

## In-Sight D900接続ガイド

### SLMP 通信設定

作成年月日： 2023年10月31日



本書は In-Sight D900を三菱電機社製シーケンサ CPU に接続し、  
SLMP 接続で制御する手順を説明します。

## 目 次

|                                                   |    |
|---------------------------------------------------|----|
| 1. システム構成.....                                    | 2  |
| 2. In-Sight 接続設定.....                             | 3  |
| 2.1 PC に IP アドレスを設定 .....                         | 3  |
| 2.2 In-Sight センサの設定 .....                         | 4  |
| 3. シーケンサの設定 .....                                 | 8  |
| 3.1 シーケンサに接続 .....                                | 8  |
| 3.2 シーケンサの IP アドレス設定 .....                        | 10 |
| 4. コントロールブロックを使った読み取り指令.....                      | 12 |
| 4.1 モニタ設定 .....                                   | 12 |
| 4.2 コントロールブロックを用いたシーケンサによる In-Sight D900の制御 ..... | 13 |
| 4.2.1 トリガ機能.....                                  | 13 |
| 4.2.2 JOB 切り替え機能 .....                            | 14 |
| 4.2.3 シーケンサと In-Sight D900間のデータの送受信 .....         | 16 |
| 4.3 補足情報.....                                     | 23 |
| 4.3.1 画像取り込みエラーの監視 .....                          | 23 |
| 4.3.2 Inspection Completed の監視 .....              | 24 |

## 改訂履歴

|     | 文書番号          | 作成日        |
|-----|---------------|------------|
| 初版  | CKKAE2021-001 | 2021/01/20 |
| 第二版 | CKKAE2023-003 | 2023/03/15 |
| 第三版 | CKKAE2023-010 | 2023/10/31 |

## ●はじめに

産業用プロトコルである SLMP の通信手順をご案内します。

## ●SLMP 通信設定

SLMP は Ethernet またはシリアル接続を介して Q、iQ-R、FX、および L シリーズのシーケンサと通信するために三菱電機が提供している標準化された通信フォーマットです。

シーケンサの Ethernet ハードウェアおよびソフトウェアを使用して、In-Sight D900(以下、D900)の I/O、データ、制御コマンドのやり取りを行います。

D900がシーケンサデバイスを定期的に監視するため、シーケンサのビットデバイス制御のみでトリガ通知等の制御が可能となります。

## 1. システム構成

### 【動作環境】

下記の機材を使用し動作確認を行っています。

[In-Sight D900]

ISD905C ファームウェア 2.7.0(436)

[PC:パソコン]

Legion Y740-15 IR Hg Windows 10 64ビット

[シーケンサ]

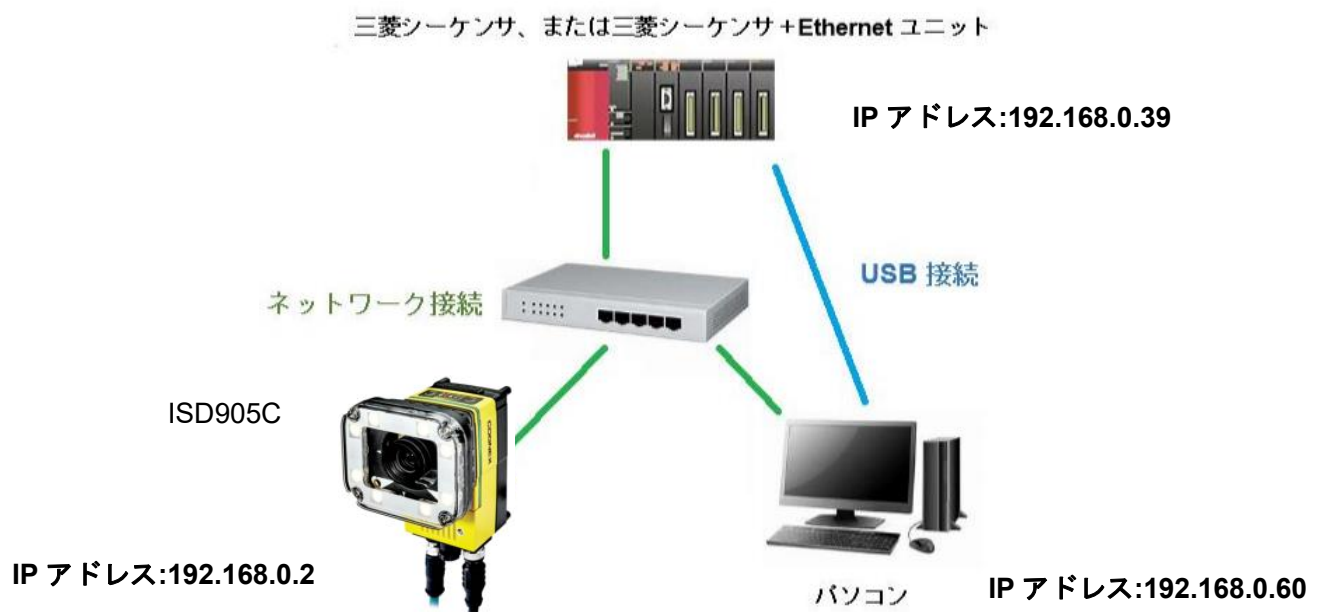
CPU : Q04UDEHCPU

[ソフトウェア]

In-Sight D900 : In-Sight Vision Suite In-Sight ViDi 1.7.0

シーケンサ : GX Works2 Version 1.590Q

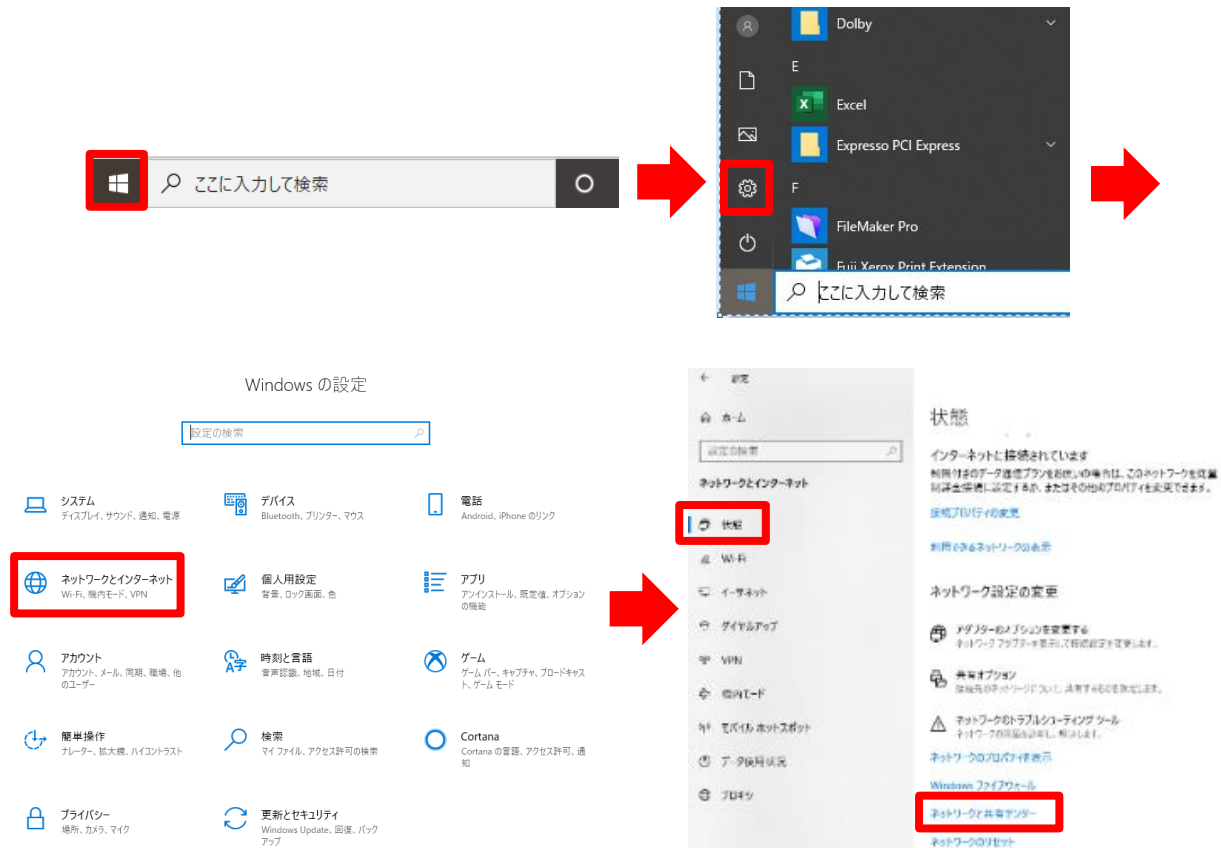
### 【接続構成図】



## 2. In-Sight 接続設定

### 2.1 PC に IP アドレスを設定

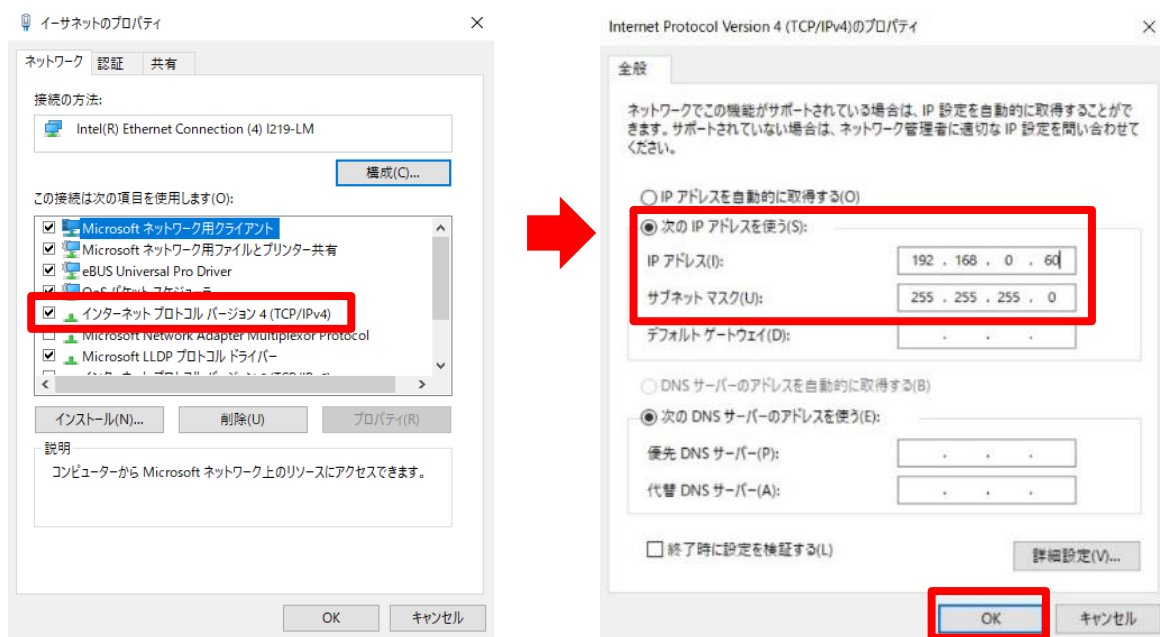
- ① PC の画面左下 Windows[スタート]をクリック > [設定]をクリック > [ネットワークとインターネット]をクリック > [状態]を選択 > [ネットワークと共有センター]をクリック



- ② アクティブなネットワークの表示一覧から[イーサネット]を選択>[プロパティ]をクリック

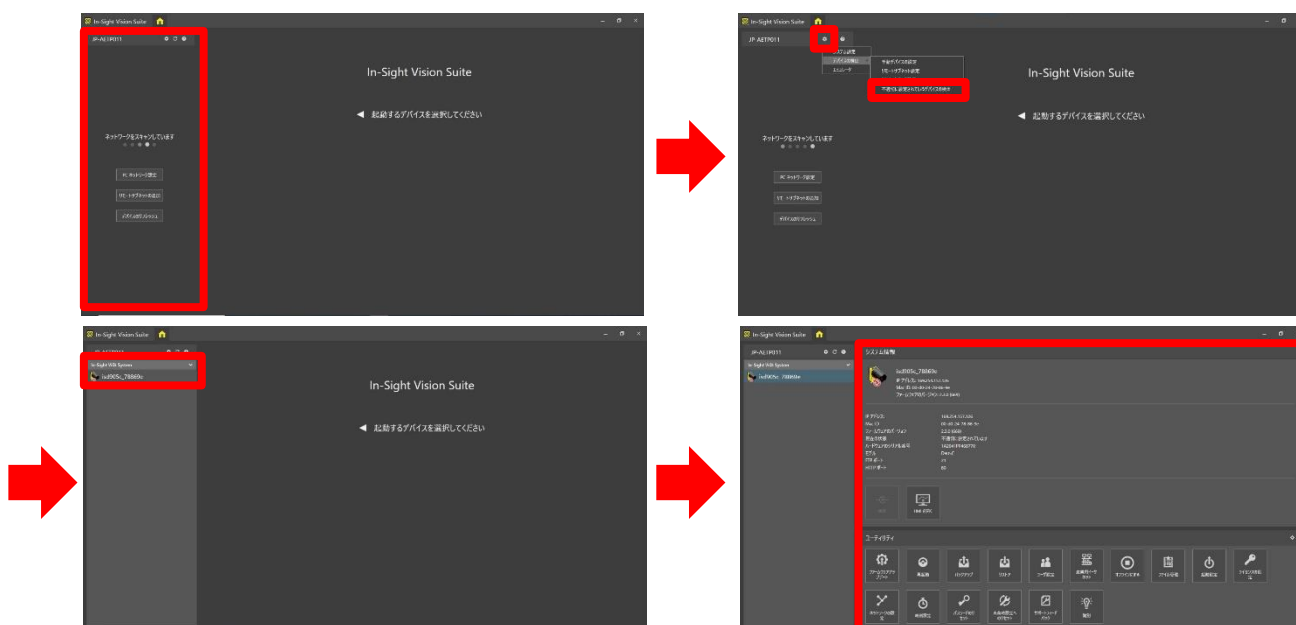


- ③ [インターネットプロトコルバージョン4(TCP/IPv4)]をダブルクリック>[次の IP アドレスを使う]を選択し、IP アドレスとサブネットマスクを設定(この例ではそれぞれ 192.168.0.60 と 255.255.255.0 を設定)>[OK]ボタンをクリック



