

DMS／HDLの別体マイコンを不要とするSoC制御技術の確立

木村 茜*
Akane Kimura

植村宙夢*
Hiromu Uemura

御田村 晃*
Akira Mitamura

神山真彦*
Masahiko Kamiyama

*三菱電機モビリティ㈱

SoC Control Technology to Reduce Separate Microcontrollers for DMS/HDL

要 旨

三菱電機モビリティ㈱は、ADAS-ECU(Advanced Driver Assistance Systems-Electronic Control Unit)、DMS(Driver Monitoring System)、HDL(High-Definition Locator)などのADAS機器製品の提供を通じて、安心安全なモビリティ社会実現に貢献している。その中でも特にDMS、HDLは高負荷処理とリアルタイム処理の両立が求められ、従来の三菱電機モビリティ㈱製システムではSoC(System on Chip)とマイコンで機能を分担し、システム要件を実現していた。しかし、近年のSoC高機能化に伴い、SoCに搭載されている複数のコアを使い分けることで、単一SoCでのリアルタイム処理と非リアルタイム処理の両立が可能になった。そこで、従来システムのマイコン機能をSoCに統合し、機能、性能、コストを検証した。

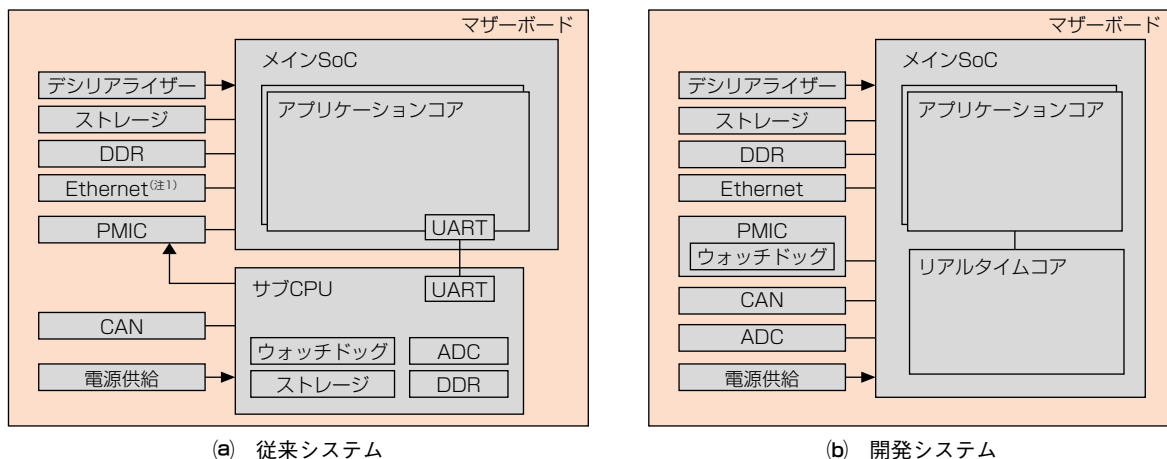
1. ま え が き

三菱電機モビリティ㈱は、人々が安心・安全に移動ができる事故のない社会の実現に向けて、車内外の情報処理技術に応用したADAS-ECU、画像認識技術に応用したDMS、高精度測位技術に応用したHDL、車外通信技術に応用したV2X(Vehicle to everything)など、ドライバーの安全な運転を支援する先進運転支援システム(ADAS)を開発している。

電動パワーステアリングのような制御系ECUには厳格なリアルタイム性が要求されるのに対して、DMSやHDLのようなADAS系ECUは画像処理など高負荷処理とリアルタイム処理の両立が必要である。

従来の三菱電機モビリティ㈱製ADAS系ECUでは、車載機器に求められる様々な性能要求を満たすため、SoCと別体マイコン(以下“マイコン”という。)の二つのCPUを搭載し、SoCに画像認識や高精度測位などシステムを特徴づけるアプリケーションを、マイコンに車載ネットワーク通信制御などリアルタイム性が求められる機能を搭載、という構成を採用してきた。しかし、このような2CPU構成のシステムは、基板上の部品点数・基板面積が増加するため原価低減が難しく、ハードウェア設計も複雑化するという問題があった。

このような問題を解決するため、リアルタイム処理が可能なリアルタイムコアが含まれる異種コア構成のSoCを使い、1CPUでありながらリアルタイム処理と高負荷処理を両立するシステム(以下“開発システム”という。)を開発した。図1



