ダウンロード・データあります

組み込みRustのライブラリ 便利クレート探偵団

第12回 組み込みRustの共通インターフェース embedded-hal v1.0.0を使ってみる

中林 智之

Rustは，組み込みで使えるプログラミング言語として注目されています．本連載ではそんなRustの組み込み開発で役立つライブラリ（クレート）を紹介します．

今回は実際にembedded-hal v1.0.0のトレイトを使って，マイコンの各種ペリフェラルにアクセスするAPIを解説します．特にAPIのうちブロッキング（同期）APIについて紹介します．

● ブロッキングAPI＝処理完了まで戻ってこないAPI

ブロッキングAPIとは，ある処理（例えばSPIで1バイト送る）が完了するまでその関数呼び出しが戻ってこない（スレッドやタスクが止まる）というシンプルなインターフェースです．実行順序が明確で分かりやすいため，まずはこのブロッキングAPIから試すのがお勧めです．

● さまざまなボードで試せる

ターゲット・ボードにはnRF52840-DK（ノルディック・セミコンダクター）を使います．これ以外のボードでも，初期化以外のHALトレイトの使い方は同じなので，ぜひお手持ちのボードで試してみてください．

本稿ではembedded-hal v1.0.0で定義されているdigital（GPIO），delayとエラーの扱いを紹介します．同じくembedded-hal v1.0.0で定義されているpwm，i2c，spiについては次回紹介します．

GPIOを扱うdigitalモジュール

digitalモジュールでは，マイコンのGPIOピンに対する入出力操作を定義しています．組み込み開発において，LEDの点灯やボタン入力の検知など，最も基本的で頻出する処理をカバーするトレイト群です．

embedded-hal v1.0.0では次のようなトレイトが定義されています．

- InputPin：GPIO入力ピンを表現するトレイト
- OutputPin：GPIO出力ピンを表現するトレイト
- Error：上記トレイトで使われる共通のエラー型

出力ピンならset_high()やset_low()，入力ピンならis_high()やis_low()を呼び出すことで，抽象化されたピン操作が可能です．

リスト1　入力に使うInputPinトレイト

```
pub trait InputPin: ErrorType {
    fn is_high(&self) -> Result<bool, Self::Error>;
    fn is_low(&self) -> Result<bool, Self::Error>;
}
```

● トレイト

エラー用のErrorトレイトを除くと3つの主なトレイトが定義されています．

▶入力に使うInputPin

InputPinはGPIOピンが入力モードのときに使うトレイトです．ピンの状態が“H”か“L”かを判定するメソッドを提供します（**リスト1**）．

- is_high()：ピンが“H”ならOk（true）を返す
- is_low()：ピンが“L”ならOk（true）を返す

例えば，ボタンが押されているかどうかのチェックに利用します．ErrorTypeトレイトを継承していますが，こちらの詳細は後ほど解説します．

▶出力に使うOutputPin

OutputPinはGPIOピンが出力モードのときに使うトレイトです．ピンを“H”または“L”に設定するためのメソッドを提供します（**リスト2**）．

- set_high()：ピンを“H”に設定する
- set_low()：ピンを“L”に設定する
- set_state()：ピンを引数で渡す状態に設定する

LEDの点灯／消灯や，ディジタル出力を使った制御などに使います．

▶出力状態を読み取るStatefulOutputPin

StatefulOutputPinは出力状態を読み取れるトレイトです．例えば「現在LEDが点灯中かどうか」を知りたい場合に使います（**リスト3**）．

OutputPinトレイトを継承しているため，

ダウンロードはこちらから．
https://github.com/tomoyuki-nakabayashi/interface-embedded-rust-crates/tree/main