

# I<sup>2</sup>C & SPI をゼロから作る③

## …I<sup>2</sup>C 送受信機能の実装

岡野 彰文

### ステップ1…MicroPythonでGPIO入出力実験

I<sup>2</sup>Cは2線式のインターフェースなので2本のGPIOピンを使い、MicroPythonでI<sup>2</sup>C信号を作ってみます。

信号の出力はこのピンに順次“H”や“L”の出力を行うことでクロックやデータを出すことができます。

データを入力するにはGPIOを入力に切り替えて“H”または“L”の値を読み取ります。

#### ● Picoのピン入出力

##### ▶ Picoでオープン・ドレインに設定する方法

I<sup>2</sup>C出力はオープン・ドレインなので、PicoのGPIOを次のように制御します。

“L”を出力する際にはピンを“L”にドライブすればよいのですが、“H”の際にはハイ・インピーダンス状態にする必要があります。

このため“L”出力は、あらかじめ“L”を出すようにした出力設定で行い、“H”出力はGPIOピンを入力に切り替えることで「オープン・ドレイン出力の“H”」の代用とします。

ハイ・インピーダンス状態になるとプルアップ抵抗の作用によって信号が“H”となります。

##### ▶ プログラム例

実際のプログラムをリスト1に示します。MicroPythonではその組み込みライブラリのmachineに含まれているPinクラスでGPIOピンを制御しま

リスト1 MicroPythonによるピン出力制御

```
from machine import Pin
# machineライブラリからPinクラスをインポート
p = Pin(0) # ピン0をpとして指定
p.value(False) # p(ピン0)を出力に0を出すよう設定

v = False # vをFalse(=0)で初期化

while (True): # 次のブロックをループ
    if v: # もしvがTrue(=1)なら
        p.init(Pin.IN) # p(ピン0)を入力設定に
    else: # もしvがFalse(=0)なら
        p.init(Pin.OUT) # p(ピン0)を出力設定に
    v = not v # vの値を反転
```

す。このコードはGPIO0を入力・出力の切り替えてトグルする例です。このピン出力にはあらかじめFalse(=0)を設定しておいて、変数vの値によってTrueなら入力設定に、Falseなら出力設定に切り替えています。このピンには外部にプルアップ抵抗が付いているため、入力設定に切り替えると“H”に変化します。

出力結果は図1のようになります。

### ステップ2…MicroPythonで基本実装部分を作る

I<sup>2</sup>Cプロトコルをビット・バンギングで作ってみたい。

ビット・バンギングを行うプログラムbbI2C.pyは本誌ウェブ・ページからダウンロードできます。

<https://www.cqpub.co.jp/interface/download/contents.htm>

#### ● 初期設定

bbI2Cクラスではクラスのイニシャライザ(`__init__()`)、図2で2つのGPIOピン(のインスタンス)が渡され、初期化します。

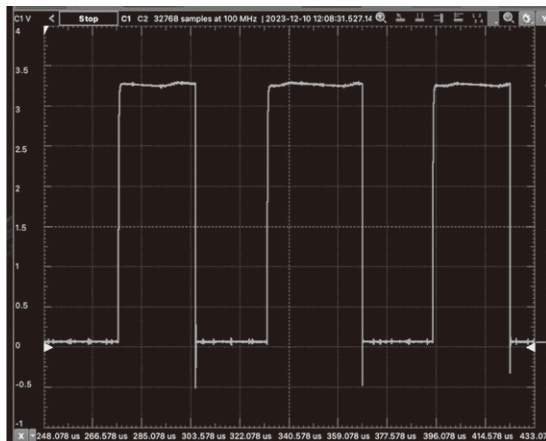


図1 GPIOピン0の出力の様子