DDR : Double Data Rate, PMIC : Power Management IC, CAN : Controller Area Network, ADC : Analog-to-Digital Converter, UART : Universal Asynchronous Receiver Transmitter

(注1) Ethernetは、富士フイルムビジネスソリューション㈱の登録商標である。

図1-システムブロック図

に示すように従来システムのマイコン機能をSoCに統合し、機能、性能、コストを検証した。これによって、ストレージの一本化、リソース共有、コア間通信、システム制御の簡素化が可能になり、部品コストを30%削減しつつ、従来の機能維持と性能向上を達成した。

本稿では、2章に開発システムの特長を、3章で評価結果を述べて、最後に4章でまとめを述べる。

2. 開発システムの特長

図2に示すように、開発システムでは、SoCのアプリケーションコアに高負荷処理を行うリアルタイムOS(QNX^(注2))やAUTOSAR^(注3) Adaptive Platform(AP)を、リアルタイムコアに電源制御や車内通信を行うAUTOSAR Classic Platform(CP)を搭載し、マイコンの統合を実現した。

(注2) QNXは、BlackBerry Limitedの登録商標である。

(注3) AUTOSARは、AUTOSAR GbRの登録商標である。

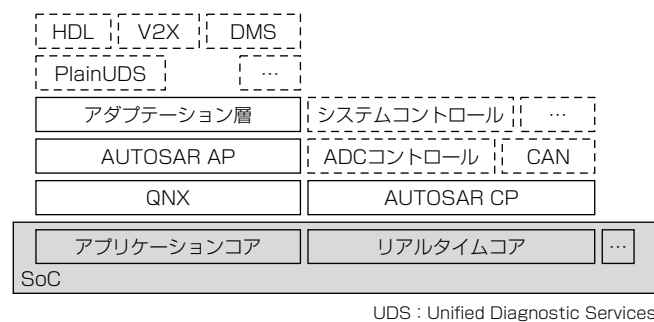


図2-開発システムのソフトウェアブロック図

2.1 開発システムの構成

先に述べた開発システムの構成内容について詳細を述べる。

2.1.1 AUTOSAR CP

AUTOSAR CPは、自動車業界の標準化組織AUTOSARが提唱する、リアルタイム性と安定性を重視した組み込みシステム向けのプラットフォーム(以下“PF”という。)である。SoCのリアルタイムコア側に搭載することで、ハードウェアの詳細を抽象化し、ソフトウェアの再利用性と交換可能性を向上させることが可能である。

2.1.2 AUTOSAR AP

AUTOSAR APは、AUTOSARが提唱する、より高度な機能や複雑なアプリケーションを実現するための演算処理に長(た)けた新しいPFである。SoCのアプリケーションコア側に搭載することで、ソフトウェアの再利用性と交換可能性、開発効率と品質の向上を目的に採用した。またAdaptive Application間にサービス通信を用いることで、アプリケーションは通信相手を意識しない開発が可能になり、APの搭載有無の切替えも容易である。

2.1.3 QNX

QNXは、カナダのQNX Software Systemsが開発したリアルタイムOSである。マイクロカーネルアーキテクチャを採用しており、その結果、高い安定性とリアルタイム性能を実現している。信頼性と即時性があり、かつ新技術の搭載も早いQNXをアプリケーションコアに搭載した。QNXは特定のPFに依存することなく様々なPFに流用可能なため、異なるECU開発向けにも相性が良い。

2.2 単一ストレージシステム

車載ECUは、制御プログラム、設定情報、ログなどのデータの保存や更新のためにマイコン内部又は外部にストレージを搭載している。従来システムでは、アクセシビリティを確保するためにSoC及びマイコンそれぞれにストレージを準

備し、マイコン向けストレージに格納した起動プログラム(ブートローダー)から起動させ、その後、SoC向けのストレージに格納したブートローダーを動作させることによって起動処理を実現していた。

開発システムでは、一つのストレージにブートローダーを含む全てのデータを格納し、単一のブートローダーから起動する仕組みを実現した。これによって、従来システムでの、マイコン及びその周辺回路の部品点数や基板面積の削減を可能にした(表1)。ただし、従来システムのマイコンで実現していた電源電圧監視及び温度監視の機能を実現するために、外付けのADC ICやサーミスターを追加搭載している。

