

第 6 部 ECHONET 個別下位通信インタフェース仕様

改定履歴

- Version1.0 2000年3月18日 制定，コンソーシアム会員内公開。
2000年7月 一般公開。
- Version1.01 2001年5月23日 コンソーシアム会員内公開。
Version1.0の追補&正誤反映版。
- Version2.00 2001年8月07日 コンソーシアム会員内公開。
電灯線 A，B方式の、電灯線方式としての統合
(A方式で統一)に伴う記述修正。
第6部における変更のある目次項目は、以下の通り。

	変更部位 (目次項目)	追加・変更概要
1	4.2.1, 4.2.2, 4.2.8	電灯線 A,B方式の1本化による記述変更。
2	4.2.19, 4.2.20	電灯線 A,B方式の1本化により削除。
3	4.3	電灯線 A,B方式の1本化による記述変更。
4	4.3.2, 4.3.3, 4.3.4, 4.3.5	電灯線 A,B方式の1本化により、項番号が変更。(V.1.01 から一つづつ繰り上がり。)

- Version2.01 2001年12月19日 コンソーシアム会員内公開。
- Version2.10Preview 2001年12月28日 コンソーシアム会員内公開。
- Version2.10Draft 2002年2月15日 コンソーシアム会員内公開。
- Version2.10 2002年3月7日 コンソーシアム会員内公開。

	変更部位 (目次項目)	追加・変更概要
1	2.1	・第2部状態遷移の改定にともない、以下のインタフェースを追加。 「完全初期化要求」、「通信停止要求」、「完全停止要求」
2	2.1	・第2部状態遷移の改定にともない、「リセット要求」を「ウォームスタート要求」に名称変更。
3	2.2	・第2部状態遷移の改定にともない、インタフェースの詳細説明を変更。
4	3.1	・第2部状態遷移の改定にともない、以下のインタフェースを追加。 「完全初期化要求」、「通信停止要求」、「完全停止要求」
5	3.1	・第2部状態遷移の改定にともない、「リセット要求」を「ウォームスタート要求」に名称変更。
6	3.2	・第2部状態遷移の改定にともない、インタフェースの詳細説明を変更。

	変更部位 (目次項目)	追加・変更概要
7	4.1	・第 2 部状態遷移の改定にともない、以下の API を追加。 「LowInitAll」、「LowStop」、「LowHalt」
8	4.1	・第 2 部状態遷移の改定にともない、「LowReset」を「LowStart」に名称変更。
9	4.2	・第 2 部状態遷移の改定にともない、API の詳細説明を変更。

・ Version2.11

2002年4月26日 コンソーシアム会員内公開。
変更のある項目は下記の通り

	変更部位 (目次項目)	追加・変更概要
1	4.2.9	・使用する構造体の下位通信ソフトウェア部の状態を追記
2	4.2.17	・構文の引数 mac の型をポインタ型に修正、また、新たに mac_len を追加
3	4.2.18	・構文の引数 map の型をポインタ型に修正、また、新たに map_len を追加
4	4.2.20	・機能の記述を訂正

・ Version3.00Draft

2002年6月12日 コンソーシアム会員内公開。
変更のある項目は下記の通り

	変更部位 (目次項目)	追加・変更概要
1	1.2	図 1.1 新規伝送メディア追加
2	2.1	下位通信ソフトウェアアドレステーブルデータ要求、マスタータ通知、ハードウェアアドレスデータ要求を追記
3	2.2	下位通信ソフトウェアアドレステーブルデータ要求、マスタータ通知、ハードウェアアドレスデータ要求の説明を追記
4	3.1	表 3.1 に下位通信ソフトウェアアドレステーブルデータ要求、マスタータ通知、ハードウェアアドレスデータ要求を追記
5	3.2	下位通信ソフトウェアアドレステーブルデータ要求、マスタータ通知、ハードウェアアドレスデータ要求の説明を追記
6	4.1	表 4.1 に LowGetAddressTableData、LowSetMasterRouterFlag、LowGetHardwareAddress を追記
7	4.2.22	LowGetAddressTableData の説明を追記
8	4.2.23	LowSetMasterRouterFlag の説明を追記
9	4.2.24	LowGetHardwareAddress の説明を追記

・ Version3.00

2002年8月29日 コンソーシアム会員内公開。

- Version3.10Draft 2002年11月8日 コンソーシアム会員内公開。
変更のある項目は下記の通り

	変更部位 (目次項目)	追加・変更概要
1	2.2	・「(23)下位通信ソフトウェアアドレステーブルデータサイズ取得」追加
2	4.1	・誤記修正 ・表4.1の表番号修正 ・表4.1のNo.修正 ・表4.1にLowReceiveStop追加
3	4.2.22	・関数 LowReceiveStop を追加

- Version3.10 2002年12月18日 コンソーシアム会員内公開。
• Version3.11 2003年3月7日 コンソーシアム会員内公開。
• Version3.12 2003年5月22日 コンソーシアム会員内公開。

変更のある項目は下記の通り

	変更部位 (目次項目)	追加・変更概要
1	3.1 3.2	・「下位通信ソフトウェアアドレステーブルデータサイズ取得」追加

- Version3.20Draft 2003年10月17日 コンソーシアム会員内公開。
• Version3.20 2004年1月8日 コンソーシアム会員内公開。

変更のある項目は下記の通り

	変更部位 (目次項目)	追加・変更概要
1	2.1	・「(27)ECHONET MAC アドレスリスト取得」「(28)マスタルータ情報取得」「(29)ハードウェアアドレス変換要求」追加
2	2.2	・「(27)ECHONET MAC アドレスリスト取得」「(28)マスタルータ情報取得」追加「(29)ハードウェアアドレス変換要求」の説明を追加
3	3.1	・表3.1の「No.5 ウォームスタート要求」「No.11 送信結果取得」を Required から Optional に変更。 ・表3.1に「No.27 ECHONET MAC アドレスリスト取得」「No.28 マスタルータ情報取得」追加「No.29 ハードウェアアドレス変換要求」の説明を追加
4	3.2	・(1)誤記修正 ・(2)sdbuf、rdbufを削除 ・(17)表3.18の表番号修正

		<ul style="list-style-type: none"> ・ (2 3) 表 3 . 2 3 の表番号を修正 ・ (2 4) の番号重複を修正 ・ (2 7)(2 8)(2 9) を追加
5	4.1	<ul style="list-style-type: none"> ・ 誤記修正 ・ 表 4 . 1 の No.7 の誤記修正 ・ 表 4 . 1 の「No.6 一時停止要求動作」、「No.7 動作再開要求」、「No.11 送信結果取得」、「No.12 送信中止要求」、「No.15 アドレス情報設定要求」の Required を Optional に変更 ・ 表 4 . 1 に「No.27 ECHONET MAC アドレスリスト取得」、「No.28 マスタルータ情報取得」追加「No.29 ハードウェアアドレス変換要求」を追加
6	4.2.8	<ul style="list-style-type: none"> ・ 誤記修正
7	4.2.18	<ul style="list-style-type: none"> ・ 誤記修正
8	4.2.27	<ul style="list-style-type: none"> ・ 関数「LowGetEchonetMACList」を追加
9	4.2.28	<ul style="list-style-type: none"> ・ 関数「LowGetMasterRouterInfo」を追加
10	4.2.29	<ul style="list-style-type: none"> ・ 関数「LowReqToHardwareAddress」を追加

- ・ Version3.21 2 0 0 4 年 5 月 2 6 日 コンソーシアム会員内公開。
- ・ Version3.21 2 0 0 5 年 10 月 1 3 日 一般公開。

- ・ エコーネットコンソーシアムが発行している規格類は、工業所有権(特許, 実用新案など)に関する抵触の有無に関係なく制定されています。エコーネットコンソーシアムは、この規格類の内容に関する工業所有権に対して、一切の責任を負いません。
- ・ 本規格発行者は有償・無償を問わず、いかなる第三者に対しても JAVA、IrDA、Bluetooth、HBS のライセンスを許諾する権限や免責を与える権限を有していません。JAVA、IrDA、Bluetooth、HBS を使用する場合、当該使用者は自己の責任と判断に基づき、上記規格について使用許可を得るなどの措置が必要です。
- ・ この書面の使用による、いかなる損害も責任を負うものではありません。

目次

第 1 章 ECHONET 個別下位通信インタフェース仕様概要	1 - 1
1 . 1 基本的な考え方.....	1 - 1
1 . 2 通信レイヤ上の位置づけ	1 - 2
第 2 章 ECHONET 個別下位通信インタフェース機能仕様	2 - 1
2 . 1 ECHONET 個別下位通信インタフェース機能一覧	2 - 1
2 . 2 ECHONET 個別下位通信インタフェース機能詳細仕様	2 - 2
第 3 章 レベル 1 ECHONET 個別下位通信インタフェース仕様	3 - 1
3 . 1 レベル 1 ECHONET 個別下位通信インタフェースのサービス一覧.....	3 - 1
3 . 2 レベル 1 ECHONET 個別下位通信インタフェースのサービス詳細仕様.	3 - 3
第 4 章 レベル 2 ECHONET 個別下位通信インタフェース仕様	4 - 1
4 . 1 レベル 2 ECHONET 個別下位通信インタフェース一覧	4 - 2
4 . 2 レベル 2 ECHONET 個別下位通信インタフェース詳細仕様	4 - 4
4 . 2 . 1 LowGetDevID	4 - 5
4 . 2 . 2 LowInit	4 - 6
4 . 2 . 3 LowRequestRun.....	4 - 7
4 . 2 . 4 LowSetTrouble	4 - 8
4 . 2 . 5 LowStart	4 - 9
4 . 2 . 6 LowSuspend.....	4 - 10
4 . 2 . 7 LowWakeUp.....	4 - 11
4 . 2 . 8 LowGetProData	4 - 12
4 . 2 . 9 LowGetStatus	4 - 14
4 . 2 . 10 LowSendData	4 - 15
4 . 2 . 11 LowGetSendResult.....	4 - 16
4 . 2 . 12 LowSendCancel.....	4 - 17
4 . 2 . 13 LowReceiveData.....	4 - 18
4 . 2 . 14 LowGetAddress	4 - 19
4 . 2 . 15 LowSetAddress.....	4 - 20
4 . 2 . 16 LowReqToMac	4 - 21
4 . 2 . 17 LowReqToID	4 - 22
4 . 2 . 18 LowReqBcastID.....	4 - 23
4 . 2 . 19 LowInitAll	4 - 24
4 . 2 . 20 LowStop.....	4 - 26
4 . 2 . 21 LowHalt.....	4 - 27
4 . 2 . 22 LowReceiveStop	4 - 28
4 . 2 . 23 LowGetAddressTableDataSize	4 - 29

4 . 2 . 2 4	LowGetAddressTableData	4 - 3 0
4 . 2 . 2 5	LowSetMasterRouterFlag.....	4 - 3 1
4 . 2 . 2 6	LowGetHardwareAddress.....	4 - 3 2
4 . 2 . 2 7	LowGetEchonetMACList	4 - 3 3
4 . 2 . 2 8	LowGetMasterRouterInfo	4 - 3 5
4 . 2 . 2 9	LowReqToHardwareAddress.....	4 - 3 6
4 . 3	イニシャル時設定情報仕様	4 - 3 7
4 . 3 . 1	電灯線下位通信ソフトウェア用初期化パラメータ仕様	4 - 3 8
4 . 3 . 2	特定小電力無線下位通信ソフトウェア用初期化パラメータ仕様	4 - 3 8
4 . 3 . 3	拡張 HBS 下位通信ソフトウェア用初期化パラメータ仕様	4 - 3 8
4 . 3 . 4	IrDA 依存下位通信ソフトウェア用初期化パラメータ仕様	4 - 3 8
4 . 3 . 5	LonTalk ^R 依存下位通信ソフトウェア用初期化パラメータ仕様....	4 - 3 8
4 . 3 . 6	Bluetooth TM 下位通信ソフトウェア用初期化パラメータ仕様 ...	4 - 3 9
4 . 3 . 7	Ethernet 下位通信ソフトウェア用初期化パラメータ仕様	4 - 3 9

