

# AUTOSARに準拠したCAN通信プロトコルと HILSシステムへの適用事例

姫路事業所 開発部 開発第3課  
服部 喜優

## 1. まえがき

近年、自動車業界において、電動化や先進運転支援システム(ADAS: Advanced Driver-Assistance Systems)などの導入により、車載ECU間の連動制御(車内ネットワーク)が増加し、ECU(Electric Control Unit)のソフトウェア開発と評価に多大な工数を要している。この状況下、ECUのソフトウェア開発工数低減と信頼性向上を目的に、欧州自動車メーカーと車載部品メーカーが共同で規格化を推進しているAUTOSAR(AUTomotive Open System Architecture)と称する標準プラットフォームの採用が広がりを見せている。三菱電機(株)姫路製作所においても、電動化ECUの開発にAUTOSARが採用されており、他種のECUも採用検討が進んでいる。

AUTOSARは、一般的なOSと同様のハードウェア資源の管理に加えて、車内ネットワーク通信の規格と通信制約、および通信データのセキュリティ保護が組み込まれている。さらに車内ネットワークの適用通信プロトコルとして、CAN(Control Area Network)、CAN-FD(CAN with Flexible Data rate)、FlexRay、Ethernetが規定されている。

本稿では、車載ECUの試験設備として使用するHILS(Hardware In the Loop Simulation)システムの開発を通して、AUTOSARに準拠したCAN-FD通信プロトコルを使用した車内ネットワーク通信の概要とシステム適用事例を紹介する。

## 2. 車内ネットワーク通信の概要

### 2.1 車内ネットワーク通信に対する要求

自動車制御の根幹を担う車内ネットワークは、以下のよう  
な観点で高い水準が要求される。

- 伝送効率：単位時間あたりどれだけのデータを送信できるか

- データ信頼性：伝送中にデータが欠落しないか
- 機密性：外部からデータを読み取られないか
- ノイズ耐性：電氣的ノイズに耐性があるか
- 開発工数：プロトコルが複雑すぎないか
- 実装コスト：実装部品や配線は少ないか

さらに、ECU間の通信データの増加に伴い、通信タイミングやデータ構成の制御を行う、ネットワークレイヤの新たな規格が求められている。

### 2.2 CAN通信とCAN-FD通信

車内ネットワーク規格のひとつであるCANは、1994年に国際規格となり、現在最も普及している車内ネットワークプロトコルである。ライン型構造のマルチ・マスター方式を採用し、シリアルバスシステムにより、少ない配線数で汎用性の高いプロトコルを実現している。

CAN-FDは、CANよりも高速にデータを伝送するために、新たに提唱された規格である。従来、CANのデータ部の最大通信速度は1Mbpsであったのに対し、CAN-FDでは8Mbpsと高速になり、CANの送受信データ単位であるフレームに載せられるデータ長も従来の8バイトから、64バイトに拡張されている。近年の車載ソフトウェアの連動制御の情報量増加から、従来のCANではデータ伝送速度が不十分なケースが増えているが、CAN-FDを導入することでバス負荷を軽減することができる。

## 3. AUTOSAR準拠の車内ネットワーク通信

本項は、AUTOSAR4.2以降に規定されているCAN-FD通信仕様について記載する。

### 3.1 PDUとコンテナ通信

#### (1) PDUとコンテナ通信の概要

AUTOSARではメッセージとCAN通信における最小データ単位であるシグナルの間に、コンテナとPDU(Protocol

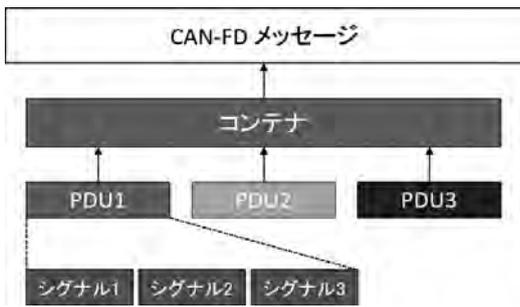


図1. コンテナ通信のレイヤ構造

Data Unit)というレイヤが新たに規定されている。

PDUはそれぞれ送信周期、データ長、内包するシグナルの情報を持っている。図1に示すように、コンテナの内部には複数のPDUが格納でき、コンテナはCAN-FDのメッセージとして送受信される。コンテナ内部の情報は、格納されるPDUによって変化し、シグナルの構成やデータ長が可変となる特徴がある。

