

遠隔地にある Bluetooth 機器間の シームレス接続手法の提案

津田 一磨¹ 鈴木 秀和¹ 渡邊 晃¹

概要: Bluetooth に代表される近距離無線技術の発達により、ホームオートメーションの普及が期待されている。今後、宅内にある近距離無線通信機器を外出先から操作したいという要求が高まると考えられる。しかし、このような機器には通信可能範囲に制限があり、外出先から直接操作することができない。本稿では、Bluetooth 機器のハードウェアとソフトウェアの間で交換されるコマンド等をインターネット経由で転送することにより、遠隔地の Bluetooth 機器へ接続する手法を提案する。提案手法によると、ユーザは Bluetooth 機器の位置を意識することなく、一般の Bluetooth アプリケーションを用いて近傍および遠隔地にある Bluetooth 機器とシームレスに接続することができる。

キーワード: Bluetooth デバイス, リモートアクセス, ホームネットワーク

A Proposal of Seamless Connection Method for Bluetooth Devices in Remote Location

KAZUMA TSUDA¹ HIDEKAZU SUZUKI¹ AKIRA WATANABE¹

Abstract: With the progress of short-range wireless technologies such as Bluetooth, home automation system is expected to become popular. In future, a request for a remote control of short-range wireless communication devices in a home from the outside is expected to increase. However, users can not control directly these devices from remote places due to the coverage. In this paper, we propose a connection method for Bluetooth devices in remote places by transmitting Bluetooth device commands exchanged between its hardware and software over the Internet. The proposed method can achieve that, users connect to Bluetooth devices located in neighborhood or remote places seamlessly without their attention of the device location.

Keywords: Bluetooth Device, Remote Access, Home Network

1. はじめに

Bluetooth や ZigBee といった近距離無線通信技術の発達により、ホームオートメーションやセンサネットワークの普及が期待されている。家庭内におけるネットワークには、配線が極力なくてすむこと、ネットワークに参加、離脱するための操作が容易であること、消費電力が少ないことなどの技術が要求される。Bluetooth は、アドホックにネットワークを構築できることや、低消費電力であること、ま

た、周波数ホッピングスペクトラム拡散を用いることにより、同じ 2.4GHz 帯の周波数を用いる端末が近接範囲に多数存在しても互いに干渉しにくい特徴がある。そのため、ホームオートメーションのみならず、コンシューマ向け医療用機器やフィットネス、エンターテインメントなど幅広い用途に適用することができる。

今後、情報家電、情報端末の技術的な進歩によって、家庭の機器が情報化され、Bluetooth などの近距離無線通信機器を搭載した家電機器が普及することが予想される。そのため、宅内の近距離無線機器を外出先から操作したいという要求が考えられる。しかし、このような機器には通信

¹ 名城大学大学院理工学研究科
Graduate School of Science and Technology, Meijo University

可能範囲に制限があり、外出先から直接操作することはできない。

遠隔地に存在する Bluetooth 機器に接続する手法として、Bluetooth の有線拡張方式 [1] や、PUCC (P2P Universal Computing Consortium) を用いた手法 [2] などが提案されている。Bluetooth の有線拡張方式では、離れた各 Bluetooth 機器の近傍に Bluetooth と有線接続インタフェースを持つゲートウェイを設置する。Bluetooth 機器はゲートウェイを仮想的に通信相手として認識し、通信を開始する。この方式では一般の Bluetooth アプリケーションを使用できるという利点がある。しかし、一方のゲートウェイ側に複数の Bluetooth 機器が存在する場合、他方のゲートウェイに同じ数だけの Bluetooth インタフェースが必要になるという課題がある。PUCC を用いた手法では、操作端末である Android 端末は PUCC を実装した Web サーバを介して通信することにより、宅内のセンサデータを閲覧したり、家電機器を操作したりできる。しかしこの手法では、操作端末や接続したい機器の位置に応じて、ユーザは接続手法やアプリケーションを使い分ける必要があるという課題がある。

本稿では、Bluetooth 端末のハードウェアとソフトウェアの間で交換されるコマンド等をインターネット経由で転送することにより、遠隔地にある Bluetooth 機器に接続する手法を提案する。この手法では、ユーザは Bluetooth 機器の位置を意識することなく、一般の Bluetooth アプリケーションを用いて宅外からでも宅内と同じ操作により宅内の Bluetooth 機器へ接続できる。

