

CaoOPCUA (OPC UA Server)

Version 1.5.0

ユーザーズ ガイド

June 25, 2024

備考:

【改版履歴】

バージョン	日付	内容
1.0.0	2014-02-10	初版.
1.0.1	2014-06-12	Appendix A. 追記
1.0.2	2016-02-02	アイテム属性の説明追記
1.0.3	2016-07-26	同じ親オブジェクトへの参照を持つ変数の対応追加.
1.1.0	2018-05-29	I8, UI8 型対応.
1.1.1	2019-02-14	ログ出力追加.
1.1.2	2020-03-12	複数書き込み対応.
1.1.3	2020-06-17	コントローラ再接続の不具合修正.
1.1.3	2020-07-08	VARIANT 型データの説明修正.
1.1.4	2020-07-13	再接続時の不具合修正.
1.1.5	2020-10-06	型変換の不具合修正.
1.1.6	2021-02-09	暗号化時の不具合修正. ユーザ名の不具合修正. ログ出力の不具合修正.
1.1.7	2021-03-12	メモリーク バグ修正.
1.2.0	2021-07-20	内部使用ライブラリの更新.
	2021-08-23	ノード ID とアクセスパスの説明追加.
1.2.1	2021-09-03	ソースコードの難読化.
1.2.2	2021-11-02	メモリーク修正.
1.3.0	2021-11-04	セキュリティポリシー追加.
1.3.1	2021-11-22	IPv6 の不具合修正.
	2021-11-25	証明書の複数 IP アドレス対応.
1.3.2	2021-11-29	AE の複数コントローラの不具合修正.
1.4.0	2022-07-29	内部使用ライブラリの更新.
1.5.0	2024-06-25	内部使用ライブラリの更新.

目次

1. はじめに	6
2. CaoOPCUA の概要	7
2.1. 概要	7
2.2. ノード ID	8
2.3. アクセスパス	8
3. CaoOPCUA サーバ	9
3.1. CaoOPCUA サーバの起動	9
3.1.1. ライセンスキーの登録	9
3.1.2. CaoOPCUA サーバの設定	10
3.1.3. コントローラと変数の設定	11
3.1.4. CaoOPCUA サーバの登録・起動	15
3.2. OPC UA クライアントからの CaoOPCUA サーバへの接続	16
3.2.1. Softing 社「UA Demo Client」	16
3.2.2. Unified Automation 社「UAexpert」	21
3.3. CaoOPCUA 詳細	25
3.3.1. アプリケーション証明書	25
3.3.2. セキュリティポリシーとセキュリティモード	25
3.3.2.1. セキュリティポリシー	25
3.3.2.2. セキュリティモード	26
3.3.2.3. 設定可能なセキュリティの組み合わせ	26
3.3.2.4. セキュリティ使用の接続時に必要な証明書	27
3.3.3. ユーザ認証と権限	27
3.3.3.1. 匿名 (Anonymous) 認証	27
3.3.3.2. ユーザ名認証	27
3.3.3.3. 証明書認証	27
3.3.3.3.1. ユーザ証明書	28
4. CaoOPCUAConfig	29
4.1. 概要	29
4.2. 操作方法	29
4.2.1. タブ入力	29
4.2.1.1. エンジンタブ	30
4.2.1.2. コントローラタブ	31

4.2.1.3. アイテムタブ	32
4.2.2. メニュー	36
4.2.2.1. ファイルメニュー	36
4.2.2.2. 編集メニュー	37
4.2.2.3. アクションメニュー	38
4.2.2.4. ヘルプメニュー	42
4.3. INI ファイル構成情報	43
5. OPC UA プロバイダ	44
5.1. 概要	44
5.2. メソッド・プロパティ	45
5.2.1. CaoWorkspace::AddController メソッド	45
5.2.2. CaoController::AddVariable メソッド	45
Appendix A. RC8 を使用した CaoOPCUA サーバ	46
Appendix A.1. CaoOPCUA サーバの起動	46
Appendix A.2. CaoOPCUA サーバのアイテム編集	53
Appendix A.3. CaoOPCUA サーバとのセキュアな接続	55

1. はじめに

本書は CAO(Contoller Access Object)のクライアントアプリケーションとして位置付けられる CaoOPCUA のユーザーズガイドです。

CaoOPCUA は CAO モジュールを 1 つの FA 機器とみなし, OPC UA(OLE for Process Control Unified Architecture)の機能を提供する OPC UA サーバです(図 1-1)。 CAO は様々な FA 機器との通信を可能にするモジュールなので, その上のアプリケーションである CaoOPCUA も当然様々な FA 機器に接続して使うことができます。

また, CaoOPCUA は OPC UA サーバの機能を提供するモジュールなので, 市販されている OPC UA クライアントソフトウェアを活用することもできます。このことは「CAO プロバイダを実装すれば, CAO クライアントアプリケーションに加えて, OPC UA クライアントアプリケーションも使える」ことを意味し, よりバラエティに富んだシステム開発が可能になります。まさに, これが CaoOPCUA の目的です。

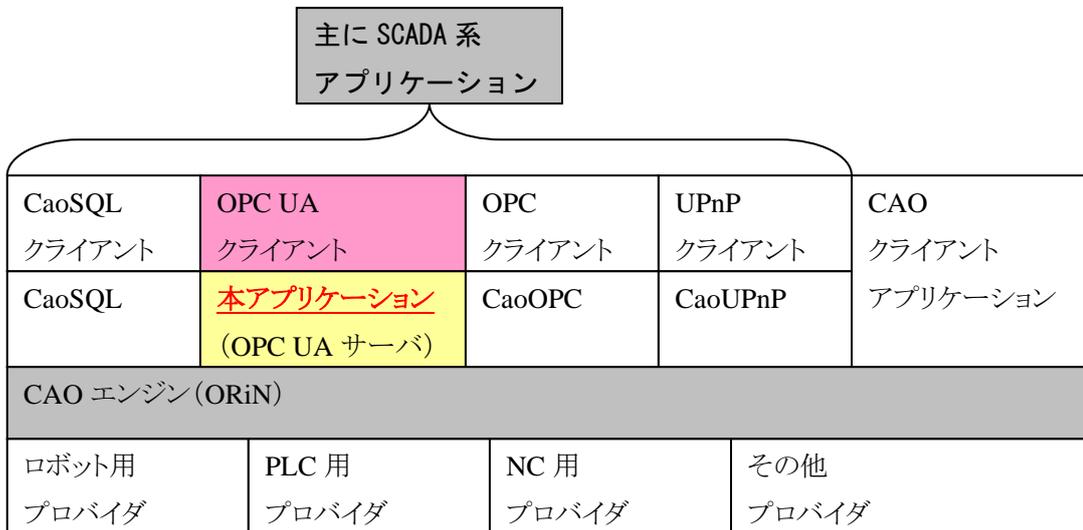


図 1-1 CAO アプリケーションの形態

2. CaoOPCUA の概要

2.1. 概要

CaoOPCUA の構成を下図に示します。CaoOPCUA は OPC UA クライアントと CAO を接続するミドルウェアであり、OPC UA クライアントからの要求を CAO の機能を使って処理する OPC UA サーバです。CaoOPCUA は1つの独立したプロセス(アウトプロセスサーバ)で、複数の OPC UA クライアントに対してサービスを提供します。

CaoOPCUA が OPC UA サーバとして公開する情報やサーバ動作オプション・アイテムは予め専用設定ツールである CaoOPCUAConfig で設定する必要があります。

CaoOPCUAConfig ツールは、CaoSQLConfig を利用し、CAO の CaoController オブジェクト、CaoVariable オブジェクトを作成するのに必要なパラメータ、CaoOPCUA の動作オプションを設定します。このツールで設定された内容は、INI ファイルと csq ファイルに保存され、CaoOPCUA の起動時に参照します。

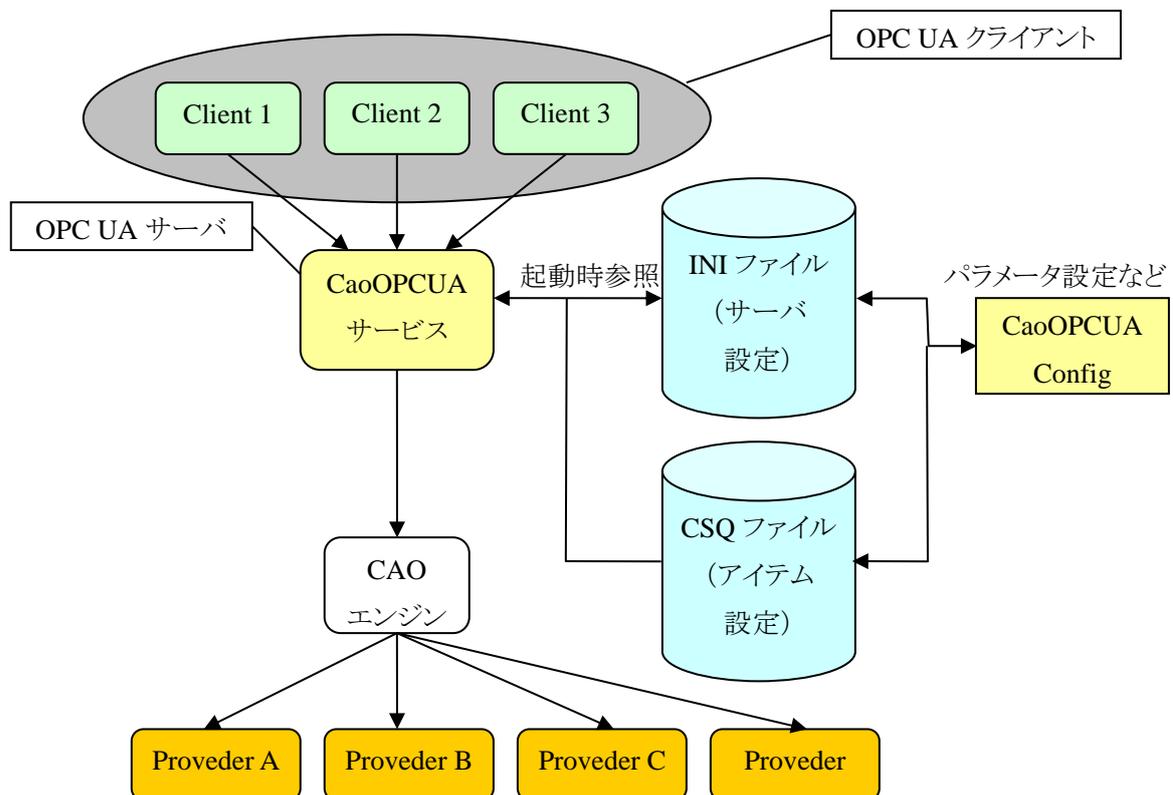


図 2-1 CaoOPCUA 概要図

2.2. ノード ID

CaoOPCUA はアイテムを一意に識別する為にノード ID を使用します。

ノード ID は次の要素で構成されます。

- ネームスペースインデックス : ノード ID の名前空間を定義します。CaoOPCUA ではコントローラの名前順に、
- ・4 から開始する連番 (CaoOPCUA のバージョン 1.1.7 以下)
 - ・5 から開始する連番 (CaoOPCUA のバージョン 1.2.0 以上)
- が割り振られます。
- 識別子タイプ : 識別子のデータ型は文字列固定となります。
- 識別子 : アイテム名となります。

2.3. アクセスパス

CaoOPCUA は OPC との互換性の為、AccessPath をサポートしています。AccessPath は固定の「OPCUA.CAO」とCaoOPCUAConfig ツールで設定されるコントローラの Name (CaoOPCUA コントローラ名)¹ が用いられます。

