

Pythonでパルスジェネレータを制御する方法（2026年版）

～ SCPIコマンドによる外部制御入門 ～

はじめに

Quantum Composers製パルスジェネレータは、SCPI（Standard Commands for Programmable Instruments）コマンドによる外部制御に対応しています。

Pythonを利用することで、

- パルス条件の自動変更
- 条件スイープ
- 長時間評価試験
- 他の計測器との連携

などを実現できます。

本記事では、Python環境の準備から、パルスジェネレータとの通信確認、基本的なSCPI制御までを紹介します。

※「なぜ実験自動化を行うのか」「どのような応用が可能か」については、別資料「AI時代の実験自動化」をご参照ください。

SCPIコマンドとは

SCPI（Standard Commands for Programmable Instruments）は、計測器制御のための標準コマンド体系です。

Quantum Composers製パルスジェネレータでは、テキスト形式のコマンドを送信することで各種設定を変更できます。

例：

```
None
:PULSE1:DELAY 1E-6
:PULSE1:WIDTH 100E-9
```

この例では、

- Channel 1 Delay = 1 μ s
- Channel 1 Width = 100 ns

を設定しています。

【ヒント】

GUIソフトウェアのSCPIログ機能を利用すると、GUI操作時に送信されるSCPIコマンドを確認できます。

設定変更時に実際にどのSCPIコマンドが送信されているかを確認できるため、Python制御用のSCPIコマンド作成にも役立ちます。

必要なもの

Python制御を始めるために、以下を準備します。

- Quantum Composers パルスジェネレータ
 - Windows PC
 - USBケーブル
 - Python
 - pySerial
-

Pythonのインストール

Python公式サイトから最新版をインストールします。

インストール時は、

「Add Python to PATH」

にチェックを入れてください。

インストール後、コマンドプロンプトで以下を実行します。

```
Shell  
python --version
```

バージョンが表示されれば正常です。

pySerialのインストール

USB仮想COMポートやRS-232通信には pySerial を利用します。

コマンドプロンプトで以下を実行します。

```
Shell
pip install pyserial
```

インストール後、

```
Shell
pip show pyserial
```

で確認できます。

パルスジェネレータを接続する

パルスジェネレータをUSBで接続します。

Windowsのデバイスマネージャーを開き、

「ポート (COMとLPT) 」

に表示されるCOM番号を確認します。

例：

```
None
USB Serial Port (COM3)
```

この場合、COM番号は「COM3」です。

最初の通信確認

まずは装置と通信できることを確認します。

SCPI共通コマンド

```
None
*IDN?
```

を送信します。

サンプルコード：

Python

```
import serial

pg = serial.Serial(
    "COM3",
    baudrate=115200,
    timeout=1
)

pg.write(b"*IDN?\r\n")

response = pg.readline()

print(response.decode())

pg.close()
```

正常に通信できる場合、

None

```
Quantum Composers,9520,...
```

のような応答が返ります。

Delayを変更する

Channel 1 の Delay を変更する例です。

Python

```
import serial

pg = serial.Serial("COM3",115200)

pg.write(b":PULSE1:DELAY 1E-6\r\n")

pg.close()
```

実行すると Delay が 1 μ s に変更されます。

Widthを変更する

Channel 1 の Width を変更する例です。

```
Python  
pg.write(b":PULSE1:WIDTH 100E-9\r\n")
```

実行すると Width が 100 ns に変更されます。

よくあるトラブル

COMポートが見つからない

USBドライバーが正しくインストールされているか確認してください。

通信エラーになる

COM番号が正しいか確認してください。

応答が返ってこない

ボーレート設定と終端文字を確認してください。

一般的には、

```
None  
\r\n
```

を使用します。

Pythonが起動しない

PATH設定が正しく行われているか確認してください。

次のステップ

ここまで確認できれば、PythonからQuantum Composers製パルスジェネレータを制御する基本環境は整っています。

次のステップとして、

- 条件スイープ自動化
- CSVログ保存
- オンロスコープ連携
- カメラ同期
- 長時間評価試験

などへ発展させることができます。

まとめ

PythonとSCPIコマンドを利用することで、Quantum Composers製パルスジェネレータを容易に外部制御できます。

まずは、

1. Pythonインストール
2. pySerialインストール
3. COMポート確認
4. *IDN? 通信確認

まで実施することをおすすめします。

通信確認ができれば、その後の自動化開発は大きく前進します。