

# WiiRemoteで学ぶプログラミング数学・物理

白井暁彦

shirai@mail.com

<http://akihiko.shirai.as/projects/BookWii/>

2008年9月30日版

## 概要

この PDF ファイルは SmartDoc によって自動生成されています。オリジナルファイルは「LearnMathPhysicsWithWiiRemote.sdoc.xml」です。

この PDF はだいたいのページ数をつかんだり、編集作業のために利用しております。最終的な印刷物とは全く違うものです。もちろん著作権は白井暁彦とその共著者にありますので無断コピーや配布はしてはいけません。

### 【この版における凡例】

- 「 」執筆中、まだ完成していない章・セクション
- 「 」日付など更新が必要な標記
- 「 」事実確認が必要な標記

# 目 次

<b>第 1 章 WiiRemote のススメ</b>	<b>1</b>
1.1 はじめに . . . . .	1
1.2 本書の構成 . . . . .	3
1.3 WiiRemote の基礎知識 . . . . .	3
1.3.1 任天堂のコンセプト . . . . .	3
1.3.2 WiiRemote 通信の仕組み . . . . .	3
1.3.3 ハードウェアスペック . . . . .	3
1.4 WiiRemote で何が創れる? . . . . .	4
1.4.1 巨大なイカロボットを操作する . . . . .	4
1.4.2 自由に氷柱を生やす(北陸先端大) . . . . .	6
1.4.3 ニオイで攻撃できるゲームを作る(金沢高専) . . . . .	6
1.4.4 超音波で触れるディスプレイを作る(東大) . . . . .	6
1.4.5 アリの巣を可視化する() . . . . .	6
1.4.6 バイオリンを作る . . . . .	7
<b>第 2 章 PC 上で利用する WiiRemote 事始め</b>	<b>8</b>
2.1 WindowsPC . . . . .	8
2.2 Bluetooth ホストアダプタとスタック . . . . .	8
2.3 HID クラス . . . . .	9
2.4 API . . . . .	9
2.5 アプリケーション / ツール . . . . .	10
2.6 コラム : WiiRemote を PC で使うのは違法? . . . . .	10
<b>第 3 章 Bluetooth で WindowsPC と接続する</b>	<b>12</b>
3.1 Bluetooth 製品を選ぶ . . . . .	12
3.1.1 IVT BlueSoleil . . . . .	12
3.1.2 東芝製スタック . . . . .	13
3.1.3 Microsoft 製スタック . . . . .	13
3.1.4 3.1.4 その他の環境について . . . . .	14
3.2 WiiRemote と PC を Bluetooth で接続する . . . . .	14
3.3 コラム : お勧めの Bluetooth 製品? . . . . .	14

<b>第 4 章 実験してみよう</b>	<b>17</b>
4.1 WiinRemote を使ってみる . . . . .	17
4.1.1 PowerPoint でプレゼンテーション . . . . .	17
4.2 GlovePIE を使ってみる . . . . .	17
4.2.1 GlovePIE スクリプト . . . . .	17
4.3 その他のツール紹介 . . . . .	18
4.3.1 WiiAcc . . . . .	18
4.3.2 WiiFlash . . . . .	18
4.4 コラム : Virtools(3DVia)+WiiRemote . . . . .	19
<b>第 5 章 センサーバーを自作する</b>	<b>21</b>
5.1 センサーバーのしくみ . . . . .	21
5.2 目には見えない赤外線 . . . . .	21
5.3 電子部品をそろえる・工作する . . . . .	24
<b>第 6 章 プログラミング環境のセットアップ</b>	<b>26</b>
6.1 Visual C++ 2008 Express Edition のセットアップ . . . . .	27
6.1.1 VC2008 のインストール . . . . .	27
6.2 WindowsDDK のセットアップ . . . . .	34
6.2.1 コラム : ISO ファイルのマウント . . . . .	39
6.2.2 コラム : VC 混在環境を解決する Version Selector . . . . .	40
<b>第 7 章 API 紹介</b>	<b>41</b>
7.1 tiny_hid.dll . . . . .	41
7.1.1 7.1.4 (temp)Wii Mice Test1 . . . . .	41
7.1.2 7.1.5 (temp)Wiimote fall detect Test1 . . . . .	41
7.1.3 Wii Remote Test1 の実行 . . . . .	42
7.1.4 wii_test1.cpp のリビルド . . . . .	43
7.1.5 wii_test1.cpp の流れ . . . . .	45
7.2 WiiYourself! . . . . .	45
7.2.1 7.2.1 Readme とライセンスの参考訳 . . . . .	45
7.2.2 7.2.2 WiiYourself!付属デモのテストとリビルド . . . . .	45
7.3 その他の API . . . . .	45
7.3.1 7.3.1 Wiim . . . . .	45
7.3.2 7.3.2 WiimoteLib . . . . .	45
7.3.3 7.3.3 API を使用しない接続方法 . . . . .	45
<b>第 8 章 WiiRemote で学ぶ、物理・数学・プログラミング</b>	<b>46</b>
8.1 8.1 最小の WiiRemote プログラム . . . . .	46
8.2 8.2 WiiRemote テルミン . . . . .	46
8.3 8.3 PowerPoint コントローラー . . . . .	46

8.4	8.4 姿勢検出 . . . . .	46
8.5	8.5 赤外線 4 点検出 . . . . .	46
8.6	8.6 赤外線 LED を使った測量 . . . . .	46
8.6.1	8.6.1 より広範囲を測定する場合 . . . . .	46
8.7	8.7 バイブレータと LED . . . . .	46
8.8	8.8 ポーリング受信と動作周波数の測定 . . . . .	46
8.9	8.9 認識の基本 . . . . .	46
8.10	8.10 Socket 通信 . . . . .	46
8.11	8.11 重力加速度を見る . . . . .	46
8.12	8.12 振りぬく力・パンチ力測定 . . . . .	46
8.13	8.13 モーション解析 . . . . .	46
<b>第 9 章 3DCG 応用編</b>		<b>47</b>
9.1	9.1 3DCG + 物理エンジンとの組み合わせ (高橋さん担当分) . . . . .	47
9.1.1	9.1.1 Direct3D9 の初期化 (固定機能?) . . . . .	47
9.1.2	9.1.2 Bullet の準備と初期化 . . . . .	47
9.1.3	9.1.3 球やボックスのような基本形状の利用 . . . . .	47
9.1.4	9.1.4 「pick」3D オブジェクトを持ち上げて遊ぶ . . . . .	47
9.1.5	9.1.5 「だるま落とし」 . . . . .	47
9.1.6	9.1.6 「ヌンチャク」ジョイントを使った例 . . . . .	47
<b>第 10 章 プロジェクトサンプル</b>		<b>48</b>
10.1	10.1 レースゲームへの応用 . . . . .	48
10.2	10.2 赤外線マーカーを使ったモーションキャプチャ . . . . .	48
10.3	10.3 お絵かきソフトへの応用 . . . . .	48
10.4	10.4 「振る」の認識 . . . . .	48
10.5	10.5 劍術アクションへの応用 . . . . .	48
10.6	10.6 フリースロー . . . . .	48
<b>第 11 章 Wii Flash を使ってみよう</b>		<b>49</b>
11.1	11.1 環境設定 (CS3 のセットアップ DL とインストール) . . . . .	49
11.2	11.2 WiiFlash のダウンロードとセットアップ . . . . .	49
11.3	11.3 WiiFlash の基本機能 . . . . .	49
11.4	11.4 サンプルのテストでみる基本的な使い方 . . . . .	49
11.5	11.5 WiiFlash プロジェクトサンプル . . . . .	49
11.5.1	11.5.1 なりきりライトセーバー (原さん) . . . . .	49
11.5.2	11.5.2 おえかき (尾崎さん) . . . . .	49
11.5.3	11.5.3 DropShadow(尾崎さん) . . . . .	49
11.5.4	11.5.4 ChunChun(原さん) . . . . .	49

<b>第 12 章</b>	<b>未整理ネタ倉庫</b>	<b>50</b>
12.1	コラム：WiiFit を作った男たち . . . . .	50
12.2	12.0.5 赤外線反射式インターフェース . . . . .	50
12.3	(大内さん提供のネタ) . . . . .	50
.1	索引 . . . . .	51
.2	参考文献 . . . . .	51
.3	登録商標 . . . . .	51

# 第1章 WiiRemoteのススメ

## 1.1 はじめに

本書を手に取った賢い読者の貴方、まず貴方に質問をしたいとおもいます。

貴方はこの本にどんな期待をしていますか？

『WiiRemoteで学ぶプログラミング数学・物理』

貴方が手に入れたい知識は WiiRemote の使い方？ プログラミング？ 数学？ 物理？ はたまた欲張りにも「そのすべて」でしょうか？ まず最初にキッパリとお断りしておきますが、この本は任天堂 Wii コンソールに関するハッキング本ではありません。ゲームを改造したり不正なコピーをするための行為は、ゲームそのものの面白さを奪うだけでなく、そのゲームタイトルを世に生み出すために魂を削って制作した人々にも、深い悲しみと経済的ダメージを与えます。貴方がゲームを愛するなら、そんな行為に時間を費やしてはいけません。

この本が目指すところは、その正反対の方向です。

WiiRemote(Wii リモコン、Wiimote 様々な呼称がありますが本書では Wi-iRemote に統一)を使って、プログラミングを学びます。ついでに数学や物理の使い方も演習を通して学びます。

目標としては、貴方が大好きであろう、コンピュータを使ったゲーム、その想像力や可能性を最大限に加速するための「最初の武器」を身につけるための「きっかけ」を与えることを目指しています。想定している読者としては、プログラミングを覚えはじめた中学生・高校生から、専門学校生や大学の1~2年生ぐらいを対象にしています。文系・理系、工学・芸術系は問いません。

そんな読者の皆さんに、WiiRemote をはじめとする新しいヒューマンインターフェイスを使った、エンタテイメントシステムの世界の「開拓の面白さ」、学校で学ぶ物理や数学の「使いこなし方」、そしてほんのすこし「世界の広さ」を理解してもらえばいいかな、と考えています。

すでにコンシューマゲーム機や PC、Flash などでプロ級のゲームプログラミングを行っているシニアエンジニア以上の読者の貴方は、きっとここまで読んで『なんだ、この本？ 役に立つかな？』と思ったかもしれません。お察しの通り、この本には任天堂の守秘義務にふれるような事は一切書かれていません(幸いなことに、私はこの6年以上コンシューマーゲーム機の開発

には関わっていません！純潔です！）。ですから Wii に関する内部仕様に関しては、ぜひ任天堂から提供されているであろう公式の技術文書を参考にしてください。しかし、公式以上の WiiRemote の使い道やヒント、価値が、本書にはあるかもしれません。

