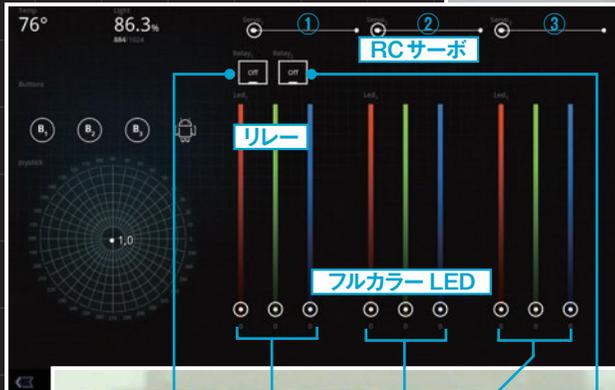
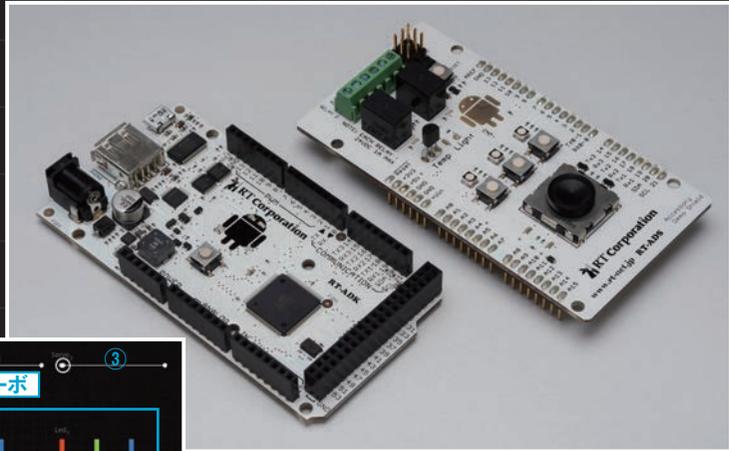
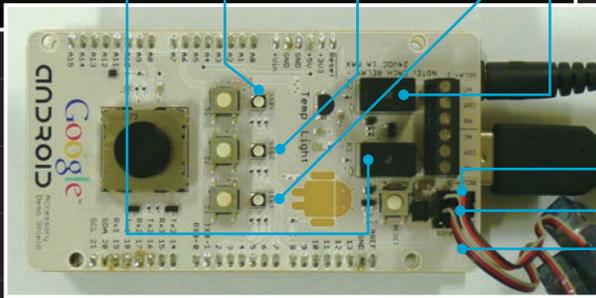




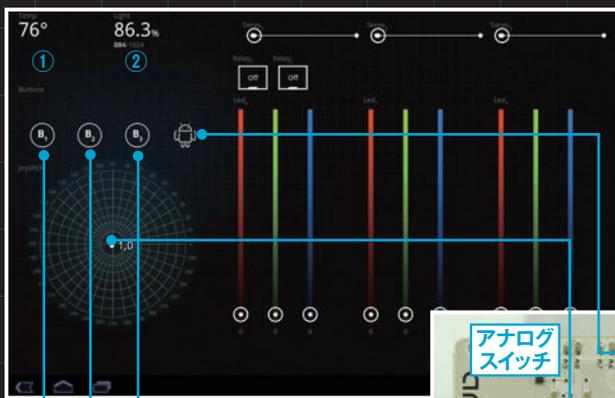
### Android端末から Arduinoを操作する



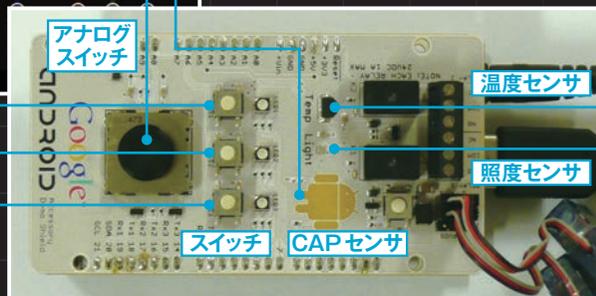
株式会社アルティのADKのリファレンスボード「RT-ADK&RT-ADS」を使ってみよう。このボードには、フルカラーLED、RCサーボなどの出力と、アナログジョイスティック、スイッチ、照度センサなどの入力といった豊富な機器が基板に実装済みのため、Android端末と組み合わせると、すぐにADKが実践できる。



GoogleのArduino用ADKキットに含まれるデモアプリ「DemoKitLaunch」を実行すると、リファレンスボードの接続を促す画面が、ADKのマスコットキャラ（ドロイド君）とともに表示される。



### Arduinoから Android端末を操作する





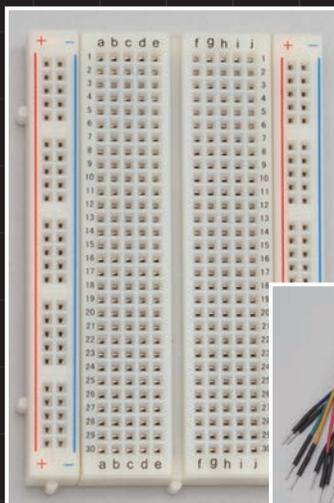
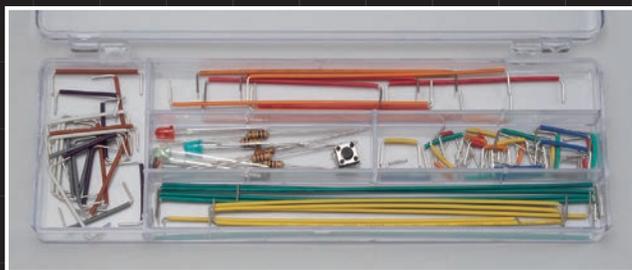
ADKがサポートしている最も手軽なボード「Arduino Uno R3」。電子工作ではよく使われるボードなので、Webなどでもさまざまな情報が入手できる。

### ■USBホストシールド

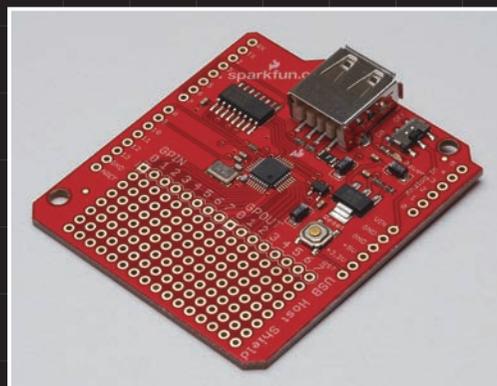
Arduino UnoにUSBホスト機能を追加するための拡張ボード「SparkFun USB Host Shield」。こちらは「Arduino はじめようキット」には入っていないので、別途入手する。また、Arduino Unoと組み合わせて使用するために、ピンソケットのハンダ付けが必要になる。本書では、ハンダ付けがはじめての方にもできるように、Appendixでその方法を写真を豊富に使って紹介している。

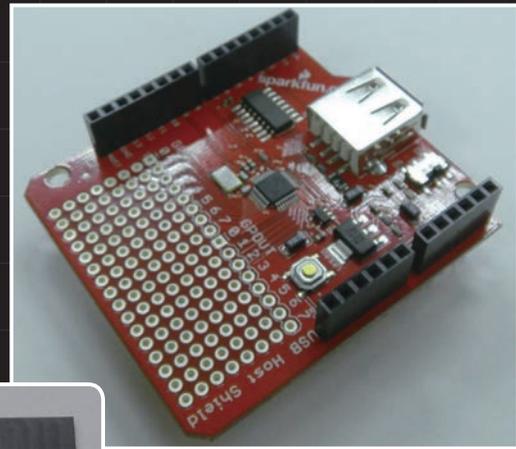
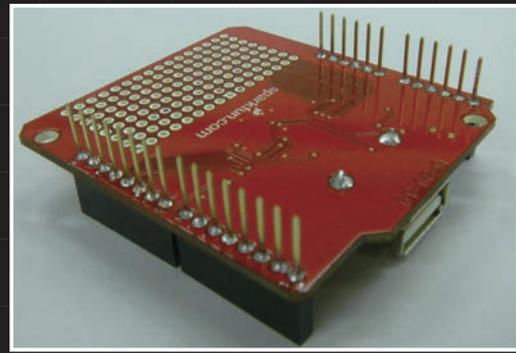
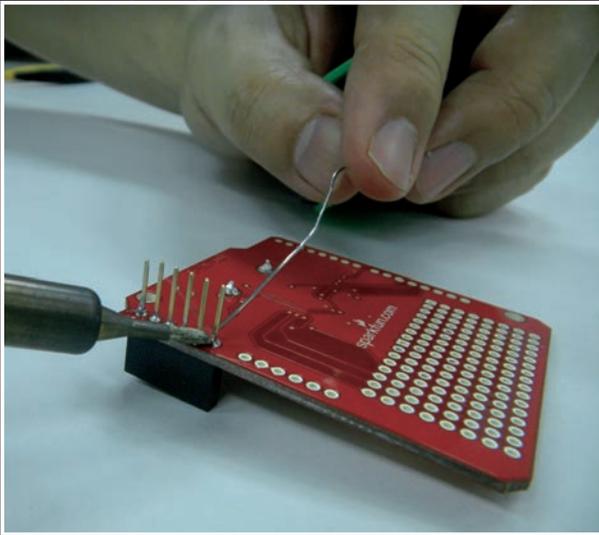
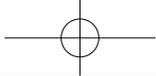
低価格で手軽にADKを楽しむなら、株式会社スイッチサイエンスが販売している「Arduino はじめようキット」がお勧め！3,980円（税込）という低価格で必要なパーツが揃うので、電子工作にはじめてチャレンジする方は、こちらを入手しよう（ただし、USBホストシールドは別売）。

