

# 自動車の電子化・電動化を支える ソフトウェア技術と課題

Software Technologies Supporting Automotive Electric and Electronic Systems

山内 信之

深谷 哲司

古森 誠司

■ YAMAUCHI Nobuyuki

■ FUKAYA Tetsuji

■ KOMORI Seiji

自動車の電子化や電動化の流れが加速するなか、ソフトウェアの重要性が増している。ソフトウェアの生産性を向上するためには、可搬性と実行効率を備えたプラットフォームが必要である。制御系ではハードウェアIP (Intellectual Property: 知的財産) を最大限に活用し性能を引き出すことが重要になる。一方、情報系に見られるマルチコア環境では、コスト効率の最適解を実現する工夫が重要になる。また、昨今のソフトウェア開発では、国際標準規格への対応を意識する必要がある。特にキーデバイスを効率的に活用するためには、事前のアーキテクチャ検討が重要となる。

東芝はAUTOSAR (Automotive Open System Architecture) 規格のアーキテクチャを導入してソフトウェアの標準化を進めるとともに、ISO 26262 (国際標準化機構規格26262) に基づく機能安全システムの構築を推進している。

With the expanding use of in-vehicle electronics and electroactuation in recent years, built-in software technologies have become increasingly important. A software platform with portability and efficiency is key to improved productivity. In control units such as application domains of an automotive powertrain, the best use of hardware intellectual property (IP) is essential to elicit real-time performance. At the same time, in multicore platforms such as an automotive information system, implementation of the optimal cost-effective solution is a strong requirement. In recent software development, conformance with international standards is also required based on a preliminary investigation of the system architecture.

Toshiba is promoting the development of functional safety systems according to the AUTOSAR (AUTomotive Open System ARchitecture) and ISO (International Organization for Standardization) 26262 standards.

## 1 まえがき

自動車の電子化や電動化の流れが加速するなか、ソフトウェアの重要性が増している。最近の車載用マイコンのトレンドに目を向けると、システムが要求する機能を効率的に実現するために、処理モジュールの専用IP化あるいはプロセッサのマルチコア化といった施策が顕著である。ソフトウェアは、これらのハードウェアを安全に制御し、性能を最大限に引き出せるようになっていなければならない。最近では、国際標準規格への対応が特に重要となっている。

東芝は、このような現状を認識したうえで、共通基盤技術に対する取組みを組織横断的に実施している。ここでは、自動車用ソフトウェアを取り巻くトレンドに目を向けながら、開発現場で直面する課題などについて述べる。

## 2 自動車用ソフトウェアの開発における課題

制御領域から情報領域まで広範囲にわたる自動車搭載機器用のソフトウェア開発を概観したとき、共通の課題として次のようなものが存在する。

- (1) マイコンとソフトウェアのインタフェース効率化  
マイコンとのインタフェースはソフトウェアの可搬性を維持する

うえで重要である。従来から叫ばれているソフトウェア開発の効率向上あるいは安全性向上のためにも、ハードウェアの影響を局所化することが重要になる。一方で、アーキテクチャの階層化がソリューションの候補になりえるが、極端な階層化はオーバヘッドの増大を招いてしまう。このような二律背反の課題をいかに解決するかという事態は、開発現場において常に発生する。

- (2) 国際標準規格への対応  
この課題は、完成車メーカーの意向や開発ツール導入費用の抑制などから非常に重要であり、特にソフトウェアに関しては、欧州主導で進められている次の規格が重要になる。

- (a) ISO 26262
- (b) AUTOSAR
- (c) Automotive SPICE (Software Process Improvement and Capability Determination: Automotive SIG (Special Interest Group) 規格)