AccessPath=OPCUA.CAO/<CaoOPCUA コントローラ名>

¹ Controller Name (Cao コントローラ名) ではないことに注意してください。

3.1.2. CaoOPCUA サーバの設定

CaoOPCUA サーバの設定を行います。

ここでは起動の為の必要最小限の設定を行います。より詳細な設定については「[4.CaoOPCUAConfig](#)」を参照してください。

- ① CaoOPCUAConfig を起動します
- ② メニューから「アクション」→「オプション」と選択して「OPC UA」タブを開きます

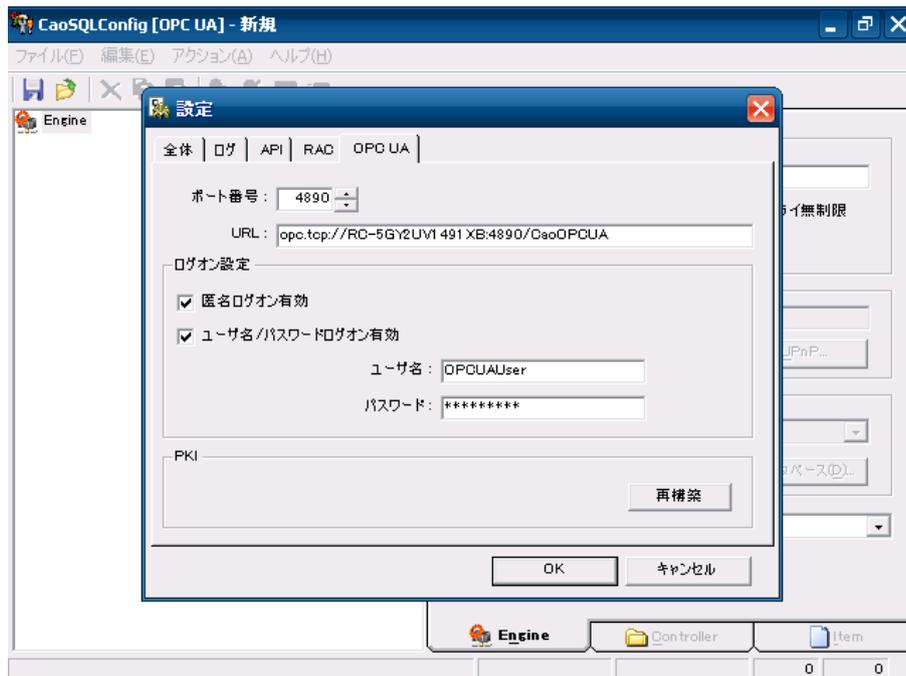


図 3-2 CaoOPCUAConfig

- ③ PKI(公開鍵基盤)を作成します
「再構築」ボタンを押下します

- ④ 以下のようにフォルダが作成されそれぞれのフォルダに必要なファイルが作成されます

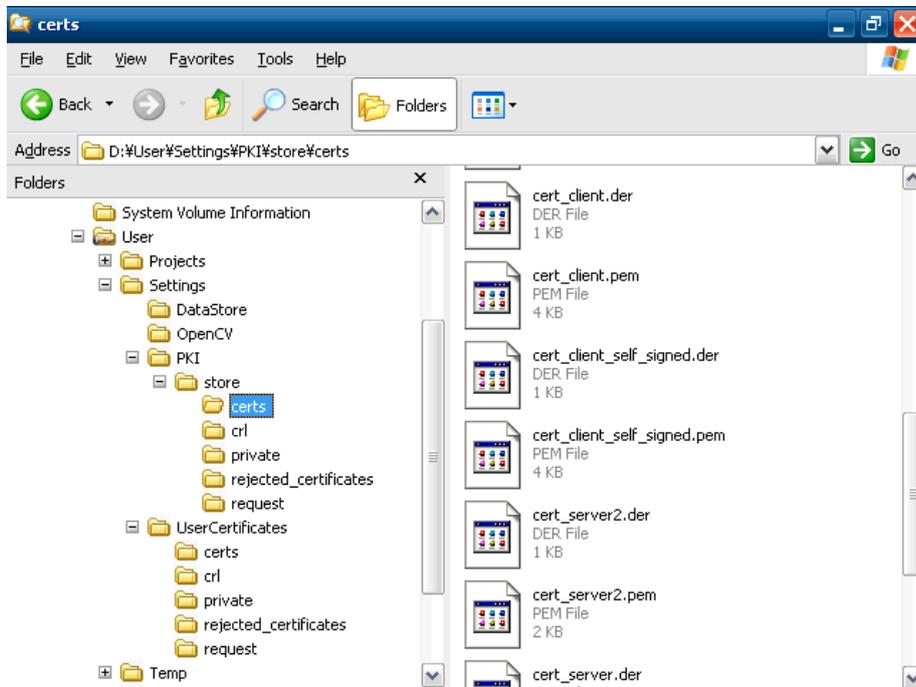


図 3-3 PKI(公開鍵基盤)フォルダ

- ⑤ 「OK」ボタンを押下し、設定ダイアログを閉じます

3.1.3. コントローラと変数の設定

- ① コントローラと変数を追加します

コントローラと変数の追加は使用する ORiN プロバイダに依存するため、説明上、以後を RC8 プロバイダが使用される前提で説明します。使用するプロバイダのユーザーズガイドを参照しながら、実際のコントローラと変数に置き換えて設定を行ってください。

- (1) メニューから「編集」→「コントローラ追加」を選択します
- (2) 任意のコントローラ名(ここでは「OPCUAServer」とします)を入力します
- (3) プロバイダ名のコンボボックスから「CaoProv.DENSO.RC8」を選択します

- (4) オプションに「Server=127.0.0.1」と入力します

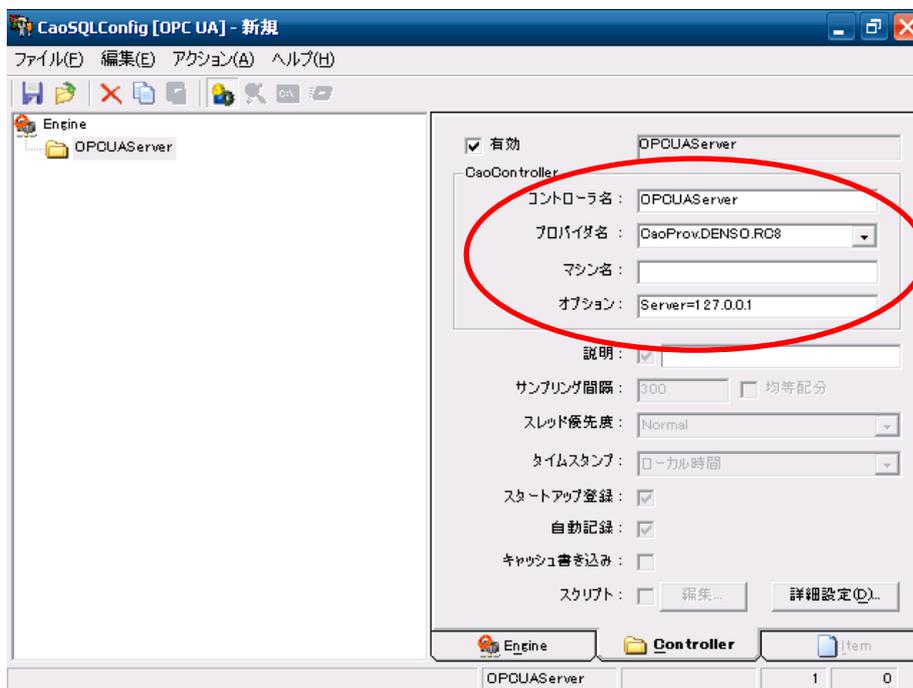


図 3-4 コントローラ設定

- (5) メニューから「編集」→「アイテム追加」を選択します

- (6) 変数名に「@CURRENT_ANGLE」を入力、クラスのコンボボックスから「Robot クラス」を選択、オブジェクト名に「Arm1」を入力し属性コンボボックスから「読み込み」を選択します

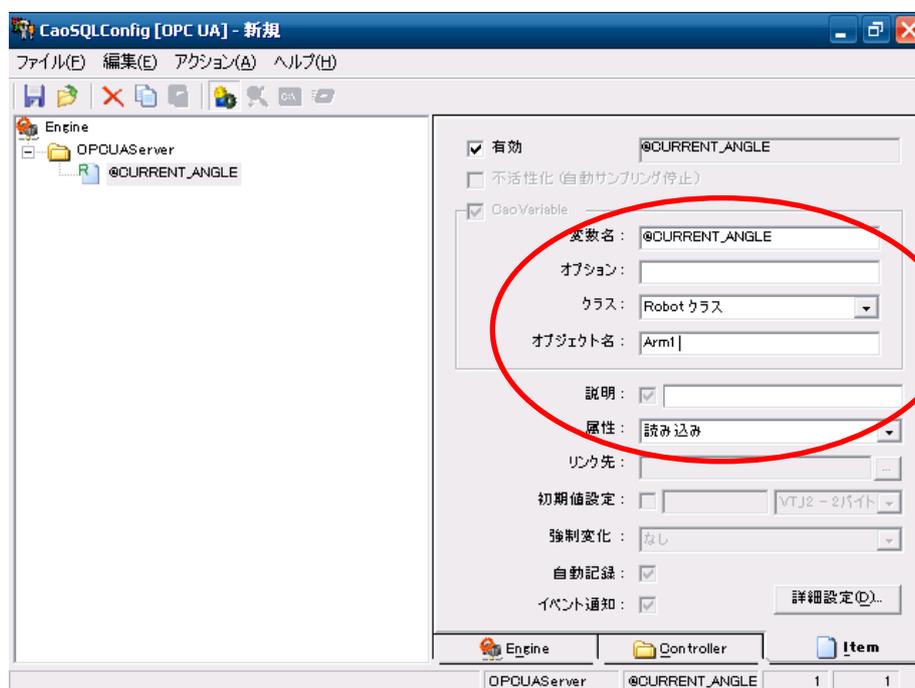


図 3-5 アイテム設定

- (7) 「詳細設定」ボタンを押下します

- (8) その他のデータの種類コンボボックスから「VT_R8 - 8バイト浮動小数点型」を選択します

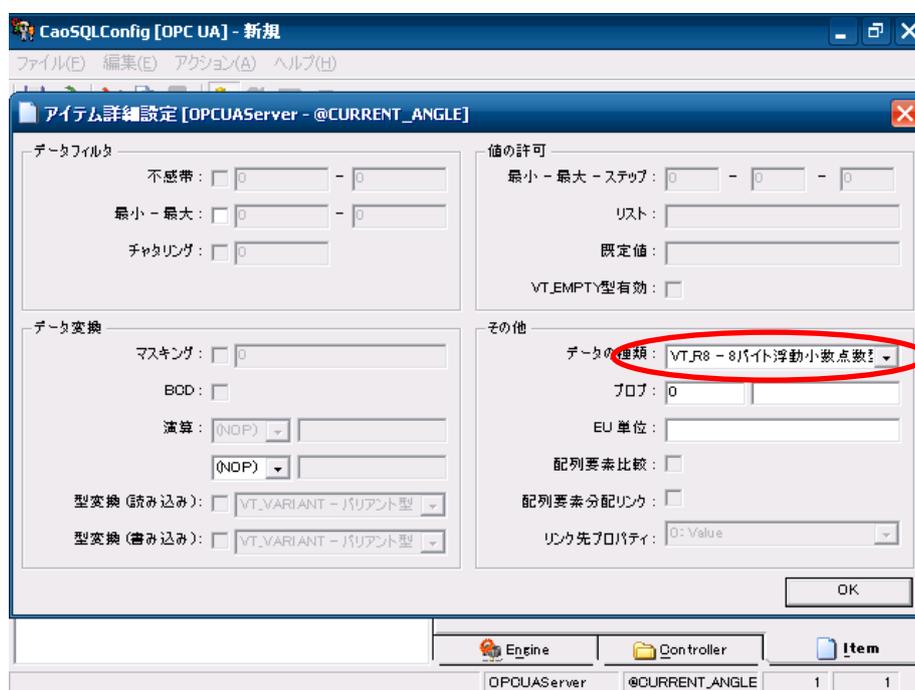


図 3-6 アイテム詳細設定

- (9) 「OK」ボタンを押下しアイテム詳細設定ダイアログを閉じます
- ② メニューから「ファイル」→「名前を付けて保存」を選択します
 - ③ 任意の場所の任意の名前（ここでは「C:\Documents and Settings\All Users\Application Data\OPCUA\CaoOPCUA.csq」とします）でこの設定を保存します
- 補足: DENSO RC8 ロボットコントローラ用にアイテム設定を行う場合はファイル名を CaoOPCUA.csq にして保存して WINCAPSIII で RC8 に送信してください。

3.1.4. CaoOPCUA サーバの登録・起動

CaoOPCUA サーバをサービスとして登録・起動します。

- ① CaoOPCUAConfig のメニューから「アクション」→「サービスとして登録」を選択します
- ② 以下のダイアログで「OK」ボタンを押下します

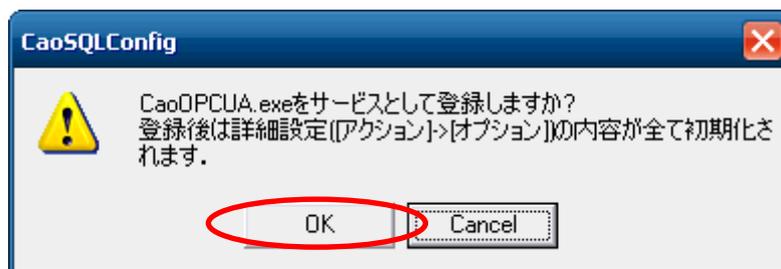


図 3-7 サービス確認ダイアログ

- ③ メニューから「アクション」→「サービスの開始」を選択します
- ④ CaoOPCUAConfig を終了します

3.2. OPC UA クライアントからの CaoOPCUA サーバへの接続

一般的な OPC UA クライアントから CaoOPCUA サーバに接続します。

3.2.1. Softing 社「UA Demo Client」

ここでは Softing 社製の「UA Demo Client」を使用して説明します

- ① 「UA Demo Client」を起動します
- ② URL 入力欄に「opc.tcp://[コンピュータ名]:[ポート番号]/CaoOPCUA」の書式で URL を入力し「Connect」ボタンを押下します

