

Bluetooth通信

Bluetooth通信

Bluetoothにはバージョンがある。

1対1の通信もN対1の通信も可能

Bluetooth4.0規格の一部に

BLE(Bluetooth Low Energy)がある

BLEと従来のものを区別するために、従来のものは

「Classic Bluetooth」と呼ばれる

Bluetooth通信

BLEを搭載していない機器ではBLE通信できない！

逆にBLE通信しかできない機器もある

(後述。15pで説明しています)

AndroidではBLE対応と非対応が混在しているので、BLE通信にのみ対応していると動かない機種がある！

Bluetooth通信

機器の種類ごとに策定されたプロトコルの使用方法
を「プロファイル」と呼ぶ
通信しようとしている機器が同じプロトコルを持っ
ている場合に限り、通信をおこなえる

参考：Wikipedia

Bluetooth通信

各Bluetooth機器は固有のMACアドレスを持っている。

プロファイル毎に決められたUUIDがある。

(<https://www.bluetooth.com/specifications/assigned-numbers/service-discovery>)

AndroidでBluetooth

AndroidではBLE搭載機種と未搭載機種が混在しているため、とりあえずClassic Bluetooth通信をすることになります。

プロファイルはSPP(Serial Port Profile)とします。

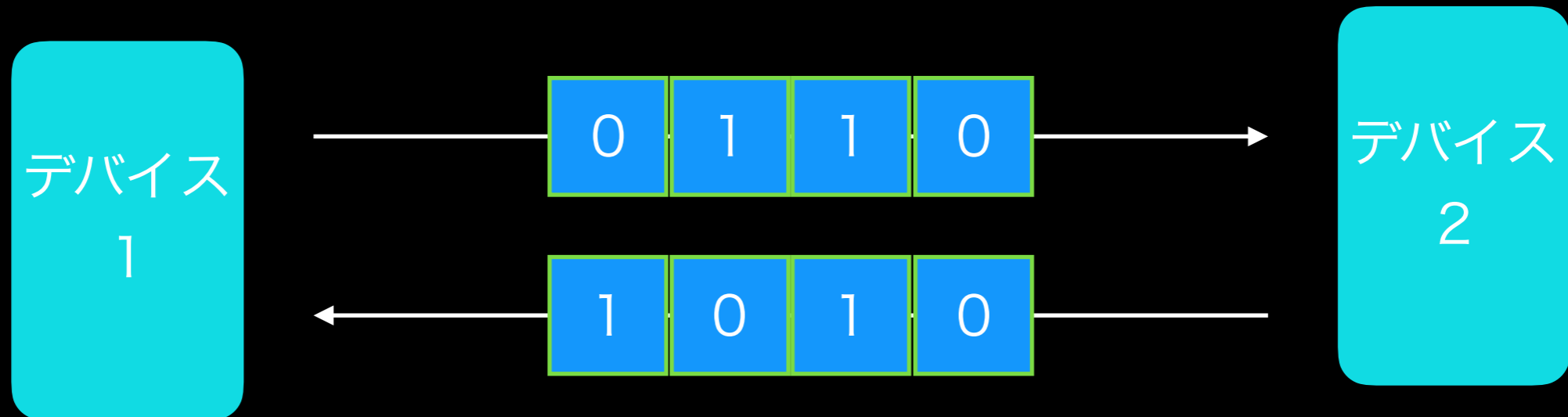
AndroidでBluetooth

SPPとは、Bluetoothをシリアルポートのように使う
プロファイルのことです。

シリアルポートとは、データを一度に1ビットずつ
送受信するあれです。

AndroidでBluetooth

要するに、こんな感じです



単純な通信です。こわくない！

