解　説

# 日曜ユビキタスのための手軽なミドルウェア

塚田 浩二

本論文では，ユビキタス・インタフェースの開発を支援するさまざまなデバイス／ツールキットを紹介する．そして，複数のデバイス／ツールキットを組み合わせて，日曜大工や日曜プログラミングのような感覚で，手軽にユビキタス・インタフェースを構築できる「日曜ユビキタスのためのミドルウェア群：MobiServer」を提案する．さらに，MobiServer を用いたシステム開発の実例を示す．

In this paper, we introduce various devices and toolkits for developing ubiquitous interfaces effectively. Moreover, we propose novel middleware, the "MobiServer", which helps users treat multiple devices and toolkits easily. Finally, we show some research projects developed with the MobiServer.

## 1　はじめに

近年，HCI (Human Computer Interaction) の研究対象は GUI (Graphical User Interface) を中心としたディスプレイの中の世界から，日常生活空間全体へと大きく広がってきた．

従来のコンピュータの入出力インタフェースは，マウス／キーボード／ディスプレイにほとんど固定されていたが，日常生活空間での利用を想定した実世界指向インタフェース (以下，ユビキタス・インタフェース) の研究が盛んに行われている [15]．このようなユビキタス・インタフェースにおいては，ユーザの周辺状況や身体的な動作を的確に取得することが重要になる [8]．

そのためには，さまざまなセンサ／アクチュエータや小型コンピュータなどを組み合わせたシステムを実装する必要がある．しかし，このような研究／開発はハードウェア／ソフトウェアの双方の技術が必要となるため，敷居の高い領域だった．

こうした状況に対応するために，ここ数年の間に，コンピュータからさまざまなセンサ／アクチュエータ群を扱うためのデバイス／ツールキットが登場してきた．

本稿では，これらのデバイス／ツールキットについて，実際に国内で入手可能な製品に絞って詳しく解説する．次に，複数のデバイス／ツールキットを組み合わせて，日曜大工や日曜プログラミングのような感覚で，手軽にユビキタス・インタフェースを構築できる「日曜ユビキタスのためのミドルウェア群：MobiServer」について紹介する．さらに，MobiServer を用いた実世界指向インタフェースの実例を示す．

## 2　デバイス／ツールキット

ここでは，ユビキタス・インタフェースを実現するための基礎技術として,「センサ／アクチュエータ」「家電／照明機器連携」「無線センサシステム」といった観点から，実用的なデバイス／ツールキットを紹介する．表 1 に，その一覧と概要を示す．

---

Lightweight middleware for developing ubiquitous interfaces

Koji TSUKADA, お茶の水女子大学 お茶大アカデミックプロダクション, Academic Production, Ochanomizu University,

コンピュータソフトウェア, Vol.xx, No.xx (2009), pp.xx–xx.

[解説論文] 2009 年 5 月 31 日受付.

**表 1　デバイス／ツールキットの一覧**

| 分類 | 名前 | 特徴 | 入手先 (一例) |
|---|---|---|---|
| センサ／アクチュエータの制御 | Phidgets | USB 接続の汎用 I/O デバイス．センサ／アクチュエータ群とライブラリが充実． | phidgets.com |
| | Gainer | USB 接続の汎用 I/O デバイス．ポート構成を変更可． | switch-science.com |
| | FT245RL | USB 接続の汎用 I/O デバイス．安価． | akizukidenshi.com |
| 家電／照明制御 | X10 | 電灯線通信で家電機器を制御．設置が容易． | smarthome.com |
| | IR | USB 経由で赤外線信号を送受信．複雑な家電機器 (AV 機器など) を制御． | buffalo-direct.com |
| | DMX | USB 経由で DMX 規格の舞台照明／演出装置を制御． | soundhouse.co.jp |
| 無線センサ | Mote | 複数の無線デバイスからセンサ情報を収集．専用ソフトウェア付属． | xbow.jp |
| | XBee | 複数の無線デバイスからセンサ情報を収集．安価．無線シリアル通信可． | switch-science.com |

### 2.1　センサ／アクチュエータの制御

ユビキタス・インタフェースにおいては，ユーザやモノの状況／動作などを認識するために，さまざまなセンサ／アクチュエータを連携させて扱う必要がある．たとえば，物理的なモノの使い心地とコンピュータをリンクさせて活用するタンジブル・インタフェース [7] では，モノの状態／動きなどを検出するためのセンサが必要となる．また，周辺感覚で捉えられる光／風／動きなどを活用したアンビエントな情報提示手法 [7] では，LED ／モーターなどのアクチュエータが必要となる．

ここでは，パソコンなどからセンサ／アクチュエータを扱うためのデバイス／ツールキットとして，Phigets [6], Gainer [9], FT245RL モジュール [4] について詳しく紹介する．

#### 2.1.1　Phidgets

Phidgets (Physical Widgets) は，物理的なデバイス (スライダー，スイッチ，LED，モーターなど) を，パソコン上の GUI と同じような感覚で利用することを目標としたツールキットの総称であり，カルガリー大学の Saul Greenberg 氏らにより開発された [6]．Phidgets は，USB でパソコンに接続するデバイス群 (以下，Phidget デバイス) と，それらを制御するライブラリ群から構成される．

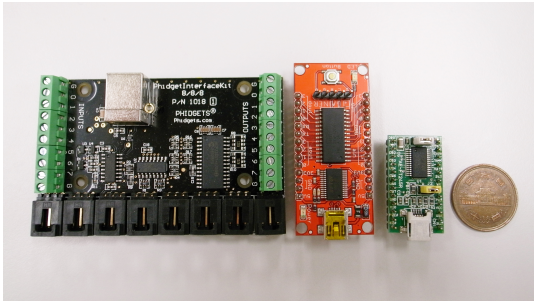

**図 1　センサ／アクチュエータを制御する汎用 USB I/O デバイス (左から，Phidget InterfaceKit, Gainer I/O, FT245RL モジュール)**

Phidget デバイスには，汎用的な I/O ボード (アナログ入力，デジタル入力／出力) に加えて，直接 USB に接続できるさまざまなセンサ／アクチュエータが用意されている．ほとんどの Phidget デバイスは，USB から直接給電できる．また，ライブラリも C#／ C++／ Java ／ Flash (ActionScript) などの一般的な言語から，Python ／ Delphi ／ LabVIEW ／ Max/MSP などのやや特殊な言語まで幅広く対応

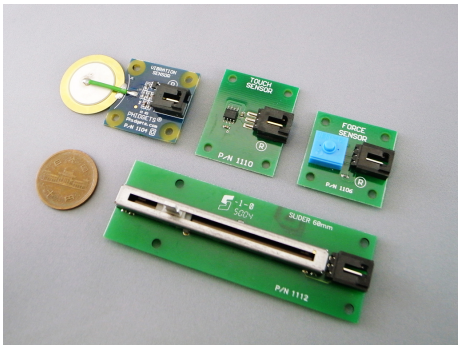


**図 2 Phidget InterfaceKit 用センサの一例．上段 左:振動センサ，中央:タッチセンサ，右:圧力センサ．下段: スライダー**

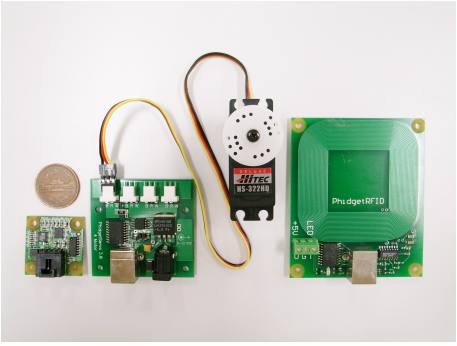


**図 3 USB 直結型の Phidget センサ．左:加速度センサ (Phidget Accelerometer)，中央: サーボコントローラ (Phidget ServoMotor)，右:RFID リーダー (Phidget RFID)**

している[†1].

