

モデルベース開発の取り組み

木谷 智哉^{*1}
Kitani Tomoya

モデルベース開発 (Model-Based Development : MBD) は 20 年以上の歴史があり、検証を行いながら設計を進めることができるため、開発時間の大幅な短縮と品質の向上が見込める開発手法である。このことから、自動車業界などを中心に広まっているが、一般化したとは言えない。

当社は 2000 年ごろから、制御装置開発にモデルベース開発を適用してきた。本稿では、当社のモデルベース開発によるプロセスおよびその取り組みについて紹介する。

キーワード：モデルベース開発、MBD、MATLAB/Simulink

1. はじめに

モデルベース開発 (Model-Based Development : MBD) とは、コンピュータ上に「モデル」を作成し、それをを用いたシミュレーションにより検証を行いながら制御システムの開発を進めていく手法である。繰り返し検証しながら設計を進めることができるため、開発時間短縮と品質の向上が見込め、自動車業界などを中心に広まっている。2021 年に「一般社団法人 MBD 推進センター」という組織が設立されるなど、今なお関心の高い分野でもある。

当社は 2000 年ごろから、高い信頼性が求められる制御装置開発にモデルベース開発を適用してきた。近年は一般産業向け案件への適用も増えている。本稿では、モデルベース開発について解説するとともに、当社のモデルベース開発プロセスとその取り組みについて紹介する。

2. モデルベース開発の特徴

モデルベース開発には大きく二つの特徴がある。一つは、視覚的にアルゴリズムを記述できるため、直感的に内容を理解しやすく、言語の制約にとらわれず設計に専念でき、かつ、機能ブロックで構築したアルゴリズムを、オートコード機能によって自動でプログラムを生成できる点である。

図 1 に、一般的な PID 制御 (※ 1) をモデル化した例を示す。

このように『入力に対し計算や変換を行い出力させる』という一連のプログラムを、モデルベース開発では各種の機能を持ったブロックを配置しつなぐことで、一連のプログラムを視覚的に記述する。

従来型の開発手法では、仕様書からシーケンス図などを書き起こし、それを手作業でコード化するため、管理が煩雑かつコード化の工程でヒューマンエラーが入り込む可能性があった。一方モデ

*1：制御システム事業部 防衛・宇宙システム部 防衛・機器グループ

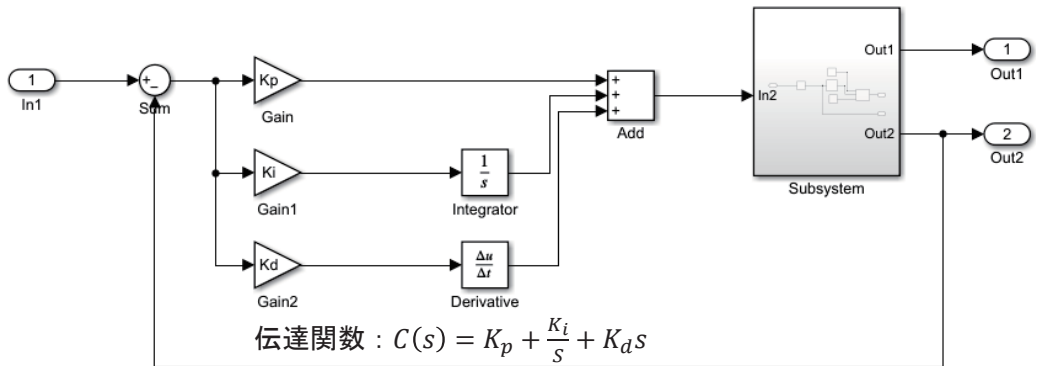


図1 モデル例 (PID 制御)

ルベース開発では、シーケンス図とプログラムが図1のように「モデル」という形で一本化され、またプログラムもオートコード機能を用いて生成されるため、コード化でのヒューマンエラーをなくしている。

もう一つの特徴は、コンピュータ・シミュレーションによる上流設計での動作検証が可能であるため、ハードウェアがない状態でもシミュレーションが容易にできるという点である。

図2に、従来型開発とモデルベース開発のプロセスを示す。従来型の開発では、設計段階の検証は机上の計算までしか行えず、ハードウェアに組み込んだ後に動作を確認している。そのため全体の工程がハードウェア開発の進捗に左右され、さ

