

CHAdeMO - ECHONET Lite 連携ガイドライン

Ver.1.00

改定履歴

- Version1.00 Draft 2022 年 2 月 18 日
- Version1.00 2022 年 5 月 16 日

全体	<ul style="list-style-type: none">・ 充(放)電器を充放電器に統一・ ESV の記号を ECHONET Lite の正式な記載に修正 (例 : SetC)・ ドキュメント名称を正式な名称に修正・ 誤記、記載漏れなど全体的に修正
3.1	<ul style="list-style-type: none">・ 表 1 内の一部、(C)、(E)を加筆
3.7	<ul style="list-style-type: none">・ 放電に関する記載について、記載場所を変更・ シーケンス 2、シーケンス 3 で図 10、図 11 の違いが分かるように説明を追記

目次

第1章 はじめに	1-1
1.1 目的	1-1
1.2 参考文献	1-2
第2章 検討対象のユースケース	2-1
2.1 はじめに	2-1
2.2 ユースケースの事例	2-2
2.2.1 充放電器を指定するケース	2-2
2.2.2 EVを指定するケース	2-2
第3章 基本シーケンス	3-1
3.1 基本的な考え方	3-1
3.2 EVの接続	3-2
3.3 EVの接続検知とサービスへの通知	3-3
3.4 遠隔監視（運転モード設定&電池残容量）	3-4
3.5 遠隔より充放電器を制御（充電開始）	3-6
3.6 遠隔より充放電器を制御（放電開始）	3-7
3.7 遠隔より充放電器を制御（充電電力/電流制御）	3-8
3.8 EVおよび充放電器の状態変化（充電終了、放電終了）	3-10
3.9 EVの離脱	3-11
第4章 EVと充放電器の紐づきの指針	4-12
4.1 基本的な考え方	4-12
4.2 充放電器で紐づける場合1	4-12
4.3 充放電器で紐づける場合2	4-13
4.4 外部端末を活用して紐づける場合	4-15
4.5 EVで紐づける場合	4-16

第1章 はじめに

1. 1 目的

世界規模で電気自動車の普及が進みつつあるなかで、電気自動車は充電器や充放電器を通じて電力システムなどの社会インフラと接続することで、分散型電源としての大きな期待がある。その際の一般的なシステム構成を図1に示す。

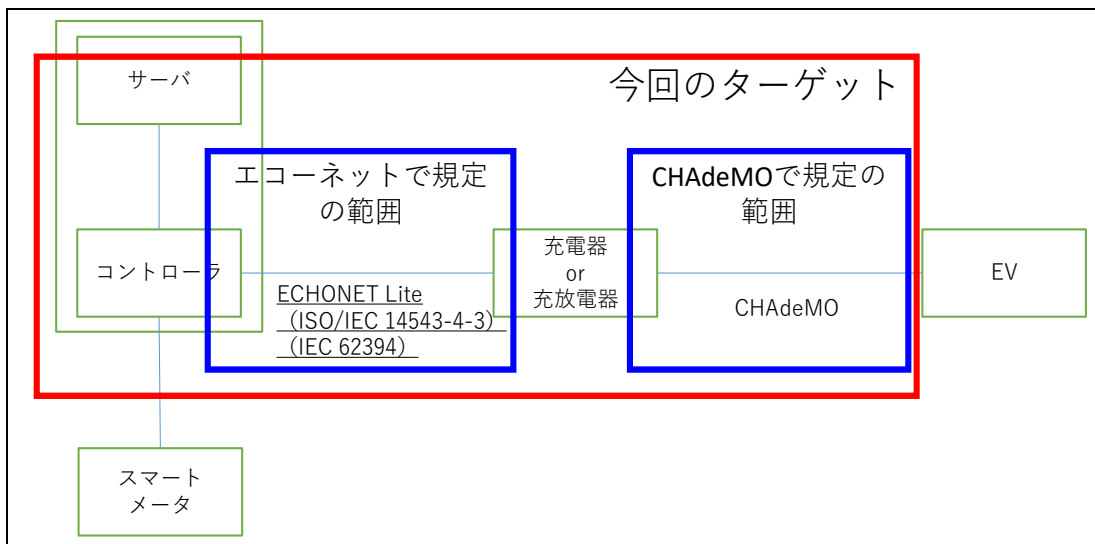


図1 一般的なシステム構成事例

一般的なシステム構成は上記の通りであるが、本ガイドラインにおいては充放電器を対象とする。サーバ上のアプリケーションやコントローラから、充放電器を経由して、電気自動車の充電・放電を制御・遠隔監視しようとする場合、充放電器において、「電気自動車と充放電器間の通信」と「コントローラと充放電器間の通信」を変換することで、コントローラやサーバ上のアプリケーションの要求を電気自動車に伝えることとなる。そのため、充放電器を開発するにあたっては、それぞれの通信仕様を参照したうえで、通信仕様間の変換については各社の暗黙知の上での開発となっている。

本ドキュメントにおいては、電気自動車と充放電器間の通信仕様は CHAdEMO 協議会および電動車両用電力供給システム協議会で策定する規定に従い、コントローラと充放電器間の通信仕様はエコネットコンソーシアムで策定する規定に従うシステムを本ドキュメントの対象とする。本ドキュメントにおいて、CHAdEMO と ECHONET Lite 間の連携機能を形式知とすることで、システム全体の相互接続性の向上を目指す。また、ECHONET Lite 規格と CHAdEMO 規格のつながりを広く国際の場合含めてアピールする。なお、電気自動車については以下「EV」と記載する。

1. 2 参考文献

コントローラと充放電器間の通信仕様に関しては、エコーネットコンソーシアムより公開している以下のドキュメントを参照するものとする。

【コントローラと充放電器間】

- ECHONET Lite 規格書 (一般公開)
- APPENDIX ECHONET 機器オブジェクト詳細規定 (一般公開)
- 電気自動車充放電器／電気自動車充電器・HEMS コントローラ間アプリケーション通信インタフェース仕様書 (一般公開)

また、EV と充放電器間の通信仕様に関しては、CHAdeMO 協議会もしくは電動車両用電力供給システム協議会より公開している以下のドキュメントを参照するものとする。

【EV と充放電器間】

- 電動自動車用充放電システムガイドライン V2H DC 版 (EVPOSSA もしくはCHAdeMO 会員向けに公開)
- CHAdeMO 標準仕様書 (CHAdeMO 会員向けに公開)

第 2 章 検討対象のユースケース

2. 1 はじめに

一般的に、IoT 機器をシステムに接続する場合、以下の機能を組み合わせること
で、システムを構築することが一般的である。

- IoT 機器のネットワークへの参入
- IoT 機器の発見
- IoT 機器の情報取得
- IoT 機器への制御
- IoT 機器の状態変化
- IoT 機器のネットワークからの離脱

しかし、EV を含めたシステムの場合は「機器 (充放電器) + EV」の構成となり、
図 2 に示すように通常の IoT 機器と検討すべき構成が異なる。

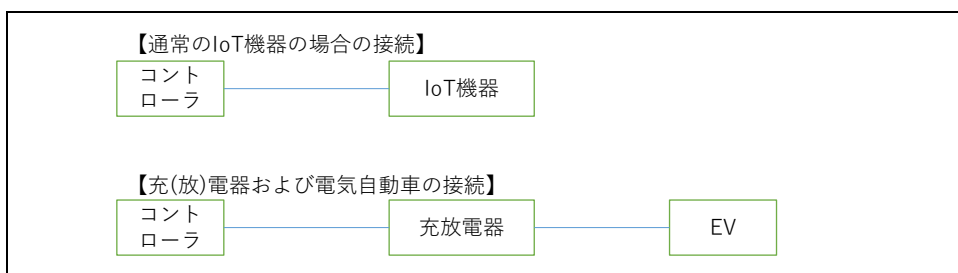


図 2 IoT 機器と EV との場合の構成の違い

コントローラによる IoT 機器の識別と異なる点として、コントローラは充放電器を識別するだけでなく、EV を指定する場合もあることである。なお、本ドキュメントで想定するシステムでは、「EV を指定する」場合は、コントローラが直接の制御対象である充放電器に接続している EV を識別する必要がある。

