



EPS用ECU試作開発におけるMBDの適用

小林 将之

1 はじめに

従来の組込み制御システム開発の多くは、ドキュメントベースの設計とハンドコーディングにより行われてきた。しかしながら、自動車分野を中心に電子制御システムの高性能・多機能化が進む一方、高品質・低コストかつ開発期間の短縮化が要求されている。KYBの代表的な電子制御システムの一つである電動パワーステアリング（以下EPS）においても開発の効率化が求められており、従来とは異なるプロセス・手法への取り組みが必須である。

近年、組込み制御システム開発の手法として設計・実装の見える化が可能なモデルベース開発（以下MBD）^{注1)}が注目されており、実際に自動車業界では広く採用され、多くの実績がある。

本報ではEPS用ECU試作開発においてMBDを導入したので、その取り組みについて紹介する。

注1) Model Based Developmentの略。

2 MBD導入の背景

2.1 従来開発手法の問題点

組込み制御システム開発には必ず仕様書が存在するが、ドキュメントベースの内容を読み手に100%伝えることは難しい。例えば読み手が仕様書の内容を取り違え、テスト工程において不具合検出された場合、設計段階までの大きな手戻りが発生してしまうというリスクがある。実際にこのようなシチュエーションは現場で頻繁に繰り返されており、膨大なリソースをかけて対応してきた経緯がある。しかし、開発の効率化が求められている現状において、従来の手法では既に対応しきれない段階にある。

2.2 MBD概要

MBDとは組込み制御システム開発のプロセスに、シミュレーション可能なCAEツールを用いることで、図1のような開発ライフサイクル^{注2)}全般を通して品質向上と開発効率向上を目指した開発手法の

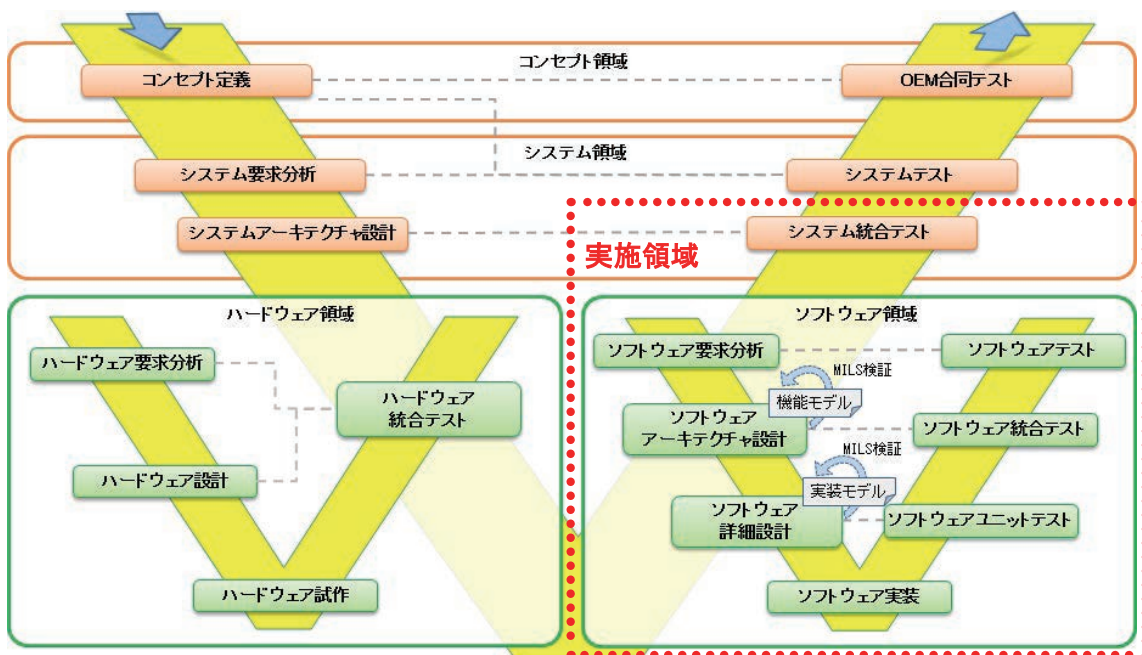


図1 開発ライフサイクルのV字モデル

ことである。V字の左側半分が設計フェーズ、右側半分がテストフェーズを表し、ソフトウェア領域のアーキテクチャ設計にて機能モデルの作成・検証、詳細設計にて実装モデルの作成・検証を実施する。今回の開発ではソフトウェア領域について取り組み、その検証としてシステム統合テストを実施した。

