

**AsReader**<sup>®</sup>

# AsReader BOX-Type

**Objective-C SDK 開発マニュアル**

**ASR-P52N**

## 修正履歴

バージョン	修正内容	日付
1.0	新規作成	2023/2/14

# 目次

<b>1</b>	<b>開発環境の構築</b>	<b>6</b>
1.1.	SDK の追加	6
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.	AsP52N オブジェクトの生成・初期化を行う (シングルトン)	8
1.3.2.	デリゲートを設定する	8
1.3.3.	検索を開始する	8
1.3.4.	AsP52N オブジェクトを接続する	8
1.3.5.	RFID データを読み取る	9
1.3.6.	RFID データの読取を停止する	14
1.3.7.	AsP52N オブジェクトの接続を切断する	14
<b>2</b>	<b>AsP52N クラス</b>	<b>16</b>
2.1.	属性	16
2.2.	メソッド	16
2.2.1.	sharedInstance	16
2.2.2.	getSDKVersion	16
2.2.3.	startSearchDevice	17
2.2.4.	stopSearchDevice	17
2.2.5.	connectServerWithIP	18
2.2.6.	disconnectServer	19
2.2.7.	startInventoryWithEPC	19
2.2.8.	stopInventory	20
2.2.9.	readTagWithAccessPassword	21
2.2.10.	writeTagWithAccessPassword	22
2.2.11.	lockTagWithAccessPassword	23
2.2.12.	killTagWithKillPassword	23
2.2.13.	setAntennaPortsStatusWithAntenna1	24
2.2.14.	getAntennaPortsStatus	25
2.2.15.	setOutputPowerLevel	25
2.2.16.	getOutputPowerLevel	26
2.2.17.	setSession	26
2.2.18.	getSession	27

2.2.19. setDHCPForMACAddress .....	27
2.2.20. setDHCPForMACAddress .....	28
2.2.21. getFirmwareVersion .....	29
2.2.22. getMessageWithResultCode .....	29
2.2.23. setSearchDeviceTimeout.....	29
2.2.24. readDataWithAntenna1 .....	30
2.2.25. writeDataWithAntenna1.....	34
2.2.26. lockTagWithAntenna1 .....	37
2.2.27. killTagWithAntenna1 .....	39
2.2.28. startInventoryWithAntenna1 .....	42
2.2.29. startInventoryTIDWithAntenna1 .....	43
2.2.30. startInventoryMoistureWithAntenna1 .....	45
2.2.31. startInventoryMagnusS3WithAntenna1 .....	47
2.2.32. stopReadData .....	49
2.2.33. setScanContinuous .....	49
2.2.34. getReaderInfo .....	50
2.2.35. getReaderAntennaPower .....	50
2.2.36. setReaderAntenna1Power .....	51
2.2.37. getReaderBasebandInformation.....	52
2.2.38. setReaderBasebandRate .....	53
2.2.39. getReaderAutoDormancy .....	54
2.2.40. setReaderAutoDormancy .....	54
2.2.41. getReaderRFIDFrequency .....	55
2.2.42. setReaderRFIDFrequency.....	56
2.2.43. getReaderRFIDFrequencyList .....	56
2.2.44. getReaderTagLog .....	57
2.2.45. setReaderTagLogFilterTime .....	57
2.2.46. getReaderFrequency .....	58
2.2.47. setReaderFrequencyAutomatic .....	59
2.2.48. getReaderEthernetMacAddress .....	59
2.2.49. getReaderTime .....	60
2.2.50. setReaderTime .....	61
2.2.51. getReaderEthernetIP .....	61
2.2.52. setReaderEthernetIP .....	62
2.2.53. factoryReset.....	63
2.2.54. restartDevice .....	64
2.2.55. getTemperature .....	64
2.3. デリゲート.....	65

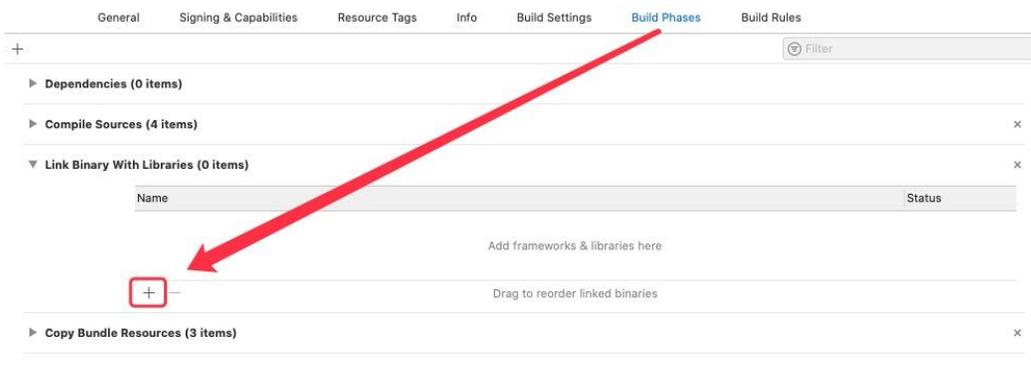
2.3.1. AsP52N:searchingIP:isFinish: .....	65
2.3.2. AsP52N:changeNetworkState:error: .....	65
2.3.3. AsP52N:didStartInventoryExecutionStatus: .....	66
2.3.4. AsP52N:didStopInventoryExecutionStatus: .....	67
2.3.5.	
AsP52N:receivedInventoryDataWithRSSI:channel:phase:antenna:PC:EPC:TID: ..	68
2.3.6. AsP52N:didKillTagExecutionStatus : .....	69
2.3.7. AsP52N:didWriteTagExecutionStatus: .....	70
2.3.8. AsP52N:didLockTagExecutionStatus: .....	71
2.3.9. AsP52N:didReadTagExecutionStatus: .....	72
2.3.10. AsP52N:receivedTagOperationData: .....	72
2.3.11. AsP52N:didSetAntennaPortsStatusExecutionStatus: .....	73
2.3.12.	
AsP52N:receivedAntennaPortsStatusAntenna1:antenna2:antenna3:antenna4:ante anna5:antenna6:antenna7:antenna8:executionStatus: .....	74
2.3.13. AsP52N:didSetOutputPowerLevelExecutionStatus: .....	75
2.3.14. AsP52N:receivedOutputPowerLevel:executionStatus: .....	76
2.3.15. AsP52N:didSetSessionExecutionStatus: .....	77
2.3.16. AsP52N:receivedSession:executionStatus: .....	78
2.3.17. AsP52N:receivedFirmwareVersion:executionStatus: .....	79
2.3.18. AsP52N:didInventoryEndExecutionStatus: .....	80
2.3.19. AsP52N:receivedReaderInfo: .....	80
2.3.20.	
AsP52N:receivedReaderAntenna1Power:antenna2Power:antenna3Power:antenna4 Power:antenna5Power:antenna6Power:antenna7Power:antenna8Power: .....	81
2.3.21. AsP52N:receivedSetAntennaPowerStatus: .....	83
2.3.22. AsP52N:receivedReaderBasebandRate:QValue:session:inventoryFlag: ....	84
2.3.23. AsP52N:receivedSetBasebandStatus: .....	85
2.3.24. AsP52N:receivedReaderAutoDormancy:dormancyTime: .....	86
2.3.25. AsP52N:receivedSetAutoDormancyStatus: .....	86
2.3.26. AsP52N:receivedReaderRFIDFrequency: .....	87
2.3.27. AsP52N:receivedSetRFIDFrequencyStatus: .....	88
2.3.28. AsP52N:receivedReaderTagLogFilterTime:RSSI : .....	88
2.3.29. AsP52N:receivedSetTagLogStatus: .....	89
2.3.30. AsP52N:receivedReaderFrequencyAutomatic:frequencyList : .....	90
2.3.31. AsP52N:receivedSetFrequencyStatus: .....	91
2.3.32. AsP52N:receivedReaderEthernetMACAddress: .....	91
2.3.33. AsP52N:receivedReaderTime: .....	92

2.3.34. AsP52N:receivedSetReaderTimeStatus: .....	92
2.3.35. AsP52N:receivedReaderEthernetIP:subnetMask:defaultGateway:preferredDNS:alternateDNS:automaticIP: .....	93
2.3.36. AsP52N:receivedSetEthernetIPStatus: .....	94
2.3.37. AsP52N:receivedFactoryResetStatus: .....	95
2.3.38. AsP52N:receivedInventoryStatus: .....	96
2.3.39. AsP52N:receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EPC:TID:userData:reserved:epcBank:moisture:temperature: .....	97
2.3.40. AsP52N:receivedWriteDataStatus: .....	99
2.3.41. AsP52N:receivedReadDataStatus: .....	99
2.3.42. AsP52N:receivedLockDataStatus: .....	100
2.3.43. AsP52N:receivedKillDataStatus: .....	101
2.3.44. AsP52N:receivedUnknowCommand: .....	102
2.3.45. AsP52N:receivedUnknowError: .....	102
2.3.46. AsP52N:receivedAlarmTemperature: .....	103
2.3.47. AsP52N:receivedGetTemperatureAlarmTemperature: .....	103
2.4. 列挙 .....	104
2.4.1. AsP52NetworkState .....	104
2.4.2. AsP52NSessionType .....	104
2.4.3. AsP52NMemoryBankType .....	104
2.4.4. AsP52NTargetMemoryType .....	105
2.4.5. AsP52NLockActionType .....	105
2.4.6. AsP52NBasebandRateStatus .....	105
2.4.7. AsP52NInventoryFlag .....	106
2.4.8. AsP52NRFIDFrequencyStatus .....	106
2.4.9. AsP52NExecutionStatus .....	106
<b>3 付録 .....</b>	<b>111</b>
3.1. Frequency List .....	111
3.1.1. AsP52NRFIDFrequencyStatus920_925MHz .....	111
3.1.2. AsP52NRFIDFrequencyStatusFCC .....	111
3.1.3. AsP52NRFIDFrequencyStatusETSI .....	113
3.1.4. AsP52NRFIDFrequencyStatusJP .....	113

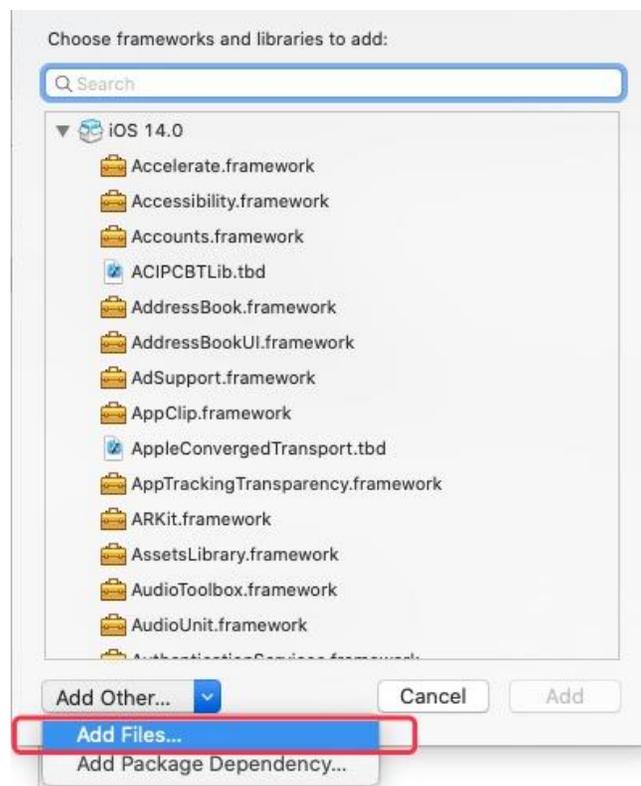
# 1 開発環境の構築

## 1.1. SDK の追加

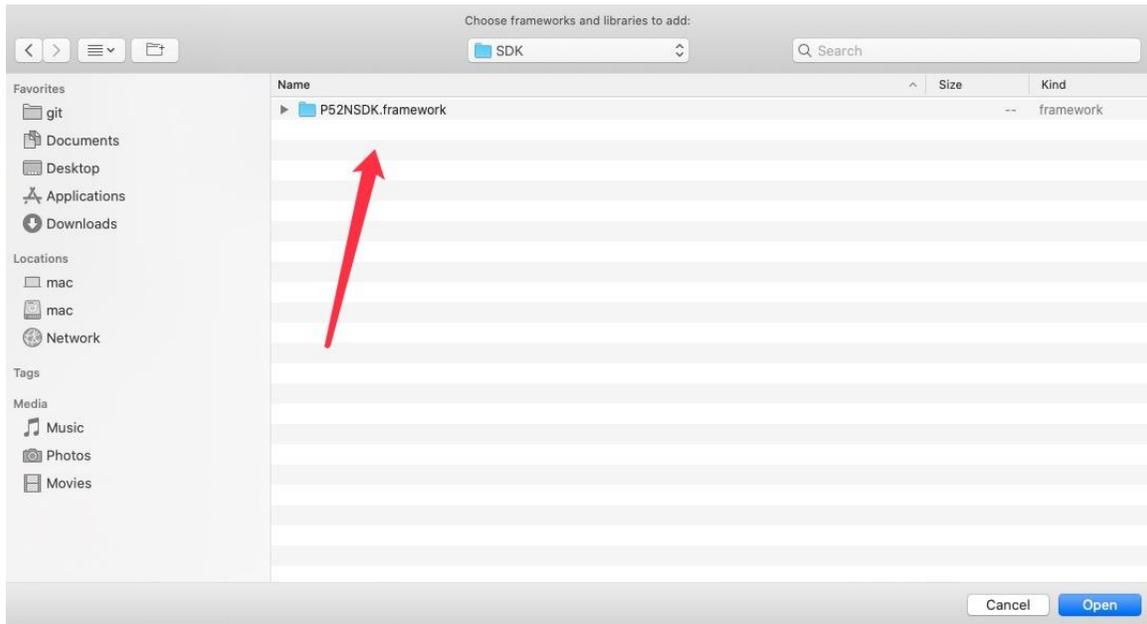
### 1. TARGET -> Build phases -> Link Binary with Libraries



### 2. 「Add Other…」-> 「Add Files…」を選択します。



3. P52NSDK.framework をクリックして、「Open」ボタンをクリックします。



4. 正常に追加されると、下の赤い枠のように表示されます。ここまでで、SDK の追加は完了になります。



## 1.2. ヘッダーファイルのインポート

Objective-CプロジェクトにSDKを使用する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。以下を参照ください。

```
#import <P52NSDK/P52NSDK.h>
```

## 1.3. SDK の使用

### 1.3.1. AsP52N オブジェクトの生成・初期化を行う (シングルトン)

```
AsP52N *asP52N = [AsP52N sharedInstance];
```

### 1.3.2. デリゲートを設定する

```
asP52N.delegate = self;
```

### 1.3.3. 検索を開始する

```
[asP52N startSearchDevice];  
NSMutableArray *arrayIPAddress = [[NSMutableArray alloc] init];  
検索した IP を受信するデリゲート  
- (void)AsP52N:(AsP52N *)AsP52N searchingIP:(NSString *)IP  
isFinish:(BOOL)isFinish{  
    if(isFinish == YES) {  
        //検索に成功  
    } else {  
        //検索を続ける  
        if (![arrayIPAddress containsObject:IP]) {  
            [arrayIPAddress addObject:IP];  
        }  
    }  
}
```

### 1.3.4. AsP52N オブジェクトを接続する

arrayIPAddress に検索した IP を選択し、接続します。

AsP52N オブジェクトを接続するメソッド :

```
[asP52N connectServerWithIP:@"192.168.1.10" remoteport:8160];
```

接続結果を受信するデリゲート :

```
- (void)AsP52N:(AsP52N *)AsP52N
```

```
changeNetworkState:(AsP52NetworkState)state error:(NSError *)error{
```

```
    if (state == Asp52NetworkStateConnected) {
        //接続に成功
    } else if (state == Asp52NetworkStateDisconnected) {
        //接続を切断
    } else if (state == Asp52NetworkStateUnknown) {
        //未知のネットワーク
    } else {
        //error その他のエラー
    }
}
```

### 1.3.5. RFID データを読み取る

#### 1.3.5.1. startInventoryWithAntenna1

[asP52N startInventoryWithAntenna1:YES antenna2:YES antenna3:YES antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES isContinueScan:YES];

##### 1.3.5.1.1. 読取開始の実行結果を受信するデリゲート

```
- (void)AsP52N:(AsP52N *)AsP52N
didStartInventoryExecutionStatus:(AsP52NExecutionStatus)statusCode{
    if (statusCode == Asp52NExecutionStatusSuccess) {
        //読取開始に成功
    } else {
        //読取開始に失敗
        getMessageWithResultCode: メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
        NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
        //errorString 失敗原因
    }
}
```

