

MATLAB EXPO 2018

# 設計・開発プロセス IEC 61508 機能安全規格 SIL 4適合に向けた 検証フローの構築

2018/10/30

株式会社 日立製作所  
制御プラットフォーム統括本部

中村 昌平

株式会社 日立製作所  
研究開発グループ  
生産イノベーションセンタ 回路システム研究部

上 蘭 巧

## 目次

---

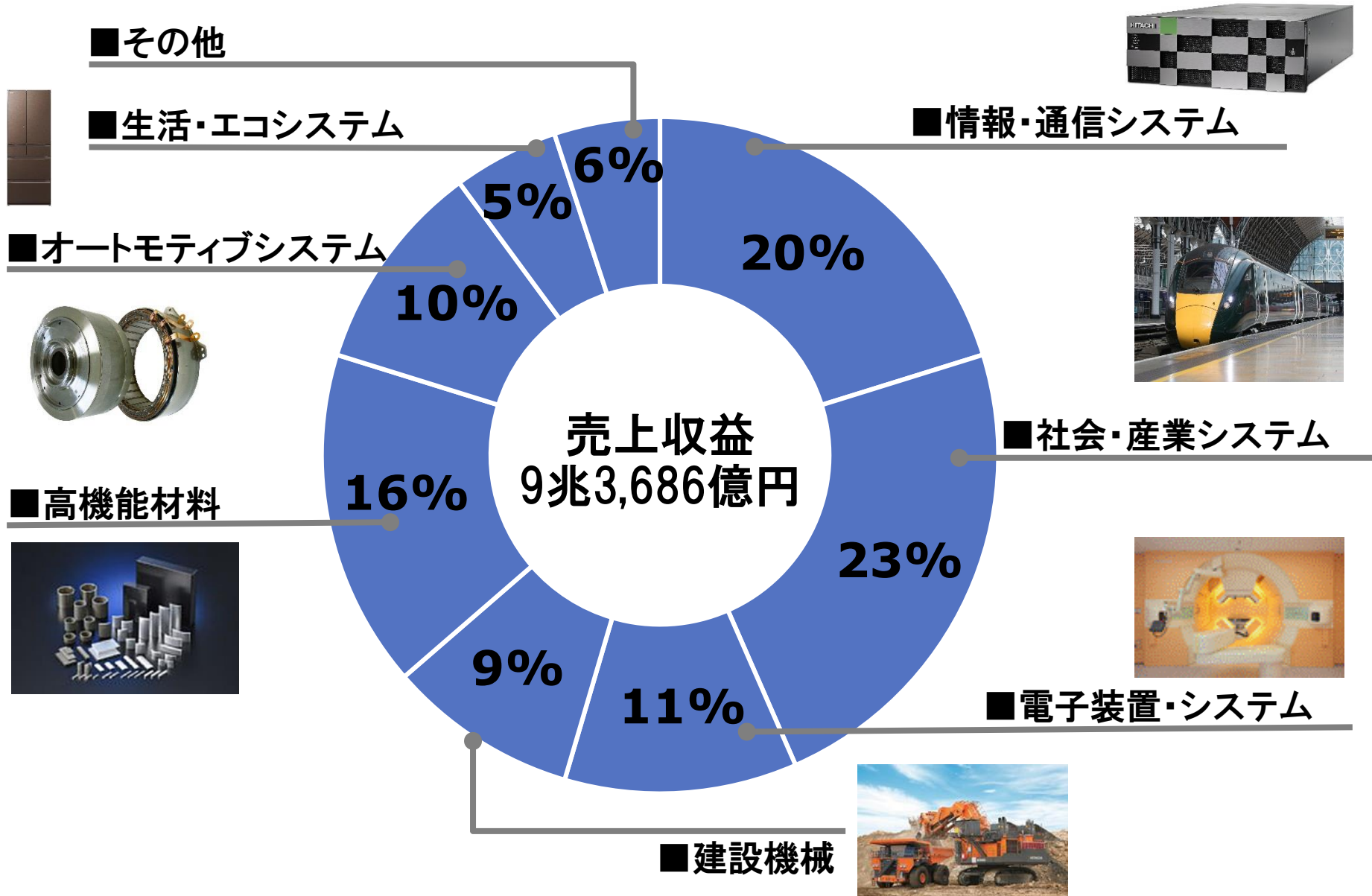
- 1 日立と開発製品のご紹介
- 2 機能安全とキーとなる要件
- 3 機能ブロックの開発
- 4 SIL 4 設計開発プロセス
- 5 Simulink®への要望
- 6 まとめ

## 目次

---

- 1 日立と開発製品のご紹介
- 2 機能安全とキーとなる要件
- 3 機能ブロックの開発
- 4 SIL 4 設計開発プロセス
- 5 Simulink®への要望
- 6 まとめ

# 1-1 日立について(2017年度)



## 社会・産業システム

### <主な製品・サービス>

- 産業・流通ソリューション
- 水ソリューション
- 産業用機器
- エレベーター
- エスカレーター
- 鉄道システム
- 原子力・自然エネルギー発電システム
- 送変電システム



# 1-3 開発製品のご紹介

## ν COSS S-zero (ニューコス エスゼロ)

- ・ミッションクリティカルな産業向け機能安全コントローラ
- ・TÜV Rheinlandから機能安全規格IEC 61508 SIL 3の認証を取得
- ・設計・開発プロセスはSIL 4に適合

認証番号968/FSP 1119.07/18

(システム設計をSIL 4に対応させることで、  
SIL 4が要求されるシステムにも使用可能)

