

# Three Anagos Project

Version 1.00  
09/26/2018

太田 薫

## 目次

1	はじめに .....	5
2	三匹のあなごプロジェクト .....	6
2.1	プロジェクトとは .....	6
2.1.1	プロジェクトの目的と成功の条件 .....	6
2.1.2	開発プロセス .....	6
2.2	三匹のあなごプロジェクトの目的と成功の条件 .....	7
2.2.1	プロジェクト・メンバー .....	8
2.3	成果物 .....	8
2.4	プロジェクト・マネジメント .....	9
2.4.1	プロジェクト・マネジメントの基礎概念 .....	9
2.4.2	プロジェクト・マネジメントのプロセス .....	11
2.4.3	ものごとの重要度について .....	12
2.4.4	三匹のあなごプロジェクトの開発工数の目標 .....	13
2.4.5	開発スケジュール .....	14
	.....	14
2.4.6	あなごの WBS .....	15
2.4.7	あなごのトレードオフマトリックス .....	15
3	三匹のあなごシステム .....	16
3.1	システムとは、 .....	16
3.1.1	システムのライフサイクル .....	16
3.1.2	システムのコンセプト .....	16
3.1.3	Systems Engineering のテクニカル・プロセス .....	17
3.1.4	開発プロセスと手法 .....	18
3.2	あなごシステムの概要 .....	19
3.2.1	三匹のあなごシステムの目的と目標 .....	19
3.2.2	システムのライフサイクル .....	19
3.2.3	ステークホルダー .....	19
3.2.4	システム構成 .....	20
3.2.5	コンセプト .....	21
4	システム設計 .....	22
4.1	システム設計とは .....	22
4.2	要求分析 .....	23
4.2.1	要求とは .....	24
4.2.2	要求分析に用いられる図やモデル 要求分析の成果物 .....	26
4.2.3	コンテクスト分析 .....	27
4.2.4	ユースケース分析 .....	28
4.3	あなごシステムの要求分析 .....	29

---

4.3.1	ステークホルダー要求.....	29
4.3.2	あなごシステムの振る舞い(シナリオ).....	30
4.3.3	あなごシステムの要求 .....	31
4.3.4	あなごシステムの要求図(SysML) .....	32
4.3.5	あなごシステムのコンテクスト分析.....	33
4.3.6	あなごシステムのユースケース分析 .....	34
4.4	機能分析 .....	35
4.5	あなごシステムの機能分析 .....	36
4.5.1	機能フロー図 「コースを完走する」.....	36
4.6	アーキテクチャ設計 .....	37
4.6.1	機能ブロック図 .....	37
4.6.2	システム構造図 .....	38
4.6.3	ステートマシン図 .....	39
4.6.4	シーケンス図.....	41
5	あなごプロジェクトの振り返り .....	44
5.1	プロジェクト目標の達成状況と成果物 .....	44
5.2	システム目標の達成状況 .....	44
5.3	地区大会の結果.....	44
5.4	反省点 .....	45
6	おわりに .....	46
7	Appendix .....	47
7.1	資料一覧 .....	47
7.2	左コース座標 .....	48
7.3	右コース座標 .....	49

---

## 変更履歴

Rev	Date	Description	Who
0.03	03/25/2018	・ドラフト	太田
0.04	03/26/2018	・プロジェクトの目的を追加。開発プロセスと手法を追加	太田
0.06	03/27/2018	・開発工数の修正	太田
0.07	03/28/2018	・開発スケジュールの追加	太田
0.11	03/31/2018	・システム設計 要求の追加	太田
0.20	04/01/2018	・はじめにを追記	太田
0.24	04/07/2018	・要求分析を修正	太田
0.26	04/12/2018	・要求を追加	太田
0.27	04/17/2018	・開発スケジュールの修正。シナリオ作成を追加	太田
0.30	04/30/2018	・あなごの要求4.3.3 を追加	太田
0.31	05/01/2018	・サブシステム設計 <b>Error! Reference source not found.</b> を追加	太田
0.32	05/11/2018	・Appendix にコース座標を追加	太田
0.33	05/26/2018	・あなご手足システム内部ブロック図を追加	太田
0.40	06/16/2018	・あなごシステム・アーキテクチャ 機能ブロック図を追加	太田
0.41	06/17/2018	・あなご判断システムの構造図を追加	太田
0.43	06/23/2018	・判断、手足システムの構造図の修正、右コースデータの追加	太田
0.50	06/30/2018	・あなごシステムのシステム図、状態遷移図、シーケンス図を追加	太田
0.51	07/14/2018	・システムのシーケンス図を追加	太田
1.00	09/25/2018	・プロジェクトの振り返りを追加	太田
1.00	09/26/2018	・おわりにを追加	太田

## 1 はじめに

我々は、モデルベース開発の技術力向上を目指し 2016 年から ET ロボコンに参加している。私は 2017 年にシステムズ・エンジニアリングの考え方が有効か検証するために参加し、17 年の活動を通して SE の有効性を確認できた。2018 年で 3 回目の参加となるが、過去 2 年の活動を振り返ると問題点が 2 つある。一つ目は、本来の目的は教育であるが、教育の場としは非効率的である。二つ目は閉ざされた教育の機会である(休日に活動できる人のみ参加が出来る)。

2018 年度は教育を目的としたチーム「三匹のあなご」を結成し、ET ロボコンの本来の目的である技術力の向上に注力した活動を行っていく。本活動は、進捗と工数の監視、管理を行うことでメンバーの負担にならないロボコンを目指し、誰でもロボコンに参加できる事を示す。技術者として重要なのは「概念の構成力」と「ひらめき力」である。この 2 つの力を伸ばす活動を行っていく。

本資料は、「三匹のあなごプロジェクト」の活動記録と設計の教科書となる事を目指し作成していく。

## 2 三匹のあなごプロジェクト

### 2.1 プロジェクトとは

「プロジェクトとは、独自のプロダクト、サービス、所産を創造するために実施される有期性の業務である」

PMBOK

	特徴	説明
1	有期性	明確な開始日と終了日がある
2	独自の成果物	以前に作成されたことが無い、独特な製品やサービスを開発できる
3	繰り返し性がない	同じプロジェクトを二度行う事はない
4	段階的な詳細化	作成される製品やサービスは徐々に段階を追って詳細化される
5	複数のメンバー	プロジェクトマネージャと複数のメンバーによって実行する

### プロジェクトの三大要素

要素	説明
スケジュール	プロジェクトの時間的要素(いつ完了するか)
リソース(コスト)	プロジェクトの金額的な要素(いくらかかるか)
スコープ	プロジェクトの実施範囲(どこまでやるか、どのレベルまでやるか)

#### 2.1.1 プロジェクトの目的と成功の条件

プロジェクトの成功とは何をもって成功というのか？

プロジェクトから獲得すべき価値を明確にした上でプロジェクトの目的を明らかにし、成功の定義を定める必要があります。

#### 2.1.2 開発プロセス

開発プロセス	特徴
ウォーターフォール型開発(V字型モデル)	比較的大規模なプロジェクト向け、工程ごとの成果物が明確、後戻りが想定されていない
反復型開発	システムをサブシステム、コンポーネントに分割し、順次開発する
アジャイル型開発	イテレーションを採用、開発対象を小さな機能に分割し、1つのイテレーションで1つの機能を開発

どれでいきましょうか？

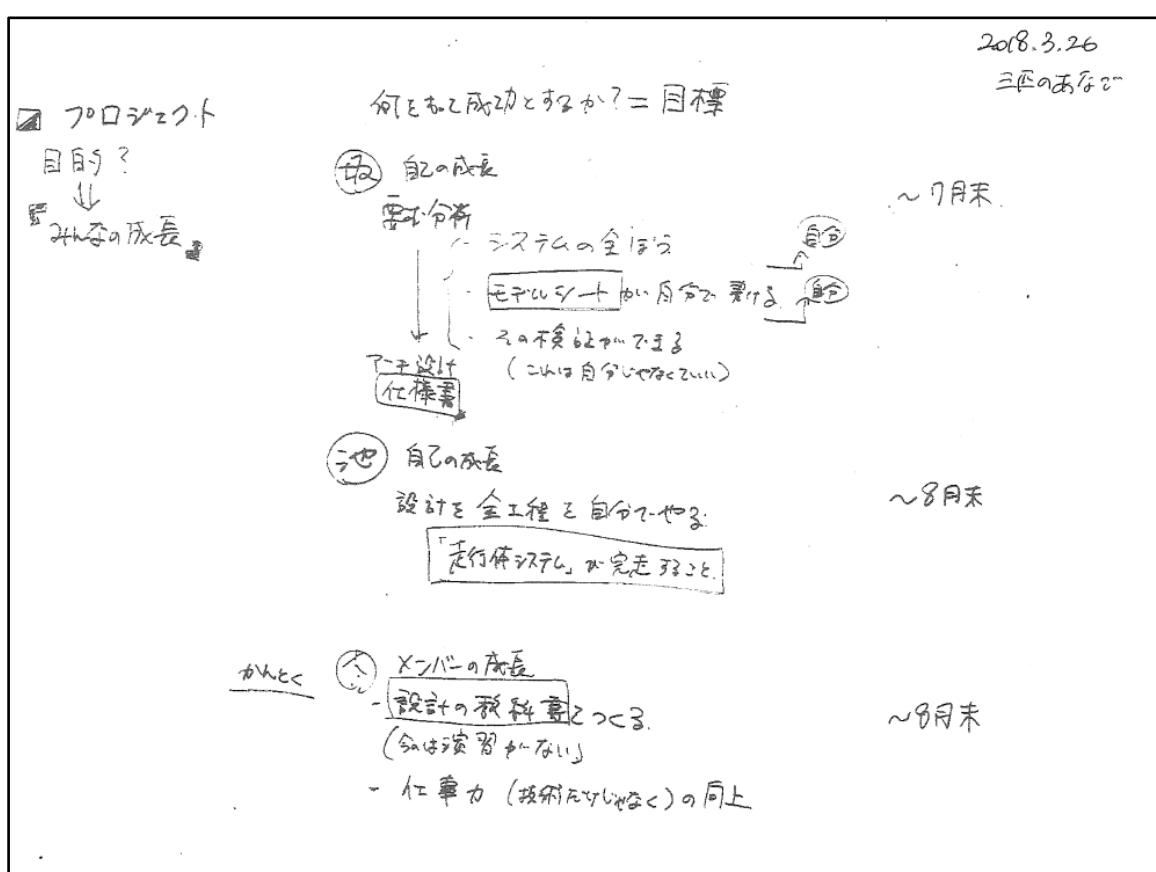
あなごチームはウォーターフォールでいく。  
他はリソース、スキル的に無理

## 2.2 三匹のあなたプロジェクトの目的と成功の条件

プロジェクトの目的：「みんなの成長」

プロジェクトの目標

目標	期日	成果物	主担当
要求分析からアーキテクチャ設計までの設計技術を習得する	7月末	モデルシート	坂下
設計の全行程を経験し、習得する	8月末	走行体システム	池上
設計の教科書をつくる	8月末	設計の教科書(この資料)	太田
目標工数の中で作業する 一人ひと月 10 時間かな～	毎月	工数管理表	太田



## 2.2.1 プロジェクト・メンバー

役職	氏名
平あなご	坂下さん
平あなご	池上さん
平あなご	おおた

## 2.3 成果物

### 成果物

成果物	備考
モデルシート	<a href="https://drive.google.com/open?id=16wBwfPEgEKD156LtEB9cFSSb-QWC328h">https://drive.google.com/open?id=16wBwfPEgEKD156LtEB9cFSSb-QWC328h</a>
走行体システム	割愛
設計の教科書	<a href="https://drive.google.com/open?id=1mGI_3Bx6igpEds4LR2FzeaMXsiatO3Z8">https://drive.google.com/open?id=1mGI_3Bx6igpEds4LR2FzeaMXsiatO3Z8</a>

## 2.4 プロジェクト・マネジメント

### 2.4.1 プロジェクト・マネジメントの基礎概念

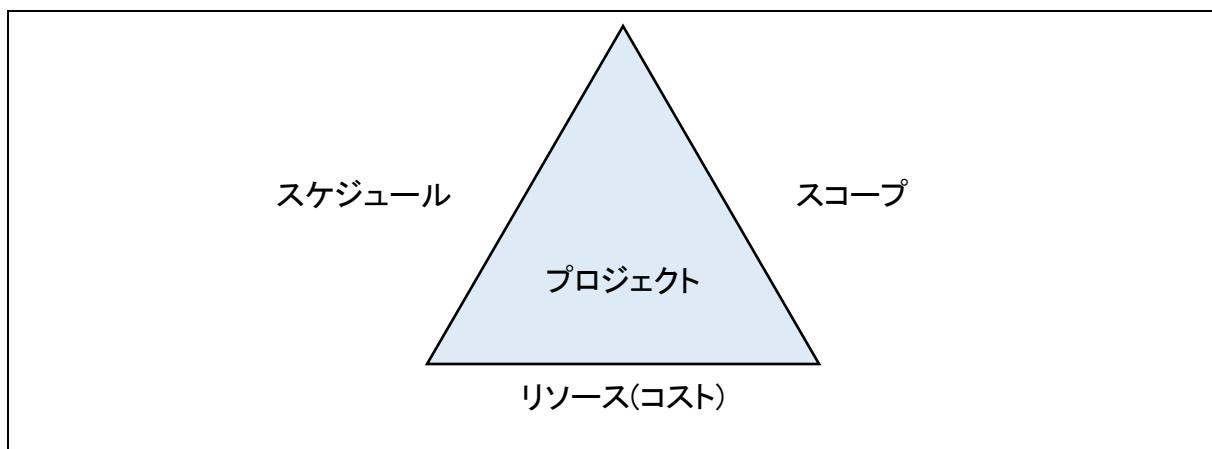
プロジェクトとは「スコープ」「リソース(コスト)」「スケジュール」の3次元の世界を目的地に向かって飛ぶ飛行機。プロジェクト・マネジメントはその飛行機を巡航する「仕組」であり「やり方」。

#### 飛行機の巡航システム

- ・飛ぶべき軌道と現在の軌道との“ズレ”をリアルタイムに検知する。
- ・検知した“ズレ”を数値化し制御機能にフィードバックする。
- ・フィードバックされた“ズレ”的大きさと方向に合わせて軌道修正のための対応を実施。

プロジェクトもそのプロジェクトの本来の正しい軌道に対する「スケジュール」「リソース(コスト)」「スコープ」の3次元上のズレを正確に数値化し巡航システムにフィードバックしなければならない。ズレを数値化する手段の一つにEVM(Earned Value Management)がある(参考文献①)。異なる次元の数値は比較できない。例えばシステム設計の進捗1%とシステム検証の進捗1%では作業量も作業の専門性も違う。比較できないということはスケジュール遅延リスクを数字で判断できない事を意味している。EVMではこの問題点を解決してくれる。

プロジェクトとは、「スケジュール」「リソース(コスト)」「スコープ」それぞれを一辺とする面積**一定**の三角形である。どれか一辺を長くしようとすれば、他の一辺または二辺を短くしなければならない(トレード・オフ・トライアングル)。プロジェクトの大失敗の原因の多くは経営判断の誤り。



複雑なプロジェクトも分解すれば、単純な作業の積み重ねになる。

*The secret of getting ahead is getting started. The secret of getting started is breaking your complex overwhelming tasks into small manageable tasks, and then starting on the first one.*

*Mark Twain*

単純な問題について時間をかけて対応することはない。複雑なままで対応しようとしても、何からどう進めれば良いか、整理できない。管理できるレベルに分解し現実的な管理を行う。

Work Breakdown Structure(WBS)

仕事(作業)を“構造的に”“漏れ無くダブリ無く”(MECE)なく分割し、整理する。

WBSを作ることは WBS のドキュメントを作ることではなく、WBS を全メンバーがコミットすること、つまり“やれる”と思えること。大事なことは形式でもツールでもなくやれると思えること。

## 2.4.2 プロジェクト・マネジメントのプロセス

PMBOK の知識エリアと各プロセスの成果物

知識エリア プロセス群	立ち上げ	計画	実行	監視 コントロール	終決
4.統合 マネジメント	4.1 プロジェクト 憲章作成	4.2 プロジェクト マネジメント計画書作成	4.3 プロジェクト実行 の指揮マネジメント	4.4 プロジェクト作業の 監視・コントロール 4.5 統合変更管理	4.6 プロジェクトや フェーズの集結
5.スコープ・ マネジメント		5.1 要求事項収集 5.2 スコープ定義 5.3 WBS 作成		5.4 スコープ検証 5.5 スコープコントロール	
6.タイム・ マネジメント		6.1 アクティビティ定義 6.2 アクティビティ順序設定 6.3 アクティビティ資源見積 6.4 アクティビティ所要期間 見積 6.5 スケジュール作成		6.6 スケジュール・コント ロール	
7.コスト・ マネジメント		7.1 コスト見積 7.2 予算設定		7.3 コスト・コントロール	
8.品質マネジメント		8.1 品質計画	8.2 品質保証	8.3 品質管理	
9.人的資源マネジメント		9.1 人的資源計画書作成	9.2 プロジェクト・チ ーム編成 9.3 プロジェクト・チ ーム育成 9.4 プロジェクト・チ ームのマネジメント		
10.コミュニケーション・ マネジメント	10.1 ステークホ ルダー特定	10.2 コミュニケーション計画	10.3 情報配布 10.4 ステークホルダ ーの期待のマネジ メント	10.5 実績報告	
11.リスク・マネジメント		11.1 リスク・マネジメント計画 11.2 リスク特定 11.3 定性的リスク分析 11.4 定量的リスク分析 11.5 リスク対応計画		11.6 リスクの監視・ コントロール	
12.調達マネジメント		12.1 調達計画	12.2 調達実行	12.3 調達管理	12.4 調達終結