### 1.3.5.1.2. 読取データを受信するデリゲート

```
- (void)AsP52N:(AsP52N *)AsP52N
receivedInventoryCustomDataWithRSSI:(NSString *)RSSI channel:(NSString
*)channel phase:(NSString *)phase antenna:(NSString *)antenna PC:(NSString
*)PC EPC:(NSString *)EPC TID:(NSString *)TID userData:(NSString *)userData
reserved:(NSString *)reserved epcBank:(NSString *)epcBank
moisture:(NSString *)moisture temperature:(NSString *)temperature {
    RSSI        RSSI データ
    channel      チャンネル情報
    phase        タグ位相情報
    antenna      アンテナ情報
    PC           PC データ
    EPC          EPC データ
    TID          TID データ
    userData     User データ
    reserved     Reserved データ
    epcBank      epcBank データ
    moisture     湿度データ
    temperature  温度データ
}
```

### 1.3.5.1.3. 読取停止の実行結果を受信するデリゲート

```
- (void)AsP52N:(AsP52N *)AsP52N
didInventoryEndExecutionStatus:(AsP52NExecutionStatus)statusCode{
    if (statusCode == AsP52NExecutionStatusSuccess) {
        //読取停止に成功
    } else {
        //読取停止に失敗
    }
    getMessageWithResultCode: メソッドを呼び出し、失敗原因を取得します。(2.2.22を参
照)
    NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
    //errorString 失敗原因。
```

```
}  
}
```

#### 1.3.5.1.4. 読取状態の実行結果を受信するデリゲート

```
- (void)AsP52N:(AsP52N *)AsP52N  
receivedInventoryStatus:(AsP52NExecutionStatus)statusCode {  
if (statusCode == AsP52NExecutionStatusSuccess) {  
//読取に成功  
} else {  
//読取に失敗  
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参  
照)  
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];  
//errorString 失敗原因  
}  
}
```

#### 1.3.5.2. startInventoryWithEPC

```
[asP52N startInventoryWithEPC:YES TID:YES PC:YES RSSI:YES phase:YES  
channel:YES antenna:YES];
```

##### 1.3.5.2.1. 読取開始の実行結果を受信するデリゲート

```
- (void)AsP52N:(AsP52N *)AsP52N  
didStartInventoryExecutionStatus:(AsP52NExecutionStatus)statusCode{  
if (statusCode == AsP52NExecutionStatusSuccess) {  
//読取開始に成功  
} else {  
//読取開始に失敗  
getMessageWithResultCode:を呼び出し、失敗原因を取得します。(2.2.22を参照)  
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];  
//errorString 失敗原因  
}  
}
```

}

### 1.3.5.2.2.読み取ったデータを受信するデリゲート

```
- (void)AsP52N:(AsP52N *)AsP52N receivedInventoryDataWithRSSI:(NSString *)RSSI channel:(NSString *)channel phase:(NSString *)phase antenna:(NSString *)antenna PC:(NSString *)PC EPC:(NSString *)EPC TID:(NSString *)TID {
    RSSI      RSSI データ
    channel   チャンネル情報
    phase     タグ位相情報
    antenna   アンテナ情報
    PC        PC データ
    EPC       EPC データ
    TID       TID データ
}
```

当該デリゲートまたは 1.3.5.2.3.

AsP52N :receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EP C:TID:userData:reserved:epcBank:moisture:temperature:を通じて読み取ったデータを受信することができます。

### 1.3.5.2.3.読み取ったデータを受信するデリゲート

```
- (void)AsP52N:(AsP52N *)AsP52N
receivedInventoryCustomDataWithRSSI:(NSString *)RSSI channel:(NSString *)channel phase:(NSString *)phase antenna:(NSString *)antenna PC:(NSString *)PC EPC:(NSString *)EPC TID:(NSString *)TID userData:(NSString *)userData reserved:(NSString *)reserved epcBank:(NSString *)epcBank moisture:(NSString *)moisture temperature:(NSString *)temperature {
    RSSI      RSSI データ
    channel   チャンネル情報
    phase     タグ位相情報
    antenna   アンテナ情報
    PC        PC データ
    EPC       EPC データ
}
```

```
TID          TID データ
userData     User データ
reserved     Reserved データ
epcBank      epcBank データ
moisture     湿度データ
temperature  温度データ
}
```

当該デリゲートまたは 1.3.5.2.2.

AsP52N:receivedInventoryDataWithRSSI:channel:phase:antenna:PC:EPC:TID:を通じて読み取ったデータを受信することができます。

#### 1.3.5.2.4 読取停止の実行結果を受信するデリゲート

```
- (void)AsP52N:(AsP52N *)AsP52N
didInventoryEndExecutionStatus:(AsP52NExecutionStatus)statusCode{
    if (statusCode == AsP52NExecutionStatusSuccess) {
//読取停止に成功
    } else {
//読取停止に失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)。
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
//errorString 失敗原因

    }
}
```

#### 1.3.5.2.5.読取情報の実行結果を受信するデリゲート

```
- (void)AsP52N:(AsP52N *)AsP52N
receivedInventoryStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//読取に成功
} else {
//読取に失敗
```

getMessageWithResultCode: メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)。

```
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
//errorString 失敗原因

}
}
```

### 1.3.6. RFID データの読取を停止する

```
[asP52N stopReadData];
読取停止を受信するデリゲート
- (void)AsP52N:(AsP52N *)AsP52N
didStopInventoryExecutionStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//読取停止に成功
} else {
//読取停止に失敗
getMessageWithResultCode: メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照) 。
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
//errorString 失敗原因
}
}
```

### 1.3.7. AsP52N オブジェクトの接続を切断する

```
[asP52N disconnectServer];
切断結果を受信するデリゲート
- (void)AsP52N:(AsP52N *)AsP52N
changeNetworkState:(AsP52NetworkState)state error:(NSError *)error{
    if (state == AsP52NetworkStateConnected) {
        //接続に成功
    } else if (state == AsP52NetworkStateDisconnected) {
        //接続を切断
    }
}
```

```
    } else if (state == AsP52NetworkStateUnknown) {  
        //未知のネットワーク  
    } else {  
        //error その他のエラー  
    }  
}
```

## 2 AsP52N クラス

AsP52N オブジェクトは ASR-P52N の接続・切断、RFID データの読取、読取停止、読取に関する設定、RFID タグに対しての読取、書き込み、ロック、キル、ハードウェア情報取得などのために使用されます。

### 2.1. 属性

属性名	属性	型	説明
delegate	nonatomic, assign	AsP52NDelegate	AsP52NDelegate ( <a href="#">2.3</a> を参照)
isConnected	nonatomic, assign, readonly	BOOL	ASR-P52N の接続状態 YES : 接続 NO : 切断

### 2.2. メソッド

#### 2.2.1. sharedInstance

+ (AsP52N *)sharedInstance;				
	引数名	In/Out	型	説明
戻り値	-	Out	AsP52N	AsP52N オブジェクト
<p>■ 説明: AsP52N オブジェクトの生成・初期化を行います (シングルトン)。</p> <p>■ サンプルコード: AsP52N *asP52N = [AsP52N sharedInstance];</p>				

#### 2.2.2. getSDKVersion

- (NSString *)getSDKVersion;
------------------------------

	引数名	In/Out	型	説明
戻り値	-	Out	NSString	SDK バージョン
<p>■ 説明:</p> <p>SDK バージョンを取得します。</p> <p>■ サンプルコード:</p> <pre>NSString *sdkString = [asP52N getSDKVersion]; //sdkString SDK バージョン。</pre>				

### 2.2.3. startSearchDevice

- (NSError *)startSearchDevice;				
	引数名	In/Out	型	説明
戻り値	-	Out	NSError	メソッドの実行結果 実行成功 : nil 実行失敗 : NSError
<p>■ 説明:</p> <p>ASR-P52N を検索します。 このメソッドを呼び出すと、「AsP52N:searchingIP:isFinish:」(<a href="#">2.3.1</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>NSError *error = [asP52N startSearchDevice]; if (error){ //メソッド実行に失敗 }else{ //メソッド実行に成功 }</pre>				

### 2.2.4. stopSearchDevice

- (NSError *)stopSearchDevice;				
	引数名	In/Out	型	説明

戻り値	-	Out	NSError	メソッドの実行結果 実行成功 : nil 実行失敗 : NSError
<p>■ 説明:</p> <p>ASR-P52N の検索を停止します。</p> <p>■ サンプルコード:</p> <pre> NSError *error = [asP52N stopSearchDevice]; if (error){ //メソッド実行に失敗 }else{ //メソッド実行に成功 } </pre>				

## 2.2.5. connectServerWithIP

<b>- (NSError *)connectServerWithIP:(NSString *)IP remoteport:(int)port;</b>				
	引数名	In/Out	型	説明
引数	IP	In	NSString	IP アドレス
引数	port	In	int	ポート (デフォルトポート : 8160)
戻り値	-	Out	NSError	メソッドの実行結果 実行成功 : nil 実行失敗 : NSError
<p>■ 説明:</p> <p>ASR-P52N を接続します。</p> <p>このメソッドを呼び出すと、「AsP52N:changeNetworkState:error:」(<a href="#">2.3.2</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <p><b>//IP アドレスは 192.168.10.10、ポートは 8160 の AsReader に接続します。</b></p> <pre> NSError *error = [asP52N connectServerWithIP:@"192.168.1.10" remoteport:8160]; if (error){ //メソッド実行に失敗 }else{ //メソッド実行に成功 } </pre>				

}

## 2.2.6. disconnectServer

- (NSError *)disconnectServer;				
	引数名	In/Out	型	説明
戻り値	-	Out	NSError	メソッドの実行結果 実行成功 : nil 実行失敗 : NSError
<p>■ 説明:</p> <p>ASR-P52N との接続を切断します。</p> <p>このメソッドを呼び出すと、「AsP52N:changeNetworkState:error:」(<a href="#">2.3.2</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>NSError *error = [asP52N disconnectServer]; if (error){ //メソッド実行に失敗 }else{ //メソッド実行に成功 }</pre>				

## 2.2.7. startInventoryWithEPC

- (void)startInventoryWithEPC:(BOOL)EPC TID:(BOOL)TID PC:(BOOL)PC RSSI:(BOOL)RSSI phase:(BOOL)phase channel:(BOOL)channel antenna:(BOOL)antenna;				
	引数名	In/Out	型	説明
引数	EPC	In	BOOL	EPC データを受信するかどうか YES: 受信する NO: 受信しない

引数	TID	In	BOOL	TID データを受信するかどうか YES: 受信する NO: 受信しない
引数	PC	In	BOOL	PC データを受信するかどうか YES: 受信する NO: 受信しない
引数	RSSI	In	BOOL	RSSI データを受信するかどうか YES: 受信する NO: 受信しない
引数	phase	In	BOOL	タグ位相情報を受信するかどうか YES: 受信する NO: 受信しない
引数	channel	In	BOOL	チャンネル情報を受信するかどうか YES: 受信する NO: 受信しない
引数	antenna	In	BOOL	アンテナ情報を受信するかどうか YES: 受信する NO: 受信しない
<p>■ 説明:</p> <p>インベントリを開始します。</p> <p>このメソッドを呼び出すと、「AsP52N:didStartInventoryExecutionStatus:」(<a href="#">2.3.3</a>を参照)、 「AsP52N:receivedInventoryDataWithRSSI:channel:phase:antenna:PC:EPC:TID:」(<a href="#">2.3.5</a> を参照)、 「AsP52N:didInventoryEndExecutionStatus:」(<a href="#">2.3.18</a>を参照)、 「AsP52N:receivedInventoryStatus:」(<a href="#">2.3.38</a>を参照)、 「AsP52N:receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EPC:TID:u serData:reserved:epcBank:moisture:temperature:」(<a href="#">2.3.39</a>を参照) デリゲートがコールバックさ れます。</p> <p>■ サンプルコード:</p> <pre>[asP52N startInventoryWithEPC:YES TID:YES PC:YES RSSI:YES phase:YES channel:YES antenna:YES];</pre>				

## 2.2.8. stopInventory

- (void)stopInventory;

■説明:

読取に関する全ての操作を停止します。

このメソッドを呼び出すと、「AsP52N:didStopInventoryExecutionStatus:」([2.3.4](#)を参照) デリゲートがコールバックされます。

■サンプルコード:

```
[asP52N stopInventory];
```

## 2.2.9. readTagWithAccessPassword

```
- (void)readTagWithAccessPassword:(uint32_t)accessPassword
                        EPC:(NSData *)EPC
                memoryBank:(AsP52NMemoryBankType)memoryBank
        startAddress:(uint16_t)startAddress
                dataLength:(uint16_t)dataLength;
```

	引数名	In/O ut	型	説明
引数	accessPassword	In	uint32_t	読取対象タグのアクセスパスワード デフォルト値 : 0x00000000
引数	EPC	In	NSData	読取対象タグの EPC データ
引数	memoryBank	In	AsP52NMemoryBankType	読取対象のメモリバンク AsP52NmemoryBankType 列挙型 ( <a href="#">2.4.3</a> を参照)
引数	startAddress	In	uint16_t	読取先メモリのスタートアドレス 単位 : word
引数	dataLength	In	uint16_t	読取メモリの長さ 単位 : word

■説明:

タグの対象メモリバンクのデータを読み取ります。

このメソッドを呼び出すと、「AsP52N:didStartInventoryExecutionStatus:」([2.3.3](#)を参照)、

「AsP52N:didReadTagExecutionStatus:」([2.3.9](#)を参照)、「AsP52N:

receivedTagOperationData:」([2.3.10](#)を参照)、

「AsP52N:didInventoryEndExecutionStatus:」([2.3.18](#)を参照)、

「AsP52N:receivedInventoryStatus:」([2.3.38](#)を参照)、

「AsP52N:receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EPC:TID:u

serData:reserved:epcBank:moisture:temperature:」(2.3.39 を参照) デリゲートがコールバックされます。

■ サンプルコード:

```
NSData *data ; //data EPC データ。
```

```
[asP52N readTagWithAccessPassword:00000000 EPC:data
```

```
memoryBank:AsP52NMemoryBankTypeEPC startAddress:0 dataLength:1];
```

## 2.2.10. writeTagWithAccessPassword

```
- (void)writeTagWithAccessPassword:(uint32_t)accessPassword
      EPC:(NSData *)EPC
      memoryBank:(AsP52NMemoryBankType)memoryBank
      startAddress:(uint16_t)startAddress
      writeData:(NSData *)writeData;
```

	引数名	In/Out	型	説明
引数	accessPassword	In	uint32_t	書込対象タグのアクセスパスワード デフォルト値 : 0x00000000
引数	EPC	In	NSData	書込対象タグの EPC データ
引数	memoryBank	In	AsP52NMemoryBankType	AsP52NMemoryBankType 列挙型 (2.4.3 を参照)
引数	startAddress	In	uint16_t	書き込み先のメモリのスタートアドレス 単位 : word
引数	writeData	In	NSData	書き込みデータ

■ 説明:

タグの対象メモリ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、「AsP52N:didWriteTagExecutionStatus:」(2.3.7 を参照) デリゲートがコールバックされます。

■ サンプルコード:

```
NSData *data ; //data EPC データ。
```

```
NSData *writeData ; //writeData 書き込み EPC データ。
```

```
[asP52N writeTagWithAccessPassword:00000000 EPC:data
```

```
memoryBank:AsP52NMemoryBankTypeEPC startAddress:0 writeData: writeData];
```

### 2.2.11. lockTagWithAccessPassword

- (void)lockTagWithAccessPassword:(uint32_t)accessPassword EPC:(NSData *)EPC targetMemory:(AsP52NTargetMemoryType)targetMemory lockAction:(AsP52NLockActionType)lockAction;				
	引数名	In/Out	型	説明
引数	accessPassword	In	uint32_t	ロック対象タグのアクセスパスワード デフォルト値 : 0x00000000
引数	EPC	In	NSData	ロック対象タグの EPC データ
引数	targetMemory	In	AsP52NTargetMemoryType	ロック対象のメモリバンク AsP52NTargetMemoryType 列挙型 ( <a href="#">2.4.4</a> を参照)
引数	lockAction	In	AsP52NLockActionType	ロックタイプ AsP52NLockActionType 列挙型 ( <a href="#">2.4.5</a> を参照)
<p>■ 説明:</p> <p>タグのメモリバンクをロック・アンロックします。 このメソッドを呼び出すと、「AsP52N:didLockTagExecutionStatus:」(<a href="#">2.3.8</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>NSData *data ; //data EPC データ。 [asP52N lockTagWithAccessPassword:12345678 EPC:data targetMemory:AsP52NTargetMemoryTypeEPC lockAction:AsP52NLockActionTypeLock];</pre>				

### 2.2.12. killTagWithKillPassword

- (void)killTagWithKillPassword:(uint32_t)killPassword EPC:(NSData *)EPC;				
	引数名	In/Out	型	説明
引数	killPassword	In	uint32_t	キル対象タグのキルパスワード パスワードは 0x00000000 に設定している 場合、キル操作無効