図2に示すように、PDUの送信トリガが発生すると、PDUデータは順次コンテナに格納される。同時に複数のPDUの送信トリガが発生した場合、一つのコンテナに複数のPDUが格納される。受信側ノードがPDUを識別できるように、PDUフレームの先頭に識別IDとPDUデータ長を表現するヘッダー情報が付与される。コンテナは、CAN-FDのメッセージとして送信され、受信側ノードはヘッダー情報をもとにメッセージをPDU毎に分解しシグナルを参照する。このような通信形式をコンテナ通信と呼ぶ。

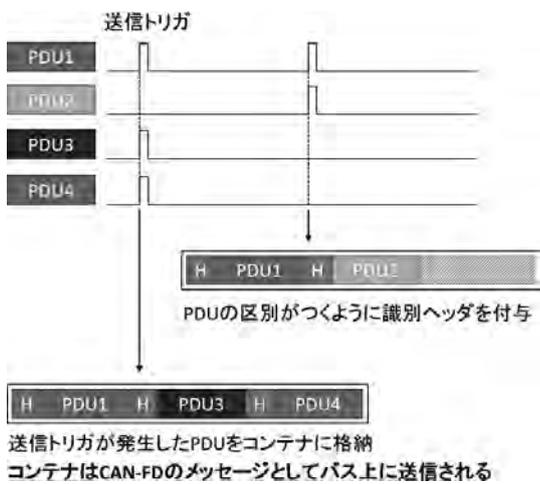


図2. コンテナの生成

(2) PDUとコンテナ通信の利点

図3に示すような、CANで構築されたネットワークとCAN-FDで構築されたネットワークをゲートウェイで連結するシステムを考えた場合、コンテナ通信を採用すると、ゲート

ウェイは従来のCANバス上に流れるメッセージをPDUと認識し、コンテナに格納してCAN-FDバスに送信することができる。ゲートウェイに組み込むソフトウェアは、受信したデータをコンテナに格納すればよい。そのため、CAN-FDのデータベース設計は、従来のCANデータベースをそのまま流用でき、設計・開発が容易になる。

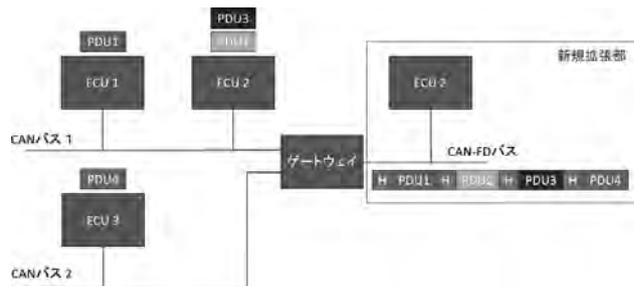


図3. コンテナ通信適用例

3.2 メッセージセキュリティ

(1) セキュリティ強化の背景

自動運転技術やADASなどの複雑なシステムは、センサから読み取った車の状態(車速や目標トルク等)を入力として、ステアリング操作やパワートレイン系の操作を行うために、車内ネットワーク通信を使用してデータを共有することで、ECU間の協調動作を実現している。

そのため、外部から不正なデータを流し込まれた際、自動車の制御に介入され、人命が脅威にさらされる可能性があり、この課題を解決するために、AUTOSARはSecOC (Secured On Board Communication)モジュールを規定している。

(2) SecOCモジュールの仕様

SecOCは外部から流し込まれた不正データを判定するために、共通鍵暗号化方式を採用している。図4に示すように、通信を行うすべてのECUは、PDU毎に対応した共通鍵を保持しており、PDUデータと共通鍵から暗号データであるMAC(Message Authentication Code)を生成する。生成されたMACの一部は、メッセージの末尾に付加され送

