

OLYMPUS



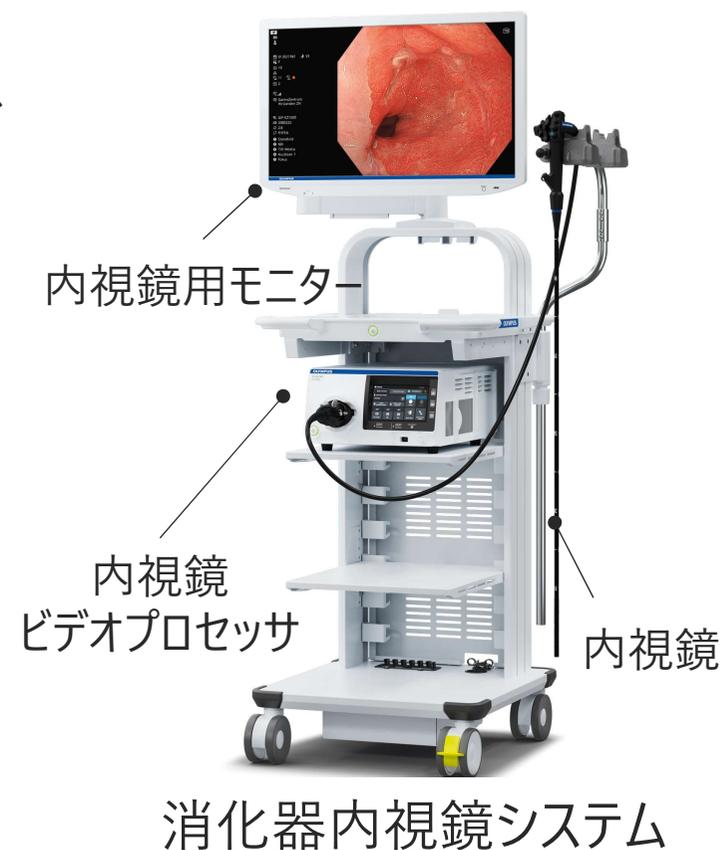
内視鏡ビデオプロセッサ開発におけるMBD適用事例報告

オリンパス株式会社 | 先進画像システム技術 佐藤朋也 | 2020年10月2日

本報告の概要

内視鏡ビデオプロセッサの画像処理機能開発へのMBD適用事例を報告

- モデルベースデザイン(MBD)は、自動車業界で広く一般的に用いられており、
 - モデルによる設計とシミュレーション
 - モデルから実装用コード(CコードやFPGAコードなど)を自動生成を特徴とした開発スタイル
- これまで、内視鏡ビデオプロセッサの画像処理機能の開発に対して、MBD適用による効果・課題を明らかにするために様々な検討を実施
- 本報告では、主に以下の事例を報告
 - ① 従来設計手法との比較検討事例
 - ② 新規画像処理機能開発へのMBD適用事例



01 オリンパスのご紹介

02 MBD適用のモチベーション

03 画像処理機能開発におけるMBDフロー

04 適用事例報告①：従来開発との比較

05 適用事例報告②：製品開発への適用

06 まとめ

01 オリンパスのご紹介

02 MBD適用のモチベーション

03 画像処理機能開発におけるMBDフロー

04 適用事例報告①：従来開発との比較

05 適用事例報告②：製品開発への適用

06 まとめ

オリンパスの事業



医療分野

内視鏡事業:Endoscopic Solutions Division
治療機器事業:Therapeutic Solutions Division



科学事業:

Scientific Solutions Division



映像事業:

Imaging Division

オリンパスは医療分野から、ライフサイエンス・産業、カメラ・オーディオ分野に至るまで、様々なソリューションを提供しています。東京をはじめとし、全世界約40の国々、35,000人以上の従業員が活躍する企業です。病気の診断と治療に貢献する、生命科学の研究に寄与する、人々の安全を守る、そして人生のかけがえのない瞬間をとらえる。100年を越え、オリンパスはこれからも、世界の人々の健康と安心、心の豊かさの実現のために、歩みを進めてまいります。

事業別売上高

436 億円 5.5%

映像事業

1,052 億円 13.2%

科学事業

2,161 億円 27.1%

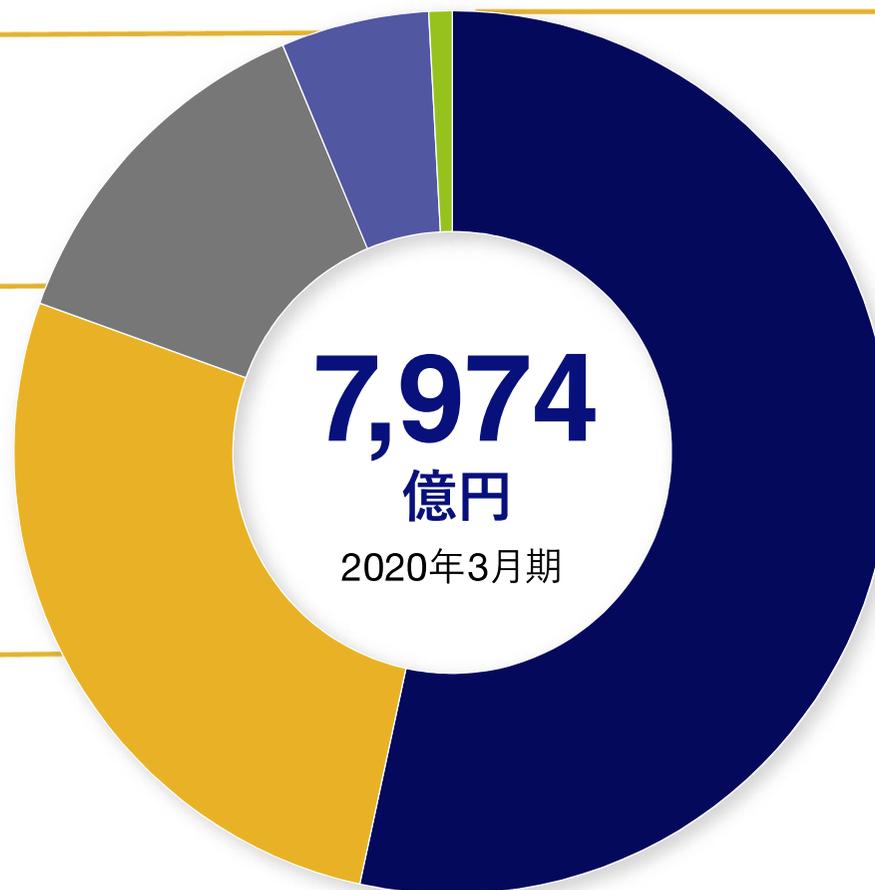
治療機器事業

68 億円 0.9%

その他

4,257 億円 53.4%

内視鏡事業



内視鏡事業 : Endoscopic Solutions

消化器内視鏡システム



消化器内視鏡
ビデオスコープシステム

外科内視鏡システム



4K内視鏡システム



3D内視鏡システム



リプロセス



内視鏡自動洗浄消毒装置

システム インテグレーション



手術室

医療サービス



リペアセンター

次世代の内視鏡システムの導入

内視鏡による診断治療の質・検査効率向上を目指した新機能を搭載
次世代の内視鏡システムを市場導入
がんなどの早期診断・低侵襲治療に貢献



次世代消化器内視鏡システム



次世代内視鏡ビデオプロセッサ



次世代内視鏡ビデオスコープ

01 オリンパスのご紹介

02 MBD適用のモチベーション

03 画像処理機能開発におけるMBDフロー

04 適用事例報告①：従来開発との比較

05 適用事例報告②：製品開発への適用

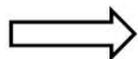
06 まとめ

医療機器開発の現状：システムの複雑化 1/2

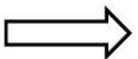
医療機器に用いられるSWは急激に複雑化している

Software in Medical Devices

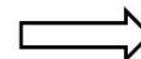
Complex software increasing rapidly in Medical Devices



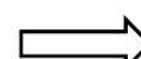
80,000 LOC*



170,000 LOC*



7 Million LOC*



Apps – software is a medical device

* LOC – Lines of Code.

