Engineering Note

SPC

Programing Guide EtherNet/IP PLC

METTLER TOLEDO

© METTLER TOLEDO 2023

No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of METTLER TOLEDO.

U.S. Government Restricted Rights: This documentation is furnished with Restricted Rights.

Copyright 2023 METTLER TOLEDO. This documentation contains proprietary information of METTLER TOLEDO. It may not be copied in whole or in part without the express written consent of METTLER TOLEDO.

COPYRIGHT

METTLER TOLEDO® is a registered trademark of Mettler-Toledo, LLC. All other brand or product names are trademarks or registered trademarks of their respective companies.

METTLER TOLEDO RESERVES THE RIGHT TO MAKE REFINEMENTS OR CHANGES WITHOUT NOTICE.

FCC Notice

This device complies with Part 15 of the FCC Rules and the Radio Interference Requirements of the Canadian Department of Communications. Operation is subject to the following conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his or her expense.

Menu

1. Overview

This Engineering Note demonstrates the integration of the METTLER TOLEDO SPC with an EtherNet/IP PLC. Go to www.mt.com/spc to download all the necessary files and documents.

Note: The configuration used in this sample code is based on the default settings：

- **Rockwell Studio5000:** **Version 32**
- **PLC:** **1769-L30ER**
- **SAI DATA FORMAT:** **8-BLOCK FORMAT**
- **SPC IP ADDRESS:** **192.168.0.55**
- **AOP FILE:** **AOP V1.01.04**

It is recommended to integrate one SPC into the PLC EtherNet/IP network and go through the sample code and each Function Block before adding more devices.

2. Setup of Project Development Environment

2.1. Confirm EDS Installation

This sample code project utilizes an EDS file for the SPC. These files can be found on www.mt.com/spc.

To confirm installation of SPC AOP file:

1. In any Studio 5000 project, right click on **Ethernet** within the I/O Configuration folder in the controller organizer.
2. Select **New Module**….

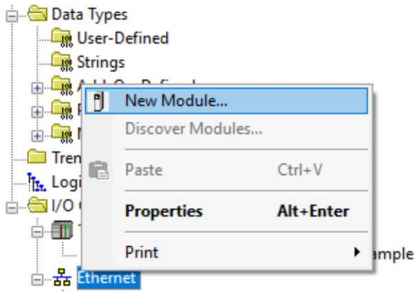

Figure 2-1: Try to add a new module to confirm EDS is installed

3. Search SPC

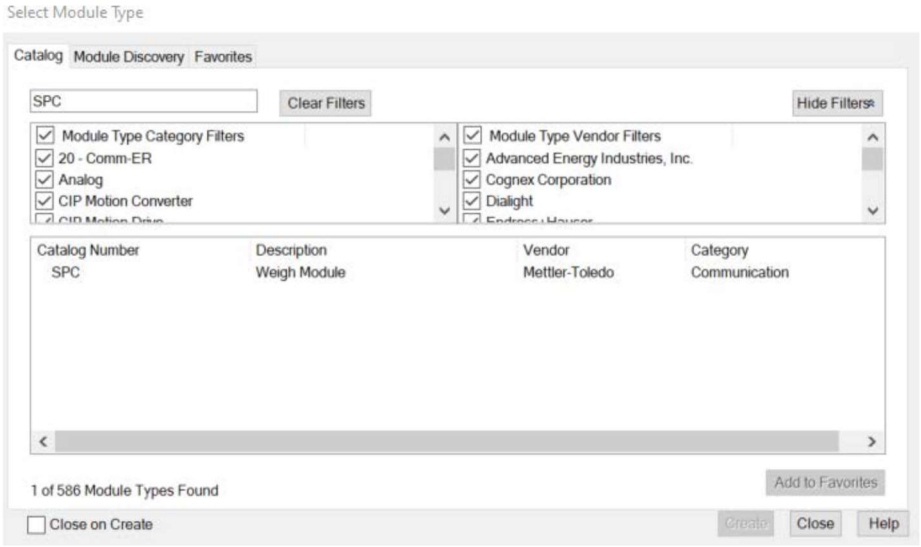

Figure 2-2: Search for SPC

If the EDS is installed, there should be an option for SPC. If the search returns no results, follow these steps to install the AOP:

1. Go to the SPC download page: www.mt.com/spc
2. Click the EDS file to begin the download.
3. Once the download is complete, unzip the folder
4. Use the EDS installation tool in Studio5000 to install the EDS.