実は著者（私、白井暁彦）は、これまで 10 年近く、ヒューマンインターフェイス、リアルタイムコンピュータグラフィックス、物理シミュレーション、触覚インターフェースといったバーチャルリアリティの基礎技術をエンタテインメント産業の実用の世界に使えるようにするための技術の研究開発を行ってきました。大学の研究室内だけでなく、企業の研究開発に身を置いた時期もありますし、日本やフランスの大学で芸術学部や工学系の学部～大学院生まで幅広くの学生の指導・教育もしてきました。特に、国際学生対抗バーチャルリアリティコンテスト (<http://ivrc.org/>) のプロデュースや、Laval Virtual ReVolution という国際 VR 作品公募展のチアマンを通して、世界中から集まつた多くの新規な（新奇な）デバイスを使ったインタラクティブ作品とその開発を行った作者に出会ってきました。多くが若い学生である、世界のステージに立つその作者たちを応援してきた経験を通して、著者は「新しい表現、すばらしい技術革新を生み出す若者」が、必ずしも「十分な基本技術に身についているわけではない」という現象に気がつきました。近い将来、ゲーム開発の現場や、研究者、電機メーカーのエンジニア、フリーランスのアーティストになっていくであろう彼らが、何に気づき、何を武器にして、新しい世界のフィールドに立っていくのか、いつも興味深く見守っています。

この本に書かれている情報の多くは、上記のような『まだ誰もやったことがない、コンピュータを使ってあたらしい表現をしよう！』としている若者に対して『攻撃力を加える武器を与えよう！』という視点によって書かれています。こういう「考え方のヒント」はなかなかまとまった書物にはしづらいものです。その証拠に、インタラクティブシステムの工学的な開発・デザインに関するまとまった書籍や、体系づけられた学問は、この 10 年以上（日本では）形になつていません。それだけ成長途上であり、経験や経験則に依存する世界もあるからです。

貴方がもし、ゲーム開発チームのシニアエンジニアなら、そんな「未知数の後輩」に対して、ある時は事細かにソースコードを紐解き、ある時は高校の数学 1 の教科書を持ち出して、熱っぽく語らなければならないときもあるでしょう（しかも開発の真っ最中に！）。しかしそのすばらしい講義で解説した、貴方なら『1 秒で理解できるような数式』を目の前に、何日もウンウンと唸っている後輩がいたら『しょうがないな、この本でも読んでおけよ』と机上に関係ある箇所を付箋付きで置いて置いてあげたりすると、時間の無駄にならなくてよいかもしれません（それぐらいの期待値で本書を購入リストに加えていただければ、外すことはないでしょう）。

この本を書き上げるにあたり、非常に数多くの友人知人、協力者の方々に

お世話になりました。全ての人に厚く御礼を申し上げたいのですが、まず出版社の九天社の大内さん、そしてゲーム産業に革命を与えた任天堂の開発者のみなさんに、言葉にできない感謝の気持ちを伝えたいと思います。そして特に技術的な協力者のみなさんについては、本文中にてお世話になった技術と共に、可能な限り本名でご紹介していきたいと思います。

最後に、激烈な昼間の仕事の合間の数少ない休暇、貴重な家族との時間の削減に文句を言いながら執筆やコーディングに捧げてくれた妻・久美子、息子の成彦と隆佳に、愛と感謝を伝えたいと思います。

さあ、読者の皆さん、次は貴方がレヴォリューションを起こす番です！

2008年冬 しらいあきひこ

## 1.2 本書の構成

## 1.3 WiiRemote の基礎知識

%現在、世界中の有志により、WiiRemote を PC で利用できるようにする取り組みがされています。ここではちょっとしたツールやアプリケーションを中心に気軽に試せるものを紹介していきたいと思います。%また実際にこのようなツールを使うために、WiiRemote と PC を接続する方法についても解説します。

### 1.3.1 任天堂のコンセプト

(WiiRemote の基本的な考え方、何がレヴォリューションなのか？) (任天堂スペック)

### 1.3.2 WiiRemote 通信の仕組み

(Wii 本体と WiiRemote、PC と WiiRemote がどのように通信を行っているか)

### 1.3.3 ハードウェアスペック

(ハードウェアスペック、公開非公開含めて) Bluetooth コントローラは「Broadcom 2042」であることが報告されています。Broadcom 社のホームページで公開されている情報によると \cite{BCM2042}、BCM2042 は、適応型周波数ホッピングと高速接続をそなえた Bluetooth 2.0 完全互換仕様で、PC 用マウスやキーボード応用に適した設計になっているチップであると解説されています。

%The BCM2042 is fully compliant with the version 2.0 Bluetooth specification, including adaptive frequency hopping and fast connection, which are essential to mouse and keyboard applications in personal computers.

## 1.4 WiiRemote で何が創れる？

ここでは WiiRemote を使って、どんな楽しいことができるか、最新の学生プロジェクトを中心に紹介していきたいと思います。

### 1.4.1 巨大なイカロボットを操作する

設計図とか仕様書のようなものは学生がゴリゴリ作ったものなのでないと思われます。IKABO 公式ページにもあるようにイカ型ロボットはロボットフェスインはこだて市民の会という組織が絡んでいて未来大がやっているのはソフトウェアの部分です。

製作者の味岡真広さんから

>-プロジェクトタイトル >-制作者もしくは研究室名

「未来大 IKABO Project」

>-プラットフォーム Visual C++ MFC アプリケーション

>-使用している API WiiYourself! - native C++ Wiimote library v0.96b.

<http://wiiyourself.gl.tter.org>

これを参考に Wii リモコンから 3 軸加速度+ボタンの情報を取得しています。

>-簡単な概要 wii リモコン+ヌンチャクから 3 軸加速度を常時取得 3 軸加速度から 3 軸の傾き情報にプログラム上で計算して変換 (PC で) その傾きの変化に合わせてイカロボットの腕の動作を決める (PC で) 有線 (USB)or ネット (DirectPlay) でイカロボット実機へ動作指示を通信 (イカロボットは簡単なマイコンポートで動作) イカロボットが目の前にいるなら目で、遠いところにいるなら、アプリに組み込んだリアルタイム動画配信によってイカロボットが動いているのを見る

>-技術的な / 芸術的なアドバンテージ wii リモコンで実物のロボットを動かすところ？ユーザの動きをイカロボットが真似てくれるネットを介してもできる

>-YouTube 等の URL (YouTube)[http://jp.youtube.com/watch?v=4P\\_alu527SY](http://jp.youtube.com/watch?v=4P_alu527SY) (IKABO 公式ページ)<http://ikarobo.com/ikabo1.html>



図 1.1: IKABO



図 1.2: IKABO



図 1.3: IKABO

- 1.4.2 自由に氷柱を生やす(北陸先端大)
- 1.4.3 ニオイで攻撃できるゲームを作る(金沢高専)
- 1.4.4 超音波で触れるディスプレイを作る(東大)
- 1.4.5 アリの巣を可視化する()

”Atta Texana Leafcutting Ant Colony: A View Underground” <http://www.siggraph.org/s2008/attendees/newtech/23.php>

The Atta project maps tunnels and chambers of a vast leafcutting ant colony. A ground-penetrating-radar scan was translated into a 3D model that can be viewed on an immersive visualization system that scales the viewer to ant size. The scanning is nondestructive and is the first time GPR has been used to map a living ant colony.

To achieve this goal, the project combines the site-specific nature of an indexical system, GPR, with the ability of an algorithm to parse the data. The model retains a formal connection with its subject and can be distributed and viewed in many different ways.

Enhanced Life When combined, all ants in the world weigh about as much as all humans. Yet we tend to view ants as tiny individuals rather than the superorganisms scientists have discovered them to be. The immersive system scales the viewer to ant size and reveals a different perspective on colony architecture.

Goal Methods of modeling ant colonies include pouring casting material into a nest, digging it up and piecing it back together, or using a bulldozer to scrape away successive layers of soil and measuring the diameter of the holes. The goal of this project was to find a nondestructive way to create a model.

Innovations This is the first time GPR has been used to model an ant colony. A method was developed to translate data from proprietary scanning software to a format suitable for high-end 3D modeling, by building layers around known voids. The system may be adaptable to other projects with similar parameters.

Vision This project is part of a larger endeavor to document a landscape using technology in nondestructive ways. One might think of threatened ecosystems as exotic and distant, but much rural land continues to disappear beneath development. There is still much to be discovered in a single patch of one's backyard.

Contributors Carol LaFayette Fred Parke Lauren Simpson Audrey Wells  
Igor Kraguljac Visualization Department, Texas A and M University

Carl J. Pierce St. Lawrence University  
Tatsuya Nakamura Starz Animation  
<http://news.bbc.co.uk/2/hi/technology/7563372.stm>  
<http://www.youtube.com/watch?v=8auq5Jx-0bI>

#### 1.4.6 バイオリンを作る

Project Title: Wiiolin Creator: Jace Miller Platform: Windows API: Wi-  
iuse for Wii remote interaction, also FMOD library for sound sample play-  
back Short abstract: The Wiiolin is a virtual violin that uses Nintendo's  
Wii controller and sensor bar to produce music in a unique way. To play the  
Wiiolin, the Wii remote is held in the left hand like the neck of a violin and  
the sensor bar is used as a bow. The motion and orientation of the sensor  
bar is analyzed at interactive rates to determine which virtual string is be-  
ing played. Button presses on the Wii remote are mapped to similar finger  
positions on the violin. The virtual bow motion and finger position infor-  
mation is combined to determine which pre-recorded note sample should be  
played. Tilting the Wii remote downwards allows the Wiiolin to be played  
like a cello, with sound samples to match the instrument. Technical/artistic  
advantage: This is the only work I am aware of that attempts to recognize  
sensor bar movement as a primary means of music generation.

I'll have to check with the professor in charge of the Immersive Visualization project at the university if he wants to publish information about the immersive system that was displaying the ant colony scenes, but I'll e-mail you back with the details if they agree.

Your book sounds like a good project. Let me know when it is finished  
and I may look for a copy of it.

## 第2章 PC上で利用する WiiRemote始め

ここでは、WiiRemote を PC で利用するための仕組み、概念を説明します。「PC で利用する」といっても、C++などを使ってディープに開発したり、既にコンパイルされているツールを使ったり、Flash などの外部のアプリケーションと連動させたりと、いろいろな方法がありますが、基本的な構成を図 \ref{fig:Layer1.png} のような階層構造で表現してみました。

下の層から「WindowsPC」、「Bluetooth ホストアダプタ」、「Bluetooth スタック」、「API」と「Win32」、そして「アプリケーション / ツール」となっています。

### 2.1 WindowsPC

ハードウェアや OS といったプラットフォームになる部分です。本書は Windows Vista と Windows XP をベースに解説していますが、Mac や Linux といったハードウェア・OS でも WiiRemote が利用できることが報告されています。

### 2.2 Bluetooth ホストアダプタとスタック

「Bluetooth ホストアダプタ」は各種 Bluetooth 機器と無線通信するハードウェアです。Bluetooth 規格に準拠したマイクロプロセッサとアンテナで構成されています。

もしお使いの PC が標準で Bluetooth を装備しており、かつ後で紹介する接続実験に成功するのであれば、新たに準備する必要はありません。PC に装備されていない場合には PC パーツ・電器店で購入してください。だいたい 2,000 円代ぐらいいから USB タイプのアダプタが入手可能です（最新・高級品である必要はありません！）。

そして、ホストアダプタと併せて、WiiRemote を PC で利用するための準備で最も重要なのが「Bluetooth スタック」と呼ばれるソフトウェアです。Bluetooth ホストアダプタのドライバにあたるソフトウェアで、各種 Bluetooth 機器との無線接続を管理します。もしこの Bluetooth スタックが WiiRemote

