDE (ex : Sublime TEXT). On peut même l'ouvrir dans un bloc note.
- On connecte Arduino avec PC.
- On récupère le nom du port Arduino 'COMx' (depuis device manager avec x un nombre quelconque)
- On remplace le nom du port par celui qui correspond à Arduino 'COMx'(il se diffère d'un PC a un autre. Dans notre cas, x = 5)

```
6 serialPort = "COM5"
```

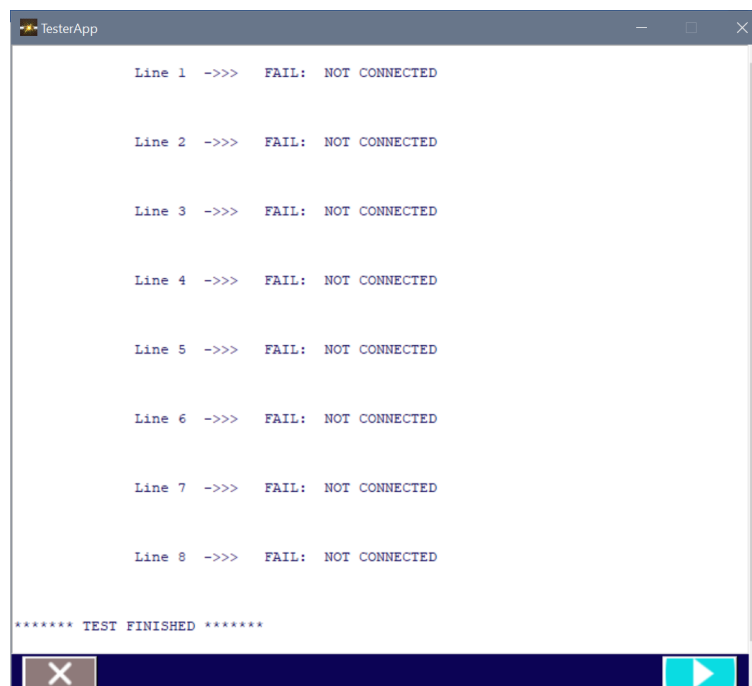
- On ouvre anaconda prompt
- On active l'environnement crée (`activate envname`)
- On se déplace vers la location du fichier **appGUI.py** :

```
D :  
cd D:\TESTER_APP_GUIDE
```

- On exécute le programme en tapant :
`python appGUI.py`
- Une fenêtre s'ouvre :



- Pour tester l'app, on clique sur le bouton 'Play', (double clic) (on rappelle que Arduino doit être connecté avec PC lors de l'exécution)
- Pour refaire le test on refait le branchement, on clique sur **CLEAR**, puis sur **PLAY**



L'arrivée à cette étape signifie qu'on peut passer à l'étape de génération de l'exécutable, cette phase consiste à convertir le code python au code machine pour qu'on puisse le transférer à une autre machine (le type de OS de la machine destinataire doit être Windows)

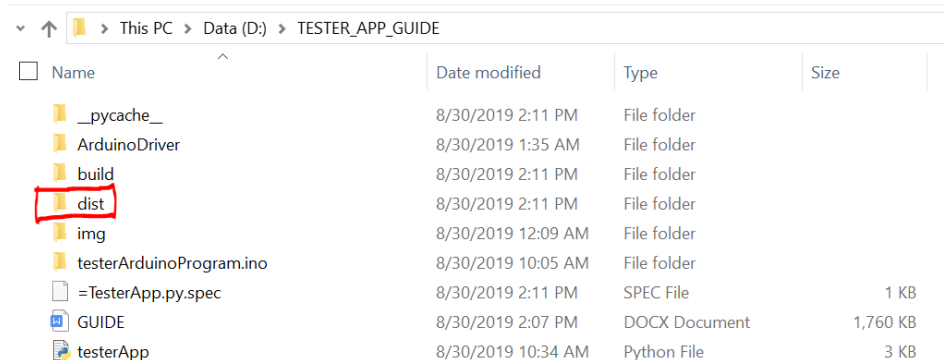
PS : il faut mentionner que plusieurs problèmes peuvent s'apparaitre durant l'installation de python et ses librairies, (genre problèmes de compatibilité, emplacement de certains fichiers, ...) c'est la partie la plus ennuyeuse,

il faut juste googler ses erreurs (ctrl+C/ ctrl+V → google) pour trouver des solutions, heureusement, python a une large/dynamique communauté !

III. GENERATION DE L'EXECUTABLE :

On commence par l'installation de **PyInstaller**

- On ouvre **Anaconda prompt**
- On exécute la commande :
`conda install pyinstaller`
- On vérifie l'installation par :
`pyinstaller --version`
- On se déplace vers l'emplacement du fichier appGUI.py
- D :
- `cd D:\TESTER_APP_GUIDE`
- On exécute la commande :
`pyinstaller testerApp.py --name==TesterApp.py -w -i img\icon.ico -hidden-import=PySerial`
- On remarque l'auto génération de quelques répertoires, ce qui nous intéresse c'est le dossier **dist**



Name	Date modified	Type	Size
__pycache__	8/30/2019 2:11 PM	File folder	
ArduinoDriver	8/30/2019 1:35 AM	File folder	
build	8/30/2019 2:11 PM	File folder	
dist	8/30/2019 2:11 PM	File folder	
img	8/30/2019 12:09 AM	File folder	
testerArduinoProgram.ino	8/30/2019 10:05 AM	File folder	
=TesterApp.py.spec	8/30/2019 2:11 PM	SPEC File	1 KB
GUIDE	8/30/2019 2:07 PM	DOCX Document	1,760 KB
testerApp	8/30/2019 10:34 AM	Python File	3 KB

- On déplace le dossier **img** vers **D:\TESTER_APP_GUIDE\dist\=TesterApp.py**.
- On renomme **dist** ... soit « **TesterAppFolder** »
- On décompresse **TesterAppFolder**
- On le transfère vers n'importe quelle machine (OS = Windows)
(Voir PARTIE II.2)

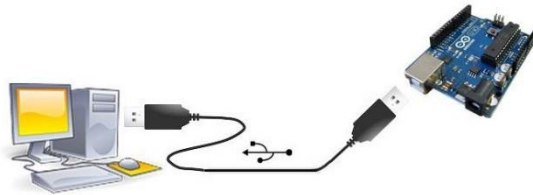
PARTIE - II


▪ L'EXPORTATION DE L'APPLICATION AU PC DESTINATAIRE :

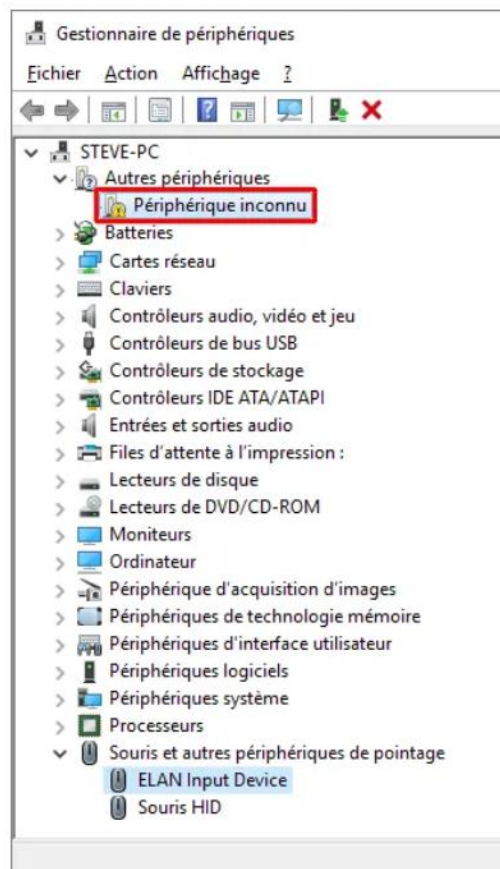
Après avoir généré l'exécutable, on le transfère vers l'ordinateur destiné au test électrique :

1. INSTALLATION DU DRIVER ARDUINO :

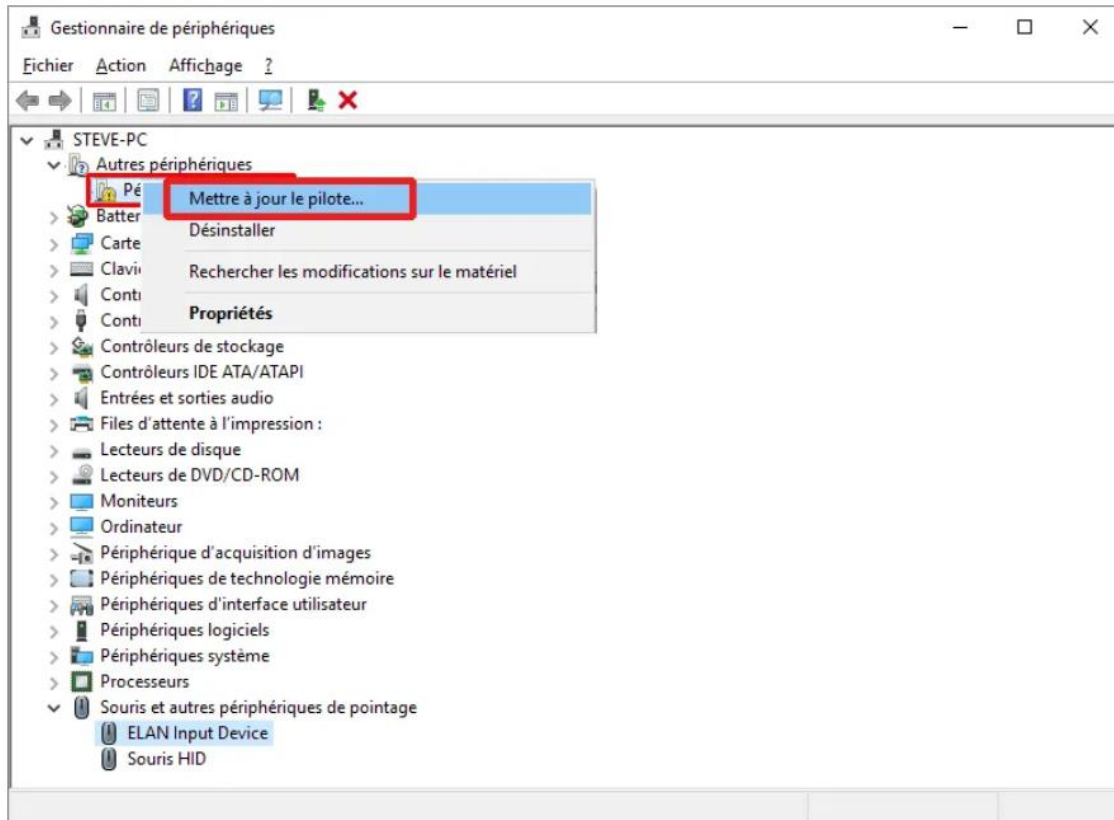
- On connecte Arduino avec le PC via le port USB



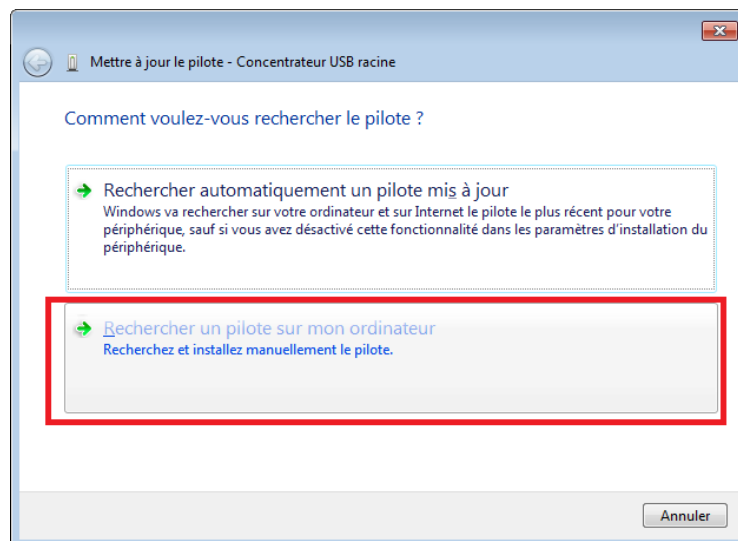
- On ouvre le gestionnaire de périphériques en faisant un clic droit sur 
- Dans le gestionnaire de périphériques, on remarque l'apparition de « Périphérique inconnu » suivi du nom du port (COM x)



- On fait un clic droit sur le périphérique non reconnu puis on clique sur **Mettre à jour le pilote...**



- On clique sur « **Recherche un pilote sur mon ordinateur** » :



- On spécifie le chemin vers le pilote d'Arduino : (la location du dossier contenant le pilote)
(D:\TESTER_APP_GUIDE\ArduinoDrivers)
- On clique sur **suivant** -> **fermer**

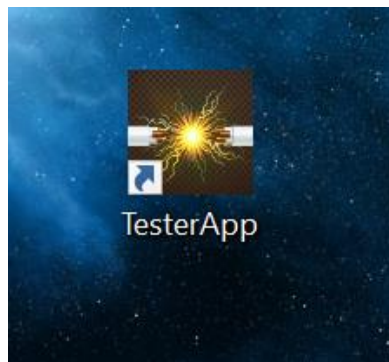
2. IMPORTATION DE L'APPLICATION :

- On Décompresse le dossier TesterApp contenant l'exécutable qu'on a généré dans la première partie.
- On ouvre le dossier ... On lance l'application en faisant double clique sur l'exécutable

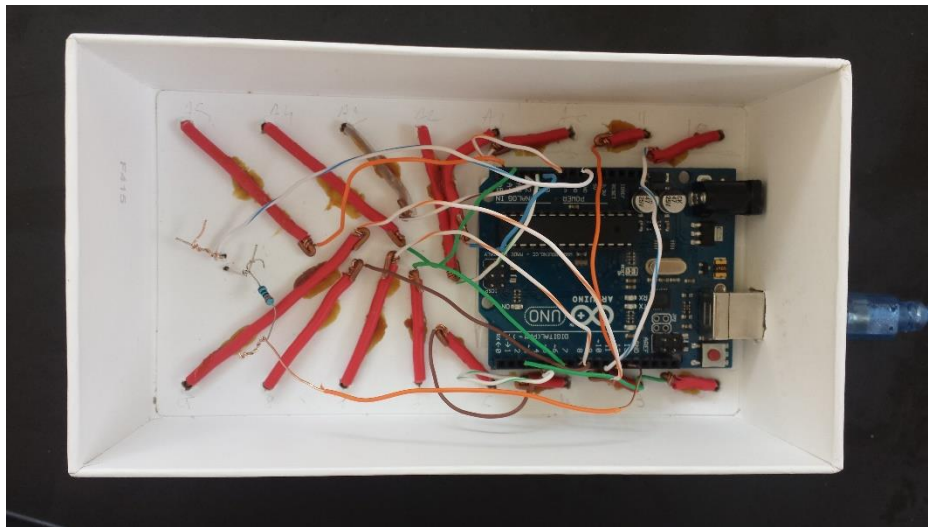
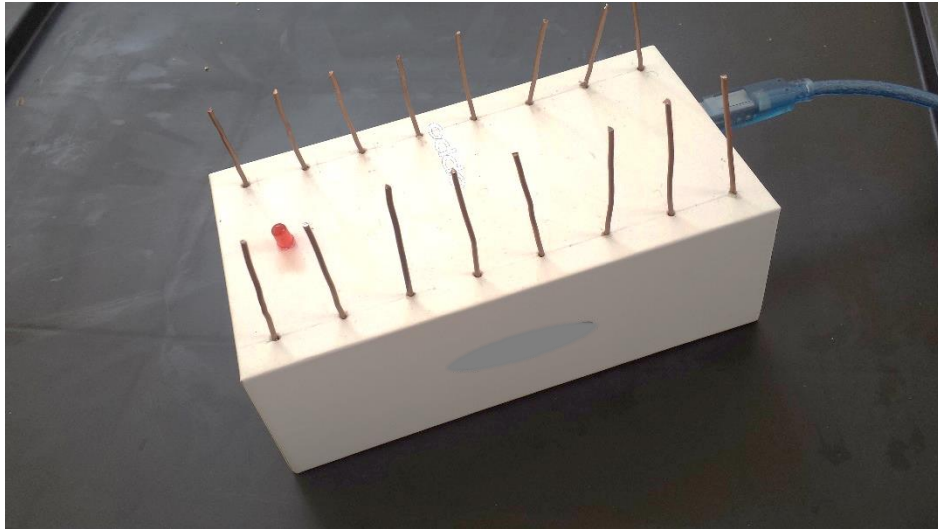
D:\TesterAppFolder\=TesterApp.py

This PC > Data (D:) > TesterAppFolder > =TesterApp.py				
<input type="checkbox"/> Name	Date modified	Type	Size	
img	8/31/2019 2:53 PM	File folder		
tcl	8/30/2019 2:12 PM	File folder		
tk	8/30/2019 2:12 PM	File folder		
_bz2	8/30/2019 12:49 AM	Python Extension Mo...	80 KB	
_ctypes	8/30/2019 12:49 AM	Python Extension Mo...	123 KB	
_hashlib	8/30/2019 12:49 AM	Python Extension Mo...	31 KB	
_lzma	8/30/2019 12:49 AM	Python Extension Mo...	244 KB	
_socket	8/30/2019 12:49 AM	Python Extension Mo...	67 KB	
_ssl	8/30/2019 12:49 AM	Python Extension Mo...	113 KB	
_tkinter	8/30/2019 12:49 AM	Python Extension Mo...	59 KB	
=TesterApp.py	8/30/2019 2:11 PM	Application	1,598 KB	
=TesterApp.py.exe.manifest	8/30/2019 2:11 PM	MANIFEST File	2 KB	
api-ms-win-core-console-l1-1-0.dll	8/13/2019 5:16 PM	Application extension	19 KB	

(On peut créer un shortcut pour faciliter l'accès a l'application)



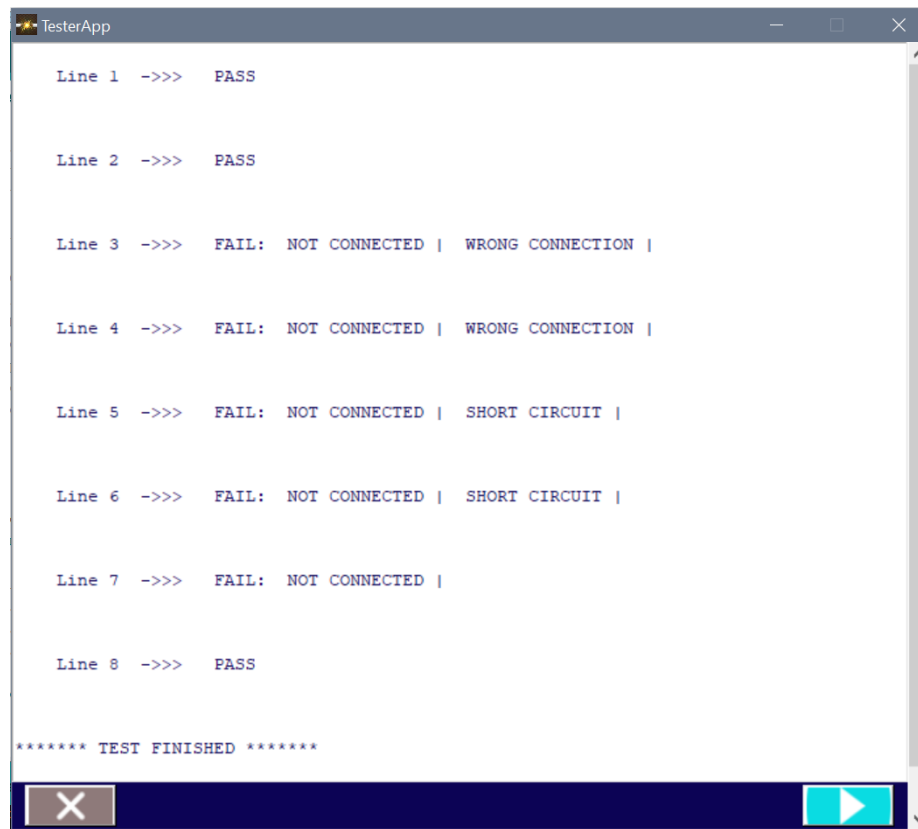
3. PROTOTYPE



- La connexion des câbles avec le boîtier se fait à travers des câbles crocodiles :



- Exemple d'exécution :



```
Line 1 ->>> PASS

Line 2 ->>> PASS

Line 3 ->>> FAIL: NOT CONNECTED | WRONG CONNECTION |

Line 4 ->>> FAIL: NOT CONNECTED | WRONG CONNECTION |

Line 5 ->>> FAIL: NOT CONNECTED | SHORT CIRCUIT |

Line 6 ->>> FAIL: NOT CONNECTED | SHORT CIRCUIT |

Line 7 ->>> FAIL: NOT CONNECTED |

Line 8 ->>> PASS

***** TEST FINISHED *****
```

LIENS UTILES :

- Installation de Anaconda :
<https://problemsolvingwithpython.com/01-Orientation/01.03-Installing-Anaconda-on-Windows/>
- Environnement virtuel sous Anaconda :
<https://riptutorial.com/fr/python/example/10797/creer-des-environnements-virtuels-avec-anaconda>
- Installation de PySerial :
<https://problemsolvingwithpython.com/11-Python-and-External-Hardware/11.01-PySerial/>
- Installation de pyInstaller
<https://anaconda.org/conda-forge/pyinstaller>