

# 1.要求分析

## 2.戦略

## 3.構造

## 4.振る舞い

いなっし〜ず

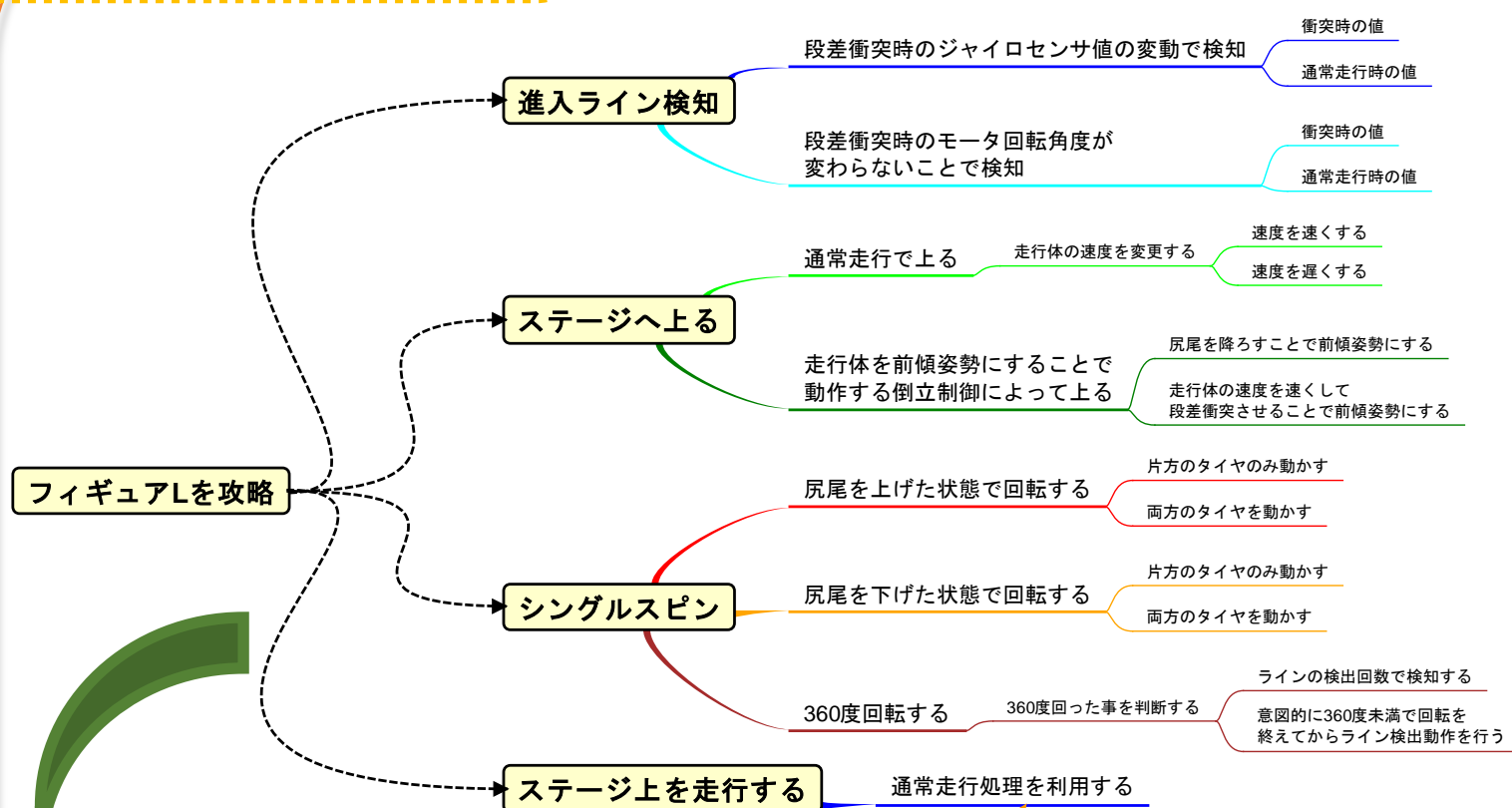
目標達成のために必要な機能をマインドマップとミスユースケースを使って抽出。