2. Bluetooth

図 1 に Bluetooth スタックの構成を示す。Bluetooth では、ソフトウェアで構成される上位層を Host、ハードウェアで構成される下位層を Controller と呼ぶ [3]。Host と Controller は HCI (Host Controller Interface) コマンド、HCI イベント、HCI データを交換することにより Bluetooth 通信を行う。HCI は Controller が管理する Bluetooth ハードウェア機能にアクセスするための共通インタフェース方式を上位層に提供する。

HCI コマンドは Host が Controller のハードウェアを制御する際に用いるメッセージであり、制御情報と管理情報を搬送する。HCI イベントは Controller が Host に対して HCI コマンドに対する応答や、他の端末からの通信を通知するために用いるメッセージである。コマンドの応答に用いられるイベントには、どの種類のコマンドにより発生したのかを示す OpCode が含まれる。他の端末からの通信により発生する HCI イベントには、送信元端末の BD ADDR (Bluetooth Device Address) もしくは後述する Handle が含まれる。このイベントは通信に用いられたコマンドの種類に応じて発生する。コマンドを実行し

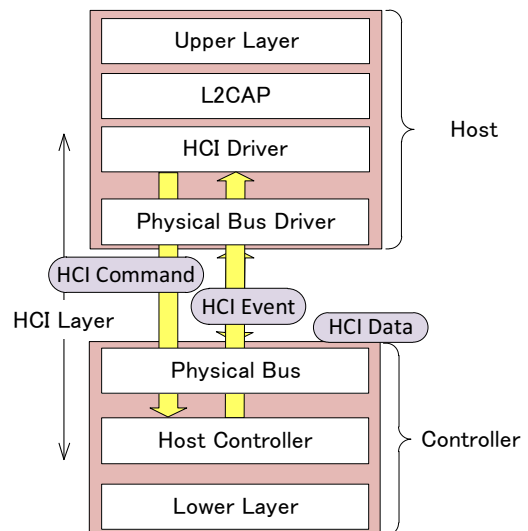


図 1 Bluetooth スタックの構成
 Fig. 1 Structure of Bluetooth stack.

て端末間の接続が完了すると、Host Controller によって Connection Handle*1 が論理リンクとして割り当てられる。Handle は端末がデータを送受信するときに用いる論理リンクを識別するための値である。HCI データは HCI Driver と Host Controller の間でデータを交換する際に用いられる。HCI データには使用する論理リンクを指定するための Handle が含まれる。

3. 既存研究

3.1 Bluetooth の有線拡張方式

Bluetooth の有線拡張方式 [1] は、Bluetooth ネットワークを有線ネットワークで拡張することにより、離れた Bluetooth 機器同士を接続する技術である。図 2 にシステム概要を示す。双方の Bluetooth 機器の付近に通信を中継するゲートウェイを設置する。このゲートウェイは Bluetooth インタフェースと有線接続インタフェースを持ち、有線により接続される*2。各 Bluetooth 機器は近傍のゲートウェイを通信相手の Bluetooth 機器と仮想的に認識して通信を行う。

Bluetooth 機器 A の Host が自身の Controller に HCI コマンドを送信すると、Controller はゲートウェイ A と Bluetooth 通信を開始する。ゲートウェイ A は、Bluetooth 機器 A との通信により発生した HCI イベントを、有線接続ネットワークによりゲートウェイ B へ送信する。ゲートウェイ B は受け取ったイベントの内容からどのようなコマンドにより通信が開始されたかを判別し、同様の通信を Bluetooth 機器 B に対して開始する。有線拡張方式は L2CAP (Logical Link Control and Adaptation Protocol) 層よりも下位に存在する HCI 層における HCI イベントおよ

*1 単に Handle とも呼ばれる。

*2 この方式では有線伝送路の通信メディアを限定していない。

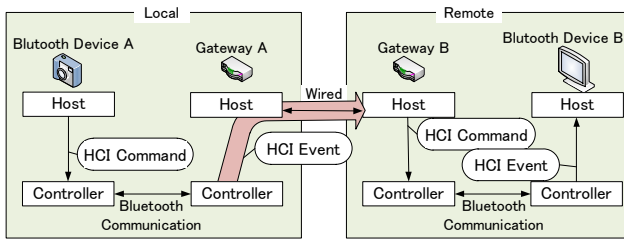


図 2 Bluetooth 有線拡張方式の概要

Fig. 2 Overview of Wired Extension Method for Bluetooth.

