

**AsReader**<sup>®</sup>

# AsReaderP252B SDK

**Objective-C SDK 開発マニュアル**

## 修正履歴

No.	バージョン	修正内容	日付
1	1.0	新規作成	2023/03/31

# 目次

<b>1</b>	<b>開発環境の構築</b>	<b>5</b>
1.1.	SDK の追加	5
1.2.	ヘッダーファイルのインポート	7
1.3.	SDK の使用	8
1.3.1.	AsReaderP252BdeviceManager オブジェクトの作成・初期化。(シングルトン)	8
1.3.2.	デリゲートの設定	8
1.3.3.	検索開始	8
1.3.4.	AsReaderP252B オブジェクトへの接続	8
1.3.5.	タグのインベントリを開始	9
1.3.6.	タグのインベントリを停止	9
1.3.7.	Barcode スキャンの開始	10
1.3.8.	Barcode スキャンを停止	10
1.3.9.	AsReaderP252B オブジェクトとの接続を切断	11
<b>2</b>	<b>AsReaderP252B クラス</b>	<b>12</b>
2.1.	属性	12
2.2.	メソッド	13
2.2.1.	setDelegate	13
2.2.2.	batteryStatus	14
2.2.3.	regionName	14
2.2.4.	serialNumber	14
2.2.5.	rFModuleVersion	15
2.2.6.	hardwareVersion	15
2.2.7.	firmwareVersion	16
2.2.8.	getMessageWithAsReaderP252BResultCode	16
2.2.9.	getAction	16
2.2.10.	powerGainRange	17
2.2.11.	inventory	17
2.2.12.	inventoryHumidityTag	18
2.2.13.	inventoryTemperatureTagWithType	18
2.2.14.	stop	19
2.2.15.	stopSync	20

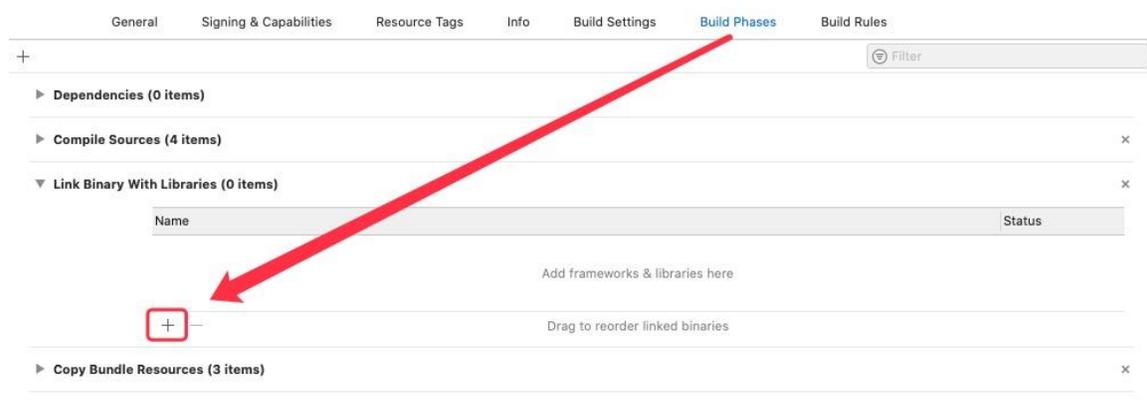
2.2.16. readMemory .....	20
2.2.17. writeMemory.....	21
2.2.18. lock .....	22
2.2.19. unlock .....	23
2.2.20. permaLock.....	23
2.2.21. kill .....	24
2.2.22. defaultParameter .....	25
2.2.23. saveParameter .....	25
2.2.24. startDecode .....	26
2.2.25. setBarcodeTimeOut .....	26
2.2.26. getBarcodeTimeOut .....	27
2.2.27. epcMaskCount.....	27
2.2.28. addEpcMask.....	28
2.2.29. addEpcMask.....	28
2.2.30. getEpcMask .....	29
2.2.31. clearEpcMask .....	29
2.2.32. setFrequencyAutomatic .....	30
2.2.33. getFrequencyAutomatic .....	30
2.2.34. setLBT .....	31
2.2.35. getLBT .....	31
2.2.36. setScanMode.....	32
2.2.37. getBaudRateList .....	32
2.2.38. setBaudRate .....	32
2.2.39. getBaudRate .....	33
<b>3 AsReaderP252BDelegate クラス.....</b>	<b>34</b>
3.1. whenAsReaderP252BIsConnected:.....	34
3.2. updateAsReaderP252BDeviceState: .....	34
3.3. readAsReaderP252BTag: .....	35
3.4. changedAsReaderP252BActionState: .....	36
3.5. accessAsReaderP252BResult: .....	36
3.6. readAsReaderP252BTemperatureTag: .....	38
3.7. readAsReaderP252BHumidityTag:.....	38
3.8. onAsReaderP252BTriggerKeyEvent: .....	39
3.9. onAsReaderP252BModeKeyEvent:.....	40
3.10. whenAsReaderP252BReceivedData: .....	40
3.11. whenAsReaderP252BWriteData: .....	41
3.12. detectBarcode: .....	41

<b>4</b>	<b>AsReaderP252BBarcode クラス</b> .....	<b>43</b>
4.1.	メソッド.....	43
4.1.1.	getBarcodeString.....	43
4.2.	列挙.....	43
4.2.1.	AsReaderP252BBarcodeType .....	43
<b>5</b>	<b>AsReaderP252BDeviceManager</b> .....	<b>47</b>
5.1.	属性.....	47
5.2.	メソッド.....	47
5.2.1.	shared .....	47
5.2.2.	startDiscovered .....	47
5.2.3.	stopDiscovered .....	48
5.3.	デリゲート.....	48
5.3.1.	didDiscoverAsReaderP252BDeviceList .....	48
<b>6</b>	<b>AsReaderP252BDevice</b> .....	<b>49</b>
6.1.	属性.....	49
6.2.	メソッド.....	49
6.2.1.	connect.....	49
6.2.2.	disconnect.....	50
<b>7</b>	<b>AsReaderP252BPacket</b> .....	<b>51</b>
7.1.	列挙.....	51
<b>8</b>	<b>AsReaderP252BRfidValues</b> .....	<b>53</b>
8.1.	AsReaderP252BLockParam .....	53
8.1.1.	属性.....	53
8.2.	AsReaderP252BSelectMaskEPCParam .....	53
8.2.1.	属性.....	53
8.3.	AsReaderP252BLbtItem.....	54
8.3.1.	属性.....	54
8.4.	列挙.....	54
8.5.	構造体.....	56

