**PROJET**

Robot 1 :

**But : à définir clairement.**

Idée de base :déplacement linéaire d’un robot sur une image de code barre(le code barre sera lu à l’aide ‘un capteur IR fixé sur le robot)

🡪lire le capteur code barre, le redessiner sur l’écran ?

🡪associer chaque code barre à un bâtiment par exemple ?

Sur un châssis un module IR perpendiculaire au sens de déplacement

2 servomoteurs à rotation continue (paramètres : min\_pulse=750, max\_pulse=2250)

* **Pour récupérer des données depuis un port série sur le PC nécessité** d’installer la librairie **« pyserial-3.4 »**

|  |  |
| --- | --- |
| **composants** | **Pattes de connexion** |
| 2 servomoteurs à rotation continue | Moteur droit D12  Moteur gauche D13 |
| Capteur InfraRouge gauche | A0 |
|  |  |

**PROJET**

Robot 3 :but à définir ?

2 servomoteurs à rotation continue (paramètres : min\_pulse=750, max\_pulse=2250)

|  |  |
| --- | --- |
| **composants** | **Pattes de connexion** |
| 2 servomoteurs à rotation continue | Moteur droit D12  Moteur gauche D13 |
|  |  |
| HC05 si besoin ? | RX module sur D1 (TX) carte  TX module sur D0 (RX) carte |
| HSR04(module ultrason) si besoin ? |  |
| Composant pour réguler les niveaux ? |  |

En fonction des besoins : sur un châssis un capteur HC05+un module ultrason + composant permettant de gérer les niveaux ???

* **Pour créer un port série sur la carte(si utilisation de bluetooth) :**

[**https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial**](https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial)

**PROJET**

Robot 2 :

**But :**

Idée de base : robot capable de suivre une ligne

🡪amélioration capteur de suivre un chemin de couleur fixé par programmation, ou à l’aide d’un bouton.

🡪amélioration possible choisir la couleur par reception d’info bluetooth

2 servomoteurs à rotation continue (paramètres : min\_pulse=750, max\_pulse=2250)

Robot possédant 2 capteurs IR, et un capteur de couleur à l’avant : TCS3200

Aide pour la librairie pulseio utilisant pulseIn : <https://learn.adafruit.com/ir-sensor/circuitpython>

|  |  |
| --- | --- |
| **composants** | **Pattes de connexion** |
| 2 servomoteurs à rotation continue | Moteur droit D12  Moteur gauche D13 |
| Capteur InfraRouge gauche  Capteur InfraRouge droit | A0  A1 |
| Capteur de couleur TCS3200 | Configuration de la fréquence  S0 D8  S1 D9  Configuration des filtres  S2 D10  S3 D11  Led éclairage  D2  Signal envoyé  OUT D3 |
|  |  |

**PROJET**

Tablette tactile :

**But :**

Idée de base : piloter une tablette tactile de nintendo DS

🡪réussir à tracer des dessins sur le PC ou piloter un personnage ?

Connexion d’une tablette tactile :pour piloter la tablette, il faudra suivre l’algorithme de base permettant de lire les tensions analogiques sur X1A et Y1A qui vous donneront la position d’un point.

Vous avez une documentation technique pour vous expliquer le principe de fonctionnement de la tablette tactile

Pour le capteur FTDI232, nécessité d’installer le driver **« CDM v2.12.28 WHQL Certified »**

* **Pour créer un port série sur la carte :**

[**https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial**](https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial)

* **Pour récupérer des données depuis un port série sur le PC nécessité** d’installer la librairie **« pyserial-3.4 »**

|  |  |
| --- | --- |
| **composants** | **Pattes de connexion** |
| Tablette tactile | X0 D6(fil bleu)  X1N D7(fil rouge)  X1A A2(fil vert)  Y0 D8(fil jaune)  Y1N D9(fil orange)  Y1A A3(fil marron) |
| Module FTDI232(permet de créer une liaison USB/série et communiquer avec le PC dans les 2 sens) | Patte D0 RX  Patte D1 TX |

**PROJET**

Station météo :

**But :**

Idée de base : récupérer des données de capteurs méteo

🡪faire des enregistrements sur fichiers, tracer des graphes

🡪exploiter les données pour prévoir si nécessité d’arroser ?

