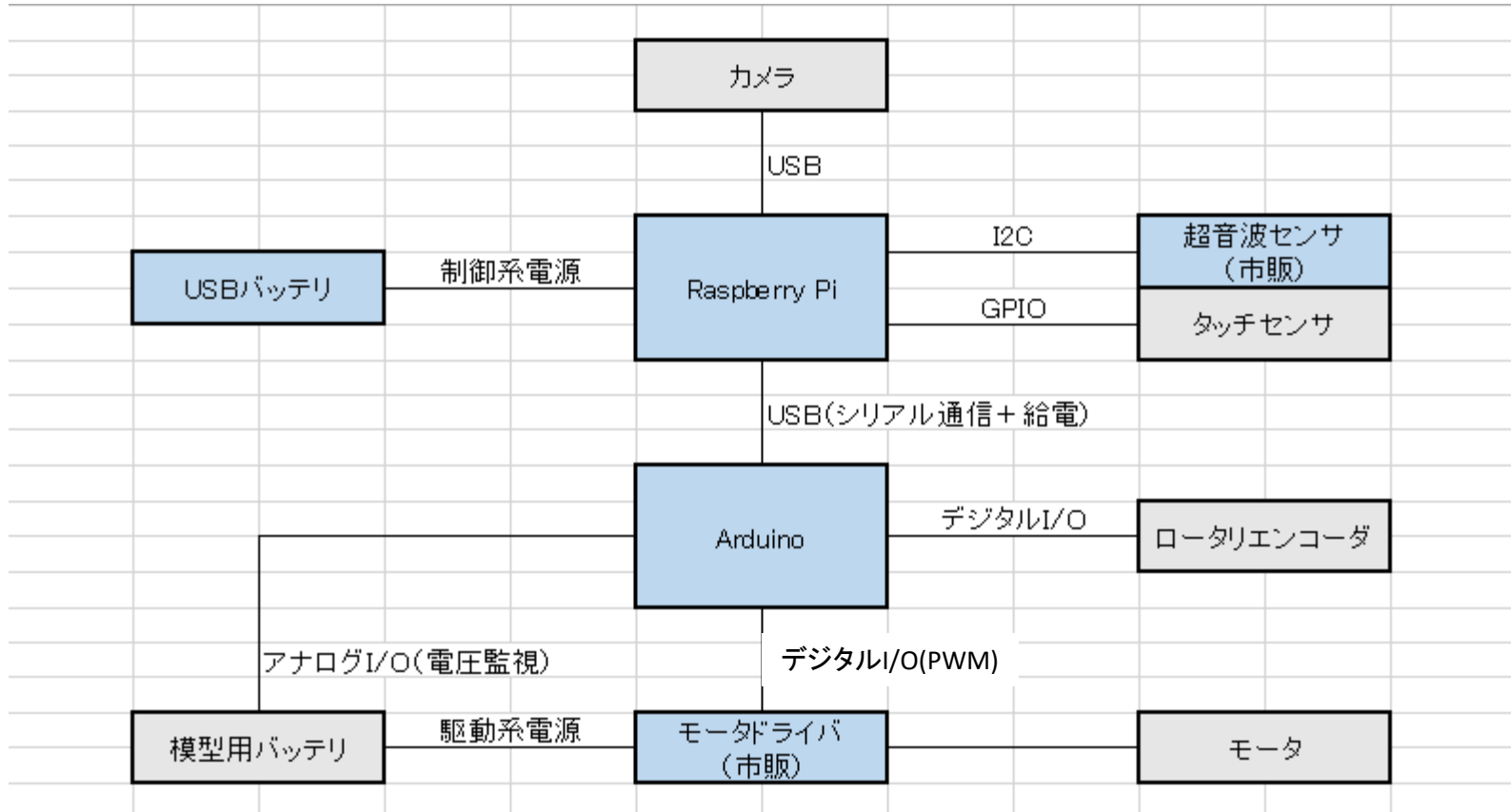


MIRS標準機 ソフトウェア解説

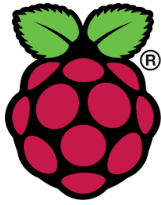
MG4編

2020/6/26

MIRS標準機の構成



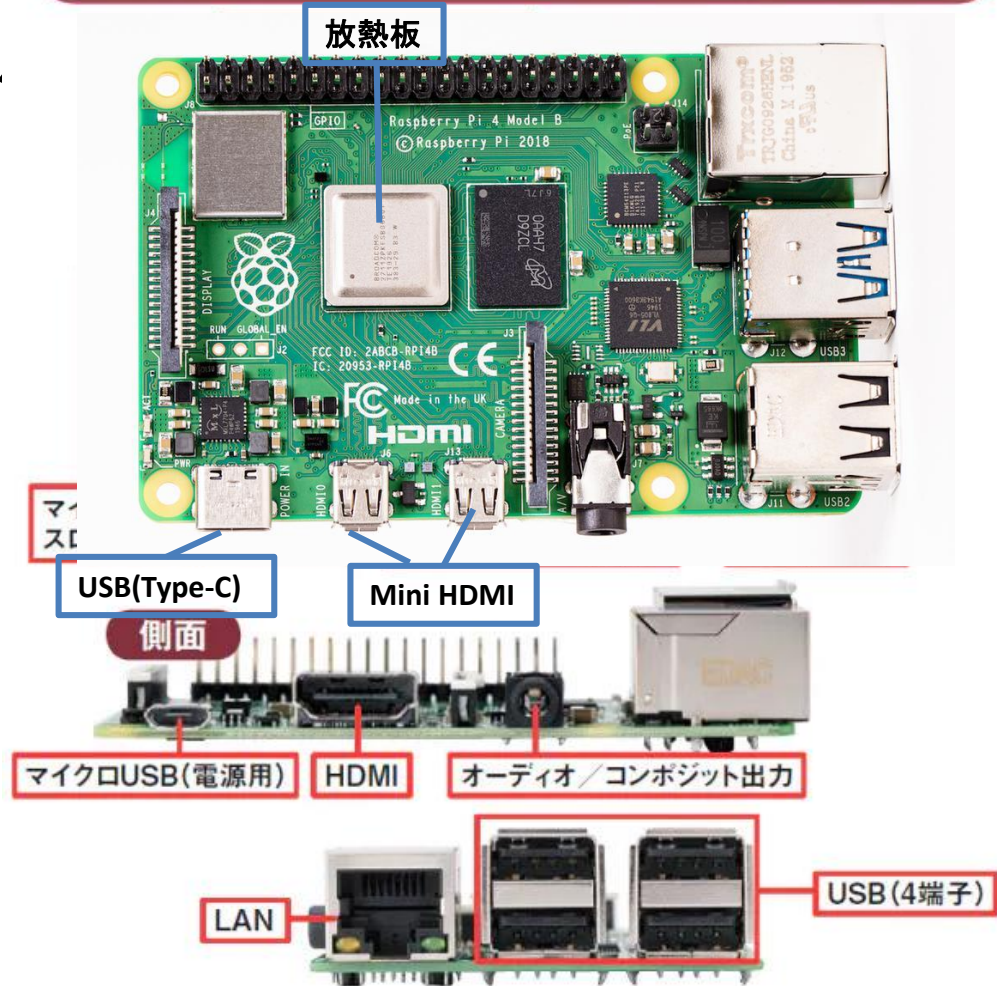
Arduino が走行制御を担い、それ以外の処理は Raspberry Pi が行っている。
Raspberry Pi と Arduino はシリアル通信により命令・データを送受信する。



Raspberry Pi 4

- らずべりーぱい
- シングルボードコンピューター
- PCと遜色ない機能・性能
 - CPU: 1.5GHz クアッドコア (ARM)
 - メモリ 4GB
 - USB, HDMI, LAN
- GPIO
 - 40pin
 - 3.3V動作
- 無線通信 (内臓)
 - WiFi 11b/g/a(5GHz)
 - Bluetooth (2.4GHz)