## 2.2 In-Sight センサの設定

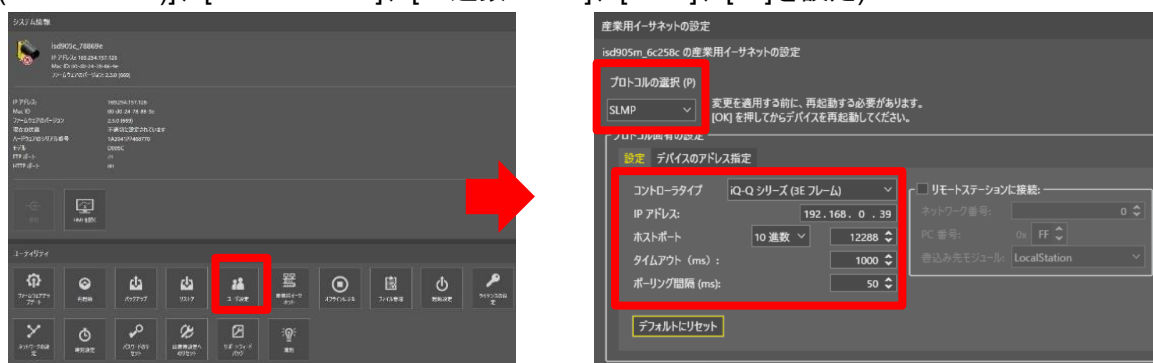
- ① In-Sight Vision Suite(以下、ISVS)を起動します。
- ② ネットワークスキャン後に画面左手の一覧に D900が追加されます。  
表示されない場合は[システム設定] > [デバイスの検出] > [不適切に設定されているデバイスの検出]をクリック > 一覧に D900が表示 > D900アイコンをクリックし、デバイスの各種設定を行う。



- ③ ユーティリティ内[ネットワークの設定]をクリック> ネットワーク設定画面内[アドレス指定]で固定を選択> IS の IP アドレスおよびサブネットマスクを設定(この例では[IP アドレス]は192.168.0.2、[サブネットマスク]は255.255.255.0)> [OK]ボタンをクリック> IS でネットワーク設定の処理が行われる間に待機メッセージが表示され、その後設定が反映されます。



- ④ ユーティリティ内[産業用イーサネット]をクリック> 産業用イーサネットの設定画面内[プロトコルの選択]で SLMP を設定> [設定]タブを選択> 以下各種設定を行う。(この例では[コントローラタイプ]、[IP アドレス]、[ポート]、[タイムアウト]、[ポーリング間隔]にそれぞれ[IQ-Q シリーズ(3E フレーム)]、[192.168.0.39]、[10進数12288]、[1000]、[50]を設定)



- ⑤ [デバイスのアドレス設定]タブを選択> 各制御ブロックの設定を行う。(この例では以下のように設定)> [再起動]ボタンをクリック。ここでシーケンサとの通信におけるデータの使用領域を設定。



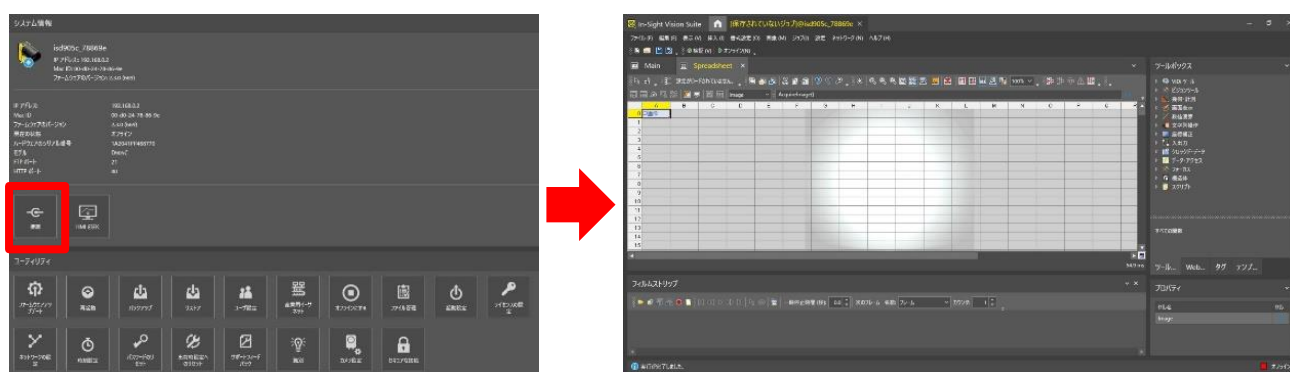
**コントロールブロック**：D900へ制御指令(トリガなど)を行う部分です。

**ステータスブロック**：D900の状態(オンライン・一時停止状態など)を表す部分です。

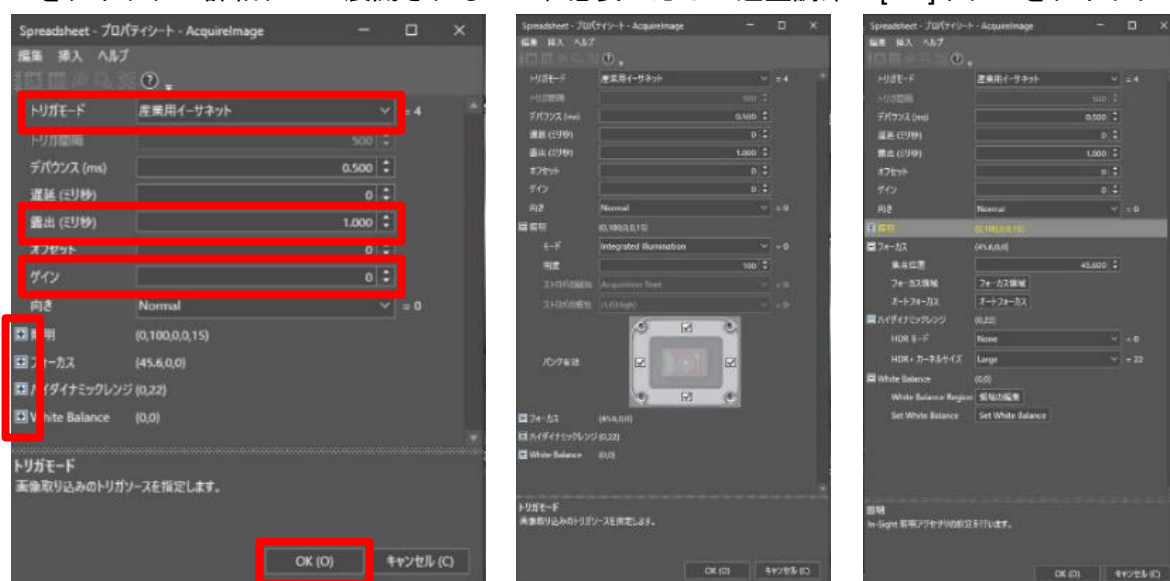
**入力ブロック**：シーケンサから D900へパラメータなどを入力する部分です。JOB 切り替えを行う時に JOBID を指定する場合はこちらで ID を設定します。

**出力ブロック**：D900からシーケンサへ測定結果などを出力する部分です。

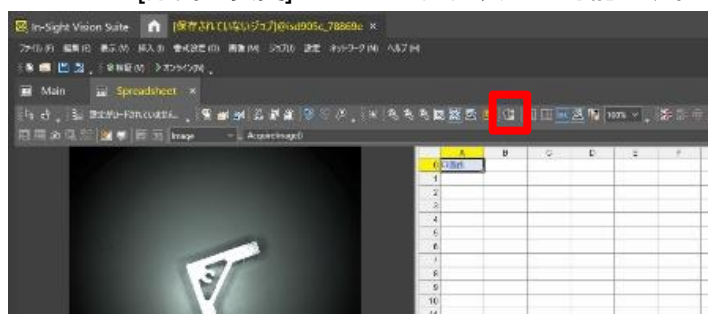
## ⑥ 画面中央[接続]ボタンをクリック > D900への接続が完了



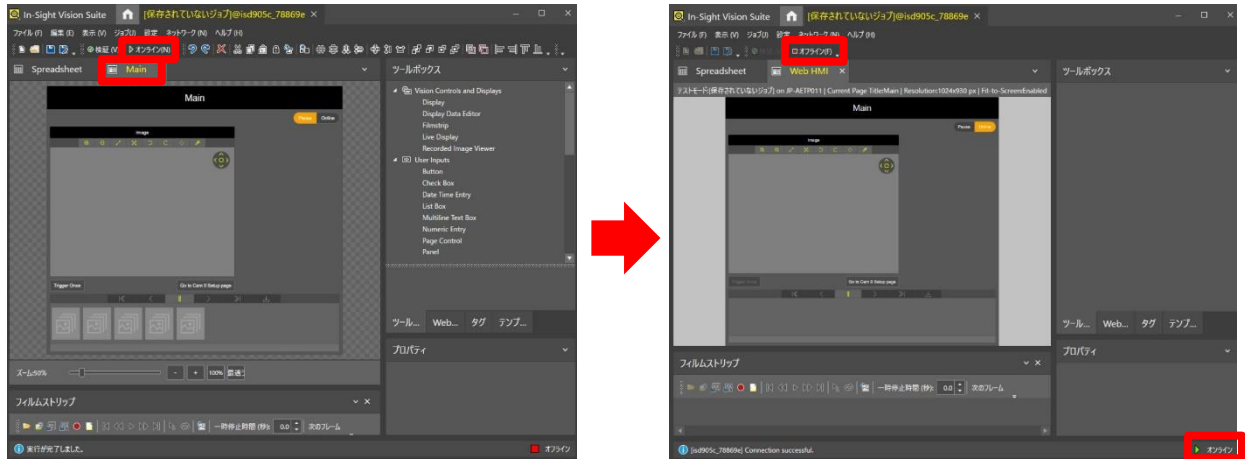
## ⑦ スプレッドシートのセル A0をダブルクリック > AcquireImage 関数で取得画像の設定を行う。(この例では[トリガモード]、[露出]、[ゲイン]にそれぞれ[産業用イーサネット]、[1]、[0]を設定) > +アイコンをクリックで詳細タブが展開されるので、必要に応じて適宜調節 > [OK]ボタンをクリック



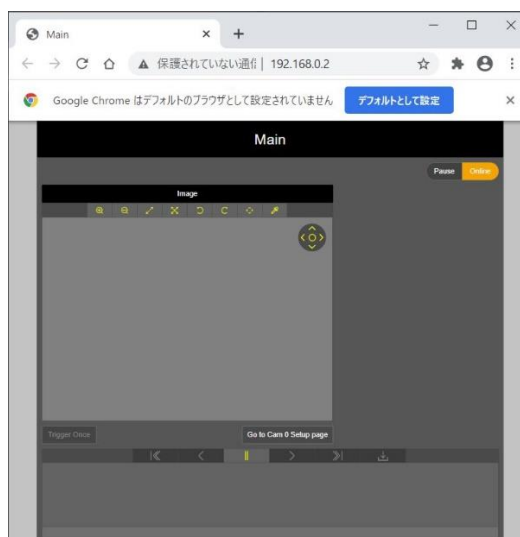
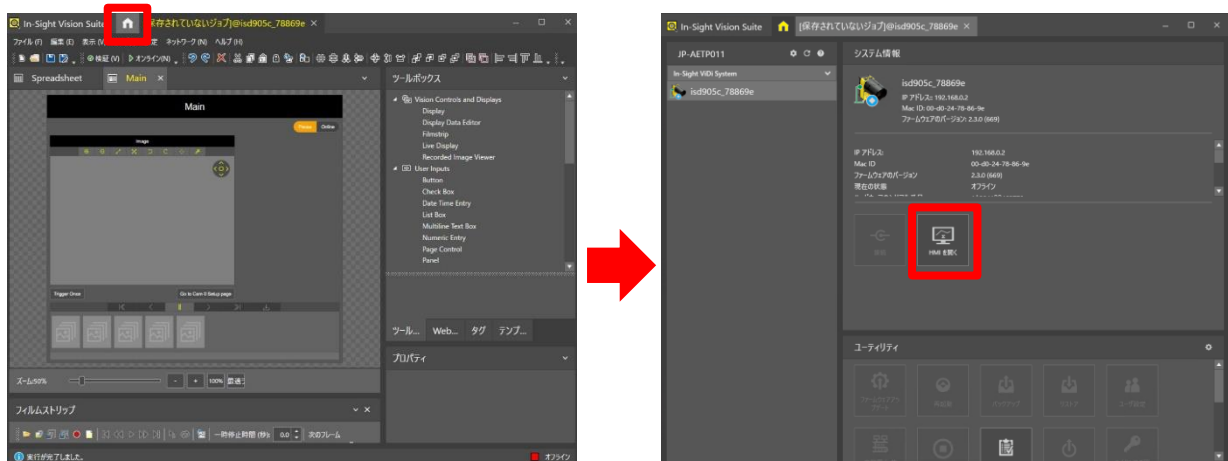
※Spreadsheet のツールバー内[分割の変更]ボタンで2画面表示も可能です。