引数	EPC	In	NSData	キル対象タグの EPC データ
<p>■ 説明:</p> <p>対象タグをキルします。</p> <p>このメソッドを呼び出すと、「AsP52N:didKillTagExecutionStatus:」(<a href="#">2.3.6</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>NSData *data ; //data EPC データ。 [asP52N killTagWithKillPassword:12345678 EPC:data];</pre>				

### 2.2.13. setAntennaPortsStatusWithAntenna1

<p>- (void)setAntennaPortsStatusWithAntenna1:(BOOL)antenna1IsOpen  antenna2:(BOOL)antenna2IsOpen  antenna3:(BOOL)antenna3IsOpen  antenna4:(BOOL)antenna4IsOpen  antenna5:(BOOL)antenna5IsOpen  antenna6:(BOOL)antenna6IsOpen  antenna7:(BOOL)antenna7IsOpen  antenna8:(BOOL)antenna8IsOpen;</p>				
	引数名	In/Out	型	説明
引数	antenna1IsOpen	In	BOOL	アンテナ 1 の使用有無を設定します。 YES : 使用する NO : 使用しない
引数	antenna2IsOpen	In	BOOL	アンテナ 2 の使用有無を設定します。 YES : 使用する NO : 使用しない
引数	antenna3IsOpen	In	BOOL	アンテナ 3 の使用有無を設定します。 YES : 使用する NO : 使用しない
引数	antenna4IsOpen	In	BOOL	アンテナ 4 の使用有無を設定します。 YES : 使用する NO : 使用しない
引数	antenna5IsOpen	In	BOOL	アンテナ 5 の使用有無を設定します。 YES : 使用する

				NO : 使用しない
引数	antenna6IsOpen	In	BOOL	アンテナ 6 の使用有無を設定します。 YES : 使用する NO : 使用しない
引数	antenna7IsOpen	In	BOOL	アンテナ 7 の使用有無を設定します。 YES : 使用する NO : 使用しない
引数	antenna8IsOpen	In	BOOL	アンテナ 8 の使用有無を設定します。 YES : 使用する NO : 使用しない
<p>■ 説明:</p> <p>アンテナの使用有無を設定します。 このメソッドを呼び出すと、「AsP52N:didSetAntennaPortsStatusExecutionStatus:」(<a href="#">2.3.11</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>[asP52N setAntennaPortsStatusWithAntenna1:YES antenna2:YES antenna3:YES antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES];</pre>				

## 2.2.14. getAntennaPortsStatus

<b>- (void)getAntennaPortsStatus;</b>
<p>■ 説明:</p> <p>アンテナのポート状態を取得します。 このメソッドを呼び出すと、 「AsP52N:receivedAntennaPortsStatus:antenna2:antenna3:antenna4:antenna5:antenna6:antenna7:antenna8:executionStatus:」(<a href="#">2.3.12</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>[asP52N getAntennaPortsStatus];</pre>

## 2.2.15. setOutputPowerLevel

<b>- (void)setOutputPowerLevel:(uint32_t)powerLevel;</b>
--

	引数名	In/Out	型	説明
引数	powerLevel	In	uint32_t	電波出力の設定値 (dBm) 範囲 : 0~36
<p>■ 説明:</p> <p>AsReader からの電波出力を設定します。</p> <p>このメソッドを呼び出すと、「AsP52N:didSetOutputPowerLevelExecutionStatus」(<a href="#">2.3.13</a>を参照)、「AsP52N:receivedSetAntennaPowerStatus」(<a href="#">2.3.21</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>[asP52N setOutputPowerLevel:36];</pre>				

### 2.2.16. getOutputPowerLevel

- (void)getOutputPowerLevel;				
<p>■ 説明:</p> <p>AsReader からの電波出力を取得します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedOutputPowerLevel:executionStatus:」(<a href="#">2.3.14</a>を参照)、「AsP52N:receivedReaderAntenna1Power:antenna2Power:antenna3Power:antenna4Power:antenna5Power:antenna6Power:antenna7Power:antenna8Power:」(<a href="#">2.3.20</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>[asP52N getOutputPowerLevel];</pre>				

### 2.2.17. setSession

- (void)setSession:(AsP52NSessionType)session;				
	引数名	In/Out	型	説明
引数	session	In	AsP52NSessionType	インベントリ処理実行時の Session 値 AsP52NSessionType 列挙型 ( <a href="#">2.4.2</a> を参照)
<p>■ 説明:</p>				

インベントリ処理実行時の Session 値を設定します。

このメソッドを呼び出すと、「AsP52N:didSetSessionExecutionStatus:」(2.3.15を参照)、  
「AsP52N:receivedSetBasebandStatus:」(2.3.23を参照) デリゲートがコールバックされます。

■ サンプルコード:

```
[asP52N setSession:AsP52NSessionType_S0];
```

## 2.2.18. getSession

- (void)getSession;

■ 説明:

インベントリ処理実行時の Session 値を取得します。

このメソッドを呼び出すと、「AsP52N:receivedSession:executionStatus:」(2.3.16を参照)、  
「AsP52N: receivedReaderBasebandRate:QValue:session:inventoryFlag:」(2.3.22を参照)  
デリゲートがコールバックされます。

■ サンプルコード:

```
[asP52N getSession];
```

## 2.2.19. setDHCPForMACAddress

- (BOOL)setDHCPForMACAddress:(NSString \*)MACAddress;

	引数名	In/Out	型	説明
引数	MACAddress	In	NSString	AsReader の MAC アドレス (フォーマットは標準の MAC アドレスになる)
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗

■ 説明:

AsReader の MAC アドレスを設定します。

■ サンプルコード:

```
BOOL status = [asP52N setDHCPForMACAddress:@"AC:BC:32:02:8D:32"];
if (status) {
//コマンド実行に成功。
```

```

} else {
//コマンド実行に失敗。
}

```

## 2.2.20. setDHCPForMACAddress

```

- (BOOL)setDHCPForMACAddress:(NSString *)MACAddress
      IP:(NSString *)IP
      subnet:(NSString *)subnet
      gateway:(NSString *)gateway
      DNSServer:(NSString *)DNSServer;

```

	引数名	In/Out	型	説明
引数	MACAddress	In	NSString	AsReaderのMACアドレス（フォーマットは標準のMACアドレスになる）
引数	IP	In	NSString	IPアドレス
引数	subnet	In	NSString	サブネットマスク
引数	gateway	In	NSString	ゲートウェイ
引数	DNSServer	In	NSString	DNS
戻り値	-	Out	BOOL	コマンドの実行結果 YES：実行成功 NO：実行失敗

### ■説明:

AsReaderのMACアドレスを設定します。

### ■サンプルコード:

```

BOOL status = [asP52N setDHCPForMACAddress:@"AC:BC:32:02:8D:32"
IP:@"192.168.1.10" subnet:@"255.255.255.255" gateway:@"255.255.255.255"
DNSServer:@"255.255.255.255"];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

### 2.2.21. getFirmwareVersion

- (void)getFirmwareVersion;

■ 説明:

ファームウェアバージョンを取得します。

このメソッドを呼び出すと、「AsP52N:receivedFirmwareVersion:executionStatus:」([2.3.17](#)を参照) デリゲートがコールバックされます。

■ サンプルコード:

```
[asP52N getFirmwareVersion];
```

### 2.2.22. getMessageWithResultCode

- (NSString \*)getMessageWithResultCode:(AsP52NExecutionStatus)resultCode;

	引数名	In/Out	型	説明
引数	resultCode	In	AsP52NExecutionStatus	AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)。
戻り値	-	Out	NSString	列挙パラメータの説明

■ 説明:

状態コードを説明テキストに変換します。

■ サンプルコード:

```
NSString *string = [asP52N
getMessageWithResultCode:AsP52NExecutionStatusSuccess];
// string 列挙パラメータの説明。
```

### 2.2.23. setSearchDeviceTimeout

- (void)setSearchDeviceTimeout:(NSTimeInterval)timeout;

	引数名	In/Out	型	説明
引数	timeout	In	NSTimeInterval	タイムアウト時間 (単位 : s) 範囲 : 10~60

■ 説明:

AsReader を検索時のタイムアウト時間を設定します。

■ サンプルコード:

```
[asP52N setSearchDeviceTimeout:60];
```

## 2.2.24. readDataWithAntenna1

```
- (BOOL)readDataWithAntenna1:(BOOL)antenna1
    antenna2:(BOOL)antenna2
    antenna3:(BOOL)antenna3
    antenna4:(BOOL)antenna4
    antenna5:(BOOL)antenna5
    antenna6:(BOOL)antenna6
    antenna7:(BOOL)antenna7
    antenna8:(BOOL)antenna8
    isContinueScan:(BOOL)isContinueScan
    isUsedTID:(BOOL)isUsedTID
    isUsedTidLength:(BOOL)isUsedTidLength
    tidLength:(int)tidLength
    isUsedUserData:(BOOL)isUsedUserData
    userDataOffset:(int)userDataOffset
    userDataLength:(int)userDataLength
    isUsedReserved:(BOOL)isUsedReserved
    reservedOffset:(int)reservedOffset
    reservedLength:(int)reservedLength
    isUsedPassword:(BOOL)isUsedPassword
    accessPassword:(NSString *)accessPassword
    isUsedMonza:(BOOL)isUsedMonza
    isUsedRfmicron:(BOOL)isUsedRfmicron
    isUsedEMSensor:(BOOL)isUsedEMSensor
    isUsedEPC:(BOOL)isUsedEPC
    epcOffset:(int)epcOffset
    epcLength:(int)epcLength
    isUsedMatch:(BOOL)isUsedMatch

    matchMemoryBankType:(AsP52NMemoryBankType)matchMemoryBankType
    matchData:(NSString *)matchData
    matchStartAddress:(uint16_t)matchStartAddress;
```

	引数名	In/Out	型	説明
引数	antenna1	In	BOOL	アンテナ 1 を使用してデータを読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna2	In	BOOL	アンテナ 2 を使用してデータを読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna3	In	BOOL	アンテナ 3 を使用してデータを読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna4	In	BOOL	アンテナ 4 を使用してデータを読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna5	In	BOOL	アンテナ 5 を使用してデータを読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna6	In	BOOL	アンテナ 6 を使用してデータを読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna7	In	BOOL	アンテナ 7 を使用してデータを読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna8	In	BOOL	アンテナ 8 を使用してデータを読み取るかどうか YES : 使用する NO : 使用しない
引数	isContinueScan	In	BOOL	データを連続して読み取るかどうか

				YES : 連続して読み取る NO : 連続して読み取りしない
引数	isUsedTID	In	BOOL	TID データを読み取るかどうか YES : 読み取る NO : 読み取りしない
引数	isUsedTidLength	In	BOOL	TID データの長さ指定を使用するかどうか YES : 読み取る NO : 読み取りしない
引数	tidLength	In	int	TID の読み取りデータ長
引数	isUsedUserData	In	BOOL	User エリアのデータを読み取るかどうか YES : 読み取る NO : 読み取りしない
引数	userDataOffset	In	int	User エリアの読み取りスタートアドレス 単位 : word
引数	userDataLength	In	int	User エリアの読み取りデータ長 単位 : word
引数	isUsedReserved	In	BOOL	Reserved エリアのデータを読み取るかどうか YES : 読み取る NO : 読み取りしない
引数	reservedOffset	In	int	Reserved エリアの読み取りスタートアドレス 単位 : word
引数	reservedLength	In	int	Reserved エリアの読み取りデータ長 単位 : word
引数	isUsedPassword	In	BOOL	パスワードを使用するかどうか YES : 使用する NO : 使用しない
引数	accessPassword	In	NSString	タグのアクセスパスワード
引数	isUsedMonza	In	BOOL	Monza QT Peek を読み取るかどうか YES : 読み取る NO : 読み取りしない
引数	isUsedRfmicron	In	BOOL	Rfmicron 温度センサーを読み取るかどうか

				か YES : 読み取る NO : 読み取りしない
引数	isUsedEMSensor	In	BOOL	Em センサーデータを読み取るかどうか YES : 読み取る NO : 受信しない
引数	isUsedEPC	In	BOOL	EPC データを読み取るかどうか。 YES : 読み取る NO : 受信しない
引数	epcOffset	In	int	EPC データの読み取りスタートアドレス 単位 : word
引数	epcLength	In	int	EPC データの読み取りデータ長 単位 : word
引数	isUsedMatch	In	BOOL	読み取り対象タグのマッチングを使用する かどうか YES : 使用する NO : 使用しない
引数	matchMemoryBankType	In	AsP52NMemoryBankType	マッチングの対象メモリバンク AsP52NMemoryBankType 列挙型 ( <a href="#">2.4.3</a> を参照)
引数	matchData	In	NSString	マッチングデータ
引数	matchStartAddress	In	uint16_t	マッチング対象のメモリバンクのスタートアドレス 単位 : word
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗

■ 説明:

対象タグを読み取ります。

このメソッドを呼び出すと、「AsP52N:didStartInventoryExecutionStatus:」([2.3.3](#)を参照)、

「AsP52N:didInventoryEndExecutionStatus:」([2.3.18](#)を参照)、

「AsP52N:receivedInventoryStatus:」([2.3.38](#)を参照)、

「AsP52N:receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EPC:TID:userData:reserved:epcBank:moisture:temperature:」([2.3.39](#)を参照)、

receivedReadDataStatus:」 ([2.3.41](#) を参照) デリゲートがコールバックされます。

■ サンプルコード:

```
NSString *data ; //data 読取データ。
BOOL status = [asP52N readDataWithAntenna1:YES antenna2:YES antenna3:YES
antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES
isContinueScan:YES isUsedTID:YES isUsedTidLength:YES tidLength:6
isUsedUserData:NO userDataOffset:0 userDataLength:0 isUsedReserved:NO
reservedOffset:0 reservedLength:0 isUsedPassword:YES accessPassword:@"00000000"
isUsedMonza:NO isUsedRfmicron:NO isUsedEMSensor:NO isUsedEPC:NO epcOffset:0
epcLength:0 isUsedMatch:YES matchMemoryBankType:AsP52NMemoryBankTypeTID
matchData:data matchStartAddress:1];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}
```

## 2.2.25. writeDataWithAntenna1

```
- (BOOL)writeDataWithAntenna1:(BOOL)antenna1
    antenna2:(BOOL)antenna2
    antenna3:(BOOL)antenna3
    antenna4:(BOOL)antenna4
    antenna5:(BOOL)antenna5
    antenna6:(BOOL)antenna6
    antenna7:(BOOL)antenna7
    antenna8:(BOOL)antenna8

writeMemoryBankType:(AsP52NMemoryBankType)writeMemoryBankType
    writeData:(NSString *)writeData
writeStartAddress:(uint16_t)writeStartAddress
    isUsedMatch:(BOOL)isUsedMatch

matchMemoryBankType:(AsP52NMemoryBankType)matchMemoryBankType
    matchData:(NSString *)matchData
matchStartAddress:(uint16_t)matchStartAddress
    accessPassword:(NSString *)accessPassword;
```

	引数名	In/Out	型	説明
引数	antenna1	In	BOOL	アンテナ 1 を使用してデータを書き込むかどうか YES : 使用する NO : 使用しない
引数	antenna2	In	BOOL	アンテナ 2 を使用してデータを書き込むかどうか YES : 使用する NO : 使用しない
引数	antenna3	In	BOOL	アンテナ 3 を使用してデータを書き込むかどうか YES : 使用する NO : 使用しない
引数	antenna4	In	BOOL	アンテナ 4 を使用してデータを書き込むかどうか YES : 使用する NO : 使用しない
引数	antenna5	In	BOOL	アンテナ 5 を使用してデータを書き込むかどうか YES : 使用する NO : 使用しない
引数	antenna6	In	BOOL	アンテナ 6 を使用してデータを書き込むかどうか YES : 使用する NO : 使用しない
引数	antenna7	In	BOOL	アンテナ 7 を使用してデータを書き込むかどうか YES : 使用する NO : 使用しない
引数	antenna8	In	BOOL	アンテナ 8 を使用してデータを書き込むかどうか YES : 使用する NO : 使用しない
引数	writeMemoryBank	In	AsP52NMemoryB	書き込み対象タグのメモリバンク

	Type		ankType	AsP52NMemoryBankType 列挙型 (2.4.3を参照)
引数	writeData	In	NSString	書き込みデータ
引数	writeStartAddress	In	uint16_t	書き込み対象タグのメモリバンクのスタートアドレス (単位: word)
引数	isUsedMatch	In	BOOL	書き込み対象タグのマッチングを使用するかどうか YES: 使用する NO: 使用しない
引数	matchMemoryBankType	In	AsP52NMemoryBankType	マッチング対象のメモリバンク AsP52NMemoryBankType 列挙型 (2.4.3を参照)
引数	matchData	In	NSString	マッチングデータ
引数	matchStartAddress	In	uint16_t	マッチング対象のメモリバンクのスタートアドレス (単位: word)
引数	accessPassword	In	NSString	対象タグのアクセスパスワード
戻り値	-	Out	BOOL	コマンドの実行結果 YES: 実行に成功 NO: 実行失敗

■ 説明:

対象タグにデータ書き込みます。

このメソッドを呼び出すと、「AsP52N:receivedWriteDataStatus:」 ([2.3.40](#)を参照) デリゲートがコールバックされます。

■ サンプルコード:

```
NSString *data ; //data 現在のデータ。
```

```
NSString *writeData ; // writeData 書き込みデータ。
```

```
BOOL status = [asP52N writeDataWithAntenna1:YES antenna2:YES antenna3:YES
```

```
antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES
```

```
writeMemoryBankType:AsP52NMemoryBankTypeEPC writeData: writeData
```

```
writeStartAddress:0 isUsedMatch:YES
```

```
matchMemoryBankType:AsP52NMemoryBankTypeEPC matchData: data
```

```
matchStartAddress:0 accessPassword:@"00000000"];
```

```
if (status) {
```

```
//コマンド実行に成功。
```

```

} else {
//コマンド実行に失敗。
}

```

## 2.2.26. lockTagWithAntenna1

```

- (BOOL)lockTagWithAntenna1:(BOOL)antenna1
    antenna2:(BOOL)antenna2
    antenna3:(BOOL)antenna3
    antenna4:(BOOL)antenna4
    antenna5:(BOOL)antenna5
    antenna6:(BOOL)antenna6
    antenna7:(BOOL)antenna7
    antenna8:(BOOL)antenna8
    targetMemory:(AsP52NTargetMemoryType)targetMemory
    lockAction:(AsP52NLockActionType)lockAction
    isUsedMatch:(BOOL)isUsedMatch

    matchMemoryBankType:(AsP52NMemoryBankType)matchMemoryBankType
    matchData:(NSString *)matchData
    matchStartAddress:(uint16_t)matchStartAddress
    accessPassword:(NSString *)accessPassword;

```

	引数名	In/Out	型	説明
引数	antenna1	In	BOOL	アンテナ 1 を使用してタグをロックするかどうか YES : 使用する NO : 使用しない
引数	antenna2	In	BOOL	アンテナ 2 を使用してタグをロックするかどうか YES : 使用する NO : 使用しない
引数	antenna3	In	BOOL	アンテナ 3 を使用してタグをロックするかどうか YES : 使用する NO : 使用しない

引数	antenna4	In	BOOL	アンテナ 4 を使用してタグをロックするかどうか YES : 使用する NO : 使用しない
引数	antenna5	In	BOOL	アンテナ 5 を使用してタグをロックするかどうか YES : 使用する NO : 使用しない
引数	antenna6	In	BOOL	アンテナ 6 を使用してタグをロックするかどうか YES : 使用する NO : 使用しない
引数	antenna7	In	BOOL	アンテナ 7 を使用してタグをロックするかどうか YES : 使用する NO : 使用しない
引数	antenna8	In	BOOL	アンテナ 8 を使用してタグをロックするかどうか YES : 使用する NO : 使用しない
引数	targetMemory	In	AsP52NTargetMemoryType	ロック操作のメモリバンク AsP52NTargetMemoryType 列挙型 ( <a href="#">2.4.4</a> を参照)
引数	lockAction	In	AsP52NLockActionType	ロック操作のタイプ AsP52NLockActionType 列挙型 ( <a href="#">2.4.5</a> を参照)
引数	isUsedMatch	In	BOOL	ロック操作対象タグのマッチングを使用するかどうか YES : 使用する NO : 使用しない
引数	matchMemoryBankType	In	AsP52NMemoryBankType	マッチング対象のメモリバンク AsP52NMemoryBankType 列挙型 ( <a href="#">2.4.3</a> を参照)

引数	matchData	In	NSString	マッチングデータ
引数	matchStartAddress	In	uint16_t	マッチング対象のメモリバンクのスタートアドレス (単位: word)
引数	accessPassword	In	NSString	対象タグのアクセスパスワード
戻り値	-	Out	BOOL	コマンドの実行結果 YES: 実行成功 NO: 実行失敗

■ 説明:

対象タグをロックします。

このメソッドを呼び出すと、「AsP52N:receivedLockDataStatus:」([2.3.42](#)を参照) デリゲートがコールバックされます。

■ サンプルコード:

NSString \*data ; //data 現在のデータ。

```

BOOL status = [asP52N lockTagWithAntenna1:YES antenna2:YES antenna3:YES
antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES
targetMemory:AsP52NTargetMemoryTypeEPC lockAction:AsP52NLockActionTypeLock
isUsedMatch:YES matchMemoryBankType:AsP52NMemoryBankTypeEPC
matchData:data matchStartAddress:0 accessPassword:@" 12345678"];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

## 2.2.27. killTagWithAntenna1

```

- (BOOL)killTagWithAntenna1:(BOOL)antenna1
antenna2:(BOOL)antenna2
antenna3:(BOOL)antenna3
antenna4:(BOOL)antenna4
antenna5:(BOOL)antenna5
antenna6:(BOOL)antenna6
antenna7:(BOOL)antenna7
antenna8:(BOOL)antenna8

```

<b>killPassword:(NSString *)killPassword</b> <b>isUsedMatch:(BOOL)isUsedMatch</b>  <b>matchMemoryBankType:(AsP52NMemoryBankType)matchMemoryBankType</b> <b>matchData:(NSString *)matchData</b> <b>matchStartAddress:(uint16_t)matchStartAddress;</b>				
	引数名	In/Out	型	説明
引数	antenna1	In	BOOL	アンテナ 1 を使用してタグをキルするかどうか YES : 使用する NO : 使用しない
引数	antenna2	In	BOOL	アンテナ 2 を使用してタグをキルするかどうか YES : 使用する NO : 使用しない
引数	antenna3	In	BOOL	アンテナ 3 を使用してタグをキルするかどうか YES : 使用する NO : 使用しない
引数	antenna4	In	BOOL	アンテナ 4 を使用してタグをキルするかどうか YES : 使用する NO : 使用しない
引数	antenna5	In	BOOL	アンテナ 5 を使用してタグをキルするかどうか YES : 使用する NO : 使用しない
引数	antenna6	In	BOOL	アンテナ 6 を使用してタグをキルするかどうか YES : 使用する NO : 使用しない
引数	antenna7	In	BOOL	アンテナ 7 を使用してタグをキルするかどうか YES : 使用する

				NO : 使用しない
引数	antenna8	In	BOOL	アンテナ 8 を使用してタグをキルするかどうか YES : 使用する NO : 使用しない
引数	killPassword	In	NSString	デフォルト値 0x00000000 でタグをキルできない
引数	isUsedMatch	In	BOOL	キル対象タグのマッチングを使用するかどうか YES : 使用する NO : 使用しない
引数	matchMemoryBankType	In	AsP52NMemoryBankType	マッチング対象のメモリバンク AsP52NMemoryBankType 列挙型 ( <a href="#">2.4.3</a> を参照)。
引数	matchData	In	NSString	マッチングデータ
引数	matchStartAddresses	In	uint16_t	マッチング対象のメモリバンクのスタートアドレス (単位 : word)
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗

■説明:

対象タグをキルします。

このメソッドを呼び出すと、「AsP52N:receivedKillDataStatus:」([2.3.43](#)を参照) デリゲートがコールバックされます。

■サンプルコード:

```
NSString *data ; //data 現在のデータ。
```

```
BOOL status = [asP52N killTagWithAntenna1:YES antenna2:YES antenna3:YES  
antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES
```

```
killPassword:@"88888888" isUsedMatch:YES
```

```
matchMemoryBankType:AsP52NMemoryBankTypeEPC matchData:data
```

```
matchStartAddress:0];
```

```
if (status) {
```

```
//コマンド実行に成功。
```

```
} else {
```

```
//コマンド実行に失敗。
}
```

## 2.2.28. startInventoryWithAntenna1

```
- (BOOL)startInventoryWithAntenna1:(BOOL)antenna1
    antenna2:(BOOL)antenna2
    antenna3:(BOOL)antenna3
    antenna4:(BOOL)antenna4
    antenna5:(BOOL)antenna5
    antenna6:(BOOL)antenna6
    antenna7:(BOOL)antenna7
    antenna8:(BOOL)antenna8
    isContinueScan:(BOOL)isContinueScan;
```

	引数名	In/Out	型	説明
引数	antenna1	In	BOOL	アンテナ 1 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna2	In	BOOL	アンテナ 2 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna3	In	BOOL	アンテナ 3 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna4	In	BOOL	アンテナ 4 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna5	In	BOOL	アンテナ 5 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna6	In	BOOL	アンテナ 6 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna7	In	BOOL	アンテナ 7 を使用して読み取るかどうか

				YES : 使用する NO : 使用しない
引数	antenna8	In	BOOL	アンテナ 8 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	isContinueScan	In	BOOL	データを連続して読み取るかどうか YES : 連続して読み取る NO : 連続して読取らない
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗

■ 説明:

タグの EPC データを読み取ります。

このメソッドを呼び出すと、「AsP52N:didStartInventoryExecutionStatus:」 ([2.3.3](#) を参照) 、

「AsP52N:didInventoryEndExecutionStatus:」 ([2.3.18](#) を参照) 、

「AsP52N:receivedInventoryStatus:」 ([2.3.38](#) を参照) 、

「AsP52N:receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EPC:TID:userData:reserved:epcBank:moisture:temperature:」 ([2.3.39](#) を参照) デリゲートがコールバックされます。

■ サンプルコード:

```

BOOL status = [asP52N startInventoryWithAntenna1:YES antenna2:YES antenna3:YES
antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES
isContinueScan:YES];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

## 2.2.29. startInventoryTIDWithAntenna1

- (BOOL)startInventoryTIDWithAntenna1:(BOOL)antenna1  
antenna2:(BOOL)antenna2  
antenna3:(BOOL)antenna3  
antenna4:(BOOL)antenna4

<b>antenna5:(BOOL)antenna5</b> <b>antenna6:(BOOL)antenna6</b> <b>antenna7:(BOOL)antenna7</b> <b>antenna8:(BOOL)antenna8</b> <b>isContinueScan:(BOOL)isContinueScan;</b>				
	引数名	In/Out	型	説明
引数	antenna1	In	BOOL	アンテナ 1 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna2	In	BOOL	アンテナ 2 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna3	In	BOOL	アンテナ 3 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna4	In	BOOL	アンテナ 4 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna5	In	BOOL	アンテナ 5 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna6	In	BOOL	アンテナ 6 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna7	In	BOOL	アンテナ 7 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna8	In	BOOL	アンテナ 8 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	isContinueScan	In	BOOL	データを連続して読み取るかどうか YES : 連続して読み取る NO : 連続して読取らない
戻り値	-	Out	BOOL	コマンドの実行結果

				YES : 実行成功 NO : 実行失敗
<p>■ <b>説明:</b></p> <p>タグの TID データを読み取ります。</p> <p>このメソッドを呼び出すと、「AsP52N:didStartInventoryExecutionStatus:」(2.3.3を参照)、 「AsP52N:didInventoryEndExecutionStatus:」(2.3.18を参照)、 「AsP52N:receivedInventoryStatus:」(2.3.38を参照)、 「AsP52N:receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EPC:TID:userData:reserved:epcBank:moisture:temperature:」(2.3.39を参照) デリゲートがコールバックされます。</p> <p>■ <b>サンプルコード: (注:asP52N は AsP52N クラスのインスタンスオブジェクトです)</b></p> <pre> BOOL status = [asP52N startInventoryTIDWithAntenna1:YES antenna2:YES antenna3:YES antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES isContinueScan:YES]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre>				

### 2.2.30. startInventoryMoistureWithAntenna1

<p>- (BOOL)startInventoryMoistureWithAntenna1:(BOOL)antenna1  antenna2:(BOOL)antenna2  antenna3:(BOOL)antenna3  antenna4:(BOOL)antenna4  antenna5:(BOOL)antenna5  antenna6:(BOOL)antenna6  antenna7:(BOOL)antenna7  antenna8:(BOOL)antenna8  isContinueScan:(BOOL)isContinueScan;</p>				
	引数名	In/Out	型	説明
引数	antenna1	In	BOOL	アンテナ 1 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna2	In	BOOL	アンテナ 2 を使用して読み取るかどうか

				YES : 使用する NO : 使用しない
引数	antenna3	In	BOOL	アンテナ 3 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna4	In	BOOL	アンテナ 4 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna5	In	BOOL	アンテナ 5 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna6	In	BOOL	アンテナ 6 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna7	In	BOOL	アンテナ 7 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna8	In	BOOL	アンテナ 8 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	isContinueScan	In	BOOL	データを連続して読み取るかどうか YES : 連続して読み取る NO : 連続して読取らない
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗

■ 説明:

湿度タグを読み取ります。

このメソッドを呼び出すと、「AsP52N:didStartInventoryExecutionStatus:」([2.3.3](#)を参照)、

「AsP52N:didInventoryEndExecutionStatus:」([2.3.18](#)を参照)、

「AsP52N:receivedInventoryStatus:」([2.3.38](#)を参照)、

「AsP52N:receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EPC:TID:userData:reserved:epcBank:moisture:temperature:」([2.3.39](#)を参照) デリゲートがコールバックされます。

## ■ サンプルコード:

```

BOOL status = [asP52N startInventoryMoistureWithAntenna1:YES antenna2:YES
antenna3:YES antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES
isContinueScan:YES];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

## 2.2.31. startInventoryMagnusS3WithAntenna1

```

- (BOOL)startInventoryMagnusS3WithAntenna1:(BOOL)antenna1
antenna2:(BOOL)antenna2
antenna3:(BOOL)antenna3
antenna4:(BOOL)antenna4
antenna5:(BOOL)antenna5
antenna6:(BOOL)antenna6
antenna7:(BOOL)antenna7
antenna8:(BOOL)antenna8
isContinueScan:(BOOL)isContinueScan;

```

	引数名	In/Out	型	説明
引数	antenna1	In	BOOL	アンテナ 1 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna2	In	BOOL	アンテナ 2 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna3	In	BOOL	アンテナ 3 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna4	In	BOOL	アンテナ 4 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna5	In	BOOL	アンテナ 5 を使用して読み取るかどうか

				YES : 使用する NO : 使用しない
引数	antenna6	In	BOOL	アンテナ 6 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna7	In	BOOL	アンテナ 7 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	antenna8	In	BOOL	アンテナ 8 を使用して読み取るかどうか YES : 使用する NO : 使用しない
引数	isContinueScan	In	BOOL	データを連続して読み取るかどうか YES : 連続して読み取る NO : 連続して読取らない
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗

■ 説明:

MagnusS3 タグを読み取ります。

このメソッドを呼び出すと、「AsP52N:didStartInventoryExecutionStatus:」 ([2.3.3](#) を参照) 、

「AsP52N:didInventoryEndExecutionStatus:」 ([2.3.18](#) を参照) 、

「AsP52N:receivedInventoryStatus:」 ([2.3.38](#) を参照) 、

「AsP52N:receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EPC:TID:userData:reserved:epcBank:moisture:temperature:」 ([2.3.39](#) を参照) デリゲートがコールバックされます。

■ サンプルコード:

```
BOOL status = [asP52N startInventoryMagnusS3WithAntenna1:YES antenna2:YES
antenna3:YES antenna4:YES antenna5:YES antenna6:YES antenna7:YES antenna8:YES
isContinueScan:YES];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}
```

## 2.2.32. stopReadData

- (BOOL)stopReadData;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>タグの読取を停止します。</p> <p>このメソッドを呼び出すと、「AsP52N:didStopInventoryExecutionStatus:」(<a href="#">2.3.4</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>BOOL status = [asP52N stopReadData]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre>				

## 2.2.33. setScanContinuous

- (void)setScanContinuous:(BOOL)isContinueScan;				
	引数名	In/Out	型	説明
引数	isContinueScan	In	BOOL	データを連続して読み取るかどうか YES : 連続して読み取る NO : 連続して読取らない
<p>■ 説明:</p> <p>連続読取モードを設定します。</p> <p>■ サンプルコード:</p> <pre>[asP52N setScanContinuous:YES]; </pre>				

### 2.2.34. getReaderInfo

- (BOOL)getReaderInfo;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■説明:</p> <p>AsReader の基本情報を取得します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedReaderInfo:」(<a href="#">2.3.19</a>を参照) デリゲートがコールバックされます。</p> <p>■サンプルコード:</p> <pre> BOOL status = [asP52N getReaderInfo]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre>				

### 2.2.35. getReaderAntennaPower

- (BOOL)getReaderAntennaPower;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■説明:</p> <p>アンテナパワーを取得します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedOutputPowerLevel:executionStatus:」(<a href="#">2.3.14</a>を参照) 、 「AsP52N:receivedReaderAntenna1Power:antenna2Power:antenna3Power:antenna4Power:antenna5Power:antenna6Power:antenna7Power:antenna8Power:」(<a href="#">2.3.20</a>を参照) デリゲートがコールバックされます。</p>				

## ■ サンプルコード:

```

BOOL status = [asP52N getReaderAntennaPower];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

## 2.2.36. setReaderAntenna1Power