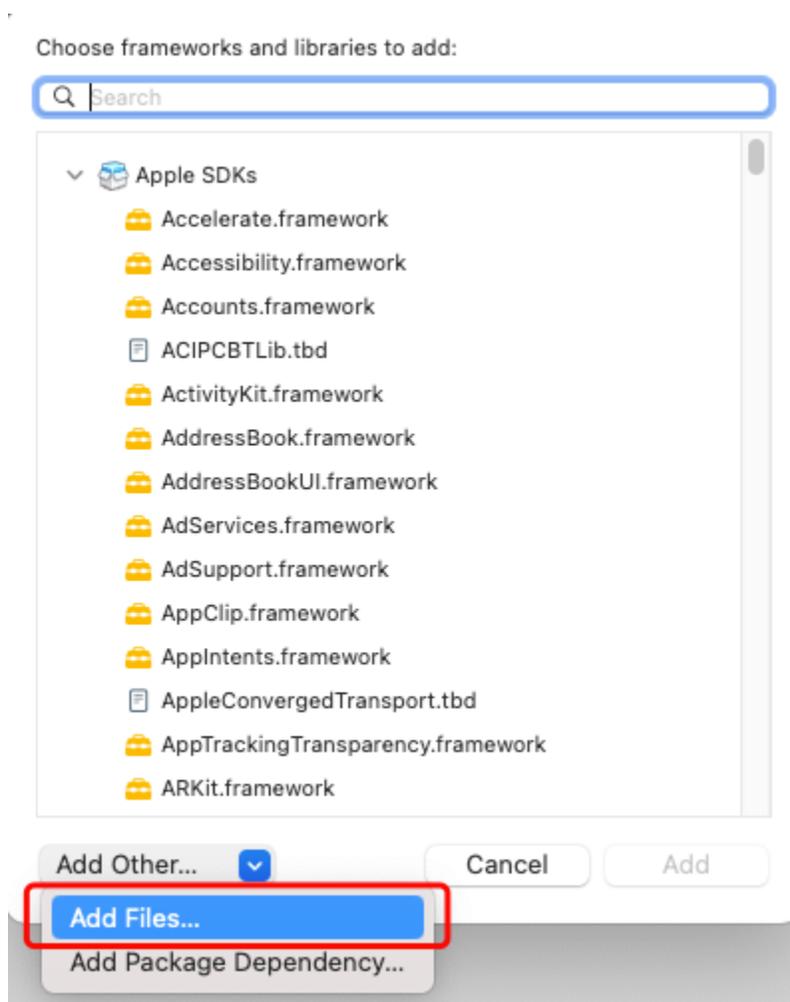
# 1 開発環境の構築

## 1.1. SDK の追加

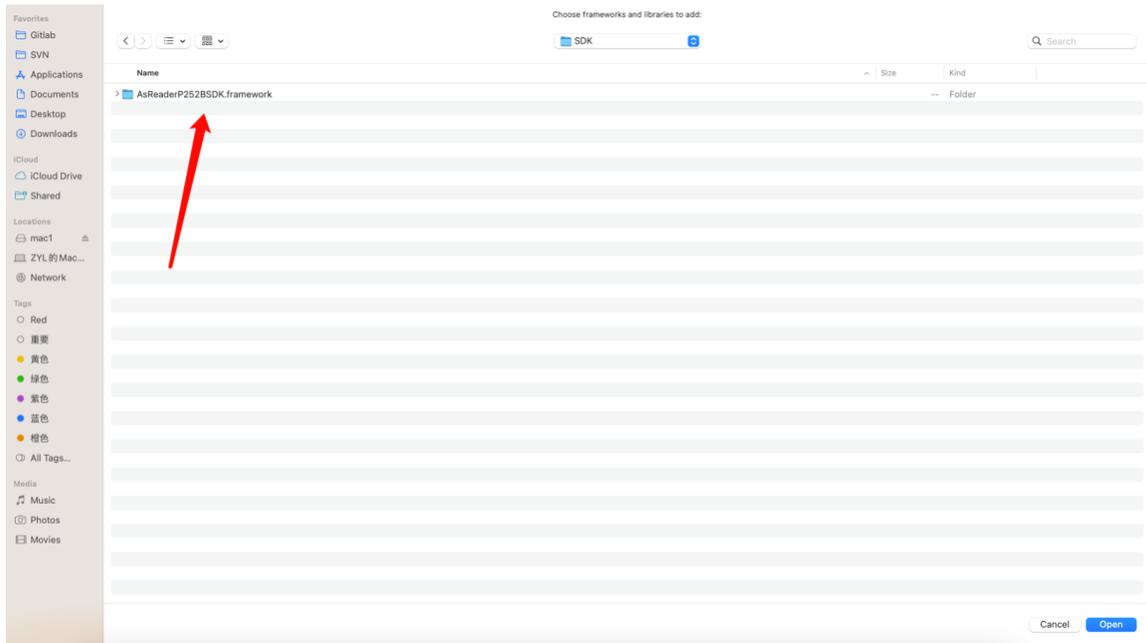
### 1. TARGET -> Build phases -> Link Binary with Libraries



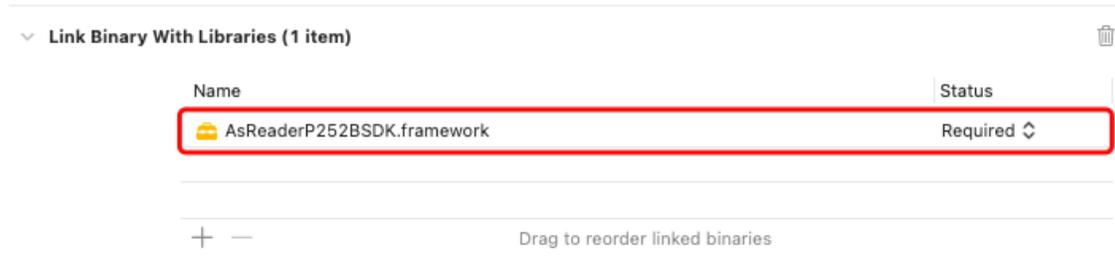
## 2. 「Add Other…」、「Add Files…」を選択



### 3. AsReaderP252BSDK.framework を追加



### 4. 追加後、下記のように表示されます。



## 1.2. ヘッダーファイルのインポート

Objective-C プロジェクトの SDK を使うクラスにヘッダーファイルをインポートする必要があります。

以下は例としてご参照ください。

```
#import <AsReaderP252BSDK/AsReaderP252BSDK.h>
```

## 1.3. SDK の使用

### 1.3.1. AsReaderP252BdeviceManager オブジェクトの作成・初期化。(シングルトン)

```
AsReaderP252BDeviceManager *bluetoothManager = [AsReaderP252BDeviceManager shared];
```

### 1.3.2. デリゲートの設定

```
bluetoothManager.delegate = self;
```

### 1.3.3. 検索開始

```
[bluetoothManager startDiscovered];
```

検索した AsReaderP252B オブジェクトの配列を受け取ります。

```
- (void)didDiscoverAsReaderP252BDeviceList:(NSArray<AsReaderP252B *> *)asReaderP252BList{  
    // asReaderP252BList の要素は AsReaderP252B オブジェクトの配列  
}
```

### 1.3.4. AsReaderP252B オブジェクトへの接続

asReaderP252BList より AsReaderP252B を取得して、接続を行います。

AsReaderP252B オブジェクトに接続するメソッド

```
[asReaderP252B connect];
```

接続結果を受信するデリゲートメソッド

```
-(void)whenAsReaderP252BIsConnected:(BOOL)isConnected{  
    if (isConnected){  
        //接続  
    } else {  
        //切断  
    }  
}
```

### 1.3.5. タグのインベントリを開始

#### 1.3.5.1. inventory

```
AsReaderP252BResultCode resultCode = [asReaderP252B inventory];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}
```

##### 1.3.5.1.1. インベントリの実行結果を受信するデリゲート

```
-(void)changedAsReaderP252BActionState:(AsReaderP252BCommandType)action
resultCode:(NSInteger)resultCode{
    //action    AsReaderP252BCommandType(7.1.1 を参照)。
    //resultCode AsReaderP252BResultCode(8.4.1 を参照)。
}
```

##### 1.3.5.1.2. インベントリしたデータを受信するデリゲート

```
-(void)readAsReaderP252BTag:(NSString *)tag
                        rssi:(float)rssi
                        phase:(float)phase
                        frequency:(float)frequency{
    //tag        タグの PCEPC 値（16 進数の文字列）。
    //rssi       タグの RSSI 値
    //phase     タグの Phase 値
    //frequency  タグの周波数
}
```

### 1.3.6. タグのインベントリを停止

```
AsReaderP252BResultCode resultCode = [asReaderP252B stop];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}
```

```
}  
インベントリ停止を受信するデリゲート  
-(void)changedAsReaderP252BActionState:(AsReaderP252BCommandType)action  
resultCode:(NSInteger)resultCode{  
    //action    AsReaderP252BCommandType(7.1.1 を参照)。  
    //resultCode AsReaderP252BResultCode(8.4.1 を参照)。  
}
```

## 1.3.7. Barcode スキャンの開始

### 1.3.7.1. startDecode

```
AsReaderP252BResultCode resultCode = [asReaderP252B startDecode];  
if (resultCode == AsReaderP252BResultNoError) {  
    //実行成功時の処理  
}else{  
    //実行失敗時の処理  
}
```

#### 1.3.7.1.1. スキャンしたバーコードデータを受信するデリゲート

```
-(void)detectBarcode:(AsReaderP252BBarcodeType)barcodeType  
    barcodeData:(NSData *)barcodeData{  
    //barcodeType AsReaderP252BBarcodeType(4.2.1 を参照)  
    //barcodeData バーコードデータ  
}
```

## 1.3.8. Barcode スキャンの停止

```
AsReaderP252BResultCode resultCode = [asReaderP252B stop];  
if (resultCode == AsReaderP252BResultNoError) {  
    //実行成功時の処理  
}else{  
    //実行失敗時の処理  
}
```

### 1.3.9. AsReaderP252B オブジェクトとの接続を切断

```
[asReaderP252B disconnect];  
//切断の結果を受信するデリゲート  
-(void)whenAsReaderP252BIsConnected:(BOOL)isConnected{  
    if (isConnected){  
        //接続  
    } else {  
        //切断  
    }  
}
```

