



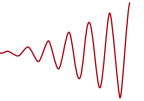
# Manuel d'utilisation du pilote Zolix

« Pilote Python Zolix »

**Date :** 01/12/2020

**Client :** PHOTONICS BRETAGNE

**Référence :** AR200273MU-01



### Approbation du document

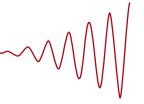
	DATE	NOM
<b>Rédigé</b>	01/12/2020	<b>Alan Royer</b>
<b>Vérifié</b>	01/12/2020	<b>Fabrice VOMSCHEID</b>

### Documents Applicables

N°	Identification	Description

### Documents de références

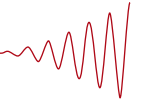
N°	Identification	Description



# Glossaire

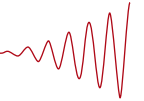
Le glossaire explicite les sigles et abréviations rencontrés dans ce document.

[illegible]



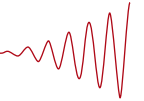
## Evolutions

Date	Version	§/Page	Changements
01/12/2020	01	*	Version initiale du document



# Table des matières

<b>1.</b>	<b>DESCRIPTION DU PROJET.....</b>	<b>6</b>
1.1.	INTRODUCTION .....	6
1.2.	CONTEXTE.....	6
1.3.	OBJET .....	6
<b>2.</b>	<b>MANUEL D'UTILISATION.....</b>	<b>7</b>
2.1.	INSTALLATION .....	7
2.1.1.	Livrables.....	7
2.1.2.	Prérequis .....	7
2.1.3.	Installation .....	7
2.2.	FONCTIONNEMENT DU SYSTEME .....	8
2.3.	INTERFACE LOGICIELLE « SERVEUR ZOLIX » .....	9
2.3.1.	Serveur socket .....	9
2.3.2.	Affichage des informations .....	10
2.4.	PILOTE CLIENT ZOLIX .....	10
2.4.1.	Configuration .....	10
2.4.2.	Utilisation des fonctions.....	11



## 1. DESCRIPTION DU PROJET

### 1.1. INTRODUCTION

Ce document constitue le Manuel d'utilisation du pilote Zolix de VFI SYSTEM à PHOTONICS BRETAGNE pour le marché intitulé « Pilote Python Zolix »

### 1.2. CONTEXTE

Dans le cadre d'un de ses projets, PHOTONICS BRETAGNE souhaite pouvoir asservir un monochromateur en le contrôlant en langage Python.

Le monochromateur est de la marque ZOLIX et le numéro du modèle est omni-λ200i.

Actuellement, pour modifier les paramètres du monochromateur (réseau, activation des filtres, longueur d'onde centrale...), PHOTONICS BRETAGNE utilise le logiciel dédié « Omnispec ». Avec ce logiciel, l'utilisateur est obligé de rentrer manuellement les paramètres souhaités. Il est donc impossible d'automatiser les expériences.

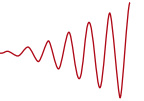
Une librairie de fonctions « Omnispec » en C++ a été fournie avec le monochromateur mais PHOTONICS BRETAGNE n'a pas réussi à les utiliser.

### 1.3. OBJET

Ce document présente l'interface logicielle du serveur Zolix ainsi que l'explication des fonctions composant le driver.

Le document est structuré de la manière suivante :

- Fonctionnement du système
- Présentation de l'interface logicielle du serveur Zolix
- Présentation du pilote client Zolix



## 2. MANUEL D'UTILISATION

### 2.1. INSTALLATION

#### 2.1.1. LIVRABLES

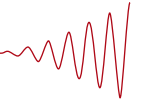
- Framework 4.7.2
- Driver USB ZOLIX
- Setup d'installation de l'interface graphique (passerelle)
- Bibliothèque de fonctions Python