以下，代表的な Phidget デバイスについて説明する．汎用 I/O ボード (Phidget InterfaceKit) は，各 8 ポートのアナログ入力／デジタル入力／デジタル出力を備えている (図 1 左)．アナログ入力は，10bit の A/D コンバーターを搭載しており，電圧を出力とするさまざまなセンサを接続できる．図 2 に Phidget InterfaceKit 用センサの一例を示す．振動センサは，接地面の振動や衝撃を検出する．タッチセンサは，静電容量の変化を用いて，人の指などが基盤面に触れたことを検出する．圧力センサは，センサ中央部の押し込み圧力を検出する．スライダー (可変抵抗) は，一般的なスライダーと同様に，つまみの水平位置を検出する．この他にも，公式 Web サイト[†2] では数十種類のセンサが販売されている他，市販センサを接続することもできる．デジタル入力は ON/OFF を検出する 1bit の入力端子であり，スライドスイッチ／プッシュスイッチなどを接続する．デジタル出力は，5V の電圧を ON/OFF することで，LED などを制御する．リレーなどを介することで，照明など高電圧機器の制御も可能である．

次に，直接 USB に接続するセンサ／アクチュエータを紹介する．加速度センサ (Phidget Accelerometer) は，USB に直結できる 2 軸／3 軸の加速度センサである (図 3 左)．たとえば，物体の移動にともなう加速度や，静止時の地軸に対する傾斜 (重力加速度) を計測できる．

RFID リーダー (Phidget RFID) は，USB に直結する RFID リーダーと 10 個程度のタグのセットである (図 3 右)．135KHz の周波数帯を利用しており，アンチコリジョン (タグの複数読み取り機能) には対応していない．なお，RFID については，拙著「RFID を使ってみよう [19]」にて，詳しく紹介している．

サーボコントローラ (Phidget ServoMotor) は，サーボモーターとコントローラをセットにした，パソコンからサーボモーターを手軽に制御できるシステムである (図 3 中央)．1 度以下の正確な単位で，サーボモーターの動作角度 (最大約 180 度) を制御することができる．なお，Phidget ServoMotor は，モーター駆動に伴う消費電力の関係で，給電は AC アダプタで行う．

価格については，Phidget InterfaceKit は約 80CAD (1CAD 89 円, 2009 年 6 月現在)，InterfaceKit 用センサが 10CAD〜30CAD 程度，Phidget Accelerometer や Phidget ServoMotor などの USB に直結するセンサ／アクチュエータが 50CAD〜100CAD 程度である．全ての Phidget デバイスは，前述の公式 Web サイトを通して，日本国内から購入

†1 なお，Phidgets を含めて今回紹介するデバイスの多くは Windows/Linux/Mac など OS を問わずに動作するが，筆者自身はすべて Windows XP にて動作検証を行っている．

†2 http://www.phidgets.com/

できる．

最後に，Phidgets の特徴についてまとめる．Phigets は，パソコンと接続するだけで利用できるセンサ／アクチュエータが充実しており，回路設計や電子工作 (半田付けなど) をあまり行わなくても，基本的な実世界指向システムを構築できる．また，ライブラリの対応言語も多いため，好みの言語でプログラミングをしやすい．このような点から，Phidgets はプログラマ (ソフトウェア／エンジニア) にとって特に扱いやすい仕様となっている．一方，後述する Gainer や FT245RL と比較すると，価格がかなり高く，原則海外の Web サイトから購入するため入手性にやや劣る．

### 2.1.2 Gainer

Gainer [9] は，IAMAS (情報科学芸術大学院大学) の小林茂氏らにより開発された国産のデバイス／ツールキットであり，USB に接続する汎用 I/O デバイス (Gainer I/O) と，それらを制御するライブラリ群から構成される．

Gainer I/O は，デジタル入力／出力とアナログ入力／出力用のポートを各 4 本ずつ，合計 16 ポート備えている (図 1 中央)．アナログ出力[†3]を備えることで，LED の明るさなどをなめらかに変化させることができる．また，これらの入出力ポートの機能を用途に応じて 8 通りに変更することが可能である．たとえば，8 ポートをアナログ入力／残り 8 ポートをアナログ出力にしたり，全てのポートをデジタル入力／デジタル出力として利用することができる[†4]．Gainer I/O は直接センサなどを差し込むコネクタを備えないため，ブレッドボードやユニバーサル基板と組み合わせて，独自にスイッチ／ LED ／センサなどを取り付ける必要がある．図 4 に，Gainer I/O の利用例を示す．ここでは，Gainer I/O をブレッドボードに取り付け，デジタル入力にプッシュスイッチを，アナログ入力に半可変抵抗を，デジタル出力とアナログ出力に LED を一つずつ接続している．

†3 Gainer I/O のアナログ出力は，PWM (Pulse Width Moduration) 方式で実現されており，出力電圧自体は変化しない．よって，厳密にはアナログ出力ではないが，ここでは Gainer の呼称に従う．

†4 これらの機能は，PSOC という動的に内部回路を書き換えることができるマイコンを用いて実現される．

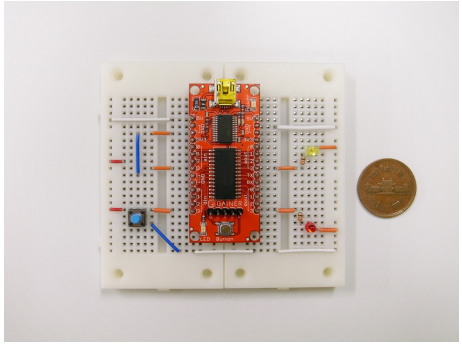

**図 4 Gainer I/O の利用例．ブレッドボードを用いて Gainer I/O とスイッチ／ LED などを配線する．**

Gainer のライブラリは，Flash ／ Max/MSP ／ Processing 向けに用意されている．また，Gainer I/O の制御は単純なシリアル通信で行われており，通信仕様も公開されているため他言語から利用することも一応可能である．

なお，Gainer I/O はハードウェアの設計／部品などの仕様が公開されている「オープンソースハードウェア」であり，SparkFun 版 Gainer I/O[†5]や，Gainer Mini[†6]など，いくつかの互換機が存在する．

価格については，Gainer I/O ／互換機共に 4000〜5000 円程度であり，国内外の Web サイト (e.g. TriggerDevice[†7], SparkFun[†8]，スイッチサイエンス[†9]など) や電子部品店舗 (マルツパーツ館[†10]) で購入できる．

最後に，Gainer の特徴についてまとめる．Gainer I/O は，Phidget InterfaceKit と比較するとやや安価で，国内で入手しやすい．また，入出力ポートの機能を用途に応じて変更できるため，単一のデバイスでも比較的柔軟な運用が行える．一方，Phidgets のように専用のセンサ／アクチュエータは用意されておらず，接続回路を自作するなどある程度の回路設計／電子工作が必要となるため，開発の敷居は高くなる．

