

# 軽量スクリプト言語を用いた 自動車ソフトウェア遠隔更新制御方式の開発

寺岡秀敏<sup>†1</sup> 山崎裕紀<sup>†2</sup> 櫻井康平<sup>†2</sup> 尾崎友哉<sup>†1</sup>

**概要:** 近年、自動車分野においても機能追加やセキュリティリスクへの対策等のため、電子制御ユニット (ECU) に搭載されたソフトウェアを販売後に更新する機会が増大している。そのような中で、デジタルテレビや携帯電話等のコンシューマエレクトロニクス分野で実用化されている無線による遠隔更新技術 (Over the air firmware update: OTA) の自動車システムへの適用が検討されている。OTA による遠隔更新では、更新処理の自動化が必要でありそのための更新制御技術が重要となる。本稿では、自動車システム向けの OTA 更新制御についての要件を整理し、その中で自動車システム特有の多様性に対応した更新制御方式を提案する。また、提案手法を車載ゲートウェイ用マイコン上に実装し、提案方式が搭載可能であることを確認した。

**キーワード:** 車載 ECU, OTA, 更新制御軽量スクリプト

## OTA Update Control Method for Vehicle System

HIDETOSHI TERAOKA<sup>†1</sup> HIROKI YAMAZAKI<sup>†2</sup>  
KOHEI SAKURAI<sup>†2</sup> TOMOCHIKA OZAKI<sup>†1</sup>

**Abstract:** Recently, the increase of amount of program code on Electric Control Units (ECUs) in vehicles causes the increase of firmware update after sales which stems from bugs in the program code. In this situation, automakers are studying to introduce over the air firmware update technology which is used in digital TV and mobile phone. In this paper, we propose an update control method for OTA. We implement the method on microcontroller for in-vehicle gateway and show that the proposed method is applicable to automotive field.

**Keywords:** In-vehicle ECU, OTA, update control, lightweight script

### 1. はじめに

#### 1.1 背景

Advanced Drive Assist System(ADAS)や自動運転の導入により、自動車の価値を差別化する要素として、ソフトウェアの価値が高まっている。そのような中、テスラモーターズ社は、2015年8月には販売済みの車両に対して、ソフトウェア更新によって有料で自動運転機能を追加するという、新たなビジネスモデルを試行している[1]。

現在、自動車が販売された後に車載 ECU のプログラム更新を行う場合、顧客 (自動車所有者) がディーラに車両を持ち込み、ディーラの整備士が専用ツールを使ってマニュアル作業で 1 台 1 台 ECU 上のプログラムを書き換えることが一般的である。しかし、この方法は、車両の持ち込み等時間がかかることでユーザの利便性が悪い。この問題を解決するため、車載 ECU に対して遠隔ソフトウェア更新を適用し、プログラム更新時のユーザ利便性を向上することが検討され始めている[2]。

携帯電話網等の無線接続を利用してソフトウェア/ファームウェアを端末に配信し、遠隔で更新する技術は OTA

(Over the Air software/firmware update: 無線によるソフトウェアの更新) と略称される。デジタルテレビや携帯電話分野では OTA は一般化され、多量の端末のソフトウェアが効率的に更新されている。OTA システムを自動車に適用する場合、修正した新プログラムを OTA センタに登録し、OTA センタは車両に修正した新プログラムを配信する。車両では、新プログラムを受信し、これを更新対象 ECU に適用する(図 1)。

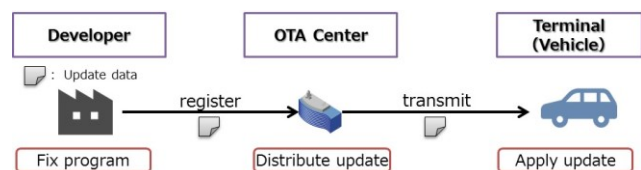


図 1 OTA システム構成

Figure 1 System configuration of an OTA system.

