



# AsReader Finger-Type

**C# SDK 開発マニュアル**

ASR-023B

## 修正履歴

バージョン	修正内容	日付
1.0	新規作成	2020/10/30
1.1	WhenReceivedBluetoothDevice メソッドのパラメーターが BluetoothDevice device から List<BluetoothDevice> devices に変更された。	2020/12/7

# 目次

前書き .....	5
<b>1. SDK の追加と使用 .....</b>	<b>6</b>
1.1. SDK の追加 .....	6
1.1.1. 新しい Windows フォームアプリケーションを作成 .....	6
1.1.2. 引用の追加 .....	6
1.2. SDK の使用 .....	9
1.2.1. 命名スペースをインポート .....	9
1.2.2. デバイスの検索 .....	9
1.2.3. デバイスの検索を停止 .....	11
1.2.4. デバイスを接続 .....	11
1.2.5. バーコードのスキャン .....	12
1.2.6. バーコードのスキャンを停止 .....	14
1.2.7. 接続を切断 .....	14
<b>2. AsFinger クラス .....</b>	<b>15</b>
2.1. メソッド .....	15
2.1.1. Connect .....	15
2.1.2. Disconnect .....	16

2.1.3.	StartScan .....	16
2.1.4.	StopScan .....	17
2.1.5.	GetBattery .....	18
2.1.6.	GetFirmwareVersion.....	18
2.1.7.	GetSDKVersion .....	19
2.1.8.	GetBluetoothDevice.....	20
2.1.9.	SendData .....	20
2.2.	デリゲート .....	21
2.2.1.	WhenAsFingerIsConnected .....	21
2.2.2.	WhenReceivedBarcodeData .....	22
2.2.3.	WhenReceivedBattery.....	23
2.2.4.	WhenReceivedFirmwareVersion .....	24
2.2.5.	WhenReceivedData .....	24
<b>3.</b>	<b>AsFingerManager クラス.....</b>	<b>26</b>
3.1.	メソッド.....	26
3.1.1.	StartSearching .....	26
3.1.2.	StopSearching .....	26
3.1.3.	GetSearchBluetoothType .....	27
3.2.	デリゲート .....	28
3.2.1.	WhenReceivedBluetoothDevice.....	28
<b>4.</b>	<b>BluetoothDevice クラス.....</b>	<b>30</b>

4.1. プロパティ .....	30
<b>5. Types クラス .....</b>	<b>31</b>
5.1. 列挙型 .....	31
5.1.1. BluetoothType .....	31