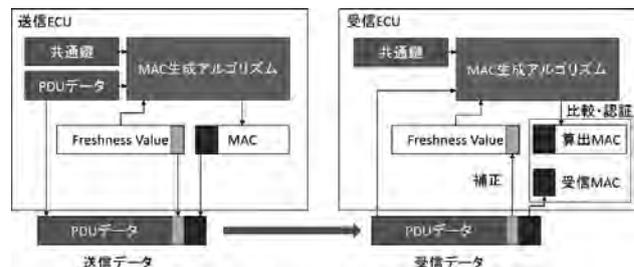


図4. セキュリティ認証手順

信される。受信側ノードは受け取ったMACの一部を、所持している共通鍵で認証し、PDUの信頼性を検証する。

しかし、MAC生成の入力値はPDUデータと共通鍵と仮定した場合、同じメッセージからは同じMACが生成される課題がある。これを利用し、過去に送信されたデータをMACを含めて保持して再送することで、外部からなりすますることが可能である。上記の方法を防ぐために、SecOCはFreshness Valueと呼ばれる、再送攻撃を防ぐカウンタが用意されている。

SecOCは、PDUデータと共通鍵以外にFreshness Valueを入力としてMACを生成する。

ECU間にはMACの検証をするために、Freshness Valueの同期をとる必要があり、Freshness Valueの一部を、PDUに付加して送信する。

#### 4. HILSシステムへの適用事例

車載ECUの試験設備であるHILSシステムは、車両の環境を模擬するために、車内ネットワーク通信を用いて車両情報をECUに送信する。今回、HILSシステムは試験対象である車載ECUに合わせてAUTOSARに準拠したCAN-FD通信を実装する必要があった。しかし、HILSシステムに実装する車内ネットワークモジュールは、試験設備のメーカーやサードパーティが提供するライブラリを使用しているが、AUTOSARに準拠したCAN-FDプロトコルについては、メーカーやサードパーティの開発が追いついて

いないのが現状である。

そのためECUの試験を実施するためには、CAN-FDモジュールを使用して、AUTOSARに準拠したCAN-FDプロトコルを独自に開発する必要があり、HILSシステムにAUTOSARに準拠したCAN-FDプロトコルを適用した事例について以下に記述する。

##### 4.1 コンテナ通信の実装

HILSシステムでコンテナ通信を実施する流れを図5に示す。

###### (1) PDUの生成

図5の破線1で囲んだ部分は、各PDUのデータを生成する箇所である。

HILSシステムは、CAN通信データベースファイルに設定しているシグナル情報と、ユーザが設定した車速や目標トルク等の物理量データから、PDUデータとヘッダーを生成する。その後、生成したデータと共通鍵、HILSシステム内部でカウントしているFreshness ValueをもとにMACを算出し、PDUデータの末尾に付加することで、PDUメッセージを生成する。

###### (2) コンテナメッセージの生成

図5の破線2で囲んだ部分は、コンテナメッセージを生成する箇所である。

前項で生成したPDUメッセージは、トリガが発生したのから順番に連結され、コンテナメッセージとして生成される。コンテナメッセージのサイズは連結したPDUのデータサイ