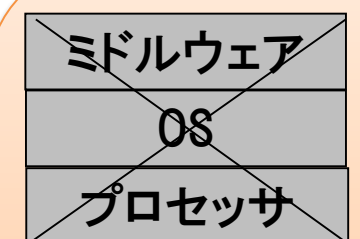


TÜV Rheinland:ドイツの国際的な第三者認証機関

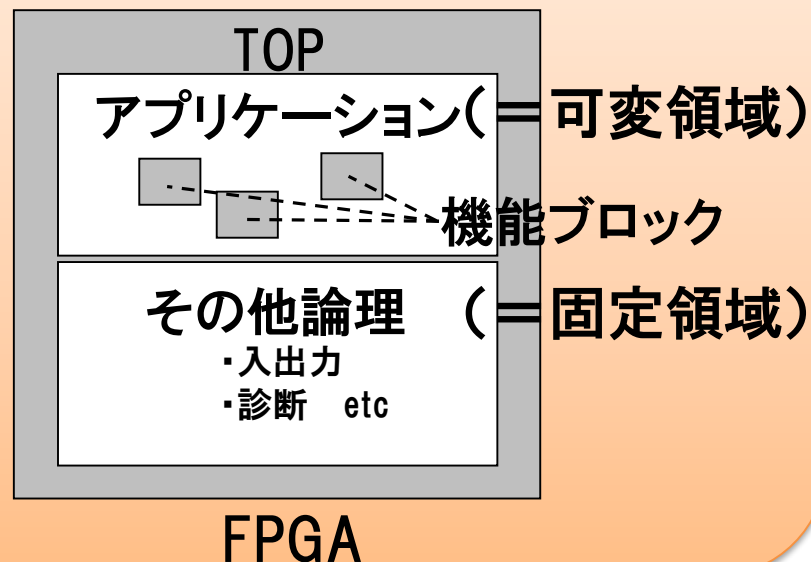
「ν COSS S-zero(ニューコス エスゼロ)」は、株式会社日立製作所の登録商標です

## 1-4 v COSS S-zeroの特徴

- 各モジュールはFPGAを実装しており、プロセッサ/ファームウェア/OS/ミドルウェアを使用しない**ハードワイヤードアーキテクチャ**
- FPGA内部は**可変領域**と**固定領域**に分けられる
- 機能ブロックを組合わせて**可変領域に配置しアプリケーションを実現する
- 固定領域は入出力、診断論理などであり、アプリケーションに依存しない



非搭載  
(ソフトウェアレス)



## 目次

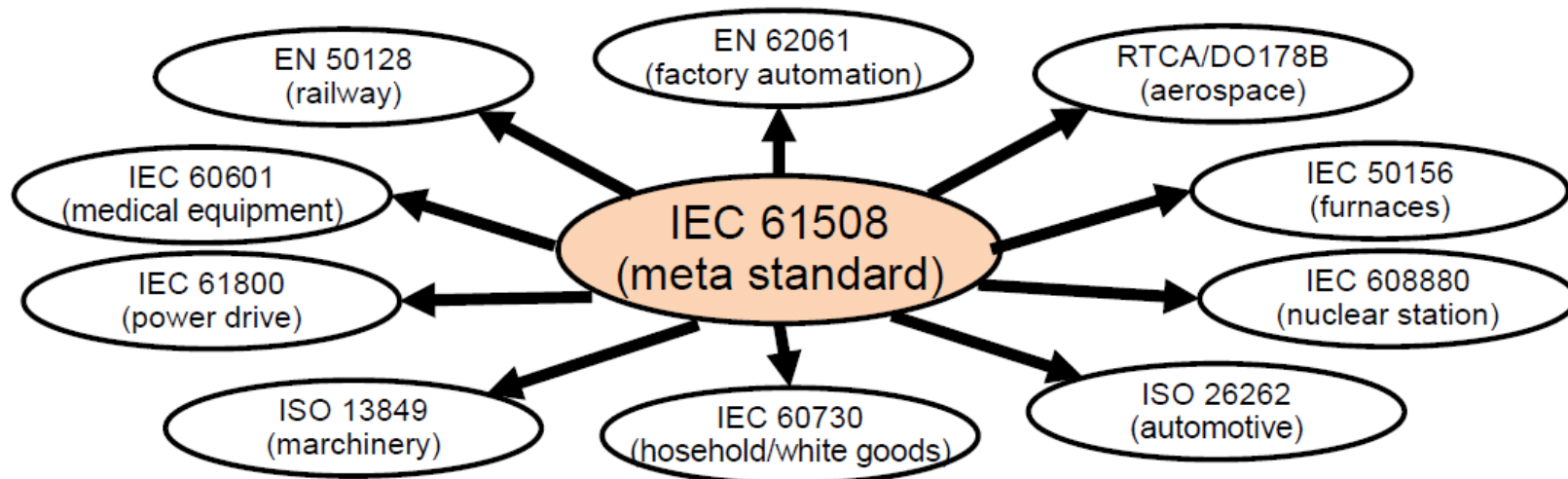
---

- 1 日立と開発製品のご紹介
- 2 機能安全とキーとなる要件
- 3 機能ブロックの開発
- 4 SIL 4 設計開発プロセス
- 5 Simulink®への要望
- 6 まとめ



