

# InfoSOSA Modbus 機器接続ガイド

株式会社 デイ・エム・シー  
<https://www.dush.co.jp/>

# 著作権および商標に関する記述

---

- 本書の著作権は、株式会社ディ・エム・シーが所有しています。
- 本書の内容の一部、または全てを無断で掲載することは禁止されています。
- 本書の内容は予告なしに変更することがあります。あらかじめご了承ください。
- 本書の内容に起源したことによるお客様の損害その他の不利益、または第三者からのいかなる請求につきましても当社はその責任を負いません。あらかじめご了承ください。
- Microsoft®、Windows®、は米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
- その他の会社および製品名は、各社の商標または登録商標です。

## 表記について

---



IS731 シリーズにのみ関わる項目です。



IS-APP(EM シリーズ)にのみ関わる項目です。



IS731、IS-APP 共通の項目です。

# 目次

---

著作権および商標に関する記述 .....	2
表記について .....	2
目次 .....	3
1章 はじめに .....	5
1.1 概要 .....	6
1.2 対応 Modbus 機器 .....	9
1.3 関連ドキュメント .....	10
2章 準備 .....	12
2.1 共通(InfoSOSA Builder) .....	13
2.2 EM シリーズ .....	14
2.3 IS731 シリーズ .....	21
3章 使用方法 .....	22
3.1 プロジェクト設定 .....	23
3.1.1 Modbus メモリ作成 .....	23
3.1.2 同時読み込みデータ数設定 .....	27
3.1.3 優先区間設定 .....	29
3.1.4 Modbus 設定のエクスポート .....	32
3.1.5 Modbus 設定のインポート .....	33
3.2 IS-APP の起動設定 .....	34
3.3 IS731 の通信設定 .....	36
3.4 Modbus 機器の値を読み書きする方法 .....	38
3.4.1 Modbus 機器のデバイスの値を表示する .....	38
3.4.2 Modbus 機器のデバイスに値を書き込む .....	41
4章 リファレンス .....	44
4.1 Modbus 通信用メモリ .....	45
4.1.1 プロパティ .....	46
4.1.2 値変化イベント (On Change Value) .....	49
4.1.3 上位通信 (InfoSOSA 専用プロトコル) .....	51
4.2 Modbus 通信用イベント .....	52
4.2.1 Link Error .....	53
4.2.2 Link Recover .....	55
4.2.3 First Sync Complete .....	56
4.3 IS-APP コマンドライン引数 .....	57
4.4 Modbus 設定ファイル .....	59
4.4.1 csv 形式(個別設定) .....	59
4.4.2 ini 形式(一括設定) .....	64
お問い合わせ .....	67



# 1章 はじめに

## 1.1 概要

---



本書は、InfoSOSA の Modbus プロトコルによる通信機能について記載しています。

InfoSOSA の画面に Modbus 機器のデバイスの値を表示したり、InfoSOSA の入力値を Modbus 機器へ反映させたりすることができます。

※ 本書では Modbus プロトコルで接続可能な機器を Modbus 機器と記載します。

Modbus 通信機能以外の InfoSOSA の機能については、別紙「InfoSOSA リファレンスマニュアル」を参照ください。



### InfoSOSA Builder のバージョン

InfoSOSA Builder は、下記のバージョンである必要があります。

InfoSOSA Builder バージョン	2.7.1 ~
------------------------	---------

※ Modbus 機能は InfoSOSA Builder のシミュレータには対応していません。



### IS-APP 関連のバージョン

IS-APP 関連のアプリケーションは、下記のバージョンである必要があります。

IS-APP バージョン	2.4.1 ~
IS-APP 設定ツールバージョン	3.1.1 ~



### EM シリーズ本体のシステムバージョン

EM シリーズ本体は、下記のシステムバージョンである必要があります。

EM シリーズ本体のシステムバージョンは、システム設定ツールから確認ください。

システム設定ツールについては、別紙「EM シリーズ ツールマニュアル」を参照ください。

対象システムバージョン	3.0.0 ~
-------------	---------

The screenshot shows the 'システム設定' (System Settings) window. On the left is a sidebar with menu items: カーソル設定, 時計設定, ロケールと言語, サウンド設定, LAN情報, LAN設定, TPキャリブレーション, 自動起動, 書き込み保護, and システム情報. The main area displays system information: 'システム設定ツールバージョン' (3.0.0), 'システムバージョン' (3.0.0, highlighted with a red box), and 'カーネルバージョン' (Linux version 4.1.15yocto-for-EMG7-falcon +g77f6154 (seedsware@seedsware-VirtualBox) (gcc version 5.3.0 (GCC) ) #1 SMP PREEMPT Wed Mar 25 10:48:40 JST 2020).

EM シリーズ本体のシステムバージョンを更新する場合は、別紙「EM シリーズ Version3.0 システム更新手順書」を参照ください。



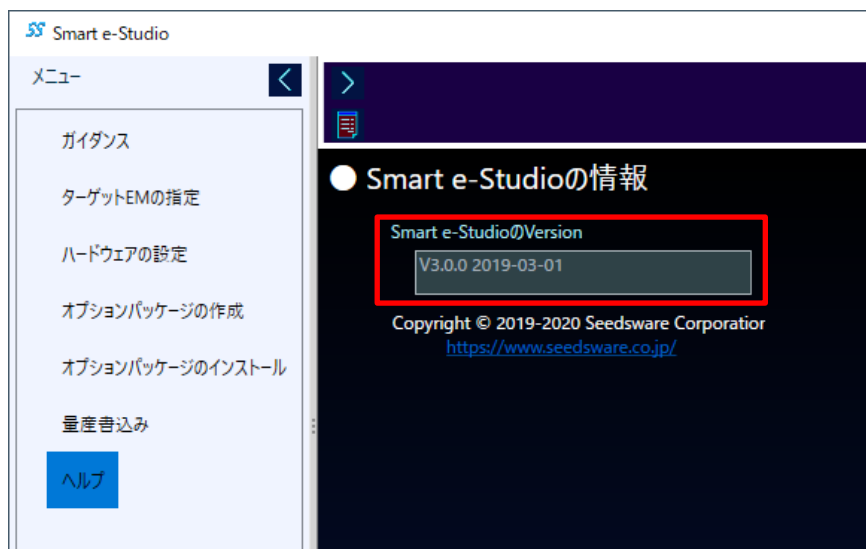
## Smart e-Studio バージョン

Smart e-Studio は、下記のバージョンである必要があります。

Smart e-Studio のバージョンは、起動後のヘルプメニューから確認ください。

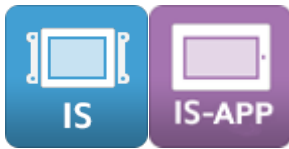
Smart e-Studio については、別紙「Smart e-Studio 取扱説明書」を参照ください。

対象 Smart e-Studio バージョン	3.0.0 ~
-------------------------	---------





## 1.2 対応Modbus機器



以下に対応した Modbus プロトコル対応機器と接続します。

項目	内容
プロトコル	Modbus/TCP スレーブモード Modbus/RTU スレーブモード  ※InfoSOSA 側がマスターになり、スレーブモードの Modbus 機器と接続します。
ファンクションコード	0x01 (コイル読み出し) 0x03 (レジスタ読み出し) 0x0F (コイル書き込み) 0x10 (レジスタ書き込み)
データ型	ビット (1bit) バイト (1byte) 符号有り/無し ワード (2byte) 符号有り/無し ダブルワード (4byte) 符号有り
接続台数	1 台

※ Modbus 機器は、InfoSOSA からコマンドを受信後 100ms 以内にレスポンスコマンドを返信してください。

※ 上記を満たした全ての Modbus 機器と接続を保証するものではありません。

## 1.3 関連ドキュメント

---

本書に関連するドキュメントは以下になります。目的に合わせて参照ください。



### InfoSOSA リファレンスマニュアル

InfoSOSA の機能／仕様について記載しています。

### InfoSOSA ビルダ操作マニュアル

InfoSOSA ビルダの操作方法について記載しています。

### InfoSOSA リリースノート

InfoSOSA のバージョンによる違いを記載しています。



### IS731 シリーズ スタートアップガイド

IS731 シリーズをご使用される方向けのマニュアルです。IS731 シリーズの特徴のご紹介、チュートリアル、IS731 シリーズ固有の機能について記載しています。



### IS-APP スタートアップガイド

EM シリーズにインストールされている HMI アプリケーション「IS-APP」について記載しています。IS-APP の特徴のご紹介、チュートリアル、IS-APP 固有の機能／仕様について記載しています。

### EM シリーズ ソフトウェア開発マニュアル

EM シリーズで動作するソフトウェアの開発方法について記載しています。

### EM シリーズ ツールマニュアル

EM シリーズにインストールされているツールの使用方法について記載しています。

### Smart e-Studio 取扱説明書

EM シリーズ用リモートメンテナンスツール「Smart e-Studio」の機能や操作方法について記載しております。EM シリーズへ Modbus 機器通信用アプリケーションのインストールに使用します。

# 2章 準備

## 2.1 共通 (InfoSOSA Builder)

---



Modbus 機能対応バージョンの InfoSOSA Builder をインストールしてください。

InfoSOSA Builder バージョン	2.7.1 ~
------------------------	---------

※ Modbus 機能は InfoSOSA Builder のシミュレータには対応していません。

インストールの方法は、別紙「InfoSOSA ビルダ操作マニュアル」を参照ください。

## 2.2 EMシリーズ



EM シリーズへ Modbus 通信対応版の IS-APP と IS-APP 設定用アプリケーション「IS-APP 設定ツール」を「Smart e-Studio」を使用してインストールする方法を記載します。

※ 「Smart e-Studio」は、Windows® PC 上で動作する EM シリーズ用リモートメンテナンスツールです。詳しくは別紙「Smart e-Studio 取扱説明書」を参照ください。

1. EM シリーズ本体と PC を LAN ケーブルで接続し、SSH プロトコルで接続できるようにネットワークの設定を行ってください。

ネットワークの設定は、ネットワーク管理者にご確認お願い致します。

項目	内容
プロトコル	SSH プロトコル (TCP、ポート番号 22)
実行アプリケーション	C:\¥Seedsware¥Smart e-Studio¥Smart e-Studio.exe ※デフォルトインストール先

2. PC に「Smart e-Studio」をインストールしてください。
3. PC にコピーした Modbus 通信対応版のインストールパッケージファイルをダブルクリックしてください。

機種によって使用するファイルが異なります。

型式	インストールパッケージファイル名
EMG7-***A8-****-**7	IS-APP_update_A8_****.em-package
EM8-***A7-****-**7	IS-APP_update_A7_****.em-package
EMG8-***A7-****-**7	
EMP-***A7-****-**7	