例) opc.tcp://192.168.100.100:4890/CaoOPCUA

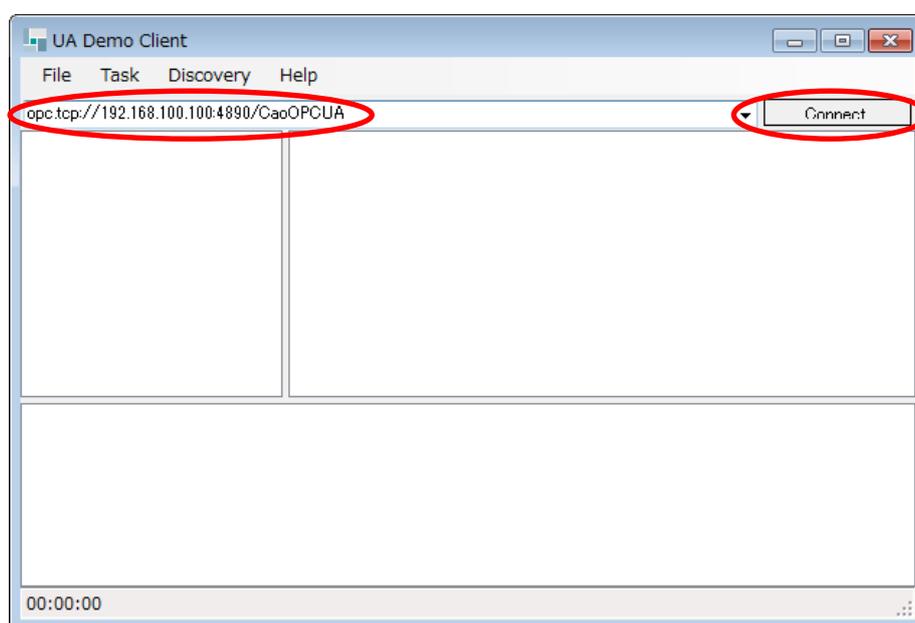


図 3-8 URL 入力→接続

- ③ 以下のダイアログが表示されるので「Security Mode」「Security Policy」のコンボボックスで「None」を選択し「OK」ボタンを押下しダイアログを閉じます

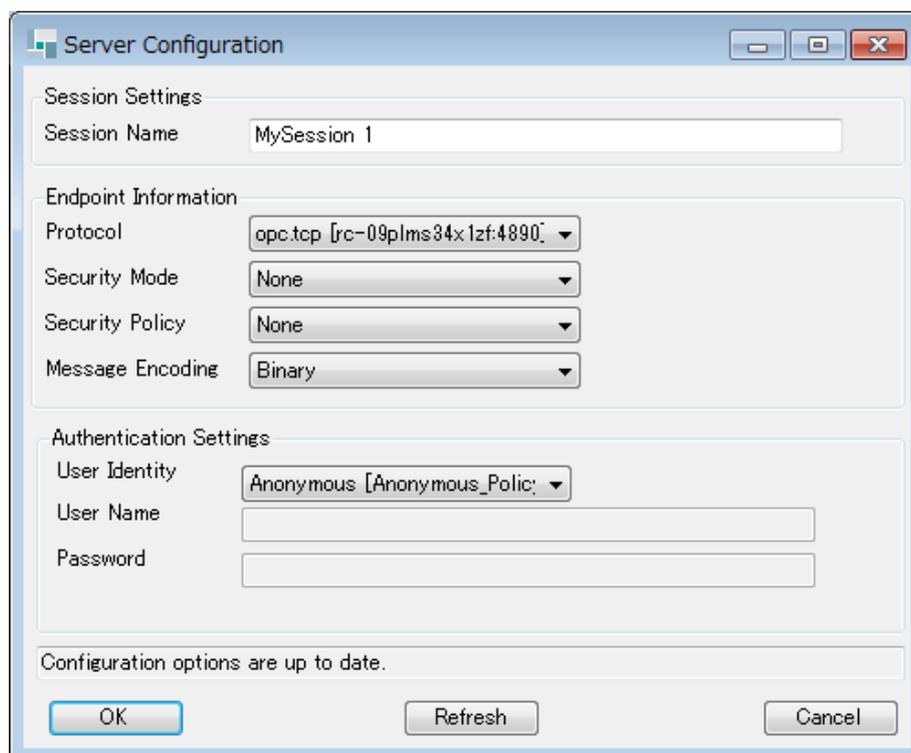


図 3-9 サーバ設定

- ④ 以下のダイアログが表示された場合、「はい」を押下してダイアログを閉じてください
- ※この警告はサーバアプリケーションの証明書がクライアントアプリケーションの証明書信頼リストに存在しない場合に発生します。この警告を出さないようにするにはサーバアプリケーションの証明書をクライアントアプリケーションの証明書信頼リストに追加してください。

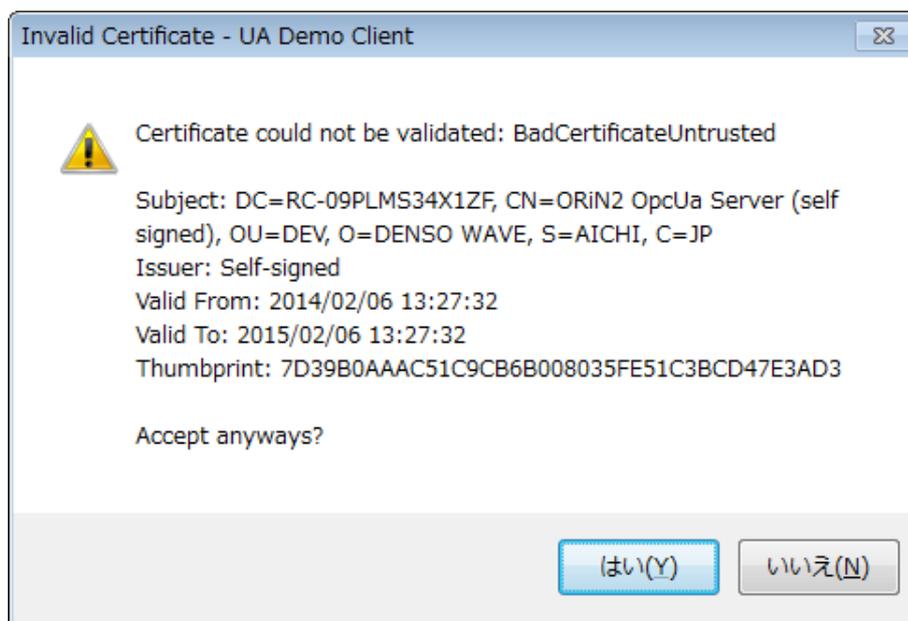


図 3-10 証明書警告

- ⑤ 正しく接続できると以下のような表示となります

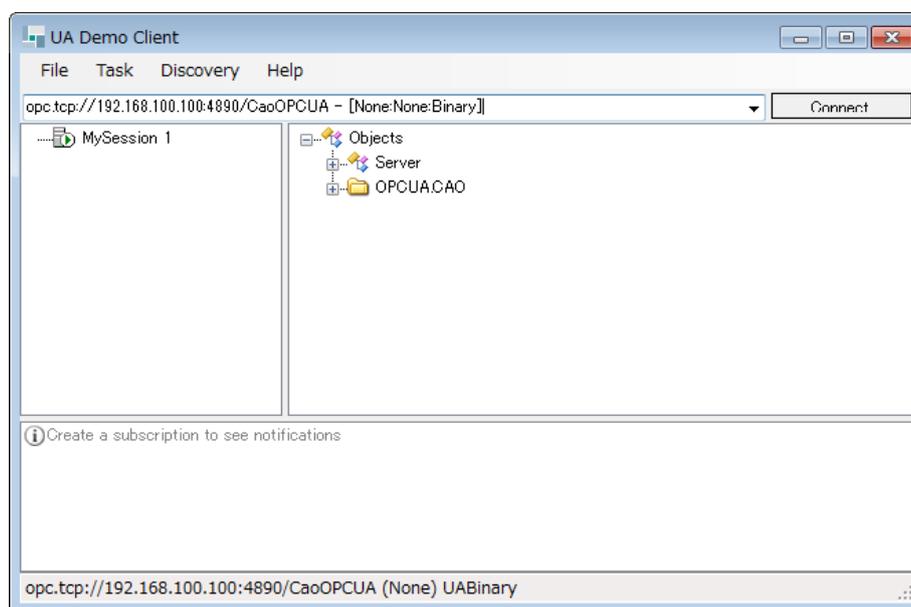


図 3-11 接続完了

- ⑥ 「OPCUA.CAO」→「OPCUAServer」と展開し「@CURRENT_ANGLE」を右クリックし「Subscribe」→「New...」と選択する
- ⑦ 表示されたダイアログで「OK」ボタンを押下する

- ⑧ 以下の画面が表示され変数「@CURRENT_ANGLE」の値が確認できます

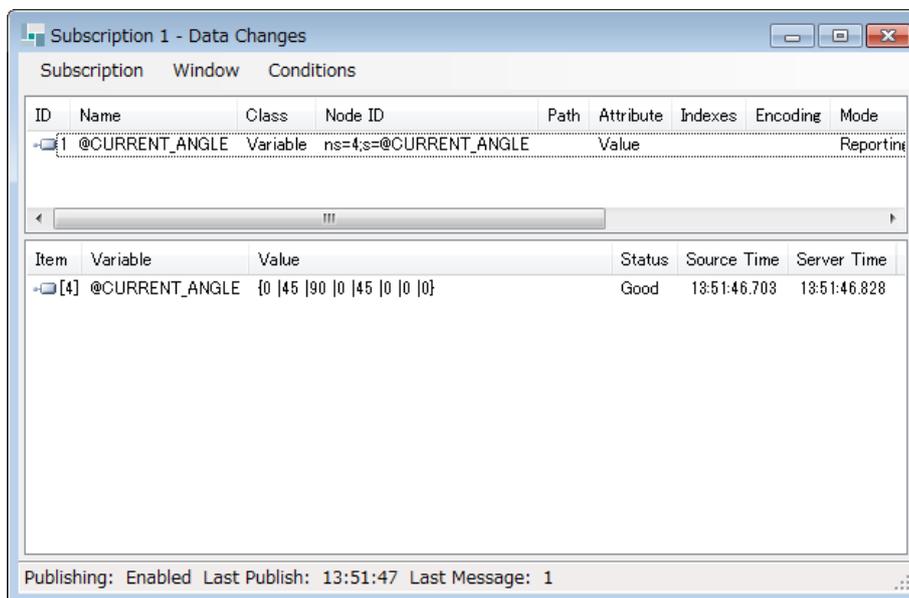


図 3-12 変数値

3.2.2. Unified Automation 社「UaExpert」

次に Unified Automation 社製の「UaExpert」を使用して説明します。

- ① 「UaExpert」を起動します
- ② メニューから「Server」→「Add」を選択します
- ③ 「<Double click to Add Server...>」をダブルクリックします

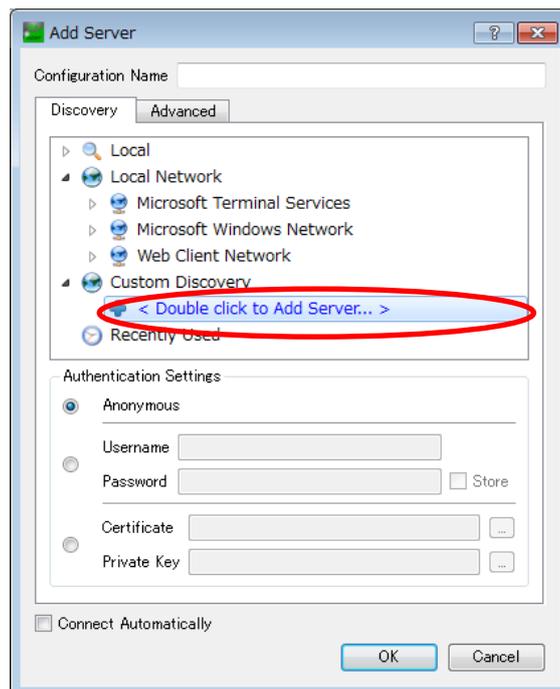


図 3-13 サーバ追加

- ④ 表示されたダイアログに「opc.tcp://[コンピュータ名]:[ポート番号]/CaoOPCUA」の書式で URL を入力し「OK」ボタンを押下しダイアログを閉じます

