

既存家電機器をスマート家電化する家電認識スマートタッグの提案

松石 浩輔[†] 沼尾 雅之[†]

[†] 電気通信大学大学院情報理工学研究科 〒182-8585 東京都調布市調布ヶ丘 1-5-1

E-mail: [†] {kosuke.matsuishi@uec.ac.jp, numao@cs.uec.ac.jp}

あらまし 近年、節電要請の高まりとともに HEMS (Home Energy Management System) が普及しつつある。ところが、既存家電機器の多くは HEMS に対応していない。

そこで我々は、家電機器の電源波形に基づく家電認識技術を応用し、既存家電機器を HEMS に対応させる家電認識スマートタッグのアーキテクチャを提案する。提案アーキテクチャは、HEMS 向けの標準規格である ECHONET Lite の拡張であり、既存の ECHONET Lite 機器との後方互換性を有する。

本稿では、既存手法と異なり、機器メーカーが家電機器のメタデータあるいはスペック（詳細規定）を公開し、これをスマートタッグが利用して機器の動作モード推定や赤外線リモコンを介した制御を行うことで、set 系操作を含む既存家電機器のスマート家電化および HEMS 対応化を実現する手法を提案する。

キーワード HEMS, スマートハウス, 家電認識, ECHONET Lite, メタデータ

1. はじめに

2015 年 12 月、国連気候変動枠組条約第 21 回締約国会議（COP21）において、温室効果ガスの排出削減を目指す国際的な枠組みであるパリ協定が採択された。ここで日本は「2013 年度比で 2030 年度までに温室効果ガス排出量を 26.0%削減する」との目標を掲げている。その実現のため家庭部門では“HEMS の普及”などにより CO₂ 排出量を約 40%削減するとしている[1]。

この HEMS (Home Energy Management System) とは、様々な家電機器や太陽光・風力・ガスなどによる発電機器、蓄電池、電気自動車などをネットワークで接続して一元的に管理し、省エネルギーな家庭を実現するシステムである。前述の COP21 の例にもあるように、HEMS の普及はいまや国家レベル、あるいは世界レベルで重要な意味をもつ。

近年、HEMS を実現するための通信プロトコルが複数提案されており、とくに日本国内ではエコーネットコンソーシアムによる ECHONET Lite[2]が普及しつつある。ところがこの ECHONET Lite には、(1)規格[3]のカバーする機器の種類が不十分であり、それにも関わらず(2)規格にない新たな機器や機能を開発者が追加することが困難である、という問題点が存在する。

そこで我々[4]は、ECHONET Lite における上記ふたつの問題点を解決する ECHONET Lite 拡張を提案している。この拡張は、従来は ECHONET 規格書 Appendix[3]に人間向けの形式で書かれていた機器のメタデータあるいはスペック（詳細規定）を機械可読な XML 形式で記述および配布することで、開発者が新たな ECHONET Lite 機器を自由に開発し、マルチベンダ環境における動作を実現するものである。この提案拡張は、既存の ECHONET Lite 規格に対し後方互換性を有する。

また我々は同時に、上記の ECHONET Lite 提案拡張および家電認識技術を用いた家電認識スマートタッグ[4]を提案している。これは家電認識技術を用い、HEMS 非対応の既存家電機器を ECHONET Lite ネットワークに参加させるものである。

本稿では、我々の ECHONET Lite に対する提案拡張[4]および家電認識スマートタッグ[4]をさらに拡張し、赤外線リモコンおよび機器の消費電力パターンによる動作モード推定を用いて、これまで不可能であった set 系操作を含む、既存家電機器のスマート家電化および HEMS 対応化を実現する手法について述べる。

2. 関連技術と既存研究

本章では、本研究の関連技術である ECHONET Lite と、家電認識手法に関する既存研究を紹介し、これらの問題点について述べる。

2.1. ECHONET Lite

ECHONET Lite[2]は、スマートハウスおよび HEMS のための通信プロトコルであり、エコーネットコンソーシアムによって策定されている。

ECHONET Lite の技術的特徴としては以下の点が挙げられる。

- (1) オブジェクト指向による機器のモデル化。対応機器を機器クラスとして定義し、それぞれの機器クラスに対し固有のプロパティが規定されている。たとえば家庭用エアコンクラスは、運転モード設定、温度設定値、室内温度設定値などのプロパティをもつ。また各クラスはクラスグループコードおよびクラスコードと呼ばれる各 1 バイトの識別子で、また各プロパティはプロパティコードまたは EPC (Echonet Property Code) と呼ばれる 1 バイトの識別子で区別される。