- ⑧ [Main]タブを選択 > この画面でオンライン時の画面設定が行えます。  
 ツールバー内[オンライン]ボタンをクリック > D900がオンライン状態となる。この時シーケンサ等の他のデバイスと通信可能。 > 再度 D900の設定を行う際はツールバー内[オフライン]ボタンをクリック



- ⑨ オンライン状態時に[In-Sight Vision Suite]をクリック > ホーム画面に戻り、[HMI を開く]ボタンをクリック > WebHMIが開きます。この画面は同じネットワーク内の他のモニタから閲覧できます。



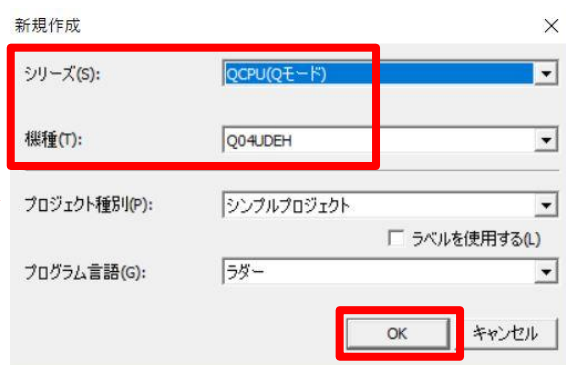
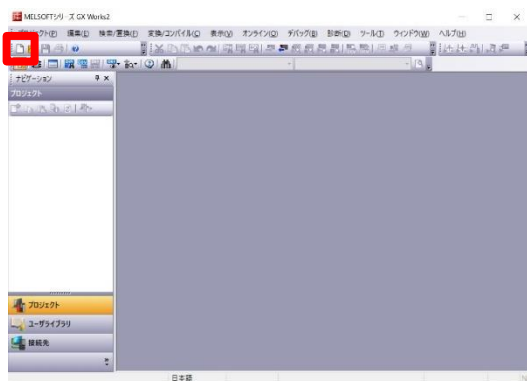
## 3. シーケンサの設定

### 3.1 シーケンサに接続

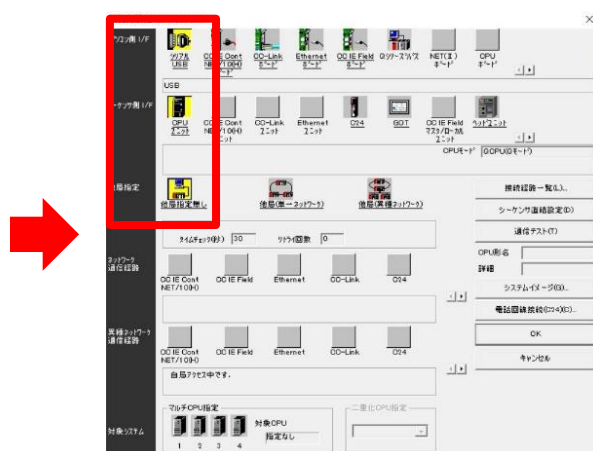
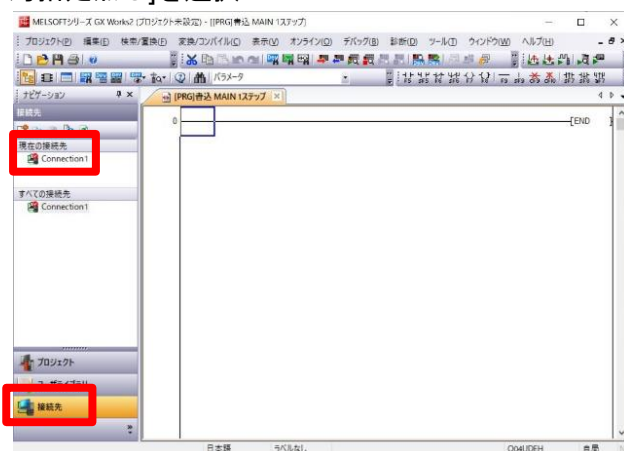
- ① [Windows スタートボタン] > [MELSOFT] 内 [GX Works2]をクリック  
またはデスクトップ上のショートカットをダブルクリック



- ② [新規作成]ボタンをクリック > 今回の例では[シリーズ]と[機種]にそれぞれ[QCPU]と[Q04UDEH]を選択し、[OK]ボタンをクリック



- ③ 画面左下[接続先]タブをクリック > 接続先一覧から[Connection1]をダブルクリック > 今回の例では[パソコン側 I/F]、[シーケンサ側 I/F]、[他局指定]にそれぞれ[シリアル USB]、[CPU ユニット]、[他局指定無し]を選択

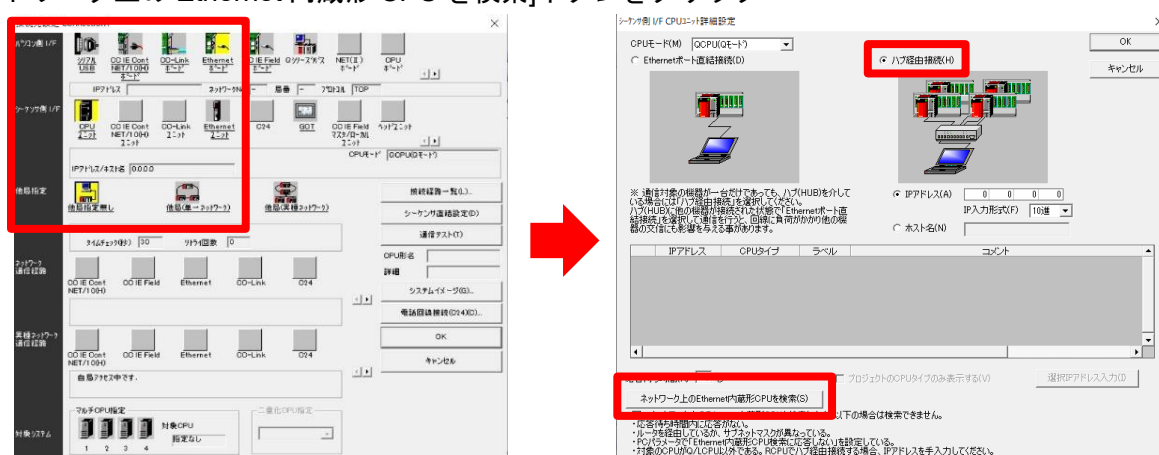


- ④ [通信テスト]ボタンをクリック > エラーが出なければ接続成功(接続できない場合は使用するシーケンサの機種を確認して再度設定して下さい) > [OK]ボタンをクリック > [OK]ボタンをクリック

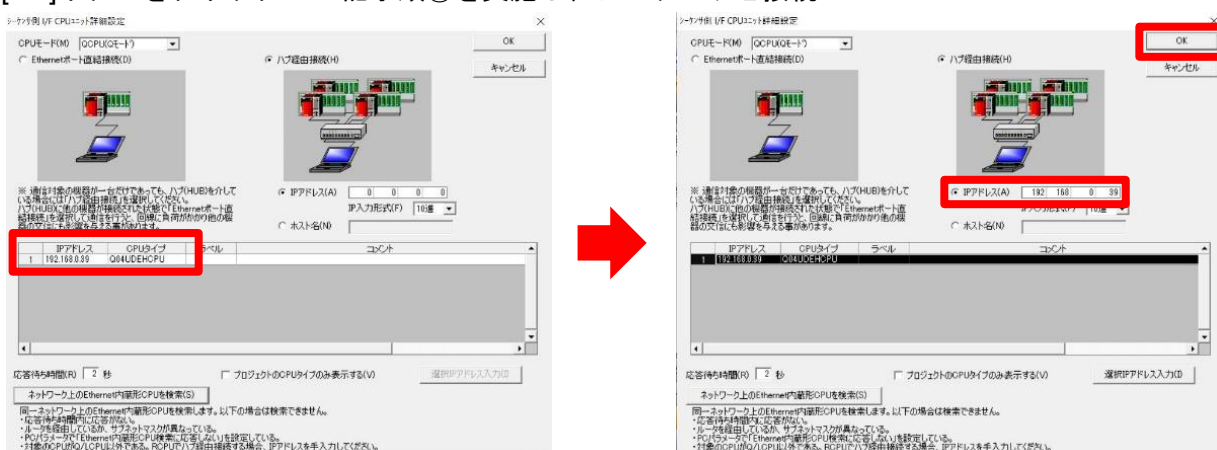


※以下、ハブ経由で Ethernet ポートを使用して接続する場合の手順

- ⑤ [パソコン側 I/F]、[シーケンサ側 I/F]、[他局指定]にそれぞれ[Ethernetポート]、[CPU ユニット]、[他局指定無し]を選択し、[CPU ユニット]アイコンをダブルクリック > [ハブ経由接続]を選択 > [ネットワーク上の Ethernet 内蔵形 CPU を検索]ボタンをクリック

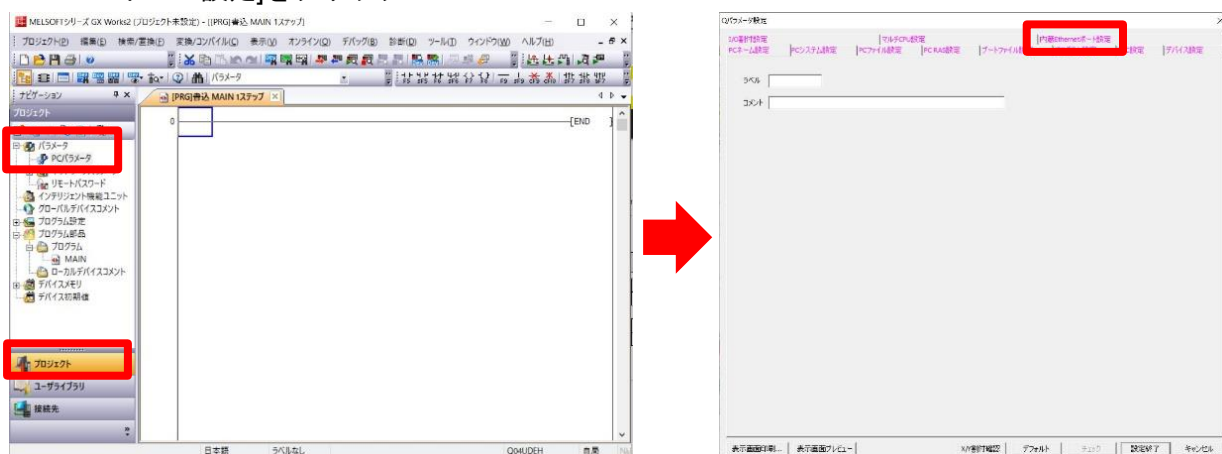


- ⑥ 一覧に接続されているシーケンサが表示されるので任意のものをクリック > IP アドレスが反映 > [OK]ボタンをクリック > 上記手順④を実施し、シーケンサと接続