これらの規格に対応するには、技術及び体制の確立が必要であり、開発現場への影響が無視できないこともある。

## 3 マイコン環境とソフトウェア

マイコンはアプリケーションごとに最適なものを選択する場

合が多く、ひと言でそのソフトウェア環境を論じるのは難しい。ここでは、自動車搭載システムの領域を大きく制御系と情報系に分けて考える。

### 3.1 制御系用プロセッサ

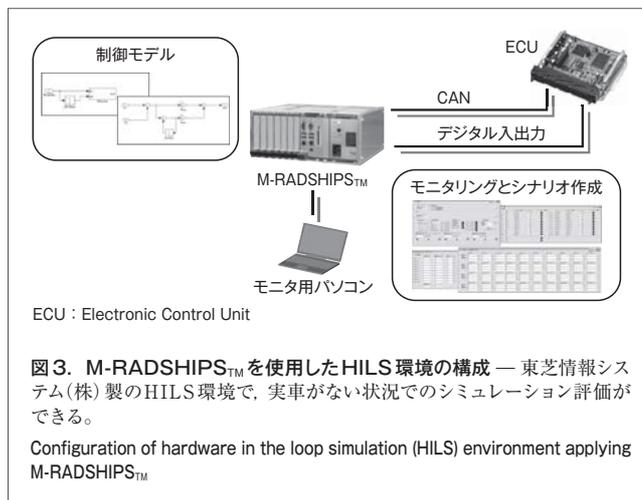
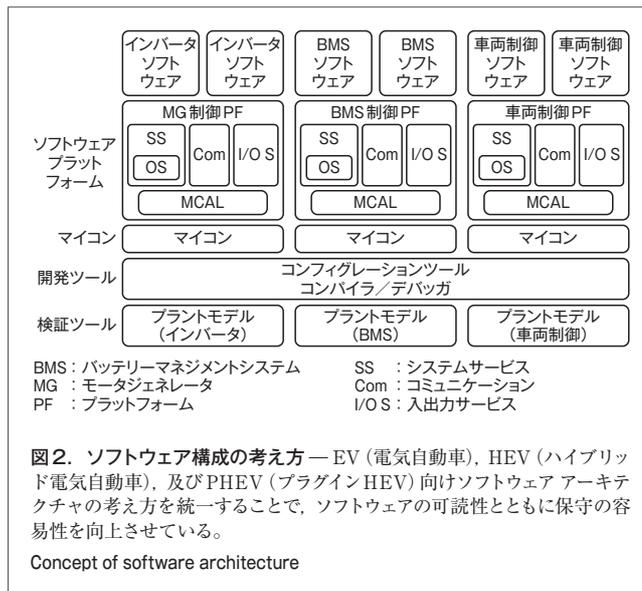
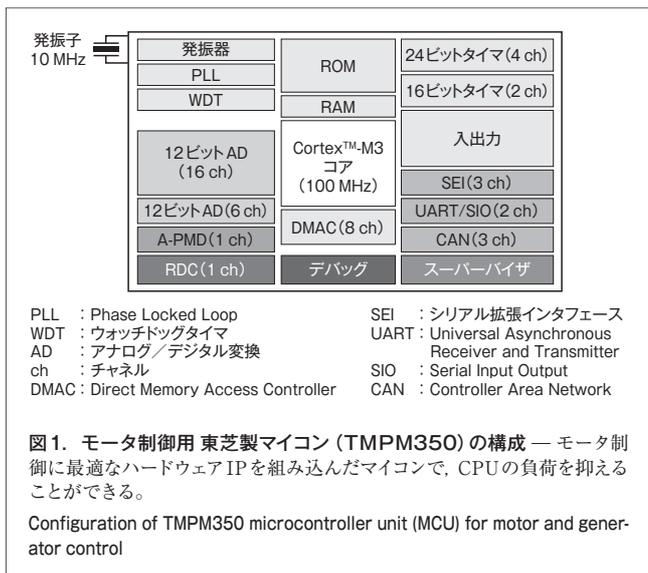
制御系システムには、高いリアルタイム性能が求められる。専用ハードウェアIPを保持することで対応する場合も多い。また、安全に対する要求も厳しく、機能安全規格への対応も必要になる。

ここで例として、当社のモータ制御用マイコン (TMPM350) の構成を図1に示す。A-PMD (Advanced Programmable Motor Driver), RDC (Resolver to Digital Converter) といったIPが設けられており、アプリケーション実行効率の向上が期待できる。同時に機能安全規格のIEC 61508 SIL3 (国際電気標準会議規格61508の安全度水準3) にも対応しており、マイコン構成部品の故障検知を効率的に行うことができる。

当社は、このマイコンを最適に制御するために、IP開発部門と連携してソフトウェアドライバのアーキテクチャを構築している。専用IPを効果的に使用するため、ファームウェアの内部動作を考慮したうえで、メインプロセッサ上の動作に最適なAPI (Application Programming Interface) を定めている。