## 2 AsReaderP252B クラス

AsReaderP252B オブジェクトは AsReader に対して、接続、切断、設定情報、ステータス取得を行います。RF タグに対してのインベントリ、読取、書き込み、ロックなどを行います。バーコードに対して読取、読取停止、バーコード設定などを行います。

### 2.1. 属性

属性名	属性	型	説明
minQ	nonatomic, assign	int	RFID インベントリ時の最小 Q 値 値の範囲 : 0~15
maxQ	nonatomic, assign	int	RFID インベントリ時の最大 Q 値 値の範囲 : 0~15
qValue	nonatomic, assign	int	RFID インベントリ時の現在の Q 値 値の範囲 : 0~15
idleTime	nonatomic, assign	int	RFID インベントリ時の電波出力停止時間 (単位 : ms)
powerGain	nonatomic, assign	int	RFID インベントリ時の電波出力値
sleepTime	nonatomic, assign	int	AsReader がスリープするまでの時間 (単位 : ms)
autoOffTime	nonatomic, assign, readonly	int	AsReader 未接続時に電源が自動でオフになるま での時間 (単位 : ms)
operationTime	nonatomic, assign	int	一回の RFID インベントリの継続時間 (単位 : ms)
inventoryTime	nonatomic, assign	int	RFID インベントリ時の電波出力時間 (単位 : ms)
linkProfileValue	nonatomic, assign	int	RFID インベントリ時の Link Profile 値 値の範囲 : 0~3
rssMode	nonatomic, assign,	BOOL	RFID インベントリ時の RSSI データの取得要否 YES : 取得する

	readonly		NO : 取得しない
continuousMode	nonatomic, assign	BOOL	RFID インベントリの連続実行要否。 YES : 連続実行有効 NO : 連続実行無効 <b>注 : AsReader 起動時は本設定は NO です。必要に応じて APP から都度 ON に設定してください。</b>
serialNumber	nonatomic, strong	NSString	AsReader のシリアル番号
accessPassword	nonatomic, strong	NSString	対象のタグのアクセスパスワード
buzzer	nonatomic, assign	AsReaderP 252BBuzz erState	AsReader のブザーの設定 AsReaderP252BBuzzerState(8.4.4 を参照)
sessionFlag	nonatomic, assign	AsReaderP 252BSessi onFlag	RFID インベントリ時の Session Flag AsReaderP252BSessionFlag(8.4.6 を参照)
inventorySession	nonatomic, assign	AsReaderP 252BSessi onType	RFID インベントリ時の Session 値 AsReaderP252BSessionType(8.4.5 を参照)

## 2.2. メソッド

### 2.2.1. setDelegate

<b>- (void)setDelegate:(id&lt;AsReaderP252BDelegate&gt;)delegate;</b>				
	<b>引数名</b>	<b>In/Out</b>	<b>型</b>	<b>説明</b>
<b>引数</b>	delegat e	In	AsReaderP252BDelegat e	AsReaderP252B のデリゲート (3 を参照)
<b>メソッドの説明:</b> AsReaderP252B のデリゲートを設定します。				
<b>サンプルコード:</b>				

```
[asReaderP252B setDelegate:self];
```

### 2.2.2. batteryStatus

- (int)batteryStatus;				
	引数名	In/Out	型	説明
戻り値	-	Out	int	電池残量 値の範囲 : 0、1、2、3、4 0: 0 1: 25% 2: 50% 3: 75% 4: 100%
<p><b>メソッドの説明:</b> AsReader の電池残量を取得します。</p> <p><b>サンプルコード:</b> int battery = [asReaderP252B batteryStatus];</p>				

### 2.2.3. regionName

- (NSString *)regionName;				
	引数名	In/Out	型	説明
戻り値	-	Out	NSString	地域名
<p><b>メソッドの説明:</b> AsReader を適用される地域情報を取得します。</p> <p><b>サンプルコード:</b> NSString *regionName = [asReaderP252B regionName];</p>				

### 2.2.4. serialNumber

```
- (NSString *)serialNumber;
```

	引数名	In/Out	型	説明
戻り値	-	Out	NSString	シリアル番号
<b>メソッドの説明:</b> AsReader のシリアル番号を取得します。				
<b>サンプルコード:</b> <pre>NSString *serialNumber = [asReaderP252B serialNumber];</pre>				

### 2.2.5. rFModuleVersion

- (NSString *)rFModuleVersion;				
	引数名	In/Out	型	説明
戻り値	-	Out	NSString	RFID モジュールのバージョン
<b>メソッドの説明:</b> AsReader の RFID モジュールバージョンを取得します。				
<b>サンプルコード:</b> <pre>NSString *rFModuleVersion = [asReaderP252B rFModuleVersion];</pre>				

### 2.2.6. hardwareVersion

- (NSString *)hardwareVersion;				
	引数名	In/Out	型	説明
戻り値	-	Out	NSString	ハードウェアのバージョン
<b>メソッドの説明:</b> AsReader のハードウェアのバージョンを取得します。				
<b>サンプルコード:</b> <pre>NSString *hardwareVersion = [asReaderP252B hardwareVersion];</pre>				

### 2.2.7. firmwareVersion

- (NSString *)firmwareVersion;				
	引数名	In/Out	型	説明
戻り値	-	Out	NSString	ファームウェアのバージョン
<p><b>メソッドの説明:</b> AsReader のファームウェアのバージョンを取得します。</p> <p><b>サンプルコード:</b>  <pre>NSString *firmwareVersion = [asReaderP252B firmwareVersion];</pre> </p>				

### 2.2.8. getMessageWithAsReaderP252BResultCode

-(NSString*)getMessageWithAsReaderP252BResultCode:(AsReaderP252BResultCode)code;				
	引数名	In/Out	型	説明
引数	code	In	AsReaderP252BResultCode	AsReaderP252BResultCode(8.4.1を参照)
戻り値	-	Out	NSString	エラー内容
<p><b>メソッドの説明:</b> エラーコードに相応するエラー内容を返します。</p> <p><b>サンプルコード:</b>  <pre>AsReaderP252BResultCode error; // SDK から返したエラーコード NSString *errorString = [asReaderP252B getMessageWithAsReaderP252BResultCode:error];</pre> </p>				

### 2.2.9. getAction

- (AsReaderP252BCommandType)getAction;				
	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BCommandType	アクションコマンド

			e	AsReaderP252BCommandType(7.1.1 を参照)
<p><b>メソッドの説明:</b> AsReader のアクションを取得します。</p> <p><b>サンプルコード:</b> AsReaderP252BCommandType type = [asReaderP252B getAction];</p>				

### 2.2.10. powerGainRange

- (AsReaderP252BMinMaxValue)powerGainRange;				
	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BMinMaxValue	電波出力範囲 AsReaderP252BMinMaxValue (8.5.1 を参照)
<p><b>メソッドの説明:</b> AsReader の電波出力の範囲を取得します。</p> <p><b>サンプルコード:</b> int min = asReaderP252B.powerGainRange.min; //設定可能な最小出力値 int max = asReaderP252B.powerGainRange.max; //設定可能な最大出力値</p>				

### 2.2.11. inventory

- (AsReaderP252BResultCode)inventory;				
	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1 を参照)
<p><b>メソッドの説明:</b> RF タグのインベントリを開始します。 このメソッドを呼び出すと、デリゲート「readAsReaderP252BTag」(3.3 を参照)、 「changedAsReaderP252BActionState:」(3.4 を参照) がコールバックされます。</p>				

**サンプルコード:**

```

AsReaderP252BResultCode resultCode = [asReaderP252B inventory];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}

```

**2.2.12. inventoryHumidityTag**

- (AsReaderP252BResultCode)inventoryHumidityTag;				
	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1を参照)
<p><b>メソッドの説明:</b> 湿度タグのインベントリを開始します。 このメソッドを呼び出すと、デリゲート「changedAsReaderP252BActionState:」(3.4を参照)、 「readAsReaderP252BHumidityTag」(3.7を参照)がコールバックされます。</p> <p><b>サンプルコード:</b> AsReaderP252BResultCode resultCode = [asReaderP252B inventoryHumidityTag]; if (resultCode == AsReaderP252BResultNoError) {     //実行成功時の処理 }else{     //実行失敗時の処理 }</p>				

