

DIADesigner
Software User Manual

www.deltaww.com

Smarter. Greener. Together.

Copyright Notice

©Delta Electronics, Inc. All rights reserved. All information contained in this user manual is the exclusive property of Delta Electronics Inc. (hereinafter referred to as "Delta") and is protected by copyright law and all other laws. Delta retains the exclusive rights of this user manual in accordance with the copyright law and all other laws. No parts in this manual can be reproduced, transmitted, transcribed, translated or used in any other ways without the prior consent of Delta.

Limitation of Liability

The contents of this user manual are only for the use of the DIAStudio manufactured by Delta. Except as defined in special mandatory laws, Delta provides this user manual “as is” and does not offer any kind of warranty through this user manual for using the product, either express or implied, including but not limited to the following:

(i) this product will meet user needs or expectations;
(ii) the information contained in the product is current and correct;
(iii) the product does not infringe any rights of any other person. Users shall bear users own risk to use this product.

In no event shall Delta, its subsidiaries, affiliates, managers, employees, agents, partners and licensors be liable for any direct, indirect, incidental, special, derivative or consequential damages (including but not limited to the damages for loss of profits, goodwill, use or other intangible losses) unless the laws contain special mandatory provisions to the contrary.
Delta reserves the right to make changes to the user manual and the products described in the user manual without prior notice and afterwards

Related Documents

Document Name	Document ID
Software Download and Installation Manual	DIAS-Manual-0005-EN
DIAScreen Software User Manual	DIAS-Manual-0004-EN

Revision History

Version	Issue Date	Revision
1.0	12/08/2020	The first version release.
1.1.0	05/20/2021	Section 2.5: Added Example Project section. Section 4.3.1: Updated File Menu. Section 4.3.2: Added printing function. Section 6.1.1.12: Added Set AS00SCM Module section. Section 6.1.1.13: Added Set AS02LC Module section. Section 8.2.2.2: Added the contents of register resource allocation settings. Section 16.8.2: Added Drive Restore Wizard section. Section 16.8.3: Added Backup & Restore section. Section 17.3: Added Print section. Section 17.4: Added Download the Project with Delta HMI via USB or SD card section.
1.1.2	12/30/2021	Section 2.1: Updated Supported Devices section. Section 4.4.3: Update device changes instructions. Section 6.1.1.1: Added the Change Firmware version. Section 6.1.1.8: Added mapping location to force lock content. Section 14.4.1.7: Added the content of each coordinate system of the oscilloscope. Section 14.4.1.8: Added the content of oscilloscope marking data points. Section 14.4.2.6: Added the content of each coordinate system of the recorder. Section 14.4.2.7: Added recorder to mark data point content.
1.2.0	07/27/2022	Section 4.2.2.4: Updated the project migrate instructions and added the notes. Section 4.4: Updated the operating instructions for new devices. Section 5.3.9: Added Inverter with Communication Card Setting section. Section 6.2: Updated parameter editor description. Section 6.3: Added Hardware Configuration Online Mode section. Section 7.2: Updated the description of the data exchange table and added the description of the SCM module data exchange table.

Version	Issue Date	Revision
		Section 7.3: Added Network View and Data Exchange Table Online Mode section. Section 9.1.4: Added Edit Library Document section. Section 9.1.5: Added Library Protection section. Section 9.1.6: Added Library Export and Import section. Section 10.1.8: Downgraded the original Section 10.2 to Section 10.1.8. Section 17.6: Added Password Management section. Section 19.4: Added Common Hot Keys section. Section 19.5: Added Data Exchange Table Online Mode Error Code Table section. Section 19.6: Added AS Series Unsupported Functions section.
1.2.1	10/28/2022	Section 2.3: Updated the operating system requirement description. Section 5.2: Added the descriptions for supporting setting the connection timeout. Section 8.2.1.2: Added the descriptions of the folder management for POU. Section 8.2.1.2, 9.1.6.3: Added support for a function block to import .fbu format. Section 8.2.1.2: Added change the contact and coil type by keyboard. Added function block auto generate variable. Section 8.2.2.2: Added button of auto allocate D/M/T/C address in Register Resource Allocation Setting. Section 8.2.2.3: Added double-click to add variable. Updated Figure data type selection for function block. Added the description of select all function for variable table. Section 8.3.3: Added 1 or 0 to express True or False in ST language. Section 8.4.2: Updated Figure, the percentage of execution code. Section 9.1.1: Added the global variable, user defined data type and folder management in library project. Section 9.2.4: Deleted the title of method 1 & method 2 Section 13.1.2: Updated the description of upload manager to select device to upload. Section 14.2.2: Added double-click to add target in monitor table. Section 16.1.1: Added double-click to add target in register edit.

Version	Issue Date	Revision
		Section 16.4.1: Added import/export function for Register Edit. Section 16.10: Added LRC/CRC Generator section. Section 17.1: Added current project and current device option in Find/Replace function. Section 19.7: Added Device and Software Version Compatible Table section.
1.2.2	02/17/2023	Section 2.5: Added Example Project 4. Section 4.2.6.1: Added Device List section. Section 7.1.2: Added the descriptions of delete connection. Section 7.2.1: Added toolbar buttons of data exchange table. Section 7.2.4: Updated the descriptions of changing the variable name. Section 8.2.2.3: Changed clear all variable address to clear all variable address and clear selected variable address. Section 8.2.2.3: Added the descriptions of the variable filter function. Section 8.3.1.2: Added the descriptions of the network folding and expanding. Section 14.4.1.3, 14.4.2.3, 14.4.4.3: Updated the descriptions of the toolbar. Section 17.1.1: Added find the variables from the program. Section 17.2.1: Added the descriptions of the bookmark function. Section 17.5.3: Added the descriptions of the programming function in the Option. Updated Section 19.7: Added Inverter EB3000 series.
1.2.3	04/13/2023	Section 2.1: Updated the supported devices. Section 3.1: Added the descriptions that DIADesigner V1.2.3 does not support Win 7. Section 14.4.4, 14.4.4.3: Updated the Oscilloscope function of the inverter. Section 19.7: Added Inverter VP3000 series.
1.3.0	11/03/2023	Section 2.1: Added support DVP-ES3/EX3/SV3/SX3 devices. Section 4.2.2: Added the descriptions of DVP project migrate. Section 4.4.3: Added Device Update Report function.

Version	Issue Date	Revision
		Section 5.3.4.4: Added the descriptions of DirectLink (USB) connection. Section 5.3.4.6: Added the descriptions of DirectLink (Ethernet) connection. Section 5.4: Added One Wire Scenario – Forwarding Function and Routing Setting section. Section 6.1.1.14 ~ 6.1.1.19: Added the descriptions of DVP-ES3/EX3/SV3/SX3 difference. Section 8.2.2: Added AS/DVP variable specification differences. Section 8.3.2.2: Added SFC element function description. Section 8.3.5: Added CFC - Continuous Function Chart. Section 10.1.3: Added the description of AS/DVP axis specification differences. Section 10.1.8.1 ~ 10.1.8.4: Updated motion mode descriptions. Section 16.11: Added Memory Card Tool – CARD Utility section. Section 17.6.1: Add AS/DVP password specification differences. Section 19.7: Added DVP-ES3/EX3/SV3/SX3 series module.
1.4.0	02/23/2024	Section 2.1: Added support DVP-ES3 (TEC) device. Section 4.3.1: Update Menu Bar. Section 4.3.1.1: Added Tool – Review Error Log File section. Section 4.3.1.2: Added Tool – Review Status Log File section. Section 7.4: Added EtherCAT Topology Configuration section. Section 7.4.1: Added EtherCAT I/O Mapping section. Section 8.2.2.3: Added the description of Variable Name. Section 8.3.1.1: Updated LD Toolbar Section 8.3.1.2: Added Insert Execute Block section. Section 8.3.1.2: Updated Typing Instructions. Section 8.3.5.1: Added unsupported instructions. Section 8.3.5.3: Updated the notes for entering instructions, added the descriptions of Setting a Label. Section 8.4.3: Added Export Program to Text Format.

Version	Issue Date	Revision
		Section 16.11: Updated CARD Utility Backup and Restore table. Section 19.7: Updated Device and Software Version Compatible Table.
1.4.2	04/08/2024	Section 1.2: Revised the note. Section 2.4.1: Revised the section. Section 2.4.2: Revised the section. Section 7.1.2: Added a note. Section 7.4.2: Added EtherCAT Online Mode section.
1.5.0	05/31/2024	Section 4.2.6.2: Added a note. Section 4.3: Revised Advanced Designer section. Section 4.3.4: Revised Development Toolbar section. Section 8.3.1.2: Revised Editing the Comment on a Variable in LD Element. Section 8.3.1.2: Revised Insert Execute Block. Section 8.3.1.2: Added Network Continuous Paste. Section 8.5.4: Added the Function Block Monitoring section. Section 9.1.4.5: Added the Multi-language Creation of Library Document section. Section 9.2.4.2: Revised the Load Library Document section. Section 10.2: Added the Axis Unit Conversion section. Section 10.3: Added the E-CAM (Electronic CAM) section. Section 14.4.4: Revised subsections. Chapter 16: Revised Auxiliary Tools chapter. Section 17.5: Revised the Option section and its subsections. Section 19.7: Updated Device and Software Version Compatible Table.
1.6.0	07/31/2024	Section 4.2.2.4: Modified the note. Section 4.3.1: Revised the section. Section 4.3.5: Renamed Communication Toolbar to Shortcut Toolbar. Section 8.3.1.2: Revised Inserting a Contact and Changing a Contact Type. Section 8.4.2: Added a note. Section 10.3.2: Revised the Using E-CAM section. Section 16.2: Added a note. Section 19.6: Revised the section.

Version	Issue Date	Revision
		Section 19.7: Updated Device and Software Version Compatible Table.
1.6.1	09/06/2024	Section 6.2.1: Revised the Parameter Settings for PLC devices section. Section 8.2.1.2: Revised the Setting and Unlocking a POU Password section. Section 8.4.2.1: Added the Program Compiler Setting section. Section 8.5: Revised the Program Monitoring section. Section 14.2.2: Revised the Adding and Variables and Registers section. Section 14.2.4: Added a note.
1.7.0	10/30/2024	Section 3.1.1: Added Optional Installation section. Section 4.2.2.4: Project migrate supports DIASelector projects. Section 4.2.6.2: Added support device description files in DDF and ESI formats. Section 7.1.4: Added the Network View Context Menu section. Section 16.11.4: Added the Create auto download execution file section. Section 19.7: Updated Device and Software Version Compatible Table.
1.8.0	12/31/2024	Section 2.1: Updated description of supported devices. Section 4.2.6.2: Updated import of device repository. Section 5.4: Added HMI in One Wire scenario. Section 6.1.2.2: Added multi-drive inverter configuration section. Section 6.2.4: Added multi-drive inverter parameter setting section. Section 6.2.6: Added Upload Configuration File of Parameters section. Section 7.1.2: Updated Toolbox description. Section 7.2.5: Added data exchange table for VFD build-in PLC. Section 14.1: Added multi-drive inverter description. Section 14.2.1: Added export function for monitor table. Section 14.4.4.3: Added multi-drive inverter description. Section 17.7: Added comparison function section. Section 19.7: Updated Device & Software Version Compatible.

Version	Issue Date	Revision
1.9.0	04/25/2025	Chapter 1: Rearranged the chapter. Chapter 2: Rearranged the chapter. Section 3.2: Added entry window function. Section 4.2: Revised the section. Section 4.3.5: Updated the shortcut toolbar Section 4.3.6: Updated the Project Explorer function. Section 6.1.1.1: Updated the descriptions of the DVP series digital I/O module. Section 6.2: Added the function of modifying the parameter list for Parameter Setting. Section 7.2.3.1: Updated the content of the EIP exchange table and variable. Section 8.2.2.5: Added Symbol Configuration section. Section 8.3.1.2: Updated the descriptions of Inserting Applied Instructions and Function Blocks. Section 8.4.2.1: Revised the section. Section 10.2.2: Updated UI interfaces of axis unit conversion. Section 19.7: Updated Device and Software Version Compatible Table. Section 19.8: Added Function and Software / Firmware Compatible Table section.
1.10.0	06/18/2025	Section 2.1: Updated the system requirement. Section 4.3.6: Revised the section. Section 5.3.7: Revised the section and added a note. Section 5.3.9: Added Inverter PROFINET Direct Connection function section. Section 6.1.1: Revised the section. Section 8.4.2.1: Revised the section. Section 14.2.1: Revised the section. Section 19.7: Updated Device and Software Version Compatible Table.
1.11.0	10/30/2025	Section 4.3.4: Added Communication Setting Quick Connection function. Section 6.1.1.2: Added Occupied D Register Range function. Section 7.2.5: Added the descriptions of DVPEN02-SL Modbus TCP data exchange. Section 8.3.1.2: Added the descriptions of toolbox instruction to Inserting Applied Instructions and Function Blocks.

Version	Issue Date	Revision
		Section 8.3.1.2: Added Block Series and Parallel. Section 14.4.4.3: Updated Monitor Setting section. Section 19.7: Updated Device and Software Version Compatible Table.
1.11.1	01/30/2026	Section 14.4.4: Added a note.
1.13.0	03/31/2026	Section 4.2.3.1: Supported Polski language. Section 6.1.1.19: Updated the description Section 6.2.4, 6.2.5: Integrated the Actual Value description in the inverter Parameter Editor. Section 8.3.1.2: Added Block Copy and Paste section, deleted Variable Mode and Address Mode section. Section 8.3.3.1: Added Indent functions to the toolbar for ST POU. Section 16.11.1: Added the project backup function of the CARD Utility. Section 17.5.3: Supported the Enable online monitor mode for variable function in the Option page. Section 19.7: Updated Device & Software Version Compatible.

Table of Contents

Chapter 1: DIADesigner Overview 1
1.1 DIADesigner Feartures 2
Chapter 2: Install, Uninstall, and Update DIADesigner 4
2.1 System Requirement 5
2.2 Installation and Update 5
2.2.1 Optional Installation 6
2.3 Launch DIADesigner and Entry Window 7
2.4 Uninstall DIADesigner 8
2.5 Update DIADesigner 9
Chapter 3: Get Started 10
3.1 Supported Devices 11
3.2 Device and Function Guide 11
3.3 Quick Start 11
3.3.1 Scenario 1 11
3.3.2 Scenario 2 13
3.4 Example Project 13
Chapter 4: User Interface 15
4.1 UI Introduction 16
4.2 Basic Designer 17
4.2.1 Project 18
4.2.2 Wizards 22
4.2.3 Setting 26
4.2.4 Guide 29
4.3 Advanced Designer 33
4.3.1 Menu Bar 35
4.3.2 Standard Toolbar 39
4.3.3 Editor Toolbar 40

4.3.4 Development Toolbar 40
4.3.5 Shortcut Toolbar 41
4.3.6 Project Explorer 42
4.4 Creating Project and Device 45
4.4.1 Add Device 46
4.4.2 Device Name 49
4.4.3 Update Device 50
Chapter 5: Communication Setting 53
5.1 Communication via COMMGR 54
5.2 Communication without COMMGR 55
5.3 COMMGR Overview 57
5.3.1 Launch COMMGR 58
5.3.2 Main Screen 58
5.3.3 Driver Setting 62
5.3.4 Driver Start/Stop 75
5.3.5 Driver Search 76
5.3.6 Delete and Modify Drivers 79
5.3.7 Communication Setting of Inverter with Built-in PLC 80
5.3.8 Communication Setting of Inverter with Communication Card 82
5.3.9 Inverter PROFINET Direct Connection Function 86
5.4. One Wire Scenario – Forwarding Function and Routing Setting 87
Chapter 6: Setting Device (Hardware Configuration) 92
6.1 Hardware Configuration 93
6.1.1 Creating Controller Configuration 93
6.1.2 Creating Inverter Configuration 125
6.2 Parameter Setting 139
6.2.1 Parameter Setting for Controller Devices 140
6.2.2 Parameter Setting for Temperature Controller 142
6.2.3 Parameter Setting for Servo Drive 144

6.2.4 Parameter Setting for Inverter 147
6.2.5 Perform Parameter Setting After Servo Drives / Inverters Upload Parameters 149
6.2.6 Upload Configuration File of Parameters 152
6.3 Hardware Configuration Online Mode 153
6.3.1 Online Mode Status Light 155
Chapter 7: Setting Network (Network Configuration) 156
7.1 Network View 157
7.1.1 Network View Description 157
7.1.2 Creating Network View in Advanced Designer 158
7.1.3 Network View Toolbar 163
7.1.4 Network View Context Menu 166
7.1.5 Network Parameter Setting 167
7.2 Data Exchange Table 167
7.2.1 Exchanging Data through Data Exchange Table 169
7.2.2 Setting Up Data Exchange Table 172
7.2.3 EtherNet/IP Data Exchange Table 174
7.2.4 CANopen Data Exchange Table 175
7.2.5 Modbus / Modbus TCP Data Exchange Table 176
7.2.6 SCM Module Data Exchange Table Setting 178
7.3 Network View and Data Exchange Table Online Mode 180
7.3.1 Network View Online Mode 181
7.3.2 Data Exchange Table Online Mode 182
7.4 EtherCAT Topology Configuration 183
7.4.1 EtherCAT I/O Mapping 188
7.4.2 EtherCAT Online Mode 190
Chapter 8: Programming 191
8.1 IEC61131-3 Programming 192
8.2 POU, VAR, Task 193

8.2.1 POU 193
8.2.2 VAR 218
8.2.3 Task 260
8.2.4 Symbol Configuration 266
8.3 Programming Language Introduction (LD, SFC, ST) 268
8.3.1 LD—Ladder Diagram 268
8.3.2 SFC—Sequential Function Chart 301
8.3.3 ST—Structured Text 331
8.3.4 C Language Editor (C Language) 362
8.3.5 CFC—Continuous Function Chart 380
8.4 Compile and Check 403
8.4.1 Check 403
8.4.2 Compile 404
8.4.3 Export Programs to Text 408
8.5 Program Monitoring 408
8.5.1 LD Program Monitoring 409
8.5.2 ST Program Monitoring 409
8.5.3 SFC Program Monitoring 410
8.5.4 Function Block Monitoring 411
8.6 Online Edit 411
8.6.1 POU Online Edit 412
8.7 Debug Mode 423
8.7.1 Enable Debug Mode 424
8.7.2 Break Point 424
8.7.3 Continue Run Execution 427
8.7.4 Single-Step Execution 428
Chapter 9: Library Management Tools 429
9.1 Library Project 430
9.1.1 Create Library Project 430

9.1.2 Project Information 432
9.1.3 Repository Project 432
9.1.4 Edit Library Document 433
9.1.5 Library Protection 438
9.1.6 Library Export and Import 440
9.2 Library Manager 444
9.2.1 Add Library 446
9.2.2 Delete Library 446
9.2.3 Library Repository 446
9.2.4 Library Usage 447
Chapter 10: Motion Tools 452
10.1 Position Plan Table 453
10.1.1 About Position Plan Table 453
10.1.2 Setting Position Plan Table 453
10.1.3 Axis Setting 454
10.1.4 Password 456
10.1.5 Position Table 458
10.1.6 Position Plan Table Simulation 460
10.1.7 Upload and Download Position Plan Table 463
10.1.8 Control Mode 464
10.2 Axis Unit Conversion 474
10.2.1 About Axis Unit Conversion 474
10.2.2 Axis Unit Conversion Setting 475
10.3 E-CAM (Electronic CAM) 478
10.3.1 About E-CAM 478
10.3.2 Using E-CAM 478
Chapter 11: HMI Interaction 482
11.1 Tag Sharing to DIAScreen 483
Chapter 12: Simulation 487

12.1 PLC Simulation 488
Chapter 13: Download / Upload and Online / Offline Mode 489
13.1 Batch Download and Upload 490
13.1.1 Download 490
13.1.2 Upload 492
13.1.3 Download / Upload Components 494
13.2 Online / Offline Mode 494
Chapter 14: Commission and Diagnostic 496
14.1 Device Information 497
14.2 Monitor Table 498
14.2.1 Creating a Monitor Table 499
14.2.2 Adding Variable and Registers 502
14.2.3 Monitor Table Context Menu 504
14.2.4 Using Monitor Table 505
14.3 Tuning Tool 507
14.3.1 Calculate Inertia Ratio 507
14.3.2 Set Up the Bandwidth 510
14.4 Trace and Oscilloscope 511
14.4.1 Data Tracer 511
14.4.2 Data Logger 520
14.4.3 Servo Device Oscilloscope 528
14.4.4 Inverter Device Trace / Oscilloscope 534
14.5 Communication Test Tools 540
Chapter 15: Temperature Control Tool 546
15.1 Monitor 547
15.1.1 Turn on the Temperature Monitor Function 547
15.1.2 Temperature Monitor Dialog 548
15.1.3 Monitor Setup 549
15.2 Program Planning 554

15.2.1 Program and Step Editing 556
15.2.2 Status Display Area 557
15.3 Station Number Automatic Synchronization 558
15.3.1 Activate Station Number Automatic Synchronization Function 558
15.3.2 Deactivate Station Number Automatic Synchronization Function 559
Chapter 16: Auxiliary Tools 560
16.1 Register Comment 561
16.1.1 Adding Items in Register Comment Dialog 563
16.1.2 Deleting Items in Register Comment Dialog 564
16.1.3 Deleting Register Comments 564
16.1.4 Deleting Comments for Selected Register Type 565
16.1.5 Deleting All Comments 565
16.2 Register Usage 565
16.3 Programming Memory Resources 568
16.4 Register Edit 569
16.4.1 Register Edit Toolbar 570
16.4.2 Data Format 571
16.4.3 Data Length 572
16.4.4 Address Edit – Write 573
16.4.5 Address Edit – Read 574
16.4.6 Address Edit – Clear all Values 575
16.4.7 Address Edit – Move to 576
16.5 Step Position 577
16.6 Set RTC 578
16.6.1 Updating Computer Time to PLC Time 579
16.6.2 Updating Custom Time to PLC Time 579
16.7 Format PLC Program Memory 580
16.7.1 Format Program Memory 581
16.7.2 Reset PLC Memory (Factory Setting) 582

16.8 Backup & Restore 582
16.8.1 Parameter Backup & Restore 582
16.8.2 Drive Restore Wizard 585
16.8.3 Backup & Restore 590
16.9 Retain Variable Initialization 594
16.10 LRC/CRC Generator 595
16.11 Memory Card Tool – CARD Utility 596
16.11.1 Backup 598
16.11.2 Restoration 604
16.11.3 Command-line Instruction Execution 609
16.11.4 Create Auto Download Executable File 615
Chapter 17: Other Functions 617
17.1 Find and Replace 618
17.1.1 Find – POU 618
17.1.2 Replace – POU 620
17.1.3 Batch Replacement Register 622
17.2 Bookmark 625
17.2.1 Bookmark in Ladder Diagram (LD) POU 626
17.2.2 Bookmark in Structured Text (ST) POU 631
17.2.3 Bookmark in C Language 636
17.3 Print 640
17.3.1 Print Form Page 641
17.3.2 Print Graphics Page 642
17.4 Download the Project with Delta HMI via USB or SD Card 644
17.5 Option 645
17.5.1 Project Setting 646
17.5.2 Environment Settings 648
17.5.3 Programming Settings 649
17.6 Password Management 652

17.6.1 DIADesigner Project 652
17.6.2 Program ID and PLC ID 653
17.6.3 POU Password 656
17.6.4 PLC Password 656
17.7 Comparison Function 658
17.7.1 Comparison Function Rules 658
17.7.2 Project Comparison 659
17.7.3 Device Comparison 661
Chapter 18: Troubleshooting 663
18.1 Error Log 664
18.1.1 Error Information 664
18.1.2 PLC Download Log 665
18.1.3 Status Change Log 666
18.2 Error Checklist 667
Chapter 19: Appendix 669
19.1 Installing USB Driver for AS Series CPU Module 670
19.1.1 Installing USB Driver in Windows 7 670
19.1.2 Installing USB Driver in Windows 8.1 674
19.1.3 Installing USB Driver in Windows 10 677
19.2 Important Points related to CPU Modules 679
19.2.1 Addresses in AS Series CPU Modules 679
19.2.2 Address Resources in AS Series CPU Modules 683
19.3 Servo Setting 683
19.4 Common Hot Keys 686
19.5 Data Exchange Table Online Mode Error Code Table 687
19.5.1 EtherNet/IP Error Code 687
19.5.2 Modbus Error Code 690
19.5.3 Modbus TCP Error Code 691
19.6 Project Migrate and Unsupported Functions 693

19.7 Device and Software Version Compatible Table 694
19.8 Function and Software / Firmware Version Compatible Table 698

Chapter 1: DIADesigner Overview

DIADesigner is a member of the DIAStudio integrated software suite, which develops software for Delta's new generation of automation products. It provides you with Delta's full range of Industrial Automation products for program editing, network and hardware configuration, parameter settings and other functions, to complete various tasks with a consistent user interface and operating experience. In addition, it also provides many auxiliary tools and multi-language systems, providing you with a convenient and efficient development environment.

1.1 DIADesigner Feartures

The DIADesigner features are as follows.

- **Integration Function**
  - DIADesigner integrates the software of all Delta products so that you do not need to switch between multiple software operations. One project can support multiple Delta products for systematic project creation.
  - Provides basic and advanced design interfaces for a consistent, friendly, and intuitive experience.
  - Legacy ISPSoft software projects are compatible with DIADesigner and can be reused.
- **Program Editing Function**
  - Adopts the international standard IEC 61131-3 programming architecture and has a rich built-in application instruction set.
  - Supports multiple languages, including ladder diagram (LD), sequential function diagram (SFC), structured language (ST) and C language (C).
- **Configuration Function**
  - Hardware Planning Tool: Used for configuration and parameter management of the hardware architecture of the controller.
  - Network Planning Tool: Used to manage and configure network architecture and data exchange mechanism of the controller.
- **Convenient Function**
  - Instant keyword filtering function helps you to confirm the correctness when setting commands or parameters.
  - Supports import / export functions of multiple objects, and can be exported / imported into txt, csv and other text formats for further use.

    Note: The import / export function can be compatible with the files exported by the legacy software. For example, the parameter file exported by ASDASoft can be imported into the parameter table of DIADesigner.

- Provides Online and Offline modes for different operating functions for online / offline devices.
- Provides a search and replacement function.
- Support multilingual such as Traditional Chinese, Simplified Chinese, English, and other languages.
- Project management uses a clear and easy-to-use Windows-level tree-like interface.

Chapter 2: Install, Uninstall, and Update DIADesigner

This section introduces how to install, uninstall, and update DIADesigner in DIAInstaller.

2.1 System Requirement

To install and run DIADesigner, your computer should meet the following minimum hardware and software requirements.

Item	System Requirement
Operating System	Windows 11 (64-bit)
CPU	Intel i5 2.4 GHz (min.)
Memory	Recommend to use 16 GB or above
Hard Disk Drive	15 GB free for DIADesigner
Monitor	Resolution: 1024 x 768 Pixels, 1920 x 1080 Pixels
Keyboard / Mouse	General keyboard mouse or Windows-compatible device
USB	Used in connection with the device
Ethernet	Used in connection with the device
Software	Need to install Microsoft.Net Framework 4.7.2

2.2 Installation and Update

The DIAInstaller assists you with downloading, installing, uninstalling, and updating DIAStudio software applications. You can download DIAInstaller from the website: diastudio.deltaww.com. For more information about DIAInstaller, refer to *Software Download and Installation Manual*.

Prerequisites

- DIAInstaller is already installed.

Follow these steps to install DIADesigner.

Prerequisites

- Register a DIAStudio account and click  to sign in.

To install DIADesigner

1. Go to **All Apps**.

2. Select DIADesigner.
3. In **Available Versions**, select a version.
4. Click **Install**.
5. (optional) In the installation dialog, select the **Save downloads to local folder** checkbox to save the installation files to your computer.
6. You can also select components to install, refer to Optional Installation.
7. Follow the on-screen instructions to complete the installation.

After installing DIADesigner, the **Installed Version** will show in **All Apps**. You can also select other **Available Versions** to install in the same environment.

2.2.1 Optional Installation

Select the required components to install to save installation space.

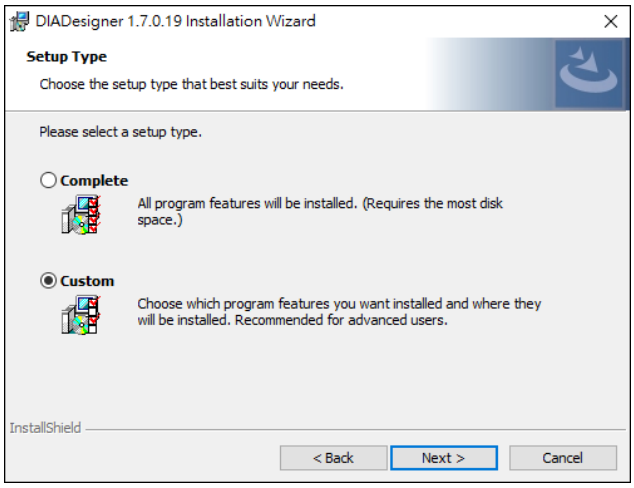
Follow these steps to perform installation.

To perform optional installation

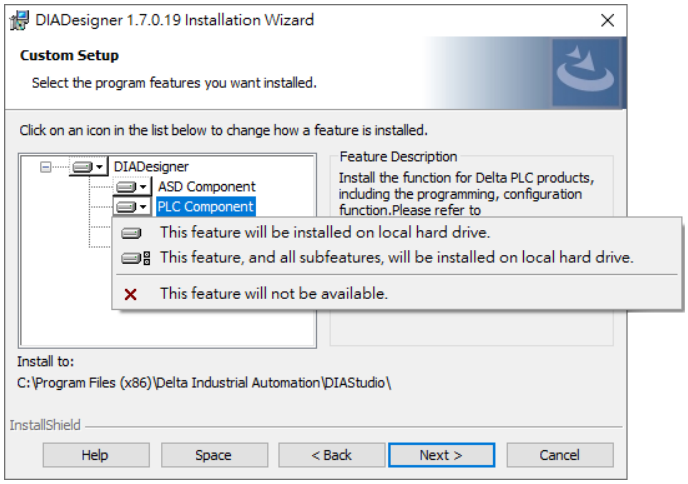
1. In the **Installation Wizard** dialog, do one of the following.
   - Select **Complete** to install all components.

     Or

   - Select **Custom** to select components to install.



2. Right-click the component that does not need to be installed and select **This feature will not be available**. The component will be no longer installed.



3. Click **Next** to continue installation.

   If you need to change the installation components, perform the installation again and reselect in the context menu in the **Custom Setup** step.

2.3 Launch DIADesigner and Entry Window

You can launch DIADesigner directly in DIAInstaller after DIADesigner is installed, or double-click the DIADesigner shortcut on the desktop to launch.

Prerequisites

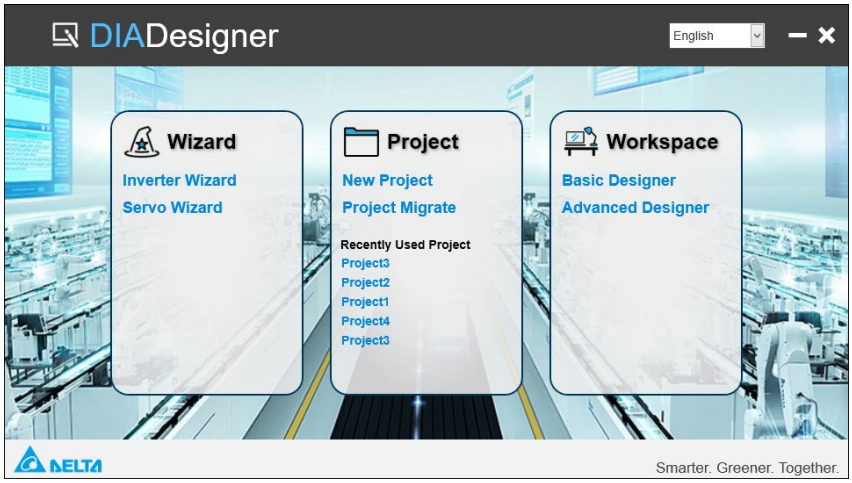
- DIADesinger of the selected version is already installed.

To launch software

1. Go to **All Apps**.
2. Select DIADesigner and then click **Launch**.



After DIADesigner is launched, the entry window will open and guide you to operate the main functions.



2.4 Uninstall DIADesigner

Follow these steps to uninstall DIADesigner that is already installed on your computer.

To uninstall DIADesigner

Do one of the following:

1. Go to **Installed**.
2. Select DIADesigner and then click **Uninstall**.
3. Follow the on-screen instructions to complete the uninstallation.
4. Restart your computer if needed. You can click **Uninstallation Summary** to view the software information and the uninstallation logs.

Or

1. Go to **Control Panel** > **Program** > **Programs and Features**.
2. In the **Program and Features** dialog, select DIADesigner and then click **Uninstall**.
3. Follow the on-screen instructions to complete the installation.
4. Restart your computer if needed. You can click **Uninstallation Summary** to view the software information and the uninstallation logs.

2.5 Update DIADesigner

If a patch or a later version is available for DIADesigner, they will show on the **Updates** pane. Follow these steps to update DIADesigner.

To update to a later version

1. Go to **Updates**.
2. In the software list, select DIADesigner.
3. In **Available Versions**, select a main version or a patch version. You can view what is new.

   Note: A later version's patch can be installed directly. For example, installing v1.1.1 does not require v1.1.0 to be installed as a necessary condition. The installation will install both v1.1.0 and v1.1.1.
4. Do one of the following:
   - Select **Update** and then follow the on-screen instructions to install.
   - Select **Download** to download the installation file to the local path and then click **Show in folder** to find the .exe file to install.

   Note: The earlier version will be uninstalled first and then install the later one.

Chapter 3: Get Started

3.1 Supported Devices

Refer to Device and Software Version Compatible Table for Delta products supported by DIADesigner. For more information, refer to the device documentation.

3.2 Device and Function Guide

DIADesigner version 1.10.0 supports the following devices and functions. Subsequent versions will continuously add devices and features.

<table>
<tr><th></th><th rowspan="2">Communication Setup</th><th rowspan="2">Network Configuration</th><th rowspan="2">Hardware Configuration</th><th rowspan="2">Parameter Setup</th><th rowspan="2">Program Edit</th><th colspan="3">Tuning Tool</th><th rowspan="2">Download</th></tr>
<tr><th>Monitoring Table</th><th>Trend & Oscilloscope</th><th>Tuning</th></tr>
<tr><td>Controller</td><td rowspan="3">CH5</td><td rowspan="3">CH7</td><td rowspan="3">CH6</td><td>CH6.2.1</td><td>CH8</td><td>CH14.2</td><td rowspan="3">CH14.4</td><td rowspan="2">N/A</td><td rowspan="3">CH13</td></tr>
<tr><td>Inverter</td><td>CH6.2.4</td><td>CH8 (Built-in PLC)</td><td>N/A</td></tr>
<tr><td>Servo Drive</td><td>CH6.2.3</td><td>N/A</td><td>N/A</td><td>CH14.3</td></tr>
</table>

3.3 Quick Start

This section uses two scenarios to describe the main operation procedure of each important software function.

3.3.1 Scenario 1

This scenario describes the operating steps for related products with Human Machine Interface (HMI), Programmable Logic Controller (PLC), Servo Drive (ASDA). You can link to the relevant sections to view function's descriptions.

- **Create a Project**
  - Launch the Software: Section 2.3
  - Create a New Project: Section 4.2.1
- **Add Devices**
  - Add Device in Project Explorer: Section 4.4.1

- **Communication Setting**
  - COMMGR Configuration: Section 5.3
  - Device Communication Setting: Section 5.1
- **Hardware Configuration Setting**
  - Module Configuration: Section 6.1
  - Parameter Setting: Section 6.2
- **Network Configuration Setting**
  - Network View Configuration: Section 7.1
  - Data Exchange Table: Section 7.2
- **Program Editing**
  - Establish a POU: Section 8.2.1
  - Establish Variables: Section 8.2.2
- **Motion Control**
  - Positioning Plan: Section 10.1
- **HMI Variable Sharing**
  - Export the Variable to DIAScreen: Chapter 11
  - Synchronize with DIAScreen Variables: Chapter 11
- **Download**
  - Batch Upload / Download: Section 13.1
- **Tuning (Commissioning)**
  - Variables and Register Monitoring Table: Section 14.2
  - Debug Mode: Section 8.7
  - Online Change: Section 8.6

3.3.2 Scenario 2

The scenario describes the operating steps for single inverter or servo drive. You can link to the relevant sections for functions' description.

- **Create a Project**
  - Inverter Wizard / Servo Wizard: Section 4.2.2.1~Section 4.2.2.2
- **Add Devices**
  - Select Device: Section 4.2.2.1~Section 4.2.2.2
- **Communication Setting**
  - Communication Setting: Section 4.2.2.1~Section 4.2.2.2
- **Hardware Configuration Setting**
  - Parameter Setting: Section 6.2
- **Tuning (Commissioning)**
  - Oscilloscope: Section 14.4.3~Section 14.4.4
  - Gain and Inertia Estimation (Applicable to servo drive): Section 14.3

3.4 Example Project

This section describes the built-in four example projects of the DIADesigner software.

- Example1_Analog_Output_Control
- Example2_CANopen_Communication
- Example3_Temperature_Control
- Example4_ Modbus TCP_Communication

The Basic Designer and Advanced Designer pages provide a sample project file opening function.

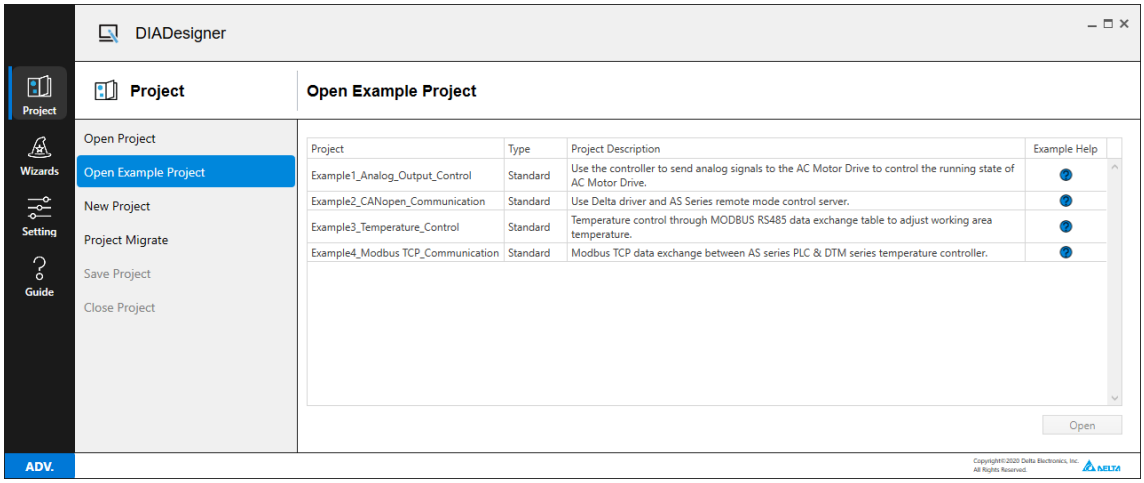
To open example project

- **Basic Designer page**

1. Click **Project** > **Open Sample Project**.
2. Do one of the following:
   - In the **Project** field, double-click the example project.
   - After selecting the example project, click **Open** at the bottom right.

     You will be directed to the Advanced Designer page and can start to work on the **Example Project**.

To understand a particular example project, click in the **Example Help** field to open the FAQ on **How to Use DIADesigner Sample Project**.



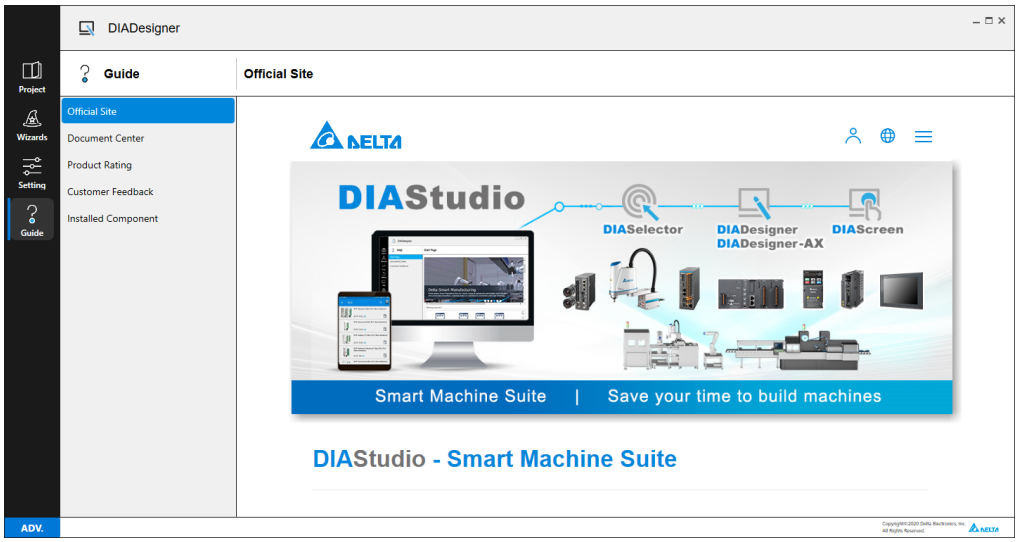
- **Advanced Designer page**

1. On the menu bar, click **File** > **Open Example Project**.
2. In the **Open** dialog, double-click the Example2_CANopen_Communication folder.
3. Select the .diade file of the project and then click **Open**.

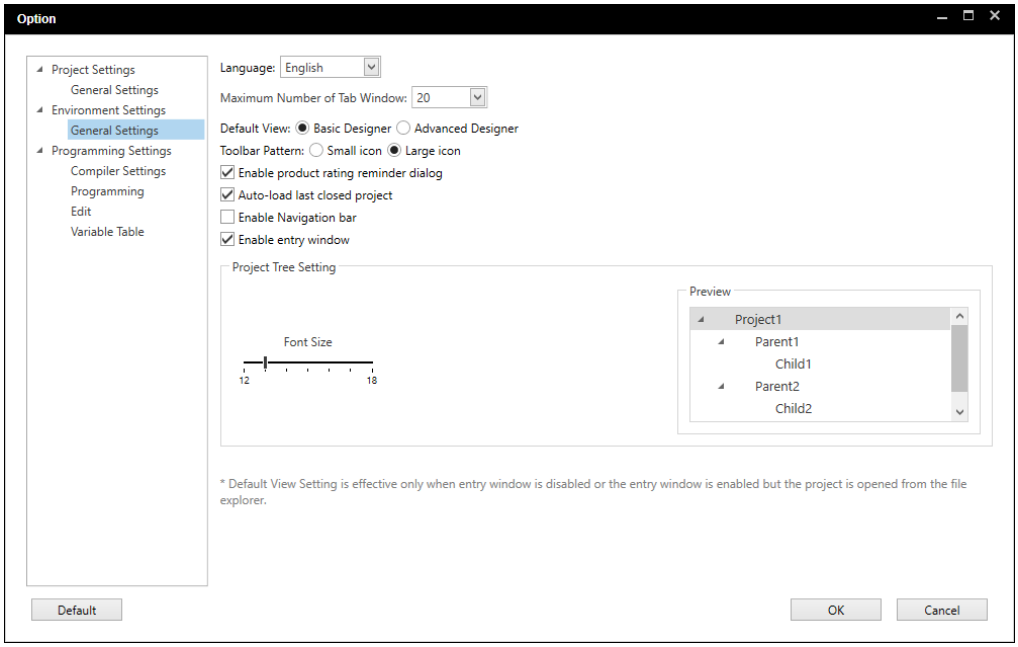
Chapter 4: User Interface

4.1 UI Introduction

DIADesigner provides basic and advanced user interfaces for different usage scenarios. Click **Basic Designer** or **Advanced Designer** on the **Workspace** of the entry window to enter the main software interface. The interface is default as Basic Designer, you can click the button at the bottom left corner to switch between Advanced Designer and Basic Designer.



In Advanced Designer, you can set the **Default View** and **Enable entry window** by clicking **Tool** > **Option** > **Environment Settings** > **General Settings** in the toolbar.

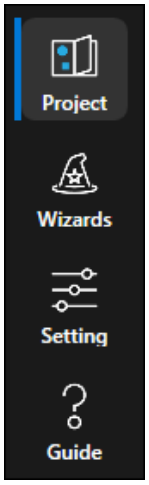


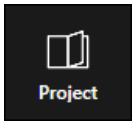
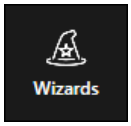
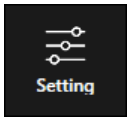
4.2 Basic Designer

The Basic Designer interface provides the following basic settings and wizard functions.

- Provides common functions such as Open Project, Open Example Project, and Project Migrate for project management.
- Provides convenient tools such as Servo Wizard and Inverter Wizard.
- Provides related link function for customer feedback and information look-up.

The following table lists the tabs in the **Basic Designer** interface and their descriptions.

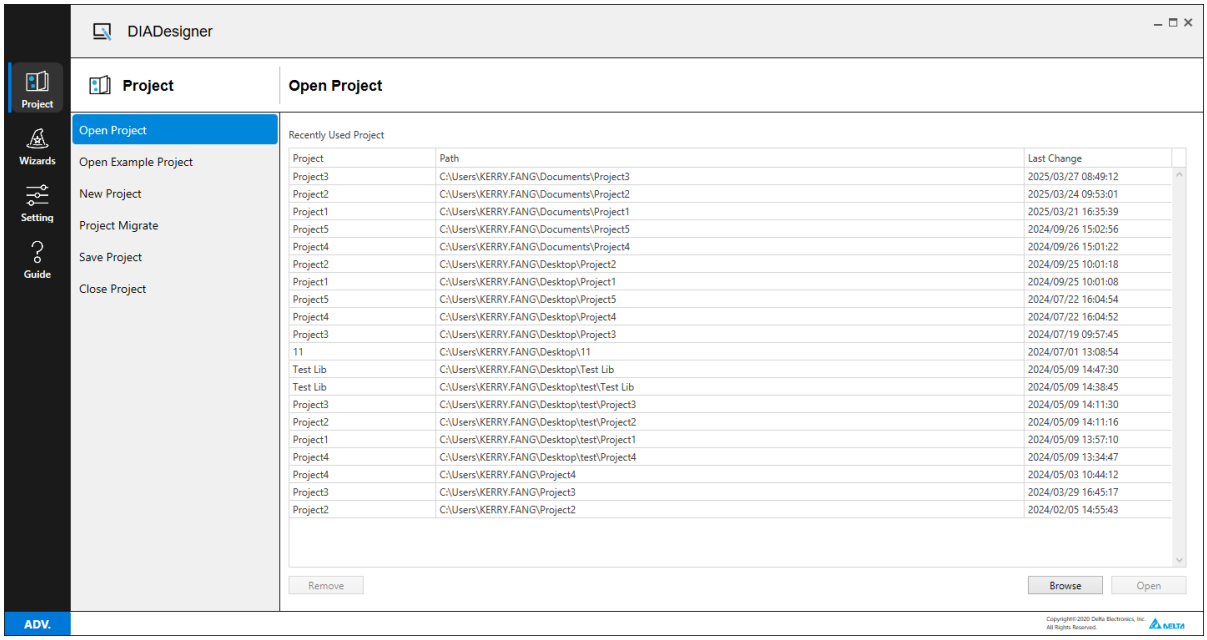


Tab	Description
	Performs project configuration and migration. • Open project • Open example project • Create new project • Project migrate (conversion of old software projects) • Save project • Close project
	Quick link for launch wizards and tools. • Inverter wizard • Servo wizard
	Set language and manage device description files. • View a list of installed devices • Install and uninstall device repository • Set language

Tab	Description
	Provides function guide wizard and guidance contact information. • View official site • Go to document center • Provide customer feedback • View current installed software component

4.2.1 Project

In the **Project** tab, you can create a project.



The **Project** tab has the following functions.

- Open Project
- Open Example Project
- New Project
- Project Migrate
- Save Project
- Close Project

4.2.1.1 Open Project

In the **Open Project** pane, you can open the previously created project.

The following table lists the functions in the **Open Project** pane and their descriptions.

Function	Description
Recently Used Project	Displays the list of recently opened projects.
Project	Displays the project name.
Path	Displays the project path. **Note**: You can enter the path manually.
Last Change	Displays the date and time of last change.
Remove	Click to remove the project from the **Recently Used Project** list, but the project files are not removed from the computer.
Browse	Click to select the project to open.
Open	Click to open the selected project from the **Recently Used Project** list.

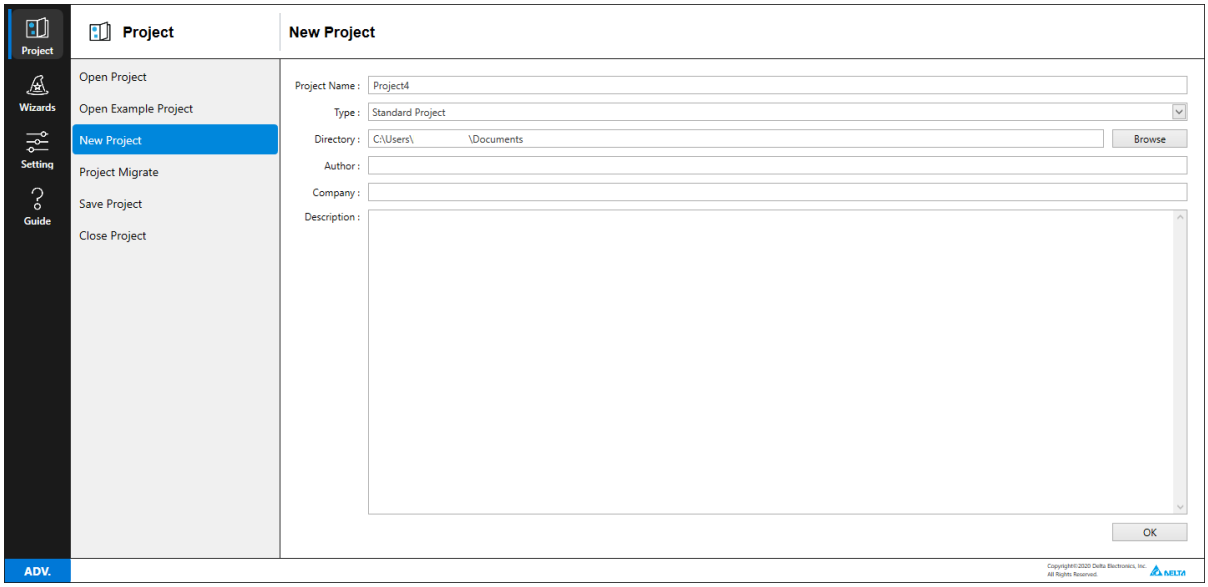
4.2.1.2 Open Example Project

In the **Open Example Project** pane, you can open an example project. For detailed introduction of opening the example project, refer to Example Project.

4.2.1.3 New Project

In the **New Project** pane, you can create a new project.

The following table lists the functions in **New Project** pane and their descriptions.

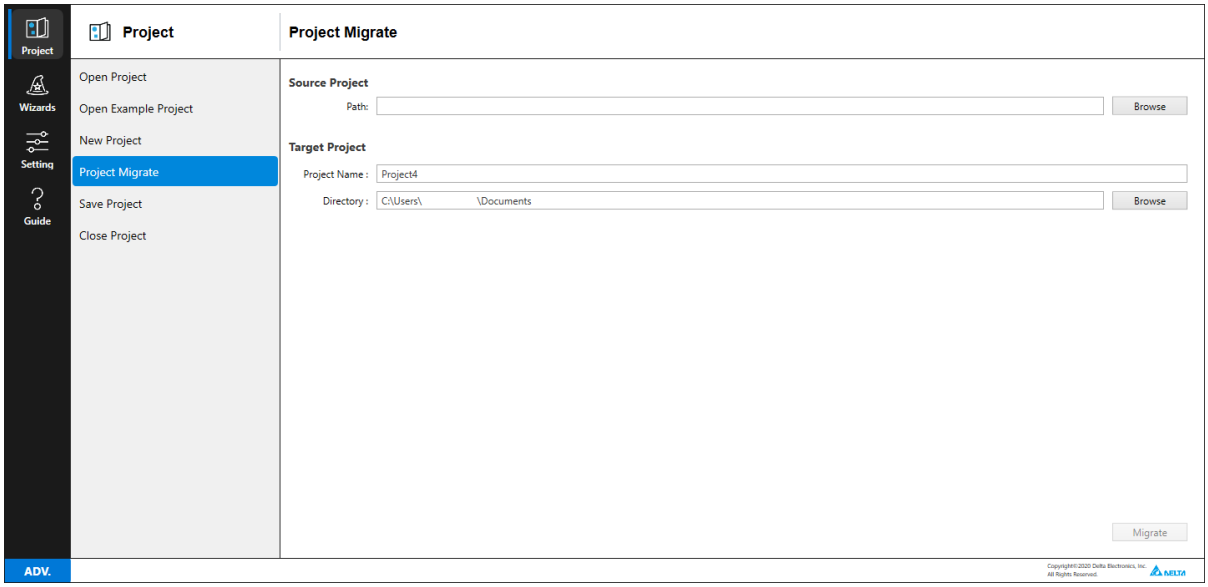


Function	Description
Project Name	Enter a name for the project.
Type	Select **Standard Project** or **Library Project**.
Directory	Displays the project path. By default, this field displays the path selected from the **Browse** function. You can also enter the path.
Browse	Click to select the path.
Author	Enter the author name.
Company	Enter the company name.
Description	Enter the project description.
OK	Click to create a new project.

4.2.1.4 Project Migrate

Migrate the projects created in legacy software such as ISPSoft / WPLSoft into DIADesigner projects. DIASelector projects (including selection results, added devices, module configuration used with the controller, and so on) can also be migrated into DIADesigner without reconfiguration.

The following table lists the functions in the **Project Migrate** pane and their descriptions.

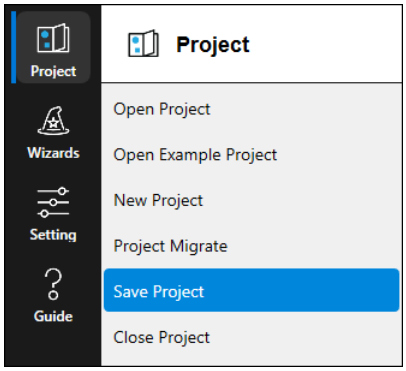


Function	Description
Source Project	
Path	Displays the path of ISPSoft / WPLSoft / DIASelector. By default, this field displays the path selected from the **Browse** function. You can also enter the path.
Browse	Click to select the path of legacy project.
Function	**Description**
Target Project	
Project Name	Enter a name for the project created after project migration.
Directory	Display the project path after project migration. By default, this field displays the path selected from **Browse** function. You can also enter the path.
Browse	Click to select the path to save the project.

Note: For functions affected during project migration, refer to Project Migrate and Unsupported Functions.

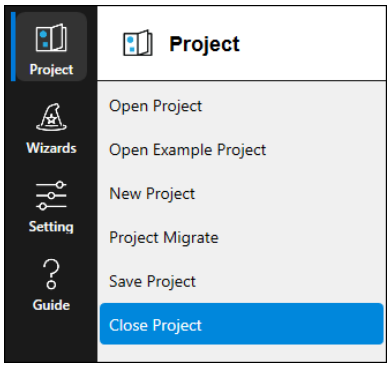
4.2.1.5 Save Project

You can use the **Save Project** function to save a project.



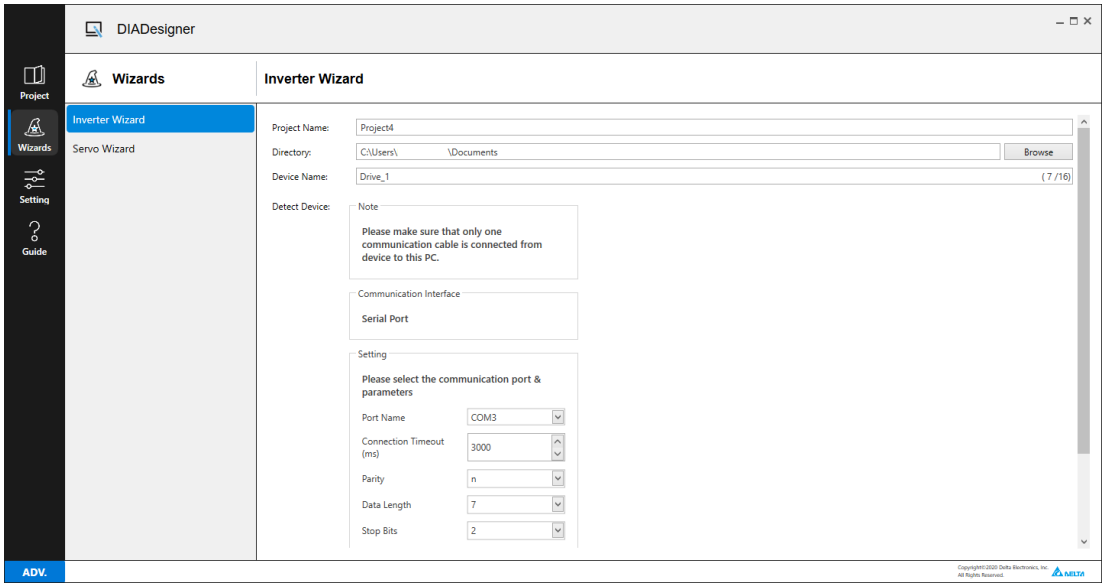
4.2.1.6 Close Project

You can use the **Close Project** function to close a project.



4.2.2 Wizards

In the **Wizards** tab, you can quickly link to various wizards and tools.



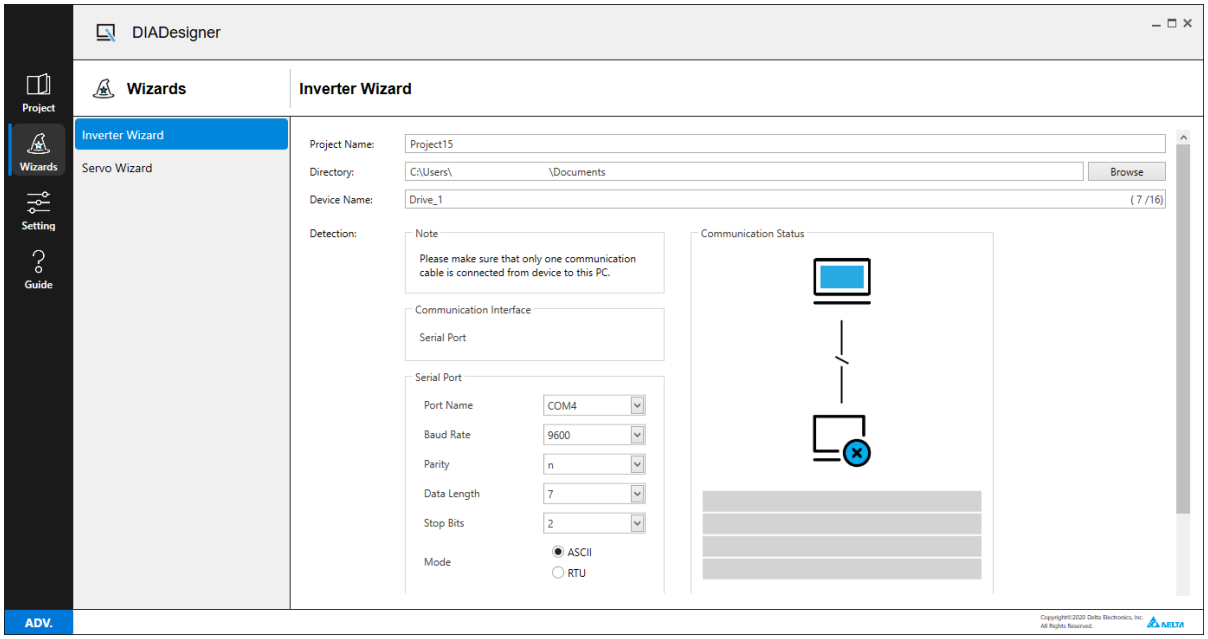
The **Wizards** tab has the following functions.

- Inverter Wizard
- Servo Wizard

4.2.2.1 Inverter Wizard

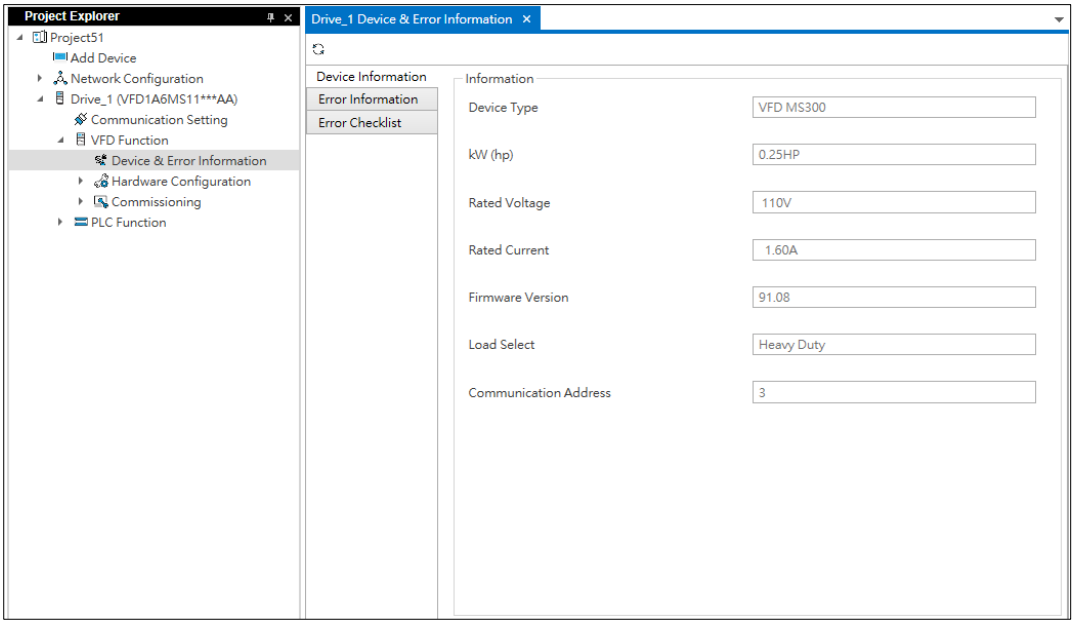
In the **Inverter Wizard** pane, you can add the inverter devices to the project and connect the inverter.

The following table lists the functions in the **Inverter Wizard** pane and their descriptions.



Function	Description
Project Name	Enter a name for the project.
Directory	Displays the project path. By default, this field displays the path selected from the **Browse** function. You can also enter the path.
Device Name	Enter a name for the inverter.
Detection	Set the communication to connect to the inverter or click **Start Detecting Device** to auto-detect the inverter. To stop detection, click **Stop Detecting Device**.
Add Device	This function is only available when a device is detected. Select the dialog as **Device & Error Information** or **Communication Test Tool** that opens after adding a device, and then click **Add Device** to create a project with the device.

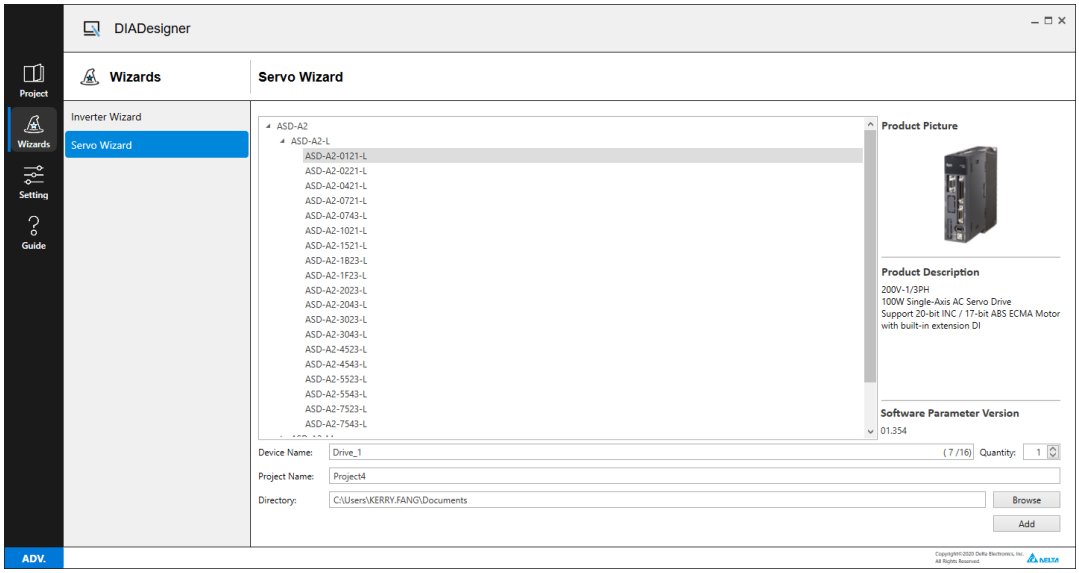
After clicking **Add Device** from the **Inverter Wizard** pane, it will jump to the Advanced Designer interface and automatically connect to the inverter with the device information or communication test tool displayed. For instructions on operating the Advanced Designer, refer to Advanced Designer.

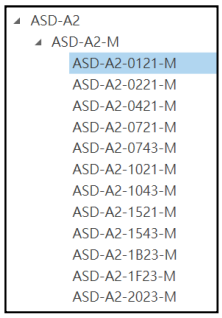


4.2.2.2 Servo Wizard

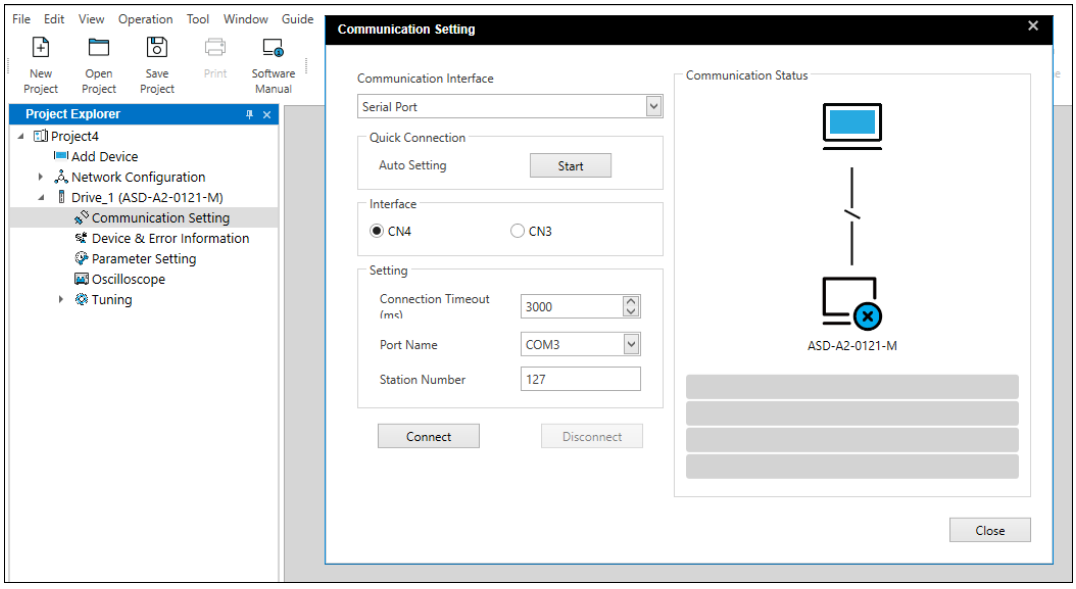
In the **Servo Wizard** pane, you can add the servo drive device to the project.

The following table lists the functions in the **Servo Wizard** pane and their descriptions.



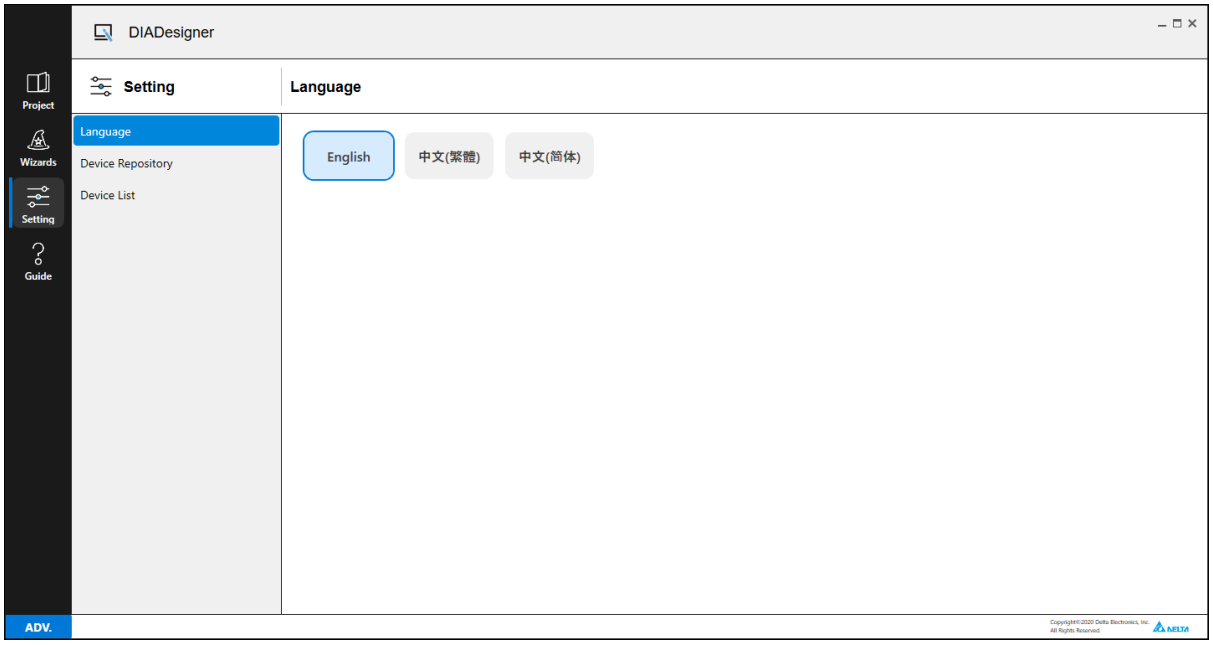
Function	Description
	Displays the servo drive series and part numbers.
Product Picture	Displays the servo drive image.
Product Description	Displays the servo drive description.
Software Parameter Version	Displays the software parameter version.
Device Name	Enter a name for the servo drive.
Quantity	Select or enter the quantity to add.
Project Name	Enter a name for the project.
Directory	Displays the project path. By default, this field displays the path selected from the **Browse** function. You can also enter the path.
Browse	Click to select the path.
Add	Click to add the servo drive to the project.

After a servo drive is selected from the **Servo Wizard** pane, a servo device project is created in **Project Explorer** with **Communication Setting** dialog opened. You can connect to the servo device directly and make the servo settings.



4.2.3 Setting

In the **Setting** tab, you can perform basic settings.



4.2.3.1 Language

In the **Language** pane, you can switch environment language.

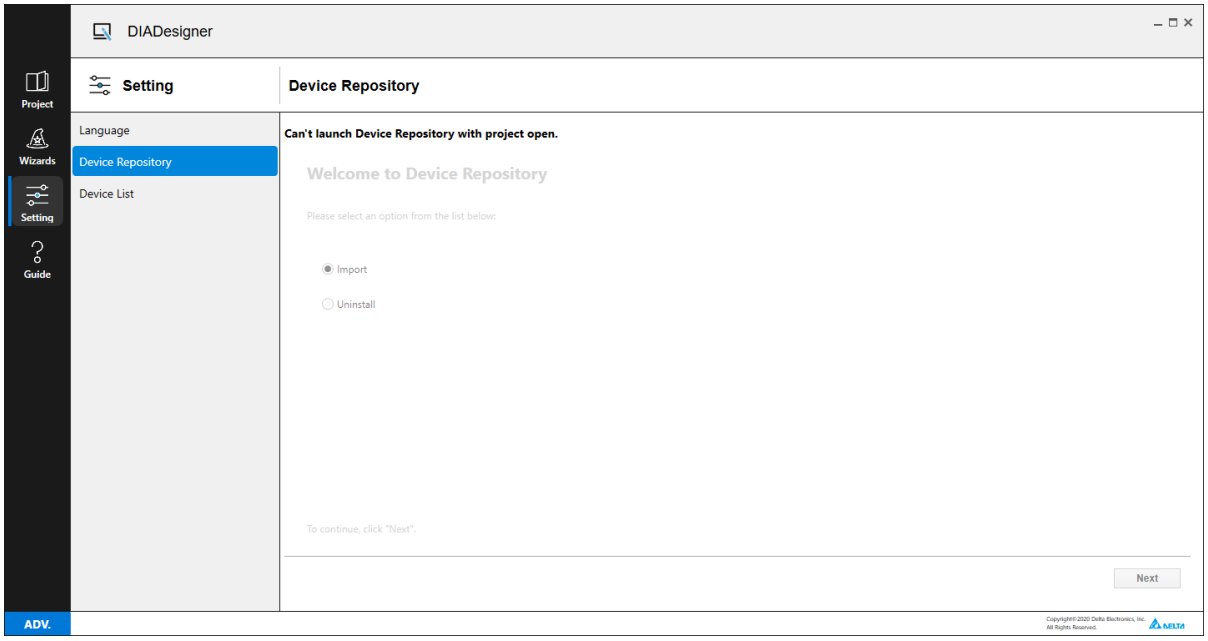
The following table lists the settings in the **Language** pane and their descriptions.

Setting	Description
English	Click to switch to English language environment.
中文（繁體）	Click to switch to Traditional Chinese language environment.
中文（简体）	Click to switch to Simplified Chinese language environment.
Polski	Click to switch to Polski language environment.

4.2.3.2 Device Repository

Device Repository is a management tool for the device description files. A device can be used in the project only when its device description file is installed.

The following table lists the functions in the **Device Repository** pane.



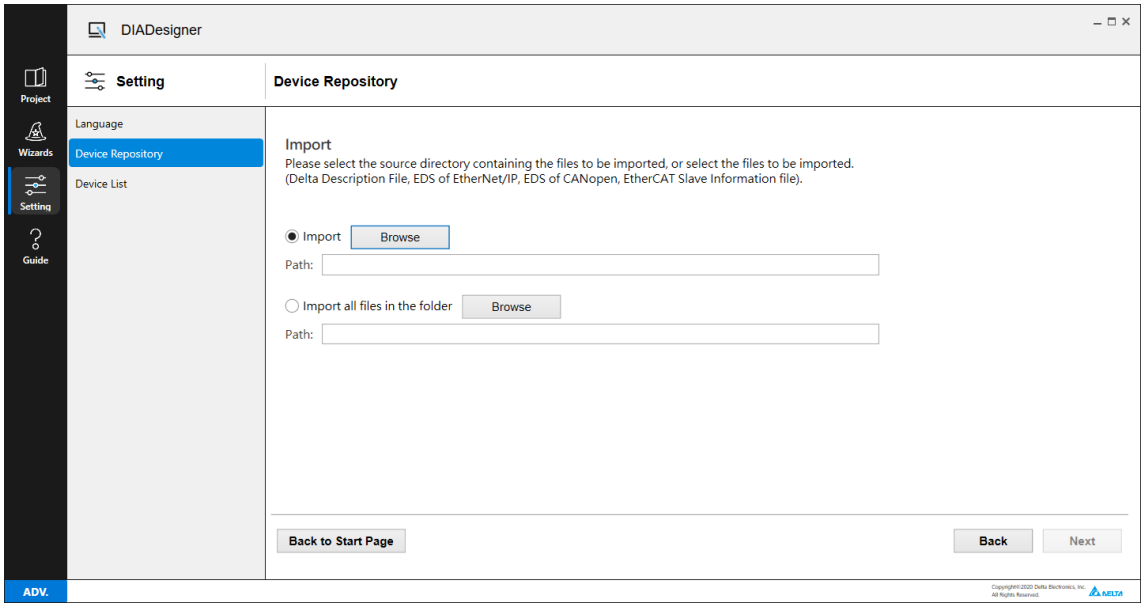
Function	Description
Import	Select to import device description files such as DDF, EDS, and ESI files.
Uninstall	Select to uninstall the installed device description file.

Note: When a project that uses a device description file is transferred to another computer and opened, the used description file will also be transferred.

Follow these steps to import Device Repository.

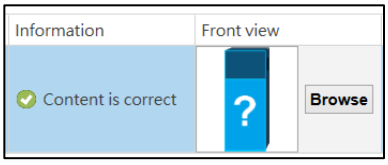
To import Device Repository

1. Click **Browse** to select the third-party electronic parameter file or folder and click **Next**.



DIADesigner checks the correctness of the imported file and displays the check results.

2. Click **Browse** to select other photos and click **Next** to confirm the imported information.



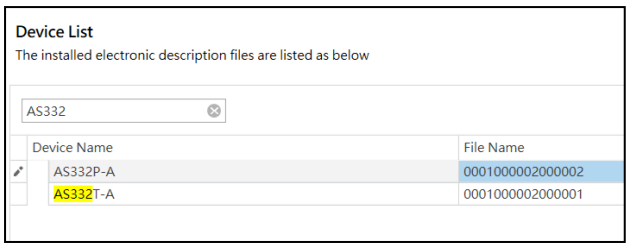
3. Click **Finish**.

Device Repository Uninstall

Select **Uninstall** and follow the guidance to select the device to be removed and delete it.

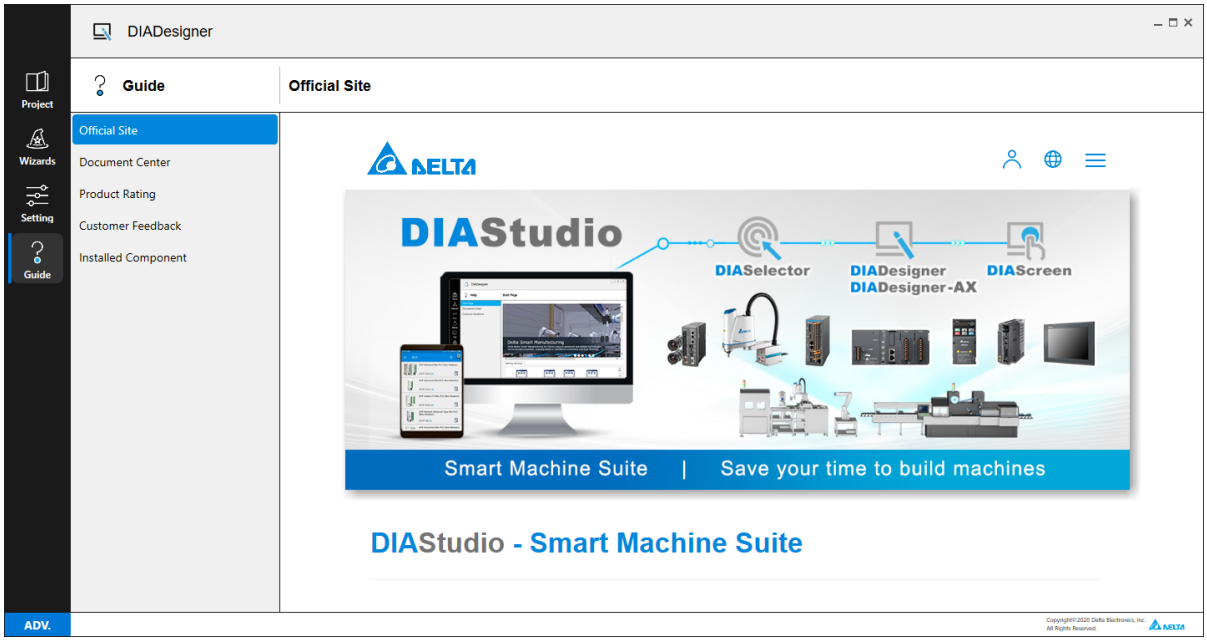
4.2.3.3 Device List

The **Device List** lists the description files (such as EDS files) currently installed in the software in a tree structure for you to look up. It also provides a text filter function. You can enter the text to filter in the input box at the top of the list, and then the device name or the file name will be listed.



4.2.4 Guide

In the **Guide** tab, you can view documents and links.

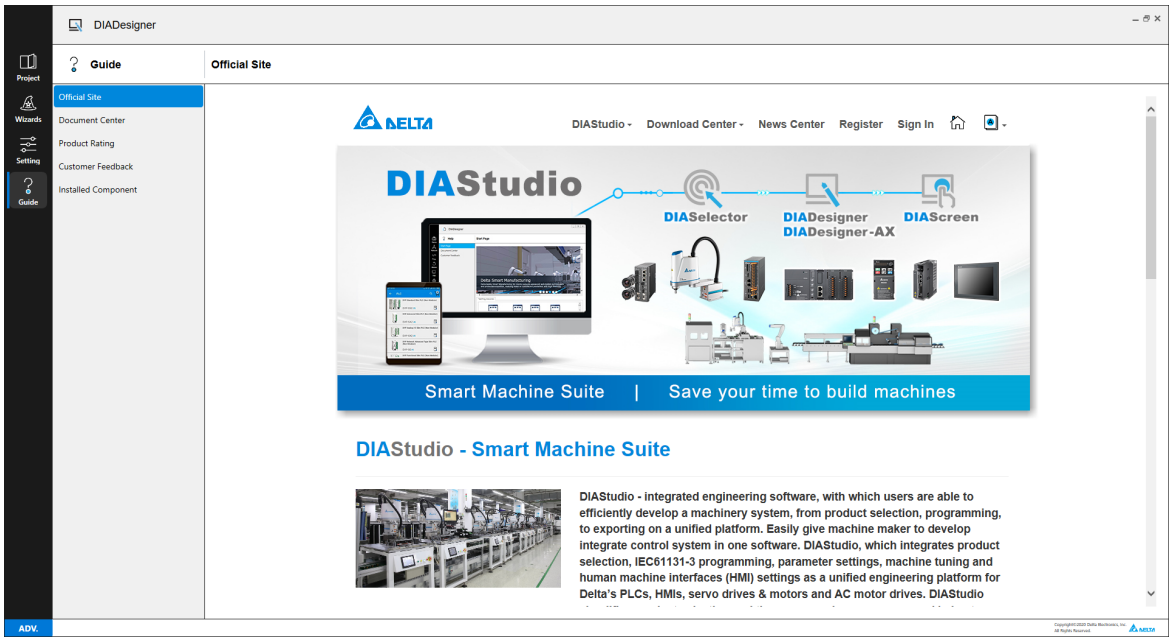


The **Guide** tab has the following functions.

- Official Site
- Document Center
- Product Rating
- Customer Feedback
- Installed Component

4.2.4.1 Official Site

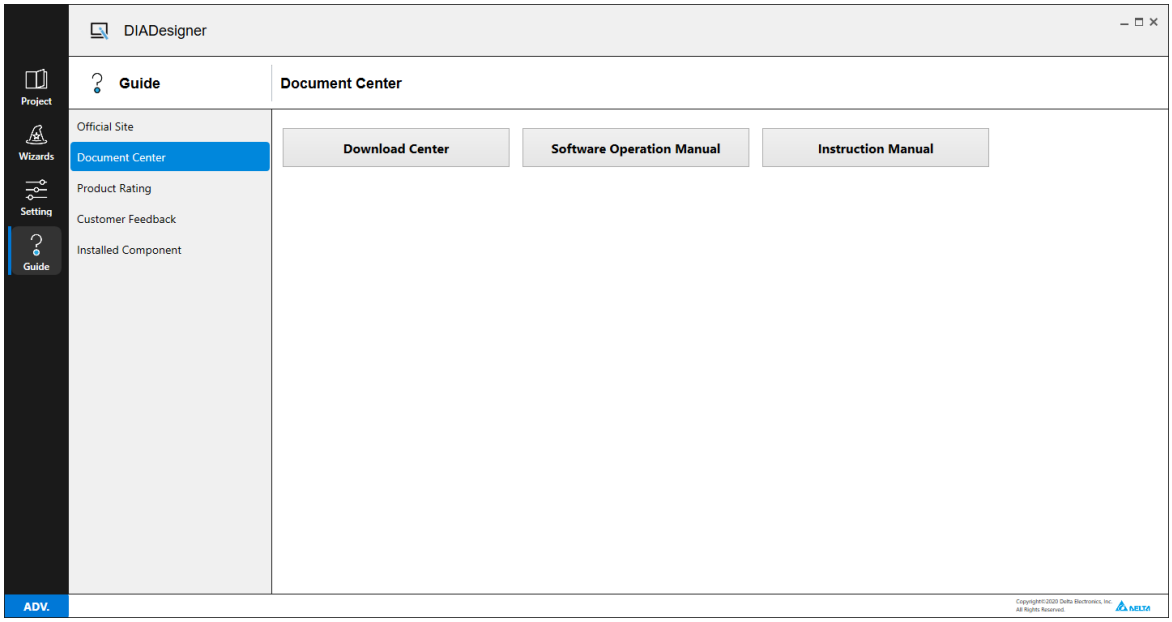
The **Official Site** pane displays the DIAStudio web page. You can get related resources here.



4.2.4.2 Document Center

In the **Document Center** pane, you can get related documents.

The following table lists the functions in the **Document Center** pane and their descriptions.

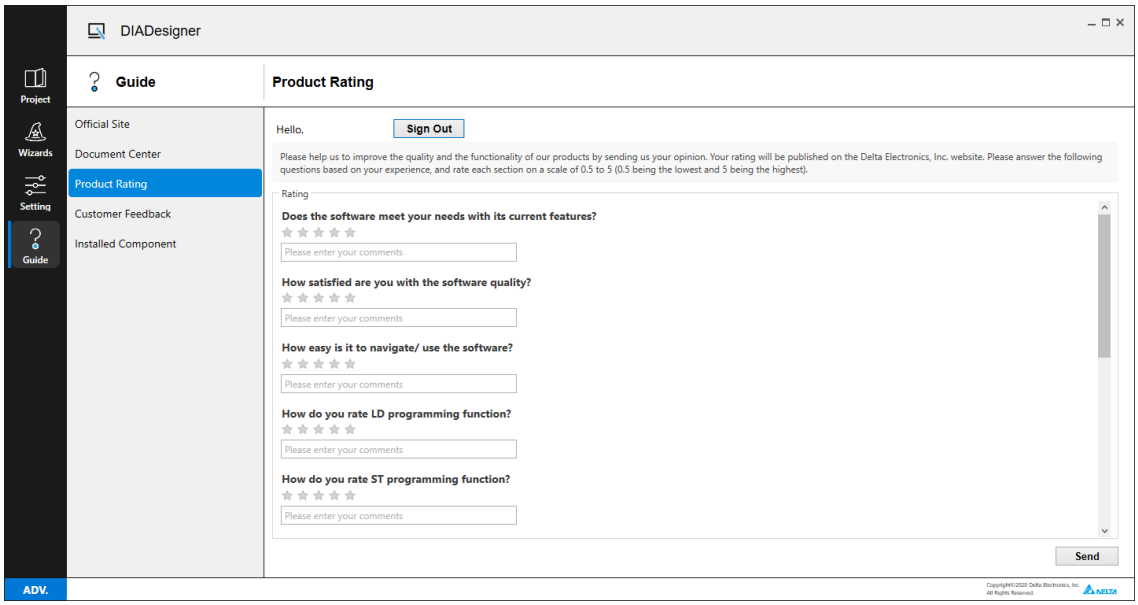


Function	Description
Download Center	Click to open the DIAStudio web page in the default web browser.
Software Operation Manual	Click to open Software Operation Manual.
Instruction Manual	Click to open Instruction Manual.

4.2.4.3 Product Rating

In the **Product Rating** pane, you can rate each function and provide your comments.

The following table lists the functions in the **Product Rating** pane and their descriptions.

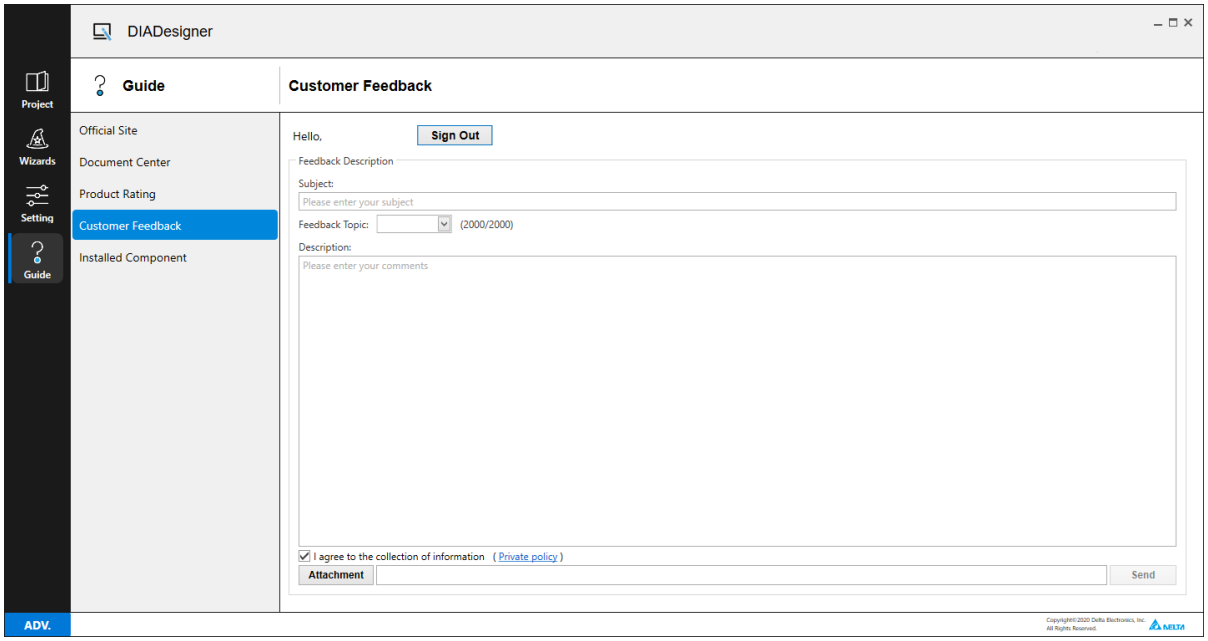


Function	Description
Sign In	Click to sign in to the DIAStudio account.
☆ ☆ ☆ ☆ ☆	Set a rating. 0 is the worst and 5 is the best.
Rating Description	Enter the rating description.
Send	Click to send feedback to Delta.

4.2.4.4 Customer Feedback

In the **Customer Feedback** pane, you can send your feedback and software issues.

The following table lists the functions in the **Customer Feedback** pane and their descriptions.



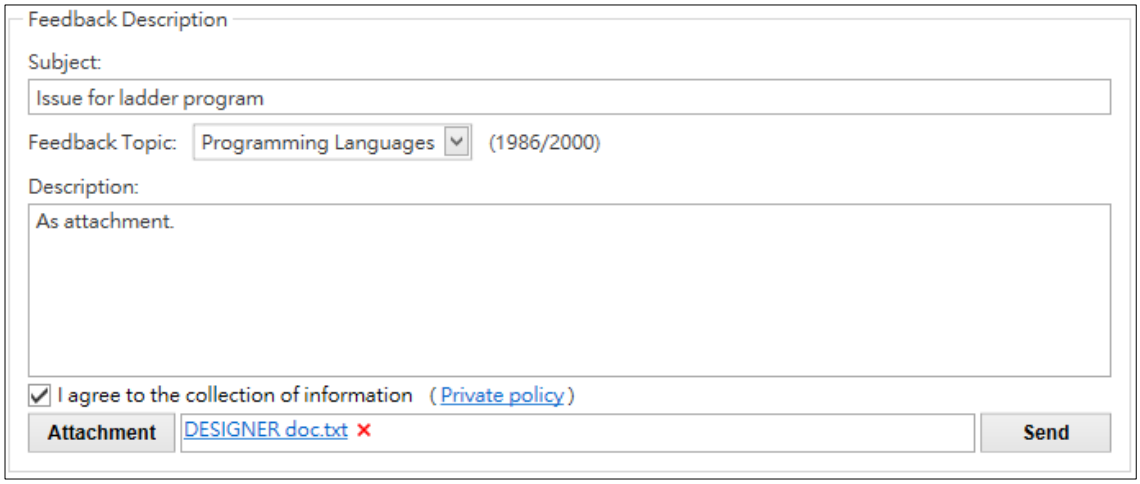
Function	Description
Sign In	Click to sign in to the DIAstudio account.
Subject	Enter the feedback subject.
Feedback Topic	Select a category of feedback topic.
Feedback Description	Enter the customer opinion.
Description	Enter your opinion.
I agree to the collection of information (Private policy)	Click to approve Delta to collect your PC information. You can click **Private Policy** to view the details.
Attachment	Click to select attachment files to send to Delta for analysis. You can select up to 3 files at a time, 2 MB max. for the files.
Send	Click to send the feedback to Delta.

Follow these steps to send customer feedback.

To send customer feedback

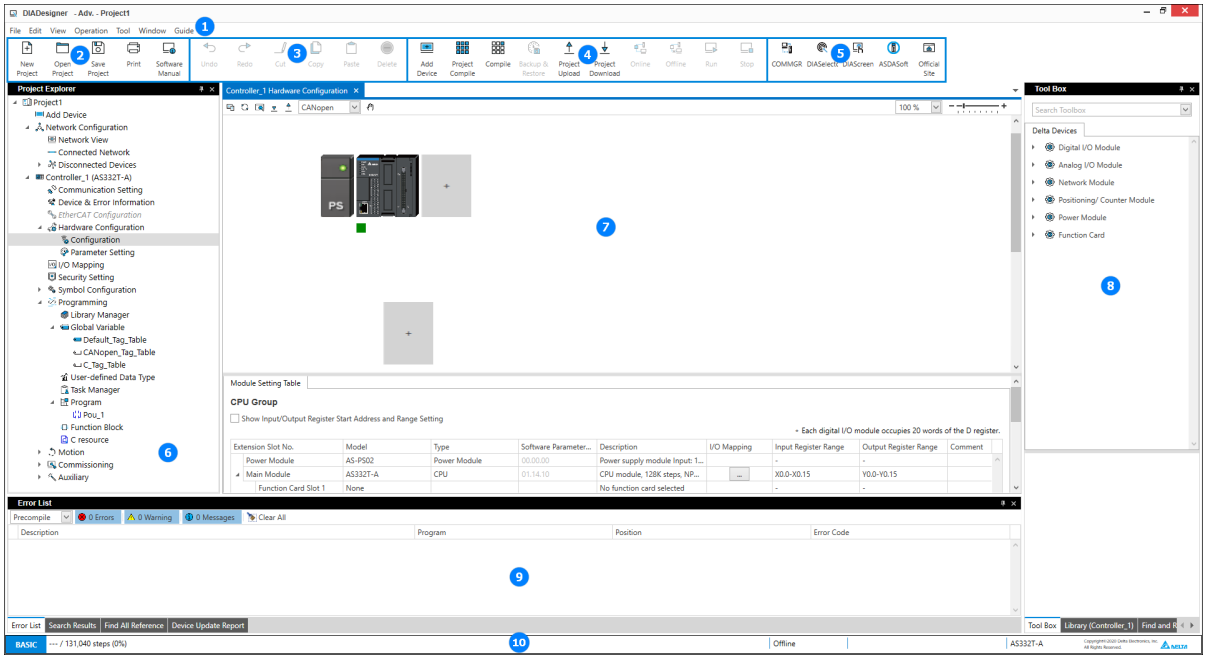
1. Click **Sign In**. Enter your account and password in the **Sign In/Sign Up** dialog.
2. Enter the feedback content and attach the related attachment according to your needs

3. Select the **I agree to the collection of information** checkbox and click **Send**.



4.3 Advanced Designer

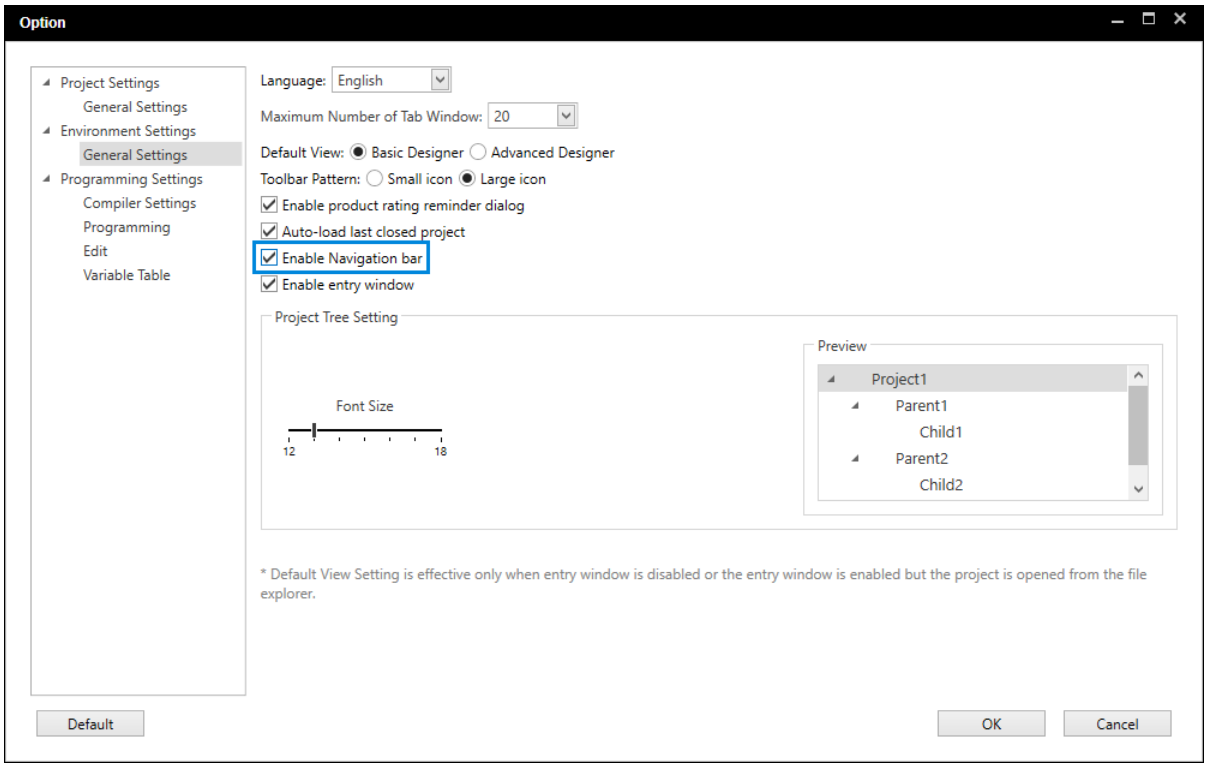
The Advanced Designer page provides a complete integrated development environment to perform the configuration, programming, and networking of devices. You can click **BASIC** at the bottom left to switch to Basic Designer page.



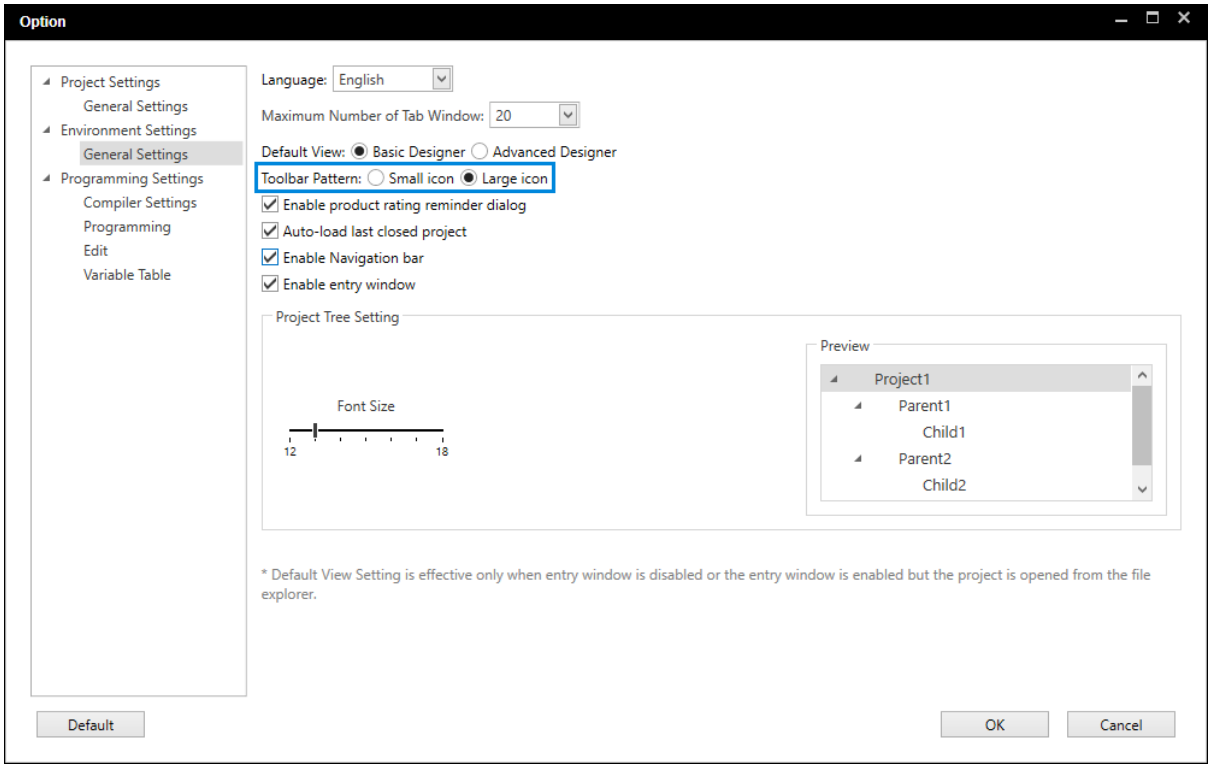
Lengend	Description
1	Menu bar
2	Standard toolbar
3	Editor toolbar

Lengend	Description
❹	Development toolbar
❺	Shortcut toolbar
❻	Project explorer
❼	Work area
❽	Auxiliary functions area
❾	Results area
❿	Status bar

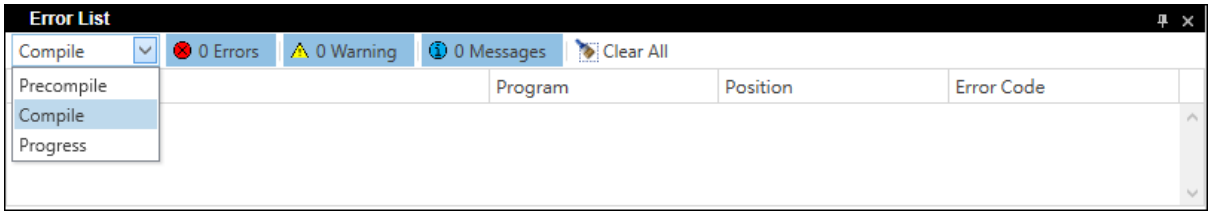
By default, when switching to the Advanced Designer page, the navigation bar of the Basic Designer page is disabled. You can go to **Tools** > **Options** > **Environment Settings** > **General Settings** and select **Enable Navigation bar** checkbox.



When switching to the Advanced Designer page without changing any settings, the toolbar icon is default as **Small icon**. You can go to **Tools** > **Option** > **Environment Settings** > **General Settings** to change the **Toolbar Pattern** to **Large icon**.



In the **Error List** pane, you can view all errors, warnings, and messages of the **Precompile**, **Compile**, or **Progress** individually.



4.3.1 Menu Bar

The following table lists the functions in the menu bar.

File

Function	Description
New Project	Click to create a new project.
Open Project	Click to open a created project.
Open Example Project	Click to open an example project.
Close Project	Click to close the current project.
Save Project	Click to save the current project.
Save As Project	Click to save this project in another name / location.

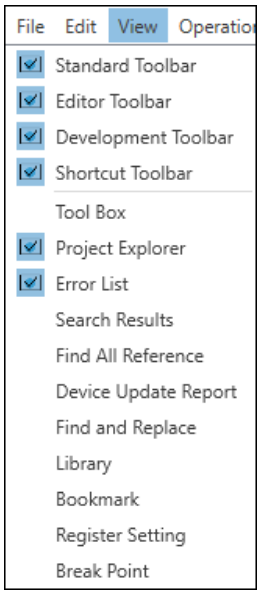
Function	Description
Print	Click to print the current project.
Project Migrate	Click to convert ISPSoft / WPLSoft software projects into DIADesigner format.
Exit Project	Click to exit the current project.
Recently Opened Project	Click to access the recently opened projects.

Edit

Function	Description
Undo	Click to undo the previous action.
Redo	Click to redo the last action.
Cut	Click to cut.
Copy	Click to copy.
Paste	Click to paste.
Delete	Click to delete.
Export	Click to open the export function.
Import	Click to open the import function.
Find and Replace	Click to open the find and replace function.
Bookmark	Click to use the related bookmark tools.

View

Click to hide or display each option.



Operation

You can perform programming-related operations. The functions are the same as those on the Development toolbar. For the functions of the Development toolbar, refer to Development Toolbar.

Tool

Function	Description
Auxiliary Tools	Click to open the related software.
Device List	Click to open the device list.
Device Repository	Click to open the device repository.
Review Error Log File	Click to review the error log file.
Review Status Log File	Click to review the status log file.
Option	Click to perform the background settings.
Language	Click to change the software UI language.

Window

Function	Description
Vertical	Click to tile the windows vertically.
Horizontal	Click to tile the windows horizontally.
Tab	Click to tile the windows as tabs.
Save Windows Layout	Click to save the current windows layout.
Apply Windows Layout	Click to apply the saved windows layout.
Windows Layout Setting	Click to set up the windows layout.
Reset Windows Layout	Click to reset the windows layout.
Auto Hide All	Click to automatically hide all windows.
Close All Tabs	Click to close all tabs at a time.

Guide

Function	Description
Official Site	Click to open the DIAStudio webpage.
About	Click to view the software information.
Manual	Click to choose to open Software Manual or Instruction Manual.

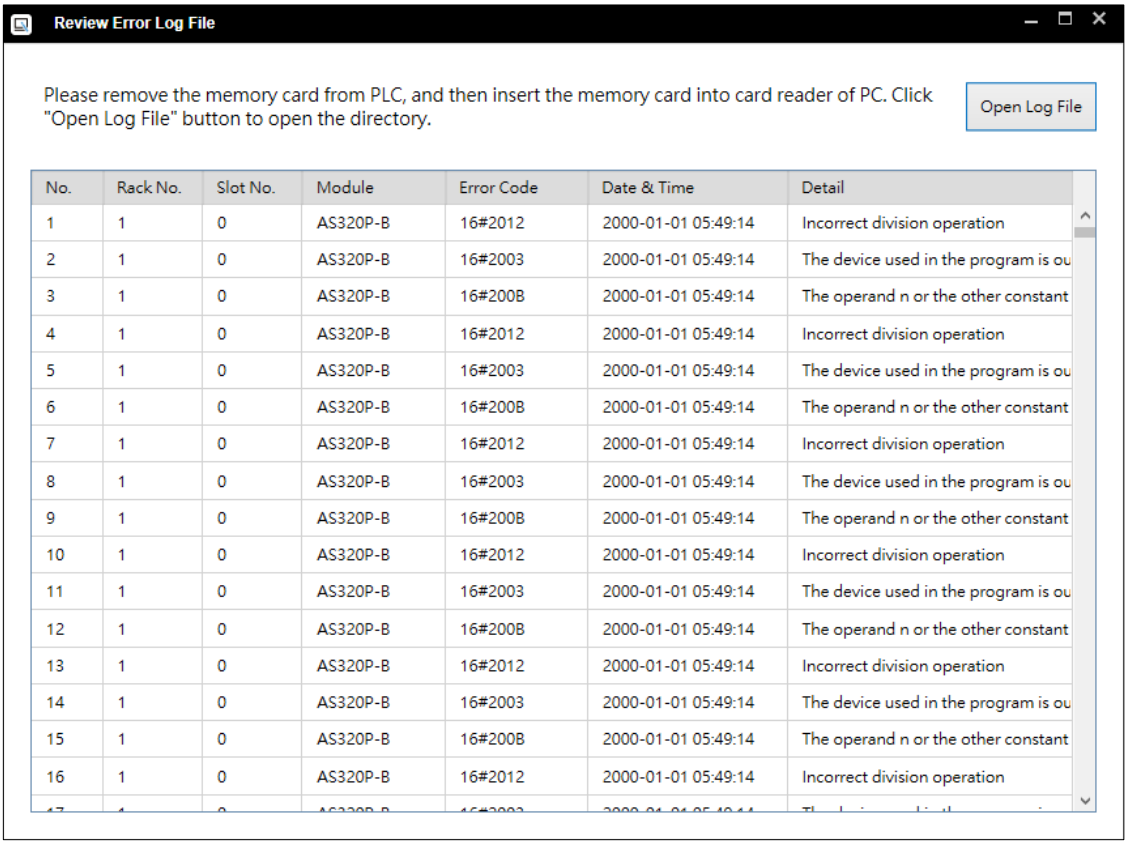
4.3.1.1 Tool – Review Error Log File

The error log file records all errors that occurred on the controller. Use **Review Error Log File** function to open the error log file to view the error information.

To open the error log file

1. Remove the SD card from the controller and insert it into the card reader of the computer.
2. On the **Menu Bar**, click **Tool** > **Review Error Log File**.
3. In the **Review Error Log File** dialog, click Open Log File.

   All error records on the controller display in the list.



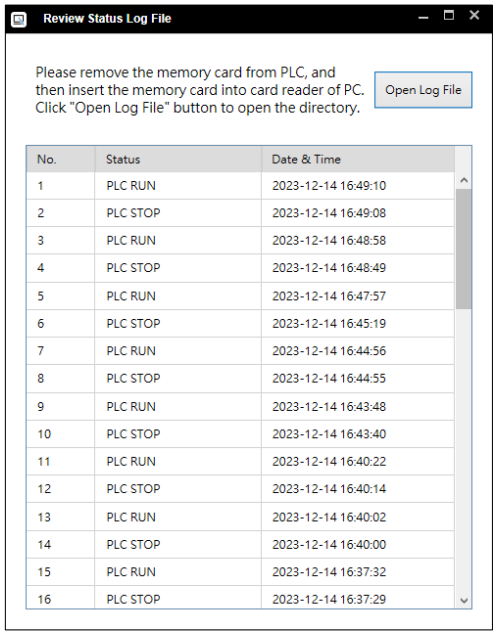
4.3.1.2 Tool – Review Status Log File

The status log file records all status on the controller. Use **Review Error Log File** function to open the status log file to view the status information.

To open the status log file

1. Remove the SD card from the controller and insert it into the card reader of the computer.
2. On the **Menu Bar**, click **Tool** > **Review Status Log File**.
3. In the **Review Status Log File** dialog, click Open Log File.

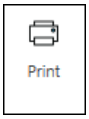
   All status records on the controller display in the list.



4.3.2 Standard Toolbar

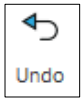
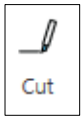
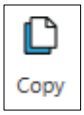
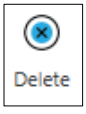
The following table lists the functions on the standard toolbar.

Function	Description
	Click to create a new project.
	Click to open a created project.
	Click to save the standard project.

Function	Description
	Click to save the library project. **Note**: This function is available after creating a library project.
	Click to print the project.
	Click to open software manual.

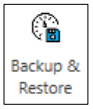
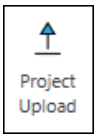
4.3.3 Editor Toolbar

The following table lists the functions on the editor toolbar.

Function	Description
	Click to undo the previous action.
	Click to redo the last action.
	Click to cut.
	Click to copy.
	Click to paste.
	Click to delete.

4.3.4 Development Toolbar

The following table lists the functions on the development toolbar.

Function	Description
	Click to add a new device.
	Click to compile project.
	Click to compile device.
	Click to execute back and restore of the inverter devices in the project.
	Click to upload projects from the device to the computer.
	Click to download projects from the computer to the device.
	Click to go online between the computer and the device.
	Click to go offline between the computer and the device.
	Click to run the device.
	Click to stop the device.
	Click to switch the status of all devices in Project Explorer from disconnection to connection according to the current communication settings.

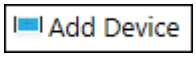
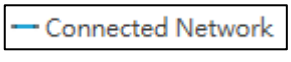
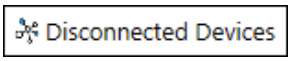
4.3.5 Shortcut Toolbar

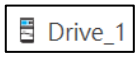
The following table lists the functions of the shortcut toolbar.

Function	Description
	Click to launch COMMGR software.
	Click to launch DIASelector software.
	Click to launch DIAScreen software.
	Click to launch ASDASoft software.
	Click to open DIAStudio web page.

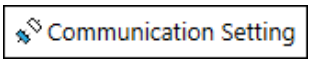
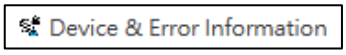
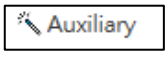
4.3.6 Project Explorer

Project Explorer lists various functions supported by the device, such as PLC programming and parameter editor. To quickly expand or collapse a single node or all nodes in **Project Explorer**, right-click the project name and select **Expand/Collapse Project Tree**. The following table lists the functions of **Project Explorer**.

Function	Description
	Double-click to add the required device.
	Double-click to set up device network.
	Double-click to open network view.
	Displays the connected network.
	Displays the unconnected devices.
	Displays the added controller.
	Displays the added servo drive.

Function	Description
	Displays the added inverter.
	Displays the added temperature controller.

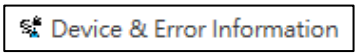
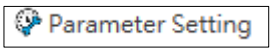
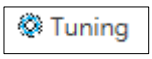
The following table lists the sub-items of the controller functions in **Project Explorer**.

Features	Description
	Double-click to set the device communication.
	Double-click to open Device & Error Information pane.
	Double-click to configure hardware of devices, such as parameter setting.
	Double-click to open Security Setting pane.
	Double-click to execute symbol configuration of devices. Specify or view the properties of variables.
	Double-click to execute programming of devices, such as global variables, user-defined data type, program, and function block.
	Double-click to set the motion control, such as position plan table.
	Double-click to execute commission of devices, such as monitor table, data tracer, and data logger.
	Double-click to use auxiliary tools, such as register edit, register resource allocation, step positioning, format PLC memory, retain variables initialization, and CARD Utility.

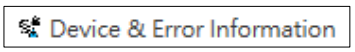
The following table lists the functions of the servo drive in **Project Explorer**.

- **ASDA-A2 and ASDA-B2 series**

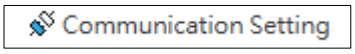
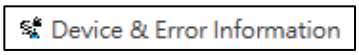
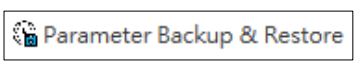
Features	Description

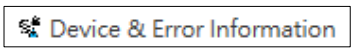
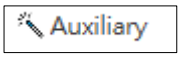
	Double-click to set the device communication.
	Double-click to open the Device & Error Information pane.
	Double-click to open the Parameter Setting pane.
	Double-click to open the Oscilloscope pane.
	Double-click to execute tuning of devices, such as gain adjustment.

- **ASDA-A3 and ASDA-B3 series**

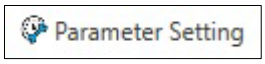
Features	Description
	Double-click to set the device communication.
	Double-click to open the Device & Error Information pane.
	Double-click to open the Servo Setting dialog. Refer to the note of Servo Setting.

The following table lists the functions of the inverter in **Project Explorer**.

Features	Description
	Double-click to set the device communication.
	Double-click to set inverter functions.
	Double-click to open the Device & Error Information pane.
	Double-click to configure hardware of devices, such as parameter setting.
	Double-click to backup and restore parameter.
	Double-click to execute commission of devices, such as trace and communication test tools.

Features	Description
	Double-click to set built-in PLC functions.
	(For PLC Function) Double-click to open the Device & Error Information pane.
	Double-click to open the Security Setting pane.
	Double-click to execute programming of devices, such as global variables, program, and function block.
	(For PLC Function) Double-click to execute commission of devices, such as monitor table.
	Double-click to use auxiliary tools, such as register edit, register resource allocation, step position, and format PLC memory.

The following table lists the functions of the temperature control in **Project Explorer**.

Features	Description
	Double-click to set the device communication.
	Double-click to open the Parameter Setting pane.
	Double-click to set the temperature controller function, such as temperature monitor, communication and station number automatic synchronization, and program planning.

4.4 Creating Project and Device

You can create a new project from the menu bar, standard toolbar, and keyboard shortcuts.

To create new project

1. Do one of the following:

- Click  on the standard toolbar.

Or

- Click **File** > **New Project** from the menu bar.

Or

- Press **Ctrl** + **C** keys simultaneously on the keyboard.

Note: The functions in the **New Project** dialog are the same as mentioned in the Basic Designer.

2. Enter the **Project Name** and select the project **Type**, **Directory**, **Author**, and other information according to your need.
3. Click **OK**.

   Project is saved in the given location with *.diade extension.

4.4.1 Add Device

You can add devices to the project manually or by scanning.

4.4.1.1 Add New Devices Manually

You can add devices to the project from the menu bar, toolbar or **Project Explorer**.

To add devices to project

1. Do one of the following:

- Click 

on the development toolbar.

Or

- Click **Operation** > **Add Device** from the menu bar.

Or

- Click 

in **Project Explorer**.

Note: The **Add Device** dialog functions are the same as mentioned in Basic

Designer.

4.4.1.2 Add by Scan

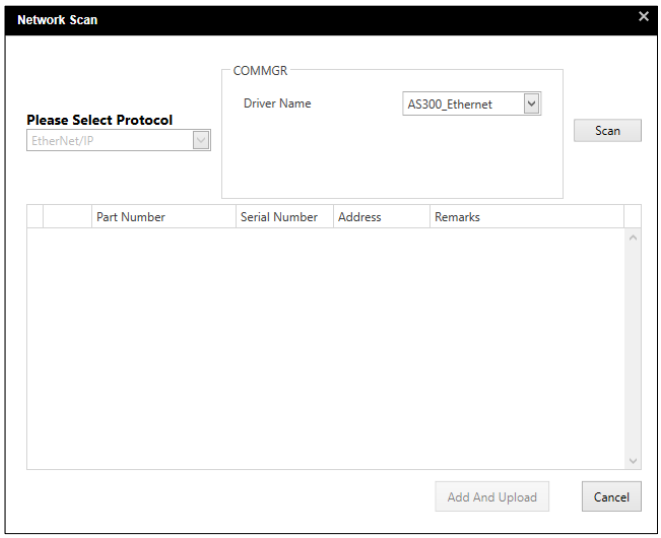
You can add Ethernet devices to DIADesigner by **Add by Scan** function.

Note: Only EtherNet/IP device scan is supported in the current version.

To add devices by scan

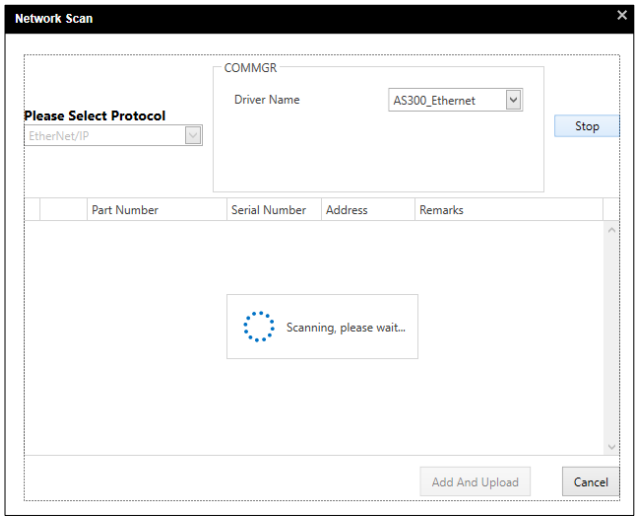
1. Open COMMGR software and create an **Ethernet** Driver.
2. In DIADesigner, click **Network Configuration** > **Network View** in **Project Explorer**.
3. Click  in the **Network View** pane.

4. Select the Ethernet driver from the **Driver Name** drop-down list in the **Network Scan** dialog.

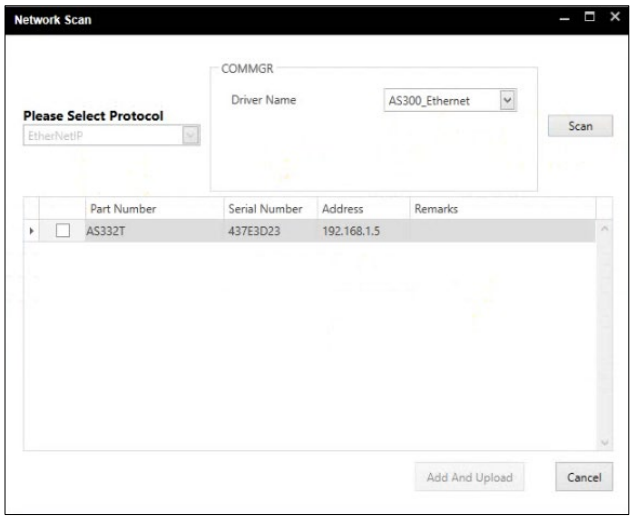


5. Click **Scan**.

   The scanning starts. Click **Stop** to stop the scanning.

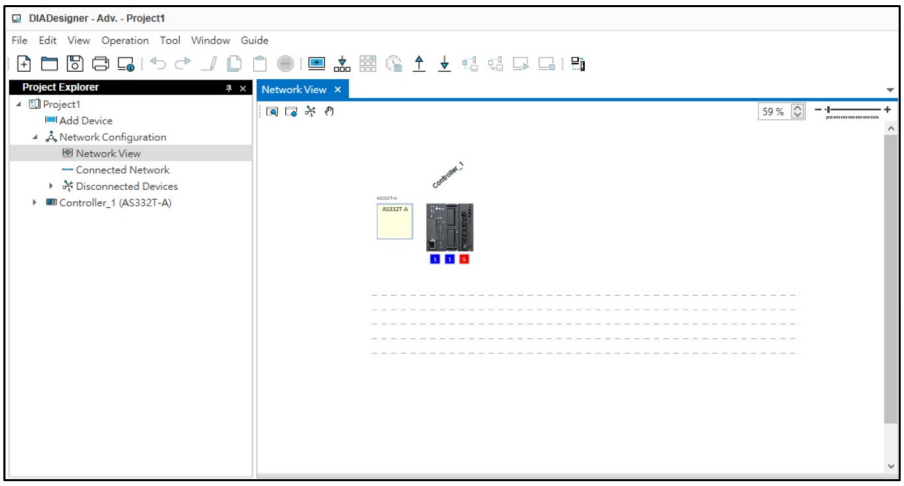


   The information of the scanned devices displays in the below device list.



6. Select the devices to add to the project and click **Add and Upload**.

   The selected device is added to the project.



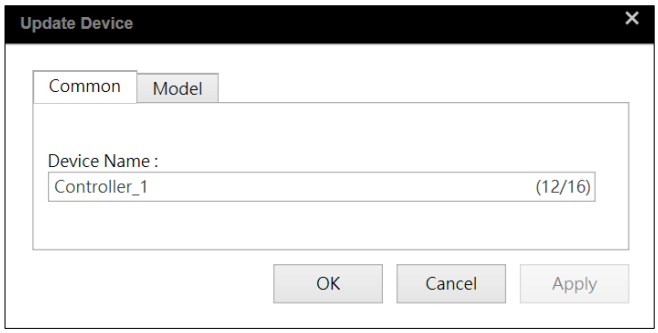
4.4.1.3 Directly Connect to Inverter

Refer to Inverter Wizard for the details about directly connecting to inverters.

4.4.2 Device Name

To modify the device name, do one of the following:

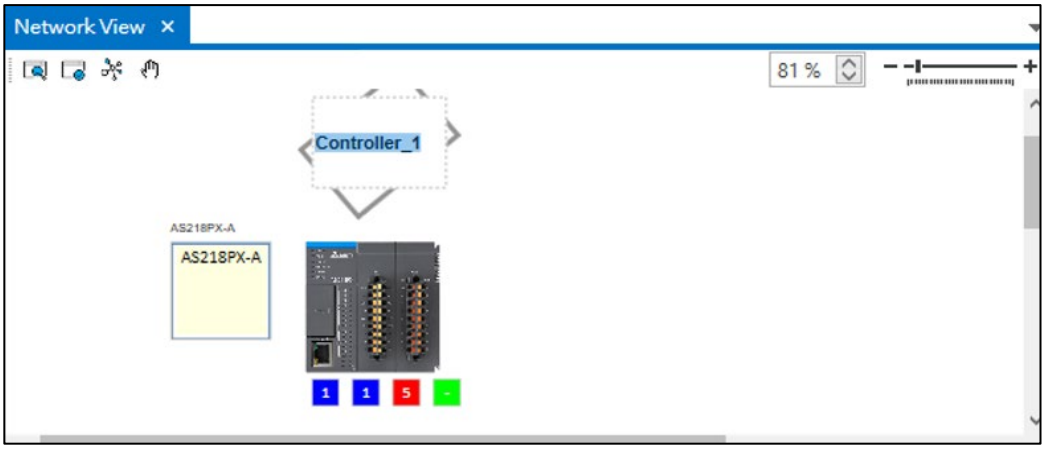
- Right-click the device in **Project Explorer** and select **Update Device**.



- Click the device in **Project Explorer**, and then press **F2**.
- Double-click **Network View** > **Network View**, and then double-click the device name above the device.

  Or

  Click the device name above the device, and then press **F2**.



4.4.3 Update Device

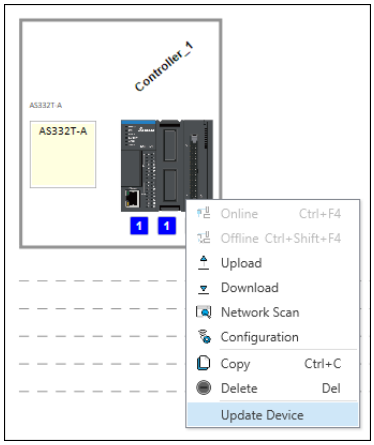
All AS Series controllers can be changed. Programs and sharable configurations retain their settings, and unshared configurations or parameters restore to their default values.

To update devices

1. Right-click the device in **Project Explorer** and select **Update Device**.

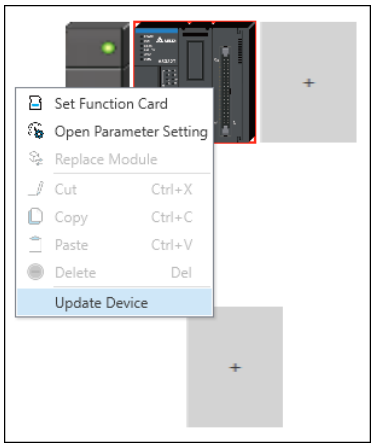
   Or

   Double-click **Network View** > **Network View**, and then right-click the device in the **Network View** pane and select **Update Device**.

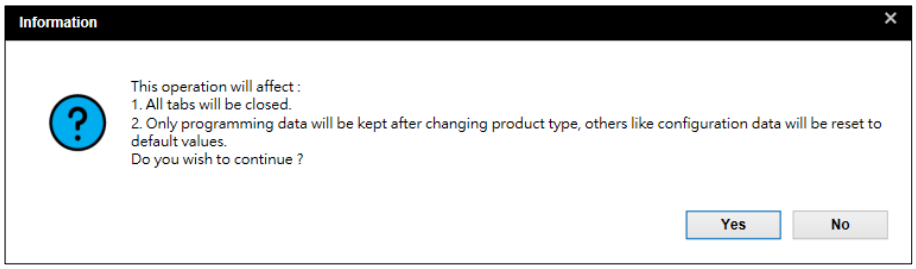


Or

Double-click **Hardware Configuration** > **Configuration** in **Project Explorer**, and then, right-click the device in the **Hardware Configuration** view and select **Update Device**.



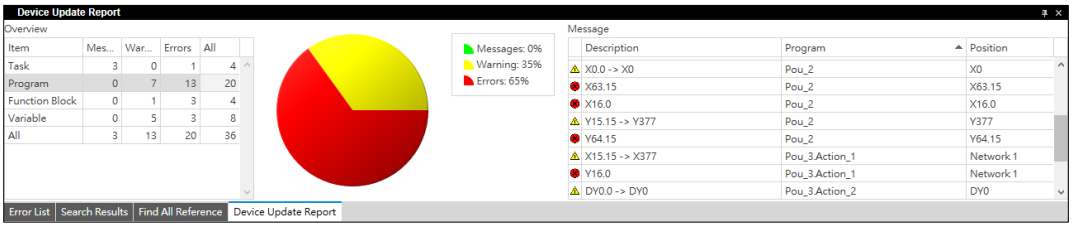
2. In the **Update Device** dialog, select **New Model** in the **Model** tab, and then click **OK**.
3. Confirm the message and click **Yes** in the pop-up dialog box.



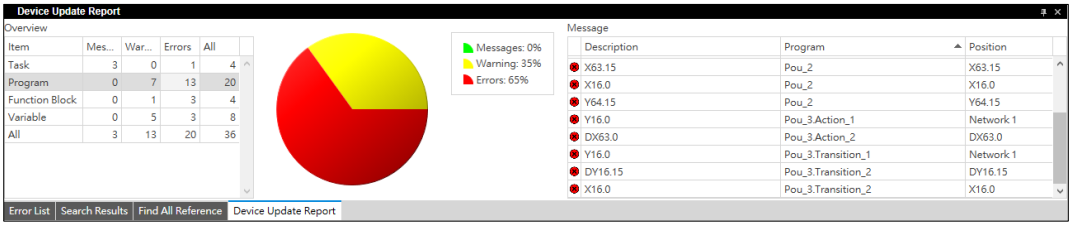
Update Device Report

The **Update Device Report** dialog appears at the bottom after the device model is updated. You can view the total number of messages, warnings, or errors for **Task**,

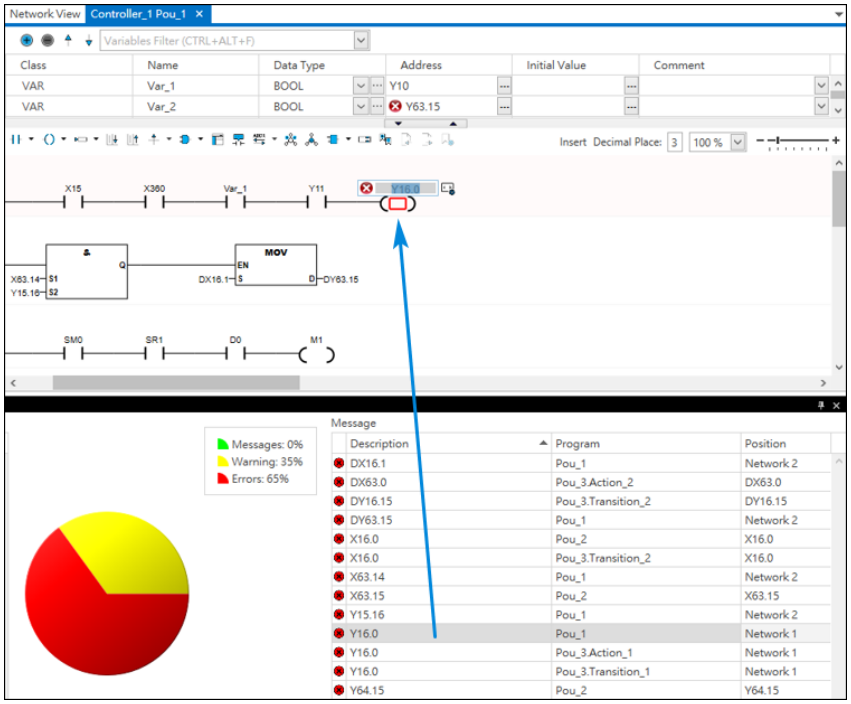
Program, **Function Block**, and **Variable** in the **Overview** area. You can also click each field to view the description, corresponding program, and corresponding position of each message in the **Message** area.



Click the field to view information (such as the **Error** field of the **Program** item), and then the message area lists the error items.



Double-clicking the error item listed in the **Message** area, the system guides you to where the error item is located (displayed at the top of the screen) to view the information or modify the parameter.



Chapter 5: Communication Setting

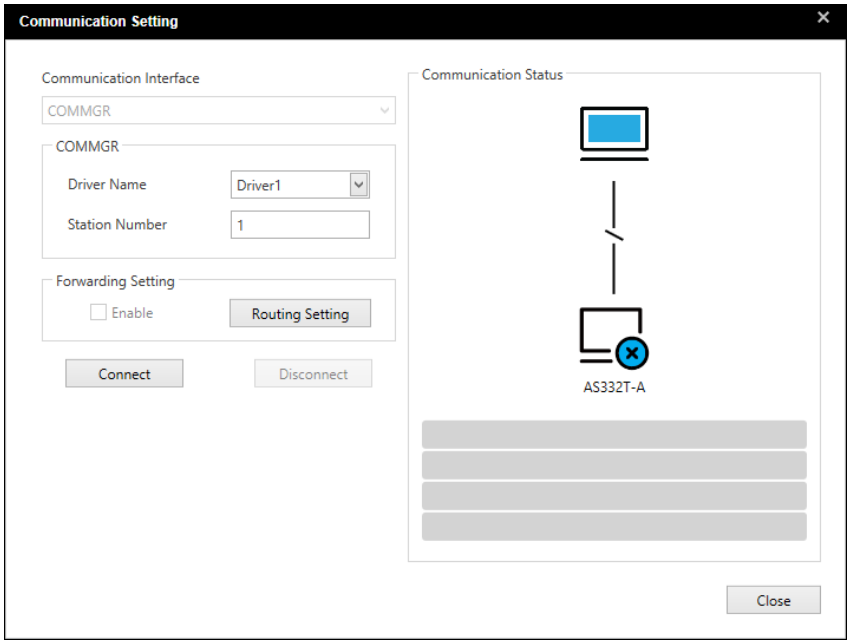
5.1 Communication via COMMGR

Using COMMGR (Communication Manager) in DIADesigner, you can quickly establish communication between DIADesigner and connected devices and perform related configuration through the communication interface function of DIADesigner.

How to open the **Communication Setting** dialog?

- In **Project Explorer**, double-click **Device** > **Communication Setting**.

The following table lists the functions in the **Communication Setting** dialog.



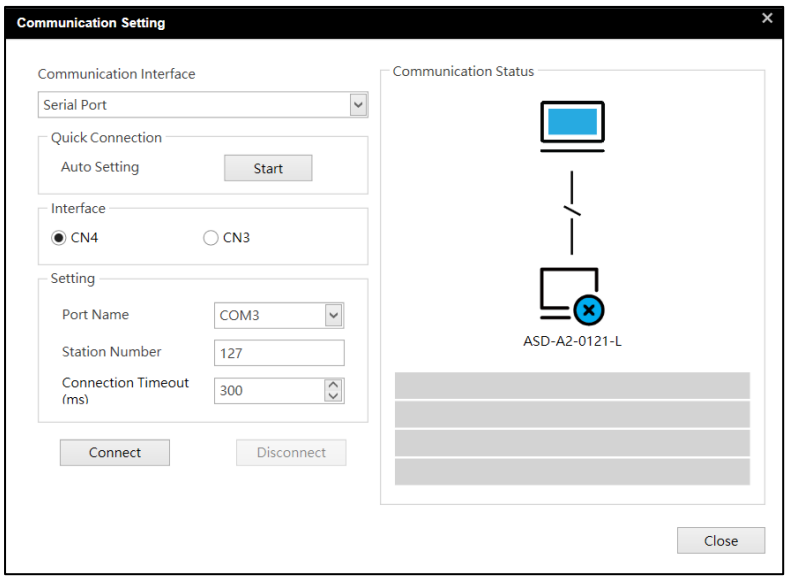
Function	Description
Communication Interface	Select the device interface. The available options include: • COMMGR • Serial Port • Ethernet (When the PC is connected to the inverter's Ethernet communication card using Ethernet) **Note**: Serial Port is available for devices other than Controller devices such as Servo Drive or AC Drive.
COMMGR – Driver Name	Select the driver created in COMMGR.
COMMGR – Station Number	Enter the station number of the device to be connected.
Forwarding Setting – Enable	Select to enable forwarding communication mode.

Function	Description
Forwarding Setting – Routing Setting	Click to set path for connecting device.
Connect	Click to establish communication between DIADesigner and the device. If the connection is unsuccessful, a timeout message appears after a while.
Disconnect	Click to disconnect the communication between DIADesigner and the device.
Communication Status	Displays the current communication status between DIADesigner and the device.

5.2 Communication without COMMGR

Select **Serial Port** for **Communication Interface** and then setup **Port Name** and parameters through the in the **Communication Setting** dialog to allow DIADesigner to directly communicate with the servo drive, inverter and other devices.

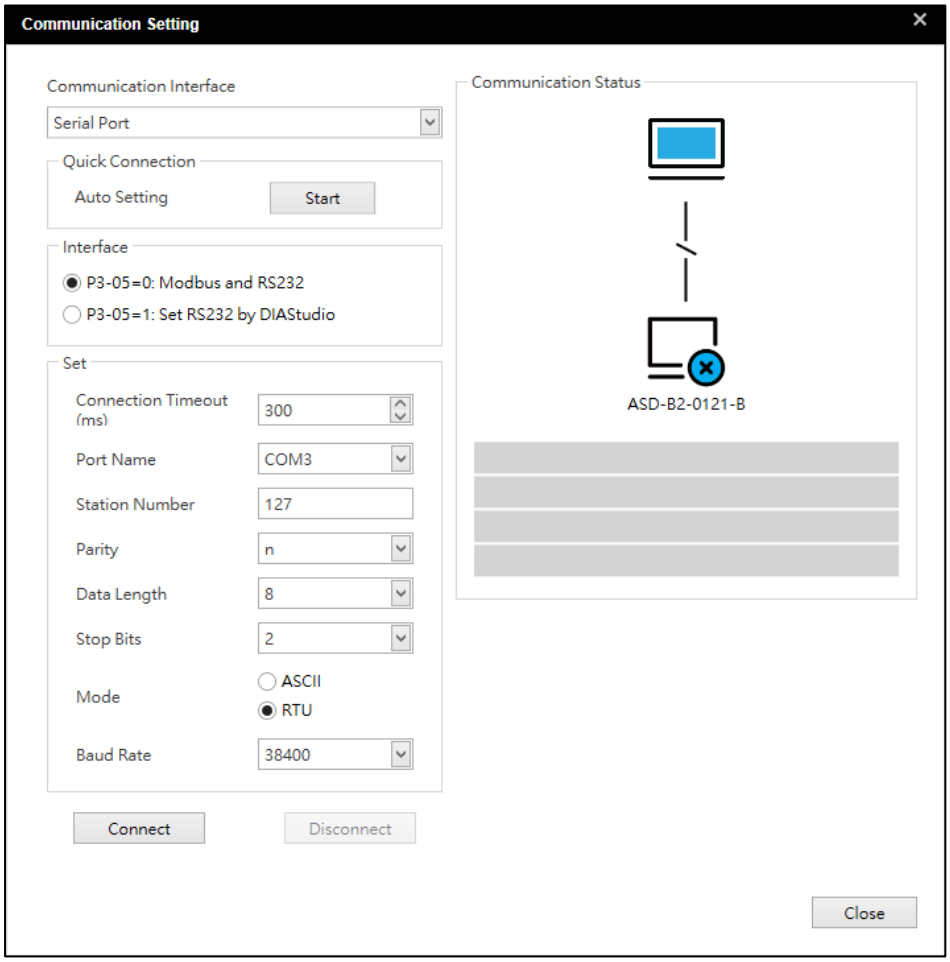
The following figure shows an example for communication using **CN4** interface of ASDA-A2 servo drive.



Click **Start** in **Quick Connection** area to establish communication automatically according to the selected interface.

- **CN4**: CN4 is the USB communication interface on the Servo Drive.
- **CN3**: CN3 is the RS-485 / RS-232 serial communication interface on the servo drive.

The following figure shows an example for communication using **CN3** interface of ASDA-A2 servo drive.

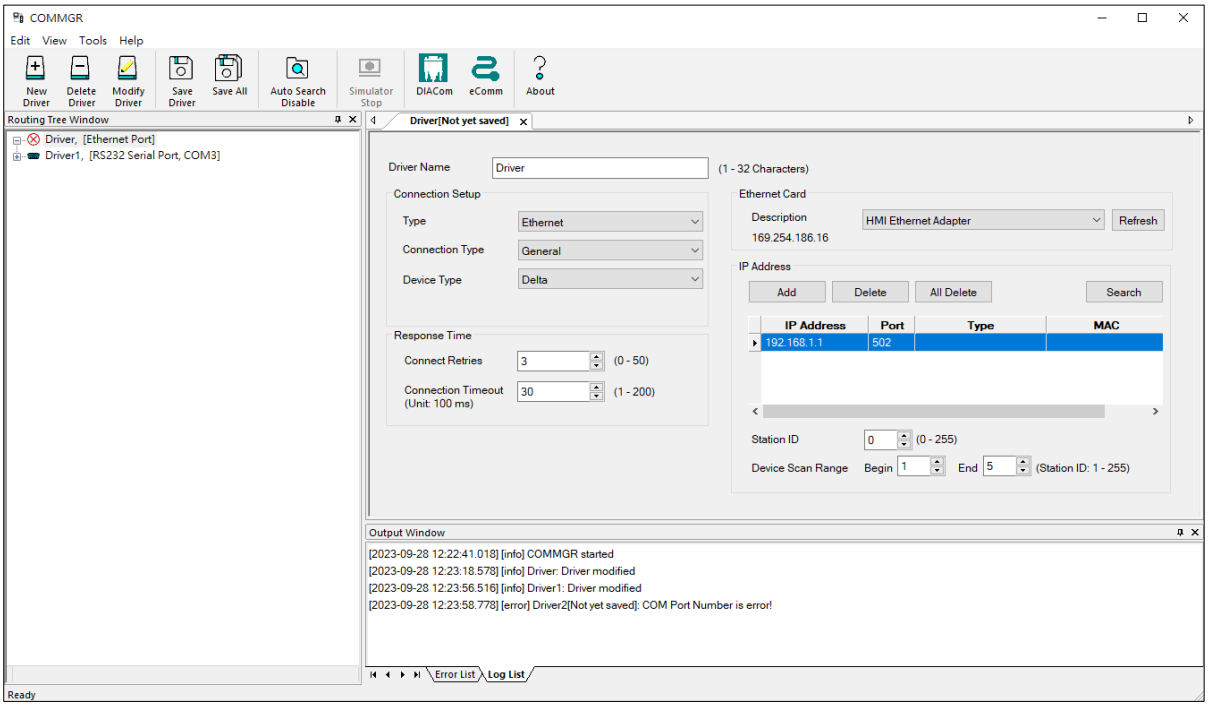


Function	Description
Connection Timeout (ms)	Select or enter the connection timeout. The connection terminates if the device doesn't respond within the set time. The value range is 100~20,000 ms.
Port Name	Select the COM port of the connected device.
Station Number	Enter the station number of the device to be connected.
Parity	Select whether to set the parity.
Data Length	Select the data length.
Stop Bits	Select whether to set the stop bits when communicating.
Mode	Select the communication mode.
Baud Rate	Select the communication speed.

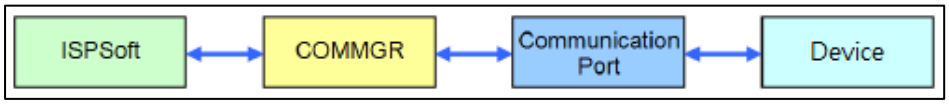
5.3 COMMGR Overview

The COMMGR is the new generation communication management tool developed by Delta Electronics, Inc. Its main function is to establish communication between DIAStudio and Delta's devices. Communication configuration becomes more convenient through the management of COMMGR.

Create communication parameters to be used according to the management list in COMMGR. The communication parameters are called **Drivers**. If a driver is started, a connection is created automatically. After specifying a driver in DIADesigner, communication is carried out at the same time.



The following figure shows communication between DIADesigner and Delta hardware product via Drivers.



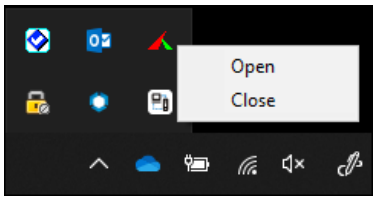
The COMMGR is a software independent of DIADesigner. It must be installed and uninstalled separately. If an earlier version of COMMGR has been installed on a computer, uninstall it before installing the new version.

Refer to Install, Uninstall, and Update DIADesigner for detailed installation and uninstallation methods.

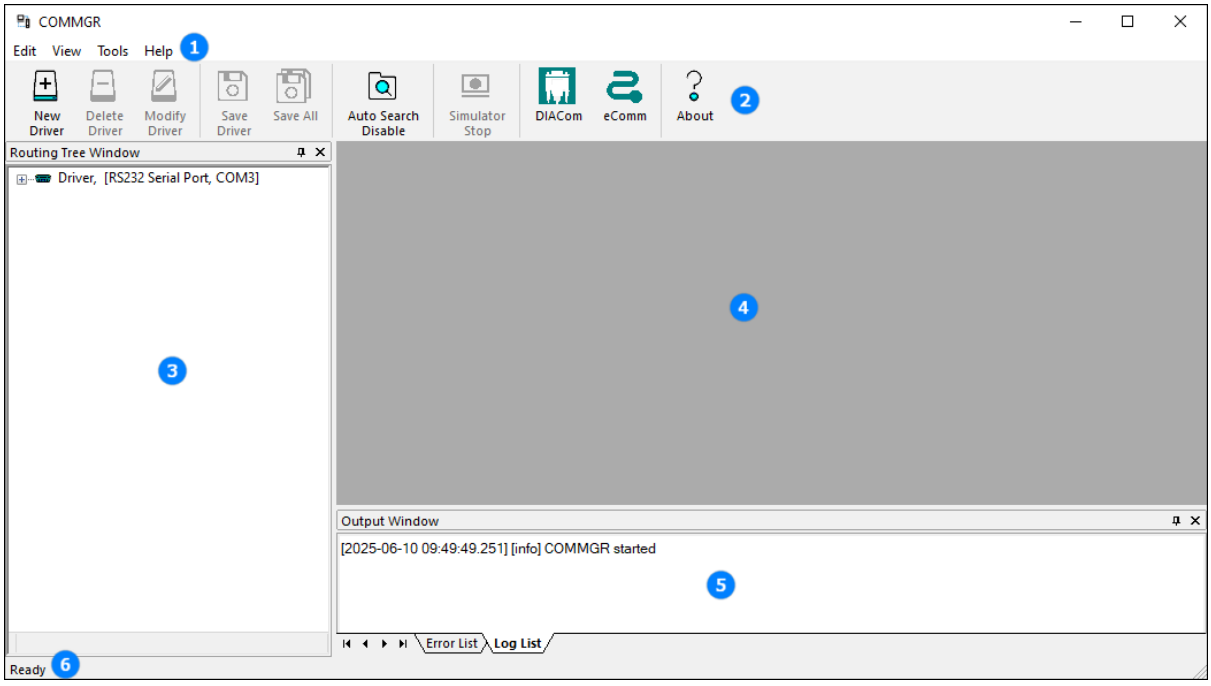
5.3.1 Launch COMMGR

Do one of the following to launch COMMGR.

- Double-click the COMMGR shortcut on the desktop or click the Windows icon at the left-bottom corner and select **Delta Industrial Automation** > **COMMGR** in Windows Start menu.
- Right-click the COMMGR icon in the system tray, and click **Open**.



5.3.2 Main Screen



Legend	Description
1	Menu bar
2	Toolbar
3	Routing Tree window
4	Driver setup page

Legend	Description
⑤	Output Window
⑥	Status bar

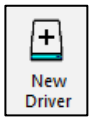
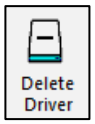
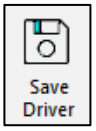
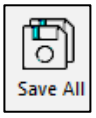
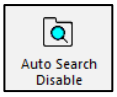
5.3.2.1 Menu Bar

The following table lists the functions in the COMMGR menu bar.

Function	Description
Edit	
New Driver	Click to add new driver.
Delete Driver	Click to delete selected driver.
Modify Driver	Click to modify selected driver.
Save Driver	Click to save the modified driver.
Save All	Click to save all modified drivers.
View	
Routing Tree Window	Click to display / hide the Routing Tree Window.
Output Window	Click to display / hide the Output Window.
Status bar	Click to display / hide the status bar.
Tools	
BOOTP/DHCP Setting Tool	Click to open the BOOTP/DHCP Setting Tool dialog.
IP Setting Tool	Click to open the IP Setting Tool dialog.
Ping Test	Click to open the Ping Test dialog.
DIACom	Click to open DIACom setting tool.
eComm	Click to open eComm software. **Note**: Use eComm with HMI products to connect to DIACloud and set up bypass.
Language	Select the language of software.
Help	
About	Click to view software information.
Software Manual	Click to open software user manual.

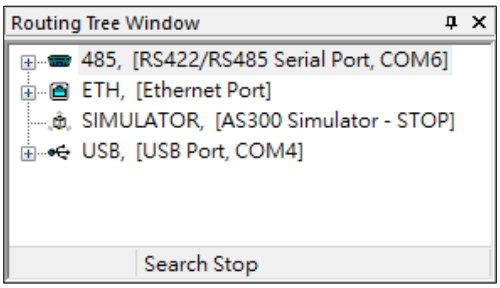
5.3.2.2 Toolbar

The following table lists the functions in the COMMGR toolbar.

Icon	Description
	Click to add new driver.
	Click to delete selected driver.
	Click to modify selected driver.
	Click to save the modified driver setting.
	Click to open the automatic search function.
	Click to turn on the automatic search function.
	Click to turn off the automatic search function.
	Click to start / stop the simulator.
	Click to open the DIACom software.
	Click to open the eComm software. Refer to *eComm User Manual* for operation instructions.
	Click to open **About** dialog.

5.3.2.3 Routing Tree Window

The **Routing Tree Window** displays the created drivers and the driver status.

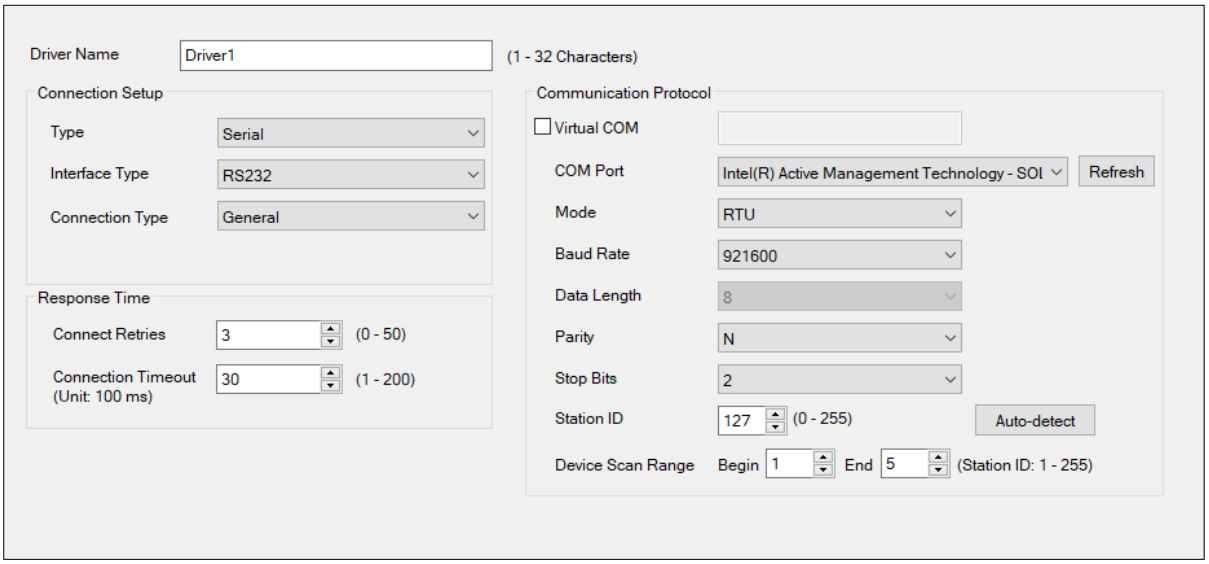


5.3.2.4 Driver Setup Page

The driver setup page displays the driver setting.

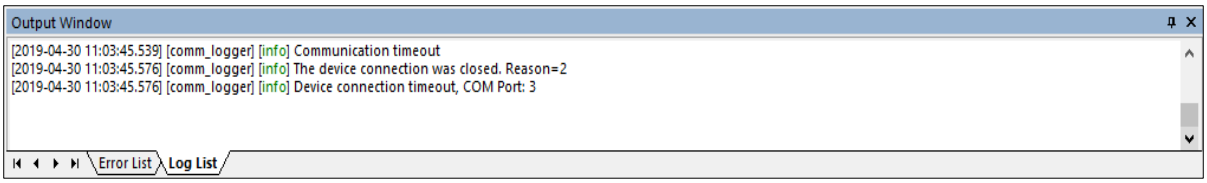
How to open the driver setup page?

- Double-click the driver to be modified in **Routing Tree Window**.
- Click the driver to be modified in **Routing Tree Window**, and click **Modify Driver** on the toolbar.



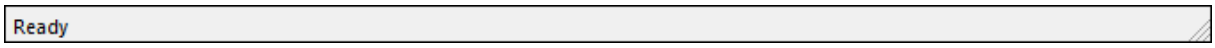
5.3.2.5 Output Window

The **Output Window** displays the **Error List** and **Log List**.



5.3.2.6 Status Bar

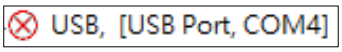
The status bar displays the status of the software.



5.3.3 Driver Setting

The COMMGR lists drivers connected to software and their communication ports. When you add a new driver and set communication parameters for it, COMMGR will create a connection according to the specified communication port specified by that driver.

Whenever the computer restarts, COMMGR also starts the driver automatically. However, if the connectivity of the driver created is lost (for example, the Ethernet Network card or the USB connection cable is removed), COMMGR will stop the driver and display Error state for the driver. Once the connectivity is back, the driver status becomes normal again. The following figure shows an example for error state.



5.3.3.1 Add Driver

Follow these steps to add a driver.

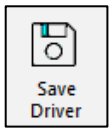
To add driver

1. Click **Edit** > **New Driver** in the menu bar or click 

on the toolbar.
2. Enter **Driver Name** in the driver setup page.
3. In the **Connection Setup** area, select **Type** and set the parameters in the **Communication Protocol** area according to your need.

The following table lists the settings in the driver setup page with their description.

Setting	Description
Driver Name	Enter the driver name. A driver name can be of maximum 32 characters. **Note**: Special characters such as *, #, ?, \, %, @ and so on cannot be used except "_".
Connection Setup – Type	Select the connection type. • Serial—A computer communicates with the Delta controller through a communication port on the computer using RS232 / RS422 / RS485. • USB—A computer can connect to the Delta controller equipped with a USB port through a USB cable. However, make sure that a USB driver is installed on a computer before the computer connects to a controller equipped with a USB port. Refer to Installing USB Driver for AS Series CPU Module for more information about installing controller USB driver. • Ethernet—A computer communicates with the Delta controller through an Ethernet port on the computer. • DIACloud—Establish a computer's virtual Ethernet port (DIACom Ethernet Adaptor) or virtual communication port (DIACom Virtual COM) through DIACom to communicate with devices connected to Delta Cloud. • DirectLink—A communication function provided by Delta Human Machine Interfaces (HMI). If a controller connects to an HMI normally, a computer can connect to the HMI through a USB cable or Ethernet, and connect to the controller indirectly. Refer to the related manuals for Delta's HMI for more information about setting up a connection and precautions for use. • Simulator—A virtual channel connecting to a controller Simulator. If you specify this type of driver in DIADesigner, COMMGR will transmit the data between DIADesigner and a Simulator. It is as if COMMGR is connected to a real controller.
Communication Protocol	Parameters vary from the selected connection type. For the parameter settings of different connections, refer to sections 5.3.3.2~5.3.3.7.
Response Time	Set the parameters of communication response. • Connect Retries—Set the number of times to resend the command when a connection error occurs. • Connection Timeout—Select an interval to resend the command.

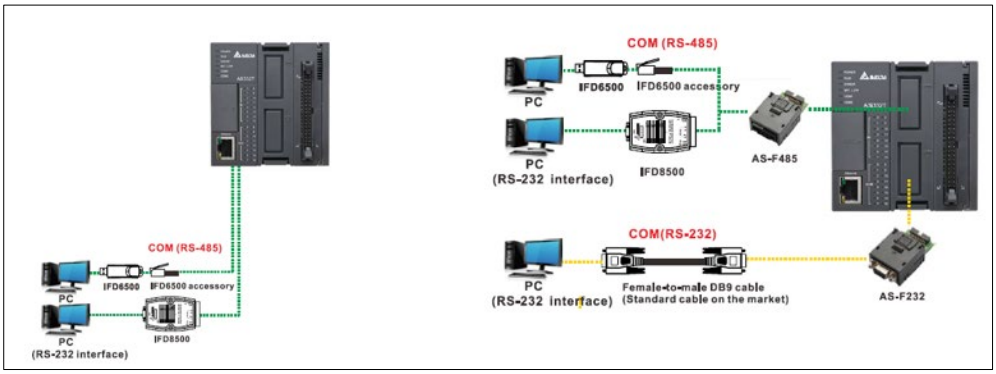
4. Click 

on the toolbar.

   The communication driver is created and displayed in **Routing Tree Window**.

5.3.3.2 RS232 / RS422 / RS485 Setting

The following figure shows connection mode.



Prerequisite

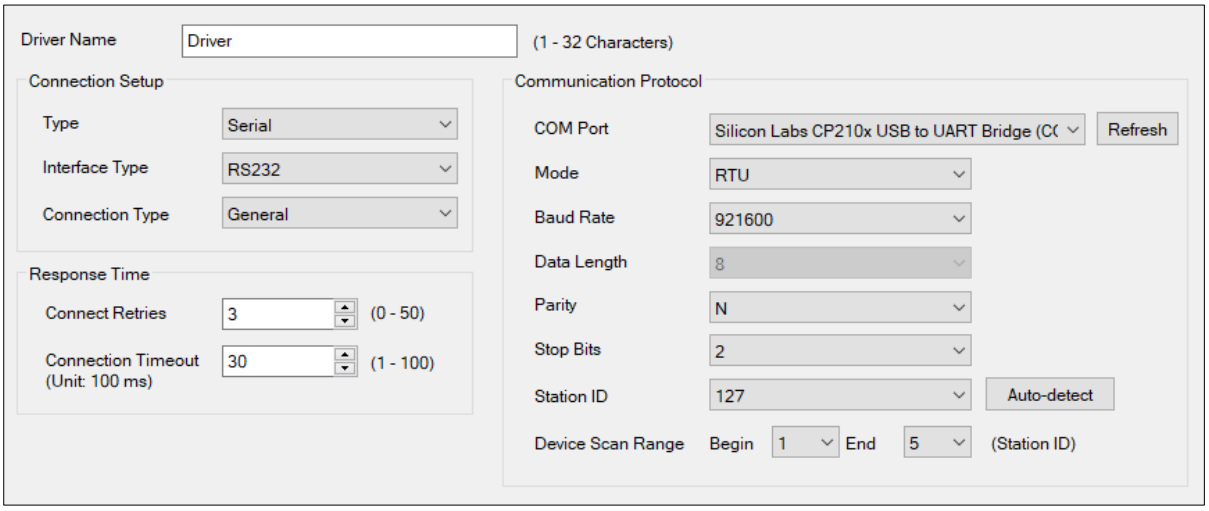
- The related driver is installed on the computer
- The computer and controller are connected and working properly.

Follow these steps to set a RS232 /RS442/ RS485 driver.

To set RS232 /RS442 /RS485 driver

1. Enter a driver name in the **Driver Name** field. Special characters cannot be used except “_”.
2. Select **Serial** from the **Type** drop-down list in the **Connection Setup** area.

3. Select a port you want to establish a connection tunnel from the **COM Port** drop-down list in the **Communication Protocol** area. The complete device name and port number will be displayed, and the content displayed on the port will be the same as the **Windows Device Manager**. You must first confirm the prerequisites to select a COM port.
4. Select **ASCII** or **RTU** from the **Mode** drop-down list.

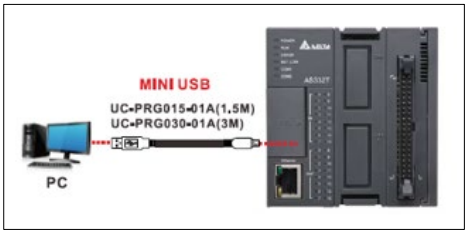


Note:

- The computer port and device port used to exchange data must have the same communication settings.
- When opening the driver setup page, the system captures the COM port information of **Windows Device Manager**, but the information will not remain updating. If a device is added to the system after the driver setup page is opened, the COM Port drop-down list will not have the device option. You need to reopen the driver setup page to have it displayed.

5.3.3.3 USB Setting

The following figure shows connection mode.



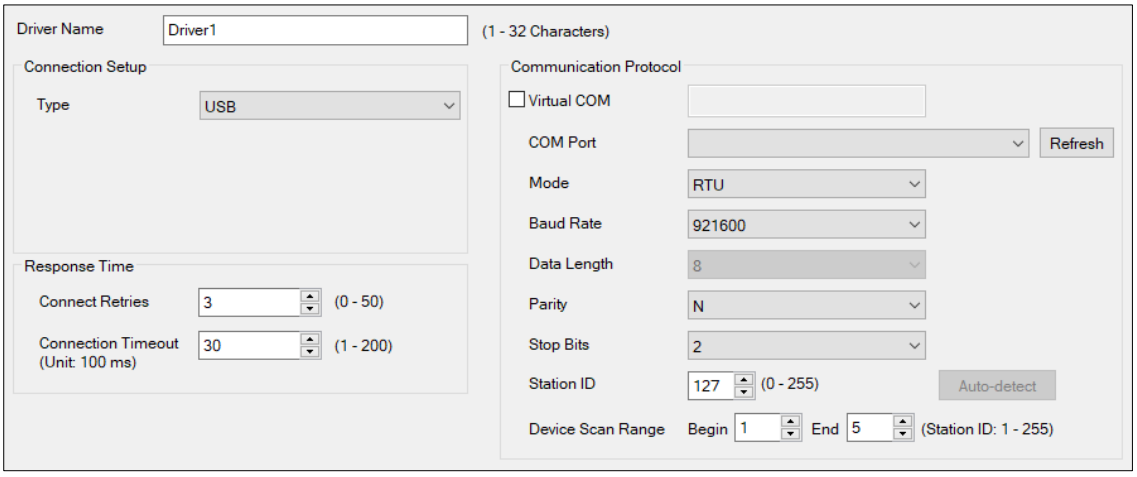
Prerequisites

- A USB driver is installed on the computer.
- The computer and controller are connected and working properly.

Follow these steps to set a USB driver.

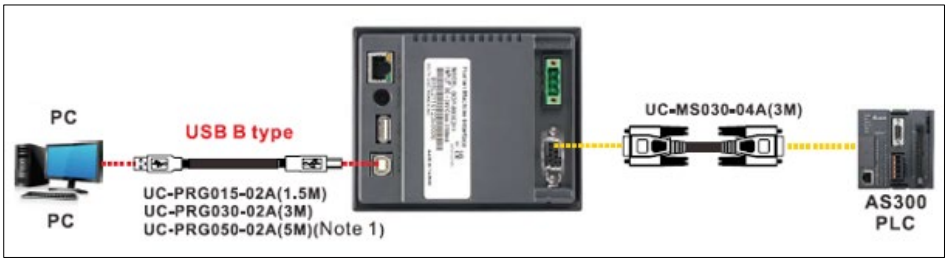
To set USB driver

1. Enter a driver name in the **Driver Name** field. Special characters cannot be used except "_".
2. Select **USB** from the **Type** drop-down list in the **Connection Setup** area.
3. Select a port you want to establish a connection tunnel from the **COM Port** drop-down list in the **Communication Protocol** area. The complete device name and port number will be displayed, and the content displayed on the port will be the same as the **Windows Device Manager**. You must confirm the prerequisites to select a COM port.



Refer to Installing USB Driver for AS Series CPU Module for more information about installing a USB Driver.

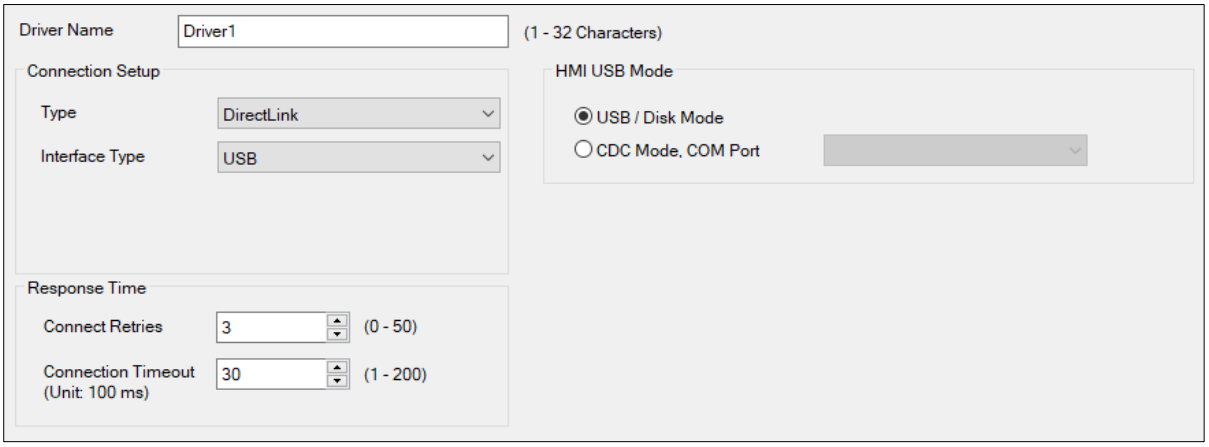
5.3.3.4 DirectLink (USB) Setting



Follow these steps to set up a DirectLink (USB) driver.

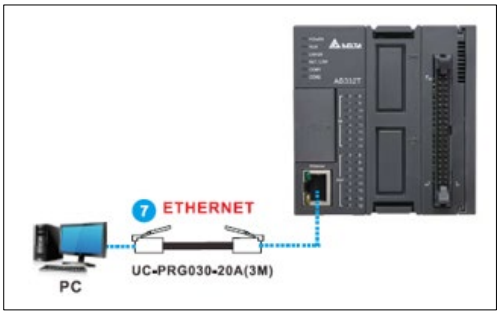
To set DirectLink (USB) driver

1. Enter a driver name in the **Driver Name** field. Special characters cannot be used except "_".
2. Select **DirectLink** from the **Type** drop-down list in the **Connection Setup** area..
3. Select the USB transfer mode of HMI in the **HMI USB Mode** area. If you select **CDC mode**, select the port you want to establish a connection channel from the **COM Port** drop-down list in the **Communication Protocol** area. The complete device name and port number will be displayed, and the content displayed on the port will be the same as the **Windows Device Manager**. For detailed connection settings and precautions for use, refer to the relevant Delta HMI manual.
4. Set the communication parameters in DIAScreen and download the parameters to the HMI to communicate via DirectLink (USB).



Note: When using DirectLink, the communication from HMI to the controller does not support RTU mode, do check the related settings in DIAScreen.

5.3.3.5 Ethernet Setting

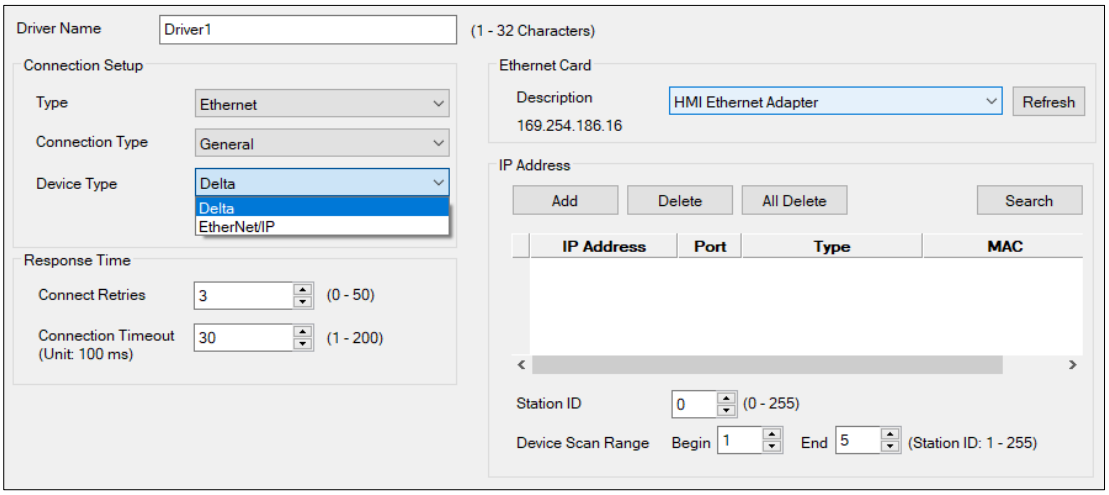


Follow these steps to set an Ethernet driver.

To set Ethernet driver

1. Enter a driver name in the **Driver Name** field. Special characters cannot be used except "_".
2. Select **Ethernet** from the **Type** drop-down list in the **Connection Setup** area.
3. Select **General** from the **Connection Type** drop-down list.
4. Select **Delta** or **EtherNet/IP** from the **Device Type** drop-down list.
   - Delta: Search for Delta devices.
   - EtherNet/IP: Search for EtherNet/IP devices.
5. Select a Network Interface Card (NIC) from the **Description** drop-down list in the **Ethernet Card** area. An IP address assigned to a network interface card selected displays in the lower left corner.
6. Owing to the characteristics of Ethernet, a computer can communicate with all devices on a network. You can create the IP addresses of devices connected to this driver in the **IP Address** area. Follow these steps to create a device IP.
   a. Some devices support the **Search** function. After you click **Search** to search for IP addresses, results display in the list.
   b. Click **Add** to add new IP address in the list.
      - Enter the IP address of a device connected in **IP Address** field.
      - Enter the communication port number specified. If a communication port number is not specified, use the default communication port number **502**.

- The **Type** field is used to display the searched device type when using the **Search** function and is not editable.
- Select a particular IP and click **Delete** or press **Delete** key to delete it.



Note:

- When you select **EtherNet/IP** for **Device Type** to search, you cannot perform a device scan for the searched device.
- When opening the driver setup page, the system captures the Ethernet card information, but the information will not remain updating. If an Ethernet card is added to the system after the driver setup page is opened or the Ethernet card setting is changed, the Ethernet card from the Description drop-down list will not be updated. You can click **Refresh** to refresh the Ethernet card.

5.3.3.6 DirectLink (Ethernet) Setting

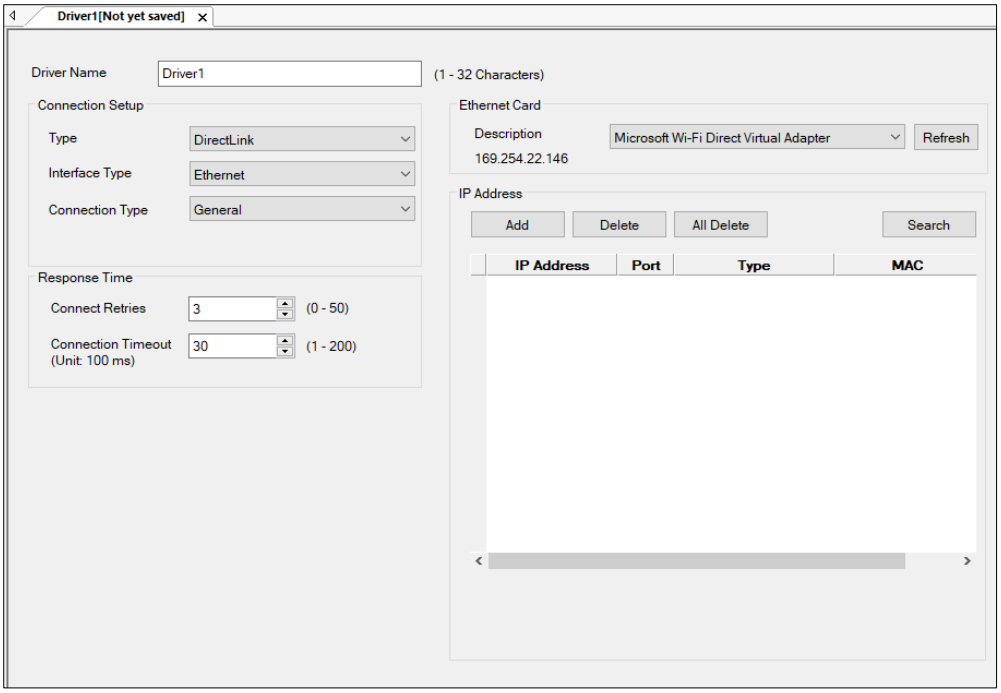


Follow these steps to set a DirectLink (Ethernet) driver.

To set DirectLink (Ethernet) driver

1. Enter a driver name in the **Drive Name** field. Special characters cannot be used except “_”.

2. Select **DirectLink** from the **Type** drop-down list in the **Connection Setup** area, and then select **Ethernet** for **Interface Type**.
3. Select an Ethernet card from the **Description** drop-down list in the **Ethernet Card** area. An IP address assigned to a network interface card selected displays in the lower left corner.
4. Owing to the characteristics of Ethernet, a computer can communicate with all devices on a network. You can create the IP addresses of devices connected to this driver in the **IP Address** area. Follow these steps to create a device IP.
   a. Delta HMI devices support the **Search** function. After you click **Search** to search for IP addresses, results display in the list.
   b. Click **Add** to add new IP address in the list.
      - Enter the IP address of a device connected in **IP Address** field.
      - Enter a specified communication port number. If a communication port number is not specified, use the default communication port number **502**.
      - The **Type** field has no effect in setting **DirectLink (Ethernet)** parameters.
5. Select a particular IP and click **Delete** or press **Delete** key to delete the it.
6. Set the communication parameters in DIAScreen and download the parameters to the HMI to communicate via DirectLink (USB).



Note:

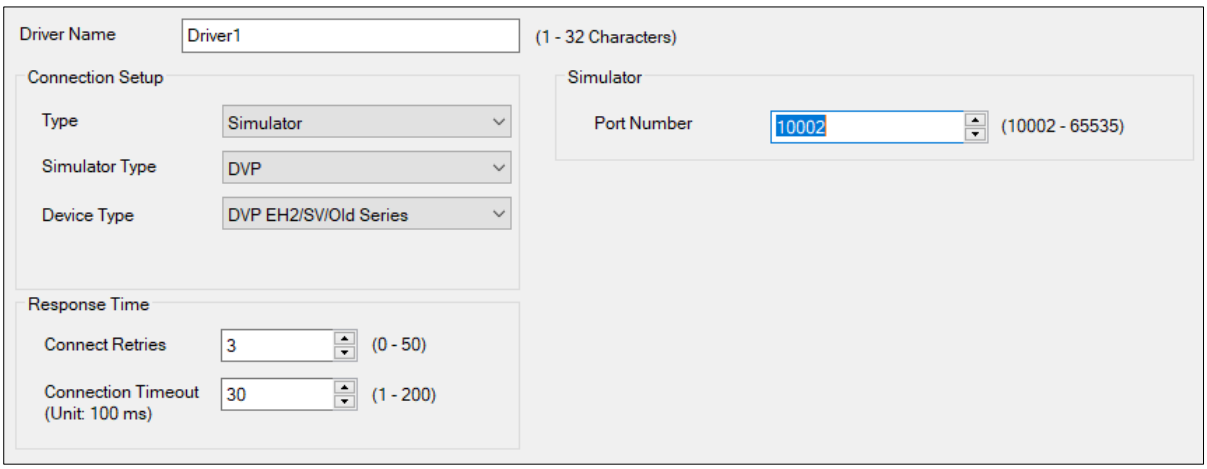
- When opening the driver setup page, the system captures the Ethernet card information, but the information will not remain updating. If a device is added to the system after the driver setup page is opened or the Ethernet card setting is changed, the Ethernet card will not be updated. You need to reopen the driver setup page to update it.
- When using DirectLink, the communication from HMI to controller does not support RTU mode, do check the related settings in DIAScreen.

5.3.3.7 Simulator Setting

Follow these steps to set a Simulator driver.

To set Simulator driver

1. Enter a driver name in the **Driver Name** field. Special characters cannot be used except "_".
2. Select **Simulator** from the **Type** drop-down list in the **Connection Setup** area.
3. Select the controller model from the **Simulator Type** drop-down list.
4. Enter a specified communication port number in the **Port Number** field in the **Simulator** area. If a communication port number is not specified, use the default communication port number.



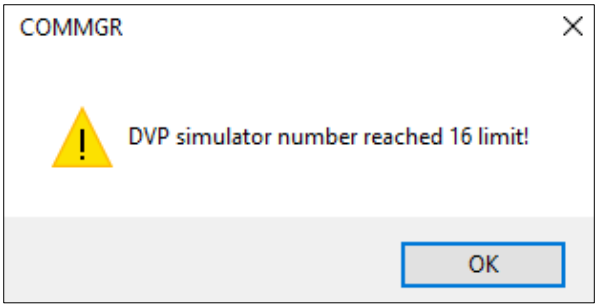
Note: You can create several Simulator drivers. But there are limitations in run mode.

When creating multiple types of simulators, their communication port number must not be repeated. A maximum of the following simulators can run at same time.

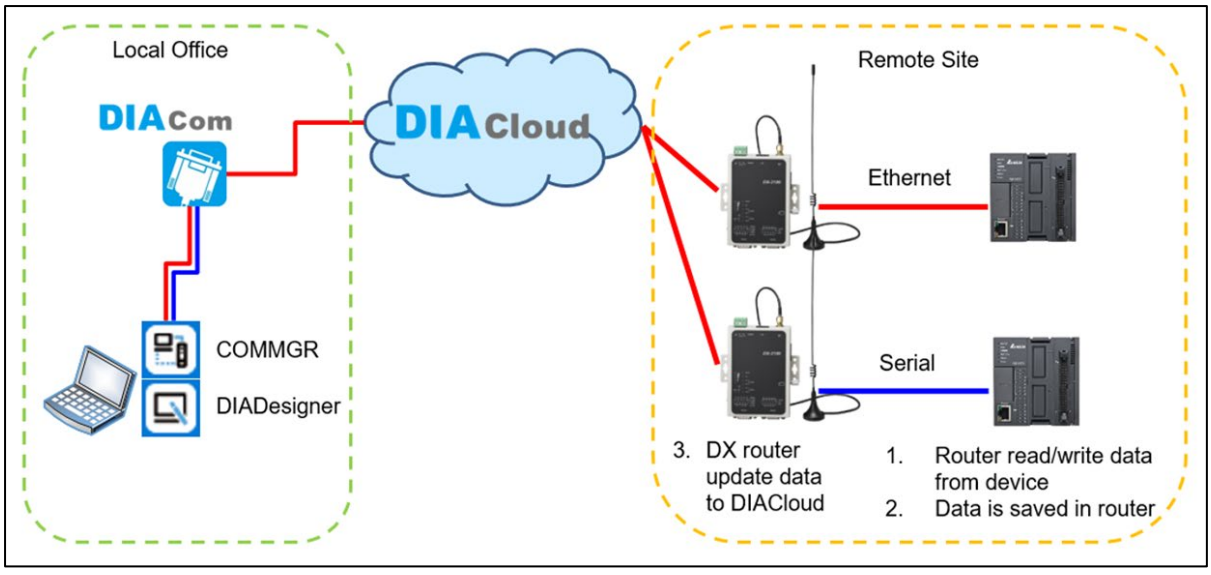
- 4 AS300 simulators
- 4 AS200 simulators
- 4 AS100 simulators
- 4 AH simulators
- 16 DVP simulators

  (Total 32 simulators can run at the same time.)

When the maximum limit for creating drivers of a particular type is reached, an alarm message appears.



5.3.3.8 Support DIACloud Function



Prerequisites

- The DX Ethernet cloud router equipment environment is established.
- A DIACom account is created and the DX device is bound to DIACloud.

For connection settings and precautions for use, refer to the related manuals for Delta DX.

