

# Scanner プロバイダ デンソー製スキャナ プロバイダ

Version 1.0.6

## ユーザーズ ガイド

November 7, 2022

備考:

## 【改版履歴】

バージョン	日付	内容
1.0.0	2018-01-09	初版
1.0.1	2018-04-19	AddController 時のオプション引数 Unicode の削除 CaoController::Execute メソッド Raw 実行時のスキャン文字列末尾に null 文字が付与されるバグを修正
1.0.2	2018-08-10	ImageCapture,MemUpload コマンドの仕様変更 Timeout オプション省略時の動作を改善 付録 A を追記
	2018-11-05	変数一覧に補足を追記
	2019-01-08	対応機種追加
1.0.3	2019-11-25	AddController に Encode オプションを追加
1.0.4	2020-02-14	AddController に Sleep オプションを追加
	2020-03-02	コードスキャナの設定方法を付録 B に追加
1.0.5	2020-07-08	@QUEUE に BSTR 型で put できないバグを修正 ImageCapture の挙動を修正 IoT Data Share での取り扱い方法を追記 マニュアルを全体的にリファイン
1.0.6	2021-01-21	AddController オプションに異常値を指定時のハンドルリークを修正. キュー利用の場合に Contoller 削除時にクリアされない問題を修正.
	2022-11-07	対応機種追加. ユーザーズガイドを微修正.

**【対応機種】**

機種	注意事項
AT シリーズ	
GT シリーズ	
SE シリーズ	SE1-BUB の RFID 制御は未対応
WS シリーズ	
QK シリーズ	
QB シリーズ	
QD シリーズ	
SF シリーズ	
SC1-QB	

## 目次

1. はじめに .....	6
1.1. 接続構成 .....	6
1.1.1. Active USB-COM ポートドライバによる USB 接続 .....	6
2. プロバイダの概要 .....	7
2.1. 概要 .....	7
2.1.1. 動作モード .....	7
2.1.1.1. イベント型モード .....	7
2.1.1.2. キュー型モード .....	7
2.1.1.3. コマンド型モード .....	7
2.1.2. 通信フォーマット .....	8
2.1.3. イメージキャプチャ機能 .....	8
2.2. メソッド・プロパティ .....	8
2.2.1. CaoWorkspace::AddController メソッド .....	8
2.2.1.1. Conn オプション .....	9
2.2.1.2. Protocol オプション .....	10
2.2.2. CaoController::AddVariable メソッド .....	10
2.2.3. CaoController::GetVariableNames メソッド .....	10
2.2.4. CaoController::Execute メソッド .....	10
2.2.5. CaoController::OnMessage イベント .....	11
2.2.6. CaoVariable::get_Value プロパティ .....	11
2.2.7. CaoVariable::put_Value プロパティ .....	11
2.3. 変数一覧 .....	12
2.3.1. コントローラクラス .....	12
2.4. エラーコード .....	13
3. コマンドリファレンス .....	14
3.1. コマンド送信 .....	14
3.1.1. CaoController::Execute (“Raw”) コマンド .....	14
3.2. 画像取得 .....	15
3.2.1. CaoController::Execute (“ImageCapture”) コマンド .....	15
3.2.2. CaoController::Execute (“MemUpload”) コマンド .....	15
3.3. メッセージキュー操作 .....	16
3.3.1. CaoController::Execute (“QueueClear”) コマンド .....	16

---

3.4. 読み取りコマンド送信 .....	16
3.4.1. CaoController::Execute (“OneShot”) コマンド .....	16
4. サンプルプログラム .....	17
5. IoT Data Share でのプロバイダ使用方法 .....	19
5.1. プロジェクト概要 .....	19
5.2. プロジェクト作成 .....	19
5.2.1. コントローラ追加 .....	19
5.2.2. アイテム追加 .....	22
5.2.3. トリガアクション設定 .....	23
付録 A. IoT Data Server での USB コードスキャナの設定方法 .....	26
付録 A.1. USB コードスキャナの COM ポート番号の確認方法 .....	26
付録 A.2. USB ポートと COM ポート番号の固定 .....	27
付録 B. コードスキャナ設定ソフトによる設定方法 .....	28
付録 B.3. 設定手順 .....	28
付録 B.4. 動作確認 .....	30

## 1. はじめに

このドキュメントは、デンソー製コードスキャナ用のプロバイダである Scanner プロバイダのユーザーズガイドです。

Scanner プロバイダは、コードスキャナからデータを出力するプロバイダです。また、コマンドを送信することでイメージキャプチャ機能を使用することもできます。

本書は、この Scanner プロバイダの機能と実装されているメソッドについて説明します。

### 1.1. 接続構成

図 1-1 が本プロバイダとコードスキャナの全体構成図になります。



図 1-1 接続構成図

コードスキャナとプロバイダはシリアル通信を行います。通信ケーブルには RS232C ケーブルと USB ケーブルが存在し、コードスキャナの種類によって対応しているケーブルが異なります。また、USB ケーブルをお使いの場合は別途仮想 COM ドライバの Active USB-COM ポートドライバ<sup>1</sup>が必要になります。

接続方式	必要ソフト
RS-232C	不要
USB	Active USB-COM ポートドライバ

#### 1.1.1. Active USB-COM ポートドライバによる USB 接続

Active USB-COM ポートドライバのインストーラに従ってインストール後、コードスキャナを接続します。

デバイスマネージャにてポート(図 1-2 を参照)を確認します。Connected と書かれている COM ポート番号が本プロバイダで接続する際に必要となるので控えておいてください。



図 1-2 USB ケーブルでスキャナを接続

<sup>1</sup> Active USB-COM ポートドライバは DENSO-WAVE の HP(<https://www.denso-wave.com/ja/>)より入手できます。

## 2. プロバイダの概要

### 2.1. 概要

コードスキャナで値を取得する方法として、コードをスキャンした時点でその値を出力するイベント(プッシュ)方式と、コードスキャナにコマンドを送信し値を取得するコマンド(プル)方式があります。

Scanner プロバイダではイベント方式で値を取得する手段として、イベントとキューを実装しており、AddController 時に選択することができます。コマンド方式で値を取得する手段として、変数クラスに @ONE\_SHOT 変数, Execute コマンドに OneShot コマンドを実装しています。

#### 2.1.1. 動作モード

##### 2.1.1.1. イベント型モード

イベント型モードではコードスキャナがデータを取得したときに OnMessage イベントを発生させます。このためクライアントは、このイベントで取得した Message オブジェクトの中に格納されているデータから値を取得することができます。

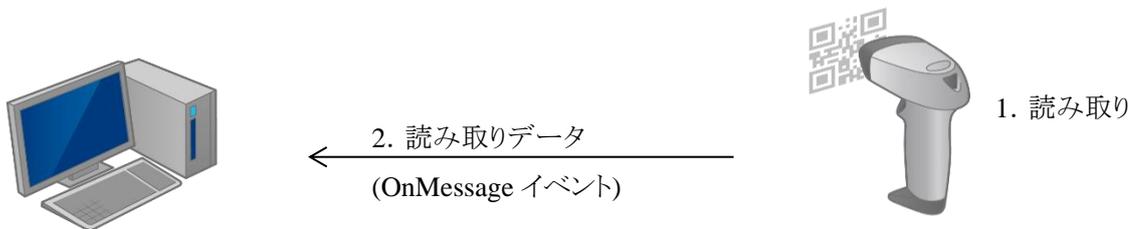
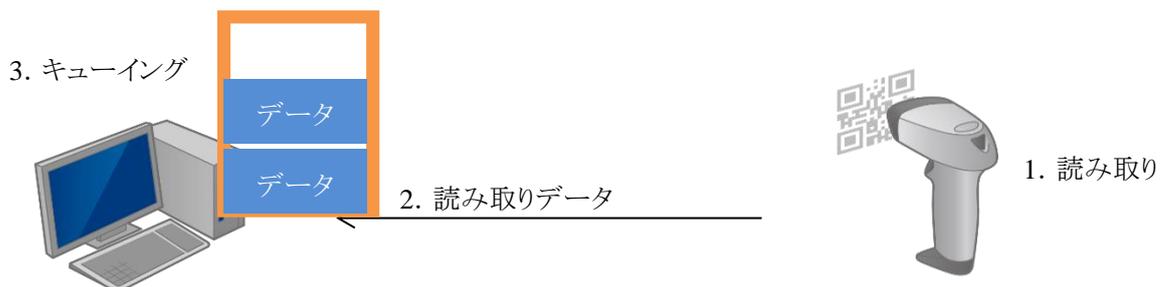