### 2.4.3 ものごとの重要度について

すべての活動は、緊急度と重要度という2つの軸によって4つの領域に分けることが出来る。大きな目標のために役立つ行動や、意義のある重要なことが私たちに働きかけることではない。それらは「緊急」の事柄ではないからだ。自ら働きかけなければならぬものである。

	緊急	緊急ではない
重要	<b>第一領域</b> <ul style="list-style-type: none"> <li>・差し迫った問題</li> <li>・締め切りのある仕事、会議</li> <li>・品質不具合の対応</li> </ul>	<b>第二領域</b> <ul style="list-style-type: none"> <li>・信頼関係の構築</li> <li>・戦略の立案、準備や計画</li> <li>・予防</li> <li>・価値観の明確化</li> <li>・勉強や自己啓発</li> <li>・人材育成</li> </ul>
重要ではない	<b>第三領域</b> <ul style="list-style-type: none"> <li>・重要でない電話</li> <li>・重要でない報告書</li> <li>・重要でない会議</li> <li>・重要でない差し迫った問題</li> <li>・みんながやっていること</li> <li>・突然の来訪</li> <li>・無意味な付き合い</li> </ul>	<b>第四領域</b> <ul style="list-style-type: none"> <li>・暇つぶし</li> <li>・意味のない活動</li> <li>・見せかけの仕事</li> <li>・現実逃避</li> <li>・待ち時間</li> </ul>

**一番大事**

第二領域が最も重要度の高い活動であるが、周りからは求められることは無い。自ら働きかけて実行していく必要がある。

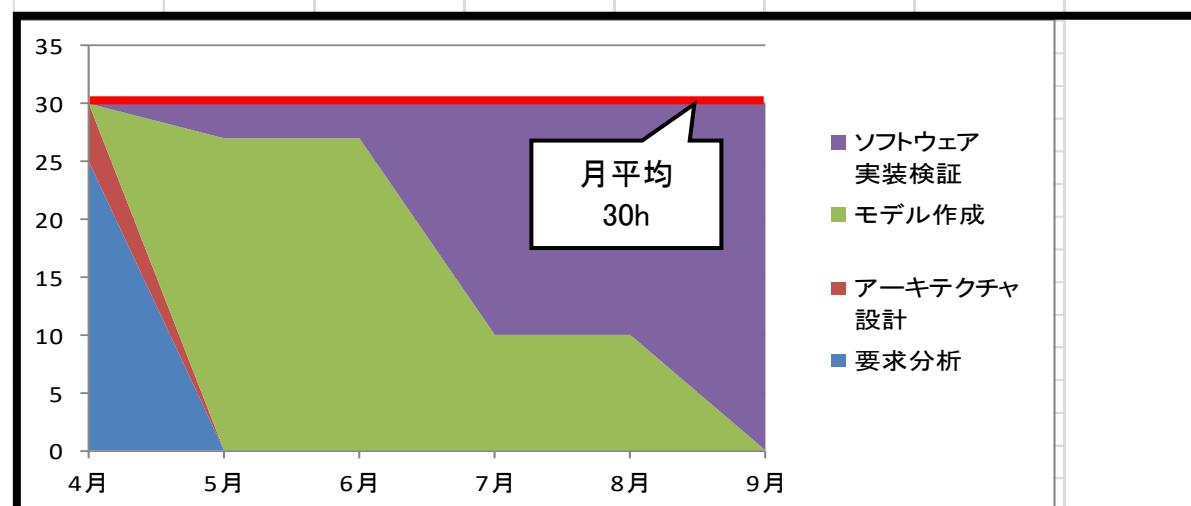
#### 2.4.4 三匹のあなたプロジェクトの開発工数の目標

月の工数を3人で30時間、一人平均10時間に抑える。(2017年は月60時間)  
4,5,6月からモデル作成を開始し、モデル作成の時間を増やす。

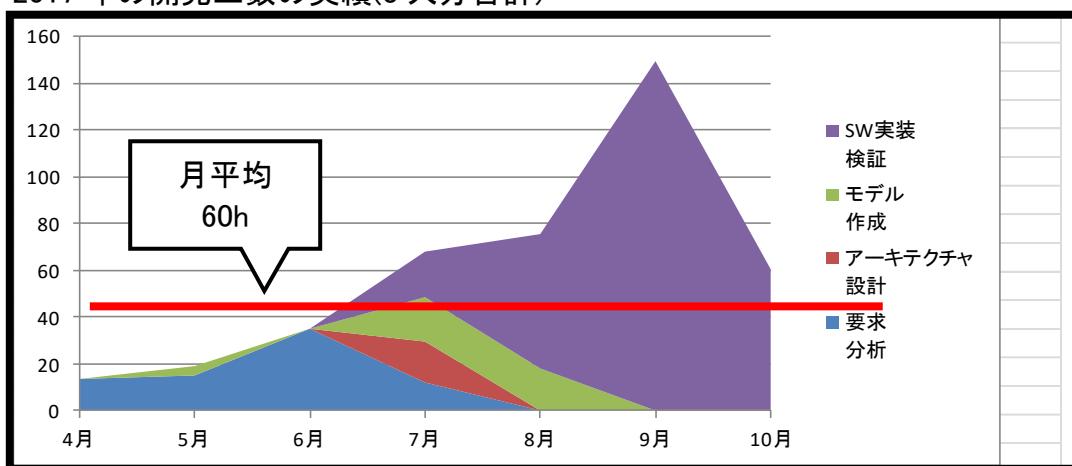
#### 2018年開発工数の目標(3人合計)

データのオリジナルと工数実績は最新の情報は 割愛

	4月	5月	6月	7月	8月	9月
要求分析	25	0	0	0	0	0
アーキテクチャ 設計	5	0	0	0	0	0
モデル作成	0	27	27	10	10	0
ソフトウェア 実装検証	0	3	3	20	20	30
月合計	30	30	30	30	30	30



#### 2017年の開発工数の実績(3人分合計)



#### 2017年の反省点

- ・ざっくと開発計画を立てていたが工数見積もり工数管理を行っていなかった。
- ・競技の結果を優先し、モデル作成が疎かになっていた。

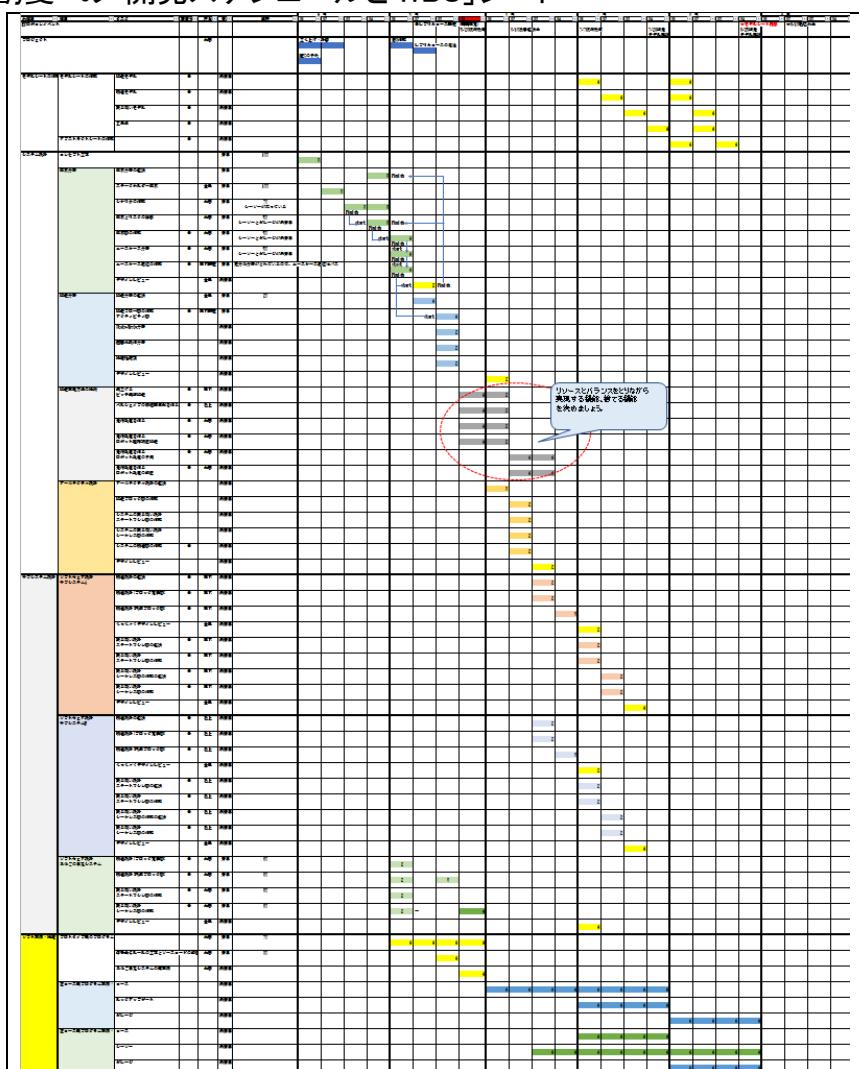
## 2.4.5 開発スケジュール

詳細なスケジュールは、割愛の「開発スケジュールとWBS」シートを参照。

	4月				5月				6月				7月				8月				9月				
	w1	w2	w3	w4	w1	w2	w3	w4	w1	w2	w3	w4	w1	w2	w3	w4	w1	w2	w3	w4	w1	w2	w3	w4	
ET ロボコン					●モデリングツール astah リリース								6/16 独自勉強会								★モデルシート提				
プロジェクト					●開発環境リリース?				●レプリカコース販売(期間限定)									7/28 試走			8/25 試走		★9/16 地区大		
システム設計	立ち上げ・計 8				WBS 作成 2				レプリカコース発 1					7/7 技術教育					★モデルシート完			★走行体システム		11月? 技術フォーラム 発表	
ソフト設計(モデル作成)	コンセプト立 5	要求分析 5	シナリオ作成 要求・リスク抽出 10		要求分析 5	アーキテクチャ設 5							リアルタイム OS 勉強会												
ソフト実装・検証					リスク対応案の検討 ロボット特性把握 10				機能モデル作 5	構造設計・モデル作 10			モデルシート作成 10				コンセプトシート作成 5							熟成	
					ロボット特性把握用のプログラム 5				振る舞い設計・モデル 10	デザインレビュー 5			デザインレビュー 10												
					実装・検証 20				実装・検証 10				実装・検証 10				システム検証 15								

## 2.4.6 あなごの WBS

~~割愛~~の「開発スケジュールと WBS」シート



## 2.4.7 あなごのトレードオフマトリックス

	柔軟性 低	柔軟性 中	柔軟性 高	備考
スコープ (機能)			●	プロジェクトの目的は機能の実現ではない。
スケジュー ル		●		機能を割り切る事で開発スケジュールは調整できる。 モデルシートの締切日、地区大会の開催日は決まっている。
資源	●			開発リソース 人 金は限られている

### 3 三匹のあなたごシステム

#### 3.1 システムとは、

システムとは、定義された目的を実現するために必要な要素の集まり。要素はソフトウェア、ハードウェア、人、サービスなどである。

*Systems are man-made, created and utilized to provide products or services in defined environments for the benefit of users and other stakeholders.*

ISO/IEC/IEEE 15288

*System is an integrated set of elements, subsystems, or assemblies that accomplish a defined objective. These elements include products(hardware,software,firmware),processes,people,information,techniques,facilities,services,and other support elements.*

INCOSE

#### 3.1.1 システムのライフサイクル

システムのライフサイクルを定義し、

- ・開発の流れを把握する
- ・競技だけではなく、ライフサイクル毎のシステムへの要求を洗い出し、仕様の抜け漏れを防ぐ
- ・開発に関わる全ての利害関係者を洗い出す

Generic life cycle (ISO/IEC/IEEE 15288:2015)

Concept Stage	Development Stage	Production Stage	Utilization Stage	Retirement Stage
			Support Stage	

#### 3.1.2 システムのコンセプト

コンセプトを立案し、成果物のイメージを明確にする。次にシステムの目的を達成するために実現したい振る舞いを明らかにする。

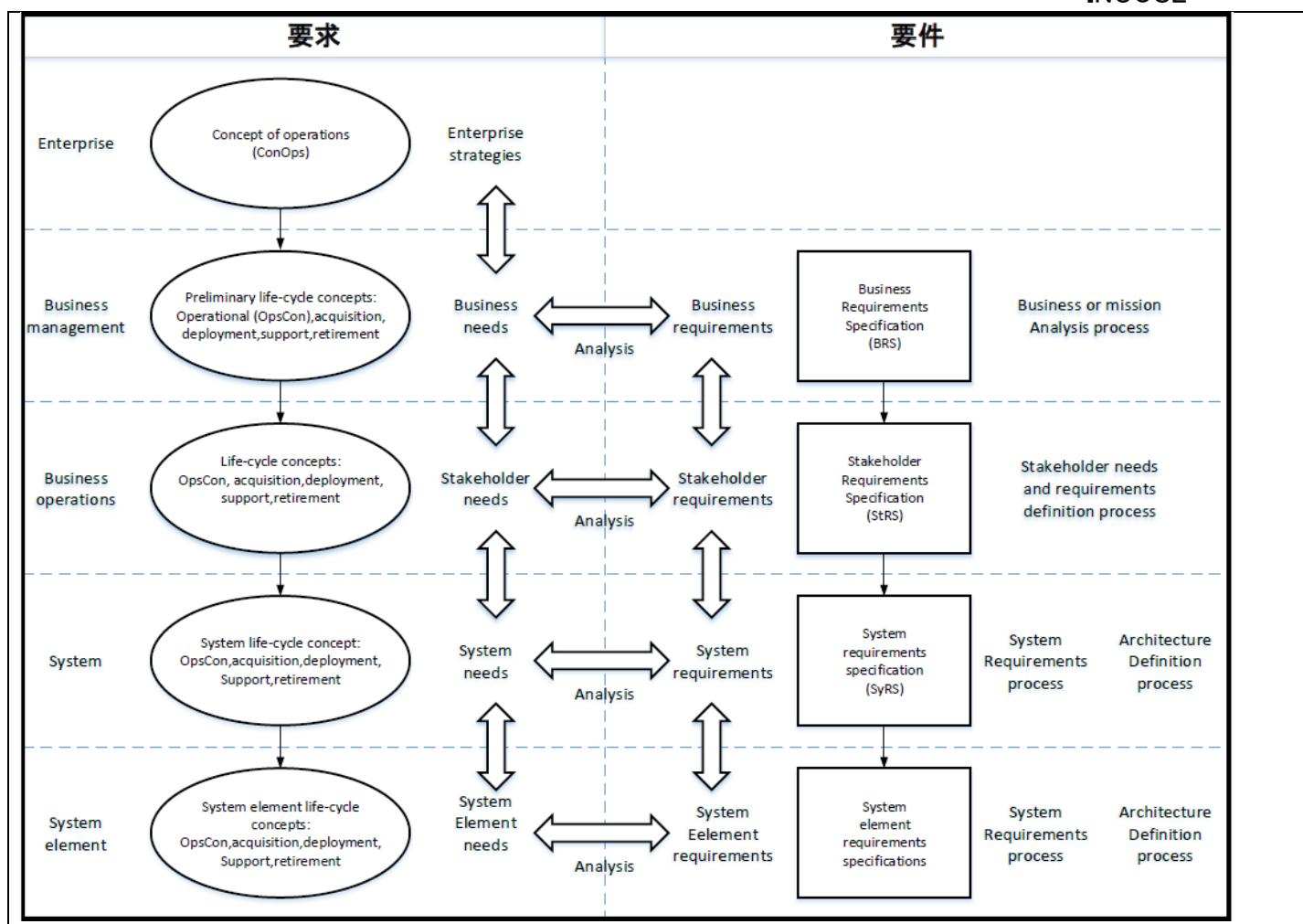
### 3.1.3 Systems Engineering のテクニカル・プロセス

*The Technical Processes are used to define the requirements for system, to transform the requirements into a effective product, to permit consistent reproduction of the product where necessary, to use the product to provide the required services, to sustain the provision of those services and to dispose of the product when it is retired from service.*

ISO/IEC/IEEE 15288

*Technical processes enable systems engineers to coordinate the interactions between engineering specialists, other engineering disciplines, system stakeholders and operators, and manufacturing. They also address confrmance with the expectations and legislated requirements of society. These processes lead to the creation of a sufficient set of requirements and resulting system solutions that address the desired capabilities within the bounds of performance, environment,external interfaces, and design constraints.*