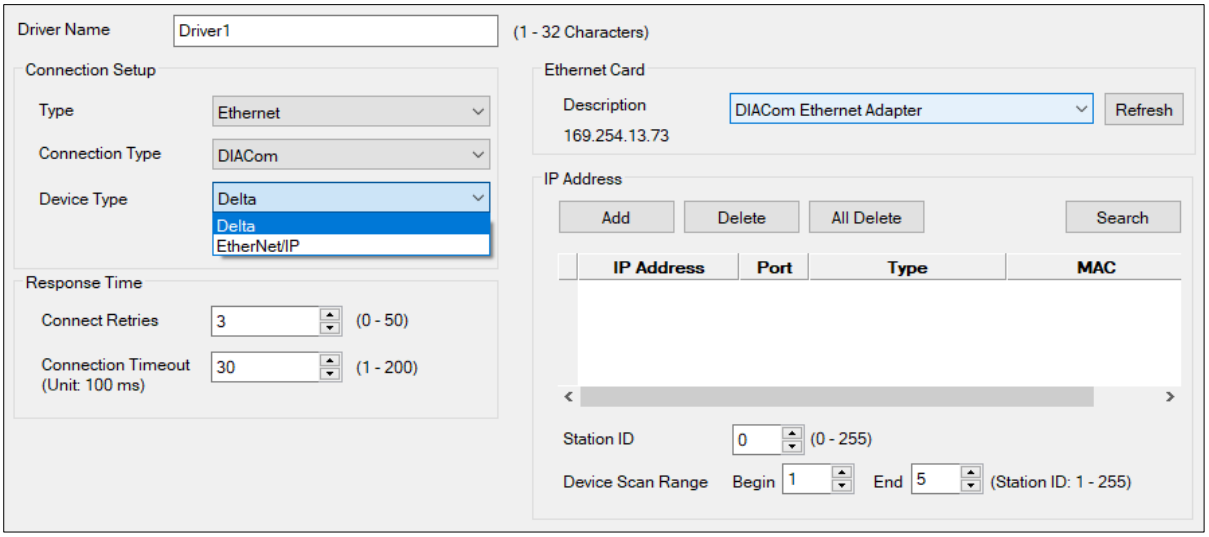
Follow these steps to add a DIACom (Ethernet) driver.

To add DIACom (Ethernet) driver

1. Enter a driver name in the **Driver Name** field. Special characters cannot be used except "_".
2. Select **Ethernet** from the **Type** drop-down list in the **Connection Setup** area.
3. Select **DIACom** from the **Connection Type** drop-down list.
4. Select **Delta** or **EtherNet/IP** from the **Device Type** drop-down list.
   - Delta: Search for Delta devices.
   - EtherNet/IP: Search for EtherNet/IP devices
5. Select the virtual network card (**DIACom Ethernet Adapter**) created by DIACom from the **Description** drop-down list in the **Ethernet Card** area. An IP address assigned to the selected Ethernet card is displayed in the lower left corner. If the Ethernet card has not been created when it is turned on, click **Refresh** after the creation to select the correct Ethernet card.
6. Owing to the characteristics of Ethernet, a computer can communicate with all devices on a network. Create the IP addresses of devices connected to this driver in the **IP Address** area. Follow these steps to create a device IP.
   a. Some devices support Search function. After clicking **Search** to search for IP addresses, results are displayed in the list. If you have not used DIACom to open the channel, a prompt message appears and you can directly open DIACom to establish the channel.
   b. Click **Add** to add new IP address in the list.
      - Enter the IP address of a device connected in **IP Address** field.
      - Enter the specified communication port number. If a communication port

number is not specified, use the default communication port number **502**.

- The **Type** field is used to display the searched device type when using the **Search** function and is not editable.
- Select a particular IP and click **Delete** or press **Delete** key to delete it.



Note:

- Using COMMGR to perform DIACom search will use network traffic.
- When opening the driver setup page, the system captures the Ethernet card information, but the information will not remain updating. If a device is added to the system after the driver setup page is opened or the Ethernet card setting is changed, the Ethernet card will not be updated. You can click **Refresh** to refresh the Ethernet card.

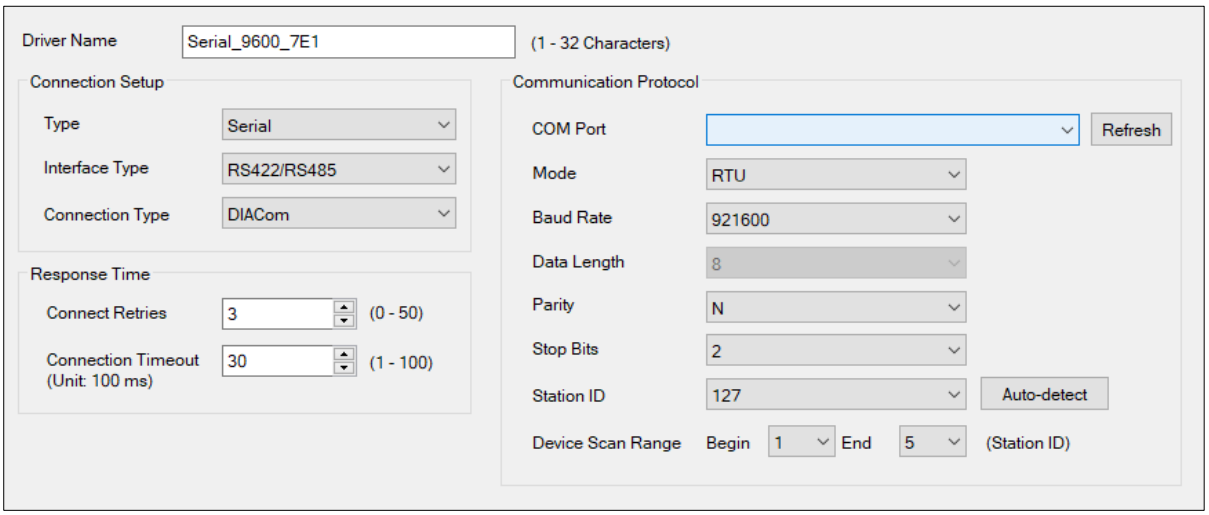
Follow these steps to add a DIACom (Serial) driver.

To add DIACom (Serial) driver

1. Enter a driver name in the **Driver Name** field. Special characters cannot be used except "_".
2. Select **Serial** from the **Type** drop-down list in the **Connection Setup** section.
3. Select **RS232** or **RS422/RS485** from the **Interface Type** drop-down list.
4. Select **DIACom** from the **Connection Type** drop-down list.
5. Select the port you want to establish a connection tunnel from the **COM Port** drop-down list. When a virtual communication port has not been established, a

prompt message appears and asks you whether to open DIACom to establish a virtual communication port, and then you can open DIACom to establish a virtual communication port. After the establishment is completed, click **Refresh** in the **Communication Protocol** area, and the complete device name (ELTIMA Virtual Serial Port) and port number will be displayed in the drop-down list in the **COM Port** field.

6. Select **ASCII** or **RTU** from the **Mode** drop-down list.



Note:

- The computer port and device port used to exchange data must have the same communication settings.
- Using COMMGR to perform DIACom search will use network traffic.

5.3.4 Driver Start/Stop

When the device is successfully connected to the computer, COMMGR will create the connection automatically; additional operation is not required. You need to set the start or stop mode only when **Simulator** is selected for **Type**.

The following section takes **AS300 Simulator Driver** as an example.

5.3.4.1 Start the Driver

Right-click the AS300 Simulator driver in **Routing Tree Window** and select **Start**. The Simulator driver starts when the status becomes **Start** from **Stop**.

5.3.4.2 Stop the Driver

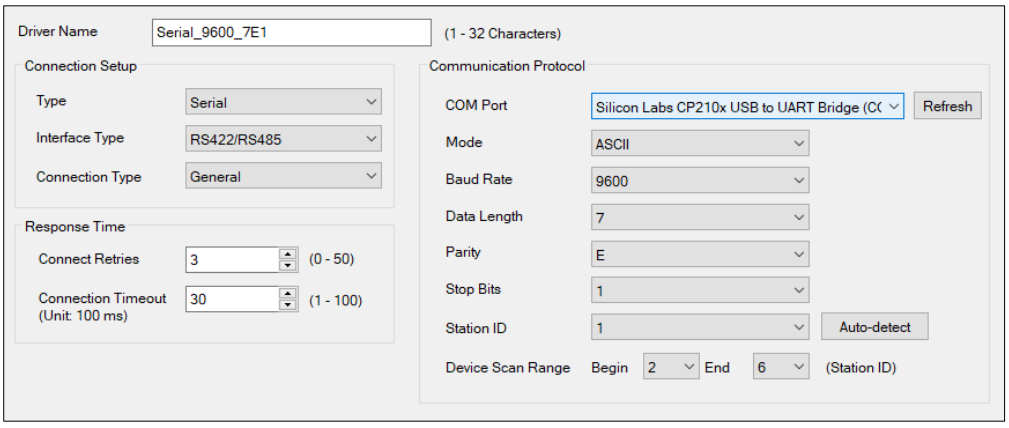
Right-click the AS300 Simulator driver in **Routing Tree Window** and select **Stop**. The Simulator driver stops when the status becomes **Stop** from **Start**.

5.3.5 Driver Search

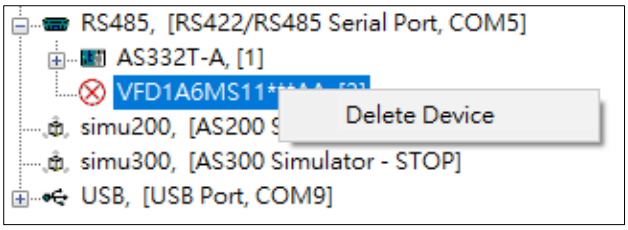
When the device is connected to the computer normally, you can search for connected devices in COMMGR.

5.3.5.1 Execute Search

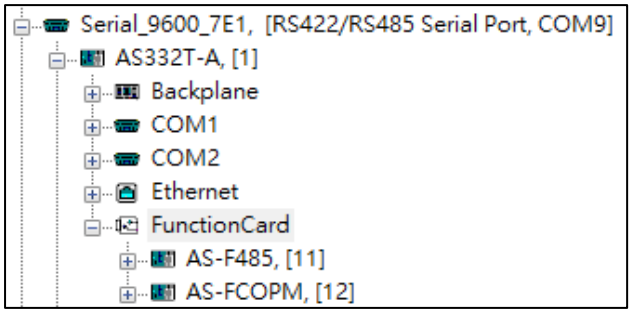
If the created driver is normally connected to the computer, click  on the left of the driver in **Routing Tree Window** to search for the device connected to this driver. The device scan range can be modified to the settings in the driver as shown in the following figure. The modified setting takes effect after saving the driver.



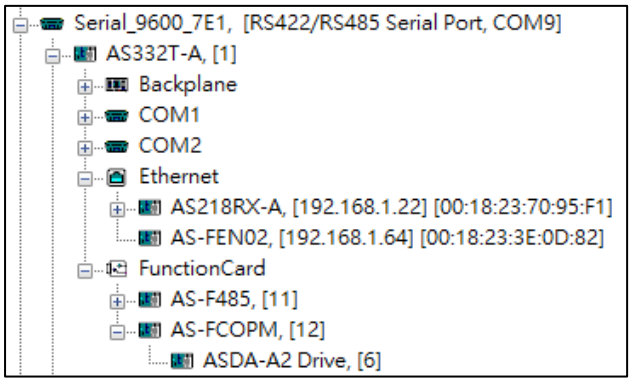
The searched devices will remain in the list when the device is offline. When searching again, an error symbol  will be displayed on the left of the device, indicating that the device is not connected normally. You can check the connection status and search again, or right-click the device and select **Delete Device**.



COMMGR can also perform device scan, which means that the computer can forward search commands through the controller to the equipment connected to the controller. Directly set the controller and its connected devices without changing the field wiring. When the controller (such as: AS332T) supports the forwarding function, expand the controller node to continue searching. Take AS332T as an example. As shown in the following figure, expand RS485 (COM1, COM2, Function Card), Ethernet or CANopen nodes, and click  on the left each node to search for connected devices.

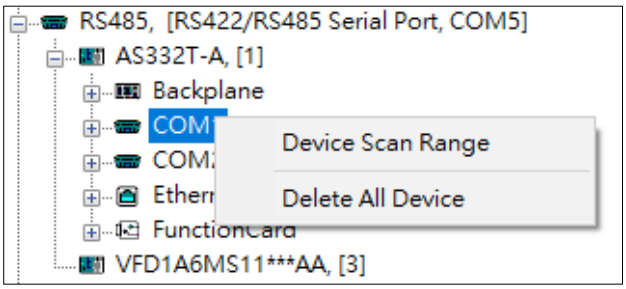


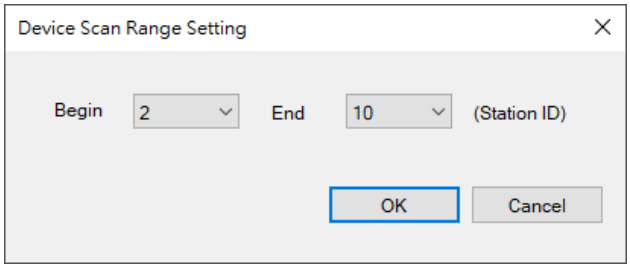
The device scan result is shown below.



Note: If the driver is Serial Port, the device scan of the same Serial Port cannot be performed; when the driver is Ethernet, the Ethernet device scan cannot be performed anymore.

Right-click each node and select **Device Scan Range**. After modify the device scan range in the **Device Scan Range Setting** dialog, click **OK** to take effect.



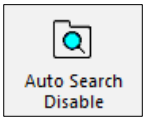


5.3.5.2 Automatic Search

When you need to continuously search for the status of the online device, click 

on the toolbar to enable the automatic search function (the icon will be grayed out and cannot be clicked, and then click  on the left of the driver in **Routing Tree Window** to continue searching the device status of this driver.

When a new device is connected, the new device will be displayed in **Routing Tree Window**; when the device is not connected normally, an error symbol  will be displayed on the left of the device. At this time, check the connection status and perform the search again, or right-click the device in the Routing Tree and select Delete Device this device.

When you want to stop to continuously search for the status of the device, click 

on the toolbar to disable the automatic search function.

5.3.6 Delete and Modify Drivers

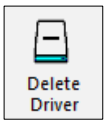
5.3.6.1 Delete Driver

Follow these steps to delete the existing driver.

To delete existing driver

1. Click the driver in the **Routing Tree window**.
2. Do one of the following to delete the driver:
   - In the menu bar, click **Edit** > **Delete Driver**.

   Or

   - Click 

on the toolbar.

   Or

   - Right-click the driver and select **Delete Driver**.

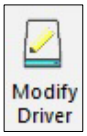
5.3.6.2 Modify Driver

Follow these steps to modify the existing driver.

To modify existing driver

1. Click the driver in **Routing Tree Window**.
2. Do one of the following:
   - In the menu bar, click **Edit** > **Modify Driver**

   Or

   - Click 

on the toolbar.

   Or

   - Right-click the **Driver** and select **Modify Driver**.

3. Modify the relevant parameters in the driver setup page according to your need. For more information, refer to Driver Setting.

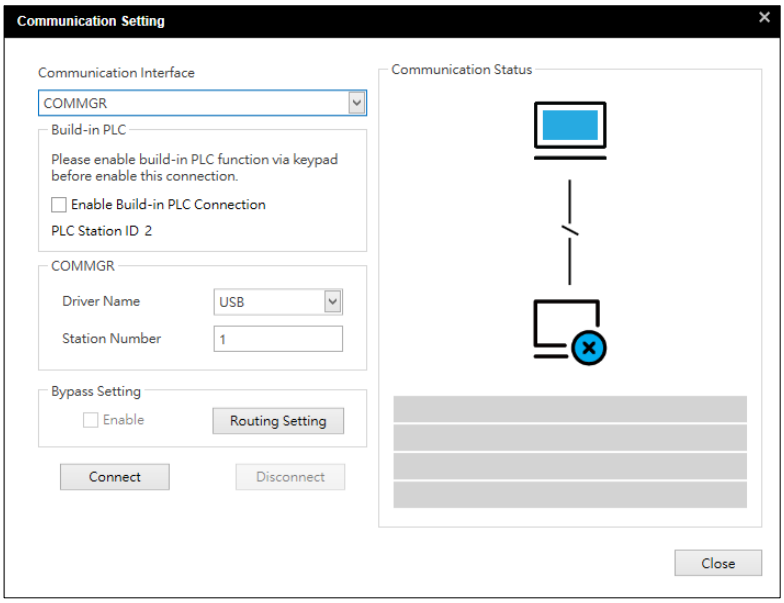
4. Click 

on the toolbar to save the modified parameters.

5.3.7 Communication Setting of Inverter with Built-in PLC

The Communication setting of inverters can be connected via **COMMGR** or **Serial Port**.

- **Connect via COMMGR**



Follow these steps to set up communication.

1. Create a driver in COMMGR and scan the station number of the inverter.

   Note: Since both the inverter and the built-in controller have a station number, you must specify the station number of the inverter.

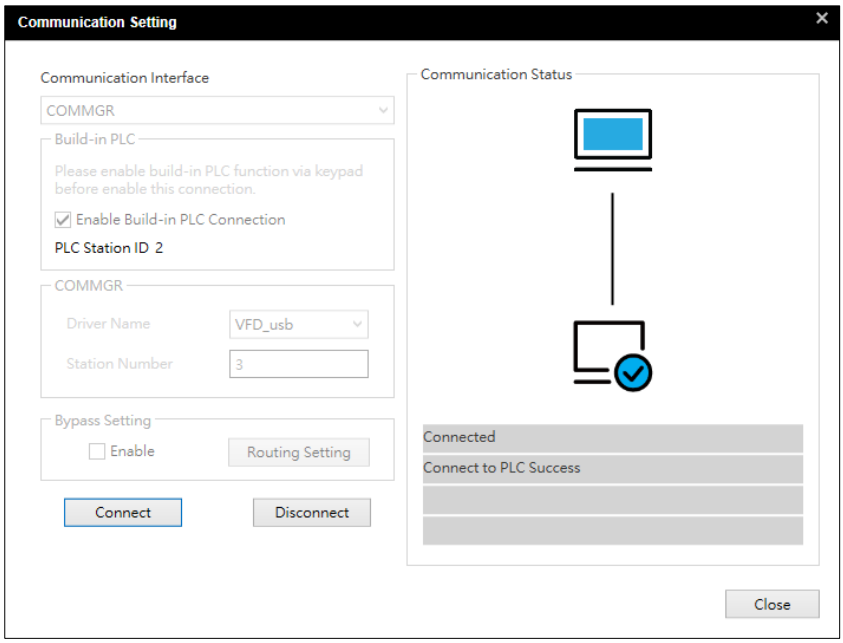
2. Add an inverter device in DIADesigner.

3. Double-click the inverter > **Communication Setting** in **Project Explorer**.

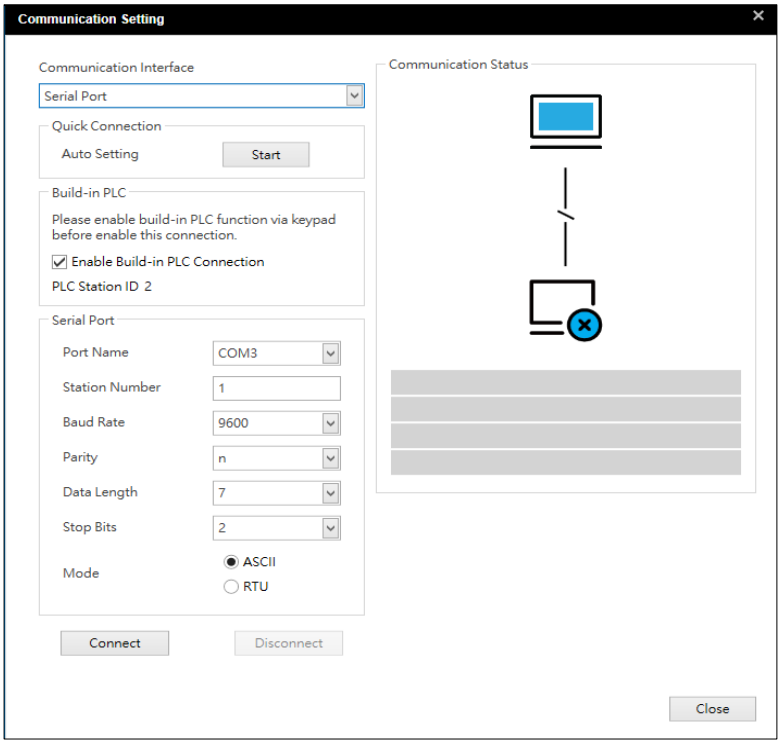
4. Select **COMMGR** from the **Communication Interface** drop-down list in the **Communication Setting** dialog, and then select **Driver Name** and **Station Number**.

5. Enable the built-in PLC functions on the operation panel before enabling the built-in PLC in the inverter.
6. Select the **Enable Build-in PLC Connection** checkbox in the **Build-in PLC** area, and then click **Connect**.

   PLC is connected successfully.



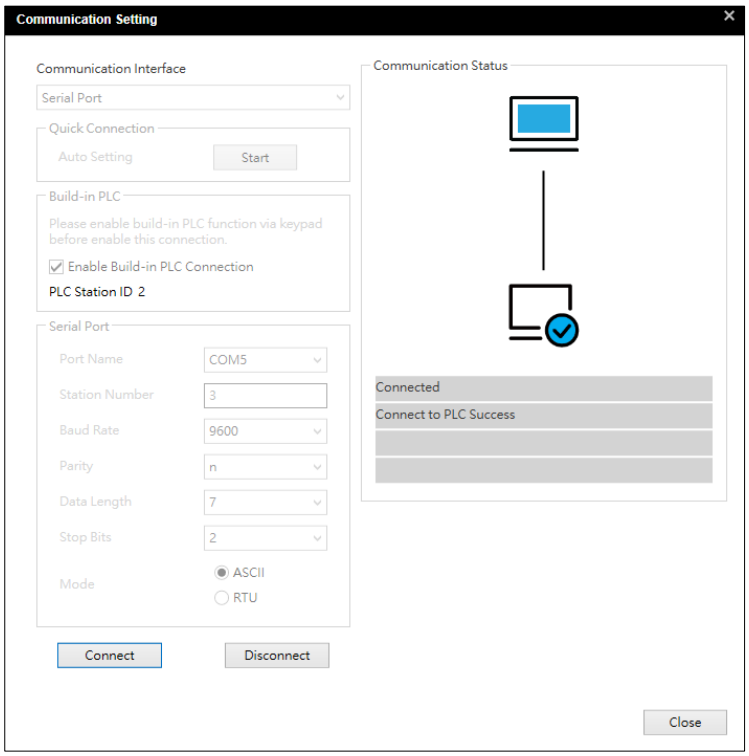
- **Connect via Serial Port**



Follow these steps to set up communication.

1. Add an inverter in DIADesigner.
2. Click the inverter > **Communication Setting** in **Project Explorer**.
3. Select **Serial Port** from the **Communication Interface** drop-down list in the **Communication Setting** dialog, and then select the serial port or click **Start** in the **Quick Connection** area.
4. Enable the built-in PLC functions on the operation panel before enabling built-in PLC in the inverter.
5. Select the **Enable Build-in PLC Connection** checkbox in the **Build-in PLC** area, and then click **Connect**.

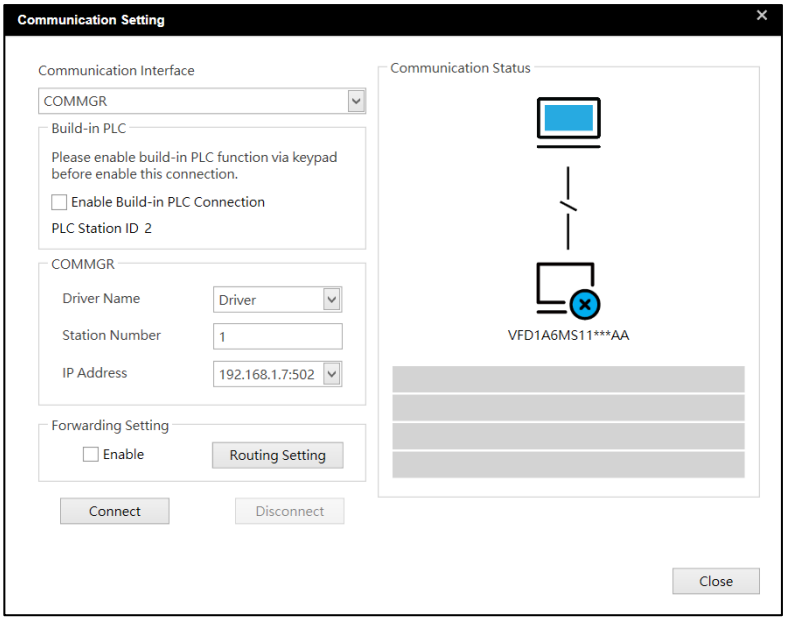
   PLC is connected successfully.



5.3.8 Communication Setting of Inverter with Communication Card

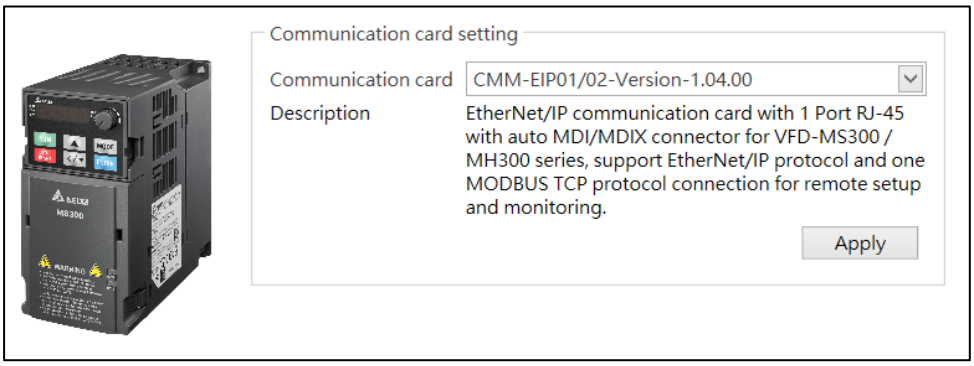
The communication setting of the inverter with communication card can be connected via COMMGR and Ethernet.

- **Connect via COMMGR**



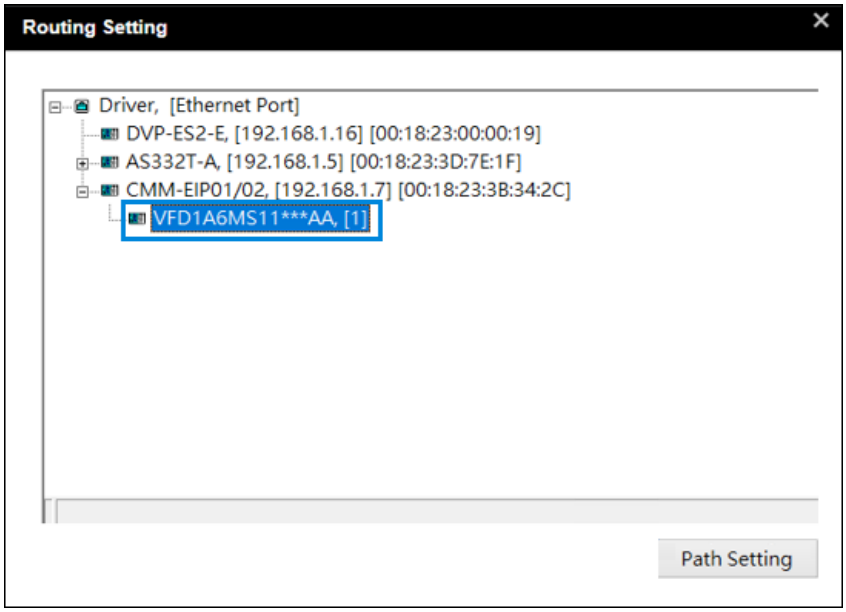
Follow these steps to set up communication.

1. Create a driver in COMMGR.
2. Add an inverter device in DIADesigner.
3. Click the inverter > **VFD Function** > **Hardware Configuration** > **Configuration** in **Project Explorer**.
4. Select CMM-EIP01/02 communication card from the **Communication card** drop-down list in the **Communication card** setting area, and then click **Apply**.



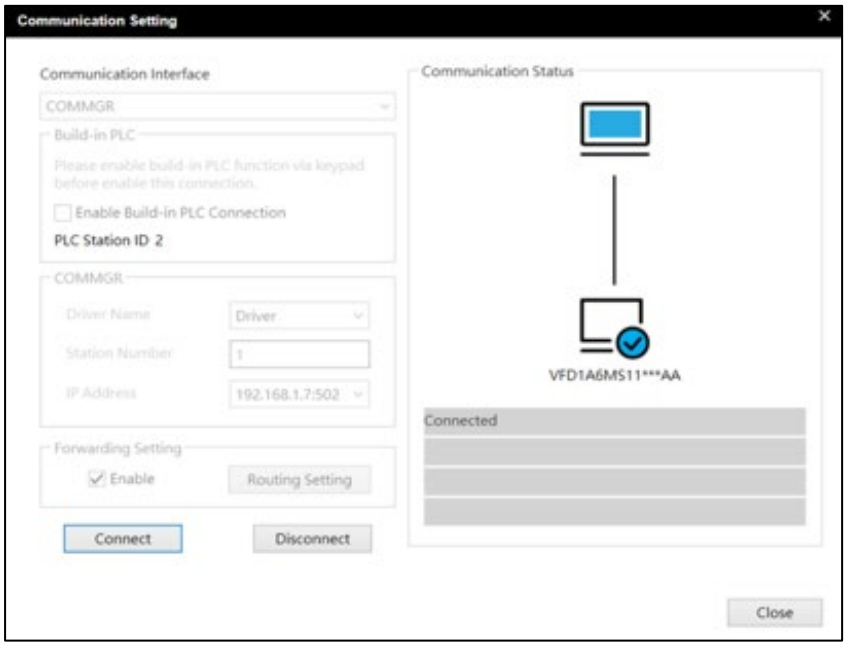
5. Click the inverter > **Communication Setting** in **Project Explorer**.
6. Select **COMMGR** from the **Communication Interface** drop-down list in the **Communication Setting** dialog, and then click **Routing Setting**.

7. Select the corresponding device and click **Path Setting** to complete the setting.

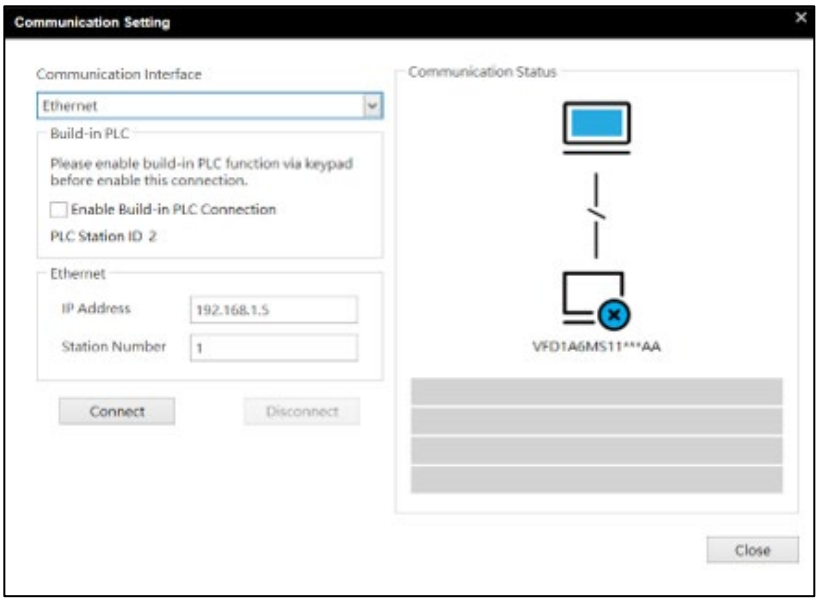


8. Select the **Enable** checkbox in the **Forwarding Setting** area and click **Connect**.

   The connection between DIADesigner and the inverter is successful.

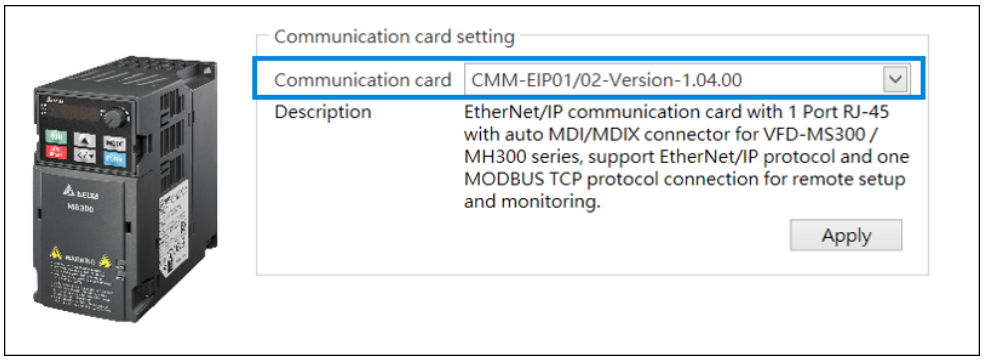


- **Connect via Ethernet**

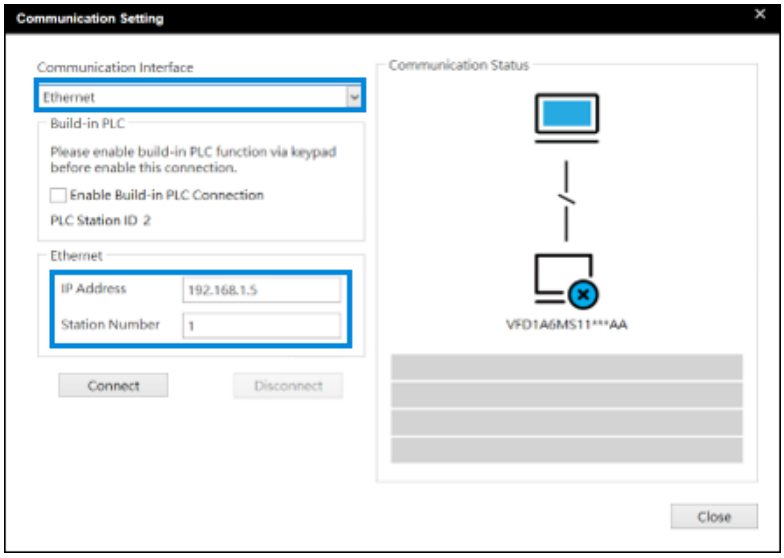


Follow these steps to set up communication.

1. Create a driver in COMMGR.
2. Add an inverter device in DIADesigner.
3. Click the device > **VFD Function** > **Hardware Configuration** > **Configuration** in **Project Explorer**.
4. Select CMM-EIP01/02 communication card from the **Communication Card** drop-down list in the **Communication card setting** area, and then click **Apply**.

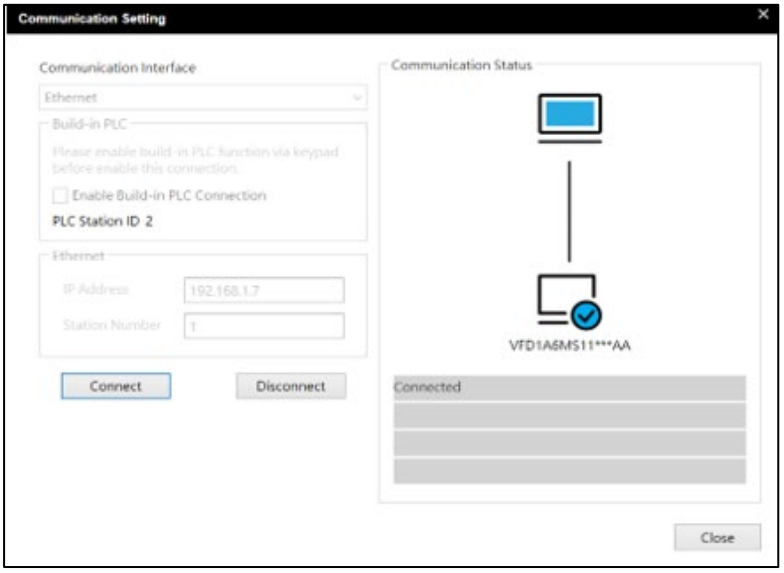


5. Click the inverter > **Communication Setting** in **Project Explorer**.
6. Select **Ethernet** from the **Communication Interface** drop-down list in the **Communication Setting** dialog and enter the **IP address** and **Station Number** of the inverter.



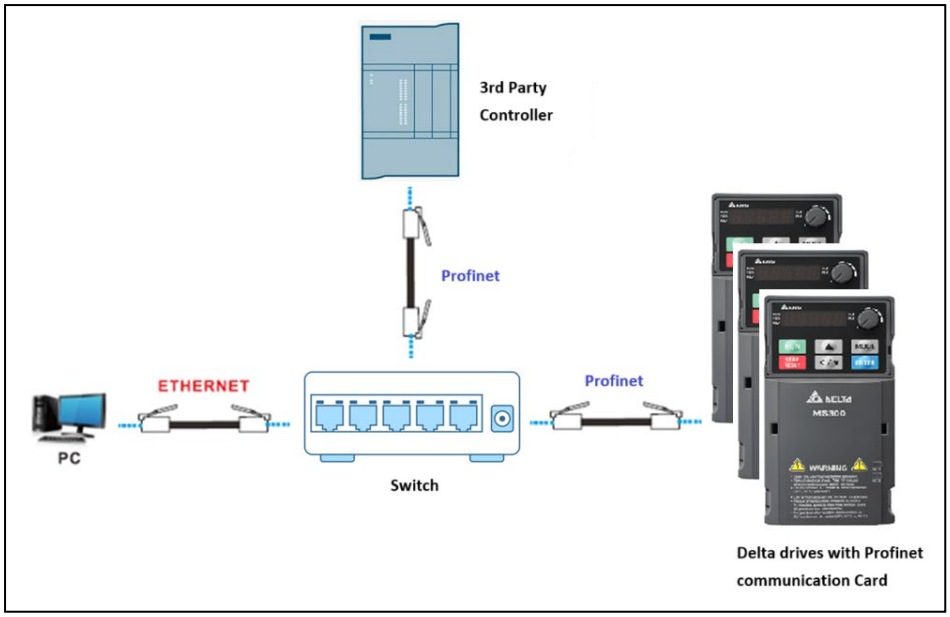
7. Click **Connect**.

   The inverter is connected successfully.



5.3.9 Inverter PROFINET Direct Connection Function

When using a third-party controller to connect a Delta inverter through PROFINET fieldbus, DIADesigner can communicate with the Delta inverter through a switch machine, realizing the convenience of setting up the Delta device using the fieldbus.



Follow these steps to set up communication.

1. Refer to Ethernet Setting to create an Ethernet driver in COMMGR and scan the Delta inverter via switch machine as shown in the above figure.
2. Add an inverter in DIADesigner.
3. Double-click the inverter > **Communication Setting** in **Project Explorer**.
4. Select the Ethernet drive from the **Communication Interface** drop-down list in the **Communication Setting** dialog and enter the **Station Number** and **IP address** of the inverter, and then click **Connect** to perform subsequent operations on the inverter in DIADesigner.

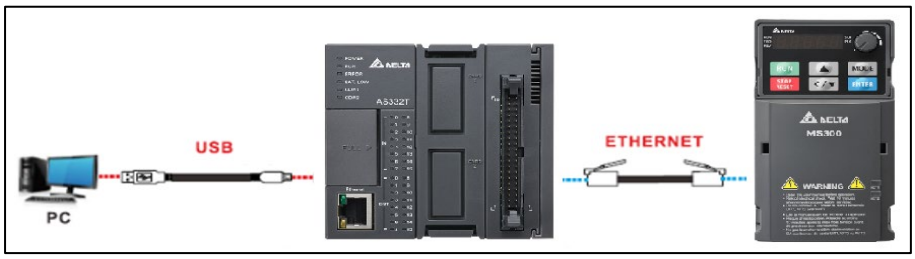
Note: This function supports only the M300 series inverter and function card. For the relevant firmware version, please contact Delta.

5.4. One Wire Scenario – Forwarding Function and Routing Setting

For the convenience in directly connecting to the terminal device, the forwarding function of DIADesigner is provided for forwarding the communication through the controller to the next lower device.

Note: At present, USB, RS485, and Ethernet are supported.

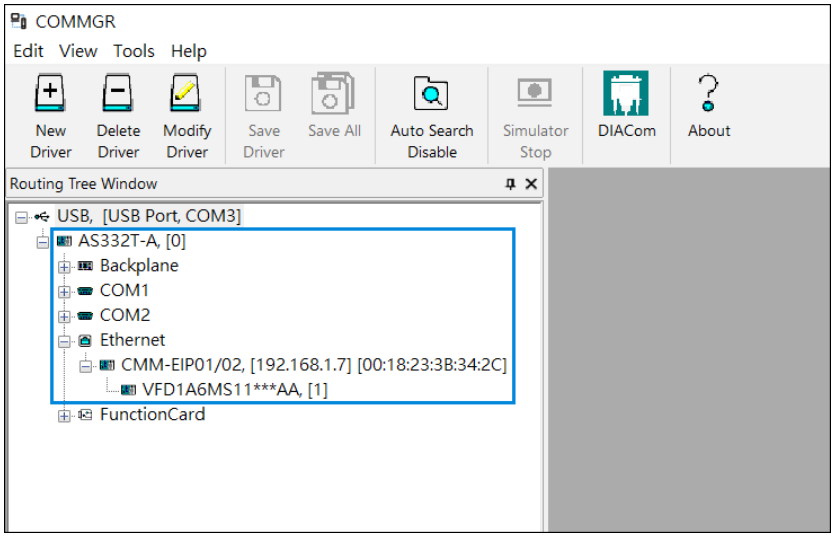
The following example shows the computer side is connected to the controller through the USB port, and then connected to the target device from the Ethernet port of the controller.



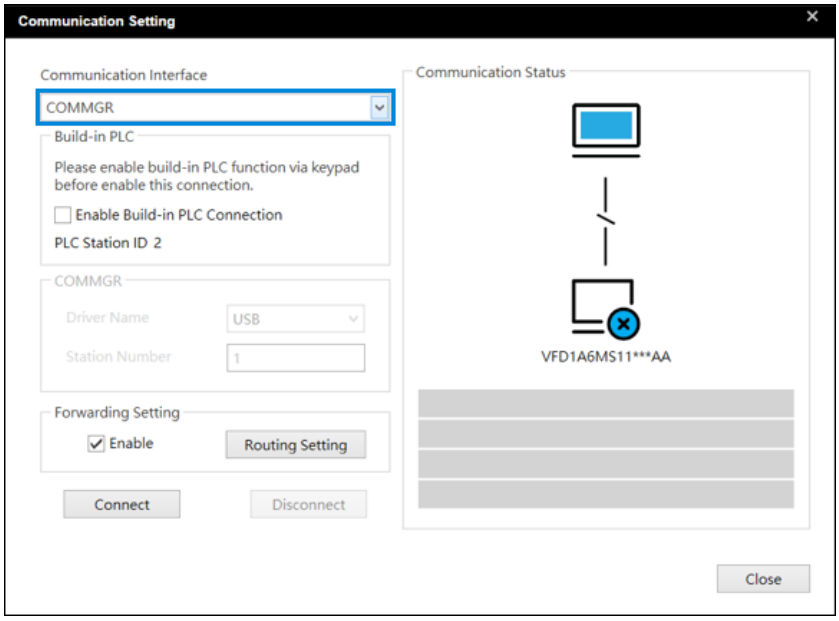
Take AS332T-A controller and VFD-MS300 AC motor drive as an example to perform the bypass settings between the devices.

Follow these steps to perform the bypass setting.

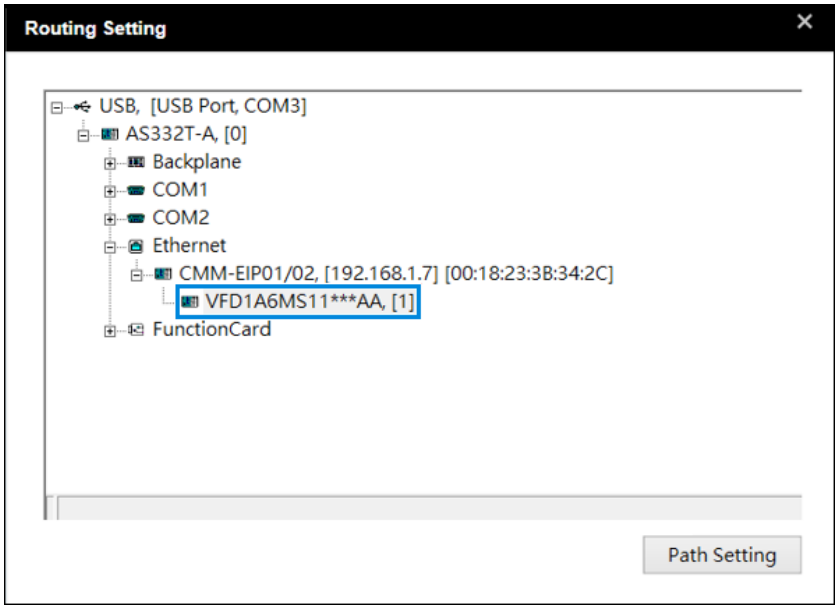
1. Create a USB driver in COMMGR.
2. Click ⊞ on the left of the USB driver in **Routing Tree Window** to scan, and the device VFD1A6MS11***AA can be scanned.



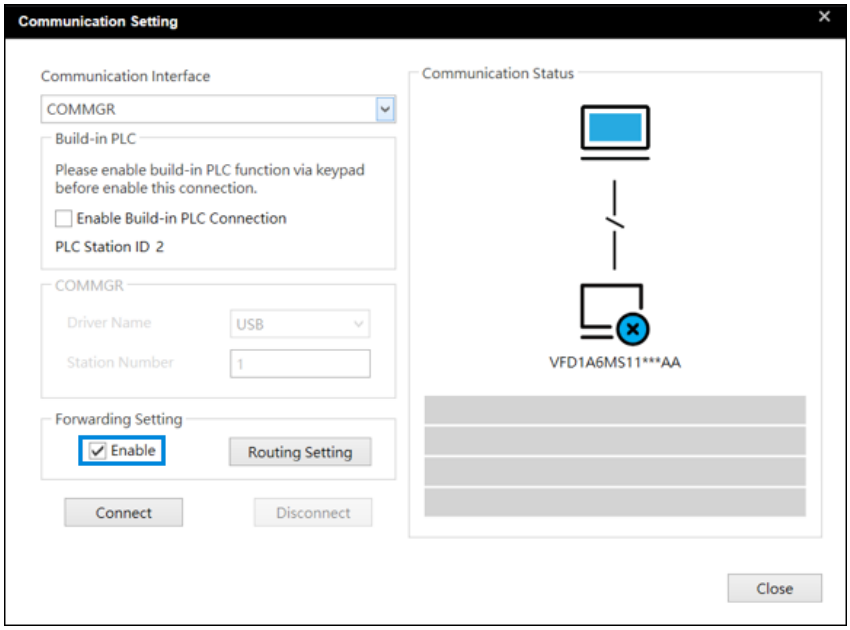
3. Create the AS332T-A controller and VFD1A6MS11***AA AC motor drive in DIADesigner.
4. Double-click **VFD1A6MS11***AA** > **Communication Setting** in **Project Explorer**.
5. Select **COMMGR** from the **Communication Interface** drop-down menu in the **Communication Setting** dialog, and then click **Routing Setting**.



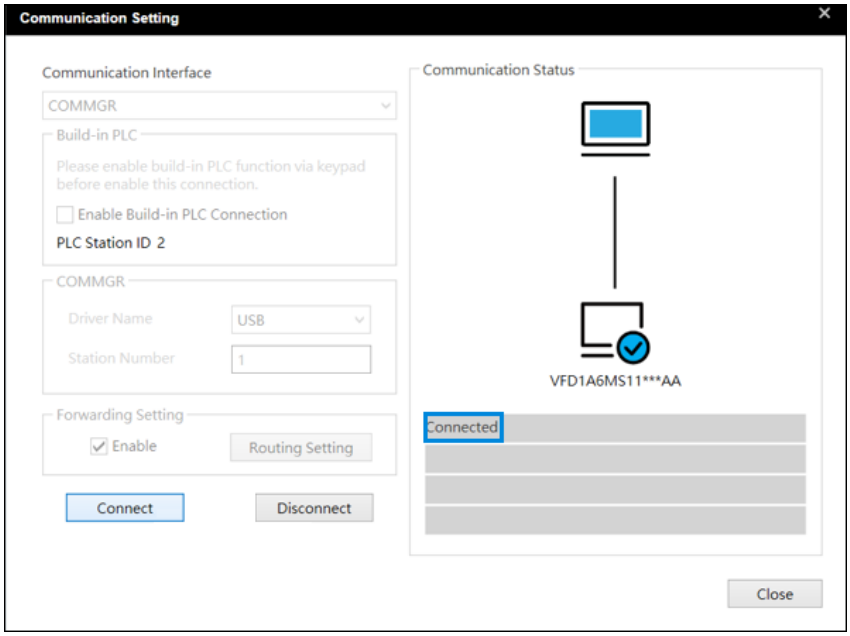
6. In the **Routing Setting** dialog, select **VFD1A6MS11***AA**, and then click **Path Setting**.



7. Select **Enable** checkbox in the **Forwarding Setting** area of the **Communication Setting** dialog, and then click **Connect**.



The Communication Status area displays “The connection is created successfully.” which means the connection is created with the device.



After the above setting steps are completed, the connection between the devices can be created, and parameters can be uploaded / downloaded on MS300 through the forwarding of the controller.

The One Wire forwarding function has the following limitations:

- The first-layer device must be a Delta controller or DOP-100 and DOP-300 series HMI, and the computer can connect the controller through USB, serial port, and Ethernet to perform the forwarding.

- The second and third-layer devices can be a Delta controller or drive, but the backend of the drive cannot perform the forwarding any longer.
- The forwarding between the Delta controller and drive supports Modbus, ModbusTCP, and EtherNet/IP fieldbus.
- The forwarding function supports up to a maximum of three-layer devices. For example, computer > PLC1 > PLC2 > Drive1.

Chapter 6: Setting Device (Hardware Configuration)

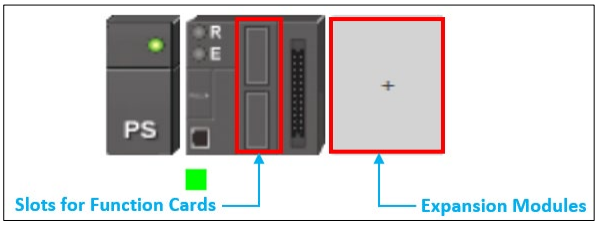
6.1 Hardware Configuration

Configure modules, remote groups, module parameter settings and perform download / upload, online detection, and other functions through the **Hardware Configuration** function.

How to open the **Hardware Configuration** page?

- In **Project Explorer**, double-click **Controller** > **Hardware Configuration** > **Configuration**.

The following takes the AS series controller as an example.



The AS series controller has expansion modules with non-backplane design placed on the right slots for function cards. AS300 series CPU and SCM modules have function card slots and AS200 / AS100 series CPU has in-built communication port for CANopen.

The **Hardware Configuration** includes **Configuration** and **Parameter Setting**.

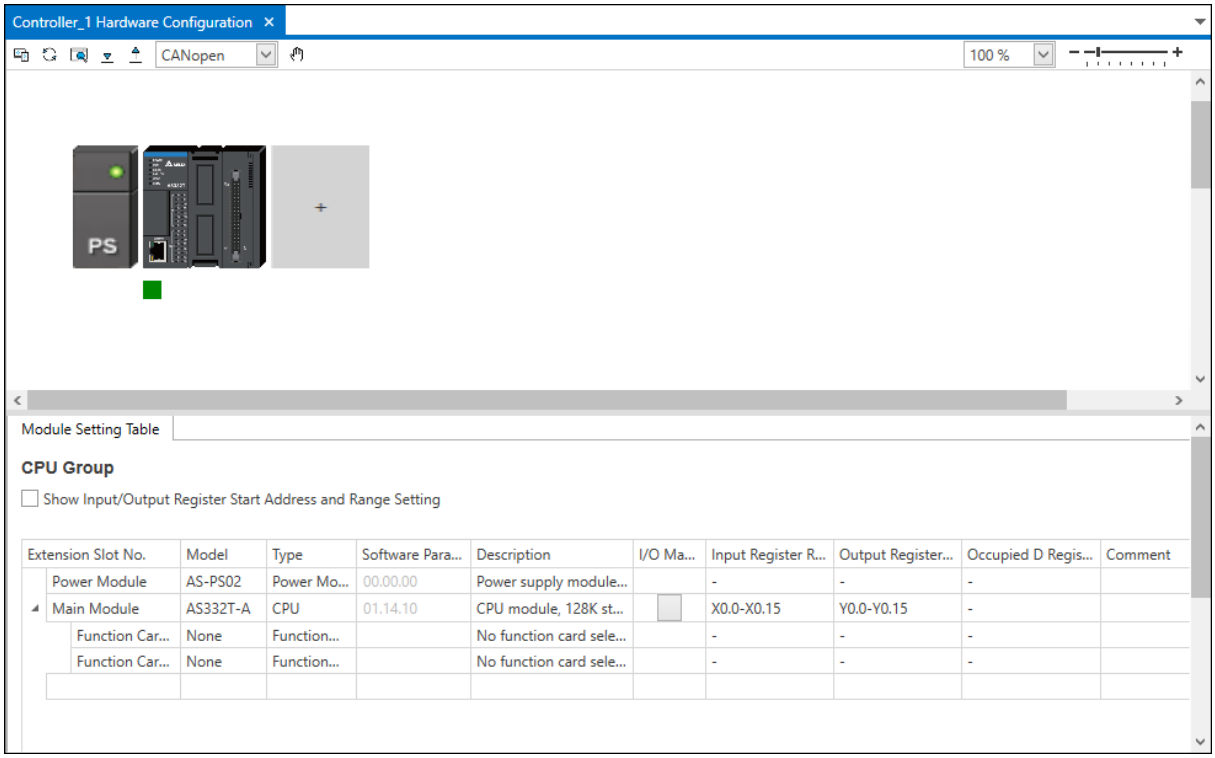
- **Configuration** provides a graphical user interface to plan the hardware configuration.
- **Parameter Setting** provides the option to set the parameter values in the device or the module. It also provides the parameter basic information for you to understand the parameter characteristic.

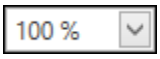
6.1.1 Creating Controller Configuration

Do the following to create configuration.

- Double-click **Controller** > **Hardware Configuration** > **Configuration** in **Project Explorer**.

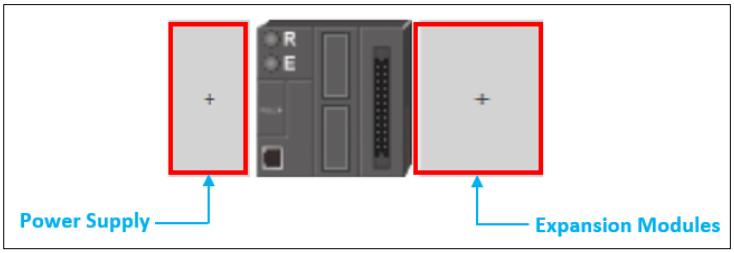
The following table describes the functions in **Hardware Configuration**.



Function	Description
	Click to hide / display module information, include information such as W*H*D.
	Click to reset register address.
	Click to perform I/O scan.
	Click to download.
	Click to upload.
	Select the remote IO network.
	Select the page display percentage.
	Click or drag to adjust the page display percentage.
	Enable the drag mode.

6.1.1.1 Adding Module

The AS series controller has a non-backplane design with an expansion module placed on the right side (**+** position) and power supply module to the left side (**+** position) of the CPU, and only one power supply module can be added.



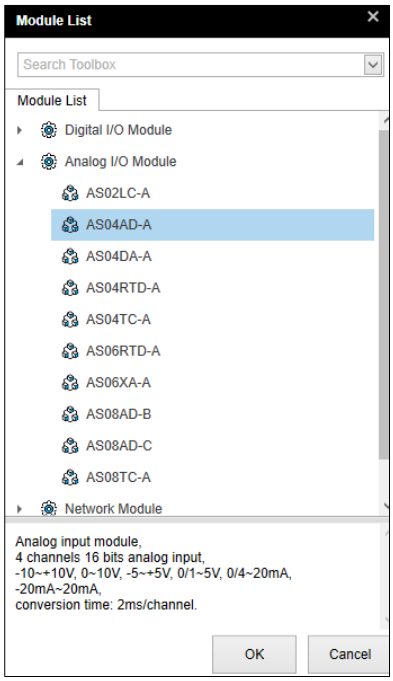
You can add modules from the **Module List** or **Toolbox**.

To add modules from the module list

1. Double-click the **+** position.
2. Expand the required option in the **Module List**.

   Note:

   - The **Model List** displays only those modules which are supported in the selected position.
   - After selecting the module, you can view the description of the module in the bottom of the **Module List**.

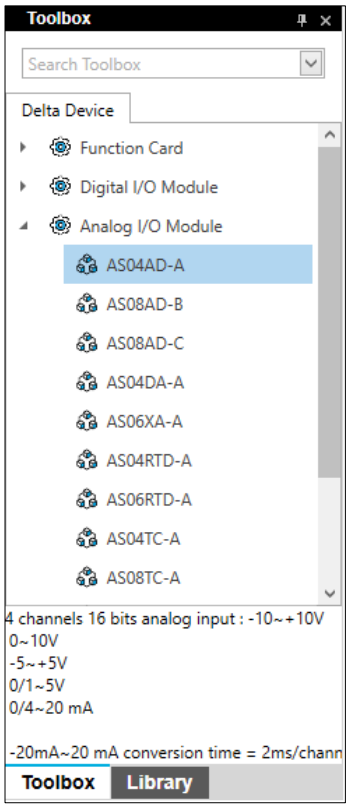


3. Double-click the required module to add it to the **Hardware Configuration** page.

To add modules from Toolbox

1. Select the required module in the **Toolbox** pane.

   Note: After selecting the module, you can view the description of the module in the bottom of the **Toolbox** pane.



2. Drag the modules to the **+** position.

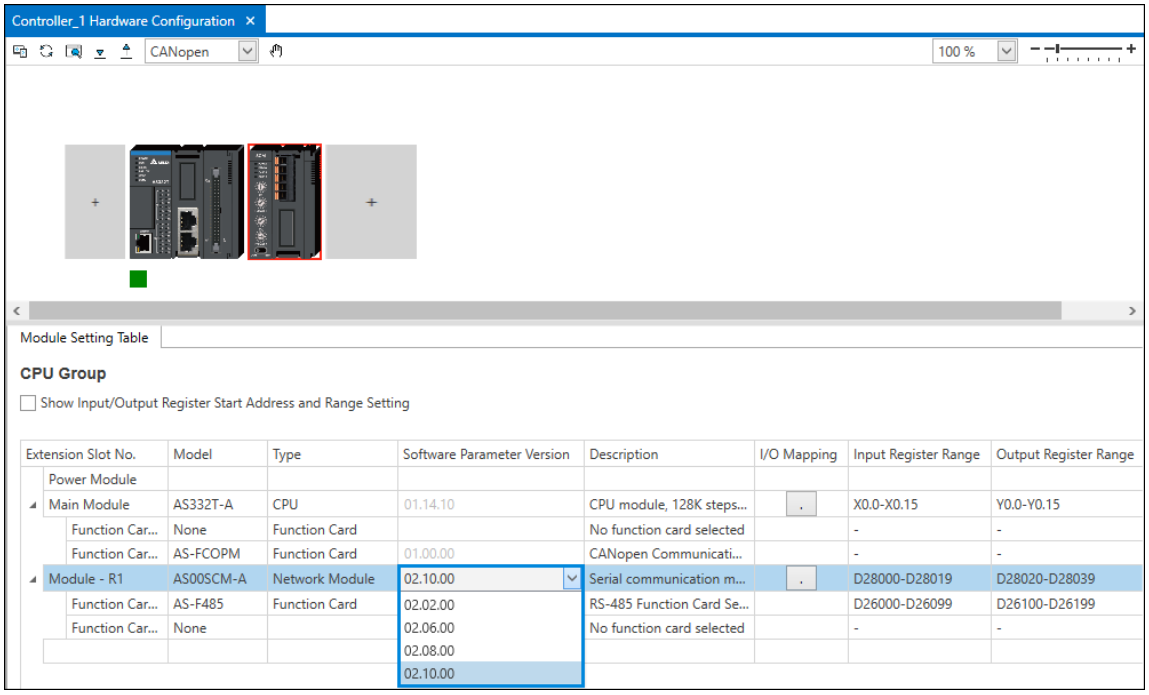
   Or

   Double-click the modules in the **Toolbox** pane after clicking the **+** position.

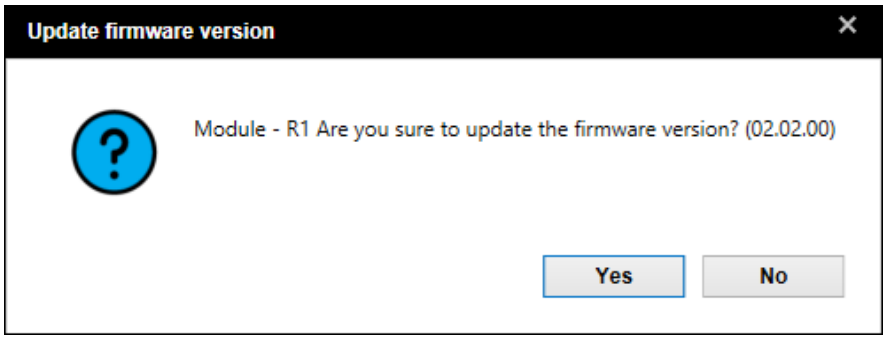
After the module is added, follow these steps to switch the module version.

To switch module version

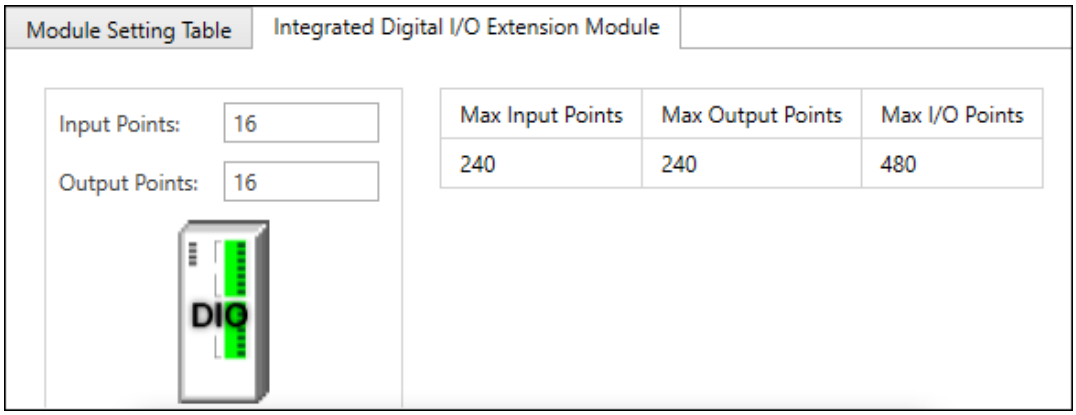
1. Select the required firmware version from the drop-down menu to change the module version.



2. In the **Update firmware version** dialog, click **Yes** to complete the version switch. Click **No** to cancel the version switch.

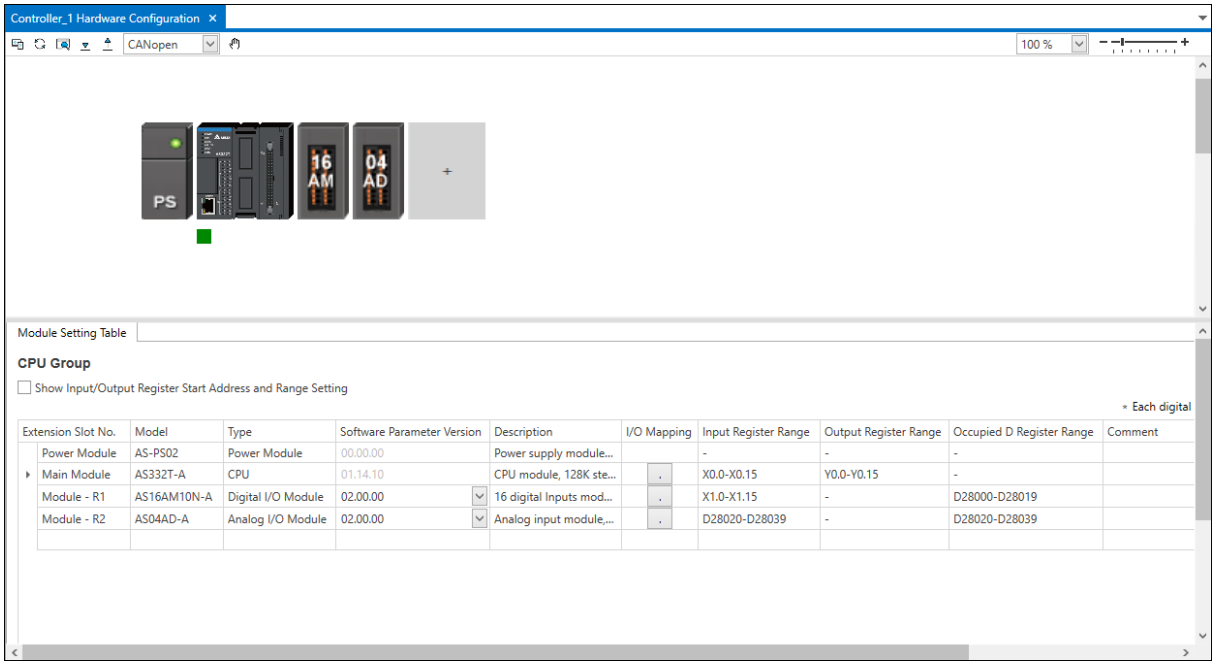


Note: The digital I/O modules of DVP series devices support only offline project configuration. When scanning or uploading the module, the total points of Integrated Digital I/O Expansion Modules are displayed.



6.1.1.2 Address Assignment of Modules

When a configuration is created in hardware view (upper pane), the **Module Setting Table** (bottom pane) automatically gets updated with address assignment. The following figure shows an example for address assignment in **Module Setting Table**.



The following table lists the items in the **Module Setting Table** tab.

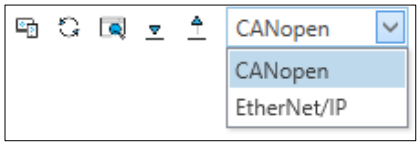
Item	Description
Show Input / Output Register Start Address and Range Setting	Select to show the I/O Register start address and range setting.
Extension Slot No.	Displays extension module number.
Model	Displays module name.
Type	Displays module type.
Software Parameter Version	Displays module firmware version.
Description	Displays module description.
I/O Mapping	Click ... to open **I/O Mapping Table** for the module.
Input Register Range	Displays input register range for the module.
Output Register Range	Displays output register range for the module.
Occupied D Register Range	Displays the D register range occupied by the module when configuring a module.
Comment	Enter comments.

6.1.1.3 Adding Remote Module

Add a remote module in the configuration according to your requirement. The device displayed in the **Hardware Configuration** page appears automatically in the **Parameter Setting** page.

To add a remote module

1. Select the type for the remote module.



2. Double-click remote module **+** position to automatically set related parameters.

   Note: The number of connected remote modules can only be modified in the **Hardware Configuration** page. **Parameter Setting** > **Function Card Setting** > **AS Remote Module No.** option does not support this feature yet.

6.1.1.4 Deleting a Module

Delete modules in the configuration according to your requirement. Do any of following to delete the module.

- Select the module to delete, and then click 

in the menu bar.

Or

- Right-click the module and select **Delete**.

Or

- Select the module to delete, and then press **Delete**.

Or

- Right-click the module in the **Module Setting Table** and select **Delete**.

Or

- Select the module to delete, and then click 

on the **Editor** toolbar.

6.1.1.5 Replace Module

Replace modules in the configuration according to your requirement.

Note: The **Replace Module** function do not support replacement of CPU and RTU modules.

To replace a module

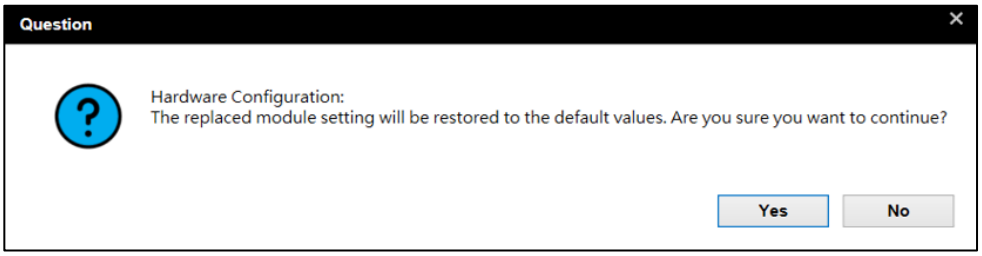
1. Right-click the module to be replaced, and then select **Replace Module**.

   Or

   Right-click the module in **Module Setting Table** to open its context menu and click **Replace Module**.

2. Expand the supported modules in the **Module List** pane, and then double-click the module which you intend to replace with the old one.

3. Click **Yes** to continue or **No** to cancel the replacement.



   Note: If you want to replace all the modules of the same type with another type, select **Replace All** checkbox in the **Module List** pane.

6.1.1.6 Copy / Paste

Copy / paste modules in the configuration according to your requirement.

To copy a module

1. Select the module to be copied, and do any of the following:

   - Click **Edit** > **Copy** on the menu bar.

   Or

   - Right-click the module, and then select **Copy**.

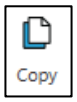
   Or

- Press **Ctrl** + **C**.

Or

- Right-click the module in **Module Setting Table**, and then select **Copy**.

Or

- Click 

on the **Editor** toolbar.

To paste a module

1. Select the location to paste the module, and do any of the following:

   - Click **Edit** > **Paste** on the menu bar .

   Or

   - Right-click **+** position, and then select **Paste**.

   Or

   - Press **Ctrl** + **V**.

   Or

   - Right-click the module in the **Module Setting Table**, and then select **Paste**.

   Or

   - Click 

on the **Editor** toolbar.

Note:

- Since AS series has a non-backplane design, when a module is pasted over another, the pasted module is inserted to the left of the existing module and the existing module shifts to the right position.
- Input/Output Register Range: Register allocation is done automatically to the pasted modules.
- Comment: Comment is copied from the source module and retained in the pasted positions.

- Internal Parameters: Parameters are not copied to the pasted module; and will reset to default.
- Corresponding D registers: Register allocation is done automatically to the pasted modules.

6.1.1.7 Cut / Paste

Cut / paste modules in the configuration according to your requirement.

Note: The **Cut / Paste** functions do not support Power Supply, CPU, RTU modules.

To cut a module

1. Select the module to be cut, and do any of the following:
   - Click **Edit** > **Cut** on the menu bar .

   Or

   - Right-click the module, and then select **Cut**.

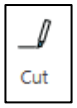
   Or

   - Press **Ctrl** + **X**.

   Or

   - Right-click the module in the **Module Setting Table**, and then select **Cut**.

   Or

   - Click 

on the **Editor** toolbar.

To paste a module

1. Select the location to paste the module, and do any of the following:
   - Click **Edit** > **Paste** on the menu bar.

   Or

   - Right-click **+** position, and then select **Paste**.

   Or

- Press **Ctrl** + **V**.

Or

- Right-click the module in the **Module Setting Table**, and then select **Paste**.

Or

- Click 

on the **Editor** toolbar.

Note:

- While executing **Cut/Paste** function, modules that are cut can only be pasted once.
- Comment, Internal Parameters will be transferred to the pasted module.
- Input / Output Register Range: Register allocation is done automatically to the pasted module location.
- Corresponding D Registers: Register allocation is done automatically to the pasted module location.

6.1.1.8 I/O Mapping

For installed modules and CPU, you can assign a variable for the parameter or channel.

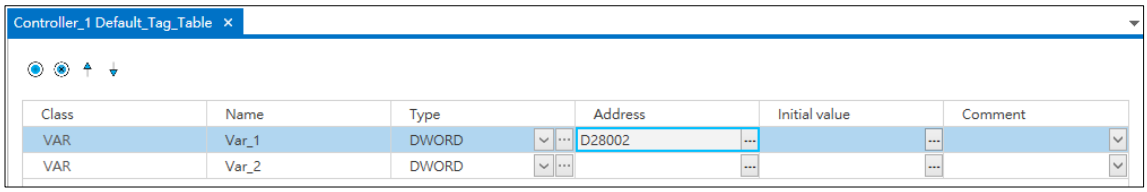
To configure I/O mapping

1. Click  in the **I/O Mapping** field of the **Module Setting Table**.
2. Select a variable table and variable from the **Variable** drop down list in the **I/O Mapping Table** page.

   Or

   Select a variable table and enter a new variable name in the **Variable** field.

   The register **Address** in the **I/O Mapping Table** is set to specified variable **Address** in the **Default Tag Table**.

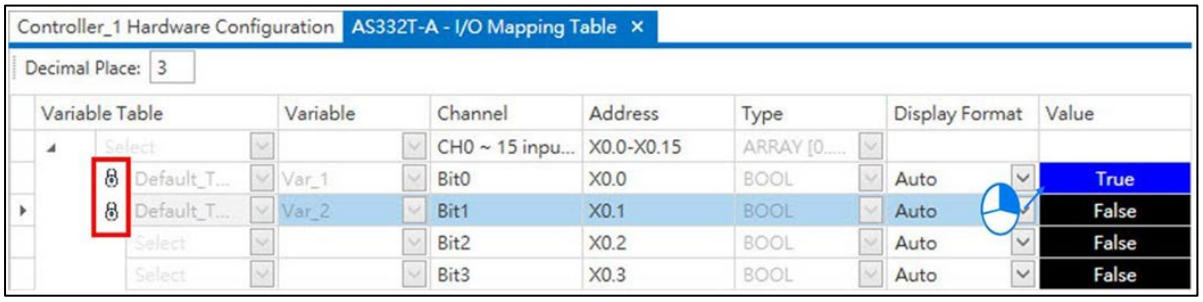


I/O Mapping Address Forced Lock Setting

When running online, right-click the output/input address of the I/O mapping table, and a forced setting lock menu appears.

Mandatory setting of lock function	Features
Mandatory On	Force the address to switch to TRUE.
Force Off	Force the address to switch to FALSE.
Dismiss	Cancel the mandatory setting of the address.
Unblock all	Remove all mandatory address setting.

After mandatory setting, the lock icon is displayed in front of the address, and the lock icon disappears after release.



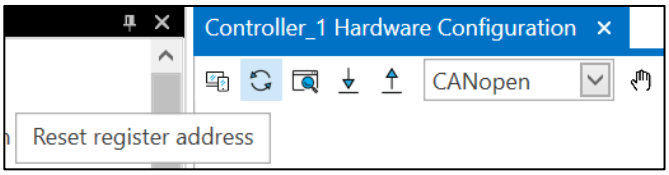
6.1.1.9 Reset Register Address

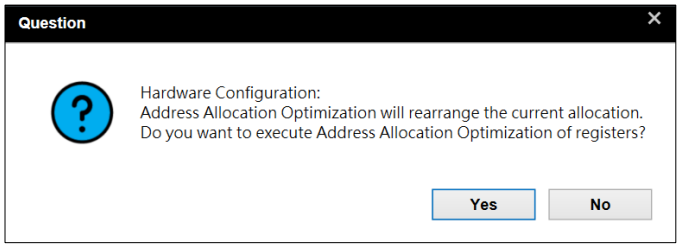
For AS series controller, since the process of module configuration involves many executions to copy / paste, cut / paste and so on, the Input / Output register range for each module can be automatically assigned by the system and not according to a continuous sequence. Use **Reset Register Address** function to reset the register addresses.

Follow these steps to reset the register address.

To reset register address

1. Click  on the toolbar in **Hardware Configuration** dialog to open a confirmation dialog.





2. Click **Yes** to reset register address or **No** to cancel the operation.

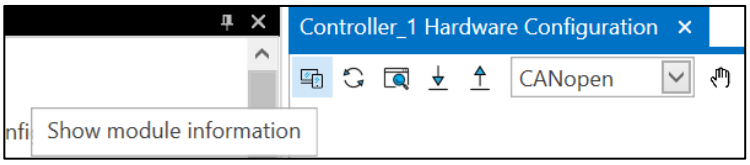
6.1.1.10 Show / Hide Module Information

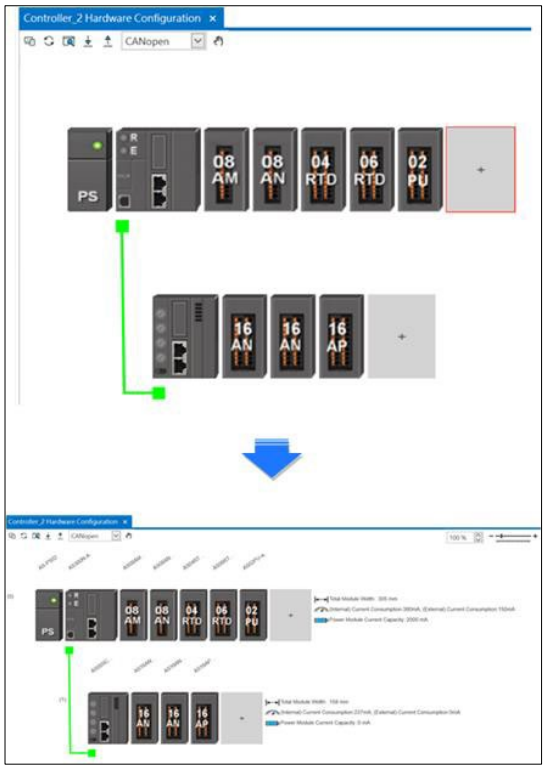
View the part number of modules, total module width (mm), Internal and External Current Consumption (mA), Power Module Capacity (mA), RTU numbers and so on, with the help of **Show Module Information**.

Follow these steps to show / hide module information.

To show / hide module information

1. Click  on the toolbar of **Hardware Configuration** dialog.





2. Click  again to hide module information.

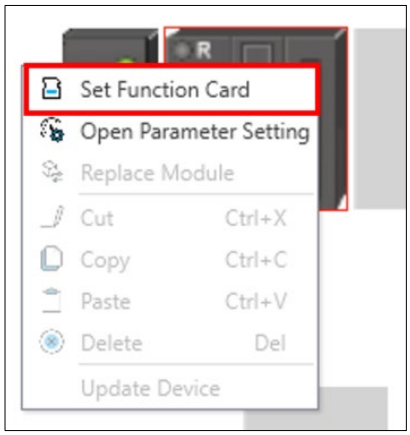
6.1.1.11 Set Function Card of Controller

The **Set Function Card** feature allows to set the function card. This function is available only for CPU & NIO modules that support function card.

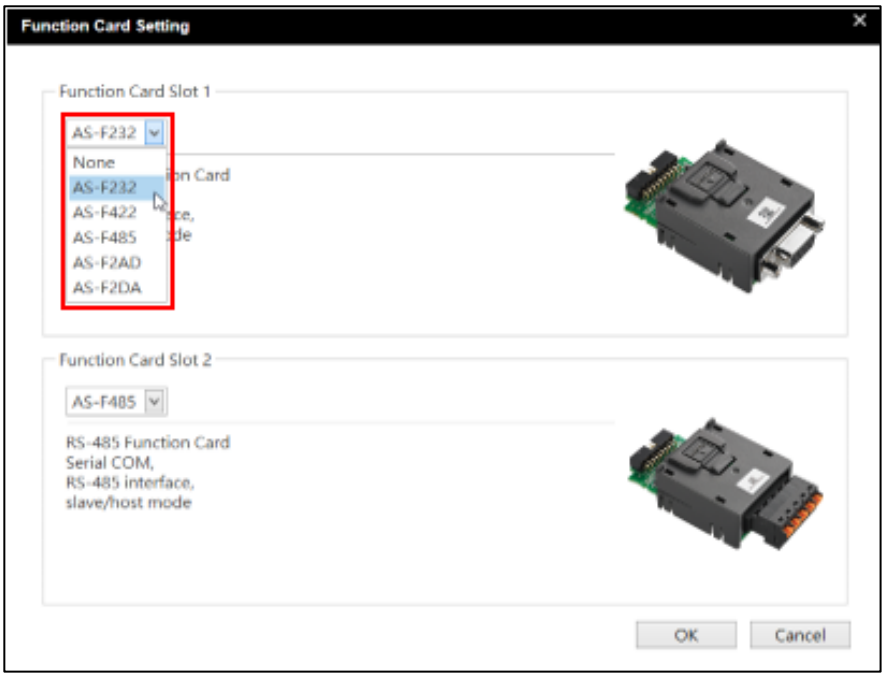
Follow these steps to set function card.

To set function card

1. Right-click the CPU module and click **Set Function Card** in the menu.

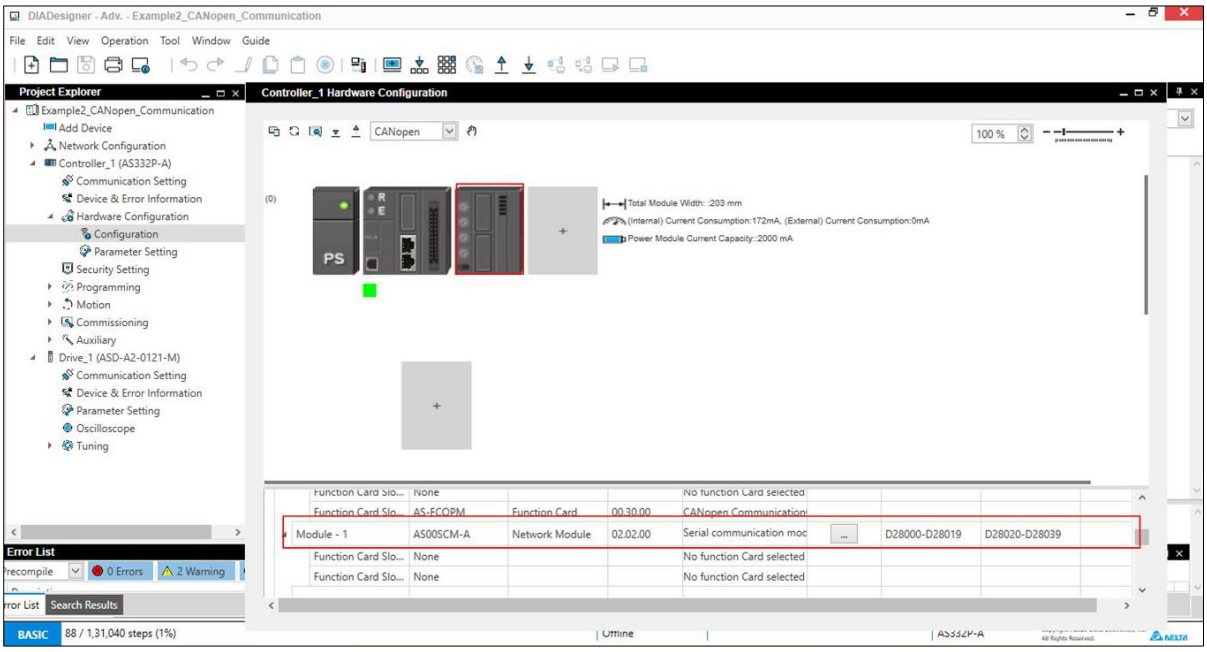


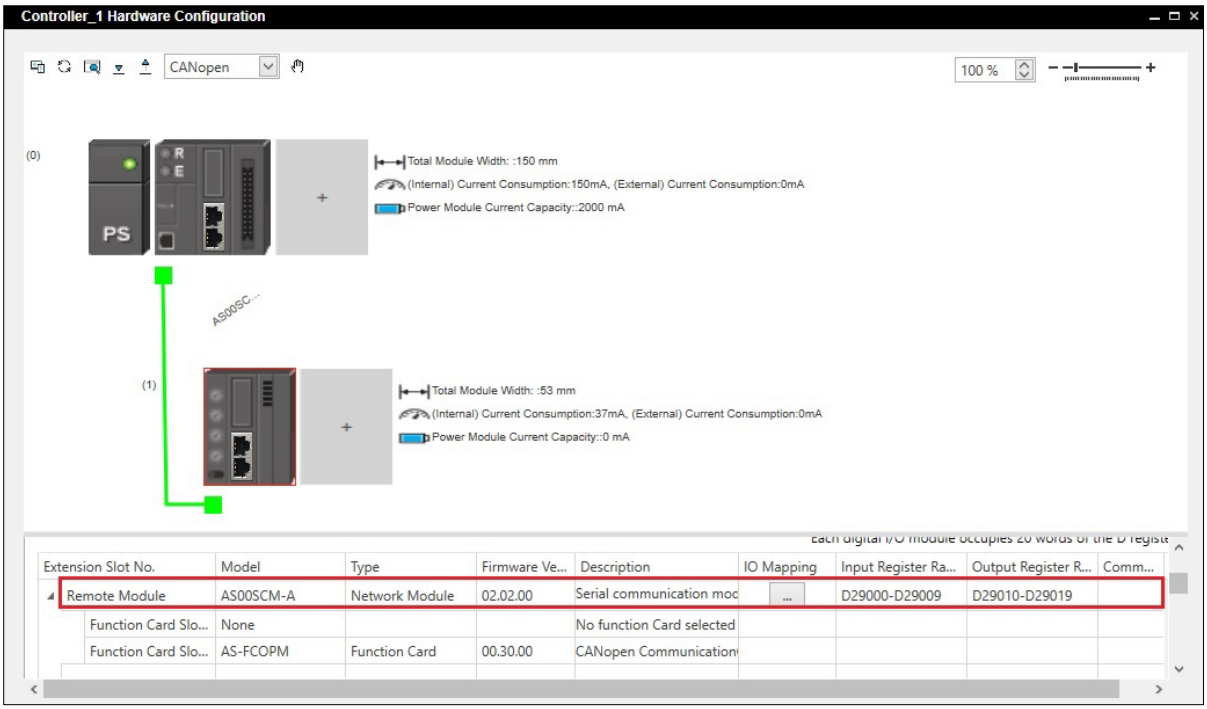
2. From the drop-down menu, select the desired function card.



6.1.1.12 Set AS00SCM module

AS00SCM-A (hereinafter referred to as SCM module) is the AS series serial communication expansion module and remote module. For detailed supported function cards and communication functions, refer to *Chapter 9 AS00SCM Communication Module of the AS series module manual*.





When AS00SCM-A is the serial communication remote module on the right side of the AS host, the communication protocols of the serial communication cards AS-F232, AS-F422 and AS-F485 support MODBUS and UD Link (user-defined format).

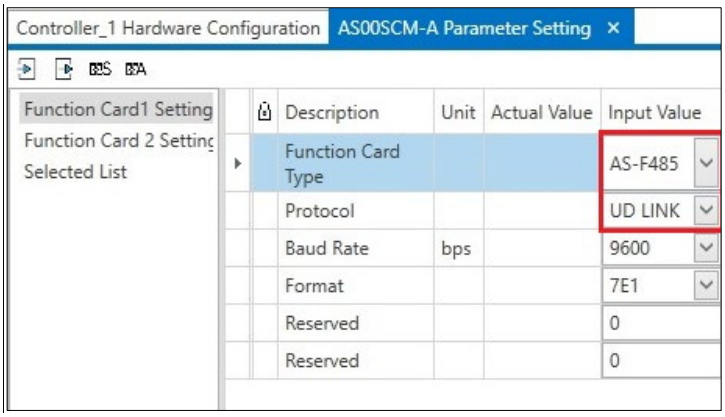
MODBUS

AS00SCM-A provides connection standard MODBUS RS-232, RS-422 and RS-485 communication equipment, through the establishment of data exchange table to achieve the purpose of data exchange with the slave station (currently not supported by the software).

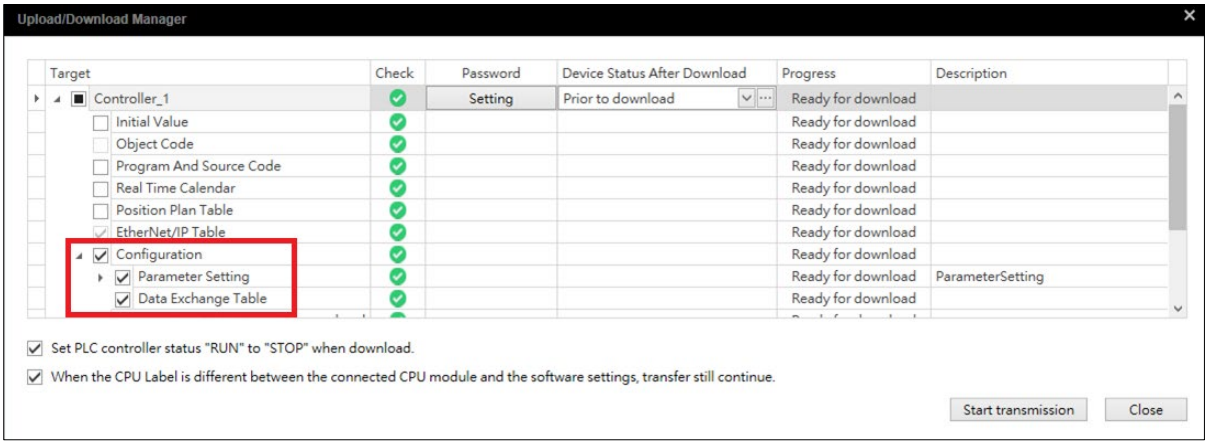
UD Link

1. Set module parameters

   Set the function card slot 1 or slot 2 to support the Function Card Type, the communication Protocol is set to **UD Link** and the related communication rate and communication format (when using UD Link, it is recommended to select a communication format with a bit length of 8, such as 8E1, 8N1, 8O2 Etc. to ensure the integrity of the transmitted data).

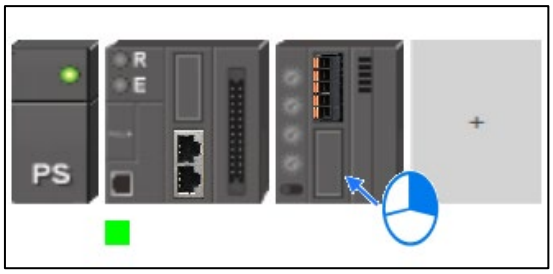


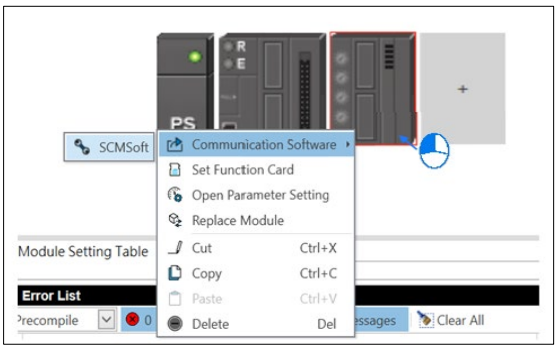
After the parameter setting is completed, download the configuration.



2. Edit SCMSoft

On the hardware configuration screen, right-click AS00SCM-A module to open the context menu.





Note: It is necessary to set up host communication before starting SCMSoft. Only Ethernet, USB, and Serial are supported. Simulators are not supported or have not been set.

3. SCMSoft Edit

   For the use of SCMSoft software, refer to *Section 9.3.2 UD Link mode of the AS series module manual*.

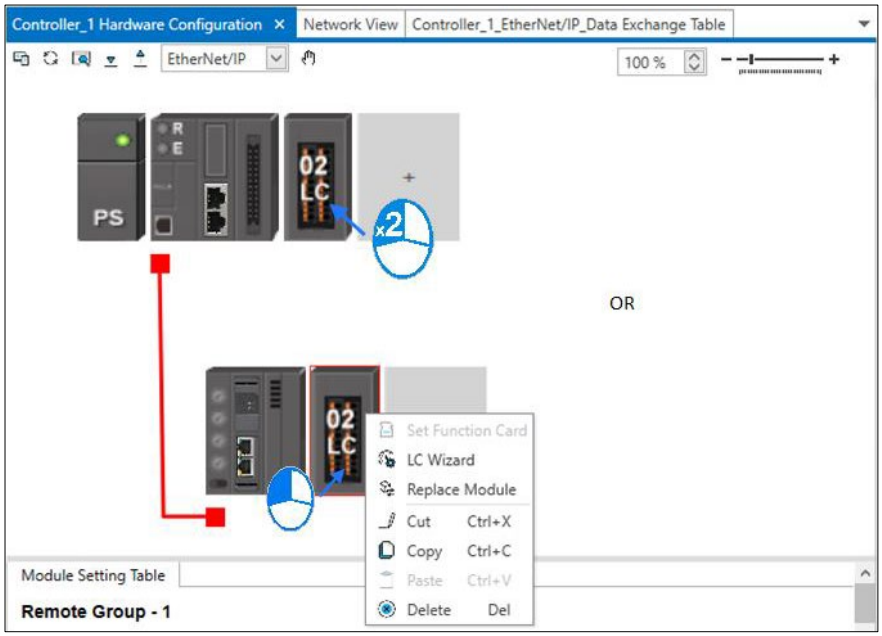
6.1.1.13 Set AS02LC Module

The AS02LC module provides the function of **LC Wizard**, which is convenient for you to make the weight display value of the calibration module match the weight of the Load Cell.

Note: Before executing the LC wizard, it is necessary to download the module configuration including AS02LC to the controller.

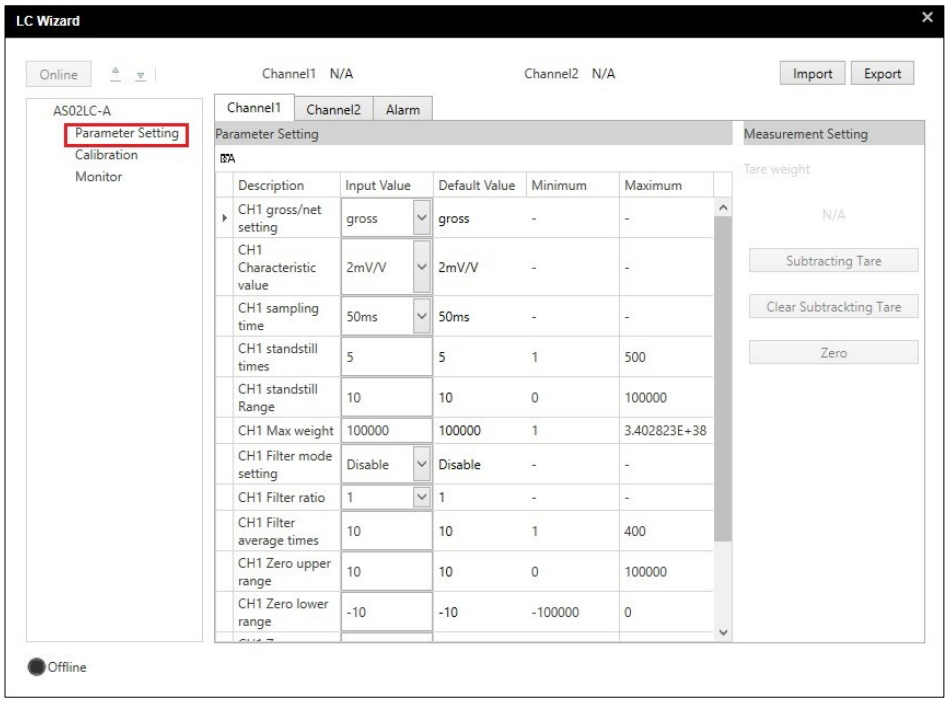
1. Open **LC Wizard**

   Double-click the module icon or right-click the module to select **LC Wizard**.

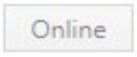


2. AS02LC module parameter setting

   Edit the parameter setting of **Channel1**, **Channel2**, and the characteristic value of the **Alarm** on the parameter editor page, and perform the actions of uploading, downloading, importing, and exporting parameters.

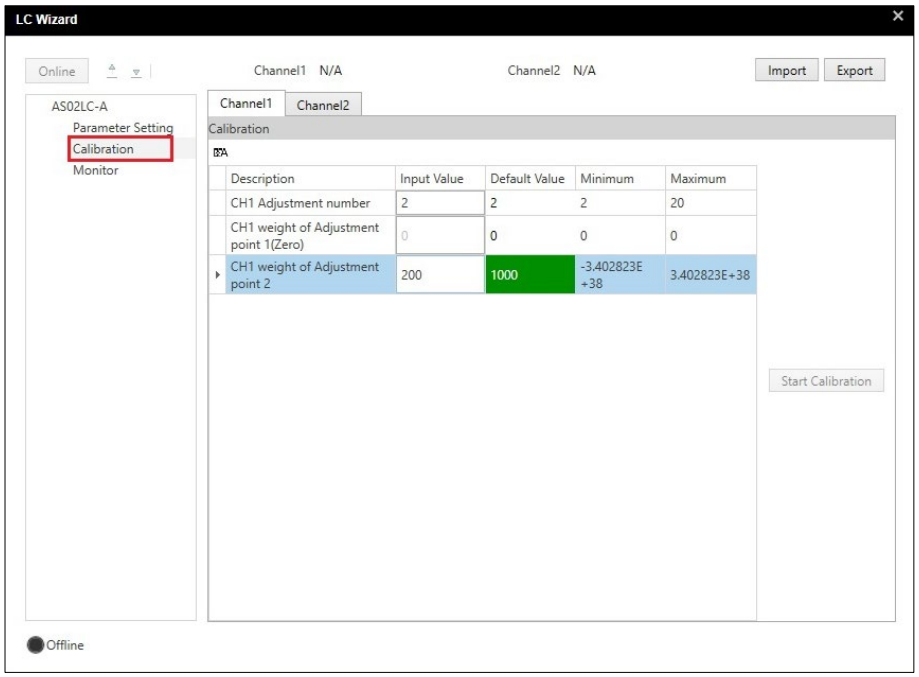


The following table describes the functions of **LC Wizard** toolbar.

Function	Description
	After clicking, execute AS02LC module parameter upload.
	After clicking, execute AS02LC module parameter download.
	After clicking, import the parameter file of the AS02LC module.
	After clicking, export the current parameter file of the AS02LC module.
	After clicking, the parameter values on the current page will all be restored to the preset values.
	After clicking, the AS02LC module enters the online state.

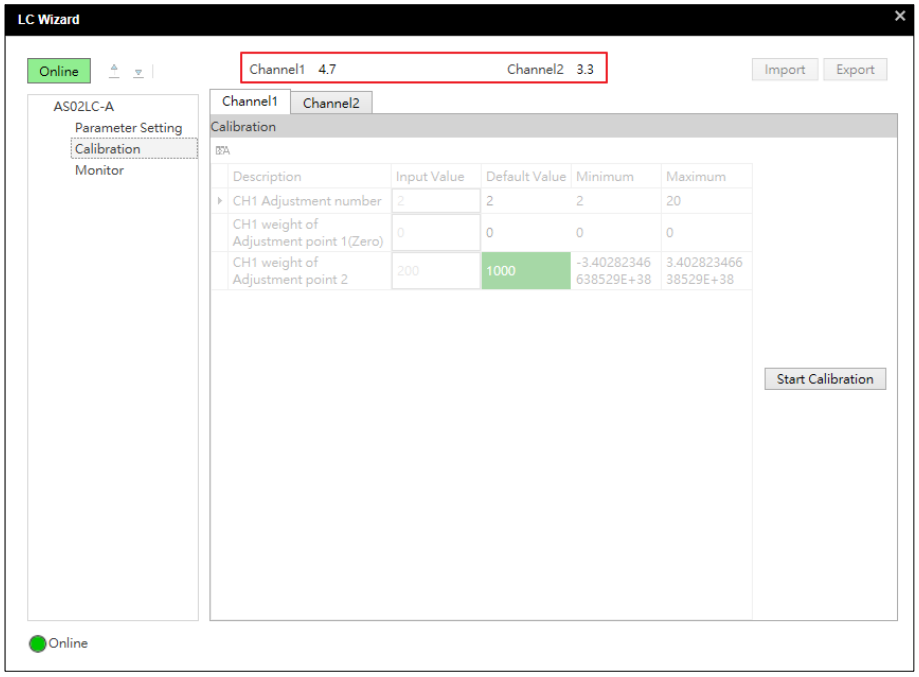
3. Edit AS02LC module calibration process parameters

   Switch to the **Calibration** process page to set the number of calibration points and the weight of the calibration points for **Channel1** and **Channel2**. The example in the figure below is a 2-point adjustment, the first point is the zero point, and the second point 200g corresponds to a value of 200. After the setting is completed, click the download button in the toolbar of the **LC Calibration Wizard** to download the setting parameters to the AS02LC module to take.



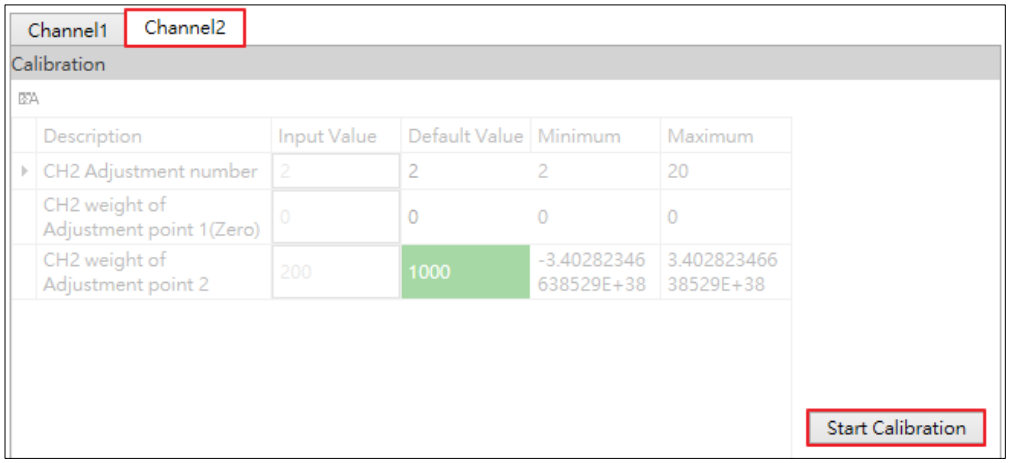
4. Enter Online Mode

Click Online to make the AS02LC module and the LC module enter the online mode. At this time, the background color of the button will change to green Online, and the current object weight can be detected in real time and displayed on the screen, as shown in the figure below. The parameters cannot be edited in the online mode.

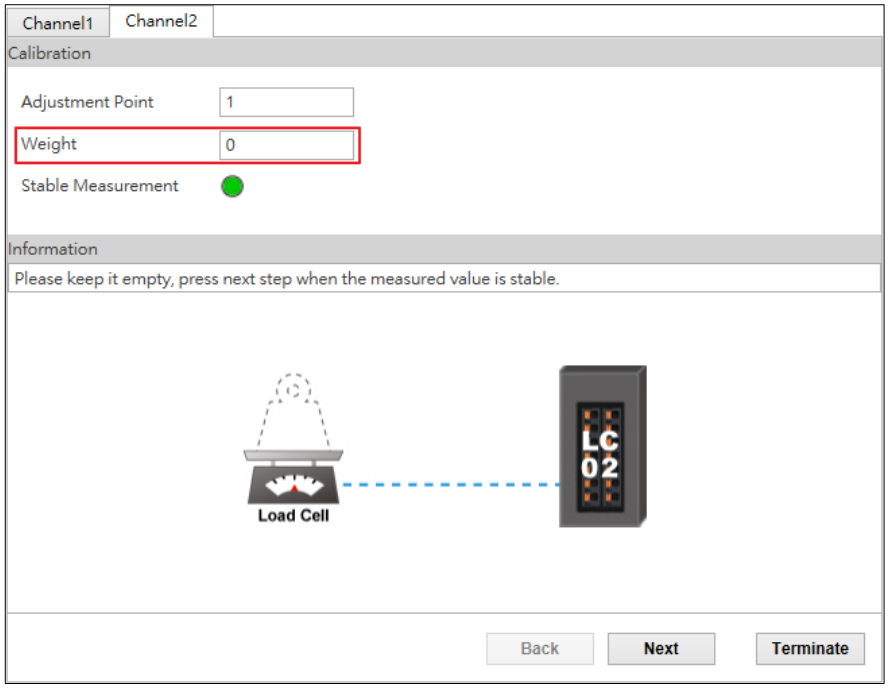


5. AS02LC module calibration process

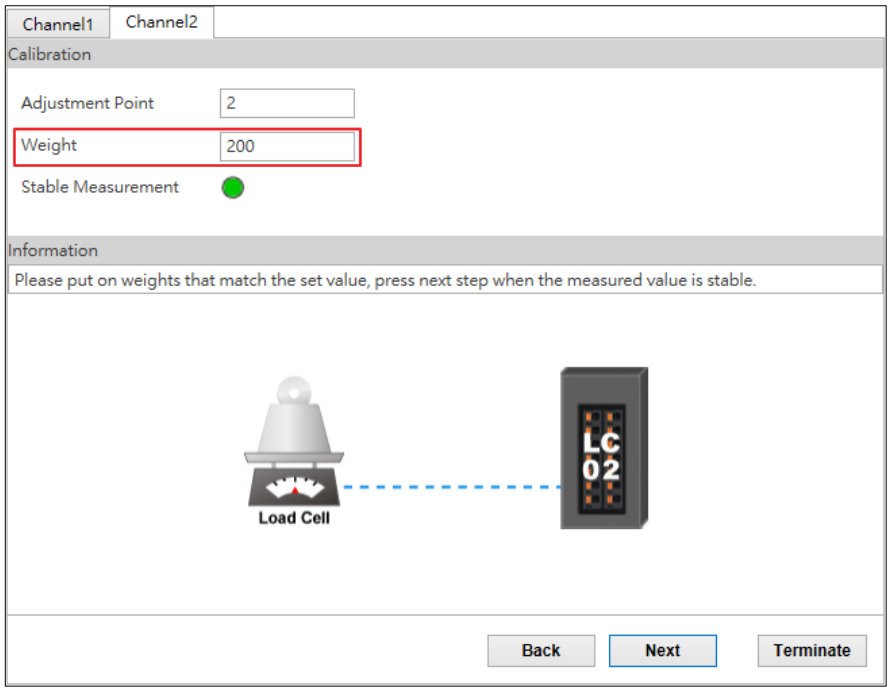
After the AS02LC module enters the online mode, select **Channel1** or **Channel2** and click Start Calibration button to enter the calibration process.



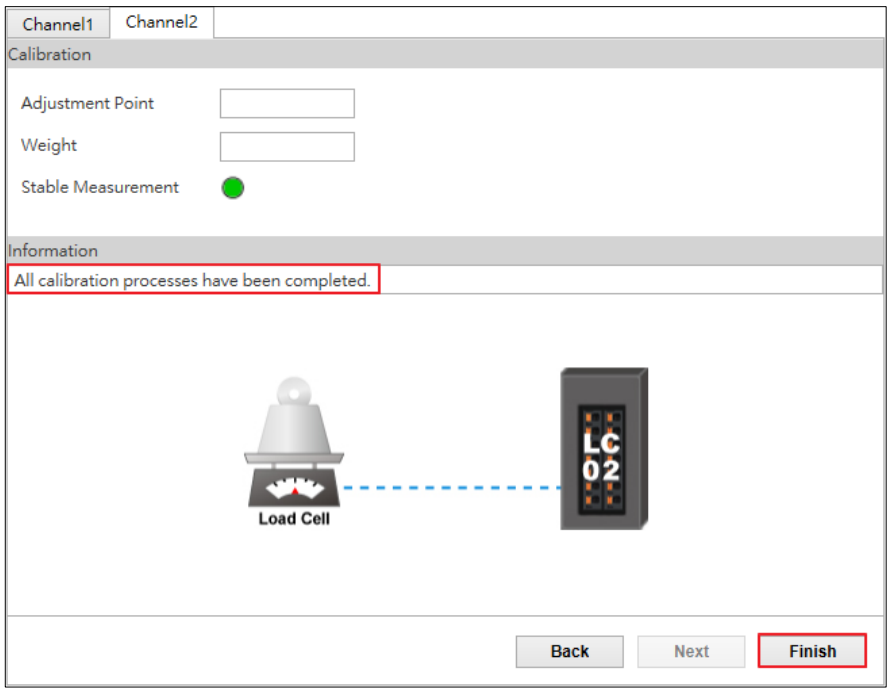
i. Keep the LC sensor empty, do not put any weight, click **Next**.



ii. Place the actual weight corresponding to the second point on the LC sensor. In this example, place a 200g weight, and then click **Next**. If it is a multi-point calibration, this step will be repeated.

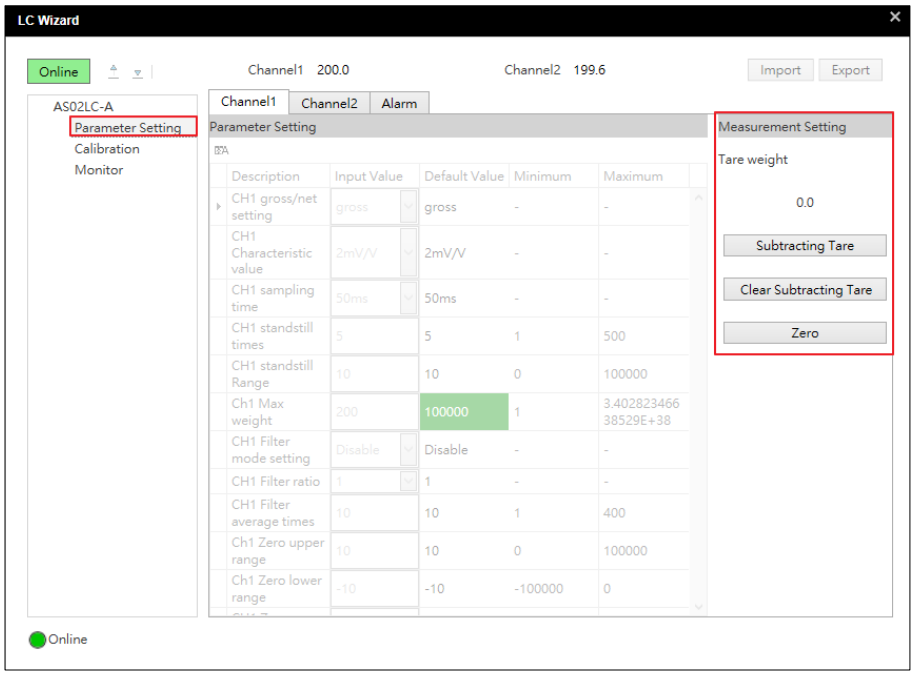


iii. Click **Finish** to complete the weight adjustment.



6. AS02LC module measurement setting

After the AS02LC module enters the online mode, there is a **Measurement Setting** field on the right side of the **Parameter Setting** page, and perform **Subtracting Tare**, **Clear Subtracting Tare**, and **Zero** actions according to the measurement requirements.

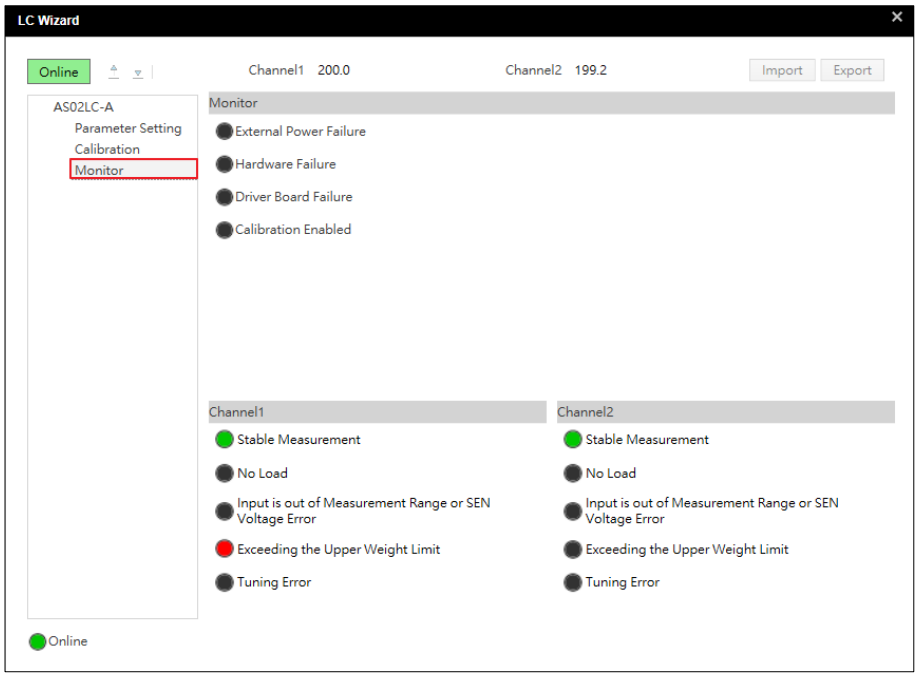


The following table describes the functions of **LC Wizard—Measurement Setting**.

Function	Description
Subtracting Tare	After clicking, the weight of the items currently placed on the channel will be removed.
Clear Subtracting Tare	After clicking, the tare weight recorded in the current channel will be restored.
Zero	After clicking, the display value of the current channel will be reset to 0.

7. LC Calibration Wizard performs monitoring

After the AS02LC module enters the online mode, switch to the **Monitor** tab to view the current status of the AS02LC module.



6.1.1.14 Set DVP Temperature Measurement Module

DVP temperature measurement module provides temperature sensor function, allowing temperature control settings through **Temperature control software**. Take **NTC Wizard** function of the DVP08NTC-S module as an example.

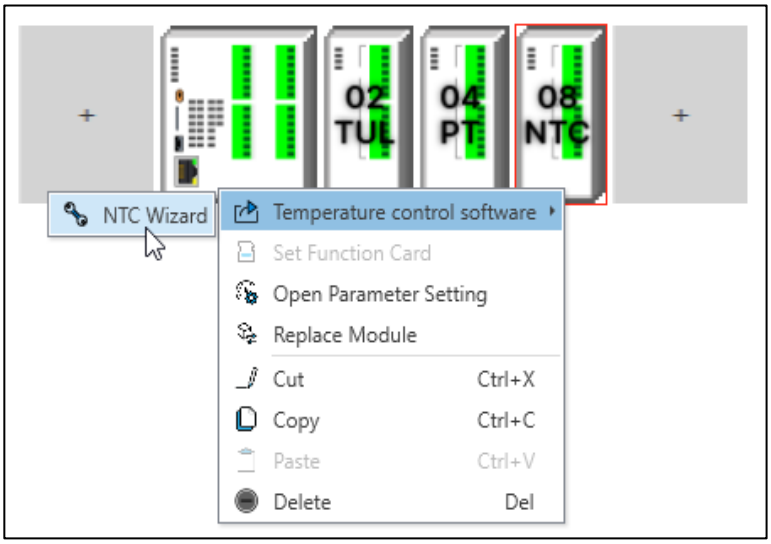
This wizard is designed for DVP08NTC-S featuring user-defined resistance / temperature table, importing / exporting csv file, downloading parameters of one or multiple modules at a time, etc. Refer to the installation manual for this module for detailed operation instructions.

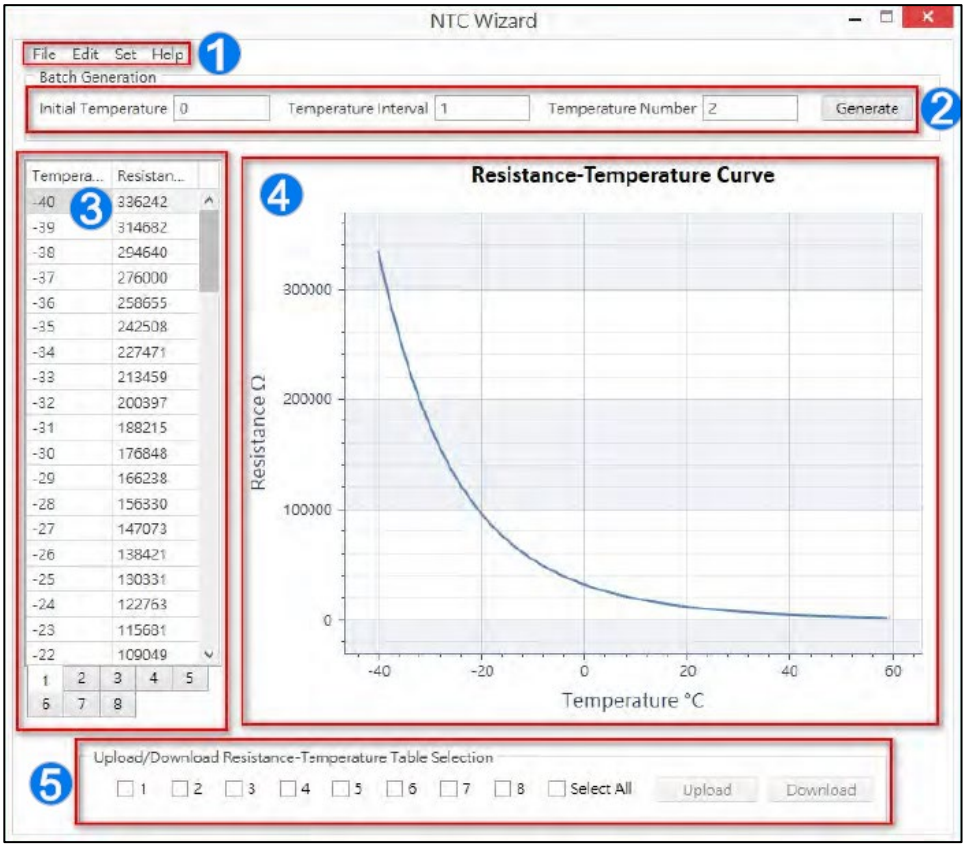
Follow these steps to set temperature control using **NTC Wizard.**

To set temperature control using NTC Wizard

1. Create **DVP-SV3** / **DVP-SX3** controller in **Project Explorer**, and double-click **Controller** > **Hardware Configuration** > **Configuration**.
2. Double-click **+** on the right of the device in the **Hardware Configuration** page to open **Module List** dialog.
3. Add **DVP08NTC-S** Temperature Measurement Module from the Module List, and then click **OK**.

4. Right-click DVP08NTC-S module, and then click **Temperature control software > NTC Wizard** in the context menu to open **NTC Wizard** dialog.





The following table describes the functions in **NTC Wizard** dialog.

Legend	Function	Description
❶	Main Menu	• **File**: Save / Open the file, Import / Export the resistance / temperature table. • **Edit**: Load default values, Clear the resistance / temperature table. • **Set**: Communication settings, RS485 direct to module or connecting to module via controller CPU. • **Help**: Operation instructions, Software information.
❷	Batch Generation	A blank form would be generated after you input the **Initial Temperature**, **Temperature Interval**, and **Temperature Number**.
❸	Resistance Table	A total of eight resistance / temperature tables are available for you to fill in the form with corresponding resistance values.
❹	Resistance Temperature Curve	The curve chart is generated according to the resistance / temperature tables on the left side.
❺	Upload/Download Resistance-Temperature Table Selection	Choose the target table to upload / download.

6.1.1.15 Set DVP01 / DVP02 / DVP201 / DVP202 / DVP211LC-SL Module

DVP01LC-SL / DVP02LC-SL / DVP201LC-SL / DVP202LC-SL / DVP211LC-SL (hereinafter referred to as LC module) are the DVP series analog I/O modules used to connect the load cell, allowing the related settings through **Weighing Software**.

LCSoft is designed for LC module, allowing you to perform system settings, parameter settings, calibration, and monitoring. Refer to *Chapter 6 of the DVPLC series operation manual* for detailed operation instructions.

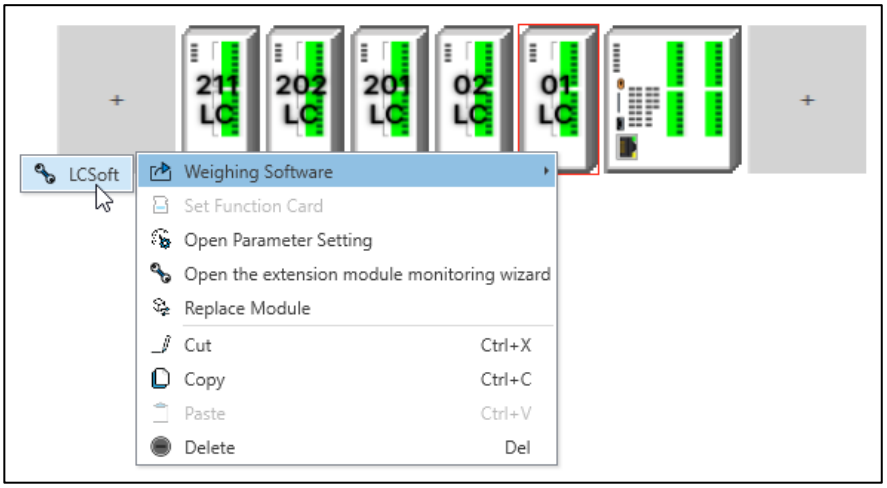
Follow these steps to perform the related setting using **LCSoft**.

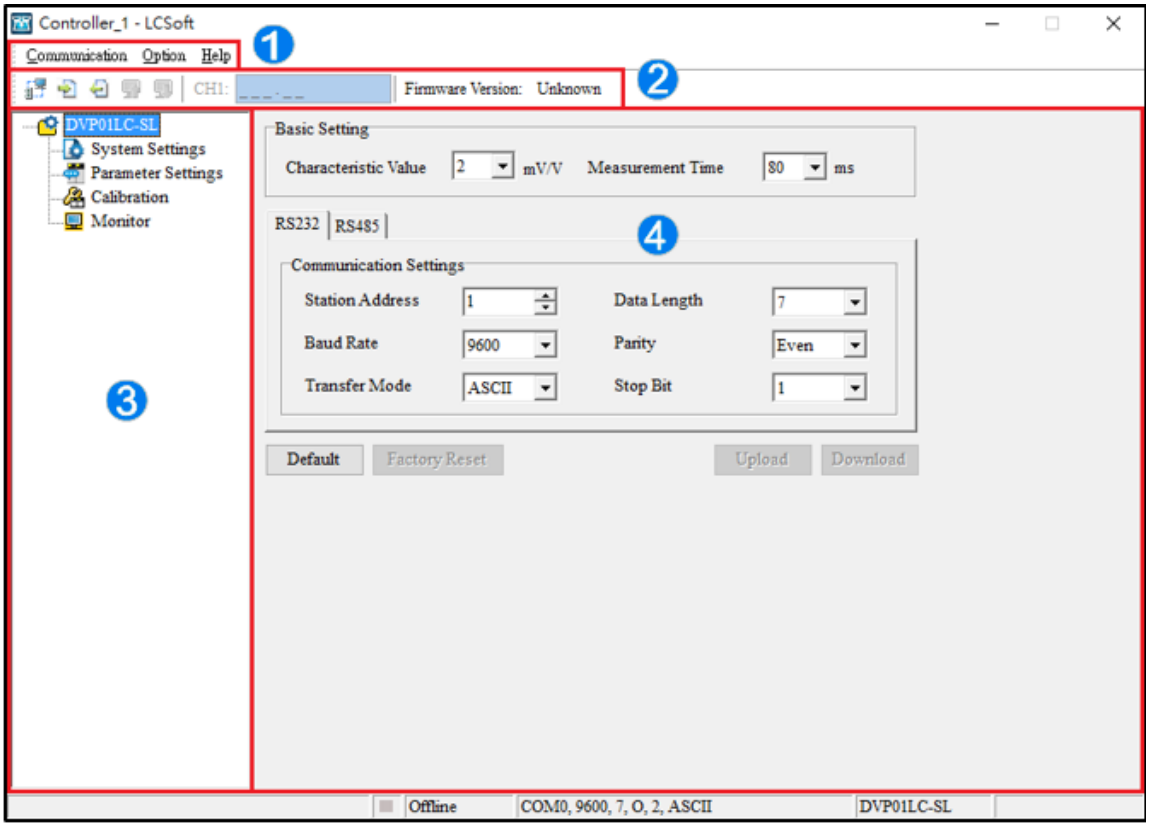
To perform setting using LCSoft

1. Create **DVP-SV3** / **DVP-SX3** controller in **Project Explorer**, and double-click **Controller** > **Hardware Configuration** > **Configuration**.
2. Double-click **+** on the left of the device in the **Hardware Configuration** dialog to open the **Module List** dialog.

3. Add **DVP01LC-SL** / **DVP02LC-SL** / **DVP201LC-SL** / **DVP202LC-SL** / **DVP211LC-SL** module as required from the **Module List** dialog, and then click **OK**.

4. Right-click LC module, and then click **Weighing Software** > **LCSoft** in the context menu to open **LCSoft** dialog.





The following table describes the functions in the **LCSoft** dialog. Refer to *Chapter 6 of the DVPLC series operation manual* for detailed operation instructions.

Legend	Function	Description
❶	Main Menu	• **Communication**: Provide **Online**, **Offline**, **Download**, **Upload** function. • **Option**: Provide **Communication Setting (PC <=> Module)** function. • **Help**: Provide software information and revision history.
❷	Toobar	Provide **Online**, **Import**, **Export**, **Upload**, **Download**, etc.
❸	Project Explorer	Provide **System Settings**, **Parameter Settings**, **Calibration**, and **Monitor** tabs.
❹	Display Area	Display the corresponding tab information selected from the project explorer.

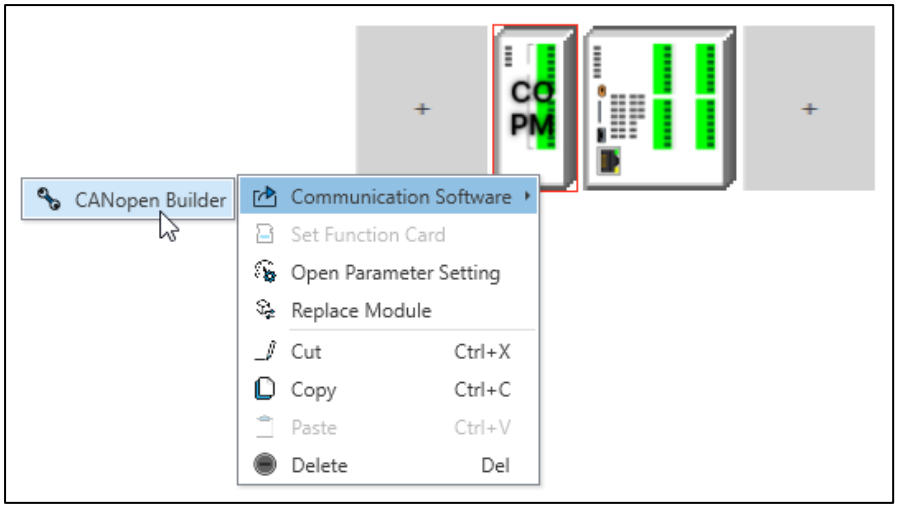
6.1.1.16 Set DVPCOPM-SL Module

DVPCOPM-SL (hereinafter referred to as COPM module) is the DVP series network module, which provides the CANopen communication function, allowing the related communication settings through **CANopen Builder**. Refer to *Chapter 4 of the DVPCOPM-SL operation manual* for detailed operation instructions.

Follow these steps to perform the settings of COPM module.

To perform settings of COPM module

1. Create **DVP-SV3** / **DVP-SX3** controller in **Project Explorer**, and double-click **Controller** > **Hardware Configuration** > **Configuration**.
2. Double-click **+** on the left of the device in the **Hardware Configuration** dialog to open **Module List** dialog.
3. Add **DVPCOPM-S** module from **Module List** dialog, and then click **OK**.
4. Right-click the COPM module, and then click **Communication Software** > **CANopen Builder** in the context menu.



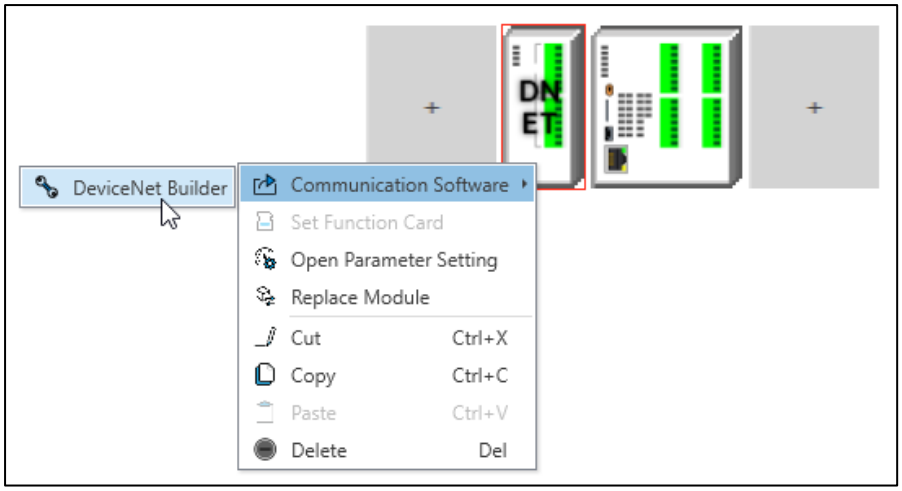
6.1.1.17 Set DVPDNET-SL Module

DVPDNET-SL (hereinafter referred to as DNET module) is the DVP series network module, which provides the DeviceNet communication function, allowing the related communication settings through **DeviceNet Builder**. Refer to *Chapter 4 of the DVPDNET-SL operation manual* for detailed operation instructions.

Follow these steps to perform the settings of DNET module.

To perform settings of DNET module

1. Create **DVP-SV3** / **DVP-SX3** controller in **Project Explorer**, and double-click **Controller** > **Hardware Configuration** > **Configuration**.
2. Double-click **+** on the left of the device in **Hardware Configuration** dialog to open **Module List** dialog.
3. Add **DVPDNET-SL** module from **Module List** dialog, and then click **OK**.
4. Right-click the DNET module, and then click **Communication Software** > **DeviceNet Builder** in the context menu. Refer to *Chapter 4 of the DVPDNET-SL operation manual* for detailed operation instructions.



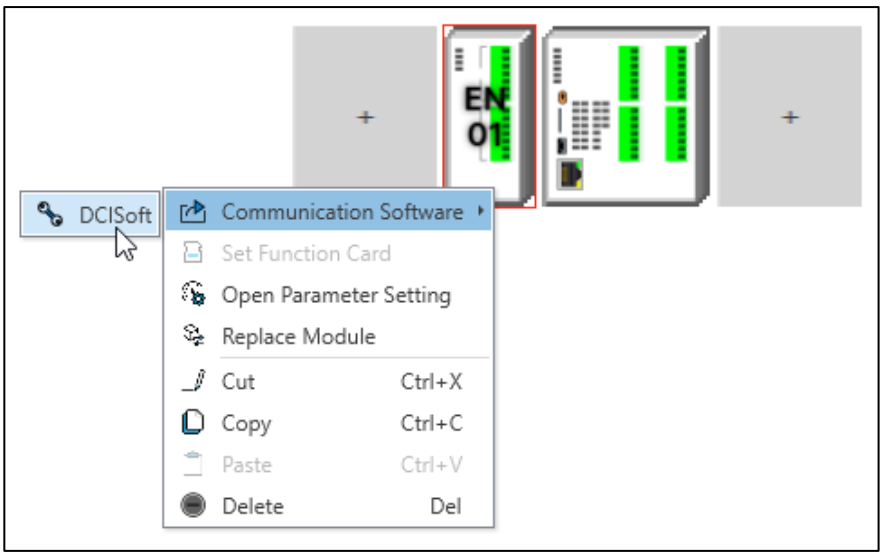
6.1.1.18 Set DVPEN01-SL Module

DVPEN01-SL (hereinafter referred to as EN01 module) is the DVP series network module, which provides the Ethernet communication function, allowing the related communication settings through **DCISoft**. Refer to *Chapter 4 of the DVPEN01-SL operation manual* for detailed operation instructions.

Follow these steps to perform the settings of EN01 module.

To perform settings of EN01 module

1. Create **DVP-SV3** / **DVP-SX3** controller in **Project Explorer**, and double-click **Controller** > **Hardware Configuration** > **Configuration**.
2. Double-click **+** on the left of the device in **Hardware Configuration** dialog to open **Module List** dialog.
3. Add **DVPEN01-SL** module from **Module List** dialog, and then click **OK**.
4. Right-click the EN01 module, and then click **Communication Software** > **DCISoft** in the context menu. Refer to *Chapter 4 of the DVPEN01-SL operation manual* for detailed operation instructions.



6.1.1.19 Set DVPSCM12-SL / DVPSCM52-SL Module

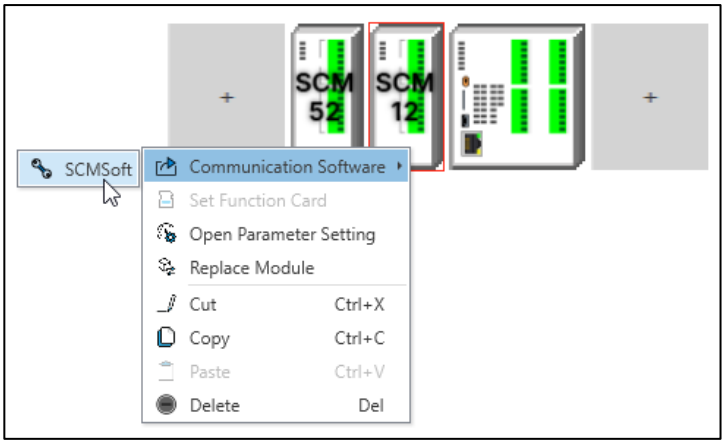
DVPSCM12-SL / DVPSCM52-SL (hereinafter referred to as SCM module) are the DVP series network modules, which provide the Modbus, UD Link, etc. communication functions, allowing the related settings through **SCMSoft**.

SCMSoft is designed for SCM module, allowing you to perform COM Port settings, MODBUS settings, COM Port historical data reviewing, etc. Refer to Chapter 5 of the DVPSCM series operation manual for detailed operation instructions.

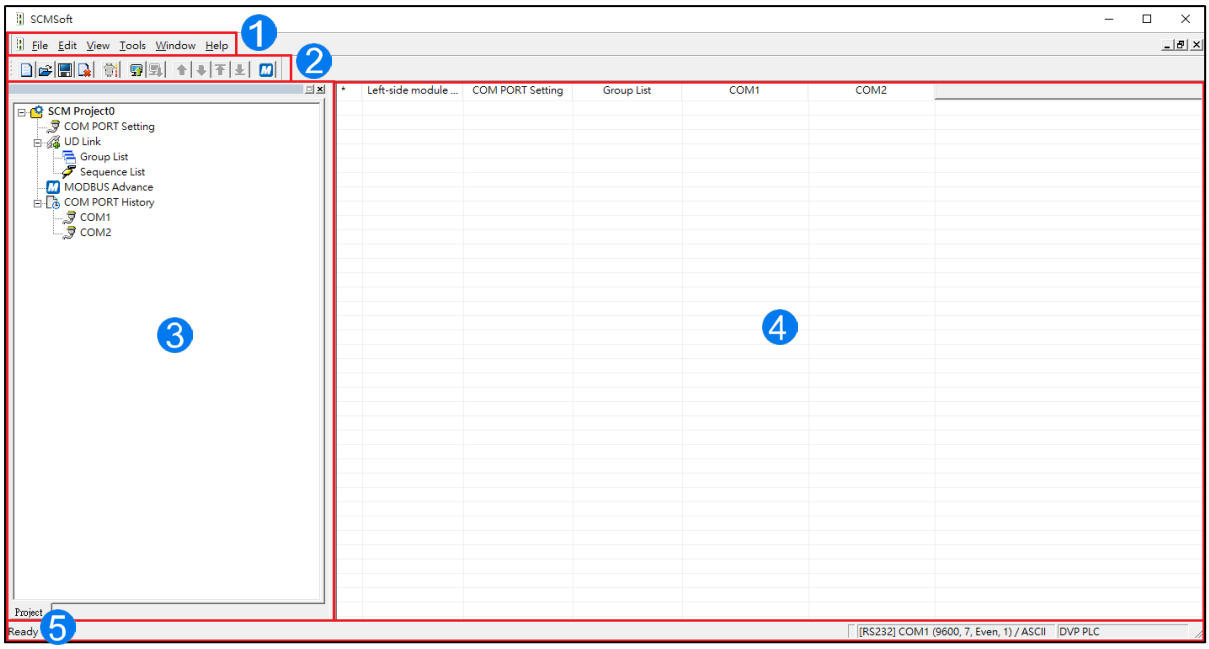
Follow these steps to perform the related setting using **SCMSoft**.

To perform setting using SCMSoft

1. Create **DVP-SV3** / **DVP-SX3** controller in **Project Explorer** and double-click **Controller** > **Hardware Configuration** > **Configuration**.
2. Double-click **+** on the left of the device in **Hardware Configuration** dialog to open **Module List** dialog.
3. Add **DVPSCM12-SL** / **DVPSCM52-SL** module as required from **Module List** dialog, and then click **OK**.
4. Right-click the SCM module, and then click **Communication Software** > **SCMSoft** in the context menu to open **SCMSoft** dialog.



The following table lists the functions in the **SCMSoft** dialog. Refer to *Chapter 5 of the DVPSCM series operation manual* for detailed operation instructions.



Legend	Function	Description
1	Main Menu	• **File**: Provides **Import Project** and **Export Project** functions. • **Edit**: Provides **Cut**, **Copy**, **Paste**, **Delete** functions. • **View**: Displays / Hides **Toolbar**, **Status Bar**, **Workspace** functions. • **Tools**: Provides **SCM Module Information**, **Change PLC Type**, **Return to Default** functions. • **Window**: Select **Cascade**, **Tile**, and **Arrange Icons**. • **Help**: View the software information.

Legend	Function	Description
❷	Toolbar	Provides **Import Project**, **Export Project**, **Communication Setting**, **Upload**, **Download**, **Move Up** / **Move Down**, **Move Top** / **Move Bottom**, **MODBUS Advance Wizard**, etc.
❸	Workspace	Provides **COM PORT Setting**, **UD Link**, **MODBUS Advance**, **COM PORT History** tabs.
❹	Display Area	Displays the corresponding tab information selected from the Workspace.
❺	Status Bar	Displays the status information in this area.

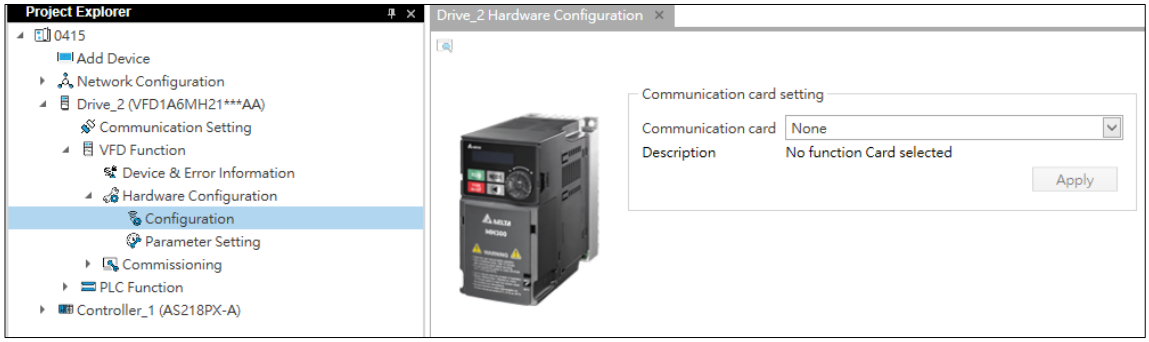
6.1.2 Creating Inverter Configuration

6.1.2.1 Configuration for Single-drive Inverter

This function is used to set communication cards. When the device supports installing communication cards, the option will be available. Follow the two methods below to set communication cards.

1. In **Project Explorer**, navigate inverter device > **VFD Function** > **Hardware Configuration** > **Configuration**.
2. The device, communication card inserted, can perform **Project Upload** (automatically load the communication card setting) after **communication** card setting is complete.

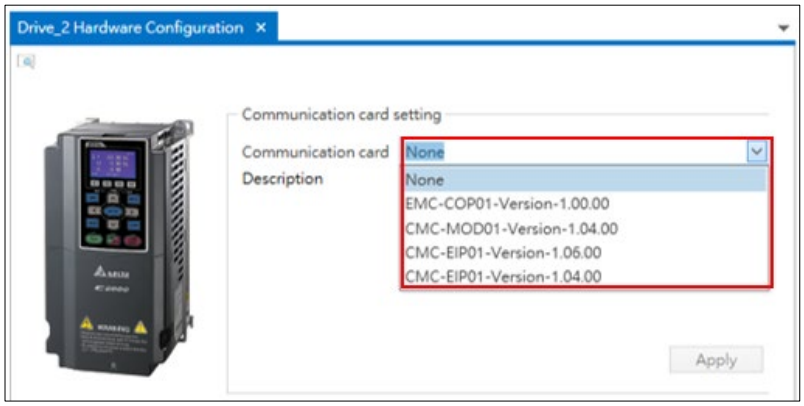
   Select the communication card setting manually or use I/O scan to detect communication card setting after communication card setting is complete.



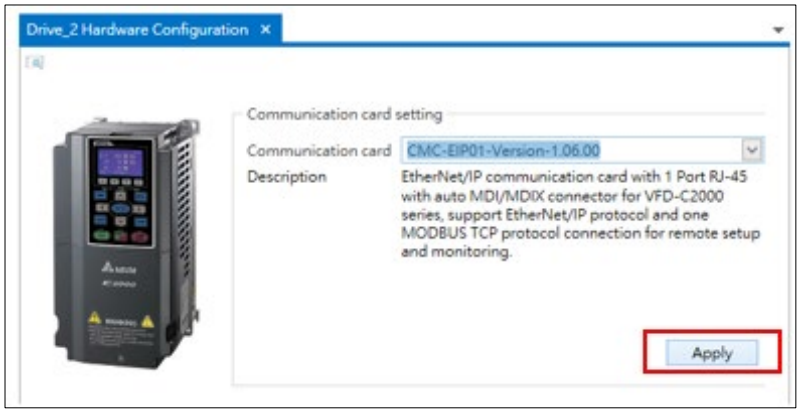
Follow these steps to set communication card.

To set communication card

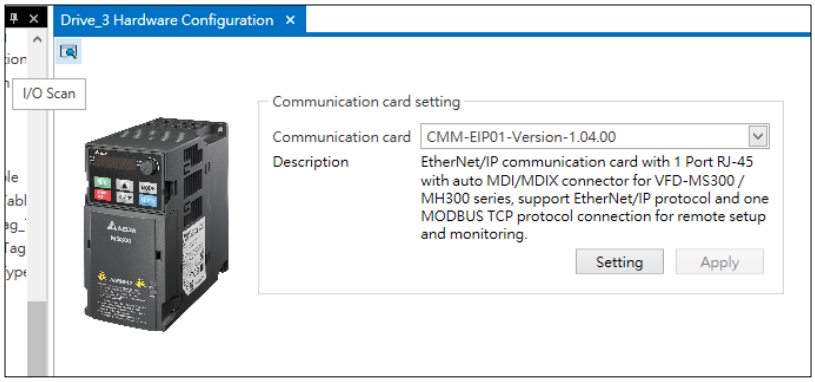
1. Select the communication card from the drop-down list.



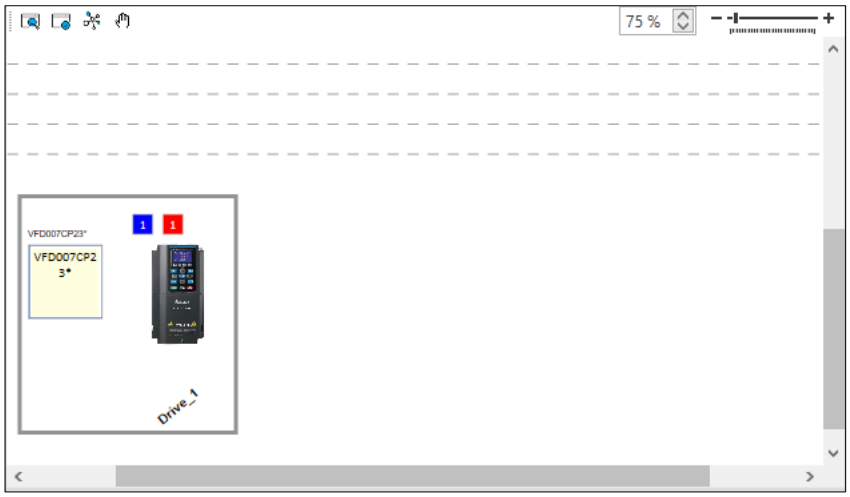
2. Click **Apply**.

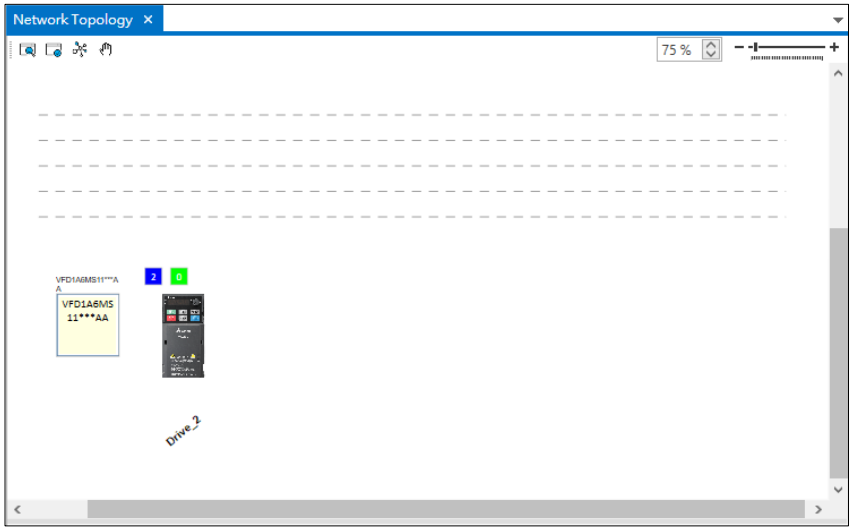


3. After communication card setting is complete, open **Hardware Configuration** view. Click  to perform I/O Scan.



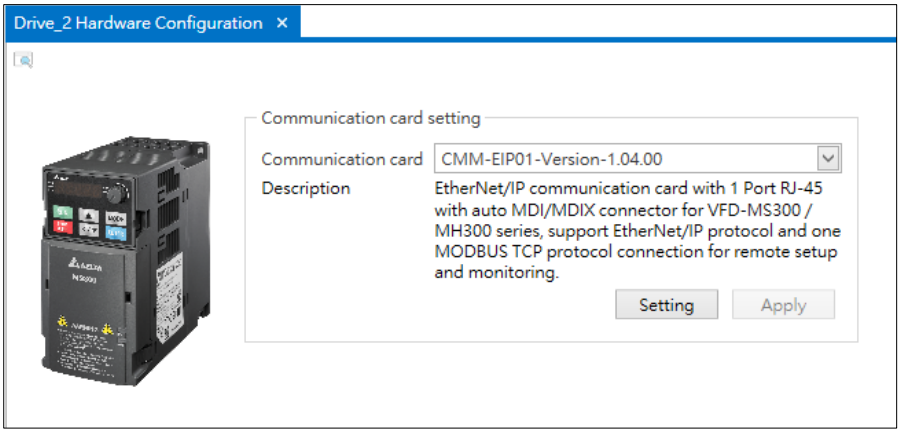
After communication card is inserted, the corresponding ports appear in **Network View**.





Parameter setting of Ethernet communication card

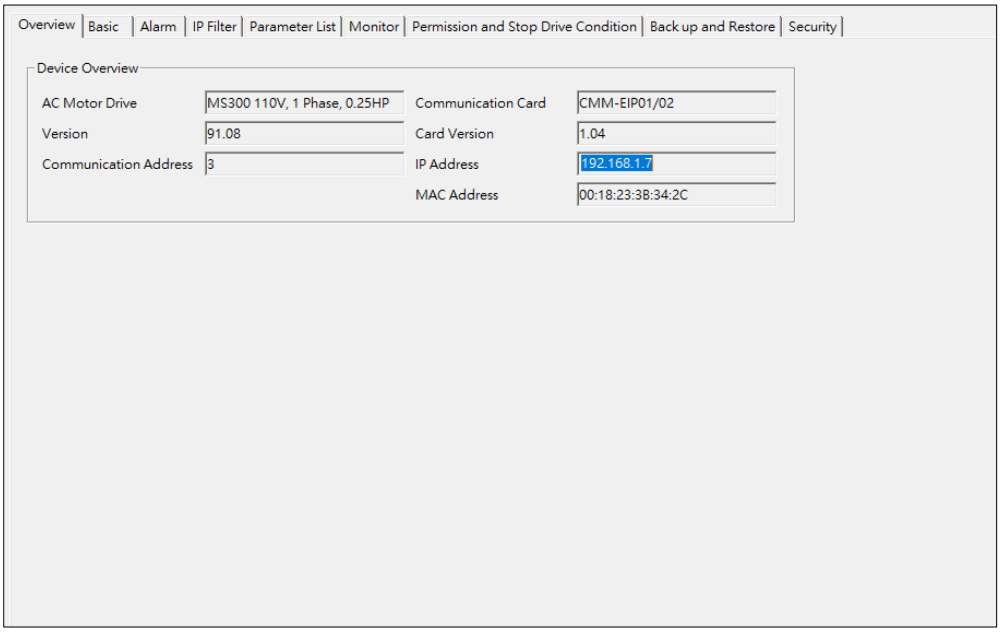
After the Ethernet communication card is inserted into the Inverter, the **Hardware Configuration** view displays the Setting button, as shown in the following figure.



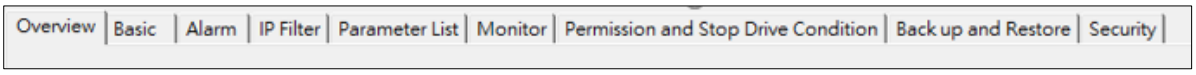
Before setting the communication card, make sure that the PC is connected to the inverter device via communication card and the communication is established between the PC and the device in DIADesigner has been successfully connected.

Click 

button in **Hardware Configuration** to open the function of communication card parameter setting. At normal connection condition, the parameter setting dialog appears, and shows the device content.



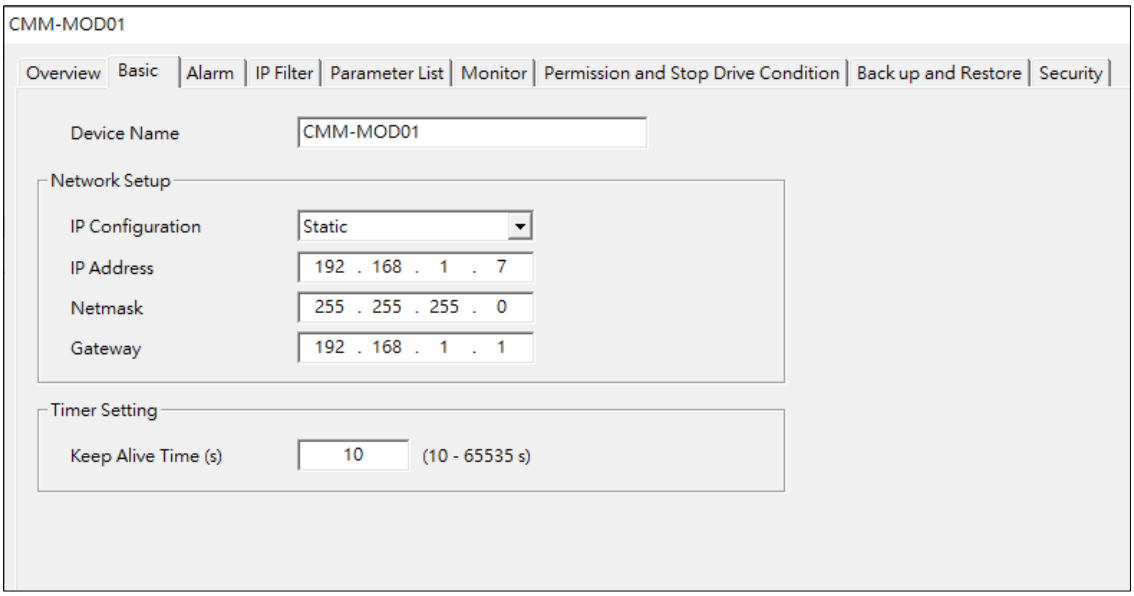
The different tabs in communication card parameter setting dialog are shown below.



After setting the communication card parameter, click **Apply** or **OK**, and the set value will take effect.

1. **Basic**

   Before all network devices connect to the Internet, each of them needs an IP address (Internet Protocol Address), which is like an ID number. Identify the ID of each device on the network by the ID number.



a. Device Name

There might be many CMM-MOD01 cards on the network. To tell which CMM-MOD01 card you want to set, you must set the device name, which helps to distinguish devices during research.

b. IP Configuration

Choose a way to obtain IP, Static IP or DHCP/BOOTP.

- Static IP: user-preset or user manually modify.
- DHCP/BOOTP: automatically update through server; therefore, there must be servers on local network.

c. IP Address

IP Address is a unique address that devices used to identify itself and communicate with other devices in the IP network. Any device connected to the IP network must have a unique IP address within the network. If a wrong IP address is used, the connection might fail, even causing other devices to disconnect.

d. Netmask

Netmask is used to set important parameters of subnet and to determine whether the IP address of the destination device is in the same subnet as the local device. If the destination address is not in the same subnet, the device will send the packet to a gateway, and the gateway will send the packet to

another subnet. If the setting is incorrect, it might cause abnormal communication.

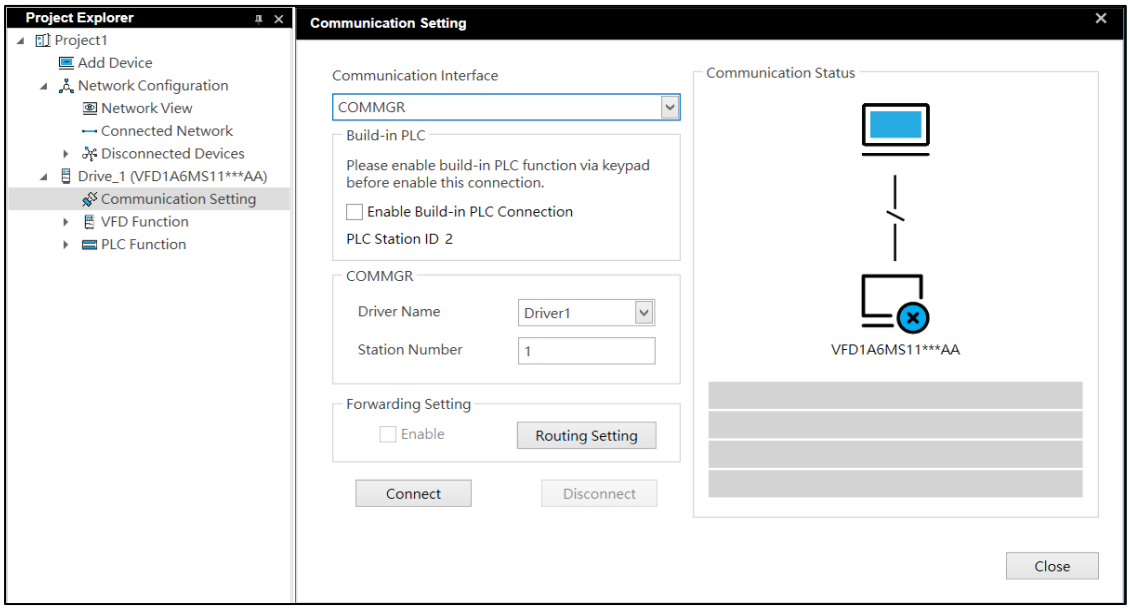
e. Gateway

A gateway is a networking device that connects two networks using different protocols. For instance, local networks need a gateway to connect with Wide Area Network (WAN). Gateway IP and devices must be in the same subnet.

f. Timer Setting

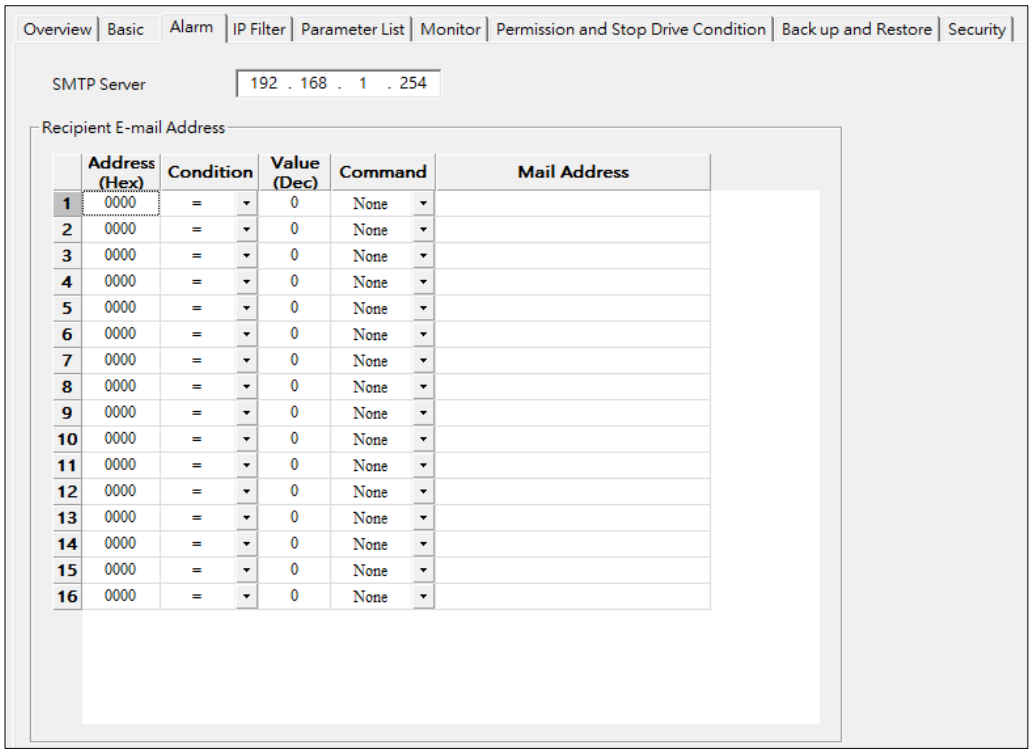
Set communication time.

Note: For correct connection in DIADesigner, after a new IP address is downloaded, click Inverter device > **Communication Setting** > select **COMMGR** from **Communication Interface** drop-down list > select a new IP address from IP Address drop-down list under COMMGR (you must enter the IP address in the IP Address field).



2. **Alarm**

When the condition of triggering email is met, the alarm messages will be sent to you via email.



a. Mail Server (SMTP)

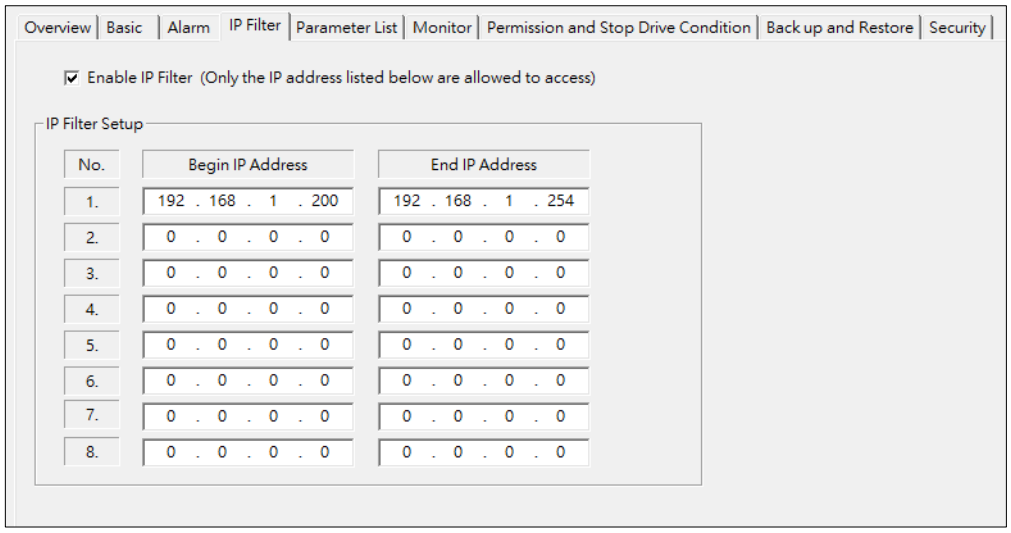
Emails are sent to SMTP, and then SMTP sends the emails to designated addresses. For example, if an email is sent to Test@delta.com.tw, and SMTP IP is “172.16.144.122”; then the email will be firstly sent to SMTP, and SMTP will send it to Test@delta.com.tw.

b. Recipient E-mail Address

When the condition of sending an email is triggered, the email will be sent to the recipient, which you set.

3. **IP Filter**

IP filter is used to restrict network connection list there is uncertain / error IP communication. The connection can be established only in the range that you set, and the rest IP addresses will be rejected.



a. Enable IP Filter

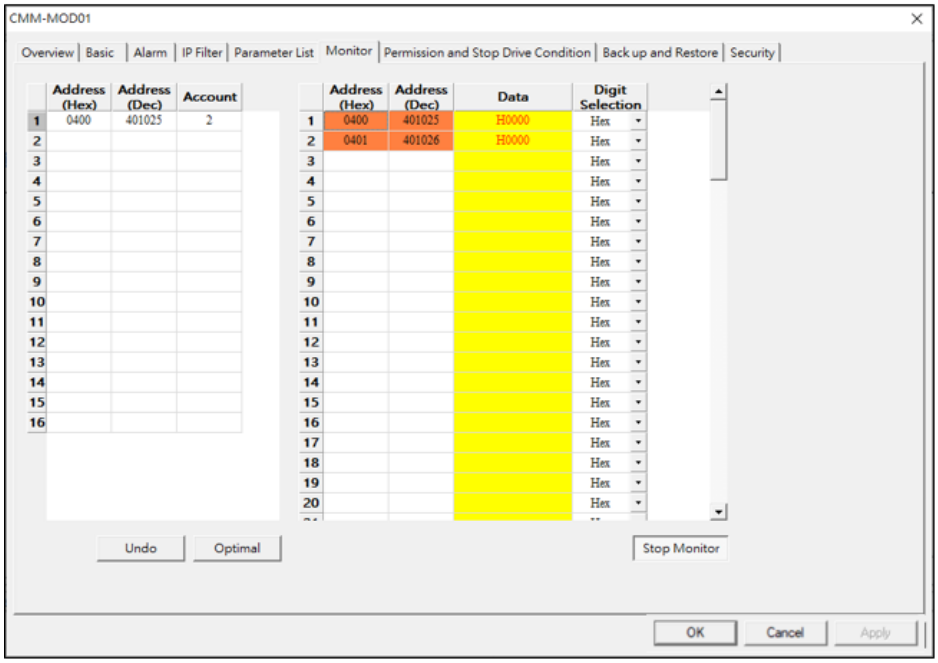
Click the checkbox to enable IP filtering. The filter function starts working according to user-defined setting.

b. IP Filter Setup

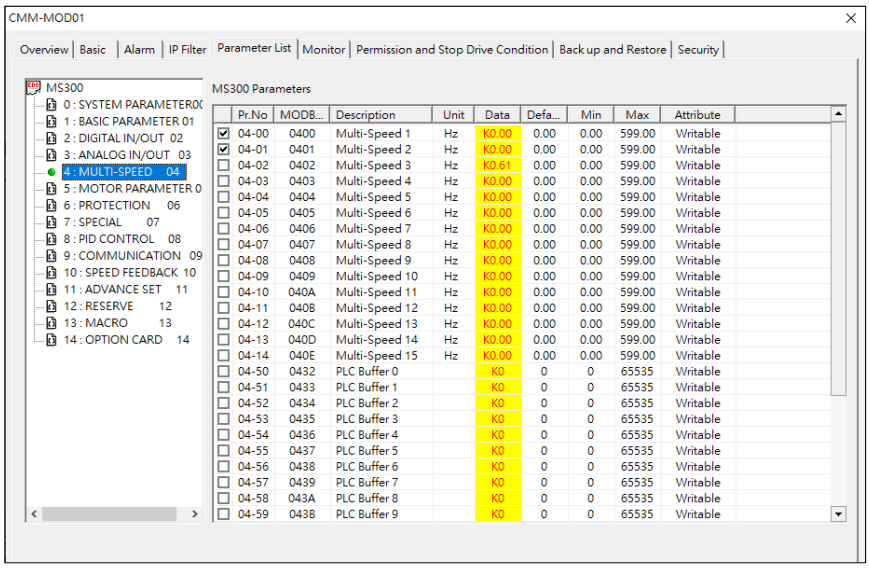
Up to eight IP address ranges can be set to establish connection at a time.

4. **Monitor**

Provides real-time parameter monitoring of devices, and decimal / hexadecimal register present value monitoring per requirement.



a. Select the parameters that you would like to monitor from the parameter list. Up to 16 sets of discontinuous parameters can be selected.

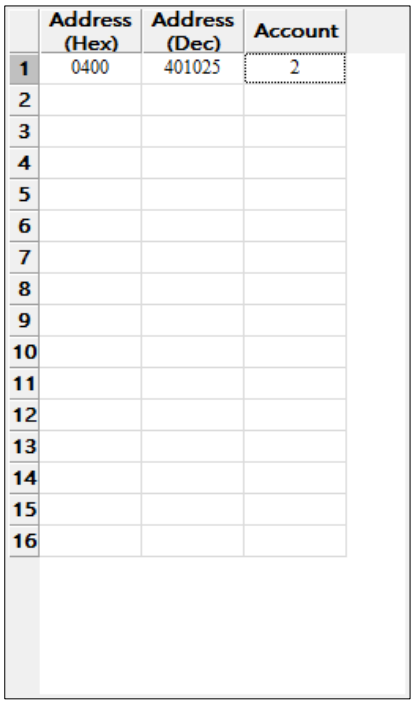


b. Switch to the **Monitor** tab. You can see the parameter addresses that just selected, including the following:

i. Address (Hex): 4-digit hexadecimal communication address of parameters.

ii. Address (Dec): 6-digit decimal communication address of parameters.

iii. Account: the continuous parameters that you want to monitor (Max. 100).

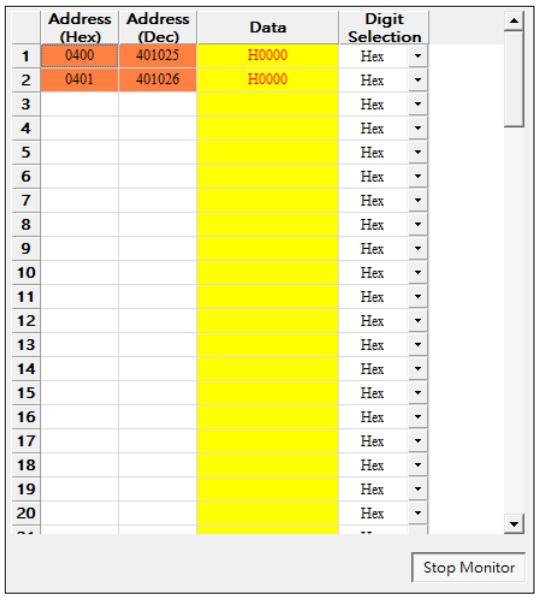


	Address (Hex)	Address (Dec)	Account
1	0400	401025	2
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			

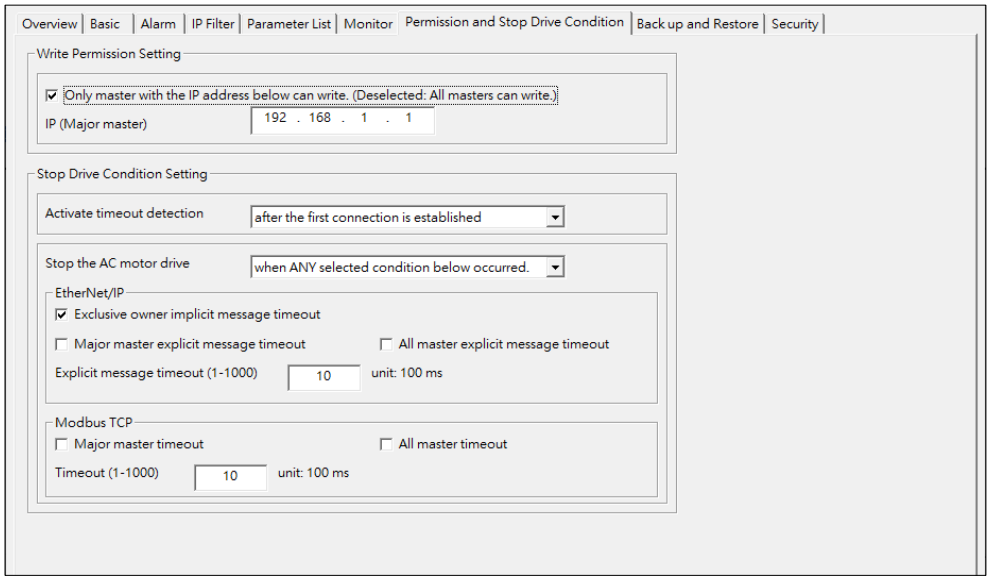
c. Click **Apply** to save the parameter addresses to be monitored to the communication card, then directly start monitoring. The content includes the following:

i. Digit Selection: From the drop-down list, select decimal or hexadecimal registers.

ii. Start / Stop Monitor: Start/stop the saved parameter monitoring; the unsaved parameters will be deleted.



5. **Permission and Stop Drive Condition Setting**



a. Grant write permission: Enable **Only master with the IP address below can write**. Only IP major master has written permission.

b. Timeout stop setting: Select the start conditions of stopping the inverter. Separately choose “timeout setting & the stop conditions of communication timeout” for “EtherNet/IP & Modbus TCP”.

6. **Backup and Restore**

In the **Backup and Restore** tab, back up the current parameter files or replace the current parameter setting by reading parameter files.

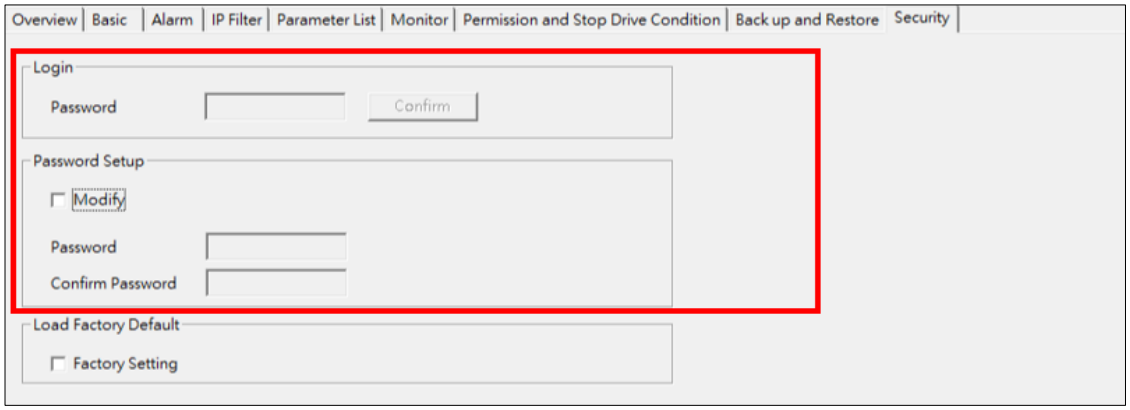


a. Backup to File: Click the **Backup to File** tab to display the **Save As** dialog, where you can choose the parameter file storage location and save the current setting to the parameter file.

b. Restore from File: Click **Restore from File** to open **Restore Option** dialog, and click **OK**, the **Open** dialog appears, where you can open a parameter file. The set value of this parameter file will replace the current setting.

7. **Security**

In the **Security** tab, set password to prevent parameters from being modified.



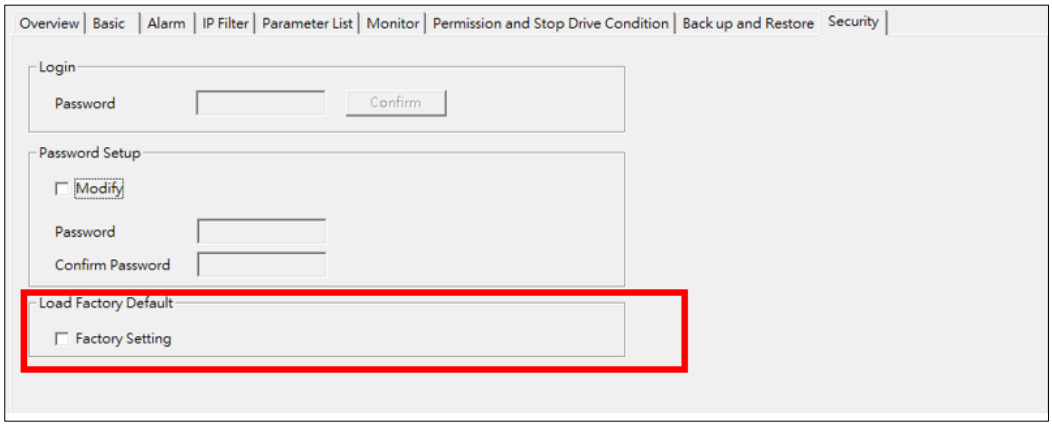
a. Password Setup: Check the modify box and enter password.

b. Confirm Password: Re-enter password.

c. Login: After password setting is saved in the communication card, every time you open the parameter setting page of communication card, the login password needs to be confirmed to unlock the lock state.

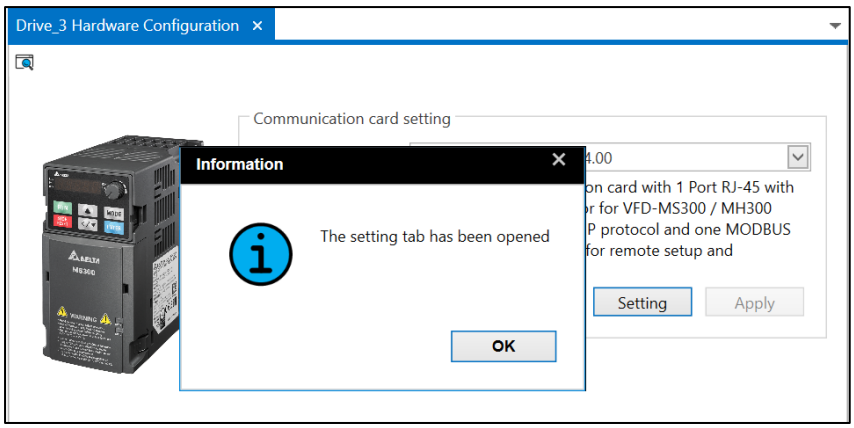
Note: Once the password state is “locked”, all parameters cannot be set/changed until the password is “unlocked”. If password is lost, by serial port (RS-485) connection, you can restore the system to factory setting to unlock the “locked” state, and all parameters will be back to the default values.

d. Load Factory Default: To make all parameters back to the default values, check the **Factory Setting** box.



Note: While returning to factory setting through serial port (RS-485) communication, modules need 10 seconds to change the setting. Do not turn off the power at this time.

After opening the parameter setting of the communication card, the **Setting** button in the hardware configuration of DIADesigner cannot be clicked. Until the parameter setting page of communication card is all closed, **Setting** will be enabled again. Click **Apply** or **OK** after communication card parameter setting is complete. The changes will be reflected once the parameters are downloaded to the device.

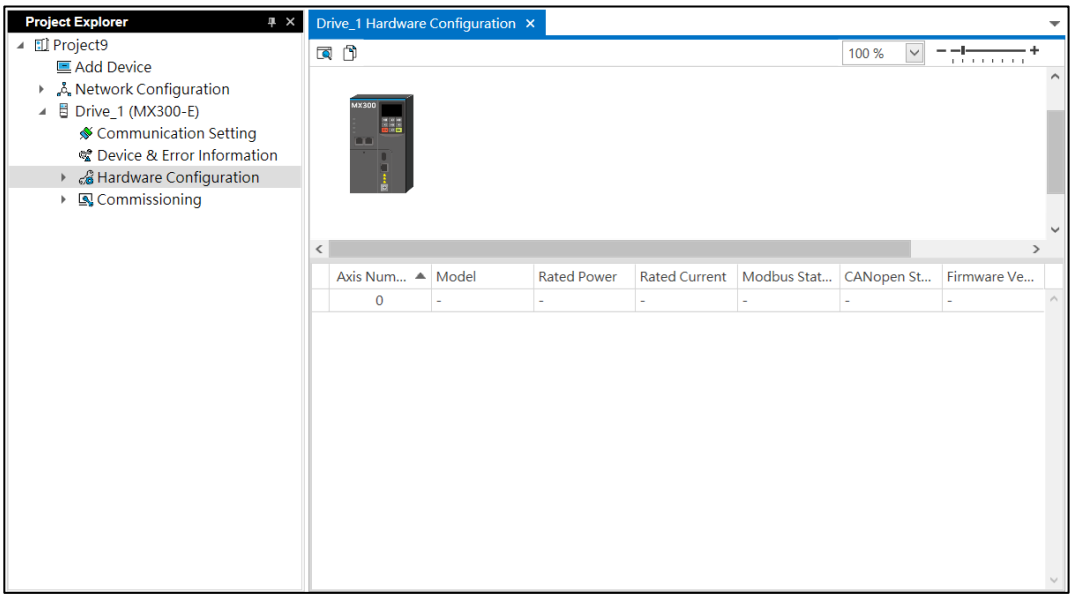


6.1.2.2 Configuration for Multi-drive Inverter

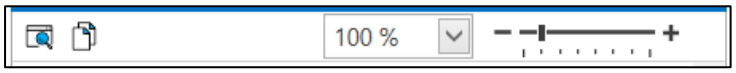
This function is used to set the configuration of the multi-drive inverter. The multi-drive inverter includes rectifier and inverter modules. After adding the rectifier module in Creating Project and Device, add the relevant inverter module in **hardware Configuration**. The following is an example of MX300-E. Follow these steps to enable **Configuration**.

To enable Configuration

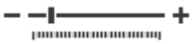
1. In **Project Explorer**, navigate inverter device > **VFD Function** > **Hardware Configuration** > **Configuration**.
2. Double-click **Configuration** to open Hardware Configuration dialog. Upper pane displays hardware view and bottom pane displays Configuration Table as shown in the following figure.



The following figure shows **Hardware Configuration** toolbar.



The following table describes the functions in **Hardware Configuration**.

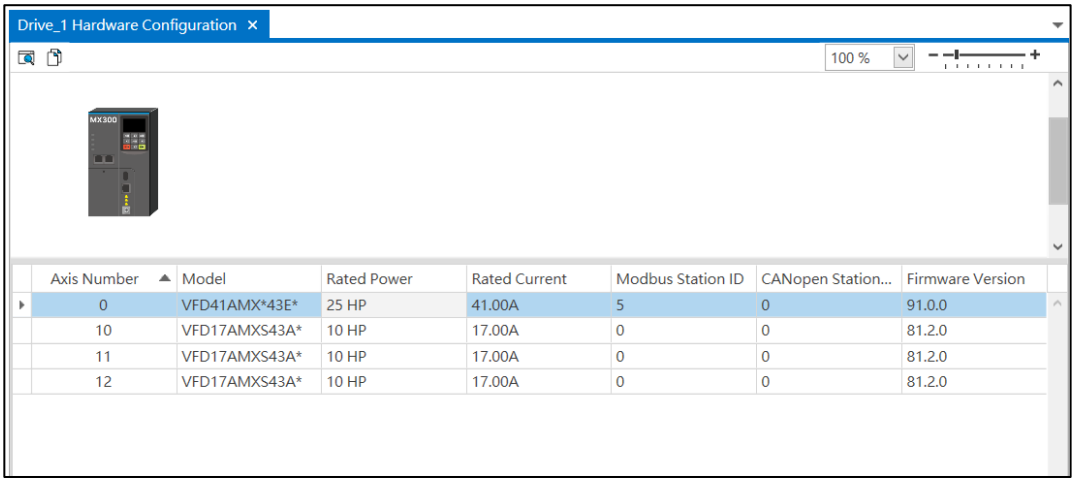
Function	Description
	Click to perform I/O scan.
	Click to perform I/O match.
	Enter or select value to zoom-in or zoom-out.
	Click and drag to zoom-in and zoom-out.

Adding inverter module

The multi-drive inverter hardware configuration only supports the configuration returned by the actual connected device after online scanning. It does not support offline configuration. First confirm that the device has completed communication settings and connection. Then follow these steps to do module configuration.

To do module configuration

1. Click **I/O Scan** on the toolbar, and DIADesigner displays the current configuration in the configuration table below, as shown in the following figure.



Axis Number	Model	Rated Power	Rated Current	Modbus Station ID	CANopen Station...	Firmware Version
0	VFD41AMX*43E*	25 HP	41.00A	5	0	91.0.0
10	VFD17AMXS43A*	10 HP	17.00A	0	0	81.2.0
11	VFD17AMXS43A*	10 HP	17.00A	0	0	81.2.0
12	VFD17AMXS43A*	10 HP	17.00A	0	0	81.2.0

2. After completing the scan, click **Save Project** to save the module configuration in the project. If the actual module configuration changes in the future, repeat the above I/O scan and update it to the current module configuration.