3章では、このキットを使って電子工作の基礎から、1つずつ部品を取り付けながら動作を確認し、Android端末と情報をやりとりするプログラムの作成などをステップアップでいねいに解説している。

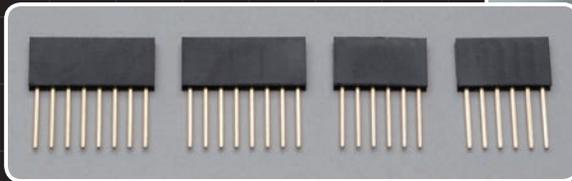


キットには、ブレッドボード、LED、抵抗、スイッチ、ワイヤー、ケーブルなど、電子工作の入門に必要なパーツがセットされている。





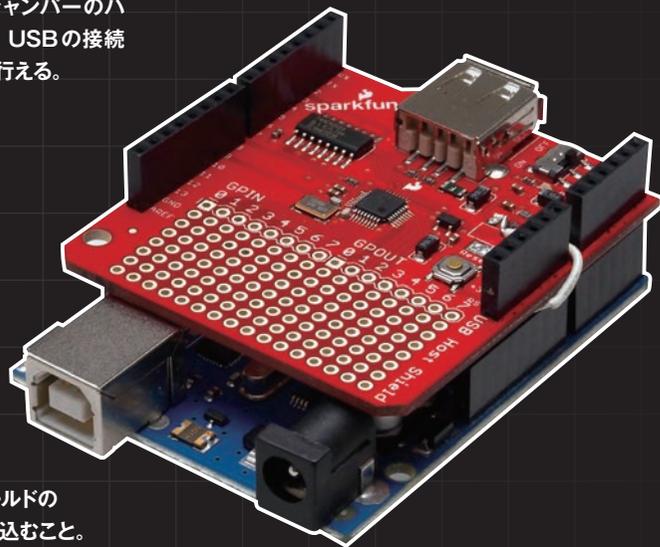
電子工作の最初のハードルが「ハンダ付け」。Appendixでは、必要な道具から、ハンダ付けのコツまでをていねいに解説している。ピンソケットは取り付けが少しずれても大丈夫なので、気軽に取り組んでみよう。所用時間は10分ほど。右のようにすべてのピンのハンダ付けが終われば完成！



Arduinoシールド用ピンソケットのセット。こちらも別売になる。



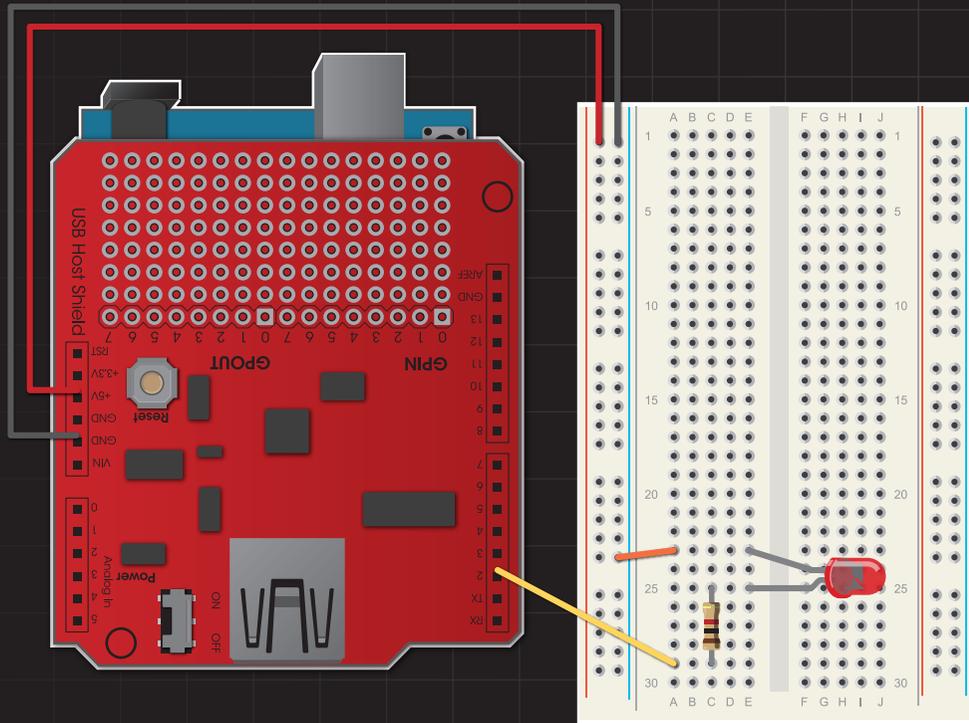
さらに基板上の5Vレギュレータを取り除き、5Vジャンパーのハンダ付けを行うと、USBの接続のみで電源供給が行える。



Arduino Unoとピンソケットのハンダ付けが終わったUSBホストシールドを写真のように組み合わせれば（スタックする）、準備完了。Arduino Unoのソケットの穴の数とUSBホストシールドのピンの数は違うので、D0番ピンのほうにそろえて差し込むこと。

## ■ブレッドボードにパーツを取り付ける

ブレッドボードにパーツを取り付けよう。ブレッドボードを使えば、ハンダ付けをしなくてもパーツを配線できるので、手軽に動作確認が行える。以下は、Androidからデジタル出力を確認するためにLEDと抵抗をブレッドボードに取り付け、Arduino+USBホストシールドと接続した配線図になる。



## ■準備完了!ADKの動作を確認しよう

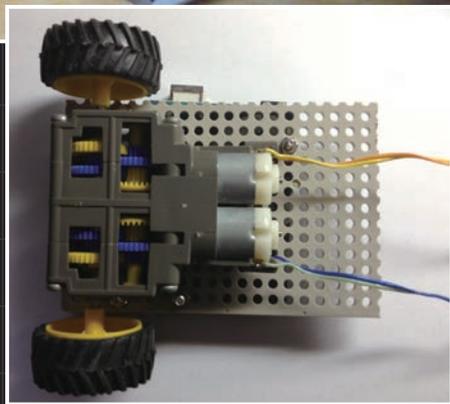
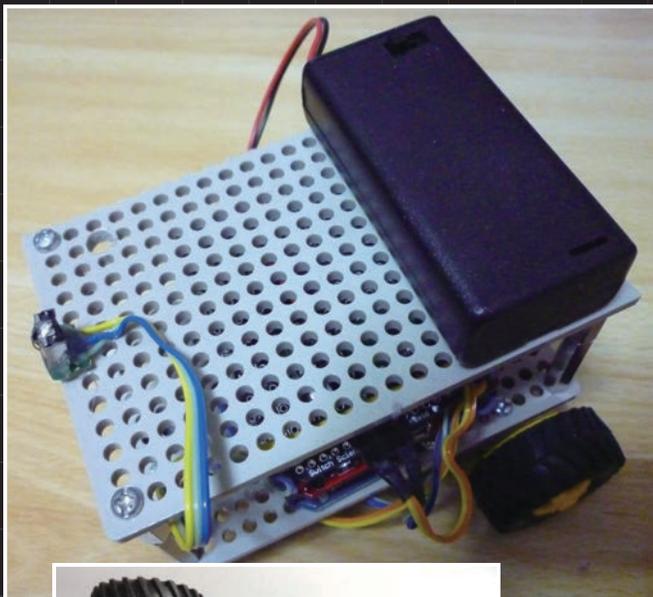
ハードウェアの準備が終わったら、Arduino Unoのスケッチ(プログラム)とAndroid端末のプログラムを用意する。ハードを接続してアプリを実行すると、動作確認が行える。これであなたも、Androidとハードウェアの世界に第一歩を踏み出した!