**2.2.13. inventoryTemperatureTagWithType**

- (AsReaderP252BResultCode)inventoryTemperatureTagWithType:(AsReaderP252BTemperatureTagType)type;				
	引数名	In/Out	型	説明
<p>-</p>				

<b>引数</b>	type	In	AsReaderP252BTemperatureTagType	温度タグのタイプ AsReaderP252BTemperatureTagType(8.4.3を参照)
<b>戻り値</b>	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1を参照)

**メソッドの説明:**  
温度タグのインベントリを開始します。  
このメソッドを呼び出すと、デリゲート「changedAsReaderP252BActionState:」(3.4を参照)、  
「readAsReaderP252BTemperatureTag」(3.6を参照)がコールバックされます。

**サンプルコード:**

```
AsReaderP252BResultCode resultCode = [asReaderP252B
inventoryTemperatureTagWithType:AsReaderP252BTemperatureTagType_0];
// Magnus-S3 タイプの温度タグをインベントリ
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}
```

## 2.2.14. stop

<b>- (AsReaderP252BResultCode)stop;</b>				
	<b>引数名</b>	<b>In/Out</b>	<b>型</b>	<b>説明</b>
<b>戻り値</b>	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1を参照)

**メソッドの説明:**  
RF タグ/バーコードのスキャンを停止します。  
本メソッドは AsReader での実行結果を待たずに実行結果を返します。  
このメソッドを呼び出すと、デリゲート「changedAsReaderP252BActionState:」(3.4を参照)がコールバックされます。

**サンプルコード:**

```

AsReaderP252BResultCode resultCode = [asReaderP252B stop];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}

```

### 2.2.15. stopSync

- (AsReaderP252BResultCode)stopSync;

	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1を参照)

**メソッドの説明:**

RF タグ/バーコードのスキャンを停止します。

本メソッドを実行後に、AsReader の実行結果を確認して、本メソッドの実行結果を返します。

このメソッドを呼び出すと、デリゲート「changedAsReaderP252BActionState:」(3.4を参照) がコールバックされます。

**サンプルコード:**

```

AsReaderP252BResultCode resultCode = [asReaderP252B stopSync];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}

```

### 2.2.16. readMemory

-(AsReaderP252BResultCode)readMemory:(AsReaderP252BMemoryBank)bank  
offset:(int)offset length:(int)length;

	引数名	In/Out	型	説明
引数	bank	In	AsReaderP252B	読取対象タグのメモリーバンク

			MemoryBank	AsReaderP252BMemoryBank(8.4.2を参照)
引数	offset	In	int	読取メモリのオフセット 単位：word
引数	length	In	int	読取メモリの長さ 単位：word
戻り値	-	Out	AsReaderP252B ResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1を参照)

**メソッドの説明:**

タグの対象メモリーバンクのデータを読取します。

このメソッドを呼び出すと、デリゲート「accessAsReaderP252BResult:」(3.5を参照)がコールバックされません。

**サンプルコード:**

```
AsReaderP252BResultCode resultCode = [asReaderP252B
readMemory:AsReaderP252BBank_EPC offset:2 length:4];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}
```

**2.2.17. writeMemory**

<b>-(AsReaderP252BResultCode)writeMemory:(AsReaderP252BMemoryBank)bank offset:(int)offset value:(NSString *)value;</b>				
	引数名	In/Out	型	説明
引数	bank	In	AsReaderP252BMemoryBank	書き込み対象タグのメモリーバンク AsReaderP252BMemoryBank(8.4.2を参照)
引数	offset	In	int	書き込みメモリのオフセット 単位：word
引数	value	In	NSString	書き込みデータ (16進数の文字列)

戻り値	-	Out	AsReaderP252BResult Code	メソッドの実行結果 AsReaderP252BResultCode(8.4.1を参照)
<p><b>メソッドの説明:</b>            タグの対象メモリーバンクにデータを書き込みます。            このメソッドを呼び出すと、デリゲート「accessAsReaderP252BResult:」(3.5を参照)がコールバックされます。</p> <p><b>サンプルコード:</b></p> <pre>AsReaderP252BResultCode resultCode = [asReaderP252B writeMemory:AsReaderP252BBank_EPC offset:2 value:@"1234"]; if (resultCode == AsReaderP252BResultNoError) {     //実行成功時の処理 }else{     //実行失敗時の処理 }</pre>				

## 2.2.18. lock

<b>- (AsReaderP252BResultCode)lock:(AsReaderP252BLockParam *)param;</b>				
	引数名	In/Out	型	説明
引数	param	In	AsReaderP252BLockParam	AsReaderP252BLockParam オブジェクト (8.1を参照)
戻り値	-	Out	AsReaderP252BResult Code	メソッドの実行結果 AsReaderP252BResultCode(8.4.1を参照)
<p><b>メソッドの説明:</b>            タグの対象メモリーバンクをロックします。            このメソッドを呼び出すと、デリゲート「accessAsReaderP252BResult:」(3.5を参照)がコールバックされます。</p> <p><b>サンプルコード:</b></p> <pre>AsReaderP252BResultCode resultCode = [asReaderP252B lock:param]; if (resultCode == AsReaderP252BResultNoError) {     //実行成功時の処理 }</pre>				

```

}else{
    //実行失敗時の処理
}

```

### 2.2.19. unlock

- (AsReaderP252BResultCode)unlock:(AsReaderP252BLockParam *)param;				
	引数名	In/Out	型	説明
引数	param	In	AsReaderP252BLockParam	AsReaderP252BLockParam オブジェクト (8.1 を参照)
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1 を参照)
<p><b>メソッドの説明:</b>            対象タグのロックされているメモリーバンクをアンロックします。            アンロック後にデフォルトパスワードを書き込みできます。            このメソッドを呼び出すと、デリゲート「accessAsReaderP252BResult:」(3.5 を参照) がコールバックされます。</p> <p><b>サンプルコード:</b>  <pre> AsReaderP252BResultCode resultCode = [asReaderP252B unlock:param]; if (resultCode == AsReaderP252BResultNoError) {     //実行成功時の処理 }else{     //実行失敗時の処理 } </pre> </p>				

### 2.2.20. permaLock

- (AsReaderP252BResultCode)permaLock:(AsReaderP252BLockParam *)param;				
	引数名	In/Out	型	説明
引数	param	In	AsReaderP252BLockParam	AsReaderP252BLockParam オブジェクト (8.1 を参照)

戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1 を参照)
<p><b>メソッドの説明:</b>            タグの対象メモリーバンクを永久にロックします。            永久にロック後は、アンロックできなくなり、書き込みもできなくなります。            このメソッドを呼び出すと、デリゲート「accessAsReaderP252BResult:」(3.5 を参照) がコールバックされます。</p> <p><b>サンプルコード:</b></p> <pre>AsReaderP252BResultCode resultCode = [asReaderP252B permaLock:param]; if (resultCode == AsReaderP252BResultNoError) {     //実行成功時の処理 }else{     //実行失敗時の処理 }</pre>				

### 2.2.21. kill

- (AsReaderP252BResultCode)kill:(NSString *)killPassword				
	引数名	In/Out	型	説明
引数	killPassword	In	NSString	タグのキルパスワード
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1 列挙を参照)。
<p><b>メソッドの説明:</b>            対象タグをキルします。            注: キルしたタグは使用できなくなります。            このメソッドを呼び出すと、デリゲート「accessAsReaderP252BResult:」(3.5 を参照) がコールバックされます。</p> <p><b>サンプルコード:</b></p> <pre>AsReaderP252BResultCode resultCode = [asReaderP252B kill: @"00000000"]; if (resultCode == AsReaderP252BResultNoError) {     //実行成功時の処理 }</pre>				

```

}else{
    //実行失敗時の処理
}

```

## 2.2.22. defaultParameter

- (AsReaderP252BResultCode)defaultParameter;				
	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1を参照)
<p><b>メソッドの説明:</b> 全てのパラメータ値をデフォルトに戻します。</p> <p><b>サンプルコード:</b>  <pre> AsReaderP252BResultCode resultCode = [asReaderP252B defaultParameter]; if (resultCode == AsReaderP252BResultNoError) {     //実行成功時の処理 }else{     //実行失敗時の処理 } </pre> </p>				

## 2.2.23. saveParameter

- (AsReaderP252BResultCode)saveParameter;				
	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1を参照)
<p><b>メソッドの説明:</b> 全てのパラメータ値を AsReader に保存します。</p>				