Match Module Configuration

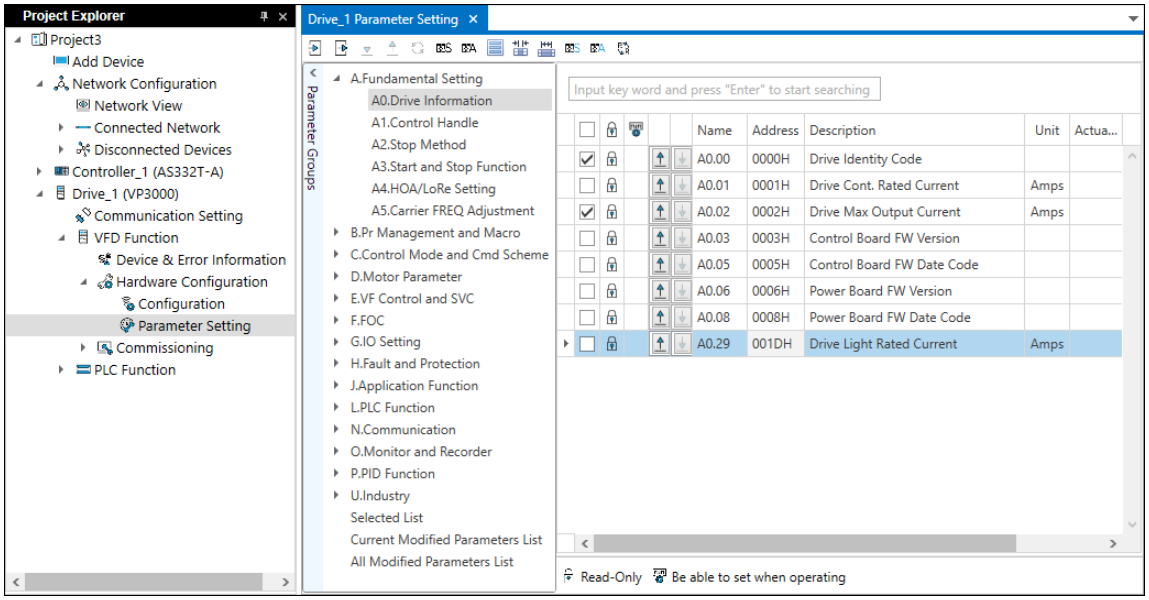
When the previously scanned inverter module configuration has been saved in the project, press the **I/O Match** button on the toolbar to confirm whether DIADesigner matches the currently connected device. If it does not match, decide whether to re-perform the **I/O scan** or change the actual hardware installation to match DIADesigner.

6.2 Parameter Setting

The **Parameter Setting** function is used to set the product parameter.

How to open **Parameter Setting** page?

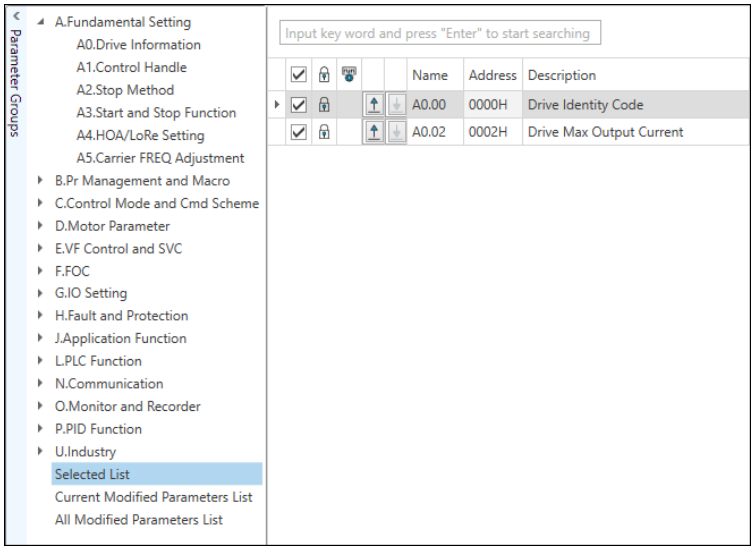
- In **Project Explorer**, double-click **Device** > **Hardware Configuration** > **Parameter Setting**.



The functions on the **Parameter Groups** pane of Parameter Setting are displayed in a tree structure. They include Selected List, Current Modified Parameters List, All Modified Parameters List, and so on.

- **Selected List**

  When selecting any categories of parameters, these parameters will be saved in the **Selected List**.



- **Current Modified Parameters List**

  After modifying the previously saved parameters, these parameters will be listed in the **Current Modified Parameters List** and cleared when the project is closed.

- **All Modified Parameters List**

  After creating a project, all previously modified parameters are listed in the **All Modified Parameters List**, and the list will remain even if the project is closed.

6.2.1 Parameter Setting for Controller Devices

After the hardware configuration is completed, set the CPU and module parameters through **Parameter Setting**. For more information on the hardware and operations of the CPU or modules, refer to Catalogs, Hardware Manual and Operation Manual.

Note: Before setting the parameters of the CPU and module, you must refer to the operation manual of each model and make sure whether the set parameters that affect the CPU module, the whole system or personnel. The parameter settings of CPU and module will only take effect after the communication download is sucessful.

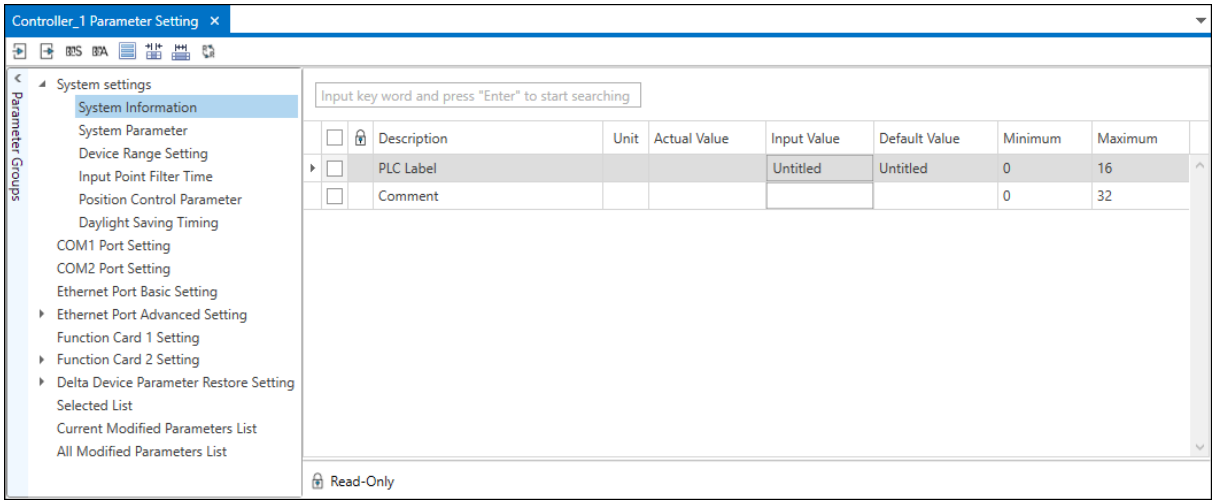
How to open the controller **Parameter Setting** page?

- Double-click **Controller** > **Hardware Configuration** > **Parameter Setting**.

Or

- Double-click the CPU or modules in the **Hardware Configuration** page.

The following table lists the functions on the toolbar in the controller **Parameter Setting** page.



Function	Description
	Click to import parameter file.
	Click to export the parameter file.
	Click to write the set parameters instantly. **Note**: Available only for the DVP series module.
	Click to read the current parameters instantly. **Note**: Available only for the DVP series module.
	Click to replace the selected parameter with its default value.
	Click to reset all the parameters to their default values.
	Click to compare the parameters of CPU.
	Enter a keyword, and the words associated with the keyword will be highlighted in yellow.

Parameters are classified into group tabs in the **Parameter Setting** dialog and divided into primary and secondary tabs. Parameters vary upon the controller module series and the selected modules.

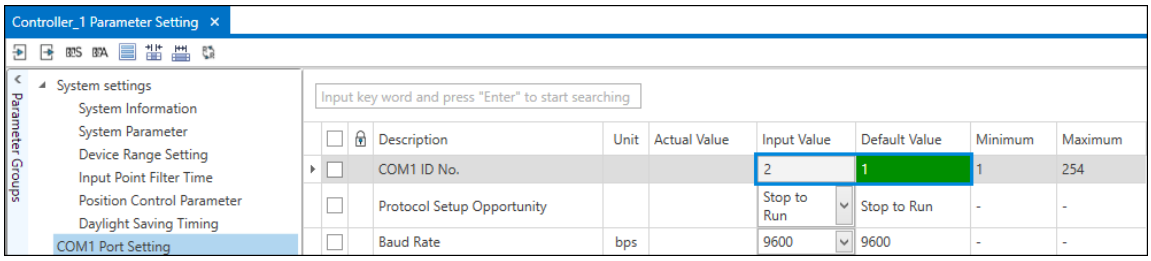
The following table lists the settings in the controller **Parameter Setting** page.

Setting	Description
Description	Displays the parameter description.

Setting	Description
Unit	Displays the parameter unit.
Actual Value	Displays the parameter actual value after upload / download.
Input Value	Enter the parameter value.
Default Value	Displays the default value of parameter.
Minimum	Displays the minimum value of parameter.
Maximum	Displays the maximum value of parameter.

Do the following to edit the controller parameters.

- Enter the parameter in the **Input Value** field. If the **Input Value** is different from the **Default Value**, then the **Default Value** column will be highlighted in green.



Note: The number of connected remote modules can only be modified in the **Hardware Configuration** page. **Parameter Setting** > **Function Card Setting** > **AS Remote Module No.** option does not support this feature yet.

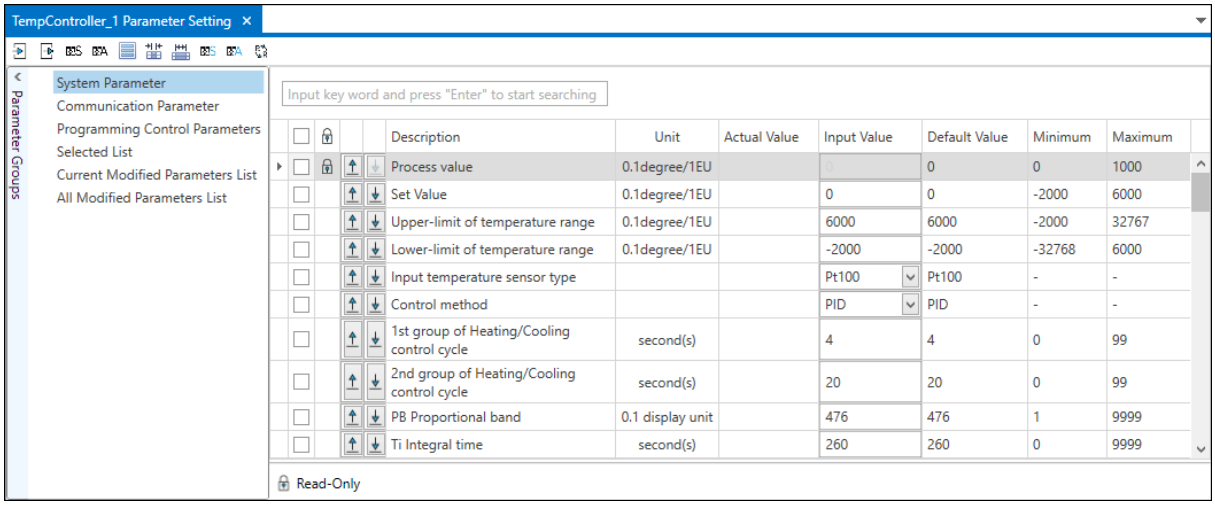
6.2.2 Parameter Setting for Temperature Controller

DIADesigner supports the relevant parameter setting of Delta temperature controller DTC series.

How to open the temperature controller **Parameter Setting** page?

- In **Project Explorer**, double-click **Temperature Controller** > **Parameter Setting**.

The following table lists the functions on the toolbar in the temperature controller **Parameter Setting** page.



Function	Description
	Click to import parameter file.
	Click to export the parameter file.
	Click to replace the default parameter input value with the default value.
	Click to replace all parameter input values with their default values.
	Click to replace the default parameter input value with the actual value.
	Click to replace all parameter input values with their actual values.
	Click to compare the parameters of CPU.
	Enter a keyword, and the words associated with the keyword will be highlighted in yellow.

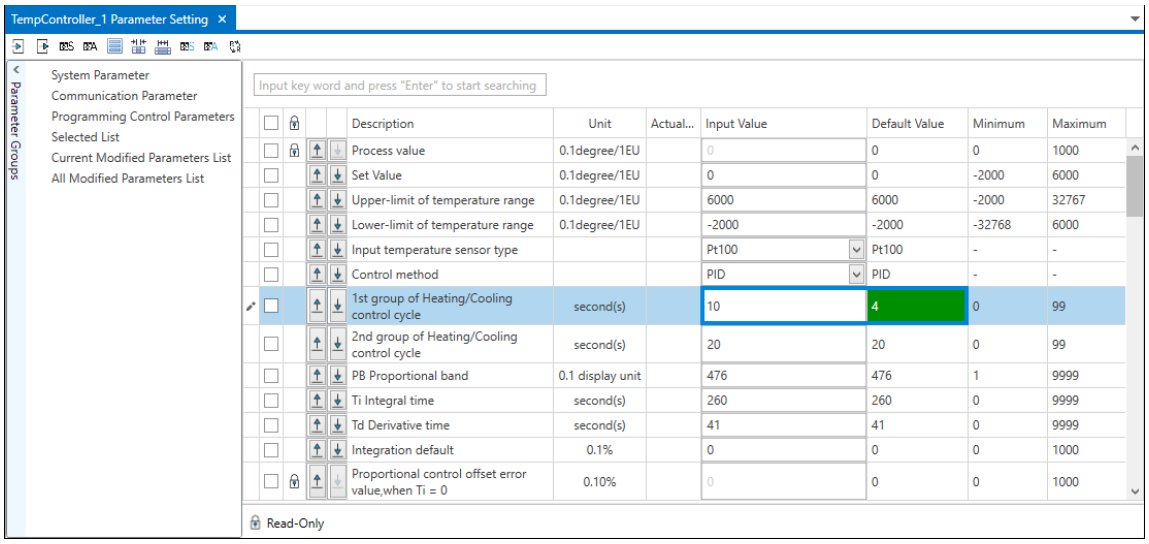
The following table lists the settings in the temperature controller **Parameter Setting** page.

Setting	Description
	The parameter is a read-only parameter.
	The parameter can be set during operation.
	Single upload / download the parameter.
Description	Displays the parameter description.
Unit	Displays the parameter unit.
Actual Value	Displays the parameter actual value after upload / download.

Setting	Description
Input Value	Enter the parameter value.
Default Value	Displays the default value of the parameter.
Minimum	Displays the minimum value of the parameter.
Maximum	Displays the maximum value of the parameter.

Do the following to edit the temperature controller parameters.

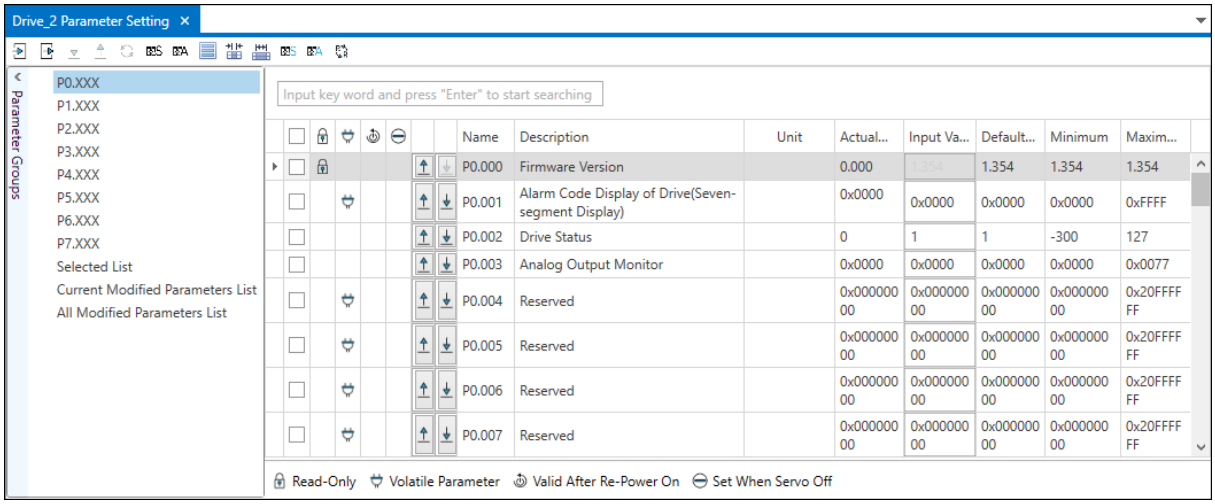
- Enter the parameter in the **Input Value** field. If the **Input Value** is different from the **Default Value**, then the **Default Value** column will be highlighted in green.



6.2.3 Parameter Setting for Servo Drive

How to open the servo device **Parameter Setting** page?

- Double-click **Servo Drive** > **Parameter Setting** in **Project Explorer**.



The following table lists the functions on the toolbar in the servo drive **Parameter Setting** page.

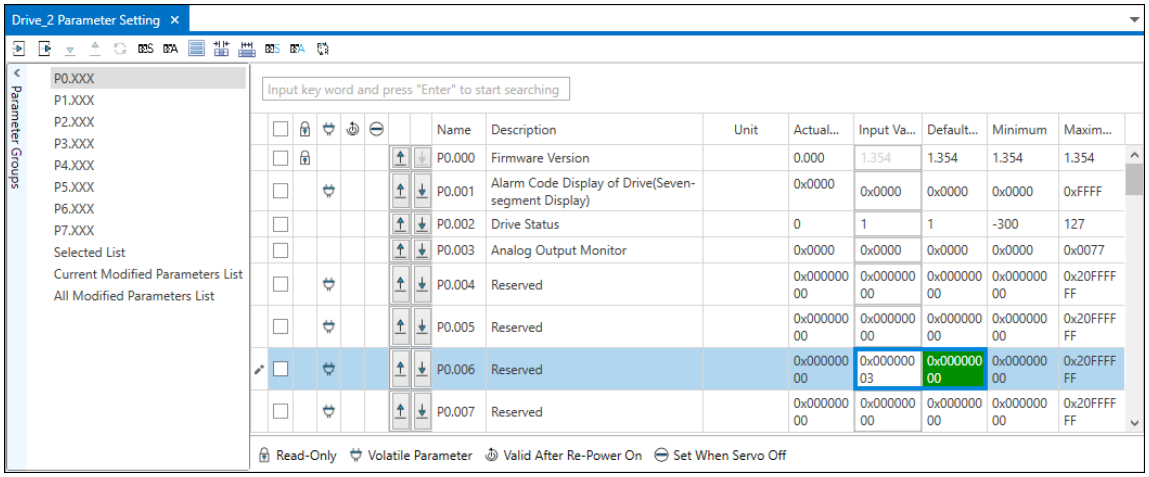
Function	Description
	Click to import parameter file.
	Click to export the parameter file.
	Click to download the parameters.
	Click to upload the parameters.
	Click to continuously update the parameters one-by-one for the display group.
	Click to replace the default parameter input value with the default value.
	Click to replace all parameter input values with their default values.
	Click to replace the default parameter input value with the actual value.
	Click to replace all parameter input values with their actual values.
	Click to compare the parameters of CPU.
	Enter a keyword, and the words associated with the keyword will be highlighted in yellow.

The following table lists the settings in the servo drive **Parameter Setting** page.

Setting	Description
	The parameter is a read-only parameter.
	The parameter will not be remembered after the device is powered off.
	The parameter value takes effect after the parameter is set and the device is restarted.
	Th parameter cannot be written while servo on.
	Single upload / download the parameter.
Name	Displays the parameter number.
Description	Displays the parameter description.
Unit	Displays the parameter unit.
Actual Value	Displays the actual parameter value after upload / download.
Input Value	Enter the parameter value.
Default Value	Displays the default value of parameter.
Minimum	Displays the minimum value for parameter.
Maximum	Displays the maximum value for parameter.

Do the following to edit the servo dirve parameters.

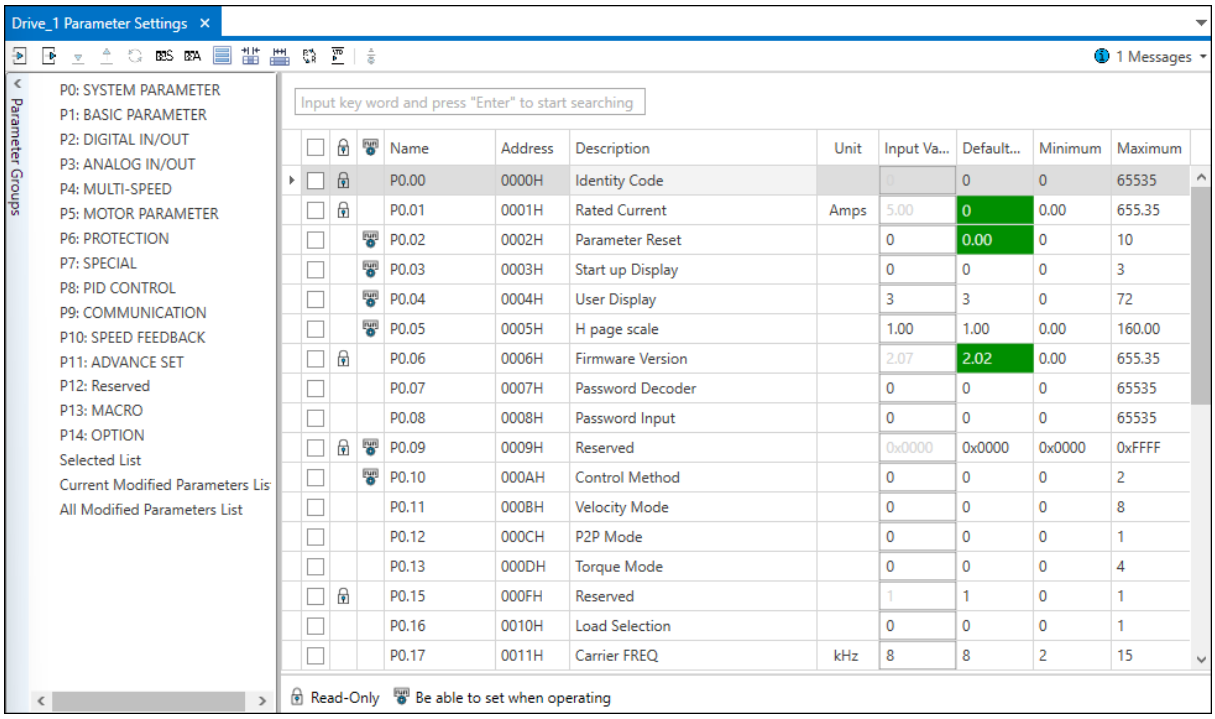
- Enter the parameter in the **Input Value** field. If the **Input Value** is different from the **Default Value**, then the **Default Value** column will be highlighted in green.



6.2.4 Parameter Setting for Inverter

How to open the inverter drive **Parameter Setting** page?

- Double-click **Inverter Drive** > **Hardware Configuration** > **Parameter Setting** in **Project Explorer**.



The following table lists the functions on the toolbar in the inverter drive **Parameter Setting** page.

Function	Description
	Click to import parameter file.
	Click to export the parameter file.
	Click to download the parameters.
	Click to upload the parameters.
	Click to continuously update the parameters one-by-one for the display group.
	Click to write the selected parameter to the default value.
	Click to write all parameters to the default values.

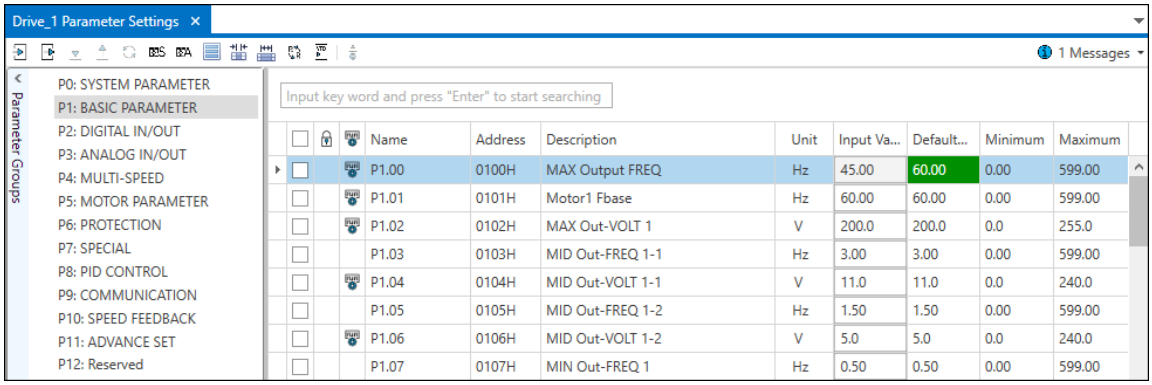
Function	Description
	Click to open the column to be displayed on the parameter settings table.
	Click to simply display the columns of the parameter settings table.
	Click to reset the parameter settings table columns to default.
	Click to compare the parameters of CPU.
	Click to import VFDSoft parameter file.
	Click to upload configuration file of parameters. Refer to Upload Configuration File of Parameters.
	Click to expand the current related message.
	Enter a keyword, and the words associated with the keyword will be highlighted in yellow.

The following table lists the settings in the inverter drive **Parameter Setting** page.

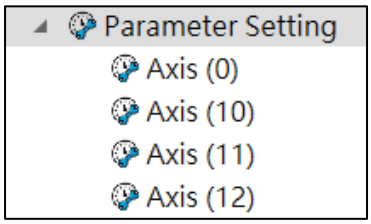
Setting	Description
	The parameter is a read-only parameter.
	Th parameter can be set when operating.
	Single upload / download the parameter.
Name	Displays the parameter number.
Address	Displays the memory address of parameter.
Description	Displays the parameter description.
Unit	Displays the parameter unit.
Input Value	Enter the parameter value.
Default Value	Displays the default value of parameter.
Minimum	Displays the minimum value for parameter.
Maximum	Displays the maximum value for parameter.

Do the following to edit the inverter drive parameters.

- Enter the parameter in the **Input Value** field. If the **Input Value** is different from the **Default Value**, then the **Default Value** column will be highlighted in green.



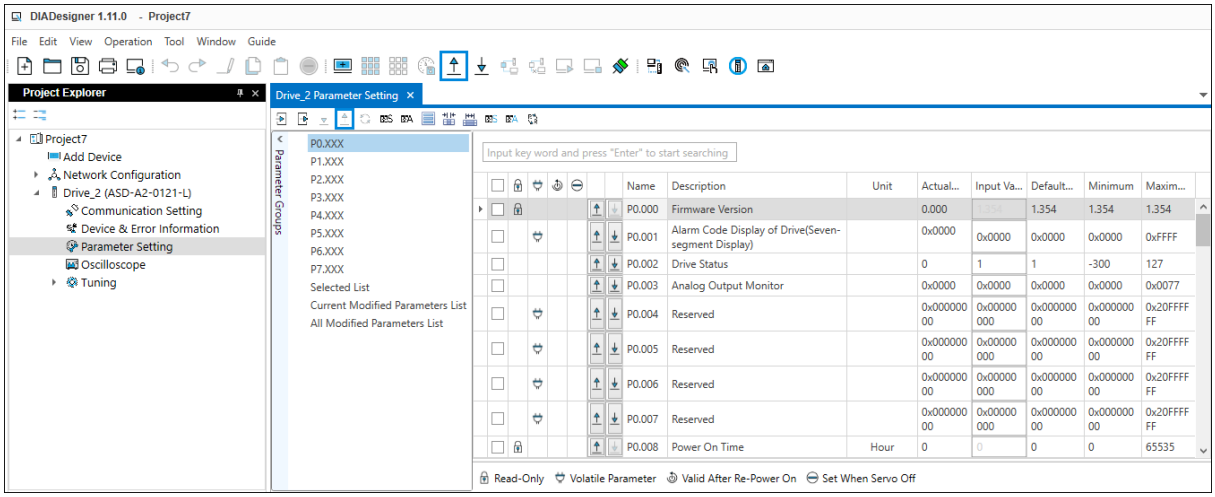
If you are using a multi-drive inverter, you need to first complete the module scan configuration (refer to Configuration for Multi-drive Inverter), and then double-click **Inverter** device > **VFD Function** > **Hardware Configuration** > **Parameter Setting** in **Project Explorer**, and the parameter editor for each axis of the current configuration inverter module will automatically list. Double-click the axis to be set to open its **Parameter Setting** page.



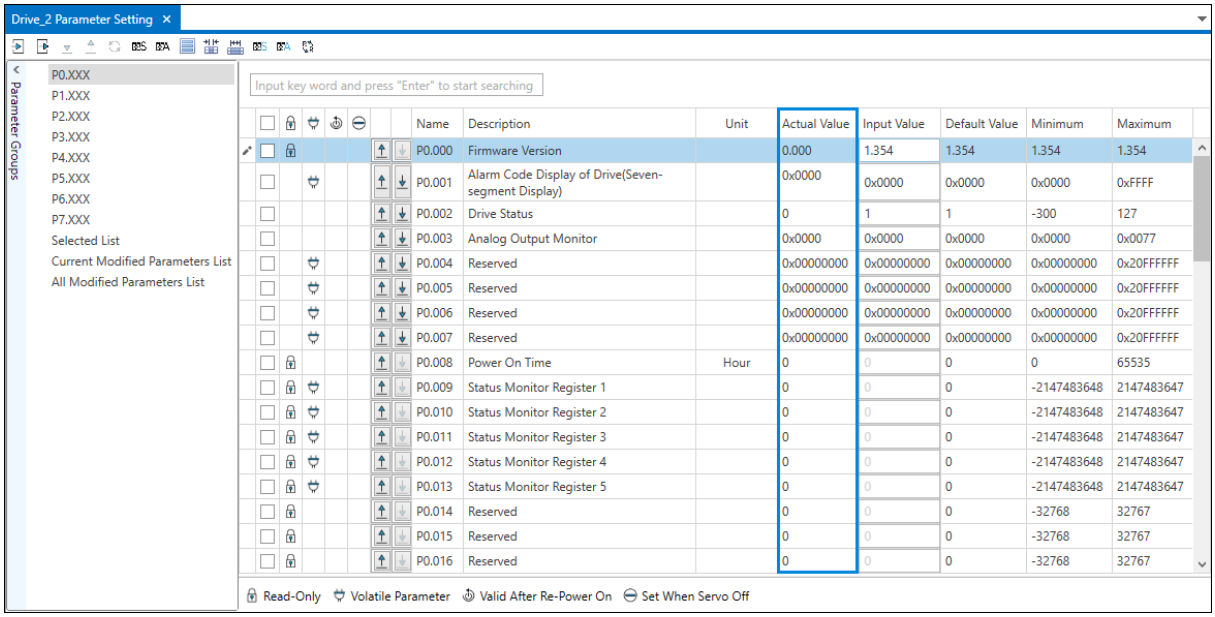
6.2.5 Perform Parameter Setting After Servo Drives / Inverters Upload Parameters

Upload servo drives to read parameter actual values and edit parameters

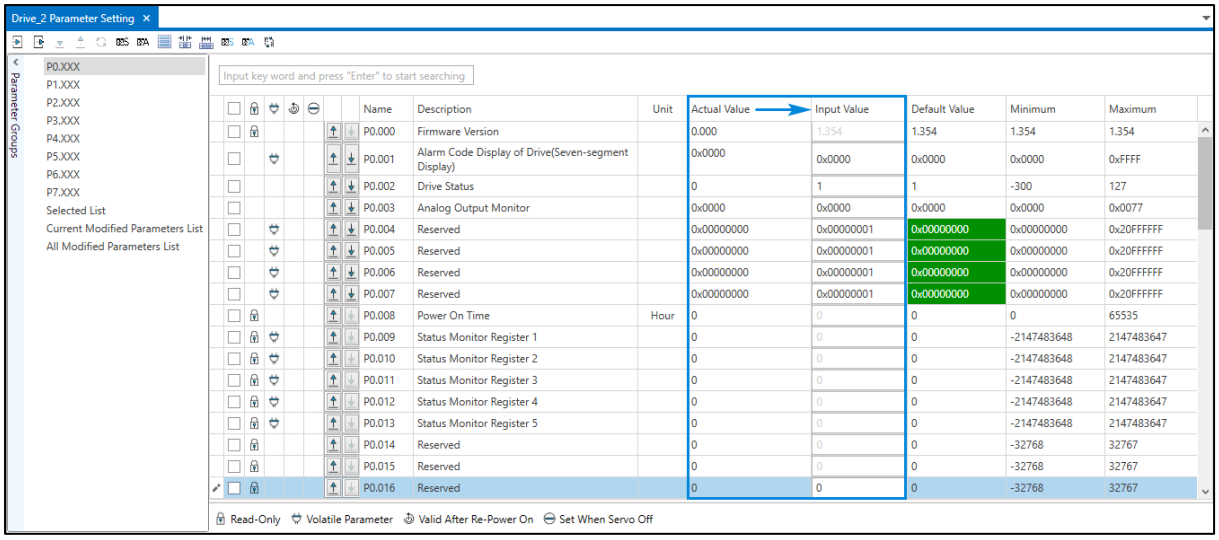
After communication is established, click  to read the parameter actual values.



Once the upload is complete, the readback parameter values will be written to the **Actual Value** column at the same time.

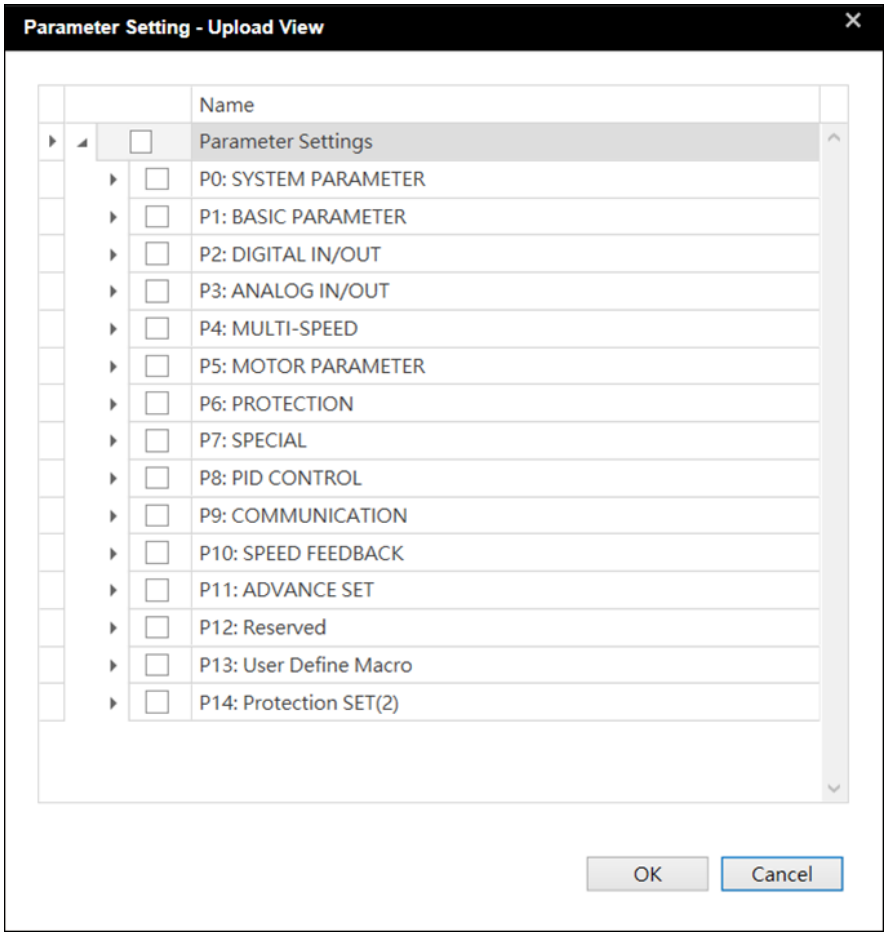


If you want to export the parameters or edit the parameters, then download to the device, click **Replace the selected parameters with the actual value** or **Replace all parameters with the actual value**. The **Actual Value** will be written to **Input Value**.



Upload inverters to read parameter actual values and edit parameters

After communication is established, click  to select the parameter to be uploaded in the **Parameter Setting - Upload View** dialog.



The read parameters are displayed in the **Input Value** fields.

Name	Address	Description	Unit	Input Value	Default Value	Minimum	Maximum
P1.00	0100H	MAX Output FREQ	Hz	60.00	60.00	0.00	599.00
P1.01	0101H	Motor1 Fbase	Hz	60.00	60.00	0.00	599.00
P1.02	0102H	MAX Out-VOLT 1	V	200.0	200.0	0.0	255.0
P1.03	0103H	MID Out-FREQ 1-1	Hz	3.00	3.00	0.00	599.00
P1.04	0104H	MID Out-VOLT 1-1	V	11.0	11.0	0.0	240.0
P1.05	0105H	MID Out-FREQ 1-2	Hz	1.50	1.50	0.00	599.00
P1.06	0106H	MID Out-VOLT 1-2	V	5.0	5.0	0.0	240.0
P1.07	0107H	MIN Out-FREQ 1	Hz	0.50	0.50	0.00	599.00
P1.08	0108H	MIN Out-VOLT 1	V	1.0	1.0	0.0	240.0
P1.09	0109H	Start Frequency	Hz	0.50	0.50	0.00	599.00
P1.10	010AH	Upper Bound FREQ	Hz	599.00	599.00	0.00	599.00
P1.11	010BH	Lower Bound FREQ	Hz	0.00	0.00	0.00	599.00
P1.12	010CH	1st ACC Time	sec	10.00	10.00	0.00	600.00
P1.13	010DH	1st DEC Time	sec	10.00	10.00	0.00	600.00
P1.14	010EH	2nd ACC Time	sec	10.00	10.00	0.00	600.00

You can export the parameters or edit the parameters, and then download them to the device.

6.2.6 Upload Configuration File of Parameters

To avoid being unable to use the software when the parameter version built-in the software does not match or has not been updated with the currently connected device parameter version, use **Upload Configuration File of Parameter** function to read the currently connected device parameter configuration file.

Prerequisites

- The software and the device are connected.

Follow these steps to execute parameter configuration upload.

To execute parameter configuration upload

1. In **Project Explorer**, double-click **inverter** > **VFD Functions** > **Hardware Configuration** > **Parameter Editor**.
2. Click  on the toolbar in the **Parameter Editor** dialog.

   The configuration file of currently connected device is uploaded to the software and presented in the parameter table.

Note:

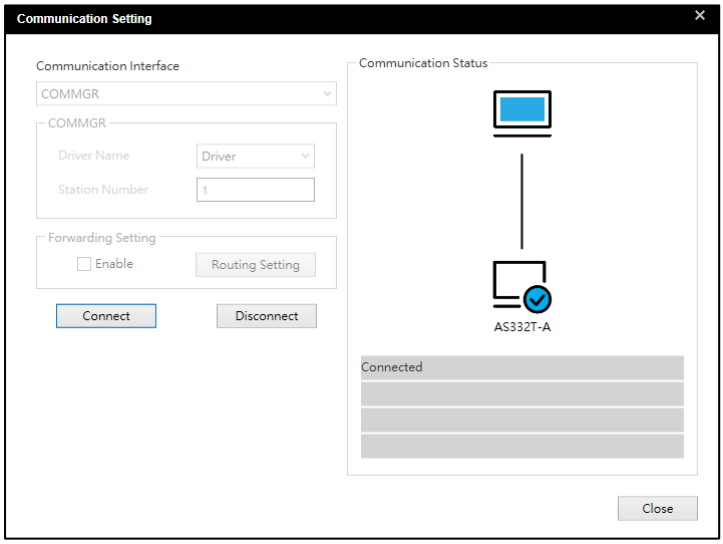
- The **Parameter Upload** function uploads the actual value of the parameters while the **Upload Configuration File of Parameter** function uploads the configuration file of the parameter table including form options, range of upper and lower limits, and current actual values (excluding user-entered setting values) so that the software can read and write the parameters of the current version on the device.
- The **Upload Configuration File of Parameter** function currently supports C2000, C2000 Plus, C2000-HS, CFP2000, CH2000, CP2000, ME300, MH300, MS300, MH300-L, MPD, MP300 models.

6.3 Hardware Configuration Online Mode

The Online mode of Hardware Configuration allows you to monitor devices in real time.

Prerequisites

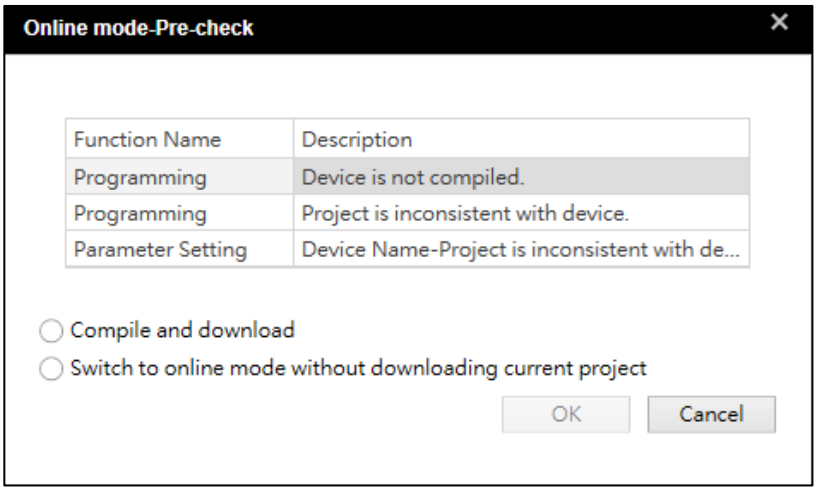
- The device is connected.



Follow these steps to use the online mode.

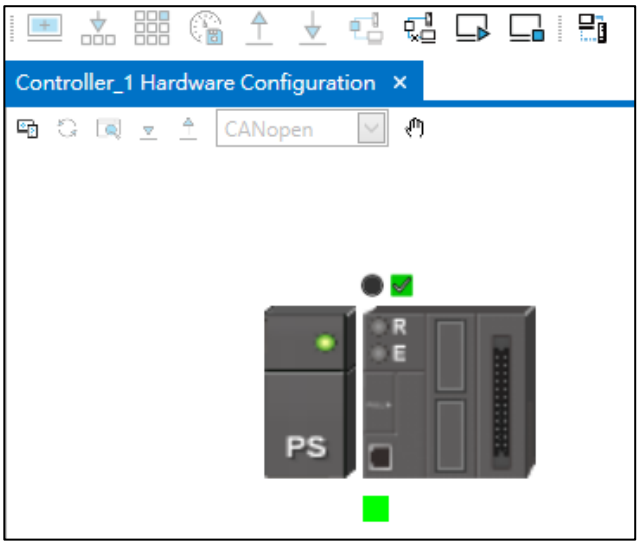
1. Click  on the toolbar.

2. If **device is not compiled**, **Project is inconsistent with device** and **Device Name-Project is inconsistent with device** condition occurred, the **Online Mode-Pre-check** dialog appears. You can choose to **Compile and download**, or **Switch to online mode without downloading current project**.



3. Click **OK** to enter the Online mode. The **Online** icon on the toolbar turns gray accordingly.

   You can monitor device in real time in the **Hardware Configuration** page.

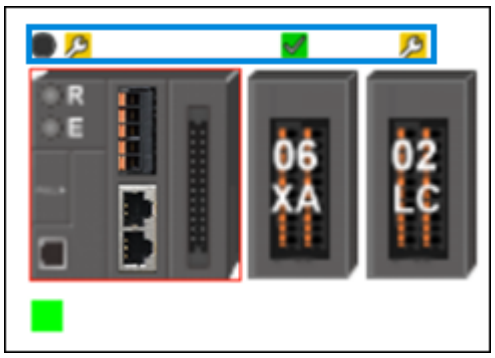


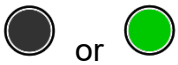
Note: When AS00SCM-A and its modules cannot be properly connected in the **Online** mode through the simulator, it needs to be connected to a physical device to connect.

6.3.1 Online Mode Status Light

The hardware configuration online mode provides status light to display device status information.

The following table lists the lights with their description.



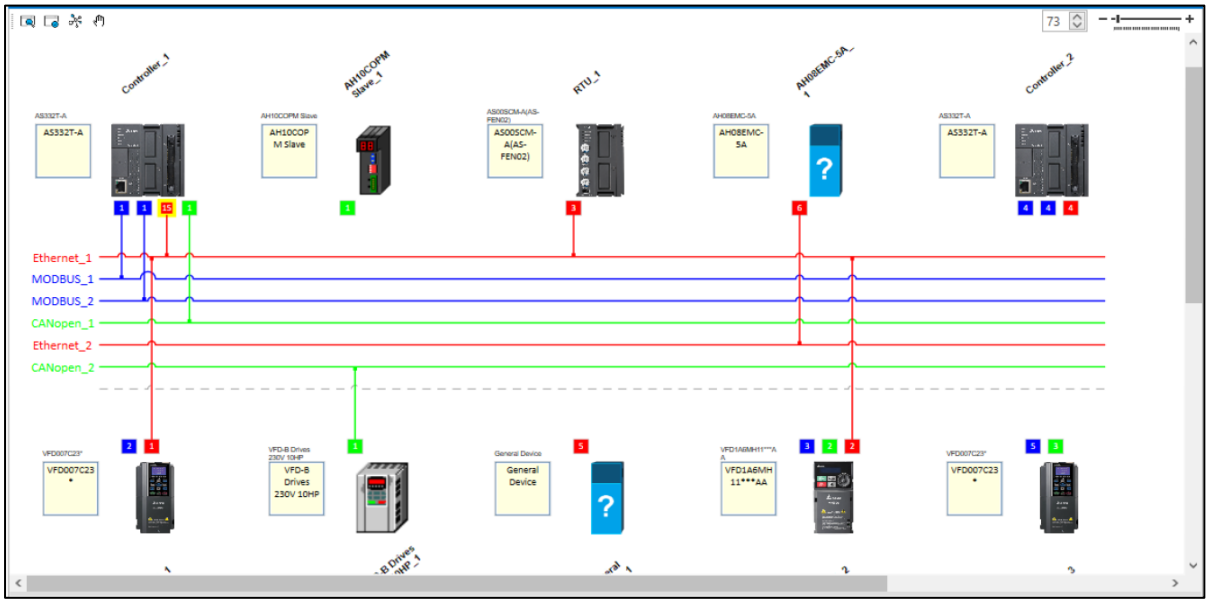
Light	Description
	Light remains lit and green while the device running. On the contrary, not extinguished, but light displays black color when the device is stopped.
	Light turns on when error occurs on the device. On the contrary, light extinguished. • There are devices in the project, but the actual connection devices do not exist. • There is no device in the project, but the actual connection exists in this device. Device in Project turns translucent gray. 
	• Green light remains lit when no error occurred. • Red wrench light remains lit when error occurred. • Yellow wrench light remains lit when device have status warning.

Chapter 7: Setting Network (Network Configuration)

7.1 Network View

Network View is an industrial network connected to the full range of Delta Industrial Automation products, which can be used according to the planned network view.

Network View is the graphical representation of connections between devices that allows you to plan networks.



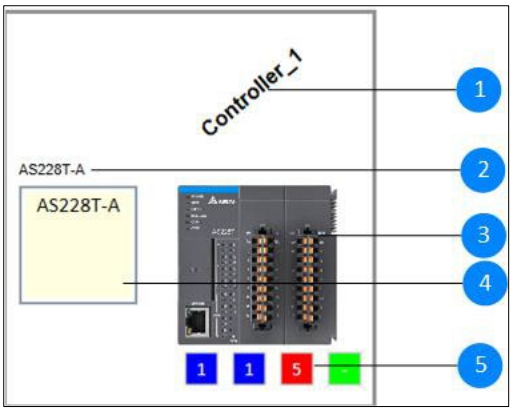
7.1.1 Network View Description

Network View displays two types of information.

- Device information
- Network information

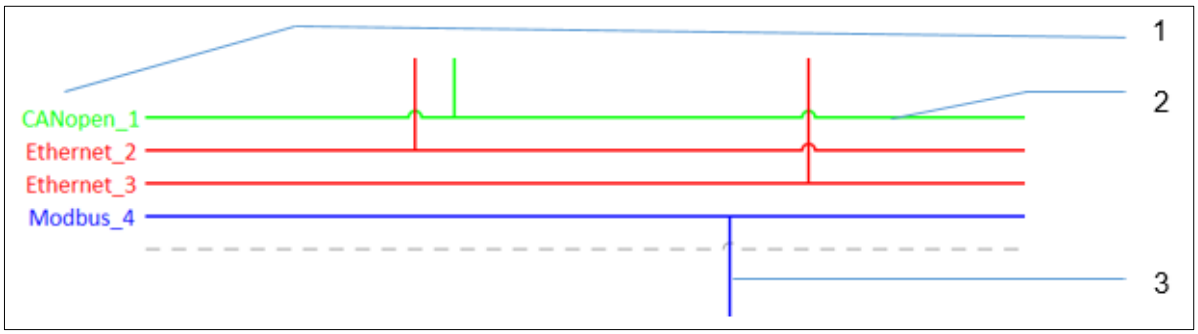
Device information includes:

1. Device name
2. Part number
3. Device image
4. A note to add text
5. Communication ports on the device with protocol support



Network information includes:

1. Name of the network
2. Network lines with colors
3. Connections to the devices



When the network connection is made between the controllers, ports, and field devices, the network lines appear in the color described in the following table.

Fieldbus or Network	Network Line Color
EtherNet/IP	Red
Modbus TCP	Red
Modbus	Blue
CANopen	Green

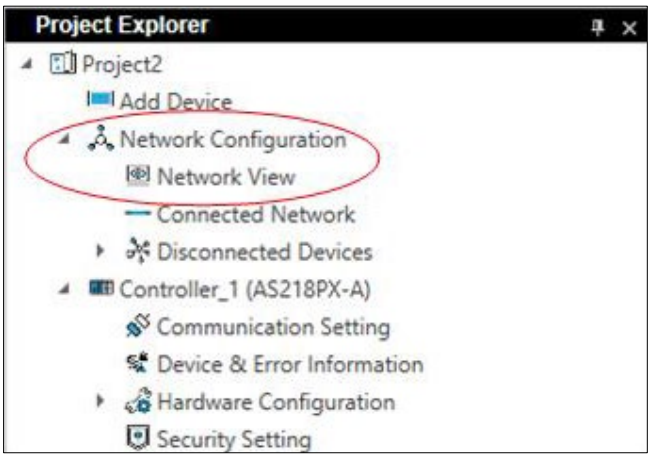
7.1.2 Creating Network View in Advanced Designer

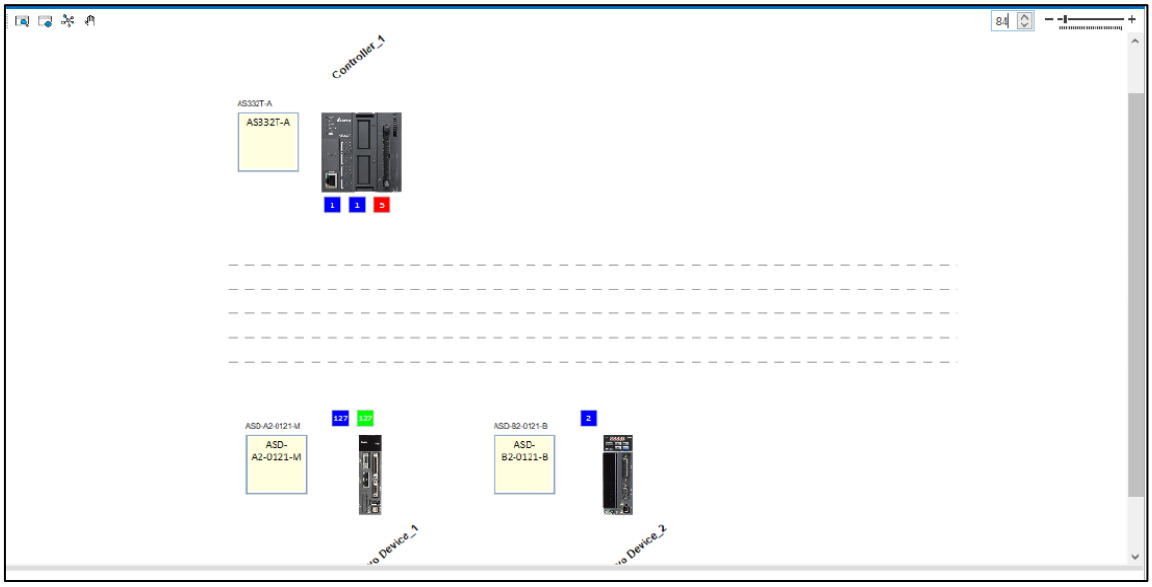
Follow these steps to create a Network View in Advanced Designer.

To create Network View

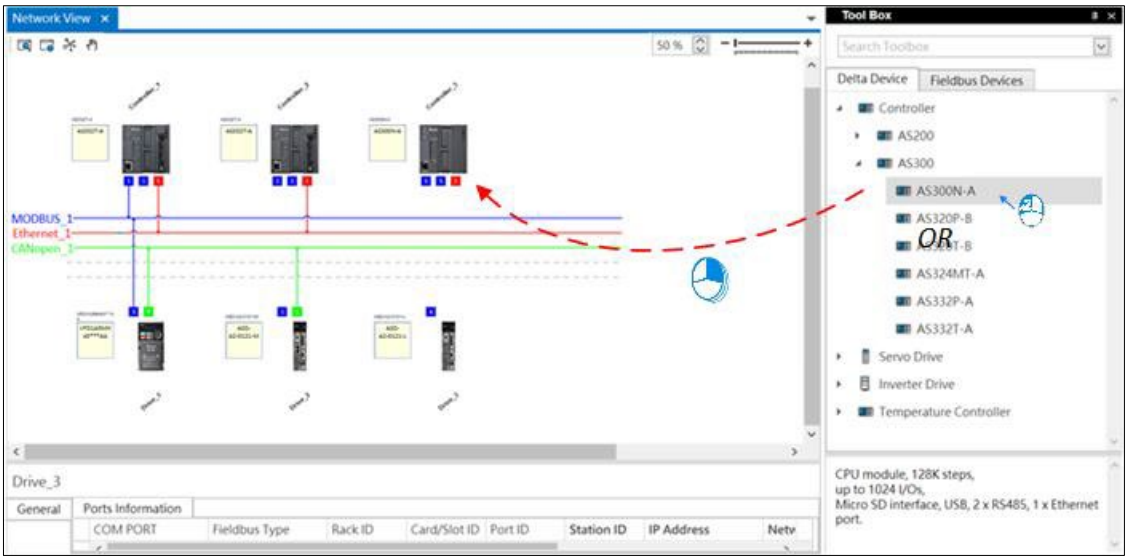
1. Expand **Network Configuration** in **Project Explorer**.

Double-click **Network View** to open **Network View** dialog. By default, dotted lines appear to create connections between the devices / networks.



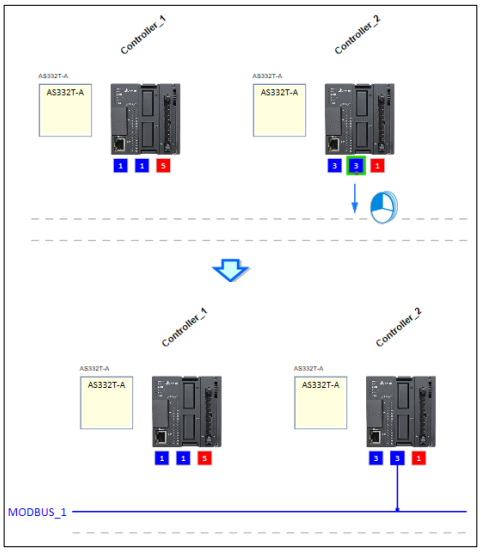


2. Drag and drop or double-click from **Toolbox** area to add devices in the **Network View**. The toolbox includes two tabs: **Delta Devices** and **Fieldbus Devices**. The **Delta Devices** tab lists Delta products. Select Delta products here to add them to the project tree to set up and use complete functions. The **Fieldbus Devices** tab is a list of third-party products stored in the software. After you import the electronic description file of the third-party product (such as EDS, etc.), it appears in this list, and you can choose to join the network view as a slave configuration. For how to import electronic description files of third-party products, refer to Device Repository.



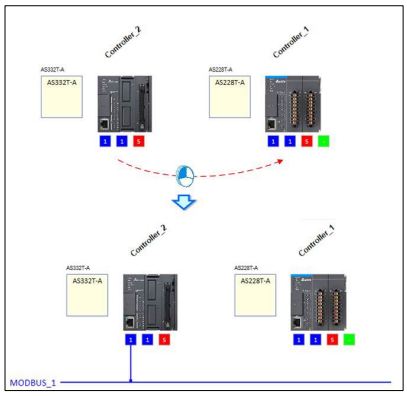
Note: You can also use **Add Device** in **Project Explorer**.

3. Click **Communication Port** and connect it to a dotted line with drag and drop. Dotted line changes to a solid line with same color as that of the device communication port (protocol) selected as shown in the following figure.



Note:

- You can also connect directly between two devices by clicking and dragging from one device port to another.

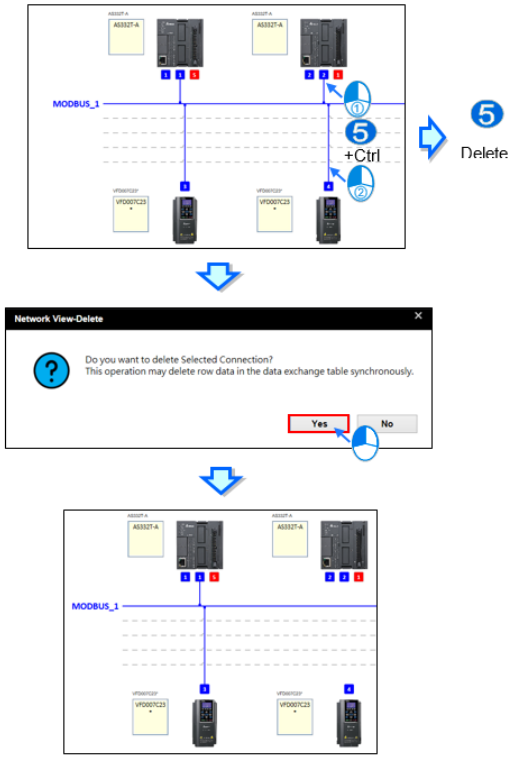


- If you double-click the device in the **Network View**, it takes to **Hardware Configuration** dialog.

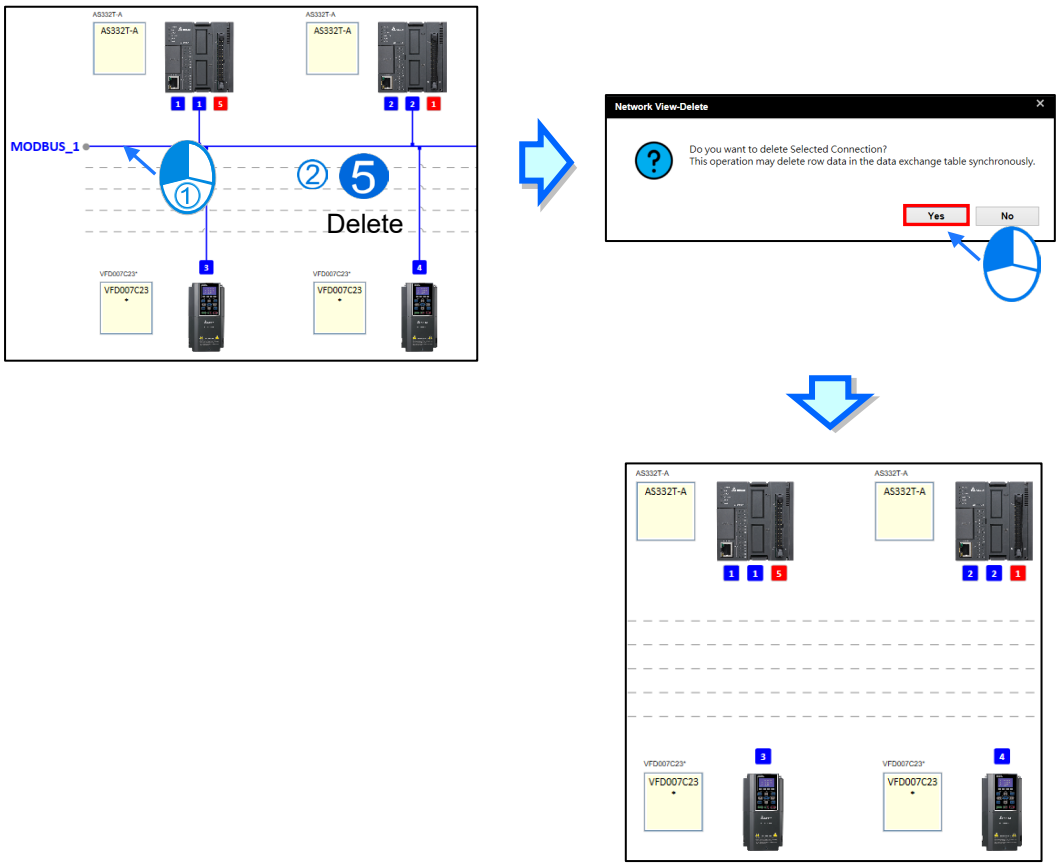
If the network is already connected, follow these steps to delete the **Selected Connection**.

To delete connection

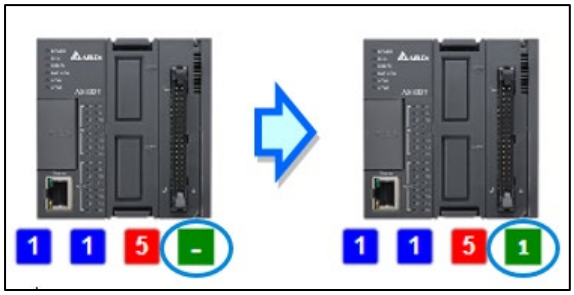
1. In the **Network View**, select the vertical line to be deleted.
2. To delete multiple vertical lines, press **Ctrl** key on the keyboard and select multiple lines to be deleted.
3. Press **Delete** key on the keyboard, and then press **Yes** in the dialog box to confirm the deletion of **Selected Connection** is completed.



To delete all the connections connected to the same network, select the horizontal line of the network and directly press **Delete** key on the keyboard, and then press **Yes** in the dialog box.



Note: When the icon status of the CANopen port is -, it means the current mode is Delta special mode, and the standard CANopen is not connectable. If you want to connect the standard CANopen, go to **Parameter Setting** and set the work mode to "CANopen DS301", and the port icon status becomes connectable. For the operation instructions on **Parameter Setting**, refer to Parameter Setting.



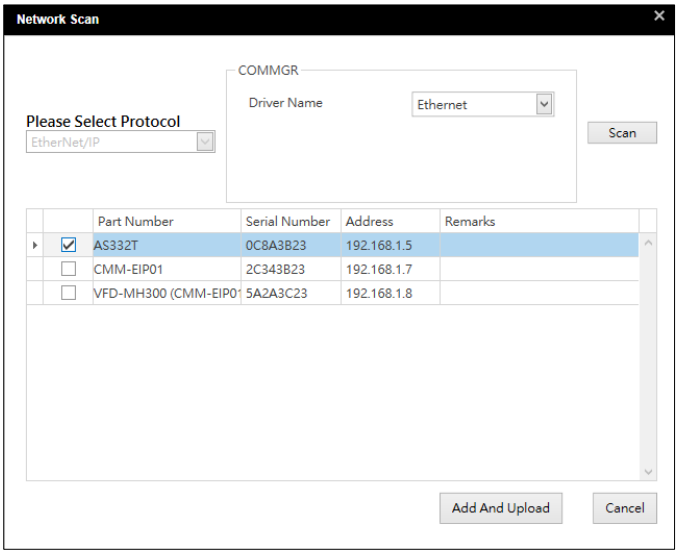
7.1.3 Network View Toolbar

The following table describes functions in **Network View** toolbar.

Toolbar Icon	Description
	Network scan by PC.
	Check network for the related parameter setting.
	Prominently remind you of the current device connection.
	Drag mode.

7.1.3.1 Network Scan

Click **Network Scan**  to scan connected devices.

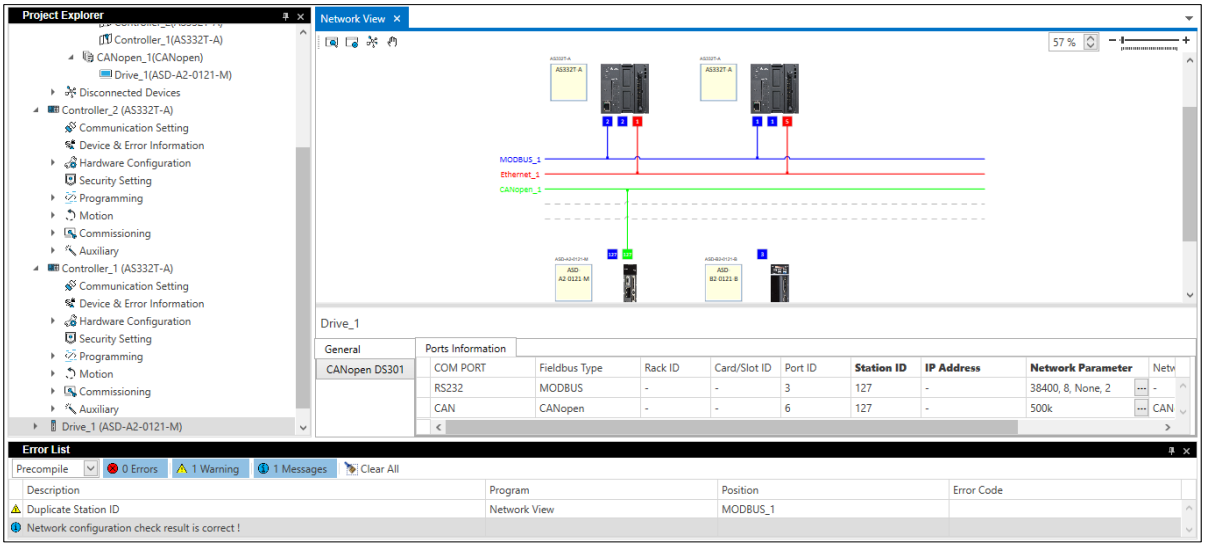


Select devices from the list and add it to the project.

Note: EtherNet/IP is the only protocol supported in **Network Scan** for the present version.

7.1.3.2 Network Check

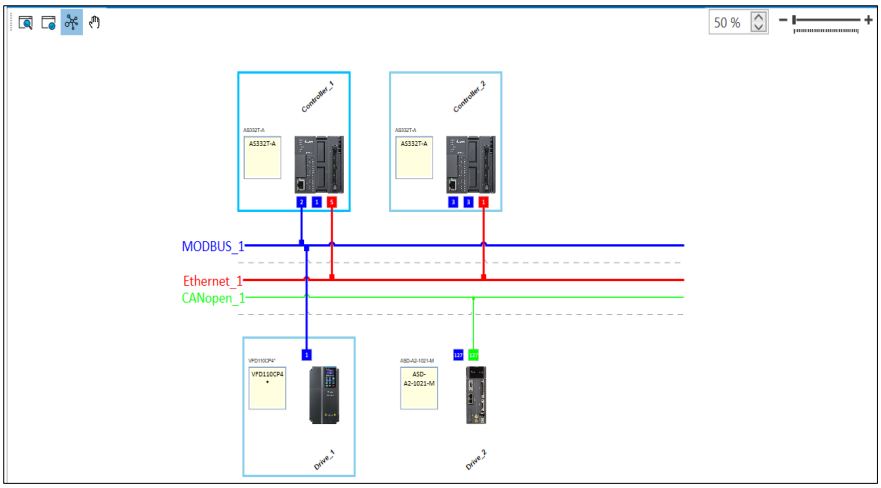
Click **Network Check**  to check network setting correctness. If there is any error, the error symbol and error message appear. The error due to parameter mismatch of a network is shown in the following figure.



7.1.3.3 Highlight Connections of User Selected Device

Click  to activate the function. After that, when you select a device, all the device connections and the connected devices will be highlighted, which is convenient for you to view the network view.

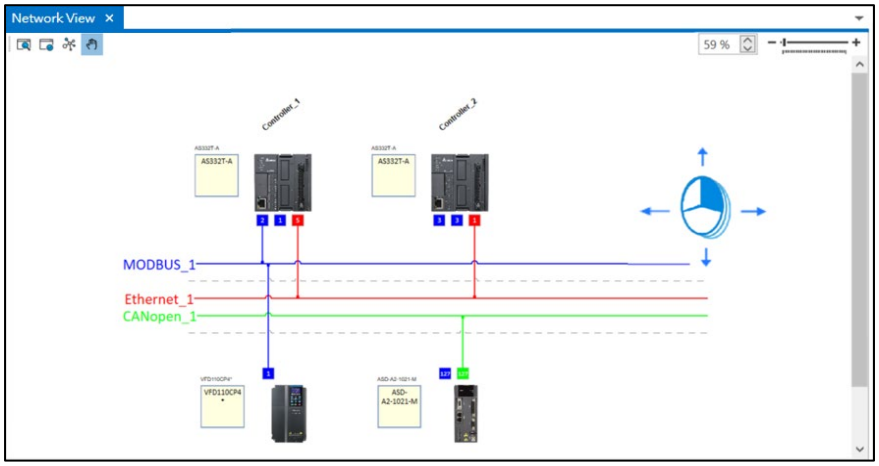
- Device: In bold blue box
- Bold horizontal / vertical network connection



Note: Click  again to stop the function.

7.1.3.4 Drag Mode

Click **Drag Mode**  to enable the function. When the cursor is an arrow, besides using scroll wheel, use scrollbar to move the screen. When drag mode is active, directly click and drag in the blank space of the display area to move the screen as shown in the following figure.



Note: Click  again to stop the function.

7.1.4 Network View Context Menu

Right-click the device or a blank space in the **Network View** pane to use shortcut functions.

The following table lists the functions in the context menu of the **Network View** pane with their description.

Function	Description
Add Device	Click to select the device to add from the **Tool Box** pane. Double-click the device or drag the device to the **Network View** from the **Tool Box**.
Quick Connection	Click to create a network connection quickly.
Select All	Click to select all objects (including the device and the communication lines) in the **Network View**.

The following table lists the functions in the device context menu with their description.

Function	Description
Online	Click to go online between the computer and the device.
Offline	Click to go offline between the computer and the device.
Upload	Click to upload the device communication.
Download	Click to download the device communication.
Network Scan	Click to scan the device network.
Configuration	Click to open the **Hardware Configuration** pane.
Copy	Click to copy the device and paste it in the **Network View**.
Delete	Click to delete the device.
Update Device	Click to open the **Update Device** dialog.

When performing a network scan through the device context menu, the software compares the scanned device with the configured device. If both devices are the same devices, the topology connection will be automatically created.

Network scan result rules

- When the scanned device has the same serial number and IP / station IDs as the configured device, they are merged into one device.
- When the scanned device has the same serial number as the configured device, but the IP / station IDs are different, a message dialog displays asking whether to change the IP / station ID.
- After the scanning, the newly added or previously configured devices create the topology connection based on the scan result.

7.1.5 Network Parameter Setting

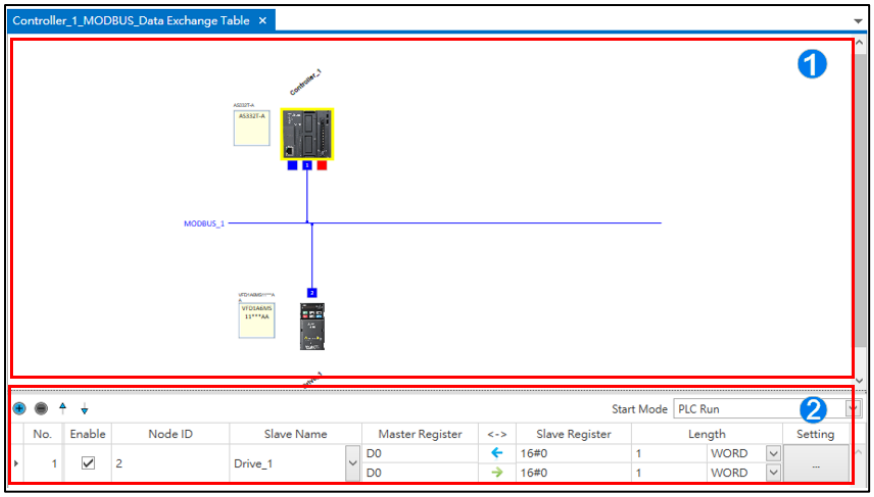
You can directly modify network parameters and IP addresses in the table at bottom of **Network View**. The modified values will be directly synchronized with the corresponding parameter values of parameter setting in **Hardware Configuration**.

7.2 Data Exchange Table

Data Exchange Table is a tabular tool for you to set up the data to be read / written from master to slave device. With the help of data exchange table, you need not to know the function code or detail protocol definition to complete the setting process.

Data Exchange Table can be separated into two sections.

1. Data Exchange Table Network View
2. Data Exchange Table



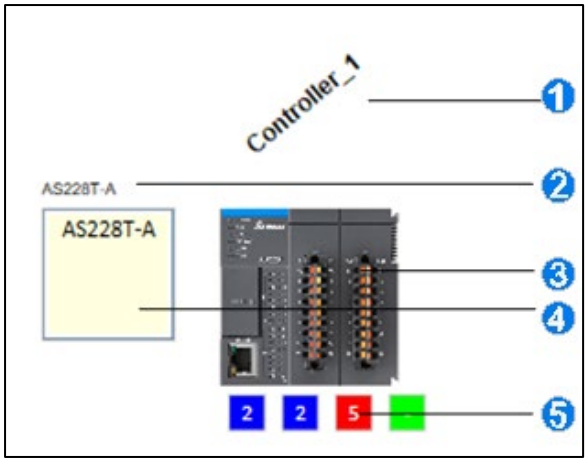
Data Exchange Table Network View

Data Exchange Table Network View displays two types of information.

- Device information
- Network information

Device information includes:

1. Device name
2. Part number
3. Device image
4. A note to add text
5. Communication ports on the device with protocol support

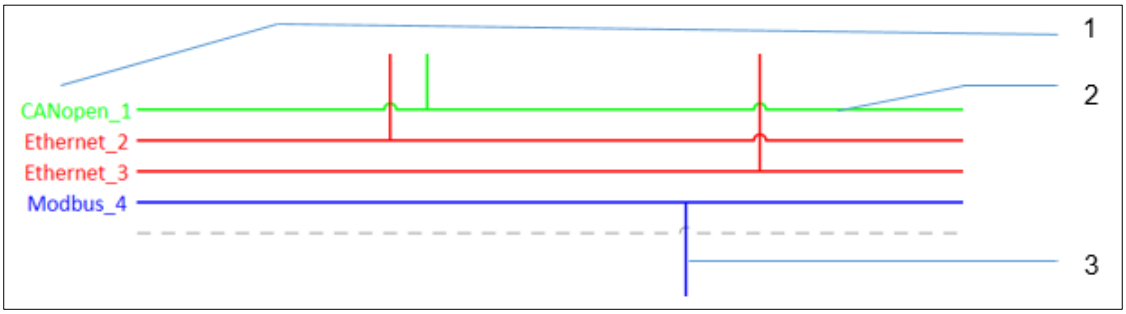


Industrial Network and Fieldbus information includes:

1. Industrial network name

2. Industrial network and fieldbus line

3. The vertical line to connect between devices



When the network connection is made between the controllers, ports, and field devices, the network lines appear in the color described in the following table.

Fieldbus or Network	Industrial Network Line Color
EtherNet/IP	**Red**
Modbus TCP	**Red**
Modbus	**Blue**
CANopen	**Green**

Note: Data Exchange Table Network View only shows the device exists in **Project Explorer**. If you add a device that does not exist in **Project Explorer** in the edit area of the data exchange table, the device information will not be displayed in the Data Exchange Table Network View area.

Data Exchange Table Area

The content of the data exchange table fields varies with different communication protocols. For details, refer to EtherNet/IP Data Exchange Table, CANopen Data Exchange Table, and MODBUS and MODBUS TCP Data Exchange Table.

7.2.1 Exchanging Data through Data Exchange Table

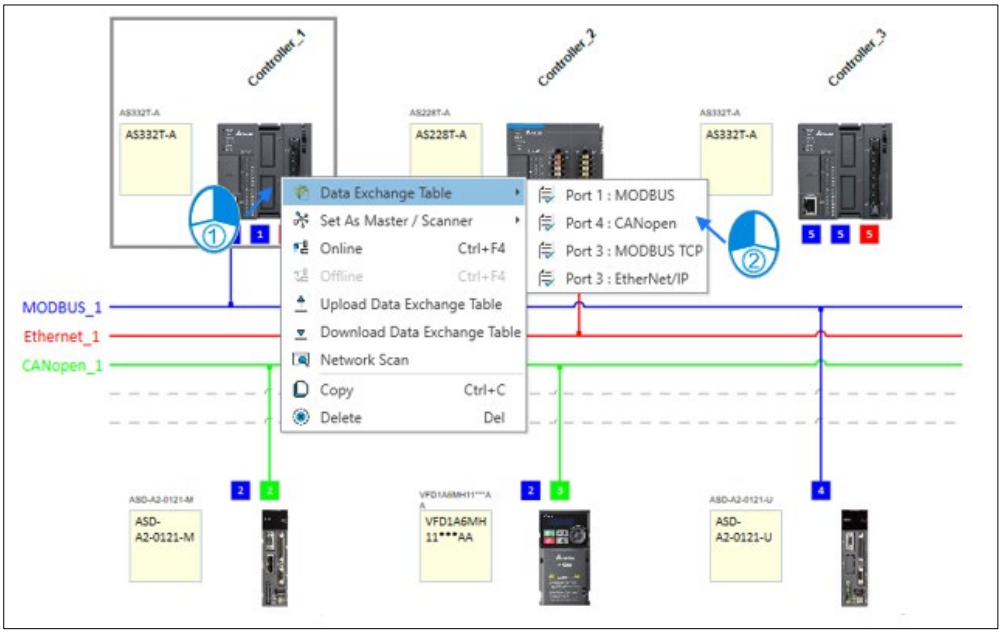
After completing the network connections of each device, follow these steps to exchange data in Advanced View.

To exchange data

Method 1

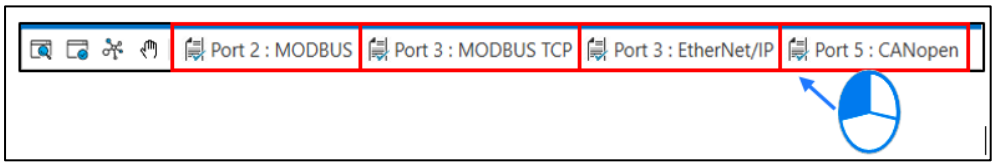
Right-click to open the context menu of a master device, and click required protocol

in **Data Exchange Table** as shown in the following figure.

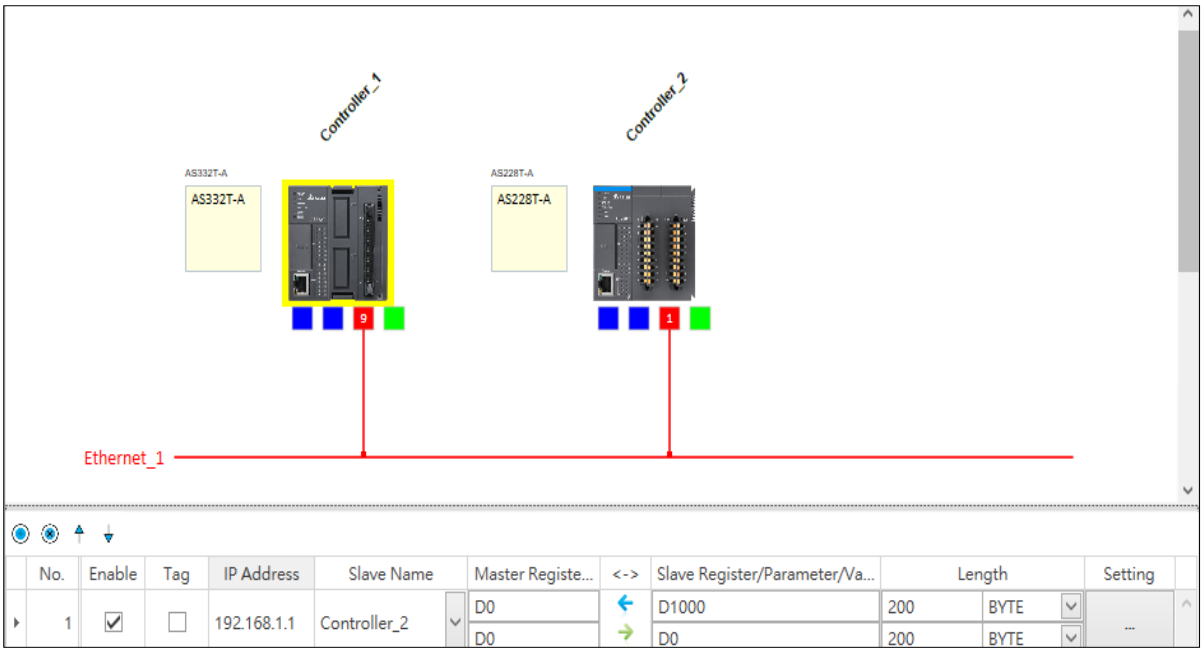


Method 2

Press the protocol button for creating a **Data Exchange Table** in the toolbar.



A **Data Exchange Table** tab of the selected protocol opens, and the connected devices of this protocol appears on it. One row of default data exchange table is also created at the bottom of this tab as shown in the following figure.

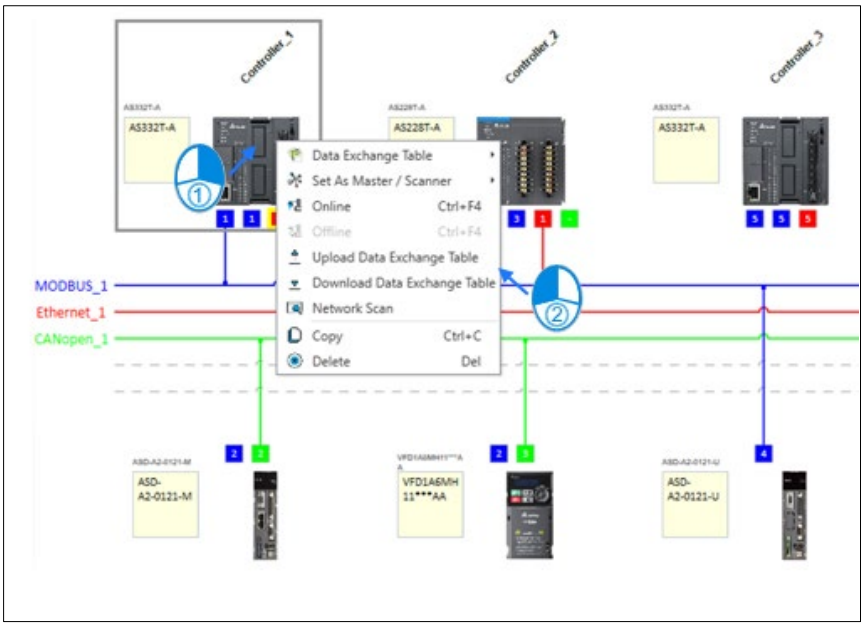


1. Add or delete a row of data exchange table and then move up/down by clicking the respective buttons in the data exchange table toolbar as shown in the following table.

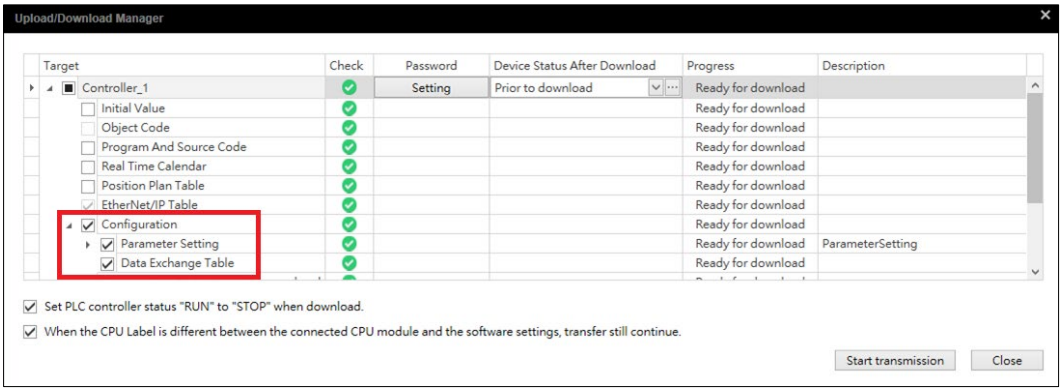
Toolbar Icon	Function
	Add
	Delete
	Up
	Down

2. Set up the parameters in each data exchange table. Refer to the Setting Up Data Exchange Table.

3. To download / upload the data exchange table to / from the master device, right-click to open context menu, and select **Download Data Exchange Table/Upload Data Exchange Table** to open **Upload/Download Manager** dialog.



The Configuration node including **Data Exchange Table** is selected to proceed with upload / download as shown in the following figure.



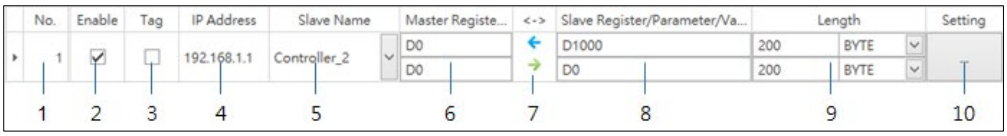
Note: You can also execute **Project Download** to open **Upload/Download Manager** dialog for **Data Exchange Table** download.

7.2.2 Setting Up Data Exchange Table

The examples of setting up the parameters in Data Exchange Table are given below.

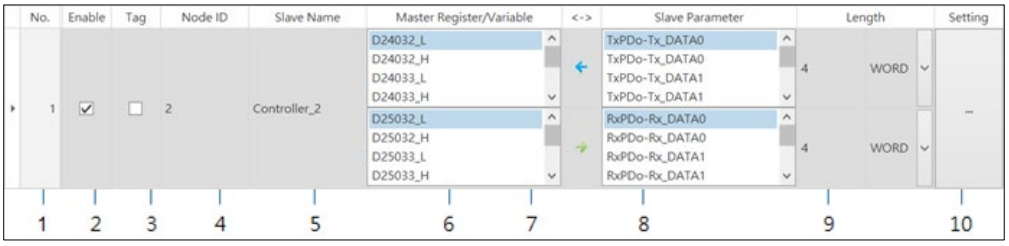
Example 1

Take Ethernet/IP data exchange table as an example; the value in D1000 of Controller_2 (192.168.1.1) is read into D0 of master, and the value in D0 of master is written into D0 of Controller_2.



Example 2

In CANopen protocol, the data exchange table means the PDO mapping between master and slave devices.



The following table describes the functions in Data Exchange Table.

Function	Description
No.	The row number of the data exchange table.
Enable	Enable the row of data exchange table. It can be enabled or disabled by the **SM** register while controllers run.
Tag	Enable to map the variables in the specific global variable table. Only CANopen and EtherNet/IP support this function.
IP Address/Node ID	Target slave IP address.
Slave Name	Target slave device name.
Master Register/Variable	Target register or variable in master device.
<->	means read. means write.
Slave Register/Parameter/Variable	Target register / parameter / variables in slave device.
Length	The data length to be transferred between master and slave.
Setting	Detail setting dialog. To set up parameters for this data exchange table of the protocol. For detailed definition of each parameter, refer to related technical documents.

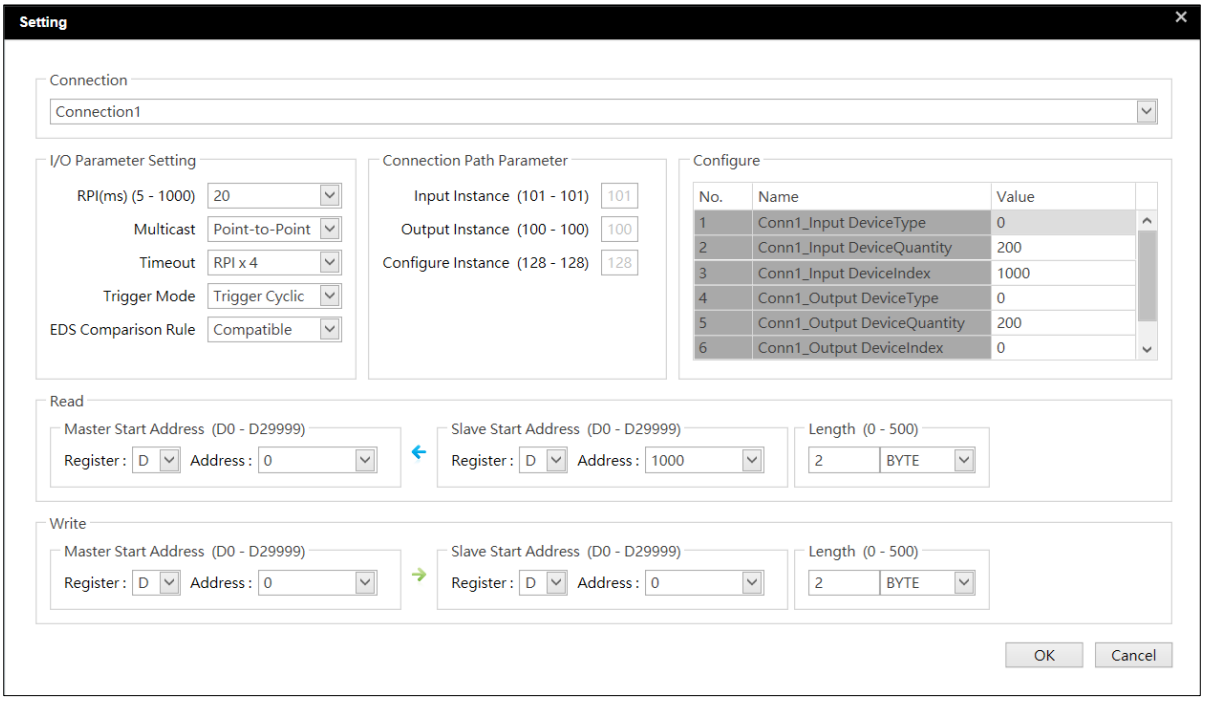
7.2.3 EtherNet/IP Data Exchange Table

The tag is enabled for EtherNet/IP only when both master and slave devices had already declared produced or consumed variable in **EIP_produced_Tag_Table** or **EIP_Consumed_Tag_Table**.

If the slave is third party device or the Delta controller which is not in the project, type variable name in **Slave Register/Parameter/Variable** directly.

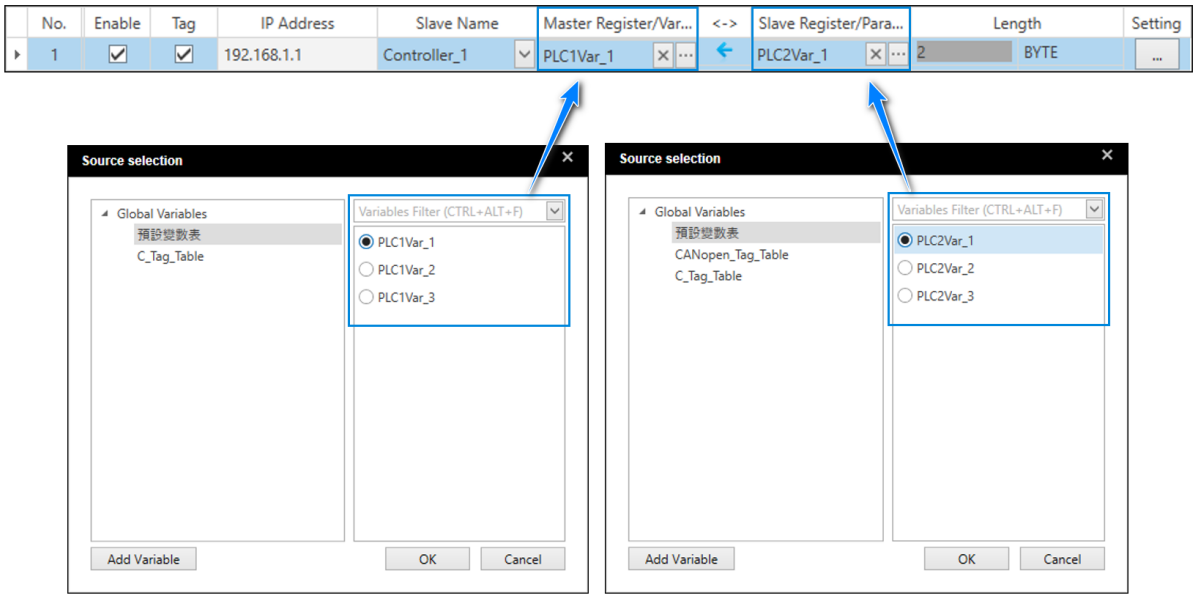


Click **Setting** to open the **Setting** dialog of **EtherNet/IP Data Exchange Table**.



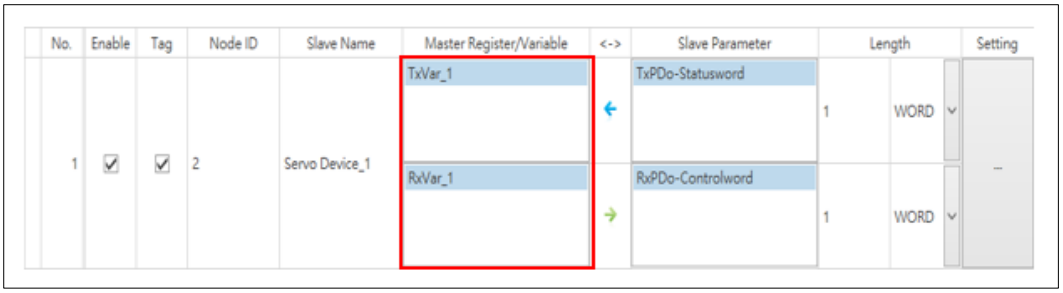
7.2.3.1 Variables

After selecting the **Variables** checkbox, you can select the corresponding variables for the master and slave registers . Once selected, the variables will be automatically added to the **EIP Produced Symbol Configuration** and **EIP Consumed Symbol Configuration** and will have EIP properties. For details, refer to Symbol Configuration.

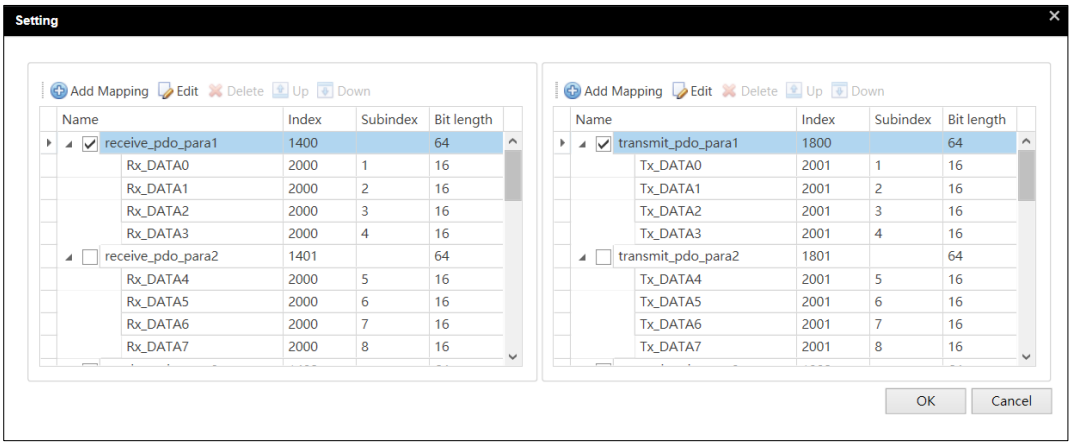


7.2.4 CANopen Data Exchange Table

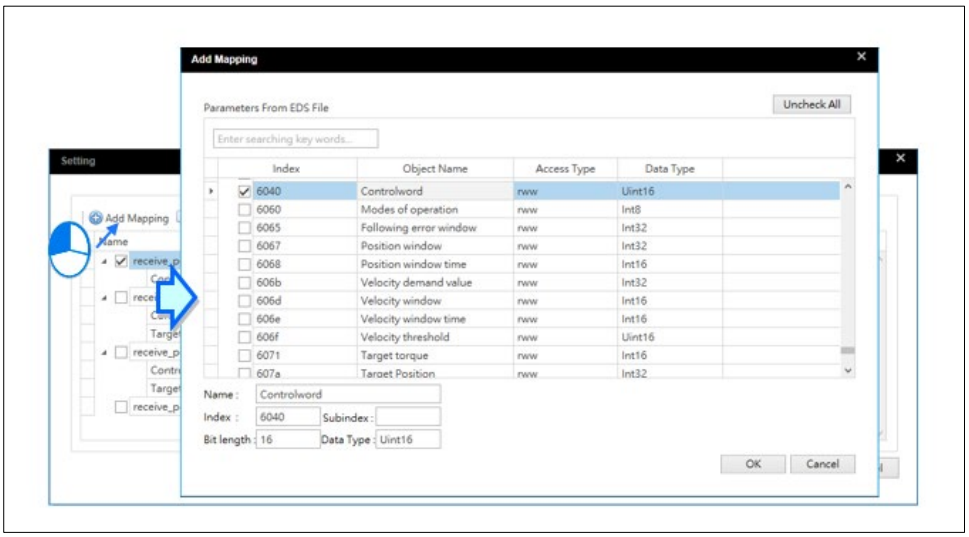
You cannot declare variable in **CANopen_Tag_Table** directly. When selecting the **Tag** checkbox in CANopen data exchange table, the default variables are created in **CANopen_Tag_Table** automatically, and change the variable name in the field.



Click **Setting** to open the **Setting** dialog of **CANopen Data Exchange Table**.



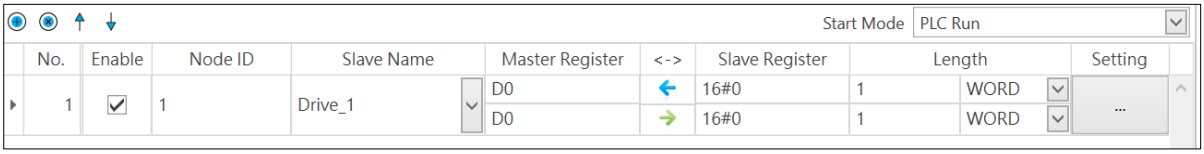
Click **Add Mapping**, then add or delete object in each PDO packet in the pop-up dialog.



7.2.5 Modbus / Modbus TCP Data Exchange Table

Modbus and Modbus/TCP do not support the function of designated variable exchange.

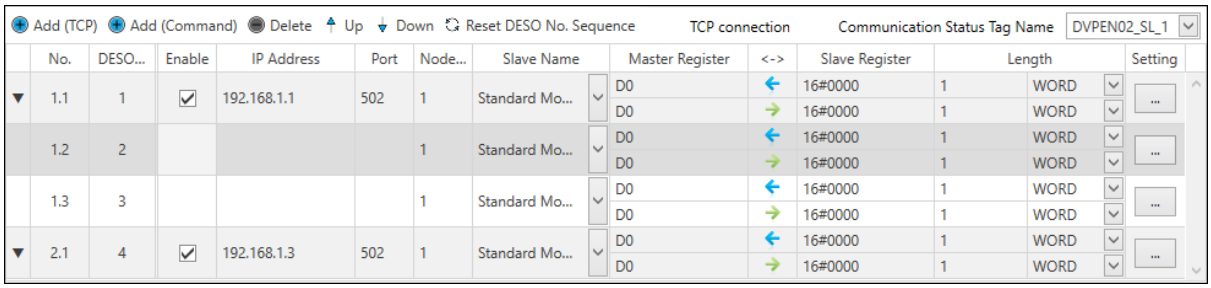
The following table lists the exchange table toolbar functions for the general model with their description.



Function	Description
	Add an exchange table.
	Delete the selected exchange table.
	Move up the exchange table.
	Move down the exchange table.
Start Mode	Select Instruction Control, PLC Run, or Always Enable the exchange table

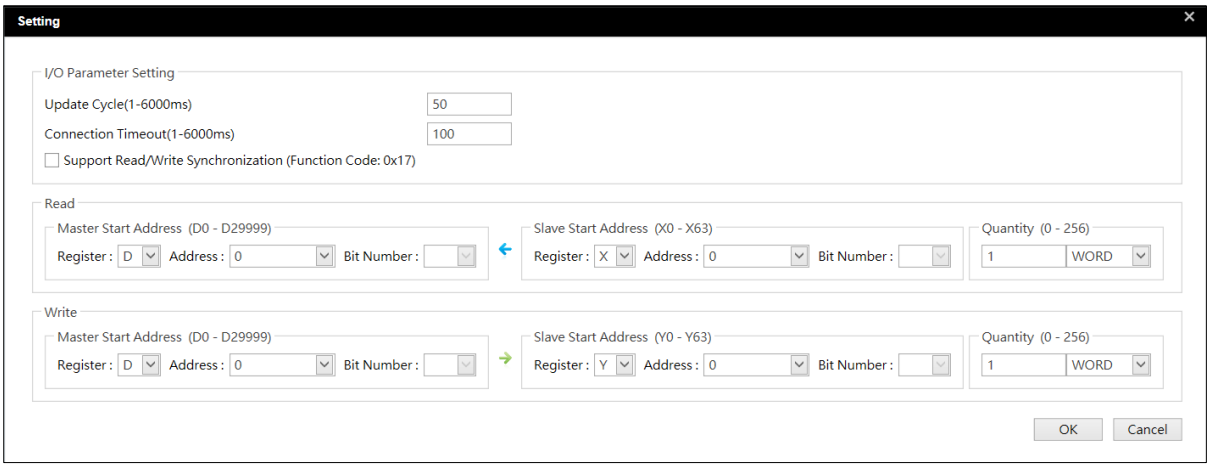
For the MODBUS TCP exchange table of DVPEN02-SL, you can use a limited number of connections to send different communications in a polling manner through the TCP connection and command connection functions. For detailed function, refer to the DESO instruction.

The following table lists the MODBUS TCP exchange table toolbar functions for DVPEN02-SL with their description.

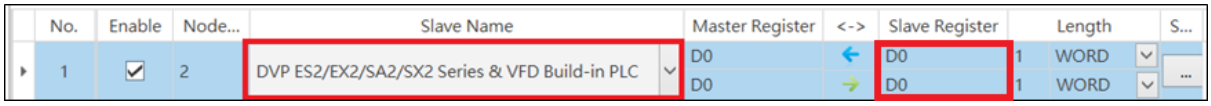


Function	Description
Add (TCP)	Add a TCP connection exchange table.
Add (Command)	Add a command connection exchange table.
Delete	Delete the selected TCP / command exchange table.
Up	Move up the exchange table.
Down	Move down the exchange table.
Reset DESO No. Sequence	Sort the DESO numbers in ascending or descending order.
TCP connection usage ModbusTCP command connection usage	Count the number of TCP / command connections in the current data exchange table.
Communication Status Tag Name	Variables and data types automatically generated by the system and will be generated in the global variable table by default.

Click **Setting** to open the **Setting** dialog of Modbus and Modbus/TCP data exchange table.



If the connection target slave station is the built-in PLC of Delta inverter, select **DVP ES2/EX2/SA2/SX2 Series & VFD Build-in PLC** in **Slave Name** drop-down menu, and then click **Slave Register** field on the right. Enter the address of the build-in PLC to save users time in finding the corresponding Modbus address.



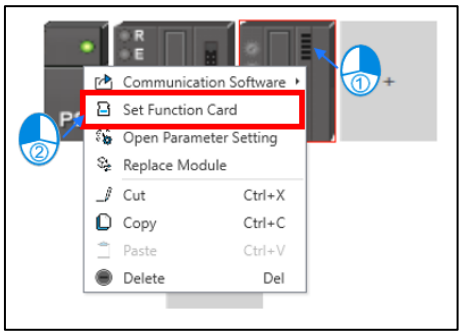
7.2.6 SCM Module Data Exchange Table Setting

After adding Network Interface Function Card, SCM module can activate Data Exchange Table function when linked to the device.

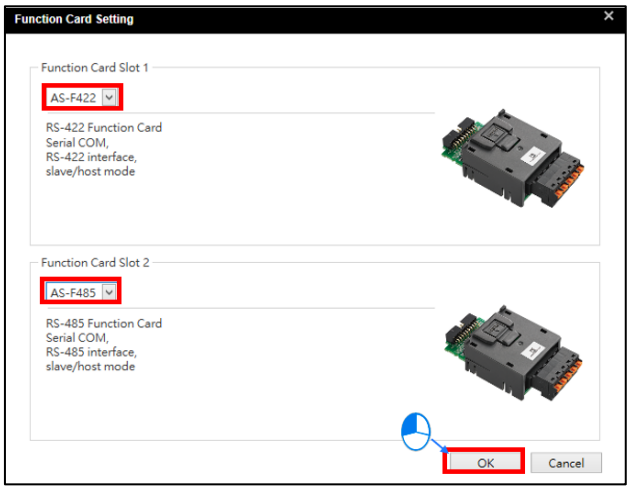
Follow these steps to access SCM Module Data Exchange Table.

To access SCM Module Data Exchange Table

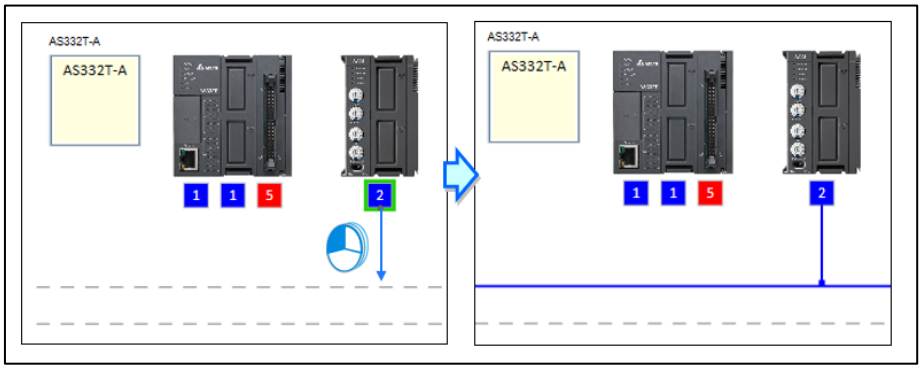
1. Add SCM module to controller through **Hardware Configuration**. (Refer to Adding Modules.)
2. Right-click SCM module and select **Set Function Card**.



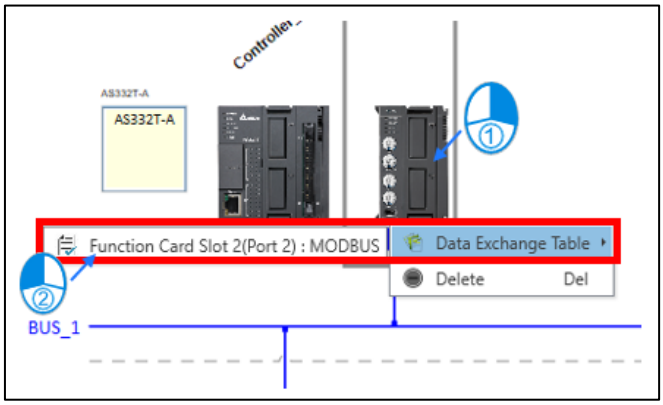
3. Choose the function card and click **OK**.

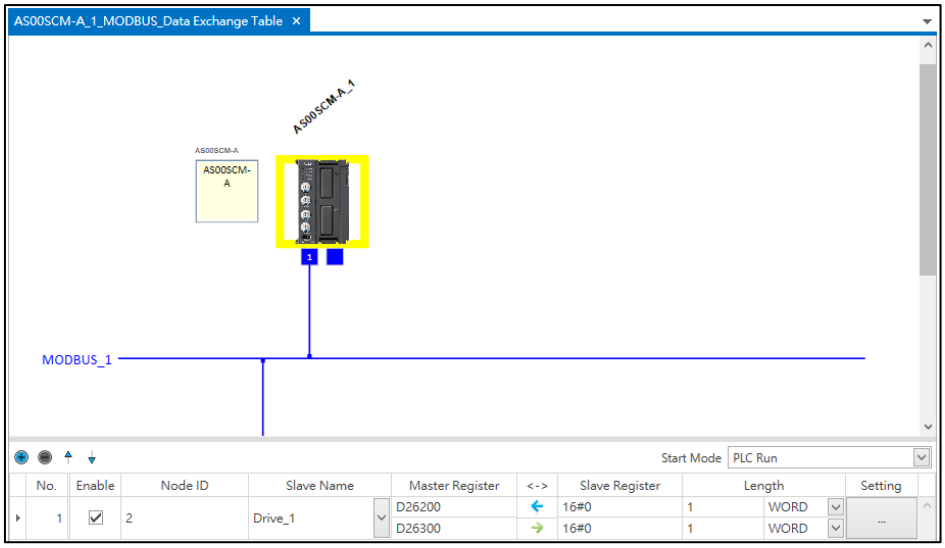


4. Click **Network Configuration** > **Network View**. Link device by dragging line from **Device Port** to horizontal industrial networks. (Refer to Creating Network View in Advanced Designer for more information.)



5. Right-click SCM module and select **Data Exchange Table** to access Data Exchange Table for further configurations.





SCM module supports Modbus protocol. For Data Exchange Table setting, refer to Modbus/Modbus TCP Data Exchange Table.

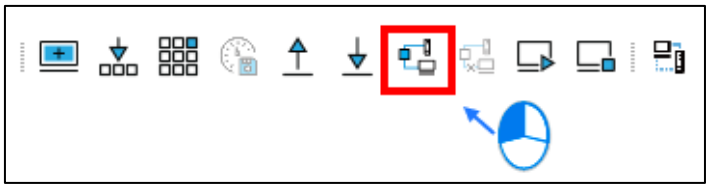
7.3 Network View and Data Exchange Table Online Mode

Network View and Data Exchange Table provide Online Mode, allowing you to monitor in Real Time.

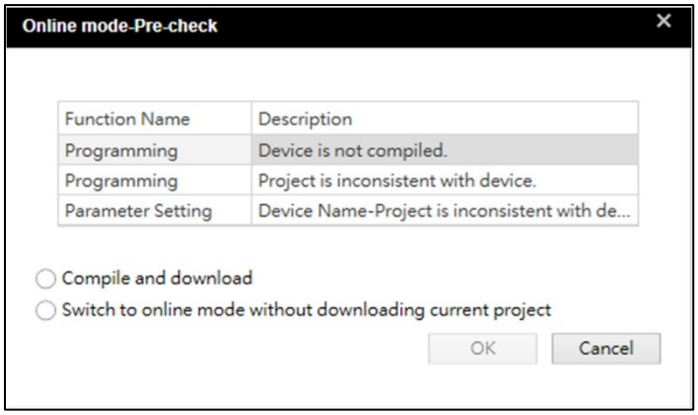
Follow these steps to enable online mode.

To enable online mode

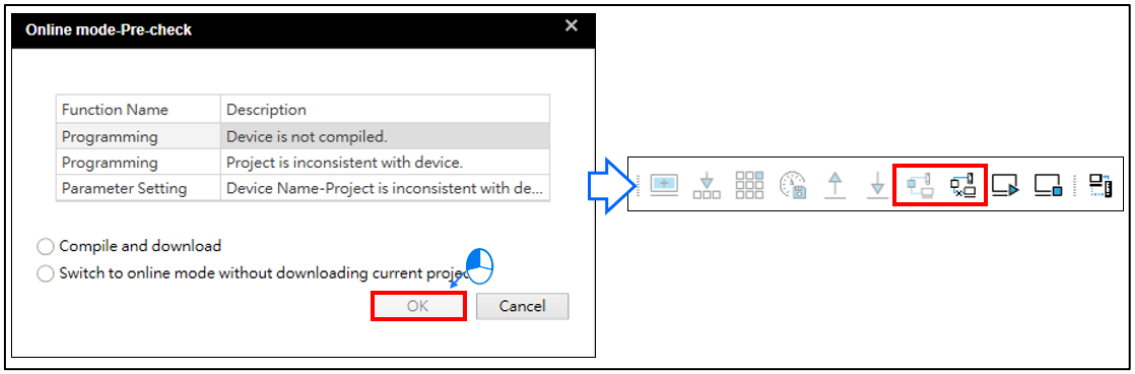
1. Confirm the device is connected successfully.
2. Click **Online**  on toolbar.



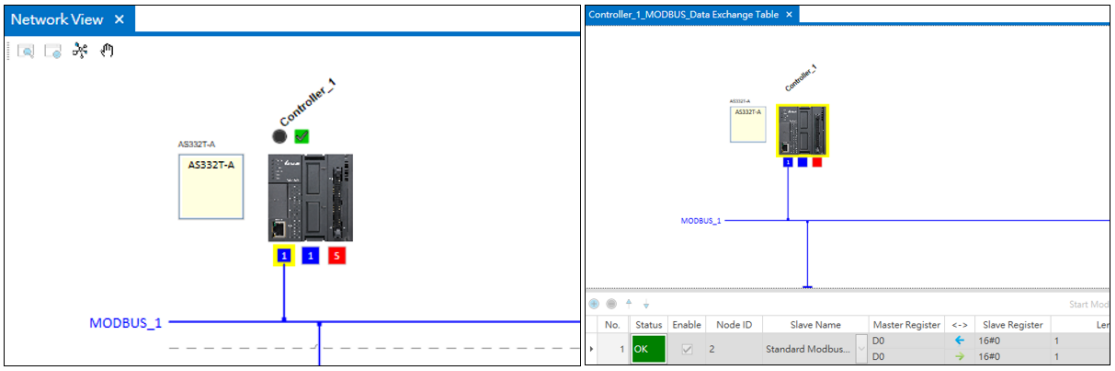
3. If "device is not compiled", "Project is inconsistent with device" and "Device Name-Project is inconsistent with device" condition occur, **Online Mode-Pre-check** dialog appears. You may choose to compile and download, or Switch to online mode without downloading current project.



4. After clicking **OK**, device enters Online Mode successfully. The **Online** icon on the toolbar turns gray accordingly.

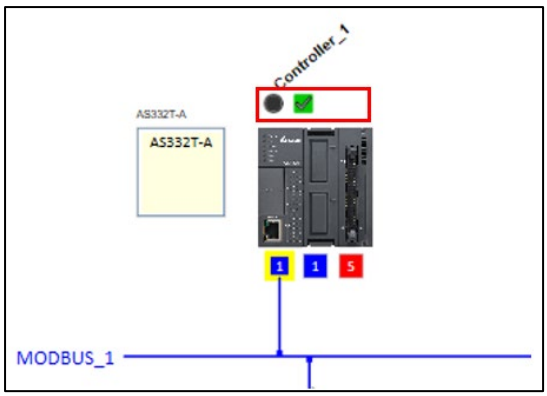


Enter **Network View** and **Data Exchange Table** for real time device monitoring.

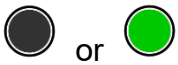


7.3.1 Network View Online Mode

Network View Online Mode provides device status information through the lights around the device icon.



The following table lists the description of lights.

Light	Description
	Green light remains lit while the device is running. On the contrary, black when stopped.
	Green light remains lit when no error occurred. Red wrench light remains lit when error occurred. Yellow wrench light remains lit when devices have status warning.

7.3.2 Data Exchange Table Online Mode

The following table lists the description of the online mode status through the data exchange table.

Status Light	Description
	Light remains lit when Data Exchange Table operates properly.
	Light remains lit and shows error message when device has error. Message will be shown in Error Description column. **Note**: Refer to Data Exchange Table Online Mode Error Code Table.
	Light remains lit when Data Exchange Table in the device is not consistent with the Data Exchange Table in the Software. At the same time, notice with red character appears above the table. Current data exchange table in the project is inconsistent with that in device, do download it first.

7.4 EtherCAT Topology Configuration

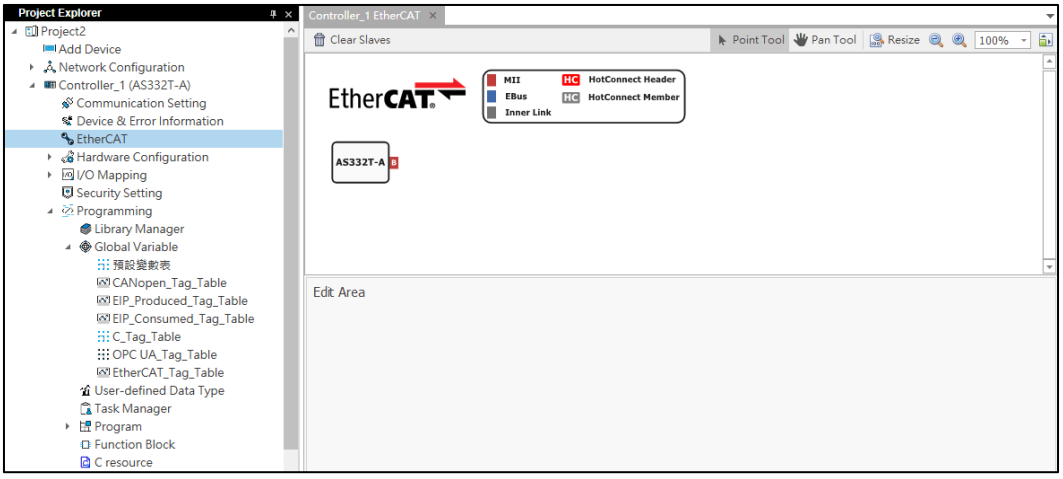
EtherCAT Topology Configuration is used to plan the topology connection relationship between the master and slave devices of EtherCAT and the right-side module configuration of the RTU module and generate the subsequence EtherCAT I/O mapping.

To open the EtherCAT Topology Configuration page

Do one of the following.

- Open after setting the function card in the **Hardware Configuration** dialog.

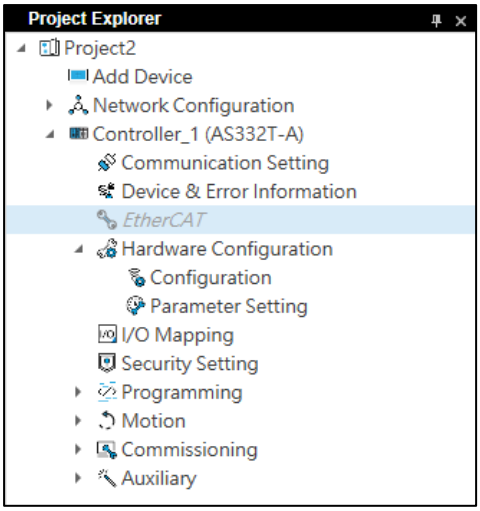
  3. In **Project Explorer**, add a controller (Take AS300 series device as an example).
  4. Double-click **Hardware Configuration** > **Configuration**.
  5. In the **Hardware Configuration** dialog, right-click the device and select **Set Function Card** in the context menu.
  6. In the **Function Card Slot 2** drop-down menu, select **AS-FECAT**.
  7. In **Project Explorer**, double-click **EtherCAT**.



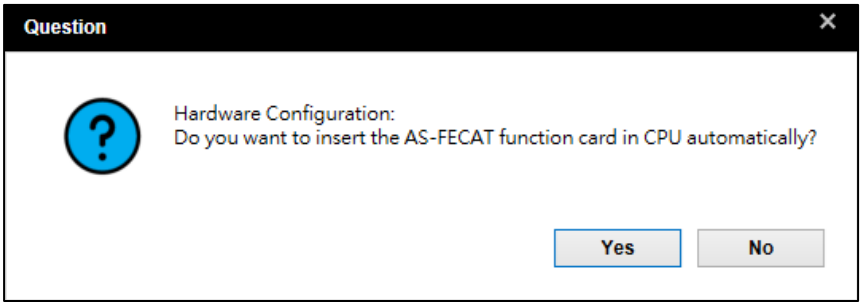
- Directly double-click **EtherCAT** to open the **EtherCAT Topology Configuration** page.

  1. In **Project Explorer**, double-click **EtherCAT.**

     Note: Before setting the function card, this function is not available.



2. Click **Yes** in the pop-up dialog box.



The following table lists the functions in the **EtherCAT Topology Configuration** page with their description.

Function / Icon	Description
Clear Slaves	Click to clear the settings of slaves.
Point Tool	Switch to this icon to select objects.
Pan Tool	Switch to this icon to press and drag the topology.
Resize	Click to resize the topology configuration screen.
	Click to zoom in / zoom out the topology configuration screen.
100%	Select the value to zoom in or zoom out.
	Click to hide / display the pan and zoom widget.

The following takes AS332T-A as an example. Follow these steps to configure the EtherCAT topology.

To configure EtherCAT topology

1. Right-click the AS332T-A controller, and select **Insert Device** in the context menu.
2. In the **Insert EtherCAT device after 'AS332T-A'** dialog, select the desired slave device to insert and click **Insert**.
3. After the device is inserted, click **Close**.

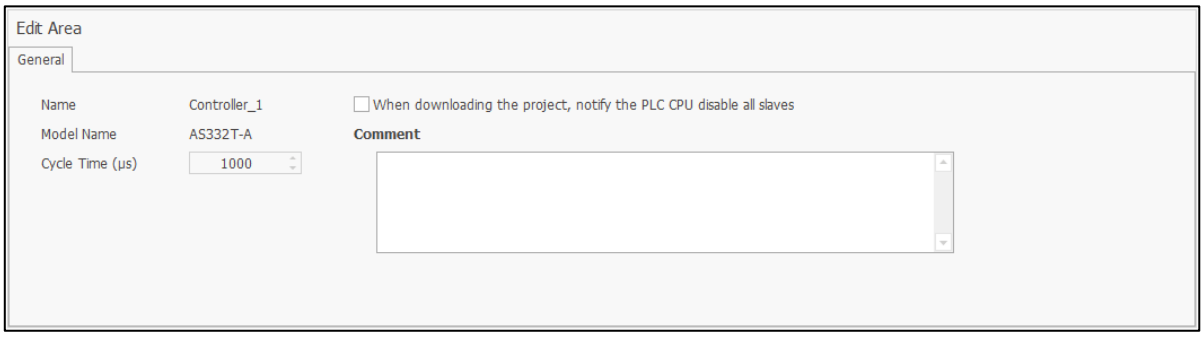
Note:

- Delta provides special instructions for AS / DVP series EtherCAT masters with ASD-A2E / ASD-A3E / ASD-B3E / ASD-E3E, C2000 / C2000 Plus, CH2000, MH300, and MS300 slaves. When configuring, it must be configured in sequence between station IDs 1~16, and no vacancies are allowed.
- For RTU or standard EtherCAT slaves, configuration must start from station ID 17, and no vacancies are allowed.

You can set the related information of each device in the **Edit Area** at the bottom of the topology configuration page.

The following table lists the settings and descriptions of each tab in the **Edit Area**.

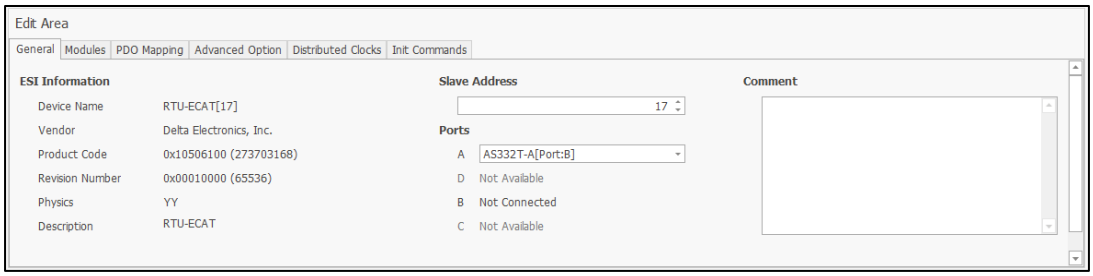
General (Master device)



Function	Description
Name	Display the name of master device.
Model Name	Display the model of master device.
Cycle Time (μs)	This item can be set according to different models.

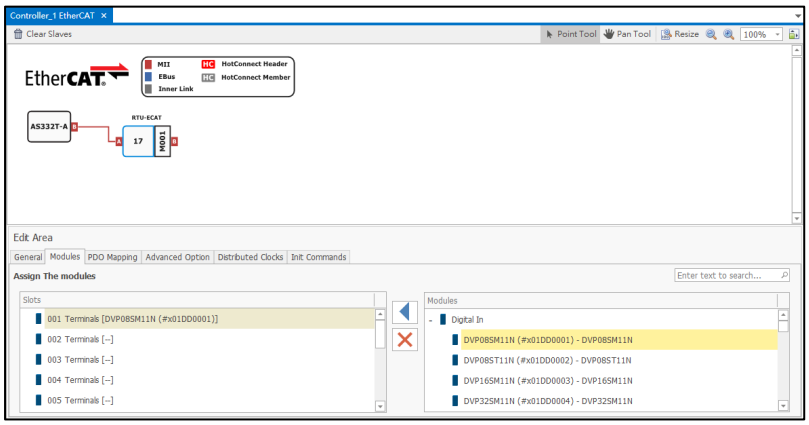
When downloading the project, notify the PLC CPU disable all slaves	If checked, all slave configurations will be cleared when downloading the project.
Comment	Add a comment for the master device.

General (Slave device)



Function	Description
ESI Information	Display the device name, vendor, product code, revision number and other information.
Slave Address	Set the slave station ID.
Ports	Correspond to the port number connecting the master / slave devices in the **EtherCAT Topology Configuration** dialog.
Comment	Add a comment for the slave device.

Module (Only available for RTU module)



In the **Modules** tab, add / remove the module to the right-side of the slave device.

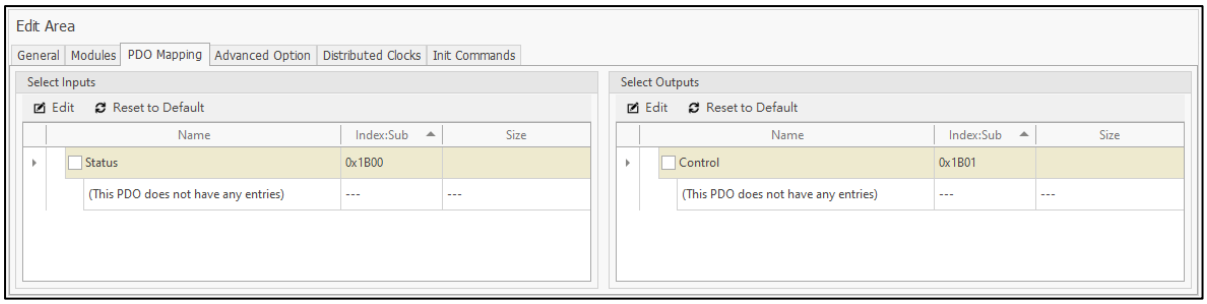
Do the following.

- To add a module, select the module to insert in the right-side **Modules** list, and

then click  to insert the module to the left-side **Slots** list.

- To delete a module, select the module to remove in the left-side **Slots** list, and then click  to remove the module.

PDO Mapping



PDO Mapping is used for data exchange between devices. Edit the input setting of this slave on the left-side **Select Inputs** area, and edit the output setting of this slave on the right-side **Select Outputs** area.

Advanced Option



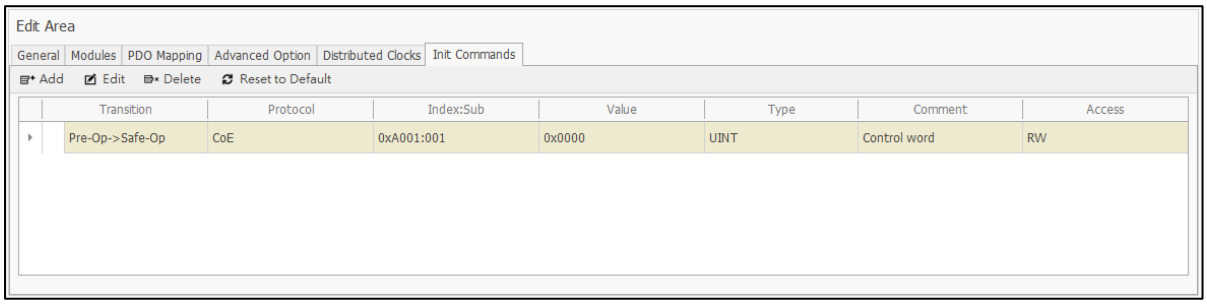
Function	Description
Check Vendor ID	If checked, the system checks the vendor ID when the device is powered on.
Check Product Code	If checked, the system checks the product code when the device is powered on.
Check Revision Number	If checked, the system checks the revision number when the device is powered on.

Distributed Clocks



Function	Description
Operation Mode	Options vary from device to device and include Free Run and SM-Synchron.
Sync Unit Cycle (µs)	Synchronize with the **Cycle Time** setting of the master device.

Init Commands



Function	Description
Add	Click to add an Init command.
Edit	Click to edit an Init command. **Note**: This function is available after the Init command is added.
Delete	Click to delete an Init commands. **Note**: This function is available after the Init command is added.
Reset to Default	Click to reset the set Init commands to the default value.

7.4.1 EtherCAT I/O Mapping

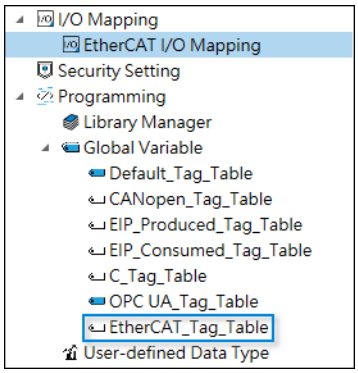
EtherCAT I/O Mapping is used to plan the corresponding relationship between EtherCAT PDO mapping and variables. After the planning of EtherCAT topology is

completed according to EtherCAT Topology Configuration, the EtherCAT I/O mapping can be used.

To open EtherCAT I/O Mapping dialog

- In **Project Explorer**, double-click **I/O Mapping** > **EtherCAT I/O Mapping**.

The variable information in this dialog corresponds to the settings configured in the **PDO Mapping** tab of the **EtherCAT Topology Configuration** dialog, and the **EtherCAT_Tag_Table** containing the corresponding variables is also automatically generated in **Global Variable**.



The following table describes the functions in **EtherCAT I/O Mapping** dialog.

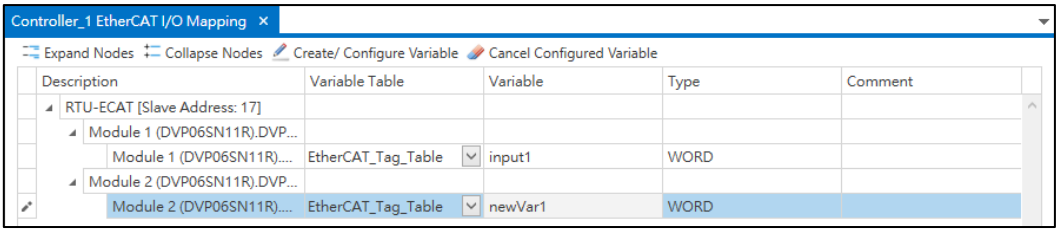
Function / Icon	Description
Expand Nodes	Click to expand the nodes of items.
Collapse Nodes	Click to collapse the nodes of items
Create/ Configure Variable	Click to automatically configure the variables.
Cancel Configured Variable	Click to clear the variables of all items.

Follow these steps to set the variable name.

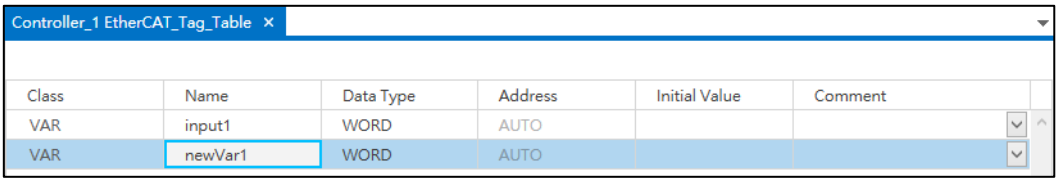
To set variable name

1. Select **EtherCAT_Tag_Table** in the **Variable Table** field of the variable item to be set, and then you can declare the new variable name in the **Variable** field.

   You can also click Create/ Configure Variable on toolbar to automatically generate the variable name.

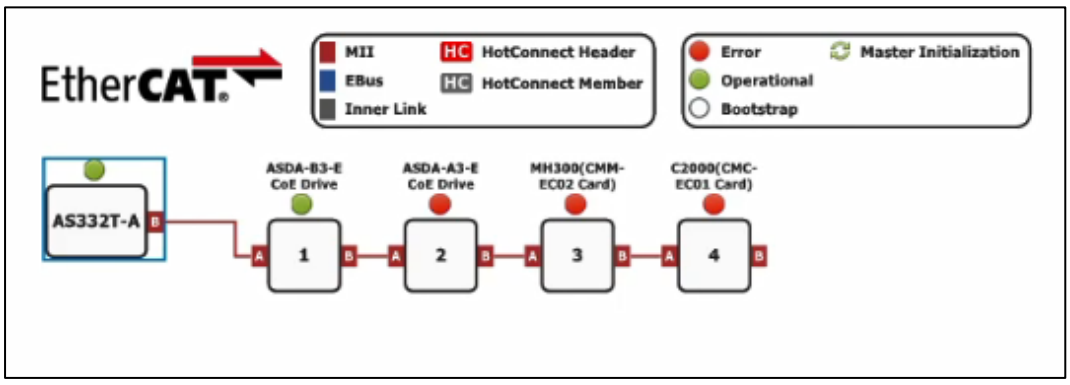


2. Confirm the set variables in the **EtherCAT_Tag_Table** dialog.

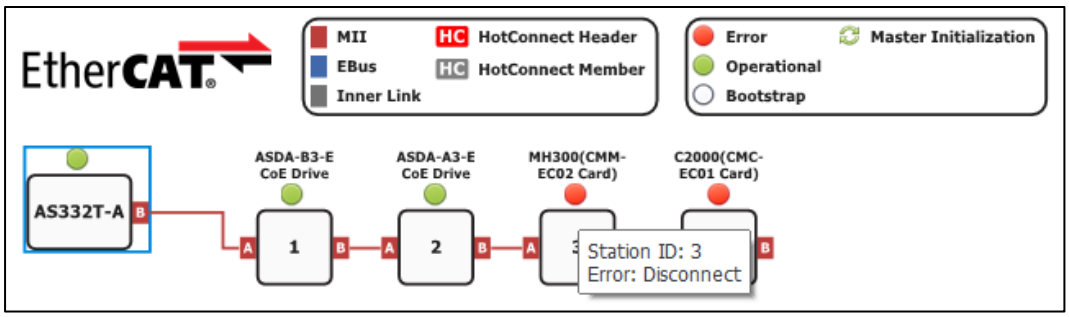


7.4.2 EtherCAT Online Mode

When DIADesigner is in the online mode, you can view the currently communication status of EtherCAT topology in the **EtherCAT Topology Configuration** dialog.



The signal above devices represents their connection status. Move the cursor to the Error signal displays an error message, and then you can fix the error according to the message.



Chapter 8: Programming

8.1 IEC61131-3 Programming

IEC 61131-3, which was published by International Electrotechnical Commission (IEC), standardizes the characteristics of the PLC programming languages. The program codes, which meet the standards, are more readable and compatible, and greatly reduces your learning curve. Based on the IEC 61131-3 standards, DIADesigner integrated programming languages that manufacturers and you could follow. The description and outline of IEC 61131-3 are as follows.

- IEC 61131-3 integrated the programming languages established by PLC developers and defined five programming languages for programming control systems:
    - Instruction List (IL)
    - Structured Text (ST)
    - Ladder Diagram (LD)
    - Sequential Function Chart (SFC)
    - Function Block Diagram (FBD)
- Besides, Continuous Function Charts (CFC) derived from the Function Block Diagrams (FBD) is also a common PLC Programming language.
- The concept related to Variables is adopted. You can replace an address with a Variable. A Program is more readable and the time of assigning addresses is saved.
- The architecture related to Program Organization Units (POU) is adopted. A traditional program is divided into several program organization units which can be developed independently. The architecture of a program can be more modular and can be maintained more easily by means of calling Functions and Function Blocks.
- Program Organization Units (POU) are managed and organized through the concept related to tasks. The development of programs is upgraded to the management of projects. The large-scale development of programs can be managed more easily.

Note: Instruction List (IL) and Function Block Diagram (FBD) are not supported.

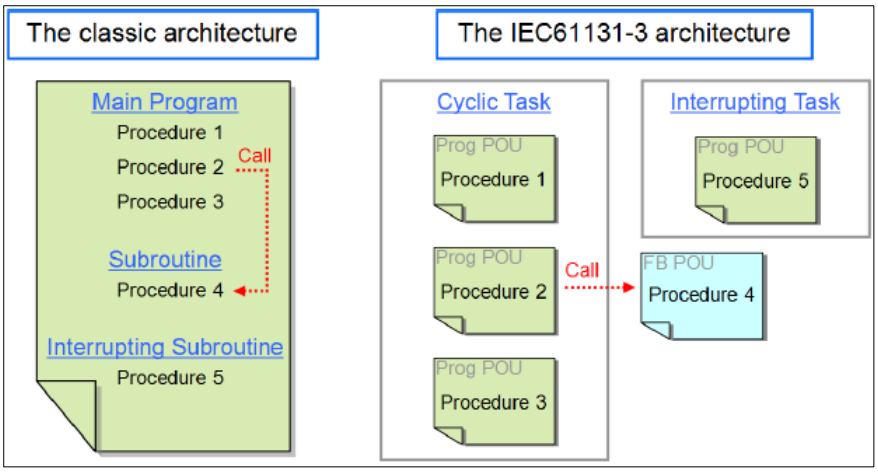
8.2 POU, VAR, Task

8.2.1 POU

Program Organization Units (POUs) and Tasks are important concepts related to IEC 61131-3. They upgrade programming from traditional program writing to project management. The following table lists the differences between traditional Ladder Diagram (LD) and IEC61131-3.

Traditional Ladder Diagram	IEC 61131-3
Main Program	● A single program is divided into several independent Program Organization Units (POUs). ● A POU created is assigned to a cyclic task.
General Subroutine	● A subroutine is a function block. ● A function block created can be called by a POU.
Interrupt Subroutine	● Different interrupt subroutines are independent POUs. ● A POU is assigned to an interrupt task.

A comparison of traditional Ladder Diagrams and IEC 61131-3 is represented in the following figure. In the classic architecture, you write a source code. In the IEC 61131-3 architecture, a program is divided into several POUs. Every POU can be developed independently and can be assigned to a task.



If a control system reaches a certain scale, the internal control procedures get quite complex and the relation between the procedures are closer. If the classic architecture is adopted, a lot of time and manpower must be invested in developing and maintaining a program. However, in the IEC 61131-3 architecture, the POUs are reusable and modular. Besides, the POUs can be integrated easily by means of the management of the tasks. The concepts related to POU and tasks are introduced in the later sections.

8.2.1.1 POU Architecture and Types

Program Organization Units (POUs) are basic elements of a PLC Program. They are different from a traditional program in a PLC. The characteristic of the program architecture introduced by IEC 61131-3 is that a program is divided into several units called Program Organization Units.

In the classic architecture, a source code for a PLC is composed of all procedures including subroutines. If the size of the program is bigger, the maintenance and the debugging of the program becomes difficult. In the IEC 61131-3 architecture, a program is divided into several units according to the functions or characteristics. It is convenient to develop and maintain a program. Besides, since the POUs are modular, different POUs can be developed by different designers. It benefits the distribution of manpower and the execution of the project.

There are two types of POUs in DIADesigner.

- Program (PROG)
- Function Block (FB)

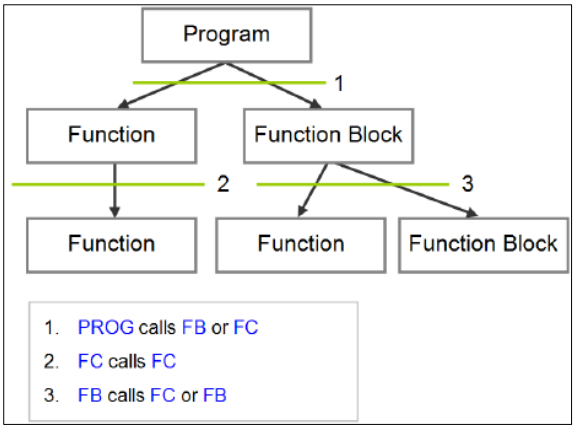
Program (PROG)

A POU of the program type plays a role according to the task to which it is assigned. If a program POU is assigned to a cyclic task, the POU of the program acts as a main program. If the POU is assigned to an interrupt task, the POU of the program acts as a subroutine. Besides, a POU of program type can call a Function Block (FB).

Function Block (FB)

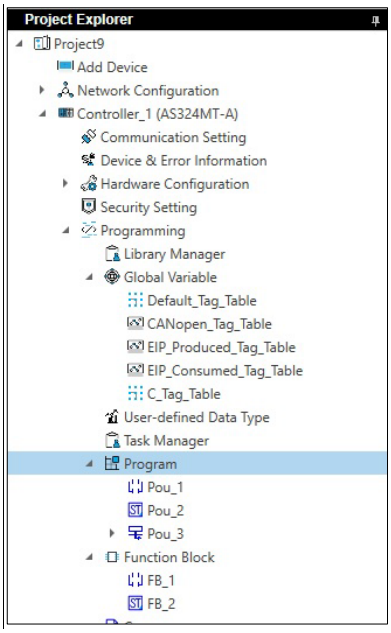
A static variable can be declared in a Function Block (FB). As a result, the value of the variable after an operation can be retained. Since the operation is performed on the value memorized in the function block and an input value, the output values may be different even if the input values are the same. Besides, a function block can call another function block.

The relation among POUs is shown in the following figure. A function (FC) is a type of POU. Since DIADesigner currently does not support Function POU, this part will not be covered in this chapter.



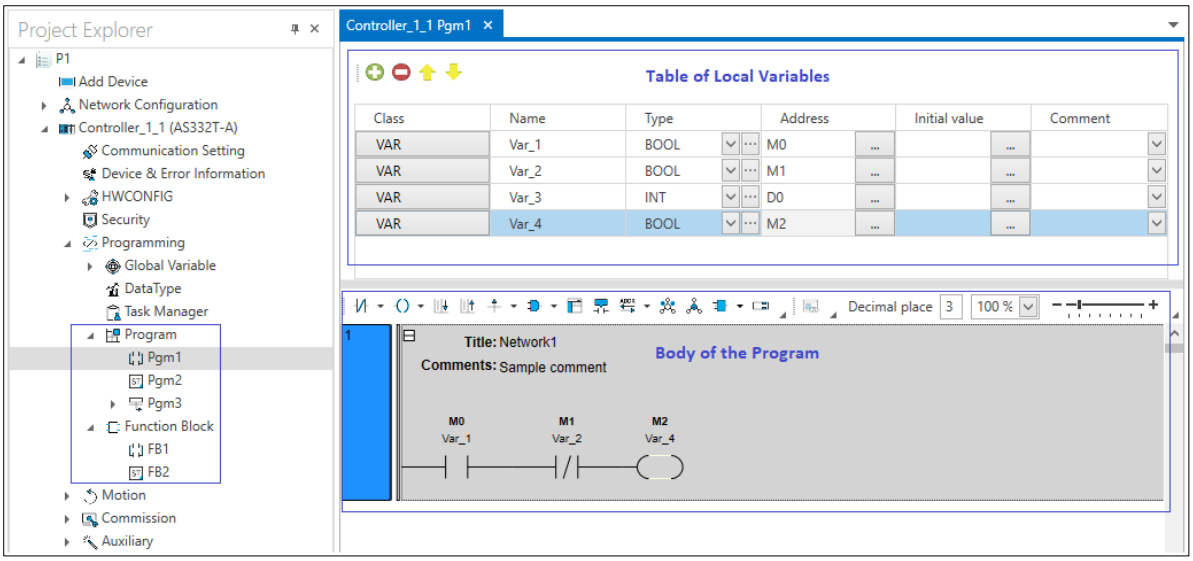
POUs in DIADesigner

In DIADesigner, you can create / manage the POUs in **Project Explorer**. The POUs of the program type and the POUs of the function block type are managed separately. Besides, the programming language in which a POU is written determines the icon representing the POU.



Icon	Description
	POUs of Program types are organized here.
	POUs of Function Block type are organized here.
	POU is in Ladder Diagram (LD).
	POU is in Structured Text (ST).
	POU is in Sequential Function Chart (SFC).
	POU is in C language (C).

Upon double-click a Program or Function Block in **Project Explorer**, a POU editing dialog appears. The POU editing dialog is composed of two parts: A table of local variables is at the upper part of the dialog and the body of the program is at the lower part of the dialog. Besides, the editing environment at the lower part of the dialog varies with the programming language used.



Refer to the following sections for more information about variable tables and program editing.

8.2.1.2 Managing Program Organization Unit (POU)

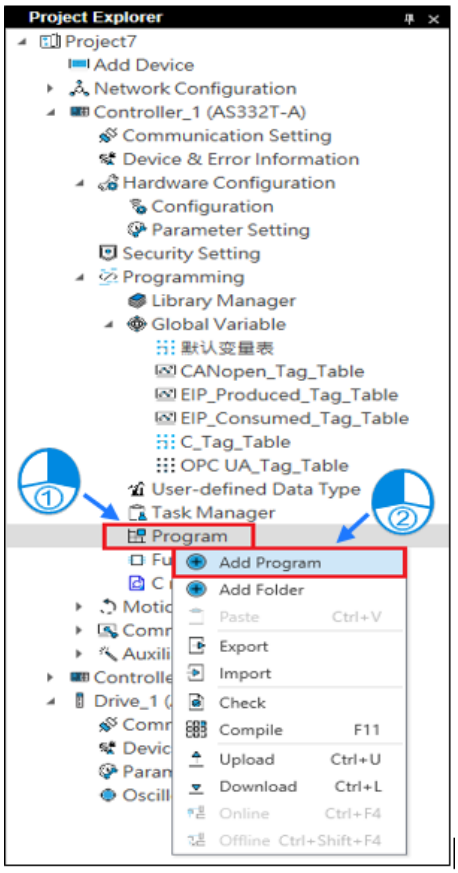
Program Type POU

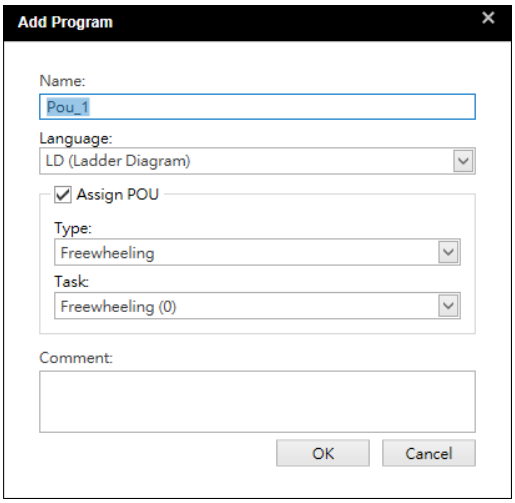
Create program type POUs in **Project Explorer** > **Controller Device** > **Programming** > **Program**.

Follow these steps to create a Program type POU.

To create Program type POU

1. Right-click **Program**.
2. Click **Add Program** in the context menu.





The following table describes the functions in the **Add Program** dialog.

Function / Icon	Description
Name	Enter a POU name.
Language	Select POU language. The available options include: • LD (Ladder Diagram) • ST (Structured Text) • SFC (Sequential Function Chart) • C (C Language)

	• CFC (Continuous Function Chart) **Note**: **LD** is the default value.
Assign POU	Set POU type and Task.
Comment	Enter the comments for POU.

3. Enter a POU **Name**, select program **Type**, a programming **Language**, enter **Comment** (optional) and click **OK** to create Program type POU.

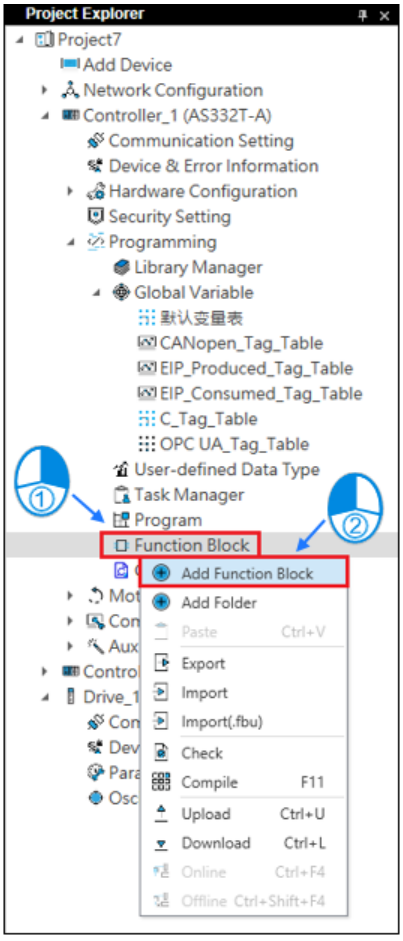
Function Block Type POU

Create function block type POUs in **Project Explorer** > **Controller Device** > **Programming** > **Function Block**.

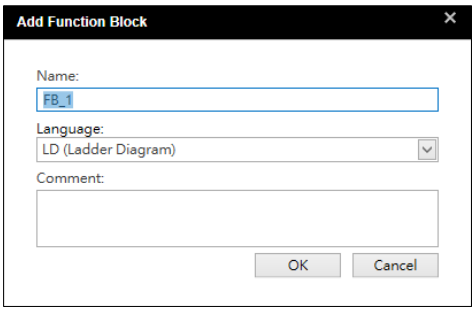
Follow these steps to create a function block type POU.

To create function block type POU

1. Right-click **Function Block**.



2. Click **Add Function Block**.



3. Enter a POU **Name**, select a programming **Language**, enter **Comment** (optional) and click **OK**.

   Note: You must set the attributes of the POU added in the **Add Function Block** dialog. The attributes of the POU of the program type is slightly different from the POU of the function block type and they are described in the following table.

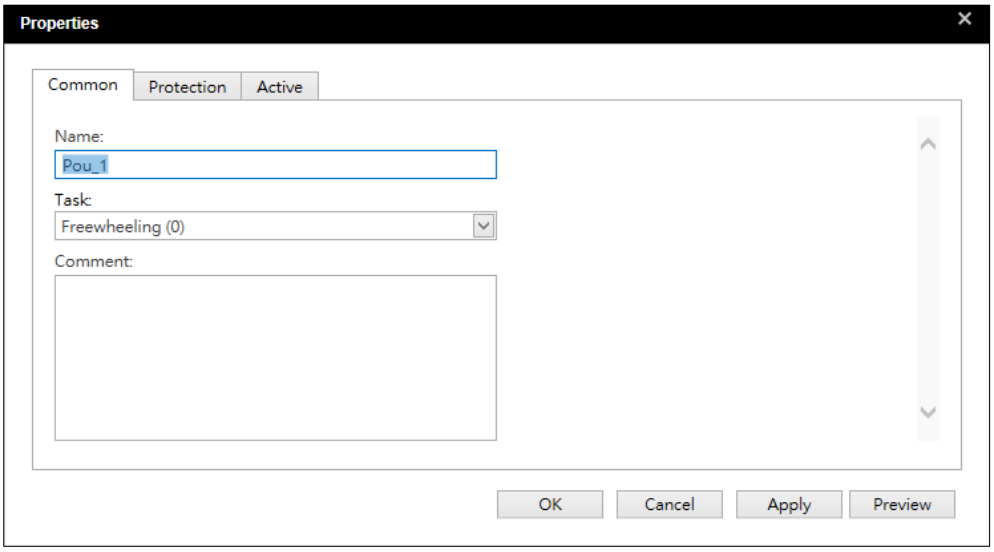
Function / Icon	Description
Name	Enter a POU name.
Language	Select Implementation language. The available options include: • LD (Ladder Diagram) • ST (Structured Text) • C (C Language) • CFC (Continuous Function Chart) **Note**: **LD** is the default value.
Comment	Enter the comments for POU.

POU Name has the following characteristics.

1. Special characters and spaces cannot be used. For example, *, #, ?, \, %, @ etc. cannot be used.
2. A POU name is composed of thirty characters at the most. It is necessary to note that a Chinese character occupies 2 characters.
3. Underline can be used, but they cannot be used continuously or used at the end of a POU name. For example, POU_1 is a valid name, but POU__1 and POU_1_ are invalid names.

POU Properties

You can view the properties of POUs by **POU Name** > Context Menu > **Properties**.



The **POU Properties** dialog displays the properties in three different tabs which are described in the following table.

Tab	Property	Description
Common	Name	Edit the program name.
	Task	Set the task.
	Comment	Edit the comment for program.
Protection	Please enter password (4 ~ 16 character)	Enter a password.
	Reconfirm	Enter the password again.
	Show password	Select to enable / disable the visibility of characters in the password.
Active	Active Check box	Select to enable / disable the program type POU.
N/A	OK	Click to save the setting.
	Cancel	Click to cancel the changes made.
	Apply	Click to save the setting and close the dialog.
	Preview	Click to the preview the setting.

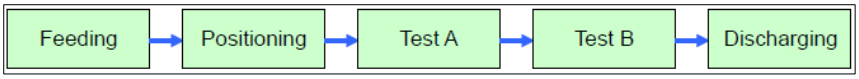
Enabling a POU

DIADesigner provides the function of disabling a POU temporarily. If a POU is disabled, it is skipped when the program is compiled and the program in the POU is not executed. However, only POUs of the program type can be disabled.

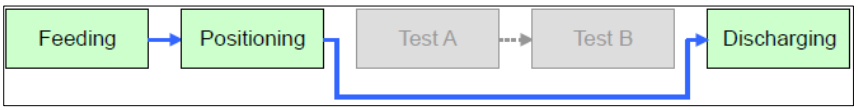
This function can be applied to the test or debug a program. To skip certain procedure during the test or debug of the program, disable the POU. After the test or debug of the program is complete, you can enable the POU.

Example

The equipment shown in the following figure is composed of five workstations and the working practices are divided into five POUs of the program type.



If the assembling of the equipment is complete, but the assembling of the test instrument is not complete, disable the POUs corresponding to Test A and Test B. The two workstations are skipped when the other parts of the equipment are adjusted.



To enable a POU, select the **Active** checkbox in the **Properties** dialog. If the Active checkbox is not selected, then the POU is disabled. A POU which is disabled is represented by a gray icon.

Deleting a POU

Delete a POU of Program type or Function Block type. But the deleted POUs cannot be recovered. When a POU is deleted, it is removed from the Task associated earlier.

Delete a POU in **Project Explorer** by any of these methods.

- Right-click **POU** to open the context menu and click **Delete**.

  Or

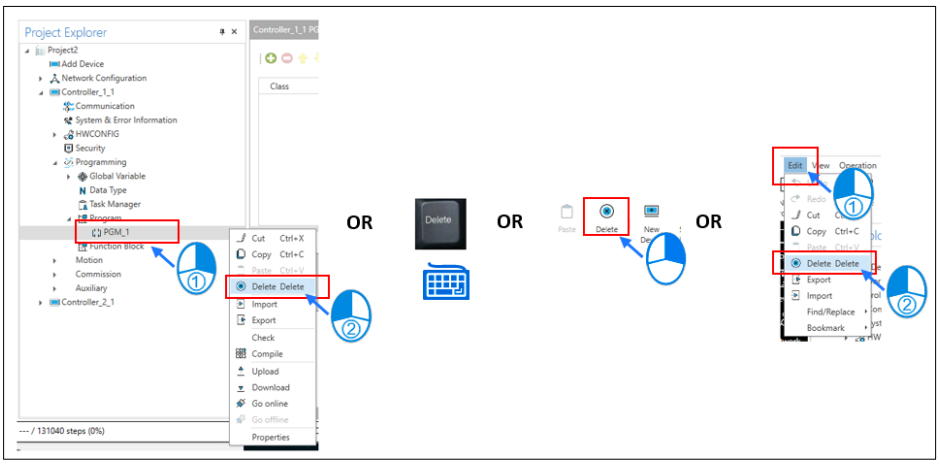
- Select the **POU** to be deleted and press **Delete** key on the keyboard.

  Or

- Select the POU to be deleted, and click  on DIADesigner toolbar.

  Or

- Select the POU to be deleted and click **Edit** > **Delete** in the menu item.



Copying a POU

Copy a POU in **Project Explorer** by any of these methods.

- Right-click **POU** to open the context menu and click **Copy**.

  Or

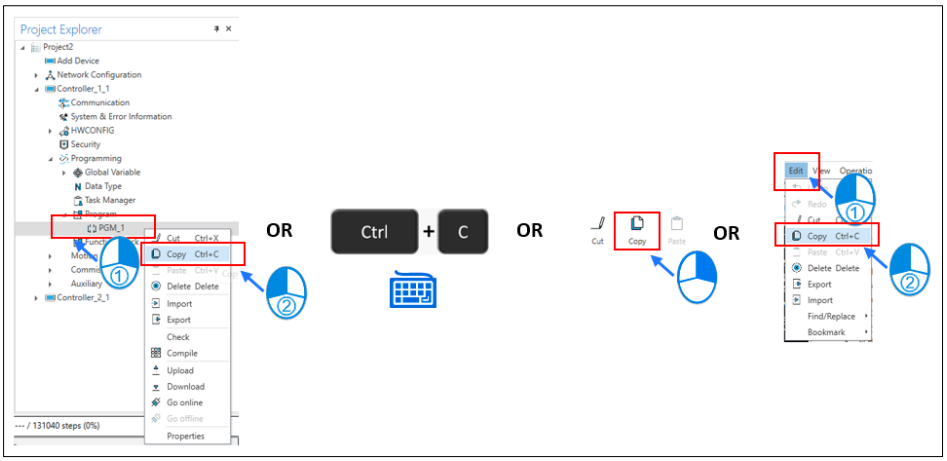
- Select the **POU** to be copied and press **Ctrl** + **C** keys simultaneously on the Keyboard.

  Or

- Select the POU to be copied, and click  on DIADesigner toolbar.

  Or

- Select the POU to be copied and click **Edit** > **Copy** in the menu item.



Pasting a POU

Paste a POU in **Project Explorer** by any of these methods.

- Right-click **Program** or **Function Block** to open the context menu and click **Paste**.

  Or

- Select **Program** or **Function Block** and press **Ctrl** + **V** keys simultaneously on the keyboard.

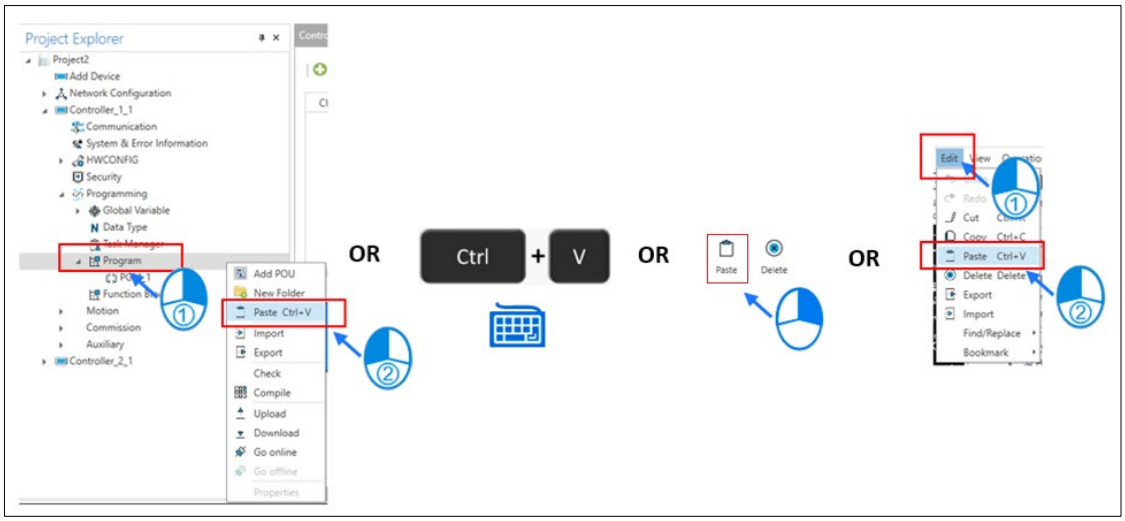
  Or

- Select **Program** or **Function Block** and click 

in the DIADesigner.

  Or

- Select **Program** or **Function Block** and click **Edit** > **Paste** in the menu item.



Note: A POU can be copied into the same or different device in the project. If the POU copied is a POU of the program type, it cannot be pasted into the Function Blocks. If the POU copied is POU of the function block type, it cannot be pasted into the Program. Besides, if the POU copied is a POU of the program type, the POU pasted will not be assigned to any task.

Setting and Unlocking a POU Password

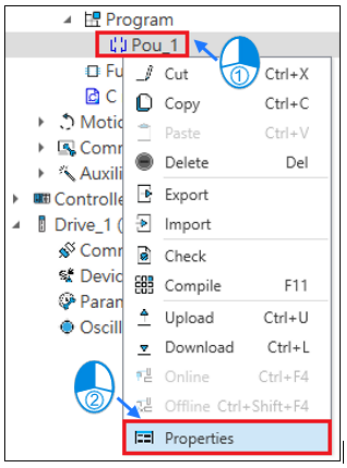
Set a password for a POU in DIADesigner. The setting and unlocking the POU password are described as follows.

When creating a POU, set a password for the POU. If you do not set a password for the POU during the time of its creation, protect the POU with a password later by opening the **Properties** dialog by clicking the **Properties** option in the context menu of the POU.

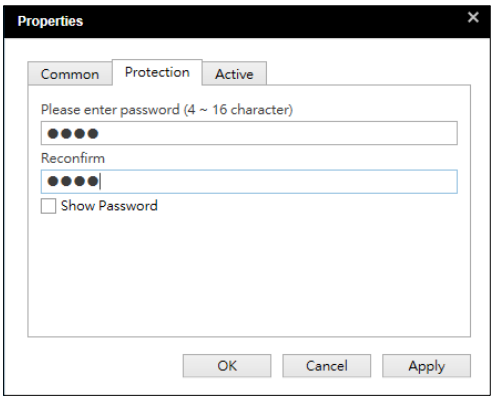
Follow these steps to set a password to POU:

To set password to POU

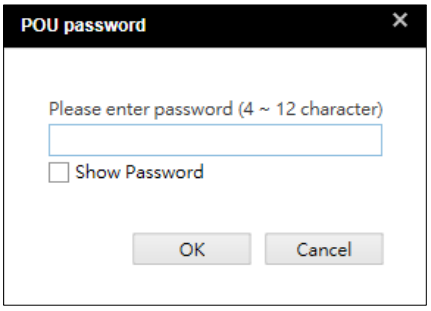
1. Right-click POU on which password must be set to open its context menu.
2. Click **Properties** in the context menu to open **Properties** dialog.



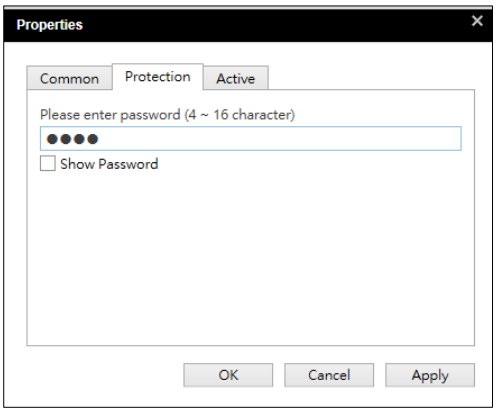
3. Type the same password in the **Please enter password** box, then in the **Reconfirm** box and click **OK / Apply** button.



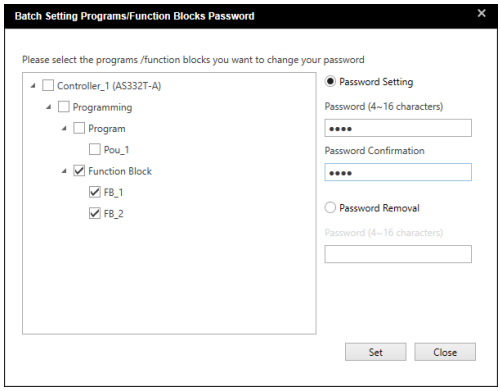
Note: English letters, numeric digits, special characters can be typed in the boxes. Once the POU is protected with password, the systems ask you to type the password whenever the **POU password** dialog is opened.



To unlock password, open **Properties** dialog again. Type the password in the **Please enter the password** box and click **OK**.



Right-click **Programming** and select **Batch Setting Programs/Function Blocks Password** to set or remove the password of multiple programs / function blocks.



Exporting a POU of Program Type

DIADesigner supports the function of exporting POUs of program type.

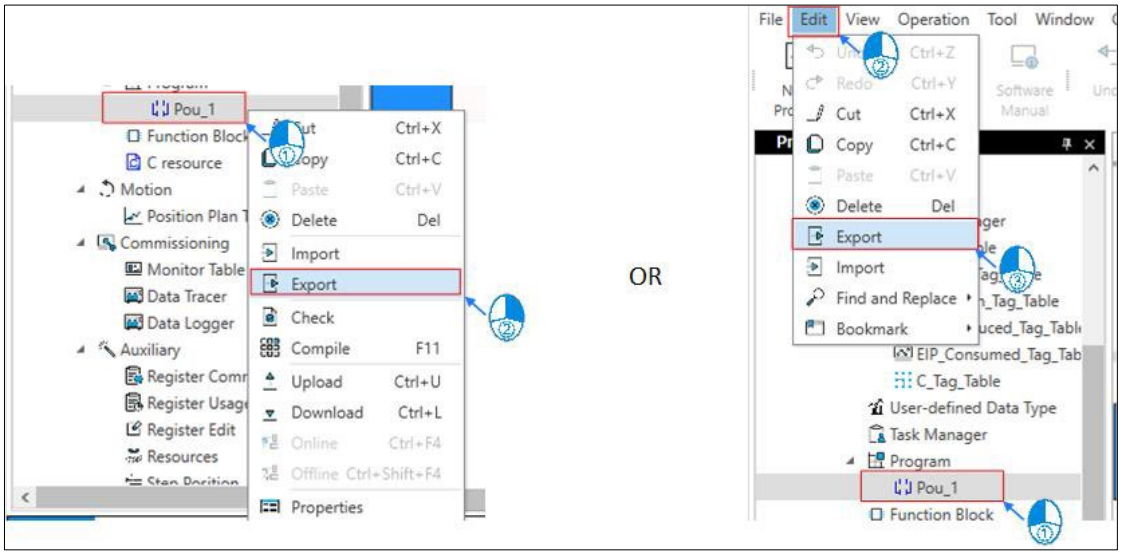
Follow these steps to export a POU of program type.

To export POU of program type

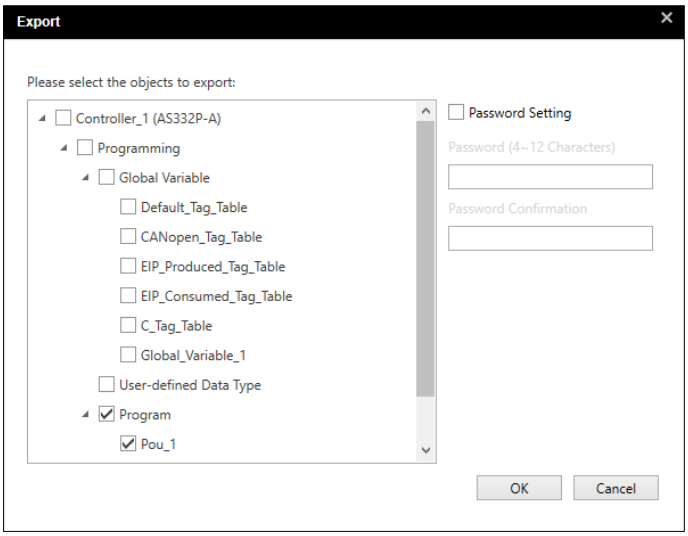
1. Right-click POU to be exported.
2. Click **Export** in the context menu.

   Or

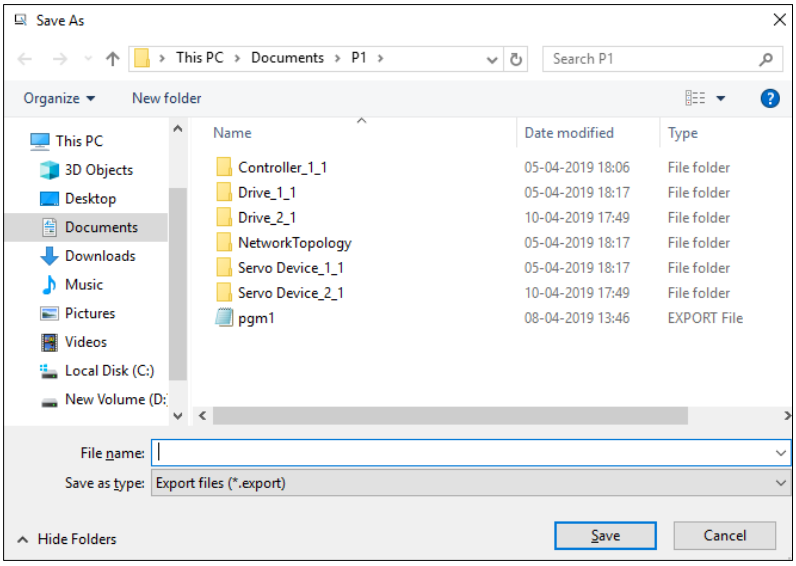
   Click **Edit** > **Export** in the menu item.



Note: POU to be exported is selected by default.



3. Select the object(s) to export, and click **OK** to open **Save As** dialog.



Note: To protect a file to be exported with a password, select the **Password Setting** checkbox, enter the password in **Password** box and enter the same once again in **Password Confirmation** box.

4. Select a particular location and click **Save**.

Importing a POU of Program Type

DIADesigner supports the function of importing POUs of program type.

Follow these steps to import a POU of program type.

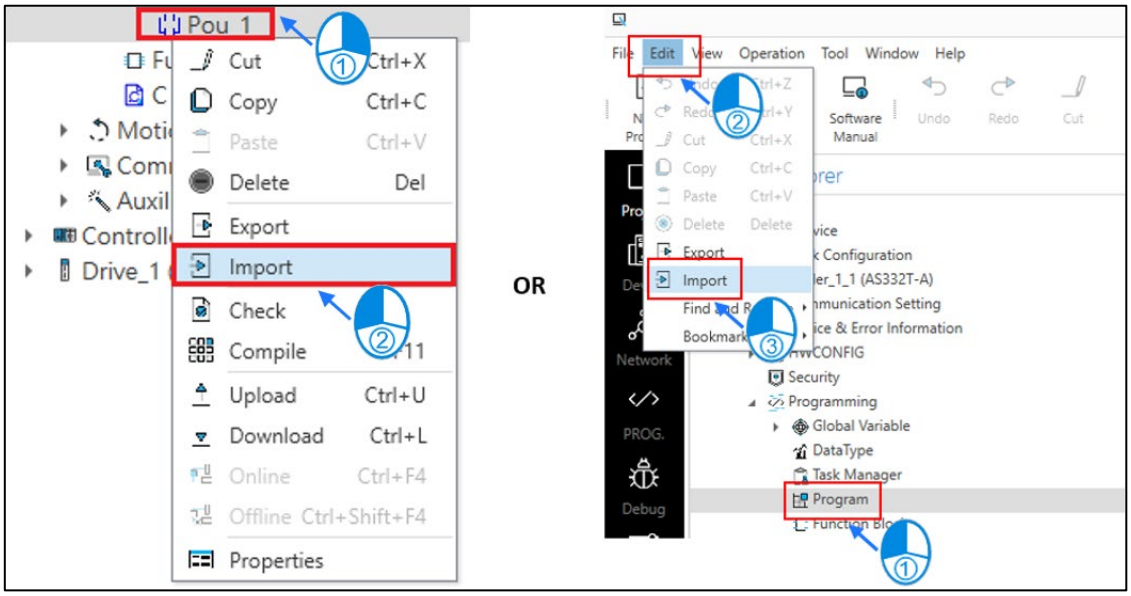
To import a POU of program type

1. Select a device in **Project Explorer** to which the program type POU has to be imported.

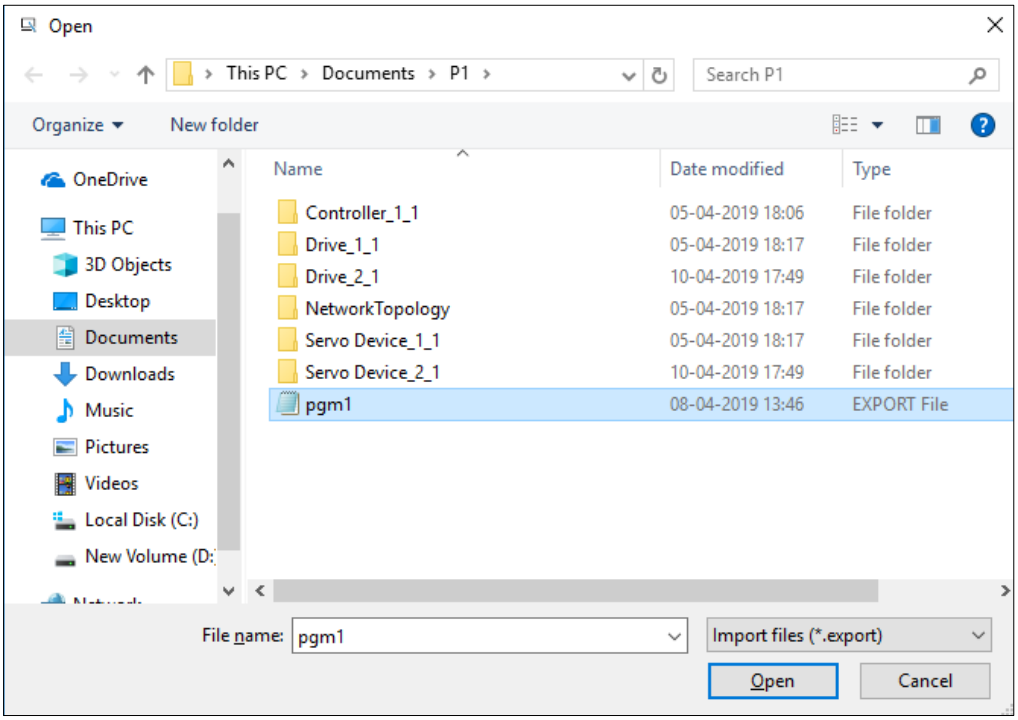
2. Right-click **Program** in the device to open the context menu and click **Import**

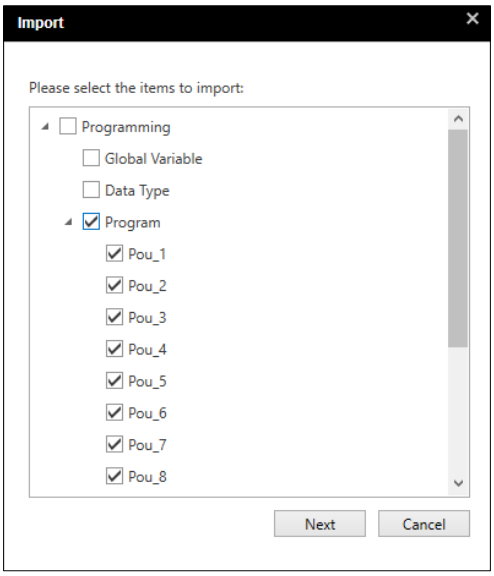
   Or

   Click **Edit** > **Import** in the menu item.



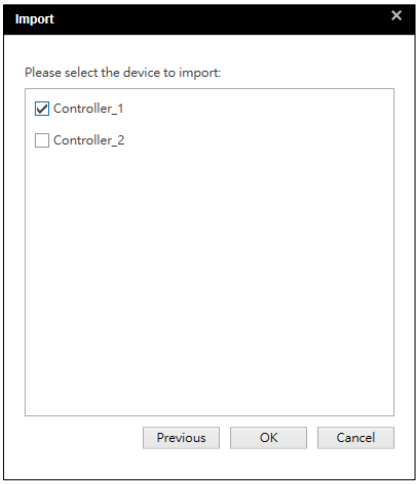
3. Select the file to be imported and click **Open** to open **Import** dialog.



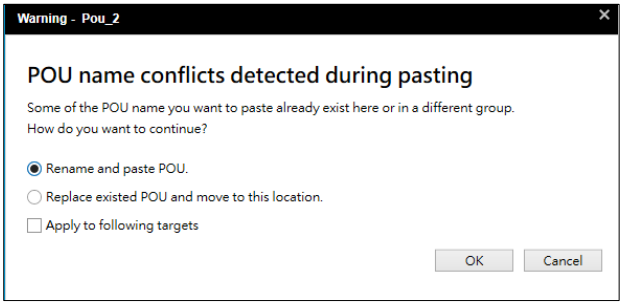


4. Select the items to be imported and click **Next**.

5. Select the device in which the POU should be placed and click **OK**.



Note: If the file is protected with a password, type the password in the **Password** for decrypting window and click **OK**. The POUs which are imported will not be assigned to any tasks. Besides, if the name of the POU which is to be imported is the same as the name of a POU in the project, it prompts to rename or replace for proceeding.



Exporting a POU of Function Block Type

DIADesigner supports the function of exporting POUs of function block type.

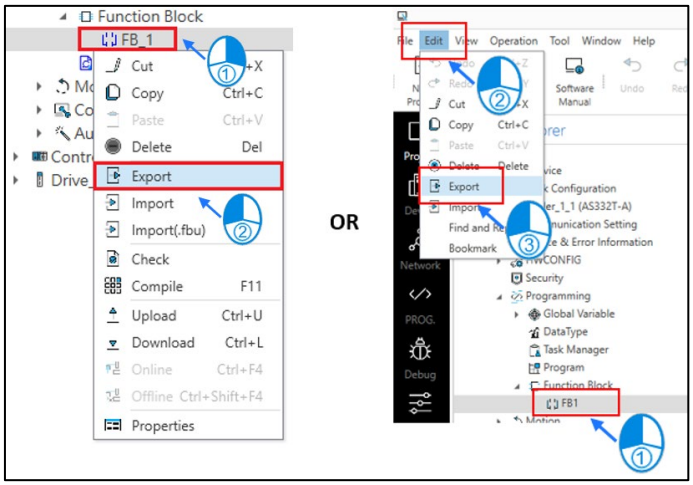
Follow these steps to export a POU of function block type.

To export a POU of function block type

1. Select the function block type POU to be exported.
2. Right-click to open the context menu and click **Export**

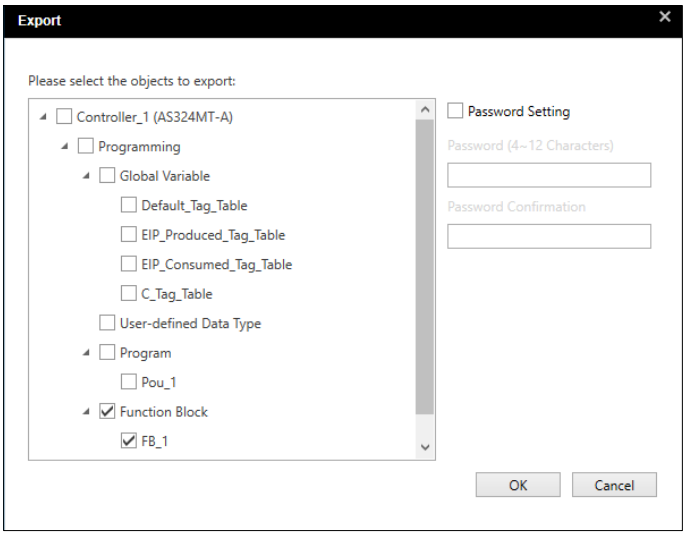
   Or

   Click **Edit** > **Export** menu item.

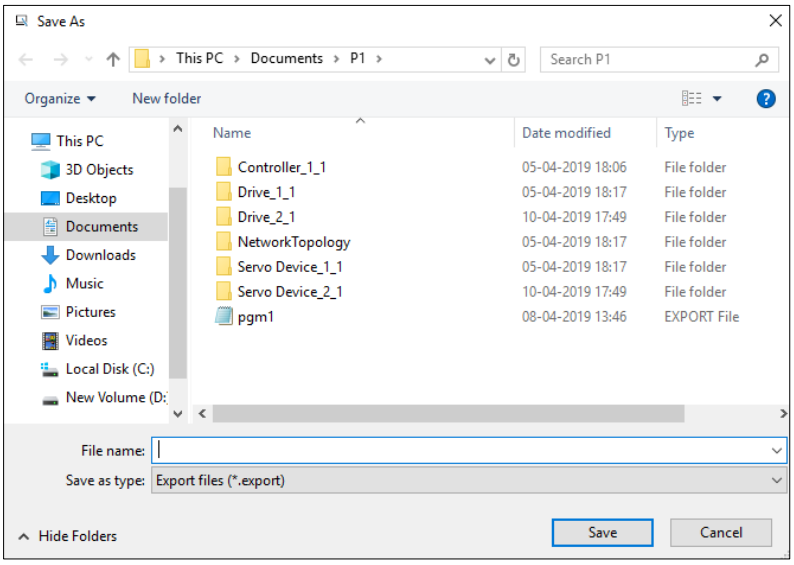


   Note: POU to be exported is selected by default.

3. Select the objects to export. Click **OK** to open **Save As** dialog.



Note: To protect a file to be exported with a password, select the **Password Setting** checkbox, enter the password in **Password** box and enter the same once again in **Password Confirmation** box.



4. Select a particular location and click **Save**.

Importing a POU of the Function Block Type

DIADesigner supports the function of importing POUs of function block type.

