

チームNo. 228 チーム名： ペーパードライバー's 所属： 宮崎大学 工学部 環境ロボティクス学科

チーム紹介、目標、意気込み

宮崎大学工学部環境ロボティクス学科の5人で構成されたチーム「ペーパードライバー's」です。
今回は2回目の出場ですが1回目よりも大きく飛躍できるように頑張ります！

目標
全国大会優勝

意気込み
今回が2度目の参加となりますが、前回果たせなかった全国大会出場、そして優勝という目標に向かってチームで協力して頑張りたいと思います。

モデルの概要

選択課題：「シーソーの攻略」

- ・ シーソーを攻略するために走行シナリオを設定し、それを基に走行体に搭載する機能やリスク、緩和技術を検討した。
- ・ 機能を実現するための要素として、シーソー攻略クラス、ライントレース走行クラス、進行状態クラスを用意し、進行状態クラスを介してシーソー攻略クラスからシーソーの攻略状況に応じてライントレース走行を制御できる構造にした。
- ・ 進行状態クラスに一つの走行シナリオの終了ごとにシーソーの進行状態を上書きし、それを参照することで走行シナリオの切り替えを行った。
- ・ リスクへの緩和技術としてジャイロセンサによる段差検知、加速度計測によるシーソー動作検知、ジャイロオフセット値操作による走行体の急加速・急減速を搭載した。

モデルの構成

1. 機能モデル

- ・ 選択した機能実現のため、シーソー攻略処理全体を大まかに分割し、3つの走行シナリオとした。
- ・ 走行シナリオ一つ一つにアクティビティ図を用意し、機能実現のための仕様を決定した。
- ・ 走行シナリオから、ユースケース図とユースケース記述を用いて走行体に搭載する機能、機能へのリスクとリスクの緩和策を検討した。ここで検討した緩和策は緩和技術として「5. 工夫点」ページに記した。

2. 構造モデル

- ・ 基本走行シナリオ、アクティビティ図による仕様を基に必要な走行処理を実現ための要素を策定・実装し、それをクラス図として表現した。

3. 振る舞いモデル

- ・ クラス図で策定された要素を用いて、アクティビティ図で示した「仕様①～③」とそれを制御する「全体制御」、処理の中で何度か共通処理として呼ばれる「ライントレース走行」「倒立走行」の実際の振る舞いをそれぞれシーケンス図として記した。
- ・ 全体制御により「仕様①～③」が呼び出され、「仕様①～③」によって「ライントレース走行」と「倒立走行」が呼び出される。

4. 工夫点

- ・ ユースケース図、ユースケース記述で検討された緩和技術の詳細を図と自然言語を用いて説明している。

目標達成のために要求される走行シナリオと予想されるリスクについて分析した。

1-1. 目標の決定

選択課題

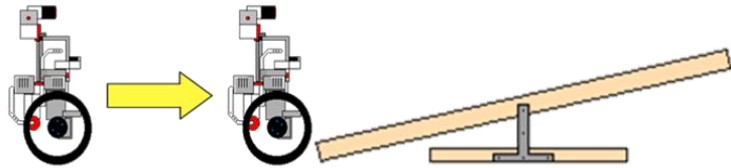
「シーソーを通過する」

走行目標

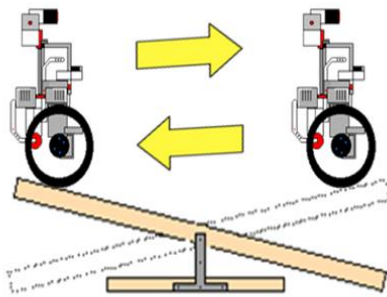
「シーソー通過「ダブル」を成立させ、転倒せずにシーソーを通過する」

1-2. 基本走行シナリオ

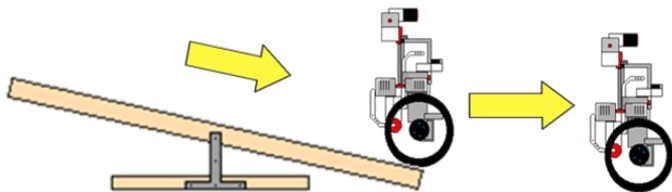
- シーソー処理全体を1~3の区間に分割し、それぞれの処理をアクティビティ図(1-3)で記述する。
- シーソー処理に必要な機能、リスク及び緩和策はユースケース(1-4)及びユースケース記述(1-5)を用いて検討した。



① シーソーを検知し、登る



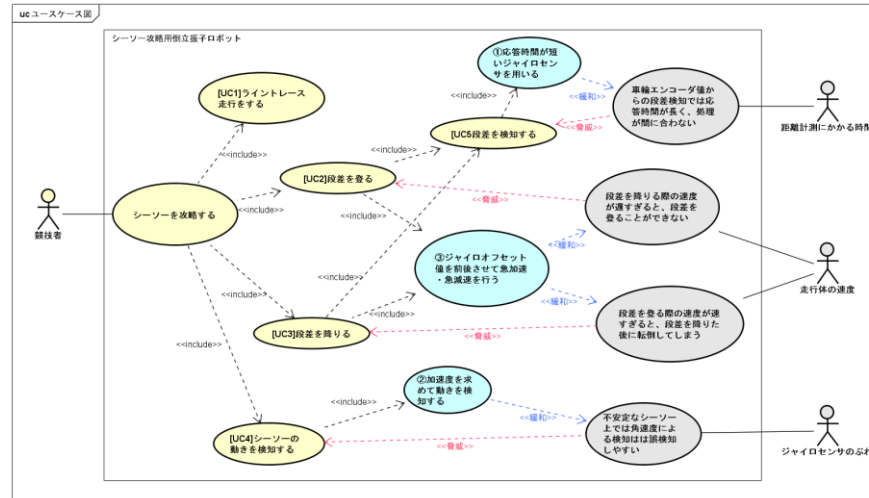
② 直進し、シーソーの動きを検知する(3回繰り返す)