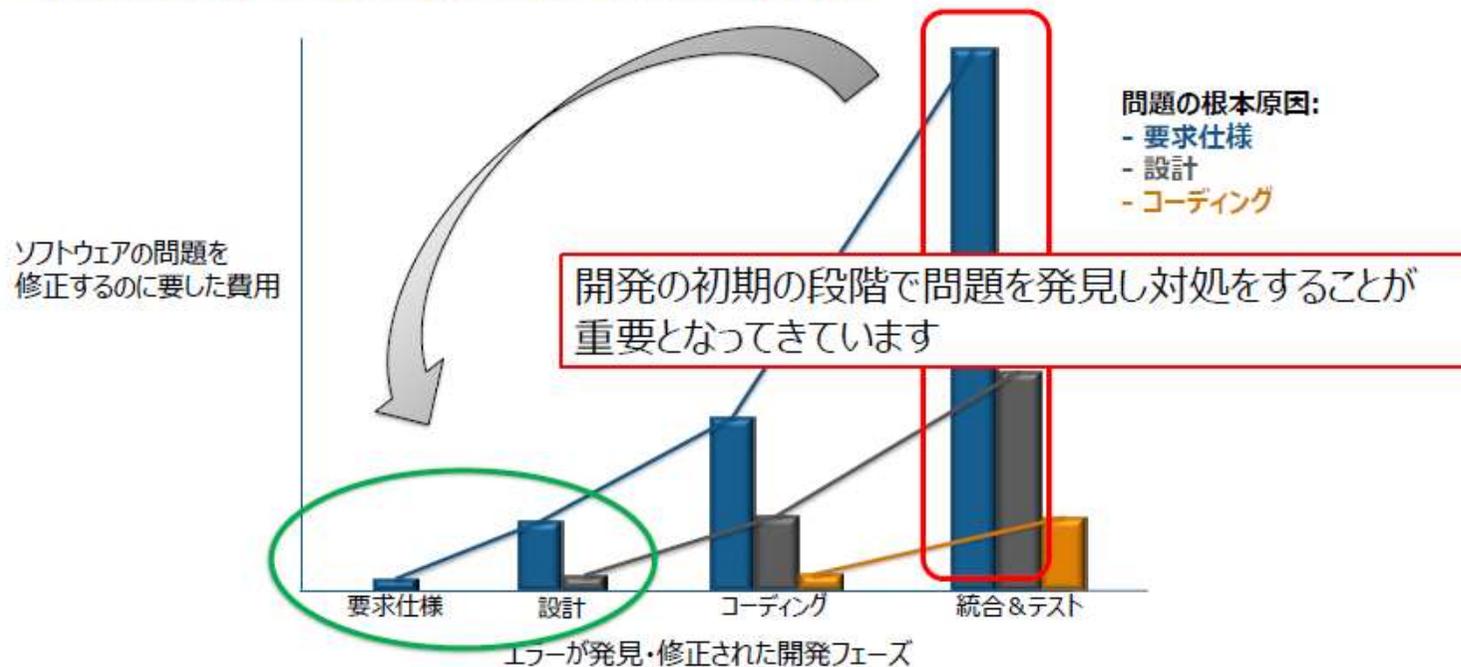
Source: Economist June 2012 article "When code can kill or cure". <http://www.economist.com/node/21556098>

医療機器開発の現状：システムの複雑化 2/2

複雑化するシステムにおいて、開発後期での手戻りが大きな問題

開発後期で問題が顕在化

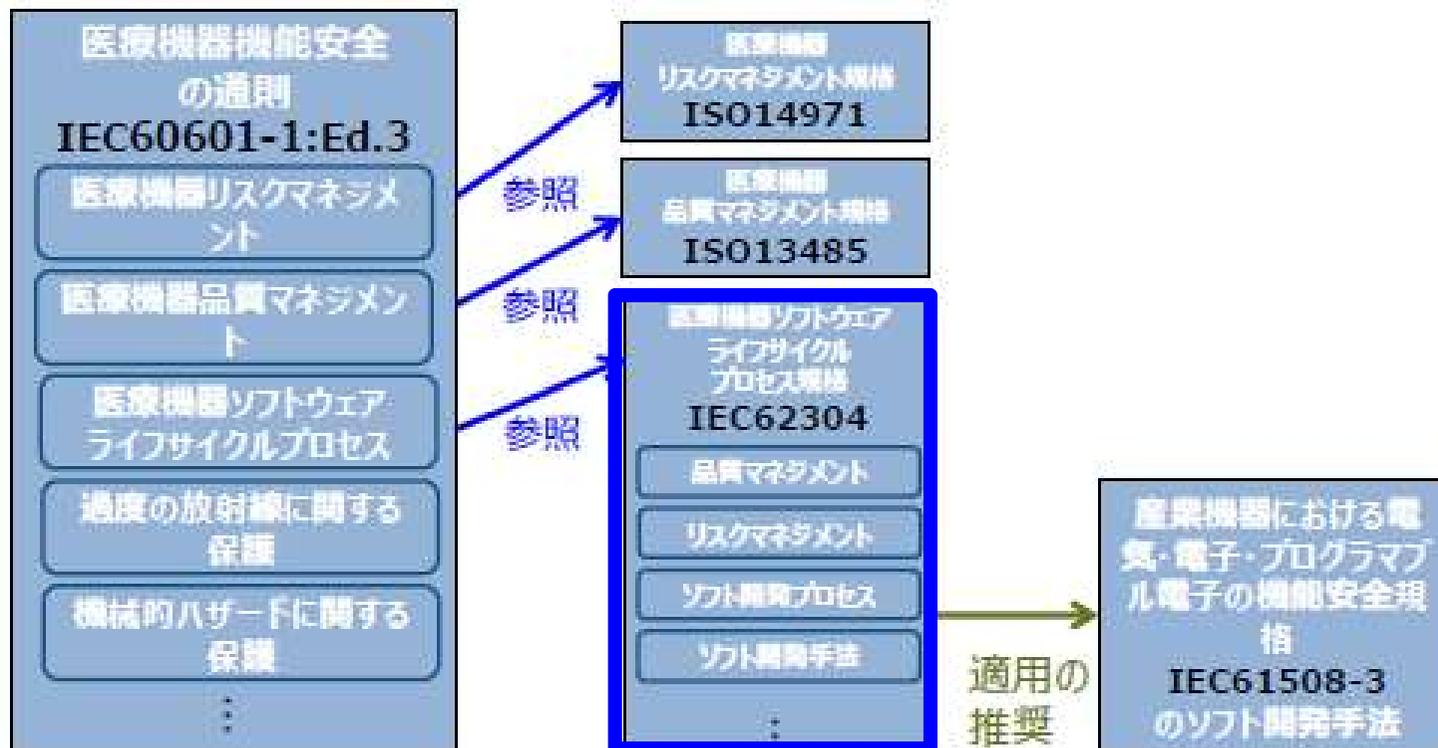
～下流で多くの不具合が発生して、手戻りが大きい～



Source: Return on Investment for Independent Verification & Validation, NASA, 2004

医療機器開発の現状：品質規格

FDA(US Food and Drug Administration)が
医療用ソフトウェア開発にIEC62304の適用を推奨



FDAがMBDの有効性について言及

IEC62304とMBD制御開発ワークフロー

Metrics for IEC 62304
The items highlighted in green can be automatically generated by Model-Based Design tools.

Software Development Process	Modeling and Simulation	Metrics
5.2 Requirements Analysis	Requirement analysis via modeling and simulation	Traceability of simulation test cases to functional requirements
5.3 Architectural design	Develop structural software models	Allocation of requirements to model components
5.4 Detailed design	Elaborate architectural model with behavior and software specifications	<ul style="list-style-type: none"> Model coverage during simulation testing Demonstrated compliance to modeling standards
5.5 Unit Implementation and Verification	<ul style="list-style-type: none"> Code generator Unit models and Software-in-the-loop (SIL) and Hardware-in-the-loop (HIL) testing 	<ul style="list-style-type: none"> Code ⇄ model ⇄ requirements traceability Unit model vs code behavior (back to back testing) Structural code coverage Coding standards compliance Absence of run-time errors (static analysis) within units
5.6 Integration and Integration Testing	Subsystem models and SIL testing	<ul style="list-style-type: none"> Subsystem simulation vs code behavior Absence of run-time errors at code interfaces
5.7 System Testing	System model and SIL testing	<ul style="list-style-type: none"> System simulation vs. code behavior Absence of run-time errors at OS and driver-level boundaries
5.8 Release	Model configuration management	<ul style="list-style-type: none"> Issue tracking/resolution Change management process conformance

※テクニカルアーツより抜粋

MBD制御開発ワークフロー



医療機器開発とMBD適用の事例報告

医療機器メーカーからMBDによる品質向上、及び効率化の事例報告が多数

Philips Healthcare Develops Smart Digital RF Power Subsystem for MRI Systems



Medrad社はMathWorksのツールによりMRI血管造影剤注入ポンプの安全性を確保



Challenge
Develop MRI syst
Solution
Use Sim and use predictal
Results

- Desi
- Trad
- Proc

課題

患者への薬育MRI血管造影

ソリューション
ポンプの圧力MathWorksの

結果

- 設計期間
- 名誉ある
- FDAの承認

[Link to user story](#)

[Link to user story](#)

ITK Engineering Develops IEC 62304-Compliant Controller for Dental Drill Motor with Model-Based Design



Dental drills featuring ITK Engineering's sensorless brushless motor control.

Challenge
Develop and implement field-oriented controller software for sensorless brushless DC motors for use in dental drills
Solution
Use Model-Based Design with Simulink, Stateflow, and Embedded Coder to model the controller and plant, run closed-loop simulations, generate production code, and streamline unit testing

- Results
- Development time halved
 - Hardware problems discovered early
 - Contract won, client confidence established

[Link to user story](#)

"Model-Based Design with Simulink enabled us to reduce costs and project risk through early verification, shorten time to market on an IEC 62304-certified system, and deliver high-quality production code that was first-time right."