\*\*\*\*\*にはバージョンが入ります。

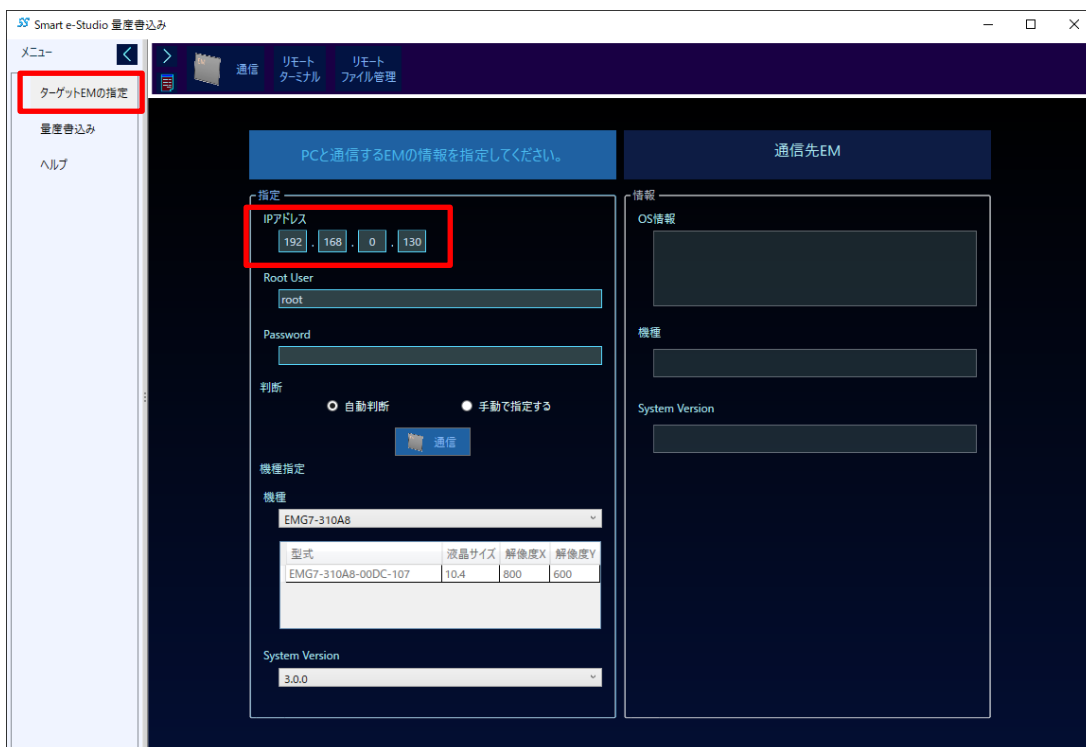
以下のバージョンのパッケージファイルをご使用ください。

Modbus 通信対応版バージョン	2411A ~
-------------------	---------

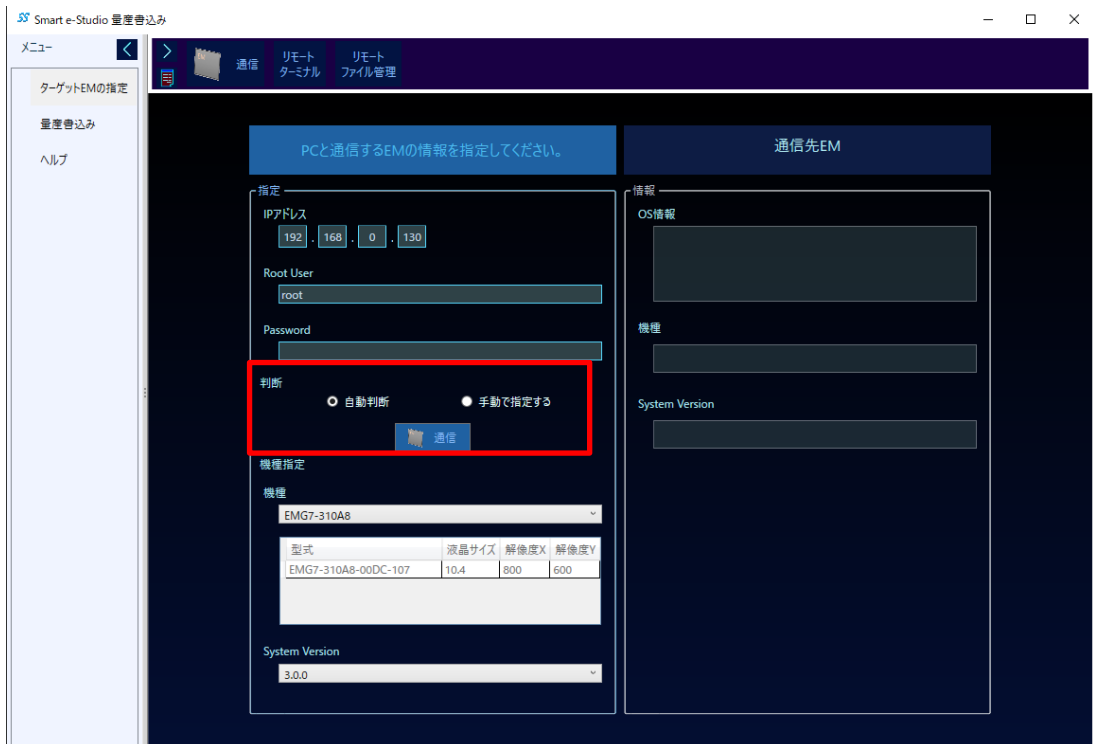
Smart e-Studio が起動します。



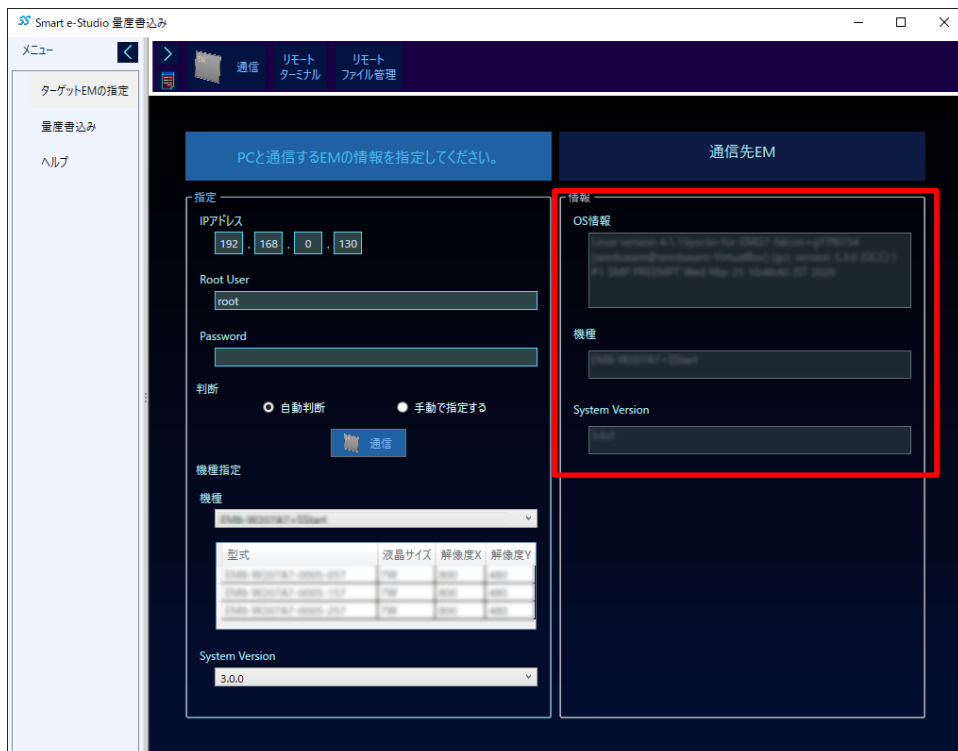
4. 「ターゲット EM の指定」 ページを開き、EM シリーズ本体の IP アドレスを入力してください。



5. 「自動判断」にチェックを入れて、「通信」ボタンを押してください。

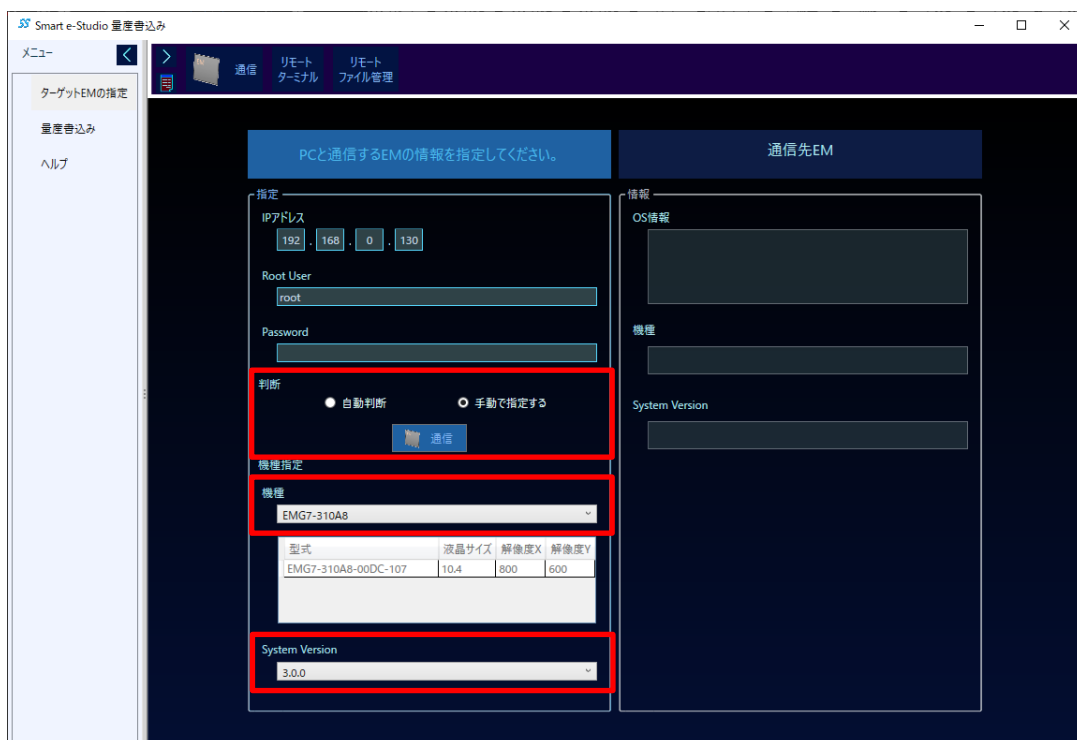


右側の欄に EM シリーズ本体の情報が表示されれば、成功です。





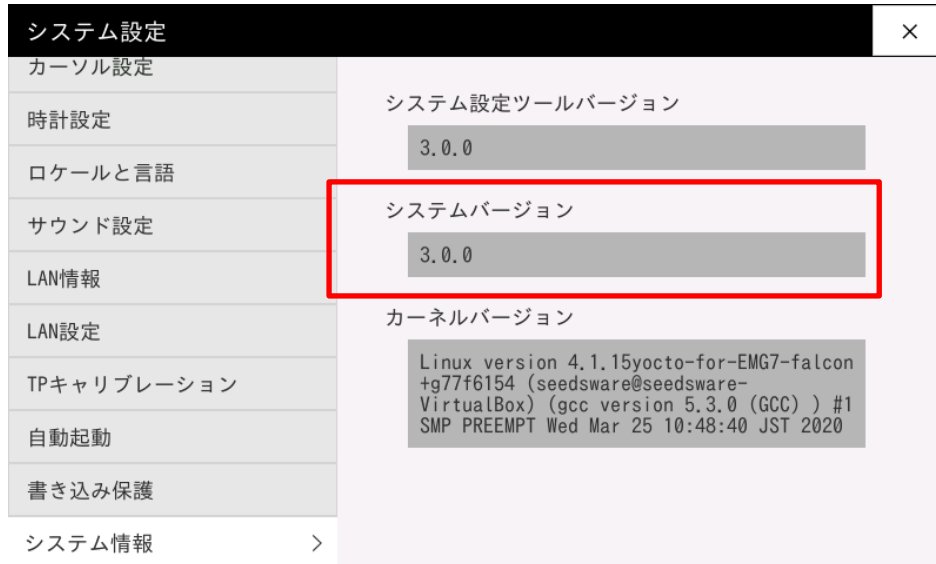
- ※ 自動判断が行えない場合は、「手動で指定する」にチェックを入れて、「機種」と「System Version」を指定後、「通信」ボタンを押してください。



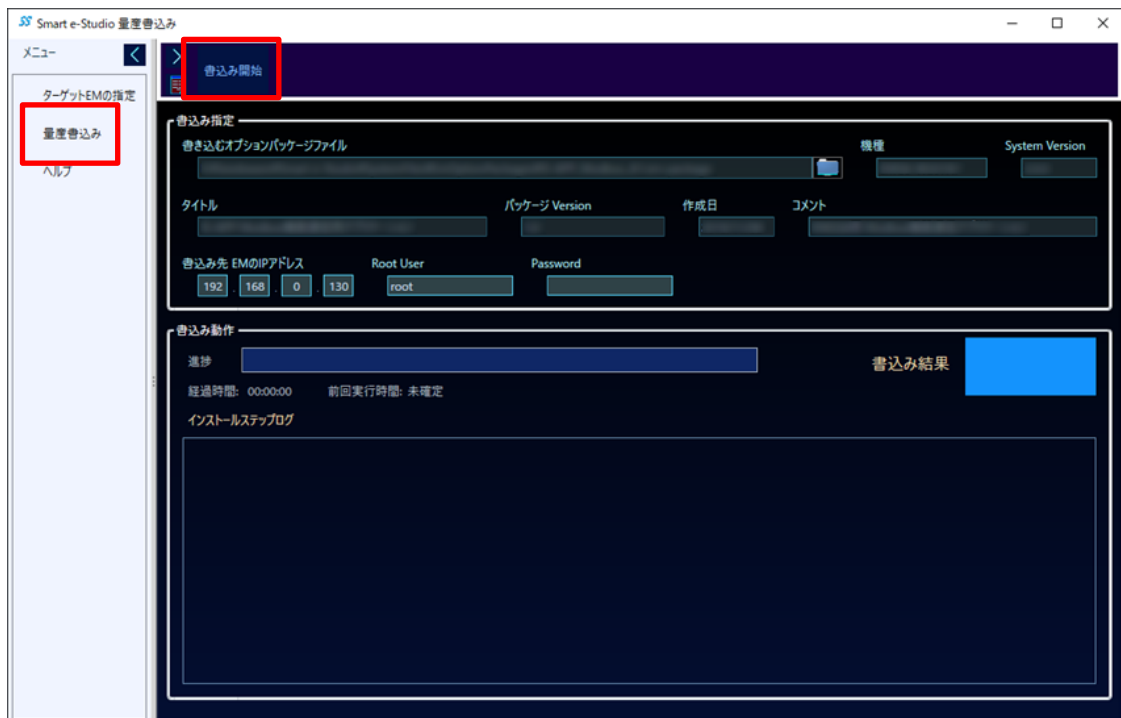
「機種」は以下からお使いの型式に合わせたものを選択してください。

機種	型式	備考
EM8-205A7	EM8-205A7-****-*07	5.7型、オープンフレームタイプ、アナログ抵抗膜モデル
EM8-205A7+SStart	EM8-205A7-****-*57	5.7型、オープンフレームタイプ、アナログ抵抗膜モデル、Smart e-Start 対応
EM8-W104A7	EM8-W104A7-****-*07	4.3型、オープンフレームタイプ、アナログ抵抗膜モデル
EM8-W104A7+SStart	EM8-W104A7-****-*57	4.3型、オープンフレームタイプ、アナログ抵抗膜モデル、Smart e-Start 対応
EM8-W207A7	EM8-W207A7-****-*07	7W型、オープンフレームタイプ、アナログ抵抗膜モデル
EM8-W207A7+SStart	EM8-W207A7-****-*57	7W型、オープンフレームタイプ、アナログ抵抗膜モデル、Smart e-Start 対応
EM8-W310A7	EM8-W310A7-****-*07	10.1W型、オープンフレームタイプ、アナログ抵抗膜モデル
EM8-W310A7+SStart	EM8-W310A7-****-*57	10.1W型、オープンフレームタイプ、アナログ抵抗膜モデル、Smart e-Start 対応
EMG7-310A8	EMG7-310A8-****-*07	10.4型、ハゼルタイプ、投影型静電容量モデル
EMG7-312A8	EMG7-312A8-****-*07	12.1型、ハゼルタイプ、投影型静電容量モデル
EMG7-W207A8	EMG7-W207A8-****-*07	7W型、ハゼルタイプ、投影型静電容量モデル
EMG8-205A7	EMG8-205A7-****-*07	5.7型、オープンフレームタイプ、投影型静電容量モデル
EMG8-205A7+SStart	EMG8-205A7-****-*57	5.7型、オープンフレームタイプ、投影型静電容量モデル、Smart e-Start 対応
EMG8-W104A7	EMG8-W104A7-****-*07	4.3型、オープンフレームタイプ、投影型静電容量モデル
EMG8-W104A7+SStart	EMG8-W104A7-****-*57	4.3型、オープンフレームタイプ、投影型静電容量モデル、Smart e-Start 対応
EMG8-W207A7	EMG8-W207A7-****-*07	7W型、オープンフレームタイプ、投影型静電容量モデル
EMG8-W207A7+SStart	EMG8-W207A7-****-*57	7W型、オープンフレームタイプ、投影型静電容量モデル、Smart e-Start 対応
EMG8-W310A7	EMG8-W310A7-****-*07	10W型、オープンフレームタイプ、投影型静電容量モデル
EMG8-W310A7+SStart	EMG8-W310A7-****-*57	10W型、オープンフレームタイプ、投影型静電容量モデル、Smart e-Start 対応
EMP-W207A7	EMP-W207A7-****-*07	7W型、ハンダントモデル
EMP-W207A7+SStart	EMP-W207A7-****-*57	7W型、ハンダントモデル、Smart e-Start 対応