表1-構成部品の比較

	従来システム	開発システム
ストレージ	2点	1点
メモリー	2点	1点
別体マイコン	✓	
X' tal, LDO, 周辺部品	✓	
ADC IC		✓
サーミスター		✓

X' tal : 水晶振動子, LDO : Low Drop Out

2.3 機能間リソース共有(又は単一メモリーシステム)

開発システムでは、3.1節で述べたストレージ以外に、メモリー、タイマー、センサーについても、SoCの高性能ハードウェアリソースをコア間で共有することで、従来のシステムと同様の機能を実現しつつ、高性能化を実現した。

表1に示すように、従来システムでは、各マイコンに専用のメモリーを搭載しており、他のマイコンの影響を受けずに機能実現が可能であった。しかし、開発システムのように複数のコアから同一メモリーにアクセスする場合、メモリーアクセス違反が問題になる。そこで、共有メモリーにはSoCのメモリー保護機能、及びOSのメモリーアクセス保護機能を利用することでメモリー分離を実現した。これによって安全にリソースを共有しながら、部品点数の削減が可能になる。

2.4 コア間通信機能

従来システムでは、SoCとマイコンの間をUART等の物理線で接続して制御していたが、開発システムでは共有メモリーと割り込みレジスターを用いたコア間通信機能を実現した。また、これまでのUARTによる通信は、半二重プロトコルを採用していたことで通信効率に課題があったが、開発システムでは、このプロトコルを全二重プロトコルに修正し、高速通信を実現した。また、マイコンが分離した従来システムでは、動作時刻の厳密な同期が難しく、故障発生時にはそれぞれのマイコンの動作ログを突き合わせて解析が必要であった。開発システムでは、同一Clockを各コアで参照することで、時刻同期が容易になる。さらに共有メモリーを用いてアプリケーションコア側からリアルタイムコア側の情報を取得することが可能なため、ログ出力機構をアプリケーションコアに集約することも容易である。

2.5 システム制御機能

従来システムから、CAN通信等から車両情報を最初に受信するマイコンが、システムの起動終了制御や電源管理を実施している。システム制御のために、2.4節に述べたマイコン間通信が確立する前、又は通信終了した後にも、複数のGPIO(General Purpose Input/Output)を使ってマイコン間や、複合電源IC間の制御を実施していた。

開発システムでは、リアルタイムコアがシステム制御を実施する。管理対象のGPIOは削減され、SoCの起動方法に準拠することで起動終了シーケンスは簡素化される。さらに、エラーの発生源やシステムの状態遷移が減少することでシステムの信頼性が向上し、開発や検証にかかるコストが削減されることも期待される。

またソフトウェア更新でも、従来のシステムでは、複数CPUのソフトウェアを順に更新するため、複雑な状態管理と更新時間の長時間化が課題であった。さらにマイコンはストレージ容量が小規模であったため、ソフトウェアイメージを二面持つことが困難な場合も多い。開発システムでは、二面化によるソフトウェア更新の簡素化が可能になった。

従来のシステムでは、システムの起動終了や、ヘルスマニター等の制御をUARTによるマイコン間通信に加えて、ハードワイヤーによるON/OFF制御(GPIO制御)によって実現していた。開発システムでは、これらの処理を共有メモ

リーや、CPUのレジスタ処理によって置き換えて、GPIOによる制御を削減し、システム制御の簡素化を実現した。これによって、従来システムで問題になっていた、GPIO線の不足によるシステムの複雑化の防止が可能になる。

また、ソフトウェア更新機能でも、開発システムによって簡素化を実現している。従来システムでは、次の課題があった。

- (1) SoC搭載のストレージのデータを更新し、その後、マイコン搭載ストレージのデータの更新を行う、逐次処理が必要である。
- (2) SoC及びマイコンのどちらかの更新が失敗すると、双方の一貫性保持のため、他方のロールバックが必要である。
- (3) マイコンに搭載されているストレージ容量が小さいためにデータの二面化ができず、複雑な制御が必要である。

開発システムでは、大容量ストレージとなるSoC向けストレージに一本化することで、従来システムでの課題を解決し、ソフトウェア更新機能の簡素化と更新時間45%短縮(更新速度(Mbps)の180%高速化)を実現した。