したがって、このようなシステム構成の違いから、通常の IoT 機器と比べて、EV の接続・離脱、EV と充放電器の組み合わせ確認などの工程が必要になる。

通常のIoT機器	充放電器およびEV
1. 機器のネットワークへの参入	1. EVの接続
2. 機器の発見	2. EVの接続検知
3. 機器の情報取得	3. EVと充放電器の組み合わせの確認
4. 機器への制御	4. EVおよび充放電器の情報取得
5. 機器の状態変化	5. 充放電器への制御
6. 機器のネットワークからの離脱	6. EVおよび充放電器の状態変化 (充電終了、放電終了)
	7. EVの離脱

まず、EV と充放電器の組み合わせ確認が必要なユースケースと、必要無いユースケースとに分けて整理する。

2. 2 ユースケースの事例

2. 2. 1 充放電器を指定するケース

充放電器を指定して情報取得、制御を行う場合については、以下のようなアプリケーションが想定される。

- 一戸建てにおいて、充放電器が 1 台のみ設置されている前提のアプリケーション
- 複数台設置されている場合においても、ピークカットや VPP サービス実行時など、エネルギー視点での充電制御・放電制御が優先され、EV の所有者などを意識する必要が無い場合
- ユーザ認証を行うアプリなどを用いて、充電・放電の制御を行う場合。この場合、アプリは充放電器を指定できれば対応可能で、EV を指定する必要は無い

2. 2. 2 EV を指定するケース

一方、EV を指定して情報取得、制御を行う場合については、以下のようなアプリケーションが想定される。

- シェアリングカー、レンタカー、社用車などで用いられる配車管理システムのように、自動車を指定した操作を行うアプリケーションと連携する場合
- EV の電池の残容量に合わせて、充電の優先順序をつける場合
- 社有車と個人所有の車が混在するような環境において、VPPなどを想定し、EV の所有者の権限などから、放電制御可能な EV と放電不可の EV を区別する必要がある場合

第3章 基本シーケンス

3. 1 基本的な考え方

第2章に示した7種類の工程を組み合わせることで、基本的にはサービスを実現することとなる。したがって、本章においては各工程において、EVと充放電器の組み合わせ確認が必要なユースケースと、必要無いユースケースとに分けてシーケンスを整理する。本章において作成するシーケンスの概要をまとめた一覧を表1に示す。

表1 シーケンス概要

	EV、ユーザともに識別不要のケース (充放電器が分かれば良いケース)	EVを識別するケース (EVを指定して制御するケース)
3.2 EVの接続	<ol style="list-style-type: none"> EVとの接続確認 (C) エコネット側へのネットワーク参入を通知 (E) 	
3.3 EVの接続検知とサービスへの通知	<ol style="list-style-type: none"> EVと充放電器を接続 (C) 充放電器はエコネット側へEVが接続した旨通知 (E) 	<ol style="list-style-type: none"> EVと充放電器を接続 (C) EVと充放電器の紐づけ <ol style="list-style-type: none"> 外部機器 (スマホなど) を用いて紐づけ 充放電器で紐づけ (C->E) EV側で紐づけ 充放電器はエコネット側へEVが接続した旨通知 (E)
3.4 遠隔監視 (運転モード設定&電池残容量)	<ol style="list-style-type: none"> アプリから指定する充放電器に対して、運転モード設定&電池残容量の取得要求送信 (E->C) EVより残容量取得 (C->E) 運転モード設定&電池残容量をアプリへ応答 	<ol style="list-style-type: none"> アプリは情報取得するEVを決定し、EVが接続している充放電器を選択 アプリから指定する充放電器に対して、運転モード設定&電池残容量の取得要求送信 (E->C) EVより残容量取得 (C->E) 運転モード設定&電池残容量をアプリへ応答
3.5 遠隔より充放電器を制御 (充電開始)	<ol style="list-style-type: none"> アプリから指定する充放電器に対して、充電制御実行 (E->C) EVにて充電開始 	<ol style="list-style-type: none"> 制御対象のEVを決定し、EVが接続している充放電器を選択 アプリから該当する充放電器に対して、充電制御実行 (E->C) EVにて充電開始
3.6 遠隔より充放電器を制御 (放電開始)	<ol style="list-style-type: none"> アプリから指定する充放電器に対して、放電制御実行 (E->C) EVにて放電開始 	<ol style="list-style-type: none"> 制御対象のEVを決定し、EVが接続している充放電器を選択 アプリから該当する充放電器に対して、放電制御実行 (E->C) EVにて放電開始

<p>3.7 遠隔より充放電器を制御 (充電電力/電流制御)</p>	<ol style="list-style-type: none"> 1. アプリから指定する充放電器に対して、充電電力 (電流) 制御実行 (E->C) 2. EVにて充電開始 	<ol style="list-style-type: none"> 1. 制御対象の EV を決定し、EV が接続している充放電器を選択 2. アプリから該当する充放電器に対して、充電制御実行 (E->C) 3. EVにて充電開始
<p>3.8 EV および充放電器の状態変化 (充電終了、放電終了)</p>	<ol style="list-style-type: none"> 1. 充放電器は充電 (放電) が終わったことを確認 2. 充放電器は充電 (放電) が終わったことを通知 (E) 3. アプリへ該当の充放電器の状態が変わったことを通知 	<ol style="list-style-type: none"> 1. 充放電器は充電 (放電) が終わったことを確認 2. 充放電器は充電 (放電) が終わったことを通知 (E) 3. アプリへ該当の充放電器の状態が変わったことを通知 4. アプリにて充放電器の状態が変わった充放電器に接続している EV を判定
<p>3.9 EV の離脱</p>	<ol style="list-style-type: none"> 1. 車両離脱時の EV と充放電器間の処理 (C) 2. 車両接続・充放電可否状態を通知 (E) 	<ol style="list-style-type: none"> 1. 車両離脱時の EV と充放電器間の処理 (C) 2. 車両接続・充放電可否状態を通知 (E) 3. (アプリにて充放電器の識別子より離脱した EV を判定)

(E): ECHONET Lite で規定している内容、(C): CHAdeMO で規定している内容、
 (無印): それ以外 (他の手段)

それぞれの EV を識別不要なユースケースについて、シーケンスを以下の各節にて記載する。

また、EV と充放電器の紐づけを行うための標準仕様は現時点では存在しておらず、EV 識別するケースについては様々な方法が考えられる。そのためシーケンスには整理せずに、第4章にて EV と充放電器の紐づけ方法の事例を示す。

3. 2 EV の接続

EV が充放電器に接続した場合のシーケンスを示す。

1. EV と充放電器のケーブルを接続する。
2. コントローラは、定期的に充放電器の「車両接続・充放電可否状態」を取得する。コントローラは取得した値が「不定」の場合は、充放電器に対し「車両接続確認」を該当する充放電器に対して、書き込み要求を送信する
3. 充放電器は「車両接続・充放電可否状態」の値が変化した場合、ECHONET Lite のネットワーク上にマルチキャストで状態が変わったことを通知する。

