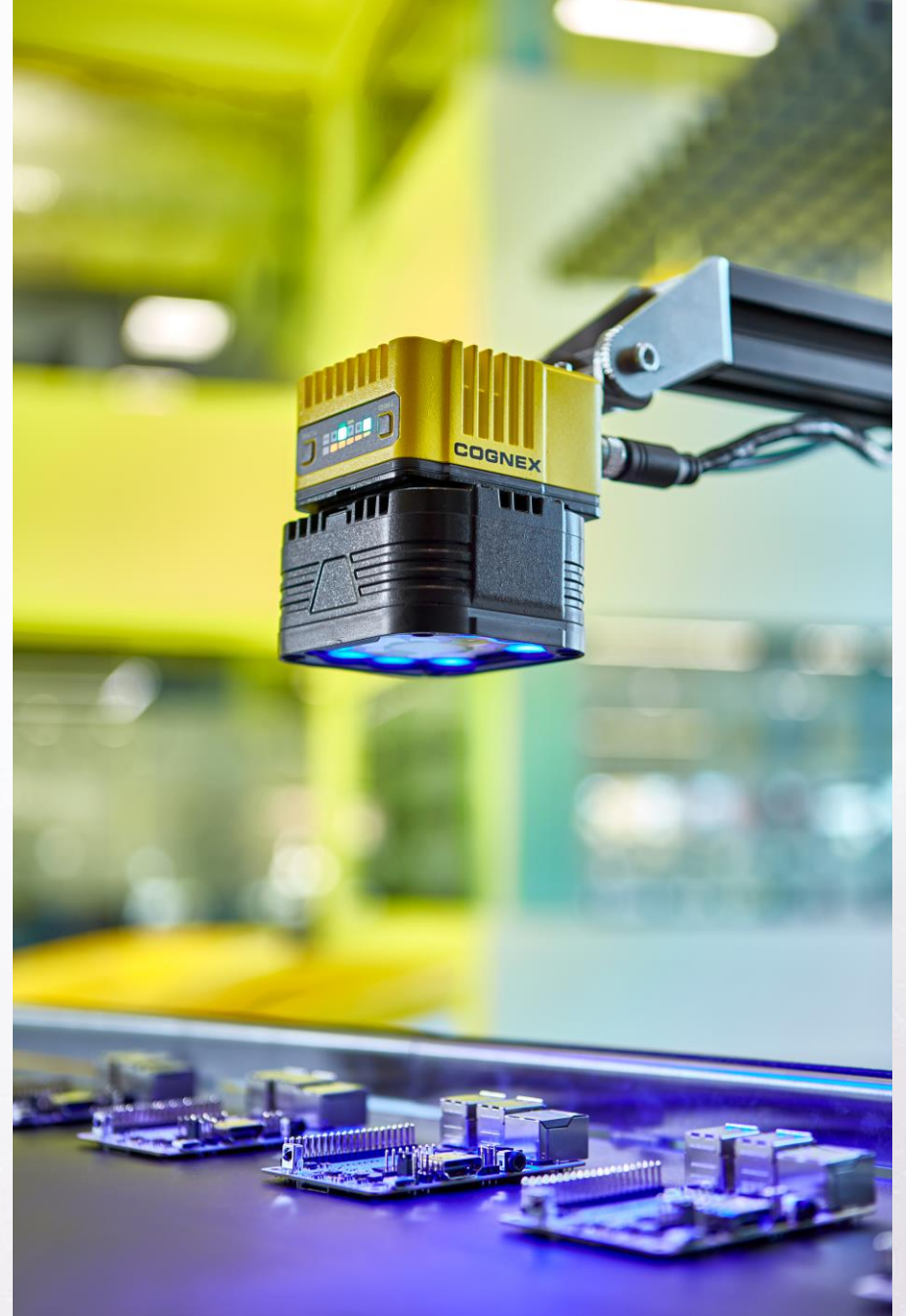


In-Sight Vision Suite

SLMP通信設定

Lead with Speed



改訂履歴

[illegible]

はじめに

- 本資料はコグネックス製マシンビジョンカメラと三菱電機製PLCのSLMP通信を行うことを目的にしています。
- 本資料では下記機種とソフトウェアで動作確認を行っております。

	ハードウェア	ソフトウェア
コグネックス	In-Sight 2800 In-Sight 3800	In-Sight Vision Suite
三菱電機	MELSEC iQ-R MELSEC Q	GX Works 3 GX Works 2

- 別の機種を使用される場合は一部設定が異なる場合がありますのでご了承ください。

目次

- 機器構成
- In-Sight Vision Suite 通信設定
- GX Works 3 通信設定
- GX Works 2 通信設定
- ジョブ切り替え
- トリガ毎のデータ送受信ジョブ作成 EasyBuilder編
- トリガ毎のデータ送受信ジョブ作成 Spreadsheet編

機器構成

The background of the slide is a dark, deep blue space filled with numerous small, glowing particles and stars. In the center, there is a bright, multi-colored sphere (yellow, green, blue, and red) that appears to be the source of a series of concentric, glowing rings or orbits that expand outwards. The overall effect is one of dynamic energy and technological sophistication.

本資料で使用した機器

- Cognex In-Sight 3800

- In-Sight 3800本体
- Ethernet通信ケーブル
- 電源ケーブル

IS3816MP-00000-SA
CCB-84901-2001-05
CCB-PWRIO-05



- 三菱電機 MELSEC iQ-R

- CPUユニット
(電源ユニット
基本ベースユニット)

R04CPU
R61P)
R33B)



- 市販品

- PC
- スイッチングハブ
- DC24V電源
- Ethernetケーブル(Cat6A)
- USB Mini-Bケーブル



本資料で使用した機器

- Cognex In-Sight 2800

- In-Sight 2800本体
- Ethernet通信ケーブル
- 電源ケーブル

IS2801M-24520-EA
CCB-84901-2001-05
CCB-PWRIO-05



- 三菱電機 MELSEC Q

- CPUユニット
(電源ユニット
基本ベースユニット)

Q04UDEHCPU
Q61SP)
Q32SB)



- 市販品

- PC
- スイッチングハブ
- DC24V電源
- Ethernetケーブル(Cat6A)
- USB Mini-Bケーブル



ソフトウェアインストール

- 事前に下記ソフトウェアをお使いのPCにダウンロードしてください。

Cognex In-Sight Vision Suite(最新版は下記リンクから)

[サポート - In-Sight ソフトウェア & ファームウェア | Cognex](#)

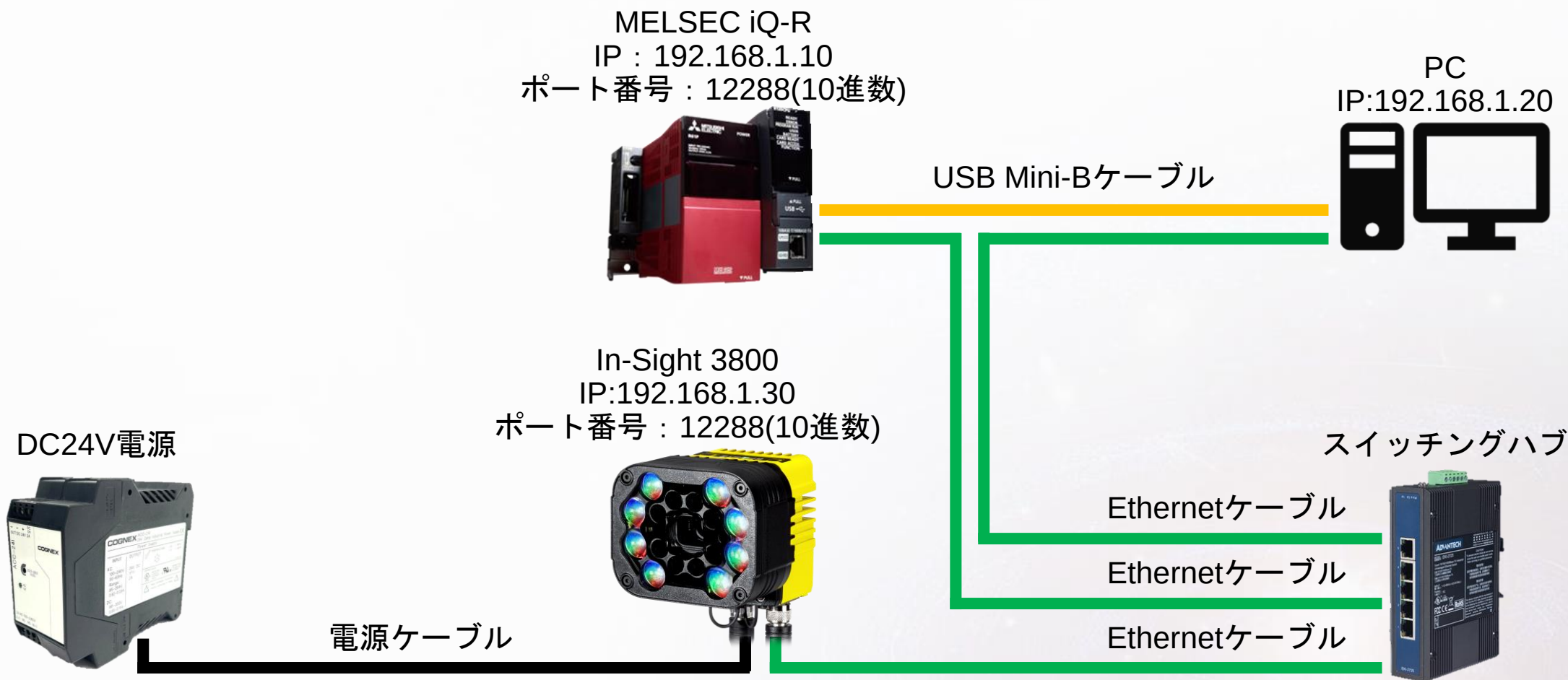


三菱電機 GX Works 3 / GX Works 2



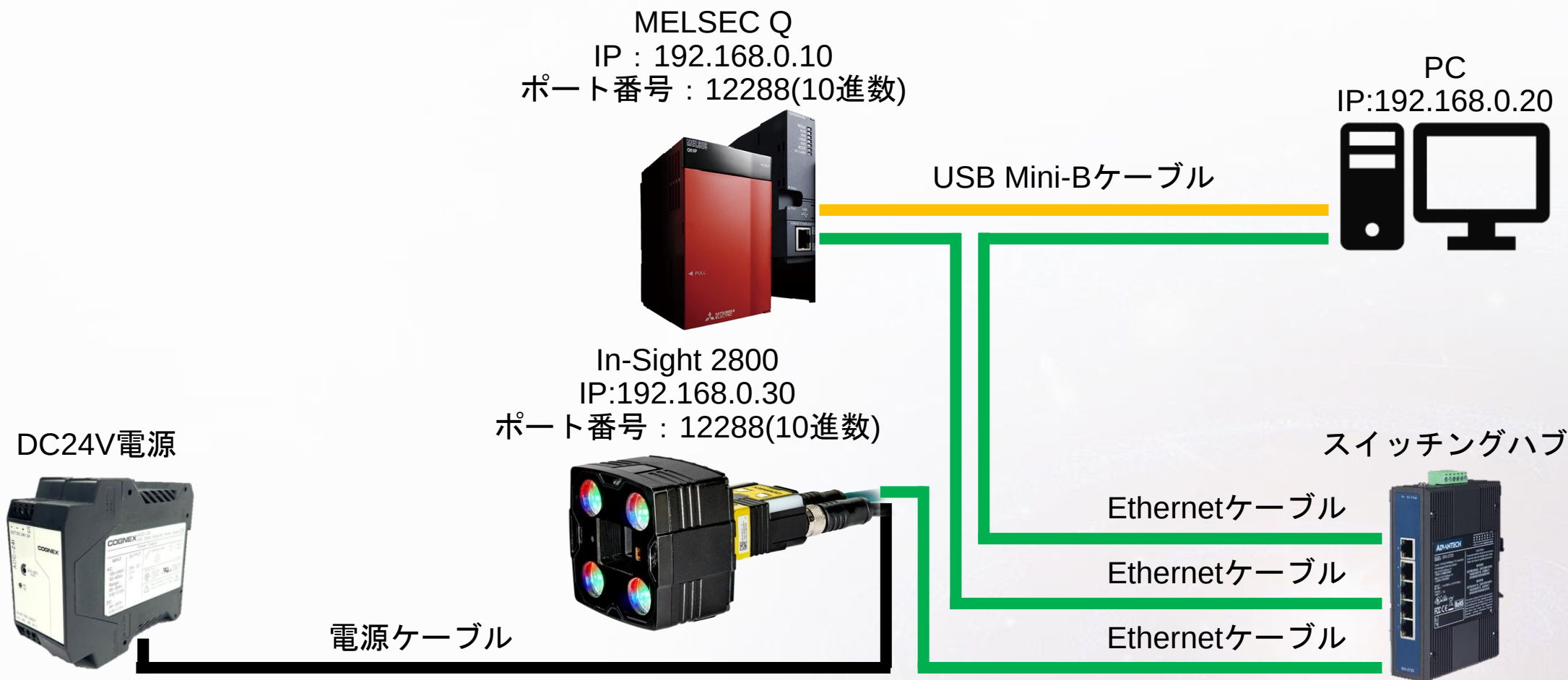
機器接続例

本資料は下図のような設定で機器接続を行います。この通りに設定して動作を確認してからお使いの機器構成に合わせて調整することを推奨いたします。



機器接続例

本資料は下図のような設定で機器接続を行います。この通りに設定して動作を確認してからお使いの機器構成に合わせて調整することを推奨いたします。





In-Sight Vision Suite

通信設定

Summary

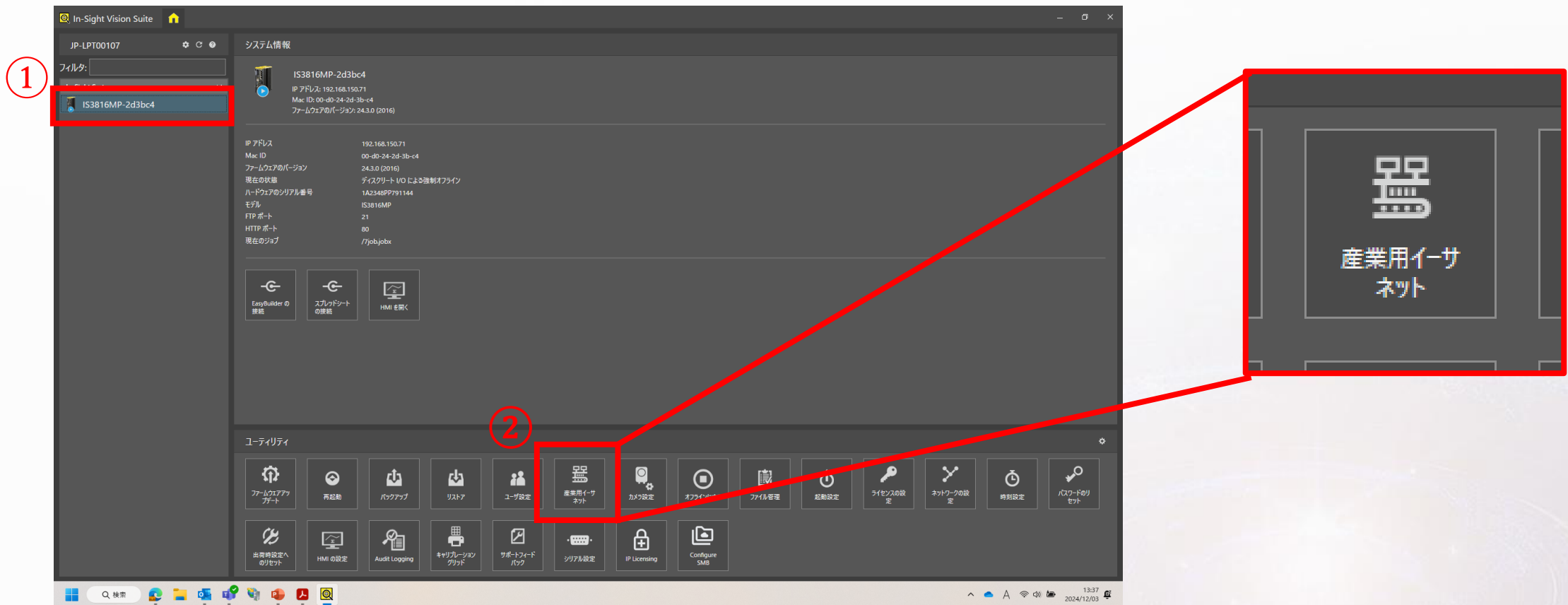
1. [産業用イーサネット設定](#)
2. [プロトコル選択](#)
3. [SLMP設定](#)
4. [データブロック](#)
5. [各ブロック詳細](#)

※お使いのPCのIPアドレス設定、ビジョンシステムのIPアドレス設定およびPCとの接続設定については下記資料をご参照ください。

[In-Sight 2800ネットワーク接続](#)

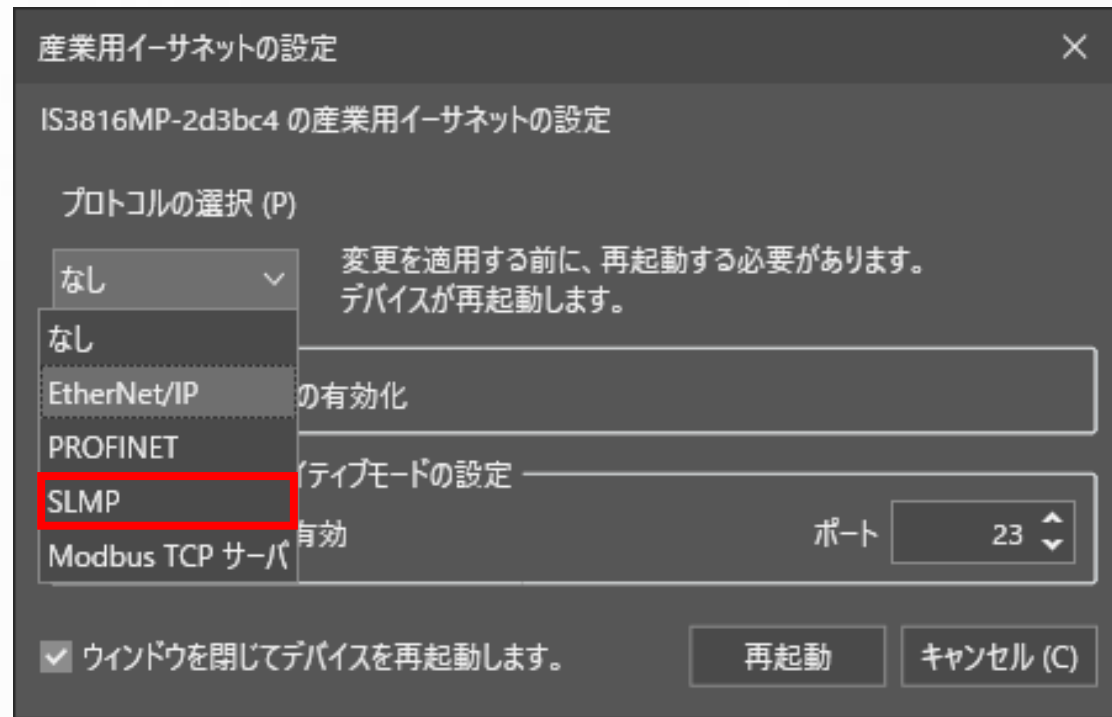
1. 産業用イーサネット設定

In-Sight Vision Suiteを立ち上げ、ビジョンセンサ(今回はIn-Sight 3800)を選択し、ユーティリティから産業用イーサネット設定を開きます。



2. プロトコル選択

プロトコルの選択(P)にて、「SLMP」を選択します。



3. SLMP設定1/4

[設定]タブの以下の項目を設定します。

- コントローラタイプ : iQ-Qシリーズ (3Eフレーム)

※iQ-Qシリーズ(3Eフレーム) : iQ-R, iQ-F, QシリーズPLC使用時に選択してください。FXシリーズ(1Eフレーム):FXシリーズPLC使用時に選択してください。

- IPアドレス : 192.168.1.10

※SLMP通信相手先PLCのIPアドレスを入力してください。本資料ではPLCのIPアドレスを192.168.1.10に設定しています。

産業用イーサネットの設定

IS3816MP-2d3bc4 の産業用イーサネットの設定

プロトコルの選択 (P)

SLMP

プロトコル固有の設定

設定

デバイスのアドレス指定

コントローラタイプ: iQ-Q シリーズ (3E フレーム)

IP アドレス: 192.168.1.10

ホストポート: 10 進数 12288

タイムアウト (ms): 1000

ポーリング間隔 (ms): 100

デフォルトにリセット

リモートステーションに接続:

ネットワーク番号: 0

PC 番号: 0x FF

優先モジュール: Local Station

OPC UA 設定

OPC-UA サーバの有効化

Telnet 経由でのネイティブモードの設定

Telnet サーバが有効

ポート: 23

OK キャンセル (C)

3. SLMP設定2/4

[設定]タブの以下の項目を設定します。

- ホストポート：12288(10進数)

※PLCとのSLMP通信に使用するポート番号を指定します。

本資料では12288(10進数) = 3000(16進数)で設定を行います。

- タイムアウト：1000(ms)

※カメラからPLCへの問い合わせに対するPLCの応答を待つ最大時間です。

本資料では1000 msで設定を行います。

3. SLMP設定3/4

[設定]タブの以下の項目を設定します。

- ポーリング間隔：100(ms)

※カメラがPLCのデバイス値を確認する時間の間隔です。

ポーリング間隔は、5 ～ 30000 ms で設定でき、カメラからPLCへ送信可能なデータ量は、使用するPLCのタイプによって異なります。

iQ/Q シリーズ: 最大 960 ワード (15360 ビット)、またはビットデバイスについては 3584 ビット。

FX シリーズ: 最大 64 ワード (1024 ビット)、またはビットデバイスについては 256 ビット。

ポーリング間隔を短くすればするほど、より高速に通信を行うことができますが、一方でPLCに多大な負荷がかかるためご注意ください。

本資料では100 msで設定を行います。

産業用イーサネットの設定

IS3816MP-2d3bc4 の産業用イーサネットの設定

プロトコルの選択 (P)

SLMP

プロトコル固有の設定

設定 デバイスのアドレス指定

コントローラタイプ: iQ-Q シリーズ (3E フレーム)

IP アドレス: 192.168.1.10

ポート: 10 進数 12288

タイムアウト (ms): 1000

ポーリング間隔 (ms): 100

リモートステーションに接続:

ネットワーク番号: 0

PC 番号: 0x FF

登録モジュール: LocalStation

デフォルトにリセット

OPC UA設定

☐ OPC-UA サーバの有効化

Telnet 経由でのネイティブモードの設定

☒ Telnet サーバが有効

ポート: 23

OK キャンセル (C)

3. SLMP設定4/4

①[デバイスのアドレス指定]タブの②以下の項目を設定します。各ブロックの詳細については、次ページ以降をご覧ください。

名前	選択中のデバイス	オフセット	デバイス数
コントロール	M - 内部	0	32ビット
ステータス	M - 内部	50	32ビット
入力ブロック	D - データレジスタ	0	10ワード※
出力ブロック	D - データレジスタ	10	50ワード※

※本資料では、入力関数作成の項目にて入力するデータ長の都合上、入力ブロックを10ワードとしています。また、出力関数作成の項目にて出力するデータ長の都合上、出力ブロックを50ワードとしています。これらは実際の運用に合わせた数に設定してください。

③設定後、再起動ボタンを押してビジョンセンサを再起動します。

産業用イーサネットの設定

IS3816MP-2d3bc4 の産業用イーサネットの設定

プロトコルの選択 (P)

SLMP

プロトコル固有の設定

設定 **① デバイスのアドレス指定**

名前	選択中のデバイス	オフセット	デバイス数	説明
コントロール	M - 内部	0	32	ビット ビジョンコントロールの PLC 先頭アドレス
ステータス	M - 内部	50	32	ビット ビジョンステータスの PLC 先頭アドレス
入力ブロック	D - データレジスタ	0	10	ワード ユーザデータの PLC 先頭アドレス
出力ブロック	D - データレジスタ	10	50	ワード 検査結果の PLC 先頭アドレス
コマンド	なし	0	0	ビット コマンド文字列の PLC 先頭アドレス
② コマンド結果	なし	0	0	ビット コマンド結果データの PLC 先頭アドレス

OPC UA設定

☐ OPC-UA サーバの有効化

Telnet 経由でのネイティブモードの設定

☒ Telnet サーバが有効

ポート 23

☒ ウィンドウを閉じてデバイスを再起動します。

③ 再起動 キャンセル (C)

4. データブロック

- SLMP を通じてビジョンシステムから送信される信号は、カメラとPLCの間でやり取りされる際に各データブロックに統合されます。操作は**コントロールブロック**と**入力ブロック**で行います。

コントロールブロック	PLCからビジョンシステムへ、制御指令(トリガなど)を行う部分
ステータスブロック	ビジョンシステムの状態(オンライン・一時停止状態など)を表す部分
入力ブロック	PLCからビジョンシステムへ、パラメータなどを入力する部分(Job ID、閾値など)
出力ブロック	ビジョンシステムからPLCへ、測定結果などを出力する部分
コマンドブロック	PLCからビジョンシステムへ、ネイティブモードコマンドを入力する部分
コマンド結果ブロック	ビジョンシステムからPLCへ、ネイティブモードコマンドの結果を出力する部分

- データブロックには、ビットタイプまたはワードタイプのデータが格納されます。ビットタイプのデータはPLCのビットレジスタに保存され、ワードタイプのデータはPLCのワードレジスタに保存されます。このため、データブロック内でビットデータとワードデータが組み合わせられることはありません。
- In-Sight Vision Suite では、コマンド機能に必要なコントロールビットおよびステータスビットを一度に処理することができます。また、ビジョンシステムとPLCの間の接続をセットアップする場合は、各データブロックの開始アドレスとデバイスタイプを設定する必要があります。
- ネイティブモードコマンドの詳細については[こちら](#)をご覧ください。

5. 各ブロック詳細1

	ビット9	ビット8	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
コントロール ブロック	予約済み	予約済み	Set Offline	予約済み	予約済み	Execute Command	Inspection Results Ack	Buffer Results Enable	Trigger	Trigger Enable
	Clear Exposure Complete	Clear Error	Initiate String Command	Set User Data	予約済み	予約済み	予約済み	予約済み	予約済み	予約済み
	External Event 5	External Event 4	External Event 3	External Event 2	External Event 1	External Event 0	予約済み	予約済み	予約済み	予約済み
									External Event 7	External Event 6
	ビット9	ビット8	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
ステータス ブロック	Inspection Completed		Online	Offline Reason			Acq Error	予約済み	Trigger Ack	Trigger Ready
	Exposure Complete	String Command Error	String Command Ack	Set User Data Ack	Error	Command Failed	Command Completed	Command Executing	Result Valid	Result Buffer Overrun
	External Event Ack 5	External Event Ack 4	External Event Ack 3	External Event Ack 2	External Event Ack 1	External Event Ack 0	予約済み	予約済み	予約済み	Job Pass
									External Event Ack 7	External Event Ack 6

5. 各ブロック詳細2

	バイト	
入力 ブロック	0~1	Command ID
	2~3	予約済み
	4~	User Data
	バイト	
出力 ブロック	0~1	Current Job ID
	2~3	Error Code
	4~5	Command Result Code
	6~7	Acquisition ID
	8~9	Inspection ID
	10~11	Inspection Result Code
	12~	Inspection Results
	バイト	
コマンド ブロック	0~1	String Command Length
	2~	String Command
	バイト	
コマンド 結果 ブロック	0~1	String Command Result Code
	2~3	String Command Result Length
	4~	String Command Result

コマンド解説

ブロック	ビット	コマンドワード	説明
コントロールブロック	M0	Trigger Enable	カメラをトリガ受付状態にするためのビット
	M1	Trigger	コマンドがセットされたとき、以下の条件が成立しているとカメラにトリガが入ります。 ・ In-Sight ビジョンシステムがオンラインである。 ・ [Trigger Enable] ビットが設定されている。 ・ トリガソースが ネットワーク(UDP)、ファクトリプロトコルに設定されている。 ※トリガソースは、Spreadsheetの画像取り込み、またはEasy Builderの画像フィルタで設定できます。
	M2	Buffer Results Enable	このビットがセットされると、[Inspection ID]、[Inspection Result Code]、[Inspection Results]は[Inspection Results Ack]が設定されるまで変化しません。最大8セットの検査がビジョンシステムのバッファに保持され、ビジョンシステムは[Results Valid] をクリアして確認応答します。[Inspection Results Ack] がクリアされ、PLCに新しい検査結果が送信されると、[Results Valid] はクリアされなくなります。[Inspection Results Ack]がクリアされ、PLCに送信される結果がビジョンシステムのバッファにない場合、[Results Valid] はクリアされたままになります。
	M3	Inspection Results Ack	[Buffer Results Enable]がセットされると、PLCによって[Inspection Results Ack] がセットされ、[Inspection ID]、[Inspection Result Code]、および[Inspection Results]の データを受信したことが確認されます。[Inspection Results Ack]をクリアすると、バッファが空でなければ、ビジョンシステムは [Results Valid] をセットします。
	M4	Execute Command	このビットがセットされると、ビジョンシステムはコマンドフィールドで指定されたジョブ ID をロードします。この信号は[Command Completed] が切り替わるまでHigh のままにしておく必要があります。この信号の立ち下がリエッジ ([Command Completed] より前) は中止要求と解釈されます。[Command Completed] が設定された後、[Execute Command] をクリアすると [Command Completed] もクリアされます。

