

IPv6 情報家電の実装と運用

石原 丈士†
上原 清彦†

米山 清二郎†
中村 公彦†

永見 健一†
榎原 基也†

† 株式会社東芝 研究開発センター

† 株式会社東芝 家電機器社

* 〒212-8582 川崎市幸区小向東芝町 1

株式会社東芝 研究開発センター 通信プラットホームラボラトリー

TEL: 044-549-2230

Email: takeshi3.ishihara@toshiba.co.jp

概要

家庭におけるインターネットは「常時接続」と「ブロードバンド」をキーワードとする新しい段階に入った。その中で家電機器のインターネット対応が注目を集めている。我々は、家電機器をIPv6を用いてインターネットに接続するための手法を考察し、それに基づき冷蔵庫と電子レンジを試作した。さらに、試作した機器をグローバルなIPv6網に接続し、試験運用を行った。本稿では、家電機器の通信方式の考察と試作機の説明を行い、試験運用の結果を報告する。

キーワード:

情報家電、IPv6、ECHONET、ホームネットワーク

1 はじめに

家庭におけるインターネットの発展は、Web ページの閲覧に代表された初期の普及段階を終え、「常時接続」と「ブロードバンド」をキーワードとする新しい段階に入った。これに伴い、ピア・ツー・ピア アプリケーションの登場や IPv6 などの新しい技術が導入され始めている。その中で家電機器のインターネット対応が注目されている。我々は、IPv6 を用いて家電機器をインターネットに接続するための手法を考察し、冷蔵庫と電子レンジを試作した。さらに、試作した機器をグローバルな IPv6 網に接続し、試験運用を行った。本稿では、家電機器の通信方式について考察し、試作品の説明を行い、試験運用の結果を報告する。また、情報家電が持つ課題について考察する。

2 情報家電の通信アーキテクチャ

本章では、情報家電の通信アーキテクチャについて議論する。

2.1 現状

現在市販されている家電機器の多くはすでにマイコンを搭載している。このマイコンは複雑な操作の自動化など様々な機能を提供する。しかしながら、制御や管理を行うアプリケーションソフトウェアに対するインターフェイスが統一されて

おらず、製造メーカーや製品ごとに異なるインターフェイスを使用している。このため現在の家電機器を何らかの方法でネットワークに接続しても、汎用的に他の家電機器マイコンと通信することは困難である。

この問題を解決するため、家電機器の管理・制御を目的とする共通インターフェイスやプロトコルが提案されている。例えば、ビデオやテレビといった AV 家電機器向けの HAVi[1] や冷蔵庫や洗濯機といった白物家電機器向けの ECHONET[2] などである。

ECHONET は家庭内の白物家電機器を相互に接続して制御・管理することを目的にした規格である。ECHONET では、家電機器同士を接続する物理媒体からアドレス体系や家電機器情報を管理するオブジェクト、その上で動作するソフトウェアに提供する API まで定義する。

2.2 インターネットを介した情報の伝達と制御

本節では、家電機器内部で管理されている情報をインターネットを介して伝達する方法および家電機器をインターネットを介して制御する方法を議論する。前述の ECHONET では、情報管理方法および制御プロトコルは閉じた空間を前提に規定されており、インターネットとは ECHONET ゲートウェイを介して接続するように規定されている。したがって、現在のバージョンではインターネットからの直接的な通信は考慮されておらず、何らかの方法を新たに定義する必要がある。なお、本稿では考察対象を冷蔵庫や電子レンジなどの白

物家電機器に限定する。

2.2.1 ECHONET over TCP/UDP/IP

ECHONET フレームの全部もしくは一部を、TCP、UDP、IP のいずれかのレベルでカプセル化して伝達し、制御する方法である(図 1)。それぞれに利点と欠点があり、簡単にまとめると以下ようになる。

