

派生開発カンファレンス2021

# 制御モデルの仕様化と 派生開発への展開

～Advanced Drive コントローラ開発へのUSDM、XDDPの適用～

\* トヨタ自動車(株) シャシー開発部

浅野 智孝

(株)デンソー 技術企画部

古畑 慶次



1. 担当製品と開発手法
2. 開発の変化と課題
3. 解決方法
4. 実施結果
5. 考察と今後の進め方
6. まとめ



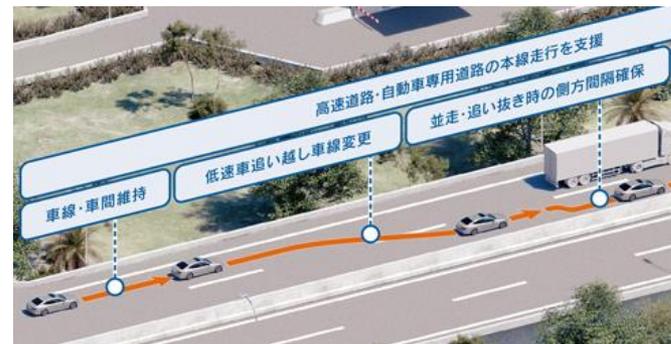
1. 担当製品と開発手法
2. 開発の変化と課題
3. 解決方法
4. 実施結果
5. 考察と今後の進め方
6. まとめ



## ■ Advanced Drive

### ● 高度運転支援

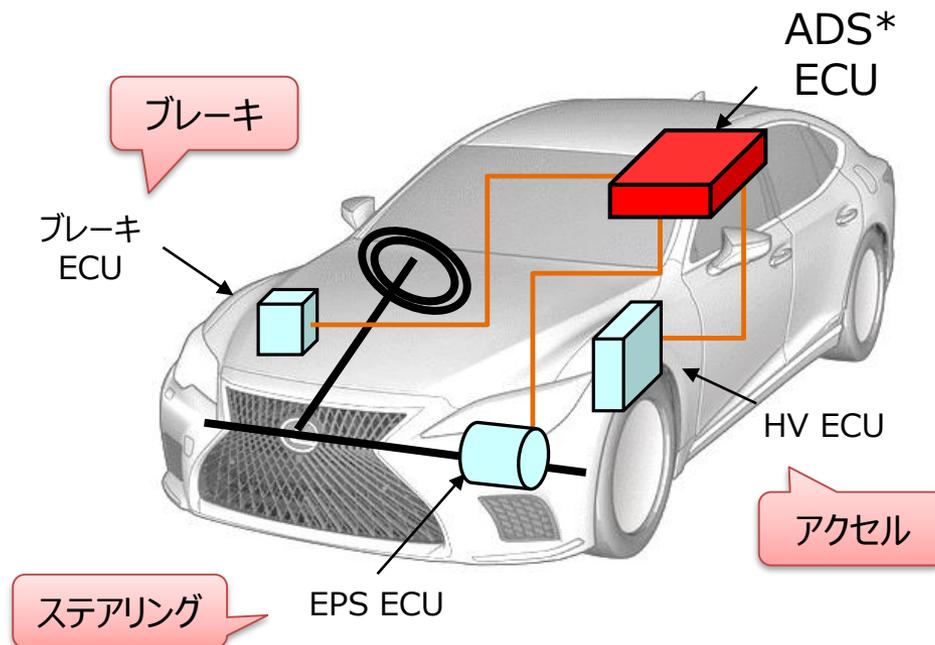
- ✓ 高速道路・自動車専用道路の走行を支援
  - 車線・車間維持
  - 車線変更（追い越し、本線からの分岐）