コマンド解説

ブロック	ビット	コマンドワード	説明
コントロールブロック	M7	Set Offline	このビットがセットされると、ビットがクリアされるまで、ビジョンシステムはオフラインになります。
	M16	Set User Data	このビットは、新しいユーザデータがビジョンシステムで読み取り可能であることを示すために使用されます。ユーザデータ領域全体を保持バッファに取得した後、ビジョンシステムは PLC にデータを読み取ったことを通知し、必要に応じて[Set User Data Ack]信号を発行してバッファに新しいデータのロードを開始することができます。
	M17	Initiate String Command	このビットがセットされると、[String Command]フィールドからデータを読み出し、文字列コマンドを実行します。コマンドを読み出すと、[String Command Ack]がセットされます。ビジョンシステムがコマンドを受信したことを確認するために、ビジョンシステムが[String Command Ack]をセットするまで、このビットを High に保持する必要があります。
	M18	Clear Error	このビットがセットされると、[Error]と[Error Code]がクリアされます。[Clear Error]は、[Error]がクリアされるまでHighを保持する必要があります。エラーがキューに入っているとき、このビットをクリアすると、[Error]と[Error Code]はキューに入れられた次のエラーコードに設定されます。
	M19	Clear Exposure Complete	Highの間、[Exposure Complete]はリセットされたままになります。このビットがLowに設定されると、次の露光完了時に[Exposure Complete]がHighに設定されます。
	M24~32	External Event	このビットを設定すると、PLC によってスプレッドシートの外部イベント関数をトリガすることができます。

コマンド解説

ブロック	ビット	コマンドワード	説明
ステータスブロック	M50	Trigger Ready	ビジョンシステムが、[Trigger]を通して新しいトリガを受け付けられる状態であることを示します。このビットは、以下の条件が成立しているとTrueになります。 ・ In-Sightビジョンシステムがオンラインである。 ・ [Trigger Enable] ビットが設定されている。 ・ トリガソースが ネットワーク(UDP)またはファクトリプロトコルに設定されている。 ・ ビジョンシステムが画像の取り込み中でない。 ※トリガソースは、Spreadsheetビューの画像取り込み、またはEasy Builderビューの画像フィルタステップで設定できます
	M51	Trigger Ack	ビジョンシステムが[Trigger]経由で正常にトリガされたことを示します。[Trigger Ack]は、[Trigger]がクリアされるまで、Trueが維持されます。さらに、このビットの立ち上がりエッジに[Acquisition ID]を同期することができます。 注意：ビジョンシステムがオフラインの時に[Trigger]がセットされていた場合、[Acq Error]と [Trigger Ack] は共にセットされます。

コマンド解説

ブロック	ビット	コマンドワード	説明									
ステータスブロック	M53	Acq Error	次のいずれかの理由により、ビジョンシステムが画像の取り込みに失敗したときに設定されるビットです。 プロパティが [産業用イーサネット] に設定されていない。 In-Sight デバイスがオンラインではない。 \$System.OnlineEnabled タグが False になっている。 In-Sight デバイスが現在、画像を取り込み中である。 In-Sight デバイスの画像バッファが不足している。 このビットは、正常な取り込みがトリガされたとき、または [Trigger Enable] ビットがトグルされたときにクリアされます。 注： [Trigger Enable] ビットが Low のときに [Trigger] が設定された場合、 [Acq Error] が生じますが、これは [Trigger] が無視されるためです。									
	M54~56	Offline Reason	<table><tr><td colspan="2">このビットは、ビジョンシステムがオフラインになった原因を示します。複数の原因が存在する場合、このフィールドは最も小さいコードを持つビットをTrueにします。</td></tr><tr><td>0：オンライン</td><td>ビジョンシステムはオンラインです。</td></tr><tr><td>1：プログラミングオフライン</td><td>ビジョンシステムのジョブを変更中です。</td></tr><tr><td>2：ディスクリートオフライン</td><td>ディスクリートI/O信号がビジョンシステムをオフラインにしています。</td></tr><tr><td>3：通信オフライン</td><td>通信プロトコルがビジョンシステムをオフラインにしています。</td></tr></table>	このビットは、ビジョンシステムがオフラインになった原因を示します。複数の原因が存在する場合、このフィールドは最も小さいコードを持つビットをTrueにします。		0：オンライン	ビジョンシステムはオンラインです。	1：プログラミングオフライン	ビジョンシステムのジョブを変更中です。	2：ディスクリートオフライン	ディスクリートI/O信号がビジョンシステムをオフラインにしています。	3：通信オフライン
このビットは、ビジョンシステムがオフラインになった原因を示します。複数の原因が存在する場合、このフィールドは最も小さいコードを持つビットをTrueにします。												
0：オンライン	ビジョンシステムはオンラインです。											
1：プログラミングオフライン	ビジョンシステムのジョブを変更中です。											
2：ディスクリートオフライン	ディスクリートI/O信号がビジョンシステムをオフラインにしています。											
3：通信オフライン	通信プロトコルがビジョンシステムをオフラインにしています。											

コマンド解説

ブロック	ビット	コマンドワード	説明
ステータスブロック	M57	Online	ビジョンシステムがオンラインのときにセットされ、ビジョンシステムがオフラインのときにクリアされます。ビジョンシステムがオフラインの場合、[Offline Reason] に理由が表示されます。
	M59	Inspection Completed	検査終了時に設定されるビットです。[Inspection ID]、[Inspection Result Code]、[Inspection Results]、または[Job Pass]のビットが PLC に送信された後にトグルされます。 注：適用にはWriteResultsBuffer関数の使用が必要です。
	M60	Result Buffer Overrun	[Buffer Results Enable]が設定され、PLCが[Inspection Results Ack]を設定して検査結果を確認せずに、ビジョンシステムが検査結果のセットを破棄した場合に設定されます。検査結果は最大8つまでビジョンシステムのバッファに保持されますが、このビットは、9番目の検査がバッファに追加されたときに設定されます。バッファに空きができるまで（検査結果が確認されるまで）、9番目の検査とそれ以降の検査はドロップされます。有効な検査が発生し、前の検査が上書きされない限り、このビットはクリアされません。
	M61	Result Valid	このビットは[Inspection ID]、[Inspection Result Code]、[Inspection Results]、または[Job Pass]のビットがセットされているときにセットされ、[Inspection Results Ack]がセットされたときにクリアされます。 注：オーバーラップモードでジョブ処理を有効にする場合は、[Buffer Results Enable]を有効にするか、[Inspection Completed]を使用して検査結果をラッチする必要があります。
	M62	Command Executing	コマンドが進行中のときに設定されるビットです。
	M63	Command Completed	コマンドが完了すると、[Command Completed]がセットされます。このとき、[Command Executing]はLowになります。コマンドが正常に完了しなかった場合、[Command Failed]もセットされます。[Execute Command] ビットがクリアされたときにクリアされます。

コマンド解説

ブロック	ビット	コマンドワード	説明
ステータスブロック	M64	Command Failed	コマンドの実行に失敗したか、または未知の[Command ID]が設定された場合、コマンド完了時にこのビットが設定されます。PLC/HMI によって新しいジョブがロードされると、このビットはクリアされますが、スプレッドシートでジョブが変更されても、このビットはクリアされません。このビットは、[Command Completed]を設定する前に常に設定されます。
	M65	Error	[Error Code]で定義されているエラーが発生したときに設定されます。 [Error Code]に関しては このページ をご覧ください
	M66	Set User Data Ack	[Set User Data]の完了を確認するとセットされます。
	M67	String Command Ack	ビジョンシステムが文字列コマンドの処理を完了し、応答を生成したときにセットされます。ビジョンシステムがこのビットをセットした後、PLCは文字列コマンドの応答データを安全に取り出すことができます。PLCが[Initiate String Command]をクリアすると、このビットはリセットされます。PLCは、新しいコマンドを開始する前に、[Initiate String Command]をクリアし、[String Command Ack]がLow状態に戻るのを待つ必要があります。
	M68	String Command Error	直前の文字列コマンドの結果が失敗したかどうかを示します。
	M69	Exposure Complete	ビジョンシステムの画像センサが露光を完了するとセットされ、[Clear Exposure Complete]によってリセットされます。ビジョンシステムの視野から検査対象を移動できることを示します。

コマンド解説

ブロック	ビット	コマンドワード	説明
ステータスブロック	M70	Job Pass	検査の合格/不合格の状態を示します。Job Pass/Fail Cell Setup ダイアログで設定された通り、直近のジョブがパスした場合にセットされます。ジョブがパスしなかった場合、このビットはクリアされます。 このビットの動作は、結果バッファリングが無効か有効かによって異なります。 バッファリングが無効： 検査が完了し、ジョブの合否セルが合格を示す場合に設定され、そうでない場合はクリアされたままです。[Job Pass]は[Inspection Completed]がセットされる前に有効です バッファリングが有効： 新しい結果が[Inspection Results]に配置され、その結果に対してJob Pass/Failセルが合格を示したときにセットされ、そうでなければクリアされたままです。この状態は、[Inspection Results Ack]をTrueに設定することによって結果が承認されるまで変化せず、その時点でビットはクリアされます。[Job Pass]は、[Inspection Completed]がセットされる前に有効です。 注：適用にはWriteResultsBuffer関数の使用が必要です。
		External Event Ack	PLCが[External Event]を受信したことを示します。

コマンド解説

ブロック	ワード	コマンドワード	説明
入力ブロック	0	Command ID	このフィールドには、ロードするジョブのジョブ ID 番号（0～999）を16 ビット整数で入力します。ジョブのロードは、PLC によって [Execute Command] ビットが設定されると実行されます。 [Command Completed]が設定されるまで、このフィールドは変更しないでください。 0x0000 ～ 0x03E7: ジョブ N に変更します (N は、0 ～ 999 のコマンド ID)。 0x4000: ジョブを名前で変更します (名前は UserData から転送されます) 0-0x0fff = 予約済み 0x1000-0x1fff = 予約済み 0x2000 = \$System.ProcessingEnabledタグを False に設定します 0x2001 = \$System.ProcessingEnabledタグを True に設定します 0x2002-0xFFFF = 予約済み
	2～	User Data	ReadUserDataBuffer関数または ReadLatchedUserDataBuffer関数を使用してスプレッドシートに読み取ることができるデータバッファです。バッファはPLC の表示どおりに入力され、ビットは、RSLogix 5000 の定義と同じ順序で表示されます。
出力ブロック	0	Current Job ID	現在セットされているジョブファイル名の先頭の番号を示します。ジョブ ID は 0 ～ 65535 の 16 ビット番号ですが、PLC ファンクションは 0 ～ 999 のジョブ ID しかロードできません。 注：ファイル名のジョブ ID 番号が 65535 より大きい場合、番号付けの上限が 65535 になるため、ジョブ ID は 65535 になります。

コマンド解説

ブロック	ワード	コマンドワード	説明		
出力ブロック	1	Error Code	発生したエラーを16ビットの数値で表します。		
			Code	Error名	説明
			0x0000	No Error	エラーなし
			0x0100	Trigger set while disabled	[Trigger Enable]がクリアされている状態で[Trigger]がセットされた。またはAcquireImage関数のトリガソースが ネットワーク(UDP)、ファクトリプロトコルに設定されている。
			0x0101	Trigger set while Offline	ビジョンシステムがオフラインの時に[Trigger]がセットされた
			0x0400	Already Executing Error	[Execute Command]がセットされたとき、[Command Executing]が既にセットされている。
			0x0401	Job load requested while Online	ビジョンシステムがオンライン中に[Executing Command]がセットされた。
			0x0402	Job load requested does not exist on the camera	[Command ID]フィールドに入力されたJob IDがカメラのメモリ内に存在しない。
			0x0410	TestRun not enabled	TestRunが有効になっていないジョブがロードされた状態で、TestRunを実行した
			0x0411	TestRun requested while Online	システムがオンライン中に TestRun の実行が要求された

コマンド解説

ブロック	ワード	コマンドワード	説明
出力ブロック	2	Command Result Code	コマンドの実行に失敗した理由を示すフィールドです。 (コードは、実行されたコマンド固有です)。 0x0000 = コマンド成功/未知の失敗 0x0002 = 誤ったコマンド 0x208D = リクエストがシステム状態に適合しない
	3	Acquisition ID	トリガソースに関係なく、各画像取り込みが完了すると増分されるフィールドです。 取り込みとその検査結果を同期化するために使用できます。[Acquisition ID] は、取り込みを開始すると変化します。PLC がトリガソースである場合、[Trigger Ack] ビットを設定する前に [Acquisition ID] が設定されます。
	4	Inspection ID	検査が完了すると増分されるフィールドです。
	5	Inspection Result Code	WriteResultsBuffer関数の [結果コード] パラメータで定義された結果コードを表示するフィールドです。
	6~	Inspection Results	WriteResultsBuffer関数を使用して、スプレッドシートからこのフィールドに検査結果データを書き込むことができます。

コマンド解説

ブロック	ワード	コマンドワード	説明
コマンドブロック	0	String Command Length	ネイティブモードコマンドの長さ
	1~	String Command	ビジョンシステムに送信されるコマンド
コマンド結果ブロック	0	String Command Result Code	ネイティブモードコマンドのレスポンスコード。 注：各ネイティブモードコマンドは、実行に成功した場合は1を返し、失敗した場合は特定のエラーコードを返します。
	1	String Command Result Length	ネイティブモードコマンドのレスポンスデータのバイト長。
	2~	String Command Result	ネイティブモードコマンドの結果文字列で、コマンドによって返されるデータのASCII文字列表現



GX Works 3

通信設定

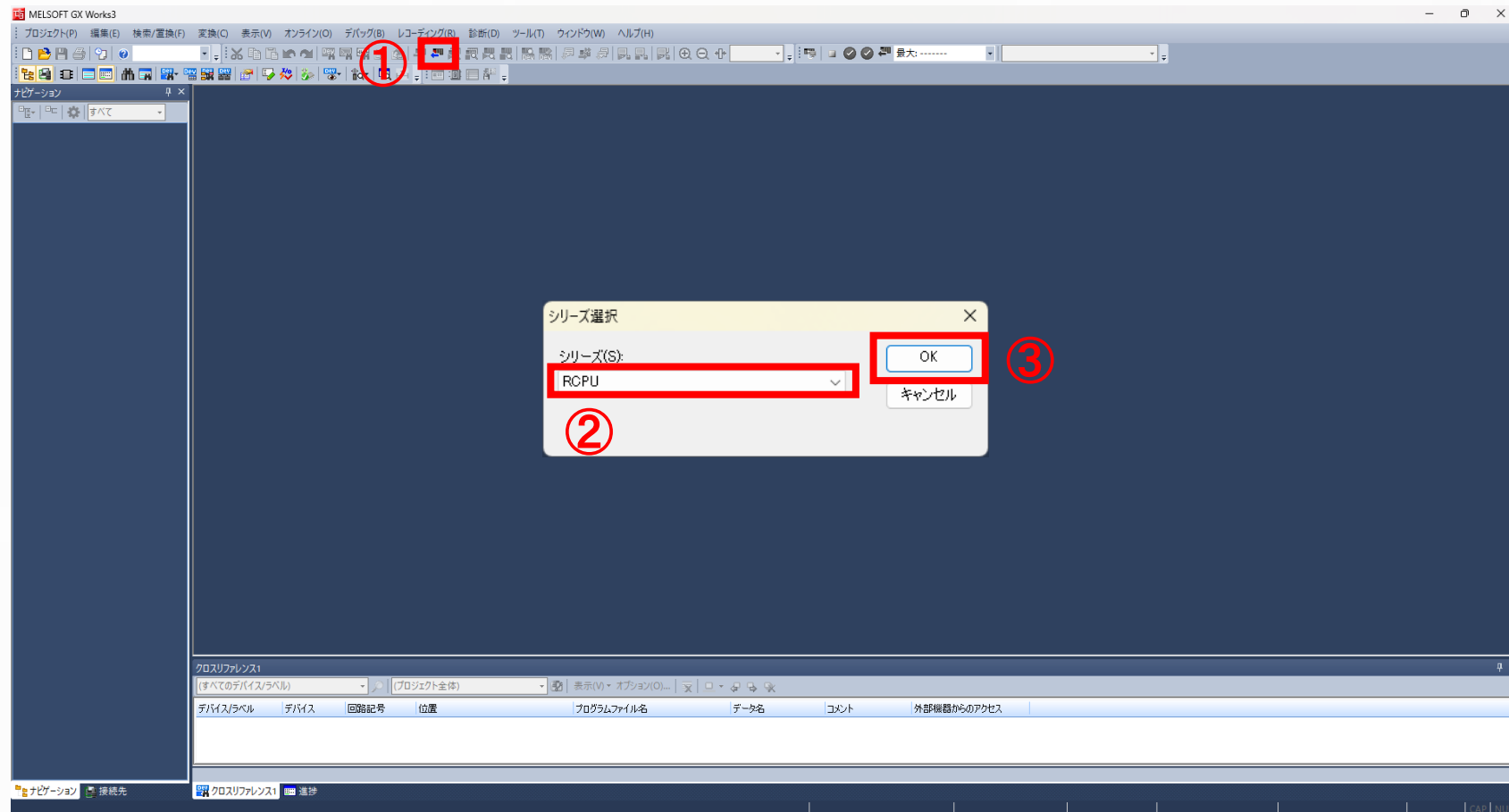
Summary

1. [PLC設定読み出し](#)
2. [PLC IPアドレス設定](#)
3. [相手機器接続構成設定](#)
4. [PLCへの設定書き込み](#)
5. [PLC本体リセット](#)
6. [SLMP接続確認](#)
7. [モニタリング設定](#)

※RUN中書き込みを許可していない場合は、CPUユニットはRUNではなくSTOPにして設定を行ってください。

1. PLC設定読み出し1/3

GX Works 3を立ち上げ、①シーケンサからの読み出しを選択し、②RCPUを選択して③OKします。



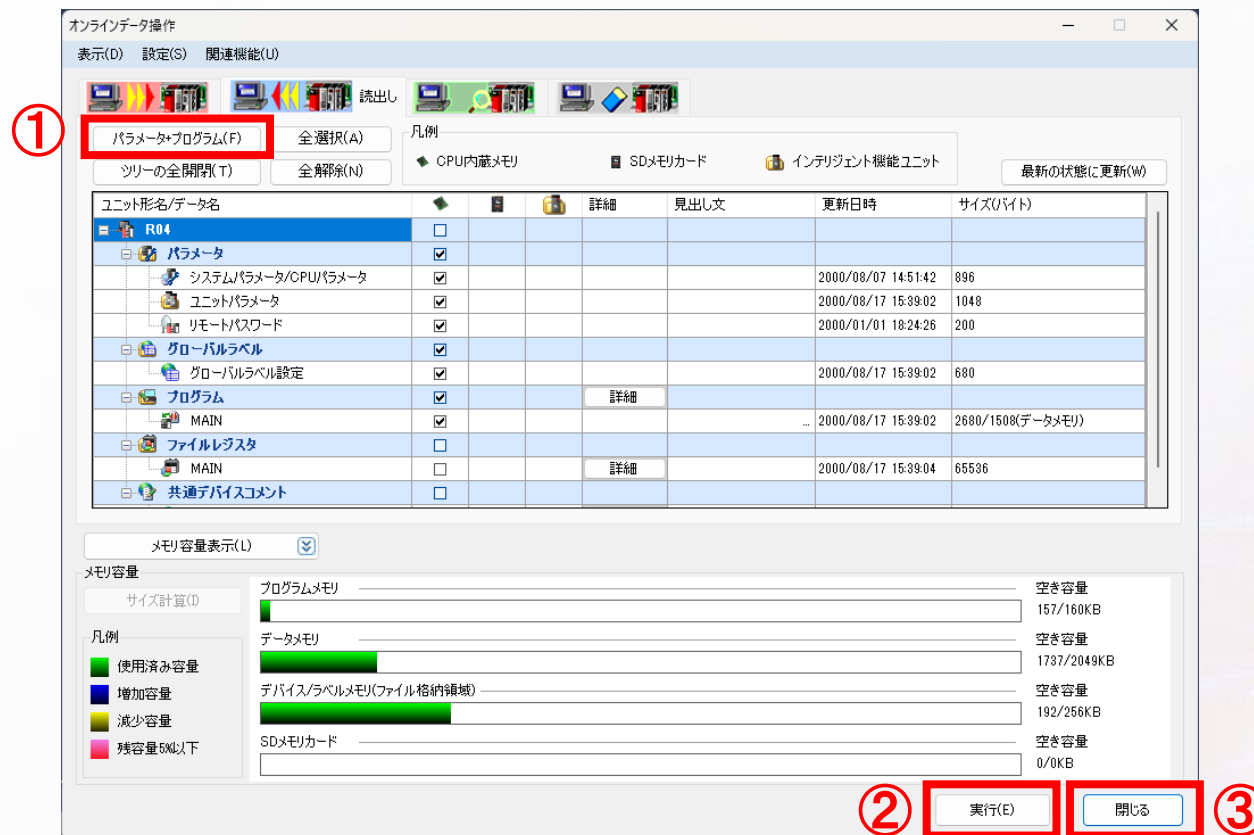
1. PLC設定読み出し2/3

①通信テスト(T)を行なったのち、②OKを押します。



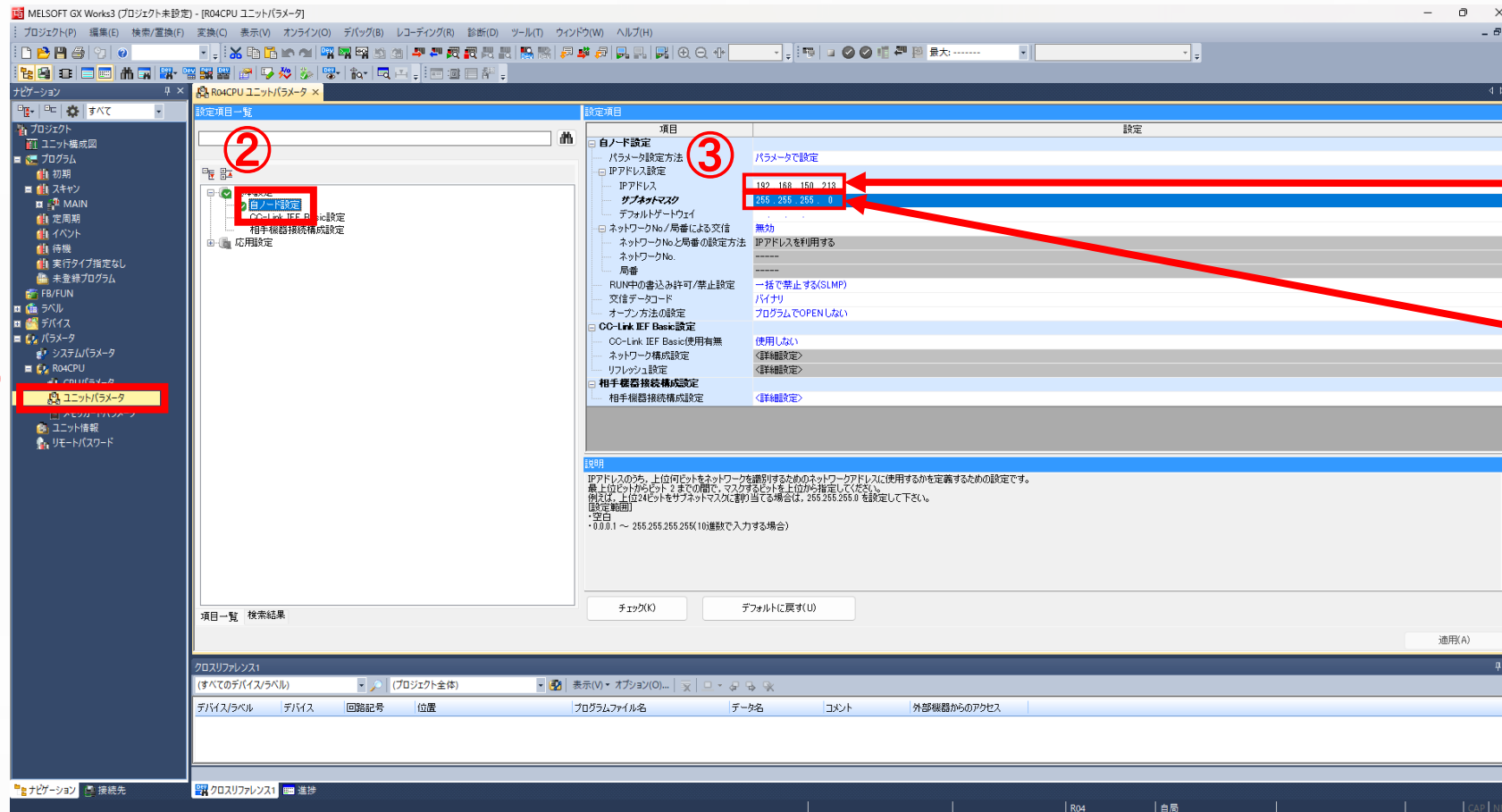
1. PLC設定読み出し3/3

①読み出す設定をパラメータ+プログラム(F)に設定し、②実行後に③ウィンドウを閉じます。



2. PLC IPアドレス設定

①パラメータ→R04CPU→ユニットパラメータから②基本設定→自ノード設定を開き、③PLCにIPアドレスを割り振ります。



③[設定]タブの以下の項目を設定します。

- IPアドレス : 192.168.1.10
- サブネットマスク : 255.255.255.0

3. 相手機器接続構成設定1/2

①基本設定→相手機器接続構成設定を開き、②詳細設定を開きます。

MELSOFT GX Works3 (プロジェクト未設定) - [R04CPU ユニットパラメータ]

プロジェクト(P) 編集(E) 検索/置換(F) 変換(C) 表示(V) オンライン(O) デバッグ(B) レコーディング(R) 診断(D) ツール(T) ウィンドウ(W) ヘルプ(H)

ナビゲーション

設定項目一覧

検索する設定項目をここに入力

基本設定

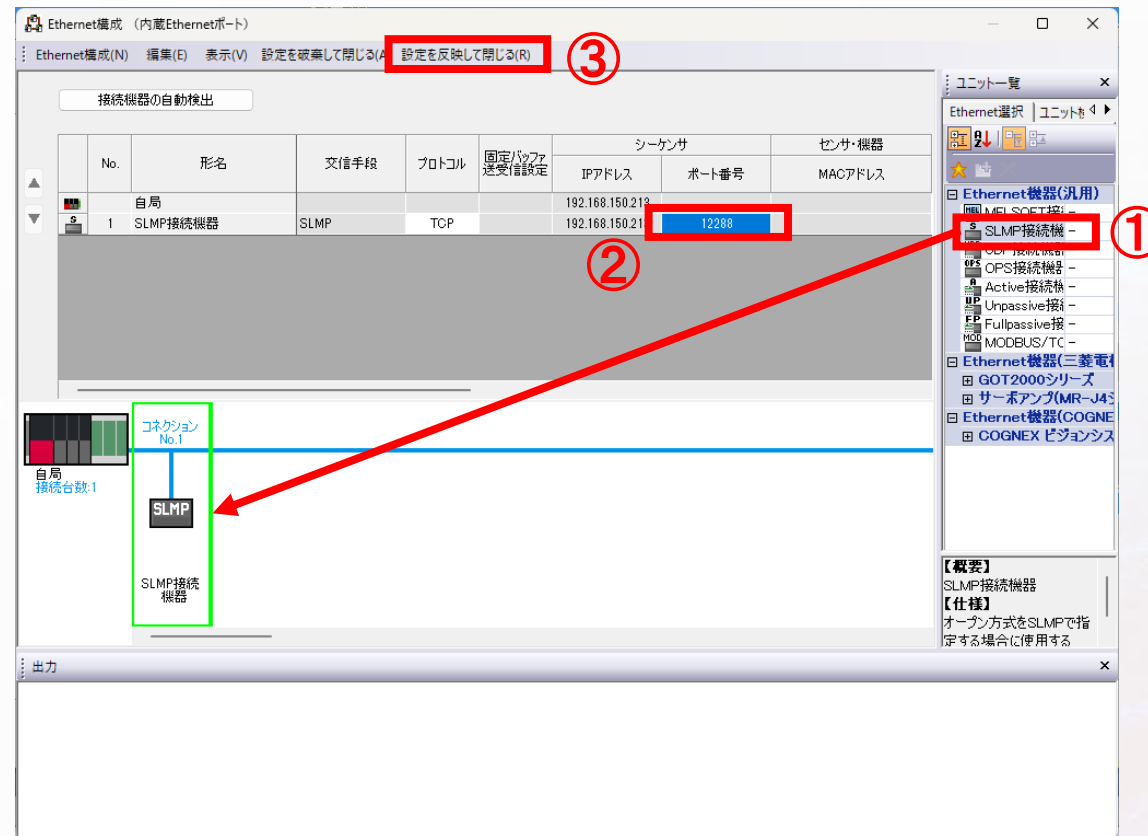
① 相手機器接続構成設定

設定項目

項目	
自ノード設定	
パラメータ設定方法	パラメータで設定
IPアドレス設定	
IPアドレス	192.168.0.39
サブネットマスク	255.255.255.0
デフォルトゲートウェイ	
ネットワークNo./局番による通信	無効
ネットワークNo.と局番の設定方法	IPアドレスを利用する
ネットワークNo.	-----
局番	-----
RUN中の書き込み許可/禁止設定	一括で禁止する(SLMP)
通信データコード	バイナリ
オープン方法の設定	プログラムでOPENしない
CC-Link IEF Basic設定	
CC-Link IEF Basic使用有無	使用する
ネットワーク構成設定	<詳細設定>
リフレッシュ設定	<詳細設定>
相手機器接続構成設定	
相手機器接続構成設定	② <詳細設定>

