

4時間目

My ドローンで使うために
全体像を知っておきたい

オープンソースのフライト・コード ArduPilot



春原 久徳

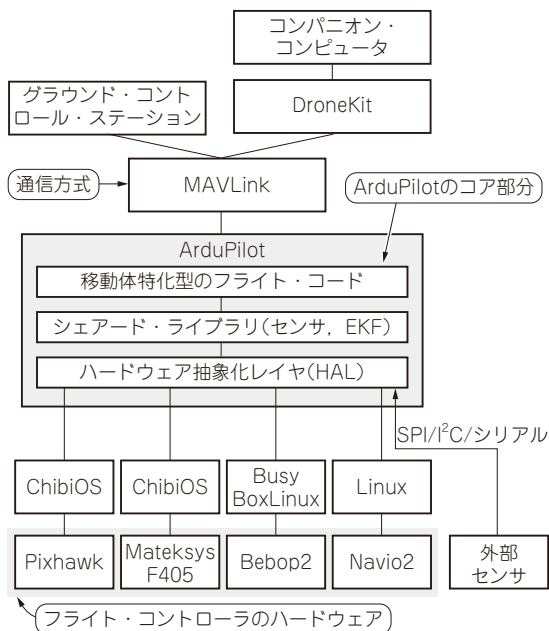


図1 ArduPilotのソフトウェア・アーキテクチャ
広義のArduPilotにはコンパニオン・コンピュータで使う通信ライブラリ(MAVLink)などのソフトウェアも含まれる

オープンソースのフライト・コード ArduPilot (アルデュパイロット) は、いろいろな移動体の制御を考慮して開発されており、独自の機体の制御にも使えます。マルチコプタなどの制御のコア部分については既に完成度が高く、独自の機体で利用する場合にもパラメータの調整だけで利用できます。

ここでは、ドローンなどのシステムの中で ArduPilot が受け持つ役割を把握するために、制御の流れと、フライト・コントローラとそれ以外のデバイスとの関係を解説します。(編集部)

こんなソフトウェア

- いろいろな機体に対応できるアーキテクチャ
図1の下方部分がフライト・コントローラなどの

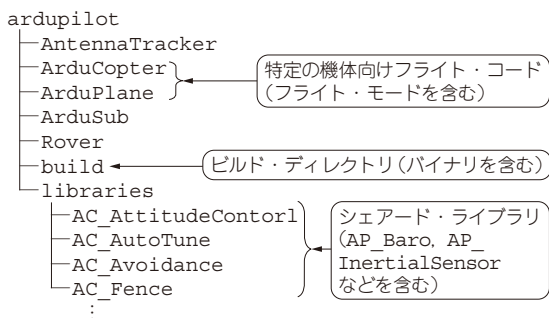


図2 ソースコードのフォルダ構造

ハードウェア部です。各フライト・コントローラに適切な OS が搭載された上で (現在は ChibiOS が標準 OS となっている), ArduPilot が動作します。狭義の意味での ArduPilot は移動体特化型のフライト・コード (Vehicle specific flight code) と呼ばれており、次の要素で構成されています。

- 機体の種類 (固定翼機, 回転翼機, ローバ, 潜水艇など) に応じたフライト・コード
- 各種センサに対応したシェアード・ライブラリ
- フライト・コントローラの違いに対応するための抽象化レイヤ (HAL)

フライト・コントローラの外部とは MAVLink と呼ばれるコミュニケーション・レイヤを通じて通信します。通信相手としては次のようなものがあります。

- **グラウンド・コントロール・ステーション (GCS)**
ドローンの外部にあるコンピュータです。地上に設置された PC などが該当します。ドローンの管理アプリケーションなどを動かします。
- **コンパニオン・コンピュータ**

ドローンに搭載するコンピュータで、フライト・コントローラとは別のものです。DroneKit を使って開発されたソフトウェアを動かします。ハードウェアとしてはシングルボード・コンピュータなどが利用されます。