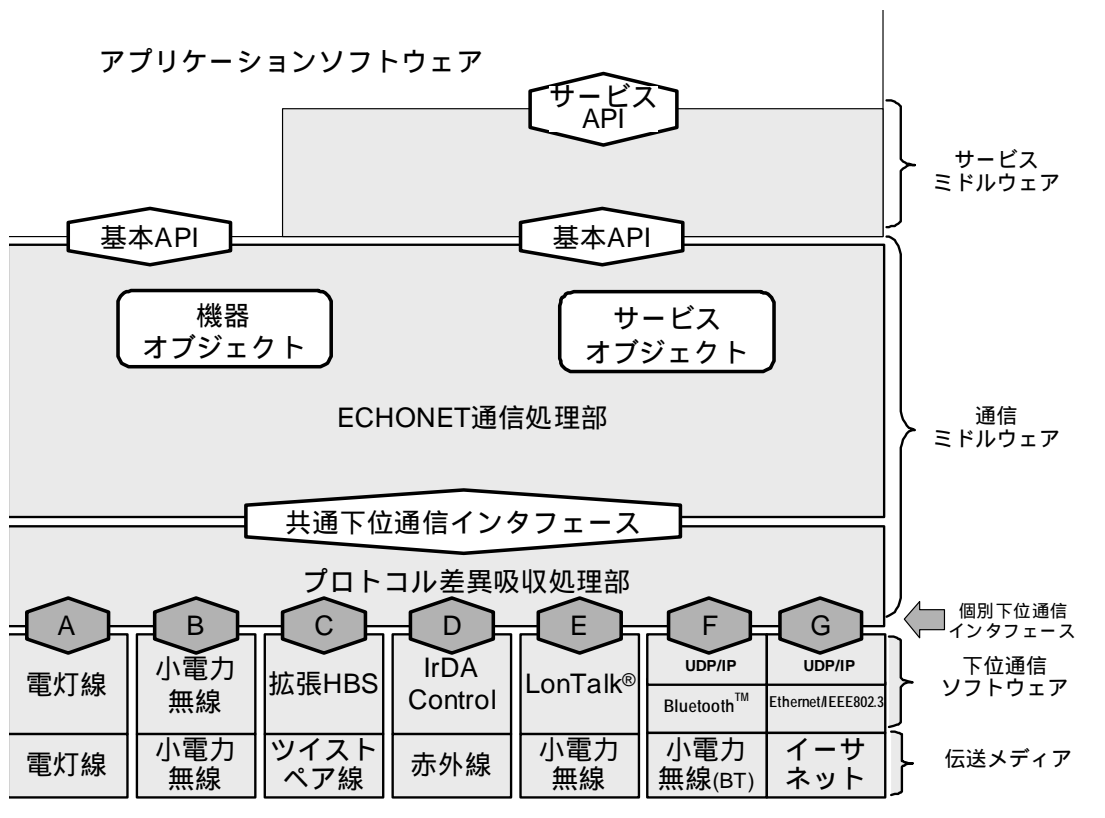
第 1 章 ECHONET 個別下位通信インタフェース仕様概要

1 . 1 基本的な考え方

「第 6 部 ECHONET 個別下位通信インタフェース仕様」の規定は、次節の図 1 . 1 にて示すプロトコル差異吸収処理部と下位通信ソフトウェアとの間の処理や情報のやり取りを実現する為のソフトウェアインタフェースの仕様に関するものである。個別下位通信インタフェースを介して、プロトコル差異吸収処理部は、下位通信ソフトウェアとのやり取りを実現することができる。この個別下位通信インタフェースの仕様では、下位通信ソフトウェアがサポートする想定でのインタフェースサービスについて、入出力データ項目、及び具体的な言語が指定された場合の関数規定を、それぞれ、個別下位通信インタフェース仕様レベル 1 , 2 として規定する。個別下位通信インタフェース仕様レベル 1 , 2 は、基本 API のレベル 1 , 2 の考えに基づく。

1.2 通信レイヤ上の位置づけ

図 1.1 に、個別下位通信インタフェースの位置づけを網掛けで示す。個別下位通信インタフェースは、プロトコル差異吸収処理部と下位通信ソフトウェアの間において処理呼出や情報のやり取りを実現することにより、通信ミドルウェアと下位通信ソフトウェアとの接続を実現するものである。



LonTalk は、米国その他の国々での Echelon Corporation の登録商標です。
 Bluetooth は Bluetooth SIG, Inc の登録商標です。
 Ethernet は Xerox Corporation の登録商標です。
 その他すべての商標は、それぞれの所有者に属するものです。

図 1.1 個別下位通信インタフェースの位置づけ

第 2 章 ECHONET 個別下位通信インタフェース機能仕様

2 . 1 ECHONET 個別下位通信インタフェース機能一覧

下位通信ソフトウェアのサポートする ECHONET 個別下位通信インタフェースの機能一覧を以下に示す。下位通信ソフトウェアは、これらのインタフェースを備えているものとする。各インタフェースの機能詳細説明は、次節にて示す。

- (1)下位通信ソフトウェア種別要求
- (2)初期化要求
- (3)動作開始要求
- (4)障害通知
- (5) ウォームスタート要求
- (6)一時停止要求
- (7)動作再開要求
- (8)プロファイル取得
- (9)ステータス取得
- (10)電文送信要求
- (11)送信結果取得
- (12)送信中止要求
- (13)受信電文要求
- (14)アドレス情報取得
- (15)アドレス情報設定
- (16)物理アドレス変換要求
- (17)NodeID 変換要求
- (18)同報先取得要求
- (19)完全初期化要求
- (20)通信停止要求
- (21)完全停止要求
- (22)停止通知
- (23)下位通信ソフトウェアアドレステーブルデータサイズ要求
- (24)下位通信ソフトウェアアドレステーブルデータ要求
- (25)マスタルータ通知
- (26)ハードウェアアドレスデータ要求(27)ECHONET MAC アドレスリスト取得
- (28)マスタルータ情報取得
- (29)ハードウェアアドレス変換要求

2.2 ECHONET 個別下位通信インタフェース機能詳細仕様

下位通信ソフトウェアがサポートする ECHONET 個別下位通信インタフェースの機能の説明を以下に示す。なお、本節の説明の中で使用されている下位通信ソフトウェアの遷移状態に関しては、第 3 部の対応する下位通信ソフトウェアを参照されたい。

(1) 搭載情報要求

下位通信ソフトウェアに関する情報(搭載している下位通信ソフトウェアの数と下位通信ソフトウェア ID)を要求する。

(2) 初期化要求

下位通信ソフトウェアに対し、コールドスタートによる初期化を実施後、通信停止状態に遷移することを要求する。このとき、下位通信ソフトウェアの保持している MAC アドレスは、破棄・更新される。

(3) 動作開始要求

下位通信ソフトウェアに対し、通信停止状態から通常動作状態に遷移することを要求する。

(4) 障害通知

プロトコル差異吸収処理部より上位層の障害(異常)状態を通知する。

(5) ウォームスタート要求

下位通信ソフトウェアに対し、ウォームスタートによる初期化を実施後、通信停止状態に遷移することを要求する。このとき、下位通信ソフトウェアが保持している MAC アドレスは変化しない。

(6) 一時停止要求

下位通信ソフトウェアに対し、通常動作状態から一時停止状態に遷移することを要求する。

(7) 動作再開要求

下位通信ソフトウェアに対し、一時停止状態を解除し、通常動作状態に遷移することを要求する。

(8) プロファイル取得

下位通信ソフトウェアに対し、プロファイルデータを要求する。本機能にて要求するプロファイルデータは、下位通信ソフトウェアの開発メーカーコードやバージョン No.等の静的情報とする。

(9) ステータス取得

下位通信ソフトウェアに対して、ステータスデータを要求する。本機能にて要求するステータスデータは、異常状態や処理状態といった下位通信ソフトウェアの動的情報とする。

(1 0) 電文送信要求

下位通信ソフトウェアに対して、指定する ECHONET 電文の送信を要求する。

(1 1) 送信結果取得

下位通信ソフトウェアに対して、この要求の直前に要求した電文送信の処理のステータスを要求する。

(1 2) 送信中止要求

下位通信ソフトウェアに対して、この要求の直前の電文送信要求に応じて行っている電文送信処理の中止を要求する。

(1 3) 受信電文要求

下位通信ソフトウェアに対して、受信した電文の受け渡しを要求する。

(1 4) アドレス情報取得

下位通信ソフトウェアに対して、保持している MAC アドレスやハウスコード情報を要求する。

(1 5) アドレス情報設定

下位通信ソフトウェアに対して、MAC アドレスやハウスコード情報を設定する。

(1 6) 物理アドレス変換要求

下位通信ソフトウェアに対して、NodeID を渡し、対応する下位通信ソフトウェアでの MAC アドレスを取得する。

(1 7) NodeID 変換要求

下位通信ソフトウェアに対して、MAC アドレスを渡し、対応する NodeID を要求する。

(1 8) 同報先取得要求

下位通信ソフトウェアに対して、同報種別指定コード（同報指定時における DEA の 1 バイト目の情報）がドメイン内同報（0x00）あるいは自サブネット内同報（0x01）の場合の同報対象指定コード（同報指定時における DEA の 2 バイト目の情報）を渡し、同報対象となる NodeID（下位通信ソフトウェア毎の同報のグループ指定に従って抽出した値）を要求する。

(1 9) 完全初期化要求

下位通信ソフトウェアに対し、コールドスタートによる初期化を実施後、通信停止状態に遷移することを要求する。このとき、ハウスコード情報、MAC アドレスは再取得される。

(2 0) 通信停止要求

下位通信ソフトウェアに対し、通信停止状態に遷移することを要求する。

(2 1) 完全停止要求

下位通信ソフトウェアに対し、停止状態に遷移することを要求する。

(2 2) 停止通知

下位通信ソフトウェアからプロトコル差異吸収処理部に対して、下位通信ソフトウェアが停止状態に遷移したことを通知する。

(2 3) 下位通信ソフトウェアアドレステーブルデータサイズ取得

下位通信ソフトウェアで保持している下位宛先アドレステーブルのデータ組数の取得を行う。

(2 4) 下位通信ソフトウェアアドレステーブルデータ取得

下位通信ソフトウェアで保持している下位宛先アドレステーブルデータの取得を行う。

(2 5) マスタルータ通知

自ノードがマスタルータであるか否かを下位通信ソフトウェアに通知する。

(2 6) ハードウェアアドレスデータ取得

下位通信ソフトウェアに対して保持しているハードウェアアドレスデータの取得を行う。

(2 7) ECHONET MAC アドレスリスト取得

下位通信ソフトウェアに対して保持している ECHONET MAC アドレスリストを取得する。

(2 8) マスタルータ情報取得

下位通信ソフトウェアに対して保持しているマスタルータ情報の取得を行う。

(2 9) ハードウェアアドレス変換要求

下位通信ソフトウェアに対して、渡した ECHONET MAC アドレス情報に対応するハードウェアアドレスを要求する。

第 3 章 レベル 1 ECHONET 個別下位通信インタフェース仕様

3.1 レベル 1 ECHONET 個別下位通信インタフェースのサービス一覧

レベル 1 ECHONET 個別下位通信インタフェースは、表 3.1 に示すサービス毎に、プロトコル差異吸収処理部と下位通信ソフトウェア間でやり取りされるデータについての規定を行う。本レベル 1 ECHONET 個別下位通信インタフェースに準拠した実装とするためには、次節で規定する入出力データ項目を備えている必要がある。ただし、複数のサービスを 1 つのサービスに統合したり、1 つのサービスを複数のサービスとして分割したりしてもかまわない。また、複数のデータ項目を 1 つのデータ項目として処理したり、或いは 1 つのデータ項目を複数のデータ項目として処理したりしてもかまわない。

表 3.1 レベル 1 ECHONET 個別下位通信インタフェース一覧表 (1 / 2)

