

# Push-pin: ピン型タグを用いたホームオートメーションのためのプログラミングシステム

福地 健太郎<sup>†1</sup> 杉本 麻樹<sup>†3</sup> Charith Fernando<sup>†3</sup>  
Shengdong Zhao<sup>†2</sup> 稲見 昌彦<sup>†3,†1</sup> 五十嵐 健夫<sup>†1,†4</sup>

Push-pin はホームオートメーションのためのエンドユーザ向けプログラミングインタフェースで、ピン型の物理タグを用いて住人が手軽にプログラムを変更することができるものである。Push-pin のプログラミングモデルは stimulus-response モデルにもとづいており、ピンを使って接続された二つの機器の片方が操作されたり変化を感知すると、もう片方の機器にネットワーク越しに信号を送信し、機器を作動させる。例えば焦電センサが人の動きを感知すると照明を点灯させる、といった設定が可能となる。本報告では、push-pin システムの設計について説明した後、想定される応用シナリオを提案する。またプロトタイプの実装について説明し、GUI との比較評価実験の結果について報告する。

## Push-pin: a Tangible Programming Interface Using Physical Tags for Home Automation

KENTARO FUKUCHI,<sup>†1</sup> MAKI SUGIMOTO,<sup>†3</sup> CHARITH FERNANDO,<sup>†3</sup>  
SHENGDONG ZHAO,<sup>†2</sup> MASAHIKO INAMI<sup>†3,†1</sup> and TAKEO IGARASHI<sup>†1,†4</sup>

A tangible programming interface called “Push-pin” is proposed, which involves end-users in designing smart home programs for home appliance automation. Programming on the Push-pin system is based on a stimulus-response model in which an appliance connected to another appliance via a network gets activated when the other appliance is activated. To interconnect two appliances, the user puts a pin associated with an output appliance such as a lamp or a robot cleaner into an input module such as a switch or a motion sensor. The input appliance sends stimulus data with the ID of the pin as the destination address.

### 1. はじめに

住宅自身が住人の日常的な行動を観察し、それに基づいて住人の生活行動を支援する、いわゆる「スマートホーム」「ホームオートメーション」の研究は、ubiquitous / pervasive computing の発展を背景に様々な進展を見せている。典型的なスマートホームの考え方では、住宅に多数のセンサ機器や自動化のための機器が埋め込まれ、また他の家電製品などとネットワーク越しに接続され、中央サーバがそれらの連携を制御す

る<sup>1)</sup>。

スマートホームの「スマート」さ、つまり住人への便利なサービスを自動的に提供するための手法としては、大きく分けて三種類の手法がある。一つは、専門家が持つ知識や経験則をアルゴリズム化し、自動プログラムとして提供するもので、例えば家屋の設計時に間取りや住人の要望などを元に専門家がアルゴリズムやパラメタを決定し、家屋に組み込むといった工程を経る。しかし、起きうる事象や要望を設計段階で予想するのは困難であり、また住人の構成が変化したり新しい機器を購入したりするなど、住環境に変動があった場合に再び専門家に依頼し、システムを変更する必要が生じる。<sup>2)</sup>。別の手法は知能的システムを用い、住人の行動を認識・学習しまた住環境の変動に適応し、住人の要望を予測して適切なサービスを提供するものである(3)など)。しかしこの種のシステムでは、誤認識や予測手法の不備などの問題で適切なサービスが得られない場合に、利用者に強い不快感を与えるこ

<sup>†1</sup> 科学技術振興機構 ERATO 五十嵐デザインインタフェースプロジェクト

Japan Science and Technology Agency, ERATO IGARASHI Design Interface Project

<sup>†2</sup> National University of Singapore

<sup>†3</sup> 慶應義塾大学  
Keio University

<sup>†4</sup> 東京大学  
The University of Tokyo

とがよく知られている (Microsoft Office のイルカなど)。Mozer は、ニューラルネットワークを応用したスマートホームの実験を通じた経験を次のように語っている (4) より引用)。“The result is likely to be that the system mispredicts often and annoys the inhabitants more than it supports them.”。三番目は、住人自身がプログラムやパラメタの調整をするというもので、住人は家庭のパソコンや携帯電話などを利用し、システムにアクセスして設定を変更する。前二つのシステムと異なり、住人自身がシステム設計に参加するためより適切なサービス設計が可能になる。またシステムの仕組みを住人自身が知っているため、予測誤りから来る不快感の軽減が期待できる。一方で、パソコンなどを使う場合、設定を変えるためにその都度パソコン上のソフトを起動する手間がかかり、またパソコンおよび設定ソフトウェアの知識が必要となる。また、サービスを提供する機器とは離れた場所に置かれた画面で専用のソフトウェアを使うこととなるため、移動やソフトウェア起動の手間がかかり、加えて対象となる機器から離れた場所で設定を変更するのは利用者の認知負荷を増やす問題がある。

