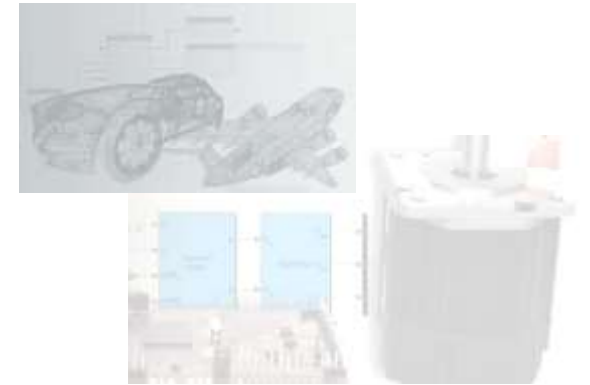


# モーター制御システム開発の効率化はモデルベースで解決

MathWorks Japan  
アプリケーションエンジニアリング部



# はじめに

## 本セミナーの対象者

- システムエンジニアリング設計者
- モーター制御開発者

## 本セッションでお伝えしたいこと

MATLAB/Simulinkを活用したシステムエンジニアリング  
とモーター制御開発の進め方



# 変化し続けるマーケットとそのニーズに応える製品の研究・開発

## ➤ 目まぐるしく変化する製品市場

- 消費者の趣向の変化
- マーケット規模の変遷に合わせた製品開発  
(新興国の隆盛・人口動態によるマジョリティの変化)
- 競合他社の新機能への対応
- 法規制の変化による新しい市場の出現



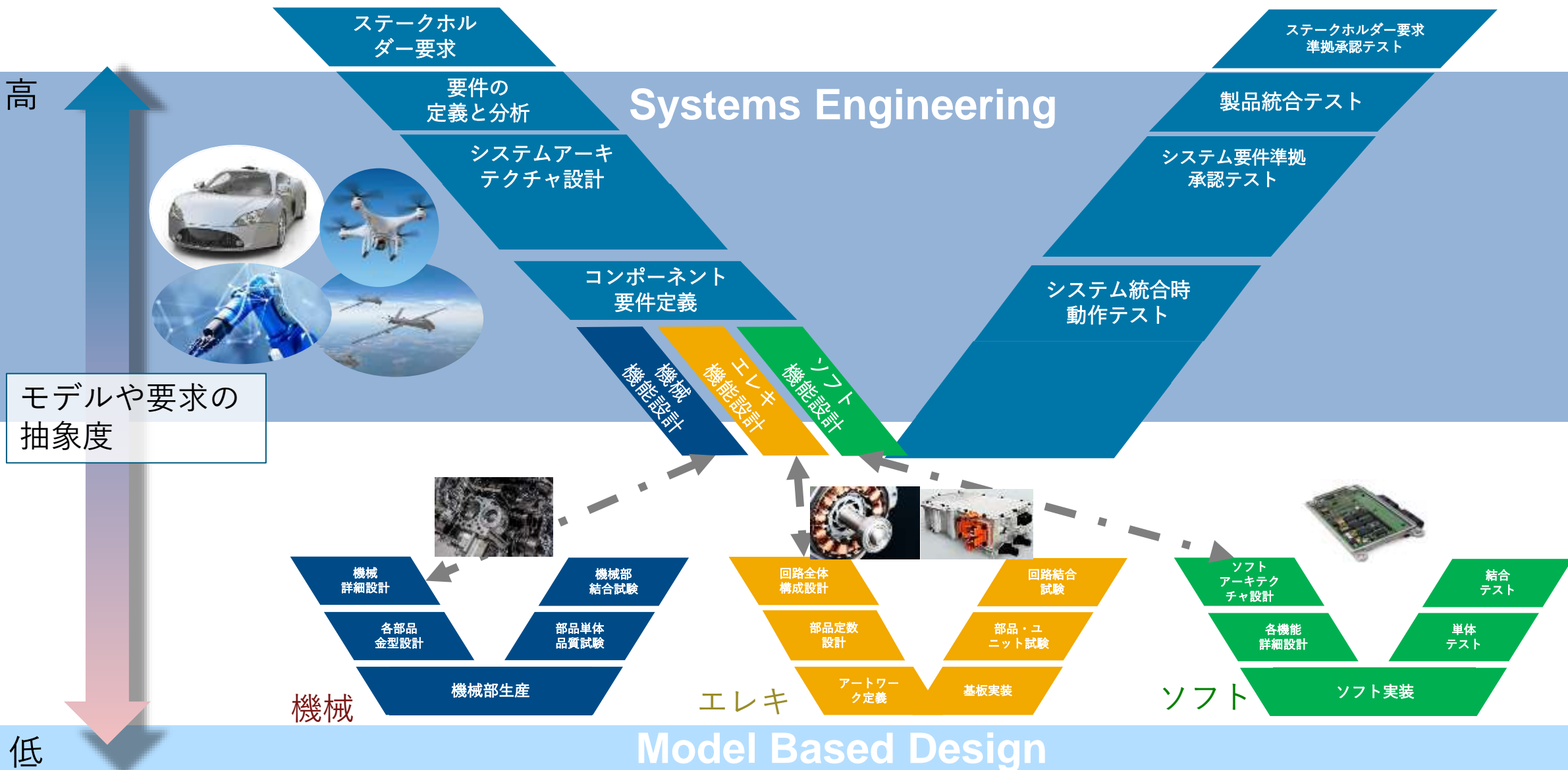
製品開発全工程を見据えた業務効率向上がキーポイント

## ➤ 高・多機能化そして高信頼性が求められる製品の研究・開発

- 電化製品: インテリジェント化・IoT
- 自動車: HEV/EV・先進走行支援システム・自動走行
- 製造・サービス: ロボットの高性能化による人的作業の自動化  
(生産ライン / 過酷・危険な業務)
- 電気・電力: 分散電源のマネジメント