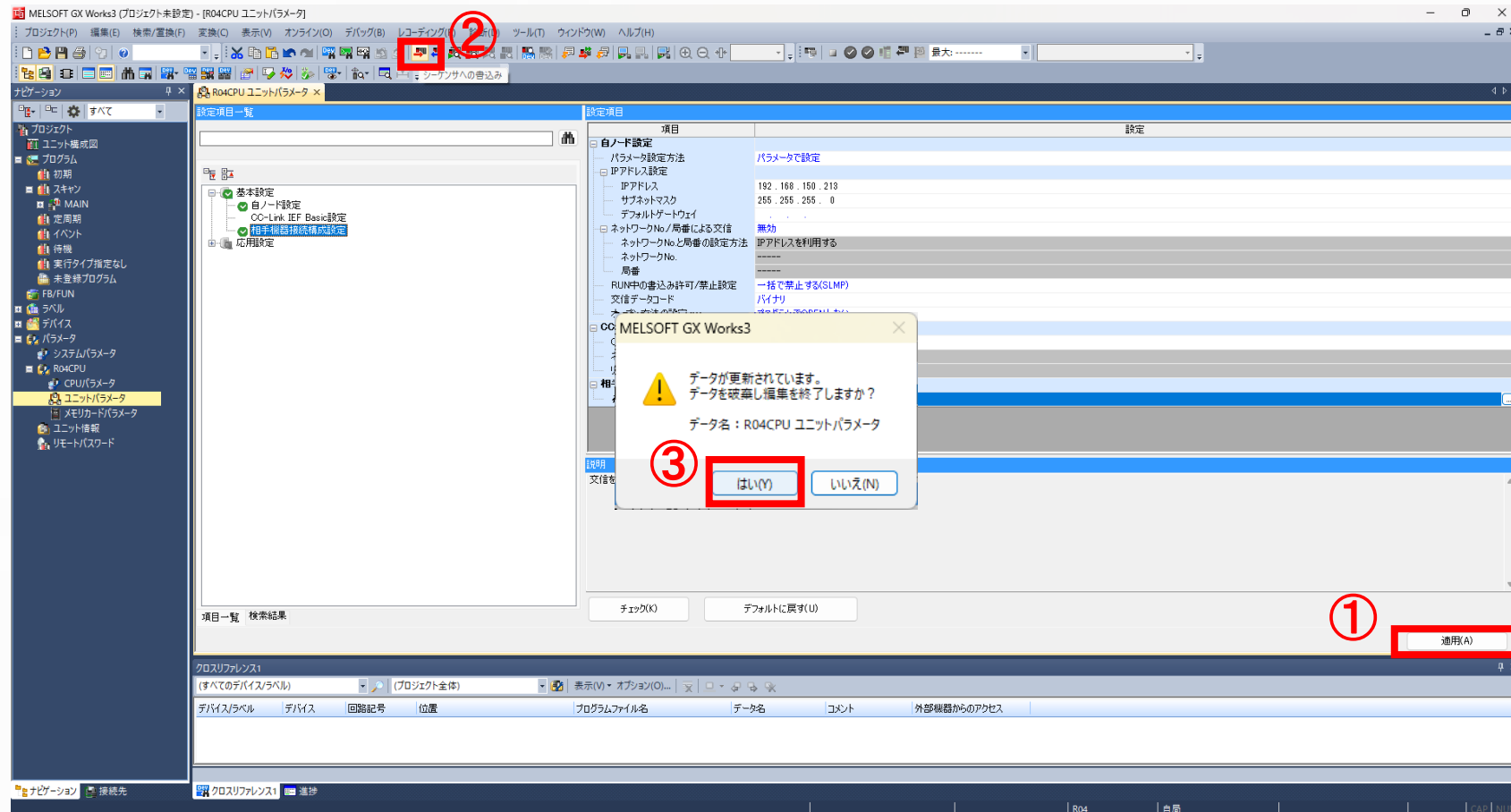
3. 相手機器接続構成設定2/2

①Ethernet機器(汎用)→SLMP接続機器を矢印に沿ってドラッグアンドドロップします。②ポート番号に「12288」を入力し、③設定を反映して閉じます。



4. PLCへの設定書き込み1/2

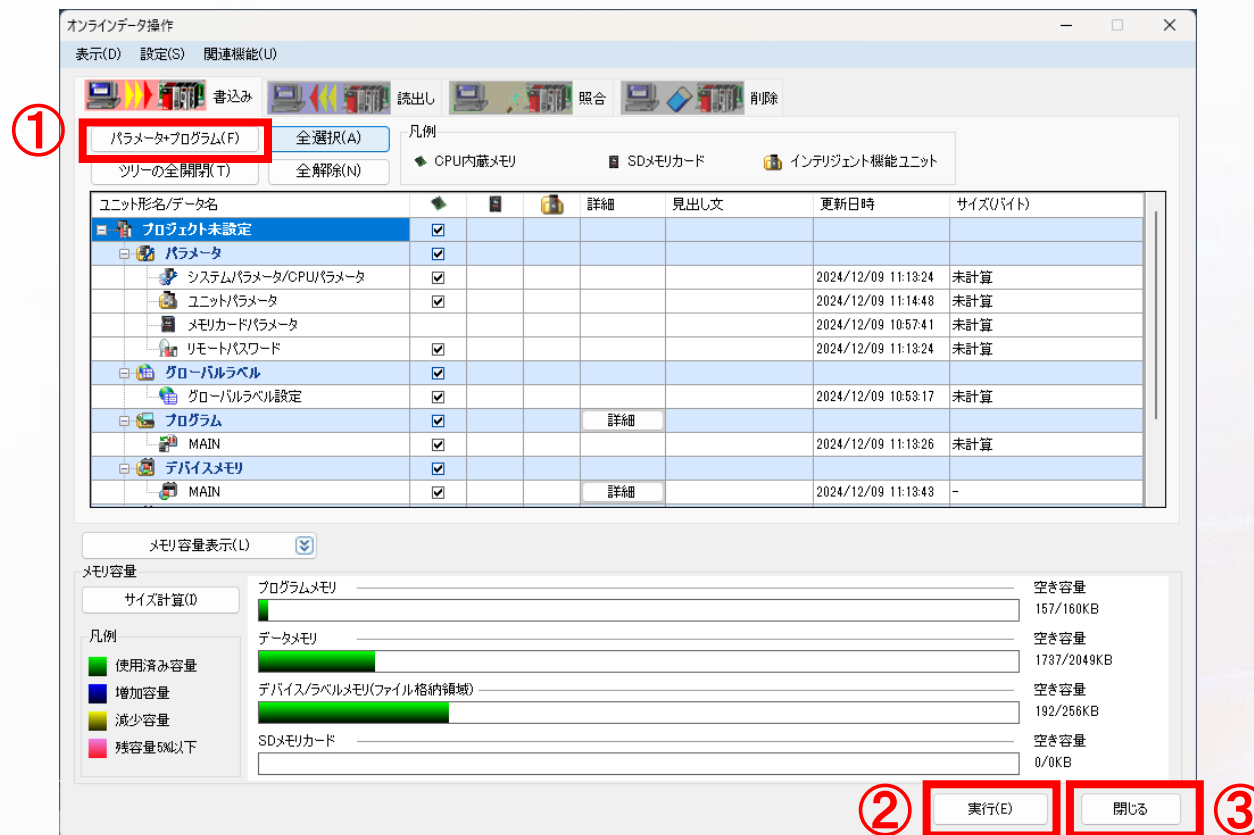
①設定を適用し、②シーケンサへの書き込みを選択し、③はい(Y)を押します。



4. PLCへの設定書き込み2/2

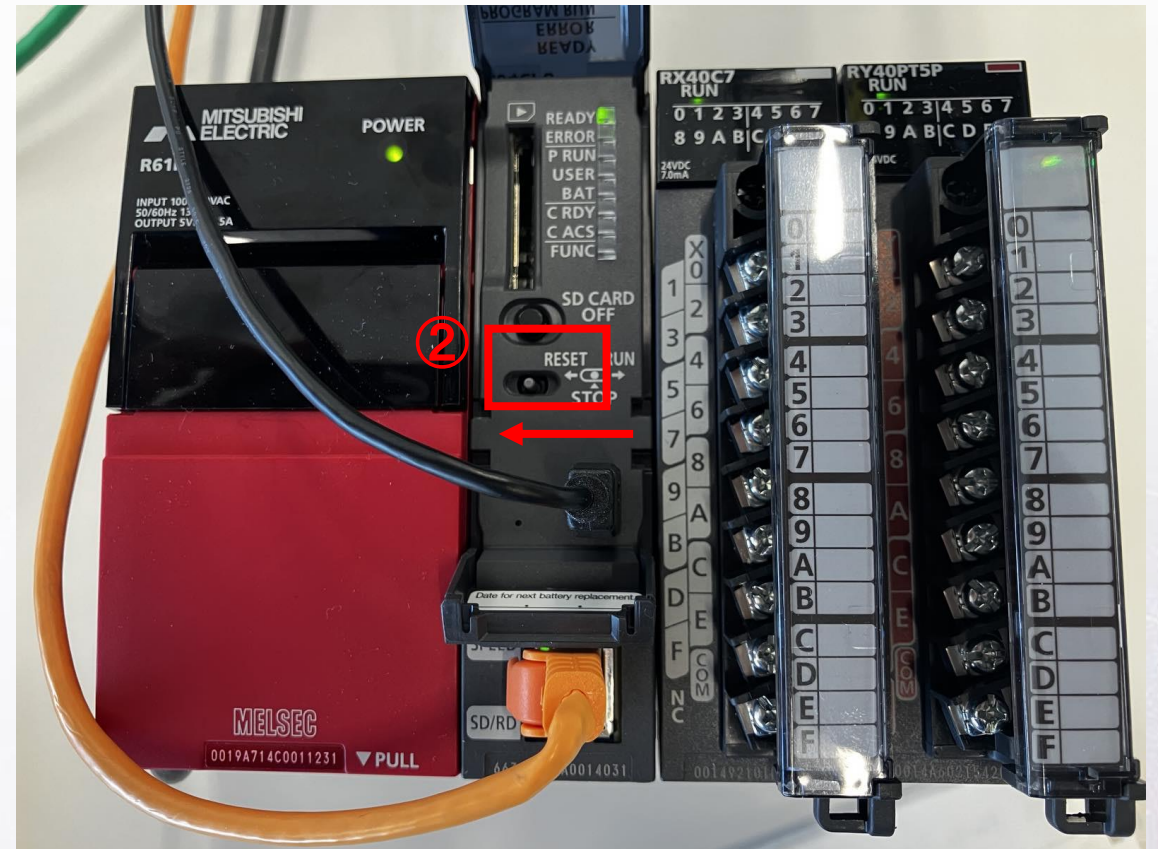
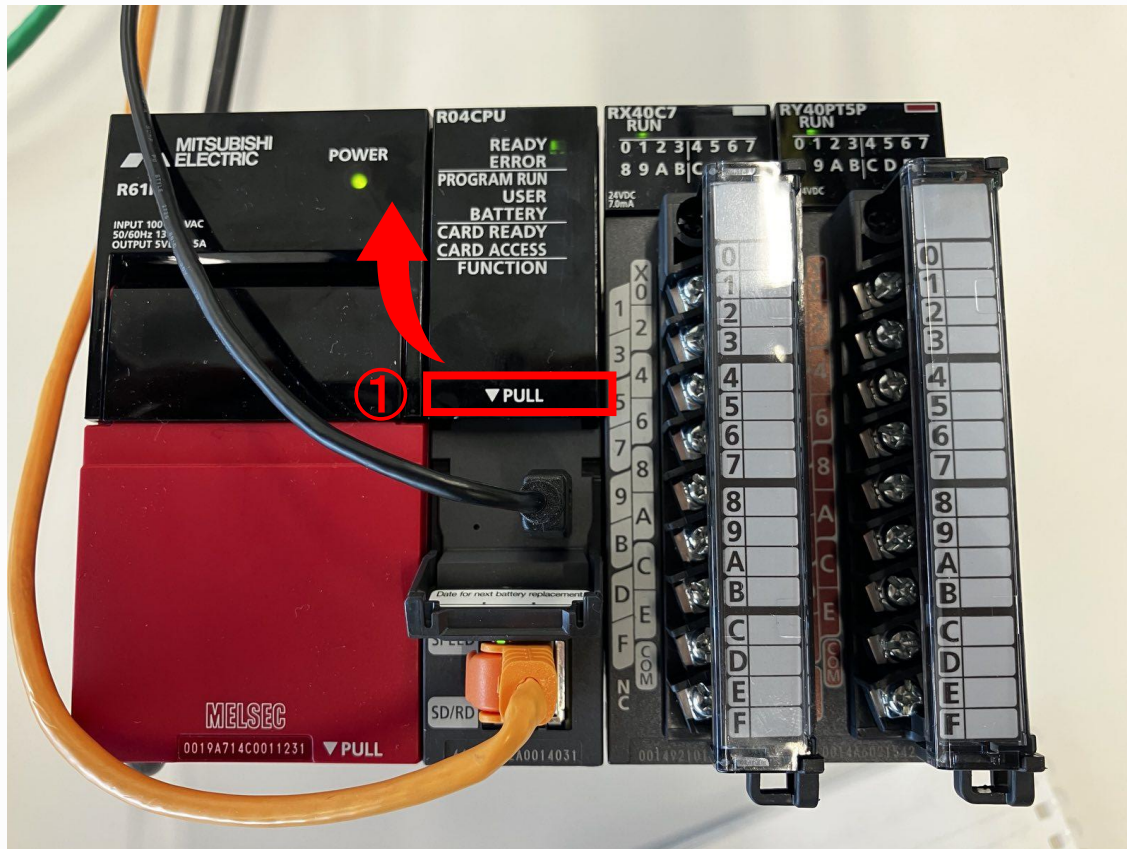
①書き込む設定をパラメータ+プログラム(F)に設定し、②実行後に③ウィンドウを閉じます。

※プログラム
書き込みは必須
ではありません。
ご使用の環境に
合わせて設定して
下さい。



5. PLC本体リセット

①CPUユニットを開き、②RESET側につまみを3秒程度倒します。リセットが完了するとカチッと音が鳴り、PLCが再起動します



6. SLMP接続確認

①診断→Ethernet診断を開き、②SLMP通信が確立していることを確認します。

Ethernet診断

対象ユニット指定
☐ ユニットNo.(U) 1枚目(ポート1) ☐ I/Oアドレス(I) 0000 ☒ CPU(M) 1号機

IPアドレス表示切替 ☒ 10進(D) ☐ 16進(H) ポート番号表示切替 ☐ 10進(D) ☒ 16進(H)

モニタ中
モニタ停止(S)

コネクション別状態 | プロトコル別状態 | 回線状態

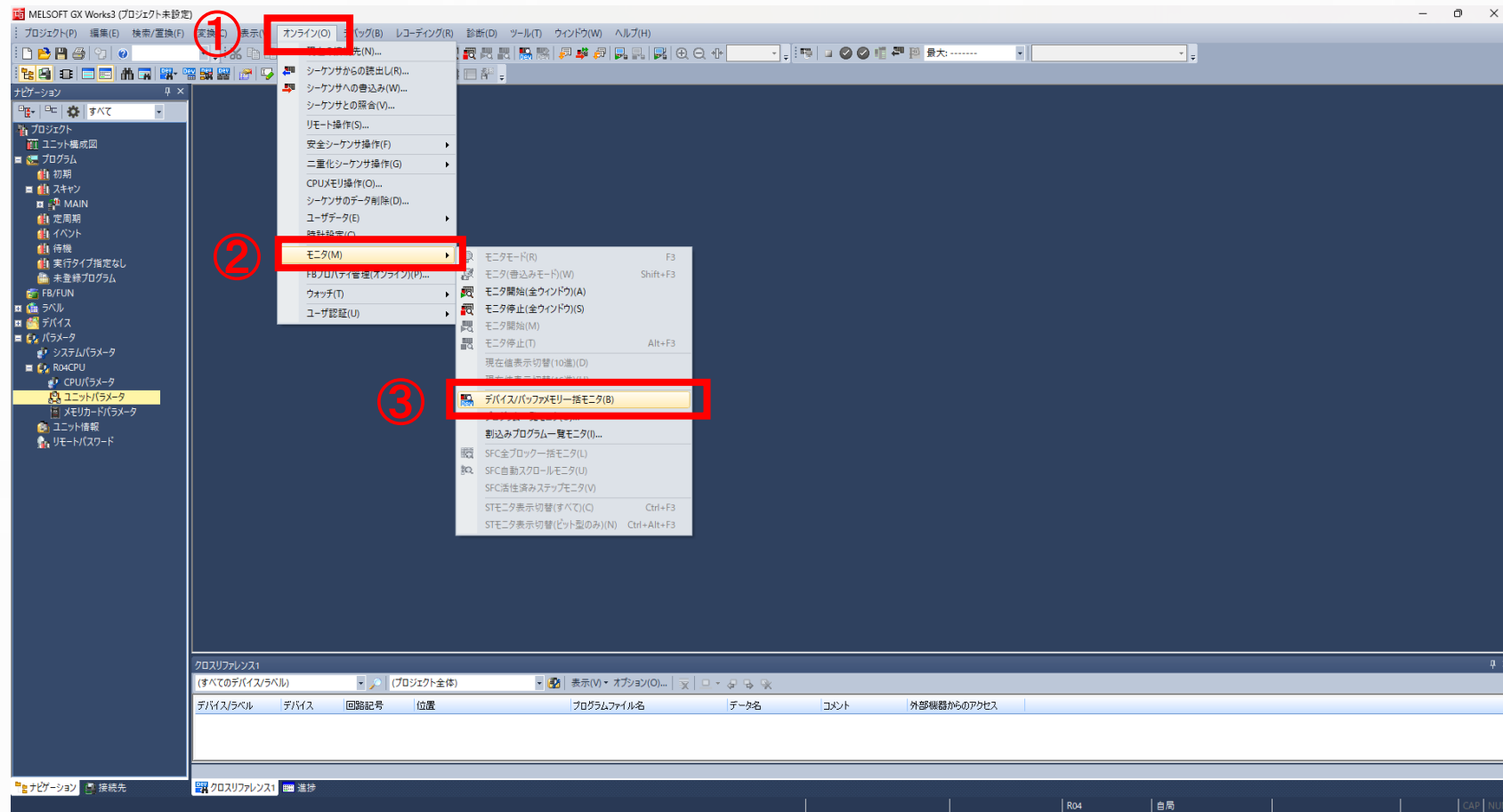
コネクションNo./機能	自局ポート番号	通信相手 通信手段	通信相手 IPアドレス	通信相手 ポート番号	最新 エラー コード	プロトコル	オープン 方式	TCP 状態	リモート パスワード 状態
1	3000	SLMP	192.168.150.71	7FC4	----	TCP	Unpassive	接続中	無効、または解除
2	----	----	----	----	----	----	----	----	----
3	----	----	----	----	----	----	----	----	----
4	----	----	----	----	----	----	----	----	----
5	----	----	----	----	----	----	----	----	----
6	----	----	----	----	----	----	----	----	----
7	----	----	----	----	----	----	----	----	----
8	----	----	----	----	----	----	----	----	----
9	----	----	----	----	----	----	----	----	----
10	----	----	----	----	----	----	----	----	----
11	----	----	----	----	----	----	----	----	----
12	----	----	----	----	----	----	----	----	----
13	----	----	----	----	----	----	----	----	----
14	----	----	----	----	----	----	----	----	----
15	----	----	----	----	----	----	----	----	----
16	----	----	----	----	----	----	----	----	----
FTPサーバ	----	----	----	----	----	----	----	切断	無効、または解除
FTPクライアント	----	----	----	----	----	----	----	切断	----
MELSOFTの直結接続	----	----	----	----	----	----	----	----	無効、または解除

最新エラーコードをクリア(E)

PINGテスト(P) 通信状態テスト(C) 閉じる

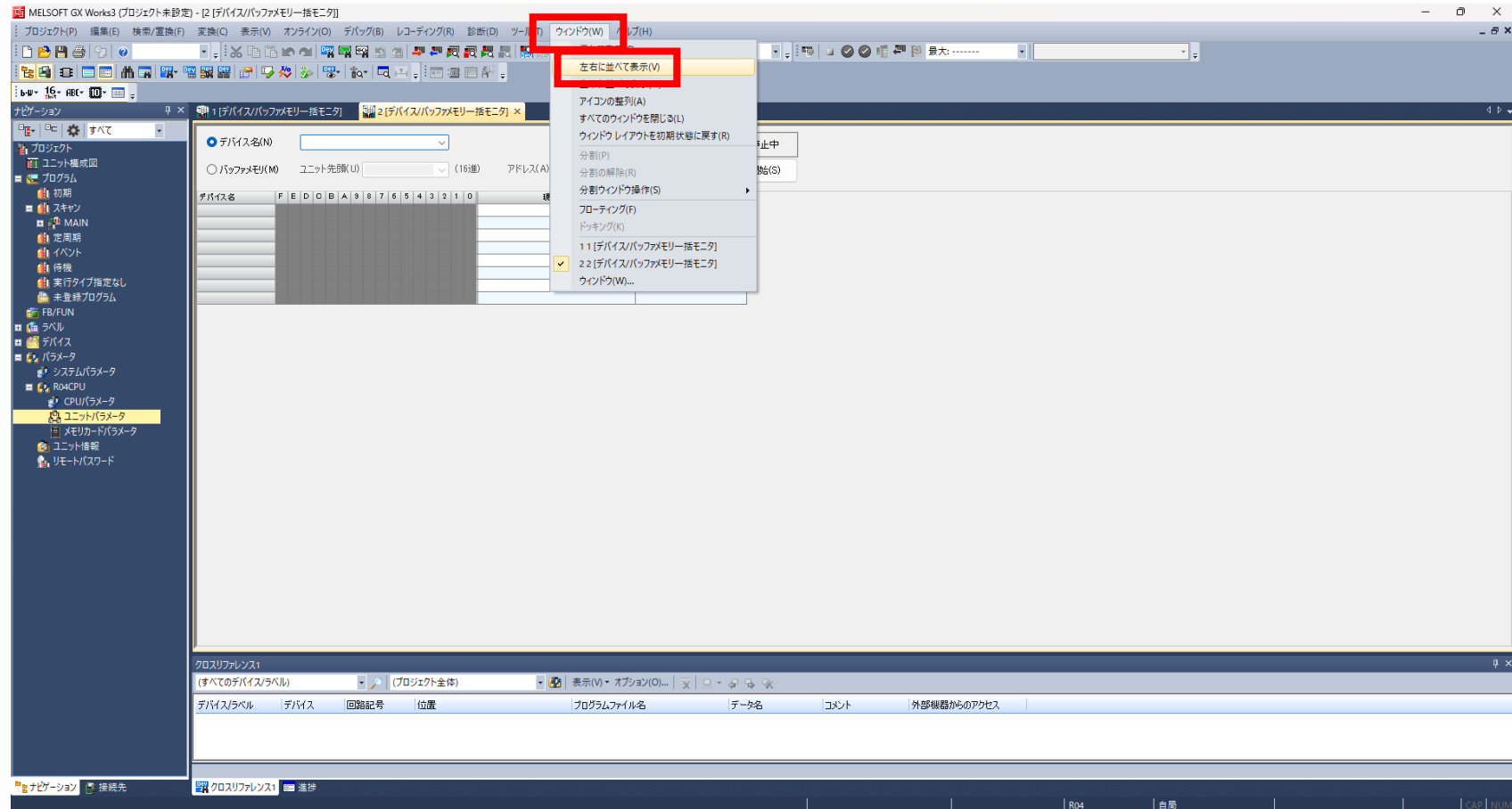
7. モニタリング設定1/3

①オンライン→②モニター→③デバイスバッファメモリー一括モニタを開きます。
同じ動作をもう一度繰り返します。



7. モニタリング設定2/3

ウィンドウ→左右に並べて表示を選択します。



7. モニタリング設定3/3

①左側のデバイスにM0、②右側のデバイスにD0と入力します。

The screenshot displays the MELSOFT GX Works3 interface with two monitoring windows. The left window is for device M0, and the right window is for device D0. Red boxes highlight the device selection dropdowns in both windows. Brackets on the right side of the D0 window indicate input and output blocks.

① 左側のデバイスにM0

② 右側のデバイスにD0

M0~M32:コントロールブロック

M50~M72:ステータスブロック

D0~D9:入力ブロック

D10~D59出力ブロック

デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値	文字列
D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
D21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-