## 前書き

この記事の主な目的：

- 開発者が SDK ライブラリを使用した Windows アプリケーションの開発と開発環境の構築のサポート。
- SDK ライブラリについての説明。

開発ツール：

- Visual Studio 2019

システム要求：

- Windows 10 1703+

# 1.SDK の追加と使用

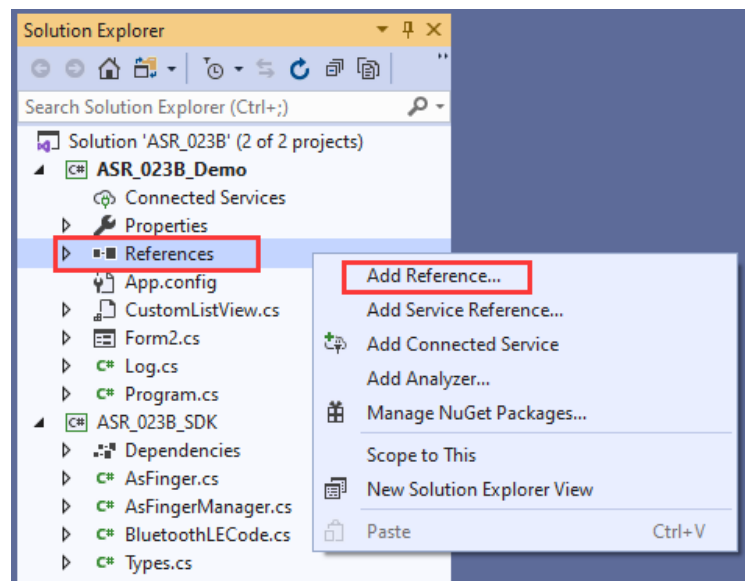
## 1.1. SDK の追加

### 1.1.1. 新しい Windows フォームアプリケーションを作成

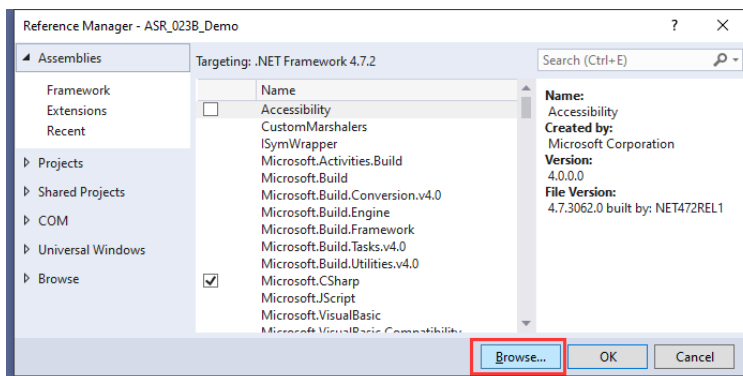
SDK をプロジェクトフォルダにコピーします。デバッグの場合は、プログラムの現在のディレクトリの `bin / Debug` フォルダにコピーします。リリースの場合は、プログラムの現在のディレクトリの `bin / Release` フォルダにコピーします。

### 1.1.2. 引用の追加

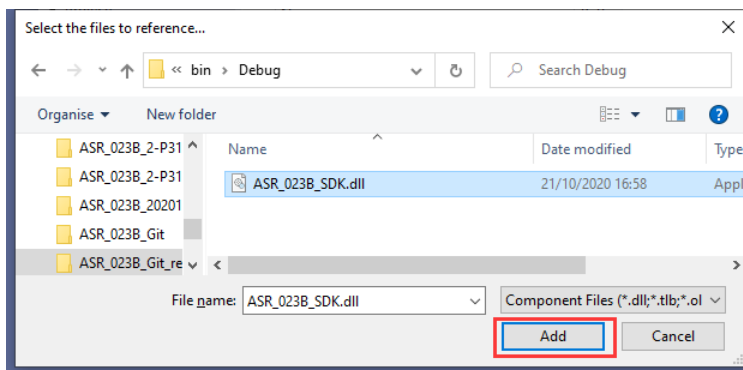
「References」を右クリックし、「Add Reference」を選択します。



