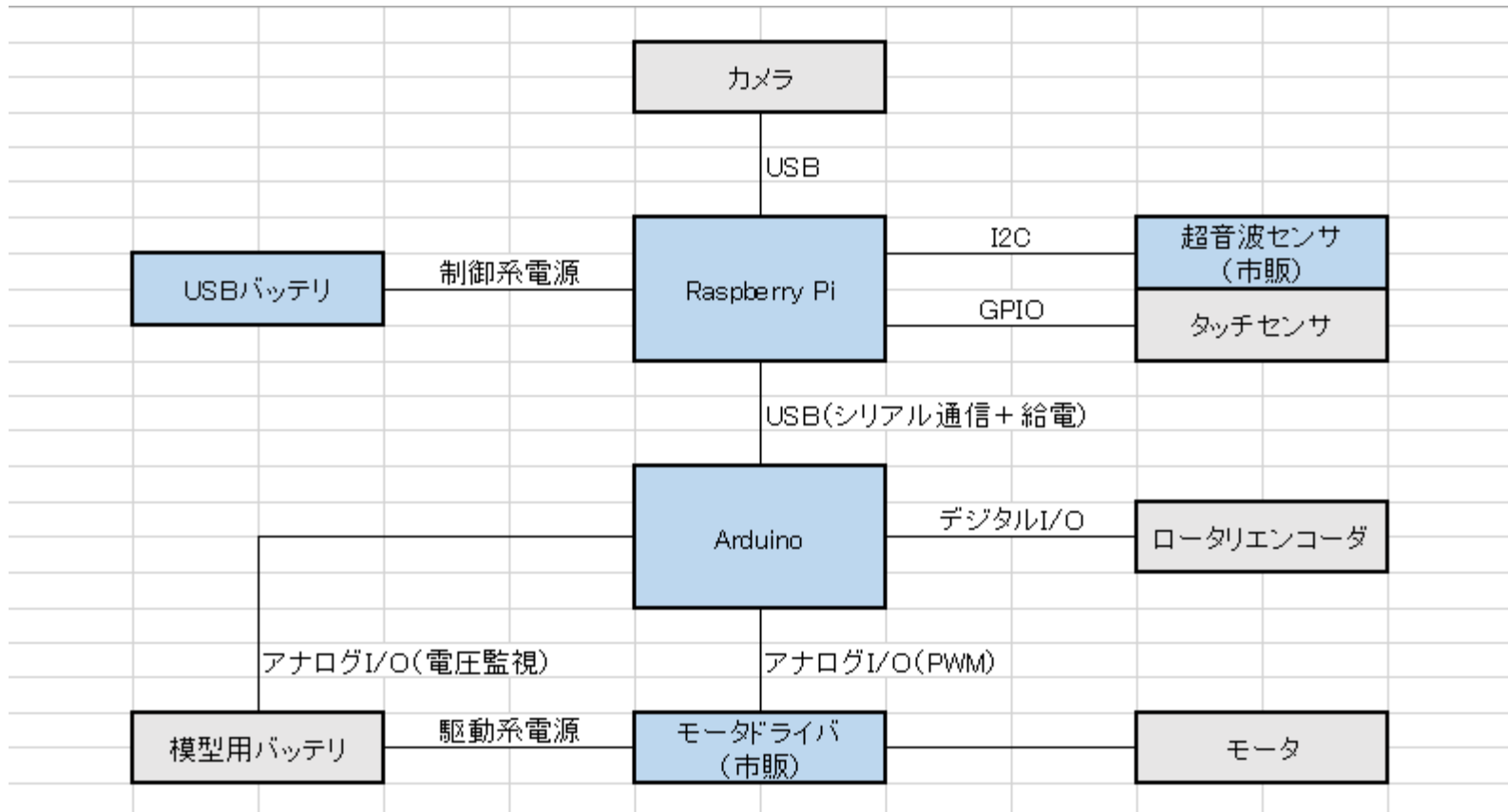


# MIRSソフトウェア解説

## MIRS標準機の開発環境と 標準プログラム

2022/5/13

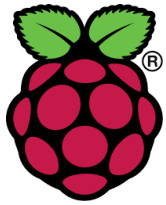
# MIRS標準機の構成



Arduino が走行制御を担い、それ以外の処理は Raspberry Pi が行っている。  
Raspberry Pi と Arduino はシリアル通信により命令・データを送受信する。