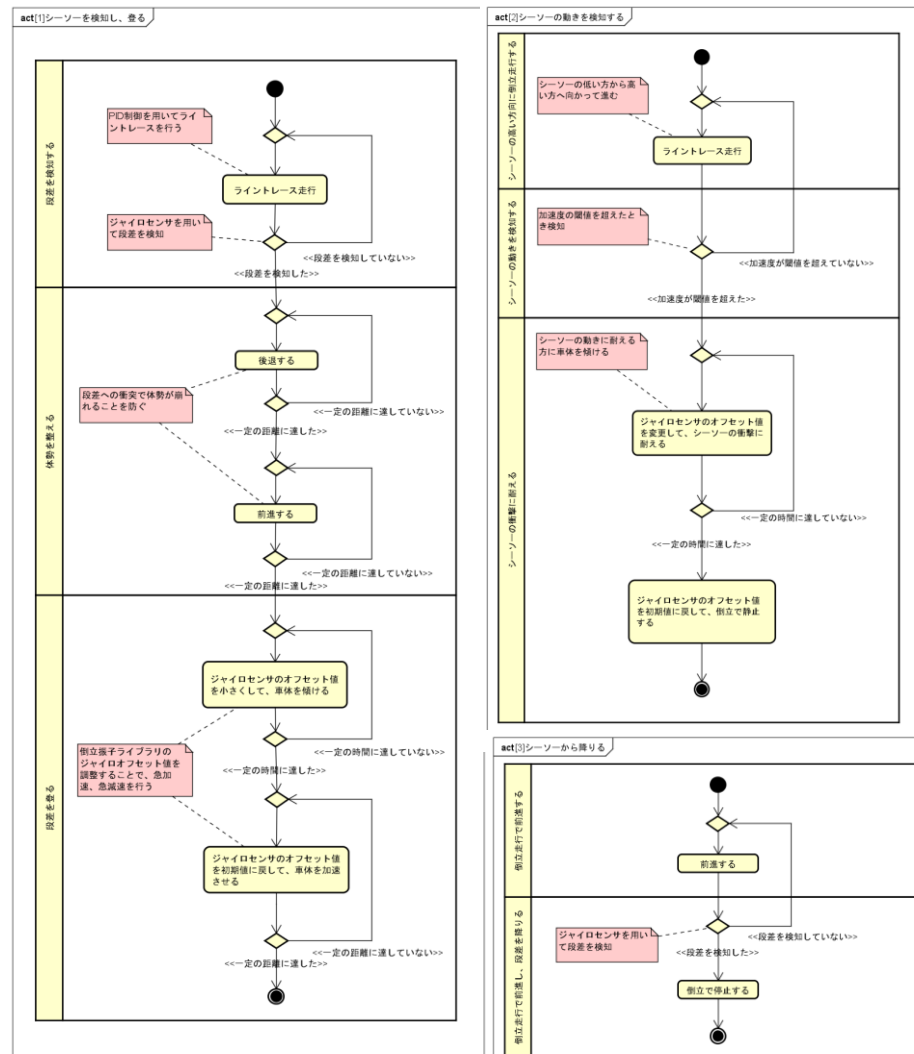
③ シーソーから降りる

1-4. ユースケース図

緩和技術①②③ (5.工夫点)