EV 接続時のシーケンスを図 3 に示す。なお、図 3 のシーケンスは必ずしも時系列での順番を示すわけではない。

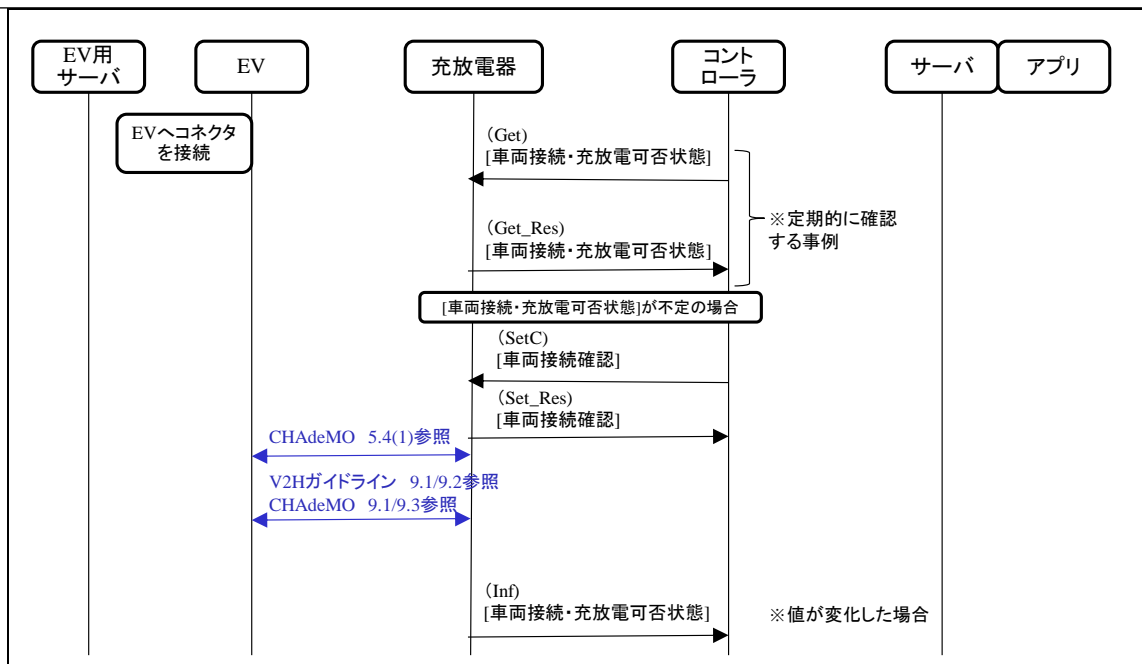


図 3 EV 接続時のシーケンス

なお、「単純に EV にケーブルがつながっていること」と「充電器が「接続」と認識しているかどうか」は意味が異なる。本ガイドラインにおいて、「接続」とは充電器が「接続」と認識していることを示す。「接続」と認識する方法は充電器の実装仕様に依存する。

3. 3 EV の接続検知とサービスへの通知

充電器が EV と接続したことを検知してからサーバ上のアプリへ EV と充電器が接続した情報が伝わるまでのシーケンスを示す。

1. 充電器は、3. 2 節の手順により EV とケーブルが接続されたことを検知すると、充電器自身が接続するネットワークに向けてマルチキャストにて「車両接続・充放電可否状態」が接続になったことを通知する。コントローラはその通知を受信することで、EV と充電器が接続した情報を取得する。
 2. もしくは、コントローラは定期的に充電器の「車両接続・充放電可否状態」の値を取得することで、EV と充電器が接続した情報を取得する。
 3. コントローラは、充電器を特定する ID と該当する充電器の「車両接続・充放電可否状態」をサーバへ送信する。サーバはその情報を取得することで、充電器への EV の接続状況、および充放電可否の情報を取得する。
- 以上のシーケンスを図 4 に示す。

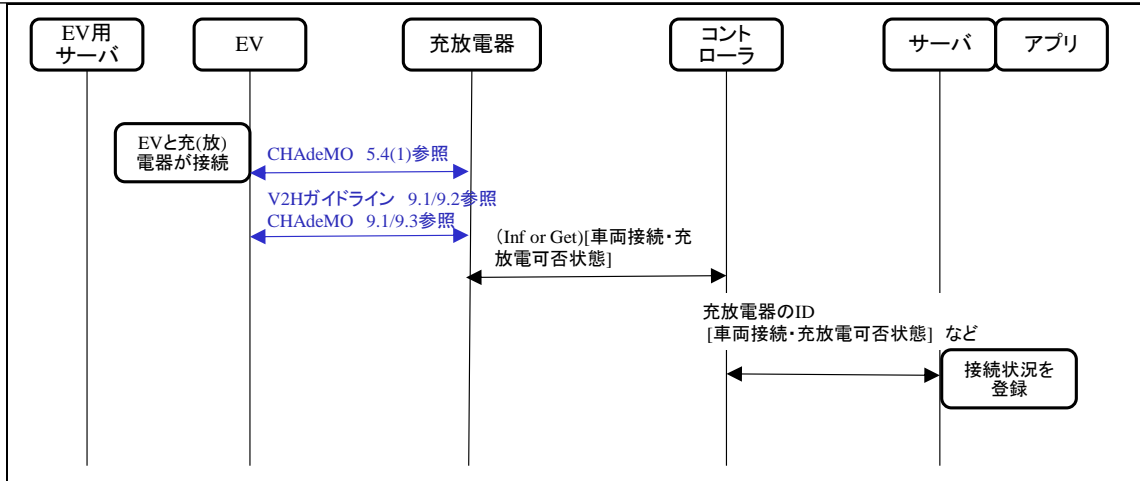


図 4 サービスへの機器登録時のシーケンス

なお、0 に示すように、充放電器と EV の紐づけ（どの充放電器にどの EV が接続しているか）を把握する必要があるケースがある。

3. 4 遠隔監視（運転モード設定 & 電池残容量）

コントローラが充放電器を遠隔監視する場合のシーケンスを以下に示す。遠隔監視を行う情報として、本節においては、充放電器が保持する情報である「運転モード設定」と EV が保持する情報である「電池残容量」を事例として説明する。

1. コントローラは充放電器に対して、「運転モード設定」と「車載電池の充電可能残容量値」を取得するための要求を送信する。
2. 電池残容量に関して、現在充電中もしくは放電中などにおいて EV と CAN 通信を行っている状態である場合、充放電器は「EV より取得した電池残容量」および「運転モード設定」をコントローラに応答する。また、EV と充放電器は接続しているが CAN 通信を行っていない場合は、充放電器は「最後に EV より取得した電池残容量」および「運転モード設定」をコントローラに応答する。充放電器が EV との接続を検知していない場合は、充放電器は EV の電池残容量に関する情報を応答できないため、コントローラに対して不可応答という形で「運転モード設定」のみを送信する。

CAN 通信を行っている状態のシーケンスを図 5 に、充放電器は EV との接続を検知しているが CAN 通信を行っていない状態のシーケンスを図 6 に、充放電器が EV との接続を検知していない場合のシーケンスを図 7 に示す。

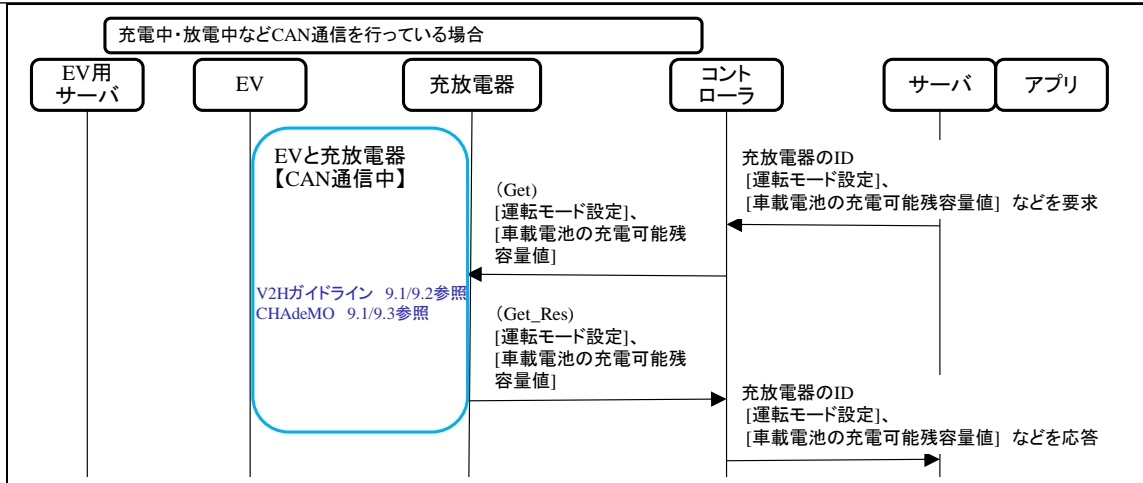


図5 遠隔監視時のシーケンス (1)