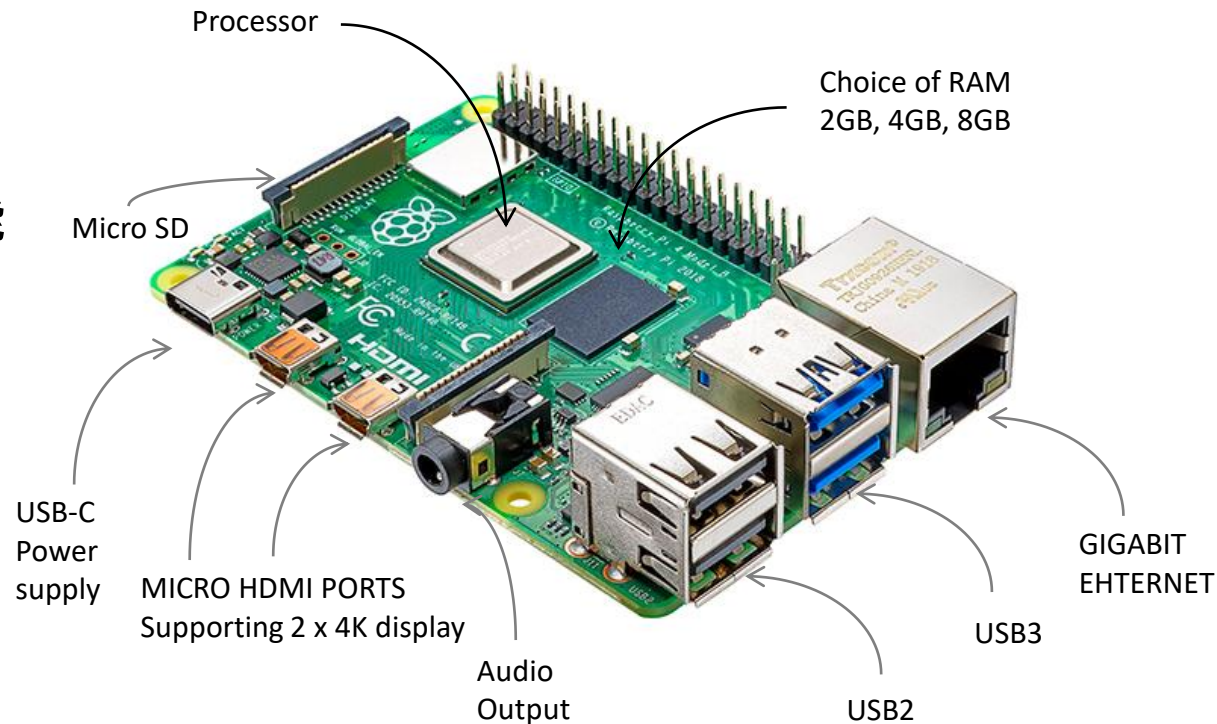
# MIRS標準機のソフトウェア開発

- MIRS標準機
  - Raspberry Pi と Arduino の2つのマイコンで制御系を構成している。
  - それぞれに標準プログラムが用意されている。
- 開発環境
  - Raspberry Pi
    - 開発言語: C言語
    - ソースコードはエディタがあれば可
    - GCC がインストールされていればコンパイルまで可
  - Arduino
    - 開発言語: Arduino言語(≒C言語)
    - Arduino IDE がインストールされているPC (Raspberry Pi含む)



# Raspberry Pi 4 (Model B)

- らずべりーぱい
- シングルボードコンピュータ
- PCと遜色ない機能・性能
  - CPU
    - 1.5GHz
    - クアッドコア
    - ARM Cortex-A72
  - メモリ 4GB
  - USB, HDMI, LAN
- 40pin GPIO



# Raspberry Pi の開発環境

- OS : Raspberry Pi OS
  - 2020年5月、[Raspbian](#) (らずびあん) から改名
  - Debian系 Linux を Raspberry Pi にカスタマイズしたもの
  - 32bit (8GBメモリ搭載用には64bitベータ版)
- ライブラリ
  - [Wiring Pi](#) : Raspberry Pi のGPIO、シリアル通信 (I2C、RS232C)
  - [OpenCV](#) : 画像処理
  - Pthread : マルチスレッド (マルチタスク)
- [標準プログラム](#)
  - 機能モジュールと機能試験のためのテストプログラム群で構成
  - C言語で記述 (9月に C++版をリリース予定)