び HCI データを転送することから、接続される Bluetooth 機器の L2CAP 層以上のプロトコル層に対して透過な接続システムである。

この方式では、ゲートウェイが通信相手の Bluetooth 機器の代理として動作する。例えばゲートウェイ B 側に複数の Bluetooth 機器が存在する場合、Bluetooth 機器 A はゲートウェイ A を宛先にして通信を行うため、ゲートウェイ B 側に存在するどの端末に対し通信を開始したらよいかかわからないという問題が発生する。また、ゲートウェイは自身の Bluetooth インタフェースと他方のゲートウェイ側の Bluetooth 機器を一对一に関連付けることにより、この問題を解決している。そのため、ゲートウェイの近傍に複数の Bluetooth 機器が存在する場合、イベントおよびデータを正しく転送するためには、ゲートウェイは複数の Bluetooth インタフェースを実装する必要がある。また、宅外から宅内の Bluetooth 機器を遠隔操作したい場合、ユーザはゲートウェイを携帯しなければならない。

3.2 PUCC を用いた手法

PUCC は異なるネットワークに接続された端末を相互に接続し、ユビキタスネットワークを構成する P2P 通信およびオーバーレイネットワーク技術の研究開発、標準化を行なっている団体である [4]。PUCC では、IP および非 IP の下位通信プロトコル上に経路制御やメッセージ制御を行う PUCC プロトコルを定義し [5]、Bluetooth や DLNA (Digital Living Network Alliance) 機器などの既存の異種ネットワーク上にオーバーレイネットワークを形成することにより、様々な機器の相互接続を実現している。ユーザは PUCC アプリケーションを用いることにより、PUCC ネットワーク上の端末と通信を行うことができる。

文献 [2] では、PUCC プロトコルにより異種ネットワークに接続されたデバイスを相互に接続するシステムが提案されている。図 3 に PUCC による Bluetooth 機器への接続シーケンスを示す。Web サーバとセンサゲートウェイに PUCC ミドルウェアを実装し、これらを PUCC ノードと呼ぶ。センサゲートウェイは事前に Bluetooth 機器と接続しておく。Web サーバは PUCC プロトコルを用いて Lookfor メッセージにより、センサゲートウェイに PUCC

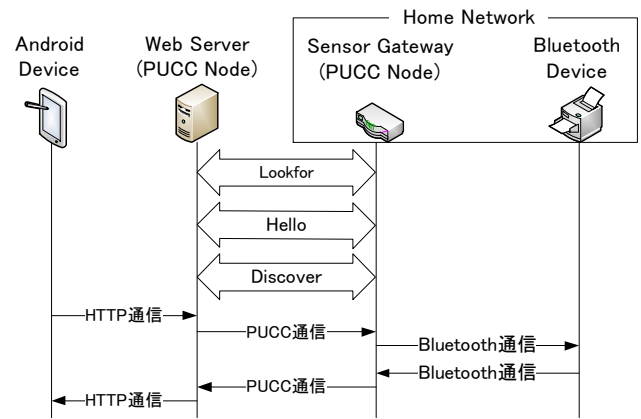


図 3 PUCC による Bluetooth 機器への接続

Fig. 3 Bluetooth device connection procedure.

で用いる ID の対応付けを行い、Hello メッセージによりセンサゲートウェイとの接続を確立する [6]。その後、定期的に Discover メッセージをセンサゲートウェイへ送信することにより、センサゲートウェイが接続している端末のデバイス・サービス情報を取得する。これにより、Web サーバは操作端末である Android 端末からの HTTP 通信による要求にしたがって、センサゲートウェイ配下に存在する端末を操作したり、宅内のセンサデータを取得して Android 端末へ送信したりできる。

この手法では、Android 端末が宅内に存在する Bluetooth 機器と通信する場合は、HTTP を用いて Web サーバを介した通信を行う。しかし、Android 端末が近傍の Bluetooth 機器と通信する場合は、Web サーバを介した通信ではなく、直接 Bluetooth 通信を行うことになる。すなわち、操作端末や接続したい機器の位置に応じて、ユーザは接続手順やアプリケーションを使い分ける必要がある。

4. 提案手法

4.1 概要

既存研究では、宅内に設置するゲートウェイへ複数の Bluetooth インタフェースを実装する必要があったり、ユーザが接続手法やアプリケーションを使い分ける必要があった。そこで本稿ではこれらの課題を解決した新たな接続手法を提案する。提案手法は以下の基本方針に基づいて実現する。