1-3. アクティビティ図

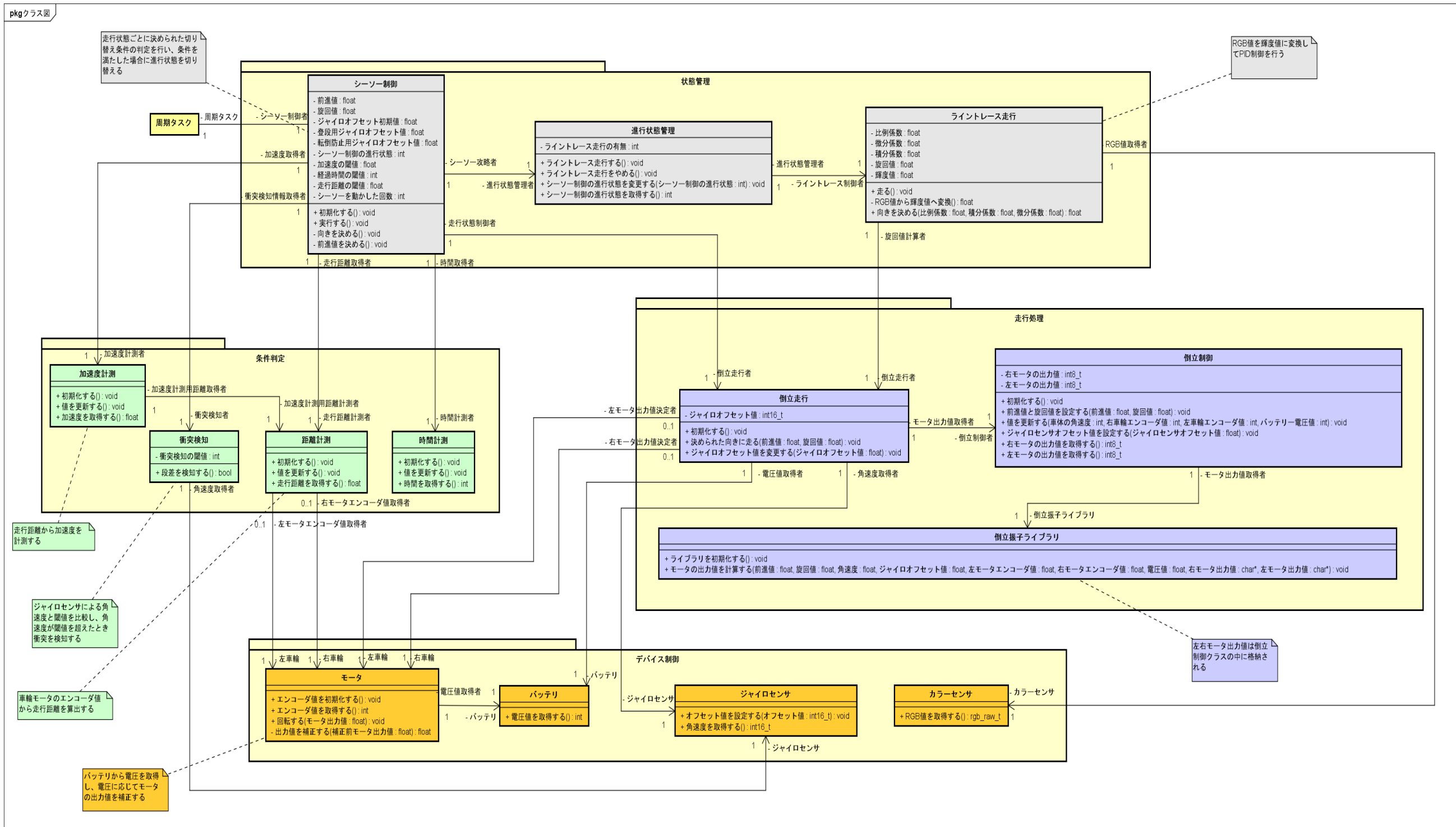


1-5. ユースケース記述

ユースケース記述・走行戦略分析	
ユースケース名	[UC1] ライトレース走行する
目的	ライン上をスムーズに走行する
事前条件	走行体が走行可能な状態であること
基本フロー	1. ロボットは終了条件を設定する 2. ロボットは各ゲイン、目標値を設定する 3. ロボットは光センサの値を取得する 4. ロボットはPID演算により、旋回角度を算出する 5. ロボットは終了条件を満たしたら、走行を終了する 6. 2~5を繰り返す
事後条件	終了条件を満たしている
ユースケース	[UC2] 段差を登る
目的	段差を登る
事前条件	段差の手前で倒立走行をしている
基本フロー	1. ロボットは倒立走行で前進する 2. ロボットは段差を検知する。(UC5) 3. ロボットは段差から一定距離下がる 4. ロボットは前進し、段差を登る 5. ユースケースを終了する
事後条件	段差の上で倒立走行している
脅威	段差を登る際の速度が遅すぎると、段差を登ることができない
緩和	ジャイロオフセット値を傾けることで急加速を行う(緩和技術③)
ユースケース	[UC3] 段差を降りる
目的	段差を降りる
事前条件	段差の上で倒立走行をしている
基本フロー	1. ロボットは倒立走行で前進する 2. ロボットは段差を検知する(UC5) 3. ロボットは前進し、段差を降りる 4. ユースケースを終了する
事後条件	段差の上で倒立走行している
脅威	段差を降りる際の速度が遅すぎると、体勢を安定させたまま段差を降りることができない
緩和	ジャイロオフセット値を傾けることで急減速を行う(緩和技術③)
ユースケース名	[UC4] シーソーの動きを検知する
目的	シーソーの動きを検知する
事前条件	シーソー上で倒立走行をしていること
基本フロー	1. ロボットは加速度の閾値を設定する 2. ロボットは左右の車輪からエンコーダ値を取得する 3. ロボットはエンコーダ値から走行距離を算出する 4. ロボットは走行距離から加速度を算出する 5. ロボットは加速度が閾値を超えたとき、シーソーが動いたと認識する 6. 2~5を繰り返す
事後条件	シーソーの動きを検知している
脅威	不安定なシーソー上ではジャイロセンサは誤検知しやすい
緩和	坂である上に動くシーソー上でのジャイロセンサの値は倒立するために細かいぶれが生じて検知に使用しにくい。そこで代わりに加速度を求めてシーソーの動きを検知する(緩和技術②)
ユースケース名	[UC5] 段差を検知する
目的	段差を検知する
事前条件	倒立走行を行っていること
基本フロー	1. ロボットは段差検知の閾値を設定する 2. ロボットはセンサの値を取得する 3. ロボットはセンサの値が閾値を超えたとき、段差の存在を認識する 4. 2~3を繰り返す
事後条件	段差を検知している
脅威	車輪エンコーダ値からの段差検知では応答時間が長く、処理が間に合わない
緩和	車輪からエンコーダ値を取得して段差検知を行うより応答時間が短いジャイロセンサを用いる。(緩和技術①)