# Arduino

- Arduino UNO
  - シングルボードマイコン
  - [Arduino 製品群](#)の中で最もスタンダード
  - 5V駆動（USB供給可）
  - IO
    - 信号レベル: 5V
    - アナログピン 5
    - デジタルピン 14
      - 割込み 2ピン
      - [PWM出力](#) 6ピン
  - CPU: ATmega328P
- 開発言語は Arduino 言語
- プログラム開発にはArduino IDE を用いる。



# Arduino の開発環境

- Arduino IDE
  - Arduino の統合開発環境
  - Windows, PC, Linux (Raspberry Pi OS 含む) 版あり
- Arduino 言語
  - C/C++をベースにしており、C言語のすべての構造と、いくつかのC++の機能をサポートしている。
  - [Arduino 日本語リファレンス](#)
  - 必須関数
    - loop関数
      - 名前のおり loop() 内の処理を繰り返す。
      - main関数の中の while(1) のようなもの
    - setup 関数
      - 起動時(loop関数実行前)に一度だけ実行する。
      - ピンモード(入出力、PWM、割込み)の設定など行う

# 標準ソフトウェア概要

- RaspberryPi にはC言語、Arduino には Arduino 言語でコーディングされた標準ソフトウェアを用意している。
- Raspberry Pi 用
  - MIRS2015の巡回警備の競技会に必要なモジュール群を用意している。
- Arduino 用
  - 直進・回転の走行制御を行う必要なモジュールと RaspberryPi との通信機能を用意している。
- それぞれに単体機能をテストするプログラムが用意している。