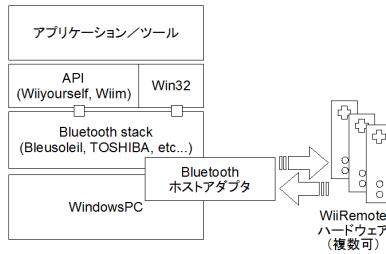


図 2.1: 図 1

との接続をサポートしていない場合、お使いの PC で WiiRemote を利用することは困難になります。新しく購入することを検討する場合、各種 Bluetooth スタックによって接続に癖があるので注意が必要です（次章を参考にしてください）。

## 2.3 HID クラス

Bluetooth スタックは、接続するハードウェアにあわせて様々なクラスを提供します。WiiRemote との接続では、HID(ヒューマンインターフェースデバイス)というクラスを利用します。実はこの HID クラスは USB のマウスなどと同じ種類のものです。「Bluetooth 接続でも、基本的には USB の HID クラスの接続と同じ」と覚えておくと良いでしょう。ディープにプログラミングをする方は、後に紹介する DDK の

## 2.4 API

HID クラスを開いて、WiiRemote との通信をする部分です。「API(Application Programming Interface)」と表現していますが、アプリケーション開発のための便利なプログラミングインターフェース、クラス構築のためのコードです。世界のハッカーたちが「Wiimote ドライバ」など様々な呼称で呼び、数多く

のオープンソースプロジェクトが開発されています。加速度センサへのアクセスなど基本的な機能に的を絞ったもの、サウンド再生までサポートするような高機能なものなど様々存在します(後に解説)。APIは実は、突き詰めればWin32の関数、特にDDK(Driver Development Kit)を利用してコーディングされています。他のプラットフォーム、例えばLinuxなどの場合は、デバイスのインストールさえ成功すれば、あとはデバイスファイルへの読み書き関数だけでアクセスできる、といった環境もあるようです。

## 2.5 アプリケーション / ツール

こちら「アプリケーション / ツール」層が本書を手に取った皆さんのが最も興味のある部分なのではないでしょうか? WiiRemoteをつかったゲームや、後に紹介するFlashFlashなどが他のプログラムと通信をする部分です。

このように「WiiRemoteをPCで利用する」とひとことで言っても、幅広く、深い部分の知識が必要になります。逆を言えば、WiiRemoteを使ったゲームやアート作品など、アイディアを実現する上では、上位の階層でのプログラミング手法はまったく依存しません。皆さんが得意としているOpenGLやSDL、DirectX、Win32やMFC、コマンドラインプログラムやC++/CLI、そしてFlashやMaxMSPといった様々なツールやプログラミング手法で利用することができる、というわけです。

## 2.6 コラム：WiiRemoteをPCで使うのは違法？

ときどき「WiiRemoteをPCで利用するのは違法では？」という発言をする人がいます。よく考えもしないで「違法」なんて言うのは失礼な話なのでですが、簡単に筆者の見解だけを述べると「No」です。将来的には違法となる可能性もあるかもしれません、まずはこれがどこの国の何法で違法なのかを考えてみるとよいと思います。例えば「PC用のマウス」をPS2やXboxに接続した場合、何が違法でしょうか? ユーザーはWiiRemoteのパッケージを開けるときにEULA(エンドユーザーライセンス承諾書)を読んだりサインする必要もありませんし、基本的には「ユーザーのリスクにおいて、サポート外」である行為であるかもしれません、それを行うことで発生する不具合や故障の修理代や損害賠償をユーザーが請求したりするのでなければ、自由です。(本書もそうなのですが) 発売側にそれ以上の権利が存在するのであれば、独占禁止法などで、ユーザではなく、発売側に違法性が疑われてしまします。

しかし業務的にWiiプラットフォームにおいてゲームを開発している人は別です。このような開発者は、それぞれの所属する企業と任天堂やパブリッシャ間ににおいて、守秘義務契約が結ばれているはずです。もしこのような開

発者がインターネットで閲覧可能な場所に、任天堂の内部資料などを公開したら、守秘義務契約違反や著作権法違反で訴えられるかもしれません（それ以前に取引停止は覚悟でしょう）。しかし、そのような開発者は元から守秘義務契約によって、任天堂より公式に提供される開発キットや技術資料を読むことができるのです。業務ではなく、ボランティアや自分の技術的興味で活動している世界中のハッカーとは本質的に立場が異なります。

しかしながらそういう法的視点とは別に、歴史的には多くのコンシューマ（家庭用）ゲーム機において、ゲームコントローラや本体間の通信方式は公開されていません。理由はいくつかありますが、独占目的と言うよりは、商業的な品質維持という目的があるかもしれません。

しかし任天堂 Wii の設計では Wii 本体と WiiRemote 間は Bluetooth 接続接続を利用しています。双方には Bluetooth ハードウェアが内蔵されていますし、WiiRemote は前述の通り Bluetooth の仕様上では HID(ヒューマンインターフェースデバイス) として定義されています。

任天堂のハードウェアエンジニアは優秀ですから、WiiRemote を PC に接続しようとする人が想定できなかったとは思えません。本気で隠蔽するのであれば、Bluetooth の接続プロトコルを修正して、許可されていない Bluetooth ホストからの接続を拒否するような仕組みや、パスキーを設定して想定されていないホストからの接続を遮断することもできたでしょう。しかし任天堂はそういう隠蔽策に労力を払うのではなく、オープンな仕様である Bluetooth をそのまま使用することを選んだようです。我々、世界中の WiiRemote 利用者はその英断に感謝すべきだと思います（結果として WiiRemote の売り上げには協力できているのかもしれません！）。

ところで Wii 本体で公式に利用できる Web ブラウザ機能「インターネットチャンネル」においても WiiRemote をユーザが利用できます。これに関しての技術仕様は任天堂のホームページに掲載されています \cite{InternetChannel}。この技術仕様には JavaScript によるボタン情報やセンサーバーをつかったカーソルの位置や傾きの取得方法が記載されています。また、「利用上の注意」として、「当社は、この拡張機能に関して、一定の商品性を有していること、特定の目的への適合性を有していること、第三者の知的財産権（特許権、著作権、商標等）を侵害していないこと等を含め、一切の法律上の保証を行いません。この拡張機能を使用したことによって被るいかなる損害に対しても、当社は責任を負いません。当社は、この拡張機能を使用する方に対して、いかなる権利をも付与するものではありません。当社は、この拡張機能に関するサポートは一切行っておりません。」と記載されています。

# 第3章 Bluetoothで WindowsPCと接続する

ここでは第2章で紹介した内容をより具体的に、WiiRemoteをWindowsPCで利用する手順について解説します。Bluetooth接続に関わる問題は、PC上でWiiRemoteでいろいろなアプリケーションを開発する上で、常に頭を悩ませるブラックボックスとなることが多いので『自分の環境では問題なく使っているよ！』という人もこのステップで、知識として知っておくことをお勧めします。

## 3.1 Bluetooth製品を選ぶ

WiiRemoteを使ったPC上の開発で、最も良いスタートを得る方法は、実績のあるBluetooth製品を選ぶことです。コンシューマーゲーム機周辺機器とはいえ、ここから先は何の保証もない世界です。先人の知恵を共有し、不要な労力を避けるためにも、まずは安定して動作する環境を準備しましょう。本節では、多く実績が報告されている順にBluetooth製品を紹介していきます。

### 3.1.1 IVT BlueSoleil

世界中のハッカーによるレポートを読んでいると、「IVT BlueSoleil(ブルーソレイユ)」Bluetoothスタック＆ドライバが最もよく使われているようです。IVT社のホームページによると \cite{IVT}、1999年からBluetoothソフトウェア製品の開発をリードし続けており、BlueSoleilは2008年4月現在で2500万ライセンスが販売されているそうです。実際、BlueSoleilはBluetoothホストアダプタを製品にしている周辺機器メーカーにOEMとして採用されていて、多くの製品において、購入したUSB Bluetoothアダプタにドライバとして同梱されています（一部機能制限版の場合もあるようです）。もちろんBlueSoleilのホームページ \cite{BlueSoleil}においてオンライン購入することができます。

2008年6月時点での最新のバージョンは「BlueSoleil 6.0 Mobile」で価格は19.95ユーロです。WiiRemoteとの接続以外にも、携帯電話との接続やワイヤレスヘッドセットなどにも利用できますので、手持ちのBluetooth製



図 3.1: BlueSoleil のホームページ

品が不満で、かつ運良く BlueSoleil がサポートしていれば、買っても損はないでしょう(特に WindowsXP から Vista に乗り換えた場合など)。BlueSoleil のホームページから「Download」を選ぶと製品版と同じソフトウェアをダウンロードできますので、動作が確認できたらライセンスを購入すると良いでしょう。

### 3.1.2 東芝製スタック

DELL や Lenovo などのノート PC に装備されている内蔵 Bluetooth スタックとして有名なものは、OEM で供給されている TOSHIBA 製のスタックです。Bluesoleil とは若干異なった挙動をするため注意が必要ですが、WiiRemote との接続は可能なものが多く、BlueSoleil との互換も多いようです。筆者が発見している東芝製スタックの問題は「4つ以上の WiiRemote と同時接続できない」という点です(ソフト的な問題なので将来的には解決するかもしれませんね)。

### 3.1.3 Microsoft 製スタック

Windows XP や Windwos Vista もコントロールパネルに「Bluetooth デバイス」というアイコンがあります。こちらは Microsoft 製の Bluetooth スタッ

クで、対応している Bluetooth 製品が装着されると動作します。しかし残念なことに、WiiRemote とは相性が悪いようです。Windows Vista 環境においてはペアリングに失敗する、接続できても値の取得に失敗する、といった多くの不具合が報告されています（意外にも Microsoft 製のスタックの方が不具合が多いというのは残念なことですが）。Windows XP 上においては、ペアリングにコツがあり「接続が完全に終了するまで 1+2 ボタンを押しっぱなしにすること」でうまく接続できるようです。この現象は、Microsoft の Bluetooth スタックが、サービスを列挙し PIN（パスワード）を求めている間に、WiiRemote の同期モードが終了してしまうということが原因かもしれません。

### 3.1.4 3.1.4 その他の環境について

Windows 環境において、上記で紹介のないその他のスタックについては、「購入できない」、「試してみたが現在のところ成功していない」、「WiiRemote とのペアリングが成功しない」といった状況です。もし手持ちの Bluetooth 製品で成功しているものがあれば是非皆さんで共有した方が良いでしょう。

また本書では扱いませんが、Windows 以外の環境では MacOS と Linux で比較的簡単に WiiRemote が利用できる状況になってきました。特に、MacOS では Bluetooth は OS の標準機能で利用でき、接続ツール、アプリケーションなど様々なソフトウェアが登場してきています。「OSX Wiimote Enabler」という、近くにある WiiRemote を見つけてペアリングしてくれるソフトウェアなどは便利そうです \cite{OSX-Wiimote-Enabler}。

MacOS と同じく、Linux 環境では Bluetooth との接続に特別なソフトウェアは必要ありません。特に最近急速にユーザ数が増えていくディストリビューション「Ubuntu」では Ubuntu7.10(Gutsy) 以降、標準的なソフトウェアで利用できるようになってきています。