AndroidでBluetooth

コードは長くなるので、全部説明しません。
重要なところだけ説明していきます。

AndroidでBluetooth

AndroidでBluetoothを使うには、
まず、

Manifest (←結構忘れてしまいますw)
パーミッションを追加しましょう

AndroidでBluetooth

Bluetoothを使えるようになるには、
端末の設定もONにしておかないといけません
OFFの場合は、ONにしてもらえるように
ユーザーに促します

AndroidでBluetooth

次に、接続先を取得します

接続履歴から接続するのは、アドレスが残っていて
認証も終わっているのですぐに接続できます

接続したことのない端末と接続するには、
ブロードキャストレシーバで
接続可能な端末を検出します
そのとき、相手の端末を検出可能に
してもらわないといけません

AndroidでBluetooth

最後に、通信を行うために
サーバーとクライアントを作ります

AndroidでBluetooth

サーバー側

アプリが動いている間はずっと受信し続けるように
します

accept()で接続先を取得して、
データを受け取ってから送信しています
最後に接続を切ります

※接続先との接続を切るだけなので、
受信は続けます

AndroidでBluetooth

クライアント側

接続履歴やブロードキャストレシーバから
接続先のアドレスを取得し、接続を開始します

`connect()`で接続できたら、

データを送信してから受信しています

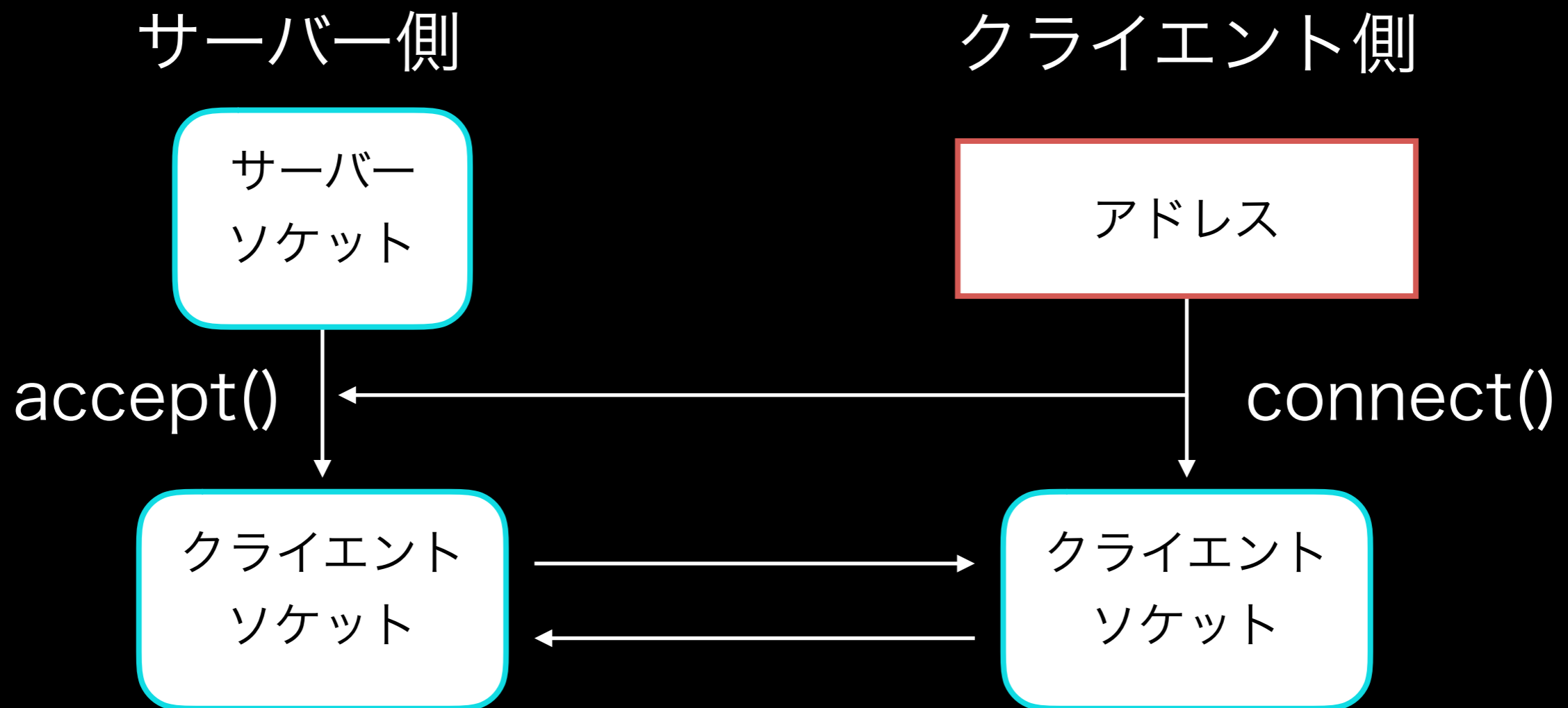
最後に、接続をきります

クライアント側は一度接続をきったら終わりです

AndroidでBluetooth

ソケットについて

通信をするときの出入り口のようなものです



AndroidでBluetooth

データの送受信はI/Oストリームでします
(C++をしたことがある人なら馴染みがあるかも！)