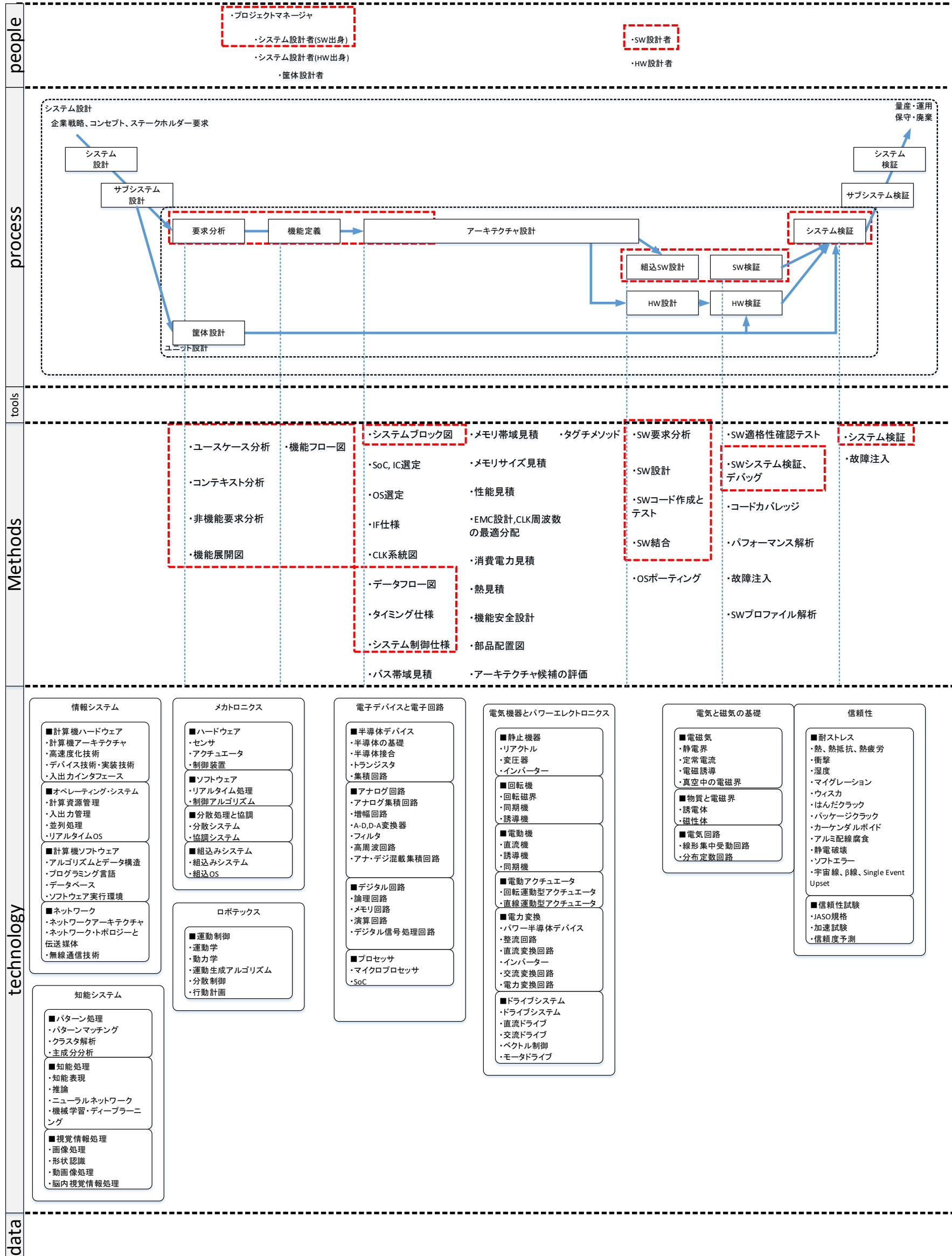
INCOSE



### 3.1.4 開発プロセスと手法

開発・設計に必要な人、プロセス、ツール、手法、技術、情報を下記に示す。

ET ロボコンで経験できる箇所



### 3.2 あなごシステムの概要

#### 3.2.1 三匹のあなごシステムの目的と目標

目的:「お客様を魅了する」

目標①:難所をクリアして完走する。

120 秒以内にガレージの前までクリアする。

ゴールまで 40 秒、LUG とシーソーは 80 秒以内でクリアする。

リモートスタートする

目標リザルトタイム:16 秒

数値化されていますか?

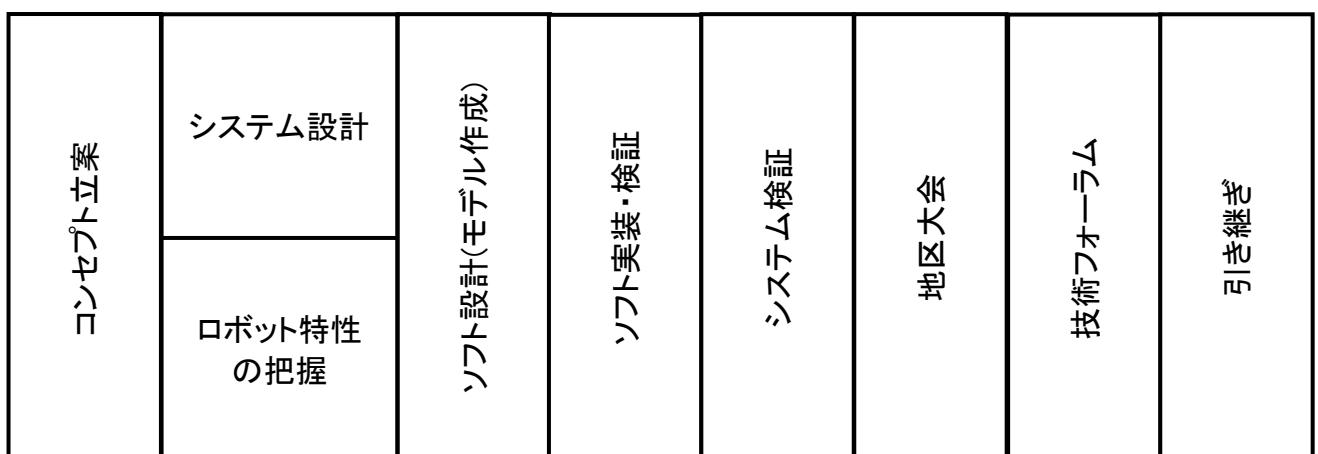
根拠は明確ですか?

何を、どれだけ達成しますか?

目標②:成功率 100%

目標③:理想の G 波形(ベルシェイプ)で走る。前後、横の躍度 0 で走行する

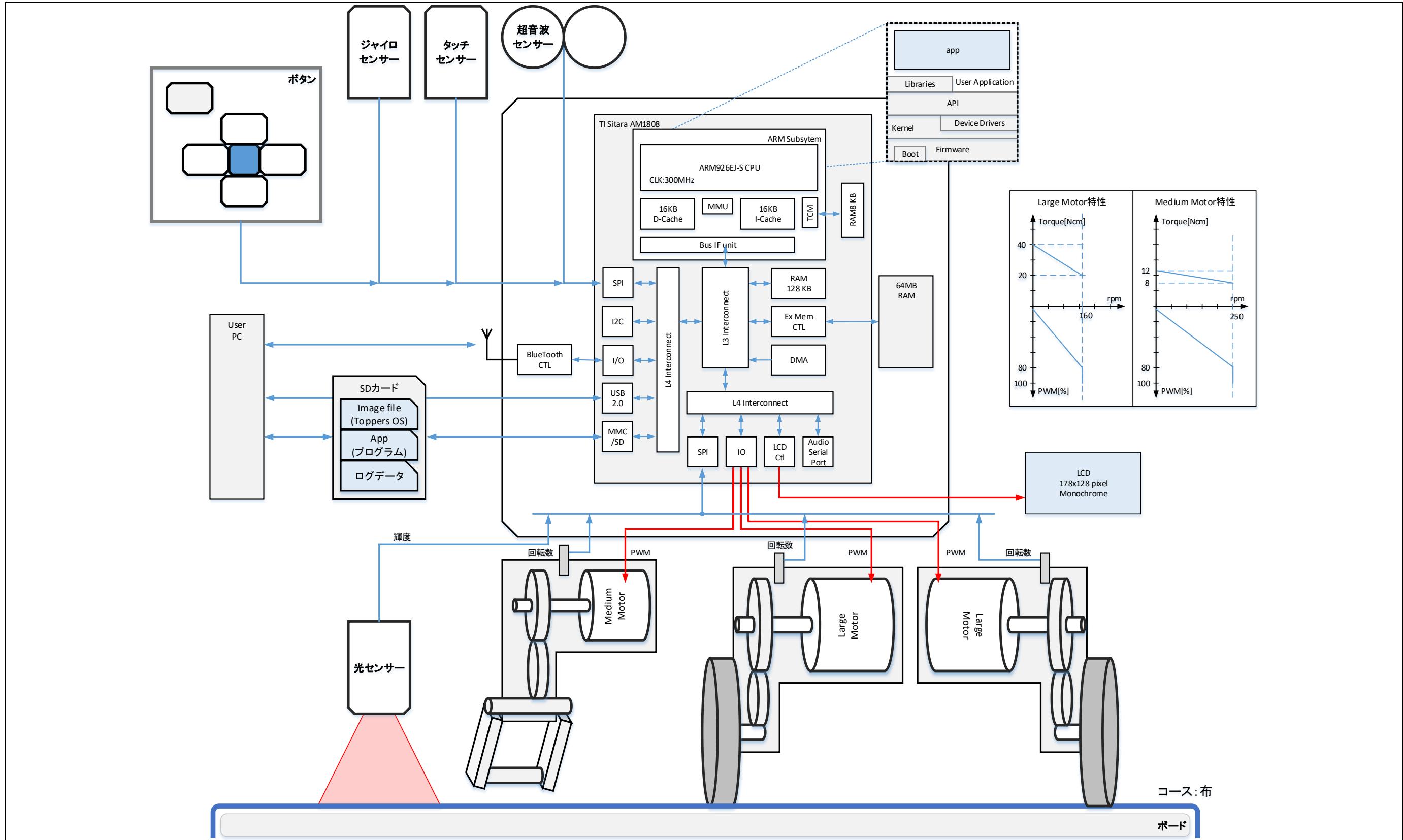
#### 3.2.2 システムのライフサイクル



#### 3.2.3 ステークホルダー

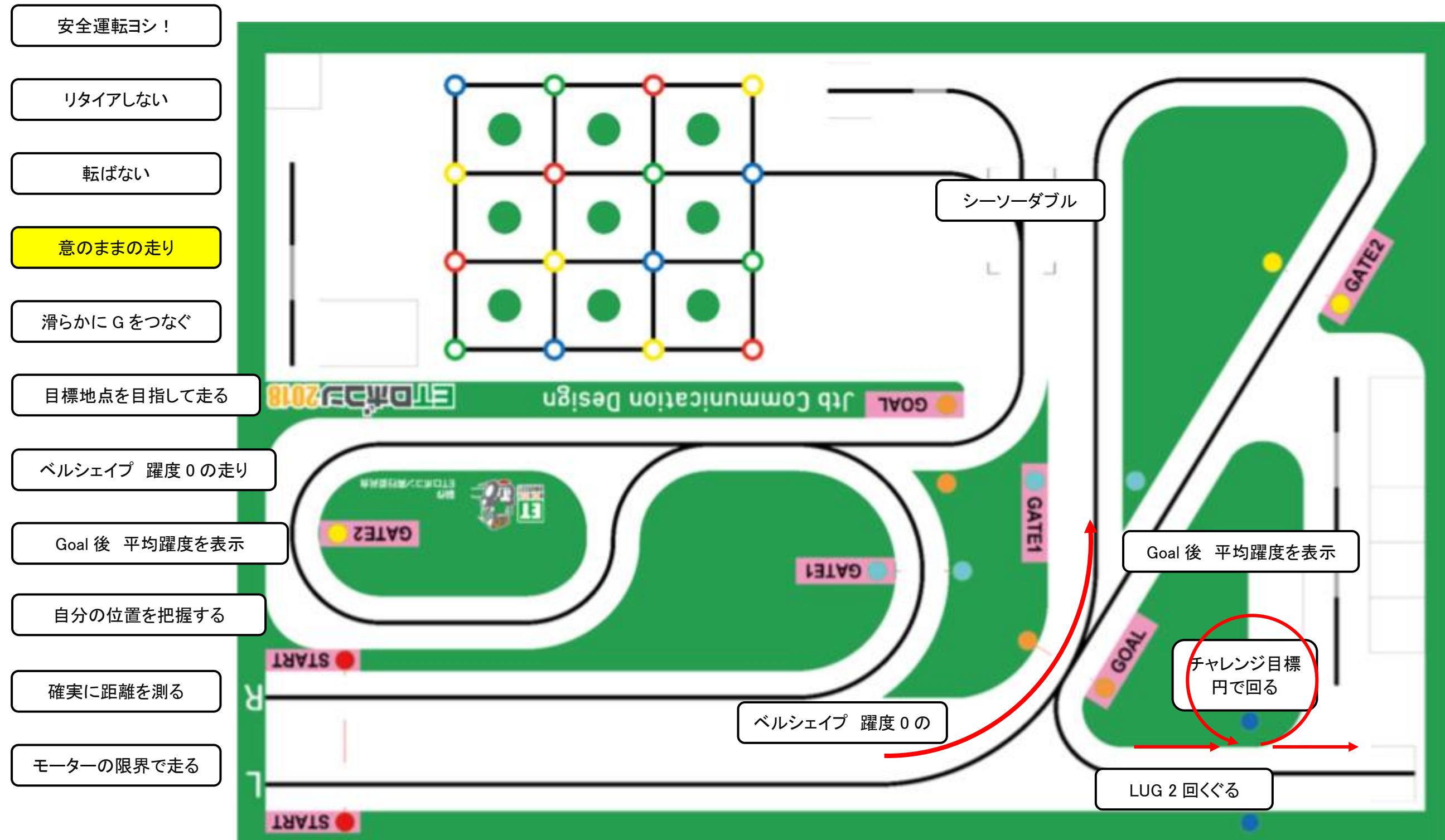
ステークホルダー	何が欲しいのか
坂下さん	割愛
池上さん	バイク 車 ゲーム
太田	みんなの平和 みんなの希望 みんなの夢
上司	
来年の参加者	
今年の参加者	
大会関係者	

### 3.2.4 システム構成



### 3.2.5 コンセプト

システムの目的	お客さんを魅了する
目標①	ゴールまで 40 秒、LUG とシーソーは 80 秒以内でクリアする。
目標②	成功率 100%
目標③	理想の G 波形(ベルシェイプ)で走る 跳度 0 で走行する