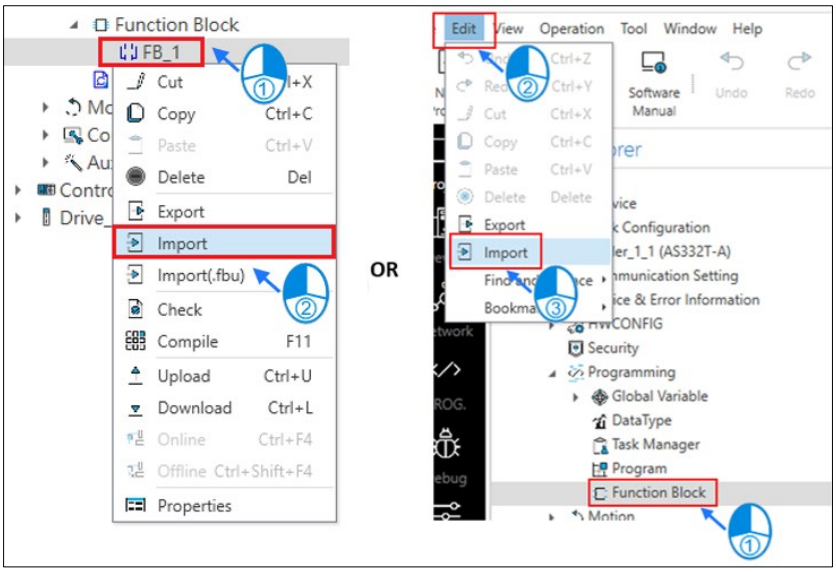
Follow these steps to import a POU of function block type.

To import POU of function block type

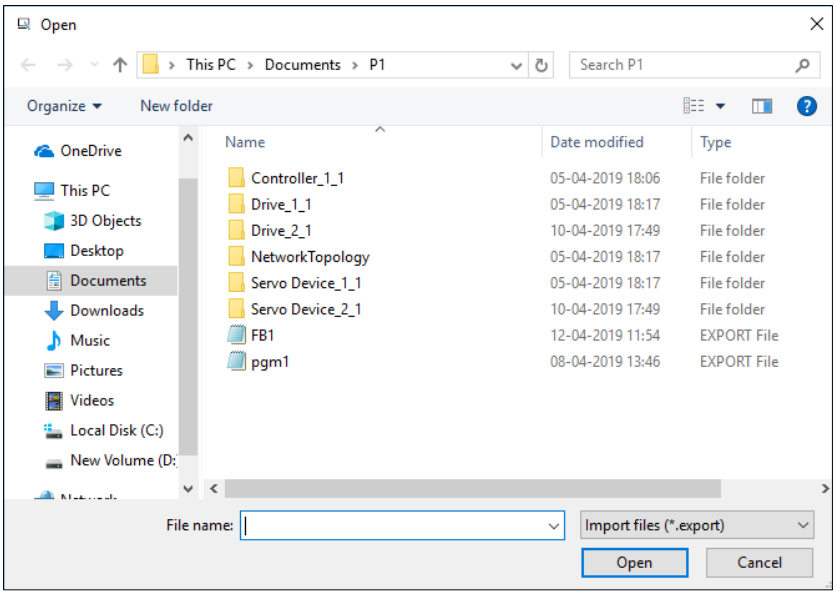
1. Select a device in **Project Explorer** to which the function block type POU has to be imported.
2. Right-click **Function Block** in the device to open the context menu and click **Import**.

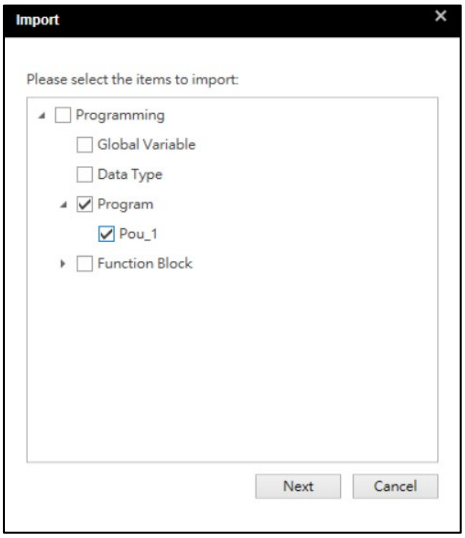
   Or

   Click **Edit** > **Import** in the menu item.



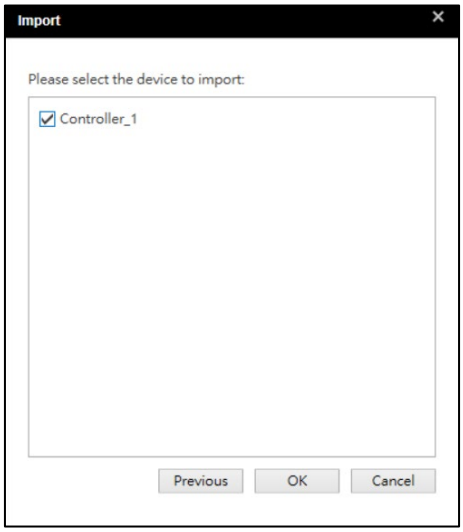
3. Select the file to import and click **Open** to open **Import** dialog.



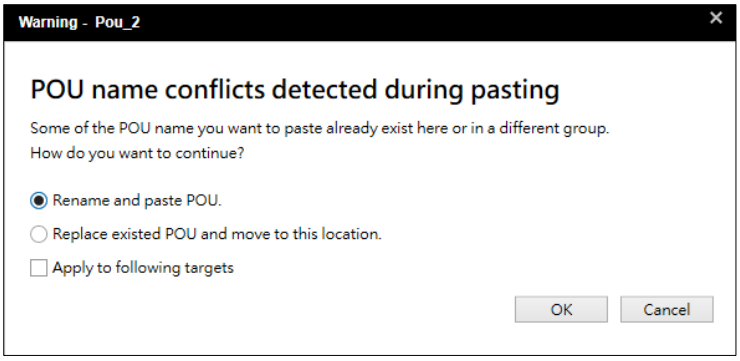


4. Select the items to be imported and click **Next**.

5. Select the device in which the POU should be placed.



Note: If the file is protected with a password, type the password in the **Password** for decrypting window and click **OK**. The POUs which are imported will not be assigned to any tasks. Besides, if the name of the POU which is to be imported is the same as the name of a POU in the project, it prompts to rename or replace for proceeding.



Importing a POU (.fbu) of the Function Block Type

DIADesigner supports the function of importing POUs of function block exported from ISPSoft .fbu file. This function increases the convenience of reusing the existing project.

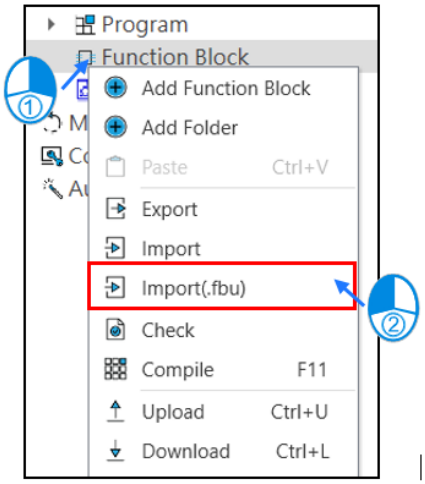
Follow these steps to import a POU (.fbu) of function block type.

To import POU (.fbu) of function block type

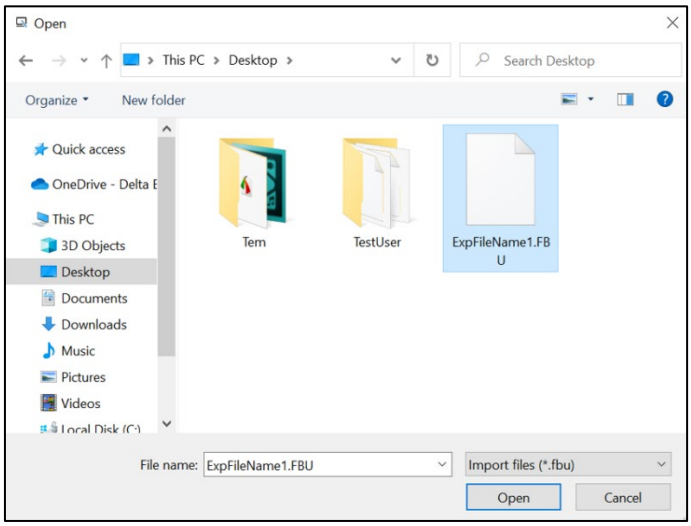
1. Select **Function Block** in **Project Explorer**.

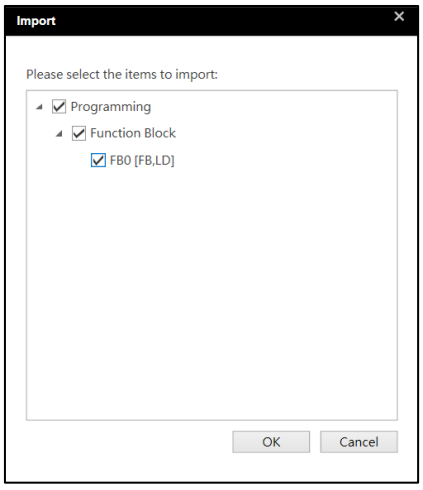
2. Right-click **Function Block** and select **Import (.fbu)** in the context menu to open

Open dialog.



3. Select the file to import and click **Open** to open **Import** dialog. The function block file extension is .fbu.





4. Select the items to be imported and click **Next** to import function block.

Note: Refer to *ISPSoft User Manual* for export function of ISPSoft.

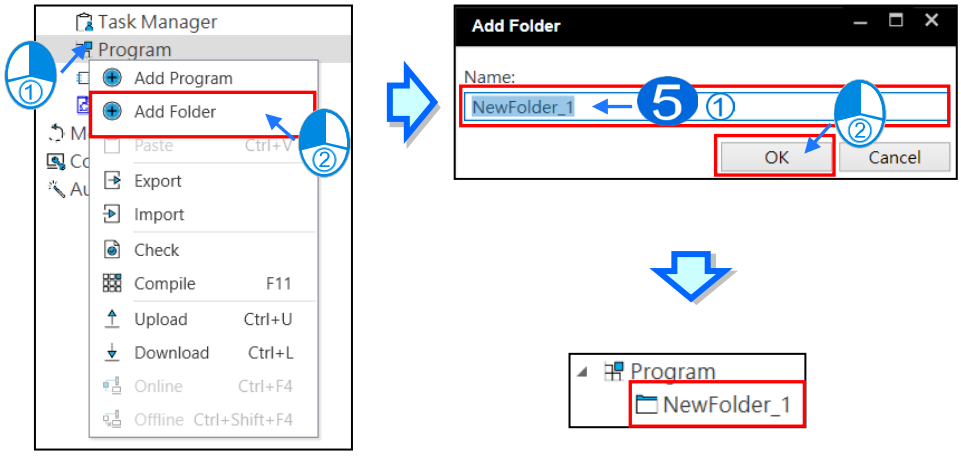
Folder management of POU

For different kinds of industries and applications, POUs always need to be sorted by some categories. DIADesigner supports to create folders for POUs in **Project Explorer**.

Follow these steps to create folders for POU.

To create folders for POU

1. Right-click **Program** or **Function Block** in **Project Explorer**.
2. Click **Add Folder** in the context menu.
3. Enter the folder name in **Add Folder** dialog, and then click **OK**.

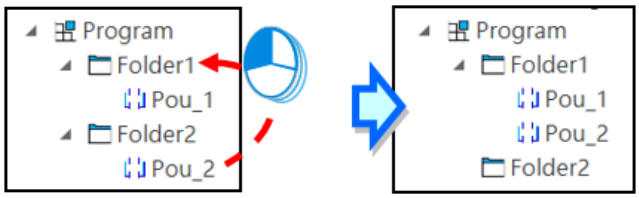


In a folder, you can create another folder or POU inside. There is no limitation of level number in DIADesigner. You can create any number of folders and also move the content into other folders.

Follow these steps to move the content to another folder.

To move content to another folder

1. Open the folder of **Program** or **Function Block** in **Project Explorer**.
2. Drag and drop the **POU** or **Folder** to the target folder.

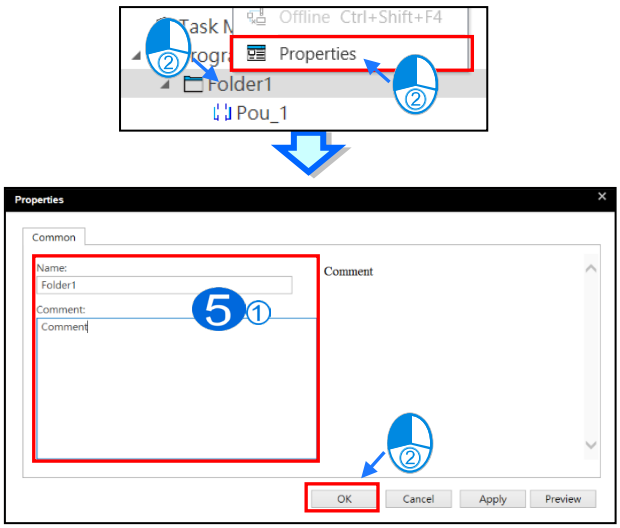


Add the comments as per the folder to your application requirements.

Follow these steps to add comments.

To add comments

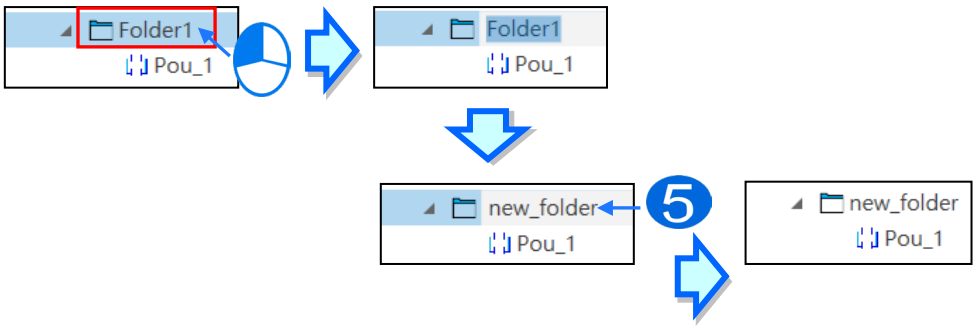
1. Open the folder of **Program** or **Function Block** in **Project Explorer**.
2. Right-click the folder in which you want to add the comments to open the context menu.
3. Click **Properties**, and then enter **Name** and **Comment**. It will also be displayed in the preview area. Click **OK** to save the result.



Follow these steps to change the folder name.

To change folder name

1. Click the folder in **Project Explorer**, and then click it again to enter the text editing mode.
2. Enter the new folder name, and then click **Enter**.



8.2.2 VAR

During the process of developing a traditional program for a PLC, it generally takes much time to manage addresses. Besides, managing or debugging the program in a big project is difficult. As a result, the concept of **Variables** in a high-level programming language is introduced in IEC 61131-3. An address in a PLC can be represented by a variable and an address can be automatically assigned to a variable. This saves the time for assigning addresses, increases the program readability and the efficiency of developing a program increases.

8.2.2.1 Application of Variables and Creation of Variables

A variable has to be declared before it is used. There are two types of variables – local variables and global variables. Local variables can only be used in the POU in which they are declared. Global variables can be used in all the POUs in a project. Besides, the identifier of a local variable in a POU can be the same as that of identifier of a local variable in another POU. However, if the identifier of a local variable declared is the same as the identifier of a global variable, the system will automatically regard the local variable in a POU as a local variable.

The regulations of creating the identifier of a variable are as follows.

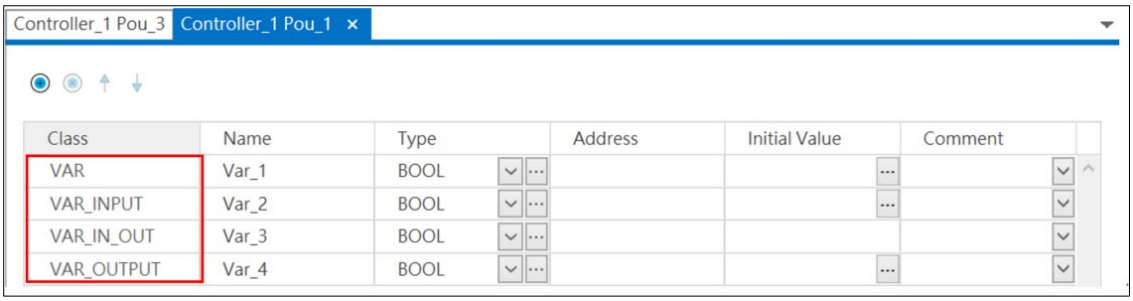
- An identifier can contain a maximum of 42 characters (a Chinese character occupies 2 characters).
- The identifier of a variable cannot be a name reserved by the system, example an instruction code, an address or a keyword. However, if a name reserved by the system is a part of the identifier of a variable, the identifier is a valid name. For example, *M0* is an invalid name, but *_M0* is a valid name.

- The name of a variable and the name of an enumeration cannot be the same.
- Spaces cannot be used. For example, *INPUT CH0* is an invalid name.
- Underlines can be used, but they cannot be used continuously or put at the end of an identifier. For example, *INPUT_CH0* is a valid identifier, but *INPUT__CH0* and *INPUT_CH0_* are invalid identifiers.
- Special characters cannot be used. For example, *, #, ?, \, %, @ etc. cannot be used.

Variable Classes

In terms of functions, variables can be classified into five classes. Variable classes supported in global variables are the same as programs (local variables). The characteristics and usage description of the programs (local variable) and function blocks (local variable) are as follows.

Variable Classes	Programs (Local Variable)		Function Blocks (Local Variable)	
	AS	DVP	AS	DVP
VAR	✓	✓	Cannot specify location	Cannot specify location
VAR_RETAIN	Cannot specify location	Cannot specify location	-	-
VAR_INPUT	-	-	Cannot specify location	Cannot specify location
VAR_OUTPUT	-	-	Cannot specify location	Cannot specify location
VAR_IN_OUT	-	-	Cannot specify location	Cannot specify location



VAR – General Variable

The variables of this class are for general operations only. The significance of a variable of this class depends on the data type of the variable or the corresponding address.

VAR_RETAIN – Latched Variable

AS series controller supports the variables of latching class. You cannot declare the variables of this class in a function block. The function of the variables of the VAR_RETAIN class is the same as the function of the variables in the VAR class. The difference between the variables of the VAR_RETAIN and the VAR classes lies in the fact that the addresses automatically assigned to the variables of VAR_RETAIN class are latched addresses. As a result, the values of the variables of this class are retained after the controller is disconnected.

Note: You cannot assign an address to a variable of this class. In addition, the data type cannot be Timer, Counter or Step.

VAR_INPUT – Variable used as an Input Pin of a Function Block

A variable of this class is used as an input pin of a function block and is declared in the function block. If a function block is called, the variable of this class can receive the input value from the caller. Besides, in a ladder diagram/function block diagram, the variables of this class are put at the left side of the function blocks and the pins which receive the values from the caller are assigned to the variables of this class.

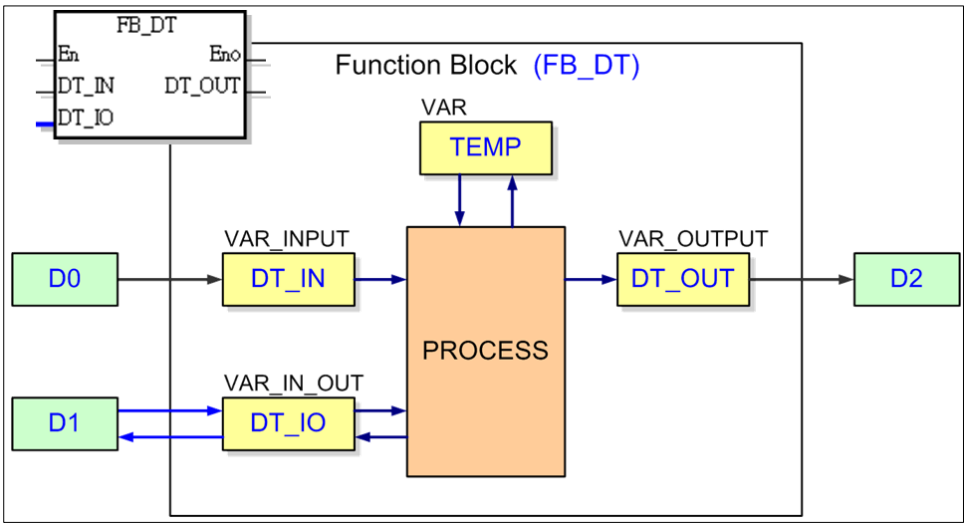
VAR_OUTPUT – Variable used as an Output Pin of a Function Block

A variable of this class is used as an output pin of a function block and is declared in the function block. After the execution of a function block is complete, the operation result is sent to the caller through the variable of this class. Besides, in a ladder diagram /function block diagram, the variables of this class are put at the right side of

the function blocks and the pins which sent the operation results to the caller are assigned to the variables of this class.

VAR_IN_OUT – Variable used as a Feedback Pin of a Function Block

A variable of this class is used as a feedback pin of a function block and is declared in the function block. For example, refer to the below function block FB_DT. When the function block is called, the caller sends the value D1 to DT_IO, which is a variable of the VAR_IN_OUT class. After the operation comes to an end, the final value of DT_IO is sent to D1. Besides in a ladder diagram/function block diagram, the variables of this class are put at the right side of the function block and the blue pins which connect to the caller are assigned to the variables of this class.



8.2.2.2 Data Types

The data type of a variable determines the significance of the value of the variable. Suppose there are two variables — VAR_1 and VAR_2. The data type of VAR_1 is BOOL and the data type of VAR_2 is WORD. When VAR_1 and VAR_2 are used in a program, VAR_1 represents a contact and VAR_2 represents a 16-bit address which can be involved in arithmetic or data transfer.

The data types supported in DIADesigner are listed in the following table. Different models support different data types and the variables which can be declared in the POU of the program type are different from the variables which can be declared in a POU of the function block type.

Data Type	Description	AS Controller		DVP Drive	
		Program	FB	Program	FB
BOOL	Boolean data type, which uses 0 and 1, or TRUE and FALSE A Boolean value represents a state of a contact	✓	✓	✓	✓
BYTE	8-bit value 8-bit data can be stored in the effective range 16#00 ~ FF (the default value is 16#00).	-	-	-	-
WORD	16-bit value 16-bit data can be stored in the effective range 16#0000 ~ FFFF (the default value is 16#0000)	✓	✓	✓	✓
DWORD	32-bit value 32-bit data can be stored in the effective range 16#00000000 ~ FFFFFFFF (the default value is 16#00000000)	✓	✓	✓	✓
LWORD	64-bit value 64-bit data can be stored in the effective range 16#0000000000000000 ~ FFFFFFFFFFFFFFFF (the default value is 16#0000000000000000)	-	-	-	-
SINT	8-bit signed integer The effective range is -128 ~ 127 The highest bit represents a positive/negative sign (the default value is 0)	-	-	-	-
INT	16-bit integer The effective range is -32768 ~ 32767 The highest bit represents a positive/negative sign (the default value is 0)	✓	✓	✓	✓
DINT	32-bit integer The effective range is -2147483648 ~ 2147483647	✓	✓	✓	✓

Data Type	Description	AS Controller		DVP Drive	
		Program	FB	Program	FB
	The highest bit represents a positive/negative sign (the default value is 0)				
LINT	64-bit integer The effective range is -9223372036854775808 ~ 9223372036854775807 The highest bit represents a positive/negative sign (the default value is 0)	-	-	-	-
USINT	8-bit unsigned integer The effective range is 0 ~ 255 (the default value is 0)	-	-	-	-
UINT	16-bit unsigned integer The effective range is 0 ~ 65535 (the default value is 0)	-	-	-	-
UDINT	32-bit unsigned integer The effective range is 0 ~ 4294967295 (the default value is 0)	-	-	-	-
ULINT	64-bit unsigned integer The effective range is 0 ~ 18446744073709551615 (the default value is 0)	-	-	-	-
REAL	32-bit floating–point value The effective range is -3.402823e+38 ~ -1.175495e-38, 0 and1.175495e-38 ~ 3.402823e+38 A decimal is represented by a 32-bit value (the default value is 0.0)	✓	✓	✓	✓
LREAL	64-bit floating–point value The effective range is -1.79769313486231e+308 ~ -2.22507385850721e-308, 0 and 2.22507385850721e−308 ~ 1.79769313486231e+308 A decimal is represented by a 64-bit value (the default range is 0.0)	-	-	-	-
ARRAY	Array data type When a variable is declared, the length of an array and the data type which is to be stored are	✓	✓	✓	✓

Data Type	Description	AS Controller		DVP Drive	
		Program	FB	Program	FB
	specified (max. length of Array is 512)				
STRING	String data type An ASCII code is represented by an 8-bit value. When a variable is declared, the length of the string is specified (A String is composed of 255 ASCII codes at most)	✓	✓	✓	✓
STEP	It is used as a recognition flag for a step	✓	✓	✓	✓
Function Block	It represents the name of a function block	✓	✓	✓	✓
COUNTER	Counter data type It represents a Counter	✓	-	✓	-
TIMER	Timer It represents a Timer	✓	-	✓	-
POINTER	It is a pointer for a word address	-	✓	-	✓
T_POINTER	It is a pointer for a Timer	-	✓	-	✓
C_POINTER	It is a pointer for a Counter	-	✓	-	✓
HC_POINTER	It is a pointer for a High-speed Counter	-	✓	-	✓
TIME	Time Format T#XXXXXXdXXhXXmXXsXXX.XXXXXXm, Unit: ms. Display Range: T#0ms~213503d23h34m33s709ms	-	-	-	-
DATE	Time Format D#Y-M-D. Display Range: D#1970-01-01~D#2106-02-07. Unit: sec.	-	-	-	-

Data Type	Description	AS Controller		DVP Drive	
		Program	FB	Program	FB
TOD	Time Format. TOD#Hr:Min:Sec.ms Display Range: TOD#00:00:00~23:59:50:59.00 . Unit: ms. If value=0, it displays TOD#00:00:00; If value=1, it displays TOD#00:00:00.001; If value=86399999, it displays TOD#23:59:59.999; If value=86400000, it displays TOD#00:00:00; If value=4294967295 it displays TOD#17:2:47.295	-	-	-	-
DT	Time Format D#Y-M-D-Hr-Min-Sec. Display Range: DT#1970-01-01-0:0:0~2106-02-07-6:28:15. Unit: s.	-	-	-	-

Assigning Address to a Variable and Setting the Initial Value of a Variable

A device is assigned to a variable according to the data type of the variable. You can set the initial value of a variable. If a project is downloaded, the initial values of the variables can be written in to the devices assigned to the variables.

The principle of assigning devices to variables are described in the following table. Different models have different assignment principle.

Model	Assignment Principle
General Rule	• Assign devices to the global variables and the local variables declared in the POUs of the program type. The system can also automatically assign devices to the global variables and the local variables declared in the POUs of the program type. • The system automatically assigns the devices to the local variables declared in the function blocks. You cannot assign devices to the local variables declared in the function blocks by themselves. • The devices assigned by the system are memories reserved by the CPU module and are not data registers or auxiliary relays. If the data type of a variable is STEP, TIMER or COUNTER, the system will automatically assign a stepping relay, a timer or a counter to this variable.
AS Controller	• The representation of X / Y address is decimal. For example, X0 of X0~X63 represents a Word, and X0.0 represents 0th Bit of it.
DVP Drive	• The representation of X / Y address is Octal. For example, X0 of X0~X7 represents a bit.

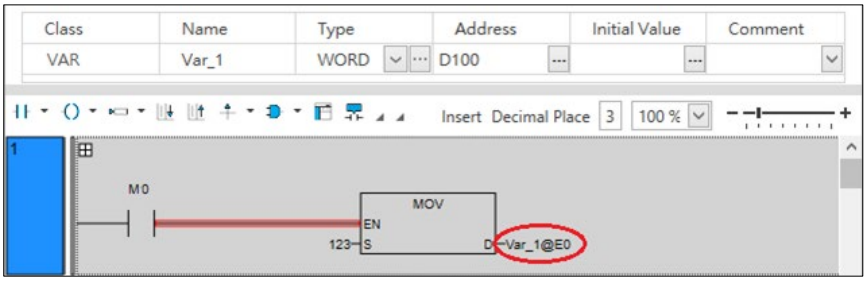
The relation between the data types and the device types which can be assigned is described in the following table.

Model	AS Controller		DVP Drive	
	Device assigned by user	Device assigned by system	Device assigned by user	Device assigned by system
BOOL	Contact M/SM/T/C/HC or bit in the D/X/Y/SR Register	Internal memory	Contact X/Y/M/SM/T/C/HC or bit in D/SR Register	Internal memory
WORD	D/X/Y/E/SR Register	Internal memory	D/E/SR Register	Internal memory
DWORD	D/X/Y/E/SR Register	Internal memory	D/E/SR Register	Internal memory
INT	D/X/Y/E/SR Register	Internal memory	D/E/SR Register	Internal memory
DINT	D/X/Y/SR Register	Internal memory	D/SR Register	Internal memory
REAL	D/X/Y/SR Register	Internal memory	D/SR Register	Internal memory
STRING	D/X/Y Register	Internal memory	D Register	Internal memory

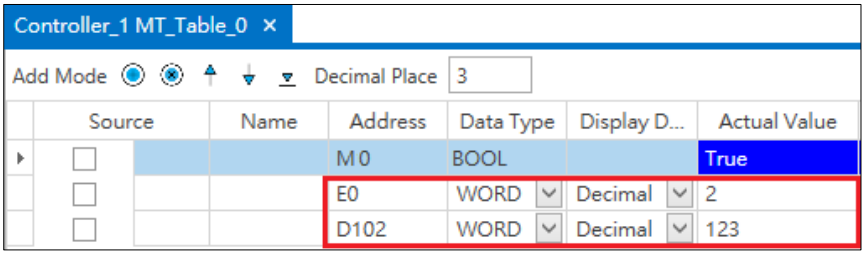
Model	AS Controller		DVP Drive	
	Device assigned by user	Device assigned by system	Device assigned by user	Device assigned by system
COUNTER	C/HC Register	C Register	C/HC Register	C Register
TIMER	T Register	T Register	T Register	T Register
ARRAY	The devices assigned to a variable, whose data type is ARRAY, depend on the array type specified. The array is composed of the devices starting from the device assigned by you or the system, and the number of devices in the array conforms to the size of the array. The devices assigned to the variable, whose data types is Array, cannot be SR/SM/E devices.			

Indirect Assignment and Modification for Register Symbol Variable

In DIADesigner, a variable can be modified by an index register. The format is Identifier@Index register. An index register can be an E register or a variable to which an E register is assigned.

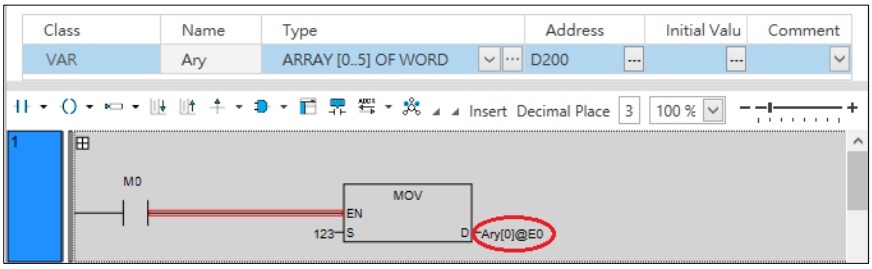


The data stored in an index register indicates the offset for an object modified. Take the above figure for example. If the value in the index register E0 is 2, VAR_1@E0 indicates that 2 is added to the device address (D100) assigned to VAR_1, that is, VAR_1@E0 represents D102. When M0 is ON, the value 123 is transferred to the data register D102.



The same rule applies to a symbol whose data type is ARRAY. The format is Identifier[Index]@Index register. The index must be a constant. If the index is a

variable, the variable whose data type is ARRAY cannot be modified by the index register. In the following figure, Ary is an array composed of five elements, and the start device address is D200. When the system compiles the program, D200~D204 are assigned Ary. If the value in the index register E0 is 2, Ary[0]@E0 indicates that 2 is added to the device address (D200) assigned to Ary[0], that is, Ary[0]@E0 represents D202, the device address assigned to Ary[2]. If the value in the index register E0 is 6, Ary[0]@E0 represents D206. D206 is not within the range of devices assigned to the array, but this usage is legal. Since the data stored in an index register indicates the offset for an object modified, you must be more careful when modifying symbol whose data type is ARRAY with an index register. Refer to the following sections for more information about the usage of arrays.



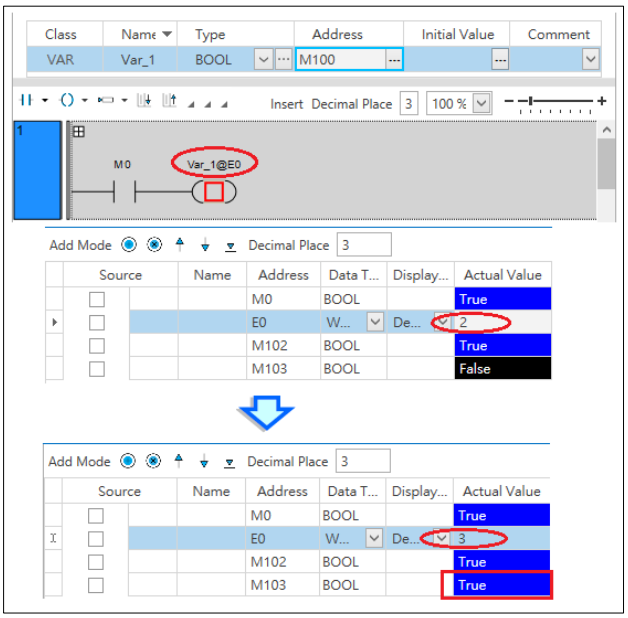
Additional remark

a. The data stored in an index register indicates the offset for a device modified. If the system automatically assigns a device to a symbol modified, the use of an index register will cause the program to be executed incorrectly because you do not know which device is assigned to the symbol.

b. If the addition of the value in an index register to a device address produces a device address which is not within the range, the PLC will operate incorrectly. As a result, you must be more careful when modifying a symbol with an index register. Take AHCPU530-EN for example. If the value in E0 is larger than 35, the operand D65500@E0 represents a device which is not within the range of data registers, and an error occurs in the CPU module.

c. If the value in an index register is changed, the register which operates differs from the original register. As a result, if the original register is not used in the program, the final value in the original register is retained. Take AHCPU530-EN for example. In the following figure, if the value in E0 is 2. M102 is ON when M0 is ON. If the value in E0 is changed from 2 to 3, the device which operates becomes

M103. M103 is ON when M0 is ON. Since M102 is not used in the program, M102 remains ON.



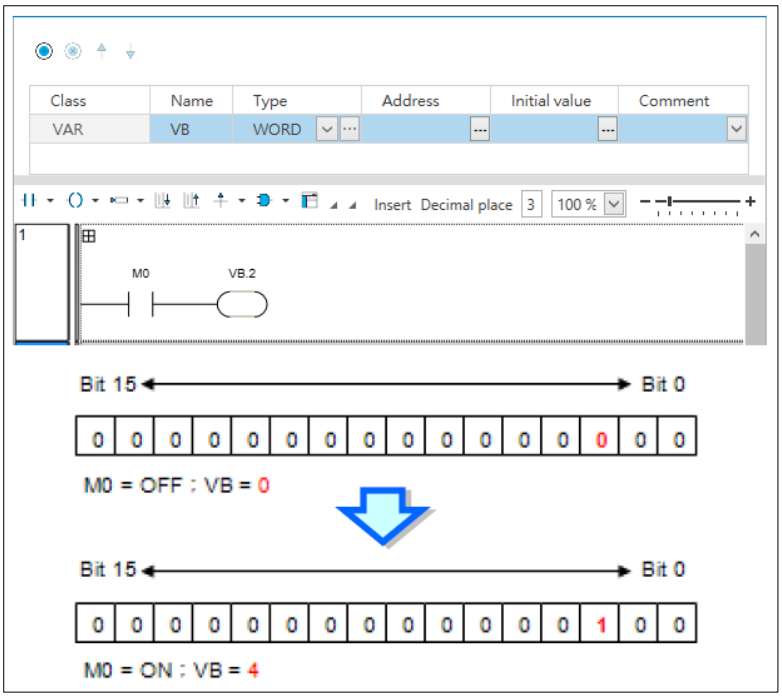
d. An index register cannot be modified by another index register. For example, E0@E1 is not allowed. As a result, before a symbol is modified by an index register, make sure that the device assigned to the symbol is not an E register.

Bit in a Device Represented by a Variable

An AS series CPU module supports the manipulation of the bits in a word device. The format is word register.bit number. For example, D0.2 indicates the bit 2 in D0 is manipulated. The same rule applies to the symbols. The format is Identifier.Bit number, e.g. VB.2. The rule also applies to the symbols whose data type is ARRAY. The format is Identifier[Index].Bit number, e.g. Ary[0].1.

The bit number must be a decimal constant. No matter what the data type of a symbol modified is, the bit number must be within the range between 0 and 15. Besides, the bits in a device represented by a symbol whose data type is BOOL, STEP, COUNTER, TIMER, a function block, HC_POINTER, C_POINTER, or T_POINTER cannot be manipulated.

In the following figure, the data type of VB is WORD. If the present value of VB is 0. When M0 is ON, the value of bit 2 in the device represented by VB is 1, that is, the value of VB is 4.



In an AH / AS series CPU module, an index register can be used with the manipulation of the bits in a device. If an index register is used with the manipulation of the bits in a device, the mark @ has high priority. Refer to the following examples. The data type of VB is WORD.

Example 1: VB.1@E0 (VB represents D100. The value in E0 is 3.)

1@E0 is interpreted first. As a result, VB.1@E0 is equivalent to VB.4. The bit which is actually manipulated is bit 4 in D100.

Example 2: VB@E0.1 (VB represents D100. The value in E0 is 3.)

VB@E0 is interpreted first. As a result, VB@E0.1 represents D103.1. The bit which is actually manipulated is bit 1 in D103.

Example 3: VB@E0.1@E1 (VB represents D100. The value in E0 is 3, and the value in E1 is 2.)

VB@E0 and 1@E1 are interpreted separately. As a result, VB@E0 represents D103, and 1@E1 represents 3. VB@E0.1@E1 represents D103.3. The bit which is actually manipulated is bit 3 in D103.

8.2.2.3 User-defined Data Type / Data Unit Type

Sometimes, basic data types are not enough for you to write your own programs for the PLC. In the IEC-61131-3 standard, a high-level language such as the user-

defined data type is adopted to create a new data type as required, allowing users to define their own variables and improving the readability and the efficiency of developing a program.

Structure

Definition of a Structure

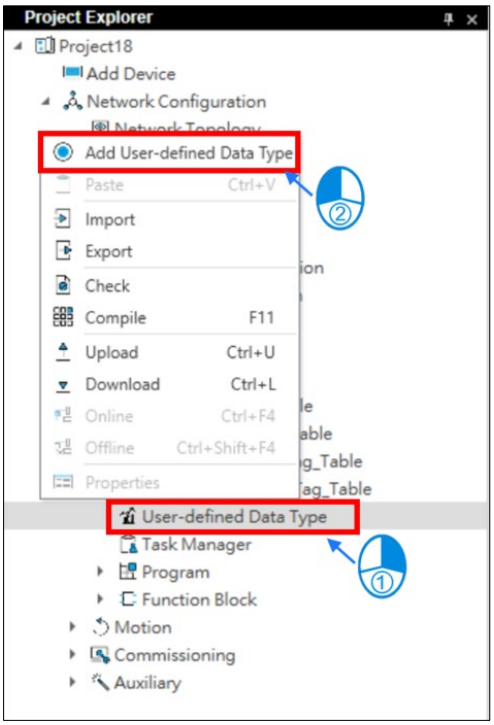
A Structure is a collection of data, consisting of same number of various data types. Structured or array elements can be elements for other Structures. A Structure is used for grouping data and can be seen as a data unit to transfer parameters.

Creating a Structure

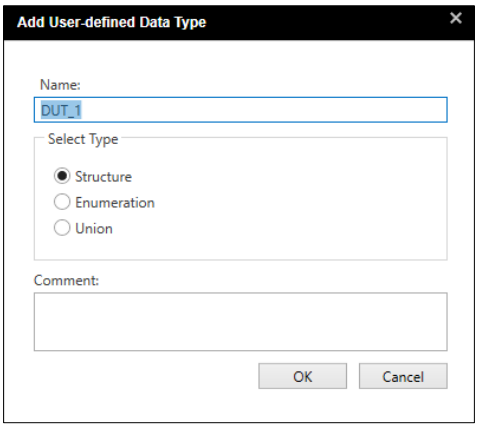
Follow these steps to create a Structure in DIADesigner.

To create Structure in DIADesigner

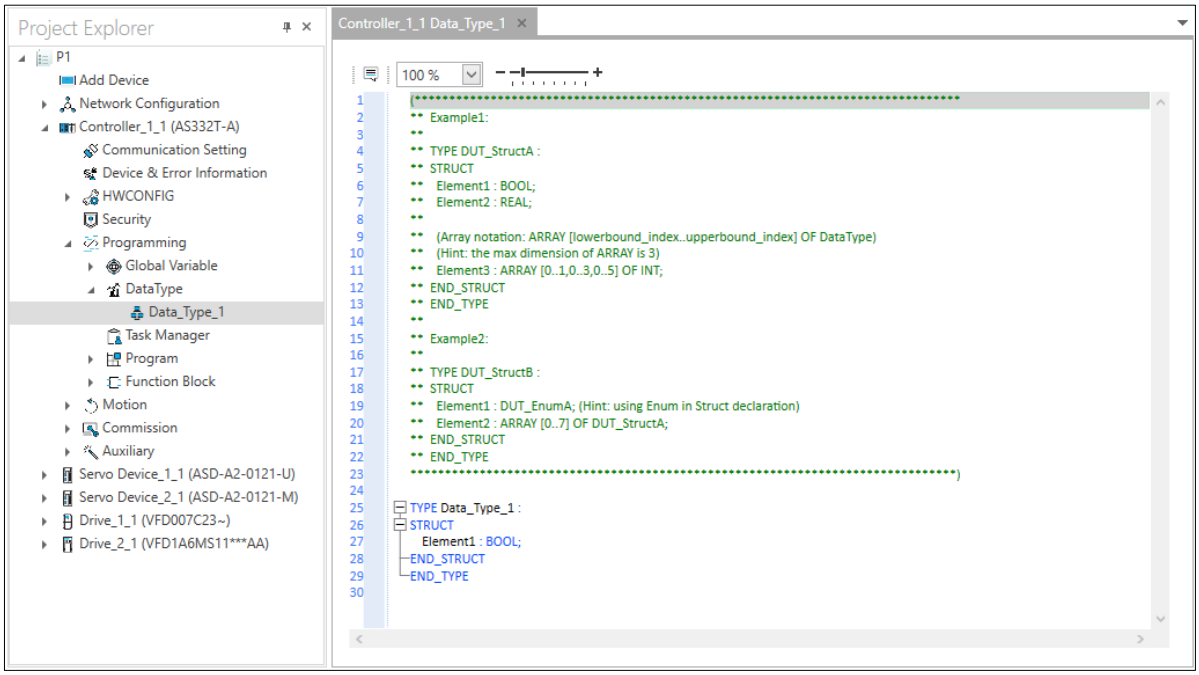
1. Right-click **User-defined Data Type** option in **Project Explorer** to open its context menu.
2. Select **Add User-defined Data Type** in the context menu.



3. Enter a **Name** for the Structure DUT, select **Structure** in **Select Type**, enter **Comment** (optional) for DUT and click **OK**.



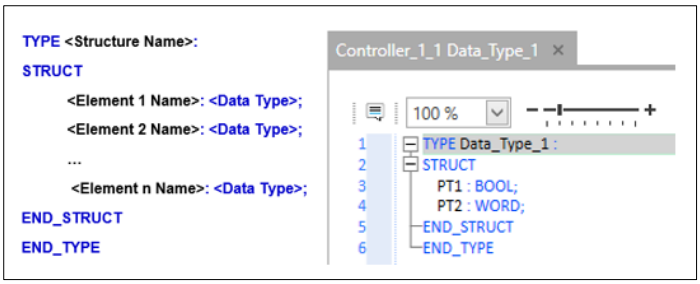
After a Structure DUT is created, open **User-defined Data Type** dialog by double-clicking the DUT created. **User-defined Data Type** dialog is shown with examples and definitions for reference in the following figure. You can edit the examples to make them as their own structures.



The Structure format is:

- Words in BLUE color are the system keywords
- Words in BLACK color are the user-defined structure names and element names.

A sample Structure is shown in the following figure.



Definitions

Structure Name: Enter the Structure Name here. It is recommended to use the same name as the name in **Project Explorer**.

Element Name: Define the Element Name.

Data Type: Define the Data Type of the Elements such as BOOL, WORD, DWORD, LWORD, INT, DINT, LINT, REAL, LREAL, ARRAY or another Structure or Enumeration or UNION.

When the element data type is ARRAY, the format is –

ARRAY [0..n,0..n,0..n**] OF <data type>;**

Note: Maximum value of n is 65535.

If the element data type is Structure, write another Structure Name. But if the data type is ARRAY, you cannot import ARRAY again. DIADesigner supports multi-dimensional Array, up to 3-dimensions.

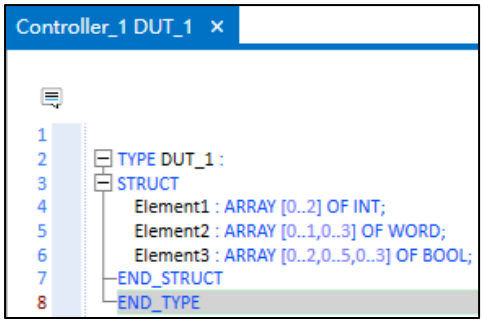
The format of a two-dimensional array is –

ARRAY [0..n**] [**0..n**] OF <data type>;**

The format of a three-dimensional array is –

ARRAY [0..n**] [**0..n**] [**0..n**] OF <data type>;**

An example of ARRAY is shown in the following figure.

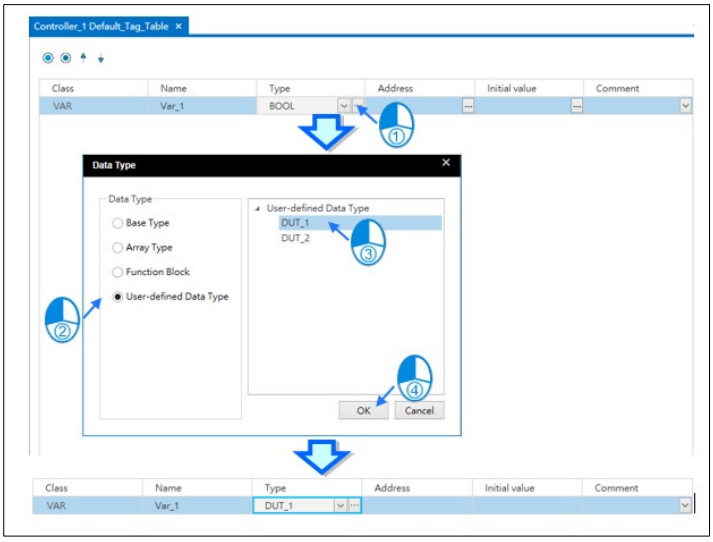


The naming rules of a Structure and an element are stated as follows.

- An identifier cannot be the same as names of the user-defined data types. The repetition of a Structure identifier is not allowed.
- The repetition of an element identifier in the same structure is not allowed.
- The identifiers of a Structure and Element can be composed of 30 characters at most and a Chinese character takes two characters.
- Keywords (names reserved for the system) are not allowed as names. Example, an instruction code, a device name, or a name given a special significance. However, if a name reserved by the system is a part of the names of a Structure or an Element, it is valid. For example, M0 is not a valid name, but _M0 is a valid name.
- The identifiers of a Structure and an Element cannot contain spaces. For example, INPUT CH0 is not a valid name.
- The identifiers of a Structure and an Element can contain underline, but it is not allowed to use 2 underlines continuously. Underlines cannot appear at the end even. For example, INPUT_CH0 is valid, but INPUT__CH0 and INPUT_CH0_ are not valid.
- The identifier of a Structure and an Element cannot contain special characters, including *, #, ?, \, %, @ etc.
- The initialization value can be declared in a Structure, for example, “Element1: WORD:=10;”.

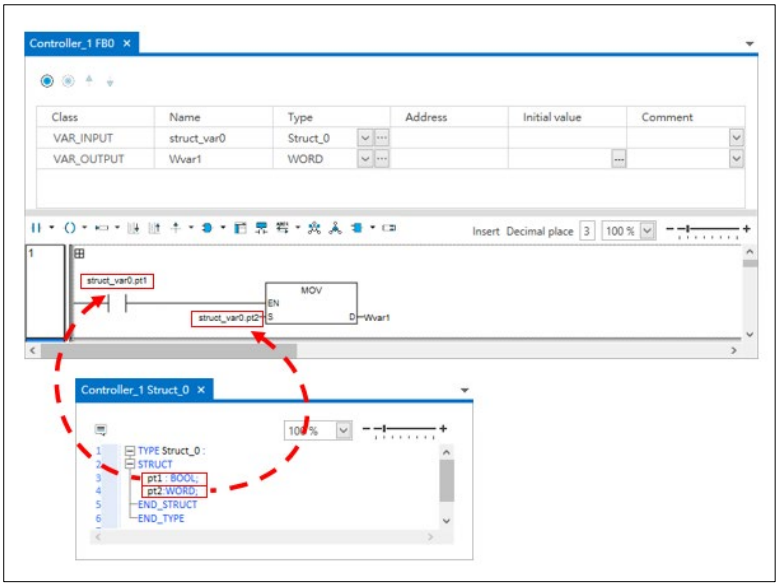
Using a Structure Variable

After defining a Structure DUT, create the variables in the Global variable table or local variable table or in function block variable table. Create a Variable and select the respective DUT type created as per requirement.

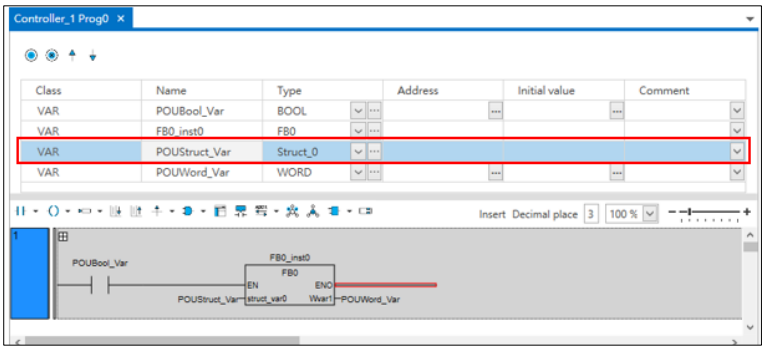


Application of Structures

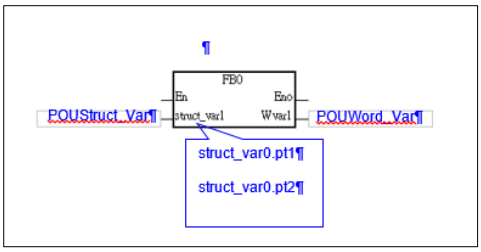
After the Structures and Variables are declared, they can be used in the POU. The data type of the Variable struct_var0 is Struct_0 in the function block as shown in the following figure. You can use “.” to assign the element for the structured variables. For example, struct_var0.pt1 means the first element pt1 is assigned as a contact to the data type BOOL operand and struct_var0.pt2 means the second element pt2 is assigned as a pin to the data type WORD operand as shown in the following figure.



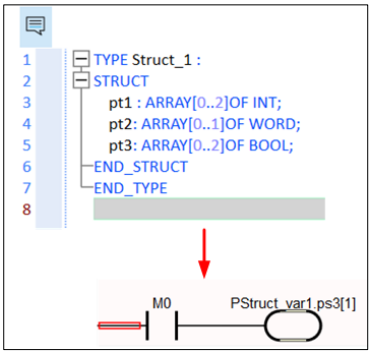
When using the aforementioned function block, Struct_0 data type variable should be declared in the POU, so that the Struct_0 data type input pin and output pin can be consistent. The elements with the same structure can use "." to assign the element for the structure type variables, for example POUStruct_var is shown in the following figure.



From the description above, we know that when the structure type variable POUStruct_Var is used in the FB0, it also delivers all the elements from the variable for struct_var0.pt1 and struct_var0.pt2 to operate. Thus, you can define the structure type variables as the input/output pin for the function block to deliver all the elements from the variable. So that the pin defining time can be saved and the program can be simpler.

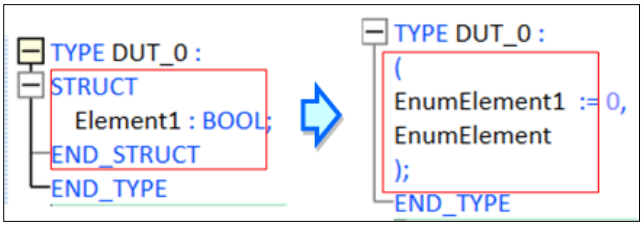


When the structure is constructed with an array element, the variables in the structure can be written as shown in the following figure. The status of M0 can be written in the [1] of the appointed element pt3 in the variable Pstruct_var1.



Change Structure to Enumeration

Change a DUT from Structure to Enumeration by editing the structure names of STRUCT and END STRUCT to “ (“ and “) ; ” as well as the element names from STRUCT to ENUM format.



Enumeration

Definition of Enumeration

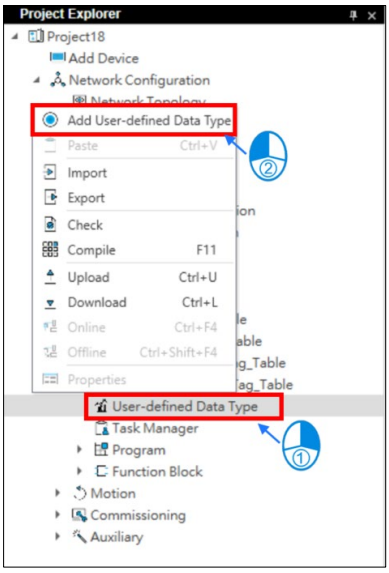
Enumerate (obtain one at a time) the elements in a collection of objects, providing an efficient way to define a set of named integral constants that may be assigned to a variable and the values that the enumeration represents.

Creating an Enumeration

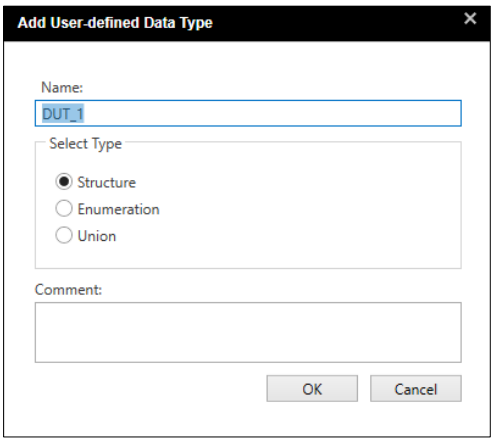
Follow these steps to create an Enumeration in DIADesigner.

To create Enumeration in DIADesigner

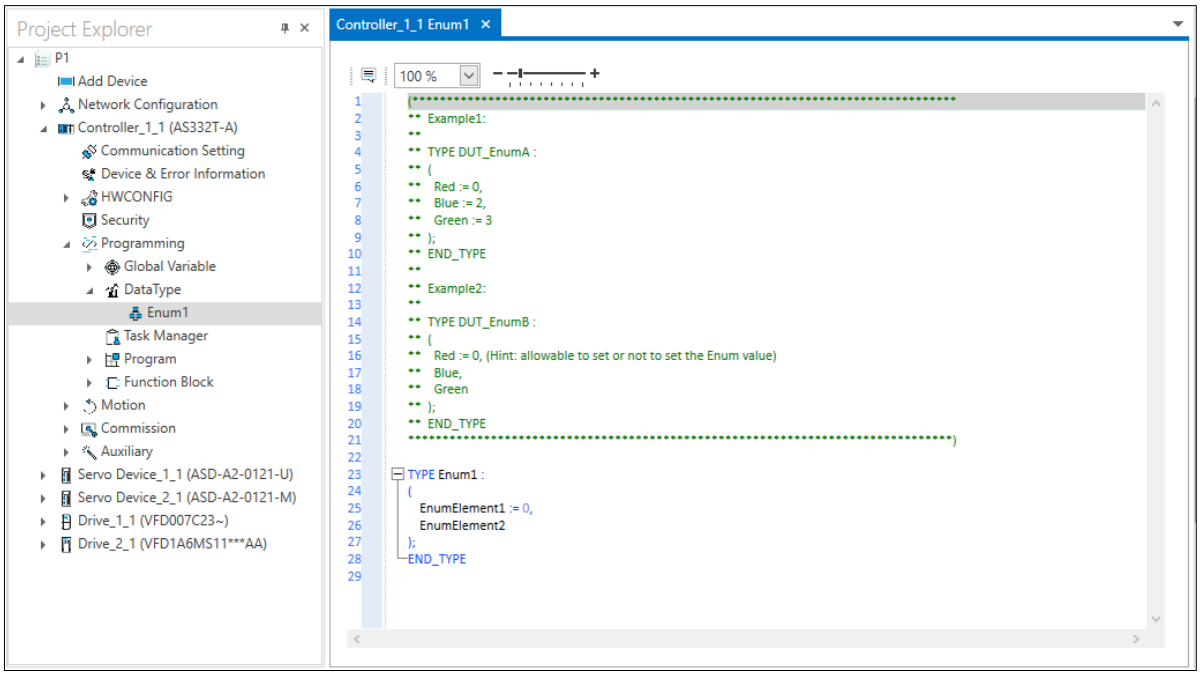
1. Right-click **User-defined Data Type** option in **Project Explorer** to open its context menu.
2. Click **Add User-defined Data Type** option in the context menu.



3. Enter a **Name** for the Enumeration DUT, select **Enumeration** in **Select Type**, enter **Comment** (optional) for DUT and click **OK**.



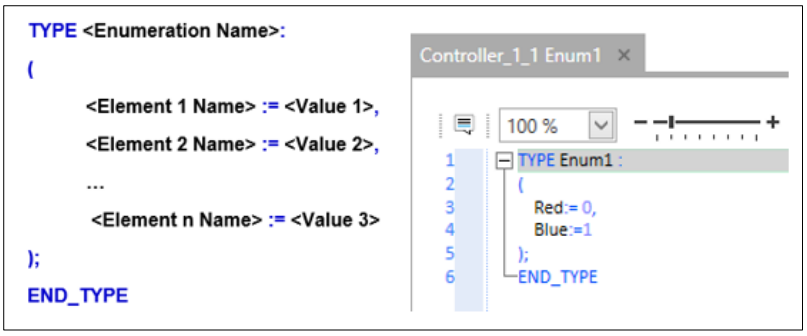
After an Enumeration DUT is created, open the **Data Type** dialog by double-clicking the DUT created. **Data Type** dialog with examples and definitions for reference is shown in the following figure. You can edit the examples to make them as their own enumerations.



The Enumeration format is:

- Words in BLUE color are the system keywords
- Words in BLACK color are the user-defined enumeration names, element names and values.

A sample enumeration is shown in the following figure.

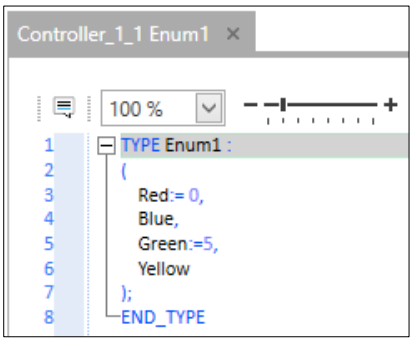


Definitions

Enumeration Name: Enter the Enumeration Name here. It is recommended to use the same name as the name in **Project Explorer**.

Element Name: Define the Element Name.

Value: Define values for every element, ranging from -32768 ~ 32767. It is acceptable to leave this field empty (blank). The system will use defaults according to the pervious element value and avoid duplications. The system will set the value of the element Blue to 1 and Yellow to 6 in the example in following figure.



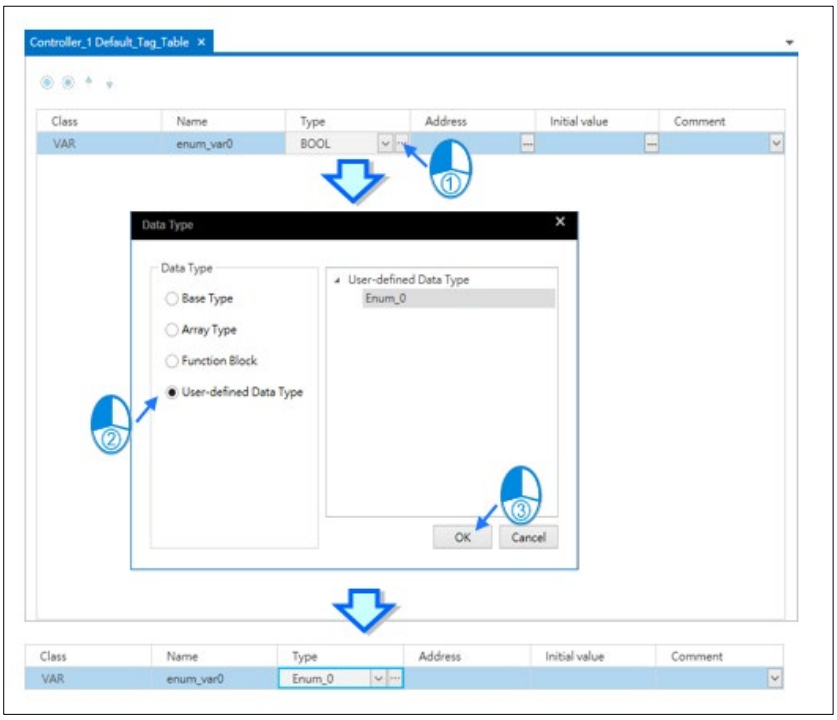
The naming rules of an Enumeration and an element are stated as follows.

- An identifier cannot be the same as names of the user-defined data types. The repetition of an Enumeration identifier is not allowed.
- The repetition of an element identifier in the same Enumeration is not allowed.
- The identifiers of an Enumeration and Element can be composed of 30 characters at most and a Chinese character takes two characters.
- Keywords (names reserved for the system) are not allowed as names. Example, an instruction code, a device name, or a name given a special significance. However, if a name reserved by the system is a part of the names of an Enumeration or an Element, it is valid. For example, M0 is not a valid name, but _M0 is a valid name.

- The identifiers of an Enumeration and an Element cannot contain spaces. For example, INPUT CH0 is not a valid name.
- The identifiers of an Enumeration and an Element can contain underline, but it is not allowed to use 2 underlines continuously. Underlines cannot appear at the end even. For example, INPUT_CH0 is valid, but INPUT__CH0 and INPUT_CH0_ are not valid.
- The identifier of an Enumeration and an Element cannot contain special characters, including *, #, ?, \, %, @ etc.
- The data size of Enumeration variables is the same as INT data size.

Using an Enumeration Variable

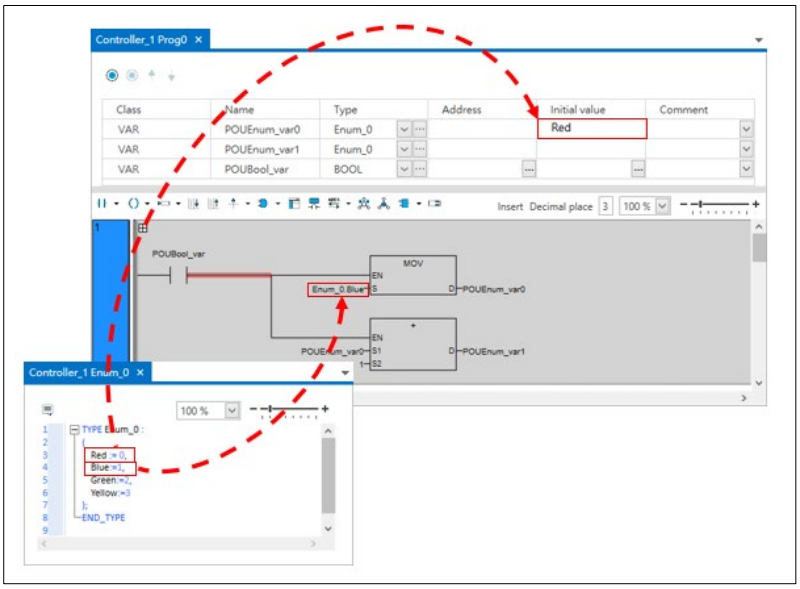
After defining an Enumeration DUT, create the variables in the Global variable table or local variable table or in function block variable table. Create a variable and select the respective DUT type created as per requirement.



Applications of Enumerations

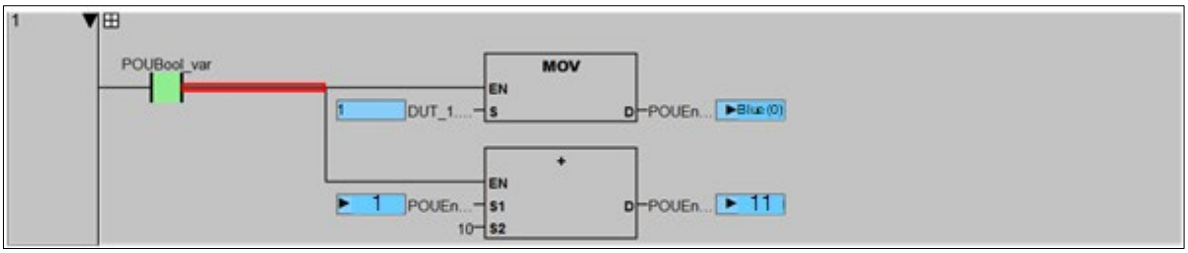
After the Enumeration and variable are declared, they can be used in a POU. The datatype of the variable POUEnum_Var0 is Enum_0. The initial value can be set by the element name or values. The following figure shows the element name – Red.

Use "." to assign the element for the enumeration variables. For instance, you can assign Enum_0.Blue to POUenum_Var0, the variable Enum_0 and execute the instruction MOV; the defined element or value is the enumeration value and after the execution of MOV, the value of POUenum_var0 is 1.



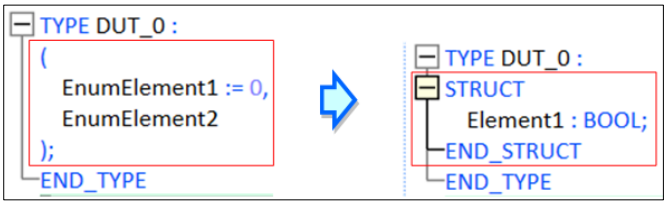
You can also perform numerical operations using the instruction + and set the value to 1, POUEnum_Var0 is incremented by 1 as the example shown in the above figure. When the value of POUEnum_Var0 is 1, after executing the instruction +, the value result is 2 and its element is Green as defined.

When running and monitoring the program, the enumeration variables will show their current defined element names. For example, when the value is 1, the POUEnum_Var0 is seen as Blue(1) as they are defined in the following figure. But if the value of the enumeration variable is not in the defined element range, the value is shown as it is; as the following example shows when the value is 11, but 11 is not in the defined range, POUenum_Var1=11 is stated as shown in the following figure.



Change Enumeration to Structure

Change a DUT from Enumeration to Structure by editing “(“ and “)” to structure names of STRUCT and END STRUCT as well as the element names from ENUM to STRUCT format.



Register Resource Allocation Setting

When compiling is executed, the system will automatically assign variables with unspecified register addresses to the internal register addresses. External devices cannot access this internal register address. Therefore, if external HMI or device access variables are required, need to be allocated to D/M/T/C register address and register resource allocation setting function.

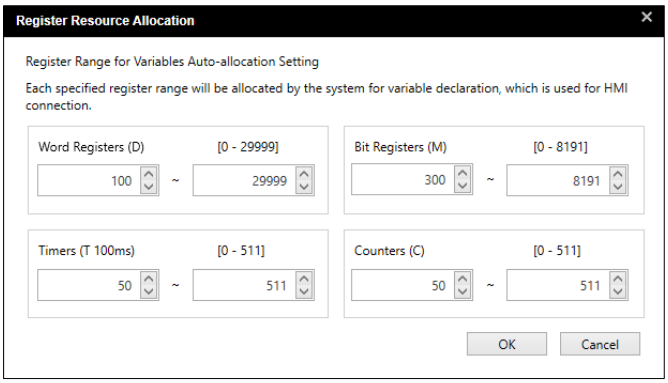
You can set the register range for automatic configuration in the project in advance, and different projects can have their own setting; in the global variable table, the automatic configuration address can be executed, and the system will only use this Devices within the range are configured, but if the number of variable symbols exceeds the set range, an error message indicating that the device resources are insufficient will be prompted.

For the description of the device address configuration, refer to the content of the address configuration and initial value of the variable symbols in this section.

Follow these steps to perform the **Register Resource Allocation** setting.

To perform Register Resource Allocation setting

1. Go to **Project Explorer** > **Controller Device** > **Auxiliary** > **Register Resource Allocation** to open **Register Resource Allocation** dialog.

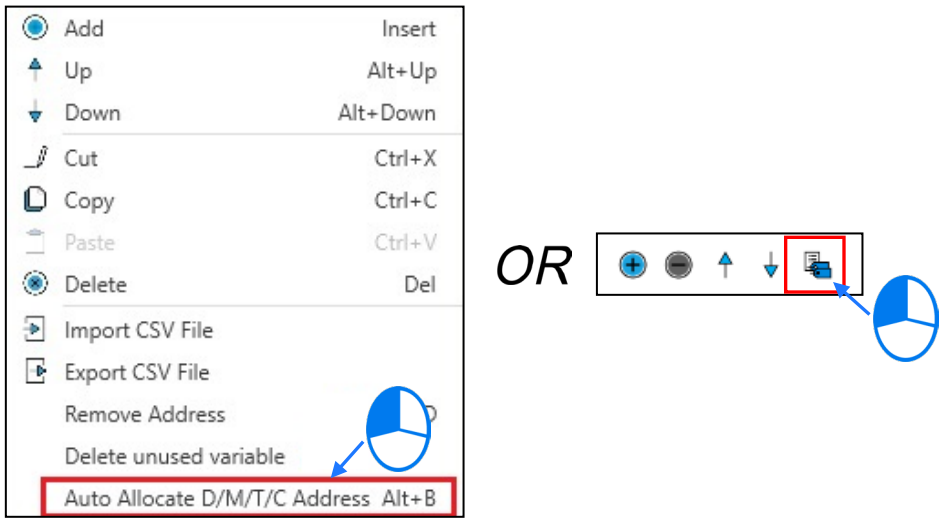


2. In the **Register Resource Allocation** dialog, set the automatic configuration range of the device (D/M/T/C).

3. Click **OK** to save the setting.

Note: To set the configuration range of various devices, various devices of different models have their own range restrictions. When entering an address that is not allowed, the system will prompt an error or automatically correct it.

Then, right-click any new variable in the global variable table, and select the **Auto Allocate D/M/T/C Address** in the right-click menu or click  on toolbar to automatically configure the variable device address, as shown in the following figure.



BOOL and ARRAY OF BOOL data types will be automatically configured in the M register; non-BOOL data types will be automatically configured in the D register; TIMER data types will be automatically configured in the T register; COUNTER data types will be automatically configured in the C register.

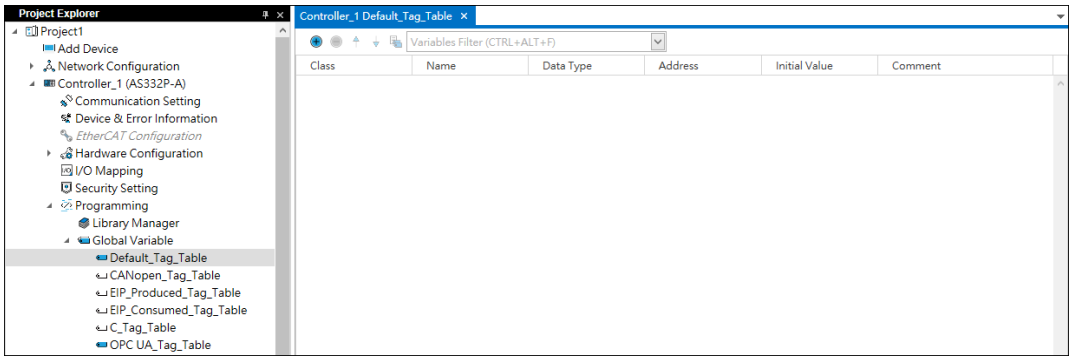
8.2.2.4 Variable Table

There are two types of variable tables in DIADesigner.

- Global variable table
- Local variable table

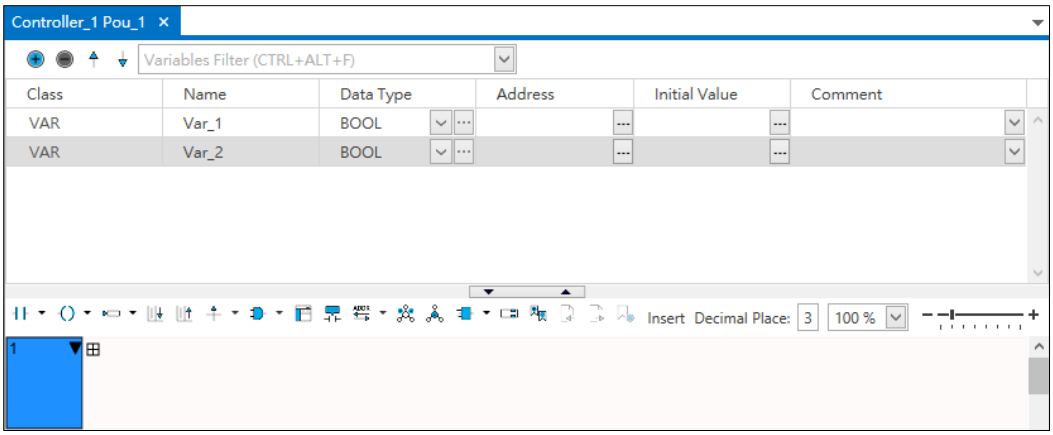
Global Variable Table

Double-click **Project Explorer** – **Global Variable** > **Default_Tag_Table** to open the Global variable table.



Local Variable Table

Local variable table in a POU is located at the top of the POU dialog.



Adding a Variable

Follow any of the following methods to add a variable in the **Global Default_Tag_Table** or in the POU local variable table.

- Click  on the toolbar of variable table.

  Or

- Right-click the variable table and click **Add**

  Or

- Press **Insert** key on the keyboard

  Or

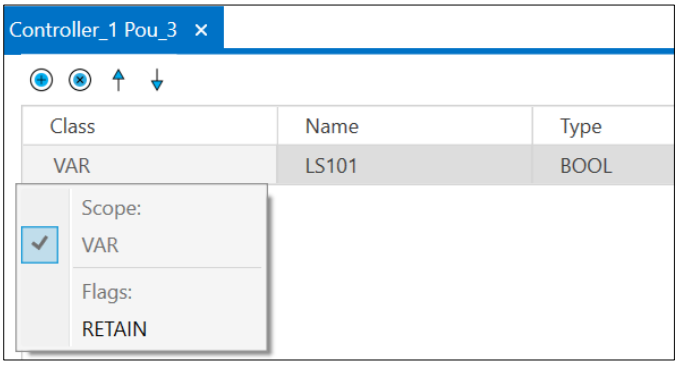
- Double-click the empty area of the variable table.

The following table lists the attributes of a Variable with their description.

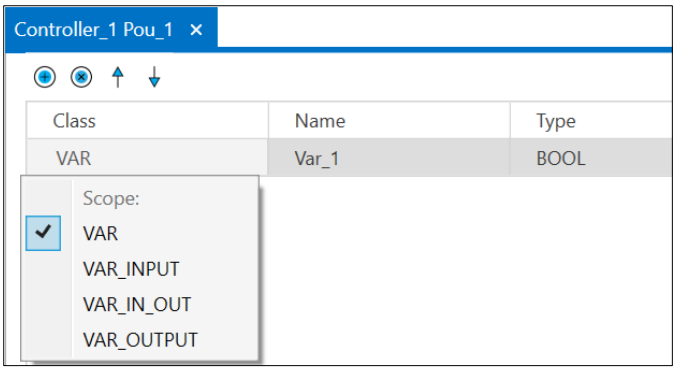
Function	Description
Class	Select the class of the variable
Name	Enter a name for the variable
Data Type	Select the datatype of the variable
Address	Select the address of the variable
Initial Value	Select or enter an initial value of the variable
Comment	Enter a comment for the variable

Variable Class

Variable Class drop-down list varies with the variable table created (Global / Local). The items in the Class drop-down list box for a global or a local variable in a POU are VAR and VAR_RETAIN.



The items in the Class drop-down list box for a POU of the function block type are VAR, VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT.

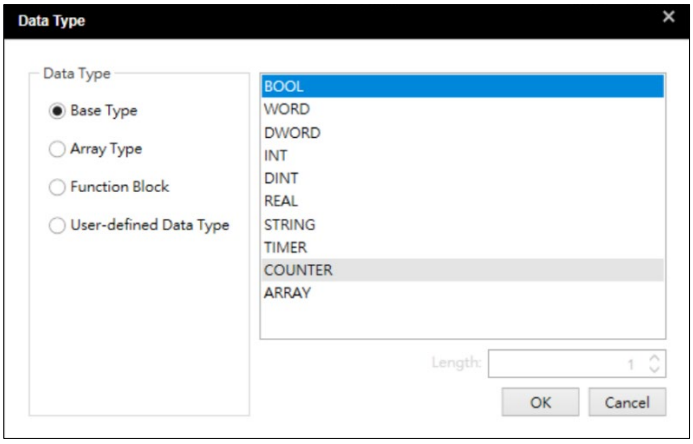


Variable Name

For DUT or Array variables, click ... in the **Name** attribute to expand the sub-item to edit the variable name.

Variable Type

Click ... in the **Type** attribute to display the **Data Type** dialog as shown in the following figure.



Select the **Data Type** of the variable needed as per requirement.

Variable Type – Base Type

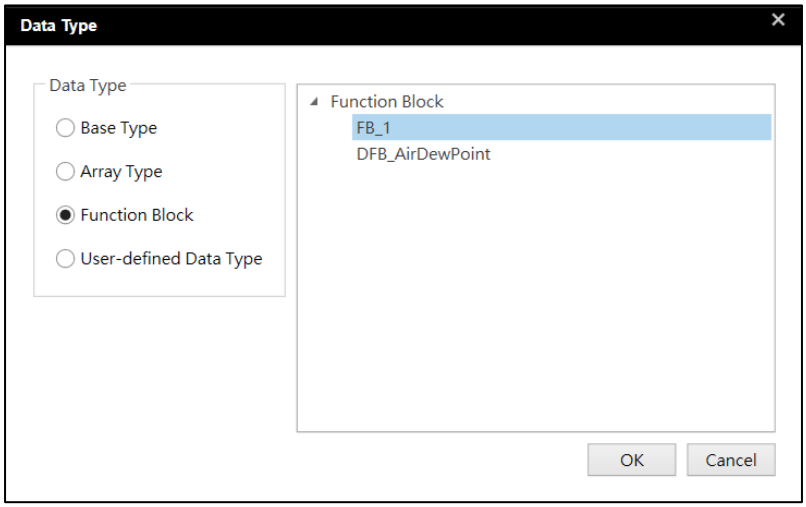
Base Type variables are:

- BOOL
- WORD
- DWORD
- INT

- DINT
- REAL
- STRING
- TIMER
- COUNTER
- ARRAY

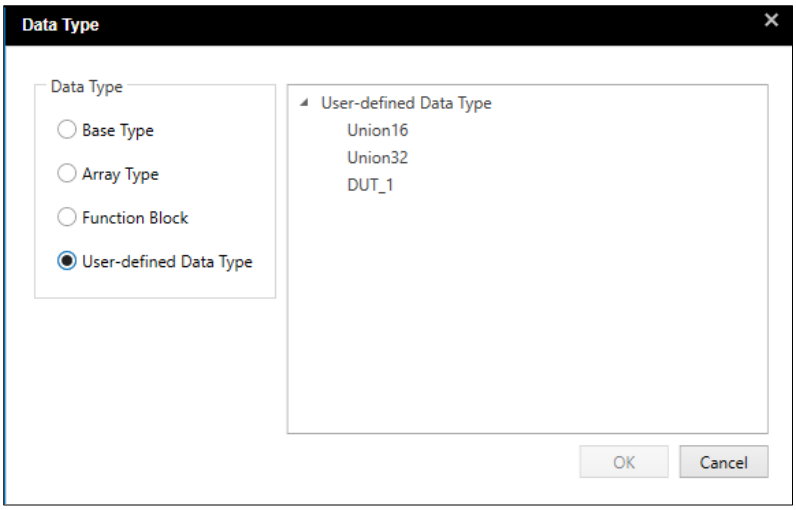
Variable Type – Function Block

Select **Function Block** option in the **Data Type** section and then select a user-defined function block or add the library function block in the **Function Block** section. If the **Data Type** of the variable is a function block, a **Function Block** entity will be generated.



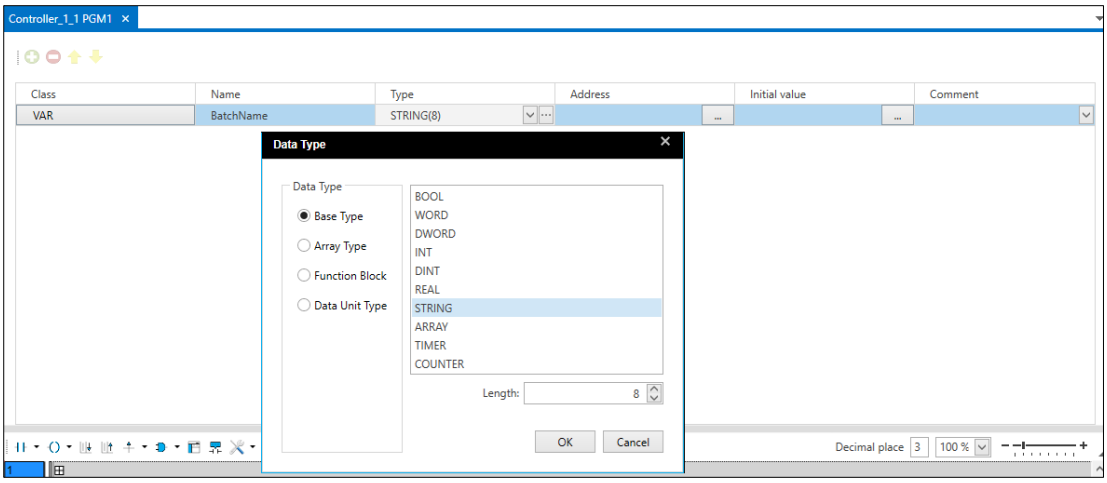
Variable Type – User-defined Data Type

Select the **User-defined Data Type** option in the **Data Type** section and then select a User-defined **Data Type** in the User-defined **Data Type** section.



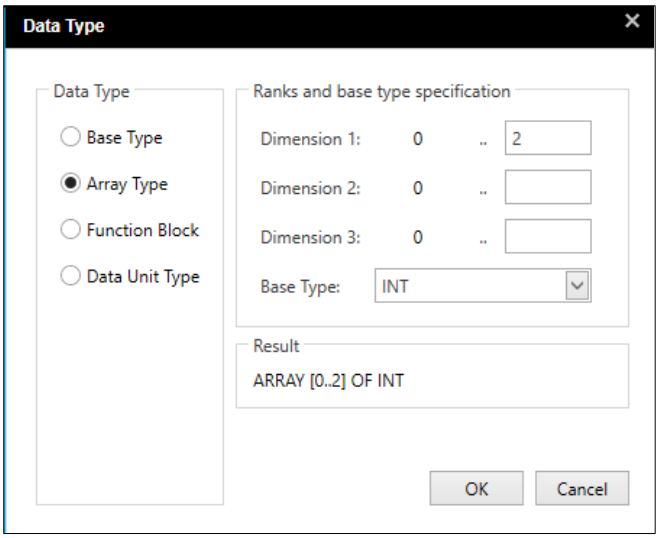
STRING

If the data type of a variable is String, set the size of the String. The size of a String is within the range between 1 character and 255 characters. The size of a String that you set is the maximum number of characters in the string. Click the increment/decrement buttons to increase or decrease the string size by 1. A String variable with size of 8 is shown in the following figure:



ARRAY

If the data type of a variable is ARRAY, select **Array Type** in the **Data Type** section. In the **Ranks and base type specification** section, at present you may select up to 3 dimensional arrays, set the data size [0..n] and the data type in **Base Type** result section displays the data type selected. An example of 1-dimensional array with INT datatype with data size 3 is shown in the following figure.



Address

Follow these steps to assign address to variables.

To assign address to variables

1. Enter the address directly in the field by keyboard entry.

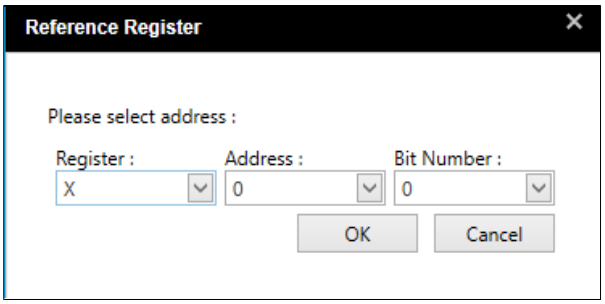
   Or

   Click  to open the **Reference Register** dialog.

2. Select **Register**, **Address** and **Bit Number**.

3. Click **OK**.

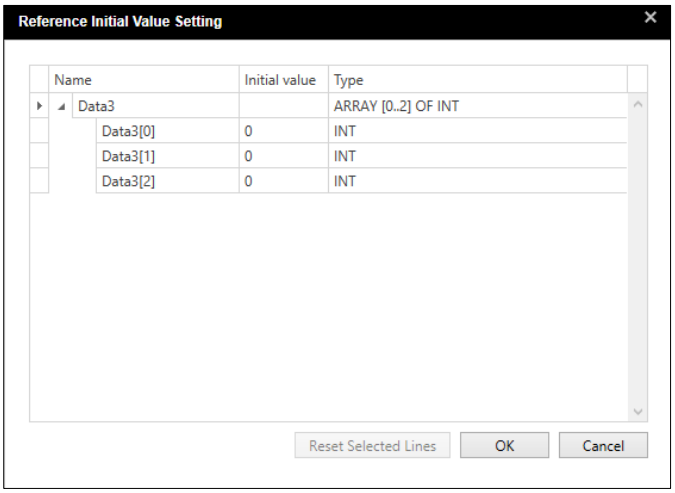
A sample address of X0.0 is shown in the following figure.



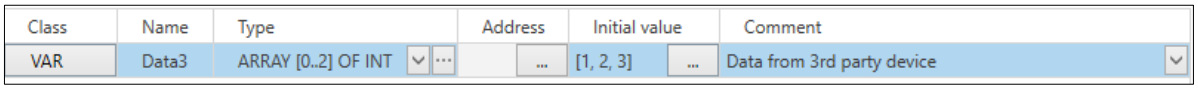
Initial Value

To set Initial Value for a variable, type the initial value directly. For variable of BOOL data type, it can be 0 or 1. For STRING data type, initial value cannot be more than maximum data size characters. Characters do not need to be put in double quotes. If

the data type of a variable is ARRAY, then upon clicking ... in **Initial value** field, the **Reference Initial Value Setting** dialog appears.

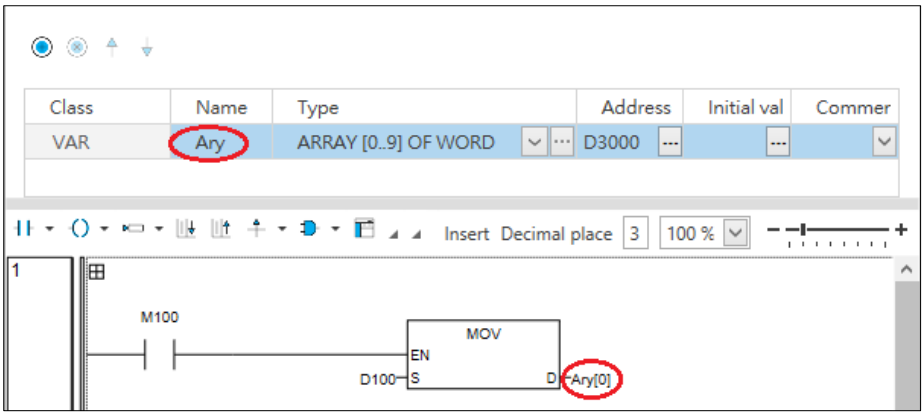


Comments: Enter comments for the variables by typing directly.



Variable whose data type is an ARRAY

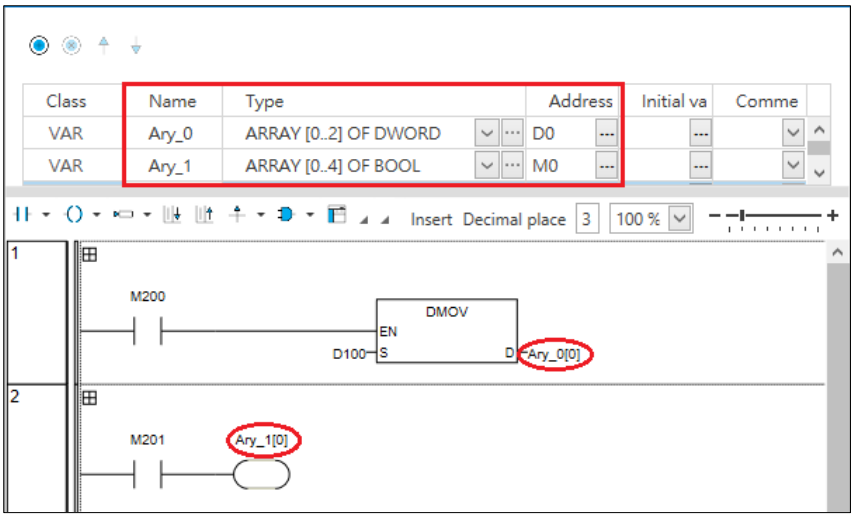
If you want to use a variable whose data type is ARRAY in a program, the expression format is **Identifier[Index]**. The index is a decimal constant. If the programming language in a project for a AS PLC CPU module is a Structured Text, the index can be a variable. The minimum index value must be 0, whether the index is a constant or a variable. The maximum index value cannot be larger than or equal to the number of elements in the array. For example, if the number of elements in an array is 10, the index value must be within the range 0 ~ 9. If the index is a variable, then the value of the variable cannot be larger than the number of elements in the ARRAY. Otherwise an error will occur during the operation.



If you declare a variable whose data type is an ARRAY, you must select an array type and set the size of the array. The number of elements in an array must be within the range 1 ~ 512. A start device address is assigned to a variable whose datatype is array according to the array type. Notice that the start device address is assigned to a variable whose data type is ARRAY cannot be an SR device, an SM device, an E device or and F device. Besides, an array composed of the devices starting from the device assigned by you or the system and the number of devices in the array conforms to the size of the ARRAY.

In the following figure, ARRAY[0..2] of DWORD in the Type… cell Ary_0 indicates that the array is composed of 3 elements and the array type is DWORD. The device address in the Address cell for Ary_0 indicates the start device address is D0. As a result, the array is composed of D0~D5 (6 WORD devices).

ARRAY[0..4] of BOOL in the Type…cell for Ary_1 indicates that the array is composed of 5 elements and the array type is BOOL. The Register address in the address cell for Ary_1 indicates that the start address is M0. As a result, the array is composed of M0~M4.

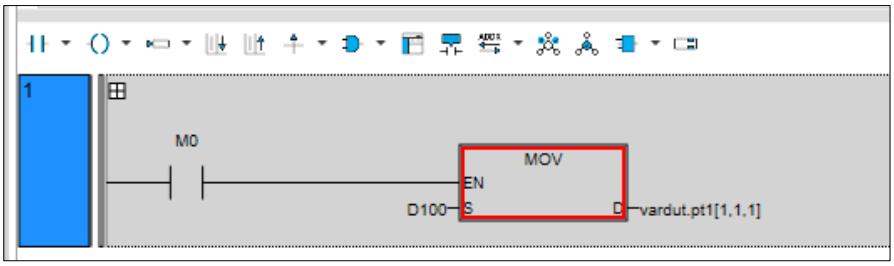


When declaring a variable whose data type is ARRAY, set the initial value of the variable.

Example: [1, 2, 3, 4, 5] in the **Initial Value** attribute for Ary_1 indicates that the initial value of Array elements.



When the element is with a two-dimensional ARRAY or three-dimensional ARRAY in the data unit type, the variable can use [n,n] or [n,n,n]. In the above example, the element pt1 is with a three-dimensional ARRAY, the data type of the variable can be vardut.pt1[1,1,1] to assign the element [1,1,1] from the ARRAY.



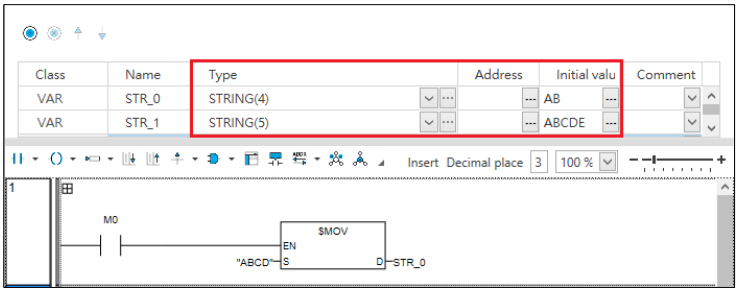
Variable whose Data Type is a STRING

In DIADesigner, a STRING is composed of ASCII codes and the ASCII codes are surrounded by double quotes. For example, “ABCD”, a variable whose data type is STRING is generally used with an applied instruction. Refer to the *Programming Manual* for more information.

AS series controller supports the variables whose data types are STRING and the data type of a variable of the VAR_INPUT class, the VAR_OUTPUT class or the VAR_IN_OUT class in a function block cannot be STRING. As a result, if you want to process a STRING in a function block, you can create the STRING as a global variable.

When declaring a variable whose data type is STRING, specify the maximum size of the STRING. The number of characters in a STRING is within the range 1 ~ 255 and one character occupies 1 byte. The number of devices assigned to the characters in a STRING must conform to the size of the STRING and one extra byte must be assigned to the ending character in the STRING. If the last character and the ending character is a STRING does not occupy the 2 bytes in a WORD device, the ending character is assigned to another device.

In the following figure, the number in the parentheses in the **Type** attribute for a variable whose data type is STRING indicates the maximum size of the STRING. Besides, the characters in the **Initial Value** attribute for a variable whose data type is STRING cannot be put in double quotes and the number of characters in the **Initial Value** cell for the variable whose data type is STRING must be less than or equal to the characters in the STRING.



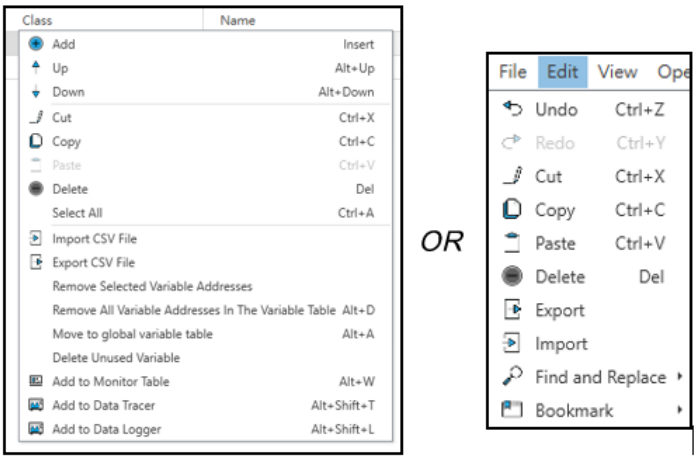
Modifying a Variable and Editing a Variable Table

To modify the attributes of a variable, edit the variable table directly. Variables can be modified in the following ways:

- Modify variables in the variable table with the help of context menu.

  Or

- Modify variables in **Edit** menu.



The following table describes the attributes in **Edit** Variables.

Item	Function
Undo	Undoing the previous action
Redo	Redoing the action which has been undone
Cut	Cut the variable selected
Copy	Copy the variable selected
Paste	Paste the variable which is cut/copied
Delete	Delete the selected variable
Select All	Select all variables.
Up	Move the variable one level up
Down	Move the variable one level down

Remove the Register Address Assigned to the Variables

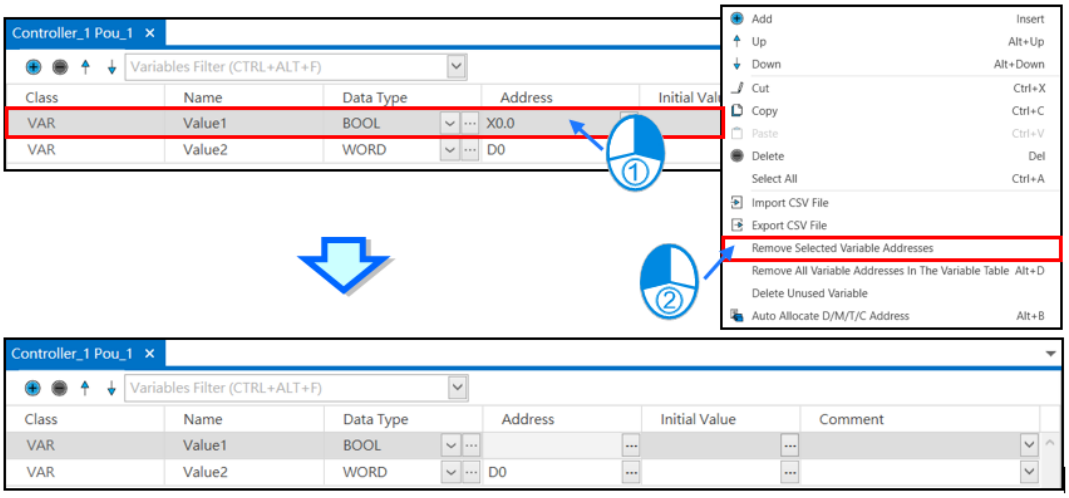
Delete the address of variable table, whether the register is assigned by you or by the system. After the program is compiled again, the system will automatically assign addresses to these variables. Besides, only the device addresses in the variable table selected are cleared, the device addresses in the other variable tables are not cleared.

Remove the selected variable addresses by following any of the below methods.

- Select one or more variables from the variable table.

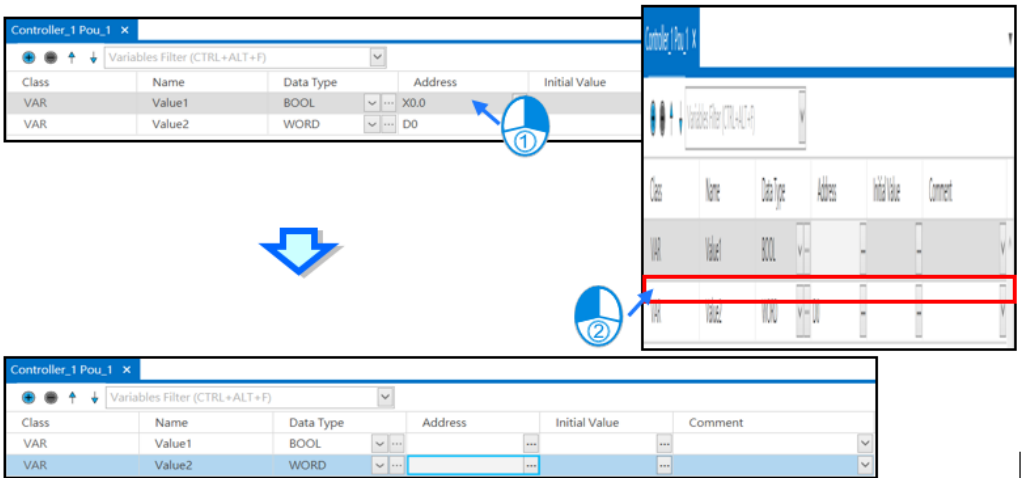
  Or

- Right-click to open the context menu and click **Remove Selected Variables Address**.



Use the following method to remove all variable addresses in a POU.

- Right-click the variable table and press **All Variable Addresses in the Variable Table**.



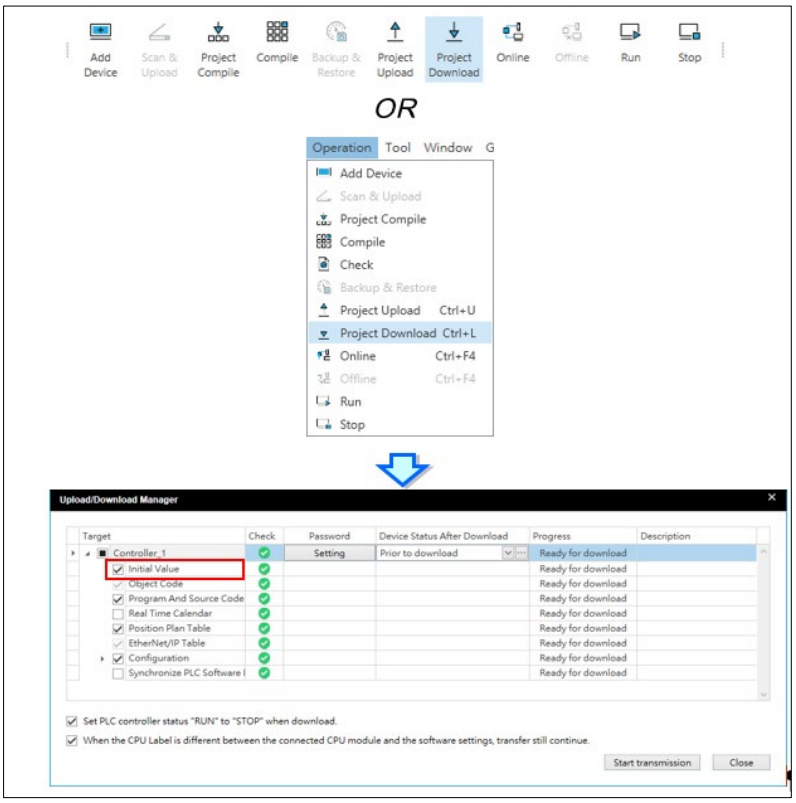
Downloading Initial Values of the Variables

Set the initial values of a variable when they declare it. If a project is downloaded, the initial values of the variables can be written into the devices assigned to the variables. However, the initial values of the variables in a project are written into the devices assigned to the variables only when the project is downloaded. If a controller is disconnected or stops running and then starts again, the values of the variables in the program in the controller will not be the initial values. To ensure that the values of the variables in a controller are the initial values whenever the controller begins to run, it is suggested that you should download the initial values of the variables. Besides, when the online edit is executed, the system does not download the initial values of the variables.

Follow these steps to write the initial value of variable to a controller while downloading a program.

To write initial value of variable to controller

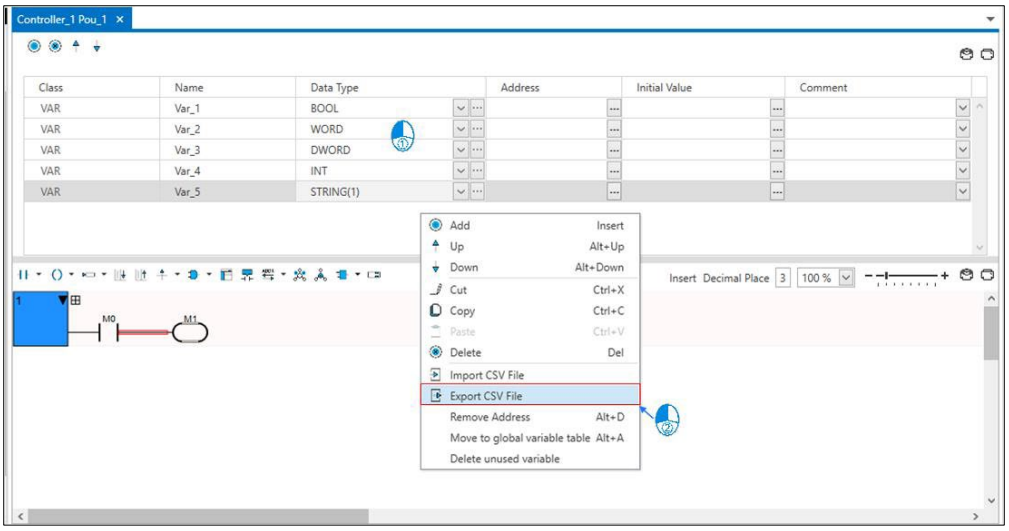
1. Click **Project Download** after the program is compiled without error.
2. Select **Initial Value** feature.
3. Click **Start Transmission**.



Exporting Variable Table

The variables created in a project can be exported as CSV files, which can be opened in Excel. You can edit / manage the exported data.

To export variables, right-click the variable table, and then select **Export CSV file** in the context menu.



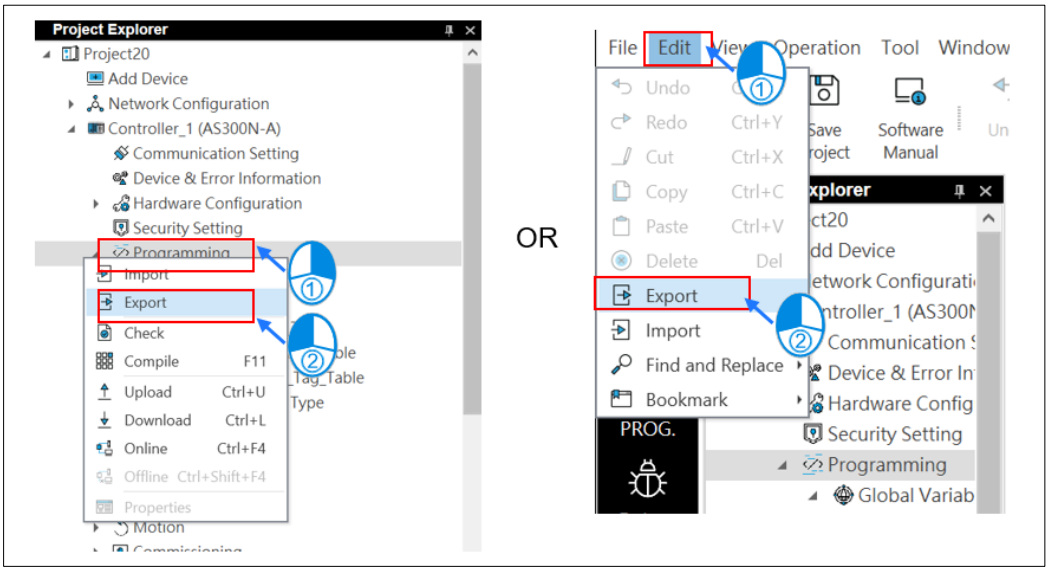
There are two methods to export global variable table.

1. Menu bar

   - Click **Edit** > **Export** to export the global variable table of the project that is currently editing.

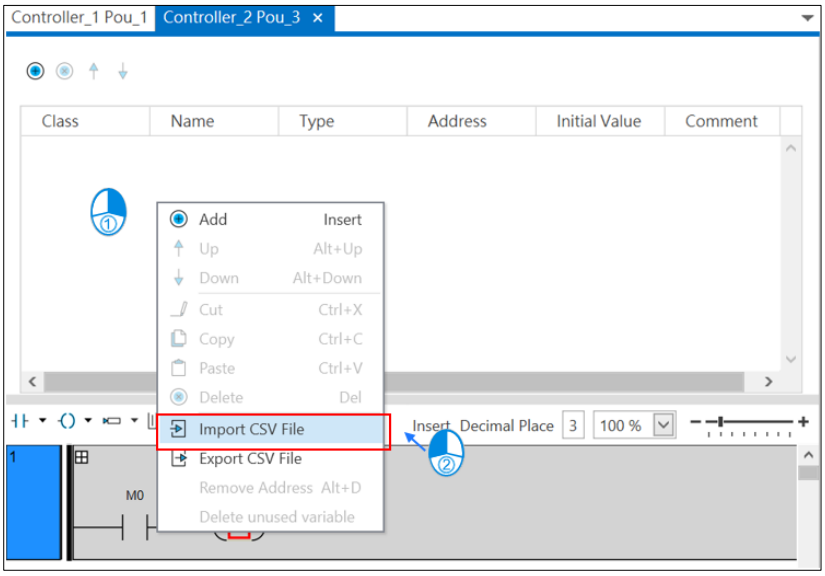
2. Project Explorer

   - Right-click the project node that is currently editing and select **Export** in the context menu.



Importing Variable Table

To import variables, right-click the variable table, and then select **Import CSV file** in the context menu.



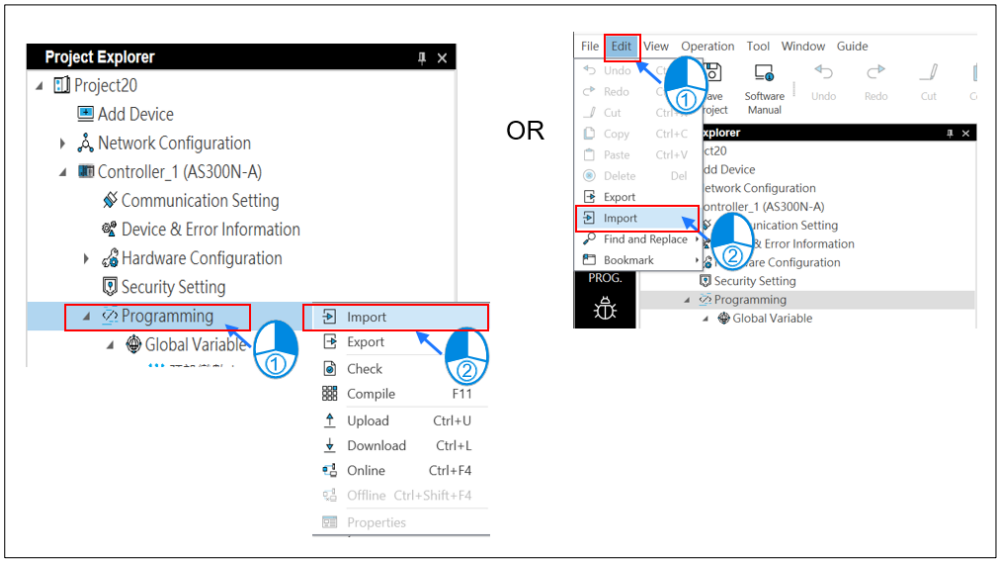
There are two methods to import global variable table.

1. Menu bar

   - Click **Edit** > **Import** to import the global variable table of the project that is currently editing.

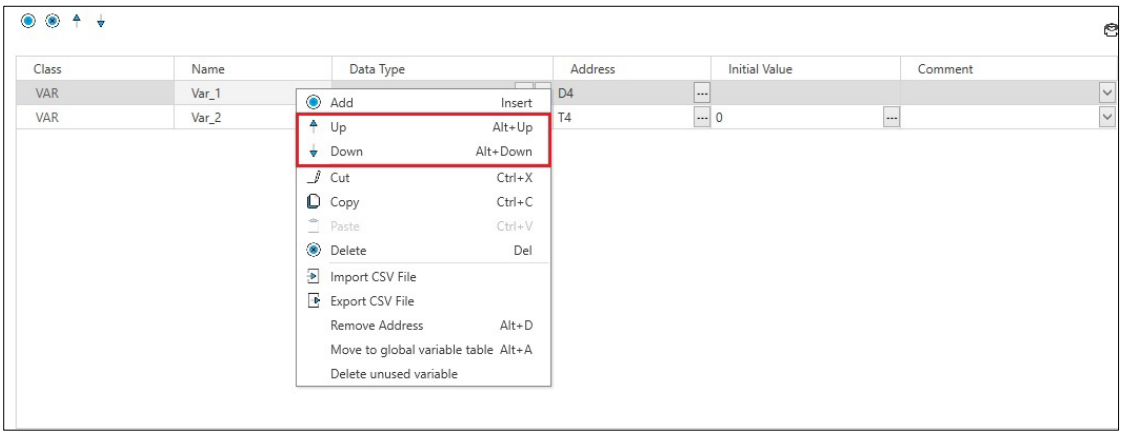
2. Project Explorer

   - Right-click the project node that is currently editing and select **Import** in the context menu.

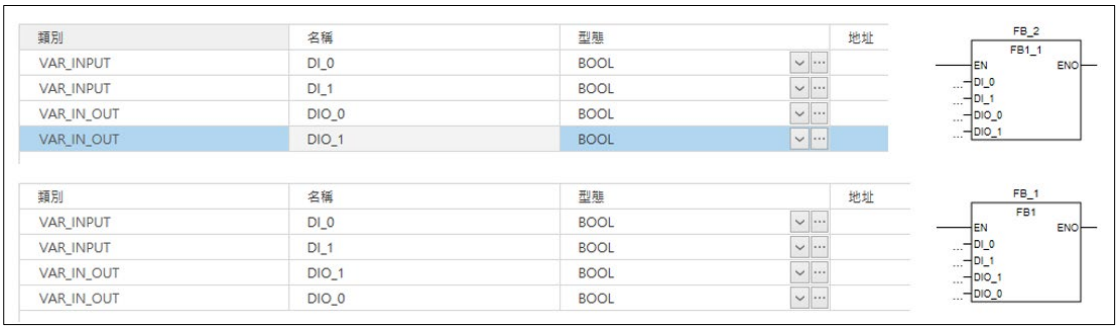


Arranging Variables

In the local variable table in a function block, the order in which the variables of the VAR_INPUT class, the variables of VAR_IN_OUT class and the variables of the VAR_OUTPUT class are arranged affects the order in which the operands in the function block are arranged. To arrange the variables in the local variable table in a function block, right-click a variable or use the Up/Down button and then move it up or down.

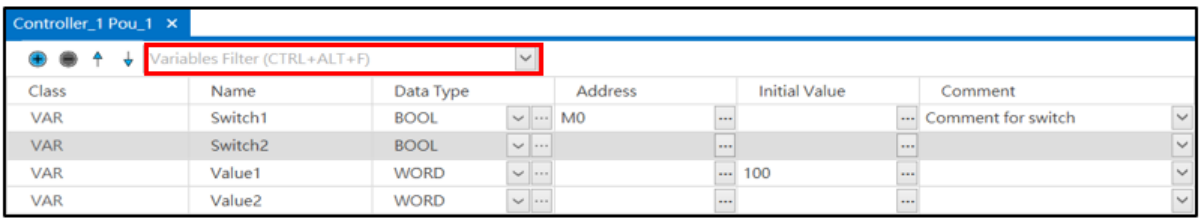


In the variable table of the function block, the arrangement order of the variables of VAR_INPUT, VAR_IN_OUT and VAR_OUTPUT affects the order of the operands of the function block.

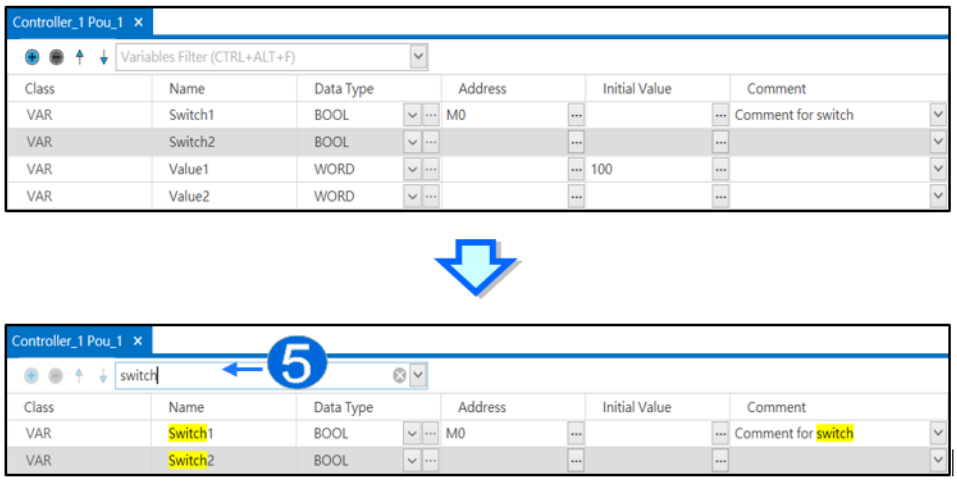


Variables Filter

Input the text to be filtered in **Variables Filter** field at the top of the variable table. The shortcut key for **Variables Filter** is **CTRL** + **ALT** + **F**.



Variables Filter helps you to filter the content by Name, Data Type, Address, Initial Value, and Comment.



8.2.3 Task

Every POU of the program type is assigned to a specific Task. Tasks determine the order in which the POUs of the program type are executed or enabled. Task is like a mission.

Managing Tasks in DIADesigner

In DIADesigner, not all the POUs of the program type in the project are executed. A POU of the program type is executed after it is assigned to a Task and the Task determines how the POU is executed. If a POU is not assigned to a Task, the POU is taken as a general source code and is saved with the project, but not to be translated into an execution code. Only the POUs of program type needs to be assigned to Tasks, Function Blocks are called by POUs.

The characteristic of the IEC 61131-3 architecture is that a program is divided into several POUs which can be developed independently. When all the POUs are compiled, they are rearranged and combined into an execution code which can be scanned step-by-step. The POUs are rearranged and combined to the tasks to which the POUs are assigned.

Task Management

In DIADesigner, a POU of the program type is assigned to a task. More than one POU can be assigned to the same task and the order in which these POUs are executed can be specified. Besides, after adding a POU of the program type, assign it to a default task.

There are three types of tasks.

- Freewheeling
- Cyclic
- Event

Freewheeling Task

A POU assigned to freewheeling task is scanned back and forth. Because there is only one freewheeling task in a project for a PLC, the POUs which are scanned are assigned to this freewheeling task. However, there are 32 freewheeling tasks in a project, which are numbered from 1 to 31. The smaller the task number, earlier the task is executed. You can enable or disable a task during the operation.

A POU which is made with Sequential Function Chart (SFC) can only be assigned to a freewheeling task. It cannot be assigned to an Interrupt task.

Cyclic Task

A POU of the program type which is assigned to a cyclic task is like a cyclic timed interrupt subroutine. If the time when cyclic should occur is reached, the POUs assigned to the cyclic task are executed in order. Besides, the number of the cyclic tasks depends on the PLC selected. The number of cyclic sources that a PLC supports is the number of cyclic tasks that the system has.

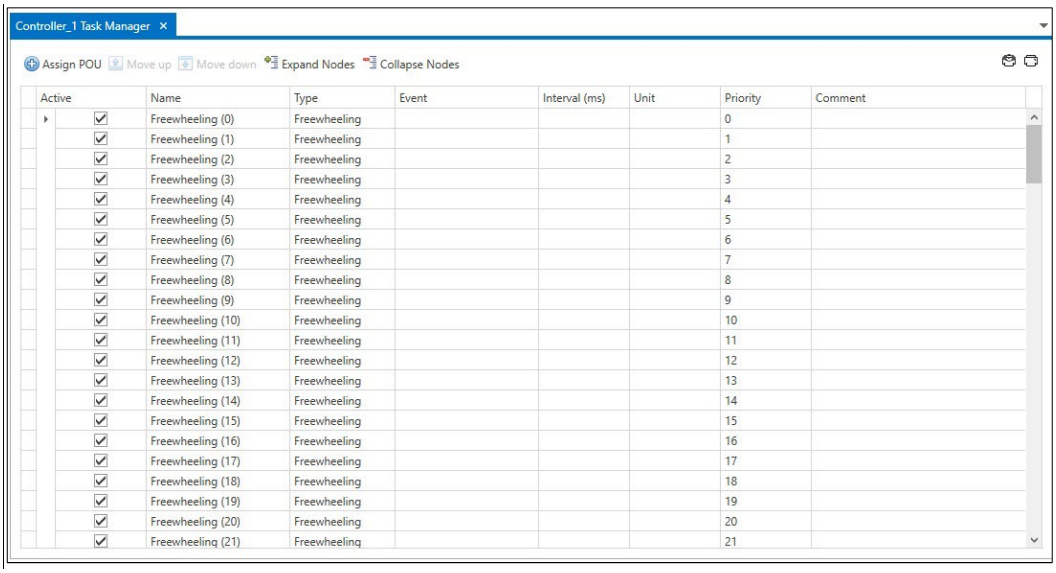
Event Task

There are several types of event tasks. For example, external interrupts, I/O interrupts, counting interrupts etc. Different PLCs provide different event tasks. You must make sure of the conditional interrupt tasks which are supported by the PLC before they assign the POUs of the program type to tasks. A POU of the program type which is assigned to an event task is like an interrupt subroutine. If the condition of an event is

met, for example, if the value of a counter reaches the set point, the POUs assigned to the interrupt task are executed in order.

Tasks in Project Explorer Area

The POUs which are assigned to tasks are listed in the **Task Manager** dialog in **Project Explorer** area. In **Project Explorer**, double-click **Task Manager**.



Task Manager dialog displays the Tasks in controller. For AS controller, the tasks applicable are:

- Cyclic
- External Input Interruption
- Hardware High-Speed Compared Interrupt
- Communication Interrupt
- External Interrupt
- High Speed Output Interruption
- Timed Interruption

All the tasks display with their names, type, event type, priority and comments.

Follow these steps to associate a program type POU to a Task.

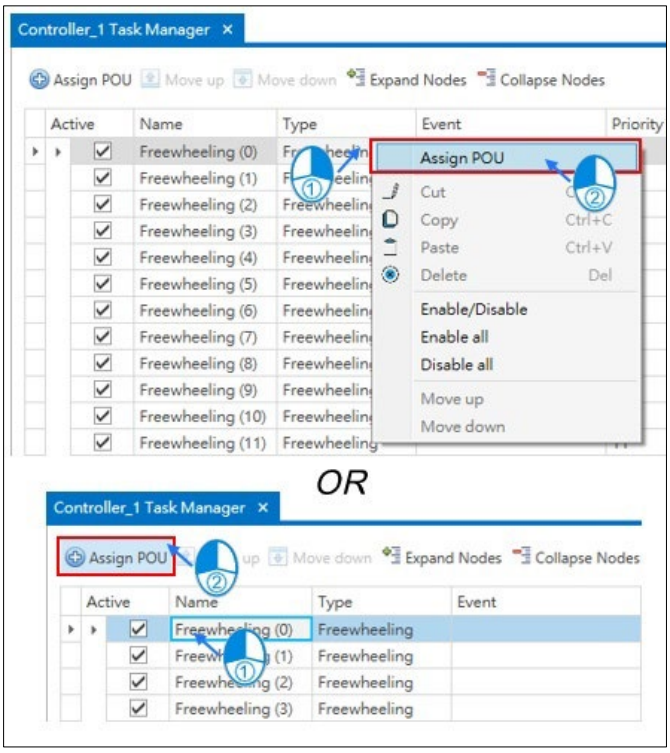
To associate program type POU to Task

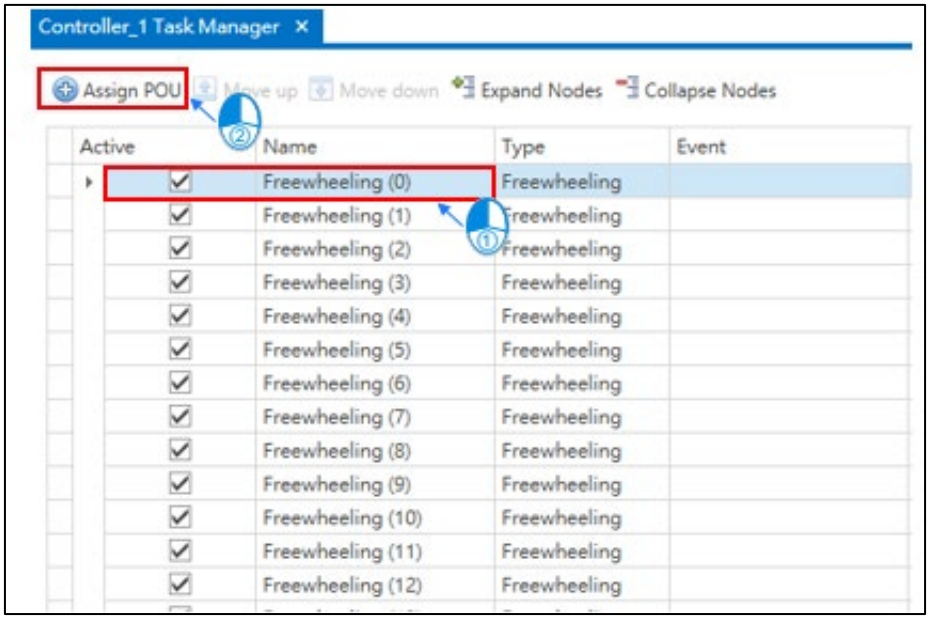
1. Click a particular task to which the POU has to be assigned.

2. Click **Assign POU** in **Task Manager** dialog.

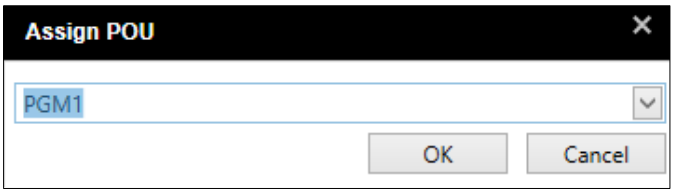
   Or

   Right-click a particular task to open the context menu and click **Assign POU** to open **Assign POU** dialog. It displays the list of program type POUs in the device which are not yet assigned to any task.

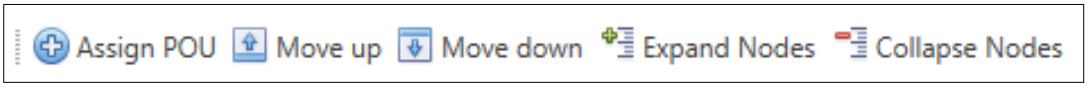




3. Select the necessary POU from the drop-down list and click **OK**.



The following figure shows the toolbar available in **Task Manager** dialog.



The following table describes the functions available in **Task Manager** dialog.

Function / Icon	Description
Assign POU	Select a Task and click **Assign POU** to assign the POU to that Task.
Move up	Select a POU assigned to a Task and click Move up one row icon to execute the current POU before the earlier one.
Move down	Select a POU assigned to a Task and click Move down one row icon to execute the current POU after the below one.
Expand Nodes	Click to expand all nodes.
Collapse Nodes	Click to collapse all nodes.

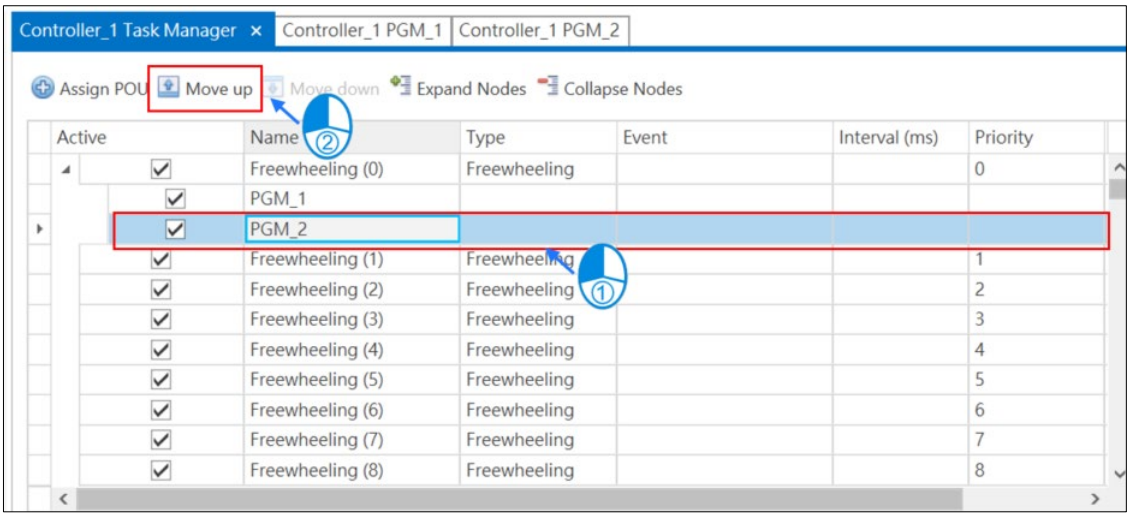
Arranging the POU Order

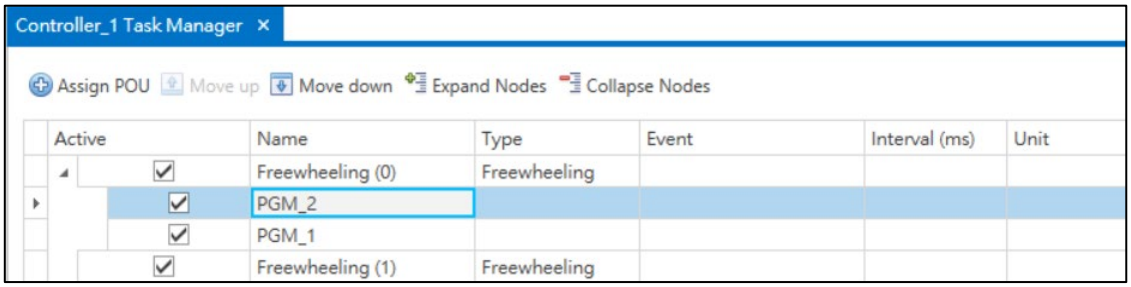
Arrange the POU order (order in which POU is executed in controller runtime), by rearranging them with the help of Move up or Move down functions in the **Task Manager** dialog.

Follow these steps to re-arrange a POU to be executed before another one.

To re-arrange POU

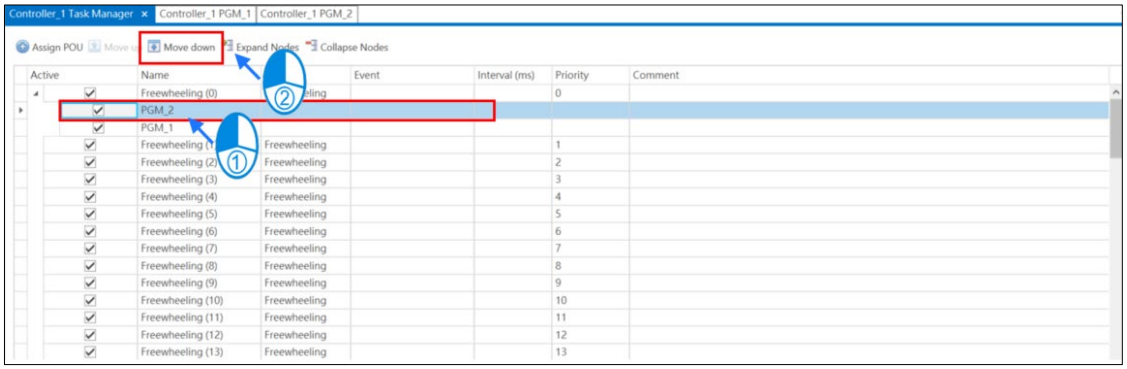
1. After creating several POUs, assign them to suitable Task in Task Manager.
2. Click a POU assigned to the above Task, for example, PGM_2.
3. Click  to re-arrange (make PGM_2 executed before PGM_1) the order as shown in the following figure.

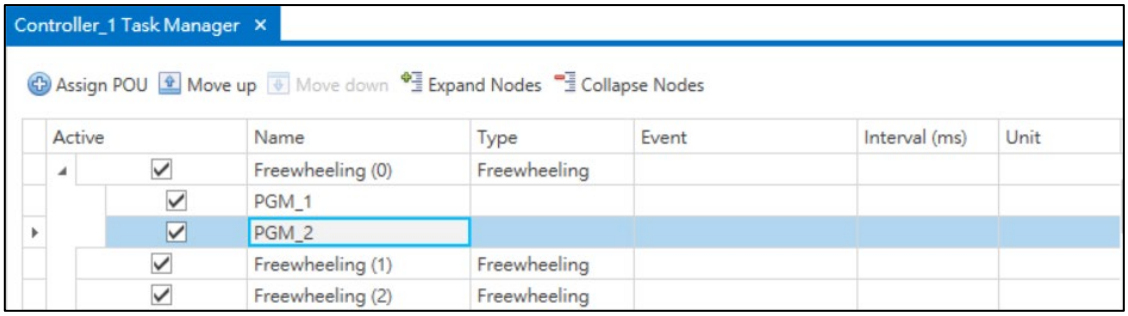




Follow these steps to re-arrange a POU to be executed after another one.

1. After creating several POUs, assign them to suitable Task in Task Manager.
2. Click a POU assigned to the above Task, for example, PGM_2.
3. Click  to re-arrange (make PGM_2 executed after PGM_1) the order as shown in the following figure.





8.2.4 Symbol Configuration

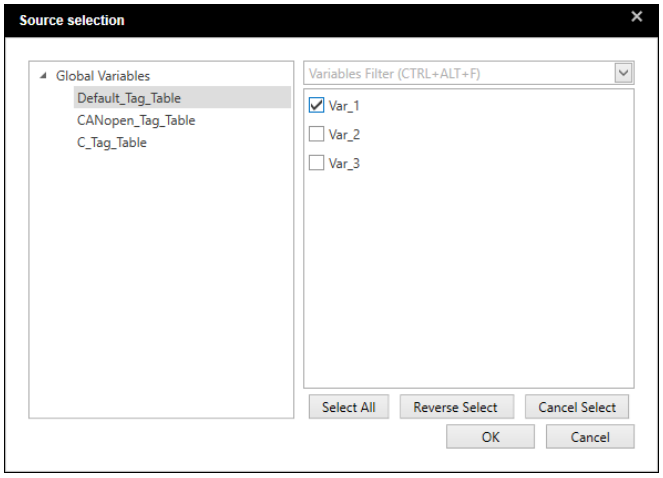
The **Symbol Configuration** function is used to set or view the variable properties. When a variable is both an OPC UA and an EIP variable, you can manage its properties in **Symbol Configuration** without declaring two variables to distinguish them.

OPC UA Symbol Configuration

Follow these steps to set global variables to have OPC UA property.

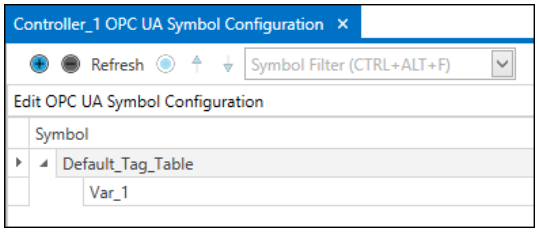
To set global variables to have OPC UA property

1. In **Project Explorer**, double-click **Device** > **Symbol Configuration** > **OPC UA Symbol Configuration**.
2. In the **Source selection** dialog, click  and select global variable.



3. Click **OK**.

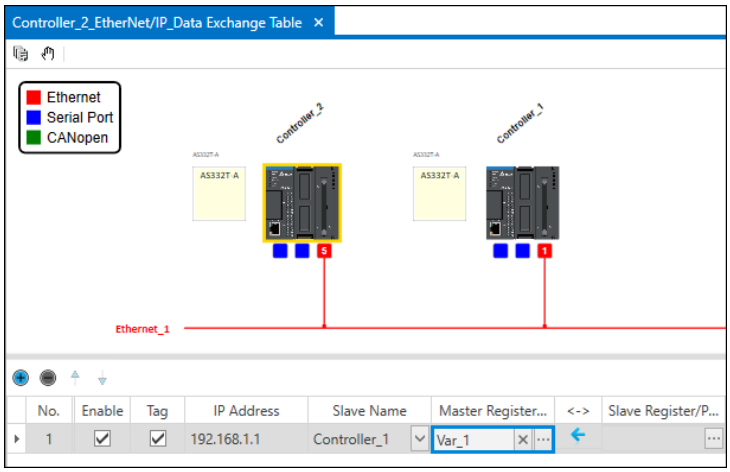
   The OPC UA property setting is completed.



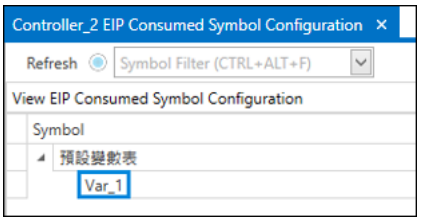
Note: To delete the OPC UA property, single-select or multiple-select variables in the **OPC UA Symbol Configuration** page and click  on the toolbar to delete them.

EIP Produced and EIP Consumed Symbol Configuration

If an EtherNet/IP data exchange table is set in the project and variables are enabled, the variable will be automatically added to the corresponding **EIP Produced Symbol Configuration** or **EIP Consumed Symbol Configuration** when the variable selected in the global variable list is an EIP Produced or EIP Consumed variable.





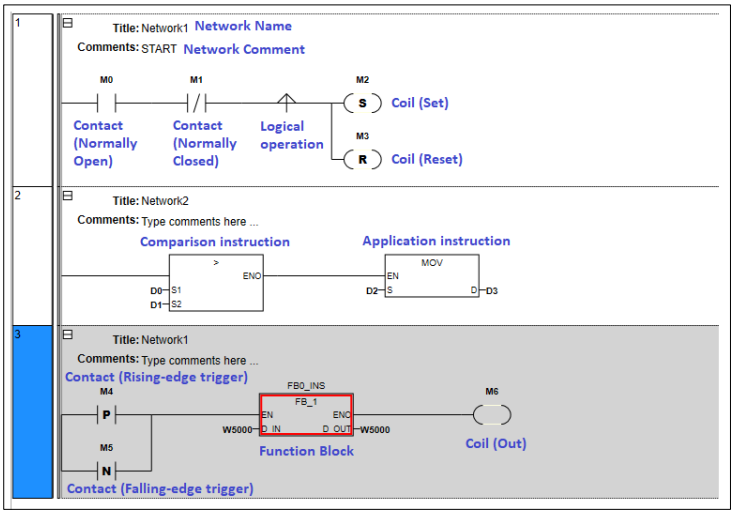


The symbol configuration will automatically update when EIP variables in the EtherNet/IP data exchange table are canceled or changed. You can also manually add or delete the EIP Produced and EIP Consumed symbol configurations. For operation instructions, refer to OPC UA Symbol Configuration.

8.3 Programming Language Introduction (LD, SFC, ST)

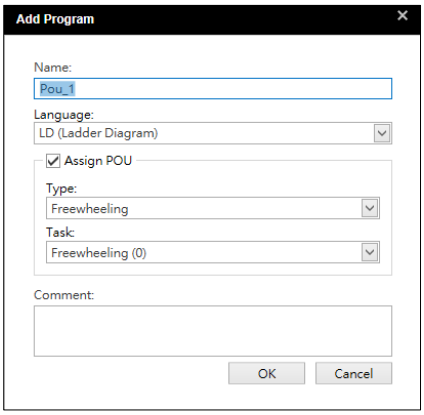
8.3.1 LD—Ladder Diagram

The **Ladder Diagram (LD)** is one of the programming languages defined by IEC 61131-3 and is widely used to create a PLC program. LD is a programming language that represents a program by a graphical diagram based on the circuit diagrams of relay logic hardware. The following figure shows a Ladder Diagram in DIADesigner.



8.3.1.1 Add LD Program

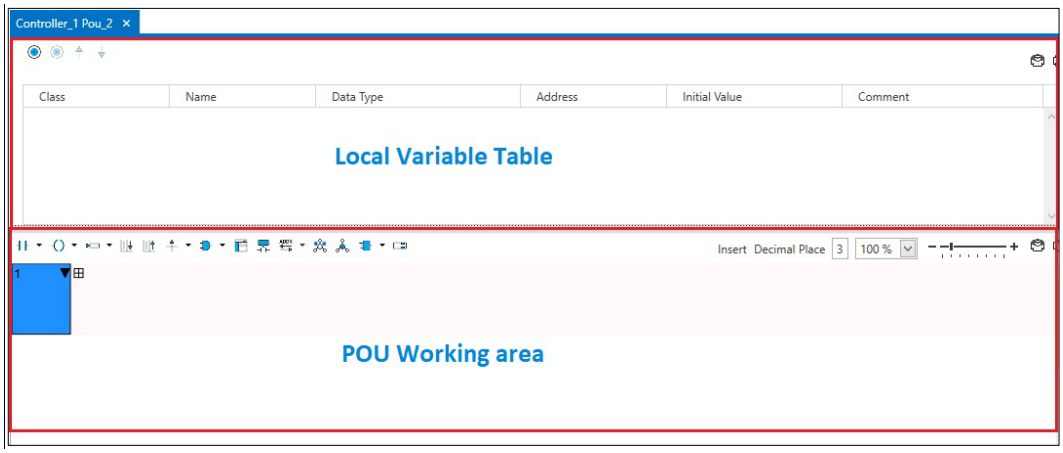
Select **LD (Ladder Diagram)** in the **Language** field in **Add Program** dialog while creating a POU of program type or function block type.



The environment in which a LD POU can be edited is shown in the following figure. In any POU dialog (program type or function block type), there are two areas.

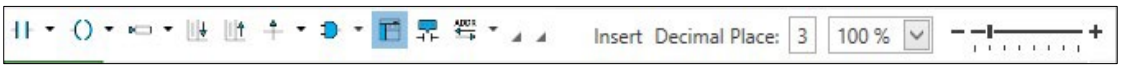
- Local Variable Table area
- POU Working area

The table in the upper part of the LD POU dialog is a Local Variable Table and the lower part is POU Working Area. After a LD POU program editing dialog is opened, the corresponding toolbar appears in the DIADesigner dialog. A blank Network is created by default.



LD Toolbar

After the POU dialog is opened, the toolbar for editing the ladder diagram appears.



The following table describes the functions in LD POU dialog.

Function / Icon	Description
	Contact (Normally Opened)
	Contact (Normally Closed)
	Contact (Rising-edge)
	Contact (Falling-edge)
	Contact (Inverse Rising-edge)
	Contact (Inverse Falling-edge)
	Insert Parallel Contact (Under)

Function / Icon	Description
	Insert Parallel Contact (Above)
	Coil (Output)
	Coil (Set)
	Coil (Reset)
	Coil (Not) **Note**: The function works only for certain devices.
	Insert Parallel Coil (Under)
	Insert Parallel Coil (Above)
	Jump
	Insert Parallel Jump (Under)
	Insert Parallel Jump (Above)
	Insert Network (Under)
	Insert Network (Above)
	Trigger (Rising-edge)
	Trigger (Falling-edge)
	Trigger (Not transition)
	Insert Parallel Trigger (Under)
	Insert Parallel Trigger (Above)
	FB Trigger (Positive Transition-sensing)
	FB Trigger (Negative Transition-sensing)
	Insert Parallel FB Trigger (Under)
	Insert Parallel FB Trigger (Above)
	& and Logical Operators
	Show / Hide Network Comment

Function / Icon	Description
	Show / Hide Register Comment
	Mode of display – Variable, Variable and Address, Address
	Enable / Disable Selected Networks
	Enable / Disable all Networks
	Insert an Instruction/Block
	Insert an Execute Block
	Select Instruction/Function Block
	Insert Parallel Instruction (Under)
	Insert Parallel Instruction (Above)
	Label
	Toggle Bookmark
	Previous Bookmark in Tab
	Next Bookmark in Tab
	Clear All Bookmarks in Tab

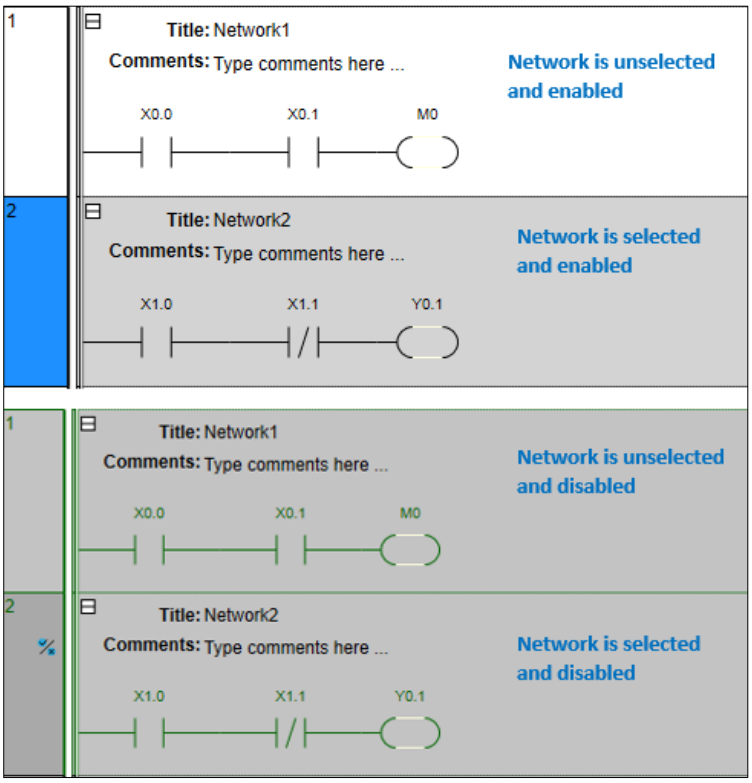
8.3.1.2 Edit LD

Networks in a Ladder Diagram

A Ladder Diagram consists of Networks. Every Network is an independent program. There is no limit on the number of objects which can be inserted in a Network in DIADesigner and therefore there is no mark which is used to connect two Networks.

Network names display in Ladder Diagram in DIADesigner. The color at the left side of a network indicates the state of the network. A network can be enabled or disabled and a network can be selected or unselected. By default, the Network created is activated. To disable a network, select them and click  to disable them. Click  again incase to enable the disabled network.





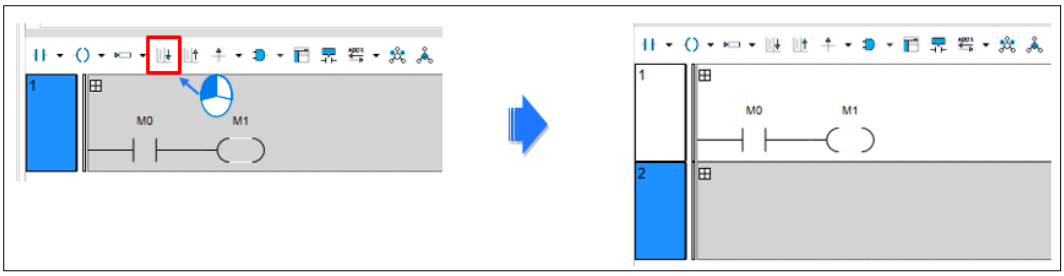
Color of the network number indicates the enable / disable and selected / unselected network as mentioned in the following table.

Selection	Enable / Disable	Network Color (Left Side)
Not Selected	Enabled	White
Selected	Enabled	Blue
Not Selected	Disabled	Gray
Selected	Disabled	Dark Gray

Add Network Below the Current Network

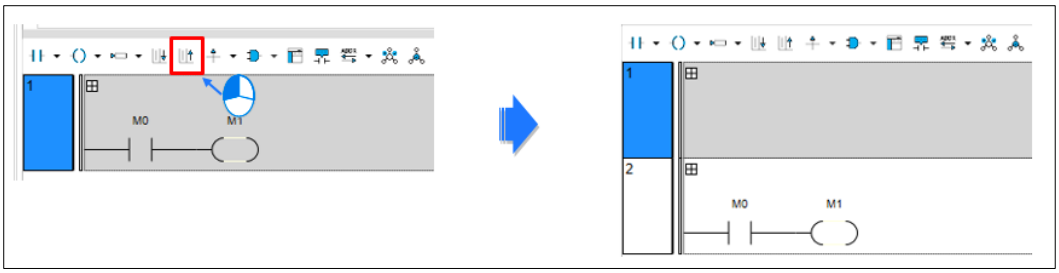
When creating a POU of program type or function block type in LD language, a blank section will be inserted in the POU dialog by default.

If you want to insert a network below the current network, select the network and click  on toolbar.



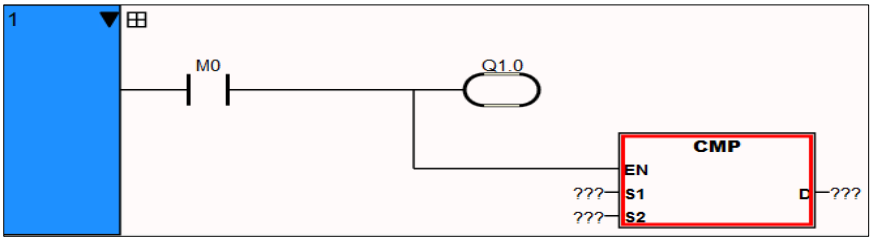
Add Network Above the Current Network

If you want to insert a network above the current network, select the network and click  on toolbar.

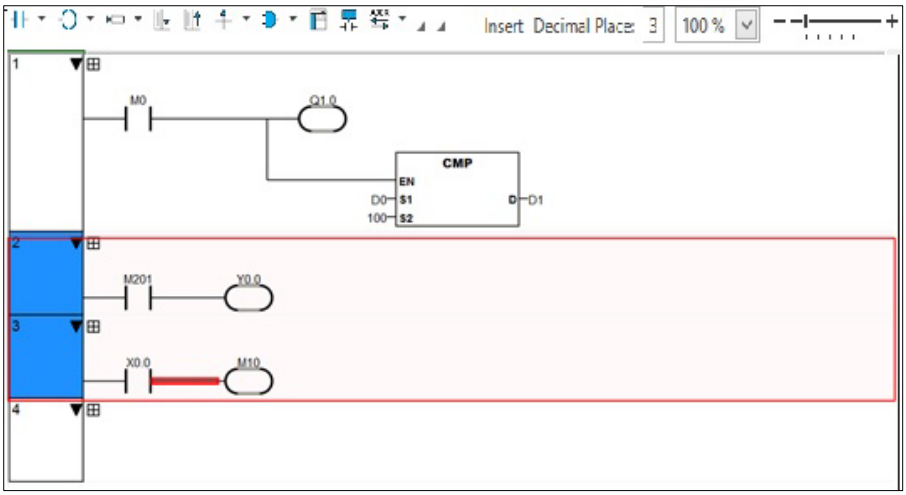


Selecting Objects

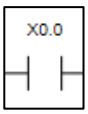
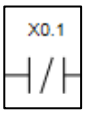
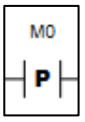
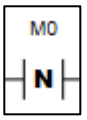
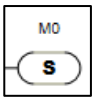
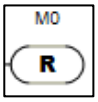
When single-selecting individual contacts / line segments / line areas / objects in the LD language, the selection will be displayed with a red frame or a red line.



If you want to select many networks, hold down **Ctrl** key on the keyboard and click the networks. You can also select a range of networks by pressing **Shift** key on the keyboard, clicking the first network within the range and the last network within the range. Copy and paste the selected networks, and the copied networks will be inserted above the current network.

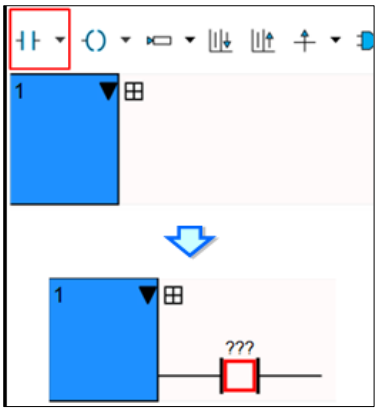


Creating a Contact and Coil

Ladder Element	Description	Representation
Contact	Normally—Open Contact (NO)	
	Normally—Closed Contact (NC)	
	Rising Edge—Triggered Contact (RE)	
	Falling Edge—Triggered Contact (FE)	
Coil	OUT	
	SET	
	RESET	

Inserting a Contact and Changing a Contact Type

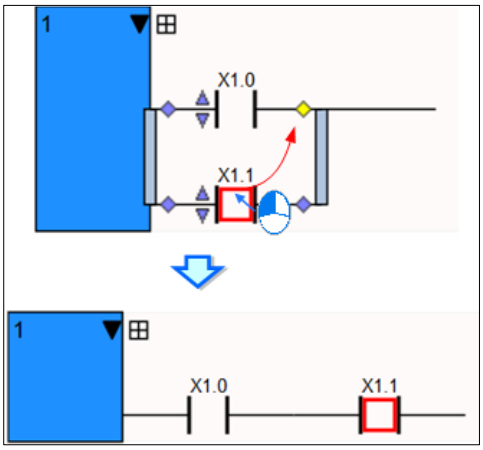
After you make sure of the position in which a contact is inserted, click  on toolbar.



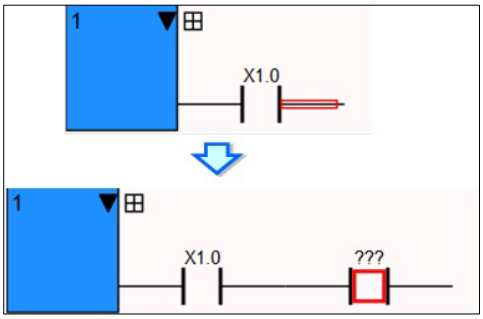
After creating a contact, drag the contact to different locations as shown in the above figure.

- Insert right side / left side contact
- Parallel above / under contact

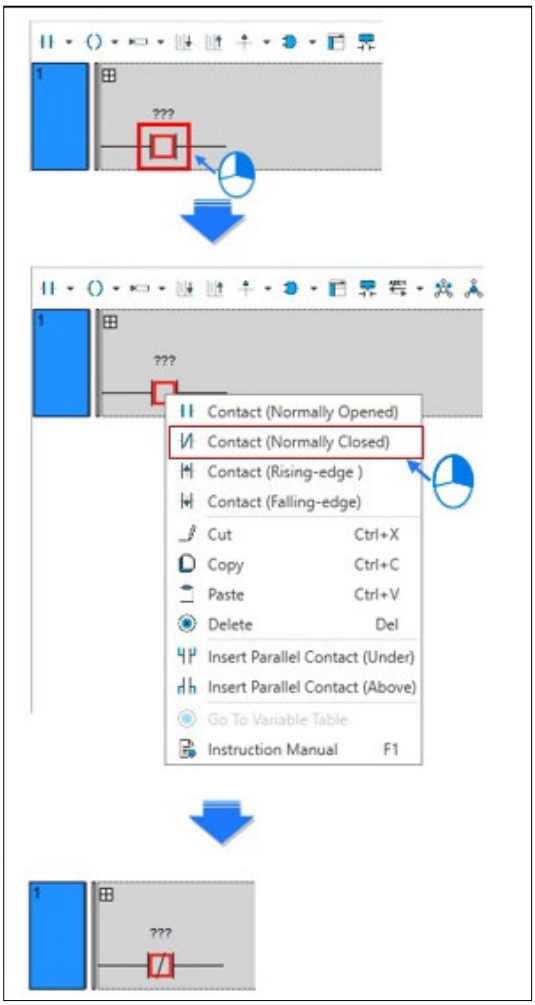
The new position in which it can be inserted is shown by the arrows in those positions. Position the cursor on the arrow, so that the arrow appears in yellow color. Release the cursor to insert the LD element in that position. The inserted element can be dragged to a new position as shown in the following figure.



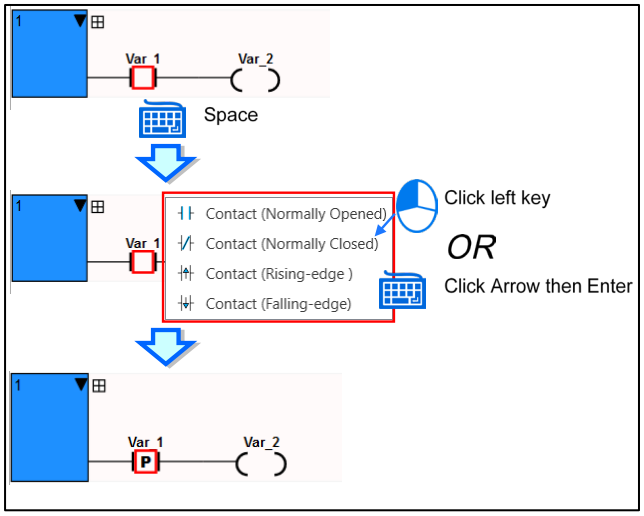
Click an element which is inserted previously and click again to insert a contact.



To change a contact type to a different one, right-click a contact and select the contact type from the context menu as per requirement.

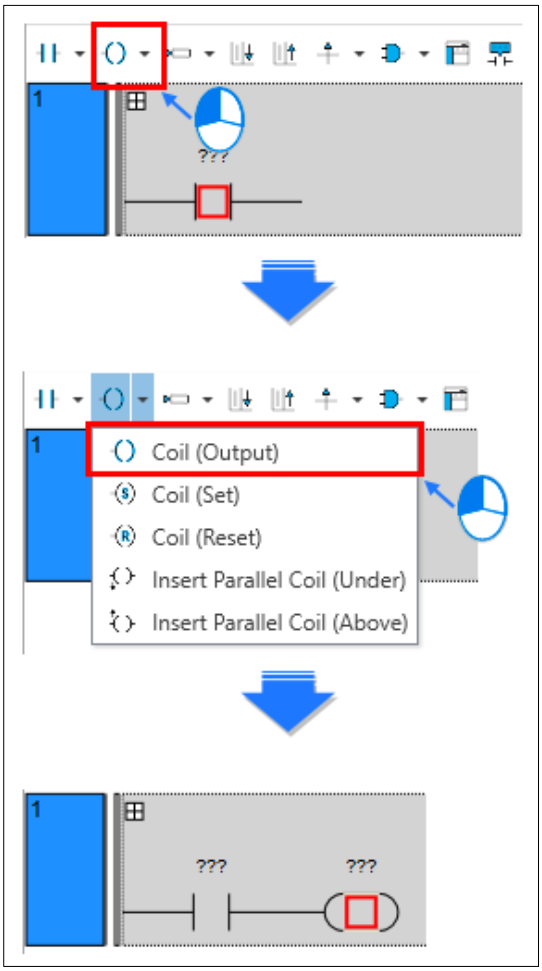


DIADesigner also supports the changing of Contact Type by keyboard. Select the contact by **Arrow** key, then press **Space** key. Select the type you want to change by **Arrow** key, and click **Enter**. The contact changes to the desired type.

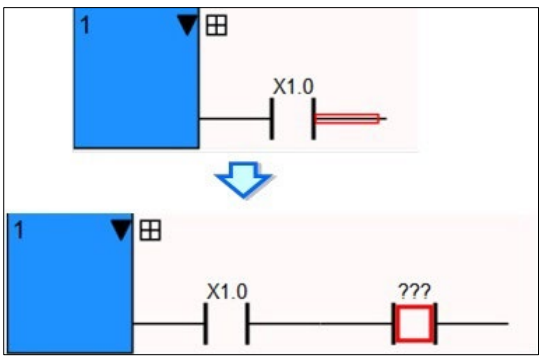


Inserting a Coil and Changing a Coil Type

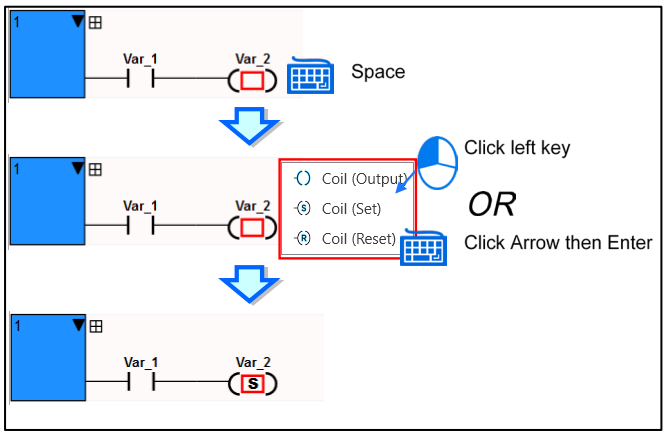
Select a line, a contact, a coil or an applied instruction. After you make sure of the position in which a coil is inserted, click  on toolbar.



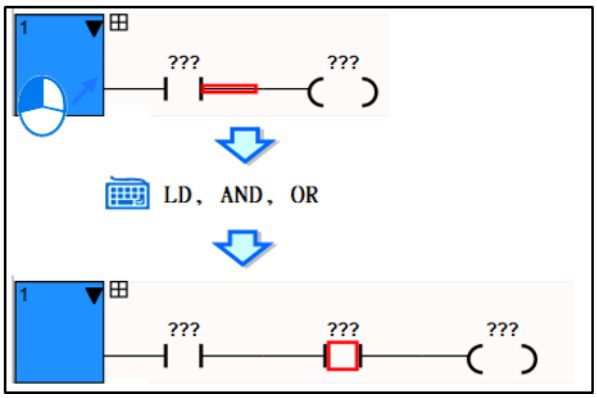
To change a coil type to a different one, right-click a coil and select the coil type from the context menu as per requirement.



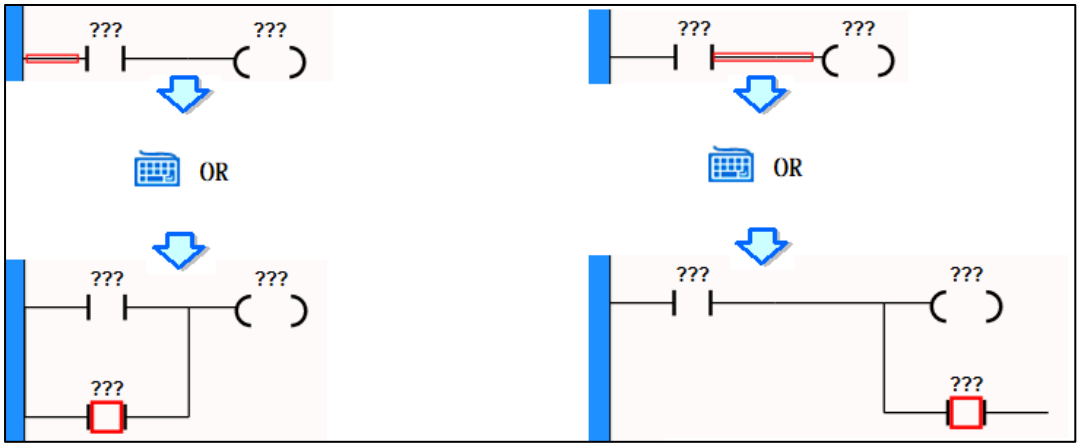
DIADesigner also supports the change of coil type by keyboard. Select the coil by **Arrow** key, then press **Space** key. Select the type you want to change by **Arrow** key and click **Enter**. The coil changes to the desired type.



The LD diagram will have different results depending on the selected position and keyboard input. When clicking the entire network and entering LD, AND, or OR, the action is to add a contact.



Change the selected line segment by the arrow keys on the keyboard and enter OR will have a different contact parallel mode.



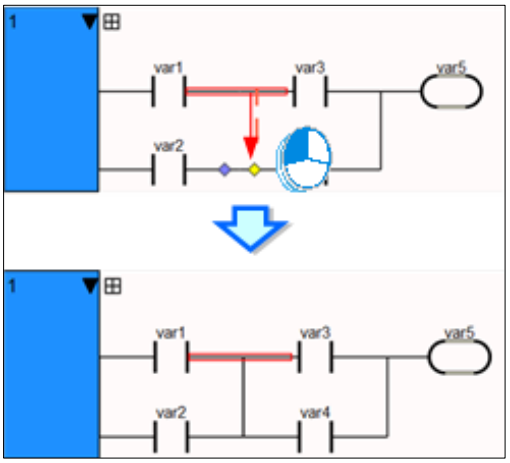
Vertical Operation

DIADesigner provides intuitive way to add or remove vertical line in LD logic.

Follow these steps to add vertical line.

To add vertical line

1. Click the horizontal line of logic.
2. Drag & drop to the next side of horizontal line to connect between both horizontal lines.



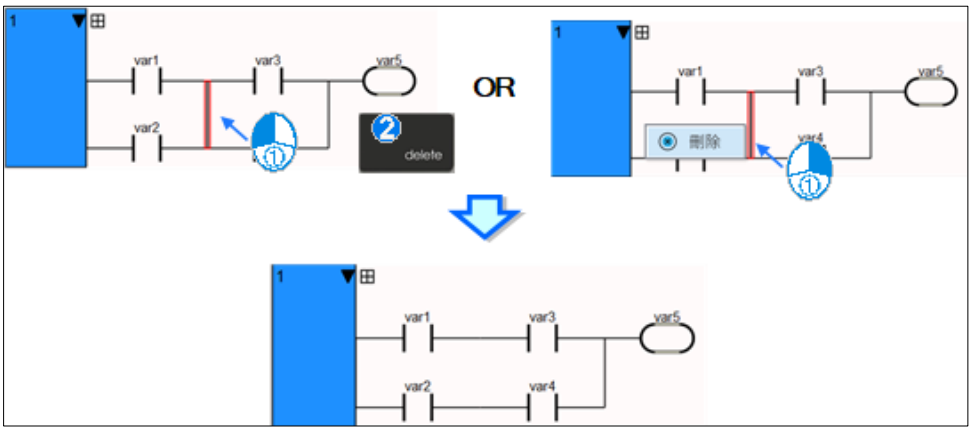
Follow these steps to delete vertical line.

To delete vertical line

1. Click the vertical line.
2. Press **Delete**.

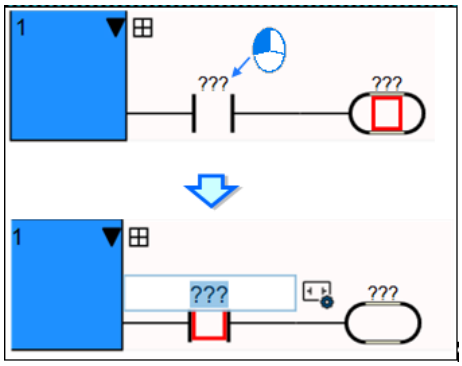
   Or

   Right-click to open context menu then click **Delete**.



Using Address, Variables and Constants in a Ladder Diagram

After you click **???** above an object inserted in a network, type an operand. Operand can be a Variable or Address.

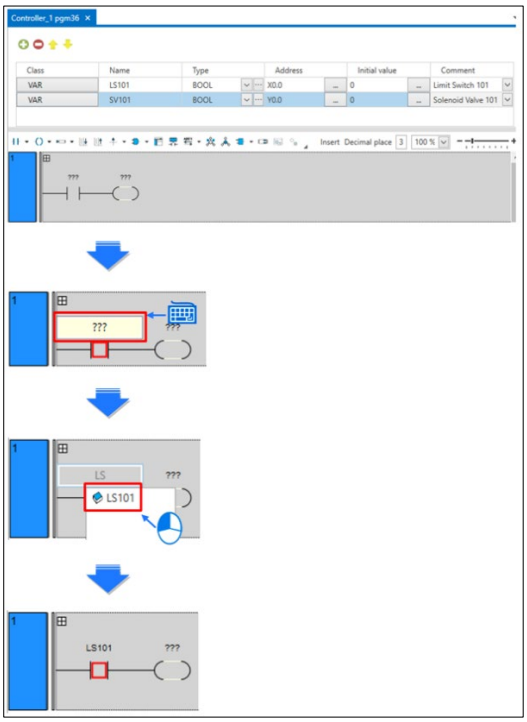


Click **???** on any LD element and enter a Variable name or Address. Address can be typed directly. In case of Variables, there are two type of assignments.

- Assigning Variables created previously.
- Enter a Variable name and declare it.

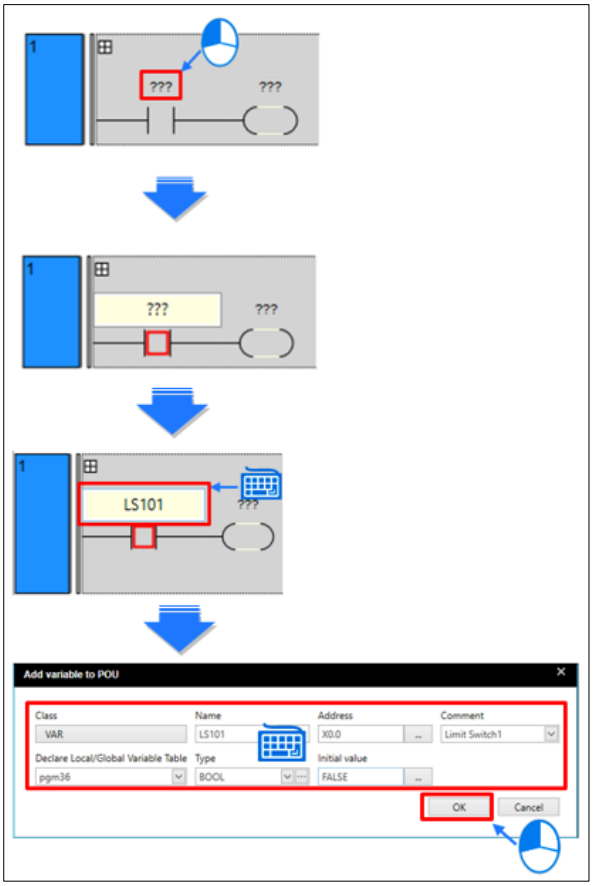
Assigning Variables Created Previously

When typing on the keyboard, the system automatically brings up the keyword menu to list the declared variables. Use the mouse to click or press the up and down keys of the keyboard to select the previously created variables.



Enter a Variable Name and Declare it

Enter the new variable name on the keyboard, and press **Enter** key on the keyboard after the input is completed, and the variable declaration dialog box pop ups to create a new variable.



Additional Remark

If you want to use constants in a Ladder Diagram in DIADesigner, the constants must be represented in the following ways.

- Decimal value: 23456 (a value which is not preceded by any mark is regarded as a decimal value)
- Hexadecimal value: 16#5BA0 (a value which is preceded by 16# is regarded as a hexadecimal value)
- Octal value: 8#55640 (a value which is preceded by 8# is regarded as an octal value)
- Binary value: 2#1011 (a value which is preceded by 2# is regarded as a binary value)

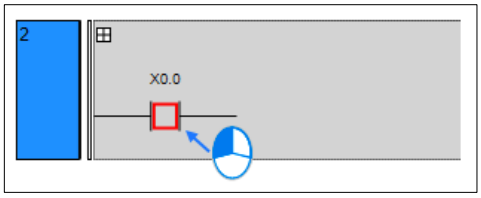
- String value: “XYZ” (a value which consists of characters and inserted in double quotes is regarded as string value)
- Boolean value: AS PLC SM400 (NO Contact) or SM401 (NC Contact) is used.

If you want to use a variable whose data type is ARRAY, the expression format is Ary[…] (Identifier[Index]). An index should be a decimal constant, less than the size of the Array and should start from 0, supporting up to three-dimensional array and separated by a comma. For example, Ary[1,2,3]. Since the Ladder Diagram (LD) doesn’t support variable type index, the indexes in a ladder diagram cannot be variables.

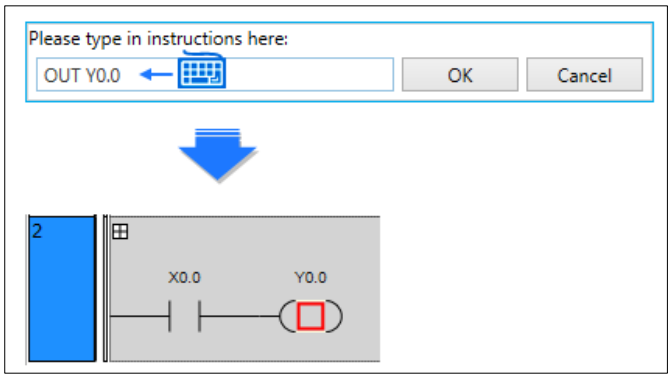
Typing Instructions

Add instructions to a Ladder Diagram by typing instructions. Do the following.

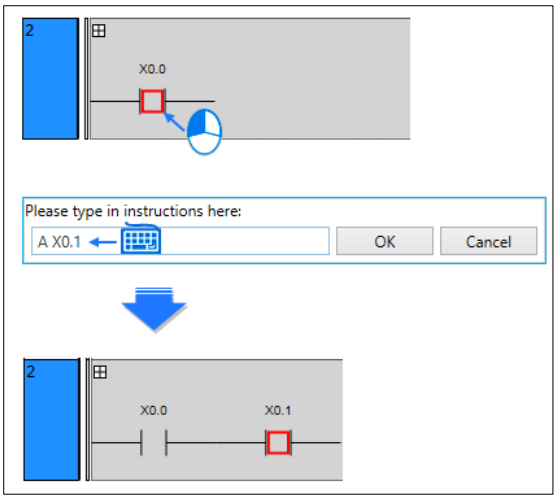
1. Select a position in which an LD object you want to insert.



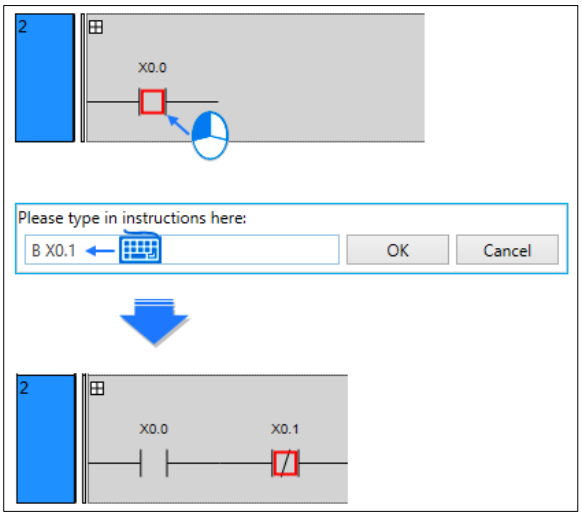
2. Type an instruction. As soon as the instruction is typed, a box which can be edited appears. After the typing of the instruction is complete, press **Enter** key on the keyboard or click **OK** at the right side of the editing box.



To insert a NO contact, type “A” followed by address and click **OK** in the editing box.



To insert a NC contact, type “B” followed by address and click **OK** in the editing box.



The ladder diagram can use P, N, and / to toggle the contact type. Do the following after the contact object is inserted.

- Press **P** key on the keyboard to toggle to the **Rising-edge** type. Press it again to toggle to the **Normally Opened** type.
- Press **N** key on the keyboard to toggle to the **Falling-edge** type. Press it again to toggle to the **Normally Closed** type.
- Press **P** + **/** keys on the keyboard to toggle to the **Inverse Rising-edge**.
- Press **N** + **/** keys on the keyboard to toggle to the **Inverse Falling-edge** type.
- Press the **/** key on the keyboard to toggle to **Normally Opened**. Press it again to toggle to the **Normally Closed** type.

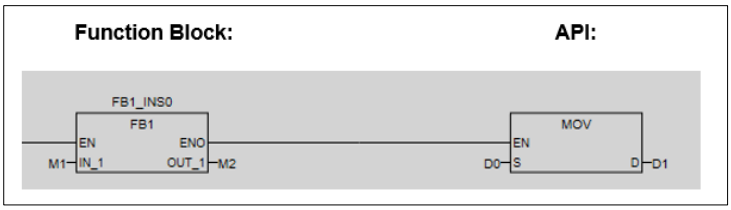
The ladder diagram can use S and R to toggle the coil type. Do the following after the coil object is inserted:

- Press **S** key to toggle to the **Set** type. Press it again to toggle to the **Output** type.
- Press **R** key to toggle to the **Reset** type. Press it again to toggle to the **Output** type.

Inserting Applied Instructions and Function Blocks

The applied instructions and function blocks in a ladder diagram in DIADesigner are represented by blocks. The pins of a block representing an applied instruction include the pins corresponding to the operands specified in the applied instruction and an **EN** pin. The pins of a block representing a function block include the pins corresponding to the variables declared in the function block and an **EN** pin. The **EN** pin of a block representing an applied instruction or a function block is connected to the logic state preceding the **EN** pin. If the logic state connected to the **EN** pin of a block representing an applied instruction or a function block is ON, the applied instruction of the function block is executed.

The pins of a block representing a function block include an **ENO** pin whereas the pins of a block representing an applied instruction do not include an **ENO** pin. An applied instruction can only be put at the end of a network in a ladder diagram and cannot be followed by any object. Besides, type a function block instance in the box above a function block.

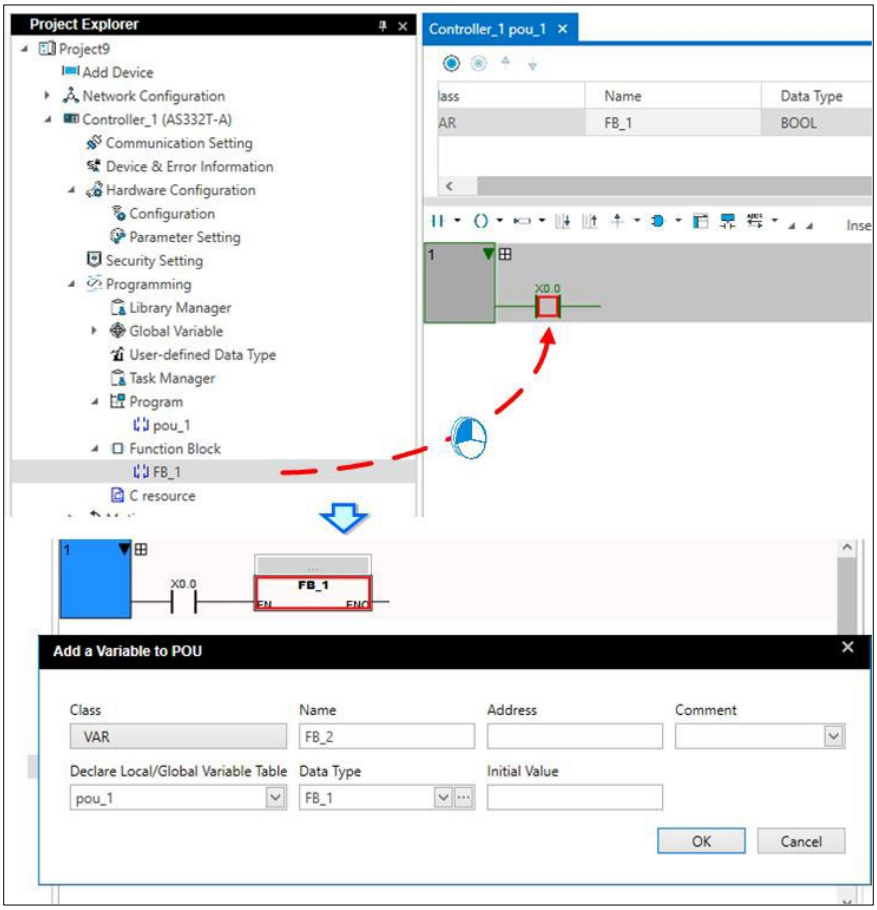


Insert an applied instruction or a function block in a ladder diagram in DIADesigner in one of the following three methods.

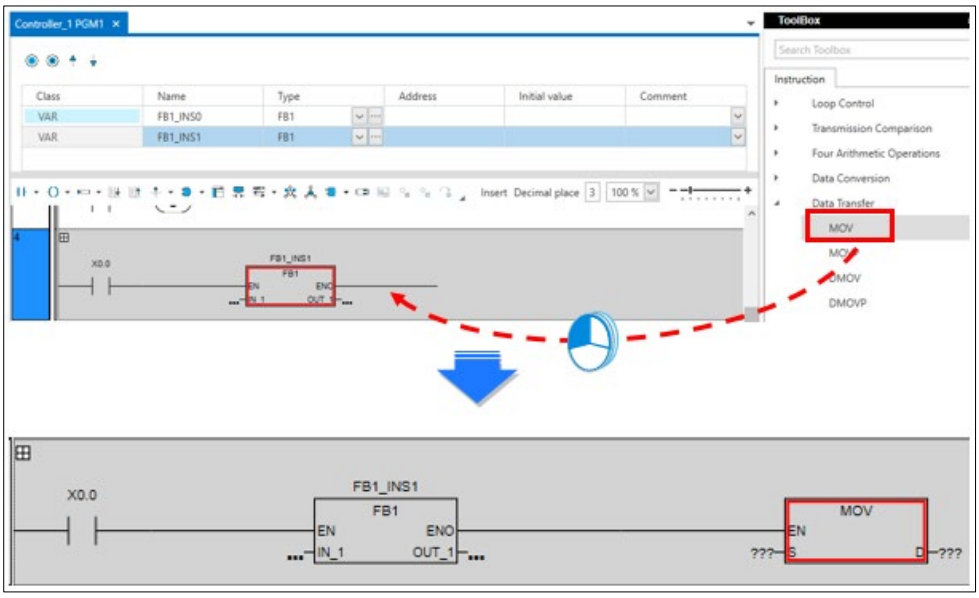
Method 1

Unfold the APIs section or the **Function Block** section in **Project Explorer** and find the item which is inserted. Select the item and drag it to the position in which it is inserted.