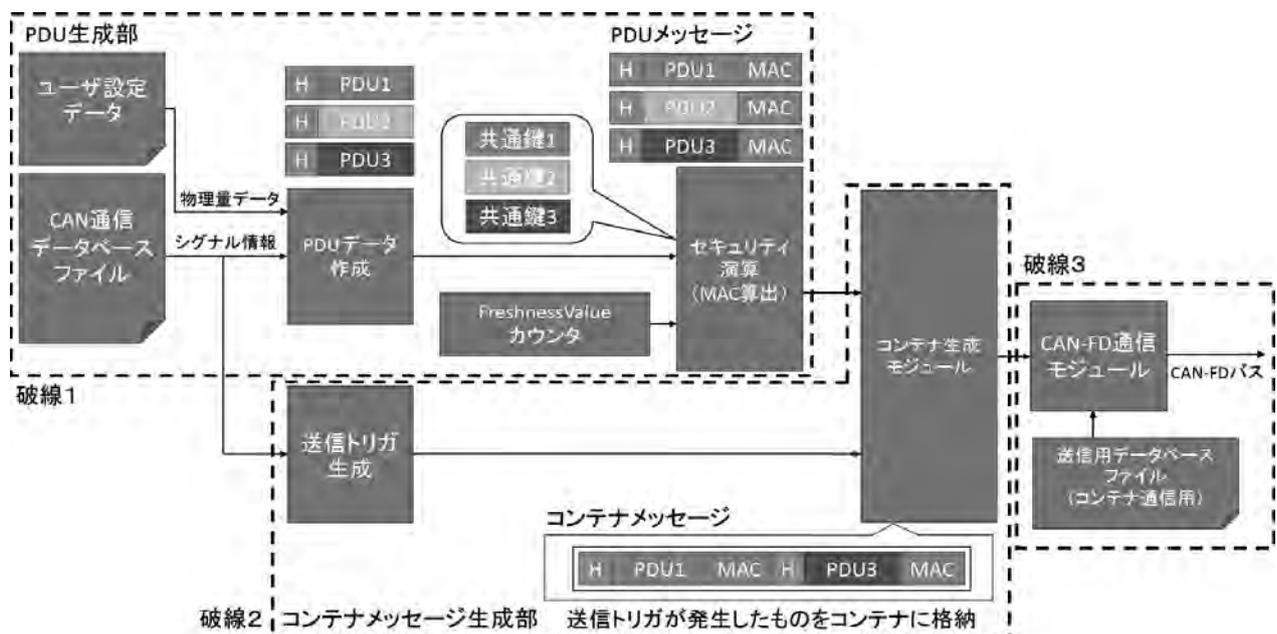


図5. コンテナ通信内部フロー

ズに応じて変化する。

(3) 送信データの形成とCAN-FDバスに送信

図5の破線3で囲んだ部分は、送信データを形成しCAN-FDバスに送信する箇所である。CAN-FD通信モジュールは、送信用データベースファイルをもとに送信データを形成し、CAN-FDバス上に送信する。

図6に示すように、従来のCAN-FD通信は、個々のPDUメッセージをCAN-FDバス上に送信していたため、CAN通信データベースファイルと送信用データベースファイルの内容が同一となり、シグナル情報やデータ長を読み取ることで、送信データを形成することができた。

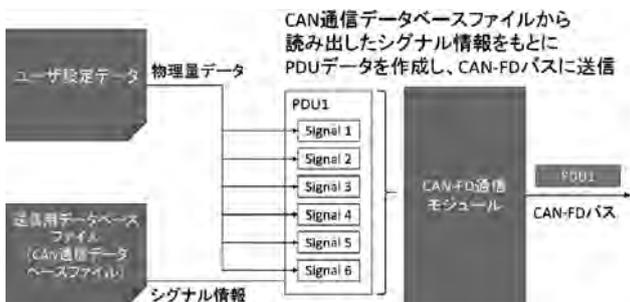


図6. 従来のCAN-FD通信モジュール使用方法

一方、コンテナ通信は、格納するPDUメッセージの数やトリガの発生順序によって、コンテナメッセージ中のシグナル格納位置やデータ長が変化するため、CAN通信データベースファイルから送信データを形成することができない。そこで、コンテナ通信用のデータベースファイルを作成し、CAN-FD通信モジュールに設定した。

コンテナ通信用のデータベースファイルには、1バイト毎に分割したダミーシグナルを定義している。図7に示すように、コンテナ生成モジュールが出力したコンテナメッセージを、バイト単位に分割して代入することにより、CAN-FD通信モジュールはコンテナメッセージのPDU構成に依存せず、分割されたバイトデータを連結して、送信データとして再構成することで、CAN-FDバス上にデータを送信することができる。