図 2-1 イベント型モードの動作フロー

##### 2.1.1.2. キュー型モード

キュー型モードでは、コードスキャナからデータを取得した際、まずはキュー(先入れ先出しバッファ)にキューイングします。クライアントプログラムは、"@QUEUE"システム変数を使って任意のタイミングでキューから値を取得することができます。100 個以上の読み取りを行った場合は破棄されます。



##### 2.1.1.3. コマンド型モード

コマンド型モードでは、システム変数"@ONE\_SHOT"で値を取得時に、コマンドを送信してデータを取得します。このモードはコードスキャナが連続読み取りモードに設定されているときのみ使用することができます。

す。



図 2-2 コマンド型モードでの動作フロー

### 2.1.2. 通信フォーマット

本プロバイダを使用する際はコードスキャナの通信フォーマットを表 2-1 の通りに設定してください。

BCC の転送だけは必ず禁止に設定してください。コードマーク、桁数等の読み取りデータに付随させる項目を「許可」にした場合は、受信データの一部とされます。適宜、分解して利用してください。

また、ヘッダやターミネータは受信データには含まれません。

表 2-1: 通信フォーマット

パラメータ	設定値
ヘッダ	なし
ターミネータ	CR
BCC の転送	禁止 (固定)

### 2.1.3. イメージキャプチャ機能

Scanner プロバイダでは一部のコードスキャナでイメージキャプチャ機能を使用することができます。

イメージキャプチャ機能を使用するには、Controller::Execute メソッドでイメージキャプチャコマンドを送信します。この後、取り込んだ映像データは OnMessage イベントでクライアントに送信されます。データは Message オブジェクトの Value プロパティにバイナリ配列として格納されています。

また、キュー型モードで起動しているときにイメージスキャナ機能を使用しても、データはキューには格納されず必ずイベントで送信されます。

## 2.2. メソッド・プロパティ

### 2.2.1. CaoWorkspace::AddController メソッド

Scanner プロバイダでは AddController 時に、通信用の接続パラメータを参照し、通信の接続を行います。このときオプションで通信形態、接続パラメータ、タイムアウトの設定を指定します。

```

AddController
(
    "<コントローラ名>", // コントローラ名 (任意文字列)

```

```

"GaoProv. DENSO. Scanner ",           // プロバイダ名. 固定.
"<マシン名>",                         // プロバイダの実行マシン名.
"<オプション>"                          // オプション文字列
)

```

以下にオプション文字列に指定するリストを示します.

表 2-2 CaoWorkspace::AddController のオプション文字列

オプション	意味
Conn=<接続パラメータ>	必須. 通信形態とその接続パラメータを設定します. 詳細は 2.2.1.1 に示します.
Trigger[=<動作モード>]	プロバイダ動作モード. ‘1’-イベント有効, ‘2’-キュー有効, ‘3’-コマンド有効 Execute(OneShot)や変数(@ONE_SHOT)を使用して読み取る. (デフォルト:1)
Protocol[=<通信オプション>]	通信時のモードやヘッダ, ターミネータの設定をします. (デフォルト:”0:0:0”) 詳細は 2.2.1.2 に示します.
Timeout[=<タイムアウト時間>]	送受信時のタイムアウト時間を指定します. (デフォルト:500[ms])
MessageMax[=<Message オブジェクト生成数>]	Scanner プロバイダが生成する Message オブジェクト数の制限を設定します. (デフォルト:100)
Encode[=<読み取りデータのエンコード>]	読み取りデータ出力時のエンコード方式を設定します. ‘1’-現在の Windows ANSI コードページ <sup>2</sup> , ‘2’-UTF-8 (デフォルト:1)
Sleep[=<受信待ち間隔>]	スキャナからの読み取りデータ受信待ち間隔を設定します. (デフォルト:1[ms], 最大 1000[ms])

### 2.2.1.1. Conn オプション

以下に Conn オプションの接続パラメータ文字列を示します. ここで角括弧(“[ ]”)内は省略可能を示します. また, 各パラメータの解説中の下線部はオプションを指定しなかったときのデフォルト値を示します.

```
“com:<COM Port>[:<BaudRate>[:<Parity>:<DataBits>:<StopBits>]]”
```

<COM Port> : COM ポート番号. ‘1’-COM1, ‘2’-COM2, ...

<BaudRate> : 通信速度. 4800, 9600, 19200, 38400, 57600, 115200.

<sup>2</sup> 使用している OS の言語に依存します.

<Parity>	:	パリティ. <u>‘N’-NONE</u> , ‘E’-EVEN, ‘O’-ODD.
<DataBits>	:	データビット数. ‘7’-7bit, <u>‘8’-8bit</u> .
<StopBits>	:	ストップビット数. <u>‘1’-1bit</u> , ‘2’-2bit.

### 2.2.1.2. Protocol オプション

以下に Protocol オプションのパラメータ文字列を示します. ここで角括弧(“[ ]”)内は省略可能を示します. また, 各パラメータの解説中の下線部はオプションを指定しなかったときのデフォルト値を示します.

“Protocol = [<Protocol>[:<Header>[:<Term>]]”

<Protocol>	:	通信プロトコル. <u>‘0’-ノープロトコル</u> , ‘1’-ACK/NAK プロトコル.
<Header>	:	ヘッダ指定. <u>‘0’-なし</u> , ‘1’-STX(0x05)
<Term>	:	ターミネータ指定. <u>‘0’-CR(0x0D)</u> , ‘1’-LF(0x0A), ‘2’-CR+LF(0x0D0A), ‘3’-ETX(0x03)

### 2.2.2. CaoController::AddVariable メソッド

変数オブジェクトを取得します. このメソッドで指定できる変数名は 2.3 変数一覧に示しているシステム変数のみです. これ以外の変数名を使用した場合, このメソッドは失敗します.

**書式** AddVariable ( <bstrVariableName:VT\_BSTR>, [<bstrOption:VT\_BSTR>] )

```
AddVariable
(
    “<変数名>”, // システム変数名
    “<オプション>” // オプション文字列 (未使用)
)
```

#### 使用例

```
// Add variable to get device version
CaoVariable val;
val = caoCtrl.AddVariable("@VERSION");

// GetValue
var value = val.Value;
```

### 2.2.3. CaoController::GetVariableNames メソッド

2.3 変数一覧に示しているシステム変数名の一覧を取得します.

### 2.2.4. CaoController::Execute メソッド

コードスキャナにコマンドを送信します. 第 1 引数にコマンド名, 第 2 引数にコマンドのパラメータを指定します. コマンド送信の可否は HRESULT で返ります.

各コマンドの詳細は 3 章コマンドリファレンスを参照してください.

