

組み込みRustのライブラリ 便利クレート探偵団



第11回

組み込みRustの共通インターフェース
embedded-hal v1.0.0

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。

組み込み開発の決定版！待望の embedded-hal v1.0.0登場

組み込みRustを使った開発で、マイコン固有の違いを意識せずに汎用的なドライバを書くための共通インターフェースとして、長らく中心的な役割を担ってきたクレートがembedded-halです。2023年12月末に待望のv1.0.0がリリースされました。

v1.0.0リリース前のバージョンはv0.2で、事実上の共通インターフェースだったとはいえ、APIの安定性や方向性に不安がありました。

embedded-halがv1.0.0になったことで、破壊的変更、つまり既存のコードがそのままでは動かなくなるような変更が加えられる心配が大きく減りました。v1.0.0のリリースは長期的な互換性を提供することを目的としています。

今回から複数回にわたって、embedded-hal v1.0.0を詳しく解説します。

embedded-halの提供する インターフェース

● 再利用性の高いドライバが書ける

embedded-hal⁽¹⁾はデバイス・ドライバを共通のインターフェースで実装するための最小限の抽象インターフェースを提供するクレートです。この最小限な設計により、特定のハードウェアに依存しない汎用的で再利用性の高いドライバが書けるのが最大の強みです。

その結果、embedded-halのインターフェースで実装されたドライバは、ベアメタル(OSなし)のマイコンだけにとどまらず、リアルタイムOSやLinux上でも動作します。

この理由から今後Rustで組み込み開発をするときは、(よほどの理由がない限り)embedded-halを使うことになるでしょう。

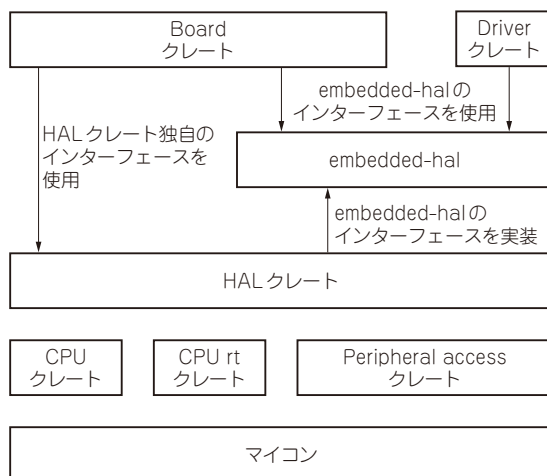


図1 embedded-hal と対応クレートの構成

HAL (Hardware Abstraction Layer) はハードウェア抽象化レイヤのことで、ハードウェアの違いを吸収するためのソフトウェア・インターフェースを提供します。

- OutputPin やI2cなどのトレイトを通じて、GPIO、I²C、SPI、PWMなどの基本的な周辺機能に対して抽象的なアクセスを可能にする
- HALの実装はマイコン固有のHALクレートに任されており、上位のドライバ・クレートはHALのトレイトに対してのみ依存すれば、特定のマイコンに縛られない設計ができる

● クレートの構成

embedded-halと対応クレートの構成は図1のようになります。driverクレートが全てのHALクレートで汎用的に動くようにするのが目的としています。

例えばリスト1はGPIOに接続したボタンとLEDを使って、ボタンを押している間LEDを点灯するプログラムです。このプログラムはembedded-halのトレイト(InputPin/OutputPin)を使ってGPIOを操作しています。そのため、embedded-halに対応したHAL