らにハードウェアを用いた検証まで問題点が発覚しないなど、ハードウェア組込試験での時間・コスト増大が一つの課題となっている。

モデルベース開発では、制御対象となるシステムをモデル化することで、ハードウェアを必要とせずにコンピュータ上のシミュレーションで詳細な検証を行うことができる。これにより、基本設計の問題点の早期洗い出しが可能となり、高い信頼性を確保することができるとともに、大きな後戻り作業を生じさせることなく次工程へと進むことができる。

※1: Proportional Integral Differential。出力値と目標値の偏差、その積分、微分の3要素によって、入力値を制御する方法

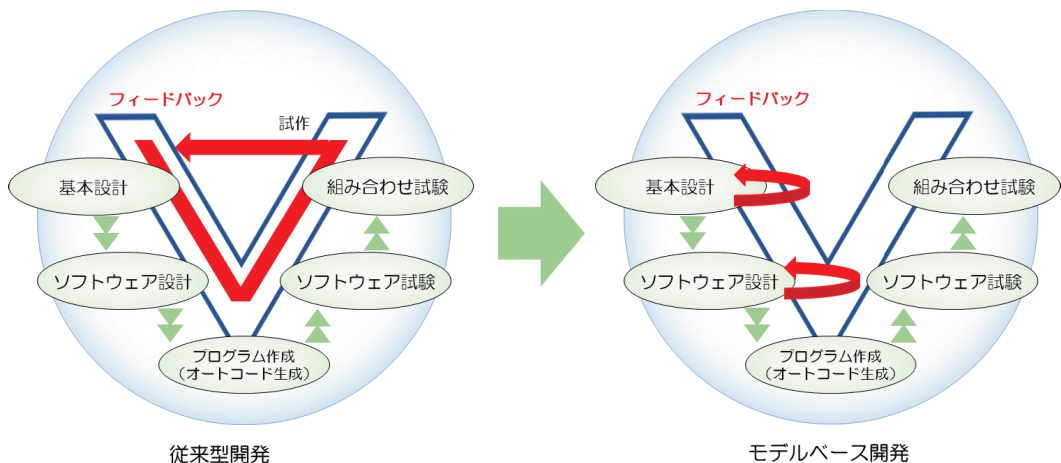


図2 従来型開発とモデルベース開発のプロセス

3. 制御装置の開発

当社の特徴として、自社開発した制御装置にモデルベース開発を用いることで、信頼性が高く耐環境性を備えた製品を提供できる点である。

開発環境として、MATLAB®/Simulink® (※2) を用いている。図3に、当社での制御装置開発におけるモデルベース開発フローを示す。

図3の赤枠箇所が、当該手法を適用している工程である。制御コントローラモデルと制御対象のモデルを組み合わせ、コンピュータ上でシミュレーションを実行する。また、それぞれのモデルからオートコード機能でプログラムを生成し、ビルド・コンパイルした実行ファイルを自社製の制御装置およびシミュレータに組み込む。その後、ハードウェア組み合わせ試験を実施し、シミュレーション検証結果と一致することで、製品の動

作検証を完了する。

※2： MathWorks 社が提供する数値解析ツール (MATLAB) およびモデルベース開発ツール (Simulink)

4. ガイドラインおよびフローの策定

4.1 ガイドライン

モデルベース開発は従来型開発と比較して大幅に開発に要する時間が短くなることから、これまでは個人あるいはごく少人数でシステムを設計・開発を担当することが多かった。しかし、モデルベース開発を推し進めていく中で徐々に適用事例が増え、チーム単位やグループ会社、あるいは他社と共同で開発する機会も増加している。