今回我々は、対象となる機器に直接差し込んで使うピン型タグを用いて、住人自身がシステムの設定を変更できるインタフェースを構築した (図 1)。提案インタフェースでは対象となる機器のうち、照明やエアコンのように住人に直接サービスを提供する機器を出力機器、スイッチやセンサなどそれらを操作するのに用いる機器を入力機器として捉える。出力機器の機能を表わすピンを、入力機器のピンスロットに差し込むとその二つはネットワーク越しに接続され、入力機器が出力機器を制御できるようになる。ピンは対応する出力機器の形状を模したものであり、またピンの挿抜は入力機器のそばで行われるため、接続の把握がしやすく、またコンピュータを使い慣れない住人でも簡単に操作ができる。例えば壁のスイッチに天井照明を関連づけるような場合であれば、対象となるスイッチそのものに働きかけるため、設定操作は直感的であり、ピンが目印となるため新しい設定も覚えやすい。また、現在の設定に不満を感じた場合でも、その場ですぐに設定を変更できる。

本報告では、関連研究について述べた後、Push-pin システムの設計について説明した後、想定される応用シナリオを提案する。またプロトタイプの実装について説明し、GUI との比較評価実験の結果について報告する。



図 1 Push-pin プロトタイプ実装。フロアランプに対応するピンをスイッチに差し込むことにより、このスイッチでフロアランプの ON/OFF ができるようになる。

## 2. 関連研究

スマートホームの研究では様々な人工知能技術を活用し、住人の要望を推測したり最適なサービスを判断し、機器の自動制御をする手法が提案されている。推測手段として、ヒューリスティクス<sup>5)</sup>・ニューラルネット<sup>3)</sup>・統計的手法<sup>6)</sup>などの利用がこれまでに提案されている。多くの研究では、センサにより住人の行動や環境の状態を認識し、家屋内の住人の行動をセンサにより認識し、そこから住人の行動パターンを抽出して、以降そのパターンに類似した行動が観測された場合にパターン中で操作されていた機器を自動制御する。また専門家の知識をシステム化し、住環境を快適に保ったり消費エネルギーを削減するといった目的にも使用されている。こうした手法の場合、センサやパターン抽出および合致判定での精度の問題のため、誤認識により要望にそぐわないサービス提供をしてしまうという問題が生じる。

Intille らは、上述したようなサービスの自動化において、システムが機器を自動的に制御するかわりに、住人に対象となる機器の操作を促すという手法を提案している<sup>7)</sup>。例えば室温を下げるために部屋の窓を開けた方がよいとシステムが判断すると、窓を自動で開くのではなく、窓枠に埋め込まれた LED を点灯し、住人に窓を開けた方がよいと示唆する。この手法は住人の予測しない、あるいは望まない動作を避けることができる。一方で、例えば留守の間に自動的に用事を片付けさせることはできず、自動化のメリットが低い。

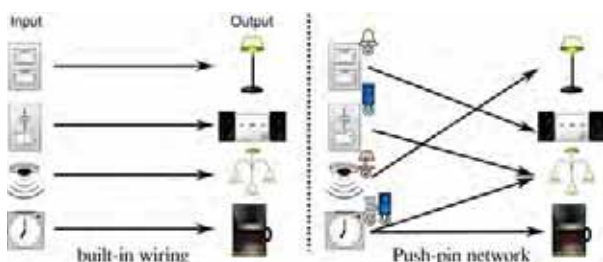


図 2 Stimulus-response モデルによる通信の模式図。入力機器と出力機器との関係は従来の建築でも直接配線という形で見られるが、その関係は固定されている。Push-pin システムではこの関係を住人が自由に変更することができる。