ソフトウェア構成をAUTOSAR規格に基づいた形で図2に示す。MCAL (Microcontroller Abstraction Layer) の実行効率に気を付けながら、モータ制御、電池制御、及び車両制御に対して、ソフトウェアアーキテクチャの考え方を統一することに注意を払った。これによって、ソフトウェアの可読性ととも保守の容易性を向上させる狙いがある。

ソフトウェアアーキテクチャとともに、開発環境も重要となる。HILS (Hardware in the Loop Simulation) により、実車がない状況でのシミュレーション評価ができることが、開発効率向上のためには必須である。例として、東芝情報システム (株)



のHILS環境 M-RADSHIPS™を図3に示す。

### 3.2 情報系用プロセッサ<sup>(1)</sup>

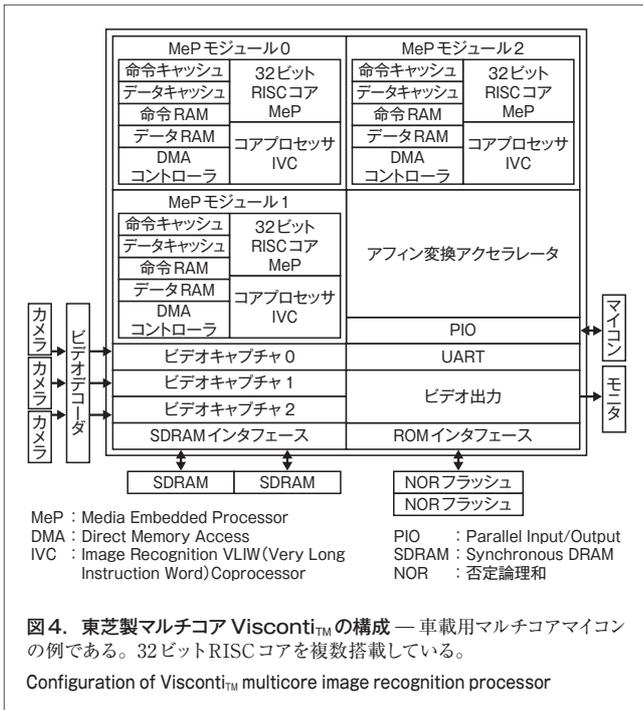
ナビゲーションシステムのような情報系の処理の傾向として、処理量の増大と制御系との連携があるが、これに対応するソリューションの一つとして、マルチコアマイコンの適用が挙げられる。

当社のマルチコアの例として、画像処理プロセッサ Visconti™の構成を図4に示す。図からわかるように、32ビットRISC (Reduced Instruction Set Computer) コアが複数含まれている。

マルチコアに関わる技術課題は多いが、ここでは、ソフトウェアプラットフォームの観点から必要な機能について述べる。単純化のため、ハードウェアとしては、密結合・共有メモリ・対称型のマルチコアであることを想定する。

マルチコアを導入した場合、コア数に比例して処理性能が向上できればよいが、実際にはこれを下回る。一般的には、次のような原因があることがわかっている。

- (1) 逐次処理部分の存在



- (2) タスクの生成時間, 準備時間, 及び消去時間
- (3) 排他処理と同期処理のオーバヘッド

タスクの逐次処理部分の占める部分が多い場合には、マルチコアの恩恵を期待することはできない。理想的なマルチコアのコア数は、アプリケーションソフトウェアの並列度に依存する。また、ソフトウェアの組み方によっても、並列化を阻害してしまう場合も多い。効率的なソフトウェアを組むためには、開発者に高いスキルが求められる。

高度な制御処理をアプリケーションソフトウェアに付加した場合には、基本ソフトウェア(OS)あるいはプラットフォームに排他・同期制御メカニズムを持たせる必要がある。この場合、一般には次の点に注意を払う必要がある。

- (1) システムの保全性
- (2) OSの利用技術

**3.2.1 システムの保全性** シングルコアとマルチコアの排他的考え方の違いに注意する必要がある。例えば、シングルコアにおける割込み禁止によるクリティカルセクションの指定は、マルチコア環境ではそのままでは適用できない可能性がある。また、カーネルモードとユーザーモードの違いによって暗黙の排他処理を行うものについても同様である。このため、シングルコアで動作していたシステムをマルチコア環境に移行する際には、OSやプラットフォームの内部構造を十分に確認することが重要になる。