(2) セッション層以上のレイヤのみを規定すること。

OSI 参照モデルにおけるトランスポート層以下の各レイヤを規定せず、TCP/IP や IEEE802.11 などの既存規格と組み合わせて利用する。

(3) 単純な通信シーケンスと軽量な通信フォーマット。

ECHONET Lite は、バイナリベースの軽量なフレームフォーマット（電文構成）と、前述のオブジェクト指向に基づく単純な通信シーケンスを規定しており、容易に実装できるとされる。

ECHONET Lite は、ホームネットワーク向けの“公知で標準的なインタフェース”であると経済産業省スマートハウス標準化検討会により評価[5]され、また対応機器も市場に出回るなど、日本国内では今後の普及が見込まれている。

2.1.1. ECHONET Lite に関する既存研究

大和田ら[6]は、ECHONET Lite の Java ライブラリである OpenECHO[7]を開発し、これを利用した Android 向け ECHONET Lite サーバアプリである Kadecot[6][8]を開発している。

郷司ら[9]は、ECHONET Lite の通信パケットを可視化するパケットアナライザを開発するため、人間向けの PDF 形式で提供されている ECHONET Lite 仕様書を、自動的に XML ファイルへと変換する試みを行っている。これによると、仕様書フォーマットの問題のため自動的な変換には限界があり、変換には人間による手直しが必要であったとされる。

我々[4]は、ECHONET Lite に存在するふたつの問題、すなわち(1)規格[3]のカバーする機器の種類が不十分であり、それにも関わらず(2)規格にない新たな機器や機能を開発者が追加することが困難である、という点に着目し、これらを解決する ECHONET Lite 拡張を提案している。これは、従来は ECHONET 規格書 Appendix[3]に人間向けの形式で書かれていた機器のスペック（詳細規定）を機械可読な XML 形式（詳細規定 XML）で記述および配布することで、開発者が新たなユーザ定義機器を自由に開発し、マルチベンダ環境における動作を実現するものである（図 1）。詳細規定 XML の記述には、既存の ECHONET Lite 規格を XML タグセットとして再定義した標準タグセットと、ユーザ定義機器を実現する拡張タグセットとを用いる。また提案拡張は、既存の ECHONET Lite 規格に対し後方互換性を有する。

2.2. 家電認識手法

家電認識手法とは、家電機器の電源電圧および消費電流のパターンからその機器の種類や個体を認識する技術であり、Appliance Load Monitoring (ALM または LM)、機器モニタリングとも言われ、主に電力使用量の理解と節電に役立てられる。

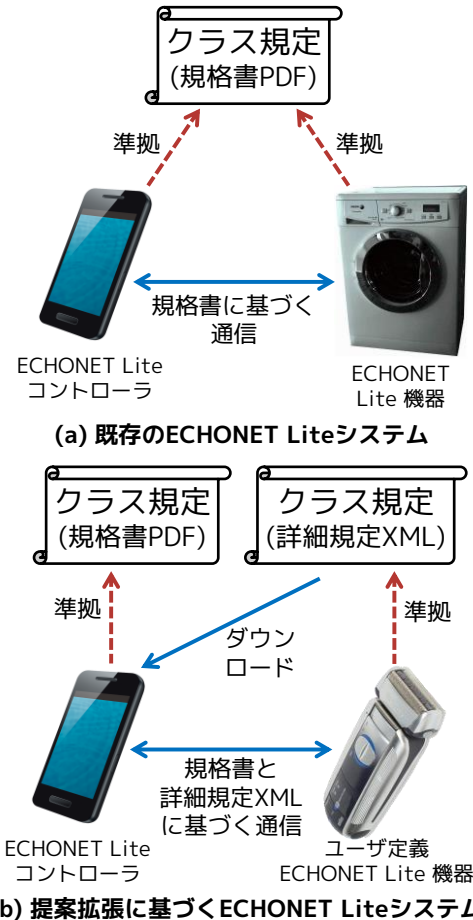


図 1 (a)既存の ECHONET Lite システムと、(b)我々の提案する拡張[4]による ECHONET Lite システムの比較

マルチベンダ環境での利用を念頭におくと、既存の ECHONET Lite システム (a) では、規格書に記載されている機器のみが実現可能であった。一方、我々の提案拡張[4]によるシステム (b) では、開発者が自ら機器のスペック（詳細規定）を XML 形式で記述し配布する。これにより開発者は、規格書にない新たな機器を自由に開発し、マルチベンダ環境での動作を実現できる。