**書式** Execute ( <bstrCommandName:VT\_BSTR>,[<vntParam : VT\_VARIANT>])  
 bstrCommandName: [in] コマンド名  
 vntParam : [in] パラメータ

### 2.2.5. CaoController::OnMessage イベント

Scanner プロバイダがデータを受信すると、CaoController クラスの OnMessage イベントとしてクライアントにデータを受け渡します。このとき、Message::Value プロパティに受信データをそのまま格納します。

OnMessage イベントは、読み取りデータ受信、イメージスキャナ機能によるキャプチャ画像受信、イメージスキャナ機能によるサムネイル画像受信の 3 通りがあり、それぞれは Message::Number プロパティにより区別されます。以下に Message::Number プロパティとデータ種別の対応を示します。

**表 2-3 Message::Number プロパティの値とデータ種別の対応**

Number プロパティ	データ種別	データ型
1	読み取りデータ	VT_BSTR
2	キャプチャ画像	VT_UI1   VT_ARRAY
3	サムネイル画像	VT_UI1   VT_ARRAY

### 2.2.6. CaoVariable::get\_Value プロパティ

2.3 変数一覧に示したシステム変数に対応した値を取得します。

### 2.2.7. CaoVariable::put\_Value プロパティ

システム変数名が、”@QUEUE”のときのみ実装されています。

指定した値が VT\_BSTR ならば、そのデータをキューに格納します。

指定した値が VT\_EMPTY ならば、バッファをクリアして、キューを空の状態にします。

## 2.3. 変数一覧

### 2.3.1. コントローラクラス

表 2-4 コントローラクラス システム変数一覧

変数名	データ型	説明	属性	
			get	put
@ONE_SHOT	VT_BSTR	コードスキャナに読み取りコマンドを送信し、一定時間内に読み取ったデータを取得します。 この変数を使用するときには、コードスキャナを連続読み取りモードに設定してください。 <b>【注意】</b> コードスキャナには”R”, ”Z”を送信していますが、未対応の場合は@ONE_SHOT は使用できません。対応状況はコードスキャナのマニュアルを参照してください。 また,CaoWorkspace::AddController メソッド時のオプション引数の Trigger が‘3’の時のみ使用可能です。	○	-
@ID	VT_BSTR	コードスキャナのシリアル番号を取得します。	○	-
@MAKER_NAME	VT_BSTR	“DENSO WAVE Inc.”	○	-
@VERSION	VT_BSTR	コードスキャナのファームウェアバージョンを取得します。	○	-
@QUEUE	VT_BSTR	スキャンした値を格納するキュー VT_BSTR 型であれば任意の文字列を Put できます。	○	○
@QUEUE_SIZE	VT_I4	キューの現在サイズ	○	-
@QUEUE_MAX	VT_I4	キューの最大値サイズ 固定値:100	○	-
@STATE	VT_I4	接続状態 0:切断状態, 1:接続状態, 2:エラー ※RS-232 接続のときのみ有効. USB 接続時は1:接続状態で固定されます。	○	-

#### <補足>

@VERSION および@ID 変数による値取得の際にかかる時間は AddController オプションで設定した Timeout に依存します。これは本プロバイダの仕様で、スキャナ応答を待つ受信スレッドが動作しており、コマンド送信にも使用するソケットを使用していることに由来します。Timeout 毎にソケットが開放されコマンド送信

ができるようになり、また取得時間にバラツキが生じます。

値取得時間を短縮したい場合は Timeout を短く設定してください。

## 2.4. エラーコード

Scanner プロバイダで定義されている独自のエラーコードを表 2-5 に示します。

表 2-5 独自エラーコード一覧

エラー名	エラー番号	説明				
E_OPTION_INVALID	0x80100001	<p>Trigger オプションが‘3’以外の時,以下のコマンド等を実行するとこのエラーが返ります.</p> <table border="1"> <tr> <td>CaoController::Execute (“OneShot”) コマンド</td> </tr> <tr> <td>CaoController::Execute (“ImageCapture”) コマンド</td> </tr> <tr> <td>CaoController::Execute (“MemUpload”) コマンド</td> </tr> <tr> <td>@OneShot 変数の取得.</td> </tr> </table> <p>Trigger オプションを‘3’に設定してやり直してください.</p>	CaoController::Execute (“OneShot”) コマンド	CaoController::Execute (“ImageCapture”) コマンド	CaoController::Execute (“MemUpload”) コマンド	@OneShot 変数の取得.
CaoController::Execute (“OneShot”) コマンド						
CaoController::Execute (“ImageCapture”) コマンド						
CaoController::Execute (“MemUpload”) コマンド						
@OneShot 変数の取得.						

その他は ORiN2 共通エラーについては、「ORiN2 プログラミングガイド」のエラーコードの章を参照してください。

## 3. コマンドリファレンス

本章では CaoController::Execute メソッドの各コマンドについて解説します。

表 3-1 CaoController::Execute コマンド一覧

コマンド	機能	
コマンド送信		
Raw	コードスキャナ専用のコマンドを送信します	p. 14
画像取得		
ImageCapture	画像を取得します	p. 15
MemUpload	画像を取得します(一部機種で使用可能)	p. 15
メッセージキュー操作		
QueueClear	メッセージキューをクリアします。	p. 16
読み取りコマンド送信		
OneShot	コードスキャナに読み取り可能コマンドを送信します。	p. 16

### 3.1. コマンド送信

#### 3.1.1. CaoController::Execute (“Raw”) コマンド

スキャナにコマンドを送信します。

指定可能なコマンドの種類は、各コードスキャナのマニュアルを参照してください。

ただし、IMAGEOUT コマンドおよび MEMUPLOAD コマンドはこのメソッドでは使用できません。使用する場合は 3 章コマンドリファレンスに記載の”ImageCapture”および”MemUpload”メソッドを使用してください。

**書式** Raw(<bstrParam >

bstrParam : [in] 送信するコードスキャナのコマンド文字列

戻り値 : [out] なし

コマンドによりスキャナから応答がある場合は、

- 動作モードが「イベント有効」の場合、OnMessage イベント
- 動作モードが「キュー有効」の場合、@QUEUE 変数

で受け取ります。

#### 使用例1

```
“LOAD” command send by Raw command
ctrl.Execute “Raw”, “LOAD#1”
```

#### 使用例1

```
“RESET” command send by Raw command
ctrl.Execute “Raw”, “RESET”
```

## 3.2. 画像取得

### 3.2.1. CaoController::Execute (“ImageCapture”) コマンド

コードスキャナから画像を取得します。



ImageCapture(<iSize >, <iType>, <iThumbnail>)

**iSize** : [in] キャプチャ画像のサイズ(VT\_UI1)  
 コードスキャナにより画像サイズが異なります。  
 詳細は各コードスキャナのマニュアルを参照してください。  
 0:1/1 サイズ  
 1:1/4 サイズ  
 2:1/16 サイズ

**iType** : [in] キャプチャ画像の取得領域(VT\_UI1)  
 0:全画像エリア  
 1:画像中央部

**iThumbnail** : [in] サムネイル画像の有無(VT\_UI1)  
 0:サムネイル送信禁止  
 1:1/4 サイズ JPEG サムネイル送信

**戻り値** : [out] なし

画像は OnMessage イベントで受け取ります。

<注意>

本 Execute コマンドで使用しているのはコードスキャナの”IMAGEOUT”コマンドですが、キャプチャ画像は BMP, サムネイル画像 JPEG に限定しています。

CaoWorkspace::AddController メソッド時のオプション引数の Trigger が ‘3’ の時のみ使用可能です。



```
Imagecapture
ctrl.Execute “ImageCapture”, Array (0, 0, 0)
```

### 3.2.2. CaoController::Execute (“MemUpload”) コマンド

コードスキャナから画像を取得します。

QB シリーズなど、一部コードスキャナでは本 Execute コマンドにより画像取得ができます。詳細は各コードスキャナのマニュアルを参照してください。



MemUpload(<iMemNumber >)

**iMemNumber** : [in] 画像メモリ番号(VT\_UI1) (0~255)  
 ※指定できる画像メモリ番号はスキャナにより異なります。詳細は各コードスキャナのマニュアルを参照してください。

**戻り値** : [out]なし

画像は OnMessage イベントで受け取ります。

<注意>

CaoWorkspace::AddController メソッド時のオプション引数の Trigger が‘3’の時のみ使用可能です。

#### 使用例

```
‘ MemUpload  
ctrl.Execute “MemUpload”, 0
```

### 3.3. メッセージキュー操作

#### 3.3.1. CaoController::Execute (“QueueClear”) コマンド

メッセージキューをクリアします。

#### 書式

QueueClear()

戻り値 : [out]なし

#### 使用例

```
‘ QueueClear  
ctrl.Execute “QueueClear”
```

### 3.4. 読み取りコマンド送信

#### 3.4.1. CaoController::Execute (“OneShot”) コマンド

コードスキャナに読み取り可能コマンドを送り、

何かを読む、もしくは、一定時間経過で読み取り停止コマンドを送ります。

#### 書式

OneShot(<iTimeout>)

iTimeout : [in] 読み取り待ち時間(VT\_UI2)  
ミリ秒で指定して下さい。

戻り値 : [out] 読み取り結果(VT\_BSTR)