IP でカプセル化 三方式の中でカプセル化に伴うオーバーヘッドが最も少ない。ただし、信頼性の保証やアプリケーションの多重化などは上位のレイヤで行う必要がある。

UDP でカプセル化 カプセル化に伴うオーバーヘッドは IP よりは多いが TCP よりは少ない。アプリケーションの多重化は UDP のポート番号を使って実現できる。パケットの誤りは UDP チェックサムを用いて検出可能であるが、到達性の保証やパケットの順序管理などは上位レイヤで行う必要がある。

TCP でカプセル化 カプセル化に伴うオーバーヘッドが三方式の中でもっとも大きい。毎回コネクションの確立と終了を伴うため効率が悪くなる可能性が高い。アプリケーションの多重化は TCP のポート番号を使って実現でき、信頼性も保証される。

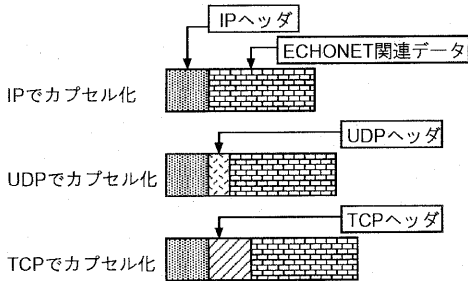


図 1: ECHONET 関連データのカプセル化三方式の比較

上記三方式の ECHONET 関連データ部分は、ECHONET フレーム全体をカプセル化する方法とフレームの一部を抽出してカプセル化する方法がある。ECHONET フレーム全体をカプセル化する場合には、一つのパケットの中にノードを識別する ECHONET アドレスと IP アドレスを併せ持つが、処理が単純なため低コストで実現できる。一方、ECHONET フレームの一部を抽出する場合、フレームの解析が必要な上、互換性問題が発生する可能性がある。

2.2.2 SNMP を利用する方法

家電機器に対応する MIB を定義し、SNMP を利用して情報の伝達と制御を行う方法である。この方法を利用すると、現在広く利用されている SNMP 資産を利用することができる。また、ECHONET オブジェクトを MIB にマッピングするこ

とも可能なため、ECHONET 資産も活用できる可能性がある。また SNMP のトラップは、センサーを有する家電機器などの接続を考慮した場合に有利である。

2.2.3 HTTP を利用する方法

現在インターネット上で最も利用されているプロトコルの一つが HTTP であり、様々なネットワーク環境で利用できることが大きな利点である。この方式では、サーバ側に適切なプログラムを用意することで、Web ブラウザが動作する任意のクライアントに対して家電機器制御インターフェイスを提供することができる。さらに単純な情報伝達にとどまらず HTTP 上でリモートプロシジャールを実現する方法も提案されており、前述の二方式より複雑な操作が実現できる可能性もある。

しかしながら、HTTP は典型的なリクエスト・レスポンス型のプロトコルであり、イベントドリブン型の通信が要求されるセンサーなどの機器に利用することは難しい。

3 試作システム

本章では、第 2 章の議論を基に、我々が今回試作した冷蔵庫と電子レンジの実装について述べる。

3.1 システム概要とプロトコル選択

今回我々が試作したシステムは、冷蔵庫と電子レンジおよびそれを管理するサーバから成る。冷蔵庫と電子レンジは独立した IPv6 ノードとして機能し、運転情報などを管理サーバに送信する。管理サーバは冷蔵庫と電子レンジから受信したデータのデータベース化、冷蔵庫温度の定期的な調査、利用者に対する蓄積データの可視化機能を持つ。

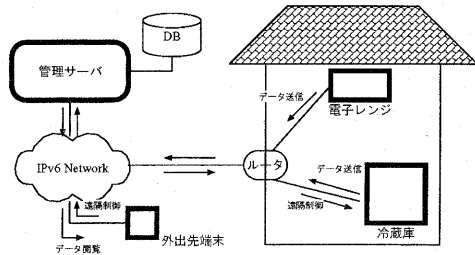


図 2: 試作したシステムの概要

冷蔵庫や電子レンジと管理サーバ間のデータ伝達には ECHONET over TCP/UDP を利用する。さらに [3] にある通り ECHONET フレームに特別な加工はせず、そのまま TCP/UDP のペイロードとする。第 2 章で述べた三方式のうちこの方式を採用した理由は、SNMP や HTTP を利用する方法は各家電ノードでサーバソフトウェアを起動する必要があり、現状では ECHONET over TCP/UDP 方式よりも多

