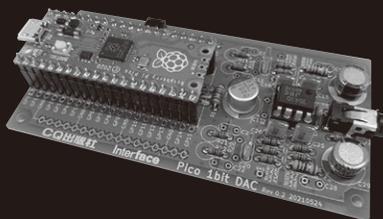


サンプリング・レートの切り替えからノイズ低減、ハイレゾ対応まで!

ラズパイPico DAC [完全版]の製作



第5回 ハイレゾ対応②…ソフトウェア&ハードウェアの実装・改造

geachlab, 丸石 康

表1 ラズパイPico DACで対応するオーディオ・フォーマット
ソフトウェアおよびハードウェアの実装・改造により、最終的に
96kHz/24ビットを含む10フォーマットに対応させる

ビット深度軸	周波数 系列軸	周波数倍率軸		
		×1	×2	×4
16ビット系	44.1kHz系	44.1kHz*	88.2kHz	176.4kHz
	48kHz系	48kHz*	96kHz	192kHz
24ビット系	44.1kHz系	44.1kHz	88.2kHz	176.4kHz
	48kHz系	48kHz	96kHz	192kHz

※：対応済みフォーマット

本誌2021年8月号の特集 第6部 第1章(以下、初出記事)で、ラズベリー・パイPicoを使ったUSBオーディオDAC「ラズパイPico DAC」の製作記事を掲載しました。

初出記事では、USBオーディオDACの原理実装に注力したので、機能や性能は限定的でした。サンプリング・レート/分解能は48kHz/16ビットの一跳で、簡易な $\Delta\Sigma$ 変調による可聴帯域の残留ノイズや、電源構造由来の残留ノイズなどの課題も残っています。本連載では、これらの課題を解決してラズパイPico DACを進化させる方法を紹介します。(編集部)

第4回(2022年3月号)では、ラズパイPico DACのハイレゾ対応「戦略」を立案しました。対応するオーディオ・フォーマットは表1に示す10フォーマットとしました。これを3軸(ビット深度軸、周波数倍率軸、周波数系列軸)に分類して、図1の要所で3軸の場合分け処理を行ない、既存の2.8224MHz/3.072MHz音源に統合して $\Delta\Sigma$ ~PWM出力処理を行います。また、ハードウェアもハイレゾ再生用に帯域拡張します。

今回は、具体的な「戦術」にあたる、ソフトウェアおよびハードウェアの実装・改造のポイントを記載し、最後に特性評価を行います。なお、図1では、第4回の図12から「FIR f特補正」の位置をCore1からCore0に移動しています。この理由は後述します(ソフトウェア実装のステップ5)。

1 ソフトウェア実装

ここでは、ソフトウェアの実装、および改造ポイントを解説します。ソースコード類は、本文中での説明用に一部改変や省略を行っている場合があります。

● ステップ1：USB各フォーマット入力対応

表1に示す10フォーマットに合わせて、USBホストに対応フォーマットを通知するようにUSBディスクリプタを拡張します。

図2に示すのは、USBディスクリプタの追加と変更点です。既存のAudio Streaming Interface Descriptorには、再生停止用のAlternate 0と、16ビット再生用のAlternate 1が用意されています。図2(a)に示すように、オーディオの停止と再生に合わせて、Alternate 0とAlternate 1が状態遷移するようになっています。

▶ 16ビット再生用…Alternate 1の変更

第1回(2021年11月号)で説明した44.1kHz/48kHz両対応のときに、16ビット再生用のAlternate 1に44.1kHz/48kHzの2つを列挙しました。今回は新たに88.2kHz/96kHz/176.4kHz/192kHzを追記します。

wMaxPacketSizeには、ISOF(Start of frame, 1kHz)単位で受信し得る最大パケット・サイズ[バイト]を記述します。従来は16ビット/48kHzのときが最大でしたが、今回は16ビット/192kHzのときが最大になるので、次のように変更します。式中の+1は、アシンクロナス転送における周波数可変対応分です。

- 変更前：

$$2[\text{Ch}] \times 2[\text{バイト/Ch}] \times \{48\text{k}[\text{Hz}]/1\text{k}[\text{Hz/SOF}] + 1\}$$

$$= 196 \text{ バイト/SOF}$$
- 変更後：

$$2[\text{Ch}] \times 2[\text{バイト/Ch}] \times \{192\text{k}[\text{Hz}]/1\text{k}[\text{Hz/SOF}] + 1\}$$

$$= 772 \text{ バイト/SOF}$$

▶ 24ビット再生用…Alternate 2の追加

Alternate 1は16ビット再生専用で定義されているので、ここに24ビット再生フォーマットを追記することはできません。