代表的な支援シーン

## ■ コントローラ（CTL）

- Advanced Drive の経路計画に追従した走行
- ステアリング、アクセル、ブレーキをドライバに代わって制御



\* Advanced Drive System



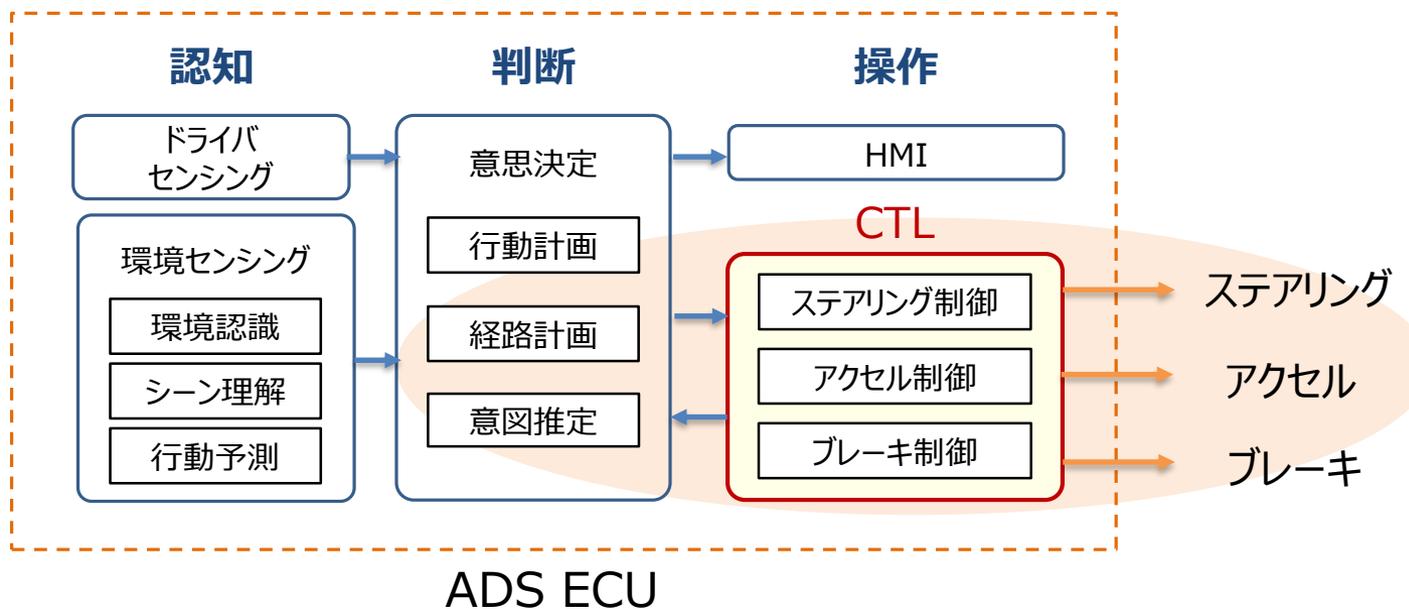
## ■ CTLの機能

機能	特徴
ステアリング、アクセル、ブレーキ制御	自然で滑らかな運転を実現
SWアップデート (ADS ECU機能)	新機能追加・市場フィードバックに対応

\* SW : ソフトウェア

## ■ システム構成

- 認知、判断して生成した経路計画に追従するようCTLで制御



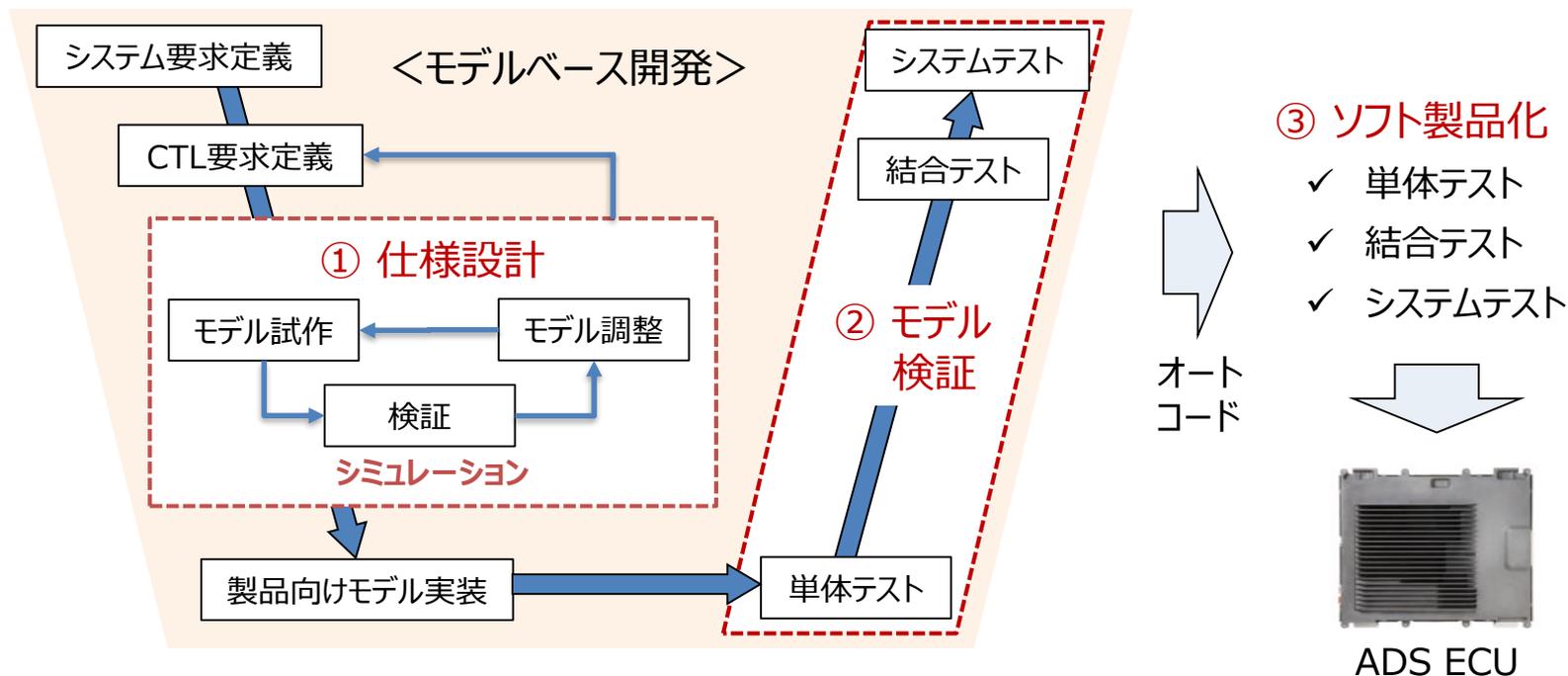


## ■ モデルベース開発を採用

- モデル\*で仕様設計・検証後、製品化

モデル\* :  
制御モデル(Simulinkモデル)