くのリソースが必要だからである。また、ECHONET over IP は、アプリケーションに対する汎用性の提供が煩雑なために採用しない。

冷蔵庫ではトランスポートプロトコルとして TCP を採用し、ECHONET over TCP を用いてデータを送受信する。TCP を採用することで、管理サーバや遠隔コントローラが確実に冷蔵庫の状態管理や操作を行うことができる。一方、電子レンジでは UDP を採用し、ECHONET over UDP を用いる。電子レンジはデータ送信機能しか持たないため、冷蔵庫ほど信頼性を必要としないと判断したからである。

3.2 冷蔵庫

今回試作した冷蔵庫について述べる。

3.2.1 概要

機能

今回試作した IPv6 冷蔵庫の機能を以下に示す。

- 冷蔵庫内部画像の閲覧
冷蔵庫内部にカメラを取り付け、冷蔵庫のタッチパネルおよび Web ブラウザ経由で冷蔵庫内部の様子を閲覧可能。
- 遠隔制御機能
Web ブラウザを用いて冷蔵庫にアクセスし、温度設定など冷蔵庫の動作に関するパラメータを遠隔から設定可能。また、Web ブラウザを用いずに ECHONET over TCP を用いて直接冷蔵庫にコマンドを送ることによっても設定可能。
- 遠隔監視機能
冷蔵庫で発生したイベント（ドアの開閉、ドアの開放など）をあらかじめ指定した特定のアドレスに対して通知可能。
- 電子メール機能
SMTP を用いて電子メールを受信し、タッチパネル上に表示する。送信も可能だが入力インターフェイスが未実装である。

ネットワーク部概略

冷蔵庫の通信・制御部分の物理的な構造およびプロトコルスタックを図 3 に示す。冷蔵庫側マイコンと PC ボードとの間は RS232C で接続し、ECHONET を用いて情報を交換する。PC ボードは ECHONET フレームと ECHONET over TCP のフォーマット変換を行う。また、Web ブラウザ経由の制御を行うため HTTP サーバが動作する。制御は冷蔵庫 HTTP サーバに用意した CGI プログラムが行い、HTTP はユーザインターフェイスの提供手段として利用する。さらに、冷蔵庫上で電子メールを受信可能にするために SMTP サーバが動作する。

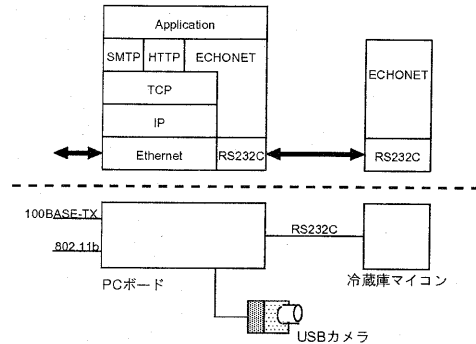


図 3: 冷蔵庫の通信・制御部分の構造

| | |
|-----------|-------------------------------------|
| CPU | intel Celeron 400MHz |
| Chipset | intel 440BX |
| VideoChip | SMI Lynx 721 |
| Memory | 64MB |
| HDD | 2GB |
| NIC | RTL8139C, PCMCIA 802.11b カード |
| 表示装置 | タッチパネルつき SVGA 液晶ディスプレイ |
| 庫内カメラ | OmniVision OV511 互換チップ使用 USB カメラ |

表 1: 冷蔵庫 PC ボードのスペック

3.2.2 ハードウェア

使用した PC ボードのスペックを表 1 に示す。有線と無線の 2 つのネットワークインターフェイスは、使用環境に応じて使い分けている。今回の PC ボードは家電として見た場合は非常にハイスペックであるが、GUI の提供と複数のデーモンプログラムによるサービス提供を行うことを考慮すると妥当なスペックだと言える。

3.2.3 ソフトウェアのバージョンと実行時サイズ

使用したソフトウェアを表 2 に示す。IPv6 スタックは OS に含まれていたものをそのまま使用している。

タッチパネルの GUI アプリケーションは Gtk+ を用いて C 言語で記述されている。画面を構成する Widget 数は約 120

| 項目 | 名称 | バージョン |
|---------------|-----------------|-------|
| OS | NetBSD | 1.5X |
| Window System | X Window System | 4.1.0 |
| Toolkit | Gtk+ | 1.2 |
| httpd | mini.httpd | 1.15c |

表 2: 冷蔵庫 PC ボードのソフトウェア

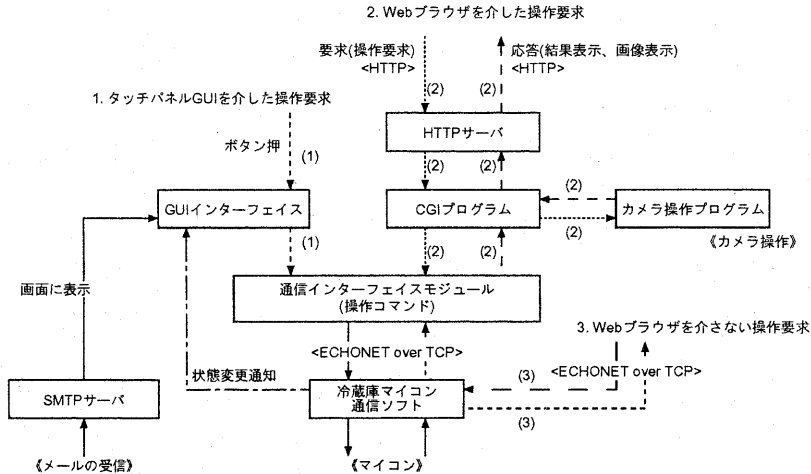


図 4: 冷蔵庫ソフトウェアの構成と処理の流れ

個であり、起動直後のメモリ使用量は約 8MB である。GUI アプリケーションは Web ブラウザとして Mozilla を利用しており、Mozilla 動作時には 20MB 以上のメモリを使用する。通信関係のソフトウェアでは、以下の機能を提供するデモンを C 言語で作成した。

- 冷蔵庫マイコンと RS232C を用いて通信を行う
- ECHONET フォーマットのデータを TCP を用いてインターネット経由で送受信する

このデモの実行時サイズは約 230KB*1 である。また、このデモンに対して ECHONET フォーマットのメッセージを送信するためのコマンドも C 言語で作成した。このコマンドの実行時サイズは約 220KB である。

3.2.4 ソフトウェアの構成と処理の流れ

図 4 に冷蔵庫 PC ボードのソフトウェアの構成と処理の流れを示す。SMTP サーバは単純にメールを受信しているだけなので、ここでは触れない。冷蔵庫 PC ボードのソフトウェアと冷蔵庫マイコンの連携は、以下の場合に発生する。

1. 冷蔵庫 GUI を介した冷蔵庫操作
タッチパネル上の操作ボタンを押すことによって生じる操作
2. Web ブラウザを介した冷蔵庫操作
Web ブラウザから冷蔵庫 PC 上の HTTP サーバにアクセスし、CGI プログラムを実行することによって生じる操作
3. Web ブラウザを介さない冷蔵庫操作および自発的なデータの送信

*1 ライブラリ等にはいっさい手を加えていない。

冷蔵庫 PC が ECHONET over TCP に従った制御メッセージを直接受信した場合に発生する操作。もしくは、冷蔵庫がマイコンの状態変化に応じて管理サーバに対して自発的にデータを送信する場合。

なお利用者から見た場合、上記操作方法の違いによる制約は無く、全て同じ機能を提供する。以下、それぞれの操作方法に対して説明を加える。なお以下の番号は図 4 中の番号に一致する。

1. 冷蔵庫 GUI を介した操作

冷蔵庫の画面上に配置された操作ボタンを利用者が押すことによって、冷蔵庫に対する操作が実行される。操作実行時には操作コマンドを起動し、起動されたコマンドが冷蔵庫マイコン通信ソフトに対して操作要求を出す。操作要求は ECHONET のフォーマットに従っている。

2. Web ブラウザを介した操作

冷蔵庫上に HTTP サーバと CGI プログラムを用意し、インターネット上から Web ブラウザを用いてアクセスすることにより冷蔵庫に対する操作を行う。CGI プログラムは操作コマンドを起動し、起動されたコマンドが冷蔵庫マイコン通信ソフトに対して操作要求を出す。CGI プログラムは操作に対する結果として、冷蔵庫の現在の状態をブラウザに返す。CGI によって起動されたコマンドがマイコン通信ソフトに対して出す操作要求は ECHONET フォーマットに従っている。

3. Web ブラウザを介さない操作および自発的なデータの送信

ECHONET over TCP で直接受信した場合にも冷蔵庫に対する操作が実行される。このような操作は、家電機器管理サーバなど操作内容が自動化された環境下で起こり、今回の試

作でも管理サーバからの定期的な状態監視を実現するために利用している。

また、冷蔵庫マイコンとは連携しないが、庫内画像の撮影は Web ブラウザで冷蔵庫 PC 上の HTTP サーバにアクセスし、CGI プログラムを介して行う。この場合、CGI プログラムからカメラ操作プログラムを起動し、結果として撮影した画像を含む Web ページをブラウザに返す。

3.3 電子レンジ

今回試作した電子レンジについて述べる。

3.3.1 概要

機能

今回試作した IPv6 電子レンジは遠隔監視機能を持つ。この機能は、電子レンジのドアを開閉した時および調理を行った時にデータをサーバに送信する。送信されるデータは、ドアの状態と開閉時刻、調理内容・調理設定時間・調理開始時刻などである。

ネットワーク部概略

電子レンジの通信部分の物理的な構造およびプロトコルスタックを図 5 に示す。冷蔵庫と異なりデータの送信専用である。

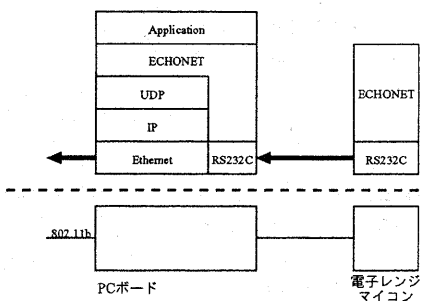


図 5: 電子レンジの通信・制御部分の構造

3.3.2 ハードウェア

使用した PC ボードのスペックを表 3 に示す。動作するソフトウェアを考えると、今回のスペックよりも低いスペックでも動作すると推測される。

3.3.3 ソフトウェアのバージョンと実行時サイズ

電子レンジの実装では、OS に NetBSD 1.5.2 を使用し、電子レンジマイコン通信部分は C 言語で新規に作成した。このプログラムの実行時サイズは約 200KB である。

| | |
|---------|--------------------|
| CPU | NS GX1 300MHz |
| Chipset | NS CX5530A |
| Memory | 64MB |
| 補助記憶装置 | 128MB コンパクトフラッシュ |
| NIC | PCMCIA 802.11b カード |

表 3: 電子レンジ PC ボードのスペック

3.3.4 ソフトウェアの構成と処理の流れ

図 6 に電子レンジソフトウェアの構成と処理の流れを示す。電子レンジはデータ送信専用であり、電子レンジマイコン通信ソフトは単純にパケットフォーマットの変換とインターネット上の管理サーバへの送信のみを行う。

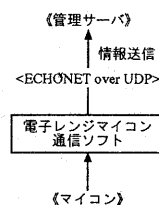


図 6: 電子レンジソフトウェアの構成と処理の流れ

3.4 家電機器管理サーバ

3.4.1 概略

冷蔵庫と電子レンジに対する家電機器管理サービスを提供するサーバも合わせて試作した。管理サーバの機能を以下に示す。

- 家電機器監視機能
冷蔵庫に対して定期的にデータ取得要求を送信し、温度変化を監視する。
- データベース機能
冷蔵庫と電子レンジから送られる各種情報をデータベース化し、保存する。
- Web サーバ機能
データベースに蓄積されたデータを加工して、Web ページとして表示する。温度変化、ドアの開閉、障害検出はグラフ化し、電子レンジの調理履歴は表にして表示する。

3.4.2 ソフトウェア

管理サーバのソフトウェアスペックを表 4 に、構成を図 7 に示す。HTTP サーバは Apache*2 であるが、Apache は 1.3

*2 <http://httpd.apache.org/>

| 項目 | 名称 | バージョン |
|-------------|---------------|--------------------|
| OS | FreeBSD | 4.4-RELEASE |
| HTTP Server | Apache PHP | 1.3系、2.0系 4.0.6 |
| Database | PostgreSQL | 7.1.3 |

表 4: 家電機器管理サーバのソフトウェア

系と 2.0 系を併用している。IPv6 と SSL との親和性を考慮して 2.0 系で TCP コネクションの終端と 1.3 系へのプロキシを行い、PHP^{*3}の処理系として 1.3 系を利用している。

家電機器通信サーバは C 言語で新規に作成し、実行時サイズは約 2MB である。家電機器通信クライアントは冷蔵庫に使用している操作コマンドと同じ物である。

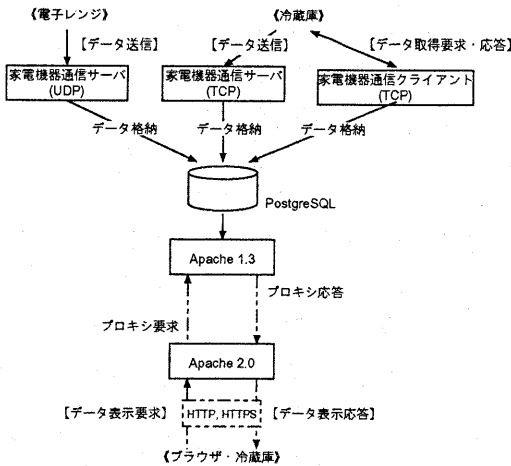


図 7: 管理サーバのソフトウェアの構成と処理の流れ

4 試験運用報告

4.1 Net.Liferium

4.1.1 概要

2001 年 12 月 15 日 (土) と 16 日 (日) にパシフィコ横浜で行われた入場者数約 2 万人規模のイベントである^{*4}。冷蔵庫と電子レンジを展示し、主に説明員が操作する形で運用した。運用期間は 14 日 (金) 17:30 から 16 日 (日) 17:00 までである。

^{*3} <http://www.php.net/>

^{*4} <http://www.key3media.co.jp/Net-Life/>

4.1.2 ネットワーク構成

イベント当日のネットワーク構成を図 8 に示す。会場の電波状態が非常に不安定だったため、冷蔵庫は有線 LAN で接続している。

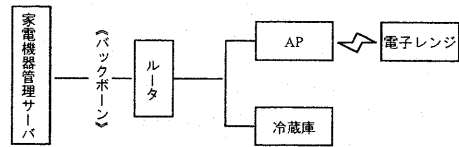


図 8: Net.Liferium のネットワークトポロジ

4.1.3 運用結果

トラフィック測定ができなかったため、サーバに登録されたデータ件数を報告する。会場と管理サーバとを結ぶネットワークのトラブルにより通信できないことがあったが、それ以外はおおむね順調に運用できた。

冷蔵庫

管理サーバが冷蔵庫の温度を 10 分に 1 回監視するように設定して運用した。管理サーバのデータ取得要求に応じて返されたデータにはデータベースレコード 3 件分 (冷蔵庫温度、冷凍室温度、外気温) が含まれている。

この運用期間中に管理サーバに蓄積されたデータの内訳を表 5(a) に示す。

電子レンジ

電子レンジから送信され、管理サーバに記録されたデータの内訳を表 5(b) に示す。表中の「調理モードなど」には調理内容、調理設定時間などが含まれる。なお実際の調理回数は 79 回であった。

4.2 ショールーム展示

2002 年 1 月上旬から 3 月末の予定で、札幌と大手町のショールームに冷蔵庫と電子レンジを展示し、テストを行っている。

4.2.1 ネットワーク構成

ネットワーク構成はアドレス以外は図 8 に示した Net.Liferium のときと同じである。冷蔵庫は電波状態を考慮し有線 LAN を利用している。

4.2.2 運用途中報告

ショールームでの実験では、Net.Liferium と同様に冷蔵庫と電子レンジのドアの開閉記録、電子レンジの調理記録などの動作記録を家電管理サーバで記録している。また、管理サー

| 登録データの種類 | 件数 |
|----------|------|
| 冷蔵室温度 | 259 |
| 冷凍室温度 | 259 |
| 外気温度 | 259 |
| 冷蔵室ドア開閉 | 512 |
| 冷凍室ドア開閉 | 61 |
| 製氷室ドア開閉 | 20 |
| ドア開放警告 | 6 |
| 総データ件数 | 1376 |

(a) 冷蔵庫

| 登録データの種類 | 件数 |
|----------|-----|
| 調理モードなど | 183 |
| ドア開閉 | 150 |
| 総データ件数 | 333 |

(b) 電子レンジ

表 5: Net.Liferium 期間中にサーバに記録されたデータの件数

バから冷蔵庫への温度監視は 10 分に 1 回の割合で設定している。現在、札幌ショールームは運用開始から 1 ヶ月経過したが、順調にデータが記録されている様子が確認できる。

5 今後の課題

5.1 最適な情報伝送方式の決定

今回の実装では、ECHONET over TCP もしくは ECHONET over UDP を採用したが、この方法が最適解とは断定できない。この方式はアドレスの重複という明らかな冗長性を持っており、伝送量を減らすことが可能である。また、SNMP を利用する方法も、SNMP 実装の小型化を行ない家電機器での利用に実用的なサイズになったならば、強力な手段となるはずである。

今後、様々な方法を実装し比較することにより、最適な伝送方式を決定しなければならない。

5.2 リンクレイヤの問題

IPv6 に限らず情報家電一般に存在する問題として、リンクレイヤの選択問題がある。リンクレイヤは有線系と無線系に大別でき、それぞれ特徴がある。

5.2.1 有線リンクレイヤの利用

有線系のリンクレイヤで主要なものは次の 3 つである。1 つはカテゴリ 5 などを利用する既存の LAN であり、1 つは電話線を利用するもの、もう 1 つは電灯線を利用するものである。

既存の LAN や HomePNA などを利用する場合、利点は既存ネットワークとの親和性と帯域の広さであり、欠点は電源ライン以外のケーブルが必要になることである。インターネットへ接続することを考えると、既存 LAN との親和性は重要であり、またアプリケーションによっては広帯域が必要になる可

能性もある。しかし、家電機器のインテリア性と利便性を考えた場合、複数本のケーブルが出ている状況は好ましくない。

一方、電灯線を利用するネットワークの場合、利点は通信専用のケーブルが不要なことであり、欠点は新技術のため安定性と既存 LAN との親和性に欠ける可能性があることである。

5.2.2 無線リンクレイヤの利用

無線系のリンクレイヤを家電機器で使用する場合には、解決すべき問題が多く残されている。例えば、使用するリンクによっては無線チャンネルや ID などのリンク固有情報を静的に与える必要があるかもしれない。

また、現在広く利用されている 802.11b の無線 LAN は、暗号化方式である WEP の脆弱性が指摘されているし、Bluetooth は 1 つのピコネットに接続可能なノード数が少ないため、あらゆる家電機器が接続する状況への対応策を考慮しておく必要がある。