No.	API 名称	機能	搭載規定
1	下位通信ソフトウェア種別要求	下位通信ソフトウェアの種類・ID を要求する。	Required
2	初期化要求	下位通信ソフトウェアのコールドスタートによる初期化を要求する。	Required
3	動作開始要求	下位通信ソフトウェアの動作開始を要求する。	Required
4	障害通知	プロトコル差異吸収処理部より上位層の障害 (異常) 状況を下位通信ソフトウェアへ通知する。	Optional
5	ウォームスタート要求	下位通信ソフトウェアに対して、ウォームスタートによる初期化を要求する。	Required
6	一時停止要求	下位通信ソフトウェアに対して、動作の一時停止を要求する。	Optional
7	動作再開要求	下位通信ソフトウェアに対して、動作の再開を要求する。	Optional
8	プロファイルデータ取得要求	下位通信ソフトウェアの静的情報を取得する。	Required
9	ステータスデータ取得要求	下位通信ソフトウェアの動的状態 (処理障害、アドレス重複等) に関する情報を取得する。	Required
10	電文送信要求	下位通信ソフトウェアに対して、電文の送信を要求する。	Required
11	送信結果取得	下位通信ソフトウェアに対して、電文の送信結果を要求する。	Optional
12	送信中止要求	下位通信ソフトウェア電文の送信の中止を要求する。	Optional
13	受信電文要求	下位通信ソフトウェアに対して、受信済みの電文の受け渡しを要求する。	Required
14	アドレス情報取得要求	下位通信ソフトウェアに対して、保持している MAC アドレスやハウスコード情報を要求する。	Required
15	アドレス情報設定要求	下位通信ソフトウェアに対して、MAC アドレスやハウスコード情報を設定する。	Optional
16	物理アドレス変換要求	下位通信ソフトウェアに NodeID を渡し、対応する MAC アドレスを要求する。	Optional

表 3 . 1 レベル 1 ECHONET 個別下位通信インタフェース一覧表 (2 / 2)

17	NodeID 変換要求	下位通信ソフトウェアに MAC アドレスを渡し、対応する NodeID を要求する。	Optional
18	同報先取得要求	下位通信ソフトウェアに、同報対象の NodeID を要求する。	Optional
19	完全初期化要求	下位通信ソフトウェアに対して、コールドスタートによる初期化を要求する。このとき、ハウスコード情報は再取得される。	Optional
20	通信停止要求	下位通信ソフトウェアに対して、通信停止状態への遷移を要求する。	Optional
21	完全停止要求	下位通信ソフトウェアに対して、停止状態への遷移を要求する。	Optional
22	停止通知	下位通信ソフトウェアからプロトコル差異吸収処理部に対し、下位通信ソフトウェアが停止状態に遷移した旨を通知する	Optional
23	下位通信ソフトウェアアドレステーブルデータサイズ取得	下位通信ソフトウェアで保持している下位アドレステーブルデータの組数を取得する。	Optional
24	下位通信ソフトウェアアドレステーブルデータ取得	下位通信ソフトウェアで保持している下位アドレステーブルデータを取得する。	Optional
25	マスターータ通知	自ノードがマスターータであるか否かを下位通信ソフトウェアに通知する。	Optional
26	ハードウェアアドレスデータ取得	下位通信ソフトウェアに対して保持しているハードウェアアドレスデータを取得する。	Optional
26	ECHONET MAC アドレスリスト取得	下位通信ソフトウェアが保持する ECHONET MAC アドレスリストを取得する。	Optional
27	マスターータ情報取得	下位通信ソフトウェアが保持するマスターータ情報を取得する。	Optional
28	ハードウェアアドレス変換要求	下位通信ソフトウェアに対して、渡した ECHONET MAC アドレス情報に対応するハードウェアアドレスを要求する。	Optional

3.2 レベル 1 ECHONET 個別下位通信インタフェースのサービス詳細仕様

前節の表 3.1 で示した各サービスにより、入出力されるデータに関しての規定を行う。以下の表中で、データの入出力の方向は、プロトコル差異収集処理部を基準とする。すなわち、「入力 (Input)」とは、プロトコル差異収集処理部から下位通信ソフトウェアに対してデータを渡すことを示し、「出力 (Output)」とは、下位通信ソフトウェアからプロトコル差異吸収処理へデータを渡すことを示す。レベル 1 ECHONET 個別下位通信インタフェースに準拠するためには、これらのデータの受け渡しが可能であればよく、受け渡し方法 (構造体を用いるとか、受け渡し用のバッファのポインタ情報を渡す等) については特に規定しない。

また、レベル 1 で示す ECHONET 個別下位通信インタフェースは、下位通信ソフトウェアの種類が変わっても同じ仕様とできることも考慮したデータの入出力をしめしている。その為、下位通信ソフトウェア種別要求以外のインタフェースで、Input として下位通信ソフトウェアの種類を示す引数を設定しているが、通常の通信装置 (下位通信ソフトウェアが複数存在しない通信装置) においては、特に指定が必要ないため、全て Optional 指定としている。

(1) 下位通信ソフトウェア種別要求 (搭載必須機能)

下位通信ソフトウェアの種別 (電灯線、小電力無線等) を要求する。表 3.2 に入出力仕様を示す。

表 3.2 下位通信ソフトウェア種別要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	-		
Output	device_id	<ul style="list-style-type: none"> ・ 下位通信ソフトウェアの種類を示す。 ・ 電灯線下位通信ソフトウェア 特定小電力無線下位通信ソフトウェア, 拡張 HBS 下位通信ソフトウェア, LonTalk[®]依存下位通信ソフトウェア, IrDA 依存下位通信ソフトウェア等の識別が可能であること。 	Required
Output	Return Value	TRUE : 取得成功、FALSE : 取得失敗	Optional

(2) 初期化要求 (搭載必須機能)

指定された情報での下位通信ソフトウェアのコールドスタートによる初期化と、その後の通信停止状態への遷移を要求する。この要求に対応する一連の処理の過程で、MAC アドレス情報は再取得される。ハウスコード情報をもつ下位通信ソフトウェアの場合、ハウスコード情報は変化しない。表 3 . 3 に入出力仕様を示す。

表 3 . 3 初期化要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Input	sfholdtime	・ 送信電文最長保持時間情報。 ・ 下位通信ソフトウェアが送信電文を保持している最大時間。この時間内に電文を送信できなかった場合は、送信電文を破棄する。	Optional
Input	rfholdtime	・ 受信電文最長保持時間情報。 ・ 下位通信ソフトウェアが受信電文を保持している最大時間。この時間内に受信した電文をプロトコル差異吸収処理部に渡せない場合は、受信電文を破棄する。	Optional
Input	sfbuf	・ 下位通信ソフトウェアが、プロトコル差異吸収処理部より受け取る送信電文用のバッファサイズを設定する。	Optional
Input	rdbuf	・ 下位通信ソフトウェアが、プロトコル差異吸収処理部へ渡す受信電文用のバッファサイズを設定する。	Optional
Input	low_mode	・ テストモードやネットワーク上の電文モニタモード等の下位通信ソフトウェアの特殊な動作モードの指定を示す。	Optional
Input	mac_ad	・ 下位通信ソフトウェアに設定する MAC アドレスを示す。	Optional
Input	mac_len	・ 下位通信ソフトウェアに設定する MAC アドレスのサイズに関する情報を示す。	Optional
Input	housecode	・ 下位通信ソフトウェアに設定するハウスコード情報を示す。	Optional
Input	housecode_len	・ 下位通信ソフトウェアに設定するハウスコード情報のサイズに関する情報を示す。	Optional
Input	lowinit	・ 下位通信ソフトウェア毎に異なる初期化パラメータを示す。	Optional
Output	Return Value	TRUE : 初期化成功、FALSE : 初期化失敗	Optional

(3) 動作開始要求 (搭載必須機能)

下位通信ソフトウェアの動作開始を要求する。表 3.4 に入出力仕様を示す。

表 3.4 動作開始要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	Return Value	TRUE : 動作開始成功、FALSE : 動作開始失敗	Optional

(4) 障害通知

プロトコル差異吸収処理部より上位層の障害 (異常) 状態を下位通信ソフトウェアへ通知する。表 3.5 に入出力仕様を示す。

表 3.5 障害通知サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Input	htrouble_no	・ 障害 (異常) 状態を示すトラブル番号を通知する。	Required
Output	Return Value	TRUE : 通知成功、FALSE : 通知失敗	Optional

(5) ウォームスタート要求

下位通信ソフトウェアのウォームスタートによる初期化と、その後の通信停止状態への遷移を要求する。この要求に対応する一連の処理の過程でハウスコード情報、MAC アドレス情報は変化しない。表 3.6 に入出力仕様を示す。

表 3.6 ウォームスタート要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	Return Value	TRUE : ウォームスタート受理、FALSE : 受理不可	Optional

(6) 一時停止要求

下位通信ソフトウェアに対して一時停止状態への遷移を要求する。表 3.7 に入出力仕様を示す。

表 3.7 一時停止要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	Return Value	TRUE : 一時停止受理、FALSE : 受理不可	Optional

(7) 動作再開要求

下位通信ソフトウェアに対して一時停止状態を解除し、通常動作状態に遷移することを要求する。表 3.8 に入出力仕様を示す。

表 3.8 動作再開要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	Return Value	TRUE : 再開成功、FALSE : 再開不可(失敗含)	Optional

(8) プロファイルデータ取得要求 (搭載必須機能)

下位通信ソフトウェアのプロファイルデータを要求する。本サービスにて要求するプロファイルデータは、下位通信ソフトウェアのソフトウェア開発メーカのメーカコードやバージョン No.等の静的情報とする。表 3 . 9 に入出力仕様を示す。

表 3 . 9 プロファイル取得サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	mac_ad	・ 保持している MAC アドレスを示す。	Required
Output	housecode	・ 保持しているハウスコード情報を示す。	Optional
Output	version_No	・ 下位通信ソフトウェアのバージョン情報を示す。	Optional
Output	maker	・ メーカコードを示す。	Optional
Output	srlen	・ 送受信可能電文長を示す。	Optional
Output	broad	・ 同報機能の有無を示す。	Optional
Output	baud	・ 伝送レートを示す。	Optional
Output	chmac_info	・ MAC アドレスから NodeID への変換情報 (変換関数アドレス情報等) を示す。	Optional
Output	chnode_info	・ NodeID から MAC アドレスへの変換情報 (変換関数アドレス情報等) を示す。	Optional
Output	chbroad_info	・ NodeID から同報先の MAC アドレスへの変換情報 (変換関数アドレス情報等) を示す。	Optional
Output	Return Value	TRUE : 正常時、FALSE : 異常時	Optional

(9) ステータスデータ取得要求 (搭載必須機能)

下位通信ソフトウェアに対して、ステータスデータを要求する。本サービスにて要求するステータスデータは、異常状態や処理状態といった動的情報とする。表 3 . 10 に入出力仕様を示す。

表 3 . 10 ステータス取得サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	state	・ 下位通信ソフトウェアの遷移状態情報。 ・ 第 3 部で下位通信ソフトウェアごとに規定されている状態を識別できること。	Required
Output	upper_trouble	・ 上位層障害として認識している情報。	Optional
Output	low_trouble	・ 下位通信ソフトウェアの障害として認識している情報を示す。	Optional
Output	low_mode	・ 動作モード (モニタモードやテストモード等) 情報を示す。	Optional
Output	Return Value	TRUE : 正常時、FALSE : 異常時	Optional

(1 0) 電文送信要求 (搭載必須機能)

下位通信ソフトウェアに対して、指定する ECHONET 電文の送信を要求する。表 3 . 1 1 に入出力仕様を示す。

表 3 . 1 1 電文送信要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Input	send_data	・ ECHONET 電文形式の送信要求データ情報。 ・ 形式はプロトコル差異吸収処理部間でやり取りされるものである。	Required
Input	d_add	・ サブネット内送信先 MAC アドレスを示す。	Required
Input	mac_len	・ サブネット内送信先 MAC アドレスサイズを示す。	Optional
Input	broad	・ 同報 / 個別の指定情報を示す。	Optional
Output	Return Value	TRUE : 正常時、FALSE : 異常時	Optional

(1 1) 送信結果取得要求

下位通信ソフトウェアに対して、「電文送信要求」で要求した電文の送信結果を要求する。表 3 . 1 2 に入出力仕様を示す。

表 3 . 1 2 送信結果取得サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	result	・ 送信中、送信正常終了、送信エラー終了、或いは送信中止中の情報を示す。	Required
Output	Return Value	TRUE : 正常時、FALSE : 異常時	Optional

(1 2) 送信中止要求

下位通信ソフトウェアが実行中の電文送信処理の中止を要求する。表 3 . 1 3 に入出力仕様を示す。

表 3 . 1 3 送信中止要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	Return Value	TRUE : 中止成功、FALSE : 中止失敗(送信済)	Optional

(1 3) 受信電文要求 (搭載必須機能)