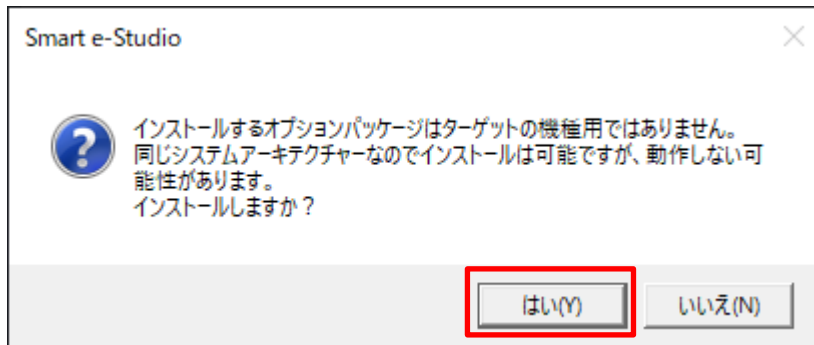
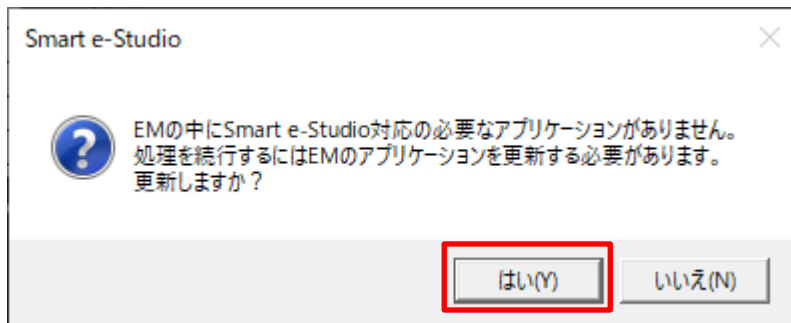
「System Version」は、ターゲットEM本体にインストールされている「システム設定ツール」を起動し、「システム情報」メニュー「システムバージョン」に表示されている値を選択してください。「システム設定ツール」については、別紙「EMシリーズツールマニュアル」を参照ください。



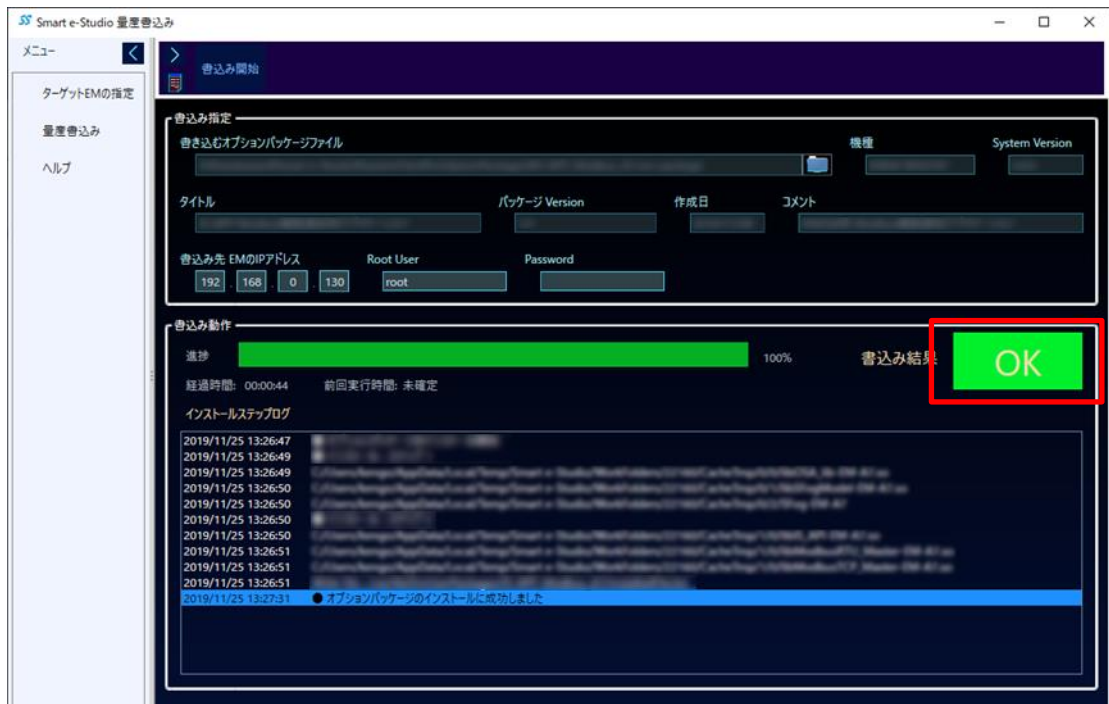
6. 通信成功の確認後、「量産書き込み」ページを開き、「書き込み開始」を押してください。



7. 以下のダイアログが表示された場合は「はい」を選択してください。



書き込み結果に「OK」が表示されると完了です。



## 2.3 IS731シリーズ

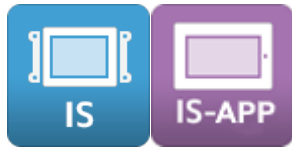
---



IS731 シリーズは「2.1 共通(InfoSOSA Builder)」のみです。本体の設定は InfoSOSA Builder からダウンロードを行うだけで、自動的に行われます。

# 3章 使用方法

## 3.1 プロジェクト設定

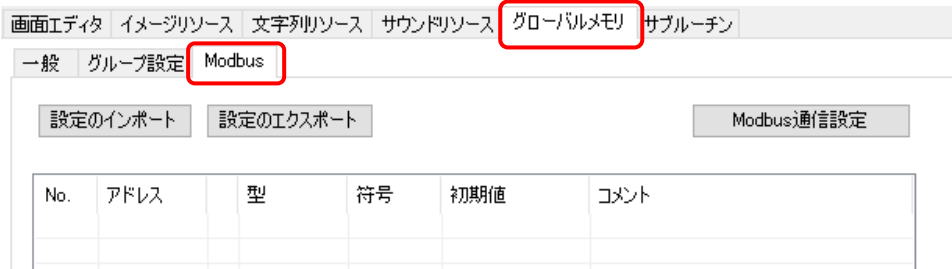


Modbus 機器と接続する為の、InfoSOSA Builder のプロジェクト（画面データ）の作成方法を記載します。

### 3.1.1 Modbus メモリ作成

Modbus 機器との通信用メモリを作成します。

1. 「グローバルメモリ」タブ内の「Modbus」タブを表示します。



2. 「追加」ボタンをクリックします。



3. 登録するメモリのアドレス範囲、パラメータの初期値を設定します。

**Modbusメモリ追加ダイアログ**

アドレス種別  
 Register     Coil

アドレス範囲  
 開始アドレス:   
 終了アドレス:

型  
 バイト     ワード     ダブルワード

符号  
 符号なし     符号あり

アンダーフロー  
 保持     ループ     クリップ

オーバーフロー  
 保持     ループ     クリップ

パラメータ	説明
アドレス種別	<p>登録する Modbus メモリの種別を「Register」「Coil」から選択します。デフォルトは「Register」です。</p> <p>「Register」の場合は、Modbus 機器に対する通信は以下のファンクションコードで行います。  <u>Read:0x03 Write:0x10</u></p> <p>「Coil」の場合は、Modbus 機器に対する通信は以下のファンクションコードで行います。  <u>Read:0x01 Write:0x0F</u></p>
アドレス範囲	<p>同期する Modbus アドレス範囲を指定します。            開始アドレス、終了アドレスを 10 進数で記載します。</p> <p>Modbus アドレスと Modbus 機器のレジスタの対応については、お使いの Modbus 機器の仕様書をご確認ください。</p> <p>登録可能な Modbus メモリ（Register、Coil 合計）はグローバルメモリと合わせて最大 2000 個です。</p>



パラメータ	説明												
型	<p>「Register」の場合は、「バイト」「ワード」「ダブルワード」から選択できます。デフォルトは「ワード」です。</p> <table border="1"> <thead> <tr> <th>型</th> <th>サイズ</th> </tr> </thead> <tbody> <tr> <td>バイト※1</td> <td>1byte</td> </tr> <tr> <td>ワード</td> <td>2byte</td> </tr> <tr> <td>ダブルワード※2</td> <td>4byte</td> </tr> </tbody> </table> <p>※1 バイトに設定した場合は、Modbus 機器の上位データ(1byte)は無視して取得します。</p> <p>※2 ダブルワードに設定した場合は、次のアドレスを上位データ (2byte) として使用します。次のアドレスと結合されます。</p> <p>コイル (COI) の場合は、「ビット」固定です。</p> <table border="1"> <thead> <tr> <th>型</th> <th>サイズ</th> </tr> </thead> <tbody> <tr> <td>ビット</td> <td>1bit</td> </tr> </tbody> </table>	型	サイズ	バイト※1	1byte	ワード	2byte	ダブルワード※2	4byte	型	サイズ	ビット	1bit
型	サイズ												
バイト※1	1byte												
ワード	2byte												
ダブルワード※2	4byte												
型	サイズ												
ビット	1bit												
符号	<p>取得したデータを「符号あり」として表示するか「符号なし」として表示するかを選択します。デフォルトは「符号なし」です。 ダブルワードの場合は「符号あり」固定になります。コイル (COI) の場合は、無効になります。</p> <table border="1"> <thead> <tr> <th>型</th> <th>符号</th> </tr> </thead> <tbody> <tr> <td>バイト</td> <td>符号あり／符号なし</td> </tr> <tr> <td>ワード</td> <td>符号あり／符号なし</td> </tr> <tr> <td>ダブルワード</td> <td>符号あり</td> </tr> </tbody> </table>	型	符号	バイト	符号あり／符号なし	ワード	符号あり／符号なし	ダブルワード	符号あり				
型	符号												
バイト	符号あり／符号なし												
ワード	符号あり／符号なし												
ダブルワード	符号あり												
アンダーフロー	<p>対象の Modbus メモリの最小値より小さな値が設定された場合の動作を選択します。デフォルトは「保持」です。コイル (COI) の場合は、無効になります。</p> <table border="1"> <thead> <tr> <th>設定</th> <th>動作</th> </tr> </thead> <tbody> <tr> <td>保持</td> <td>アンダーフローが発生する直前の値を保持します。</td> </tr> </tbody> </table>	設定	動作	保持	アンダーフローが発生する直前の値を保持します。								
設定	動作												
保持	アンダーフローが発生する直前の値を保持します。												

パラメータ	説明									
	ループ	下回った値を最大値から引いた値を対象メモリの値とします。								
	クリップ	対象メモリに最小値を設定します。								
オーバーフロー	対象の Modbus メモリの最大値より大きな値が設定された場合の動作を定義します。デフォルトは「保持」です。コイル（COI）の場合は、無効になります。									
	<table border="1"> <thead> <tr> <th>設定</th> <th>動作</th> </tr> </thead> <tbody> <tr> <td>保持</td> <td>オーバーフローが発生する直前の値を保持します。</td> </tr> <tr> <td>ループ</td> <td>上回った値を最小値に足した値を対象メモリの値とします。</td> </tr> <tr> <td>クリップ</td> <td>対象メモリに最大値を設定します。</td> </tr> </tbody> </table>		設定	動作	保持	オーバーフローが発生する直前の値を保持します。	ループ	上回った値を最小値に足した値を対象メモリの値とします。	クリップ	対象メモリに最大値を設定します。
設定	動作									
保持	オーバーフローが発生する直前の値を保持します。									
ループ	上回った値を最小値に足した値を対象メモリの値とします。									
クリップ	対象メモリに最大値を設定します。									

以下のように Modbus 機器通信用メモリが登録されます。

画面エディタ   イメージリソース   文字列リソース   サウンドリソース   グロ

一般   グループ設定   Modbus

設定のインポート   設定のエクスポート

No.	アドレス		型	符号	初期値
0001	REG00000	*	ワード	符号なし	
0002	REG00001	*	ワード	符号なし	
0003	REG00002	*	ワード	符号なし	
0004	REG00003	*	ワード	符号なし	
0005	REG00004	*	ワード	符号なし	
0006	REG00005	*	ワード	符号なし	

### 3.1.2 同時読み込みデータ数設定

接続する Modbus 機器に対して、1 回の通信でまとめて読み込みを行うアドレス数の最大数を設定します。ファンクションコードごとに設定します。

値を増やすほどパフォーマンスは向上しますが、最大値は接続する Modbus 機器の仕様書をご確認ください。

※ Modbus 機器側の最大値より大きな値を設定すると通信エラーになります。

1. 「グローバルメモリ」タブ内の「Modbus」タブを表示します。

画面エディタ イメージリソース 文字列リソース サウンドリソース **グローバルメモリ** サブルーチン

一般 グループ設定 **Modbus**

設定のインポート 設定のエクスポート **Modbus通信設定**

No.	アドレス		型	符号	初期値	コメント
0001	REG00000	*	ワード	符号なし		
0002	REG00001	*	ワード	符号なし		
0003	REG00002	*	ワード	符号なし		

2. 「Modbus 通信設定」ボタンをクリックします。

画面エディタ イメージリソース 文字列リソース サウンドリソース グローバルメモリ サブルーチン

一般 グループ設定 Modbus

設定のインポート 設定のエクスポート **Modbus通信設定**

No.	アドレス		型	符号	初期値	コメント
0001	REG00000	*	ワード	符号なし		
0002	REG00001	*	ワード	符号なし		
0003	REG00002	*	ワード	符号なし		

