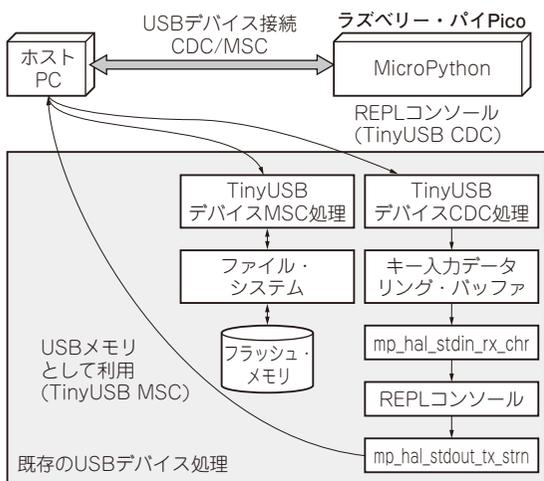
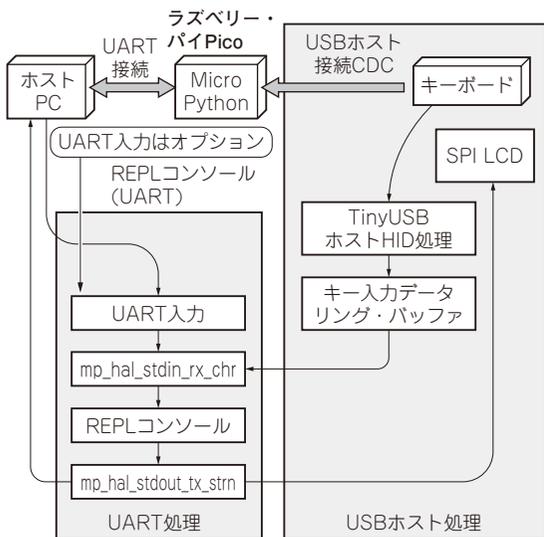


# HIDクラスを使ったMicroPython REPLコンソール

関本 健太郎



(a) 既存のUSBデバイス接続環境



(b) MicroPythonにUSBキーボードおよびSPI LCDを接続した環境

図1 2つのUSBデバイス接続環境

### ● マイコンでも動かせるPython「MicroPython」

MicroPython<sup>(1)</sup>は、メモリや動作速度が十分でないマイコンに最適化されている言語処理系です。Python3をベースに効率的な実装とし、Python標準ライブラリのサブセットを含んでいます。MicroPython

はインタラクティブ・プロンプト (REPL: Read-Eval-Print Loop)、任意精度の整数、クロージャ、リスト内包表記、ジェネレータ、例外処理などの高度な機能をサポートしています。それでもわずか256Kバイトのコード・スペースと、16KバイトのRAMに収まり、実行できるほどコンパクトです。

### ● USBデバイスのCDCおよびMSCクラスをサポートする

ラズベリー・パイ Pico (以降、Pico) の登場とともに、MicroPythonはPicoプログラム開発環境の1つとして、RP2040マイコンをサポートしています。USBデバイス機能は、Pico SDK環境と同じようにTinyUSBライブラリが利用されています。ただし、Pico SDKのGitサブモジュールとして含まれるTinyUSBではなく、それとは別にMicroPythonのGitサブモジュールとして利用されています。

2022年5月時点では、MicroPythonのRP2040マイコン向けのサポートは、USBデバイス機能をTinyUSBのCDCおよびMSCクラスのコンポ・デバイスとして実装しています。

### ● 本章でトライすること

本章では、USBデバイス接続の代わりにTinyUSBのUSBホスト機能のHIDクラスの機能を利用します。そしてMicroPythonのREPLコンソール処理にUSBキーボードを接続できるように機能拡張します。また、表示デバイスとして、シリアル接続の小型液晶ディスプレイを利用します(図1)。ターゲットの環境として、PicoにSPI LCDを接続した環境(写真1)と、Pico Display Pack<sup>(2)</sup>(Pimoroni)を接続した環境(写真2)を取り上げます。

## MicroPythonがサポートする内部アーキテクチャ

MicroPythonがサポートするPicoの内部アーキテクチャ(図2)は、大きく分けると以下です。

- マイコンのブート機能
- MicroPythonのコンパイラ機能
- バイト・コードを実行するMicroPythonの仮想マシン