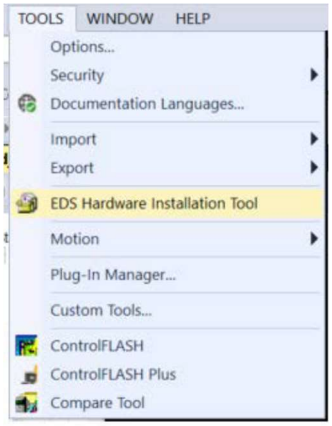

Figure 2-3**: Use the EDS Hardware Installation Tool to complete installation**

2.2. Import Example as a New Project

To import the examples, Studio5000 V24 or above is required.

1. To import the project to Studio5000, click **File-> Open**.

2. Select the .ACD file and click open. The project will load.

Figure 2-4: Import Project

2.3. Import an Example to an Existing Project

Add an SPC to the I/O Configuration in the existing project. See the first steps of Section 3 for more information on how to complete this. Using the name **SPC** and the IP Address **192.168.0.55** will require no changes to the sample code. If a different name or IP address is required, steps explaining what changes to make are provided below.

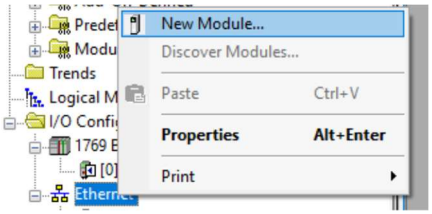

Figure 2-5: Add SPC to the existing project

Copy the Add-On Instructions from the Add-On Instructions folder in the Controller Organizer of the sample project and paste in the same location in the existing project.
Figure 1-6: Copy/Paste AOIs

Figure 2-6: Copy/Paste AOIs

Copy the controller tags from the sample code project and paste in the controller tags of the existing project. Make sure not to copy the **SPC:I** and **SPC:O** tags since those are already present in the existing project.

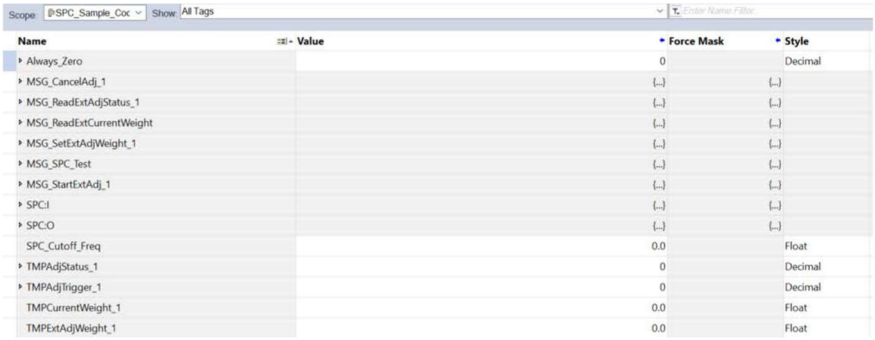

Figure 2-7: Copy/Paste Controller Tags

Copy the Main Program local tags from the sample project and paste in the tags for the existing project.

Figure 2-8: Copy/Paste Main Program Local Tags

Copy the **MT_SPC_Application** routine from the sample project and paste in the existing project.

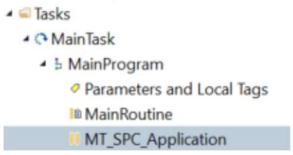

Figure 2-9: Copy/Paste the Routine

Make sure something in the existing project calls the MT_SPC_Application. Any AOIs that automatically monitor weight conditions will not run if nothing calls this routine.

7. If a name other than **SPC** was used as the name of the weight sensor in the project, replace every use of **SPC** in the AOI instances with the name given to the weight sensor in the project.