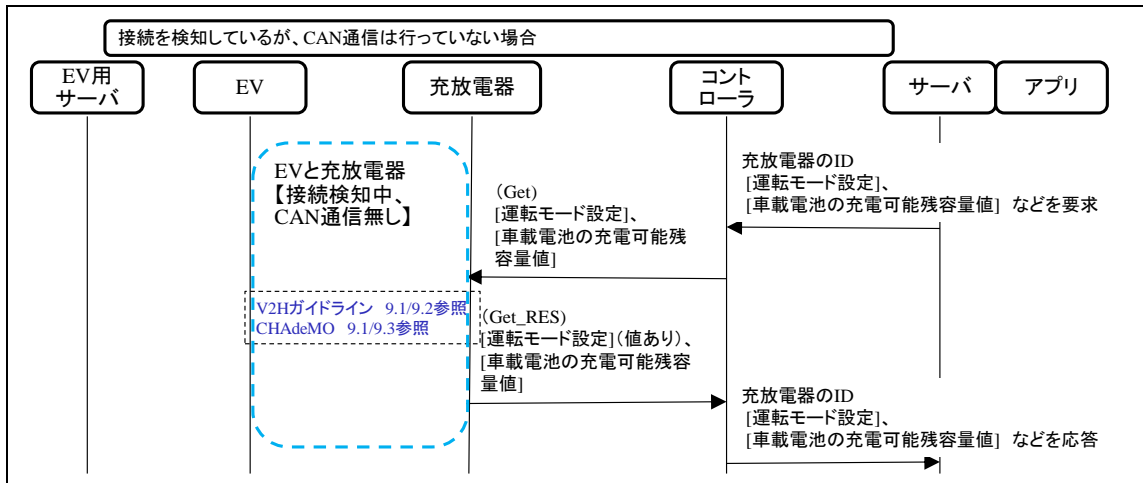


図6 遠隔監視時のシーケンス (2)

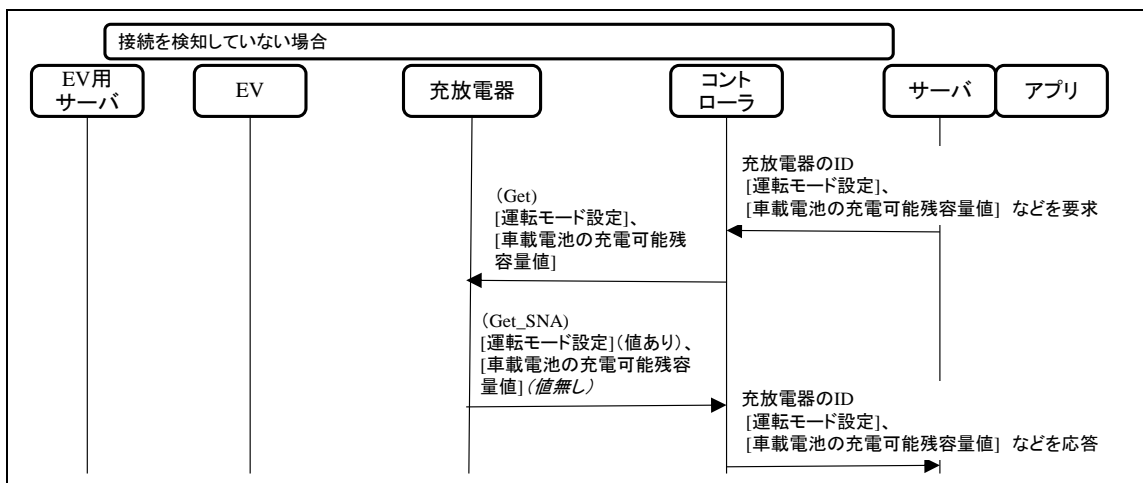


図7 遠隔監視時のシーケンス (3)

3. 5 遠隔より充放電器を制御（充電開始）

コントローラが充放電器に対して充電開始を実行するシーケンスを以下に示す。

1. コントローラは充放電器の「車両接続・充放電可否状態」を取得し、充電可能な状態の場合に、充放電器に対して「運転モード設定プロパティ」を充電として書き込み要求を送信する。
 2. 充放電器は、EVとの接続を検知していない状態などで充電できない状態の場合に、充電を要求する書き込み要求を受信しても、充電を開始しない。
 3. 充放電器はEVに対して充電の開始を要求する。
 4. 充電が開始すると、充放電器は運転モード設定が「充電」に変更した場合、ECHONET Liteのネットワーク上に「運転モード設定」が充電になったことをマルチキャストで通知する。
 5. またコントローラは、上記マルチキャストによる通知を受信できないケースに備え、書き込み要求送信後、制御が正常に実行できているか確認するため、該当する充放電器に対して「運転モード設定」の読み出し要求を送信する。
- 以上のシーケンスを図8に示す。

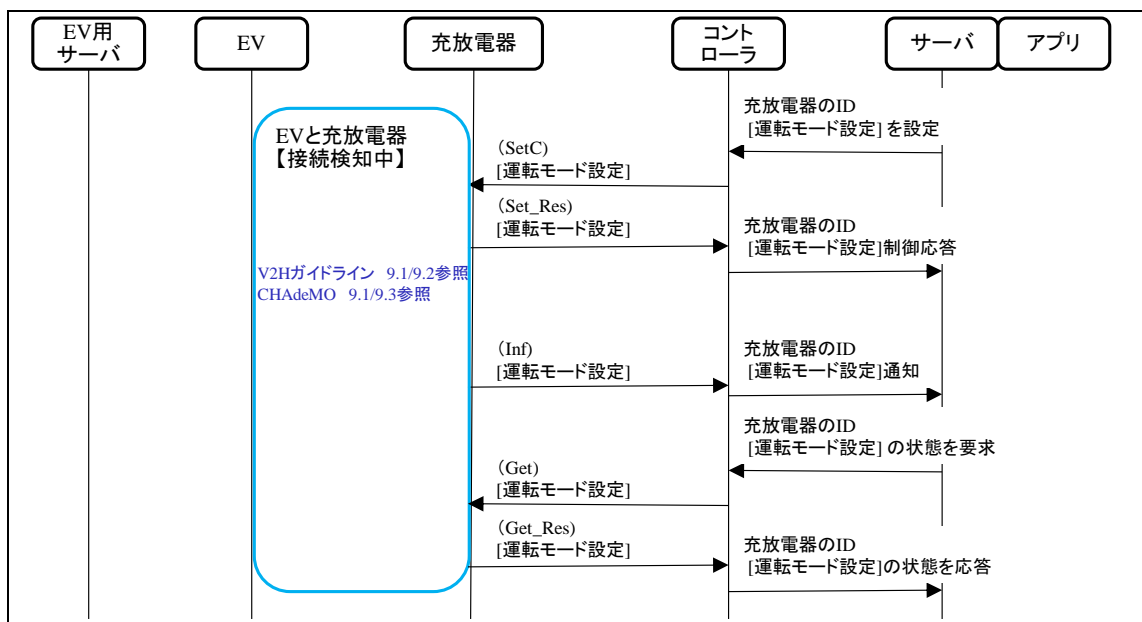


図8 充電開始時のシーケンス

3. 6 遠隔より充放電器を制御（放電開始）

コントローラが充放電器に対して放電開始を実行するシーケンスを以下に示す。

1. コントローラは充放電器の「車両接続・充放電可否状態」を取得し、放電可能な状態の場合に、充放電器に対して「運転モード設定プロパティ」を放電として書き込み要求を送信する。
 2. 充放電器は、EVとの接続を検知していない状態などで放電できる状態でない場合に、放電を要求する書き込み要求を受信しても、放電を開始しない。
 3. 充放電器はEVに対して放電の開始を要求する。
 4. 放電が開始すると、充放電器は運転モード設定が「放電」に変更した場合、ECHONET Liteのネットワーク上に「運転モード設定」が放電になったことをマルチキャストで通知する。
 5. またコントローラは、上記マルチキャストによる通知を受信できないケースに備え、書き込み要求送信後、制御が正常に実行できているか確認するため、該当する充放電器に対して「運転モード設定」の読み出し要求を送信する。
- 以上のシーケンスを図9に示す。