「Browse」 ボタンをクリックします。

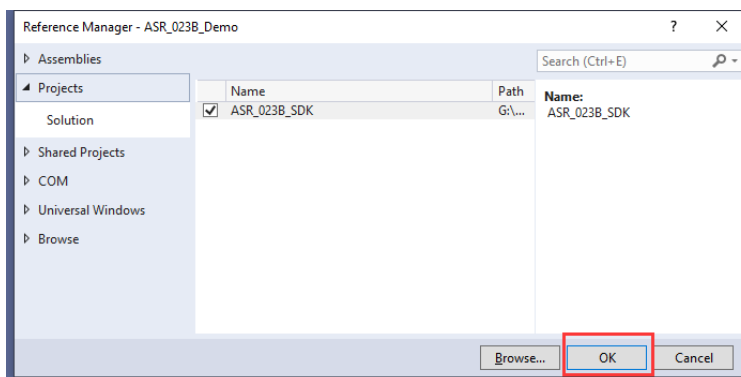


プロジェクトパスで SDK ファイルを選択し、「Add」 ボタンをクリックします。

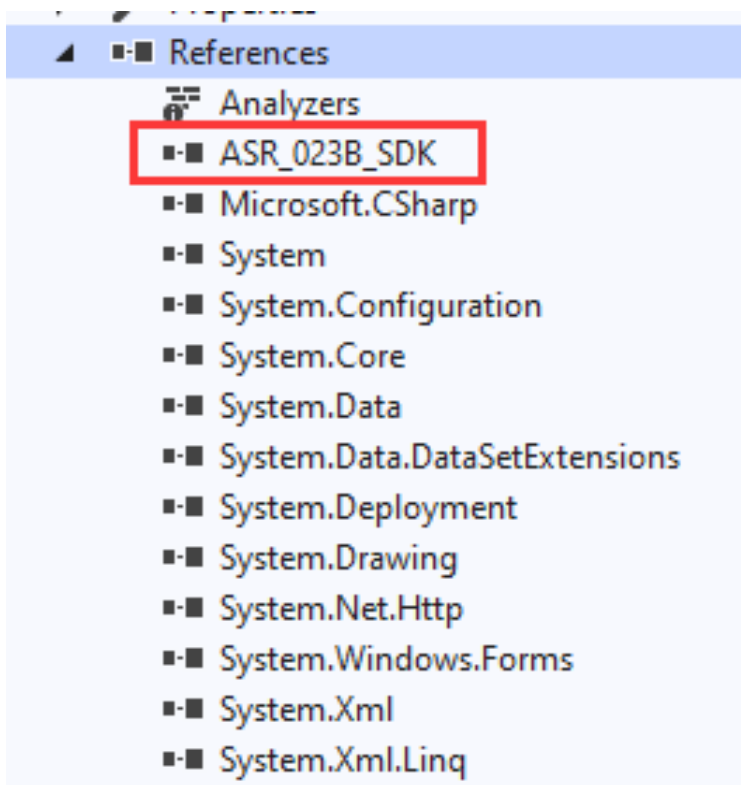




Reference Manager 画面の「OK」ボタンをクリックすると、SDK ファイルが追加されます。



追加完了後、下図通り：



## 1.2. SDK の使用

SDK を使用するために、デバイスの接続からバーコードのスキャンまでの一連のプロセスについて簡単に説明します。

### 1.2.1. 命名スペースをインポート

```
using ASR_023B_SDK;
```

### 1.2.2. デバイスの検索

#### 1. オブジェクトを取得

AsFingerManager クラスの「public static AsFingerManager SharedAsFingerManager ()」メソッドを呼び出して、AsFingerManager のシングルトンオブジェクトを取得します。

```
asFingerManager = AsFingerManager.SharedAsFingerManager();
```

#### 2. デリゲートの指定

(1) パラメータタイプは BluetoothDevice になるメソッドを作成します。このメソッドは、AsReader Finger-Type デバイスが検索されたときのコールバック処理を監視するために使用します。（以下はサンプルコードです。）

```
private void WhenReceivedBluetoothDevice(List<BluetoothDevice> devices)
{
    //It is to process the searched AsReader Finger-Type devices.
}
```

(2) WhenReceivedBluetoothDevice デリゲートを指定します。

作成したメソッドを AsFingerManager クラスの WhenReceivedBluetoothDevice デリゲートに指定します。

```
asFingerManager.whenReceivedBluetoothDevice += WhenReceivedBluetoothDevice;
```

### 3. 検索の実行

AsFingerManager クラスの「public void StartSearching (Types.BluetoothType type)」メソッドを呼び出して、AsReader Finger-Type デバイスを検索します。このメソッドを呼び出すには BluetoothType タイプのパラメーターを渡す必要があります。このパラメーターは、AsReader Finger-Type デバイスを検索時使用される Bluetooth モード (Bluetooth モード : SPP および BLE) を制限するために使用されます。AsReader Finger-Type デバイスが検索されると、[1.2.2-2-\(1\)](#)で作成されたメソッドがコールバックされます。

```
Types.BluetoothType bluetoothType = Types.BluetoothType.BluetoothBLE;
//Types.BluetoothType bluetoothType = Types.BluetoothType.BluetoothSPP;
asFingerManager.StartSearching(bluetoothType);
```

### 1.2.3. デバイスの検索を停止

AsFingerManager クラスの「public void StopSearching ()」メソッドを呼び出して、AsReader Finger-Type デバイスの検索を停止します。

```
asFingerManager.StopSearching();
```

### 1.2.4. デバイスを接続

#### 1. オブジェクトを取得

AsFinger クラスの「public static AsFinger SharedAsFinger()」メソッドを呼び出して、AsFinger のシングルトンオブジェクトを取得します。

```
asFinger = AsFinger.SharedAsFinger();
```

