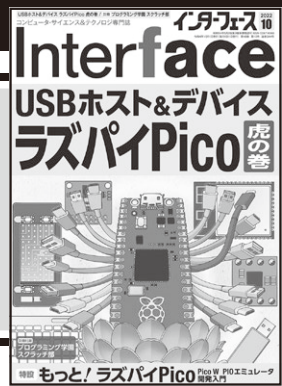


ご購入はこちら

# ラズパイ Pico の USB 活用



## 第1回 USBベンダ・クラスを使ったUSB-I<sup>2</sup>Cブリッジ

関本 健太郎

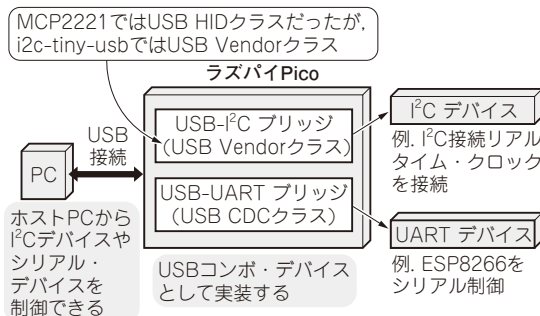
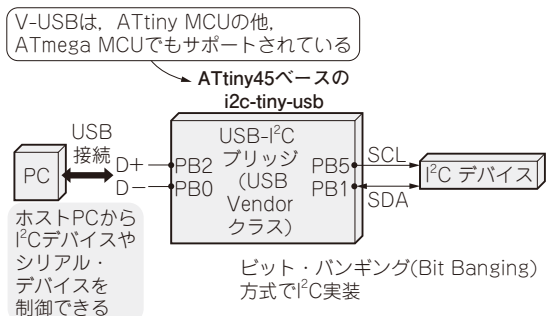
図1 i2c-tiny-usbベースのUSB-I<sup>2</sup>C/UARTブリッジ概要

図2 ATtiny45ベースのi2c-tiny-usb

2022年10月号特集は「USBホスト&デバイス ラズパイPico 虎の巻」でした。特集では、TinyUSBのサンプル・プログラムを詳しく解説したり、USBホスト、USBデバイスの製作事例を紹介したりしました。その中で、USB-I<sup>2</sup>Cブリッジについて、次のように解説しました。I<sup>2</sup>Cブリッジ製品には、

1. USB HIDクラスを利用したもの(10月号特集第3部第1章)
2. USBベンダ・クラスを利用したもの(今回)
3. USB CDCクラスを利用したもの(次回)

があります。10月号では、USB-I<sup>2</sup>Cブリッジの実現にUSB HIDクラスを用いましたが、USBベンダ・クラスを利用する方法もあります。USBベンダ・クラスを利用したものに、i2c-tiny-usbというプロジェクト(<https://github.com/harbaum/i2c-tiny-usb>)があります。

i2c-tiny-usbプロジェクトは、オープンソース/オープンハードウェア・プロジェクトです。一昔前にホストPCのプリンタ・ポートのピンを利用してデジタル入出力を実装していたユースケースの代替を目的に、USBポートを利用してI<sup>2</sup>Cデバイスを制御することを目的としています。

i2c-tiny-usbはLinuxの標準カーネル・モジュールに含まれています。本記事では、TinyUSBのベンダ・クラスの実装例を参考にi2c-tiny-usbをPicoで実装していきます(図1)。

## ハードウェア

i2c-tiny-usbインターフェースのハードウェアは、安価で入手しやすいATtiny45(マイクロチップ・テクノロジー、20MHz、フラッシュ:4Kバイト、RAM:256バイト)で構成されています。i2c-tiny-usbインターフェースのUSBインターフェースは、ソフトウェアで実装されており、AVRの2つのピン(PB0とPB2)を使用します(図2、図3)。

このソフトウェア実装は、ロースピードUSBのみで動作します。I<sup>2</sup>Cインターフェースは、v-usbと呼ばれており、ビット・バンギング・アプローチ(マイコンの汎用I/Oポートでソフトウェアのみで実装することをこう呼ぶ)を使って実装されています。

ATtiny45のハードウェアでサポートされているI<sup>2</sup>C(twi)インターフェースは、USB動作に必要なチップのハードウェア・ピンにバインドされているため、I<sup>2</sup>Cには使っていません。代替用ビット・バンギングI<sup>2</sup>Cインターフェースは、完全にI<sup>2</sup>C互換ではない可能性があるため、全てのI<sup>2</sup>Cデバイスがこのバスで正しく機能することは保証していません。

i2c-tiny-usbは、ソフトウェアで調整可能なI<sup>2</sup>Cクロック遅延を提供し、I<sup>2</sup>Cクロックを構成できるようにします。デフォルトの遅延は10 $\mu$ sです。I<sup>2</sup>Cビット・バンギング・コードの追加の遅延により、これは