🡪faire une interface visuelle sous pygame

Capteur I2C TSL2561 ou HTU21D ou BME280 (température, humidité, pression, altitude)

Capteur **BME280** laision I2C bornes SDA, SCL (adresse 0x77)

Sur carte trinket SDA(D0) et SCL(D2)

* Lien pour utilisation capteur I2C bme280 : <https://learn.adafruit.com/circuitpython-on-raspberrypi-linux/i2c-sensors-and-devices>
* **Pour créer un port série sur la carte :**

[**https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial**](https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial)

* **Pour récupérer des données depuis un port série sur le PC nécessité** d’installer la librairie **« pyserial-3.4 »**

|  |  |
| --- | --- |
| **composants** | **Pattes de connexion** |
| Capteur BME280 donnant pression, température, humidité  Liaison I2C | Capteur carte Trinket  SCL D2  SDA D0  GND GND  3.3V 3V0 |
|  |  |
|  |  |

**PROJET**

Jeu simon :

**But :**

Idée de base : réaliser un jeu avec des DELs de couleurs

🡪possibilité de faire une interface sous pygame

4DELs RVB + 4 boutons poussoirs

Pour les DELs RVB, il suffit de mettre à l’état haut ou bas les sorties analogiques

Pour le capteur FTDI232, nécessité d’installer le driver **« CDM v2.12.28 WHQL Certified »**

* **Pour récupérer des données depuis un port série sur le PC nécessité** d’installer la librairie **« pyserial-3.4 »**
* **Pour créer un port série sur la carte :**

[**https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial**](https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial)

|  |  |
| --- | --- |
| **composants** | **Pattes de connexion** |
| DEL1 | Rouge D2  Vert D3  Bleu D4 |
| DEL2 | Rouge D5  Vert D6  Bleu D7 |
| DEL3 | Rouge D8  Vert D9  Bleu D10 |
| DEL4 | Rouge D11  Vert D12  Bleu D13 |
| Module FTDI232(permet de créer une liaison USB/série et communiquer avec le PC dans les 2 sens) | Patte D0 RX  Patte D1 TX |
| Bouton poussoir  (lecture analogique) | A2  A3  A4  A5 |

+ 1 Buzzer à retester (sinon utiliser musique sous pygame), mais plus de patte disponible

**PROJET**

Jeu tourelle :

**But :**

Idée de base : piloter une tourelle avec laser à l’aide d’un joystick qui permet de viser une cible pendant un temps donné. Si la cible est atteinte compter 1 point et remettre le laser(donc la position des servos) dans une position aléatoire

🡪possibilité de faire une interface sous pygame qui affiche les points et permet de suivre la rotation et l’inclinaison de la tourelle.

* Pour les servomoteurs ils peuvent bouger de 0 à 90°

Paramètres pour configurer les MLI :