### 3. ファンクションコードごとの最大データ数を設定します。

値を増やすほどパフォーマンスは向上しますが、最大値は接続する Modbus 機器の仕様書をご確認ください。

※ Modbus 機器側の最大値より大きな値を設定すると通信エラーになります。

**Modbus通信設定ダイアログ**

**最大データ数設定**

ファンクションコードごとの最大データ数を設定します。  
値を増やすほどパフォーマンスは向上しますが、最大値は接続するModbus機器側の仕様をご確認ください。  
※Modbus機器側の最大値より大きな値を設定すると通信エラーになります。

Register	
0x03(複数読み出し最大データ数/ワード):	16
0x10(複数書き込み最大データ数/ワード):	16
Coil	
0x01(複数読み出し最大データ数/ビット):	16
0x0F(複数書き込み最大データ数/ビット):	16

**優先区間設定**

優先してデータを取得するアドレス範囲を指定する

Register		
開始アドレス:	<input type="text"/>	終了アドレス: <input type="text"/>
Coil		
開始アドレス:	<input type="text"/>	終了アドレス: <input type="text"/>

通常区間待機時間(msec):

### 3.1.3 優先区間設定

登録する Modbus メモリ数に比例して、全てのデバイスの値の読み込みに必要な時間が増加する為、Modbus 機器側の値が変わってから、InfoSOSA 上の画面の表示への反映までの時間が増加します。

その場合、優先して値を読み込むアドレスを設定することが可能です。

1. 「グローバルメモリ」タブ内の「Modbus」タブを表示します。

画面エディタ | イメージリソース | 文字列リソース | サウンドリソース | **グローバルメモリ** | サブルーチン

一般 | グループ設定 | **Modbus**

設定のインポート | 設定のエクスポート | **Modbus通信設定**

No.	アドレス		型	符号	初期値	コメント
0001	REG00000	*	ワード	符号なし		
0002	REG00001	*	ワード	符号なし		
0003	REG00002	*	ワード	符号なし		

2. 「Modbus 通信設定」ボタンをクリックします。

画面エディタ | イメージリソース | 文字列リソース | サウンドリソース | **グローバルメモリ** | サブルーチン

一般 | グループ設定 | **Modbus**

設定のインポート | 設定のエクスポート | **Modbus通信設定**

No.	アドレス		型	符号	初期値	コメント
0001	REG00000	*	ワード	符号なし		
0002	REG00001	*	ワード	符号なし		
0003	REG00002	*	ワード	符号なし		

3. 「優先してデータを取得するアドレス範囲を指定する」にチェックを付けます。

Modbus通信設定ダイアログ

最大データ数設定

ファンクションコードごとの最大データ数を設定します。  
値を増やすほどパフォーマンスは向上しますが、最大値は接続するModbus機器側の仕様をご確認ください。  
※Modbus機器側の最大値より大きな値を設定すると通信エラーになります。

Register

0x03(複数読み出し最大データ数/ワード): 16

0x10(複数書き込み最大データ数/ワード): 16

Coil

0x01(複数読み出し最大データ数/ビット): 16

0x0F(複数書き込み最大データ数/ビット): 16

優先区間設定

優先してデータを取得するアドレス範囲を指定する

Register

開始アドレス: 終了アドレス:

Coil

開始アドレス: 終了アドレス:

通常区間待ち時間(msec): 100

設定 キャンセル

4. 優先区間に設定する「開始アドレス」と「終了アドレス」を指定してください。  
Register、Coilそれぞれ指定できます。

Modbus通信設定ダイアログ

最大データ数設定

ファンクションコードごとの最大データ数を設定します。  
値を増やすほどパフォーマンスは向上しますが、最大値は接続するModbus機器側の仕様をご確認ください。  
※Modbus機器側の最大値より大きな値を設定すると通信エラーになります。

Register

0x03(複数読み出し最大データ数/ワード): 16

0x10(複数書き込み最大データ数/ワード): 16

Coil

0x01(複数読み出し最大データ数/ビット): 16

0x0F(複数書き込み最大データ数/ビット): 16

優先区間設定

優先してデータを取得するアドレス範囲を指定する

Register

開始アドレス: 0 終了アドレス: 5

Coil

開始アドレス: 終了アドレス:

通常区間待ち時間(msec): 100

設定 キャンセル

5. 通常区間の待機時間を指定してください。

単位はミリ秒です。一度通常区間のデバイスの値を読み込んだ後、待機する時間です。

※ 本設定は同期時間ではございません。実際の同期時間は、通常区間に設定したメモリの数に比例します。（待機時間経過に読み込みを開始するため、設定値以上になります）

Modbus通信設定ダイアログ

最大データ数設定

ファンクションコードごとの最大データ数を設定します。  
 値を増やすほどパフォーマンスは向上しますが、最大値は接続するModbus機器側の仕様をご確認ください。  
 ※ Modbus機器側の最大値より大きな値を設定すると通信エラーになります。

Register

0x03(複数読み出し最大データ数/ワード): 16

0x10(複数書き込み最大データ数/ワード): 16

Coil

0x01(複数読み出し最大データ数/ビット): 16

0x0F(複数書き込み最大データ数/ビット): 16

優先区間設定

優先してデータを取得するアドレス範囲を指定する

Register

開始アドレス: 0 終了アドレス: 5

Coil

開始アドレス: 終了アドレス:

通常区間待機時間(msec): 100

設定 キャンセル

優先区間に設定されたアドレスには「\*」が付きます。

画面エディタ イメージリソース 文字列リソース サウンドリソース グローバ

一般 グループ設定 Modbus

設定のインポート 設定のエクスポート

No.	アドレス	型	符号	初期値
0001	REG00000	* ワード	符号なし	
0002	REG00001	* ワード	符号なし	
0003	REG00002	* ワード	符号なし	
0004	REG00003	* ワード	符号なし	
0005	REG00004	* ワード	符号なし	
0006	REG00005	* ワード	符号なし	
0007	REG00006	ワード	符号なし	
0008	REG00007	ワード	符号なし	
0009	REG00008	ワード	符号なし	
0010	REG00009	ワード	符号なし	
0011	REG00010	ワード	符号なし	
0012	COI00000	ビット	符号なし	
0013	COI00001	ビット	符号なし	

### 3.1.4 Modbus 設定のエクスポート

Modbus メモリ設定と Modbus 通信設定は csv 形式のファイルにエクスポートすることが可能です。

※ Modbus メモリに設定された値変化イベントアクションはエクスポートされません。

1. 「グローバルメモリ」タブ内の「Modbus」タブを表示します。

The screenshot shows a web interface with a top navigation bar containing tabs: 画面エディタ, イメージリソース, 文字列リソース, サウンドリソース, グローバルメモリ (highlighted with a red box), and サブルーチン. Below this is a sub-navigation bar with tabs: 一般, グループ設定, and Modbus (highlighted with a red box). The main content area has three buttons: 設定のインポート, 設定のエクスポート, and Modbus通信設定 (highlighted with a blue box). Below the buttons is a table with the following data:

No.	アドレス		型	符号	初期値	コメント
0001	REG00000	*	ワード	符号なし		
0002	REG00001	*	ワード	符号なし		
0003	REG00002	*	ワード	符号なし		

2. 「設定のエクスポート」ボタンをクリックします。

This screenshot is identical to the previous one, but the '設定のエクスポート' button is now highlighted with a red box, indicating it has been clicked.

3. 指定した場所に csv 形式のファイルが保存されます。

エクスポートした csv ファイルは、編集することが可能です。  
詳細は「4.4 Modbus 設定ファイル」を参照ください。



### 3.1.5 Modbus 設定のインポート

エクスポートした Modbus 設定ファイルをインポートすることが可能です。

※ 現在の設定は上書きされますのでご注意ください。

1. 「グローバルメモリ」タブ内の「Modbus」タブを表示します。

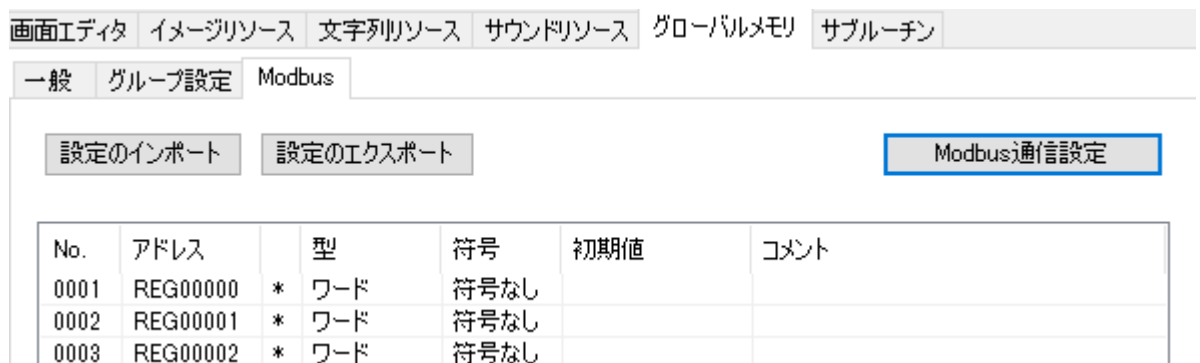


2. 「設定のインポート」ボタンをクリックします。



3. Modbus 設定ファイルを選択します。

4. Modbus メモリ設定、Modbus 通信設定が設定されます。



## 3.2 IS-APPの起動設定



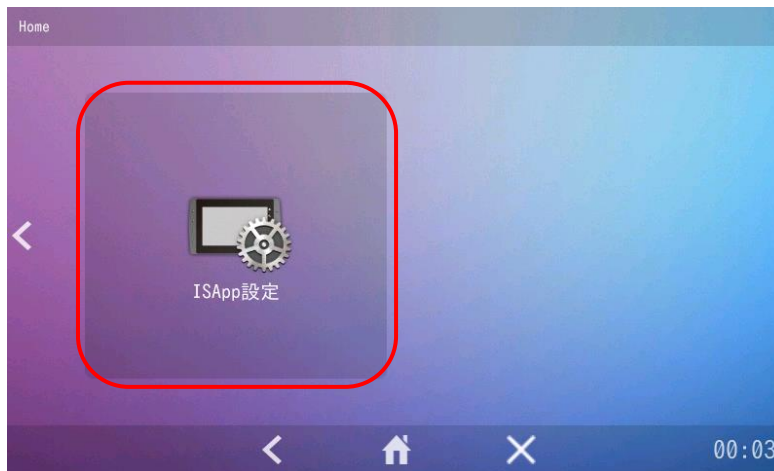
IS-APP は、通信設定を起動時のコマンドライン引数で行います。「IS-APP 設定ツール」をご使用頂くとコマンドライン引数が設定された IS-APP の起動スクリプトが簡単に作成できます。

IS-APP 設定ツールを使用しない場合は、「4.3 IS-APP コマンドライン引数」を参照ください。

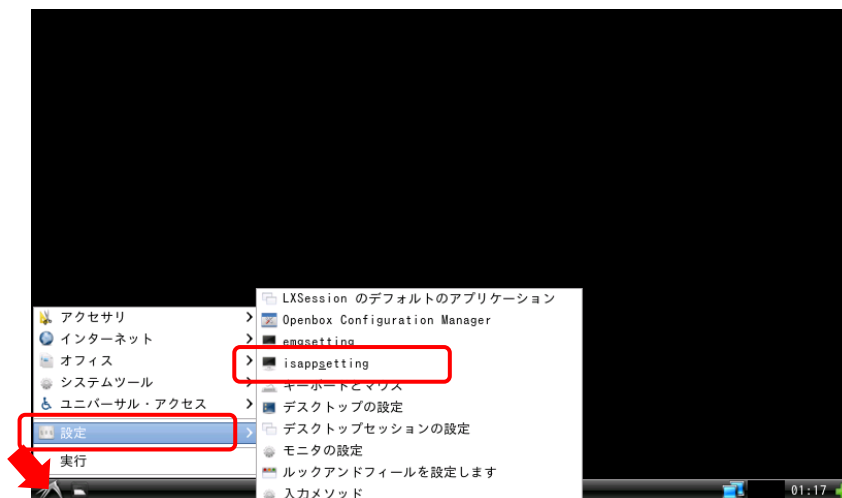
通信を行う Modbus 機器の IP アドレスを指定します。

1. EM Linux 本体上で「IS-APP SETTING」ツールを起動します。

[EMG ランチャー] - [ISApp 設定]



[スタートメニュー] - [設定] - [isappsetting]



実行ファイル
/usr/bin/isapp_setting

詳しい使い方は、別紙「IS-APP スタートアップガイド」を参照ください。

2. 以下の通信先の設定を行い、OK ボタンをタッチします。

TCP の場合：

アドレス、ポート番号、局番設定

RTU の場合：

通信ポート、局番設定、種別、通信速度、パリティ



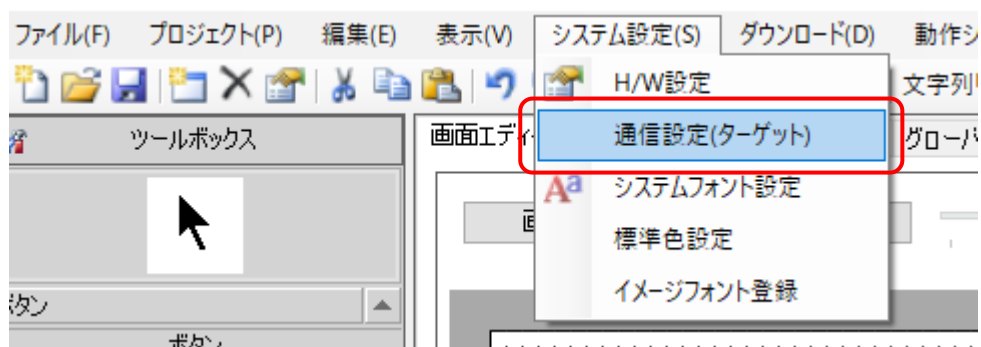
上記手順で、Modbus 機器の通信先が設定された IS-APP 起動スクリプト「/mnt/user/isapp\_run.sh」が作成されます。