Michael Schwarz
ITK Engineering

Model-Based Design導入による効果 第三者機関による市場調査結果

World Industry: MBD vs. non-MBD

World Industry: \$10,000/developer/month	MBD	Non-MBD
Developer time (Months)	13.4	13.1
% behind schedule	44.3%	49.9%
Months behind	3.7	3.7
% cancelled	10.1%	12.1%
Months lost to cancellation	4.2	4.7
SW developers/project	11.0	16.3
HW developers/project	8.9	10.9
Total project developers	19.9	27.3
Average developer months/project	266.2	358.1
Developer months lost to schedule	32.4	49.8
Developer months lost to cancellation	8.5	15.5
Total developer months/ project	307.1	423.4
Average developer cost/project	\$2,662,098	\$3,580,843
Average cost to delay	\$323,977	\$497,835
Total developer cost/project	\$2,986,075	\$4,078,677

Advantage of Model-Based Design **36.6%**

出展: Model-Based Design in Practice – A Survey of Outcomes for Engineers and Business Leaders/Embedded Market Focus

自社開発へのMBD適用のモチベーション

✓ 医療機器開発の現状

- システムの複雑化
- 品質向上と効率化

✓ 外部環境の変化

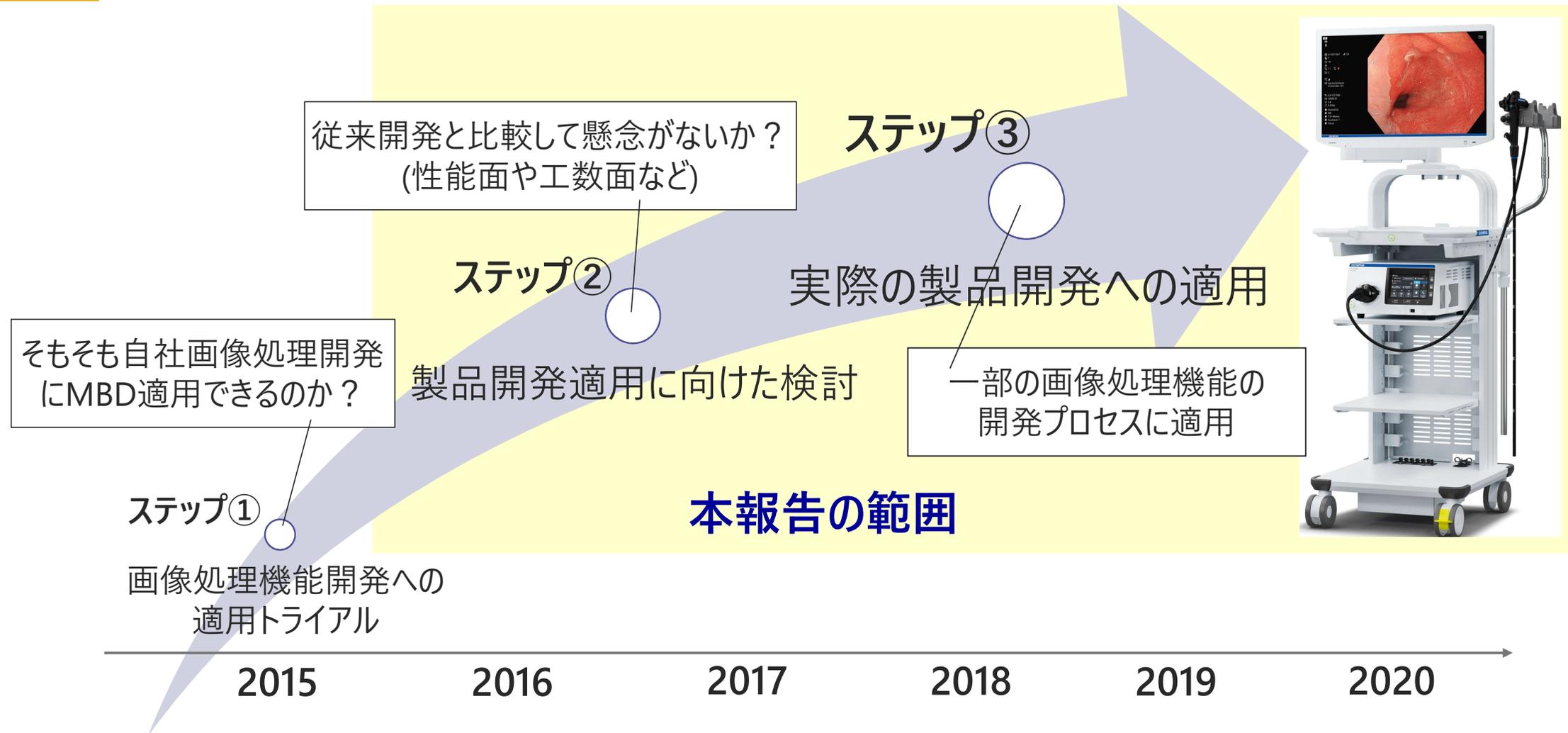
- 2016年 FDAがMBDの有効性を言及
- 医療機器メーカーからもMBDの取り組み事例報告多数

内視鏡ビデオプロセッサ開発へのMBD適用検討をスタート
(画像処理アルゴリズム開発からHW(FPGA)実装への適用)