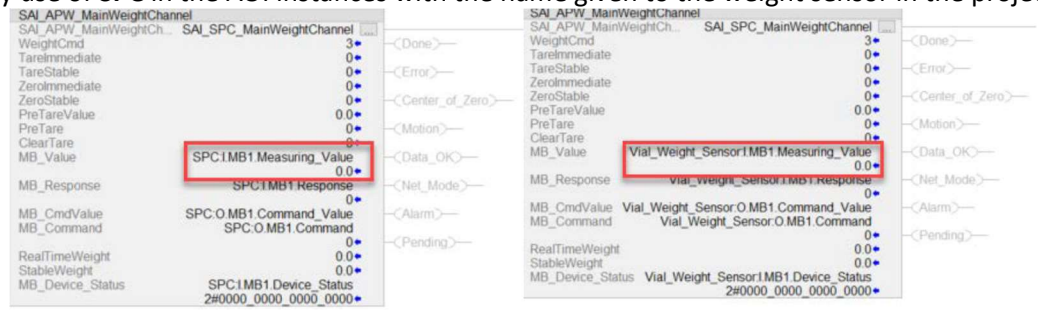

Figure 2-10: Example of name "Vial_Weight_Sensor" used in project

2.4. Configure Controller Type

Please note that this is only necessary if using the sample code as the basis for the PLC project. If importing the routine and AOIs into an already existing project, this is unnecessary.

Right-click the project's controller, select **Properties**, and set the controller type.

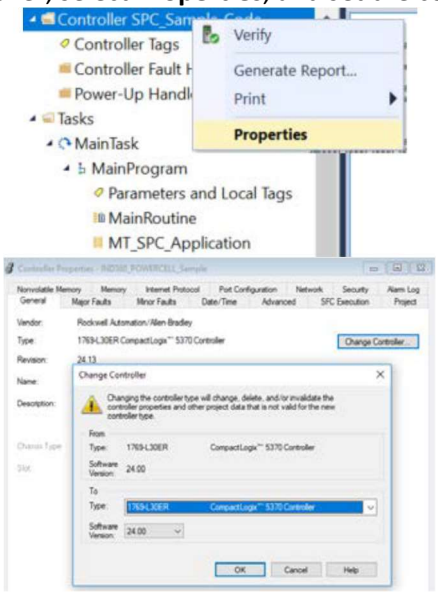

Figure 2-11: Configure controller type

3. Add-On Instructions (AOI)

About the configuration of the Messages in the add-on instructions:
Add-On Instructions that use acyclic communication functions, such as external adjustment need to create and configure Message variables, set paths and indicate which device to use for acyclic communication. The path of each device is different when multiple devices are networked.

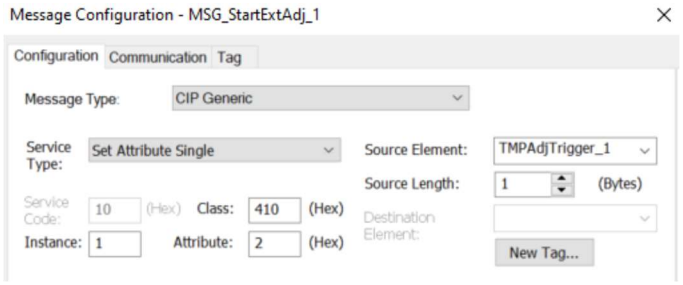

Figure 3-1: Configure Message Variable

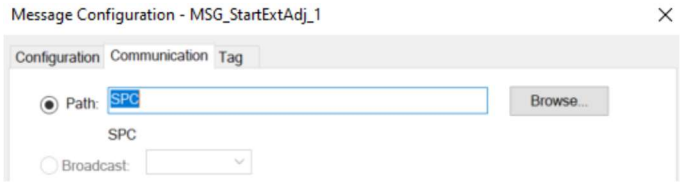

Figure 3-2: Set the path of the Message variable

Message configuration must be accessed via the AOI block in the main program. The message configuration cannot be accessed inside of the AOI logic files.

Figure 3-3: Access Message Configuration

3.1. Cyclic Weight Data

Read the real-time and stable weight from the sensor. When performing zero and tare commands, the weight will stop updating.

Trigger execution of stable tare, stable zero, immediate tare, immediate zero and clear tare by setting that particular bit high. The response can be read, and there are flags for execution success and failure to indicate the result.