## 3.3 IS731の通信設定

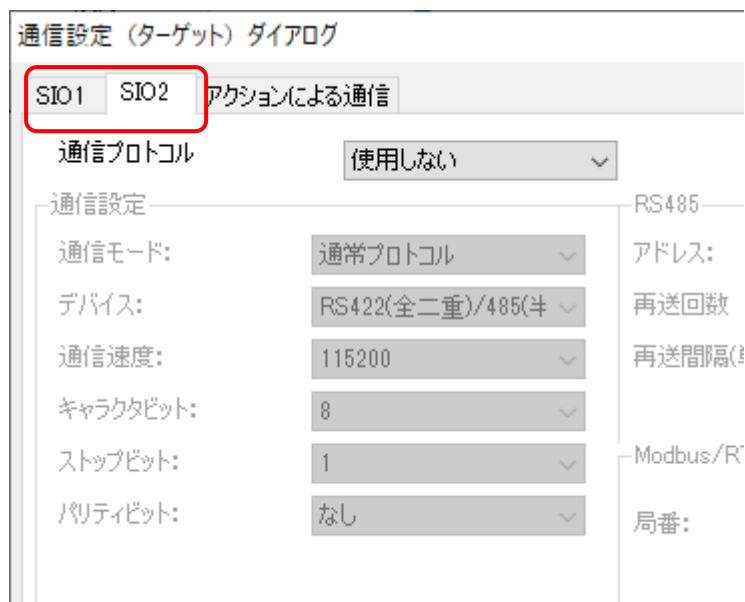


IS731 シリーズは、通信設定を InfoSOSA Builder から行います。

1. 「システム設定」→「通信設定（ターゲット）」を開きます。



2. 使用するポートのタブ（SIO1 または SIO2）を選択します。



3. 通信プロトコルに「Modbus/RTU」を選択します。

The screenshot shows the '通信設定 (ターゲット) ダイアログ' (Communication Settings (Target) Dialog) with the 'アクションによる通信' (Communication by Action) tab selected. The '通信プロトコル' (Communication Protocol) dropdown menu is highlighted with a red box and set to 'Modbus/RTU'. The '通信設定' (Communication Settings) section includes the following parameters:

通信設定	値
通信モード	通常プロトコル
デバイス	RS422(全二重)/485(半)
通信速度	115200
キャラクタビット	8
ストップビット	1
パリティビット	なし

The 'RS485' section includes the following parameters:

RS485	値
アドレス	1
再送回数	3
再送間隔(単位:10msec)	100

The 'Modbus/RTU' section includes the following parameter:

Modbus/RTU	値
局番	1

4. 通信を行う Modbus 機器に合わせて、「通信速度」「パリティビット」「局番」を設定します。

The screenshot shows the '通信設定 (ターゲット) ダイアログ' (Communication Settings (Target) Dialog) with the 'アクションによる通信' (Communication by Action) tab selected. The '通信プロトコル' (Communication Protocol) dropdown menu is set to 'Modbus/RTU'. The '通信設定' (Communication Settings) section is highlighted with a red box and includes the following parameters:

通信設定	値
通信モード	通常プロトコル
デバイス	RS422(全二重)/485(半)
通信速度	115200
キャラクタビット	8
ストップビット	1
パリティビット	なし

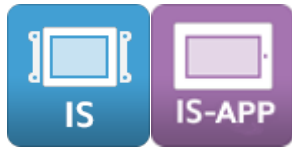
The 'RS485' section includes the following parameters:

RS485	値
アドレス	1
再送回数	3
再送間隔(単位:10msec)	100

The 'Modbus/RTU' section includes the following parameter:

Modbus/RTU	値
局番	1

## 3.4 Modbus機器の値を読み書きする方法

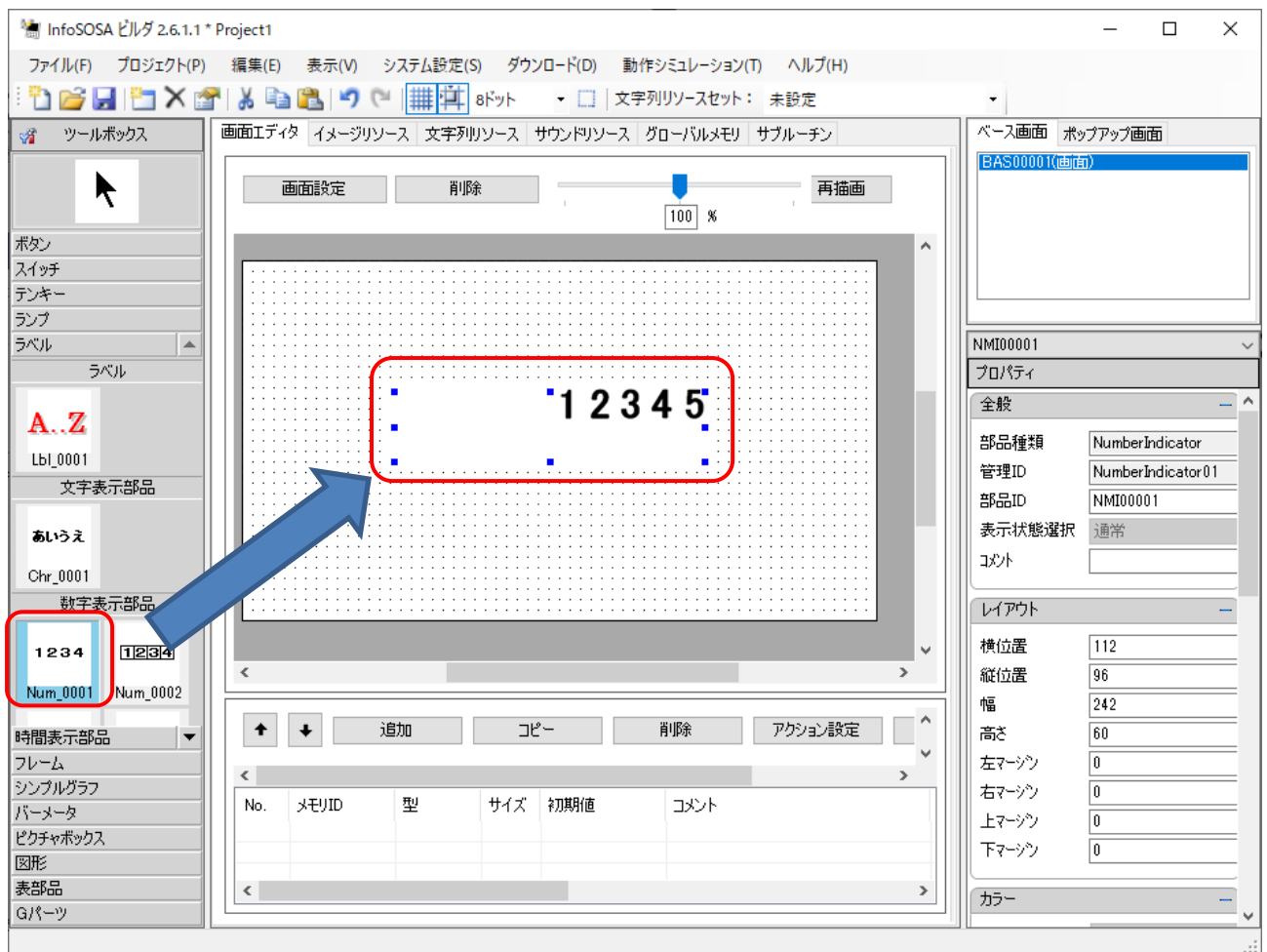


Modbus メモリの値が自動的に Modbus 機器のデバイスの値と同期されます。

### 3.4.1 Modbus 機器のデバイスの値を表示する

Modbus メモリを数字表示部品にリンクすることで表示できます。

1. ツールボックスから数字表示部品を配置します。ドラッグアンドドロップで配置できます。



2. 数値表示部品の詳細プロパティを開きます。ダブルクリックで開きます。

詳細プロパティダイアログ

全般  
部品種類: NumberIndicator 表示状態選択: [ ]  
部品ID: NMI00001 コメント: [ ]

標準プロパティ 拡張プロパティ アクション

レイアウト  
横位置: 112 左マージン: 0  
縦位置: 96 右マージン: 0  
幅: 242 上マージン: 0  
高さ: 60 下マージン: 0

カラー  
文字色: [ ]  
背景色: [ ]  
透過設定: False

リンクデータ  
メモリ種類: [ ]  
メモリID: [ ]  
関係キー: [ ]

データ  
値: 12345  
表示桁数: 10

動作  
有効設定: True  
表示設定: True  
フリック設定: False  
タッチ音: パターン6  
イベント: [ ]  
画面遷移先: [ ]

数字、時間表示  
表示タイプ: 画像イメージ  
数字のイメージ: デフォルト  
半角/全角: 半角

イメージ  
動作イメージ: [ ]  
通常イメージ: [ ]  
機能無効: [ ]

文字列  
文字列: 12345  
文字描画方法: 未設定  
文字フォント: システムフォント  
文字サイズ: 16

水平位置: 右 垂直位置: 上

設定 キャンセル

3. リンクデータに Modbus メモリを指定します。

詳細プロパティダイアログ

全般  
部品種類: NumberIndicator 表示状態選択: [ ]  
部品ID: NMI00001 コメント: [ ]

標準プロパティ 拡張プロパティ アクション

レイアウト  
横位置: 112 左マージン: 0  
縦位置: 96 右マージン: 0  
幅: 242 上マージン: 0  
高さ: 60 下マージン: 0

カラー  
文字色: [ ]  
背景色: [ ]  
透過設定: False

リンクデータ  
メモリ種類: Modbusメモリ  
メモリID: REG00000  
関係キー: [ ]

データ  
値: 0  
表示桁数: 10

動作  
有効設定: True  
表示設定: True  
フリック設定: False  
タッチ音: パターン6  
イベント: [ ]  
画面遷移先: [ ]

数字、時間表示  
表示タイプ: 画像イメージ  
数字のイメージ: デフォルト  
半角/全角: 半角

イメージ  
動作イメージ: [ ]  
通常イメージ: [ ]  
機能無効: [ ]

文字列  
文字列: 0  
文字描画方法: 未設定  
文字フォント: システムフォント  
文字サイズ: 16

水平位置: 右 垂直位置: 上

設定 キャンセル

項目	値
メモリ種類	Modbus メモリ
メモリ ID	REG00000

上記の設定で、この数字表示部品はレジスタの Modbus アドレス「00000」の値を表示します。

メモリ ID が表示する Modbus 機器のアドレスになります。

最初の3文字が REG の場合は「レジスタ」を COI の場合は「コイル」を示します。  
 続く5文字は「Modbus アドレス（10進数表記）」を示します。

例：

メモリ ID	Modbus アドレス(10進数)
REG00010	Modbus アドレス（レジスタ）の 00010
REG00250	Modbus アドレス（レジスタ）の 00250
COI00010	Modbus アドレス（コイル）の 00010

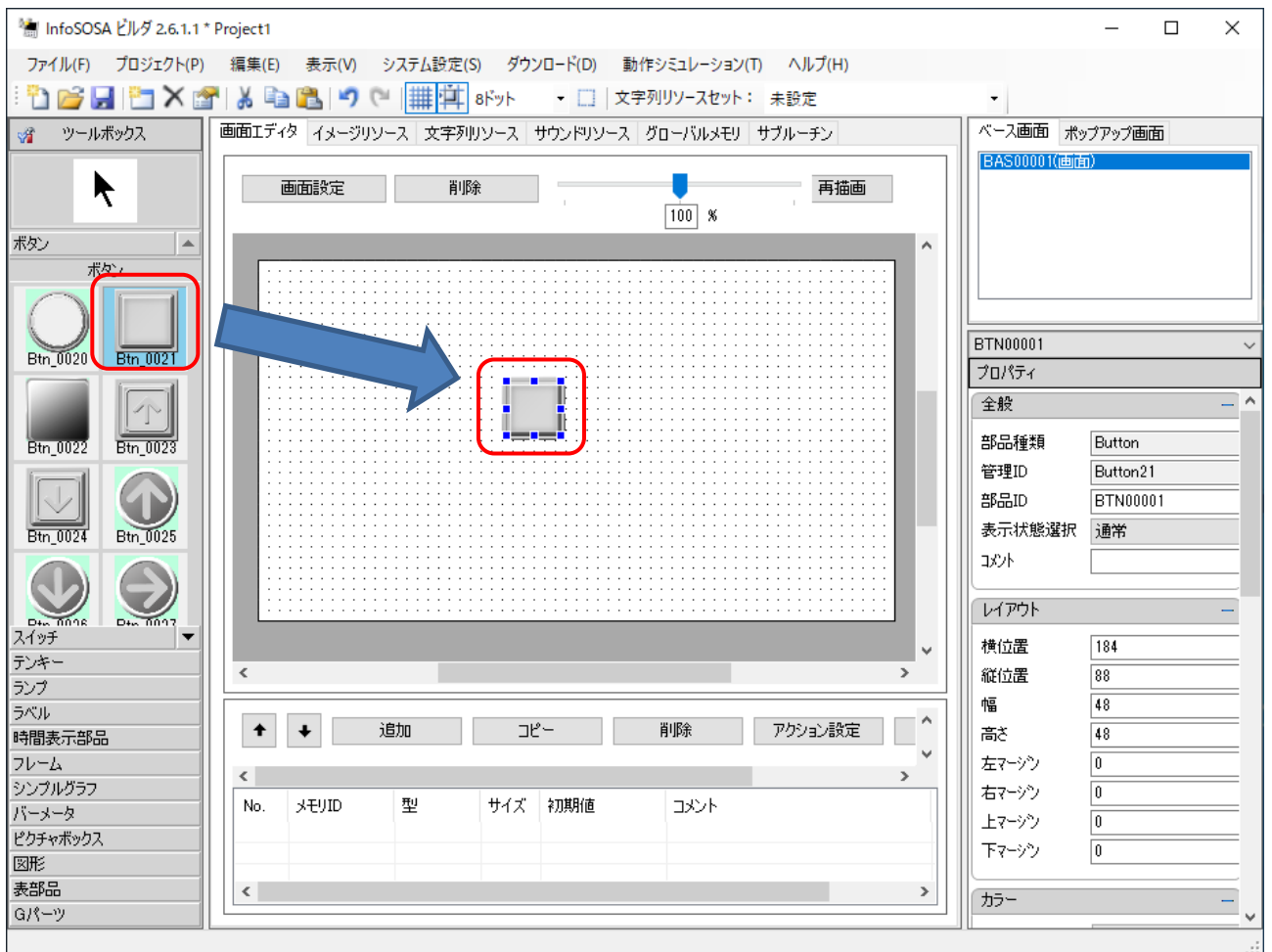
※ Modbus アドレスと Modbus 機器のレジスタの対応については、お使いの Modbus 機器の仕様書をご確認ください。