## 3.2 WiiRemote と PC を Bluetooth で接続する

（PTM-UBT3S をつかってステップバイステップで解説。Windows Vista は結局安定して動作する環境がないので割愛します。）

## 3.3 コラム：お勧めの Bluetooth 製品？

上記のように、現状はインフラとも言うべき WiiRemote との接続の根幹に、ブラックボックス化した Bluetooth スタックを経由した HID クラスを



図 3.2: プリンストンテクノロジー社「PTM-UBT3S」

利用するしかありません。ところがこのスタックは未保証（当然ですが…）な上に、不安定なため、最初のうちは不可解な動作やトラブルといったことに悩まされることも多いでしょう。公開されているフリーウェアもアプリケーション作者が自分の環境で利用しているハードしか試しておらず、特定の環境でしか動かない…といったこともよく見受けます。本書では、可能な限り、安心して利用できるハードウェアを使うことをお勧めします。欲を言えば、WiiRemote の接続を完全に安定して行えるような特別な Bluetooth スタックや HID クラスがあればいいのですが…現状はありようのまま使うしかありません。

フリーウェア「WiinRemote」の作者の HP では、動作確認済み環境を開いていますので参考になるかもしれません (<http://onakasuita.org/wii/>)。ただし、対応している OS は Windows2000 から XP,2003 まで、Vista については情報がありません（公開当時では BlueSoleil すら Vista 対応できていなかつたことも原因でしょう）。

筆者が個人的に愛用しているのはプリンストンテクノロジー社の「PTM-UBT3S」です。とても小さな USB コネクタサイズの Bluetooth ホストアダプタで、同梱されているスタックは東芝製ですが、Windows Vista での動作確認もとれています。

本当は Bluetooth 製品それぞれに動作確認情報を出せば良いのですが、調べてみると製品のパッケージには同梱されているスタックの情報は多くの場合記載されていません。購入して調査した後「製品に同梱されている」と標記したところで、製品のバージョンやリビジョン、インストールされる側の OS によって全く異なる、というケースもありました（おそらく同梱するソフトウェアのライセンス料が製品の価格の大部分を占めるからでしょう）。

幸いなことに Bluetooth 製品は日々、低価格化が進んでいます。購入に失敗したら買い直してもそれほど痛い価格ではなくなってきました。むしろ読者の皆さんのが問題にしているのは、ターゲットプラットフォームで、「Windows XP か Vista か？」ではないでしょうか？この問題も頭が痛いのですが、最近

では特に Vista 特有のアプリケーションでなければ、WindowsXP なのではないかな、と考えています。もちろん Windows 系でなくて良いのであれば、Linux や MacOS のほうがオープンで、余計な出費をしなくて良い状況であるといえます！

# 第4章 実験してみよう

## 4.1 WiinRemote を使ってみる

WiinRemote は 2006 年末、Wii 本体が発売された直後にもっとも有名になった WiiRemote を PC で利用できるユーティリティです。「おなかすいた族」(<http://onakasuita.org/wii/>) にて公開されています。WiiRemote を使って Windows のカーソルを動かしたり、加速度センサや赤外線センサの状態を観察することができます。

2007 年 1 月に公開された「WiinRemote\\_v2007.1.13.zip」が最終版ですが、現在でも多くのユーザに利用されているようです。

スクリーンショット、

### 4.1.1 PowerPoint でプレゼンテーション

PowerPoint での利用方法等を紹介

## 4.2 GlovePIE を使ってみる

GlovePIE は最も使用されているコントローラエミュレーターです。Carl Kenner によって開発されています。もともとヴァーチャルリアリティのためのデータグローブ「5DT」で、さまざまなゲームをプレイするためにジョイスティックやマウスをエミュレーションするために開発されていたようですが、その成長の過程で WiiRemote をサポートし、有名になりました。特徴として、すべてのエミュレーションを専用のスクリプトで記述します。例えば「A ボタンをショット、B ボタンをボムに」といったゲームそれぞれの割り当てを、自分で書くことができるのです。

[http://carl.kenner.googlepages.com/glovepie\\_download](http://carl.kenner.googlepages.com/glovepie_download)

スクリーンショット、使い方の流れなど。

### 4.2.1 GlovePIE スクリプト

GlovePIE スクリプトについてサンプル

このサンプルコードで、センサバーを使って WiiRemote をマウスのように使うことができます。

リスト 1: IRMouse.PIE

---

```
// Wiimote mouse script for Windows desktop
// Requires a sensor bar

var.ButtonFreezeTime = 250ms
var.PointerBump = KeepDown(Pressed(wiimote.A),var.ButtonFreezeTime) or \
    KeepDown(Pressed(wiimote.B),var.ButtonFreezeTime)
Wiimote.Led1 = true

// Mouse movement
if wiimote.PointerVisible but not var.PointerBump then
    mouse.x = wiimote.PointerX
    mouse.y = wiimote.PointerY
end if

// Mouse Buttons
mouse.LeftButton = Wiimote.A and KeepDown(Wiimote.PointerVisible,0.5s)
mouse.RightButton = Wiimote.B and KeepDown(Wiimote.PointerVisible,0.5s)
mouse.MiddleButton = Wiimote.Home and KeepDown(Wiimote.PointerVisible,0.5s)
mouse.XButton1 = Wiimote.Minus
mouse.XButton2 = Wiimote.Plus

// Mouse Wheel
if wiimote.Up then
    mouse.WheelUp = true
    wait 30ms
    mouse.WheelUp = false
    wait 30ms
end if
if wiimote.Down then
    mouse.WheelDown = true
    wait 30ms
    mouse.WheelDown = false
    wait 30ms
end if
```

---

### 4.3 その他のツール紹介

#### 4.3.1 WiiAcc

VisualBasic で作成された、加速度センサの値を確認するプログラムです。

#### 4.3.2 WiiFlash

WiiFlash は Flash 環境で WiiRemote を使えるようにするツールです。

## 4.4 コラム：Virtools(3DVia)+WiiRemote

CAD(Computer Aided Design) や 3DStudioMax などの 3D データから、インターラクティブな 3D グラフィックスアプリケーションを制作できるツールに「Virtools(ヴァーツールス)」があります (<http://www.virtools.com/>)。世界最大の CAD メーカー、フランス「ダッソーシステムズ (Dassault Systemes)」のグループで、最近では「3dvia」というブランドで展開されています。

Virtools は高価な産業向け製品であることもあり、日本ではそれほど有名ではありませんが、欧米では最も利用されている可視化プラットフォームです(筆者もフランスでは日常的に使っていました!)。モデルや画像などのリソースに GUI だけでほとんどのインターラクティブデザイン関係が作れてしまう、という点が画期的で、プログラミング不要です。もちろん深い開発をする場合はプログラミングも必要ですが、レンダラーやプラグインなどほとんどのソースは公開されています。

産業用の CAD データから、Virtools を使って Web 上で公開されるインターラクティブな商品広告の開発、プロトタイプ製品のユーザ試験などを開発することができますが、もともと Virtools はゲーム用プロトタイピングツールでした。ゲーム開発のプロトタイプ、つまりゲームの流れや面白さの根幹に関わる部分の設計をプランナーが GUI で行って、最終工程である最適化やプラットフォームの独自部分などをプログラミングで行う、といったゲーム開発手法です。

Virtools は公式にライセンスを受けたツール開発パートナーですので、ソニー PSP 用や任天堂 Wii 用の Virtools が存在します。任天堂 Wii の開発ライセンスを持っているゲーム開発企業であれば、WiiRemote 関係のプラグイン入手することもできるそうです。

Virtools の開発者コミュニティは活発で、スワップミート (<http://www.theswapmeet-forum.com/>) で様々な情報やソース、プラグインが共有されています(PC 上で利用できる WiiRemote のプラグイン関係も多数あります)。将来ゲーム制作を目指す学生さんや、フリーのゲーム企画者にとって、これは協力なソリューションです。Virtools を使って PC 上でゲームのプロトタイプを制作すれば、すばやくアイディアを形にできますし、資金や技術的なリスクを回避できるからです。

筆者も Virtools 上で WiiRemote を利用するゲームを何件か開発しましたが(後述)、プラグインの開発を含め非常に高速に、凝ったものを作ることができます。お勧めです。是非 Virtools のホームページから各種デモを見てみてください。

->

### Virtools(3DVia)に関する情報

Virtools(英語) <http://www.virtools.com/>

日本では三徳商事とクレッセントが代理店を行っています。コンテンツの制作サポートや技術サポート、教育支援、学校向けライセンス販売などリセラー各社の得意分野がありますから、ぜひ問い合わせてみてください。

三徳商事 (Virtools 関係) <http://www.san-toku.co.jp/VirtoolsOnlinePage/VirtoolsIndex.htm>

株式会社クレッセント <http://www.crescentvideo.co.jp/virtools/>

### WiiYourself! on Virtools.

Se'bastion Kuntz (<http://cb.nowan.net/blog/>)

# 第5章 センサーバーを自作する

WiiRemote の赤外線センサは非常に多機能で高速ですが、このままの状態では、センサーバーを Wii 本体に接続していなければ使えません。せっかく PC で WiiRemote が使えるので、Wii 本体がなくてもよいように、センサーバーの仕組みを知り、自作に挑戦してみましょう。

WiiRemote の加速度センサーだけ使う予定の読者の方や、センサーバーを Wii 本体に接続して利用する方は、この節は読み飛ばしていただいてもかまいません。

## 5.1 センサーバーのしくみ

センサーバーは、名前だけ聞くと『中にセンサーが入っている』ように聞こえますが、実際には赤外線センサーは WiiRemote 内に実装されており、センサーバー内部にセンサーは存在しません。

センサーバー内部には、左右にそれぞれ 5 つの赤外線 LED が実装されています。Wii 本体と接続しているケーブルは、電源ケーブルで、信号を送って同期したり、変調（周波数を変えて明度や速度を調整すること）したり、といった凝ったことはしていないようです。単純に直流電流を使って、同じ明るさで点灯しています。

同期や変調といった複雑な電子回路の場合は、自分で作るのは大変ですが、LED 点灯回路ぐらいであればそれほど難しくはありません（中学生レベルの電子回路です）。この LED 点灯回路を赤外線 LED を用いて自作すれば、センサーバーは必要なくなります。PC で WiiRemote を利用するのに、いちいち Wii 本体を起動してセンサーバーを点灯させる必要はありませんし、赤外線センサーを使った自作の作品を利用する上の自由度もでてきます。

## 5.2 目には見えない赤外線

さて、ここでは赤外線について学んでおきましょう（十分に知識がある方は読み飛ばしてください）。まず、世の中の光にはすべて「波長」があります。波長が変わると色が変わります。虹やプリズムを通して太陽の光を分解してみることができますが「赤橙黄緑青藍紫」という順番に並んで見えます。赤色に近くなればなるほど長い波長、紫色に近くなればなるほど短い波長です。



図 5.1: センサーバー赤外線をデジカメで撮影した様子

人間が肉眼で見ることができる波長「可視光」は実際には限りがあって、赤の 800nm(ナノメートル) から紫の 400nm 程度といわれています(人によって個人差はあります)。そしてこの赤より長い波長、紫より短い波長をそれぞれ「赤外線」「紫外線」と呼んでいます。

赤外線には 800nm 近辺の「近赤外線」と 900nm 以上の「遠赤外線」があります。それ以上波長が長くなると「マイクロ波」「メートル波」「電波」と呼ばれます。同様に紫外線もさらに波長が短くなると「X 線」や「ガンマ線」など呼び方が異なります。呼び方が異なっても結局のところ、光は光です。人体に吸収されたりされなかったり、お湯が沸いたり、無線通信できたり、と各波長ごとにいろんな利用上の特性がありますが、特に WiiRemote を使う上では波長 1000 ~ 800nm の近赤外線の特徴を知っておくとよいでしょう(分野によっては 2.5  $\mu$  メートルぐらいまでを近赤外と呼ぶこともあります)。