```

- (BOOL)setReaderAntenna1Power:(int)antenna1Power
      antenna2Power:(int)antenna2Power
      antenna3Power:(int)antenna3Power
      antenna4Power:(int)antenna4Power
      antenna5Power:(int)antenna5Power
      antenna6Power:(int)antenna6Power
      antenna7Power:(int)antenna7Power
      antenna8Power:(int)antenna8Power;

```

	引数名	In/Out	型	説明
引数	antenna1Power	In	int	アンテナ 1 の電波出力 (単位 : dBm) 範囲 : 0~36
引数	antenna2Power	In	int	アンテナ 2 の電波出力 (単位 : dBm) 範囲 : 0~36
引数	antenna3Power	In	int	アンテナ 3 の電波出力 (単位 : dBm) 範囲 : 0~36
引数	antenna4Power	In	int	アンテナ 4 の電波出力 (単位 : dBm) 範囲 : 0~36
引数	antenna5Power	In	int	アンテナ 5 の電波出力 (単位 : dBm) 範囲 : 0~36
引数	antenna6Power	In	int	アンテナ 6 の電波出力 (単位 : dBm) 範囲 : 0~36
引数	antenna7Power	In	int	アンテナ 7 の電波出力 (単位 : dBm) 範囲 : 0~36
引数	antenna8Power	In	int	アンテナ 8 の電波出力 (単位 : dBm)

				範囲：0～36
戻り値	-	Out	BOOL	コマンドの実行結果 YES：実行成功 NO：実行失敗
<p><b>■説明:</b> アンテナの電波出力を設定します。 このメソッドを呼び出すと、「AsP52N:didSetOutputPowerLevelExecutionStatus」(<a href="#">2.3.13</a>を参照)、「AsP52N:receivedSetAntennaPowerStatus」(<a href="#">2.3.21</a>を参照)デリゲートがコールバックされます。</p> <p><b>■サンプルコード:</b>  <pre> BOOL status = [asP52N setReaderAntenna1Power:36 antenna2Power:36 antenna3Power:36 antenna4Power:36 antenna5Power:36 antenna6Power:36 antenna7Power:36 antenna8Power:36]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre> </p>				

### 2.2.37. getReaderBasebandInformation

<b>- (BOOL)getReaderBasebandInformation;</b>				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES：実行成功 NO：実行失敗
<p><b>■説明:</b> AsReader 使用中のベースバンドパラメータを取得します。 このメソッドを呼び出すと、 「AsP52N:receivedReaderBasebandRate:QValue:session:inventoryFlag:」(<a href="#">2.3.22</a>を参照)デリゲートがコールバックされます。</p> <p><b>■サンプルコード:</b>  <pre> BOOL status = [asP52N getReaderBasebandInformation]; </pre> </p>				

```

if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

### 2.2.38. setReaderBasebandRate

- (BOOL)setReaderBasebandRate:(AsP52NBasebandRateStatus)basebandRate QValue:(int)qValue session:(AsP52NSessionType)session inventoryFlag:(AsP52NInventoryFlag)inventoryFlag;				
	引数名	In/Out	型	説明
引数	basebandRate	In	AsP52NBasebandRateStatus	EPC のポーレート AsP52NBasebandRateStatus 列挙型 ( <a href="#">2.4.6</a> を参照)
引数	qValue	In	int	Q 値 範囲：0～15
引数	session	In	AsP52NSessionType	Session 値 AsP52NSessionType 列挙型 ( <a href="#">2.4.2</a> を参照)
引数	inventoryFlag	In	AsP52NInventoryFlag	Session フラグ AsP52NInventoryFlag 列挙型 ( <a href="#">2.4.7</a> を参照)
戻り値	-	Out	BOOL	コマンドの実行結果 YES：実行成功 NO：実行失敗
<p>■説明:</p> <p>AsReader 使用するベースバンドのパラメータを設定します。 このメソッドを呼び出すと、「AsP52N:receivedSetBasebandStatus:」(<a href="#">2.3.23</a>を参照) デリゲートがコールバックされます。</p> <p>■サンプルコード:</p> <pre> BOOL status = [asP52N setReaderBasebandRate:AsP52NBasebandRateStatus400KHz </pre>				

```

QValue:0 session: AsP52NSessionType_S0 inventoryFlag:AsP52NInventoryFlag_AB];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

### 2.2.39. getReaderAutoDormancy

- (BOOL)getReaderAutoDormancy;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>自動スリープ時間を取得します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedReaderAutoDormancy:dormancyTime:」(<a href="#">2.3.24</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre> BOOL status = [asP52N getReaderAutoDormancy]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre>				

### 2.2.40. setReaderAutoDormancy

- (BOOL)setReaderAutoDormancy:(BOOL)autoDormancy time:(int)time;				
	引数名	In/Out	型	説明
引数	autoDormancy	In	BOOL	自動スリープを使用するかどうか YES : 使用する NO : 使用しない

引数	time	In	int	自動スリープ時間（単位：10ms） 範囲：0～65535
戻り値	-	Out	BOOL	コマンドの実行結果 YES：実行成功 NO：実行失敗

■ 説明:  
自動スリープを設定します。  
このメソッドを呼び出すと、「AsP52N:receivedSetAutoDormancyStatus」（[2.3.25](#)を参照）デリゲートがコールバックされます。

■ サンプルコード:  

```

BOOL status = [asP52N setReaderAutoDormancy:YES time:10];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

### 2.2.41. getReaderRFIDFrequency

<b>- (BOOL)getReaderRFIDFrequency;</b>				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES：実行成功 NO：実行失敗

■ 説明:  
RFID 周波数を取得します。  
このメソッドを呼び出すと、「AsP52N:receivedReaderRFIDFrequency」（[2.3.26](#)を参照）デリゲートがコールバックされます。

■ サンプルコード:  

```

BOOL status = [asP52N getReaderRFIDFrequency];
if (status) {
//コマンド実行に成功。
} else {

```

```
//コマンド実行に失敗。
}
```

## 2.2.42. setReaderRFIDFrequency

- (BOOL)setReaderRFIDFrequency:(AsP52NRFIDFrequencyStatus)frequency;				
	引数名	In/Out	型	説明
引数	frequency	In	AsP52NRFIDFrequencyStatus	RF 周波数 AsP52NRFIDFrequencyStatus 列挙型 ( <a href="#">2.4.8</a> を参照)
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗

**■説明:**  
 RFID 周波数を設定します。  
 このメソッドを呼び出すと、「AsP52N:receivedSetRFIDFrequencyStatus」([2.3.27](#)を参照) デリゲートがコールバックされます。

**■サンプルコード:**  

```
BOOL status = [asP52N setReaderRFIDFrequency:AsP52NRFIDFrequencyStatusJP];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}
}
```

## 2.2.43. getReaderRFIDFrequencyList

- (NSArray *)getReaderRFIDFrequencyList:(AsP52NRFIDFrequencyStatus)frequency;				
	引数名	In/Out	型	説明
引数	frequency	In	AsP52NRFIDFrequencyStatus	RF 周波数 AsP52NRFIDFrequencyStatus 列挙型 ( <a href="#">2.4.8</a> を参照)
戻り値	-	Out	BOOL	RF の周波数リスト

■ **説明:**

RF 周波数リストを取得します。

このメソッドを呼び出すと、「AsP52N:receivedSetRFIDFrequencyStatus」 ([2.3.27](#) を参照) デリゲートがコールバックされます。

■ **サンプルコード:**

```
NSArray *dataArray = [asP52N getReaderRFIDFrequencyList:frequency];
```

## 2.2.44. getReaderTagLog

### - (BOOL)getReaderTagLog;

	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗

■ **説明:**

設定した読取結果をフィルターするパラメータを取得します。

このメソッドを呼び出すと、「AsP52N:receivedReaderTagLogFilterTime:RSSI : 」 ([2.3.28](#) を参照) デリゲートがコールバックされます。

■ **サンプルコード:**

```
BOOL status = [asP52N getReaderTagLog];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}
```

## 2.2.45. setReaderTagLogFilterTime

### - (BOOL)setReaderTagLogFilterTime:(int)filtertime RSSI:(int)rssi;

	引数名	In/Out	型	説明
引数	filtertime	In	int	同一タグの読取間隔 (単位 : 10ms)

				範囲 : 0~65535
引数	rsssi	In	int	RSSI 閾値
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>タグを読取後のフィルター条件を取得します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedSetTagLogStatus」 (<a href="#">2.3.29</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre> BOOL status = [asP52N setReaderTagLogFilterTime:10 RSSI:5]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre>				

## 2.2.46. getReaderFrequency

- (BOOL)getReaderFrequency;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>周波数を取得します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedReaderFrequencyAutomatic:frequencyList : 」 (<a href="#">2.3.30</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre> BOOL status = [asP52N getReaderFrequency]; if (status) { //コマンド実行に成功。 } else { </pre>				

```
//コマンド実行に失敗。
}
```

## 2.2.47. setReaderFrequencyAutomatic

- (BOOL)setReaderFrequencyAutomatic:(BOOL)frequencyAutomatic frequencyList:(NSString *)frequencyList;				
	引数名	In/Out	型	説明
引数	frequencyAutomatic	In	BOOL	自動的に周波数を選択するかどうか YES : 使用する NO : 使用しない
引数	frequencyList	In	NSString	周波数リスト 周波数範囲は付録 <a href="#">3.1</a> を参照
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■説明:</p> <p>AsReader の稼動周波数を設定します。 このメソッドを呼び出すと、「AsP52N:receivedSetFrequencyStatus」(<a href="#">2.3.31</a>を参照) デリゲートがコールバックされます。</p> <p>■サンプルコード:</p> <pre>BOOL status = [asP52N setReaderFrequencyAutomatic:YES frequencyList:@"13,18,19"]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } }</pre>				

## 2.2.48. getReaderEthernetMacAddress

- (BOOL)getReaderEthernetMacAddress;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果

				YES : 実行成功 NO : 実行失敗
<p><b>■説明:</b> イーサネットの MAC アドレスを取得します。 このメソッドを呼び出すと、「AsP52N:receivedReaderEthernetMACAddress:」(<a href="#">2.3.32</a>を参照) デリゲートがコールバックされます。</p> <p><b>■サンプルコード:</b>  <pre> BOOL status = [asP52N getReaderEthernetMacAddress]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre> </p>				

## 2.2.49. getReaderTime

- (BOOL)getReaderTime;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p><b>■説明:</b> リーダーのシステム時間を取得します。 このメソッドを呼び出すと、「AsP52N:receivedReaderTime:」(<a href="#">2.3.33</a>を参照) デリゲートがコールバックされます。</p> <p><b>■サンプルコード:</b>  <pre> BOOL status = [asP52N getReaderTime]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre> </p>				

## 2.2.50. setReaderTime

- (BOOL)setReaderTime:(long)readerTime;				
	引数名	In/Out	型	説明
引数	readerTime	In	long	リーダーのシステム時間
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>リーダーのシステム時間を設定します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedSetReaderTimeStatus:」(<a href="#">2.3.34</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>//システム時間を 2023-02-09 16:17:30 に設定します。 BOOL status = [asP52N setReaderTime: 1675930650000]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre>				

## 2.2.51. getReaderEthernetIP

- (BOOL)getReaderEthernetIP;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>イーサネットの IP アドレスを取得します。</p> <p>このメソッドを呼び出すと、</p> <p>「AsP52N:receivedReaderEthernetIP:subnetMask:defaultGateway:preferredDNS:alternateDNS:automaticIP:」(<a href="#">2.3.35</a>を参照) デリゲートがコールバックされます。</p>				

■ サンプルコード:

```

BOOL status = [asP52N getReaderEthernetIP];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

## 2.2.52. setReaderEthernetIP

- (BOOL)setReaderEthernetIP:(NSString *)ip subnetMask:(NSString *)subnetMask defaultGateway:(NSString *)defaultGateway preferredDNS:(NSString *)preferredDNS alternateDNS:(NSString *)alternateDNS automaticIP:(BOOL)automaticIP;				
	引数名	In/Out	型	説明
引数	ip	In	NSString	IP アドレス
引数	subnetMask	In	NSString	サブネットマスク
引数	defaultGateway	In	NSString	デフォルトゲートウェイ
引数	preferredDNS	In	NSString	プライマリ DNS
引数	alternateDNS	In	NSString	バックアップ DNS
引数	automaticIP	In	BOOL	動的 IP/固定 IP YES : 動的 IP NO : 固定 IP
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>イーサネットの IP アドレスを設定します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedSetEthernetIPStatus:」 (<a href="#">2.3.36</a> を参照) デリゲートがコールバックされます。</p> <p>注 : このメソッドを呼び出すと接続が切断され、再接続が必要です。</p>				

■ サンプルコード:

```

BOOL status = [asP52N setReaderEthernetIP:@"192.168.1.10"
subnetMask:@"255.255.255.255" defaultGateway:@"255.255.255.255"
preferredDNS:@"255.255.255.255" alternateDNS:@"255.255.255.255"
automaticIP:YES];
if (status) {
//コマンド実行に成功。
} else {
//コマンド実行に失敗。
}

```

### 2.2.53. factoryReset

- (BOOL)factoryReset;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>出荷時設定に戻します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedFactoryResetStatus:」(<a href="#">2.3.37</a>を参照) デリゲートがコールバックされます。</p> <p>注 : このメソッドを呼び出すと接続が切断され、再接続が必要です。</p> <p>■ サンプルコード:</p> <pre> BOOL status = [asP52N factoryReset]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre>				

## 2.2.54. restartDevice

- (BOOL)restartDevice;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>AsReader を再起動します。</p> <p>注 : このメソッドを呼び出すと接続が切断され、再接続が必要です。</p> <p>■ サンプルコード: (注:asP52N は AsP52N クラスのインスタンスオブジェクトです)</p> <pre>BOOL status = [asP52N restartDevice]; if (status) { //コマンド実行に成功。 } else { //コマンド実行に失敗。 } </pre>				

## 2.2.55. getTemperature

- (BOOL) getTemperature;				
	引数名	In/Out	型	説明
戻り値	-	Out	BOOL	コマンドの実行結果 YES : 実行成功 NO : 実行失敗
<p>■ 説明:</p> <p>AsReader の温度を取得します。</p> <p>このメソッドを呼び出すと、「AsP52N:receivedGetTemperatureAlarmTemperature」(<a href="#">2.3.47</a>を参照) デリゲートがコールバックされます。</p> <p>■ サンプルコード:</p> <pre>BOOL status = [asP52N restartDevice]; if (status) { //コマンド実行に成功。 } </pre>				

```

} else {
//コマンド実行に失敗。
}

```

## 2.3. デリゲート

### 2.3.1. AsP52N:searchingIP:isFinish:

```
- (void)AsP52N:(AsP52N *)AsP52N searchingIP:(NSString *)IP isFinish:(BOOL)isFinish;
```

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	IP	Out	NSString	IP アドレス
引数	isFinish	Out	BOOL	検索状態 YES : 検索完了 NO : 引き続き検索

#### ■ 説明:

IP アドレスを受信します。

startSearchDevice ([2.2.3](#)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:

```

NSMutableArray *arrayIPs = [[NSMutableArray alloc] init];
- (void)AsP52N:(AsP52N *)AsP52N searchingIP:(NSString *)IP isFinish:(BOOL)isFinish{
    if(isFinish == YES) {
        //検索完了
    } else {
    }
}

```

### 2.3.2. AsP52N:changeNetworkState:error:

```
- (void)AsP52N:(AsP52N *)AsP52N
changeNetworkState:(AsP52NetworkState)state error:(NSError *)error;
```

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	state	Out	AsP52NetworkState	AsReader の接続状態 AsP52NetworkState 列挙型 ( <a href="#">2.4.1</a> を参照)
引数	error	Out	NSError	エラーが発生する時、内容を返す

■ 説明:  
接続状態を受信します。  
接続状態が変化された時にコールバックされます。

■ サンプルコード:

```
- (void)AsP52N:(AsP52N *)AsP52N changeNetworkState:(AsP52NetworkState)state
error:(NSError *)error{
    if (state == AsP52NetworkStateConnected) {
        //接続に成功
    } else if (state == AsP52NetworkStateDisconnected) {
        //接続を切断
    } else if (state == AsP52NetworkStateUnknown) {
        //未知のネットワーク
    } else {
        //error その他のエラー
    }
}
```

### 2.3.3. AsP52N:didStartInventoryExecutionStatus:

- (void)AsP52N:(AsP52N *)AsP52N didStartInventoryExecutionStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

■ 説明:

インベントリ開始の実行結果を受信します。

startInventoryWithEPC ([2.2.7](#)を参照)、readTagWithAccessPassword ([2.2.9](#)を参照)、readDataWithAntenna1 ([2.2.24](#)を参照)、startInventoryWithAntenna1 ([2.2.28](#)を参照)、startInventoryTIDWithAntenna1 ([2.2.29](#)を参照)、startInventoryMoistureWithAntenna1 ([2.2.30](#)を参照)、startInventoryMagnusS3WithAntenna1 ([2.2.31](#)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)AsP52N:(AsP52N *)AsP52N
didStartInventoryExecutionStatus:(AsP52NExecutionStatus)statusCode{
    if (statusCode == AsP52NExecutionStatusSuccess) {
//読取開始に成功
    } else {
        //読取開始に失敗
        getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)。
        NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
    }
}
```