#### 2. デリゲートの指定

(1) パラメータタイプが bool のメソッドを作成します。

このメソッドは、AsReader Finger-Type デバイスの接続ステータスが変更されたときのコールバック処理に使用します。パラメータ isConnected が true の場合は接続されていることを意味し、false の場合は接続されていないことを意味します。(以下はサンプルコードです。)

```
private void WhenAsFingerIsConnected(bool isConnected)
{
    //It is to process the result of the connection being successful or not.
}
```

(2) whenAsFingerIsConnected デリゲートを指定します。

で作成したメソッドを AsFinger クラスの whenAsFingerIsConnected デリゲートに指定します。

```
asFinger.whenAsFingerIsConnected = WhenAsFingerIsConnected;
```

### 3. デバイスを接続

AsFinger クラスの「publicvoidConnect (BluetoothDevice device) 」メソッドを呼び出します。渡されるパラメーター値は、[1.2.2-2-\(1\)](#)で作成されたメソッドがコールバックされるときに渡される AsReaderFinger-Type デバイスオブジェクトリスト内の任意のデバイスオブジェクトです。AsReader Finger-Type デバイスを接続します。デバイスが正常に接続されると、[1.2.4-2-\(1\)](#)で作成されたメソッドがコールバックされます。この時点で、パラメーターisConnected に渡される値は true です。

```
BluetoothDevice device = DeviceDic[deviceID];  
asFinger.Connect(device);
```

## 1.2.5. バーコードのスキャン

### 1. デリゲートの指定

(1) パラメータタイプが `byte[]` のメソッドを作成します。

このメソッドは、AsReader Finger-Type デバイスがバーコードデータをスキャンされたときのコールバック処理に使用されます。(以下はサンプルコードです。)

```
private void WhenReceiveBarcodeData(byte[] data)
{
    //It is to process the scanned barcode data
}
```

(2) `whenReceivedBarcodeData` デリゲートを指定します。

作成したメソッドを `AsFinger` クラスの `whenReceivedBarcodeData` デリゲートに指定します。

```
asFinger.whenReceivedBarcodeData = WhenReceiveBarcodeData;
```

## 2. バーコードのスキャン

`AsFinger` クラスの「`public void StartScan()`」メソッドを呼び出して、バーコードをスキャンします。バーコードがスキャンされると、[1.2.5-1-\(1\)](#)で作成されたメソッドがコールバックされます。このとき、パラメーター `data` に渡される値は、スキャンされたバーコードデータです。

```
asFinger.StartScan();
```

### 1.2.6. バーコードのスキヤンを停止

AsFinger クラスの「public void StopScan()」メソッドを呼び出して、バーコードのスキヤンを停止します。

```
asFinger.StopScan();
```

### 1.2.7. 接続を切断

AsFinger クラスの「public void Disconnect()」メソッドを呼び出して、AsReader Finger-Type デバイスを切断します。デバイスが切断されると、[1.2.4-2-\(1\)](#)で作成されたメソッドがコールバックされます。この時点で、パラメーターisConnected に渡される値は false です。

```
asFinger.Disconnect();
```

## 2. AsFinger クラス

### 2.1. メソッド

#### 2.1.1. Connect

関数名	void Connect(BluetoothDevice device);			
パラメータ	パラメータ名	In/Out	タイプ	説明
	device	-	BluetoothDevice	AsReader Finger-Type の Mac アドレス、名称、および Bluetooth モードを含む BluetoothDevice オブジェクト
返り値	-	-	-	-

**説明:**  
AsReader Finger-Type を接続します。  
このメソッドを呼び出した後、AsFinger クラスの「WhenAsFingerIsConnected」 ([2.2.1](#) を参照) デリゲートを介して接続結果がコールバックされます。

**呼び出し条件:**  
AsReader Finger-Type を取得し、AsFingerManager クラスの「WhenReceivedBluetoothDevice」 ([3.2.1](#) を参照) より指定のデリゲート ([1.2.2-2-\(1\)](#) を参照) がコールバックされ、かつコールバックされるパラメータが空ではない場合に、このメソッドを呼び出せます。