- ユーザは自身の位置や接続したい Bluetooth 機器の場所に関係なく、常に同じ動作で Bluetooth 機器に接続できる。
- ユーザが使用するアプリケーションは特別な機能を持たない一般の Bluetooth アプリケーションとする。
- ユーザは操作端末以外に携帯する機器は持たない。操作端末は主としてスマートフォンを想定し、インターネットに接続できるものとする。
- 遠隔地の Bluetooth 機器に接続するために必要な変更

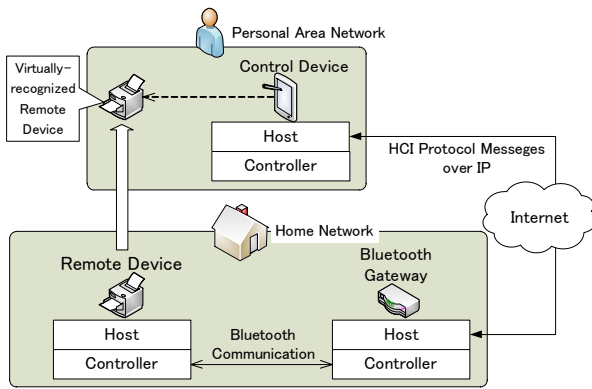


図 4 システム構成
Fig. 4 System configuration.

は、ユーザの操作端末と宅内のみとする。すなわち、インターネット上に特別なサーバを必要としない。

- 操作端末や Bluetooth 機器の台数に応じてハードウェア要件を変更する必要はない。

図 4 にシステム構成を示す。本稿では、宅内に存在する接続対象の Bluetooth 機器を RD (Remote Device)、外出先の操作端末を CD (Control Device) と呼ぶ。CD は直接 RD と Bluetooth 通信できないため、通信を中継するためのゲートウェイ (以下、BGW (Bluetooth Gateway)) を宅内のネットワークに設置する。BGW は Bluetooth インタフェースと Ethernet または Wi-Fi などの IP インタフェースを 1 つずつ実装する。提案手法では、CD の Host と Controller の間で交換される HCI プロトコルメッセージ³をインターネット経由で BGW に転送する。BGW は受信した HCI プロトコルメッセージに応じて宅内の RD と接続する。そのため、CD は BGW の Bluetooth インタフェースを自身のインタフェースとして扱うことができ、遠隔地の RD が近傍に存在しているかのように認識することができる。

4.2 トンネル構築

HCI プロトコルメッセージをインターネット経由で転送するために、CD と BGW の間に IP トンネルを構築する事前準備として、BGW は DDNS (Dynamic DNS) サーバへ FQDN (Fully Qualified Domain Name) とブロードバンドルータのグローバル IP アドレスを登録しておく。CD から RD へ接続できるように、ブロードバンドルータにはポートフォワーディングの設定が行われているものとする。

CD はトンネル構築用アプリケーションにより、BGW の FQDN から取得できるブロードバンドルータの IP アドレスを用いて、BGW と UDP/IP トンネルを構築する。なお、CD と BGW 間はインターネットを経由した通信となるため、事前共有鍵を用いて暗号化および認証を行うもの

*3 本稿では HCI コマンド、HCI イベント、HCI データをこのように総称する。

表 1 トンネルテーブル

Table 1 Tunnel table.

カラム名	サイズ	説明
OpCode	16bit	HCI コマンドの OpCode の値
RD ADDR	48bit	RD の BD ADDR の値
Handle	12bit	リンクを識別する Handle の値
Dest IP address	32bit	IP アドレス
Dest port	16bit	ポート番号

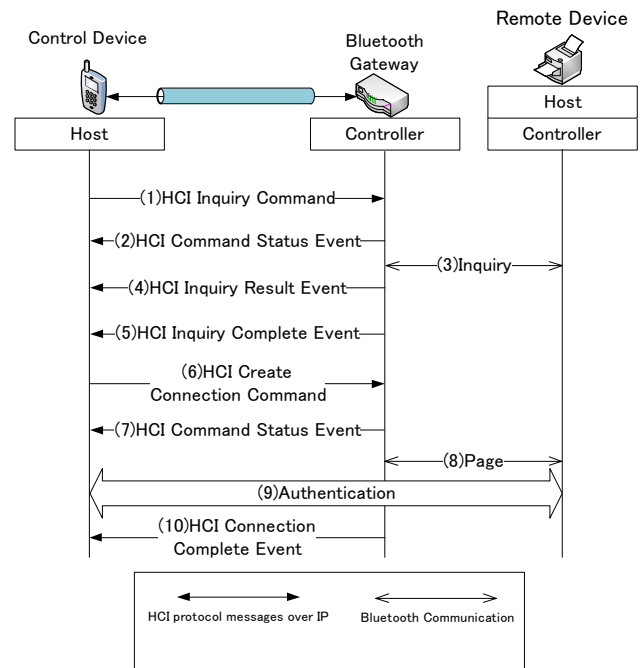


図 5 接続シーケンス

Fig. 5 Connection sequence.

とする。CD と BGW は HCI プロトコルメッセージを転送する際に使用する IP トンネルを決定するため、トンネル構築時にそれぞれ表 1 に示すトンネルテーブルへトンネル構築処理に用いた IP アドレスとポート番号を記録する。

4.3 遠隔地の Bluetooth 機器への接続

図 5 に HCI コマンドと HCI イベントを転送し、CD と RD が接続するシーケンスを示す。以下に CD が一般の Bluetooth アプリケーションを起動して RD と接続するまでの手順を示す。

- (1) Bluetooth アプリケーションは近隣の Bluetooth 端末の探索を開始すると、これにより、CD の Host から CD の Controller に向けて HCI Inquiry コマンドが送信される。このとき、CD は HCI プロトコルメッセージを BGW へ転送するために、この HCI コマンドをフックする。CD はトンネルテーブルを参照して、構築したトンネルの宛先 IP アドレスとポート番号を取得する。CD はこの IP アドレスとポート番号を用いて、フックした HCI コマンドをカプセル化して IP イ

インタフェースより BGW へ送信する。

BGWはこのパケットをデカプセル化してHCI Inquiry コマンドを取り出し、自身の Controller へコマンドを渡す。これにより、BGWが近隣のBluetooth機器を探索する処理が行われる。このとき、BGWは今後発生するHCIイベントをCDへ転送するために、コマンドに含まれるOpCodeをトンネルテーブルへ記録する。

(2) BGWではコマンド実行中を示すHCI Command Status イベントが発生し、BGWのControllerからHostへ渡される。BGWは発生したイベントに含まれるOpCodeを参照し、トンネルテーブルからCDのIPアドレスとポート番号を取得する。この情報を用いてBGWは発生したHCIイベントをカプセル化してCDへ転送する。

CDは受信したパケットからHCIイベントを取り出し、自身のHostへ渡す。このイベントはBGWがデバイス探索中であることを示すものであるが、CDは自身のControllerが近隣のデバイスを探索しているものと見なすことができる。

(3) BGWのControllerはHCIコマンドにしたがってInquiry*4を行う。これにより、BGWはRDのBD ADDRを取得する。

(4) RDのBD ADDRを取得すると、BGWではHCI Inquiry Result イベントが発生する。このイベントにはイベントの種類を示す情報が記載されており、対応するOpCodeを一意に特定することができる。この関係を利用して、トンネルテーブルからカプセル化に必要な情報を取得する。その後、BGWは取得したRDのBD ADDRを通知するため、カプセル化したHCI Inquiry Result イベントをCDへ転送する。

(5) Inquiryが完了すると、HCI Inquiry Complete イベントが発生するため、BGWは(4)と同様の手順によりこのHCIイベントをCDへ転送し、Inquiryの完了を通知する。以上の処理により、CDはRDのBD ADDRを取得することができる。なお、この後にRDの機器名などを取得する処理が行われるが、上記と同じ手順で行われ、Bluetoothアプリケーションに発見されたRDの情報が表示される。

(6) ユーザが接続したいBluetooth機器としてRDを選択すると、CDはRDへ接続要求を行う。CDのHostからControllerへ渡されるHCI Create Connection コマンドをフックして、(1)と同様の手順によりBGWへ転送する。

BGWは受信したHCIコマンドをデカプセル化して、自身のControllerへ渡す。HCI Create Connection コ

マンドにはRDのBD ADDRが含まれているため、このHCIコマンドを受信したBGWは、コマンドに含まれるRDのBD ADDRとOpCodeをトンネルテーブルへ記録する。

(7) BGWは(2)と同様に、RDに接続中であることを示すHCI Command Status イベントをCDへ転送する。

(8) BGWはHCI Create Connection コマンドにしたがってPage*5を行う。これにより、BGWとRDの間でHCIの下位層でリンクが確立される。

(9) デバイス認証を行うため、CDとRDはAuthenticationを実行する。CDとRDが初めて接続する場合はペアリング処理を行う。PINコードによる認証を行ってペアリングが完了する。

(10) Authenticationの後、RDとの接続が確立されると、BGWのControllerでHCI Connection Complete イベントが発生する。このイベントに含まれるRDのBD ADDRを用いて、BGWはトンネルテーブルを参照し、構築したトンネルのIPアドレスを得る。このIPアドレスを宛先にして、BGWはHCI Connection Complete イベントをカプセル化してCDへ転送する。

以上の処理により、CDは遠隔地にあるRDとの接続を完了し、データ通信を開始することができる。また、BGWはRDとの接続を確立すると、HCIデータを転送する宛先を決定するために、HCI Connection Complete イベントに含まれるHandleをトンネルテーブルへ記録しておく。

4.4 データ通信

ここでは、CDとRD間のデータ通信手順を示す。図6にCDからRDへのデータ通信シーケンスを示す。

(1) CDはRDへデータを送信する際、まずBGWが受信

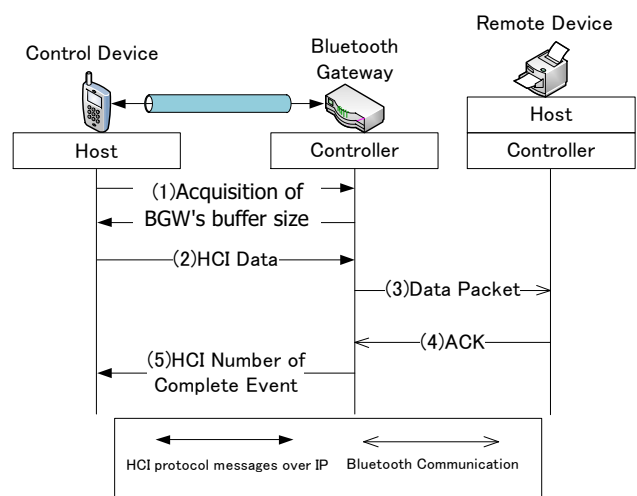


図6 CDからRDへのデータ通信シーケンス

Fig. 6 Data communication sequence from CD to RD.

*4 デバイス探索のこと。

*5 デバイス呼び出しのこと。

可能なバッファサイズを確認する。

- (2) 取得したバッファサイズに基づいて CD は送信データを複数の HCI データに分割し、これらをカプセル化して BGW へ転送する。
- (3) BGW は受信した HCI データをデカプセル化後、RD へデータパケットを送信する。
- (4) データパケットを受信した RD は正しく受信したことを示す ACK を BGW へ応答する。
- (5) ACK を受信した BGW の Controller は、自身が受信可能なバッファサイズを Host へ通知するために、HCI Number of Complete イベントを渡す。BGW はこの HCI イベントをフックして、カプセル化してから CD へ転送する。以後、CD は通知された BGW のバッファサイズにあわせて、データ送信を行う。

次に、RD から CD へのデータ通信シーケンスを図 7 に示す。

- (1) CD が RD からデータを受信する際、まず CD が受信可能なバッファサイズを BGW へ通知する。
- (2) RD は BGW に対してデータパケットを送信する。
- (3) RD からデータパケットを受信した BGW は、データパケットを正しく受信したことを示す ACK を RD へ応答する。
- (4) BGW の Controller はデータパケットを受信すると、その内容を含んだ HCI データを Host へ渡す。ここで、HCI データに含まれる Handle を用いてトンネルテーブルから IP アドレスを取得し、カプセル化した HCI データを CD に転送する。
- (5) HCI データを受信した CD は、BGW へ HCI Host

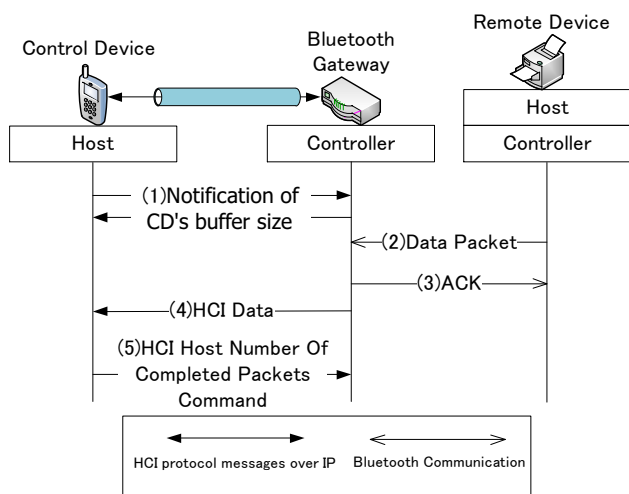


図 7 RD から CD へのデータ通信シーケンス

Fig. 7 Data communication sequence from RD to CD.

Number Of Completed Packets コマンドを転送し、CD が受信可能なバッファサイズを通知する。以後、BGW は RD から受信したデータを CD のバッファサイズにあわせて転送する。

CD の Host と BGW の Controller が協調して HCI プロトコルメッセージの交換を行うことにより、2 台の Bluetooth 機器をひとつの Bluetooth 機器であるかのようにふるまうことができる。これにより、CD は BGW の位置に存在していると見なす事ができ、RD が近傍にあるかのように通信できる。提案手法では BGW が HCI プロトコルメッセージに含まれる OpCode や Handle、Bluetooth 機器のデバイスアドレスを細かくチェックすることにより、メッセージの転送先の CD を一意に特定することができ、複数の CD がそれぞれ異なる RD へ同時に接続しても正しくメッセージを転送でき、BGW に実装する Bluetooth インタフェースはひとつでよい。

5. 実装に関する検討と評価

5.1 実装に関する検討

提案手法は Bluetooth スタック内で交換される HCI プロトコルメッセージをフックする必要がある。そこで、カーネルソースが公開されており、Google 社が開発したスマートフォン向け OS である Android が採用している Linux への実装を検討した。CD のモジュール構成を図 8 に、BGW

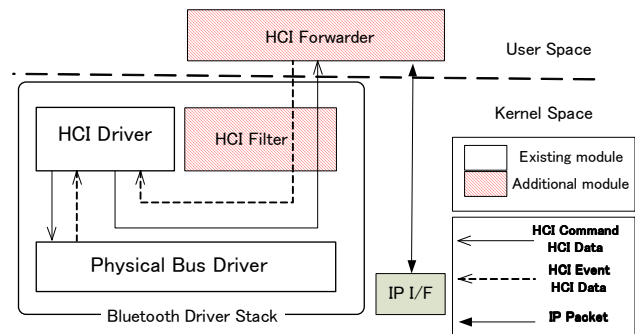


図 8 CD のモジュール構成

Fig. 8 Module configuration of CD.

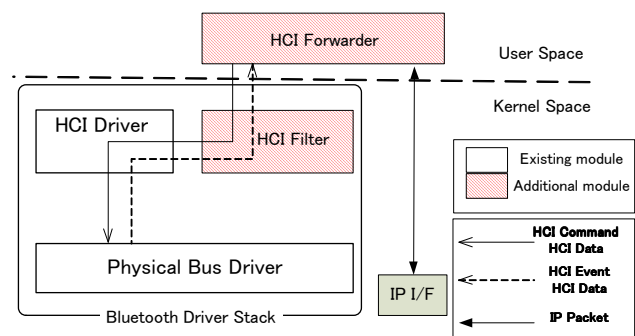


図 9 BGW のモジュール構成

Fig. 9 Module configuration of BGW.

のモジュール構成を図 9 に示す。Linux カーネル空間にある Bluetooth Driver Stack 内に HCI Filter モジュールを、ユーザ空間に HCI Forwarder モジュールを追加する。HCI Filter モジュールは Host と Controller の間で交換される HCI プロトコルメッセージをフックする機能を持つ。

HCI Filter モジュールがフックした HCI プロトコルメッセージは全て HCI Forwarder モジュールに渡される。CD の HCI Filter モジュールでは HCI Driver から Physical Bus Driver へ送信される HCI プロトコルメッセージをフックする。CD は Host の役割を果たすため、HCI Forwarder モジュールから渡されたメッセージは HCI Driver に渡す。BGW の HCI Filter モジュールは Physical Bus Driver から HCI Driver へ送信される HCI プロトコルメッセージをフックする。BGW は Controller の役割を果たすため、HCI Forwarder モジュールから渡されたメッセージは Physical Bus Driver に渡す。

HCI Forwarder モジュールはトンネル構築処理および CD と BGW の間で送受信する IP パケットの処理を行う機能を持つ。そのため、HCI Forwarder モジュールがトンネルテーブルを保持する。HCI Forwarder モジュールは HCI Filter モジュールから受信した HCI プロトコルメッセージをカプセル化して BGW へ送信する。また、BGW から転送されてきた HCI プロトコルメッセージをデカプセル化して、HCI Filter モジュールへ送信する。