家電認識手法には、複数機器の合成電流波形を観測する手法と、個別機器の波形をそれぞれ観測する手法が存在する。とくに後者の手法としては、スマートタップにおいて電流の平均値、ピーク値、実効値を収集して機器の種類を認識する手法[10]、電流の平均値や電圧と電流の位相差に基づき機器の種類、稼働状況、稼働位置を認識する手法[11]、電流波形を主成分分析し機器の特徴量とする手法[12]のほか、多数の先行研究が存在する[13]。また我々[11]は、機器の特徴量として周波数成分を利用し、電源電圧および電源周波数の変動に頑健な手法を提案している。

2.2.1. 家電認識手法と HEMS に関する既存研究

Saitoh ら[10]および伊藤ら[11]は、それぞれ個別の研究として、家電認識手法を用いて既存家電機器をホームネットワークに接続し、ホームサーバにおいて機器の消費電力可視化および電源のオン・オフ制御を行うシステムを提案している。

加藤ら[12]は、家電認識結果をホームサーバに蓄積し、機器の監視および制御にとどまらず、電力可視化や高消費電力機器の買い替え提案などといったサービスを実現するためのプラットフォームを提案している。

また我々[4]は、家電認識手法[11]および我々の ECHONET Lite 拡張[4]を用いて、HEMS 非対応の既存家電機器を ECHONET Lite に対応させる家電認識スマートタップのアーキテクチャを提案している。

2.3. 既存研究の問題点

家電認識手法と HEMS に関する既存研究[10][11][12]では、家電機器を認識しホームサーバによって制御することはできるが、その通信プロトコルがクローズドなものであり、他のシステムと組み合わせることが考慮されていないという問題がある。

また我々の手法を含む既存研究[4][10][11][12]では、家電機器の制御方法として、電源回路の接続・切断による強制的な電源制御のみが可能であり、その他のソフトウェア的な制御ができないという問題がある。このため、たとえばエアコンの温度制御やテレビのスリープ（待機モードへの移行）といった、HEMS における基礎的な機能が実現できない。

そこで本稿では、我々の ECHONET Lite に対する提案拡張および家電認識スマートタップ[4]を改良し、赤外線リモコンおよび機器の消費電力パターンによる動作モード推定を用いて、これまで不可能であった既存家電機器のほぼ完全なスマート家電化および HEMS 対応化を実現するアーキテクチャについて述べる。

3. 提案アーキテクチャ

本稿で提案する家電認識スマートタップの概略を図 2 に示す。提案アーキテクチャは、主にスマートタップ、家電認識 Web サーバ、詳細規定サーバの 3 要素から構成される。以下ではこれら提案アーキテクチャの構成について述べたのち、機器のスペックを記述する詳細規定 XML について述べる。

3.1. 提案アーキテクチャの構成要素

3.1.1. スマートタップ

スマートタップは、家庭内に複数設置され、コンセントによって HEMS 非対応の既存家電機器に接続される。タップはこの家電機器の電源波形を取得し、その特徴量を抽出して家電認識 Web サーバへと問い合わせる。その応答として、タップに接続された家電機器の種類、機種、メーカーおよび詳細規定 URI を取得する。これらの応答に基づき、接続された家電機器があたかも ECHONET Lite 対応機器であるかのようにふるまう。すなわち本スマートタップは、**ECHONET Lite 家電プロキシ**であると言える。

またスマートタップは、家電認識 Web サーバから得

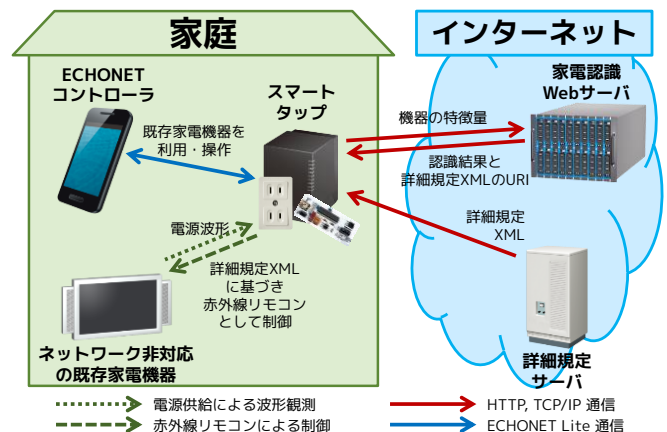


図 2 本稿で提案する家電認識スマートタップのアーキテクチャ概略

た詳細規定 URI に基づき、詳細規定サーバにアクセスして、接続された家電機器のスペック（詳細規定 XML）を取得する。このスペックに基づき、家電機器の動作モードを推定し、またタップが有する赤外線リモコンを介して家電機器の制御を行うことで、既存家電機器のほぼ完全なスマート家電化を実現する。すなわち本稿の家電プロキシは、我々の既存研究[4]を含む既存研究と比べて機能面で大幅な拡張がなされている。