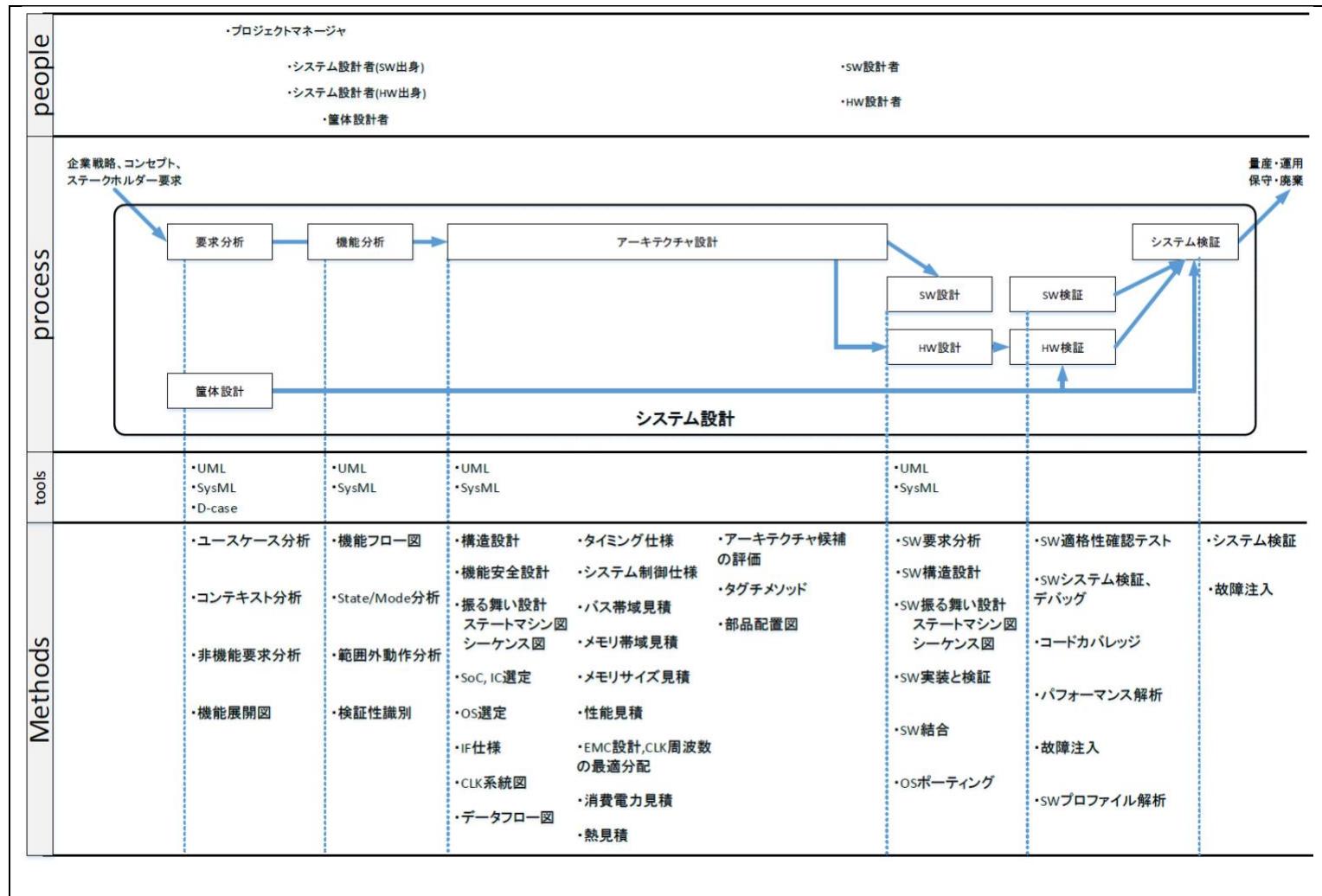


## 4 システム設計

### 4.1 システム設計とは

ステークホルダの要求を要求分析しシステム要件を定義、要件を満たすアーキテクチャ設計を行い、サブシステムへの要求（システム仕様書）を導く活動。

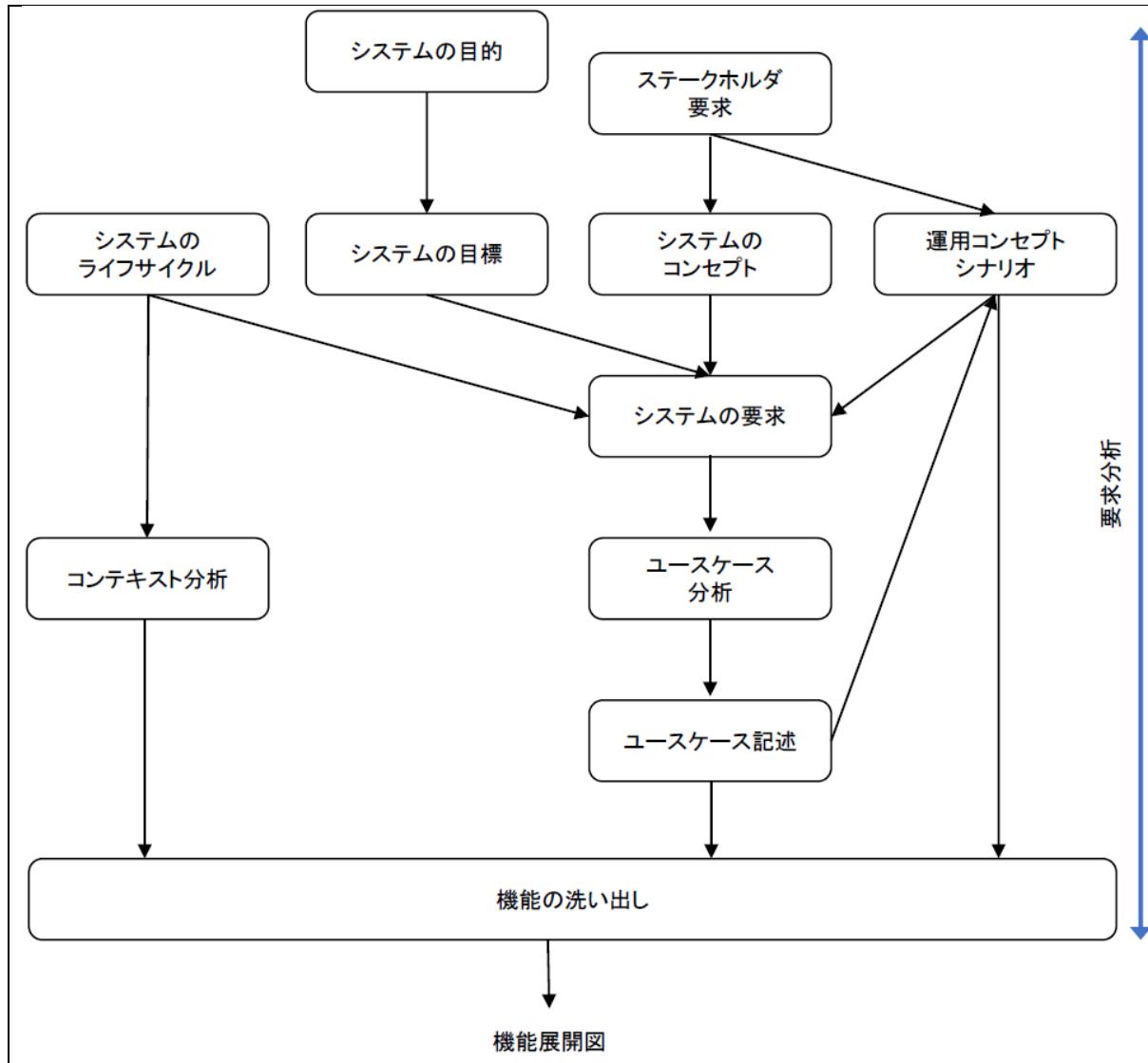
下記にシステム設計のプロセスと手法を示す。



## 4.2 要求分析

要求分析とは、ステークホルダの要求をシステムの要件へと変換する作業である。  
システムの境界と成果物を明確にする作業である。

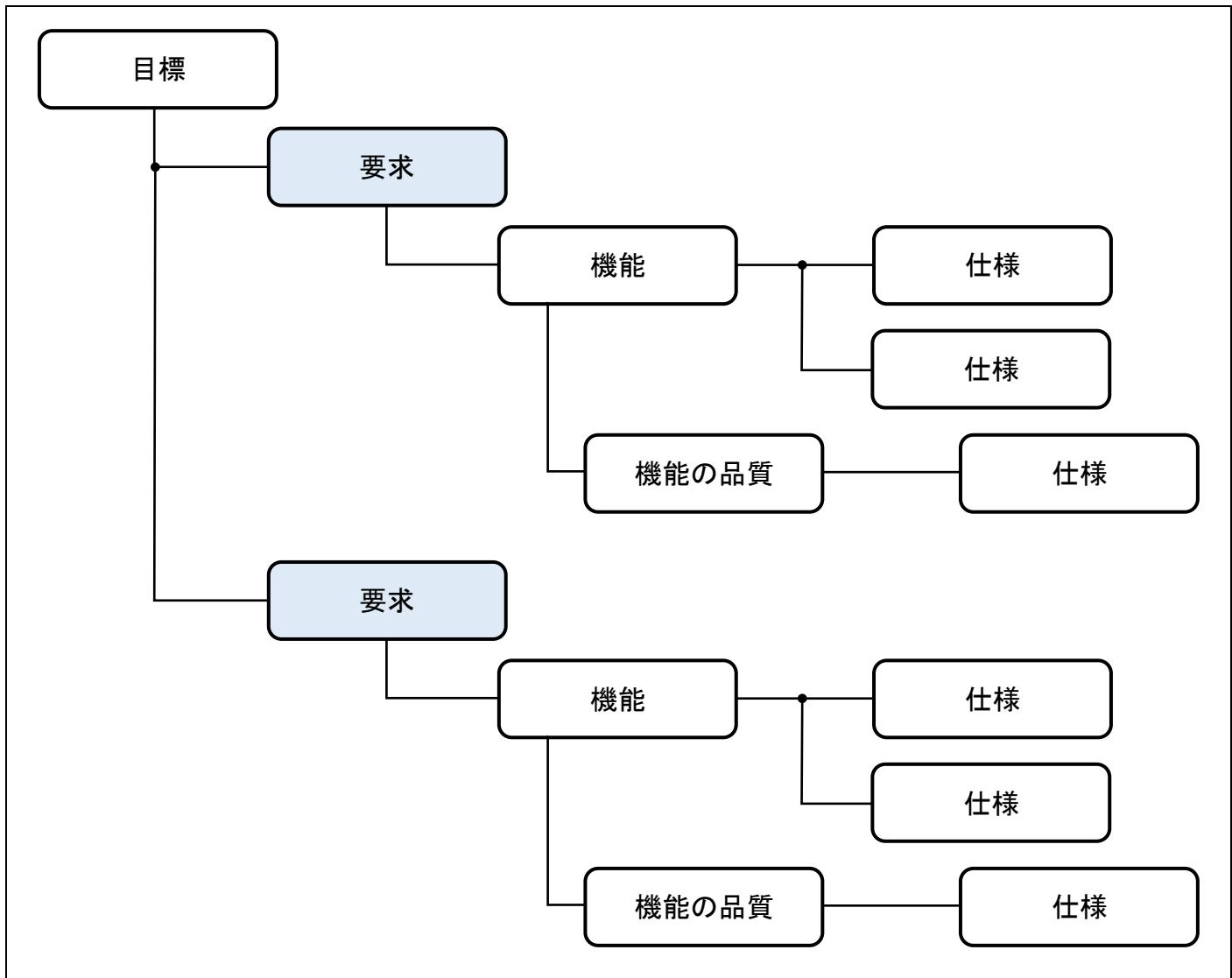
下記に要求分析のプロセスを示す。



#### 4.2.1 要求とは

製品が「何を、どれほどよく、どういった状態で」、与えられた目的を達成するかを決めるもの合意、計画、仕様のような標準となる文章

ANSI/EIA-632



## 要求の種類

要求	説明
機能	システムが目的を実現するために持つべき機能に対する要求
性能	システムが目的を実現するために持つべき性能に対する要求
外部インターフェース	システムと外界とのインターフェースに対する要求
環境	システムが動作する環境に関する要求
リソース	システムが目的を実現するためにリソースに関する要求
物理的	システムに対する物理的要求(容積、サイズ、重量)
設計	設計基準の要求
プロセスと手法	製造プロセスおよび品質管理要求

## ISO9126:品質特性 機能要求と非機能要求（ソフトウェアの品質要求）

機能要求	非機能要求
機能性要求	信頼性要求 使用性要求 効率性要求 保守性要求 移植性要求

要求には機能要求と非機能要求がある。

機能要求はシステムが「何を」を実行する必要があるかを示す。

非機能要求は機能上の局面が「どの程度」達成されるかを示す。

## 要求の品質

要求の品質	説明
完全性	各要求は、当該要求を実現するために必要となる、制約および条件を含んだ、全ての情報を含んでいること
無矛盾性	要求は、矛盾が無いこと
トレース可能性	要求は、上位および下位要求がトレース可能であること
テスト可能性	要求は、検証可能であること
単一性	要求は、二重定義されていないこと

## 品質要素

品質要素	説明
当たり前品質	できていて当たり前、できなければ不満を引き起こすもの
一元的品質	できていれば満足、できていなければ不満を引き起こすもの
魅力的品質	できていれば満足、できていなくても仕方がないと受けとられるもの

## 製品品質特性(ISO/IEC25000 シリーズ)

製品品質特性	説明
機能適合性	機能が利用者のニーズにどれくらい合っているか
性能効率性	システムの実行時間が効率的か、資源の使い方が効率的か
互換性	ほかのシステムと適切につながるか、共存できるか
使用性	システムがどの程度使いやすいか
信頼性	不具合が起きない度合い
セキュリティ	情報資産へのアクセス制御、保全性を保つ度合い
保守性	システムの保守、修正が容易か
移植性	違う環境へシステムを移すことがどの程度容易にできるか

### 4.2.2 要求分析に用いられる図やモデル 要求分析の成果物

#### 要件定義書

- ・システムが持つ機能とその機能の仕様が明確になっている事。
- ・制約や性能・品質の要件が明確になっている事。

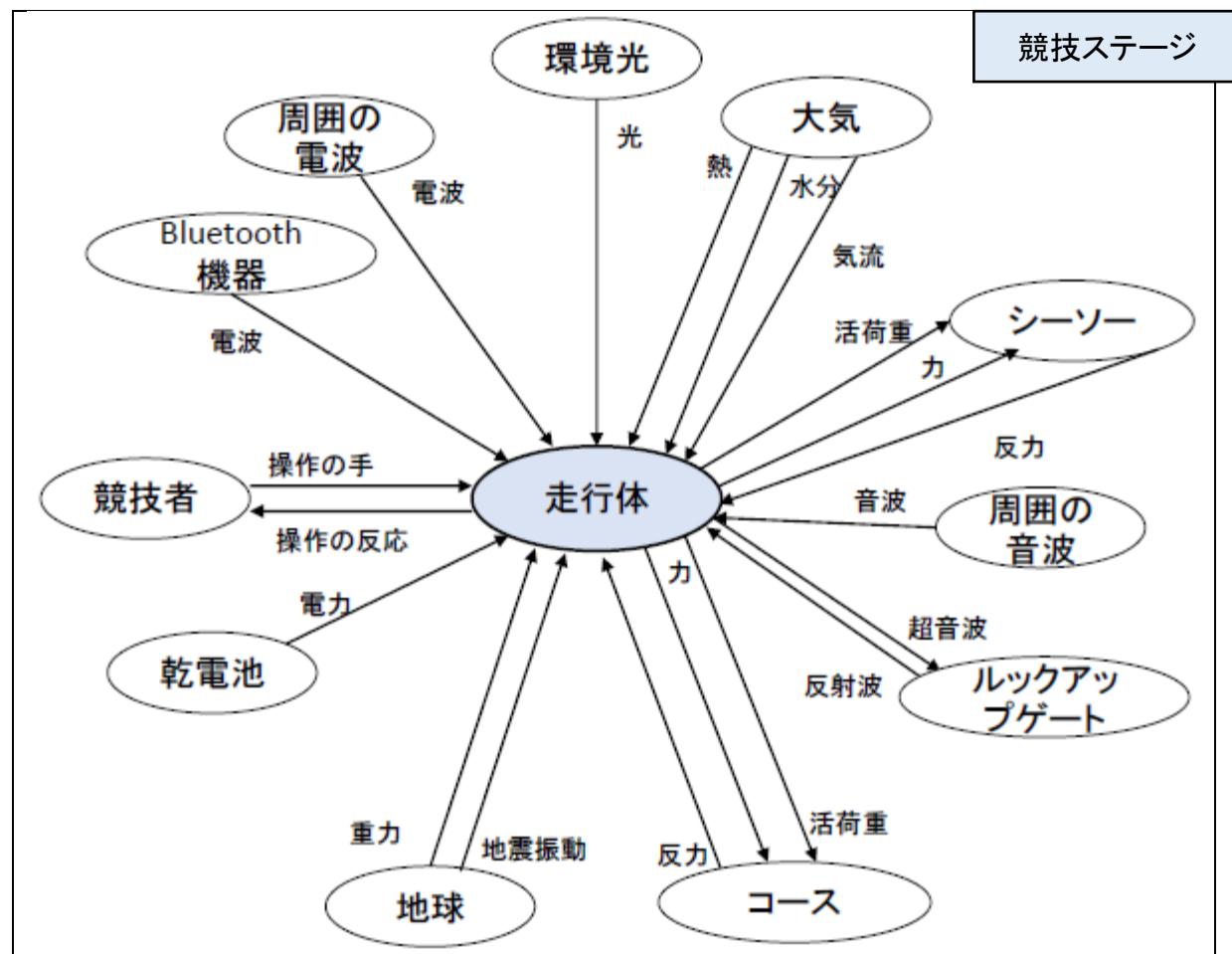
#### 用いられる図やモデル、表、記述法

項目	Systems Engineering (SysML)	UML
要求図	●	
D-CASE		
マインドマップ		
ユースケース図	●	●
ユースケース記述		
コンテキスト図	●	
シナリオ/基本フロー		
アクティビティ図	●	●
図解		
機能一覧表		
ミスユースケース図	●	●
ミスユースケース記述		

#### 4.2.3 コンテキスト分析

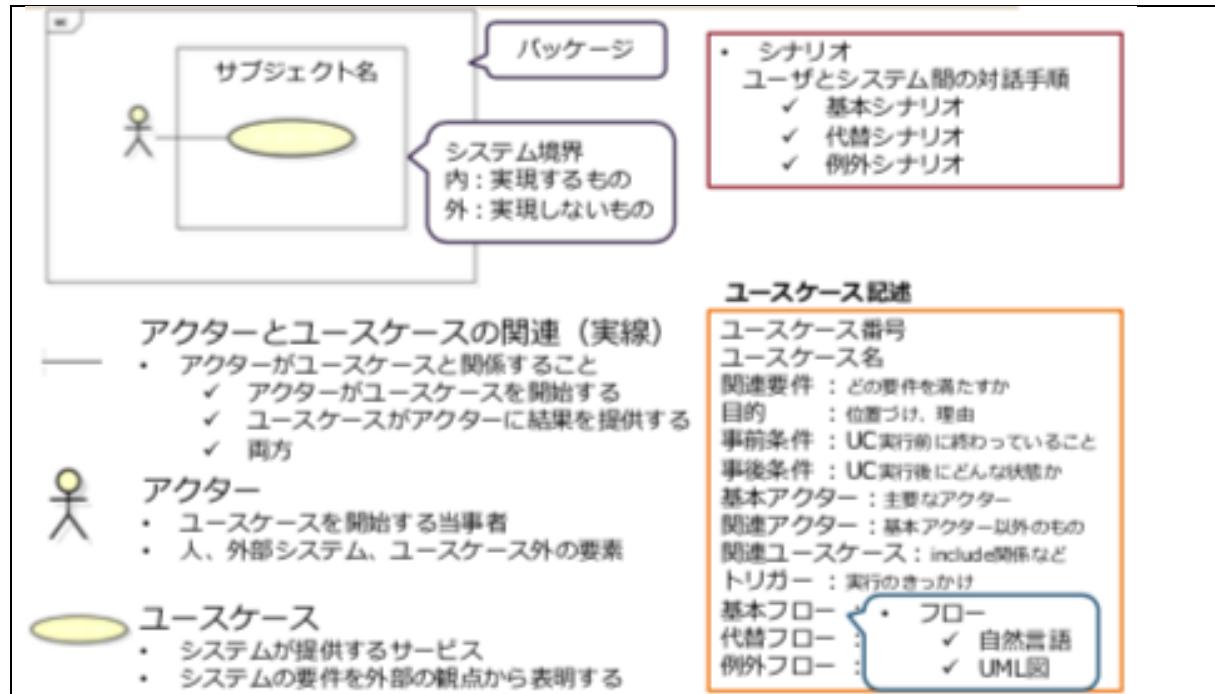
ライフサイクルのステップ毎にコンテキスト図を作成することで、外部から受ける影響、外部に与える影響を把握する。