- 前ページの基本走行シナリオより、必要な走行処理を実現する機能を実装した。
- 機能を実現するために必要なソフトウェア構造を分析し、クラス図で記述して以下に示した。

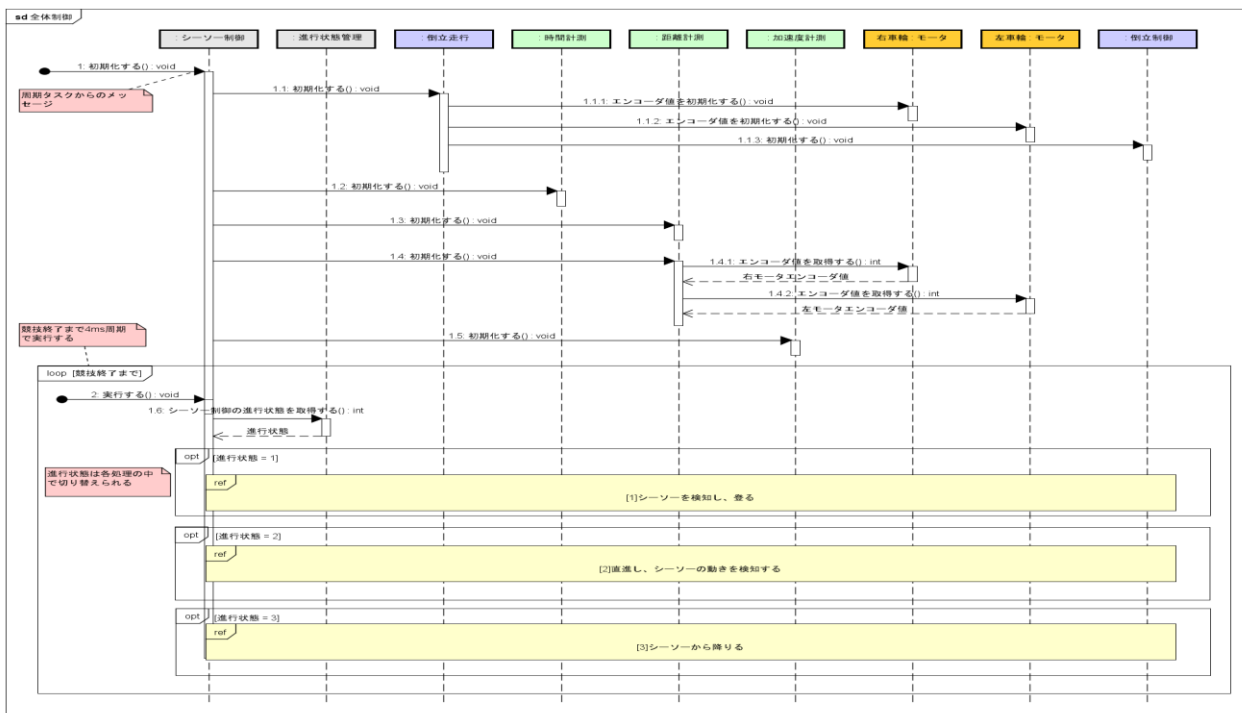
2. クラス図



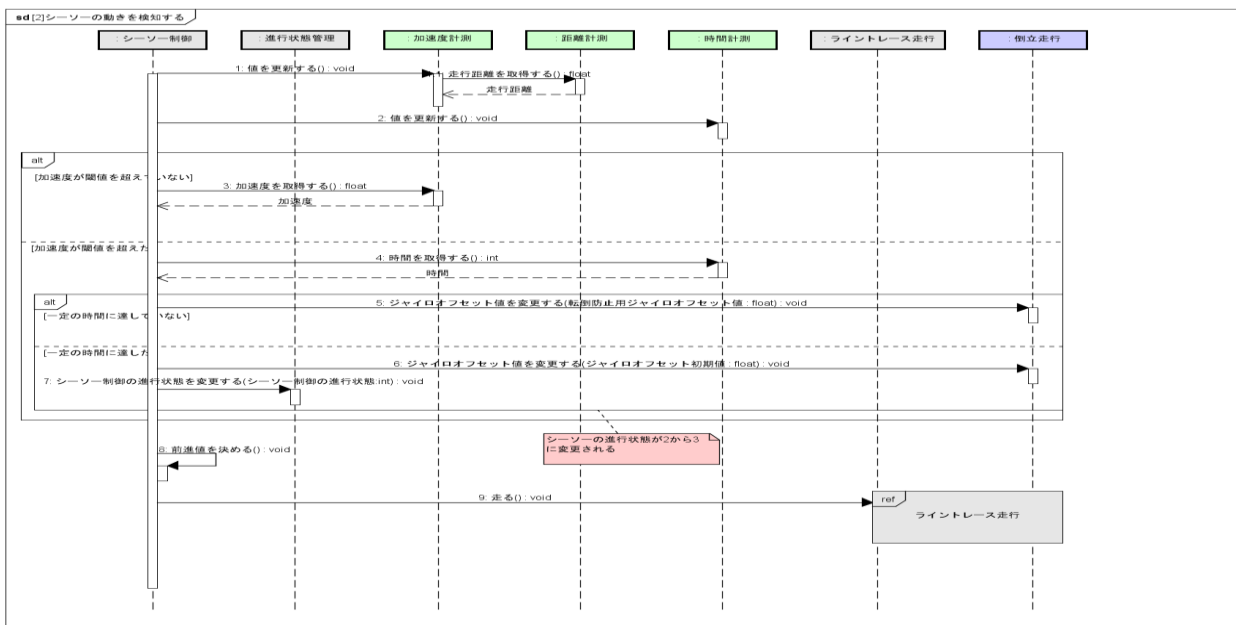
- アクティビティ図で示した①、②と全体制御の一連の動作をシーケンス図で記述した。
- 共通の処理であるライントレース走行、倒立走行の振る舞いは分けて記述した。