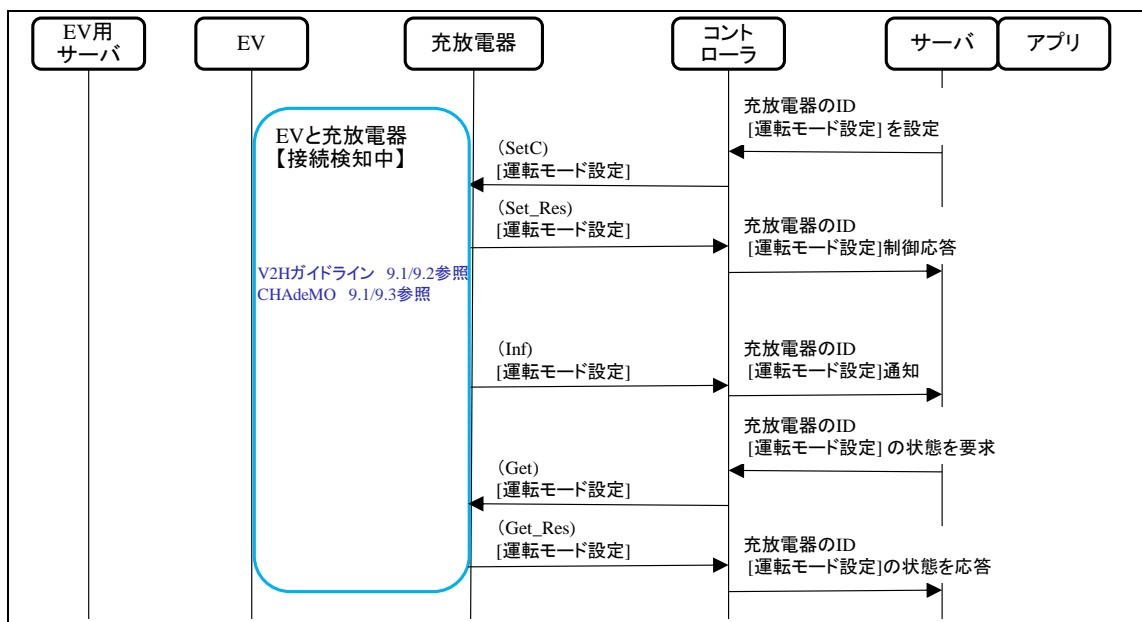


図9 放電開始時のシーケンス

3. 7 遠隔より充放電器を制御（充電電力／電流制御）

コントローラが充放電器に対して充電電力、充電電流、放電電力、放電電流を変更するための制御を実行するシーケンスを以下に示す。なお、「充電」に関する電力もしくは電流値を変更する場合の事例で記載しているが、放電関連を制御する場合は、充電をそれぞれ対象の放電に関するパラメータに置き換えることで対応可能である。

1. コントローラは充放電器に対して「充電電力設定値」を設定するために書き込み要求を送信する。
2. 現在充電中もしくは放電中などにおいて EV と CAN 通信を行っている場合、充放電器は EV から指定される「充電電流上限値」の範囲内である「充電電力設定値」もしくは「充電電流設定値」を受信した場合は、指定された値で充電を行う。ただし、EV から指定される「充電電流上限値」を超過する値の「充電電力設定値」もしくは「充電電流設定値」を受信した場合は、「充電電流上限値」の範囲内にて充電を行う（図 10 に示すシーケンス）。
3. また、EV と充放電器は接続しているが EV と CAN 通信を行っていない場合、および EV と充放電器は接続していない場合、充放電器は「充電電力設定値」もしくは「充電電流設定値」を受信した値を保持しておく。なお、EV と充放電器が CAN 通信を開始した後に充電を実施する時点で、EV から指定される「充電電流上限値」の範囲内かどうかの確認を行い、「充電電流上限値」の範囲内の場合は指定された値で充電を行い、「充電電流上限値」を超過している場合は「充電電流上限値」の範囲内にて充電を行う（図 11 に示すシーケンス）。
4. コントローラは、書き込み要求送信後、制御が正常に実行できているか確認するため、該当する充放電器に対して「充電電力設定値」の読み出し要求を送信する。

以上のシーケンスを図 10、および図 11 に示す。

一般的に、充放電器は EV から指定される「充電電流上限値」および「放電電流上限値」の範囲内で充電・放電が可能である。例えば、「EV が指定する充電電流上限値」>「コントローラが指定する充電電流設定値」である場合、充電電流設定値の電流値で基本的には充電する。また、「EV が指定する充電電流上限値」<「コントローラが指定する充電電流設定値」である場合、充電電流上限値の電流値で基本的には充電する。

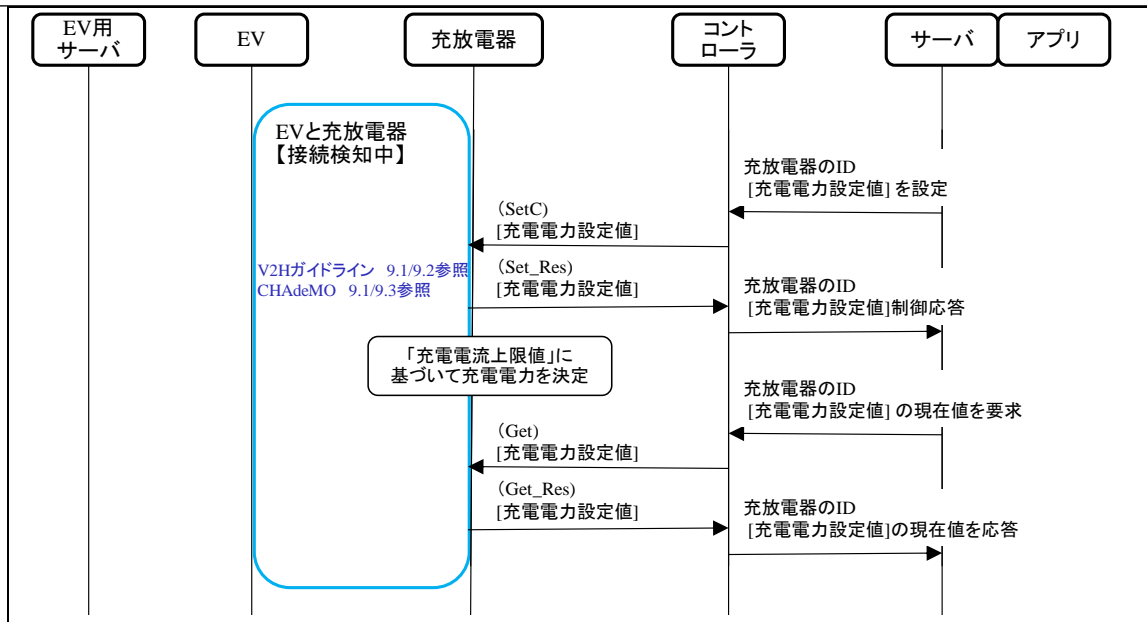


図 10 充電電力設定時のシーケンス (1)

