STAMP/STPAによる
踏切制御システムの安全性要求分析

2017. 11. 28

東日本旅客鉄道株式会社
JR東日本研究開発センター
〇国藤　隆　岡田　明正　阿満　利仁

発表内容

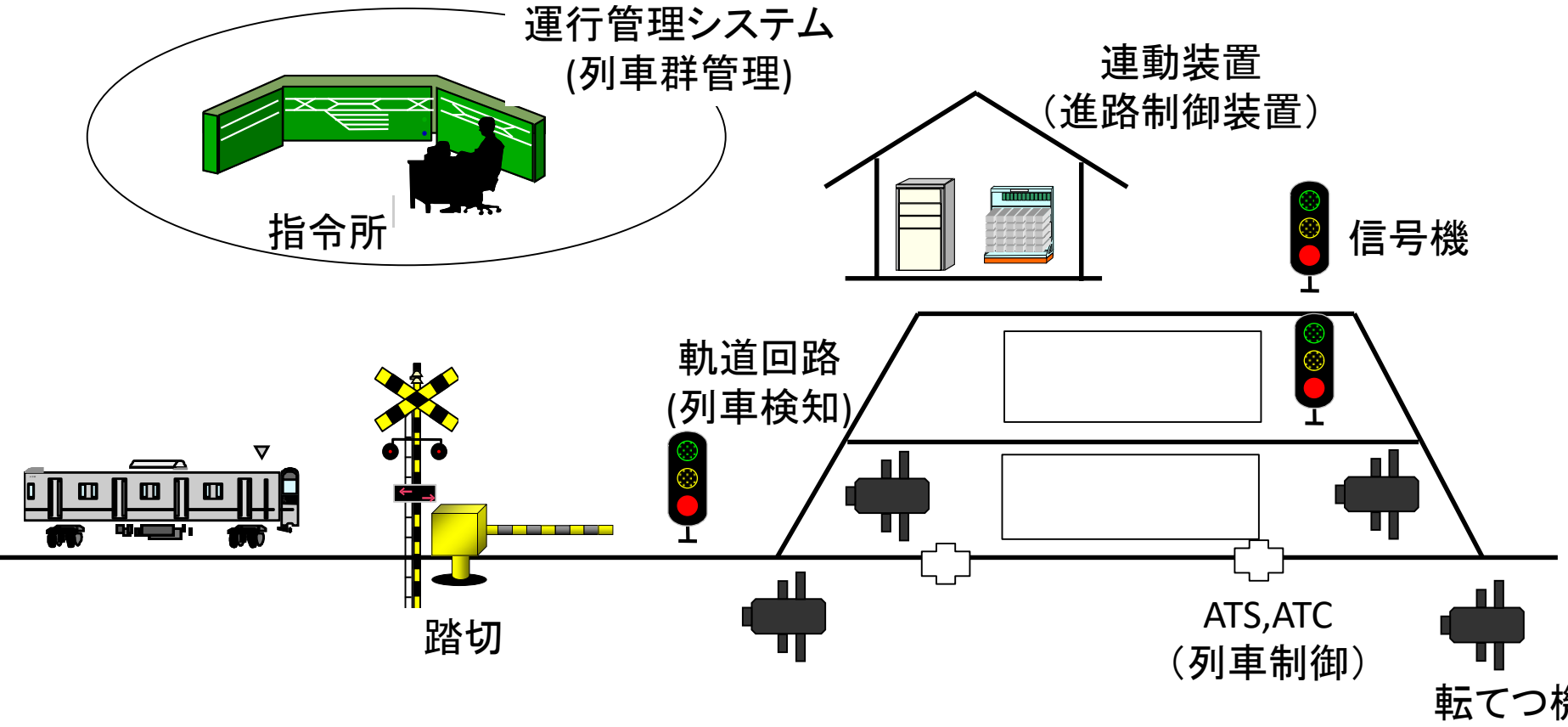
1. 背景
2. STAMP理論
3. STAMPにもとづく安全解析の試行
4. 課題と今後の展望

1.　背景

- 信号保安システムの役割と発展
- 信号保安システムの変遷
- 安全解析手法の位置づけ
- 主な安全解析手法と課題
- STAMP/STPAの信号保安システムへの適用の可能性

鉄道における信号保安システムの役割

衝突・脱線等を防止し、安全を確保
列車位置の把握，ダイヤに基づいた進路制御

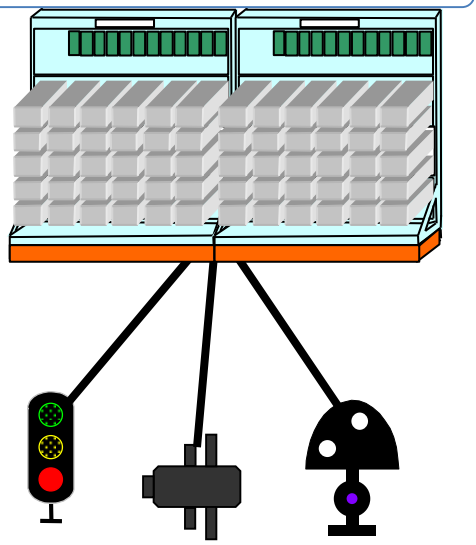


信号保安システムの変遷

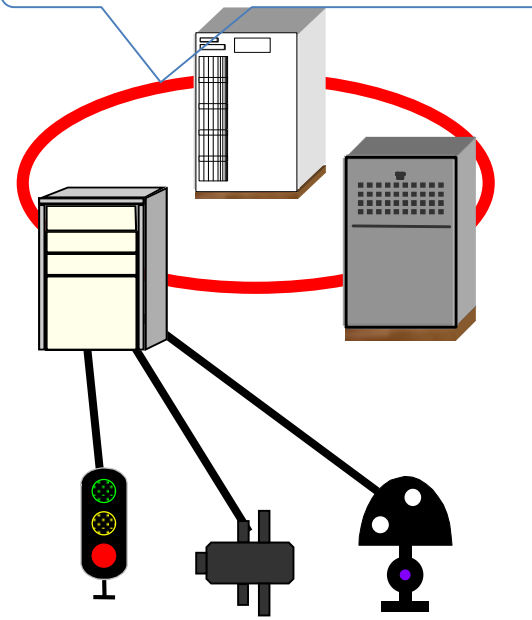
ソフトウェア・ネットワークを活用した形態に変化
⇒ **集中制御から分散制御へ、サブシステム間の相互作用の増加**