<注意>

CaoWorkspace::AddController メソッド時のオプション引数の Trigger が‘3’の時のみ使用可能です。

#### 使用例

```
Private read As String  
‘ OneShot  
Set read = ctrl.Execute(“OneShot”, 500)
```

## 4. サンプルプログラム

以下にコードスキャナが読んだコードデータをテキストボックスに出力するサンプルを示します。

プロバイダはイベントモードで動作し、コードスキャナからの読み取りデータを受信後、OnMessage で出力します。

### List 4-1 SampleText.frm

```
Private eng As CaoEngine
Private WithEvents ctrl As CaoController

Private Sub Form_Load()

    Set eng = New CaoEngine

    ' Scanner との接続処理
    Set ctrl = eng.Workspaces(0).AddController("Sample", _
        "CaoProv.DENSO.Scanner", _
        "", _
        "Conn=com:1:38400:N:8:1, _
        Trigger=1, _
        Protocol=0:0:0")

End Sub

' データの受信
Private Sub ctrl_OnMessage(ppCaoMess As CAOLib.ICaoMessage)

    text1.Text = ppCaoMess.Value

End Sub
```

以下にイメージキャプチャ機能で取り込んだ画像をファイルに保存する場合のサンプルを示します。

**List 4-2****SampleImage.frm**

```
Private eng As CaoEngine
Private WithEvents ctrl As CaoController

Private Sub Form_Load()

    Set eng = New CaoEngine

    ' Scanner との接続処理
    Set ctrl = eng.Workspaces(0).AddController("Sample", _
        "CaoProv.DENSO.Scanner", _
        "", _
        "Conn=com:1:38400:N:8:1, _
        Trigger=1, _
        Protocol=0:0:0")

End Sub

' イメージキャプチャコマンドの送信
Private Sub Command1_Click()

    ctrl.Execute "ImageCapture", Array(0,0,0)

End Sub

' 画像の受信
Private Sub ctrl_OnMessage(ByVal pICaoMess As CAOLib.ICaoMessage)
    ' 画像データを一時ファイルに保存
    Open "picture.bmp" For Binary As #1
    Dim byData() As Byte
    byData = pICaoMess.Value
    Put #1, , byData
    Close #1
End Sub
```

## 5. IoT Data Share でのプロバイダ使用方法

本節では Scanner プロバイダを IoT Data Share にてプロジェクトを作成して使用する例を説明します。

### 5.1. プロジェクト概要

スキャナの読み取りデータをアイテムに格納するプロジェクトの作成方法を説明します。

作成するプロジェクトは以下の条件で作成しています。

IoT Data Share	バージョン	1.10.0
スキャナ	種類	GT20Q-SM
	設定	全項目デフォルト 特にトリガスイッチは「オートオフモード」
COM ポート番号		5

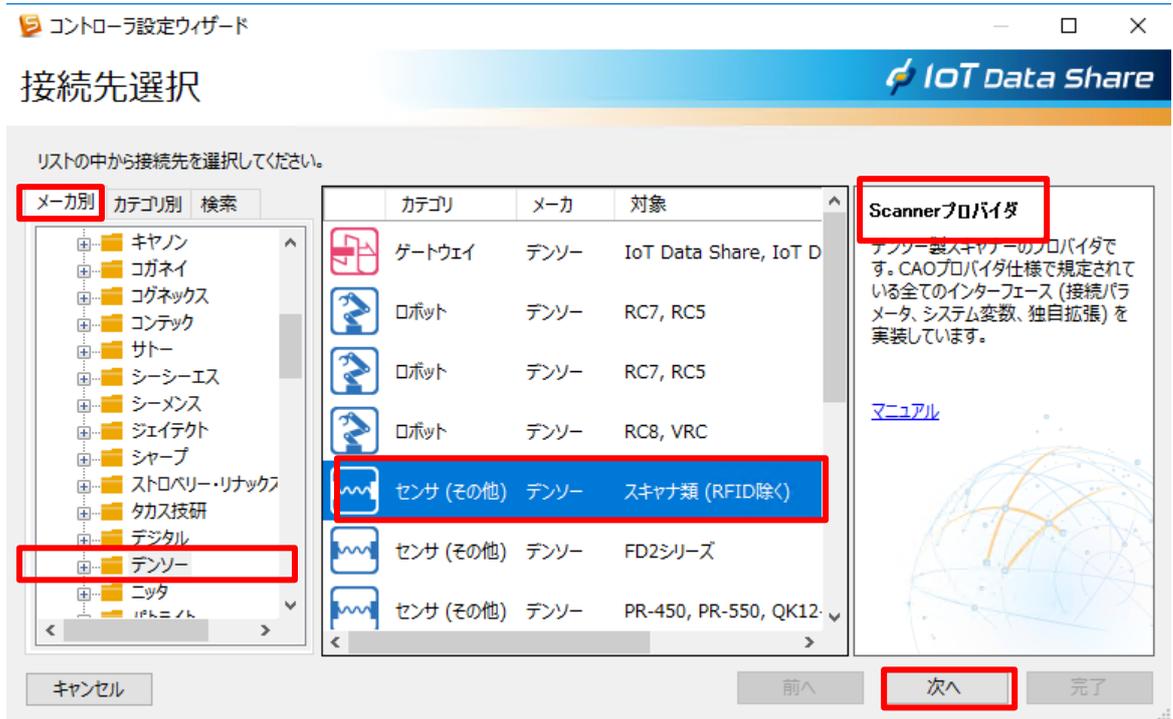
### 5.2. プロジェクト作成

#### 5.2.1. コントローラ追加

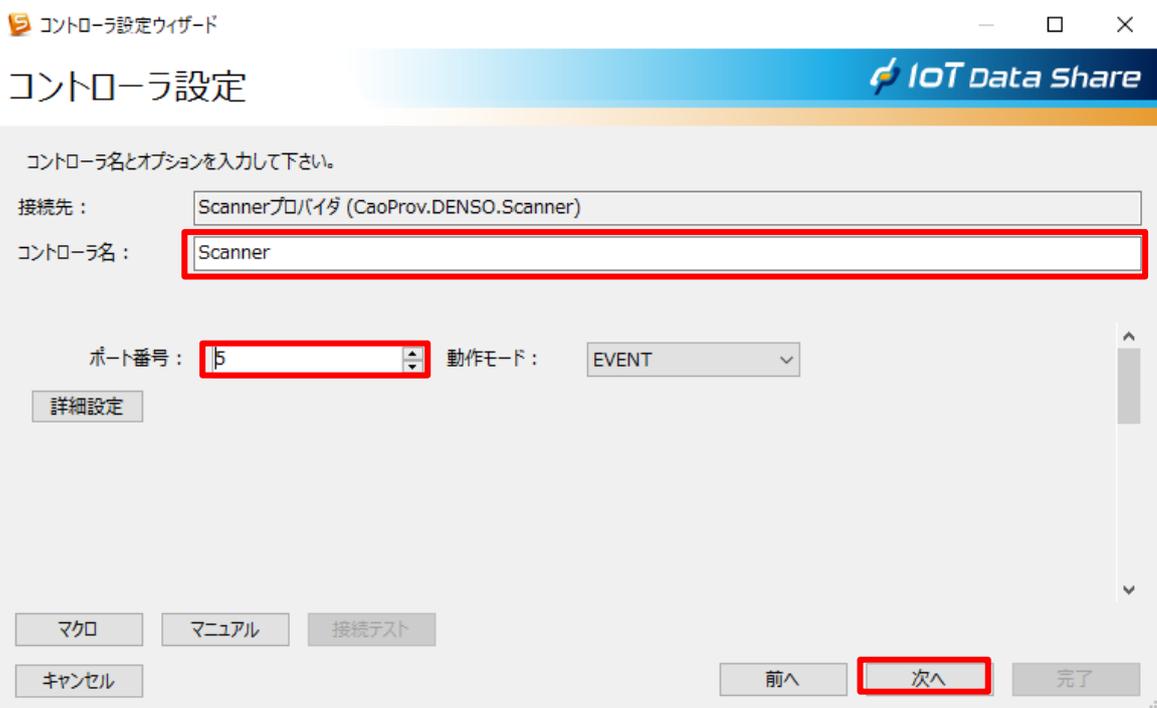
プロジェクトを作成し、Scanner プロバイダを使うコントローラを設定します。