**3.2.2 OSの利用技術** 次のような技術課題がある。

- (1) システム中におけるタスク制御方法
  - (2) 排他機能, 同期機能, 及び通信機能の使用方法
- システム中におけるタスク制御方法は、マルチコア環境上で

のタスクの起動, 停止といった場合のタスクの特定方法を含む。

排他・同期・通信機能の使用方法に関しては、スピンロックやバリアなどの並列処理に必要なプリミティブな制御機構が必要になる。これらの実現のためには、キャッシュメモリやバスなどの影響を考慮する必要がある。各コアにローカルメモリを持ち、ライトバック方式をとる場合にはハードウェアによるアービトレーション機構も必須になる。いずれにしても、ハードウェアとソフトウェアを連携させたアーキテクチャ設計が重要である。

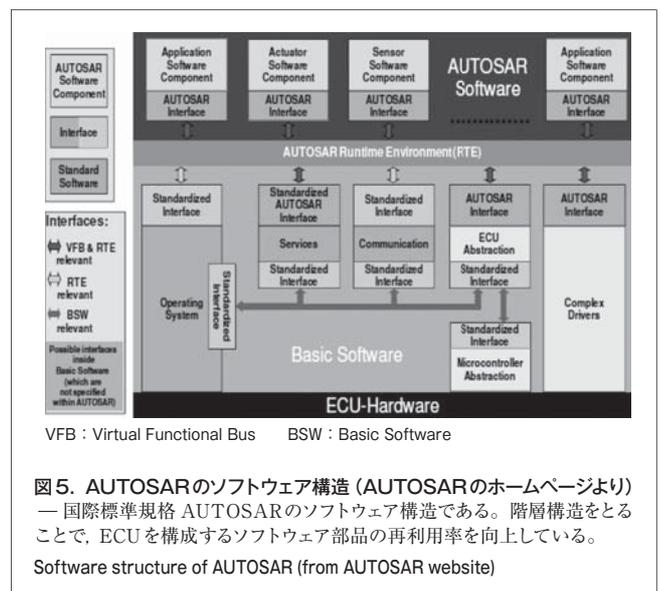
## 4 国際標準規格対応

国際標準規格である AUTOSAR や ISO 26262 は、業界としてその導入を進めようとしていることから、ソフトウェアの開発にとっても極めて重要になる。一方で、明確な手法が固まっていない、あるいは要求スキルが高いといった理由で、依然課題があるように思われる。

### 4.1 AUTOSAR

AUTOSAR は、欧州のメーカーが中心となって 2003 年 7 月に設立したコンソーシアムであり、自動車用ソフトウェアにおける標準的なソフトウェアプラットフォームのアーキテクチャを定めている。マイコン抽象化を実現する MCAL や、ECU (Electronic Control Unit) 抽象化を実現する ECUAL (ECU Abstraction Layer)、サービスを提供する SL (Service Layer) といった階層構造が設けられており、アプリケーションは RTE (Runtime Environment) と呼ばれる API で明確に分離されている。階層構造をとることで、ECU を構成するソフトウェア部品の再利用率を向上させている (図 5)。

AUTOSAR の最新バージョンは 4.0 であり、ISO 26262 を強く意識した機能安全コンセプトが盛り込まれたことが特徴と



なっている。

この規格は今後の自動車業界で急速に広まることが予想されており、当社も、AUTOSARアーキテクチャを導入し、ソフトウェアの標準化を目指しているが、次のような点において課題があることもわかってきた。

- (1) リアルタイム性能
- (2) ハードウェア機能の最適化
- (3) 適用領域

**4.1.1 リアルタイム性能** ソフトウェア階層性の実現において、実行効率が想定以上に犠牲になってしまう場合がある。これはAUTOSARの抽象化層で、その下位層の差異を十分に吸収することが期待されていることに起因している。実装上の工夫によりこの課題は吸収できるが、高いプログラミングスキルが必要となる。当社はアプリケーション分析とアーキテクチャ設計を十分に行うことで、移植性とともに行う効率に優れたソフトウェアの開発を目指している。