そのような中で、モデル製作に関する指標が課題の一つとなっている。MathWorks 社や MBD 推進センターから、モデルベース開発に関するガイ

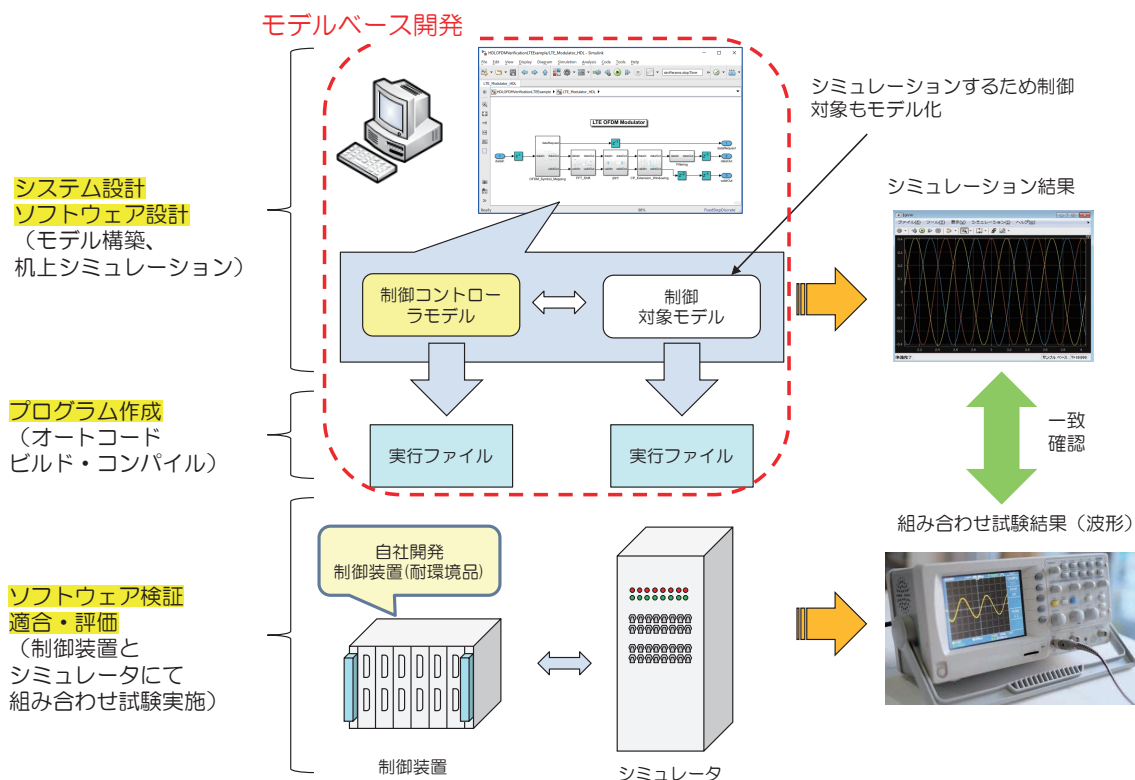


図3 推進系ロジックのモデルベース開発フロー

ドラインが開示されているが、それらは命名規則などの基本的な内容にとどまっており、モデルの製作や開発フロー全体にそのまま適用できる内容となっていない。また、自動車業界特有のルールに基づいているものが多く、一般的でない内容もある。

当社では、既存のガイドライン、および当社のこれまでの開発事例で設定していたモデル製作ルールを元に、独自のモデル製作ガイドラインを策定する活動に取り組んでいる。

図4に策定したガイドラインの一部を示す。既存のガイドラインを取り入れながらも独自に規定しており、後述する開発フローとの紐づけ、モデルの検証方法を規定することで、ガイドラインが適切に運用されることをサポートする。

4.2 開発フロー

ガイドラインを掲げる目的の一つに品質の担保があるが、そのためにはガイドラインが適切に運用される必要がある。ガイドラインの運用でとくに重要なことは、策定したガイドラインを状況に応じてアップデートすることと、開発案件で確実に適用することである。

ガイドライン適用にあたっては、ガイドラインの項目数が多いことが課題であるが、モデルベース開発フローのステップ(要素)をライブラリ化し、ガイドラインとの紐づけを行うことで、必要なガイドラインを明確化することとした。

図5に開発フローライブラリ、図6に開発フロー作成イメージを示す。

開発開始時に、開発フローライブラリから必要

ルールID	md_cm_0017		
タイトル	トリガー信号の名前		
ルール			
サブID	記述内容		
a	<p>トリガー信号の接続元のブロック名と、接続先の条件付きサブシステムのサブシステム名は、同じ名称(設計意図を示すキーワード)が含まれるようにする。 なお、接続先の条件入力ブロックのブロック名については変更しない。</p> <p>【正】</p> <p>【誤】</p> <p>適用対象 Simulinkモデル全体</p> <p>適用フロー MD_CM_0001</p> <p>確認方法 モデルアドバイザー</p> <p>MATLAB Ver ALL</p> <p style="text-align: right;">最終改訂 0</p>		
根拠			
サブID	記述内容		
a	<ul style="list-style-type: none"> ・信号線の接続ミスを防止する(自動チェッカーで検出できるようになる) ・トリガー名は2022aではデフォルトで非表示でミスの防止にならないため変更しない 		