- ・カテゴリごとにアイテムを抽出。 力学的、電気的、電磁気的、作業的、対環境など
- ・インターフェース条件を書き込む。



#### 4.2.4 ユースケース分析

システムが利用者に提供すべき働きやサービスをモデル化します。



#### ユースケース記述の例

UC名	光環境のキャリブレーションをする
概要	走行に必要な各種センサ値を取得し保存する
アクター	ユーザー
事前条件	ユーザーが走行するコースを選択していること
事後条件	白黒基準値に異常がなく保存が完了していること
基本系列	<ol style="list-style-type: none"> <li>1.ユーザーは、白線上に走行体を設置し、タッチセンサを押下する</li> <li>2.システムは、光センサ値を取得し白色基準値として保存する</li> <li>3.ユーザーは、黒線上に走行体を設置し、光センサを押下する</li> <li>4.システムは、光センサ値を取得し黒色基準値として保存する</li> <li>5.システムは、白色基準値が黒色基準値より大きいことを 確認する</li> </ol>
代替系列	プリセットされた白黒基準値を使用する
例外系列	基本系列5で白色基準値が黒色基準値以下であった場合、LEDやスピーカーを用いてユーザーに警告を行い、基本系列1に戻る

## 4.3 あなごシステムの要求分析

### 4.3.1 ステークホルダー要求

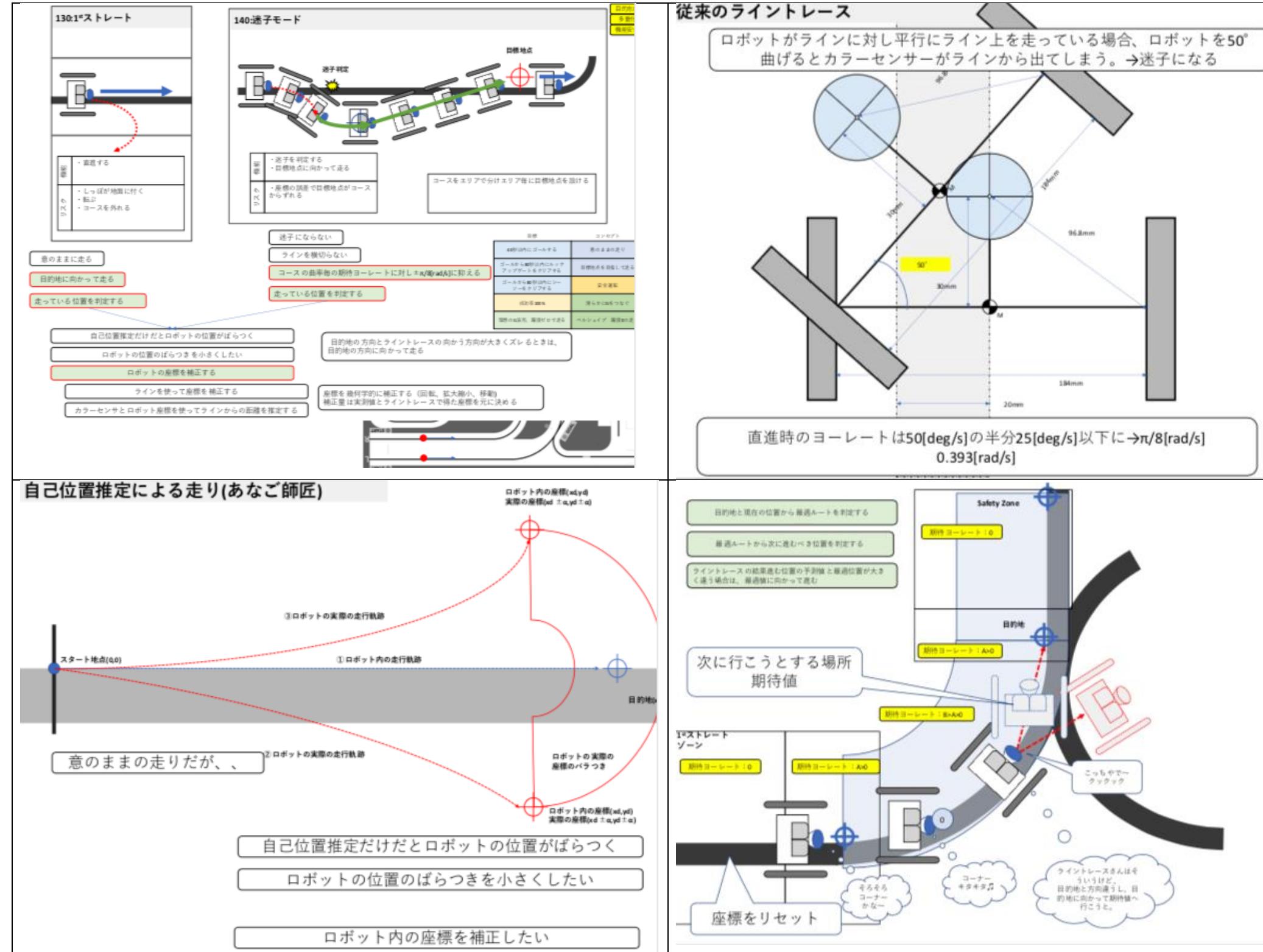
最新の情報は~~削除~~「ステークホルダー要求」シートを参照

一部を下記に示す(20180430 現在)

No	目標	コンセプト	ユースケース
1	40秒以内にゴールする ゴールから80秒以内にルックアップゲートをクリアする ゴールから80秒以内にシーソーをクリアする 成功率 100% 理想のG波形、躍度ゼロで走る	安全運転 意のままの走り 滑らかにGをつなぐ 目標地点を目指して走る ペルシェイプ 跳度0の走り ゴール後に平均躍度を表示する 自分の位置を把握する 確実に距離を測る モーターの限界で走る チャレンジ:LUGを円軌道で回る ライフサイクル	ステークホルダー要求
2		大会	転ばないで欲しい
3		大会	しっぽ倒立モードの時に布の巻き込みを検出したい(スリップ検出)
4		大会	タイヤのスリップを検出したい
5		大会	まっすぐ走って欲しい
6		大会	しっぽの動作を高速にしたい
7		大会	ロボットの座標を知りたい
8		大会	コースを外れないでほしい(グリーンにいかない)
9		大会	コースを外れた時にコースに復帰してほしい
10		大会	目標車輪速度でロボットを制御したい
11		大会	目標の位置にピタッと止まりたい
12		大会	ペルシェイプなG波形で走りたい
13		大会	ロボットの座標を補正したい
			コースの曲率に合わせて走って欲しい
			コースを走行する
			ルックアップゲートをクリアする
			シーソーをクリアする
			ガレージで止まる
			競技終了後にスタートラインに戻る
			競技前のキャリブレーション
			検証・デバック

### 4.3.2 あなごシステムの振る舞い(シナリオ)

あなごシステムの要求機能とリスク、リスク対策を抽出するために競技時のシナリオを作成した。最新の情報は割愛を参照。



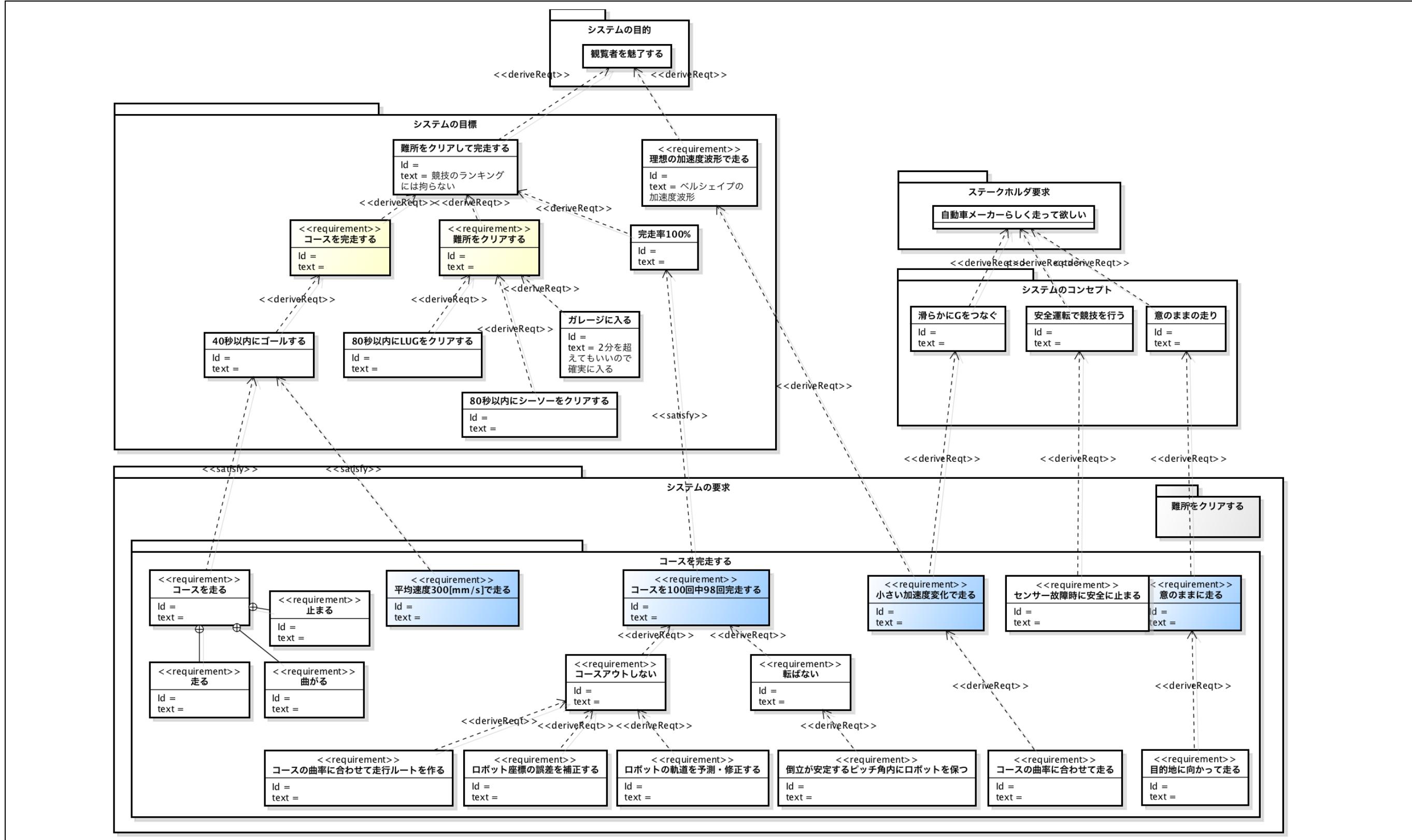
### 4.3.3 あなごシステムの要求

ステークホルダー要求、シナリオから抽出した要求を整理した。最新の情報は割愛の「機能要求」シートを参照

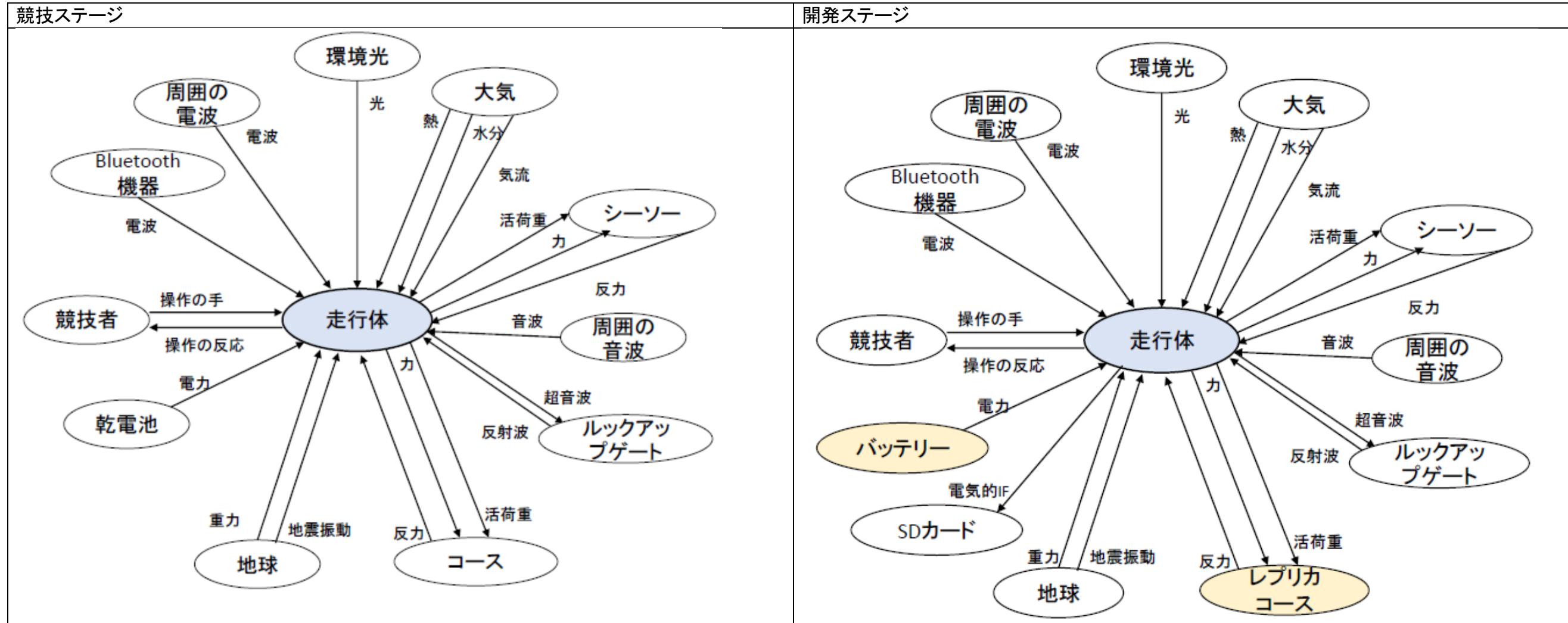
一部を下記に示す(20180430 現在)

機能要求(ユースケース)	1 次機能要求	2 次機能要求	非機能要求	1 次非機能要求	2 次非機能要求
コースを完走する	走り出す	走行体を立てる 二輪で倒立する 走り出す			
	コースを走る		意のままに走る	目的地に向かって走る	目的地とロボットの位置から最適な走行ルートを作る
			滑らかに走る	コースの曲率に合わせて走る	
			コースを 40 秒以内に完走する	平均速度 秒速〇〇mm で走る	
			コースを 100 回中 98 回完走する	コースアウトせずに走行	ラインの曲率に合わせた走行
				転ばない	G の立ち上げを滑らかにする
				地区大会迄に目標達成	開発を効率化

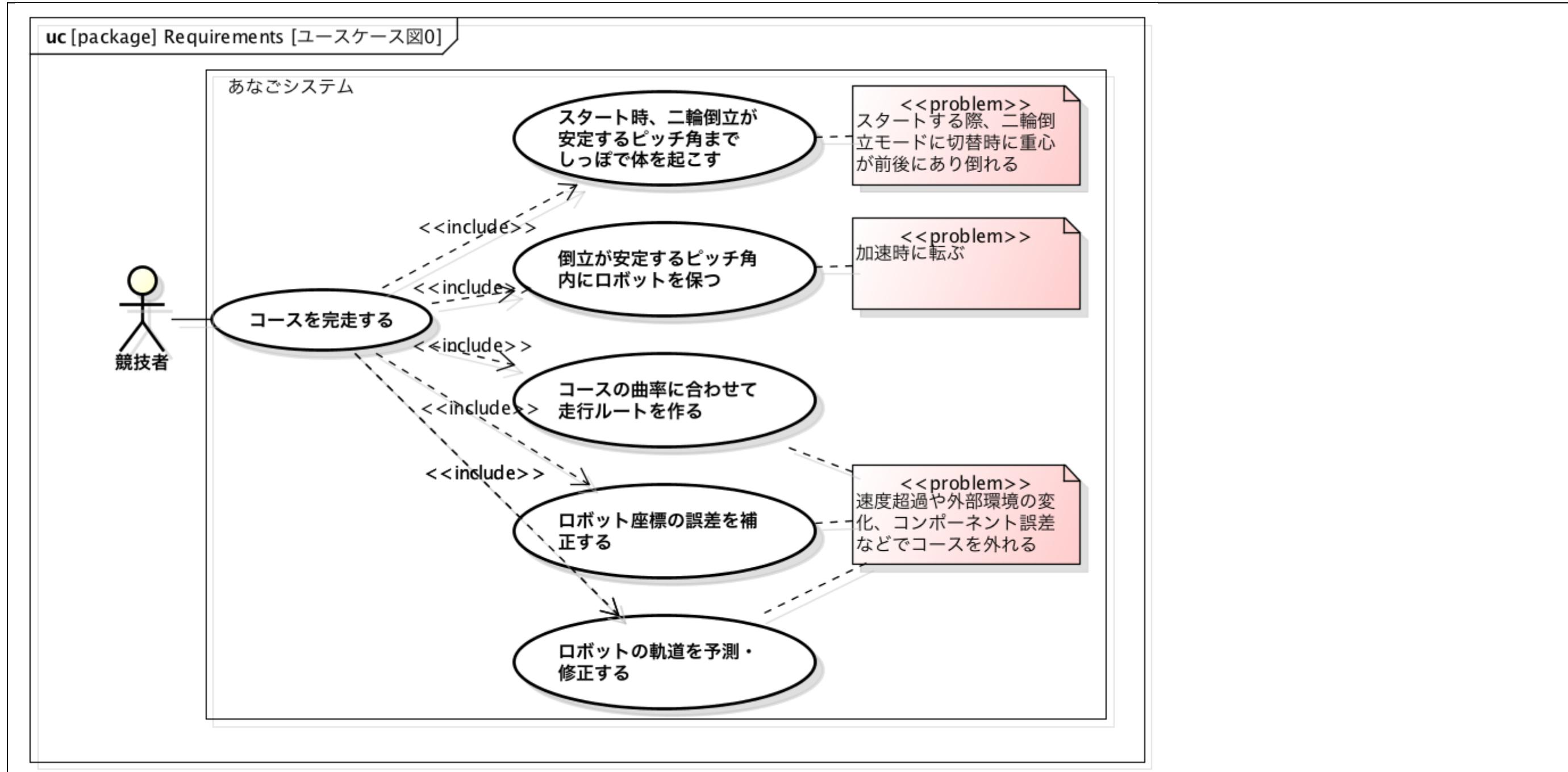
#### 4.3.4 あなごシステムの要求図(SysML)