I/Oストリームとは、

InputとOutputのストリームです
Inputで読み込み（ここでは受信）、
Outputで書き込み（ここでは送信）
します

こわくない！単純です

AndroidでBluetooth

コードは私のGithubにあげておきます

https://github.com/TomokoKondo/bluetooth_sample

AndroidでBluetooth

私のサンプルでは、

「新規デバイス検出」ボタンを押して300秒以内に
別の端末で

「新規デバイス検出」ボタンを押した端末に接続を
試みると、一度だけ一番上のエディットテキストの
文字を送受信します

もう一回通信するには、また同じことをしないと
通信できません！

へっぼいサンプルですw

AndroidでBluetooth

参考にしたサイト

[http://techbooster.jp.net/android/
application/5191/](http://techbooster.jp.net/android/application/5191/)

全部で3部構成となっています。

このリンク先はその1です。

ここからその2とその3にリンクできます。

かなり分かりやすく説明してくれています。

AndroidでBluetooth

分かりやすく説明してくれているのですが、
残念なことにタイプミスやら何やらがあって
コピペじゃ使えません・・・

説明だけ読んであとは私のコードをみてください
頑張ってコメントを書くようにしたので、
多少見やすいかと！

(クライアントとサーバーのところはちょっと力尽きてますw)

Tips

BLEについて

BLE

従来のBluetoothよりも大幅に省電力化された。
従来の仕様からは独立しており、
後方互換性をもたない。

BLE

Androidでは、4.3（APIレベル18）から導入された。

iOSでは、iOS 4Sから搭載された。

iOS 4に対応させようとする、MFiの取得が必要となる。

BLE

BLEに対応しているかどうかを「Bluetooth Smart」「Bluetooth Smart Ready」「Bluetooth」の3つで分類される。

Bluetoothは、クラシックのみ

Bluetooth Smartは、BLEのみ

Bluetooth Smart Readyは、どちらとも通信できる

GATT

GATTとは、BLE端末同士が接続してデータの送受信を行う際に基づく仕様である。

GATT

- ・ データを転送する際の最小単位を「キャラクタリスティックス」と呼ぶ
- ・ 「ディスクリプタ」はキャラクタリスティックスを構成する属性を定義する
- ・ キャラクタリスティックスが集まって、1つの機能を構成するものを「サービス」と呼ぶ
- ・ 複数のサービスが集まって構成される全体の機能を「プロファイル」と呼ぶ
- ・ プロファイルに基づいてBLE機器同士がデータのやりとりを行う

GATT

先ほどのスライドの説明を図にすると、

GATT

プロフィール

サービス

キャラクタースティクス

ディスクリプタ

ディスクリプタ

キャラクタースティクス

サービス

GATT

こんな感じになります。

ペリフェラル

- ・ BLE機器そのもの
- ・ 周辺にの一定距離に対し、定期的にアドバタイズを行う
- ・ セントラルによるキャラクタリスティクスの操作を受け付ける

セントラル

- ・ BLE機器を利用するもの
- ・ 周辺のBLE機器をスキャンし、任意のGATTあるいはサービスに接続を行う
- ・ 接続したサービスのキャラクターリスティクスを操作し、情報取得や機器の操作を行う

アドバタイズ

- ・ ペリフェラルが自分のもつサービスを接続可能な状態にし、周辺に機器情報を発信すること
- ・ アドバタイズされていないサービスは、セントラルからは発見できない

BLE

この辺の用語をおさえておけば、BLEも怖くない！（はず）

参考にしたサイト

[http://qiita.com/miyatay/items/
4d4ce4ccd7905ddc0144](http://qiita.com/miyatay/items/4d4ce4ccd7905ddc0144)

[http://qiita.com/HideMatsu/items/
6a6758eca86500881126](http://qiita.com/HideMatsu/items/6a6758eca86500881126)