

# AI時代の実験自動化

## ～Pythonで始める計測・制御の自動化～

### はじめに

近年、生成AIの普及により、プログラミング経験が少ない方でもソフトウェア開発に取り組みやすくなっています。

これまで実験自動化は、プログラミング経験者や専門のソフトウェア開発者が行うものという印象がありました。しかし現在では、やりたい実験手順やフローチャートを整理し、AIを活用しながらプログラムを作成することも現実的になっています。

Quantum Composers パルスジェネレータは、SCPIコマンドによる外部制御に対応しており、Pythonと組み合わせることで様々な実験シーケンスを構築できます。

本資料では、Pythonを活用した実験自動化の考え方や活用例についてご紹介します。

---

### なぜ今、実験自動化なのか

研究や開発の現場では、

- 条件を変えながら測定を繰り返したい
- 長時間の評価試験を行いたい
- 夜間や休日でも測定を継続したい
- 複数機器を連携させたい

といった要望があります。

一方で、

- プログラミングは専門外
- ソフトウェア開発に時間を割けない

という理由から、自動化を見送っていたケースも少なくありません。

近年は生成AIを活用することで、Pythonコードの作成やデバッグ支援を受けながら開発を進めることが可能になりました。

実験自動化は、以前よりも身近な選択肢になっています。

---

## Quantum Composers と Python制御

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

Pythonからコマンドを送信することで、

- Delay設定変更
- Width設定変更
- 繰り返し周波数変更
- Burst条件変更
- 出力制御

などをプログラムから実行できます。

これにより、フロントパネルやGUIで繰り返し行っていた操作を、自動化することが可能になります。

---

## Pythonを活用するとできること

例えば以下のような測定シーケンスを実行できます。

- Delayを変更しながら測定
- Widthを変更しながら測定
- 周波数を変更しながら評価試験
- Burst条件を変更しながら評価
- 長時間試験の自動実行
- 一定時間ごとの測定実行

人が手作業で行っていた繰り返し作業を、プログラムに従って実行できます

---

## 活用例

### カメラ同期

パルスジェネレータを用いてカメラ露光やレーザー発振のタイミングを同期します。Pythonを活用することで、同期条件を変更しながら測定シーケンスを実行できます（図1）。

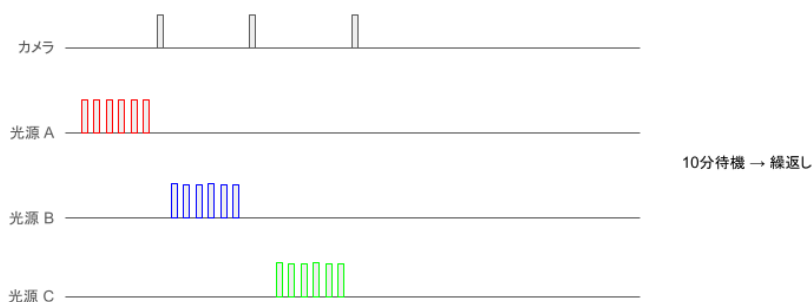


図1 光源切替とカメラ同期を組み合わせた自動測定シーケンス例

## 複数条件での自動測定

DelayやWidthなどの設定を変更しながら測定を繰り返し実行します。  
最適条件の探索やパラメータ評価を効率化できます。  
Pythonを活用することで、あらかじめ設定した複数条件を順番に実行できます（図2）。



図2 条件変更を伴う自動測定シーケンス例

---

## 波長切替測定

フィルタホイールやAOTFなどと組み合わせることで、波長条件を切り替えながら測定を行うシステムを構築できます。

---

## 植物工場・光応答評価

照射条件やタイミングを変更しながら長時間評価を行う用途にも活用できます。

---

## 長時間評価試験

数時間から数日にわたり、あらかじめ設定したシーケンスを自動実行できます。

---

## Do It Yourself の時代へ

以前は、

「ご自分でプログラミングできれば可能です」

とご案内することが多くありました。

現在は、生成AIを活用しながらプログラムを作成できるようになり、研究者や技術者自身が実験自動化に取り組みやすくなっています。

フローチャートや測定手順を整理し、それをPythonプログラムとして実装することも以前より身近になりました。

---

## 実験自動化のご相談も歓迎します

Quantum Composers パルスジェネレータを活用した実験自動化について、

- こんなことはできる？
- こういう測定を自動化したい
- Python制御に興味がある

などがありましたら、お気軽にご相談ください。

実験シーケンスの検討やパルスジェネレータの活用方法についてご案内いたします。

---

## 初めての方へ

Python環境の構築方法や基本的な通信確認手順については、

「Pythonでの制御方法（2026版）」

をご参照ください。

初めての方は、こちらの資料を参照して始めてください。

# はじめに

## AIとPythonで実験を自動化しませんか？

近年、ChatGPTをはじめとする生成AIの普及により、プログラミングのハードルは大きく下がりました。

従来であれば、実験装置の自動制御や測定システムの構築には専門的なプログラミング知識が必要でした。しかし現在では、AIを活用することで、研究者や技術者自身が実験自動化に取り組みやすい環境が整いつつあります。