†5 http://www.sparkfun.com/commerce/product_info.php?products_id=8480

†6 http://www.gainer-mini.jp/

†7 http://www.triggerdevice.com/

†8 http://www.sparkfun.com/

†9 http://www.switch-science.com/

†10 http://www.marutsu-group.jp/

また，オフィシャルにライブラリが用意されている言語が Flash ／ Max/MSP ／ Processing というデザイナー／アーティスト向けに偏っているため，一般的なプログラマにとってはやや扱いづらい．

このように，Gainer は，デザイン／アート系の学生のように、普段プログラミングや電子回路を専門としていない人々が、電子回路とプログラミング言語を併せて勉強するといった用途に適している．

#### 2.1.3 FT245RL(USB パラレルモジュール)

FT245RL モジュール [4] は，秋月電子通商†11で開発された USB に接続する汎用 I/O デバイスである (図 1 右)．FTDI 社†12の FT245RL という IC を用いて，合計 8 ポートのデジタル入力／出力を扱うことができる．スイッチ／ LED などの単純なセンサ／アクチュエータの制御が可能だが，センサからのアナログ入力を扱うことはできない．また，直接センサなどを差し込むコネクタを備えないため，ブレッドボードやユニバーサル基板と組み合わせて，独自にスイッチ／ LED ／センサなどを取り付ける必要がある．

ソフトウェアについては，FTDI 社の提供する DLL を用いて，C ／ C++／ C#といった言語で開発を行う．価格は 1000 円弱であり，秋月電子通商の Web サイト／秋葉原の店舗で購入できる．

このように，FT245RL モジュールは，デジタル入力／出力しか扱えず，周辺回路も自作する必要がある．しかし，価格が Phidget InterfaceKit や Gainer I/O と比較して非常に安く，パソコンからデジタル入出力 (スイッチ／ LED など) を扱う上で，最も安価な選択肢の一つである．

### 2.2 家電／照明機器連携

ユビキタス・インタフェースにおいて，テレビ／エアコン／照明などの家電／照明機器をわかりやすく制御することは重要なテーマの一つである [22]．

ここでは，こうした家電／照明機器をパソコンなどから手軽に制御するためのデバイス／ツールキットとして，X10 ／赤外線学習リモコン／ DMX について紹介する．

†11 http://www.akizukidenshi.com/

†12 http://www.ftdichip.com/

#### 2.2.1 X10

X10 [20] は，電灯線通信を用いて，家電／照明機器の電源 (オン／オフや電圧) を制御するための規格である．

X10 システムは，コンセントと家電／照明機器の間に取り付ける X10 デバイスモジュール (以下，X10 デバイス)，それらを制御するためのコントローラ (以下，X10 コントローラ)，パソコンから X10 デバイスを管理するための通信インタフェース (以下，X10 インタフェース) から構成される (図 5)．

X10 デバイスは個別に ID を設定することが可能であり，最大 256 種類の機器を操作することができる．X10 コントローラは，据え置き型のリモコンのような形状であり，複数の X10 デバイスをボタンなどで操作できる．X10 インタフェースは，パソコンから RS232 (シリアルポート) 経由で電灯線通信を行うための通信装置であり，X10 デバイスを直接制御したり，X10 コントローラの操作を検出する．

ソフトウェアについては，X10 インタフェースを制御するライブラリなどは用意されておらず，公開されている通信仕様に従ってシリアル通信で制御を行う．X10 システムは，海外の Web サイト [20] などから購入できる．価格は，X10 デバイスが 15USD (1USD= 約 98 円，2009 年 6 月現在) 前後，コントローラが 20〜30USD 程度，通信インタフェースが 40〜70USD 程度である．

X10 の特徴は，電灯線通信を用いることで，既存の電源ラインをインフラとして利用できるため，照明などの単純で数が多い機器の制御に向いている点である．特に大量に必要となる X10 デバイスは 1 つ 1500 円程度で購入できるため，現実的な価格で家庭内の照明制御システムなどを構築できる．一方，基本的に電源のオン／オフと電圧の制御しかできないため，AV 機器など複雑な家電機器の制御は難しい．

#### 2.2.2 赤外線学習リモコン

USB 赤外線学習リモコンは，一般的なパソコンから USB 経由で赤外線信号を制御して，さまざまな家電機器を制御できるデバイスである．ここでは，現在入手性が高い製品として，Buffalo PC-OP-RS1 ／玄人志向 KURO-RS (以下，KURO-RS) [10] を紹介

**図 5　X10 システムの一例．左: X10 インタフェース (Smarthome 1132b)，中央: X10 デバイス (Powerhouse X10 Lamp Module PLM03)，右: X10 コントローラ (Powerhouse X10 Mini Timer)**

**図 6　USB 赤外線学習リモコン (玄人志向 KURO-RS)**

する．

KURO-RS は，1 つの赤外線受信部と 2〜4 つの赤外線送信部を中心に構成される．赤外線の学習の際には，KURO-RS に USB 経由でコマンドを送り学習モードに切り替えた後に，赤外線受信部に向かって任意の家電リモコンから信号を送る．赤外線データは，バイナリ形式のファイルとしてパソコンに保存される．赤外線の送信は，KURO-RS に USB 経由で赤外線データを送ることで実現する．

パソコンから見ると，KURO-RS は USB シリアルポートとして認識され，公開されている通信仕様に従ってシリアル通信で制御できる．KURO-RS は，家電量販店や，国内 Web サイト (e.g. バッファローダイレクト[†13]) から 6000 円前後で購入可能である．

KURO-RS の特徴は，USB 経由でさまざまな赤外線信号を学習し，パソコンに保存して再利用できる点である．現在普及している複雑な家電機器 (AV 機器，エアコンなど) のほとんどは赤外線リモコンでの制御に対応しているため，KURO-RS はこうした家電機器を制御する手法として適している．

### 2.2.3　DMX

DMX512 (以下，DMX) [3] は，舞台照明／演出装置を制御するための RS485 互換の通信規格である．DMX システムは，さまざまな DMX デバイス，それらを制御するための DMX コントローラ，およびパソコンと接続するための通信インタフェース (以下，DMX インタフェース) から構成される (図 7)．DMX デバイスは，フルカラー LED 照明から特殊な演出装置 (e.g. フォグマシン／バブルマシン) まで多岐にわたっており，特に色彩や雰囲気を演出する本格的な照明環境の構築に適している．

DMX システムは,「チャンネル」と「データ」を用いて制御される．チャンネルは，DMX デバイス毎に設定できる ID ／ポート番号を兼ねた概念であり，1〜512 の範囲で指定する．データは，各チャンネルを制御する値であり，0〜255 の範囲で指定する．図 7 に示すフルカラー LED 照明 (American DJ 製 P36 LED) の場合，ベースとなるチャンネルを照明の DIP スイッチで設定した上で，そこから 3 チャンネルを「赤色」「緑色」「青色」の明るさ制御に利用する．たとえば，ベースチャンネルが 1 の場合，チャンネル 1 を 255，チャンネル 2 を 255，チャンネル 3 を 0 と指定すると，照明は黄色 (赤色＋緑色) に点灯する．なお，DMX デバイスは専用のケーブルを用いて数珠つなぎに接続でき，チャンネルが重複しない限り個別に制御できる．

ソフトウェアについては，DMX インタフェースを制御するための DLL が公開されており，C ／ C++ ／ C#などで開発を行う形になる．DMX システムの価格は，DMX デバイス (フルカラー LED 照明) は 6000 円〜，DMX コントローラは 5000 円〜，DMX インタフェースは 1 万円〜である．