### 目標

フィギュアルを100%突破する

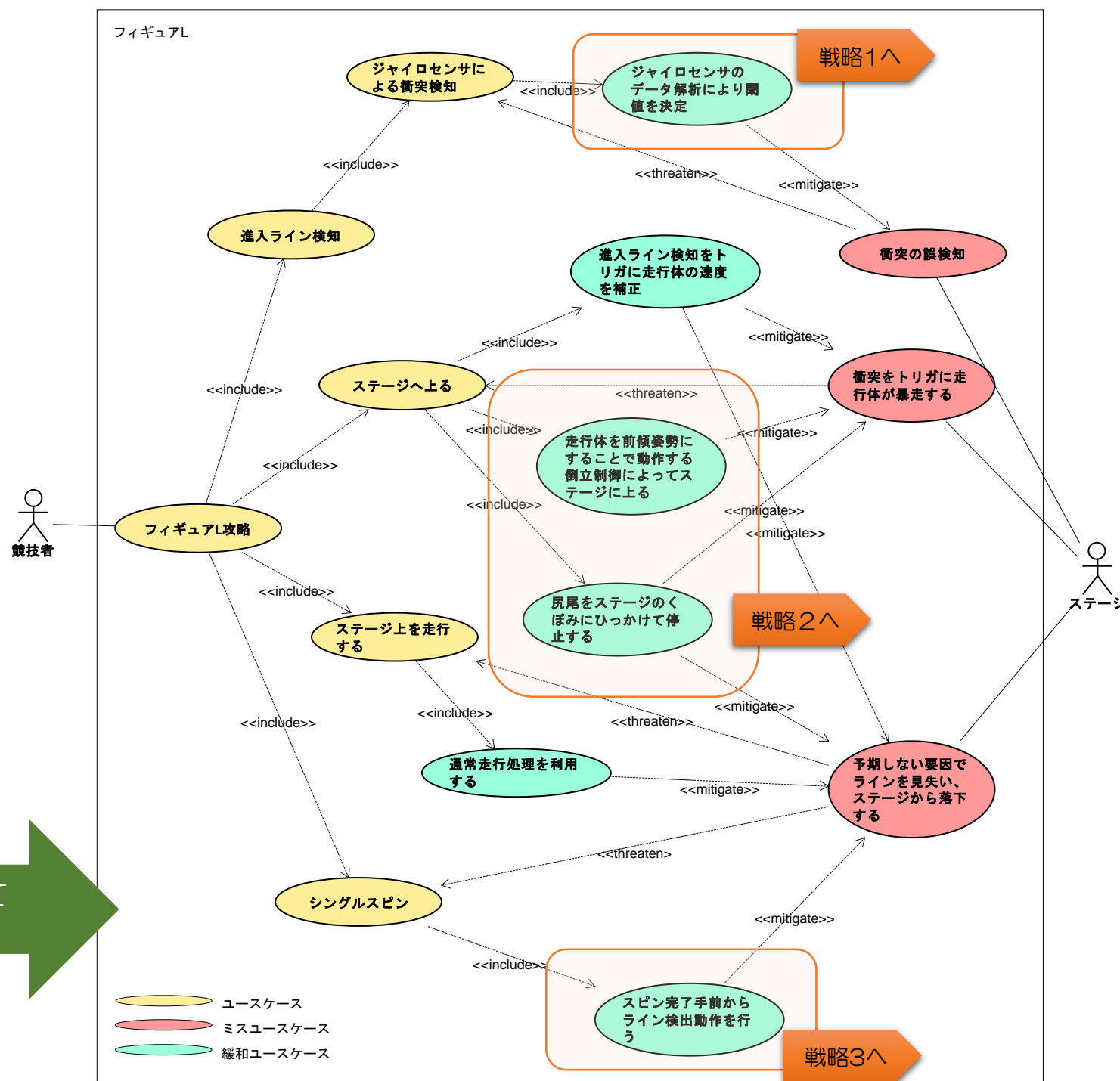
- ①フィギュアル攻略に必要な主機能をマインドマップで分析
- ②”100%突破”するためのリスクをミスユースケース図で洗い出し