下位通信ソフトウェアが受信している電文を要求する。表 3 . 1 4 に入出力仕様を示す。

表 3 . 1 4 電文受信要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	receive_data	・ ECHONET 電文形式の受信データを示す。 ・ 形式はプロトコル差異処理部間でやり取りされるものである。	Required
Output	s_add	・ サブネット内送信元 MAC アドレスを示す。	Required
Output	mac_len	・ サブネット内送信元 MAC アドレスサイズ情報を示す。	Optional
Output	Return Value	TRUE : 正常時、FALSE : 異常時(受信データ無しの指示等異常指示コード)	Optional

(1 4) アドレス情報取得要求 (搭載必須機能)

下位通信ソフトウェアが保持しているアドレス情報を要求する。表 3 . 1 5 に入出力仕様を示す。

表 3 . 1 5 アドレス情報取得サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	mac_ad	・ 保持している MAC アドレスを示す。	Required
Output	mac_len	・ MAC アドレスのサイズ情報を示す。	Optional
Output	houscode	・ 保持しているハウスコード情報を示す。	Optional
Output	houscode_len	・ ハウスコード情報のサイズを示す。	Optional
Output	Return Value	TRUE : 正常時、FALSE : 異常時(nodeID 未設定や指定 device_id 異常等指示コード)	Optional

(1 5) アドレス情報設定要求

下位通信ソフトウェアに対して、アドレス情報を設定する。表 3 . 1 6 に入出力仕様を示す。

表 3 . 1 6 アドレス設定サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Input	mac_ad	・ 設定する MAC アドレスを示す。	Required
Input	mac_len	・ MAC アドレスサイズ情報を示す。	Optional
Input	houscode	・ 設定するハウスコード情報を示す。	Optional
Input	houscode_len	・ ハウスコード情報のサイズを示す。	Optional
Output	Return Value	TRUE : 正常時、FALSE : 異常時(設定不可等指示コード)	Optional

(1 6) 物理アドレス変換要求

下位通信ソフトウェアに対して、NodeID を渡し、対応する下位通信ソフトウェアでの MAC アドレスを要求する。表 3 . 1 7 に入出力仕様を示す。

表 3 . 1 7 物理アドレス変換要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Input	node_id	・ 変換を要求する NodeID を示す。	Required
Output	mac_ad	・ 指定した NodeID に対応する MAC アドレスを示す。	Required
Output	mac_len	・ MAC アドレスサイズ情報を示す。	Optional
Output	Return Value	TRUE : 正常時、FALSE : 異常時(変換不可等指示コード)	Optional

(1 7) NodeID 変換要求

下位通信ソフトウェアに対して、MAC アドレスを渡し、対応する NodeID (下位通信ソフトウェア毎の変換規則に従って変換した値) を要求する。表 3 . 1 8 に入出力仕様を示す。

表 3 . 1 8 NodeID 変換要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Input	mac_ad	・ 変換を要求する MAC アドレス情報を示す。	Required
Output	mac_len	・ MAC アドレスサイズ情報。	Optional
Output	node_id	・ 指定した MAC アドレスに対応する NodeID を示す。	Required
Output	Return Value	TRUE : 正常時、FALSE : 異常時(変換不可等指示)	Optional

(1 8) 同報先取得要求

下位通信ソフトウェアに対して、同報種別指定コード (同報指定時における DEA の 1 バイト目の情報) がドメイン内同報 (0x00) あるいは自サブネット内同報 (0x01) の場合の同報対象指定コード (同報指定時における DEA の 2 バイト目の情報) を渡し、同報対象となる NodeID 情報 (下位通信ソフトウェア毎の同報のグループ指定に従って抽出した値) を要求する。表 3 . 1 9 に入出力仕様を示す。

表 3 . 1 9 同報先取得要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Input	broad_adinfo	・ 同報対象指定コードを示す。	Required
Output	node_num	・ 同報対象となる NodeID の数を示す。	Required
Output	node_idinfo	・ 同報対象となる NodeID 情報を示す。	Required
Output	Return Value	TRUE : 正常時、FALSE : 異常時(変換不可等指示コード)	Optional

(1 9) 完全初期化要求

下位通信ソフトウェアに対して、コールドスタートによる初期化と、その後の通信停止状態への遷移を要求する。この要求に対応する一連の処理の過程でハウスコード情報、MAC アドレス情報は再取得される。

表 3 . 2 0 完全初期化要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	software_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Required
Input	p_init	・ 初期化パラメータを指定する。 ・ 送信電文最長保持時間、受信電文最長保持時間等含むが初期化対象下位通信ソフトウェアにより具体的な内容は異なる。	Required
Output	Return Value	TRUE : 初期化成功、FALSE : 初期化失敗	Optional

(2 0) 通信停止要求

下位通信ソフトウェアに対して、通信停止状態への遷移を要求する。表 3 . 2 1 に入出力仕様を示す。

表 3 . 2 1 一時停止要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	software_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Required
Output	Return Value	TRUE : 一時停止受理、FALSE : 受理不可	Optional

(2 1) 完全停止要求

下位通信ソフトウェアに対して、停止状態への遷移を要求する。表 3 . 2 2 に入出力仕様を示す。

表 3 . 2 2 完全停止要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	software_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Required
Output	Return Value	TRUE : 一時停止受理、FALSE : 受理不可	Optional

(2 2) 停止通知

下位通信ソフトウェアが停止状態へ遷移したことを ECHONET 通信処理部に通知する。
表 3 . 2 3 に入出力仕様を示す。

表 3 . 2 3 停止通知サービス入出力データ一覧

方向	データ名	内容・条件	備考
Output	software_id	・ 停止状態に遷移した下位通信ソフトウェアを示す。	Required
Input	Return Value	TRUE : 通知受理、 FALSE : 受理不可	Optional

(2 3) 下位通信ソフトウェアアドレステーブルデータサイズ取得(Optional)

下位通信ソフトウェアが下位アドレステーブルで保持しているデータ組数の取得を行う。
表 3 . 2 4 に入出力仕様を示す。

表 3 . 2 4 下位通信ソフトウェアアドレステーブルデータサイズ取得サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	data_number	・ 下位アドレステーブルデータで保持しているアドレステーブルの組数を示す。	Required
Output	Return Value	TRUE : 正常時、 FALSE : 異常時	Optional

(2 4) 下位通信ソフトウェアアドレステーブルデータ取得 (Optional)

下位通信ソフトウェアで保持している下位アドレステーブルデータを取得する。出力データはデータ組数とデータ組数分のデータ組からなる。各データ組はハードウェアアドレス、MAC アドレス、及びマスタールータであることを示すフラグよりなる。

表 3 . 2 5 下位通信ソフトウェアアドレステーブルデータ取得サービス
入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	data_number	・ 下位アドレステーブルデータで保持しているアドレステーブルの組数を示す。	Required
Output	ListOfHardwareaddresses	・ 下位通信ソフトウェアのアドレステーブルで保持しているハードウェアアドレスのリストを示す。	Required
Output	ListOfMac_ad	・ 下位通信ソフトウェアのアドレステーブルで保持している MAC アドレスのリストを示す。	Required
Output	ListOfMasterRouter_Flag	・ 下位通信ソフトウェアのアドレステーブルで保持しているアドレスに対応するノードがマスタールータであるか否かを示す識別子のリスト。マスタールータであれば 1、そうでなければ 0 となる。	Required
Output	Return Value	TRUE : 正常時、FALSE : 異常時	Optional

(2 5) マスタルータ通知 (Optional)

自ノードがマスタルータであるか否かを下位通信ソフトウェアに設定する。

表 3 . 2 6 マスタルータ通知サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Input	masterRouter_Flag	・ マスタルータであれば 1、そうでなければ 0 を指定する。	Required
Output	Return Value	TRUE : 正常時、FALSE : 異常時	Optional

(2 6) ハードウェアアドレスデータ取得 (Optional)

下位通信ソフトウェアに対して保持しているハードウェアアドレスデータを取得する。

表 3 . 2 7 ハードウェアアドレスデータ取得サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	Device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	hardwareaddress	・ 自ノードのハードウェアアドレスを示す。	Required
Output	Return Value	TRUE : 正常時、FALSE : 異常時	Optional

(2 7) ECHONET MAC アドレスリスト取得

下位通信ソフトウェアが保持する ECHONET MAC アドレスリスト取得する。

表 3 . 2 8 ECHONET MAC アドレスリスト取得サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	mac_addr_list	・ 下位通信ソフトウェアが保持している MAC アドレスリストを示す。	Required

(2 8) マスタルータ情報取得

下位通信ソフトウェアが保持するマスタルータリストを取得する。

表 3 . 2 9 マスタルータ情報取得サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Output	result,	・ マスタルータの存在の有無	Required
Output	master_node_id	・ マスタルータの NodeID	

(2 9) ハードウェアアドレス変換要求

下位通信ソフトウェアに対して、渡した NodeID に対応するハードウェアアドレスを要求する。

表 3 . 3 0 ハードウェアアドレス変換要求サービス入出力データ一覧

方向	データ名	内容・条件	備考
Input	device_id	・ 下位通信ソフトウェア種別要求サービスにより取得した下位通信ソフトウェア ID を指定する。	Optional
Input	mac_len	・ 変換対象の MAC アドレス長	Required
Input	mac	・ 変換対象の MAC アドレスのポインタ	
Output	hardwareaddress	・ ハードウェアアドレスへのポインタを示す。	Required
Output	hardwareaddress_len	・ ハードウェアアドレスのサイズへのポインタを示す。	Required

第 4 章 レベル 2 ECHONET 個別下位通信インタフェース仕様

本章では、レベル 2 ECHONET 個別下位通信インタフェースとして、本インタフェースを利用して開発するソフトウェアの互換性を考慮した API の詳細規定を示す。本章で示す規定は、下位通信ソフトウェアに API 処理を搭載する（プロトコル差異吸収処理部にて下位通信ソフトウェアの処理を呼び出す）形式とした。

レベル 2 ECHONET 個別下位通信インタフェースについて、ECHONET 規格 Version 2.10 においては、ANSI 規格の C 言語を対象としたもの（以下「C 言語用」と呼ぶ）を規定する。

4.1 レベル 2 ECHONET 個別下位通信インタフェース一覧

C 言語用レベル 2 ECHONET 個別下位通信インタフェースの関数として、以下の 29 個のものを規定する。レベル 2 の Optional とは、関数として実装しなくてもよいことを示すが、その機能を実現している場合には、レベル 2 準拠とするには本項にて示す関数を実現するものとする。

表 4.1 レベル 2 ECHONET 個別下位通信インタフェース関数一覧表 (1 / 2)

No.	API 名称	API 関数名	機能	備考
1	下位通信ソフトウェア種別要求	LowGetDevID	下位通信ソフトウェアの種類・ID を要求する。	Required
2	初期化要求	LowInit	下位通信ソフトウェアの初期化を要求する。	Required
3	動作開始要求	LowRequestRun	下位通信ソフトウェアの動作開始を要求する。	Required
4	障害通知	LowSetTrouble	プロトコル差異吸収処理部より上位層の障害(異常)状況を下位通信ソフトウェアへ通知する。	Optional
5	ウォームスタート要求	LowStart	下位通信ソフトウェアに対して、ウォームスタート処理を要求する。	Required
6	一時停止要求	LowSuspend	下位通信ソフトウェアに対して、動作の一時停止を要求する。	Optional
7	動作再開要求	LowWakeUp	下位通信ソフトウェアに対して、動作の再開を要求する。	Optional
8	プロファイルデータ取得要求	LowGetProData	下位通信ソフトウェアのプロファイルデータ(静的情報)を取得する。	Required
9	ステータスデータ取得要求	LowGetStatus	下位通信ソフトウェアの動的状態(処理障害、アドレス重複等)情報を取得する。	Required
10	電文送信要求	LowSendData	下位通信ソフトウェアに対して、電文の送信を要求する。	Required
11	送信結果取得	LowGetSendResult	下位通信ソフトウェアに対して、電文の送信結果を要求する。	Optional
12	送信中止要求	LowSendCancel	下位通信ソフトウェア電文の送信の中止を要求する。	Optional
13	受信電文要求	LowReceiveData	下位通信ソフトウェアに対して、受信済みの電文の受け渡しを要求する。	Required
14	アドレス情報取得要求	LowGetAddress	下位通信ソフトウェアで認識している Mac アドレスやハウスコード等のアドレス情報を取得する。	Required

表 4. 1 レベル 2 ECHONET 個別下位通信インタフェース関数一覧表 (2 / 2)