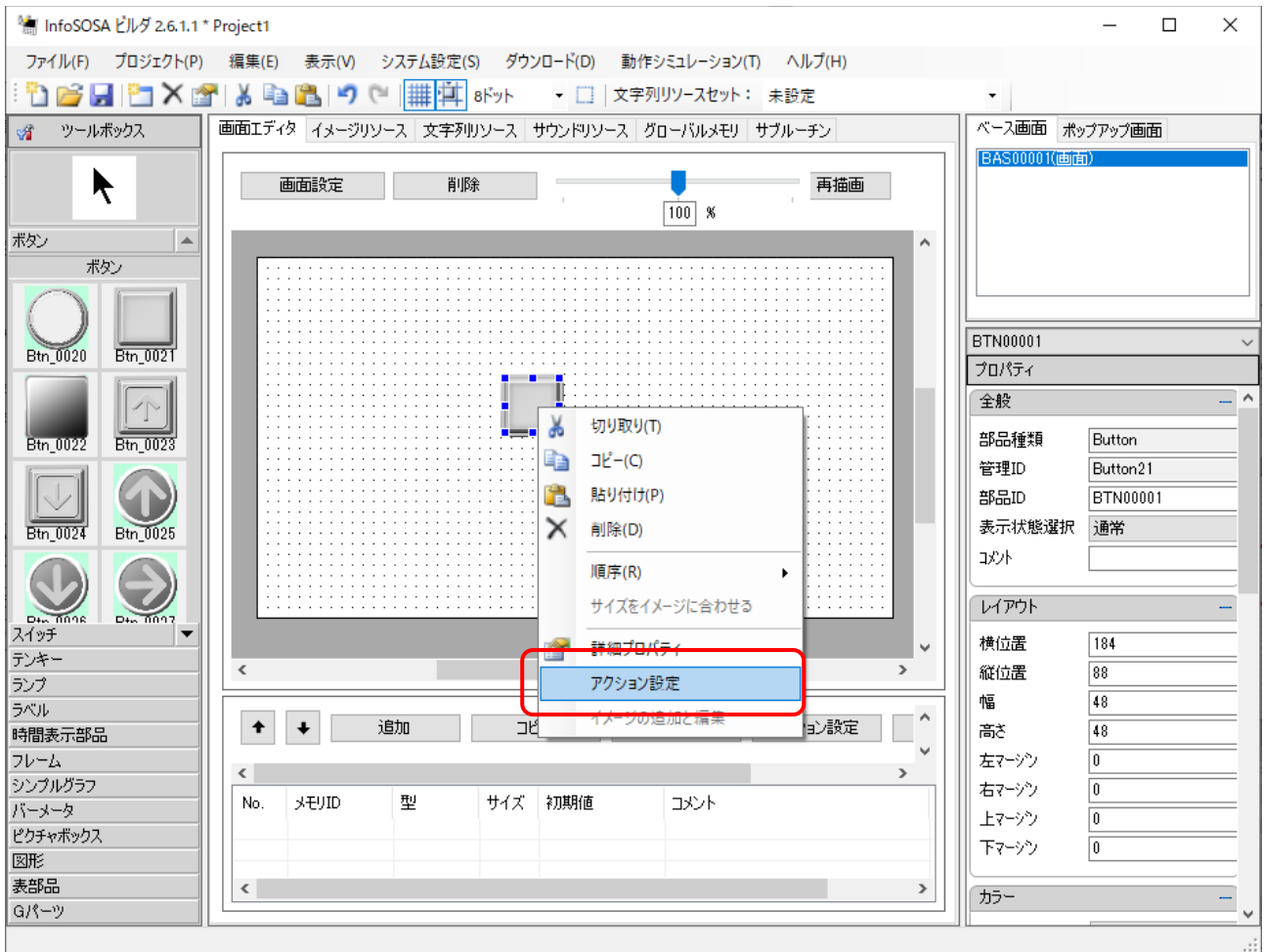
## 3.4.2 Modbus 機器のデバイスに値を書き込む

Modbus メモリに値を設定することで、自動的に Modbus 機器のデバイスに値が書き込まれます。

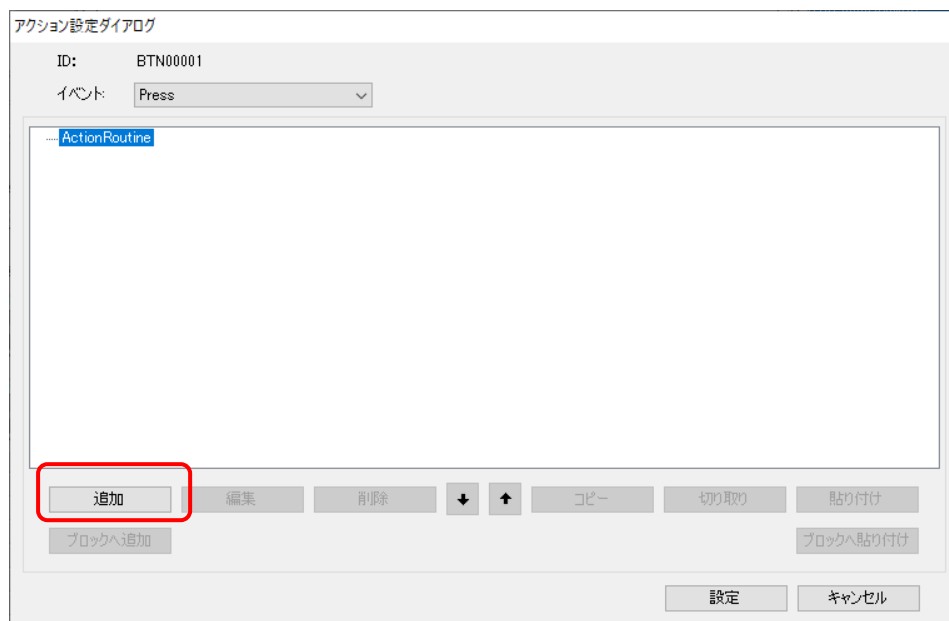
1. ボタンを配置します。ドラッグアンドドロップで配置できます。



2. ボタンのアクション設定をクリックします。ボタンを右クリックすると表示されます。



3. 追加をクリックします。



4. アクショングループに「数値演算」、アクションに「値設定」を選択します。

The screenshot shows the 'アクション追加ダイアログ' (Action Add Dialog) with the following settings:

- アクション**
  - アクショングループ: 数値演算 (checked)
  - アクション: 値設定 (checked)
- パラメータ**
  - メモリ種類: (empty dropdown)
  - メモリID(数値型): (empty dropdown)
  - 設定値: (empty text input)

5. メモリ種類に「Modbusメモリ」、メモリIDに「REG00000」、設定値に「10」を入力します。

The screenshot shows the 'アクション更新ダイアログ' (Action Update Dialog) with the following settings:

- アクション**
  - アクショングループ: 数値演算
  - アクション: 値設定
- パラメータ**
  - メモリ種類: Modbusメモリ
  - メモリID(数値型): REG00000
  - 設定値: 10

上記の設定で、このボタンを押すと Modbus 機器の Modbus アドレス「00000」に 10 を書き込みます。

このように Modbus メモリに値をセットすると、自動的に同期されます。

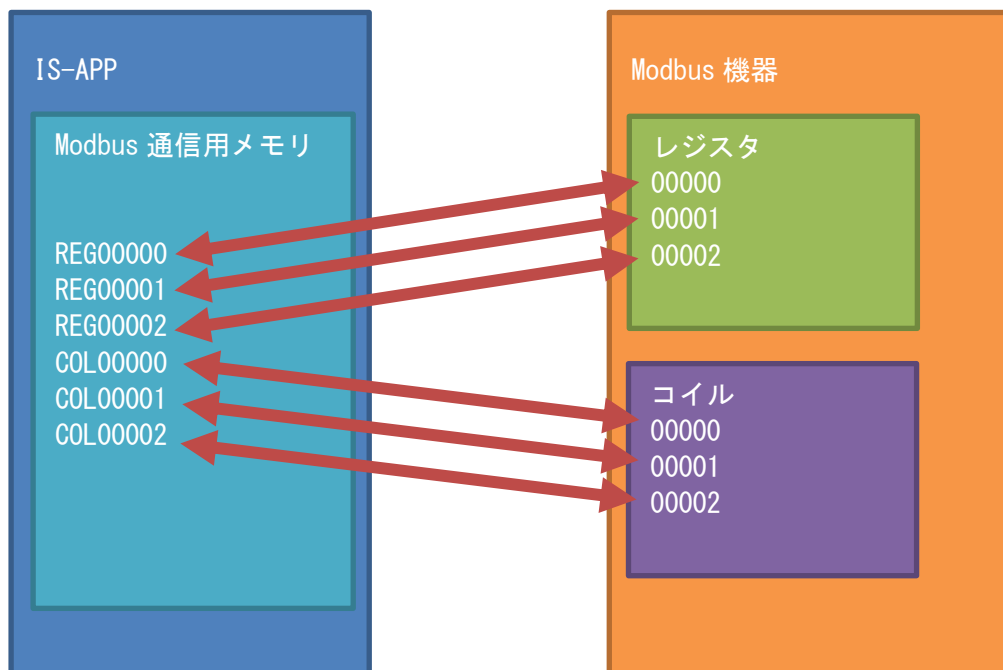
- ※ Modbus アドレスと機器のレジスタアドレスの対応については、お使いの Modbus 対応機器の仕様書をご確認ください。
- ※ 同じ Modbus メモリに短時間に連続で値を書き込んだ場合、負荷を軽減する為、Modbus 機器に書き込まれるのは最終値のみになる場合があります。

# 4章 リファレンス

## 4.1 Modbus通信用メモリ



Modbus 通信用メモリは、指定したアドレスの値と双方向に自動的に同期されます。



常に Modbus 機器側の値を Read しており、Modbus 通信用メモリは Modbus 機器側の指定したアドレスの値を表示します。

Modbus 通信用メモリに値をアクション等で設定すると、Modbus 機器側へ値を Write します。

### 注意

Modbus 通信用メモリの登録数が多いほど、反映に必要な時間は増加します。反映が遅い場合は、優先して同期するメモリ区間を指定できます。

Modbus 通信用メモリの値を変更すると、Read を停止して優先的に Write 処理を行います。

同じ Modbus 通信用メモリに連続で値を書き込んだ場合、Write 処理が行われるのは、最終値のみの場合があります。

## 4.1.1 プロパティ

Modbus 通信用メモリのプロパティについて記載します。

詳細プロパティダイアログ

アドレス:	<input type="text" value="REG00000"/>	型:	<input type="text" value="ワード"/>
初期値:	<input type="text"/>	符号:	<input type="text" value="符号なし"/>
最小値:	<input type="text" value="0"/>	アンダーフロー:	<input type="text" value="保持"/>
最大値:	<input type="text" value="65535"/>	オーバーフロー:	<input type="text" value="保持"/>
		値変化イベント:	<input type="text" value="なし"/>
コメント:	<input type="text"/>		

アクション設定      設定      キャンセル

項目	説明								
アドレス	<p>最初の3文字がREGの場合は「レジスタ」をCOIの場合は「コイル」を示します。</p> <p>続く5文字がModbus アドレス（10進数）を示します。</p> <p>このModbus 機器側のアドレスの値が本メモリと同期します。</p> <p>Modbus アドレスと機器のレジスタアドレスの対応については、お使いのModbus 対応機器の仕様書をご確認ください。</p> <p>1アドレスは2byteです。</p>								
型	<p>レジスタ（REG）の場合は、「バイト」「ワード」「ダブルワード」から選択できます。</p> <p>デフォルトは「ワード」です。</p> <table border="1"> <thead> <tr> <th>型</th> <th>サイズ</th> </tr> </thead> <tbody> <tr> <td>バイト※1</td> <td>1byte</td> </tr> <tr> <td>ワード</td> <td>2byte</td> </tr> <tr> <td>ダブルワード※2</td> <td>4byte</td> </tr> </tbody> </table> <p>※1 バイトに設定した場合は、Modbus 機器の上位データ(1byte)は無視して取得します。</p> <p>※2</p>	型	サイズ	バイト※1	1byte	ワード	2byte	ダブルワード※2	4byte
型	サイズ								
バイト※1	1byte								
ワード	2byte								
ダブルワード※2	4byte								