#### 2.1.2. PREREQUIS

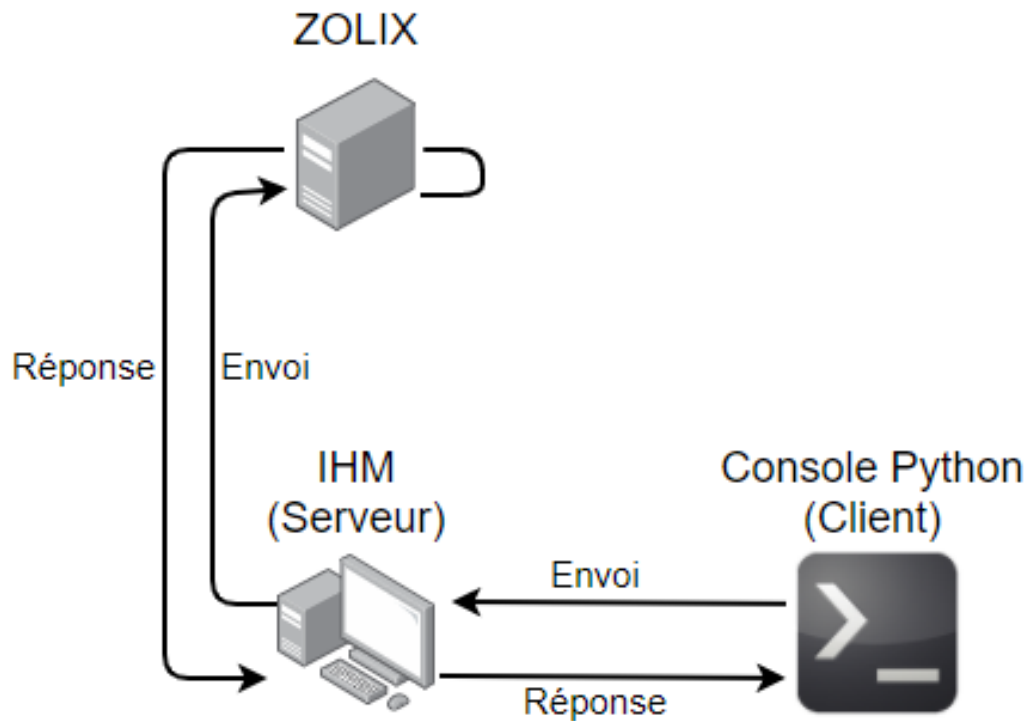
Le système d'exploitation du PC hôte de la passerelle (« Serveur Zolix ») doit être Windows 7 32Bits.

#### 2.1.3. INSTALLATION

1. Installation du Framework 4.0  
(« Dependances\dotNetFx40\_Full\_x86\_x64.exe »)
2. Installation du driver USB. Désigner le dossier du pilote USB ZOLIX (dossier « Dependances\Driver USB Zolix ») à la détection par Windows de la connexion USB.
3. Installation du setup du SDK OmniSpec (fichier « Setup.exe » du dossier « Dependances\Omnispec-Setup »)
4. Installation du Setup de la passerelle « Setup Serveur Zolix v1.0.0»
5. Intégrer la bibliothèque **zolix\_gateway** dans votre projet PYTHON (« Dependances\Python Zolix Gateway »).