以下にMBDの特徴、及び導入におけるメリットを示す。

①仕様書の明確化

仕様・機能の表現をモデルという共通言語に置き換えることで、直感的に理解しやすい明確な仕様書となる。海外拠点とのコミュニケーションにも有効である。

②フロントローディング

仕様書が実行可能なモデルであることから、その都度シミュレーションによる仕様の妥当性を検証することができる。これをMILS^{注3)}といい、制御モデルとプラントモデル^{注4)}を組み合わせることで設計段階から開発対象の動作確認が可能である。開発の上流工程を重視することで、下流工程からの手戻りリスクを削減できる。

③自動コード生成

モデルから自動でソースコードを生成することができるので、プログラマのヒューマンエラーやスキルによる可読性、実行効率などの品質レベルを一定にできる。なおかつコーディングに掛かる工数の大幅削減が可能である。また、コード生成により仕様書とソフトウェアが必ずイコールの関係となるため、管理が容易である。

④再利用

過去に開発したモデルをライブラリ管理することで、仕様書レベルでの再利用が容易となる。その結果、開発効率の向上だけでなく、ノウハウの蓄積や資産の増強にも繋がる。これは制御モデルだけではなく、プラントモデルにも言えることである。

注2) 製品の構想から開発、運用、保守といった一連のプロセスを定義したもの。

注3) Model In The Loop Simulationの略。制御対象モデルとコントローラモデルを組み合わせる行うシミュレーション。

注4) モータ等の制御対象の挙動を運動方程式に置き換えた物理モデル。

3 ソフトウェア構成

今回のソフトウェアは、開発するモデルをアプリケーション部に限定するためAUTOSAR^{注5)}準拠の

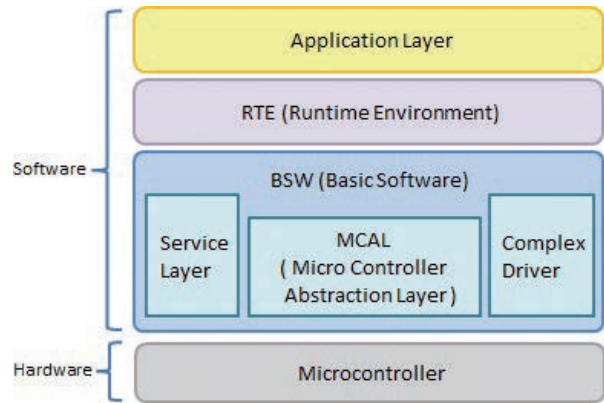


図2 AUTOSAR構成

構成とした(図2)。

Application Layerがモデルで開発するアプリケーション部に該当する。

BSWはHardwareとSoftwareを繋ぐ階層であり、OSを提供するService LayerやMCAL^{注6)}、高度な機能を使用するためのComplex Driver等から構成される。

RTEはBSWとApplication Layer間を繋ぐコミュニケーション階層である。

上記のような階層構造を持つことにより、アプリケーションは再利用可能な一つのモジュールとして扱うことができ、Hardwareを意識することなく開発が可能となる。

注5) Automotive Open System Architectureの略。車載ソフトウェアプラットフォームの標準化を制定している組織及び規格のこと。

注6) Microcontroller Abstraction Layerの略。マイコン内部にアクセスするためのソフトウェアモジュール。

4 開発プロセス

以下、開発プロセスに従い本開発で実施した内容について紹介する。

4.1 要求分析

開発対象のコンセプトから要求を抽出し、要求を実現するための具体的な方法をシステム、ソフトウェア仕様に落とし込む(図3)。

4.2 機能モデル

アーキテクチャ設計にて、要求分析のアウトプットに基づいて機能をモデル化する(図4)。この機能モデルにてMILSを実施し、フェイルセーフを含むシステム動作や要求が実現可能であるかを検証・確認する。モデル作成にあたっては、モデルの可読性向上を図るため、MAAB^{注7)}をベースとし記述ルールを定めたモデリングガイドラインを作成した。