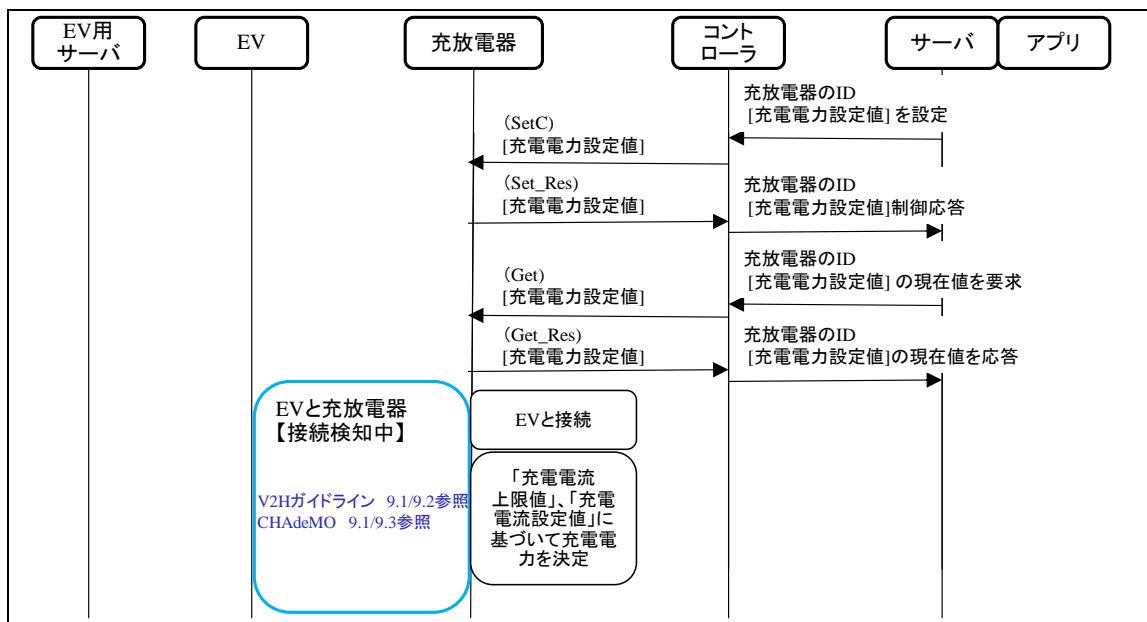


図 11 充電電力設定時のシーケンス (2)

3. 8 EV および充放電器の状態変化（充電終了、放電終了）

充放電器が充電、また放電を終了したと判断した場合のシーケンスを以下に示す。本節においては、「充電終了」に関する場合の事例で記載しているが、放電終了時も同様のシーケンスでの対応が可能である。充電終了する場合として、「EVの車載電池が満充電になる場合」、「事前にコントローラから充放電器に充放電量を指定されており、その指定された充電量（もしくは放電量）に達した場合」などがある。

1. 充放電器は充電終了と判断した場合、EV に対して充放電制御の終了を指示する。もしくは EV より充放電制御の終了を受信すると充電終了と判断する。
 2. 充放電器は運転モード設定が「停止」、もしくは「待機」に変更したことを ECHONET Lite のネットワーク上にマルチキャストで通知する。運転モード設定が「停止」、もしくは「待機」のいずれの状態になるかは、その時の充放電器の状態に依存する。なお、ECHONET Lite のネットワーク上へのマルチキャスト通信と上記 EV との充放電制御の終了に関する通信のタイミングについては、充放電器の実装仕様次第である。
 3. コントローラは、運転モード設定の通知を受信すると、コントローラサーバに対して充放電器の ID とともに運転モード設定が変更したことを通知する。
- 以上のシーケンスを図 12 に示す。

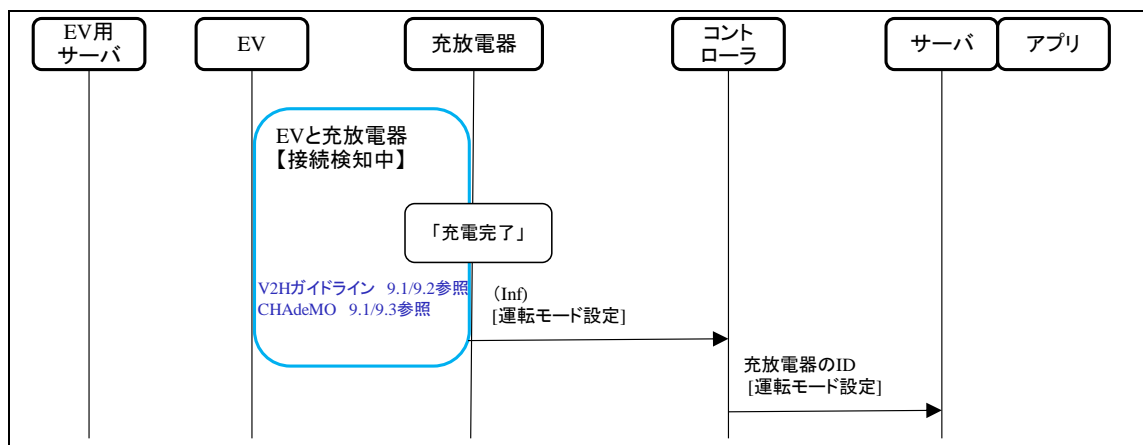


図 12 充電終了時のシーケンス

3. 9 EV の離脱

EV からケーブルを外して充放電器から離脱した場合のシーケンスを以下に示す。基本的には、充電中や放電中にケーブルを着脱することはない。

1. 充放電器は、EV が離脱されたことを検知すると、「車両接続・充放電可否状態」の値を「車両未接続」に変更し、ECHONET Lite のネットワーク上にマルチキャストで通知する。
2. コントローラは、車両接続・充放電可否状態の通知を受信すると、コントローラサーバに対して充放電器の ID とともに車両接続・充放電可否状態が車両未接続であることを通知する。

以上のシーケンスを図 13 に示す。

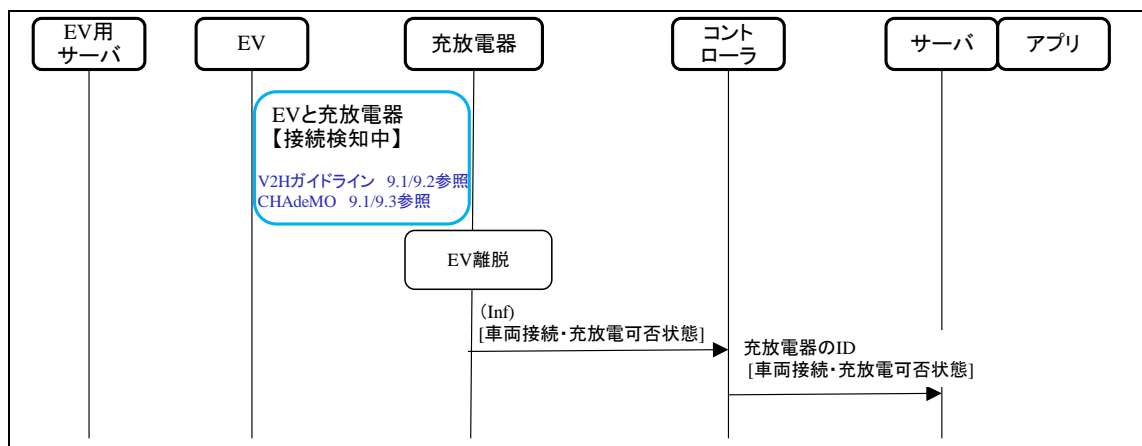


図 13 EV 離脱時のシーケンス

第 4 章 EV と充放電器の紐づけの指針

4. 1 基本的な考え方

EV と充放電器との組み合わせをコントローラやサーバ上のアプリケーションが識別するための方法として、以下の事例があげられる。

- 充放電器で EV と充放電器の ID を統合する
 - CHAdEMO プロトコルを用いて、充放電器は EV より車両 ID を取得する
 - 外部端末などを使って、EV の ID を充放電器に入力する
- 外部端末が EV と充放電器の ID を統合する
- EV が EV と充放電器の ID を統合する

本章において、それぞれの場合における紐づけの事例と、参照すべき各機器間の通信仕様を整理する。なお、CHAdEMO、および ECHONET Lite で規定している車両 ID の定義については、本ガイドラインでは特に指定しない。