3. シーケンス図(1)

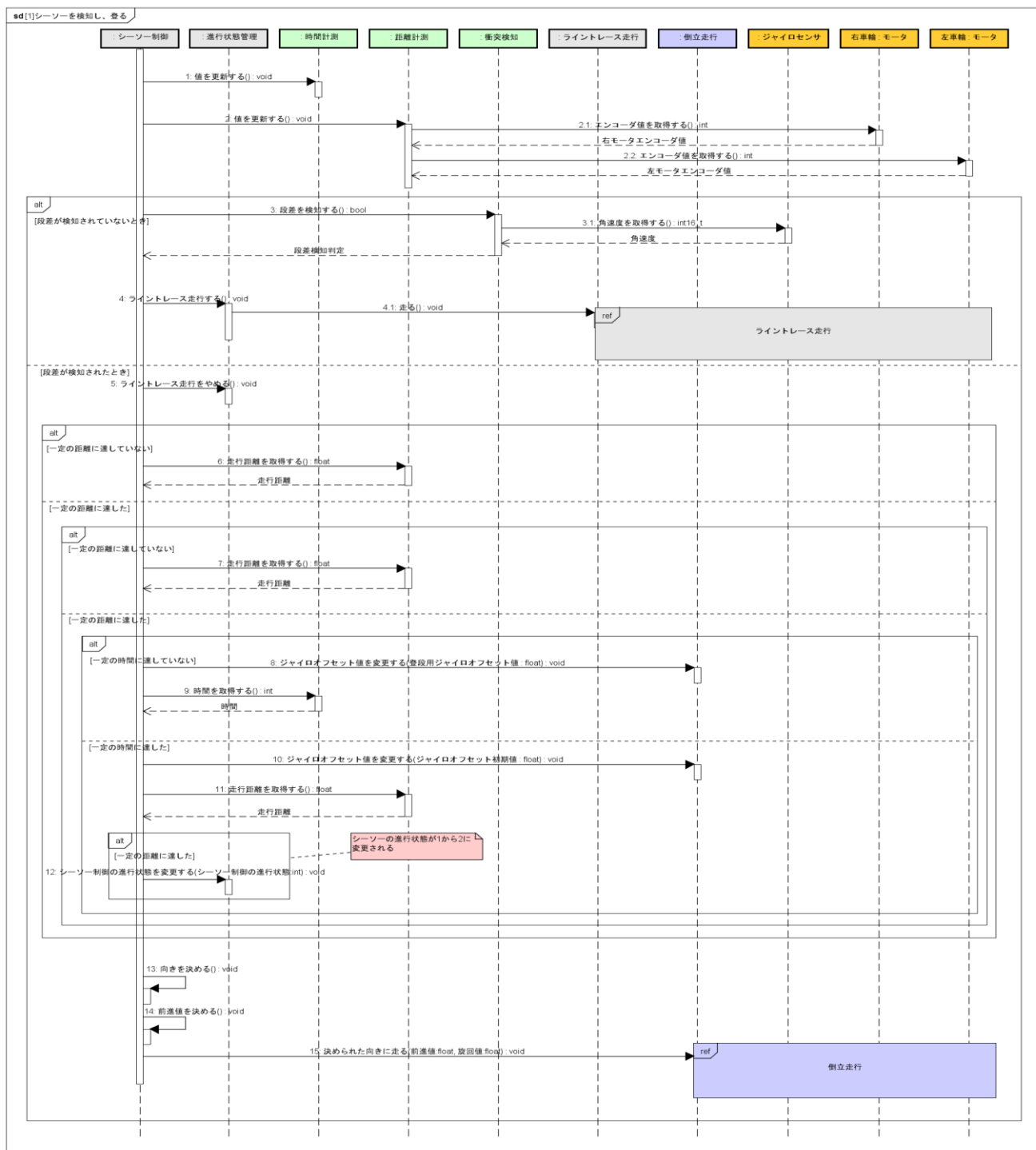
3-1. 全体制御



3-3. ②直進し、シーソーの動きを検知する



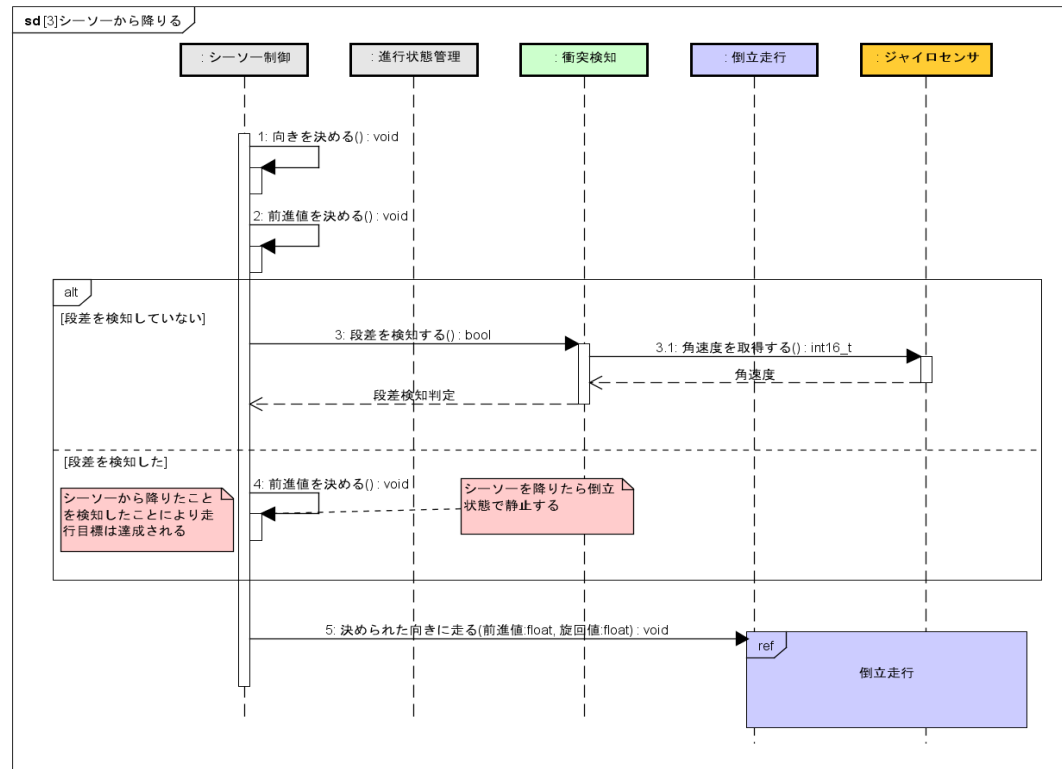