### 2.3.4. AsP52N:didStopInventoryExecutionStatus:

- (void)AsP52N:(AsP52N \*)AsP52N  
didStopInventoryExecutionStatus:(AsP52NExecutionStatus)statusCode;

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

#### ■ 説明:

インベントリ停止の実行結果を受信します。

stopInventory ([2.2.8](#)を参照)、stopReadData ([2.2.32](#)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)AsP52N:(AsP52N *)AsP52N
didStopInventoryExecutionStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//読取停止に成功
} else {
//読取停止に失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
}

```

### 2.3.5. AsP52N:receivedInventoryDataWithRSSI:channel:phase:antenna:PC:EPC:TID:

```

- (void)AsP52N:(AsP52N *)AsP52N
receivedInventoryDataWithRSSI:(NSString *)RSSI
channel:(NSString *)channel
phase:(NSString *)phase
antenna:(NSString *)antenna
PC:(NSString *)PC
EPC:(NSString *)EPC
TID:(NSString *)TID;

```

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	RSSI	Out	NSString	タグの RSSI データ
引数	channel	Out	NSString	タグのチャンネル情報
引数	phase	Out	NSString	タグ位相情報
引数	antenna	Out	NSString	アンテナ情報
引数	PC	Out	NSString	タグの PC データ
引数	EPC	Out	NSString	タグの EPC データ
引数	TID	Out	NSString	タグの TID データ

■説明:  
インベントリデータを受信します。

startInventoryWithEPC ([2.2.7](#)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)AsP52N:(AsP52N *)AsP52N
receivedInventoryDataWithRSSI:(NSString *)RSSI
    channel:(NSString *)channel
    phase:(NSString *)phase
    antenna:(NSString *)antenna
    PC:(NSString *)PC
    EPC:(NSString *)EPC
    TID:(NSString *)TID {
// RSSI   タグの RSSI データ。
startInventoryWithEPC メソッドの引数 RSSI は NO の場合、戻り値は「@'''」になります。
// channel タグのチャンネル情報。
startInventoryWithEPC メソッドの引数 channel は NO の場合、戻り値は「@'''」になります。
// phase   タグのタグ位相情報。
startInventoryWithEPC メソッドの引数 phase は NO の場合、戻り値は「@'''」になります。
// antenna タグのアンテナ情報。
startInventoryWithEPC メソッドの引数 antenna は NO の場合、戻り値は「@'''」になります。
// PC      タグの PC データ。
startInventoryWithEPC メソッドの引数 PC は NO の場合、戻り値は「@'''」になります。
// EPC     タグの EPC データ。
startInventoryWithEPC メソッドの引数 EPC は NO の場合、戻り値は「@'''」になります。
// TID     タグの TID データ。
startInventoryWithEPC メソッドの引数 TID は NO の場合、戻り値は「@'''」になります。
}
```

### 2.3.6. AsP52N:didKillTagExecutionStatus :

- (void)AsP52N:(AsP52N \*)AsP52N

didKillTagExecutionStatus:(AsP52NExecutionStatus)statusCode;

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

■ **説明:**

タグキルの実行結果を受信します。

killTagWithKillPassword ([2.2.12](#)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)AsP52N:(AsP52N *)AsP52N
didKillTagExecutionStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//タグキルに成功
} else {
//タグキルに失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

### 2.3.7. AsP52N:didWriteTagExecutionStatus:

- (void)AsP52N:(AsP52N \*)AsP52N  
didWriteTagExecutionStatus:(AsP52NExecutionStatus)statusCode;

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

■ **説明:**

タグ書き込みの実行結果を受信します。

writeTagWithAccessPassword ([2.2.10](#)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)AsP52N:(AsP52N *)AsP52N
didWriteTagExecutionStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//タグの書き込みに成功
} else {
```

```

//タグ書き込みに失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}

```

### 2.3.8. AsP52N:didLockTagExecutionStatus:

- (void)AsP52N:(AsP52N \*)AsP52N

didLockTagExecutionStatus:(AsP52NExecutionStatus)statusCode;

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 (2.4.9を参照)

■説明:

タグロックの実行結果を受信します。

lockTagWithAccessPassword (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)AsP52N:(AsP52N \*)AsP52N

didLockTagExecutionStatus:(AsP52NExecutionStatus)statusCode {

if (statusCode == AsP52NExecutionStatusSuccess) {

//タグロックに成功

} else {

    //タグロックに失敗

getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)

NSString \*errorString = [asP52N getMessageWithResultCode:statusCode];

// errorString 失敗原因

}

}

### 2.3.9. AsP52N:didReadTagExecutionStatus:

- (void)AsP52N:(AsP52N *)AsP52N didReadTagExecutionStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト。
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
<p>■ 説明:</p> <p>タグ読取の実行結果を受信します。 readTagWithAccessPassword (<a href="#">2.2.9</a>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>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N didReadTagExecutionStatus:(AsP52NExecutionStatus)statusCode { if (statusCode == AsP52NExecutionStatusSuccess) { //タグの読取に成功 } else { //タグの読取に失敗 getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照) NSString *errorString = [asP52N getMessageWithResultCode:statusCode]; // errorString 失敗原因 } }</pre>				

### 2.3.10. AsP52N:receivedTagOperationData:

- (void)AsP52N:(AsP52N *)AsP52N receivedTagOperationData:(NSData *)tagOperationData;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト。
引数	tagOperationData	Out	NSData	読み取ったデータ
<p>■ 説明:</p> <p>読み取ったデータを受信します。</p>				

readTagWithAccessPassword ([2.2.9](#)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)AsP52N:(AsP52N *)AsP52N
receivedTagOperationData:(NSData *)tagOperationData {
// tagOperationData 読み取ったデータ。
}
```

### 2.3.11. AsP52N:didSetAntennaPortsStatusExecutionStatus:

- (void)AsP52N:(AsP52N \*)AsP52N  
didSetAntennaPortsStatusExecutionStatus:(AsP52NExecutionStatus)statusCode;

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

■ 説明:

アンテナ設定の実行結果を受信します。

setAntennaPortsStatusWithAntenna1 ([2.2.13](#)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)AsP52N:(AsP52N *)AsP52N
didSetAntennaPortsStatusExecutionStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//アンテナ設定に成功
} else {
//アンテナ設定に失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

### 2.3.12. AsP52N:receivedAntennaPortsStatusAntenna1:antenna2:antenna3:antenna4:antenna5:antenna6:antenna7:antenna8:executionStatus:

```

- (void)AsP52N:(AsP52N *)AsP52N
receivedAntennaPortsStatusAntenna1:(BOOL)antenna1IsOpen
antenna2:(BOOL)antenna2IsOpen
antenna3:(BOOL)antenna3IsOpen
antenna4:(BOOL)antenna4IsOpen
antenna5:(BOOL)antenna5IsOpen
antenna6:(BOOL)antenna6IsOpen
antenna7:(BOOL)antenna7IsOpen
antenna8:(BOOL)antenna8IsOpen
executionStatus:(AsP52NExecutionStatus)statusCode;

```

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	antenna1IsOpen	Out	BOOL	アンテナ 1 の状態 YES : オン NO : オフ
引数	antenna2IsOpen	Out	BOOL	アンテナ 2 の状態 YES : オン NO : オ
引数	antenna3IsOpen	Out	BOOL	アンテナ 3 の状態 YES : オン NO : オフ
引数	antenna4IsOpen	Out	BOOL	アンテナ 4 の状態 YES : オン NO : オフ
引数	antenna5IsOpen	Out	BOOL	アンテナ 5 の状態 YES : オン NO : オフ
引数	antenna6IsOpen	Out	BOOL	アンテナ 6 の状態 YES : オン NO : オフ
引数	antenna7IsOpen	Out	BOOL	アンテナ 7 の状態

				YES : オン NO : オフ
引数	antenna8IsOpen	Out	BOOL	アンテナ 8 の状態 YES : オン NO : オフ
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

■ 説明:

アンテナのオン・オフ状態を受信します。

getAntennaPortsStatus ([2.2.14](#)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)AsP52N:(AsP52N *)AsP52N
receivedAntennaPortsStatus:(BOOL)antenna1IsOpen
    antenna2:(BOOL)antenna2IsOpen
    antenna3:(BOOL)antenna3IsOpen
    antenna4:(BOOL)antenna4IsOpen
    antenna5:(BOOL)antenna5IsOpen
    antenna6:(BOOL)antenna6IsOpen
    antenna7:(BOOL)antenna7IsOpen
    antenna8:(BOOL)antenna8IsOpen
executionStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//アンテナ状態の受信に成功
} else {
//アンテナ状態の受信に失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

### 2.3.13. AsP52N:didSetOutputPowerLevelExecutionStatus:

```
- (void)AsP52N:(AsP52N *)AsP52N
didSetOutputPowerLevelExecutionStatus:(AsP52NExecutionStatus)statusCode;
```

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト。
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

■説明:

電波出力設定の実行結果を受信します。

setOutputPowerLevel ([2.2.15](#)を参照)、setReaderAntenna1Power ([2.2.36](#)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)AsP52N:(AsP52N *)AsP52N
didSetOutputPowerLevelExecutionStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//電波出力設定に成功
} else {
//電波出力設定に失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

### 2.3.14. AsP52N:receivedOutputPowerLevel:executionStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedOutputPowerLevel:(uint8_t)outputPowerLevel executionStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	outputPowerLevel	Out	uint8_t	電波出力 (単位: dBm) 範囲: 0~36
引数	statusCode	Out	AsP52NExecutio	実行結果

			nStatus	AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
<p>■説明:</p> <p>電波出力を受信します。</p> <p>getOutputPowerLevel (<a href="#">2.2.16</a>を参照)、getReaderAntennaPower (<a href="#">2.2.35</a>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>注：アンテナ 1 の電波出力になります。</p> <p>■サンプルコード:</p> <pre> - (void)AsP52N:(AsP52N *)AsP52N receivedOutputPowerLevel:(uint8_t)outputPowerLevel executionStatus:(AsP52NExecutionStatus)statusCode { if (statusCode == AsP52NExecutionStatusSuccess) { //電波出力の受信に成功。 } else { //電波出力の受信に失敗。 getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照) NSString *errorString = [asP52N getMessageWithResultCode:statusCode]; // errorString 失敗原因 } } </pre>				

### 2.3.15. AsP52N:didSetSessionExecutionStatus:

- (void)AsP52N:(AsP52N *)AsP52N didSetSessionExecutionStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
<p>■説明:</p> <p>Session 設定の実行結果を受信します。</p> <p>setSession (<a href="#">2.2.17</a>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>				

■ サンプルコード:

```
- (void)AsP52N:(AsP52N *)AsP52N
didSetSessionExecutionStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//Session 設定に成功
} else {
//Session 設定に失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

### 2.3.16. AsP52N:receivedSession:executionStatus:

```
- (void)AsP52N:(AsP52N *)AsP52N
receivedSession:(AsP52NSessionType)session
executionStatus:(AsP52NExecutionStatus)statusCode;
```

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	session	Out	AsP52NSessionType	AsP52NSessionType 列挙型 (2.4.2を参照)
引数	statusCode	Out	AsP52NExecutionStatus	AsP52NExecutionStatus 列挙型 (2.4.9を参照)

■ 説明:

Session の値を受信します。  
getSession (2.2.18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)AsP52N:(AsP52N *)AsP52N
receivedSession:(AsP52NSessionType)session
executionStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//Session 値の受信に成功
// session AsP52NSessionType 列挙型 (2.4.2を参照)
```

```

} else {
    // Session 値の受信に失敗
    getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
    NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
    // errorString 失敗原因
}
}

```

### 2.3.17. AsP52N:receivedFirmwareVersion:executionStatus:

```

- (void)AsP52N:(AsP52N *)AsP52N
receivedFirmwareVersion:(NSString *)firmwareVersion
executionStatus:(AsP52NExecutionStatus)statusCode;

```

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	firmwareVersion	Out	NSString	ファームウェアバージョン
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 (2.4.9を参照)

#### ■説明:

ファームウェアバージョンを受信します。

getFirmwareVersion (2.2.21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)AsP52N:(AsP52N *)AsP52N
receivedFirmwareVersion:(NSString *)firmwareVersion
executionStatus:(AsP52NExecutionStatus)statusCode {
    if (statusCode == AsP52NExecutionStatusSuccess) {
        //ファームウェアバージョン情報の受信に成功
        // firmwareVersion ファームウェアバージョン
    } else {
        //ファームウェアバージョン情報の受信に失敗
        getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
        NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
        // errorString 失敗原因
    }
}

```

```
}
}
```

### 2.3.18. AsP52N:didInventoryEndExecutionStatus:

- (void)AsP52N:(AsP52N *)AsP52N didInventoryEndExecutionStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
<p>■ 説明:</p> <p>読取停止の実行結果を受信します。</p> <p>startInventoryWithEPC (<a href="#">2.2.7</a>を参照)、readTagWithAccessPassword (<a href="#">2.2.9</a>を参照)、readDataWithAntenna1 (<a href="#">2.2.24</a>を参照)、startInventoryWithAntenna1 (<a href="#">2.2.28</a>を参照)、startInventoryTIDWithAntenna1 (<a href="#">2.2.29</a>を参照)、startInventoryMoistureWithAntenna1 (<a href="#">2.2.30</a>を参照)、startInventoryMagnusS3WithAntenna1 (<a href="#">2.2.31</a>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>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N didInventoryEndExecutionStatus:(AsP52NExecutionStatus)statusCode { if (statusCode == AsP52NExecutionStatusSuccess) { //読取停止に成功 } else { //読取停止に失敗 getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照) NSString *errorString = [asP52N getMessageWithResultCode:statusCode]; // errorString 失敗原因 } }</pre>				

### 2.3.19. AsP52N:receivedReaderInfo:

```
- (void)AsP52N:(AsP52N *)AsP52N
receivedReaderInfo:(NSDictionary *)readerInfo;
```

	引数名	In/Out	型	説明
引数	AsP52N	In	AsP52N	AsP52N オブジェクト
引数	readerInfo	Out	NSDictionary	AsReader 情報のディクショナリー ディクショナリー情報は以下通り Key : deviceSerialNumber Value : リーダーの製造番号 Key : appVersion Value : アプリケーションのバージョン Key : applicationCompilation Value : アプリケーションのコンパイル時間 Key : basebandCompilation Value : ベースバンドのコンパイル時間 Key : systemVersion Value : システムのバージョン Key : powerOnTime Value : リーダー起動時間

■ 説明:

ファームウェア情報を受信します。

getReaderInfo ([2.2.34](#)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)AsP52N:(AsP52N *)AsP52N
receivedReaderInfo:(NSDictionary *)readerInfo {
// readerInfo AsReader 情報を保存するディクショナリー
}
```

### 2.3.20. AsP52N:receivedReaderAntenna1Power:antenna2Power:antenna3Power:antenna4Power:antenna5Power:antenna6Power:antenna7Power:antenna8Power:

```
- (void)AsP52N:(AsP52N *)AsP52N
receivedReaderAntenna1Power:(int)antenna1Power
antenna2Power:(int)antenna2Power
antenna3Power:(int)antenna3Power
antenna4Power:(int)antenna4Power
```

<b>antenna5Power:(int)antenna5Power  antenna6Power:(int)antenna6Power  antenna7Power:(int)antenna7Power  antenna8Power:(int)antenna8Power;</b>				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	antenna1Power	Out	int	アンテナ 1 の電波出力 (単位: dBm) 範囲: 0~36
引数	antenna2Power	Out	int	アンテナ 2 の電波出力 (単位: dBm) 範囲: 0~36
引数	antenna3Power	Out	int	アンテナ 3 の電波出力 (単位: dBm) 範囲: 0~36
引数	antenna4Power	Out	int	アンテナ 4 の電波出力 (単位: dBm) 範囲: 0~36
引数	antenna5Power	Out	int	アンテナ 5 の電波出力 (単位: dBm) 範囲: 0~36
引数	antenna6Power	Out	int	アンテナ 6 の電波出力 (単位: dBm) 範囲: 0~36
引数	antenna7Power	Out	int	アンテナ 7 の電波出力 (単位: dBm) 範囲: 0~36
引数	antenna8Power	Out	int	アンテナ 8 の電波出力 (単位: dBm) 範囲: 0~36
<p>■ 説明:</p> <p>アンテナの電波出力を受信します。  getOutputPowerLevel (<a href="#">2.2.16</a>を参照) 、getReaderAntennaPower (<a href="#">2.2.35</a>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>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedReaderAntenna1Power:(int)antenna1Power antenna2Power:(int)antenna2Power antenna3Power:(int)antenna3Power antenna4Power:(int)antenna4Power</pre>				

```

antenna5Power:(int)antenna5Power
antenna6Power:(int)antenna6Power
antenna7Power:(int)antenna7Power
antenna8Power:(int)antenna8Power {
// antenna1Power アンテナ 1 の電波出力。
// antenna2Power アンテナ 2 の電波出力。
// antenna3Power アンテナ 3 の電波出力。
// antenna4Power アンテナ 4 の電波出力。
// antenna5Power アンテナ 5 の電波出力。
// antenna6Power アンテナ 6 の電波出力。
// antenna7Power アンテナ 7 の電波出力。
// antenna8Power アンテナ 8 の電波出力。
}