前述のとおり赤外線は目に見ることができませんが、可視光に近いため、目に見える光に似た拡散や反射が観察できます。目に見えないという理由から、自動ドアの接触センサー(フォトインタラプタ)や、テレビのリモコン、軍用や監視カメラの暗視スコープ(ノクトビジョン)などに使われています。こうしてみると、街の中は赤外線センサーだらけなのです！赤外線は熱を発する物体なら何でも、黒体放射で観測することができます(温度に応じて波長が変わります)。白熱電球なども熱と赤外線を出しますが、WiiRemote のような可視光や熱が必要でないセンシングに使う場合は効率が良くないので、より安定した赤外線光源として、赤外線 LED の発光を使う場合が多いよう

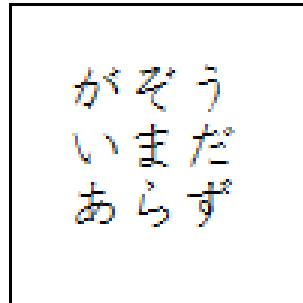


図 5.2: ( 未描画イラスト) 波長と虹

です \cite{HikariMap}。センシング用途では、赤外線 LED 光源にあわせて、フォトダイオード (PD) やフォトトランジスタといった特定の波長の光に対して反応する半導体とセットで利用されます。TV のリモコンや携帯電話や PC の近距離通信に使う IRDA もこの赤外線 LED と半導体素子のセット構成されています。

目には見えない赤外線ですが、デジカメや携帯、Web カメラなどの画像センサーを使うことで、赤外線を画像として見ることができます。本来、これらのカメラに利用されている画像センサーである「CCD」や「CMOS」は、特定された幅の波長の光しか電子に変換できないのですが、赤・緑・青のそれぞれの人間が画像として利用するための波長以外にほんの少し、可視光の外側の波長に対するセンサー感度を利用して「見えないものを見る」ことができます。夜中や暗闇でも撮影できるカメラ「ノクトビジョン」や「ナイトスコープ」は、この仕組みを使って目に見えない赤外線という明かりをつかって撮影できるのです (フィルターを使って赤外線域を見えなくしている製品もあります)。この知識は非常に有効で、センサーバーを自作したときや動作確認をする上で、赤外線が見えるカメラを手元においておくと、非常に便利です。

さて WiiRemote の場合はどうなっているのでしょうか？公開されている資料を見る限りでは、任天堂が台湾の PIXART 社に特注して開発した、特別な赤外線画像センサーを利用しているようです。推測の範囲を出ませんが、PSD(Position Sensing Device) という低解像度の半導体で、画像ではなく赤外線の明かりの重心の位置を高速に出力できます。一般的な画像処理が毎秒 30-60 フレーム程度なのに対して、この PSD は 2 次元の重心位置を出力するだけなら毎秒 400 フレームぐらい出せるデバイスもあります。同じような赤外線センサーはゲーム製品では「剣神ドラゴンクエスト」などにも利用されていたことがあります。WiiRemote に利用されているセンサーの特徴は「フレーム内に入った複数の赤外線の点が高速に出力できる」という点のようですが (コストの安さも重要な利点でしょうね！)。

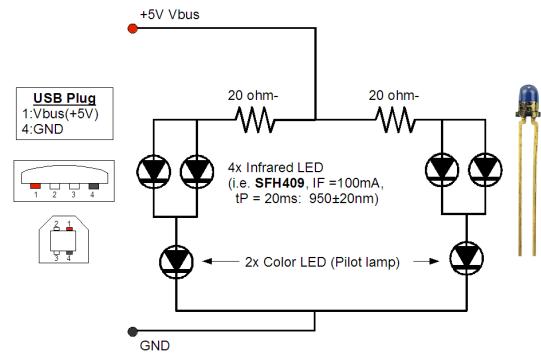


図 5.3: 自作 USB センサーの回路例

### 5.3 電子部品をそろえる・工作する

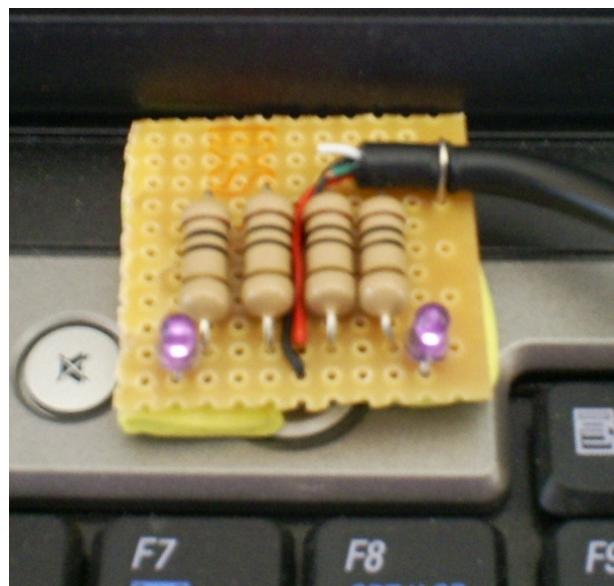


図 5.4: 小型の自作 USB センサーバーの制作例

## 第6章 プログラミング環境のセットアップ

ここでは、これ以降の開発の基本となる、Windows 環境で WiiRemote を使ったプログラミングを行うための環境構築について解説します。

本書では WiiRemote のためのプログラミングに必要なコーディング言語として、もっとも一般的な C/C++を解説します。現在、世界中のボランティアにより、Java をはじめ様々な言語に移植されてきていますが、環境の違いなどによる混乱を防ぎ、シンプルに理解できるよう、C/C++言語を採用しました。WiiRemote のようなデバイスとの通信プログラムにおいては、実際にはオブジェクト指向プログラミングに特化した実装は少ないため「(C++ではなく)C 言語しか知らない」という読者にも理解は難しくないはずです。

以後の解説は「プログラミングの経験はあるが普段使わないので忘れてしまった」、「(大学の研究や自主制作などで)WiiRemote を使いたいがプログラミングは苦手...」、「普段はゲームプログラミングなどでグラフィックスプログラミングなどをしているが、デバイス制御などはほとんど経験がない」といった幅広い読者を考慮して解説します（そのため、若干冗長な解説になる場合もあることをお許しください）。

セットアップの流れとしては以下のようになります。

**Visual C++のセットアップ** コンパイラ、IDE(統合開発環境)である Microsoft の Visual C++ 2008(以下、VC2008)をインストールします（無料で利用できます）。Visual Studio をお使いの場合や、VC2003 以降をお使いの方は、特に新規にインストールし直す必要はありません。

**WindowsDDK のセットアップ** Windows Driver Development Kit(以下、DDK)のセットアップを行います（無料で利用できます）。こちらの DDK には、HID デバイスと PC の通信を円滑に行うためのライブラリが含まれています。なお、DDK は最新のマイクロソフト用語では Windows Driver Kit(WDK)と呼ばれるようになりました（WDK には DDK に開発したドライバのテストツールなどが付属しています）。

**API のセットアップ** 上記 (1),(2) だけでも WiiRemote を使ったプログラミングはできるのですが、WiiRemote の機能を使いこなすためには世界中の開発者が開発した API(ドライバ、SDK など呼び方は様々ですが

本書では Application Programming Interface;API に統一) を利用するのが近道です。オープンソースで開発され、公開されているものが多く、たいていは無料で利用できます(ライセンスによる)。本書では主に「WiiYourself!v1.0」を利用して解説します。

サンプルプログラムのリコンパイル 最後に簡単なプログラムのリコンパイルと動作確認を行います。

## 6.1 Visual C++ 2008 Express Edition のセットアップ

ここでは Microsoft Visual C++ 2008 Express Edition のセットアップ、サンプル開発について解説します。既に Visual C++ や .NET といった開発環境をお使いの方は、読み飛ばしていただいても問題ありません。

### 6.1.1 VC2008 のインストール

まずマイクロソフトのホームページから Microsoft Visual C++ 2008 Express Edition をダウンロードします。

「Web インストール(ダウンロード)」をクリックすると、Web インストール版セットアップファイル「vcsetup.exe」をダウンロードすることができます。このホームページには Visual Studio 製品を使うまでの役に立つ情報がたくさんあります。とりあえず VC2008 を使ってみたい方は「はじめての方のためのインストール方法」を読んでみると良いでしょう。

必要なハードディスク容量の確認、起動中のアプリケーションの終了などをやってからインストールウィザード「vcsetup.exe」を起動します。

「セットアップの品質向上プログラム」はチェックしてもしなくとも、どちらでもかまいません。「次へ」をクリックして進みます。

ライセンス条項をよく読んで「同意する」を選んでください。また「Visual Studio でオンラインの RSS コンテンツを受信して表示できるようにする」も特に問題がなければチェックしましょう。イベントやサービスパックなどの更新情報が VC2008 起動直後に表示されるスタートページに自動的に表示されるようになります(ここで RSS 受信の設定は後でもオプション 環境スタートアップで変更できます)。「次へ」をクリックして進みます。

#### Visual C++ 2008 Express Edition

<http://www.microsoft.com/japan/msdn/vstudio/express/>



図 6.1: Microsoft Visual Studio Express 製品のホームページ

「インストールオプション」を選択します。ここで表示されている 3 つのオプションは、どれもインストールしなくても問題ありません。「MSDN Express ライブラリ」は F1 キーで呼び出せるドキュメントで、オンライン版の方が充実しているのですが、筆者は電車の中でコーディングをすることが多いのでインストールしています（オンライン版と統合して利用できます）。「SQL Server 2005」、「Silverlight」は使う予定がなければインストールしなくて良いでしょう。「次へ」をクリックして進みます。

「コピー先フォルダ」とダウンロードパッケージのリストです。「インストールするフォルダ」は本書ではデフォルトのままとして解説します。ダウンロードリストの中に「Windows SDK」が入っているのが助かります（以前の Express Edition では別途インストールする必要がありました）。「インストール」をクリックするとダウンロードとインストールが実行されます。

以上は WindowsXP シリーズにおける VC2008 セットアップの流れです（Windows Vista の場合は「管理者権限で実行」をする必要がある点などいくつか細かい点が異なりますがほぼ同じ流れです）。興味がある方は VC2008 のウィザードやオンラインチュートリアルなどを使って簡単なプログラミングを試してみると良いでしょう。