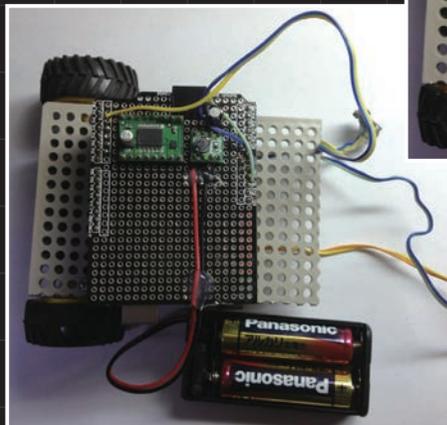
アナログ入力、デジタル入出力を動作確認するためのすべてのパーツを組み込んだ完成例。

## ■Androidから赤外線でロボットを操縦する

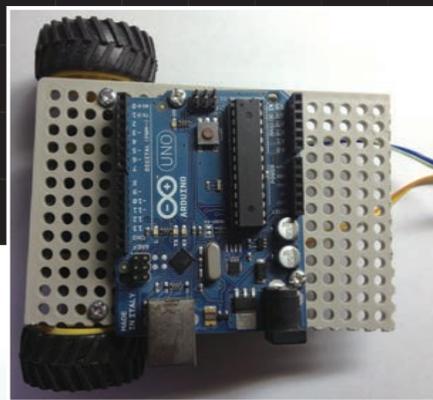
タミヤの「楽しい工作シリーズ」のパーツを使って、オリジナルの車両型ロボットを作成してみた。Android端末から、赤外線で操縦ができる！



ロボットの底面。ギアボックスが取り付けられている。



ユニバーサルプレートにArduinoを設置する。穴の位置の関係で、少し傾いた状態になる。



Arduinoと赤外線リモコン受信モジュールを組み込んだ自作シールドを合体させる。



ADKのさまざまな応用事例も満載！

## ■Androidで温度モニタリング

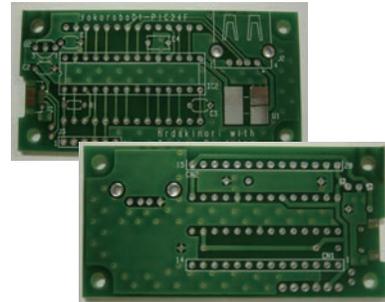
電子工作でよく使われる「PICマイコン」もADKに対応している。PICマイコンで温度センサーの情報を表示する作例を紹介!



### ■マイコン基板の販売

株式会社テクノロードのサイトでは、自作基板を販売中!

<http://techno-road.com/shop.html>

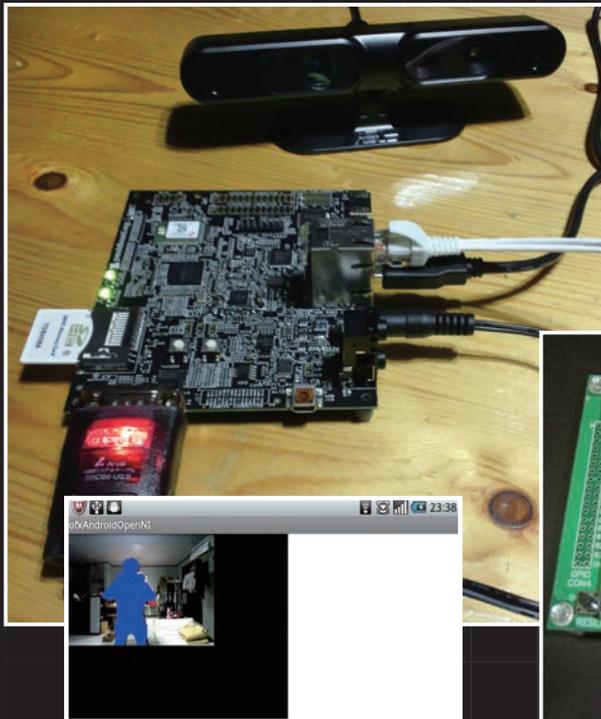


●基板のみ: 500円 (税込)

●部品込み: 2,980円 (税込)

## ■KinectとAndroidの接続

KinectとAndroid端末を接続することで、アプリの可能性がさらに広がる!



## ■FPGAボード+Android

FPGA基板をAndroid端末とMicroBridgeで接続した例



# はじめに

Googleが創り出した「Android」は、スマートフォンやタブレットなど数多くのデバイスで使われています。Androidに対応したアプリケーションやゲームの総数は、2011年の12月で40万本に達しています。Androidの開発者も増え続けており、Androidは主要なプラットフォームとしての地位を確立したといえるでしょう。

一方、スマートフォンやタブレットを考える上でiOSを搭載したiPhoneやiPadを除外することはできません。これらのデバイスは非常に洗練されており、利用者や適用シーンを今後も広げていくことは間違いないでしょう。そのため、AndroidとiOSの両方のプラットフォームで動くアプリケーションを開発することは、多くの開発者が興味を持っており、HTML5などの技術が目目される大きな要因とも言えます。

このような状況のなか、Googleから「ADK」と呼ばれる技術が発表されました。本書を手にとっている方はADKに興味を持っていることでしょう。このADKは、Android端末と外部ハードウェアを接続するための仕様です。この仕様が公開されたことにより、Androidが持つ可能性に大きな注目が集まっています。

Androidは登場した頃から、ロボットや専用機器などのハードウェアを制御するためのプラットフォームとなり得ると言われてきました。このAndroidによるハードウェア制御の領域に、多くのAndroid技術者が目を向けるきっかけを作ったことは、ADKの功績の1つと言えるでしょう。

本書は、Androidからハードウェアを制御する方法を広く知ってもらうため、ハードウェアの基礎となる電子工作から、作成したハードウェアを制御するためのプログラミング方法を解説します。

趣味でAndroidから動くロボットを作ってみたい方や、業務で初めてハードウェア制御を開発することになった方に最適な入門書を目指しました。本書がAndroidとハードウェアの世界に踏み出すための一歩になれば幸いです。

2012年4月 執筆者一同

## ■商標ほか

- GoogleおよびGoogleロゴ、AndroidおよびAndroidロゴ、Google PlayおよびGoogle Playロゴは、Google Inc.の商標または登録商標です。
- AndroidロボットのデザインはGoogleが作成、提供しているコンテンツを複製したもので、クリエイティブコモンズの表示3.0ライセンスに記載の条件に従って使用しています。
- OracleとJavaは、Oracle Corporationおよびその子会社、関連会社の米国およびその他の国における登録商標です。
- Eclipseは、開発ツールプロバイダのオープンコミュニティであるEclipse Foundation, Inc.により構築された開発ツール統合のためのオープンプラットフォームです。
- Wi-FiおよびWi-Fiロゴは、Wi-Fi Allianceの登録商標です。
- Bluetoothは、米国Bluetooth SIG, Inc.の登録商標です。
- PIC、PICkit、MPLAB IDEは、米国Microchip Technology, Inc.の米国およびその他の国における登録商標です。
- チョロQは、株式会社タカラトミーの登録商標です。
- その他、本書に記載されている社名、商品名、製品名、ブランド名、システム名などは、一般に商標または登録商標で、それぞれ帰属者の所有物です。
- 本文中には、®、©、TMは明記していません。