USBやLANなどパソコン並みの機能を搭載



Raspberry Pi の開発環境

- OS : [Raspbian](#) (らずびあん)
 - Debian系 Linux を Raspberry Pi にカスタマイズしたもの
 - 最新バージョン 20.04 (メジャー番号偶数、マイナー番号04が長期サポート版)
- ライブラリ
 - [Wiring Pi](#): GPIO、シリアル通信 (I2C、serial)
 - [OpenCV](#): 画像処理 (Ver2.0 / 3.0)
 - 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

Arduino

- Arduino UNO
 - [Arduino 製品群](#)中のスタンダード
 - シングルボードマイコン
 - 5V駆動（USB供給可）
 - IO
 - 5V信号
 - 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 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](#)
 - loop関数
 - main関数の中の while 文 のようなもの
 - setup 関数
 - ピンモード(入出力、PWM、割込み)を設定する

標準プログラム概要

- RaspberryPi には C 言語、Arduino には Arduino 言語でコーディングされた標準プログラムを用意している。
- RaspberryPi 用標準プログラムは、MIRS2015の巡回警備の競技会を実施する上で必要なモジュールで構成されている。
- Arduino 用標準プログラムは、直進・回転の走行制御を行うために必要なモジュールと RaspberryPi との通信モジュールで構成されている。
- 全ての単体機能をテストするプログラムが用意されている。
- 電子機械設計演習の実施までに新バージョンを提供する予定(現バージョンは 3.1/ 3.0)

標準ソフトのドキュメント

- MIRSMG4D 管理台帳



MIRSMG4D ドキュメント管理台帳

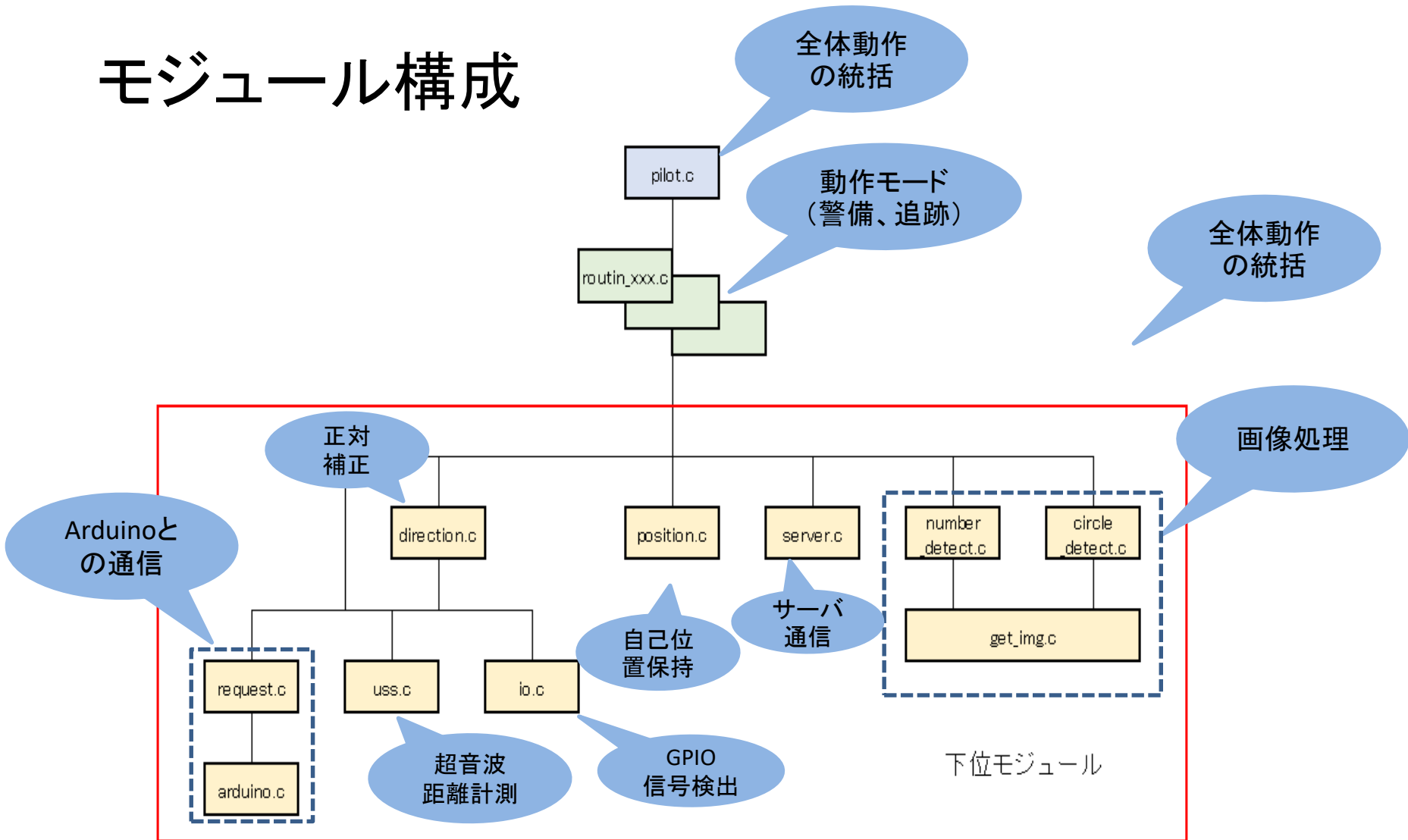
保護されていない通信 | www2.denshi.numazu-ct.ac.jp/mirsd2/mirsmg4d/