手順	内容
1	<p>スタートページの「コントローラ設定ウィザードを起動する」を押します。</p> 

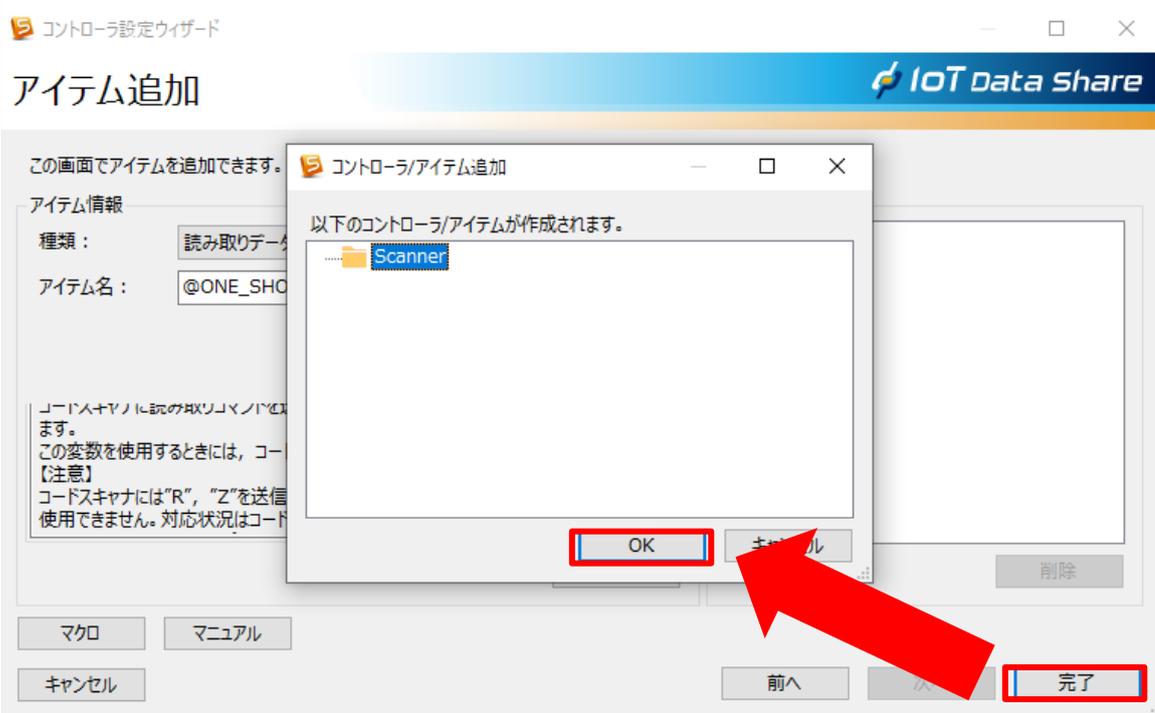
- 2 メーカー別欄から「デンソー」、Scanner プロバイダを選択し「次へ」ボタンを押します。



- 3 「ポート番号」に 1.1.1 章で確認した COM ポート番号を入力し、「次へ」ボタンを押します。コントローラ名は任意の名前を入力してください。例では Scanner を入力しています。

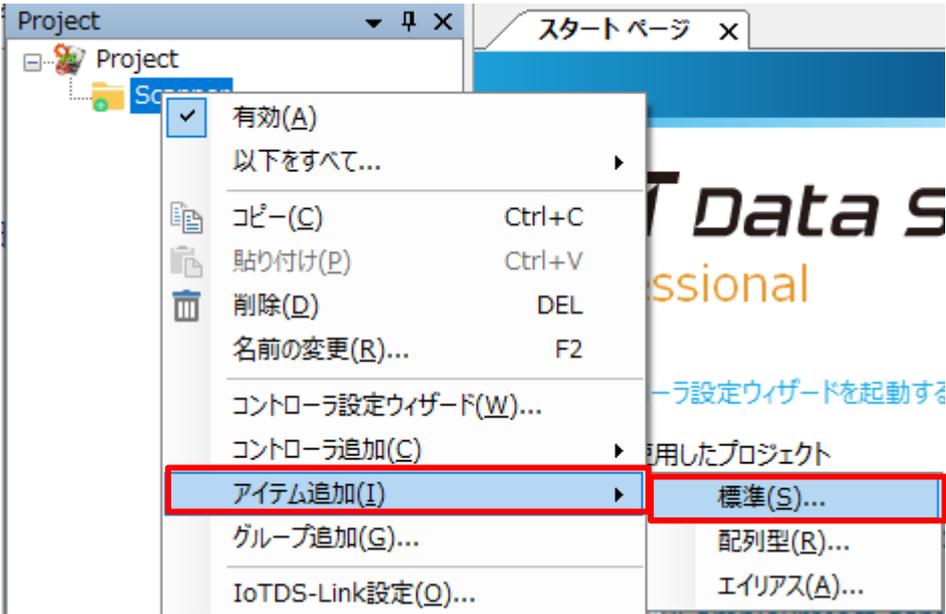
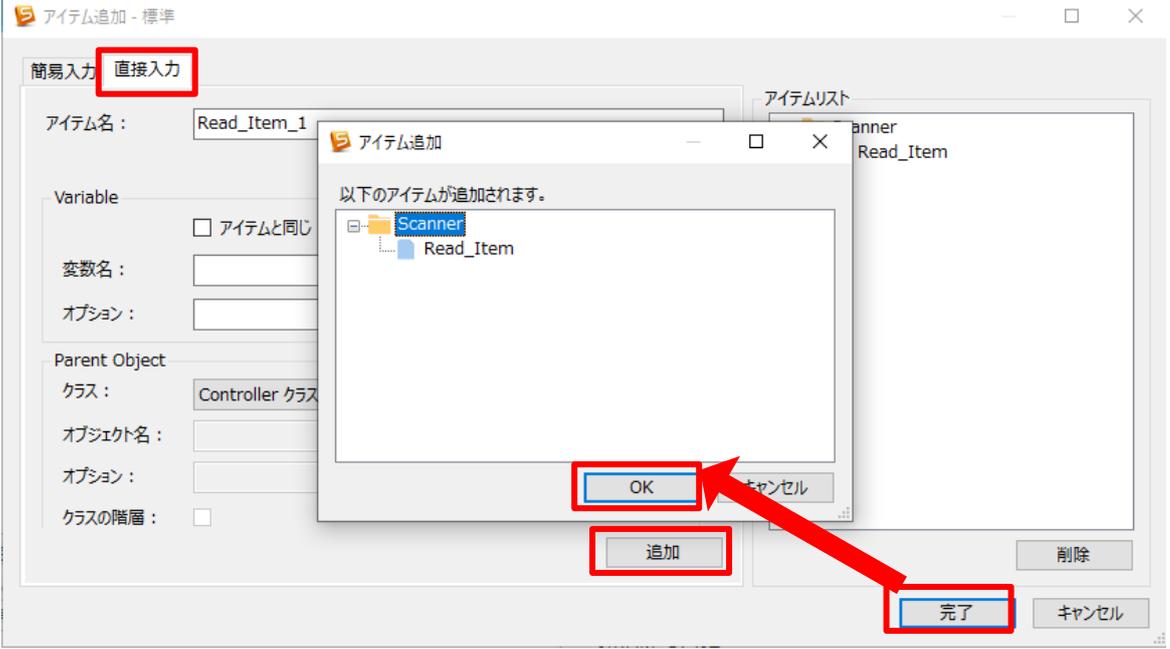