- 本書はソシム株式会社が出版したもので、本書に関する権利、責任はソシム株式会社が保有します。
- 本書に記載されている情報は、2012年4月現在のものであり、URLなどの各種の情報や内容は、ご利用時には変更されている可能性があります。
- 本書の内容は参照用としてのみ使用されるべきものであり、予告なしに変更されることがあります。また、ソシム株式会社がその内容を保証するものではありません。本書の内容に誤りや不正確な記述がある場合も、ソシム株式会社はその一切の責任を負いません。
- 本書に記載されている内容の運用によって、いかなる損害が生じても、ソシム株式会社および著者は責任を負いかねますので、あらかじめご了承ください。
- 本書のいかなる部分についても、ソシム株式会社との書面による事前の同意なしに、電気、機械、複写、録音、その他のいかなる形式や手段によっても、複製、および検索システムへの保存や転送は禁止されています。

# 本書について

## 想定する読者

本書では、Androidアプリの開発経験はあるが、電子工作の経験はない技術者を対象としています。ハードウェアの組み立てや制御が必要となる電子工作は、はじめの一步の敷居が高く、誰もが簡単に試せる状況にはありません。本書では、部品の調達方法から組み立て、Androidアプリからのハードウェア制御の方法を解説して、電子工作の世界へ踏み出す手助けになるように構成しました。あとは、みなさんの新しいものへ挑戦する気合いを用意するだけです。

Androidアプリの開発経験がない方にも、本書は楽しめるようになっていきます。インターネットにはスマートフォンやタブレットで動くAndroidアプリの作り方の解説が溢れており、すばらしい内容の書籍も多数刊行されています。それらを参考にすれば、Androidアプリの開発方法を学びながら、本書で電子工作を学ぶこともできるでしょう。

また開発者ではない方は、1章やAppendixを読んでいただければ、Androidと外部ハードウェアの関係や応用事例を知ることができます。

## 本書の構成

本書の構成は、次のようになっています。なお、各章の執筆者は、巻末の著者紹介を参照してください。2章と3章が基礎編で、読み終えると電子工作でハードウェアを作り、そのハードウェア上で動かすプログラムを作成し、Android上のプログラムから作ったハードウェアを制御するための基本的なスキルが習得できます。

4章以降は、これらの技術を使った応用例となります。みなさんが作りたいと考えている「アプリケーション」や「ハードウェア」を作成する際の参考にしてください。

### 1章 ADK (Open Accessory Development Kit) の概要とハードウェアの制御

Android端末からのハードウェア制御の概要を説明します。この章で本書で取り扱う技術の全体像をつかんでください。

### 2章 ADK ボードを動かそう

基礎編として、Google 公式の「ADK」対応ハードウェアの使い方と、「ADK」を使ったプログラミングの概要を学びます。

### 3章 Arduino UnoとUSB ホストシールドで手軽にADK

ここでは、「ADK」対応のハードウェアを部品から組み立てます。電子工作にはじめての方でもチャレンジできるように、ステップを追って解説しました。1つずつ部品を取り付けながら動作を確認し、

Android端末と情報をやりとりするプログラムを作成していきます。

### 4章 Androidで赤外線通信

3章までの知識を踏まえて、赤外線でチョコQを操作する応用例を解説します。さらに自作のロボットの製作とチョコQの操作で作ったプログラムで、ロボットを制御する方法も紹介します。

### 5章 Androidで温度モニタリング

もう1つの応用例として、温度センサーの情報をAndroid端末で表示する方法を解説します。ここでは、これまでの章で紹介したADK対応のハードウェアである「Arduino」ではなく、「PICマイコン」を使用した事例を紹介します。

### 6章 MicroBridgeの利用

ADKは対応するAndroidのバージョンが限られていますが、この章で紹介するAndroid Debug Bridge (ADB) を利用することで、ほぼすべてのAndroid端末でハードウェアの制御を行うことができます。本章では、その概要と応用例を紹介します。

### Appendix

電子工作では避けて通れないハンダ付けの方法や、さまざまな応用事例など、本文では紹介しきれなかったテーマを紹介します。

## サンプルプログラムのダウンロード

本書に掲載したサンプルプログラムのソースコードなどは、以下のWebページよりダウンロードできます。

#### ●掲載プログラムのダウンロードWeb サイト

<http://www.socym.co.jp/>

※上記サイトより、書籍名で検索してください。

ファイルをダウンロード後、アーカイブファイルを解凍して、適当なフォルダに配置してご利用ください。なお、サンプルプログラムによって、用意するハードウェア、ライブラリなどが異なりますので、サンプルプログラムの実行にあたっては、本文の該当ページを必ずご確認ください。

本書に掲載したサンプルプログラムは、プログラミングの学習のための参照用のものであり、エラー処理は厳密に行っていないなど、実用を保証するものではありません。あらかじめご了承の上、ご利用ください。なお、本書に掲載したサンプルプログラムの著作権は、著者に帰属します。

# 目次

|        |     |
|--------|-----|
| はじめに   | 003 |
| 本書について | 004 |

## Chapter 1 ADK (Open Accessory Development Kit) の概要とハードウェアの制御 019

|                                               |     |
|-----------------------------------------------|-----|
| 1-1 Androidによるハードウェアの制御                       | 020 |
| 1-1-1 Androidを使うことのメリット                       | 020 |
| 1-1-2 Androidと外部機器の接続方法                       | 021 |
| ■モバイルネットワーク/Wi-Fi                             | 022 |
| ■Bluetooth/NFC                                | 023 |
| ■USB                                          | 023 |
| 1-2 ADK (Open Accessory Development Kit)の概要   | 024 |
| 1-2-1 ADK (Open Accessory Development Kit)とは? | 024 |
| 1-2-2 ADK誕生の経緯                                | 024 |
| 1-2-3 ホストモードとアクセサリモード                         | 025 |
| 1-2-4 USB HostとUSB Accessory                  | 025 |
| 1-2-5 ADKが使えるボードの一覧                           | 026 |
| 1-2-6 ADKの可能性                                 | 026 |
| 1-3 ADKによる開発の概要                               | 028 |
| 1-3-1 開発環境の構成                                 | 028 |
| ■手順1: 外部機器プログラムの開発                            | 029 |
| ■手順2: Androidアプリの開発                           | 029 |
| ■手順3: 稼働確認                                    | 030 |
| 1-3-2 Androidの開発環境                            | 030 |
| 1-3-3 外部機器の開発環境                               | 032 |

## Chapter 2 ADK ボードを動かそう 035

|                        |     |
|------------------------|-----|
| 2-1 ADKボードの概要          | 036 |
| 2-1-1 ADKボードとは         | 036 |
| 2-1-2 ADKボードの入手        | 036 |
| 2-1-3 ハードウェア機器と開発環境の関係 | 037 |

|                                      |     |
|--------------------------------------|-----|
| 2-2 ADKボード(Arduino)の開発環境             | 038 |
| 2-2-1 開発環境のセットアップ                    | 038 |
| 2-2-2 Arduino開発環境のダウンロード             | 038 |
| 2-2-3 CapSense libraryのダウンロードとインストール | 039 |
| 2-2-4 The ADK packageのダウンロードとインストール  | 040 |
| 2-2-5 Arduinoへのファームウェアの書き込み          | 041 |

|                                 |     |
|---------------------------------|-----|
| 2-3 Android端末の開発環境              | 044 |
| 2-3-1 Android 2.3.4用のAPIのインストール | 044 |
| 2-3-2 ADKのサンプルプロジェクトのビルドと実行     | 045 |
| 2-3-3 デモアプリを操作する                | 047 |
| 2-3-4 各機器の接続方法のまとめ              | 049 |
| 2-3-5 Wi-Fi経由でのデバッグ             | 050 |