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
MIRSMG4D-SOFT-0001	Arduino ソフトウェア 解説	鈴木静男	A01	牛丸真司	2017/04/19	Ver 3.1
MIRSMG4D-SOFT-0002	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
MIRSMG4D-SOFT-0003	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
MIRSMG4D-SOFT-0004	RaspberryPi ソフトウェア 解説	鈴木静男	A01	牛丸真司	2017/04/19	Ver 3.0
MIRSMG4D-SOFT-0005	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
MIRSMG4D-SOFT-0006	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
MIRSMG4D-SOFT-0007	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)

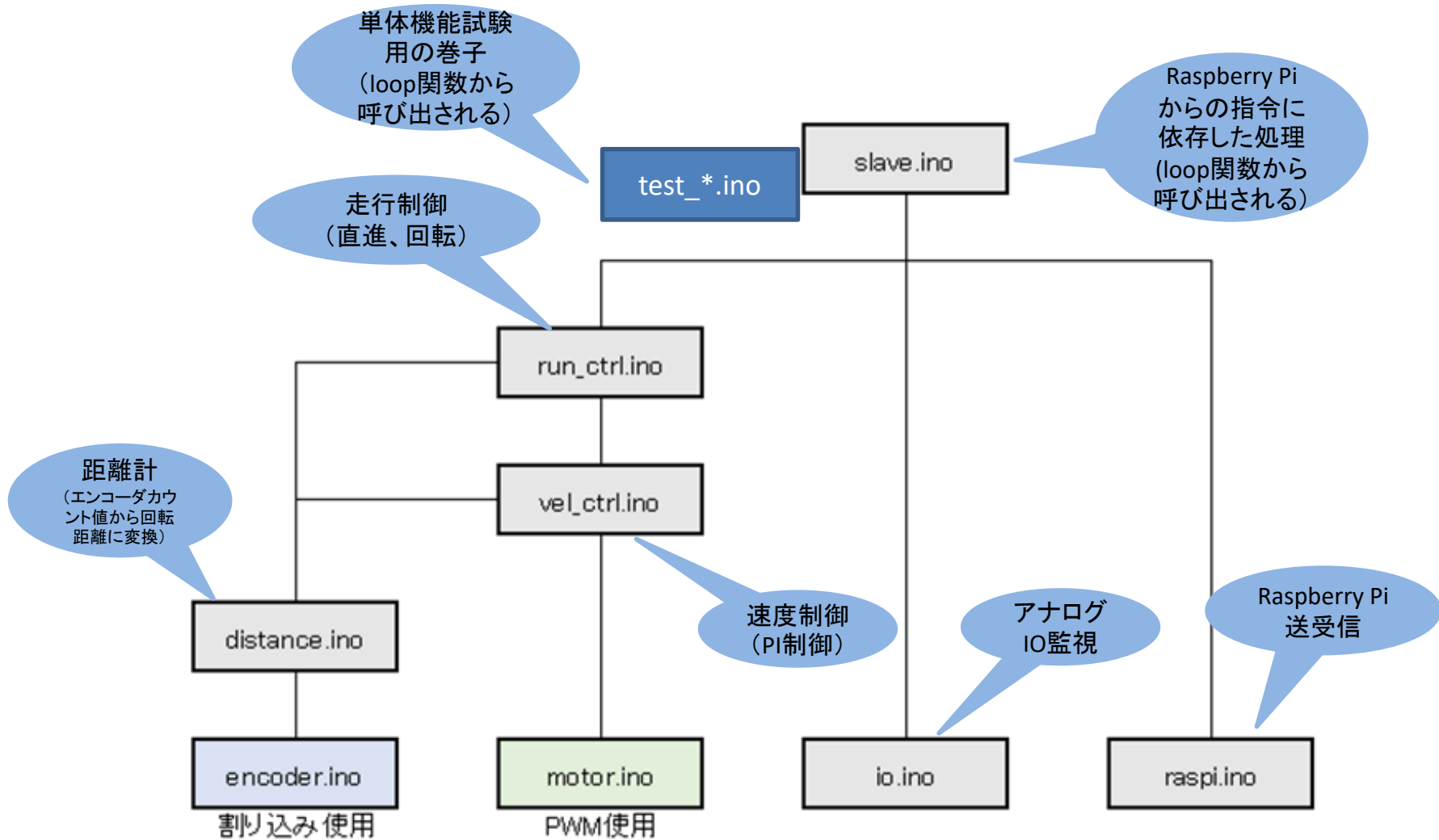