#### 4.3.5 あなごシステムのコンテクスト分析

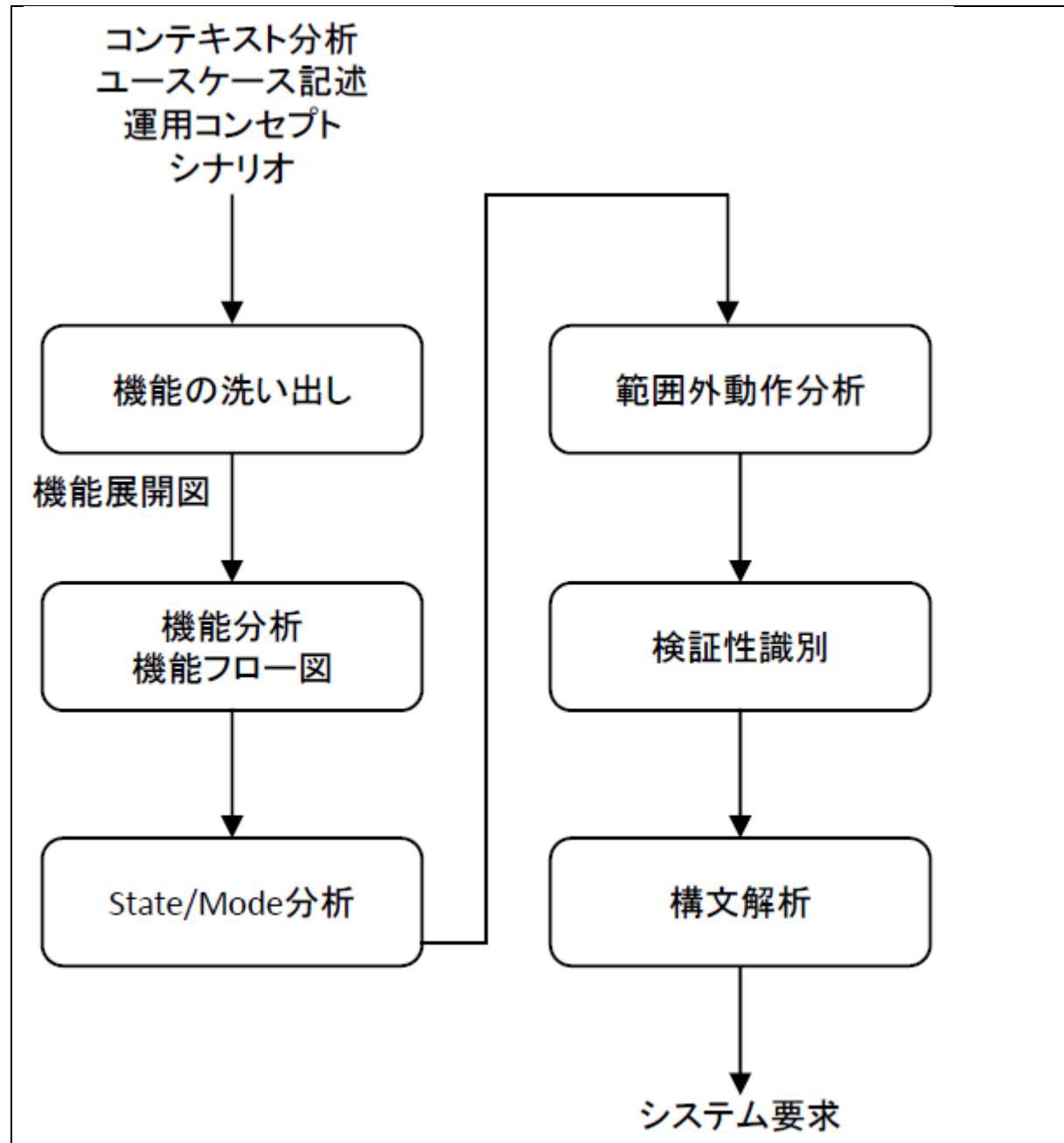


#### 4.3.6 あなごシステムのユースケース分析



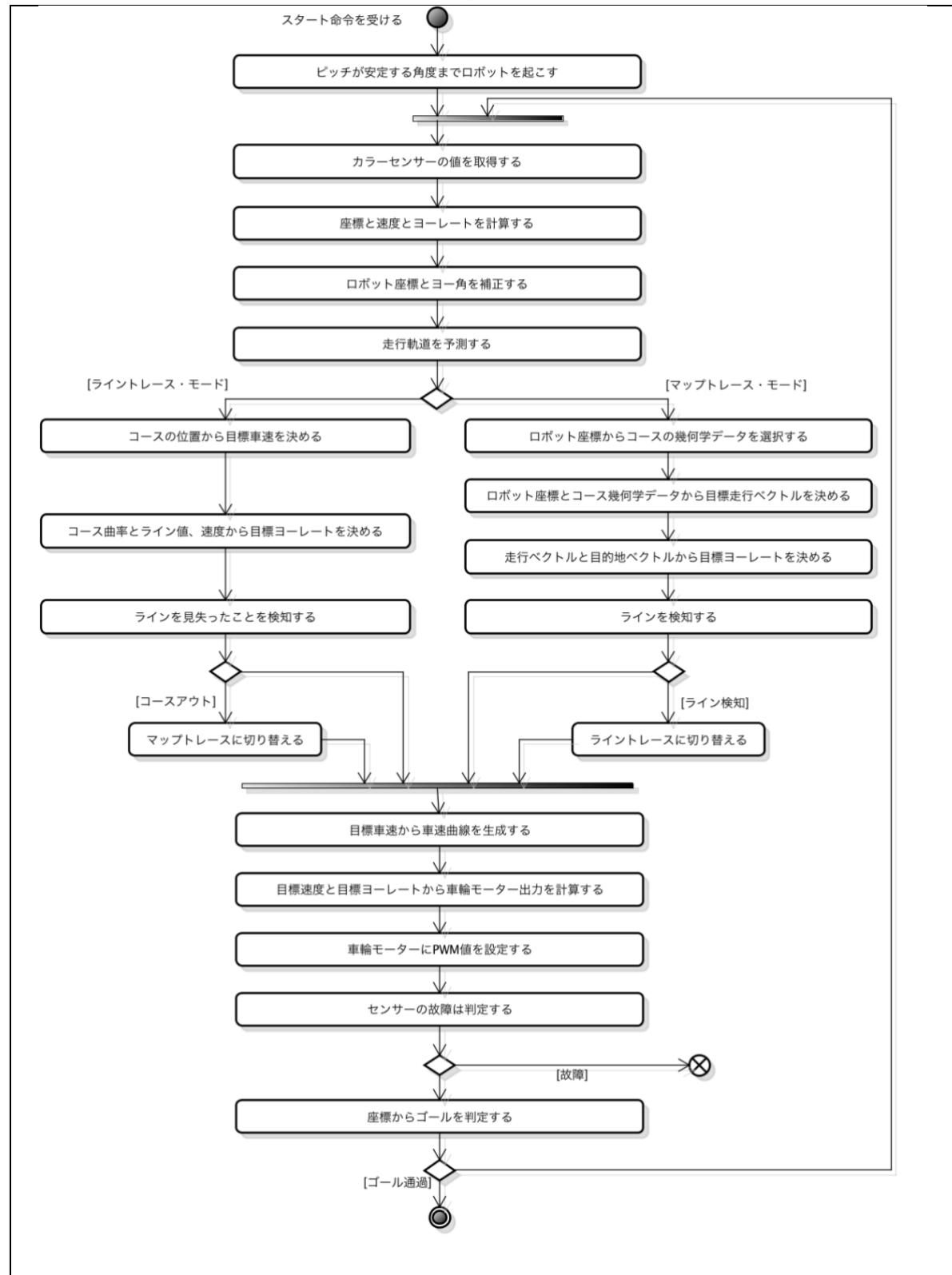
#### 4.4 機能分析

機能分析の流れ



## 4.5 あなごシステムの機能分析

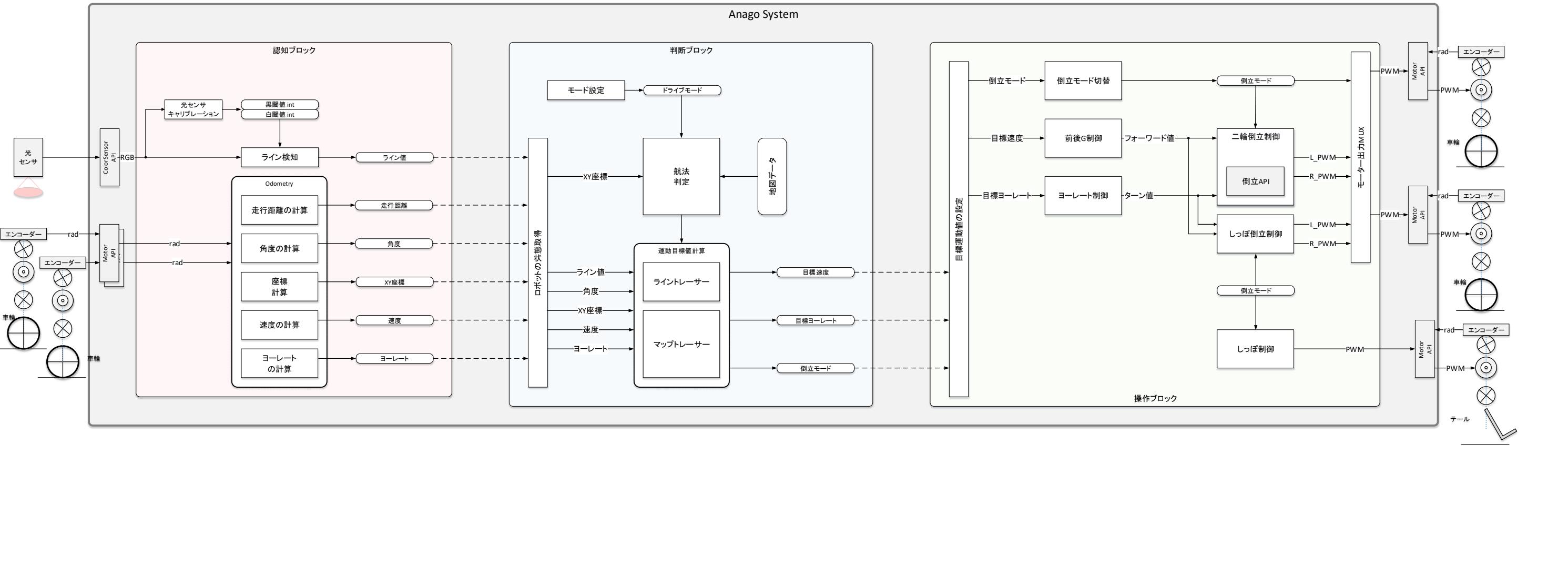
### 4.5.1 機能フロー図 「コースを完走する」



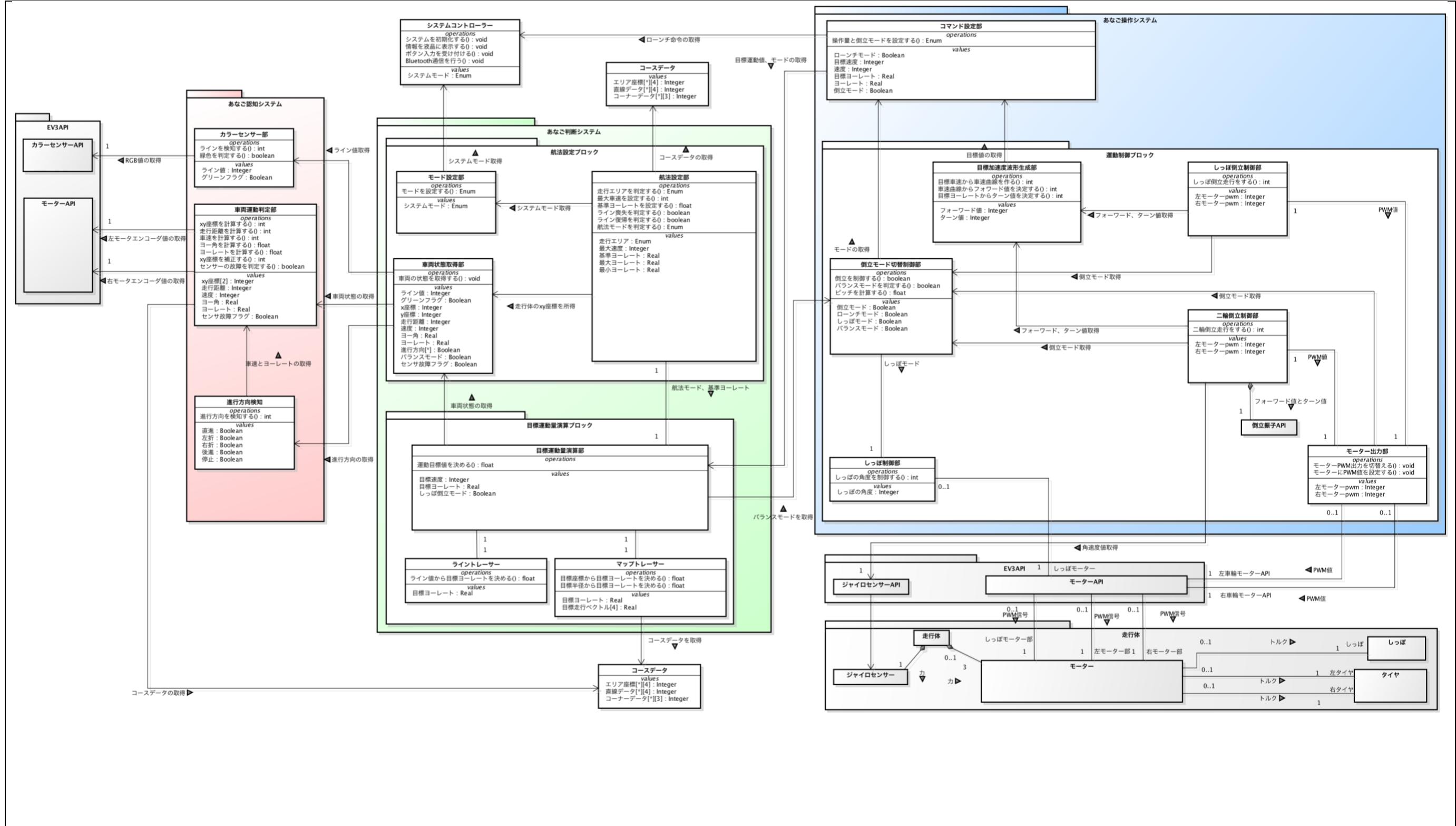
## 4.6 アーキテクチャ設計

### 4.6.1 機能ブロック図

Anago System Diagram  
Kaoru Ota

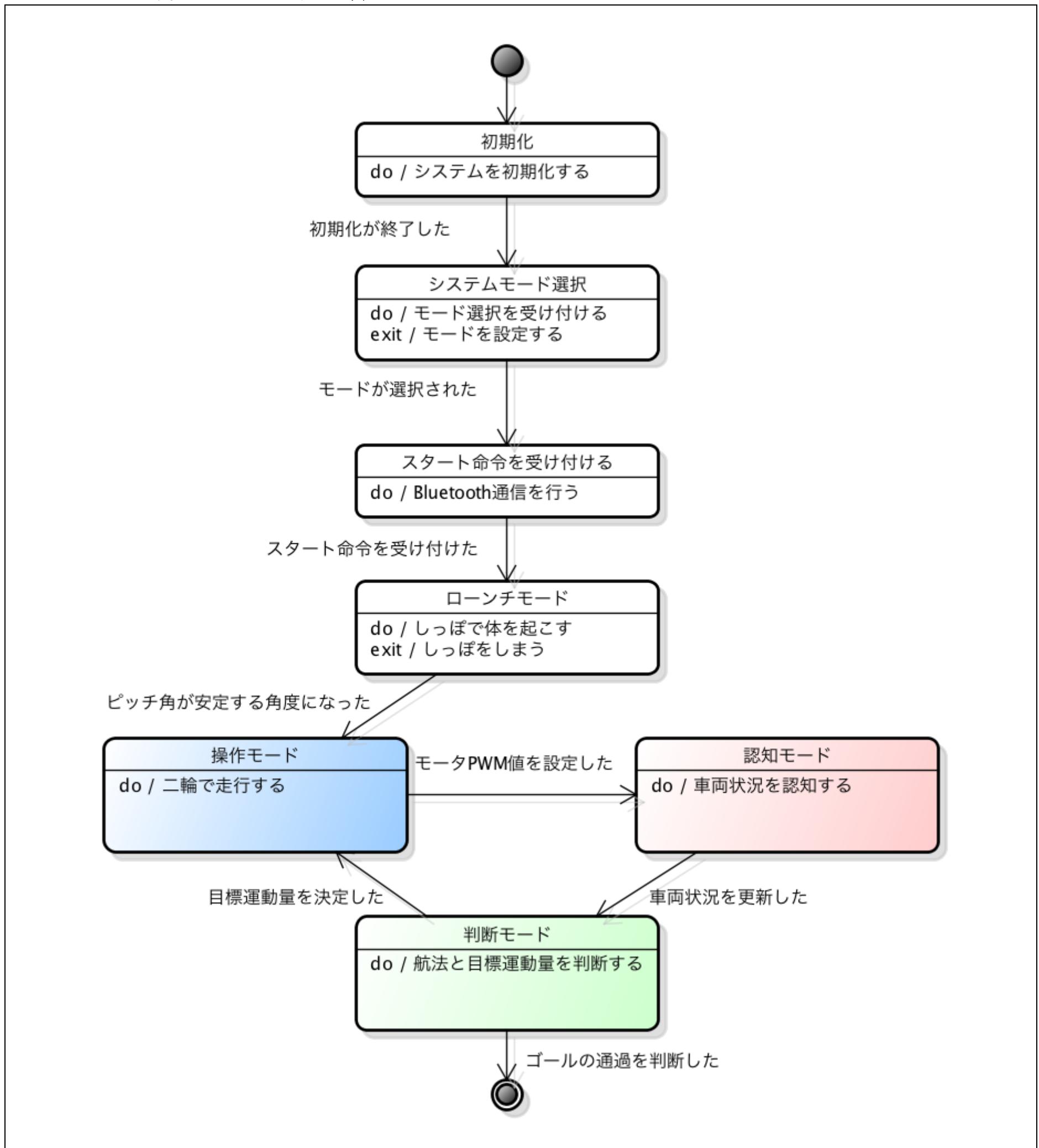


#### 4.6.2 システム構造図

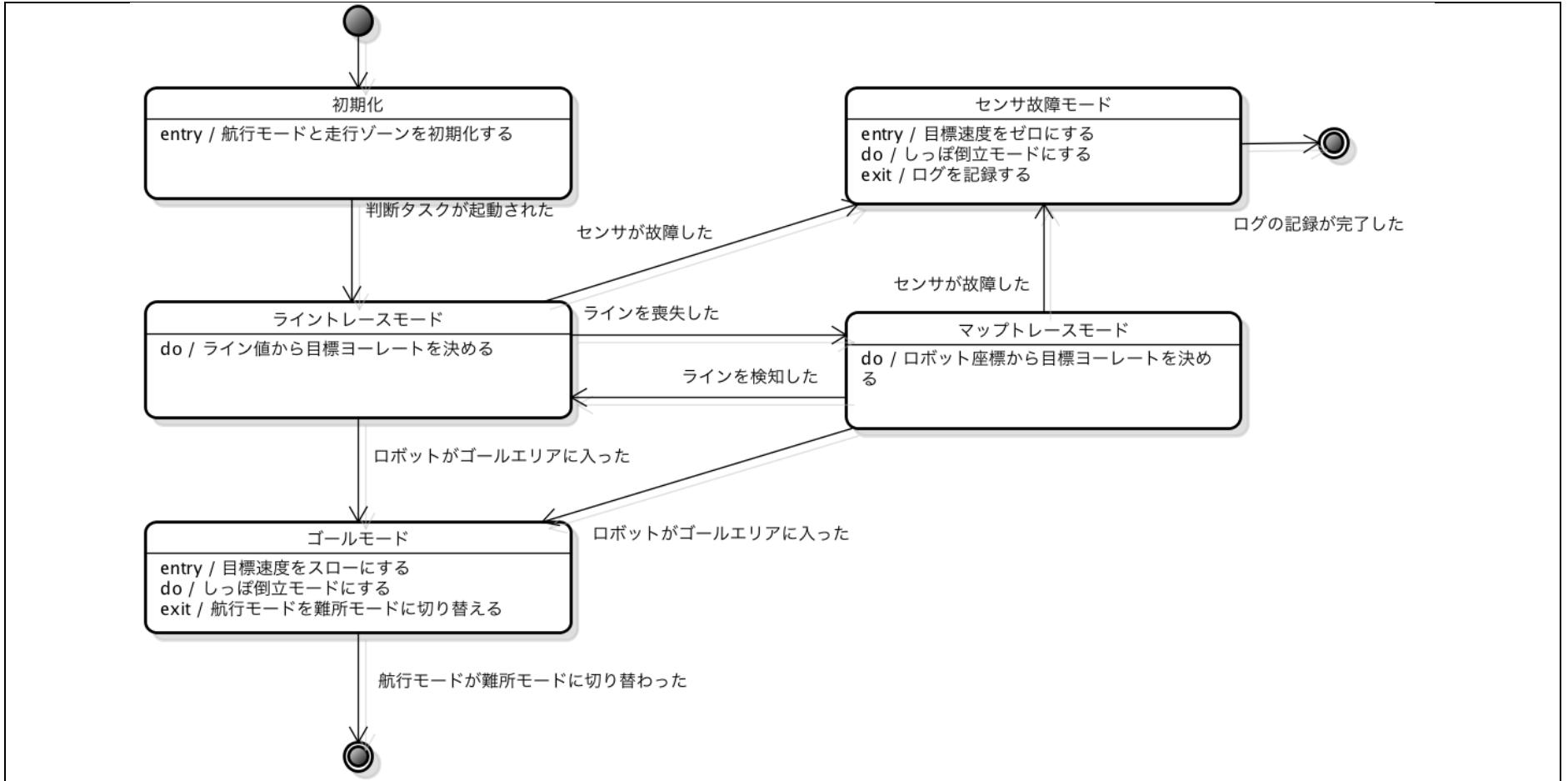


#### 4.6.3 ステートマシン図

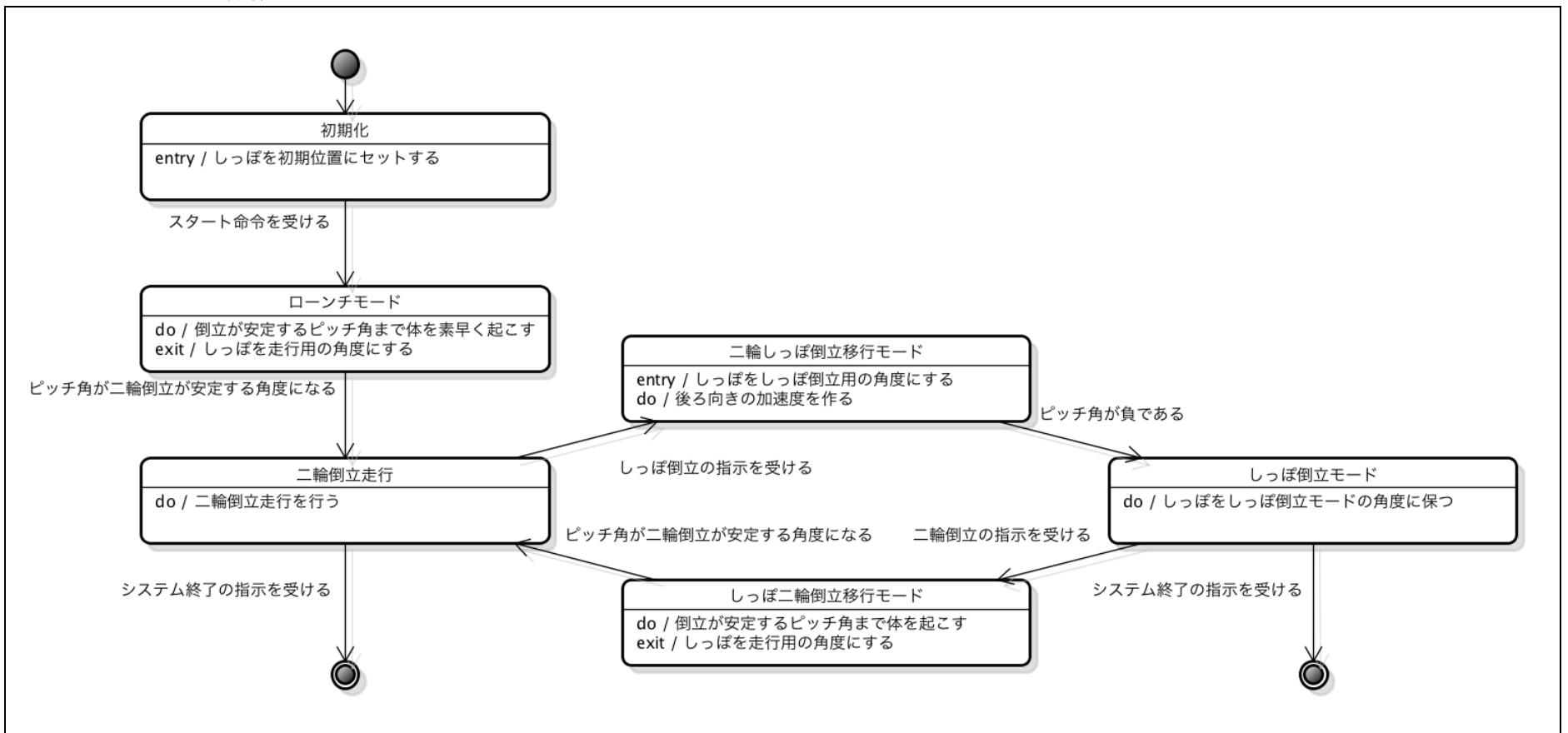
##### 4.6.3.1 あなごシステム ステートマシン図



## 4.6.3.2 判断システム ステートマシン図

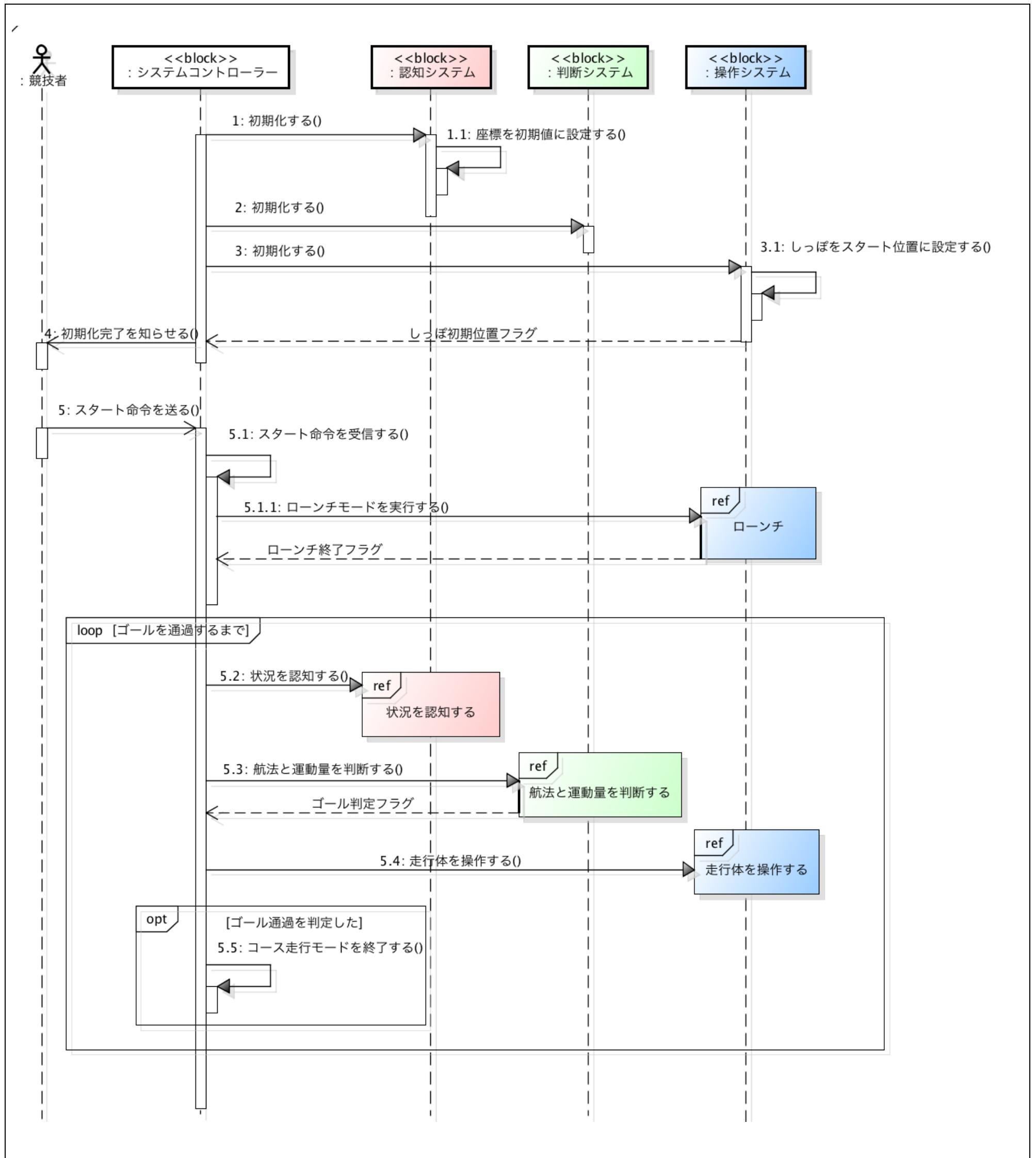


## 4.6.3.3 操作システム ステートマシン図

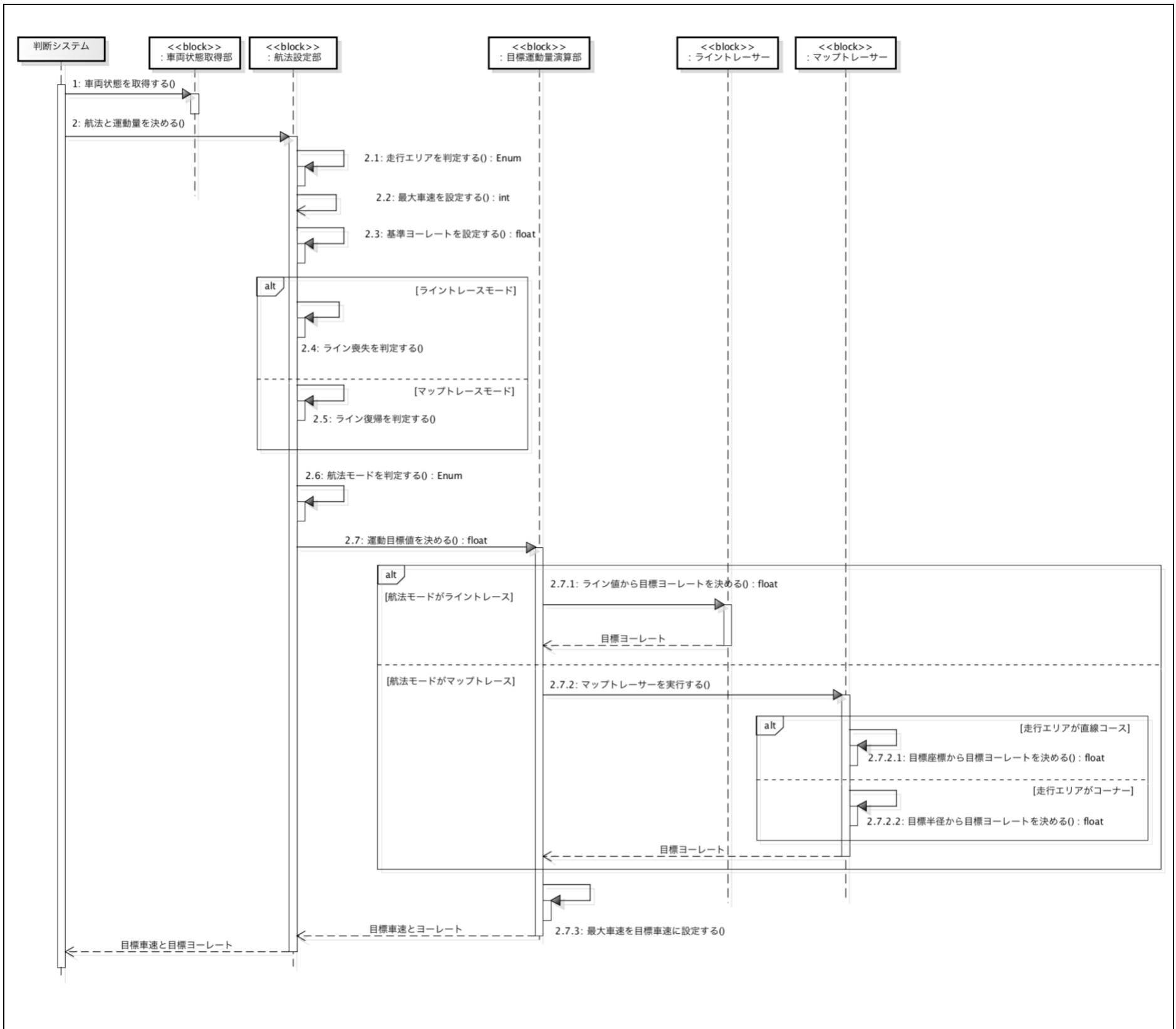


#### 4.6.4 シーケンス図

##### 4.6.4.1 あわごシステム シーケンス図

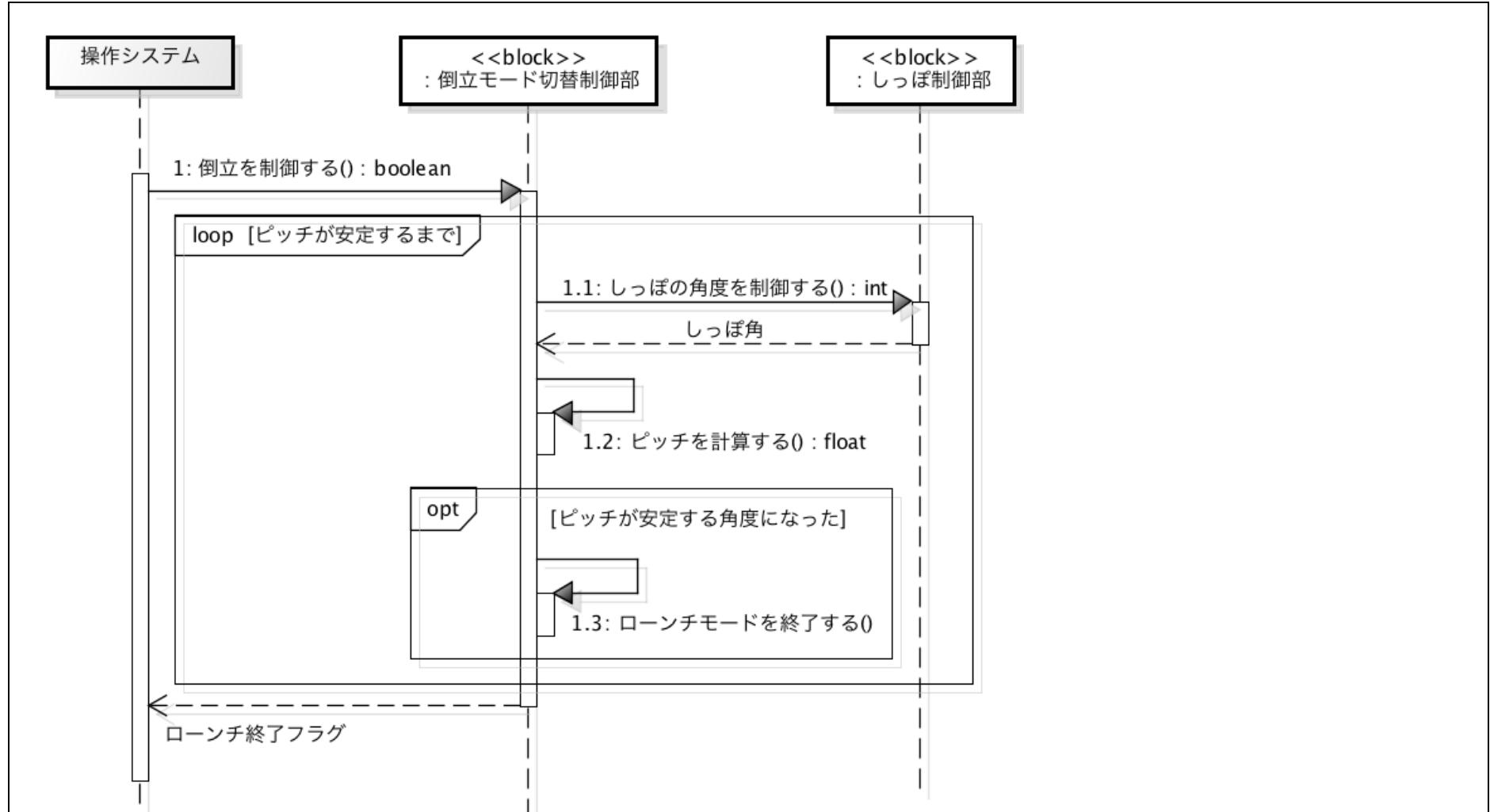


#### 4.6.4.2 判断システム シーケンス図

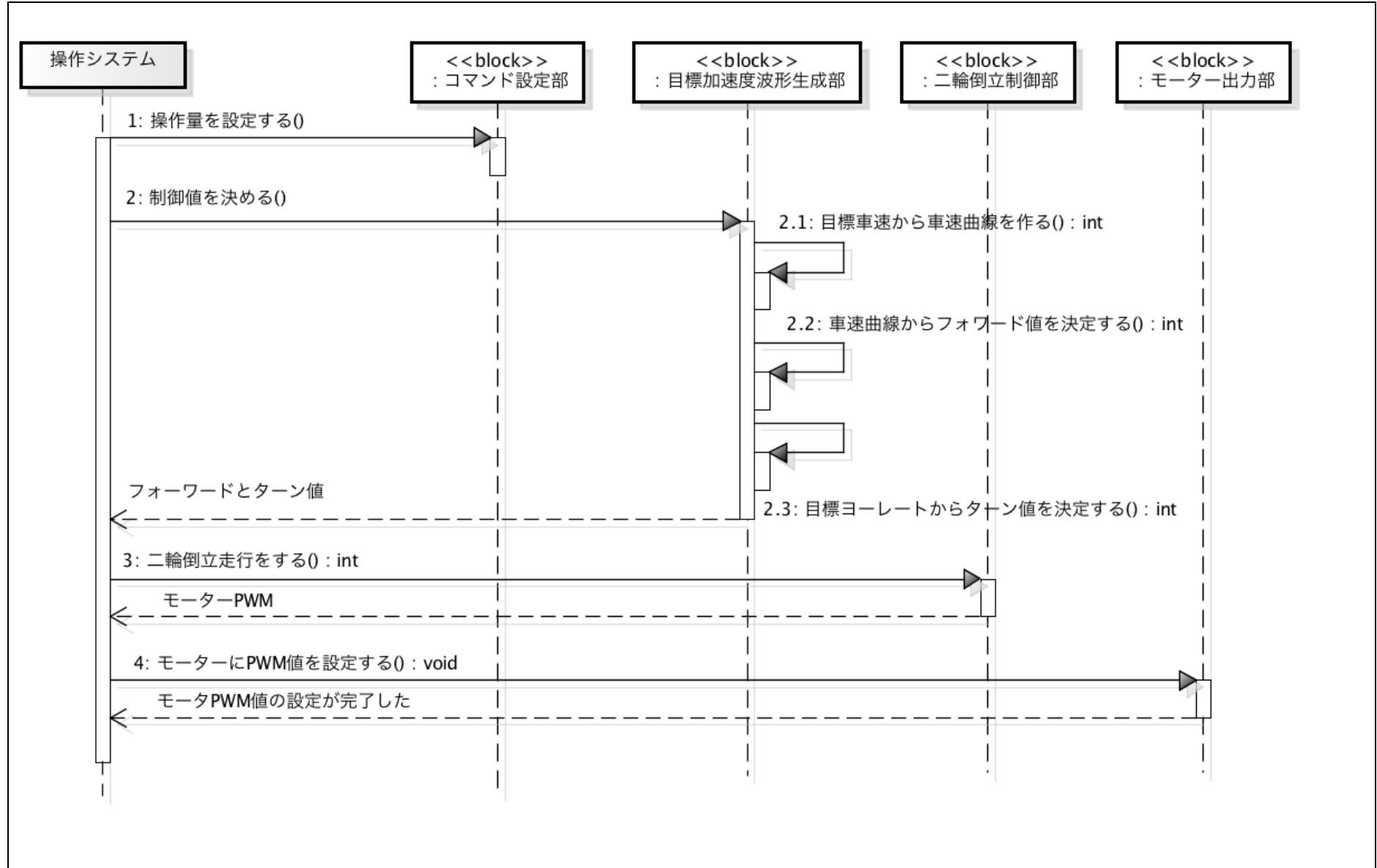


#### 4.6.4.3 操作システム シーケンス図

「ローンチ」のシーケンス図



「走行体を操作する」のシーケンス図



## 5 あなたプロジェクトの振り返り

### 5.1 プロジェクト目標の達成状況と成果物

#### 目標と達成状況

目標	主担当	達成状況
要求分析からアーキテクチャ設計までの設計技術を習得する	坂下	一通りかじるも習得までは至らず。
設計の全行程を経験し、習得する	池上	一通りかじるも習得までは至らず。
設計の教科書をつくる	太田	資料は完成させた。内容は、、、
目標工数の中で作業する 一人ひと月 10 時間かな～	太田	7,8,9 月は月 10 時間をオーバー 8 月以降は工数管理をしていない

#### 成果物

成果物	
モデルシート	<a href="https://drive.google.com/open?id=16wBwfPEgEKD156LtEB9cFSSb-QWC328h">https://drive.google.com/open?id=16wBwfPEgEKD156LtEB9cFSSb-QWC328h</a>
走行体プログラム	寄愛
設計の教科書	<a href="https://drive.google.com/open?id=1mGI_3Bx6igpEds4LR2FzeaMXsiatO3Z8">https://drive.google.com/open?id=1mGI_3Bx6igpEds4LR2FzeaMXsiatO3Z8</a>