### 攻略に必要な主機能を分析



### リスクを洗い出す

- ①ミスユースケースでリスクを整理
- ②各ミスユースケースに対して緩和ユースケースを検討



# 1. 要求分析

# 2. 戦略

# 3. 構造

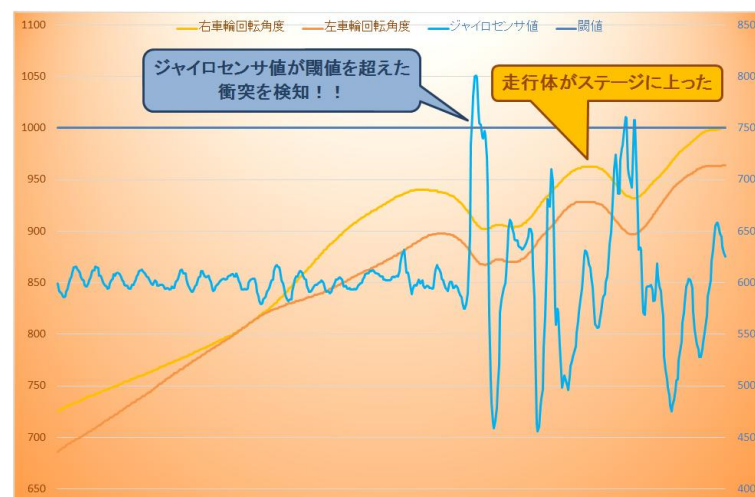
# 4. 振る舞い

いなっし〜ず

目的達成のための戦略を記載。ミスユースケースの分析結果に対して実現方法を検討。

## 戦略1) 衝突検知

衝突時のジャイロセンサ値の変動を解析して適正なセンサ値を割り出す。



速度50の場合、  
閾値750で正しく検知可能

## 戦略2) 尻尾を使ってステージに上る

進入ラインの検知

ステージに上る

尻尾をひっかけて停止



ステージに衝突した際のジャイロセンサ値を利用して、進入ラインを**正確に検知**する。



尻尾を下げることで走行体を前傾姿勢にする。**走行体を前傾姿勢から垂直に戻そうとする倒立制御**の特徴を利用してステージに上る。

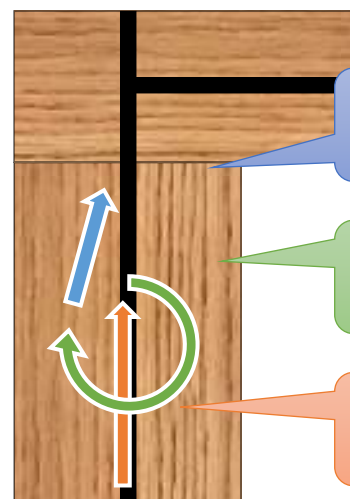


ステージに上った際、走行体を安定させるために一度**停止させる**。**尻尾をステージのくぼみにひっかける**ことで走行体が垂直になり、**安定した位置**で停止する。