Function Block



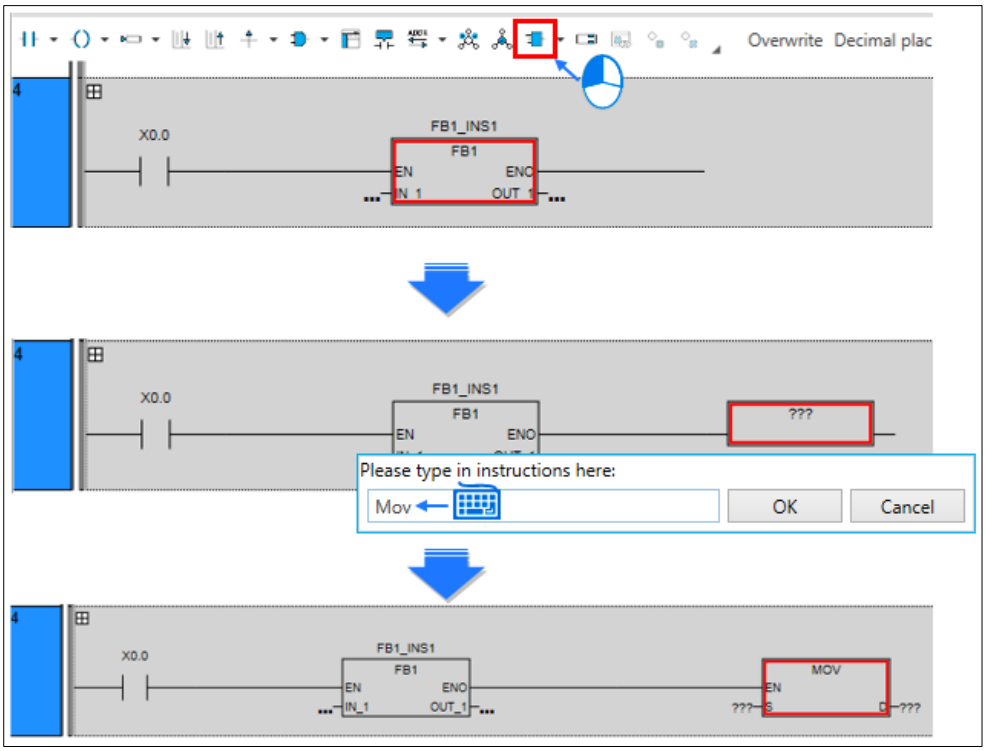
API



Note: When the mouse cursor is held over an instruction in the toolbox, a description of the instruction will appear.

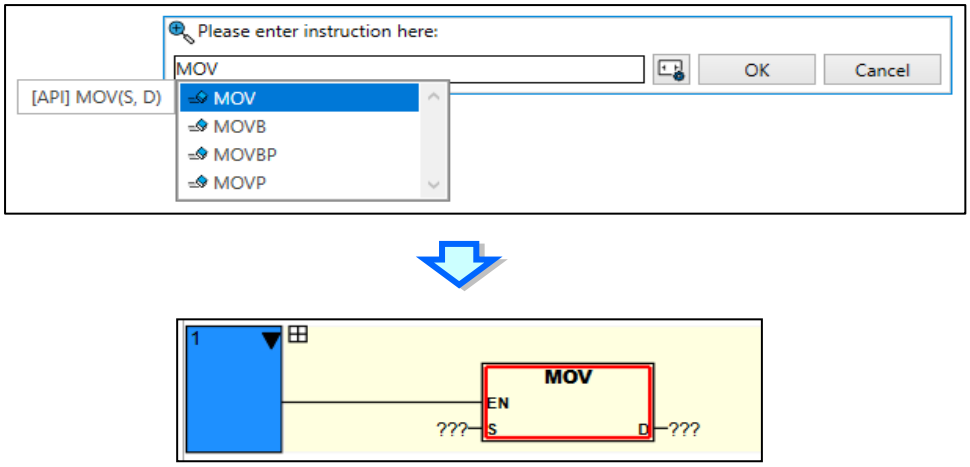
Method 2

Click the position in which an applied instruction or a function block is inserted, click  on toolbar to insert an empty block. Enter an instruction or function block name in the added block.

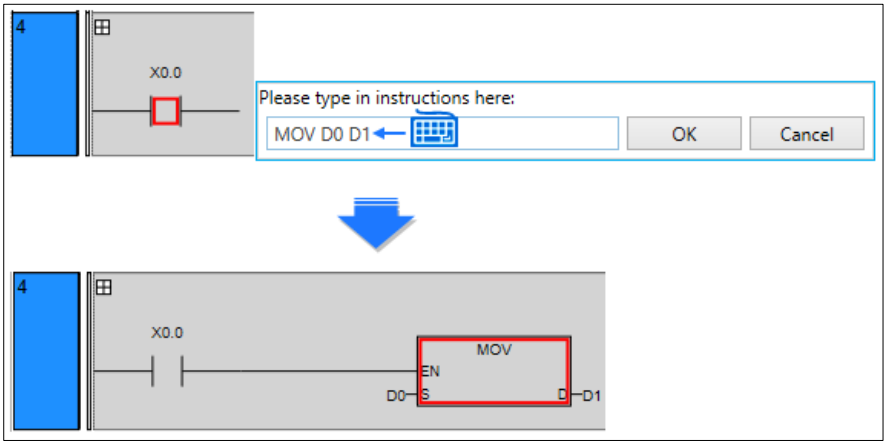


Method 3

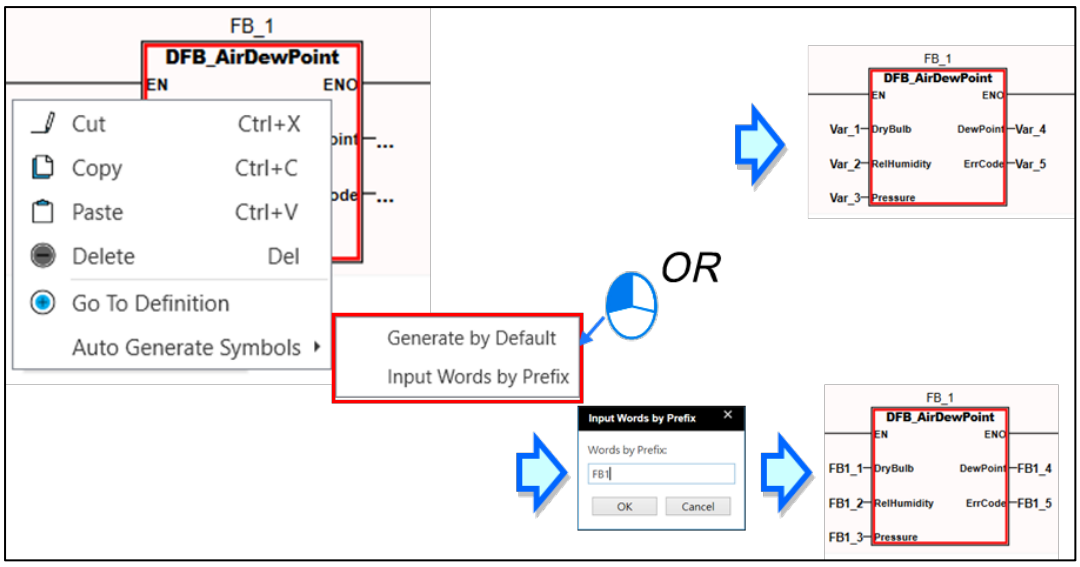
Click the position in which an instruction or a function block is to be inserted and type the applied instruction or the function block definition. The pop-up instruction input box will display the candidate list the syntax format (the applied instruction and the function block diagram are case-insensitive). After typing of the applied instruction or function block definition is complete, press **Enter** or click **OK**.



To insert a function block, type the function block definition and you cannot type the operands specified in the function block. If you want to insert an applied instruction, you must type the applied instruction, and you can type the operands specified in the applied instruction. Do not have to type all the operands specified in the applied instruction. The system assigns the operands typed to the corresponding boxes and **???** appears in the boxes where there are no operands. Besides, if the operands typed include variable which is not declared, the **Add a Variable to the POU** dialog appears after pressing **Enter** key on the keyboard. Refer to the instruction manual of each model for detailed instruction syntax.

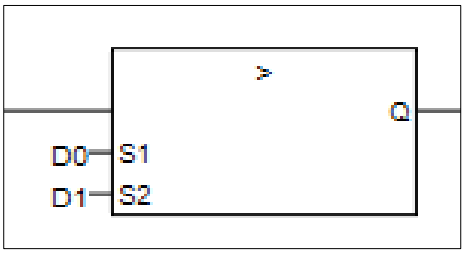


Right-click **Function Block** to open the context menu. The variable of this function block pin can be generated by the **Auto Generate Symbols** function. Select **Generate by Default** to generate the variables with a combination of FB name + pin name or select **Input Words by Prefix** to generate the variables with a user-defined prefix.

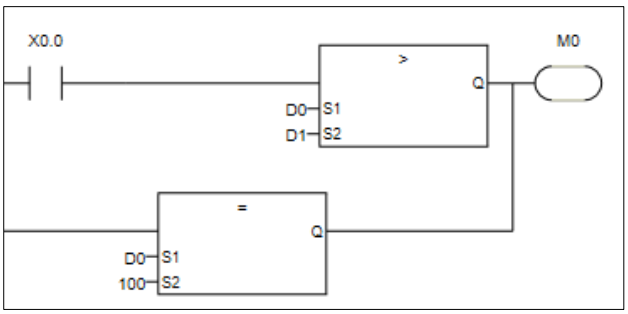


Creating Comparison Contact

An example of a Comparison Contact in DIADesigner is shown in the above figure. **>** in the block represents a comparison type. **S1** and **S2** are objects which are compared. The data format which is accepted depends on the comparison type. **Q** can drive a coil or connect to a contact in series.



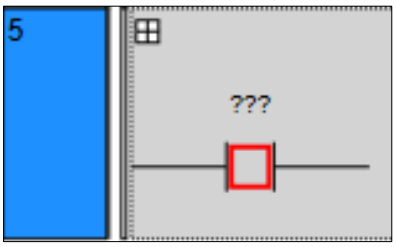
The comparison contacts in a ladder diagram are similar to general contacts in that they can be connected to other contacts in series or parallel, as shown in the following figure.



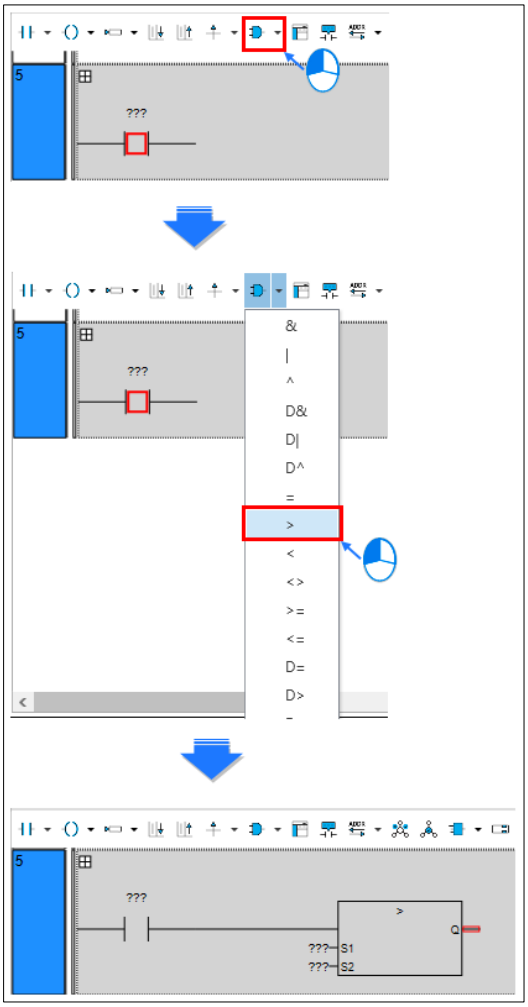
A comparison contact can be inserted not only in one of the three ways described in previous section, but also by following steps.

To insert comparison contact

1. Click a line or contact or an instruction.



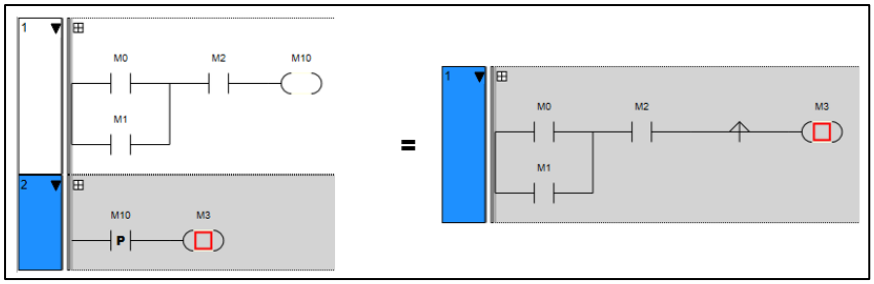
2. Click  on toolbar and then select a type of comparator.



3. Type the Variables or Address for the S1 and S2 pins.

Inserting a Block Logic Instruction

To take the operation result of a block as a condition of a rising edge-triggered circuit, you traditionally must create two networks in a ladder diagram. However, you can combine two networks by means of using a block logic instruction.



The following table describes the block logic instructions available in DIADesigner.

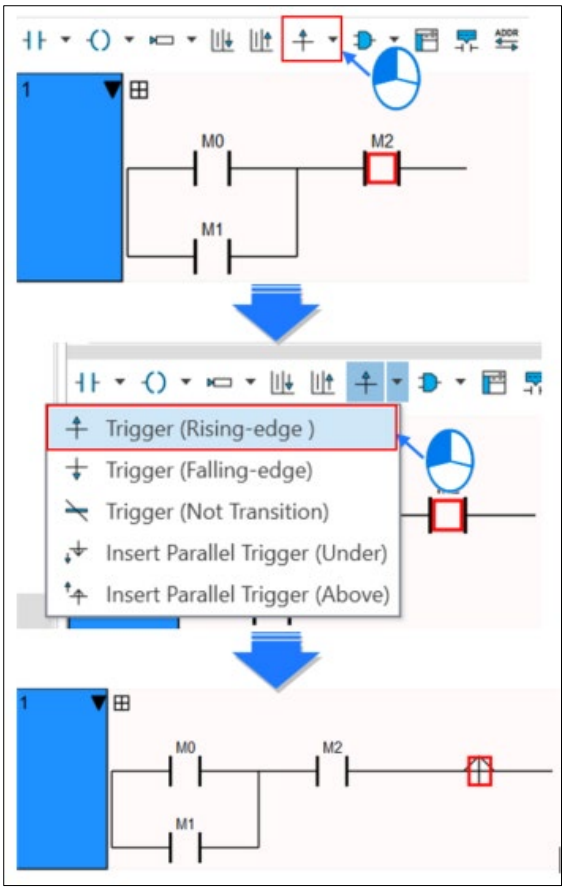
Representation	Instruction	Function
	NP	Click to insert Positive Transition Trigger. When the block preceding this block logic instruction is turned from OFF to ON, the output device is ON.
	PN	Click to insert Negative Transition Trigger. When the block preceding this block logic instruction is turned from ON to OFF, the output device is ON.
	INV	Click to insert Not Transition Trigger. The operation result of the block preceding this block logic instruction is inverted and the inversion result is sent.
	NP-	Click to insert a Parallel Trigger in parallel under the existing line or contact.
	-NP	Click to insert a Parallel Trigger in parallel above the existing line or contact.

Refer to the programming manual of each model for the instructions of FB_NP and FB_PN.

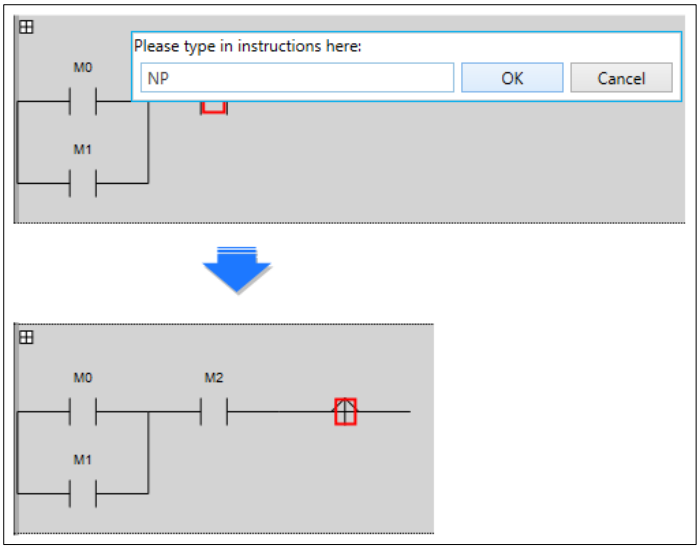
Insert a block logic instruction by following methods.

- Clicking the respective icon on toolbar
- Typing the instruction

Clicking Respective Block Logic Instruction from Toolbar

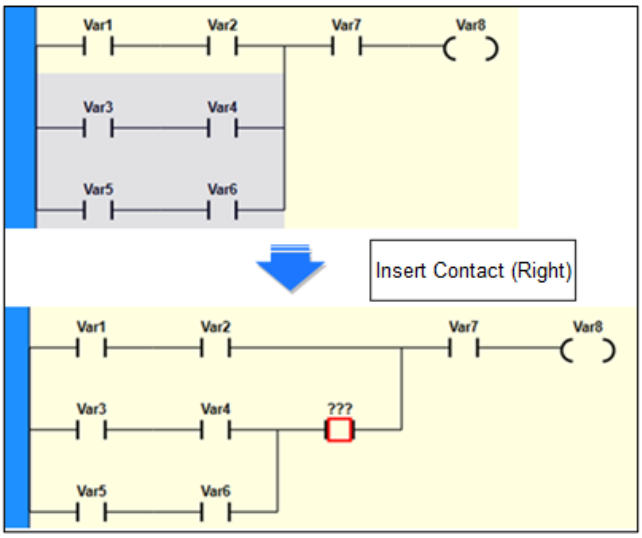


Typing Block Logic Instructions



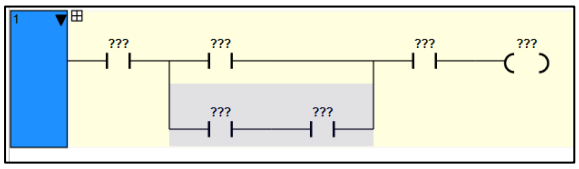
Block Series and Parallel

To add series or parallel operation to a block, you can right-click the block and select Insert Contact (Left) or Insert Contact (Right) or Insert Parallel Contact (Above) or Insert Parallel Contact (Under) according to your requirement. The following figure shows an example of inserting a contact (right) in a block.

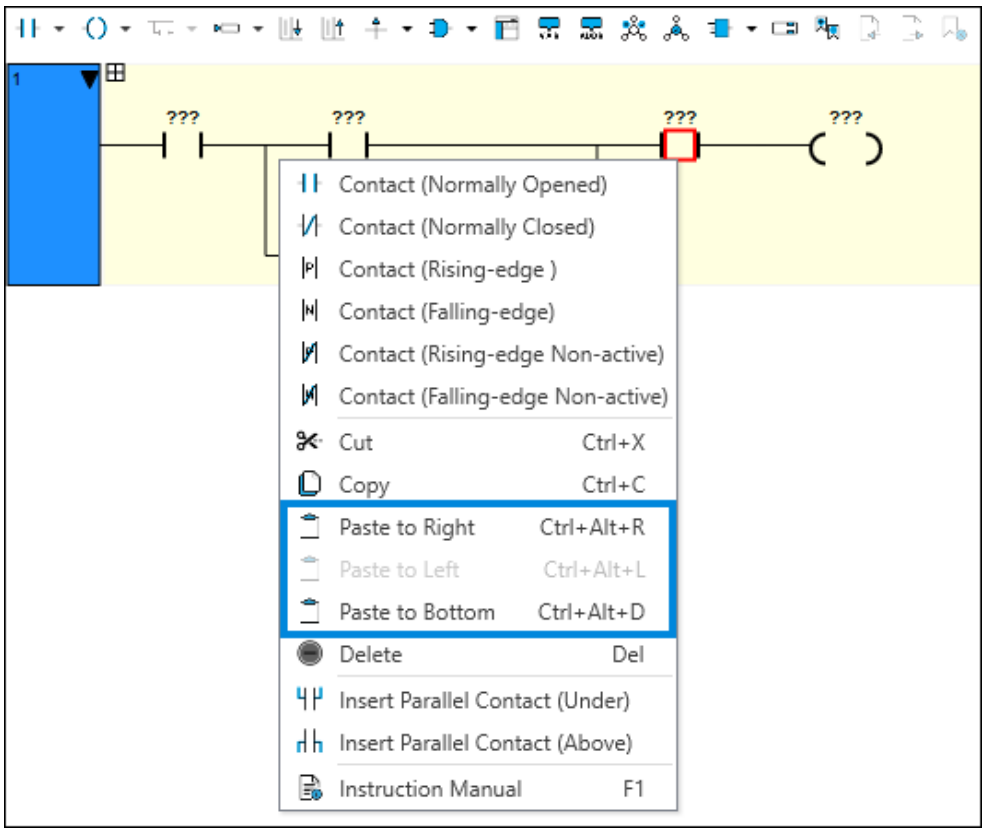


Block Copy and Paste

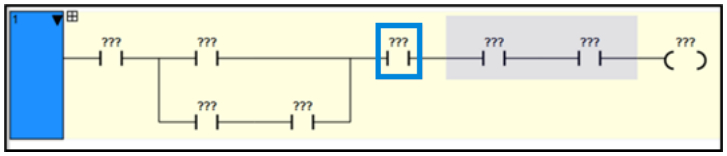
To copy a block and paste it, you can copy the block, as shown in the following example, to copy two contact objects.



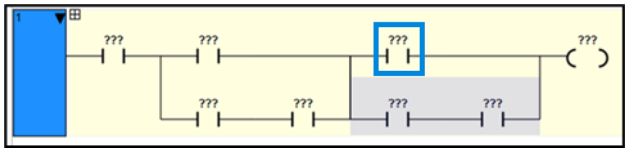
Then, choose to paste it to the right or below the desired position.



The following figure demonstrates pasting the copied block on the right of the selected contact.



The following figure demonstrates pasting the copied block below the contact.

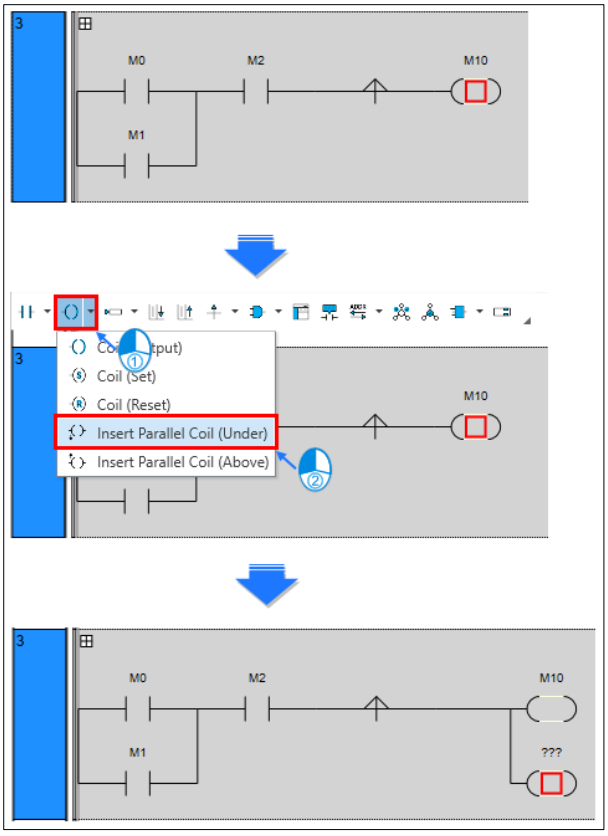


Note: **Paste to Left** can be selected only when the coil is selected.

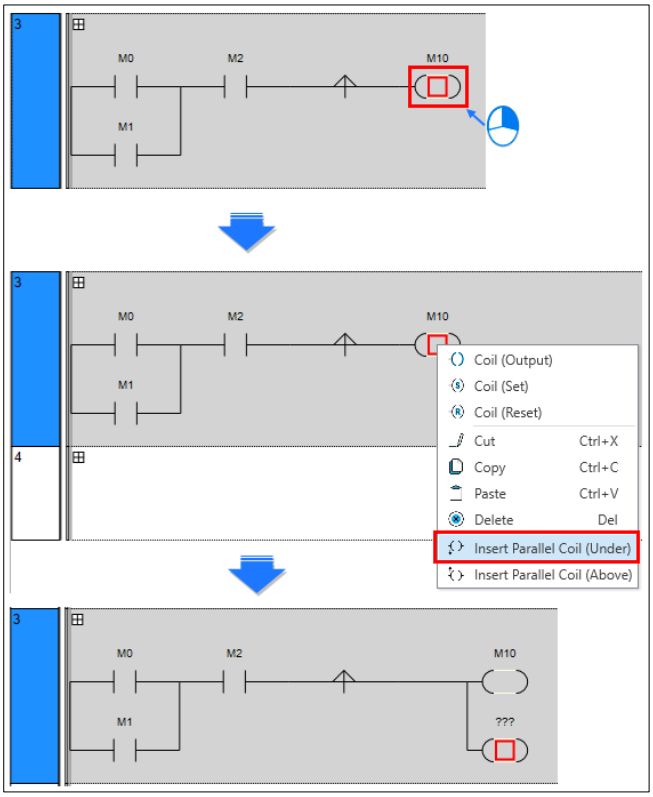
Creating Multiple Outputs

If you want to create multiple outputs, insert a coil or an applied instruction first. After this, insert multiple outputs from toolbar or from the context menu of the output coil/applied instruction.

Clicking Multiple Output icon on Toolbar



Selecting Multiple Output Coils from Context Menu



Putting a Label

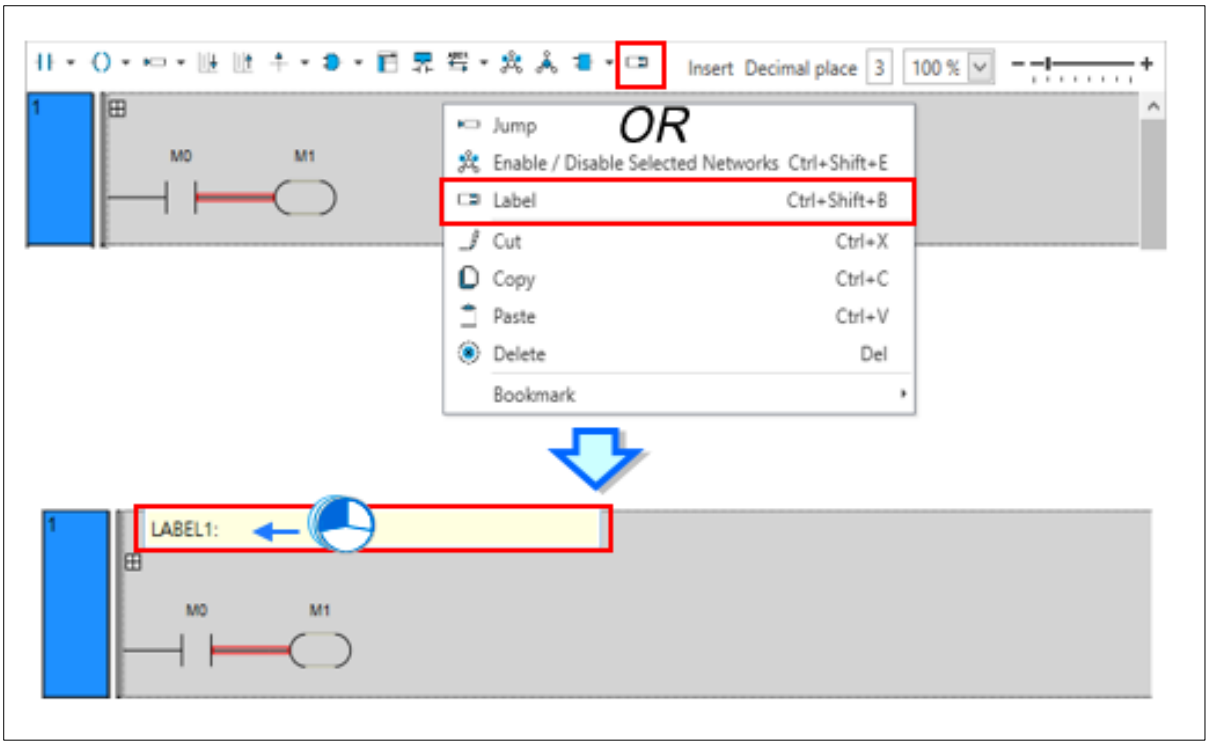
If you want to use a jump instruction, put a label on the network to which the execution of the program jumps. A label is put at the top side of the network.



Follow these steps to insert a Label in an LD Network.

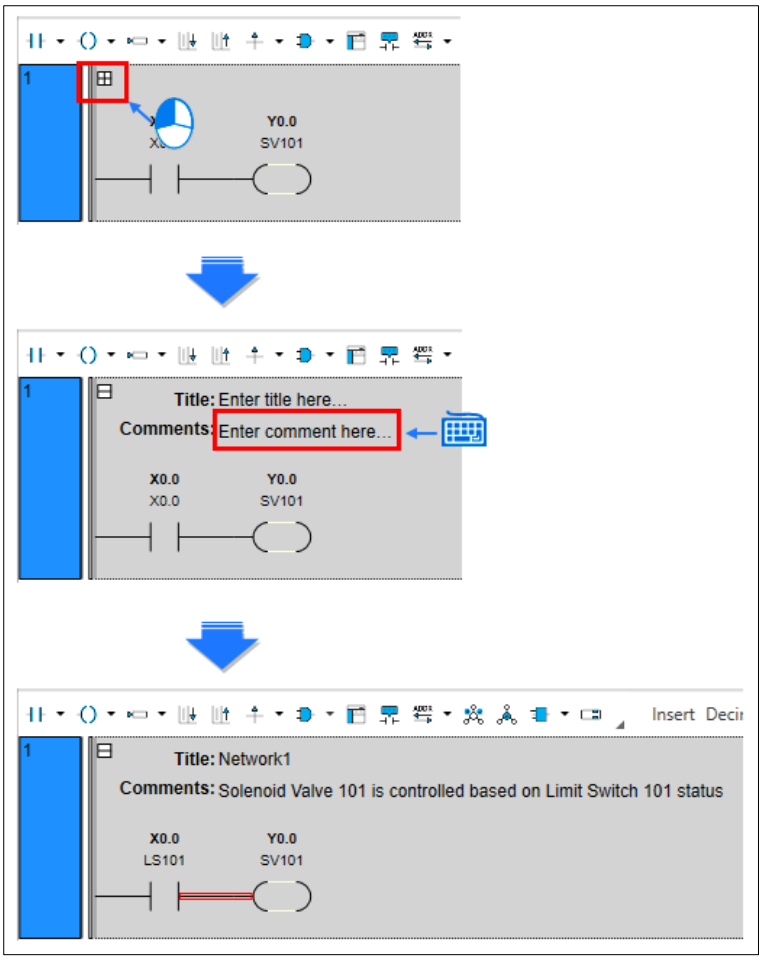
To insert Label in LD Network

1. Click the network at which a label is to be added.
2. Click  on toolbar or in the context menu.
3. Enter **Label** name.
4. Press **Enter** key on the keyboard after a label is typed. The system automatically puts a colon at the right side of the label.



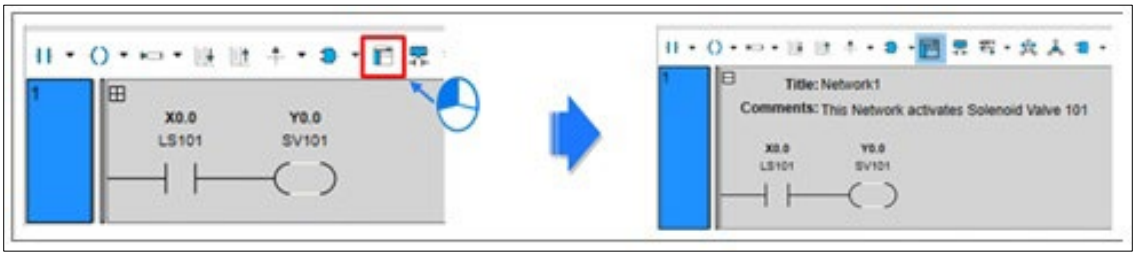
Editing Title and Comment

In **Ladder Diagram** in DIADesigner, you can not only make comments on Variables and Address, but also add descriptive titles and comments on the Networks. Click above **Networks** to expand **Title** and **Comments** boxes to enter title and comments.

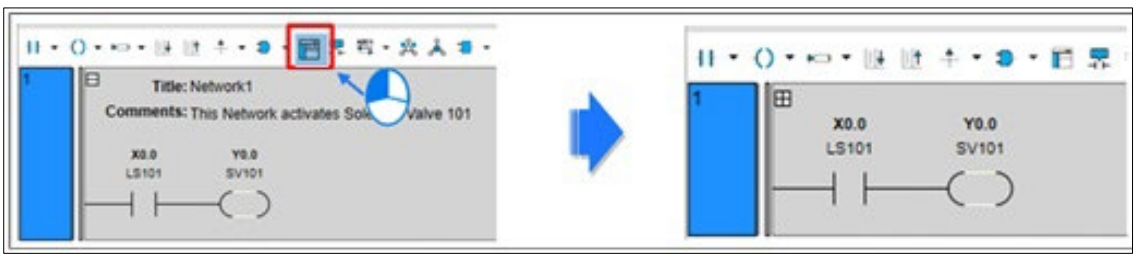


Show / Hide Title and Comments on a Network

After clicking  on toolbar, all the titles and comments on the networks of the current POU will show / hide.

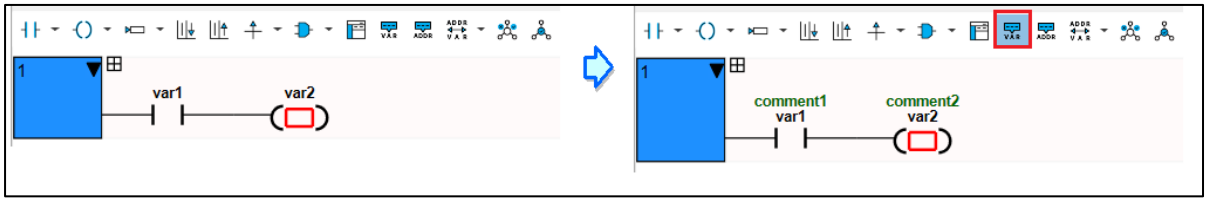


Click the same icon once again to hide them.

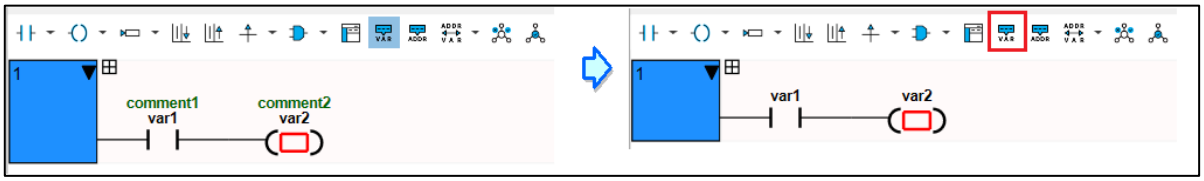


Editing the Comment on a Variable in LD Element

You can select whether to display the comments for variables or registers. On the toolbar, click  to display the comments of variables in the LD program and click  to display the comments of registers in the LD program.



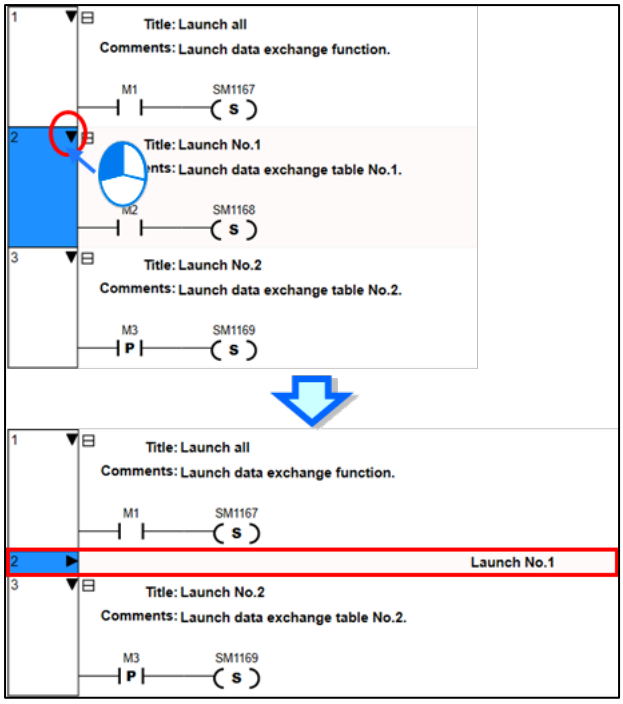
To hide the comments, can click the same icon once again.



After clicking the comment position above a variable or register name, type a comment in the box which appears. Press **Enter** after the editing is complete.

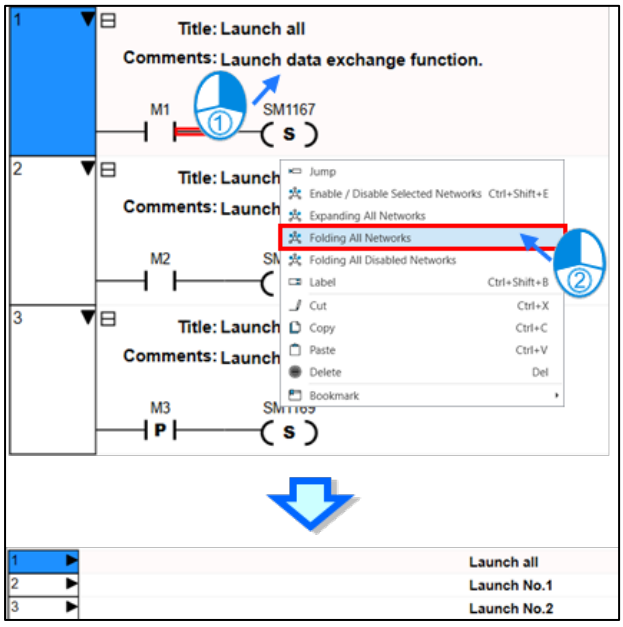
Network Folding and Expanding

When you edit many program networks, some of the networks can be folded to have a simple display, which is convenient for the program query. Pressing  at the header in the unfolding network can fold the network, and the folded network only displays the program heading.

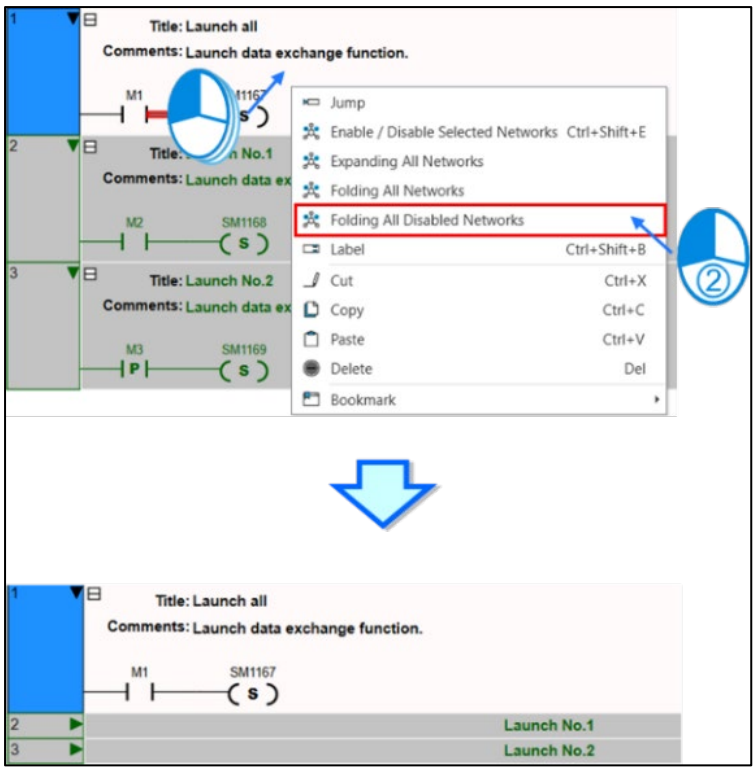


On the other hand, pressing ▼ at the network header in the folded network can expand the network again.

If you would like to fold all the networks, right-click any network and click **Folding All Networks** in the context menu, then all the networks can be folded, as shown in the following figure.



If you would like to fold all the disabled networks only, right-click any network and click **Folding All Disabled Networks** in the context menu, then all the disabled networks can be folded, as shown in the following figure.



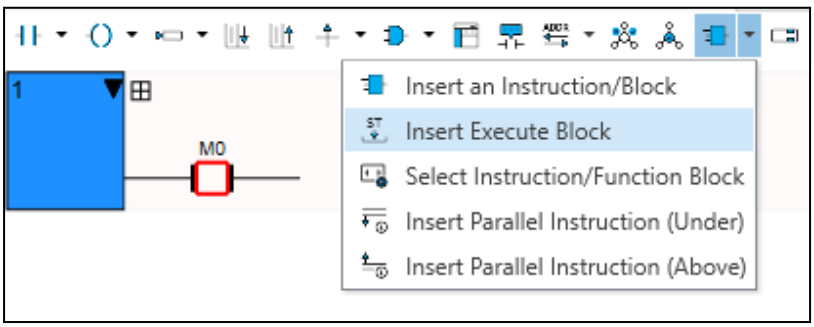
Right-click any network and click **Expanding All Networks** in the context menu, then all the folded networks can be expanded at once.

Insert Execute Block

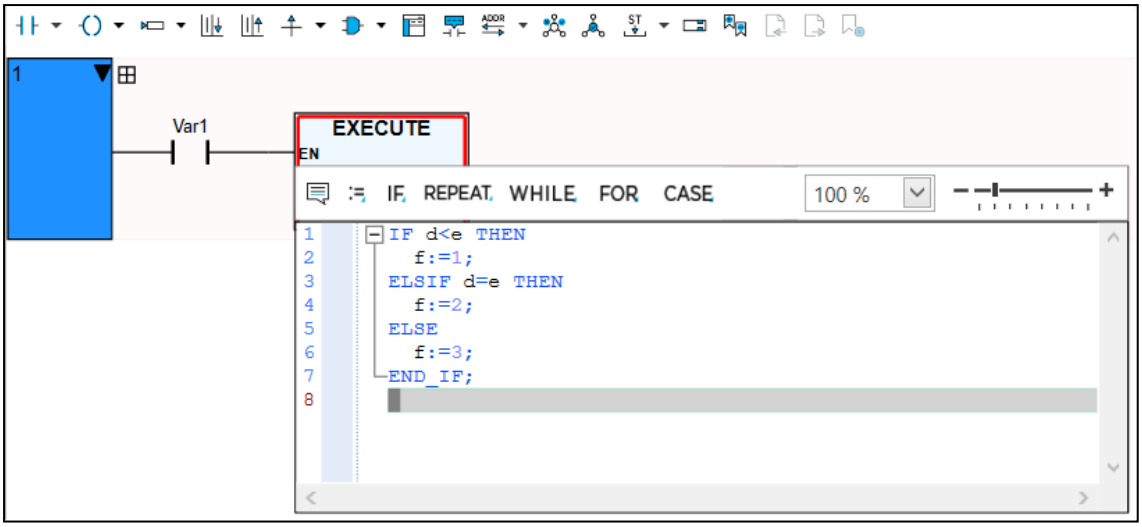
When writing the LD program in the ladder diagram POU, use this function to write the ST program at the same time and monitor and edit the execution block program online. Follow these steps to insert an execute block.

To insert execute block

1. On the toolbar, click the arrow icon next to , and then select **Insert Execute Block**.

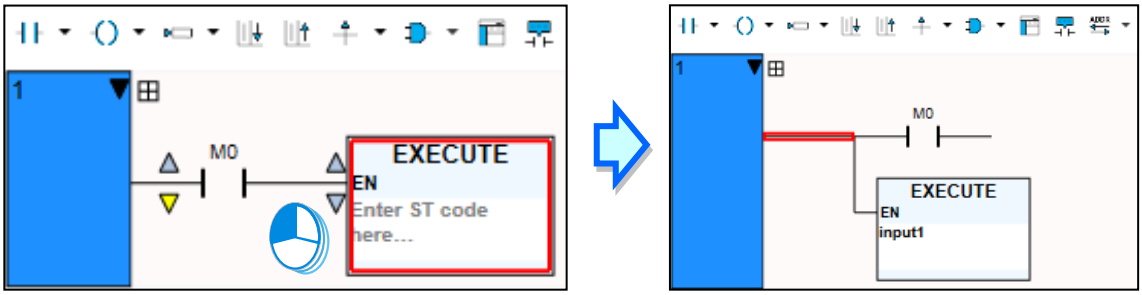


2. Click the input box of execute block (the same as the instruction editing dialog in ST POU) and enter the instruction.



After inserting an execute block, you can move it and resize it. Do the following.

- Press and drag the execute block to the desired arrow position.



- Press and hold the border of the execute block, and then drag it to the appropriate position and release the mouse to adjust the size of edit area.

For more information, refer to ST—Structured Text.

Note:

- The execute block is only supported in DIADesigner version 1.4.0 and later. Earlier versions of DIADesigner cannot open POU projects with execute blocks. You can update the DIADesigner to the latest version in DIAInstaller.
- The debug mode is not yet supported in execute blocks and cannot place break points. Break points can only be placed on objects in the LD POU.

Network Continuous Paste

In the ladder diagram, use the network continuous paste function to paste networks in batch and set the increment value for the element address in the network.

Following these steps to paste networks in batch.

To paste networks in batch

1. Right-click the network to be copied, and then select **Copy**.
2. Right-click the network to be pasted continuously, and then select **Network Continuous Paste**.
3. In the **Network Continuous Paste** dialog, you can:
   - Set the **Number of Continuous Pastings** to paste the networks continuously.
   - Set the **Increment Value**, the element address will increase according to the set increment value.

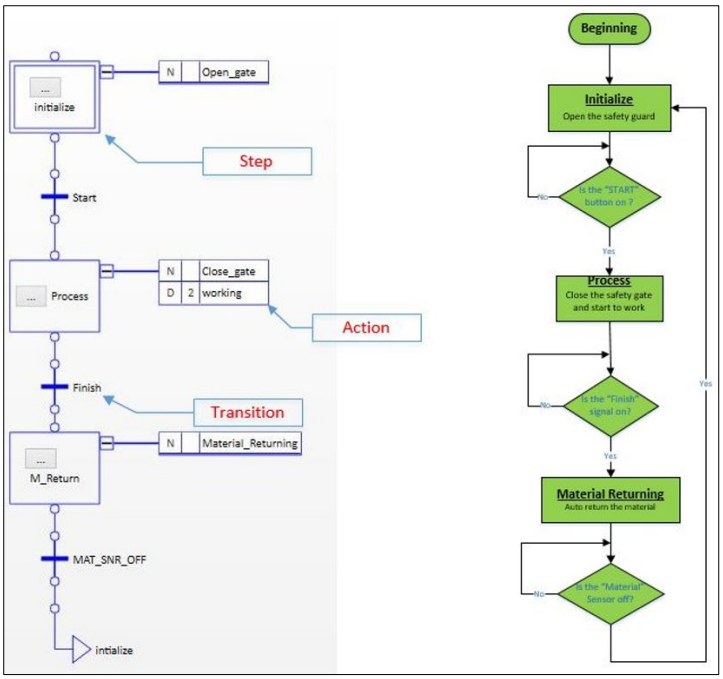
   You can also click **Increment Value Batch Setting** to set the increment value in batch.
4. Click **OK**.

   The copied network is pasted continuously according to the setting.

8.3.2 SFC—Sequential Function Chart

The **Sequential Function Chart (SFC)** is one of the five languages defined by IEC 61131-3 standard. It is a powerful graphical technique for describing the sequential behavior of a control program.

A Sequential Function Chart and its corresponding flowchart is shown in the following figure. The main components of a sequential function chart are the Steps with associated Actions, Transitions and associated Logic conditions and directed links between Steps and Transitions.

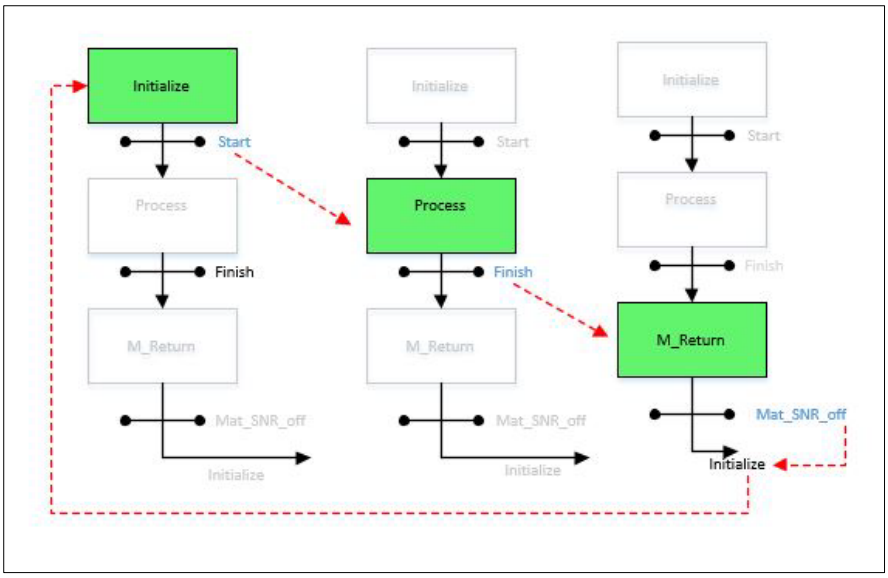


Principle of a Sequential Function Chart

A sequence in a Sequential Function Chart is depicted as a series of steps shown as rectangular boxes connected by vertical lines. Each **Step** represents a particular state of the system being controlled. A **Transition** is associated with a condition which, when true, causes the Step before the Transition to stop and allows the Transition to be activated.

Steps in a Sequential Function Chart can be active or inactive. Actions are only executed for active steps. If a Step is inactive, the Transition following it will not affect the execution of the program, and the actions associated with the Step will not be executed.

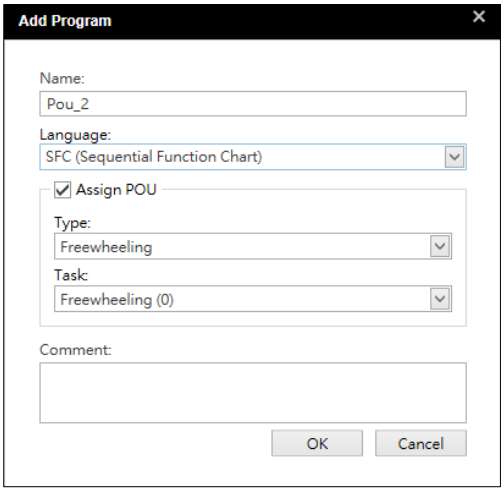
In a Sequential Function Chart in DIADesigner, the Steps must alternate with Transitions. If there are two continuous Steps or two continuous Transitions in a Sequential Function Chart, the program cannot be compiled.



The above example is the structure of a Sequential Function Chart. DIADesigner also provides other flexible programming functions, such as the use of divergent and convergent paths, the use of modular actions and transitions and the qualification of actions. These programming functions are introduced in the following sections.

8.3.2.1 Add SFC Program

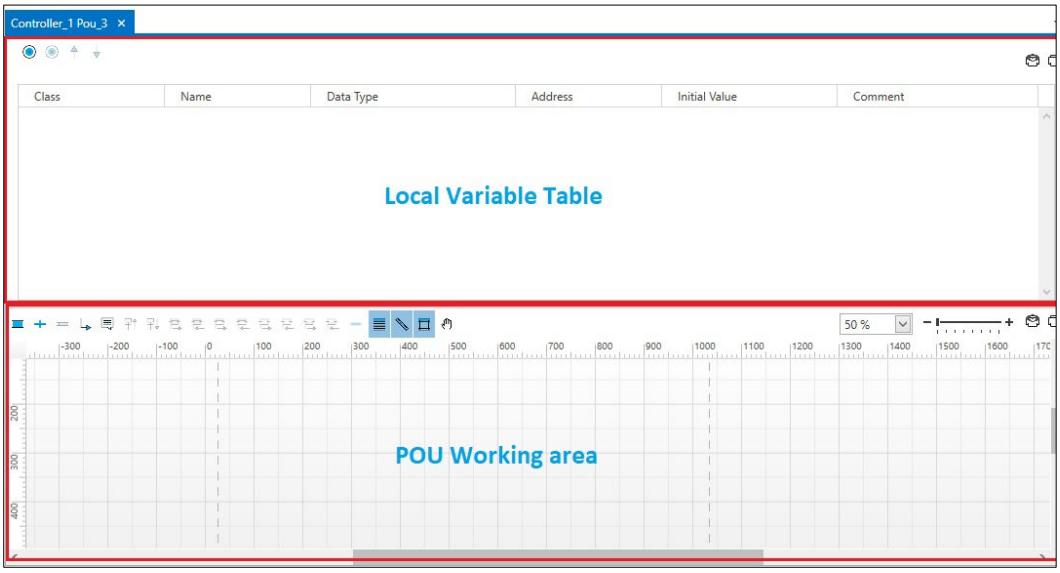
Select **SFC (Sequential Function Chart)** in the **Language** field in **Add Program** dialog while creating a POU of program type. Refer to POU Architecture and Types for detailed information.



The environment in which an SFC can be edited is shown in the following table. In any POU dialog (program type or function block type), there are two areas.

- Local Variable table area
- POU Working area

The table in the upper part of the SFC POU dialog is a Local Variable Table and the lower part is POU working area. After a program editing dialog to create an SFC POU is opened, the corresponding toolbar appears in the DIADesigner dialog.



SFC Toolbar

After the POU dialog is opened, the toolbar for editing the Sequential Function Chart appears as shown in the following figure.



The following table describes the functions in SFC POU dialog.

Function / Icon	Description
	Click to add a Step.
	Click to add a Transition.
	Click to add Action.
	Click to insert a Jump.
	Click to add Comment.
	Click to insert a Step and a Transition above.
	Click to insert a Step and a Transition below.
	Click to add a simultaneous Step to the right direction.
	Click to add a simultaneous Step to the left direction.
	Click to add a Step and a Transition to the right direction.
	Click to add a Step and a Transition to the left direction.
	Click to add a Transition to the right direction.
	Click to add a Transition to the left direction.
	Click to add a Divergence of Sequence Selection & Simultaneous Convergence to the right direction.
	Click to add a Divergence of Sequence Selection & Simultaneous Convergence to the left direction.
	Click to collapse Action List.

Function / Icon	Description
	Click to display / hide Grid.
	Click to display / hide Ruler.
	Click to display / hide Page Range.
	Click to switch to Drag mode.

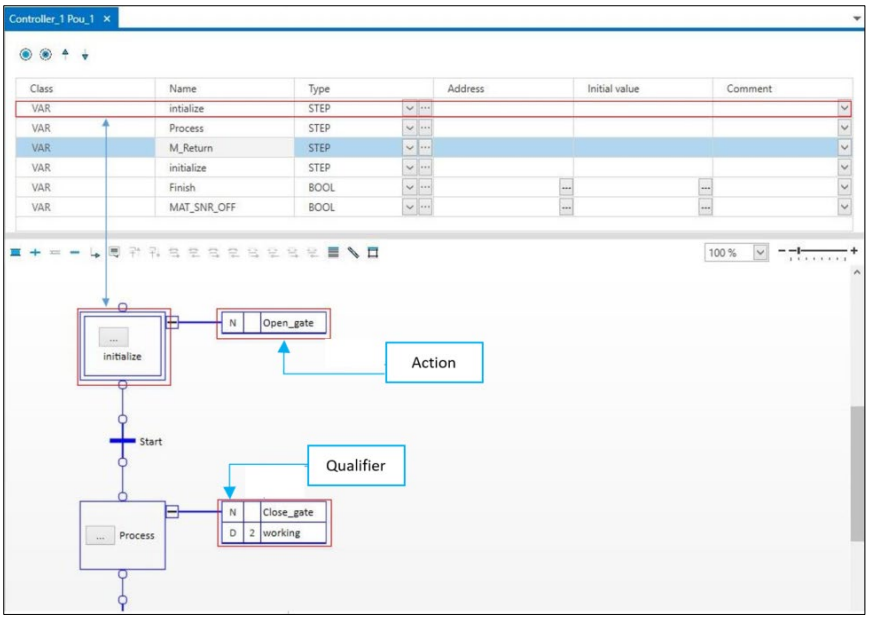
8.3.2.2 SFC Elements

SFC POU elements can be added from the toolbar to the POU working area. Arrange the position of each element by dragging in the working area, and the elements can also be dragged to create connections to form a logical relationship of SFC.

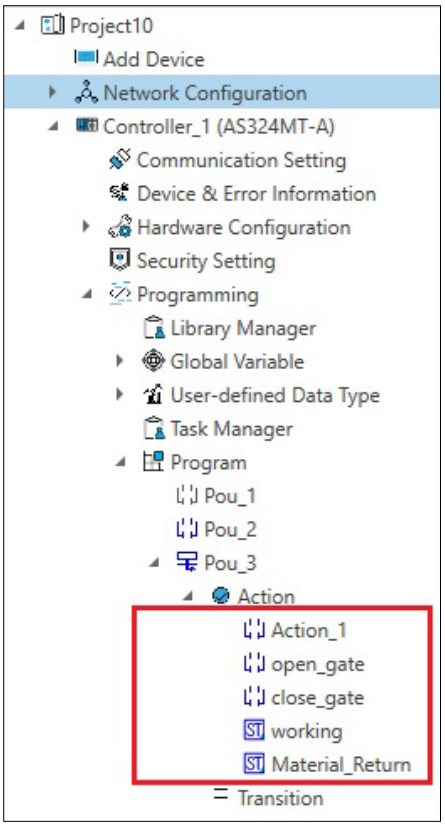
Steps and Actions

Every Step in DIADesigner must be assigned a variable whose data type is **STEP**, and functions as a state flag. If the flag corresponding to a Step is ON, the Step is active. Every variable in AS series PLC whose datatype is STEP occupies an **S** register in the PLC. Refer to the Operation Manual for more information about the S registers.

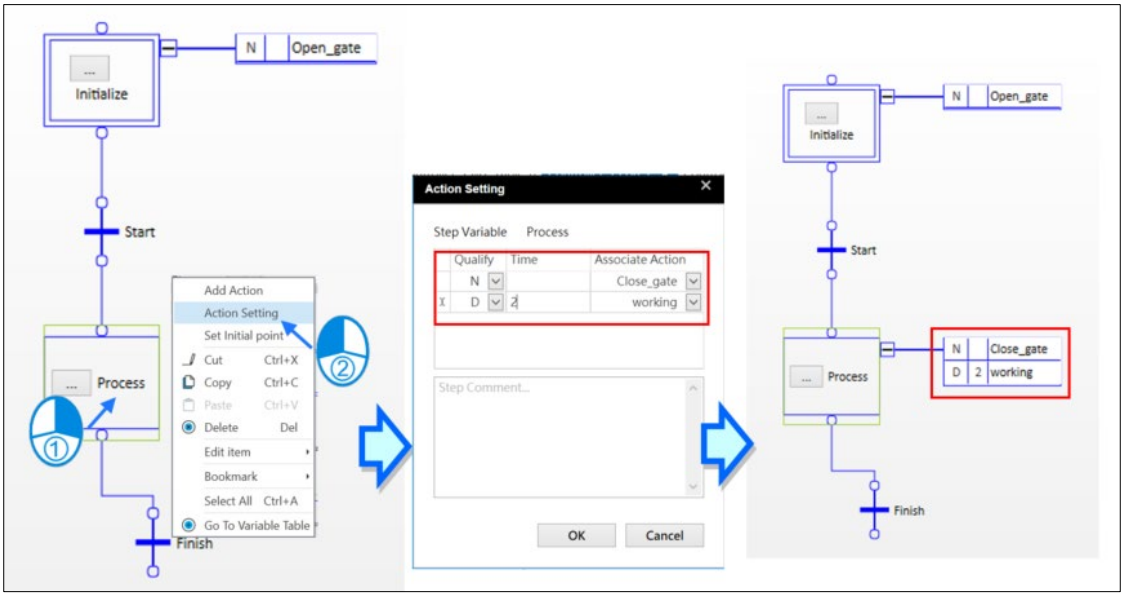
After a Step is activated, the actions associated with the Step are executed according to the qualifiers. In DIADesigner, the actions can be modularized, and can be associated with different Steps. A Step can be associated with more than one action. Besides, every action has a qualifier that determines when the action is executed.



In DIADesigner, an action is a program code, and can be developed by means of a ladder diagram, a structured text, etc. The actions which are created are listed in **Project Explorer**. The POU dialog in which an action can be created differs from the window for a programming language in that there is no local variable table in the POU dialog in which an action can be created. In DIADesigner, the actions and the transitions in a sequential function chart share the same local variable table.

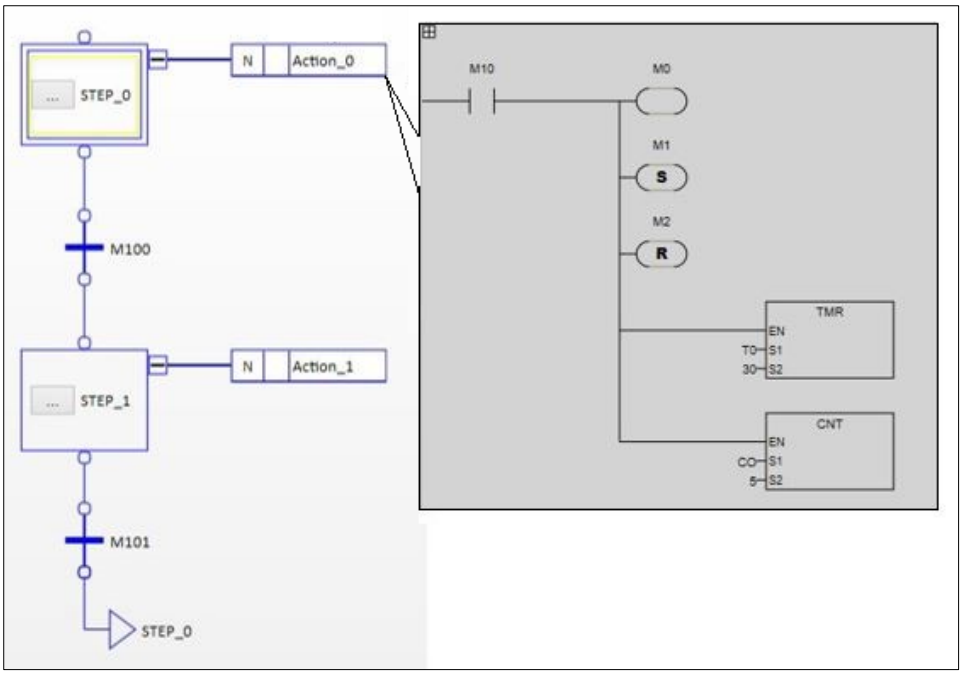


To edit a sequential function chart, firstly create actions, then assign the actions to steps and assign qualifiers to the actions. To perform this, right-click the step to be set, then click **Action Setting** in the context menu. In the **Action Setting** dialog, add actions and set qualifiers.

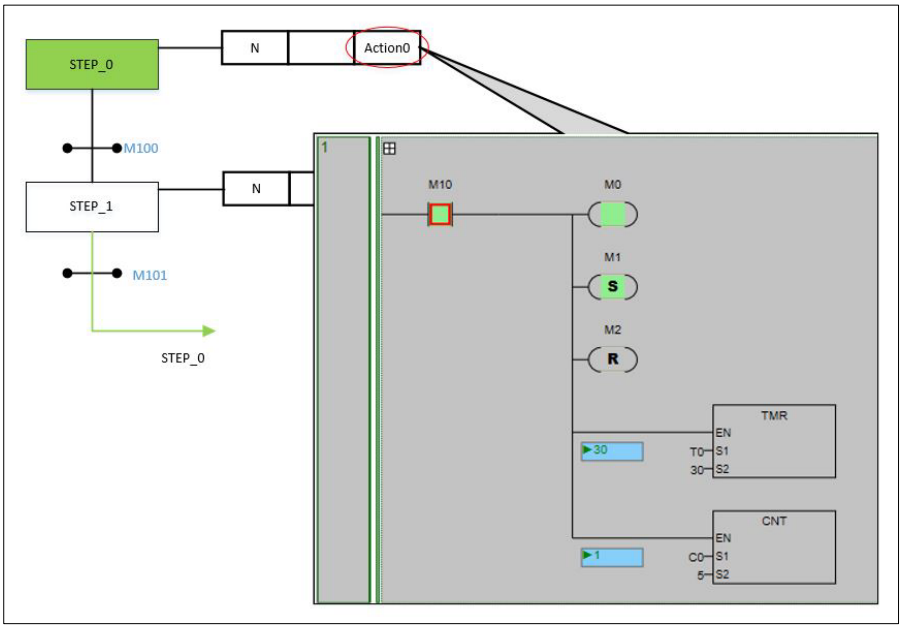


Final scans are important procedures during the execution of a sequential function chart. After the execution of an action stops, the system automatically executes a procedure to disable all the outputs in an action. If a final scan is executed, the coils driven by the instruction OUT is OFF, the application instructions and the function blocks will not be executed. The Timers are reset, the states of the coils driven by the instruction SET or RESET will remain unchanged, the Counters will stop counting, the states of the contact of the counters will remain unchanged, and the values in the counters will remain unchanged.

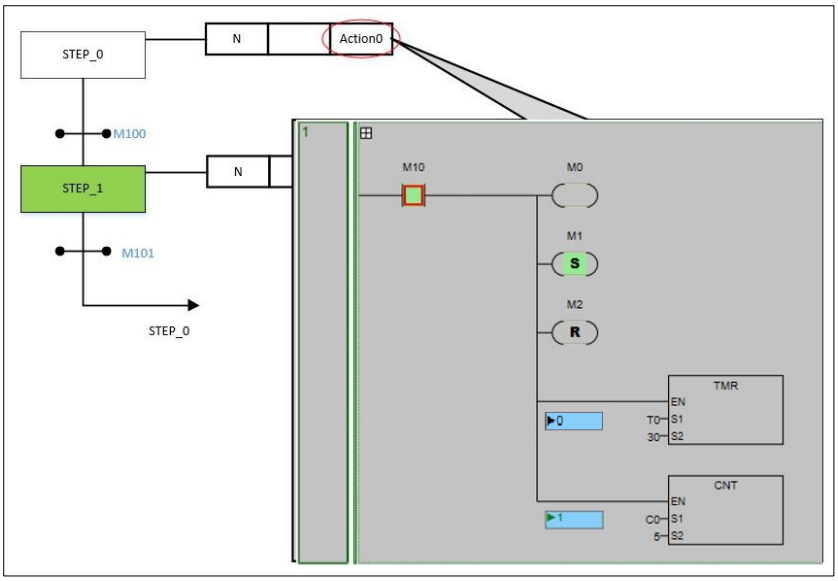
Action0 in the following figure is an action associated with STEP_0 and is an N action. When STEP_0 is deactivated and STEP_1 is activated, the system executes the final scan.



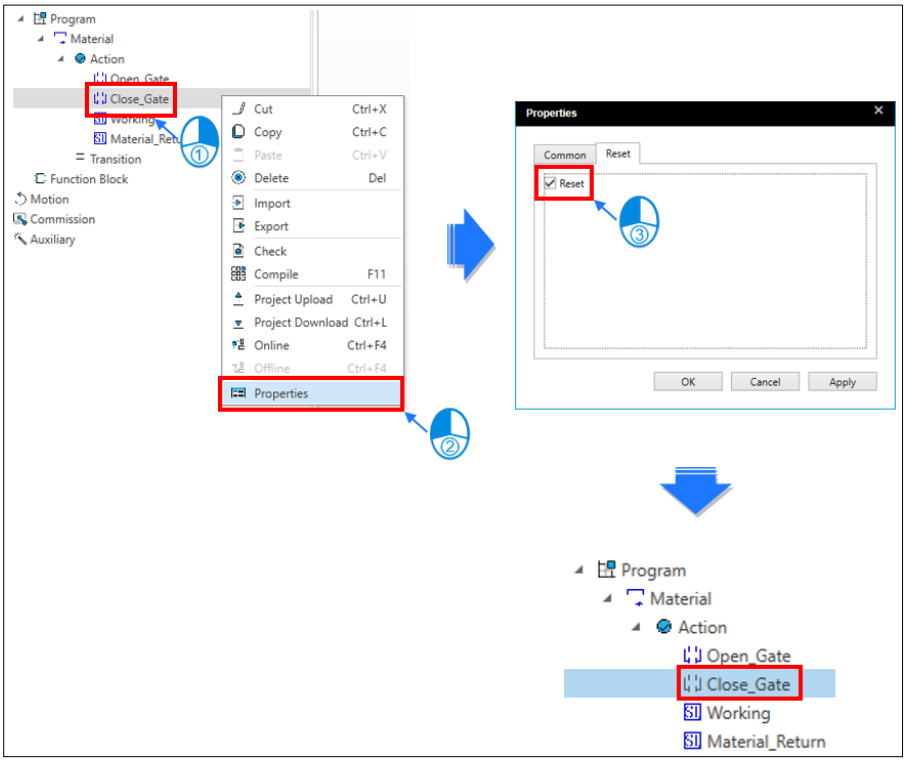
STEP_0 in the following figure is activated. M10 in Action0 is ON and therefore M0 is ON, M1 is set to ON, M2 is reset to OFF, the value in T0 is 30 and the value in C0 is 5.



STEP_1 in the following figure is activated. Although M10 is ON, M0 is OFF and T0 is reset. The state of M1, the state of M2 and the value in C0 remain unchanged.



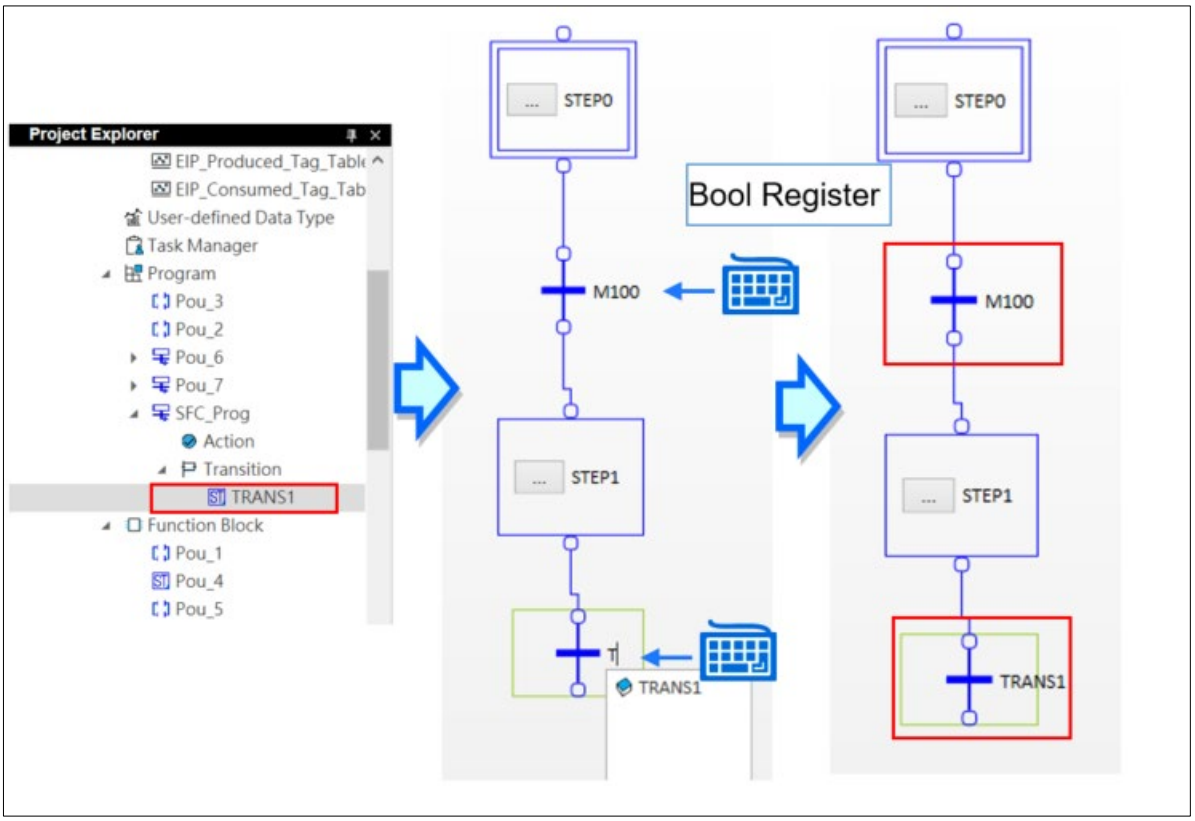
For AH5x1, AH560 backup series models, AHxxEMC and AS series models, you can set whether to execute "Final Scan" for each action. The default is to run the Final Scan in the DIADesigner. To disable this function, right-click the action that does not require a Final Scan to open its context menu and then uncheck the **Action/Transition** > **Reset**. After that, you may notice that the action that does not require a Final Scan is changed from blue color icon to gray color icon. The state of an action that does not run a Final Scan will stay the same as the previous state does, while the state of an action that runs a Final Scan is cleared.



Transitions

When a Transition is true, the active step immediately before this transition becomes inactive and the step immediately after this transition becomes active. A Transition can be true anytime, but for a transition to activate the step following it, the step preceding it must be active when the transition is true.

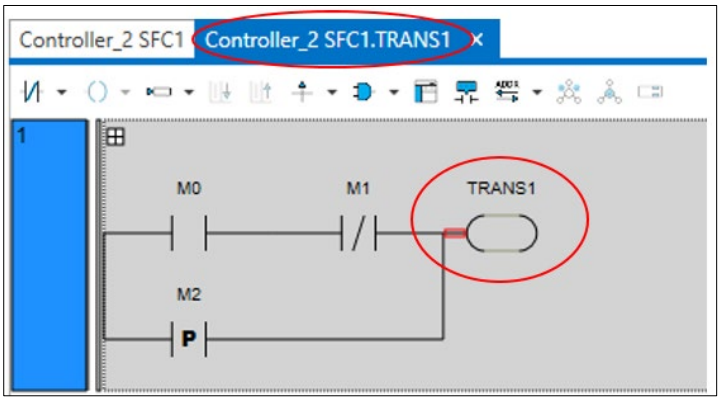
In DIADesigner, a Transition can be a Boolean register or variable or a program code. If a transition is a logical operation, you can create a transition program, and assign the program to the transition. During operation, click a transition to be set, then (1) use keyboard to enter Transition or (2) select Transition / Boolean register in the drop-down list as shown in the following figure.



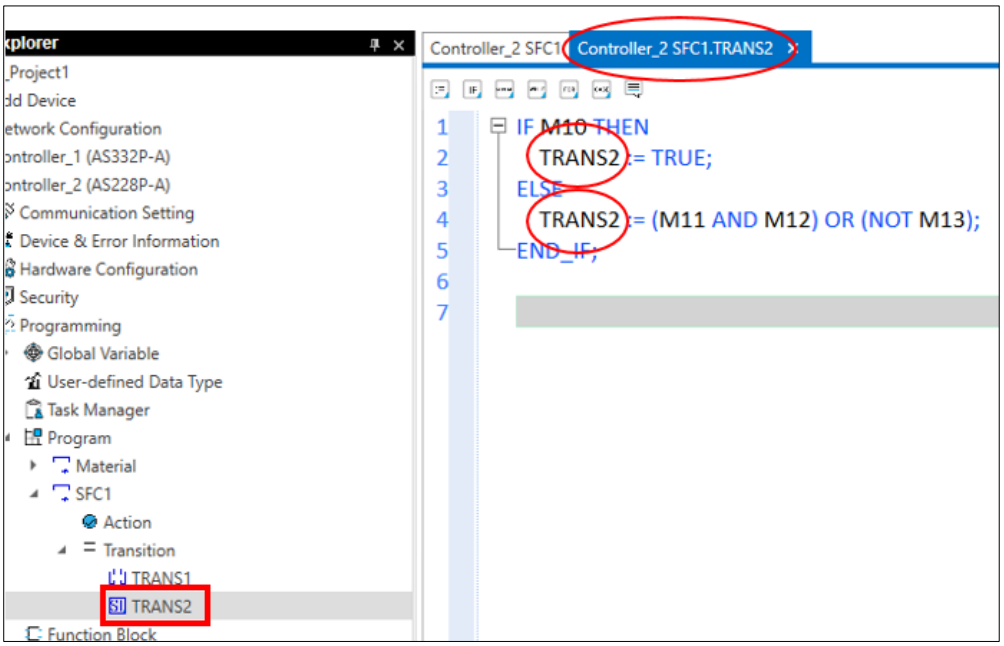
The programming languages which currently can be used to create transition programs include ladder diagrams (LD) and structured texts (ST). The Boolean state in a transition program must be sent to a variable name which is the same as the transition program name, whether the programming language used to create the transition program is a ladder diagram or a structured text. The variable need not be declared. However, if there is a variable name in a variable table which is the same as a transition program name, an error occurs when the program is compiled.

Besides, applied instructions and function blocks cannot be used in a transition program, but comparison contacts and block logic instructions (NP, PN, and INV) can be used in a transition program.

If the programming language used to create a transition program is a ladder diagram, the program code of the transition program must be composed of one network, and the output contact must be assigned a variable name which is the same as the transition program name. Besides, multiple outputs are not allowed in a transition program created by means of a ladder diagram.



If the programming language used to create a transition program is a structured text, there is no limit on the number of lines of program code. However, a Boolean state in a transition program must be assigned to a variable name which is the same as the transition program name.



When creating a transition program, a variable name which is the same as the transition program name is added to the program code of the transition program. You only must create a conditional operation. Because the actions and the transitions in a sequential function chart share the same local variable table, there are not local variable tables in the program editing windows for the transitions.

You cannot create a complex transition program. If a transition condition must undergo a complex operation, you can create the complex operation in an action associated with a step, create a condition flag in the action, and assign the flag to the transition following the step.

Although you do not need to declare the variables in the transition programs, the system assigns memories which function as flags to the variables. As a result, make sure that the variables in the transition programs can be assigned a specific logic state. Otherwise, an error will occur if the program is executed.

Take the transition program which is created by means of a structured text below for example.

During the execution of the transition program for the first time, TRANS4 is ON if the result of (M0 AND M1) is ON. Since TRANS4 is ON, it passes control to the next step. After the control is passed to the step preceding the transition program again, the transition program is executed again. If the result of (M0 AND M1) is OFF, the statement following THEN will not be executed. TRANS4 will still be ON because it is not updated. The system will evaluate the transition condition as TRUE, and the control is passed to the next step.

Adding an **ELSE** section to the transition program in the following figure is a better way to ensure that the variable TRANS4 is assigned a specific logic state. You must pay attention to the same problem in a CASE structure.

```
1  IF (M0 AND M1) THEN
2      Tran4 := TRUE;
3  END_IF;
```

Simultaneous Divergence and Divergence of Sequence Selection

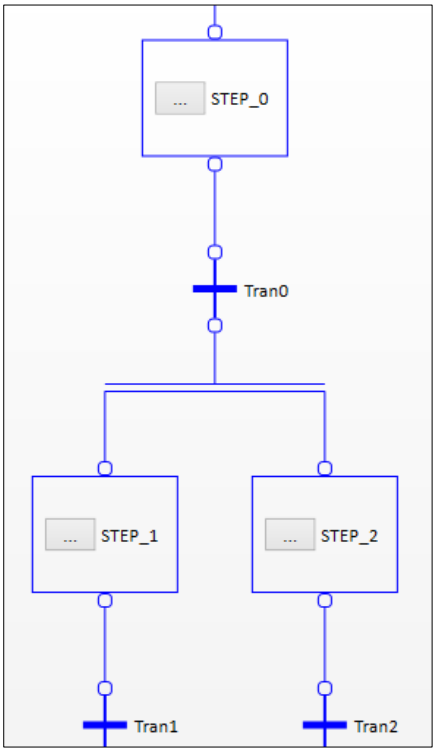
You can create divergent paths in a sequential function chart. There are two types of divergences.

- Simultaneous divergence
- Divergence of sequence selection

Simultaneous divergence

A Simultaneous divergence is shown in the following figure. The divergence steps are connected by a double line. The characteristics of a simultaneous divergence is that before entering the divergent steps, there is a shared transition above. When the transition is true, the steps are active.

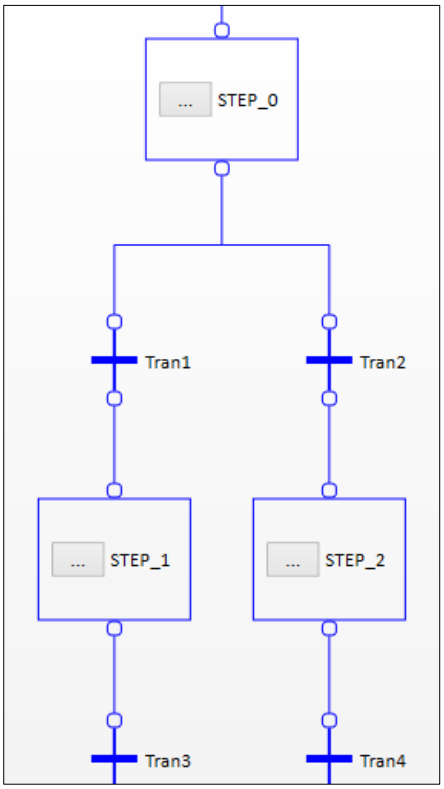
When Trans0 is true, STEP_0 becomes inactive and STEP_1 and STEP_2 becomes active.



Divergence of Sequence Selection

The characteristics of divergence of sequence selection is that the divergent steps have their own transitions above. When one transition is true, the step following the transition becomes active and the other transitions are not evaluated. Consequently, if there are divergent paths, only one path is taken at a time. Besides, the transitions are evaluated from left to right. If several transitions are true simultaneously, only the step following the left-most transition is activated.

When the Tran2 is true, STEP_0 becomes inactive and STEP_2 becomes active. Trans1 is not evaluated and STEP_1 remains inactive. The following figure shows a Divergence of Sequence Selection.



Simultaneous Convergence and Convergence of Sequence Selection

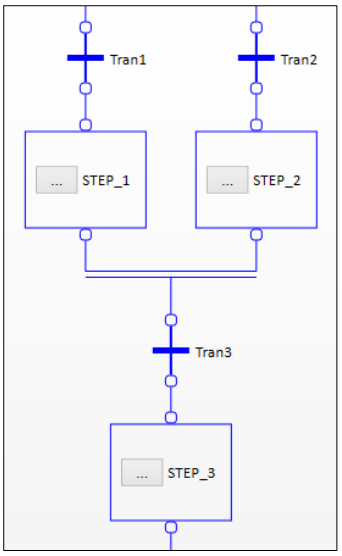
The divergent paths in a SFC must be converged. There are two types of convergences.

- Simultaneous Convergence
- Convergence of Sequence Selection

Simultaneous Convergence

A Simultaneous Convergence is shown in the following figure. The convergent steps are connected by a double line. The characteristics of a simultaneous convergence are that the convergent steps are connected to the same transition below. The transition will not be evaluated until the convergent steps preceding the transition are activated.

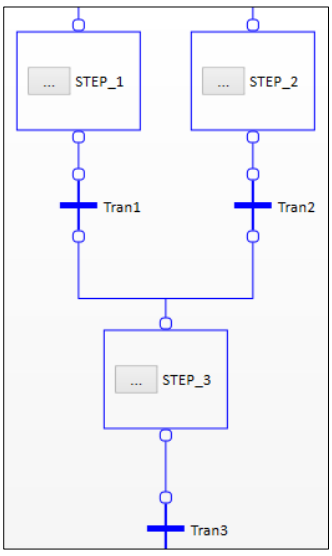
After STEP_1 and STEP_2 are activated, Tran3 is evaluated. When Tran3 is true, STEP_1 and STEP_2 becomes inactive and STEP_3 becomes active.



Convergence of Sequence Selection

The following figure shows a convergence of sequence selection. The characteristics of a convergence of sequence selection is that the convergent steps have their own transitions. When one transition is true, the next step become active. If there is another convergent step which is activated, the step is active until the transition following the step is true. When the transition is true, the step following the transitions becomes active. Consequently, if a convergence of sequence selection is used in an SFC, several steps following the convergence may be activated. If you use a convergence of sequence selection in an SFC, you must make sure that the program is not confused.

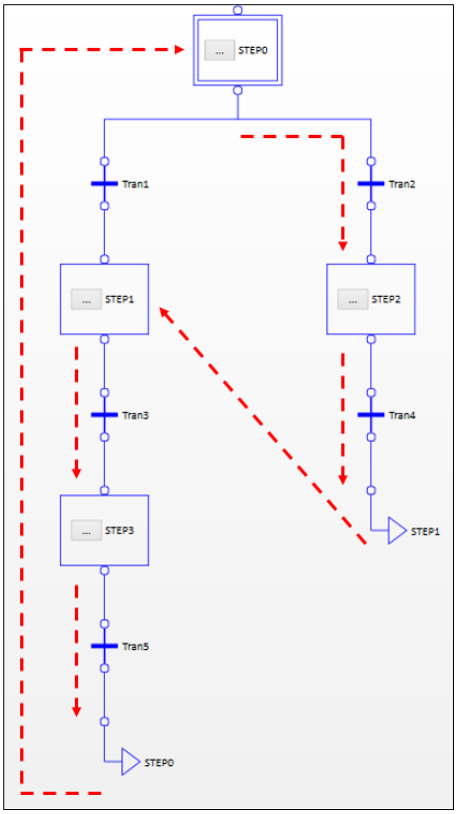
When Tran1 is true, STEP_1 becomes inactive and STEP_3 becomes active. If Tran2 is true, STEP_2 is inactivated and STEP_3 is activated again.



Jump

The use of a jump structure in an SFC results in the passing of control to a step which is specified. There is at least one jump point indicating that the passing of control will occur once the transition preceding the jump point is true in an SFC. You can use jump structures quite freely. If you use a jump structure in an SFC, you must make sure that the program is not confused.

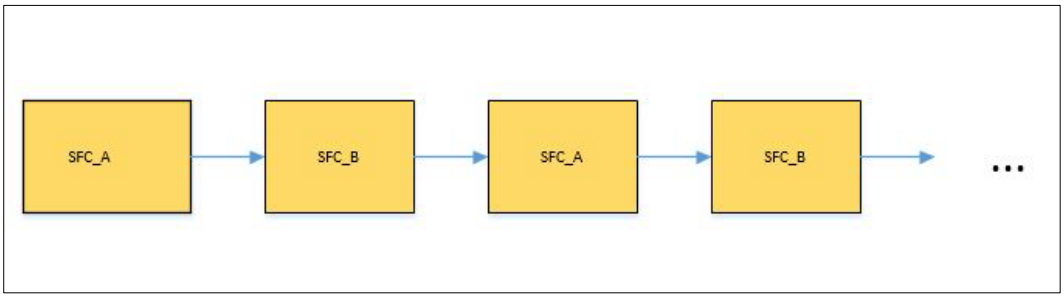
The program in the following figure is an SFC where there is a divergence of sequence selection. If Tran2 is true, control is passed to STEP2. When Tran4 is true, it passes control to STEP1 through a jump. When Tran5 is true, it passes control to STEP0.



If you use a jump structure in an SFC, the step specified must be a step in the same SFC. If the step specified is not in the same SFC and is declared in a local variable table or the global variable table, an error occurs when the program is compiled.

If control needs to be passed from a step in an SFC to a step in another SFC, you can make use of the characteristics of steps and the characteristics of transitions. You can declare the step or the transition which is the destination jump in the global variable table. When the condition of the jump is met, the step or the transition declared is set to ON. The condition of the jump can be created in an action.

The following figure shows that SFC_A is executed first. SFC_B is executed after SFC_A is executed and SFC_A is executed after SFC_B is executed. You can achieve the jumps by means of the characteristics of transitions.

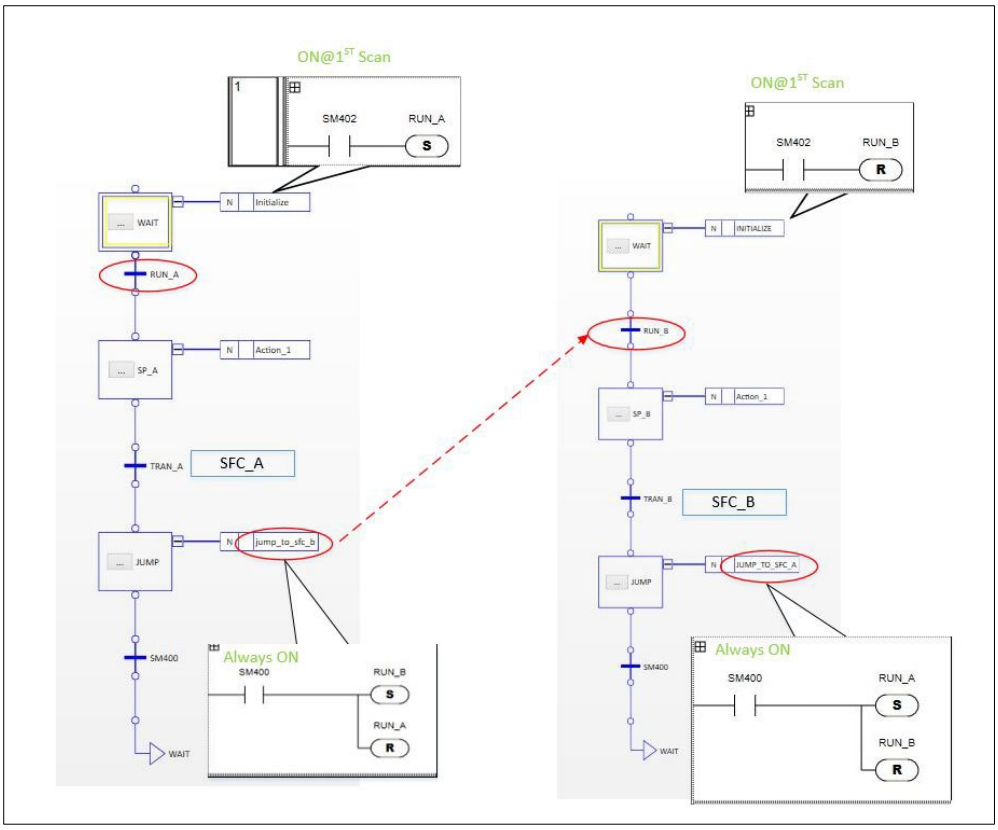


The steps SP_A and SP_B in the following example are used to express the sequential function chart which is executed. They are preceded and followed by steps. Besides, the transitions RUN_A and RUN_B are declared in the global variable table.

When the program is executed, the step WAIT in SFC_A and the step WAIT in SFC_B are activated, and the action INITIALIZE associated with WAIT in SFC_A and the action INITIALIZE associated with WAIT in SFC_B are executed. INITIALIZE in SFC_A sets the transition RUN_A to ON during the first scan cycle, and therefore control is passed to SP_A. INITIALIZE in SFC_B sets the transition RUN_ B to OFF during the first scan cycle, and therefore control is not passed to SP_B.

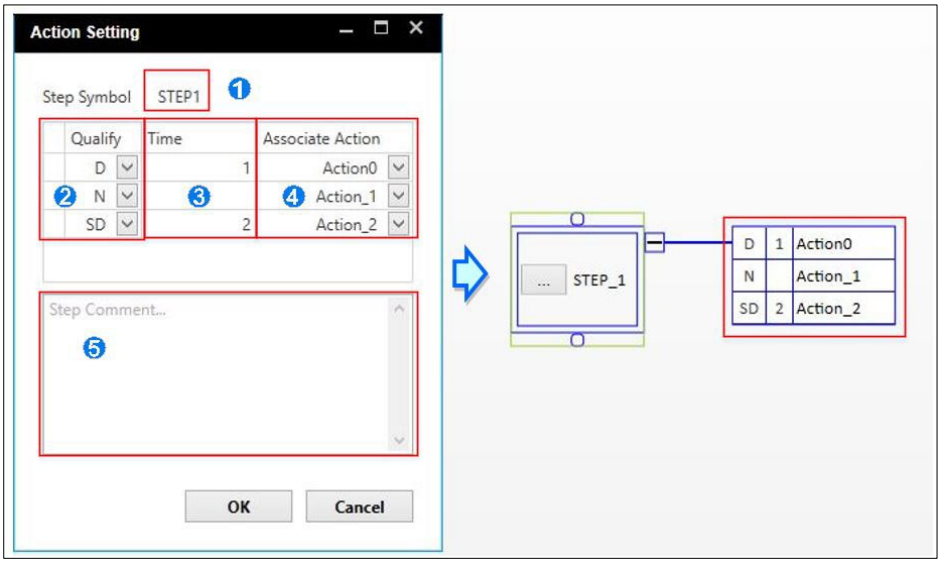
When the step JUMP in SFC_A is activated, the action associated with JUMP sets the transition RUN_B to ON, and resets RUN_A to OFF. When RUN_B is set to ON, control is passed to the step SP_B. Besides, the transition following JUMP in SFC_A is a special relay which is always ON, and therefore control is passed to the step WAIT. Since RUN_A is set to OFF, control is not passed to SP_A. RUN_A will not be set to ON until the action associated with JUMP in SFC_B is executed.

The use of WAIT and JUMP in SFC_A serves to set RUN_B to ON, and the use of WAIT and JUMP in SFC_B serves to set RUN_A to ON. WAIT and JUMP in SFC_A and WAIT and JUMP in SFC_B have little significance. If the program is applied to an AS series CPU module, you can assign the qualifier P or P1 to the actions associated these steps.



Set Qualifier of Action

In DIADesigner, you can define the execution of an action by qualifying the action.



1. Displays the step name.
2. Select qualifiers for the actions associated with the step.
3. Select qualifiers related to time.

❹ Select actions which are qualified.

❺ Make comments on the step.

Note: In the SFC sequential function diagram, for AS series models, the unit is ms.

Qualifier Types

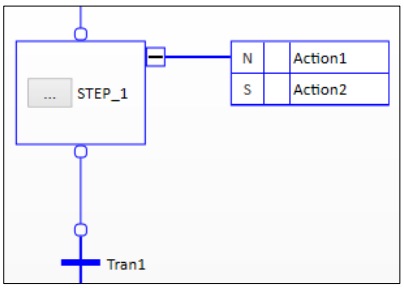
N (Normal)

When a step is activated, the N action associated with the step is executed. When the step is inactivated, the system executes a final scan to disable all the outputs in the N action. However, the outputs which use the instructions similar SET are not disabled.

S (Set)

When a step is activated, the S action associated with the step is executed. Even if the step is inactivated, the S action will still be executed.

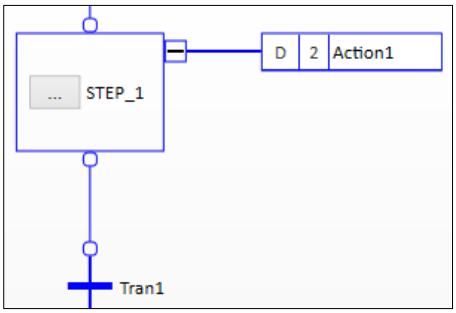
When STEP_1 in the following figure is activated, Action1 and Action2 are executed. When Tran1 is true, STEP_1 becomes inactive. Once STEP_1 is inactivated, the execution of Action1 will stop, the system will execute a final scan, and Action2 will still be executed.



D (Delay)

The D action associated with a step is not executed at the time when the step is activated. The execution of the D action is delayed for a certain amount of time. If the transition following the step becomes true before the D action is executed, the step is inactivated, control is passed to the next step, and the D action is not executed. When the step is activated again, the delay is measured again.

Action1 in the following figure is not executed at the time when STEP_1 is executed. The execution of Action1 is delayed for two seconds. If STEP_1 is inactivated before Action1 is executed, Action1 will not be executed.



SD (Set Delay)

The SD action associated with a step is not executed at the time when the step is activated. The execution of the SD action is delayed for a certain amount of time. If the SD action is executed, it will still be executed after the step is inactivated. Besides, if the transition following the step becomes true before the SD action is executed, control is passed to the next step, and the delay will still be measured. A certain amount of time has elapsed before the SD action is executed.

DS (Delay Set)

A DS action is similar to a SD action. However, if the transition following a step becomes true before the SD action associated with the step is executed, the delay will not be measured, and the SD action will not be executed. When the step is activated again, the delay is measured again. If the DS action is executed, it will still be executed after the step is inactivated.

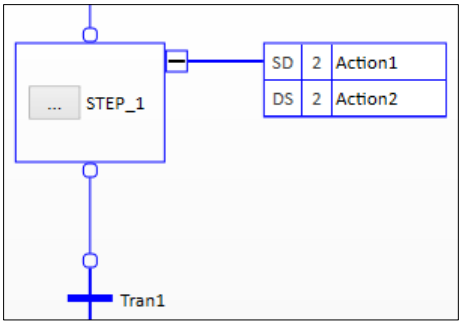
Action1 and Action2 are not executed at the time when STEP_1 is activated. The execution of Action1 and the execution of Action2 are delayed for two seconds. There are two situations which need to be considered.

a) STEP_1 is activated for more than two seconds.

Action1 and Action2 are executed after STEP_1 has been executed for two seconds. Even if STEP_1 is inactivated, Action1 and Action2 will still be executed.

b) STEP_1 is activated for less than two seconds.

After control is passed to the next step, the delay of Action1 will still be measured. Two seconds has elapsed before Action1 is executed. However, after the control is passed to the next step, the delay of Action2 will not be measured, and Action2 will not be executed.



L (Limit)

When a step is activated, the L action associated with the step is executed. If the L action has been executed for a certain amount of time before control is passed to the next step, the execution of the L action will stop automatically, and the system will execute a final scan. If the control is passed to the next step before the time specified has elapsed, the execution of the L action will stop.

SL (Set Limit)

When a step is activated, the SL action associated with the step is executed. If the SL action has been executed for a certain amount of time before control is passed to the next step, the execution of the SL action will stop automatically, and the system will execute a final scan. If the control is passed to the next step before the time specified has elapsed, the execution of the SL action will stop after the time specified has elapsed.

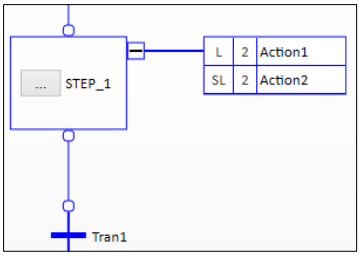
When STEP_1 is activated, Action1 and Action2 are executed. There are two situations which need to be considered.

a) STEP_1 is activated for more than two seconds.

After Action1 and Action2 have been executed for two seconds, the execution of Action1 and the execution of Action2 will stop, and the system will execute final scans.

b) STEP_1 is activated for less than two seconds.

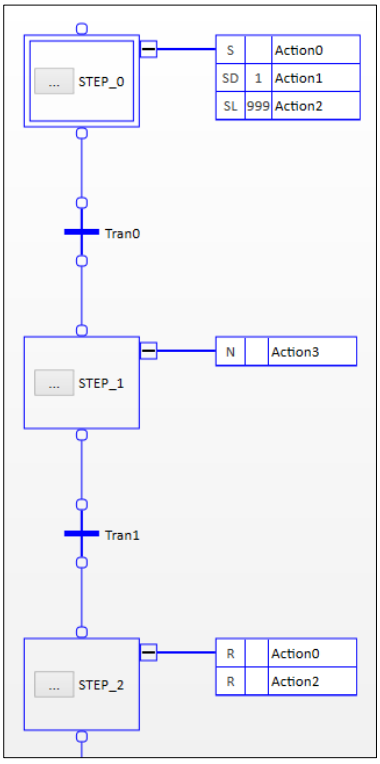
After control is passed to the next step, the execution of Action1 will stop, and the system will execute a final scan. The execution of Action2 will stop after two seconds have elapsed.



R (Reset)

When a step is activated, the R action related to the step is not executed, and the system executes a final scan. An action qualified by S, SD, DS, or SL will not be executed if it is qualified by R.

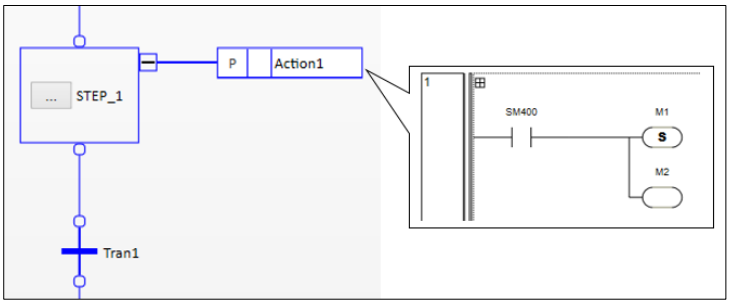
Suppose that Action0, Action1, and Action2 are executed during the activation of STEP_0, and is executed until control is passed to STEP_2. After the control is passed to STEP_2, the execution of Action0 and the execution of Action2 will stop, and the system will execute final scans. Since Action1 is not qualified by R, it will still be executed after the control is passed to STEP_2.



P (Pulse)

During the first scan cycle, the P action associated with a step is executed when the step is activated. During the second scan cycle, even if the step is activated, the outputs are disabled, and the system will execute a final scan.

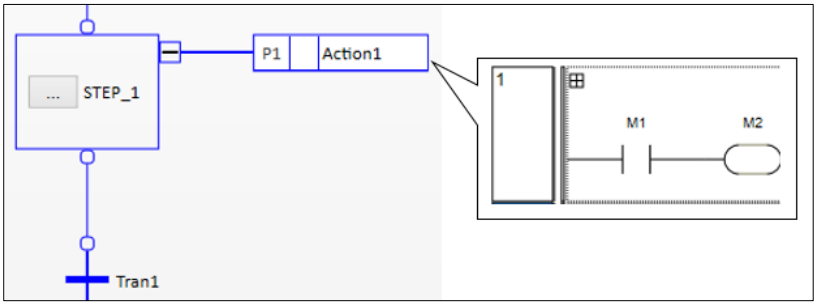
During the first scan cycle, Action1 is executed when STEP_1 is activated. Since SM400 is ON, M1 and M2 are ON. During the second scan cycle, the system inactivates Action1. Although SM400 is still ON, M2 becomes OFF. M1 is driven by the instruction SET, and therefore it remains ON.



P1 (Raising Pulse)

The P1 action associated with a step is executed only when the step is activated during the first scan cycle. Besides, the system will not execute a final scan after the execution of the P1 action stops. Consequently, although an action where the instruction OUT or TMR is used is allowed to be qualified by P1, a warning appears when the program is compiled.

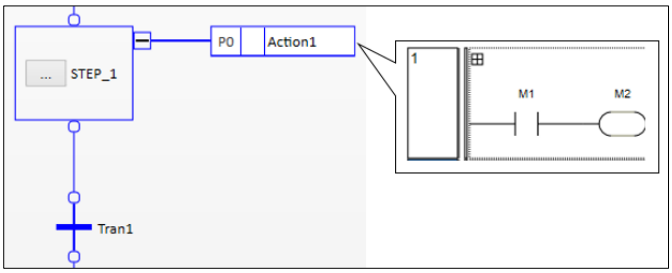
During the first scan cycle, Action1 is executed when STEP_1 is activated. If M1 is ON, M2 is ON. Action1 is executed only when STEP_1 is activated during the first scan cycle. Besides, the system will not execute a final scan after the execution of Action1 stops. Consequently, the state of M2 will not change even if the state of M1 changes.



P0 (Falling Pulse)

The P0 action associated with a step is executed only when the step is inactivated during the first scan cycle. Besides, the system will not execute a final scan after the execution of the P0 action stops. Consequently, although an action where the instruction OUT or TMR is used is allowed to be qualified by P1, a warning appears when the program is compiled.

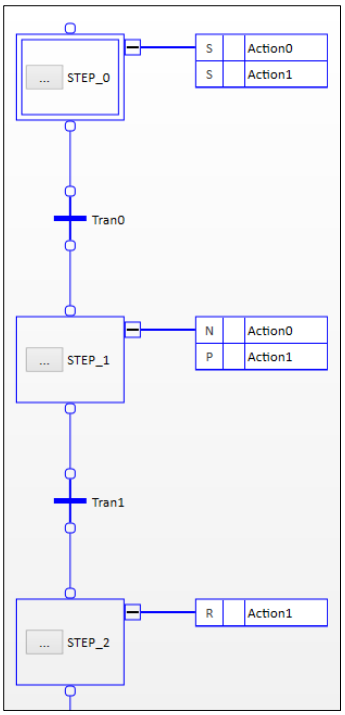
When STEP_1 is activated, Action1 is not executed. Action1 will not be executed until Tran1 is true. If M1 is ON, M2 is ON. Action1 is executed only when STEP_1 is inactivated during the first scan cycle. Besides, the system will not execute a final scan after the execution of Action1 stops. Consequently, the state of M2 will not change even if the state of M1 changes.



Important Points about Qualifying an Action

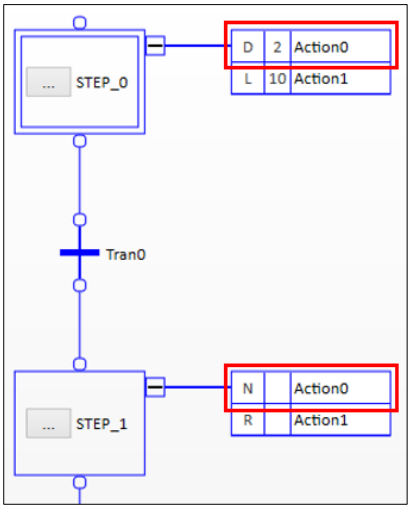
Once an action qualified by S is executed, it will still be executed even if it is qualified by N or P. The execution of the action stops only when the action is qualified by R. If there is an action which is qualified by S, you must check whether to assign R to the action to stop the execution of the action.

When STEP_0 is activated, Action0 and Action1 are executed. After control is passed to STEP_1, Action0 and Action1 will still be executed even if they are qualified by N and P respectively. The execution of Action1 will stop after control is passed to STEP_2. Action0 is never qualified by R, and therefore it is executed all the time.



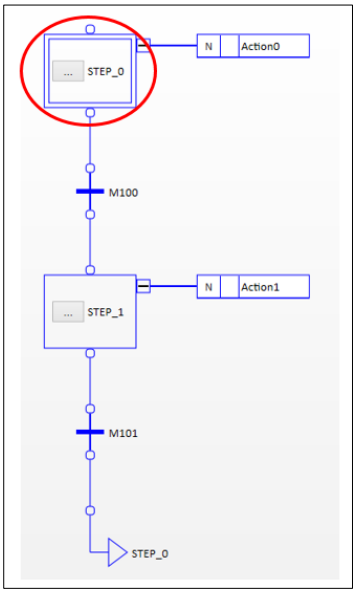
Except the R qualifier, when assigning a time-related qualifier for an Action, the qualifier can only be used in the SFC instruction once. In other words, if an action is qualified by D, SD, DS, L, or SL, no other qualifiers can be used for the action, except the R qualifier.

In the following example, we can see that in the STEP_0, the Action0 and Action1 are qualified by D and L respectively. And in the STEP_1, the Action0 and Action1 are qualified by N and R. The usage of Action 0 being assigned with D in the STEP_0 and N in the STEP_1 is wrong and will cause error to occur. The usage of Action 1 being assigned with Lin the STEP_0 and R in the STEP_1 is correct, since R can be added to an Action with more than 1 qualifier.



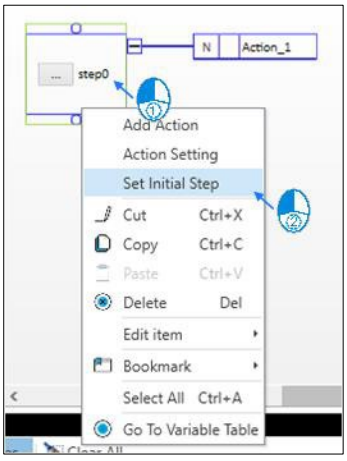
Initial Step

During the execution of a sequential function chart, every step is activated in turn and cyclically. It is necessary to define a step as an initial step which is activated when the program is executed for the first time. In DIADesigner, the initial step in a sequential function chart is drawn as a box with a double line. Besides, there is only one initial step in a sequential function chart.



Any step in a sequential function chart in an AS series CPU module can be defined as the initial step in the sequential function chart. When a sequential function chart in an AS series CPU module is executed for the first time during the operation of the CPU module, the initial step in the sequential function chart is the step which is activated first.

Right-click a step that you want it to be as the initial step and select **Set Initial Point** to complete the setting as shown in the following figure.

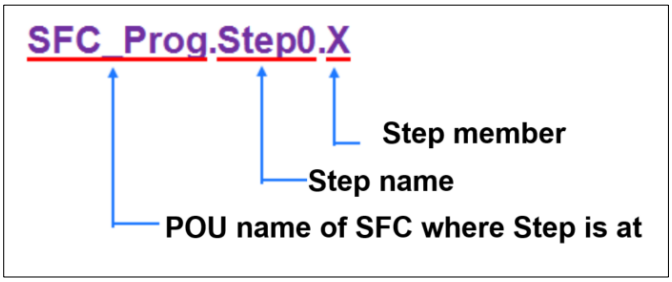


8.3.2.3 SFC Internal Properties

To make it easier for users to use certain property parameters of SFC while programming, internal properties are available for you to call among POUs. The property parameters can be divided into three categories: Step, Action, and Transition, which will be introduced in the following sub-sections.

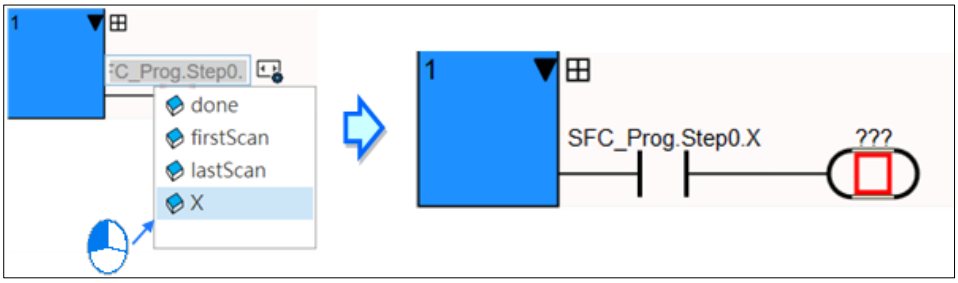
Step Property Parameter

If you need to call Step property parameters in the POU of SFC. Property parameter is presented in the form of the structure as shown in the following figure. The structure is composed of “POU name of SFC where Step is at”, “Step name”, and “Step member”. Select appropriate POU and Step names first, and then decide the needed Step member.



Type	Member	Description	Read / Write
Step	X	When Step is running, the SFC property parameter will be ON, and this SFC property parameter usage will replace the past S register.	Read only
	done	When **T** property of Step (currently, Step has executed time) >= **timePreset** property of Step, the value will be ON.	Read only
	counter	When **X** property of Step changes from OFF to ON, the property parameters count adds one.	Read only
	timePreset	Determine the length of time that **done** property of Step changes from OFF to ON. (Unit: **ms**; data length: **DWORD**)	Read only
	maxTime	Record T property of Step, the Max. time that has ever been timed. (Unit: **ms**; data length: **DWORD**)	Read only
	T	How long has Step been executing. (Unit: **ms**; data length: **DWORD**)	Read only

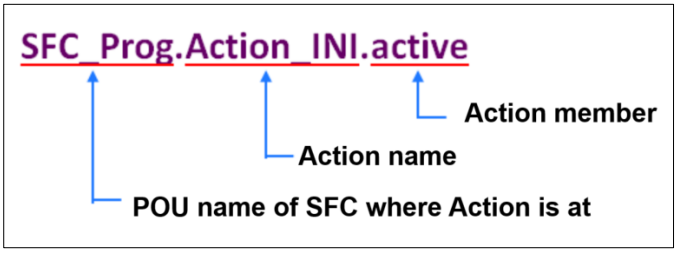
Take the member X of Step in SFC as example: To start execution at Step 1 in POU SFC_Prog of SFC to trigger a normally open contact in POU of LD program as ON, the program description will be as follows. Use a normally open contact in POU of the LD program, click above the contact to open the drop-down list, select the POU name of SFC program, Step name, and the name of member property parameter, and finish the selection.



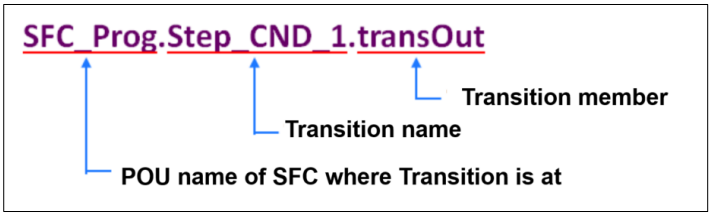
After that, when Step1 in POU of SFC program is running, the normally open contact, SFC_Prog.Step1.X, of the LD program will be ON.

Action and Transition Property Parameters

If you need to call Action or Transition property parameter in the POU of SFC. Property parameters are presented in the form of the structure as shown in the following figure. The Action structure is composed of “POU name of SFC where Action is at”, “Action name”, and “Action member”. Select appropriate POU and Action names first, and then decide Action member.



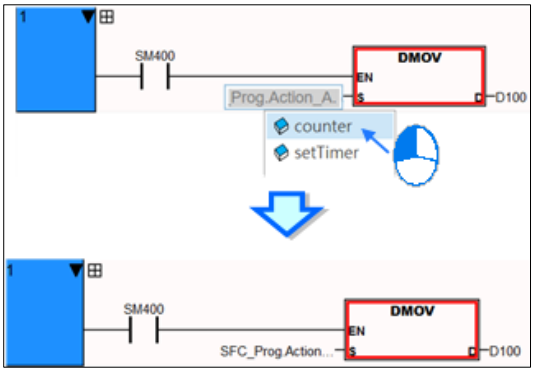
The Transition structure is composed of “POU name of SFC where Transition is at”, “Transition name”, and “Transition member”. Select appropriate POU and Transition names first, and then decide Transition member.



The following table shows different members of Action and Transition and their functions.

Type	Member	Description	Read / Write
Action	enabled	When qualifier of Action is TRUE, this SFC property parameter is ON, and you can tell if Action is enabled.	Read only
	active	When the first Action execution is complete, the SFC property parameter will be ON.	Read only
	rstDisable	When the Final scan function is enabled, the value is OFF. When the Final scan function is disabled, the value is ON.	Read only
	counter	When the **active** property of Action changes from OFF to ON, the property parameter count adds one.	Read only
	setTimer	Time display of the time-related qualifier. (Unit: **ms**; data length: **DWORD**)	Read only
Transition	transOut	When the Transition condition is True: ON When Transition is forced on: ON	Read only

Take the member counter of Action in SFC as example: To monitor the number of times the active property parameter of Action changes from 0 to 1 and send this active property parameter to D100 register, the program description will be as the following. Use SM400 normally open contact and SMOV instruction in the POU of LD program, click the S pin of DMOV, select POU SFC_Prog of SFC program, Action name, the name of member property parameter, and finish the selection. After that, enter D100 in the D pin of the DMOV instruction.



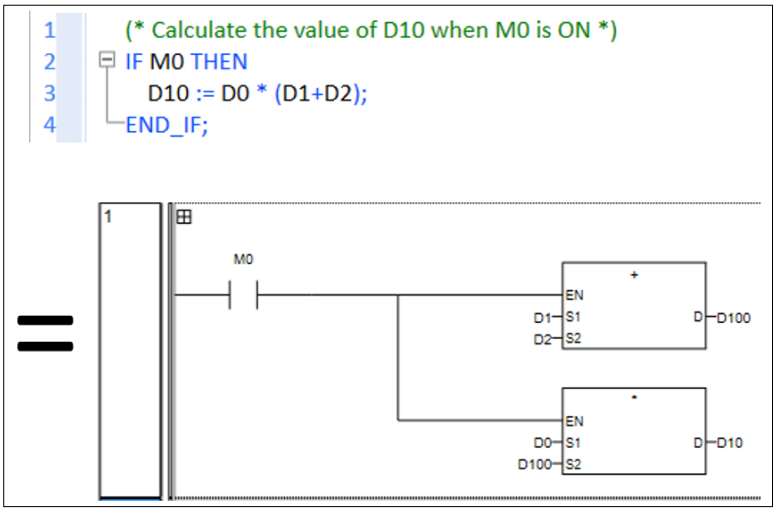
After that, when the program is running, every time when the FC_Prog.Action.Act_1.Active action is complete, the SFC_Prog.Action.Act_1.counter counter adds one.

8.3.3 ST—Structured Text

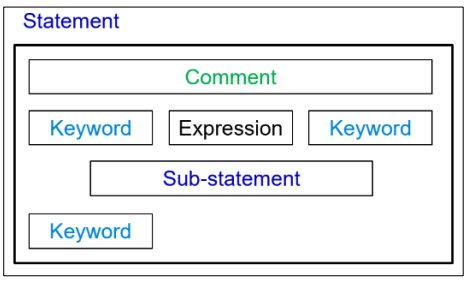
The **Structured Text (ST)** is a high-level language similar to C or PASCAL. It is composed of texts. Compared with Instruction Lists (IL), Structured Text is more readable. It is very flexible and intuitive for writing control algorithms. People who are trained in programming languages often find it easiest language to use for programming control logic. Besides, Structured Texts provide familiar programming environment for programmers who are familiar with C language.

Basic Structure of Structured Text

Structured Text has the same function as the ladder diagram as shown in the following figure. Structure of the structured text is similar to the syntax of a high-level language. A structured text is composed of statements.

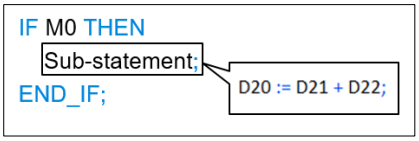


The Structure of the structured text is shown in the following figure.

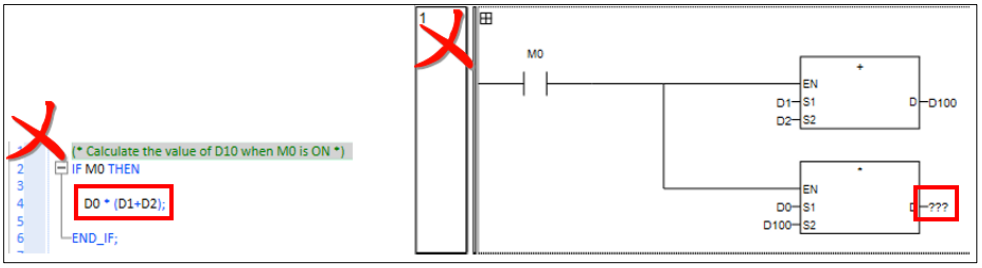


Statement

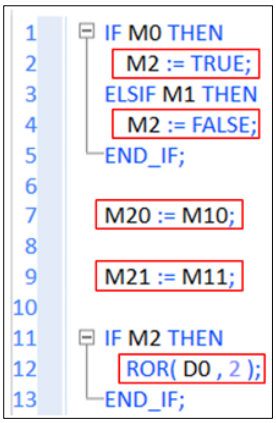
The statement is a basic unit. It represents a complete task which can be executed. A complete statement may not be represented by a single line, but it always ends with a semicolon.



A complete statement corresponds to a network in ladder diagram. It represents a task. For example, let's take the scenario in the following figure. The value of the arithmetic expression D0 * (D1+D2) is calculated and the result of the operation is stored in D10. The arithmetic expression in the red frame in the following figure is not a valid statement. It is an expression. It is not a concrete task. It is like the ladder diagram shown in the following figure. The ladder diagram lacks an operand. It is not a complete program which can be executed.



A structured text is composed of statements and a statement is composed of elements. An example of a structured text is shown in the following figure. The complete statements in the structured text are in the red frames.



Note: In a Structured Text, ":=" means 'to assign' and "=" means "if equal to'

Expression

An expression is an important element of a statement. It represents a value, for example, TRUE or FALSE of Boolean value, or 16 or 3 of integral value. An expression can be an arithmetic expression, a constant, a variable, or a device.

Examples

- M0 & M1 (Boolean value)

  The logical operator & takes M0 and M1 and performs the logical & operation on M01 and M1. The result of the operation is a Boolean value.

- M0 = FALSE (Boolean value)

  The expression is a conditional expression. If M0 is ON, the Boolean value is FALSE. If M0 if OFF, then the Boolean value is TRUE.

- M0 (Boolean value)

  This expression takes the value of M0 as its Boolean value. If M0 is ON, the Boolean value is TRUE, if M0 is OFF, the Boolean value is FALSE.

- D1 + D2 (value)

  The value of the arithmetic expression D1+D2 is calculated.

- D0 (value)

  The value represented by the expression is the value stored in D0.

- D2 = D0 + D1

  If the value of the arithmetic expression D0+D1 is equal to the value stored in D2, the Boolean value is TRUE. If the value of the arithmetic expression D0+D1 is not equal to the value stored in D2, the Boolean value is FALSE.

- D2 : = D0 + D1; (statement)

  This is a statement rather than an expression. The value of the arithmetic expression D0+D1 is calculated and the result of the operation is stored in D2. This statement is composed of the expression D2 and the expression D0+D1.

  In the following example, the program is mainly composed of two IF statements. When the first IF statement is executed, the system evaluates the expression M0. If M0 is ON, the value of the arithmetic expression D21+D22 is calculated and the

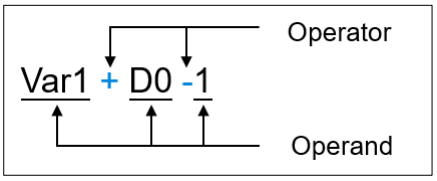
result of the operation is stored in D20. When the second IF statement is executed, the system judges whether the value of the arithmetic expression D0*(D1+D2) is equal to the value stored in D20. If the value of the arithmetic expression D0*(D1+D2) is equal to the value stored in D20, the Boolean value is TRUE and the value of M1 is TRUE. If the value of the arithmetic expression D0*(D1+D2) is not equal to the value stored in D20, the value of M1 is FALSE.

```
1   (* This logic calculates the sum of D21 and D22 and stores in D20 when M0 is TRUE *)
2   IF M0 THEN
3       D20 := D21 + D22;
4   END_IF;
5
6   //This logic sets the value of M1 based on value of D20
7   IF D20 = D0 * (D1 + D2) THEN
8       M1 := TRUE;
9   ELSE
10      M1 := FALSE;
11  END_IF;
```

Operand and Operator

Operand and **Operators** are basic elements of an expression. An Operand is the Object of an expression and an operator is a variable or function representing an operation. For example, in the expression D0+D1, D0 and D1 are operands and + is an operator.

An expression can be composed of operands and operators. It can also be composed of a single operand. The operands in an expression can be devices, variables or constants.



Rules of precedence affect which values form operands for which operators. If the operator in an expression share the same precedence, the order of the operation is carried out from left to right. The operands used in a Structured Text in DIADesigner are described in the following table.

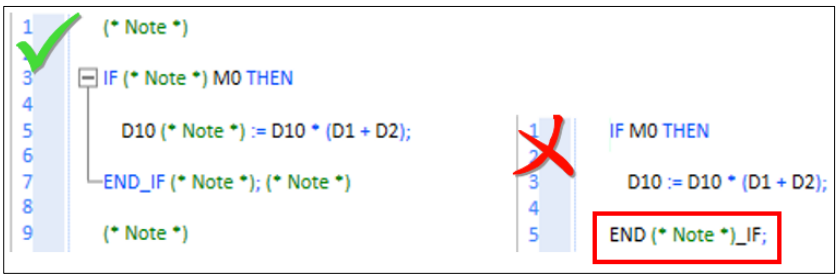
Symbol	Function	Data Format		Example of an expression		Precedence
		Operand	Result of an operation (Value of an expression)	Expression	Value	
( )	The parenthetical part of an expression is calculated first	Unspecified	Unspecified	(D0 + 6) * 3	33	Highest
**	Exponentiation	Floating-point number	Floating-point number	2.0 ** 5.0	3.2E+1	
-	Negative sign	Signed number	Signed number	-(D0 + 3)	-8	
NOT	Logical Negation	Boolean value	Boolean value	**NOT** M0	TRUE	
*	Multiplication	Any value	Any value	D0 * 3	15	
/	Division	Any value	Any value	15 / D0	3	
MOD	Remainder	Integer	Integer	D0 MOD 3	2	
+, -	Addition, Subtraction	Any value	Any value	D0 + 3	8	
<, >, <=, >=	Comparison	Any value	Boolean value	D0 > 2	TRUE	
=, <>	Equal to, Not Equal to	Any value	Boolean value	D0 <> 2	TRUE	
		Boolean value		M0 = TRUE	FALSE	
AND, &	Logical AND operation	Boolean value	Boolean value	M0 & M1	FALSE	
XOR	Logical Exclusive OR operation	Boolean value	Boolean value	M0 XOR M1	TRUE	
OR	Logical inclusive OR operation	Boolean value	Boolean value	M0 OR M1	TRUE	Lowest

Keyword and Comment

In computer programming, a keyword is a word which has a special meaning to the programming language. For example, TRUE and FALSE in a program represent Boolean values and the IF statements in the example above represent tasks which are executed. Besides, applied instructions are regarded as keywords. In order to prevent a program from being compiled incorrectly, the words which are regarded as keywords in the program cannot be declared to be variables. However, the words which are regarded as keywords in a program can be parts of the variables in the programs. For example, FIFO and _IF are allowed.

In a structure text, the words between (* and *) are regarded as Comments. When a program is compiled, the system automatically skips the words between (* and *) in the program. Besides, if the structure of a structured text is not destroyed, comments can be put anywhere in the structured text. However, you must be careful about the readability of the structured text.

Example: The comments in the left example are valid. The comment in the right-side example divides the keyword END_IF into two parts and therefore an error will occur if the program is compiled.



Using a Variable whose Data Type is ARRAY in a Structured Text

If you want to use a variable whose datatype is ARRAY in a structured text, the expression format is identifier [index]. The index is a constant or a variable. However only the indexes in a project for an AS series CPU module can be variables. The minimum index value must be 0, whether the index is a constant or a variable. The maximum index value cannot be larger than or equal to the number of elements in an array. For example, if the number of elements in an Array is 10, the index values must be between 0 and 9.

```
1  IF M0 THEN
2
3      Ary_A[0] := Ary_B[IDX];
4
5  END_IF;
```

Besides, pay attention to the following points when you use a variable whose data type is array.

- You can only manipulate a single element in an array. A variable which represents an array cannot be used as an operand.
- The arrays that DIADesigner supports are one-dimensional arrays, but the structure from the Data Unit Type can support up to three-dimensional arrays. This type of variables can be written as shown in the following figure.

```
1  IF M0 THEN
2
3      vardut.pt1[1,1,1] := D0;
4
5  END_IF;
```

- If an index is a variable, and the value of the variable is not within a valid range, the PLC will continue running and no error will occur. However, the data taking part in the operation may be incorrect.
- If the index for a variable is constant, the variable can be modified by an index register. If the index register for a variable is variable, the variable cannot be modified by an index register.

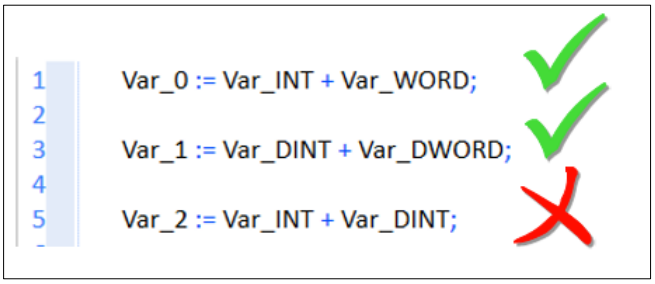
Important Points about Structured Texts

- If the structure of the keywords or the statements in a structured text is not destroyed, starting new lines and leaving blank spaces is allowed.
- In principle, full width characters and half width characters in DIADesigner are regarded as different characters. Avoid using full width characters lest they should be identified incorrectly.
- Structured Text language is case-insensitive. IF and if are considered to be equivalents of if (The C language is case-sensitive.)
- To use constants in a ST POU in DIADesigner, the constants must be represented

in the following ways.

- Decimal value: 23456 (A value which is not preceded by any mark is regarded as a decimal value.)
- Hexadecimal value: 16#5BA0 (A value which is preceded by 16#)
- Octal value: 8#55640 (A value which is preceded by 8#)
- Binary value: 2#1011 (A value which is preceded by 2#)
- String: “XYZ” (Characters are put in double quotes)
- Boolean value (BOOL): TRUE or FALSE, and 1 or 0 are also allowed.

- In a ST POU in DIADesigner, the data types of the operands in an expression must be the same. However, if the data type of one of the operands is INT and the data type of the other operand is WORD, both operands can take part in the operation. Likewise, if the data type of one of the operands is DINT and the datatype of the other operand is DWORD, the two operands can take part in the operation as well. If the data lengths of the operands in an expression are different, then the operands cannot take part in the operation.



- If the data type of a Variable is WORD, DWORD, the system regards it as a signed integer during the operation.
- Although there is no limit to the number of lines, you still must consider the capacity of the memory in the PLC.
- The sections of a ST in DIADesigner can be copied / cut / pasted. You can copy the edited text into a ST in DIADesigner.
- The following instructions are not supported in a ST.

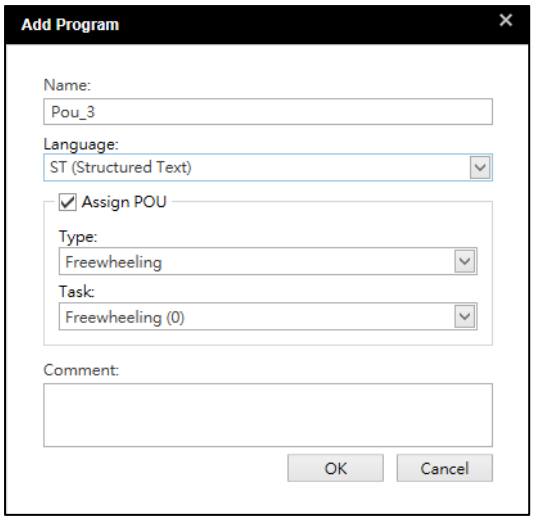
AS series CPU modules
All the Instruction List (IL) instructions are not supported. For example, LD, LDI, OUT, SET, RST, LD and LDP are not supported.
All the Pulse type API instructions are not supported. For example, ROLP, RORP, BCDP, TRDP are not supported.
The four Arithmetic operations instructions, the Logical operation instructions, the Contact-type Logic operation instructions, and the Structure creation instructions are not supported. In addition, instructions including PLS, PLF, MOV, DMOV, DFMOV, $MOV, MC, MCR, CJ, JMP, GOEND are not supported. Also, instructions such as CNT, DCNT used in counter values and TMR instruction used by counter in subroutine program are not supported.

- The following instructions are supported in a ST but with limitation.

AS series CPU modules
All the high-speed output instructions (API 27xx) for example, DPLSY and DDRVI are supported. Refer to the DDRVI instruction for application.
Use of DCNT instruction for high speed counters is supported. But, statements need to be executed so that the values can be updated.
Communication instructions are supported but only after flag instruction setup is completed. The instructions used in the program procedures can be scanned for execution.

8.3.3.1 Add ST Program

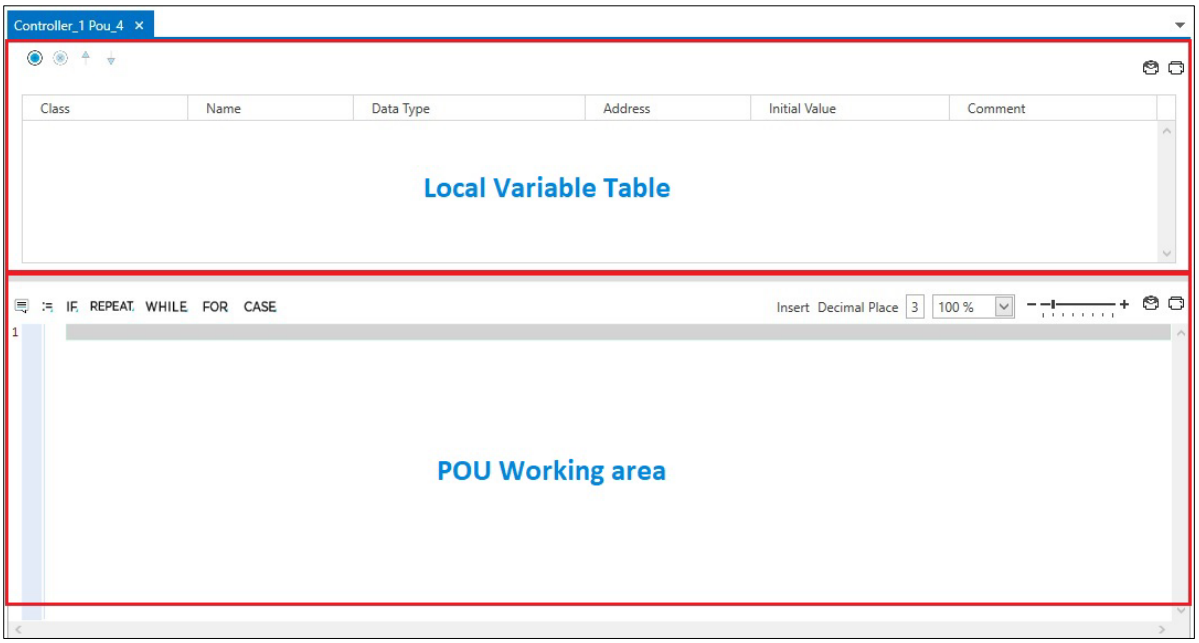
Select **ST (Structured Text)** in the **Language** field in **Add POU** dialog while creating a POU of program type or function block type. Refer to POU Architecture and Types for detailed information.



The environment in which a ST can be edited is shown in the following figure. In any POU dialog (program type or function block type), there are two areas.

- Local Variable table area
- POU Working area

The table in the upper part of the ST POU dialog is a Local Variable Table, and the lower part is POU working area. After a program editing dialog to create a ST POU is opened, the corresponding toolbar appears in the DIADesigner dialog. There are line numbers on the left side of the working area.



ST Toolbar

After the POU dialog is opened, the toolbar for editing the Structured Text appears.

The following table describes the functions in ST POU dialog.



Function / Icon	Description
	Add / delete line comments
	Assignment Structure :=
IF	Conditional Structure IF

Function / Icon	Description
REPEAT	Loop Structure REPEAT
WHILE	Loop Structure WHILE
FOR	Loop Structure FOR
CASE	Conditional Structure CASE
	Increase indent
	Decrease indent
	Auto indent
	Toggle bookmark
	Previous bookmark in tab
	Next bookmark in tab
	Clear all bookmarks in tab

8.3.3.2 ST Expressions and Statements

Assignment Structure – :=

Format

```
Address/Variable := Expression;
```

Description

The value represented by the expression at the right side of **:=** is assigned to the address or the variable at the left side of **:=**.

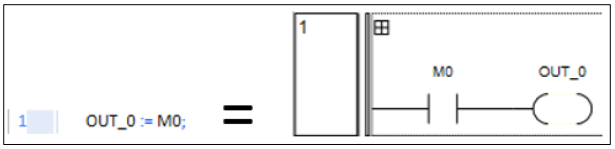
Rules and Limits

- The expression at the right side of **:=** can be a constant, an address, a variable or an arithmetic expression. Only an address or a variable can be at the left side of the **:=**.
- The operands at both sides of **:=** must conform to the following rules.

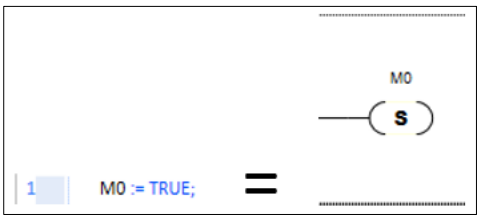
- One operand is an M / S / T / C / HC address, and the other operand is a variable whose data type is BOOL / STEP / TIMER / COUNTER (the address assigned to the operand which is a variable is a bit address)
- One operand is a D / L address, and the other operand is a variable whose data type is not BOOL / STEP / REAL / STRING.
- One operand is a T / C address, and the other operand is a D/L address or a variable whose data type is WORD / INT / TIMER / COUNTER.
- One operand is a D / L / HC address or a variable whose datatype is COUNTER and to which a HC address is assigned. The other operand is a variable whose datatype is DWORD / DINT
- One operand is a variable whose datatype is WORD / DWORD / LWORD and the other operand is a variable whose data type is INT / DINT / LINT. The data length of the operand at the left side of **:=** must be larger than or equal to the data length of the operand at the right side of **:=**.
- Both operands are variables whose datatypes are REAL / LREAL / STRING. The data lengths of the operands must be the same.

Examples

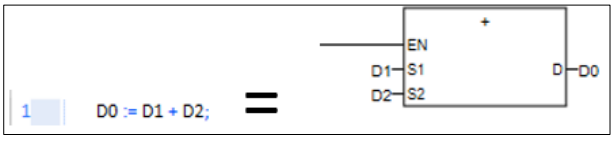
Example 1: The value of M0 is assigned to the variable OUT_0.



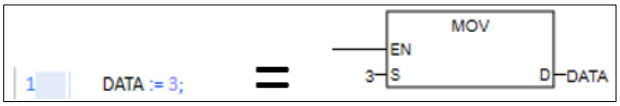
Example 2: M0 is set to ON.



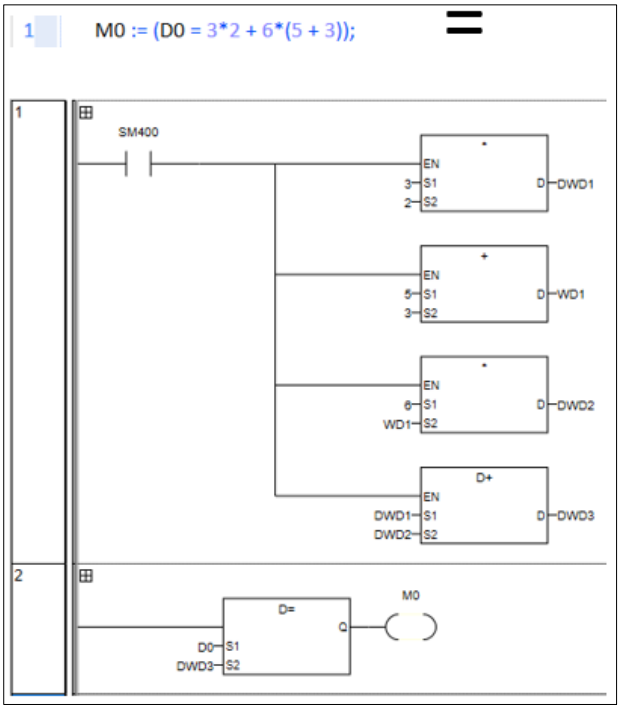
Example 3: The value in D2 is added to the value in D1. The sum is stored in D0.



Example 4: 3 is assigned to the variable DATA.



Example 5: The system judges whether the value of the arithmetic expression 3*2+6*(5+3) is equal to the value stored in D0. The judgement result is sent to M0.



Example 6: The following table lists the examples of incorrect use of the assignment structure.

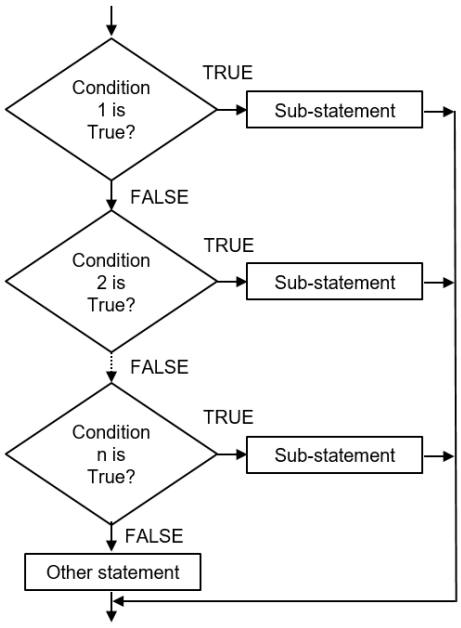
Statement	Description
D0:= M0;	The datatype of one operand must not be different from the datatype of the other operand.
V_WORD:= V_DWORD;	The data length of the operand at the left side of **:=** must not be less than the data length of the operand at the right side of **:=**.
D0:= V_REAL;	The value of a variable whose datatype is REAL cannot be assigned to a D address.
V_LREAL:= V_REAL;	If one operand is a variable whose datatype is REAL, the other operand must be a variable whose datatype is REAL. If one operand is a variable whose datatype is LREAL, the other operand must be a variable whose datatype is LREAL.

Statement	Description
V_DWORD:= V_TIMER;	The value of a variable whose datatype is Timer cannot be assigned to a variable whose datatype is DWORD, but can be assigned to a variable whose datatype is WORD.
V_WORD:= HC0;	The value in a HC address cannot be assigned to a variable whose datatype is WORD, but can be assigned to a variable whose datatype is DWORD

Conditional Structure – IF

Format

```
IF Expression 1 THEN
   Sub-statement 1;
ELSIF Expression 2 THEN
   Sub-statement 2;
..
ELSE
   Sub-statement n;
END_IF;
```



Description

If the value of **Expression 1** is TRUE, **Sub-statement 1** is executed. If the value of **Expression 1** is FALSE, **Expression 2** is evaluated. If the values of all the Expressions are FALSE, **Other sub-statement** are executed.

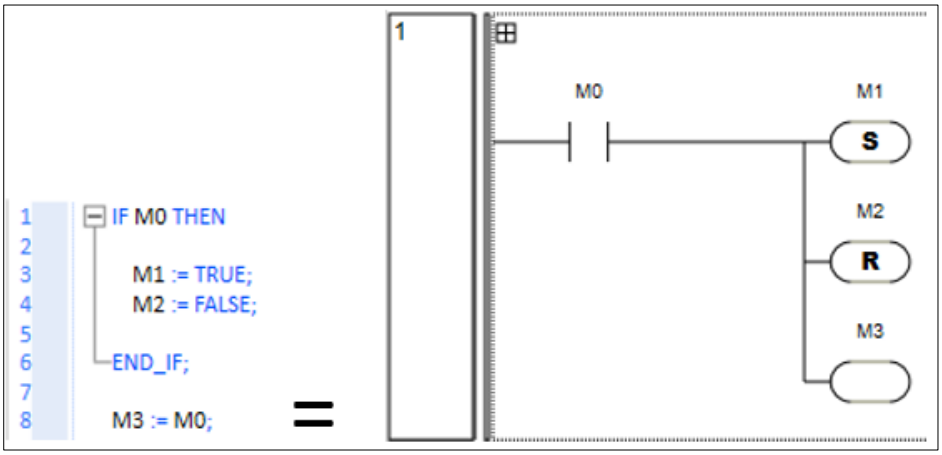
Rules and Limits

- A Boolean expression can be an address, a variable or an arithmetic expression. It cannot be a constant.
- A sub-statement can be a valid statement, or an **IF** statement.
- There is not limit on the number of sub-statements following **THEN** or **ELSE**.

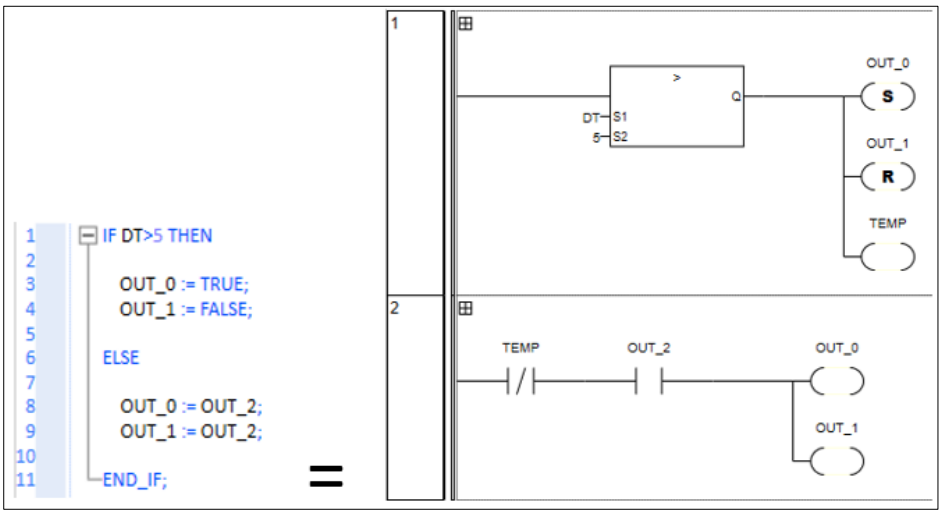
- The **ELSIF** sections and the **ELSE** section can be omitted. If the **ELSE** section is omitted and the values of all the Boolean expressions are FALSE, no sub-statement is executed.
- Add **ELSIF** sections to a conditional structure at will. There is only one **ELSE** section in a condition structure. The **ELSE** section in a condition structure is at the last section of the conditional structure.

Examples

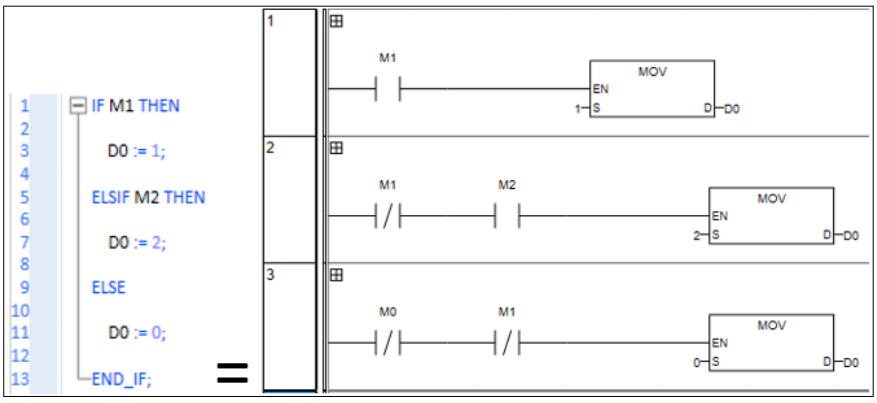
Example 1: If M0 is ON, M1 is set to ON and M2 is set to OFF. If M0 is OFF, then the step will end. No matter the value of M0 is, the value of M0 is assigned to M0.



Example 2: If the value of DT is larger than 5, OUT_0 is set to ON and OUT_1 is reset to OFF. If the value of DT is not larger than 5, the value of OUT_2 is assigned to Out_0 and Out_1.

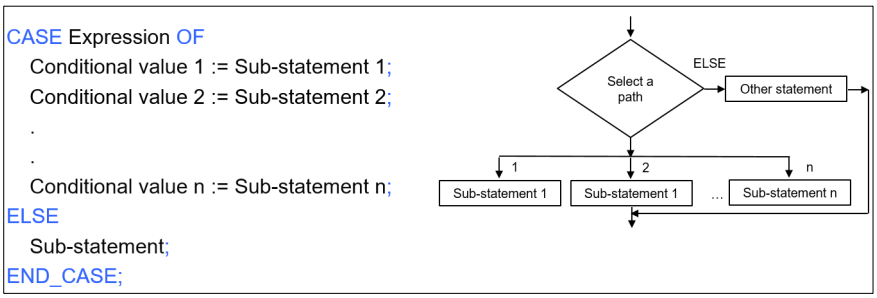


Example 3: If M1 is ON, the value in D0 is 1. If M1 is OFF, the PLC will evaluate M2. If M2 is ON, the value in D0 is 2, If M2 is OFF, the value in D0 is 0.



Conditional Structure – CASE

Format



Description

The PLC judges whether the value of **Expression** is on the list of condition values. If the value of **Expression** is the same as a condition value on the list, the sub-statement following the condition value is executed. If the value of **Expression** is not on the list of condition values, **Other sub-statement** under the keyword **ELSE** are executed.

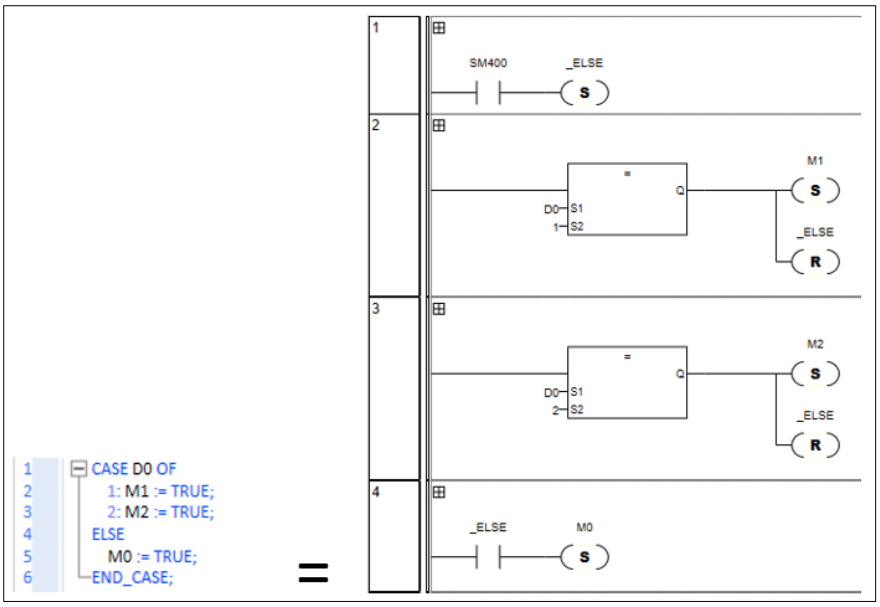
Rules and Limits

- **Expression** in the above figure can be an address, a variable or an arithmetic expression. It cannot be a constant. The data type of Expression must be INT (-32768 ~ 32767) or DINT (-2147483648 ~ 2147483647). If an address is used as Expression, the system will automatically regard the value in the address as an integer. Besides, if an arithmetic expression is used as Expression and the result of the operation is not an integer, the result is rounded down to the nearest whole digit.

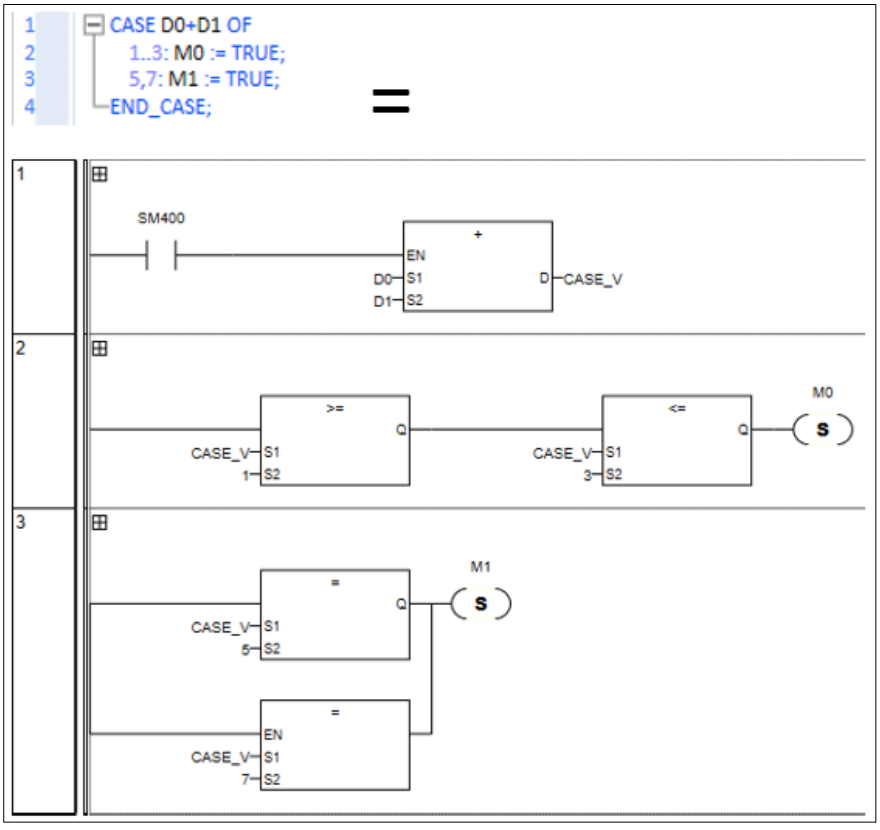
- A condition value on the list cannot be the same as another condition value on the list and the condition values on the list must be integers. The range of condition values depends on the data type of Expression. If the datatype is INT, the condition values must be in the range of -32768 ~ 32767. If the datatype is DINT, the condition values must be in the range of -2147483648 ~ 2147483647.
- If the sub-statements following some conditions values are same, you can combine them.
    - If the condition values are not continuous values, they are separated by commas. For example, **1, 3, 5: Sub-statement** means that the sub-statement is executed if the value of **Expression** is 1, 3 or 5.
    - If the condition values are continuous values, the range of condition values can be indicated by the mark "…". For example, **3..6: Sub-statement** means that the sub-statement is executed if the value of **Expression** is in the range of 3 to 6 (including 3 and 6). If the mark ".." is used, the value at the left side of ".." must be less than the value at the right side of "..". For example, 6..3 is not valid.
- A sub-statement can be valid statement or another structure.
- There is not limit on the number of sub-statements following a condition value.
- The **ELSE** section can be omitted. If the ELSE section is omitted, and the value of **Expression** is not on the list of condition values, no sub-statement is executed.
- After a sub-statement is executed, the execution of the program will jump out of the **CASE** structure. No Jump instruction is needed.

Examples

Example 1: If the value of D0 is 1, M1 is ON. If the value of D0 is 2, M2 is ON. If the value of D0 is neither 1 nor 2, M0 is ON.



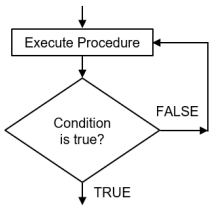
Example 2: The value of arithmetic expression D0+D1 is calculated first. If the result of the operation is in the range of 1 to 3, M0 is ON. If the result of the operation is 5 or 7, M1 is ON. If the result of the operation is not on the list of condition values, no sub-statement is executed and the execution of the program will jump out of this structure.



Loop Structure – REPEAT

Format

```
REPEAT
   Sub-statement;
UNTIL Expression;
END_REPEAT;
```



Description

The sub-statement is executed and then the **Expression** (BOOL) is evaluated. If the Expression is evaluated as TRUE, the loop is terminated. If the value of the Expression is FALSE, the program in the structure is executed repeatedly until the value of the Expression is TRUE.

Rules and Limits

- The **Boolean Expression** can be an address, a variable or an arithmetic expression. It cannot be a constant.
- The sub-statement is executed before the **Expression** is evaluated and therefore the sub-statement is executed at least once.
- The sub-statement can be a valid statement or another structure. There is not limit on the number of sub-statements in a **REPEAT** structure. One loop structure can be embedded in another loop structure. There are at most 64 loop structures.
- If the value of the **Expression** following **UNTIL** is TRUE, the loop is terminated. To prevent an infinite loop, the operand in the **Expression** following **UNTIL** cannot be an address or a variable which cannot be updated.
- The X address in a program are updated after the program is scanned. To use X address as the operand of the **Expression**, add an **I/O update instruction** to the structure, or update the X through an interrupt subroutine.

Examples

Example 1: The initial value of DT is 0. In the loop structure, the values of DT increase progressively by 2. The loop will not be terminated until the values of DT is larger than 100.

```
1     DT := 0;
2   REPEAT
3       DT := DT + 2;
4     UNTIL DT > 100
5   END_REPEAT;
```

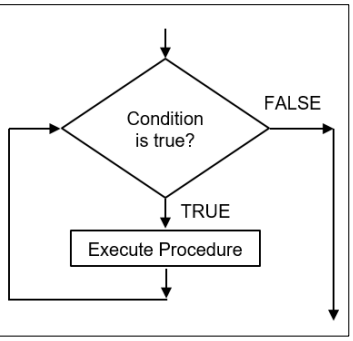
Example 2: The initial value in D0 is 1 and the initial value in D10 is 5. After the loop is terminated, the value in D0 is %*4*3*2*1.

```
1     D10 := 5;
2
3     D0 := 1;
4
5   REPEAT
6       D0 := D0 * D10;
7       D10 := D10 - 1;
8     UNTIL D10 = 0
9   END_REPEAT;
```

Loop Structure – WHILE

Format

```
WHILE Expression DO
   Sub-statement;
END_WHILE;
```



Description

The **Expression** (BOOL) is evaluated. If the value of the Expression is TRUE, the sub-statement is executed. This repeats until the value of the Expression becomes FALSE.

Rules and Limits

- The **Expression** can be an address, a variable or an arithmetic expression. It cannot be a constant.
- The sub-statement is executed if the value of the Expression is TRUE and therefore the sub-statement may not be executed.

- The sub-statement can be a valid statement or another structure. There is no limit on the number of sub-statements in a **WHILE** structure. One loop structure can be embedded in another loop structure. There are at most 64 loop structures.
- If the value of the **Expression** following **WHILE** is FALSE, the loop is terminated. To prevent an infinite loop, the operand in the **Expression** following **WHILE** cannot be an address or a variable which cannot be updated.
- The X address in a program is updated after the program is scanned. If you want to use an X address as the operand of the Boolean expression, you must add an **I/O update instruction** to the structure or update the X through an interrupt subroutine.

Examples

Example 1: The initial value of DT is 0. In the loop structure, the value of DT increases progressively by 2. The loop will not be terminated until the value of DT is larger than 100.

```
1     DT := 0;
2
3   WHILE DT <= 100 DO
4       DT := DT + 2;
5   END_WHILE;
```

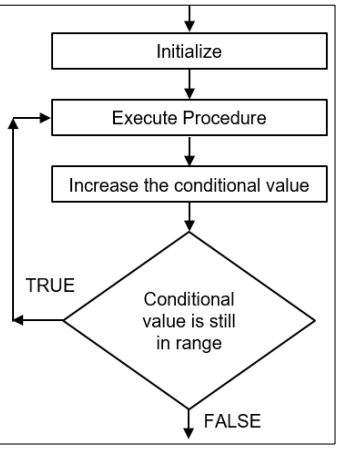
Example 2: The REPEAT loop checks the Boolean expression after the sub-statement is executed. The sub-statement in the REPEAT structure is executed once. After the REPEAT loop is terminated, the value in D0 becomes 1. The WHILE loop checks the Boolean expression before the sub-statement is executed. The Boolean expression in the WHILE structure is evaluated as FALSE and therefore the sub-statement in the WHILE structure is not executed. After the WHILE loop is terminated, the value in D1 will still be 0.

```
1     M0 := TRUE;
2     D0 := 0;
3     D1 := 0;
4
5   REPEAT
6       D0 := D0 + 1;
7   UNTIL M0
8   END_REPEAT;
9
10  WHILE (NOT M0) DO
11      D1 := D1 + 1;
12  END_WHILE;
```

Loop Structure – FOR

Format

```
FOR Address or Variable := Start Value TO End Value BY Increment DO
   Sub-statement;
END_FOR;
```



Description

The initial value of the **Variable** or the **Address** is the **Start Value**. Whenever the loop body is executed, the value of the **Variable** or the **Address** increases by the **Increment.** If the value of the **Variable** or the **Address** is in the range of the **Start Value** to the **End Value**, the loop body is executed. If the value of the **Variable** or the **Address** is not in the range of the **Start Value** to the **End Value**, the loop is terminated.

Rules and Limits

- The **Start Value / End Value / Increment** value can be an address, a variable, an arithmetic expression or a constant. The datatype of the **Start Value / End Value / Increment** must be INT (-32768 ~ 32767) or DINT (-2147483648 ~ 2147493647). If an address is used as the **Start Value / End Value / Increment**, the system will automatically regard the value in the address as an integer. Besides, if an arithmetic expression is used as the **Start Value / End Value / Increment** and the result of the operation is not an integer, the result is rounded down to the nearest whole digit.
- If the **End Value** is less than the **Start Value**, the increment must be given a negative number. Otherwise, there is an infinite loop.
- The datatype of the variable must be INT or DINT. If the variable or the address takes part in the operation, prevent the value of the variable or the address from resulting in an infinite loop.

- The sub-statement can be valid statement or another structure. There is not limit on the number of sub-statements in a **FOR** structure. One loop structure can be embedded in another loop structure. There are at most 64 loop structures.
- After the value of the variable or the address increases by the **Increment**, the system will check whether the value of the variable or the address is in the range of the **Start Value** to the **End Value**. If an overflow occurs after the value of the variable or the address increases by the **Increment**, there is an infinite loop.

Example

The initial value of the variable INDEX is 1. Whenever the loop body is executed, the value of the INDEX increases by 2. After the loop body is executed for the fifth time, the value of the INDEX will increase by 2 and the value of the INDEX becomes 11. Since the value INDEX is not in the range of 1 to 9, the loop is terminated. The initial value of the variable SU is 0. After the loop is terminated, the value of SU will be the odd sum of squares of 1 to 9, that is, $1^2+3^2+5^2+7^2+9^2$.

```
1    SU := 0;
2
3    FOR INDEX := 1 TO 9 BY 2 DO
4        SU := INDEX * INDEX + SU;
5    END_FOR;
```

Blank Statement

Format

;

Description

If a statement is needed, but no action needs to be executed, a blank statement is used.

Example

FOR loop in the following figure shows that no action is executed. Owing to the blank statement, the execution of the program is delayed. Adjust the time for which the execution of the program is delayed by adjusting the number of times the loop body is executed.

```
1   FOR D100:= 1 TO 3000 BY 1 DO
2       ;
3   END_FOR;
```

RETURN Statement

Format

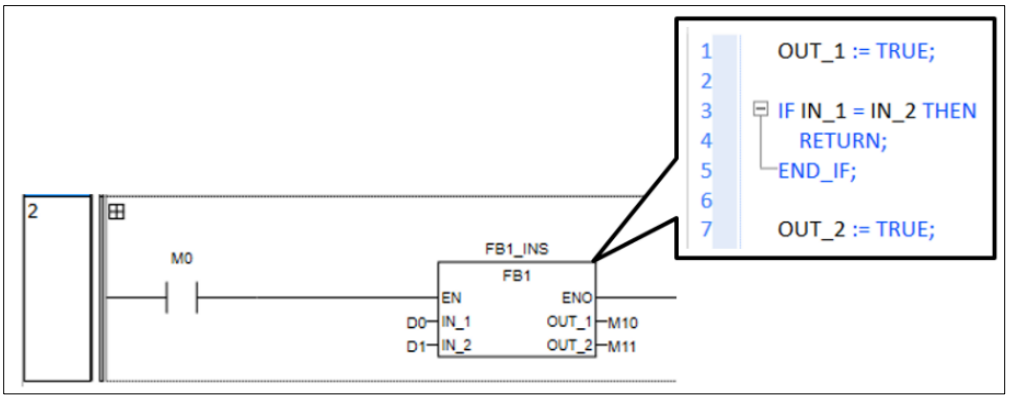
```
RETURN;
```

Description

A RETURN statement is used in a function block in a structured text. It ends the program in the function block.

Example

The programming language used to create the program in Function Block FB1 is a Structured Text. After the upper part of the program is executed, IN_1 is compared with IN_2. If IN_1 is equal to IN_2, then the execution will leave the current program. If IN_1 is not equal to IN_2, then, the program in the Function Block is executed.



EXIT Statement

Format

```
EXIT;
```

Description

An **EXIT** statement is used in a **REPEAT** loop, a **WHILE** loop or a **FOR** loop. It is used to exit the loop.

Examples

Example 1: The loop in the following figure is a **REPEAT** loop. The loop will not be terminated until the value of DT is 0. The IF structure in the loop is used with an **EXIT** statement. The initial value in D0 is 0. It increases by 1. The loop will not be terminated until the value in D0 is 100. As a result, the loop body is executed 100 times at most. In the following figure, the initial value of DT is 110. After the loop body is executed for the 100th time, the value of DT becomes 10. In other words, the final value of DT is 10. The value of DT will not decrease to 0.

```
58      DT := 110;
59      D0 := 0;
60
61      REPEAT
62          DT := DT - 1;
63          D0 := D0 + 1;
64
65          IF D0 = 100 THEN
66              EXIT;
67          END_IF;
68
69      UNTIL DT = 0
70      END_REPEAT;
```

Example 2: The loop in the following figure is a **FOR** loop. The **Start Value** is 1, the **Increment** is 1, the **End Value** is determined by the value of the DT variable. The value of the arithmetic expression D0 * D10 is calculated and the result of the operation is stored in D0. The IF structure in the loop is used with an EXIT statement. The loop will not be terminated until the value in D0 is larger than 100. When the value in D10 increases to 5, the value in D0 is 1*2*3*4*5 and the loop is terminated.

```
72      DT := 10;
73      D0 := 1;
74
75      FOR D10 := 1 TO DT BY 1 DO
76
77        D0 := D0 * D10;
78
79        IF D0 > 100 THEN
80          EXIT;
81        END_IF;
82
83      END_FOR;
```

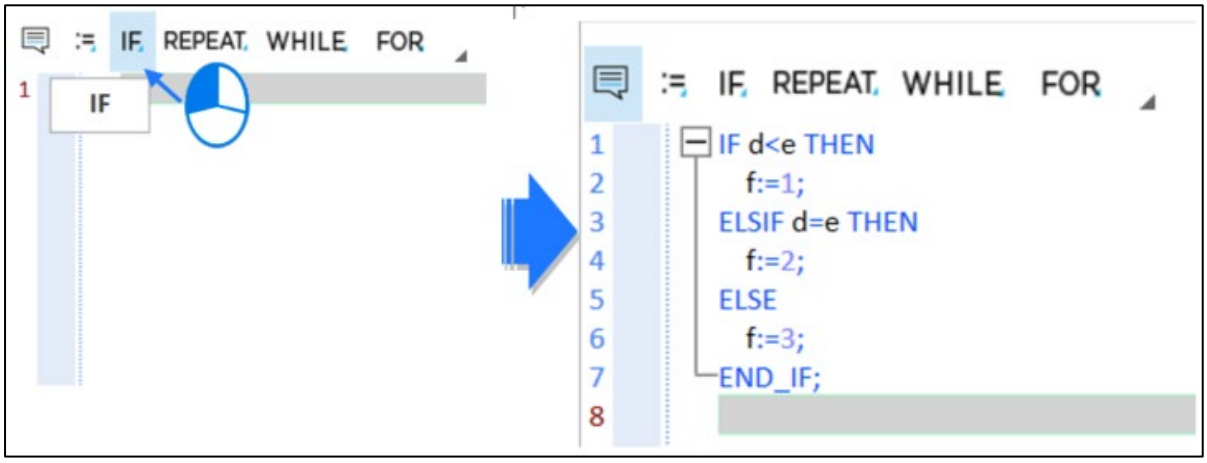
8.3.3.3 Edit ST Program

Editing ST Statement

The way in which a ST program is edited is similar to the way in which a text file is edited with a general text editor. To edit ST program, type or modify the text in the working area. To start a new line of text at a specific point, press **Enter** on the keyboard.

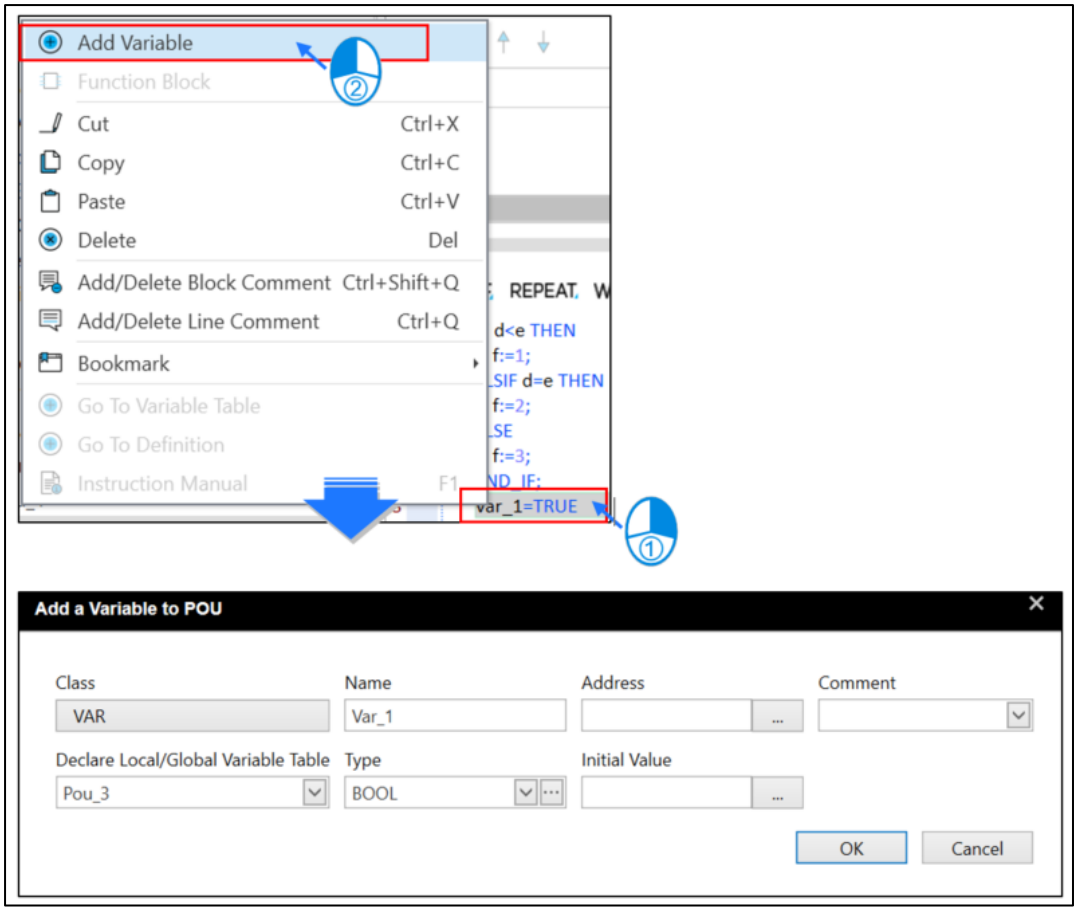


Besides, click a line which can be edited and insert a conditional or loop structure by clicking the respective icon on the toolbar.



If you want to select any amount of text, click where you want to begin the selection, hold down the left mouse button, and drag the pointer over the text that you want to select. You can also click at the start of the selection, scroll to the end of the selection, and hold down **Shift** key on the keyboard while you click where you want the selection to end.

If you type a Variable which has not been declared, right-click the variable to open its context menu. Click **Add Variable** in the context menu to open **Add a Variable to POU** dialog. Select the required properties and click **OK**.



Line Comment

After creating a program, if you do not want to compile and execute some portion of the ST, comment them with the help of **Line Comment**. Compiler will ignore the lines which are commented and do not execute them. Commented lines display in green color prefixed with //.

Follow these steps to edit / delete line comment.

To edit / delete line comment

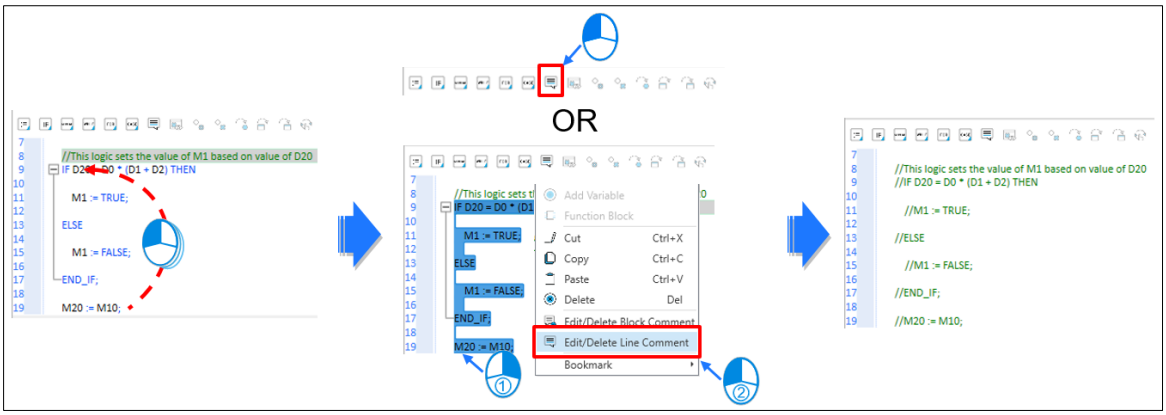
1. Select the lines to be commented
2. Click  on **ST POU** toolbar.

   Or

   Right-click lines to open the context menu and select **Edit/Delete Line Comment**.

   Or

   Press **Ctrl** + **Q** keys simultaneously on the keyboard.



3. Repeat the above steps once again to delete the line comment.

Block Comment

Block Comment is one simpler way to comment the lines in a ST program. Instead of prefixing all the lines, the compiler will insert **(*** in the beginning of the first commented line and a ***)** at the end of last commented line. Commented lines appear in **green** color.

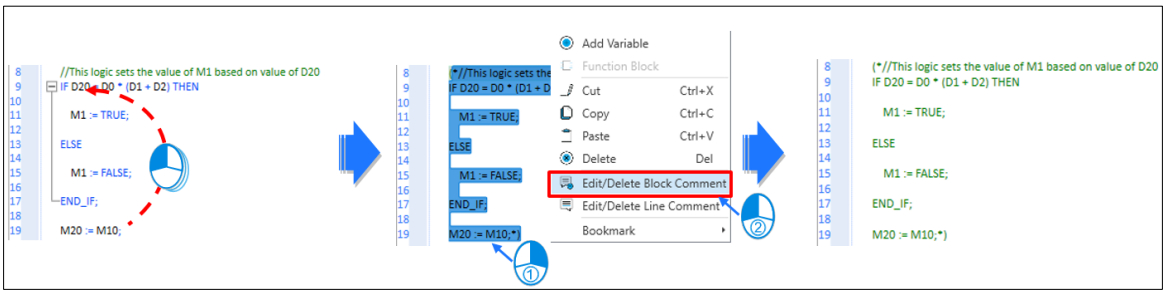
Follow these steps to edit / delete a block comment.

To edit / delete block comment

1. Select the lines to be commented.
2. Right-click the lines to open the context menu and select **Edit/Delete Block Comment**

   Or

   Press **Ctrl** + **Shift** + **Q** keys simultaneously on the keyboard.



3. Repeat the above steps once again to delete the block comment.

Applied Instruction Structure

Format

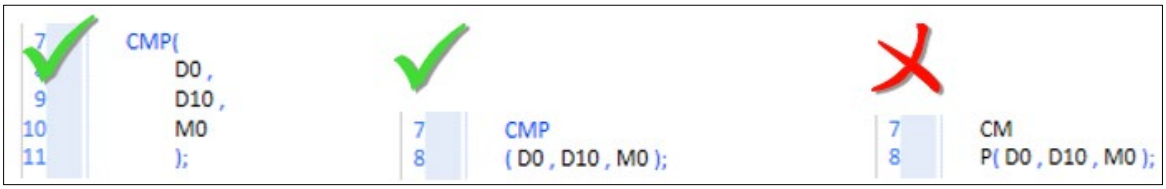
```
Name of API (Operand 1, Operand 2, …, Operand n);
```

Description

The applied instruction is executed.

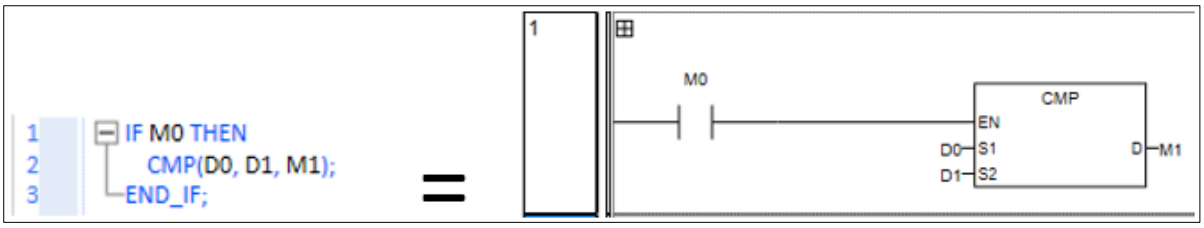
Rules and Limits

- The state of the **EN** pin need not be specified in the applied instruction. When the statement is executed, the applied instruction is executed. An **IF** statement functioning as an **EN** pin can be used with the applied instruction.
- As long as the structure of the operands and the structure of the keywords are not destroyed, starting new lines is allowed.
- The applied instruction is a statement, and therefore it ends with a semicolon.
- The operands specified in the applied instruction must be put in parenthesis.
- The order in which the operands are arranged cannot be changed. Refer to the Instruction Manual for more details.
- The applied instruction in DIADesigner do not support the assignment of values and therefore `M0 := CMP( D0 , D10 , M0 );` is an invalid statement.
- As long as the structure of the applied instruction is unchanged, starting new lines is allowed.

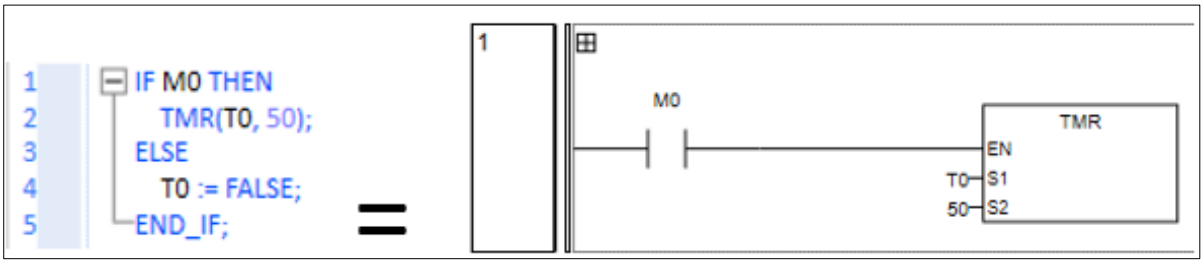


Examples

Example 1: The instruction **CMP** is used with an **IF** statement.



Example 2: The state of **EN** pin need not be specified in an applied instruction. If the instruction **TMR** is used, a statement about resetting the timer is required. Otherwise, the value of the counter will not become zero. For example, line 4 in the following figure is a statement about resetting the timer.



Function Block Structure

Format

```
Name of FB instance (Input pin 1 := Input operand 1,
                     Input pin 2 := Input operand 2,
                     …
                     Output pin 1 => Output operand 1,
                     Output pin 2 => Output operand 2,
                     …
                     Output pin n => Output operand n  );
```

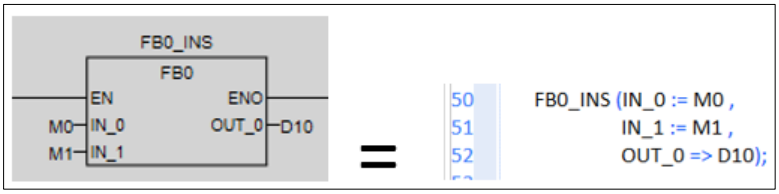
Description

Function Block (FB) is executed.

Rules and Limits

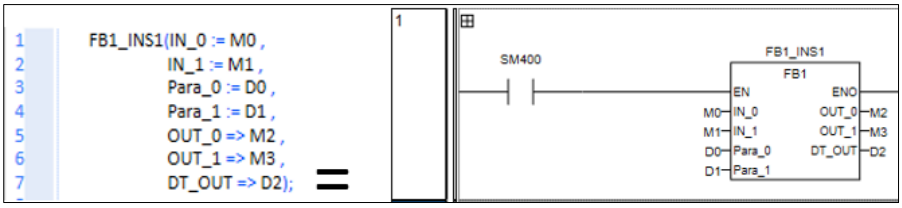
- The state of the EN pin of the **Function Block** instance need not be specified. When the statement is executed, the FB instance is executed. An **IF** statement functioning as an EN pin can be used with the applied instruction.
- As long as the structure of the operands and the structure of the keywords are not destroyed, starting new lines is allowed.
- The FB is a statement and therefore it ends with a semicolon.

- The input/output pins of the FB and the corresponding operands must be put in parenthesis. A pin and the corresponding operand form a group. The groups are separated by commas. The datatypes of the operands are specified in the FB definition.
- The FB before a parenthesis is a **FB instance** rather than a **FB definition**, that is, the variable before the parenthesis is a variable whose data type is a function block.
- The assignment mark for an input pin is **:=** and the assignment mark for an output pin is **=>**.
- The pins and the operands are arranged according to the definition of the FB. The arrangement cannot be changed. The EN pin and the ENO pin are executed. The input pins are listed above all the output pins. A pin of the VAR_IN_OUT class is an input pin. Refer to the ladder diagram and the structure text in the following example for more information.