GX Works 2

通信設定

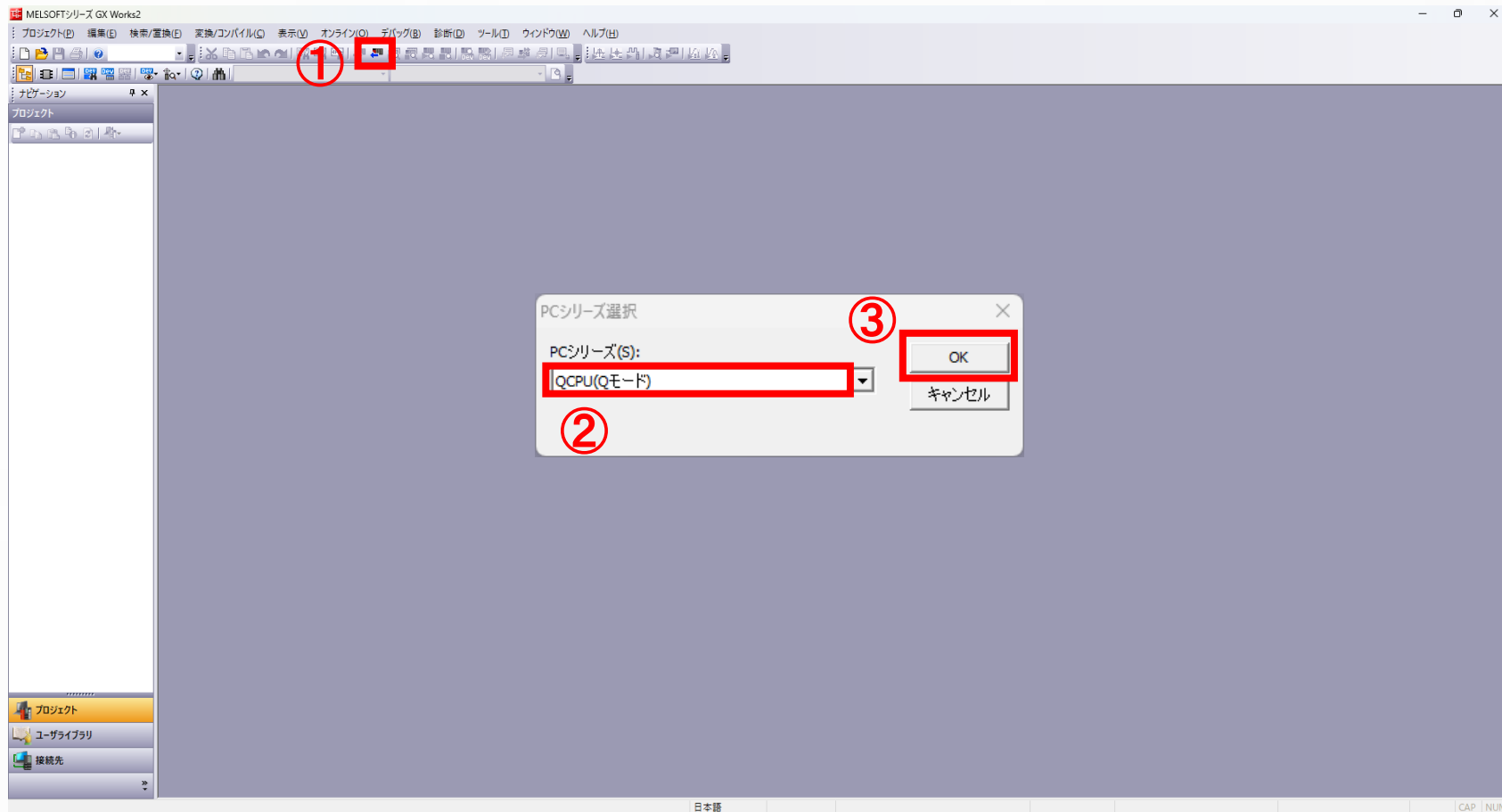
Summary

1. PLC設定読み出し
2. PLC IPアドレス設定
3. 内蔵Ethernetポートオープン設定
4. PLCへの設定書き込み
5. PLC本体リセット
6. SLMP接続確認
7. モニタリング設定

※RUN中書き込みを許可していない場合は、CPUユニットはRUNではなくSTOPにして設定を行ってください。

1. PLCの設定読み出し1/3

GX Works 2を立ち上げ、①PC読出を選択し、②QCPUを選択して③OKします。



1. PLCの設定読み出し2/3

①通信テスト(T)を行なったのち、②OKを押します。



1. PLC設定読み出し3/3

①読み出す設定をパラメータ+プログラム(P)に設定し、②実行後に③ウィンドウを閉じます。



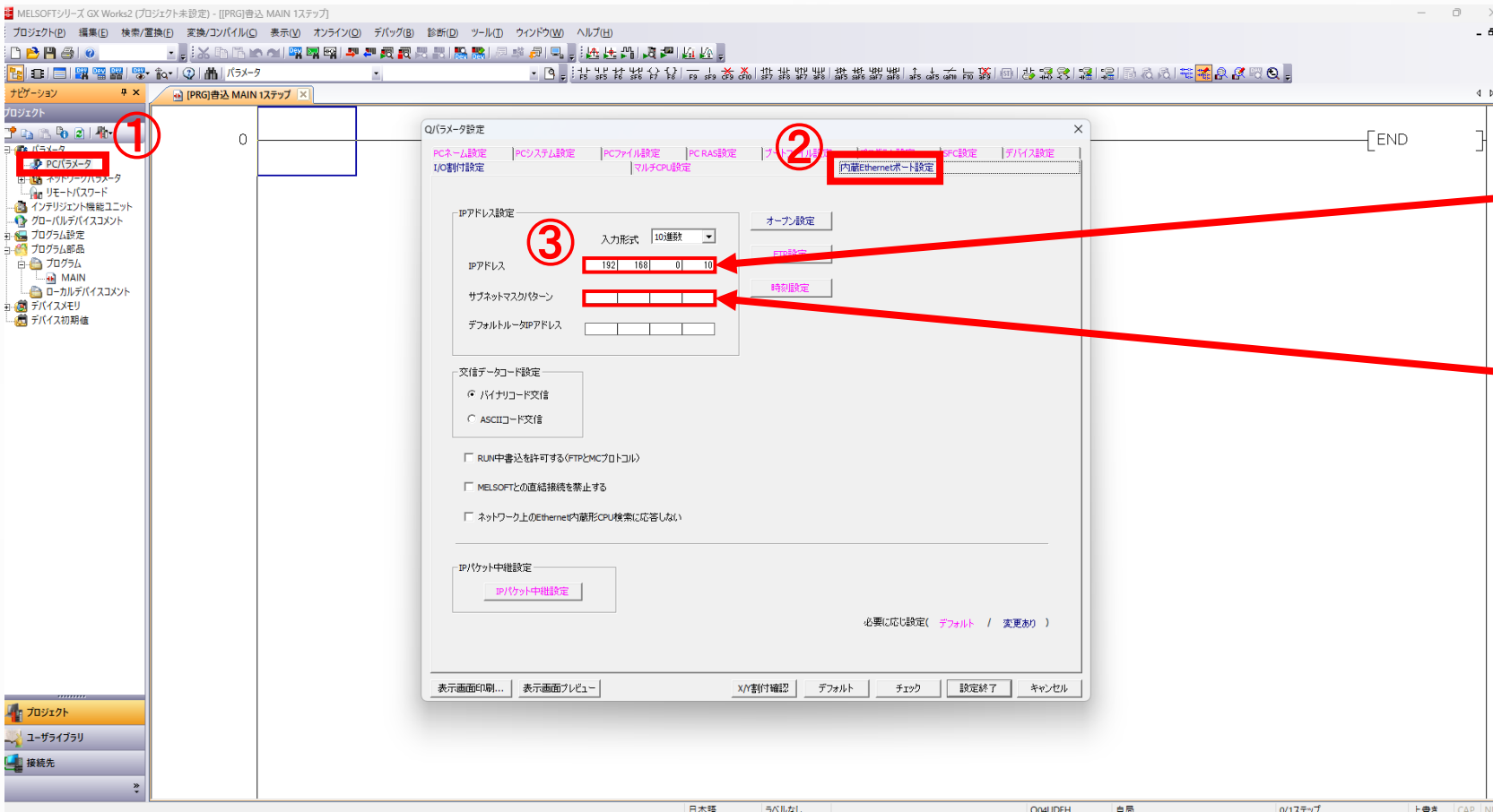
2. PLC IPアドレス設定

①パラメータ→PCパラメータから②内蔵Ethernetポート設定を開き、③PLCにIPアドレスを割り振ります。

③[設定]タブの以下の項目を設定します。

- IPアドレス : 192.168.0.10

- サブネットマスクパターン : 255.255.255.0

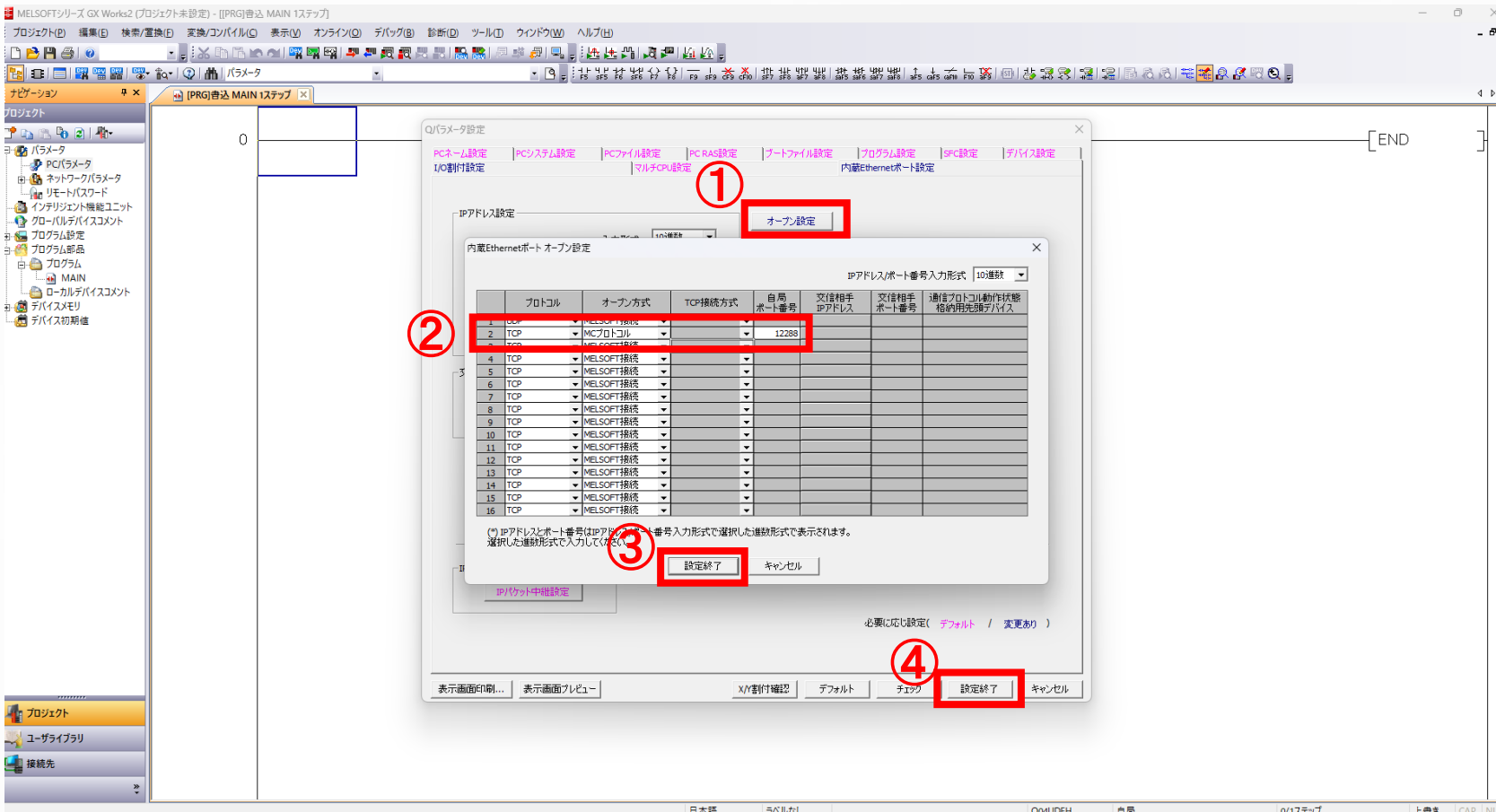


3. 内蔵Ethernetポートオープン設定

①オープン設定を開き、②2行目以降に設定を入力します。その後、③設定終了→④設定終了でウィンドウを閉じます。

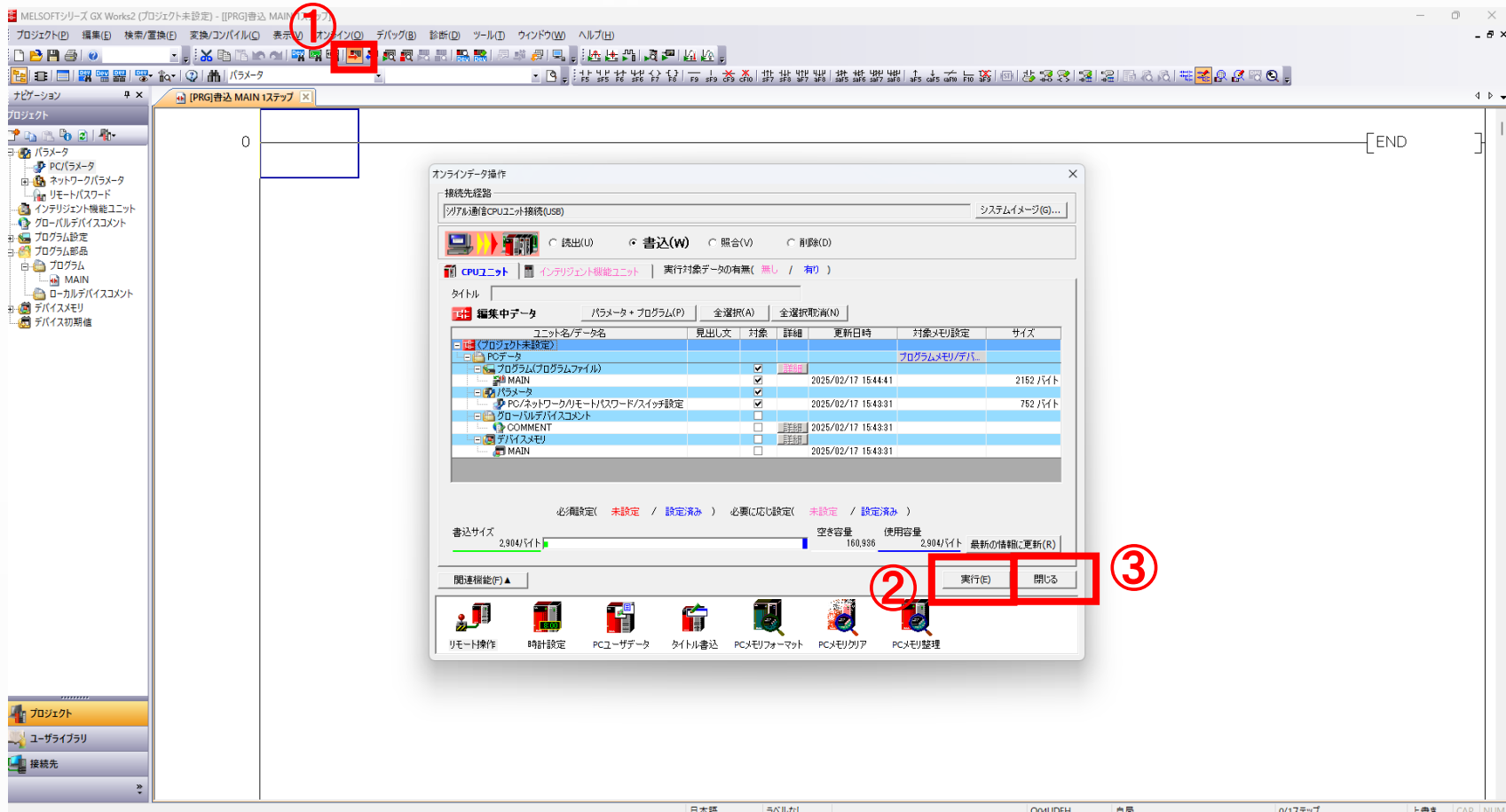
②内蔵Ethernetポートオープン設定を2行目以降に行います。

- プロトコル：TCP
- オープン方式：MCプロトコル
- 自局ポート番号：12288(10進数)



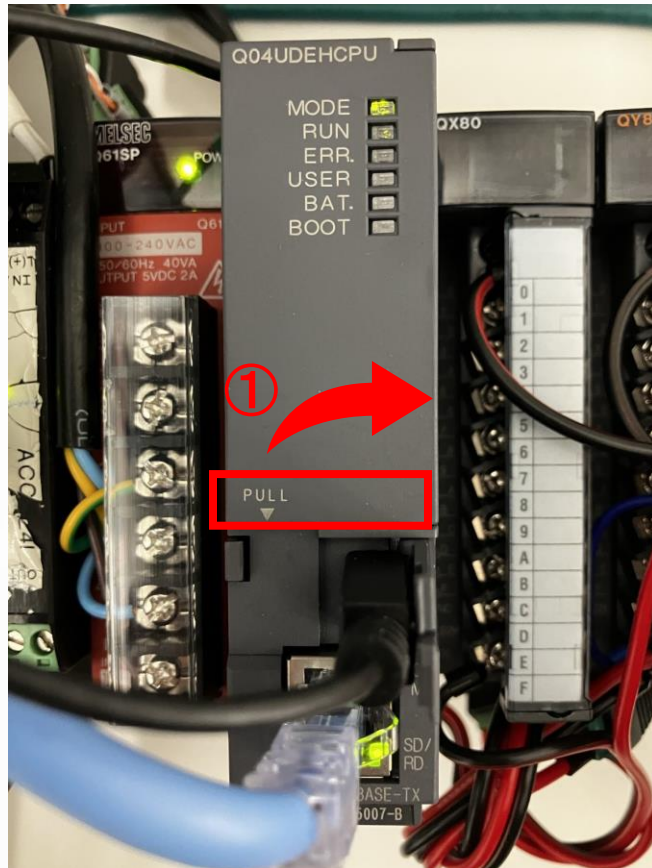
4. PLCへの設定書き込み

①PC書込みを選択し、②設定書き込みを実行したのち、③設定を閉じます。



5. PLC本体リセット

①PLCのCPUユニットを開き、②RESET側につまみを3秒程度倒します。リセットが完了するとカチッと音が鳴り、PLCが再起動します



6. SLMP接続確認

①診断→Ethernet診断を開き、②SLMP通信が確立していることを確認します。

Ethernet診断

対象ユニット指定
○ ユニットNo.(U) 1枚目 ② I/Oアドレス(I) 0000 CPU(M) 1号機

IPアドレス表示切替
● 10進(D) ○ 16進(H)

ポート番号
○ 10進(D) ● 16進(H)

パラメータ状態 | エラー履歴 | **コネクション別状態** | 回線状態 | 時刻設定状態

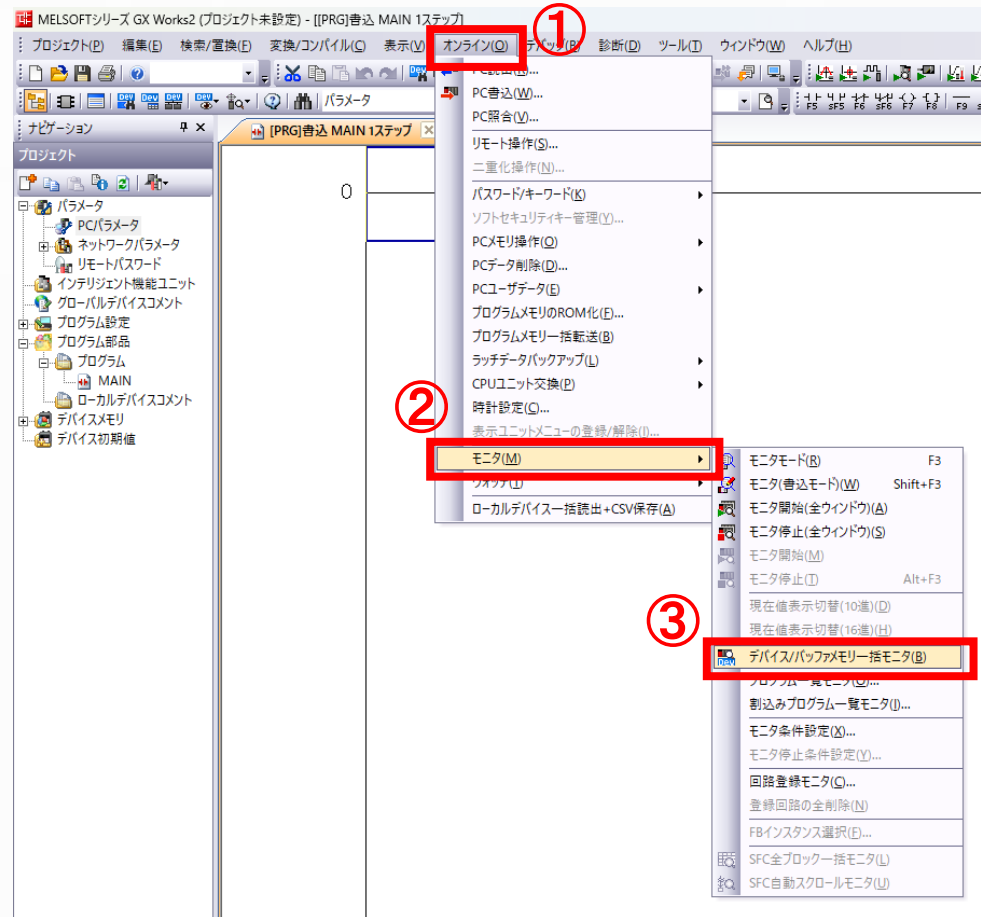
コネクションNo. /機能	自局 ポート番号	通信相手 IPアドレス	通信相手 ポート番号	最新 エラー コード	プロトコル	オープン 方式	TCP 状態	リモート パスワード 状態	アンロック 異常回数	強制 無効化 状態
1	---	0.0.0.0	---	---	UDP	MELSOFT接続	切断	無効、またはロック	0	許可
2	3000	192.168.0.30	7FC8	4183	TCP	MCプロトコル	接続中	無効、またはロック	0	許可
3	3001	0.0.0.0	---	---	TCP	MCプロトコル	切断	無効、またはロック	0	許可
4	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
5	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
6	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
7	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
8	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
9	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
10	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
11	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
12	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
13	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
14	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
15	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
16	---	0.0.0.0	---	---	TCP	MELSOFT接続	切断	無効、またはロック	0	許可
FTPサーバ	---	0.0.0.0	---	---	---	---	切断	無効、またはロック	0	許可
MELSOFTの直結接続	---	0.0.0.0	---	---	---	---	---	無効、またはロック	0	許可

最新エラーコードをクリア(E) アンロック異常回数をクリア(N) 選択行の無効化を解除(A) 選択行を強制無効化(F)

PINGテスト(P) 折返しテスト(L) COM.ERR消灯(C) モニタ開始(S) モニタ停止(T) 閉じる

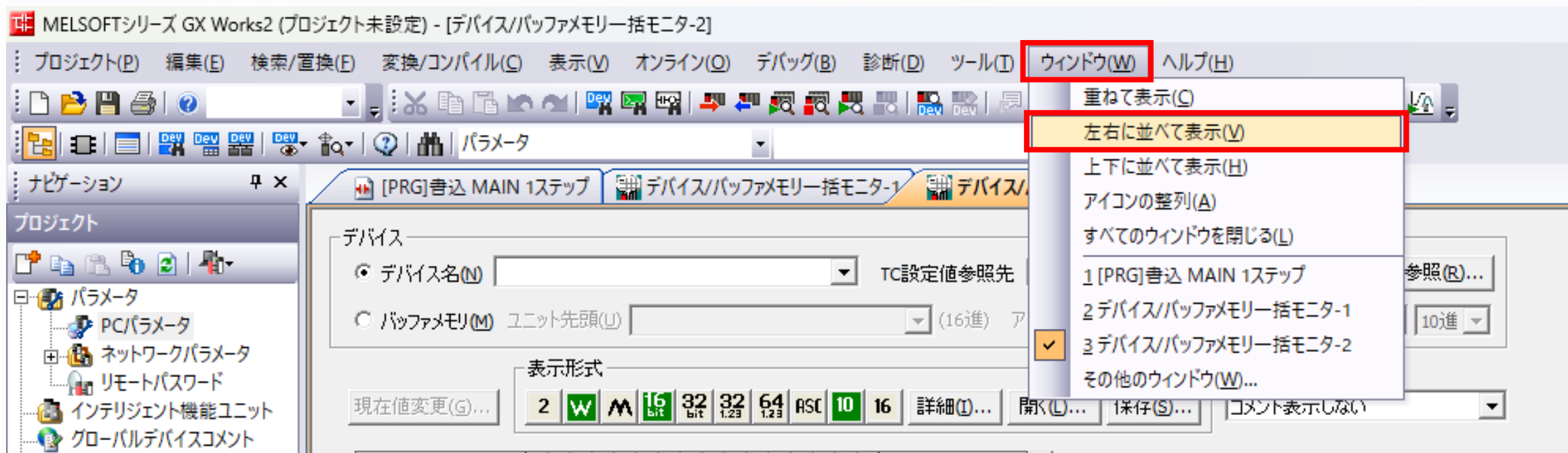
7. モニタリング設定1/3

①オンライン→②モニター→③デバイスバッファメモリー一括モニタを開きます。
同じ動作をもう一度繰り返します。



7. モニタリング設定2/3

ウィンドウ→左右に並べて表示を選択します。



7. モニタリング設定3/3

①左側のデバイスにM0、②右側のデバイスにD0と入力します。

The screenshot shows two side-by-side windows in MELSOFT GX Works2, both titled 'デバイス/バッファメモリ一括モニター' (Device/Buffer Memory Batch Monitor). The left window is for device 'M0' and the right window is for device 'D0'. Both windows show a table of bit addresses and their current states.

Left Window (Device M0):

- Device name: M0
- Table columns: 9, 8, 7, 6, 5, 4, 3, 2, 1, 0
- Table rows: M0 to M420
- Annotations:
 - M0~M32: コントロールブロック (Control Block)
 - M50~M72: ステータスブロック (Status Block)

Right Window (Device D0):

- Device name: D0
- Table columns: 9, 8, 7, 6, 5, 4, 3, 2, 1, 0
- Table rows: D0 to D42
- Annotations:
 - D0~D9: 入力ブロック (Input Block)
 - D10~D109: 出力ブロック (Output Block)

ジョブ切り替え

Summary

※あらかじめジョブファイル名の先頭に0~999の値が入ったジョブを複数作り、ビジョンセンサ内に保存してください。この数値がジョブIDとなります。

本資料では3job.jobxから1job.jobxの切り替えを行います。本資料の設定はあくまで一例ですので、実際の運用とは合致しない点がある旨をご了承ください。

タイミングチャート

初期状態

操作① : D0 ジョブID入力

操作② : M7 0→1(カメラオフライン)

操作③ : M4 0→1(コマンド入力)

ジョブ切り替わり

操作④ : M4 1→0

操作⑤ : M7 1→0(カメラオンライン)

タイミングチャート

操作は**入力ブロック**および**コントロールブロック**で行います。

Command ID(D0)

(入力ブロック)

※Job IDとして1をセット

Set Offline(M7)

(コントロールブロック)

Execute Command(M4)

(コントロールブロック)

※Complete/Fail確認後に落とす

Command Executing(M62)

(ステータスブロック)

※ジョブの切り替え時間

Command Complete(M63)

(ステータスブロック)

※失敗の場合はCommand Failed(M64)



初期状態

初期状態

- ①D0 ジョブID入力
- ②M7 0→1(カメラオフライン)
- ③M4 0→1(コマンド入力)
ジョブ切り替わり
- ④M4 1→0
- ⑤M7 1→0(カメラオンライン)

デバイス	9	8	7	6	5	4	3	2	1	0	デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値
M0	0	0	0	0	0	0	0	0	0	1	D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M10	0	0	0	0	0	0	0	0	0	0	D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M20	0	0	0	0	0	0	0	0	0	0	D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M30	0	0	0	0	0	0	0	0	0	0	D3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M40	0	0	0	0	0	0	0	0	0	0	D4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M50	0	0	1	0	0	0	0	0	0	1	D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M60	1	0	0	0	0	0	0	0	0	0	D6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M70	0	0	0	0	0	0	0	0	0	0	D7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
											D8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
											D9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
											D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
											D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

3job@IS3816MP-2d3bc4

×

ファイル (F) センサ(S) 表示 (M)

...

▶ オンライン

▼

3jobがセットされている状態からスタートします。

出力ブロックD10(Current Job ID)に現在ビジョンシステムにセットされているジョブのID(3)が表示されます。

操作①

初期状態

①D0 ジョブID入力

②M7 0→1(カメラオフライン)

③M4 0→1(コマンド入力)

ジョブ切り替わり

④M4 1→0

⑤M7 1→0(カメラオンライン)

①

デバイス	9	8	7	6	5	4	3	2	1	0	デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値	
M0	0	0	0	0	0	0	0	0	0	1	D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
M10	0	0	0	0	0	0	0	0	0	0	D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M20	0	0	0	0	0	0	0	0	0	0	D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M30	0	0	0	0	0	0	0	0	0	0	D3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M40	0	0	0	0	0	0	0	0	0	0	D4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M50	0	0	1	0	0	0	0	0	0	0	D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M60	1	0	0	0	0	0	0	0	0	0	D6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M70	0	0	0	0	0	0	0	0	0	0	D7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
											D8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	3	
											D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

In-Sight Vision Suite

3job@IS3816MP-2d3bc4

×

ファイル (F)

センサ(S)

表示 (V)

①D0(Command ID)にジョブIDを入力します。ここでは1jobに切り替えるため、1を入力しています。

操作②

初期状態

①D0 ジョブID入力

②M7 0→1(カメラオフライン)

③M4 0→1(コマンド入力)

ジョブ切り替わり

④M4 1→0

⑤M7 1→0(カメラオンライン)

②

デバイス	9	8	7	6	5	4	3	2	1	0	デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値		
M0	0	0	1	0	0	0	0	0	0	1	D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1		
M10	0	0	0	0	0	0	0	0	0	0	D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M20	0	0	0	0	0	0	0	0	0	0	D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M30	0	0	0	0	0	0	0	0	0	0	D3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M40	0	0	0	0	0	0	0	0	0	0	D4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M50	0	0	0	0	0	0	0	0	0	0	D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M60	1	0	0	0	0	0	0	0	0	0	D6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M70	0	0	0	0	0	0	0	0	0	0	D7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
											D8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	3	
											D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

In-Sight Vision Suite 3job@IS3816MP-2d3bc4 ×

ファイル (F) センサ(S) 表示 (M)

▶ オンライン

②M7(Set Offline)を立てて、カメラをオフライン強制状態(黄色)にします。このとき、M50(Triple Ready)およびM57(Online)は立ち下がり、M54とM55(Offline Reason)が立ちます。

操作③

初期状態

①D0 ジョブID入力

②M7 0→1(カメラオフライン)

③M4 0→1(コマンド入力)

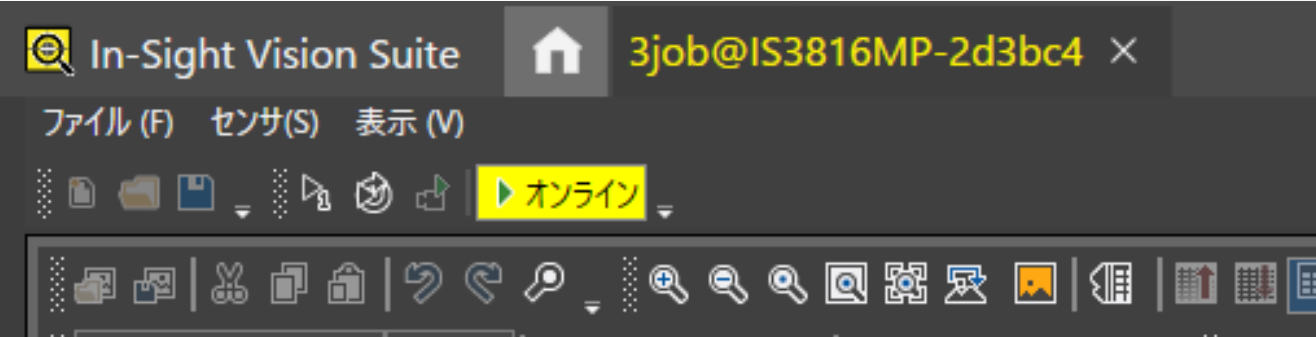
ジョブ切り替わり

④M4 1→0

⑤M7 1→0(カメラオンライン)

③

デバイス	9	8	7	6	5	4	3	2	1	0	デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値		
M0	0	0	1	0	0	1	0	0	0	1	D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1		
M10	0	0	0	0	0	0	0	0	0	0	D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M20	0	0	0	0	0	0	0	0	0	0	D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M30	0	0	0	0	0	0	0	0	0	0	D8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M40	0	0	0	0	0	0	0	0	0	0	D4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M50	0	0	0	0	1	1	0	0	0	0	D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M60	1	0	0	0	0	0	0	0	1	0	D6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M70	0	0	0	0	0	0	0	0	0	0	D7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
											D8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	3	
											D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