消化器内視鏡システム

これまでの自社取り組みと本報告の範囲



01 オリンパスのご紹介

02 MBD適用のモチベーション

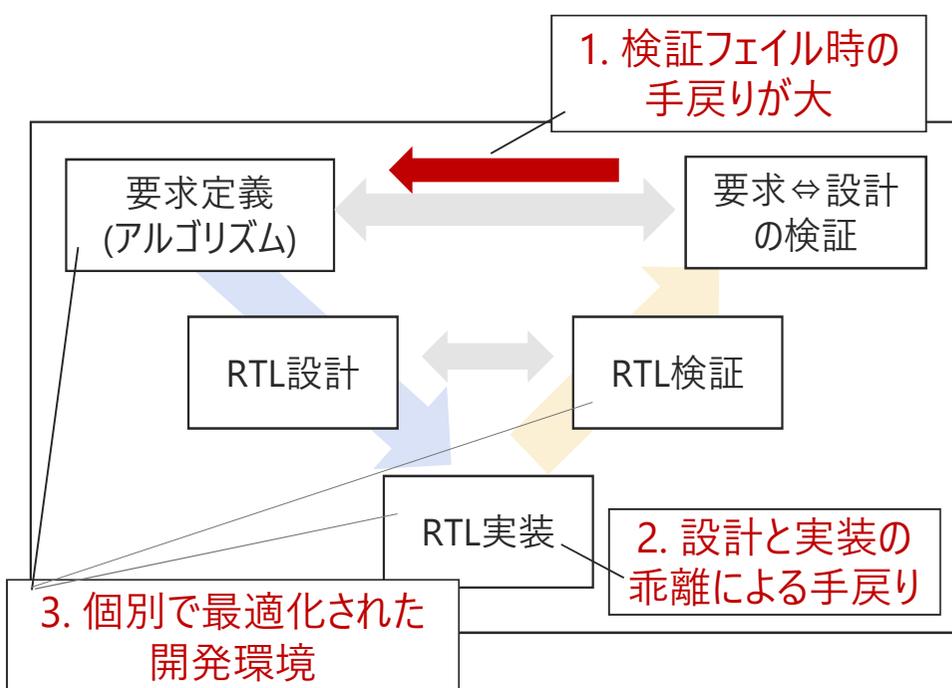
03 画像処理機能開発におけるMBDフロー

04 適用事例報告①：従来開発との比較

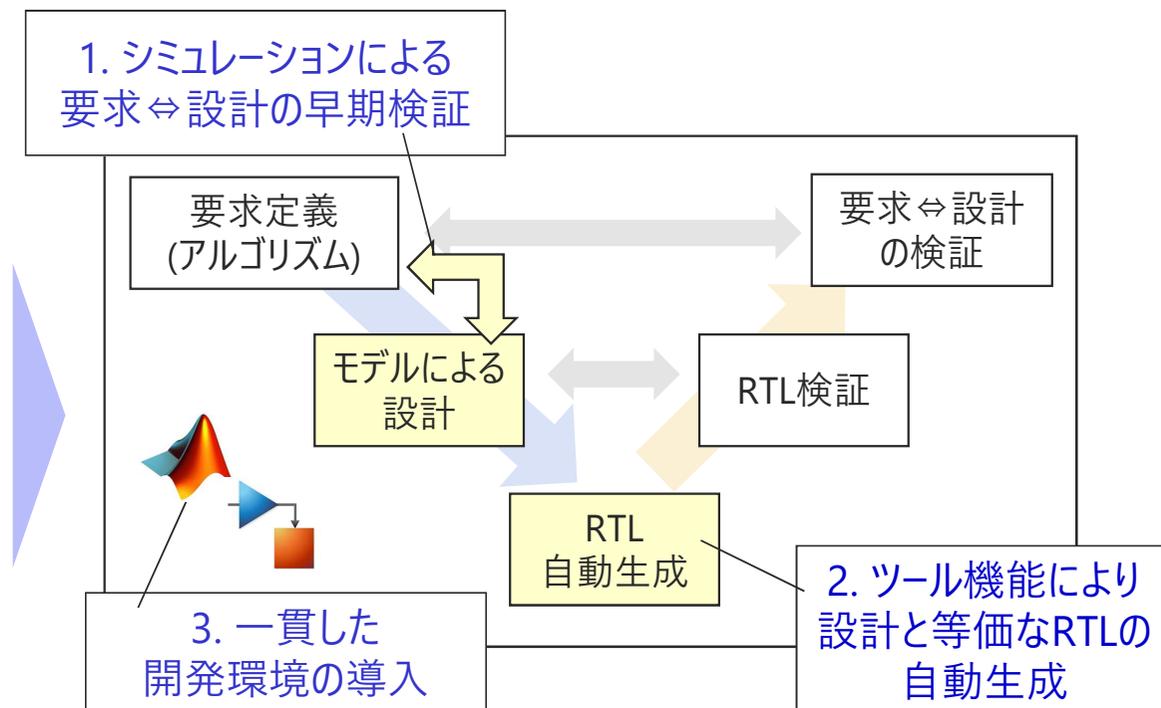
05 適用事例報告②：製品開発への適用

06 まとめ

画像処理開発(アルゴリズム⇒HW化)へのMBD適用の狙い



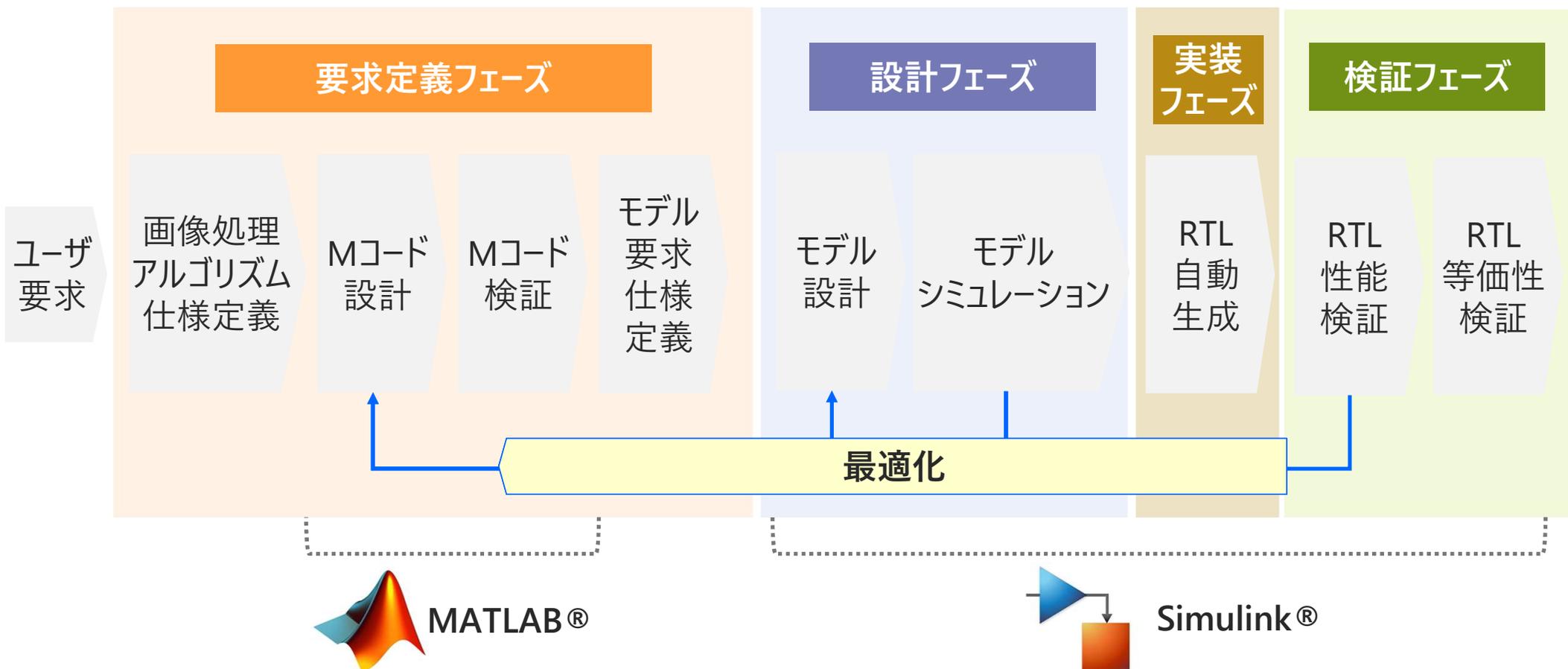
従来開発フロー



MBD開発フロー

MATLAB®/Simulink®を用いたMBD開発フローによる開発効率の向上

MATLAB®/Simulink®を用いたMBD開発フローの詳細



要求定義フェーズ

ユーザ要求



例：内視鏡画像上の
わずかなコントラストを強調したい

画像処理 アルゴリズム仕様定義

内視鏡画像の
色調を・・・

Mコード設計 (HWのリファレンsprogram)

```
2  
3 % filtering処理  
4 y_fil = imfilter(In_img(:,:,1), fil_Y,'replicate');  
5 cb_fil = imfilter(In_img(:,:,2), fil_C,'replicate');  
6 cr_fil = imfilter(In_img(:,:,3), fil_C,'replicate');  
7 % |  
8 % round処理  
9  
10  
11  
12
```

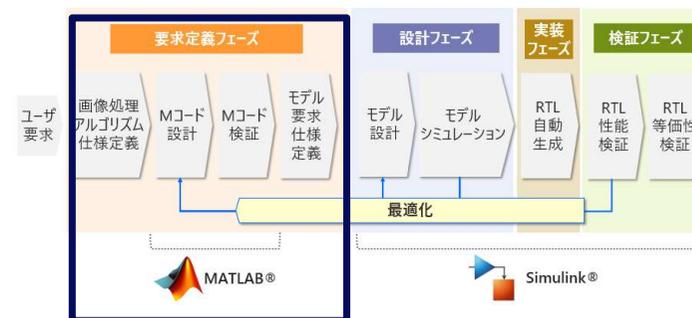
Mコード検証 (画質検証)

従来处理

新規処理

モデル要求仕様定義

1. 入出力情報
In/Out, Bit幅・・・
2. HW化時の要求
レイテンシ、
フィルタ演算・・・



画像処理アルゴリズムをHW化するための要求の定義

設計フェーズ

モデル要求仕様定義

1. 入出力情報
In/Out, Bit幅...
2. HW化時の要求
レイテンシ、
フィルタ演算...

Mコード設計 (HWのリファレンスプログラム)

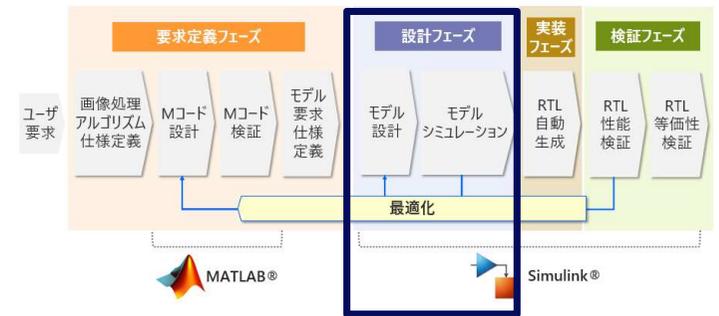
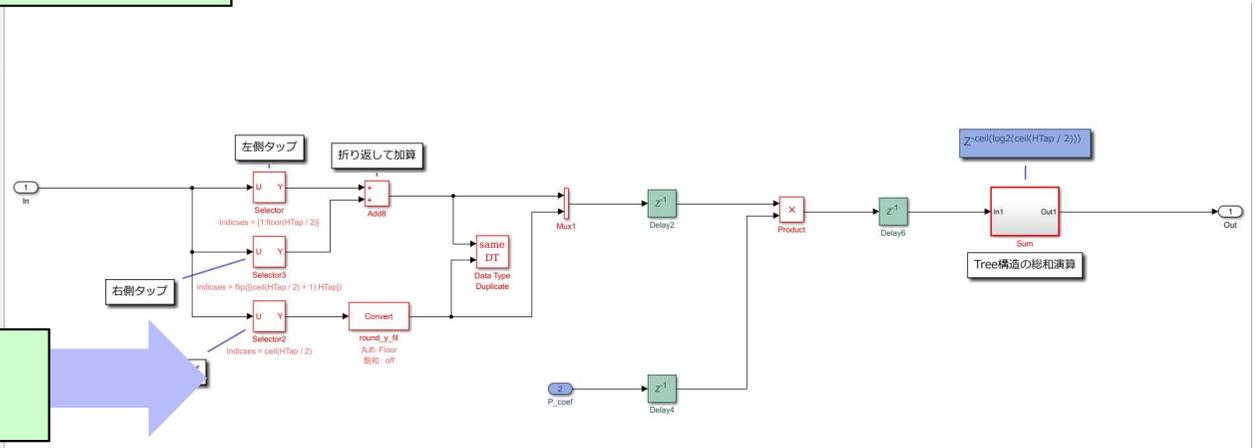
```

3 % filtering処理
4 y_fil = infilter(In_Jug(:,:,1), fil_Y,'replicate');
5 cb_fil = infilter(In_Jug(:,:,2), fil_G,'replicate');
6 cr_fil = infilter(In_Jug(:,:,3), fil_O,'replicate');
7 %
8 % round処理
9 y_fir = round(y_fil);
10 cb_fil = round(cb_fil);
11 cr_fil = round(cr_fil);

```

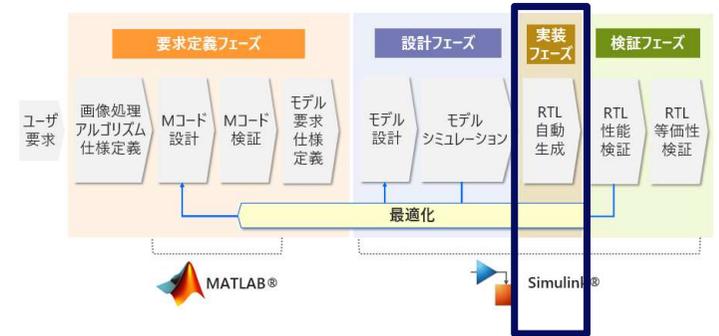
モデル設計

モデルシミュレーション



シミュレーションにより要求 ⇄ 設計(モデル)を検証

実装フェーズ



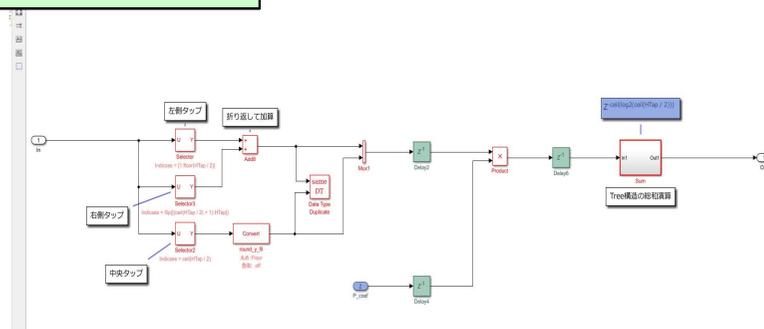
Mコード設計 (HWのリファレンスプログラム)

```

1 %% matrix演算
2
3 % filtering処理
4 y_fil = infilter(In_Img(:,1), fil_y,'replicate');
5 cb_fil = infilter(In_Img(:,2), fil_cb,'replicate');
6 cr_fil = infilter(In_Img(:,3), fil_cr,'replicate');
7
8 %
9 % round処理
10 y_fil = round(y_fil);
11 cb_fil = round(cb_fil);
12 cr_fil = round(cr_fil);
13