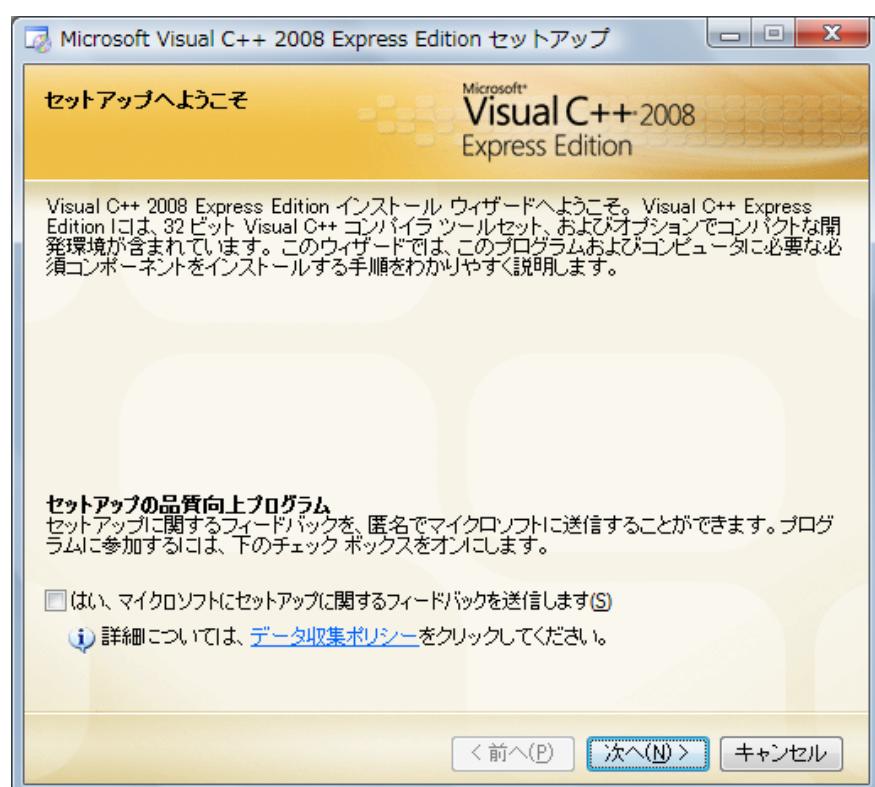


図 6.2: VC2008 インストール ウィザード (step1)

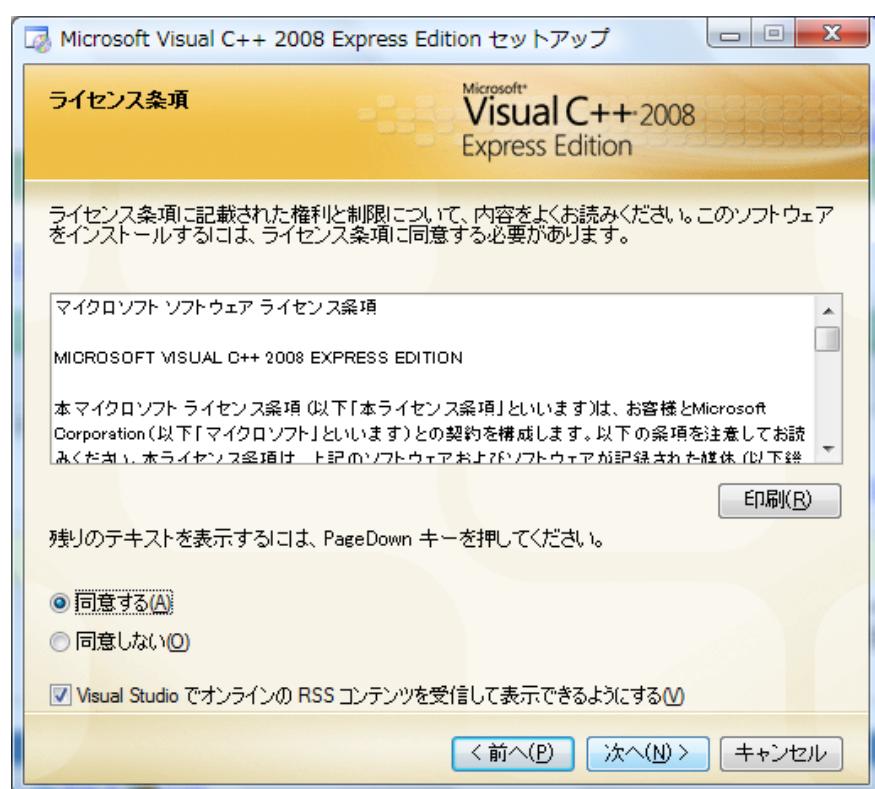


図 6.3: VC2008 インストール ウィザード (step2)

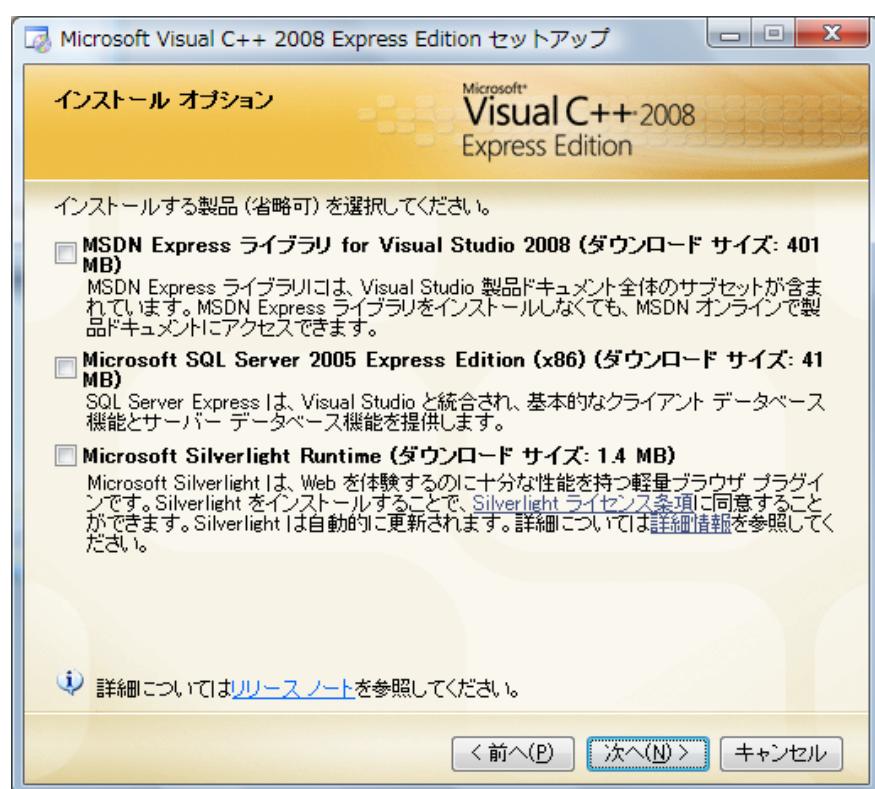


図 6.4: VC2008 インストール ウィザード (step3)

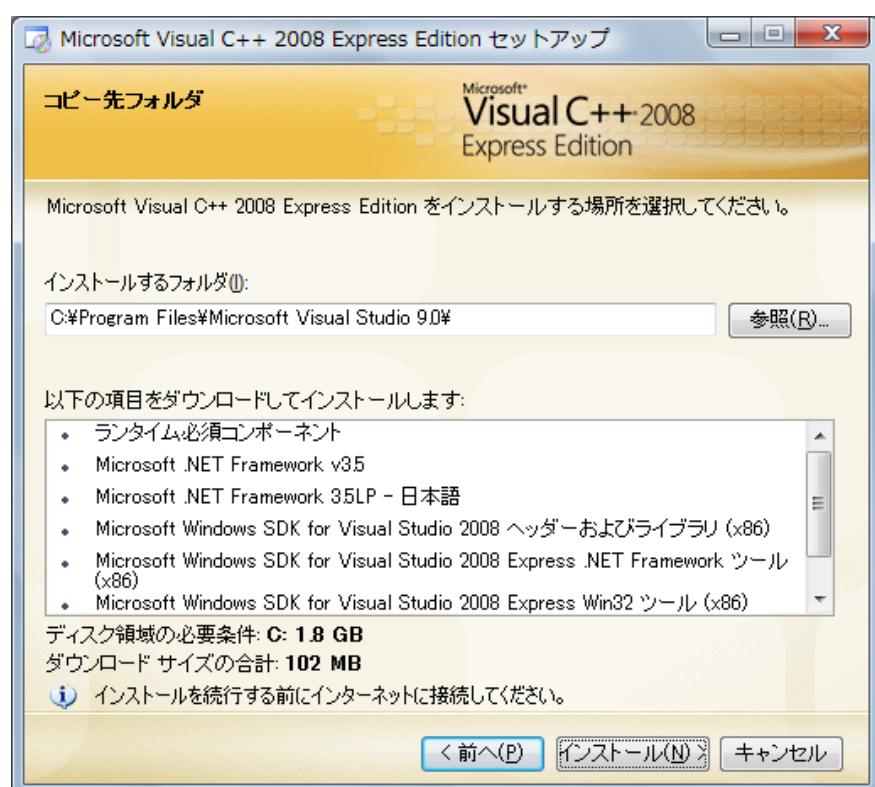


図 6.5: VC2008 インストール ウィザード (step4)

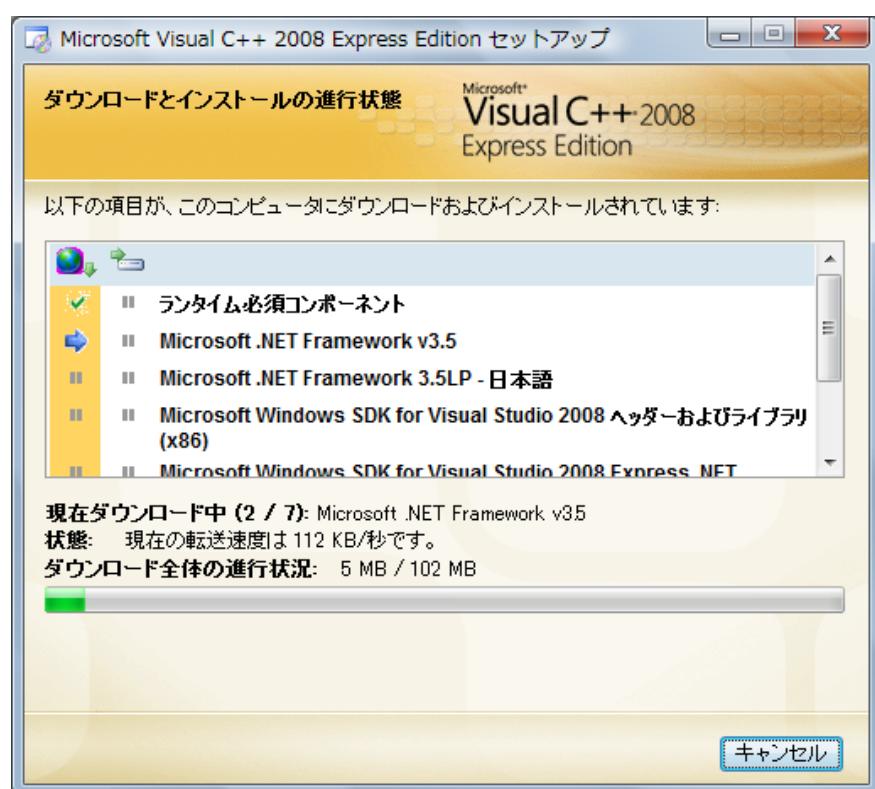


図 6.6: VC2008 インストール ウィザード (step5)

## 6.2 WindowsDDK のセットアップ

VC2008 のセットアップが終わったら、次は DDK(WDK) のセットアップを行います。DDK はその名が示すとおり、ドライバ開発のためのキットであり、Windows の OS 内部とユーザのアプリケーションプログラムの中間に位置するドライバプログラムを開発しやすくするためのヘッダやライブラリが含まれています。WiiRemote を使ったプログラミングでは主に Bluetooth 経由の通信のためにこのヘッダやライブラリを必要とします（すべてのファイルが必要になるわけではありません、詳細は後述）。