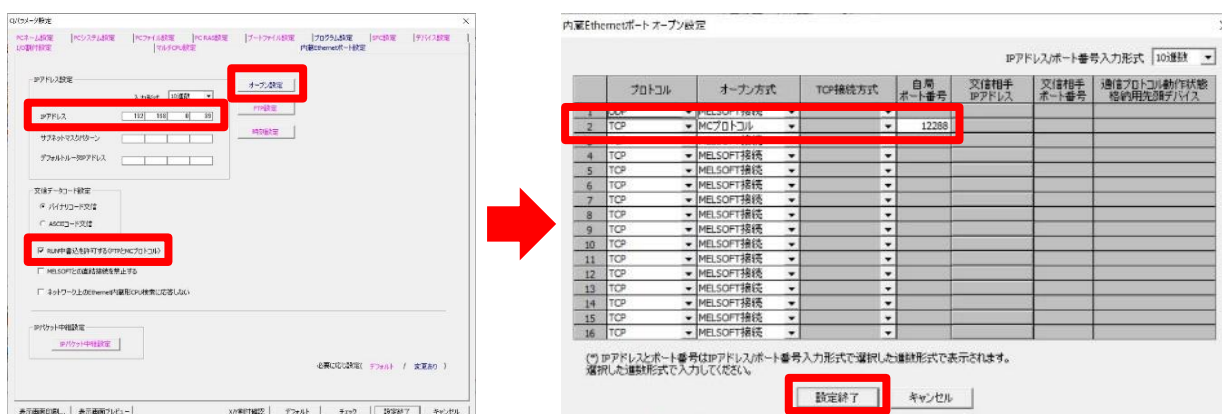


## 3.2 シーケンサの IP アドレス設定

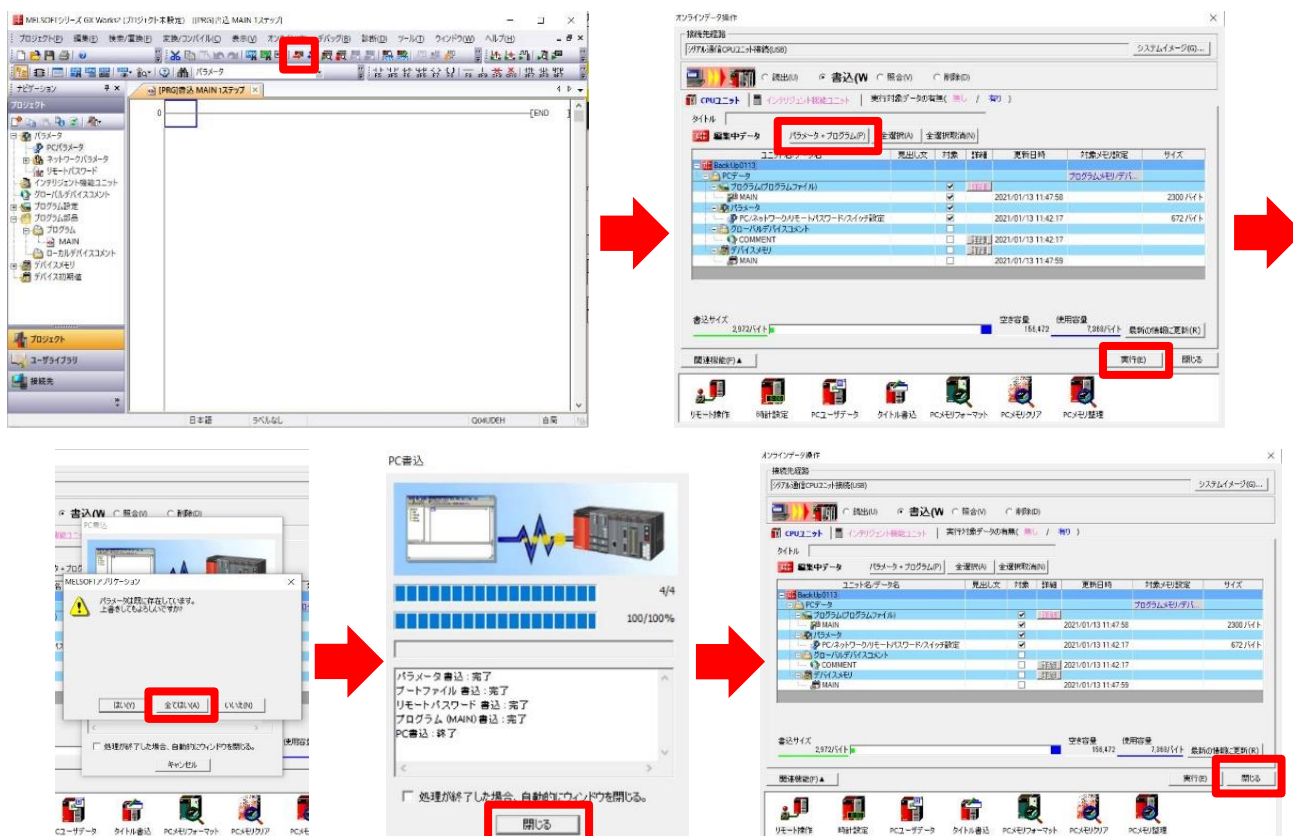
- ① 画面左下[プロジェクト]タブをクリック > [パラメータ]内[PC パラメータ]をダブルクリック > [内蔵 Ethernet ポート設定]をクリック



- ② この例では IP アドレスを[192.168.0.39]に設定 > [RUN 中書き込みを許可する]にチェック > [オープン設定]ボタンをクリック > この例では内蔵 Ethernet ポートオープン設定画面の2行目で[プロトコル]、[オープン方式]、[自局ポート番号]にそれぞれ[TCP]、[MC プロトコル]、[12288]を選択 > [設定終了]ボタンをクリック > [設定終了]ボタンをクリック

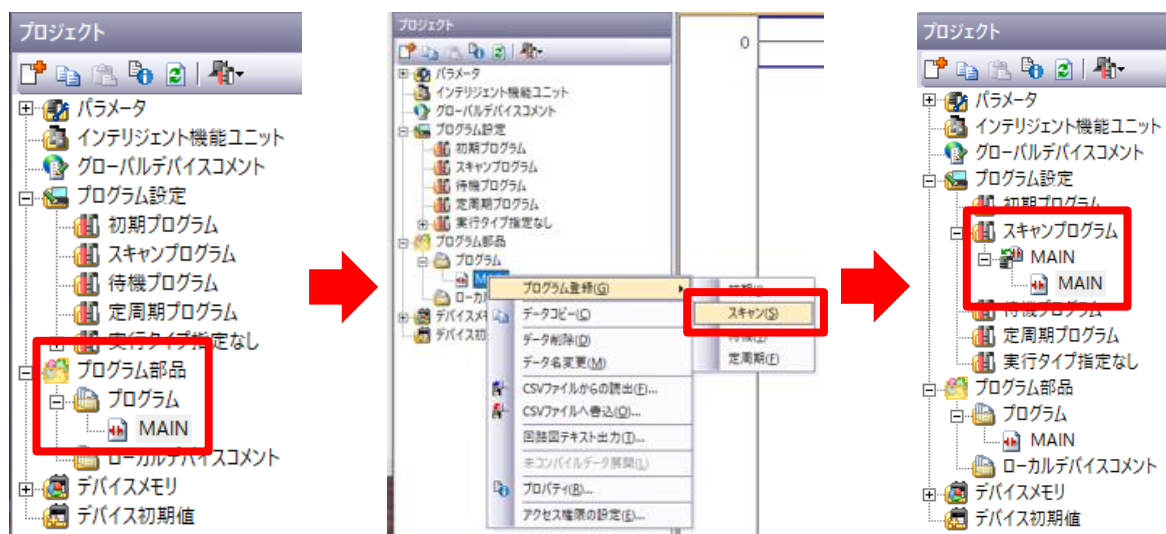


- ③ シーケンサ本体の[RUN/STOP]スイッチを[STOP]にする > GX Works のホーム画面上部にあるツールバー内[PC 書込]をクリック > [パラメータ+プログラム]ボタンをクリック > [実行]ボタンをクリック > [全てはい]ボタンをクリック > PC 書込が終了したことを確認し、[閉じる]ボタンをクリック > [閉じる]ボタンをクリック > シーケンサ本体の[RUN/STOP]スイッチを一旦[RESET]にし、その後[RUN]にする



※以下、作成したプログラムをシーケンサの RUN 実行時にスキャンさせる場合の手順

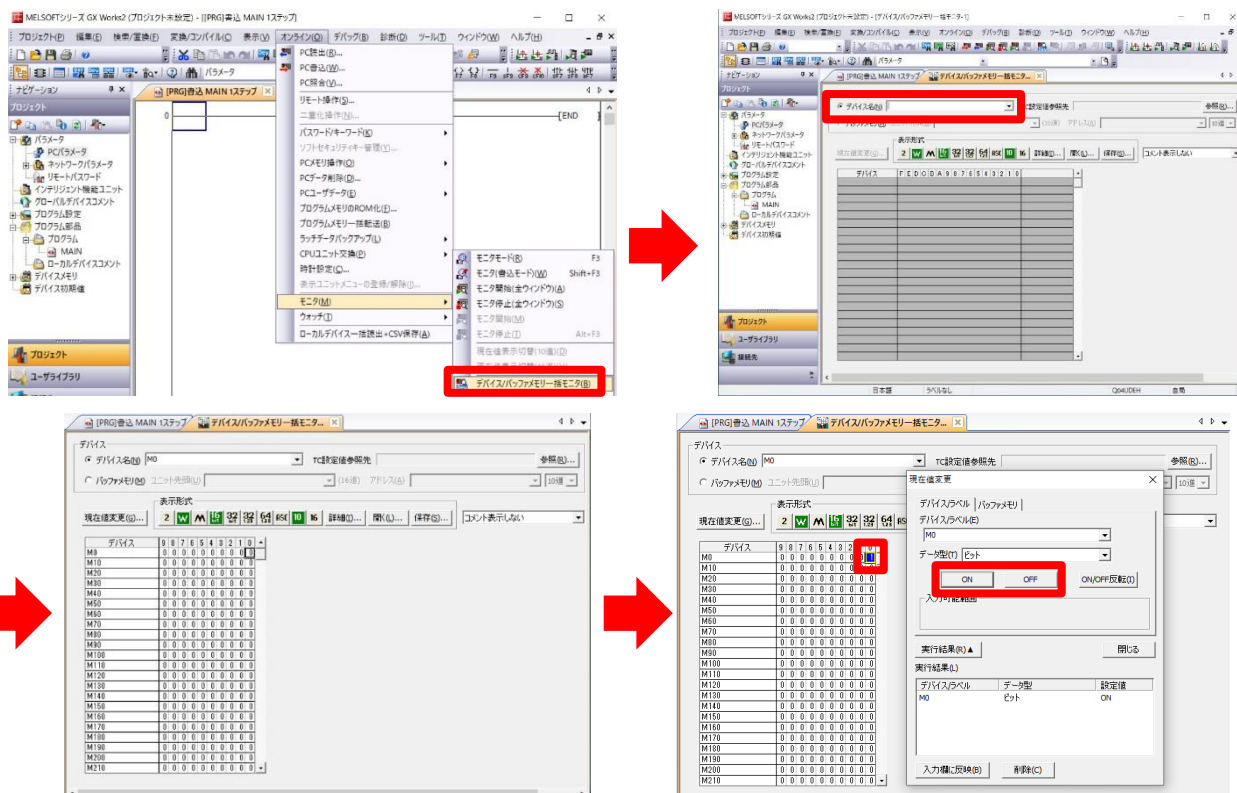
- ④ [プログラム部品] > [プログラム] > 任意のプログラム、この例では[MAIN]を右クリック > [プログラム登録] > [スキャン]をクリック > [プログラム設定] > [スキャンプログラム]内にプログラム[MAIN]が追加



## 4. コントロールブロックを使った読み取り指令

### 4.1 モニタ設定

- ① メニューバー内[オンライン] > [モニタ] > [デバイスバッファメモリー一括モニタ]をクリック > [デバイス名]に任意のデバイス名を入力し、[Enter]キーを押す > 任意の番地をダブルクリック > [ON/OFF]ボタンでビットの ON/OFF を操作

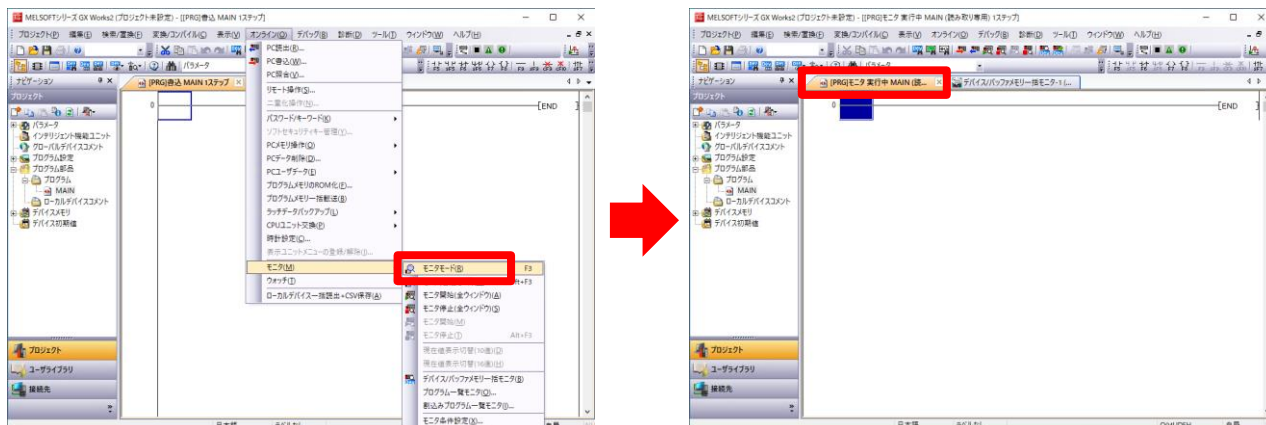


※以下、モニタモードの設定手順

モニタモードにすると各デバイスの ON/OFF 等の状態をラダー図上で確認できます。

ラダー図を変更する場合は再度書込モードにして下さい。

- ② メニューバー内[オンライン] > [モニタ] > [モニタモード]をクリック > [モニタモード]になっていることを確認



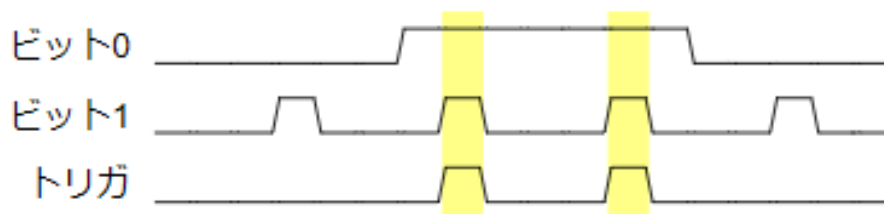
## 4.2 コントロールブロックを用いたシーケンサによる In-Sight D900の制御