### IEC 61508

#### IEC(国際電気標準会議)が制定した基本安全規格



### SIL(Safety Integrity Level)

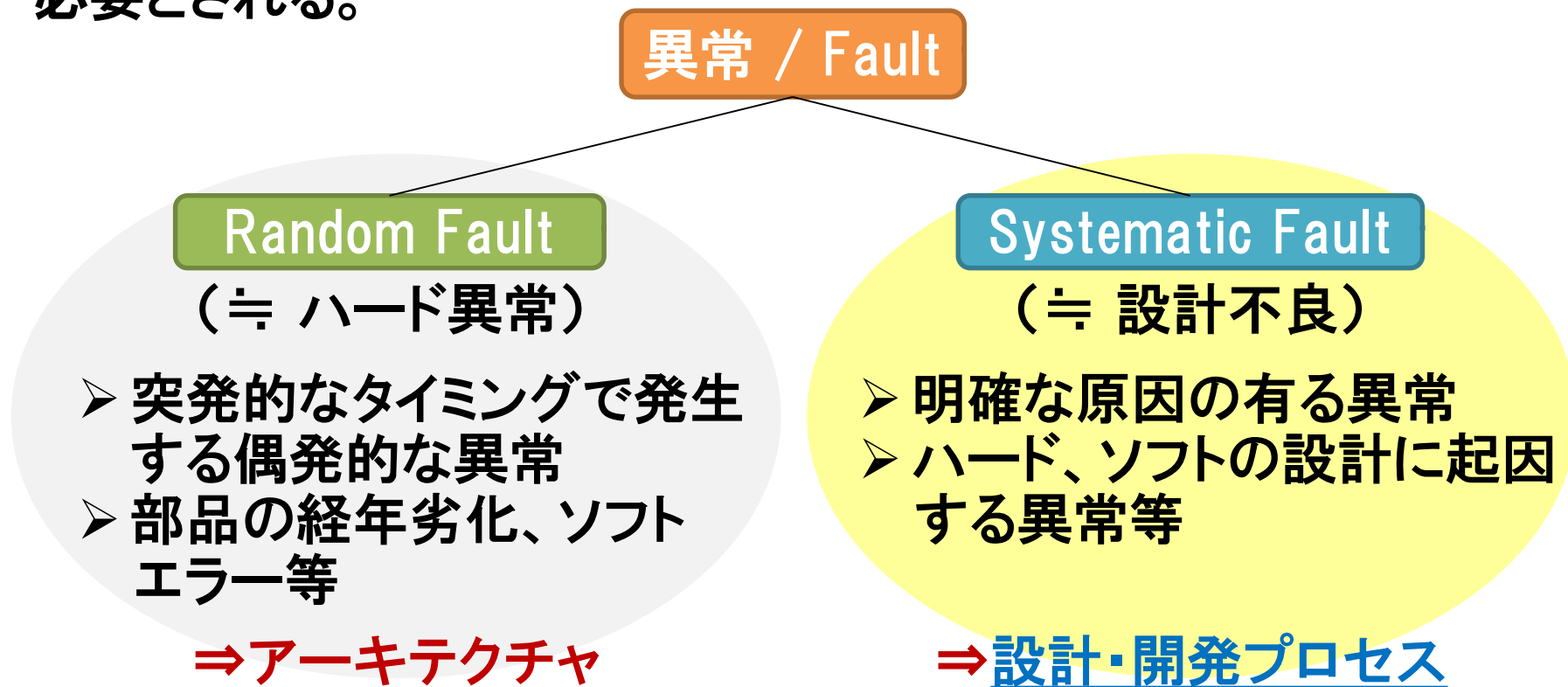
- ・システムの安全性を表す尺度
- ・4段階あり、SIL 4が最高水準

| 安全度水準<br>(SIL) | 安全機能の作動要求時の<br>危険側機能故障平均確率 |
|----------------|----------------------------|
| 4              | $10^{-5}$ 以上 $10^{-4}$ 未満  |
| 3              | $10^{-4}$ 以上 $10^{-3}$ 未満  |
| 2              | $10^{-3}$ 以上 $10^{-2}$ 未満  |
| 1              | $10^{-2}$ 以上 $10^{-1}$ 未満  |

出典: IEC 61508-1 Table 2

### IEC 61508で求められるFault対策

故障の原因となるFault(異常)にはRandom FaultとSystematic Faultの2種類が存在し、双方に対する安全対策(リスク低減策)が必要とされる。