まずこちらのマイクロソフトの DDK のホームページを訪問し、「Windows Server 2003 SP1 DDK」の ISO ファイルを入手してください。

「Windows Server 2003 SP1 DDK」とありますが、WindowsXP 等でも利用できます。「Windows XP SP1 DDK」以前の DDK(NT,98,2000など)は既にサポートが終了していますので持っていても使用しない方が良いでしょう。現状の主力 OS である、Windows Vista、Windows Server 2003、Windows XP、そして Windows 2000 上で動作するドライバをビルドするには、この Windows Server 2003 SP1 DDK に含まれる「Windows 2000 向けビルド環境」を使用する必要があります（この WDK 内の Windows 2000 用ビルド環境には、更新されたヘッダーとライブラリが含まれています）。

まず、ダウンロードした ISO ファイルを使って CD-ROM を作成します。ISO ファイルをドライブとしてマウントできる仮想 CD のようなソフトウェアがあればそちらでもかまいません。

作成した CD-ROM を挿入し「setup.exe」を実行します。

エンドユーザライセンス承諾書 (EULA) を確認します。

インストール先は本書ではデフォルトの「C:\WINDDK\3790.1830」とします。

ここでインストールするファイルを選択します。デフォルトのまま、もしくはすべてを選択しても良いのですが、ヘッダファイルのような小さなテキストファイルが 700MB 以上ありますのでインストールに軽く 1 時間ぐらいかかるかもしれません。

大量のファイルをインストールすることに特に抵抗がない方はすべてにチェックを入れても良いでしょう。余計なファイルは必要ない、という方は最低限「Build Environment」の該当するプラットフォームにチェックが入っていればよいでしょう（Windows XP Headers, X86 Libraries）。具体的には「C:\WINDDK\3790.1830\inc\wxp」にある「hidpi.h, hidusage.h, setupapi.h」というヘッダファイルと、「C:\WINDDK\3790.1830\lib\wxp\i386\」

### Windows Server 2003 SP1 DDK

<http://www.microsoft.com/japan/whdc/DevTools/ddk/default.mspx>



図 6.7: Windows Driver Development Kit のインストール



図 6.8: EULA の確認

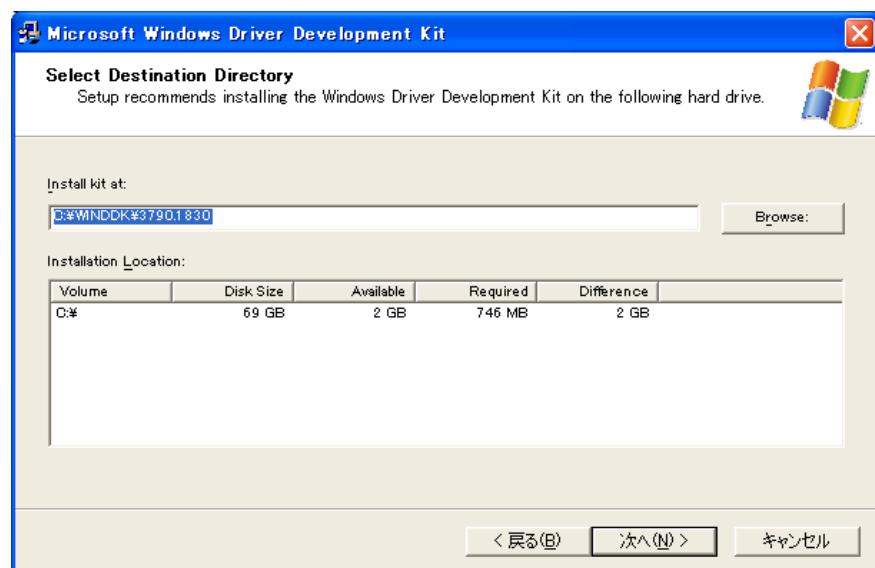


図 6.9: DDK インストール先の指定



図 6.10: DDK インストールするファイルの選択

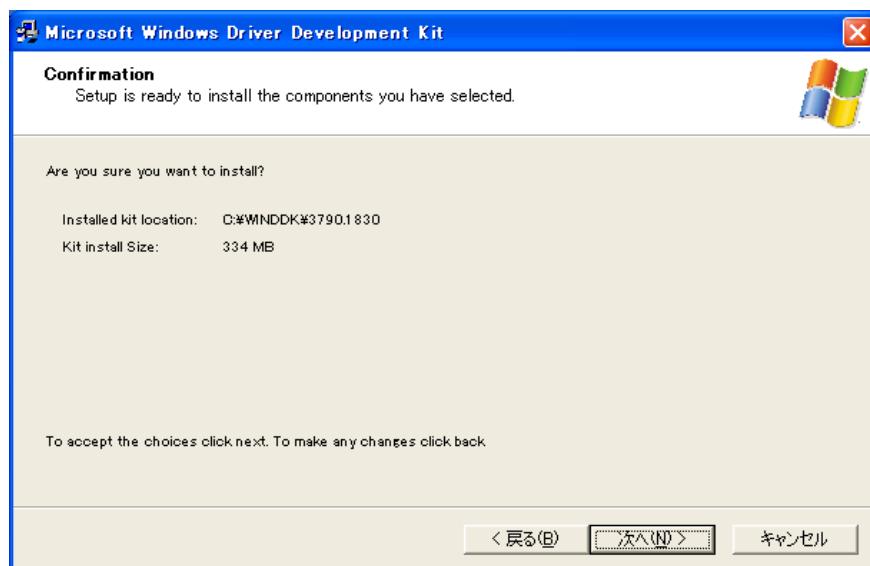


図 6.11: DDK インストール、1 時間強必要。

にある「hid.lib, setupapi.lib」というライブラリファイルが必要です。その他のファイル、ツール類はインストールして試してみても良いですが、本書では扱いません。

DDK のインストールが終わったら、さっそく次の章でリビルドを試してみます。

### 6.2.1 コラム：ISO ファイルのマウント

本章で紹介したように、マイクロソフトのホームページからダウンロードできる ISO ファイルは ISO イメージファイルとよばれる形式で、本来 CD-ROM や DVD-ROM で流通するべきデータディスクを復元・作成するための形式です。MSDN サブスクリプションなどでマイクロソフトのオンラインサポートを契約しているユーザには馴染みがある配布方法で、マイクロソフトが提供する CD を作成するソフトウェアや一般的の CD ライティングソフトウェア

を使って目的とする CD-R を作成することができます。しかし今回の DDK は 1 回しか使いませんし、一度インストールてしまえば、あとはファイルコピーでも問題ないので、そのためにわざわざ CD-R を焼くのは勿体ない感じがします。それに CD/DVD ドライブを内蔵していないノート PC を使っている場合は一度別の PC にインストールした後に必要なファイルをコピーすることになり面倒です。このような場合に便利なのは「仮想 CD」と呼ばれるソフトウェアを利用することです。多くの仮想 CD ソフトウェアが ISO ファイルをサポートしており、CD-ROM ドライブやディスクがなくても、まるで実際に CD-ROM ドライブがあるような環境を利用できます。

以前は仮想 CD といえば「Daemon Tools」というフリーウェアが有名でしたが、アドウェア(無料で利用できる代わりに広告が表示される)になり、その後、広告を表示するソフトウェアがスパイウェア(ユーザが気がつかないうちに個人情報などを第三者に送るソフトウェア)であるという報告がされて以来、後継の「Alcohol 52%」がよく使われるようになってきています。こちらもフリー版はアドウェアなのですが、インストールされる Internet Explorer のツールバー「free-downloads.net Toolbar」はコントロールパネルの「プログラムの追加と削除」でアンインストールできるようです。

### 6.2.2 コラム：VC 混在環境を解決する Version Selector

もし、既に Visual C++ や Visual Studio の過去のバージョン(.NET2003/2005など)をお使いで、かつ 2008 年時点の最新版である Visual C++ 2008 Express(以下 VC2008)を試してみたいの読者は、ここでちょっと回り道をして試してみることをおすすめします。VC2008 は Version Selector という機能があり、異なるバージョンの Visual Studio 製品の混在を可能にします。各バージョンにおいてソリューションファイルは拡張子「.sln」と変わりませんが、自動的にこの拡張子「.sln」に関連づけられたアプリケーションである「Version Selector」がファイル内部のバージョン記述を自動的に読み取り、ファイルがダブルクリックされたときは適切なバージョンの VC を起動します。この機能のおかげで複数のバージョンの開発環境を安全してインストールできるようになります。

#### Alcohol 52%(フリー版)のホームページ

<http://trial.alcohol-soft.com/en/>

# 第7章 API紹介

## 7.1 tiny\_hid\_dll

### 7.1.1 7.1.4 (temp)Wii Mice Test1

### 7.1.2 7.1.5 (temp)Wiimote fall detect Test1

「tiny\hid\ dll」は加古氏 (<http://kako.com/>) による、Bluetooth 経由の HID へのアクセスを容易にする API です。2006 年末、世界でもかなり早い段階で公開された API は非常にシンプルで、OpenHidHandle , ReadReport , WriteReport , CloseHidHandle という 4 つのローレベル関数のみで構成されていますが、加古氏のホームページにおいて、いくつかの基本機能 (ボタン、加速度センサ、赤外線センサ等) がソースコードとともに公開されており、WiiRemote を学習する上で非常に勉強になります。

また tiny\hid\ dll は比較的多くの Bluetooth スタックをサポートしています。筆者の環境では BlueSoleil をはじめ、Toshiba の Bluetooth スタックなど、様々な環境で安定した動作が確認できました。

さて、本格的な WiiRemote を使った開発をはじめるにあたり、加古氏に敬意を込めつつ、ホームページで公開されているいくつかのプログラムについて解説していきたいと思います。

この加古氏のページで紹介されているプログラムは、特徴として Win32 コマンドラインや SDL(Simple DirectMedia Layer) を使った一見地味なものが多いのですが、コードが短く整理されており理解しやすく、さらに Visual Studio の IDE を使用していません。具体的には VC の含まれているコマンドラインコンパイラを使用したバッチファイルを使ってコンパイル、リンクを行います。

#### 加古氏の HP 「小ネタ 2006-019」

[http://www.kako.com/neta/2006-019/2006-019.html}](http://www.kako.com/neta/2006-019/2006-019.html)