Quantum Composers社製パルスジェネレータは、USB、RS-232、Ethernetなどを介した外部制御に対応しており、Pythonとの組み合わせによって柔軟な実験システムを構築できます。

本ガイドでは、Python未経験者の方にも分かりやすく、AI時代における実験自動化の考え方や活用例をご紹介します。

---

## なぜ今、実験自動化の提案なのか

### 従来は高いハードルがあった

これまでは、

- Pythonの学習
- 通信ライブラリの調査
- シリアル通信の理解
- SCPIコマンドの習得
- デバッグ作業

などが必要でした。

そのため、

「やってみたいが難しそう」

と感じる方も少なくありませんでした。

### AIによって状況が変わった

現在では、ChatGPTなどの生成AIを活用することで、

「やりたいこと」

を自然言語で説明するだけで、Pythonコードのたたき台を作成できます。

例えば、

「QC9520のパルス幅を100  $\mu$ sから1 msまで変化させるPythonコードを書いてください」

と依頼するだけで、基本的なプログラムを作成できます。

さらに、

- エラー内容の解析
- コード修正案の提案
- 通信設定の確認

などもAIが支援してくれます。

---

## Quantum Composersでできること

PythonからSCPIコマンドを送信することで、パルスジェネレータの各種設定を制御できます。

主な制御項目：

- Delay変更
- Width変更
- Burst設定
- Trigger設定
- 出力ON/OFF
- 設定保存／呼び出し

研究用途では、これらを組み合わせることで様々な実験シーケンスを構築できます。

---

## 実験自動化の活用例

### カメラ同期

レーザー、パルスジェネレータ、カメラを同期させ、測定条件を自動変更しながら画像取得を行う。

---

### 波長切替実験

フィルタホイールやAOTFと連携し、

- 波長切替
- パルス幅変更
- 繰返し周波数変更

を自動化する。

---

## 植物工場・光応答評価

LED光源を制御し、

- ・赤色光
- ・青色光
- ・緑色光
- ・近赤外光

などを自動切替しながら測定を行う。

---

## 長時間測定

数時間から数日に及ぶ実験を自動実行し、

- ・条件変更
- ・測定結果保存
- ・ログ記録

を自動化する。

---

## Python未経験者でも始められる

### 必要なもの

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

### 最初の一步

まずはパルスジェネレータと通信できることを確認します。

代表的な確認方法として、

\*IDN?

コマンドを送信し、装置から応答が返ってくることを確認します。

ここまで確認できれば、基本的な制御環境は整っています。

---

## Do It Yourself のすすめ

以前であれば、実験自動化システムの構築は専門業者やソフトウェア開発会社へ依頼するケースも多く見られました。

しかし現在では、AIを活用することで、研究者や技術者自身が実験自動化に取り組める時代になっています。

まずは、

「パルス幅を変更する」

「100回繰り返し測定する」

といった小さな自動化から始めてみてください。

その積み重ねが、より高度な実験システム構築につながります。

---

## コラム

### 「Pythonでの制御方法」（2024版）が生まれたきっかけ

本資料の元となった「Pythonでの制御方法」（2024版）は、実際のユーザーサポート案件から生まれました。

新たに導入したQC9528で、既存装置では正常に動作していたPythonプログラムが動作しないというお問い合わせがあり、実機をお預かりして検証を実施しました。

その過程で、

- Python環境構築
- pyserial導入
- シリアル通信確認
- SCPI通信確認

などを一から調査することになりました。

最終的にはFirmwareに起因する問題であることが判明し、Firmware更新によって解決しました。

その際に得られた知見を整理したものが、「Pythonでの制御方法」（2024版）です。

現在ではAIを活用することで、当時よりもはるかに容易に実験自動化へ取り組めるようになっています。

---

## 関連資料

- Pythonでの制御方法（2024版）
  - SCPIコマンドリファレンス
  - 94xx/95xx GUI活用ガイド（予定）
-