## 2-3 SIL 4 適合のキーとなる要件

### SIL 4適合のキーとなる要件

検証済ソフトコアの使用

形式検証中心のフロー



### アプリケーション要求

浮動小数点演算が必要

ブロック図ベースの設計環境



浮動小数点演算を含む  
機能ブロックライブラリの開発

MATLAB®/Simulinkを活用した  
SIL 4設計開発プロセスの構築

## 2-3 SIL 4 適合のキーとなる要件

### SIL 4適合のキーとなる要件

検証済ソフトコアの使用

形式検証中心のフロー



### アプリケーション要求

浮動小数点演算が必要

ブロック図ベースの設計環境



浮動小数点演算を含む  
機能ブロックライブラリの開発

MATLAB®/Simulinkを活用した  
SIL 4設計開発プロセスの構築

## 目次

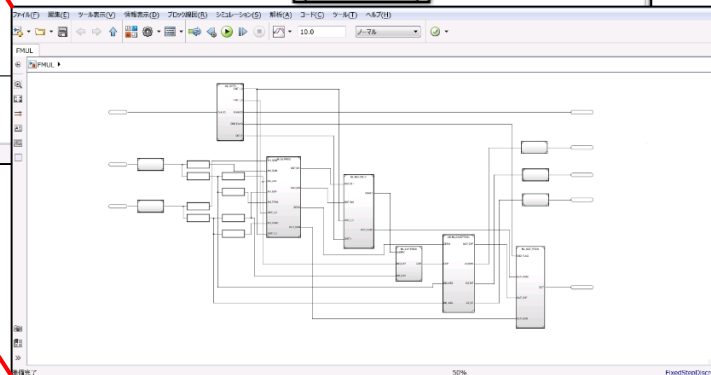
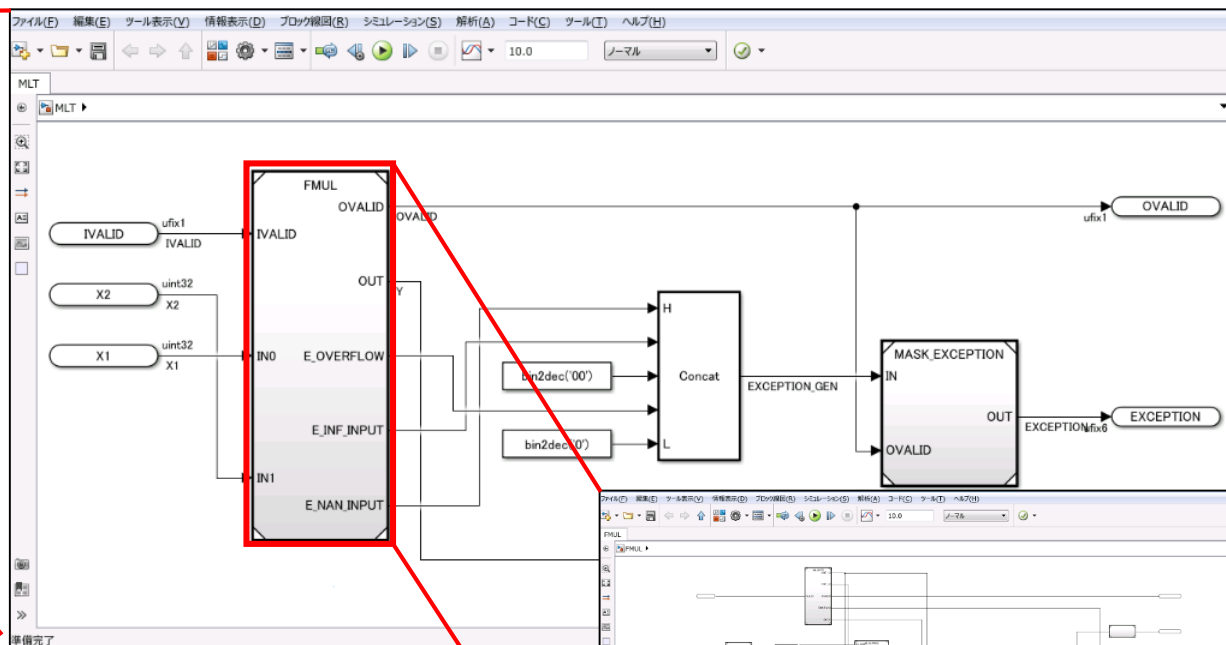
---

- 1 日立と開発製品のご紹介
- 2 機能安全とキーとなる要件
- 3 機能ブロックの開発**
- 4 SIL 4 設計開発プロセス
- 5 Simulink®への要望
- 6 まとめ

# 3-1 機能ブロックライブラリの開発

## 浮動小数点演算を含む機能ブロックライブラリをSimulinkで開発

- ・浮動小数点演算ブロック $※(+, -, \times, \div, \sqrt{\dots})$  ※ IEEE 754準拠
- ・デジタル機能ブロック(AND, OR, Flip Flop, ...)
- ・その他機能ブロック(Switch, Timer, ...) など60種以上



例)浮動小数点  
乗算器

＊開発当初、HDL Coder™は浮動小数点のHDLコード生成に  
未対応だったが、R2016bから対応可能となっている。

## 目次

---

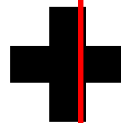
- 1 日立と開発製品のご紹介
- 2 機能安全とキーとなる要件
- 3 機能ブロックの開発
- 4 SIL 4 設計開発プロセス**
- 5 Simulink®への要望
- 6 まとめ