下記は D900の**コントロールブロック**の定義です。

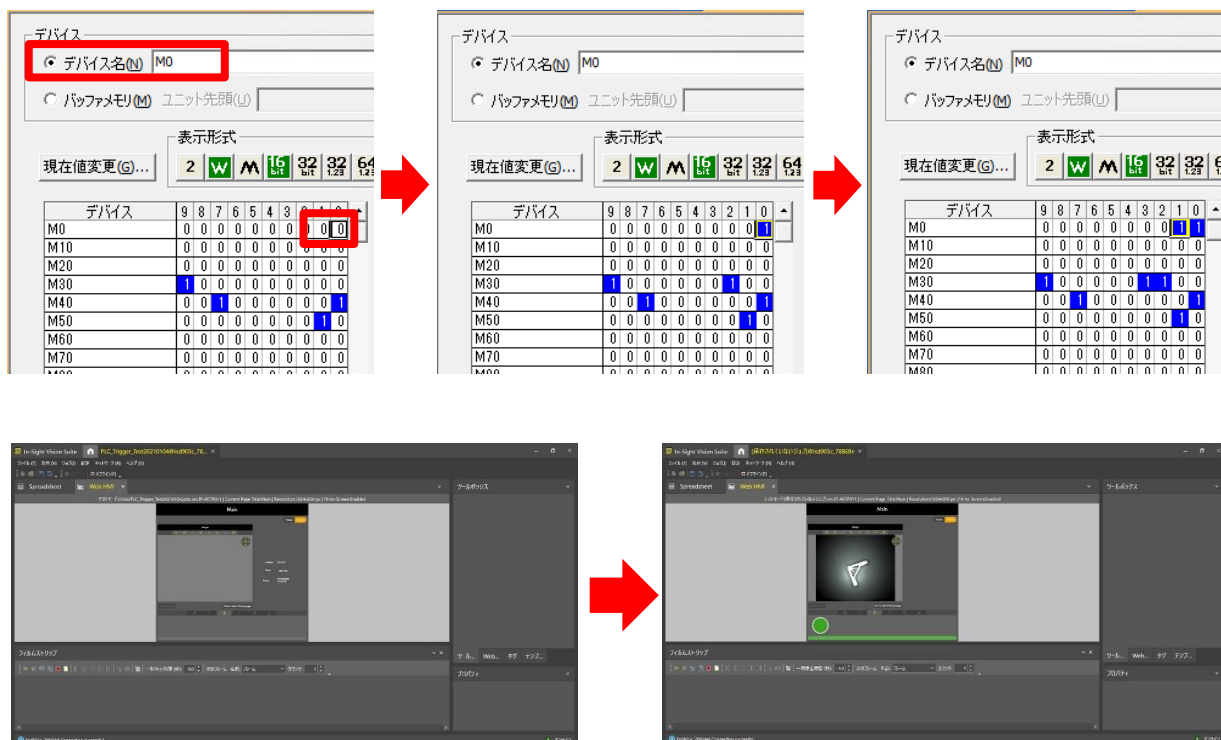
| Byte | Bit 7            | Bit 6            | Bit 5            | Bit 4            | Bit 3                   | Bit 2                 | Bit 1                   | Bit 0            |
|------|------------------|------------------|------------------|------------------|-------------------------|-----------------------|-------------------------|------------------|
| 0    | Set Paused       | Reserved         |                  | Execute Command  | Inspection Results Ack  | Buffer Results Enable | Trigger                 | Trigger Enable   |
| 1    | Reserved         |                  |                  |                  |                         |                       |                         |                  |
| 2    | Reserved         |                  |                  |                  | Clear Exposure Complete | Clear Error           | Initiate String Command | Set User Data    |
| 3    | External Event 7 | External Event 6 | External Event 5 | External Event 4 | External Event 3        | External Event 2      | External Event 1        | External Event 0 |

### 4.2.1 トリガ機能

トリガを掛けるには、まず D900 の**コントロールブロック**バイト 0 のビット 0 に対応する M0 (Trigger Enable)を 0→1 にしてからビット 1(Trigger)を 0→1 にします。続いてトリガを掛けるには、一旦ビット 1(Trigger)を 1→0 にリセットしてから、あらためて 0→1 にセットするとトリガが掛かります。ビット 1(Trigger)でトリガを掛ける際、ビット 0(Trigger Enable)は 1 にしておく必要がありますので、運用中はビット 0(Trigger Enable)を常に 1 にしておきます。



- ① 「GX Works2」にてデバイス名に M0と入力し、PC の Enter キーを押す > 該当箇所のビットをクリック > [ON]ボタンをクリック



## 4.2.2 JOB 切り替え機能

下記は D900の**入力ブロック**の定義です。

| Byte       | Signal                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 ... 1    | Command ID. The available IDs are as follows: <ul style="list-style-type: none"> <li>0x0000–0x03E7: Change to job <i>N</i> (where <i>N</i> is the command ID from 0–999).</li> <li>0x3E8–0x7CF: Change to recipe <i>N</i> (where <i>N</i> is the command ID from 1000–1999).</li> <li>0x2000: Clear Online Enabled (only supported when the vision system is Online).</li> <li>0x2001: Set Online Enabled (only supported when the vision system is Online).</li> <li>0x2002: Reboot system.</li> <li>0x4000: Change job by name (the name is transferred from UserData).</li> <li>0x4001: Change recipe by name (the name is transferred from UserData).</li> </ul> |
| 2 ... 3    | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| 4 ... 1920 | User Data                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |

D900が一時停止状態の時 JOB の切り替えが可能となります。レシピとジョブを切り替える方法は名前を指定するものと ID を指定するものがありますが、ここでは ID を指定したジョブの切り替え手順について説明します。

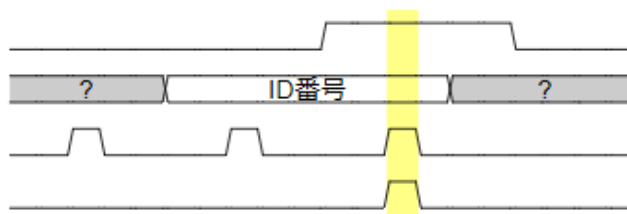
D900の**コントロールブロック**バイト 0のビット7に対応する M7 (Set Paused)を0→1にすると D900が一時停止状態になります。D900の**入力ブロック**バイト0に対応する D0(Command ID)で切り替える JOB の ID 番号をセットします。**コントロールブロック**バイト 0のビット4(Execute Command) を0→1にすると JOB 切り替えが実行されます。ID 番号に該当するものがない場合は JOB の切り替えは行われません。

ビット7(一時停止状態への変更指令)

D0(JOBID を指定)

ビット4(JOB 切り替えの実行指令)

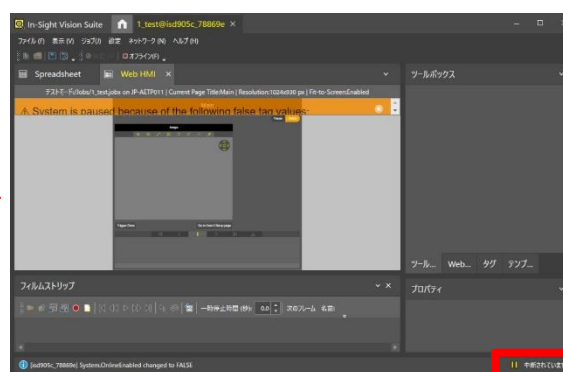
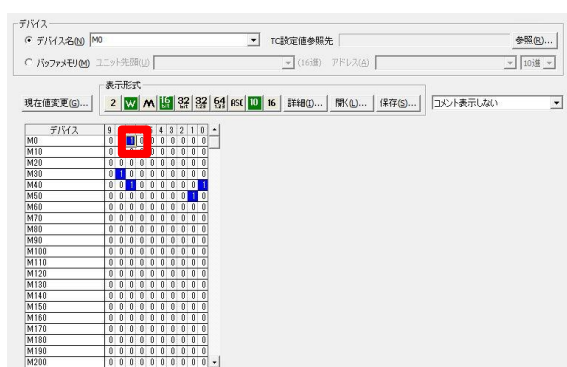
JOB 切り替え



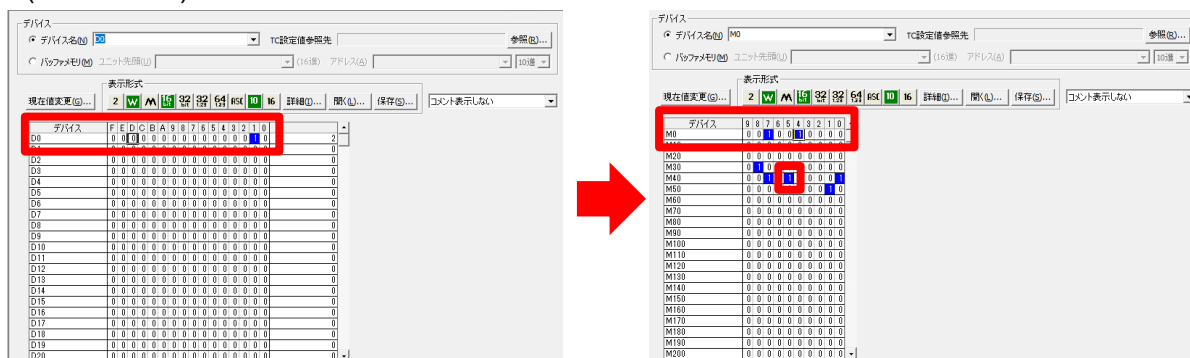
- ① 事前に JOB ファイル名の先頭に1\_、2\_、・・・、というような認識用の ID 番号を振っておきます。

| ファイル管理         |                      |      |        |  |
|----------------|----------------------|------|--------|--|
| Jobs           |                      |      |        |  |
| isd905c_78869e |                      |      |        |  |
| Name           | Date Modified        | Type | Size   |  |
| 1_test         | 7/14/2020 7:45:00 AM | jobx | 770 KB |  |
| 2_test         | 7/14/2020 7:44:00 AM | jobx | 770 KB |  |
| 3_test         | 7/14/2020 6:40:00 AM | jobx | 770 KB |  |

- ② ビット7(Set Paused)を0→1にする > Set Paused コマンドが実行され D900が一時停止状態になる。



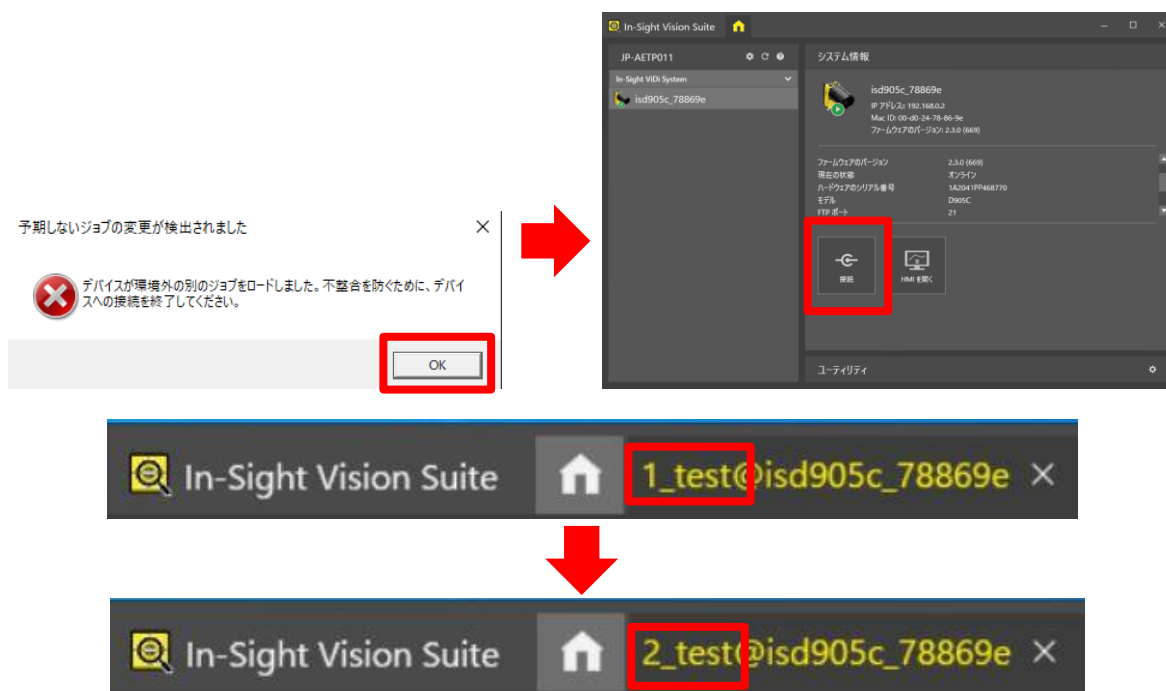
- ③ 次に切り替える JOB の ID をセットします。この例では D900の**入力ブロック**バイト0に対応する D0(Command ID)に番号2をセットし、ID 番号2の JOB ファイルをロードします。  
D0のビット1に1をセット(2進数表記10は10進数表記で2) > **コントロールブロック**バイト 0のビット4に対応する M4(Execute Command)に1をセット > D900の**ステータスブロック**バイト1のビット5(Command Completed)に対応する M45が ON になったら M4(Execute Command)、M7(Set Paused)を1→0にする



下記は D900の**ステータスブロック**の定義です。

| Byte | Bit 7                | Bit 6                | Bit 5                | Bit 4                | Bit 3                | Bit 2                  | Bit 1                | Bit 0                |
|------|----------------------|----------------------|----------------------|----------------------|----------------------|------------------------|----------------------|----------------------|
| 0    | Online               |                      | Offline Reason       |                      | Acq Error            | Reserved               | Trigger Ack          | Trigger Ready        |
| 1    | Error                | Command Failed       | Command Completed    | Command Executing    | Results Valid        | Results Buffer Overrun | Inspection Completed | System Busy          |
| 2    |                      | Reserved             |                      | Job Pass             | Exposure Complete    | String Command Error   | String Command Ack   | Set User Data Ack    |
| 3    | External Event Ack 7 | External Event Ack 6 | External Event Ack 5 | External Event Ack 4 | External Event Ack 3 | External Event Ack 2   | External Event Ack 1 | External Event Ack 0 |