3.1.2. 家電認識 Web サーバ

家電認識 Web サーバは、インターネット上にただひとつ存在し、公開されている。このサーバには、さまざまな家電機器における家電認識のための特徴量と、その家電機器のメーカー・機種・詳細規定 URI が登録されており、スマートタップなどのクライアントからの問い合わせに応じて家電認識処理を行う。また家電メーカーは、その製造および販売する家電機器の情報を本サーバに登録する。

この家電認識 Web サーバは、一般的な REST スタイルの Web API を有し、そのリソースとして家電機器 (/appliance)、家電機器メーカー (/manufacturer)、家電機器のクラス (/applianceClass) および家電認識器 (/recognizer) をもつ。本サーバを利用して家電認識を行うには、家電認識器リソースの HTTP GET メソッドを呼び、クエリ文字列に家電機器の特徴量（有効電力 w 、力率 pf など）を与えると、その認識結果が JSON 形式で得られる。たとえば、本サーバに対して図 3(a) のようなリクエストを送信すると、その応答として図 3(b) のような JSON 形式の認識結果が得られる。

家電認識 Web サーバのもっとも重要な機能は、家電認識手法に基づき、HEMS 非対応の既存家電機器とそのスペック（詳細規定 XML）とを紐付けるという点である。通常のスマート家電機器であれば、機器自身が自らのスペックを知っており、自身のもつ機能をネッ

```
1 GET /recognizer/?v=1&w=26.5&pf=0.893&pa=0.550&aa=0.0110&rmsa=0.286&hz=50.0&hn=4.462,176.236,2.907
```

図 3(a) 家電認識 Web サーバへのリクエスト例

```
1 [
2   {
3     "rank": 1,
4     "score": 0.95,
5     "appliance": {
6       "applianceId": "1",
7       "applianceName": "リビング扇風機",
8       "applianceModelNumber": "F-LN5",
9       "applianceClasses": [
10        {
11          "applianceClassId": 1,
12          "applianceClassName": "扇風機"
13        }
14      ],
15      "manufacturerId": "toshiba",
16      "manufacturerName": "東芝",
17      "echonetSpec": "http://.../f-ln5.xml"
18    }
19  ]
20 ]
```

図 3(b) 図 3(a)に対する家電認識 Web サーバの応答例

トワーク上や通信相手にアナウンスする機能があらかじめ備わっている．ところが既存家電機器は，自分自身の機種や機能すなわちスペックを機器外部から取り出すことができない．そのため，既存家電機器をスマート家電化するためには，機器とそのスペックとを外部で紐付ける必要があり，本稿における提案ではこの機能を家電認識 Web サーバが担う．

3.1.3. 詳細規定サーバ

詳細規定サーバは，各家電メーカーがそれぞれ保有および管理し，インターネット上に公開されている．詳細規定サーバには，各家電メーカーが販売する家電機器のスペックが詳細規定 XML の形で公開されており，このスペックをスマートタップとその (ECHONET Lite ネットワークにおける) 通信相手機器がダウンロードし利用する．

3.2. 詳細規定 XML とその拡張

詳細規定 XML は，2.1.1 節で述べたように我々の既存研究[4]における提案であり，既存の ECHONET 規格では人間向け PDF として提供されている機器のスペック (詳細規定) を，機械可読な XML 形式で表現するものである．詳細規定 XML の記述には，既存の ECHONET Lite 規格を XML 名前空間として再定義した **標準名前空間 be** (Base Echonet) と，ユーザ定義機器を実現する **拡張名前空間 ee** (Extended Echonet) を用いる．