## 4-1 SIL 4 適合のキーとなる要件

### SIL 4適合のキーとなる要件

検証済ソフトコアの使用

形式検証中心のフロー



### アプリケーション要求

浮動小数点演算が必要

ブロック図ベースの設計環境

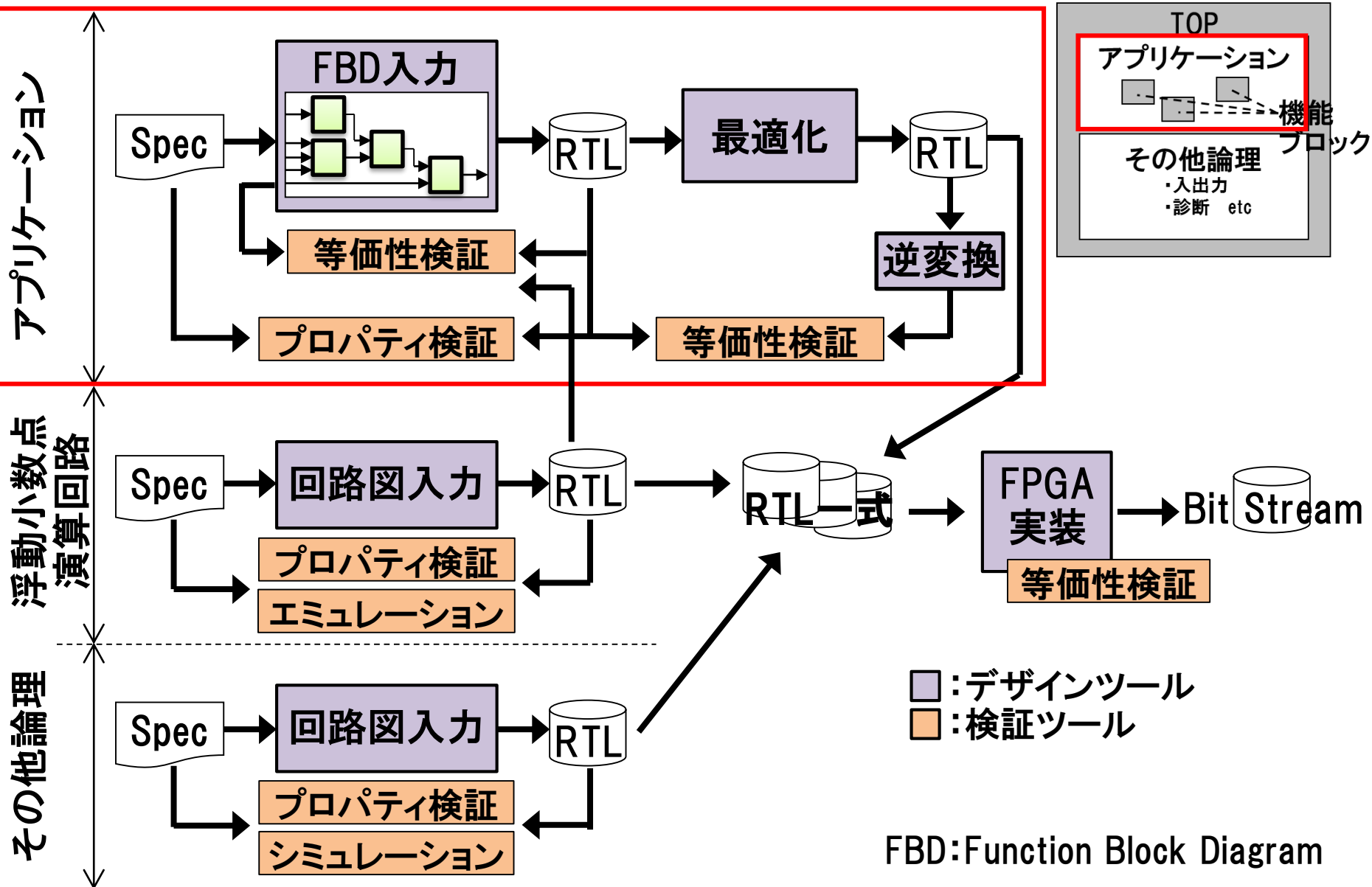


浮動小数点演算を含む  
機能ブロックライブラリの開発

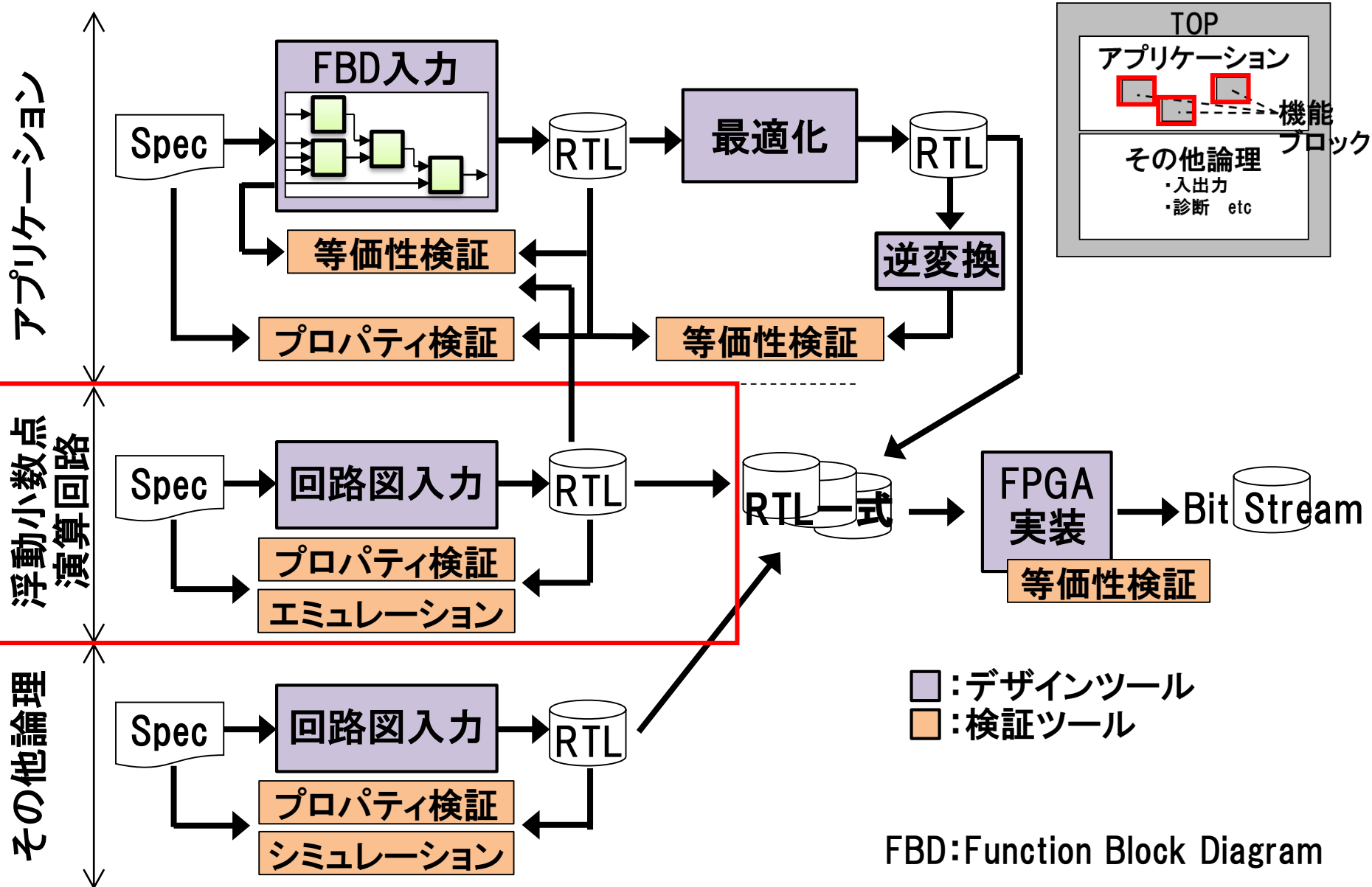
MATLAB®/Simulinkを活用した  
SIL 4設計開発プロセスの構築