3-2. ①シーソーを検知し、登る



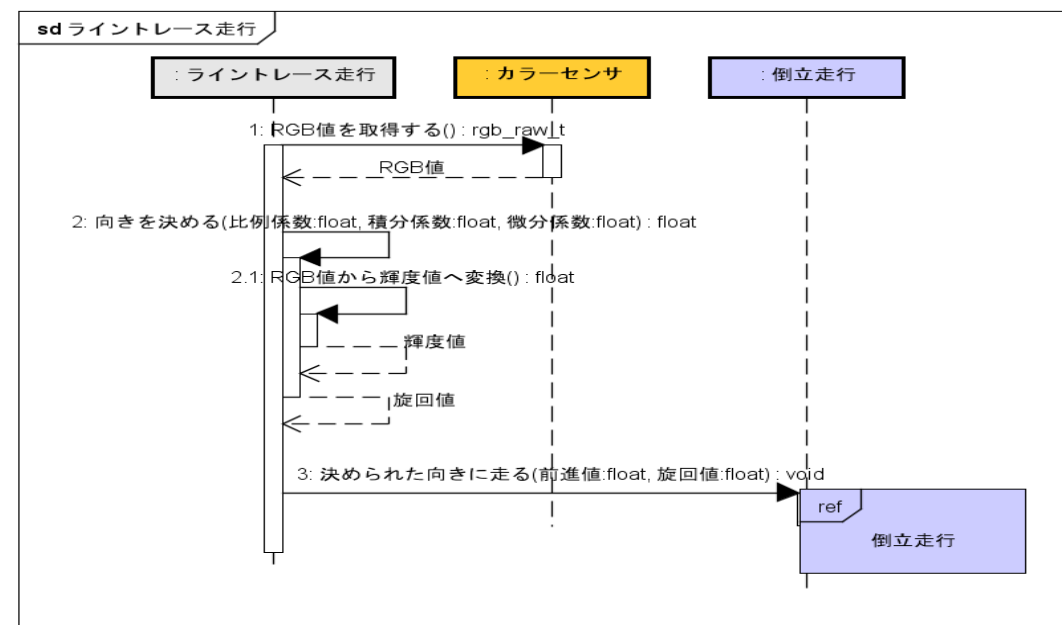
アクティビティ図で示した③と、共通処理として分割したライトレース走行、倒立走行の一連の動作をシーケンス図で記述した。

4. シーケンス図(2)