```

### 2.3.21. AsP52N:receivedSetAntennaPowerStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedSetAntennaPowerStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
<p>■説明:</p> <p>アンテナパワー設定の実行結果を受信します。 setOutputPowerLevel (<a href="#">2.2.15</a>を参照)、setReaderAntenna1Power (<a href="#">2.2.36</a>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>■サンプルコード:</p> <pre> - (void)AsP52N:(AsP52N *)AsP52N receivedSetAntennaPowerStatus:(AsP52NExecutionStatus)statusCode { if (statusCode == AsP52NExecutionStatusSuccess) { //アンテナパワー設定に成功 } else { //アンテナパワー設定に失敗 </pre>				

```

getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}

```

### 2.3.22. AsP52N:receivedReaderBasebandRate:QValue:session:inventoryFlag:

- (void)AsP52N:(AsP52N *)AsP52N receivedReaderBasebandRate:(AsP52NBasebandRateStatus)basebandRate QValue:(int)qValue session:(AsP52NSessionType)session inventoryFlag:(AsP52NInventoryFlag)inventoryFlag;				
	引数名	In/ Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	basebandRate	Out	AsP52NBasebandRateStatus	EPC のポーレート AsP52NBasebandRateStatus 列挙型 (2.4.6を参照)
引数	qValue	Out	int	Q 値 範囲 : 0~15
引数	session	Out	AsP52NSessionType	Session 値 AsP52NSessionType 列挙型 (2.4.2を参照)
引数	inventoryFlag	Out	AsP52NInventoryFlag	Session フラグ AsP52NInventoryFlag 列挙型 (2.4.7を参照)
<p>■ 説明:</p> <p>ベースバンド情報を受信します。 getSession (2.2.18を参照)、getReaderBasebandInformation (2.2.37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>■ サンプルコード:</p> <p>- (void)AsP52N:(AsP52N *)AsP52N</p>				

```

receivedReaderBasebandRate:(AsP52NBasebandRateStatus)basebandRate
    QValue:(int)qValue
    session:(AsP52NSessionType)session
    inventoryFlag:(AsP52NInventoryFlag)inventoryFlag {
// basebandRate AsP52NBasebandRateStatus 列挙型 (2.4.6を参照)。
// qValue      Q値。
// session      AsP52NSessionType 列挙型 (2.4.2を参照)。
// inventoryFlag AsP52NInventoryFlag 列挙型 (2.4.7を参照)。
}

```

### 2.3.23. AsP52N:receivedSetBasebandStatus:

- (void)AsP52N:(AsP52N \*)AsP52N

receivedSetBasebandStatus:(AsP52NExecutionStatus)statusCode;

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 (2.4.9を参照)

#### ■説明:

ベースバンドパラメータ設定の実行結果を受信します。

setSession (2.2.17参照)、setReaderBasebandRate (2.2.38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)AsP52N:(AsP52N *)AsP52N
```

```
receivedSetBasebandStatus:(AsP52NExecutionStatus)statusCode {
```

```
if (statusCode == AsP52NExecutionStatusSuccess) {
```

```
//ベースバンド情報の設定に成功
```

```
} else {
```

```
    //ベースバンド情報の設定に失敗
```

```
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
```

```
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
```

```
// errorString 失敗原因
```

```
}
```

```
}
```

### 2.3.24. AsP52N:receivedReaderAutoDormancy:dormancyTime:

- (void)AsP52N:(AsP52N *)AsP52N receivedReaderAutoDormancy:(BOOL)autoDormancy dormancyTime:(int)dormancyTime;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	autoDormancy	Out	BOOL	自動スリープを使用するかどうか YES : 使用する NO : 使用しない
引数	dormancyTime	Out	int	自動スリープ時間 (単位 : 10ms) 範囲 : 0~65535。
<p>■ 説明:</p> <p>自動スリープのパラメータを受信します。 getReaderAutoDormancy (<a href="#">2.2.39</a>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>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedReaderAutoDormancy:(BOOL)autoDormancy dormancyTime:(int)dormancyTime { // autoDormancy 自動スリープを使用するかどうか。 // dormancyTime 自動スリープ時間。 }</pre>				

### 2.3.25. AsP52N:receivedSetAutoDormancyStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedSetAutoDormancyStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
<p>■ 説明:</p>				

自動スリープ設定の実行結果を受信します。

setReaderAutoDormancy ([2.2.40](#)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)AsP52N:(AsP52N *)AsP52N
receivedSetAutoDormancyStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//自動スリープ設定に成功
} else {
//自動スリープ設定に失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

### 2.3.26. AsP52N:receivedReaderRFIDFrequency:

- (void)AsP52N:(AsP52N \*)AsP52N

receivedReaderRFIDFrequency:(AsP52NRFIDFrequencyStatus)frequency;

	引数名	In/Output	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	frequency	Out	AsP52NRFIDFrequencyStatus	RF 周波数 AsP52NRFIDFrequencyStatus 列挙型 ( <a href="#">2.4.8</a> を参照)。

■ 説明:

RFID 周波数を受信します。

getReaderRFIDFrequency ([2.2.41](#)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)AsP52N:(AsP52N *)AsP52N
receivedReaderRFIDFrequency:(AsP52NRFIDFrequencyStatus)frequency {
// frequency AsP52NRFIDFrequencyStatus 列挙型 (2.4.8を参照) 。
}
```

### 2.3.27. AsP52N:receivedSetRFIDFrequencyStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedSetRFIDFrequencyStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
<p>■説明:</p> <p>RFID 周波数設定の実行結果を受信します。 setReaderRFIDFrequency (<a href="#">2.2.42</a>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>■サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedSetRFIDFrequencyStatus:(AsP52NExecutionStatus)statusCode { if (statusCode == AsP52NExecutionStatusSuccess) { // RFID 周波数設定に成功 } else { // RFID 周波数設定に失敗 getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(<a href="#">2.2.22</a>を参照)。 NSString *errorString = [asP52N getMessageWithResultCode:statusCode]; // errorString 失敗原因 } }</pre>				

### 2.3.28. AsP52N:receivedReaderTagLogFilterTime:RSSI :

- (void)AsP52N:(AsP52N *)AsP52N receivedReaderTagLogFilterTime:(int)filtertime RSSI:(int)rssi;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト。

引数	filtertime	Out	int	同一タグに対するフィルタ時間（単位： 10ms） 範囲：0～65535
引数	RSSI	Out	int	RSSI 閾値

■ 説明:  
読み取り結果のフィルター条件を受信します。  
getReaderTagLog ([2.2.43](#) 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)AsP52N:(AsP52N \*)AsP52N  
receivedReaderTagLogFilterTime:(int)filtertime  
RSSI:(int)rssi {  
// filtertime 同一タグに対するフィルタ時間。  
// RSSI RSSI 閾値。  
}

### 2.3.29. AsP52N:receivedSetTagLogStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedSetTagLogStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

■ 説明:  
読み取り結果のフィルター条件設定の実行結果を受信します。  
setReaderTagLogFilterTime ([2.2.44](#) 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)AsP52N:(AsP52N \*)AsP52N  
receivedSetTagLogStatus:(AsP52NExecutionStatus)statusCode {  
if (statusCode == AsP52NExecutionStatusSuccess) {  
//タグログ設定に成功

```

} else {
    //タグログ設定に失敗
    getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
    NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
    // errorString 失敗原因
}
}

```

### 2.3.30. AsP52N:receivedReaderFrequencyAutomatic:frequencyList

:

- (void)AsP52N:(AsP52N \*)AsP52N  
receivedReaderFrequencyAutomatic:(BOOL)frequencyAutomatic  
frequencyList:(NSString \*)frequencyList;

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	frequencyAutomatic	Out	BOOL	自動的に周波数を選択するかどうか。 YES : 自動選択 NO : 自動選択しない
引数	frequencyList	Out	NSString	周波数リスト 周波数範囲は付録 <a href="#">3.1</a> を参照

■説明:

周波数を受信します。

getReaderFrequency (2.2.45 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)AsP52N:(AsP52N *)AsP52N
receivedReaderFrequencyAutomatic:(BOOL)frequencyAutomatic
frequencyList:(NSString *)frequencyList {
    // frequencyAutomatic 自動的に周波数を選択するかどうか
    // frequencyList      周波数リスト
}

```

### 2.3.31. AsP52N:receivedSetFrequencyStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedSetFrequencyStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
<p>■説明:</p> <p>周波数設定の実行結果を受信します。 setReaderFrequencyAutomatic (<a href="#">2.2.46</a>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>■サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedSetFrequencyStatus:(AsP52NExecutionStatus)statusCode { if (statusCode == AsP52NExecutionStatusSuccess) { //周波数設定に成功 } else { //周波数設定に失敗 getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照) NSString *errorString = [asP52N getMessageWithResultCode:statusCode]; // errorString 失敗原因 } }</pre>				

### 2.3.32. AsP52N:receivedReaderEthernetMACAddress:

- (void)AsP52N:(AsP52N *)AsP52N receivedReaderEthernetMACAddress:(NSString *)ethernetMACAddress;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	ethernetMACAddress	Out	NSString	イーサネット MAC アドレス

	SS			
<p>■ 説明:</p> <p>イーサネット MAC アドレスを受信します。  getReaderEthernetMacAddress (2.2.47 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>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedReaderEthernetMACAddress:(NSString *)ethernetMACAddress { // ethernetMACAddress イーサネット MAC アドレス }</pre>				

### 2.3.33. AsP52N:receivedReaderTime:

<p>- (void)AsP52N:(AsP52N *)AsP52N  receivedReaderTime:(NSString *)readerTime;</p>				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	readerTime	Out	NSString	リーダーのシステム時間
<p>■ 説明:</p> <p>AsReader のシステム時間を受信します。  getReaderTime (<a href="#">2.2.48</a> 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>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedReaderTime:(NSString *)readerTime { // readerTime AsReader のシステム時間 }</pre>				

### 2.3.34. AsP52N:receivedSetReaderTimeStatus:

<p>- (void)AsP52N:(AsP52N *)AsP52N  receivedSetReaderTimeStatus:(AsP52NExecutionStatus)statusCode;</p>				
	引数名	In/Out	型	説明

引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)。
<p>■ 説明:</p> <p>AsReader システム時間設定の実行結果を受信します。 setReaderTime (<a href="#">2.2.49</a>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>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedSetReaderTimeStatus:(AsP52NExecutionStatus)statusCode { if (statusCode == AsP52NExecutionStatusSuccess) { //システム時間設定に成功 } else { //システム時間設定に失敗 getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照) NSString *errorString = [asP52N getMessageWithResultCode:statusCode]; // errorString 失敗原因 } }</pre>				

### 2.3.35. AsP52N:receivedReaderEthernetIP:subnetMask:defaultGateway:preferredDNS:alternateDNS:automaticIP:

<pre>- (void)AsP52N:(AsP52N *)AsP52N receivedReaderEthernetIP:(NSString *)ip subnetMask:(NSString *)subnetMask defaultGateway:(NSString *)defaultGateway preferredDNS:(NSString *)preferredDNS alternateDNS:(NSString *)alternateDNS automaticIP:(BOOL)automaticIP;</pre>				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	ip	Out	NSString	IP アドレス

引数	subnetMask	Out	NSString	サブネットマスク
引数	defaultGateway	Out	NSString	デフォルトゲートウェイ
引数	preferredDNS	Out	NSString	プライマリ DNS
引数	alternateDNS	Out	NSString	バックアップ DNS
引数	automaticIP	Out	BOOL	動的 IP/固定 IP YES : 動的 IP NO : 固定 IP

■ 説明:

イーサネット IP アドレスパラメータを受信します。

getReaderEthernetIP ([2.2.50](#)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)AsP52N:(AsP52N *)AsP52N
receivedReaderEthernetIP:(NSString *)ip
    subnetMask:(NSString *)subnetMask
    defaultGateway:(NSString *)defaultGateway
    preferredDNS:(NSString *)preferredDNS
    alternateDNS:(NSString *)alternateDNS
    automaticIP:(BOOL)automaticIP {
// 設定した IP アドレスパラメータ
}
```

### 2.3.36. AsP52N:receivedSetEthernetIPStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedSetEthernetIPStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
■ 説明:				

イーサネット IP アドレスパラメータ設定の実行結果を受信します。

setReaderEthernetIP ([2.2.51](#) 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)AsP52N:(AsP52N *)AsP52N
receivedSetEthernetIPStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//IP アドレスパラメータ設定に成功
} else {
// IP アドレスパラメータ設定に失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22 を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

### 2.3.37. AsP52N:receivedFactoryResetStatus:

- (void)AsP52N:(AsP52N \*)AsP52N

receivedFactoryResetStatus:(AsP52NExecutionStatus)statusCode;

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

■ 説明:

出荷時設定復旧の実行結果を受信します。

factoryReset ([2.2.52](#) 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)AsP52N:(AsP52N *)AsP52N
receivedFactoryResetStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//出荷時設定復旧に成功
} else {
//出荷時設定復旧に失敗
}
```

```
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

### 2.3.38. AsP52N:receivedInventoryStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedInventoryStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 (2.4.9を参照)

**■説明:**  
 インベントリの実行結果を受信します。  
 startInventoryWithEPC (2.2.7を参照)、readTagWithAccessPassword (2.2.9を参照)、readDataWithAntenna1 (2.2.24を参照)、startInventoryWithAntenna1 (2.2.28を参照)、startInventoryTIDWithAntenna1 (2.2.29を参照)、startInventoryMoistureWithAntenna1 (2.2.30を参照)、startInventoryMagnusS3WithAntenna1 (2.2.31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)AsP52N:(AsP52N *)AsP52N
receivedInventoryStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//インベントリ成功
} else {
//インベントリ失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

### 2.3.39. AsP52N:receivedInventoryCustomDataWithRSSI:channel:phase:antenna:PC:EPC:TID:userData:reserved:epcBank:moisture:temperature:

```
- (void)AsP52N:(AsP52N *)AsP52N
receivedInventoryCustomDataWithRSSI:(NSString *)RSSI
    channel:(NSString *)channel
    phase:(NSString *)phase
    antenna:(NSString *)antenna
    PC:(NSString *)PC
    EPC:(NSString *)EPC
    TID:(NSString *)TID
    userData:(NSString *)userData
    reserved:(NSString *)reserved
    epcBank:(NSString *)epcBank
    moisture:(NSString *)moisture
    temperature:(NSString *)temperature;
```

	引数名	In/Output	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	RSSI	Out	NSString	タグの RSSI データ
引数	channel	Out	NSString	タグのチャンネル情報
引数	phase	Out	NSString	タグの位相情報
引数	antenna	Out	NSString	アンテナ情報
引数	PC	Out	NSString	タグの PC データ
引数	EPC	Out	NSString	タグの EPC データ
引数	TID	Out	NSString	タグの TID データ
引数	userData	Out	NSString	タグのユーザエリアデータ
引数	reserved	Out	NSString	タグの Reserved エリアのデータ
引数	epcBank	Out	NSString	タグの EPC エリアデータ
引数	moisture	Out	NSString	タグの湿度データ
引数	temperature	Out	NSString	タグの温度データ