After the zero and tare commands are completed, the AOI will automatically restore whatever command is in WeightCmd and weight will be reported again. Typical values for WeightCmd are 0 (report gross weight) or 3 (report net weight). The DataOK bit is reset to 0 during overload, underload, adjustment and several other scenarios which can be used to judge abnormal conditions.

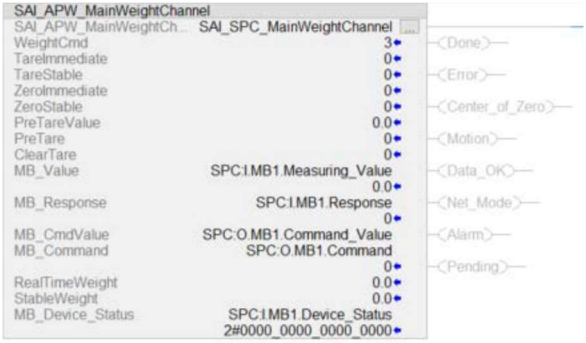

Figure 3-4: SAI_APW_MainWeightChannel AOI

Input Parameters	Data Type	Description
WeightCmd	INT	Use this value to request the SPC to report weight. When a zero or tare cyclic command is sent, the SPC stops reporting weight. This AOI will automatically restore this command once the zero or tare command completes. 0 or 1 = Report gross weight 2 = Report tare weight 3 = Report net weight 5 = Report gross weight value (with internal resolution) 6 = Report tare weight value (with internal resolution) 7 = Report net weight value (with internal resolution)
TareImmediate	BOOL	Set = 1 to issue tare command regardless of whether the weight value is stable or not. Net weight will not automatically be reported after tare command is issued. Recommended to use cyclic command 3 (report net weight) in WeightCmd input to receive net weight
TareStable	Bool	Set = 1 to issue tare command to SPC when weight is stable. Command will timeout if remain

		within the stability criteria (+/- 1d within 0.3 seconds default) for a predefined timeout range (3 seconds default). Net weight will not automatically be reported after tare command is issued. Recommended to use cyclic command 3 (report net weight) in WeightCmd input to receive net weight
ZeroImmediate	BOOL	Set = 1 to issue zero command regardless of whether the weight value is stable or not. This is only intended for minor changes to the zero point due to drifting. Command will return an error if weight value is not within the zero range.
ZeroStable	BOOL	Set = 1 to issue zero command to SPC when weight is stable. Command will timeout if remain within the stability criteria for a predefined timeout range. This is only intended for minor changes to the zero point due to drifting. Command will return an error if weight value is not within the zero range.
PreTareValue	Real	Configure with preset tare value. This value will not be sent to SPC until Pretare input is set to 1.
Pretare	BOOL	Set = 1 when ready to perform preset tare using the value from PreTareValue.
ClearTare	BOOL	Set = 1 to clear the current tare value.
MB_Value	Real	This should always be set to the MB1.Measuring_Value of the SPC. This will provide weight data for the AOI
MB_Response	INT	This should always be set to MB1.Response value of the SPC. Once a cyclic command is successfully executed, MB_Response = MB_Command. The AOI uses this information to detect if a command has been executed successfully or if an error has occurred.
MB_Device_Status	INT	This should always be set to MB1.Device_Status of the SPC. The device status is read in and then split out into individual status bits as outputs of the AOI.
Output Parameters	Data Type	Description
DataOK	BOOL	This bit gets set to 0 when the device is still operational but the value being reported cannot be guaranteed to be valid. The following conditions cause the Data Okay bit to be set to 0: • Device is powering up • Device is in setup mode • Device is in test mode • Over capacity condition occurs