近年、車載機 (カーナビ) においても、遠隔での地図やソフトウェアの更新が導入されている。前述の、テスラモーターズ社の取り組みも、OTA を自社車両に適用することで実現されている。

<sup>†1</sup> 株式会社日立製作所 研究開発グループ  
Hitachi, Ltd. Research & Development Group  
<sup>†2</sup> 日立オートモティブシステムズ株式会社  
Hitachi Automotive Systems, Ltd.

## 1.2 自動車システムの構成

図 2 に OTA によるソフト更新の対象となる自動車システムの構成を示す。図中の ECU の色が異なるのは、HW リソースなどの特徴の異なる ECU であることを示す。

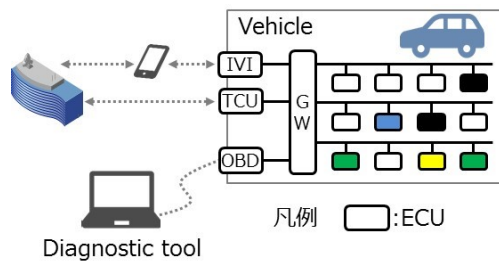


図 2 自動車システムの構成

Figure 2 Configuration of vehicle system.

これまで OTA が適用されてきたコンシューマエレクトロニクス製品と異なり、自動車システムは 1 つ 1 つが携帯端末に相当するような多数のデバイス (ECU) で構成され、これらが連携して機能を実現する複雑なシステムである。自動車 1 台当たりの ECU の搭載数は 2025 年には平均 30.4 個になると予想されている [3]。

従来の ECU に加えて、カーナビや Audio 等の機能を備える In-vehicle infotainment (IVI) や車両外部との通信機能を備える Telematics Control Unit (TCU) が搭載され、Telematics center と通信することでさまざまなサービスが提供されている。

また、車両には整備士が専用ツール (Diagnostic tool) を接続して故障診断などを行うための On-board diagnostic (OBD) コネクタが備えられ、現在、ECU のソフトウェア更新は本コネクタ経由で Diagnostic tool から実行される。

近年、これら車両外部とのインタフェースからのサイバー攻撃の事例 [4][5] が報告されており、その対策として IVI, TCU や外部接続機器および OBD のような外部インタフェースとの間で Firewall として機能するセキュリティゲートウェイの搭載が進みつつある。

## 1.3 自動車システムのソフトウェア更新における特徴

ソフトウェア更新の対象となる ECU の組み合わせは、電気自動車やガソリン自動車などの車種によっても異なる。このため、システム構成として 1 デバイス-複数ソフトウェアの場合が多いコンシューマエレクトロニクス分野の製品と比較すると、ソフトウェア更新対象は非常に多様である。

図 3 に示す通り、自動車システムのソフトウェア更新は静的・動的両面で多様である。すなわち、ガソリン車、電気自動車といった車種により ECU の構成が異なる。さらに、同じガソリン車でもモデルにより ECU 構成が異なる (静的多様性)。例えば、車種による違いに伴い、ガソリン車の更新はイグニッション OFF で、電気自動車の更新は充電中に、など更新条件が異なる可能性がある。また、同じ車種でも構成要素となる ECU ごとに、それぞれ更新条件

や更新手順が異なる可能性がある。他にも、同じ車両でも、1 回目の更新と 2 回目の更新では更新対象となる ECU が異なる可能性もある (動的多様性)。例えば、1 回目の更新では単一の ECU を更新するが、2 回目の更新では複数の ECU を更新する必要がある、などのバリエーションが考えられる。

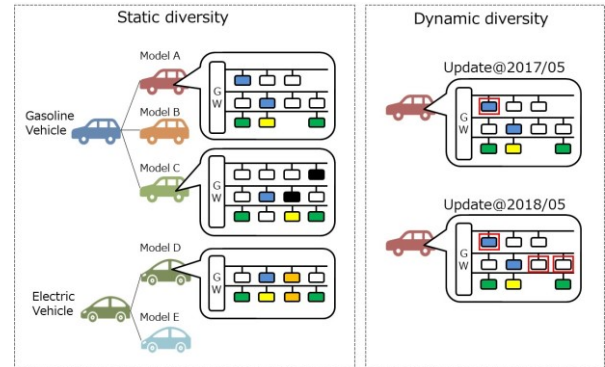


図 3 自動車システムの多様性。

Figure 3 Overview of diversity of vehicle configuration.