最後に，DMX の特徴について述べる．DMX シス

†13　http://www.buffalo-direct.com/

**図 7 DMX システムの一例．左: DMX インタフェース (DASLIGHT 製 DVC2 128-M)，右: DMX デバイス (American DJ 製 P36 LED )**

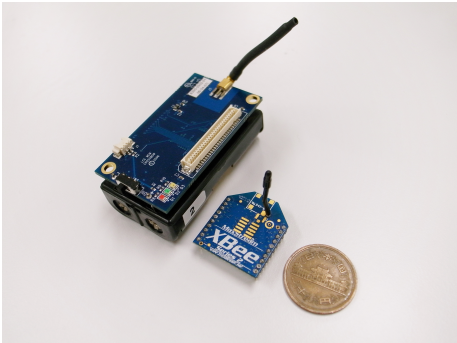

**図 8 無線センサシステムの一例．左: Crossbow Mote，右: XBee RF Module**

テムは全体的に高価であり，X10 や赤外線のように既存のインフラを活用することもできない．しかし，プロの舞台演出で利用されていることからも分かるように，さまざまな形状／特徴を持つフルカラー照明などが用意されており，空間の雰囲気を大きく変えるような高度な照明制御システムを構築できる．

### 2.3 無線センサシステム

家庭など生活環境における人々の状況や行動を認識するためには，複数のセンサを環境に設置する必要がある．こうした複数のセンサ情報を効率的に収集するためには，デバイス単体でセンサ情報を取得し，無線で送受信可能な無線センサシステムが適している．ここでは，代表的な無線センサシステムとして，CrossBow Mote と XBee を紹介する．

#### 2.3.1 CrossBow Mote

CrossBow Mote[13](以下，Mote) は， IEEE 802.15.4 に準拠した小型の Zigbee †14互換の無線モジュールであり，複数の Mote デバイスと，パソコンに接続する通信インタフェース，およびパソコン上で動作する専用ソフトウェアから構成される (図 8 左).

Mote デバイスは，さまざまなセンサ (加速度，温度，GPS など) を内蔵したセンサモジュールと組み合わせることで，用途に応じた無線センサシステムとして利用できる．

パソコン上の専用ソフトウェア (MoteView) は，各無線モジュールの動作状況を GUI で確認できる．また，無線モジュールから送られてきたセンサデータは，MoteView によって自動的に SQL データベースに保存される．よって，ユーザはデータベース処理を行うプログラムを書くだけで，センサデータを利用できる．

Mote デバイスの価格は無線モジュールが 25,000 円程度，センサモジュールが 25,000〜50,000 円程度，通信インタフェースが 50,000 円程度，MoteView が 100,000 円程度であり，国内代理店[13] で購入できる．

Mote の特徴は，数種類のセンサモジュールが用意されており，基本的なセンシングであれば電子工作を必要としない点である．また，MoteView がセンサデータを SQL データベースに格納してくれるため，SQL を扱うプログラムを書くことで，センサデータの処理を行うことができる．一方，Mote デバイス自体が高価であるため，多数の Mote デバイスを生活環境で利用するのはかなり敷居が高くなる．

#### 2.3.2 XBee

XBee[21] は，Mote 同様 IEEE 802.15.4 に準拠した小型の無線通信モジュールである (図 8 右)．パソコンと USB で接続して，ファームウェアを書き換えることで，「ピアツーピアの無線シリアル通信モジュール」や「メッシュ／スターネットワークを形成する Zigbee 互換の無線センサモジュール」として動作する．

†14 低速で転送距離が短い代わりに、安価で消費電力が少ないことを特徴とする短距離無線通信規格.

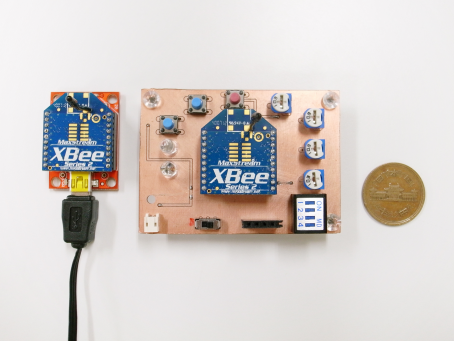


**図 9　無線センサシステムとしての XBee デバイスの利用例．左: パソコンに USB 接続するホストデバイス，右: 無線センサとして利用する XBee デバイス．4 つの可変抵抗 (アナログ入力)，3 つのプッシュスイッチ (デジタル入力) などを備え，ホストデバイスに入出力ポートの状態を一定間隔で送信する．**

無線シリアル通信機能は，各種マイコンなどと組み合わせて，パソコンとの接続を無線化するためによく利用される．無線センサ機能は，複数の XBee デバイス (Router/EndPoint) に接続されたセンサデータを，パソコンに USB 経由で接続された Xbee ホストデバイス (Coordinator) でまとめて受信し，パソコン上でセンサデータ処理することができる (図 9).

XBee デバイスの価格は，3000〜4000 円程度であり，国内外の Web サイト (Digi International [21], 前述のスイッチサイエンスなど) で購入できる．

XBee の特徴は，無線シリアル通信モジュール／無線センサモジュールのどちらとしても扱える上に，価格が安価である点である．特に，無線センサモジュールとして扱う場合，Mote よりずっと安価で手軽に購入できる．一方，専用のセンサなどは存在しないため，センサを接続するために回路設計や電子工作をある程度行う必要がある．また，開発用のライブラリなども付属しないため，通信仕様に従ってシリアル通信制御を行うプログラムを自作する必要がある．

## 3　MobiServer

前章で紹介したように，ユビキタス・インタフェースの開発をさまざまな側面から支援するデバイス／ツールキットが登場してきており，これらをうまく組み合わせて利用できれば，日曜プログラミングのような感覚で実世界指向システムを手軽に開発できる「日曜ユビキタス」を実現できるように思われる．

しかし，実際にシステム開発を行ってみると，これらのデバイス／ツールキットの扱い方はそれぞれ全く異なっている．たとえば，Phidgets ／ Gainer などは特定の言語向けのライブラリが用意されており比較的扱いやすいが，FT245RL ／ DMX などは DLL を直接制御することがある．また，X10 ／ USB 赤外線リモコン／ XBee などは，通信仕様を理解した上で，シリアルポートを介してコマンドの送受信を行う必要がある．このように，各デバイス／ツールキット毎にプログラミングの作法が異なるため，初心者にとって学習はまだまだ困難であった．さらに，複数のデバイス／ツールキットを組み合わせて使う場合は，熟練者にとっても開発が煩雑になりがちであった．

そこで，筆者は，これらの多様なデバイス／ツールキット群を統一の作法で手軽に扱うためのミドルウェア群「MobiServer」を開発／公開している [12]．図 10 に，MobiServer のコンセプトを示す．MobiServer は Phidget ／ Gainer ／ FT245RL などのセンサ／アクチュエータ制御や，X10 ／赤外線／ DMX などの家電制御，XBee などの無線センサ制御などを手軽に扱うことができるミドルウェア群の総称である．一例として，Phidgets の制御を行う PhidgetServer のスクリーンショットを図 11 に示す．