**サンプルコード：**

```

AsReaderP252BResultCode resultCode = [asReaderP252B saveParameter];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}

```

**2.2.24. startDecode****- (AsReaderP252BResultCode)startDecode;**

	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode(8.4.1 を参照)

**メソッドの説明：**

バーコード読取を開始します。

このメソッドを呼び出すと、デリゲート「detectBarcode:」(3.12を参照)がコールバックされます。

**サンプルコード：**

```

AsReaderP252BResultCode resultCode = [asReaderP252B startDecode];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}

```

**2.2.25. setBarcodeTimeOut****- (AsReaderP252BResultCode)setBarcodeTimeOut:(int)barcodeTimeOut;**

	引数名	In/Out	型	説明
引数	barcodeTimeOut	In	int	タイムアウト時間 (単位 : s) 値の範囲 : 4~300
戻り値	-	Out	AsReaderP252B	メソッドの実行結果

			resultCode	AsReaderP252BResultCode(8.4.1 を参照)
<p><b>メソッドの説明:</b>          バーコードスキャン時のタイムアウト時間を設定します。</p> <p><b>サンプルコード:</b></p> <pre>AsReaderP252BResultCode resultCode = [asReaderP252B setBarcodeTimeOut:4]; if (resultCode == AsReaderP252BResultNoError) {     //実行成功時の処理 }else{     //実行失敗時の処理 }</pre>				

### 2.2.26. getBarcodeTimeOut

<b>- (int)getBarcodeTimeOut;</b>				
	引数名	In/Out	型	説明
戻り値	-	Out	int	タイムアウト時間 (単位 : s)
<p><b>メソッドの説明:</b>          バーコードをスキャン時のタイムアウト時間を取得します。</p> <p><b>サンプルコード:</b></p> <pre>int barcodeTimeout = [asReaderP252B getBarcodeTimeOut];</pre>				

### 2.2.27. epcMaskCount

<b>- (int)epcMaskCount;</b>				
	引数名	In/Out	型	説明
戻り値	-	Out	int	EPC マスクの数
<p><b>メソッドの説明:</b>          AsReader に設定される EPC マスクの数を取得します。</p> <p><b>サンプルコード:</b></p>				

```
int count= [asReaderP252B epcMaskCount];
```

### 2.2.28. addEpcMask

-(AsReaderP252BResultCode)addEpcMask:(int)offset length:(int)length mask:(NSString *)mask;				
	引数名	In/Out	型	説明
引数	offset	In	int	マスクのオフセット (単位 : bit)
引数	length	In	int	マスクの長さ (単位 : bit)
引数	mask	In	NSString	マスク値 (16 進数の文字列)
戻り値	-	Out	AsReaderP252BResultC ode	メソッドの実行結果 AsReaderP252BResultCode(8.4. 1 を参照)

**メソッドの説明:**  
EPC マスクを追加します。

**サンプルコード :**

```
AsReaderP252BResultCode resultCode = [asReaderP252B addEpcMask:32 length:16  
mask:@"1234"];  
if (resultCode == AsReaderP252BResultNoError) {  
    //実行成功時の処理  
}  
else{  
    //実行失敗時の処理  
}
```

### 2.2.29. addEpcMask

-(AsReaderP252BResultCode)addEpcMask:(AsReaderP252BSelectMaskEPCParam *)mask;				
	引数名	In/Out	型	説明
引数	mask	In	AsReaderP252BSelectM askEPCParam	AsReaderP252BSelectMaskEPCP aram オブジェクト (8.2 を参照)
戻り値	-	Out	AsReaderP252BResultC	メソッドの実行結果

			ode	AsReaderP252BResultCode (8.4.1 を参照)
<p><b>メソッドの説明:</b> AsReaderP252BSelectMaskEPCParam オブジェクトのフォーマットで EPC マスクを追加します。</p> <p><b>サンプルコード:</b></p> <pre>AsReaderP252BResultCode resultCode = [asReaderP252B addEpcMask:mask]; if (resultCode == AsReaderP252BResultNoError) {     //実行成功時の処理 }else{     //実行失敗時の処理 }</pre>				

### 2.2.30. getEpcMask

<b>- (AsReaderP252BSelectMaskEPCParam *)getEpcMask:(int)index;</b>				
	引数名	In/Out	型	説明
引数	index	In	int	EPC マスクのインデックス
戻り値	-	Out	AsReaderP252BSelectMaskEPCParam	AsReaderP252BSelectMaskEPCParam オブジェクト (8.2 を参照)
<p><b>メソッドの説明:</b> 指定したインデックスの EPC マスクを取得します。</p> <p><b>サンプルコード:</b></p> <pre>AsReaderP252BSelectMaskEPCParam *mask= [asReaderP252B getEpcMask:0];</pre>				

### 2.2.31. clearEpcMask

<b>- (AsReaderP252BResultCode)clearEpcMask;</b>				
	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode (8.4.1 を参照)

**メソッドの説明:**

設定した EPC マスクデータを削除します。

**サンプルコード:**

```
AsReaderP252BResultCode resultCode = [asReaderP252B clearEpcMask];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}
```

**2.2.32. setFrequencyAutomatic**

- (void)setFrequencyAutomatic:(BOOL)isAutomatic;

	引数名	In/Out	型	説明
引数	isAutomatic	In	BOOL	ホッピングモード YES : オートホッピング有効 NO : オートホッピング無効

**メソッドの説明:**

RFID インベントリでオートホッピングモードを使用するかどうかを設定します。

**サンプルコード:**

```
[asReaderP252B setFrequencyAutomatic:YES];
```

**2.2.33. getFrequencyAutomatic**

- (BOOL)getFrequencyAutomatic;

	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	ホッピングモード YES : オートホッピング有効 NO : オートホッピング無効

**メソッドの説明:**

RFID インベントリで使用するホッピングモードを取得します。

**サンプルコード:**

```
BOOL status = [asReaderP252B getFrequencyAutomatic];
if (status) {
    //オートホッピング有効時の処理
} else {
    //オートホッピング無効時の処理
}
```

**2.2.34. setLBT**

- (void)setLBT:(NSArray \*)table;

	引数名	In/Out	型	説明
引数	table	In	NSArray	周波数リスト AsReaderP252BLbtItem オブジェクト (8.3 を参照)

**メソッドの説明:**

RFID インベントリの周波数リストを設定します。

**サンプルコード:**

```
[asReaderP252B setLBT:array];
```

**2.2.35. getLBT**

- (NSArray \*)getLBT;

	引数名	In/Out	型	説明
戻り値	-	Out	NSArray	周波数リスト

**メソッドの説明:**

RFID インベントリの周波数リストを取得します。

**サンプルコード :**

```
NSArray * lbtArray = [asReaderP252B getLBT];
```

**2.2.36. setScanMode**

**- (void)setScanMode:(AsReaderP252BScanMode)scanMode;**

	引数名	In/Out	型	説明
引数	scanMode	In	AsReaderP252BScanMode	スキャンモード AsReaderP252BScanMode (7.1.2を参照)

**メソッドの説明:**

AsReader のスキャンモードを設定します。

**サンプルコード :**

```
[asReader setScanMode: AsReaderP252BRFIDScanMode];
```

**2.2.37. getBaudRateList**

**- (NSMutableArray \*)getBaudRateList;**

	引数名	In/Out	型	説明
戻り値	-	Out	NSMutableArray	ボーレートリスト

**メソッドの説明:**

AsReader のボーレートリストを取得します。

**サンプルコード :**

```
NSMutableArray *array = [asReaderP252B getBaudRateList];
```

**2.2.38. setBaudRate**

**- (AsReaderP252BResultCode)setBaudRate:(int)baudRate;**

	引数名	In/Out	型	説明
引数	baudRate	In	int	ボーレートリストにある番号
戻り値	-	Out	AsReaderP252BResultCode	メソッドの実行結果 AsReaderP252BResultCode (8.4.1 を参照)

**メソッドの説明:**  
AsReader が使用するボーレートの番号を設定します。(値: ボーレートリストにある番号)

**サンプルコード:**

```
AsReaderP252BResultCode resultCode = [asReaderP252B setBaudRate:1];
if (resultCode == AsReaderP252BResultNoError) {
    //実行成功時の処理
}else{
    //実行失敗時の処理
}
```

