

第2章

シリアル通信/PWM/ウォッチドッグ・タイマなど

GPIO活用リファレンス

宮田 賢一

本稿では、ラズベリー・パイ Pico (以降、Pico) の各種ペリフェラルを MicroPython でプログラミングする方法を紹介します。

GPIO や PWM (Pulse Width Modulation), A-D コンバータなど、マイコンでよく使う機能は、モジュー

ルが用意されているので簡単に利用できます。

プロトタイプ開発などですぐ機能を実現したい場合に便利です。

表1 ペリフェラル用のクラスとメソッドの一覧(ソフトウェア実装の SoftI2C と SoftSPI は省略)

クラス	メソッド	動作	クラス	メソッド	動作	
Pin	Pin	Pinオブジェクトを生成するコンストラクタ	I2C	readfrom_mem_into	メモリ・デバイスのスレーブからバッファ・サイズを読み込む	
	value	ピンの値を取得または設定する		scan	I2Cアドレスをスキャンする	
	high	ピンの値を1にする		writeto	スレーブにバッファ・サイズを書き込む	
	init	ピンを再初期化する		writeto_mem	メモリ・デバイスのスレーブに書き込む	
	irq	ピンに割り込みハンドラを設定する		writevto	スレーブにバッファ・プロトコルをもつオブジェクトから書き込む	
	low	ピンの値を0にする		SPI	コンストラクタ	
	off	ピンの値を0にする		read	指定したバイト数を読み込む	
	on	ピンの値を1にする	readinto	指定したバッファ・サイズを読み込む		
	toggle	ピンの値をトグルする	write	指定したバッファを書き込む		
ADC	ADC	A-Dコンバータ・オブジェクトを生成するコンストラクタ	SPI	deinit	バスをOFFにする	
	read_u16	アナログ値を読み取る		init	バスを初期化する	
UART	UART	コンストラクタ		write_readinto	リード・バッファに読み込みながらライト・バッファを書き込む	
	any	ブロックせずに読み込めるバイト数を返す		PWM	PWM	PWMオブジェクトを生成するコンストラクタ
	read	指定したバイト数を最大で読み込む			deinit	PWMを無効化する
	readinto	指定したバッファ・サイズを最大で読み込む	duty_ns		デューティをns単位で設定する	
	readline	1行を読み込む	duty_u16		デューティをカウンタ値単位で設定する	
	write	バッファを書き込む	freq	PWMの周波数を取得または設定する		
	sendbreak	ブレイク信号を送信する	WDT	WDT	コンストラクタ	
I2C	I2C	I2Cオブジェクトを生成するコンストラクタ		feed	タイマにフィードする	
	readinto	バスからbytes型オブジェクトを読み込みACK確認を行う	Timer	Timer	コンストラクタ	
	start	バスをスタート・コンディションにする		deinit	タイマを無効化する	
	stop	バスをストップ・コンディションにする	init	タイマを初期化する		
	write	バスにbytes型オブジェクトを書き込みACK確認を行う	Signal	Signal	Signalオブジェクトを生成するコンストラクタ	
	init	バスを初期化する		value	信号レベルを取得または設定する	
	readfrom	スレーブから指定バイト数を読み込む		off	信号をOFFにする	
	readfrom_into	スレーブからバッファ・サイズを読み込む		on	信号をONにする	
readfrom_mem	メモリ・デバイスのスレーブから指定バイト数を読み込む					