例) opc.tcp://192.168.100.100:4890/CaoOPCUA

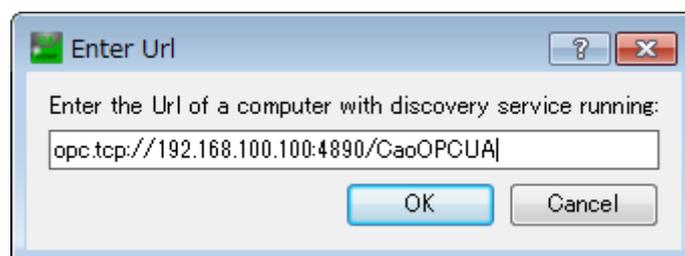


図 3-14 URL 入力

- ⑤ ④で追加したサーバが表示されているので選択・ダブルクリックし「ORiN2 OpcUa Server」をダブルクリック→「None - None」を選択します

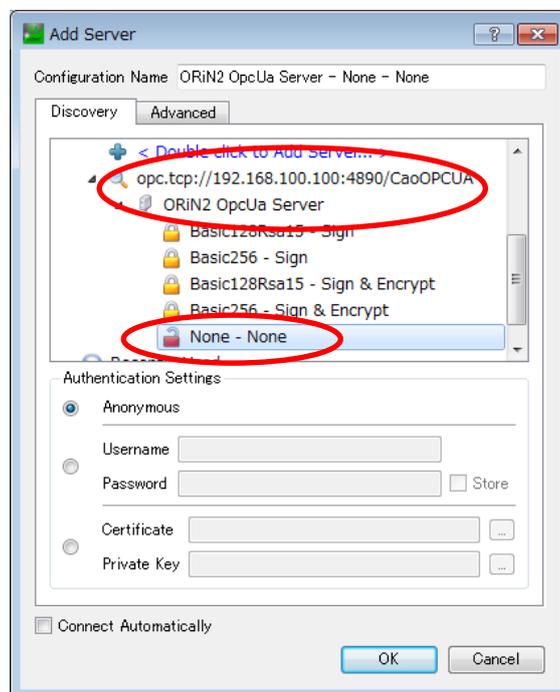


図 3-15 サーバ選択

- ⑥ 「OK」ボタンを押下し Add Server ダイアログを閉じます
- ⑦ 追加したサーバが選択されている状態でメニューから「Server」→「Connect」を選択します

⑧ 以下のダイアログが表示された場合、何れかを選択し「OK」ボタンを押下します

Accept this certificate permanently – 今後この証明書を受け入れます

Accept this certificate temporarily for this session – 今回だけ一時的にこの証明書を受け入れます

※この警告はサーバアプリケーションの証明書がクライアントアプリケーションの証明書信頼リストに存在しない場合に発生します。この警告を出さないようにするにはサーバアプリケーションの証明書をクライアントアプリケーションの証明書信頼リストに追加してください。



図 3-16 証明書警告

⑨ 「Address Space」から「OPCUA.CAO」→「OPCUAServer」と選択します

⑩ 「@CURRENT_ANGLE」を「Default DA View」にドラッグアンドドロップします

⑪ 変数「@CURRENT_ANGLE」の値が確認できます

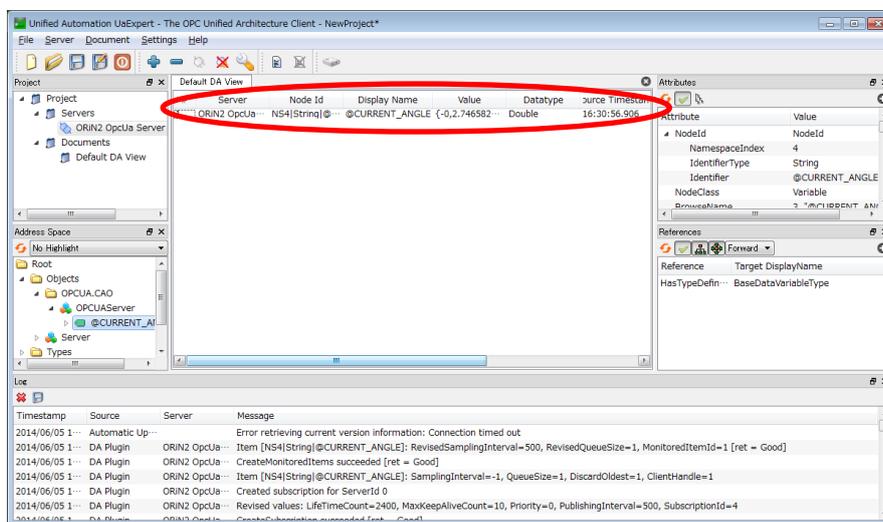


図 3-17 UaExpert による接続

3.3. CaoOPCUA 詳細

3.3.1. アプリケーション証明書

CaoOPCUA サーバが起動するためにはアプリケーションの証明書とプライベートキーファイルが存在する必要があります。これらは設定保存先の「¥PKI¥store」以下の「certs」フォルダ、「private」フォルダの中にあります。

「certs」フォルダ中の「cert_server_self_signed.der」が証明書ファイルです。

「private」フォルダ中の「private_key_server_self_signed.pem」がプライベートキーファイルです。

正しいこれらのファイルが存在しない場合、サービスの起動は失敗します。

3.3.2. セキュリティポリシーとセキュリティモード

CaoOPCUA ではセキュリティの要件に応じてセキュリティポリシーとセキュリティモードの組み合わせを定義し使い分けることが可能になっています。

3.3.2.1. セキュリティポリシー

セキュリティポリシーには以下の設定があります。

表 3-1 セキュリティポリシー

None	セキュリティなし。
Basic128Rsa15	非推奨。この機能は互換性のために残されています。 2
Basic256	非推奨。この機能は互換性のために残されています。 2
Basic256Sha256	平均的なセキュリティ。
Aes128Sha256RsaOaep	高いセキュリティ。
Aes256Sha256RsaPss	最高のセキュリティ。

² 暗号化アルゴリズムに脆弱性が発見されています。使用は避けてください。

3.3.2.2. セキュリティモード

セキュリティモードには以下の設定があります。

表 3-2 セキュリティモード

None	セキュリティなし
Sign	メッセージに署名は付けるが、暗号化はしない
SignAndEncrypt	メッセージに署名を付け、かつ暗号化する

3.3.2.3. 設定可能なセキュリティの組み合わせ

CaoOPCUA で設定可能なセキュリティポリシーとセキュリティモードの組み合わせは以下の通りです。

表 3-3 設定可能なセキュリティの組み合わせ

セキュリティポリシー	セキュリティモード
None	None
Basic128Rsa15	Sign
Basic128Rsa15	SignAndEncrypt
Basic256	Sign
Basic256	SignAndEncrypt
Basic256Sha256	Sign
Basic256Sha256	SignAndEncrypt
Aes128Sha256RsaOaep	Sign
Aes128Sha256RsaOaep	SignAndEncrypt
Aes256Sha256RsaPss	Sign
Aes256Sha256RsaPss	SignAndEncrypt

3.3.2.4. セキュリティ使用の接続時に必要な証明書

CaoOPCUA ではセキュリティを使用した接続時 (None/None 以外) に証明書による認証が必要です。これを行うにはクライアントアプリケーションの証明書ファイルをCaoOPCUA サーバのPKI (公開鍵基盤) ストア (設定保存先の「¥PKI¥store」フォルダ) の「cert」フォルダに保管することで実現します。

以上でセキュリティを使用した接続が可能となります。接続設定については各 OPC UA クライアントアプリケーションのマニュアルなどを参照してください。

3.3.3. ユーザ認証と権限

CaoOPCUA ではユーザ認証を行いその情報から接続ユーザに適切な権限を付与します。ユーザ認証には以下のものがあります。

表 3-4 ユーザ認証と権限

匿名認証	匿名ユーザによる認証	変数: 読み取り専用
ユーザ名による認証	ユーザ名とパスワードによる認証	変数: 読み書き
証明書による認証	証明書とプライベートキーによる認証	変数: 読み書き

3.3.3.1. 匿名 (Anonymous) 認証

匿名 (Anonymous) ユーザを使用した認証です。CaoOPCUA では匿名認証の場合、変数は読み取り専用となります。この認証を行うには特に設定の必要はありません。

3.3.3.2. ユーザ名認証

ユーザ名とパスワードを使用した認証です。CaoOPCUA ではユーザ名認証の場合、変数は読み書き可能となります。使用できるユーザ名はサーバの設定で行ったものとなります。接続設定については各 OPC UA クライアントアプリケーションのマニュアルなどを参照してください。

3.3.3.3. 証明書認証

証明書とプライベートキーを使用した認証です。CaoOPCUA では証明書認証の場合、変数は読み書き可能となります。

3.3.3.3.1. ユーザ証明書

証明書認証を行うためにはユーザ証明書が必要です。

以下の手順を行います。

- ① クライアントの PC で x.509 証明書とプライベートキーの各ファイルを用意します
- ② CaoOPCUA サーバのユーザ証明書ストア(設定保存先の「¥UserCertificate¥certs」)に x.509 証明書ファイルを保管します
- ③ OPC UA クライアントアプリケーションで証明書認証を行って接続します
接続設定については各 OPC UA クライアントアプリケーションのマニュアルなどを参照してください

4. CaoOPCUAConfig

4.1. 概要

CaoSQLConfig の引数に“OPCUA”を付けて実行(“CaoSQLComfig.exe OPCUA”)することで CaoOPCUA 用 CaoSQLConfig(以下 CaoOPCUAConfig)を起動します。CaoOPCUAConfig は、マシン内の CaoOPCUA の動作パラメータ設定や、CAO のコントローラオブジェクト、CAO の変数オブジェクトを作成する際に必要なパラメータを設定するツールです。ここで設定した情報は INI ファイルと csq ファイルに記録され、CaoOPCUA の起動時に読み込まれます。

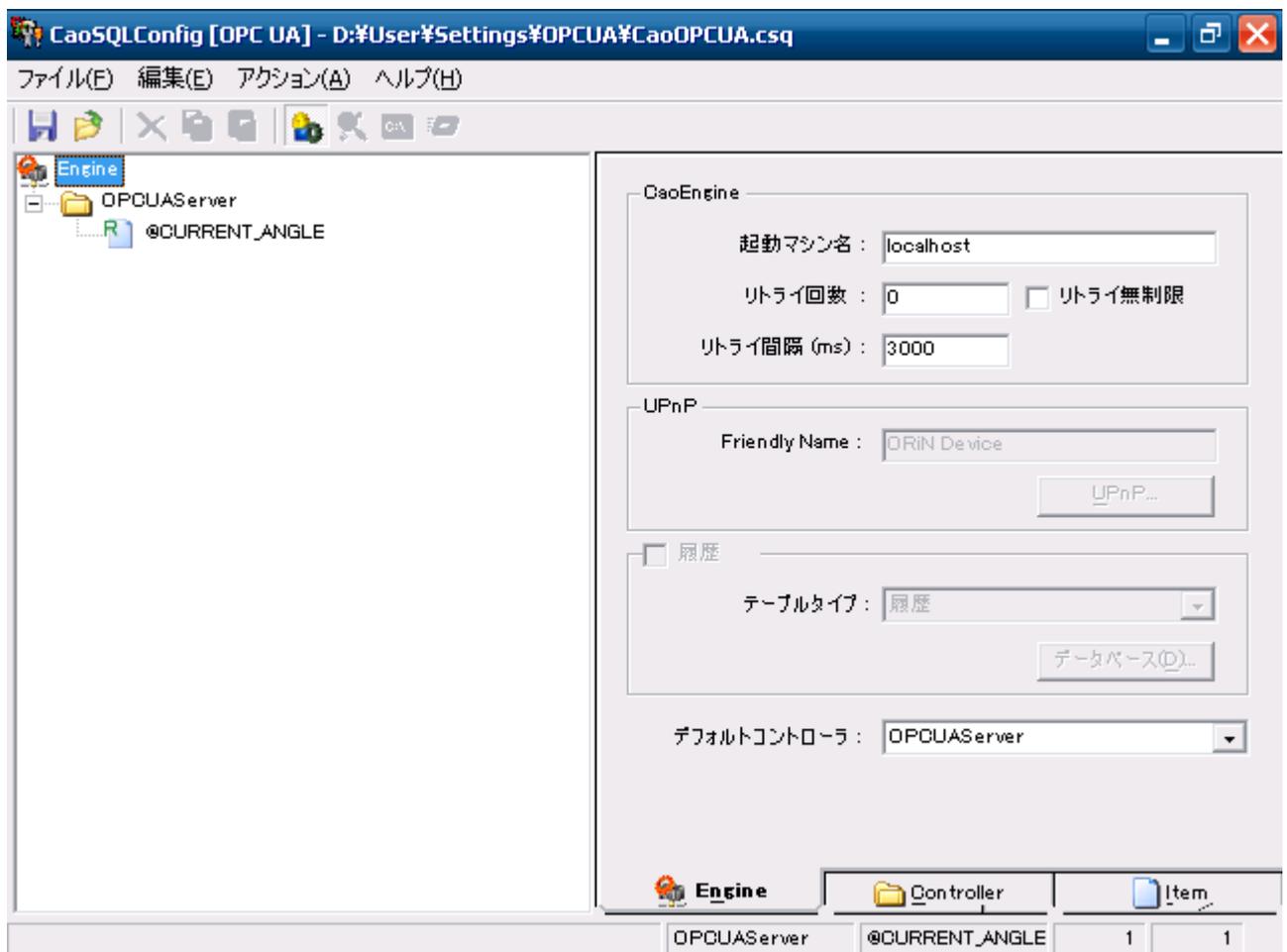


図 4-1 CaoOPCUAConfig 画面

4.2. 操作方法

4.2.1. タブ入力

画面右のタブはツリービューのノードを選択状態にすることによってその情報を示します。エンジンのノー

ドを選択するとタブはエンジンの情報を表示します。コントローラのノードを選択するとタブはそのコントローラの情報を表示します。アイテムのノードを選択するとタブはそのアイテムの情報を表示します。