4. 2 充放電器で紐づける場合 1

CHAdEMO プロトコルを用いて、充放電器は EV より車両 ID を取得する場合の構成事例を図 14 に示す。

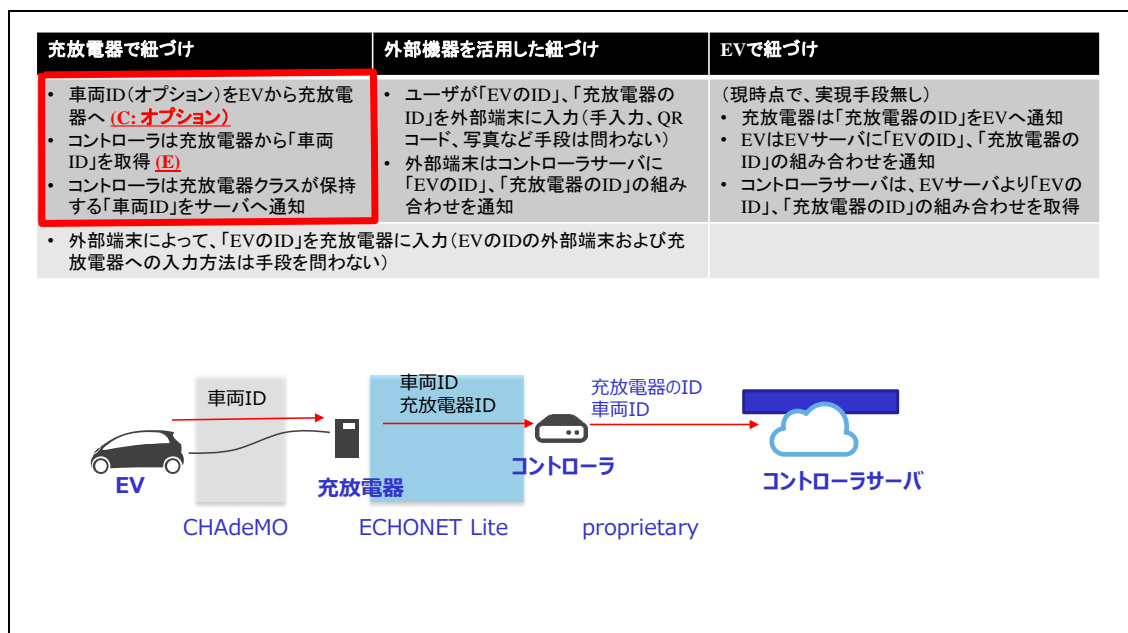


図 14 充放電器にて EV と充放電器の紐づけを行う場合(1)

充放電器は、EV より車両 ID を取得する。コントローラは充放電器に対して、「車両 ID プロパティ」を取得することで、充放電器と EV との紐づけを確定することができる。各機器間の通信仕様を表 2 に示す。

表 2 機器間に用いる通信仕様リスト

機器間	通信仕様
EV と充放電器間	<ul style="list-style-type: none"> CHAdeMO 標準仕様書 V2H/L ガイドライン
充放電器とコントローラ間	<ul style="list-style-type: none"> ECHONET Lite 規格 APPENDIX ECHONET 機器オブジェクト詳細規定 電気自動車充放電器／電気自動車充電器・HEMS コントローラ間アプリケーション通信インタフェース仕様書
コントローラとサーバ間	<ul style="list-style-type: none"> 基本的には同一ベンダーでの運用を想定 通信仕様は各社独自仕様

4. 3 充放電器で紐づける場合 2

外部端末を用いて、充放電器は EV の ID を取得する場合の構成事例を図 15 に示す。

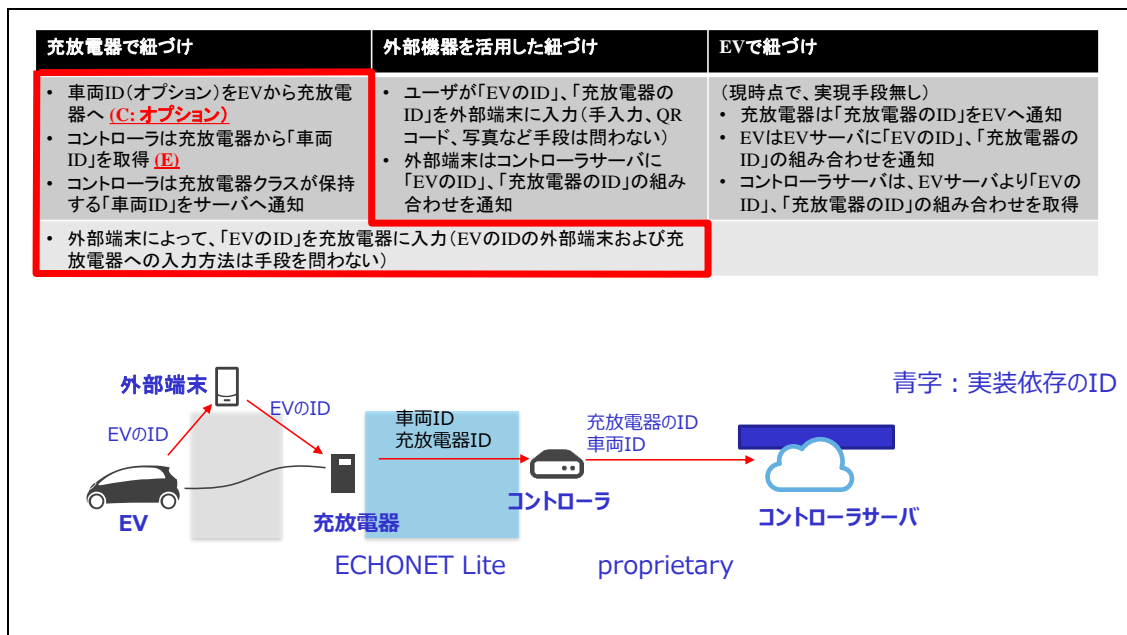


図 15 充放電器にて EV と充放電器の紐づけを行う場合(2)

外部端末は、EV の ID を EV から取得する。ここで示す「EV の ID」とは、以下のような事例が考えられる

- EV が保持する ID
- 外部端末を導入するサービス事業者が EV に付与する ID (例えば、QR コードを EV に貼付しそれを外部端末で読み取る、手動で外部端末に ID を入力)

するなど)

外部端末は、EV の ID を充放電器に設定する。設定方法については、外部端末と充放電器間で規定する。コントローラは充放電器に対して、「車両 ID プロパティ」を取得することで、充放電器と EV との紐づけを確定することができる。

各機器間の通信仕様は下記に示す。

表 3 機器間に用いる通信仕様リスト

機器間	通信仕様
EV と外部端末間	<ul style="list-style-type: none"> 外部端末を導入するシステム仕様に準じる 必ずしも EV と外部端末間は通信を行うわけではない。例えば、手入力、EV に貼付する QR コード読み取りなど
外部端末と充放電器間	<ul style="list-style-type: none"> 外部端末を導入するシステム仕様に準じる 必ずしも外部端末と充放電器間は通信を行うわけではない。例えば外部端末に表示しデータを充放電器に手入力することも想定される
充放電器とコントローラ間	<ul style="list-style-type: none"> ECHONET Lite 規格 APPENDIX ECHONET 機器オブジェクト詳細規定 電気自動車充放電器／電気自動車充電器・HEMS コントローラ間アプリケーション通信インタフェース仕様書
コントローラとサーバ間	<ul style="list-style-type: none"> 基本的には同一ベンダーでの運用を想定 通信仕様は各社独自仕様