図7. コンテナ通信用データベースファイル使用方法

前述の方法は、CAN-FD通信モジュールの内部を変更せず、送信用データベースファイルの情報を書き換えることで対応しているため、コンテナ通信と従来のCAN-FD通信の両方を共存して使用することができる。

4.2 セキュリティ演算の実装

HILSシステムは、ECUとフィードバック制御を実現するために、高速で正確な演算周期が要求される。特に車内通信に関する機能は、送受信スケジュールに忠実に送信するためにもデータ生成、セキュリティ演算が規定時間内に完了することが必要条件となる。

特に今回実装したセキュリティ演算は、入力データを攪拌するため計算量が多く、算出に時間がかかる。そのため演算を規定時間内に完了するために、以下の設計を採用した。

(1) S-Functionブロックの使用

HILSシステムの開発にはモデル言語であるMATLAB/Simulinkを使用しているが、Simulinkを使用して大量の配列データを扱ったり、特定のビット位置に書き込む操作は煩雑な記述になる。

また、高級言語であるSimulinkで作成したアプリケーションをHILSシステムのハードウェアに書き込む際、コンパイラは一度C言語に変換しロードモジュールを生成するため、実行効率が低下する恐れがある。

そこでHILSシステムでは、PDUデータ作成ロジックとセキュリティ演算ロジックを、Simulinkのブロック線図にC言語で記述したロジックを追加することができるS-Functionブロックに記述することで上記の課題を解消し、さらにモデル言語の長所であるソフトウェアの視認性・保守性を損なうことなく、高速な演算性能を確保した。

(2) HILSシステムのマルチプロセッサ化

HILSシステムには、4つのプロセッサが実装されているが、従来のシステムでは、1つのプロセッサですべての処理を行っていた。

今回、セキュリティ演算を規定時間内に完了させるためにマルチプロセッサシステムを採用し、一つのプロセッサに演算機能を独立させた。マルチプロセッサの構成を図8に示す。マルチプロセッサ化でプロセッサ間のデータ受渡し時間による処理遅延に対処するため、演算プロセッサとの動作周期をマスタプロセッサよりも高速に設定し、マスタプロセッサでのデータを生成後、演算プロセッサがデータを受け取るまでの待機時間を短縮した。

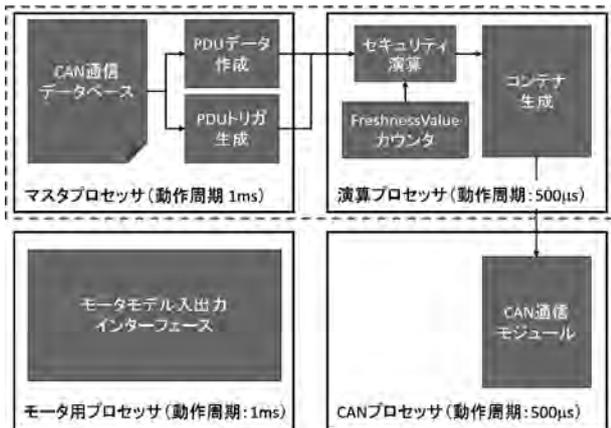


図8. マルチプロセッサ構成図

## 5. 今後の展開

AUTOSARは、業界全体での採用が加速しており、今後ますますECUへの適用が見込まれる。今回、HILSシステムは試験対象であるECUに合わせてAUTOSARに準拠したCAN-FD通信を実装したが、車載Ethernetの車内ネットワークが実装される可能性も大いにある。今後、HILSシステムに、ECUの試験装置として他の車両通信ネットワークを実装していく。

## 6. むすび

AUTOSAR準拠の車内ネットワークを、HILSシステムに適用することができた。今後もECUの試験設備として要求される機能を実装していく。

最後に、本システムの開発に際し御指導、御協力いただいた関係者各位に深く感謝申し上げます。

### 執筆者紹介



服部 喜優      ハットリ ノブマサ  
2016年入社。主にHILSシステムのソフトウェア開発に従事。現在、姫路事業所開発部開発第3課。