```

モデル設計



RTL自動生成

```

6 USE IEEE.numeric_std.ALL;
7
8 ENTITY alpha1ST_RGB2YCbCr IS
9     PORT ( CLK
10          : IN std_logic;
11          RESET_X
12          : IN std_logic;
13          i_R
14          : IN std_logic_vector(
15            0_CbCr_0
16            );
17          i_G
18          : IN std_logic_vector(
19            0_CbCr_1
20            );
21          i_B
22          : IN std_logic_vector(
23            0_CbCr_2
24            );
25          o_CbCr_0
26          : OUT std_logic_vector(
27            0_CbCr_0
28            );
29          o_CbCr_1
30          : OUT std_logic_vector(
31            0_CbCr_1
32            );
33          o_CbCr_2
34          : OUT std_logic_vector(
35            0_CbCr_2
36            );
37          );
38 END alpha1ST_RGB2YCbCr;
39
40 ARCHITECTURE rtl OF alpha1ST_RGB2YCbCr IS
41
42     -- Signals
43     SIGNAL Selector_out1
44     : vector_of_unsigned10(0 TO
45     1);
46     SIGNAL Delay4_out1
47     : vector_of_unsigned10(0 TO
48     1);
49     SIGNAL Constant_out1
50     : vector_of_signed16(0 TO 5);
51     SIGNAL Demux1_out1
52     : vector_of_signed16(0 TO 1);
53     SIGNAL Delay3_out1
54     : vector_of_signed16(0 TO 1);
55     SIGNAL Product1_cast
56     : vector_of_signed11(0 TO 1);
57     SIGNAL Product1_mul_temp
58     : vector_of_signed27(0 TO 1);
59     SIGNAL Product1_out1
60     : vector_of_signed26(0 TO 1);
61     SIGNAL Delay5_out1
62     : vector_of_signed26(0 TO 1);
63     SIGNAL Selector1_out1
64     : vector_of_unsigned10(0 TO
65     1);
66     SIGNAL Delay9_out1
67     : vector_of_unsigned10(0 TO
68     1);
69     SIGNAL Demux1_out2
70     : vector_of_signed16(0 TO 1);
71     SIGNAL Delay6_out1
72     : vector_of_signed16(0 TO 1);
73     SIGNAL Product3_cast
74     : vector_of_signed11(0 TO 1);
75     SIGNAL Product3_mul_temp
76     : vector_of_signed27(0 TO 1);
77     SIGNAL Product3_out1
78     : vector_of_signed26(0 TO 1);
79     SIGNAL Delay13_out1
80     : vector_of_signed26(0 TO 1);
81

```

MALTA[®]/Simulink[®] toolbox (HDL Coder)により、モデルからRTLコードの自動生成

01 オリンパスのご紹介

02 MBD適用のモチベーション

03 画像処理機能開発におけるMBDフロー

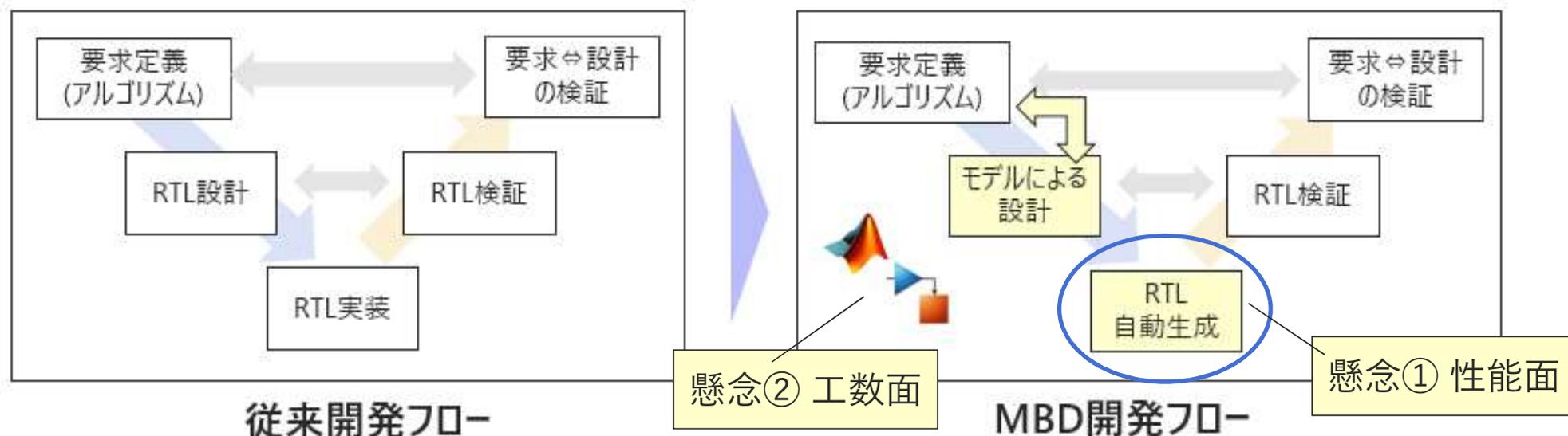
04 適用事例報告①：従来開発との比較

05 適用事例報告②：製品開発への適用

06 まとめ

適用事例報告①：従来開発との比較

MBD開発フローの製品開発適用時に想定される懸念について検討



- ✓ 懸念① 性能面：従来と同程度の性能の自動生成コードを生成可能か？
- ✓ 懸念② 工数面：開発環境、開発フローが大きく変わるが、かえって工数が増加しないか？

検討方法

複数の既存画像処理アルゴリズムについて、
スキルの異なるエンジニアでMBD開発を実施

✓ 対象画像処理アルゴリズム：

計 13 個のアルゴリズム (ローパスフィルター, ガンマ補正, 拡大縮小, シャープネス強調 など)

✓ 実施者：

計 5 名 (ツールスキル, RTL設計, 画像処理アルゴリズム設計のスキルの異なるエンジニア)

✓ 評価項目：

1. 性能(RTLの回路規模, 速度, レイテンシ)

2. 開発工数※

※ 従来開発時の実績算出が困難であったため
MBD開発フローの実績値の絶対評価とした

実施者	MATLAB [®] 経験	Simulink [®] 経験	RTL 設計経験	画像処理 アルゴリズム 設計経験
A	5年以上	1～3年	3～5年	1～3年
B	1年未満	1年未満	経験なし	経験なし
C	経験なし	経験なし	3～5年	経験なし
D	5年以上	5年以上	5年以上	5年以上
E	5年以上	5年以上	5年以上	経験なし