詳細規定 XML の例として，扇風機プロキシクラスの詳細規定 XML を図 4 に示す．このプロキシクラスは，提案アーキテクチャにおけるスマートタップが自

```
1 <!-- 扇風機プロキシクラスの定義 -->
2 <be:classDefinition
3   be:classGroupCode="0F" be:classCode="10">
4   <be:name>扇風機プロキシクラス</be:name>
5   <ee:iconUri>http://.../fan.png</ee:iconUri>
6
7   <be:properties> <!-- プロパティ一覧 -->
8     <!-- 動作状態 (電源On/Off) プロパティ -->
9     <be:switchProperty be:epc="80" be:length="1"
10      be:get="true" be:set="true">
11       <be:name>動作状態 (電源On/Off)</be:name>
12       <be:option be:edt="30">
13         <be:name>オン</be:name>
14         <ep:irCommand>8240BF12ED0000</ep:irCommand>
15       </be:option>
16       <be:option be:edt="31">
17         <be:name>オフ</be:name>
18         <ep:irCommand>8240BF12ED0000</ep:irCommand>
19       </be:option>
20     </be:switchProperty>
21
22     <!-- 風量プロパティ -->
23     <be:levelProperty be:epc="A0" be:length="1"
24      be:get="false" be:set="true">
25       <be:name>風量</be:name>
26       <!-- 弱のときプロパティ値0x31 -->
27       <be:option be:edt="31">
28         <be:name>弱</be:name>
29         <ep:irCommand>8240BF01FE0000</ep:irCommand>
30         <ep:powerConsumptionPattern>
31           <ep:powerState
32             ep:watt="28" ep:durationSec="1" />
33         </ep:powerConsumptionPattern>
34       </be:option>
35       <!-- 強のときプロパティ値0x32 -->
36       <be:option be:edt="32">
37         <be:name>強</be:name>
38         <ep:irCommand>8240BF02FD0000</ep:irCommand>
39         <ep:powerConsumptionPattern>
40           <ep:powerState
41             ep:watt="40" ep:durationSec="5" />
42         </ep:powerConsumptionPattern>
43       </be:option>
44     </be:levelProperty>
45   </be:properties>
46 </be:classDefinition>
```

図 4 扇風機プロキシクラスを規定する
詳細規定 XML の例

動的に生成する，タップのコンセントに接続された家電機器 (この場合は扇風機) を表す ECHONET 機器クラスのひとつである．図 4 では，この扇風機プロキシクラスのクラスコード (2 行目)，クラスの名称 (4 行目)，On/Off プロパティ (7～20 行目)，風量プロパティ (22～40 行目) などを規定している．

本稿における提案では，この詳細規定 XML[4]に新たな名前空間である **ECHONET 家電プロキシ名前空間 ep** (Echonet Proxy) を追加する．この名前空間は，機器の赤外線リモコンコマンドを示す **ep:irCommand** タグと，機器の動作モードによる電力消費パターンを示す **ep:powerConsumptionPattern** タグを有する．以下ではこの家電プロキシ名前空間について述べる．

```

1 namespace xsd = "http://www.w3.org/2001/
  XMLSchema-datatypes" # データ型
2 namespace be = "..." # 標準名前空間
3 namespace ee = "..." # 拡張名前空間
4 namespace ep = "..." # ECHONET家電プロキシ名前空間
5
6 start = # XML のルートノードは classDefinitions 要素
7 element be:classDefinitions {
8   element be:classDefinition {
9     classIdentifier, # クラス識別子
10    element be:name { text }, # 機器の名前
11    element ee:iconUri { text }, # 機器アイコンURI
12    element be:properties { # プロパティ一覧
13      ( element be:numericalProperty { pc } # 数値型
14      | element be:dateProperty { pc } # 日付型
15      | element be:timeProperty { pc } # 時刻型
16      | element be:levelProperty { pc } # レベル型
17      | element be:characterProperty { pc } # 文字列型
18      | element be:resetProperty { pc } # リセット型
19      | element be:switchProperty { pc } # 切り替え型
20      | element be:selectionProperty { pc } # 選択型
21      | element be:compositeProperty { pc } # 複合型
22      | element be:unclassifiedProperty { pc } # その他
23    )*
24  },
25  element ee:relationships { # 継承・集約
26    element ee:subClassOf { classIdentifier }*,
27    element ee:aggregates { classIdentifier }*
28  }
29 }+
30 }
31
32 classIdentifier = # クラスの識別子
33 attribute be:classGroupCode { xsd:hexBinary },
34 attribute be:classCode { xsd:hexBinary },
35
36 pc = # 各プロパティを表す *Property 要素の中身
37 # Echonet Property Code
38 attribute be:epc { xsd:hexBinary },
39 # データサイズ[バイト]
40 attribute be:length { xsd:decimal },
41 # get 操作可能: true
42 attribute be:get { xsd:boolean },
43 # set 操作可能: true
44 attribute be:set { xsd:boolean },
45 # get 操作必須: true
46 attribute be:getMandatory { xsd:boolean }?,
47 # set 操作必須: true
48 attribute be:setMandatory { xsd:boolean }?,
49 # 状態変更時アナウンス必須なら true
50 attribute be:statusChangeAnnounce { xsd:Boolean }?,
51 # プロパティの最小値
52 attribute be:min { xsd:hexBinary }?,
53 # プロパティの最大値
54 attribute be:max { xsd:hexBinary }?,
55 element be:name { text }, # プロパティ名
56 element be:unit { text }, # プロパティ値の単位
57 element be:option { # プロパティの設定値
58   attribute be:edt { xsd:hexBinary }, # 設定値
59   element be:name { text }, # この設定値の意味
60
61   # このプロパティ値を実現する赤外線コマンド
62   element pe:irCommand { xsd:hexBinary }?,