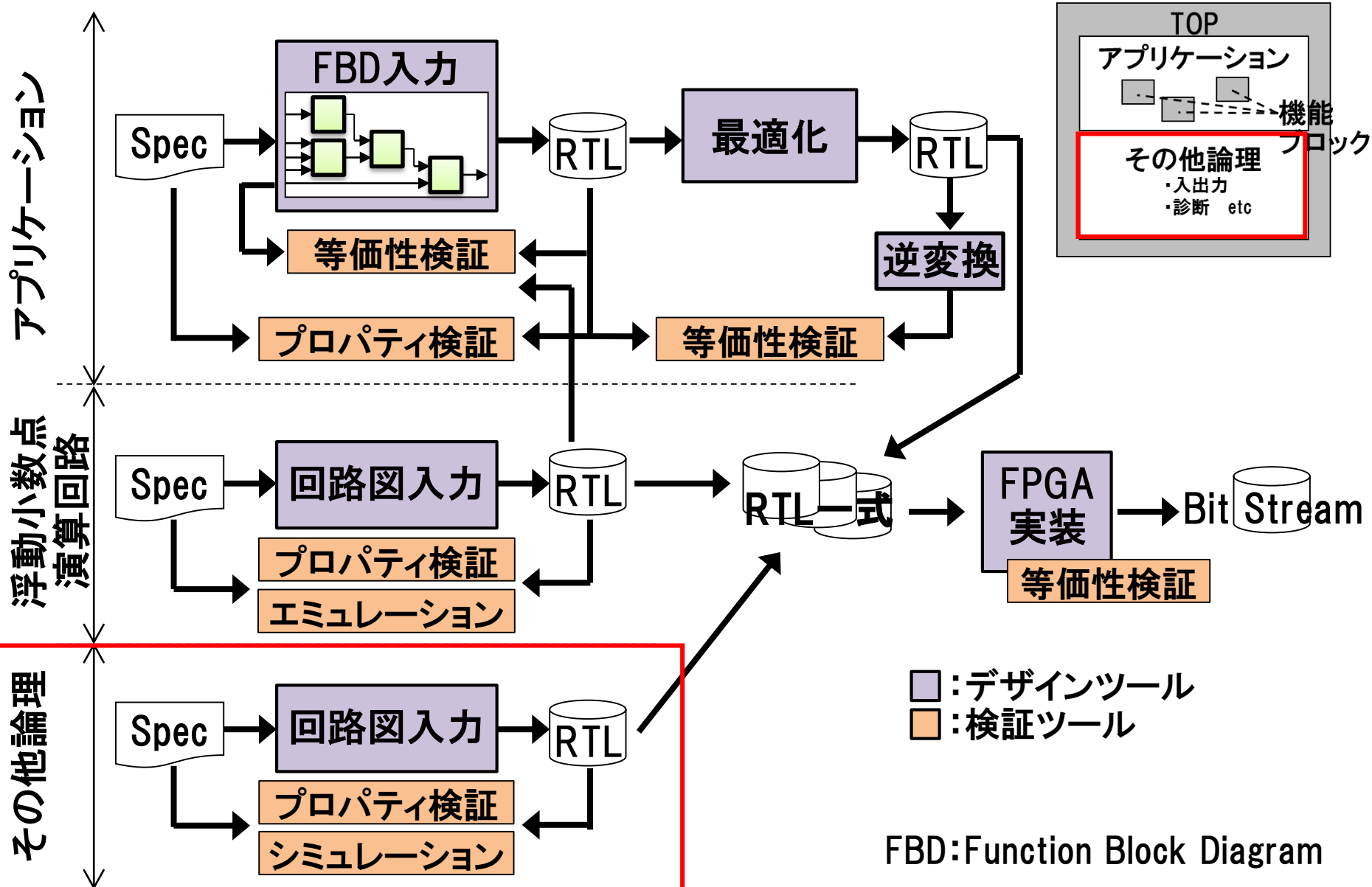
## 4-2 SIL 4 設計・開発プロセス



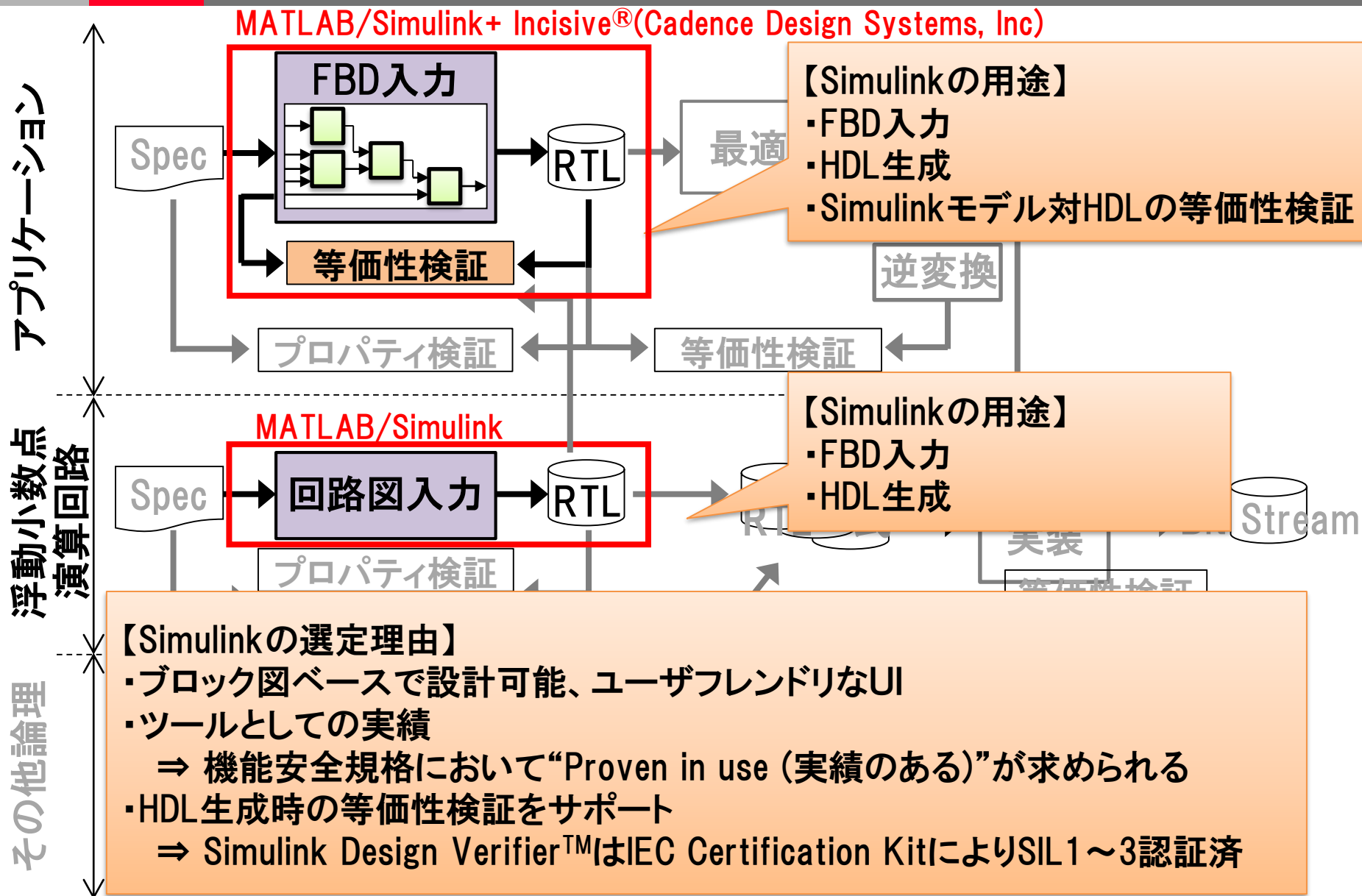
## 4-2 SIL 4 設計・開発プロセス



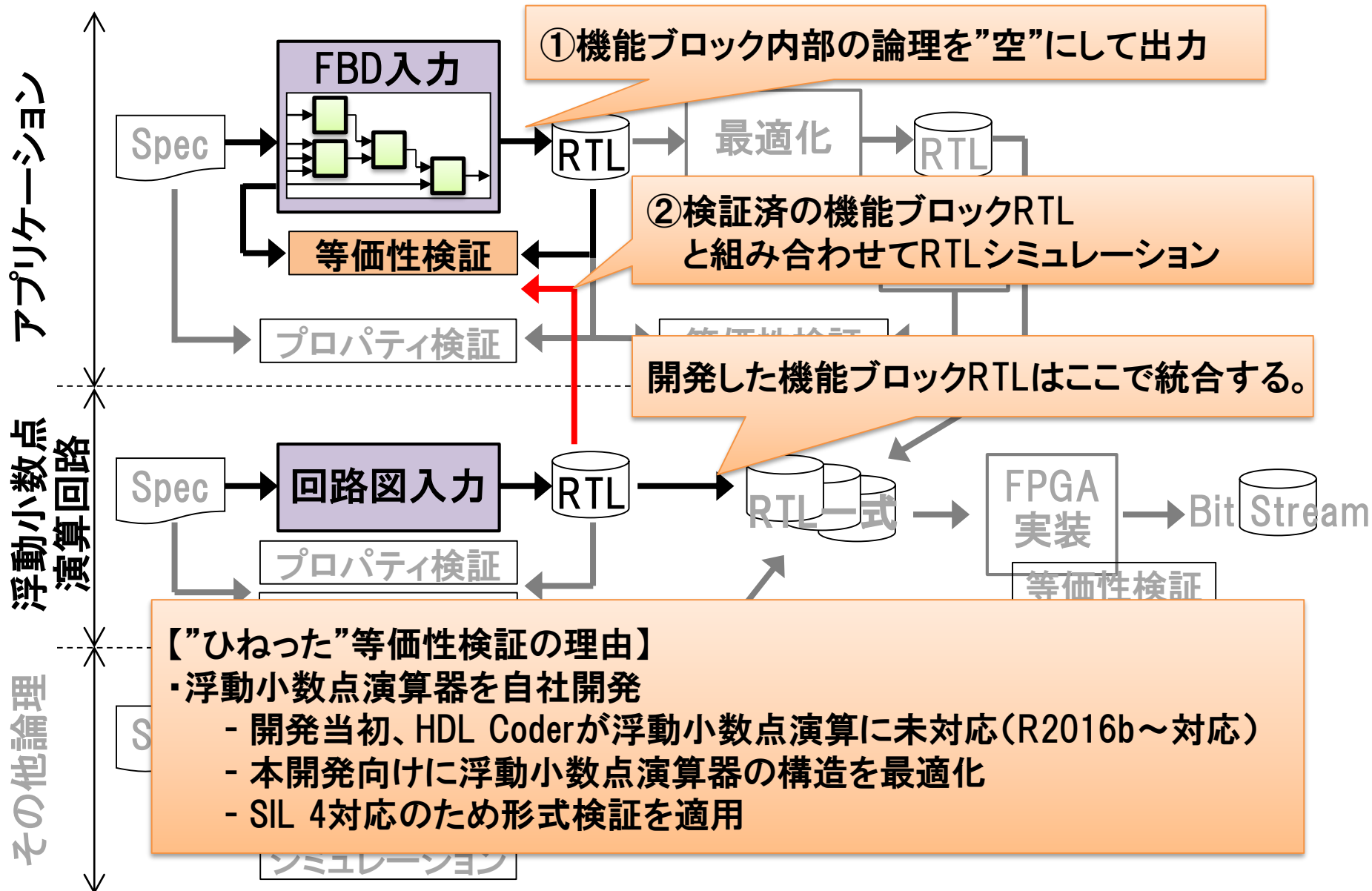
## 4-2 SIL 4 設計・開発プロセス



## 4-3 MATLAB/Simulinkの適用



## 4-4 ”ひねった”等価性検証



## 目次

---

- 1 日立と開発製品のご紹介
- 2 機能安全とキーとなる要件
- 3 機能ブロックの開発
- 4 SIL 4 設計開発プロセス
- 5 Simulink®への要望**
- 6 まとめ

### 良いところ

- ・別ツールを起動することなく、直ぐにモデルシミュレーションし、結果を確認可能  
(FBD入力ツールと、シミュレータ、波形ビューアが一体となっているのが便利)
- ・HDL Coderで出力したHDLが(比較的)可読性がある  
(作成したSimulinkモデルが比較的単純だったためかもしれませんが)