		- When the A/D converter is at its limit • Under capacity condition occurs When the A/D converter is at its limit
Motion	Bool	Motion bit is high when the weight value is not stable based on the stability criteria. ZeroStable and TareStable commands will not complete while the Motion bit is high.
NetMode	BOOL	NetMode = 1 after a tare command has been executed. Just because NetMode = 1, does not mean net weight is being reported by the SPC. Net weight must be requested by the PLC (cyclic command 3).
Alarm	BOOL	Bit will go high when alarm conditions are present. Bit will automatically go low when no alarm conditions are present. See SAI manual for more information on what causes the Alarm to go high.
Center_of_Zero	BOOL	Bit will go high when criteria is met for weight value to be at center of zero.
MB_Command	INT	This should always be set to MB1.Command tag of the SPC. Value of the last cyclic command sent to the SPC. Once successfully executed, MB_Response = MB_Command.
RealTimeWeight	Real (32 bits)	Current weight of the sensor. This value is updated constantly while the AOI is enabled.
StableWeight	Real (32 bits)	Latest stable weight reading of the sensor. This value does not update whenever the Motion bit is high.
Done	BOOL	Will be latched high when zero or tare command has successfully completed. When a new zero or tare command begins, bit will be unlatched until command completes successfully
Error	BOOL	Will be latched high when zero or tare command fails to complete. When a new zero or tare command begins, bit will be unlatched until a command fails to complete.

3.2. Fast Weighing Channel

The main weighing channel for SPC transmits weight updates at a rate of 92 Hz and includes filters optimized for gross-tare weighing.

The fast weighing channel for SPC transmits weight update rates at a rate of 366 Hz. This fast weighing channel is tailored to address the needs of filling applications where precision is required.

This increased weight update rate enables the user to make control decisions much more quickly than in previous versions. The two channel concept and an 8-block Standard Automation Interface (SAI) format allows the user to simultaneously observe weight values at 92Hz and at 366Hz. This allows separate filter settings on the Main Weight Channel (adaptive or linear filter) and the Fast Weight Update Channel (linear filter). With this functionality, the Fast Weight Update Channel is optimized on speed, and the Main Weight Channel can be tuned for more stability.

Figure 3-5: SAI_APW_FastWeightChannel AOI

Input Parameters	Data Type	Description
MB_Measuring_Value	Real	This should always be set to the MB2.Measuring_Value of the SPC. This will provide weight data for the AOI
MB_Response	INT	This should always be set to MB2.Response value of the SPC. Once a cyclic command is successfully executed, MB_Response = MB_Command. The AOI uses this information to detect if a command has been executed successfully or if an error has occurred.
Output Parameters	Data Type	Description
MB_Command	INT	This should always be set to MB2.Command tag of the SPC. Value of the last cyclic command sent to the SPC. Once successfully executed, MB_Response = MB_Command.
Fast_Weight	Real (32 bits)	Current weight from the fast weighing channel of the sensor. This value is updated constantly while the AOI is enabled.

3.3. Communication Heart Beat Monitoring

Monitoring communication between the controller and SPC. If Alive bit is set, cyclic communications between the controller and weight sensor are active.

Figure 3-6: SAI_CheckAlive AOI

Input Parameters	Data Type	Description
HeartBeat	BOOL	This should always be set to MB1.Heart_Beat bit of SPC. This bit will pulse on and off each second if cyclic communications between the SPC and the controller are established
Output Parameters	Data Type	Description
Alive	BOOL	This bit = 1 if cyclic communications are established between the SPC and the controller.

3.4. External Adjustment

An external weight adjustment procedure can be performed via this AOI. First, write the weight value of the test weight to be used in External_Weight. Set Start = 1 when ready to begin adjustment.

LoadWeight will go high when the SPC is prepared for the adjustment weight to be placed. Place the adjustment weight on the sensor that matches the value in CurrentWeight. Unload_Weight will go high to signal when the test weight should be removed. The adjustment weight may need to be added and removed from the sensor multiple times before the adjustment is complete.

If the Done or Error flag is set, the adjustment is complete.
The adjustment process can be cancelled any time by setting Cancel = 1.
The DataOK bit is cleared during the calibration process.

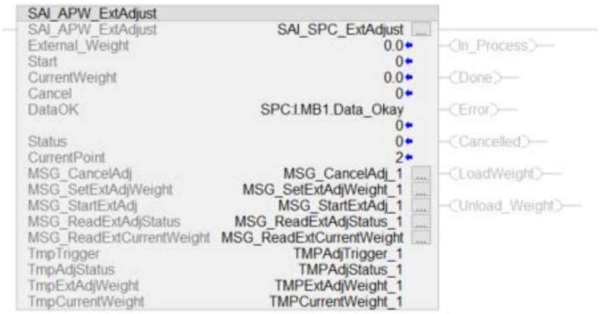

Figure 3-7: SAI_APW_ExtAdjust AOI

Input Parameters	Data Type	Description
External_Weight	Real	Set = weight value of test weight that will be placed on the sensor for the adjustment process
Start	BOOL	Set = 1 once adjustment weight value is configured to begin process.
DataOK	BOOL	Set = MB1.Data_Okay value of SPC
Cancel	BOOL	Set = 1 to cancel adjustment process
Output Parameters	**Data Type**	**Description**
CurrentWeight	REAL	The value of the test weight that should be placed on the scale
In_Process	BOOL	Goes high while the external adjustment is in process.