```

```

63   # このプロパティ値のときの電力消費パターン
64   element pe:powerConsumptionPattern {
65     # ある電力消費の状態と、その状態が続く時間との組
66     element pe:powerState {
67       ( attribute pe:watt { xsd:float } # 有効電力
68       | attribute pe:voltampere { xsd:float } # 皮相電力
69       | attribute pe:powerfactor { xsd:float } # 力率
70     ),
71     attribute pe:durationSecs { xsd:float } # 継続時間
72   }+
73 }*

```

図 5 RELAX NG による詳細規定 XML のスキーマ

RELAX NG とは、文書スキーマ定義言語の国際標準 (ISO/IEC 19757-2) であり、XML スキーマ定義に一般的に用いられる XML Schema とほぼ同等の機能を持ちながら、より人間にとって扱いやすいという特徴をもつ。本図では RELAX NG Compact Syntax により詳細規定 XML のスキーマを示す。

標準名前空間 be および拡張名前空間 ee は我々の既存研究[4]における提案であり、本稿の提案は ECHONET 名前空間 ep である。この提案では、あるプロパティに対するひとつの設定値を表す be:option タグの子要素として、その設定値を実現するための赤外線リモコンコマンドを表す ep:irCommand タグと、その動作モードにおける電力消費パターンを表す ep:powerConsumptionPattern タグを追加する。提案スマートタップはこれらの情報に基づき、コンセントに接続された家電機器の操作と動作モードの推定を行う。

扇風機プロキシクラス (図 4) の風量プロパティ (22 ~ 44 行目) には、その設定値「弱」「強」を規定するふたつの be:option タグ (26~34・35~43 行目) が存在する。これら be:option タグの子要素として、ep:irCommand タグ (29・38 行目) および ep:powerConsumptionPattern タグ (30~33・39~42 行目) がある。

ep:irCommand タグは、その設定値 (親要素である be:option が規定する設定値) を家電機器に設定するための赤外線リモコンのコマンドを示す。たとえば図 4 の 29 行目では、この扇風機に赤外線リモコンコマンド 8240BF01FE0000 を送信すると、風量が「弱」に設定されることを規定している。

一方 ep:powerConsumptionPattern タグは、その設定値における電力消費パターンを、子要素である ep:powerState タグの並びとして記述する。たとえば図 4 の 39~42 行目では、この扇風機の風量が「強」のとき、5 秒間の平均消費電力は 40W であることが示されている。また ep:PowerState タグを複数並べることで、たとえば「間欠運転モードでは、100W で 1 分間動作したのち 1W で 14 分間待機する」という動作を繰り返すといった、より複雑な電力消費パターンも表現できる。

以上の詳細規定 XML に対する拡張を用いることで、本稿における提案アーキテクチャでは、スマートタップが機器のリモコン制御と動作モード推定を行う。これにより提案アーキテクチャは、既存研究では不可能であった家電プロキシからのソフトウェア的な家電機器の監視・制御を実現する。なお参考のため、本稿における拡張を含む詳細規定 XML のスキーマを図 5 に示す。

4. 実験

本章では，提案アーキテクチャの実現可能性を示す実験について報告する．

実験にあたり，提案アーキテクチャを構成するスマートタップ，家電認識 Web サーバ，詳細規定サーバを用意した．スマートタップの実装には 32bit マイコンである mbed NUCLEO-F401RE，汎用 PC，USB 赤外線リモコン[15]および ECHONET Lite ライブラリである OpenECHO[7]を用いた．家電認識 Web サーバはインターネット上のサーバとし，我々の家電認識手法[14]によって表 1 に示す機器を認識可能とした．詳細規定サーバはインターネット上の HTTP サーバを用い，表 1 に示す機器の家電プロキシクラスを表す詳細規定 XML を公開した．このうちテレビおよび HDD レコーダの詳細規定 XML では，電源操作（オン/オフ），チャンネル設定などのプロパティを規定し，`ep:irCommand` タグにより赤外線リモコンのコマンドを記述した．また扇風機およびドライヤーの詳細規定 XML では，動作モード（風量または出力）プロパティを規定し，`ep:powerConsumptionPattern` タグにより各動作モード設定値における機器の電力消費パターンを記述した．