- 4 「完了」ボタンを押し、表示されるダイアログで「OK」ボタンを押します。



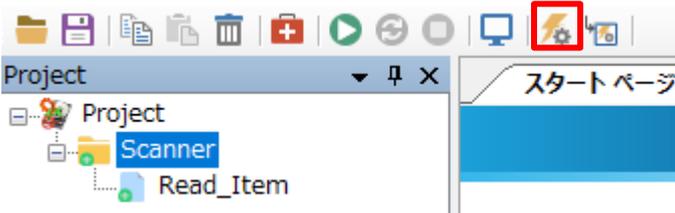
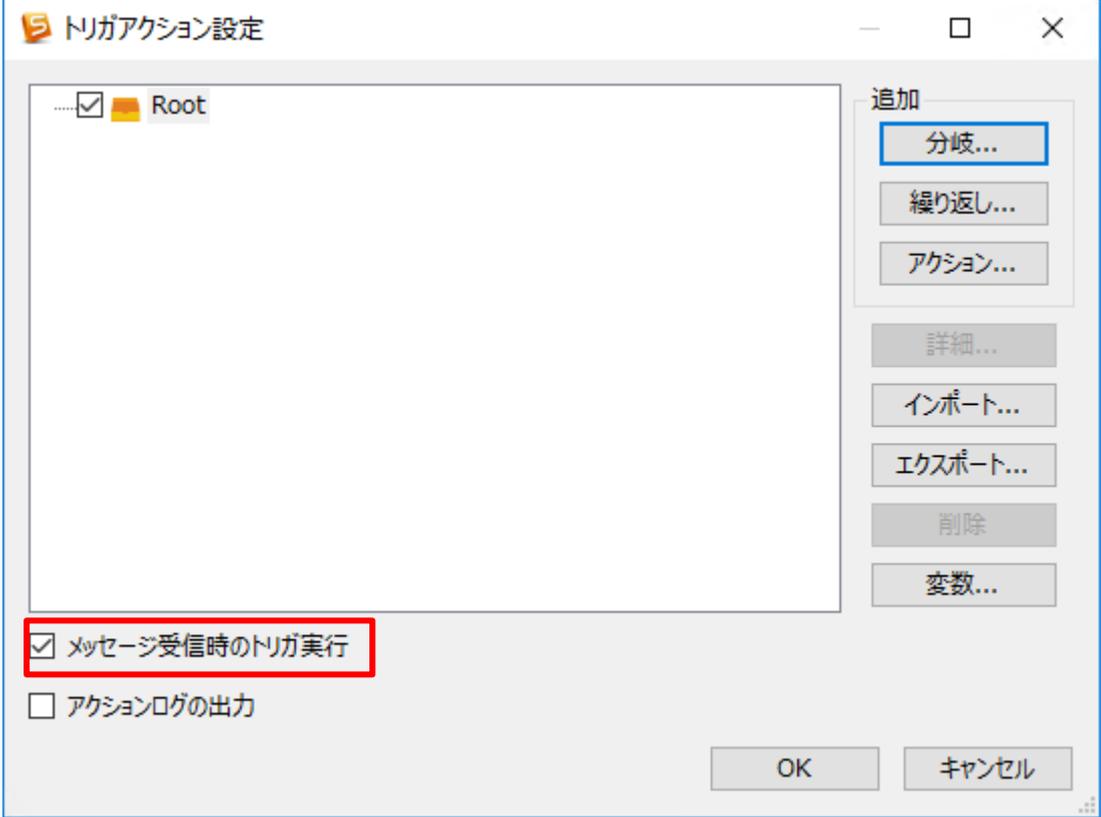
## 5.2.2. アイテム追加

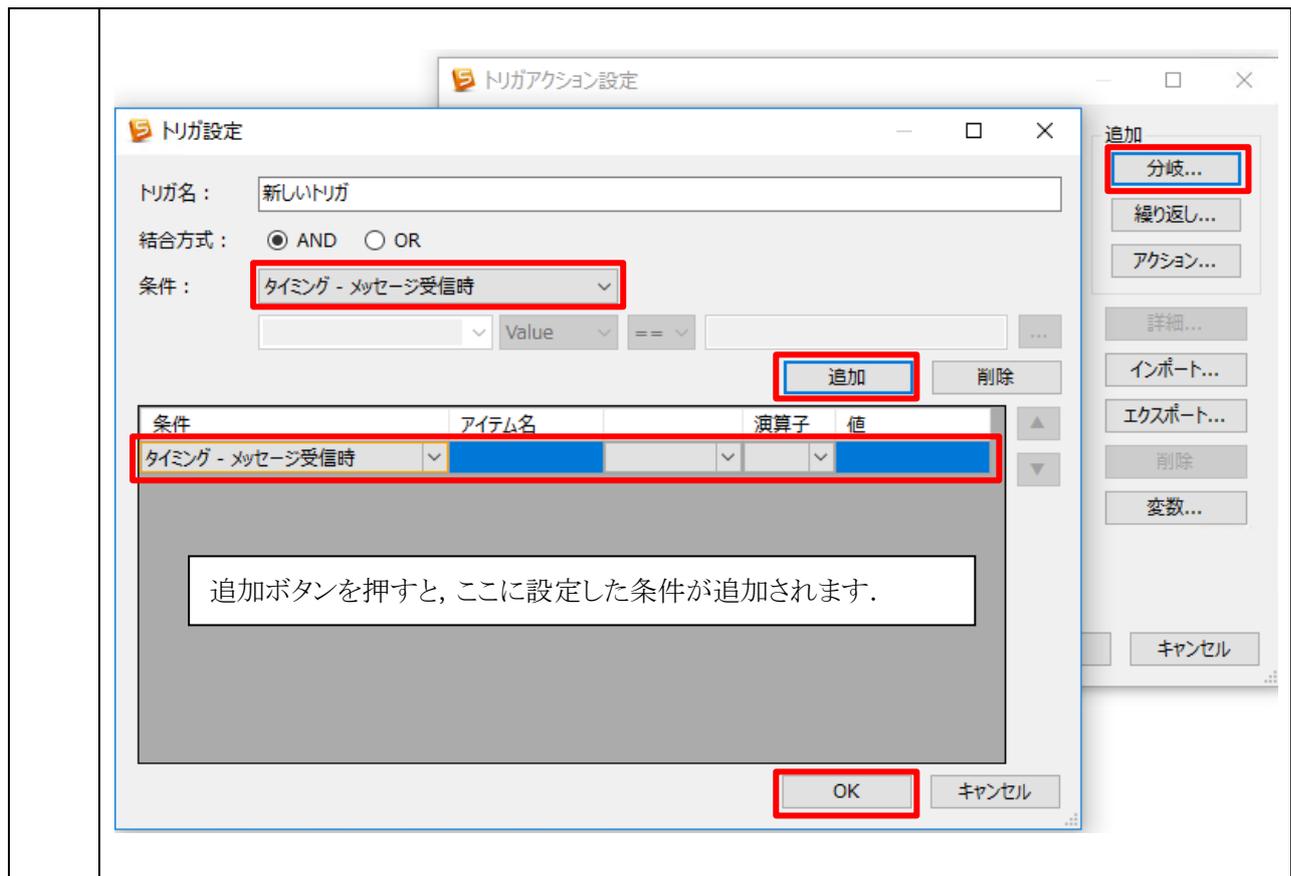
読み取りデータを格納するアイテムを追加します。

手順	内容
1	<p>5.2.1 章で追加したコントローラを右クリックし、「アイテム追加」→「標準」を押します。</p> 
2	<p>直接入力タブを選択, 追加ボタン, 完了ボタンの順に押し, 表示されるダイアログで OK ボタンを押します. アイテム名は任意のアイテム名を入力してください. 例では Read_Item を入力しています.</p> 