③M4(Execute Command)を立てます。このとき
M62(Command Executing)が立ちます。

ジョブ切り替わり

初期状態

- ①D0 ジョブID入力
- ②M7 0→1(カメラオフライン)
- ③M4 0→1(コマンド入力)

ジョブ切り替わり

- ④M4 1→0
- ⑤M7 1→0(カメラオンライン)

デバイス	9	8	7	6	5	4	3	2	1	0	デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値		
M0	0	0	1	0	0	1	0	0	0	1	D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
M10	0	0	0	0	0	0	0	0	0	0	D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M20	0	0	0	0	0	0	0	0	0	0	D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M30	0	0	0	0	0	0	0	0	0	0	D3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M40	0	0	0	0	0	0	0	0	0	0	D4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M50	0	0	0	0	0	0	0	0	0	0	D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M60	1	0	0	0	0	0	0	1	0	0	D6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M70	0	0	0	0	0	0	0	0	0	0	D7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
											D8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
											D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

In-Sight Vision Suite

1job@IS3816MP-2d3bc4

ファイル (F) センサ(S) 表示 (V)

しばらくするとM62(Command Executing)が立ち下がり、かわりに M63(Command Completed)が立ち、ジョブが切り替わります。

操作④

初期状態

- ①D0 ジョブID入力
- ②M7 0→1(カメラオフライン)
- ③M4 0→1(コマンド入力)
ジョブ切り替わり
- ④M4 1→0
- ⑤M7 1→0(カメラオンライン)

デバイス	9	8	7	6	5	4	3	2	1	0
M0	0	0	1	0	0	0	0	0	0	1
M10	0	0	0	0	0	0	0	0	0	0
M20	0	0	0	0	0	0	0	0	0	0
M30	0	0	0	0	0	0	0	0	0	0
M40	0	0	0	0	0	0	0	0	0	0
M50	0	0	0	0	1	1	0	0	0	0
M60	1	0	0	0	0	0	0	0	0	0
M70	0	0	0	0	0	0	0	0	0	0

デバイス名 F E D C B A 9 8 7 6 5 4 3 2 1 0 現在値

D0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1

D1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

D2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

D3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

D4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

D5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

D6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

D7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

D8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

D9 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

D10 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

D11 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

④M4(Execute Command)を立ち下げます。
M63(Command Completed)が連動して立ち下がります。

操作⑤

初期状態

- ①D0 ジョブID入力
- ②M7 0→1(カメラオフライン)
- ③M4 0→1(コマンド入力)
ジョブ切り替わり
- ④M4 1→0
- ⑤M7 1→0(カメラオンライン)

⑤

デバイス	9	8	7	6	5	4	3	2	1	0	デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値	
M0	0	0	0	0	0	0	0	0	0	1	D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
M10	0	0	0	0	0	0	0	0	0	0	D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M20	0	0	0	0	0	0	0	0	0	0	D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M30	0	0	0	0	0	0	0	0	0	0	D3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M40	0	0	0	0	0	0	0	0	0	0	D4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M50	0	0	1	0	0	0	0	0	0	1	D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M60	1	0	0	0	0	0	0	0	0	0	D6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
M70	0	0	0	0	0	0	0	0	0	0	D7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
											D8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
											D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
											D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

In-Sight Vision Suite 1job@IS3816MP-2d3bc4 ×

ファイル(F) センサ(S) 表示(M)

オンライン

⑤M7(Set Offline)を立ち下げます。

M54,M55(Offline Reason)は立ち下がり、M50(Trieger Ready)とM57(Online)が立ち上がります。

トリガ毎のデータ送受信
ジョブ作成
EasyBuilder編

Summary

1. データ入出力について
2. EasyBuilder起動
3. 検査ジョブ作成
4. データ入力設定
5. ジョブの保存と起動設定
6. データ入力動作確認
7. データ出力のためのジョブ編集
8. データ出力設定
9. データ出力動作確認
10. データ入力タイミングチャート
11. データ出力タイミングチャート

1. データ入出力について

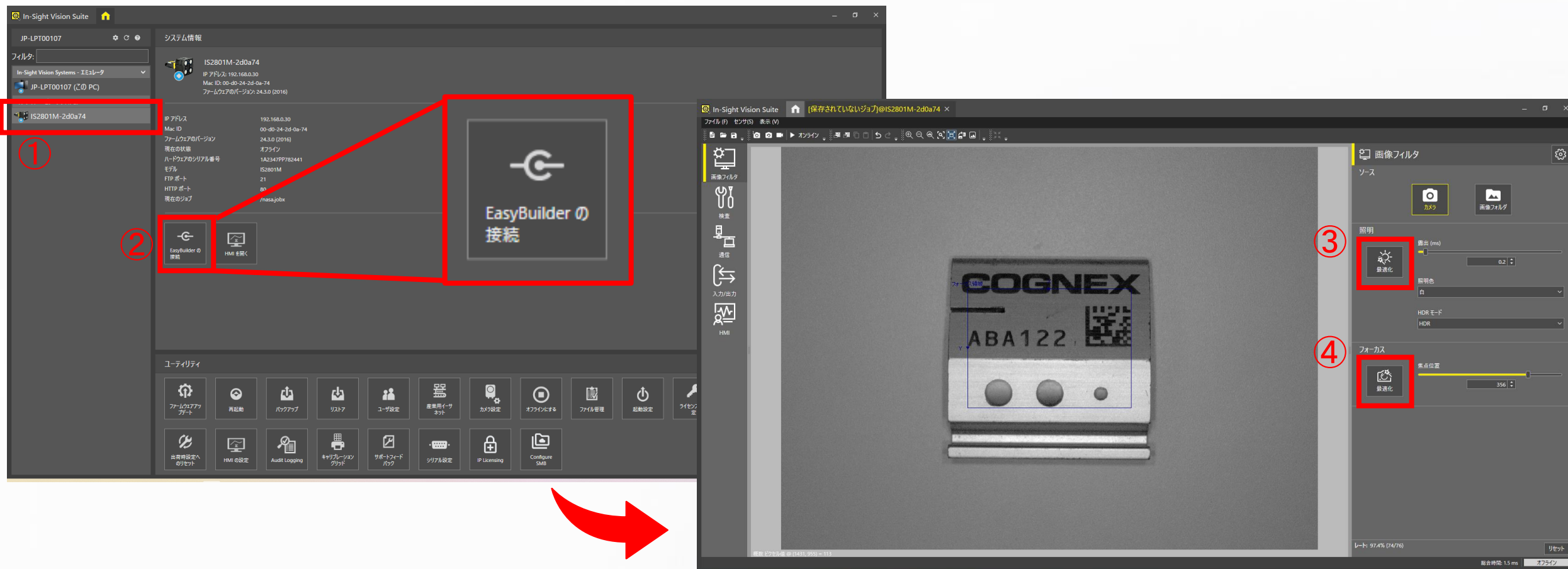
- ビジョンセンサ内での実行順は下記の通りです。

画像取り込み → データ入力 → ジョブ実行 → データ出力

- 本資料では、まずPLCからビジョンセンサへデータ入力を行います。次に、入力されたデータとビジョンセンサが測定した結果をビジョンセンサ内で照合し、合否をPLCへ出力するジョブを作成することを目的として各種設定を行います。
- なお、本資料の設定はあくまで一例ですので、実際の運用とは合致しない点がある旨をご了承ください。

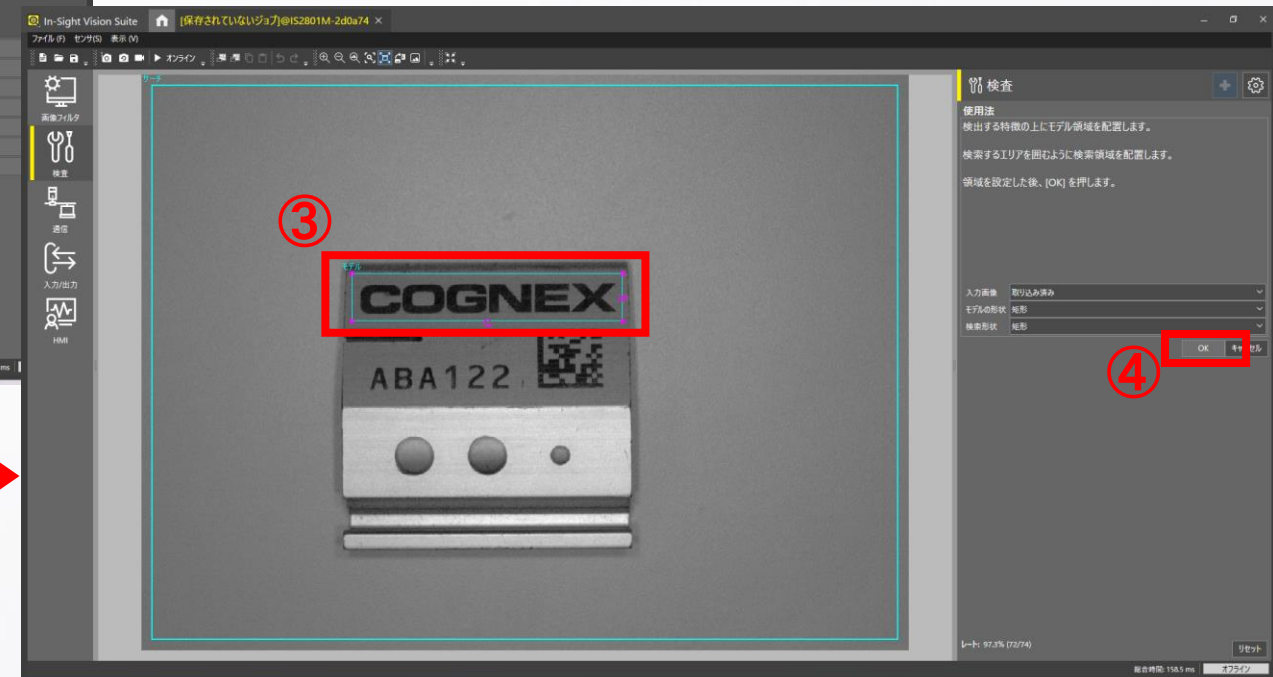
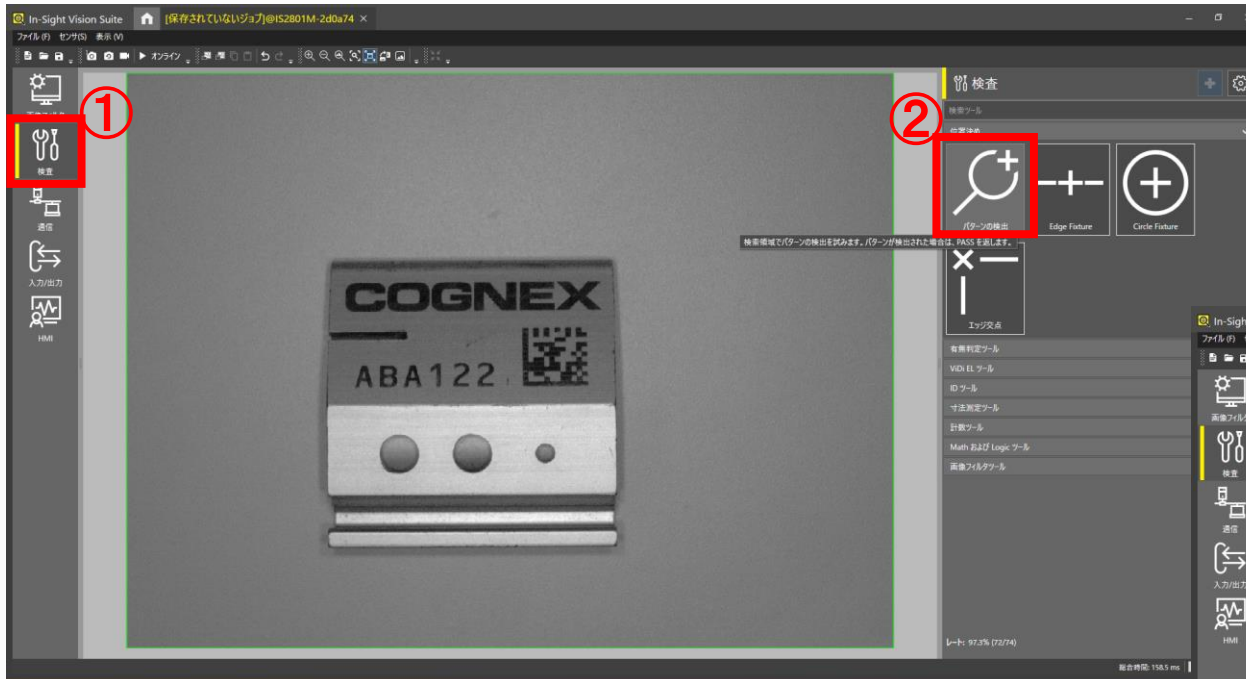
2. EasyBuilder起動

In-Sight Vision Suiteを立ち上げ、ビジョンセンサ(今回はIn-Sight 2800)を選択し、EasyBuilderの接続を選択します。また、照明とフォーカスを最適化しておきます。



3. 検査ジョブ作成1(パターンマッチ)

検査→「パターンの検出」を選択し、「COGNEX」の文字列を登録します。



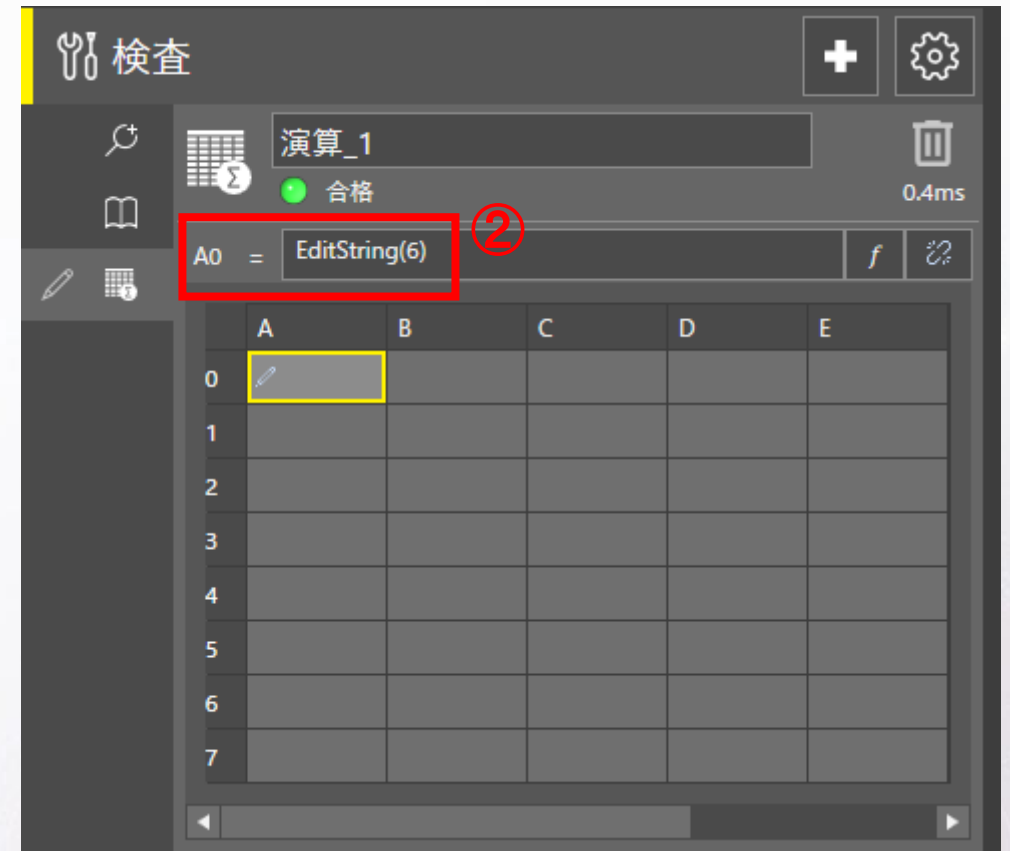
3. 検査ジョブ作成2(OCR)

ViDi ELツール内の「読み取りツール」を選択し、適切な検査領域と文字サイズを設定します。



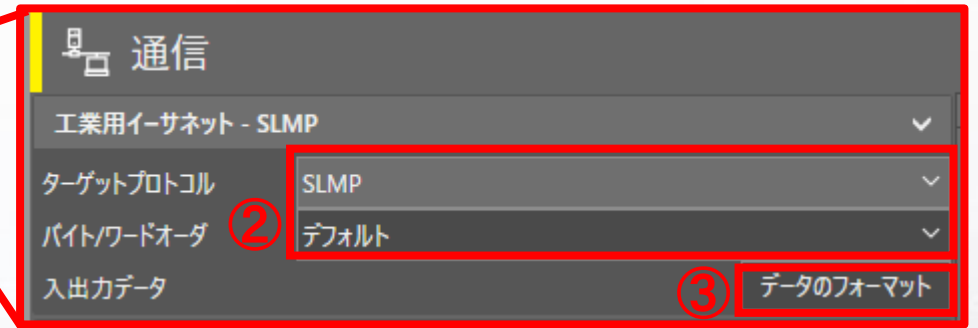
3. 検査ジョブ作成3(Mathツール)

MathおよびLogicツールの「演算」ツールを選択し、A0セルに”EditString(6)”と入力します。



4. データ入力設定1

- ①アプリケーションステップから「通信」を選択し、工業用イーサネット設定を行います。
- ②ターゲットプロトコル：SLMP、バイト/ワードオーダ：デフォルトを選択し、③データのフォーマット設定に移ります。



4. データ入力設定2

上側「入力」画面にて、右上の「+」ボタンから下記のように入力データを追加します。
設定終了後、右上「×」ボタンでフォーマット設定を抜けます。

1行目：パターン1 → アクセプトしきい値

2行目：演算1 → A0 (バイト数を6に指定)

入力	データタイプ	バイト数	値
パターン1.アクセプトしきい値	16ビット 整数	2	0
演算1.A0	文字列	6	0

データの追加、削除、入力順の変更ができます。

データタイプについては次ページで解説します

4. データ入力設定3

それぞれのデータタイプにおいてPLCがビジョンセンサに入力する値の形式は表の通りです。

データタイプ	バイト数	セルに入力される値の形式
ビット	1	0/1
8ビット整数	1	-128 ~ 127
8ビット符号なし整数	1	0 ~ 255
16ビット整数	2	-32,768 ~ 32,767
16ビット符号なし整数	2	0 ~ 65,535
32ビット整数	4	-2,147,483,648 ~ 2,147,483,647
32ビット浮動小数点数	4	単精度実数
文字列	任意	文字列※
バイト変換文字列	任意	文字列※

※セルにはASCII変換された文字列が入力されます。

「文字列」はPLCワード内で15~8ビットに1文字目、7~0ビットに2文字目が割り当てられて入力されます。

「バイト変換文字列」はPLCワード内で7~0ビットに1文字目、15~8ビットに2文字目が割り当てられて入力されます。

5. ジョブの保存と起動設定

- ①ファイル→ジョブに名前を付けて保存(本資料では3job.jobx)します。
- ②ステータスをオンラインにしてジョブを閉じます。
- ③ホーム画面の起動設定をクリックします。
- ④起動モードをオンライン、ジョブの起動パスを3job.jobxに設定します。

① In-Sight Vision Suite の「ファイル」メニューから「ジョブに名前を付けて保存」を選択します。

② In-Sight Vision Suite のジョブステータスを「オンライン」に変更します。

③ ホーム画面の「起動設定」アイコンをクリックします。

④ 「起動設定」ダイアログで、起動モード(S)を「オンライン」にし、ジョブの起動パス(P)を「/3job.jobx」に設定します。

6. データ入力動作確認

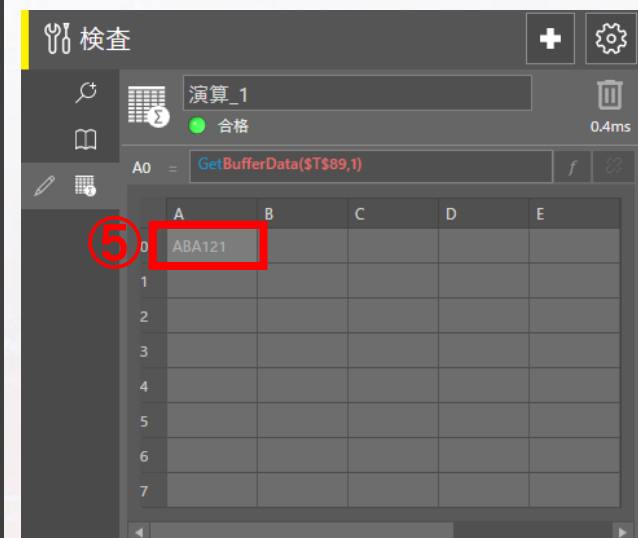
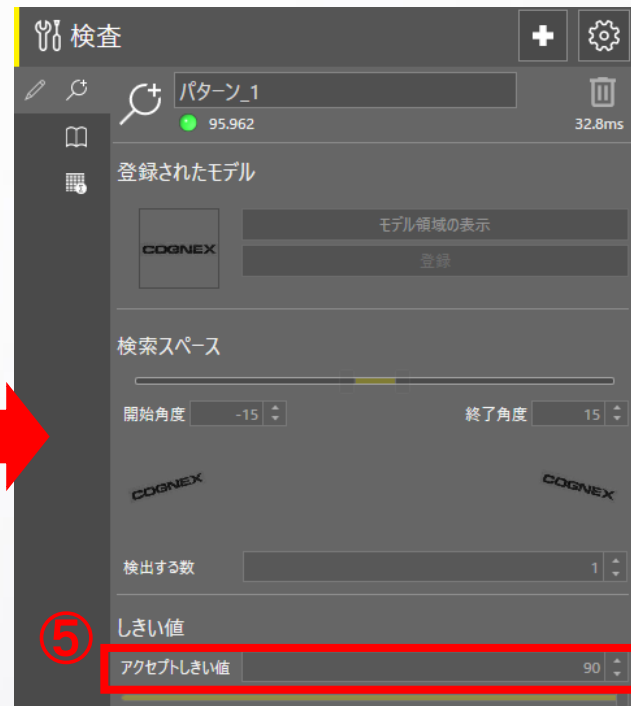
データ入力をするには入力ブロックの[User Data] (今回はD2~D9)にデータを書き込みます。
今回はD2へ「90」という文字列を、D3~D5へ「ABA122」という数値を書き込むため、
①GX Works 2にて下図のようにビットを立てます。(表示はASCIIにしています。)

次にコントロールブロックにて、M0(Trigger Enable)が立っている状態で、②M16(Set User Data)、③M2(Buffer Results Enable)のビットを順番に立て、④最後にM1(Trigger)のビットを立てると、⑤入力データが反映されます。

1011010(2進数)= 90(10進数)

デバイス	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	
D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D2	①	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	0
D3	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1
D4	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	0	1
D5	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	22

デバイス	9	8	7	6	5	4	3	2	④	③	0
M0	0	0	0	0	0	0	0	0	1	1	1
M10	0	0	0	1	0	0	0	0	0	0	0
M16	0	0	0	0	0	0	0	0	0	0	0

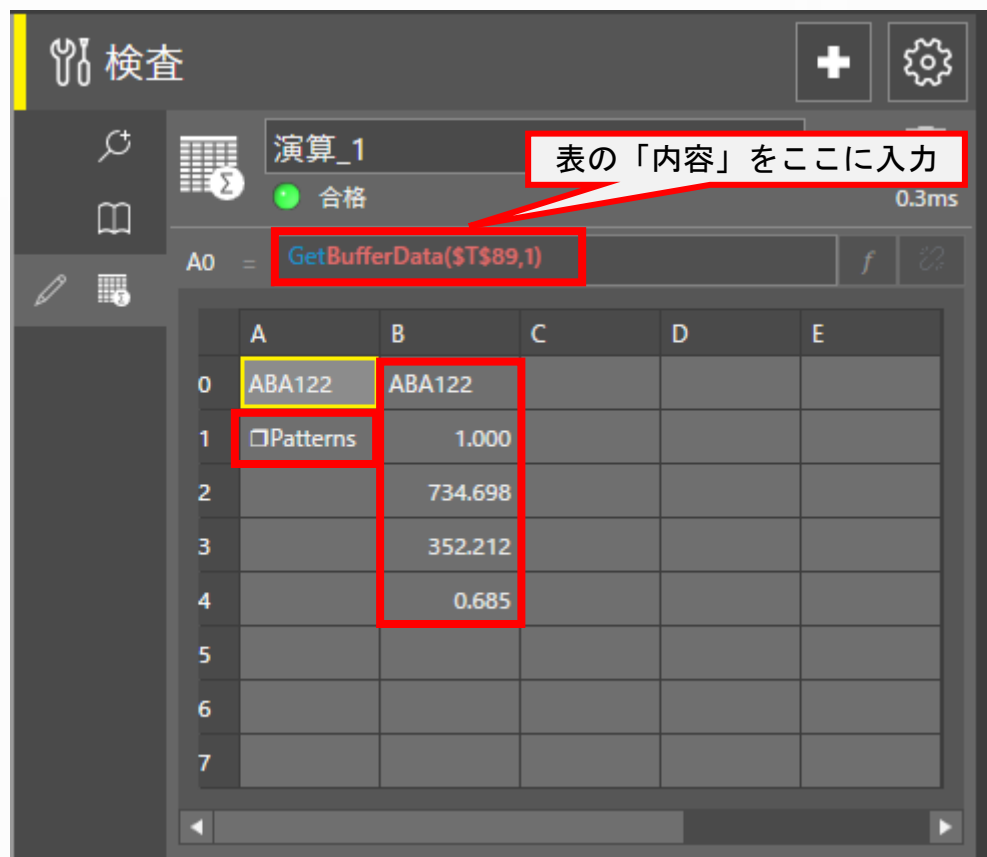


7. データ出力のためのジョブ編集

次に、データ出力のためのジョブ編集を行います。

In-Sight Vision Suiteにて、ジョブをオフラインにし、演算ツールを編集します。

各セルに下表「内容」のように入力します。



セル	内容	説明
A1	パターン_1.OutputFixtures	パターン構造体を示します。
B0	読み取り_1.ReadText	OCRの読み取り結果を示します。
B1	Exact(A0, B0)	A0とB0の文字列照合結果を出力します。 (照合○→1、照合×→0)
B2	GetX(A1)	登録パターンの重心X座標を示します。
B3	GetY(A1)	登録パターンの重心Y座標を示します。
B4	GetAngle(A1)	登録パターンの角度を示します。

8. データ出力設定1

①アプリケーションステップから「通信」を選択し、②データのフォーマットを開きます。



8. データ出力設定2

下側「出力」画面にて、右上の「+」ボタンから下記のように出力データを追加します。
設定終了後、右上「×」ボタンでフォーマット設定を抜け、
ジョブを保存してオンラインに戻します。

- 1行目：演算1 → B0 (バイト数を6に指定)
- 2行目：演算1 → B1 (データタイプをビットに指定)
- 3行目：演算1 → B2
- 4行目：演算1 → B3
- 5行目：演算1 → B4



②

出力	データタイプ	バイト数	値
演算1.B0	文字列	6	ABA122
演算1.B1	ビット	1	1.000
演算1.B2	32ビット 浮動小数点数	4	734.698
演算1.B3	32ビット 浮動小数点数	4	352.212
演算1.B4	32ビット 浮動小数点数	4	0.685

③

④

データタイプについては
次ページで解説します

①

データの追加、
削除、入力順
の変更ができます。

+
-
↑
↓

00000 41 42 41 31 32 32 01 00 ABA122..
00008 a9 ac 37 44 1b 1b b0 43 ..7D...C
00016 6f 75 2f 3f ou/?

8. データ出力設定3

それぞれのデータタイプにおいてビジョンセンサがPLCに出力する値の形式は表の通りです。

データタイプ	バイト数	出力するセルの値の形式
ビット	1	0または1
8ビット整数	1	-128 ~ 127
8ビット符号なし整数	1	0 ~ 255
16ビット整数	2	-32,768 ~ 32,767
16ビット符号なし整数	2	0 ~ 65535
32ビット整数	4	-2,147,483,648 ~ 2,147,483,647
32ビット浮動小数点数	4	単精度実数
文字列	任意	文字列※
バイト変換文字列	任意	文字列※

※PLCにはASCIIコードが出力されます。

「文字列」はPLCワード内で15~8ビットに1文字目、7~0ビットに2文字目が出力されます。

「バイト変換文字列」はPLCワード内で7~0ビットに1文字目、15~8ビットに2文字目が出力されます。

9. データ出力動作確認1

GX Works 2にて、M0(Triiger Enable)が立っている状態でM1(Trigger)を立てると、ビジョンセンサからPLC(D16~)にデータが出力されます。



検査

演算_1
合格 0.3ms

B1 = Exact(A0, B0)

	A	B	C	D	E
0	ABA122	ABA122			
1	Patterns	1.000			
2		685.184			
3		383.835			
4		0.067			
5					
6					
7					

※ASCII文字表示

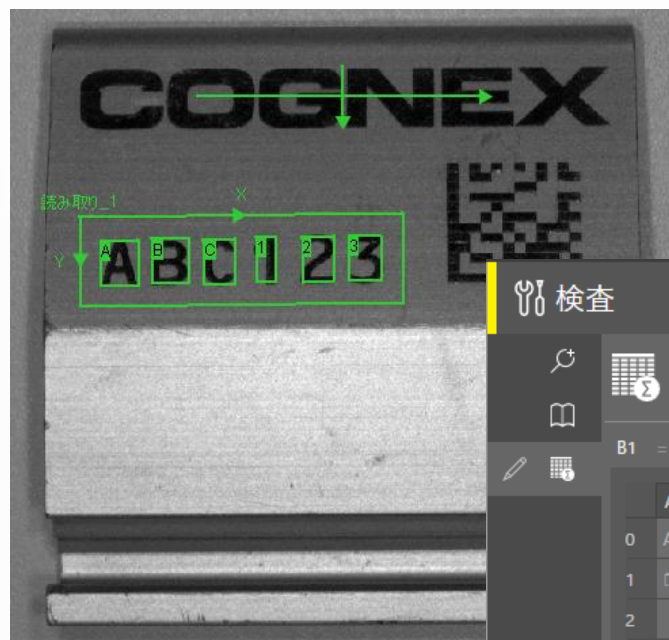
D16	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	AB
D17	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	1	A1
D18	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	22
D19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	..
D20	0	1	0	0	1	0	1	1	1	1	0	0	0	1	0	0	TK
D21	0	1	0	0	0	1	0	0	0	0	1	0	1	0	1	1	+D
D22	1	1	1	0	1	0	1	0	1	1	0	1	1	0	0	1	ル.
D23	0	1	0	0	0	0	1	1	1	0	1	1	1	1	1	1	YC
D24	0	1	0	0	1	1	1	0	1	1	0	1	1	0	1	0	LN
D25	0	0	1	1	1	1	0	1	1	0	0	0	1	0	0	0	. =

※32ビット単精度実数表示

D16	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	2.812286E-009
D17	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	1	
D18	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	-----
D19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
D20	0	1	0	0	1	0	1	1	1	1	0	0	0	1	0	0	685.1838379
D21	0	1	0	0	0	1	0	0	0	0	1	0	1	0	1	1	
D22	1	1	1	0	1	0	1	0	1	1	0	1	1	0	0	1	383.8347473
D23	0	1	0	0	0	0	1	1	1	0	1	1	1	1	1	1	
D24	0	1	0	0	1	1	1	0	1	1	0	1	1	0	1	0	0.0665566
D25	0	0	1	1	1	1	0	1	1	0	0	0	1	0	0	0	

9. データ出力動作確認1

印字の異なるワークでは、文字列一致の出力が0となります。



検査				
演算_1				
合格				
0.3ms				
B1 = Exact(A0, B0)				
A	B	C	文字列	E
0 ABA122	ABC123			
1 Patterns	0.000			
2	675.022		文字列一致	
3	397.749			
4	0.005		ワーク座標	
5				
6				
7				

※ASCII文字表示

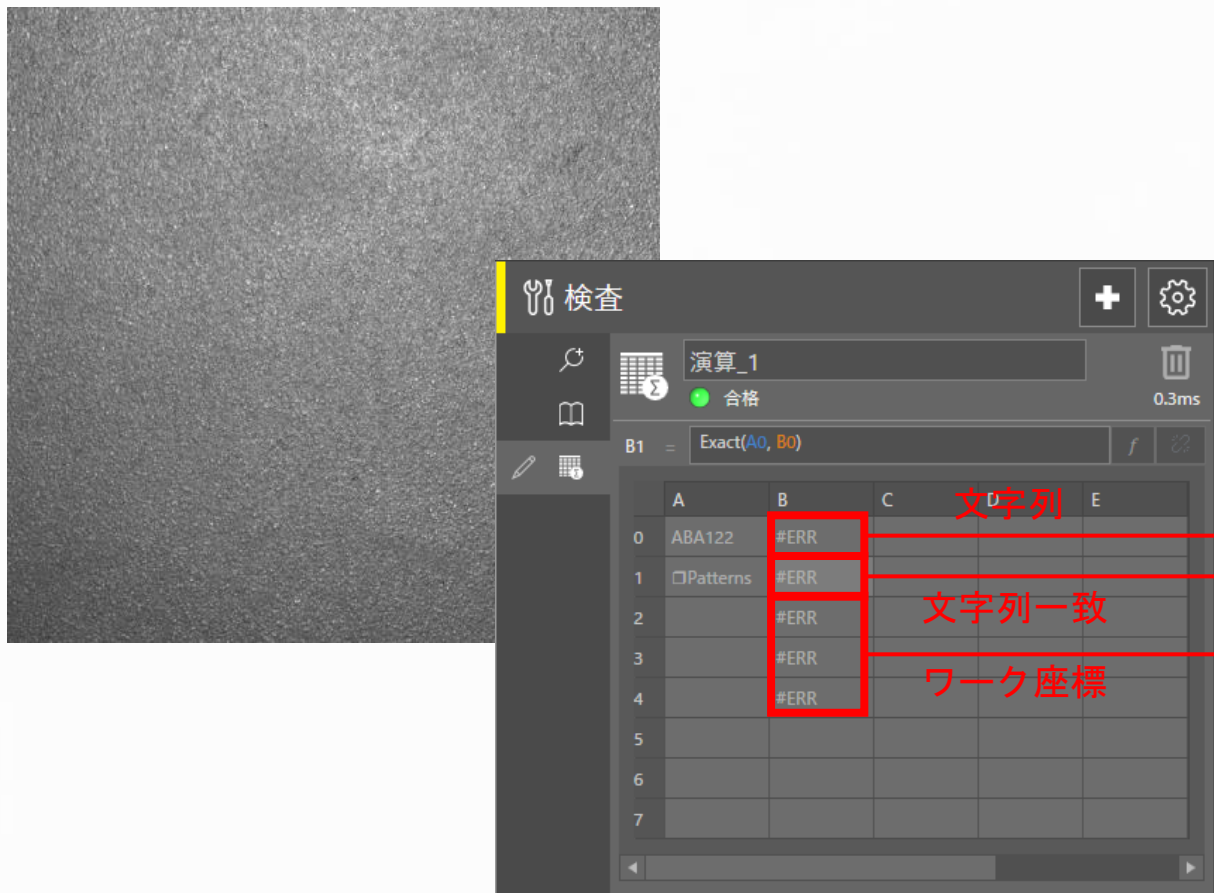
D16	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	AB
D17	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	1	C1
D18	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	28
D19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D20	1	1	0	0	0	0	0	1	0	1	1	0	0	0	0	1	c7
D21	0	1	0	0	0	1	0	0	0	0	1	0	1	0	0	0	(D
D22	1	1	0	1	1	1	1	1	1	1	0	1	0	0	1	1	モ°
D23	0	1	0	0	0	0	1	1	1	1	0	0	0	1	1	0	二C
D24	0	0	0	1	0	1	0	0	1	0	0	1	1	0	0	0	..
D25	0	0	1	1	1	0	1	1	1	0	0	1	0	1	0	1	..

※32ビット単精度実数表示

D16	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	2.841390E-009
D17	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	1	
D18	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	-----
D19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
D20	1	1	0	0	0	0	0	1	0	1	1	0	0	0	0	1	675.0216675
D21	0	1	0	0	0	1	0	0	0	0	1	0	1	0	0	0	
D22	1	1	0	1	1	1	1	1	1	1	0	1	0	0	1	1	397.7486267
D23	0	1	0	0	0	0	1	1	1	1	0	0	0	1	1	0	
D24	0	0	0	1	0	1	0	0	1	0	0	1	1	0	0	0	0.0045496
D25	0	0	1	1	1	0	1	1	1	0	0	1	0	1	0	1	

9. データ出力動作確認3

ワークがない場合は、データが出力されません。



※ASCII文字表示

[illegible]

※32ビット単精度実数表示

[illegible]

10. データ入力タイミングチャート

操作は**入力ブロック**および**コントロールブロック**で行います。

User Data
(入力ブロック)

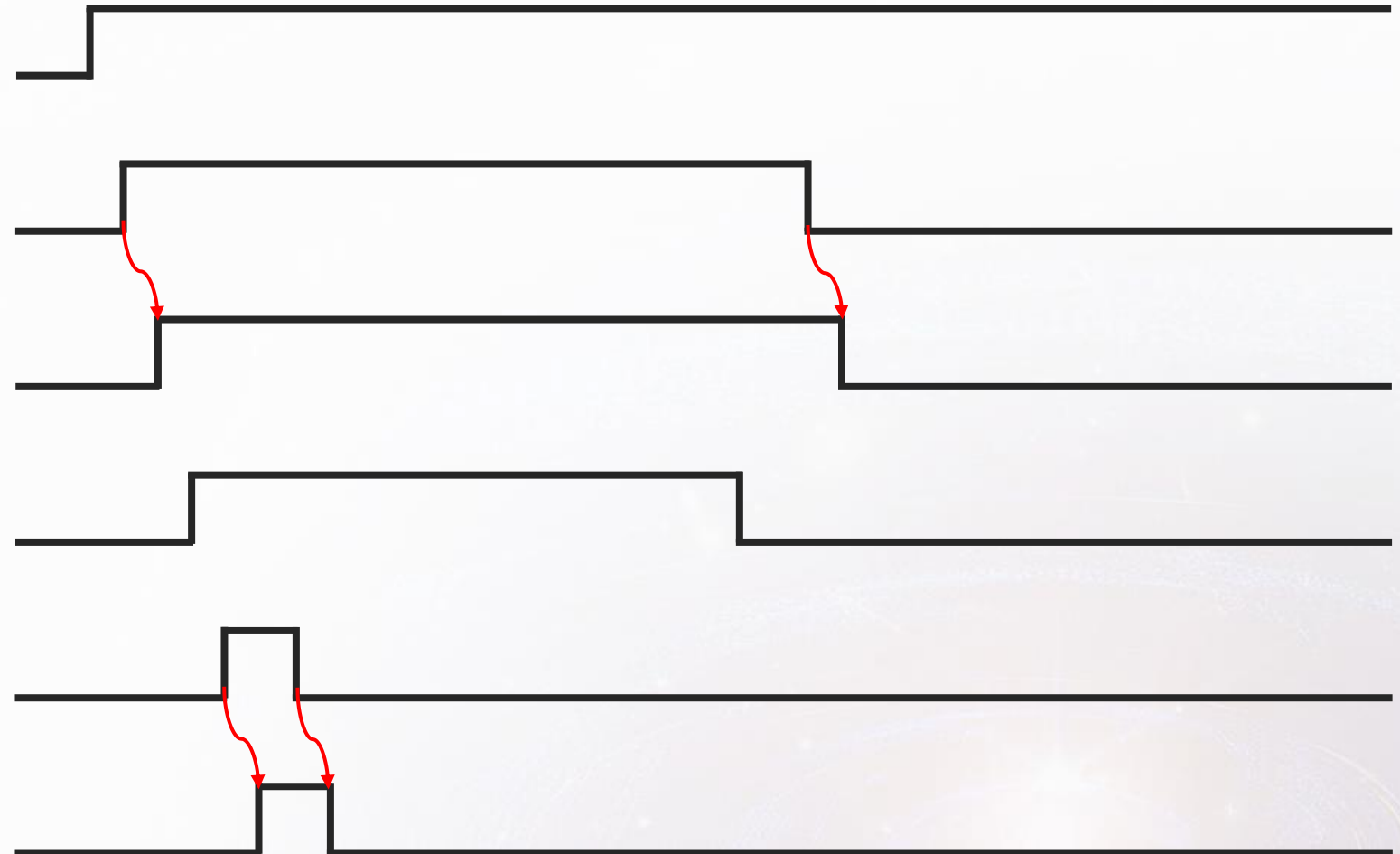
Set User Data(M16)
(コントロールブロック)

Set User Data Ack(M66)
(ステータスブロック)

Buffer Results Enable(M2)
(コントロールブロック)

Trigger(M1)
(コントロールブロック)

Result Valid(M61)
(ステータスブロック)



11. データ出力タイミングチャート

Trigger Ready
(Highの時に、Triggerを入力可能)

Trigger
(Trigger Ack立ち上がりでLowへ)

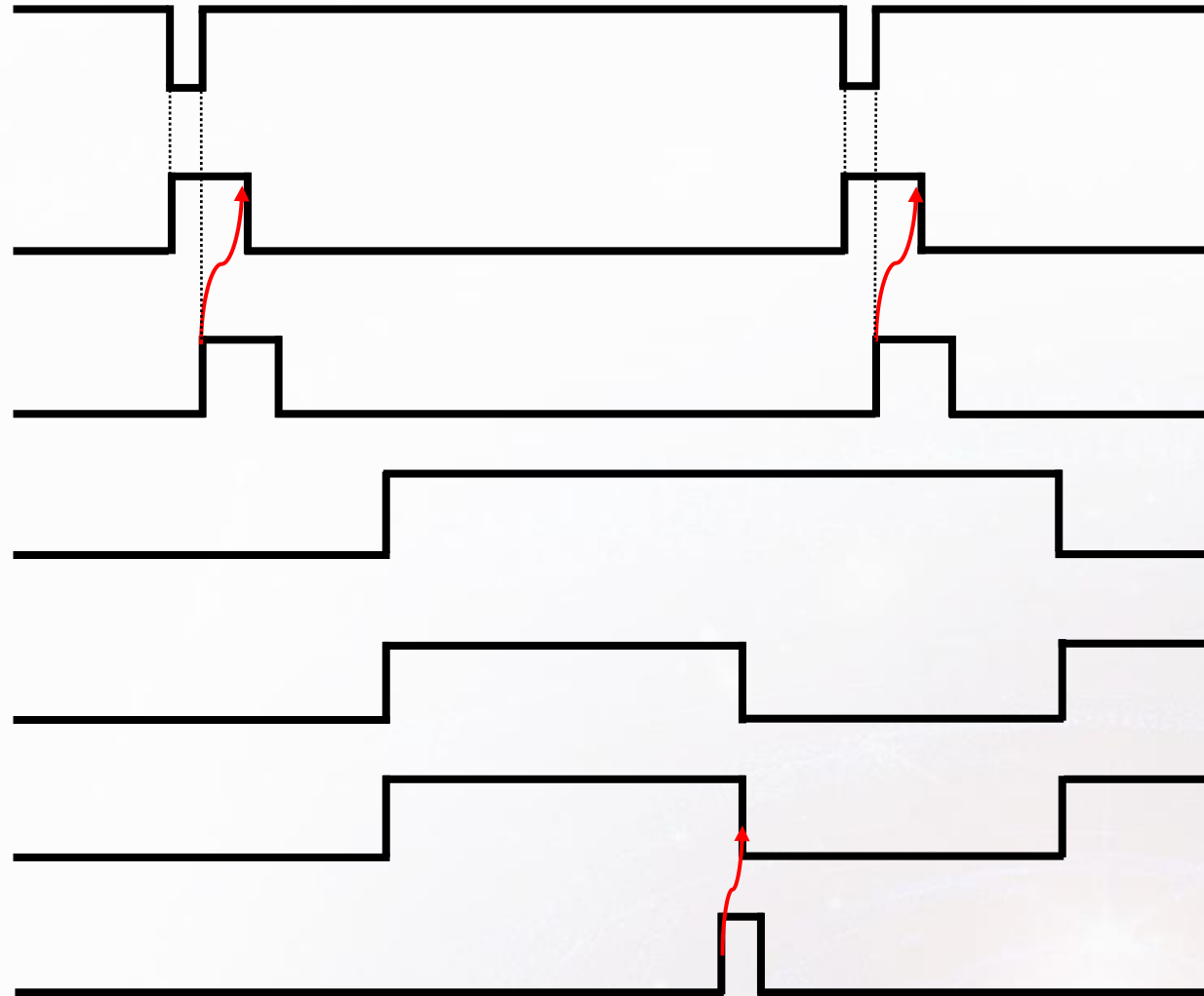
Trigger Ack

Inspection Completed
(トグル動作)

Inspection Results

Results Valid

Results Ack



トリガ毎のデータ送受信
ジョブ作成
Spreadsheet編

Summary

1. 入出力関数について
2. Spreadsheet 起動
3. 出力用データの関数作成
4. データ出力関数
5. ジョブの保存と起動設定
6. データ出力関数動作確認
7. データ入力関数
8. 入力されたデータの利用
9. データ入力関数動作確認
10. データ入力タイミングチャート

1. 入出力関数について

- In-Sight Vision Suite Spreadsheetにて、ビジョンシステムとPLCとの間でデータをやり取りするときは、Spreadsheetにて下記の関数を使用します。

PLCからビジョンシステムへの入力：FormatInputBuffer, ReadUserData, GetBufferData

ビジョンシステムからPLCへの出力：FormatOutputBuffer, WriteResultsBuffer

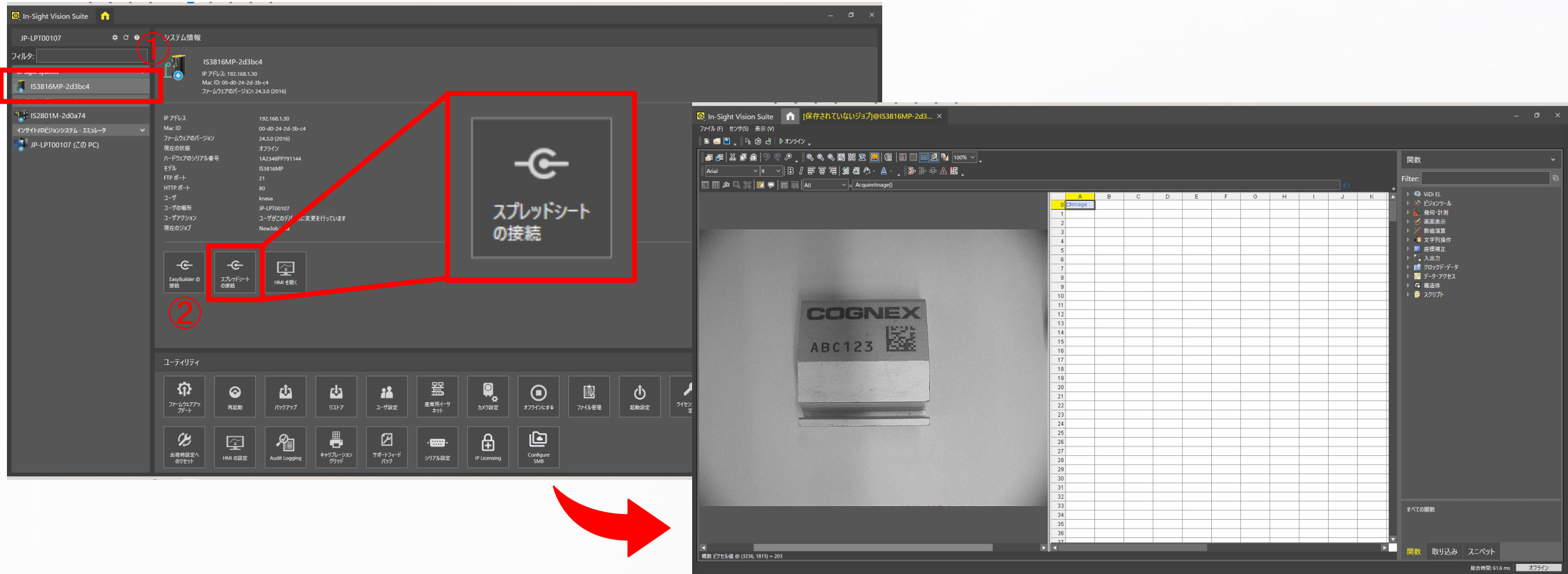
- ビジョンシステム内での実行順は下記の通りです。

画像取り込み → データ入力 → ジョブ実行 → データ出力

- 本資料では、まずビジョンシステムからPLCへのデータ出力を行います。次にPLCからビジョンシステムへデータ入力を行います。最後に、入力されたデータとビジョンシステムが測定した結果をビジョンシステム内で照合し、合否をPLCへ出力するジョブを作成することを目的として各種設定を行います。
- 尚、本資料の設定はあくまで一例ですので、実際の運用とは合致しない点がある旨をご了承ください。

2. Spreadsheet起動

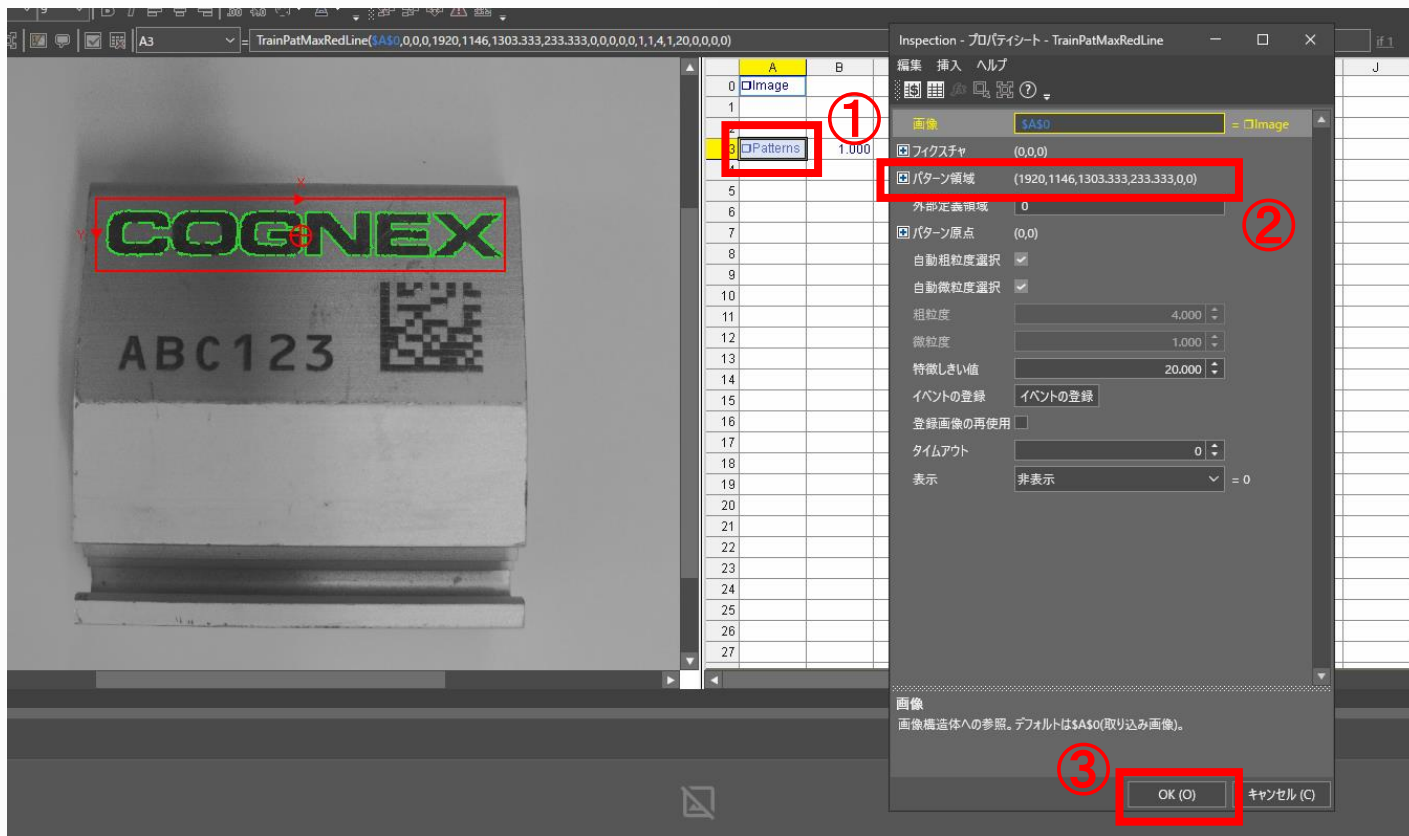
In-Sight Vision Suiteを立ち上げ、ビジョンセンサ(今回はIn-Sight 3800)を選択し、スプレッドシートの接続を選択します。



3. 出力用データの関数作成1/4(位置座標)

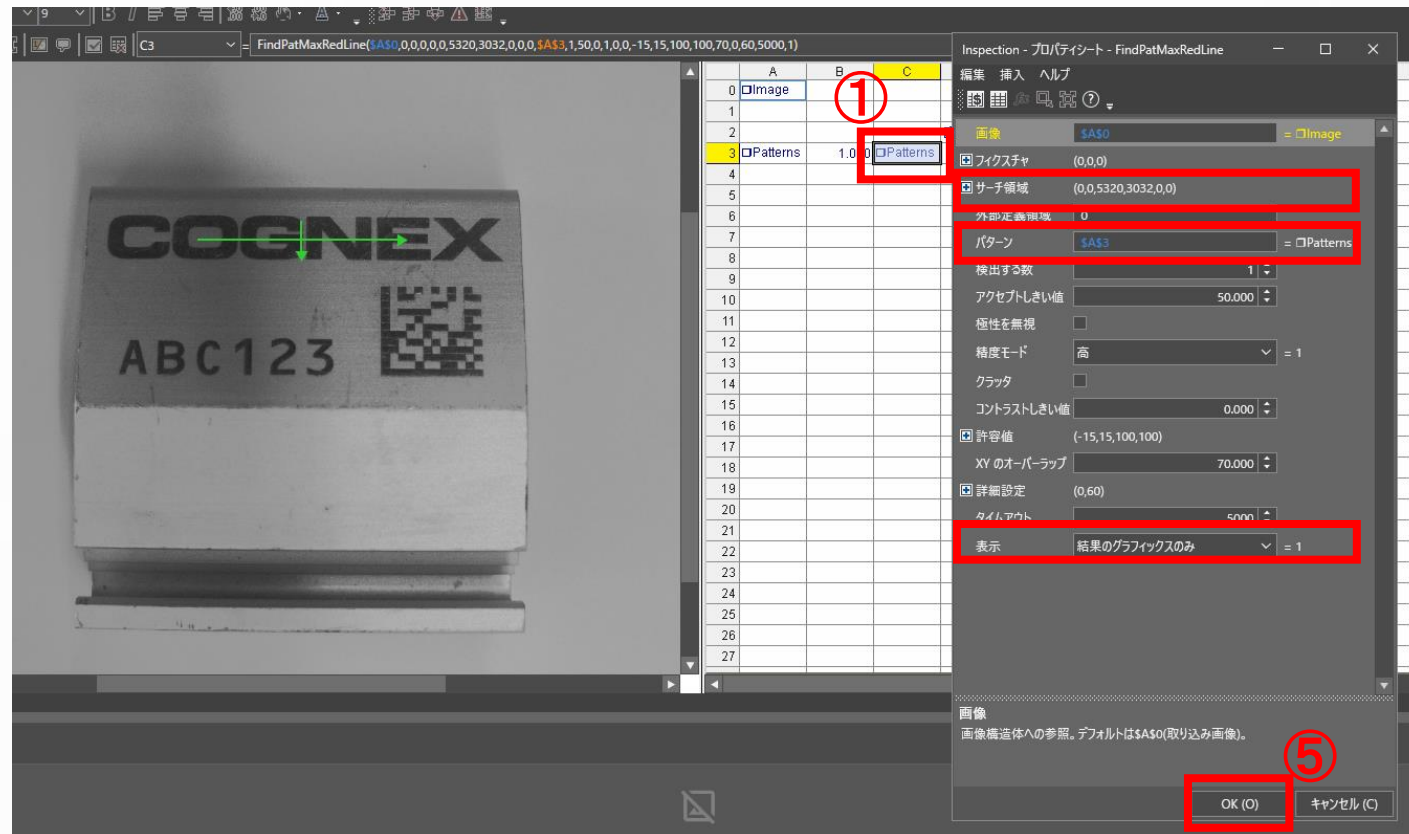
まず、PLCへ出力したいデータを得るための関数を作成します。本資料ではワークの位置座標および角度をPLCへ出力します。

①A3セルにTrainPatMaxRedLine関数を挿入し、②パターン領域を左図赤枠のように設定し、③OKを押します。



3. 出力用データの関数作成2/4(位置座標)

次に、①C3セルにFindPatMaxRedLine関数を挿入し、②サーチ領域で全領域、③パターンでA3セル(先ほどのTrainPatMaxRedLine関数のセル)を参照します。また、④表示は結果をグラフィックスのみに設定し、⑤OKを押します。



3. 出力用データの関数作成3/4(文字列)

①A6セルにViDiELRead関数を挿入し、②フィクスチャX座標Y座標角度でそれぞれ③E3, F3, G3セルを参照します。④領域は左図のように文字の範囲を選択し、学習モードは「Pre-Trained Model v2 Standard」を選択して⑤文字サイズを設定します。最後に⑥OKを押します。

The screenshot displays the Cognex ViDiELRead software interface. On the left, a grayscale image of a label with the text 'A B C 1 2 3' is shown. A red box highlights the text area, and a green box highlights the characters 'A B C 1 2 3'. The main window shows the 'Inspection - プロパティシート - ViDiELRead' dialog. The 'フィクスチャ' (Fixture) section is set to '\$A\$0'. The '領域' (Area) section is set to '(-627.463,259.492,756.667,203.333,0.564)'. The '学習モード' (Learning Mode) is set to 'Pre-Trained Model v2 Standard'. The '文字サイズ' (Text Size) is set to 2. The 'フィールド化の一致' (Field Consistency) is set to '一致' (Consistent). The 'サンプル' (Sample) is set to '0'. The 'OK' button is highlighted with a red box.