プログラムの構築や設定変更を簡単にし、住人自身の参与を目指したものとしては、コンピュータ画面を要するものでは X10 や INSTEON などの製品を対象とした市販ソフトウェアに加え、邸内位置情報計測技術と PDA を使ったものがある<sup>8)</sup>。実世界指向を取り入れたシステムの提案としては、バーコードを使ったもの<sup>9)</sup>、ボタンを順番に押すもの<sup>10)11)</sup>、手で触れるジェスチャを使うもの<sup>12)</sup>、RFID カードで触れるもの<sup>13)</sup>がある。いずれの手法も設定内容はシステム内に論理的に保存されるため、住人がそれを一目で把握することはできない。

物理タグによる接続という観点では、綾塚らによる tranSticks<sup>14)</sup> が先行研究として挙げられ、我々のシステムはこれに触発されたものである。tranSticks では音声・映像情報など様々なデジタルストリームやファイルの転送元/先の設定に用いられる。我々が提案するシステムは、邸内機器という物理的な動作を伴うホームオートメーションサービスを対象としている点が異なる。

### 3. Stimulus-response モデル

Push-pin システムではプログラミングモデルとして、stimulus-response モデル<sup>15)</sup>を採用している。このモデルではすべての設定は刺激 (stimulus) と反応 (response) の組で表わされる。入力機器が操作されたり反応があった場合 (例: センサが動きを感知した) にそこから stimulus が発信され、出力機器は受信した stimulus に反応して、その機能を作動させる (例: 明かりをつける)。提案システムではこの関係を、出力機器に対応するピン型のタグを入力機器に挿入することで記述できる (図 2)。

Stimulus-response モデルは単純であり理解がしやすい。また、ホームオートメーションの様々なアイデアがこのモデルで記述することができる。例えば文献 16) に作例として上げられている「電話で喋っている



図 3 ピンのプロトタイプ実装。左より、フロアランプ (2 種)・扇風機・テレビ・オーディオプレイヤーに対応したものを。

ときにテレビの音量を下げる」「雨が降っていない日は芝生に散水する」「毎週月曜日に、ゴミ出しのリマインダを示す」などは、いずれもカレンダーやマイク、天気情報などの入力機器があれば実現できる。

### 4. Push-pin システム概説

Push-pin システムはモジュール・ピン・ネットワークから構成される。モジュールは制御の対象となる家電機器や、スイッチやセンサなど住人が操作したり環境の様子を認識するための機器を指す。すべてのモジュールはネットワークに接続され、モジュール間の stimulus-response 結合がピンにより設定される。

#### 4.1 モジュール

モジュールは大きく分けて入力モジュールと出力モジュールに分類される。入力モジュールは、入力のための部品 (例: ボタン・温度センサ) を一つ以上持ち、またそれぞれの入力部品毎に対応する一つ以上のピンスロットを持つ。出力モジュールは実際に仕事をする機器に、ネットワークアダプタと制御部を組み込んだものであり、またそれに対応するピンを持つ。出力モジュールが持つ機能にはそれぞれ ID が割り当てられており、またその機能に対応するピンにも同じ ID が埋め込まれている。このピンを入力モジュールのピンスロットに差し込むと、それらが仮想的に接続される。

ボタンが押されたりセンサに反応があるなどして入力モジュールが作動すると、モジュールはピンスロットに差し込まれているピンの ID を読み取り、どのような入力があったかを示すデータとその ID を合わせて、ネットワークに送信する。出力モジュールはネットワークから信号を受信し、信号に含まれる ID が自身が持つ機能 ID と合致した場合に、受け取った付加データに対応した動作を行う。

一部の機器は、入力と出力の両方のモジュールとして機能することもできる。例えばコーヒーメーカーは他の入力モジュールから信号を受け取ってコーヒーの抽出を開始する出力モジュールであると同時に、コーヒーの抽出が終わった事を他の出力モジュールに伝えるためのピンスロットを設けることで、入力モジュール

ルとしても動作しうる。

モジュールには入力・出力モジュールの他に、中間モジュールが使われる。論理演算や条件分岐、信号加工などの機能を持ち、入力モジュールと出力モジュールの間に挟んで高度な自動処理のために用いられる。