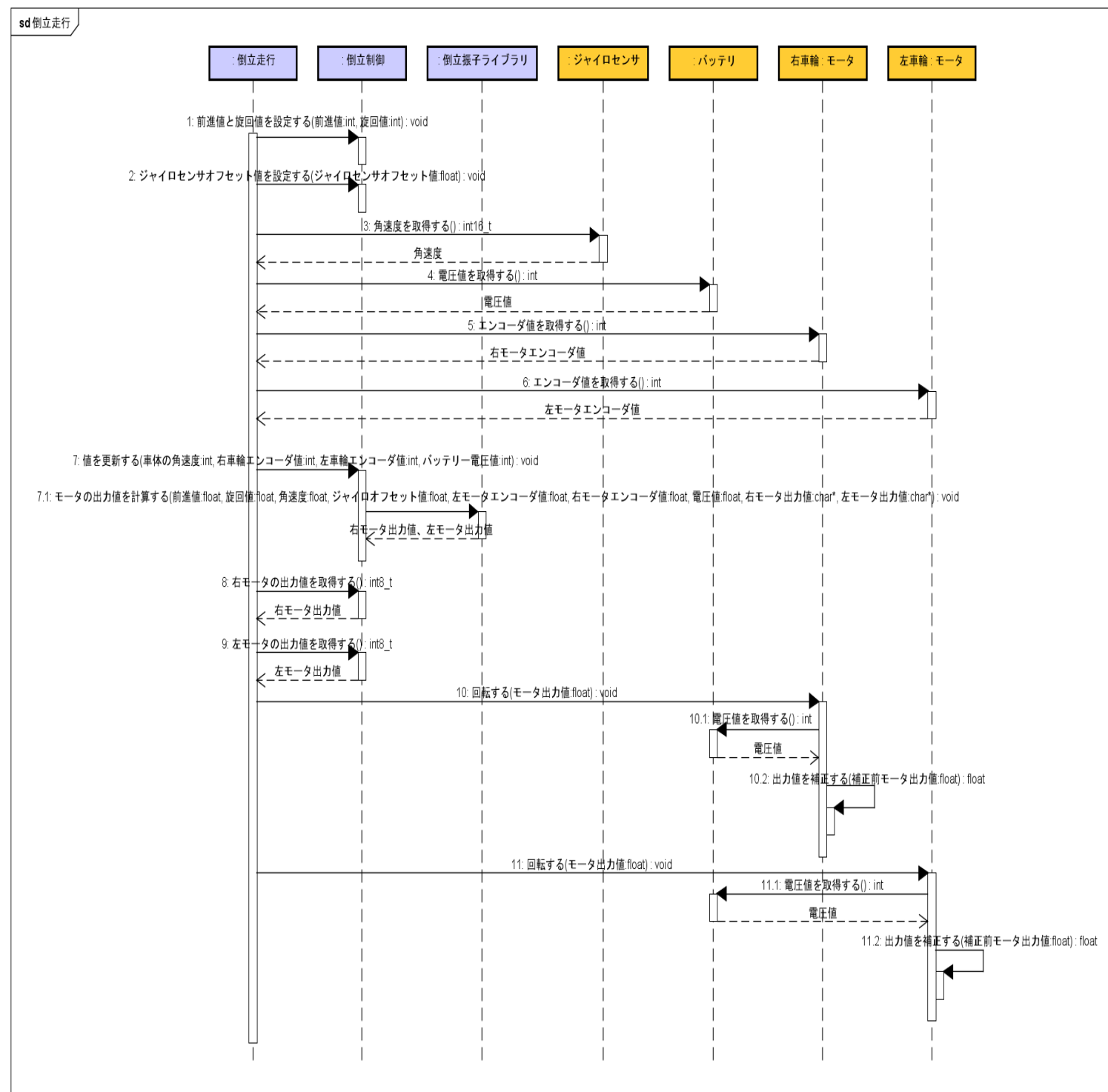
4-1. ③シーソーから降りる



4-2. ライトレース走行



4-3. 倒立走行

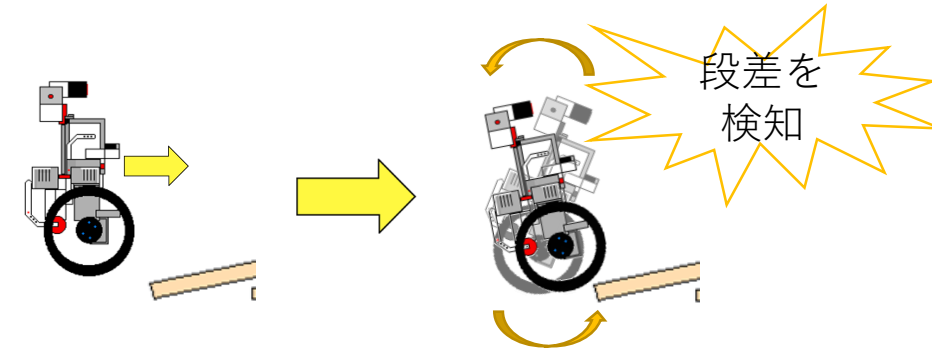
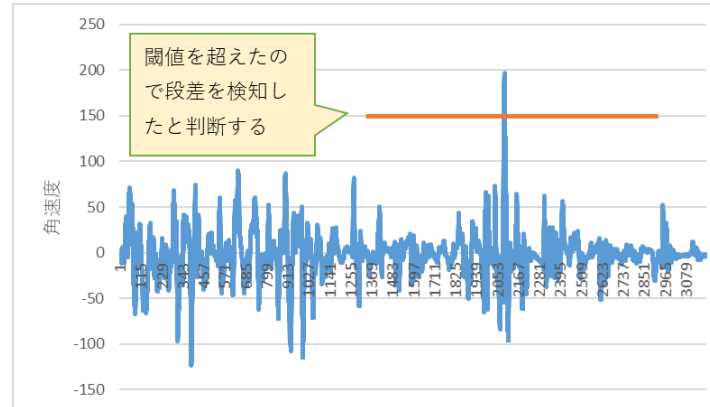