注意することとして、入力されたデータの整合性のチェックはおこなっていないため、実際に CaoOPCUA を起動した時にエラーが発生し、CaoOPCUA のリストから外れてしまう可能性がありますので CaoOPCUAConfig でデータを入力する時はデータの確認を慎重に行う必要があります。

4.2.1.1. エンジンタブ

エンジンタブでは図 3-2 に示した形式で情報を設定・表示します。エンジンの情報は自由に設定できます。CaoOPCUA ワークスペース名はデフォルトの名前が適応され、またデフォルトコントローラ名に関しては現在存在しているコントローラ名のみ選択ができます。



図 4-2 エンジンタブ

ここで設定可能な項目は以下の通りです。

[起動マシン名] – 起動マシン名

CAO.exe の起動コンピュータ名を入力します。何も入力されていない場合はローカル起動します。

[リトライ回数, リトライ無制限] – リトライ回数, リトライ無制限

チェックボックスを入れると CaoController のオープンで失敗した場合、リトライすることができます。リトラ

イ回数は無制限をチェックする, 又は回数を入力します.

[リトライ間隔 – リトライ間隔(ms)]

リトライ間隔は一度失敗してから, リトライをするまでの間隔をミリ秒単位で入力します.

[デフォルトコントローラ] – デフォルトコントローラ

未対応です.

4.2.1.2. コントローラタブ

コントローラタブでは図 4-3 に示した形式で情報を設定・表示します.

図 4-3 コントローラタブ

コントローラの情報には自由に設定できますが、「CaoOPCUA コントローラ名」だけは登録された情報の中でユニークな情報を持つことが要求される為、メニューの名前変更でしか変更できません。またコントローラ名に関しては大文字と小文字の区別はされず、数値のみの名前は許可しません。また、記号の内、¥\$#は使用できません。

ここで設定可能な項目は以下の通りです.

[コントローラ有効] – 有効

このコントローラの有効/無効設定です。無効に設定された場合は CaoOPCUA 実行時に起動されません。

[CaoOPCUA コントローラ名]

CaoOPCUA コントローラ名を設定します。

[Cao コントローラ名, プロバイダ名, コンピュータ名, オプション]

– コントローラ名, プロバイダ名, マシン名, オプション

AddController をおこなう際に渡すパラメータです。入力した値がそのまま使用されます。

[詳細情報] – 説明

未対応です。

4.2.1.3. アイテムタブ

アイテムタブでは図 4-4 に示すような情報を設定・表示する。コントローラタブと同様にアイテム名のみはメニューの名前変更でないと変更は行えません。またアイテム名に関しては大文字と小文字の区別はされず、数値のみの名前は許可しません。また、記号の内、¥\$#! は使用できません。

図 4-4 アイテムタブ

ここで設定できる項目は以下の通りです。

[アイテム有効] – 有効

このアイテムを有効/無効にする設定です。無効に設定された場合はサンプリングされません。

[アイテム名] – 名前

アイテム名を設定します。このアイテム名は ItemID として使用されます。

[変数名] – 変数名

CAO の変数オブジェクト名を指定します。大文字・小文字は区別されます。CaoVariable オブジェクトの取得(ICaoController::GetVariable)時に使用されます。

[オプション] – オプション

CaoVariable オブジェクトの作成時に付加したいオプションを指定します。CaoVariable オブジェクトの取得(ICaoController::GetVariable)時に使用されます。

[クラス] – クラス

変数クラスの種類を選びます。CaoVariable オブジェクトの取得(ICaoController::GetVariable)時に使用されます。

[属性] – 属性

アイテムの属性を設定します。

配列型の値を扱うアイテムは、デバイスから取得した実際の値の情報から配列の要素数を決めています。このため、アイテムの属性に「書き込み」を設定すると、値を取得しない動作になるため、配列型とはみなさず単一型アイテムとして扱う仕様となっています。

配列型アイテムの値を書き換える必要がある場合は属性に「読み込み/書き込み」を設定する必要があります。

[初期化設定] – 値の初期値設定

アイテムの値に初期値を設定します。実機に対してはほとんど必要ありませんが、コントローラに DataStore プロバイダなどを設定している場合に利用するといいいでしょう。CaoVariable オブジェクトの取得(ICaoController::GetVariable)時に[属性]が書き込み可能なアイテムに対してのみ設定されます。

[詳細情報] – 説明

未対応です。

[アイテム詳細設定] – 詳細設定

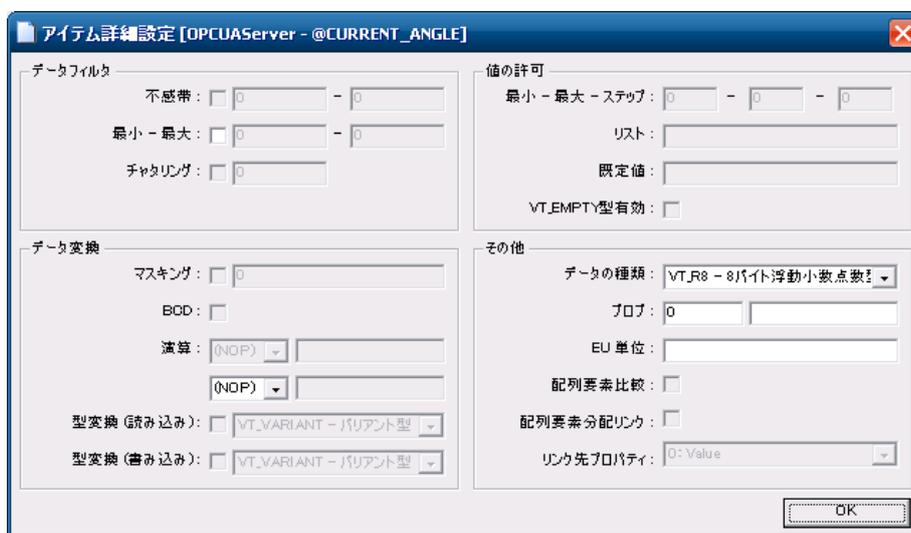


図 4-5 アイテム詳細設定ダイアログ

[Low EU, High EU 設定] – 最小-最大

未対応です。

[データタイプ設定] – データの種類

アイテムのネイティブ型(Canonical DataType)を設定します。設定可能な型は表 4-1 の通りです。

表 4-1 アイテムのデータタイプ

データ型	バイト数	説明
VT_BOOL	2	-1:True, 0:False
VT_I1	1	符号付整数
VT_UI1	1	符号なし整数
VT_I2	2	単精度(16ビット)符号付整数
VT_UI2	2	単精度(16ビット)符号なし整数
VT_I4	4	倍精度(32ビット)符号付整数
VT_UI4	4	倍精度(32ビット)符号なし整数
VT_R4	4	単精度(32ビット)浮動小数
VT_R8	8	倍精度(64ビット)浮動小数
VT_CY	8	通貨型 小数点以下 4 桁の固定小数点の数値で、スケールが 10000 倍の 8 バイト符号付き整数で格納
VT_BSTR	可変	文字列 UNICODE 文字と NULL ターミネータ
VT_DATE	8	VT_R8 と同じ 1899/12/30 からの通算日時
VT_VARIANT	可変	VARIANT 型

[BLOB サイズ, BLOB 設定] – プロブ

未対応です。

[EU Unit 設定] – EU 単位

未対応です。

4.2.2. メニュー

CaoOPCUAConfig における基本的な操作はメニューバーから選択します。一部ツリービューを右クリックして選択できるメニューもありますが、メニューバーから選択できる機能と全く同等です。

4.2.2.1. ファイルメニュー

作成したデータのセーブやロード等の処理を行います。

開く,保存,名前をつけて保存..で使用されるファイルの拡張子は“**csq**”です。

インポート,エクスポートで使用されるファイルの拡張子は“**csx**”です。

CSV ファイルからインポート, CSV ファイルにエクスポートで使用されるファイルの拡張子は“**csv**”です。

[新規作成] – 新規

新規に csq ファイルを作成します。

[ファイルから読み込み] – 開く...

指定した csq ファイルから設定情報を読み込みます³。

読み込みをキャンセルした場合には読み込み結果は反映されません。

[ファイル保存] – 保存

現在の設定内容を csq ファイルに上書き保存します。

[名前を付けて保存] – 名前をつけて保存...

現在の設定内容を csq ファイルに保存します。

[インポート] – インポート...

csx ファイルからノードを選択しているノードに追加します。

[エクスポート] – エクスポート...

選択しているノードを csx ファイルに出力します。

[CSV ファイルからインポート] – CSV ファイルからインポート...

csv ファイルからノードを選択しているノードに追加します。

[CSV ファイルにエクスポート] – CSV ファイルにエクスポート...

選択しているノードを csv ファイルに出力します。

³ Win9x/Me では OS の制限から 64K 以上のサイズのファイルの読み込みの場合はデータに欠損が生じる場合があるので注意して下さい。

[終了] – 終了

CaoOPCUAConfig のプログラムを終了します。

4.2.2.2. 編集メニュー

アイテムの追加削除などの編集処理を行います。

[コントローラ追加] – コントローラ追加

CaoOPCUA で読み込むコントローラの新規追加を行います。既存のコントローラと同じ名前のコントローラは追加できません(大文字・小文字の区別をしません)。有効な名前が入力されるとツリーに追加され、その詳細情報は画面右のタブから入力します。

[アイテム追加] – アイテム追加

ツリービューで選択されたノードにアイテムを追加します。選択されたコントローラに既に登録されているアイテムと同じ名前のアイテムは追加できません(大文字・小文字の区別をしません)。有効な名前が入力されるとツリーに追加され、詳細情報は画面右のタブから入力します。

[名前変更] – 名前の変更

ツリービューで選択されたコントローラ/アイテムの名前を変更します。すでに登録されているコントローラ/アイテムの名前に変更はできません(大文字・小文字の区別をしません)。

[削除] – 削除

ツリービューで選択されたノードを削除します。コントローラの場合はコントローラに登録されたアイテムを再帰的に全て削除します。ツリーでアイテムが選択された状態でキーボードから Del キーを押下することによっても同等の処理が行えます。

[ノードのコピー] – コピー

ツリーで選択されたノードをコピーします。コントローラの場合は登録されたアイテムもコピーされます。但し、トリガ情報はコピーされません。下記の“貼り付け”操作でコピーした情報のクローンをノードに追加できます。

[貼り付け] – 貼り付け

コピーされた情報を新しい名前を入力してノード貼り付けます。

[ソート] – ソート

選択されているレベルのノードを昇順ソートします。

4.2.2.3. アクションメニュー

CaoOPCUA.exe の実行や実行時の設定に関する設定を行います。設定は INI ファイルに保存されます。

[オプション設定] – オプション

CaoOPCUA のオプション設定ダイアログが表示されます。

[全体タブ] – 全体

プロセスの優先度や使用言語など、CaoOPCUA 全般に関する設定を行うことができます。



図 4-6 CaoOPCUA 環境設定(全体)