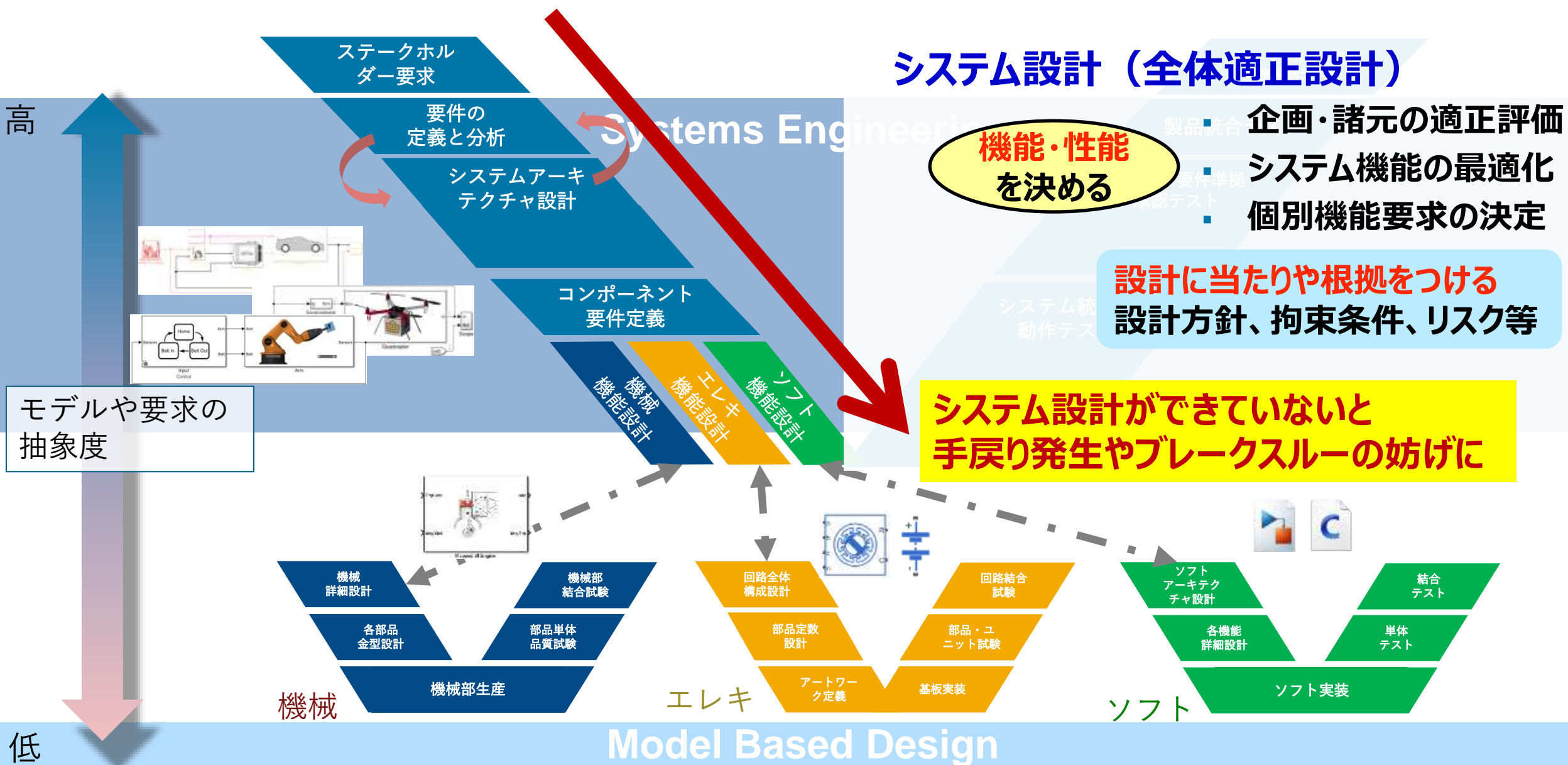


# Systems EngineeringとModel Based Designの関係



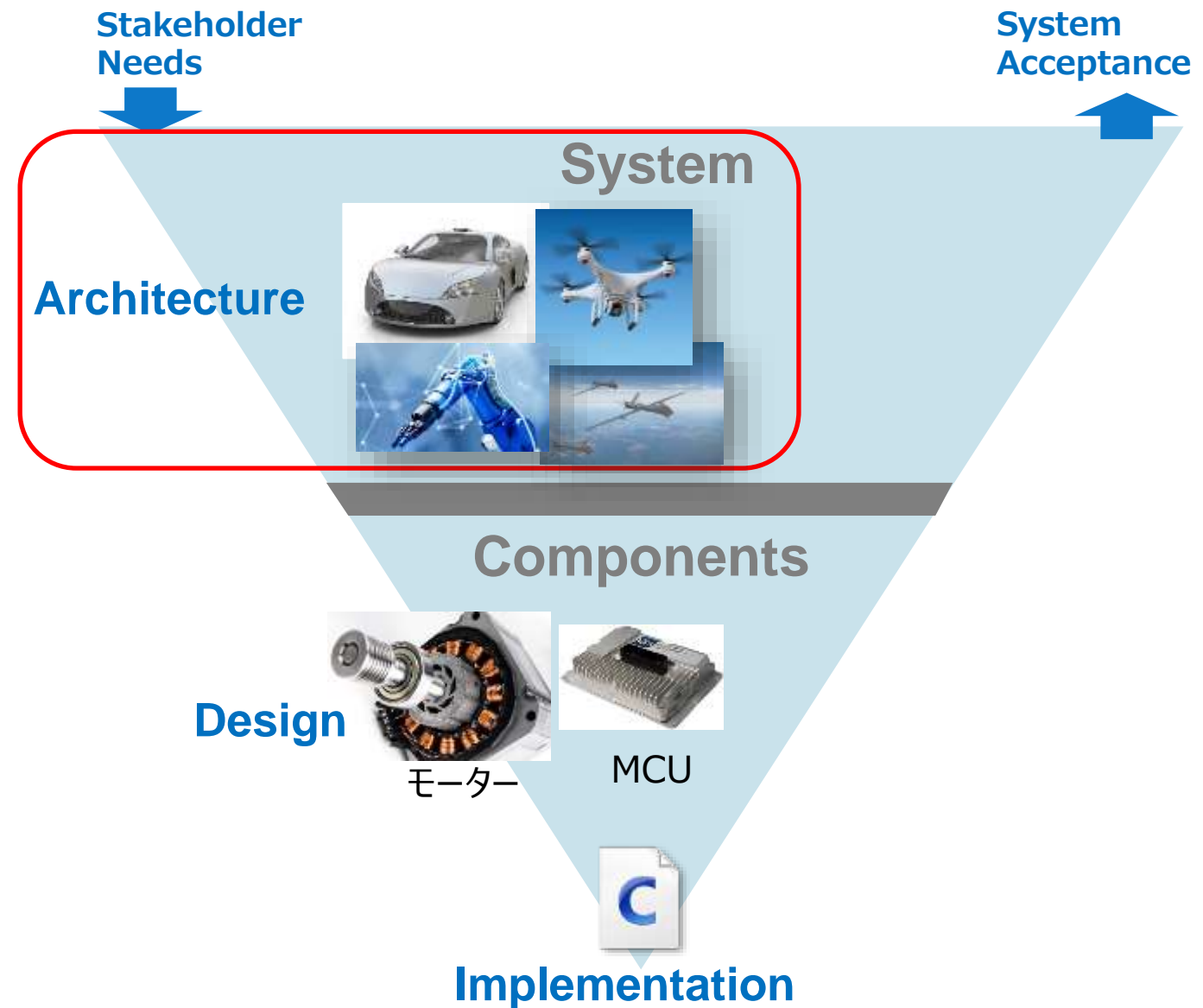


# システム設計でシミュレーションが活用できていない影響



# アジェンダ

- システムのアーキテクチャ設計について
- モーター制御を例にコンポーネント設計の手順
- コード生成 & 実装



# システムは複数のシステムやコンポーネントを組合せ

## System Of System (SoS): 複数システムを組み合わせで構成されるシステム



モーター



バッテリー



インバータ



ブレーキ



ギア

etc



モーター



バッテリー



インバータ



ギア



プロペラ



カメラ

etc



モーター



バッテリー



インバータ



ギア



グリッパ



カメラ

etc



モーター



バッテリー



インバータ



ギア

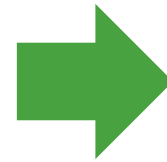


プロペラ



タイヤ

# 製品の企画・構想におけるコンセプト検討



全て試作するのは非現実的！

シミュレーションを活用して検討

EV? HV?

燃費が良い構成は？

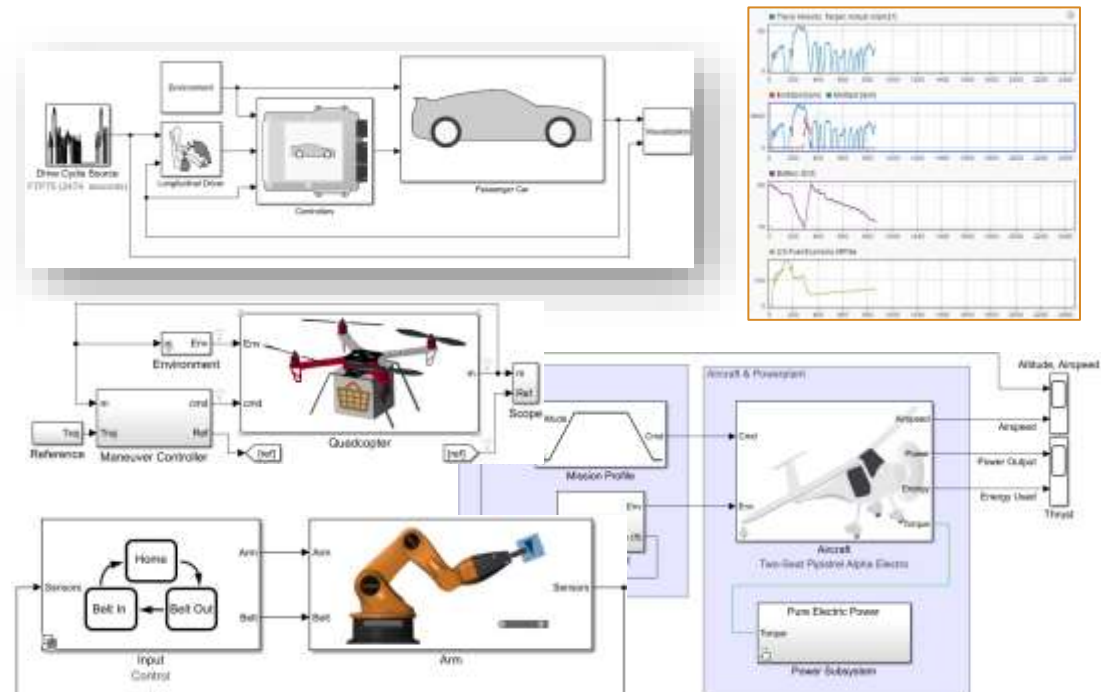
飛行時間は？

加速性の良いのは？

最適なギア比は？

モータの数/配置は？

バッテリー容量は？



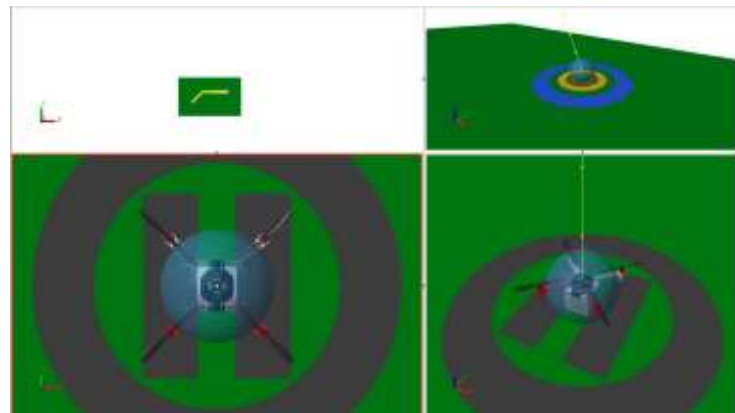




# シミュレーションを活用したコンセプト検討



[Powertrain Blockset](#) / [Vehicle Dynamics Blockset](#)  
[Simscape製品群](#)



[Simscape製品群](#)  
[UAV Toolbox](#)



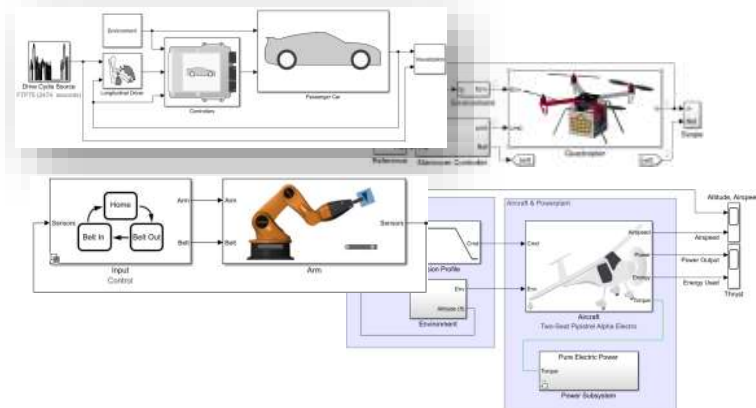
[Robotics System Toolbox](#)  
[Simscape製品群](#)



[Simscape製品群](#)  
[Aerospace Toolbox](#) / [Aerospace Blockset](#)