リスク分析で検討した緩和技術を図やグラフを用いて説明した。

①段差検知

[UC5]で使用

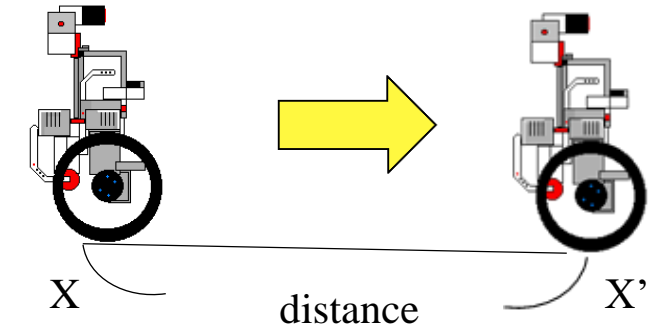
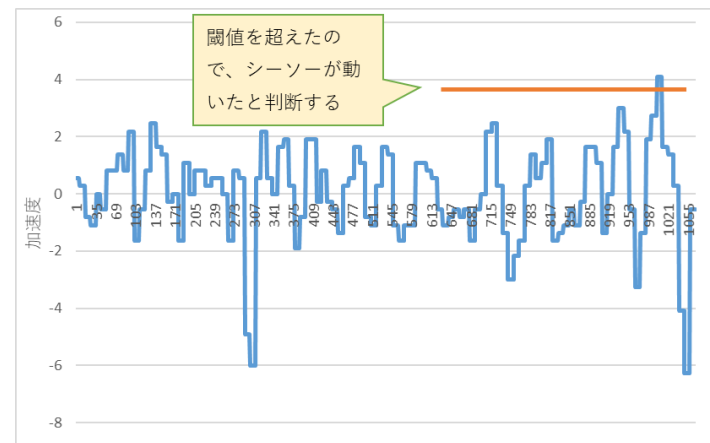
段差に衝突したときの角速度が、事前に設定した閾値を超えた際に検知したと判定する。



②シーソー動作検知

[UC4]で使用

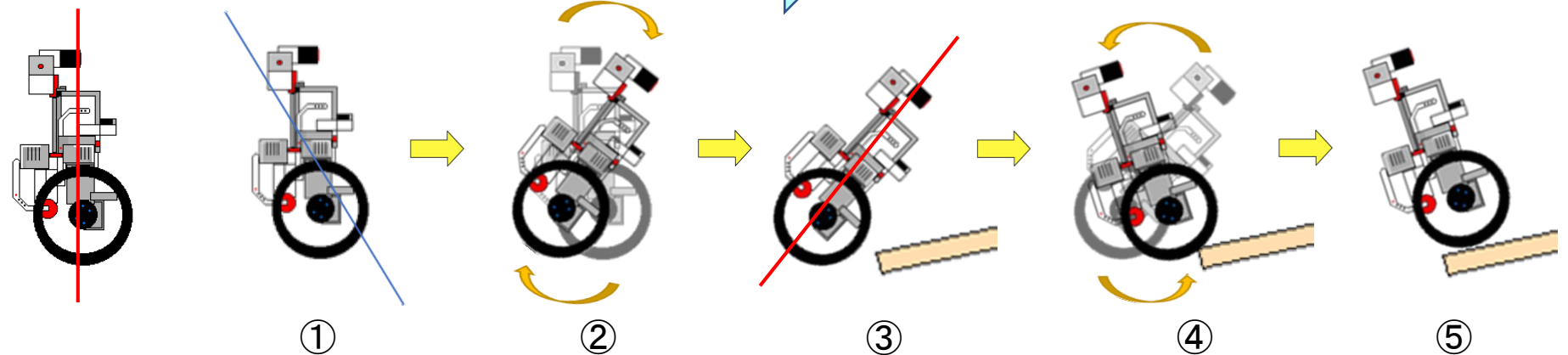
$X' - X$ を distance として distance を二回微分することで加速度を算出し、事前に設定した閾値を超えたときに検知したと判定する。



③ジャイロオフセット操作による急加速・急減速

[UC2],[UC3]で使用

- 赤線
初期のジャイロオフセット値
- 青線
操作したジャイロオフセット値



<(例)急加速による登段>

- ①図の青線のようにジャイロオフセット値を入力することで自身が傾いていると誤認させる
- ②正しい傾きに戻そうと車体を傾ける
- ③図の赤線のように初期のジャイロオフセット値に戻す
- ④再び直立状態に戻そうと②の時よりも大きく車体を傾ける
- ⑤走行体を傾けるときに加速を利用して段差を登る

- ジャイロオフセット値を操作することによって走行体に自分が傾いていると誤認させ、傾きを修正する際の加速を利用して急加速・急減速を可能にした。
- 図では急加速について説明しているが、加速時とは逆方向のオフセット値を入力することで、同様の仕組みで急減速も行うことができる。