No.	API 名称	API 関数名	機能	備考
15	アドレス情報設定 要求	LowSetAddress	下位通信ソフトウェアに対して Mac アドレスやハウスコード等のアドレス情報を設定する。	Optional
16	物理アドレス変換 要求	LowReqToMac	NodeID から対応する Mac アドレスへの変換を要求する。	Optional
17	NodeID 変換要求	LowReqToID	Mac アドレスから対応する NodeID への変換を要求する。	Optional
18	同報先アドレス取 得要求	LowReqBcastID	同報対象の NodeID を要求する。	Optional
19	完全初期化要求	LowInitAll	下位通信ソフトウェアのハウスコード情報の再取得をともなう初期化を要求する。	Optional
20	通信停止要求	LowStop	下位通信ソフトウェアに対して、通信停止を要求する。	Optional
21	完全停止要求	LowHalt	下位通信ソフトウェアに対して、完全停止を要求する。	Optional
22	停止通知受信	LowReceiveStop	下位通信ソフトウェアに対して、停止通知の受け渡しを要求する。	Required
23	下位通信ソフトウ ェアアドレステー ブルデータサイズ 取得	LowGetAddressTab leDataSize	下位通信ソフトウェアで保持している下位アドレステーブルのデータの組数の取得を行う。	Optional
24	下位通信ソフトウ ェアアドレステー ブルデータ取得	LowGetAddressTab leData	下位通信ソフトウェアで保持している下位アドレステーブルデータの取得を行う。	Optional
25	マスタールータ通知	LowSetMasterRout erFlag	自ノードがマスタールータであるか否かを下位通信ソフトウェアに通知する。	Optional
26	ハードウェアアド レスデータ取得	LowGetHardwareA ddress	下位通信ソフトウェアに対して自ノードのハードウェアアドレスデータの取得を行う。	Optional
27	ECHONET MAC アドレスリスト取 得	LowGetEchonetMA CList	下位通信ソフトウェアが保持する ECHONET MAC アドレスリストを取得する。	Optional
28	マスタールータ情報 取得	LowGetMasterRout erInfo	下位通信ソフトウェアが保持するマスタールータ情報を取得する。	Optional
29	ハードウェアアド レス変換要求	LowReqToHardwar eAddress	下位通信ソフトウェアに対して渡した ECHONET MAC アドレス情報に対応するハードウェアアドレスを要求する。	Optional

4.2 レベル2 ECHONET 個別下位通信インタフェース詳細仕様

本節では、表4.1にて示した各関数の詳細仕様を、以下の7つの項目について示す。

- (1) 名称
関数名称を示す。
- (2) 機能
機能を説明する。
- (3) 構文
関数の構文を示す。
- (4) 説明
引数や、変数の詳細仕様について説明する。
- (5) 戻り値
戻り値を示す。
- (6) 使用する構造体
構造体があれば、構造体の仕様を示す。
- (7) 注意事項・制限事項
注意事項や制限事項があれば示す。

4 . 2 . 1 LowGetDevID

(1) 名称

下位通信ソフトウェア種別要求関数

(2) 機能

下位通信ソフトウェアの種類を示す下位通信ソフトウェア ID 要求する。

(3) 構文

```

BOOL LowGetDevID (
    unsigned char *device_id    /*[OUT] 下位通信ソフトウェア ID */
)

```

(4) 説明

device_id	: 下位通信ソフトウェア ID
電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk ^R	0x61 ~ 0x6F
Bluetooth TM	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

(5) 戻り値

0 : 取得失敗
1 : 取得成功

(6) 使用する構造体

特に無し。

(7) 注意事項・制限事項

この関数は、「初期化要求関数 : LowInit」や「動作開始要求関数 : LowRequestRun」より前に呼び出されることを前提とする。

4 . 2 . 2 LowInit

(1) 名称

初期化要求関数

(2) 機能

下位通信ソフトウェアに対して、MAC アドレス再取得を伴う初期化 (コールドスタート) を要求する。この要求を受けた下位通信ソフトウェアは、コールドスタートを行い、通信停止状態に遷移した後、初期化パラメータを自身に設定する。

(3) 構文

```

BOOL LowInit (
    unsigned char    device_id,          /* [IN] 下位通信ソフトウェア ID */
    LOW_INIT_DATA *init_data,          /* [IN] 初期化パラメータへのポインタ(1) */
    void *low_init   /* [IN] 初期化パラメータへのポインタ(2) */
)
    
```

(4) 説明

device_id	: 下位通信ソフトウェア ID	電灯線	0x11 ~ 0x1F
		特定小電力無線	0x31 ~ 0x3F
		拡張 HBS	0x41 ~ 0x4F
		IrDA_Control	0x51 ~ 0x5F
		LonTalk ^R	0x61 ~ 0x6F
		Bluetooth TM	0x71 ~ 0x7F
		Ethernet	0x81 ~ 0x8F

***init_data** : 共通規定項目の初期化パラメータへのポインタ。

***low_init** : 個々の下位通信ソフトウェア毎に異なる初期化パラメータへのポインタ。パラメータ内容は、個別下位通信ソフトウェア毎に規定する (4 . 3 節参照)

(5) 戻り値

0 : 初期化失敗
 1 : 初期化成功

(6) 使用する構造体

```

typedef struct {
    short          sfholdtime,          /* 送信電文最長保持時間情報 */
    short          rfholdtime,          /* 受信電文最長保持時間情報 */
    unsigned char  low_mode,           /* 動作モード指定 */
                                     0x00 通常動作モード。
                                     0x01 テスト・メンテナンスモード
                                     (詳細は、特に規定しない)
    short          mac_len,             /* MAC アドレス長 */
    unsigned char  mac_ad[6],          /* MAC アドレス */
} LOW_INIT_DATA
    
```

mac_ad[6]以外は、特に初期化データが無い場合には、NULL を設定する。

mac_ad[6]は、mac_len が NULL 設定の場合には、意味が無いものとする。(mac_len が NULL の場合には、MAC アドレス設定無しとする。)

(7) 注意事項・制限事項

本関数は、すでに下位通信ソフトウェアがコールドスタート状態、またはウォームスタート状態に

ある場合は、初期化失敗を戻り値とする。

4 . 2 . 3 LowRequestRun

(1) 名称

動作開始要求関数

(2) 機能

下位通信ソフトウェアの動作開始を要求する。この要求を受けた下位通信ソフトウェアは通常動作状態に遷移する。

(3) 構文

```

BOOL LowRequestRun (
    unsigned char device_id    /*[IN] 下位通信ソフトウェア ID */
)

```

(4) 説明

device_id	: 下位通信ソフトウェア ID	電灯線	0x11 ~ 0x1F
		特定小電力無線	0x31 ~ 0x3F
		拡張 HBS	0x41 ~ 0x4F
		IrDA_Control	0x51 ~ 0x5F
		LonTalk ^R	0x61 ~ 0x6F
		Bluetooth TM	0x71 ~ 0x7F
		Ethernet	0x81 ~ 0x8F

(5) 戻り値

0 : 起動失敗
1 : 起動成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

下位通信ソフトウェアが通信停止状態にない場合、本関数の戻り値は起動失敗となる。

4 . 2 . 4 LowSetTrouble

(1) 名称

障害通知関数

(2) 機能

プロトコル差異吸収処理部より上位層の障害（異常）状態を、下位通信ソフトウェアへ通知する。

(3) 構文

```

BOOL LowSetTrouble (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    char htrouble_no /*[IN] 上位レイヤトラブル番号 */
)

```

(4) 説明

device_id	: 下位通信ソフトウェア ID	電灯線	0x11 ~ 0x1F
		特定小電力無線	0x31 ~ 0x3F
		拡張 HBS	0x41 ~ 0x4F
		IrDA_Control	0x51 ~ 0x5F
		LonTalk ^R	0x61 ~ 0x6F
		Bluetooth TM	0x71 ~ 0x7F
		Ethernet	0x81 ~ 0x8F

htrouble_no : トラブル番号。

- 1 トラブル解消
- 1 アプリケーションソフト異常
- 2 ECHONET 通信処理部異常
- 3 プロトコル差異吸収処理部異常

(5) 戻り値

0 : 通知失敗
1 : 通知成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

異常が通知されている間は、下位通信ソフトウェアでは以下の動作を行う。

・電文受信時処理

電文受信処理を行わないか、或いは受信電文を廃棄する。

・プロトコル差異吸収処理部からの電文送信要求

エラーリターンする。

4 . 2 . 5 LowStart

(1) 名称

ウォームスタート要求関数

(2) 機能

下位通信ソフトウェアに対して、MAC アドレスを保持しての初期化 (ウォームスタート) を要求する。この要求を受けた下位通信ソフトウェアはウォームスタートし、通信停止状態に遷移する。

(3) 構文

```

BOOL LowStart (
    unsigned char device_id /*[IN] 下位通信ソフトウェア ID */
)

```

(4) 説明

device_id	: 下位通信ソフトウェア ID	電灯線	0x11 ~ 0x1F
		特定小電力無線	0x31 ~ 0x3F
		拡張 HBS	0x41 ~ 0x4F
		IrDA_Control	0x51 ~ 0x5F
		LonTalk ^R	0x61 ~ 0x6F
		Bluetooth TM	0x71 ~ 0x7F
		Ethernet	0x81 ~ 0x8F

(5) 戻り値

0 : 要求失敗
1 : 要求成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、すでに下位通信ソフトウェアがコールドスタート状態、またはウォームスタート状態にある場合は、要求失敗を戻り値とする。

本要求を受けた時のウォームスタート処理として以下の処理を行う。

- ・送受信バッファのクリア
- ・上位レイヤ障害設定のリセット
- ・各種ステータス・ワーク領域のリセット
- ・通信ハードウェア部のリセット

4 . 2 . 6 LowSuspend

(1) 名称

一時停止要求関数

(2) 機能

下位通信ソフトウェアの一時停止を要求する。この要求を受けた下位通信ソフトウェアは一時停止状態に遷移する。

(3) 構文

```

BOOL LowSuspend (
    unsigned char device_id /*[IN] 下位通信ソフトウェア ID */
)

```

(4) 説明

device_id	: 下位通信ソフトウェア ID	電灯線	0x11 ~ 0x1F
		特定小電力無線	0x31 ~ 0x3F
		拡張 HBS	0x41 ~ 0x4F
		IrDA_Control	0x51 ~ 0x5F
		LonTalk ^R	0x61 ~ 0x6F
		Bluetooth TM	0x71 ~ 0x7F
		Ethernet	0x81 ~ 0x8F

(5) 戻り値

0 : 一時停止失敗
1 : 一時停止成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、下位通信ソフトウェアが通常動作以外の状態にあるときは、一時停止失敗を戻り値とする。

下位通信ソフトウェアは、本要求を受けた時に電文送信途中であれば、一連の送信処理を終了し、一時停止状態に遷移する。また、電文の受信途中であれば、受信電文を廃棄し、処理を終了する。

一時停止中には以下の動作を行う。

- ・電文の受信
行わない。
- ・ECHONET 通信制御処理部からの電文送信要求
エラーリターンする。

4 . 2 . 7 LowWakeUp

(1) 名称

動作再開要求関数。

(2) 機能

下位通信ソフトウェア一時停止解除を要求する。この要求を受けた下位通信ソフトウェアは通常動作状態に遷移する。

(3) 構文

```

BOOL LowWakeUp (
    unsigned char device_id /*[IN] 下位通信ソフトウェア種別 ID */
)
    
```

(4) 説明

device_id	: 下位通信ソフトウェア ID	電灯線	0x11 ~ 0x1F
		特定小電力無線	0x31 ~ 0x3F
		拡張 HBS	0x41 ~ 0x4F
		IrDA_Control	0x51 ~ 0x5F
		LonTalk ^R	0x61 ~ 0x6F
		Bluetooth TM	0x71 ~ 0x7F
		Ethernet	0x81 ~ 0x8F

(5) 戻り値

0 : 再開失敗
 1 : 再開成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、下位通信ソフトウェアが一時停止状態以外の状態にあるときは、再開失敗を戻り値とする。

4 . 2 . 8 LowGetProData

(1) 名称

プロファイルデータ取得要求関数

(2) 機能

下位通信ソフトウェアのプロファイルデータと、下位通信ソフトウェアが保持する特殊処理関数のアドレスを取得する。本関数にて要求するプロファイルデータは、ソフトウェア開発メーカー名やバージョン No.等の下位通信ソフトウェアプロファイルクラスのプロパティ値情報である。