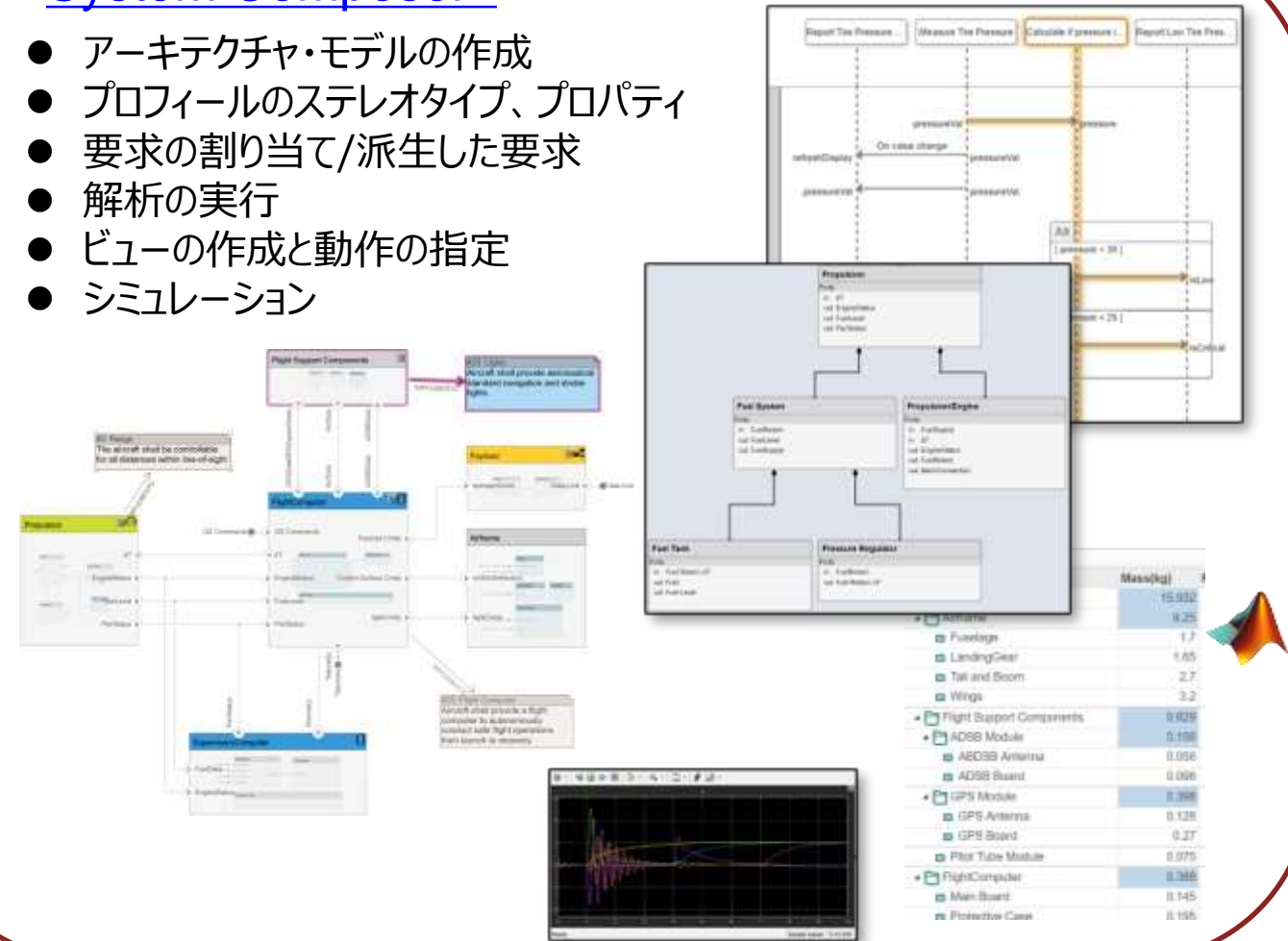
# システム構成の検討 & 定義

## システムアーキテクチャを検討



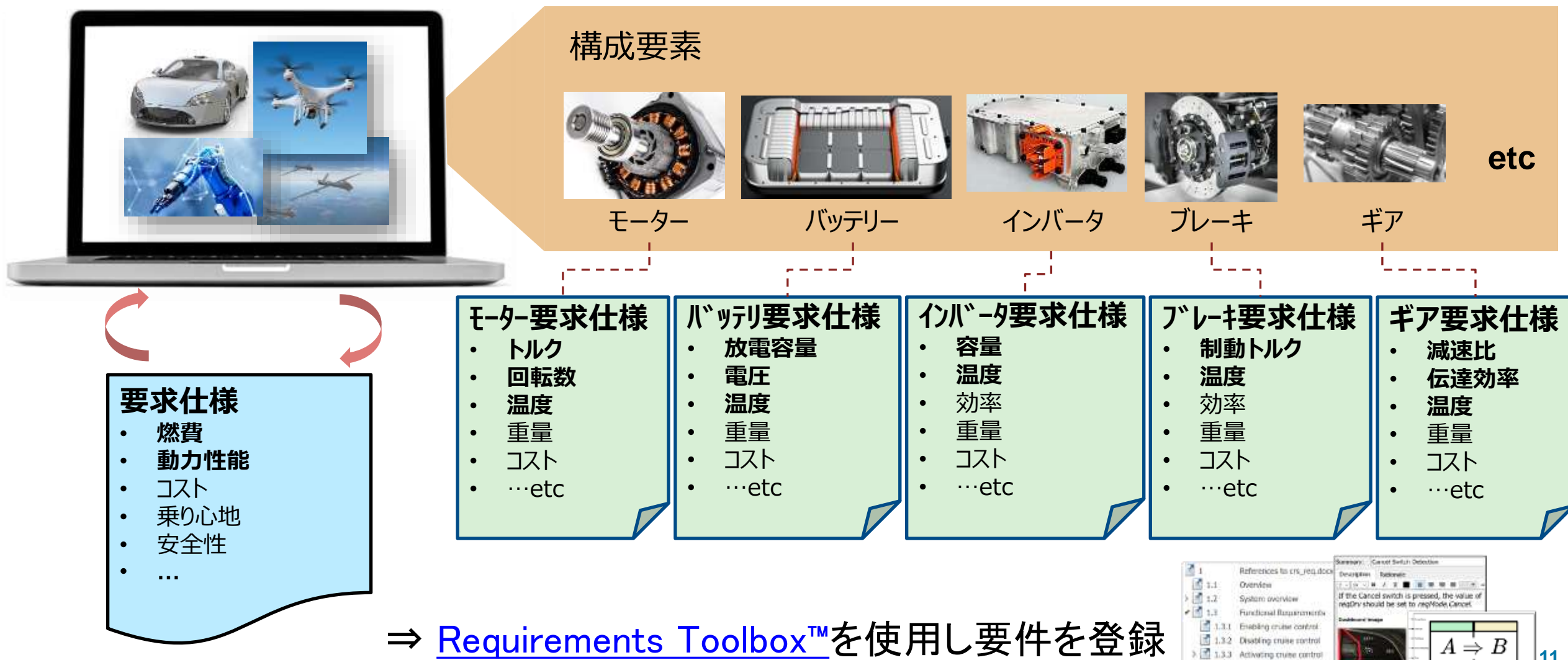
### System Composer™

- アーキテクチャ・モデルの作成
- プロフィールのステレオタイプ、プロパティ
- 要求の割り当て/派生した要求
- 解析の実行
- ビューの作成と動作の指定
- シミュレーション



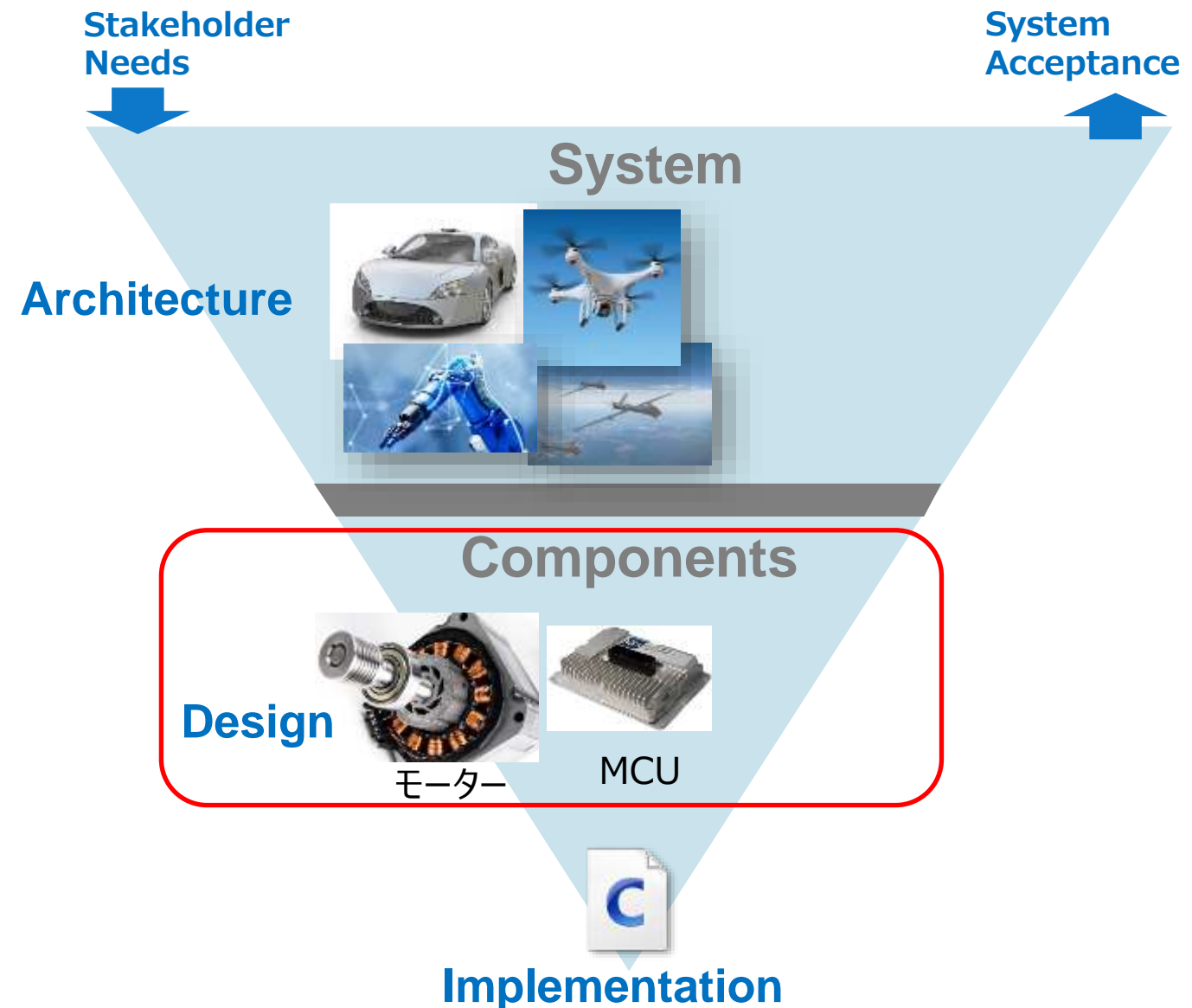
# 要求仕様の定義

システム要求を満たす形で、モジュール毎の要求を定義



# アジェンダ

- システムのアーキテクチャ設計について
- モーター制御を例にコンポーネント設計の手順
- コード生成 & 実装



# モーター制御システム設計 & 性能検討のワークフロー

1

上位要求の  
取得・整理



Requirement Toolbox™

## モーター要求仕様

- トルク
- 回転数
- 温度
- 重量
- コスト
- …etc

2

モーター  
モデル作成

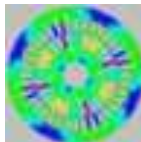
検討したいモーターを選択し  
モーターの諸元を設定する

Simscape Electrical™

PMSM



巻線界磁モータ



FEM

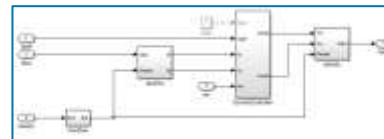
【新規モーター設計時】  
電磁解析ソフトの諸元  
値を活用

3

制御設計  
(アルゴ検討)