# MIRS標準ソフトウェアのドキュメント

- MIRSMG4D 管理台帳

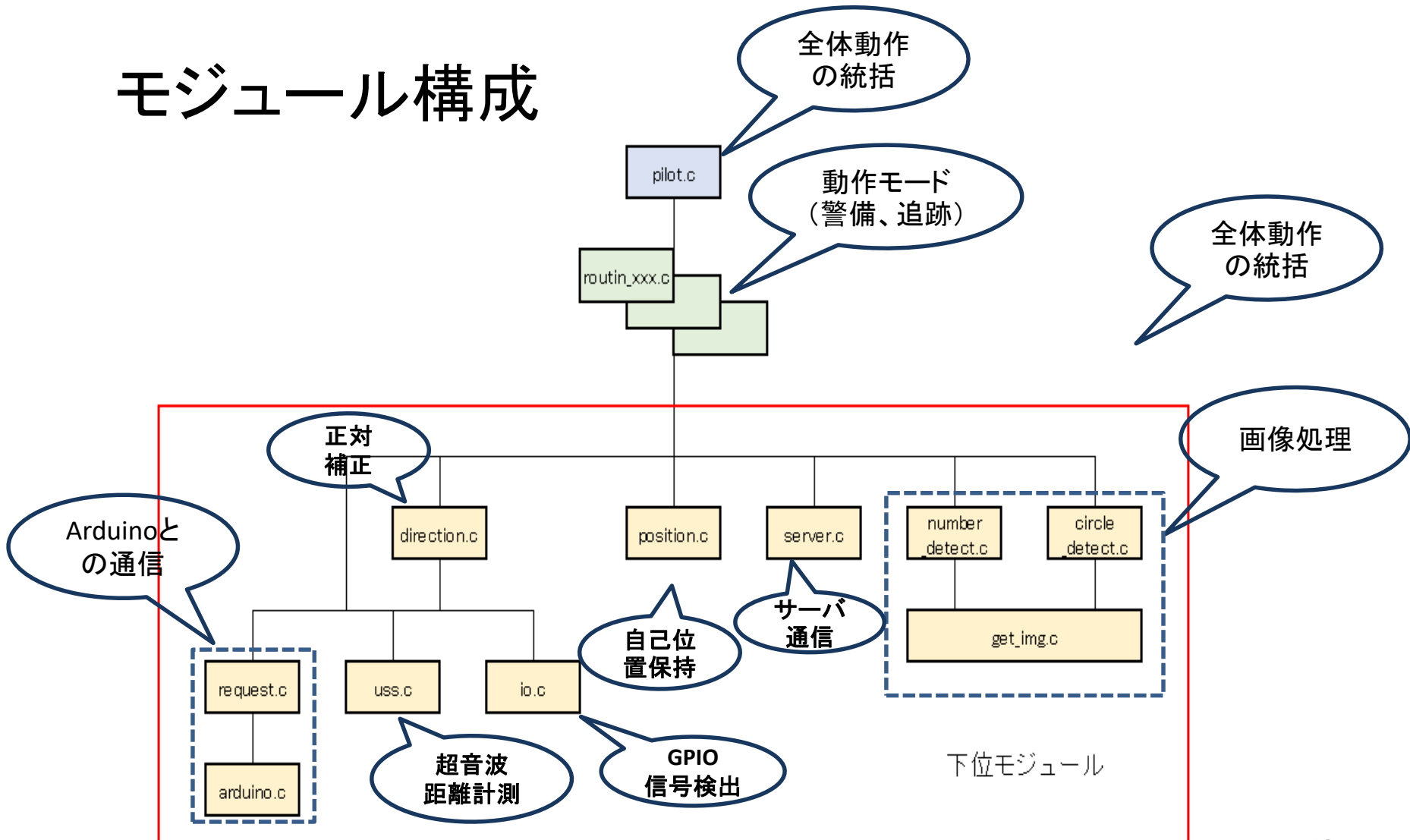


ソフトウェア

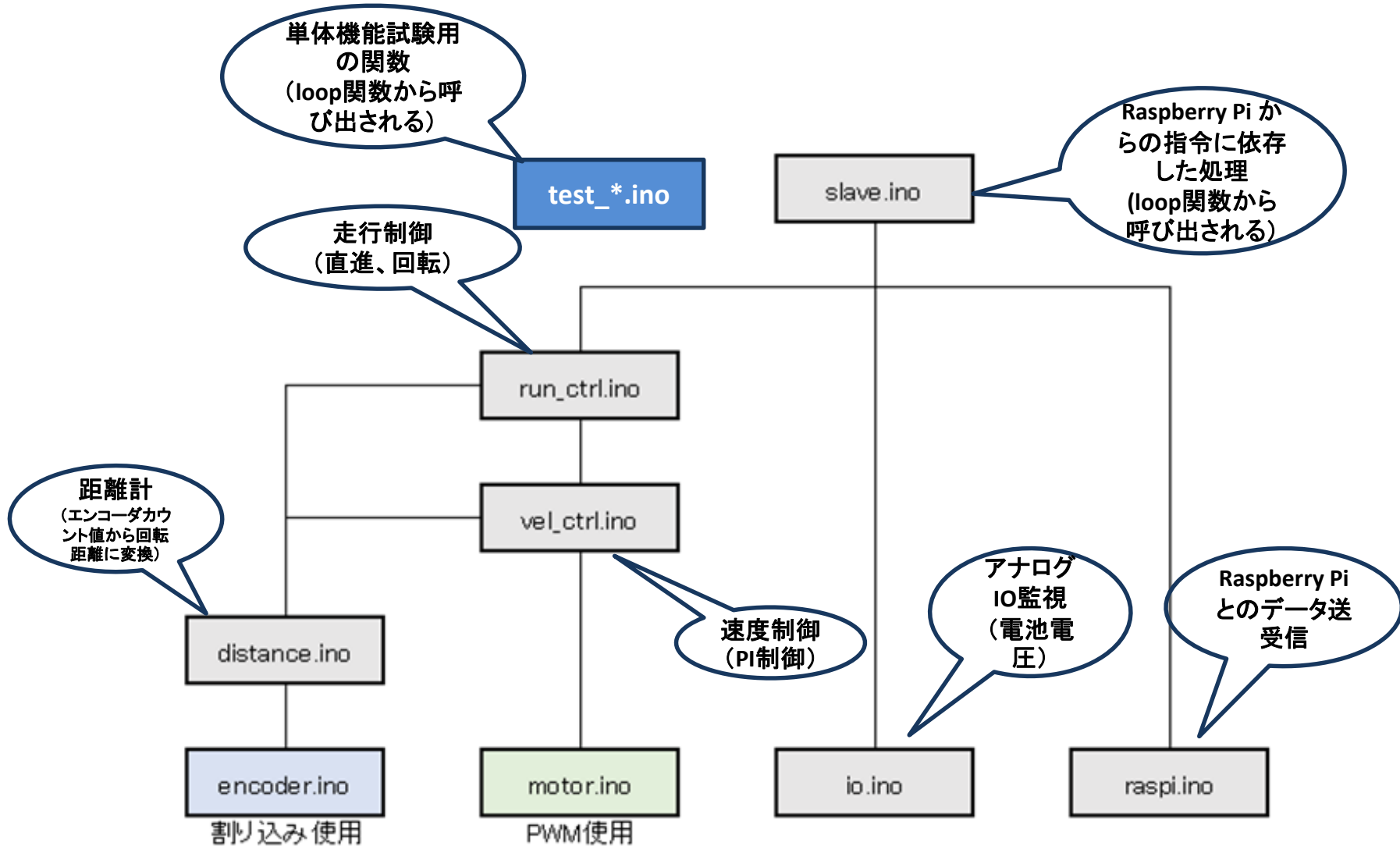
| ドキュメント番号                           | ドキュメント名称                    | 採番者  | 版数  | 作成者  | 登録日        | 備考        |
|------------------------------------|-----------------------------|------|-----|------|------------|-----------|
| <a href="#">MIRSMG4D-SOFT-0001</a> | Arduino ソフトウェア 解説           | 鈴木静男 | A01 | 牛丸真司 | 2017/04/19 | Ver 3.1   |
| <a href="#">MIRSMG4D-SOFT-0002</a> | Arduino ソフトウェア 関数レファレンス     | 鈴木静男 | A01 | 牛丸真司 | 2017/04/19 | Ver 3.1   |
| <a href="#">MIRSMG4D-SOFT-0003</a> | Arduino ソフトウェア ソースコード       | 鈴木静男 | A01 | 牛丸真司 | 2019/05/01 | Ver 3.1.3 |
| <a href="#">MIRSMG4D-SOFT-0004</a> | RaspberryPi ソフトウェア 解説       | 鈴木静男 | A01 | 牛丸真司 | 2017/04/19 | Ver 3.0   |
| <a href="#">MIRSMG4D-SOFT-0005</a> | RaspberryPi ソフトウェア 関数レファレンス | 鈴木静男 | A01 | 牛丸真司 | 2017/04/19 | Ver 3.0   |
| <a href="#">MIRSMG4D-SOFT-0006</a> | RaspberryPi ソフトウェア ソースコード   | 鈴木静男 | A01 | 牛丸真司 | 2019/05/01 | Ver 3.0.1 |
| <a href="#">MIRSMG4D-SOFT-0007</a> | ソケット通信によるプロセス間通信            | 牛丸真司 | A01 | 牛丸真司 | 2018/12/21 |           |