#### 4.2 ピン

ピンはすべての出力モジュールに付属するもので、その外観は、対応している機能を連想させるものとなっている。ピンの大きさは一般的に用いられるプッシュピン程度を想定しており、指先でつまんで挿抜がしやすい程度には大きく、一方で日常生活を邪魔しない程度には小さいことを目指している。またピンの外観は、ピンの差し替えによる設定の変更時だけでなく、普段の生活で、スイッチやセンサがどの機器に関連づけられているかを一目で判断できることが求められる。そのため、ある程度視認性を確保できる大きさが必要である。図 3 にプロトタイプとして我々が製作したピンの写真を載せる。

ピンには ID のみ埋め込まれており、パッシブタグとして機能する。入力モジュールはピンスロットに差し込まれたピンの ID を読み取る。複数のピンが一つの入力部品に差されている場合は、同じ操作データ信号がそれぞれの ID に一斉に送信される。

#### 4.3 ネットワークと通信プロトコル

モジュール同士は有線もしくは無線ネットワーク、あるいはその混成ネットワーク (INSTEON<sup>10</sup>) を参照) を介して通信する。有線ネットワークは電力線通信を利用するのが、電力供給や配線コストの面から好ましい。電力線通信による家電間通信としては、X10 や INSTEON などの先行商品があり、有効性が確認されている。一方無線ネットワークの場合は、機器の配置の自由度が高く、赤外線リモコンのような機器も対象とすることができる。

モジュール間でやりとりされるメッセージのプロトコルについて説明する。メッセージはネットワーク上を流れるパケットで実装される。一つのパケットには、ピン ID とペイロードデータが含まれる。ペイロードには一つ以上のイベント情報が記される。イベント情報は型 (種類) 情報と値からなる。現在の実装では、ON/OFF の指示のみ含むバイナリ型と、0 から 255 の数値を表わす整数型が定められている。バイナリ型は機器の ON/OFF 制御に用いられ、整数型は音量や明るさの制御に用いられる。

### 5. シナリオ例

以下にいくつかのシナリオ例を挙げ、push-pin シス

テムを用いて一般家庭でどのようなホームオートメーションの設定が記述できるかを示す。シナリオ中に、どのような入出力モジュールが考えられるかもあわせて示している。なお、入出力モジュールやそれを使ったシナリオの一部は、すでにプロトタイプとして実装されている (6 節を参照)。

#### 5.1 真っ暗な玄関と掃除機の騒音よ、サヨウナラ

ナオミは最近帰宅が遅く、玄関を開けたら暗い土間でスイッチを探して壁を探るのが日課になっている。玄関を開けたら自動で玄関の照明がつくようにしたかったので、ナオミは焦電センサモジュールを買ってきて、玄関の天井に取り付け、玄関の照明用のピンを壁のスイッチからセンサモジュールに移した。これで、玄関を開けると焦電センサが反応して自動的に照明をつけてくれるようになった。

この自動化がうまくいったことにナオミは気を良くし、もう一つ家に帰った時に気になっていた事を解決しようとする。たまにいつもより早く帰ると、居間で掃除ロボットがまだ動いていることが多く、それが騒がしいのが気に障っていたのだ。そこで、掃除ロボットを買ったときについてきた「ロボットをベースに戻す」ピンを、玄関に設置した焦電センサモジュールに差し込んだ。

このシナリオでは、入力モジュールは後から追加設置できることを前提とし、今までスイッチで操作していた玄関照明を自動化する方法、また同じ「家に帰ったら」という条件に反応する仕事を増やすのに、入力モジュールに他のピンを差すことで実現できることを示した。

#### 5.2 目覚めのコーヒーと音楽

時計モジュールは定期的に機器を動作させたい場合に役立つ。多くの家電機器はそれぞれタイマーを内蔵し自動起動に対応しているが、タイマーはそれぞれ別個に設定しないとまらないため、全部の機器の設定を一度に変えたい場合でも同じ作業を機器の回数分繰り返さねばならない。このシナリオでは、複数の機器を同期させて動かすための方法を示す。