- ① 仕様設計 : モデル試作、検証 (シミュレーション) を繰り返し設計
- ② モデル検証 : 製品向けの実装モデルを作成し実車で検証
- ③ ソフト製品化 : 実装モデルをオートコードで製品化



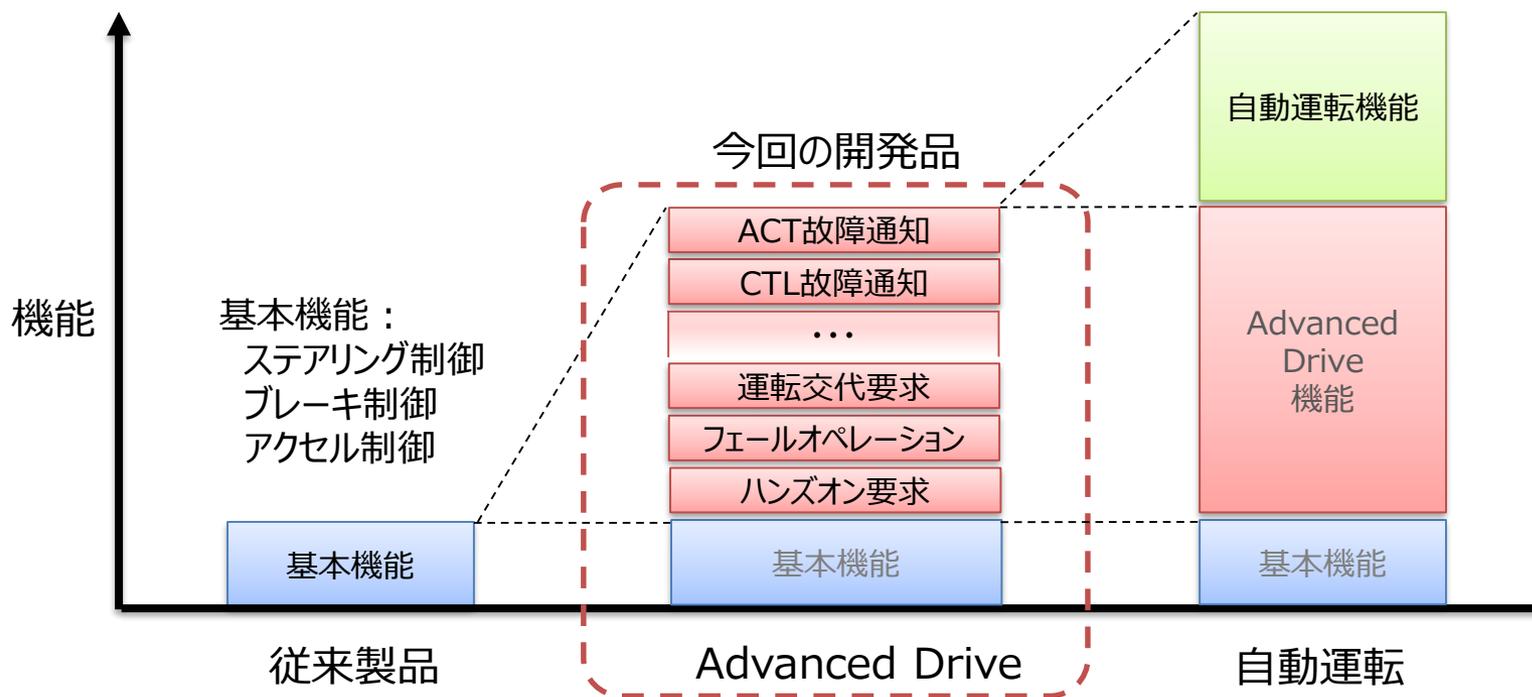


1. 担当製品と開発手法
2. 開発の変化と課題
3. 解決方法
4. 実施結果
5. 考察と今後の進め方
6. まとめ



## ■ 自動運転に向けた開発の加速

- CTLに対する急激な機能拡張
  - ✓ 運転交代要求、フェールオペレーション、ハンズオン要求など

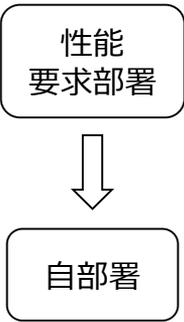
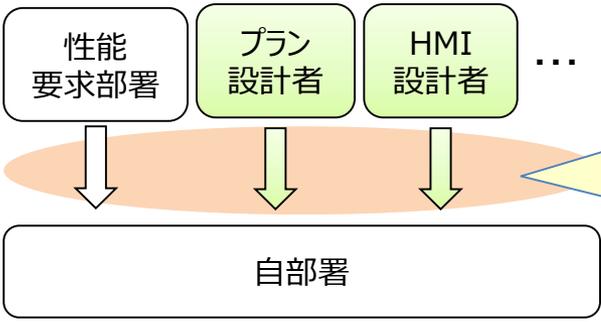