- ④ エラーメッセージが表示された場合は[OK]ボタンをクリック > [接続]ボタンをクリック  
JOB ファイルが1\_test から2\_test へ切り替わります。



## 4.2.3 シーケンサと In-Sight D900間のデータの送受信

下記は D900の**出力ブロック**の定義です。

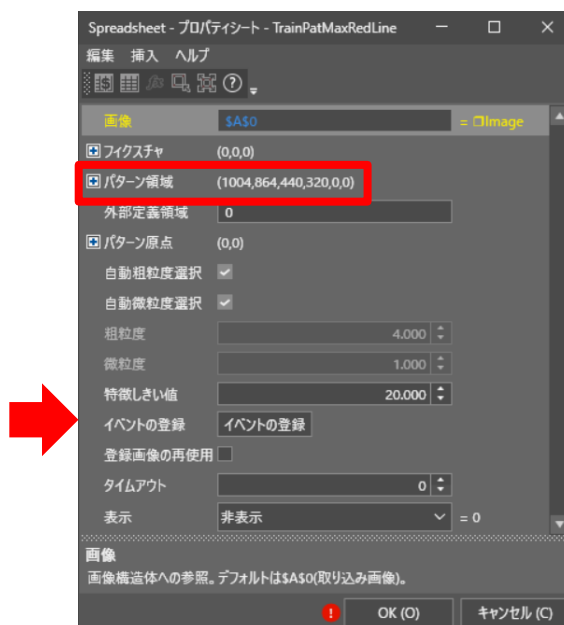
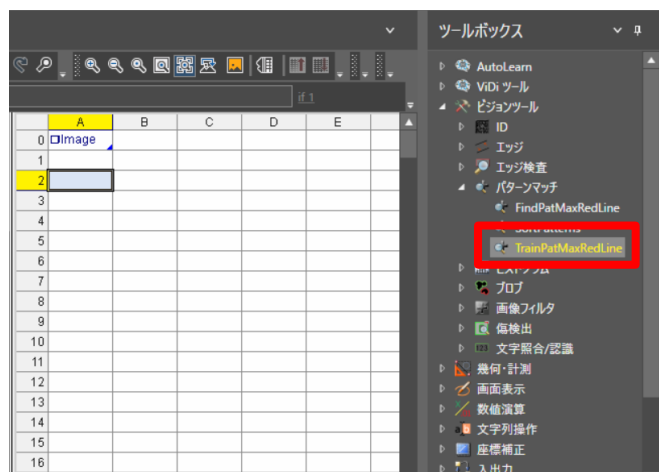
| Byte | Signal                 |
|------|------------------------|
| 0..1 | Current Job ID         |
| 2..3 | Error Code             |
| 4..5 | Acquisition ID         |
| 6..7 | Inspection ID          |
| 8..9 | Inspection Result Code |
| 10.. | Inspection Results     |

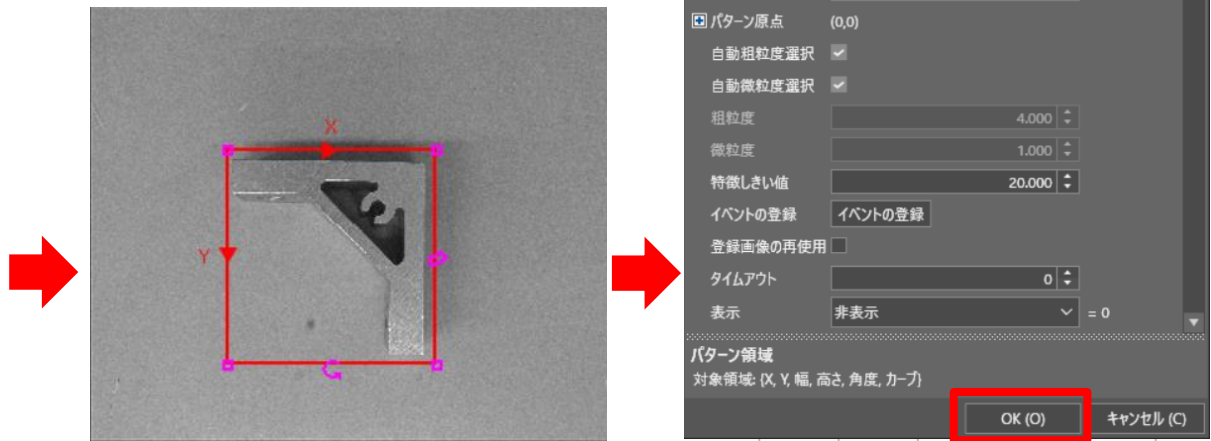
ここではパターンマッチツールを使用して、以下を例にデータの送受信手順を説明します。  
シーケンサから D900へ送信した数値データはパターンマッチツールのパラメータに反映し、  
パターンマッチツールの検出結果は D900からシーケンサへ送信します。

パターンマッチツールのパラメータ値は、D900の**入力ブロック**バイト4以降に対応する D2以降の User Data 領域に設定します。D900の**コントロールブロック** バイト2のビット0(Set User Data)に対応する M16が0→1に変更されると、**ステータスブロック** バイト2のビット0(Set User Data Acq)に対応する M48が0→1となり、User Data 領域の変更が反映されます。データ読み取り更新イベントが起きると User Data の値が D900に反映されます。

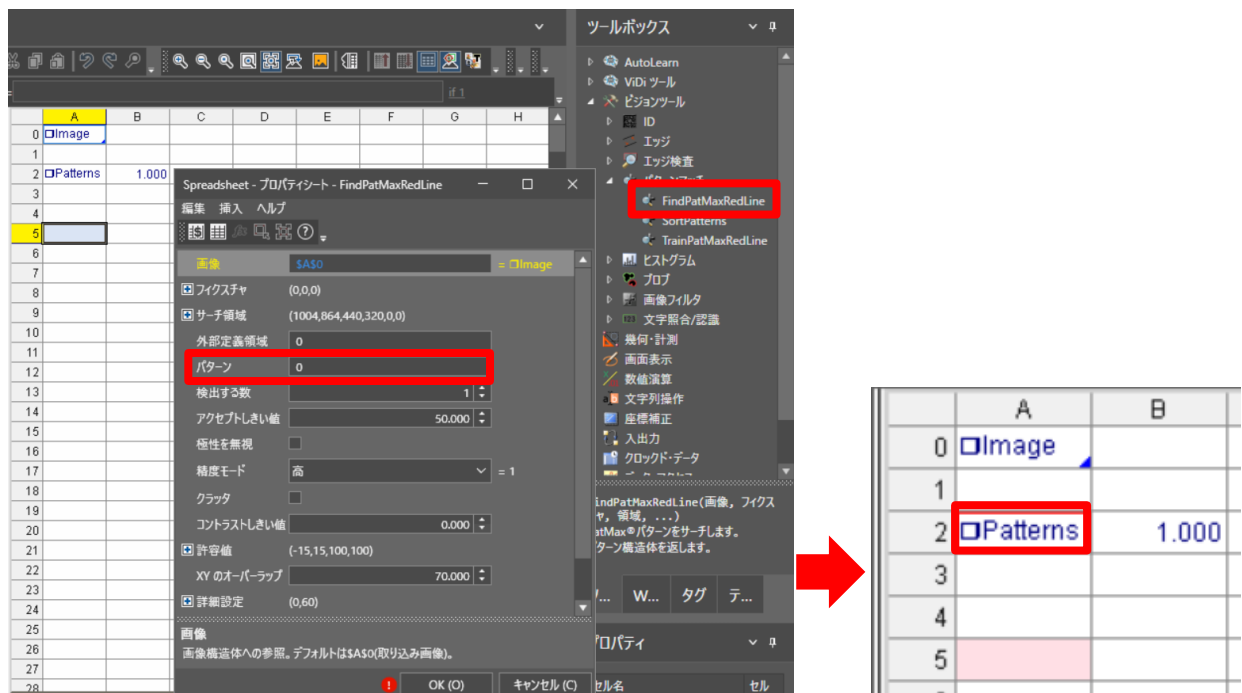
データ読み取り更新イベントが起きると、パターンマッチツールの検出結果は、D900の**出力ブロック** バイト10以降に対応する D105以降の Inspection Results 領域に出力されます。

- ① [ツールボックス]内、[ビジョンツール] > [パターンマッチ] > [TrainPatMaxRedLine]関数をスプレッドシートのセル A2へドラッグ&ドロップ > TrainPatMaxRedLine 関数のプロパティシート内、[パターン領域]をダブルクリック > 任意の登録モデル領域を選択し、PC の Enter キーを押す > [OK]ボタンをクリック

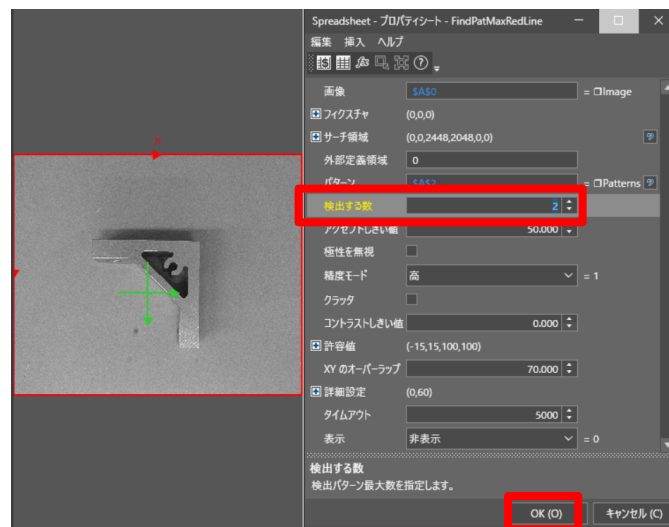
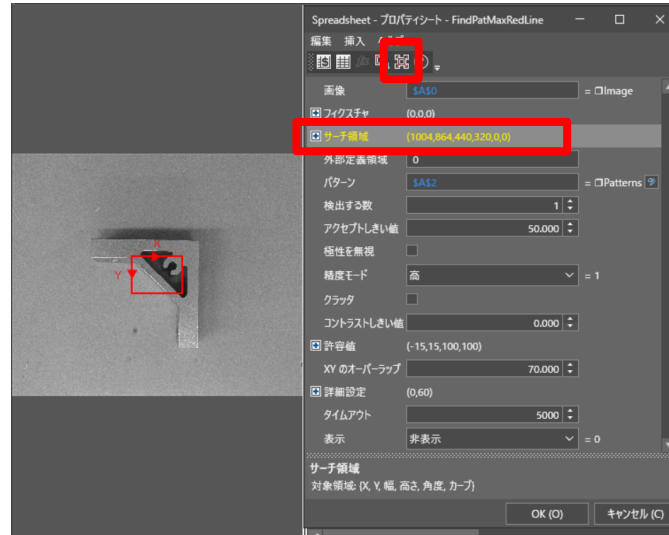




- ② [FindPatMaxRedLine]関数をスプレッドシートのセル A5へドラッグ & ドロップ > FindPatMaxRedLine 関数のプロパティシート内、[パターン]をダブルクリック > セル A2をクリックし、PC の Enter キーを押す



- ③ FindPatMaxRedLine 関数のプロパティシート内、[サーチ領域]をクリックし、[領域の最大化]ボタンをクリック > サーチ領域が視野全体に変更 > [検出する数]を[2]に設定 > 必要に応じてその他のパラメータを適宜調節 > [OK]ボタンをクリック > スプレッドシート上にパターンサーチの検出結果が自動出力される > 視野内にサンプルワークを2つ置き、[トリガ]ボタンをクリック > サンプルワーク2つ分の検出結果が表示されます。

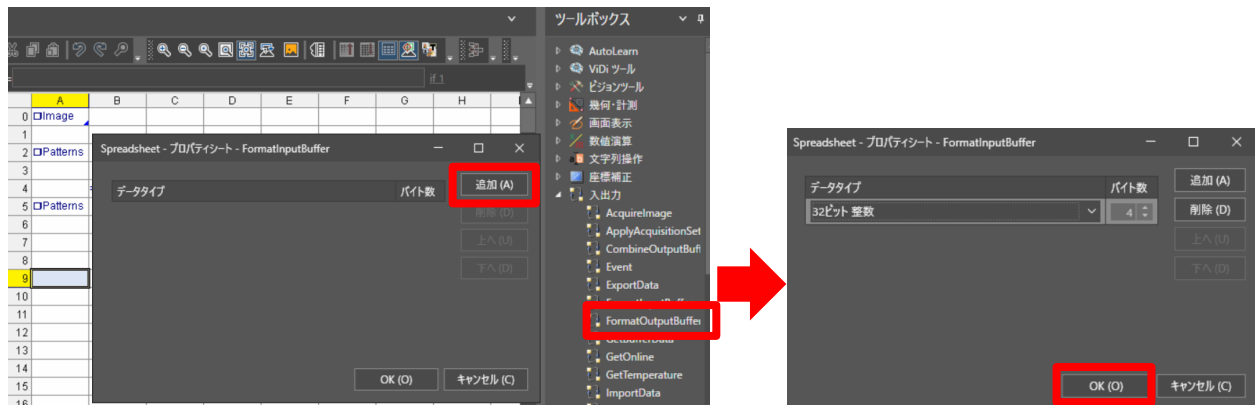


|   | A        | B     | C        | D        | E      | F       | G      |
|---|----------|-------|----------|----------|--------|---------|--------|
| 0 | Image    |       |          |          |        |         |        |
| 1 |          |       |          |          |        |         |        |
| 2 | Patterns | 1.000 |          |          |        |         |        |
| 3 |          |       |          |          |        |         |        |
| 4 | 番号       | X     | Y        | 角度       | スケール   | 得点      |        |
| 5 | Patterns | 0.000 | 1134.471 | 1176.979 | -0.005 | 100.000 | 95.400 |
| 6 |          | 1.000 | #ERR     | #ERR     | #ERR   | #ERR    | 0.000  |