### 2.2.39. getBaudRate

- (int)getBaudRate;				
	引数名	In/Out	型	説明
戻り値	-	Out	int	ボーレートリストにある番号

**メソッドの説明:**  
AsReader が使用しているボーレートの番号を取得します。(値: ボーレートリストにある番号)

**サンプルコード:**

```
int baudRate= [asReaderP252B getBaudRate];
```

## 3 AsReaderP252BDelegate クラス

### 3.1. whenAsReaderP252BIsConnected:

- (void)whenAsReaderP252BIsConnected:(BOOL)isConnected;				
	引数名	In/Out	型	説明
引数	isConnected	Out	BOOL	接続状態 YES : 接続済み NO : 切断
<p><b>メソッドの説明:</b> AsReader の接続状態を受信します。 connect (6.2.1 を参照) 、disconnect (6.2.2 を参照) メソッドを呼び出した後、コールバックされます。</p> <p><b>サンプルコード:</b></p> <pre>-(void)whenAsReaderP252BIsConnected:(BOOL)isConnected{     if (isConnected){         //接続     } else {         //切断     } }</pre>				

### 3.2. updateAsReaderP252BDeviceState:

- (void)updateAsReaderP252BDeviceState:(AsReaderP252BResultCode)error;				
	引数名	In/Out	型	説明
引数	error	Out	AsReaderP252BResultCode	AsReaderP252BResultCode(8.4.1 列挙を参照)
<p><b>メソッドの説明:</b> 一部メソッド/プロパティの実行にエラーが発生したときにコールバックされます。 *メソッド/プロパティについては、次の章を参照してください。</p>				

2.1.属性、2.2.2.batteryStatus、2.2.3.regionName、2.2.4.serialNumber、  
 2.2.5.rFModuleVersion、2.2.7.firmwareVersion、2.2.22.defaultParameter、  
 2.2.27.epcMaskCount、2.2.28.addEpcMask、2.2.29.addEpcMask、2.2.30.getEpcMask、  
 2.2.31.clearEpcMask、2.2.34.setLBT、2.2.35.getLBT、  
 2.2.36.setScanMode。

**サンプルコード:**

```
- (void)updateAsReaderP252BDeviceState:(AsReaderP252BResultCode)error {
    //error AsReaderP252BResultCode(8.4.1 列挙を参照)
}
```

### 3.3. readAsReaderP252BTag:

```
- (void)readAsReaderP252BTag:(NSString *)tag
                        rssi:(float)rssi
                        phase:(float)phase
                        frequency:(float)frequency;
```

	引数名	In/Out	型	説明
引数	tag	Out	NSString	タグの PCEPC 値 (16 進数の文字列)
引数	rssi	Out	float	タグの RSSI 値
引数	phase	Out	float	タグの Phase 値
引数	frequency	Out	float	タグの周波数値

**メソッドの説明:**

インベントリしたデータを受信します。

inventory (2.2.11 を参照) メソッドを呼び出した後、コールバックされます。

**サンプルコード:**

```
- (void)readAsReaderP252BTag:(NSString *)tag
                        rssi:(float)rssi
                        phase:(float)phase
                        frequency:(float)frequency{
    //tag      タグの PCEPC 値 (16 進数の文字列)
    //rssi     タグの RSSI 値
    //phase    タグの Phase 値
```

```
//frequency タグの周波数値
}
```

### 3.4. changedAsReaderP252BActionState:

```
-(void)changedAsReaderP252BActionState:(AsReaderP252BCommandType)action
resultCode:(NSInteger)resultCode;
```

	引数名	In/Out	型	説明
引数	action	Out	AsReaderP252BCommandType	AsReaderP252BCommandType (7.1.1 列挙を参照)
引数	resultCode	Out	AsReaderP252BResultCode	AsReaderP252BResultCode(8.4.1 列挙を参照)

#### メソッドの説明:

インベントリ開始とインベントリ停止の実行結果を受信します。

メソッド inventory (2.2.11 を参照)、inventoryHumidityTag (2.2.12 を参照)、inventoryTemperatureTagWithType (2.2.13 を参照)、stop (2.2.14 を参照)、stopSync (2.2.15 を参照) を呼び出した後、コールバックされます。

#### サンプルコード:

```
-(void)changedAsReaderP252BActionState:(AsReaderP252BCommandType)action
resultCode:(NSInteger)resultCode{
    //action AsReaderP252BCommandType(7.1.1 列挙を参照)。
    //resultCode AsReaderP252BResultCode(8.4.1 列挙を参照)。
}
```

### 3.5. accessAsReaderP252BResult:

```
-(void)accessAsReaderP252BResult:(AsReaderP252BResultCode)errorCode
actionState:(AsReaderP252BCommandType)action
epc:(NSString *)epc
data:(NSString *)data
rssi:(float)rssi
phase:(float)phase
frequency:(float)frequency;
```

	引数名	In/Out	型	説明
引数	error	Out	AsReaderP252BResultCode	AsReaderP252BResultCode(8.4.1 列挙を参照)
引数	action	Out	AsReaderP252BCommandType	AsReaderP252BCommandType(7.1.1 列挙を参照)
引数	epc	Out	NSString	タグの EPC 値 (16 進数)
引数	data	Out	NSString	読み取ったタグデータ (16 進数)
引数	rsssi	Out	float	タグの RSSI 値
引数	phase	Out	float	タグの Phase 値
引数	frequency	Out	float	タグの周波数値

**メソッドの説明:**

タグの読取、書込、ロック、キルメソッドの実行結果を受信します。

メソッド readMemory (2.2.16 を参照)、writeMemory (2.2.17 を参照)、lock (2.2.18 を参照)、unlock (2.2.19 を参照)、permaLock (2.2.20 を参照)、kill (2.2.21 を参照)、startDecode (2.2.24 を参照) を呼び出した後、コールバックされます。

**サンプルコード:**

```

-(void)accessAsReaderP252BResult:(AsReaderP252BResultCode)error
                        actionState:(AsReaderP252BCommandType)action
                        epc:(NSString *)epc
                        data:(NSString *)data
                        rssi:(float)rssi
                        phase:(float)phase
                        frequency:(float)frequency{
    //error AsReaderP252BResultCode(8.4.1 列挙を参照)
    //action AsReaderP252BCommandType(7.1.1 列挙を参照)
    //epc タグの EPC 値 (16 進数)
    //data 読み取ったタグデータ (16 進数)
    //rssi タグの RSSI 値
    //phase タグの Phase 値
    //frequency タグの周波数値
}

```

### 3.6. readAsReaderP252BTemperatureTag:

<pre> -(void)readAsReaderP252BTemperatureTag:(NSString*)tag         rssi:(float)rssi         phase:(float)phase         frequency:(float)frequency         temperature:(float)temperature; </pre>				
	引数名	In/Out	型	説明
引数	tag	Out	NSString	読み取ったタグデータ (16 進数)
引数	rssi	Out	float	タグの RSSI 的値
引数	phase	Out	float	タグの Phase 値
引数	frequency	Out	float	タグの周波数値
引数	temperature	Out	float	タグの温度値

**メソッドの説明:**  
 温度タグのデータを受信します。  
 メソッド inventoryTemperatureTagWithType (2.2.13 を参照) を呼び出した後、コールバックされます。

**サンプルコード:**

```

-(void)readAsReaderP252BTemperatureTag:(NSString*)tag
        rssi:(float)rssi
        phase:(float)phase
        frequency:(float)frequency
        temperature:(float)temperature{
    //tag 読み取ったタグデータ (16 進数)
    //rssi タグの RSSI 値
    //phase タグの Phase 値
    //frequency タグの周波数値
    //temperature 温度
}

```

### 3.7. readAsReaderP252BHumidityTag:

<pre> -(void)readAsReaderP252BHumidityTag:(NSString*)tag         rssi:(float)rssi </pre>				
--	--	--	--	--

<b>phase:(float)phase frequency:(float)frequency humidity:(float)humidity;</b>				
	引数名	In/Out	型	説明
引数	tag	Out	NSString	タグデータ (16 進数)
引数	rssi	Out	float	タグの RSSI 的値
引数	phase	Out	float	タグの Phase 値
引数	frequency	Out	float	タグの周波数値
引数	humidity	Out	float	タグの乾湿度値

**メソッドの説明:**  
 乾湿度タグのデータを受信します。  
 メソッド inventoryHumidityTag (2.2.12 を参照) を呼び出した後、コールバックされます。

**サンプルコード:**

```

-(void)readAsReaderP252BHumidityTag:(NSString*)tag
    rssi:(float)rssi
    phase:(float)phase
    frequency:(float)frequency
    humidity:(float)humidity{
    //tag タグデータ (16 進数)
    //rssi タグの RSSI 値
    //phase タグの Phase 値
    //frequency タグの周波数値
    //humidity 乾湿度
}