### 5.2 システム目標の達成状況

#### システム目標と達成状況

目標	達成状況
120 秒以内にガレージの前までクリアする。	難所のガレージとシーソーがクリア出来ず
ゴールまで 40 秒、LUG とシーソーは 80 秒以内でクリアする。	地区大会の結果 左コースの走行タイム:36.8 秒 LUG:OK, ガレージ:NG 右コースの走行タイム:38.6 秒 シーソー:NG
リモートスタートする	達成
目標リザルトタイム:16 秒	地区大会の結果:42.4 秒
成功率 100%	左コース:70% 右コース:0%
理想の G 波形(ベルシェイプ)で走る。前後、横の躍度 0 で走行する	未検証 躍度 0 ではないが、 前年のシステムに比べ改善されている。

### 5.3 地区大会の結果

総合準優勝

競技:6 位

モデル:A-

CS 大会選抜チームに選ばれた。

ET ロボコン 2018 中四国地区大会 大会結果

<http://chushikoku.etrobo.jp/result.php>

ET ロボコン 2018 全国ランキング表

<http://etroboring.azurewebsites.net/DevPrimary/2018>

<http://etroboring.azurewebsites.net/DevPrimary/2018/L>

## 5.4 反省点

### ■メンバーのライフスタイルやスキルに合ったタスクの割り振りが出来なかつた

メンバーAを基準にWBSを作成、各メンバーにタスクを割り振ったが、当然出来ない。

#### ・一週間でロボコン活動が出来る時間(平均)

メンバー	時間[h]
A	8
B	2
C	4

メンバーAを基準にタスクを検討したので、数時間単位のタスクとなっていた。

まとまった時間がないと完了しない。

ただタスクが小さすぎるとメンバーの主体性がなくなる、、、ジレンマ。次回はどうしよう。

#### ・ロボコン活動が出来る場所

メンバー	自宅	MEC	エレキ棟
A	○	○	×
B	×	○	×
C	×	○	×

MECには毎週行けない、平日もいけない。数週間に一度の活動となる。

設計やプログラムのような作業を継続して完了させるようなタスクは出来ない。

プロジェクト当初の MECでの活動は少なくして出来るだけ自宅で活動する方針が間違っていた。

#### ・保有スキル

メンバー	システム設計	ソフト設計	SysML	C++	走行体の知識
A	○	○	○	○	○
B	×	×	×	×	×
C	×	×	×	×	△

メンバー全員がハード屋さん、ソフト設計するための基礎が無かった。

ほとんどのタスクがソフト設計、モデリング、プログラミングのスキルが必要。

勉強すればタスクをこなせると思っていたが、

ロボコン活動に使える少ない時間、勉強だけで終わってしまう。

月10時間の工数の中で勉強と作業の両立に無理があった?

ソフトのスペシャリストじゃないと月10時間で設計できなかった?

#### ・開発環境

メンバー	モデリング	プログラム
A	○	○
B	○	×
C	○	○

開発環境が整っていないと、設計のタスクは出来ない。

自分で開発環境のPCを準備する必要がある。PCが無い場合は、出費が痛い、、、10万円くらい