## 戦略3) シングルスピン

回転を途中で止めてライントレースに切り替えることで、ラインを見失わないようにする。

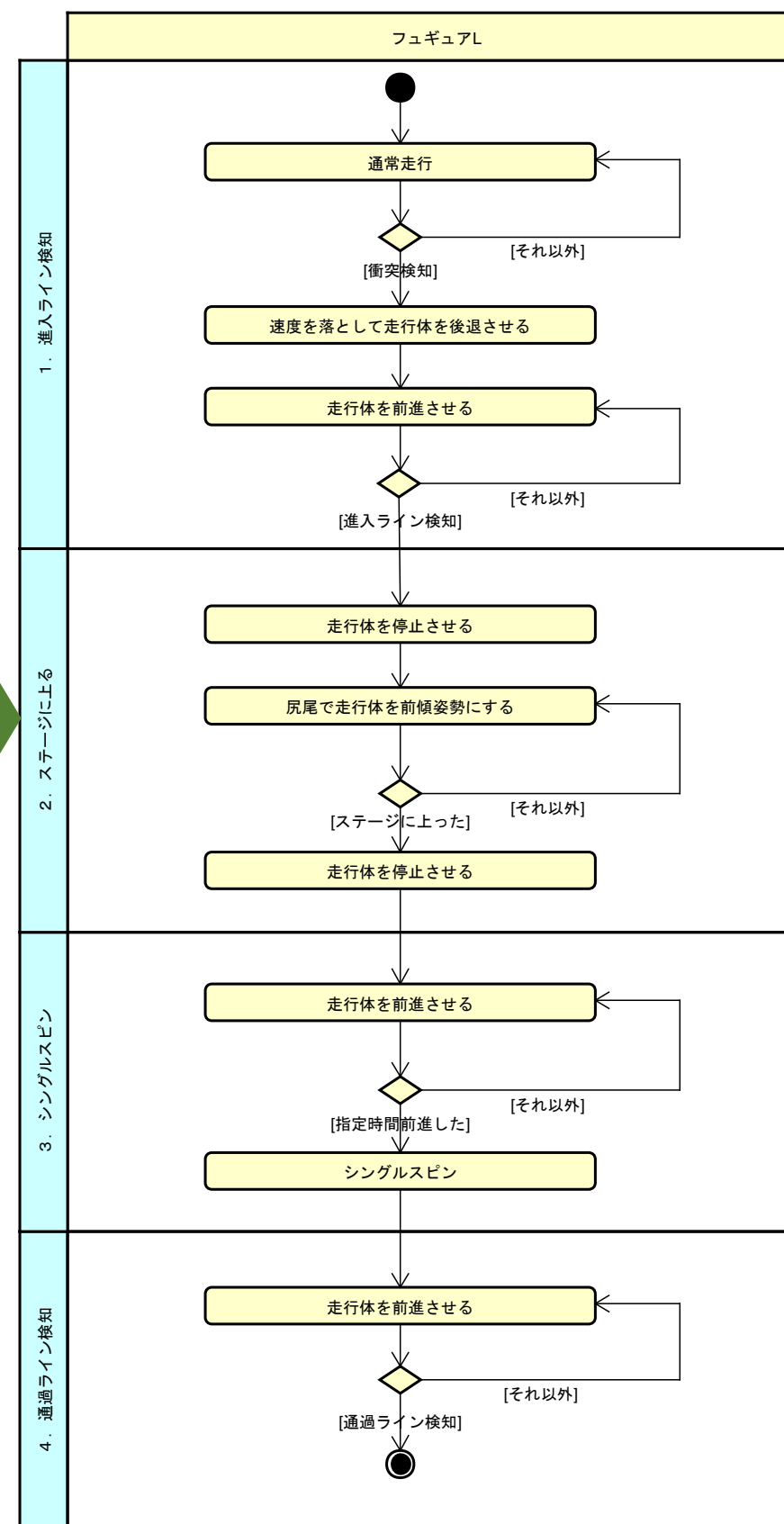


③ライントレースで  
ライン復帰

②右回転させる

①中央まで  
移動させる

## フュギュアル突破のシナリオ



解析で割り出したジャイロセンサ値を利用して進入ラインを正確に検知する。確実にステージに上るために一度後退して一定速度で前進する。

尻尾を使って走行体のバランスを変えてステージに上り、尻尾をくぼみにひっかけて停止する。  
ここが一番の見どころ!!!

ステージの中央まで移動して、シングルスピン。ラインを見失わないように回転を途中で止めてライン検知する。

最後は慎重に。確実にラインを検知する。通過ラインから降りて、通常走行に移行する。

ソフトの静的な構造を記載。緩和ユースケース実現のための適正なソフト構造を検討。

### 目標達成に必要な要素をクラス図で構築

フィギュアルで使用するクラス

- ・ 走行(走行全般機能)
- ・ 尻尾(尻尾の角度制御)
- ・ 倒立制御(倒立制御算出)
- ・ HW(デバイス制御)