#### ■説明:

インベントリしたデータを受信します。

startInventoryWithEPC ([2.2.7](#)を参照)、readTagWithAccessPassword ([2.2.9](#)を参照)  
readDataWithAntenna1 ([2.2.24](#)を参照)、startInventoryWithAntenna1 ([2.2.28](#)を参照)、  
startInventoryTIDWithAntenna1 ([2.2.29](#)を参照)、startInventoryMoistureWithAntenna1  
([2.2.30](#)を参照)、startInventoryMagnusS3WithAntenna1 ([2.2.31](#)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)AsP52N:(AsP52N *)AsP52N
receivedInventoryCustomDataWithRSSI:(NSString *)RSSI
    channel:(NSString *)channel
    phase:(NSString *)phase
    antenna:(NSString *)antenna
    PC:(NSString *)PC
    EPC:(NSString *)EPC
    TID:(NSString *)TID
    userData:(NSString *)userData
    reserved:(NSString *)reserved
    epcBank:(NSString *)epcBank
    moisture:(NSString *)moisture
    temperature:(NSString *)temperature {
// RSSI      RSSI データ      引数を設定していない場合、戻り値は「@'''」になります。
// channel   チャンネル情報   引数を設定していない場合、戻り値は「@'''」になります。
// phase     タグ位相情報     引数を設定していない場合、戻り値は「@'''」になります。
// antenna   アンテナ情報     引数を設定していない場合、戻り値は「@'''」になります。
// PC        PC データ。      引数を設定していない場合、戻り値は「@'''」になります。
// EPC       EPC データ      引数を設定していない場合、戻り値は「@'''」になります。
// TID       TID データ      引数を設定していない場合、戻り値は「@'''」になります。
// userData  User エリアデータ 引数を設定していない場合、戻り値は「@'''」になります。
// reserved  Reserved エリアのデータ 引数を設定していない場合、戻り値は「@'''」になります。
// epcBank   epcBank データ   引数を設定していない場合、戻り値は「@'''」になります。
// moisture  湿度データ      引数を設定していない場合、戻り値は「@'''」になります。
// temperature  温度データ    引数を設定していない場合、戻り値は「@'''」になります。
}
```

### 2.3.40. AsP52N:receivedWriteDataStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedWriteDataStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)
<p>■説明:</p> <p>データ書き込みの実行結果を受信します。 writeDataWithAntenna1 (<a href="#">2.2.25</a>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>■サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedWriteDataStatus:(AsP52NExecutionStatus)statusCode { if (statusCode == AsP52NExecutionStatusSuccess) { //データ書き込みに成功 } else { //データ書き込みに失敗 getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照) NSString *errorString = [asP52N getMessageWithResultCode:statusCode]; // errorString 失敗原因 } }</pre>				

### 2.3.41. AsP52N:receivedReadDataStatus:

- (void)AsP52N:(AsP52N *)AsP52N receivedReadDataStatus:(AsP52NExecutionStatus)statusCode;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型

(2.4.9 を参照)

## ■ 説明:

データ読取の実行結果を受信します。

readDataWithAntenna1 ([2.2.24](#) 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)AsP52N:(AsP52N *)AsP52N
receivedReadDataStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//データ読取に成功
} else {
//データ読取に失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22 を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}
```

## 2.3.42. AsP52N:receivedLockDataStatus:

```
- (void)AsP52N:(AsP52N *)AsP52N
receivedLockDataStatus:(AsP52NExecutionStatus)statusCode;
```

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 ( <a href="#">2.4.9</a> を参照)

## ■ 説明:

タグロックの実行結果を受信します。

lockTagWithAntenna1 ([2.2.26](#) 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)AsP52N:(AsP52N *)AsP52N
receivedLockDataStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
```

```

//タグロックに成功
} else {
    //タグロックに失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}

```

### 2.3.43. AsP52N:receivedKillDataStatus:

- (void)AsP52N:(AsP52N \*)AsP52N  
receivedKillDataStatus:(AsP52NExecutionStatus)statusCode;

	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	statusCode	Out	AsP52NExecutionStatus	実行結果 AsP52NExecutionStatus 列挙型 (2.4.9を参照)

■説明:

タグキルの実行結果を受信します。

killTagWithAntenna1 (2.2.27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)AsP52N:(AsP52N *)AsP52N
receivedKillDataStatus:(AsP52NExecutionStatus)statusCode {
if (statusCode == AsP52NExecutionStatusSuccess) {
//タグキルに成功
} else {
    //タグキルに失敗
getMessageWithResultCode:メソッドを呼び出し、失敗原因を取得します。(2.2.22を参照)
NSString *errorString = [asP52N getMessageWithResultCode:statusCode];
// errorString 失敗原因
}
}

```

### 2.3.44. AsP52N:receivedUnknowCommand:

- (void)AsP52N:(AsP52N *)AsP52N receivedUnknowCommand:(NSData *)data;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	data	Out	NSData	未知のコマンド
<p>■ 説明:</p> <p>未知のコマンドを受信します。 SDK が未知のコマンドを受信した後、コールバックされます。</p> <p>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedUnknowCommand:(NSData *)data { // data 未知のコマンド。 }</pre>				

### 2.3.45. AsP52N:receivedUnknowError:

- (void)AsP52N:(AsP52N *)AsP52N receivedUnknowError:(NSError *)error;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	error	Out	NSError	エラー
<p>■ 説明:</p> <p>SDK 内部に発生した未知エラーを受信して、アプリケーションに戻します。 SDK 内部にエラーが発生した後、コールバックされます。</p> <p>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedUnknowError:(NSError *)error { // error エラー。 }</pre>				

### 2.3.46. AsP52N:receivedAlarmTemperature:

- (void)AsP52N:(AsP52N *)AsP52N receivedAlarmTemperature:(NSString *)alarmTemperature currentTemperature:(NSString *)currentTemperature;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	alarmTemperature	Out	String	過熱警報
引数	currentTemperature	Out	String	現在の温度
<p>■ 説明:</p> <p>RF タグをインベントリした時に AsReader は過熱になった場合の温度を受信します。 RF タグをインベントリした時に、過熱になった場合にコールバックされます。</p> <p>■ サンプルコード:</p> <pre>- (void)AsP52N:(AsP52N *)AsP52N receivedAlarmTemperature:(NSString *)alarmTemperature currentTemperature:(NSString *)currentTemperature { }</pre>				

### 2.3.47. AsP52N:receivedGetTemperatureAlarmTemperature:

- (void)AsP52N:(AsP52N *)AsP52N receivedGetTemperatureAlarmTemperature:(NSString *)alarmTemperature currentTemperature:(NSString *)currentTemperature;				
	引数名	In/Out	型	説明
引数	AsP52N	Out	AsP52N	AsP52N オブジェクト
引数	alarmTemperature	Out	String	過熱警報
引数	currentTemperature	Out	String	現在の温度
<p>■ 説明:</p> <p>取得した温度を受信します。 getTemperature (<a href="#">2.2.55</a>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>■ サンプルコード:</p>				

```
- (void)AsP52N:(AsP52N *)AsP52N receivedAlarmTemperature:(NSString
*)alarmTemperature currentTemperature:(NSString *)currentTemperature {
}
```

## 2.4. 列挙

### 2.4.1. AsP52NetworkState

定義	説明
AsP52NetworkStateUnknown	未知のネットワーク
AsP52NetworkStateConnected	接続
AsP52NetworkStateDisconnected	切断
AsP52NetworkStateError	エラー発生

### 2.4.2. AsP52NSessionType

定義	説明
AsP52NSessionType_S0	Session0
AsP52NSessionType_S1	Session1
AsP52NSessionType_S2	Session2
AsP52NSessionType_S3	Session3

### 2.4.3. AsP52NMemoryBankType

定義	説明
AsP52NMemoryBankTypeReserved	Reserved エリア
AsP52NMemoryBankTypeEPC	EPC エリア

AsP52NMemoryBankTypeTID	TID エリア
AsP52NMemoryBankTypeUser	ユーザーエリア

#### 2.4.4. AsP52NTargetMemoryType

定義	説明
AsP52NTargetMemoryTypeKill	キルパスワードエリア
AsP52NTargetMemoryTypeAccess	アクセスパスワードエリア
AsP52NTargetMemoryTypeEPC	EPC エリア
AsP52NTargetMemoryTypeTID	TID エリア
AsP52NTargetMemoryTypeUser	ユーザーエリア

#### 2.4.5. AsP52NLockActionType

定義	説明
AsP52NLockActionTypeUnlock	アンロック
AsP52NLockActionTypeLock	ロック
AsP52NLockActionTypePermaUnlock	永久アンロック
AsP52NLockActionTypePermaLock	永久ロック

#### 2.4.6. AsP52NBasebandRateStatus

定義	説明
AsP52NBasebandRateStatus40KHz	Tari = 25us、FM0、LHF=40KHz
AsP52NBasebandRateStatus250KHz	Tari = 25us、Miller4、LHF=250KHz
AsP52NBasebandRateStatus300KHz	Tari=25us、Miller4、LHF=300KHz

AsP52NBasebandRateStatus400KHz	Tari=6.25us、FM0、LHF=400KHz
AsP52NBasebandRateStatus320KHz	Tari=25us、Miller4、LHF=320KHz
AsP52NBasebandRateStatusAuto	AsReader が自動的に設定する

### 2.4.7. AsP52NInventoryFlag

定義	説明
AsP52NInventoryFlag_A	Flag A のみを読み取る
AsP52NInventoryFlag_B	Flag B のみを読み取る
AsP52NInventoryFlag_AB	Flag A と Flag B を読み取る

### 2.4.8. AsP52NRFIDFrequencyStatus

定義	説明
AsP52NRFIDFrequencyStatus920_925MHz	GB 920~925MHz
AsP52NRFIDFrequencyStatusFCC	FCC 902~928MHz
AsP52NRFIDFrequencyStatusETSI	ETSI 866~868MHz
AsP52NRFIDFrequencyStatusJP	JP 916.8~920.8MHz

### 2.4.9. AsP52NExecutionStatus

定義	説明
AsP52NExecutionStatusSuccess	成功
AsP52NExecutionStatusCommandExecutionError	コマンド実行エラー
AsP52NExecutionStatusPacketSendFailed	パケット送信に失敗
AsP52NExecutionStatusReadDataLabelNotRespond	タグが無反応

AsP52NExecutionStatusReadDataCRCError	CRC エラー
AsP52NExecutionStatusReadDataAreaLocked	データエリアがロックされる
AsP52NExecutionStatusReadDataAreaOverflows	データエリアのオーバーフロー
AsP52NExecutionStatusReadDataAccessPasswordError	アクセスパスワードエラー
AsP52NExecutionStatusReadDataOtherLabelsError	その他タグエラー
AsP52NExecutionStatusReadDataOtherReaderError	その他リーダーエラー
AsP52NExecutionStatusWriteMemoryAntennaPortParameterError	アンテナのポート引数エラー
AsP52NExecutionStatusWriteMemorySelectedParameterError	引数選択エラー
AsP52NExecutionStatusWriteMemoryWriteParameterError	引数書き込みエラー
AsP52NExecutionStatusWriteMemoryCRCCheckError	CRC チェックエラー
AsP52NExecutionStatusWriteMemoryInsufficientPower	パワー不足
AsP52NExecutionStatusWriteMemoryDataAreaOverflow	データエリアのオーバーフロー
AsP52NExecutionStatusWriteMemoryDataAreaLocked	データエリアがロックされる
AsP52NExecutionStatusWriteMemoryAccessPasswordError	アクセスパスワードエラー
AsP52NExecutionStatusWriteMemoryOtherLabelError	その他タグエラー
AsP52NExecutionStatusWriteMemoryLabelMissing	タグがロスされる
AsP52NExecutionStatusWriteMemoryStatusCommandError	リーダーコマンド送信エラー
AsP52NExecutionStatusLockMemoryAntennaPortError	アンテナのポートエラー
AsP52NExecutionStatusLockMemorySelectedParameterError	引数選択エラー

AsP52NExecutionStatusLockMemoryLockOperationParameterError	ロック操作の引数エラー
AsP52NExecutionStatusLockMemoryCRCCheckError	CRC チェックエラー
AsP52NExecutionStatusLockMemoryInsufficientPower	パワー不足
AsP52NExecutionStatusLockMemoryDataOverflow	データエリアのオーバーフロー
AsP52NExecutionStatusLockMemoryDataLocked	データエリアがロックされる
AsP52NExecutionStatusLockMemoryAccessPasswordError	アクセスパスワードエラー
AsP52NExecutionStatusLockMemoryOtherLabelError	その他タグエラー
AsP52NExecutionStatusLockMemoryLabelMissing	タグがロスされる
AsP52NExecutionStatusLockMemoryCommandError	リーダーコマンド送信エラー
AsP52NExecutionStatusKillMemoryAntennaPortParameterError	アンテナのポート引数エラー
AsP52NExecutionStatusKillMemorySelectedParameterError	引数選択エラー
AsP52NExecutionStatusKillMemoryCRCCheckError	CRC チェックエラー
AsP52NExecutionStatusKillMemoryInsufficientPower	パワー不足
AsP52NExecutionStatusKillMemoryInactivationPasswordError	キルパスワードエラー
AsP52NExecutionStatusKillMemoryOtherLabelError	その他タグエラー
AsP52NExecutionStatusKillMemoryLabelMissing	タグがロスされる
AsP52NExecutionStatusKillMemoryInstructionError	リーダーコマンド送信エラー
AsP52NExecutionStatusAntennaPowerPortParameterNotSupport	リーダーのハードウェアがポート引数をサポートしない
AsP52NExecutionStatusAntennaPowerParameterNotSupport	リーダーのハードウェアがパワー引数をサポートしない

AsP52NExecutionStatusAntennaPowerSaveFailed	保存に失敗
AsP52NExecutionStatusAutoDormancyModeParameterErrors	モード引数エラー
AsP52NExecutionStatusAutoDormancyOtherParameterErrors	その他引数エラー
AsP52NExecutionStatusAutoDormancySaveFailed	保存に失敗
AsP52NExecutionStatusSetRFIDFrequencyBandParameterNotSupported	リーダーのハードウェアが周波数引数をサポートしない
AsP52NExecutionStatusSetRFIDFrequencySaveFailed	保存に失敗
AsP52NExecutionStatusTagLogParameterError	引数エラー
AsP52NExecutionStatusTagLogSaveFailed	保存に失敗
AsP52NExecutionStatusFrequencyChannelNotInFrequencyBand	チャンネル番号は現在の周波数帯にない
AsP52NExecutionStatusFrequencyInvalidFrequencyPoints	無効の周波数
AsP52NExecutionStatusFrequencyOtherParameterErrors	その他引数エラー
AsP52NExecutionStatusFrequencySaveFailed	保存に失敗
AsP52NExecutionStatusReaderTimeRTCSettingFailed	RTC 設定に失敗
AsP52NExecutionStatusEthernetIPReaderIPParameterError	リーダー IP 引数エラー
AsP52NExecutionStatusClearCacheError	キャッシュクリアエラー
AsP52NExecutionStatusFactoryResetError	出荷時設定の復元エラー
AsP52NExecutionStatusInventoryErrorAntennaPortParameterError	アンテナのポート引数エラー
AsP52NExecutionStatusInventoryErrorSelectionReadParameterError	読取選択引数エラー
AsP52NExecutionStatusInventoryErrorTIDReadParameterError	TID 読取引数エラー

terError	
AsP52NExecutionStatusInventoryErrorUserDataAreaReadParameterError	ユーザーエリア読取引数エラー
AsP52NExecutionStatusInventoryErrorReservedAreaReadParameterError	Reserved エリア読取引数エラー
AsP52NExecutionStatusInventoryErrorOtherParameterError	その他引数エラー
AsP52NExecutionStatusBasebandRateNotSupported	リーダーがサポートしていないベースバンドレート
AsP52NExecutionStatusBasebandQValueParameterError	Q 値引数エラー
AsP52NExecutionStatusBasebandSessionParameterError	Session 引数エラー
AsP52NExecutionStatusBasebandInventoryParameterError	読取引数エラー
AsP52NExecutionStatusBasebandOtherParametersError	その他引数エラー
AsP52NExecutionStatusBasebandSaveFailed	保存に失敗

## 3 付録

### 3.1. Frequency List

AsP52NRFIDFrequencyStatus 列挙 ([2.4.8](#)を参照) に対応するチャンネル値に関する説明です。

#### 3.1.1. AsP52NRFIDFrequencyStatus920\_925MHz

Channel	Frequency (MHz)	Channel	Frequency (MHz)
0	920.625	8	922.625
1	920.875	9	922.875
2	921.125	10	923.125
3	921.375	11	923.375
4	921.625	12	923.625
5	921.875	13	923.875
6	922.125	14	924.125
7	922.375	15	924.375

#### 3.1.2. AsP52NRFIDFrequencyStatusFCC

Channel	Frequency (MHz)	Channel	Frequency (MHz)
0	902.750	25	915.250
1	903.250	26	915.750
2	903.750	27	916.250

3	904.250	28	916.750
4	904.750	29	917.250
5	905.250	30	917.750
6	905.750	31	918.250
7	906.250	32	918.750
8	906.750	33	919.250
9	907.250	34	919.750
10	907.750	35	920.250
11	908.250	36	920.750
12	908.750	37	921.250
13	909.250	38	921.750
14	909.750	39	922.250
15	910.250	40	922.750
16	910.750	41	923.250
17	911.250	42	923.750
18	911.750	43	924.250
19	912.250	44	924.750
20	912.750	45	925.250
21	913.250	46	925.750
22	913.750	47	926.250
23	914.250	48	926.750
24	914.750	49	927.250

### 3.1.3. AsP52NRFIDFrequencyStatusETSI

Channel	Frequency (MHz)	Channel	Frequency (MHz)
0	865.700	3	867.500
1	866.300	4	868.100
2	866.900		

### 3.1.4. AsP52NRFIDFrequencyStatusJP

Channel	Frequency (MHz)	Channel	Frequency (MHz)
0	916.800	3	920.400
1	918.000	4	920.600
2	919.200	5	920.800