実現機能増加により、ステークホルダや試作も増加



## ■ 要求獲得の変化

- 実現機能の増加によるステークホルダの多様化
- 要求の伝え方が部署ごとにバラバラ (統一されていない)
- 試作回数が増えることによる要求発行回数の増加

観点	従来製品	Advanced Drive
ステークホルダ		
方法	文書	文書、メール、打合せ
頻度	1回	複数回

問題①

要求獲得の  
混乱と仕様の  
抜け・漏れ



## ■ モデルベース開発の導入

- 導入の狙い
  - ✓ 仕様設計の高速化
- 制御モデルの特徴
  - ✓ シミュレーションで検証が可能
  - ✓ モデルの意図・目的が分かりづらい



<開発の懸念点>

- ✓ 機能増加による開発者増員
- ↓
- ✓ モデルの理解不足による生産性低下

問題②

制御モデル特有の難解さ

## ■ リリース頻度の増加

- 開発中：
  - ✓ 新規機能が多いため、試作回数増加
- 開発後（発売後）：
  - ✓ 新機能追加のSWアップデート



<開発の懸念点>

- ✓ 制御モデルの修正回数の増加
- ↓
- ✓ 変更漏れ、干渉のリスクの増大

問題③

変更点・影響範囲の検討不足



## ■ 問題

- ① 要求獲得の混乱と仕様の抜け・漏れ
- ② 制御モデル特有の難解さ
- ③ 変更点・影響範囲の検討不足



## ■ 課題

### <仕様化の課題>

- 要求と制御モデルのトレーサビリティの確保
- 制御モデルの仕様化

### <開発プロセスの課題>

- 派生開発を配慮したプロセスへの変更

仕様化と開発プロセスの課題に取り組む



1. 担当製品と開発手法
2. 開発の変化と課題
- 3. 解決方法**
4. 実施結果
5. 考察と今後の進め方
6. まとめ



## ■ 課題

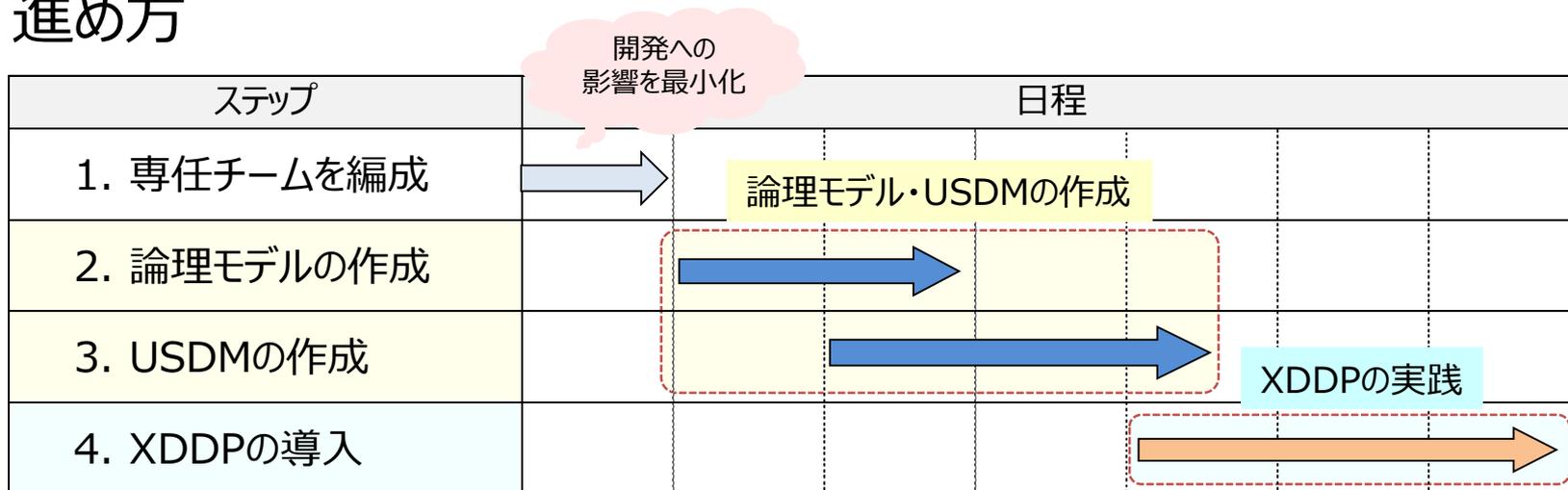
- 要求と制御モデルのトレーサビリティの確保
- 制御モデルの仕様化
- 差分開発を配慮したプロセスへの変更