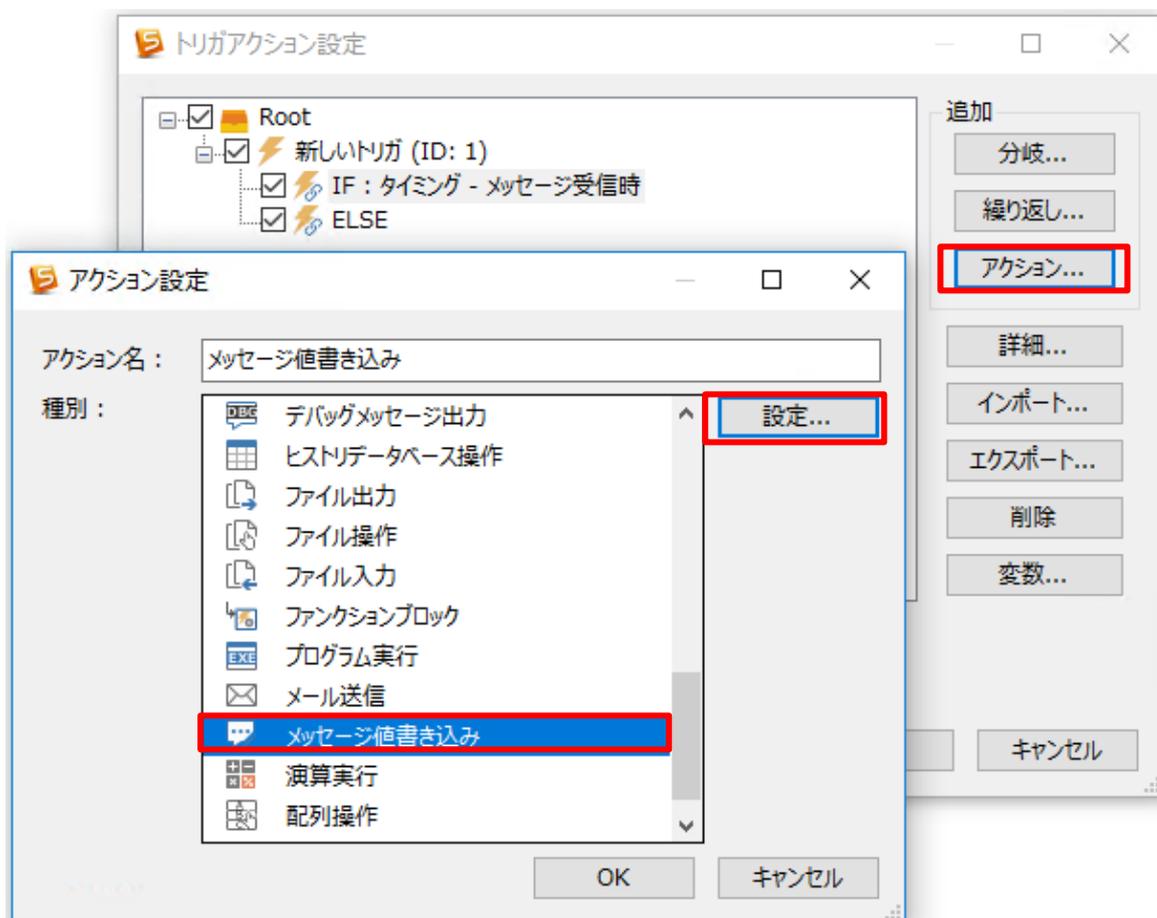
### 5.2.3. トリガアクション設定

読み取りデータを 5.2.2 章で追加したアイテムに格納するようにトリガアクションを設定します。

手順	内容		
1	<p>ツールバーより赤枠のアイコンを押します。</p> 		
2	<p>「メッセージ受信時のトリガ実行」にチェックを入れます</p> 		
3	<p>分岐ボタンを押し、トリガ設定ダイアログを表示させます。 このダイアログで以下の内容を設定し、追加ボタン、OK ボタンの順に押します。</p> <table border="1" data-bbox="279 1758 1444 1803"> <tr> <td>条件</td> <td>タイミング - メッセージ受信時</td> </tr> </table>	条件	タイミング - メッセージ受信時
条件	タイミング - メッセージ受信時		



- 4 「アクション」ボタンを押し、アクション設定ダイアログ一覧から「メッセージ値書き込み」を選択、設定ボタンを押します。



- 5 メッセージ値書き込み設定では下図の設定をし、「OK」ボタンを押します。



## 付録A. IoT Data Server での USB コードスキャナの設定方法

本節では IoT Data Server で Scanner プロバイダを使う際の COM ポート番号確認と設定方法を説明します。

なお、IoT Data Server の Version 1.3.0 以降では USB タイプのコードスキャナを使用することができます。

### 付録A.1. USB コードスキャナの COM ポート番号の確認方法

AddController に必要な COM ポート番号を確認します。

手順	内容																																										
1	コードスキャナを USB ポートに挿し込みます。																																										
2	<p>トップ画面から「システム設定」を選択します。</p> 																																										
3	<p>左側ツリーから「拡張デバイス」→「シリアル通信」を押すと、COM ポート番号一覧が表示されます。その中から「DENSO WAVE Active USB-COM Port」で「COM* - Connected」となっている COM ポート番号を確認します。下図の例では COM5 が相当します。</p> <p>システム設定 - 拡張デバイス - シリアル通信</p> <table border="1"> <thead> <tr> <th>ネットワーク</th> <th>COM</th> <th>Port Name</th> </tr> </thead> <tbody> <tr> <td>Ethernet 1</td> <td>COM1</td> <td>Communications Port</td> </tr> <tr> <td>Ethernet 2</td> <td>COM2</td> <td>Communications Port</td> </tr> <tr> <td>Ethernet 3</td> <td>COM3</td> <td>Communications Port</td> </tr> <tr> <td>プロキシ</td> <td>COM4</td> <td>Communications Port</td> </tr> <tr> <td>ファイアウォール</td> <td>COM5 - Connected</td> <td>DENSO WAVE Active USB-COM Port</td> </tr> <tr> <td>Windows資格情報</td> <td>COM6 - Disconnected</td> <td>DENSO WAVE Active USB-COM Port</td> </tr> <tr> <td>拡張デバイス</td> <td>COM7 - Disconnected</td> <td>DENSO WAVE Active USB-COM Port</td> </tr> <tr> <td>シリアル通信</td> <td>COM8 - Disconnected</td> <td>DENSO WAVE Active USB-COM Port</td> </tr> <tr> <td>CONTEC</td> <td>COM9 - Disconnected</td> <td>DENSO WAVE Active USB-COM Port</td> </tr> <tr> <td>Hilscher</td> <td>COM10 - Disconnected</td> <td>DENSO WAVE Active USB-COM Port</td> </tr> <tr> <td>HMS</td> <td></td> <td></td> </tr> <tr> <td>PHOENIX CONTACT</td> <td></td> <td></td> </tr> <tr> <td>パスワード</td> <td></td> <td></td> </tr> </tbody> </table>	ネットワーク	COM	Port Name	Ethernet 1	COM1	Communications Port	Ethernet 2	COM2	Communications Port	Ethernet 3	COM3	Communications Port	プロキシ	COM4	Communications Port	ファイアウォール	COM5 - Connected	DENSO WAVE Active USB-COM Port	Windows資格情報	COM6 - Disconnected	DENSO WAVE Active USB-COM Port	拡張デバイス	COM7 - Disconnected	DENSO WAVE Active USB-COM Port	シリアル通信	COM8 - Disconnected	DENSO WAVE Active USB-COM Port	CONTEC	COM9 - Disconnected	DENSO WAVE Active USB-COM Port	Hilscher	COM10 - Disconnected	DENSO WAVE Active USB-COM Port	HMS			PHOENIX CONTACT			パスワード		
ネットワーク	COM	Port Name																																									
Ethernet 1	COM1	Communications Port																																									
Ethernet 2	COM2	Communications Port																																									
Ethernet 3	COM3	Communications Port																																									
プロキシ	COM4	Communications Port																																									
ファイアウォール	COM5 - Connected	DENSO WAVE Active USB-COM Port																																									
Windows資格情報	COM6 - Disconnected	DENSO WAVE Active USB-COM Port																																									
拡張デバイス	COM7 - Disconnected	DENSO WAVE Active USB-COM Port																																									
シリアル通信	COM8 - Disconnected	DENSO WAVE Active USB-COM Port																																									
CONTEC	COM9 - Disconnected	DENSO WAVE Active USB-COM Port																																									
Hilscher	COM10 - Disconnected	DENSO WAVE Active USB-COM Port																																									
HMS																																											
PHOENIX CONTACT																																											
パスワード																																											

<RS-232C の場合> RS-232C ポートとポート番号が既に紐付いているので、IoT Data Server のマニュアルを参照してください。
--

## 付録A.2. USB ポートと COM ポート番号の固定

IoT Data Server では予め USB ポートの数だけ Active USB-COM ポートドライバがインストールされており、コードスキャナを挿し込むと一時的に小さい順番で COM ポート番号が割り当てられます。IoT Data Server の電源を切ると COM ポート番号はリセットされるため、再度電源を入れてコードスキャナを挿し込むと、ポート番号が変わる可能性があります。