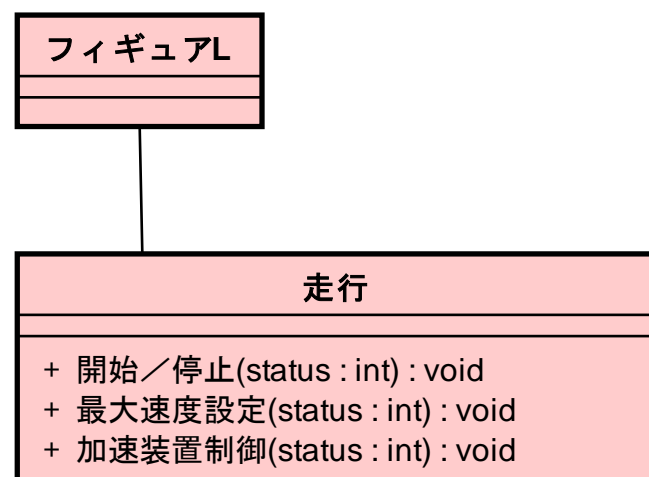
### 難所攻略の為、走行をカスタマイズ

緩和ユースケース実現の為に、

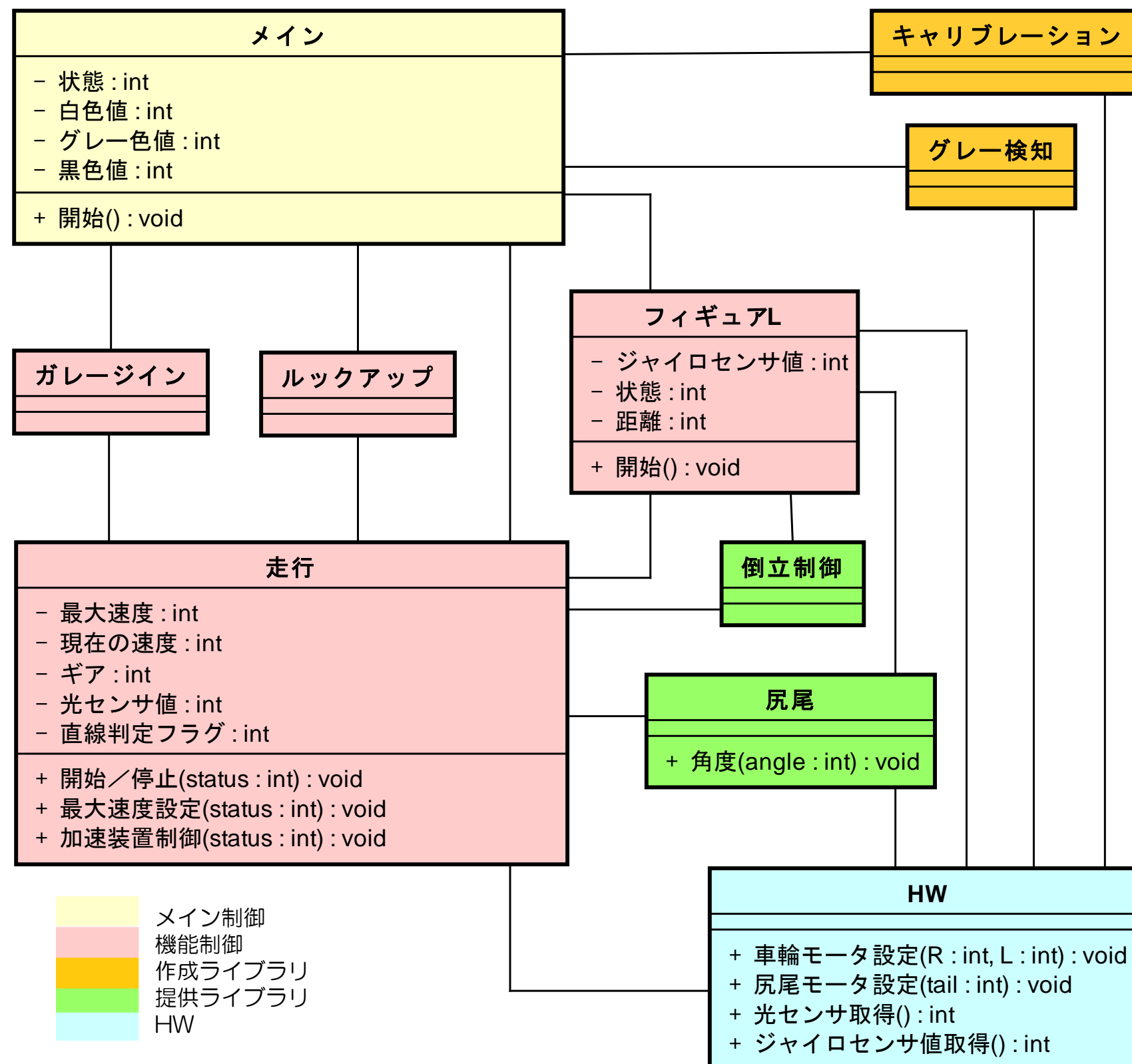
「走行」をカスタマイズ！

以下のmethodを作成

- ・ **進入ライン検知をトリガに走行体の速度を補正**  
=>最大速度設定、加速装置制御
- ・ **通常走行を利用する**  
／**スピン完了手前からライン検出動作を行う**  
=>開始/停止