## 1.4 課題

自動車システムに OTA を適用する場合、従来ディーラーにて整備士がマニュアルで実施していた更新手続きを車載更新制御装置が自動で実行する必要がある。

本研究では、自動車システムへの OTA 適用を実現する更新制御方式を提案する。特に、自動車システム特有といえる前述の多様性に着目し、多様性への対応が可能な制御方式を提案する。

## 2. 自動車システム向け OTA 更新制御機能

図 4 に、本研究における自動車向け OTA ソフトウェア更新システムの全体像を示す。本研究では、更新制御機能は車載ゲートウェイに配置し、そこで OTA センタから取得するパッケージに含まれる更新スクリプトに基づいて制御を実行する構成を提案する。更新スクリプトは、ISO 標準である Open Test sequence eXchange (OTX) (ISO13209) [6][7] および Open Diagnostic data eXchange (ODX) (ISO22901) [8] で記述されたシーケンスおよびパラメータをゲートウェイ上で実行可能な軽量な形式に変換されたものを用いる。ここで、OTX/ODX ファイルは自動車メーカーや ECU の部品メーカーにより作成され、OTA センタで軽量スクリプトに変換されて新プログラムそのものや差分データとともにパッケージに同梱される構成を想定する。スクリプトの作成が自動車メーカーと部品メーカーで行われるのは、自動車全体や複数の ECU にまたがる部分は自動車メーカー、ECU 固有の部分は部品メーカーが担当することを想定するためである。

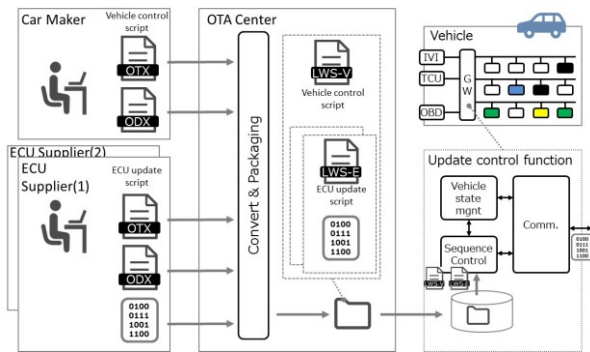


図 4 自動車向け OTA ソフトウェア更新システム全体像.

Figure 4 Overview of OTA software update system for automotive.

本章では最初に更新制御機能の要件を整理し、次に、要件に基づく機能配置についての検討結果を述べる。その後、自動車システム特有の課題といえる多様性に対応するための更新シーケンス制御の柔軟性確保の方法についての検討結果を述べる。

## 2.1 OTA 更新制御機能の要件

表 1 に更新制御機能への要件を示す。

表 1 更新制御機能の要件

Table 1 Requirements of OTA update control function.

Req#	内容
1	更新シーケンス制御が可能であること。
2	車両状態の把握が可能であること。
3	外部からの更新データの取得が可能であること
4	セキュリティ機能を備えること

### (1)更新シーケンス制御

更新制御機能は、車両状態等に基づいてソフトウェア更新のプロセスの開始・中断や、Controller area network (CAN) などの車内ネットワークを介した ECU への更新開始指示、更新データ転送および ROM 書き換えなどの制御を行う必要がある。1.3 に示したような多様性を持つ自動車システムに対して OTA を適用する場合、更新制御機能がガソリン車や電気自動車などの車種に起因する手順の相違や、ECU の違いに起因する手順の相違に柔軟に対応可能であることが特に重要となる。本課題については、2.3 にて詳細要件の整理を行う。

### (2)車両状態の把握

更新制御機能は、前術のシーケンス制御を行うため、また、車両状態を OTA センタと同期するため、車両状態を把握できる必要がある。ここでの車両状態には、イグニッション OFF/ON、電源状態、走行状態、ユーザとのインタラクション結果、有線/OTA でのソフトウェア更新状態などが含まれる。

### (3)更新データの取得

更新制御機能は、更新対象 ECU に適用する新プログラ

ムや付加情報など更新用のデータを外部から取得する必要がある。また、必要に応じて取得したデータを蓄積する必要がある。

### (4)セキュリティ機能

更新制御機能は、OTA センタとの通信等のセキュリティを確保し、取得した更新データの妥当性を保証する必要がある。

## 2.2 OTA 更新制御機能の配置

本研究では、前述の要件を満たす更新制御機能の配置先はゲートウェイであることを前提とする。表 2 に、機能配置に関する比較結果を示す。比較の観点と比較観点とした理由は以下の通りである。

### (A).ECU へのアクセス性

状態把握および更新制御のため、車両内の ECU へのアクセス性が容易であることが重要となるため。

### (B).セキュリティ

更新制御に当たっては、セキュリティ確保が重要であるため。

### (C).OTA センタへのアクセス性

OTA センタからの更新データの取得、車両状態の OTA センタとの同期のためには OTA センタへのアクセス性が容易であること重要となるため。

### (D).ハードウェアリソース (CPU、メモリ)

更新制御はソフトウェアで実現することを想定した場合、更新制御機能が搭載可能なリソースを持つことが重要であるため。

表 2 更新制御機能の配置先比較

Table 2 Comparison of location of OTA update control function.

配置先	(A)	(B)	(C)	(D)
OTA Center	×	×	◎	◎
IVI	△	△	○	◎
TCU	△	△	○	○
Gateway	◎	○	△	△
ECU	○	◎	×	×

更新制御機能を OTA センタに配置する場合、OTA センタ-自動車間の通信が保証されていない場合 ECU へのアクセスは悪い。また、セキュリティに関しても、OTA センタと ECU 間の通信を保護するためにセキュリティ保護がなされていない場合、セキュリティリスクが高まる危険がある。一方で、更新データは OTA センタで生成・管理されるためデータ取得は容易である。また、組み込み端末であるそのほかの配置先と比較した場合、リソースには余裕があると言える。

更新制御機能を IVI に配置する場合、同一の車両内に存在する ECU へのアクセスは OTA センタと比較して良好といえる。一方で外部ネットワーク接続される上に、多様なアプリケーションが搭載されて Personal computer 相当の端

末となりつつある IVI から ECU へのアクセスを容易にする場合、セキュリティリスクが高まる懸念がある。IVI は、携帯電話を用いたテザリング等で外部ネットワークに接続されている場合が多いため、データ取得は容易である。また、そのほかの配置先と比較した場合、OTA センタほどではないが、リソースには余裕があると言える。

更新制御機能を TCU に配置する場合、同一の車両内に存在する ECU へのアクセスは OTA センタと比較して良好といえる。一方で外部ネットワーク接続される TCU から ECU へのアクセスを容易にする場合、セキュリティリスクが高まる懸念がある。TCU は、常に外部ネットワークに接続されているため、データ取得は容易である。また、近年の LTE 等高速回線に接続する機能をもつ場合、そのほかの配置先と比較した場合、IVI ほどではないが、リソースには余裕がある場合が多い。

更新制御機能をゲートウェイに配置する場合、同一の車両内に存在し、診断通信等のためにほぼすべての ECU への通信が必要なために ECU へのアクセスは非常に良好といえる。また、元々の外部からのセキュリティ保護のためのコンポーネントとしての位置づけから、セキュリティリスクも少ないと考えられる。一方で、外部ネットワークとの接続は TCU や IVI を介して行う必要があるため、データ取得は少し困難となる。また、現在の車載ゲートウェイ用マイコンのリソースは TCU ほど潤沢ではない場合が多い。

更新制御機能を ECU に配置する場合、同一の車両内に存在するため ECU へのアクセスは良好であると言えるが、他の ECU へのアクセスはゲートウェイを経由する必要がある場合がある。ゲートウェイで保護された領域に配置することになるため、セキュリティ保護は良好であると言える。外部ネットワークへの接続はゲートウェイおよび TCU または IVI を経由する必要があるため OTA センタへの接続性は悪い。また、大半の ECU では、本来の制御機能以外を動作させるためのリソースの余裕がない場合が多いと考えられる。

以上の比較結果から、配置先として優位であるのは IVI またはゲートウェイであると考えられる。しかしながら、IVI が搭載されない車両がある点、ソフト更新に当たってはセキュリティが重要である点を考慮し、ゲートウェイを更新制御機能の配置先とした。

なお、ここでは、OTA 更新制御専用のデバイスを搭載すること、更新対象の ECU それぞれに更新制御機能を搭載することはそれぞれコストアップにつながることから検討の対象外とした。

## 2.3 OTA 更新制御の柔軟性確保

1.3 に示したような多様な環境を考慮した場合は、要件(1)の更新シーケンス制御は、さらに、以下のような要件を満たすことにより制御の柔軟性を確保する必要がある。

表 3 OTA 更新のシーケンス制御における要件

Table 3 Requirements of OTA update sequence control .

Req#	内容
1-1	処理実行有無の制御
1-2	状態判定閾値の変更
1-3	処理順序の入れ替え
1-4	新規処理の追加
1-5	省リソース

### (2-1) 処理実行有無の制御

各シーケンスにおいて、ECU の異常状態を示す Diagnostic trouble code(DTC)の確認をする／しない、ダウンロードしたソフトウェアの適用可否についてユーザの許諾確認をする／しないなどの処理実行有無の制御が必要となる。

### (2-2) 状態判定閾値の変更

ソフト更新可否判断のための電池残量などの判定条件の変更等が可能である必要がある。

### (2-3) 処理順序の入れ替え

シーケンス内の処理順序の変更が可能である必要がある。

### (2-4) 新規処理の追加

状態確認対象 ECU の追加やその手順の追加が可能である必要がある。

(例：ガソリン車：イグニッション状態の確認。

電気自動車：イグニッション状態の確認＋充電状態の確認)

### (2-5) 省リソース

車載ゲートウェイに搭載するため、省リソースで動作する必要がある。

前述の要件を満たすために、それぞれの車種・モデルごとに更新制御ソフトウェアをカスタマイズすることは開発コストが上昇してしまうという問題がある。

このような問題に対応するための方法として、以下の 2 通りの方法が考えられる。

#### (a) パラメータによる制御方式

OTA センタからダウンロードするパッケージに、更新制御に関するパラメータを設定し、更新制御ソフトウェアはパラメータに基づいて制御を行う。

#### (b) スクリプトによる制御方式

OTA センタからダウンロードするパッケージに、シーケンスを記述したスクリプトを同梱し、更新制御ソフトウェアは当該スクリプトを実行して制御を行う。

パラメータ方式は、リソースの消費が少ないが、順番の入れ替えや新規処理の追加など要件を満たせない場合がある。また、同一車両において更新キャンペーンごとに異なる可能性のあるパターンも網羅してパラメータ設計を行うことは困難である。

一方、スクリプトを用いる方式は、要件#1-1～1-4 を満た



すが、小リソースな機器上での実行は困難な可能性がある。  
そこで、本研究では要件の満足性が高いと考えられるスクリプト方式を小リソースな機器上で実行可能な方式を検討する。

## 2.4 ISO 標準技術によるスクリプト記述

整備士が、自動車システムの診断や、ECU のソフトウェアを更新する際に用いる専用ツールやツールと ECU の通信インタフェース向けの規格として UDS (Unified Diagnostic Service) (ISO14229)[9], ODX, OTX が標準化されている。

UDS は診断装置と ECU 間の通信規格を定めている。具体的には、CAN 等で接続した ECU との間で送受信するデータを定義するものであり、診断処理 (サービスと呼ばれる) 毎にバイナリフォーマットが規定されている。

ODX では前述の UDS で規定された診断サービスで利用するデータの定義を行う記述方式が規定されている。具体的には、ECU から情報を取得する処理やリセットを行う処理、あるいは ECU のソフトウェアを更新する処理で扱うデータの内容を XML で記述する。

OTX は、診断サービスの処理シーケンスを記述するための標準技術である。具体的には、後述の UDS で規定され、ODX でデータが記述された診断サービスを、どの順番で実行するか、ECU と通信した結果によって条件分岐する、というような処理の流れを XML で記述する。

図 5 に、これら ISO 標準技術の概要を示す。

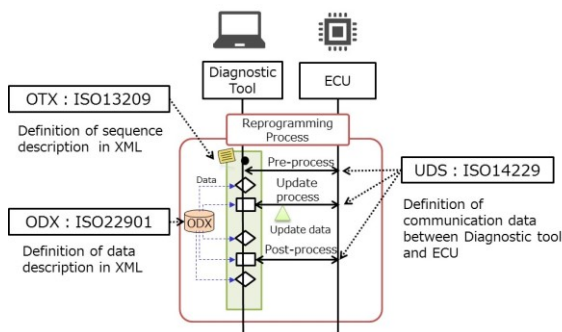


図 5 ISO 標準の概要.

Figure 5 Overview of ISO standards.

このような、標準化された記法で手順を記述することにより自動車メーカー同士や自動車メーカーと部品メーカー間で更新制御手続きの共有が容易になる。また、記述に当たって市販ツールを利用することも可能である。

一方で、これら標準は XML での記述を前提としているため、リソースの少ない車載ゲートウェイ上で解析・実行することは困難である。そこで、本研究では ISO 標準で作成されたスクリプトを、OTA センタで軽量スクリプトに変換して車載ゲートウェイ上で実行する方式を提案する。

## 2.5 軽量スクリプト言語を用いた車載ゲートウェイアーキテクチャ

図 6 に、車載ゲートウェイ上の更新制御ソフトウェアのアーキテクチャ概略を示す。

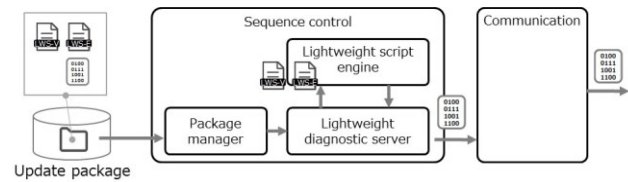


図 6 更新制御ソフトウェアアーキテクチャ概略.

Figure 6 Overview of update control software architecture.

Package manager は、更新パッケージを解析し、スクリプトや ECU への送信データを分離する。

Lightweight diagnostic server は、更新パッケージに含まれるスクリプトを読み出し、Lightweight script engine を呼び出す。また、軽量スクリプトに ECU への命令を抽象化した API を提供し、軽量スクリプトからの呼び出しに応じて UDS コマンドの生成や受信した応答の解釈を行い、結果をスクリプトに伝える。さらに、更新パッケージに含まれる ECU への送信データを取得し、軽量スクリプトからの指示に従って ECU に送信する。

Lightweight script engine は軽量スクリプトの実行環境であり、OTA センタから受信したスクリプトを実行する。本研究では、組み込みスクリプト言語の中でも軽量の Lua[10] を更新制御スクリプトとして利用する。

## 3. 評価

### 3.1 評価環境

提案方式の車載システムへの搭載性を評価するため、車載ゲートウェイ向けのマイコンに前述のアーキテクチャのソフトウェアを実装し、メモリ使用量を評価した。図 7 に評価環境の外観を示す。

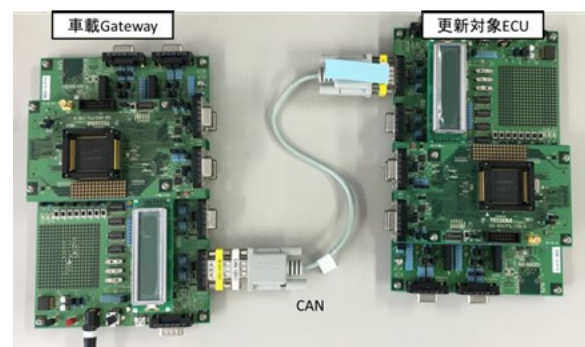


図 7 評価環境外観

Figure 7 Appearance of evaluation platform

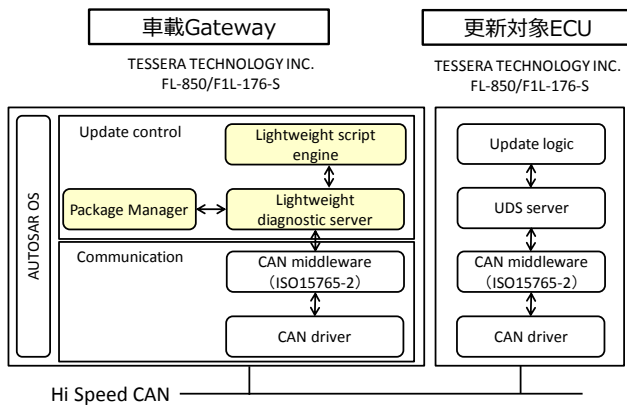


図 8 評価環境機能配置

Figure 8 Functional layout of evaluation platform.

表 4 評価環境

Table 4 Specification of evaluation platform.

項目	値と説明		
CPU	80MHz	Renesas RH850/F1L	
ROM	2MB		
RAM	160KB		
CAN	帯域	500Kbps	HI-Speed CAN
	プロトコル	ISO14229(UDS), ISO15765-2[11]	

本研究では、テセラ・テクノロジー社製のマイコン評価ボード FL-850/F1L-176-S[12]にルネサスエレクトロニクス社製の RH850/F1L[13]を搭載し評価環境を構成した。本評価では、車載ゲートウェイ及び更新対象 ECU に同じ評価ボードを用いそれぞれに異なるソフトウェアを実装した。評価用車載ゲートウェイ及び更新対象 ECU の機能配置を図 8 に示す。車載ゲートウェイにはスクリプトで規定された手順に従い更新シーケンスの制御を行う更新シーケンス制御部、ISO15765-2 に対応した CAN 通信を行う CAN ミドル及び CAN ドライバを搭載する。更新制御部は、提案の通り Package manager, Lightweight diagnostic server および Lightweight script engine で構成する。また、更新対象 ECU には、同様の通信ミドルウェアおよび車載ゲートウェイからの指示に従って更新を実行する機能を搭載する。CAN の通信帯域は 500Kbps を利用した。

評価には図 9 に示す簡易的な更新シーケンスを用いた。本評価の簡易シーケンスでは、ECU に対するプログラム書き換えモードへの変更、更新プログラムの転送と ROM 書き込み、書き込み後の新プログラムの検証を行う。

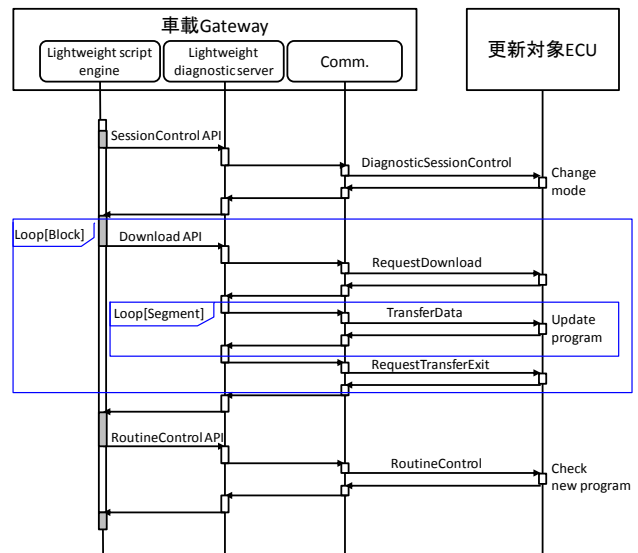


図 9 評価用更新シーケンス

Figure 9 Update sequence for evaluation.

### 3.2 評価結果

評価環境上で前述のシーケンスを実行した際の車載ゲートウェイにおける Lightweight script engine の RAM および ROM の使用量を表 5 に示す。

表 5 提案方式のメモリ使用量[byte]

Table 5 Memory consumption of proposed method [byte].

Item	Value
RAM	46,898
プログラム(ROM)	66,039

本評価により、メモリ使用量の観点からは提案方式が車載ゲートウェイ用マイコンに搭載可能であることが確認できた。

### 4. 関連研究

OTA によるソフトウェア更新制御に関して、Boer らは ECU ごとの更新手順の相違への対応という課題に対し、Head Unit に OSGi フレームワーク[14]を搭載して更新制御機能をバンドル (Java のアプリケーションプログラム) として配信する方式を提案している[15]。しかしながら、OSGi フレームワークを小メモリな車載ゲートウェイに搭載することは困難であると考えられるため本研究では、Lua スクリプトを用いる方式を提案した。

### 5. おわりに

本研究では、自動車システムにおける OTA によるソフトウェアの遠隔更新向けの更新制御方式を提案した。提案に当たり、更新制御機能への要件を整理し、当該要件に基づいて更新制御機能の配置先を提案した。さらに、従来 OTA が適用されてきたコンシューマシステムと異なる自動車システムの多様性という課題に対して、スクリプトを用いて

更新シーケンスの制御を行う方式を提案し、車載ゲートウェイ向けのマイコンにこれを実装してメモリ消費量の観点からその搭載性を確認した。

今後は、更新の高信頼化やセキュリティなど、自動車システムへの OTA 適用課題についての検討を行う。

## 参考文献

- [1][https://www.teslamotors.com/sites/default/files/pdfs/release\\_notes/tesla\\_model\\_s\\_software\\_7\\_1.pdf](https://www.teslamotors.com/sites/default/files/pdfs/release_notes/tesla_model_s_software_7_1.pdf)(2016-04-16 参照)
- [2]寺岡秀敏; 中原章晴; 黒澤憲一. 車載 ECU 向け差分更新方式. 情報処理学会論文誌コンシューマ・デバイス & システム (CDS), 2017, 7.2: 41-50.
- [3]富士キメラ総研ニュースリリース [http://www.group.fuji-keizai.co.jp/press/pdf/170517\\_17042.pdf](http://www.group.fuji-keizai.co.jp/press/pdf/170517_17042.pdf) 2017/09/04 参照
- [4]Miller, C., & Valasek, C. (2015). Remote exploitation of an unaltered passenger vehicle. Black Hat USA, 2015.
- [5]Foster, I. D., Prudhomme, A., Koscher, K., & Savage, S. (2015, August). Fast and Vulnerable: A Story of Telematic Failures. In WOOT.
- [6]ISO13209-2 Road vehicles – Open Test sequence eXchange format(OTX) – Part2:Core data model specification and requirements
- [7]ISO13209-3 Road vehicles – Open Test sequence eXchange format(OTX) – Part3:Standard extensions and requirements
- [8]ISO22901-1 Road vehicles – Open Diagnostic Data Exchange(ODX) – Part1: Data model specification
- [9]ISO14229-1 Road vehicles -- Unified diagnostic services (UDS) – Part1: Specification and requirements
- [10]<https://www.lua.org/> 2017/09/04 参照
- [11]ISO 15765-2:2011 Road vehicles — Diagnostic communication over Controller Area Network (DoCAN) — Part 2: Transport protocol and network layer services
- [12]<http://www.tessera.co.jp/fl/fl1-176.html>(2016-09-19 参照)
- [13]<http://japan.renesas.com/products/mpumcu/rh850/rh850f1x>
- [14]<https://www.osgi.org/> (2017/09/04 参照)
- [15]de Boer, Gerrit, Peter Engel, and Werner Praefcke. "Generic remote software update for vehicle ecus using a telematics device as a gateway." Advanced Microsystems for Automotive Applications 2005. Springer Berlin Heidelberg, 2005. 371-380.

1 CAN は、Bosch 社の登録商標です。