項目	説明										
	<p>ダブルワードに設定した場合は、次のアドレスを上位データ（2byte）として使用します。（ダブルワードに変更した場合は、自動的に次のアドレスのメモリと統合されます）</p> <p>コイル（COI）の場合は、「ビット」固定です。</p> <table border="1"> <thead> <tr> <th>型</th> <th>サイズ</th> </tr> </thead> <tbody> <tr> <td>ビット</td> <td>1bit</td> </tr> </tbody> </table>	型	サイズ	ビット	1bit						
型	サイズ										
ビット	1bit										
符号	<p>取得したデータを「符号あり」として表示するか「符号なし」として表示するかを選択します。 デフォルトは「符号なし（ワード）」です。</p> <table border="1"> <thead> <tr> <th>型</th> <th>符号</th> </tr> </thead> <tbody> <tr> <td>バイト</td> <td>符号あり／符号なし</td> </tr> <tr> <td>ワード</td> <td>符号あり／符号なし</td> </tr> <tr> <td>ダブルワード</td> <td>符号あり</td> </tr> <tr> <td>ビット</td> <td>符号なし</td> </tr> </tbody> </table>	型	符号	バイト	符号あり／符号なし	ワード	符号あり／符号なし	ダブルワード	符号あり	ビット	符号なし
型	符号										
バイト	符号あり／符号なし										
ワード	符号あり／符号なし										
ダブルワード	符号あり										
ビット	符号なし										
初期値	<p>初回通信時に設定された初期値を Modbus 機器に書き込みます。 未設定の場合は、最初に Modbus 機器のデータを取得します。 デフォルトは「未設定」です。</p>										
最小値	<p>対象の Modbus メモリの最小値が表示されます。 直接編集はできません。 「型」と「符号」のプロパティによって決まります。</p>										
最大値	<p>対象の Modbus メモリの最大値が表示されます。 直接編集はできません。 「型」と「符号」のプロパティによって決まります。</p>										
アンダーフロー	<p>対象の Modbus メモリの最小値より小さな値が設定された場合の動作を定義します。 デフォルトは「保持」です。</p> <table border="1"> <thead> <tr> <th>設定</th> <th>動作</th> </tr> </thead> <tbody> <tr> <td>保持</td> <td>アンダーフローが発生する直前の値を保持します。</td> </tr> <tr> <td>ループ</td> <td>下回った値を最大値から引いた値を対象メモリの値とします。</td> </tr> <tr> <td>クリップ</td> <td>対象メモリに最小値を設定します。</td> </tr> </tbody> </table>	設定	動作	保持	アンダーフローが発生する直前の値を保持します。	ループ	下回った値を最大値から引いた値を対象メモリの値とします。	クリップ	対象メモリに最小値を設定します。		
設定	動作										
保持	アンダーフローが発生する直前の値を保持します。										
ループ	下回った値を最大値から引いた値を対象メモリの値とします。										
クリップ	対象メモリに最小値を設定します。										

項目	説明								
オーバーフロー	<p>対象の Modbus メモリの最大値より大きな値が設定された場合の動作を定義します。 デフォルトは「保持」です。</p> <table border="1" data-bbox="660 360 1331 781"> <thead> <tr> <th data-bbox="660 360 951 407">設定</th> <th data-bbox="951 360 1331 407">動作</th> </tr> </thead> <tbody> <tr> <td data-bbox="660 407 951 533">保持</td> <td data-bbox="951 407 1331 533">オーバーフローが発生する直前の値を保持します。</td> </tr> <tr> <td data-bbox="660 533 951 658">ループ</td> <td data-bbox="951 533 1331 658">上回った値を最小値に足した値を対象メモリの値とします。</td> </tr> <tr> <td data-bbox="660 658 951 781">クリップ</td> <td data-bbox="951 658 1331 781">対象メモリに最大値を設定します。</td> </tr> </tbody> </table>	設定	動作	保持	オーバーフローが発生する直前の値を保持します。	ループ	上回った値を最小値に足した値を対象メモリの値とします。	クリップ	対象メモリに最大値を設定します。
設定	動作								
保持	オーバーフローが発生する直前の値を保持します。								
ループ	上回った値を最小値に足した値を対象メモリの値とします。								
クリップ	対象メモリに最大値を設定します。								
値変化イベント	<p>値変化イベント (On Change Value) を有効にし、アクション設定が行えるようになります。 対象の Modbus メモリの値が変化した時に、任意の処理を実行することができますようになります。</p>								
コメント	備考欄として自由にメモなどを記入できます。								



## 4.1.2 値変化イベント(On Change Value)

値変化イベントを使用すると、Modbus 機器側の値を変えることで、InfoSOSA 側でその値に応じた処理（アクション）を実行することができます。

例えば、特定レジスタの値を変えると表示画面を値に応じた画面に遷移させることが可能です。

対象メモリの「値変化イベント」プロパティを「あり」に設定すると「アクション設定」が可能になります。

イベント ID
ON_CHANGE

詳細プロパティダイアログ

アドレス:	REG00000	型:	ワード
初期値:		符号:	符号なし
最小値:	0	アンダーフロー:	保持
最大値:	65535	オーバーフロー:	保持
コメント:			
	値変化イベント:	あり	

アクション設定      設定      キャンセル

アクション設定ダイアログ

ID: REG00000

イベント: On Change Value

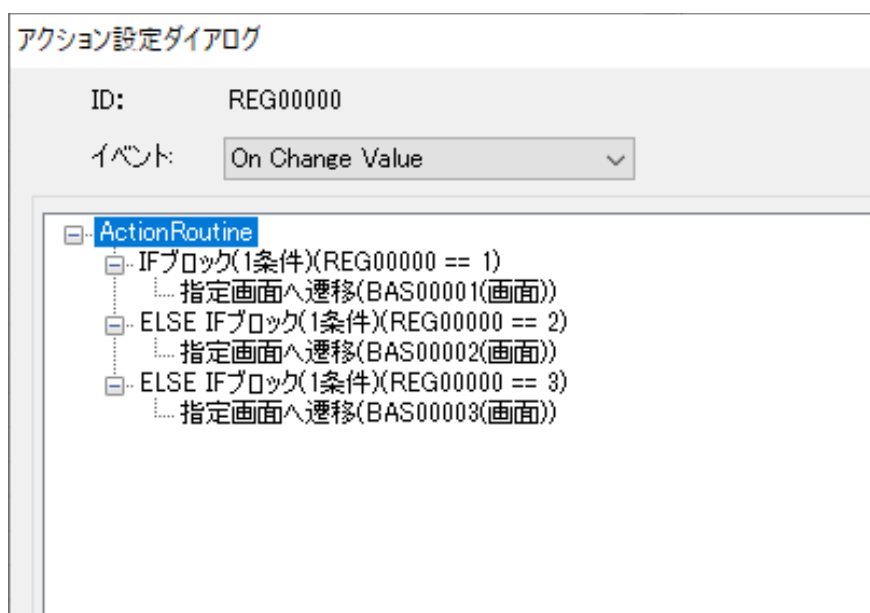
--- ActionRoutine

追加   編集   削除   ↓   ↑   コピー   切り取り   貼り付け

ブロックへ追加      ブロックへ貼り付け

設定      キャンセル

このように設定すると「REG00000」の値に応じて、表示する画面を切り替えることが可能です。



## 注意

値変化イベントを設定したメモリの値を高速で変更すると、値変化イベントが繰り返し発生し、パフォーマンスを与える可能性がありますので、最適な変更頻度になるようご注意ください。

値変化イベント内で、値変化イベントを設定したメモリの値を変更する際は、循環しないようご注意ください。

### 4.1.3 上位通信(InfoSOSA 専用プロトコル)

上位通信コマンドでも Modbus メモリへアクセスすることが可能です。

上位通信コマンドとは、汎用の Modbus プロトコルではなく、InfoSOSA の専用プロトコルで InfoSOSA に値を設定したり、取得したりする機能です。主にマイコンポートとの通信に使用します。

項目	値
所属 ID	@GLBMEM
メモリ ID	アドレス
プロパティ	VALUE

Modbus メモリ「REG00000」に InfoSOSA プロトコルでアクセスする為には、以下のようなコマンドになります。

【設定】

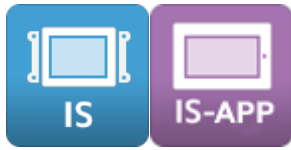
**PA01, @GLBMEM. REG00000. VALUE, 50**

【取得】

**PA02, @GLBMEM. REG00000. VALUE**

上位通信の詳細は、別紙「InfoSOSA リファレンスマニュアル」の「上位通信」の章を参照ください。

## 4.2 Modbus通信用イベント



Modbus 機器通信用イベントの詳細を記載します。

「Modbus」タブの「アクション設定（グローバル）」ボタンから、Modbus 通信がタイムアウトになった時に実行する処理（アクション）、通信が復旧した時の処理（アクション）、初回同期が完了した時の処理（アクション）を設定できます。

画面エディタ イメージリソース 文字列リソース グローバルメモリ シートキー設定 サブルーチン

一般 グループ設定 Modbus

設定のインポート 設定のエクスポート Modbus通信設定

No.	アドレス	型	符号	初期値	コメント
0001	REG00000	* ワード	符号なし		
0002	REG00001	* ワード	符号なし		
0003	REG00002	* ワード	符号なし		
0004	REG00003	* ワード	符号なし		
0005	REG00004	* ワード	符号なし		
0006	REG00005	* ワード	符号なし		
0007	REG00006	* ワード	符号なし		
0008	REG00007	* ワード	符号なし		
0009	REG00008	* ワード	符号なし		
0010	REG00009	* ワード	符号なし		
0011	REG00010	* ワード	符号なし		

追加 削除 アクション設定 アクション設定(グローバル) プロパティ

## 4.2.1 Link Error

Modbus 通信が切断された時に実行する処理（アクション）を設定できます。

### [ご注意]

Modbus 機器は、InfoSOSA からコマンドを受信後 100ms 以内にレスポンスコマンドを送ってください。レスポンスコマンドが遅れると Link Error イベントが発生します。

イベント ID
LINKERROR

アクション設定ダイアログ

ID: LINKERROR

イベント: Link Error

.... Action Routine

このように設定すると、通信エラーが発生した場合に、エラー画面を表示することが可能です。

アクション設定ダイアログ

ID: LINKERROR

イベント: Link Error

☐ Action Routine  
    ... ポップアップ画面Aの表示をON(POPA0001(画面) (X=100, Y=100))

エラー内容は以下の環境変数を参照することで取得可能です。

環境変数 ID
LINK_ERR

この環境変数にエラーコードが設定されます。

エラーコード	プロトコル	カテゴリ	エラー内容
1001	TCP	通信エラー	TCP コネクションエラー、データ通信タイムアウト
1002	TCP	通信エラー	TCP ヘッダ異常
1003	RTU	通信エラー	データ通信タイムアウト
1004	RTU	通信エラー	CRC エラー
1101	共通	データエラー	Modbus 機器側からのエラーレスポンスを受信 ファンクションコードをサポートしていない
1102	共通	データエラー	Modbus 機器側からのエラーレスポンスを受信 指定した Modbus アドレスが範囲外
1103	共通	データエラー	Modbus 機器側からのエラーレスポンスを受信 指定した値が範囲外
1104	共通	データエラー	Modbus 機器側からのエラーレスポンスを受信 その他エラー
1901	共通	その他エラー	RTU 通信時、指定したデバイスファイルが存在 しない その他のエラー

## 4.2.2 Link Recover

「Link Error」の発生後、Modbus 通信が復旧した時に行う処理（アクション）を設定できます。

イベントID

LINKRECOVER

アクション設定ダイアログ

ID: LINKRECOVER

イベント: Link Recover

----- Action Routine

このように設定すると、通信エラーが復旧した時、エラー画面を消すことが可能です。

アクション設定ダイアログ

ID: LINKRECOVER

イベント: Link Recover

Action Routine  
└─ ポップアップ画面Aの表示をOFF

通信復帰後は、完了していない変更処理が実施されます。

## 4.2.3 First Sync Complete

Modbus メモリの初回同期が完了した時に行う処理（アクション）を設定できます。  
Modbus メモリが1つも登録されていない場合は、本イベントは発生しません。  
2回目以降の同期完了時は発生しません。

イベント ID
FSYNCCOMP

アクション設定ダイアログ

ID: FSYNCCOMP

イベント: First Sync Complete

----- Action Routine

起動時はローディング画面を表示し、読み込み完了後に自動的に最初の画面に遷移するような設定が可能です。

アクション設定ダイアログ

ID: FSYNCCOMP

イベント: First Sync Complete

☐ Action Routine  
└─ 指定画面へ遷移(TOP(画面))



## 4.3 IS-APPコマンドライン引数



### Modbus

IS-APP の Modbus プロトコル通信機能を有効にします。

本引数で Modbus プロトコル通信機能を有効にしていない場合は、Modbus 機器との通信は行われません。

#### 引数名

-u または --modbus

#### 書式

Modbus/TCP マスターの場合：※デフォルト

-u tcp <通信先アドレス> <通信先ポート> <局番>

Modbus/RTU マスターの場合：

-u rtu <デバイスファイル> <局番> <通信規格> <通信速度> <パリティ>

パラメータ	内容
<通信先アドレス>	通信する Modbus 機器の IP アドレスを指定します。(10 進数) デフォルト：192.168.0.100
<通信先ポート>	通信する Modbus 機器のポート番号を指定します。(10 進数) 0~65535 デフォルト値：502
<デバイスファイル>	シリアルポートのデバイスファイルを指定します。 デフォルト：SIO2 のデバイスファイルが自動設定されます SIO1：/dev/com1 SIO2：/dev/com2
<局番>	通信する Modbus 機器の局番を指定します。(10 進数) 0~255 デフォルト：1
<通信規格>	シリアルインタフェースの通信規格を指定します。 RS232C：rs232c RS422：rs422 RS485：rs485 ※デフォルト

パラメータ	内容
<通信速度>	通信速度を指定します。 4800 : 4800bps 9600 : 9600bps 19200 : 19200bps 38400 : 38400bps 57600 : 57600bps 115200 : 115200bps ※デフォルト
<パリティ>	パリティを指定します。 none : パリティ無し ※デフォルト odd : 奇数パリティ even : 偶数パリティ

例 :

Modbus/TCP マスターの場合 :

```
is_app -r /mnt/user/data -u tcp 192.168.0.100 502 1
```

Modbus/RTU マスターの場合 :

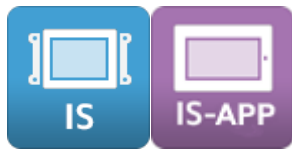
```
is_app -r /mnt/user/data -u rtu /dev/com2 1 rs485 115200 none
```