(3) 構文

```

BOOL LowGetProData (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    LOW_PRO_DATA *pro_data,
                                /*[OUT] プロファイルデータ*/
    short (**chmacfunc) (unsigned char node_id, unsigned char *mac),
                                /*[OUT] NodeID MAC アドレス変換関数アドレス*/
    unsigned char (**chnodefunc) (unsigned char *mac),
                                /*[OUT] MAC アドレス NodeID 変換関数アドレス*/
    void(**broadfunc) (const char bcast, char map[32])
                                /*[OUT] 同報先取得関数アドレス*/
)
    
```

(4) 説明

device_id	: 下位通信ソフトウェア ID	電灯線	0x11 ~ 0x1F
		特定小電力無線	0x31 ~ 0x3F
		拡張 HBS	0x41 ~ 0x4F
		IrDA_Control	0x51 ~ 0x5F
		LonTalk ^R	0x61 ~ 0x6F
		Bluetooth TM	0x71 ~ 0x7F
		Ethernet	0x81 ~ 0x8F

***pro_data** : 下位通信ソフトウェアのプロファイルデータ構造体へのポインタ。

****chmacfunc** : NodeID から下位通信ソフトウェア固有の MAC アドレスへの変換関数のアドレスへのポインタが返される。NodeID と MAC アドレスが一致している下位通信ソフトウェアや、単純な線形変換を行う下位通信ソフトウェアでは NULL が返される。
渡される関数の引数の仕様は以下のとおりである。

node_id : [in] 変換前 NodeID

mac : [out] 変換後の MAC アドレス

この関数は、戻り値として MAC アドレスのサイズ (バイト数) を返す。

****chnodefunc** : 下位通信ソフトウェア固有の MAC アドレスから NodeID への変換関数のアドレスへのポインタが返される。NodeID と MAC アドレスが一致している下位通信ソフトウェアや、単純な線形変換を行う下位通信ソフトウェアでは NULL が返される。
渡される関数の引数の仕様は以下のとおり。

mac : [in] 変換前の MAC アドレス

この関数は戻り値として変換後の NodeID を返す。

****broadfunc** : 同報先取得関数のアドレスへのポインタが返される。
同報機能のある下位通信ソフトウェアでは、NULL が返される。
渡される関数の引数の仕様は以下のとおり。

bcast : [in] ドメイン内同報、あるいは自サブネット内同報指定時の同報対象指定コード。
map[32] : [out] 同報先 NodeID のビットマップへの配列を返す。同報先 NodeID とビットの対応を以下に示す。

map[0]-bit0 : NodeID 0 (0x00)
map[0]-bit1 : NodeID 1 (0x01)

.....
map[1]-bit0 : NodeID 8 (0x08)
map[1]-bit1 : NodeID 9 (0x09)

.....
map[31]-bit7 : NodeID 255 (0xFF)

(5) 戻り値

0 : 取得失敗
1 : 取得成功

(6) 使用する構造体

```
typedef struct {
    unsigned char ver[3];          /* 下位通信ソフトウェアバージョン No. */
    unsigned char maker[3];       /* メーカーコード */
    short mac_len;                /* MAC アドレス長 */
    unsigned char mac_ad[6];      /* MAC アドレス */
    unsigned char mac_mask[6];    /* MAC アドレスマスク値 */
    short house_len;             /* ハウスコード長 */
    short *housecode;            /* ハウスコード情報へのポインタ */
    short slen;                  /* 送信可能電文長 */
    short rlen;                  /* 受信可能電文長 */
    short broad;                 /* 同報機能の有無(0 : 機能無、1 : 機能有) */
    short baud;                  /* 伝送レート */
} LOW_PRO_DATA
```

(7) 注意事項・制限事項

特記無し。

4 . 2 . 9 LowGetStatus

(1) 名称

ステータスデータ取得要求関数

(2) 機能

下位通信ソフトウェアに対して、下位通信ソフトウェアのステータスデータを要求する。
本機能にて取得できるステータスデータは、異常状態や処理状態といった動的情報である。

(3) 構文

```

BOOL LowGetStatus (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    LOW_STATUS *status /*[OUT] 下位通信ソフトのステータス */
)
    
```

(4) 説明

device_id	: 下位通信ソフトウェア ID	電灯線	0x11 ~ 0x1F
		特定小電力無線	0x31 ~ 0x3F
		拡張 HBS	0x41 ~ 0x4F
		IrDA_Control	0x51 ~ 0x5F
		LonTalk ^R	0x61 ~ 0x6F
		Bluetooth TM	0x71 ~ 0x7F
		Ethernet	0x81 ~ 0x8F

*status : ステータスデータ構造体へのポインタが返される。

(5) 戻り値

0 : 取得失敗
1 : 取得成功

(6) 使用する構造体

```

typedef struct {
    char upper_trouble; /* 上位レイヤ障害コード(0 ~ 127)
                        障害無し及びトラブル解消時は(0) */
    char low_trouble; /* 下位通信ソフトウェア部障害コード(0 ~ 127)
                      障害無し及びトラブル解消時は(0) */
    char low_mode; /* 動作モードコード
                  通常動作時は(0)
                  メンテナンス等テストモード時は(1)
                  モニタモード時は(2) */
    short state; /* 下位通信ソフトウェア部状態
                 LOW_STS_STOP : 0 停止状態
                 LOW_STS_INI : 1 コールドスタート状態
                 LOW_STS_RUN : 2 通常動作状態
                 LOW_STS_ESTOP : 3 エラー停止状態
                 LOW_STS_RST : 4 ウォームスタート状態
                 LOW_STS_CSTOP : 5 通信停止状態
                 LOW_STS_SPD : 6 一時停止状態
    } LOW_STATUS;
    
```

(7) 注意事項・制限事項

特記無し。

4 . 2 . 1 0 LowSendData

(1) 名称

電文送信要求関数

(2) 機能

下位通信ソフトウェアに ECHONET 電文を送信することを要求する。

(3) 構文

```
short LowSendData (
    unsigned char    device_id, /*[IN] 下位通信ソフトウェア種別 ID */
    const unsigned char *buf, /*[IN] 送信データへのポインタ */
    short snd_sz, /*[IN] 送信データサイズ */
    const unsigned char *da, /*[IN] 送信先物理アドレス */
    unsigned char broad, /*[IN] 同報指定 */
)
```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

*buf : 送信する ECHONET 電文へのポインタを指定する。ここで渡される ECHONET 電文は、第 2 部 4 . 2 節で規定されているプロトコル差異吸収処理部間でやり取りされるものである。

snd_sz : 送信データサイズを指定する。

*da : 自サブネット内送信先 MAC アドレスへのポインタを指定する。
broad がドメイン内一斉同報、あるいは自サブネット内同報の時には、このパラメータは使用されず、下位通信ソフトウェアにおける一斉同報が行われる。

broad : 同報指定を行う。

0x00 : 同報指定無し、または指定サブネット内一斉同報

0xFF : ドメイン内同報、または自サブネット内同報

(5) 戻り値

LOW_BUFFER_FULL(0)	: バッファフルエラー
LOW_NO_ERROR(1)	: 送信受理
LOW_BUFFER_SIZE_ERROR(2)	: バッファサイズエラー
LOW_STATE_ERROR(3)	: 下位通信ソフトウェア内部エラー

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、指定された下位通信ソフトウェアが通常動作状態にない場合は、下位通信ソフトウェア内部エラーを戻り値とする。

4 . 2 . 1 1 LowGetSendResult

(1) 名称

送信結果取得要求関数

(2) 機能

下位通信ソフトウェアが、電文送信関数 (ClcSendData) に応じて最近に行った ECHONET 電文の送信結果を要求する。

(3) 構文

```
short LowGetSendResult (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    unsigned char *result /*[OUT] 送信結果 */
)
```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

result : 送信結果 0x00:送信成功、0x01:送信失敗、0xFF:無応答

(5) 戻り値

LOW_CANCEL(0)	: 送信中止時
LOW_NO_ERROR(1)	: 正常時
LOW_NO_SENDEND(2)	: 送信中 (送信未完了)
LOW_INTERNAL_ERROR(3)	: 下位通信ソフトウェア内部エラー

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、下位通信ソフトウェアが通常動作状態にない場合は、下位通信ソフトウェア内部エラーを戻り値とする。

result が意味を持つのは、戻り値が正常時 (NO_ERROR) の場合のみである。

4 . 2 . 1 2 LowSendCancel

(1) 名称

送信中止要求関数

(2) 機能

下位通信ソフトウェアが、電文送信関数(ClcSendData)に応じて行っている ECHONET 電文の送信中心を要求する。

(3) 構文

```
unsigned char    LowSendCancel (
    unsigned char    device_id    /*[IN]  下位通信ソフトウェア ID */
)
```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

(5) 戻り値

LOW_CANCEL(0) : 送信済みにより中止処理実行無し
 LOW_NO_ERROR(1) : 正常時
 LOW_INTERNAL_ERROR(3) : 下位通信ソフトウェア内部エラー

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、下位通信ソフトウェアが通常動作状態にない場合は、下位通信ソフトウェア内部エラーを戻り値とする。

下位通信ソフトウェアは、この要求を受け取ると、送信バッファ中の全ての電文を廃棄する。

4 . 2 . 1 3 LowReceiveData

(1) 名称

受信電文要求関数。

(2) 機能

下位通信ソフトウェアが保持している、受信 ECHONET 電文を要求する。

(3) 構文

```
short LowReceiveData (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    unsigned char *buf, /*[IN] 受信バッファへのポインタ */
    short buf_sz /*[IN] 受信バッファサイズ */
    short *rcv_sz /*[OUT] 受信データサイズ */
    unsigned char *sa /*[OUT] 送信元 MAC アドレス */
)
```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

*buf : 受信バッファへのポインタ(1バイト目は EDC)を指定する。

buf_sz : 受信バッファのサイズを指定する。

rcv_sz : 受信データのサイズが返される。

sa : 送信元 MAC アドレスが返される。

(5) 戻り値

LOW_NO_RECEIVE(0)	: 受信データ無し
LOW_NO_ERROR(1)	: 正常時(受信データ有り)
LOW_BUFFER_SIZE_ERROR(2)	: バッファサイズエラー
LOW_INTERNAL_ERROR(3)	: 下位通信ソフトウェア内部エラー

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、指定された下位通信ソフトウェアが通常動作状態にない場合は、下位通信ソフトウェア内部エラーを戻り値とする。

4.2.14 LowGetAddress

(1) 名称

アドレス情報取得要求関数

(2) 機能

下位通信ソフトウェアが保持しているアドレス情報を要求する。

(3) 構文

```

BOOL LowGetAddress (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    short mac_len, /*[OUT]MAC アドレス長*/
    unsigned char mac_ad[7], /*[OUT]MAC アドレス*/
    unsigned char mac_mask[7], /*[OUT]MAC アドレスマスク値*/
    short *housecode_len, /*[OUT]ハウスコード情報のサイズへのポインタ/
    unsigned char *housecode /*[OUT]ハウスコード情報へのポインタ*/
)

```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

mac_len : MAC アドレスのサイズが返る。

mac_ad : MAC アドレスが返る。

housecode_len : ハウスコード情報のサイズへのポインタが返る。0x00 は、ハウスコード情報を必要としないことを示す。

housecode : ハウスコード情報へのポインタが返る。

(5) 戻り値

0 : アドレス取得失敗
 1 : アドレス取得成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

特記無し。

4 . 2 . 1 5 LowSetAddress

(1) 名称

アドレス情報設定要求関数

(2) 機能

下位通信ソフトウェアにアドレス情報を設定する。

(3) 構文

```

short   LowSetAddress (
    unsigned char   device_id,           /*[IN]下位通信ソフトウェア ID */
    short mac_len,           /*[IN]MAC アドレス長*/
    unsigned char mac_ad[7], /*[IN]MAC アドレス*/
    unsigned char mac_mask[7], /*[IN]MAC アドレスマスク値*/
    short housecode_len, /*[IN]ハウスコード情報のサイズ*/
    unsigned char *housecode /*[IN]ハウスコード情報へのポインタ*/
)

```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

mac_len : MAC アドレスのサイズを指定する。
0x00 は、MAC アドレス設定要求無しを示す。

mac_ad : MAC アドレスを設定する。

housecode_len : ハウスコード情報のサイズを指定する。
0x00 は、ハウスコード設定要求無しを示す。

***housecode** : ハウスコード情報へのポインタを指定する。

(5) 戻り値

LOW_NO_CHEANGE(0) : ソフトウェアでは変更不可

LOW_NO_ERROR(1) : 正常時

LOW_INTERNAL_ERROR(3) : 下位通信ソフトウェア内部エラー

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

特記無し。

4 . 2 . 1 6 LowReqToMac

(1) 名称

物理アドレス変換要求関数

(2) 機能

下位通信ソフトウェアに対して、渡した NodeID に対応する MAC アドレスを要求する。

(3) 構文

```

BOOL LowReqToMac (
    unsigned char device_id, /* [IN] 下位通信ソフトウェア ID */
    unsigned char node_id, /* [IN] 変換対象の NodeID */
    unsigned char *mac, /* [OUT] 変換された MAC アドレスへのポインタ */
    short *mac_len /* [OUT] 変換された MAC アドレスサイズへのポインタ */
)

```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

node_id : 変換前の NodeID を設定する。
*mac : 変換後の MAC アドレスへのポインタが返る。
*mac_len : 変換された MAC アドレスのサイズへのポインタが返る。

(5) 戻り値

0 : 変換失敗
1 : 変換成功

(6) 使用する構造体

特に無し。

(7) 注意事項・制限事項

特記無し。

4 . 2 . 1 7 LowReqToID

(1) 名称

NodeID 変換要求関数

(2) 機能

下位通信ソフトウェアに対して、渡した MAC アドレスに対応する NodeID を要求する。

(3) 構文

```

BOOL LowReqToID (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    short mac_len /*[IN] 変換対象の MAC アドレス長 */
    unsigned char *mac, /*[IN] 変換対象の MAC アドレス */
    unsigned char *node_id, /*[OUT] 変換された NodeID */
)
    
```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

mac_len : 変換前の MAC アドレス長

mac : 変換前の MAC アドレスを指定する。

***node_id** : 変換後の NodeID へのポインタが返る。

(5) 戻り値

0 : 変換失敗
 1 : 変換成功

(6) 使用する構造体

特に無し。

(7) 注意事項・制限事項

特記無し。

4 . 2 . 1 8 LowReqBcastID

(1) 名称

同報先取得要求関数

(2) 機能

下位通信ソフトウェアに対して渡したドメイン内同報、あるいは自サブネット内同報指定時の DEA の同報対象指定コードから、対象となる NodeID を抽出する。

(3) 構文

```

BOOL LowReqBcastID (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    unsigned char bcast, /*[IN] 同報対象指定コード */
    short *map_len, /*[OUT] 送信先ノードのアドレス長*/
    unsigned char *map /*[OUT] 送信先ノードのアドレス情報 */
)
    
```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

bcast : 対象となる同報対象指定コード (ドメイン内同報、あるいは自サブネット内同報指定時の DEA の 2 バイト目の同報対象指定コード)。

map_len : 変換された NodeID を示すビットマップへのアドレス長

map : 変換された NodeID を示すビットマップへのアドレスを返す。同報先 NodeID とビットの対応を以下に示す。

map[0]-bit0	: NodeID 0 (0x00)
map[0]-bit1	: NodeID 1 (0x01)
.....	
map[1]-bit0	: NodeID 8 (0x08)
map[1]-bit1	: NodeID 9 (0x09)
.....	
map[31]-bit7	: NodeID 255 (0xFF)

(5) 戻り値

0 : 変換失敗
1 : 変換成功

(6) 使用する構造体

特に無し。

(7) 注意事項・制限事項

下位通信ソフトウェアに同報機能がある場合には本関数は必要ない。

4 . 2 . 1 9 LowInitAll

(1) 名称

完全初期化要求関数

(2) 機能

下位通信ソフトウェアに対し、ハウスコード情報、MAC アドレス再取得を伴う初期化(コールドスタート)を要求する。この要求を受けた下位通信ソフトウェアはコールドスタートし、通信停止状態に遷移した後、初期化パラメータを自身に設定する。

(3) 構文

```

BOOL LowInitAll (
    unsigned char    device_id,      /*[IN]  下位通信ソフトウェア ID */
    LOW_INIT_DATA *lowinit_data,    /*[IN]  初期化パラメータへのポインタ(1) */
    void            *low_init       /*[IN]  初期化パラメータへのポインタ(2) */
)

```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線 0x11 ~ 0x1F

特定小電力無線 0x31 ~ 0x3F

拡張 HBS 0x41 ~ 0x4F

IrDA_Control 0x51 ~ 0x5F

LonTalk[®] 0x61 ~ 0x6F

Bluetooth[™] 0x71 ~ 0x7F

Ethernet 0x81 ~ 0x8F

***lowinit_data** : 下位通信ソフトウェア共通規定項目の初期化パラメータへのポインタ。

***low_init** : 個々の下位通信ソフトウェア毎に異なる初期化パラメータへのポインタ。パラメータ内容は、個別下位通信ソフトウェア毎に規定する。

(5) 戻り値

0 : 初期化失敗

1 : 初期化成功

(6) 使用する構造体

```

typedef struct {
    short          sfholdtime, /* 下位通信ソフトウェア送信電文最長保持時間情報 */
    short          rfholdtime, /* 下位通信ソフトウェア受信電文最長保持時間情報 */
    unsigned char  low_mode,   /* 動作モード指定 */
    short          mac_len,    /* MAC アドレス長 */
    unsigned char  mac_ad[7]  /* MAC アドレス */
} LOW_INIT_DATA

```

mac_ad[7]以外は、特に初期化データが無い場合には、NULLを設定する。

mac_ad[7]は、mac_len が NULL 設定の場合には、意味が無いものとする。(mac_len が NULL の場合には、MAC アドレス設定無しとする。)

(7) 注意事項・制限事項

本関数は、下位通信ソフトウェアがコールドスタート状態、またはウォームスタート状態、通信停止状態にある場合は、初期化失敗を戻り値とする。

ハウスコード情報を使用しない下位通信ソフトウェアには対しては、初期化要求と同じ処理となる。

4 . 2 . 2 0 LowStop

(1) 名称

通信停止要求関数

(2) 機能

下位通信ソフトウェアに対し、通信停止を要求する。この要求を受けた下位通信ソフトウェアは通信停止状態に遷移する。

(3) 構文

```
BOOL LowStop (
    unsigned char    device_id    /*[IN]下位通信ソフトウェア ID */
)
```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

(5) 戻り値

0 : 通信停止失敗
1 : 通信停止成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、下位通信ソフトウェアが通常動作以外の状態にあるときは、通信停止失敗を戻り値とする。

下位通信ソフトウェアは、本要求を受けた時に電文送信途中であれば、一連の送信処理を終了し、通信停止状態に遷移する。また、電文の受信途中であれば、受信電文を廃棄し、処理を終了する。

一時停止中には以下の動作を行う。

- ・電文の受信
行わない。
- ・ECHONET 通信制御処理部からの電文送信要求
エラーリターンする。

4 . 2 . 2 1 LowHalt

(1) 名称

完全停止要求関数

(2) 機能

下位通信ソフトウェアに対し、完全停止を要求する。この要求を受けた下位通信ソフトウェアは停止状態に遷移する。

(3) 構文

```

BOOL LowHalt (
    unsigned char    device_id    /*[IN]下位通信ソフトウェア ID */
)

```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

(5) 戻り値

0 : 完全停止失敗

1 : 完全停止成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、下位通信ソフトウェアがコールドスタート状態、ウォームスタート状態にあるときは、停止失敗を戻り値とする。

下位通信ソフトウェアは、本要求を受けた時に電文送信途中であれば、一連の送信処理を終了し、停止状態に遷移する。また、電文の受信途中であれば、受信電文を廃棄し、処理を終了する。

停止中には以下の動作を行う。

- ・電文の受信
行わない。
- ・ECHONET 通信制御処理部からの電文送信要求
エラーリターンする。

4 . 2 . 2 2 LowReceiveStop

(1) 名称

停止通知要求関数。

(2) 機能

下位通信ソフトウェアが停止通知を送信した場合に、受信する。

(3) 構文

```
short LowReceiveStop (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
)
```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk [®]	0x61 ~ 0x6F
Bluetooth [™]	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

(5) 戻り値

LOW_NO_RECEIVE(0) : 停止通知無し
 LOW_NO_ERROR(1) : 停止通知あり
 LOW_INTERNAL_ERROR(3) : 下位通信ソフトウェア内部エラー

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

本関数は、指定された下位通信ソフトウェアから停止通知を受信した時に、LOW_NO_ERROR(1)を返す。

4 . 2 . 2 3 LowGetAddressTableDataSize

(1) 名称

下位通信ソフトウェアアドレステーブルデータサイズ取得関数

(2) 機能

下位通信ソフトウェアで保持している下位アドレステーブルデータの組数の取得を行う。

(3) 構文

```

BOOL LowGetAddressTableDataSize (
    unsigned char device_id, /*[IN]下位通信ソフトウェア ID*/
    unsigned char *data_number /*[OUT]データ組数*/
)

```

(4) 説明

LowGetAddressTableData()とセットで使用することを想定した関数である。出力データは下位通信ソフトウェアで保持しているアドレステーブルのデータ組数で、data_numberの示す領域に格納される。

device_id : 下位通信ソフトウェア識別情報。

電灯線 0x11 ~ 0x1F

特定小電力無線 0x31 ~ 0x3F

拡張 HBS 0x41 ~ 0x4F

IrDA_Control 0x51 ~ 0x5F

LonTalk^R 0x61 ~ 0x6F

BluetoothTM 0x71 ~ 0x7F

Ethernet 0x81 ~ 0x8F

data_number : 下位アドレステーブルデータで保持しているアドレステーブルの組数のポインタ。

(5) 戻り値

0 : 完全停止失敗

1 : 完全停止成功

(6) 注意事項

なし

4 . 2 . 2 4 LowGetAddressTableData

(1) 名称

下位通信ソフトウェアアドレステーブルデータ取得関数

(2) 機能

下位通信ソフトウェアで保持している下位アドレステーブルデータの取得を行う。

(3) 構文

```

BOOL LowGetAddressTableData (
    unsigned char    device_id,          /*[IN]  下位通信ソフトウェア ID  */
    unsigned char*   data_number,       /*[IN/OUT] データ組数*/
    ADDRESSTABLE*   adresstable        /*[OUT] アドレステーブル構造体 */
)

```

(4) 説明

入力データ(data_number)は、MidGetAddressTableDataSize で取得したアドレステーブルの組数のポインタ。

出力データは、実際に格納したアドレステーブルの組数と、各データ組のハードウェアアドレス、MAC アドレス、及びマスターータであることを示すフラグからなる構造体の配列データからなる。

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk ^R	0x61 ~ 0x6F
Bluetooth TM	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

data_number : 下位アドレステーブルで保持しているアドレステーブルのデータ組数へのポインタを示す。

adresstable : 下位アドレステーブルデータで保持しているハードウェアアドレス、MAC アドレス、及びマスターータであることを示すフラグを収容するアドレステーブル構造体の配列の先頭ポインタ。

(5) 戻り値

0 : 失敗

1 : 成功

(6) 使用する構造体

```

typedef struct{
    unsigned char hardwareaddress_size; /*[OUT] ハードウェアアドレスのデータサイズ*/
    unsigned char hardwareaddress[8]; /*[OUT] ハードウェアアドレス。下位バイト詰めで格納。*/

    unsigned char node_id; /*[OUT] MAC アドレス */
    unsigned char masterrouter_flag; /*[OUT] 対応するノードがマスターータであるか否かを示す識別子。マスターータであれば1、そうでなければ0。*/
}ADDRESSTABLE

```

(7) 注意事項・制限事項

第 2 引数 data_number は、入出力兼用のためデータ内容が上書きされるので注意が必要。

4 . 2 . 2 5 LowSetMasterRouterFlag

(1) 名称

マスタールータ通知関数

(2) 機能

自ノードがマスタールータであるか否かを下位通信ソフトウェアに通知する。

(3) 構文

```

BOOL LowSetMasterRouterFlag (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    unsigned char masterRouter_Flag /*[IN] マスタールータ識別フラグ */
)

```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk ^R	0x61 ~ 0x6F
Bluetooth TM	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

masterRouter_Flag : マスタールータであれば 1、そうでなければ 0 を指定する。また未確定時は 0 を指定する。

(5) 戻り値

0 : 失敗
1 : 成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

無し。

4 . 2 . 2 6 LowGetHardwareAddress

(1) 名称

ハードウェアアドレスデータ取得関数

(2) 機能

下位通信ソフトウェアに対して保持しているハードウェアアドレスデータの取得を行う。

(3) 構文

```

BOOL LowGetHardwareAddress (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    unsigned char* hardwareaddresssize, /*[OUT]ハードウェアアドレスのサイズ*/
    unsigned char* hardwareaddress, /*[OUT] ハードウェアアドレス */
)

```

(4) 説明

出力データはハードウェアタイプとハードウェアアドレス、MAC アドレスである。

device_id : 下位通信ソフトウェア識別情報。

電灯線 0x11 ~ 0x1F

特定小電力無線 0x31 ~ 0x3F

拡張 HBS 0x41 ~ 0x4F

IrDA_Control 0x51 ~ 0x5F

LonTalk^R 0x61 ~ 0x6FBluetoothTM 0x71 ~ 0x7F

Ethernet 0x81 ~ 0x8F

Hardwareaddresssize : 下位通信ソフトウェアで保持しているハードウェアアドレスのサイズのポインタを示す。

hardwareaddress : 下位アドレステーブルデータで保持しているアドレスに対応するハードウェアハードウェアアドレスのポインタを示す。

(5) 戻り値

0 : 失敗

1 : 成功

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

無し。

4 . 2 . 2 7 LowGetEchonetMACList

(1) 名称

ECHONET MAC アドレスリスト取得関数

(2) 機能

下位通信ソフトウェアが保持する ECHONET MAC アドレスリストを取得する。

(3) 構文

```

BOOL LowGetEchonetMACList (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    unsigned char *mac_addr_list /*[OUT] ECHONET MAC アドレスリスト */
)

```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線 0x11 ~ 0x1F

特定小電力無線 0x31 ~ 0x3F

拡張 HBS 0x41 ~ 0x4F

IrDA_Control 0x51 ~ 0x5F

LonTalk^R 0x61 ~ 0x6FBluetoothTM 0x71 ~ 0x7F

Ethernet 0x81 ~ 0x8F

*mac_addr_list : 下位通信ソフトウェアが保持している ECHONET MAC アドレスリストの配列(要素数 32)へのポインタである。以下のようなデータとする。

ECHONET MAC アドレスリストは、下図の 3 2 バイトのテーブルにおいて、存在する ECHONET MAC アドレス (1 6 進表記) を示すビット位置に 1 をセットして 1 バイト目から順に列挙する。

	ビット0	ビット1	ビット2	ビット3	ビット4	ビット5	ビット6	ビット7
1 バイト目	F 8	F 9	F A	F B	F C	F D	F E	F F
2 バイト目	F 0	F 1	F 2	F 3	F 4	F 5	F 6	F 7
3 バイト目	E 8	E 9	E A	E B	E C	E D	E E	E F
4 バイト目	E 0	E 1	E 2	E 3	E 4	E 5	E 6	E 7
5 バイト目	D 8	D 9	D A	D B	D C	D D	D E	D F
6 バイト目	D 0	D 1	D 2	D 3	D 4	D 5	D 6	D 7
7 バイト目	C 8	C 9	C A	C B	C C	C D	C E	C F
8 バイト目	C 0	C 1	C 2	C 3	C 4	C 5	C 6	C 7
9 バイト目	B 8	B 9	B A	B B	B C	B D	B E	B F
1 0 バイト目	B 0	B 1	B 2	B 3	B 4	B 5	B 6	B 7
1 1 バイト目	A 8	A 9	A A	A B	A C	A D	A E	A F
1 2 バイト目	A 0	A 1	A 2	A 3	A 4	A 5	A 6	A 7
1 3 バイト目	9 8	9 9	9 A	9 B	9 C	9 D	9 E	9 F
1 4 バイト目	9 0	9 1	9 2	9 3	9 4	9 5	9 6	9 7
1 5 バイト目	8 8	8 9	8 A	8 B	8 C	8 D	8 E	8 F
1 6 バイト目	8 0	8 1	8 2	8 3	8 4	8 5	8 6	8 7
1 7 バイト目	7 8	7 9	7 A	7 B	7 C	7 D	7 E	7 F
1 8 バイト目	7 0	7 1	7 2	7 3	7 4	7 5	7 6	7 7
1 9 バイト目	6 8	6 9	6 A	6 B	6 C	6 D	6 E	6 F
2 0 バイト目	6 0	6 1	6 2	6 3	6 4	6 5	6 6	6 7
2 1 バイト目	5 8	5 9	5 A	5 B	5 C	5 D	5 E	5 F
2 2 バイト目	5 0	5 1	5 2	5 3	5 4	5 5	5 6	5 7
2 3 バイト目	4 8	4 9	4 A	4 B	4 C	4 D	4 E	4 F
2 4 バイト目	4 0	4 1	4 2	4 3	4 4	4 5	4 6	4 7
2 5 バイト目	3 8	3 9	3 A	3 B	3 C	3 D	3 E	3 F
2 6 バイト目	3 0	3 1	3 2	3 3	3 4	3 5	3 6	3 7
2 7 バイト目	2 8	2 9	2 A	2 B	2 C	2 D	2 E	2 F
2 8 バイト目	2 0	2 1	2 2	2 3	2 4	2 5	2 6	2 7
2 9 バイト目	1 8	1 9	1 A	1 B	1 C	1 D	1 E	1 F
3 0 バイト目	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7
3 1 バイト目	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F
3 2 バイト目	0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7

(5) 戻り値

- 0 : 異常時
- 1 : 正常時

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

ECHONET MAC アドレスが 2 バイトとなる電灯線が伝送メディアの場合は、下位 1 バイトをリストアップする。

4 . 2 . 2 8 LowGetMasterRouterInfo

(1) 名称

マスターータ情報取得関数

(2) 機能

下位通信ソフトウェアが保持するマスターータ情報を取得する。

(3) 構文

```

BOOL LowGetMasterRouterInfo (
    unsigned char device_id, /*[IN] 下位通信ソフトウェア ID */
    unsigned char *result, /*[OUT] マスターータの存在の有無*/
    unsigned char *master_mac_addr /*[OUT] マスターータの ECHONET MAC アドレス */
)

```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線	0x11 ~ 0x1F
特定小電力無線	0x31 ~ 0x3F
拡張 HBS	0x41 ~ 0x4F
IrDA_Control	0x51 ~ 0x5F
LonTalk ^R	0x61 ~ 0x6F
Bluetooth TM	0x71 ~ 0x7F
Ethernet	0x81 ~ 0x8F

*result : マスターータの存在の有無へのポインタを示す。

0x00 : 無し
0x01 : 有り

*master_mac_addr : マスターータの ECHONET MAC アドレスの値へのポインタを示す。

(5) 戻り値

0 : 異常時
1 : 正常時

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

ECHONET MAC アドレスが 2 バイトとなる電灯線が伝送メディアの場合は、下位 1 バイトをリストアップする。

4 . 2 . 2 9 LowReqToHardwareAddress

(1) 名称

ハードウェアアドレス変換要求関数

(2) 機能

下位通信ソフトウェアに対して渡した ECHONET MAC アドレス情報に対応するハードウェアアドレスを要求する。

(3) 構文

```

BOOL LowGetHardwareAddress (
  unsigned char    device_id,          /*[IN]  下位通信ソフトウェア ID */
  short            mac_len,           /*[IN]  変換対象の MAC アドレス長*/
  unsigned char    *mac,              /*[IN]  変換対象の MAC アドレスのポインタ */
  unsigned char*   hardwareaddress,  /*[OUT] ハードウェアアドレスへのポインタ*/
  unsigned char*   hardwareaddress_len /*[OUT] ハードウェアアドレスのサイズへのポインタ*/
)

```

(4) 説明

device_id : 下位通信ソフトウェア識別情報。

電灯線 0x11 ~ 0x1F

特定小電力無線 0x31 ~ 0x3F

拡張 HBS 0x41 ~ 0x4F

IrDA_Control 0x51 ~ 0x5F

LonTalk^R 0x61 ~ 0x6FBluetoothTM 0x71 ~ 0x7F

Ethernet 0x81 ~ 0x8F

mac_len : 変換前の MAC アドレス長

*mac : 変換前の MAC アドレスのポインタを指定する。

hardwareaddress : 変換後のハードウェアアドレスへのポインタが返る。

hardwareaddress_len : 変換後のハードウェアアドレスのサイズのポインタが返る。

(5) 戻り値

0 : 異常時

1 : 正常時

(6) 使用する構造体

無し。

(7) 注意事項・制限事項

無し。

4.3 イニシャル時設定情報仕様

本節では、下記 6 つの下位通信ソフトウェア毎に、「初期化要求：LowInit」関数（下記備考参照）の引数の初期化パラメータポインタ「*low_init」が示す領域にて規定する初期化パラメータ仕様を示す。

- (1) 電灯線下位通信ソフトウェア
- (2) 特定小電力無線下位通信ソフトウェア
- (3) 拡張 HBS 下位通信ソフトウェア
- (4) IrDA 依存下位通信ソフトウェア
- (5) LonTalk[®]依存下位通信ソフトウェア
- (6) Bluetooth[™]下位通信ソフトウェア
- (7) Ethernet 下位通信ソフトウェア

備考：LowInit 関数の構文。

```

BOOL LowInit (
    short    device_id,           /*[IN] 下位通信ソフトウェア種別 ID */
    LOW_INIT_DATA *init_data,    /*[IN] 初期化パラメータへのポインタ(1) */
    void     *low_init           /*[IN] 初期化パラメータへのポインタ(2) */
)

```

4.3.1 電灯線下位通信ソフトウェア用初期化パラメータ仕様

```

typedef struct {
    short          sbuf_len;      /* 送信バッファサイズ */
    short          *sbuf;        /* 送信バッファへのポインタ */
    short          rbuf_len;      /* 受信バッファサイズ */
    short          *rbuf;        /* 受信バッファへのポインタ */
} PLCA_INIT_DATA

```

4.3.2 特定小電力無線下位通信ソフトウェア用初期化パラメータ仕様

```

typedef struct {
} RF_INIT_DATA

```

4.3.3 拡張 HBS 下位通信ソフトウェア用初期化パラメータ仕様

```

typedef struct {
    short          sbuf_len;      /* 送信バッファサイズ */
    short          *sbuf;        /* 送信バッファへのポインタ */
    short          rbuf_len;      /* 受信バッファサイズ */
    short          *rbuf;        /* 受信バッファへのポインタ */
} HBS_INIT_DATA

```

4.3.4 IrDA 依存下位通信ソフトウェア用初期化パラメータ仕様

```

typedef struct {
    short          sbuf_len;      /* 送信バッファサイズ */
    short          *sbuf;        /* 送信バッファへのポインタ */
    short          rbuf_len;      /* 受信バッファサイズ */
    short          *rbuf;        /* 受信バッファへのポインタ */
    short          mac_table_len; /* MAC アドレス変換テーブルサイズ */
    short          *mac_table;   /* MAC アドレス変換テーブルへのポインタ */
} IRDA_INIT_DATA

```

4.3.5 LonTalk^R依存下位通信ソフトウェア用初期化パラメータ仕様

```

typedef struct {
    short          sbuf_len;      /* 送信バッファサイズ */
    short          *sbuf;        /* 送信バッファへのポインタ */
    short          rbuf_len;      /* 受信バッファサイズ */
    short          *rbuf;        /* 受信バッファへのポインタ */
} LON_INIT_DATA

```

4 . 3 . 6 Bluetooth™ 下位通信ソフトウェア用初期化パラメータ仕様

```
typedef struct {
    short      sbuf_len;      /* 送信バッファサイズ */
    short      *sbuf;        /* 送信バッファへのポインタ */
    short      rbuf_len;     /* 受信バッファサイズ */
    short      *rbuf         /* 受信バッファへのポインタ */
} BT_INIT_DATA
```

4 . 3 . 7 Ethernet 下位通信ソフトウェア用初期化パラメータ仕様

```
typedef struct {
    short      sbuf_len;     /* 送信バッファサイズ */
    short      *sbuf;       /* 送信バッファへのポインタ */
    short      rbuf_len;    /* 受信バッファサイズ */
    short      *rbuf        /* 受信バッファへのポインタ */
} ETHERNET_INIT_DATA
```