Linux に実装されている Bluetooth Driver Stack (BlueZ) [7] に上記モジュールを追加することにより、CD および BGW におけるフック機能やトンネル構築・転送機能を実現する。改造を加えた Bluetooth Driver Stack を通常のカーネルモジュールと同じように Linux カーネルへ組み込むことにより、実機への実装が可能である。なお、Android に搭載されている Bluetooth Stack は Linux と同じスタックが採用されているため、スマートフォンへの実装も可能であると考えられる [8]。

5.2 定性評価

表 2 に遠隔地に存在する近距離無線機器へ接続する手法の比較を示す。提案手法では、宅内に設置する BGW を CD の Bluetooth インタフェースとして扱うことにより、宅外からでも宅内と同じ操作により RD へ接続できる。この時、ユーザは CD や RD の位置を意識することなく、常に一般の Bluetooth アプリケーションを用いて通信を開始できる。さらに、CD へゲートウェイに相当する機能を実装するため、ユーザは CD 以外に携帯する機器を持つ必要がない。CD は主としてスマートフォンを想定しているため、ユーザは接続場所を制限されず、いつでもどこからでも宅内の Bluetooth 機器へ接続できる。また、CD へ独自の機能を追加することにより、CD は BGW を介して RD と Bluetooth 通信を行う。そのため、インターネット

表 2 既存技術との比較

Table 2 Comparison with existing technologies.

	有線拡張	PUCG	提案手法
アプリケーションの使い分け	なし	あり	なし
専用アプリケーション	不要	必要	不要
ゲートウェイの携帯	必要	不要	不要
特別なサーバの設置	不要	必要	不要
Bluetooth インタフェース数	複数	1	1
外出先の端末への機能実装	不要	必要	必要

上に特別なサーバを設置して通信を中継する必要がない。複数の CD がそれぞれ異なる RD と接続する場合、BGW が CD を識別して RD からの応答を転送するため、BGW の Bluetooth インタフェースはひとつでよい。しかし、提案手法では外出先の操作端末への機能実装が必要である。Linux に含まれる Bluetooth Driver Stack に改造を加えることにより、フックなどの機能を実現することができる。

6. まとめ

本稿では、Bluetooth の HCI プロトコルメッセージをフックし、インターネットを経由して転送することにより、宅内の Bluetooth 機器に接続する手法を提案した。提案手法によると、ユーザは自身の位置や接続したい Bluetooth 機器の場所を意識することなく、常に一般の Bluetooth アプリケーションを用いて Bluetooth 機器へ接続できる。

今後は、トンネル構築に関わるセキュリティの検討および Linux カーネルへの実装を行う。また、提案手法を用いた場合の Bluetooth 通信のスループットおよび遅延に関する評価を行う予定である。

参考文献

- [1] 井波政朗, 丹 康雄: Bluetooth ネットワークの有線拡張方式に関する検討, 信学技報 CS2003-117, Vol. 103, No. 415, pp. 47-52 (2003).
- [2] 田中 剛, 伊藤崇洋, 加藤悠一郎, 峰野博史, 水野忠則: Android 端末を用いた異種ネットワークデバイス連携システムの開発, DICOMO2011 論文集, pp. 1257-1264 (2011).
- [3] Bluetooth SIG: *Adopted Bluetooth Core Specifications Ver4.0*, <http://www.bluetooth.org/Technical/Specifications/adopted.htm>
- [4] P2P Universal Computing Consortium: PUCG. <http://www.pucc.jp/>
- [5] Sumino, H., Ishikawa, N., Murakami, S., Kato, T. and Hjelm, J.: PUCG Architecture, Protocols and Applications, *Proc. of IEEE Consumer Communications and Networking Conference (CCNC) 2007*, pp. 788-792 (2007).
- [6] 小佐野智之, 石川憲洋, 北川和裕, 長坂文夫: 携帯電話向けプリンティングプロトコルの設計と実装, 情報処理学会研究報告, Vol. 2007-UBI-13, No. 10, pp. 69-74 (2007).
- [7] BlueZ Project: BlueZ. <http://www.bluez.org/>
- [8] Android Open Source Project: Bluetooth. <http://www.kandroid.org/online-pdk/guide/bluetooth.html>