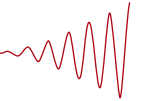
## 2.2. FONCTIONNEMENT DU SYSTEME



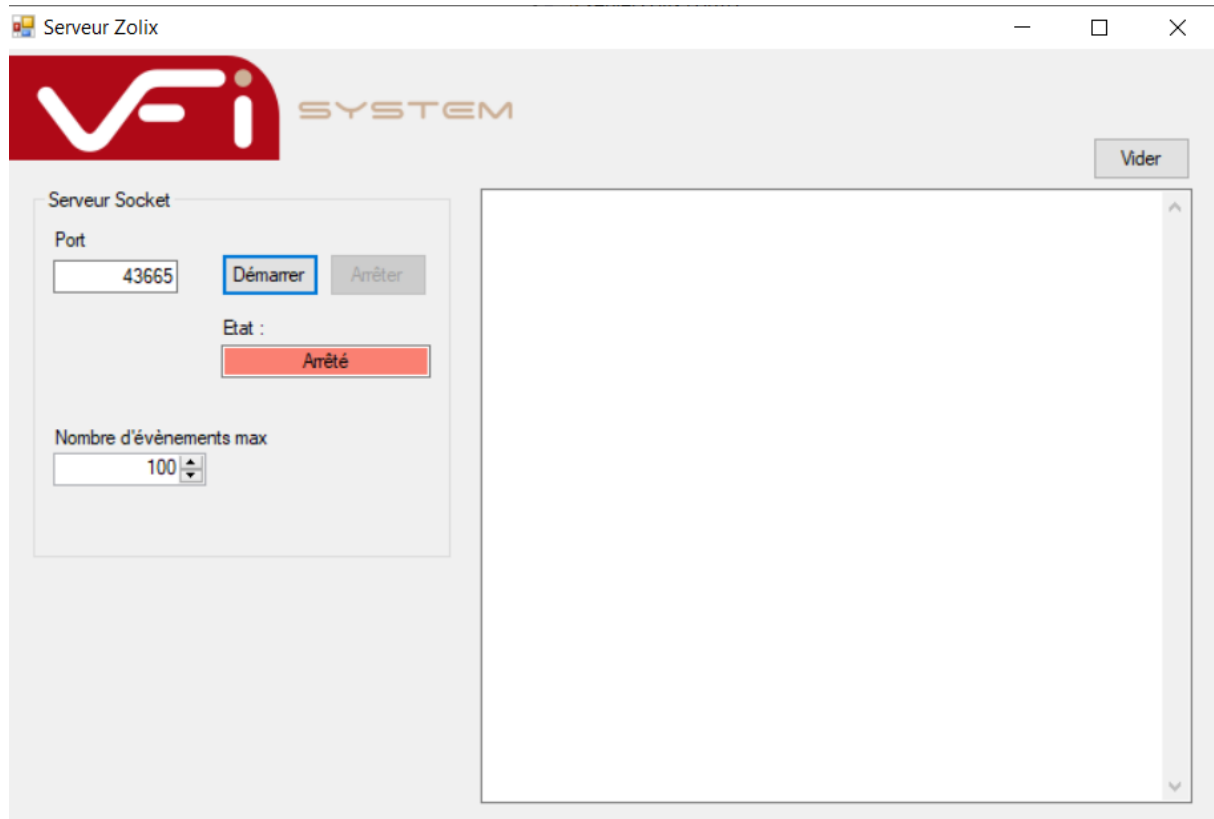
Les étapes à réaliser pour mettre en place l'environnement sont les suivantes :

1. Connecter le PC pilote avec le monochromateur en USB
2. Lancer l'interface graphique : « Serveur Zolix »
3. Démarrer le serveur avec bouton « Démarrer » sur l'IHM
4. Configurer le client
5. La librairie est prête à être exploitée.





## 2.3. INTERFACE LOGICIELLE « SERVEUR ZOLIX »



### 2.3.1. SERVEUR SOCKET

Pour activer le serveur et rendre l'appareil accessible au réseau il est nécessaire d'entrer au préalable l'information suivante :

- **Port**

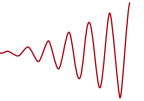
Pour gérer le statut du serveur, deux boutons sont à disposition :

- Bouton **Démarrer**
- Bouton **Arrêter**

Une fois lancé, le serveur sera en écoute sur toutes les adresses IPV4 et adresses locales disponibles.

Un indicateur permet de vérifier l'état de la connexion en temps réel :

- **Etat de la connexion**



### **2.3.2. AFFICHAGE DES INFORMATIONS**

Après avoir lancé le serveur il est possible de suivre les interactions entre ce dernier et le client dans la fenêtre de suivi des événements à droite. On retrouvera des informations telles que :

- Les adresses en écoute
- Le port d'écoute
- Les commandes reçues
- Les résultats envoyés
- Etat de la connexion

Pour éviter toute saturation, le journal sera limité aux N derniers événements (Paramètre « Nombre d'événement max » sur l'IHM).

## **2.4. PILOTE CLIENT ZOLIX**

### **2.4.1. CONFIGURATION**

Avant de pouvoir appeler les fonctions Python il est nécessaire

1. D'importer la classe ZolixGateway :

```
from zolix.app.zolix_gateway import ZolixGateway
```

1. Modifier les informations du serveur dans le fichier gateway\_configuration.py.

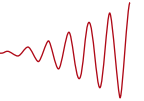
```
server_ip = AdresseIP  
server_port = PortIP
```

2. Instancier la classe ZolixGateway :

```
zolix_gateway = ZolixGateway(gateway_configuration.server_ip,  
gateway_configuration.server_port)
```

3. Se connecter :

```
zolix_gateway.connect_to_server()
```



### **2.4.2. UTILISATION DES FONCTIONS**

cf. : voir la documentation technique de Zolix qui liste toutes les fonctions utilisables (EN-OperationManual.pdf).

Pour utiliser une fonction, il faut faire appel à l'objet ZolixGateway instancié au point 2 di paragraphe précédent.

Exemple d'utilisation d'une fonction :

```
zolix_gateway.set_total_steps(56245)
```