USB ポートと COM ポート番号とを永続的に紐付ける方法は、下記手順になります。

既に COM ポート番号が割り付けられている USB ポートに接続する場合、下記手順は不要です。

手順	操作内容
1	IoT Data Server の電源が入っていない状態でコードスキャナを USB 接続します。
2	IoT Data Server の電源を入れると数回再起動し、コードスキャナを認識します。
3	IoT Data Server のトップ画面から[システム設定]ー[シリアル通信]画面を確認します。
4	[DENSO WAVE Active USB-COM Port(COM**) - Connected]となっている項目を確認します。以降、手順 1 で接続した物理 USB ポートに確認した COM ポート番号が割り振られます。

## 付録B. コードスキャナ設定ソフトによる設定方法

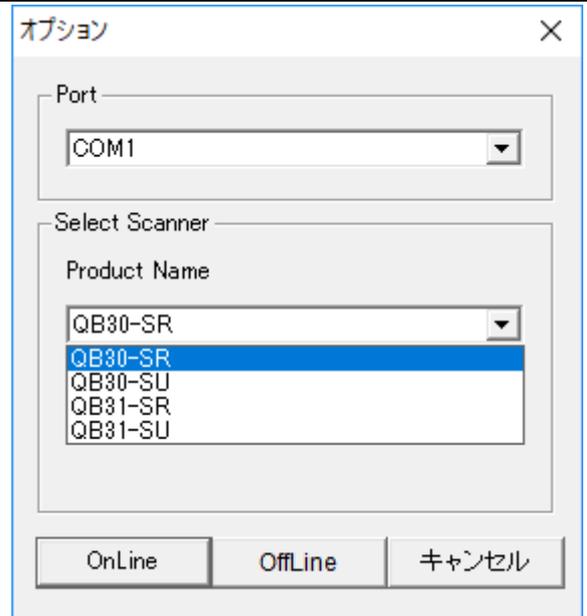
この付録ではコードスキャナ QB30-SR を例に, 設定ソフト(QB Setting)を使つての@ONE\_SHOT 変数を使用する例を示します. この付録で使う製品とバージョンは表 5-1 の通りです.

表 5-1 動作確認製品とバージョン

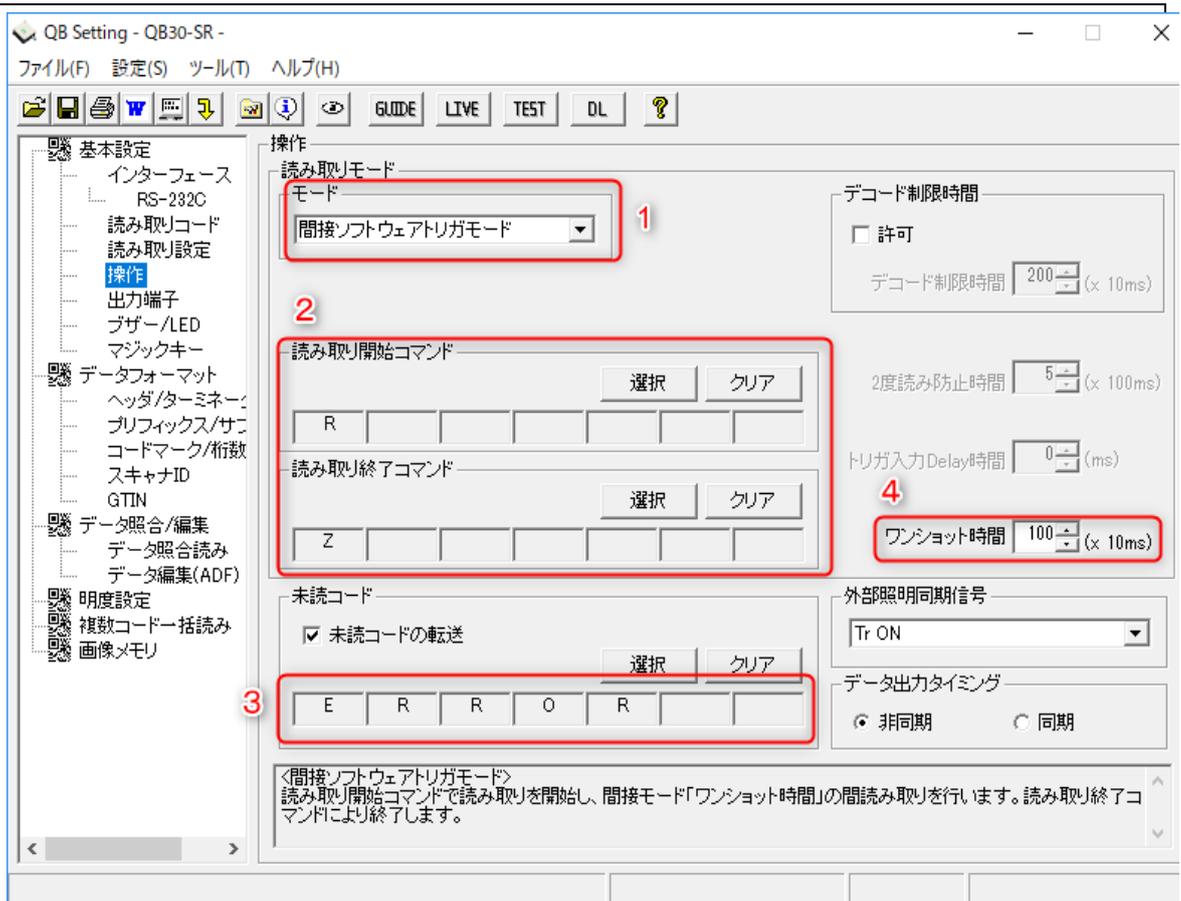
製品	バージョン
QB30-SR	1.19
QB Setting	1.2.1

### 付録B.3. 設定手順

手順	内容
1	<p>QB Setting を起動します.</p> <p>起動直後の画面(右図参照)では, 接続している COM ポート番号と接続する QB30 の機器種類を選択してください.</p> <p>OnLine を押すと, 選択した COM ポートと機器種類を使って接続し設定画面が表示されます.</p> <p>OffLine を押すと, 機器種類にあった設定画面が表示されます.</p>



2



左端のツリーから「操作」を選択し、上図のように設定します。

各設定項目の概要は以下のとおりです。

番号	概要
1	読み取りモード: 間接ソフトウェアモード コマンドで読み取り開始/停止を制御できるモード。
2	読み取り開始コマンド: R 読み取り開始コマンドとして送るコマンドを設定します。@ONE_SHOT 変数では読み取り開始コマンドは R を想定していますので、ここでは”R”を設定ください。  読み取り停止コマンド: Z 読み取り停止コマンドとして送るコマンドを設定します。@ONE_SHOT 変数では読み取り停止コマンドは Z を想定していますので、ここでは”Z”を設定ください。
3	未読コード: 任意 読み取れないままワンショット時間が経過すると返る文字列を設定します。
4	ワンショット時間: 任意 読み取り開始コマンド受信後に読み取りする時間です。この時間を過ぎてもコードが読めない場合、コードスキャナは未読コードを送ります。

5	 <p>赤枠で囲ったアイコンを押して、設定を QB30-SR に反映させます。</p>
---	---

## 付録B.4. 動作確認

この項では CaoTester を使って動作確認します。

手順	内容
1	<p>CaoTester を起動し、以下のように AddController オプションを設定します。</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">AddController:</div> <p style="text-align: center;">Conn=com:1,Trigger=3,Timeout=1500</p> <p>Com ポート番号は接続できる番号に変更してください。Timeout はコードスキャナに設定したワンショット時間以上に設定してください。</p>
2	@ONE_SHOT 変数を AddVariable します。
3	<p>QR コードを読める位置に置き、GetValue します。</p> <p>読めた場合: GetValue 横に読み取った内容が表示されます。</p> <p>読めなかった場合: コードスキャナに設定した未読コードが表示されます。</p> <p>※コードスキャナのワンショット時間 &gt; AddController の Timeout オプションの状況で読めなかった場合、未読コードは返らずタイムアウトエラー(HRESULT)が返ります。未読コードにより読み取り可否を判別する場合は、ワンショット時間 &lt; AddController の Timeout オプションとなるように設定してください。</p>