**サンプルコード:** (注 : asFinger は AsFinger クラスのシングルトンオブジェクト)  
asFinger.Connect(device);



### 2.1.2. Disconnect

関数名	void Disconnect();			
パラメータ	パラメータ名	In/Out	タイプ	説明
	-	-	-	-
返り値	-	-	-	-
<p><b>説明:</b></p> <p>AsReader Finger-Type を切断します。</p> <p>このメソッドを呼び出した後、AsFinger クラスの「WhenAsFingerIsConnected」 (<a href="#">2.2.1</a> を参照) デリゲートを介して接続状態がコールバックされます。</p> <p><b>呼び出し条件:</b></p> <p>AsReader Finger-Type を接続した後、「WhenAsFingerIsConnected」 (<a href="#">2.2.1</a> を参照) より指定のデリゲート (<a href="#">1.2.4-2-(1)</a>を参照) がコールバックされ、かつコールバックされるパラメータ isConnected が「true」の場合に、このメソッドを呼び出せます。</p> <p><b>サンプルコード:</b> (注 : asFinger は AsFinger クラスのシングルトンオブジェクト)</p> <pre>asFinger.Disconnect();</pre>				

### 2.1.3. StartScan

関数名	void StartScan();			
パラメータ	パラメータ名	In/Out	タイプ	説明
	-	-	-	-
返り値	-	-	-	-
<p><b>説明:</b></p> <p>バーコードのスキャンを開始します。</p> <p>このメソッドを呼び出した後、AsFinger クラスの「WhenReceivedBarcodeData」 (<a href="#">2.2.2</a> を参照) を介してスキャンしたバーコードデータがコールバックされます。</p>				

**呼び出し条件：**

AsReader Finger-Type を接続した後、「WhenAsFingerIsConnected」(2.2.1を参照)より指定のデリゲート(1.2.4-2-(1)を参照)がコールバックされ、かつコールバックされるパラメータ isConnected が「true」の場合に、このメソッドを呼び出せます。

**サンプルコード：**(注：asFinger は AsFinger クラスのシングルトンオブジェクト)

```
asFinger.StartScan();
```

**2.1.4. StopScan**

関数名	void StopScan();			
参数	参数名	In/Out	类型	描述
	-	-	-	-
返回值	-	-	-	-

**説明：**  
バーコードのスキャンを停止します。

**呼び出し条件：**  
AsReader Finger-Type を接続した後、「WhenAsFingerIsConnected」(2.2.1を参照)より指定のデリゲート(1.2.4-2-(1)を参照)がコールバックされ、かつコールバックされるパラメータ isConnected が「true」の場合に、このメソッドを呼び出せます。

**サンプルコード：**(注：asFinger は AsFinger クラスのシングルトンオブジェクト)  
asFinger.StopScan();

### 2.1.5. GetBattery

関数名	void GetBattery();			
パラメータ	パラメータ名	In/Out	タイプ	説明
	-	-	-	-
返り値	-	-	-	-
<p><b>説明:</b></p> <p>現在接続されている AsReader Finger-Type デバイスのバッテリー残量を取得します。 このメソッドを呼び出した後、AsFinger クラスの「WhenReceivedBattery」 (<a href="#">2.2.3</a> を参照) を介して取得したバッテリー残量がコールバックされます。</p> <p><b>呼び出し条件:</b></p> <p>AsReader Finger-Type を接続した後、「WhenAsFingerIsConnected」 (<a href="#">2.2.1</a> を参照) より指定のデリゲート (<a href="#">1.2.4-2-(1)</a> を参照) がコールバックされ、かつコールバックされるパラメータ isConnected が「true」の場合に、このメソッドを呼び出せます。</p> <p><b>サンプルコード:</b> (注 : asFinger は AsFinger クラスのシングルトンオブジェクト)</p> <pre>asFinger.GetBattery();</pre>				

### 2.1.6. GetFirmwareVersion

関数名	void GetFirmwareVersion();			
パラメータ	パラメータ名	In/Out	タイプ	説明
	-	-	-	-
返り値	-	-	-	-
<p><b>説明:</b></p> <p>現在接続されている AsReader Finger-Type のファームウェアバージョン情報を取得します。 このメソッドを呼び出した後、AsFinger クラスの「WhenReceivedFirmwareVersion」 (を参照) を介して取得したファームウェアバージョン情報がコールバックされます。</p>				