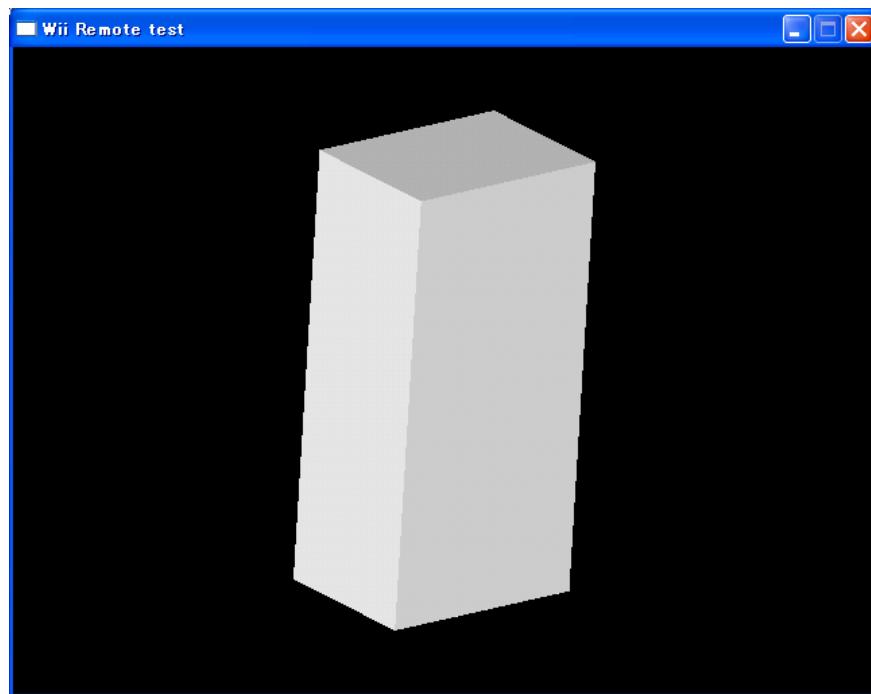


図 7.1: Wii Remote Test1 のスクリーンショット

### 7.1.3 Wii Remote Test1 の実行

まず最初のプログラムである「Wii Remote Test1」について解説していきたいと思います。上記のホームページから「wii\_test1.zip」とソースコード「wii\_test1\_src.zip」をダウンロードして好きなディレクトリに展開ください。

Bluetooth スタックから WiiRemote を 1 個接続して、WiiRemote を机など水平な場所に置いてから、実行ファイル「wii\_test1.exe」を起動してみましょう。

うまく起動できた場合、WiiRemote のバイブルータがブルッと振るえ、すべての LED が消灯した後、ウィンドウに 3D で WiiRemote 調の白い直方体が表示されます。WiiRemote を傾けたりすると、現在の WiiRemote の姿勢に近い傾きで高速な 3D でレンダリングします。A ボタンを押すと傾斜がリセットされます。キーボードの ESC か WiiRemote の HOME ボタンを押すことで終了します。

## トラブルシューティング

このプログラム「wii\_test1」はシンプルですが、WiiRemote プログラミングに求められる多くの基本が集約されています。もしうまく起動できなかつた場合は、この先のプログラミングにおいても問題が起きる可能性がありますので、ここでしっかりと確認しておきましょう。

「Wii-remote cannot open」というエラーが発生する これは最もありえるケースです。原因としては Bluetooth スタックにおいて WiiRemote が 1 個もつながっていない状態が考えられます。接続を確認しましょう。BlueSoleil が評価版の場合 5MB の転送量を超えており、WiiRemote としばらく通信がなかったために WiiRemote がスリープしている、などの場合も可能性としてあります。

WiiRemote が接続されているのに「Wii-remote cannot open」エラーが発生する 残念ながらお使いの Bluetooth スタックは tiny\_hid.dll での接続方法では接続できない可能性があります。WiiYourself! や WiinRemote など他のツールや API を試してみてください。それでも上手く接続できない場合は別の Bluetooth アダプタの購入を検討することをおすすめいたします。

直方体は表示されるが動かない Bluetooth スタックを確認してみてください。  
もし過去に WiiRemote を複数接続したことがあれば、その接続プロファイル（多くの場合アイコンがあります）を全部削除して、もういちど WiiRemote のペアリングからやり直してみてください。

その他の不具合 次節のソース解説を読むと解ることがあるかもしれません！

### 7.1.4 wii\_test1.cpp のリビルド

Wii Remote Test1 のソースコードのパッケージを観察してみると、VC.NET2003 をコマンドラインで利用する環境で開発されているようです。build.bat をダブルクリックするとリビルドすることができます。たしかにこのようなシンプルなコマンドラインが中心のプログラムなら、大がかりな Visual Studio の IDE を使う必要はないかもしれません（UNIX の make のような感覚で使えるので筆者は気に入っています）。本書ではここまで VC2008（出版時点では最新の開発環境）をセットアップしてきておりますので、使用している PC に VC2008 がデフォルトのままインストールされているという前提で、build.bat を以下のように修正して、新しいバッチファイル「build\_VC9.bat」を作成します。

```

@echo "----- 環境変数の設定 -----"
rem set environment for VC2008 (VC9)
call "C:\Program Files\Microsoft Visual Studio \
    9.0\VC\bin\vcvars32.bat"

@echo "----- C++ソースをコンパイルします -----"
cl wii_test1.cpp /D HAVE_OPENGL /MD /link user32.lib sdl\sdl.lib \
    sdl\sdlmain.lib
    opengl32.lib tiny_hid_dll.lib /SUBSYSTEM:WINDOWS
pause

```

図 7.2: build\_VC9.bat

```

C:\WINDOWS\system32\cmd.exe
"----- 環境変数の設定 -----"
C:\shirai\WiiDev\wii_test1_src>rem set environment for VC2008 (VC9)
C:\shirai\WiiDev\wii_test1_src>call "C:\Program Files\Microsoft Visual Studio 9.0\VC\bin\vcvars32.bat"
C:\shirai\WiiDev\wii_test1_src>"C:\Program Files\Microsoft Visual Studio 9.0\Common7\Tools\vsvars32.bat"
Setting environment for using Microsoft Visual Studio 2008 x86 tools.
"----- C++ソースをコンパイルします -----"

C:\shirai\WiiDev\wii_test1_src>cl wii_test1.cpp /D HAVE_OPENGL /MD /link user32.lib sdl\sdl.lib /sdlmain.lib opengl32.lib tiny_hid_dll.lib /SUBSYSTEM:WINDOWS
Microsoft(R) 32-bit C/C++ Optimizing Compiler Version 15.00.21022.08 for 80x86
Copyright (C) Microsoft Corporation. All rights reserved.

wii_test1.cpp
Microsoft (R) Incremental Linker Version 9.00.21022.08
Copyright (C) Microsoft Corporation. All rights reserved.

/out:wii_test1.exe
user32.lib
sdl\sdl.lib
sdl\sdlmain.lib
opengl32.lib
tiny_hid_dll.lib
/SUBSYSTEM:WINDOWS
wii_test1.obj

C:\shirai\WiiDev\wii_test1_src>pause
続行するには何かキーを押してください . . .

```

図 7.3: build\_vc9.bat によるコマンドラインリビルド

編集したバッチファイルを保存してダブルクリックするとコマンドプロンプト画面が表示され、コンパイルとリンクが実行されます。特にソースコード「wii\_test1.cpp」を改変していないければ、無事に「wii\_test1.obj」「wii\_test1.exe」「wii\_test1.exe.manifest」が生成されているはずです。このディレクトリには必要なDLLがそろっていないので、このままでは実行できません。既にダウンロードしてきたビルド済みの「wii\_test1」フォルダにある「SDL.dll(ver.1.2.11.0)」と「tiny\_hid\_dll.dll」をコピーってきてから「wii\_test1.exe」をダブルクリックしてみましょう。無事に実行されましたか？

### 7.1.5 wii\_test1.cpp の流れ

(コラム) 加速度センサの値から直方体の角度に変換する

(コラム) マイクロ秒以上の精度を持った Sleep2() 関数

## 7.2 7.2 WiiYourself!

### 7.2.1 7.2.1 Readme とライセンスの参考訳

【Readme.txt】

【License.txt】

### 7.2.2 7.2.2 WiiYourself!付属デモのテストとリビルド

## 7.3 7.3 その他の API

### 7.3.1 7.3.1 Wiim

懐かしゲーム「Wiim Landar」

Wiim Landar の改良

### 7.3.2 7.3.2 WiimoteLib

### 7.3.3 7.3.3 API を使用しない接続方法

## 第8章 WiiRemoteで学ぶ、物理・数学・プログラミング

- 8.1 8.1 最小の WiiRemote プログラム
- 8.2 8.2 WiiRemote テルミン
- 8.3 8.3 PowerPoint コントローラー
- 8.4 8.4 姿勢検出
- 8.5 8.5 赤外線 4 点検出
- 8.6 8.6 赤外線 LED を使った測量
  - 8.6.1 8.6.1 より広範囲を測定する場合
- 8.7 8.7 バイブレータと LED
- 8.8 8.8 ポーリング受信と動作周波数の測定
- 8.9 8.9 認識の基本
- 8.10 8.10 Socket 通信
- 8.11 8.11 重力加速度を見る
- 8.12 8.12 振りぬく力・パンチ力測定
- 8.13 8.13 モーション解析

# 第9章 3DCG 応用編

## 9.1 9.1 3DCG + 物理エンジンとの組み合わせ (高橋さん担当分)

9.1.1 9.1.1 Direct3D9 の初期化 (固定機能?)

9.1.2 9.1.2 Bullet の準備と初期化

9.1.3 9.1.3 球やボックスのような基本形状の利用

9.1.4 9.1.4 「pick」3D オブジェクトを持ち上げて遊ぶ

9.1.5 9.1.5 「だるま落とし」

9.1.6 9.1.6 「ヌンチャク」ジョイントを使った例

# 第10章 プロジェクトサンプル

10.1 10.1 レースゲームへの応用

10.2 10.2 赤外線マーカーを使ったモーションキャプチャ

10.3 10.3 お絵かきソフトへの応用

10.4 10.4 「振る」の認識

10.5 10.5 剣術アクションへの応用

10.6 10.6 フリースロー

# 第11章 Wii Flash を使ってみよう

11.1 11.1 環境設定 (CS3のセットアップ DLとインストール)

11.2 11.2 WiiFlashのダウンロードとセットアップ

11.3 11.3 WiiFlash の基本機能

11.4 11.4 サンプルのテストでみる基本的な使い方

11.5 11.5 WiiFlash プロジェクトサンプル

11.5.1 11.5.1 なりきりライトセーバー (原さん)

11.5.2 11.5.2 おえかき (尾崎さん)

11.5.3 11.5.3 DropShadow(尾崎さん)

11.5.4 11.5.4 ChunChun(原さん)

# 第12章 未整理ネタ倉庫

## 12.1 コラム：WiiFit を作った男たち

%{Wii Fit 開発インタビュー } %http://wii.com/jp/articles/wii-fit/crv/vol2/index.html 「時雨殿」の開発者だというところが面白いですね。

## 12.2 12.0.5 赤外線反射式インターフェース

CMU の研究者による「マイノリティ・リポート」式ユーザインタフェースの作り方。

## 12.3 (大内さん提供のネタ)

また、センサーバですが、任天堂から単体でも出ているらしいので、その辺の情報も紹介したいです。今は販売していないんでしょうか? - 任天堂、部品販売で「センサーバー」など取り扱い開始 http://www.inside-games.jp/news/200/20068.html

白井追記部品販売コーナーで買えます。普通為替もしくは定額小為替を使い郵便で購入します。http://www.nintendo.co.jp/n10/parts/index.html  
http://shop.nintendo.co.jp/GoodsList.do?CATEGORY\_ID=sensorbar 1,785円送料 450 円

それと Web を見ていて、以下のページでやっていることとかおもしろかったです。一応、参考までに URL はっておきます。- MIT の研究グループ、Wii リモコンを利用したリズムゲームを披露 http://japan.cnet.com/news/ent/story/0,2000056022,20374627,00.htm

Wii のセンサーバーを応用して、おもしろデバイスを自作してみた http://myhome.cururu.jp/kacom/blog/article/21001739973

## 付録

**.1 索引**

**.2 参考文献**

**.3 登録商標**

Wii\texttrademark Wii は任天堂の登録商標です。