[プロセス優先度] – プロセス優先度

CaoOPCUA のプロセス優先度を設定します。優先度に対する調整は以下の通りです。

リアルタイム > 高 > **通常** > アイドル

[ロケール ID] – ロケール ID

使用する言語 ID を設定します。

[ログ設定タブ] – ログ

CaoOPCUA のログ出力に関する設定を行うことができます。

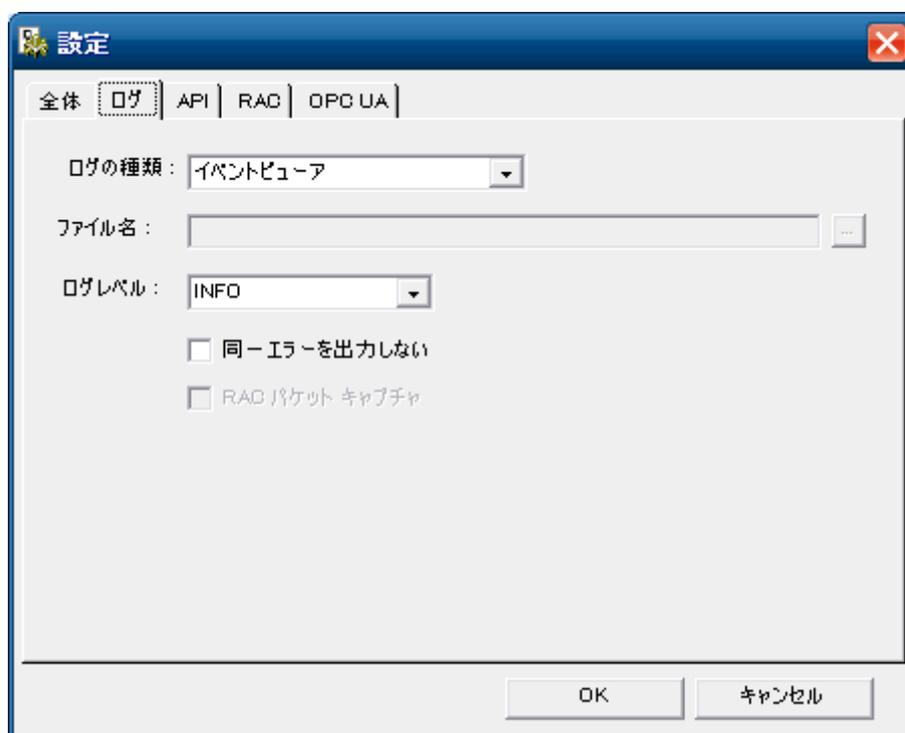


図 4-7 CaoOPCUA 環境設定(ログ)

[ログタイプ] – ログの種類

CaoOPCUA.exe のログの出力タイプを選択します。

ログの出力タイプは以下のものを選択することができます。

表 4-2 ログタイプ

出力先	備考
コンソール	コンソールに出力します
メッセージボックス	メッセージボックスに出力します(サービス起動時)
イベント ビューア	イベントビューアに出力します(サービス起動時)
デバッグ ビューア	デバッグ出力します。
テキストファイル	指定したテキストファイルに出力します。

[ファイル名] – ファイル名

Log Type を Text File にした場合にログを出力するファイルパスを設定します。

[ログ出力レベル] – ログレベル

ログの出力レベルを設定します。ログレベルの設定は、以下の 5 つのレベルから選択することができます。“FATAL”がもっとも深刻度が高く、“DEBUG”に近づくほど深刻度は低くなります。標準では“INFO”に設定されています。

FATAL > ERROR > WARN > **INFO** > DEBUG

[同エラー出力抑制] – 同一エラーを出力しない

同じエラーの出力抑制設定を行います。チェックを入れると、同じエラーが何度発生しても 1 度しかエラーメッセージを出力しません。

[API 設定タブ] – API

設定できる項目はありません。

[RAC 機能設定タブ] – RAC

設定できる項目はありません。

[OPC UA 設定タブ] – OPC UA

CaoOPCUA の設定に関する設定を行うことができます。

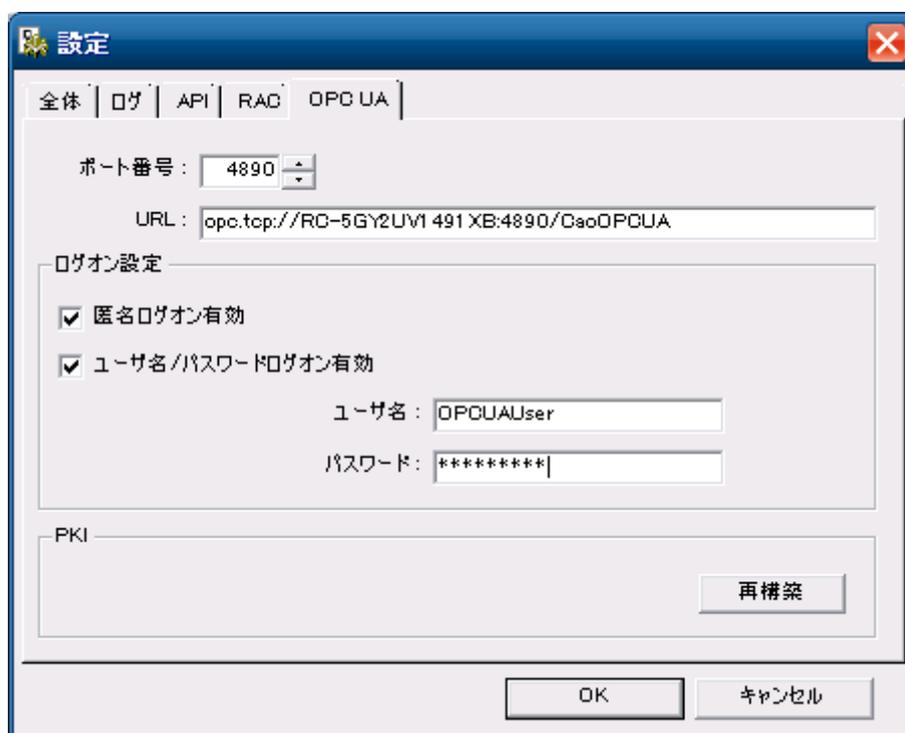


図 4-8 CaoOPCUA 環境設定 (OPC UA)

[ポート番号設定] – ポート番号

CaoOPCUA サーバが使用するポート番号を指定します。他のアプリが使用していないものを指定する必要があります。

[URL 表示] – URL

クライアントから接続時に使用する URL を表示します。設定したポート番号とコンピュータ名から自動生成されます。

[ログオン設定] – ログオン設定

ログオンに関する設定を行うことができます。

[匿名ログオン有効設定] – 匿名ログオン有効

匿名 (Anonymous) ユーザでのログオンが有効かを設定します。チェックを外すと匿名ユーザでのログオンが無効になります。

[ユーザ名／パスワードログオン有効設定] – ユーザ名／パスワードログオン有効

ユーザ名とパスワードでのログオンが有効かを設定します。チェックを外すとユーザ名とパスワードでのログオンが無効になります。

[ユーザ名] – ユーザ名

ユーザ名とパスワードでのログオンが有効な場合のユーザ名を設定します。「ユーザ名／パスワードログオン有効」にチェックが入っていないと設定できません。

[パスワード] – パスワード

ユーザ名とパスワードでのログオンが有効な場合のパスワードを設定します。「ユーザ名／パスワードログオン有効」にチェックが入っていないと設定できません。入力したパスワードは「*」で表示されます。

[PKI(公開鍵基盤)設定] – PKI

PKI(公開鍵基盤)に関する設定を行うことができます。

[PKI(公開鍵基盤)再構築] – 再構築

PKI(公開鍵基盤)に必要な各フォルダ・ファイルを作成します。

作成するパスは以下のルールで決定されます。

- ① 環境変数「CAOOPCUA⁴」が存在する場合、この値で指定されたパス
- ② ①がない場合、特殊フォルダ「CSIDL_COMMON_APP_DATA」+「¥OPCUA」のパス

4.2.2.4. ヘルプメニュー

ヘルプやライセンス登録のメニューです。

[バージョン情報]

バージョン情報を表示します。

⁴ RC8 に設定されている環境変数

4.3. INI ファイル構成情報

CaoOPCUAConfig によって設定する INI ファイルの構成を以下に示します。

CaoOPCUAConfig では、太字で示した項目を変更することができます。

INI ファイルの名称: CaoOPCUA.ini

[]: セクション名

<>: キー名

[System]			
---< ProcessPriority >	DWORD	CaoOPCUA プロセス優先度	
---< LCID >	DWORD	ロケール ID	
[Log]			
---< LogType >	DWORD	ログタイプ設定	
---< FileName >	String	ログタイプがファイルの場合のパス名	
---< LogLevel >	DWORD	ログレベル設定	
---< NotRepeatError >	DWORD	エラーログの繰り返し設定	
[Server]			
---< ServerPort >	DWORD	OPC UA サーバのポート番号	
---< AutoStart >	DWORD	OPC UA サーバの有効/無効スイッチ	
[Logon]			
---< AnonymousLogon >	DWORD	OPC UA サーバの匿名ユーザログオンの有効/無効スイッチ	
---< UserLogon >	DWORD	OPC UA サーバのユーザ名/パスワードログオンの有効/無効スイッチ	
---< UserName >	String	OPC UA サーバのユーザ名	
---< UserPassword >	String	OPC UA サーバのパスワード	
[Path]			
---< CSQPath >	String	csq ファイルのパス名	

5. OPC UA プロバイダ

5.1. 概要

本章では CAO から OPC UA(OLE for Process Control Unified Architecture)サーバを介して、PLC(Programmable Logic Controller)にアクセスする手段を与える「OPC UA 接続用の CAO プロバイダ」(以下、OPC UA プロバイダ)について簡単に解説をおこないます。OPC UA プロバイダの詳細については、「[OPC UA プロバイダユーザーズガイド](#)」を参照してください。

このプロバイダにより、CAO 対応ツールは、ロボットのみならず OPC UA サーバを持つ PLC や表示盤に対してもロボットと同様にアクセスすることができます。

この OPC UA プロバイダは、OPC UA サーバが保持する Item の値を CaoVariable オブジェクトから参照することを可能としています。

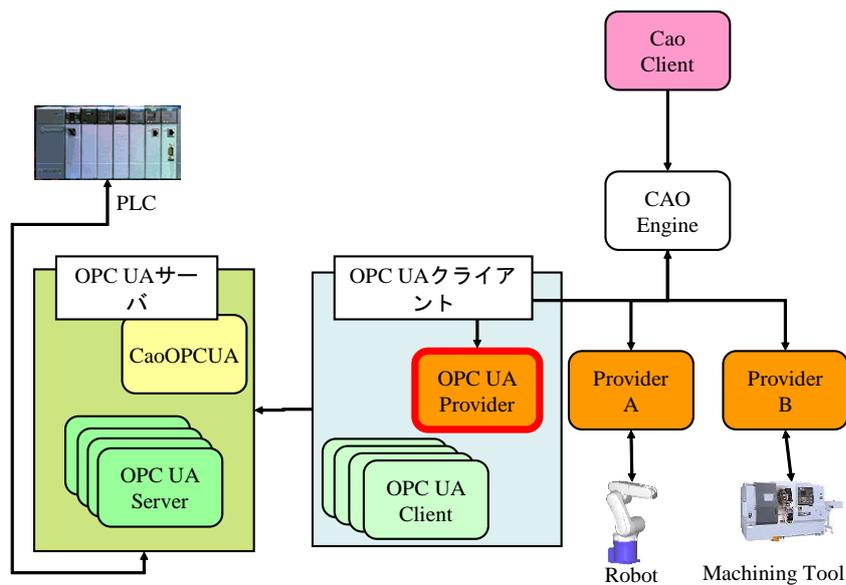


図 5-1 OPC UA 接続用の CAO プロバイダ

5.2. メソッド・プロパティ

5.2.1. CaoWorkspace::AddController メソッド

CAO の CaoWorkspace クラスの AddController メソッドの仕様を下記に示します。

```
AddController
(
    "<Controller 名>",           // コントローラ名
    "CaoProv. OPCUA",           // 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 (OPC UA オプション)
)
```

ここで、<プロバイダ名>は固定、<実行マシン名>は他のプロバイダと同じ意味です。
<オプション>の書式は「[OPC UA プロバイダユーザーズガイド](#)」を参照してください。

以下にサンプルを示します。

```
AddController
(
    "OPCUAServer",
    "CaoProv. OPCUA",           // プロバイダ名は固定.
    "",                         // localhost で起動
    "Server=opc.tcp://10.6.235.64:4890/CaoOPCUA, AccessPath=OPCUA.CAO/OPCUAServer, Security=0:0"
);
```

5.2.2. CaoController::AddVariable メソッド

CAO の CaoController::AddVariable メソッドの仕様を下記に示します。

```
CaoController::AddVariable
(
    "<変数名>",                 // アイテム ID.
    "<オプション>"              // オプション文字列 (OPC UA オプション)
)
```

OPC UA プロバイダを使う場合は、この引数を次のように設定します。

<変数名> ::= <アイテム ID>

<オプション> ::= [[AccessPath=<アクセスパス>]]

<オプション>の書式は「[OPC UA プロバイダユーザーズガイド](#)」を参照してください。

以下にサンプルを示します。

```
AddVariable
(
    "@CURRENT_ANGLE",         // アイテム ID が@CURRENT_ANGLE
    ""
);
```

Appendix A. RC8 を使用した CaoOPCUA サーバ

DENSO RC8 ロボットコントローラは、本サービスの CaoOPCUA が組み込まれています。このサービスを有効化するには別途 RC8 用 OPC UA サーバライセンスをお求めください。RC8 毎に1ライセンス必要です。