制御アルゴを構築し  
性能確認

Motor Control Blockset™



応答  
オーバーシュート  
外乱への感度 等

4

上位要求の  
更新・修正

要求の妥当性確認/  
モーター選定・アドバイス

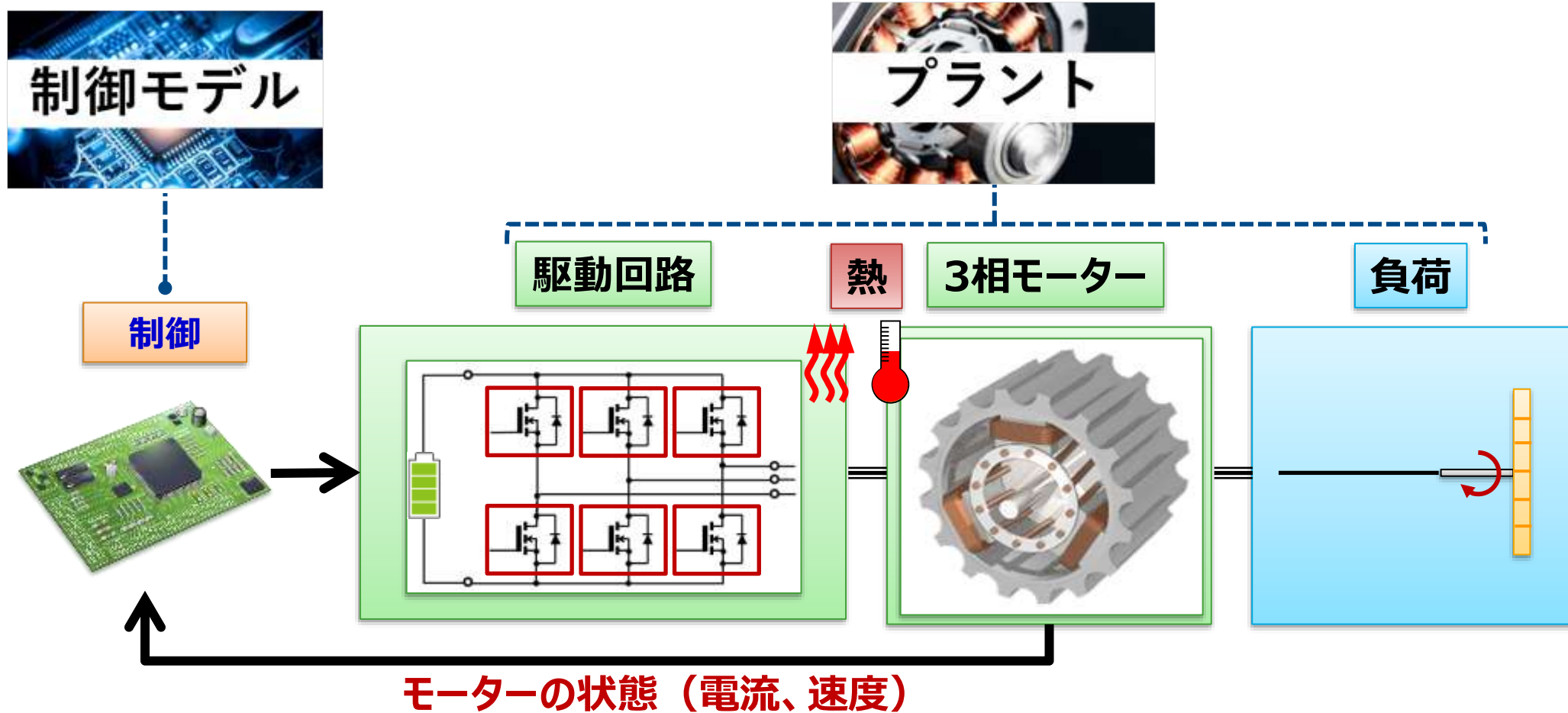






# モーター制御開発

～ 例：PMSMの速度制御～

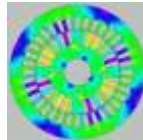




# 永久磁石同期モーター (PMSM) の詳細モデル

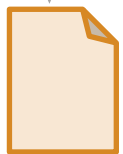
モーターの振動・発熱に起因するトルクリップル・電力損失を考慮した詳細な制御設計が可能

## FEM (有限要素法) モーターの電磁界解析

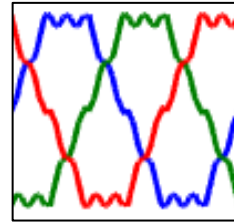


- JMAG (JSOL)
- ANSYS (Maxwell)
- Femtet (Murata Software)
- MotorCAD (Motor Design Ltd)
- MotorSolve (Infolytica)

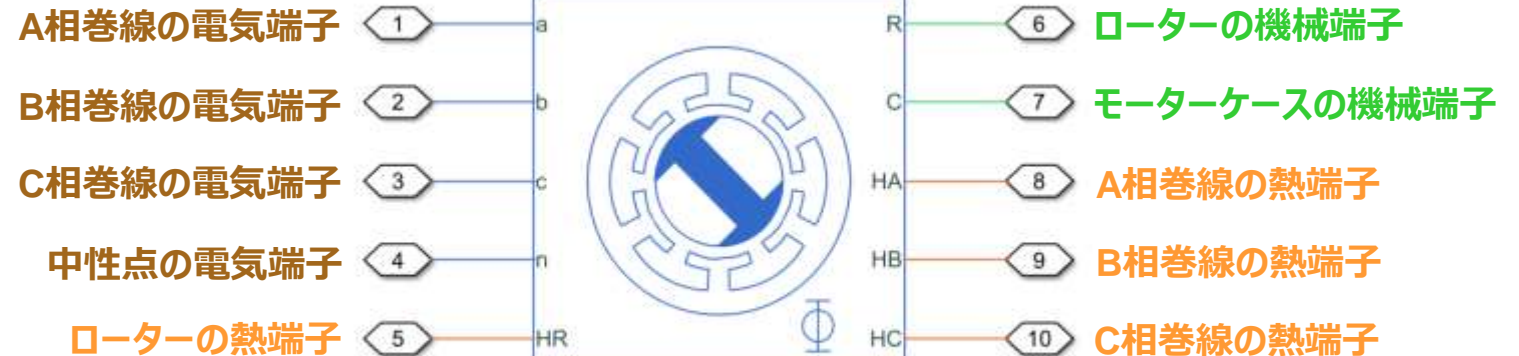
エクスポート



データ  
(FEMフォーマット)



- 電流やローター角度に応じた非線形性  
(磁気飽和、トルクリップル)
- ローター、ステーター巻線の温度変化



FEM-Parameterized  
PMSM

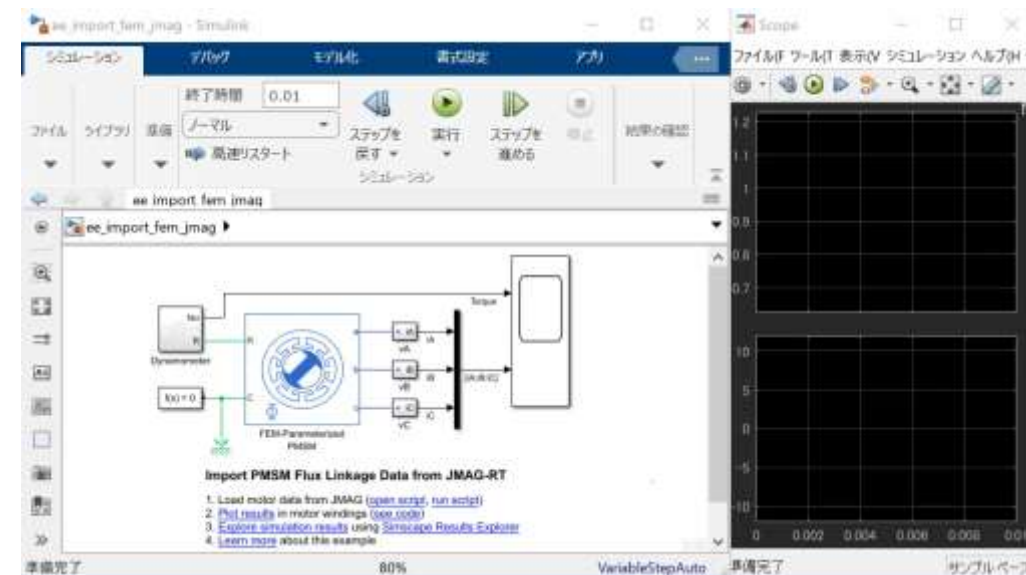
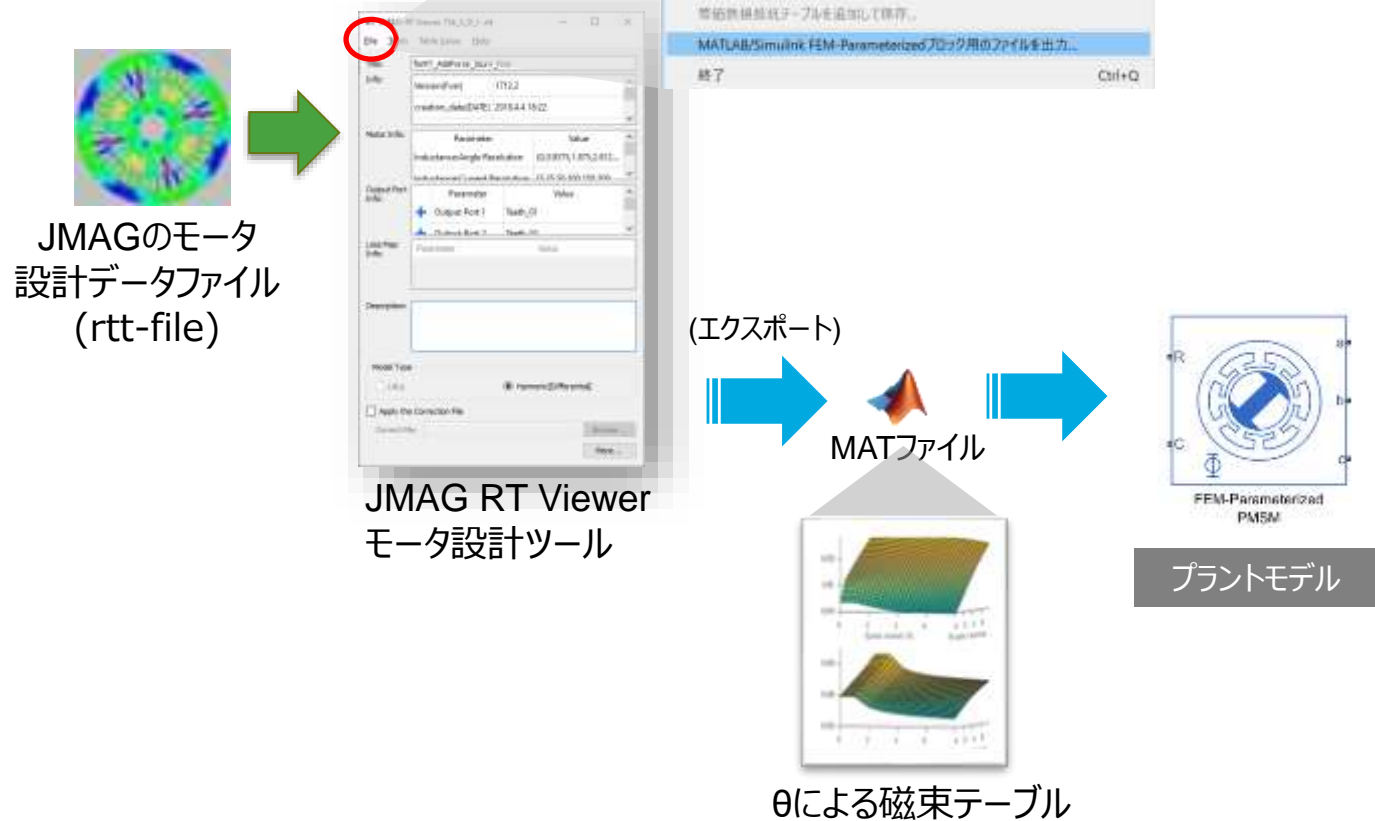
設定	説明
モデリングオプション	3次元鎖交磁束データ   熱端子を表示
電気	2次元偏微分データ   熱端子なし
鎖交磁束データの形式	2次元偏微分データ   熱端子を表示
巻線タイプ	3次元偏微分データ   熱端子なし
ニュートラル端子の表示	3次元偏微分データ   熱端子を表示
極対数	4次元偏微分データ   熱端子なし
Parkの表示形式データの規則	4次元偏微分データ   熱端子を表示
	3次元鎖交磁束データ   熱端子なし
	3次元鎖交磁束データ   熱端子を表示