太郎は朝起きたらすぐにコーヒーを飲むのが日課だ。時計モジュールにはコーヒーマーカの抽出開始ピンが差し込み、抽出終了スロットにはオーディオプレーヤーが、フェーダーモジュールを介して接続されている。朝 7 時になると時計モジュールはまずコーヒーマーカを起動する。コーヒーが出来上がると信号がフェーダーモジュールに届く。フェーダーモジュールは中間モジュールで、信号を受け取ると、出力モジュールに 0% から 100% までの値を、設定された速度で送る。



図 4 試作した入力モジュール。左からスイッチモジュール・ボリュームモジュール・焦電センサモジュール。



図 5 試作した出力モジュール。既存のランプや扇風機を、リレーにより ON/OFF する。

オーディオプレーヤはその信号に応じて、音量を最小から最大にしながら音楽を再生する。お陰で太郎が音楽に起こされた時には、もうコーヒーが用意できている。朝 6 時に時刻を買いたい場合は、時計モジュールのみ設定を変えれば済む。

## 6. プロトタイプ実装

これまでに説明したコンセプトを実証するためのプロトタイプを我々は実装した。

ピンとピンスロットには、音楽用途でよく使われる 2.5 mm のモノジャックコネクタを使用した。ピンの ID は、異なる抵抗値の抵抗をピンに接続することで実装した (図 3 参照)。現在は 5 種類の ID が用意されている。モジュール制御には PIC コントローラを用い、通信には、ZigBee を使った無線通信を採用した。電源には家庭用 AC 電源または乾電池を使用している。

これまでに入力モジュールとしてスイッチ・ボリューム・焦電センサ・時計モジュールを、また出力モジュールとして汎用の電源モジュールを製作した。図 4 および図 5 に試作モジュールの写真を示す。焦電センサモ



図 6 実験に用いた部屋および機器。被験者は照明などの機器をスイッチモジュールのスイッチに割り当ててを求められる。GUI を使う場合は、被験者は PC を操作して設定を変更する。

ジュールは、人を感知すると ON メッセージを送信し、その後人を感知しなくなってから 10 秒経過すると OFF メッセージを送信する。汎用電源モジュールは、ON/OFF メッセージを受け取ると内部のリレーを操作し、モジュール上のコンセントに接続された機器への電源供給をメッセージに応じて制御する。電源モジュールの ID は実験で様々な機器を接続することを想定し、接続した機器に応じて変更できるようになっている。

## 7. 評価実験

前節で紹介したプロトタイプ実装を用い、ピンの差し替えによる設定変更のインタフェースの有効性を評価した。この調査では居間を模した部屋の中で、スイッチモジュールと複数の家電機器との接続設定を被験者にしてもらった。提案システムに加え、比較用に GUI で設定を変更できるシステムを用い、設定変更にかかる速さを測定し、また質問紙により主観的评价を行った。

### 7.1 実験デザイン

3 m × 5 m 程度の、仕切られた実験スペースにはソファ・コーヒーテーブルおよびコンピュータが置かれた (図 6)。比較対象となる GUI システムはこのコンピュータ上で動作する。操作対象となった出力モジュールは二つの同型のフロアランプ・テーブルランプ・扇風機・LED 照明の五つで、また入力モジュールとして三つのスイッチモジュールを用いた。一つのスイッチモジュールには 2 個のスイッチがあり、合計 6 個のスイッチが対象となった。一つのスイッチには 2 個のピンスロットが設けられている。三つのスイッチモジュールは部屋の入口とコーヒーテーブルの上、



図 7 評価実験で用いたピン

およびテレビセットの横に設置された。実験で用いたピンは図 7 に示したような、貼られたラベルで識別するピンである。二つのフロアランプに対応するピンは同じラベルが貼られており、セッションの開始時に被験者はそれらを区別するためにラベルへの書き込みを求められる。

被験者は、6 個のスイッチに五つの機器を割り当てることを求められる。割り当ては紙に書かれて渡された。なお、この中で二つのフロアランプはその設置場所により区別される。被験者は一回のセッションで四つの課題にあたった。セッション開始時は機器はどのスイッチにも割り当てられていない。課題 1 は五つの機器をまず指定されたスイッチに割り当てる。課題 2 と 3 では一部の機器を、他のスイッチに割り当て直す。課題 4 では、一度すべての割り当てを解除した後、被験者自身に好きなように割り当ててもらった。すべての課題で被験者は、割り当て作業が終わった後に実際にスイッチを操作して、正しく割り当てられたことを確認してもらった。課題にかかった時間は、割り当てを記した紙を渡してからこの確認作業が終了するまでとした。セッションが終了すると被験者に、質問紙に記入してもらった。各質問は 7 段階のリッカートスケールで回答させるもので、質問は表 1 に示した。Push-pin セッションと GUI セッションの両方を各被験者に行ってもらった後、インタビューを実施した。