以下に RC8 で CaoOPCUA サーバをセットアップする手順を示します。

Appendix A.1. CaoOPCUA サーバの起動

RC8 では以下の方法で CaoOPCUA サーバの起動を行います。

1. ライセンス登録

1-1 ティーチングペンダントを起動し、「設定」→「オプション」→「機能拡張」と操作します

1-2 「追加」を押下し、24 桁のライセンスキーを入力します

1-3 図 A-1 のように「OPC UA Server」の機能が有効になっていることを確認します

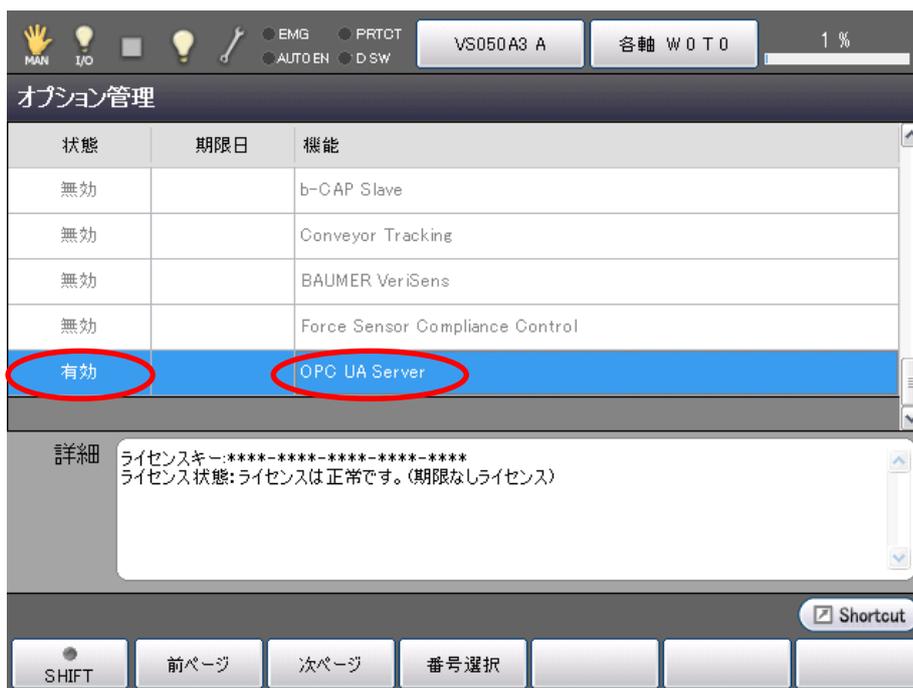


図 A-1 ライセンスキーの登録

2. OPC UA サーバ初期化設定

2-1 ティーチングペンダントを起動し、「設定」→「通信と起動権」→「OPC-UA」サーバと操作します

2-2 以下の内容で設定し、「保存」を押下します

[サーバ有効] : ON
[ポート番号] : 4890
[ログオン] : 匿名 ON, ユーザ名/パスワード ON
[ユーザ名] : 任意 (ここでは「OPCUAUser」)
[パスワード] : 任意

※詳しくは「[CaoOPCUA サーバの設定](#)」を参照してください

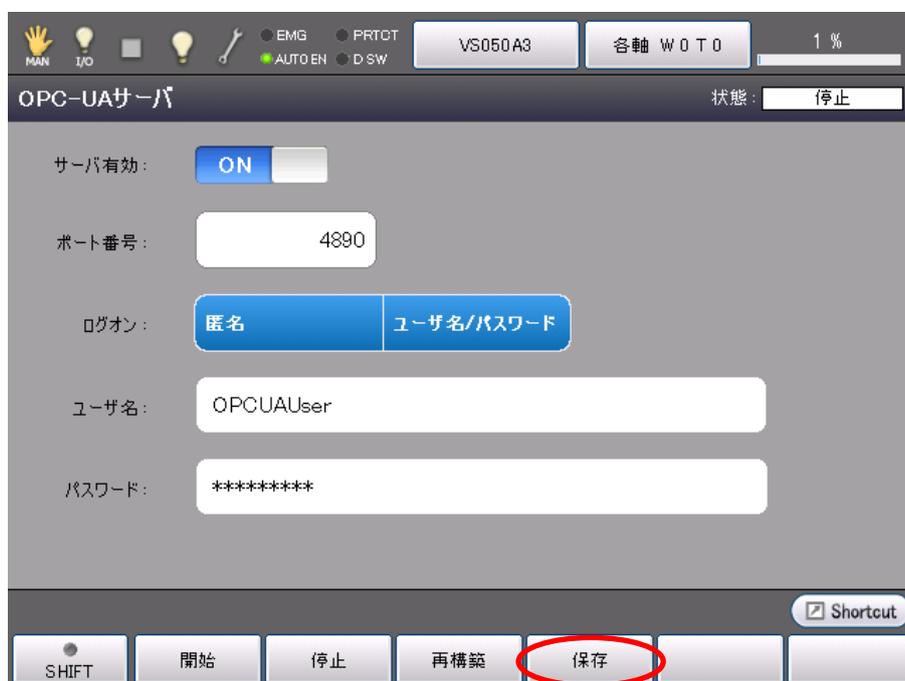


図 A-2 設定の保存

2-3 「再構築」を押下しセキュリティデータの再構築を行います

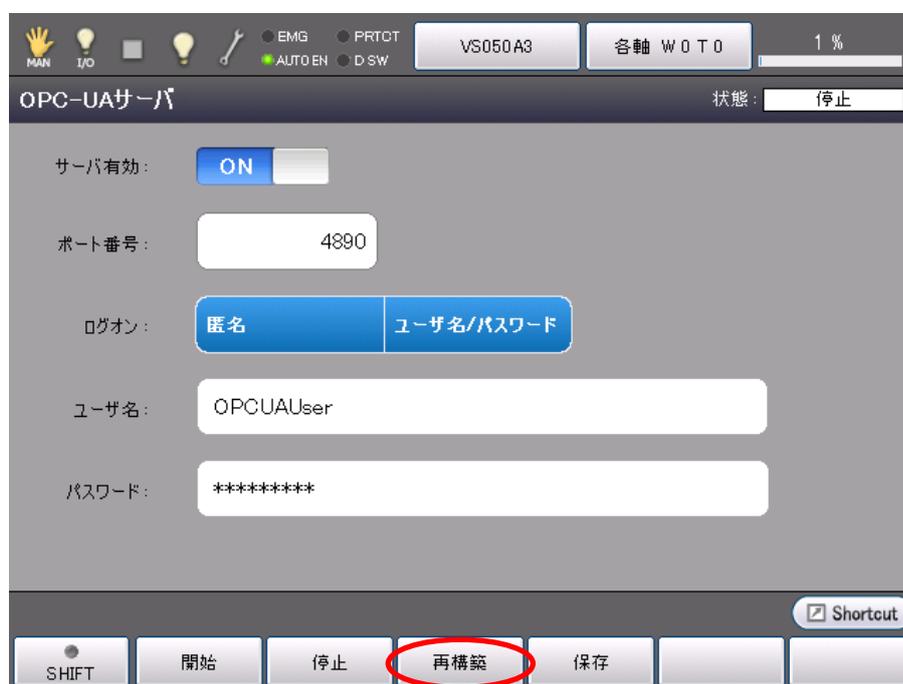


図 A-3 セキュリティデータ再構築

3. 公開アイテム設定 (csq ファイルの作成)
 - 3-1 「[コントローラと変数の設定](#)」の設定を行います
詳しくは「[CaoOPCUAConfig](#)」を参照ください
 - 3-2 設定アイテム情報を csq に保存してください

4. アイテムデータを RC8 に送信

4-1 WINCAPSIII を起動し、プロジェクトを新規作成(コントローラから情報を取得して新規作成)します

詳しくは「WINCAPSIII ガイド」を参照ください

4-2 作成したプロジェクトのフォルダの「プロジェクト名.ExData¥OPCUA¥」に「3.」で作成した「CaoOPCUA.csq」をコピーします(ここではプロジェクト名を「OPCUA_Trans」で作成)