**min\_pulse=1000, max\_pulse=2000, actuation\_range=90**

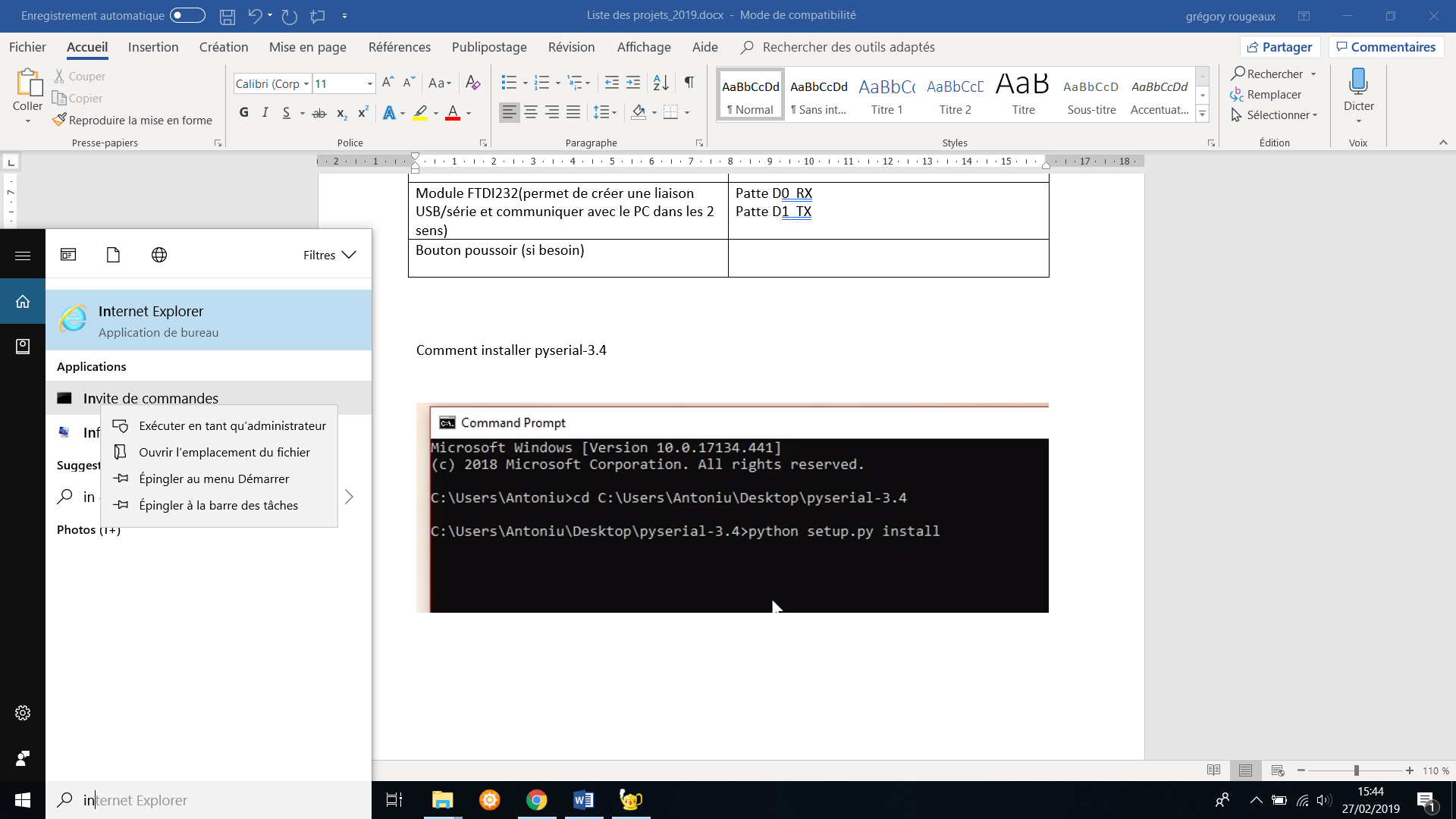
* Pour le capteur FTDI232, nécéssité d’installer le driver **« CDM v2.12.28 WHQL Certified »**
* **Pour récupérer des données depuis un port série sur le PC nécessité** d’installer la librairie **« pyserial-3.4 »**
* **Pour créer un port série sur la carte :**

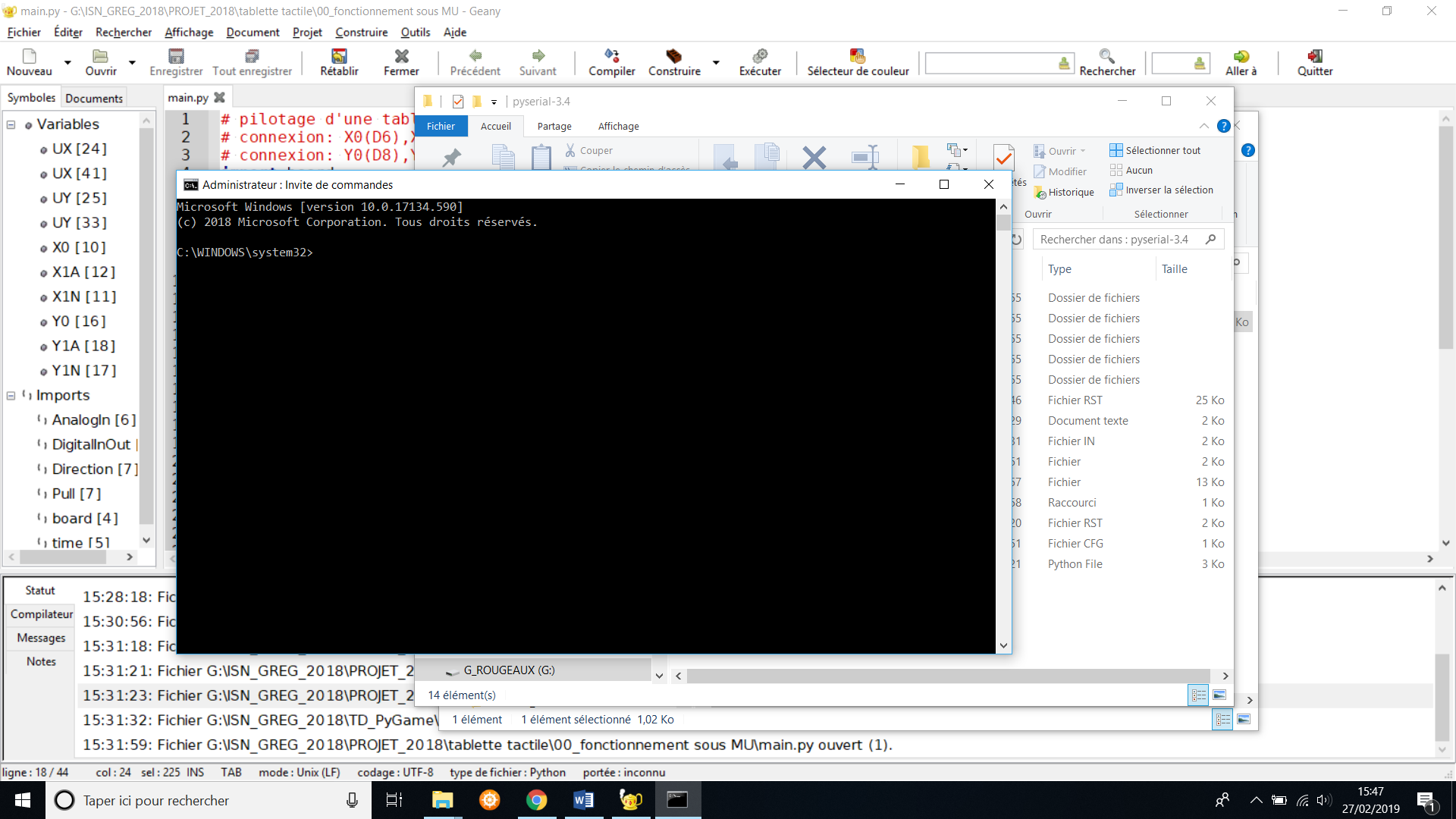
[**https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial**](https://learn.adafruit.com/adafruit-metro-m4-express-featuring-atsamd51/circuitpython-uart-serial)

|  |  |
| --- | --- |
| **composants** | **Pattes de connexion** |
| servoMoteur bas | D2 |
| servoMoteur haut | D3 |
| laser | D7 |
| Joystick(tension entre 0 et 3.3 V) | Y A2  X A3 |
| Capteur cible(photoresistance)  Possibilité de régler la tension reçue avec le potentiometer positionné derrière la cible | A4 |
| Module FTDI232(permet de créer une liaison USB/série et communiquer avec le PC dans les 2 sens) | Patte D0 RX  Patte D1 TX |
| Bouton poussoir (si besoin) | À voir si besoin |

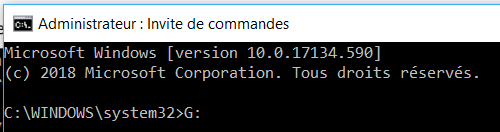
**Comment installer pyserial-3.4**

Ouvrir l’invite de commande windows en faisant un clic-droit🡪Exécuter en tant qu’administrateur

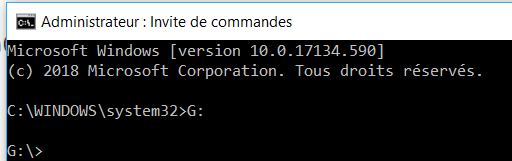




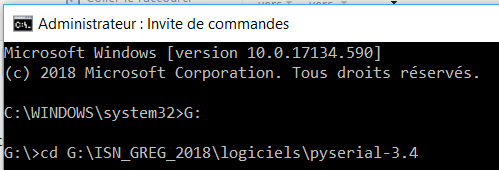
Taper le nom du lecteur sur lequel il faut se positionner par exemple, pour ma clé USB, G :



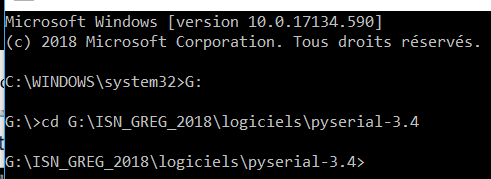
Puis valider avec entrée, vous êtes maintenant positionné sur ce lecteur



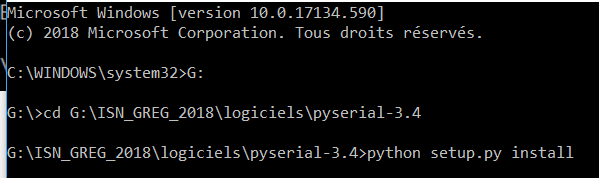
Taper l’instruction cd suivi du chemin contenant le dossier pyserial-3.4



Valider avec entrée



Taper la ligne de commande python setup.py install



Une fois l’installation réalisée avec succès taper exit dans l’invite de commandes