# 標準プログラム (Raspberry Pi)

## モジュール構成



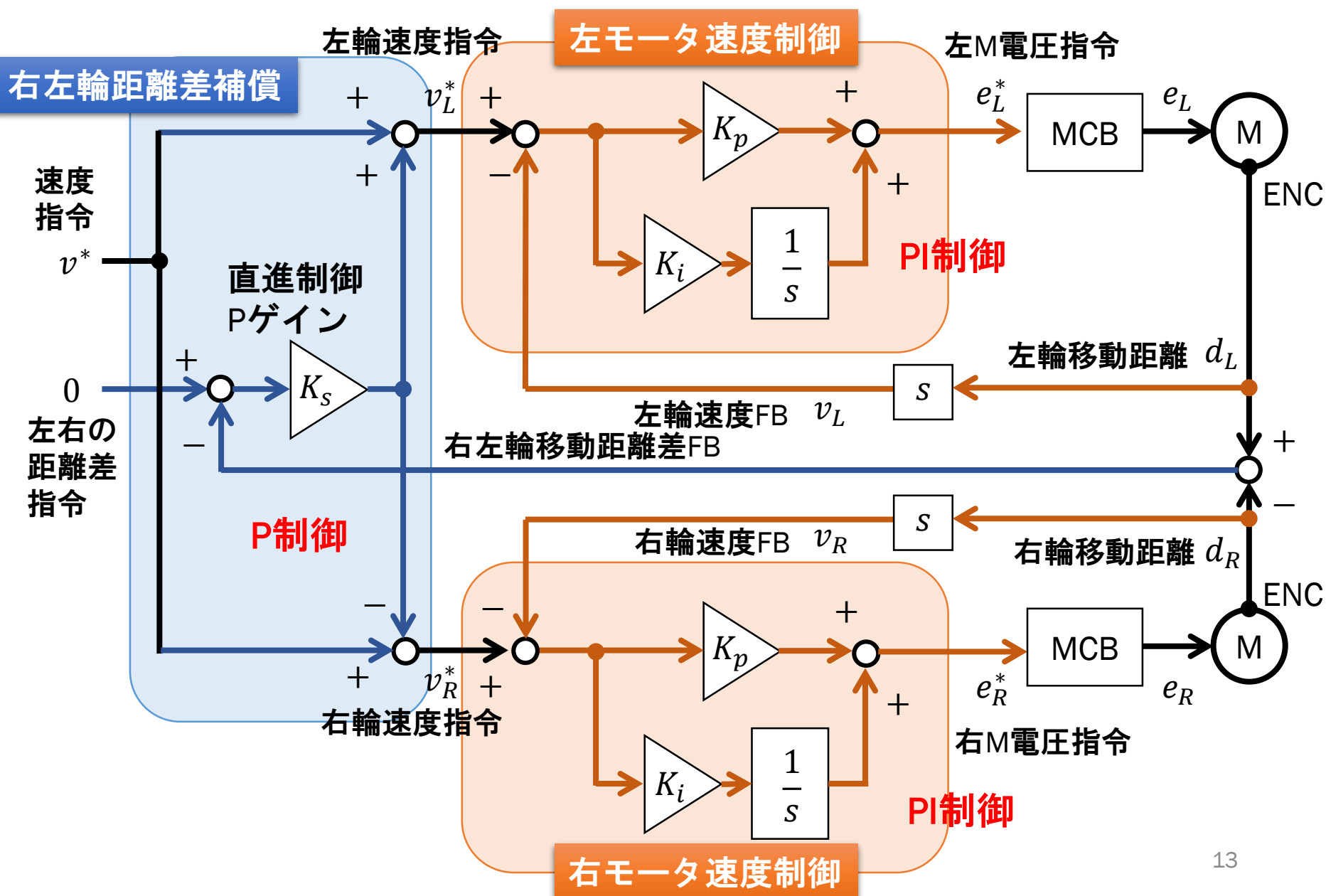
# 標準プログラム (Arduino) のモジュール構成



# 走行制御

- モータはPWM信号で実効電圧(=平均電圧)を変化させて制御する。(参考ページ)
- ロータリーエンコーダのA層信号を用いた割り込み処理により、エンコーダの矩形信号エッジのカウントを行う。
- ロータリーエンコーダのカウント値を利用して、左右のモータの速度制御をPI制御で行う。
- ロータリーエンコーダのカウント値を利用して、機体の直進・回転制御をPID(P/PD/PID)制御で行う。

# 直進制御ブロック線図



# ソフトウェア開発と試験

- 組み込みシステムの開発（MIRS開発を含む）では、ソフトとハードの並行開発が一般的
- ハードウェアの完成度に対応した試験の実施
  - 最終のシステム試験（統合試験）はハードウェアが完成してからでないと出来ないが、ハードウェアの完成度（や有無）に対応した試験が可能
  - コーディングは、ハードウェアがなくても可能（コーディングとコンパイルチェックは同レベル）

# ハードウェアの完成度に対応した試験

- 単体（機能）試験
  - その機能を確認するハードウェアのみがあれば可能
  - 実機にマウントされている必要はない
- ハードウェアの仮想化
  - 実際のハードウェアの別ハードウェアまたは事前に用意したデータで代用する。
  - 例 タッチセンサ押下 → キーボード入力  
カメラ画像 → 事前に用意した画像の読み込み  
通信 → 相手から受け取るデータを事前に用意
- ソフトウェアの仮想化
  - 例：画像認識処理 → データ（クラス）の読み込み

# 他言語での開発

- 最近のMIRSオリジナル開発では、画像処理系や通信系など一部（または全て）を Python や C++で記述するチームが増えている。
  - OpenCV 3. xのAPIは C言語に非対応（メンテナンス対象外）（C++, Python, Javaに対応）
  - 画像処理系は Python で書かれたソースコードのサンプルが多い。（特に DL 系）
- RaspberryPi の標準プログラム
  - 9月にC++版をリリース予定
  - 全て Python で書き換えたチームもあり（MIRS2101）