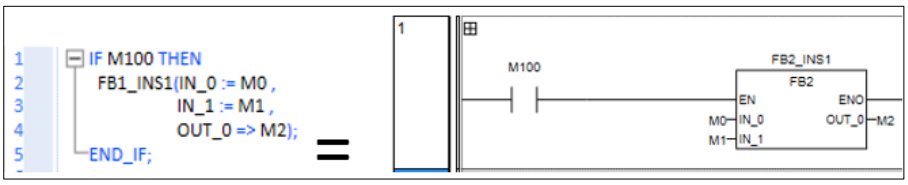


Examples

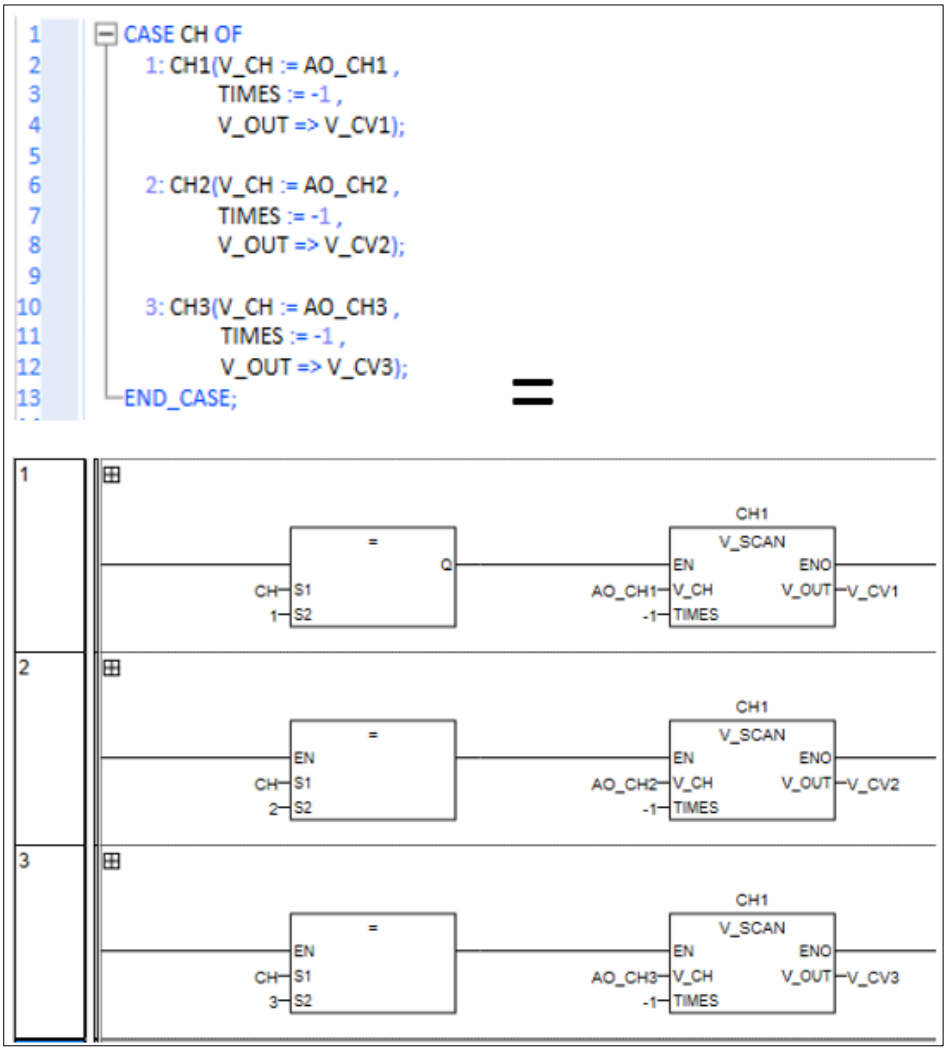
Example 1: When a statement which calls a function block is executed, the function block instance is executed.



Example 2: A function block is used with an IF statement.



Example 3: The value of the variable CH determines which function block is executed.



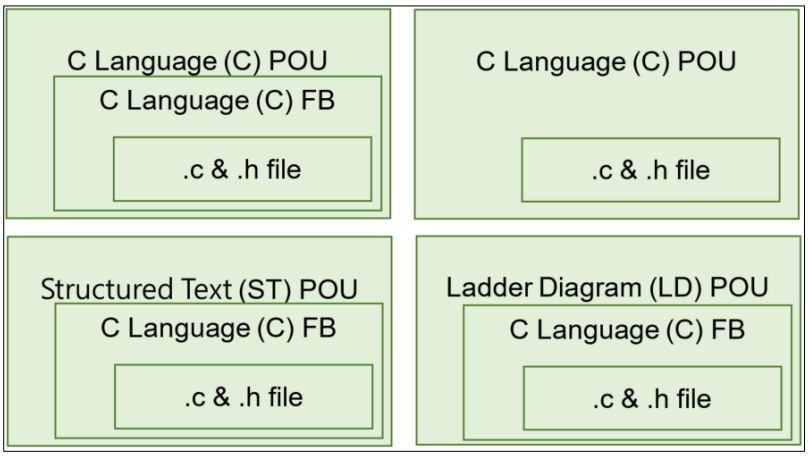
8.3.4 C Language Editor (C Language)

The characteristic of C language is that the entire program is constructed by many text descriptions. Compared with the instruction list (IL), the C language is more readable and can express a grammar similar to mathematical equations. Complex arithmetic, so for a program that needs to perform complex calculations, C language is a more suitable development language for such applications.

Note: Before downloading the C language program to PLC, confirm whether the AS host firmware is V1.06.00 or higher.

8.3.4.1 C Language Structure Description

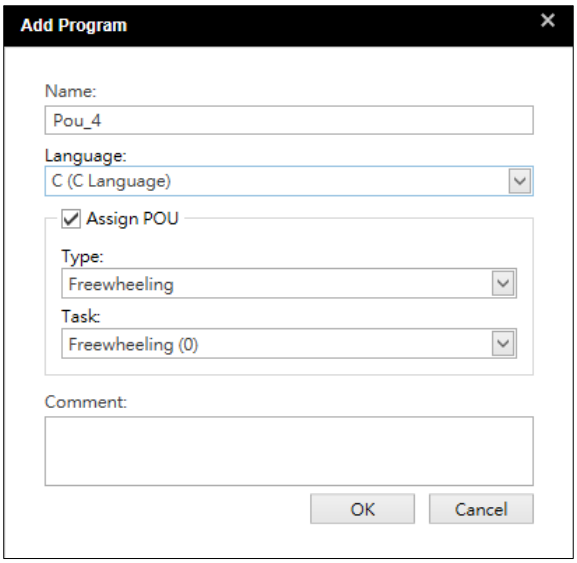
This editing language is suitable for programmers who are already familiar with C language editing. To start editing in C language, refer to the following figure for the editing language structure supported by C language; you can first use the common DEV-C and TurboC on the market And editing software such as Visual Studio for editing, compiling and testing, and then using the import function provided by DIADesigner, import the completed C language ".C" and ".h" files and use them in C language (C) FB or C language (C) POU, as shown in the following figure, can also use C language (C) FB in structured language (ST) POU or ladder diagram (LD) POU.



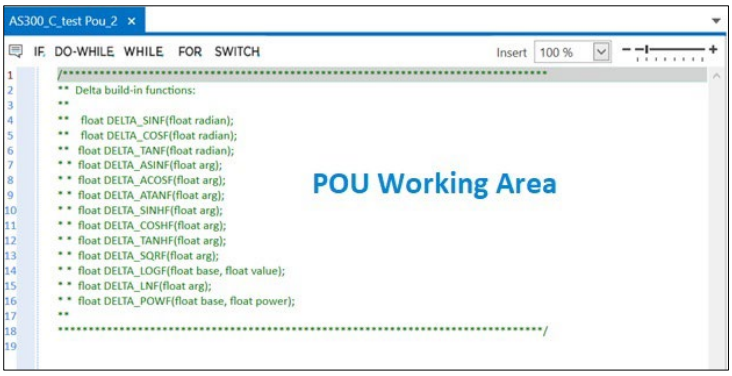
8.3.4.2 C Language Editing Environment Description

The C Code execution code storage space (ROM) provided by AS series hosts is 64K Bytes, and the variable declaration available space (RAM) is 8K Bytes. The above two storage spaces are independent of the existing PLC program space. After each compilation, it is recommended to pay attention to the remaining space from the compile in Error list message box.

When creating a program or function block POU, the **C (C Language)** programming language can be selected in the **Language** field of the **Add Program** dialog.

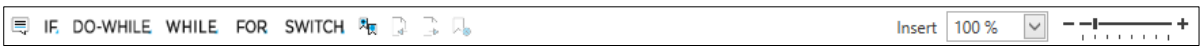


The C POU editing environment is shown in the following figure. The editor contains POU working area.



C POU toolbar

The following figure shows the toolbar of **C POU** dialog.

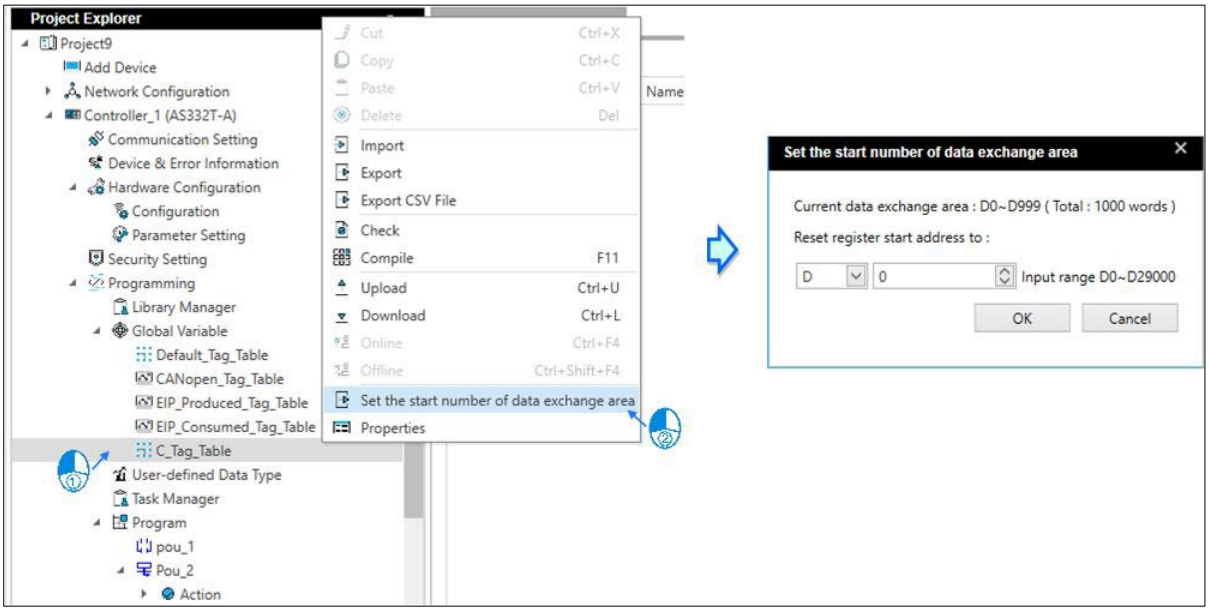


The following table describes the functions in the **C POU** dialog.

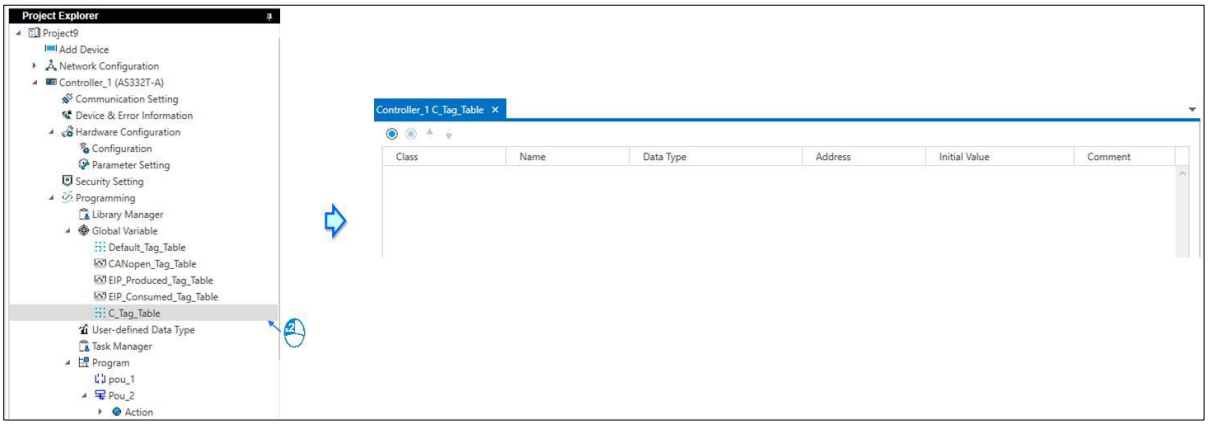
Function	Description
	Add / delete line comment
IF	Conditional structure IF
DO-WHILE	Loop structure DO-WHILE
WHILE	Loop structure WHILE
FOR	Loop structure FOR
SWITCH	Condition Structure SWITCH CASE

Set the start number and C_Tag_Table of the data exchange area

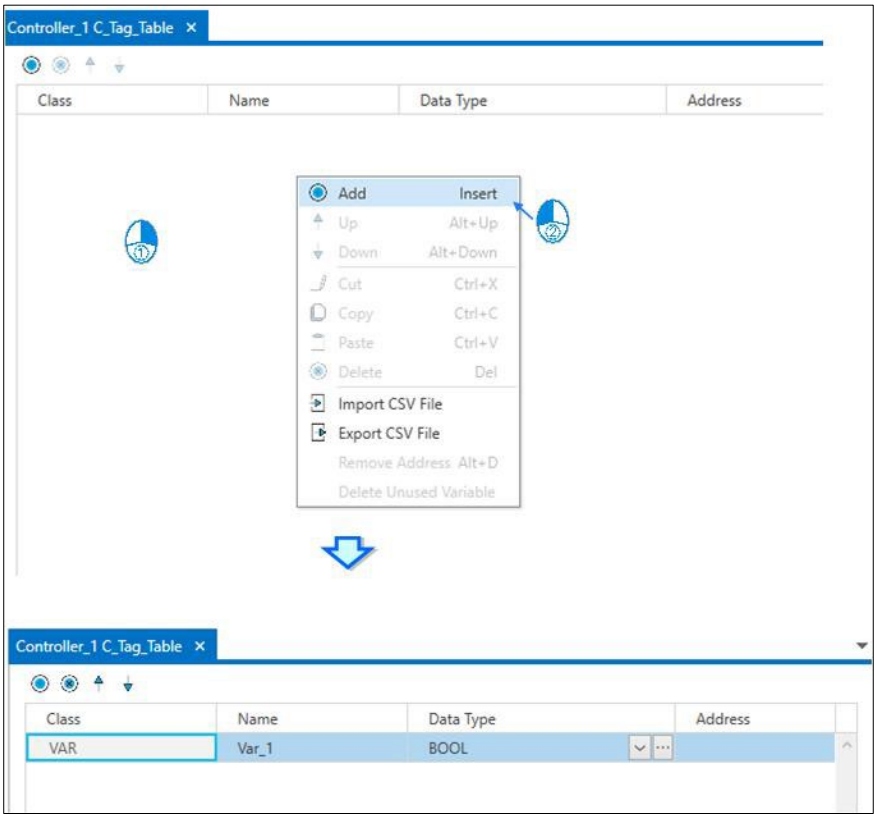
When declaring variables, use matrix (Array) and other methods, but it must be noted that the memory that can be declared for the variables is limited, so it cannot be used beyond the scope, otherwise it may seriously cause the PLC to crash. You must add variable symbols in the C language main symbol table of global symbols. There are 1000 D devices in the device addresses that can be used by C language symbols. Set the starting address range by themselves. Go to **Program Edit** > **Global Variables** > **C_Tag_Table**, right-click the mouse, and then click **Set Data Exchange Area** in the right-click context menu. The start address can be filled in from D0~D29000, and the system automatically sets 1000 consecutive numbers in the D device.



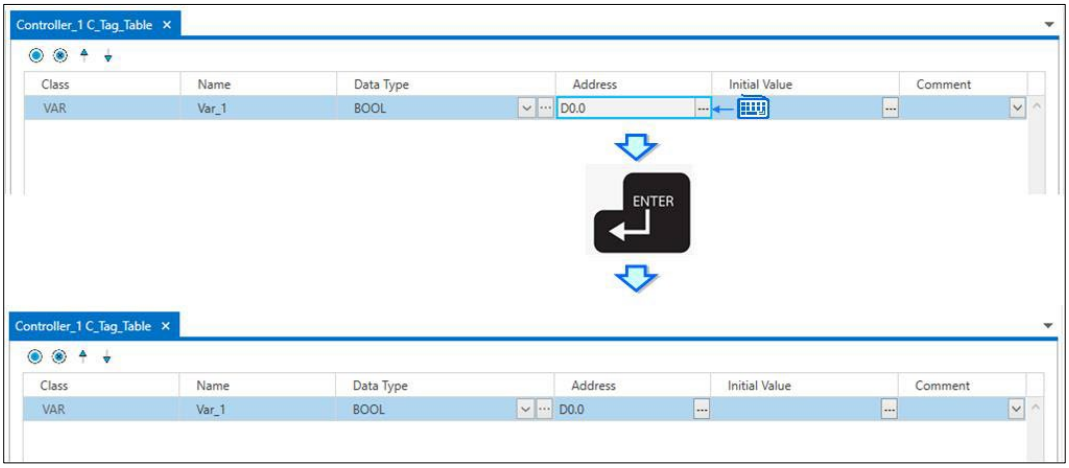
After setting **D0** as the starting position, double-click **C_Tag_Table** and the content of C_Tag_Table appears.



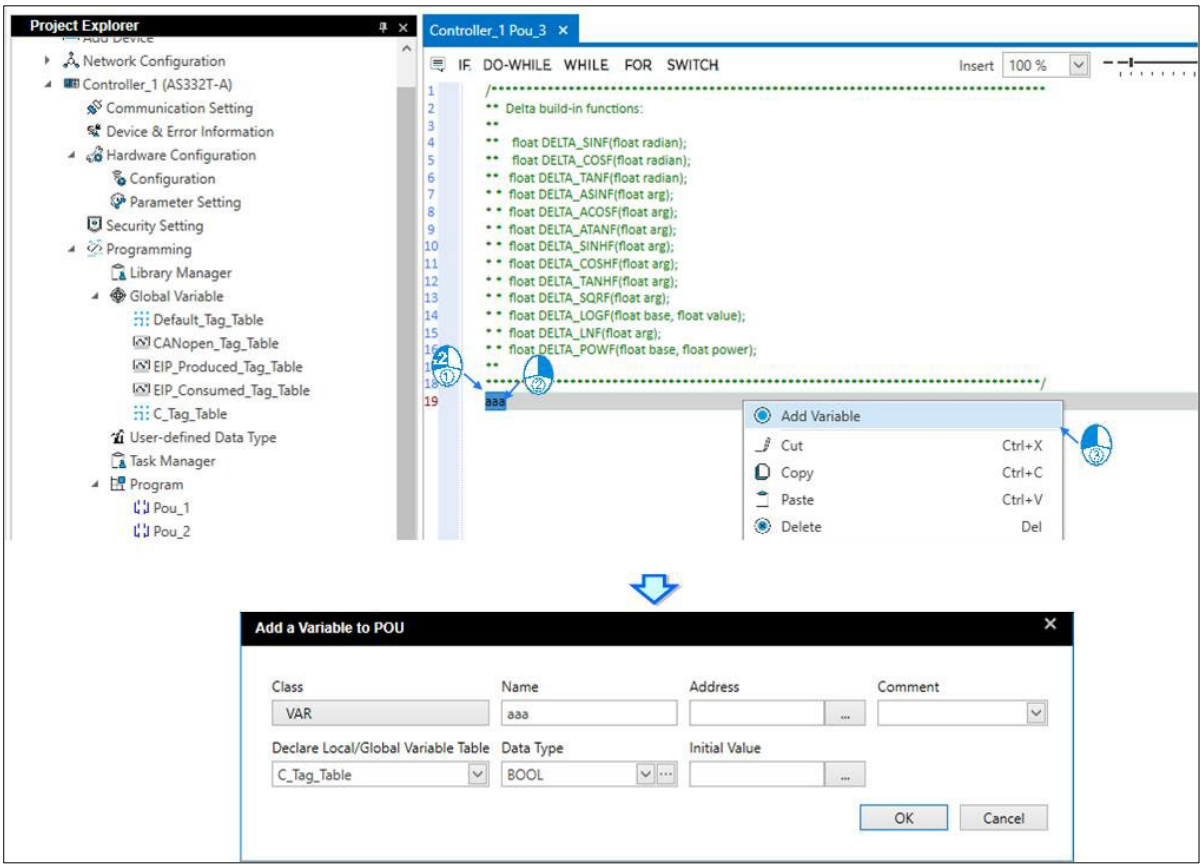
In the blank space of C_Tag_Table, right-click the mouse. Select **Add (Insert)** from the context menu, and the default variables appear for editing.



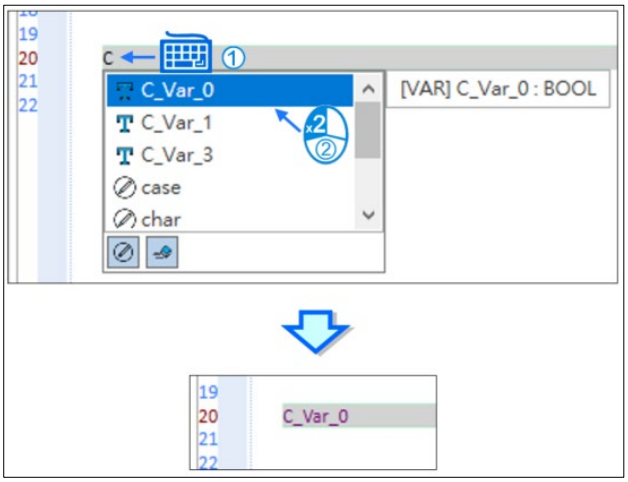
C_Tag_Table supports four types of data types: BOOL, WORD, DWORD and REAL. The starting address of the address is the range set by the C language data exchange area. The example in the figure below shows the D0.0 address setting for the newly added variable. The address field of the variable can only be filled in the range set by the C data exchange area, and this variable can be used directly in the C language program, but also equivalent to a global variable that can be used in other programming languages.



You can also enter undeclared variable symbols in the C language editing area. After entering, double-click the text with the mouse and press the right button, and then click **Add Variable** in the shortcut menu to automatically export the variable symbols. After finishing the setting, the corresponding setting variable appears in C_Tag_Table.



In the process of program input, the matching variable symbols will be automatically filtered from the created variable symbols for you to choose. Use the Up / Down keys on the keyboard or directly click the left mouse button in the menu to complete enter.

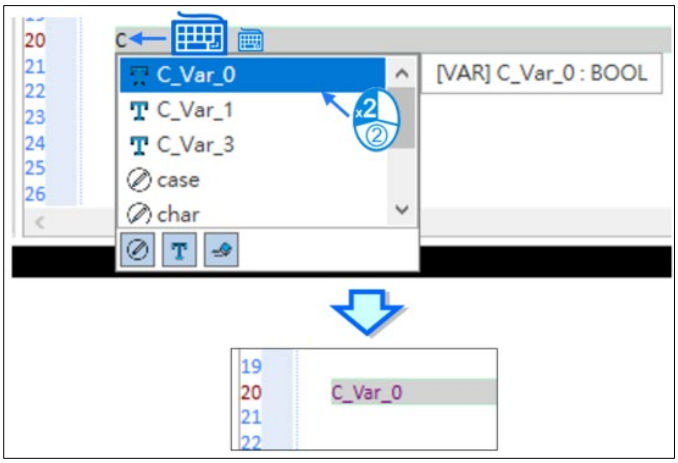


Edit C language

In the C editing workspace, only standard C language writing is supported, other high-level syntax such as PLC instructions, C++ or Visual C, etc. are not supported. The operation method is similar to that of a general text editor. You only need to directly use the keyboard to input or modify text on the work area according to the format of C grammar when editing, and users can execute line break when pressing the keyboard **Enter** key.



In the process of program input, the matching variable symbols will be automatically filtered from the created variable symbols for you to choose. Use the Up / Down keys on the keyboard, or directly click the left mouse button in the menu to complete enter.



Line Comment

After the C language program is created, if the explanatory text or part of the description of the C language program you want to add does not need to be compiled, the line annotation function can be used at this time to eliminate the description that does not need to be compiled. The keyword for line comments is //, the string after this keyword turns green, and the green string represents the comment string that the compiler will ignore.

Follow these steps for adding / deleting line comments.

To add / delete line comments

1. Select one or more lines to be executed.

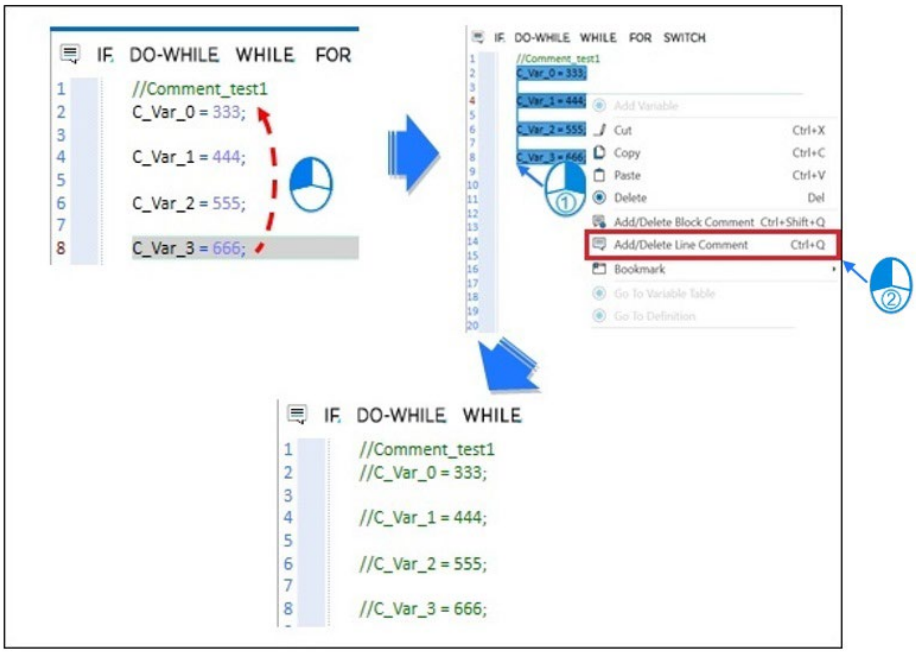
2. Click the pattern button  on toolbar.

   Or

   Right-click to open the context menu and select **Add/Delete Line Comment**.

   Or

   Click **CTRL** + **Q** on the keyboard at the same time to select line. The selected line becomes line comment text (green string).



3. Repeat the above steps to delete the line comment keywords.

Block comment

Block comments provide another way of commenting besides line comments. The keywords of the block comment are /* and */, which are placed before and after the string to be annotated, respectively. The text between these two keywords will turn green, representing the comment string, which will be automatically ignored by the compiler during compilation Comment string.

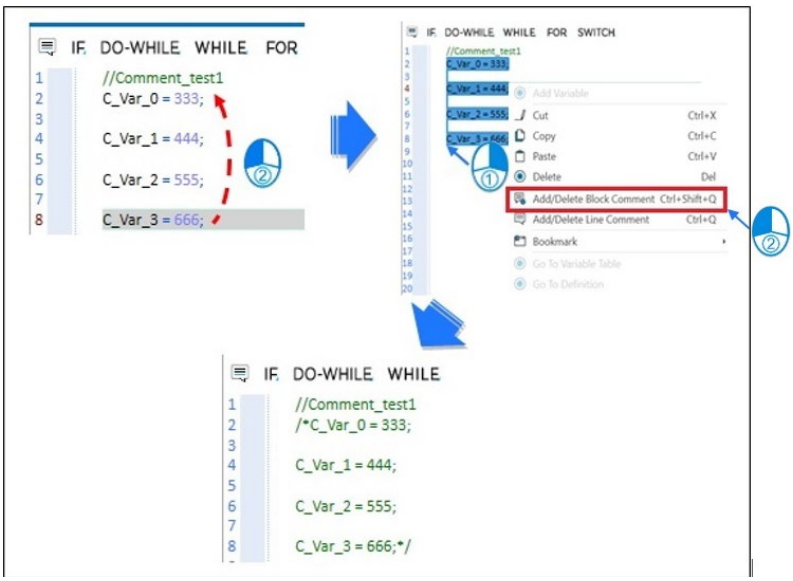
Follow these steps for adding / deleting block comments.

To add / delete block comment

1. Select the string to be annotated.
2. Right-click to open the context menu and select **Add/Delete Block Comment**.

   Or

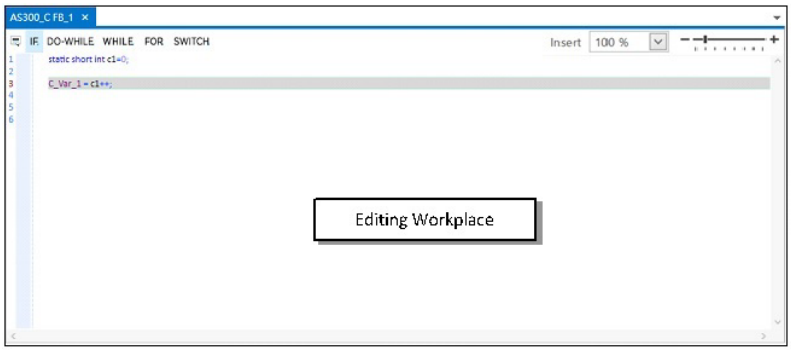
   Click **CTRL** + **SHIFT** + **Q** on the keyboard at the same time to select string. The selected string becomes block comment text (green string).



3. Repeat the above steps to delete the block comment keywords.

Function block (FB) call structure

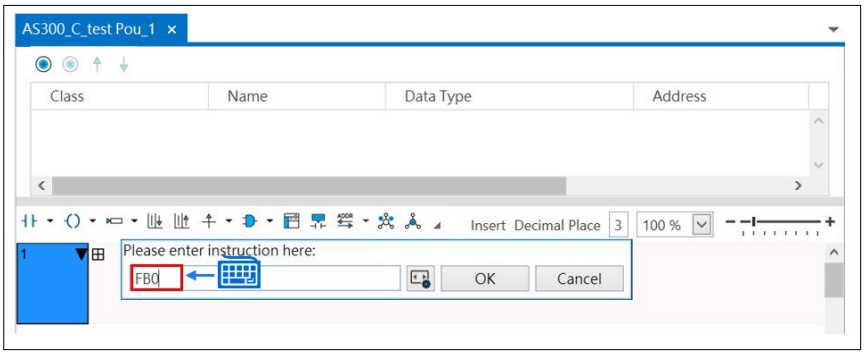
The editing method of C function block is the same as that of C program. According to the format of C syntax, directly use the keyboard to enter or modify text in the work area.



There are two ways to insert C language function block into LD language program.

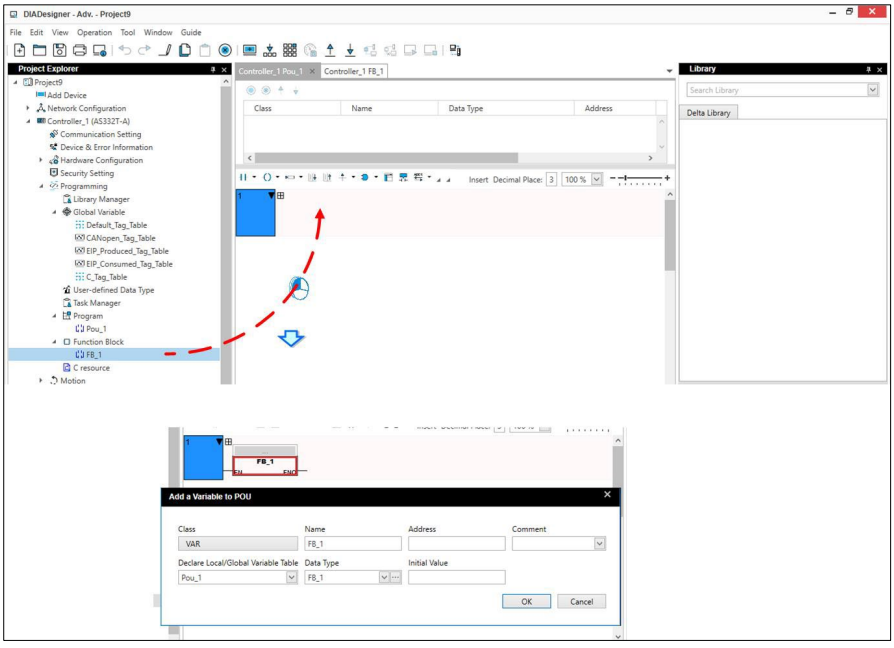
Method 1

According to the function block name, directly use the keyboard to input in the editing work area of LD language.



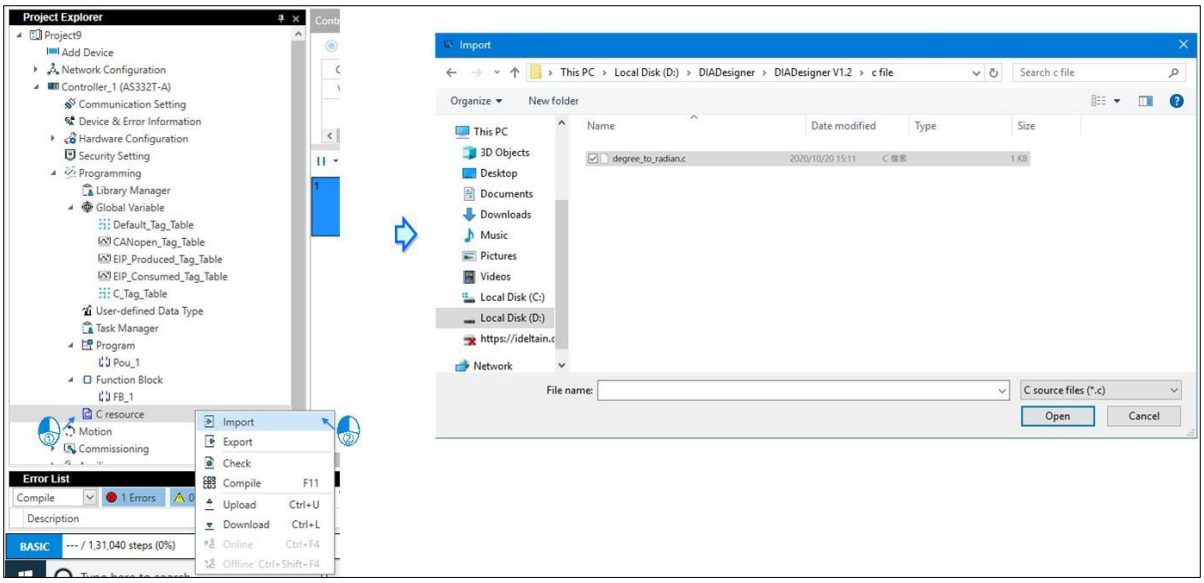
Method 2

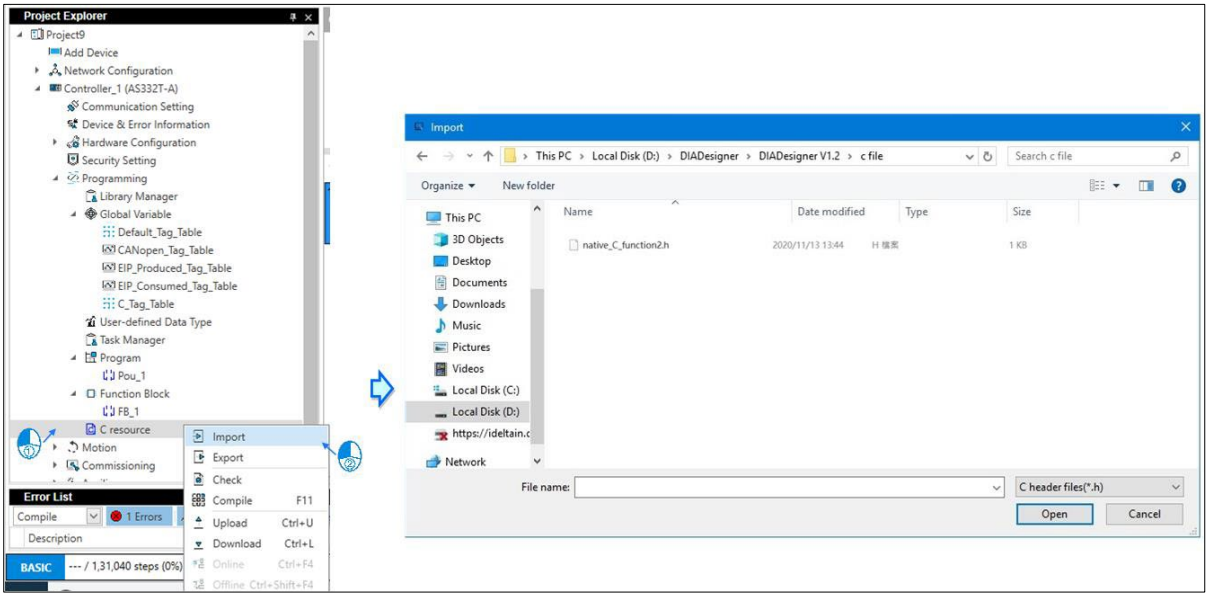
Expand the list of function blocks in **Project Explorer**, find the item you want to insert in it, press and hold the left mouse button on the item, then drag it to the position you want to insert and release the left button.



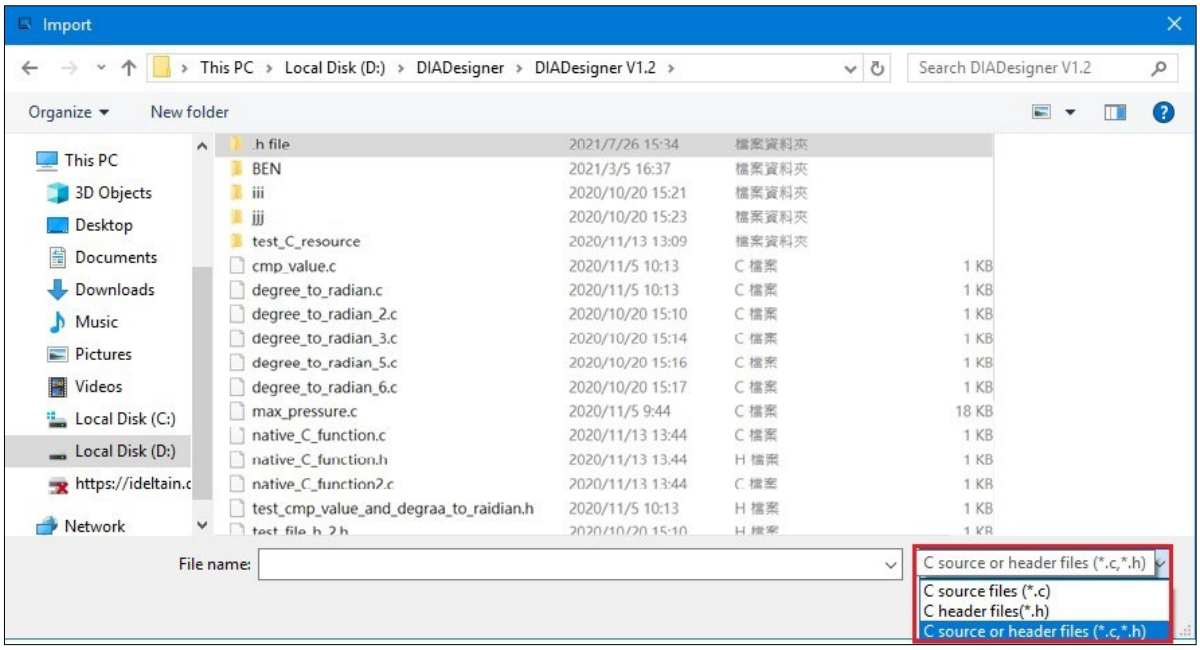
Import / Export C resource file content

To import the completed C language .**c** and .**h** files, right-click the **C resource file**, then click the **Import** item in the context menu, and then select the file to be imported. Add the C file and select the C file to be imported, as shown in the figure below. To import .H files, select C headers (*.h) in the **File Name** item.

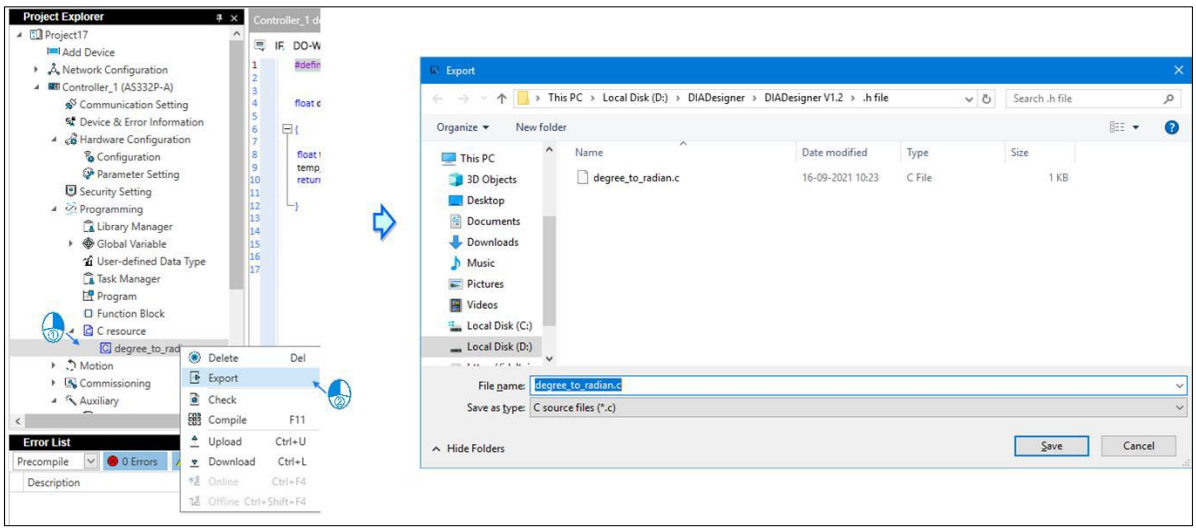


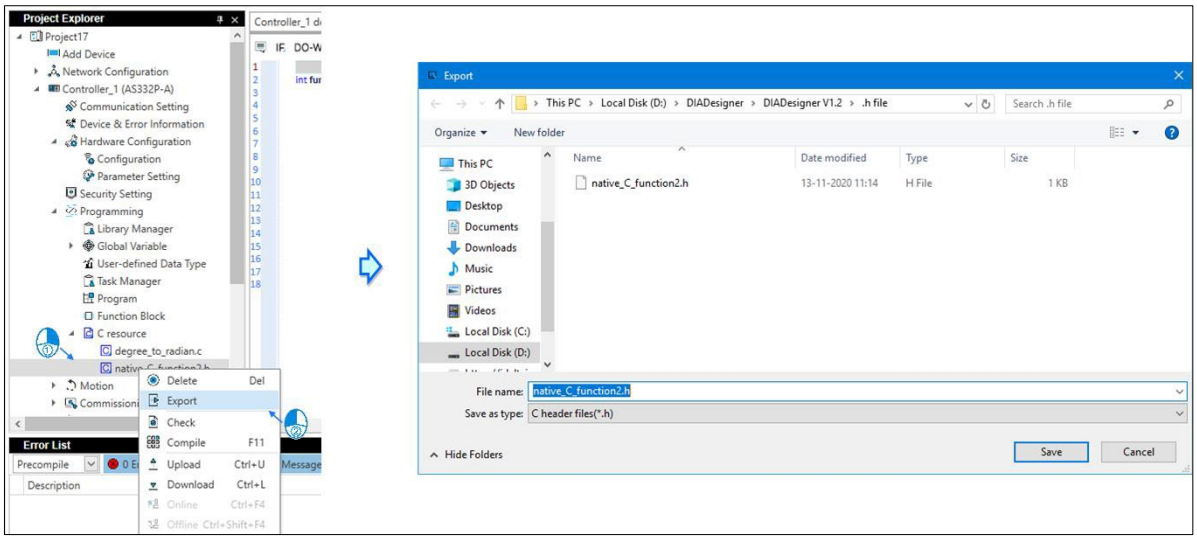


You can also choose to display both the .**c** and .**h** files. At this time, the .**c** and .**h** files can be selected for batch import.



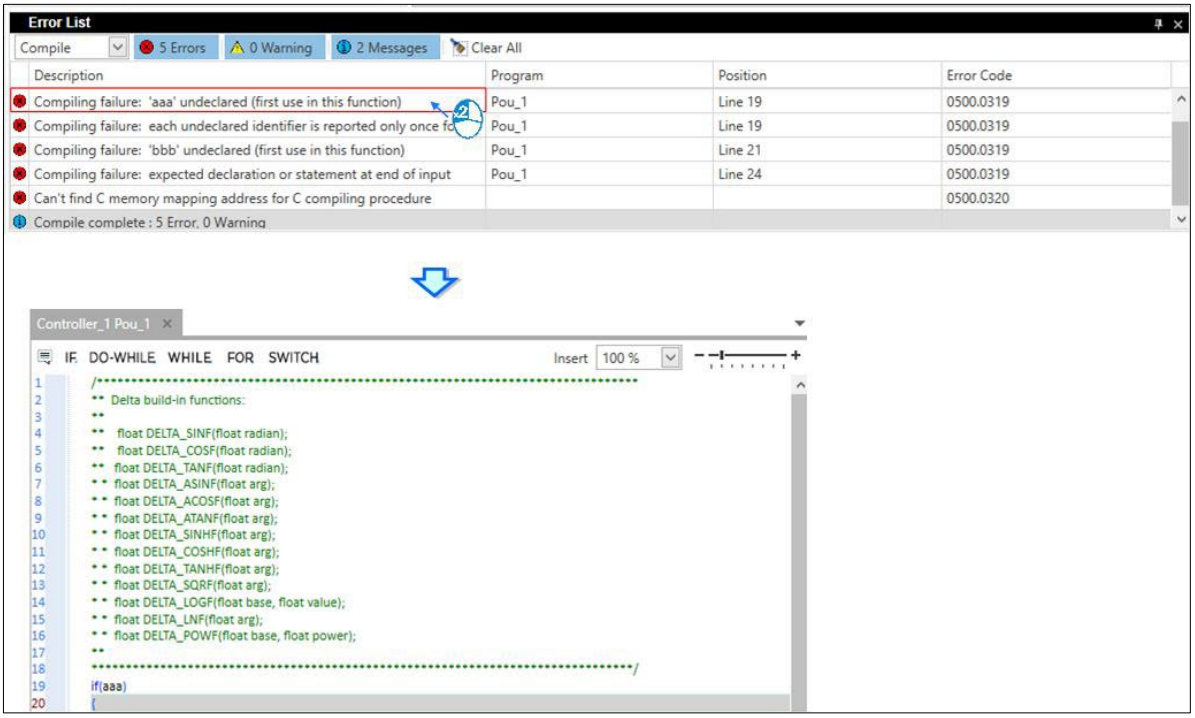
Similarly, you can also click the **Export** item in the C resource file to do the export, or directly click the **Export** item in the .c/.h file to do the export.





8.3.4.3 C Language Compilation and Debugging

DIADesigner provides PLC built-in trigonometric function and exponential library, which can be used by you without announcing, but it does not provide other external standard C library (such as math.h) on the market to save PLC internal provision C code storage space. DIADesigner will display the compiled error message. After double-clicking the error description, jump directly to the error location to modify the function. Monitor the existing PLC programs and function blocks online, but cannot perform line by line editing of the C language program. C Code for monitoring.



8.3.4.4 Built-in Library Command Description

The four arithmetic add, subtract, multiply, and divide commonly used integer type or floating-point number type are all built-in in DIADesigner and can be used without additional declaration.

The following table lists the name of the built-in trigonometric function.

C Function Name	DIADesigner Edit Name	Instruction Format	Function Description
sinf	DELTA_SINF	Float DELTA_SINF (float radian);	Sin operation
cosf	DELTA_COSF	Float DELTA_COSF (float radian);	Cos operation
tanf	DELTA_TANF	Float DELTA_TANF (float radian);	Tan operation
asinf	DELTA_ASINF	Float DELTA_ASINF (float arg);	Sin^{-1} operation
acosf	DELTA_ACOSF	Float DELTA_ACOSF (float arg);	Cos^{-1} operation
atanf	DELTA_ATANF	Float DELTA_ATANF (float arg);	Tan^{-1} operation
sinhf	DELTA_SINHF	Float DELTA_SINHF (float arg);	Sinh operation
coshf	DELTA_COSHF	Float DELTA_COSHF (float arg);	Cosh operation
tanhf	DELTA_TANHF	Float DELTA_TANHF (float arg);	Tanh operation

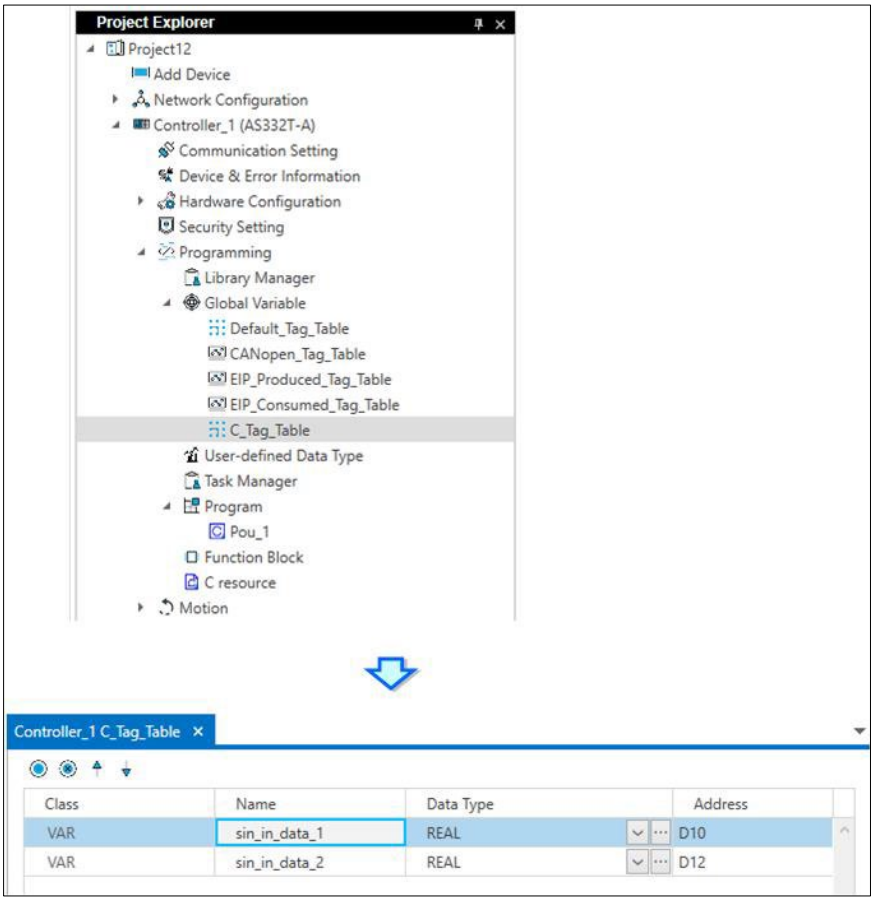
The following table lists the name of the built-in exponential function.

C Function Name	DIADesigner Edit Name	Instruction Format	Function Description
sqrf	DELTA_SQRF	Float DELTA_SQRF (float value);	Square root operation
logf	DELTA_LOGF	Float DELTA_LOGF (float base，float value);	Log operation
lnf	DELTA_LNF	Float DELTA_LNF (float arg);	Ln operation
powf	DELTA_POWF	Float DELTA_POWF (float base，float power);	Exponentiation

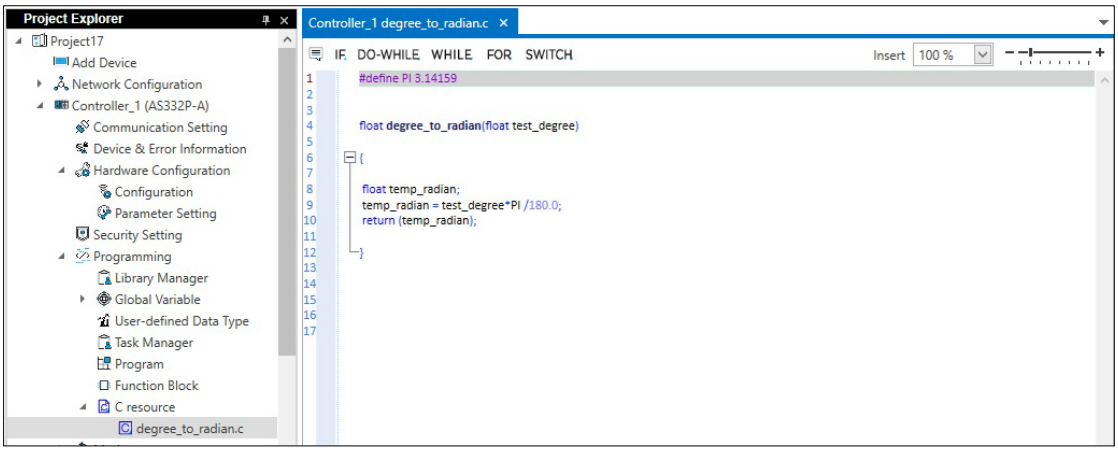
8.3.4.5 Example Operation Instructions

Example: Output a Sin function value every 10ms with each 5-degree transformation and use the data oscilloscope function to record the output value of the Sin function.

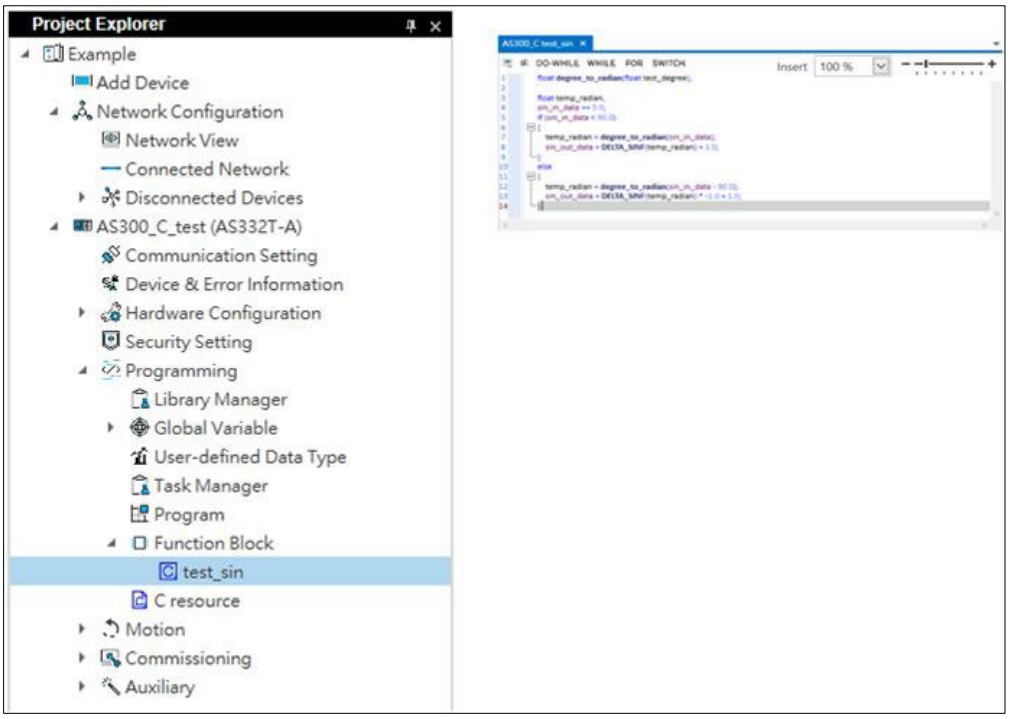
1. First create the variable names of the variables shared by controller and C language program, refer to C Language Editing Environment Description.



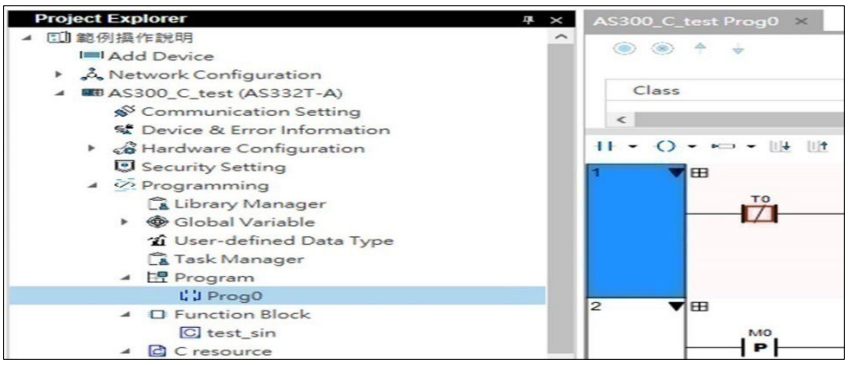
2. Import a function degree_to_radian.c that you have edited and completed the test. Refer to C Language Editing Environment Description. This step can be omitted according to the situation.



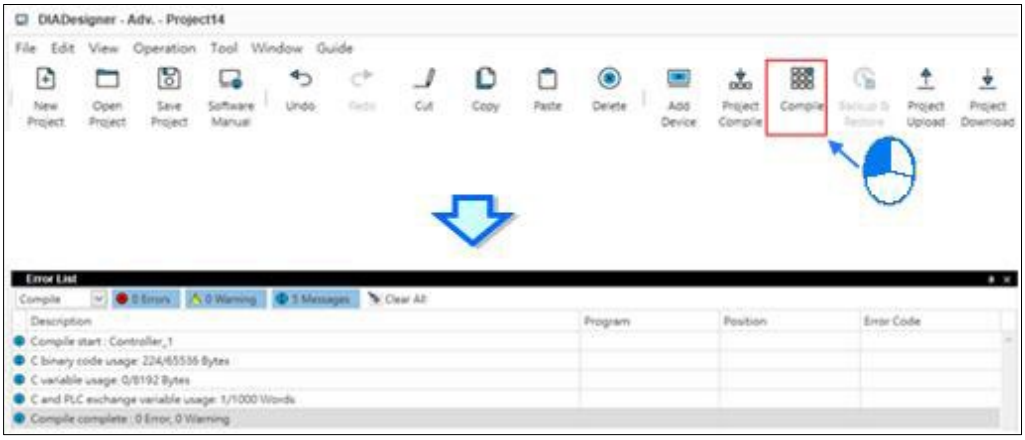
3. Edit a new function block test_sin and write in the calculation process, including degree_to_radian function library as shown in the following figure.



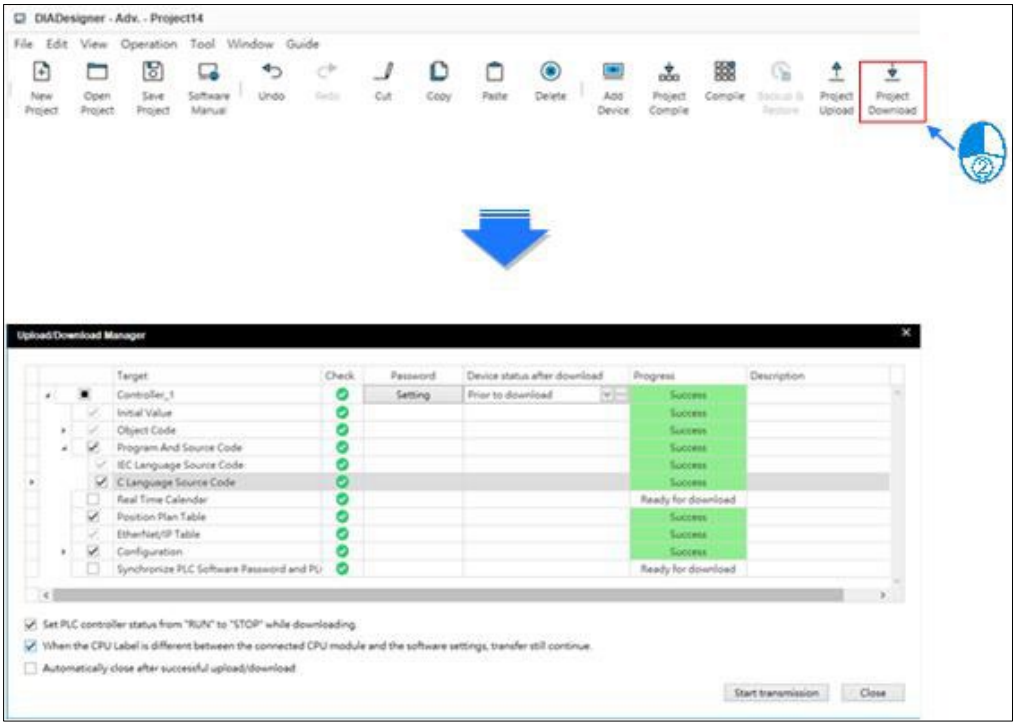
4. Create a program (POU), and write a function block (FB) that calls test_sin every 10ms.



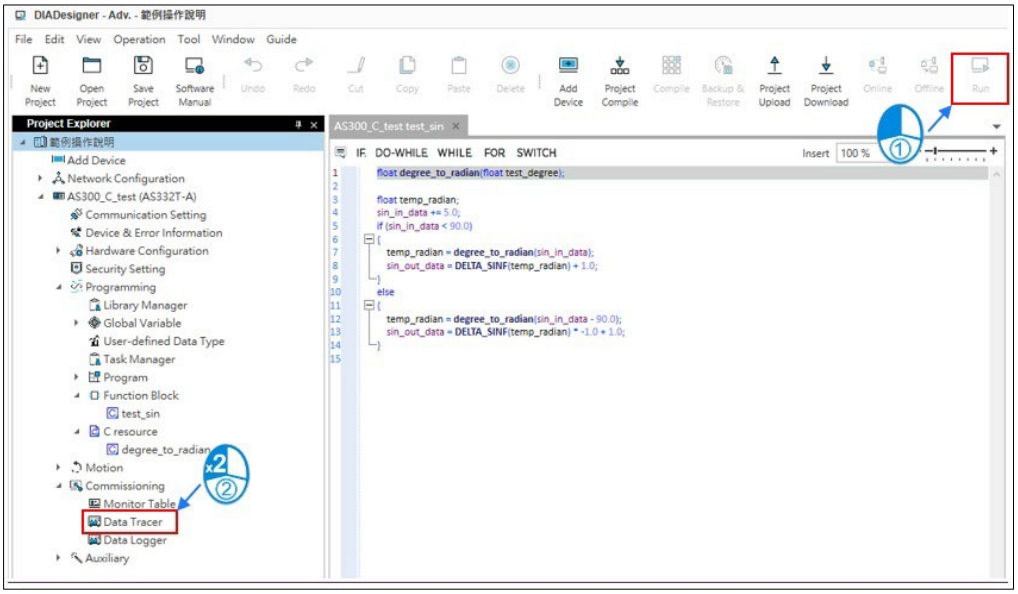
5. Click **Compile** .



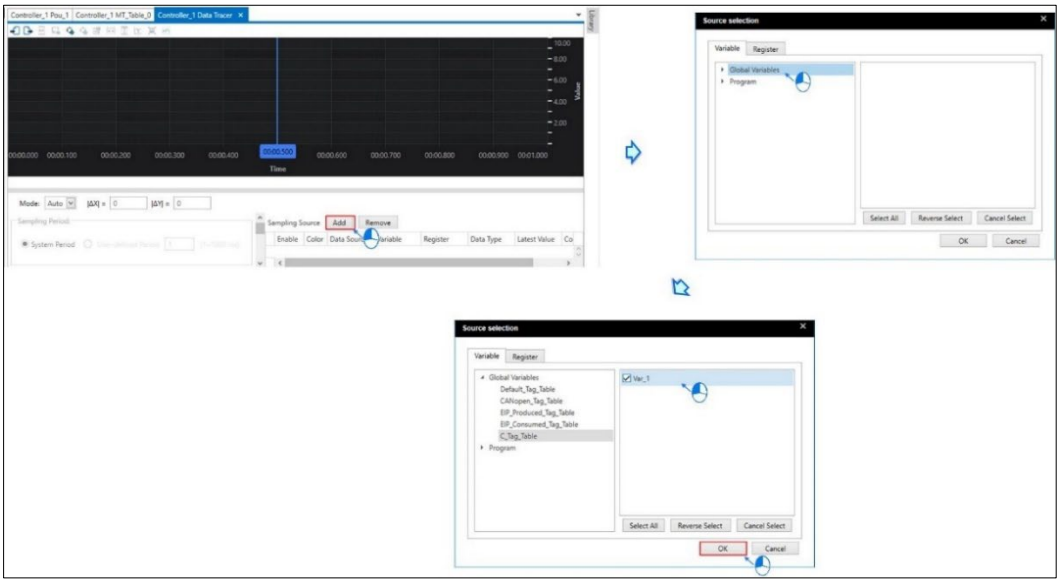
6. After the compiling (Compiler) is successful, the communication is set and the controller program can be downloaded.



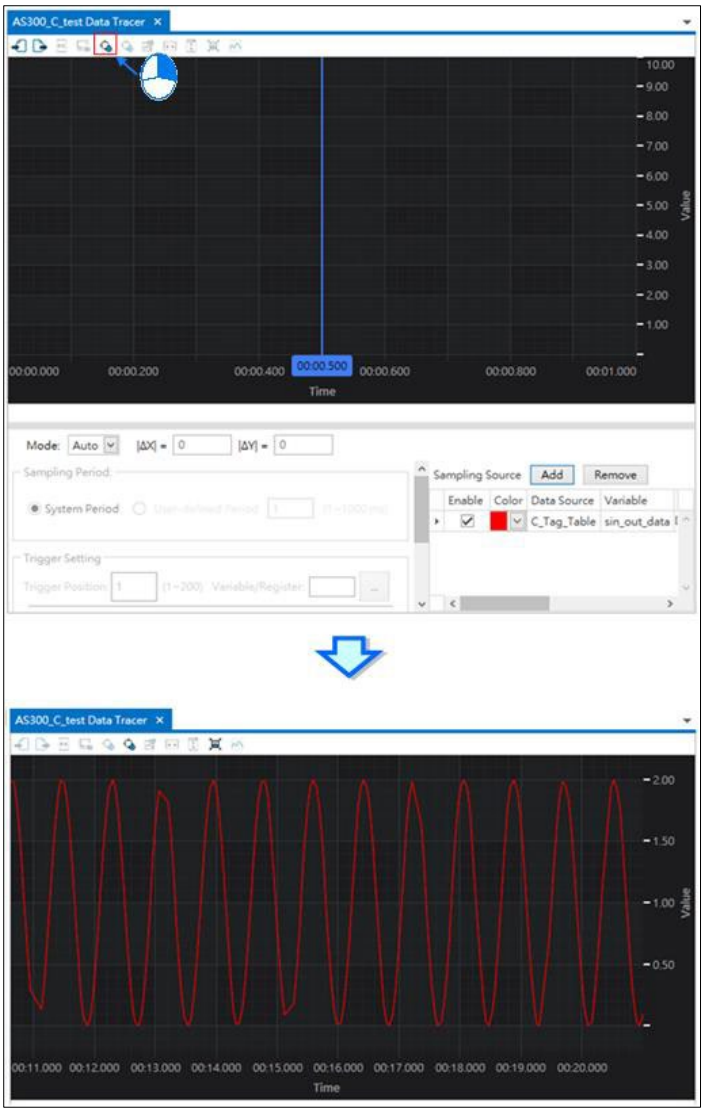
7. After starting PLC **Run**, turn on the **Data Tracer**.



8. The Data Tracer sets sampling parameters and measurement. Click **Add**, double-click **Global Variables**, click **C_Tag_Table** in the **Select Variables** dialog, and select the variables to be monitored. Then click **OK**.



9. Click **Start Tracer** to perform real-time monitoring of the Data Tracer.



8.3.5 CFC—Continuous Function Chart

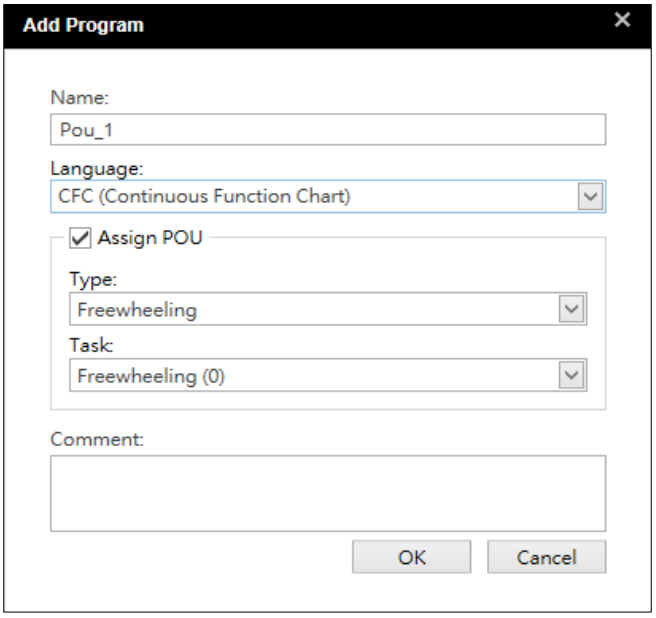
The **Continuous Function Chart (CFC)** is a PLC programming language in extension to the IEC 61131-3 standard. CFC is often used in the area of motion control. The most significant feature is its graphical programming language based on the function block diagram language. Comparing to other PLC programming languages, CFC can provide a more logical relation between input and output, and a clearer execution order of the program. The operation principle of CFC will not be covered in this chapter; this chapter will only focus on the creation and operation of CFC in DIADesigner.

8.3.5.1 Things to Note When Using CFC

- It supports keyboards but does not support the instructions of LD, LDI, LDP, LDF, NP, PN, etc. Type the register or the name of the variable, and the system will add input node in the back.
- Entered data is not case-sensitive, any uppercase or lowercase character can be entered. For example, OUT, Out and out are all correct.
- Common positional numeral systems
  - Decimal range: 2345
  - Hexadecimal range: 16#5BA0
  - Floating point number: 4.123
- There is no limit on the number of the loops for CFC to edit. But the size of an edited programs should not be larger than the memory size.
- The input pin of nodes, logic gates, instructions, and function blocks can only be from one origin. But the targets of the output pin can be multiple.

8.3.5.2 Add CFC Program

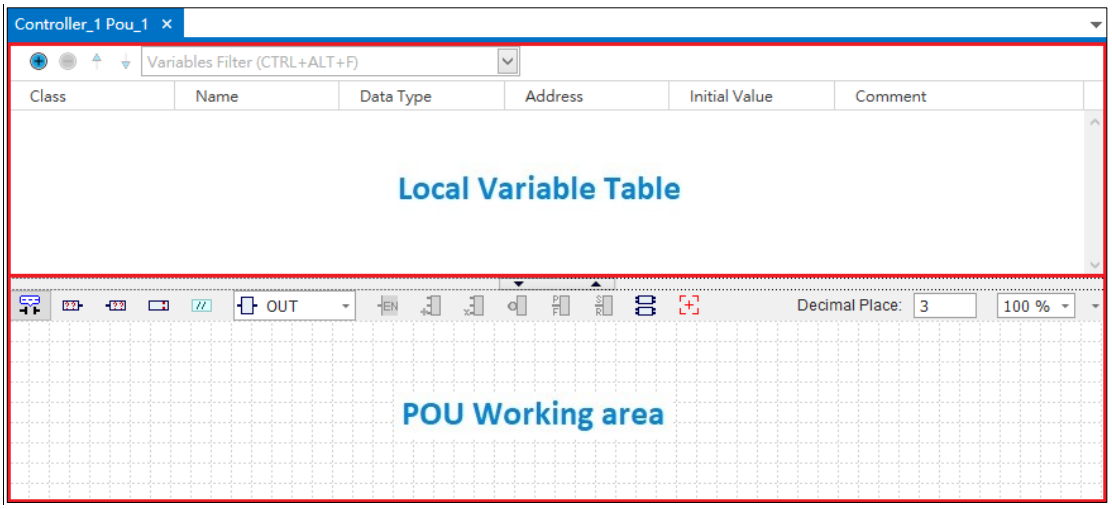
Select **CFC (Continuous Function Chart)** in the **Language** field in **Add Program** dialog while creating a POU of program type or function block type.



The environment in which a CFC POU can be edited is shown in the following figure. In any POU dialog (program type or function block type), there are two areas.

- Local Variable Table area
- POU Working area

The table in the upper part of the CFC POU dialog is a Local Variable Table, and the lower part is POU Working Area. After a CFC POU program editing dialog is opened, the corresponding toolbar appears in the DIADesigner dialog. A blank Network is created by default.



CFC Toolbar

After the POU dialog is opened, the toolbar for editing the Continuous Function Chart appears.



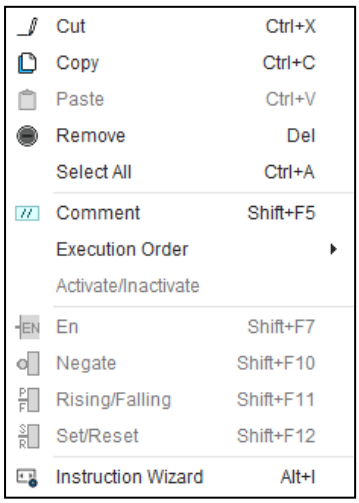
The following table describes the functions in CFC POU dialog.

Function / Icon	Description
	Click to display / hide the commands on the registers.
	Click to insert an input node.
	Click to insert an output node.
	Click to insert a label.
	Click to insert a comment.

Function / Icon	Description
	Click to select a CFC block. You can insert OUT, AND, OR and other blocks from the drop-down menu.
	Click to enable / disable function blocks or the EN pin of the instruction.
	Click to add an input node to the AND / OR block.
	Click to delete the input node from the AND / OR block.
	Click to insert inverse logic.
	The input is rising / falling.
	Click to set / reset the input.
	Auto Arrange: All the objects will be sorted according to the data flow.
	Click to auto resize to fit the screen to see all objects.
	Options: Click to select the execution order. You can select Auto Data Flow Mode or Explicit Execution Order Mode.
	Click to toggle the label.

Context Menu for CFC Programming

Right-click in the working area to open context menu.



The following table describes the functions in the context menu.

Function	Description
Cut	Click to cut the selected object.
Copy	Click to copy the selected object.
Paste	Click to paste the lately cut / copied object to the editing position.
Remove	Click to delete the selected object.
Select All	Click to select all objects in the POU working area.
Comment	Click to insert a comment at the cursor position.
Execution Order	• When the option is set to Auto Data Flow Mode: Switch the display of execution order. • When the option is set to Explicit Execution Order Mode: Set up the execution order of the logic gates, instructions, and function blocks. Data flow: all objects will be run according to the original set order. Topology: all the objects will be sorted and executed in an order of top-to-bottom and left-to-right.
Activate/Inactivate	Click to activate / inactivate the selected object.
EN	Click to enable / disable function blocks or the EN pin of the instructions.
Negate	Click to insert inverse logic.
Rising/Falling	The input is rising / falling.
Set/Reset	Click to set / reset the input.
Instruction Wizard	Open Instruction Wizard dialog for selecting the instruction.

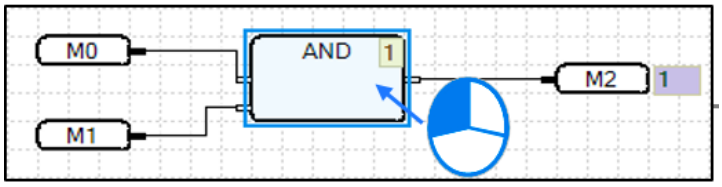
8.3.5.3 Edit CFC

Selecting Objects

Select the objects in the CFC by mouse-clicking. After the objects are selected, perform edit, copy, paste, etc.

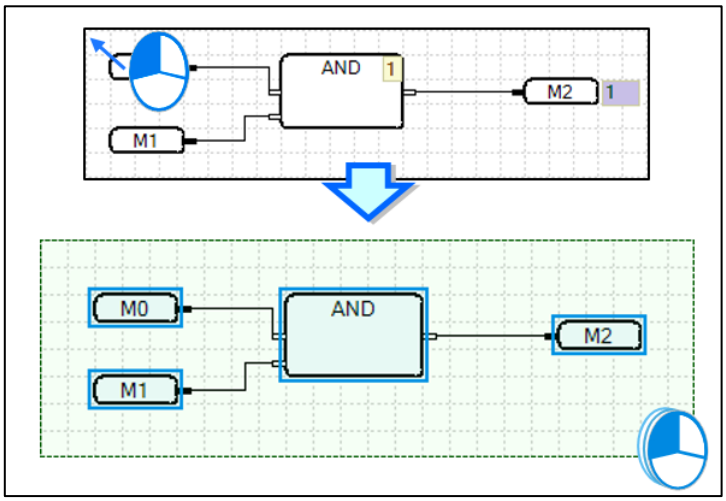
- **Select an object**

  Move the cursor to the object to be selected and left-click, or use the directional buttons on the keyboard to select the object. The selected object will be in blue; as for unselected object, it is in white as shown in the following figure.



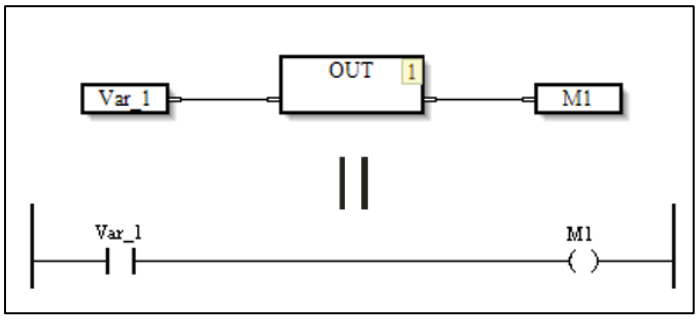
- **Select multiple objects**

  Click and drag the mouse to select the area of objects.

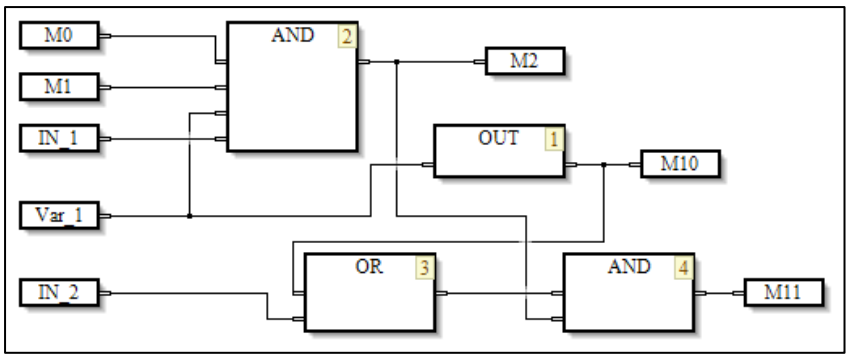


Input / Output Nodes and Logic Gates

Nodes and logic gates are factors to construct a CFC. Input nodes can be the variable status of the input origin or the status of the register. The output nodes are the results from the operation of logic gates. The following example illustrates the same program in two programming languages, the CFC and LD. Input nodes and output nodes cannot be linked directly; that is why logic gates are needed.



Use logic gates such as AND gate and OR gate to link the objects to create various complicated loops as shown in the following figure.



Inserting Nodes or Gates

Insert a node using the following two methods as required.

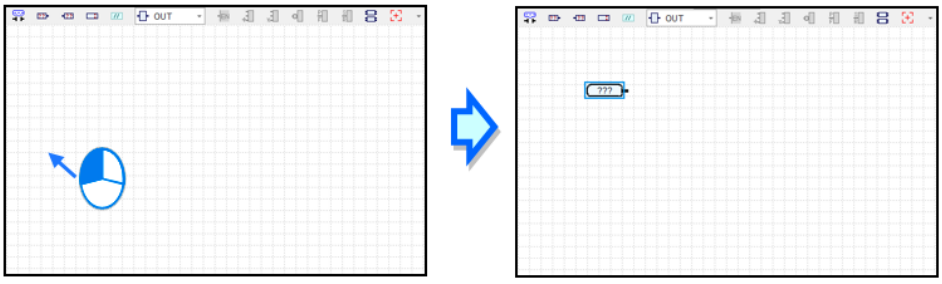
To insert node

Method 1: Use the icons from the toolbar

1. Click  on toolbar.

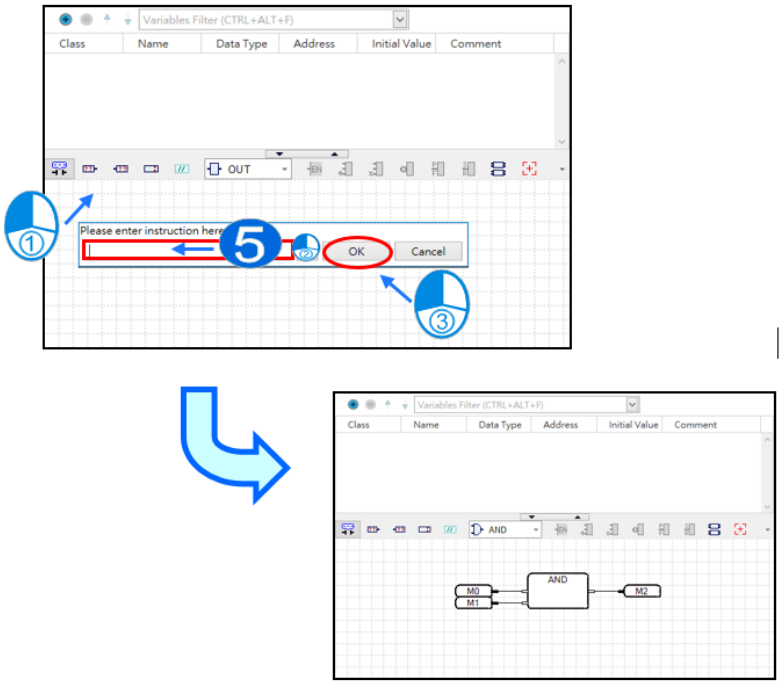


2. Click any position where you would like to insert the object in the working area.



Method 2: Use a keyboard

1. Click in the working area and enter instruction using the keyboard directly. After pressing **Enter** on the keyboard or clicking **OK** on the instruction editing dialog, the instruction appears.
2. Click any position where you would like to use the instruction, and then the instruction creation is completed.



Note:

- The instruction entered is not case-sensitive.
- When unsupported instruction or the contents of non-compatible devices are entered, they will be regarded as variables. After clicking **OK**, **Add a Variable to POU** dialog appears. If the content contains special characters, such as ABC??!!, this dialog does not appear.

The following table lists the supported instructions of keyboard.

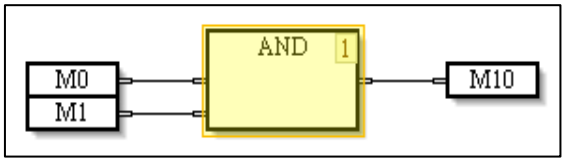
Instruction	Description
OUT	Create an OUT gate.
AND	Create an AND gate.
OR	Create an OR gate.
Name of a register	Create an input node for the register.
Name of an instruction	Create an instruction block.
Name of a function block	Create a function block.

Add / Delete a Pin

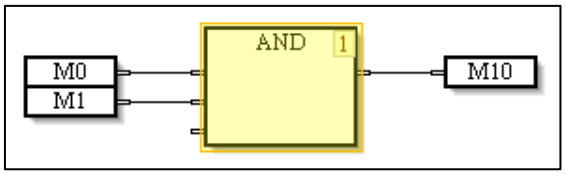
AND and OR gates can be used for adding and deleting a pin, suitable for programs with multiple conditions.

To add / delete Pin

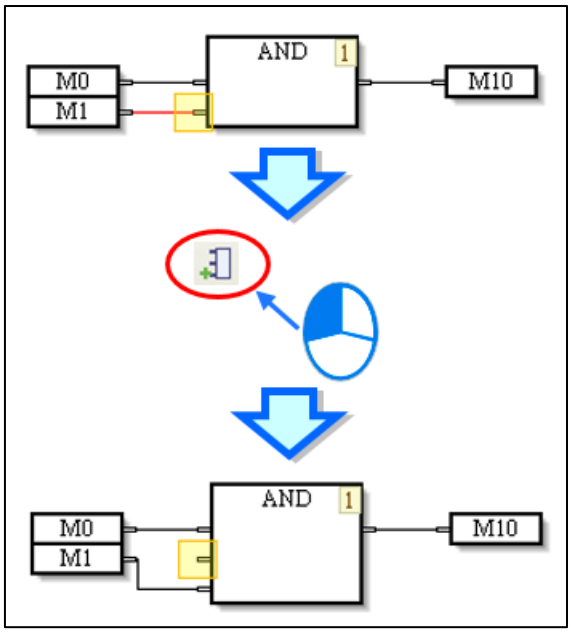
1. Click to select the logic gate.



2. Click **Add Pin**  on toolbar to add a pin to the logic gate.



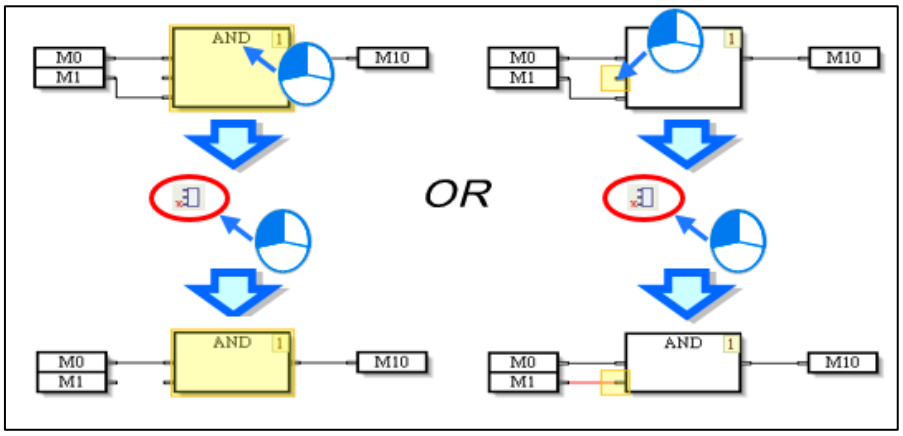
3. If a pin needs to be added before a specific pin, click to select the pin to add a new pin before it, and then click **Add Pin**. A newly added pin appears right before the previously selected pin, as shown in the following figure.



4. Click **Remove Pin**  to delete an input node from the AND / OR gates

   Or

Click and select a specific pin and click **Remove Pin** to delete.

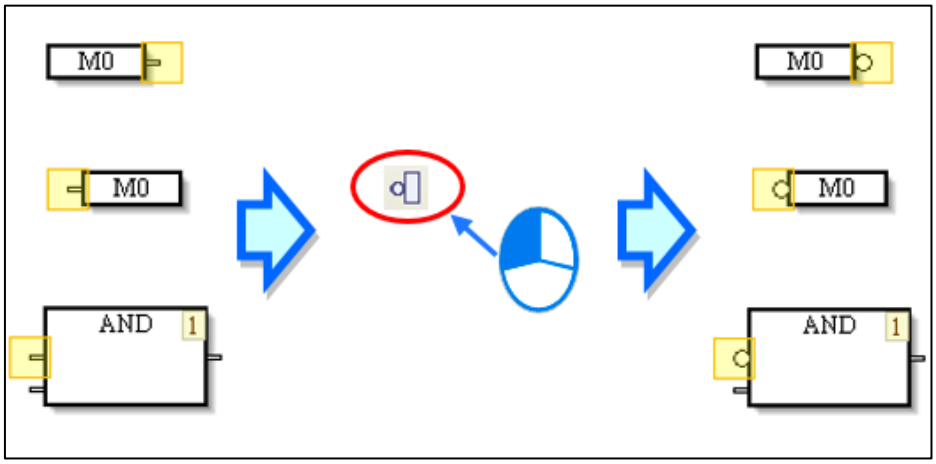


Changing a Pin Type

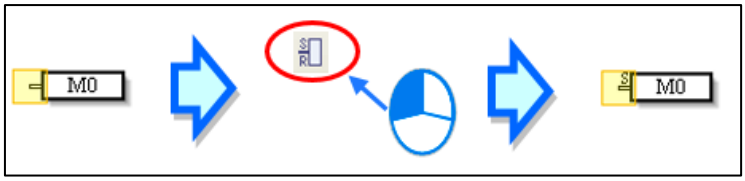
Follow these steps to change a pin type and design a complex continuous function chart.

To change pin type

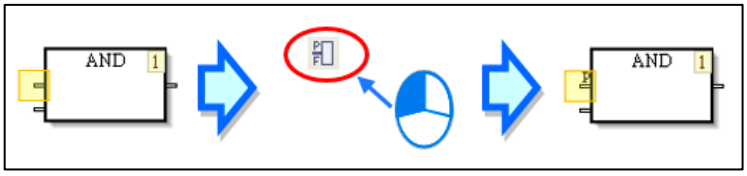
1. For negating, select the pin of a node, a pin of a logic gate, a pin of a function block, or a pin of an instruction, and then click **Negate** on toolbar or the context menu, the state of the pin will be negated. If you select a pin whose state is negated, and then click **Negate** again, the state of the pin will not be negated.



2. For setting a pin, select the pin of an output node, and then click **Set/Reset** on toolbar or the context menu, the pin will be set. For resetting a pin, if you select a pin which is set, and then click **Set/Reset**, the pin will be reset. If you select a pin which is reset, and then click the icon again, the pin will not be reset.

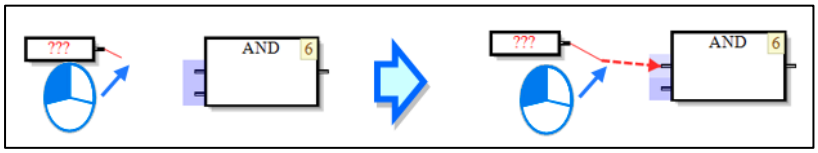


3. For rising, select a pin of a logic gate, a pin of a function block, or a pin of an instruction, and then click **Rising/Falling** on toolbar or the context menu, the pin will be rising-edge triggered. For falling, when you select a pin which is rising edge-triggered, and then click **Rising/Falling**, the pin will be falling edge-triggered. If you select a pin which is falling edge-triggered, and then click the icon again, the pin will not be falling edge-triggered.



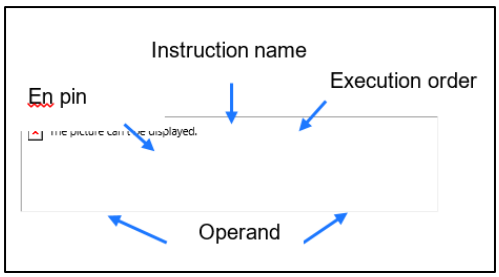
Connecting Objects and Canceling Connections

When you click a pin of an object and hold down the left mouse button, the target pins to which the pin can be connected are indicated by gray. When you drag the mouse cursor to the target pin of an object, an arrow appears. After you release the left mouse button, the two pins will be connected. To change the connection or cancel the connection, drag the line to a new target pin or to the blank.

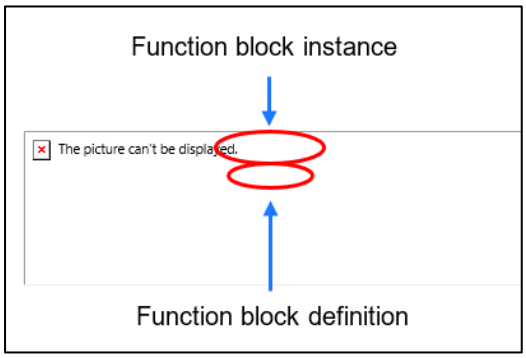


Instructions and Function Blocks

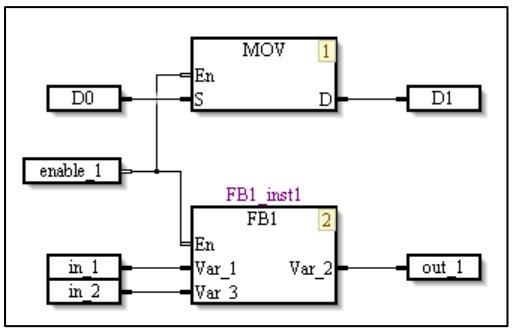
In a continuous function chart, an instruction name and operands are shown in a block. The orders in which the instructions in a continuous function chart are executed are indicated by the numbers in the upper-right corners of the blocks. An instruction is executed only when the logic state connected to the En pin of the block is ON.



The following figure shows a function block in a continuous function chart. It is executed only when the logic state connected to the En pin is ON. Besides, the text on the top of the function block is the name given to the function block, i.e., a **Function Block Instance**.

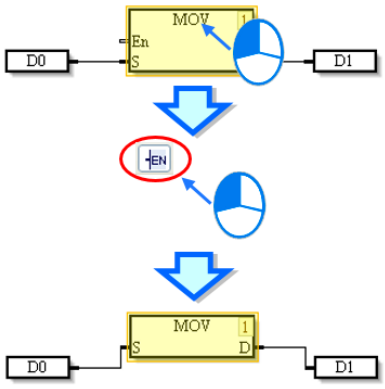


If you need to enable an instruction and a function block by a condition, connect the input node of the condition to the En pin of the block representing the instruction and the En pin of the function block. When the input node is ON, the instruction and the function block is executed in order. From the following figure, enable_1 controls the instruction and the function block.



If an instruction or a function block does not need to be controlled by a condition, select the block representing the instruction or the function block, and then click **EN**  on toolbar or the context menu. After the En pin of a block representing the instruction or the En pin of a function block is disabled, the instruction or the function

block will be executed if it is the instruction's turn or the function block's turn to be executed. If you select an instruction or a function block whose En pin is disabled, and then click this icon again, the En pin of the instruction or the En pin of the function block will be enabled.



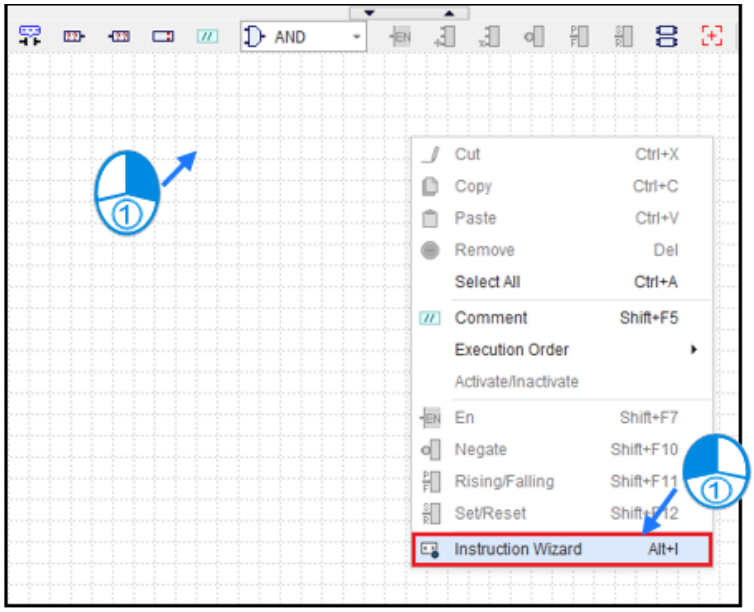
Inserting an Instruction

Follow these steps to insert an instruction using any of the following two methods.

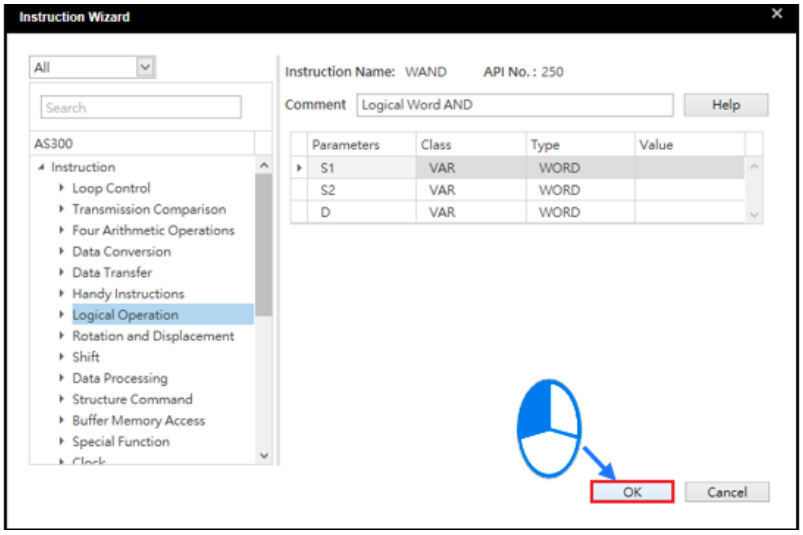
To insert instruction

- **Method 1**: Select from Instruction Wizard dialog.

  1. Right-click the working area and click **Instruction Wizard** in the context menu.

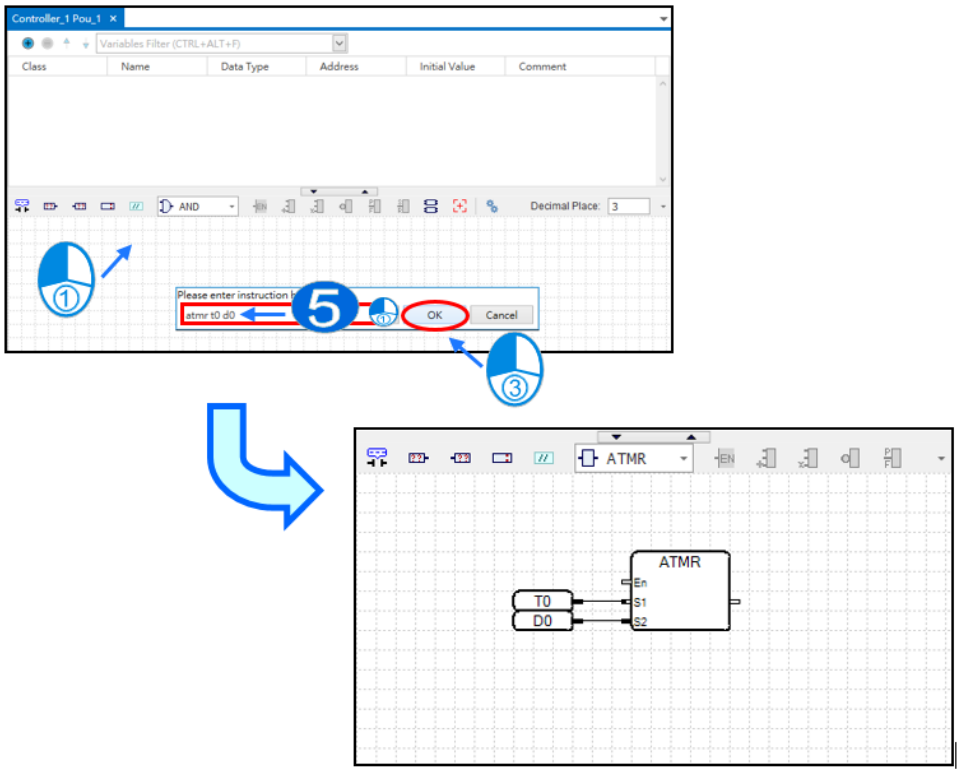


  2. In the **Instruction Wizard** dialog, select an instruction to be added and click **OK**.



- **Method 2**: Use a keyboard

  Click the working area and type an instruction by the keyboard directly, the instruction edit dialog appears. After typing, press the **Enter** key on the keyboard or click **OK** in the instruction edit dialog. If you move the mouse cursor and click any position where the instruction is to be added in the working area, the instruction will be inserted.



Note:

- The instruction entered is not case-insensitive.
- When unsupported instructions or the contents of non-compatible devices are entered, they will be regarded as variables. After clicking **OK**, **Add a Variable to POU** dialog appears. If the content contains special characters, such as ABC??!!, this dialog does not appear.

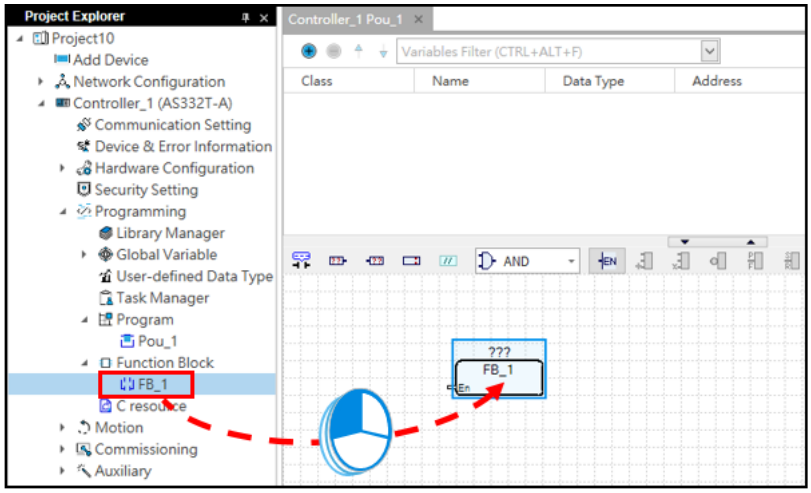
Inserting a Function Block

Follow these steps to insert a function block using any of the following methods.

To insert function block

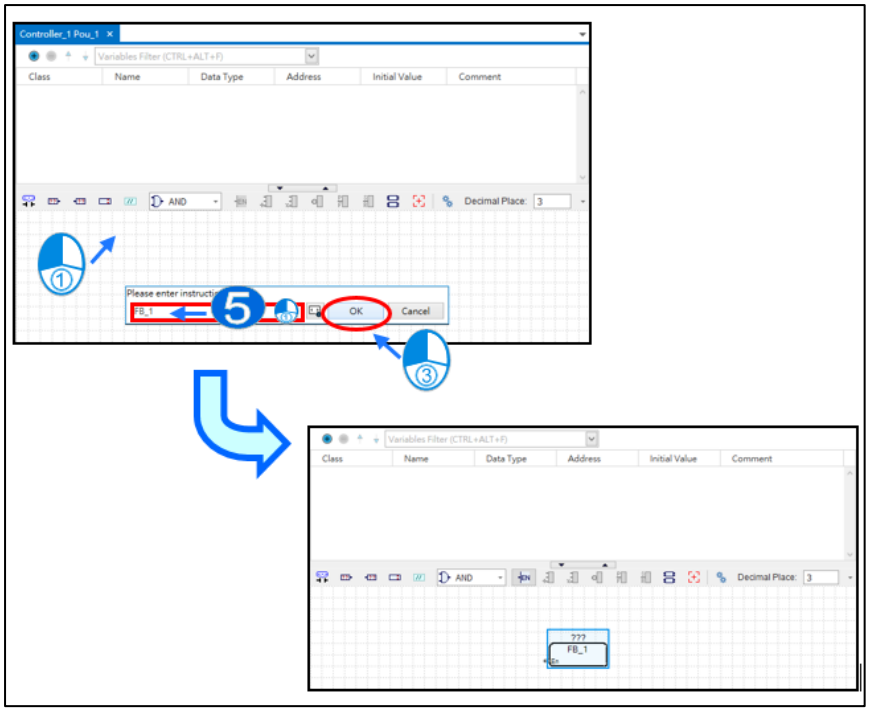
Method 1: Dragging a function block definition

Click the function block definition to be added and drag it to any place to add it in the working area.



Method 2: Use a keyboard

Click the working area and enter the function block definition by the keyboard directly, the instruction edit dialog appears. After typing, press the **Enter** key on the keyboard or click **OK** in the instruction edit dialog. If you move the mouse cursor and click any position where the instruction is to be added in the working area, the function block will be inserted.



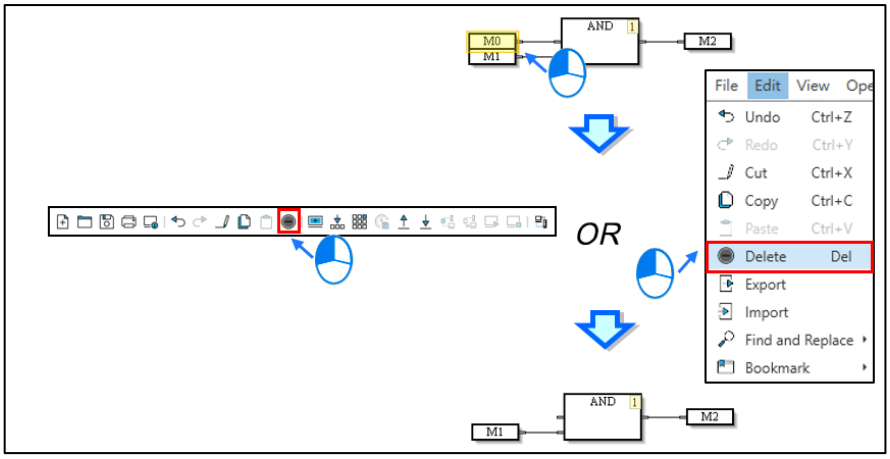
Note:

- The function block definition entered is not case-insensitive.
- When unsupported function block definition or the contents of non-compatible devices are entered, they will be regarded as variables. After clicking **OK**, the **Add a Variable to POU** dialog appears. If the content contains special characters, such as ABC??!!, this dialog does not appear.

Deleting objects

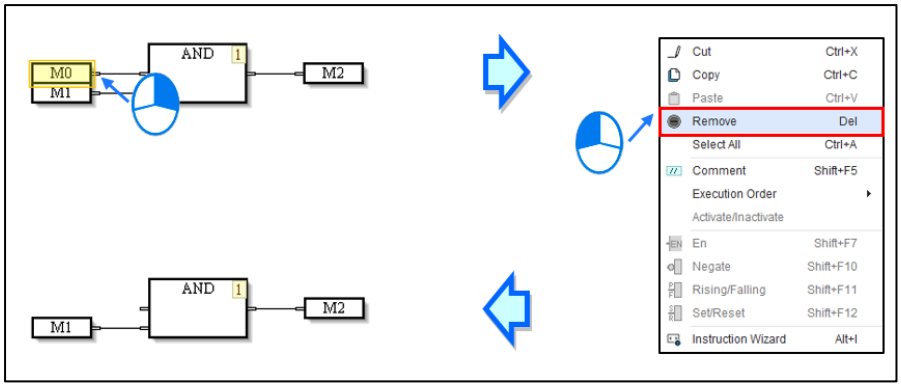
To delete a single object or several objects in a continuous function chart, use any of the following methods.

- Select an object, and click **Edit** > **Delete** in the menu bar, or click on toolbar.



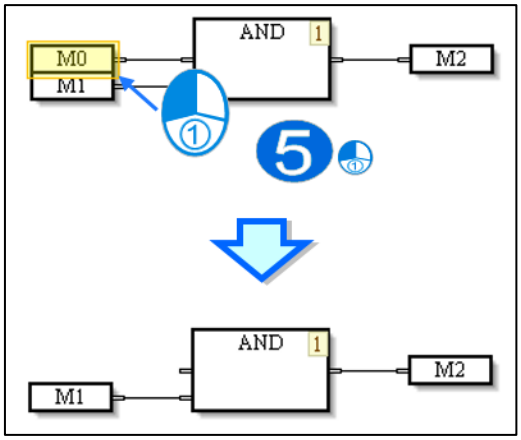
Or

Select and right-click an object, and then click **Remove** in the context menu.



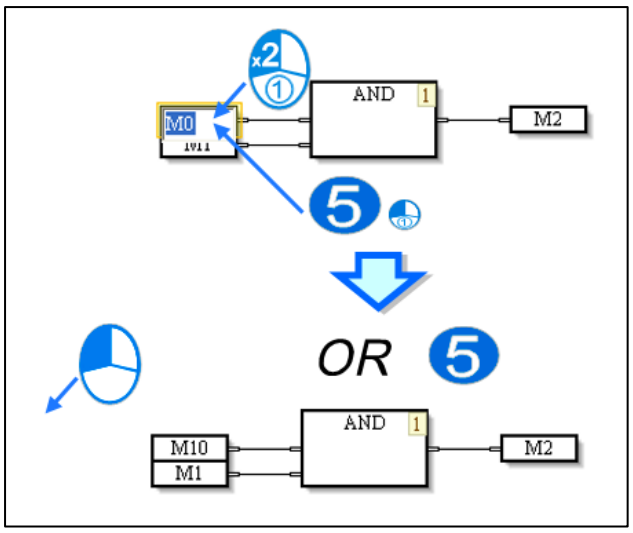
Or

Select an object and press **Delete** key on the keyboard.

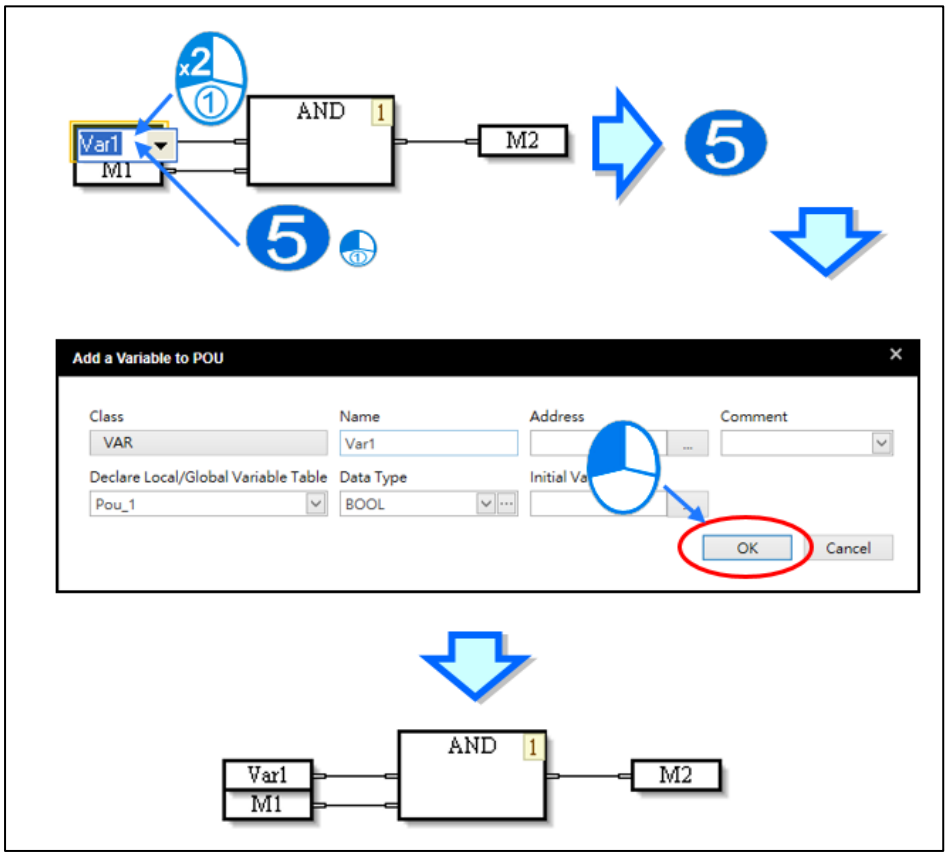


Editing Registers or Variables

To edit a node, double-click the node and enter a name, and then click the blank in the working area or press **Enter** key on the keyboard.



If you enter a variable in a node, the **Add a Variable to POU** dialog appears after pressing the **Enter** key on the keyboard. After setting a variable and click **OK** in the **Add a Variable to POU** dialog, the declaration of the variable is completed.

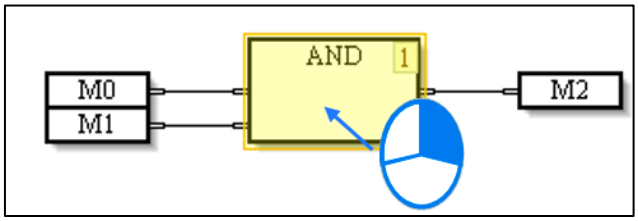


Activating / Inactivating an Object

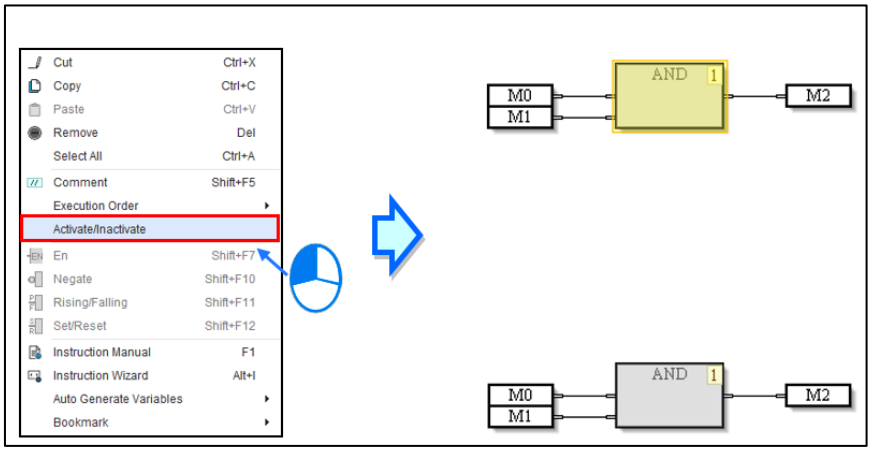
If an object in a continuous function chart inactivated, the compiling of the continuous function chart will skip the object. You can temporarily inactivate some parts of a program.

To activate / inactivate object

1. Right-click an object to be inactivated to open the context menu.



2. Click **Activate/Inactivate** in the context menu. The object becomes dark as shown in the following figure.



3. To activate an object, follow the steps above and click **Activate/Inactivate** in the context menu.

Inserting a Comment

Follow these steps to insert a comment in a continuous function chart. There are following two methods of inserting a comment.

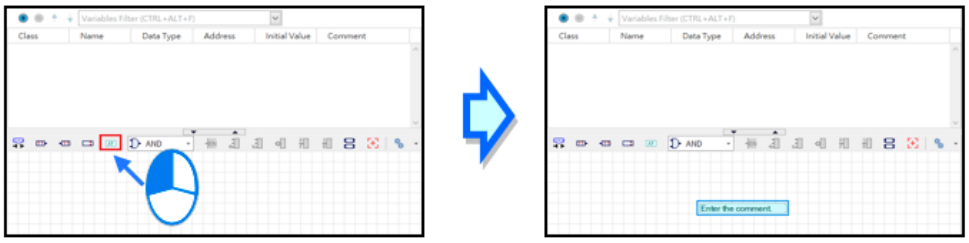
To insert comment

Method 1: Insert a comment in any position.

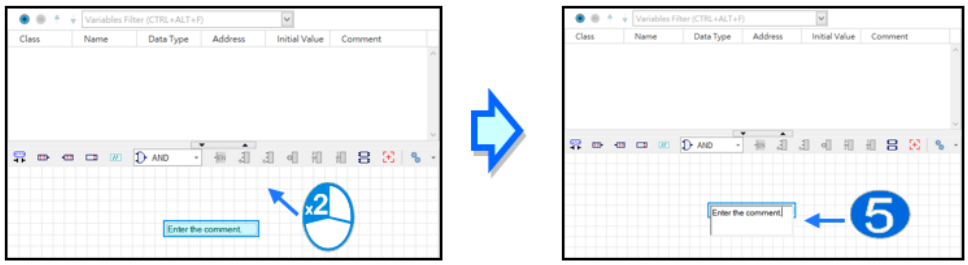
1. Click **Comment**  on toolbar.



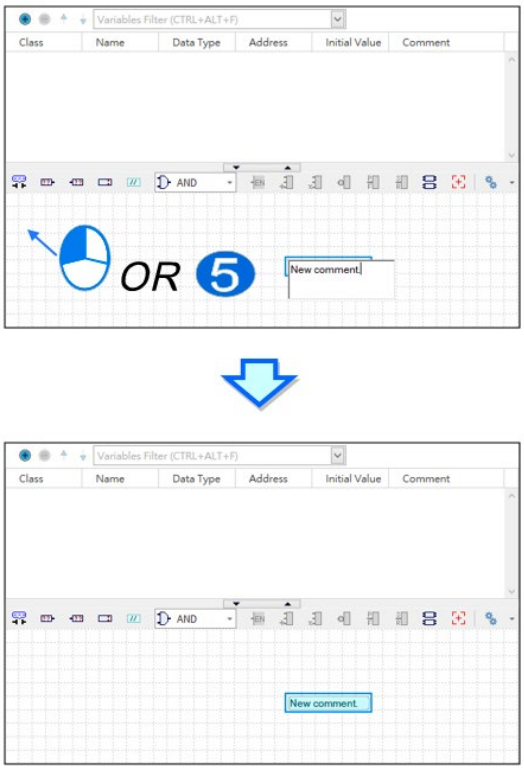
2. Click any position in the working area, a comment will be inserted, as shown in the following figure.



3. Double-click the inserted comment to open a dialog. Enter the content of the comment in this dialog. To start a new line of text at a specific point, press the **Shift** + **Enter** keys on the keyboard.

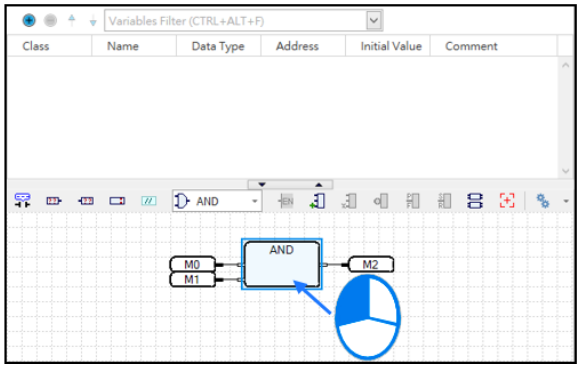


4. After you enter a comment, click the blank in the editing dialog, or press the **Enter** key on the keyboard. This type of comment is independent of the other types of objects in the working area. It does not vary with the deletion or the movement of another type of object.



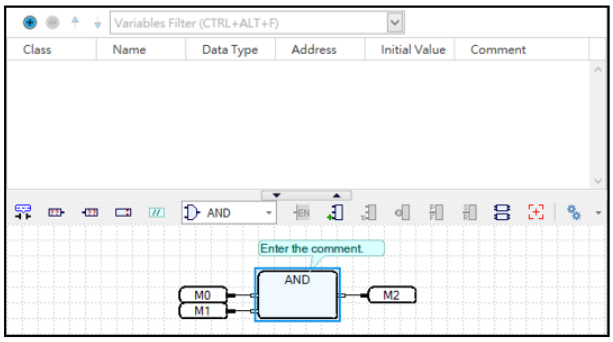
Method 2: Insert a comment binding with an object

1. Click the object in which a comment will be inserted, as shown in the following figure.

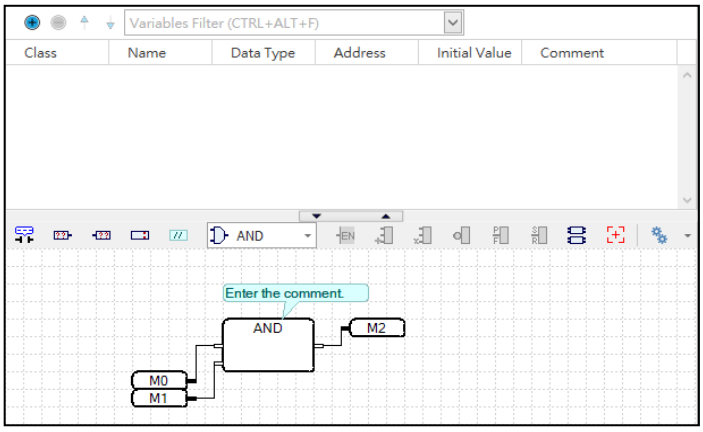


2. Click **Comment**  on toolbar. The comment binding with the object appears as shown in the following figure. You can edit the comment.



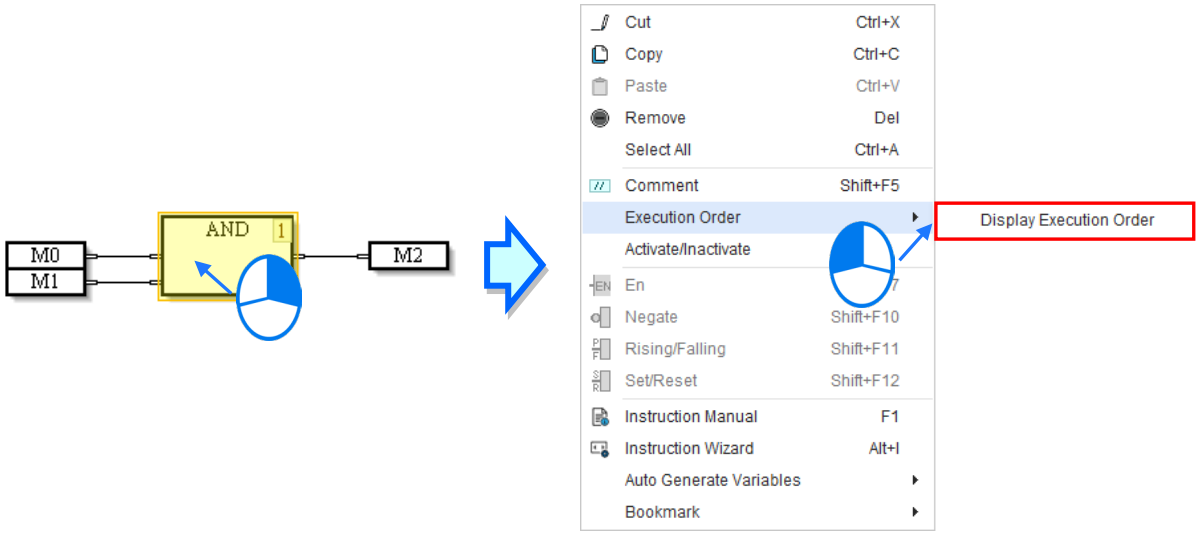


3. The position of the comment varies with the movement of the object. When you move the object, the comment moves.

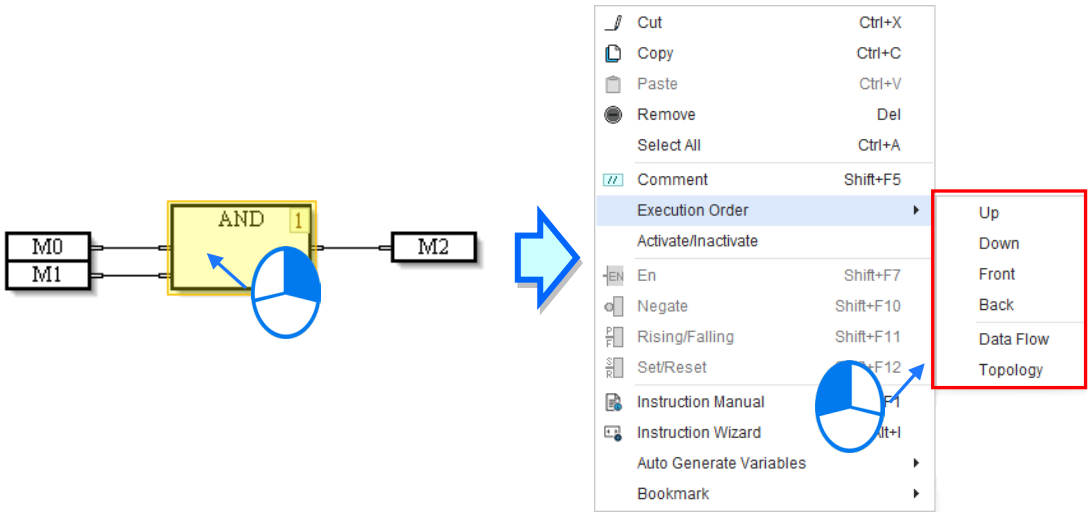


Changing the Order in which Objects are Executed

When the execution order is set to **Auto Data Flow Mode**, the execution will automatically sequence according to the topology position and data flow. The mode is the default setting, and you do not need to set the execution order.



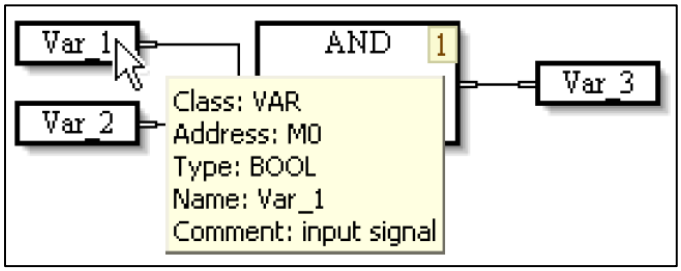
When the execution order is set to **Explicit Execution Order Mode**, the numbers in the upper right corners of the objects in a continuous function chart indicate the order in which the objects are executed. To change the number in the upper right corner of an object, right-click the object, and expand **Execution Order i**n the context menu to set the execution order of an object. If **Data Flow** is set, the order in which the objects are executed will be determined by the data flow of the objects. If **Topology** is set, the order in which the objects are executed will be determined by the relative positions of the objects.



Note: If the execution order is changed, the result would also be different. The **Auto Arrange**  on toolbar can be used by users who are not familiar with the order.

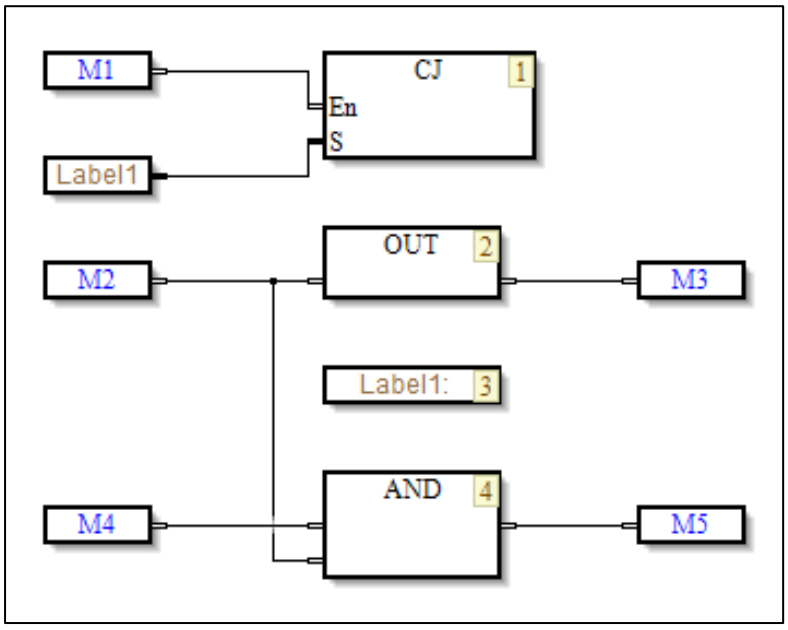
Displaying / Hiding Information of Registers

If you move the mouse cursor to a register or a variable after clicking **Show / Hide Register Comment**  on toolbar, the information about the register or the variable appears. The information includes a register address and a comment.



Setting a Label

When setting a label of skip instruction, click **Label**  on toolbar to add a label and enter the label name, as Label 1. Then enter the corresponding label name at pins CJ and JMP, setting the label for the target of the skip instruction. The order of labels is executed by referring to the execution order number in the upper right corners of the objects.



8.4 Compile and Check

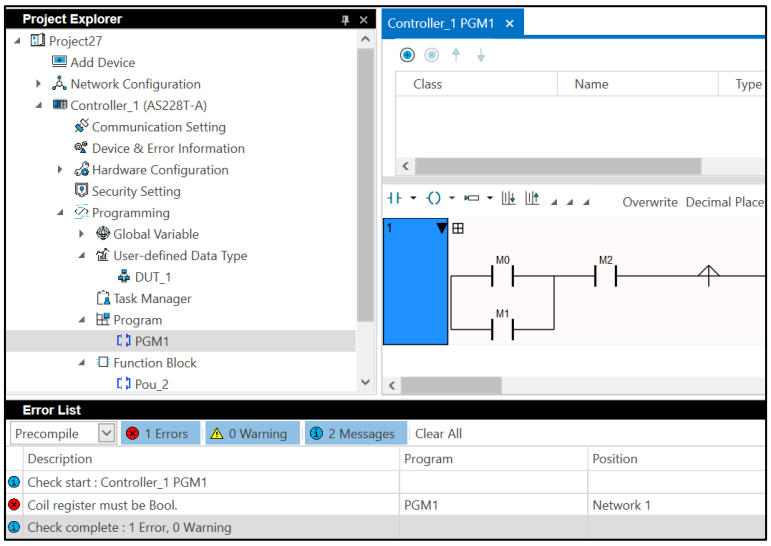
After editing the programs in IEC programming languages, check the syntax of the programming language with **Check** or compile the project or device with the help of **Compile**. The syntax and the structure in the present dialog are checked with **Check**. Execution code for project or device or program is generated only if **Compile** is successful, without any errors. It is possible to compile the entire project or individual device or individual program.

8.4.1 Check

Follow these steps to check a program POU.

To check program POU

1. Select the program POU to be checked.
2. Right-click program POU to open the context menu and click **Check** or Click **Operation** > **Check** in the menu bar. The **Precompile** starts the **Check** operation and **Error List** dialog displays the check results with errors, warnings and messages if any.



Note: If **Check** results contain errors or warnings, double-click the message in **Error List** dialog to navigate to the respective item.

8.4.2 Compile

Follow these steps to compile a project or a device or a program POU.

To compile device / program

1. Select the device or program POU to be compiled.
2. Click 

on **Development** toolbar.

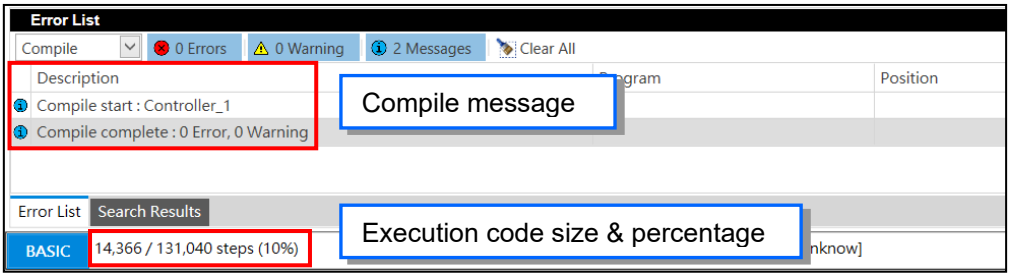
   Or

   Press **F11** key on the keyboard.

   Or

   Right-click the device or program POU to open context menu and click **Compile**.

   Compile completion and **Error List** dialog displays the compile message results with errors, warnings and messages if any. The percentage of the execution code size appears in the status bar also.



Note:

- If compile results contain errors or warnings, double-click the message in **Error List** dialog to navigate to the respective item.
- Compilation is performed based on the currently selected device. For example, after editing the program for controller A and clicking **Compile**, compilation will be performed on controller A. However, after editing the program for controller A, click controller B in **Project Explorer** and click **Compile**, then compilation will be performed on controller B.

Follow these steps to compile the project.

To compile project

- Click 

on **Development** toolbar.

  Or

- Press **Ctrl + F11** on the keyboard.

  The project compilation starts the compilation operation and displays the compilation result on the error list (any errors, warnings and messages).

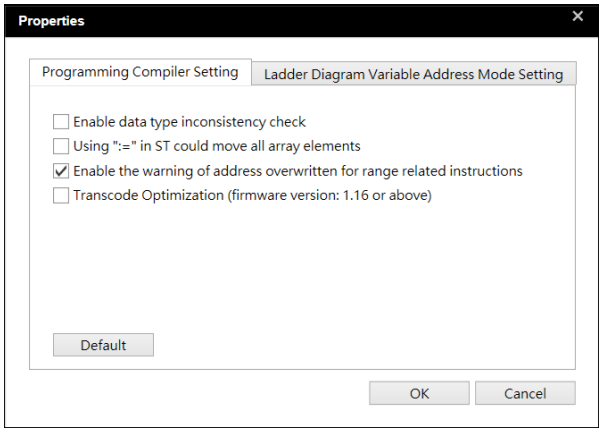
8.4.2.1 Programming Properties Setting

This setting includes the compiling rule or options during program editing.

How to open programming properties dialog?

- In **Project Explorer**, right-click **Device** > **Programming** and select **Properties**.

Programing Compiler Setting



- **Enable data type inconsistency check**

  When using MOV / DMOV instructions, DIADesigner defaults to be compatible with variables of different data types for migration. If you need a precise hint of data type difference when compiling, right-click **Programming** and select **Properties**. In the **Properties** dialog, select the **Enable data type inconsistency check** checkbox, and you will be notified with a warning during compilation.

The following table lists the warning for inconsistency of input / output variable types when using MOV / MOVP instructions.

Output / Input	WORD	INT (Counter Timer)	DWORD	DINT	REAL
WORD	Pass	Pass	The data length is inconsistent	The data length is inconsistent	The data length is inconsistent
INT (Counter Timer)	Pass	Pass	The data length is inconsistent	The data type is inconsistent	The data type is inconsistent
DWORD	The data length is inconsistent	The data length is inconsistent	Pass	Pass	Pass
DINT	The data length is inconsistent	The data type is inconsistent	Pass	Pass	The data type is inconsistent
REAL	The data length is inconsistent	The data type is inconsistent	Pass	The data type is inconsistent	Pass

The following table lists the warnings for inconsistency of input / output variable types when using DMOV / DMOVP instructions.

Output / Input	DWORD	DINT	REAL
DWORD	Pass	Pass	Pass
DINT	Pass	Pass	The data type is inconsistent
REAL	Pass	The data type is inconsistent	Pass

- **Using ":=" in ST could move all array elements**

  If you want to move array elements in ST POU, for example, to write the program as ArrayVar1:=ArrayVar2; select this option. The controller will move all

elements in array variables during running. If not selected, only the first element will be moved.

- **Enable the warning of address overwritten for range related instructions**

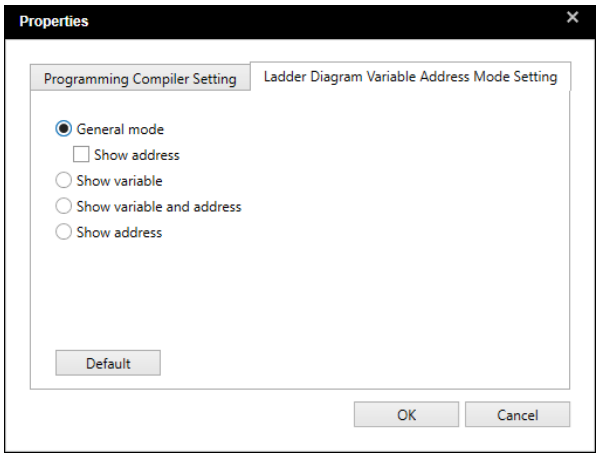
  When using BMOV instruction, if the input pin is set to a register address, the same address may be written repeatedly. Selecting this option will prompt you by an alarm message.

- **Transcode Optimization**

  If selected, the software and firmware will adopt the new compilation method to reduce the program size after compiling and increase the execution efficiency.

  Note: This function needs to be used with DIADesigner version 1.9.0 or above, AS series firmware version 1.16 or above, and DVP series firmware version 1.10 or above.

Ladder Diagram Variable Address Mode Setting



Set the program content to be displayed in the ladder diagram using the variable name or register address.

Option	Description
General mode	If a variable has a specified address, the screen only displays the method entered by users. For example, when a variable is entered, only the variable is displayed; when an address is entered, only the address is displayed. Select the **Show address** checkbox to display the address specified by the variable.
Show variable	If a variable has a specified address, the corresponding variable is displayed no matter entering the variable or

Option	Description
	address. Address not occupied by variable is displayed as address.
Show variable and address	If a variable has a specified address, the variable and address are displayed no matter entering the variable or address. Variable with unspecified address is displayed as variable; and address not occupied by variable is displayed as address.
Show address	If a variable has a specified address, the variable and address are displayed as the corresponding address no matter entering the variable or address. Variable with unspecified address is displayed as variable.

8.4.3 Export Programs to Text

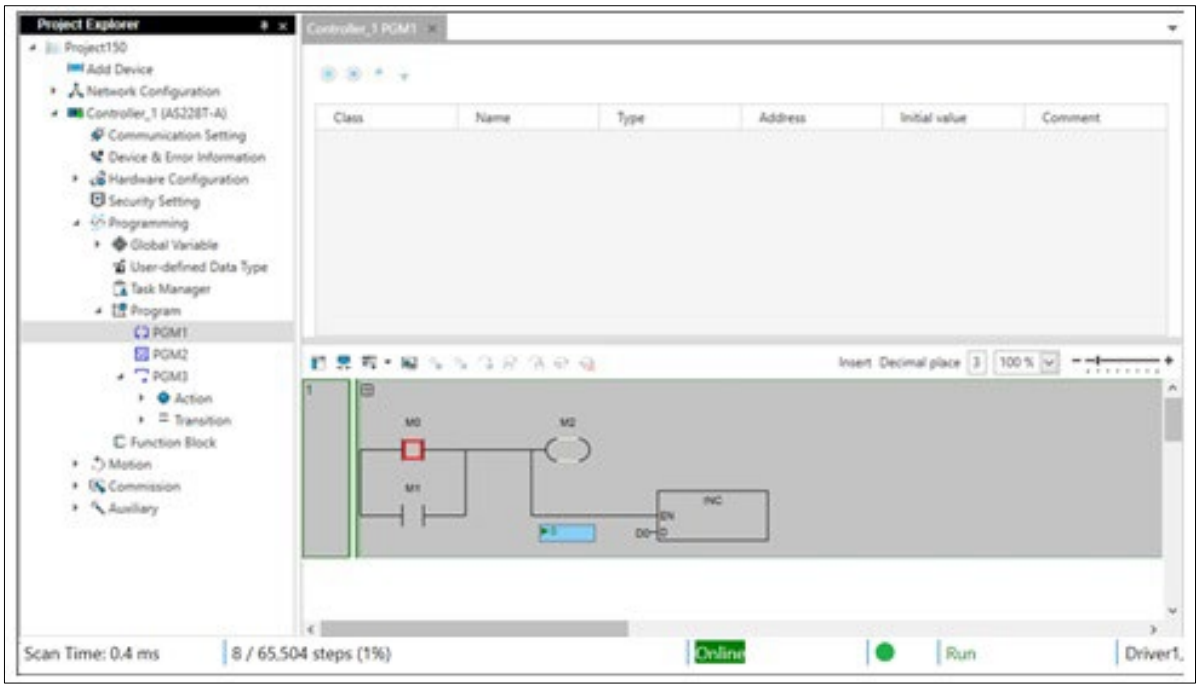
The code compiled by the program can be converted into a text file through this function for document archiving. Follow these steps to export the programs to text format.

To export programs to text

1. In **Project Explorer**, right-click **Programming** and select **Export Programs to Text** in the context menu.
2. In the **Save As Project** dialog, select the desired location to save and modify the file name as required, and then click **Save**.

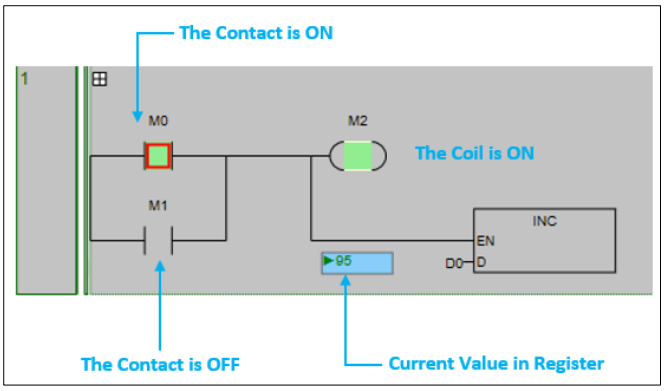
8.5 Program Monitoring

In the online program monitoring, you can view the present scan time and the status of the PLC in the status bar. You can also set the On and Off status of each element in the program using the mouse or switch between the statuses using the + and – keys on the keyboard, and enter the setting value to monitor the output changes of the program.



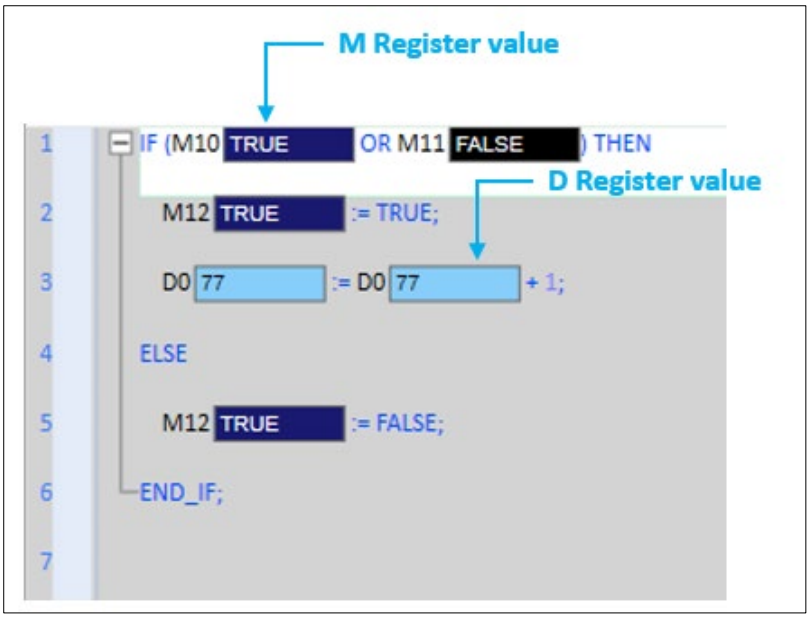
8.5.1 LD Program Monitoring

In Online program monitoring, LD POU displays the status or values of Contacts, Coils, APIs, Address, Variables etc. The following figure shows a Ladder Diagram POU in the program monitoring.



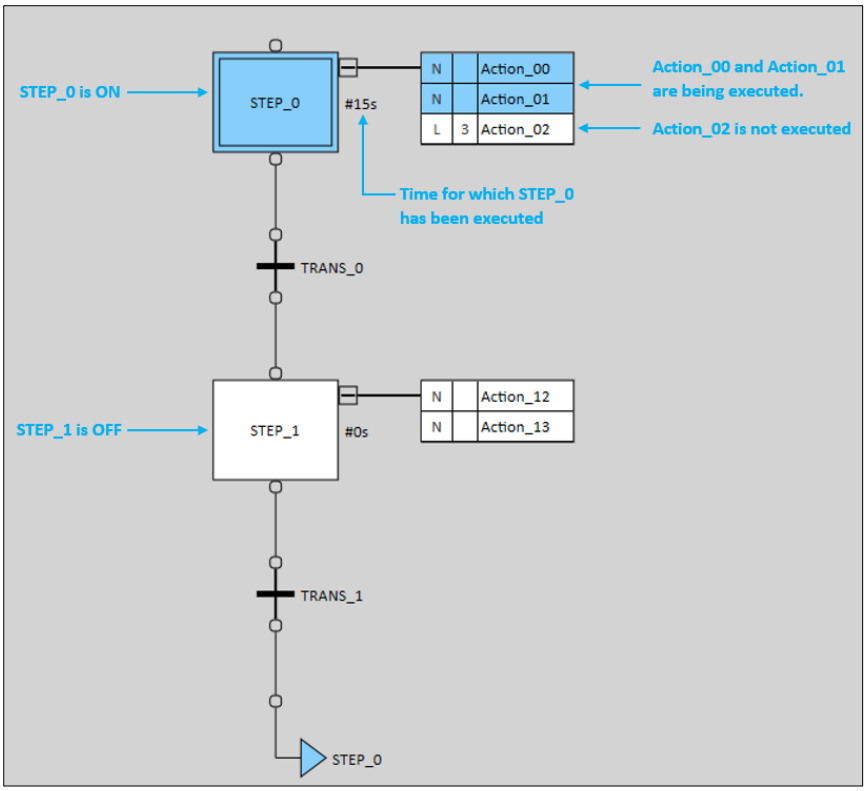
8.5.2 ST Program Monitoring

In online program monitoring, ST POU displays the status of Registers or Values. Boolean Address or Variables display with the TRUE or FALSE status and other registers with the actual values. The following figure shows a Structured Text POU in the program monitoring.



8.5.3 SFC Program Monitoring

In Online program monitoring, not only the SFC, but the Actions and Transitions are also monitored. The following figure shows a Sequential Function Chart POU in the program monitoring.

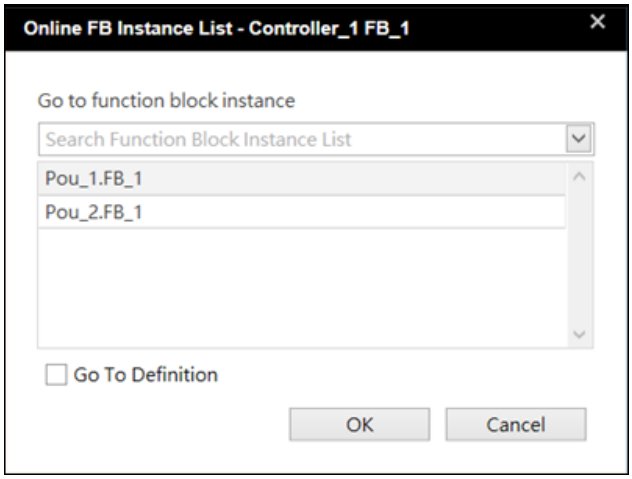


8.5.4 Function Block Monitoring

You can monitor the function block status referenced in multiple POU through the online FB instance list function.

To open Online FB Instance List in online mode

- In **Project Explorer**, double-click the function block node.



You can also directly double-click the function block object in the POU. The monitoring page of the function block instance displays.

8.6 Online Edit

When the system is in online mode, and the PLC is running, you can modify the program monitored through the function Online Edit. Refer to the following table for more information about the limits set on the editing of programs in **Online Edit** mode.

Programming Language	Restriction in Online Edit mode
Ladder Diagram	• Only one Network can be edited at a time. • You are allowed to add a Network. • After a Network is added, you are allowed to edit only the newly added Network, previous Networks cannot be edited.
Structured Text	There is no limit on the number of lines which can be edited.

Programming Language	Restriction in Online Edit mode
Sequential Function Chart	• A Sequential Function Chart cannot be edited in online. • Only **Actions** and **Transitions** can be modified in online. • The limit set on the editing of an **Action** or **Transition** depends upon the language used to create the Action or Transition.
C Language	Online editing is not supported.
Common	• DIADesigner must be in **Online** mode with the Controller. • Controller must be in **RUN** mode. • Edit only one POU at a time. • You cannot modify or add / delete **Variables**.

During **Online Edit**, you have the option to **Download** (Only changes) in:

- Object Code – Compiled code by PC.
- Object Code + Program and Source Code – created by users.

8.6.1 POU Online Edit

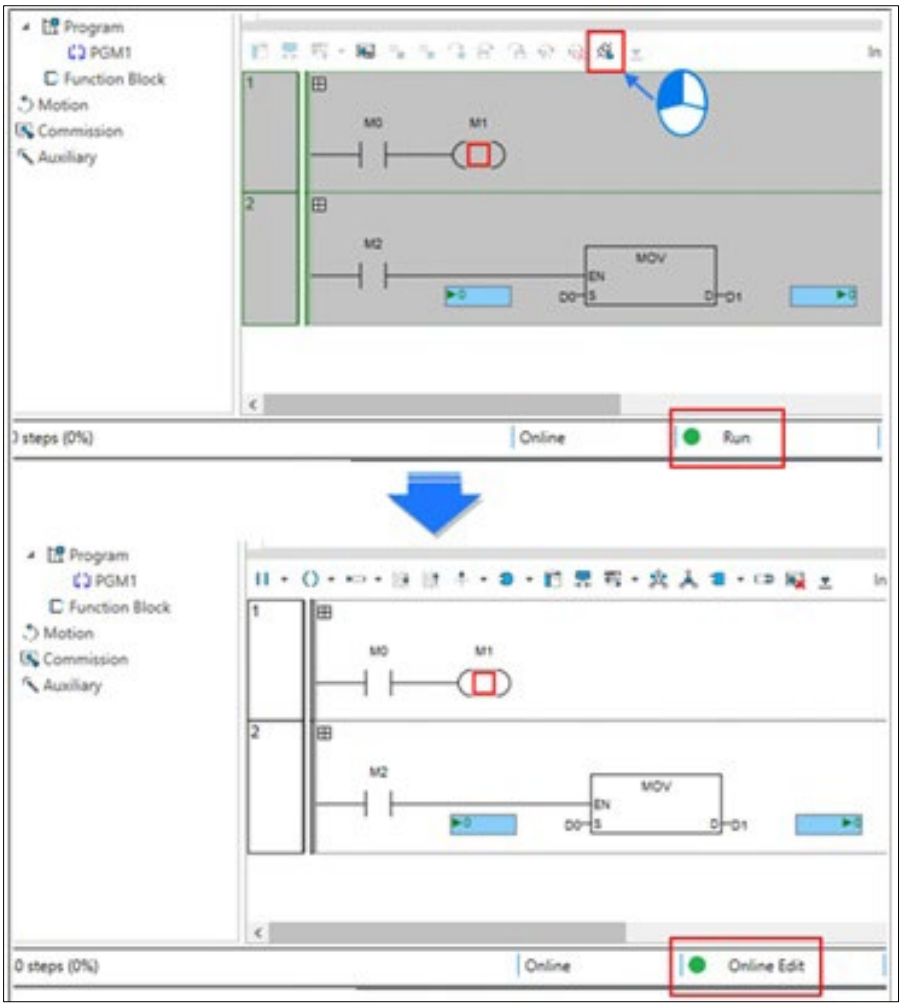
8.6.1.1 Online Edit in Ladder Diagram (LD) POU

Follow these steps to perform **Online Edit** in a Ladder Diagram (LD) POU.

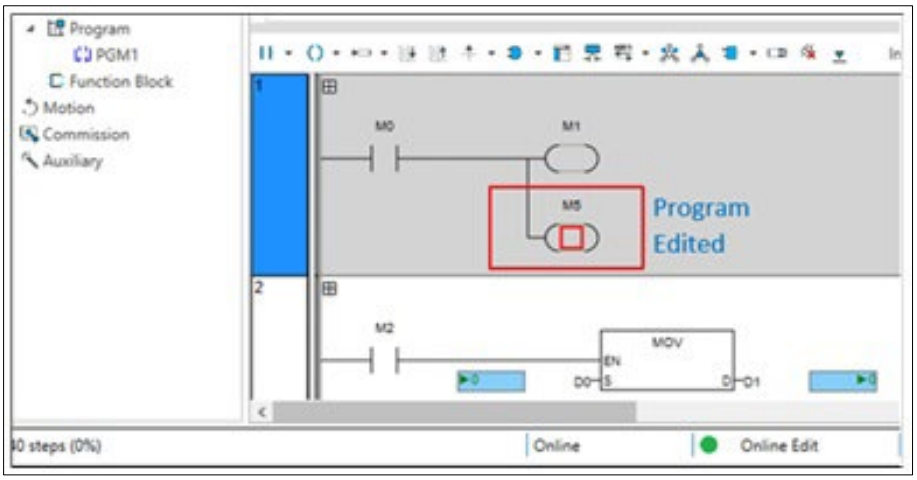
To perform Online Edit in a Ladder Diagram POU

1. Open the Ladder Diagram POU and click  on LD POU toolbar.

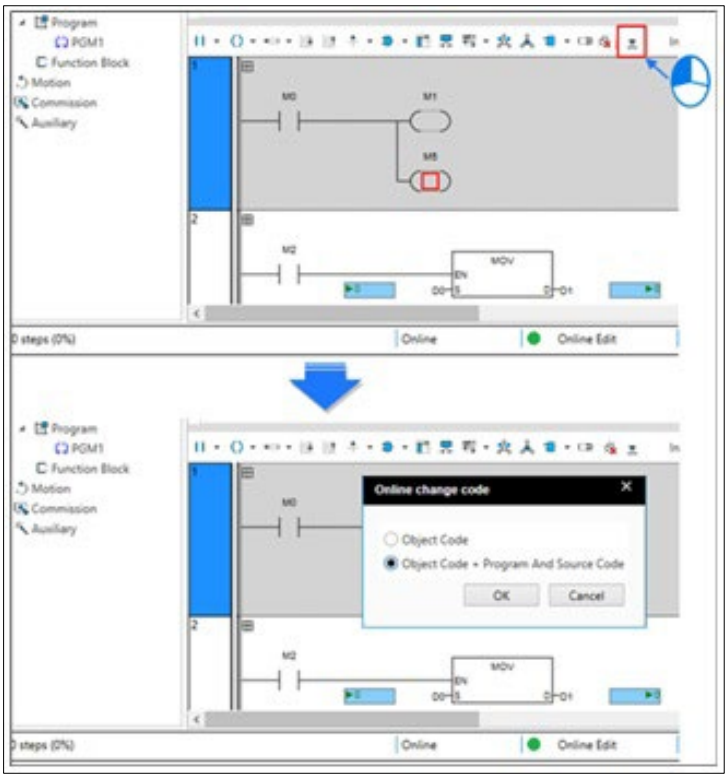
   DIADesigner enters **Online Edit** mode. Status bar displays **Online Edit** in mode area as shown in the following figure.



2. Edit the LD POU.

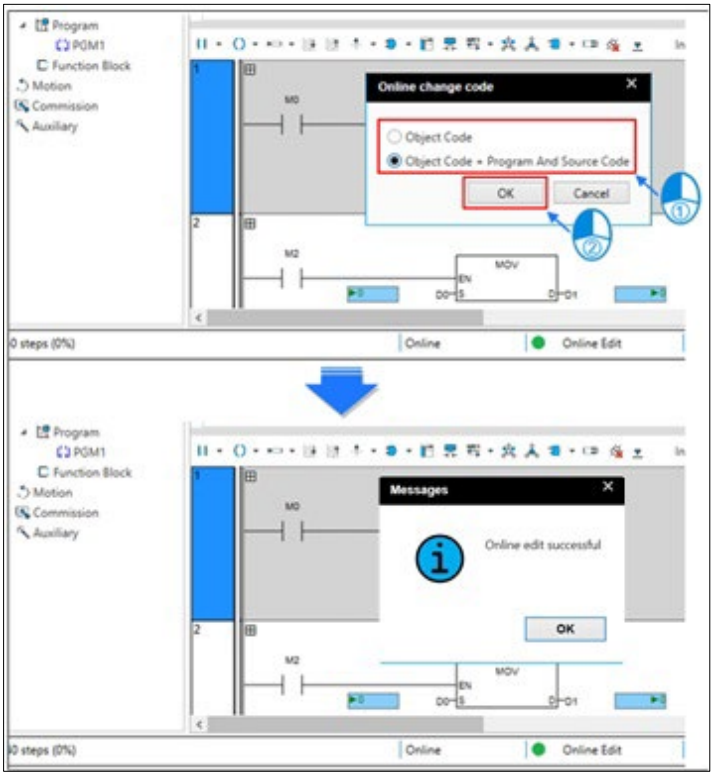


3. After the editing is complete, click  on LD POU toolbar to open Online Change Code dialog.

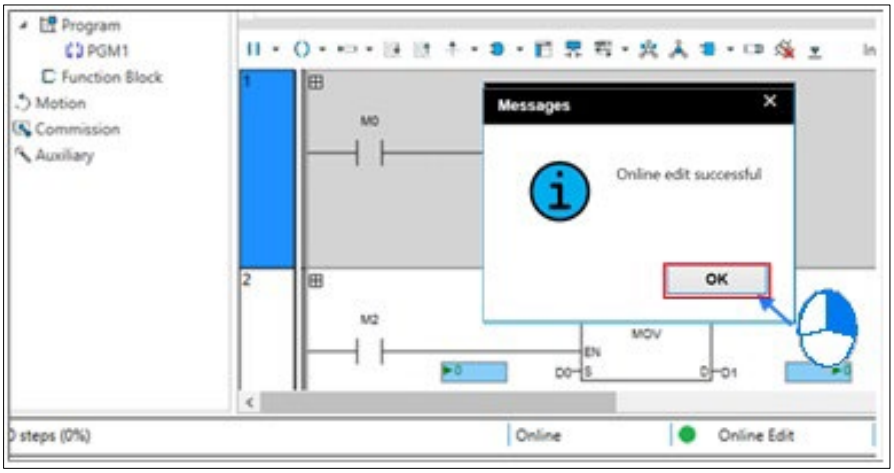


4. Select **Object Code** or **Object Code & Program and Source Code** option as needed and click **OK** in the **Online Change Code** dialog.

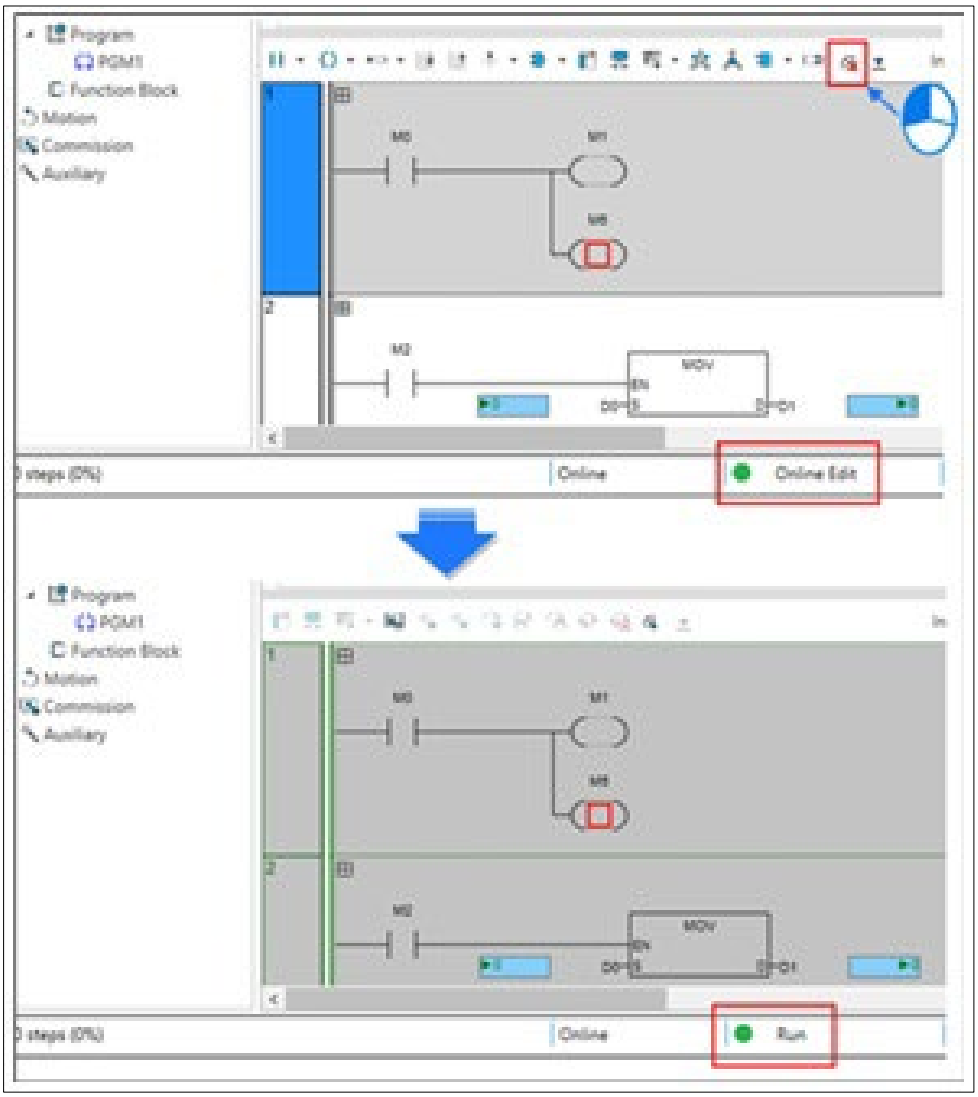
   The pop-up dialog – **Online edit successful** appears as shown in the following figure.



5. Click **OK** in the pop-up dialog.



6. Click  on LD POU toolbar to stop **Online Edit**. DIADesigner returns to **RUN** mode.



8.6.1.2 Online Edit in Structured Text (ST) POU

Follow these steps to perform **Online Edit** in a Structured Text (ST) POU.

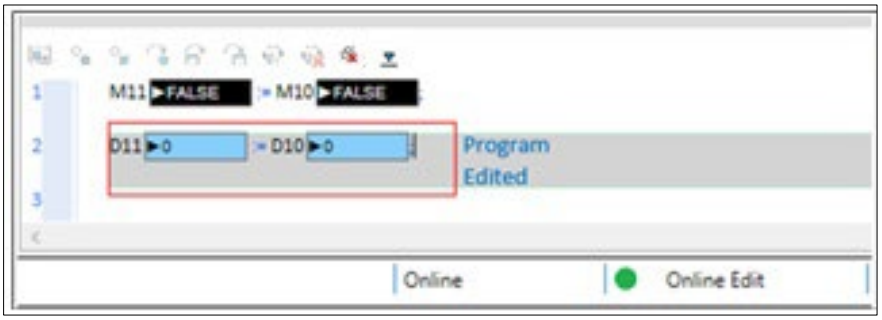
To perform Online Edit in a structured text POU

1. Open the Structured Text POU, and click  on ST POU toolbar.

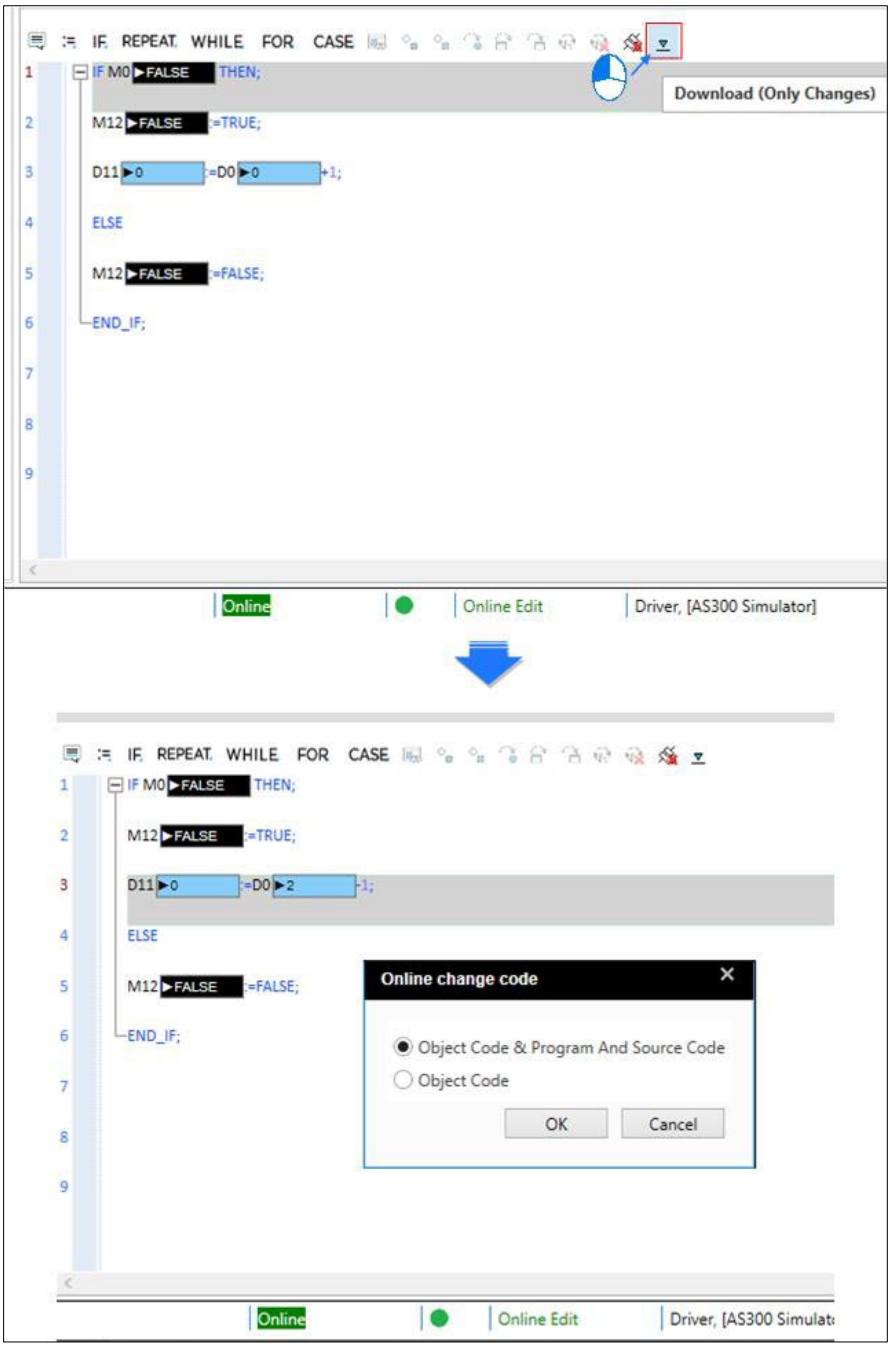
   DIADesigner enters **Online Edit** mode. Status bar displays **Online Edit** in mode area as shown in the following figure.



2. Edit the ST POU.

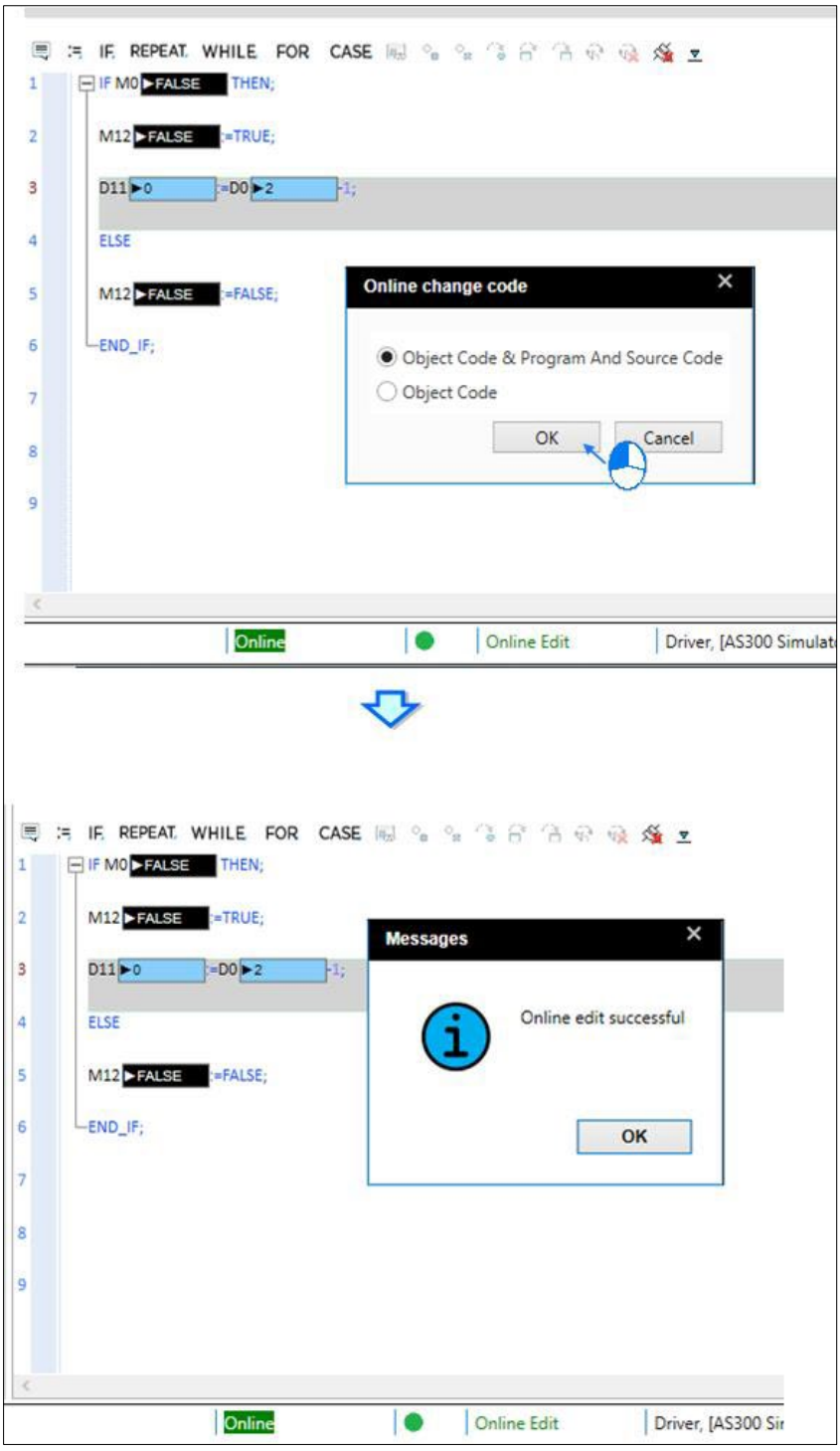


3. After the editing is complete, click  on ST POU toolbar to open **Online change code** dialog.

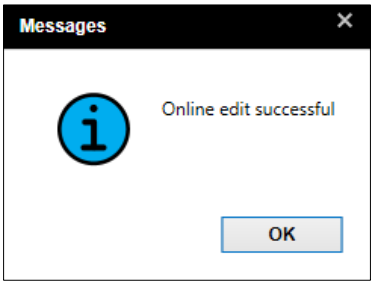


4. Select **Object Code** or **Object Code & Program and Source Code** option as needed and click **OK** in the **Online change code** dialog.

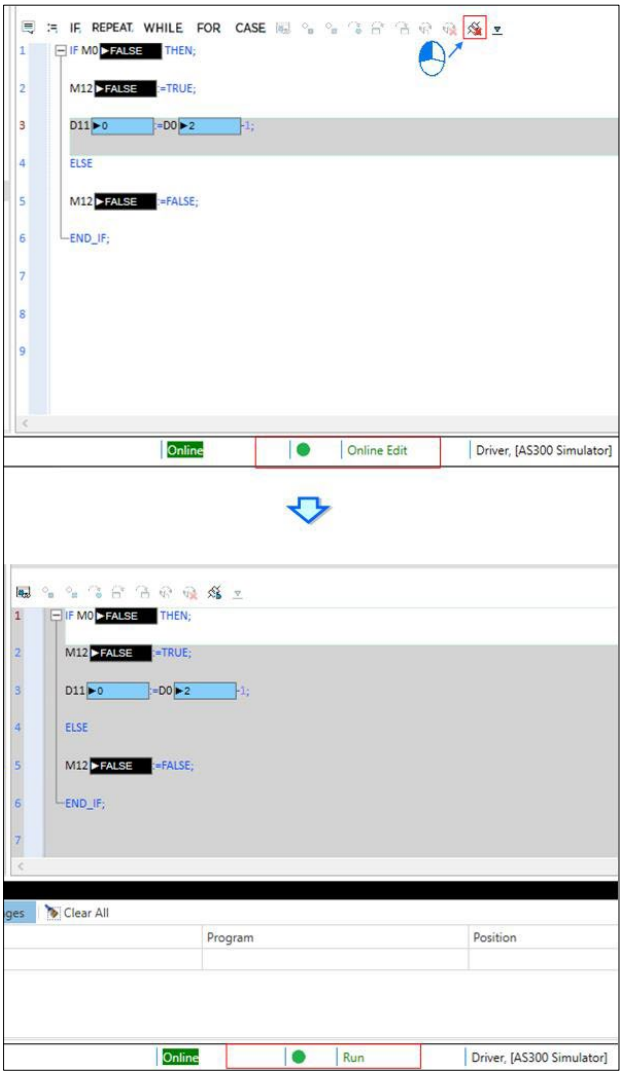
   The **Message** dialog displays **Online edit successful**.


IF REPEAT WHILE FOR CASE
IF M0 FALSE THEN;
M12 FALSE :=TRUE;
D11 0 :=D0 2 -1;
ELSE
M12 FALSE :=FALSE;
END_IF;
Online change code
Object Code & Program And Source Code
Object Code
OK
Cancel
Online
Online Edit
Driver, [AS300 Simulat
IF REPEAT WHILE FOR CASE
IF M0 FALSE THEN;
M12 FALSE :=TRUE;
D11 0 :=D0 2 -1;
ELSE
M12 FALSE :=FALSE;
END_IF;
Messages
Online edit successful
OK
Online
Online Edit
Driver, [AS300 Sir

5. Click **OK**.


Messages
Online edit successful
OK

6. Click  on ST POU toolbar to stop **Online Edit**. DIADesigner returns to **RUN** mode.



8.6.1.3 Online Edit in Sequential Function Chart (SFC) POU

SFC POU cannot participate in an **Online Edit**. But the Actions and Transitions POUs which are made with LD or ST can take part in Online Edit. Procedure is same as that mentioned in the previous sections.

8.6.1.4 Online Edit – Stop and Recover

During Online Edit, after finishing editing the code, choose **Stop Online Edit**. This results in two scenarios.

- Stop Online Edit and Recover – Yes

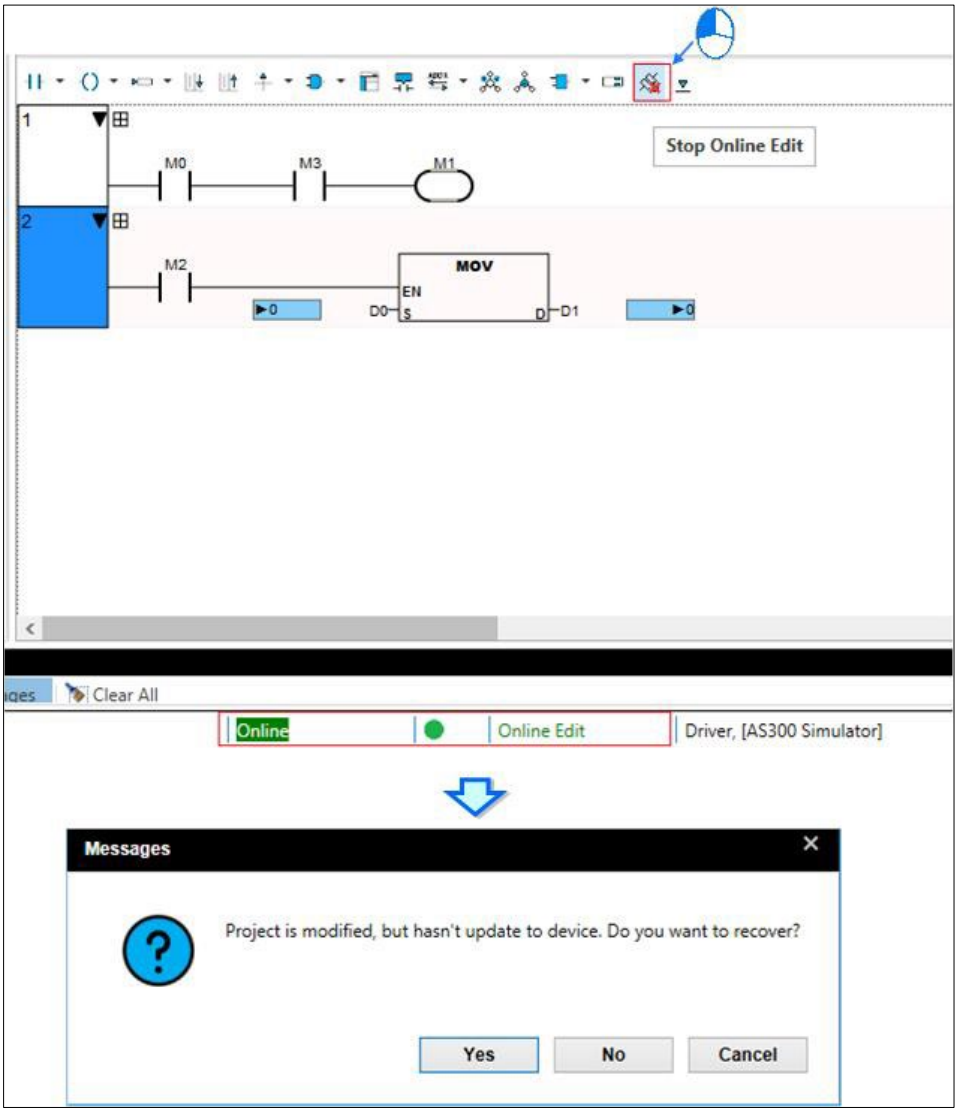
- Stop Online Edit and Recover – No

In either way, DIADesigner goes in Offline mode with the controller. After that, compile the program and download the compiled program to the controller later, which is further explained in the following sections.

8.6.1.5 Stop Online Edit and Recover – Yes

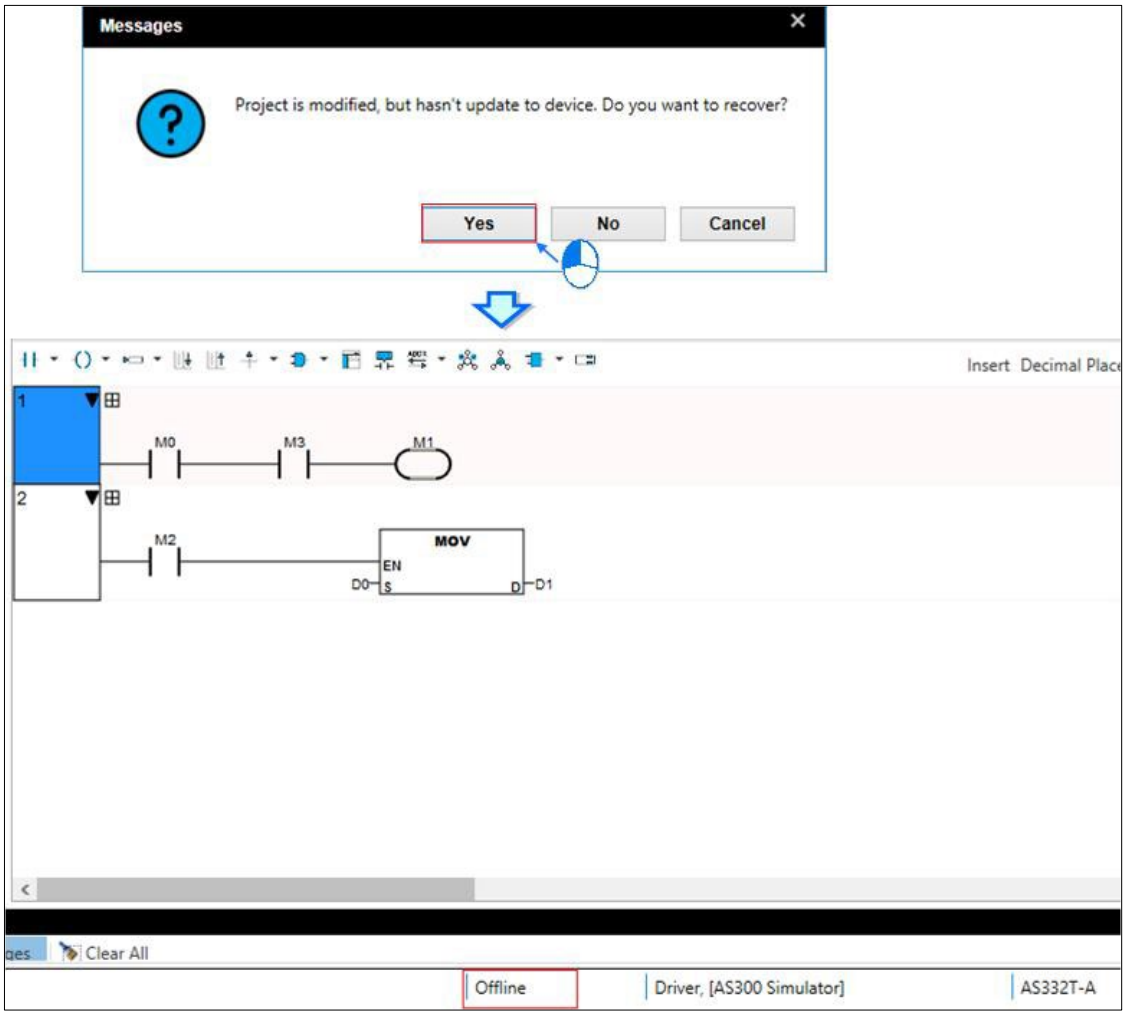
1. During **Online Edit**, after finishing the edit, if you feel to cancel operation, click .

   The **Messages** dialog – **Project is modified, but hasn't update to device**. **Do you want to recover?** appears as shown in the following figure.



