

事前のアルゴリズム検証や組み込み後の動作検証にイイ

Pythonを取り入れてマイコンの開発効率アップ

篠原 規将

組み込み開発では、センサなどのアナログ信号の評価や、作った製品の評価をする場合にオシロスコープなどの測定器を使ってデータを取得し、フィルタなどの信号処理をPCで行って解析します。

しかし、デジタル・インターフェースを持つセンサから情報を取得する場合や製品のテストで測定器を常設できない場合など、測定器では直接測れないことがあります。その際は、マイコンを使って必要なデータを取得し、PCに取り込んでデータを処理します。

組み込み開発の現場ではC/C++を使うことが多いため、取得したデータを処理する場合にもC/C++を利用することが多いです。しかし、データ処理をC/C++だけで行おうとするとプログラムのコード数が多くなり、デバッグ作業が増え、逆に開発効率を下げてしまうことがあります。

そこで、Python言語を取り入れることで、それぞれの言語の得意分野を生かして開発の効率を上げられます。

マイコンのファームウェアはC/C++で作るとしても、その前段階でアルゴリズム検証などにPythonを利用できます。

組み込み開発にPythonを使う理由

● データの加工にイイ

マイコンのA-Dコンバータやオシロスコープなどを

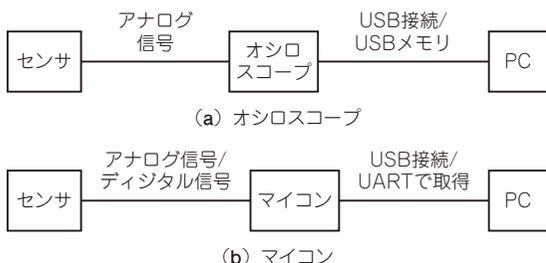


図1 センサ・データをPCに取り込む方法

使って取得した測定器のデータは、ノイズなどの問題もありそのままでは使えない場合もあります。この場合、データをPCで処理して使える形にします(図1)。

Pythonはフリーで使うことができ、グラフ化や、信号処理などのライブラリも豊富に公開されています(表1)。使い方に関してもネット上に情報が豊富です。

● 動作の検証にイイ

図2に示すようにPythonからC言語で作成したDLLファイルを読み込むこともできます。

Windows環境を使用している場合、このDLLファイル作成にはVisual Studioを使うと便利です。Visual Studioで「ダイナミック・リンクライブラリ(DLL)」のプロジェクトを作成し、C言語を使ってマイコン上で行う処理を実装します。コンパイルすると、Pythonから読み込むことができるDLLファイルを生成できます。

モジュール定義ファイルを作成し、DLLファイル

表1 マイコン開発によく使うPythonライブラリ

ライブラリ名	用途
Matplotlib	グラフの作成、データの見える化
NumPy	数値計算
SciPy	信号処理、入力信号のサンプリング・レート変換、フィルタ処理によるデータの平滑化

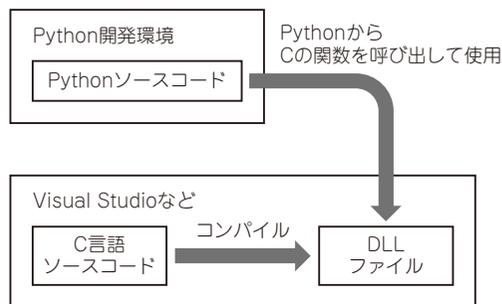


図2 PythonからCの関数を呼び出せる