### クラス図





# 1.要求分析

# 2.戦略

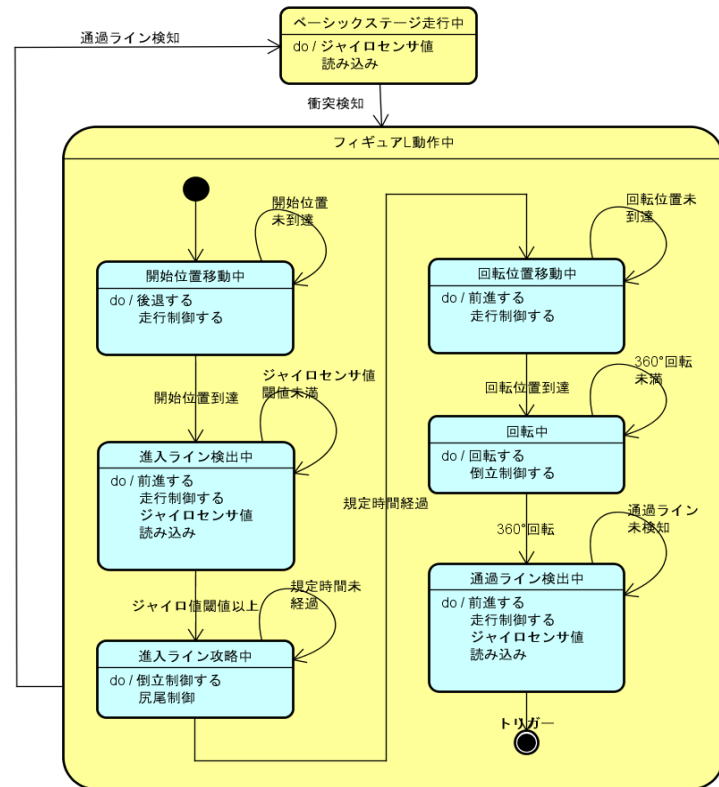
# 3.構造

# 4.振る舞い

いなっし〜ず

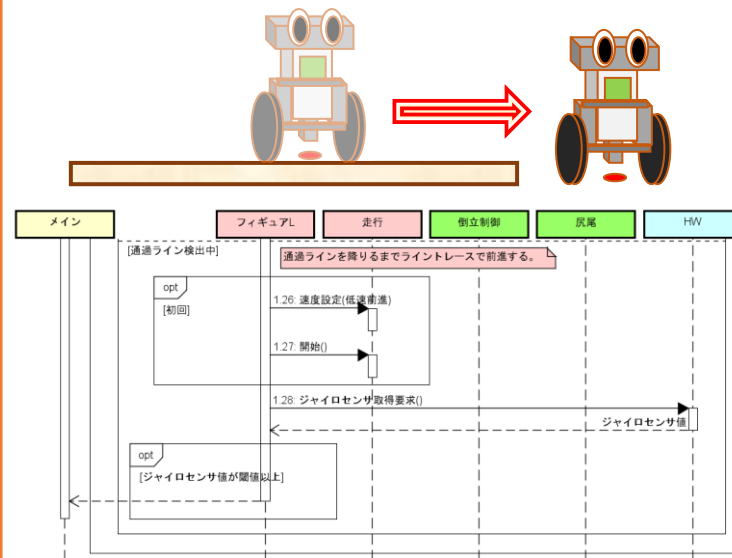
フィギュアL突破のためのソフトウェアの振る舞いを記載。

戦略に基づき、  
フィギュアL攻略時の状態を6段階に分けて検討。  