### ■デザインレビューが出来なかつた

### ■工数管理を8月以降やらなかつた

### ■欲に負けた

プロジェクトの目的は勉強だったが、やっぱり競技に勝ちたいとなった。

## 6 おわりに

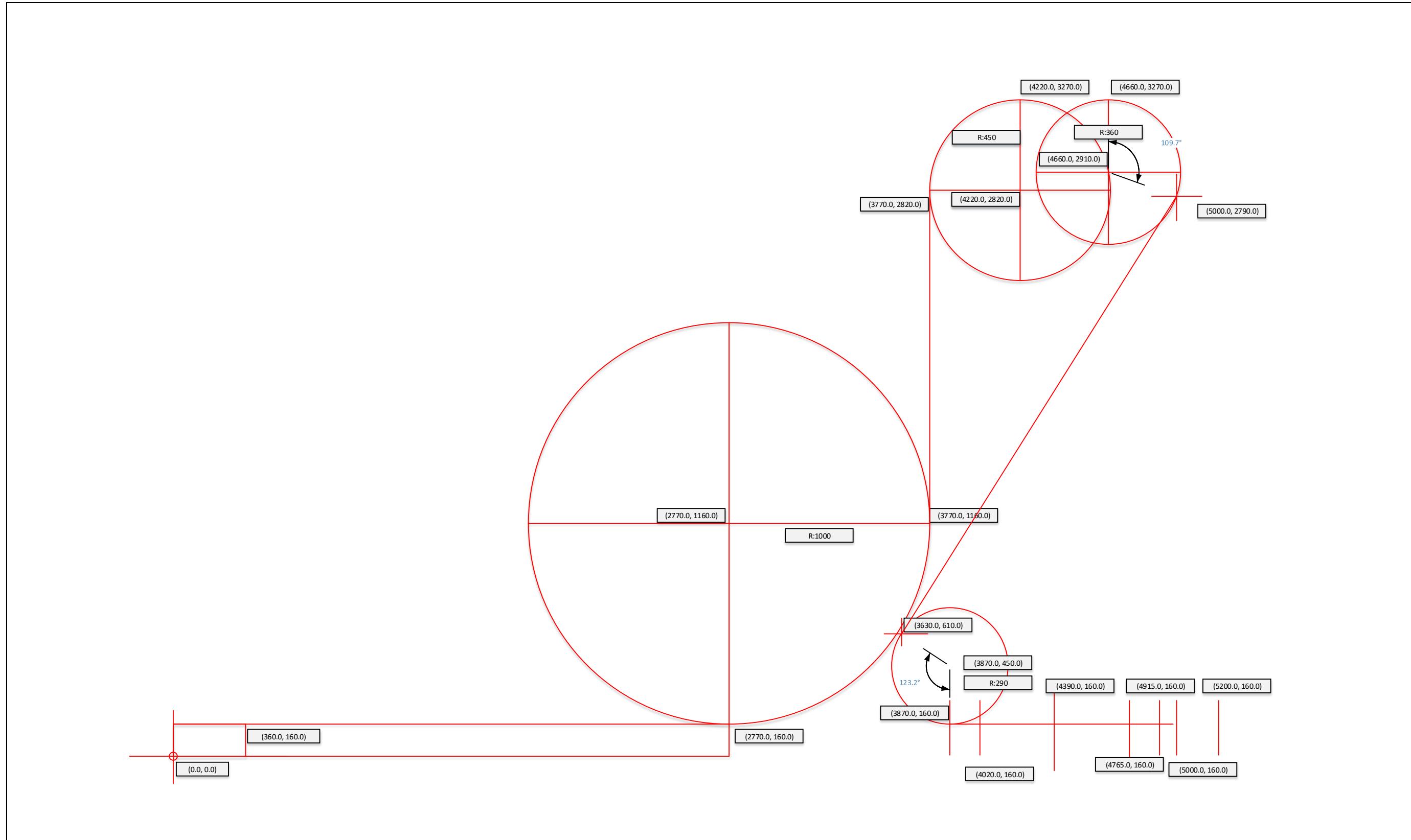
本プロジェクトでは、進捗と工数の監視、管理を行うことでメンバーの負担にならないロボコンを目指したが、WBS のタスクが各メンバーのライフスタイルやスキルに合ったタスクとなっていなかったためにメンバーへの負担が発生した。WBS のタスクを検討する時は、各メンバーのライフサイクルも考慮したタスクの決定が重要である事を学んだ。

## 7 Appendix

### 7.1 資料一覧

資料	URL
Three Anagos Project	<a href="#">割愛</a>
モデルシート	<a href="https://drive.google.com/open?id=16wBwfPEgEKD156LtEB9cFSSb-QWC328h">https://drive.google.com/open?id=16wBwfPEgEKD156LtEB9cFSSb-QWC328h</a>
走行体プログラム	左コース <a href="#">割愛</a> 右コース <a href="#">割愛</a>
設計の教科書	<a href="https://drive.google.com/open?id=1mGI_3Bx6igpEds4LR2FzeaMXsiatO3Z8">https://drive.google.com/open?id=1mGI_3Bx6igpEds4LR2FzeaMXsiatO3Z8</a>
WBS 工数管理	<a href="#">割愛</a>
ステークホルダー要求	<a href="#">割愛</a>
あなごシナリオ	<a href="#">割愛</a>
あなごシステム図	<a href="https://drive.google.com/open?id=16Wg5aufgn3I_cqgeZ8tcOUZVq3fmn7TL">https://drive.google.com/open?id=16Wg5aufgn3I_cqgeZ8tcOUZVq3fmn7TL</a>
あなごのモデル	<a href="#">割愛</a>
コースデータ	<a href="https://drive.google.com/open?id=1Ih-fZX68pgRuQWVPbCW-5Oj-Bgrn3pgo">https://drive.google.com/open?id=1Ih-fZX68pgRuQWVPbCW-5Oj-Bgrn3pgo</a>
地区大会チェックリスト	<a href="#">割愛</a>
ET ロボコン 2018 中四国地区大会 大会結果	<a href="http://chushikoku.etrobo.jp/result.php">http://chushikoku.etrobo.jp/result.php</a>
ET ロボコン 2018 全国ランキング表	<a href="http://etroboranking.azurewebsites.net/DevPrimary/2018">http://etroboranking.azurewebsites.net/DevPrimary/2018</a>

## 7.2 左コース座標



## 7.3 右コース座標