## デフォルト値

省略した場合は以下の設定になります。

パラメータ	内容
有効/無効	無効

## 4.4 Modbus設定ファイル



### 4.4.1 csv 形式 (個別設定)

エクスポートすると csv 形式で保存されます。  
Modbus メモリを設定を個別に設定できます。

サンプル:

```
:ModbusType, ReadMaxNum, WriteMaxNum
RegisterMaxData, 16, 16
CoilMaxData, 16, 16
:PriorityAddress, StartAddress, EndAddress
PriorityRegister, 0, 1
PriorityCoil, 0, 1
:NormalAddress, WaitTime
NormalCommon, 100
:ModbusType, Address, DataType, InitialValue, Signed, UnderFlow, OverFlow, Comment
Register, 00000, Byte, 10, Unsigned, Retention, Retention, 状態 1
Register, 00001, Word, , Unsigned, Loop, Loop, 温度
Register, 00002, DWord, , Signed, Clip, Clip, 回転数
Coil, 00000, Bit, 1, Unsigned, Retention, Retention, スタート
Coil, 00001, Bit, , Unsigned, Retention, Retention, 状態 2
Coil, 00002, Bit, , Unsigned, Retention, Retention, 緊急停止
```

※セミコロンから始まる行はコメント行になります。

※文字コードは UTF-8 になります。

#### 1 行目 (最大データ数設定・レジスタ)

パラメータ	値	説明
ModbusType	RegisterMaxData	固定値
ReadMaxNum	1 ~ 接続する Modbus 機器の最大値	1 回の読み処理で読み可能な最大のレジスタ数を 10 進数で記載します。 値を増やすほどパフォーマンスは向上しますが、接続する Modbus 機器側の最大値を超えると通信が行えなくなります。
WriteMaxNum	1 ~ 接続する Modbus 機器の最大値	1 回の書き処理で書き可能な最大のレジスタ数を 10 進数で記載します。 値を増やすほどパフォーマンスは向上しますが、接続する Modbus 機器側の最大値を超えると通信が行えなくなります。

パラメータ	値	説明

## 2行目（最大データ数設定・コイル）

パラメータ	値	説明
ModbusType	CoilMaxData	固定値
ReadMaxNum	1～ 接続する Modbus 機器の最大値	1 回の読み処理で読み可能な最大のコイル数を 10 進数で記載します。 値を増やすほどパフォーマンスは向上しますが、接続する Modbus 機器側の最大値を超えると通信が行えなくなります。
WriteMaxNum	1～ 接続する Modbus 機器の最大値	1 回の書き処理で書き可能な最大のコイル数を 10 進数で記載します。 値を増やすほどパフォーマンスは向上しますが、接続する Modbus 機器側の最大値を超えると通信が行えなくなります。

## 3行目（優先区間設定・レジスタ）

パラメータ	値	説明
PriorityAddress	PriorityRegister	固定値
StartAddress	6 行目以降で登録するレジスタのアドレス範囲の値	優先区間に設定するレジスタの開始アドレスと終了アドレスを記載します。
EndAddress		値は StartAddress ≤ EndAddress である必要があります。 StartAddress と EndAddress を両方入力しない場合は、レジスタは全て通常区間になります。

## 4行目（優先区間設定・コイル）

パラメータ	値	説明
PriorityAddress	PriorityCoil	固定値
StartAddress	6 行目以降で登録するコイルのアドレス範囲の値	優先区間に設定するコイルの開始アドレスと終了アドレスを記載します。
EndAddress		値は StartAddress ≤ EndAddress である必要があります。 StartAddress と EndAddress を両方入力しない場合は、コイルは全て通常区間になります。

### 5行目（通常区間設定・レジスタ/コイル共通）

パラメータ	値	説明
NormalAddress	NormalCommon	固定値
WaitTime	1~3599999	通常区間の待機時間を記載します。単位はミリ秒です。※本設定は同期時間ではございません。実際の同期時間は、通常区間に設定したメモリの数に比例します。（待機時間経過に読み込みを開始するため、設定値以上になります）

### 6行目以降（Modbus メモリ設定）

パラメータ	値	説明								
ModbusType	Register Coil	<p>レジスタ用通信メモリを作成する場合は「Register」と定義してください。</p> <p>「Register」と定義した場合は、Modbus 機器に対する通信は以下のファンクションコードで行います。 <u>Read:0x03 Write:0x10</u></p> <p>コイル用通信メモリを作成する場合は「Coil」と定義してください。</p> <p>「Coil」と定義した場合は、Modbus 機器に対する通信は以下のファンクションコードで行います。 <u>Read:0x01 Write:0x0F</u></p>								
Address	0~ 接続する Modbus 機器の最大値	このメモリの同期する Modbus アドレスを 10 進数で記載します。								
DataType	Byte Word DWord Bit	<p>ModbusType が「Register」の場合は、「Byte」「Word」「DWord」から選択できます。</p> <table border="1" data-bbox="916 1507 1441 1697"> <thead> <tr> <th>型</th> <th>サイズ</th> </tr> </thead> <tbody> <tr> <td>Byte<sup>※1</sup></td> <td>1byte</td> </tr> <tr> <td>Word</td> <td>2byte</td> </tr> <tr> <td>DWord<sup>※2</sup></td> <td>4byte</td> </tr> </tbody> </table> <p>※1 Byte に設定した場合は、Modbus 機器の上位データ(1byte)は無視して取得します。</p> <p>※2</p>	型	サイズ	Byte <sup>※1</sup>	1byte	Word	2byte	DWord <sup>※2</sup>	4byte
型	サイズ									
Byte <sup>※1</sup>	1byte									
Word	2byte									
DWord <sup>※2</sup>	4byte									

パラメータ	値	説明										
		<p>DWord に設定した場合は、次のアドレスを上位データ (2byte) として使用します。(次のアドレスは使用できません)</p> <p>ModbusType が「Coil」の場合は、「Bit」固定です。</p> <table border="1"> <thead> <tr> <th>型</th> <th>サイズ</th> </tr> </thead> <tbody> <tr> <td>Bit</td> <td>1bit</td> </tr> </tbody> </table>	型	サイズ	Bit	1bit						
型	サイズ											
Bit	1bit											
InitialValue	Data Type で指定した型の範囲内	<p>このメモリの初期値を設定します。</p> <p>初期値を設定した場合、初回通信時に設定された初期値を Modbus 機器に書き込みます。</p> <p>未設定の場合は、最初に Modbus 機器のデータを取得します。</p>										
Signed	Unsigned Signed	<p>取得したデータを「符号あり(Signed)」として表示するか「符号なし(Unsigned)」として表示するかを選択します。</p> <table border="1"> <thead> <tr> <th>型</th> <th>符号</th> </tr> </thead> <tbody> <tr> <td>Byte</td> <td>Signed/Unsigned</td> </tr> <tr> <td>Word</td> <td>Signed/Unsigned</td> </tr> <tr> <td>DWord</td> <td>Signed</td> </tr> <tr> <td>Bit</td> <td>Unsigned</td> </tr> </tbody> </table>	型	符号	Byte	Signed/Unsigned	Word	Signed/Unsigned	DWord	Signed	Bit	Unsigned
型	符号											
Byte	Signed/Unsigned											
Word	Signed/Unsigned											
DWord	Signed											
Bit	Unsigned											
UnderFlow	Retention Loop Clip	<p>このメモリの最小値より小さな値が設定された場合の動作を定義します。</p> <table border="1"> <thead> <tr> <th>設定</th> <th>動作</th> </tr> </thead> <tbody> <tr> <td>Retention</td> <td>アンダーフローが発生する直前の値を保持します。</td> </tr> <tr> <td>Loop</td> <td>下回った値を最大値から引いた値を対象メモリの値とします。</td> </tr> <tr> <td>Clip</td> <td>対象メモリに最小値を設定します。</td> </tr> </tbody> </table>	設定	動作	Retention	アンダーフローが発生する直前の値を保持します。	Loop	下回った値を最大値から引いた値を対象メモリの値とします。	Clip	対象メモリに最小値を設定します。		
設定	動作											
Retention	アンダーフローが発生する直前の値を保持します。											
Loop	下回った値を最大値から引いた値を対象メモリの値とします。											
Clip	対象メモリに最小値を設定します。											

パラメータ	値	説明								
OverFlow	Retention Loop Clip	<p>このメモリの最大値より大きな値が設定された場合の動作を定義します。</p> <table border="1"> <thead> <tr> <th>設定</th> <th>動作</th> </tr> </thead> <tbody> <tr> <td>Retention</td> <td>オーバーフローが発生する直前の値を保持します。</td> </tr> <tr> <td>Loop</td> <td>上回った値を最小値に足した値を対象メモリの値とします。</td> </tr> <tr> <td>Clip</td> <td>対象メモリに最大値を設定します。</td> </tr> </tbody> </table>	設定	動作	Retention	オーバーフローが発生する直前の値を保持します。	Loop	上回った値を最小値に足した値を対象メモリの値とします。	Clip	対象メモリに最大値を設定します。
設定	動作									
Retention	オーバーフローが発生する直前の値を保持します。									
Loop	上回った値を最小値に足した値を対象メモリの値とします。									
Clip	対象メモリに最大値を設定します。									
Comment	256 文字以内	このメモリに対しての説明を自由に記述することができます。文字コードは UTF-8 になります。								

## 4.4.2 ini 形式 (一括設定)

インポート専用の形式です。複数の Modbus メモリをまとめて登録できます。  
ini 形式で登録した Modbus メモリの初期値は以下のデフォルト値で登録されます。

項目	値
型	ワード
符号	符号なし
初期値	未設定
アンダーフロー	保持
オーバーフロー	保持
優先区間設定	無し

サンプル:

```
:Section definition
[Section]
SectionNum=3
SectionName1=Register1
SectionName2=Register2
SectionName3=Coil1

:Section1 (Register)
[Register1]
Address=0, 10
MaxWriteData=16
MaxReadData=16

:Section2 (Register)
[Register2]
Address=30, 40

:Section3 (Coil)
[Coil1]
Address=0, 15
MaxWriteData=16
MaxReadData=16
```

※セミコロンから始まる行はコメント行になります。



[Section]

パラメータ	値	説明
SectionNum	1～	セクションの総数を指定します。 セクションを追加、削除する場合は、この値を修正してください。
SectionName1～  添字は 1 から連番で記述してください 例： SectionName1 SectionName2 SectionName3	Register1～ Coil1～  添字は 1 から連番で記述してください 例： Register1 Register2 Coil1 Coil2	セクション名を定義します。  レジスタ用通信メモリを作成する場合は「Register」と定義してください。  「Register」と定義した場合は、Modbus 機器に対する通信は以下のファンクションコードで行います。 <u>Read:0x03 Write:0x10</u>  コイル用通信メモリを作成する場合は「Coil」と定義してください。  「Coil」と定義した場合は、Modbus 機器に対する通信は以下のファンクションコードで行います。 <u>Read:0x01 Write:0x0F</u>

[Register1]

パラメータ	値	説明
Address	[開始 Modbus アドレス],[終了 Modbus アドレス]	登録するレジスタの開始 Modbus アドレス、終了 Modbus アドレスを 10 進数で記載します。  ※登録可能な Modbus 機器との通信用メモリ（レジスタ、コイル合計）はグローバルメモリと合わせて最大 2000 個になります。
MaxWriteData	1～ 接続する Modbus 機器の最大値	1 回の書込処理で書込可能な最大のレジスタ数を 10 進数で記載します。 値を増やすほどパフォーマンスは向上しますが、接続する Modbus 機器側の最大値を超えると通信が行えなくなります。

パラメータ	値	説明
MaxReadData	1～ 接続する Modbus 機器の最大値	1 回の読み処理で読み可能な最大のレジスタ数を 10 進数で記載します。 値を増やすほどパフォーマンスは向上しますが、接続する Modbus 機器側の最大値を超えると通信が行えなくなります。

[Register2]

パラメータ	値	説明
Address	0,10	登録する Modbus アドレスが連続していない場合は、別セクションとして記述してください。  「Register」の MaxWriteData と MaxReadData は[Register1]のセクションで設定していますので、2つ目のセクションでは不要です。  ※登録可能な Modbus 機器との通信用メモリ（レジスタ、コイル合計）はグローバルメモリと合わせて最大 2000 個になります

[Coil1]

パラメータ	値	説明
Address	[開始 Modbus アドレス],[終了 Modbus アドレス]	登録するコイルの開始 Modbus アドレス、終了 Modbus アドレスを 10 進数で記載します。  ※登録可能な Modbus 機器との通信用メモリ（レジスタ、コイル合計）はグローバルメモリと合わせて最大 2000 個になります
MaxWriteData	1～ 接続する Modbus 機器の最大値	1 回の書き処理で書き可能な最大のコイル数を 10 進数で記載します。 値を増やすほどパフォーマンスは向上しますが、接続する Modbus 機器側の最大値を超えると通信が行えなくなります。
MaxReadData	1～ 接続する Modbus 機器の最大値	1 回の読み処理で読み可能な最大のコイル数を 10 進数で記載します。 値を増やすほどパフォーマンスは向上しますが、接続する Modbus 機器側の最大値を超えると通信が行えなくなります。

# お問い合わせ

---

本書に関するお問い合わせは、下記へお願い致します。

## お電話でのお問い合わせ

 **06-6147-6645**

株式会社ディ・エム・シー 大阪技術センター

受付時間：平日 9：00～17：00

※土日・祝祭日・年末年始を除く

## メールでのお問い合わせ

お問い合わせフォームで受け付けています。下記からご連絡ください。



[www.dush.co.jp/contact/](http://www.dush.co.jp/contact/)

## よくあるご質問と回答集



[www.dush.co.jp/support/faq/](http://www.dush.co.jp/support/faq/)

---

2023年12月 第10版

発行所 株式会社ディ・エム・シー

〒108-0074 東京都港区高輪 2-18-10 高輪泉岳寺駅前ビル 11F

TEL：(03)-6721-6731 (代) FAX：(03)-6721-6732

URL：https://www.dush.co.jp/

本製品及び本書は著作権法によって保護されていますので、無断で複写、複製、転載、改変する事は禁じられています。