## 2-4 ADKのシンプルアプリケーション 052

|                                                |     |
|------------------------------------------------|-----|
| 2-4-1 ADKシンプルアプリの開発方針                          | 052 |
| 2-4-2 Android端末のプログラムの概要                       | 052 |
| 2-4-3 AndroidManifest.xml                      | 053 |
| (1) USB Accessoryの使用を宣言                        | 054 |
| (2) 起動アクティビティ                                  | 054 |
| (3) USB接続されたときに起動するアクティビティ                     | 054 |
| 2-4-4 accessory_filter.xml                     | 055 |
| 2-4-5 main.xml                                 | 055 |
| 2-4-6 no_device.xml                            | 056 |
| 2-4-7 UsbAccessoryActivity.java                | 058 |
| (1) アプリ非起動時に、USBが接続されると自動的に立ち上がるアクティビティ        | 058 |
| (2) DemoKitActivityを実行する                       | 058 |
| 2-4-8 DemoKitActivity.java                     | 059 |
| (1) インテントアクション名をAdkBaseActivityに通知             | 063 |
| (2) アクティビティ起動時の処理                              | 064 |
| (3) ADKボードが接続されたときの処理                          | 064 |
| (4) ADKボードが切断されたときの処理                          | 064 |
| (5) ADK通信受信処理1/2(受信スレッド)                       | 064 |
| (6) ADK通信受信処理2/2(GUI用のハンドラ)                    | 065 |
| (7) ADK通信送信関数sendCommand()                     | 065 |
| (8) USBが接続しているときのLayout表示                      | 065 |
| (9) USBが接続していないときのLayout表示                     | 066 |
| 2-4-9 AdkBaseActivity.java                     | 066 |
| (1) ACTION_USB_PERMISSIONの定義                   | 070 |
| (2) USB接続状態を監視するBroadcastReceiverとmUsbReceiver | 070 |
| (3) アプリ起動時の処理onCreate()                        | 071 |
| (4) アプリ起動時の処理onResume()                        | 072 |
| (5) 画面の縦横向きなどが変化したときに現在の状態を保持                  | 072 |
| (6) アクセサリ開始処理                                  | 072 |
| (7) アクセサリ終了処理                                  | 072 |
| (8) Android←ADKボードの受信                          | 072 |
| (9) Android→ADK Boardへ送信                       | 073 |

|            |                                  |     |
|------------|----------------------------------|-----|
| <b>2-5</b> | <b>ADKボード(Arduino)のソースコードの解説</b> | 074 |
| 2-5-1      | Arduinoとは                        | 074 |
| 2-5-2      | setup()関数                        | 075 |
|            | (1) デバッグ用USB通信の初期化               | 076 |
|            | (2) IOの初期化                       | 076 |
|            | (3) ADKボードの通信開始                  | 076 |
| 2-5-3      | loop()関数                         | 077 |
|            | (1) 接続確認                         | 080 |
|            | (2) 受信処理(Android→Arduino)        | 080 |
|            | (3) 受信内容によって処理を行う                | 080 |
|            | (4) 送信処理(Android←Arduino)        | 081 |

Chapter 3 **Arduino UnoとUSBホストシールドで手軽にADK** ..... 087

|            |                                                    |     |
|------------|----------------------------------------------------|-----|
| <b>3-1</b> | <b>低価格でADKを楽しむための機器と概要</b>                         | 088 |
| 3-1-1      | 本章で取り上げる機器                                         | 088 |
| 3-1-2      | Arduino Unoとは                                      | 089 |
| 3-1-3      | SparkFun USB Host Shield(USBホストシールド)とは             | 089 |
| <b>3-2</b> | <b>USBホストシールドを使うための準備</b>                          | 091 |
| 3-2-1      | ピンソケットのハンダ付けに必要な部材                                 | 091 |
| 3-2-2      | ピンソケットのハンダ付け                                       | 093 |
| <b>3-3</b> | <b>Arduino Uno+USBホストシールドとRT-ADK&amp;RT-SDKの違い</b> | 094 |
| 3-3-1      | 仕様の違い                                              | 094 |
| 3-3-2      | メリットと制約                                            | 094 |
| <b>3-4</b> | <b>開発環境の構築</b>                                     | 096 |
| 3-4-1      | 開発プラットフォームはArduino 1.0を利用                          | 096 |
| 3-4-2      | Arduino 1.0のダウンロード                                 | 096 |
| 3-4-3      | Arduino 1.0のインストール                                 | 098 |
| <b>3-5</b> | <b>まずはArduinoを使ってみよう</b>                           | 103 |
| 3-5-1      | ArduinoだけでLEDを点滅させる                                | 103 |
| 3-5-2      | Arduino単体での動作確認の手順                                 | 103 |
| 3-5-3      | LEDを取り付けて動作確認                                      | 108 |
| 3-5-4      | ArduinoにUSBホストシールドをつないでLチカ                         | 110 |
| <b>3-6</b> | <b>USBホストシールド用ライブラリのインストールと動作確認</b>                | 112 |
| 3-6-1      | USBホストシールド用ライブラリのダウンロードとインストール                     | 112 |
| 3-6-2      | USBホストシールド用ライブラリの動作確認                              | 114 |
|            | ■ADK接続が可能かどうかを確認するスケッチ                             | 115 |
|            | ■ソースコードの解説                                         | 115 |
|            | (1) ADKを利用するためのライブラリの読み込み                          | 115 |

|                      |     |
|----------------------|-----|
| (2) 外部インテントの指定       | 116 |
| (3) Androidとの接続      | 116 |
| (4) データのやりとり         | 116 |
| (5) ADKが接続していない場合の処理 | 116 |
| (6) loopの繰り返し間隔      | 116 |
| ■スケッチのインストール         | 116 |
| ■Android端末を接続して動作確認  | 117 |

**3-7 イチからはじめるADK** ..... 119

|              |                                                              |     |
|--------------|--------------------------------------------------------------|-----|
| <b>3-7-1</b> | <b>デジタルとアナログの入出力を行うプログラムの作成</b>                              | 119 |
| (1)          | ブレッドボード上にパーツを配置する                                            | 120 |
| (2)          | Arduino 1.0でスケッチを作成し、Arduino Uno+USBホストシールドにアップロード           | 120 |
| (3)          | Arduino Uno+USBホストシールドにAndroidを接続し、外部インテントが起動するかどうかをダイアログを確認 | 120 |
| (4)          | Androidプロジェクトを作成し、Androidにインストール                             | 120 |
| (5)          | Arduino Uno+USBホストシールドとAndroidを接続し、動作確認                      | 120 |
| <b>3-7-2</b> | <b>入出力に必要なパーツ</b>                                            | 120 |
| <b>3-7-3</b> | <b>ブレッドボードの配線</b>                                            | 121 |
|              | ■ブレッドボードの構造                                                  | 121 |
|              | ■ジャンパー線による配線                                                 | 122 |

**3-8 Androidからデジタル出力(Lチカ)** ..... 126

|              |                                          |     |
|--------------|------------------------------------------|-----|
| <b>3-8-1</b> | <b>デジタル出力(Lチカ)の概要</b>                    | 126 |
| <b>3-8-2</b> | <b>赤色LEDをブレッドボードに取り付ける</b>               | 127 |
| <b>3-8-3</b> | <b>Arduinoのスケッチの解説</b>                   | 128 |
| (1)          | LED用のピンの指定                               | 129 |
| (2)          | Arduinoから呼び出すインテントの指定                    | 129 |
| (3)          | 出力ポートの指定                                 | 129 |
| (4)          | ADKからのデータの読み込み                           | 130 |
| (5)          | データの処理                                   | 130 |
| (6)          | LEDへのデジタル出力                              | 130 |
| <b>3-8-4</b> | <b>Androidプロジェクトの作成</b>                  | 131 |
| <b>3-8-5</b> | <b>AndroidManifest.xmlの作成</b>            | 132 |
| (1)          | USBアクセサリのライブラリの使用を指定                     | 133 |
| (2)          | USB接続されたときに起動する外部インテントを指定                | 134 |
| (3)          | USB接続された際の外部インテント動作のフィルタ設定               | 134 |
| <b>3-8-6</b> | <b>accessory_filter.xmlの作成</b>           | 135 |
| <b>3-8-7</b> | <b>レイアウトXMLの作成</b>                       | 137 |
| (1)          | 画面の中央に配置                                 | 138 |
| (2)          | トグルボタンの記述                                | 138 |
| <b>3-8-8</b> | <b>Javaプログラムの作成</b>                      | 139 |
|              | ■AdkLedOnOffProjActivityクラス              | 139 |
|              | ■ソースコードの解説                               | 143 |
| (1)          | レイアウトされたトグルボタン用インスタンス                    | 143 |
| (2)          | レイアウトされたトグルボタンとインスタンスmToggleButton1の結び付け | 143 |
| (3)          | トグルボタンを押したときの処理                          | 143 |
| (4)          | トグルボタンの押下に応じたコマンドのセット                    | 144 |

