

第2章 **固定小数点演算/フィルタ/オシレータの工夫で**

リアルタイム処理のために 軽量化! シンセサイザの製作

石垣 良

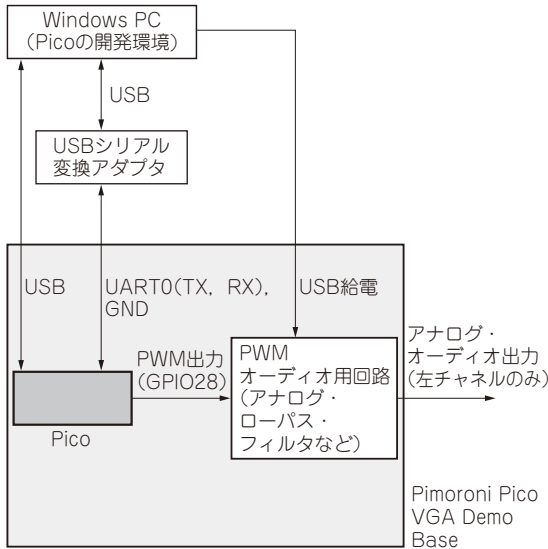


図1 試作したPicoシンセサイザの構成

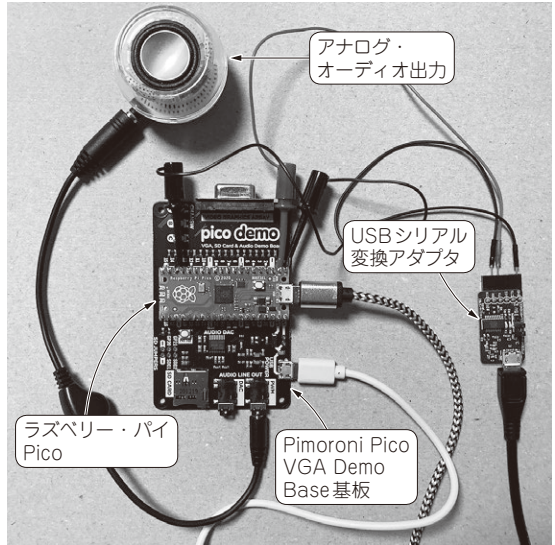


写真1 Picoシンセサイザの外観

主に Arduino Uno (AVRマイコン) で動くシンセサイザを作ってきた筆者が、ラズベリー・パイ Pico (以降、Pico) の信号処理性能を把握するために、シンプルなシンセサイザを試作しました。Picoでリアルタイム信号処理を行う場合は、なるべく固定小数点演算を使用するのが良さそうです。

ハードウェアの構成

● **市販のPWMオーディオ用回路搭載基板を利用**
図1が試作したPicoシンセサイザのシステム構成、写真1が外観です。筆者はPicoの開発にWindows PCを使っています^{注1}。

PicoにはD-Aコンバータが搭載されていません。本稿では、制御が簡単なPWMでオーディオ出力(PWMキャリア周波数は48kHz、サンプリング周波

数も同じ)を行います。また、文献(1)の「3.4.1. PWM Audio」で説明されている参考回路が実装された、市販品のPimoroni Pico VGA Demo Base基板を使用します。この基板は以下で構成します。

- ロジック・バッファ
- カットオフ周波数が約7.2kHzのアナログ・ローパス・フィルタ
- カットオフ周波数が約33.9Hzのアナログ・ハイパス・フィルタ

アナログ・オーディオ出力(左チャンネルのみ)は、外部アンプのライン入力に接続します。プログラムの開始と電源OFF時にポップ音(ポップ・ノイズ)が発生する点に注意が必要です。

シンセサイザのプログラムはUART通信(通信速度は115200bps)で、制御やモニタリングを行います。写真1ではICクリップを用いてUART0 TX (Picoの1ピン), RX (2ピン), GNDを、USB-シリアル変換アダプタに接続しています。

注1: Windowsの他にもMacやラズベリー・パイを使用可能(メインでサポートされるのはラズベリー・パイ)。

(1) Hardware design with RP2040, Release 1.4.1, 2021年4月13日。
<https://datasheets.raspberrypi.org/rp2040/hardware-design-with-rp2040.pdf>