さらにスマートタップの動作を確認するため，我々の ECHONET Lite 提案拡張に対応した ECHONET Lite コントローラ（以下コントローラと記す）を実装した．このコントローラでは，詳細規定 XML の `be:properties` タグおよび `be:option` タグを参照し，各プロパティの操作ボタンを自動的に生成するものとした．たとえば図 4 の例であれば，動作状態プロパティに対する「オン」「オフ」ボタンと，風量プロパティに対する「弱」「強」ボタンを自動的に生成するよう実装した．

4.1. 実験方法

実装したスマートタップとコントローラおよび表 1 の家電機器を用い，以下の手順で実験を行った．

表 1 実験に用いる家電機器

機器の種類	メーカー	機種	詳細規定 XML で提供する操作
テレビ	東芝	REGZA 22A2	赤外線リモコンによる電源操作（オン/オフ） チャンネル変更 音量変更
HDD レコーダ	東芝	REGZA D-BZ510	赤外線リモコンによる電源操作（オン/オフ） チャンネル変更 録画操作（開始/停止）
扇風機	東芝	F-LN5	消費電力パターンによる動作モード推定（オフ/弱/中/強）
ドライヤー	日立	HD-N1240	消費電力パターンによる動作モード推定（Dry/Set/Cool）

- (1) コントローラとスマートタップの起動
- (2) 家電機器の電源プラグをスマートタップのコンセント口へ接続し，家電機器を電源オン
- (3) コントローラ画面からの家電機器の操作（テレビと HDD レコーダのみ）
- (4) 動作モードの推定結果の確認（扇風機とドライヤーのみ）
- (5) 家電機器の電源オフとタップからの取り外し

4.2. 実験結果

はじめに，手順(1)におけるコントローラの画面を図 6 に示す．図 6(a)では，画面中ほどの「127.0.0.1」「ECHONET Lite コントローラ」という部分にコントローラ自身が，画面下部の「192.168.255.199」「家電認識スマートタップ」という部分にスマートタップが表示されている．

続いて，家電機器としてテレビを用いた場合の手順(2)におけるコントローラ画面を図 6(b)に示す．画面最

Name	Icon	Class Code	Instance Code	Properties
▼ 127.0.0.1				
ECHONET Lite コントローラ		0x05ff	0x01	動作状態: ON
▼ 192.168.255.199				
家電認識スマートタップ		0x0f01	0x01	動作状態: ON

図 6(a) 実験手順(1)におけるコントローラの画面

Name	Icon	Class Code	Instance Code	Properties
▼ 127.0.0.1				
ECHONET Lite コントローラ		0x05ff	0x01	動作状態: ON
▼ 192.168.255.199				
▼ 家電認識スマートタップ		0x0f01	0x01	動作状態: ON
テレビプロキシ (REGZA A22 Series)		0x0f20	0x01	即時消費電力(W): 100 動作状態: ON 積算消費電力(Wh): 0 音量を下げる 音量を上げる 前のチャンネル 次のチャンネル 電源On/Off

図 6(b) 手順(2)におけるコントローラの画面（テレビの場合）

Name	Icon	Class Code	Instance Code	Properties
▼ 127.0.0.1				
ECHONET Lite コントローラ		0x05ff	0x01	動作状態: ON
▼ 192.168.255.199				
▼ 家電認識スマートタップ		0x0f01	0x01	動作状態: ON
扇風機プロキシ (TOSHIBA F-LN5)		0x0f22	0x01	即時消費電力(W): 34 動作状態: ON 積算消費電力(Wh): 3 風量: 中

図 6(c) 手順(2)におけるコントローラの画面（扇風機の場合）

下部にテレビを表す ECHONET 家電プロキシオブジェクトが表示されており、このオブジェクトのもつ Set 要求が可能なプロパティが、画面右下の「音量を下げる」ボタンなどとして表示されている。

手順(3)において、これらコントローラ画面上のボタンを操作したところ、実際にタップに接続されたテレビや HDD レコーダが操作された。すなわち ECHONET Lite 非対応の家電機器を、ECHONET Lite プロトコルを介して操作することができた。これは、コントローラ上のボタン操作が、ECHONET プロパティの set 操作としてコントローラからスマートタップに伝わり、さらにタップの有する赤外線リモコンを介して、テレビやレコーダへの赤外線リモコンコマンドとして送信され実現したものである。

また、家電機器として扇風機を用いた場合の手順(2)におけるコントローラ画面を図 6(c)に示す。画面最下部に扇風機を表す ECHONET 家電プロキシオブジェクトが表示されており、このオブジェクトのもつ Get 要求が可能なプロパティが、画面右下の「風量：中」として表示されている。これは、動作モードプロパティへの Get 要求を受け取ったスマートタップが、扇風機の詳細規定 XML を解釈し、そこに示された動作モードごとの電力消費パターンに現在の消費電力を照合することで推定した動作モード「中」が、コントローラの画面に表示されているものである。