$$\phi_d(i_d, i_q, \theta)$$

$$\phi_q(i_d, i_q, \theta)$$

$$T(i_d, i_q, \theta)$$

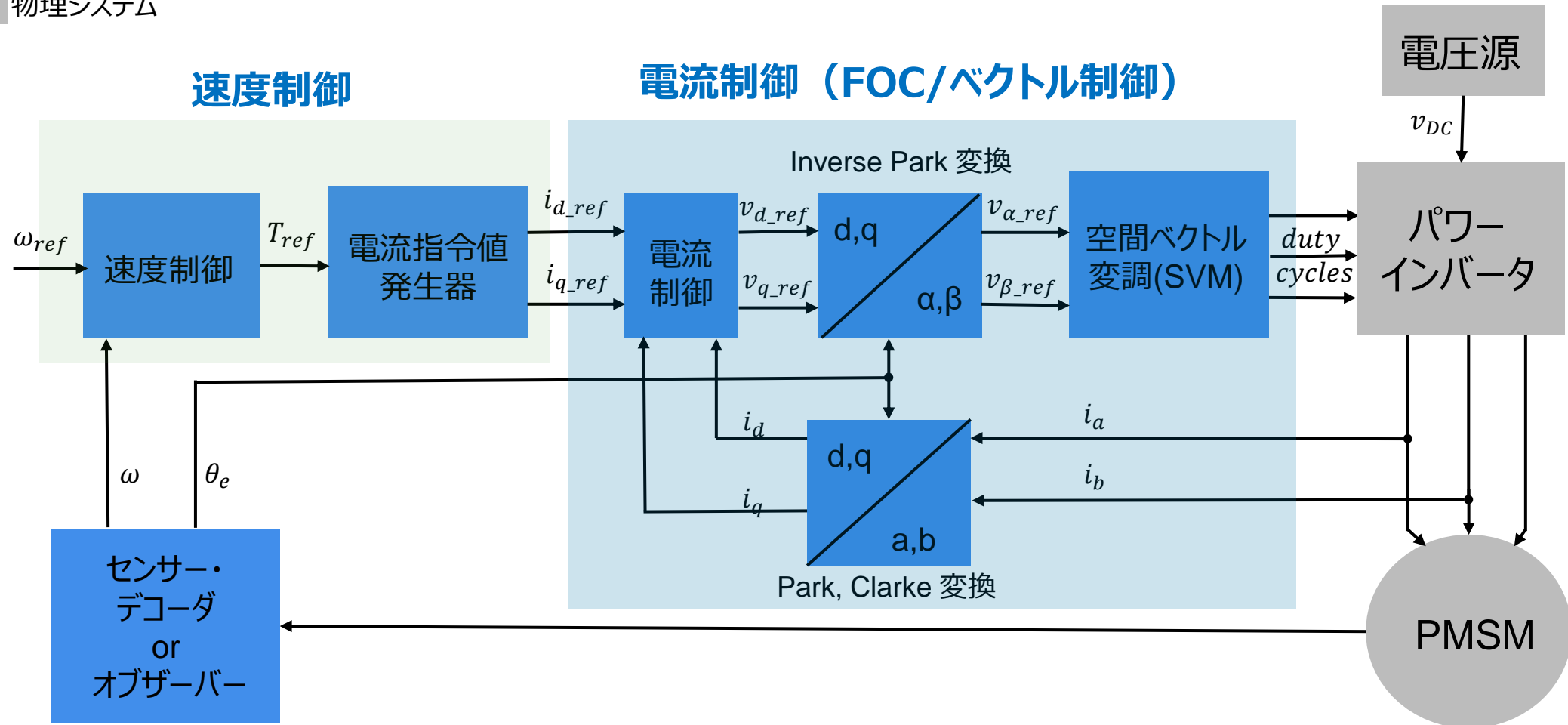
# JMAG-RTを活用した詳細モデルの作成



詳細 3 相モーターモデル [Link](#)