MobiServer は，(1) デバイスの動作テストを行うための GUI と (2) TCP ソケット経由でデバイスを制御するサーバーから構築される．(1) については，デバイスの動作確認を視覚的に行えるため，初心者でも理解しやすい．さらに，熟練者にとっても，システムトラブルの際に問題がハードウェア／ソフトウェアのどちらにあるのかを素早く判断する材料として有効である．(2) については，表 2 に示すような，シンプルなテキストメッセージを TCP ソケット経由で送受信するだけで，さまざまな言語から手軽にデバイスを制御することができる．

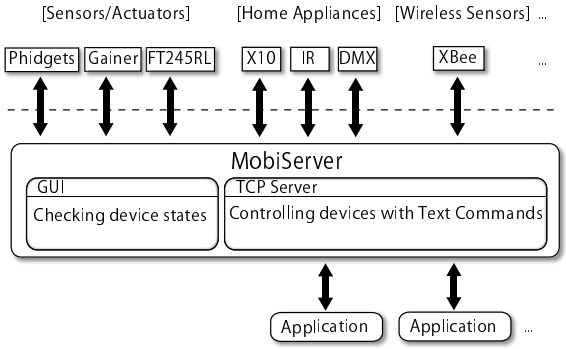


**図 10 MobiServer のコンセプト**

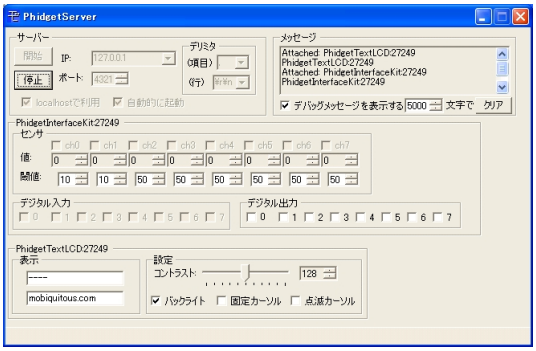


**図 11 PhidgetServer2.0 のスクリーンショット**

### 3.1 種類

次に，MobiServer のソフトウェア群について簡単に紹介する．全てのソフトウェアは，筆者 Web サイト [12] からダウンロード可能である†15．

**PhidgetServer2.0** USB 接続のセンサ／アクチュエータ群 Phidgets を制御する．対応する Phidget デバイスをパソコンに USB で接続すると，デバイスに応じた GUI が自動的に表示される (図 12 上段)．以下，全てのソフトウェア群に共通するが，(1) GUI を用いて，入力系デバイスの状態を確認したり，出力系デバイスを操作する，(2) TCP ソケットでテキストコマンドを送受信することでデバイスを制御する，ことができる．

†15 原稿執筆時点では，まだ未公開のソフトウェアも存在するが，開発は完了しており順次公開を進めていく．

**表 2 MobiServer のコマンド例**

| |
|---|
| Phidgets,In,InterfaceKit,27249,Analog,0,512<br>【Phidget InterfaceKit (ID:27249) のアナログ入力 ポート 0 が 512 へと変化】 |
| Phidgets,Out,Servo,3633,0,230<br>【Phidget ServoMotor (ID:3633) のポート 0 のモータ角度を 230 に変更】 |
| X10,Out,A,1,Power,1<br>【アドレス A-1 の X10 モジュールの電源を On にする】 |
| X10,Out,B,1,Bright,5<br>【アドレス B-1 の X10 ランプモジュールを明るくする】 |
| IR,Out,TV,Sharp1,Power<br>【テレビ (シャープ製) の電源を入れる】 |
| IR,Out,Audio,Sony2,Play<br>【オーディオコンポ (ソニー製) の再生を開始する】 |
| Parallel,In,A4001WtG,3,1<br>【USB パラレルモジュール (ID: A4001WtG) のポート 3 が High に変化】 |
| Parallel,Out,0,0,1<br>【USB パラレルモジュールのポート 1 を High に変更】 |

**GainerServer** USB 接続の汎用 I/O デバイス Gainer I/O を制御する．COM ポートと Gainer I/O の動作モードを指定することで，動作モードに応じた GUI が自動的に表示される (図 12 中段)．

**ParallelServer** FT245RL (USB パラレル) モジュールを用いて，デジタル I/O を制御する．デバイスを接続して，ParallelServer 上のボタンを押すと，デバイスをスキャンして GUI を表示する (図 12 下段)．

**X10Server** 電灯線通信を行う X10 デバイスを用いて，照明／家電の電源を制御する．X10 デバイスのアドレスなどを選択／登録すると，対応する GUI が表示される (図 13 上段左)．

**IRServer** KURO-RS (USB 赤外線学習リモコン) を用いて，さまざまな家電機器を赤外線で制御する．IRServer には，ある程度の数の赤外線信号がプリセットとして登録されており，製品分類／メーカー名を選択すると，対応する GUI が表示される (図 13 上段右)．また，赤外線の学習も GUI から手軽に行える．

**DMXServer** DMX インタフェースを用いて，フルカラー LED 照明などの DMX デバイスを制御する．DMX デバイス名と対応チャンネルを指定すると，対応する GUI (スライダ) が表示される (図 13 中段)．

**XBeeServer** パソコンに接続された XBee ホストデバイス (Coordinator) を用いて，複数の XBee デバイス (Router/EndPoint) に接続されたセンサデータを無線で受け取る．XBeeServer は，受信したセンサデータを解析し，検出された XBee デバイスに応じた GUI を自動的に追加する (図 13 下段)．

### 3.2 使い方

次に，MobiServer に共通する基本的な使い方を説明する．

**0. デバイスの接続**

各サーバーに対応するデバイスを PC に接続し，デバイスの接続操作を行う．詳細は対応デバイスによって異なるため，ここでは割愛する．

**1. サーバーの起動／停止**

サーバーを起動するためには，フォーム左上の「サーバー」において，任意のポートを指定し，「開始」ボタンを押す (図 14)．

- 標準では IP は自動的に取得される．「ローカルホストで利用する」をチェックしておくと，ローカルホスト (127.0.0.1) でサーバーを起動する．
- 各コマンドの区切り子 (終端文字) は「デリミタ (行)」のプルダウンメニューから選択できる．三種類の改行コード (”¥r¥n”, ”¥r”, ”¥n”) と，NULL 文字 (”¥0”) †16のいずれかを選択する．

†16 ActionScript(Flash) から扱う場合，NULL 文字がソケットの区切り子となる．

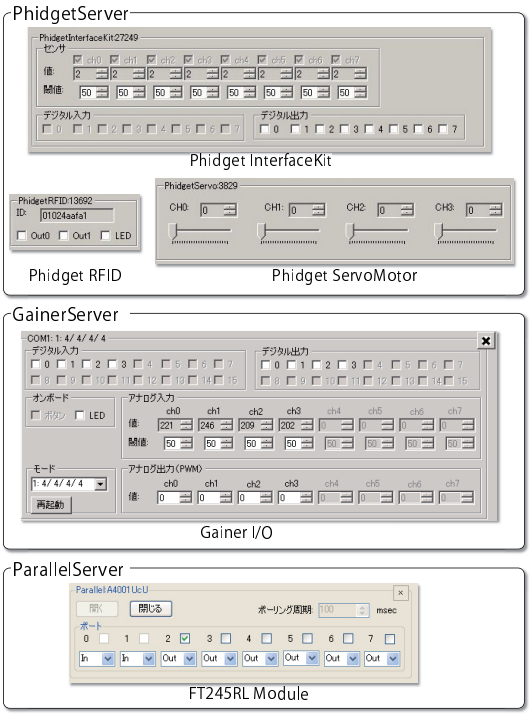


**図 12 MobiServer の GUI 事例 1．上段: PhidgetServer，中段: GainerServer，下段: ParallelServer**

- 各コマンド内の区切り子は「デリミタ (項目)」からカンマ (”,”) とタブ文字 (”¥t”) のいずれかを選択する．

次に，開始に成功すると，確認メッセージが表示され，停止ボタンがアクティブになる．この状態で，クライアントの接続を待機する． 最後に，サーバーを停止する場合は，「停止」ボタンを押すことで，全てのクライアントとの接続を切断し，サーバーを終了する．