2. Click **Yes**.

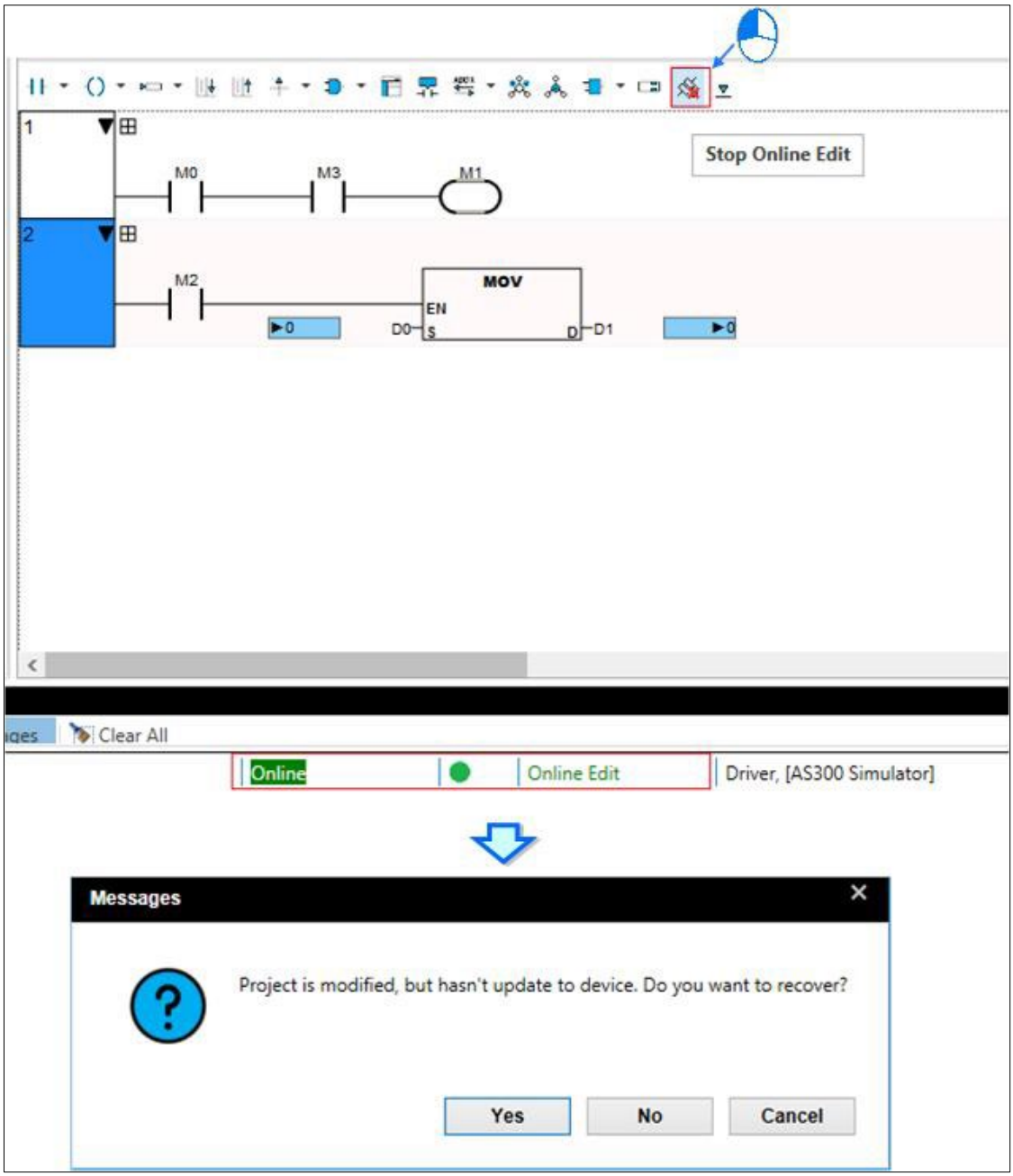
Program edit is ignored, that is - program code before the Online Edit mode is restored and the status becomes **Offline** (Recover) as shown in the following figure.



8.6.1.6 Stop Online Edit and Recover – No

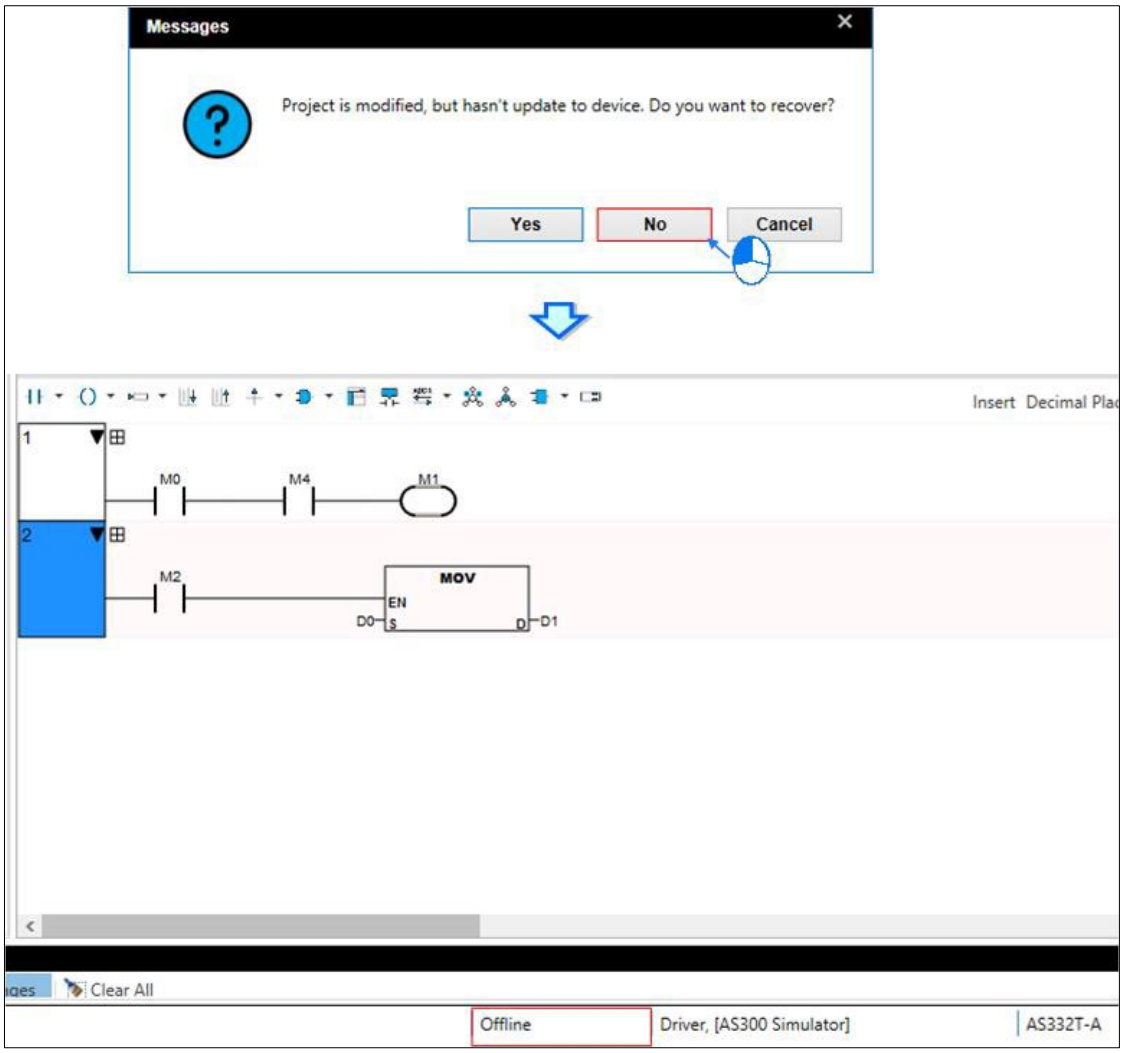
1. During **Online Edit**, after finishing the edit, if you feel to cancel operation, click .

   The **Messages** dialog appears as shown in the following figure.



2. Click **No**.

   Program edited remains intact and the status becomes **Offline** (**No Recover**) as shown in the following figure.



8.7 Debug Mode

DIADesigner provides debugging functions for controllers to test programs by means of **Debug Mode**. In DIADesigner, the debug mode is supported for all CPU modules for all the programming languages. However, debug mode for an SFC can only be executed for the Actions which are ON. Besides, if the state of a Transition in an SFC does not make the transition from one step to another step, you must change the state of the transition.

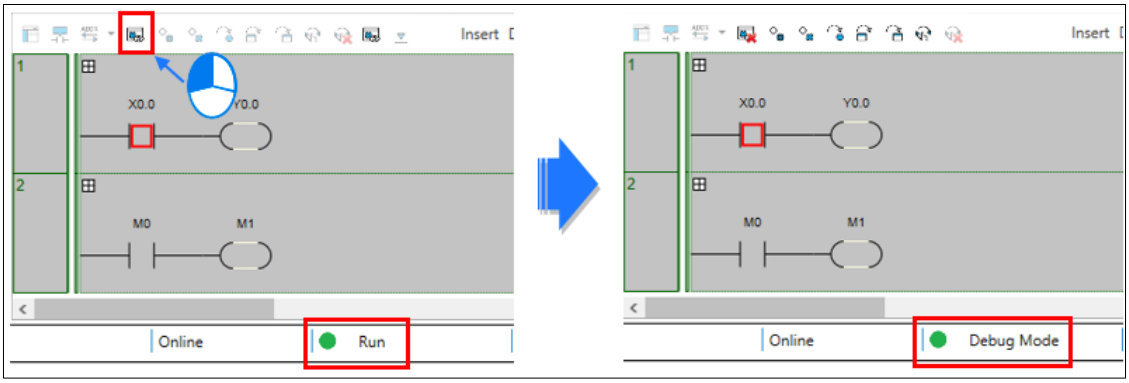
8.7.1 Enable Debug Mode

Debug Mode can be enabled only when the CPU is in **Online** mode. When **Debug Mode** is enabled, the status of the device in DIADesigner displays **Debug Mode**.

Follow these steps to enable **Debug Mode**.

To enable Debug Mode

1. Click  on POU toolbar. Controller enters **Debug Mode** and status bar displays **Debug Mode** in the status area as shown in the following figure.



Note: After clicking **Debug Mode** icon, if CPU does not enter debug mode, that means another user on the network is debugging the program in the CPU module.

8.7.2 Break Point

A **Break Point** is an intentional stop marked in the application code at positions to check the behavior at those stop points. In **Debug Mode**, you are allowed to set the **Break Point** in an application. Maximum of 10 Break Points can be added to an AS series PLC.

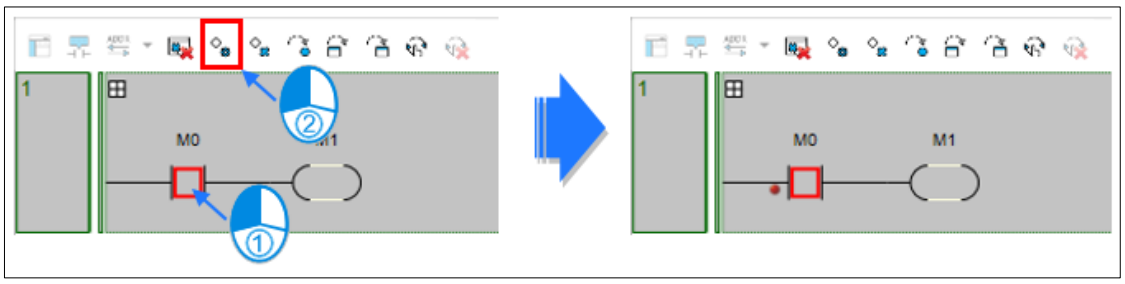
8.7.2.1 Insert a Break Point

Follow these steps to insert a **Break Point.**

To insert Break Point

1. Open the POU and click  on POU toolbar. DIADesigner enters **Debug Mode**.
2. Select the position where you intend to add a **Break Point**.

3. Click  on POU toolbar to insert **Break Point** as shown in the following figure.



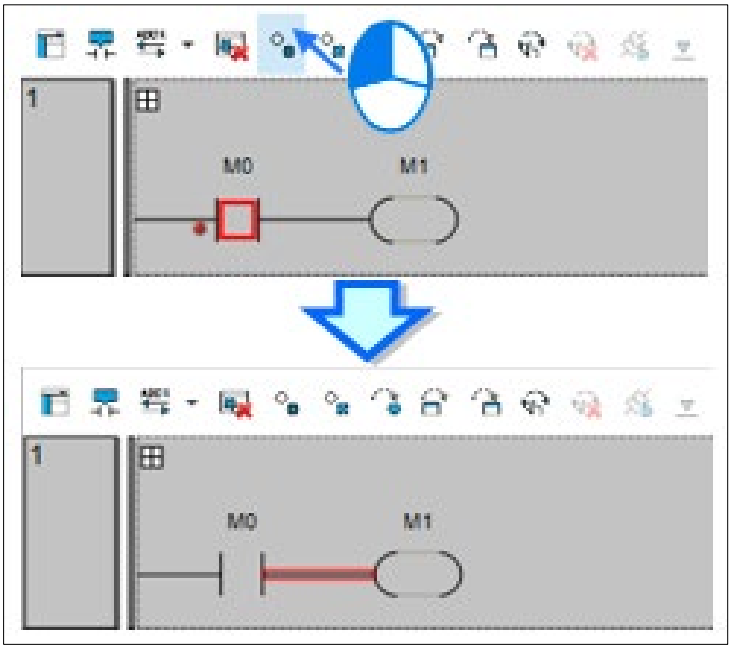
8.7.2.2 Delete a Break Point

Follow these steps to delete the **Break Point**.

To delete Break Point

1. Select the position where **Break Point** has to be deleted

2. Click  on POU toolbar to delete **Break Point**.

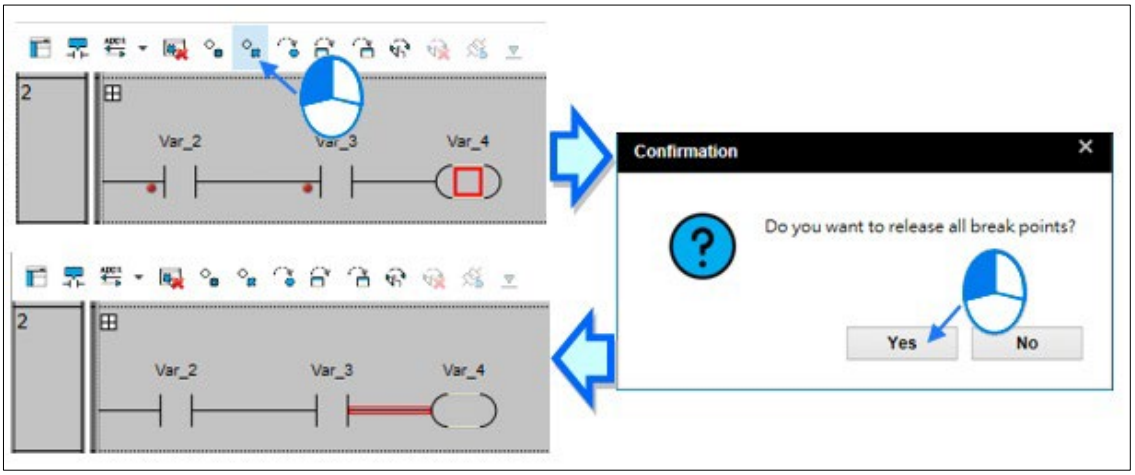


8.7.2.3 Delete all Break Points

Follow these steps to delete all Break Points.

To delete all Break Points

1. Click  on POU toolbar.
2. A message appears – **Do you want to release all break points?**
3. Click **Yes**.

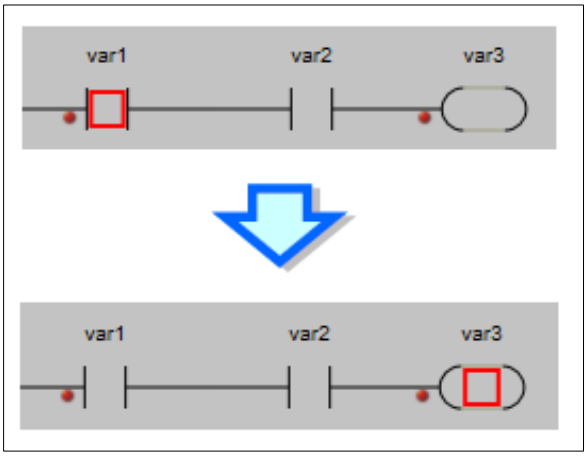


8.7.2.4 Debug the Application Code

Follow these steps to debug the application code after inserting Break Points.

To debug application code

1. Click  on POU toolbar to stop the execution of the program at a Break Point, as shown in the following figure.



2. Click  again on POU toolbar. The execution of the program continues, and will stop at the next Break Point.

   Note: The position where the execution of the program stops is the part of the program which has not been executed.

8.7.3 Continue Run Execution

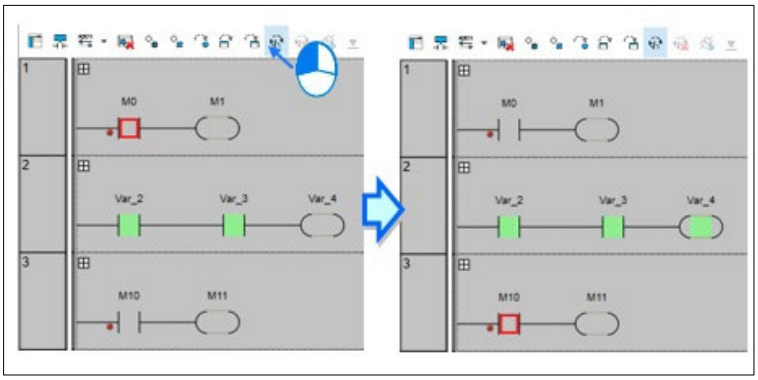
Continue Run Execution function helps to execute the portion of the application code between one Break Point and the next Break Point.

Follow these steps to perform Continuous Run Execution.

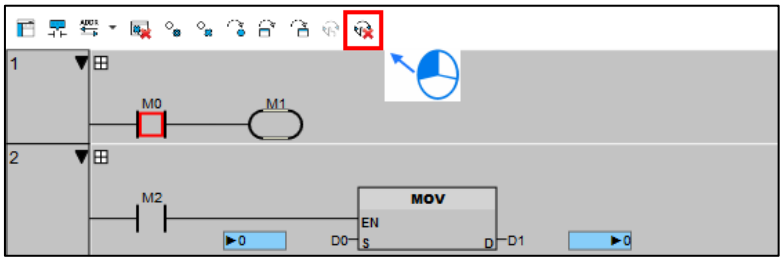
To perform Continuous Run Execution

1. Click  on POU toolbar to stop the execution of the program at a Break Point.

2. Click  on POU toolbar again. The execution of the program continues and will stop at the next Break Point as shown in the following figure.



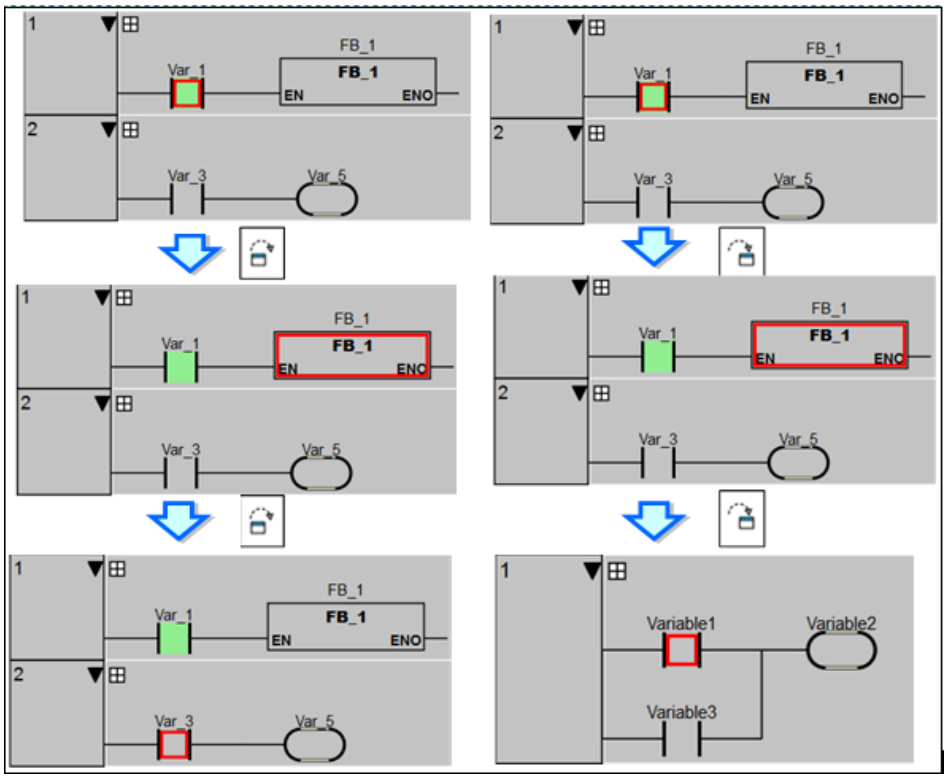
   Note: If there are no more Break Points, the program is scanned repeatedly after  is clicked.



8.7.4 Single-Step Execution

After clicking  or  on the toolbar, the part of the program selected is executed. If a function block is met, the function block is executed after  is clicked, and the first step in the function block is executed after  is clicked. However, if the function block selected is protected with a password, or the state of the EN pin of the function block selected is OFF, the first step in the function block will not be executed whether  or  is clicked. Besides, to perform single-step execution in a structured text, add breakpoints to the structured text. To perform single-step execution in the POUs following a sequential function chart, add breakpoints to the POUs. The difference between  and  is shown in the following figure.

Note: After the controller is running, first press **Start Debug Mode**  to perform the following actions.



Chapter 9: Library Management Tools

9.1 Library Project

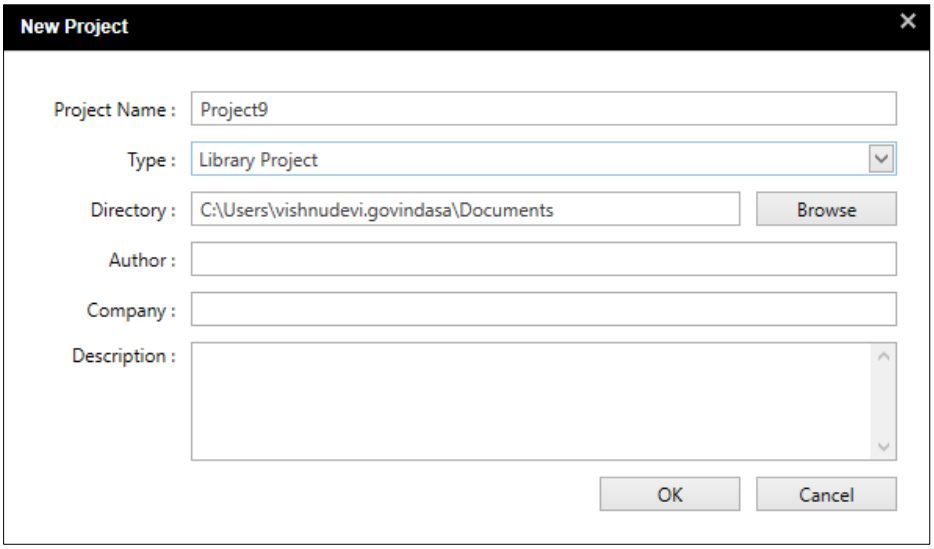
The use of library projects can reduce the time spent to develop repetitive functions in the project development process. You only need to install the built library project to modify the content and include it in version control for subsequent maintenance. The above can greatly reduce the cost of design, development and implementation, testing and document writing.

9.1.1 Create Library Project

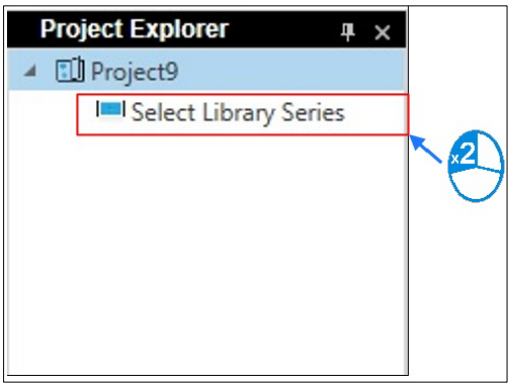
Follow these steps to create a project.

To create project

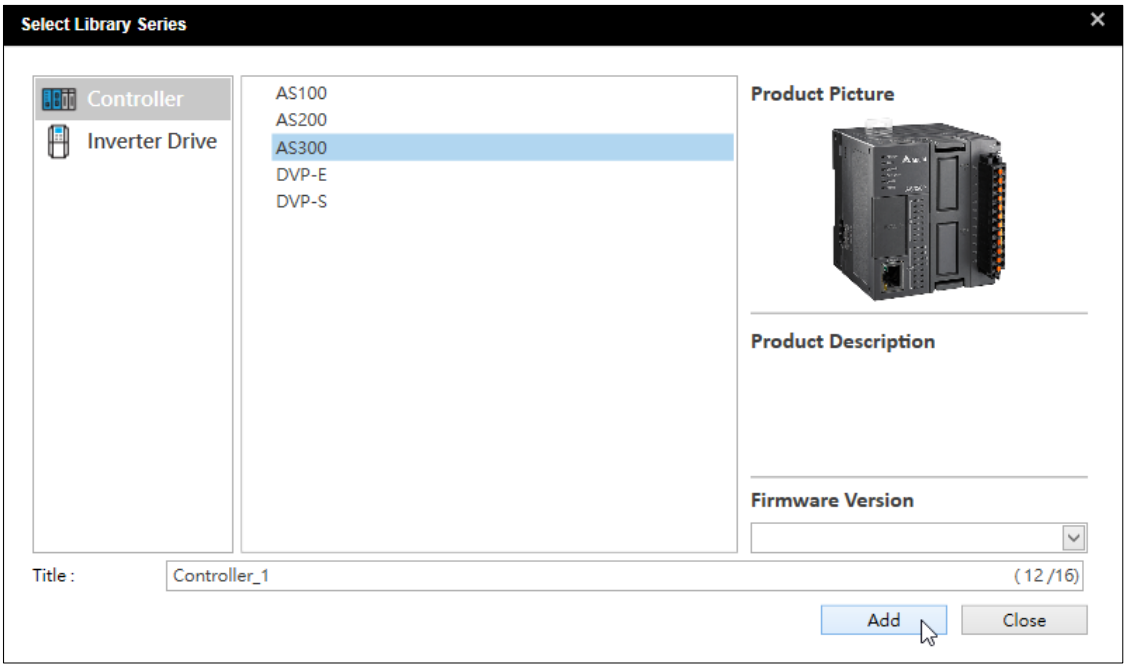
1. Create a new project with **Library Project** option in the **Type** drop-down.



2. After the library project is created, double-click **Select Library Series** in **Project Explorer** as the corresponding model of the library.



3. For example, consider AS300, click **OK**; after selecting, the library project only supports the specified AS300 series models, and the library project cannot be installed on other models.

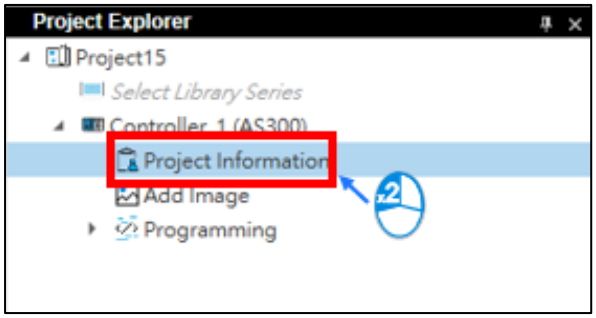


After creating the library project, develop the library in this project. The following table describes the functions of library project.

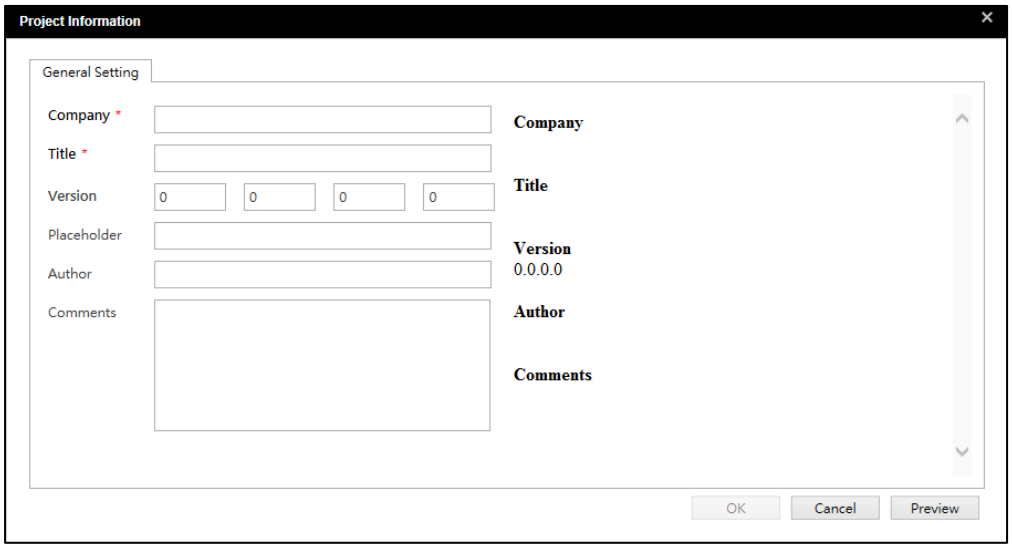
Function	Description
Project Information	Enter library project information, like the title and version information. Refer to Project Information.
Add Image	To store the photos for comments to refer in this project. Refer to Add Image.
Programming - Library Manager	Management of the other libraries which are been referred by this library project. Refer to Library Manager.
Programming - Global Variable	Create global variable for this library. The operation is the same as standard project. Refer to Variables Table.
Programming - User-defined Data Type	Create user-defined data type for this library. The operation is the same as standard project. Refer to User-defined Data Type.
Programming - Function Block	Create function blocks in the library. The operation is the same as standard project, and it can add folders to organize. Refer to Manage a Program Organization Unit (POU).

9.1.2 Project Information

Edit the contents of the library project, and double-click **Project Information** to edit the library project version information before archiving.



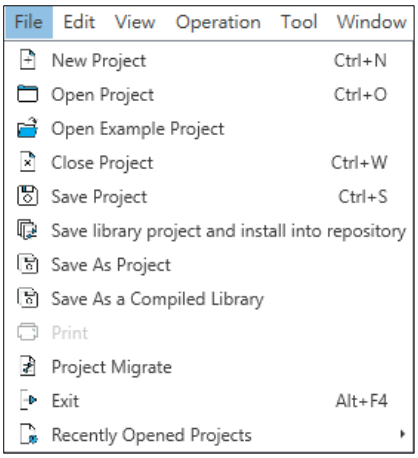
The required fields in **Project Information** are: **Company** and **Title** as shown in the following figure.



Note: AS series and VFD series do not support Placeholder.

9.1.3 Repository Project

The library project provides two archive formats: (1) the library project will be saved as DIA library project format when clicking **Save Project** or **Save As Project**, and (2) the library project will be saved when clicking **Save As a Compiled Library** for compiled library format. The difference is that the latter is a library file whose content is protected. Once closed, it cannot be opened and modified again. In the library manager after installing and adding a new compilation library, it cannot be opened to view the contents.



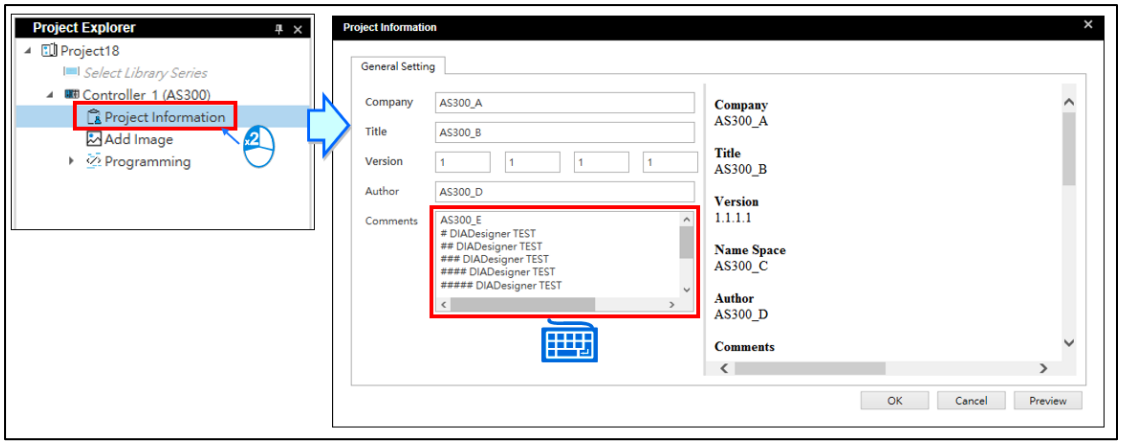
9.1.4 Edit Library Document

Edit Project Information and Function Block document through Markdown syntax and **Add Image** feature.

Note: For detailed Markdown syntax introduction, refer to markdownguide.org/basic-syntax.

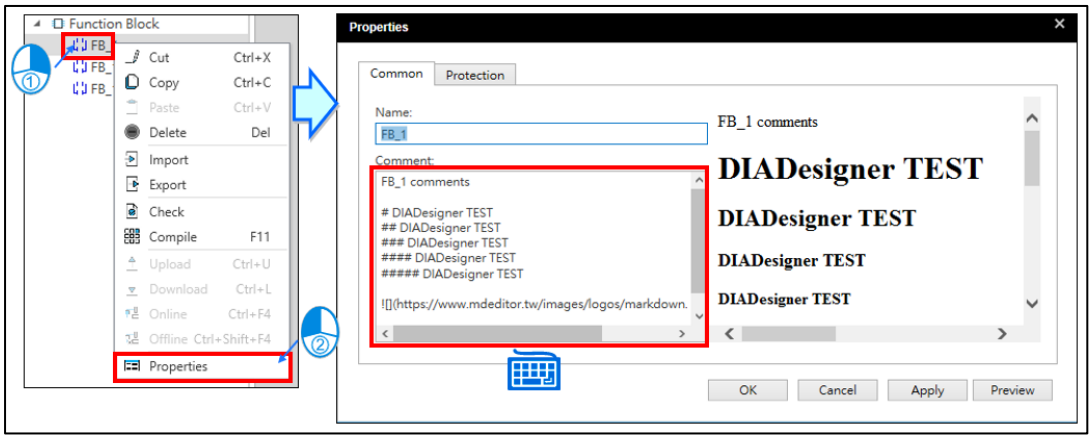
9.1.4.1 Project Information Document Editing

Double-click **Project Explorer** > **Controller** > **Project Information** to open **Project Information** dialog. Use Markdown syntax in **Comment** to edit the document.



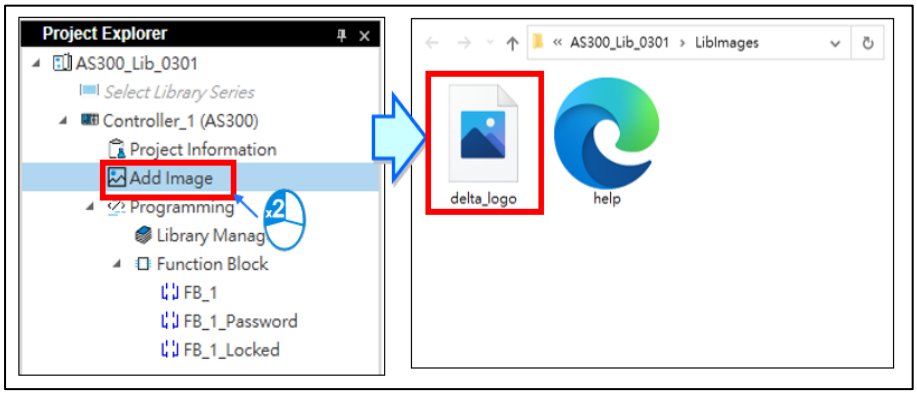
9.1.4.2 Function Block Document Editing

Right-click **Project Explorer** > **Controller** > **Programming** > **Function Block**. Click **Properties** option to open **Properties** dialog. Use Markdown syntax in **Comment** to edit the document.



9.1.4.3 Add Image

Double-click **Project Explorer** > **Controller** > **Add Image** to open file folder location. After placing the image in the folder, add the image to document with Markdown syntax.





9.1.4.4 Examples of Markdown Commands

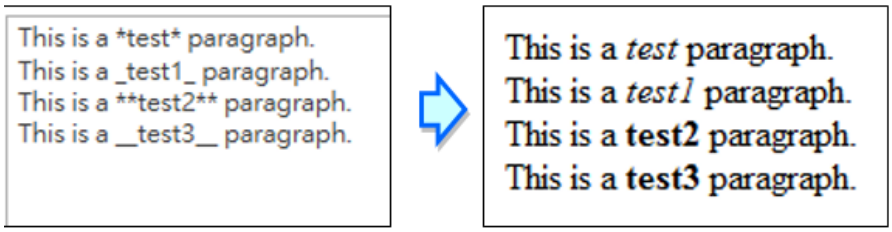
This section introduces the basic Markdown commands and provides examples. For detailed Markdown syntax, refer to markdownguide.org/basic-syntax.

Paragraphs and Line Breaks

Enter two blank characters at the end of the line and click **Enter** on keyboard to automatically wrap the line.

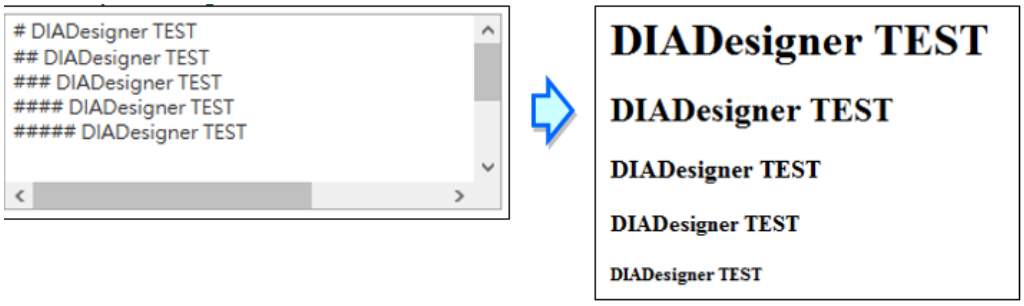
Italic and Bold

Add an asterisk (*) or underscore (_) before and after the word to emphasize, and the word will be italicized. Text is bolded by adding two asterisks (*) or underscores (_) before and after them.



Set Heading

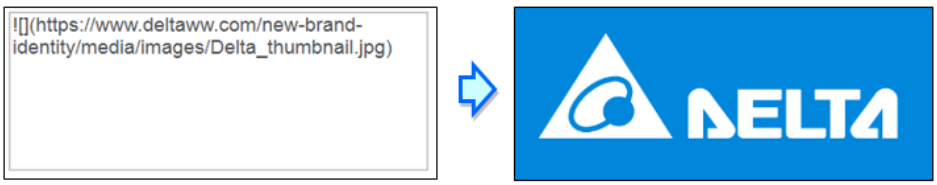
It provides six heading levels. Add a **#** before the text for the first level heading, add two **#** for the second level heading, and so on.

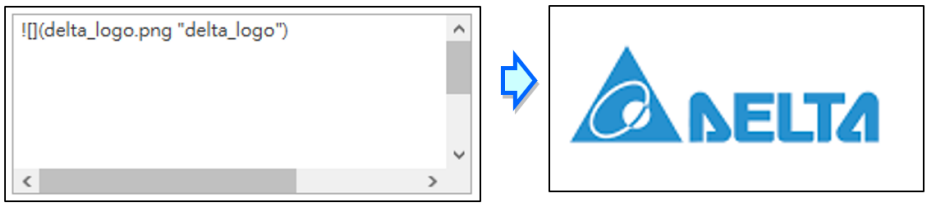


Insert Picture

The syntax is as follows: an exclamation point, with square brackets, and the alt text for the image inside. In addition to ordinary brackets, put the URL of the image, or the name of the file in the folder mentioned in Add Images.

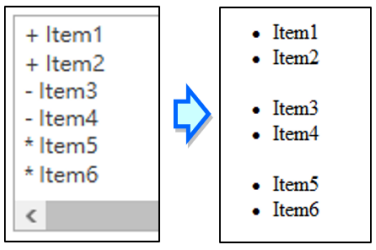
Note: Currently Markdown does not support image resizing, and the original size of the image will be displayed.



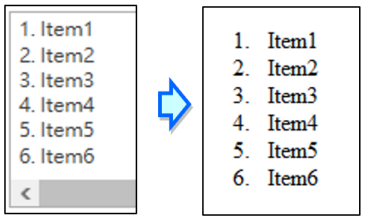


Insert List

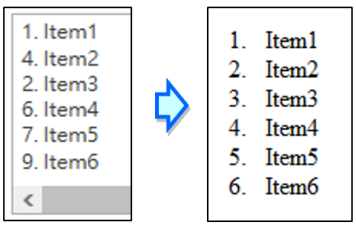
Markdown supports ordered and unordered lists. Unordered lists use asterisks, plus signs, or minus signs as list markers.



Ordered lists use numbers followed by a period as list markers.



Note: If the list markers are not in sequential numerical order, the output is corrected to the correct numerical order.



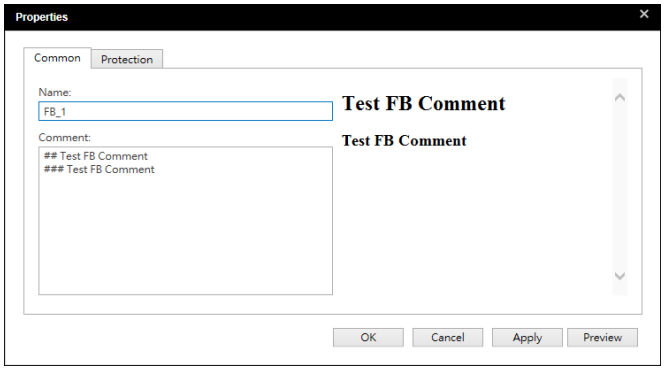
9.1.4.5 Multi-language Creation of Library Document

To display library comments in different languages, create multi-language documents. Take English as an example. Follow these steps to create multi-language documents.

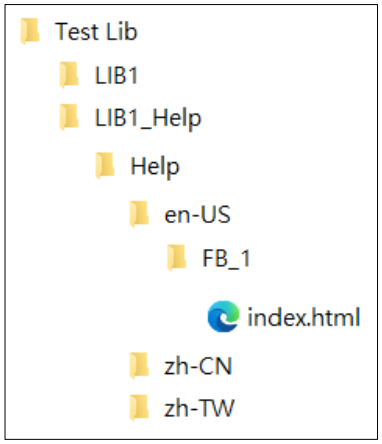
To create multi-language documents

1. Add a function block and enter the comment, and then click **OK**.
2. Save the library project.

   In the library project storage path, a folder named **Library Name_Help** is automatically created.



3. In the folder, you can find the **en-US** folder and comment document (.html)



4. Copy all the contents in the **en-US** folder to the language folder where you want to display the multi-language. Take Traditional Chinese as an example, copy the contents to the **zh-TW** folder.
5. Use the text editor to change the content of the **index.html** file under the **zh-TW folder** to Traditional Chinese, and then save it.

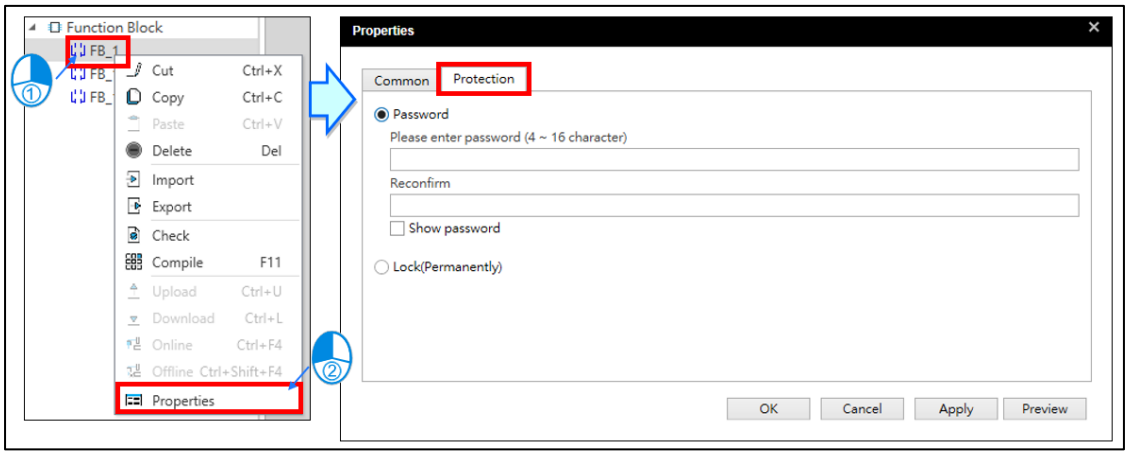


Repeat the same steps for the other languages according to your needs. After the edit of the multi-language document is complete, the multi-language content will automatically display when you import the library project into the standard project. If the content of the document is changed, refer to Load Library Document to load the changed multi-language document for viewing.

9.1.5 Library Protection

While saving project with DIA Library Project format, you can set password or lock specific function block to protect development content. You can also choose to save the compiled library as an encrypted method, refer to Repository Project.

Right-click **Project Explorer** > **Controller** > **Programming** > **Function Block**, choose **Properties** and click **Protection** tab in the **Properties** dialog.



9.1.5.1 Password Protection

Enter password to use the function block after protection is set up.

Follow these steps to complete password setting.

To complete password setting

1. Enter customized password in **Please enter password** and **Reconfirm** field.
2. Click **Apply** to finish password setting.

9.1.5.2 Cancel Password Protection

Follow these steps to cancel password setting.

To cancel password setting

1. Enter customized password in **Please enter password** field.
2. Click **Apply** to cancel password setting.

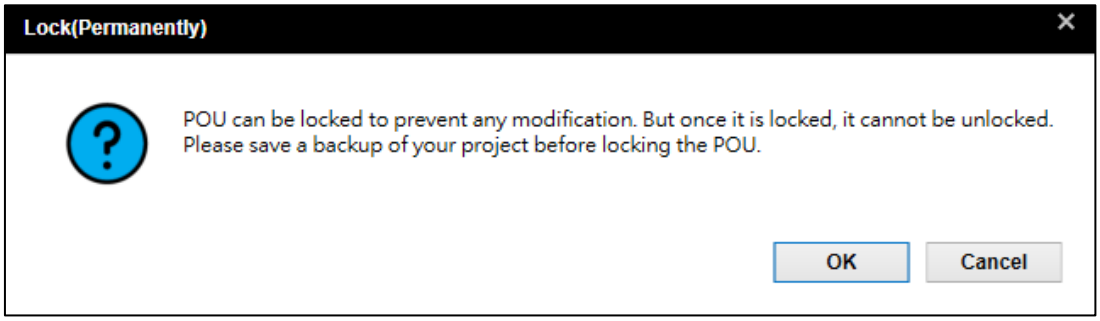
9.1.5.3 Lock Protection

After applying Lock Protection, the function block will be locked permanently.

Follow these steps to apply Lock Protection.

To apply Lock Protection

1. Check **Lock(Permanently)** option and click **Apply**.
2. Warning message appeared, click **OK** to finish setting.



9.1.6 Library Export and Import

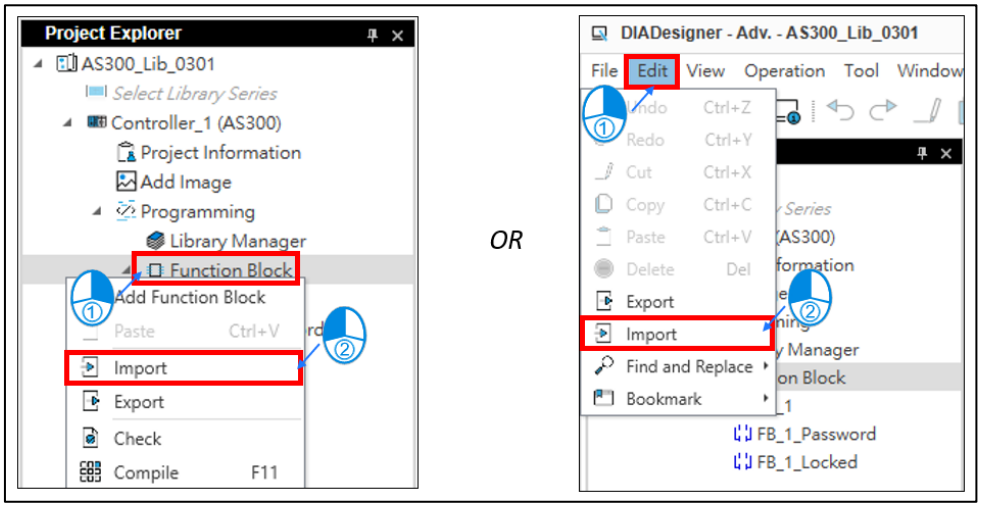
Library Project supports the function to Export / Import Function Block.

9.1.6.1 Library Export

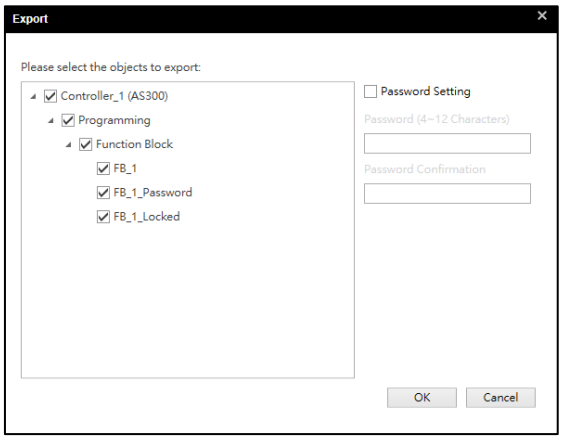
Follow these steps to perform Export function.

To perform Export function

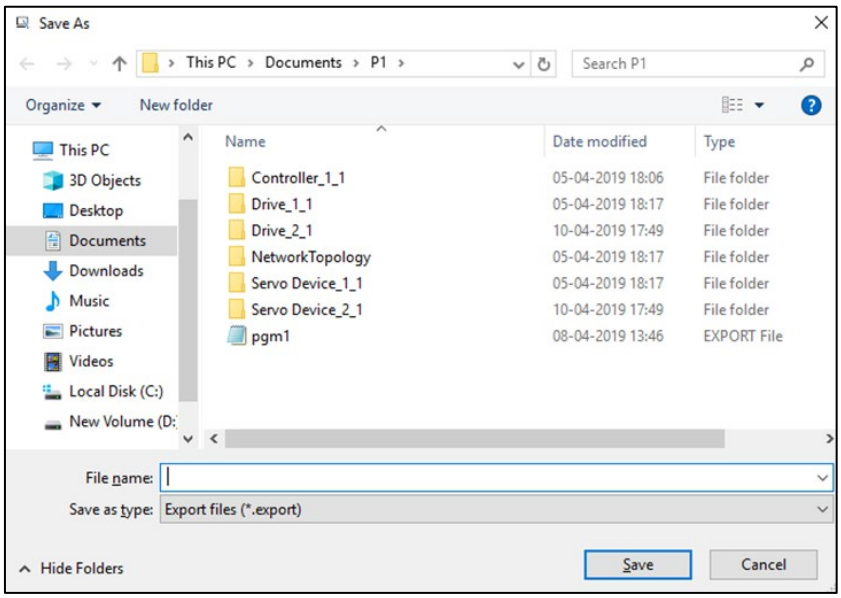
1. Right-click the Function Block to be exported in **Project Explorer**.
   Click **Export** in the menu.
   Or
   Click **Edit** > **Export** on toolbar.



Export dialog appears. The chosen Function Block will be checked automatically as shown in the following figure.



2. Choose Function Blocks and press **OK** to open Save As dialog. To protect the export file, check **Password Setting** and input **Password** and **Password Confirmation** accordingly.



3. Choose the path and click **Save** to finish export.

9.1.6.2 Library Import

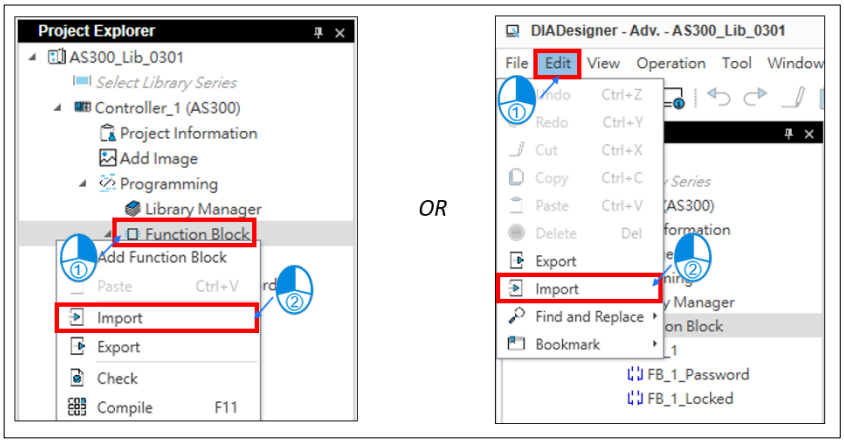
Follow these steps to perform Import function.

To perform Import function

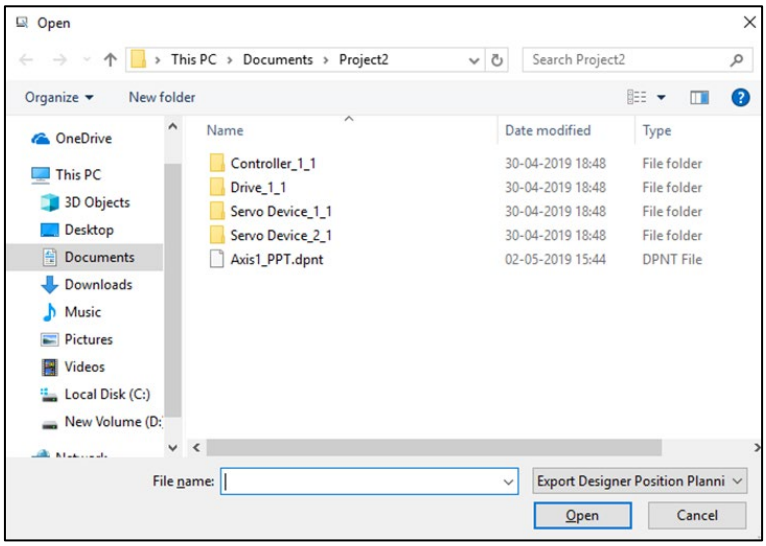
1. Choose the Library series to be imported from the **Project Explorer**.
2. Click **Import** from the menu.

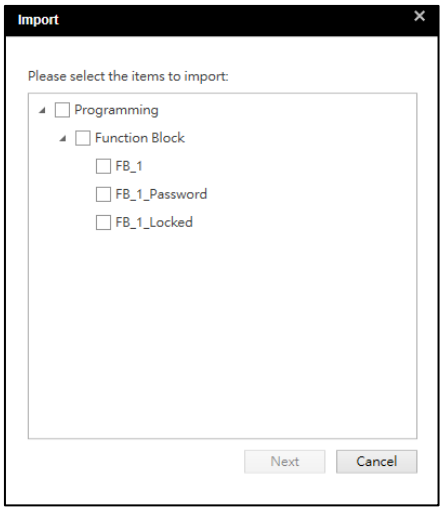
   Or

   Click **Edit** > **Import** on toolbar.



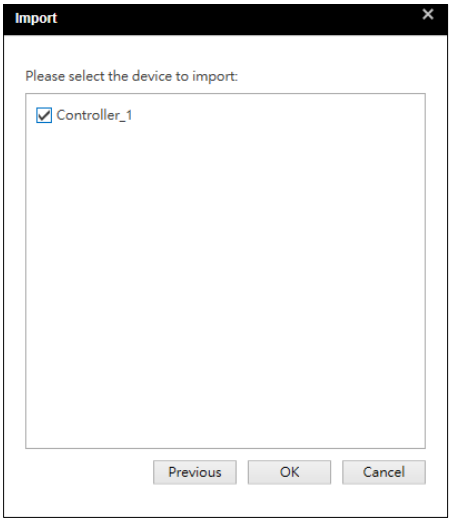
3. Choose the file to import and click **Open** to open **Import** dialog.



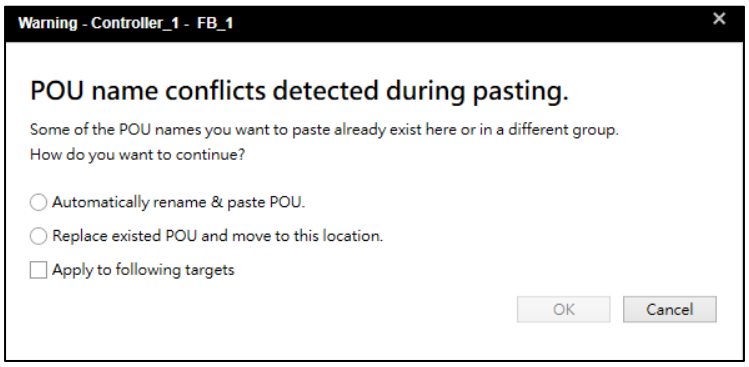


4. Choose the item to import and click **Next**.

5. Select the device to import and click **OK** to import POU.



If the file is protected, a password is required to be entered to perform following actions. If the imported POU have a same name as the POU in the device, an alert message appears. You can choose to rename or replace the POU in the device.



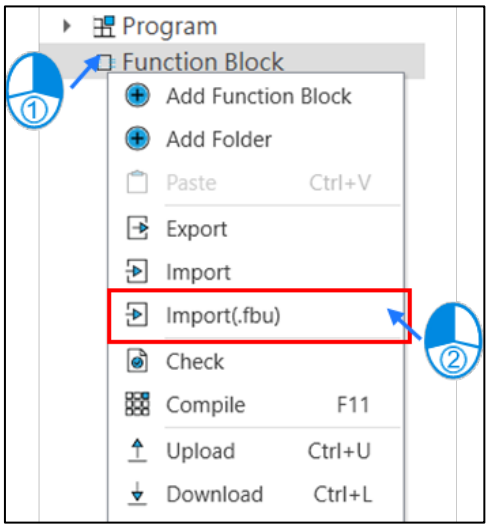
9.1.6.3 Library Import (.fbu)

DIADesigner supports the function of importing POUs of function block exported from ISPSoft .fbu file. This function increases the convenience of reusing the existing project.

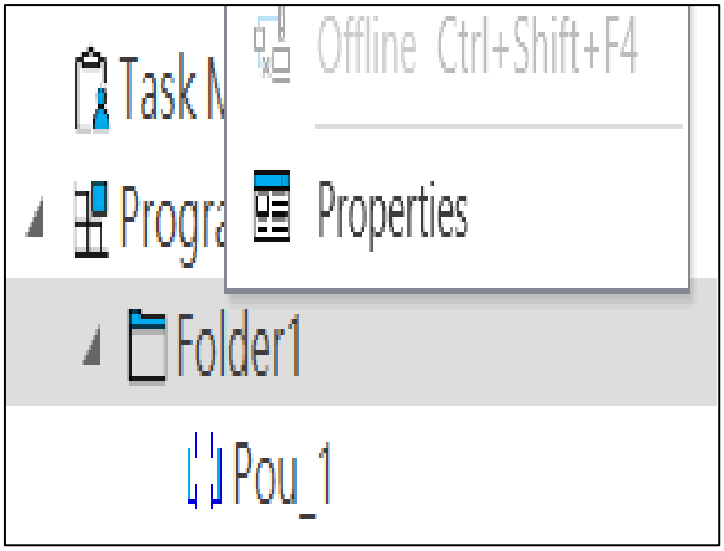
Follow these steps to import a POU (.fbu) of Function Block type.

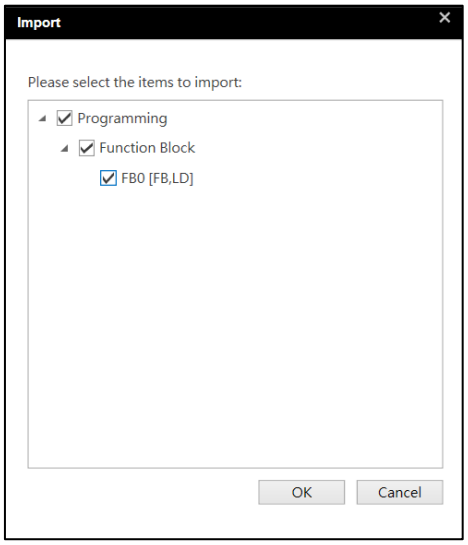
To import a POU (.fbu) of Function Block type

1. Select **Function Block** in **Project Explorer**.
2. Right-click **Function Block** to open the context menu and click **Import (.fbu)**.



3. Select the file to import and click **Open**. The function block file extension is .fbu.





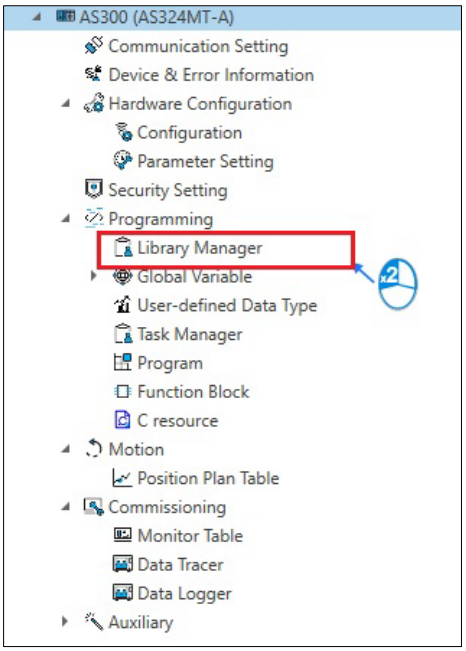
4. Select the items to import and click **OK** to import Function Block.

Note: Refer to *ISPSoft User Manual* for export function of ISPSoft.

9.2 Library Manager

The library can be installed, removed, added, and deleted in the library manager (the format must be DIA Library Project and Compiled Library Files), and can be used for calling in the program.

1. Double-click **Controller Device** > **Programming** > **Library Manager** in **Project Explorer** to open **Library Manager** dialog.



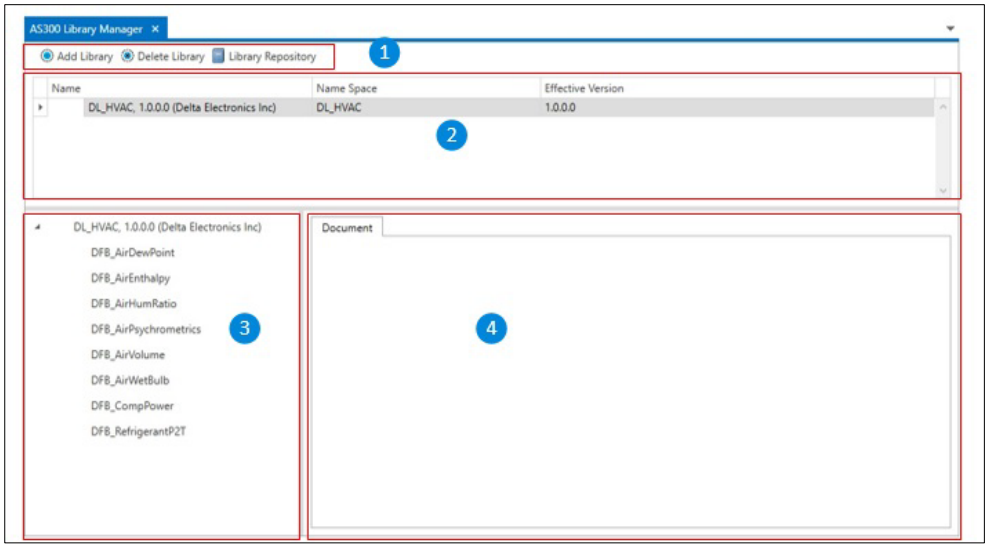
2. There are following areas in the **Library Manager** dialog.

1. Toolbar: Provide library **Resource Library** (installation, removal), **Add Library**, and **Delete Library**.

2. List of newly added libraries.

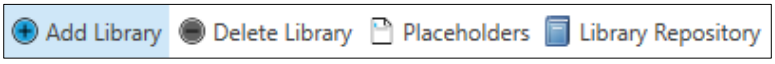
2. When the left mouse button clicks on an item, there will be a corresponding annotation on the right (4); Double-click the project, the original content code will be opened, if it is Compile library, a warning message will pop up and it cannot be opened.

4. Annotations corresponding to each item.

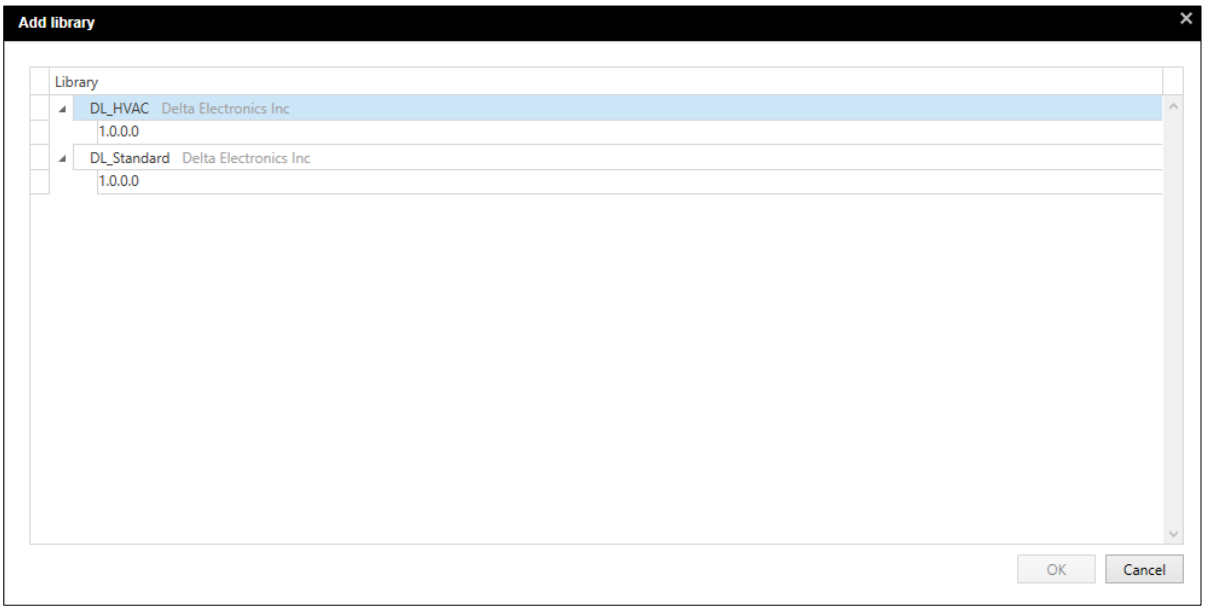


9.2.1 Add Library

Click **Add Library** for the installed library project.

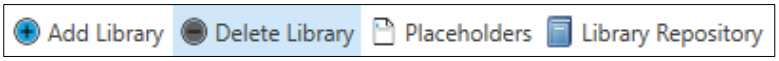


It adds on the library manager page, and you can use the content in the program.



9.2.2 Delete Library

Click **Delete Library** to delete the library project.



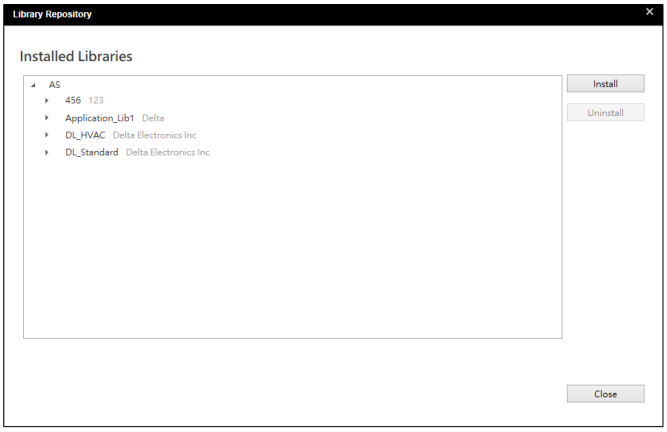
It will be deleted on the library manager page. If you refer to the deleted library in the program, the compilation will report an error.

9.2.3 Library Repository

The library can be used for installation and removal of library projects and library compilation files.



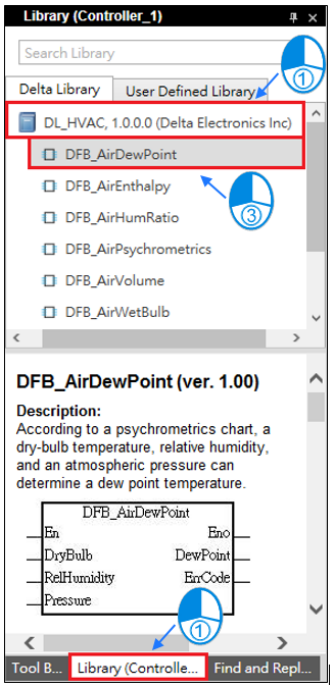
After installation, the library version, company and version number will be displayed. Multiple different versions can be installed with the same library name, and you can click the specified version after installation; the built-in Delta library cannot be removed.



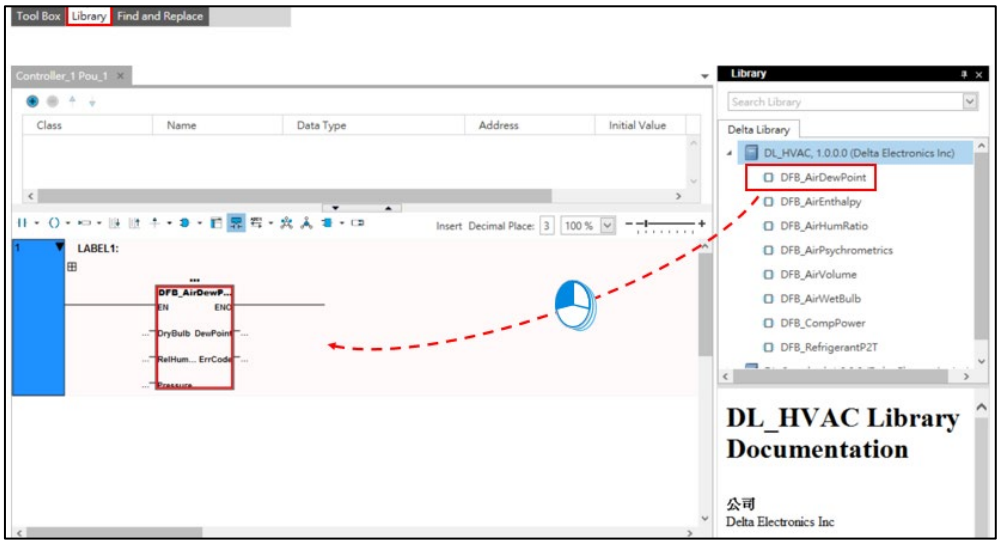
9.2.4 Library Usage

After the library project is installed and added, call and use the custom data type and function block provided under the library project in the program area. The following takes the built-in DL_HVAC library of Delta as an example, and users want to call in the LD POU DFB_AirDewPoint under DL_HVAC.

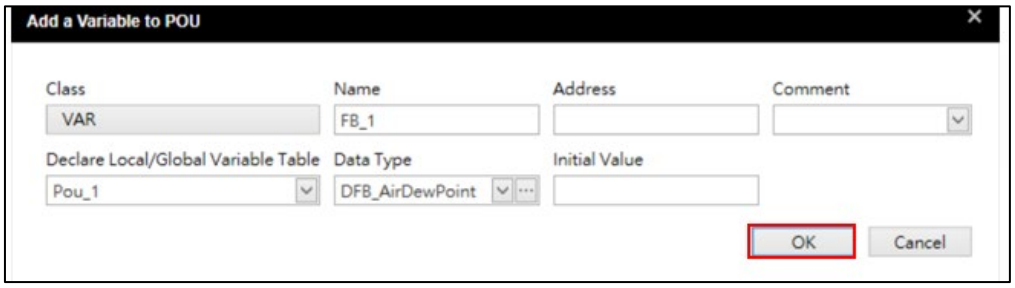
1. Find DFB_AirDewPoint in the **Library** dialog on the right side of the screen.

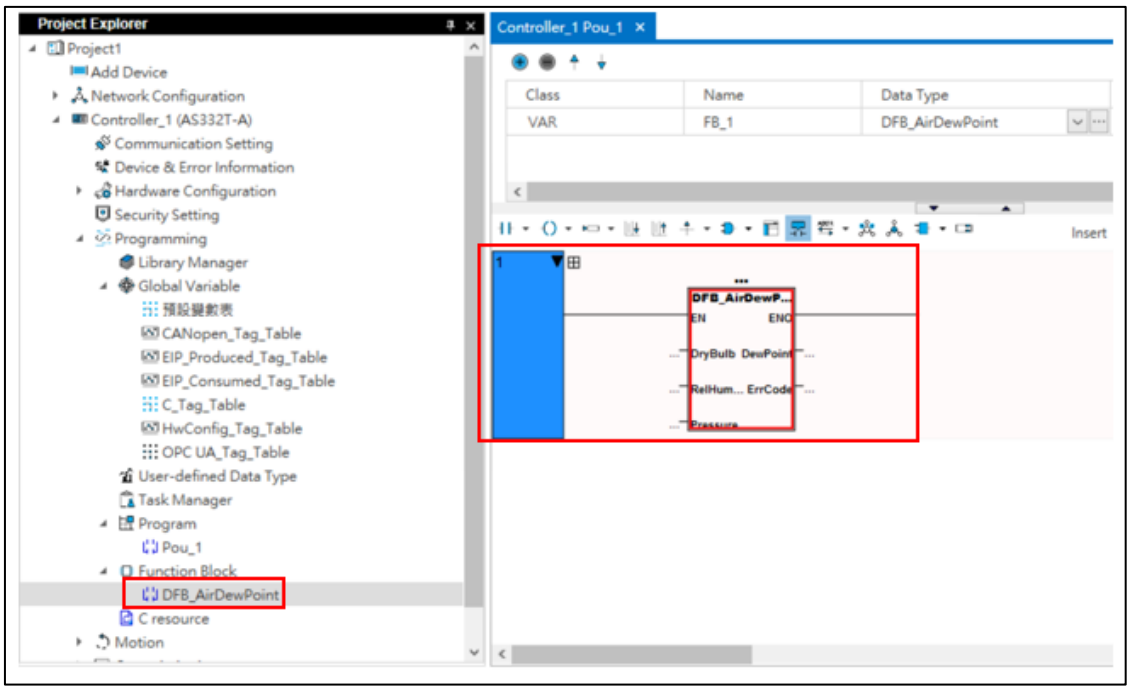


2. Drag DFB_AirDewPoint to the POU screen to add commands to the project.

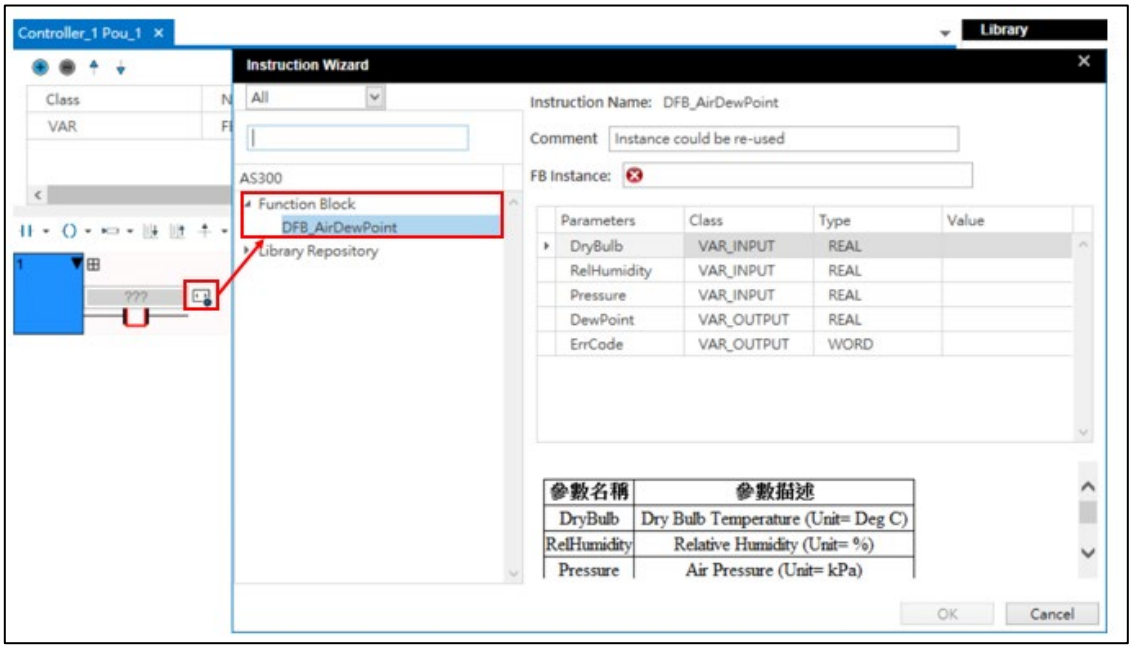


3. Add the corresponding parameter, click **OK**, the command is successfully added to the project.

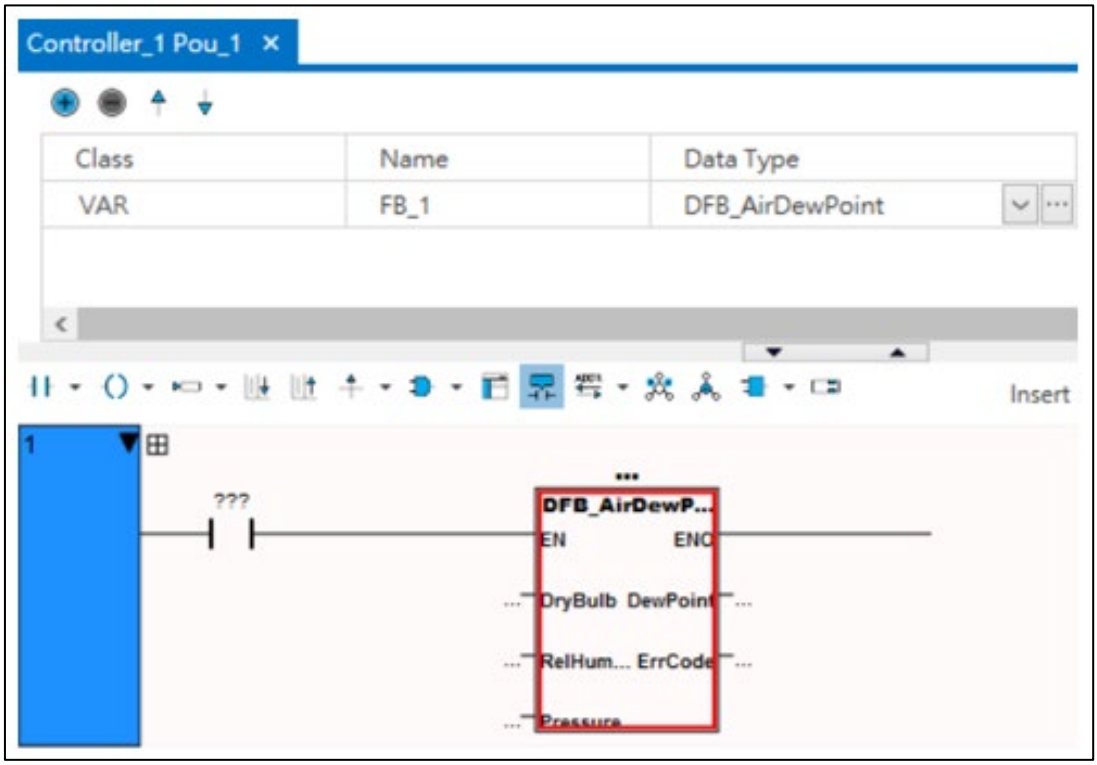




4. After the command is added to the project, the DFB_AirDewPoint under the selected resource library can be called again through the command wizard.

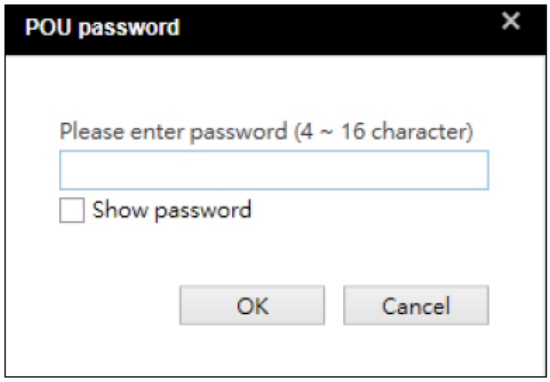


5. After selecting DFB_AirDewPoint, it can be successfully used in the program area.



9.2.4.1 Edit Password Protected Function Block in Library

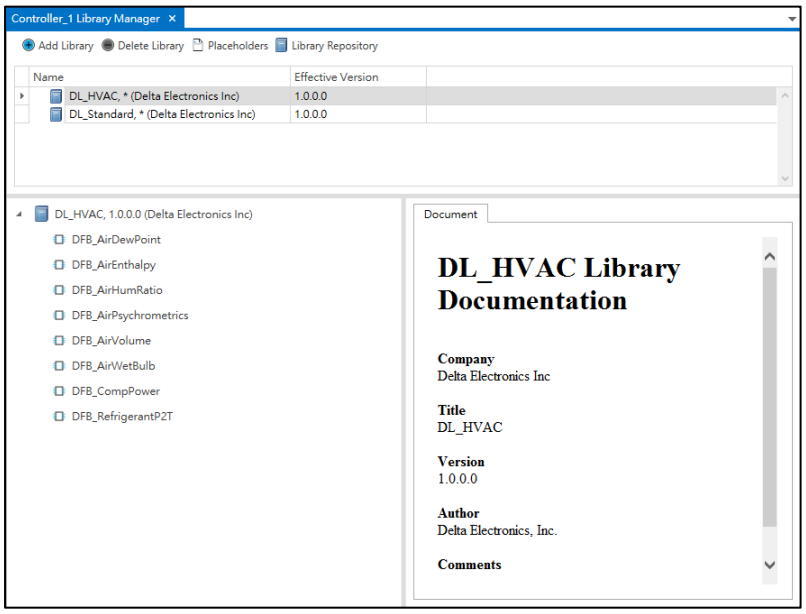
To edit a password protected function block in Library, enter password in the password dialog first.



Note: Function Block cannot be edited if it is protected by Lock (Permanently) or save as Compiled Library.

9.2.4.2 Load Library Document

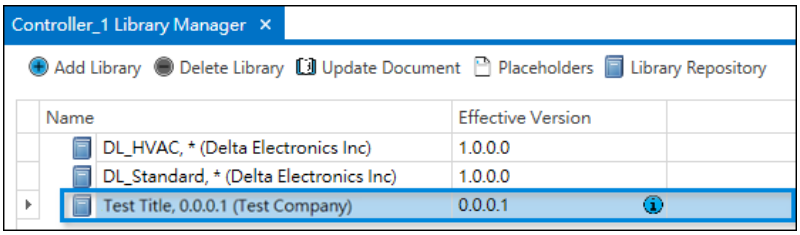
Select the library name in **Library Manager** and view the corresponding content in the bottom right pane.



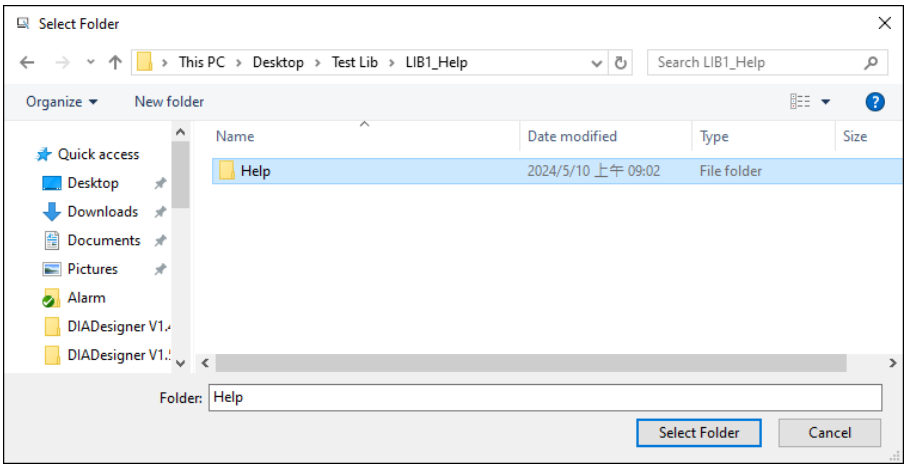
If you have created the multi-language documents and make changes to them, you can view the updates through the Update Document function.

Use the Update Document function to view the updates

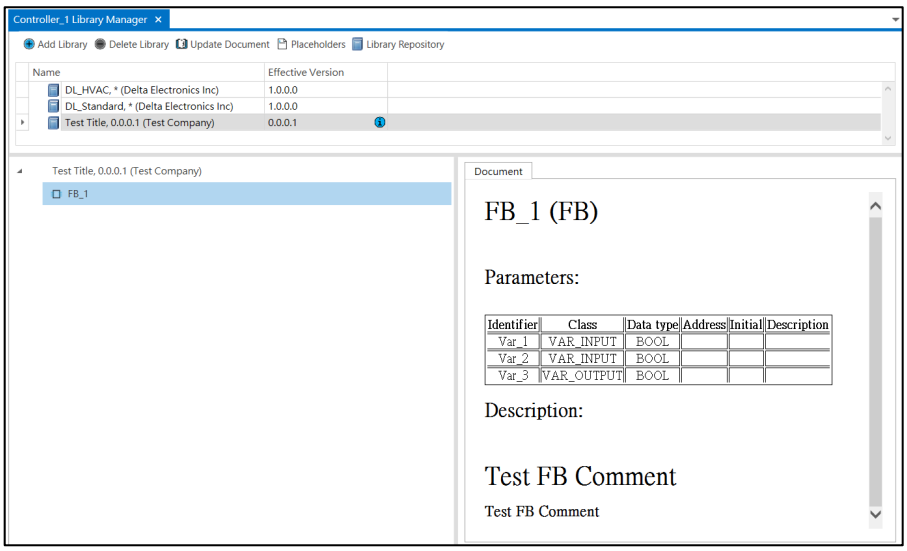
1. After performing the steps in Multi-language Creation of Library Document, click **Add Library**.
2. Select the library to update the document and click **Update Document**.



3. Click **Help** folder under the library directory and click **Select Folder**.



The multi-language content of the function block in the directory has been updated.



Chapter 10: Motion Tools

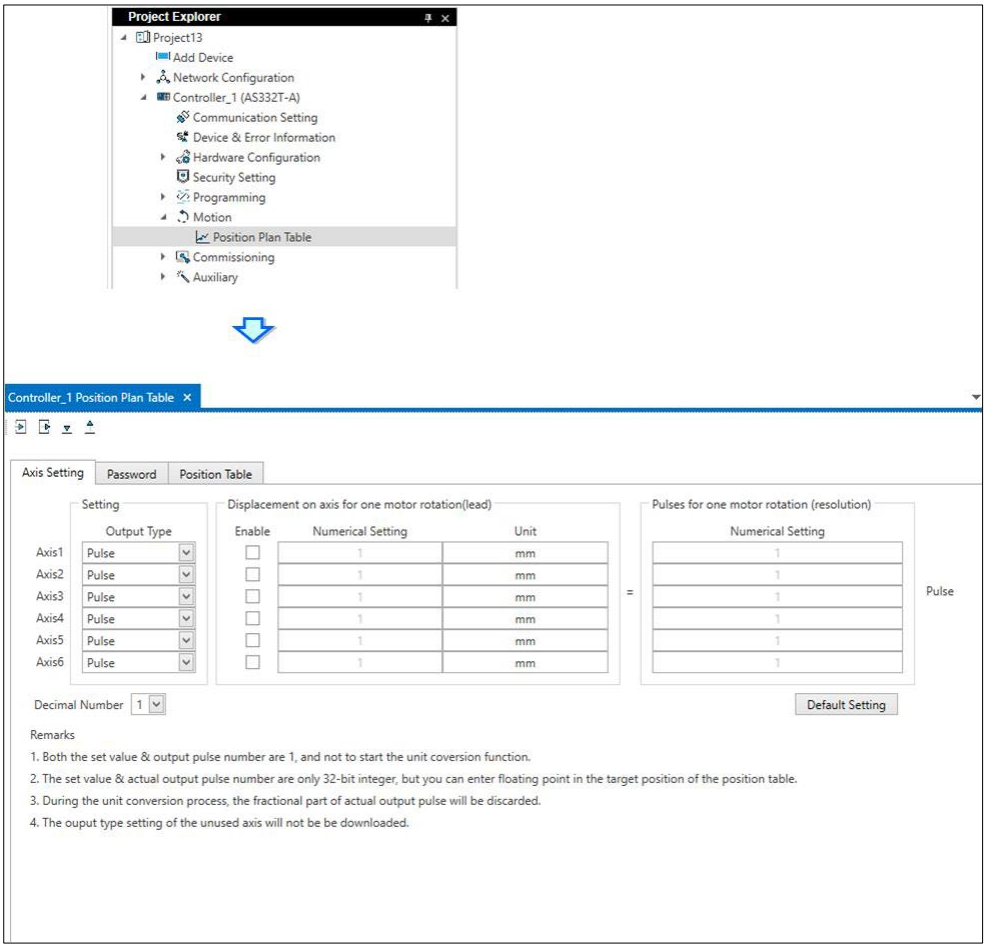
10.1 Position Plan Table

10.1.1 About Position Plan Table

DIADesigner provides an easy method to set up the single-axis or 2-axis motion control. It is not required to know the complicated motion control rules to complete the setup. Create and download the **Position Plan Table** and then along with the instructions, the controller can output axis motions in the set number and order accordingly.

10.1.2 Setting Position Plan Table

Double-click **Controller** > **Motion** > **Position Plan Table**.



The **Position Plan Table** dialog has three tabs.

- Axis Setting

- Password
- Position Table

The following table describes the icons in **Position Plan Table** toolbar.

Icon	Description
	Click to import Position Plan Table.
	Click to export Position Plan Table.
	Click to download Position Plan Table.
	Click to upload Position Plan Table.

10.1.3 Axis Setting

Set control output for each axis in the **Axis Setting** tab. The number of the axis display varies according to the supported controller (e.g. AS300 series – 6 axis, DVP series – 4 axis).

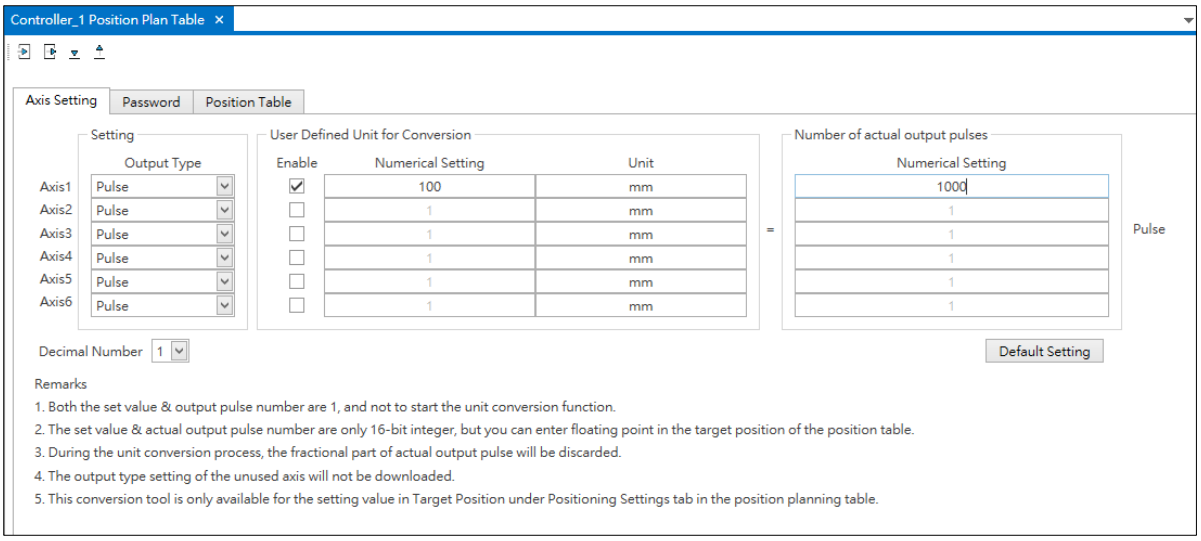
The following table describes the functions in **Axis Setting** tab.

Function	Description
Setting	
Output Mode	Select **Output Type** for the axis from drop-down list. The available options include: • Pulse • Pulse/Dir • AB Phase **Note**: **Pulse** is the default value. When executing 2-axis interpolation movement, **DO NOT** choose **Pulse** in **Output Mode**, instead choose **Pulse/Dir** or **AB Phase**.
User Defined Unit for Conversion	
Enable	Select **Enable** checkbox of the axis which needs to make a unit conversion.
Numerical Setting	Enter the value for **Numerical Setting** of the axis.
Unit	Enter the engineering unit to set the unit conversion between a value and actual pulse output in the **Position Plan Table**.

Function	Description
Setting	
Number of Actual Output Pulse	
Numerical Setting	Enter the pulses for one motor rotation for unit conversion purpose.
Default Setting	Click **Default Setting** to restore the default values on the **Axis Setting** tab.
Decimal Number	Decimal number decides the upper limit of the number of decimal digits in a simulation view when there are decimals in a unit conversion. The available options include: • 0~6 **Note**: **1** is the default value.

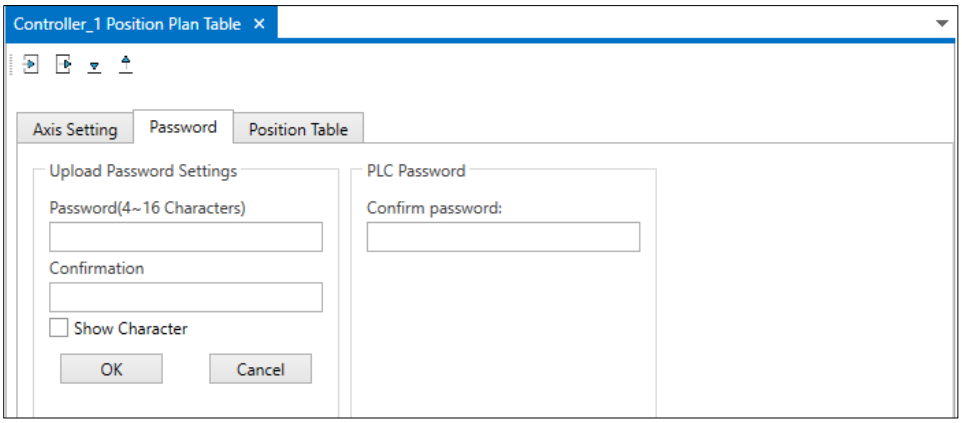
Example

There are 1000 pulse output every 100 units for Axis 1 with the unit of mm in the **Position Plan Table**, which displays in the simulation view as shown in the following figure.



10.1.4 Password

The **Password** tab is used for protecting the **Position Plan Table** which has been downloaded to the CPU module from being uploaded or read by other people.



The following table describes the functions in the **Password** tab.

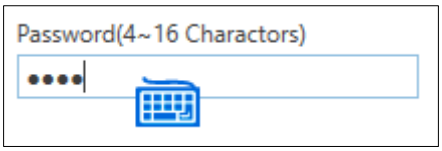
Function	Description
Upload Password Settings	
Password	Enter password to protect the **Position Plan Table** while uploading from PLC. **Note**: 4~16 characters can be provided.
Confirmation	Enter the above password once again to confirm it.
Show Character	Click to make the characters visible while typing the password.
OK	Click to save the setting in **Password** tab.
Cancel	Click to cancel the changes done.
PLC Password	
Confirm Password	Enter PLC password.

10.1.4.1 Setting Password

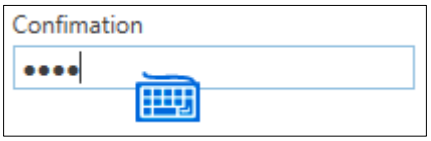
Follow these steps to set up password for the **Position Plan Table**.

To set password for Position Plan Table

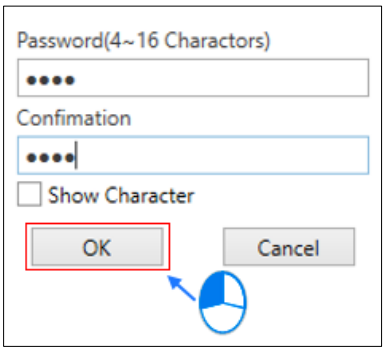
1. Enter a password in the **Password** box.



2. Enter the same password in the **Confirmation** box.



3. Click **OK**.

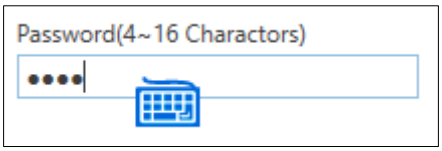


10.1.4.2 Cancelling Password

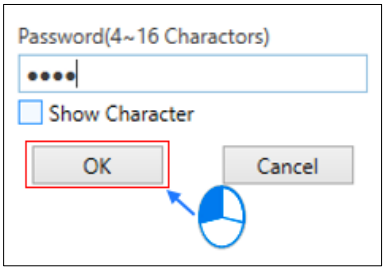
Follow these steps to cancel the password set for a **Position Plan Table.**

To cancel password set for Position Plan Table

1. Enter the password in the **Password** box.



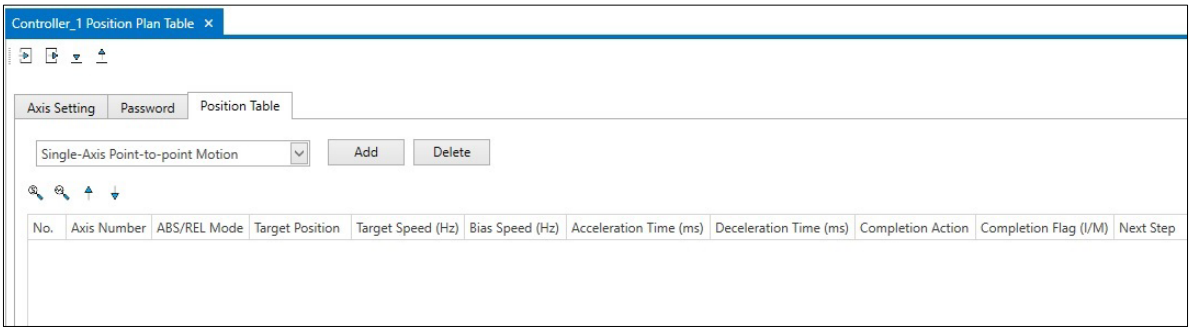
2. Click **OK**.



Note: After the password setting are downloaded to the CPU module, a dialog appears asking you to type the correct password whenever an upload of the **Position Plan Table** is needed. Note that the upload password only protects the **Position Plan Table** in the CPU module from being read by other people. To download a new **Position Plan Table** to the CPU module where there has already been one, a new **Position Plan Table** will overwrite the existing one whether there is an upload password set for the existing one in the CPU module or not.

10.1.5 Position Table

In the **Position Table** tab, you can add, modify or delete a position plan.



The following table describes the functions available in **Position Table** tab.

Function / Icon	Description
Position Table mode drop-down list	Select the **Position Table** mode. The available options include: • Single-Axis Point-to-point Motion • Single-Axis Multi Segment Motion • 2-axis Linear Interpolation • 2-axis Arc Interpolation **Note**: The **Single-Axis Point-to-point Motion** is the default value.
Add	Click to add items.
Delete	Click to delete items.
	Click to perform single-axis simulating mode.
	Click to simulate by selecting item.
	Click to move an item up.

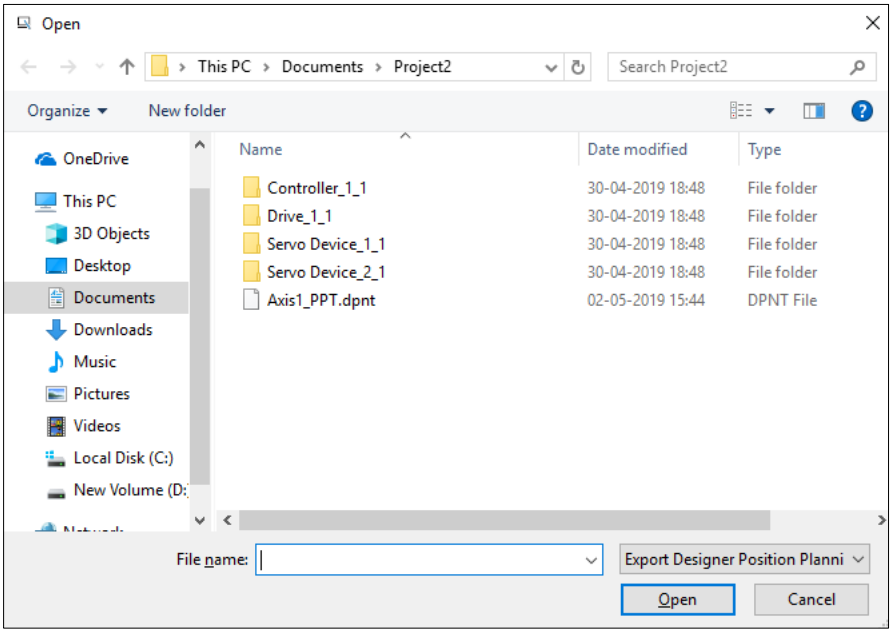
Function / Icon	Description
	Click to move an item down.

10.1.5.1 Import Position Plan Table

Follow these steps to import Position Plan Table.

To import Position Plan Table

1. Click  to open **Open** dialog.



2. Select the DIADesigner Position Planning Table file (*.dpnt) to import.

   Note: You can also import the Position Plan Table file (*.pnt) from ISPSoft software.

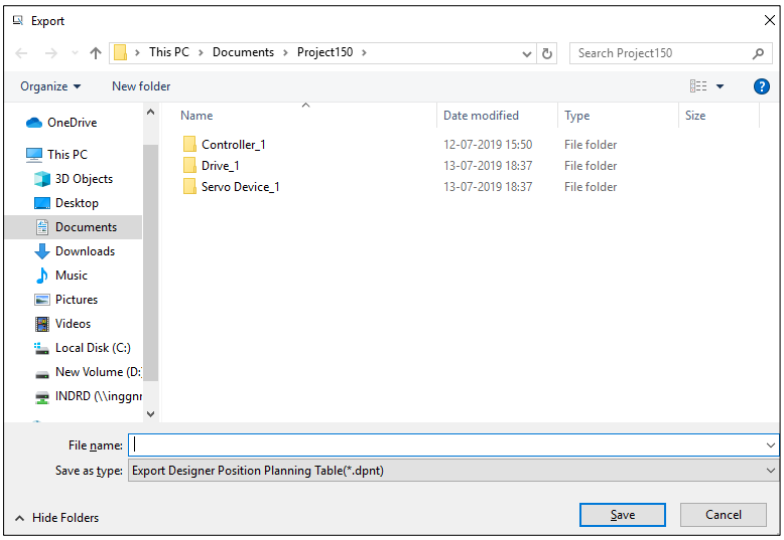
3. Click **Open**.

10.1.5.2 Export Position Plan Table

Follow these steps to export Position Plan Table.

To export Position Plan Table

1. Click  to open **Export** dialog.

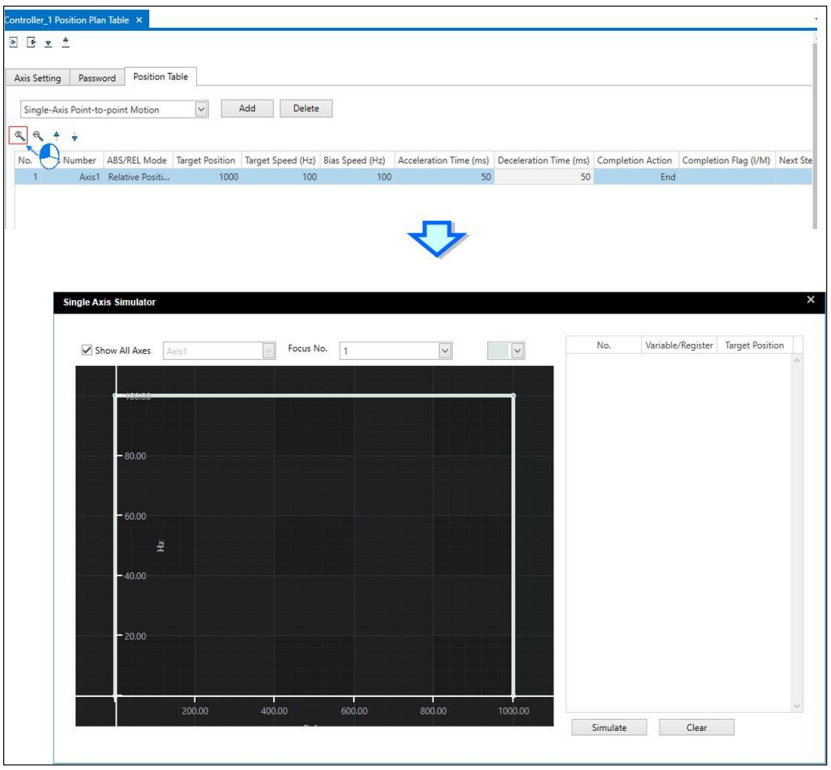


2. Enter a name for the *.dpnt file to be created.

3. Click **Save**.

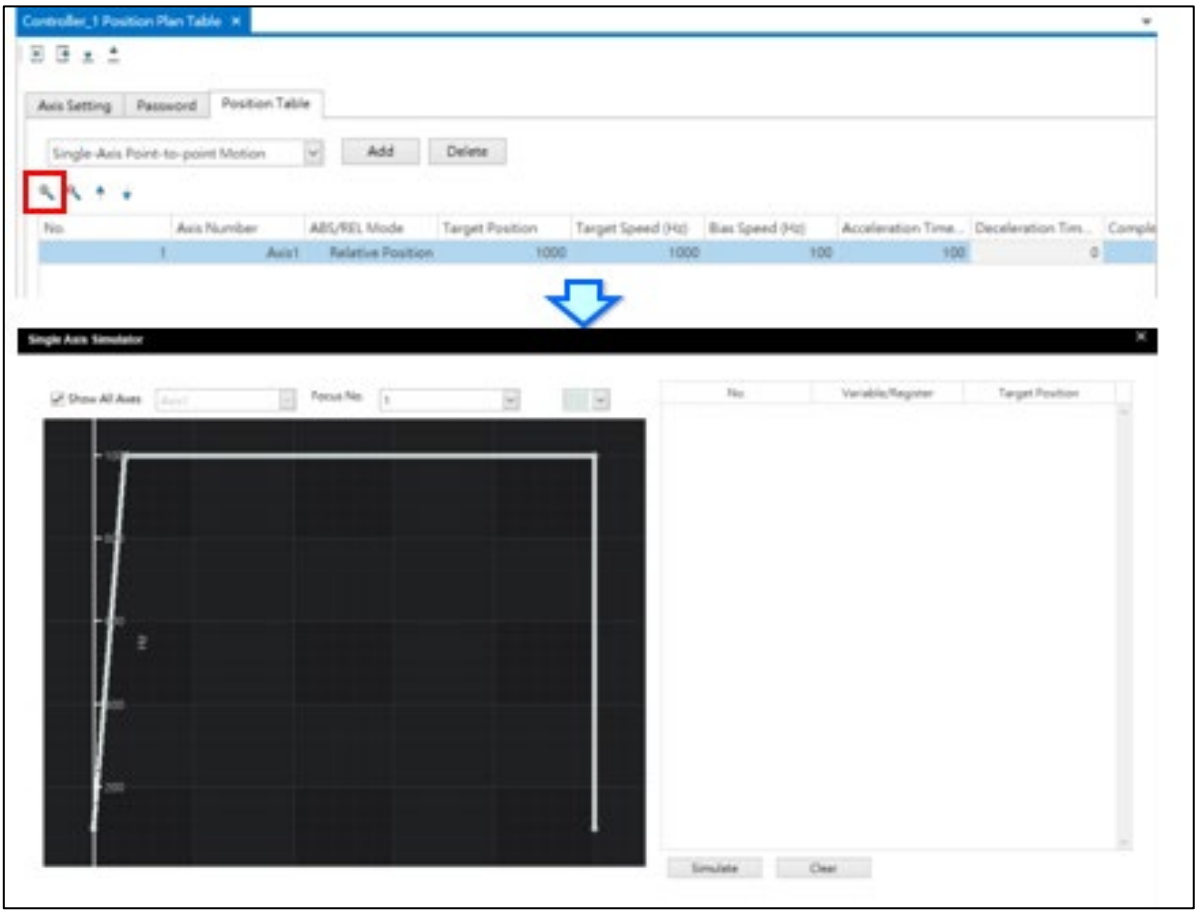
10.1.6 Position Plan Table Simulation

After a **Position Plan Table** is set, the path for the table is shown through a graphical simulation. Click any row of the table that is to be simulated, and then click  or  to open **Position Plan Table Simulation** dialog.



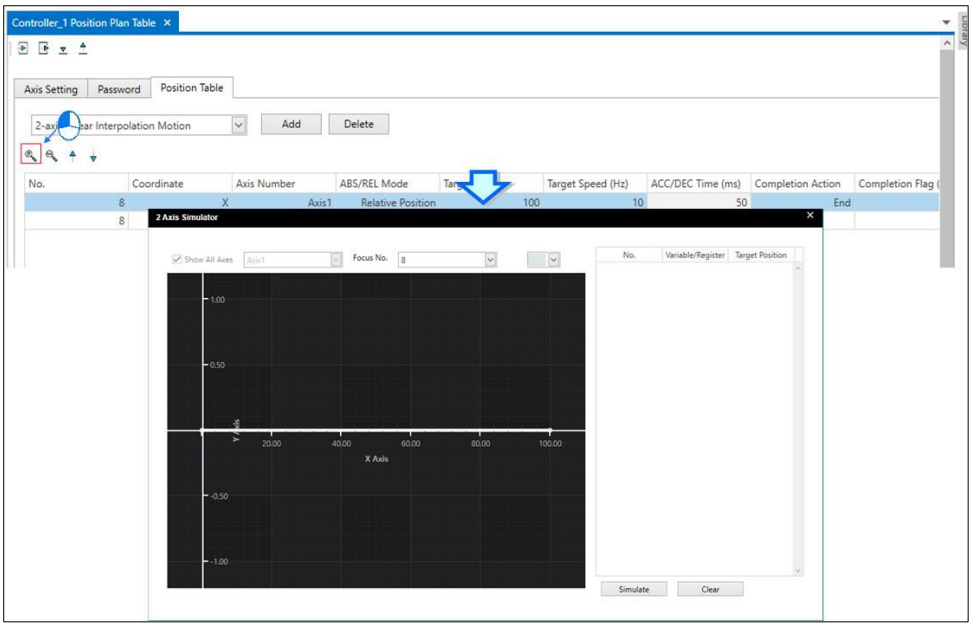
For the 2-axis linear interpolation and 2-axis arc interpolation, in the mixture of two motion modes, the continuous path is shown in the same simulation view such as that in the above window including the linear interpolation and arc interpolation.

For the single-axis point-to-point motion and single-axis multi-segment motion, only the path of the same motion mode can be shown in the simulation view. The following is the simulation view of the single-axis point to point motion.

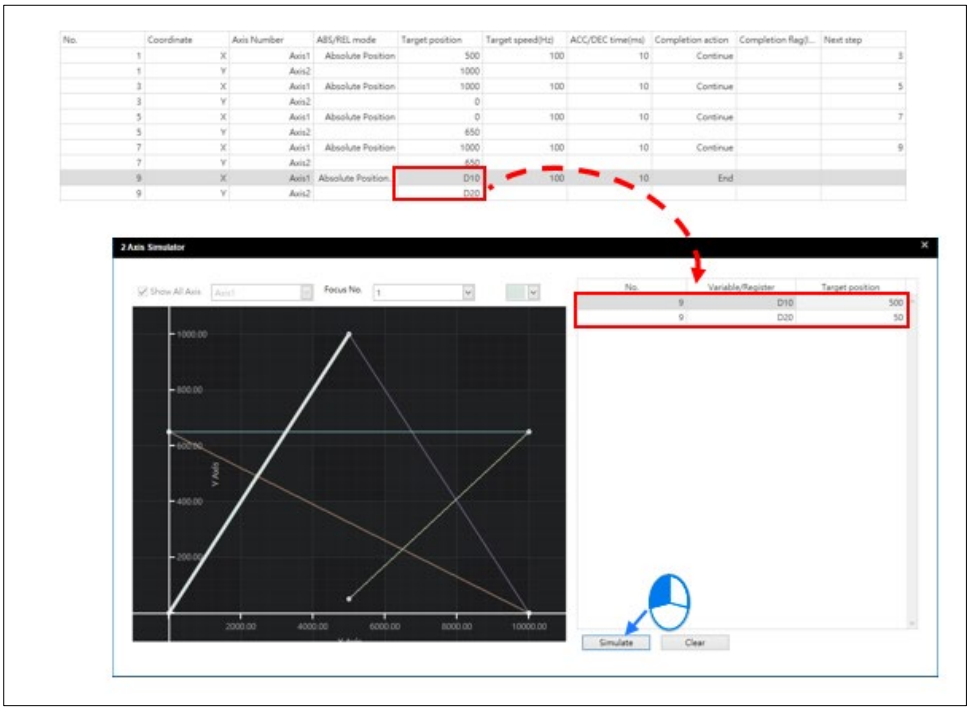


The Position Plan Table also supports the simulation starting from a row of the position table in one single mode. First, select the row of the table as the start position for the simulation. After  is clicked, the simulation dialog displays a simulation view starting from the selected row of the table until the end.

In the following window, the drawn path starts from number 5, which is part of the linear interpolation in the previous simulation view.

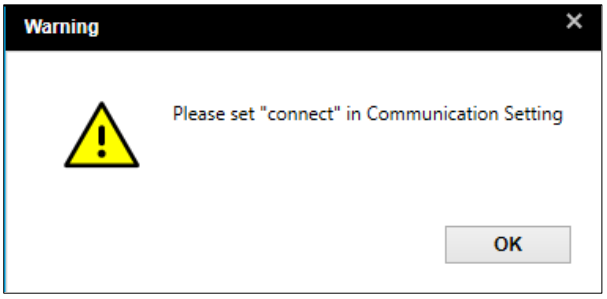


If there is any target position sat by a variable or a register, the variable/register list will display in the right side of simulation window. Set the value in **Target Position** box of those in the Variable/Register field then press **Simulate** to display on the chart in the left side.



10.1.7 Upload and Download Position Plan Table

When uploading or downloading **Position Plan Table**, connect DIADesigner to a host. If the host is not connected, a warning message appears, prompting you to connect DIADesigner to the host.



Note: When the PLC password is set and you want to upload and download the Position Plan Table, go to the **Password** tab to enter the PLC password.

10.1.7.1 Download Position Plan Table

Follow these steps to download **Position Plan Table** to a host.

To download Position Plan Table to host

1. Click  to download **Position Plan Table** to a host.

   Position Plan Table download status **Download success** displays as shown in the following figure.



10.1.7.2 Upload Position Plan Table

Follow these steps to upload **Position Plan Table** from a host.

To upload Position Plan Table from host

1. Click  to upload **Position Plan Table** from a host.

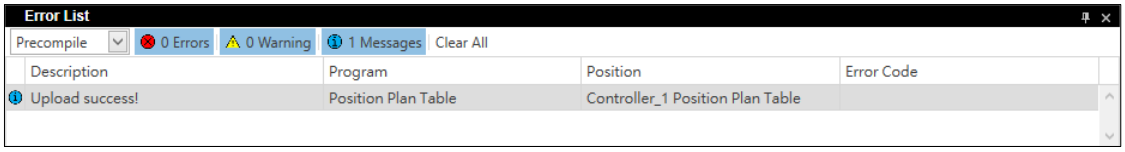
   Position Plan Table Password dialog appears as shown in the following figure.



Note: The **Position Plan Table Password** dialog appears only if a password is set during previous download. If the password is not set, then DIADesigner will start uploading directly.

2. Enter password and click **OK**.

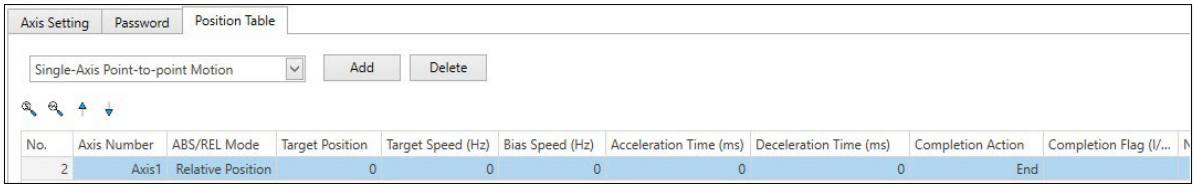
   Position Plan Table upload status – **Upload success!** appears as shown in the following figure.



10.1.8 Control Mode

10.1.8.1 Single-Axis Point-to-point Motion

Set the point-to-point positioning control of an axis in this mode. The following figure shows an example of **Single-Axis Point-to-point Motion** control.



- **No.**: The row number which is created in order automatically by the DIADesigner. Maximum number: **200**.
- **Axis Number**: Select the axis which is controlled by the **Position Plan Table**.
- **ABS/REL Mode**:
  - **Relative Mode**: Use the current coordinates of axis to add or reduce position, values are required for each field.

- **Absolute Mode**: Use absolute coordinates of axis as target position, values are required for each field.
- **Relative Mode (Variable or Address / Target position)**: Use the current coordinates of axis to add or reduce position, and the target position can be variable or address.
- **Absolute Mode (Variable or Address / Target position)**: Use absolute coordinates of axis as target position, and the target position can be variable or address.
- **Relative Mode (Variable or Address / Target position & Speed)**: Use the current coordinates of axis to add or reduce position, and the target position and speed can be variable or address.
- **Absolute Mode (Variable or Address / Target position & Speed)**: Use absolute coordinates of axis as target position, and the target position and speed can be variable or address.
- **Relative Mode (Variable or Address / Target position & Speed & Acceleration and Deceleration Time)**: Use the current coordinates of axis to add or reduce position, and the target position, speed, acceleration time, and deceleration time can be variable or address.
- **Absolute Mode (Variable or Address / Target position & Speed & Acceleration and Deceleration Time)**: Use absolute coordinates of axis as target position, and the target position, speed, acceleration time, and deceleration time can be variable or address.

- **Target Position**: Type a target coordinate. In the **Position Plan Table**, the **ABS/REL Mode** includes **Relative Position (Variable/Address)** or **Absolute Position (Variable/Address)** that requires the use of **D** register or global variables to input the target position. When using a **D** register, the data takes up to the length of two **D** registers.

  Example: If you enter D100, it will take up the length of D100 and D101. When using global variables, the address must be a variable of D register in DWORD or DINT format which will also take up the length of two D registers.

- **Target Speed (Hz)**: Type a target speed in this field.

- **Bias Speed (Hz)**: The speed when the positioning starts and ends.
- **Acceleration Time (ms)**: The time when the **Target Speed** is reached.
- **Deceleration Time (ms)**: The time when the stop speed is reached.
- **Completion Action**: Selecting **Continue** means to move to the execution of the next row of the table after the present table execution is completed. Selecting **End** means to end the action after the present table execution is completed.
- **Completion Flag (I/M)**: The register is set to **ON** when the present table execution is completed. And, I number and or the name of M register can be typed only when **End** in **Completion Action** column is selected. For the interrupt numbers supported, refer to the application manual. The ten I numbers: - I510 ~ I519 are supported.
- **Next Step**: Type the number of the next segment of **Position Plan Table**. Blank means completion. The setting is automatically effective after **End** in **Completion Action** column is selected. **Next Step** is not restricted to the same axis number.

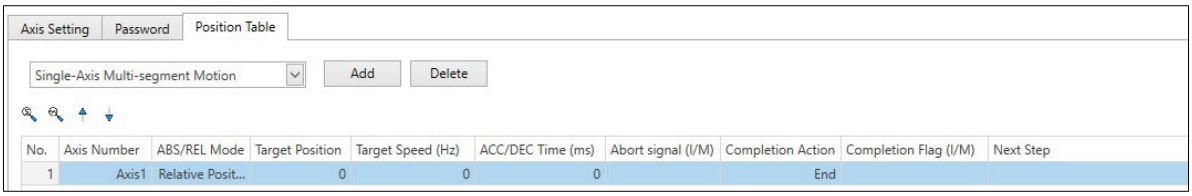
  Remark: The same axis number could not have more than two table numbers or position output instructions which are output together.

Note:

- The **Target Speed (Hz)** cannot be set to 0. The **Target Position** also cannot be set to 0 in the **Relative Mode**.
- I number can be used more than once. You must determine through which row number the I number can be completely output.
- The instruction which the start table outputs is API2718 TPO. API2719 TPWS can be used to modify the output parameters while the PLC is running.

10.1.8.2 Single-Axis Multi-segment Motion

In **Single-Axis Multi-segment Motion** control mode, you can control the positioning of one axis and make every-segment motion followed by the next-segment motion at the target speed. Or abort the segment and to perform the next-segment motion. The following figure shows an example of **Single-Axis Multi-segment Motion** control.



- **No.**: The row number created in order automatically. Maximum number: **200**.
- **Axis Number**: Select the axis which is controlled by the **Position Plan Table**.
- **ABS/REL Mode**:
  - **Relative Mode**: Use the current coordinates of axis to add or reduce position, values are required for each field.
  - **Absolute Mode**: Use absolute coordinate of axis as target position, values are required for each field.
  - **Relative Mode (Variable or Address)**: Use the current coordinates of axis to add or reduce position, and the target position can be variable or address.
  - **Absolute Mode (Variable or Address)**: Use absolute coordinate of axis as target position, and the target position can be variable or address.
  - **Relative Mode (Variable or Address / Target position & Speed)**: Use the current coordinates of axis to add or reduce position, and the target position and speed can be variable or address.
  - **Absolute Mode (Variable or Address / Target position & Speed)**: Use absolute coordinates as target position, and the target position and speed can be variable or address.
  - **Relative Mode (Variable or Address / Target position & Speed & Acceleration and Deceleration Time)**: Use the current coordinates of axis to add or reduce position, and the target position, speed, acceleration time, and deceleration time can be variable or address.
  - **Absolute Mode (Variable or Address / Target position & Speed & Acceleration and Deceleration Time)**: Use absolute coordinates as target position, and the target position, speed, acceleration time, and deceleration time can be variable or address.
- **Target position**: Users can type a target coordinate. In the Position Plan Table,

the **ABS/REL Mode** includes **Relative Position (Variable/Address)** or **Absolute Position (Variable/Address)** that requires the use of **D** register or global variables to input the target position. When using a **D** register, the data takes up to the length of two **D** registers.

Example: If you enter D100, it will take up the length of D100 and D101. When using global variables, the address must be a variable of D register in DWORD or DINT format which will also take up the length of two D registers!

- **Target Speed (Hz)**: Type a target speed in this field.
- **ACC/DEC Time (ms)**: The time when the **Target Speed** is reached and stop speed is reached.
- **Abort Signal (I/M)**: The next segment of Position Plan Table is performed if the register of interrupt **signals** is set to **ON** before the current table is completed. And, I number or the name of M register can be typed. For the interrupt numbers supported, refer to the product programming manual.
- **Completion Action**: Selecting **Continue** means to move to the execution of the next row of the table after the present table execution is completed. Selecting **End** means to end the action after the present table execution is completed.
- **Completion Flag (I/M)**: The **register** is set to **ON** when the present table execution is completed. And, I number and or the name of M register can be typed only when **End** in **Completion Action** column is selected. For the interrupt numbers supported, refer to the application manual. The ten I numbers: - I510 ~ I519 are supported.
- **Next Step**: Type the number of the next segment of Position Plan Table. Blank means completion. The setting is automatically effective after **End** in **Completion Action** column is selected. **Next Step** is not restricted to the same axis number.

  Remark: The same axis number could not have more than two table numbers or position output instructions which are output together.

Note:

- The **Target Position** cannot be set to 0 in the relative mode. The **Target Position** of two adjacent row numbers cannot be set to the same value in the **Absolute Position** mode.

- If the value in the **ACC/DEC Time** is not 0, the acceleration/deceleration is fixed to 1ms. If the value in the **ACC/DEC Time** is set to 0, it means there is no acceleration/deceleration.
- I number can be used more than once. You must determine through which row number the I number can be completely output.
- The instruction which the start table outputs is API2718 TPO. API 2719 TPWS can be used to modify the output parameters while the PLC is running.

10.1.8.3 2-axis Linear Interpolation Motion

Set the linear interpolation motion of two axes in the XY plane. Two rows of the **Position Plan Table** are occupied when this mode is used. The following figure shows **2-axis Linear Interpolation Motion** control.



- **No.**: The row number is automatically created orderly in the input area. It will take up two rows of the table for 2-axis motion mode. In other words, when in 2-axis mode, there are a total of 100 groups in the table and the maximum number: **100**.
- **Coordinate**: The axis which X and Y axes coordinate stand for in the simulation view.
- **Axis Number**: Select the axis which is controlled by the **Position Plan Table**.
- **ABS/REL Mode**:
  - **Relative Mode**: Use the current coordinates of axis to add or reduce position, values are required for each field.
  - **Absolute Mode**: Use absolute coordinate as target position, values are required for each field.
  - **Relative Mode (Variable or Address / Target position)**: Use the current coordinates of axis to add or reduce position, and the target position can be variable or address.
  - **Absolute Mode (Variable or Address / Target position)**: Use absolute

coordinate as target position, and the target position can be variable or address.

- **Relative Mode (Variable or Address / Target position & Speed)**: Use the current coordinates of axis to add or reduce position, and the target position and speed can be variable or address.
- **Absolute Mode (Variable or Address / Target position & Speed)**: Use absolute coordinate as target position, and the target position and speed can be variable or address.
- **Relative Mode (Variable or Address / Target position & Speed & Acceleration and Deceleration Time)**: Use the current coordinates of axis to add or reduce position, and the target position, speed, acceleration time, and deceleration time can be variable or address.
- **Absolute Mode (Variable or Address / Target position & Speed & Acceleration and Deceleration Time)**: Use absolute coordinate as target position, and the target position, speed, acceleration time, and deceleration time can be variable or address.

- **Target Position**: Type a target coordinate. In the Position Plan Table, the **ABS/REL Mode** includes **Relative Position (Variable/Address)** or **Absolute Position (Variable/Address)** that requires the use of D register or global variables to input the target position. When using a D register, the data takes up to the length of two D registers.

  Example: If you enter D100, it will take up the length of D100 and D101. When using global variables, the address must be a variable of D register in DWORD or DINT format which will also take up the length of two D registers!

- **Target Speed (Hz)**: Type a target speed in this field.
- **ACC/DEC Time (ms)**: The time when the **Target Speed** is reached and Stop **Speed** is reached.
- **Completion Action**: Selecting **Continue** means to move to the execution of the next row of the table after the present table execution is completed. Selecting **End** means to end the action after the present table execution is completed.
- **Completion Flag (I/M)**: The register is set to **ON** when the present table

execution is completed. And, I number and or the name of M register can be typed only when **End** in **Completion Action** column is selected. For the interrupt numbers supported, refer to the application manual. The ten I numbers: - I510 ~ I519 are supported.

- **Next Step**: Type the number of the next segment of Position Plan Table. Blank means completion. The setting is automatically effective after **End** in **Completion Action** column is selected. **Next Step** is not restricted to the same axis number.

  Remark: The same axis number could not have more than two table numbers or position output instructions which are output together.

Note:

- The PLC will automatically run at the set speed when the **Target Speed** or the **ACC/DEC Time** is set to 0. The **Target Position** cannot be set to 0 in the **Relative Position** mode. The **Target Position** of two adjacent row numbers cannot be set to the same value in the **Absolute Position** mode.
- I number can be used more than once. You must determine through which row number the I number can be completely output.
- The instruction which the start table outputs is API2718 TOP. API2720 TPWL can be used to modify the output parameters while the PLC is running.

10.1.8.4 2-axis Arc Interpolation Motion

Set the Arc Interpolation Motion of two axes on the XY plane. Two rows of Position Plan Table are occupied by using the **2-axis Arc Interpolation Motion** control mode every time. The following figure shows a sample **2-axis Arc Interpolation Motion** control mode.



- **No.**: The row number is automatically created orderly in the input area. It will take

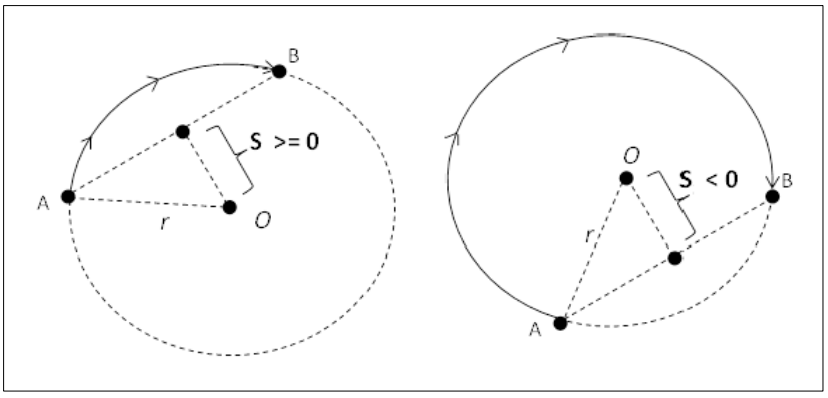
up two rows of the table for 2-axis motion mode. In other words, when in 2-axis mode, there are a total of 100 groups in the table and the maximum number is **100**.

- **Coordinate**: The axis which X and Y axes coordinate stand for in the simulation view.
- **Axis Number**: Select the axis which is controlled by the **Position Plan Table**.
- **ABS/REL Mode**:
  - **Relative Mode**: Use the current coordinates of axis to add or reduce position, values are required for each field.
  - **Absolute Mode**: Use absolute coordinate of axis as target position, values are required for each field.
  - **Relative Mode (Variable or Address)**: Use the current coordinates of axis to add or reduce position, and the target position can be variable or address.
  - **Absolute Mode (Variable or Address)**: Use absolute coordinate of axis as target position, and the target position can be variable or address.
  - **Relative Mode (Variable or Address / Target position & Speed)**: Use the current coordinates of axis to add or reduce position, and the target position and speed can be variable or address.
  - **Absolute Mode (Variable or Address / Target position & Speed)**: Use absolute coordinate of axis as target position, and the target position and speed can be variable or address.
  - **Relative Mode (Variable or Address / Target position & Speed & Acceleration and Deceleration Time**: Use the current coordinates of axis to add or reduce position, and the target position, speed, acceleration time, and deceleration time can be variable or address.
  - **Absolute Mode (Variable or Address / Target position & Speed & Acceleration and Deceleration Time**: Use absolute coordinate of axis as target position, and the target position, speed, acceleration time, and deceleration time can be variable or address.
- **Target Position**: Type a target coordinate. In the Position Plan Table, the

ABS/REL Mode includes **Relative Position (Variable/Address)** or **Absolute Position (Variable/Address)** that requires the use of D register or global variables to input the target position. When using a D register, the data takes up to the length of two D registers.

Example: If you enter D100, it will take up the length of D100 and D101. When using global variables, the address has to be a variable of D register in DWORD or DINT format which will also take up the length of two D registers!

- **Tangential Speed (Hz)**: **Movement** in a circular path in three-dimensions and the magnitude of that vector is the tangential speed.
- **Center Offset Value (Pulse)**: The **distance** from the center of the line between the arc target point and start point to the center of a circle, such as S in the following figure. From point A to point B, a clockwise rotation will generate an arc less than 180^0 if S>0 and an arc more than 180^0 is S<0.



- **Direction**: Select **Clockwise** motion or **Counterclockwise** motion.
- **Completion Action**: Selecting **Continue** means to move to the execution of the next row of **the** table after the present table execution is completed. Selecting **End** means to end the action after the present table execution is completed.

- **Completion Flag (I/M)**: The **register** is set to **ON** when the present table execution is completed. And, I number and or the name of M register can be typed only when **End** in **Completion Action** column is selected. For the interrupt numbers supported, refer to the application manual. The ten I numbers: -I510~I519 are supported.
- **Next Step**: You can type the number of the next segment of **Position Plan Table**. Blank means completion. The setting is automatically effective after **End** in **Completion Action** column is selected. **Next Step** is not restricted to the same axis number.

  Remark: The same axis number could not have more than two table numbers or position output instructions which are output together.

Note:

- The PLC automatically calculates the output speed in this function. The **Target Position** cannot be set to 0 in the **Relative Position** mode. The **Target Position** of two adjacent row numbers cannot be set to the same value in the **Absolute Position** mode.
- I number can be used more than once. You must determine through which row number the I number can be c completely output.
- The instruction which the start table outputs is API2718 TPO. API2721 TPWL can be used to modify the output parameters while the PLC is running.

10.2 Axis Unit Conversion

10.2.1 About Axis Unit Conversion

Axis provides an interface from motion control to an actual output device. You can control the axes through multiple motion function blocks, and output the parameter converted from the axis unit conversion function to the target servo drive to achieve motion control.

10.2.2 Axis Unit Conversion Setting

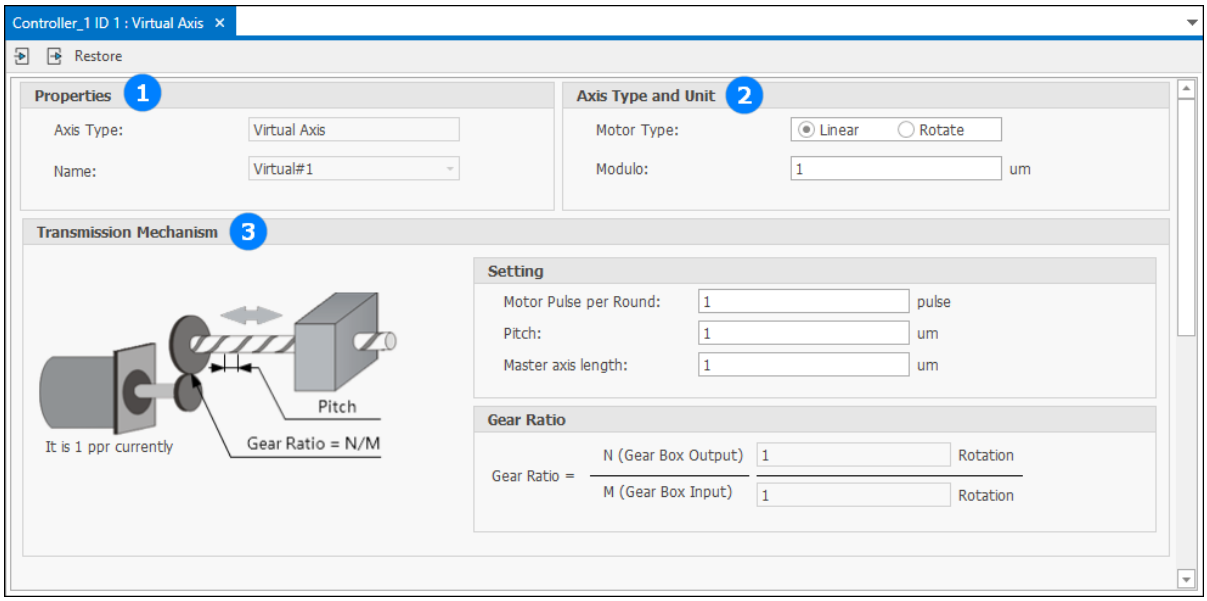
Set the parameter for axes through the axis unit conversion tool.

In **Project Explorer**, double-click **Controller** > **Motion** > **Axis Unit Conversion**. The axes supported by the model will be listed. The types include Virtual Axis, Encoder Axis, and Slave Axis.

Note: Axis unit conversion needs to be used with E-CAM and motion function block. For details about the usage, refer to the product user manual.

The following table lists the settings in each axis parameter page and their descriptions.

Virtual Axis



1 Properties

Function	Description
Axis Type	Shows the axis type.
Name	Shows the axis name.

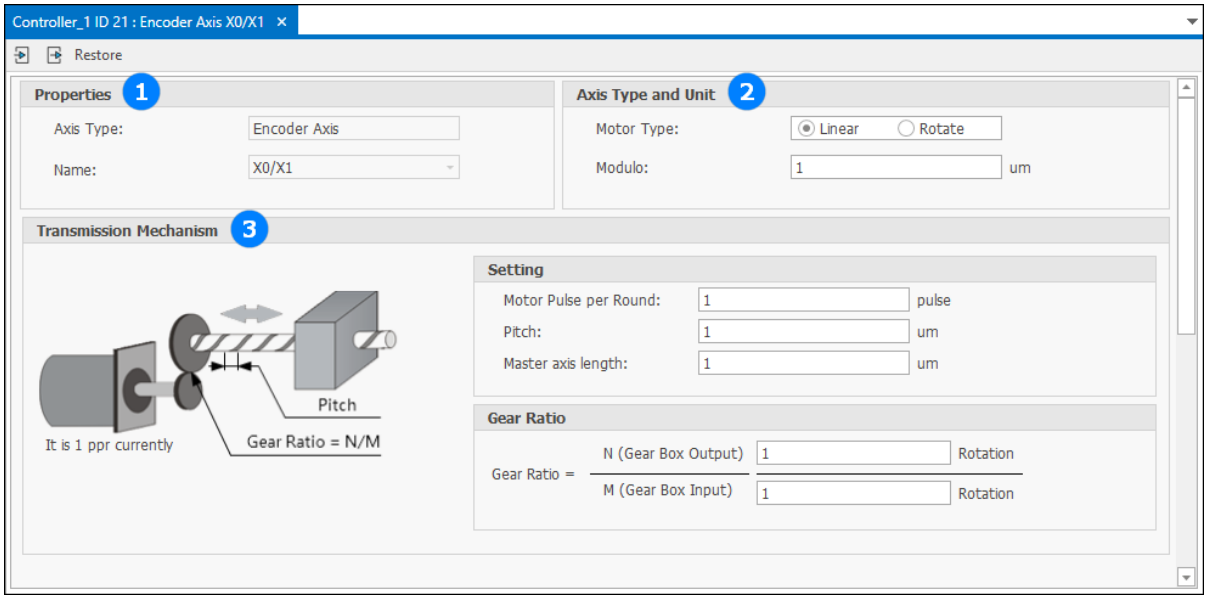
2 Axis Type and Unit

Function	Description
Motor Type	Select the linear or rotate type.
Modulo	Enter a circle range. The default value is **1**.

③ Transmission Mechanism

Function	Description	Value
Setting	Set the parameter value for the selected device. The settings differ by type.	The default value is **1**.

Encoder Axis



① Properties

Function	Description
Axis Type	Shows the axis type.
Name	Shows the axis name.

② Axis Type and Unit

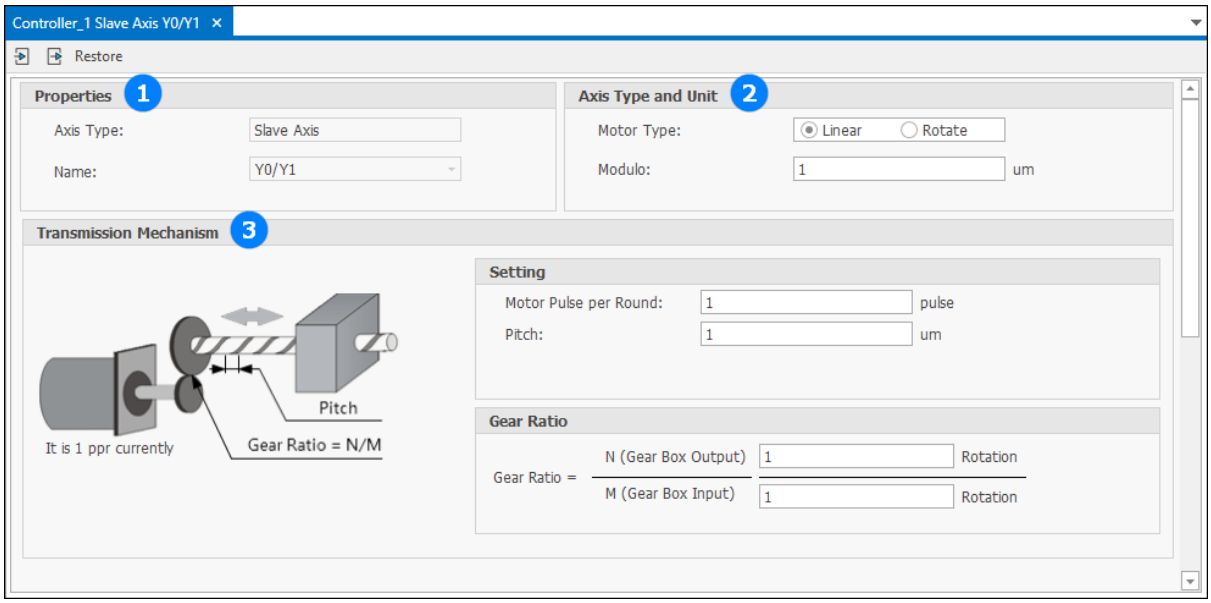
Function	Description
Motor Type	Select the linear or rotate type.
Modulo	Enter a circle range. The default value is **1**.

③ Transmission Mechanism

Function	Description	Value

Setting	Set the parameter value for the selected device. The settings differ by type.	The default value is **1**.
Gear Ratio	Enter the gear ratio's numerator and denominator.	The default value is **1**.

Slave Axis



1 Properties

Function	Description
Axis Type	Shows the axis type.
Name	Shows the axis name.

2 Axis Type and Unit

Function	Description
Motor Type	Select the linear or rotate type.
Modulo	Enter a circle range. The default value is **1**.

③ Transmission Mechanism

Function	Description	Value
Setting	Set the parameter value for the selected device. The settings differ by type.	The default value is **1**.
Gear Ratio	Enter the gear ratio's numerator and denominator.	The default value is **1**.

10.3 E-CAM (Electronic CAM)

10.3.1 About E-CAM

A mechanical cam is an input object with irregular shape. It makes a follower move regularly by coming into contact with the follower. A traditional mechanical cam is composed of a cam, a follower, and a support. When a cam rotates, the follower will go move according to the shape of the cam. For an operation of motion control, the relative movements between axes are achieved by the rotation of a cam.

Use a cam chart to define the relation between a follower and a cam to simulate the movements of a cam controlled by a controller, making a slave axis to move by the master axis according to their defined relation. To modify the cam, you only need to modify the electronic cam data in the software instead of modifying the mechanical design.

10.3.2 Using E-CAM

Follow these steps to create an E-CAM chart.

To create an E-CAM chart

1. In **Project Explorer**, add a controller device.

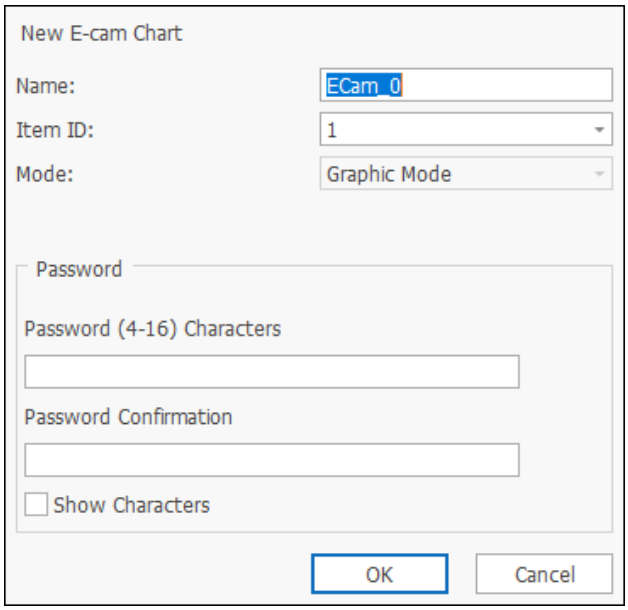
   Note: Currently applicable to the DVP-ES3, C2000 Plus, and CH2000 models.

2. Double-click **Controller** > **Motion**.
3. Right-click **E-CAM** and select **Add**.

4. In the **New E-cam Chart** dialog, configure the settings.

Note: E-CAM needs to be used with axis unit conversion and motion function block. For more details about the usage, refer to the product user manual.

The following table describes the functions of **New E-cam Chart** dialog.



Function	Description
Name	Enter the name of the E-CAM chart. **Note**: This setting varies from the device model.
Item ID	Select the ID of the E-CAM chart. **Note**: This setting varies from the device model.
Mode	Select the mode. You can refer to the product user manual for mode type. **Note**: This setting varies from the device model.
Password	Enter the password to protect the E-CAM data according to your needs. **Note**: This setting varies from the device model.

The following table describes the functions of E-CAM chart.

Toolbar

Icon	Description
	Click to import the CAM data.
	Click to export the CAM data.

Icon	Description
	Click to set the property of CAM. You can also right-click the created E-CAM chart in Project Explorer and select **Properties**.
	Click to configure the security settings.
	Click to collapse / expand the Displacement graph.
	Click to collapse / expand the Velocity graph.
	Click to collapse / expand the Acceleration graph.
	Click to show the maximum / minimum value in the graph.

Data Setting Area

Icon	Description
	Click to import the CAM data.
	Click to export the CAM data.

The data of the data setting area is synchronized with the E-CAM curve area.

E-CAM Curve Area

Icon	Description
	Click to switch the selection tool.
	Click to insert a control point.
	Click to delete a control point.
	Click to insert a line segment including two control points.

The corresponding E-CAM position, velocity, and acceleration are displayed according to the parameter set on the data setting area. You can hold the left mouse button and drag the control point to change parameter.

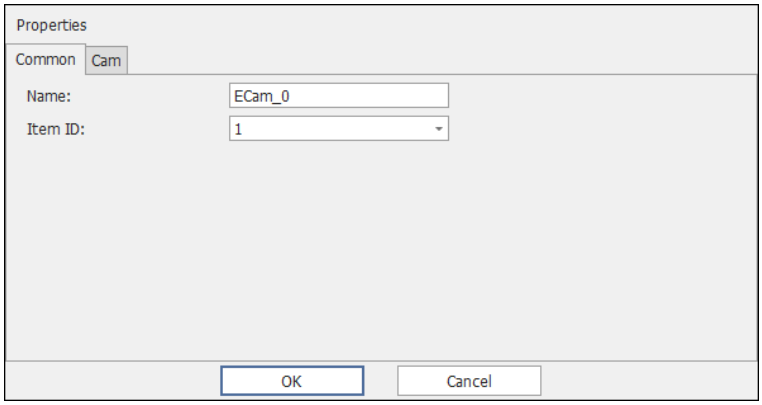
On the graph, you can do the following.

- Right-click to insert a control point, line segment, or set the property.

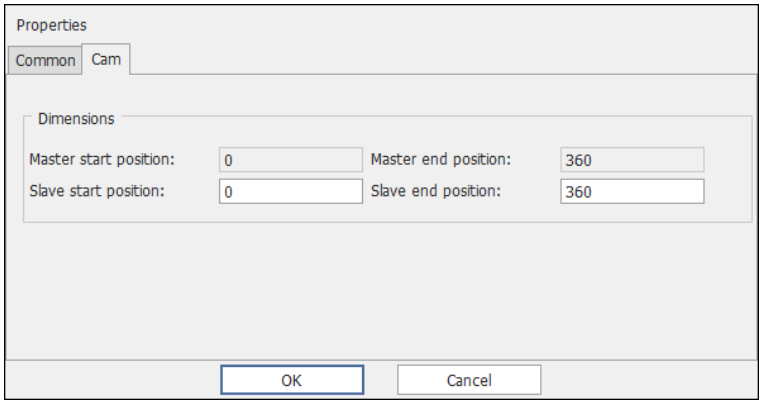
- Right-click the control point to delete the point or set it as a line.

The following table describes the functions of **Properties** dialog.

Common



Function	Description
Name	Enter the name of the CAM chart.
Item ID	Select the ID of the CAM chart.



Function	Description
Master start position	The minimum value displayed on the master axis in the graph. The value cannot be set and is fixed as 0.
Master end position	The maximum value displayed on the master axis in the graph. The value cannot be set and is fixed as 360.
Slave start position	Set the minimum value displayed on the slave axis in the graph. The default value is **0**.
Slave end position	Set the maximum value displayed on the slave axis in the graph. The default value is **360**.

Chapter 11: HMI Interaction

11.1 Tag Sharing to DIAScreen

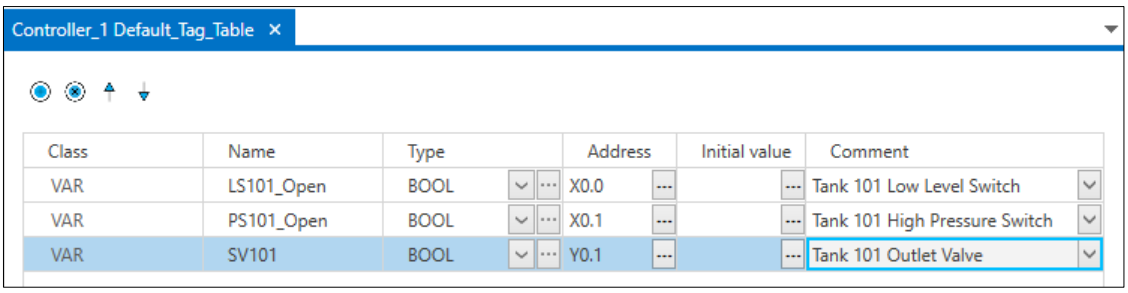
Tag means the variable that is inherited between DIADesigner & DIAScreen. You can share the Variables with **Address** created in DIADesigner with DIAScreen application, so that you can directly access those Variables in DIAScreen via **DIA Tag** function.

Note: Only **DOP-100 HMI series** supports **DIA Tag** sharing feature in the current version.

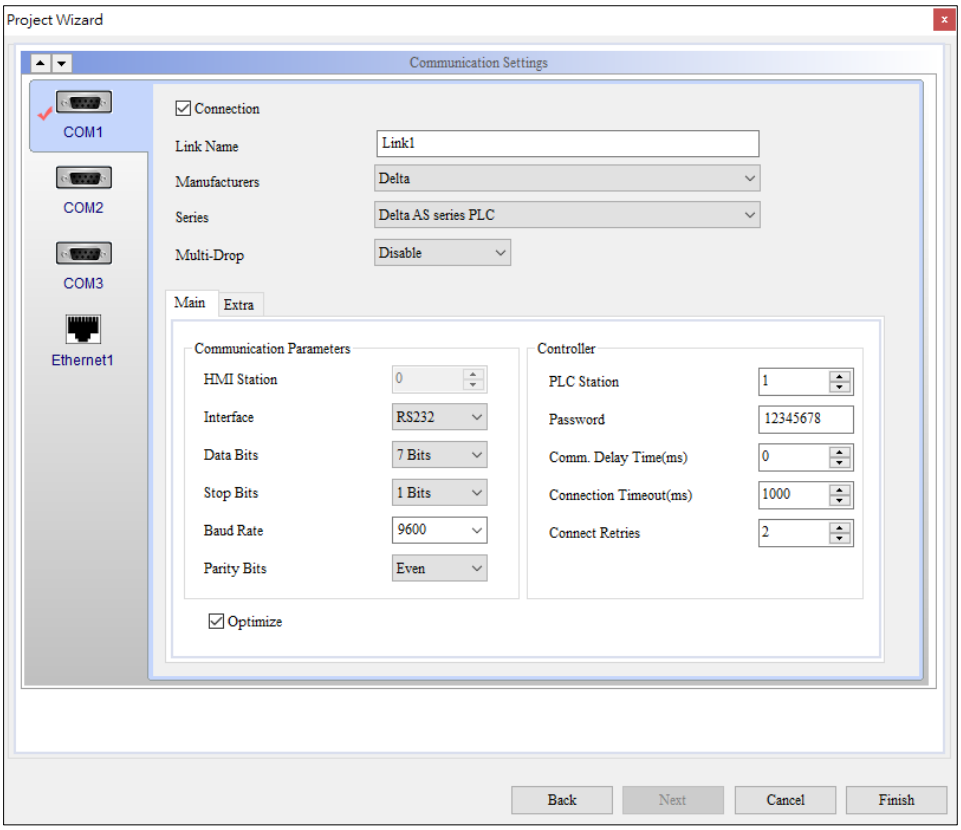
Follow these steps to perform tag sharing to DIAScreen.

To perform tag sharing to DIAScreen

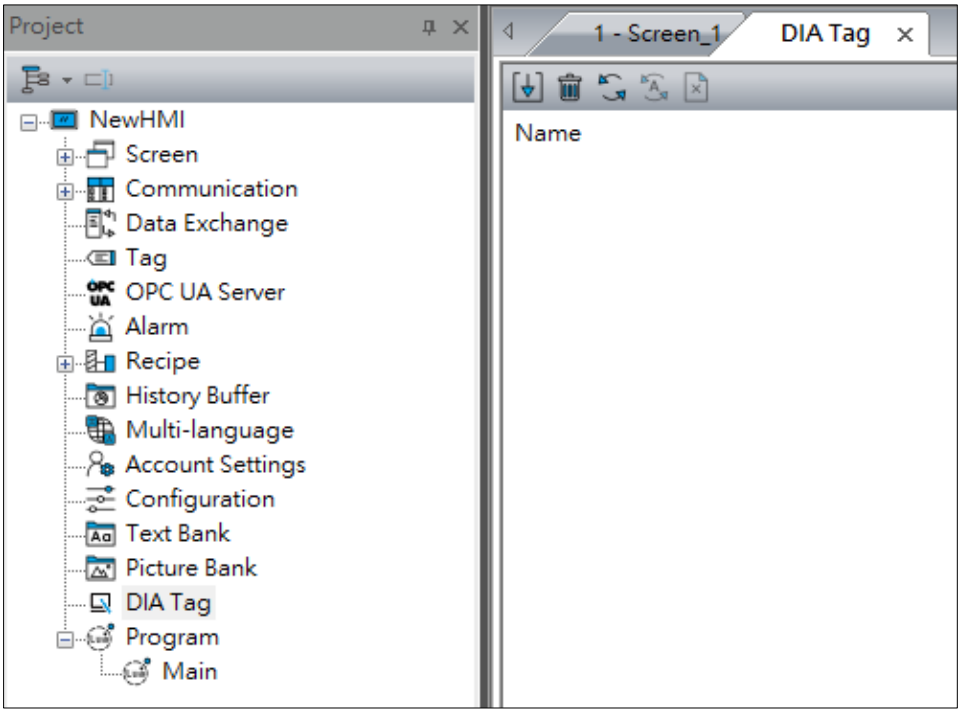
1. Create variables with address in any global variable table of **Controller** > **Programming** > **Global Variable** > **Default_Tag_Table**.



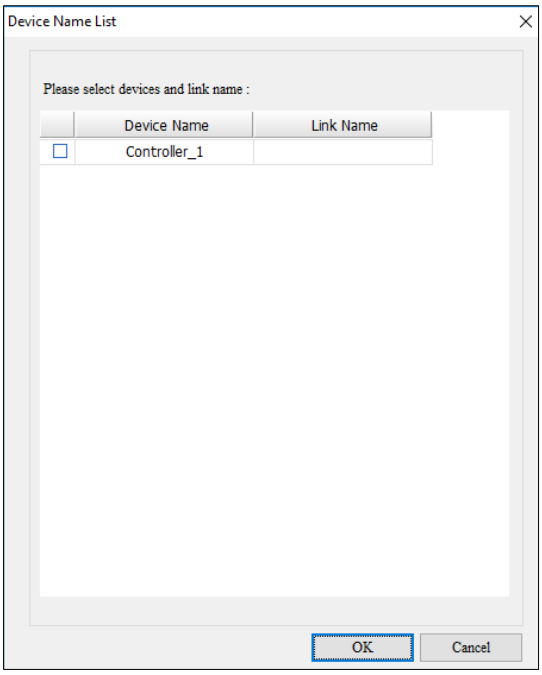
2. Compile the DIADesigner device or project.
3. Launch DIAScreen.
4. Create a DIAScreen project with DOP-100 HMI series. During project creation, enter a **Link Name**, select **Delta** in **Manufacturers** drop-down list, select **Delta AS series PLC** in the **series** drop-down list as shown in the following figure.



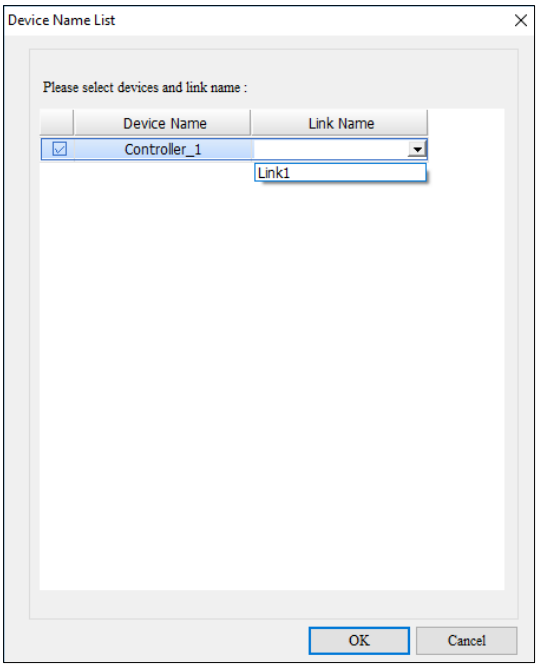
5. Double-click **DIA Tag** function in the **Project Explorer** in DIAScreen to open **DIA Tag** dialog.



6. Click  in the **DIA Tag** dialog to open **Device Name List** dialog.

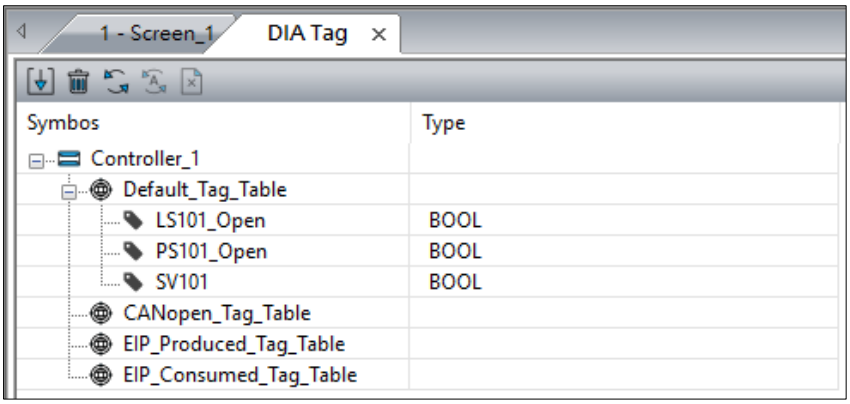


7. Select controllers in the **Device Name** and select **Link Name.**

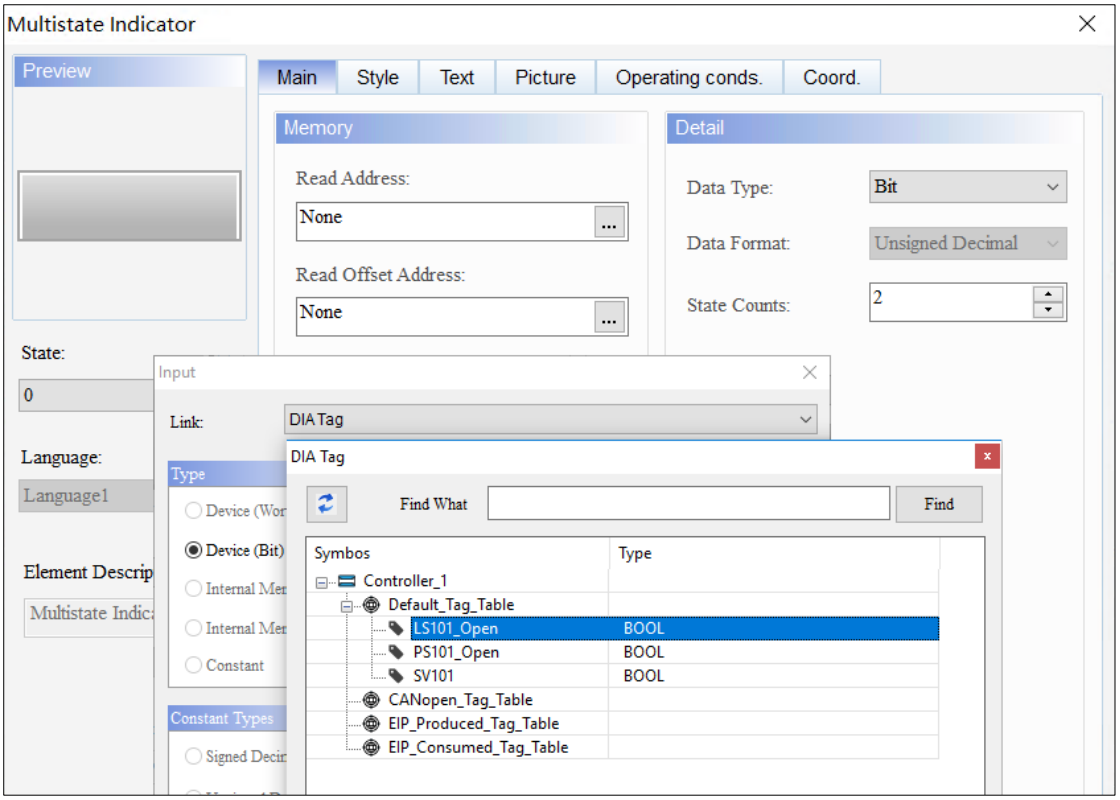


8. Click **OK**.

DIADesigner tags are now shared with DIAScreen so that the Variables can be used for different kinds of elements, animations, etc. as shown in the following figure.



Example: Select **DIA Tag** function to select the DIADesigner variables as shown in the following figure.



Chapter 12: Simulation

12.1 PLC Simulation

In the current version of DIADesigner, AS100, AS200, and AS300 PLC Simulation is supported.

Note: Refer to Driver Setting for more information.

Follow these steps to perform PLC Simulation for AS100 / AS200 / AS300 PLCs.

To perform PLC Simulation

1. Create a Communication Driver of AS100 Simulator / AS200 Simulator / AS300 PLC Simulator in **COMMGR** and **Start** the driver.
2. Add AS100 / AS200 / AS300 devices as per your requirements in DIADesigner.
3. Create programs, variables and so on, as per your requirement.
4. Compile the device.

   Note: Refer to Compile and Check for more information.
5. Double-click **Controller** > **Communication Setting** in **Project Explorer** of DIADesigner to open **Communication Setting** dialog.

   Note: Refer to Communication via COMMGR for more information.
6. Select Simulation Driver created in Step (1) in the **Driver Name** field in **Communication Setting** dialog.
7. Click **Connect**.

   The **Connected** status displays in the **Communication Setting** dialog.
8. Perform **Download** operation.

   Note: Refer to Batch Download and Upload for more information.
9. Go **Online** with the device.

   Open the POU or Monitor Table or other options and perform the monitoring, etc.

   Note: Refer to Download / Upload and Online / Offline Mode for upload / download and online / offline functions.

Chapter 13: Download / Upload and Online / Offline Mode

13.1 Batch Download and Upload

Download / upload the programs, functions or parameters in projects from / to multiple devices simultaneously in DIADesigner.

13.1.1 Download

Follow these steps to download project from PC to devices.

To download project

1. Compile controller models or inverters with built-in controller. Other field devices do not need to be compiled.

   Note: Compile must be successful without any errors.

2. Select communication **Driver** in **Device** > **Communication Setting** dialog.

3. Select the device and click  on **Development** toolbar.

   Or

   Right-click the device to open its context menu and select **Download**.

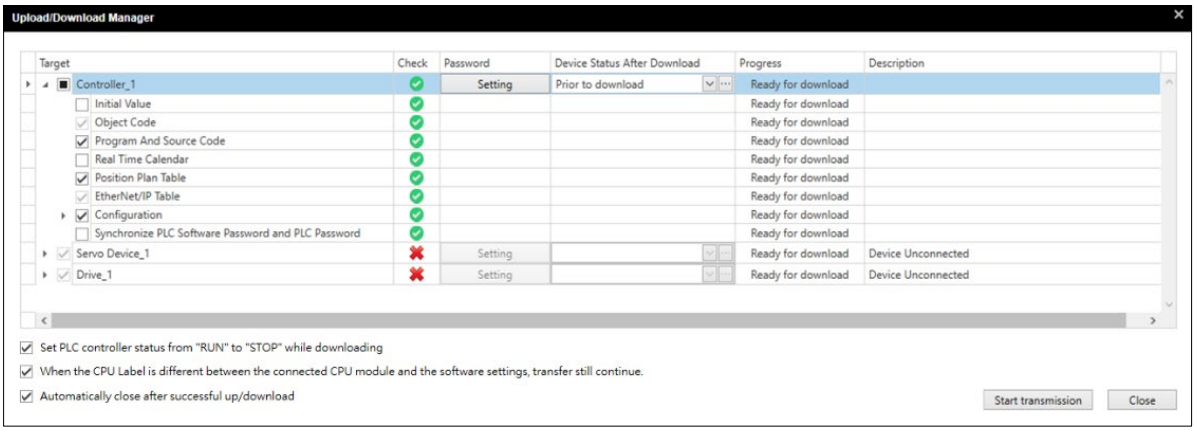
   Or

   Select the device and click **Operation** > **Project Download** in menu bar.

   Or

   Select the device and press **Ctrl** + **L** keys simultaneously on the keyboard.

   The **Upload/Download Manager** dialog appears as shown in the following figure.



The following table describes the functions in the **Download Manager** dialog.

Function	Description
Target	Display the target device name as defined in the project in the form or a tree node. Components (functions) display below the device name (depends on device).
Check	Display whether the download check is passed ( ) or failed ( ).
Password	Click **Setting** to display the **Password** dialog.
Device status after download	Select the device status intended after download action. The available options include: • Prior to Download • Run • Stop **Note**: The default value is **Prior to Download**.
Progress	Display the progress. **Example**: Ready for Download, % progress and so on.
Description	Display the description of the device connection status.
Set PLC controller status from “RUN” to “STOP” while downloading	If any of PLC is RUN mode, you can check this box to switch to STOP mode when download to those PLC.
When the CPU Label is different between the connected CPU module, and the Software settings, transfer is still continue	Check to continue download even if the PLC label is different.
Automatically close after successful upload/download	Check to close the **Upload/Download Manager** dialog after successful transformation.
Start transmission	Click to start upload / download.
Close	Click to close the **Upload/Download Manager** dialog.

4. Select the components to be downloaded.
5. Click **Start Transmission**.

   Project or components selected are downloaded to the device.

13.1.2 Upload

Upload the DIADesigner or ISPSoft project downloaded to the controller to the PC for further editing and operation.

Follow these steps to upload or transfer project from PLC to PC.

To upload or transfer project

1. Create a device or select the device whose program or component needs to be uploaded.
2. Select the Communication Driver in **Device** > **Communication Setting**.
3. Select the device and click  on **Development** toolbar

   Or

   Right-click the device to open its context menu and select **Upload**.

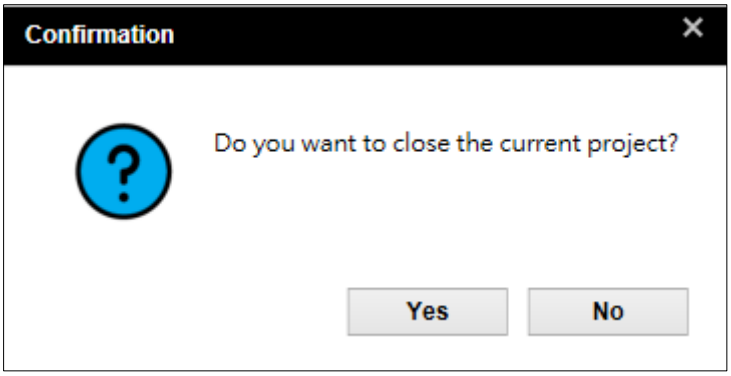
   Or

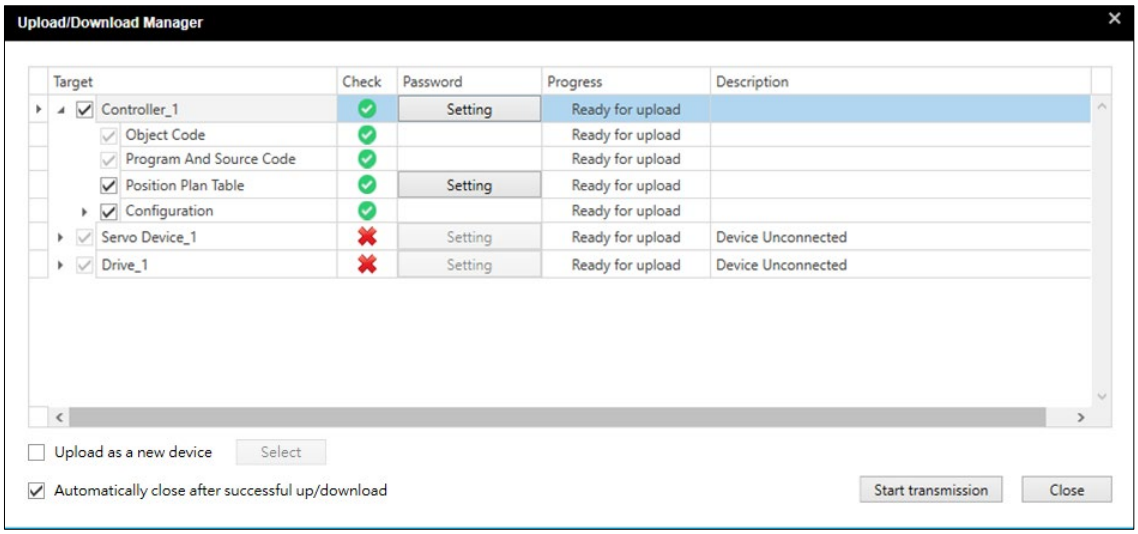
   Select the device and click **Operation** > **Project Upload** in menu bar.

   Or

   Select the device and press **Ctrl** + **U** keys simultaneously on the keyboard.

   Confirmation dialog appears. Click **Yes** to close current project, and upload data to a new project. Click **No** to upload data to current project.





The following table describes the functions in **Upload Manager** during **Upload**.

Function	Description
Target	Select the devices to be uploaded.
Check	Display whether the upload check is passed ( ) or failed ( ).
Password	Click **Setting** to display the **Password** dialog.
Progress	Display the progress. **Example**: Ready for Upload, % progress and so on.
Description	Display the description.
Upload as a new device	Check to add a new device to the project for uploading.
Automatically close after successful upload / download	Check it to automatically close **Upload/Download Manager** after uploading / downloading.
Start transmission	Click to start upload.
Cancel	Click to cancel upload.

4. Select the devices to be uploaded.
5. Click **Start Transmission**.

   Project or components selected are uploaded to the PC.

13.1.3 Download / Upload Components

The items in the **Upload / Download Manager** dialog vary according to the device or series selected. The items, which are not editable, are grayed out.

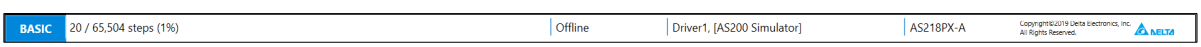
The following table describes transmission items in the **Upload / Download Manager**.

Item	Description
Initial Value	**Initial Value** of **Variables**.
Object Code	Object Code is what a PC compiler produces. **Note**: This item is selected by default.
Program and Source Code	IEC source code: the programming language specified by IEC61131-3. C source code: a program edited in C language
Real time Calendar	Real Time Calendar (RTC) of PC / device.
Position Plan Table	Position Plan Table created by users.
EtherNet / IP Table	EtherNet / IP Table.
Configuration	**Hardware Configuration** setting created by users.
Synchronize PLC Software Password and PLC Password	During the transmission of the project, the project password and the PLC password become the same.

13.2 Online / Offline Mode

Perform the addition, deletion, modification of the specified hardware, programs, variables, address assignment and so on, in the **Offline** mode and edit the project / device. After editing is complete, compile and generate the object code which is to be downloaded to PLC. At last, the programs and setting will be written to the target device by downloading.

When a controller or device is not connected to the DIADesigner, the status bar of the software displays **Offline** in its status area as shown in the following figure. By default, the operation status of DIADesigner is in **Offline** mode.



After connecting to devices, enter **Online** mode to check the status of devices.

Follow any of these methods to enter **Online** mode.

To enter online mode

- Click  on **Development** toolbar.

  Or

- Click **Operation** > **Online** in menu bar.

  Or

- Right-click device to open its context menu and click **Online**.

  Or

- Press **Ctrl** + **F4** keys simultaneously on the keyboard.

  DIADesigner enters **Online** mode for the device connected as shown in the following figure.

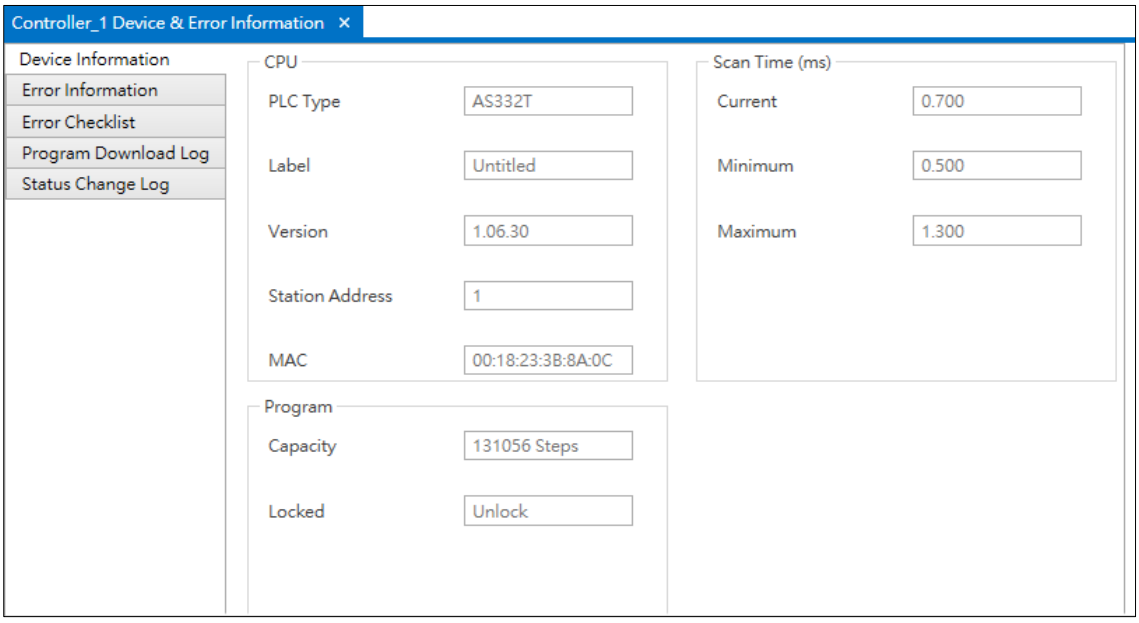


Chapter 14: Commission and Diagnostic

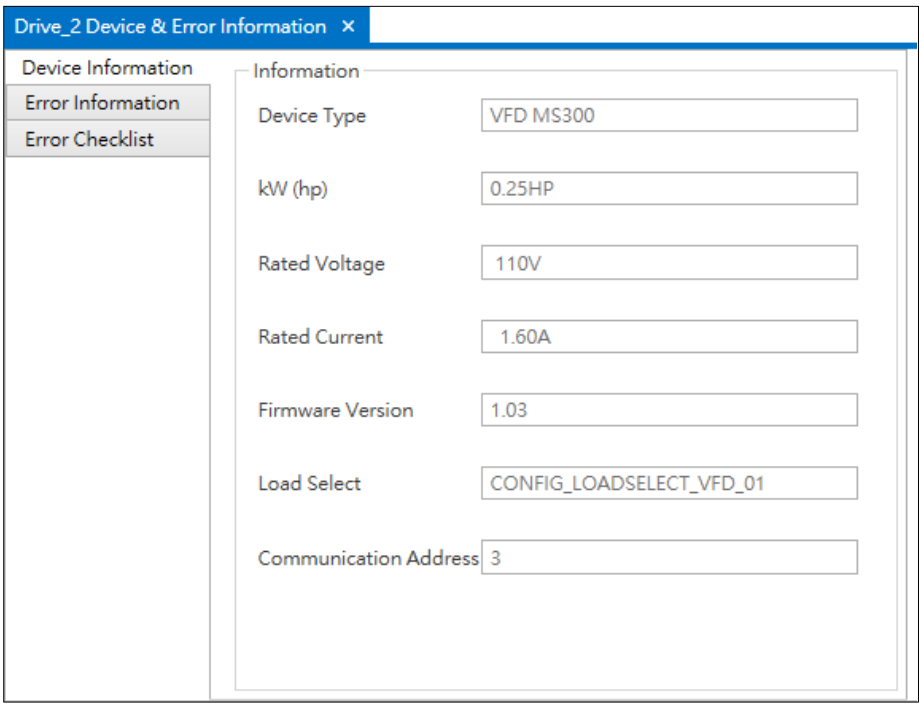
14.1 Device Information

If DIADesigner is connected to a device, the system retrieves information from the device after navigating to **Controller** > **Device & Error Information** > **Device Information** in the **Project Explorer**. The **Device Information** tab appears by default. System information is different for different product series.

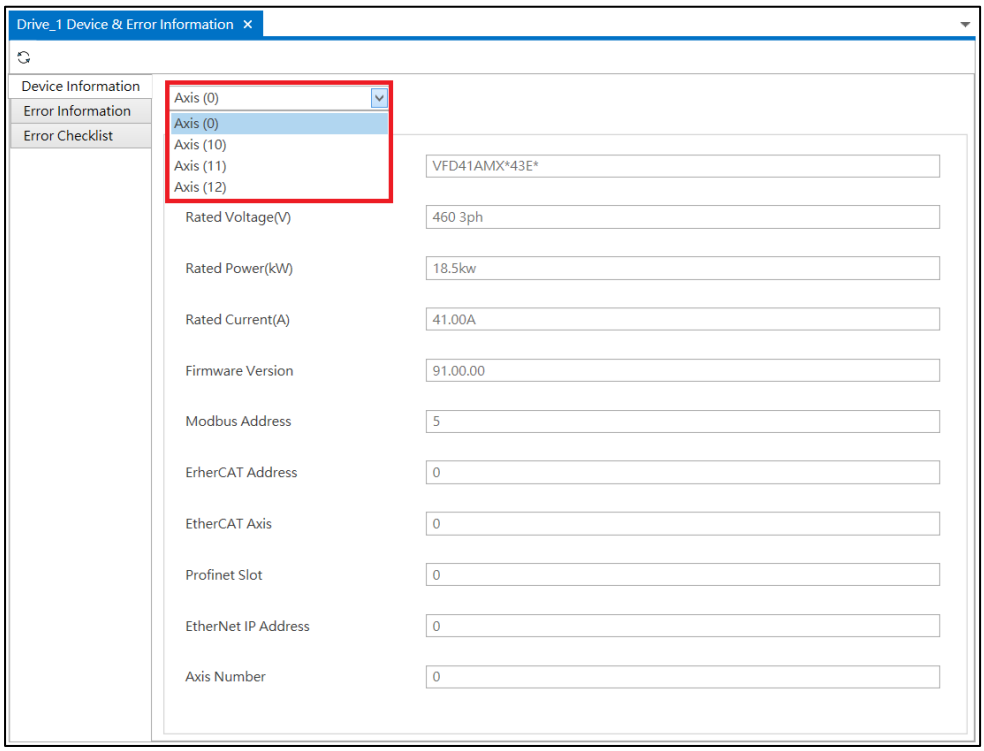
The following figure shows an example of inverter (AC Motor Drive).



The following figure shows an example of **Device Information** for MS300 inverter.



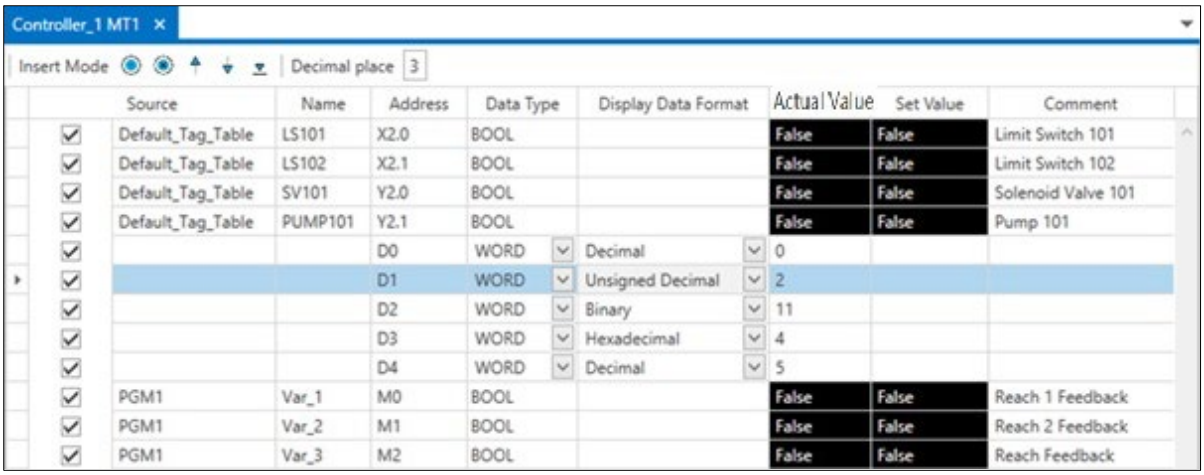
If you use a multi-drive inverter product, select the axis number you want to view from the drop-down menu at the top of the screen, and the corresponding axis information appears. The following is an example of the MX300 series inverter.



14.2 Monitor Table

Check the status and values of Variables – Global Variables or POU Local Variables and Registers during testing and commissioning with the help of **Monitor Table** in DIADesigner. Locate **Monitor Table** in DIADesigner in **Controller** > **Commissioning** > **Monitor Table**.

The following figure shows a sample **Monitor Table**.



The following table describes the functions in the **Monitor Table**.