番号	X	Y	角度	スケール	得点	検出数			
Patterns	1.000	Patterns	0.000	2746.847	1178.891	-0.350	100.000	94.548	1.000

3. 出力用データの関数作成4/4(有無)

J2~J4セルに左の表のような文字列および関数を入力します。

C3セルはFindPatMaxRedLine関数で、GetNFound関数は発見したパターン数を返します。これにより、視野にワークがある場合J4セルに0が、ない場合は1が表示され、ワーク無しのNGを出力できます。

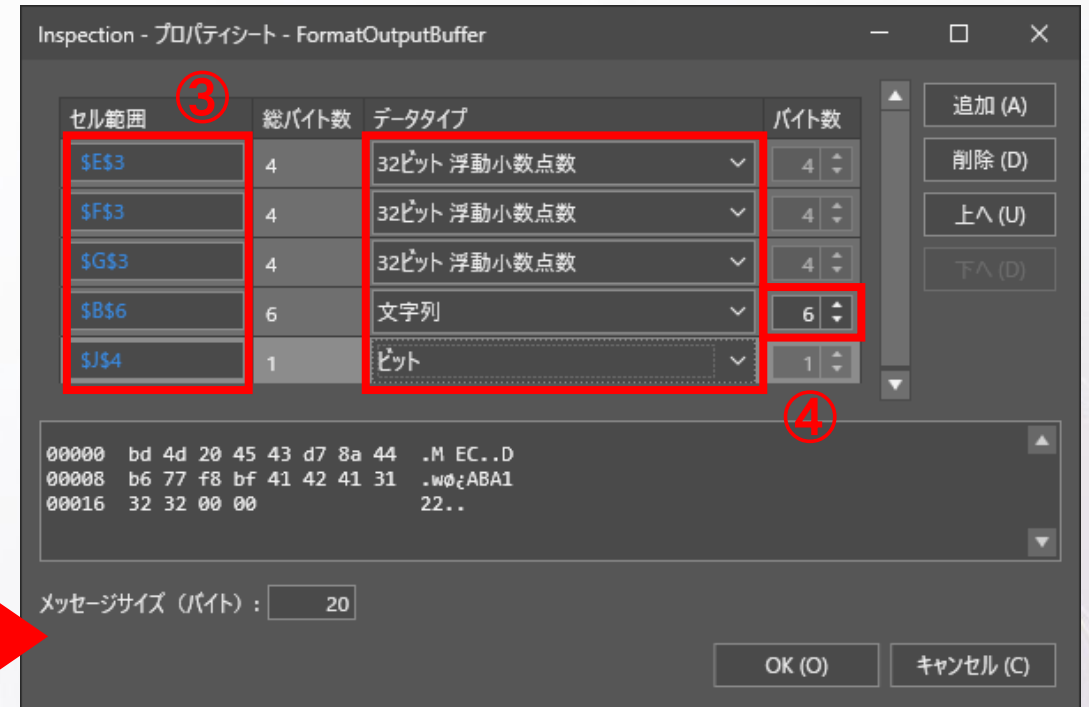
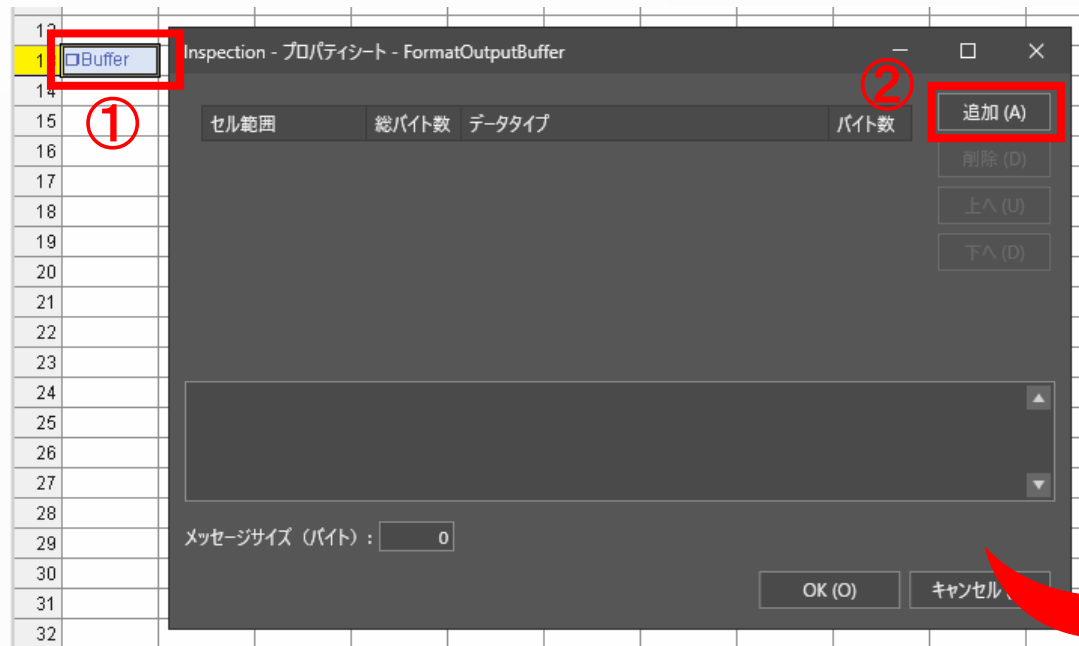
J2	'有無
J3	GetNFound(C3)
J4	If(J3 = 0, 1, 0)

	J
2	有無
3	1.000
4	0.000

4. データ出力関数1/3

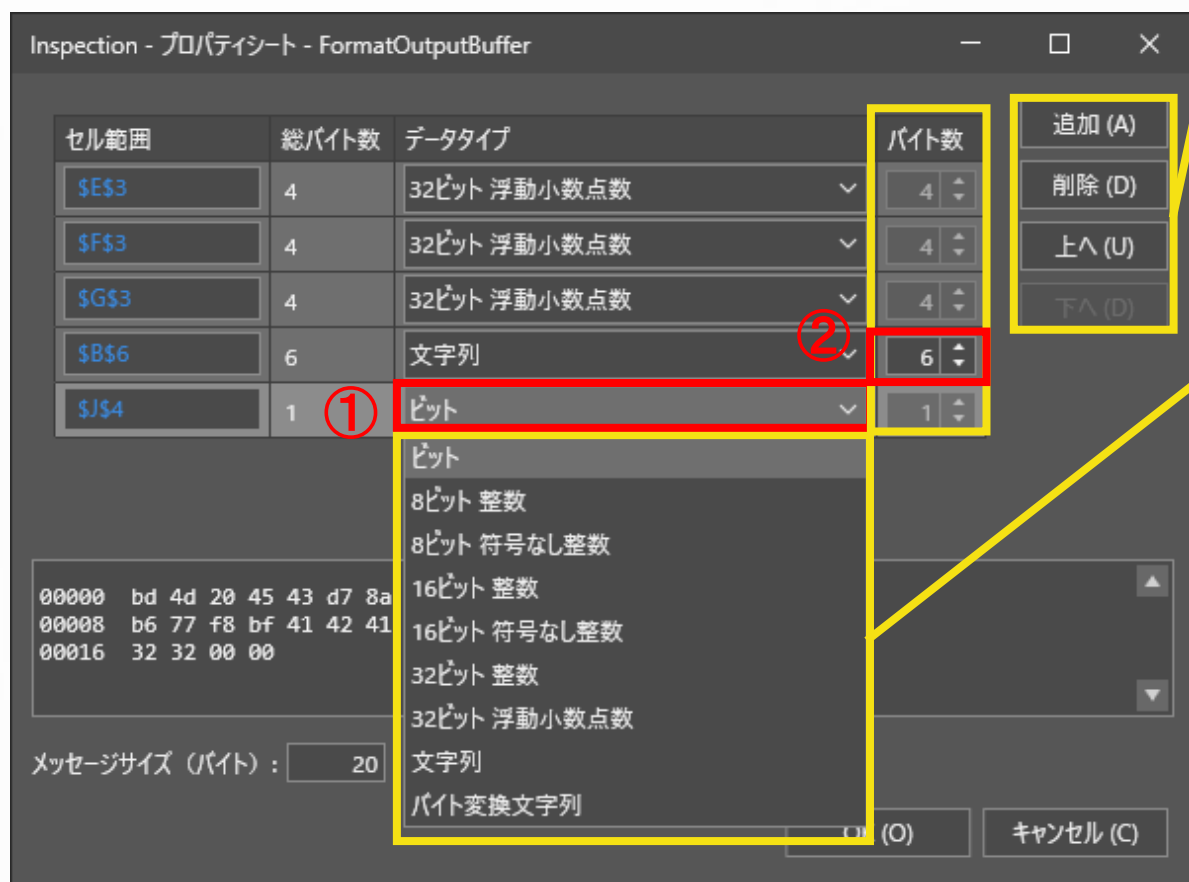
得られた出力データ1~3を実際にPLCへ出力する関数を作ります。

①A13セルにFormatOutputBuffer関数を挿入し、②追加ボタンを押して出力したいデータのセルを参照します。②を繰り返すことで複数のデータを出力することが可能です。③ここではE3(X座標), F3(Y座標), G3(角度), B6(文字列), J4(ワーク無)を選択します。④また、本資料ではデータタイプおよびバイト数を画像のように設定します。詳細は次ページにて解説します。



4. データ出力関数2/3

それぞれのデータタイプにおいて出力する値の形式は表の通りです。本資料では、J4セルが0または1の値のため①ビットを選択し、B6セルにて6文字の文字列を出力するため②バイト数を6に設定します。



データの追加、削除、出力順の変更ができます。

データタイプ	バイト数	出力するセルの値の形式
ビット	1	0または1
8ビット整数	1	-128 ~ 127
8ビット符号なし整数	1	0 ~ 255
16ビット整数	2	-32,768 ~ 32,767
16ビット符号なし整数	2	0 ~ 65535
32ビット整数	4	-2,147,483,648 ~ 2,147,483,647
32ビット浮動小数点数	4	単精度実数
文字列	任意	文字列※
バイト変換文字列	任意	文字列※

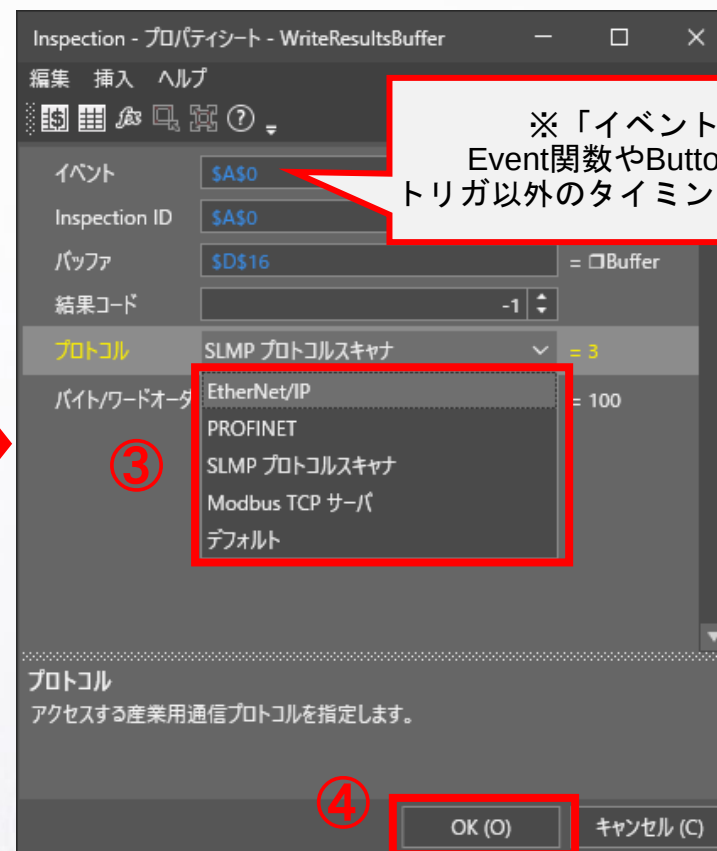
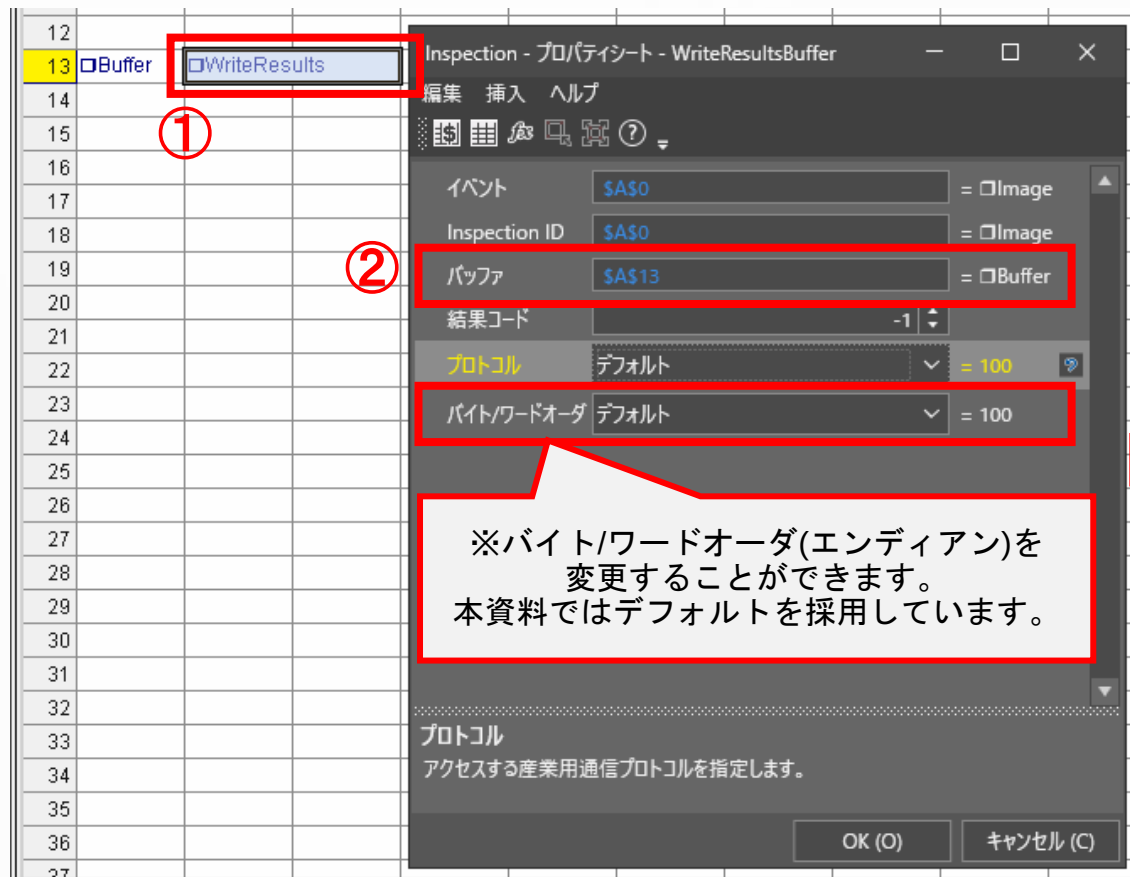
※PLCにはASCIIコードが出力されます。

「文字列」はPLCワード内で15~8ビットに1文字目、7~0ビットに2文字目が出力されます。

「バイト変換文字列」はPLCワード内で7~0ビットに1文字目、15~8ビットに2文字目が出力されます。

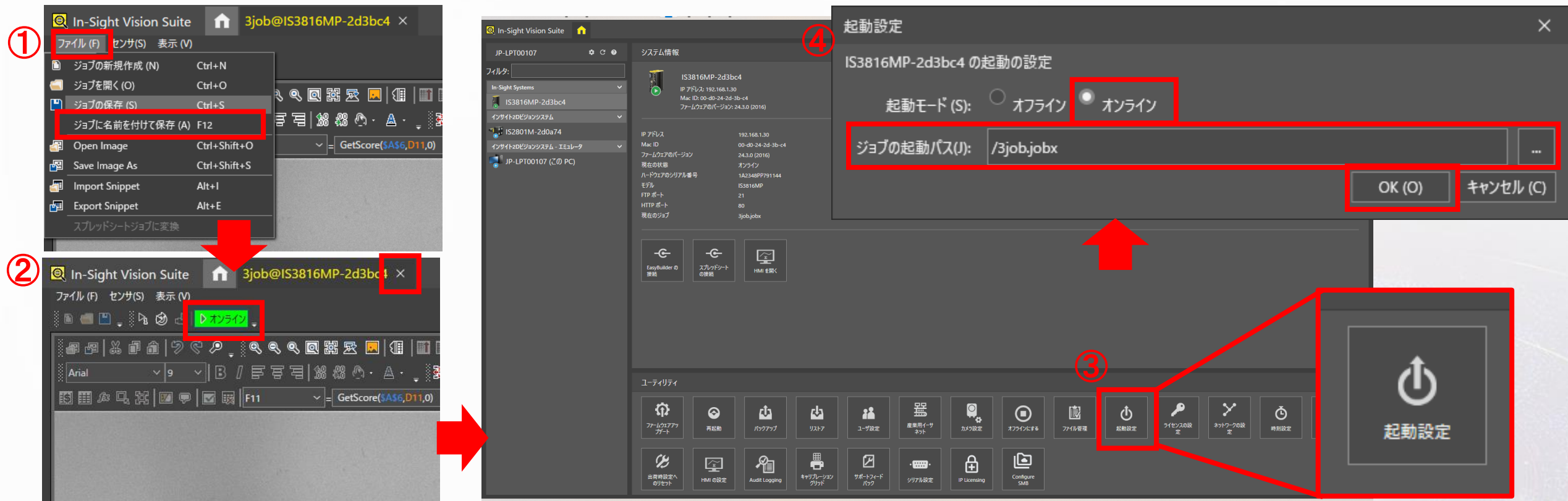
4. データ出力関数3/3

①B13セルにWriteResultsBuffer関数を挿入し、②バッファの項目にて、A13セル (FormatOutputBuffer関数のセル)を参照します。③プロトコルの項目にて、SLMPプロトコル スキャナを選択し④OKをクリックします。



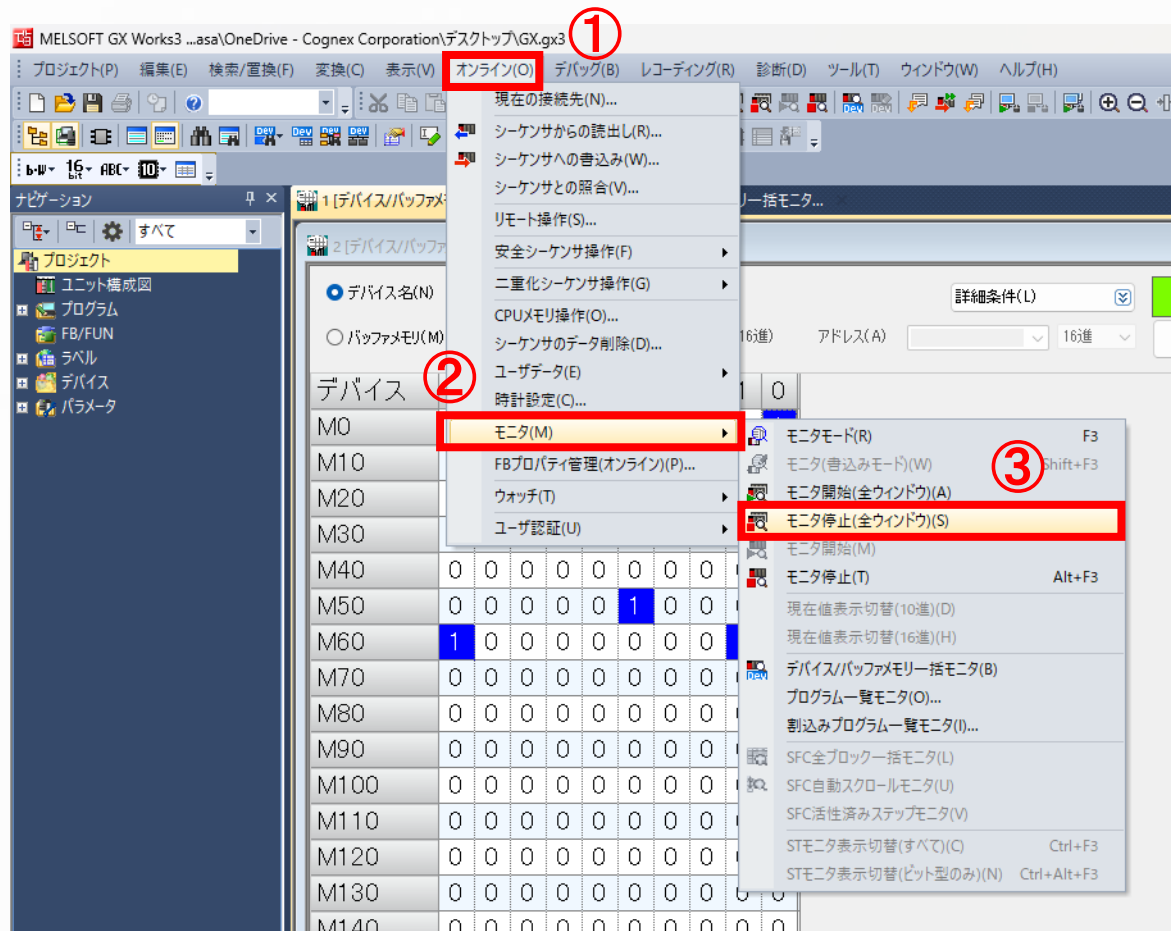
5. ジョブの保存と起動設定

- ①ファイル→ジョブに名前を付けて保存します。(本資料では3job.jobx)
- ②ステータスをオンラインにしてジョブを閉じます。
- ③ホーム画面の起動設定をクリックします。
- ④起動モードをオンライン、ジョブの起動パスを3job.jobxに設定します。



6. データ出力動作確認1/5

GX Works 3にて、①オンライン→②モニター→③モニタ停止(全ウィンドウ)を選択します。



6. データ出力動作確認2/5

次に、①データ表示形式を②単精度実数に変更し、③モニタ開始をクリックします。

① データ表示形式を②単精度実数に変更し、③モニタ開始をクリックします。

16ビット整数[符号付き](1)
16ビット整数[符号なし](2)
32ビット整数[符号付き](3)
32ビット整数[符号なし](4)
64ビット整数[符号付き](6)
64ビット整数[符号なし](7)
② 単精度実数(R)
倍精度実数(L)

デバイス名(N) M0 詳細条件(L) モニタ停止中
○ バッファメモリ(M) ユニット先頭(U) (16進) アドレス(A) 16進 ③ モニタ開始(S)

デバイス	9	8	7	6	5	4	3	2	1	0
M0	0	0	0	0	0	0	0	0	0	1
M10	0	0	0	0	0	0	0	0	0	0
M20	0	0	0	0	0	0	0	0	0	0
M30	0	0	0	0	0	0	0	0	0	0
M40	0	0	0	0	0	0	0	0	0	0
M50	0	0	0	0	0	1	0	0	0	0
M60	1	0	0	0	0	0	0	0	1	0
M70	0	0	0	0	0	0	0	0	0	0

1 [デバイス/バッファメモリ一括モニタ]

デバイス名(N) D10 詳細条件(L) モニタ停止中
○ バッファメモリ(M) ユニット先頭(U) (16進) アドレス(A) 10進 ③ モニタ開始(S)

デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値	文字列
D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D13	0	0	0	0	1	0	0	1	1	0	0	1	1	0	0	0	2456	・
D14	0	0	0	0	1	0	0	1	1	0	0	1	1	0	0	0	2456	・
D15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D16	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	18882	ツI
D17	0	1	0	0	0	1	0	1	0	0	1	0	0	1	0	0	17700	\$E
D18	0	1	0	0	0	1	1	1	0	1	0	1	0	1	0	1	18261	UG
D19	0	1	0	0	0	1	0	0	0	0	0	0	1	1	0	0	17420	D

6. データ出力動作確認3/5



この状態でワークを視野に入れ、トリガを立てると位置座標、文字列のデータが正しく出力されています。

	A	B	C	D	E	F	G	H	I	J
0	Image									
1										
2				番号	X	Y	角度	スケール	得点	有無
3	Patterns	1.000	Patterns	0.000	2702.309	843.763	-0.120	100.000	88.89	1.000
4										0.000
5		文字列	合格	番号	文字	得点	第2候補文	第2候補得点		
6	VIDIELRe	ABC123	1.000	0.000	A	84.296	6	0.000		
7				1.000	B	95.636	D	0.663		
8				2.000	C	94.955	G	1.953		
9				3.000	1	95.719	M	0.000		
10				4.000	2	99.567	-	0.000		
11				5.000	3	86.580	B	0.000		
12										
13	Buffer	WriteResults								
14										

デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値	文字列
D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..
D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5.007418e-033	..
D13	0	0	0	0	1	0	0	1	1	1	1	1	0	0	0	0		..
D14	0	0	0	0	1	0	0	1	1	1	1	1	1	0	0	0	3.520062e-042	..
D15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D16	1	1	1	0	0	1	0	0	1	1	1	1	0	0	1	1	2.702309e+003	!
D17	0	1	0	0	0	1	0	1	0	0	1	0	1	0	0	0		(E
D18	1	1	1	1	0	0	0	0	1	1	0	1	1	1	0	1	8.437635e+002	シ・
D19	0	1	0	0	0	1	0	0	0	1	0	1	0	0	1	0		RD
D20	1	1	1	0	0	0	1	1	1	1	0	0	1	0	1	1	-1.200634e-001	ヒ・
D21	1	0	1	1	1	1	0	1	1	1	1	1	0	1	0	1		□
D22	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	2.841390e-009	AB
D23	0	0	1	1	0	0	0	1	0	1	0	0	0	0	1	1		C1
D24	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	1.836542e-041	23
D25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..

6. データ出力動作確認4/5



位置座標、文字列が正しく出力されています。

	A	B	C	D	E	F	G	H	I	J
0	Image									
1										
2				番号	X	Y	角度	スケール	得点	有無
3	Patterns	1.000	Patterns	0.000	2708.453	518.900	2.838	100.000	55.49	1.000
4										0.000
5		文字列	合格	番号	文字	得点	第2候補文	第2候補得点		
6	VIDIELRe	ABA122	1.000	0.000	A	92.311	W		0.001	
7				1.000	B	98.986	D		0.067	
8				2.000	A	96.313	6		0.000	
9				3.000	1	94.247	Q		0.001	
10				4.000	2	96.826	P		0.000	
11				5.000	2	99.311			0.000	
12										
13	Buffer	WriteResults								
14										

デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値	文字列
D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..
D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4.237046e-033	..
D13	0	0	0	0	1	0	0	1	1	0	1	0	0	0	0	0		..
D14	0	0	0	0	1	0	0	1	1	0	1	0	0	0	0	0	3.475220e-042	..
D15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D16	0	1	0	0	0	1	1	1	0	0	1	1	1	1	1	0	2.708453e+003	>G
D17	0	1	0	0	0	1	0	1	0	0	1	0	1	0	0	1		DE
D18	1	0	1	1	1	0	0	1	1	0	0	1	1	0	0	1	5.189000e+002	匯
D19	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1		D
D20	1	0	1	0	0	0	1	0	1	1	1	0	1	0	0	1	2.838068e+000	鬚
D21	0	1	0	0	0	0	0	0	0	0	1	1	0	1	0	1		5@
D22	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	2.812286e-009	AB
D23	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	1		A1
D24	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	1.800669e-041	22
D25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..