4. 4 外部端末を活用して紐づける場合

外部端末が充放電器の ID および EV の ID を取得し、紐づけを行う場合の構成事例を図 16 に示す。

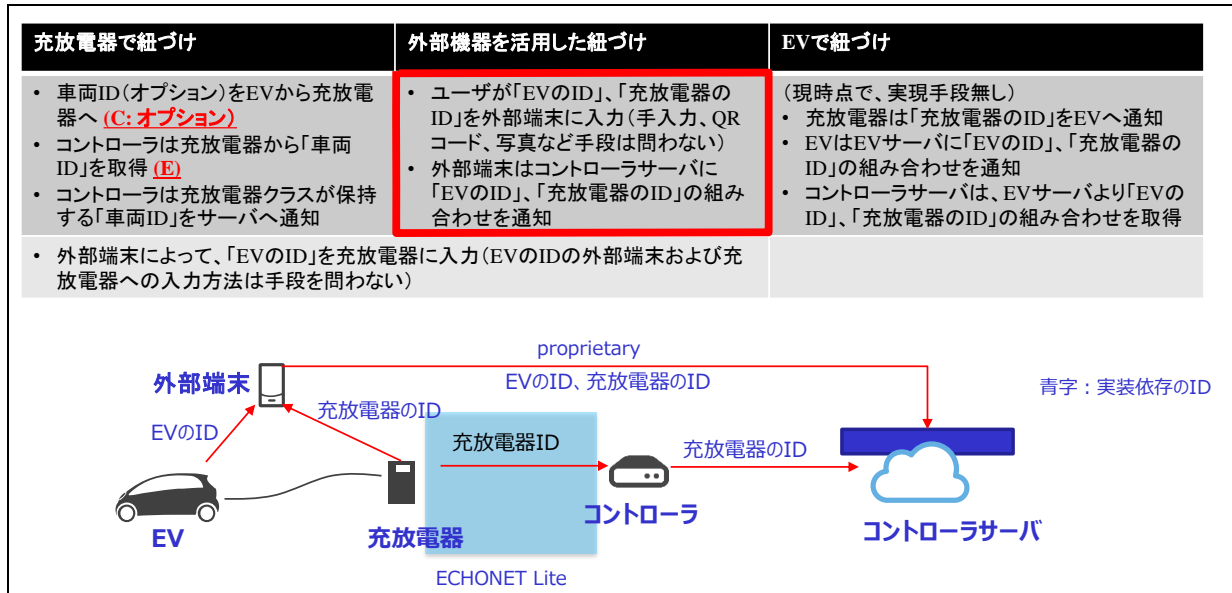


図 16 外部端末にて EV と充放電器を紐づける場合

外部端末は、EV の ID を EV から取得し、充放電器の ID を充放電器から取得する。ここで示す「EV の ID」、および「充放電器の ID」とは、以下のような事例が考えられる

- EV が保持する ID、充放電器が保持する ID
- 外部端末を導入するサービス事業者が EV、および充放電器に付与する ID (例えば、QR コードを EV に貼付しそれを外部端末で読み取る、手動で外部端末に ID を入力するなど)

外部端末は、EV と充放電器から取得した ID を元にして、充放電器と EV との紐づけを確定することができる。各機器間の通信仕様は下記に示す。

表 4 機器間に用いる通信仕様リスト

機器間	通信仕様
EV と外部端末間	<ul style="list-style-type: none"> 外部端末を導入するシステム仕様に準じる 必ずしも EV と外部端末間は通信を行うわけではない。例えば、手入力、EV に貼付する QR コード読み取りなど
外部端末と充放電器間	<ul style="list-style-type: none"> 外部端末を導入するシステム仕様に準じる 必ずしも充放電器と通信を行うわ

	<ul style="list-style-type: none"> • 基本的には同一ベンダーでの運用を想定 • 通信仕様は各社独自
外部端末とコントローラサーバ間	<ul style="list-style-type: none"> • 基本的には同一ベンダーでの運用を想定 • 通信仕様は各社独自
充放電器とコントローラ間	<ul style="list-style-type: none"> • ECHONET Lite 規格 • APPENDIX ECHONET 機器オブジェクト詳細規定 • 電気自動車充放電器／電気自動車充電器・HEMS コントローラ間アプリケーション通信インタフェース仕様書
コントローラとコントローラサーバ間	<ul style="list-style-type: none"> • 基本的には同一ベンダーでの運用を想定 • 通信仕様は各社独自仕様

4. 5 EV で紐づける場合

EV は充放電器の ID を何らかの通信仕様によって取得する場合の構成事例を図 17 に示す。なお、現時点では定められた通信仕様の規程はなく将来規定された場合の想定である。

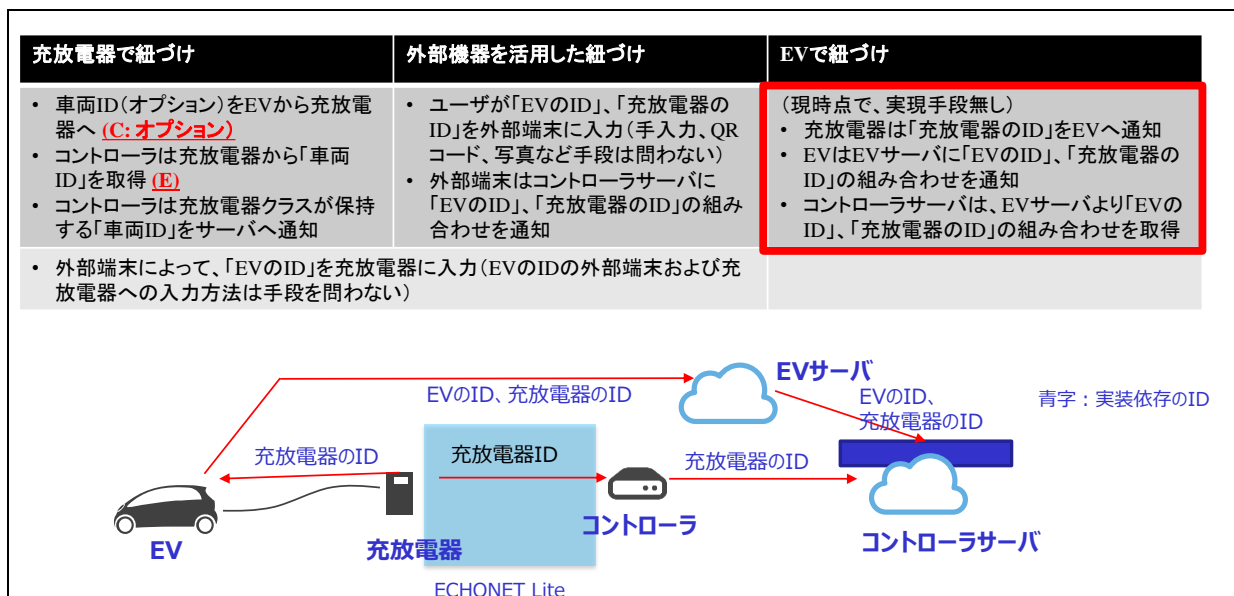


図 17 EV にて EV と充放電器を紐づける場合

EV は充放電器の ID を充放電器から取得し、EV にて EV と充放電器の紐づけを確定し、EV サーバに情報をアップロードする。コントローラサーバは、例えば充放電器の ID をキーにして、EV サーバより充放電器と EV の組み合わせ情報を取得する。ここで示す「EV の ID」、および「充放電器の ID」とは、以下のような

事例が考えられる

- EV が保持する ID、充放電器が保持する ID

EV は、EV 自身の ID と充放電器から取得した ID を元にして、充放電器と EV との紐づけを確定することができる。各機器間の通信仕様は下記に示す。

表 5 機器間に用いる通信仕様リスト

機器間	通信仕様
EV と充放電器間	<ul style="list-style-type: none">・ 現状対応する仕様無し
EV と EV サーバ間	<ul style="list-style-type: none">・ 基本的には同一ベンダーでの運用を想定・ 通信仕様は各社独自
充放電器とコントローラ間	<ul style="list-style-type: none">・ ECHONET Lite 規格・ APPENDIX ECHONET 機器オブジェクト詳細規定・ 電気自動車充放電器／電気自動車充電器・HEMS コントローラ間アプリケーション通信インタフェース仕様書
コントローラとコントローラサーバ間	<ul style="list-style-type: none">・ 基本的には同一ベンダーでの運用を想定・ 通信仕様は各社独自仕様
EV サーバとコントローラサーバ間	<ul style="list-style-type: none">・ 両社間の契約に基づいて接続・ 通信仕様（セキュリティ含む）は、両者合意の元、サービスに基づいた通信仕様を策定