5.2.3 リンクレイヤ選択の基準

最終的なリンクレイヤの選択は家電上に実装されるアプリケーションが要求する帯域とセキュリティレベル、家電機器のインテリア性と利便性を考慮して選択することになる。例えば動画を扱う機器では高速なリンクが要求されるし、逆に散発的に小さなデータを送受信する機器では低速なリンクでも十分である。さらに、頻繁に移動する機器では無線系が有利であるし、強度なセキュリティが必要であれば、通信範囲が目に見える有線系が有利であろう。

5.3 セキュリティ

家電機器をネットワークに接続する際には、パソコンを接続する場合と同程度にセキュリティが重要な問題となる。しかし、暗号化や認証のための演算は高コストであるため、全ての家電機器が対応できるわけではない。実際に製品化する際には、通信相手、通信内容、実行時コスト、製造コストといっ

た要因を考慮して慎重に決定しなければならない。

また、通信内容を暗号化したり電子証明書を交換したりする場合には、暗号鍵の生成や交換、証明書の扱いが大きな問題となる。任意の相手と安全な通信を行うためには自動鍵交換などが必須であるが、社会インフラとして利用可能になるまでには時間がかかる。

さらにクラッキング対策も重要な問題である。IPv6 ではエンド・ツー・エンドな通信の重要性が説かれているが、そのためにはエンドノードが十分安全でなければならない。特に、今回試作した冷蔵庫のように家電機器でサーバプログラムを起動し、サービスを提供する場合には、十分注意して家電機器を設計しなければならない。

5.4 セキュリティホールとその修正

現在のインターネットの大部分を構成するパソコンではソフトウェアに修正を加えることが比較的容易なため、セキュリティホールが見つかった場合でも迅速に対応可能である。ところが家電機器の場合、ソフトウェアの修正は非常に難しい問題となる。その理由を以下に示す。

- 家電機器では発売後に製品を修正するという概念が一般的ではない
家電機器では発売後に修正を加えることは欠陥と見なされる傾向が強く、修正が困難である。
- ソフトウェアが変更不可能な記録媒体に格納される可能性がある
家電機器ではソフトウェアが ROM に焼き込まれることがある。そのような場合には後から修正できない。もし修正する場合には、機器の使用を中止して部品を交換しなければならない。
- 使用者の知識レベルを想定できない
ソフトウェアの修正が可能な場合でも、すべての使用者が修正する技能を有するとは限らない。修正方法が煩雑では修正してもらえない可能性がある。メーカーが遠隔から修正を加えたり、特定のボタン操作で自動的に修正されるような仕組みが必要である。

以上の理由から、問題を最小限にするソフトウェアの開発、ソフトウェアが修正可能な製品の構造、容易な修正方法の確立が家電機器のインターネット対応には必要不可欠であると言える。

6 おわりに

本稿では、はじめに IPv6 に対応した家電機器の通信アーキテクチャを検討した。そしてそれに基づき、IPv6 に対応した冷蔵庫と電子レンジを試作し、その構成を説明した。また、試作した機器を実際に運用し、その途中経過を報告した。

本研究は通信・放送機構 (TAO) の委託研究テーマ「アプリケーション開発基盤システム及びネットワーク構築システムに係わる研究、IPv6 情報家電ネットワークサービス提供のための情報家電管理に関するアプリケーション開発基盤システムの研究開発」の一環として行ったものである。

本研究で参加しているイベントおよび実証実験は IPv6 普及・高度化推進協議会^{*5} の活動の一環として行ったものである。

参考文献

- [1] HAVi Specification, The HAVi Organization, <http://www.havi.org/>
- [2] エコネット規格書, エコネットコンソーシアム, <http://www.echonet.gr.jp/>, 2001
- [3] 米山 清二郎, 石原 丈士ら, IPv6 情報家電の運用モデルの設計, 電子情報通信学会インターネットアーキテクチャ研究会 第3回研究会, IA2001-18, November 2001

^{*5} <http://www.v6pc.jp/>