### 改善して欲しいところ

- ・今回の開発では、Simulinkのモデル参照を使って機能ブロック(FB)を配置している。  
多数(～10,000)のモデル参照を使うと一般的なPCではHDL生成に長時間かかる
- ・きれいな回路図を作成するという点でみると、
  - 配線がきれいに引きづらい(細かいことですが、配線を等間隔に引きづらい)
  - キャンバスを自由に使いすぎる  
(好きなだけFBを置けてしまう。キャンバスを制限したい)

## 目次

---

- 1 日立と開発製品のご紹介
- 2 機能安全とキーとなる要件
- 3 機能ブロックの開発
- 4 SIL 4 設計開発プロセス
- 5 Simulink®への要望
- 6 まとめ



- ・ミッションクリティカルな産業向け機能安全コントローラ  
    ↳ COSS S-zero(ニューコス エスゼロ)を開発
  - 機能安全規格IEC 61508 SIL 3の認証を取得
  - 設計・開発プロセスSIL 4適合  
(システム設計をSIL 4に対応させることで、  
SIL 4が要求されるシステムにも使用可能)
- ・ MATLAB/Simulink含むツールの適用、機能ブロックの開発、  
形式検証中心のフロー構築により、SIL 4適合かつユーザビリティに  
優れたアプリケーション設計が可能となった
- ・ ユーザーの利便性を向上させるため、Simulinkを利用した  
ツールを開発中

**END**



**設計・開発プロセス IEC 61508 機能安全規格 SIL 4適合に向けた  
検証フローの構築**

2018/10/30

株式会社 日立製作所  
制御プラットフォーム統括本部

中村 昌平

株式会社 日立製作所  
研究開発グループ  
生産イノベーションセンタ 回路システム研究部

上藺 巧

**HITACHI**  
Inspire the Next 

- ・検証対象論理の動作がアサーションを満足することを確認
- ・検証空間全体でアサーションを満足していればPASS、そうでなければFAIL
- ・制約をかけることで、不必要な検証空間を除外

アサーション/制約の例:

「ある時点でpが成立したならば、その後いつかqが成立する」  
「信号AとBは同時に'H'にならない」