**呼び出し条件 :**

AsReader Finger-Type を接続した後、「WhenAsFingerIsConnected」 ([2.2.1](#) を参照) より指定のデリゲート ([1.2.4-2-\(1\)](#)を参照) がコールバックされ、かつコールバックされるパラメータ isConnected が「true」の場合に、このメソッドを呼び出せます。

**サンプルコード:** (注 : asFinger は AsFinger クラスのシングルトンオブジェクト)

```
asFinger.GetFirmwareVersion();
```

**2.1.7. GetSDKVersion**

関数名	string GetSDKVersion();			
パラメータ	パラメータ名	In/Out	タイプ	説明
	-	-	-	-
返り値	-	Out	String	SDK バージョン
<p><b>説明:</b> SDK バージョンを取得します。</p> <p><b>呼び出し条件 :</b> なし</p> <p><b>サンプルコード:</b> (注 : asFinger は AsFinger クラスのシングルトンオブジェクト) string ver = asFinger.GetSDKVersion(); // ver: SDK バージョン</p>				

### 2.1.8. GetBluetoothDevice

関数名	BluetoothDevice GetBluetoothDevice();			
パラメータ	パラメータ名	In/Out	タイプ	説明
	-	-	-	-
返り値	-	Out	BluetoothDevice	BluetoothDevice オブジェクト ( <a href="#">4 BluetoothDevice クラス</a> を参照)
<p><b>説明:</b> 現在の AsFinger クラスの BluetoothDevice オブジェクトを返します。</p> <p><b>呼び出し条件:</b> なし</p> <p><b>サンプルコード:</b> (注 : asFinger は AsFinger クラスのシングルトンオブジェクト) asFinger.GetCurrentDevice();</p>				

### 2.1.9. SendData

関数名	void SendData(byte[]data);			
パラメータ	パラメータ名	In/Out	タイプ	説明
	data	In	byte[]	カスタムデータ
返り値	-	-	-	-
<p><b>説明:</b> ユーザーカスタムデータを送信します。 このメソッドを呼び出した後、AsFinger クラスの「WhenReceivedData」 (2.2.5 を参照) を介して取得したデータがコールバックされます。</p> <p><b>呼び出し条件:</b></p>				

AsReader Finger-Type を接続した後、「WhenAsFingerIsConnected」(2.2.1 を参照) より指定のデリゲート (1.2.4-2-(1)を参照) がコールバックされ、かつコールバックされるパラメータ isConnected が「true」の場合に、このメソッドを呼び出せます。

**サンプルコード:** (注 : asFinger は AsFinger クラスのシングルトンオブジェクト)

```
string data = @"%%FIRM-VER";
asFinger.SendData(System.Text.UTF8Encoding.ASCII.GetBytes(data));
```

## 2.2. デリゲート

### 2.2.1. WhenAsFingerIsConnected

デリゲート名	delegate void WhenAsFingerIsConnected(bool isConnected);			
パラメータ	パラメータ名	In/Out	タイプ	説明
	isConnected	In	Bool	AsReader Finger-Type の接続状態 : true : 接続済 false : 未接続
返り値	-	-	-	-
<p><b>説明:</b> AsReader Finger-Type の接続状態を監視します。</p> <p><b>コールバック条件 :</b></p> <ol style="list-style-type: none"> <li>Connect ( を参照) と Disconnect ( を参照) メソッドを呼び出した後、このデリゲートをコールバックします。</li> <li>AsReader Finger-Type との接続状態が変更されると、このデリゲートがコールバックされます。</li> </ol> <p><b>サンプルコード:</b> WhenAsFingerIsConnected whenAsFingerIsConnected = WhenAsFingerIsConnectedProc;</p>				

```

void WhenAsFingerIsConnectedProc(bool isConnected){
    if (isConnected) {
        //接続済の処理
    } else {
        //未接続の処理
    }
}
}
whenAsFingerIsConnected(isConnected);
//isConnected : 接続状態。true : 接続済 ; false : 未接続

```

### 2.2.2. WhenReceivedBarcodeData

デリゲート名	delegate void WhenReceivedBarcodeData(byte[] data);			
パラメータ	パラメータ名	In/Out	タイプ	説明
	data	In	byte[]	AsReader Finger-Type によって読み取られたバーコードデータ
返り値	-	-	-	-