Done	BOOL	Latched high when adjustment completes successfully. Unlatched when another span adjustment begins.
Error	BOOL	Latched high if an error occurred and adjustment could not complete. Unlatched when another span adjustment begins. Check the errors of the messages for this AOI to troubleshoot
Cancelled	BOOL	Latched high if the adjustment process was cancelled. Unlatched when another span adjustment begins.
LoadWeight	BOOL	Set high when user needs to load test weight corresponding to the value of CurrentWeight output.
Unload_Weight	BOOL	Set high when user needs to remove test weight from the weight sensor
Status	INT	Value generally only needed for troubleshooting purposes. 0 = span adjustment has not begun or has completed. 2047 = span adjustment in process. 2045 = Ready for next adjustment point.
CurrentPoint	DINT	Value = current adjustment point being calibrated

In/Out Parameters	Data Type	Description
MSG_CancelAdj	Message	**Message Type:** CIP Generic **Service Type:** Set Attribute Single **Class:** 410 (Hex) **Instance:** 1 **Attribute:** 4 (Hex) **Source Element:** TmpAdjTrigger_1 **Source Length:** 1 (Bytes) **Communication -> Path:** Browse for the appropriate SPC
MSG_SetExtAdjWeight	Message	**Message Type:** CIP Generic **Service Type:** Set Attribute Single **Class:** 410 (Hex) **Instance: 1** **Attribute:** 9 (Hex) **Source Element:** TMPExtAdjWeight_1 **Source Length:** 4 (Bytes) **Communication -> Path:** Browse for the appropriate SPC
MSG_StartExtAdj	Message	**Message Type:** CIP Generic **Service Type:** Set Attribute Single **Class:** 410 (Hex) **Instance:** 1

		Attribute: 2 (Hex) **Source Element:** TMPAdjTrigger_1 **Source Length:** 1 (Bytes) **Communication -> Path:** Browse for the appropriate SPC
MSG_ReadAdjStatus	Message	**Message Type:** CIP Generic **Service Type:** Get Attribute Single **Class:** 410 (Hex) **Instance:** 1 **Attribute:** 7 (Hex) **Destination Element:** TmpAdjStatus_1 **Communication -> Path:** Browse for the appropriate SPC
MSG_ReadExtCurrentWeight	Message	**Message Type:** CIP Generic **Service Type:** Get Attribute Single **Class:** 410 (Hex) **Instance:** 1 **Attribute:** 11 (Hex) **Destination Element:** TmpCurrentWeight_1 **Communication -> Path:** Browse for the appropriate SPC
TmpTrigger	SINT	Temporary value for use with AOI message
TmpAdjStatus	INT	Temporary value for use with AOI message
TmpExtAdjWeight	REAL	Temporary value for use with AOI message
TmpCurrentWeight	REAL	Temporary value for use with AOI message

4. Steps to Add New SPCs

Because EtherNet/IP uses IP addresses to distinguish between devices, when multiple SPCs are networked, the default IP address of each sensor must first be modified.

Each SPC must have a different IP address.