**4.1.2 ハードウェア機能の最適化** マイコンに具備した特定処理用ハードウェアIPの性能を十分に生かすことが課題となる。この根幹には、標準化とカスタム化の二律背反的な要素が存在する。AUTOSARの思想や作法に従いながら、特殊なハードウェアIPを効率的に制御できる構造を設計段階から十分に検討する必要がある。

**4.1.3 適用領域** AUTOSARをどの製品領域に適用するかも検討の余地がある。例えば、マルチメディア系で見られるオープン指向のトレンドに追従するためには、相応のOSプラットフォームの導入を検討したほうが良いケースもある。市場での浸透度合いを考慮すると同時に、開発やサービス提供の容易さを重要なファクターと見なす必要がある。

## 4.2 ISO 26262<sup>(2)</sup>

ISO 26262は、自動車分野における機能安全規格である。この規格は2004年から検討が始まり、現在もISOのTC(専門委員会)22/SC(分科委員会)3/WG(作業グループ)16で継続中で、2011年6月に国際標準規格として制定が予定されている。

この規格の基本的な考え方は、故障によるリスクを分析し、リスクを低減する機能(安全機能)を組み込んだ設計を行うことである。また、機能安全の実現のために、適切な開発技術と手法を用いることが推奨されている。

機能安全をソフトウェア領域に適用していくために考慮しなければならないものとして、次のようなものがあると考えている。

- (1) 開発プロセス
- (2) ハードウェア機能安全機構のサポート

**4.2.1 開発プロセス** 基本的にV字プロセスに基づいて工程が規定され、V字左側の設計工程に対応する検証に関して、形式手法といった研究レベルの手法が記述されている。形式手法は、数学的にシステムの仕様記述を行うことを前提としているため、曖昧さを排除することができ、各工程に対する

検証の正しさを客観的に示すことが可能である。

一方で、形式手法の習得には高いスキルが求められるとされている。また、組込みシステムのモデル記述自体に確立されていない要素もあるなど、適用には試行錯誤的なところがある。

したがって、規格に推奨の記述があるからといって、これをそのまま適用しようとした場合には、製品開発への影響が大きくなるケースも多い。半形式手法や使用実績による安全性の証明などを適切に実施しながら工程を進めていく必要があると考えている。

**4.2.2 ハードウェア機能安全機構のサポート** 規格のパート5(ハードウェア)において、マイコンに対する機能安全の適用が記述されている。当社も、SIL3に適用可能なマイコンをリリースしており、故障検出の精度向上と異常検出時間の高速化に寄与している。

このようなハードウェアを活用し、機能安全システムを構築するためには、システムレベルの安全要求を元にソフトウェアを作成することが必要である。このため、アプリケーション開発者とハードウェア設計者が協調してアーキテクチャ設計を実施することが重要となる。

## 5 あとがき

ここでは、自動車用ソフトウェアの開発において課題となる技術事項の概略について述べた。今後、ソフトウェアの重要性が高くなるにつれて、知的財産戦略に基づいた標準化への対応も重要になる。当社も、考慮すべき技術領域の多様化に追随するよう、取組みを強化していく。

## 文献

- (1) 安川英樹, 他. SpursEngine™のLSIアーキテクチャ. 東芝レビュー. 63, 11, 2008, p.33-36.
- (2) 余宮尚志, 他. ソフトウェアを中心とした安全設計技術. 東芝レビュー. 65, 7, 2010, p.37-40.



山内 信之 YAMAUCHI Nobuyuki

自動車システム事業統括部 自動車ソフトウェア技術部参事。  
自動車向け基盤ソフトウェア技術の開発に従事。JARI, JasPar,  
自動車技術会会員。  
Automotive Software Technology Dept.



深谷 哲司 FUKAYA Tetsuji

ソフトウェア技術センター ソフトウェアイノベーション戦略部長。  
ソフトウェア工学の研究・開発、及び東芝グループのソフトウェア開発力強化に従事。  
Innovation Strategy Dept.



古森 誠司 KOMORI Seiji

セミコンダクター社 システム・ソフトウェア推進センター ソフトウェア開発技術担当グループ長。組込みソフトウェア開発技術の開発に従事。情報処理学会会員。  
System & Software Solution Center