6. データ出力動作確認5/5

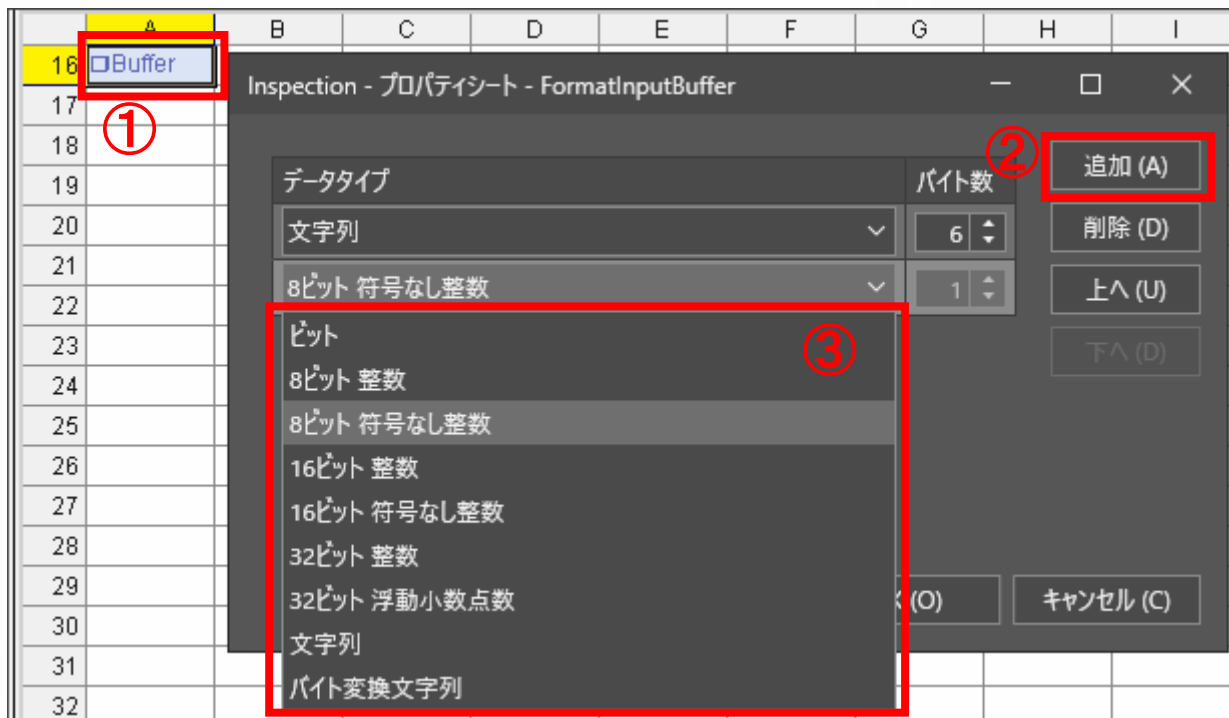


ワークがないことがD25に示されています。

	A	B	C	D	E	F	G	H	I	J
0	Image									
1										
2				番号	X	Y	角度	スケール	得点	有無
3	Patterns	1.000	Patterns	0.000	#ERR	#ERR	#ERR	#ERR	0.00	0.000
4										1.000
5	文字列	合格	番号	文字	得点	第2候補文	第2候補得点			
6	#ERR	#ERR	#ERR	0.000	#ERR	0.000	#ERR	0.000		
7				1.000	#ERR	0.000	#ERR	0.000		
8				2.000	#ERR	0.000	#ERR	0.000		
9				3.000	#ERR	0.000	#ERR	0.000		
10				4.000	#ERR	0.000	#ERR	0.000		
11				5.000	#ERR	0.000	#ERR	0.000		
12										
13	Buffer	WriteResults								
14										

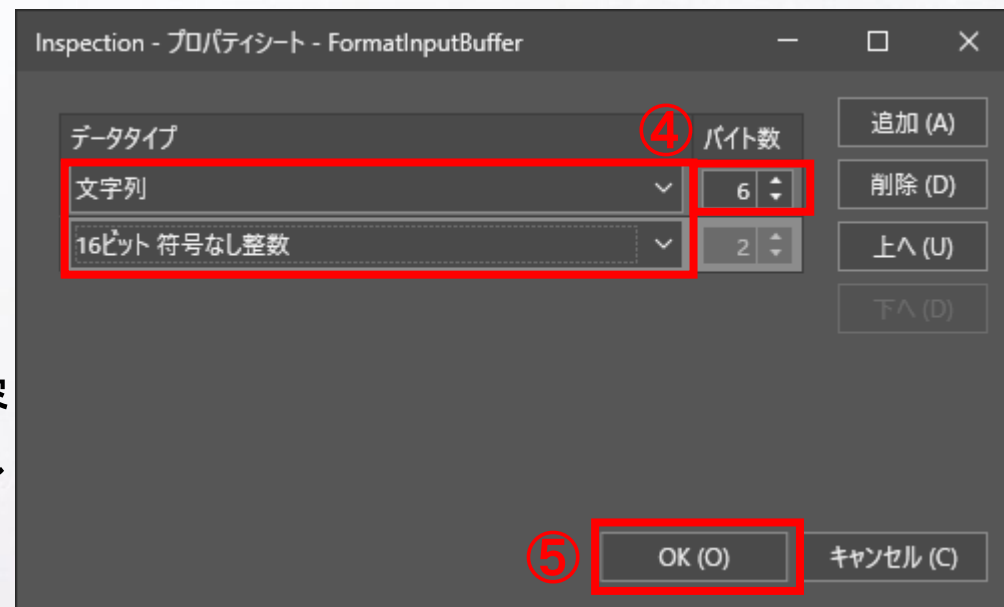
デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値	文字列
D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..
D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5.031492e-033	..
D13	0	0	0	0	1	0	0	1	1	1	0	1	0	0	0	1		Δ.
D14	0	0	0	0	1	0	0	1	1	1	0	1	0	0	0	1	3.521463e-042	Δ.
D15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..
D17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..
D18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..
D19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..
D20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..
D21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..
D23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		..
D24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9.183550e-041	..
D25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1		..
D26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000e+000	..

7. データ入力関数1/4



PLCからビジョンセンサへデータを入力する関数を作成します。

①A16セルにFormatInputBuffer関数を挿入し、②追加ボタンを押し、③入力したいデータタイプを選択します。②③を繰り返すことで複数のデータを入力することが可能です。

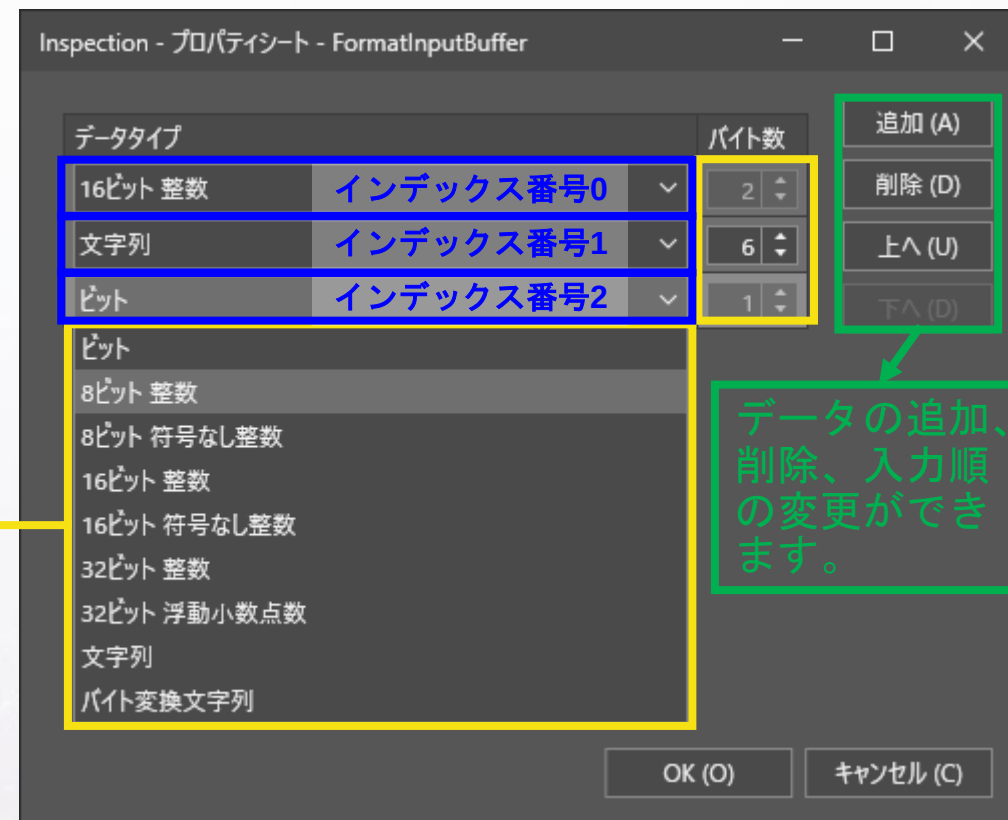


本資料では「ワーク文字列(6文字)」「中心からの許容ワーク位置(pixel)」の2種類のデータをPLCから入力します。このため、データタイプおよびバイト数を④のように設定します。⑤最後にOKを押します。

7. データ入力関数2/4

それぞれのデータタイプにおいてビジョンシステムに入力される値の形式は表の通りです。
各入力データには0からはじまる**インデックス番号**が上から順に割り振られます。

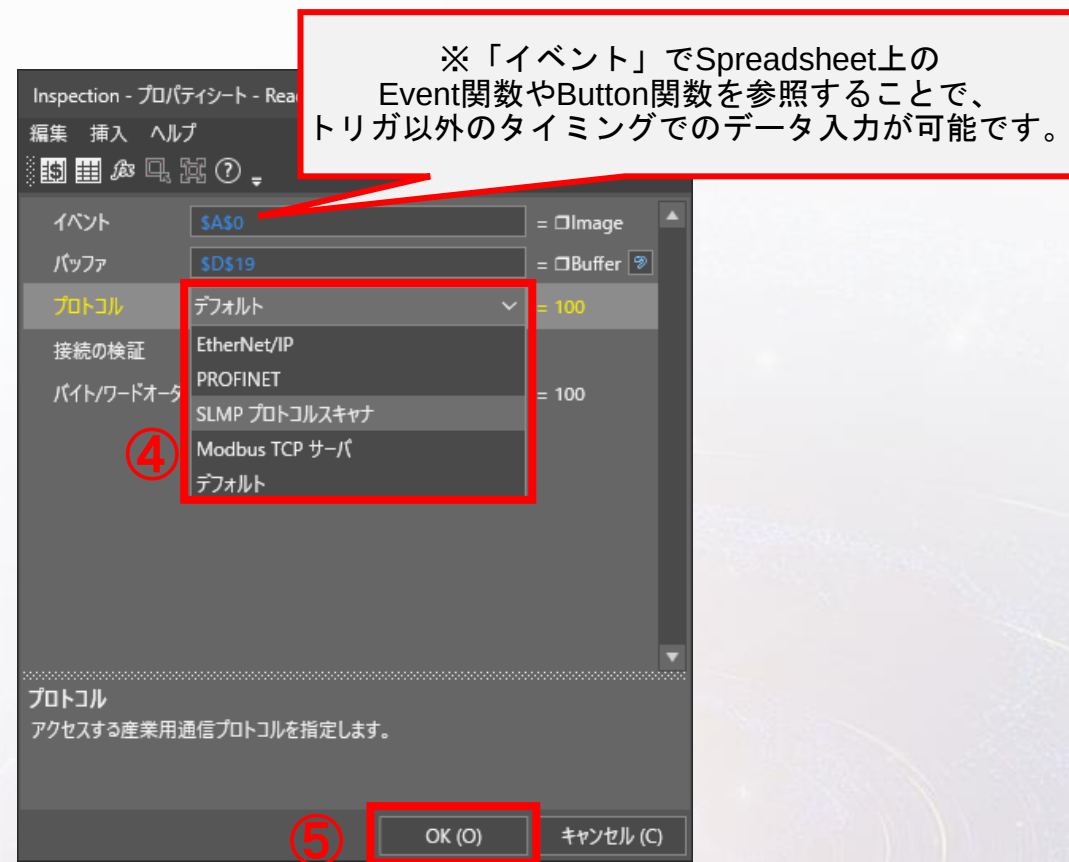
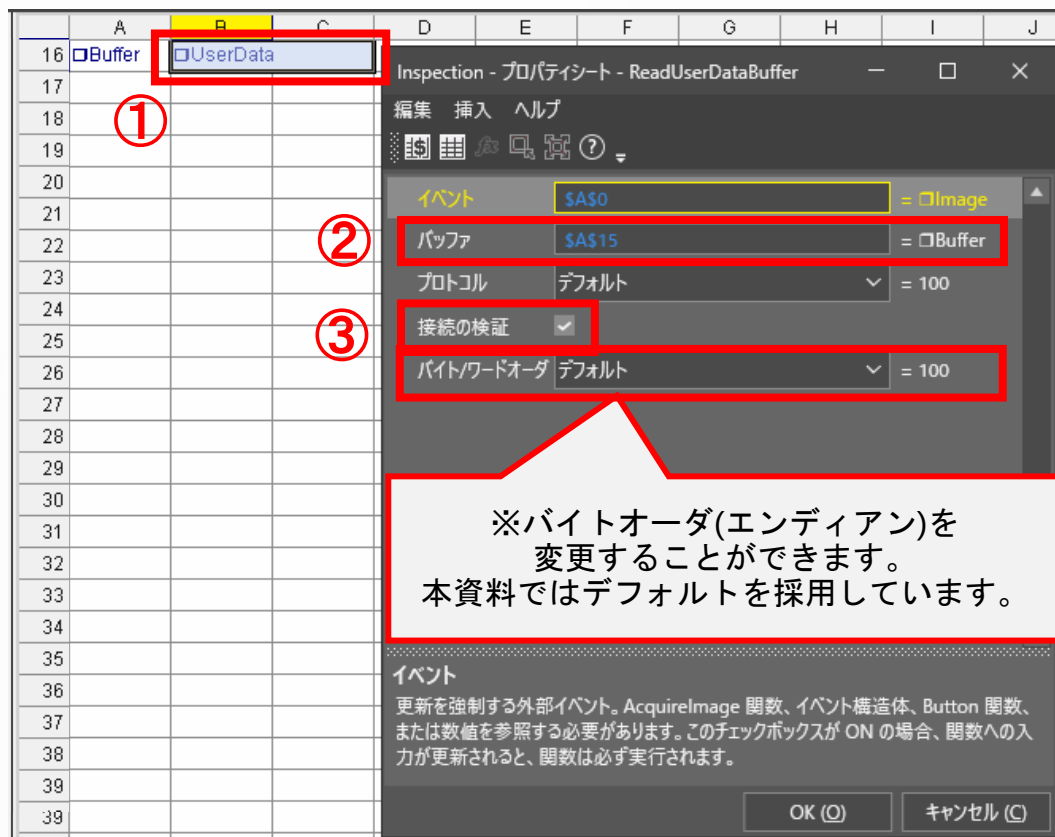
データタイプ	バイト数	セルに入力される値の形式
ビット	1	0/1
8ビット整数	1	-128 ~ 127
8ビット符号なし整数	1	0~255
16ビット整数	2	-32,768 ~ 32,767
16ビット符号なし整数	2	0 ~ 65,535
32ビット整数	4	-2,147,483,648 ~ 2,147,483,647
32ビット浮動小数点数	4	単精度実数
文字列	任意	文字列※
バイト変換文字列	任意	文字列※



※セルにはASCII変換された文字列が入力されます。
「文字列」はPLCワード内で15~8ビットに1文字目、
7~0ビットに2文字目が割り当てられて入力されます。
「バイト変換文字列」はPLCワード内で7~0ビットに
1文字目、15~8ビットが割り当てられて入力されます。

7. データ入力関数3/4

①B16のセルにReadUserDataBuffer関数を挿入します。②バッファの項目にてA15(FormatInputBuffer関数のセル)を参照し、③接続の検証にチェックを入れます。④プロトコルの項目にて、SLMPプロトコルスキャナを選択し、⑤OKをクリックします。



7. データ入力関数4/4

最後に、C15~D16のセルに表のように文字列および関数を挿入します。

C15	D15
'文字列	'許容位置
C16	D16
GetBufferData(\$B\$16, 0)	GetBufferData(\$B\$16, 1)

ReadUserBuffer関数のセル
インデックス番号

	A	B	C	D
15			文字列	許容位置
16	□Buffer	□UserData		0.000

これにより、C16セルに文字列、D16セルに許容位置(0~65535ピクセル)を入力することが可能となります。

8. 入力されたデータの利用1/2

①F15~16のセルに右の表のように文字列および関数を挿入します。次に結果を出力するため、②A13セルのFormatOutputBuffer関数を開き、③追加を押して④F16セルを参照し、データタイプでビットを選択します。⑤最後にOKを押します。

F15	'文字列一致
F16	Exact(B6, C16)

	A	B	C	D	E	F	G	H
1								
2				番号	X	Y	角度	スケール
3	□Patterns	1.000	□Patterns	0.000	2462.561	1549.276	-0.111	100
4								
5		文字列	合格	番号	文字	得点	第2候補文字	第2候補
6	□ViDiELRe:ABC123		1.000	0.000	A	96.311	B	0
7				1.000	B	99.643	P	0
8				2.000	C	96.332	S	0
9				3.000	1	94.248	3	0
10				4.000	2	97.985	-	0
11				5.000	3	92.314	5	0
12								
13	□Buffer	□WriteResults						
14								
15			文字列	許容位置				
16	□Buffer	□UserData		0.000				
17								



Inspection - プロパティシート - FormatOutputBuffer

セル範囲	総バイト数	データタイプ	バイト数
\$E\$3	4	32ビット 浮動小数点数	4
\$F\$3	4	32ビット 浮動小数点数	4
\$G\$3	4	32ビット 浮動小数点数	4
\$B\$6	6	文字列	6
\$J\$4	1	ビット	1
\$F\$16	1	ビット	1

00000 f8 e8 19 45 d6 a8 c1 44 eè.EÖ.ÁD
 00008 d9 8e e2 bd 41 42 43 31 ù.â.ABC1
 00016 32 33 00 00 23..

メッセージサイズ (バイト) : 20

追加 (A) 削除 (D) 上へ (U) 下へ (D)

OK (O) キャンセル (C)

8. 入力されたデータの利用2/2

- ①G15~H16のセルに右の表のように文字列および関数を挿入します。
- 次に②A13セルのFormatOutputBuffer関数を開き、③追加を押して
- ④G15セル, H15セルを参照し、データタイプでビットを選択します。
- ⑤最後にOKを押した後、再度ジョブを保存しておきます。

G15	H15
'X位置OK	'Y位置OK
G16	H16
InRange(E3, 2660-D16, 2660+D16)※	InRange(F3, 1516-D16, 1516+D16)※

※2660, 1516はIS3816の中心座標を表しています。
お使いの機種に合わせて設定してください。

	A	B	C	D	E	F	G	H	I
0	Image								
1									
2				番号	X	Y	角度	スケール	得点
3	Patterns	1.000	Patterns	0.000	2462.654	1548.434	-0.101	100.000	88.11
4									
5		文字列	合格	番号	文字	得点	第2候補文字	第2候補得点	
6	VIDIELRe	ABC123	1.000	0.000	A	97.269	8	0.000	
7				1.000	B	99.574	P	0.049	
8				2.000	C	97.289	G	0.362	
9				3.000	1	94.248	3	0.000	
10				4.000	2	98.270	-	0.000	
11				5.000	3	94.249	5	0.000	
12									
13	Buffer	WriteResults							
14									
15		文字列	許容位置	文字列一致	X位置OK	Y位置OK			
16	Buffer	UserData	0.000	0.000	0.000	0.000			

	A	B	C	D	E	F	G	H	I
0	Image								
1									
2									
3	Patterns	1.000	Patterns						
4									
5		文字列	合格						
6	VIDIELRe	ABC123							
7									
8									
9									
10									
11									
12									
13	Buffer	WriteResults							
14									
15		文字列	許容位置	文字列一致	X位置OK	Y位置OK			
16	Buffer	UserData	0.000	0.000	0.000	0.000			

Inspection - プロパティシート - FormatOutputBuffer

セル範囲	総バイト数	データタイプ	バイト数
\$G\$3	4	32ビット 浮動小数点数	4
\$B\$6	6	文字列	6
\$J\$4	1	ビット	1
\$F\$16	1	ビット	1
\$G\$16	1	ビット	1
\$H\$16	1	ビット	1

メッセージサイズ (バイト): 20

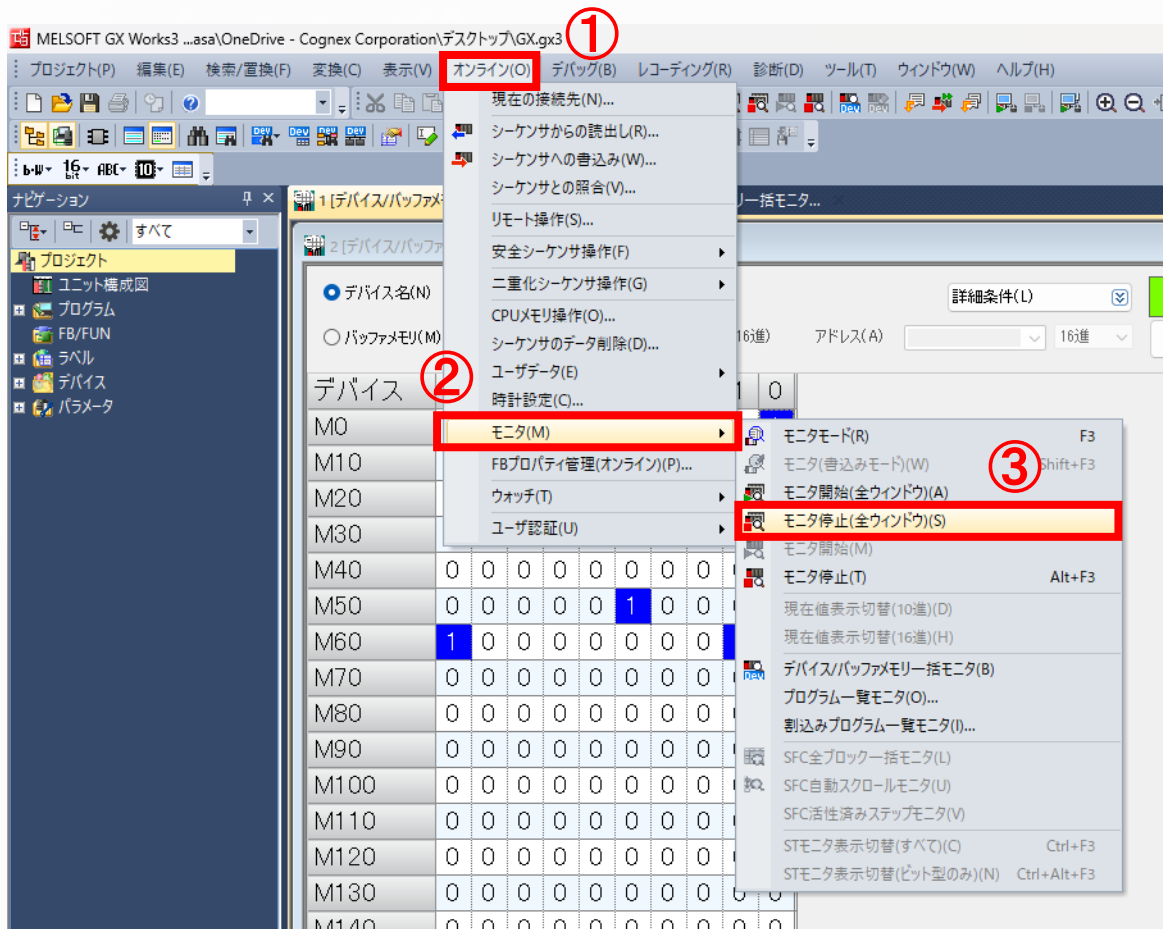
0000 76 ea 19 45 e1 8d c1 44 vÈ.Eä.ÁD
 0008 2d 43 ce bd 41 42 43 31 -CÍ.ABC1
 0016 32 33 00 00 23..

追加 (A) 削除 (D) 上へ (U) 下へ (D)

OK (O) キャンセル (C)

9. データ入力動作確認1/4

GX Works 3にて、①オンライン→②モニタ→③モニタ停止(全ウィンドウ)を選択します。



9. データ入力動作確認2/4

次に、①データ表示形式を②16ビット整数[符号なし]に変更し、③モニタ開始をクリックします。

The screenshot displays the MELSOFT GX Works3 software interface. The left window shows the 'データ表示形式' (Data Display Format) menu with '16ビット整数[符号なし]' (16-bit unsigned integer) selected. The right window shows the '1 [デバイス/バッファメモリ一括モニタ]' (1 [Device/Buffer Memory Batch Monitor]) window. The 'デバイス名(N)' (Device Name) is set to 'D10'. The 'モニタ開始(S)' (Start Monitor) button is highlighted with a red box and a circled '3'. The data table below shows the current values for device D10.

デバイス名	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値	文字列
D10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D13	0	0	0	0	1	0	0	1	1	0	0	1	1	0	0	0	2456	*
D14	0	0	0	0	1	0	0	1	1	0	0	1	1	0	0	0	2456	*
D15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D16	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	18882	7I
D17	0	1	0	0	0	1	0	1	0	0	1	0	0	1	0	0	17700	\$E
D18	0	1	0	0	0	1	1	1	0	1	0	1	0	1	0	1	18261	UG
D19	0	1	0	0	0	1	0	0	0	0	0	1	1	0	0	0	17420	D

9. データ入力動作確認3/4

データ入力をするにはまず、①入力ブロックにデータを書き込みます。今回はD2~D4へ「ABC123」という文字列を、D5へ「200」という数値を書き込むため、写真のようにビットを立てます。

②次にコントロールブロックにて、M16(Set User Data)、③M2(Buffer Results Enable)のビットを順番に立て、④最後にM1(Trigger)のビットを立てると、⑤データが入力されます。

デバイス	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	現在値	文字列
D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..
D2	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	16961	AB
D3	0	0	1	1	0	0	0	1	0	1	0	0	0	0	1	1	12611	C1
D4	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	13106	23
D5	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	200	ネ

デバイス	9	8	7	6	5	4	3	2	1	0
M0	0	0	0	0	0	0	0	1	1	1
M10	0	0	0	1	0	0	0	0	0	0
M20	0	0	0	0	0	0	0	0	0	0

	A	B	C	D
15			文字列	許容位置
16	□Buffer	□UserData		

	A	B	C	D
15			文字列	許容位置
16	□Buffer	□UserData	ABC123	200.000

9. データ入力動作確認4/4

文字列一致の結果、X位置のOK/NG、Y位置のOK/NGの結果が出力ブロックに表示されます。

	F	G	H
15	文字列一致	X位置OK	Y位置OK
16	1.000	1.000	1.000



10. データ入力タイミングチャート

操作は**入力ブロック**および**コントロールブロック**で行います。

User Data
(入力ブロック)

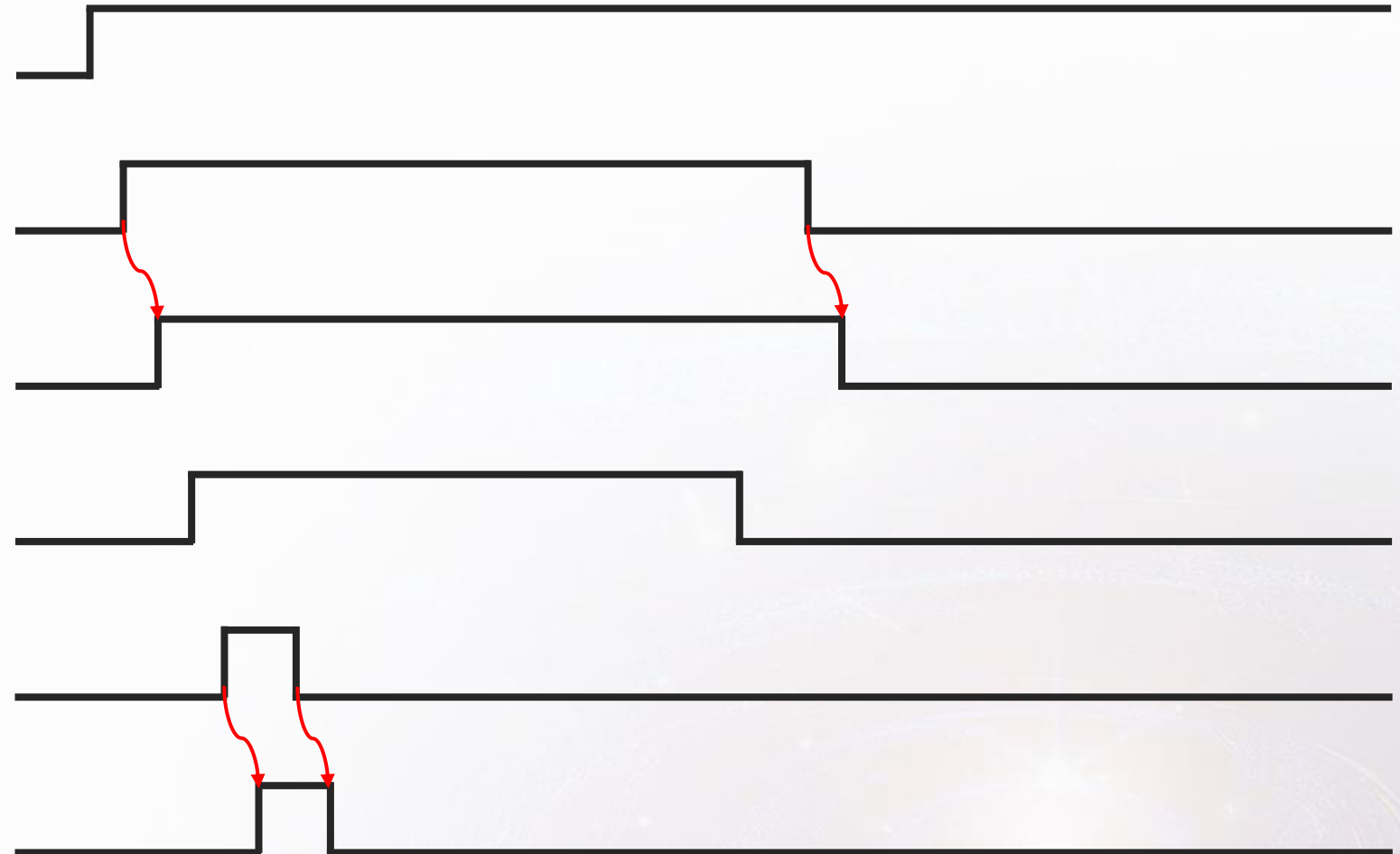
Set User Data(M16)
(コントロールブロック)

Set User Data Ack(M66)
(ステータスブロック)

Buffer Results Enable(M2)
(コントロールブロック)

Trigger(M1)
(コントロールブロック)

Result Valid(M61)
(ステータスブロック)





Thank You !