## 7.2 GUI ツール

評価実験では比較対象として、GUI により割り当てを変更できるシステムを用いた (図 8)。我々は X10 や INSTEON 用に市販されているいくつかの GUI ツールを参考に、シンプルなエディタを独自に製作した。PC は ZigBee ネットワークを通じてモジュールと通信し、実際のスイッチの操作で割り当てられた機器を操作できる。

画面上には部屋の間取りが背景として表示され、ソ

表 1 Questions

Q1	配線のやり方はすぐに理解することができた
Q2	配線が実際の機器でも正しく動作することに確信が持てた
Q3	実際の動作確認の手間が面倒だった
Q4	どの が実際の部屋に置かれたどの機器と対応しているか、簡単に理解できた
Q5	配線を変えるために するのは、面倒だ
Q6	配線作業中、どう配線すればよいのかとまどろことがあった
Q7	最後の実験で、頭の中で思い描いた配線を、すぐに実現に移すことができた
Q8	を使った配線システムを自宅でも使ってみたいと思う

には、セッションで用いたシステムに応じた言葉が入る

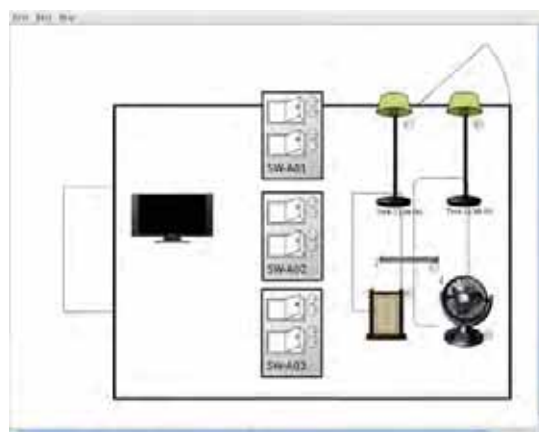


図 8 GUI による配線エディタの画面。

ファヤコーヒーテーブルが図示されている。各モジュールはアイコンとしてあらかじめ表示されており、これらはドラッグ操作により動かすことができる。入力モジュールと出力モジュールとを接続するには、まずどちらかのモジュールのアイコンに設けられた矢印マーカーをまずクリックする。するとそこから直線がマウスポインタに向かって伸びるようになる。その状態で相手モジュール上の矢印マーカーをクリックすると二つのマーカーが直線で結ばれ、二つのモジュールは接続される。配線を変更するには、どちらかの端のマーカーをクリックすると、再び他のマーカーを選択できる状態になる。その状態で背景をクリックすると、配線を消すことができる。

GUI ツールの初期画面を図 8 に示す。GUI セッションでは課題 1 の前にまずモジュールアイコンを被験者に操作してもらい、部屋の実情にあうように配置しなおしてもらおう。なお、画面内の部屋の向きは、被験者から見ると実際の部屋とは反対の向きになっている。

## 7.3 被験者

13 人の被験者 (男 8:女 5) が実験に参加した。実験前に質問紙により被験者の背景調査をした。被験者の年齢は 20 ~ 50 歳代で、8 名は毎日 1 時間超コンピュータを使い、2 名は毎日 1 時間以下、3 名は週

に2~3回程度使用すると回答した。我々は被験者を、毎日1時間超使うグループをコンピュータ熟練者、残りを初心者として二つのグループに分け、それぞれのグループで半数(初心者グループでは3名、熟練者グループでは4名)がGUIセッションを先に、残りがpush-pinセッションを最初に行い、続いて残りのセッションを行った。

#### 7.4 結果

図9に、各課題の達成時間を示した。GUIセッションの達成時間には、コンピュータやGUIツールを起動するのに要した時間は計上されていない。また、課題1にかかった時間には、ツール起動時に初期位置からアイコンを再配置する時間は含まれない。なお、再配置には平均78秒(標準偏差5秒)を要していた。