1. Connect SPC to via APW-Link software. Navigate to the communication tab and modify the Device Address

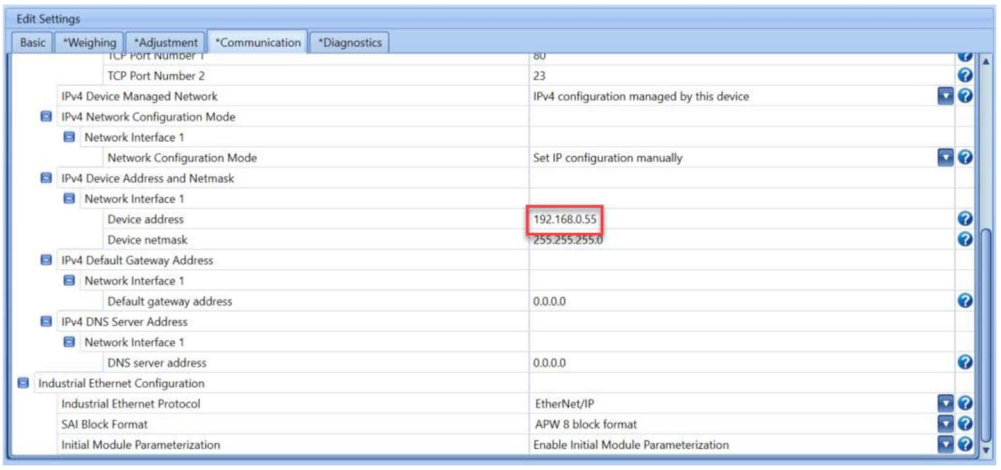

Figure 4-1: APW-Link Communication Menu

2. Add an SPC to **I / O Configuration-> Ethernet** in Studio5000.

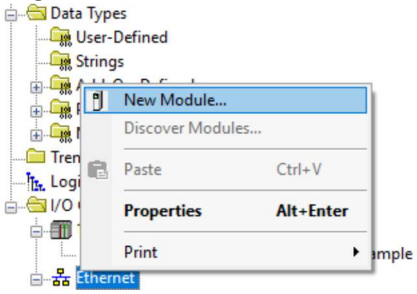

Figure 4-2: Add a device

3. Configure the name and IP address. Each device must have a unique name and IP address. Once the device is configured, click **Change**.

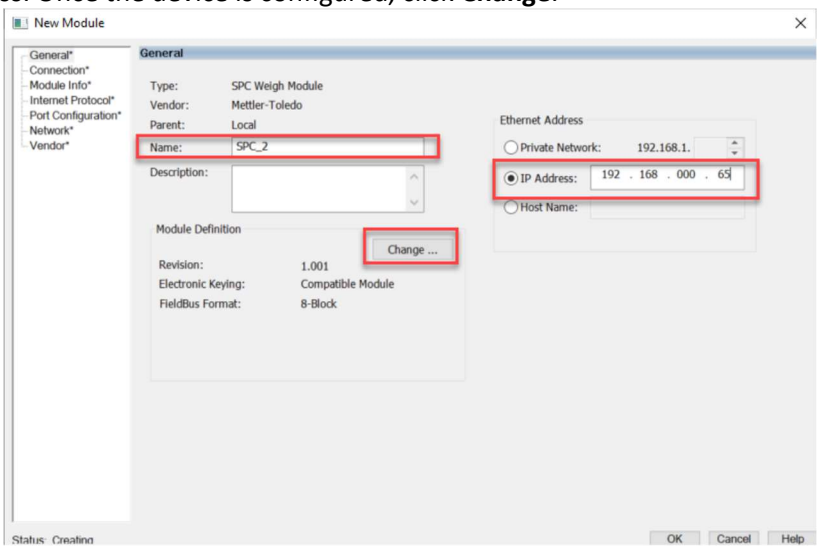

Figure 4-3: Configure name and IP address

4. Select **I/O 8 Block Format** to make the sample code function with minimal changes. Select 2 Block if it is not necessary to receive data from the fast weighing channel.

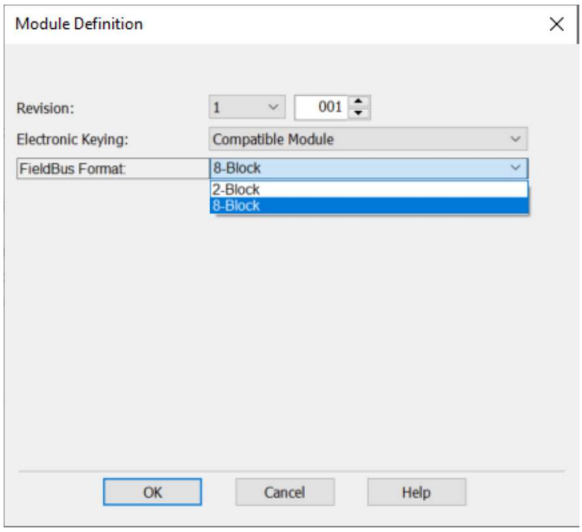

Figure 4-4: Module Definition Configuration

5. Copy the controller tags relating to the sample code and paste in the same location in order to create a duplicate set of tags. Please note that since all tags end with "_1", Studio 5000 will create duplicates that all end with "_2" instead.