それぞれの状態の動作を示す。



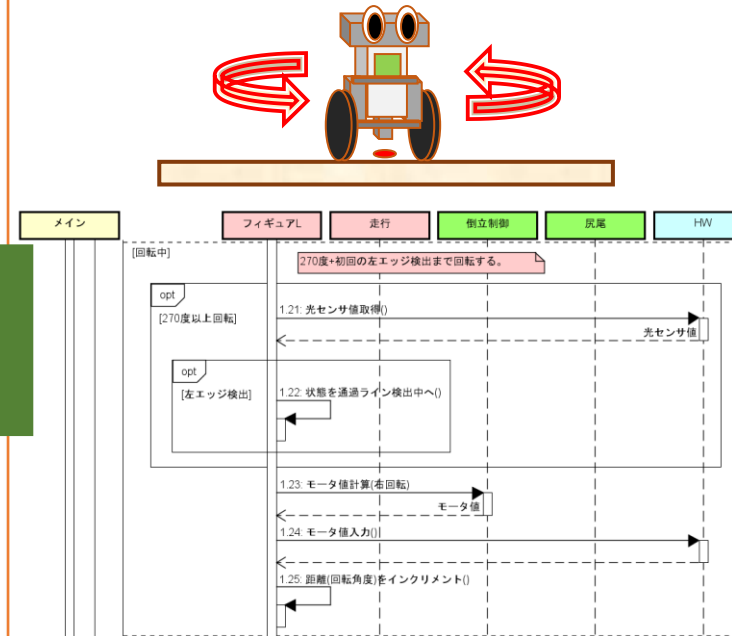
## ⑥通過ライン検出中

通過ラインを検知しフィギュアLを終了



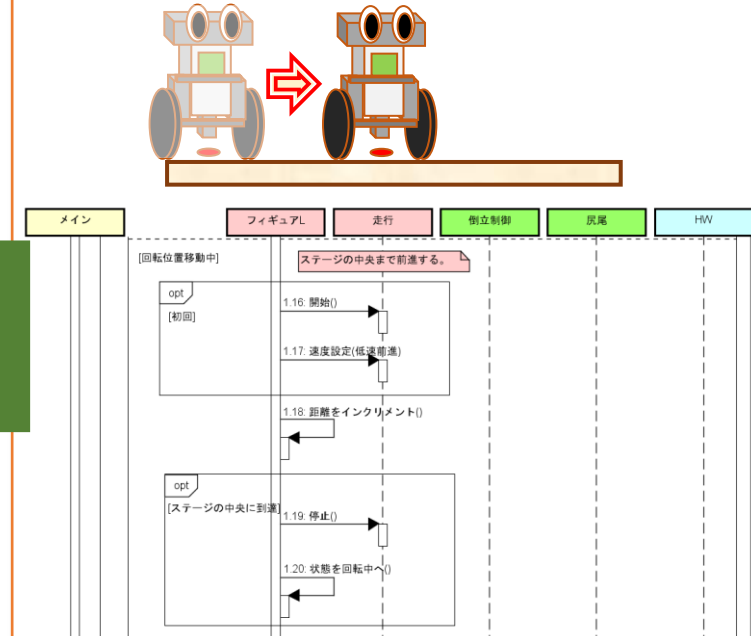
## ⑤回転中

シングルスピンの



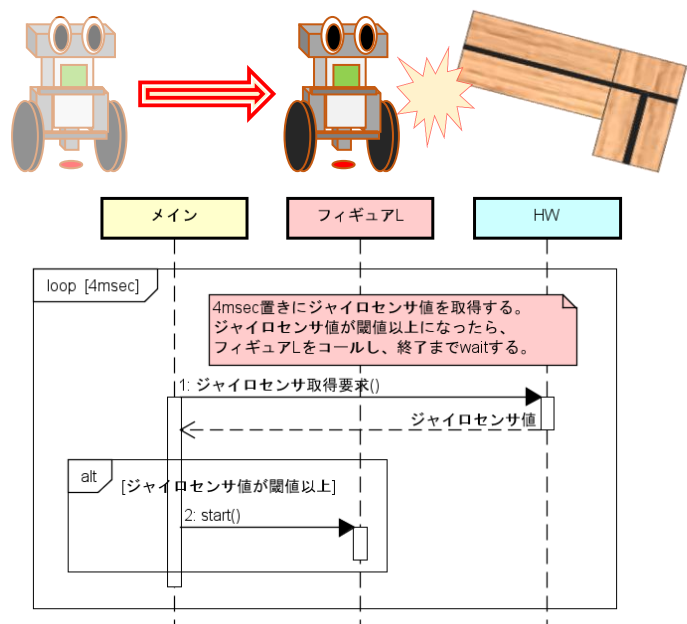
## ④回転位置移動中

ステージの中央まで移動する