以上により、提案アーキテクチャによる既存家電機器のスマート家電化、とくに既存手法では不可能であった、赤外線リモコンによる set 系操作が実現可能であることが確認された。

5. おわりに

本稿では、我々の ECHONET Lite に対する提案拡張 [4]および家電認識スマートタップ [4]をさらに拡張し、新たな家電認識スマートタップのアーキテクチャを提案した。提案アーキテクチャは、HEMS 非対応の既存家電機器とそのスペックとを家電認識 Web サーバによって紐付け、さらに赤外線リモコンによる家電機器の制御と、消費電力パターンに基づく家電機器の動作状態推定を行うことで、これまで不可能であった set 系操作を含む、既存家電機器のスマート家電化および HEMS 対応化を実現するものである。本稿では実験により、提案アーキテクチャの実現可能性を検証した。

今後の課題としては、より実用的な環境での利用を検証することが挙げられる。また、家電機器の動作モード推定をスマートタップだけでなく家電認識 Web サーバと連携して行うことや、詳細規定 XML にセマンティック Web とオントロジーの概念を導入することも考えられる。

参 考 文 献

- [1] Japan: Submission of Japan's Intended Nationally Determined Contribution (INDC) (online), available from http://www4.unfccc.int/submissions/INDC/Published%20Documents/Japan/1/20150717_Japan's%20INDC.pdf (accessed 2016-01-09).
- [2] エコネットコンソーシアム：ECHONET Lite 規格書 Ver.1.12（日本語版），エコネットコンソーシアム（オンライン），入手先 http://www.echonet.gr.jp/spec_v112_lite（参照 2016-01-09）。
- [3] エコネットコンソーシアム：APPENDIX ECHONET 機器オブジェクト詳細規定 Release G Revised，エコネット規格（一般公開），エコネットコンソーシアム（オンライン），入手先 http://www.echonet.gr.jp/spec_object_rg_revised（参照 2016-01-09）。
- [4] 松石浩輔，沼尾雅之：ECHONET Lite プロトコルに対応した家電認識スマートタップの提案，マルチメディア，分散，協調とモバイル (DICOMO2015) シンポジウム論文集，pp.217-223 (2015)。
- [5] 経済産業省：スマートハウス標準化検討会中間取りまとめ（案），経済産業省（オンライン），入手先 <http://www.meti.go.jp/press/2011/02/20120224007/20120224007-2.pdf>（参照 2016-01-09）。
- [6] 大和田茂：スマートハウスの『いじれる化』を実現するための Kadecot プロジェクト概要，情報処理学会論文誌コンシューマ・デバイス&システム (CDS)，vol.2, no.3, pp.16-22 (2012)。
- [7] SonyCSL：OpenECHO，GitHub（オンライン），入手先 <https://github.com/SonyCSL/OpenECHO>（参照 2016-01-09）。
- [8] SonyCSL：Kadecot Project Page（オンライン），入手先 <http://kadecot.net/>（参照 2016-01-09）。
- [9] 郷司飛馬，寺澤卓也，羽田久一：ECHONET Lite に準拠したテストツールの設計と実装，情報処理学会第 75 回全国大会講演論文集，vol.2013, no.1, pp.215-216 (2013)。
- [10] Saitoh, T., Aota, Y., Osaki, T., et al.: Current Sensor Based Non-intrusive Appliance Recognition for Intelligent Outlet. *Proc. The 23rd International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC 2008)*, pp.349-352 (2008)。
- [11] 伊藤雅仁，大亦寿之，井上智史ほか：消費電力波形の特徴を利用した家電機器検出手法と制御システム。情報処理学会論文誌，vol.44, no.1, pp.95-105 (2003)。
- [12] 加藤丈和，豊村鉄男，山崎達也：情報・エネルギー統合ネットワークのための電力センシング情報からの家電認識とその応用。電子情報通信学会技術研究報告，vol.108, no.399, pp.133-138 (2009)。
- [13] Ridi, A., Gisler, C. and Hennebert, J.: A Survey on Intrusive Load Monitoring for Appliance Recognition. *Proc. 2014 22nd International Conference on Pattern Recognition (ICPR2014)*, pp.3702-3707 (2014)。
- [14] 服部陽彦，沼尾雅之：電源環境の変化に強い家電認識手法の提案。情報処理学会研究報告，Vol. 2012-HCI-150, No.1, pp.1-6 (2012)。
- [15] BitTradeOne：USB 接続赤外線リモコン KIT（オンライン），入手先 <http://bit-trade-one.co.jp/BTOpicture/Products/005-RS/>（参照 2016-01-11）。