#### 説明:

AsReader Finger-Type がバーコードを読み取るときに返されるデータを監視します。

#### コールバック条件 :

1. 「StartScan」 (2.1.3 を参照) メソッドを呼び出すか、AsReader Finger-Type のトリガーボタンを押して、スキャンが成功したら、このデリゲートをコールバックします。

#### サンプルコード:

```

WhenReceivedBarcodeData whenReceivedBarcodeData =
WhenReceivedBarcodeDataProc ;
void WhenReceivedBarcodeDataProc(byte[] data){
    //AsReader Finger-Type によって読み取られたバーコードデータを処理します。
}

```

```
whenReceivedBarcodeData(data); //data : AsReader Finger-Type によって読み取られたバ
ーコードデータ
```

### 2.2.3. WhenReceivedBattery

デリゲート名	delegate void WhenReceivedBattery(int battery);			
パラメータ	パラメータ名	In/Out	タイプ	説明
	battery	In	Int	AsReader Finger-Type のバッテリ残量
返り値	-	-	-	-

**説明:**

AsReader Finger-Type から返されるバッテリー値を監視します。

**コールバック条件:**

1. GetBattery (を参照) メソッドを呼び出した後、このデリゲートをコールバックします。

**サンプルコード:**

```
WhenReceivedBattery whenReceivedBattery = WhenReceivedBatteryProc;
void WhenReceivedBatteryProc(int battery) {
    //取得した AsReader Finger-Type のバッテリー残量を処理します。
}
whenReceivedBattery(battery); //battery : 取得した AsReader Finger-Type のバッテリー残量
```



## 2.2.4. WhenReceivedFirmwareVersion

デリゲート名	delegate void WhenReceivedFirmwareVersion(string ver);			
パラメータ	パラメータ名	In/Out	タイプ	説明
	ver	In	String	AsReader Finger-Type のファームウェアバージョン情報
返り値	-	-	-	-
<p><b>説明:</b> AsReader Finger-Type から返されるファームウェアバージョン情報を監視します。</p> <p><b>コールバック条件 :</b></p> <ol style="list-style-type: none"> <li>1. GetFirmwareVersion (を参照) メソッドを呼び出した後、このデリゲートをコールバックします。</li> </ol> <p><b>サンプルコード:</b></p> <pre> WhenReceivedFirmwareVersion whenReceivedFirmwareVersion = WhenReceivedFirmwareVersionProc; void WhenReceivedFirmwareVersionProc(string ver){     //取得した AsReader Finger-Type のファームウェアバージョン情報を処理します。 } whenReceivedFirmwareVersion(ver); //ver : 取得した AsReader Finger-Type のファームウェアバージョン情報 </pre>				

## 2.2.5. WhenReceivedData

デリゲート名	delegate void WhenReceivedData(byte[] data);			
パラメータ	パラメータ名	In/Out	タイプ	説明
	data	In	byte[]	AsReader Finger-Type から受信したデータ

返り値	-	-	-	-
<p><b>説明:</b></p> <p>AsReader Finger-Type から返されるデータを監視します。</p> <p><b>コールバック条件 :</b></p> <ol style="list-style-type: none"><li>1. 「StartScan」 (<a href="#">2.1.3</a> を参照) メソッドを呼び出すか、AsReader Finger-Type のトリガーボタンを押して、スキャンが成功したら、このデリゲートをコールバックします。</li><li>2. 「GetBattery」 (<a href="#">2.1.5</a> を参照) メソッドを呼び出した後、このデリゲートをコールバックします。</li><li>3. 「GetFirmwareVersion」 (<a href="#">2.1.6</a> を参照) メソッドを呼び出した後、このデリゲートをコールバックします。</li><li>4. 「SendData」 (<a href="#">を参照</a>) メソッドを呼び出した後、このデリゲートをコールバックします。</li></ol> <p><b>サンプルコード:</b></p> <pre>WhenReceivedData whenReceivedData = ReceivedDataProc; void ReceivedDataProc(byte[] data){     //AsReader Finger-Type から受信したデータを処理します。 } whenReceivedData(data);// data : AsReader Finger-Type から受信したデータ</pre>				