モジュール構成



機能毎にモジュール分割

1モジュール=1ファイル

標準プログラム (Arduino) のモジュール構成



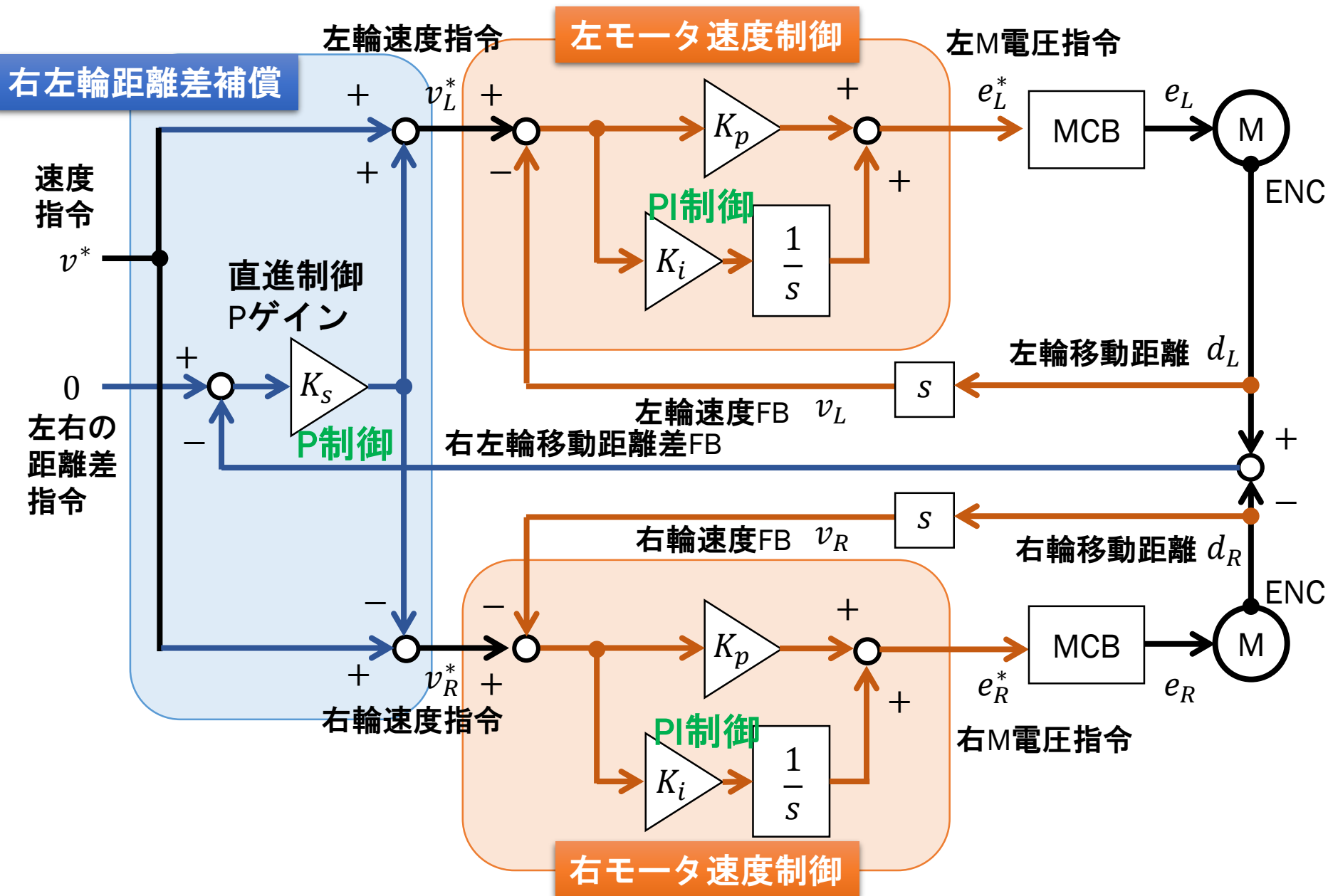
機能毎にモジュール分割

1モジュール=1ファイル

走行制御

- PWM制御により実効電圧を変化させて、モータ出力を制御する。[\(参考ページ\)](#)
- [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](#)によりエンコーダカウントが行い、モータの回転角(回転速度)を検出する。
- 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PD制御で行う。

直進制御ブロック線図



走行制御

- モータの速度制御 (PI制御)

$$\Delta v_{r,l}(k) = v_{r,l}(k) - v_{r,l}^{ref}$$

$$v_{r,l}(k) = K_p \Delta v_{r,l}(k) + K_i \sum_{k'=1}^k \Delta v_{r,l}(k')$$

- 機体の直進制御 (PD制御)

$$\Delta\theta(k) = \theta_l(k) - \theta_r(k)$$

$$v_l(k) = V_{ref} - K_s * \Delta\theta(k) - K_{sd}(\Delta\theta(k) - \Delta\theta_{k-1})$$

$$v_r(k) = V_{ref} + K_s * \Delta\theta(k) + K_{sd}(\Delta\theta(k) - \Delta\theta_{k-1})$$

ソフトウェア開発の動向

- GUI(ユーザインターフェース)の搭載
 - HDMIへのLCD接続(ラズパイ上での実装)
 - スマホ、タブレットへの実装

MIRS制御プログラムとの通信機能が必要になる
- 音声認識・画像認識機能の搭載
 - Raspberry Pi
 - 他のデバイス上
- Raspi 上のMIRS制御プログラムを C++/C#等で実装 (これまでは計画倒れ)

通信機能の実装

- MIRSの行動制御 C（標準プログラム）
- ユーザーインターフェイス Java, JavaScript
- 画像認識処理 python

- Raspberry Pi 上で動作させる場合
 - Raspi上でのプロセス間通信 （C言語応用演習で実施）

- 異なるデバイスで動作させる場合
 - 双方のデバイスで実装する
 - ソケット通信（WiFiで接続した場合）
 - その他の通信手段（クラウドの利用など）