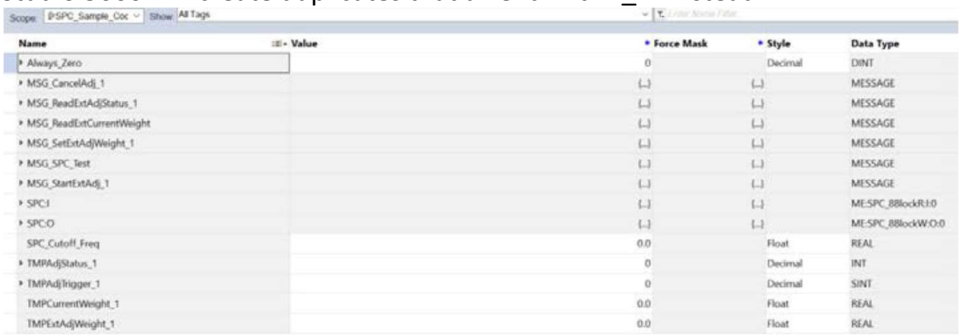

Figure 4-5: Copy/Paste Tags to Create Duplicates

6. Copy and paste the Add-On Instructions and configure the instance name along with the input and output parameters. Each device must correspond to a unique instance of the AOI. As shown in the figure below, both devices call the AOI SAI_CheckAlive, but the corresponding instances are SAI_CheckAlive and SAI_CheckAlive_1. Notice that the Heartbeat parameter is also configured with different devices for these two instances. Refer to section 2, **Add-On Instructions (AOI)**, for information on configuring parameters for a particular AOI. Make sure that all tags for the second device for instance now end in "_2" as opposed to "_1" for the first device.

Figure 4-6: Two instances of the SAI_CheckAlive AOI for two SPCs

7. Repeat steps 1 to 6 until all devices are configured.

5. Frequently Asked Questions

Q: How do I access the parameters in the AOI variables within my PLC program?

A: You can use the format "instance_name.parameter" to access parameters in your PLC program. For example, if we create an instance of the SAI_CheckAlive AOI and name the instance "SPC_Comm", we can monitor the alive bit by looking at "SPC_Comm.Alive"

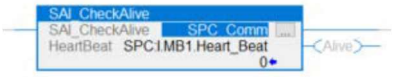

Figure 5-1: SAI_CheckAlive AOI with different instance name

Q: Does my AOI instance always have to match the name of the AOI?

A: No. The AOI instance can be named anything as long as the name is unique. They must be unique so that if we are using multiple of the same AOI, we can distinguish between them in the code. See Figure 4-1 for an example of an AOI instance name that does not match the AOI name but is still valid.

Q: An AOI is very close to what I want to do in my PLC logic, but I need to make a few changes. How can I do that?

A: If it necessary to view or modify the logic of an AOI, simply use the Controller Organizer view in Studio 5000. Navigate to Add-On Instructions, expand the AOI you are interested in viewing, and double-click **Logic**. The Organizer view will show the ladder logic used in the AOI, and the logic can be changed as necessary for your particular application.

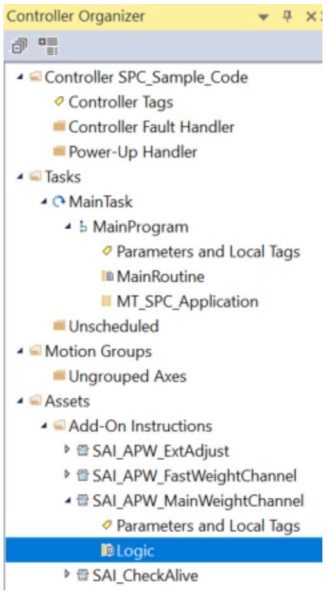

Figure 5-2: Example of AOI ladder logic