結果より、課題1および3では、push-pinシステムはGUIツールに比べて有意に速い( $p = 0.083$  and  $p = 0.033$ , Welchのt検定による)。課題2および4でも平均値ではpush-pinの方が速いが、有意差は見られなかった。詳しく調べてみた結果、初心者グループと熟練者グループとで結果に違いが見られた(図10)。課題1と3においては、初心者グループにおいて有意な差が見られた( $p < 0.05$ )が、熟練者グループでは大きな差が見られないことが判明した。

図11に、主観的評価の結果を示す。図では右側(7点)がシステムに肯定的な評価となっている。Mann-Whitney-Wilcoxon検定を適用した結果、Q2(正しく動作することに確信が持てた)に対する回答ではpush-pinが有意傾向( $p = 0.08$ )を示したが、Q5(接続は面倒だった)への回答では有意に( $p = 0.03$ )、またQ8(自宅でも使ってみよう)では有意な傾向として、push-pinに対して否定的な評価を得た。

Push-pinセッションを先に行ったグループとGUIセッションを先に行ったグループとの比較では、いずれの項目でも有意差は見られなかった。

#### 8. 議論

速度比較の結果より、push-pinシステムはGUIに比べ、パソコンを使い慣れない人では有意に優れていることが示された。さらにこの結果には、GUIツールの起動時間は含まれていないため、実際の場面ではさらに時間がかかることとなる。提案システムはパソコンを生活において、ちょっとした不満を解消するための割り当変更のような場面ではGUIツールによる設定は手間となる。

しかし主観的評価の結果ではQ5(接続作業は面倒だった)でpush-pinに否定的な評価がなされている。

インタビューによると、「設定変更の度にスイッチのところまで歩いていくのが面倒」「ピンを差し替える時に、目的のピンがどのスイッチに差さっているかわからなくなった」といった回答が被験者より得られた。このうち前者については、今回の実験では割り当て指示は逐一ソファに戻って渡されていたため、移動の手間が生じていた。しかし、例えばスイッチを操作している時に割り当てに不満を持った場合などは、その場で設定をすぐに変更できるという利点もあり、場面によってはこの手間は生じない。これについては、長期運用に基いた議論が必要である。

Q2(正しく動作することに確信が持てた)への回答では提案システムは肯定的に評価された。多くの被験者はピンを差すと同時にスイッチを操作して動作確認をしており、こうした点が評価されたものと我々は推測している。またGUIセッションでは最初のアイコンの再配置で手間どる被験者も多かった。画面上のアイコンと実際の機器との対応を把握するのに困難を感じた被験者が多かつと推察される。しかし一方で、GUIツールは部屋全体を画面上で把握できるのがよいとインタビューで回答した被験者もいた。対象となる空間が家全体であればよりその傾向が強まる可能性もある。空間把握の問題については今後、対象を家全体に広げて実験を試みたい。

#### 9. まとめと今後の課題

ホームオートメーションの設定を住人自身が手軽に変更できるインタフェース「push-pin」を提案した。Stimulus-responseモデルに基き、入力と出力の関係をピン状のタグを用いて設定できるもので、ピン自体が対応する機器の形状を模したものとなっており把握が用意であり、また相手側の機器に直接ピンを差して使うので直感的であるという利点がある。本報告ではコンセプトの提案とともにプロトタイプ実装を示した。また、既存システムでは一般的なGUIによる設定との比較評価実験を実施し、早さや設定の確信度の主観評価において有利であることを示したが、主観評価の他の項目では否定的な評価も得られた。

今後の課題としては、対象を家全体とした上で長期運用実験を実施した上での議論が必要なることを挙げる。また、様々なホームオートメーションが提案システムを用いて実際に構築できるか、さらなる検証が課題である。その際には、例えばコーヒーの粉が切れている・センサが壊れたなどの例外に対する処理の記述についてもサポートする方法を考えたい。

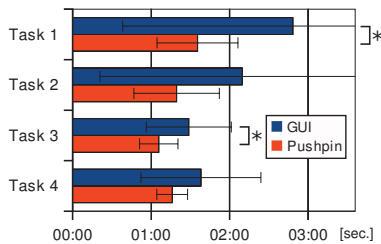


図9 課題1-4の結果(速度)