## 3. AsFingerManager クラス

### 3.1. メソッド

#### 3.1.1. StartSearching

関数名	void StartSearching(Types.BluetoothType type);			
パラメータ	パラメータ名	In/Out	タイプ	説明
	type	In	Types.BluetoothType	Bluetooth モード : BluetoothBLE :BLE モード BluetoothSPP :SPP モード
返り値	-	-	-	-
<p><b>説明:</b> AsReader Finger-Type を検索します。 このメソッドを呼び出した後、AsFingerManager クラスの「WhenReceivedBluetoothDevice」(を参照) デリゲートを介して検索した AsReader Finger-Type デバイスをコールバックされます。</p> <p><b>呼び出し条件 :</b> システムの Bluetooth がオンになっています。</p> <p><b>サンプルコード:</b> (注 : asFingerManager は AsFingerManager クラスのシングルトンオブジェクト) asFingerManager.StartSearching(Types.BluetoothType.BluetoothBLE);</p>				

#### 3.1.2. StopSearching

関数名	void StopSearching();
-----	-----------------------

パラメータ	パラメータ名	In/Out	タイプ	説明
	-	-	-	-
返り値	-	-	-	-

**説明:**  
AsReader Finger-Type の検索を停止します。

**呼び出し条件 :**

1. システムの Bluetooth がオンになっています。
2. デバイスを検索しています。つまり StartSearching (を参照) メソッドが呼び出されます。

**サンプルコード:** (注 : asFingerManager は AsFingerManager クラスのシングルトンオブジェクト)

```
asFingerManager.StopSearching();
```

### 3.1.3. GetSearchBluetoothType

関数名	Types.BluetoothType GetSearchBluetoothType();			
パラメータ	パラメータ名	In/Out	タイプ	説明
	-	-	-	-
返り値	-	Out	Types.BluetoothType	AsReader Finger-Type を検索するとき使用される Bluetooth モード

**説明:**  
AsReader Finger-Type を検索するとき使用される Bluetooth モードを取得します。

**呼び出し条件 :**  
なし

**サンプルコード:** (注 : asFingerManager は AsFingerManager クラスのシングルトンオブジェクト)

```
asFingerManager.GetSearchBluetoothType();
```

## 3.2. デリゲート

### 3.2.1. WhenReceivedBluetoothDevice

デリゲート名	delegate void WhenReceivedBluetoothDevice(List<BluetoothDevice> devices);			
パラメータ	パラメータ名	In/Out	タイプ	説明
	devices	In	List<BluetoothDevice>	検索した AsReader Finger-Type デバイスオブジェクトリスト
返り値	-	-	-	-
<p><b>説明:</b> システム Bluetooth がデバイスを検索したときに返される AsReader Finger-Type デバイスオブジェクトリストを監視します。</p> <p><b>コールバック条件 :</b> システムの Bluetooth がオンになっています。「StartSearching」 (3.1.1 を参照) メソッドを呼び出し、AsReader Finger-Type を検索されると、このデリゲートをコールバックします。</p> <p><b>サンプルコード:</b>  <pre>WhenReceivedBluetoothDevice whenReceivedBluetoothDevice = WhenReceivedBluetoothDeviceProc; void WhenReceivedBluetoothDeviceProc(List&lt;BluetoothDevice&gt; devices) {     //検索した AsReader Finger-Type デバイスオブジェクトリストを処理します。</pre> </p>				

```
}
```

```
whenReceivedBluetoothDevice(devices); // devices : 検索した AsReader Finger-Type デバイ  
スオブジェクトリスト
```

## 4. BluetoothDevice クラス

### 4.1. プロパティ

プロパティ名	プロパティ	タイプ	説明
DeviceBluetoothType	get/set	Types.BluetoothType	AsReader Finger-Type で使用される Bluetooth モード
DeviceName	get/set	String	AsReader Finger-Type デバイス名
DeviceID	get/set	String	AsReader Finger-Type の Mac アドレス

## 5. Types クラス

### 5.1. 列挙型

#### 5.1.1. BluetoothType

定義	説明
BluetoothBLE	BLE Bluetooth モード
BluetoothSPP	SPP Bluetooth モード