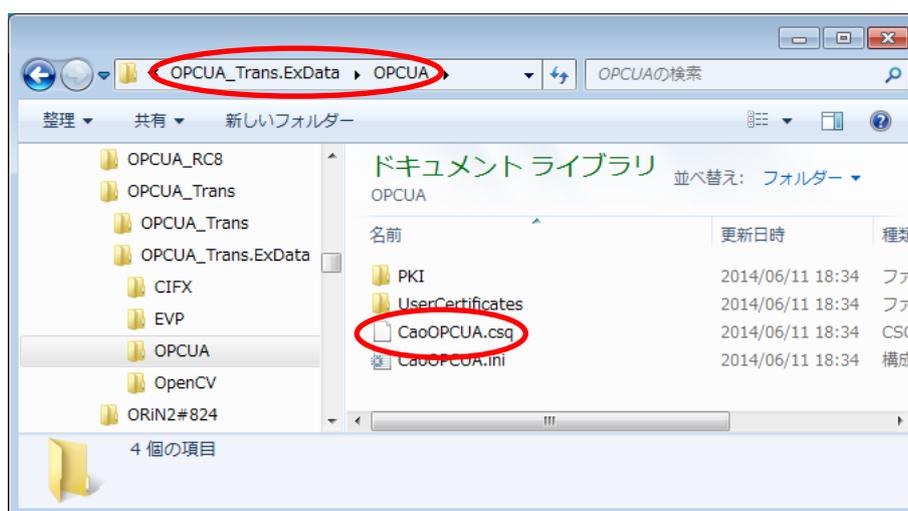


図 A-4 「CaoOPCUA.csq」のコピー

4-3 WINCAPSIII のメニューから「通信」→「データ送受信」を選択します

4-4 表示されたダイアログの「WINCAPSIII」欄の「その他」→「OPCUA」にチェックを入れます

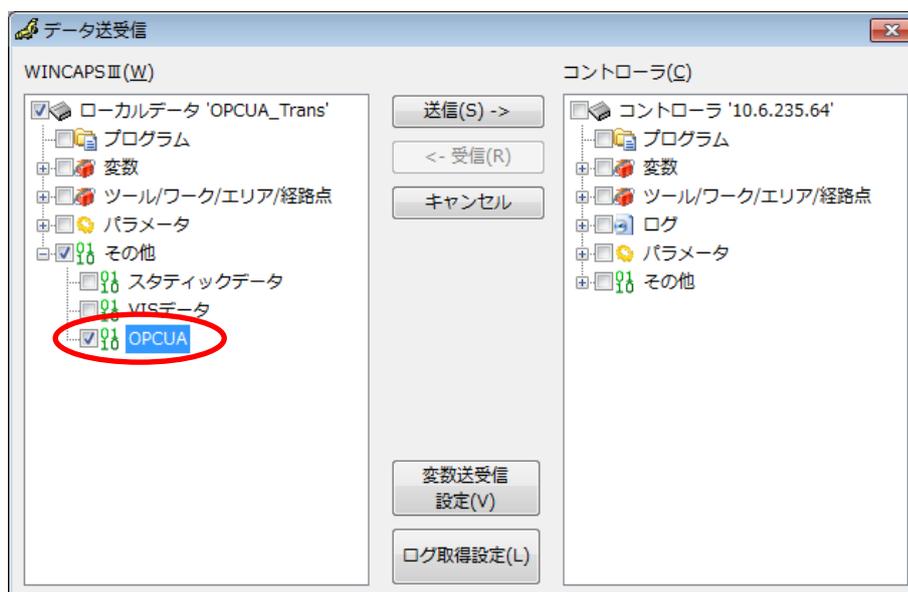


図 A-5 データ送受信

4-5 「送信」ボタンを押下します

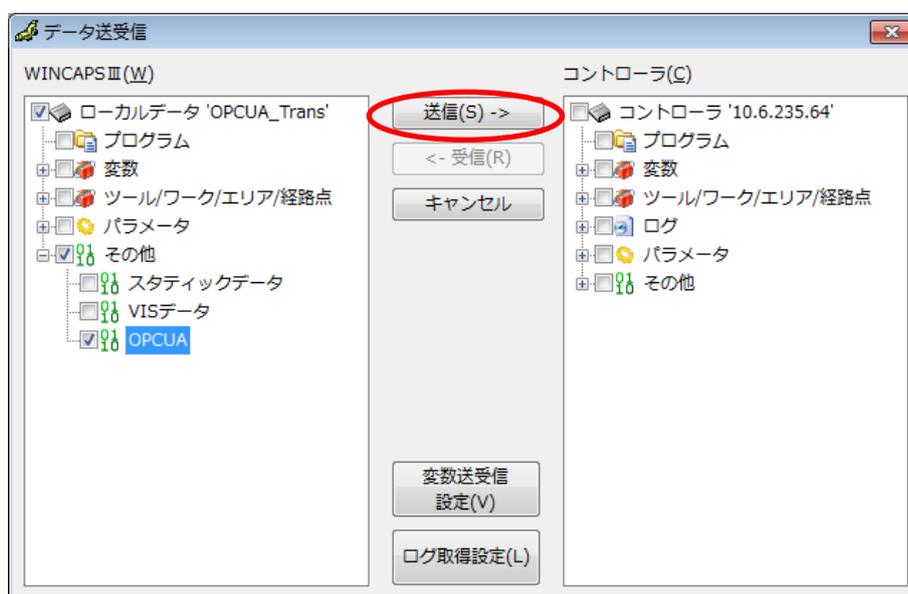


図 A-6 送信

5. OPC UA サーバの起動

5-1 ティーチングペンダントを起動し、「設定」→「通信と起動権」→「OPC-UA」サーバと操作します

5-2 「開始」を押下します

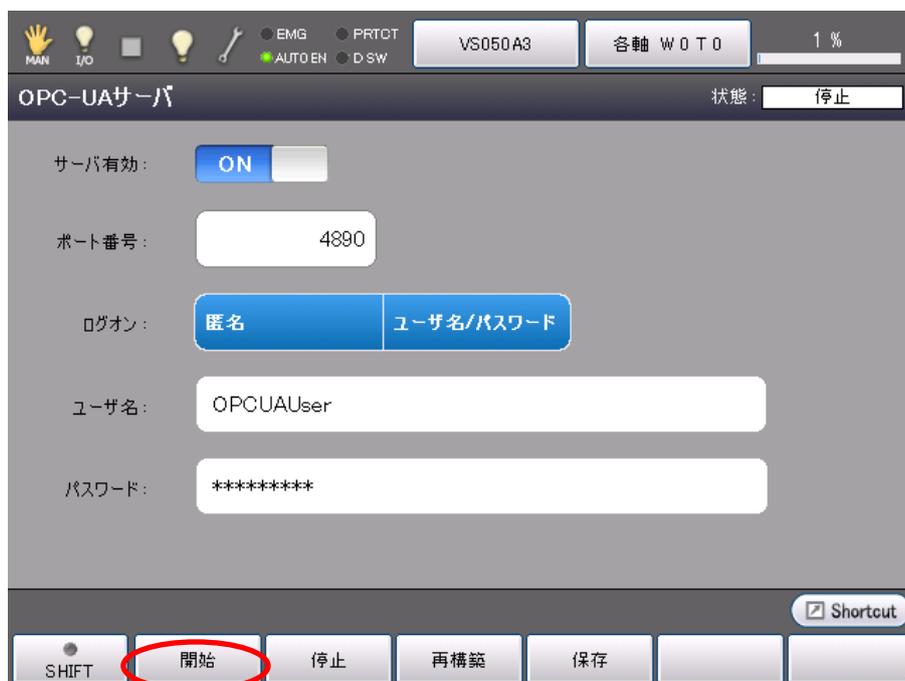


図 A-7 OPC UA サーバ開始

5-3 画面の「状態:」欄が「実行中」となっていることを確認します

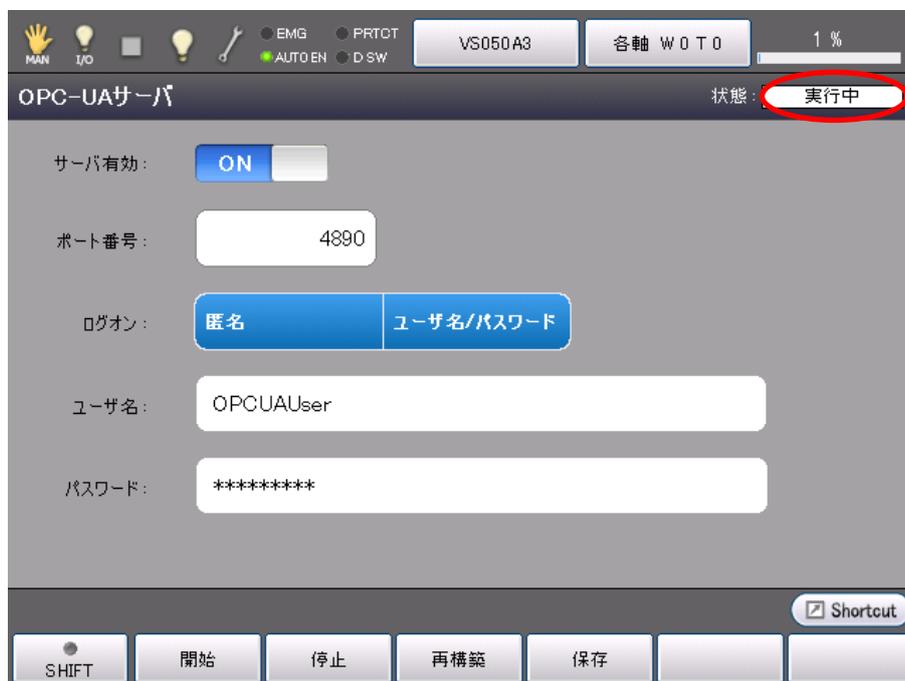


図 A-8 OPC UA サーバ実行中

Appendix A.2. CaoOPCUA サーバのアイテム編集

RC8 のアイテムデータを編集する場合は以下のように行います。

1. アイテムデータの受信

1-1 WINCAPSIII を起動し、プロジェクトを新規作成 (コントローラから情報を取得して新規作成) します
詳しくは「WINCAPSIII ガイド」を参照ください

1-2 WINCAPSIII のメニューから「通信」→「データ送受信」を選択します

1-3 表示されたダイアログの「コントローラ」欄の「その他」→「OPCUA」にチェックを入れます

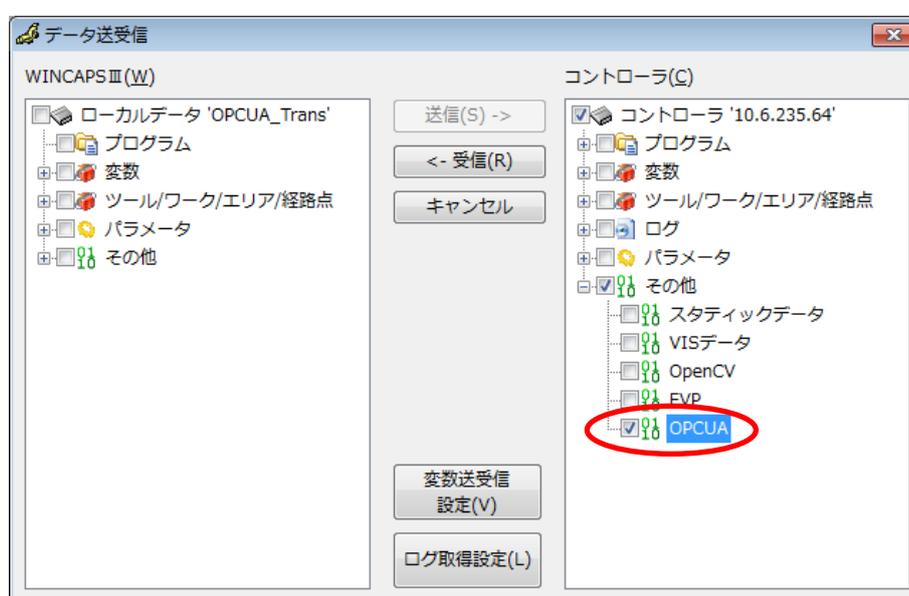


図 A-9 データ送受信

1-4 「受信」ボタンを押下します

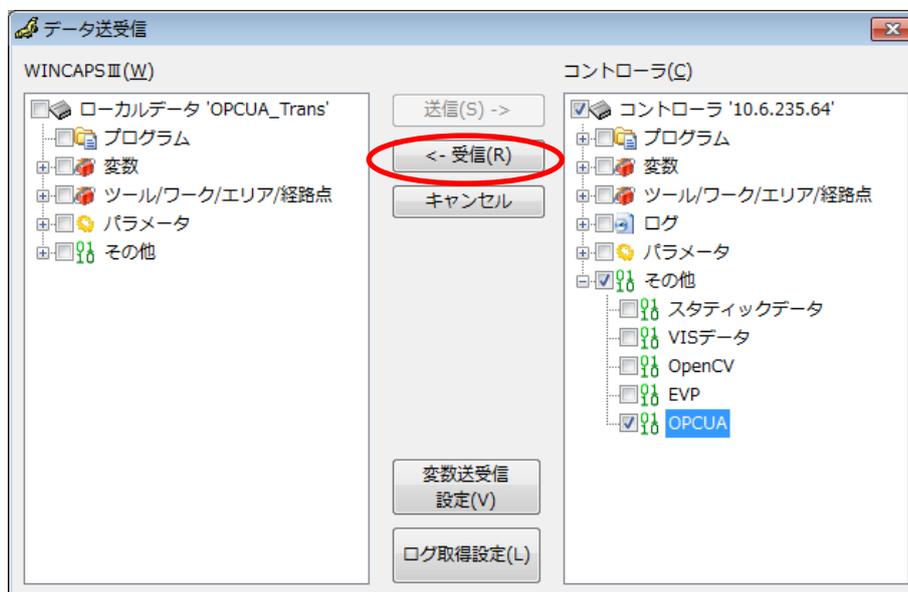


図 A-10 受信

1-5 作成したプロジェクトのフォルダの「プロジェクト名.ExData¥OPCUA¥」に受信したデータが保存されます(ここではプロジェクト名を「OPCUA_Trans」で作成)

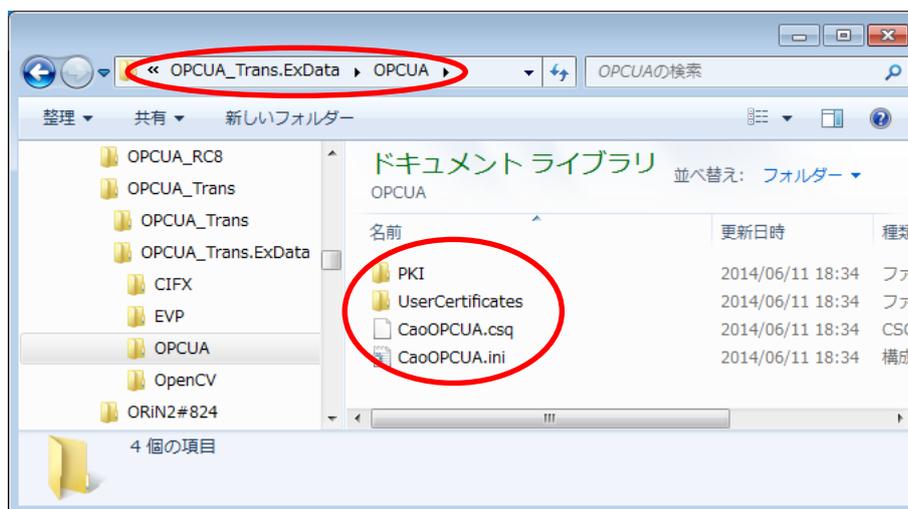


図 A-11 受信データ

2. アイテムデータの編集

2-1 CaoOPCUAConfig を起動し, A.2.1. で受信した「CaoOPCUA.csq」を開きます

2-2 「[アイテム追加](#)」～の手順でアイテムの追加等の編集を行います

詳しくは「[CaoOPCUAConfig](#)」を参照ください

2-3 「CaoOPCUA.csq」を上書き保存します

3. アイテムデータの送信

「[A.2.4-3](#)」～の手順でアイテムの送信を行います

Appendix A.3. CaoOPCUA サーバとのセキュアな接続

CaoOPCUA サーバとのセキュアな接続を確立する為には各 OPC UA クライアントが持っている証明書ファイルと必要に応じてユーザ証明書を WINCAPSIII を使用して RC8 に転送する必要があります。

1. クライアントアプリケーションの証明書ファイルを用意します

詳しくは各クライアントアプリケーションのマニュアルなどを参照ください

2. データの受信を行います

「[A.2.1](#)」の手順で WINCAPSIII プロジェクトを作成し, データを受信します

3. クライアントアプリケーションの証明書ファイルをコピーします

作成したプロジェクトのフォルダの「プロジェクト名.ExData¥OPCUA¥PKI¥store¥certs¥」に用意したクライアントアプリケーションの証明書ファイルをコピーします

4. ユーザの証明書ファイルをコピーします

作成したプロジェクトのフォルダの「プロジェクト名.ExData¥OPCUA¥UserCertificate¥certs¥」に用意したユーザの証明書ファイルをコピーします

詳しくは「[3.3.3.3.1](#)」を参照ください

補足:この操作は「匿名アカウント」または「ユーザ名/パスワード」でログオンする場合には必要ありません

5. データの送信を行います

「[A.2.4-3](#)」～の手順でデータの送信を行います

6. セキュアな接続を行います

詳しくは「[3.3.2.](#)」を参照ください