図4 ガイドラインの抜粋

ステップID	SP_CM_0001
タイトル	入出力の定義
インプット図書	アウトプット図書
XXX-XX-XXXX_要求仕様書	XXX-XX-XXXX_入出力信号表
適用ガイドライン	nr_cm_0006 md_cm_0007

ステップID	SP_CM_0002
タイトル	仕様の可視化
インプット図書	アウトプット図書
XXX-XX-XXXX_要求仕様書	XXX-XX-XXXX_機能ブロック図
XXX-XX-XXXX_入出力信号表	XXX-XX-XXXX_タイムチャート
適用ガイドライン	xx_xx_XXXX xx_xx_XXXX xx_xx_XXXX

ステップID	SP_CM_0003
タイトル	スケルトンモデルの作成
インプット図書	アウトプット図書
XXX-XX-XXXX_入出力信号表	スケルトンモデル
XXX-XX-XXXX_機能ブロック図	
適用ガイドライン	nr_cm_0006 md_cm_0007

図5 開発フロー ライブラリイメージ

なステップを選択し、ラインでつなぐことで開発フローを規定、WBS（※3）としての機能も持ちながら各ステップで必要なガイドラインを明確にすることで、ガイドライン整合性確認を容易とした。

また、開発フローの作成で都度ガイドラインが抽出されるため、アップデート可否を併せて検討することで、ガイドラインを適宜見直すこととした。

当社ではガイドラインと並行して、開発フローのライブラリの策定活動についても取り組んでいる。

※3： Work Breakdown Structure。プロジェクトを作業単位に分解して構造化する手法

5. おわりに

従来は、シミュレーション結果を数値およびグラフなどで評価していたが、自動運転制御など制御対象の挙動を視覚的に表示することが当たり前となっている。その流れの中で、広く使われるようになってきたモデルベース開発による当社のプロセスおよびその取り組みについて紹介した。

今後は、これまでの経験を活かし、制御技術だけでなく、その結果が正しいか否かを容易で直感的に検証できるシミュレーション技術の強化についても取り組んでいく。

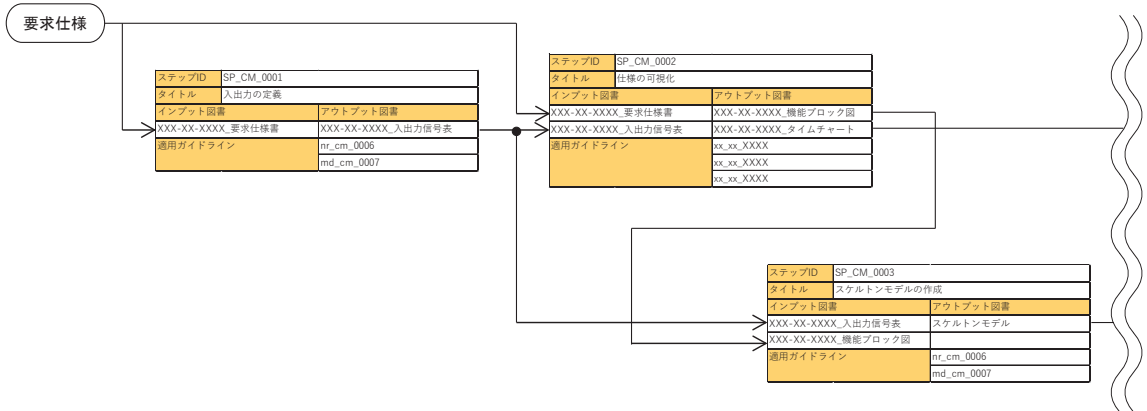


図6 開発フロー 作成イメージ

制御システム事業部
防衛・宇宙システム部
防衛・機器グループ
木谷 智哉

TEL. 045-759-2488
FAX. 045-759-2491