|                                                      |            |
|------------------------------------------------------|------------|
| (5) 入出力用のストリームを確保 (今回は出力のみ) .....                    | 144        |
| (6) Android アプリから Arduino (USB ホスト) への出力データの準備 ..... | 144        |
| (7) 出力ストリームに buffer [] 配列データを書き込む .....              | 144        |
| <b>3-8-9 デジタル出力 (Lチカ) の動作確認 .....</b>                | <b>144</b> |
| <b>3-9 Android からアナログ出力 (LED の PWM 出力) .....</b>     | <b>149</b> |
| <b>3-9-1 アナログ出力 (LED の PWM 出力) の概要 .....</b>         | <b>149</b> |
| <b>3-9-2 青色 LED をブレッドボード上に取り付ける .....</b>            | <b>150</b> |
| <b>3-9-3 Arduino のスケッチの解説 .....</b>                  | <b>152</b> |
| (1) PWM で使用するピンの指定 .....                             | 153        |
| (2) コマンドデータの処理 .....                                 | 153        |
| (3) PWM 値の出力 .....                                   | 153        |
| <b>3-9-4 Android プロジェクトと XML ファイルの作成 .....</b>       | <b>154</b> |
| ■ AndroidManifest.xml の追加 .....                      | 154        |
| ■ accessory-filter.xml の作成 .....                     | 155        |
| ■ レイアウト XML の作成 .....                                | 155        |
| <b>3-9-5 Java プログラムの作成 .....</b>                     | <b>156</b> |
| ■ ソースコードの解説 .....                                    | 160        |
| (1) レイアウトされたシークバー用インスタンス .....                       | 160        |
| (2) レイアウトされたシークバーとインスタンス mSeekBar1 の結び付け .....       | 160        |
| (3) シークバーを動かしたときの処理 .....                            | 160        |
| (4) シークバーをスライドした量 (0 ~ 255) をコマンドデータに変換 .....        | 160        |
| (5) 出力ストリームに buffer [] 配列データを書き込む .....              | 160        |
| <b>3-9-6 アナログ出力 (LED の PWM 出力) の動作確認 .....</b>       | <b>161</b> |
| <b>3-10 Android へのデジタル入力 (タクトスイッチ) .....</b>         | <b>162</b> |
| <b>3-10-1 デジタル入力 (タクトスイッチ) の概要 .....</b>             | <b>162</b> |
| <b>3-10-2 タクトスイッチをブレッドボード上に取り付ける .....</b>           | <b>163</b> |
| <b>3-10-3 Arduino のスケッチの解説 .....</b>                 | <b>164</b> |
| (1) デジタル入力で使用するピンの指定 .....                           | 165        |
| (2) 入力ポートの指定 .....                                   | 165        |
| (3) 最初のタクトスイッチの状態を保存 .....                           | 165        |
| (4) デジタル入力の読み込み .....                                | 166        |
| (5) コマンドデータの送信 .....                                 | 166        |
| (6) USB アクセサリへの書き込み .....                            | 166        |
| (7) タクトスイッチの状態をシリアルモニターに出力 .....                     | 166        |
| <b>3-10-4 シリアルモニターの確認 .....</b>                      | <b>166</b> |
| <b>3-10-5 Android プロジェクトと XML ファイルの作成 .....</b>      | <b>168</b> |
| ■ AndroidManifest.xml の追加 .....                      | 168        |
| ■ accessory-filter.xml の作成 .....                     | 169        |
| ■ レイアウト XML の作成 .....                                | 170        |
| <b>3-10-6 Java プログラムの作成 .....</b>                    | <b>170</b> |
| ■ ソースコードの解説 .....                                    | 175        |
| (1) Runnable インターフェースを実装してスレッドを使う .....              | 175        |
| (2) レイアウトされたテキストビュー用インスタンス .....                     | 175        |
| (3) レイアウトされたテキストビューとインスタンスの結び付け .....                | 175        |
| (4) 入出力用のストリームを確保 (今回は入力のみ) .....                    | 175        |
| (5) run () メソッドに書かれた処理がスレッドとして実行される .....            | 176        |

|                                                              |            |
|--------------------------------------------------------------|------------|
| (6) スレッドがスタートしたら実行される処理 .....                                | 176        |
| (7) 入力ストリームから buffer [] 配列にデータを読み込む .....                    | 176        |
| (8) メッセージの内容を作る .....                                        | 176        |
| (9) ハンドラの宣言と処理 .....                                         | 176        |
| <b>3-10-7 デジタル入力 (タクトスイッチ) の動作確認 .....</b>                   | <b>176</b> |
| <b>3-11 Android へのアナログ入力 (光センサ : CdS セル) .....</b>           | <b>178</b> |
| <b>3-11-1 アナログ入力 (光センサ : CdS セル) の概要 .....</b>               | <b>178</b> |
| <b>3-11-2 光センサ (CdS セル) をブレッドボード上に取り付ける .....</b>            | <b>179</b> |
| <b>3-11-3 Arduino のスケッチの解説 .....</b>                         | <b>180</b> |
| (1) アナログ入力で使用するピンの指定 .....                                   | 181        |
| (2) アナログ入力ピンの初期化は不要 .....                                    | 181        |
| (3) アナログ入力の処理 .....                                          | 181        |
| (4) 光センサからアナログ入力の読み取り .....                                  | 182        |
| (5) 上位 8 ビットの取り出し .....                                      | 182        |
| (6) 下位 8 ビットの取り出し .....                                      | 182        |
| <b>3-11-4 シリアルモニターの確認 .....</b>                              | <b>182</b> |
| <b>3-11-5 Android プロジェクトと XML ファイルの作成 .....</b>              | <b>183</b> |
| ■ AndroidManifest.xml の追加 .....                              | 183        |
| ■ accessory-filter.xml の作成 .....                             | 184        |
| ■ レイアウト XML の作成 .....                                        | 185        |
| <b>3-11-6 Java プログラムの作成 .....</b>                            | <b>185</b> |
| ■ ソースコードの解説 .....                                            | 191        |
| (1) レイアウトされたテキストビュー用インスタンス .....                             | 191        |
| (2) レイアウトされたトグルボタンとインスタンスの結び付け .....                         | 191        |
| (3) 入力ストリームから buffer [] 配列にデータを読み込む .....                    | 191        |
| (4) メッセージの内容を作る .....                                        | 191        |
| (5) メッセージは CdsMsg クラスを使っているため、CdsMsg クラスの<br>インスタンスに格納 ..... | 191        |
| (6) 光センサ (CdS セル) の状態を保持するクラス .....                          | 191        |
| (7) 8 ビットの byte 型変数を、符号ビットを含めてビットマスクで取り出す .....              | 192        |
| (8) 上位 8 ビットを 8 ビット左シフトし下位 8 ビットと合わせる .....                  | 192        |
| <b>3-11-7 アナログ入力 (光センサ : CdS セル) の動作確認 .....</b>             | <b>192</b> |
| <b>3-12 Android からデジタル / アナログの入出力すべてを制御する .....</b>          | <b>194</b> |
| <b>3-12-1 デジタル / アナログ入出力の概要 .....</b>                        | <b>194</b> |
| <b>3-12-2 パーツをブレッドボード上に取り付ける .....</b>                       | <b>196</b> |
| <b>3-12-3 Arduino のスケッチの解説 .....</b>                         | <b>197</b> |
| (1) デジタル / アナログ入出力で使用するピンの指定 .....                           | 199        |
| (2) 外部インテントの指定 .....                                         | 199        |
| (3) ピンの初期化 .....                                             | 199        |
| (4) Android との接続処理 .....                                     | 200        |
| (5) データの読み込み .....                                           | 200        |
| (6) デジタル出力 (LED のオン / オフ) の処理 .....                          | 200        |
| (7) アナログ出力 (LED の明暗の制御) の処理 .....                            | 200        |
| (8) デジタル入力 (タクトスイッチのオン / オフ) の処理 .....                       | 200        |
| (9) アナログ入力 (光センサのアナログ量) の処理 .....                            | 200        |
| <b>3-12-4 シリアルモニターの確認 .....</b>                              | <b>200</b> |

|        |                                |     |
|--------|--------------------------------|-----|
| 3-12-5 | AndroidプロジェクトとXMLファイルの作成       | 202 |
|        | ■AndroidManifest.xmlの追加        | 202 |
|        | ■accessory-filter.xmlの作成       | 203 |
|        | ■レイアウトXMLの作成                   | 203 |
| 3-12-6 | Javaプログラムの作成                   | 204 |
|        | ■ソースコードの解説                     | 211 |
|        | (1) 入出力用ストリームを用意する             | 211 |
|        | (2) レイアウトされたウィジェットに対するインスタンス   | 211 |
|        | (3) レイアウトされたウィジェットとインスタンスの結び付け | 211 |
|        | (4) デジタル出力処理                   | 212 |
|        | (5) アナログ (PWM) 出力処理            | 212 |
|        | (6) デジタル出力をバイト配列にセット           | 212 |
|        | (7) アナログ (PWM) 出力をバイト配列にセット    | 212 |
|        | (8) タクトスイッチのオン/オフ入力指定コマンド      | 212 |
|        | (9) 光センサのアナログ入力指定コマンド          | 212 |
|        | (10) タクトスイッチ・オン/オフ入力の表示        | 213 |
|        | (11) 光センサ・アナログ入力の表示            | 213 |
| 3-12-7 | デジタル/アナログ入出力の動作確認              | 213 |
| 3-12-8 | デジタル/アナログ入出力のまとめ               | 215 |