## ■ アプローチ

- 論理モデルの作成
  - ✓ 要求と制御モデルの対応をとる
- 機能仕様書の作成 ← USDM
  - ✓ 論理モデルの構造を活用

- 派生開発プロセスの導入 ← XDDP

## ■ 進め方

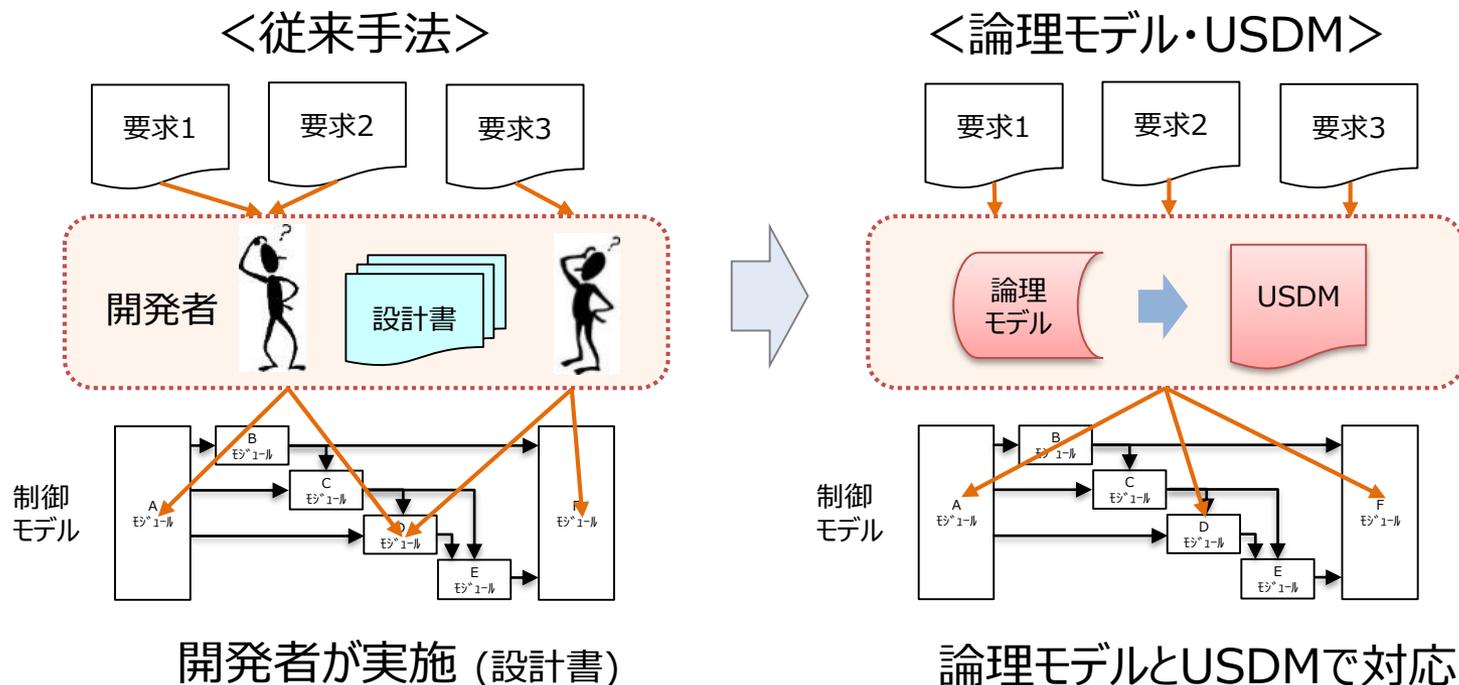




## ■ 目的

- 要求と制御モデルの対応の精度を向上する [論理モデル]
- 仕様の意図、内容を把握しやすくする [論理モデル・USDM]

