

C言語版デバイス制御の 潜在的バグを続々発見

高野 祐輝

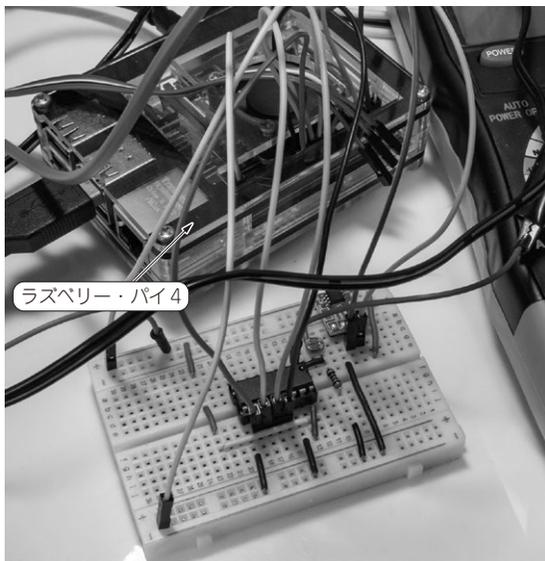


写真1 ラズベリー・パイ4と各種センサで構成した環境モニタ
センサとの接続にはI²CやSPIを利用している

本章では、Rustとラズベリー・パイ4を用いて、温度と明るさを取得する簡単なソフトウェアを実装します。I²Cインターフェースを持つ温度センサADT7410(アナログ・デバイセズ)と、明るさセンサとしてアナログ値を出力するCdSセルを利用します。アナログ値はA-DコンバータMPC3208(マイクロチップ・テクノロジー)を用いて、SPIインターフェースでマイコンに取り込みます(写真1)。

I²CとSPIを利用するためには、`rppal`というラズベリー・パイ用のライブラリを利用します。

また、Rustでの実装に加え、C言語での実装も示し、Rustとの比較を行います。C言語のプログラムでは`WiringPi`というライブラリを利用します。この過程において、C言語版に潜んでいたバグがRustでビルド時に発見できる例を見ていきます。

さらに、Rustを用いたI²CやSPIデバイスの扱い方と、Rust特有の抽象化方法についても説明します。

本章で利用するOSは、Raspberry Pi OS 11.6 (Linux

3V3	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3V3	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7
ID_SD	27	28	ID_SC
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21

図1 ラズベリー・パイ4のピン配置

5.15.84-v8+)となります。

ステップ1…ハードウェアの製作

図1は、ラズベリー・パイ4のピン配置です。GPIOピンに対して、信号の入力と出力を行います。

● 温度センサADT7410

▶ I²Cに対応

温度センサADT7410はI²C(Inter-Integrated Circuit)に対応しています。I²Cには、クロック同期SCLと、データ転送SDA、電源源である V_{DD} とGND線が必要です。

I²CやSPIは、現在ではデバイス・ドライバが用意されていることがほとんどで、ライブラリを利用すると利用可能です。

▶ ラズベリー・パイ4との接続

図2は、ADT7410のピン配置とラズベリー・パイ4