3. 評価

2章で述べたマイコンの削除によるシステムの変化点について検証を行った。

3.1 起動性能

マイコンを削除し、一つのSoCにマルチコア、マルチOSの大規模システムを実現する上で課題になるのが起動時間である。システムが大規模になるほど、ソフトウェアイメージは大きくなり、イメージを不揮発性メモリーから読み出し、実行する時間は長くなる。車載ECUの高速起動は必須であり、ADAS系ECUでは電源投入後200msほどでCAN通信応答を返すことなどが要求される。

従来システムでは、マイコンが内蔵フラッシュメモリーから必要最小限のデータをロードして、起動時の高速応答性能を実現していた。開発システムでは、ストレージを一本化しているが、従来どおり、リアルタイムコア向けのプログラムと、アプリケーションコア向けプログラムを分けて管理し、起動時の高速応答性が必要なプログラムだけを先にロードすることで、高速応答性を実現している。図3に示すように、通信応答に必要な最小限のプログラムを先に起動し、残りのプログラムを並行して起動することで、従来の起動の高速応答性能を維持した。また機能維持の観点でも、ストレージへのアクセスタイミングや領域を分割することで、従来システム同様に、システム異常が発生した場合に単一コアの処理でエラーログを保存することなども可能になった。

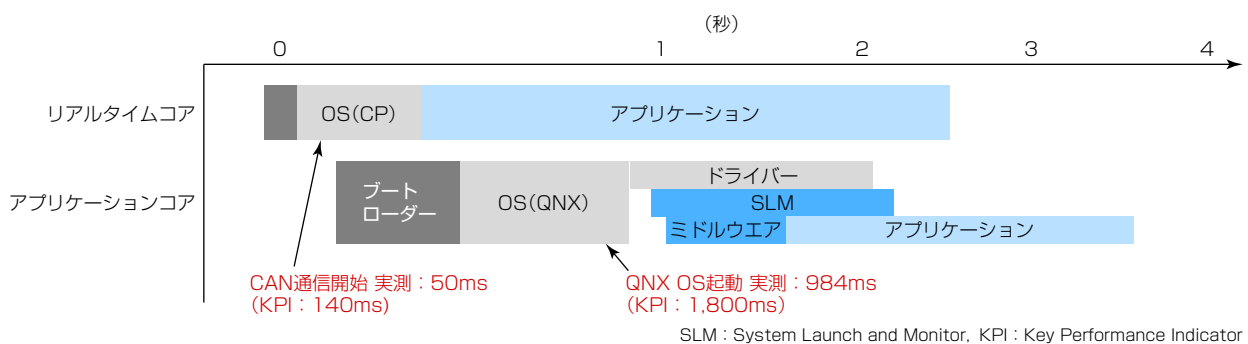


図3-開発システムの起動シーケンス図

3.2 コア間通信性能

従来システムでは、物理線を介した通信であったため、ノイズや干渉の影響を比較的受けやすく、1回の通信転送サイズを大きくすることが困難であった。一方、開発システムでは、物理線を介さず共有メモリーを使用したコア間通信が可能になり、転送サイズの増加とともに数十倍以上の通信速度が実現可能になった(図4)。

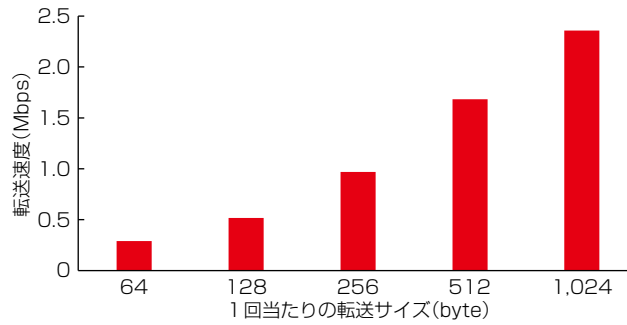


図4-コア間通信スループット

4. む す び

複数のマイコンを搭載するシステムは、基板上の部品点数・基板面積が増加するため原価低減が難しく、ハードウェア設計も複雑化し、ソフトウェア設計も複数CPUのPFを開発する非効率さが課題であった。

そこで、ヘテロジニアス型マルチコアのSoCに搭載されるリアルタイムコアに従来のマイコンの機能を搭載し、マルチOSシステムを構築した。従来システムのマイコン機能をSoCに統合し、機能、性能、コストを検証した。これによって、ストレージの一本化、リソース共有、コア間通信、システム制御の簡素化が可能になり、部品コストを30%削減しつつ、従来の機能維持と性能向上を達成した。

