

# CITIZEN

モバイルプリンター

Androidポートパッケージマニュアル

Bluetooth, Wi-Fi, USB

Ver. 1.05

**CITIZEN SYSTEMS JAPAN CO., LTD.**

## 目次

1. はじめに
2. メソッド

バージョン	対象ライブラリ	説明
1.00		公開
1.01		RequestHandlerの説明を追加
1.02	1.064以降	サポートするインターフェースにUSBを追加
1.03	1.071以降	複数接続をサポートするWiFiPortConnectionクラスを追加 BluetoothのConnectにConnectSecureを追加 WiFiのConnectの説明を補足、修正
1.04	1.083以降	表紙のロゴを変更 若干の間違いを修正
1.05	1.092以降	setMacFilterを追加

## 1. はじめに

この「Androidポートパッケージマニュアル」では、Bluetooth、Wi-FiまたはUSBを使用して接続を行うメソッドについて説明します。

## 2. メソッド

BluetoothPortクラス、WiFiPortクラスおよびUSBポートクラスで定義されています。

## 2.1 BluetoothPort

### 2.1.0 getInstance

public static BluetoothPort getInstance()

BluetoothPortのインスタンスを戻します。

### 2.1.1 connect

public void connect(String address) throws IOException

public void connectSecure(String address) throws IOException

インターフェイスを接続先のアドレスに接続します。

- address

接続先のターゲットホストアドレスです。

アドレス形式は区切り記号で区切られた12文字です。(例: 12:34:56:78:90:AB)

### 2.1.2 connect

public void connect(BluetoothDevice device) throws IOException

public void connectSecure(BluetoothDevice device) throws IOException

インターフェイスを特定のBluetoothデバイスに接続します。

- device

接続先のBluetoothデバイスオブジェクトです。

### 2.1.3 disconnect

public void disconnect() throws IOException

現在の接続を切断します。

### 2.1.4 getInputStream

public InputStream getInputStream() throws IOException

この接続の入カストリームを戻します。

### 2.1.5 getOutputStream

public OutputStream getOutputStream() throws IOException

この接続の出カストリームを戻します。

### 2.1.6 isConnected

public boolean isConnected()

インターフェイスが接続されているステータスを返します (trueまたはfalse)。

**注:** 戻り値は必ずしも実行時点の接続の状態を示すわけではありません。

すでに接続されているプリンターの電源を切った場合、"IsConnected()"は、正しい値を返しません。

### 2.1.7 setMacFilter

public void setMacFilter(Boolean filter)

BluetoothのMacアドレスフィルターオプションを設定します。

- filter

true: フィルターを使います。

false: フィルターを使いません。(初期値)

フィルターは、有効なBDアドレス (Macアドレス) のBluetoothモジュール以外とは通信しないようにするための機能です。

## 2.2 Wi-Fi

### 2.2.0 getInstance

public static WiFiPort getInstance()

WiFiPortのインスタンスを戻します。

### 2.2.1 connect

public void connect(String address) throws IOException

アドレスで指定された接続先に接続します。

- address

接続先のターゲットホストアドレスです。

Android 3.0 以降において、UIスレッド(メインスレッド)からネットワーク処理を行うと例外がスローされる仕様となりました。この仕様により、本 SDK の接続処理は、別スレッド上にコーディングして頂く必要があります。例としてサンプルコードをご参照ください。

--

例外メッセージ: android.os.NetworkOnMainThreadException

### 2.2.2 connect

public void connect(String ip, int port) throws IOException

IPアドレスとポートで指定された接続先に接続します。

- ip

接続先のターゲットホストアドレスです。

- port

接続先のターゲットホストのポートです。初期値は9100です。

Android 3.0 以降において、UIスレッド(メインスレッド)からネットワーク処理を行うと例外がスローされる仕様となりました。この仕様により、本 SDK の接続処理は、別スレッド上にコーディングして頂く必要があります。例としてサンプルコードをご参照ください。

--

例外メッセージ: android.os.NetworkOnMainThreadException

### 2.2.3 disconnect

public void disconnect() throws IOException

現在の接続を切断します。

### 2.2.4 getInputStream

public InputStream getInputStream() throws IOException

この接続の入力ストリームを戻します。

### 2.2.5 getOutputStream

public OutputStream getOutputStream() throws IOException

この接続の出力ストリームを返します。

### 2.2.6 isConnected

public boolean isConnected()

インターフェイスが接続されていたステータスを返します (trueまたはfalse)。

**注:** 戻り値は必ずしも実行時点の接続状態を示すわけではありません。

すでに接続されているプリンターの電源を切った場合、"isConnected()"は、正しい値を返しません。

## 2.3 RequestHandler

### 2.3.1 RequestHandler の使い方

RequestHandler にはRunnableインターフェイスが実装されています。

BluetoothかWiFiPortの接続が確立したら、下記のようにRequestHandlerスレッドを動作させなければなりません。

(USBPortConnectionかWiFiPortConnectionを使った場合は除く)

接続がクローズするまで動いている必要があります。

接続がクローズしたら、スレッドはinterrupt()を使って終了させなければなりません。

さらに詳細の情報が必要な場合は、SDKのサンプルコードを参照してください。

```
Thread handlerThread;
.....
//実行ルーチン
handlerThread = new Thread(new RequestHandler());
handlerThread.start();
.....

//終了ルーチン NullPointerException を避けるため
If ((handlerThread != null) && (handlerThread.isAlive()))
    handlerThread.interrupt();
.....
```

USBかWiFiPortConnectionクラスを使って接続した場合は、RequestHandlerを動作させないこと。

## 2.4 USB

このクラスは、USB OTG (USB On The Go)によるプリンターへの接続をサポートしています。

USB OTGケーブルが必要で、通常のマイクロUSBからUSBへの変換アダプターでは動作しません。  
また、OTG機能をサポートしたAndroid端末が必要になります。

### 2.4.1 USB Port Class

```
public USBPort(UsbManager usbManager)
```

- コンストラクターメソッドです。

[パラメーター]

- usbManager

```
public USBPortConnection connect_device(int model)
```

- USBデバイスオブジェクトに接続します。

[パラメーター]

- model : プリンターモデル (USBPortクラスのCMP\_PORT\_USB)

[戻り値]

- USBPortConnection : USBPortConnectionオブジェクト

### 2.4.2 USB PortConnection Class

USBPortConnectionオブジェクトは、USB接続が確立した時に生成されます。

このオブジェクトは、ESCPOSPrinter、CPCLPrinter、AndroidMSRクラスのコンストラクタのパラメーターとして使われます。

```
public void close()
```

- 接続を閉じます。



## 2.5 WiFiPortConenction

このクラスは、WiFiインターフェース経由で複数のプリンターへの接続をサポートします。

WiFiPortConnectionのインスタンスはWiFiPort classのopen()メソッドにより生成されます。

### 2.5.1 WiFiPort Class

```
public static WiFiPort getInstance()
```

- WiFiPortのインスタンスを戻します。

```
public WiFiPortConnection open(String address)
```

- WiFiPortConnectionのインスタンスを戻します。

[/パラメーター]

- address : 接続先のホストのIPアドレス

### 2.5.2 WiFiPortConnection Class

USBPortConnectionオブジェクトは、USB接続が確立した時に生成されます。

このオブジェクトは、ESCPOSPrinter、CPCLPrinter、AndroidMSRクラスのコンストラクタのパラメーターとして使われます。

```
public void close()
```

- 接続を閉じます。

```
public boolean isConnected()
```

- 接続されているインターフェースのステータスを戻します。(true または false).