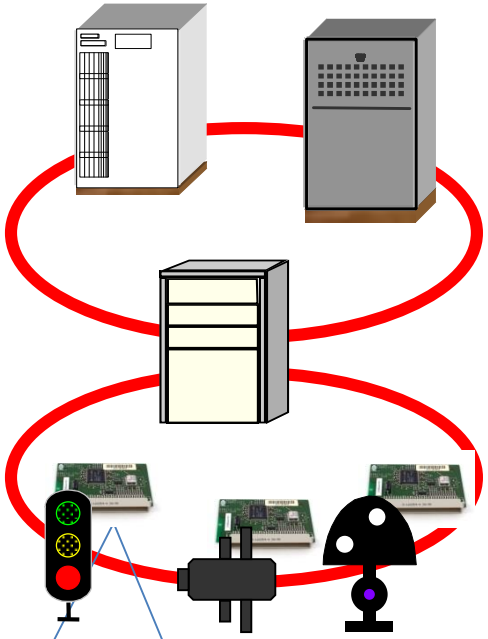
○連動装置の場合

ハードロジック・電圧制御



ソフトロジック・ ネットワーク化

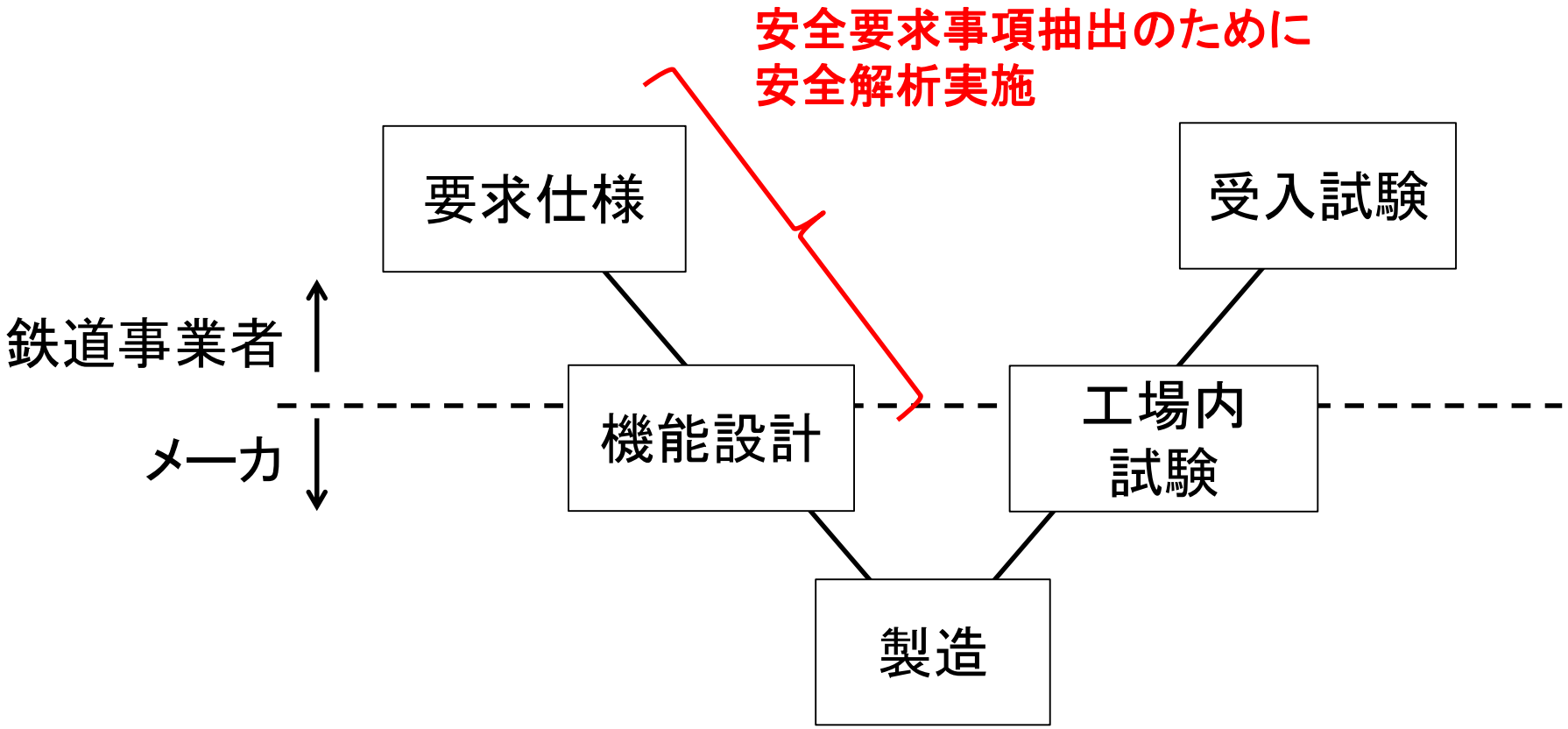




インテリジェント化・情報制御

安全解析の位置づけ

- システムの安全性の向上
- 安全性の根拠を明確化



主な安全解析手法と課題

- FTA
  - トップダウン
  - 望ましくない事象を決め，それの発生要因を系統的に分析
- FMEA
  - ボトムアップ
  - ある故障モードから発生しうる不具合を分析
- HAZOP
  - 分析対象に対して，早・遅等のガイドワードを用いて，潜在危険性を分析

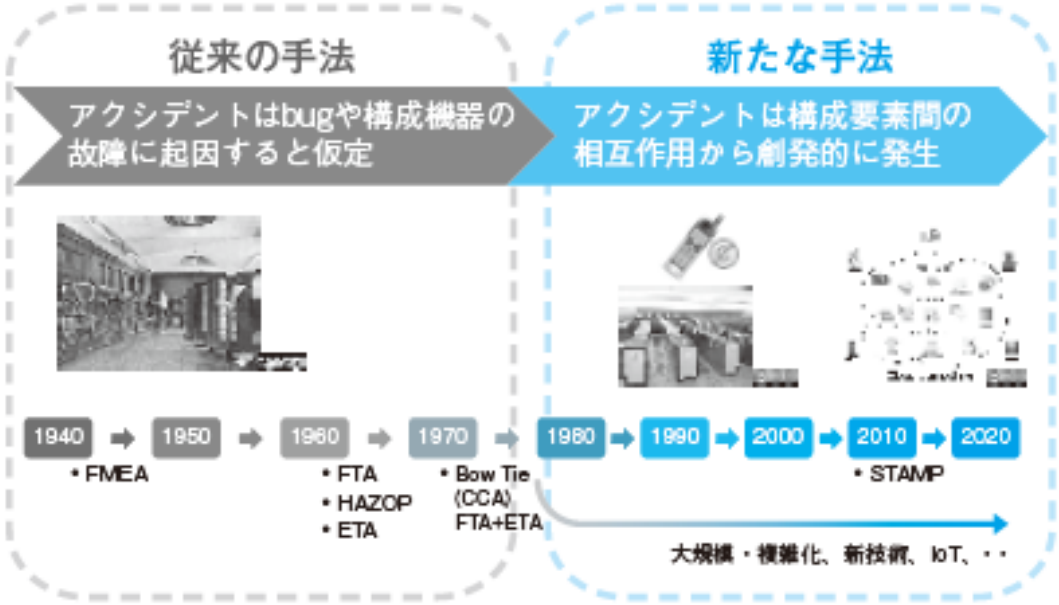
サブシステム間の相互作用が発生するシステム
に適用するためには工夫が必要

2.　STAMP理論

- STAMPの概念
- STAMPの事故モデル
- STAMPモデルにもとづく安全解析手法

STAMP(System-Theoretic Accident Model and Processes)の概念

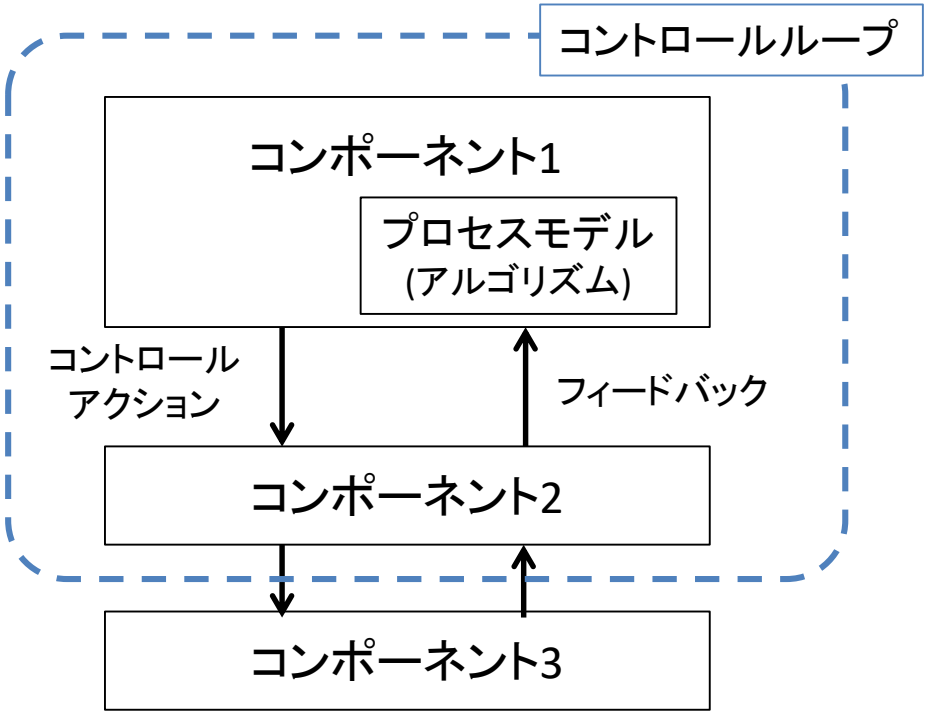
- MITのNancy Leveson教授が提案したシステム理論に基づく事故モデル
  - システムの安全性は構成要素の相互作用から創発されるもので、個々の要素を分割して分析するべきでない
  - 現代のシステムのアクシデントの多くは、システム構成要素の故障によって起きるのでなく、システムの中で安全のための制御を行う要素(コントローラー)と制御される要素(被コントロールプロセス)の相互作用が働かないことによって起きる



出典：SEC journal Vol.12 No.1 Jul. 2016, IPA

STAMPの事故モデル

■ STAMPにおける相互作用モデル (コントロールストラクチャ)



現実のシステムは、コントロールループが複合した、あるいは入れ子となった複雑な構造を持つが、コントロールループの単位に分解して分析が可能と考える

■ 事故の発生プロセス

適切なコントロールアクションが
他のコンポーネントに提供されない
(不適切な相互作用)

安全制約を損ない，事故が発生

安全制約：システムを安全に保つための要件、制約

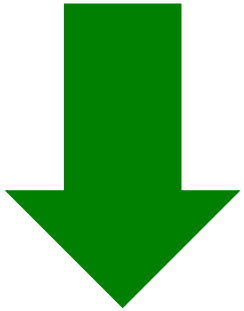
STAMPにもとづく主な安全解析手法

- STPA(System Theoretic Process Analysis)
  - トップダウンのハザード分析手法
  - 事前にアクシデント、ハザード、安全制約の定義と、コントロールストラクチャの構築が必要
  - 安全制約を損なう非安全なコントロールアクション(UCA: Unsafe Control Action)につながるハザード要因(HCF: Hazard Causal Factor)をガイドワードを用いて導出
- CAST(Causal Analysis using System Theory)
  - 事故事例に関わる装置間の関係をSTAMPを用いて表現
  - ボトムアップの事故分析手法
  - 事故事例を分析し、安全制約を抽出

STAMPの信号保安システムへの適用の可能性

論理のソフトウェア化・
システムの複雑化が進む信号保安システム

システムレベルでの
安全要求事項の把握



システムを対象とした安全解析手法である
STAMP/STPAを適用することがより良い結果を生むのではないか？

3.　STAMP理論にもとづく安全解析の試行

- 解析の対象
  - 構内踏切の制御論理
- 鉄道における踏切制御
- STAMP/STPA
  - 新たに試作した構内踏切制御論理の分析

解析の対象

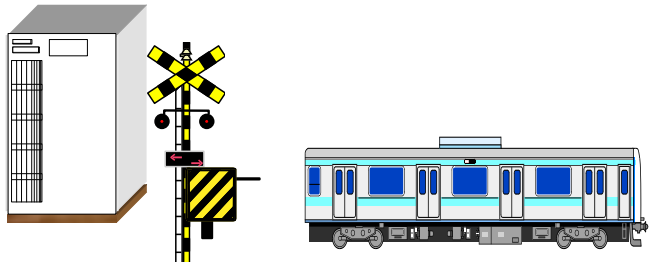
構内踏切制御論理を題材にSTAMP/STPAを実施

*構内踏切→駅構内に設置されている踏切

○　踏切制御の不具合の社会的な影響は大きく、少しでも安全性を高める方法が必要

○　施工、保守の負担を減らせるソフトウェア制御による踏切制御装置の開発のために、システム開発に適した安全解析手法が必要

リレーによる論理構築（現在）



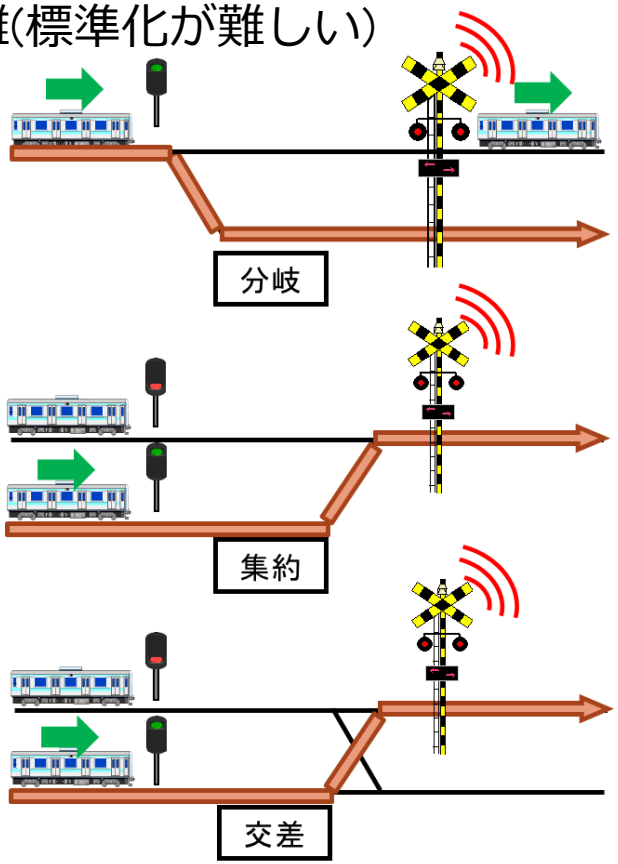
ソフトウェアによる制御

鉄道における踏切制御

踏切制御のソフトウェア化

- 中間：比較的容易であり実用化
- 構内：制御パタンが多様
  制御論理のソフトウェア化は困難(標準化が難しい)

列車数	中間	構内
単一 列車・車両	単行	単行
	—	折返し
	—	分割
複数 列車・車両	続行	続行
	対向	対向
	—	分岐
	—	集約
	—	交差
	—	併合



駅構内の踏切制御不具合は、複数列車による踏切制御の相互作用の不備に起因するものが多い
(ある列車が警報させた踏切を他の列車が警報を終止させる)

STAMP/STPA分析の実施手順

- Step.0-1: ハザードの識別
- Step.0-2: Control structure diagramの作成
- Step.1: 非安全なControl actionとハザードシナリオの分析
- Step.2: Control loop diagramによるハザード要因の分析
- Step.3: ハザード要因に対する安全制約の識別

STAMP/STPA Step.0-1
～ハザードの識別 ～

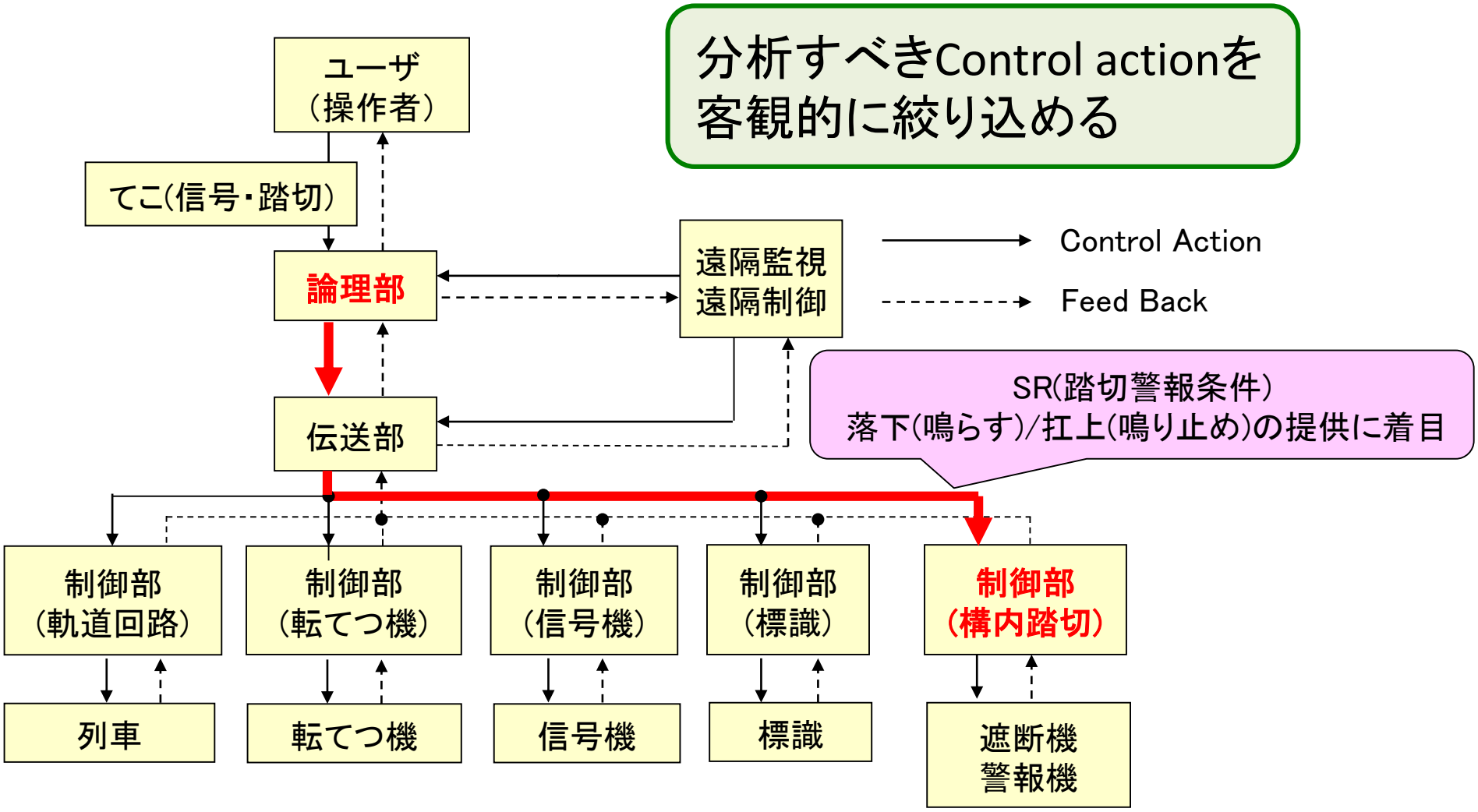
- 安全上問題となるハザード
  - H1: 列車が在線で踏切が遮断しない(無遮断)
  - H2: 踏切が遮断後に列車が在線にも関わらず開く (遮断不良)
  - H3: 警報時間の不足（法令への抵触）

- 運用性の低下につながるハザード
  - H4: 警報時間の過剰
  - H5: 列車が非在線で踏切が遮断する

＊ハザードの定義
事故をもたらす要因，または事故の誘因となる条件が顕在または潜在する状態

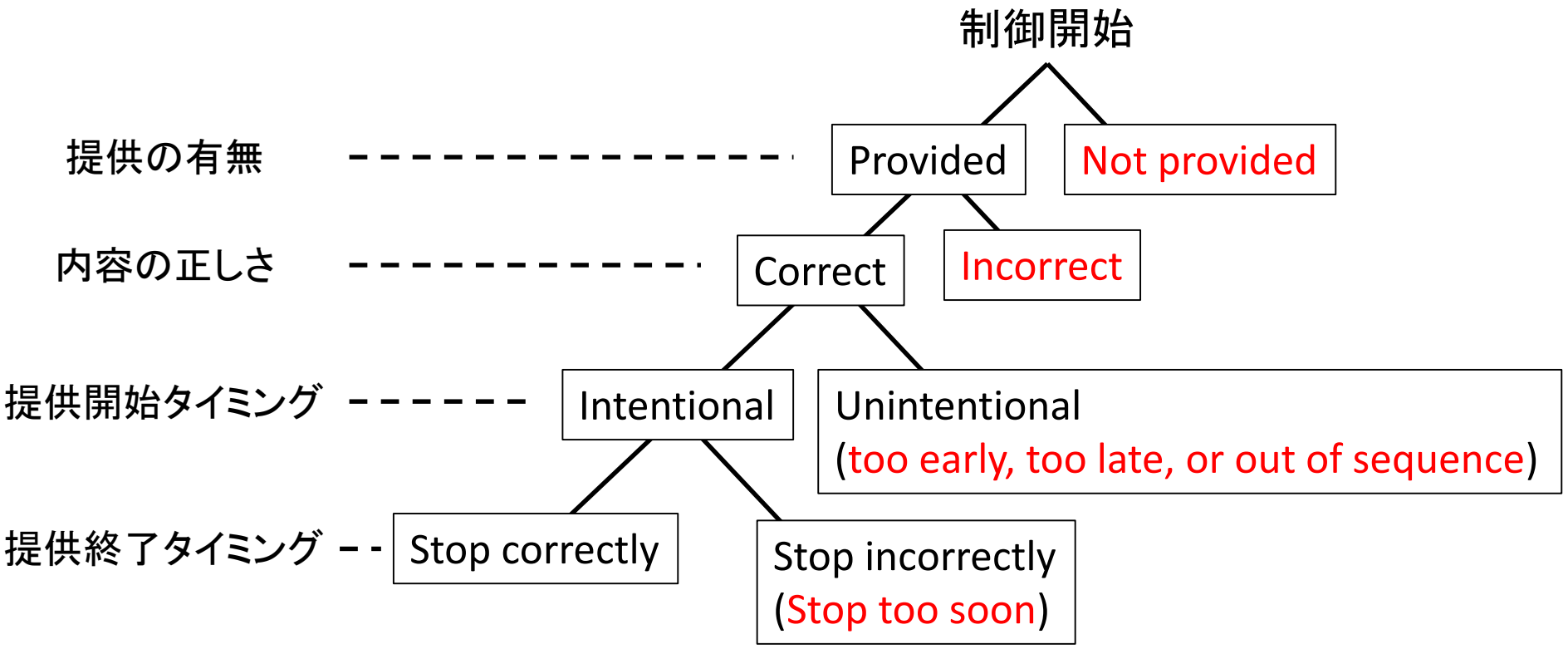
STAMP/STPA Step.0-2
～Control structure diagramの作成～

分析対象のコントロールストラクチャ



STAMP/STPA Step.1
～非安全なControl actionとハザードシナリオの分析(ひな形)～

制御の提供に関する網羅的なガイドワード用いて，各コントロールアクションでハザードが起きうるシナリオがあるかを分析可能



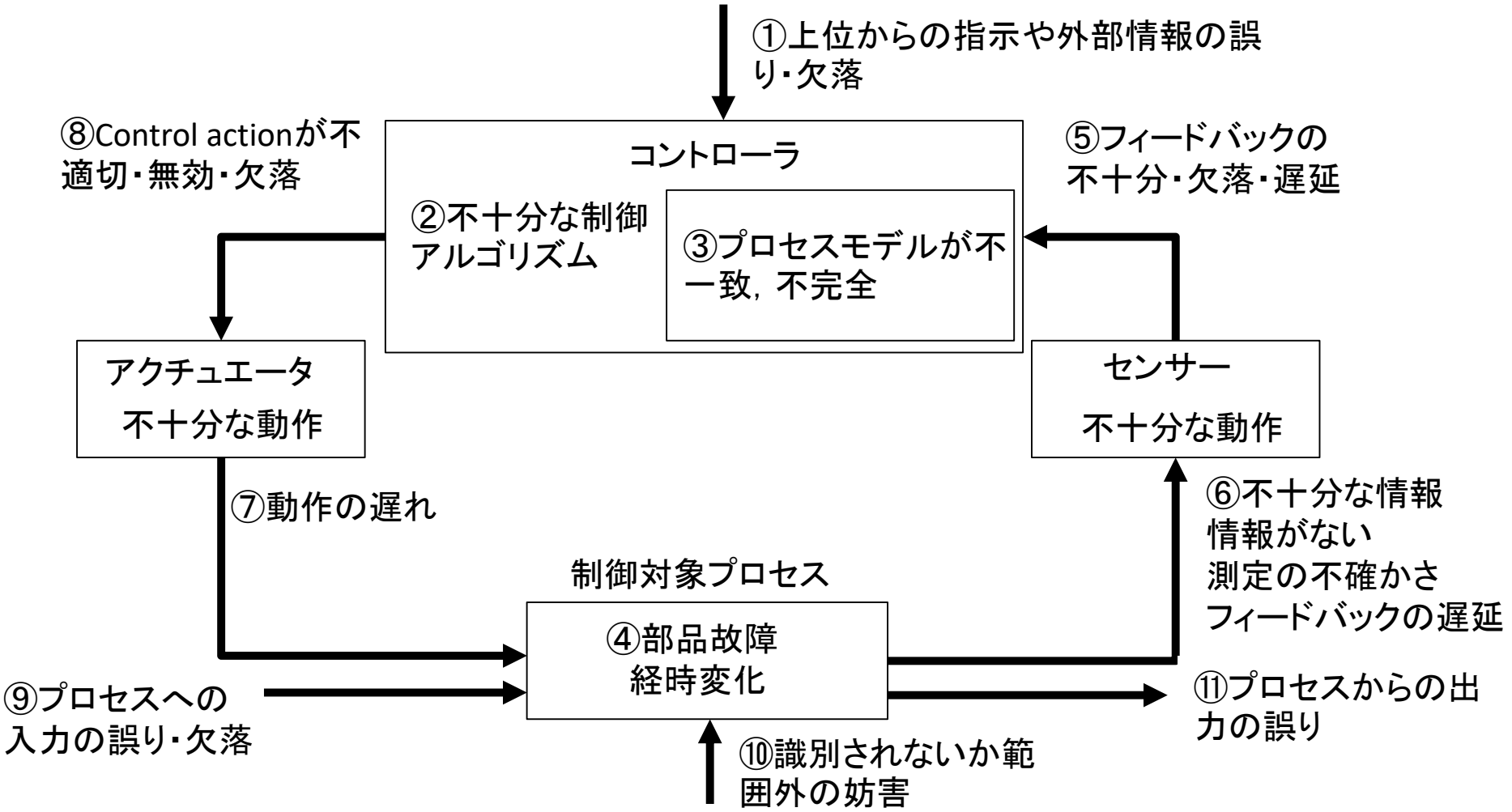
STAMP/STPA Step.1
～非安全なControl actionとハザードシナリオの分析(試行)～

Control action	Not Provided	Incorrectly provided	Too early, too late	Stop too soon
SR 落下	**ハザードH1に至る(1) [無遮断]**	ハザードH5に至る	**ハザードH3に至る(2) [警報時間不足]**	ハザードH4に至る
SR 扛上	ハザードH4に至る	**ハザードH2に至る(3) [遮断不良]**	**ハザードH3に至る(4) [警報時間不足]**	ハザードH4に至る

No	ハザードシナリオ
(1)	列車が警報進路(踏切を警報させる区間)に実際に在線しており，SR落下条件が整っている状況において、SRが提供されないとハザードH1「列車が在線で踏切が遮断しない」に至る
(2)	列車が警報進路に実際に在線しており，SR落下条件が整っている状況において，遅れてSR落下が提供されると，ハザードH3「警報時間の不足」に至る
(3)	列車が警報進路に実際に在線しており，SR扛上条件が整っていない状況において，SR扛上が提供されると，ハザードH2「踏切が遮断後に列車が在線にも関わらず開く」に至る
(4)	列車が警報進路に実際に在線しておらず，SR扛上条件が整っていない状況において，早まってSR扛上が提供されると，ハザードH3「警報時間の不足」に至る

STAMP/STPA Step.2
～Control loop diagramによるハザード要因の分析(ひな形)～

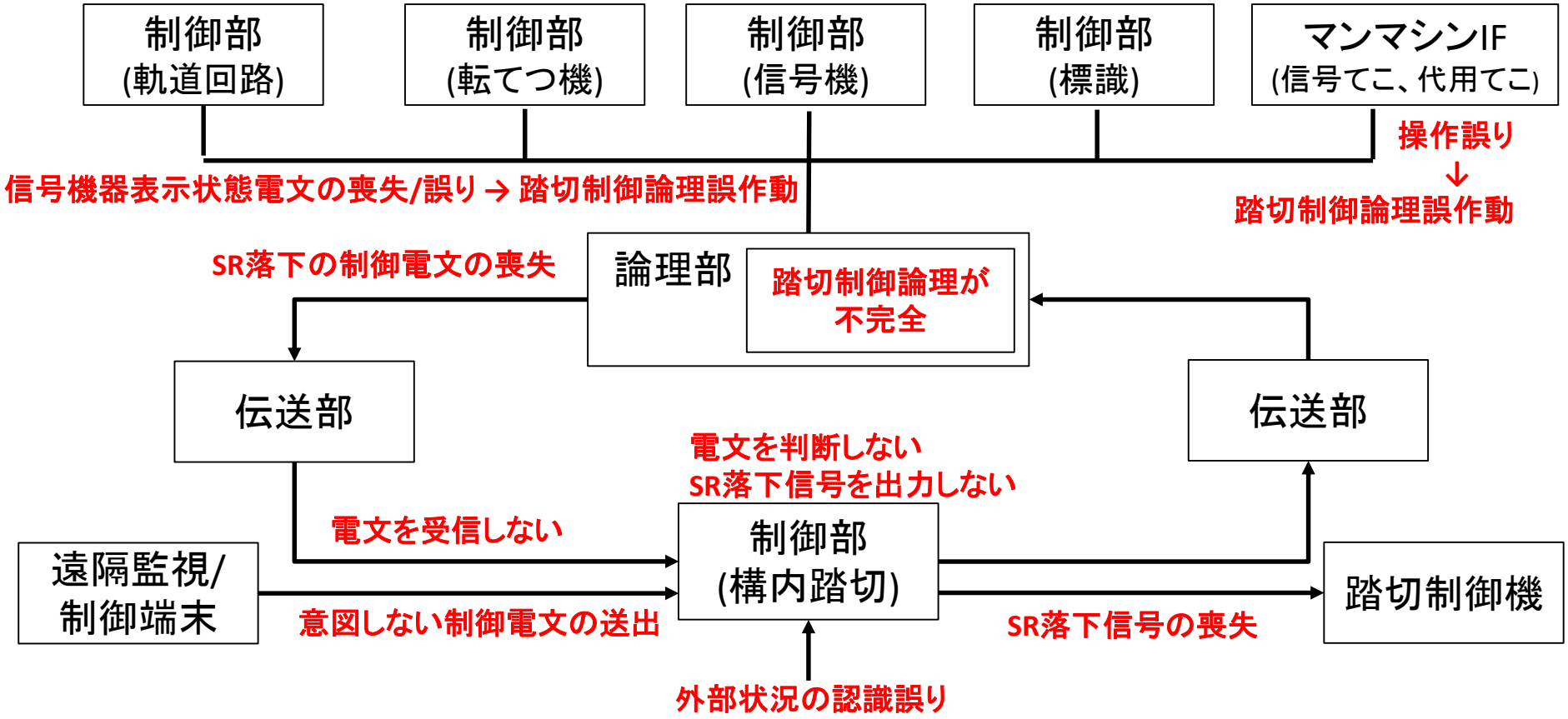
非安全なControl actionが発生する要因を11のガイドワードで分析



STAMP/STPA　Step.2
～Control loop diagramによるハザード要因の分析(試行)～

分析事例：ハザードNo.(1)

- 列車が警報進路に実際に在線しており，SR落下提供が正当である状況においてSR落下が踏切制御機に提供されないと、ハザードH1「列車が在線で踏切が遮断しない」に至る



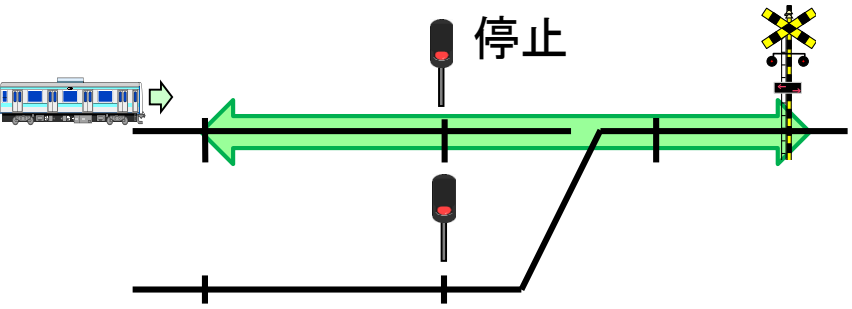
STAMP/STPA Step.3
～ハザード要因に対する安全制約の識別～

ハザード要因	安全制約（要求）
信号機器表示状態電文の喪失/誤り	電文を喪失しない/誤らない，あるいは喪失/誤りを検知する
操作誤り	危険側となる操作は受け付けない
SR落下制御電文の喪失/誤り	電文を喪失しない/誤らない，あるいは喪失/誤りを検知する
意図しない制御電文の送出	端末の利用制限を設ける
論理部からの電文を受信しない SR落下条件を判断しない，あるいは判断しても落下制御を行わない 外部状況の認識誤り	論理部からの電文を喪失しない，あるいは喪失を検知する 論理部からの電文を誤らない，あるいは誤りを訂正する
踏切制御論理(プロセスモデル)が不完全	※プロセスモデルをさらにSTAMP/STPAで解析

一般的な踏切制御手順

非警報
(SR扛上)

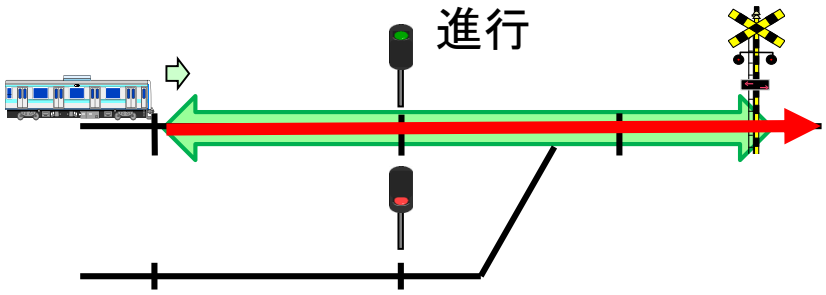
停止



①進路設定イベント

警報準備
(SR扛上)

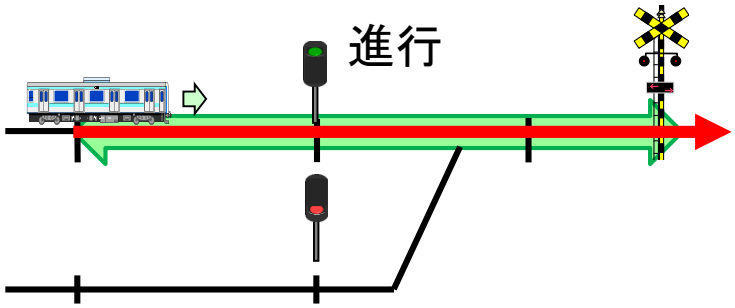
進行



②列車進入イベント

警報
(SR扛上→落下)

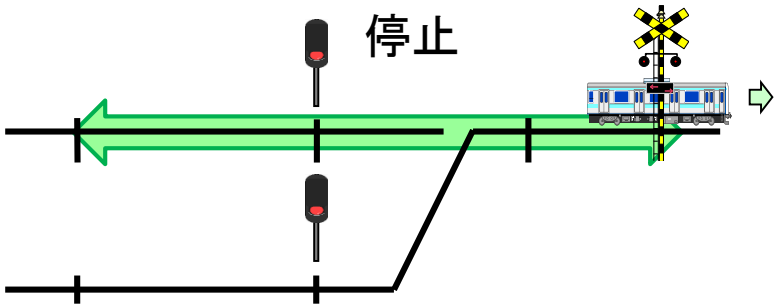
進行



③列車進出イベント

警報
(SR落下→扛上)

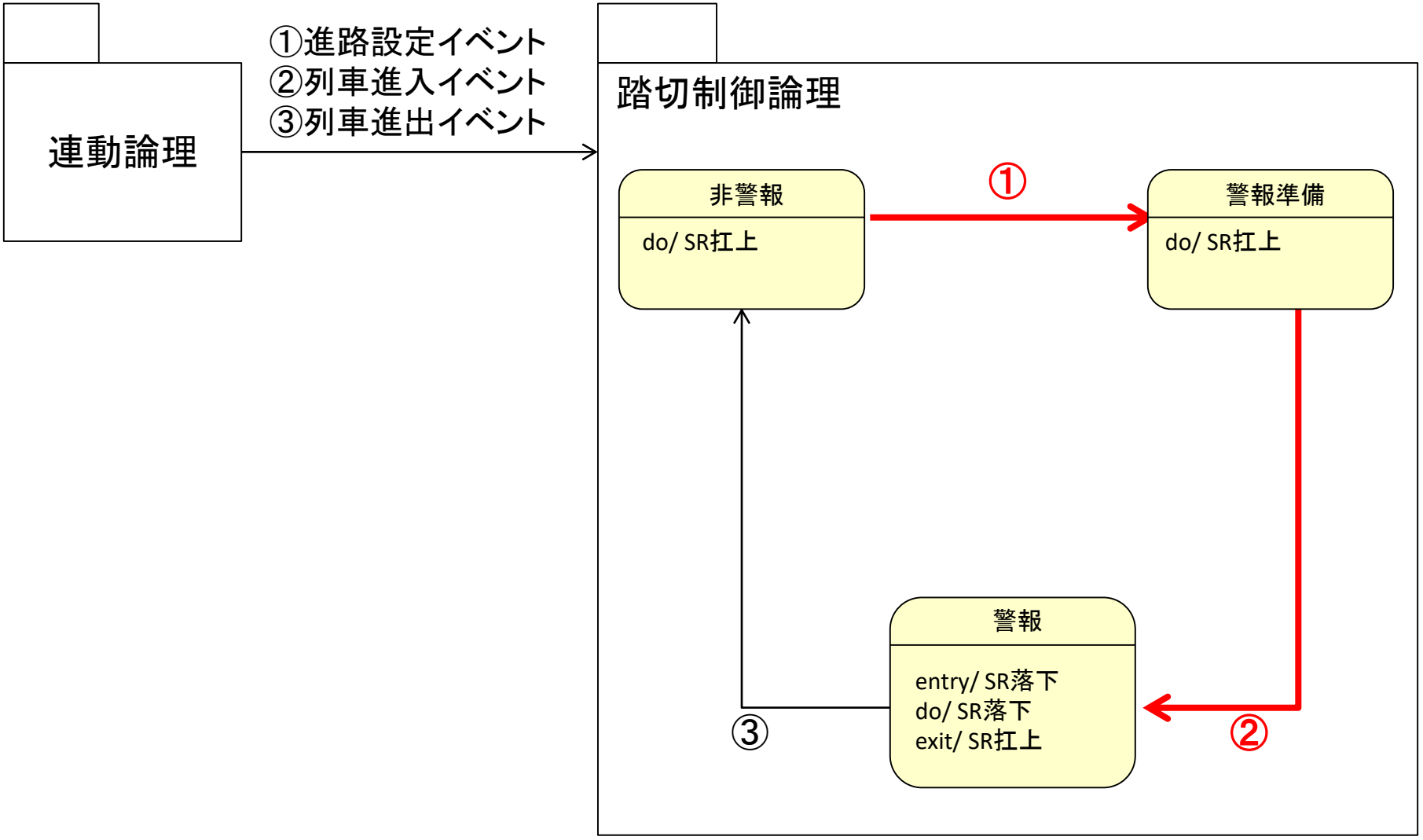
停止



踏切警報区間
(警報進路)

進路の設定

連動論理と踏切制御論理の相互作用



STAMP/STPA Step.1
～非安全なControl actionとハザードシナリオの分析(試行)～

Control action	Not Provided	Incorrectly provided	Too early, too late	Stop too soon
SR 落下	**ハザードH1に至る(1)**	ハザードH5に至る	**ハザードH3に至る(2)**	ハザードH4に至る
SR 扛上	ハザードH4に至る	**ハザードH2に至る(3)**	**ハザードH3に至る(4)**	ハザードH4に至る

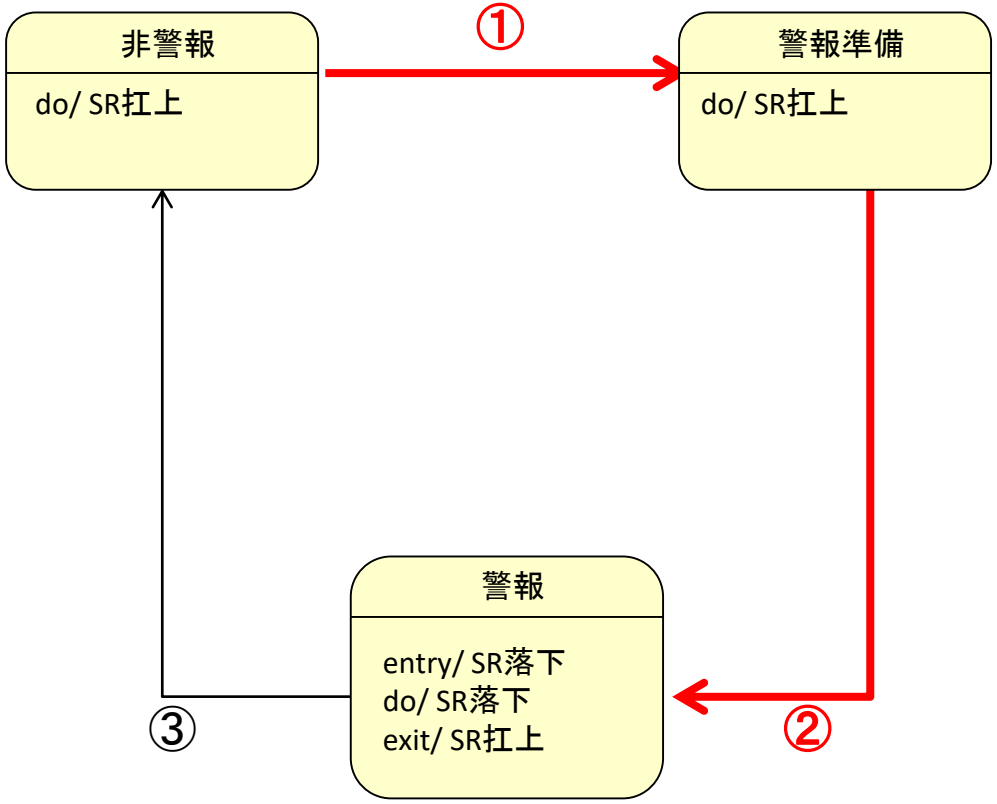
No	ハザードシナリオ
(1)	警報準備状態において、連動論理から進入イベントが通知されない
(2)	警報状態に遷移したとき、規定時間内にSR落下制御が行われない
(3)	警報状態において、連動論理から列車進出イベントが通知されないにも関わらずSR扛上制御が行われる
(4)	警報状態から非警報状態に遷移した時、規定時間より早くSR扛上制御が行われる

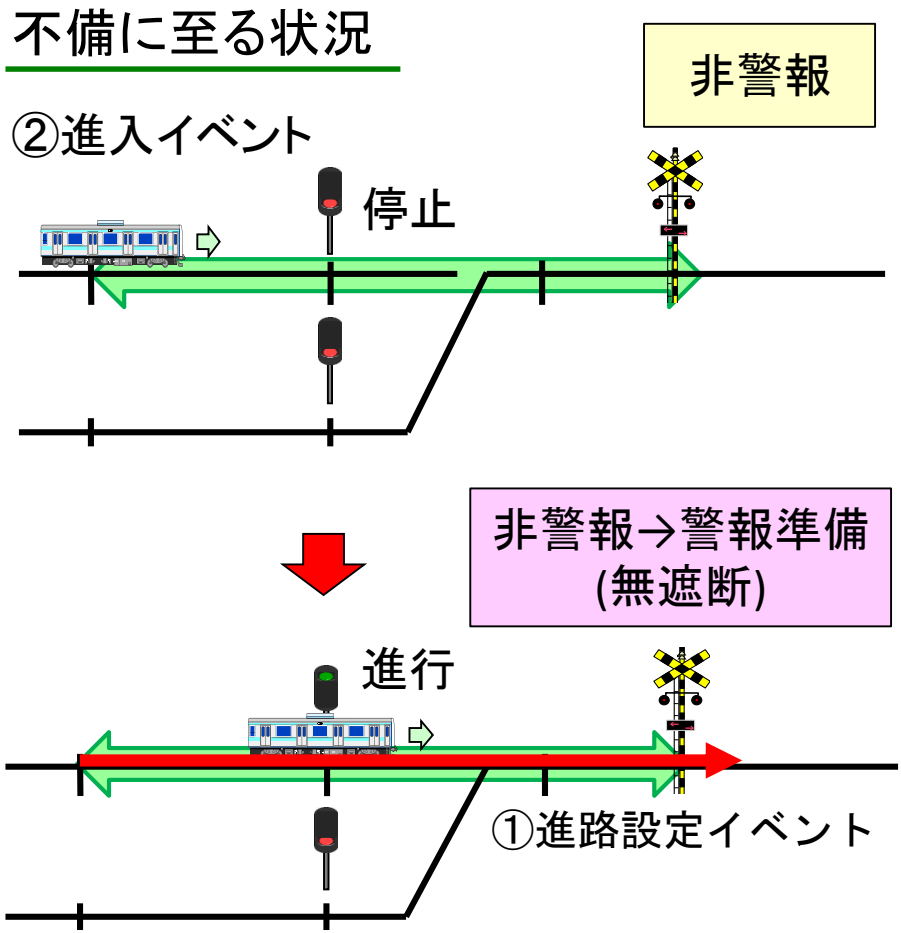
ハードウェア故障なし、ソフトウェアのバグなし(仕様通りであるという意味。ただし、要求仕様が正しいことを意味しない)という前提において発生する可能性のあるハザードシナリオは(1)のみ

ハザードシナリオ(1)の要因分析

状態遷移図不備の検出

要求：信号機進行制御＆踏切警報区間に進入で踏切警報開始
今回の仕様での状態遷移：①信号機進行制御 → ②列車進入
不具合の例：②列車進入→①信号機進行制御



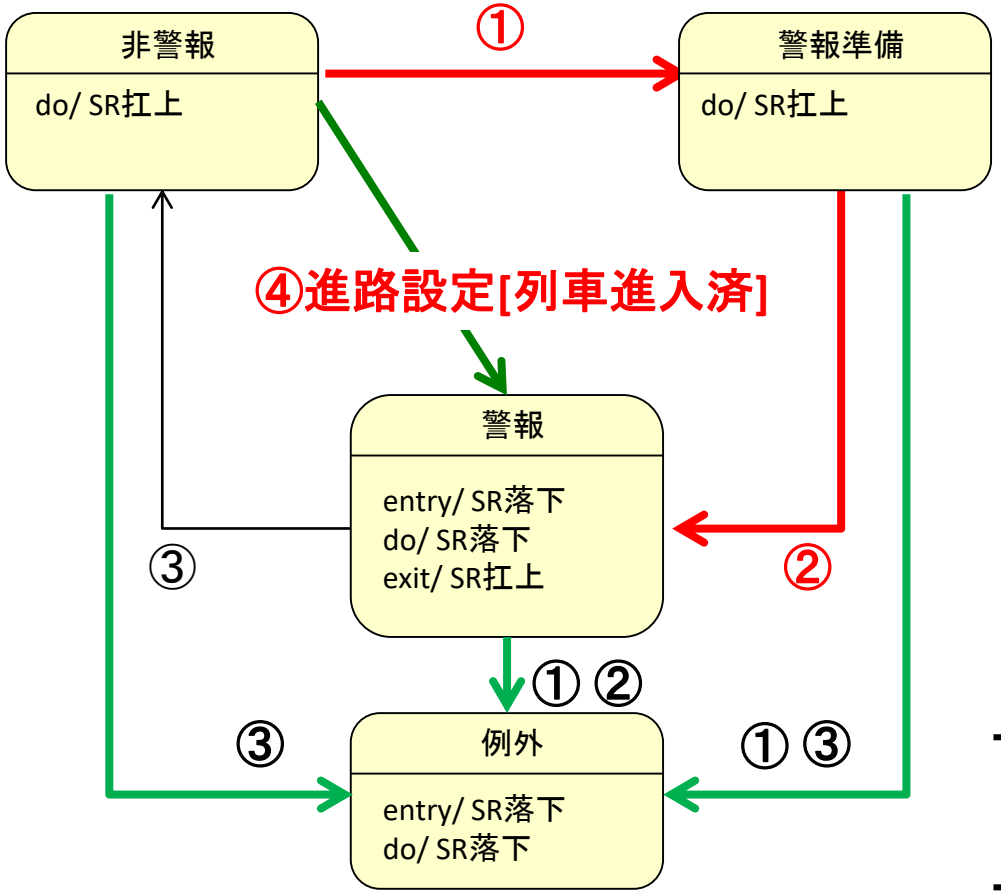


STAMP/STPA Step.3
～ハザード要因に対する安全制約の識別～

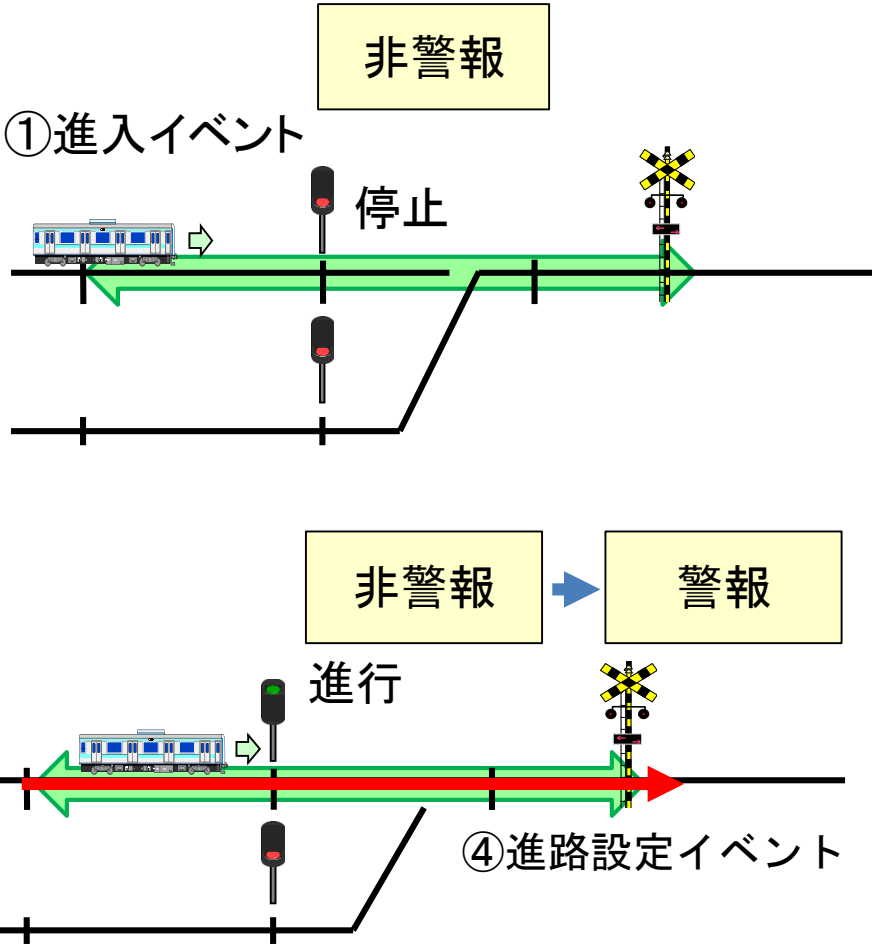
ハザード要因	安全制約（要求）
仕様で規定した以外の順序でイベントが発生	正常なイベント発生シーケンスを定義し、それ以外の順序で発生したイベントを受信した場合、例外処理(強制安全制御)を実行する

分析結果にもとづく状態遷移図(仕様)の修正

修正後の状態遷移図



警報に至る状況



STAMPにもとづく安全解析の試行(まとめ)

- STAMP/STPA
  - STAMPを基礎とすることで，分析すべき安全を損なう要因（Control action）が明確になる
  - Control structure，Control loop diagramの明示、及びガイドワードの活用により，網羅的かつ属人性を軽減した分析が可能
  - システムの構造を段階的にブレイクダウンしていきながら、繰り返し分析を行うことで、ハザード要因を漏れなく抽出し、それに対する安全性要求を定義することが可能
- STAMP/CAST
  - 具体的な事故事例から抽象的な安全性要求に順次遡っていくことができ、どの段階で誤りを作り込んだかが明確となる
  - 多くの事故事例を分析することで、根本の安全要求に抜けが無いかを発見することが可能
- STPAとCASTの両方を適用することで、より網羅性の高い安全要求を定義することができる

4.　今後の課題

- 一般的なシステム開発における課題としての要求と設計の乖離
  - STAMPにより安全性要求の網羅性は高まるが、要求が正しく、もれなく設計に反映されていることをどうすれば確認できるか
    - STAMPは、要求段階、設計段階を通して活用可能
    - STAMPでUCAをもれダブりなく抽出するには経験が必要
- 他のリスク分析手法との組合せ
  - Control loop diagramによるハザード要因分析の段階では、装置単体の故障、通信異常等、FTA,FMEAの活用が有効な場面もある
  - 仕様不備の洗い出しには形式的手法の活用が有効
- 形式的手法との組み合わせ
  - Control structure diagramによるUCA抽出とハザードシナリオの分析には、定理証明による検査手法の活用が有効と考えられる
  - プロセスモデルのハザード要因分析では、モデル検査手法の活用が有効と考えられる
  - 計算時間が膨大となりがちである、形式的手法による自動検査では、ハザード発生箇所を局所化していくことのできるSTAMP手法との相性が良いのではないかと考えられる

形式的手法にもとづくリスク分析のアイデア

- 要求段階
  - システムの相互作用モデル(Control Structure)をモデル言語(Sys-MLなど)で記述
  - 相互作用の記述をフォーマル言語に変換
  - 相互作用におけるUCAとハザードシナリオを定理証明器により網羅的に抽出(ガイドワードを含む命題を生成し証明)
- 設計段階
  - 要求段階で得られた最終の相互作用モデル(リスクへの対処済)から設計モデル(Scadeなど)を生成
  - モデル検査により、動作の正当性、妥当性を検証(リスクへの対処を行った箇所を中心に対策の正当性、妥当性を検証)