```

### 3.8. onAsReaderP252BTriggerKeyEvent:

<b>-(BOOL)onAsReaderP252BTriggerKeyEvent:(BOOL)status;</b>				
	引数名	In/Out	型	説明
引数	status	Out	BOOL	AsReader の Trigger キーの状態 YES : Trigger キーを押下する NO : Trigger キーを離れる

戻り値	-	In	BOOL	YES : SDK デフォルト操作を実行する Trigger キーを押下する : スキャン開始 Trigger キーを離れる : スキャン停止 NO : SDK デフォルト操作を実行しない
<b>メソッドの説明:</b> AsReader の Trigger キーを押下、離れた後、コールバックされます。				
<b>サンプルコード:</b> <pre>-(BOOL)onAsReaderP252BTriggerKeyEvent:(BOOL)status{     //status   AsReader の Trigger キーの状態 }</pre>				

### 3.9. onAsReaderP252BModeKeyEvent:

- (BOOL)onAsReaderP252BModeKeyEvent:(BOOL)status;				
	引数名	In/Out	型	説明
引数	status	Out	BOOL	AsReader の Mode キーの状態 YES : Mode キーを押下する NO : Mode キーを離れる
戻り値	-	In	BOOL	YES : Barcode/RFID モードを切り替える NO : 現在のモードを保持する
<b>メソッドの説明:</b> AsReader の Mode キーを押下、離れた後、コールバックされます。				
<b>サンプルコード:</b> <pre>-(BOOL)onAsReaderP252BModeKeyEvent:(BOOL)status{     //status   AsReader の Mode キーの状態 }</pre>				

### 3.10. whenAsReaderP252BReceivedData:

- (void)whenAsReaderP252BReceivedData:(NSData *)receivedData;
---

	引数名	In/Out	型	説明
引数	receivedData	Out	NSData	コマンド
<b>メソッドの説明:</b> AsReader から返したコマンドを受信します。				
<b>サンプルコード:</b> <pre>- (void)whenAsReaderP252BReceivedData:(NSData *)receivedData{     //receivedData コマンド }</pre>				

### 3.11. whenAsReaderP252BWriteData:

- (void)whenAsReaderP252BWriteData:(NSData *)writeData;				
	引数名	In/Out	型	説明
引数	writeData	Out	NSData	コマンド
<b>メソッドの説明:</b> AsReader へ送信したコマンドを受信します。				
<b>サンプルコード:</b> <pre>- (void)whenAsReaderP252BWriteData:(NSData *)writeData{     //writeData コマンド }</pre>				

### 3.12. detectBarcode:

-(void)detectBarcode:(AsReaderP252BBarcodeType)barcodeType barcodeData:(NSData *)barcodeData;				
	引数名	In/Out	型	説明
引数	barcodeType	Out	AsReaderP252BBarcodeType	バーコードタイプ AsReaderP252BBarcodeType(4.2.1 列挙を参照)
引数	barcodeData	Out	NSData	バーコードデータ

**メソッドの説明:**

スキャンしたバーコードデータを受信します。

メソッド startDecode (2.2.24 を参照) を呼び出した後、コールバックされます。

**サンプルコード:**

```
-(void)detectBarcode:(AsReaderP252BBarcodeType)barcodeType  
    barcodeData:(NSData *)barcodeData{  
    //barcodeType AsReaderP252BBarcodeType(4.2.1 列挙を参照)  
    //barcodeData バーコードデータ  
}
```

## 4 AsReaderP252BBarcode クラス

### 4.1. メソッド

#### 4.1.1. getBarcodeString

+ (NSString *)getBarcodeString:(AsReaderP252BBarcodeType)barcodeType;				
	引数名	In/Out	型	説明
引数	barcodeType	In	AsReaderP252BBarcodeType	AsReaderP252BBarcodeType(4.2.1 列挙を参照)
戻り値	-	Out	NSString	列挙 AsReaderP252BBarcodeTypeに相応するバーコードタイプ
<p><b>メソッドの説明:</b> バーコードタイプを取得します。</p> <p><b>サンプルコード:</b>  <pre>NSString *barcodeType = [AsReaderP252BBarcode getBarcodeString:AsReaderP252BBarcodeTypeCode39];</pre> </p>				

### 4.2. 列挙

#### 4.2.1. AsReaderP252BBarcodeType

定義	説明
AsReaderP252BBarcodeTypeNoRead	未知なタイプ
AsReaderP252BBarcodeTypeCode39	Code39

AsReaderP252BBarcodeTypeCode11	Code11
AsReaderP252BBarcodeTypeCodabar	Codabar
AsReaderP252BBarcodeTypeEAN13	EAN-13
AsReaderP252BBarcodeTypeCode128	Code128
AsReaderP252BBarcodeTypeEAN13With2Supps	EAN-13 with 2 Supps
AsReaderP252BBarcodeTypeIndustrial2Of5	Industrial 2 of 5
AsReaderP252BBarcodeTypeEAN13With5Supps	EAN-13 with 5 Supps
AsReaderP252BBarcodeTypeIATA2Of5	IATA 2 of 5
AsReaderP252BBarcodeTypeMSI	MSI
AsReaderP252BBarcodeTypeInterleaved2Of5	Interleaved 2 of 5
AsReaderP252BBarcodeTypeEAN128	EAN-128
AsReaderP252BBarcodeTypeCode93	Code93
AsReaderP252BBarcodeTypeUPCE1	UPC-E1
AsReaderP252BBarcodeTypeUPCA	UPC-A
AsReaderP252BBarcodeTypeUPCE1With2Supps	UPC-E1 with 2 Supps
AsReaderP252BBarcodeTypeUPCAWith2Supps	UPC-A with 2 Supps
AsReaderP252BBarcodeTypeUPCAWith5Supps	UPC-A with 5 Supps
AsReaderP252BBarcodeTypeUPCE1With5Supps	UPC-E1 with 5 Supps
AsReaderP252BBarcodeTypeTriopticCode39	Trioptic Code39
AsReaderP252BBarcodeTypeUPCE0	UPC-E
AsReaderP252BBarcodeTypeBooklandEAN	Bookland EAN
AsReaderP252BBarcodeTypeUPCE0With2Supps	UPC-E with 2 Supps

AsReaderP252BBarcodeTypeCouponCode	Coupon Code
AsReaderP252BBarcodeTypeUPCE0With5Supps	UPC-E with 5 Supps
AsReaderP252BBarcodeTypeGS1DataBarLimitedRSSLimited	GS1 DataBar Limited (RSS-Limited)
AsReaderP252BBarcodeTypeEAN8	EAN-8
AsReaderP252BBarcodeTypeGS1DataBarRSS14	GS1 DataBar (RSS-14)
AsReaderP252BBarcodeTypeEAN8With2Supps	EAN-8 with 2 Supps
AsReaderP252BBarcodeTypeGS1DataBarExpandedRSSExpanded	GS1 DataBar Expanded (RSS-Expanded)
AsReaderP252BBarcodeTypeEAN8With5Supps	EAN-8 with 5 Supps
AsReaderP252BBarcodeTypeMatrix2Of5	Matrix 2 of 5
AsReaderP252BBarcodeTypeChinaPostChinese2Of5	China Post (Chinese 2 of 5)
AsReaderP252BBarcodeTypeCode32	Code32
AsReaderP252BBarcodeTypeUKPlessey	UK Plessey
AsReaderP252BBarcodeTypeISBT128	ISBT128
AsReaderP252BBarcodeTypePDF417	PDF417
AsReaderP252BBarcodeTypeAztec	Aztec
AsReaderP252BBarcodeTypeMicroPDF417	MicroPDF417
AsReaderP252BBarcodeTypeQR	QR
AsReaderP252BBarcodeTypeDataMatrix	DataMatrix
AsReaderP252BBarcodeTypeMicroQR	Micro QR
AsReaderP252BBarcodeTypeHanXinCode	HanXin Code
AsReaderP252BBarcodeTypeMaxicode	Maxicode

AsReaderP252BBarcodeTypeITF14	ITF-14
AsReaderP252BBarcodeTypeITF6	ITF-6
AsReaderP252BBarcodeTypeAIM128	AIM 128
AsReaderP252BBarcodeTypeISSN	ISSN
AsReaderP252BBarcodeTypeISBN	ISBN
AsReaderP252BBarcodeTypeGS1Databar	GS1-Databar