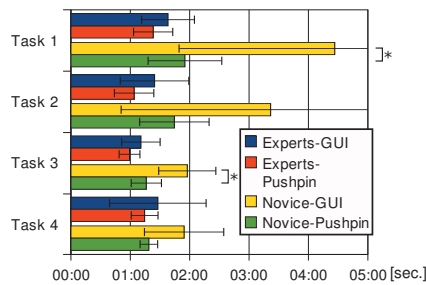


図10 初心者グループと熟練者グループ間での速度比較

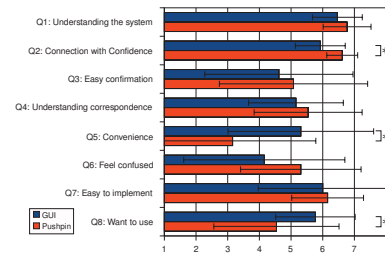


図11 主観的評価の結果。右側が肯定的回答。

## 参考文献

- Chan, M., Estève, D., Escriba, C. and Campo, E.: A review of smart homes-Present state and future challenges, *Computer Methods and Programs in Biomedicine*, Vol.91, No.1, pp.55-81 (2008).
- O'Brien, J., Rodden, T., Rouncefield, M. and Hughes, J.: At home with the technology: an ethnographic study of a set-top-box trial, *ACM Trans. Comput.-Hum. Interact.*, Vol. 6, No. 3, pp.282-308 (1999).
- Mozer, M.C.: The Neural Network House: An Environment that Adapts to its Inhabitants, *The American Association for Artificial Intelligence Spring Symposium on Intelligent Environments*, pp.110-114 (1998).
- Norman, D.A.: *The Design of Future Things*, Basic Books (2007).
- Elger, G. and Furugren, B.: SmartBO - an ICT and computer based demonstration home for disabled people, *Improving the Quality of Life for the European Citizen: Technology for Inclusive Design and Equality*, Assistive Technology Research Series, Vol.4, IOS Press, pp. 392-395 (1998).
- Matsuoka, K.: Aware home understanding life activities, *Toward a Human-friendly Assistive Environment: ICOST 2004: 2nd International Conference On Smart homes and health Informatics*, Assistive Technology Research Series, Vol.14, IOS Press, pp.186-193 (2004).
- Intille, S.S.: Designing a Home of the Future, *IEEE Pervasive Computing*, Vol.1, No.2, pp. 76-82 (2002).
- Sakamura, K.: Challenges in the age of ubiquitous computing: a case study of T-Engine, an open development platform for embedded systems, *ICSE '06: Proceedings of the 28th international conference on Software engineering*, ACM, pp.713-720 (2006).
- Siio, I., Masui, T. and Fukuchi, K.: Real-world interaction using the FieldMouse, *UIST'99: Proceedings of the 12th annual ACM symposium on User interface software and technology*, ACM, pp.113-119 (1999).
- Darbee, P.: *INSTEON: The Details*, Smarthome Technology (2005).
- Iwasaki, Y., Kawaguchi, N. and Inagaki, Y.: Touch-and-Connect: A connection request framework for ad-hoc networks and the pervasive computing environment, *First IEEE Annual Conference on Pervasive Computing and Communications (PerCom 2003)*, IEEE, pp. 20-29 (2003).
- Rekimoto, J., Ayatsuka, Y. and Kohno, M.: Synctap: An Interaction Technique for Mobile Networking, *Human-Computer Interaction with Mobile Devices and Services (Mobile HCI 2003)*, Springer Berlin / Heidelberg, pp.104-115 (2003).
- Kawsar, F., Nakajima, T. and Fujinami, K.: Deploy spontaneously: supporting end-users in building and enhancing a smart home, *UbiComp '08: Proceedings of the 10th international conference on Ubiquitous computing*, pp. 282-291 (2008).
- Ayatsuka, Y. and Rekimoto, J.: tranSticks: physically manipulatable virtual connections, *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, pp.251-260 (2005).
- Wolber, D. and Fisher, G.: A demonstrational technique for developing interfaces with dynamically created objects, *UIST '91: Proceedings of the 4th annual ACM symposium on User interface software and technology*, ACM, pp.221-230 (1991).
- Ledford, J.: *25 Home Automation Projects for the Evil Genius*, McGraw-Hill (2006).