**2. サーバーへの接続**

まず，各サーバーの動作している IP/Port を確認する．たとえば，IP:127.0.0.1，ポート:4321 で動作しているサーバーに telnet を用いて 接続する場合，以下のようにコマンドプロンプトから入力する．

- % telnet 127.0.0.1 4321

**3. デバイスの接続/切断**

デバイスが新たに接続/切断されると，以下のようなメッセージがサーバー→クライアントへ送信される．

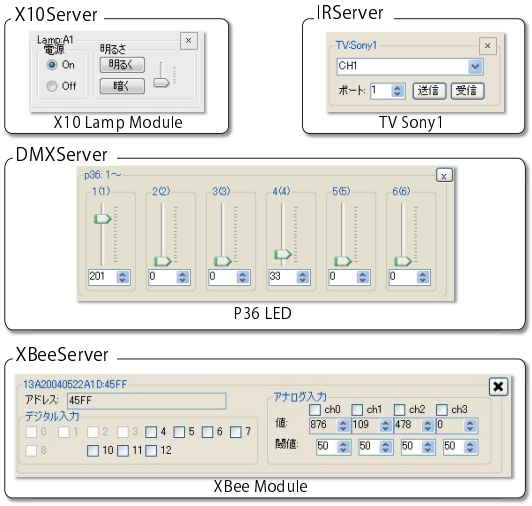


**図 13　MobiServer の GUI 事例 2．上段左: X10Server，上段右: IRServer，中段: DMXServer，下段: XBeeServer**

**図 14　MobiServer の TCP サーバー機能**

- Phidgets,Attach,InterfaceKit,27249
  【Phidget InterfaceKit (ID:27249) が接続】
- X10,Attach,A,1,Lamp
  【X10 Lamp モジュール (アドレス:A-1) が接続】
- Phidgets,Detach,InterfaceKit,27249
  【Phidget InterfaceKit (ID:27249) が切断】
- X10,Detach,A,1
  【X10 モジュール (アドレス:A-1) が切断】

**4. デバイスからの入力**

入力系デバイスの状態が変化すると，以下のようなメッセージがサーバー→クライアントに送信される．

- Phidgets,In,InterfaceKit,27249,Analog,0,512
  【Phidget InterfaceKit (ID: 27249) のアナログ入力 (ポート 0) の入力が 512 に変化】
- Parallel,In,A4001WtG,3,1
  【USB Parallel モジュール (ID: A4001WtG) のデジタル入力 (ポート 3) が High に変化】

**5. デバイスへの出力／設定**

出力系デバイスの制御，および入力系デバイスの設定は，以下のようなメッセージをクライアント→サーバーに送信することで行う．

- Phidgets,Out,InterfaceKit,27249,Digital,0,1
  【Phidget InterfaceKit のデジタル出力 (ポート 0) を High にする】
- X10,Out,A,1,Power,1
  【X10 モジュール (アドレス: A-1) の電源を On/Off する】
- IR,Out,TV,Sharp1,Power
  【赤外線デバイス (TV-Sharp1) の電源を入れる】
- Parallel,Out,0,0,1
  【Parallel モジュールのデジタル出力 0 を High にする】
- Phidgets,Set,InterfaceKit,27249,Threshold,0,50
  【Phidget InterfaceKit のアナログ入力 (ポート 0) の閾値を 50 に設定】
- Parallel,Set,0,2,0
  【Parallel モジュール (ID 未指定=全て) のデジタルポート 2 を入力モードにする】

なお，詳細な通信仕様や各ソフトウェア固有の GUI については，筆者 Web ページ [12] にて詳しく紹介している．

## 4　システム実例

本章では，MobiServer を活用した実世界指向システムの実例を示す．筆者らは，MobiServer を用いて実際にさまざまなユビキタス・インタフェースの開発を進めている．ここでは，タグタンス [18]，InPhase [16]，SyncDecor [17]，イルゴール [14] について紹介する．

タグタンス [18] は，フックに洋服をかけるだけで，手軽に洋服を撮影／タグ付けしてデジタル化できるシステムである．図 15 のように，一般的な二枚扉式のタンスの扉の内側の片方に，USB カメラ／照明／液晶ディスプレイを，もう片方にフックセンサを取り付ける．まず，ユーザがタンスを開くと，扉に付けられたリードスイッチで開閉を検出し，自動的に撮影用の照明が点灯する．次に，ハンガーに掛った洋服を

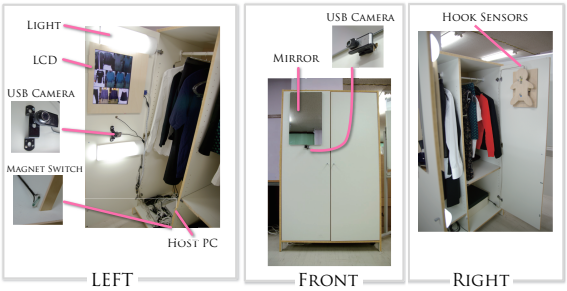


**図 15　タグタンスの外観**

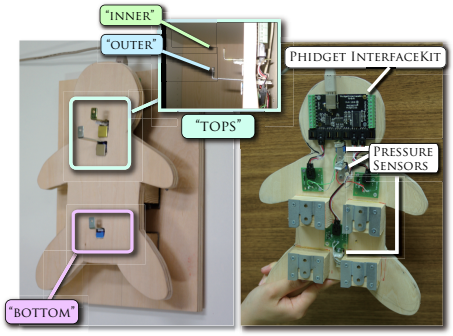


**図 16　タグタンスのフックセンサ**

フックセンサに掛けることで，衣服の画像が自動的に USB カメラで撮影される．フックセンサには，複数のフックが取り付けられており，フックを掛ける位置に応じて，洋服の種別 (e.g. インナー，アウター，ボトム) を判別することができる (図 16)．よって，撮影画像には，こうした洋服の種別や大まかな重さ，および撮影時刻などが自動的にタグ付けされる．さらに，Flickr などの Web サービスに直接撮影画像をアップロードすることもできる．タグタンスのシステム構成を図 17 に示す．フックセンサは 3 つの圧力センサが内蔵され，Phidget InterfaceKit ／ PhidgetServer を用いて制御する．扉の開閉はリードスイッチを用いて取得され，照明はリレーを用いて ON/OFF する．これらの制御は，FT245RL モジュール／ ParallelServer を用いて行われる．

InPhase [16] は，遠隔地にいる人々の行動が偶然一致したことを伝達することで，話題のきっかけ／親近感／連帯感などを誘発する遠隔コミュニケーションシ

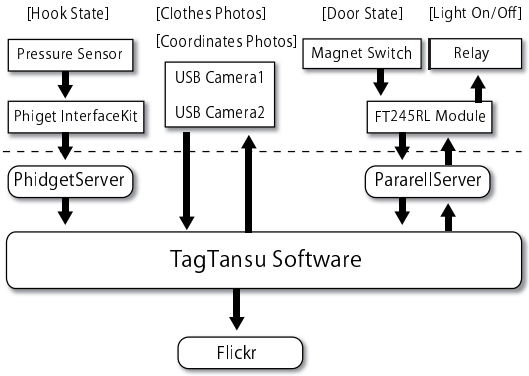


**図 17　タグタンスのシステム構成**

ステムである．家庭における人々の行動を検出するために，ドア／冷蔵庫／窓の開閉をリードスイッチで，ソファの着席／離席を圧力センサで検出する．センサの設置場所が分散することから，それぞれのセンサを XBee デバイスに接続し，無線センサシステムとして利用する．XBeeServer を用いることで，多数の XBee デバイスからのセンサデータを手軽に処理することができる．

SyncDecor [17] は，遠隔地間の照明やテレビ，ゴミ箱などの状態を同期させることで，遠距離恋愛中のカップルや単身赴任の夫婦間のコミュニケーションを豊かにすることを目指したシステムである．照明の制御に X10 デバイス／ X10Server を，テレビの制御に USB 赤外線リモコン／ IRServer を，ゴミ箱の蓋の開閉制御に Phidget ServoMotor ／ PhidgetServer を利用している．

イルゴール [14] は，オルゴールのメタファを用いて，家庭の様子を音で表現するシステムである．オルゴールの蓋の開閉を Phidget Accelerometer で，ゼンマイの動作状態を Phidget InterfaceKit に接続されたロータリーセンサで検出し，これらを PhidgetServer で制御している．

このように，MobiServer を用いれば，Phidgets ／ FT245RL ／ X10 ／赤外線／ XBee などのさまざまなデバイス／ツールキットを組み合わせたシステムを同一の作法で制御できるため，効率的に開発を進めることができる．

## 5 議論

本論文では，主にパソコンから利用できるデバイス／ツールキットを中心に紹介してきたが，マイコンベースのシステムを手軽に開発できるデバイス／ツールキットも急速に普及しつつある．ここでは，その代表的な存在である Arduino [11] について詳しく紹介し，パソコンベースとマイコンベースの開発について議論する．

### 5.1 Arduino

Arduino [11] は，USB 経由でプログラムを書き換え可能な汎用 I/O デバイス (以下，Arduino デバイス) と，Processing に統合された開発環境から構成されるデバイス／ツールキットである．Arduino デバイスは，Atmel 社の AVR マイコン[†17]をベースに構成されており，デジタル入力／デジタル出力／アナログ入力／ PWM 出力／シリアル通信機能などを備える．

Arduino デバイスは，Gainer と同様に，オープンソースハードウェアとして仕様が公開されており，さまざまな特徴を持つ互換機が存在する．以下に，代表的な Arduino デバイスを示す．

**Arduino Diecimila/Duemilanove** 最も普及している標準的な Arduino デバイスである (図 18 左)．単に Arduino と呼ぶ場合，Arduino Diecimila/Duemilanove (以下，オリジナル Arduino) を示す．両者の名称の違いは，製造時期によるものであり，機能的に大差はない[†18]．USB シリアル変換アダプタと USB ポートが搭載されており，パソコンから USB 経由で給電／プログラム書換／通信を行うことができる．Arduino 用として売られている拡張ボードの多くは，オリジナル Arduino を対象としている．

**Wiring I/O** [1] 基板サイズが大きい代わりに入出力ポートが多い Arduino デバイスであり，機能的にはほとんどオリジナル Arduino と同一である．なお，Wiring は Arduino の元となったプロジェクトであり，Processing に統合された開発環境などは Wiring から Arduino に受け継がれたものである．

**Arduino mini** 親指大の小型 Arduino デバイスである (図 18 中央左)．入出力ポート数はオリジナル Arduino とほぼ同一だが，USB 接続用の部品を備えないため，パソコンに接続するためには専用の USB 変換アダプタなどが必要となる．

**Boarduino** ブレッドボードに直接差せるように，縦長の形状として裏面にピンを出した Arduino 互換機．形状以外は，オリジナル Arduino とほぼ同一である．

**LilyPad Arduino** [2] 衣服などに縫い付けて利用する特殊な Arduino 互換機 (図 18 中央右)．手芸と電子工作の統合を目指した個性的なアプローチである．図 19 に示すように，専用のバッテリー／センサ／ LED などの基板が用意されており，円形の基板の周囲にあるピン穴同士を導電性糸でつなぐことで配線を行う．Arduino mini と同様に，パソコンに接続するためには専用の USB 変換アダプタなどが必要となる．

**Funnel I/O** [5] LilyPad Arduino をもとに開発された Arduino 互換機であり，無線通信モジュール XBee と組み合わせて利用する (図 18 右)．無線通信機能やバッテリー駆動端子／充電機能などを備える一方，Arduino のプログラム書換を USB 経由で行うことができないため，やや敷居が高い面がある．

ソフトウェアについては，Processing の開発環境を用いて，C ／ C++をベースに簡易化した独自言語を用いて開発を行う．プログラムはパソコン上でコンパイルされ，実行ファイルを Arduino デバイスに書き込む．

Arduino デバイスの価格は，2000 円〜4000 円程度であり，国内外の Web サイト (e.g. 前述の SparkFun，スイッチサイエンスなど) や電子部品店舗 (前述のマルツパーツ館など) で購入できる．

最後に，Arduino の特徴についてまとめる．Arduino は，Phidgets や Gainer と混同されることも多

†17 http://www.atmel.com/

†18 余談だが，Arduino はイタリアで開発が始まったプロジェクトであり，Arduino Diecimila/Duemilanove の基板の裏にはシルクスクリーンでイタリア地図が描かれている．

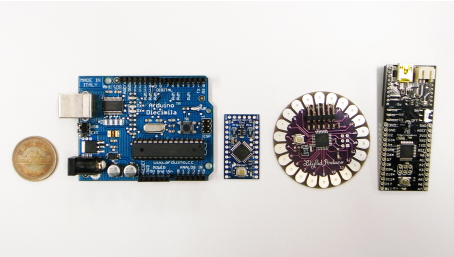

**図 18 Arduino 互換デバイス．左から，Arduino Diecimila，Arduino mini，LilyPad Arduino，Funnel I/O**

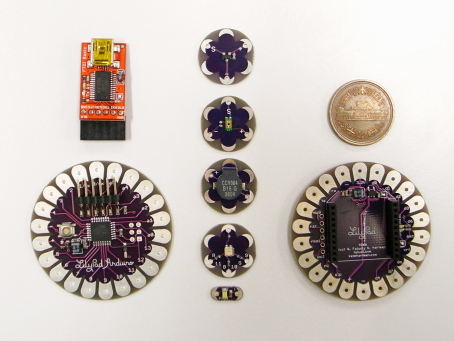

**図 19 LilyPad Arduino と専用センサ／アクチュエータ．左上: USB 接続アダプタ，左下:LilyPad Arduino，中央上から 温度センサ，光センサ，振動モーター，フルカラー LED，白色 LED，右: XBee 接続アダプタ**

いが，前述したように目的はかなり異なる．すなわち，Phidgets や Gainer がパソコン上からセンサ／アクチュエータを制御するのに対して，Arduino はスタンドアロンで動作するマイコン開発を簡易化することが主目的である．

Arduino の機能自体は Microchip Technology 社の PIC[†19]や Atmel 社の AVR などの従来のマイコンと変わらないが，(1) 専用のプログラム書込装置 (ライター) や通信回路 (USB シリアル変換部など) が不要，(2) 開発環境がフリーで Web からダウンロード可能，(3) 国内で 3000 円前後で手軽に購入可能，という特徴を持つ．このように，デバイス自体が入手しやすく，USB でパソコンに接続するだけでプログラム開発を始められる，という敷居の低さから，Arduino は急速に普及しつつある．

### 5.2 パソコンベースとマイコンベース

ここでは，ユビキタス・インタフェースの実装技術として，MobiServer ＋パソコンベースのツールキットの組み合わせ (以下，パソコンベース) と，Arduino のようなマイコンベースのツールキット (以下，マイコンベース) について比較し，議論する．

まず，パソコンベースのメリットは，(1) 多様なデバイス／ツールキットを利用可能，(2) 開発効率が高い，(3) 拡張性が高い，の 3 点である．(1) については，MobiServer と目的に応じた既存のデバイス／ツールキットを使い分けることで，デバイス開発を最小限に抑えることができる．たとえば，独自の赤外線リモコンを Arduino を用いて作ることは可能だが，そのためには，前述の KURO-RS に相当するハードウェア／ソフトウェアを自前で開発する必要があるため，開発工程が増大する．(2) は，MobiServer を用いることで，ハードウェア／ソフトウェアの問題を分離しやすく，さまざまな言語を利用できるため，開発効率が向上する点である．(3) は，さまざまなコンテンツ (画像，音楽など) や Web サービスとの連携が行いやすいため，システムの応用性／拡張性に優れる点である．

一方，マイコンベースの場合，(1) 小型で単体動作可能，(2) 低コストというメリットがある．(1) については，動作にパソコンが必要ないため，デバイスの小型化が容易であり，バッテリーでスタンドアロンで動作させることができる．(2) についても，パソコンを省くことで，特に大量にデバイスを作る場合，コストがかなり安くなる可能性がある．

こうした点を総括すると，システムにパソコンを組み込むことがサイズ／コスト的に可能な場合[†20]は，パソコンベースの開発を行った方が効率的だと考える．特に，画像／音楽などのコンテンツや Web サー

†19 http://www.microchip.com/

†20 最近では，ネットブックとよばれる，Windows XP 搭載の小型ノートパソコンが 3 万円前後で手に入るため，パソコンのサイズ／コストの問題はかなり緩和されている．

ビスとの連携を前提としたアプリケーションの開発は，マイコンベースではかなり面倒である．

その一方，スタンドアロンで動作することが重要なモバイル／ウェアラブルインタフェース[15] を実装する場合や，多数の小型デバイスを相互に連携させたい場合†21にはマイコンベースが有力な選択肢となる．

最後に，Arduino の登場は，マイコンベースの開発の敷居を大きく下げることには貢献したが，実際の開発プロセス自体はさほど大きく変化していない．今後，マイコンベースの開発を効率化するような実用的なミドルウェアの登場を期待したい．

## 6 おわりに

本論文では，ユビキタス・インタフェースの開発を支援するさまざまなデバイス／ツールキットについて説明した．そして，複数のデバイス／ツールキットを組み合わせて，日曜大工や日曜プログラミングのような感覚で，手軽にユビキタス・インタフェースを構築できる「日曜ユビキタスのためのミドルウェア群：MobiServer」を紹介した．さらに，MobiServer を用いたシステム開発の実例を示し，パソコンベース／マイコンベースの開発について議論した．本稿を，読者らのユビキタス・インタフェースの開発に役立てていただければ幸いである．

## 参考文献


[1] H. Barragan. *Wiring: Prototyping Physical Interaction Design*, 2004. Available: wiring.org.co.

[2] L. Buechley, M. Eisenberg, J. Catchen, and A. Crockett. The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceeding of the ACM Conference on Human Factors in Computing Systems (CHI 2008)*, pp. 423–432. ACM, 2008.

[3] DMX512. Available: www.soundhouse.co.jp/material/dmx.asp.

[4] FT245RL USB パラレル変換モジュール. Available: akizukidenshi.com/catalog/g/gK-01799/.

[5] Funnel. Available: www.yapan.org/main/funnel/.

[6] S. Greenberg and M. Boyle. Customizable Physical Interfaces for Interacting with Conventional Applications. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2002)*, pp. 31–40. ACM Press, 2002. Available: www.phigets.com.

[7] H. Ishii and B. Ullmer. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 97)*, pp. 234–241. ACM Press, 1997.

[8] 暦本 純一. 実空間に拡張する直接操作環境－複数コンピュータ環境への身体的アプローチ. 身体性とコンピュータ, pp. 295–308, 2000.

[9] S. Kobayashi, T. Endo, K. Harada, and S. Oishi. GAINER: a reconfigurable I/O module and software libraries for education. In *Proceedings of the 2006 Conference on New Interfaces for Musical Expression (NIME 2006)*, pp. 346–351, 2006. Available: www.gainer.cc.

[10] 玄人指向 KURO-RS 赤外線学習リモコン. Available: www.kuroutoshikou.com/modules/display/?iid=928.

[11] D. A. Mellis, M. Banzi, D. Cuartielles, and T. Igoe. Arduino: An Open ElectrPrototyping Platform. In *Alt CHI, CHI 2007*, Apr. 2007. Available: www.arduino.cc.

[12] MobiServer: 日曜ユビキタスのための手軽なミドルウェア. Available: www.mobiquitous.com/mobiserver.

[13] Crossbox Mote. Available: www.xbow.jp.

[14] M. Oki, K. Tsukada, K. Kurihara, and I. Siio. HomeOrgel: Interactive music box for aural representation. In *Adjunct Proceedings of Ubicomp2008*, pp. 45–46, Sept. 2008.

[15] 椎尾 一郎, 安村 通晃, 福本 雅明, 伊賀 聡一郎, 増井 俊之. モバイル&ユビキタスインタフェース. ヒューマンインタフェース学会論文誌, 5(3):313–322, 2003.

[16] H. Tsujita, K. Tsukada, and I. Siio. InPhase: a communication system focused on "happy coincidences" of daily behaviors. In *Extended Abstracts of CHI 2009*, pp. 3401–3406. ACM Press, Apr. 2009.

[17] 辻田 眸, 塚田 浩二, 椎尾 一郎. 遠距離恋愛者間のコミュニケーションを支援する日用品 "SyncDecor" の提案. 日本ソフトウェア科学会論文誌 (コンピュータソフトウェア), 26(1):25–37, 2009.

[18] K. Tsukada, H. Tsujita, and I. Siio. TagTansu:A Wardrobe to Support Creating a Picture Database of Clothes. In *Adjunct Proceedings of Pervasive2008*, pp. 49–52, May 2008.

[19] 塚田 浩二. RFID を使ってみよう 概要編. In *Software Design*, pp. 88–99. 技術評論社, Nov. 2004. Available: www.mobiquitous.com/pub/sd200411.html.

[20] X10. Available: www.smarthome.com/.

[21] XBee RF Modules. Available: www.digi-intl.co.jp/.

[22] 安村通晃. ヒューマンコンピュータインタラクションの現状と将来. システム/制御/情報, pp. 159–164, 2003.


†21 ただし，無線センサシステムのようにデータを一方向に送信するだけなら，前述の XBee ＋ XBeeServer を用いた方が手軽である．