## 5 AsReaderP252BDeviceManager

### 5.1. 属性

属性名	属性	型	説明
delegate	nonatomic, weak	AsReaderP252BDeviceManager Delegate	AsReaderP252BDeviceManagerDelegate(5.3を参照)

### 5.2. メソッド

#### 5.2.1. shared

+ (AsReaderP252BDeviceManager *)shared;				
	引数名	In/Out	型	説明
戻り値	-	Out	AsReaderP252BDeviceManager	AsReaderP252BDeviceManager オブジェクト(5を参照)
<p><b>メソッドの説明:</b> AsReaderP252BDeviceManager オブジェクト (シングルトンモード) を生成・初期化します。</p> <p><b>サンプルコード:</b> AsReaderP252BDeviceManager *asReaderP252BDeviceManager= [AsReaderP252BDeviceManager shared];</p>				

#### 5.2.2. startDiscovered

- (void)startDiscovered;				
<p><b>メソッドの説明:</b> AsReader を検索します。 このメソッドを呼び出した後、デリゲート「didDiscoverAsReaderP252BDeviceList:」(5.3.1を参照)が</p>				

コールバックされます。

**サンプルコード:**

```
[asReaderP252BDeviceManager startDiscovered];
```

### 5.2.3. stopDiscovered

**- (void)stopDiscovered;**

**メソッドの説明:**

AsReader の検索を停止します。

**サンプルコード:**

```
[asReaderP252BDeviceManager stopDiscovered];
```

## 5.3. デリゲート

### 5.3.1. didDiscoverAsReaderP252BDeviceList

**-(void)didDiscoverAsReaderP252BDeviceList:(NSArray<AsReaderP252B\*> \*)asReaderP252BList;**

	引数名	In/Out	型	説明
<b>引数</b>	asReaderP252BList	Out	NSArray	検索された AsReader のリスト (配列要素は AsReaderP252B オブジェクト)

**メソッドの説明:**

検索された AsReaderP252 オブジェクトの配列を受信します。(メソッドは複数回コールバックされます) startDiscovered (5.2.2 を参照) メソッドを呼び出した後、コールバックされます。

**サンプルコード:**

```
- (void)didDiscoverAsReaderP252BDeviceList:(NSArray<AsReaderP252B *> *)asReaderP252BList{
    // asReaderP252BList は AsReaderP252B オブジェクトの配列
}
```

## 6 AsReaderP252BDevice

### 6.1. 属性

属性名	属性	型	説明
currentAccessory	nonatomic, strong, readonly	EAAccessory	現在の EAAccessory オブジェクトを取得する
deviceName	nonatomic, strong, readonly	NSString	現在の AsReader 名を取得する
isConnected	nonatomic, assign, readonly	BOOL	AsReader の接続状態を取得する YES : 接続中 NO : 切断

### 6.2. メソッド

#### 6.2.1. connect

- (void)connect;
<p><b>メソッドの説明:</b> AsReader へ接続します。 このメソッドを呼び出した後、デリゲート「whenAsReaderP252BIsConnected:」(3.1 を参照) がコールバックされます。</p> <p><b>サンプルコード:</b> [asReaderP252B connect];</p>

## 6.2.2. disconnect

- (void)disconnect;

**メソッドの説明:**

AsReader との接続を切断します。

このメソッドを呼び出した後、デリゲート「whenAsReaderP252BIsConnected:」（3.1 を参照）がコールバックされます。

**サンプルコード:**

```
[asReaderP252B disconnect];
```

## 7 AsReaderP252BPacket

### 7.1. 列挙

#### 7.1.1. AsReaderP252BCommandType

定義	説明
AsReaderP252BCommandInventory	インベントリ中
AsReaderP252BCommandReadMemory	メモリ読み取り
AsReaderP252BCommandWriteMemory	メモリ書き込み
AsReaderP252BCommandKill	タグキル
AsReaderP252BCommandLock	タグロック
AsReaderP252BCommandUnlock	タグアンロック
AsReaderP252BCommandPermaLock	タグ永久ロック
AsReaderP252BCommandStop	オペレーション停止
AsReaderP252BCommandDefaultParam	デフォルトパラメータに戻す
AsReaderP252BCommandSaveParam	パラメータ保存
AsReaderP252BCommandBuzzerStart	ブザー開始
AsReaderP252BCommandDecodeStart	バーコードスキャン開始

## 7.1.2. AsReaderP252BScanMode

定義	説明
AsReaderP252BRFIDScanMode	RFID モード
AsReaderP252BBarcodeScanMode	Barcode モード

## 8 AsReaderP252BRfidValues

### 8.1. AsReaderP252BLockParam

#### 8.1.1. 属性

属性名	属性	型	説明
killPassword	nonatomic	BOOL	キルパスワードエリアの制御要否 YES : 制御する NO : 制御しない
accessPassword	nonatomic	BOOL	アクセスパスワードエリアの制御要否 YES : 制御する NO : 制御しない
epc	nonatomic	BOOL	EPC エリアの制御要否 YES : 制御する NO : 制御しない
tid	nonatomic	BOOL	TID エリアの制御要否 YES : 制御する NO : 制御しない
user	nonatomic	BOOL	User エリアの制御要否 YES : 制御する NO : 制御しない

### 8.2. AsReaderP252BSelectMaskEPCParam

#### 8.2.1. 属性

属性名	属性	型	説明
offset	nonatomic	int	マスクのオフセット

length	nonatomic	int	マスクの長さ
mask	nonatomic, strong	NSString	マスク値 (16 進数)

## 8.3. AsReaderP252BLbtItem

### 8.3.1. 属性

属性名	属性	型	説明
mSlot	nonatomic	int	LBT 周波数リストの使用する周波数のインデックス
mIsUsed	mIsUsed	BOOL	LBT 周波数リストにある特定周波数の使用要否 YES : 使用する NO : 使用しない
frequency	nonatomic, strong	NSString	LBT 周波数リストの周波数値

## 8.4. 列挙

### 8.4.1. AsReaderP252BResultCode

定義	説明
AsReaderP252BResultNoError	成功
AsReaderP252BResultOtherError	未知なエラー
AsReaderP252BResultNotConnected	AsReader 未接続
AsReaderP252BResultInvalidParameter	送信したパラメータが無効
AsReaderP252BResultUndefined	未定義なエラー

AsReaderP252BResultMemoryOverrun	アクセスするメモリーは範囲を超えた
AsReaderP252BResultMemoryLocked	タグがロックされた
AsReaderP252BResultInsufficientPower	バッテリー不足
AsReaderP252BResultHandleMismatch	ハンドラーが不一致
AsReaderP252BResultCRCError	CRC エラー
AsReaderP252BResultCommandFormatError	コマンドフォーマットエラー
AsReaderP252BResultOutOfRetries	アクセスパスワードエラー、キルパスワードエラー
AsReaderP252BResultTimeout	タイムアウト

#### 8.4.2. AsReaderP252BMemoryBank

定義	説明
AsReaderP252BBank_Reserved	Reserved メモリーバンク
AsReaderP252BBank_EPC	EPC メモリーバンク
AsReaderP252BBank_TID	TID メモリーバンク
AsReaderP252BBank_User	User メモリーバンク

#### 8.4.3. AsReaderP252BTemperatureTagType

定義	説明
AsReaderP252BTemperatureTagType_0	温度タグのタイプ : Magnus-S3

#### 8.4.4. AsReaderP252BBuzzerState

定義	説明
AsReaderP252BBuzzer_Off	ブザー音オフ

AsReaderP252BBuzzer_Low	ブザー音小
AsReaderP252B3Buzzer_High	ブザー音大

### 8.4.5. AsReaderP252BSessionType

定義	説明
AsReaderP252BSession_S0	inventoried S0
AsReaderP252BSession_S1	inventoried S1
AsReaderP252BSession_S2	inventoried S2
AsReaderP252BSession_S3	inventoried S3

### 8.4.6. AsReaderP252BSessionFlag

定義	説明
AsReaderP252BSessionFlag_A	A only
AsReaderP252BSessionFlag_B	B only
AsReaderP252BSessionFlag_AB	A or B

## 8.5. 構造体

### 8.5.1. AsReaderP252BMinMaxValue

定義	型	説明
min	int	設定可能な電波出力最小値
max	int	設定可能な電波出力最大値