## ■ 要求と制御モデルの対応



要求と制御モデルを論理モデル・USDMで対応をとる

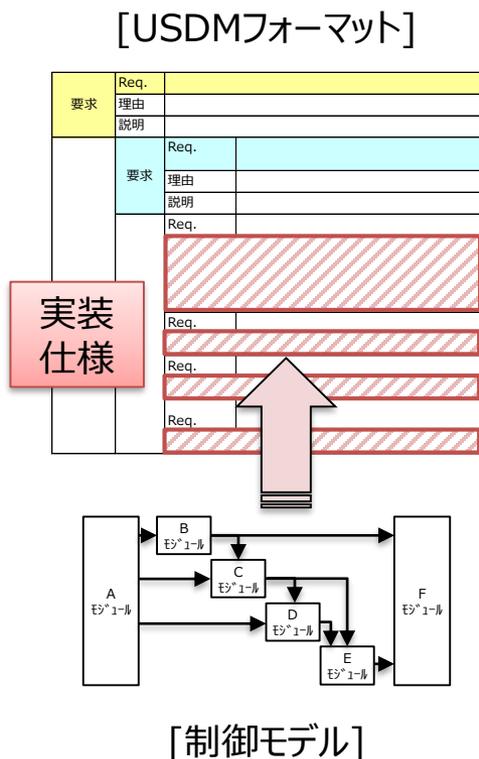


## ■ 方針

- USDMの階層構造を活用し、実装仕様から論理モデルを構成する

## ■ 方法

### (1) 実装仕様の抽出

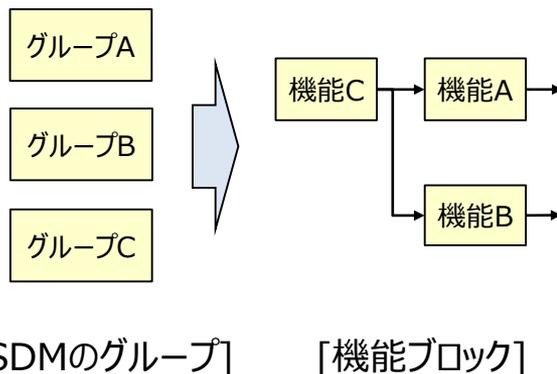


### (2) 機能ブロックの作成

- ✓ 実装仕様を層別 (要求を考慮)

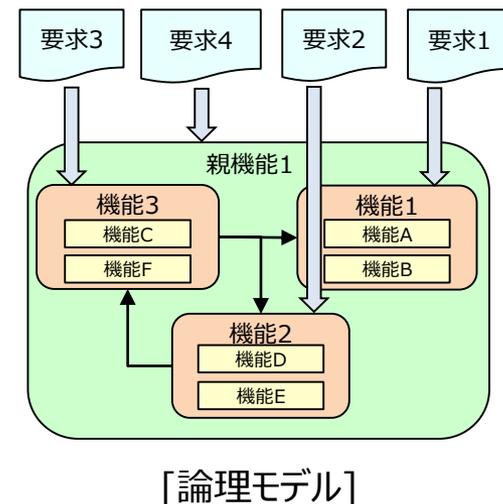
観点を規定して、論理的に層別する

- ✓ グループ単位で機能ブロックに変換



### (3) 論理モデルの構成

- ✓ 機能ブロックから論理モデルを作成
- ✓ 要求と機能の対応関係を調整





- 目的： 制御モデルの「機能仕様書+TM」を作成する
- 方針： 論理モデルの構造に基づいてUSDМを再構成する
- 方法

### (1) USDМを再構成

- ✓ 論理モデルの構造の反映

### (2) 実装仕様の抽象化

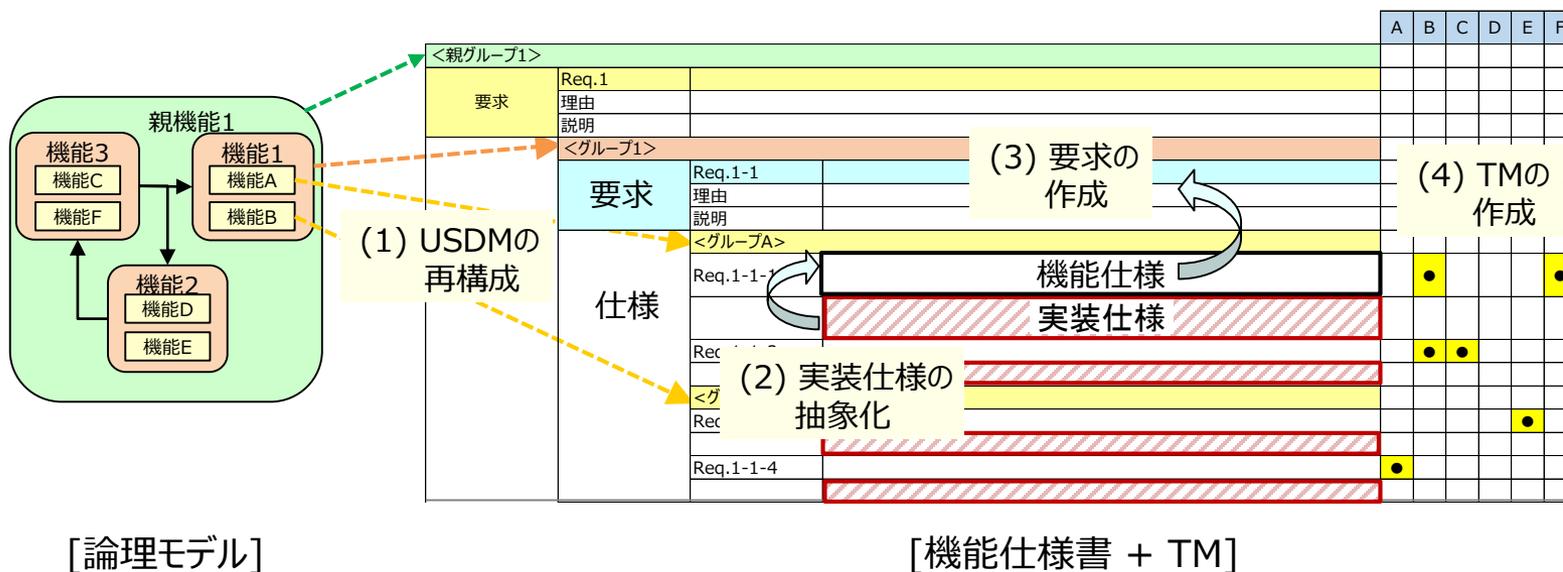
- ✓ 実装仕様から機能仕様に変換

### (3) 要求の作成

- ✓ グループ内の仕様を抽象化

### (4) TMの作成

- ✓ 機能仕様と制御モデルの対応





## ■ 目的

- 変更点、影響範囲を特定する精度の向上

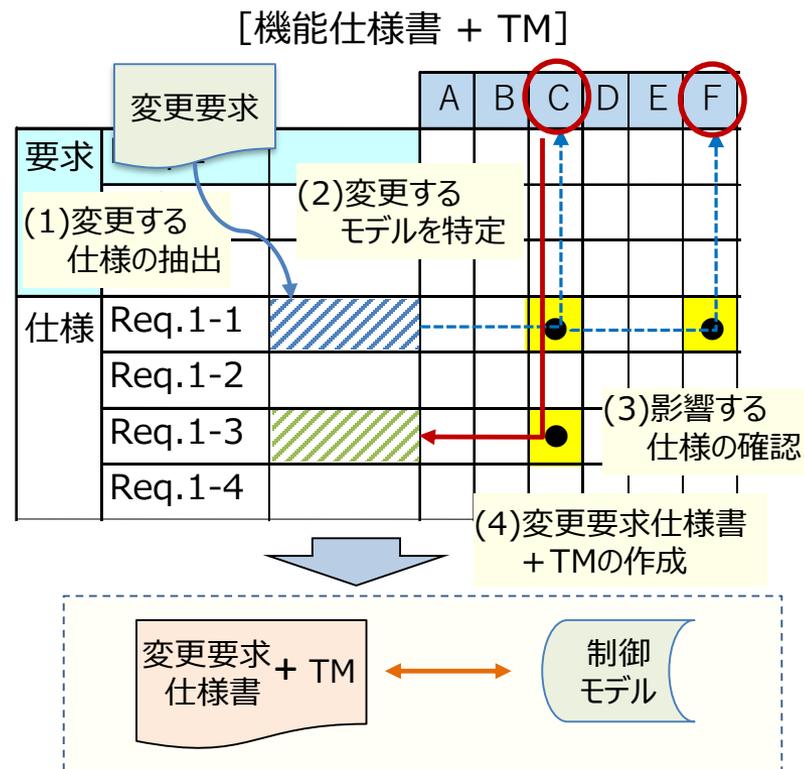
## ■ 方針

- 変更要求仕様書 + TMを導入
  - ✓ 機能仕様書を活用して変更要求仕様書を作成する

変更設計書：  
従来の設計書を使用

## ■ 変更要求仕様書の作成

- (1) 変更する仕様の抽出
  - ✓ 機能仕様書上の仕様から特定
- (2) 変更するモデルを特定
  - ✓ 仕様に対応するモデルをTM上で特定
- (3) 影響する仕様の確認
  - ✓ 変更するモデルに対応する仕様を特定
- (4) 変更要求仕様書 + TMの作成
  - ✓ (1)~(3) の情報を制御モデルで確認





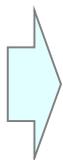
1. 担当製品と開発手法
2. 開発の変化と課題
3. 解決方法
- 4. 実施結果**
5. 考察と今後の進め方
6. まとめ



## ■ 仕様書の品質

- USDMの効果

- ✓ 仕様の意図、内容の把握が容易になった



- レビューの精度が上がった
  - 管理職、他のモジュール担当からの指摘の増加
- 開発に参画しやすくなった
  - USDM+TMで仕様とモデルの対応が可能に

	制御モデル／制御仕様書	USDM + TM																																																																
内容		<table border="1"><thead><tr><th></th><th></th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr></thead><tbody><tr><td>要求</td><td>Req.1</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td>理由</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td>説明</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>仕様</td><td>Req.1-1</td><td></td><td></td><td>●</td><td></td><td></td><td>●</td></tr><tr><td></td><td>Req.1-2</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td>Req.1-3</td><td></td><td></td><td>●</td><td></td><td></td><td></td></tr><tr><td></td><td>Req.1-4</td><td></td><td></td><td></td><td></td><td></td><td></td></tr></tbody></table>			A	B	C	D	E	F	要求	Req.1								理由								説明							仕様	Req.1-1			●			●		Req.1-2								Req.1-3			●					Req.1-4						
		A	B	C	D	E	F																																																											
要求	Req.1																																																																	
	理由																																																																	
	説明																																																																	
仕様	Req.1-1			●			●																																																											
	Req.1-2																																																																	
	Req.1-3			●																																																														
	Req.1-4																																																																	
特徴	<ul style="list-style-type: none"><li>• 制御ロジック中心の記述</li><li>• 仕様は数式、制御表現中心</li></ul>	<ul style="list-style-type: none"><li>• 要求と仕様を自然言語で表現</li><li>• TM で仕様とモデルを対応</li></ul>																																																																



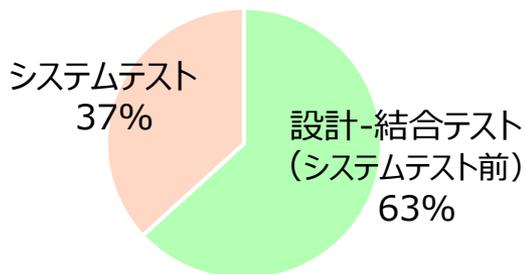
## ■ 制御モデルの品質

- 制御モデルの問題点を確認
  - ✓ USDM作成過程で要求と実装仕様の対応から問題点を抽出

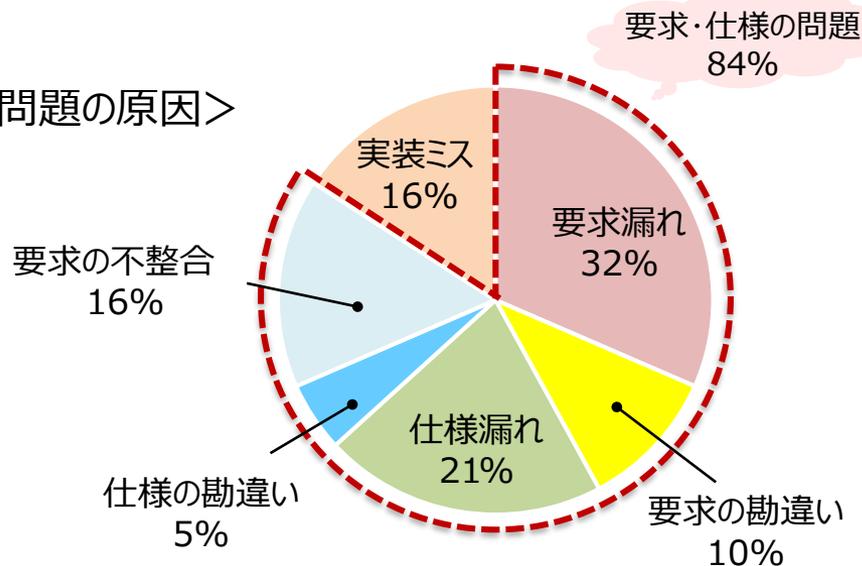


- 問題点の修正
  - ✓ 必要箇所の修正を完了
- ✓ 問題の発見工程 (流出した場合)

要求と制御モデルの  
トレーサビリティを確保



<問題の原因>



システムテストで発見する問題を  
USDM作成過程で抽出

要求・仕様の問題を解消し制御モデルの品質向上につながった



## ■ 変更点の品質 ~不具合数の変化~



XDDP 導入後は変更箇所の不具合なし



1. 担当製品と開発手法
2. 開発の変化と課題
3. 解決方法
4. 実施結果
5. 考察と今後の進め方
6. まとめ



## ■ 考察

- 仕様化・差分開発への取り組み
  - ✓ 各ステークホルダーの要求を抜け・漏れなく仕様書に反映できた
  - ✓ 変更点、影響範囲に対し体系的な検討ができるようになった
- 論理モデル・USDMによる仕様化
  - ✓ 実装仕様・論理モデルはUSDMの要求・仕様導出に有効である
  - ✓ 機能仕様書 + TMは派生開発の品質を飛躍的に向上させる

## ■ 今後の進め方

- 開発者の仕様化技術向上
  - ✓ 実践を通じたトレーニングを行い、継続的な能力向上を進める
- 検証プロセスの改善
  - ✓ 変更要求仕様書、機能仕様書からのテスト設計方法を検討する



1. 担当製品と開発手法
2. 開発の変化と課題
3. 解決方法
4. 実施結果
5. 考察と今後の進め方
6. まとめ



## ■ 背景

- Advanced Drive 実現 → 開発項目の急増
  - ✓ ステークホルダ、試作回数の増加 / モデルベース開発の導入

## ■ 開発の課題

- 要求とモデルのトレーサビリティを確保した仕様化
- 派生開発プロセスへの移行

## ■ 解決策

- 論理モデル、USDМによる機能仕様書作成
- XDDP (変更要求仕様書 + TM) の導入

## ■ 実施結果

- 仕様書、制御モデルの品質が向上し、不具合ゼロを達成

## ■ 今後の進め方

- 開発者の仕様化技術向上 / 検証プロセスの改善



# TOYOTA