06_GUI of the robot

Introduction

The GUI allows the user to send instructions to the robot and permits the robot to send back relevant information. It is a useful tool for demonstration and debug purposes by showing the state of the robot to the user in a user-friendly way.

This GUI has been developed in Python 3.7, minors adjustments are necessary to work with previous versions of Python. The window of the GUI is programmed with PyQt4 which adds flexibility to the GUI as PyQt4 makes it extremely simple to add and remove fields in the window.

What you need to use/change the GUI

- Python 3.7
- PyQt4
- Simple text editors (kate, sublime text, etc)
- Bluetooth connectivity (either with Bluetooth embedded in your computer of with Bluetooth dongle)

How the script works

- An object "Window" is created In this class, the window is divided into as many parts as you want, there is "graphical" parts (hereinafter referred as groupBox) which will be the different parts of the window and of course some functions to interact with those graphical elements .
- In those groupBox, you create widgets which can be labels, buttons, radio buttons, texts, icons, drawings, etc (see PyQT4 documentation for further details).
- If one widget has to be reused in another part of the windows, it is declared as "self.widget" for example: `self.distObst = QtGui.QLabel("N/A")` because the text "N/A" is to be actualized in a function outside the groupbox `distObst` was declared in.

- The communication with the COM port is handled in the window object, to identify the COM port on your laptop corresponding to the Bluetooth module (which can be a external Bluetooth dongle), Go to Device manager->Ports(COM & LPT). If the script can't connect to the Bluetooth module, the application will fail to open and you will get this message in the command prompt: `serial.serialutil.SerialException: could not open port 'COM11': FileNotFoundError(2, 'The system cannot find the file specified.', None, 2)`

- There is no interrupts in python so there is a few different workaround, the one used in the script is the polling of the serial communication in a dedicated thread as to not slow down the rest of the application.

Use the GUI as an executable

The script once finished is transformed into an executable for Windows using Pyinstaller (other solution are available but this the most commonly used). Transforming the script into an executable allows every Windows powered computer to use the GUI with out python and its packages.

See here how to install Pyinstaller.

How to transform python script into an executable (step by step):

1. Open the command prompt and go to the folder containing the python script
2. Type “pyinstaller --onefile RobotGUI.py”, This command creates a few folder, the one containing the executable application is called “dist”
3. Move the picture “logo.png” in the “dist” folder
4. Launch the application by clicking on RobotGUI.exe