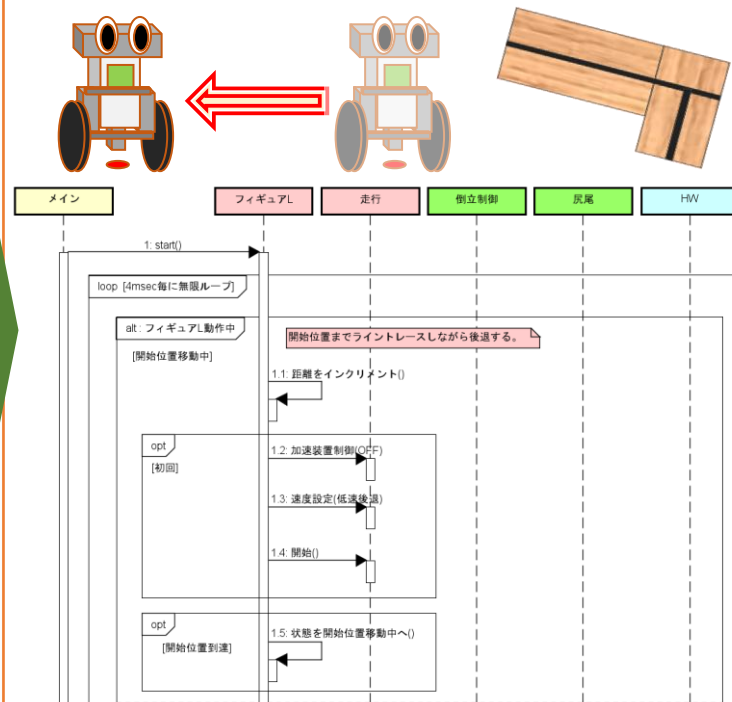
## ①ベーシックステージ走行中

進入ラインに衝突



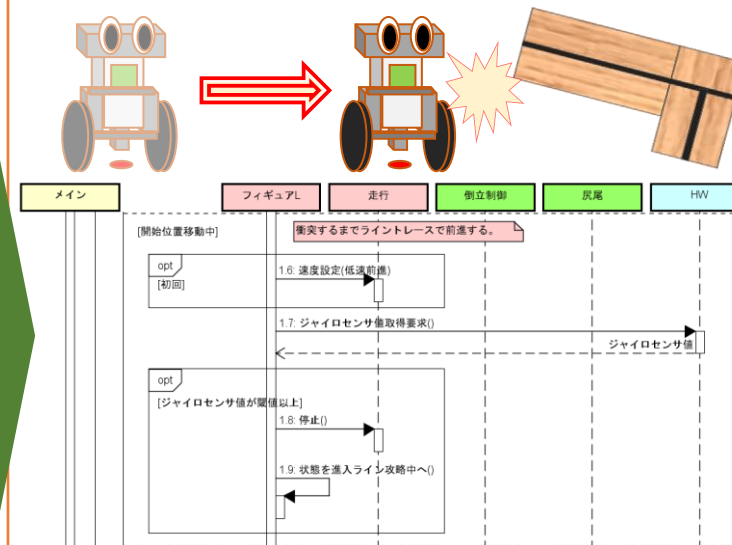
## ②開始位置移動中

一度後退する



## ③進入ライン検出中

進入ライン直前まで前進する



## ④進入ライン攻略中

尻尾を使ってステージに上る