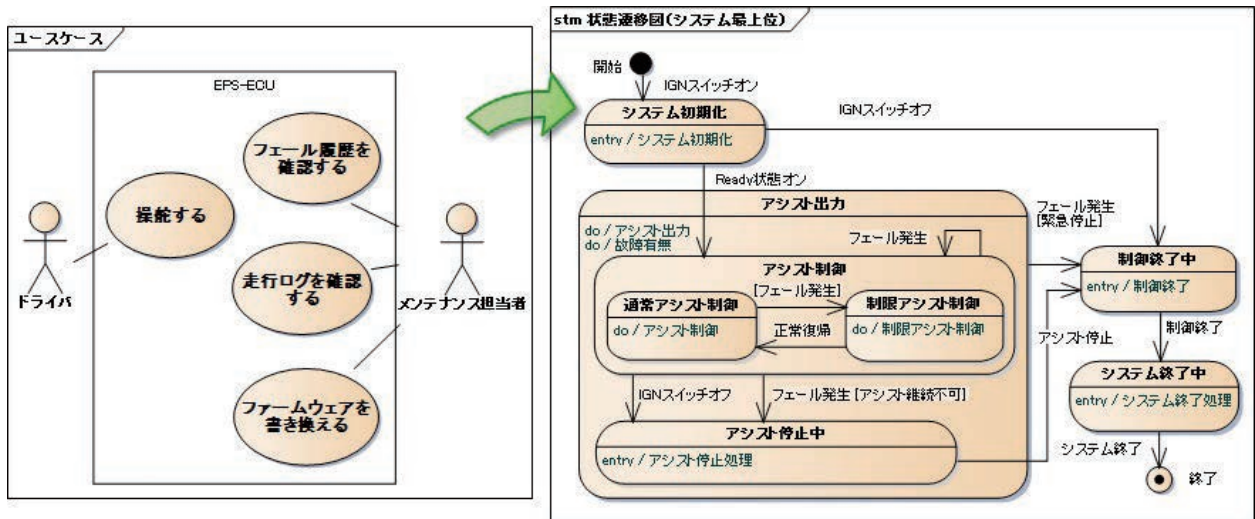


図3 システム・ソフトウェア要求分析

モデリングツールは当社内で使用実績が多い Mathworks社のMATLAB[®]／Simulink[®](注8)を採用した。なお、プラントモデルは過去に当社で開発したモデルから流用している。

注7) Mathworks Automotive Advisory Boardの略。

Mathworks製品における記述ルール等の規約を定めたガイドライン。

注8) アルゴリズム開発、システムシミュレーションの

ためのグラフィカル環境。MATLAB[®]、Simulink[®]は Mathworks社の登録商標である。

4.3 実装モデル

ECU実装におけるプログラムのメモリ・パフォーマンスを意識して機能モデルを修正する。連続系モデルの離散化やコンポーネント単位での分割、変数の型の適正化等の修正を施し、Back-to-Backテストにより機能モデルと実装モデルが等価であることを