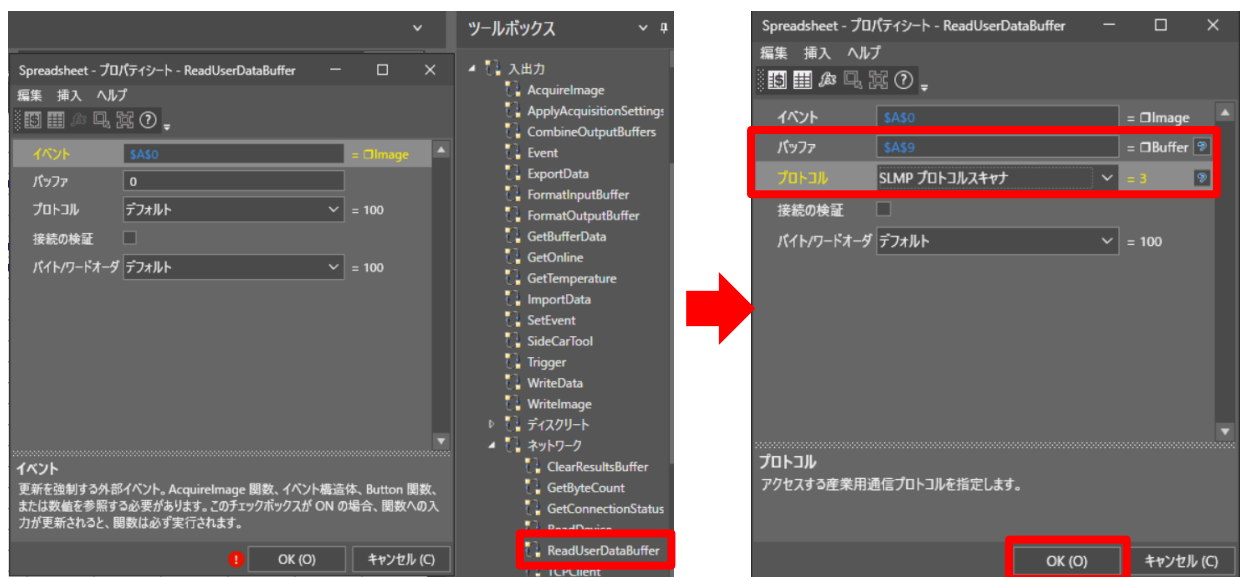
|    | A        | B     | C        | D        | E      | F       | G      |
|----|----------|-------|----------|----------|--------|---------|--------|
| 0  | Image    |       |          |          |        |         |        |
| 1  |          |       |          |          |        |         |        |
| 2  | Patterns | 1.000 |          |          |        |         |        |
| 3  |          |       |          |          |        |         |        |
| 4  | 番号       | X     | Y        | 角度       | スケール   | 得点      |        |
| 5  | Patterns | 0.000 | 686.817  | 1314.963 | -4.965 | 100.000 | 66.061 |
| 6  |          | 1.000 | 1691.796 | 1208.558 | -4.287 | 100.000 | 53.191 |
| 7  |          |       |          |          |        |         |        |
| 8  |          |       |          |          |        |         |        |
| 9  |          |       |          |          |        |         |        |
| 10 |          |       |          |          |        |         |        |
| 11 |          |       |          |          |        |         |        |
| 12 |          |       |          |          |        |         |        |
| 13 |          |       |          |          |        |         |        |
| 14 |          |       |          |          |        |         |        |
| 15 |          |       |          |          |        |         |        |
| 16 |          |       |          |          |        |         |        |
| 17 |          |       |          |          |        |         |        |
| 18 |          |       |          |          |        |         |        |
| 19 |          |       |          |          |        |         |        |
| 20 |          |       |          |          |        |         |        |

- ④ ここからはシーケンサから D900へ数値データを送信するための設定手順です。

[FormatInputBuffer]関数をスプレッドシートのセル A9へドラッグ&ドロップ > FormatInputBuffer関数のプロパティシート内、[追加]ボタンをクリック > [OK]ボタンをクリック

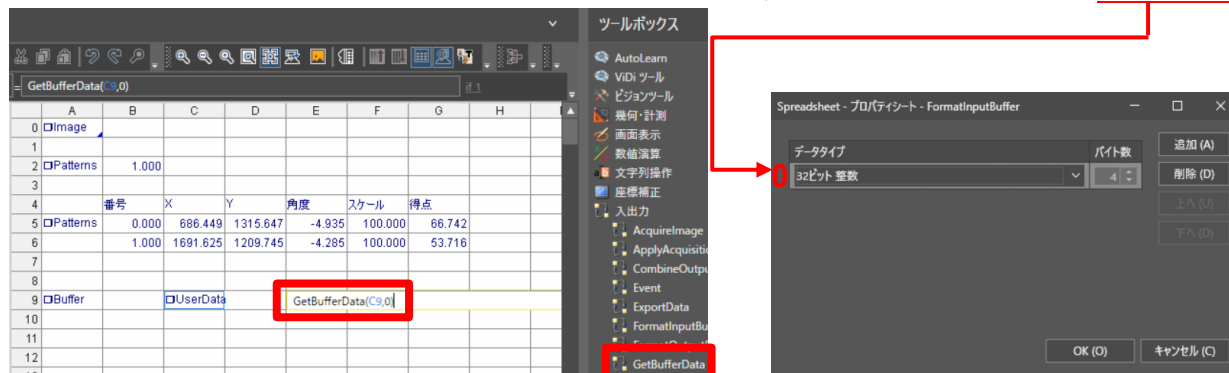


- ⑤ [ReadUserDataBuffer]関数をスプレッドシートのセル C9へドラッグ & ドロップ > ReadUserDataBuffer関数のプロパティシート内、[バッファ]をダブルクリックし、スプレッドシートの FormatInputBuffer 関数(セル A9)を参照 > [プロトコル]を[SLMP プロトコルスキャナ]に設定 > [OK]ボタンをクリック

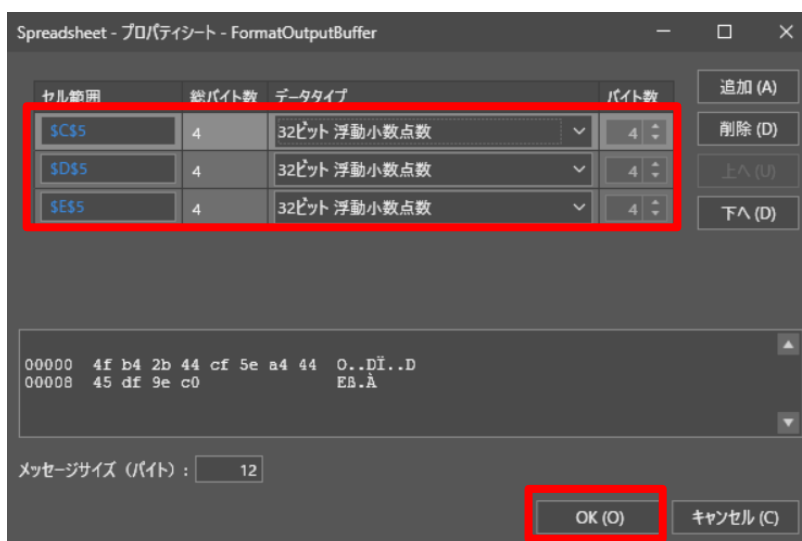
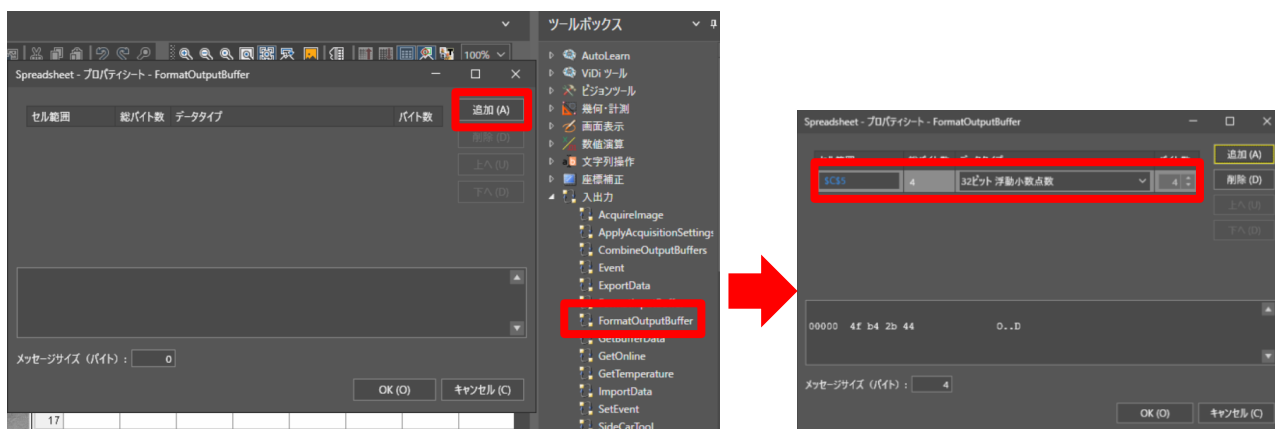


- ⑥ [GetBufferData]関数をスプレッドシートのセル E9へドラッグ＆ドロップ > GetBufferData 関数の引数を設定 > PC の Enter キーを押す > スプレッドシート内に[GetBufferData]関数が追加(この例ではセル E9に GetBufferData(C9,0)が追加) > ここまでの手順により、シーケンサから D900へ数値データが送信できるようになります。

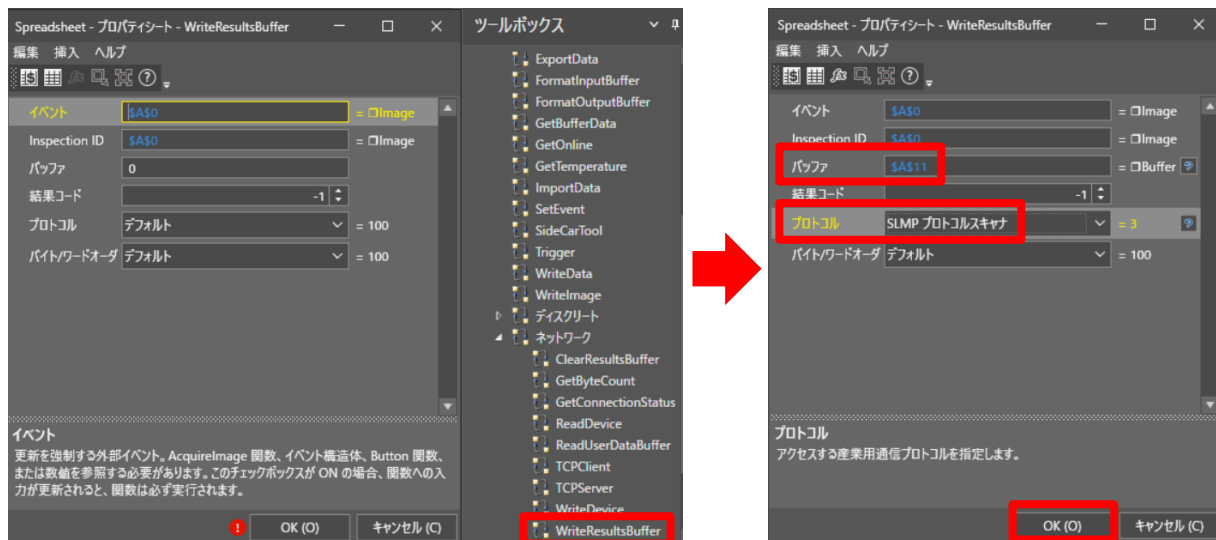
GetBufferData(ReadUserDataBuffer セル, インデックス番号)