結果：1. 性能面

計13個の画像処理アルゴリズム トータル 回路規模, レイテンシ比較結果

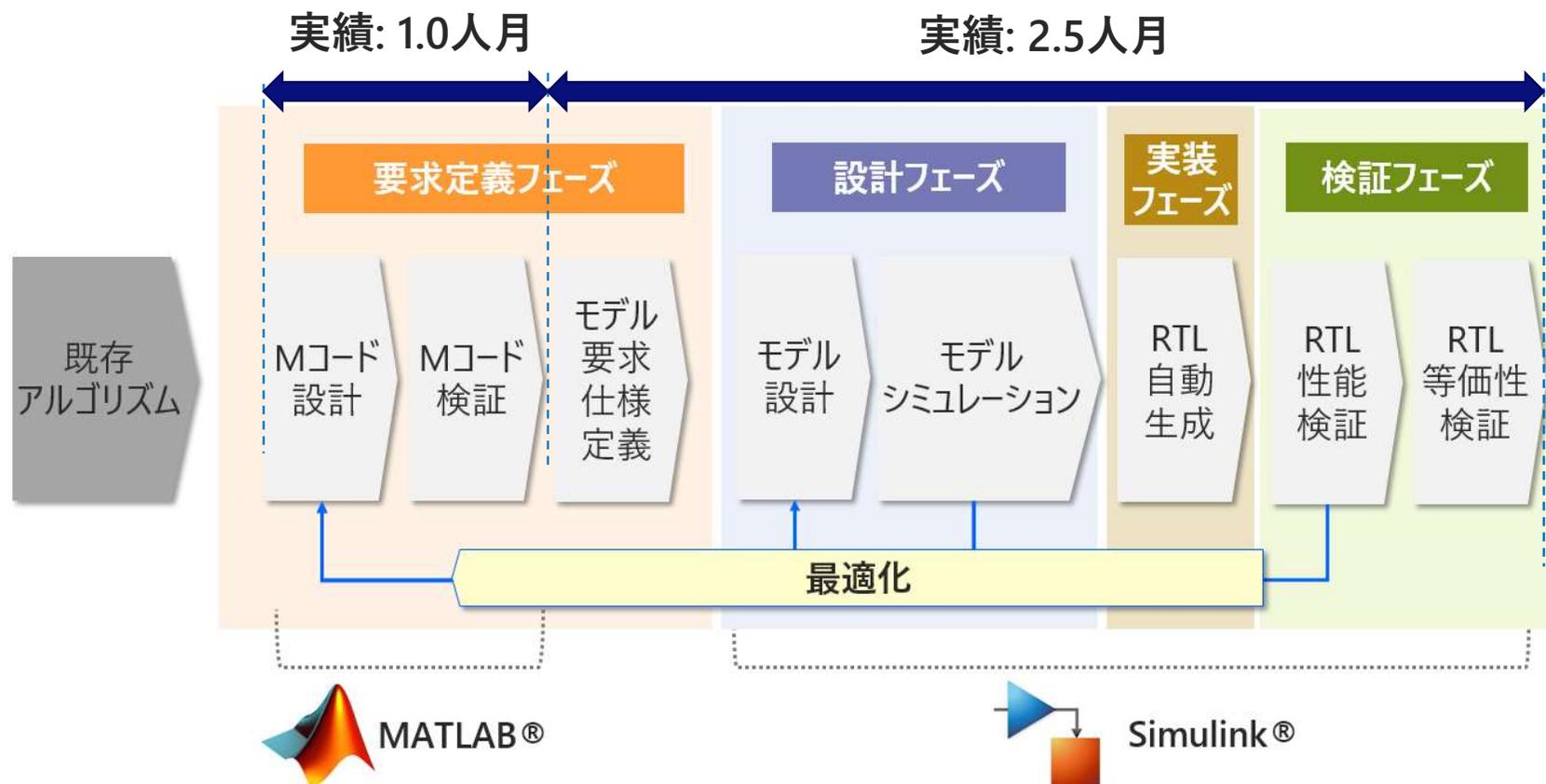
	Slice Reg	LUTs	BRAM		DSP 48	レイテンシ
			RAM 18	RAM 36		
従来開発	20207	14829	11	50	234	7line + 99 cycle
MBD開発	19087	16869	44	37	272	7line + 132 cycle

検証に使用したFPGAデバイス: Xilinx Kintex-7(XC7K325, -1)

- 1) 回路規模 : 従来開発 ≒ MBD開発
- 2) 回路速度 : Fmax > 目標値(74.25MHz)を達成
- 3) レイテンシ : 従来開発 ≒ MBD開発(同Line遅延)

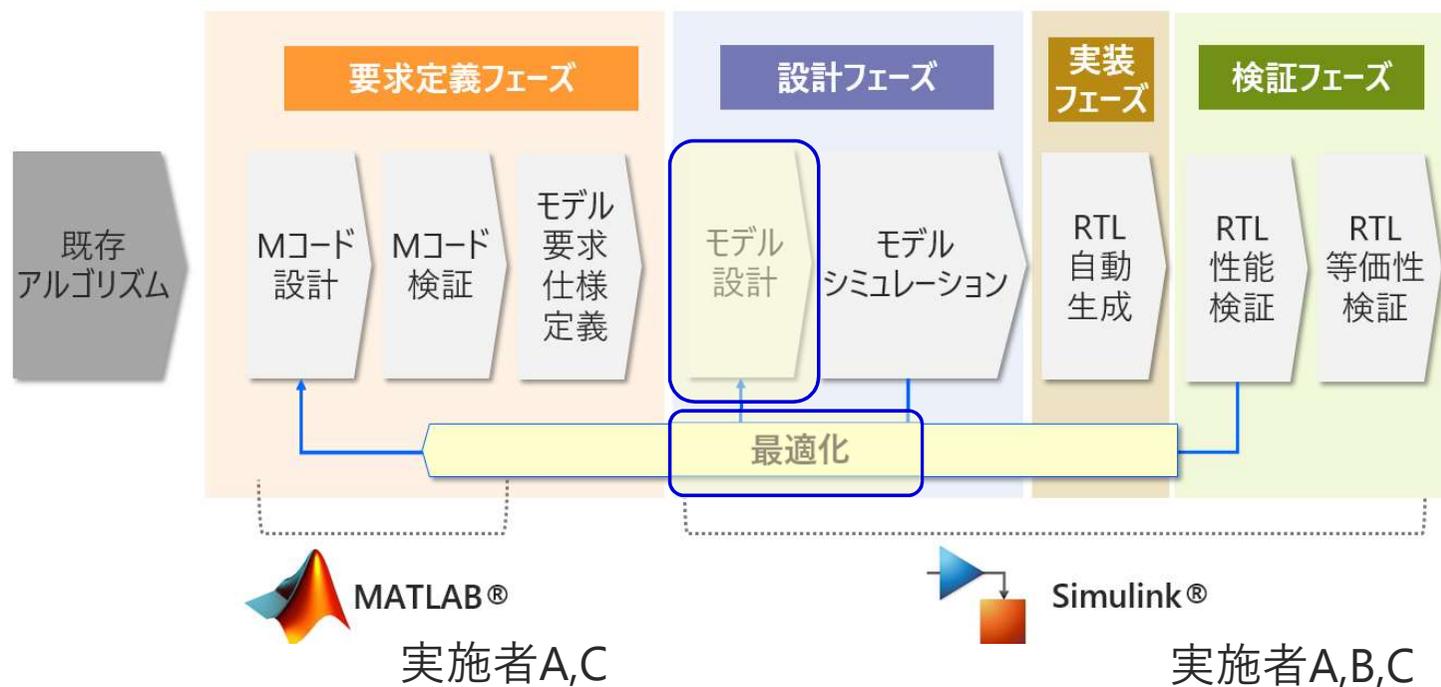
従来開発と同程度の性能を実現可能

結果：2. 開発工数



リーズナブルな実績工数で実現可能

苦労した点：設計者からの声



実施者B,C



Mコードからモデルにどうやって置き換えたらいいの？

実施者A,C



RTLで記述のイメージわくがモデルでの表現がわからない

実施者A,B,C



自動生成RTLの回路規模が大きい、最適にするには？

特にモデル設計と最適化に苦労

施策：モデル設計のためのガイドライン策定

ガイドライン

・画像処理IPのModeling基本ルール、及びサンプルモデルをまとめる
 ⇒ modelling時に最低限抑えておきたいポイントを絞って、記載する
 ⇒ さらに詳細なモデリングガイドラインに関しては、Mathworks社の

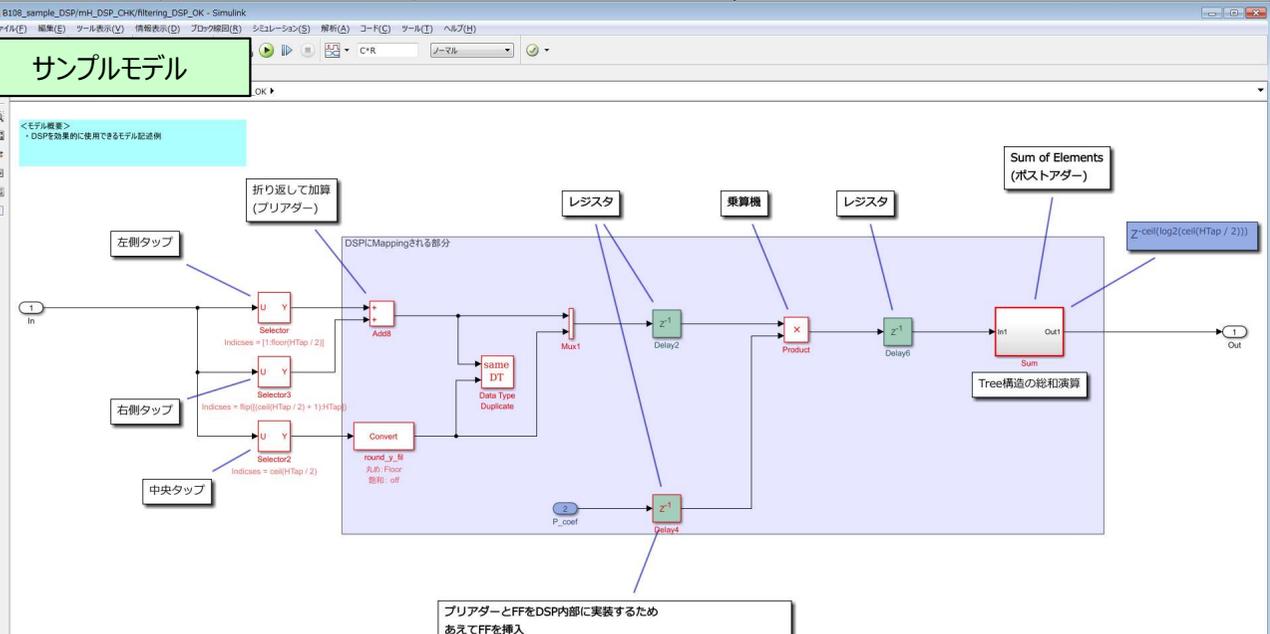
モデル化ルール

% 本ガイドラインは matlab ver : R2016b を対象とする

No	項目	対応レベル 必須 > 推奨 > 参考	概要	
A	001	ツール設定	必須	hdlsetup
A	002	ツール設定	必須	ソルバー設定
A	003	ツール設定	参考	データのエキスポート
A	004	ツール設定	必須	HDL Coder設定
A	005	ツール設定	参考	ワークアラウンドバイザーによるTgmモデル生成
B	001	モデル化	必須	モデル記載ルール
B	002	モデル化	必須	ルックアップテーブル
B	003	モデル化	必須	ゲインブロックの記述
B	004	モデル化	必須	効率的なDSPへの演算
B	005	モデル化	参考	Simulink上で画像処理Simを実施する

No	項目	対応レベル 必須 > 推奨 > 参考	理由	サンプルモデル
1	モデル 背景色	推奨	入力ポートについて号(パラメータなど)	
2	モデル 背景色	必須	Delayブロックの色	
3	モデル 説明文 (Simulinkモデル上でダブルクリックでテキスト入力)	必須	サブシステムなど、モデル化ルールを記載する	

サンプルモデル



画像処理特有の処理に特化したモデル化ガイドライン作成とサンプルモデルの提供

01 オリンパスのご紹介

02 MBD適用のモチベーション

03 画像処理機能開発におけるMBDフロー

04 適用事例報告①：従来開発との比較

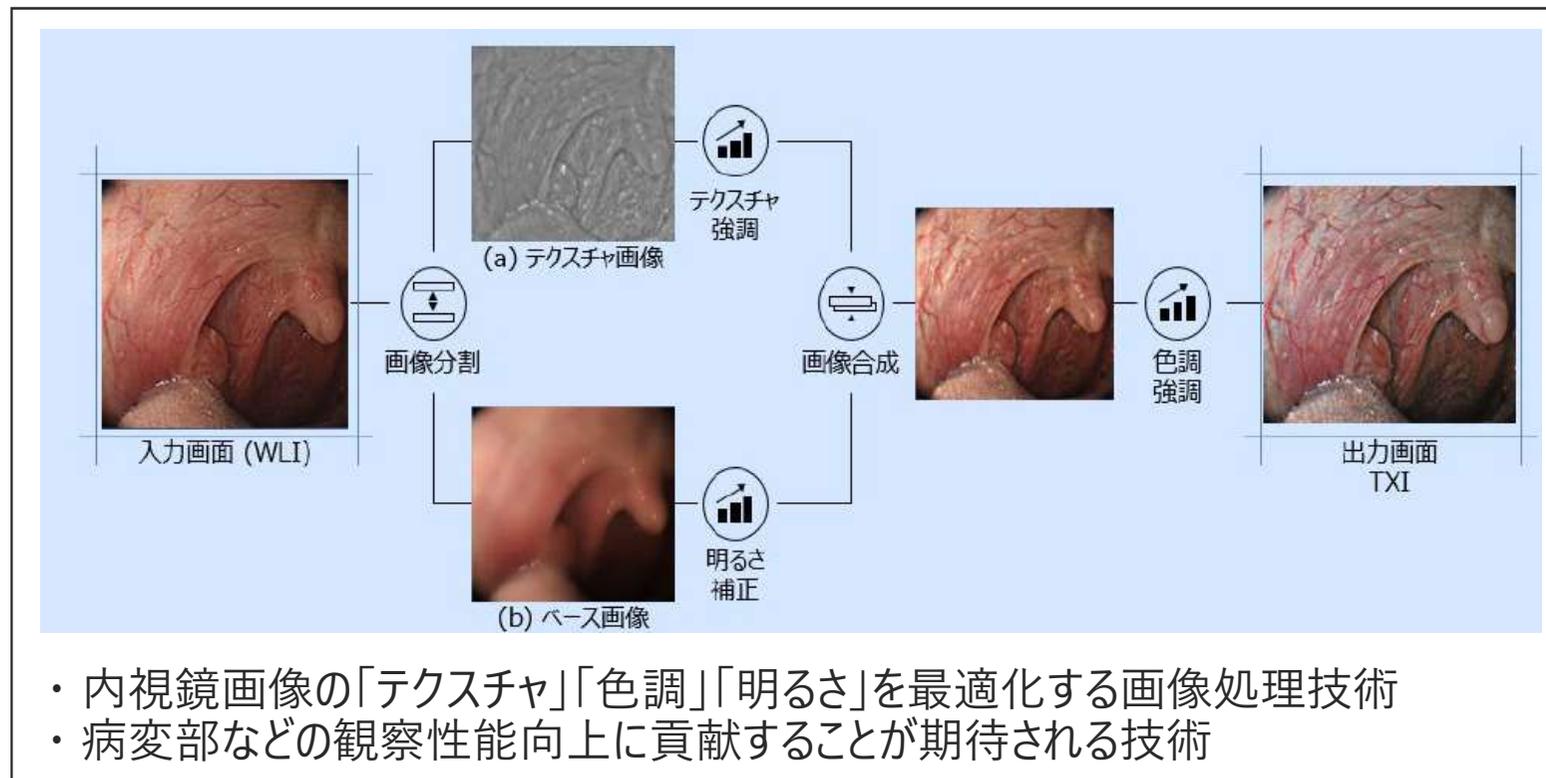
05 適用事例報告②：製品開発への適用

06 まとめ

適用事例報告②：製品開発への適用



次世代内視鏡ビデオプロセッサ



- ・ 内視鏡画像の「テクスチャ」「色調」「明るさ」を最適化する画像処理技術
- ・ 病変部などの観察性能向上に貢献することが期待される技術

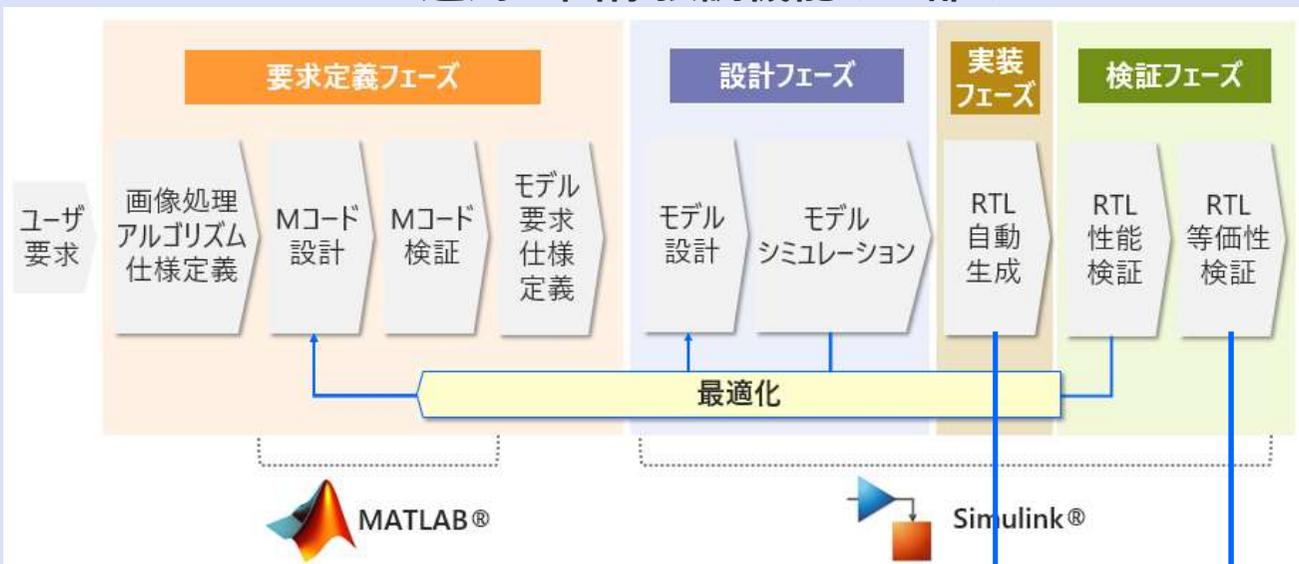
新画像強調技術(TXI: Texture and Color Enhancement Imaging)

次世代内視鏡システム搭載の新画像強調技術の開発にMBDを適用

製品開発 MBD適用 ワークフロー：実施の流れ

MBD適用：画像強調機能の一部のIP

画像処理
アルゴリズム
開発者



- ✓ アルゴリズム開発者が対象IPの一連の開発を担当
- ✓ 自動生成RTLはHW開発者が全体デザインへ統合

HW
開発者

従来開発適用

デザイン統合フェーズ

全体デザインへの統合

全体デザインでの検証

デザイン
リリース

製品開発へのMBD適用のポイント 1/2

✓ HW開発者との協調開発

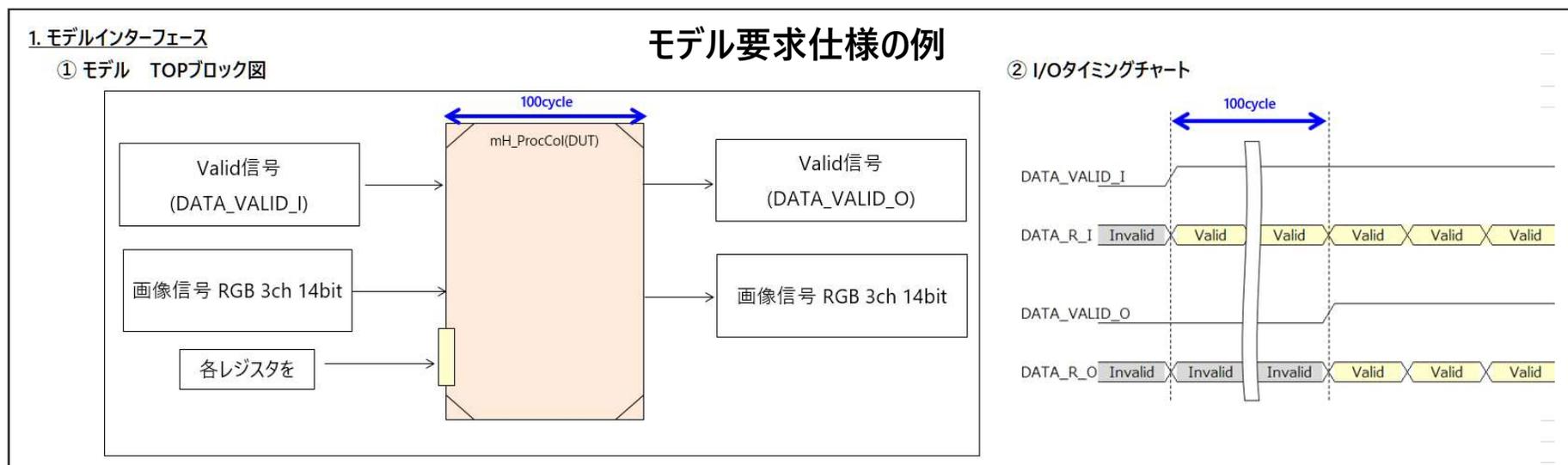
- ・ その他IPとのインターフェースの統一化が必要 ⇒ モデル要求仕様として定義



画像処理アルゴリズム開発者



HW開発者



製品開発へのMBD適用のポイント 2/2

✓ ソフトウェアバリデーション

- ・ 自社規定に則り、MALTAB®/Simulink®のバリデーションの実施

試験用モデルと自動生成コードの確認

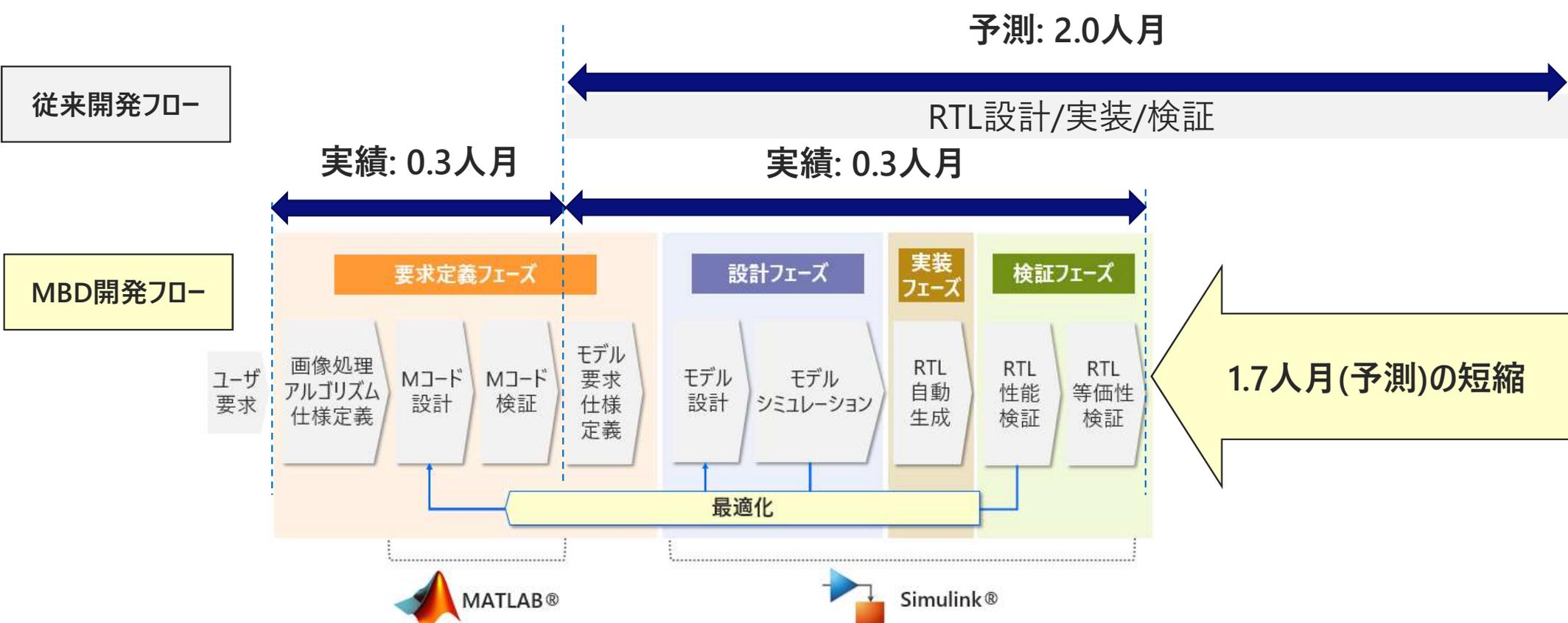
バリデーション用試験手順の定義

試験仕様			
機能名及び機能概要	試験手順	試験データ	合格基準(想定動作)
VHDL自動生成: 単一のサブシステムを使用したモデルのVHDLコード生成が出来る	① モデル CSV_MDL001.slx を開く ② 別タブの手順に沿って、VHDLを自動生成する。 ③ 生成されたVHDLの記述を確認する。	CSV_MDL001.slx [単一サブシステム モデル]	1. 単一のVHDLファイルが生成されること。 ① モデルデータから、.vhd ファイルが生成されること ② サブシステム名称と同一名称のvhdファイルが生成されること 2. モデルと同等のVHDLが生成されること。 ③ VHDLのI/O名とサブシステムのI/O名が同一であること。 ※ ただし、CLK、RESETは自動生成時に付与されるため、モデルには存在しない。CLK名称、RESET名称は、それぞれ設定名称 "CLK", RESET_X"であること。 ④ リセットが同期リセットになっていること。 ⑤ サブシステム内部のブロックと同等の演算、及びFFが漏れなくVHDLコードとして記述されていること。



The image displays a Simulink model and its corresponding VHDL code. The Simulink model on the left shows a control loop with several blocks: a gain block (1), a summing junction (2), a delay block (3), a gain block (4), a summing junction (5), and a delay block (6). The VHDL code on the right is annotated with numbers 1 through 6, showing the mapping between the blocks and the code logic. The code includes declarations for signals and processes for the control loop.

MBD開発実績（設計/検証フェーズでの工数比較）



MBD開発による設計/検証効率向上の可能性を見出した

01 オリンパスのご紹介

02 MBD適用のモチベーション

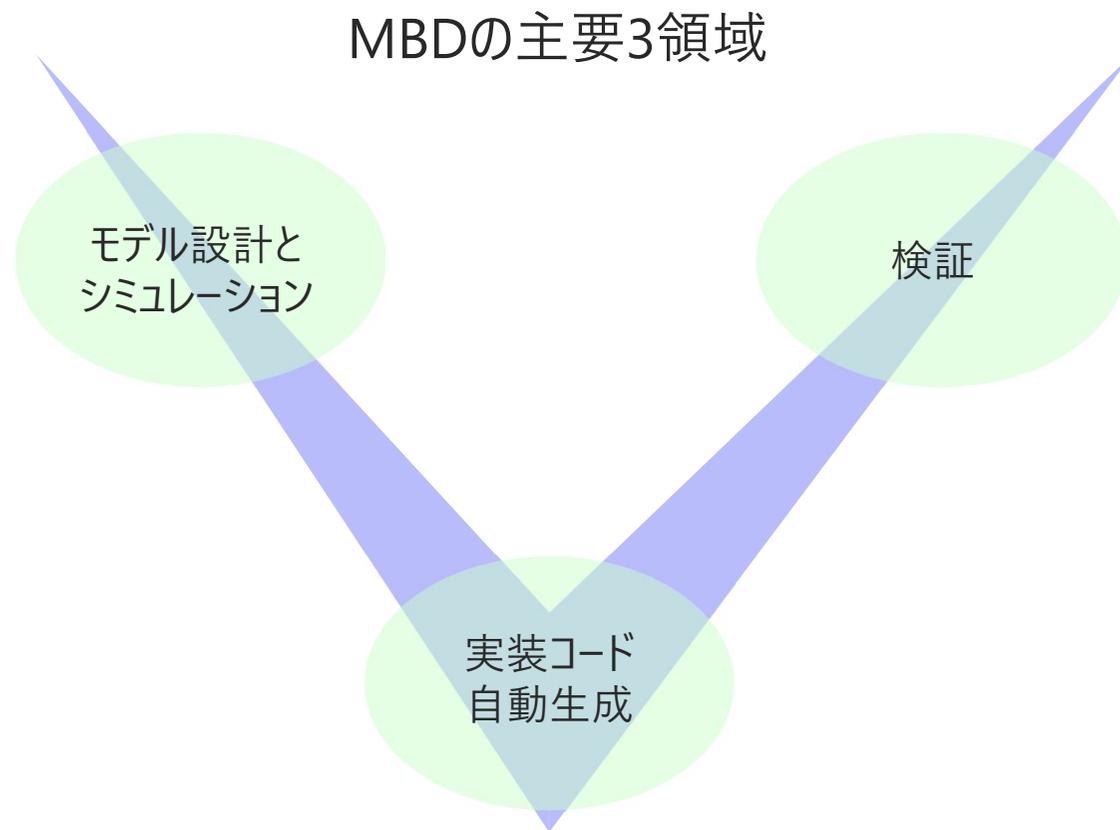
03 画像処理機能開発におけるMBDフロー

04 適用事例報告①：従来開発との比較

05 適用事例報告②：製品開発への適用

06 まとめ

まとめ：本取り組みの成果



- ✓ 自社の事例として、画像処理分野へのMBD適用で効率向上の可能性を見出せたこと
- ✓ さらに、内視鏡製品開発への適用実績ができたこと

OLYMPUS

A thick yellow horizontal line that tapers at both ends, positioned directly below the word "OLYMPUS".