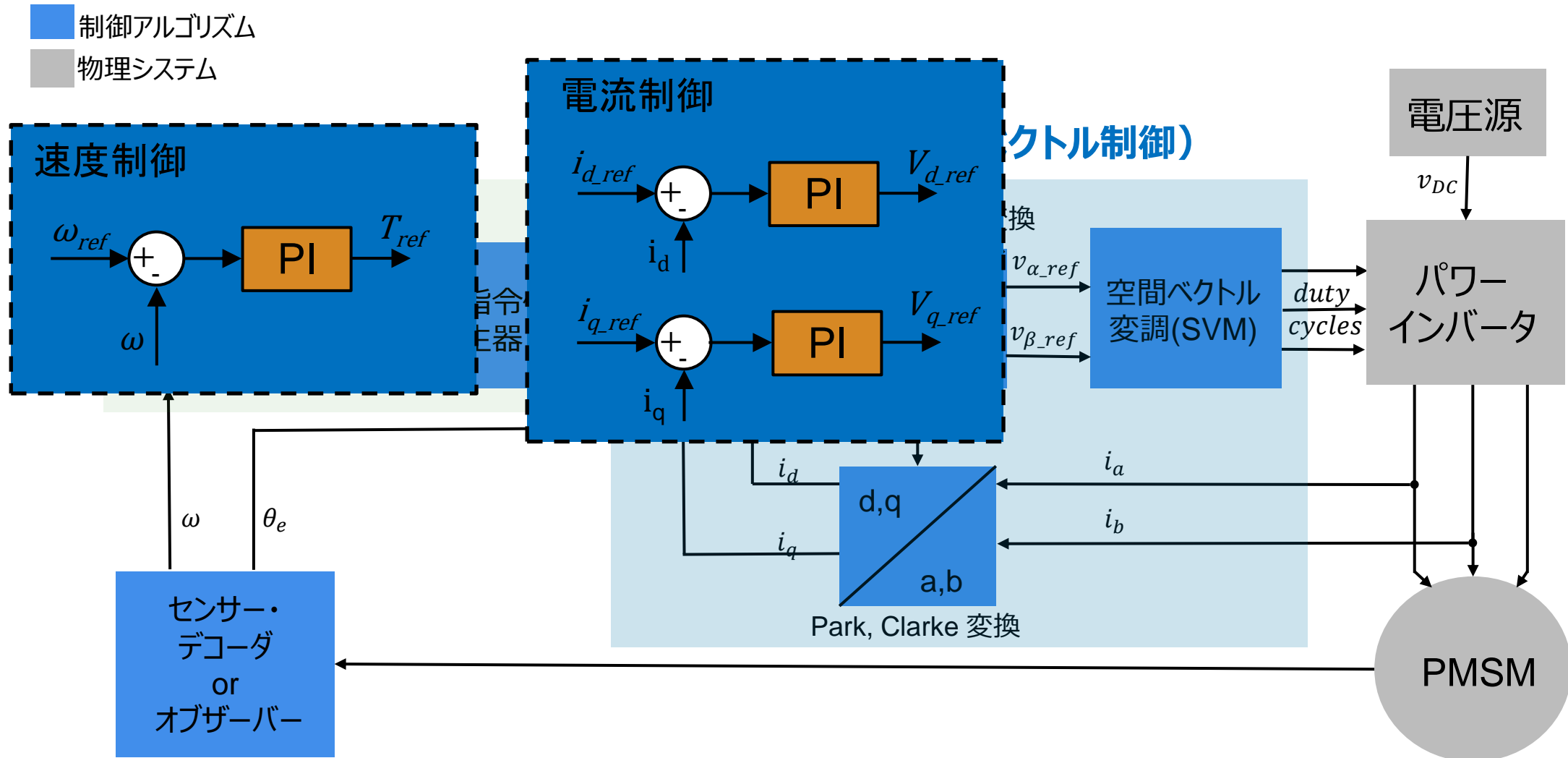
# モーター・速度制御 ブロック図

■ 制御アルゴリズム  
■ 物理システム





# モーター・速度制御 ブロック図





## 目標性能を達成 ベクトル制御のゲイン値を計算で求めます

- **mcb\_SetControllerParameters**関数で、モーターの諸元値よりPIゲイン値を計算します

- **PI 制御ゲインの自動計算**

- d軸電流ループ :  $PI\_params.Kp\_Id / Ki\_id$
- q軸電流ループ :  $PI\_params.Kp\_Iq / Ki\_Iq$
- 速度ループ :  $PI\_params.Kp\_Speed / Ki\_Speed$
- 弱め磁束制御ループ :  $PI\_params.Kp\_Flux / Ki\_Flux$

```
%% Set PWM Switching frequency
PWM_frequency = 20e3; %Hz // converter s/w freq
T_pwm = 1/PWM_frequency; %s // PWM switching time period

%% System Parameters // Hardware parameters
% Motor parameters
pmsm = mcb_SetPMSMMotorParameters('BLY171D');

%% Inverter Parameters
inverter = mcb_SetInverterParameters('BoostXL-DRV8305');

SI_System = mcb_SetSISystem(pmsm);

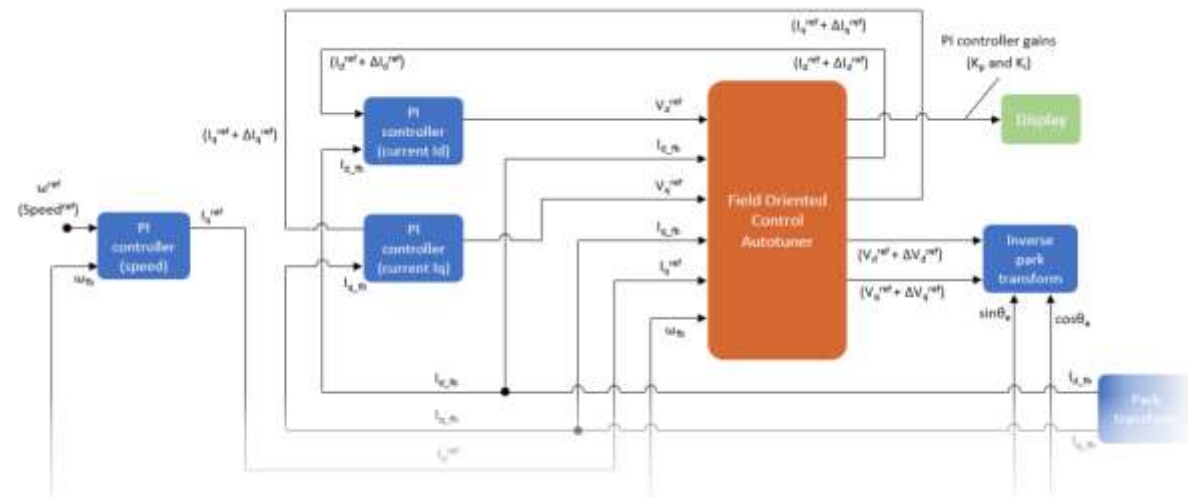
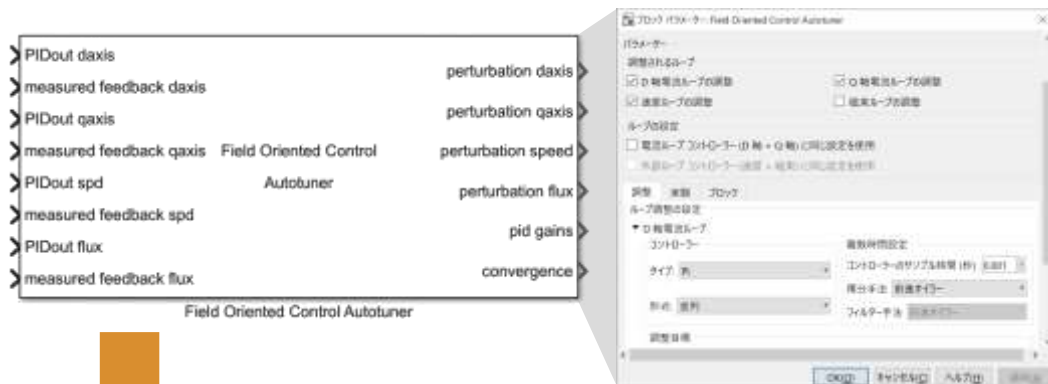
%% Set Sample Times
Ts = T_pwm; %sec // Sample time step for controller
Ts_inverter = T_pwm/2; %sec // Simulation time step for average value i
Ts_speed = 10*Ts; %Sec // Sample time for speed controller

%% Controller design // Get ballpark values!
PI_params = mcb_SetControllerParameters(pmsm,inverter,SI_System,T_pwm,Ts,Ts_speed);
```



目標性能を達成  
時間のかかるベクトル制御のゲイン調整作業を効率化できます

## FOC Autotunerブロック



### PI 制御ゲインの自動調整

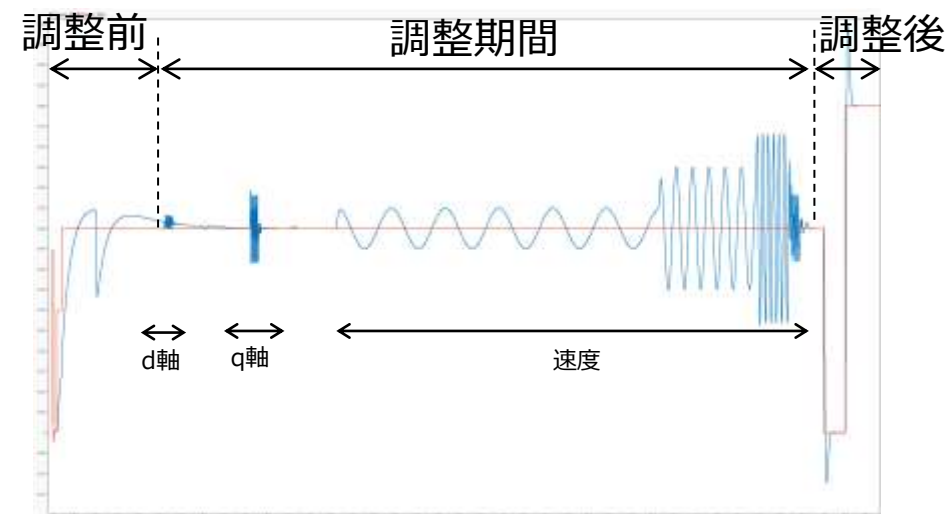
- d 軸電流ループ
- q 軸電流ループ
- 速度ループ
- 磁束ループ

C コード生成



ハードウェア実装も可能

※画像はSpeedgoatハードウェア



# JMAG-RTを活用し制御設計で活用

JMAGのモータ  
設計データファイル  
(rtt-file)



JMAG RT Viewer  
モータ設計ツール

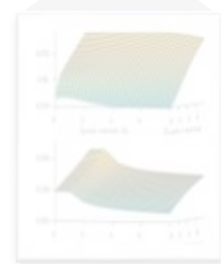
(エクスポート)



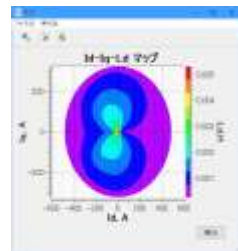
(エクスポート)



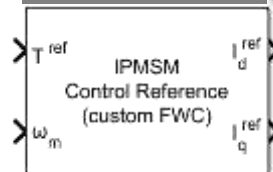
MATファイル



$\theta$ による磁束テーブル



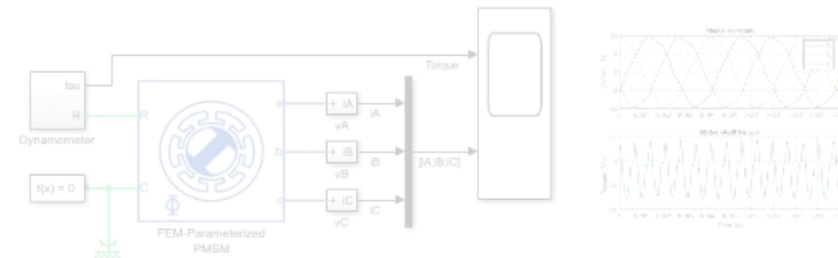
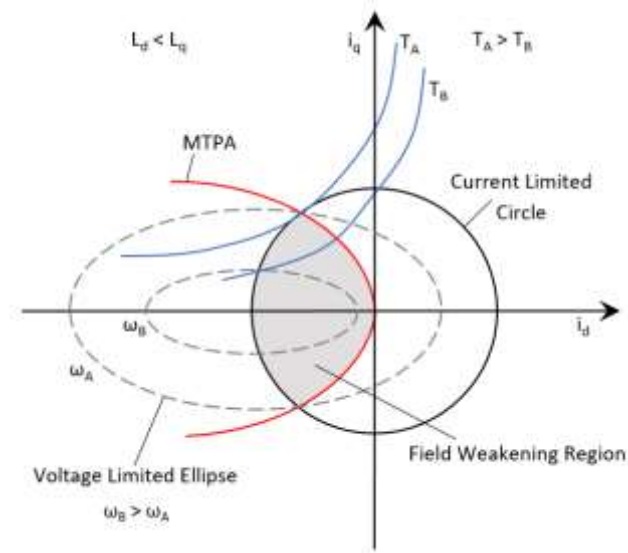
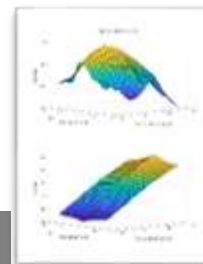
制御ブロック



LUT based PMSM Control Reference  
 $i_d(T_{ref}, \omega_{rpm}), i_q(T_{ref}, \omega_{rpm})$ ,  
テーブル設定



プラントモデル



詳細 3 相モーターモデル [Link](#)





# モーター制御システム設計 & 性能検討のワークフロー

1

上位要求の  
取得・整理



Requirement Toolbox™

モーター要求仕様

- トルク
- 回転数
- 温度
- 重量
- コスト
- …etc

2

モーター  
モデル作成

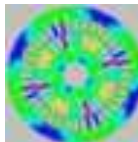
検討したいモーターを選択し  
モーターの諸元を設定する

Simscape Electrical™

PMSM



巻線界磁モータ



JMAG-RT

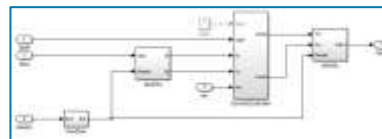
【新規モーター設計時】  
電磁解析ソフトの諸元  
値を活用

3

制御設計  
(アルゴ検討)

制御アルゴを構築し  
性能確認

Motor Control Blockset™



応答  
オーバーシュート  
外乱への感度 等

4

上位要求の  
更新・修正

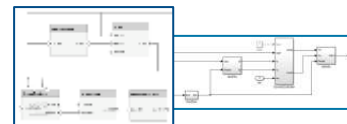
要求の妥当性確認/  
モーター選定・アドバイス



5

ソフトアーキ設計  
/ 制御設計

ソフトウェアの  
基本設計・詳細設計



System Composer™/Simulink

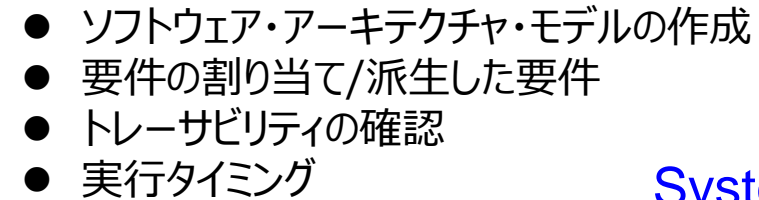
6

検証  
/ コード生成

モデル/コード検証  
自動コード生成



V&amp;V製品 / Embedded Coder™

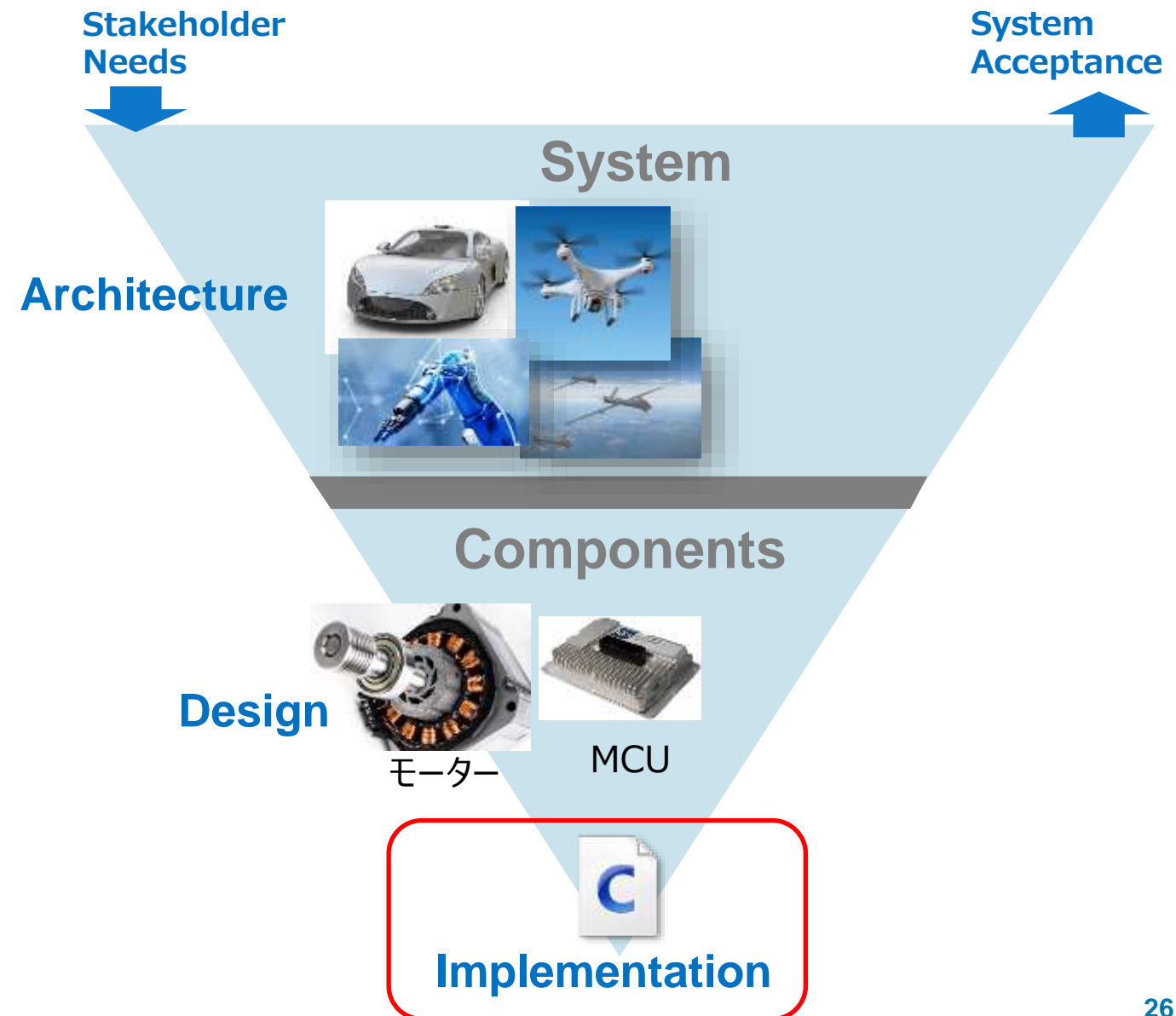


# System Composer™

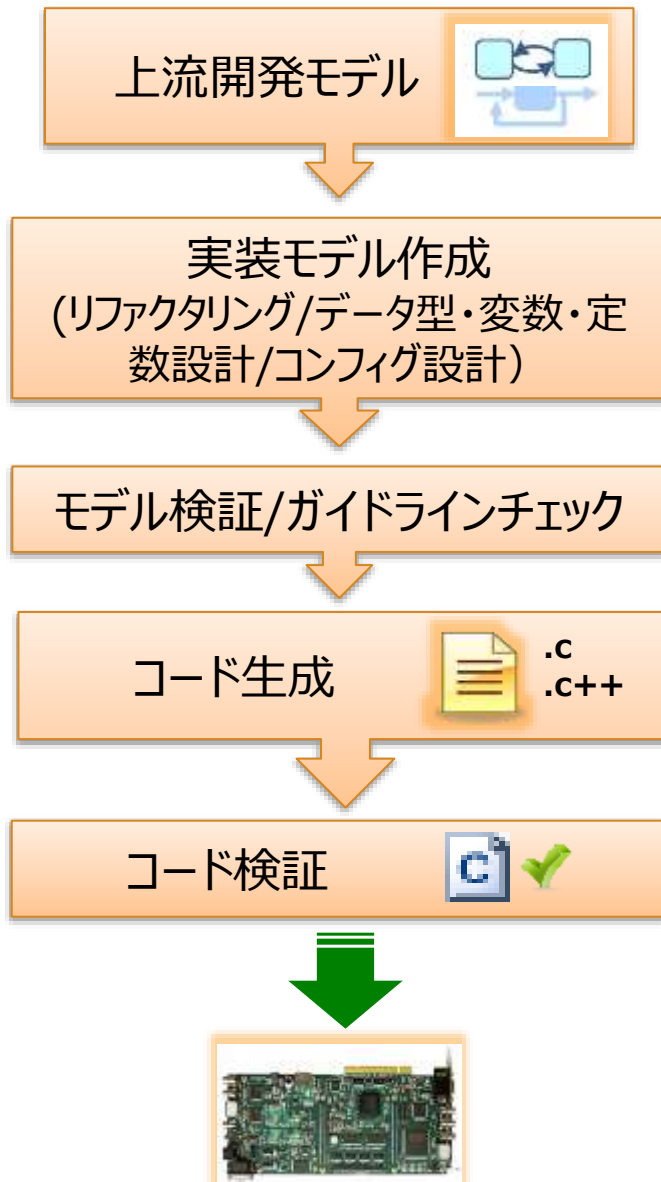


# アジェンダ

- システムのアーキテクチャ設計について
- モーター制御を例にコンポーネント設計の手順
- ➡ ■ コード生成 & 実装



# 制御モデルの量産Cコード生成 フロー



- ロジック開発・設計
- コード生成可能、かつ効率的なコードを生成するようにモデルを作成・修正
- 量子化/固定小数点化の影響を確認
- 関数I/F・コード書式の設計
- 期待値との比較/カバレッジ測定/設計エラーチェック 等
- **モデル生成コードの利用は無料**
- 静的解析/コーディングエラーチェック

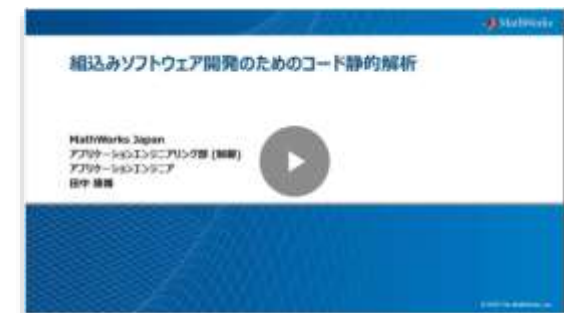
設定には、いくつかポイントがあります  
以下のビデオもご参考ください



CLICK



CLICK

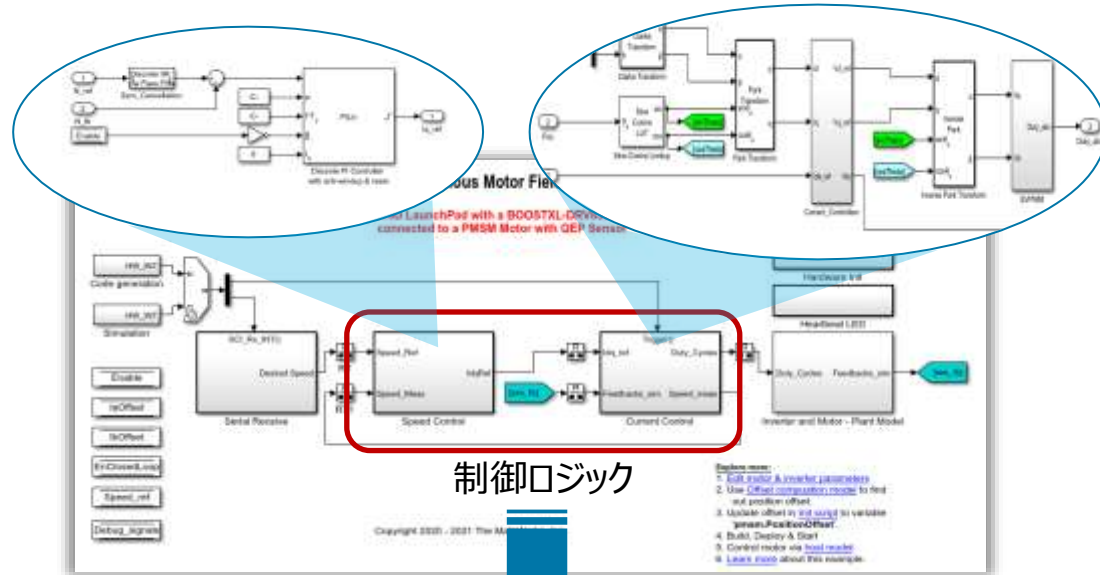


CLICK

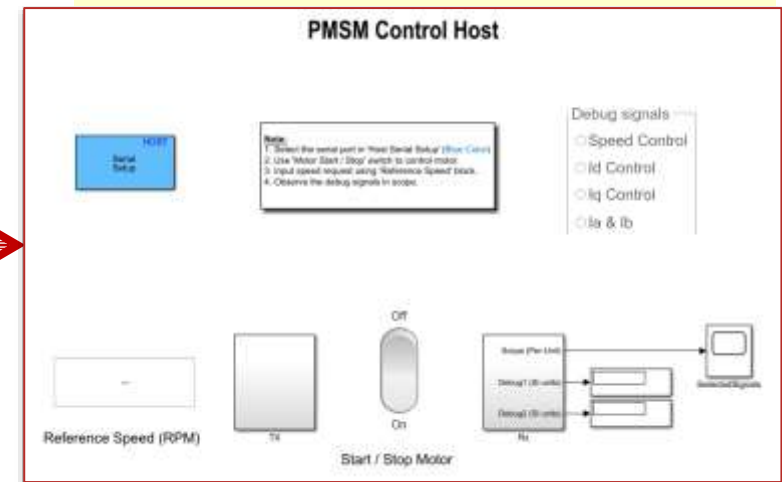




# 制御ロジックのマイコン実装 & モーターによる動作



## ホストモデル (実機とシリアル通信)



- シリアル通信の設定
- 速度指令値の変更
- 制御指令 (開始/停止)
- モーターの各種時間波形を Scope に表示

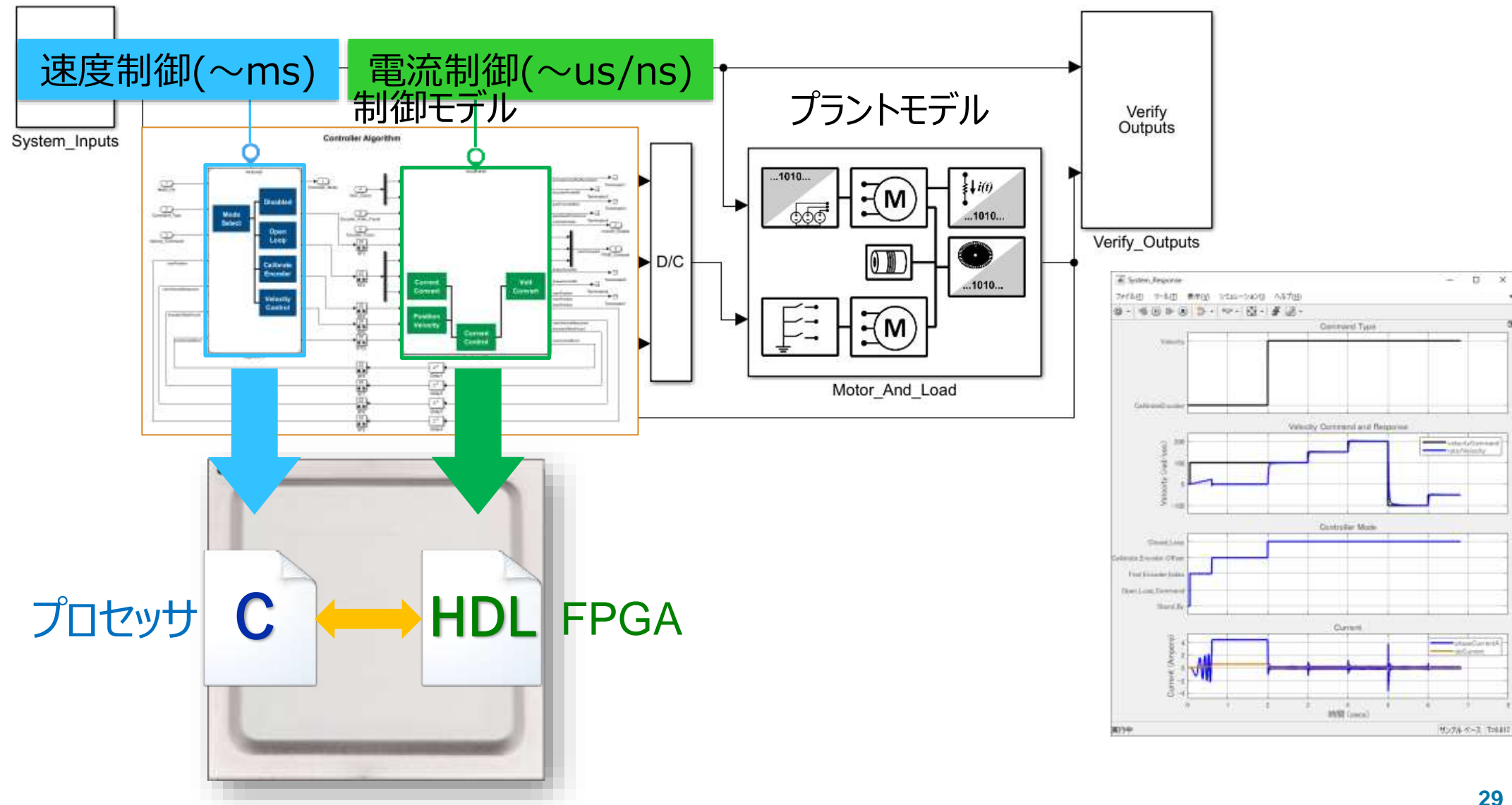


## TI社製のモーターハードウェアキット



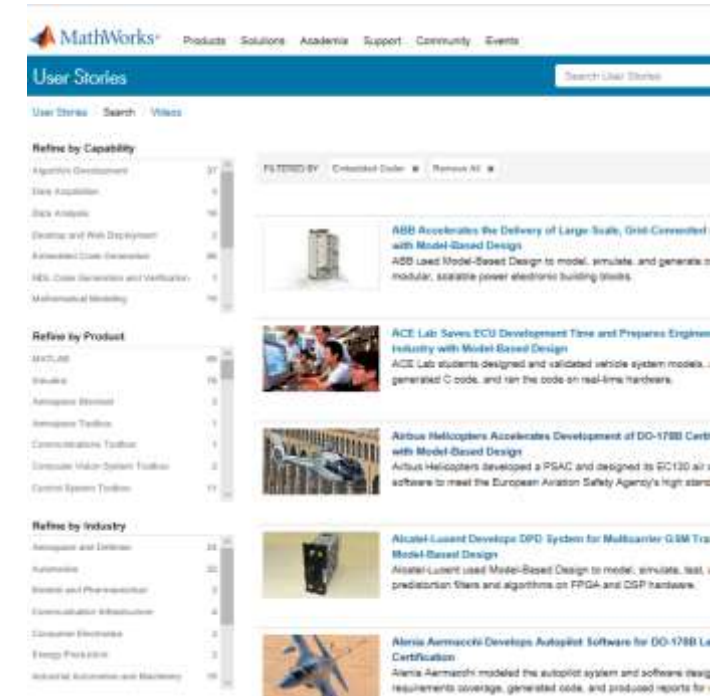


# 制御モデルをSoC / FPGAにも実装可能



# 多くのお客様がモデルベースデザインで高い ROI を達成

ABB	生産性が10倍向上
Airbus	ソフトウェアのテスト時間を2/3に短縮
ATB	2倍のスピードでプロジェクトを完了
B&R	KPI が30%以上改善
BAE	ハンドコーディングの1.5~2倍の効率化
Baker Hughes	新規プロジェクトのリソースを50%削減
Chery	年間200万ドル近くを削減
CNH	開発期間50%削減、生成コードは即座に運用可能
Continental	6ヶ月の労力を削減、検証時間を50%削減
Danfoss	開発期間を最大15%短縮
Embraer	開発期間を6ヶ月以上短縮
Honeywell	生産性が5倍向上
IVECO	開発期間40%短縮
KARI	開発期間半減、設計の繰り返し最小化
KOSTAL	認証にかかる時間を30%短縮
Lear	開発期間40%短縮、保証問題報告0件
Lockheed Martin	開発効率2倍、設計更新は1日で



詳細はユーザー事例 Web をご覧ください

[https://www.mathworks.com/company/user\\_stories.html](https://www.mathworks.com/company/user_stories.html)

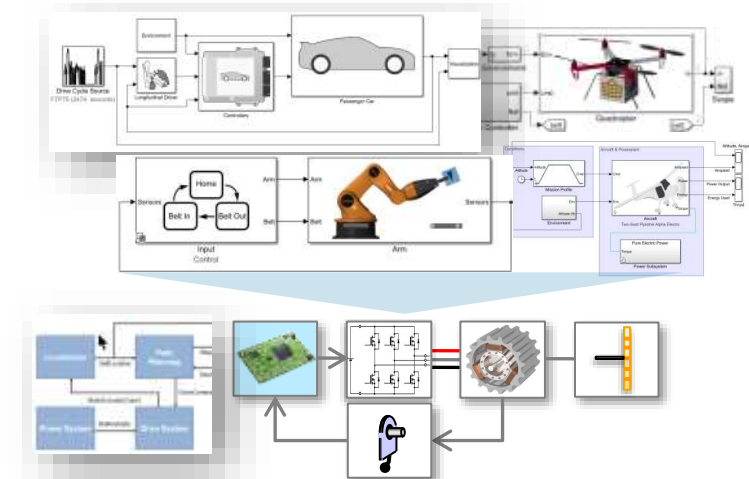
**40~50% の  
コスト削減を実現**

## まとめ

- 本セッションの説明内容の振り返り
  - ✓ MATLAB/Simulinkを活用したシステムエンジニアリングによる開発フローの進め方
- MATLAB製品の強み
  - ✓ 一気通貫で設計意図を伝搬する製品群 & 効率的に作業が行える環境を提供
  - ✓ シミュレーション高速化における様々なアプローチ



システム開発で、MATLAB製品を是非ご利用下さい。





Accelerating the pace of engineering and science

© 2023 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [www.mathworks.com/trademarks](http://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.