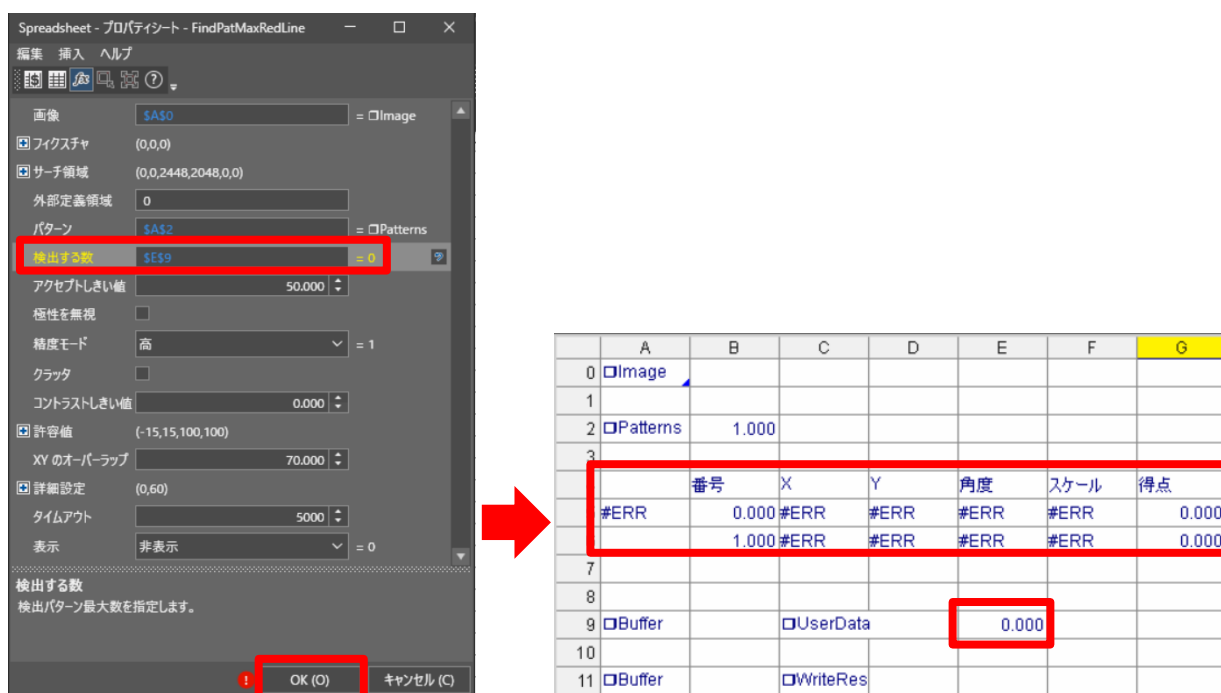
- ⑦ 次に D900からシーケンサへ数値データを送信するための設定手順です。  
[FormatOutputBuffer]関数をスプレッドシートのセル A11へドラッグ＆ドロップ > FormatOutputBuffer 関数のプロパティシート内、[追加]ボタンをクリック > スプレッドシートのセル C5を参照 > 同様の手順にてセル D5、セル E5を参照 > [OK]ボタンをクリック



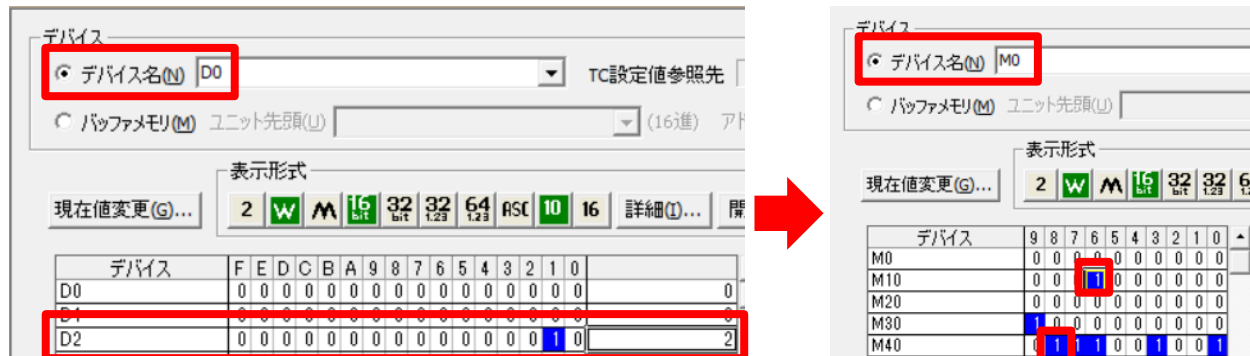
- ⑧ [WriteResultsBuffer]関数をスプレッドシートのセル C11へドラッグ & ドロップ > WriteResultsBuffer 関数のプロパティシート内、[バッファ]をダブルクリックし、スプレッドシートの FormatOutputBuffer 関数(セル A11)を参照 > [プロトコル]を[SLMP プロトコルスキャナ]に設定 > [OK]ボタンをクリック > ここまでの手順により、D900からシーケンサへ数値データが送信できるようになります。



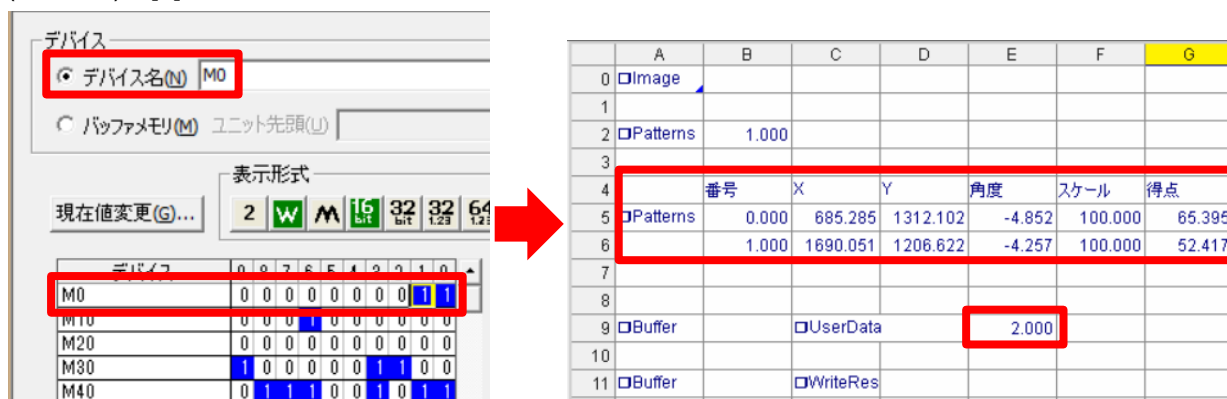
- ⑨ スプレッドシートの FindPatMaxRedLine 関数(セル A5)をダブルクリック > FindPatMaxRedLine 関数のプロパティシート内、[検出する数]をダブルクリックし、スプレッドシートの GetBufferData 関数(セル E9)を参照 > [OK]ボタンをクリック > この時点では FindPatMaxRedLine 関数の[検出する数]が0になるためエラーとなります。 > D900をオンライン状態にします。(7ページ手順⑧参照)



- ⑩ 「GX Works2」にてデバイス名に D0 と入力し、PC の Enter キーを押す > この例では D900 の **入力ブロック** バイト4に対応する D2(User Data)に番号2をセットします。(2進数表記10は10進数表記で2) > デバイス名に M0 と入力し、PC の Enter キーを押す > **コントロールブロック** バイト 2 のビット0に対応する M16(Set User Data)に1をセット > D900 の **ステータスブロック** バイト2のビット0に対応する M48 (Set User Data Acq)が1となります。



- ⑪ D900 の **コントロールブロック** バイト0のビット0に対応する M0 (Trigger Enable)を0→1にしてから M1(Trigger)を0→1にする > トリガがかかる(更新イベントが起きる) > GetBufferData 関数 (セル E9)に[2]が出力 > FindPatMaxRedLine 関数で2つのパターンが検出できます。



- ⑫ デバイス名に D100 と入力し、PC の Enter キーを押す > [実数(32ビット)]ボタンをクリック > D900 の **出力ブロック** バイト10以降に対応する D105以降の Inspection Results 領域に FormatOutputBuffer 関数で選択した3つの値(スプレッドシートのセル C5、D5、E5)が出力



## 4.3 補足情報

### 4.3.1 画像取り込みエラーの監視

以下、画像取り込みエラー発生回数の測定例

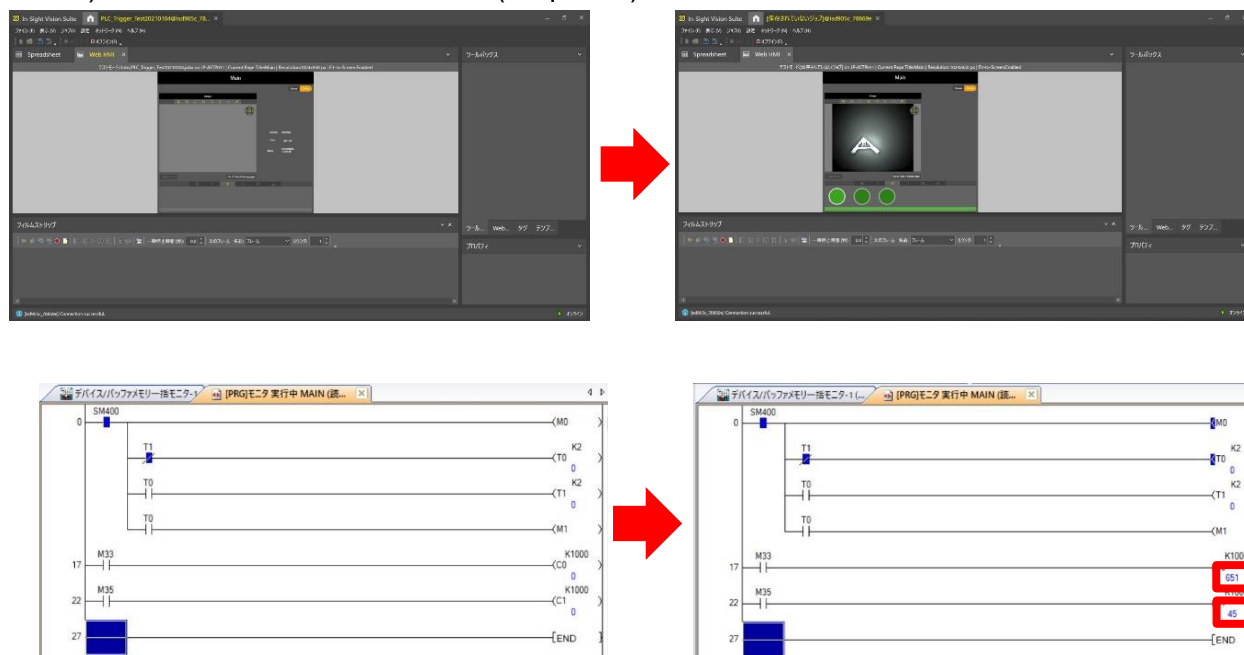
- ① D900には処理時間が長くなるようにあらかじめプログラムスキャン時の処理に負荷がかかるようなジョブをロードしておきます。「GX Works2」にて11ページ手順③、④を参考に、作成した以下のプログラムをシーケンサに書き込みます。

※赤枠内のプログラムは M0(Trigger Enable)を常に ON にし、タイマー(T)を使って0.2秒毎に M1(Trigger)が0→1→0→1→0→....というように ON/OFF を繰り返す動作を行います。

※青枠内のプログラムはカウンタ(C)を使って M33(Trigger Ack)、M35(Acq Error)が ON になった回数をカウントします。以下の例では引数を K1000とし、1000回までカウントできるようにしています。

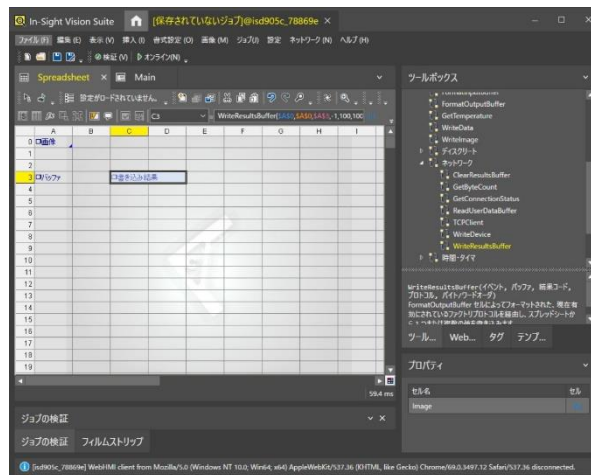


- ② D900をオンライン状態にする > GX Works2にて12ページ手順②を参考にモニタモードに設定 > シーケンサ本体の [RUN/STOP]スイッチを[RUN]に設定 > しばらく動作させトリガ回数 M33(Trigger Ack)と画像取り込みエラー回数 M35(Acq Error)を確認

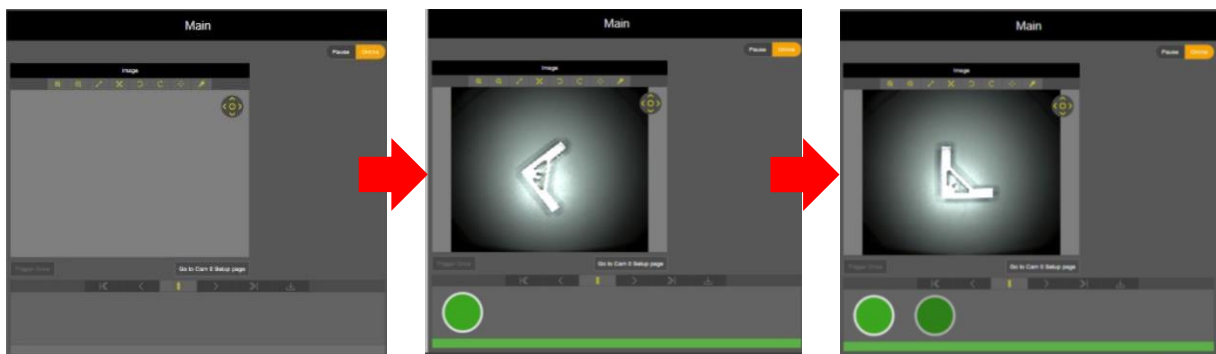


## 4.3.2 Inspection Completed の監視

- ① D900に WriteResultsBuffer 関数を使用した JOB(以下の例は WriteResultBuffer 関数と FormatOutputBuffer 関数のみを使用した JOB)をロード > D900をオンライン状態にする。



- ② トリガを掛けるとその都度 D900のステータスブロックバイト1のビット2(Inspection Completed)に対応する M41が ON/OFF を繰り返します。



| 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  | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| M40  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| M50  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| M60  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| M70  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

トリガ

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

トリガ

| デバイス | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|---|---|---|---|---|---|---|---|---|---|
| M0   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 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 | 1 | 0 | 0 |
| M40  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| M50  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| M60  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| M70  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |