C181055 CPSC1 – PYTHON DEMO SCRIPTS



Important note: any Graphical User Interface (GUI), script or programming code is provided 'As Is' without any express or implied warranty of any kind and is provided only to assist users getting acquainted with the CPSC1 controller, available commands and parameters and methods to communicate with the controller to drive positioners, actuators or stages.

Please understand that all scripts and code are simply demo's and therefore are not thoroughly tested and may still contain bugs or have features missing.

JPE encourages users to develop their own higher-level control / user software.

1. INTRODUCTION

Demo scripts written in the Python programming language are available for download on the JPE website. These programs can be used for simple interaction with the CPSC1 controller and the supported modules installed (CADM RSM, OEM etc) to control positioners, actuators or stages.

2. REQUIREMENTS

All code is written using Python version 3.9.

All demo scripts communicate with the CPSC1 (only) via the USB Virtual COM port interface. Therefor it is also required to connect the CPSC to the host device via USB and have the <u>pyserial</u> package installed. GUI's have been created using the <u>tkinter</u> package.

<u>Tip</u> : install the <u>A</u> and modules. If r	<u>naconda</u> distribution. T necessary, install the _{PY}	his Python devenues of the second s The second s	lopment platform includes all required in the second second second second second second second second second se	vired packages
	NDA.NAVIGATOR			Connect V
✿ Home	Search Environments	Q Installed	Channels Update index	pyserial X
The Environments	base (root)	Name	✓ T Description	Version
🗳 Learning		🗹 pyserial	O Python serial port access library	3.5
After installing A scripts. Use Spyc	naconda, start the Spy ler to run/start any of th	r der IDE and nav ne scripts.	igate to the folder containing the	Python demo



BANK: ING Bank NV IBAN: NL 93 INGB 0679351523 BIC: INGBNL2A TRADE REGISTER: 14081561 VAT: NL 8134.92.841.B01 Page 1/5



C181055 CPSC1 – PYTHON DEMO SCRIPTS



3. THE BASICS

To get started simply download the zip file and unpack the contents to a folder of choice.

Important note: inside this folder there's a (sub)folder called \CpscInterfaces\. Inside there is a Python file called CpscSerialInterface.py. Each demo script uses this file to communicate with the CPSC1, so do not change the name of this file or the name of the folder containing this file.

Running a script 3.1

Using the Spyder IDE, double-click on a .py file and press the green 'Play' button to start the script. If the script contains a GUI, this should pop-up on the screen. Close a GUI using the "X"-button on the top-right of the GUI.

😨 Spyder (P	ython 3.9)	© CPSC1 BaseDrive v0.2 - □ ×
File Falls Co	and Course Due Debug Conseles Designs Table View Hole	COM port: 1 🚖 Address: 1 🛁 Frequency [Hz]: 600 🕏 Drive Factor: 1 🛁
File Edit Se	arch source kun Debug Consoles Projects loois view Help	Baudrate: 115200 Step Size 1%: 100 🕏 Stage Type: CLA2201
	/	Snow Commands: I Temperature [K]: 2/3 👻 [No.] of Steps: 0
	PSC1 Python Scrints/CPSC Due GL (FEX) IT-V() 2 DV	
D: 00_CMP (C		DIKEU (PgUp) STOP [End] DIKET (PgDw)
- CPSC	.1_BaseDrive_GU1-v0.2.py ×	Command History
1	*****	
2	# File name: CPSC1 BaseDrive GUI vX.v.pv	
3	# Creation: 2022-12-07	
4	# Last Update: 2022-12-21	
5	# Author: JPE	
6	# Python version: 3.9. requires pyserial and tkinter	
7	# Description: MVP GUT controlling CPSC1 "Basedrive" mode of or	
8	# Disclaimer: This program is provided 'As Is' without any exp	
9	# implied warranty of any kind.	
10		
11		
12	verNumber = 100.21	
13		· · · · · · · · · · · · · · · · · · ·
14	# 3rd party imports	Device Manager
15	import time	Clear command history Info Driver Check Positioner Check (Windows)
16	import tkinter as tk	
17		JPE - Driven by innovation jpe-innovations.com



BANK: ING Bank NV IBAN: NL 93 INGB 0679351523 BIC: INGBNL2A VAT: NL 8134.92.841.B01



C181055 CPSC1 – PYTHON DEMO SCRIPTS

If a script does not contain a GUI, any feedback will be "printed" to the console (IPython console in the Spyder IDE). Also understand that if a script does not contain a GUI, parameters must be set inside the script code prior to running the script. This is typically done near the top of the script:



3.2 General tips and tricks

6199 AG Maastricht-Airport

The Netherlands

Most (GUI-)scripts should be straight forward to use, but this section contains a few tips and tricks to get up to speed more quickly.

<u>Tip:</u> each GUI contains a COM port selection box and a Baud Rate input field. Set accordingly.					
COM port: 1 🚖					
Baudrate: 115200					
<u>Tip:</u> check the Send Commands che pushing a button. Together with th how to use the available software c	eckbox to display the actual commands send to the controller when e Software User Manual this is a quick way to get an understanding on commands.				
Show Commands:					
<u>Tip</u>: the Command History window is set) and responses from the contr to the controller in blue and respon	shows commands to the controller (if the Show Commands checkbox roller. Internal script messages are displayed in black text, commands ses in green. Any errors will be displayed in red:				
Command History					
> Stop movement					
> STP 1 < Stopping the stage.					
> Stop movement > STP 1					
Cannot connect to controller! Please check if CPSC1 is connecte number and Baudrate has been set.	d to the host via USB and the correct COM port				
<u>Tip:</u> some GUI scripts have addition to perform some checks (Driver Che	nal buttons to display controller (firmware) Information (Info button) or eck, Positioner Check).				
JPE					
4 +31 (0) 43 3585 777 E	BANK: ING Bank NV				
www.jpe.nl	BAN: NL 93 INGB 0679351523				

TRADE REGISTER: 14081561

VAT: NL 8134.92.841.B01

Page 3 / 5

C181055 CPSC1 – PYTHON DEMO SCRIPTS



Clear command history Info Driver Check Positioner Check (Windows)						
Command History						
<pre>> Get firmware version information > /VER < CPSCl version: v8.0.20220221 > FIV 1 < Slot 1: CADM2.7.3.20210802 > FIV 2 < Slot 2: CADM2.7.3.20210802 > FIV 3 < Slot 3: CADM2.7.3.20201222 > FIV 4 < Slot 4: RSM.7.3.20210802 > FIV 5 < Slot 5: Error, Address is invalid. > FIV 6 < Slot 6: Error, Address is invalid.</pre>						
<u>Tip:</u> some buttons have a keyboard shortcut. This way you can quickly start or stop a movement using a keystroke. The corresponding keyboard shortcut is displayed between []-brackets.						
DIR=0 [PgUp] STOP [End] DIR=1 [PgDw]						

4. SERIAL COMMUNICATION DETAILS

Most demo scripts contain a GUI created with the tkinter package and most code in such a script purely serves to create a GUI, setup the buttons and user inputs etc. Basically, only the following is required to send commands to and receive messages from the controller:

Import CpscSerialInterface.py:

from CpscInterfaces import CpscSerialInterface as CpscSerial

And execute the function:

with CpscSerial.CpscSerialInterface([COMx], [BaudRate]) as usbVcp: response = usbVcp.WriteRead([command], [termination])

The WriteRead function sends a [command] (formatted as string) via the USB Virtual COM port to the controller. Any return message from the controller will be stored in response. If [termination] is set to 1, the function will add a "\r\n" after each [command] and will strip "\r\n" from any received message. [termination] set to 0 (zero) won't do this and it'll be up to the user to add/remove these string termination characters. By default [termination] is set to 1 in each demo script. [COMx] and [BaudRate] should be formatted as strings, for example: COM1 and 115200.

BANK: ING Bank NV IBAN: NL 93 INGB 0679351523 BIC: INGBNL2A TRADE REGISTER: 14081561 VAT: NL 8134.92.841.B01



C181055 CPSC1 – PYTHON DEMO SCRIPTS

Good practice is to use a try - except when sending a command, so that the script won't crash immediately if the Virtual COM port cannot be opened/used. For example:

try:

```
with CpscSerial.CpscSerialInterface('COM1', '115200') as usbVcp:
       response = usbVcp.WriteRead(cmdStp, 1)
except IOError:
   print('Communication error')
```



BANK: ING Bank NV IBAN: NL 93 INGB 0679351523 BIC: INGBNL2A 6199 AG Maastricht-Airport TRADE REGISTER: 14081561 VAT: NL 8134.92.841.B01

Page 5 / 5