Chapter 4 **Androidで赤外線通信** ..... 219

|       |                                |     |
|-------|--------------------------------|-----|
| 4-1   | 赤外線でチョロQを動かす                   | 220 |
| 4-1-1 | 赤外線通信とは                        | 220 |
| 4-1-2 | チョロQハイブリッドとは                   | 221 |
| 4-1-3 | 通信プロトコルの解析の準備                  | 221 |
| 4-1-3 | 通信プロトコルの解析                     | 223 |
| 4-2   | ArduinoとPCからチョロQを操作する          | 225 |
| 4-2-1 | 送信機の回路                         | 225 |
| 4-2-2 | Arduinoのスケッチ作成                 | 226 |
|       | ■delayMicroseconds関数を使用する上での注意 | 226 |
| 4-2-3 | ソースコードの解説                      | 229 |
|       | ■定義                            | 229 |
|       | ■初期化                           | 229 |
|       | ■ループ処理                         | 229 |
|       | ■ctrlIr関数による赤外線LEDの制御          | 230 |
|       | ■sendData関数によるデータ送信            | 230 |
| 4-2-4 | シリアルモニターの起動                    | 231 |
| 4-3   | Androidから送信機を操作する              | 232 |
| 4-3-1 | Arduinoのスケッチの改造                | 232 |
| 4-3-2 | ソースコードの解説                      | 235 |
|       | (1) ADKの設定                     | 235 |
|       | (2) ADKを起動                     | 235 |
|       | (3) Androidと接続しているかどうかのチェック    | 235 |
|       | (4) Androidからのデータを受信           | 235 |

|                    |     |
|--------------------|-----|
| (5) Androidへデータを送信 | 235 |
|--------------------|-----|

4-4 **Androidプログラムの作成** ..... 236

|       |                                                    |     |
|-------|----------------------------------------------------|-----|
| 4-4-1 | 作成するプログラムの概要                                       | 236 |
| 4-4-2 | ソースコードと解説                                          | 236 |
|       | ■BaseActivity.java                                 | 236 |
|       | ■ADKQsteerActivity.java                            | 241 |
|       | (1) 制御用のデータクラス                                     | 246 |
|       | (2) 制御データの定義                                       | 246 |
|       | (3) 制御データのリスト                                      | 246 |
|       | (4) 操作用のView                                       | 246 |
|       | (5) バンドデータ                                         | 246 |
|       | (6) ADKボードにコマンドを送信する                               | 246 |
|       | (7) 制御データクラスから送信するデータを生成する                         | 246 |
|       | (8) BaseActivity#writeBufferメソッドを使用してADKボードへデータを送信 | 247 |
|       | (9) 制御データ用のリストに制御データクラスを追加する                       | 247 |
|       | (10) タッチイベントのコールバックを登録する                           | 247 |
|       | (11) ラジオボタンの状態変化に応じてバンドを変更する                       | 247 |
|       | (12) ボタンクリック時の動作                                   | 247 |
|       | (13) タッチ時の動作                                       | 247 |
|       | (14) ADKボードとの通信が開始された際にコールされるメソッド                  | 247 |
|       | (15) ADKボードとの通信が切断された際にコールされるメソッド                  | 247 |
|       | (16) ADKボードからデータを受信するスレッド                          | 247 |
|       | ■AndroidManifest.xml                               | 248 |
|       | ■main.xml                                          | 249 |
|       | ■dimens.xml                                        | 252 |
|       | ■string.xml                                        | 252 |
|       | ■accessory_filter.xml                              | 253 |
| 4-4-3 | プログラムの実行                                           | 253 |

4-5 **ロボットに赤外線通信機能を実装** ..... 254

|       |                     |     |
|-------|---------------------|-----|
| 4-5-1 | ロボットの構成             | 254 |
| 4-5-2 | ロボットに搭載する回路         | 254 |
|       | ■モータドライバICを利用する理由   | 256 |
| 4-5-3 | ロボットの製作             | 257 |
|       | ■材料の用意              | 257 |
|       | ■回路の実装              | 259 |
|       | ■ギアボックスの設置          | 259 |
|       | ■Arduinoの設置         | 260 |
|       | ■自作シールドの取り付け        | 261 |
| 4-5-4 | 送信機の改造              | 262 |
| 4-5-5 | 受信側Arduinoのプログラム    | 262 |
| 4-5-5 | 受信側Arduinoのプログラムの解説 | 267 |
|       | (1) ロボットのバンド設定      | 267 |
|       | (2) 各状態でのスピード       | 267 |
|       | (3) ピン設定            | 267 |
|       | (4) 制御データの定義        | 268 |
|       | (5) モータ1を回す         | 268 |
|       | (6) setup関数         | 268 |

|                          |     |
|--------------------------|-----|
| (7) loop 関数              | 268 |
| (8) スタートデータ検出待ち          | 268 |
| (9) バンドデータを取得            | 269 |
| (10) 制御データを取得            | 269 |
| (11) バンドデータが一致する場合のみ動作する | 269 |
| (12) バンドデータが一致しない場合は停止   | 269 |
| 4-5-6 まとめ                | 269 |

**Chapter 5 Androidで温度モニタリング** ..... 273

|                              |     |
|------------------------------|-----|
| <b>5-1 各種機器の準備と概要</b>        | 274 |
| 5-1-1 温度センサーの接続              | 274 |
| 5-1-2 使用するマイコン               | 274 |
| <b>5-2 PIC24Fの開発環境の構築</b>    | 276 |
| 5-2-1 必要なソフトウェア              | 276 |
| 5-2-2 開発環境のダウンロード            | 276 |
| 5-2-3 統合開発環境のインストール          | 278 |
| 5-2-3 C言語コンパイラのインストール        | 280 |
| 5-2-4 USB Frameworkのインストール   | 281 |
| 5-2-5 デバッグ環境の構築              | 282 |
| <b>5-3 マイコン基板の作成とファームウェア</b> | 285 |
| 5-3-1 マイコン基板の回路図と部品          | 285 |
| 5-3-2 温度センサー                 | 286 |
| 5-3-3 マイコン基板の制作              | 286 |
| 5-3-4 ファームウェア                | 288 |
| <b>5-4 プロジェクトの作成</b>         | 290 |
| 5-4-1 空のプロジェクトの作成            | 290 |
| 5-4-2 プロジェクトの設定              | 292 |
| ■プロジェクトフォルダ                  | 292 |
| ■ヘッダーファイル                    | 292 |
| ■USB Framework 関連のファイル       | 293 |
| <b>5-5 マイコンのソースコードと解説</b>    | 295 |
| 5-5-1 マイコンのソースコード            | 295 |
| ■main.c                      | 295 |
| ■HardwareProfile.h           | 303 |
| 5-5-2 ソースコードの解説              | 303 |
| (1) コンフィグレーション               | 303 |
| (2) コマンド定義                   | 304 |
| (3) パケット構造体                  | 304 |
| (4) ADKの情報                   | 304 |
| (5) メイン関数                    | 304 |

|                                   |     |
|-----------------------------------|-----|
| (6) USB Frameworkの初期化             | 304 |
| (7) Android Host DriverへADKの情報を設定 | 304 |
| (8) USB Frameworkのタスクを呼び出し        | 304 |
| (9) USBから読み込み処理                   | 304 |
| (10) 読み込み完了時の処理                   | 304 |
| (11) 読み出したデータがある場合                | 305 |
| (12) 書き込み処理が必要な場合                 | 305 |
| (13) A/D変換コマンドへの応答                | 305 |
| (14) A/D変換の初期化                    | 305 |
| (15) A/D変換の読み込み                   | 305 |

**5-6 Androidのソースコードと解説** ..... 306

|                             |     |
|-----------------------------|-----|
| <b>5-6-1 Androidのソースコード</b> | 306 |
| ■AndroidManifest.xml        | 306 |
| ■accessory_filter.xml       | 307 |
| ■main.xml                   | 307 |
| ■ThermometerActivity.java   | 308 |
| <b>5-6-2 ソースコードの解説</b>      | 313 |
| (1) accessory_filterの設定     | 313 |
| (2) タイマーの設定                 | 313 |
| (3) 受信したAD値を計算              | 313 |

**5-7 温度センサーアプリの実行結果** ..... 314

|                |     |
|----------------|-----|
| 5-7-1 アプリの実行画面 | 314 |
| 5-7-2 今後の拡張    | 315 |

**Chapter 6 MicroBridgeの利用** ..... 319

**6-1 MicroBridgeを使うための開発環境の構築** ..... 320

|                                        |     |
|----------------------------------------|-----|
| 6-1-1 MicroBridgeとは                    | 320 |
| 6-1-2 必要なハードウェアとソフトウェア                 | 320 |
| ■ハードウェア                                | 320 |
| ■ソフトウェア                                | 321 |
| 6-1-3 Arduinoの開発環境にMicroBridgeライブラリを追加 | 321 |

**6-2 Arduinoからadbのコマンドを実行** ..... 324

|                              |     |
|------------------------------|-----|
| 6-2-1 動作確認の手順                | 324 |
| 6-2-2 サンプルスケッチの解説            | 326 |
| (1) ヘッダファイル                  | 327 |
| (2) adbEventHandler コールバック関数 | 327 |
| (3) USBポートをシリアルポートとして使用      | 328 |
| (4) MicroBridgeの初期化          | 328 |
| (5) addConnection関数          | 328 |
| (6) ポーリング                    | 328 |

|            |                                             |     |
|------------|---------------------------------------------|-----|
| <b>6-3</b> | <b>デモアプリを動かそう</b>                           | 329 |
| 6-3-1      | デモアプリを動かすための環境                              | 329 |
| 6-3-2      | サーボモータの準備                                   | 329 |
| 6-3-3      | Androidアプリの準備                               | 330 |
| 6-3-4      | デモプログラムの実行                                  | 332 |
| 6-3-5      | Arduinoのソースコードの解説                           | 333 |
|            | (1)ヘッダファイル                                  | 334 |
|            | (2)サーボオブジェクトの生成                             | 334 |
|            | (3)Connectionオブジェクトの生成                      | 334 |
|            | (4)送信時間の調整                                  | 334 |
|            | (5)不要な変数                                    | 335 |
|            | (6)データ受信時の動作                                | 335 |
|            | (7)シリアル通信の設定                                | 335 |
|            | (8)Arduinoが起動した時間                           | 335 |
|            | (9)サーボ用の信号を出すピン番号の設定                        | 335 |
|            | (10)TCP/IP通信の開始                             | 335 |
|            | (11)入力ピンの値の取得                               | 335 |
| 6-3-6      | Androidのソースコードの解説                           | 335 |
|            | ■ServoControlクラス                            | 336 |
|            | (1)インスタンスの生成とスレッドの開始                        | 337 |
|            | (2)JoystickViewクラスのインスタンスの生成                | 337 |
|            | ■JoystickViewクラス                            | 337 |
|            | (1)コールバックリスナーの登録                            | 339 |
|            | (2)タッチイベントの処理                               | 339 |
| <b>6-4</b> | <b>応用例：ArduinoからAndroid端末のカメラのシャッターを切る</b>  | 340 |
| 6-4-1      | ハードウェアの準備                                   | 340 |
| 6-4-2      | ソースコードの解説                                   | 341 |
|            | (1)グローバル変数の定義                               | 343 |
|            | (2)ADBのイベントハンドラ                             | 343 |
|            | (3)ピンモードの設定                                 | 343 |
|            | (4)ADBサブシステムの初期化                            | 343 |
|            | (5)ADB用のストリームがオープンしている時にAndroid上のshellへ書き込み | 343 |
|            | (6)スイッチを押した際の処理                             | 343 |
| 6-4-3      | まとめ                                         | 343 |

Appendix **付録** ..... 347

|            |                          |     |
|------------|--------------------------|-----|
| <b>A-1</b> | <b>ピンソケットのハンダ付け</b>      | 348 |
| A-1-1      | USBホストシールドをADKで使えるようにする  | 348 |
|            | ■ピンソケットのハンダ付け(難易度：★☆☆)   | 348 |
|            | ■5Vのジャンパーのハンダ付け(難易度：★★☆) | 348 |
|            | ■ハンダ付け作業に必要な道具           | 349 |
| A-1-2      | ピンソケットのハンダ付け             | 350 |
|            | ■ピンソケットのハンダ付けのコツ         | 350 |
|            | ■ピンソケットのハンダ付けの手順         | 351 |

|       |                              |     |
|-------|------------------------------|-----|
| A-1-3 | 5Vのジャンパーのハンダ付け               | 354 |
|       | ■USBホストシールドにジャンパーのハンダ付けを行う理由 | 354 |
|       | ■レギュレータ除去と5Vジャンパーのハンダ付けの手順   | 356 |

**A-2** **Androidとハードウェア機器の接続方法** ..... 360

|       |                        |     |
|-------|------------------------|-----|
| A-2-1 | Androidとハードウェアのさまざまな接続 | 360 |
| A-2-2 | Wi-Fi接続                | 360 |
|       | ■通信プロトコル               | 361 |
|       | ■Wi-Fi接続の長所            | 361 |
|       | ■Wi-Fi接続の短所            | 361 |
| A-2-3 | Bluetooth通信            | 362 |
|       | ■Bluetooth接続の長所        | 363 |
|       | ■Bluetooth接続の短所        | 363 |
| A-2-4 | 赤外線通信                  | 363 |
|       | ■赤外線通信の長所              | 363 |
|       | ■赤外線通信の短所              | 363 |
| A-2-5 | USB Accessory接続        | 363 |
|       | ■USB Accessory接続の長所    | 363 |
|       | ■USB Accessory接続の短所    | 364 |
| A-2-6 | USB Debug接続            | 364 |
|       | ■USB Debug接続の長所        | 364 |
|       | ■USB Debug接続の短所        | 364 |
| A-2-7 | USB HOST接続             | 364 |
|       | ■USB HOST接続の長所         | 364 |
|       | ■USB HOST接続の短所         | 364 |

**A-3** **ロボットの製作事例** ..... 365

|       |                                |     |
|-------|--------------------------------|-----|
| A-3-1 | Android用ロボット開発キット「Coroid」      | 365 |
|       | ■Android端末からロボットを無線LANで制御      | 365 |
|       | ■Android端末に、無線LANカメラ映像を動画表示    | 365 |
| A-3-2 | IOIOボードでラジコン戦車操縦               | 366 |
| A-3-3 | Android 端末から Robot と Stage を制御 | 367 |

**A-4** **KinectとAndroidを接続する** ..... 369

|       |                            |     |
|-------|----------------------------|-----|
| A-4-1 | KinectとXtion Pro Live      | 369 |
| A-4-2 | AndroidでNUIアプリケーションを開発する環境 | 370 |
| A-4-3 | 距離センサーを使ったアプリケーション         | 373 |

|      |     |
|------|-----|
| 索引   | 376 |
| 著者紹介 | 382 |

|                                            |     |
|--------------------------------------------|-----|
| ハードウェア屋から見た Android                        | 013 |
| IP アドレスを確認する                               | 050 |
| ADK ボードのシリアル出力                             | 081 |
| Android とマイコンをシリアル通信 (RS232C、UART) でつなげる方法 | 083 |
| Arduino をはじめよう!                            | 086 |
| ダイアログのメッセージの変更                             | 118 |
| 抵抗のカラーコードと LED の電流制限抵抗を計算するアプリ             | 124 |
| ADK で困った時の対処法                              | 146 |
| Arduino を利用した初心者向けプログラミング環境 Eduino         | 216 |
| 画像処理ライブラリ (OpenCV) の活用                     | 270 |
| マイコン基板の入手                                  | 288 |
| ロバスト性の話                                    | 316 |
| プログラマブルロジック FPGA の紹介と接続方法                  | 344 |