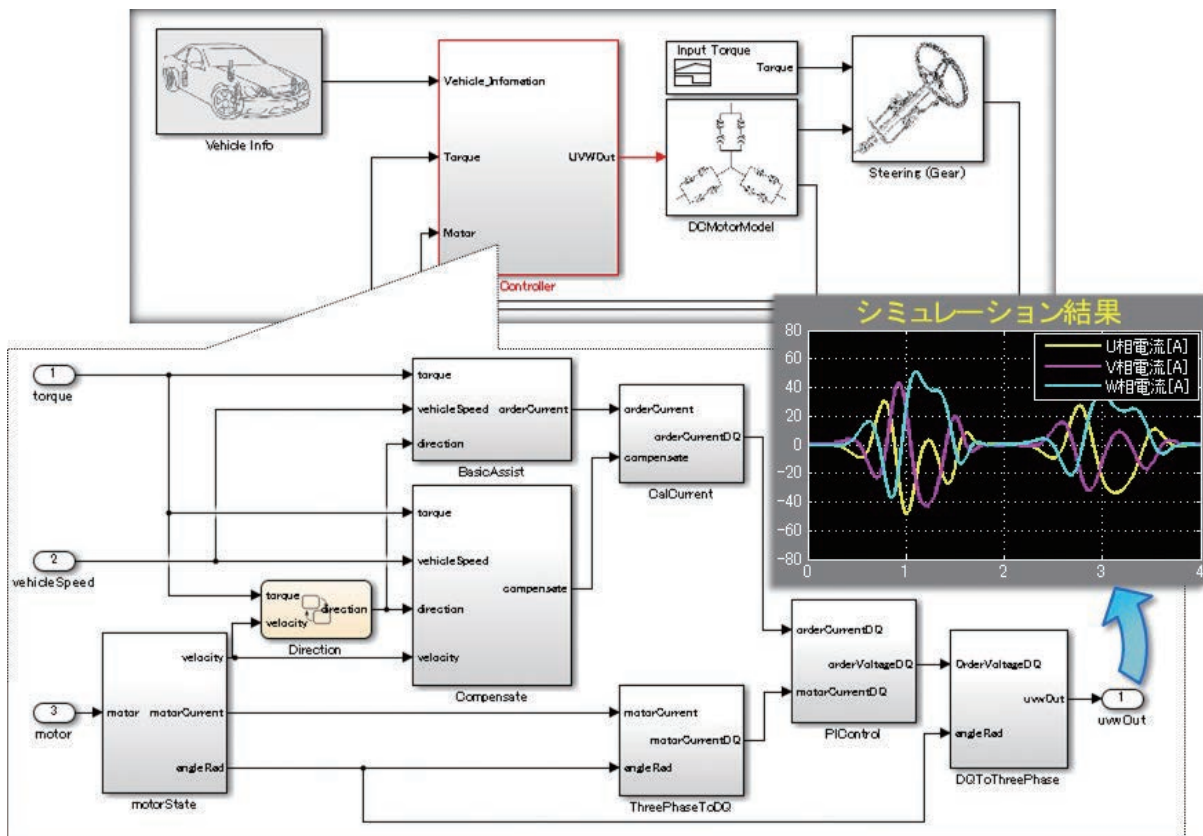


図4 MATLAB[®]／Simulink[®]で作成した機能モデル

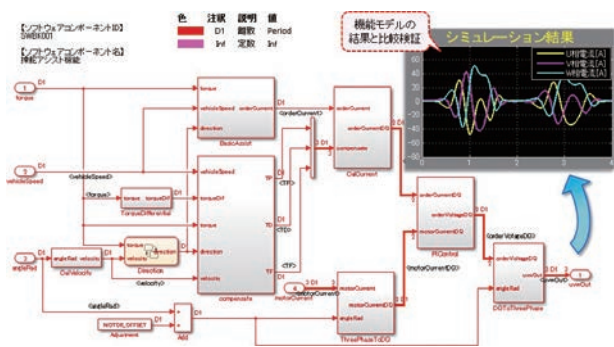


図5 MATLAB®/Simulink®で作成した実装モデル

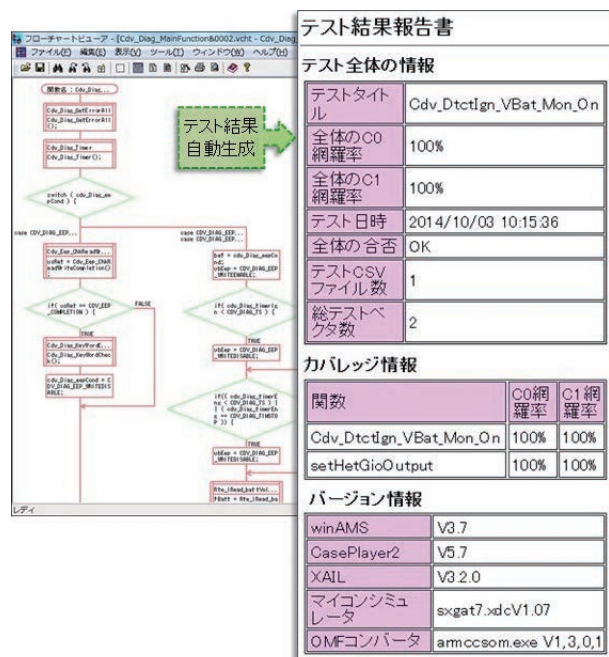


図6 カバレッジ計測

確認する(図5)。

4.4 実装

実装モデルに対しコード生成を実施する。生成されたCコードは静的解析ツールでMISRA-C^{注9)}ルールの適合性をチェックし、不適合な項目に対し処置を施す。

注9) ソフトウェア(C言語)における安全性、可搬性、信頼性の確保を目的としたコーディング規約。

4.5 テスト

以下、本開発で実施した主なテストについて説明する。

4.5.1 ソフトウェアユニットテスト

ソフトウェアユニット毎にマイコンシミュレータによるユニットテストを実施する。Back-to-Backテスト、カバレッジ計測により詳細設計に対する整合性を検証する(図6)。また、ISO26262^{注10)}対応のユニットテストツールを使用することでエビデンスが自動生成される。

注10) 自動車の電気・電子に関する機能安全規格。

4.5.2 ソフトウェア統合テスト

ソフトウェアユニットを段階的に統合し、マイコンシミュレータでテスト用プログラムと組合せてテストを実施する。統合したソフトウェアが正しく動作することを確認し、ソフトウェアアーキテクチャ設計に対する適合性を検証する(図7)。

4.5.3 ソフトウェアテスト

統合したソフトウェアを実装したECUにセンサと制御対象を組合せてテストを実施する。また、マイコンにおけるメモリ使用量やCPU占有率も計測し、ソフトウェア要求に対する整合性を検証する。

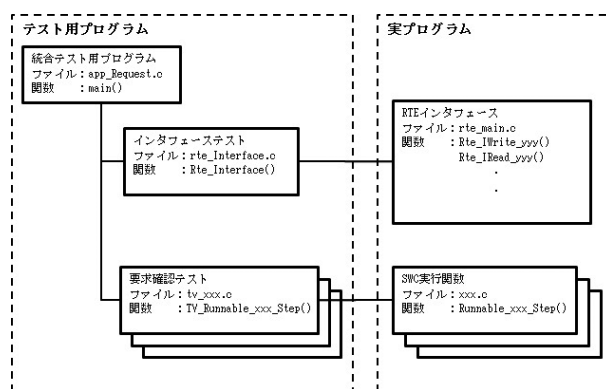


図7 ソフトウェア統合テスト時におけるプログラム構造

4.5.4 システム統合テスト

システムを構成する個々の要素を段階的に統合する。テストにはステアリングとギヤボックスが組み込まれている台上試験機を使用し、基本機能の他にロバスト性の評価も実施する(写真1)。台上試験機にてステアリング操作した際のシステム動作波形を図8に示す。

システムアーキテクチャ設計に対する適合性を検証し、テスト結果のエビデンスより、システムが完成していることを確認する。

5 まとめと課題

開発の上流工程にモデルを適用することで、設計段階で要求仕様の妥当性を先行検証することが可能となり、品質が向上した。更にモデルをコミュニケーションツールとして活用することで、プロジェクトメンバー間の共通認識が高まり、開発過程でのコミュニケーションが円滑となった。その結果、作業効率が向上し、多人数開発におけるMBDの優位性を確認することができた。また、ガイドライン制定等の実施環境整備にも取り組み、MBD本格導入に

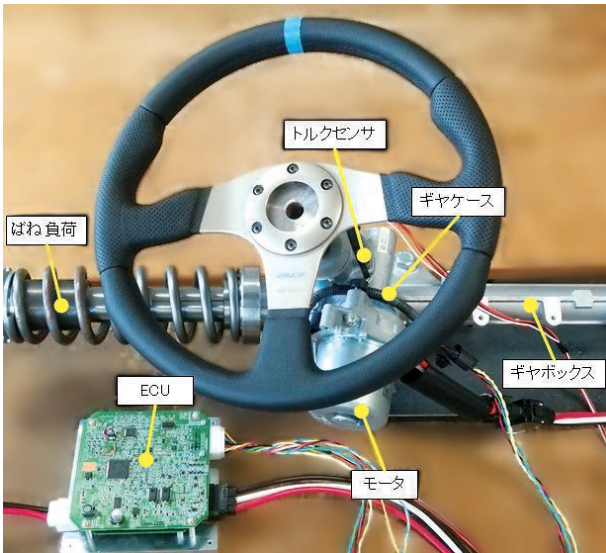


写真1 台上試験機

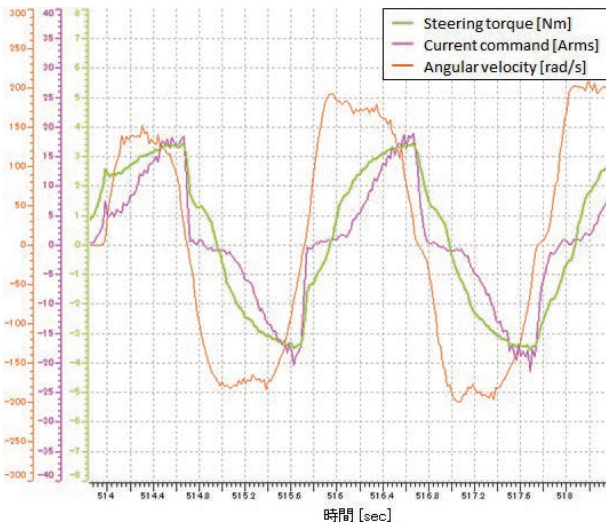


図8 ステアリング左右90°操作時の動作波形

向けての基盤強化を図った。今後はEPSの派生開発や他プロジェクトに水平展開をしていきたい。

著者



小林 将之

2010年入社。技術本部電子技術センター開発室。ソフトウェア開発に従事。

しかしながら幾つか課題も挙げられる。MBDはツールを多用することから、それぞれのツールに関する基礎知識の習得が必須といえる。そのためMBDエンジニアの育成には、多くの時間を要することである。更にHILS^{注11)}等といった環境構築、人材育成を含む開発全体において、莫大な費用が発生するため、取り組みのハードルが高い。

注11) Hardware In The Loop Simulationの略。実ECUに実車を模擬したモデルを組み合わせて行うシミュレーション。

6 おわりに

近年における自動車技術の進歩は著しく、今や運転支援機能の搭載車両は珍しくない。運転支援機能はレーンキープアシストを代表に、EPSと関わりが深いものが多い。EPSに「操舵アシスト+α」の機能を持たせることはシステムの複雑化を意味し、当然ソフトウェアは肥大化を辿る。このような状況において、MBDの標準プロセス化は必然であるといえる。

また、自動車業界においてはMBD以外にも今回適用したAUTOSARやISO26262、フォールトトレラント^{注12)}設計等々、話題は絶えない。このようなニーズの変化や規格対応のため、我々技術者は常に業界の動向に注目しながら、環境・体制の準備をしていなければならない。

注12) システムの一部に障害が発生した場合でも停止することなく継続して動作し続けるようにすること。