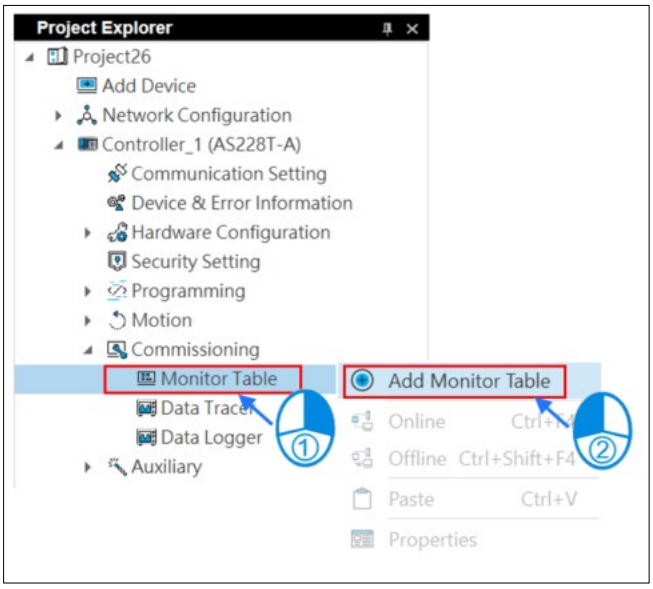
Function	Description
Source	Display the source of the variable.
Name	Display the name of the variable.
Address	Display the address of the variable/register.
Data Type	Display the datatype of the variable/register.
Display Data Format	Select a format in which the register value is represented. The available options include: • Decimal • Unsigned Decimal • Binary • Hexadecimal • Floating point • Scientific symbols • ASCII **Note**: The default value is **Decimal**.
Actual Value	Display the value of the variable / register in Online mode.
Set Value	Enter the value to be written to the variable in Online mode.
Comment	Display the comment of the variables / address.

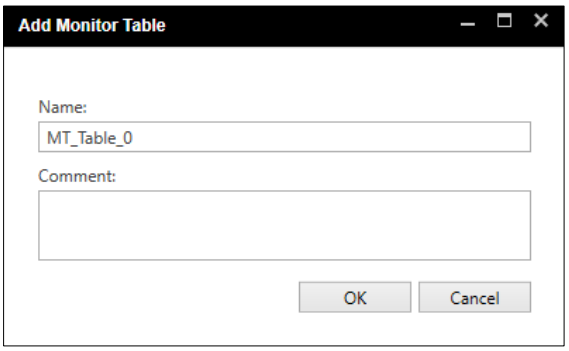
14.2.1 Creating a Monitor Table

Follow these steps to create a Monitor Table.

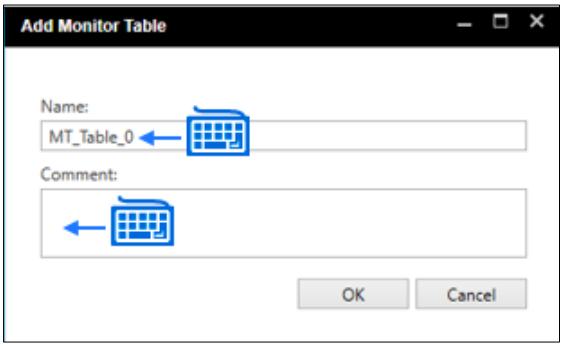
To create Monitor Table

1. Click **Controller** > **Commissioning** > **Monitor Table**.
2. Right-click **Monitor Table** to open its context menu and select **Add Monitor Table**.

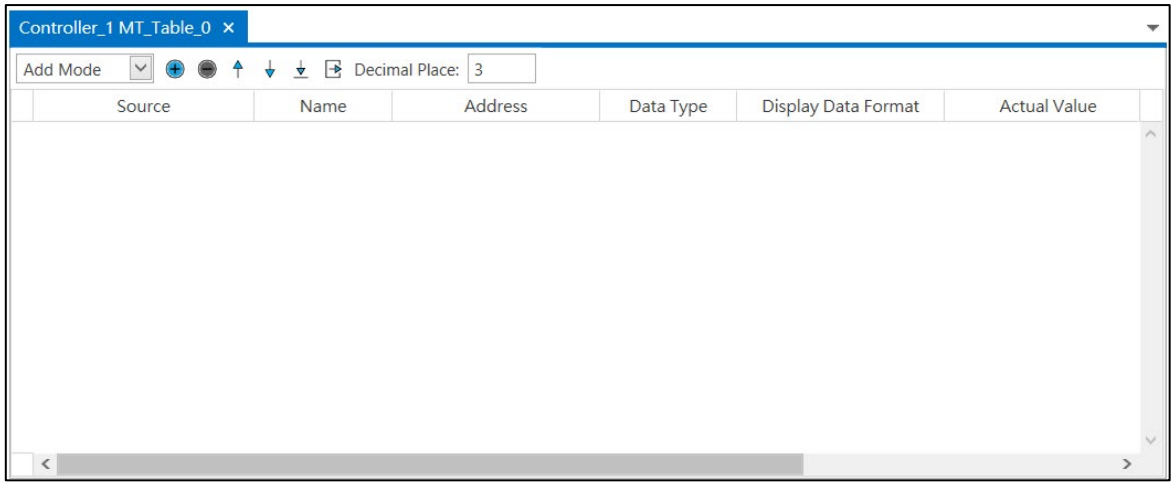




3. Enter name in the **Name** field and a comment for this **Monitor Table** in the **Comment** field (optional).



4. Click **OK**.



Note: When the **Monitor Table** is created, by default it displays in the DIADesigner work area. If the dialog is closed, double-click **Monitor Table** in **Project Explorer** to open **Monitor Table** dialog. You can create several Monitor Tables in the project.

The following table describes the functions in the context menu of **Monitor Table** in **Project Explorer**.

Function	Description
Add Monitor Table	Click to open **Add Monitor Table** dialog to add a new monitor table.
Online	Set the device in online mode.
Offline	Set the device in offline mode.
Paste	Paste the monitor table.
Properties	Click to open **Properties** dialog.

The following table shows the detailed descriptions of the context menu of each monitor table node.

Function	Description
Cut	Cut the selected monitor table.
Copy	Copy the selected monitor table.
Paste	Paste copied/cut monitor table.
Delete	Delete monitor table.
Online	Set the device in online mode.

Function	Description
Offline	Set the device in offline mode.
Properties	Click to rename the monitor table and edit comments.

The following figure shows toolbar of **Monitor Table**.

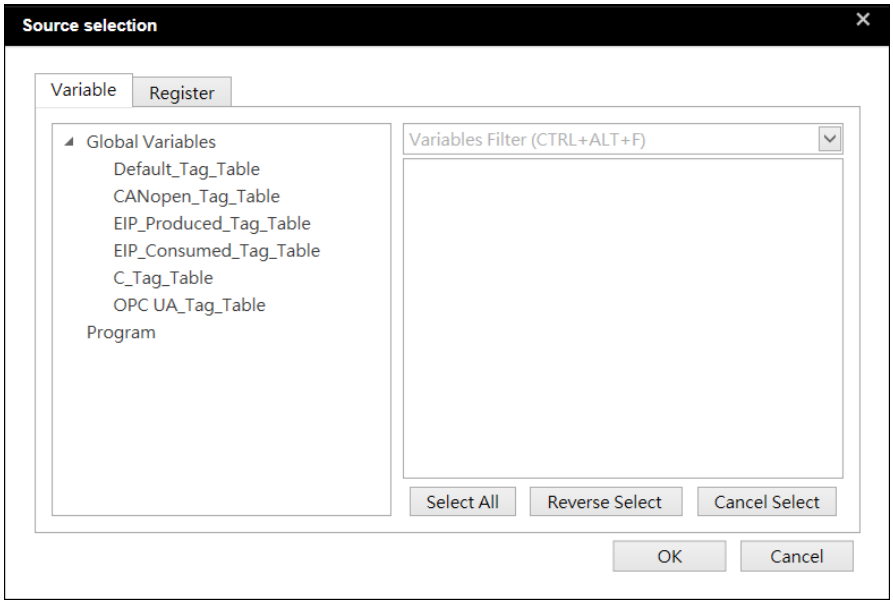


The following table describes the functions in the **Monitor Table** toolbar.

Function	Description
Add Mode	Click to select **Add Mode** so that the entry can be added in the end.
Insert Mode	Click to select **Insert Mode** so that the entry can be added in between the existing table lines.
	Click to open **Source selection** dialog to add Variables / Registers.
	Click to delete the selected items.
	Click to move an item above its present level.
	Click to move an item below its present level.
	Click to download **Monitor Table**.
	Click to import the content and values of files to the current monitor table.
	Click to export the content and current value to CSV file.
Decimal Place	Enter decimal places.

14.2.2 Adding Variable and Registers

Click  to open **Source selection** dialog to add variables and registers or double-click the empty area of the Monitor Table to add.



The **Source selection** dialog has two tabs.

- Variable
- Register

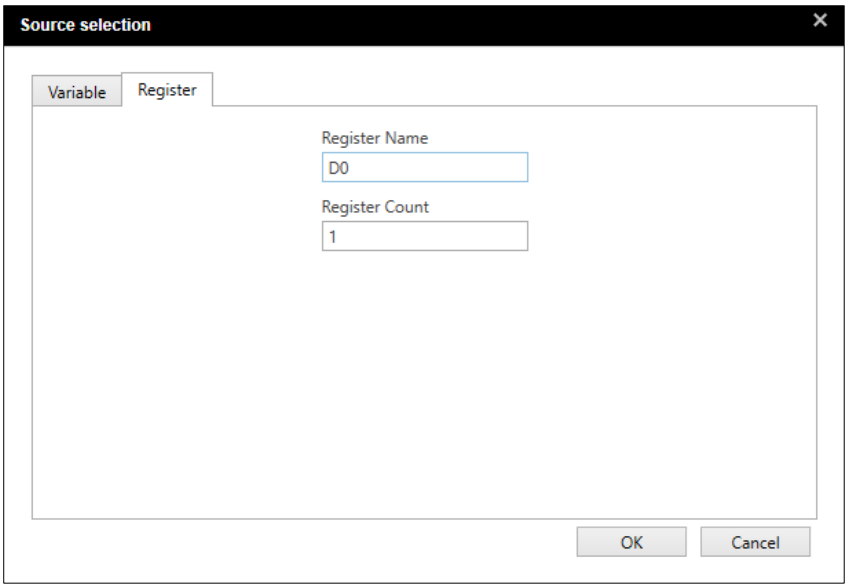
The **Variable** tab appears by default in the **Source selection** dialog. Select **Global Variables** or **Program** category. The **Global Variables** display **Default_Tag_Table**, **CANopen_Tag_Table**, **EIP_Produced_Tag_Table**, **EIP_Consumed_Tag_Table**, **C_Tag_Table**, and **OPCUA_Tag_Table**.

The **Program** displays all the POUs created in the controller in **Programming** > **Program**. Based on the item selected on left, the respective variables display in the right side of the **Variable** tab.

The following table describes the functions in the **Variable** tab of **Source selection** dialog.

Function	Description
Variables Filter	Enter part of the variable name to filter out the relevant variables.
Select All	Click to select all variables.
Reverse Select	Click to revert the selected variables.
Cancel Select	Click to cancel all the variables.

If you want to add **Register**(s) to the **Monitor Table**, click **Register** tab.

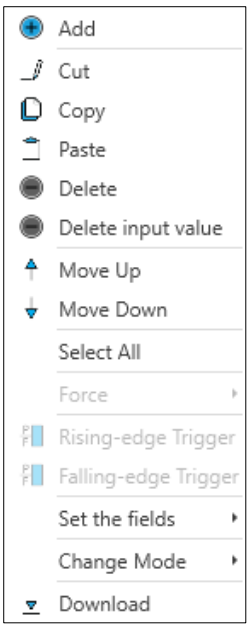


The following table describes the functions in the **Register** tab of **Source selection** dialog.

Function	Description
Register Name	Enter the starting register.
Register Count	Enter the number of registers needed from the starting register.

14.2.3 Monitor Table Context Menu

The following figure shows the context menu in **Monitor Table** dialog.

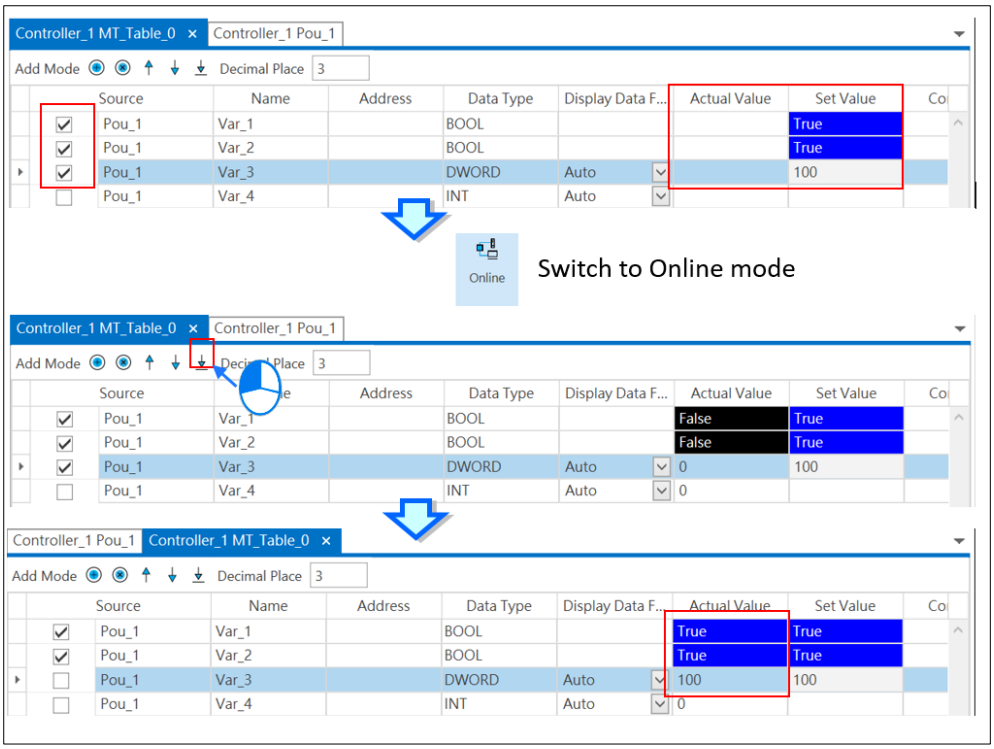


The following table describes the functions in the **Monitor Table** context menu.

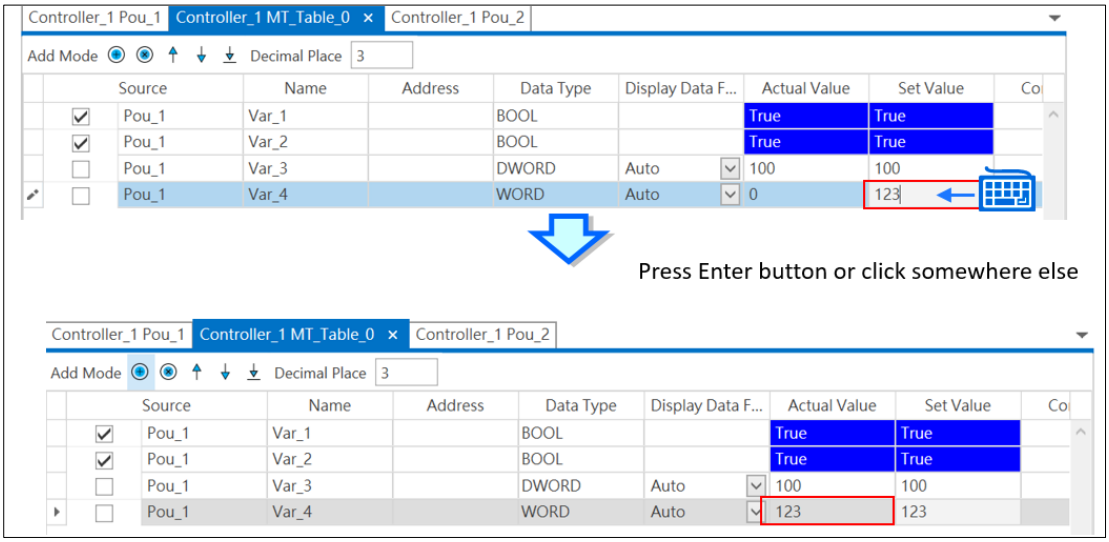
Function	Description
Add	Click to open **Source selection** dialog to add variables / registers.
Cut	Click to cut the selected item.
Copy	Click to copy the selected item.
Paste	Click to paste the cut/copied item.
Delete	Click to delete the selected items.
Delete input value	Click to delete current input value.
Move Up	Click to move an item above its present level.
Move Down	Click to move an item below its present level.
Select All	Click to select all items.
Force	Click to force On / Off Boolean Variables.
Rising-edge Trigger	Send OFF for one scan time and then ON.
Falling-edge Trigger	Send ON for one scan time and then OFF.
Set the fields	Click to hide / display the column property. Tick mark the left side of the function means the respective column is shown. **Source**, **Data Type**, **Display Data Format**, **Actual Value**, **Set Value**, and **Comment** can be displayed or hidden. **Note**: All columns display by default.
Change Mode	Click to select **Add Mode** / **Insert Mode**.
Download	Click to download the selected items

14.2.4 Using Monitor Table

In online mode, monitor items after adding a monitor table and monitor variables / registers. In offline mode, select the items to be set by checking the checkboxes on the left, and set the values in **Set Value** field. Then, switch to the online mode, do batch download of the selected variables or register values by clicking the download button on toolbar or the context menu.



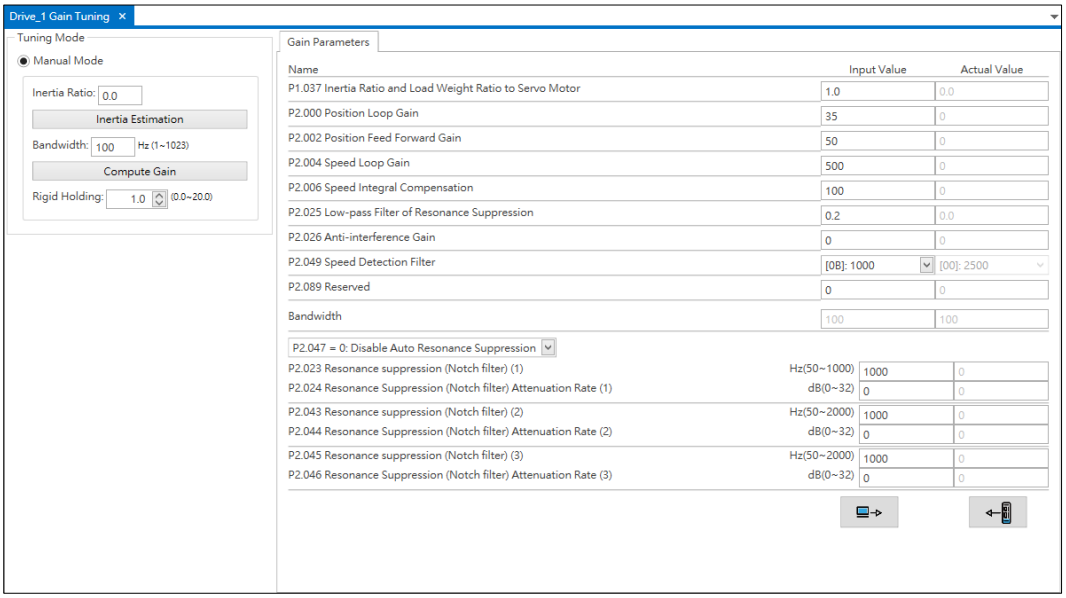
In the online mode, when setting **Set Value**, press **Enter** or click somewhere else with mouse to write the values to **Actual Value**. For BOOL data type, double-click the **Set Value** to toggle state.



Note: If the variables in the monitoring table are deleted, or other problems such as synchronization failure occur, the value of the **Actual Value** field becomes N/A.

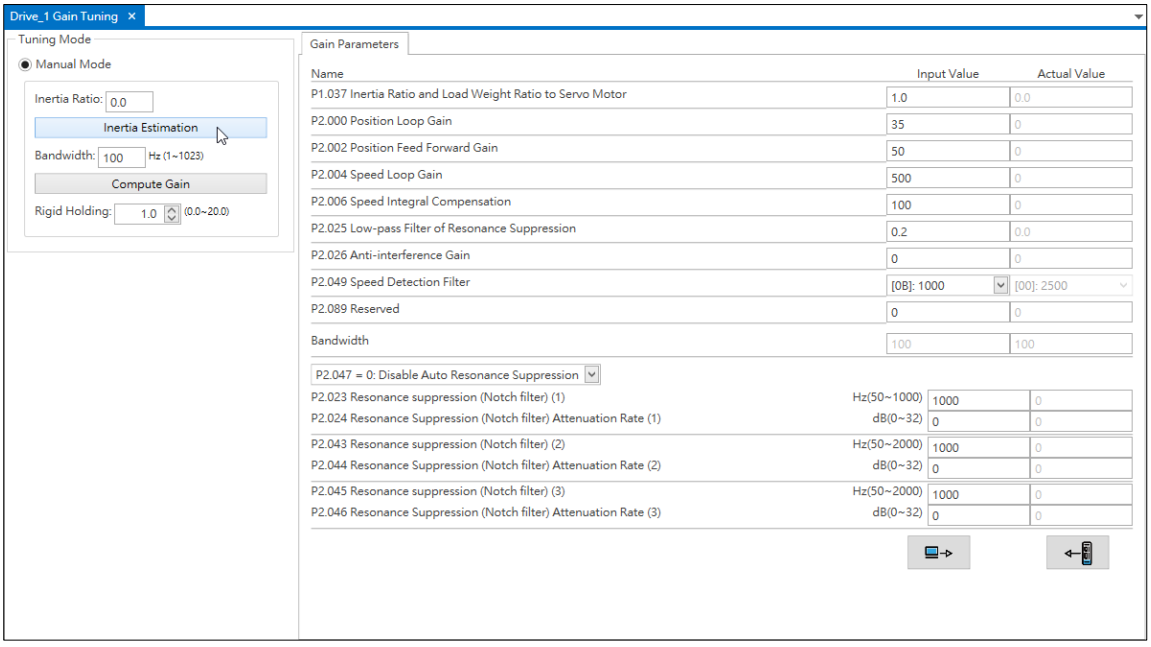
14.3 Tuning Tool

Tune the Servo Drive in **Gain Tuning**. Locate Gain Tuning in **Project Explorer** in **Servo Device** > **Tuning** > **Gain Tuning**. Double-click **Gain Tuning** to open **Gain Tuning** dialog.

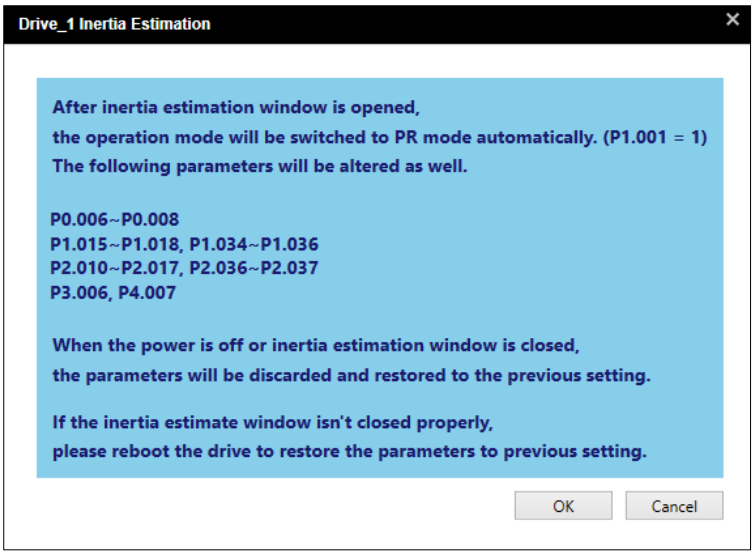


14.3.1 Calculate Inertia Ratio

Click **Inertia Estimation** to open **Inertia Estimation** dialog.



A warning message appears, as shown in the following figure. Click **OK** to continue.



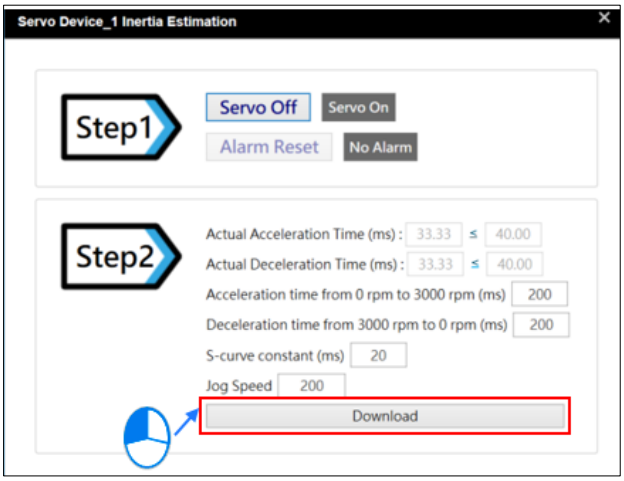
The **Inertia Estimation** dialog opens to estimate inertia ratio step-by-step.

To estimate inertia ratio

1. Click **Servo On** button. If there is alarm, clear the alarm by clicking **Alarm Reset** button.

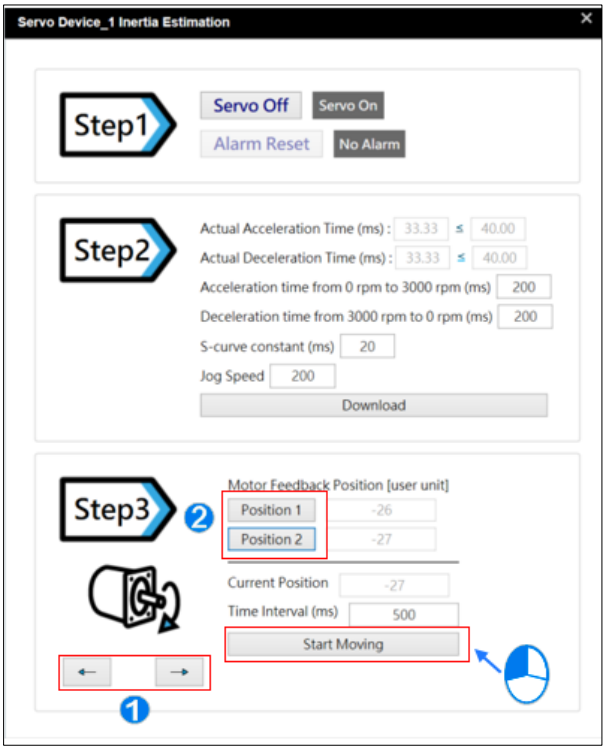


2. Enter Acceleration / Deceleration Time, **S-curve constant**, and **Jog Speed**, then click **Download**.



3. Arrows in ❶ indicates to move the motor to the desired position and use **Position 1** and **Position 2** in ❷ indicates to set the positions. After the positions are set,

click **Start Moving** to do the inertia estimation.

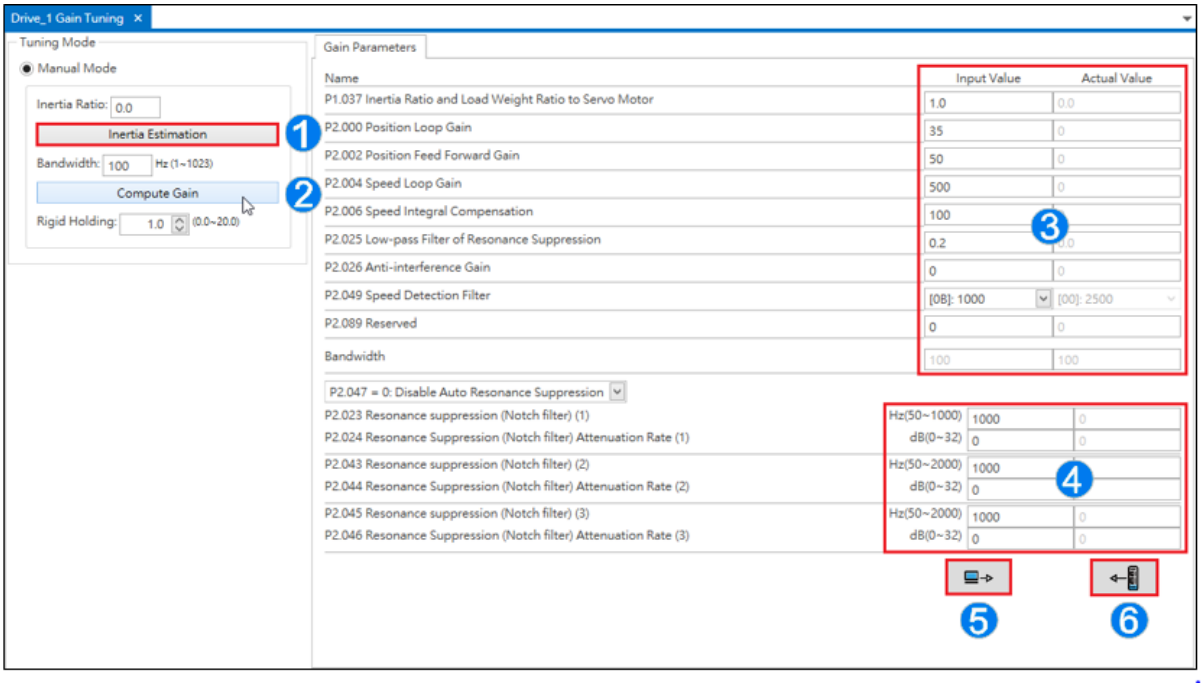


4. After the **Inertia Ratio** becomes stable, go back to Step 2 to set a higher inch speed. Then click **Start Moving** in Step 3 to perform another inertia estimation.
5. Repeat Steps 2 to 4 until the value of the **Load Inertia Ratio** does not change significantly with the increase in speed.
6. Click **Setting** to update the value of **Inertia Ratio**.



14.3.2 Set Up the Bandwidth

After calculating **Inertia Ratio**, the next step is to set up the Servo Drive **Gain**.



The following table describes the functions in the **Gain Parameters** in **Gain Tuning** dialog.

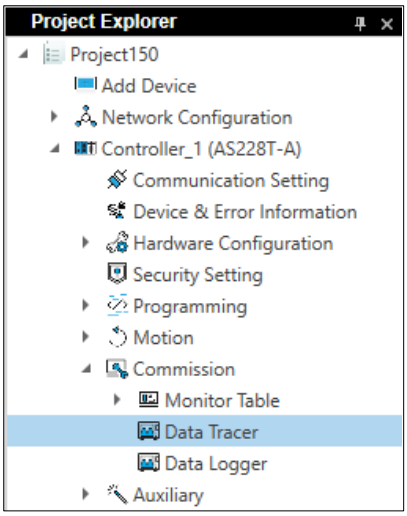
Function	Description
❶	Enter **Bandwidth**.
❷	Click **Compute Gain** to calculate the gain parameters.
❸	Gain parameters. • **Input Value** from the calculation of **Bandwidth**. It can be entered by users or entered automatically after calculating the **Gain** by function 2. • **Actual Value** from the actual value in servo drive. To obtain by function 6.
❹	To set up the resonance suppression filter.
❺	To download **Input Value** to servo drive **(Actual Value)**.
❻	To upload the parameters from servo drive to show in **Actual Value**.

14.4 Trace and Oscilloscope

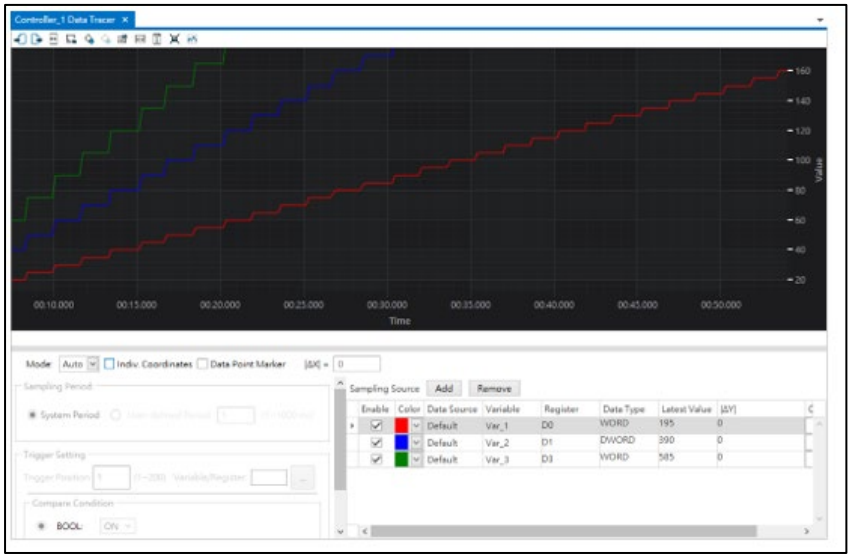
View the waveforms or trends of value in **Address** and **Register** with the help of **Data Tracer** and **Data Logger**. Locate them in **Controller** > **Commissioning**.

14.4.1 Data Tracer

The **Data Tracer** is used for the real-time data collection of values or states in variables or addresses after some trigger condition is met to perform analysis with the waveforms or trends. Once the dialog is closed, the data tracing stops. Locate **Data Tracer** in the **Project Explorer** in **Controller** > **Commissioning** > **Data Tracer**.



Double-click **Data Tracer** to open **Data Tracer** dialog.



The **Data Tracer** dialog has the following areas.

- Data Trace toolbar
- Trace Display area
- Sampling Source and Parameter Setting

14.4.1.1 Data Trace Toolbar

The **Data Trace** toolbar provides icons for data trace management as shown in the following figure.

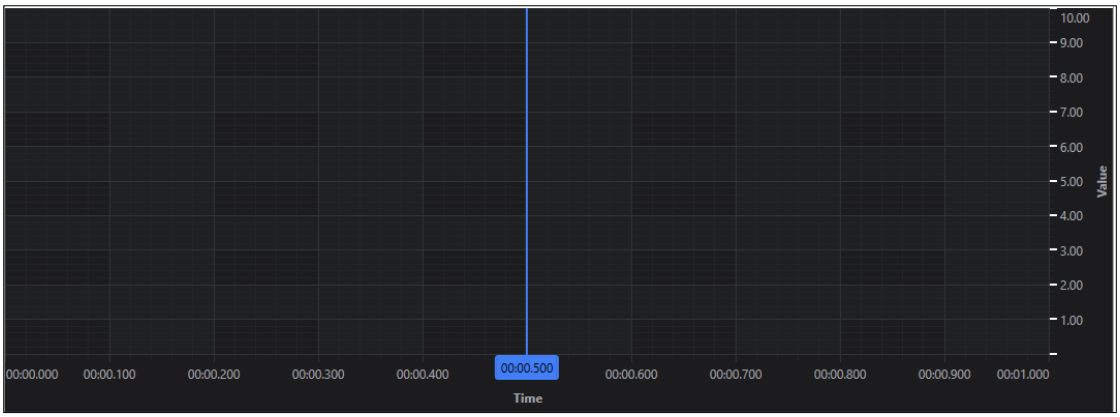


The following table describes the functions in **Data Tracer** toolbar.

Function / Icon	Description
	Open the sample setting from a specified path. Click to display **Open** dialog (*.trascp file).
	Save the sample setting to a specified path. Click to display **Save As** dialog (*.trascp file).
	Export the trend data to a specified path. Click to display **Save As** dialog (*.csv file).
	Click to take a screenshot of the trend display. (*.png, *.jpg & *.bmp)
	Click to start Data Trace.
	Click to stop Data Trace.
	Click to clear Data Trace.
	There are two vertical lines in the trend display area to measure the trend.
	There are two horizontal lines in the trend display area to measure the trend.
	Click to zoom fit, same as double-click the Trace Display Area.
	Click to open **FFT Analysis** dialog.

14.4.1.2 Trace Display Area

The variable trends are displayed in real-time.

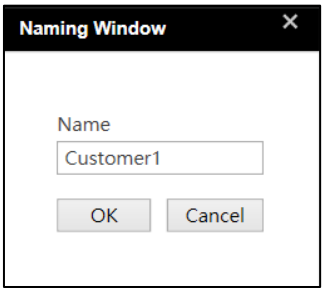


14.4.1.3 Sampling Source and Parameter Setting

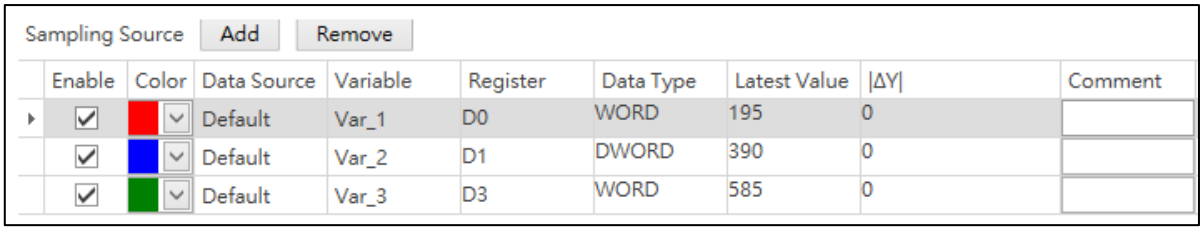
The **Sampling Source** and the **Parameter Setting** are at the bottom of the log display, and can be set through the toolbar.



The following table describes the toolbar functions.

Function	Description
Quick Setting	Save the set monitoring channel list. Click **Add** to add the selected monitoring channels to the list. A **Naming Window** appears to type the name. Select the saved monitoring list from the drop-down list. Click **Delete** to delete the selected list. 
Mode	Select **Auto** or **Trigger**.
Indiv. Coordinates	Make each channel corresponds to the individual Y axis of its own.
Data Point Marker	Mark each sampling data point on the waveform.
|ΔX|	Display the time interval of the X axis. Click  on toolbar to enable this function.

The **Sampling Source** area is on the right-side bottom of the **Data Tracer** dialog as shown in the following figure.



The following table describes the functions in the **Sampling Source** area in the **Data Tracer** dialog.

Function	Description
Add	Click to open **Select Variable** dialog. Select **Variables** / **Registers** to add sample source.
Remove	Click to remove the sample source selected.
Enable	Click the checkbox to enable the sample source.
Color	Select the trace color from the color pallet.
Data Source	Display the source of data e.g. **Program** name.
Variable	Display the variable name e.g. LT101.
Register	Display the register name e.g. D100.
Data Type	Display the data type of the register.
Last Value	Display the last value.
|△Y|	Enter the Y-axis field (Sampling Source) value to start Data Trace recording
Comment	Display the comment of variable or register.

14.4.1.4 Adding Variables or Registers to Data Tracer

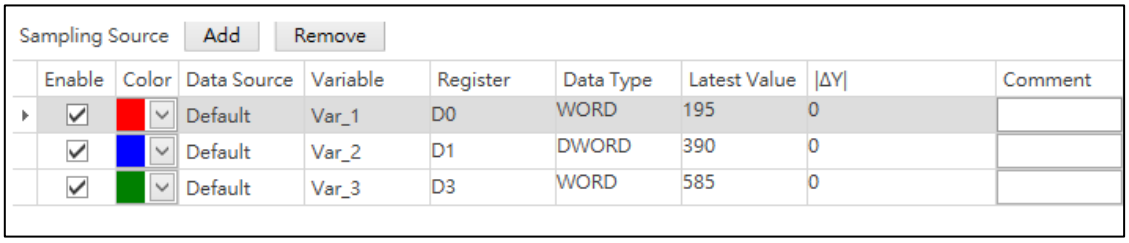
Follow these steps to add variables or registers to **Data Tracer**.

To add variables / registers to Data Tracer

1. Click **Add** in the **Sample Source** area to open **Select Variable** dialog.

   The **Select Variable** dialog contents are the same as the one described earlier.

2. Click **OK** to finish sample source selection for variables or registers. The following figure shows an example of sample sources for **Data Tracer**.

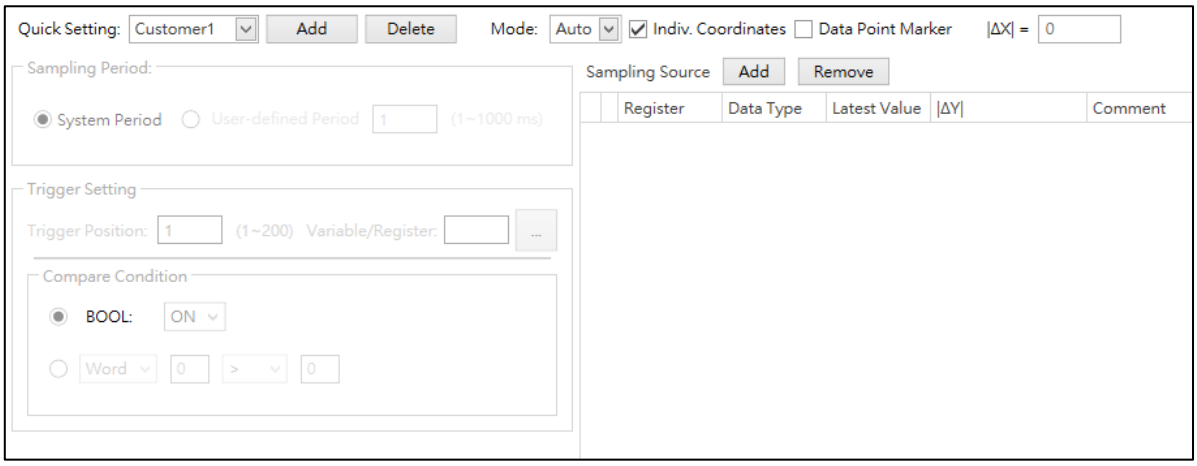


14.4.1.5 Parameter Setting

Data Trace can be activated in two modes.

- Auto
- Trigger

Sampling Period and **Trigger Setting** are available on the left-side bottom of the **Data Trace** dialog.



14.4.1.6 Sampling Period and Trigger Setting Functions

The following table describes the functions in **Sampling Period** and **Trigger Setting** area in the **Data Tracer** dialog.

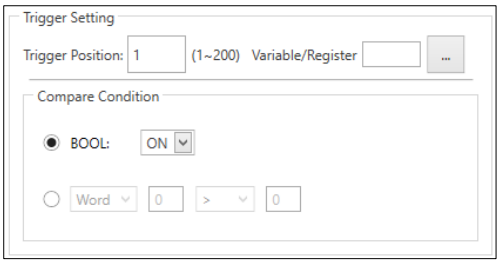
Function	Description
Mode	Select the **Sampling Period** and **Trigger Setting** mode. The available options include: • Auto • Trigger **Note**: The default value is **Auto**. In **Auto** mode, **Sampling Period** and **Trigger Setting** fields are grayed out. They are enabled in the **Trigger** mode.
Indiv. Coordinates	Refer to Individual Coordinates.
Data Point Marker	Refer to Data Point Marker.
(ΔX)	Enter the X-axis field (time) to start Data Trace recording.
Sampling Period – System Period	Click to select **System Period**. It means sampling is done once in every scan cycle of the Controller.
Sampling Period – User-defined Period	Click to select **User-defined Period** and enter the sampling period time in the range of (1 ~ 1000ms).

Function	Description
Trigger Setting – Trigger Position	Enter the trigger positions as per your requirement. The PLC read 200 pieces of sample data and display the data in the Data Tracer if the trigger condition is met. The trigger position setting is different for one-shot trigger and continuous trigger.
Trigger Setting – Variable/Register	Click  to open the **Select Variable** dialog to select the trigger setting variable / register on whose values the trigger for sampling must be performed.
Trigger Setting – Compare Condition - BOOL	Click to select **BOOL** radio button if the trigger variable / register is Bool. Select the trigger condition from the drop-down list. The available options include: • ON • OFF • Rising Edge Triggered • Falling Edge Triggered • Rising or Falling Edge Triggered **Note**: The default Variable / Register type is **BOOL**, and the default trigger value is **ON**.
Trigger Setting – Compare Condition	Select the non-Boolean variable / register for trigger. The available options include: • Word • DWord • Real Select the comparison condition and value. The available options include: • >= • < • == • != • ~ • !~ **Note**: The default non-Boolean trigger variable / register is **Word**, and the default comparison condition is **>**.

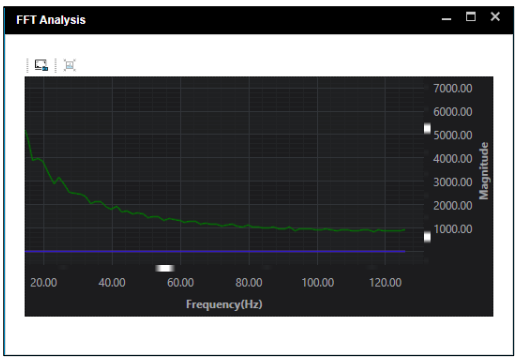
The following figure shows **Sampling Period** setting.



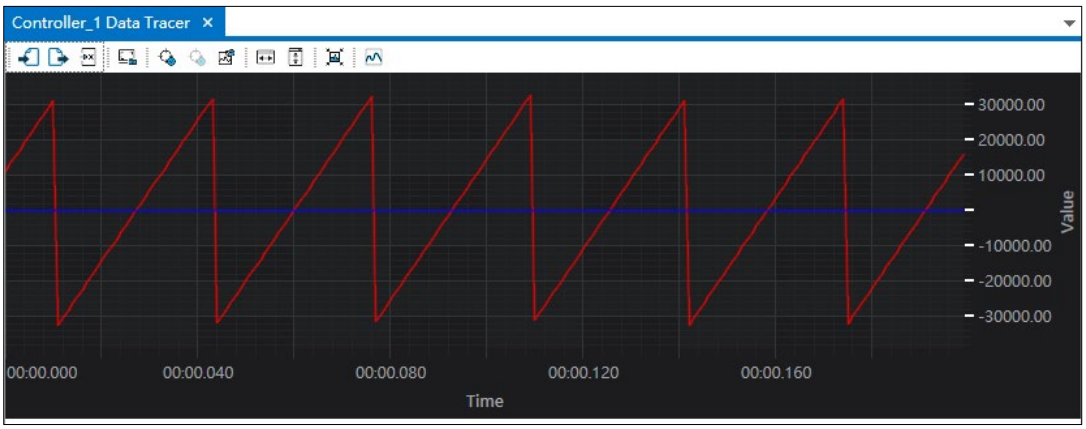
The following figure shows **Trigger Setting**.



The following figure shows sample **FFT Analysis** dialog where the magnitude and the frequency can be compared.

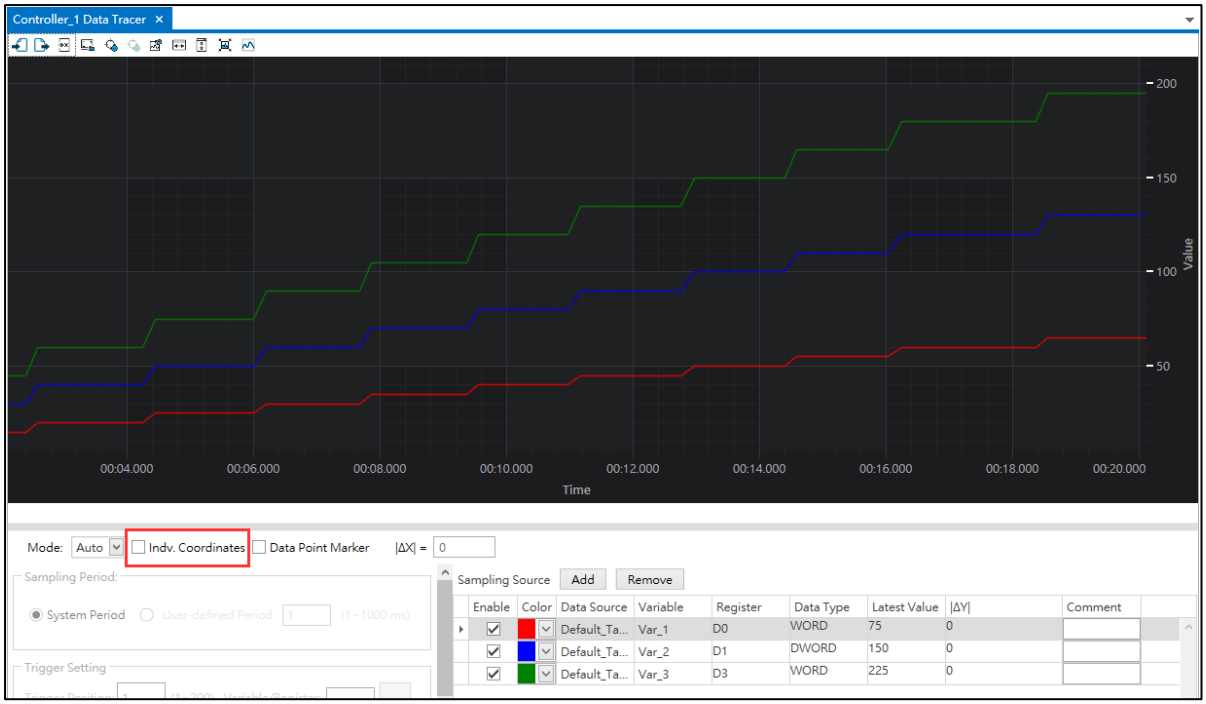


The following figure shows the result of Data Tracer. You can hold down the right button of the mouse and drag the trend presentation position, and use the mouse wheel to zoom in and out of the presentation range.



14.4.1.7 Individual Coordinates

The screen displays data according to the individual coordinate system functions. When the **Indiv. Coordinates** is not checked, the data of the data oscilloscope share the Y-axis coordinate, which is displayed on the screen as shown in the following figure.

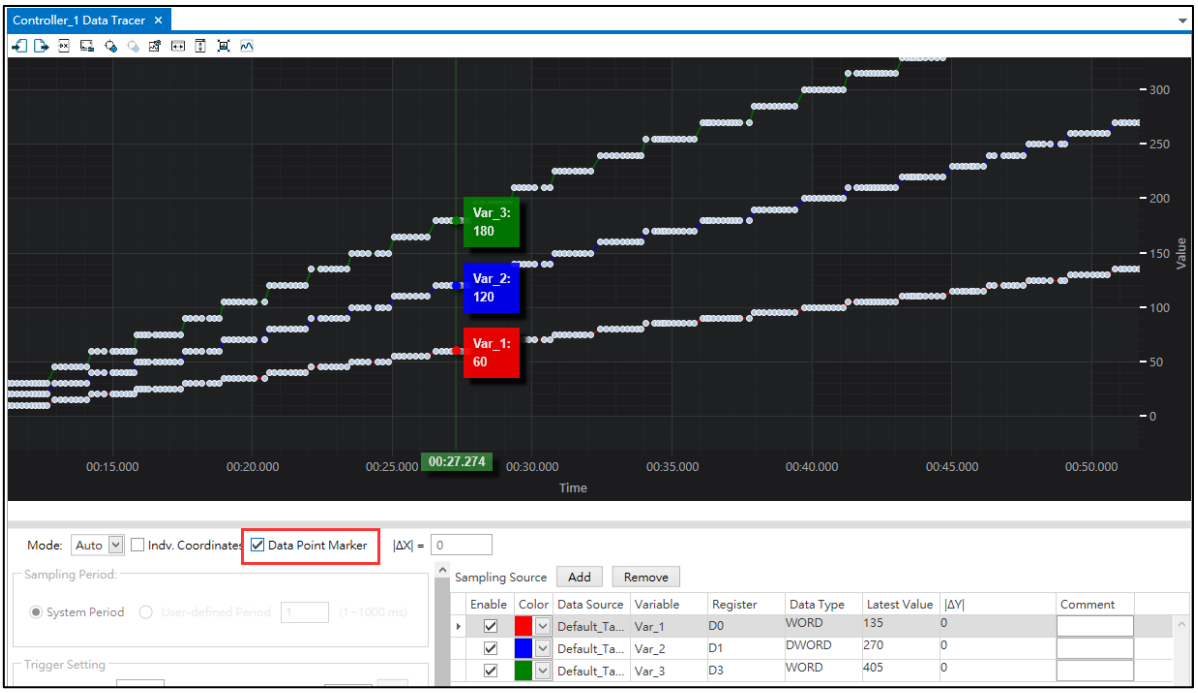


When the **Indiv. Coordinates** function is checked, the data of the oscilloscope can choose different variables, and the Y-axis coordinate value displays the value range of the variable. After clicking a variable below, you can use the left mouse button to drag the Y-axis coordinate axis to arbitrarily scale or adjust the up and down position of the variable sampling waveform. In this way, the relative position of each waveform on the screen can be adjusted.



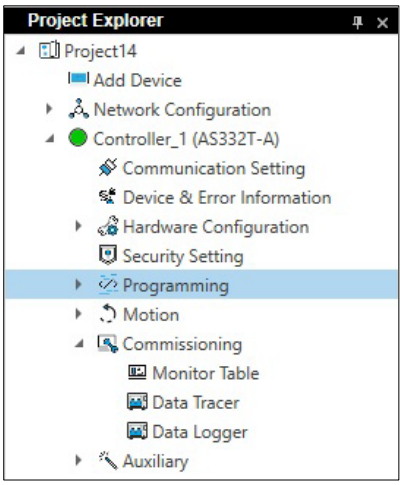
14.4.1.8 Data Point Marker

When the **Data Point Marker** function is checked, the data oscilloscope displays all data acquisition points. Move the mouse on the screen to display the variable name and value of the data point, as shown in the following figure.

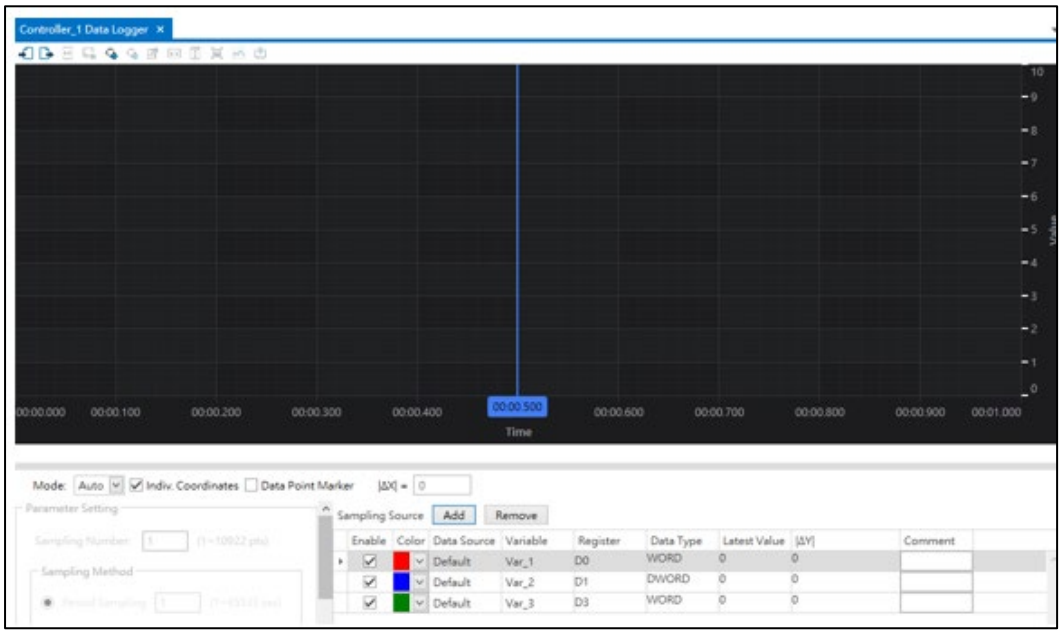


14.4.2 Data Logger

The **Data Logger** is used for the long-term recording of values or states in variables / registers, trending and storing them in controller and the memory card of the controller for analysis of trends. Locate Data Logger in **Project Explorer** in **Controller** > **Commissioning** > **Data Logger**.



Double-click **Data Logger** to open **Data Logger** dialog.



The **Data Logger** dialog has following areas.

- Data Logger toolbar
- Log Display area
- Sampling Source and Parameter Setting area

14.4.2.1 Data Logger Toolbar

The following figure shows **Data Logger** toolbar that provides icons for data log management.



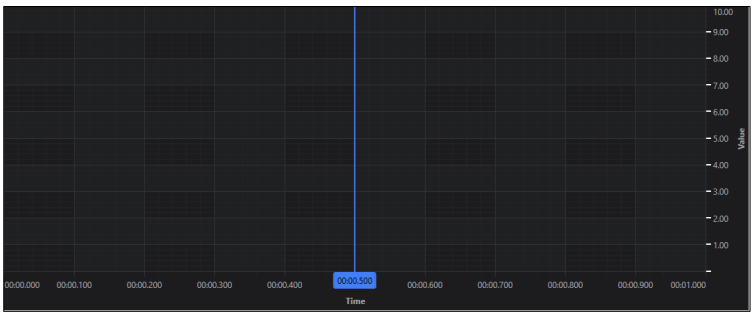
The following table describes Data Logger toolbar functions.

Function	Description
	Open the Sampling setting from a specified path. Click to display **Open** dialog (*.logscp file).
	Save the Sample setting to a specified path. Click to display **Save As** dialog (*.logscp file).
	Export the trend data to a specified path. Click to display **Save As** dialog (*.csv file).
	Click to take a screenshot of the trend display (*.png, *.jpg & *.bmp).

Function	Description
	Click to start Data Log.
	Click to stop Data Log.
	Click to clear Data Log.
	There are two vertical lines in the trend display area to measure the trend.
	There are two horizontal lines in the trend display area to measure the trend.
	Click to zoom fit, same as double-click the Logger Display Area.
	Click to open **FFT Analysis** dialog.
	Click to write records to memory card.

14.4.2.2 Log Display

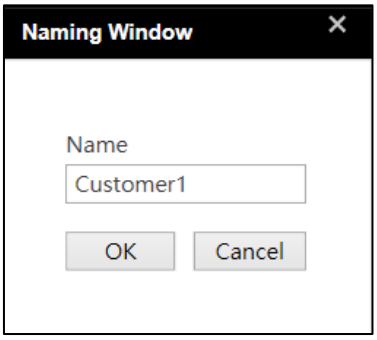
The variable trends are displayed in real-time and stored. The following figure shows **Log Display** area of **Data Logger**.



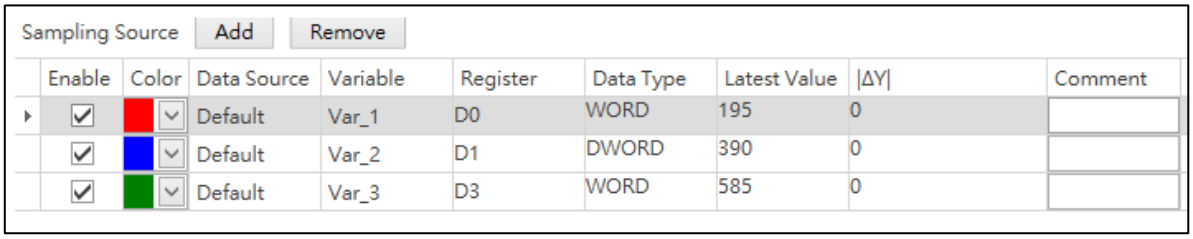
14.4.2.3 Sampling Source and Parameter Setting

The **Sampling Source** and **Parameter Setting** area are at the bottom of the **Log Display** area.



Function	Description
Quick Setting	Save the set monitoring channel list. Click **Add** to add the selected monitoring channels to the list. A **Naming Window** appears to type the name. Select the saved monitoring list from the drop-down list. Click **Delete** to delete the selected list. 
Mode	Select **Auto** or **Record**.
Indiv. Coordinates	Make each channel corresponds to the individual Y axis of its own.
Mark Data Point	Mark each sampling data point on the waveform.
|ΔX|	Display the time interval of the X axis. Click  on toolbar to enable this function.

The **Sampling Source** area is at the right-side bottom of the **Data Logger** dialog.



The following table describes the functions in the **Sampling Source** area in the **Data Logger** dialog.

Function	Description
Add	Click to open **Select Variable** dialog. Select **Variables** / **Registers** to add sample source.
Remove	Click to remove the sample source selected.
Enable	Click the checkbox to enable the sample source.

Function	Description
Color	Select the trace color from the color pallet.
Data Source	Display the source of Data e.g. **Program** name.
Variable	Display the variable name e.g. LT101.
Register	Display the register name e.g. D100.
Data Type	Display the data type of the register.
Last Value	Display the last value.
|ΔY|	Enter the Y-axis field (Sampling Source) value to start Data Trace recording.
Comment	Display the comment of variable or register.

14.4.2.4 Adding Variables or Registers to Data Logger

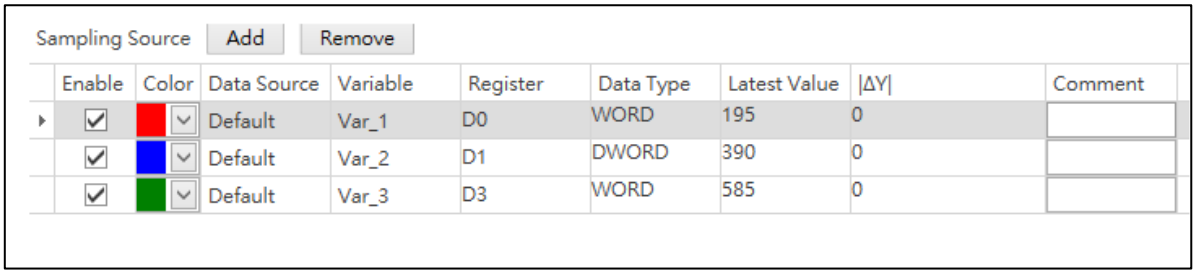
Follow these steps to add variables or registers to **Data Logger**.

To add variables / registers to Data Logger

1. Click **Add** in the **Sample Source** area to open **Select Variable** dialog.

   Note: The **Select Variable** dialog contents are the same as the one described earlier.

2. Select variables or registers in **Select Variable** dialog. Click **OK** to finish sample source selection for variables or registers. The following figure shows an example of sample sources for **Data Logger**.



14.4.2.5 Parameter Setting

Data Log can be activated in two modes.

- Auto

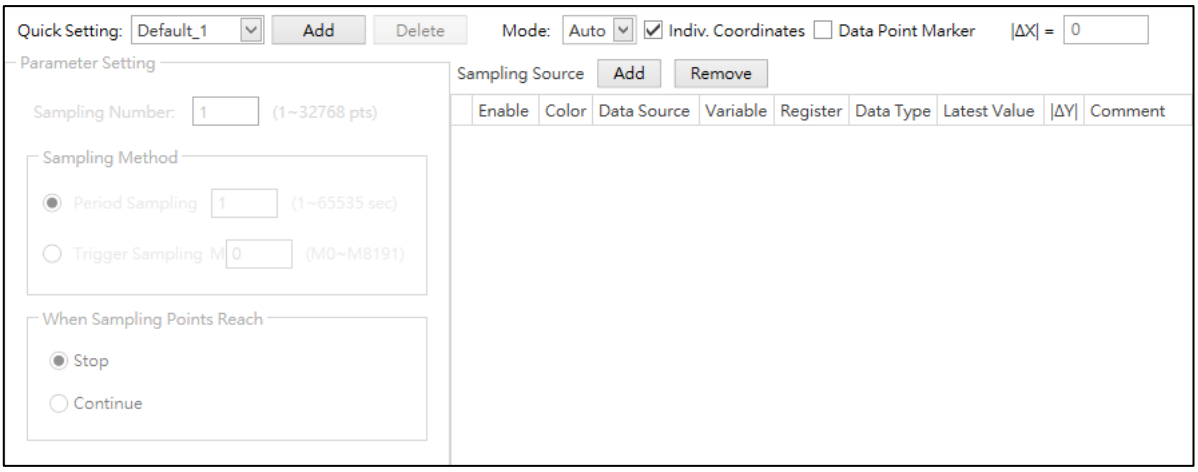
- Record

The following table describes the functions in the **Parameter Setting** area in the **Data Logger** dialog.

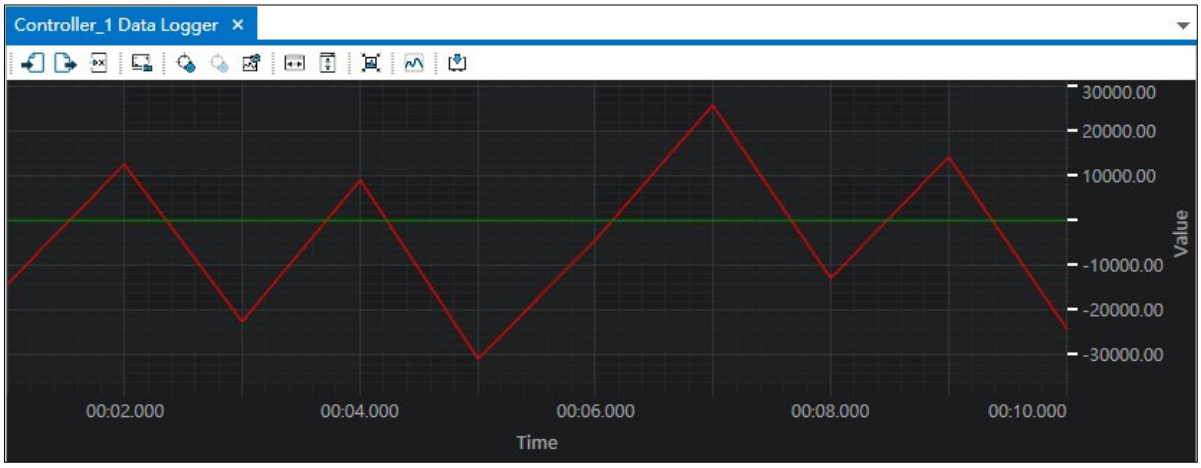
Function	Description
Mode	Select the **Sampling** and **Parameter Setting** mode. The available options include: • Auto • Record **Note**: The default value is **Auto**. In **Auto** mode, **Parameter Setting** field is grayed out. They are enabled in the **Record** mode.
Indiv. Coordinates	Refer to Individual Coordinates System Function.
Data Point Marker	Refer to Data Point Marker.
(ΔX)	The difference value of 2 vertical line.
Parameter Setting – Sampling Number	Enter the sampling number in 1~32768 pts range.
Parameter Setting – Sampling Method	Select the sampling method. The available options include: • Periodic Sampling • Trigger Sampling **Note**: The default value is **Periodic Sampling**. When using **Periodic Sampling**, setup the sample period and the **Data Logger** executes from the **Sampling Source** every period. When using the **Trigger Sampling**, setup the trigger register (M register).
Parameter Setting – Sampling Method – Period Sampling	Click to select **Sampling Method** - periodic sampling. Enter the sampling period in 1~65535 sec.
Parameter Setting – Sampling Method – Trigger Sampling M	Click to select **Sampling Method** – triggered sampling. Enter M register for sample trigger. If the trigger register is ON, the Data Logger executes the sampling once and after completing the sampling process, it automatically turns the device to OFF. **Note**: If the user program tries to execute other actions to the M Register, then the ON / OFF function of the device can be affected.
Parameter Setting – When Sample Points Reach	Select the action when the Data Logger has recorded the number of set sample points. The available options include: • Stop

Function	Description
	• Continue **Note**: The default value is **Stop**.

The **Parameter Setting** area is used for setting sample parameters related with the **Data Logger**.

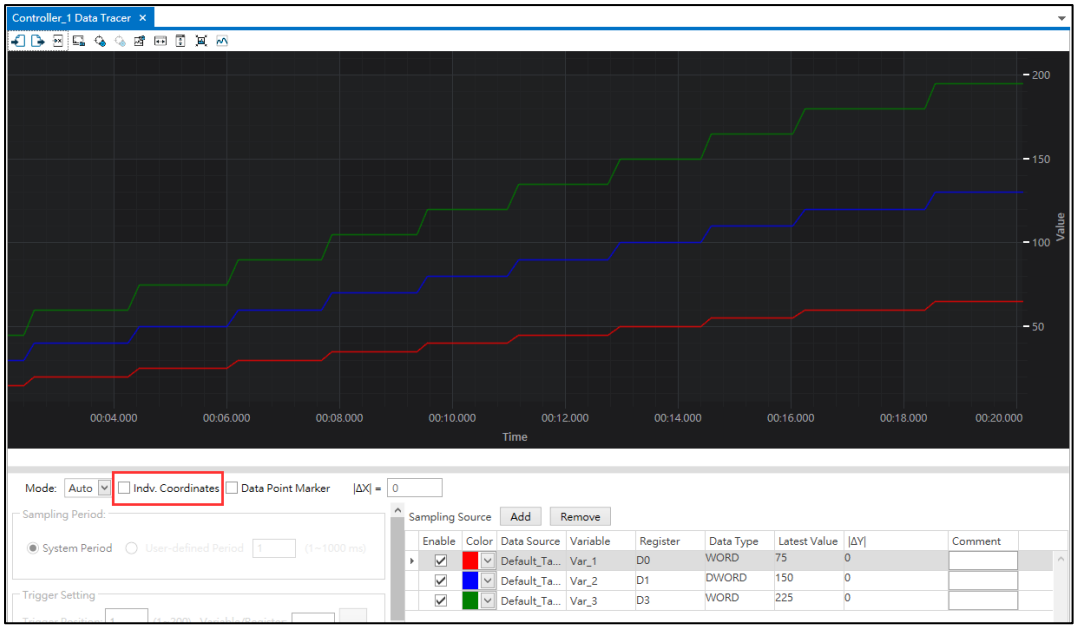


The following figure shows the result of **Data Logger**. You can hold down the right button of the mouse and drag the trend display position, and use the mouse wheel to zoom in and out of the presentation range.



14.4.2.6 Individual Coordinate System Function

The screen displays data according to the individual coordinate system functions. When the **Indiv. Coordinates** is not checked, the data of the data logger share the Y-axis coordinate, which is displayed on the screen as shown in the following figure.

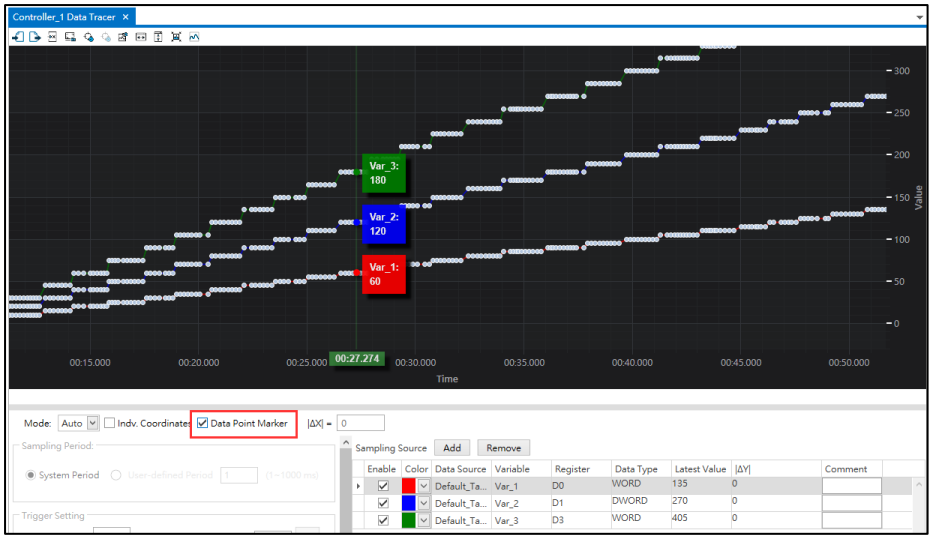


If the **Indiv. Coordinates** is checked, the data of the data oscilloscope can select different variables, and the Y-axis coordinate value displays the numerical range of the variable. After clicking a variable below, you can use the left mouse button to drag the Y-axis coordinate axis to arbitrarily scale or adjust the up and down position of the variable sampling waveform. In this way, the relative position of each waveform on the screen can be adjusted.



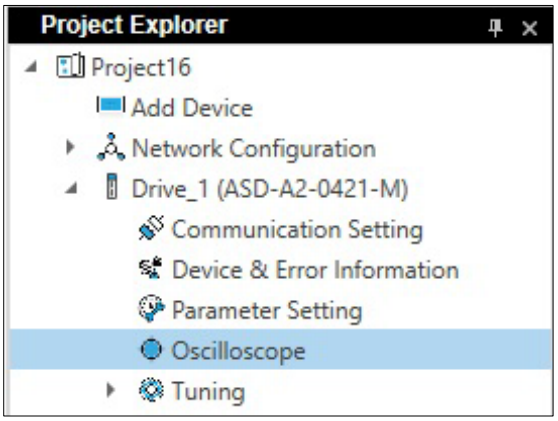
14.4.2.7 Data Point Marker

When the **Data Point Marker** function is checked, the data logger displays all the data capture points, and the variable name and value of the data point can be displayed by moving the mouse on the screen, as shown in the following figure.

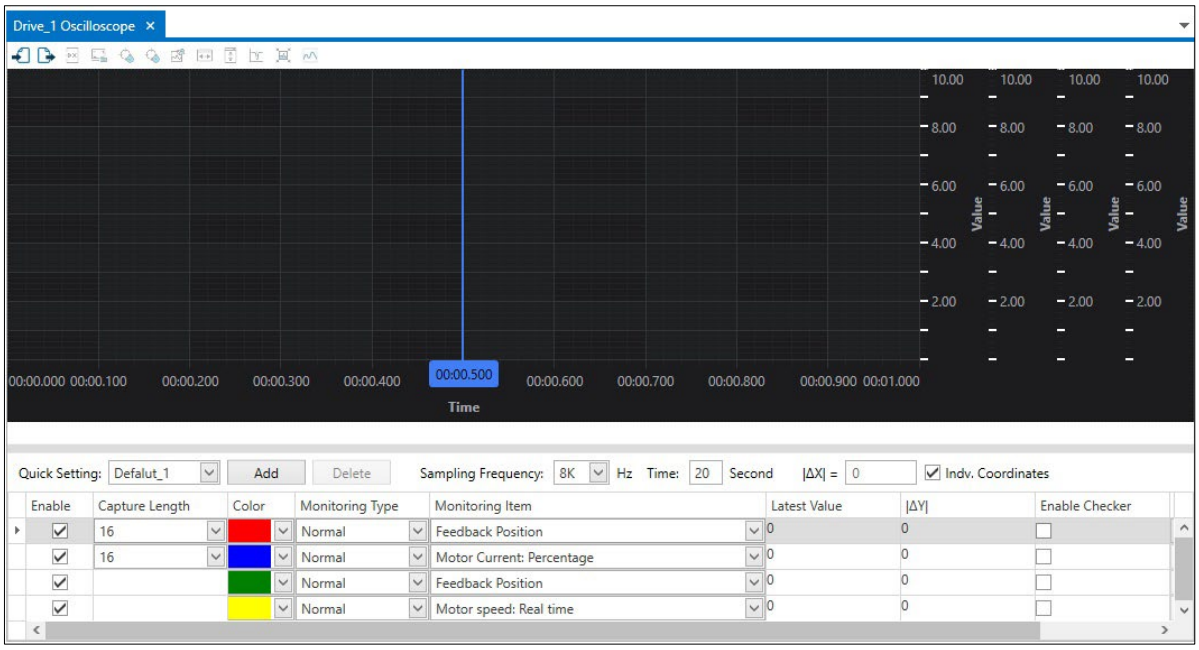


14.4.3 Servo Device Oscilloscope

Servo Device Oscilloscope is used for the real-time servo drive status. Locate Servo Device Oscilloscope in **Servo Device** > **Oscilloscope** in **Project Explorer**.



Double-click **Oscilloscope** to open **Oscilloscope** dialog.

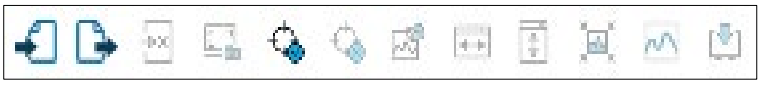


The **Servo Drive Oscilloscope** dialog has the following areas.

- Oscilloscope toolbar
- Display area
- Monitoring Channel Setting

14.4.3.1 Oscilloscope Toolbar

The following figure shows **Oscilloscope** toolbar.



The following table describes the functions in Servo Device **Oscilloscope** toolbar.

Function	Description
	Open the sample setting and points from a specified path. Click to open **Open** dialog (*.asdscp file).
	Save the sample setting and points to a specified path. Click to open **Save As** dialog (*.asdscp file).
	Export the trend data to a specified path. Click to open **Save As** dialog (*.csv file).
	Click to take a screenshot of the trend display (*.png, *.jpg & *.bmp)
	Click to start scope.

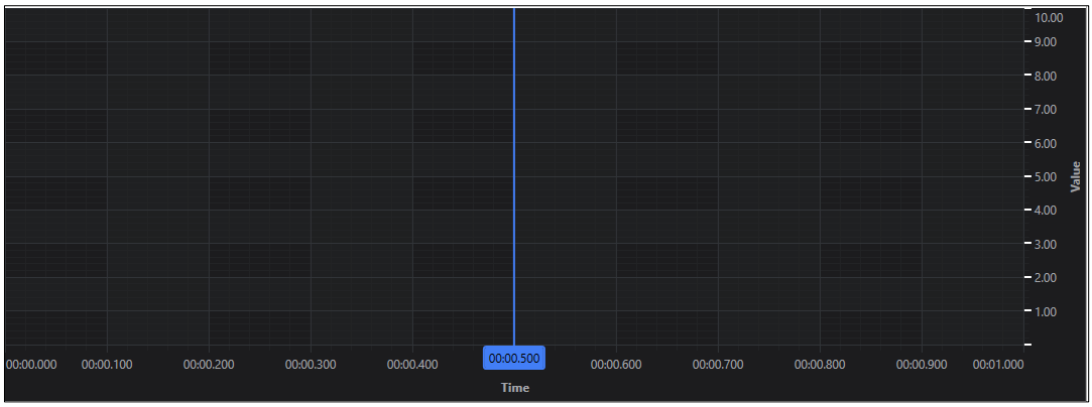
Function	Description
	Click to stop scope.
	Click to clear scope display.
	There are two vertical lines in the trend display area to measure the trend.
	There are two horizontal lines in the trend display area to measure the trend.
	Click to zoom fit, same as double-click the trend display area.
	Click to open **FFT Analysis** dialog.

14.4.3.2 Oscilloscope Display Area

The selected monitoring channels are displayed in real-time. You can hold down the right button of the mouse and drag the display area presentation position, and use the mouse wheel to zoom in and out of the presentation range. X axis of the screen represents time. Unit is second (sec). Y axis of the screen represents data. Y axis on the right side is allocated according to the channel you select. Data unit displayed in Y axis is marked with the same coordinates when in initial status.

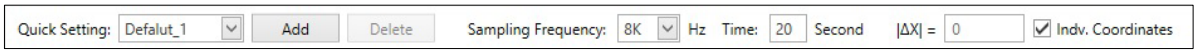
It is suitable for the monitor items which have the same unit. If the monitor items have different unit, it is suggested to select **Indiv. Coordinates** in the top part of **Monitoring Channel Setting** area. After clicking a variable below, you can use the left mouse button to drag the Y-axis coordinate axis to arbitrarily scale or adjust the up and down position of the variable sampling waveform. In this way, the relative position of each waveform on the screen can be adjusted.





14.4.3.3 Monitoring Setting

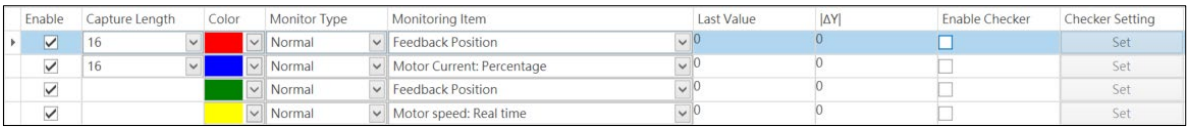
The area below the display area is the **Monitoring Setting** area. Upper part of **Monitoring Setting** area is to set the display condition of display area.



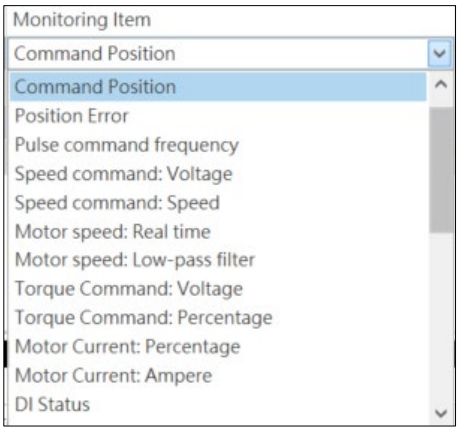
The following table describes the functions in the **Monitoring Setting** area in **Servo Drive Oscilloscope** dialog.

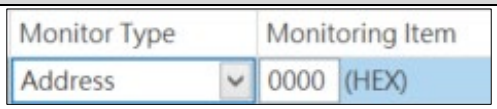
Function	Description
Quick Setting	To store the saved monitoring channel list. Click **Add** to save the current selected monitoring channels into list. A **Naming Window** appears to type the name. Select the saved monitoring channel list from drop-down manual. Click **Delete** to delete the selected list.
Sampling Frequency	To select the sampling frequency in **Display Area**. To select 8 KHz to monitor 4 monitoring channels or 16 Hz to monitor 2 monitoring channels.
Time	Adjust the resolution of time axis (X axis of Scope). The resolution of scope image can be enhanced. The setting range is from 20 seconds ~ 120 seconds)
|ΔX|	Display the time interval of X axis when enabling the function of vertical line measurement from **Servo Drive Oscilloscope** toolbar  .
Indiv. Coordinates	Channel with individual coordinate shows the actual monitor variables.

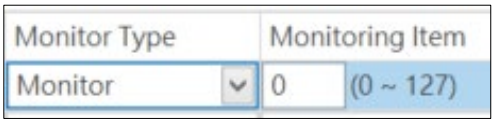
Channel Setting could be found in bottom part of **Monitoring Setting** area.

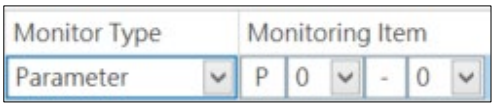


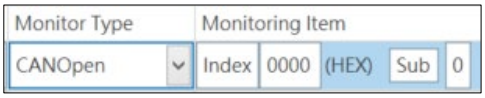
The following table describes the functions in the **Channel Setting** in **Servo Drive Oscilloscope** dialog.

Function	Description
Enable	Select the desire channel.
Capture Length	Parameter format are of two types, 16-bit and 32-bit. The standard setting of channel data is 16-bit. Thus, when you check 32-bit to expand the amount of one channel, ASDA-Soft automatically closes another channel to support the selected one.
Color	Change the color of data curve.
Monitor Type	Select the command source. The available options include: • Normal • Address • Variables • Parameters • CANopen **Note**: The default value is **Normal**.
Monitoring Item	Based on the selection of **Monitor Type**, this area shows different setting. **Normal** Directly access the built-in monitor items. Select the desired monitor item from the drop-down menu.  **Address** Enter the specific position format to access monitor variable, parameter and CANopen Object. Unit is hexadecimal.

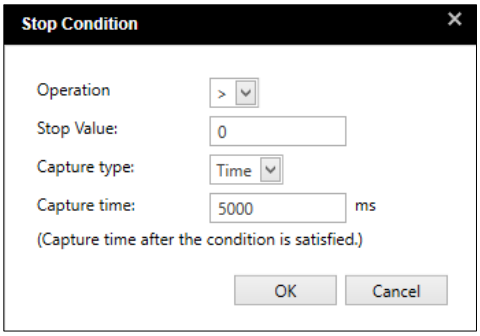
<table>
<tr><th>Function</th><th>Description</th></tr>
<tr><td></td><td>


The following table provides descriptions of the setting format.
<table>
<tr><th>Monitor Variable</th><th>Specify Parameter</th><th>CANopen Object</th></tr>
<tr><td>0x100000<u>XX</u>
XX = Monitor variable code
Range: 0x00h ~ 0x7Fh</td><td>0x20002<u>X</u> <u>YY</u>
YY = Parameter number
X = Parameter group
number</td><td>0x20<u>XX</u> <u>YYYY</u>
YYYY = Index
XX = Sub-Index</td></tr>
</table>
Example: Enter 0x01 to access 0x10000001, 0x2000212C to access parameter P1-44, and 0x200260C1 to access OB Index 60C1h, Sub 2.
Variables
Enter the specific monitor variable code. Unit is 0 ~ 127 in decimal. Refer to servo drive user manual for the monitor variable list.


Parameters:
Enter the desire monitoring parameter.


CANopen
Enter the desired access CANopen objects data. Enter the specified monitoring position, index, then specify the flag position, Sub-Index.
For example, if Index is set to 60C1h, Sub-Index is set to 02.


</td></tr>
<tr><td>Last Value</td><td>Indicate the value of selected point.</td></tr>
<tr><td>|ΔY|</td><td>Display the time interval of Y axis when enabling the function of vertical line measurement from Servo Drive Oscilloscope toolbar  .</td></tr>
<tr><td>Enable Checker</td><td>Enable the Checker Setting option of this channel.</td></tr>
</table>

Function	Description
Checker Setting	Click to display **Stop Condition** dialog.  **Operation** Select the operator from the drop-down menu.  **Stop Value** Enter the stop value. **Capture Type** Select from **Time** or **Points**. **Capture Time** Enter the capture time in display area after the condition is satisfied.

14.4.4 Inverter Device Trace / Oscilloscope

The inverter device status can be monitored in real-time through the inverter trace or oscilloscope, and the oscilloscope can support high-speed sampling of less than 1 ms.

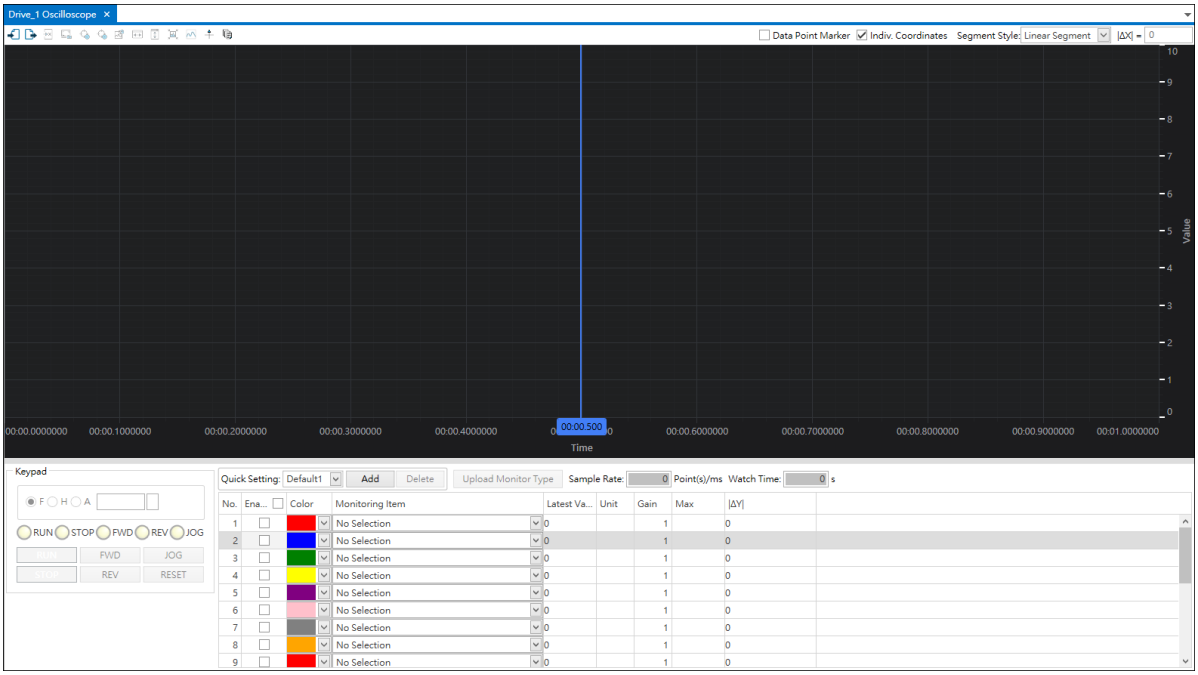
Note:

- Currently, MS300, MH300-HS, MX300, and VP3000 series models support the **Oscilloscope** function, and other models support the **Trace** function.
- When monitoring trace or oscilloscope, you can use parameter setting to read and write single parameter simultaneously.

To open trace or oscilloscope page

- Click **Inverter** > **VFD Function** > **Commissioning** > **Trace (Oscilloscope)** in the **Project Explorer**.

The inverter trace or oscilloscope view is shown in the following figure.



14.4.4.1 Toolbar

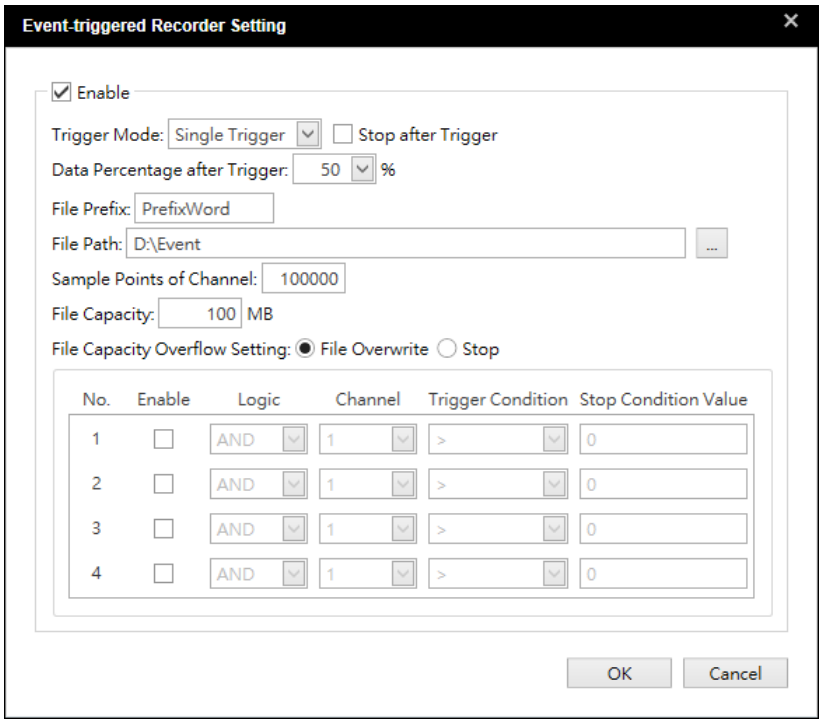
The following table describes the functions on the toolbar in the trace (oscilloscope).



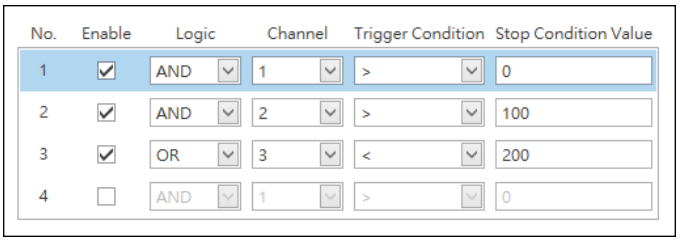
Icon	Description
	Click to open sampling setting and points.
	Click to save sampling setting and points in *.vfdscp file format.
	Click to export points in *.csv file format.
	Click to take a screenshot of the display area. (*.png, *.jpg & *.bmp).
	Click to start monitoring.
	Click to stop monitoring.
	Click to clear screen.

Icon	Description
	Click to display two vertical lines to measure the distance between waveforms.
	Click to display two horizontal lines to measure the distance between waveforms.
	Click to zoom fit, same as double-click the trend display area.
	Click to open the **FFT Analysis** dialog.
	Click to open the **Event-triggered Recorder Setting** dialog.
	Click to open the **Recorder Setting** dialog.

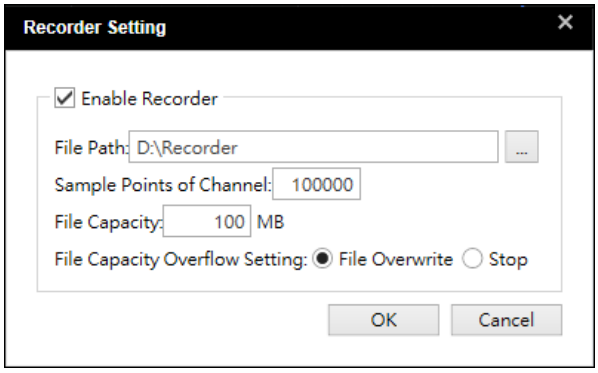
The following table describes the functions of the **Event-triggered Recorder Setting** dialog.



Function	Description
Enable	Click to enable the event-triggered recorder function.
Trigger Mode	Select the trigger mode. The default is **Single Trigger**. If **Single Trigger** is selected, the **Stop after Trigger** checkbox is available.

Function	Description
Data Percentage after Trigger	Set how much data before and after the condition will be recorded when meeting the trigger condition. For example, if selecting 20, the data of 20% after triggering and data of 80% before triggering will be recorded.
File Prefix	Set the default prefix to be added before the filename when saving the file.
File Path	Select the file path.
Sample Points of Channel	Set the sample points of single channel. The default value is 100000.
File Capacity	Set the file capacity the file path can save. The default value is 100.
File Capacity Overflow Setting	Set to overwrite the oldest file or stop recording when the file capacity exceeds the limit value.
Condition Setting	Set the trigger condition. For example, trigger the recorder when channel 1 value > 0 and channel 2 value > 100, or channel 3 value < 200. 

The following table describes the functions of the **Recorder Setting** dialog.

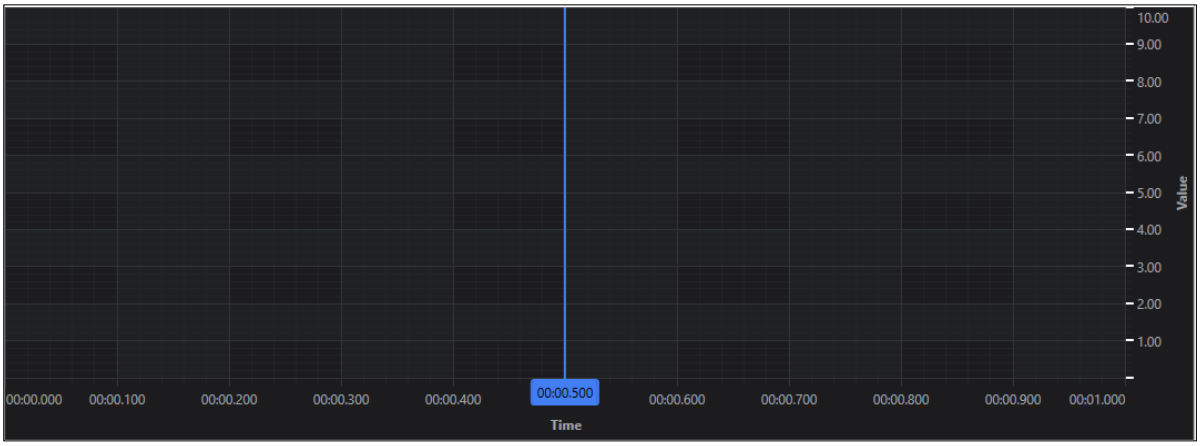


Function	Description
Enable Recorder	Check to enable the function.
File Path	Select file saving path.
Sample Points of Channel	Set sampling points of every channel.

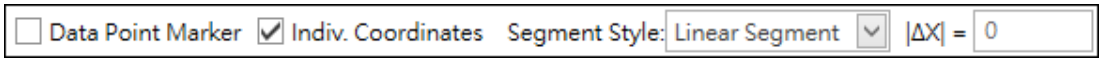
Function	Description
File Capacity	Set total file capacity limit of file path.
File Capacity Overflow Setting	Select overwrite of stop when the file capacity is overflow.

14.4.4.2 Display Area

Display the selected monitoring channel in real-time. You can hold down the right button of the mouse and drag display area presentation position, and use the mouse wheel to zoom in and out of the presentation range.

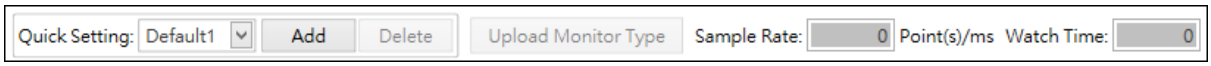


- The X axis represents time. The unit is sec.
- The Y axis represents the data, which corresponds to the selected channels. By default, each channel shares a Y axis, which is suitable for monitoring items with the same unit. However, if the unit of monitoring items are different, it is recommended to select **Indiv. Coordinates** checkbox on display area settings bar. After clicking a variable below, use the left mouse button to drag Y-axis coordinate axis to arbitrarily scale or adjust the up and down position of the variable sampling waveform. In this way, the relative position of each waveform on the screen can be adjusted.



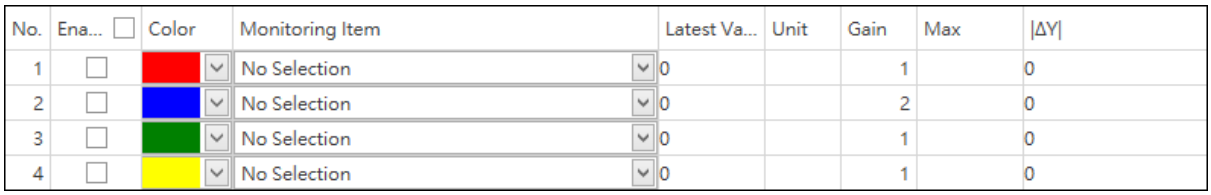
14.4.4.3 Monitoring Setting

The following table describes the functions of the monitoring settings bar.



Function	Description
Quick Setting	Select the set monitoring channel list.
Add	Click to add the selected monitoring channels to the list.
Delete	Click to delete the selected list.
Upload Monitor Type	Click to upload the content of monitoring items supported by the device. **Note**: Only applicable to the Oscilloscope function.
Sample Rate	Display the sample rate that the current monitoring channel can achieve. **Note**: Only applicable to the Oscilloscope function.
Point(s)/ms Watch Time	Display the length of time of the curve on the screen. The curve data that exceeds the time will be cleared. **Note**: Only applicable to the Oscilloscope function.

The following table describes the functions in the channel setting area.

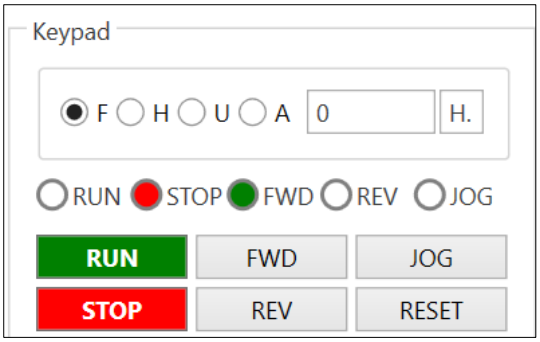


Function	Description
Enable	Select to display a channel.
Color	Select the channel color.
Axis Number	Select the axis number of the multi-drive inverter.
Table Number	Select the table number of the parameter group.
Monitoring Item	Select the built-in monitoring items or enter the keyword to filter the item.
Latest Value	Display the latest value of a channel.
Unit	Change the corresponding unit according to monitoring items.
Gain	Move the waveform position according to gain.
Max	Display the maximum value of the waveform.
|ΔY|	Display the spacing between Y axis and measuring lines. On the toolbar, click  to enable this setting.

14.4.4.4 Keypad

Control inverter, adjust speed, monitor status, and read / write parameters through keypad. To adjust speed and start / stop inverter, set frequency command source and operation command source parameters to be controlled by communication. Take VFD-C2000 Plus as an example. Set parameters P0.20 and P0.21 as 1 and 2 respectively. Parameters vary from device model. For more information, refer to the corresponding device model documents.

The following table describes the functions in the **Keypad** setting area.



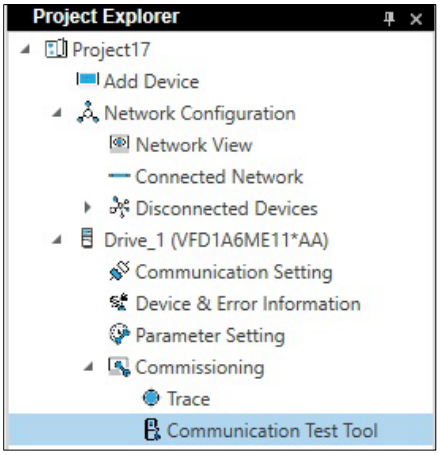
Function	Description
F	Display command frequency. Enter command frequency in the text field.
H	Display output frequency.
U	User-defined. Display DC-BUS voltage by default.
A	Display output current.
RUN / STOP	Click to run or stop the inverter.
FWD / REV	Click to forward or reverse the operation of inverter.
JOG	Click to perform the jog.
REST	Click to clear the error code.

14.5 Communication Test Tools

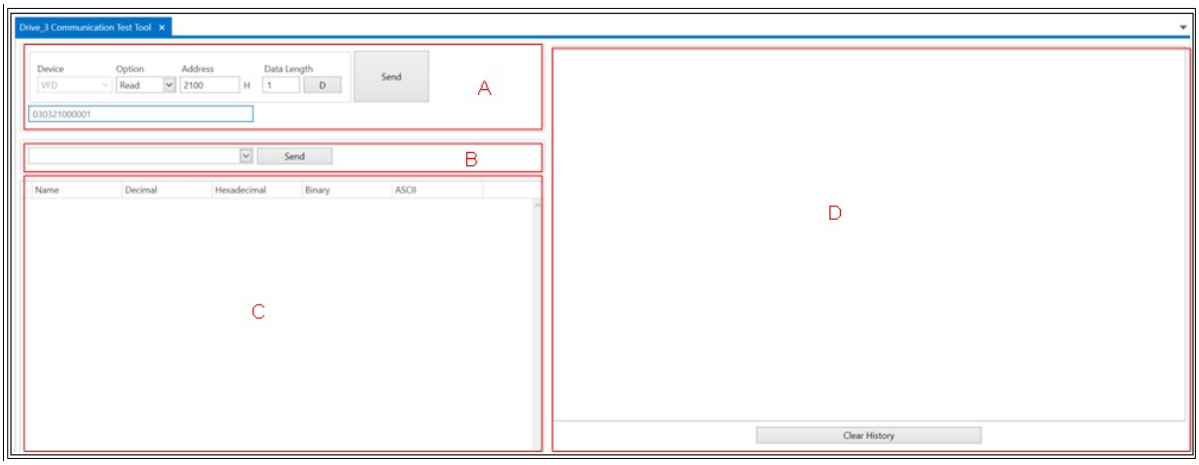
A command transceiver is based on Modbus communication. During testing and commissioning, it can quickly read / write the related data of inverter (or the built-in

PLC) through Modbus communication. Before executing the function, make sure that the communication setting of the device has been successfully connected.

Navigate **Inverter Device** (for example, MH300) > **VFD function** > **Commissioning** > **Communication Test Tool**.



The following figure shows **Communication Test Tool** tab.



The view of **Communication Test Tool** includes the following area.

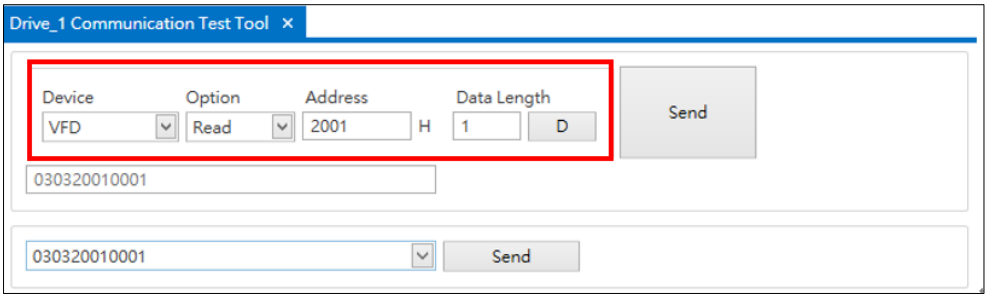
1. Communication format setting (form)
2. Communication format setting (input box)
3. Data reading result
4. Delivering history

The following table describes the functions of the **Communication Test Tool** area.

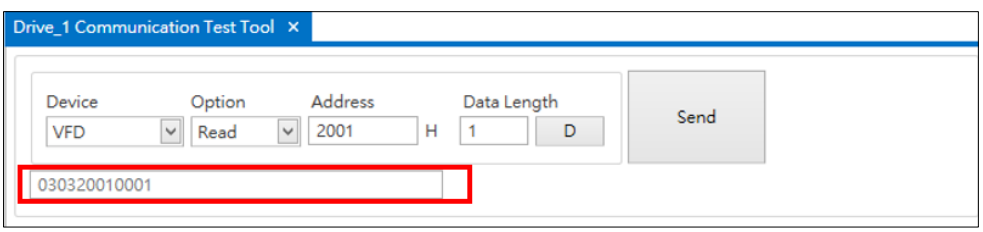
Function	Description
Device	Select a device that you want to read / write. Choose this inverter or its built-in PLC. For choosing PLC, check the **Enable Built-in PLC Connection** box. • VFD • PLC
Option	Choose a command to the device, • Read • Write
Address	Hex. address of read / write Modbus
Data Length	When selecting “read the address data”, assign the data length.
Data (Text box)	When selecting “write the address data”, assign the data content.

Example: Refer to the following steps to perform the read / write testing of frequency command functions to VFD 2001H address.

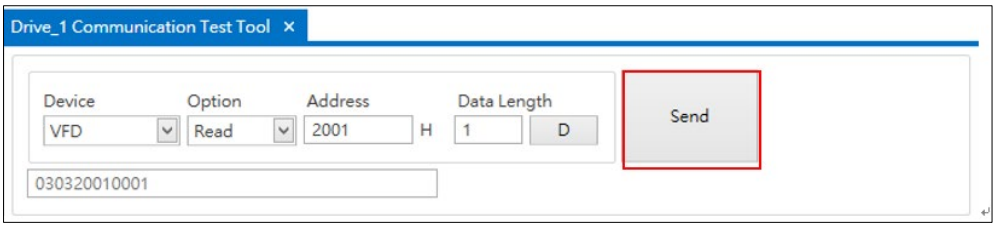
- Device: VFD
  Option: Read
  Address: 2001
  Data Length: 1



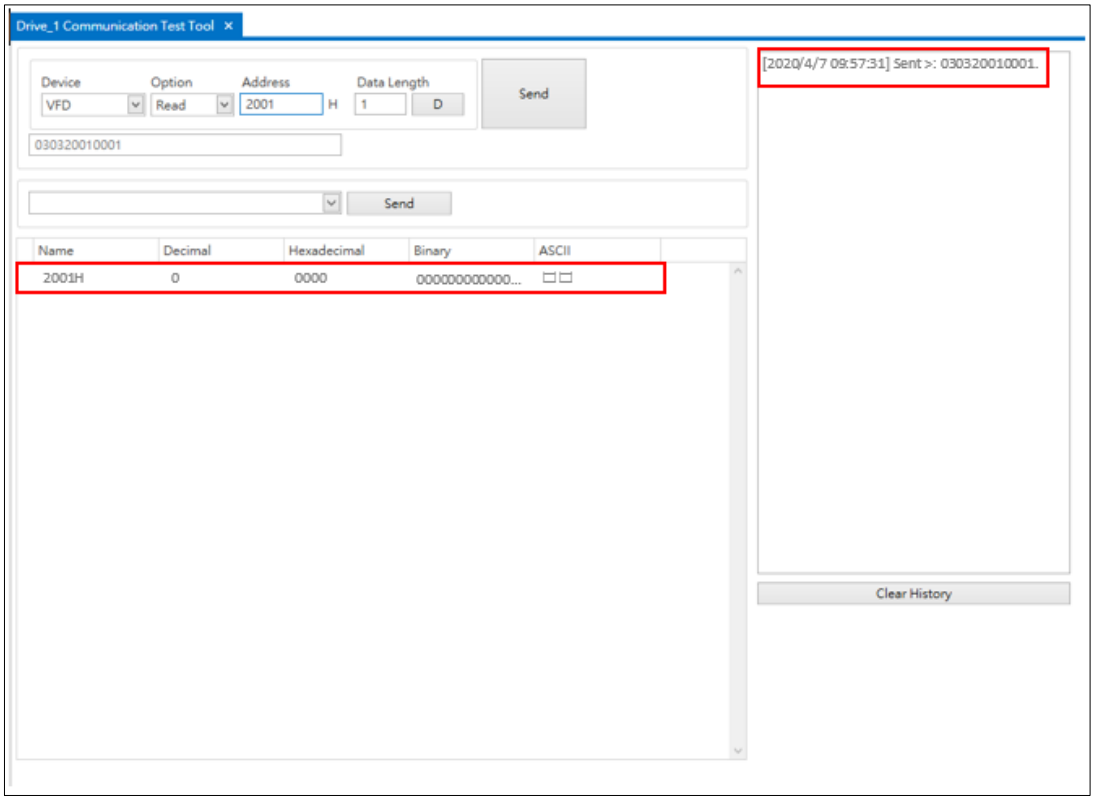
1. After filling in the columns, you can see the Modbus command value in the text box.



2. Confirm the value and click **Send**.



3. Data reading result shows the 2001H address content value (currently 0) in different ways. On the right, the delivering history block shows the delivery record.



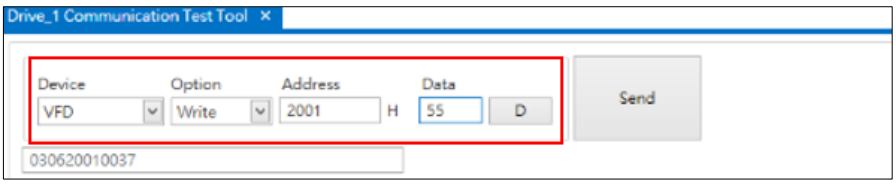
4. Change the 2001H address content to 55.

- Device: VFD

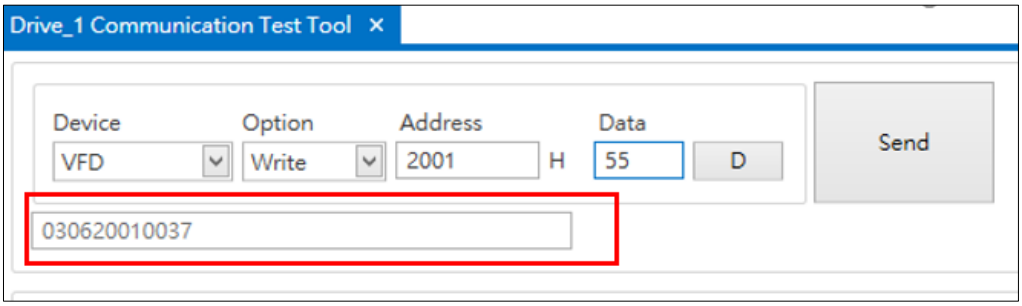
  Option: Write

  Address: 2001

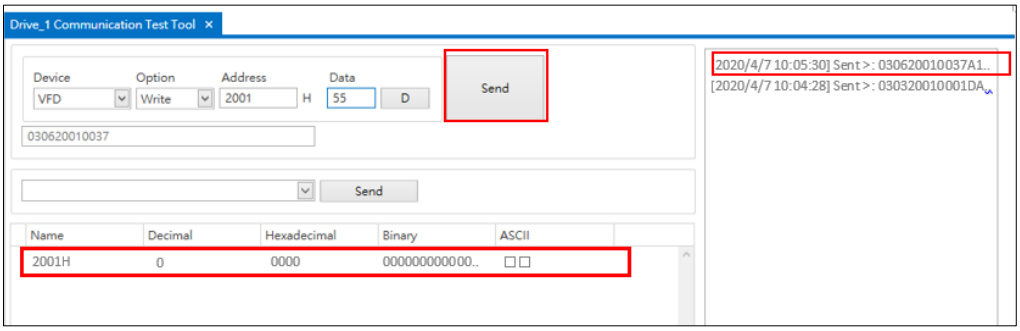
  Data Length: 55 (D: decimal) or 37 (H: hex)



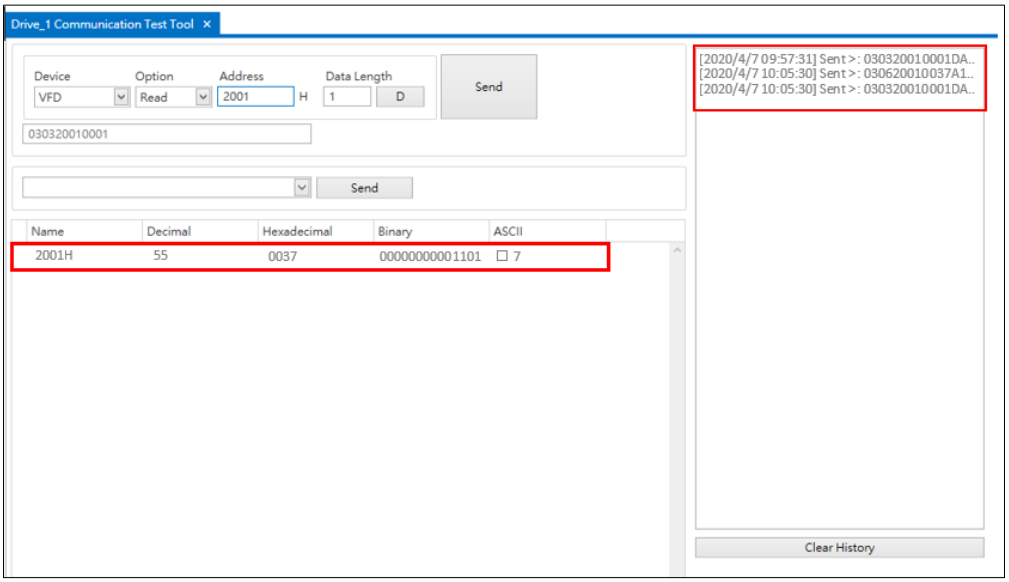
5. After filling in the columns, you can see the Modbus command value in the text box.



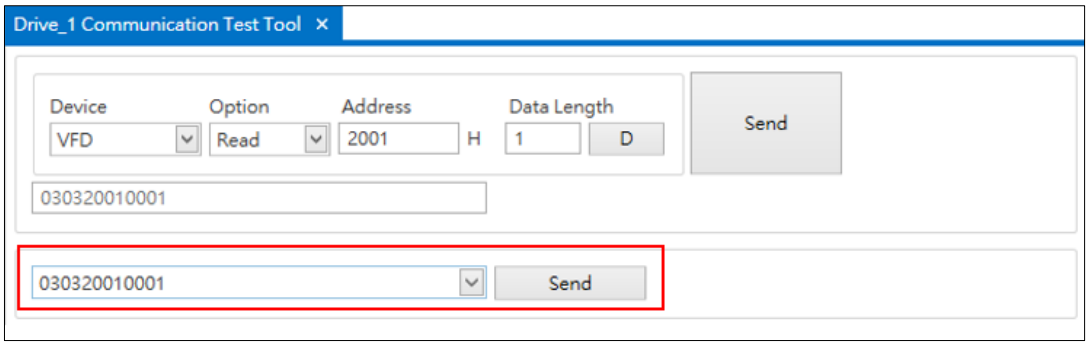
6. Confirm the value and click **Send**. A new delivering record will be added on the right.



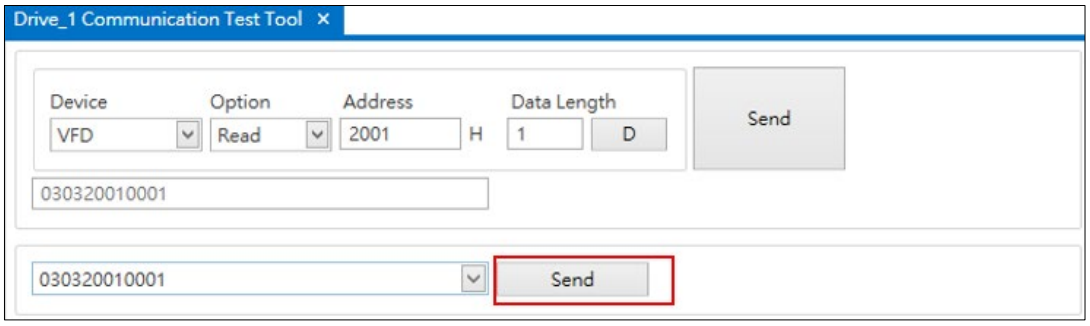
7. Repeat Step 1 to Step 4, and make sure the current 2001H content value is 55, which represents successful write.



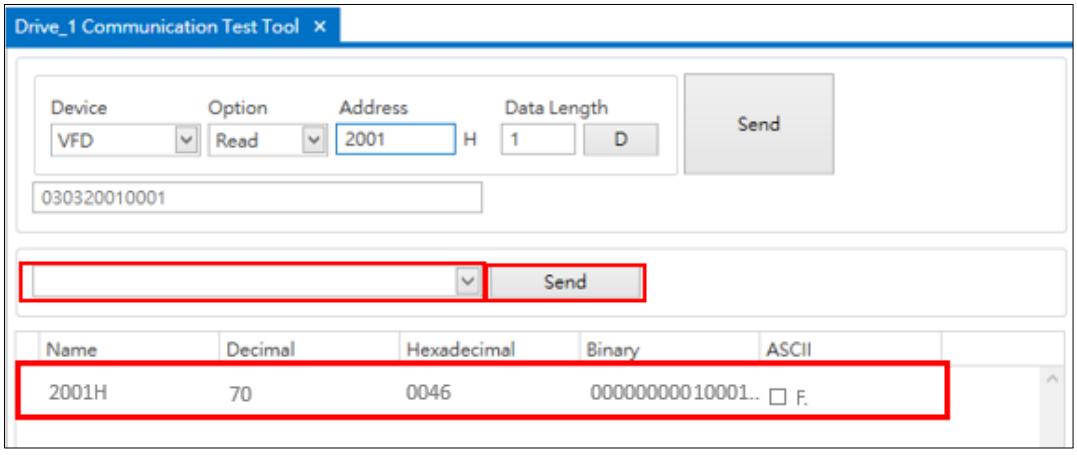
8. To write value 70 (decimal) to 2001H, change the column value or directly enter 030320010001 in the second text box, or double-click a record in the delivery history block. The record will be loaded to the text box.



9. Confirm the value and click **Send**.



10. Enter 030320010001, as current 2001H content value, in the second text box. The value 70 represents success.

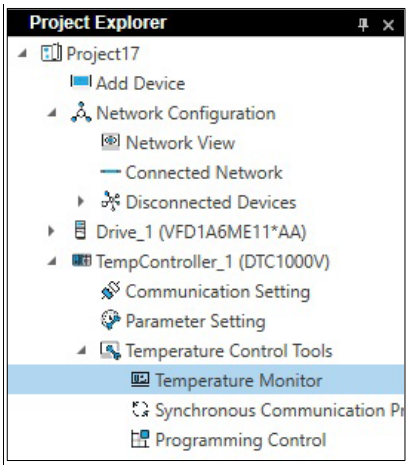


Chapter 15: Temperature Control Tool

15.1 Monitor

Use temperature monitor to observe the set temperature and the current temperature value through the monitoring screen, and can also directly edit the control method of the temperature controller online and adjust the related temperature control parameters.

This function can be found in **Temperature Control Tools** > **Temperature Monitor**.

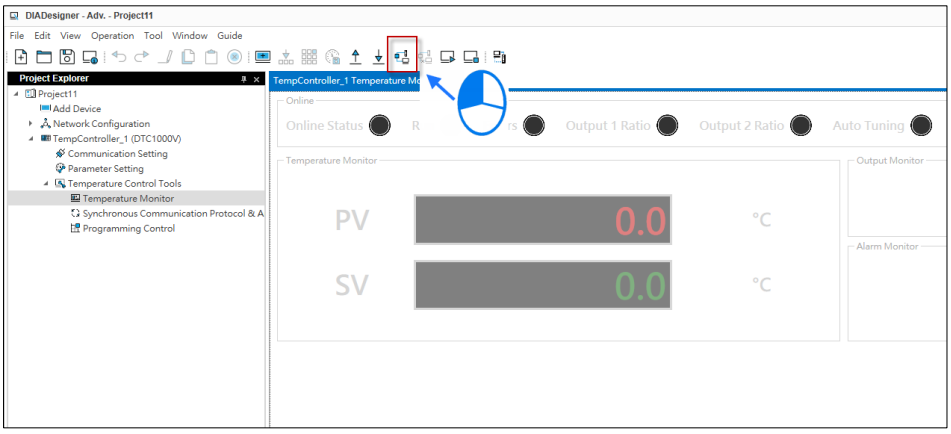


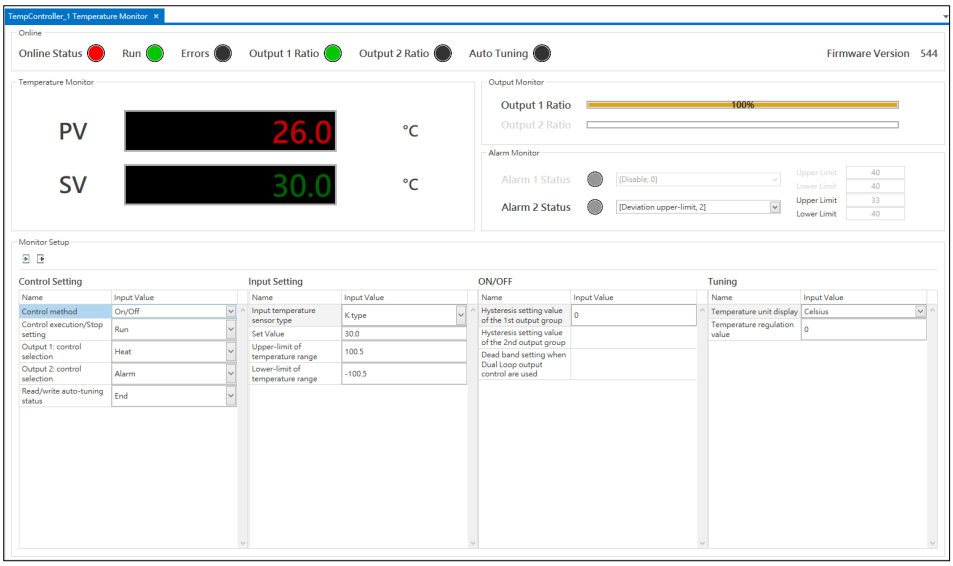
15.1.1 Turn on the Temperature Monitor Function

Follow these steps to enable **Temperature Monitor** function of temperature controller.

To enable Temperature Monitor

1. Open **Temperature Controller** > **Temperature Control Tools** > **Temperature Monitor,** and double-click **Temperature Monitor**.

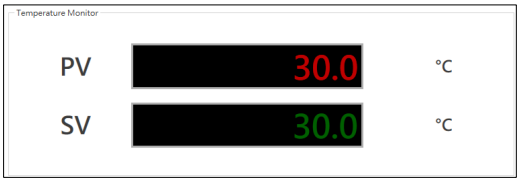




15.1.2 Temperature Monitor Dialog

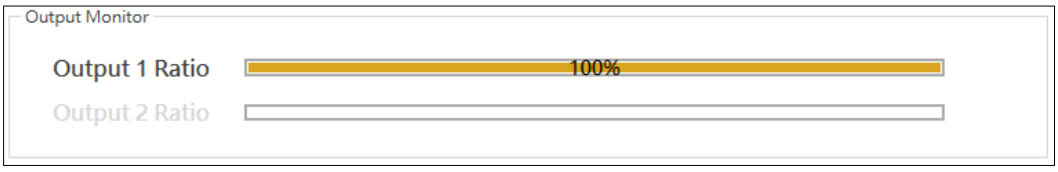
1. Temperature Monitor

   You can see the temperature measured by the current device's temperature controller, referred to as PV (Present Value), and the set target temperature, referred to as SV (Setpoint Value) here.



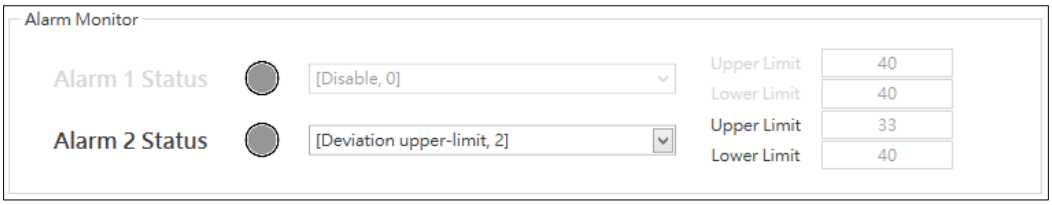
2. Output Monitor

   Set the output type, and use Output Monitor block to know the current heating / cooling output status or output percentage.



3. Alarm Monitor

   Set the alarm mode through the drop-down menu and also set the upper / lower temperature limit of the alarm, and learn the current alarm status of the device from this **Alarm Monitor** area.



15.1.3 Monitor Setup

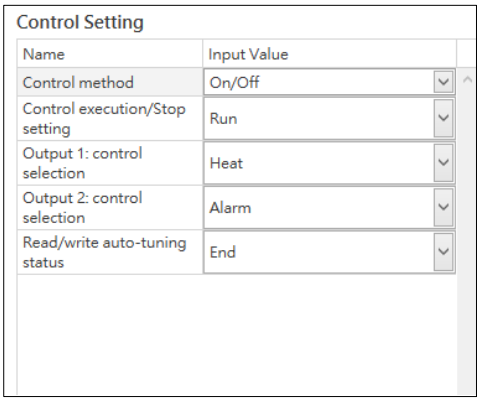
The following figure shows Monitor Setup toolbar.



The following table describes the functions of the Monitor Setup toolbar.

Icon	Description
	Parameter import function. After clicking, an **Open** dialog appears to select the parameter file to import.
	Parameter export function. After clicking, the **Save As** dialog appears to select the storage location where the parameter file is expected to be exported.

- Control Setting

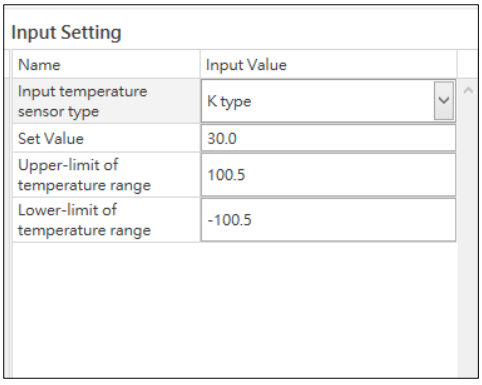


The following table describes the functions in the **Control Setting** area.

Function	Description
Control method	1. PID: Through PID control parameters such as proportional, integral and derivative, it can output precise temperature control. 2. On / Off control: switch control, switch control when the temperature reaches the set

Function	Description
	condition. 3. Manual control: Set the temperature control with a fixed output. 4. PID programmable: for user programmable PID temperature control.
Control execution / Stop setting	1. Implementation 2. Run 3. Stop 4. The Program stops 5. Program Pause
Output 1 – control selection	1. Heating 2. Cooling 3. Alarm 4. Proportional output (applicable to equipment models C and L)
Output 2 – control selection	1. Heating 2. Cooling 3. Alarm
Read / write auto-tuning status	1. Auto tuning start: When switching to start, the AT light will flash and PID control parameters will start to be calculated. After tuning is completed, it will automatically switch to "Auto tuning stop". 2. Auto tuning stops

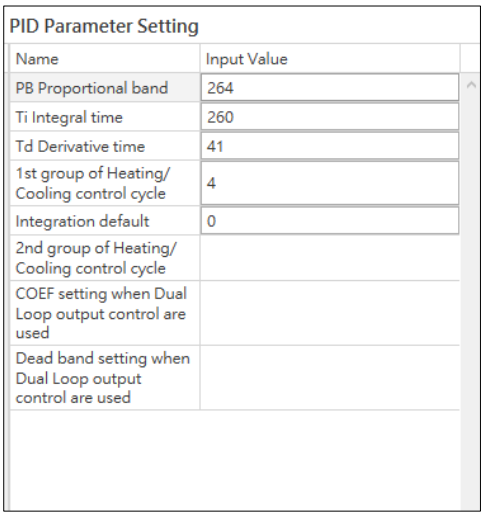
- Input Setting

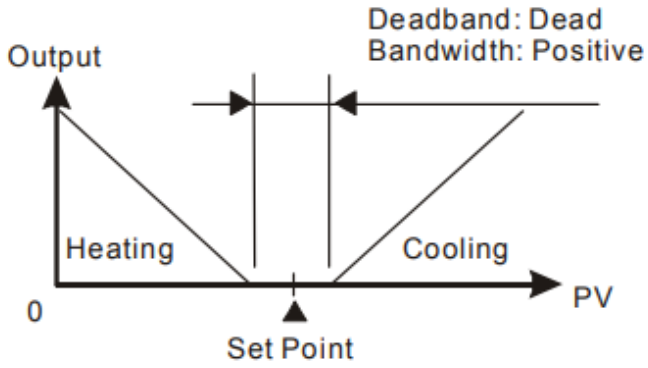
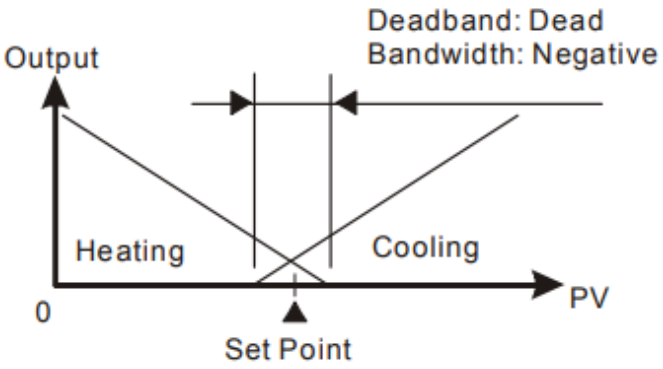


Function	Description
Input Temperature sensor type	1. Thermocouple pair K type 2. Thermocouple pair J type

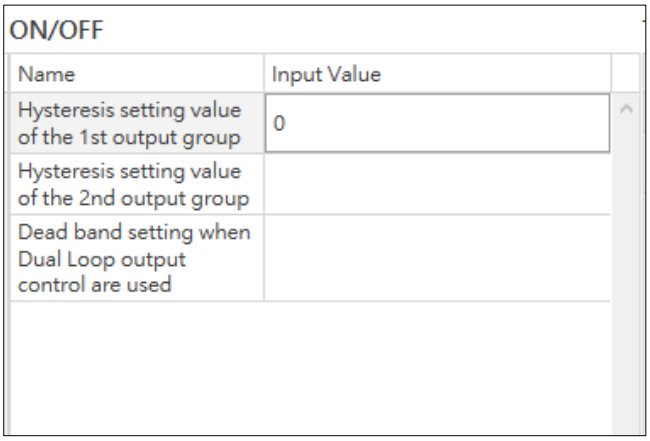
Function	Description
	3. Thermocouple pair T type 4. Thermocouple pair E type 5. Thermocouple pair N type 6. Thermocouple pair R type 7. Thermocouple pair S type 8. Thermocouple pair B type 9. Thermocouple pair L type 10. Thermocouple pair U type 11. Thermocouple TXK type 12. Platinum resistance thermometer JPt100 13. Platinum resistance thermometer Pt100 14. 0~5V linear voltage input 15. 0~10V linear voltage input 16. 0~20mA linear current input 17. 4~20mA linear current input 18. 0~50mV linear voltage input
Set Value	Target temperature setting value
Enter the upper-limit of temperature range	SV set value upper limit
Enter the lower-limit of temperature range	SV set value lower limit

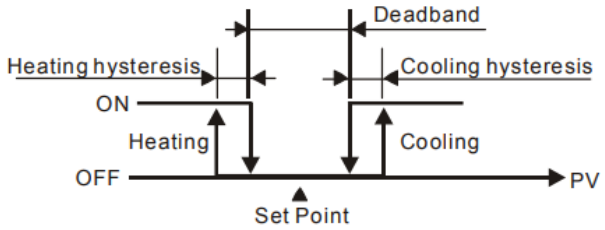
- PID Parameter Setting



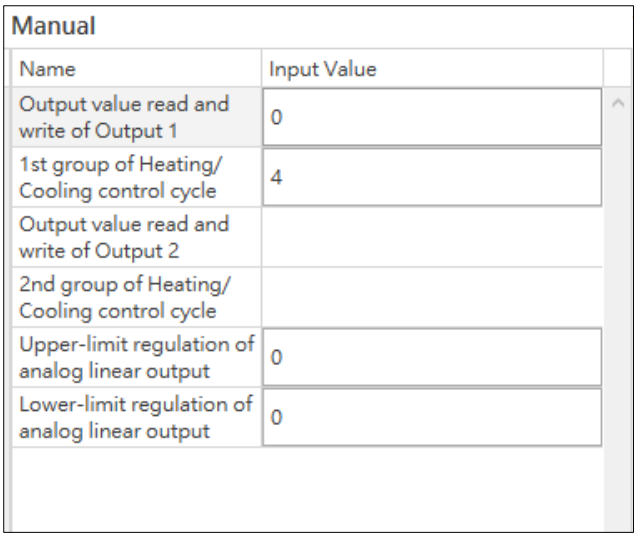
Function	Description
PID setting value	1. PB proportional band setting value 2. Ti integral value constant set value 3. Td differential control constant setting value 4. Preset value of integration 5. COEF setting for dual output **Note**: PID setting value is recommended to be calculated by auto tuning function.
Control cycle	Control cycle time, unit (seconds). **Example**: If the output is 60%, the control cycle time is 10 seconds The operation output time occupies 6 seconds, and the stop operation time is 4 seconds.
Dead band setting when dual output	1. Dead band is positive  2. Dead band is negative 

- ON/OFF



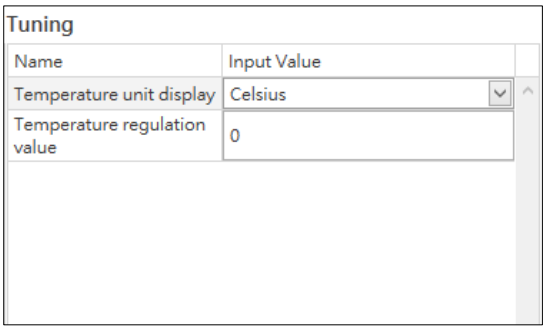
Function	Description
Hysteresis setting value	If it is higher than SV setting value + hysteresis setting value or lower than SV setting value - hysteresis setting value, heating / cooling gets activated.
Dead band setting when dual output	

- Manual



Function	Description
Output read and write	Set the percentage of output.
Control cycle	Control cycle time, unit (seconds). **Example**: If the output is 60%, the control cycle time is 10 seconds. The operation output time occupies 6 seconds, and the stop operation time is 4 seconds.
Analog linear output upper limit adjustment / offline adjustment	Output upper / lower limit scale adjustment, 2.8 μA (1.3 mV).

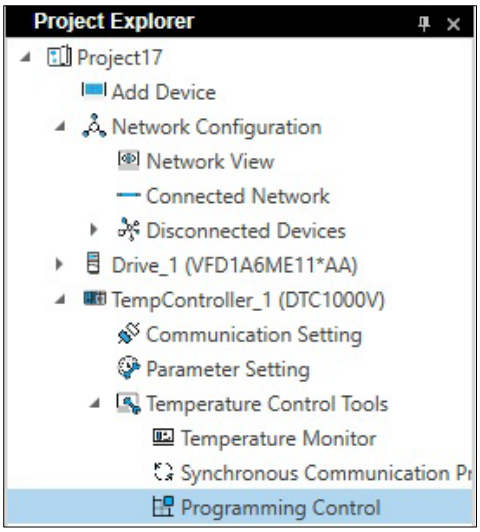
- Tuning



Function	Description
Temperature unit display	1. Fahrenheit 2. Celsius
Temperature regulation value	Used to adjust the error temperature.

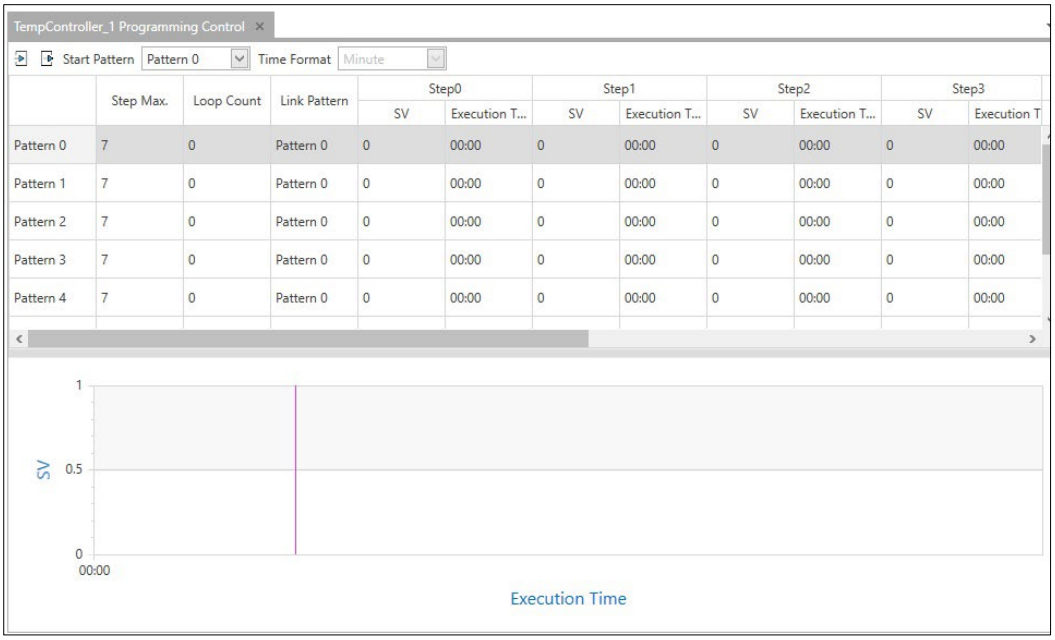
15.2 Program Planning

Plan the temperature control program in the program planning function. Locate program planning function through **Temperature Controller** > **Temperature Control Tool** > **Programming Control**.



Double-click **Programming Control** to open **Programming Control** dialog.

In **Programming Control** dialog, the upper program editing area can be divided into 8 programs (Pattern 0 ~ 7). Each program has 8 steps (Step 0 ~ 7), and each has a link program (Link Pattern), execution loop (Loop Count), and effective step (Step Max.). The lower status display area displays the execution curve after program editing.



The following figure shows toolbar of **Programming Control** dialog.



The following table describes the functions of the Programming Control toolbar.

Function / Icon	Description
	Import function. After clicking, an **Open** dialog appears to select the programming control file to import.
	Export function. After clicking, the **Save** dialog appears to select the storage location where the programming control file is expected to be exported.
Start Pattern	Select the start pattern from the drop-down list.
Time Format	Depending on the model, different time formats of SV can be set. DTC only supports the time format as minutes.

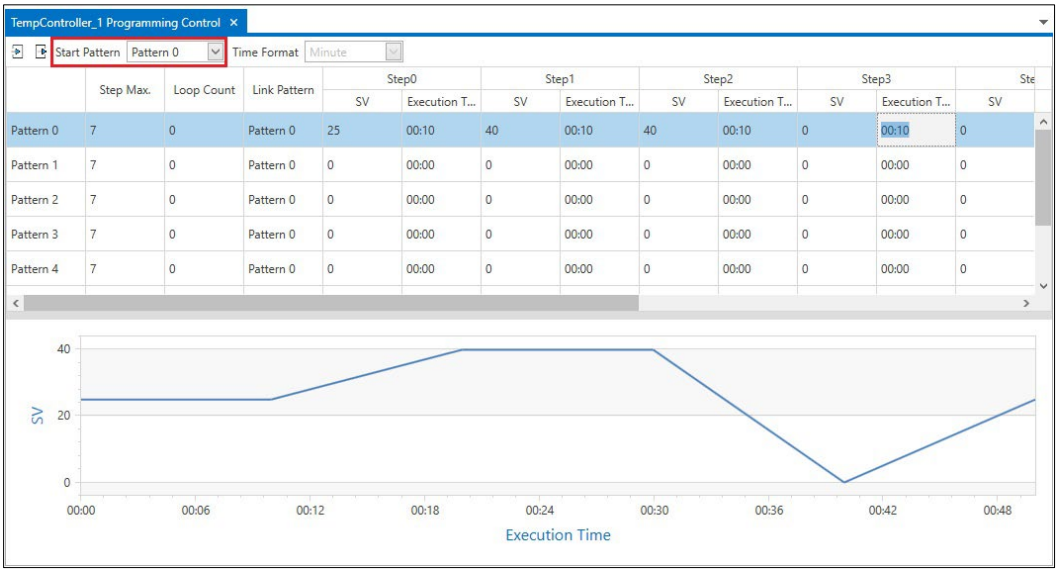
15.2.1 Program and Step Editing

Follow these steps to edit program.

To edit program

1. Set the steps: In the program of initial **Start Pattern** program setting, set the steps, including the setting of two parameters, SV and execution time, which means that the PV value must reach the SV setting value after the execution time.

   The lower area display diagram shows the temperature control curve after program editing, and the screen is shown in the following figure.

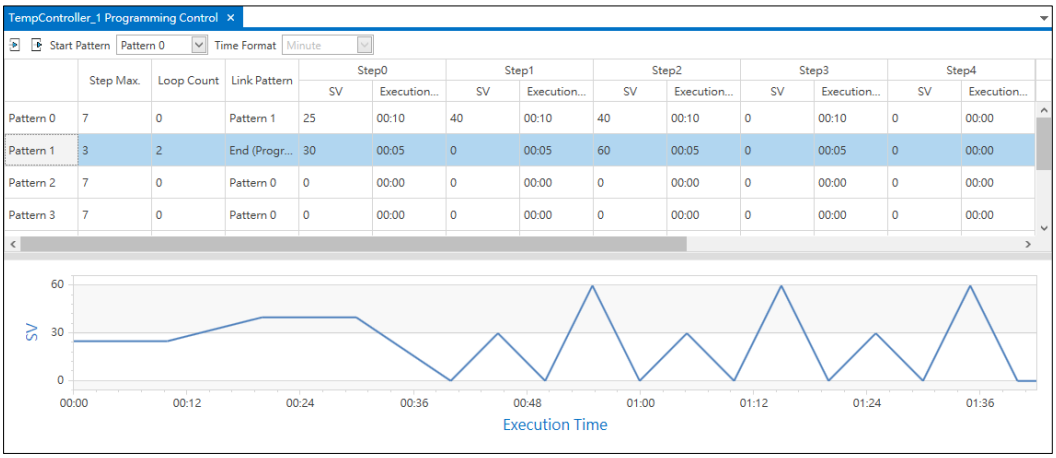


2. Set the **Link Pattern**: It is the pattern that executes after the program is executed. If it is set to a value other than program 0 ~ 7, it means that the program ends after the program is executed, and the target value remains at the setting of the last step.

3. Set **Loop Count**: It is the number of additional loops that the program executes. For example, if this parameter item is set to 2, it means that the program needs to be executed two additional times, even the original one needs to be repeated three times in total.

4. **Effective Step Max**: It is the number of steps executed by each pattern (can be set from 0 to 7). For example, if this item is set to 3, it means that the corresponding style executes step 0 ~ step 3, and the remaining steps does not execute and cannot be edited also.

   The lower status display diagram shows the temperature control curve after program editing, and the screen is shown in the following figure.



15.2.2 Status Display Area

It can display the effective program temperature control curve after programming control. The horizontal axis is the execution time and the vertical axis is SV. The maximum coordinate value changes automatically according to the maximum time and temperature set by the program. Move the mouse to any position to display the target temperature and estimated execution time at that point, as shown in the following figure.



15.3 Station Number Automatic Synchronization

Use **Station Number Automatic Synchronization** function to connect all the devices in series. After the function is turned on, the communication specifications and station numbers of the serially connected devices will be directly changed, and the station number setting will be based on the station number of the first station.

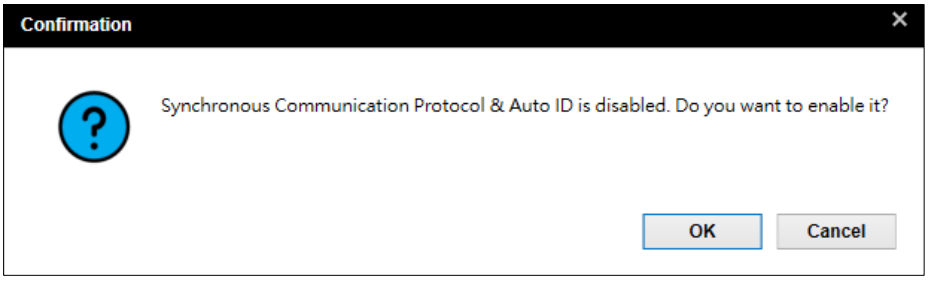
This function can be enabled or disabled in **Temperature Control Tool** > **Automatic Synchronization of Communication and Station Number**.

15.3.1 Activate Station Number Automatic Synchronization Function

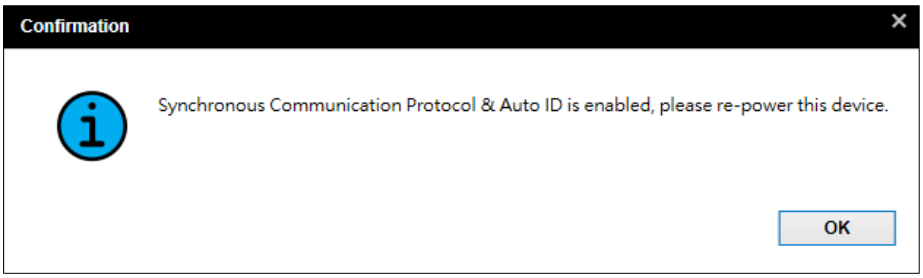
Follow these steps to activate **Station Number Automatic Synchronization** function.

To activate Station Number Automatic Synchronization function

1. Click **Temperature Control Tool** > **Synchronous Communication Protocol & Auto ID** and station number to open **Confirmation** dialog.



2. Click **OK**. A dialog that has successfully opened the automatic synchronization function appears, and notify you to restart power.



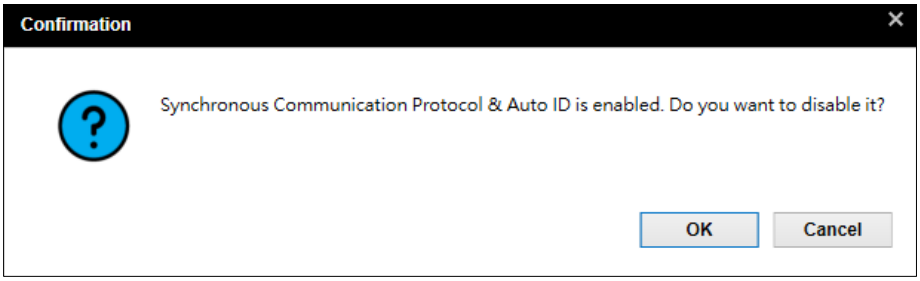
3. Click **OK** to turn power on again.

15.3.2 Deactivate Station Number Automatic Synchronization Function

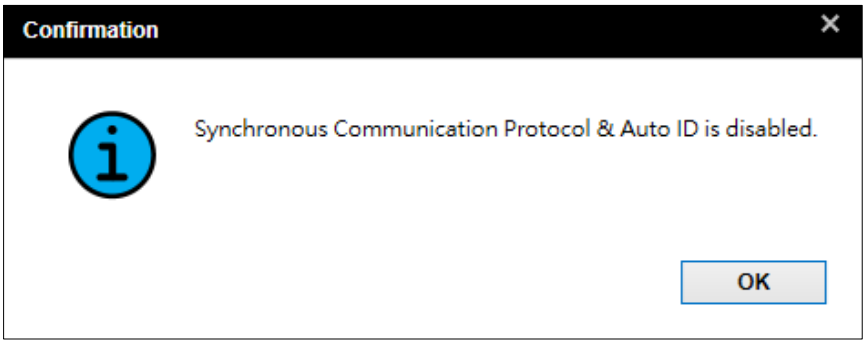
Follow these steps to deactivate **Station Number Automatic Synchronization** function.

To deactivate Station Number Automatic Synchronization function

1. Click **Temperature Control Tool** > **Synchronous Communication Protocol & Auto IDB,** and station number to open a Confirmation dialog.



2. Click **OK**. A dialog with the automatic synchronization function closed appears.

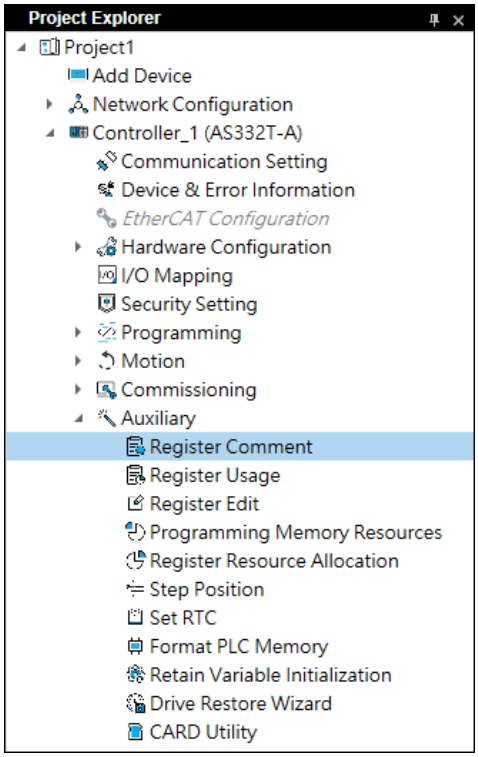


3. Click **OK** to close automatic synchronization function.

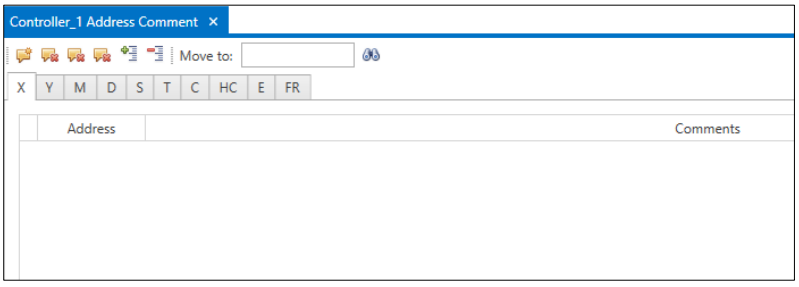
Chapter 16: Auxiliary Tools

16.1 Register Comment

Manage the comments for registers with the help of **Register Comment**. Locate it in **Controller** > **Auxiliary** > **Register Comment**.



Double-click **Register Comment** to open **Register Comment** dialog. It displays the register names with corresponding comments in the Address and Comments columns respectively.



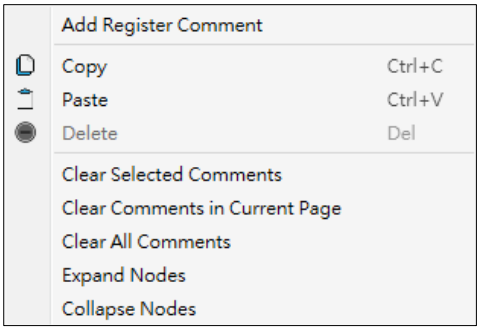
The following figure shows the toolbar of **Register Comment** dialog, which helps to manage the register comments.



The following table describes the functions of the toolbar in **Register Comment** dialog.

Function	Description
	Click to add Register Comment. Displays the **Add Register Comment** dialog.
	Click to clear comments of selected items.
	Click to clear comments of all items in the page.
	Click to clear comments of all items.
	Click to import CSV file.
	Click to export current tab to CSV file.
	Click to expand nodes.
	Click to collapse nodes.
	Enter a device name to locate.
	Click to locate the register whose name is entered in the **Move to** field.

Right-click in the **Register Comment** dialog to open the context menu.



The following table describes the functions in context menu in **Register Comment** dialog.

Function	Description
Add Register Comment	Click to add Register Comment. Display the **Add Register Comment** dialog.
Copy	Click to copy the selected item.
Paste	Click to paste the selected item.

Function	Description
Delete	Click to delete the selected item.
Clear Selected Comments	Click to clear comments of selected items.
Clear Comments in Current Page	Click to clear comments of all items in the page.
Clear All Comments	Click to clear comments of all items.
Expand Nodes	Click to expand nodes.
Collapse Nodes	Click to collapse nodes.

16.1.1 Adding Items in Register Comment Dialog

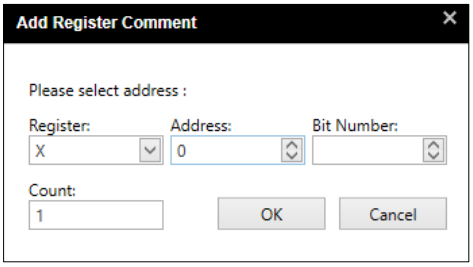
Follow these steps to add items in the **Register Comment** dialog.

To add items in Register Comment dialog

1. Click  on toolbar.

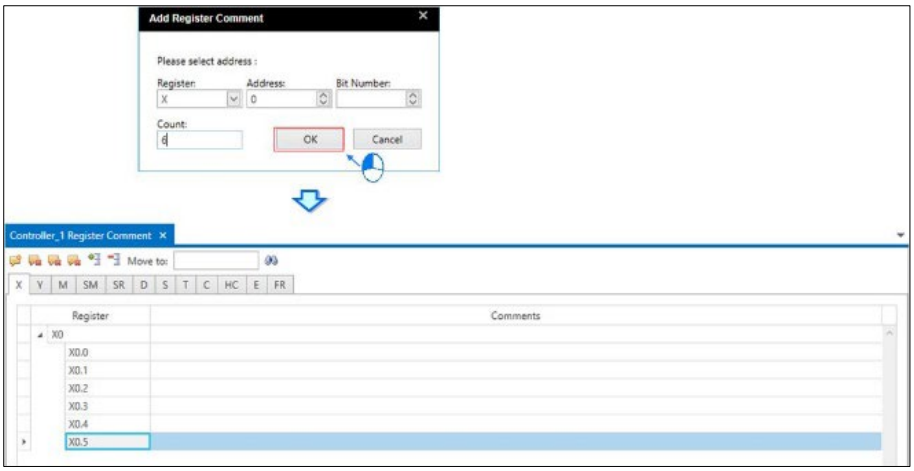
   Or

   Right-click in the **Register Comment** dialog to open context menu, and click **Add Register Comment** to open **Add Register Comment** dialog. You can also double-click the empty area of Register Comment to add.



2. Select **Register**, **Address**, **Bit Number**, and **Count** of items needed.
3. Click **OK** to add registers to the **Register Comment** dialog.

Example: The following figure shows an example of adding X Registers from 0.0 to 0.5.



Enter the comments for respective registers in the **Comments** column.

16.1.2 Deleting Items in Register Comment Dialog

Follow these steps to delete items in the **Register Comment** dialog.

To delete items in the Register Comment dialog

1. Select the item to be deleted.
2. Right-click register and address items that need to be deleted to open its context menu and select **Delete**.

   Or

   Press **Delete** key in the keyboard.

16.1.3 Deleting Register Comments

Follow these steps to delete Register Comments.

To delete Register Comments

1. Select the Register(s).
2. Click on toolbar.

   Or

   Right-click the Register and Address whose comments has to be deleted, and click **Clear Selected Comments** on the context menu.

16.1.4 Deleting Comments for Selected Register Type

Follow these steps to delete comments of all Register Addresses in a Register type.

To delete comments of Register Addresses in a Register type

1. Select the Register type (example X, Y, M etc.).
2. Click  on icon toolbar.

   Or

   Right-click in the **Register Comment** dialog, and click **Clear Comments in the Current Page** in the context menu.

16.1.5 Deleting All Comments

Follow these steps to delete the comments for all Addresses in the Register Comments.

To delete comments for all Addresses in Register Comments

1. Click  on icon toolbar.

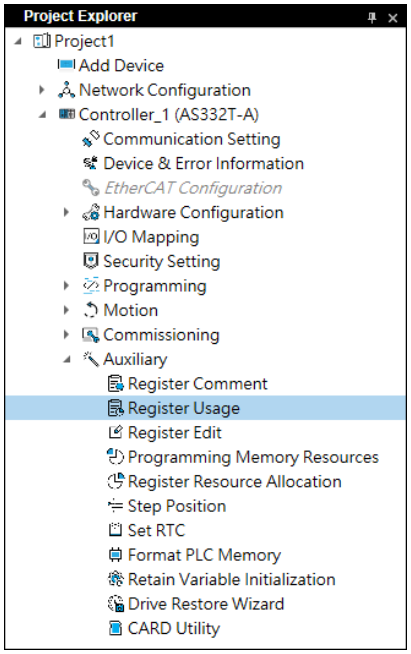
   Or

   Right-click in the **Register Comment** dialog, and click **Clear All Comments** in the context menu.

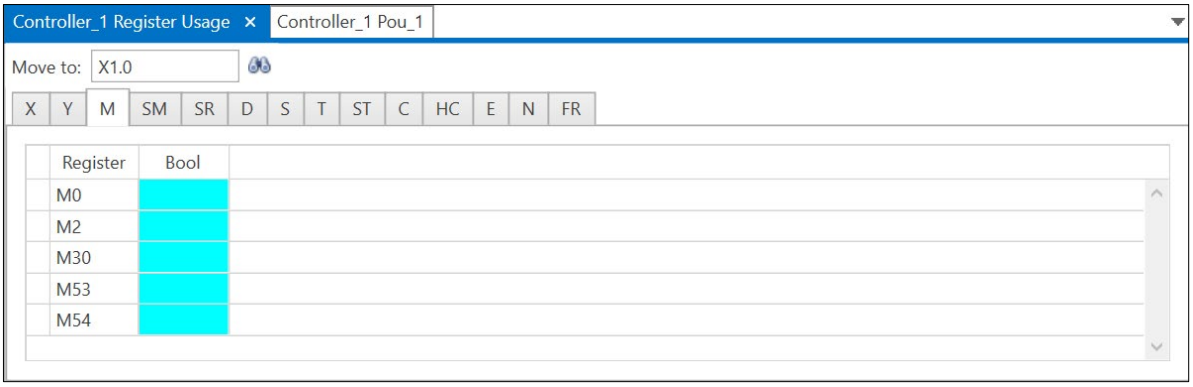
16.2 Register Usage

Refer to the usage of Registers in the project with the help of **Register Usage** in DIADesigner. Locate Register Usage in **Controller** > **Auxiliary** > **Register Usage**.

Double-click **Register Usage** to open **Register Usage** dialog.



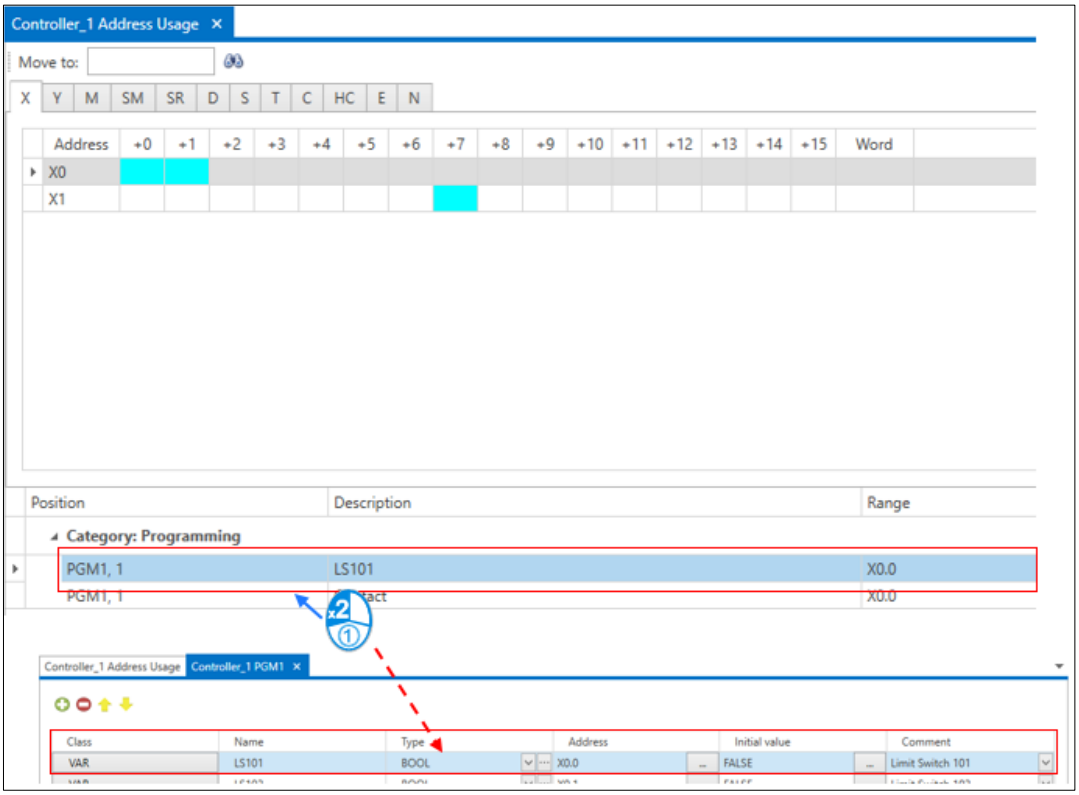
The **Register Usage** dialog consists of a toolbar and two panes – top and bottom. Toolbar has a **Move to** function to locate the particular register. Top pane displays the register and addresses. Bottom pane displays the locations where they are used. The addresses which are used in the POU, Global Variable – Default Tag Table, etc. display in cells filled with blue color.



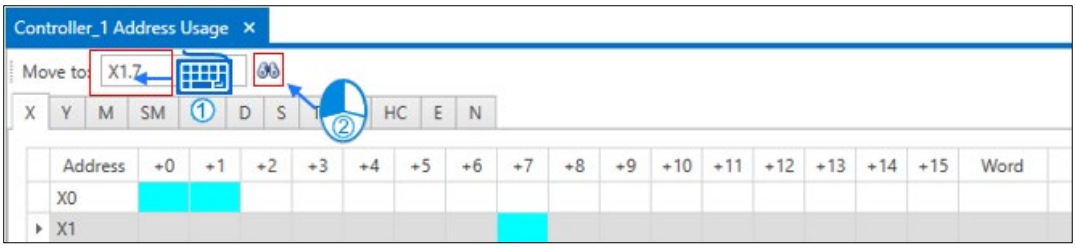
Example: The addresses – X0.0, X0.5 in the following figure are used. Click the blue color cells; based on the usage, the bottom panel displays the locations where the addresses are used. The following figure shows an example of address X0.0 usage location.



Click any address (blue-color cells) in the upper pane of **Register Usage** and usages of this register appears at the bottom pane. Double-click a line in the bottom pane to navigate to the respective location. The following figures shows a sample navigation for Register X with Address 0.0 in POU PGM1.



Locate a particular address in **Register Usage** dialog with the help of **Move to** function located on the toolbar. Enter an address and click  to navigate to that address. The following figure shows an example of navigation to X1.7.

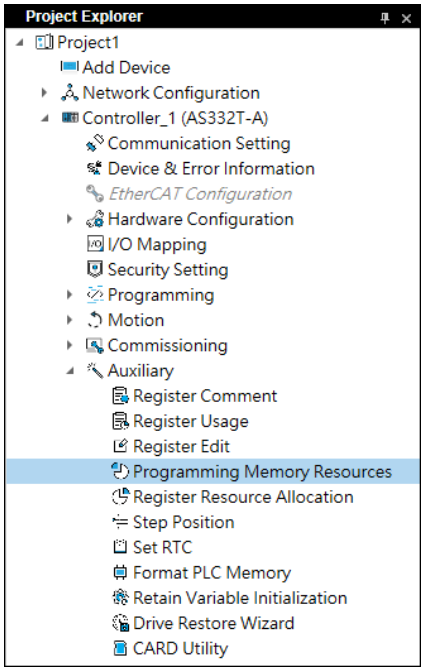


Note:

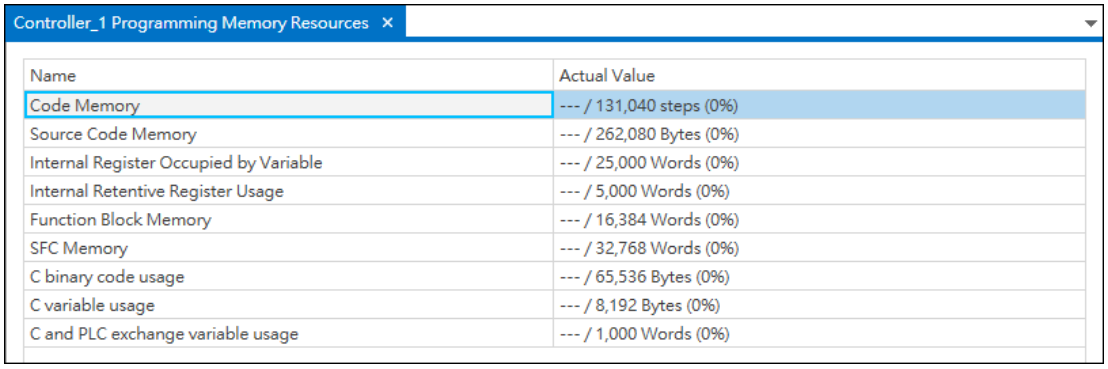
- If a block operation instruction (such as ZRST) is used, the register address range in the block will be considered occupied.
- The register usage only counts the usage condition in the program. The settings of other configuration functions, for example, the register addresses occupied by the data exchange table will not be counted.

16.3 Programming Memory Resources

Keep a check on the memory usage of the controller device with the help of Programming Memory Resources. Locate the Resources in **Controller** > **Auxiliary** > **Programming Memory Resources**.



Double-click **Programming Memory Resources** in **Project Explorer** to open the **Programming Memory Resources** dialog.

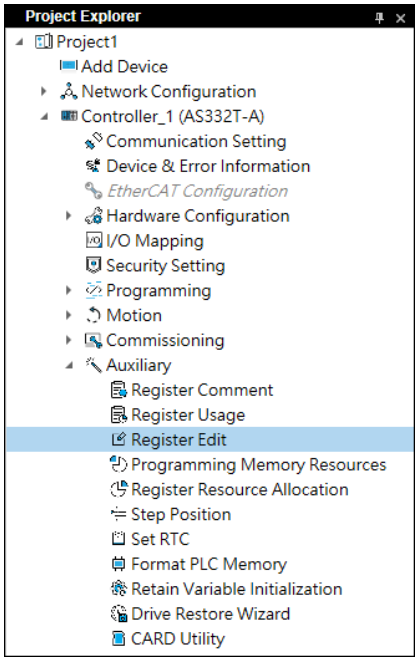


The following table describes the functions in **Programming Memory Resources** dialog.

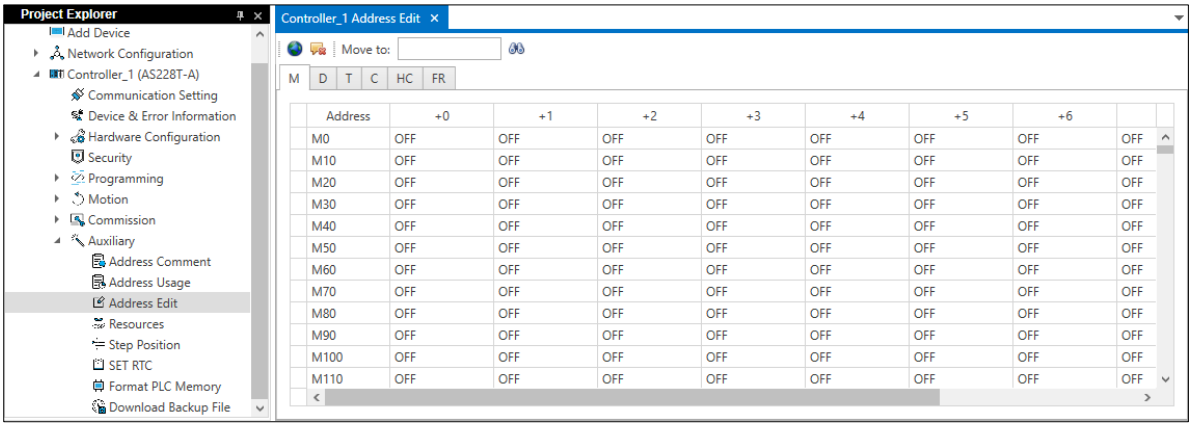
Function	Description
Code Memory	Percentage of memory used by user programs in Steps.
Source Code Memory	Percentage of memory used by whole project in Steps.
Internal Register Occupied by Variable	Percentage of used retentive register in Words.
Internal Retentive Register Usage	Percentage of register occupied by variable declaration in Words.
Function Block Memory	Percentage of used Function Block Memory in Words.
SFC Memory	Percentage of user SFC Memory in Words.
C binary code usage	The usage rate of the C Code execution code in the storage space (64K Byte).
C Variable usage	The usage rate of C language variables occupying variable declaration free space (8K Byte).
C and PLC exchange variable usage	The usage rate of the C_Tag_Table variable (1000 Words).

16.4 Register Edit

Edit the values in M, D, T, C, HC, and FR registers in controller by Register Edit. Read / write value from / to controller by Register Edit. Locate Register Edit in **Controller** > **Auxiliary** > **Register Edit**.



Double-click **Register Edit** in **Project Explorer** to open **Register Edit** dialog.

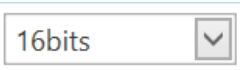


The **Register Edit** dialog has a toolbar and the register editing area.

16.4.1 Register Edit Toolbar

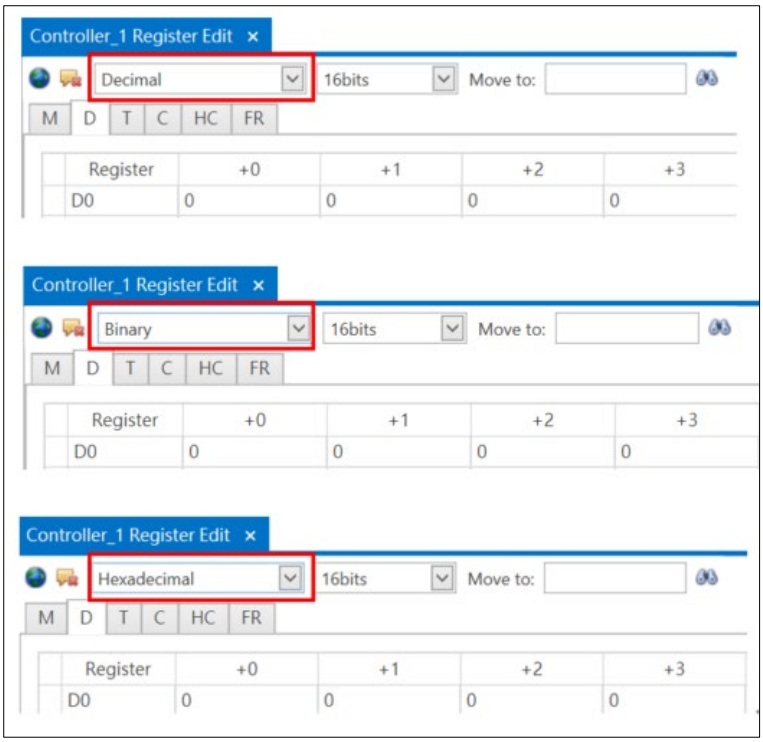
The following table describes the toolbar functions of **Register Edit** dialog.

Function / Icon	Description
	Click to open **Transfer Setup** dialog, only available in offline mode.
	Click to clear or reset the values of all registers in the **Current Register** dialog (e.g. M, D etc.) to 0.
	Import file in CSV format.

Function / Icon	Description
	Export file in CSV format.
	Click to select data type length from the drop-down list. • Decimal • Unsigned Decimal • Hexadecimal • Binary • Floating Point • ASCII **Note**: This field displays for all register types except M. The default value is **Decimal**.
	Click to select data Length type from the drop-down list. • 16bits • 32bits **Note**: This field is applicable for all register types except M and HC. The default value is **16bits**.
	Enter the register number to be navigated.
	Click to navigate to the location of the Register value entered in the **Move to** field.
	Click the register letter to open the respective register tab.

16.4.2 Data Format

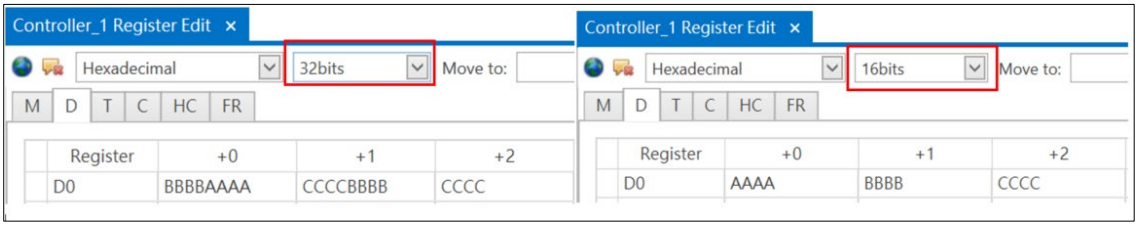
The following figure shows an example of Decimal, Binary, and Hexadecimal format for D0, D1, and D2 registers.



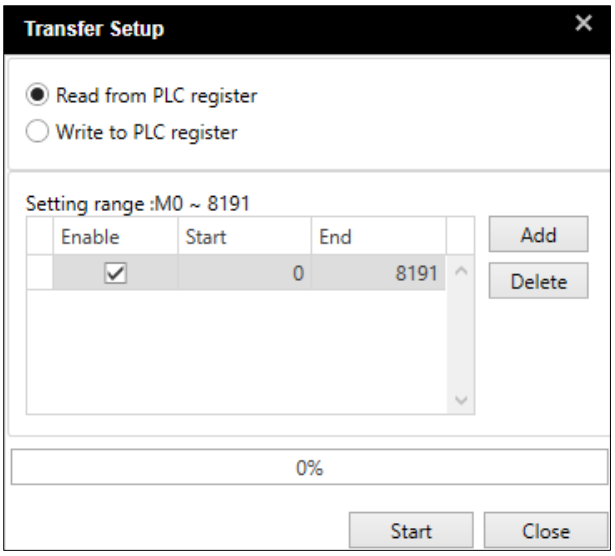
16.4.3 Data Length

Select 16bits or 32bits options to display the register value. If 32bits option is selected, then the value in the cell for a register includes the value in the next register.

Assume that the selected data format is Hexadecimal, and if the values in registers D0 = AAAA, D1 = BBBB, D2 = CCCC, the **Register Edit** dialog displays the values as shown in the following figure for **16bits** and **32bits** options.



The following figure shows **Transfer Setup** dialog.



The following table describes the functions in **Transfer Setup** dialog.

Function	Description
Read from PLC register	Click to select read values from PLC registers. **Note**: **Read from PLC register** is selected by default.
Write to PLC register	Click to select write values to PLC registers.
Setting range	Display the register range for the selected register type, you can add registers in separate segment. **Example**: M0 ~ M8191 for M, D0 ~ D29999 for D etc.
Enable	Click to check / uncheck the address edit lines. **Note**: Address edit lines are checked by default.
Start *(column)*	Enter the start value of address edit. **Note**: **0** is the default value.
End	Enter the end value of address edit. **Note**: Default value is the full range of register type selected.
Add	Click to add address edit range lines.
Delete	Click to delete address edit range lines.
Start *(button)*	Click to start the transfer of register values (read/write).
Close	Click to close **Transfer Setup** dialog.

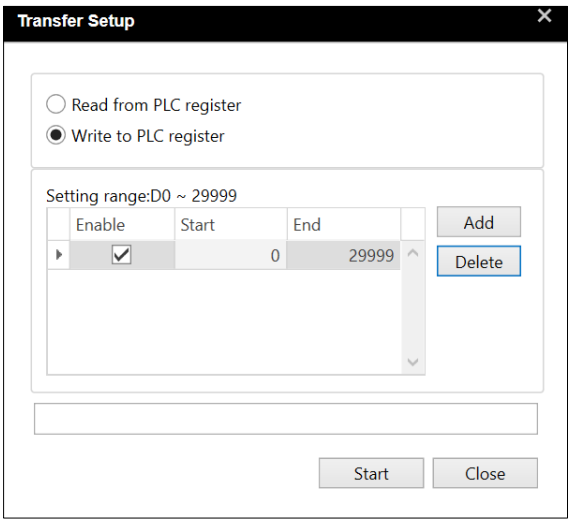
16.4.4 Address Edit – Write

Follow these steps to perform write register values to PLC.

To perform write register values to PLC

1. Click a register type tab.

2. To write register values to the PLC, edit the register values first, and click  to open **Transfer Setup** dialog.



Note: Before values are written into the PLC registers, make sure that the operation does not affect the system, or cause damage to the system and staff!

3. Click **Write to PLC register**.

4. Click **Add** to add extra address lines, if the register range to be written is not continuous.

5. Edit **Start** and **End** values according to your requirement.

6. Click **Start**.

   Register values in the PLC is written as per editing done and range selected. The status appears in the **Transfer Setup** dialog in the progress bar.

7. Click **Close** to close **Transfer Setup** dialog.

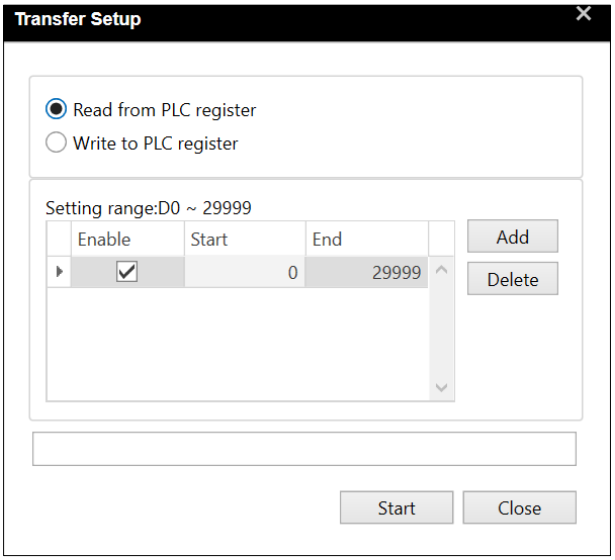
16.4.5 Address Edit – Read

Follow these steps to perform read register values from PLC.

To perform read register values from PLC

1. Click a register type tab.

2. Click  to open **Transfer Setup** dialog (**Read from PLC register** is selected by default).



3. Click **Add** to add extra address lines, if the register range to be read is not continuous.
4. Edit **Start** and **End** values according to your requirement.
5. Click **Start**.

   Register values in the PLC are read as per editing done and range selected. The status appears in the **Transfer Setup** dialog in the progress bar.
6. Click **Close** to close **Transfer Setup** dialog.

16.4.6 Address Edit – Clear all Values

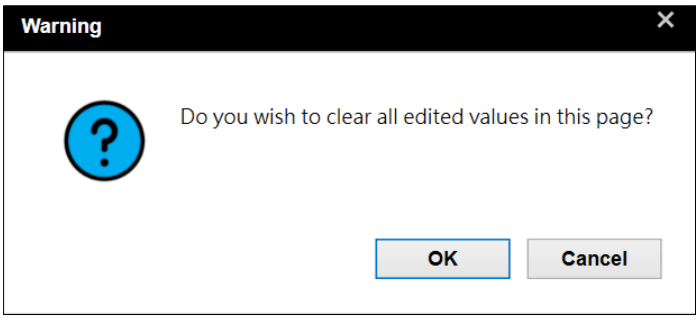
Follow these steps to clear the values of all registers in a register tab.

To clear values of all registers in register tab

1. Click .

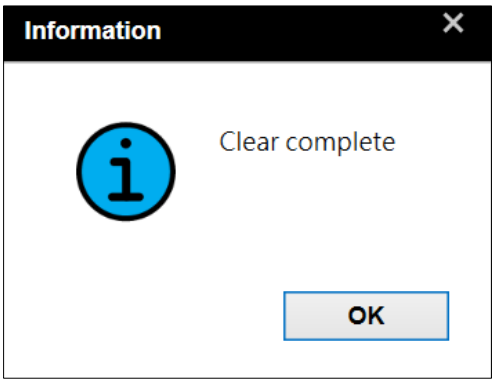
   Or

   Right-click in the register tab to open its context menu, and click **clear all edited values in this page** to open an alert message.



2. Click **OK**.

   All register values in the selected tab (register type selected) are reset to 0 and **Information** dialog appears as shown in the following figure.



Note: Only the value in the present edited page (register type) is reset to 0. Values edited in other pages (register types) remain intact.

16.4.7 Address Edit – Move to

Enter a register address in the **Move to** field and navigate to the respective address.

Follow these steps to move to a register location.

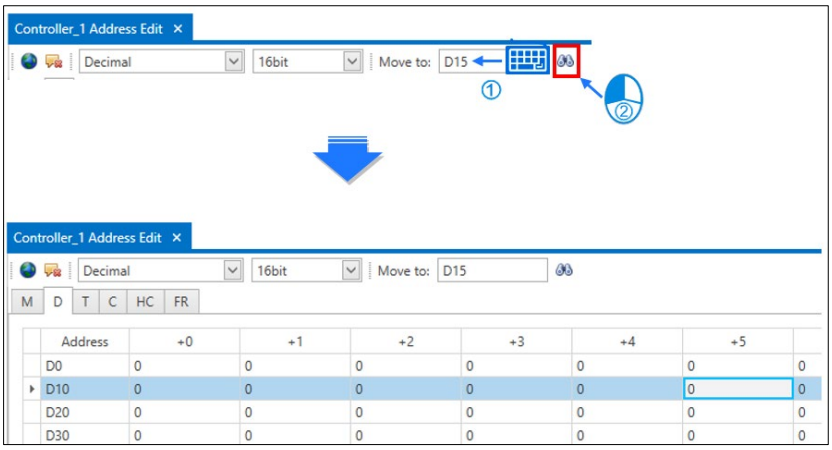
To move to a register location

1. Enter a register value in **Move to** field.
2. Click .

   Or

   Press **Enter** key on the keyboard.

   System will navigate to the particular location entered in **Move to** field.



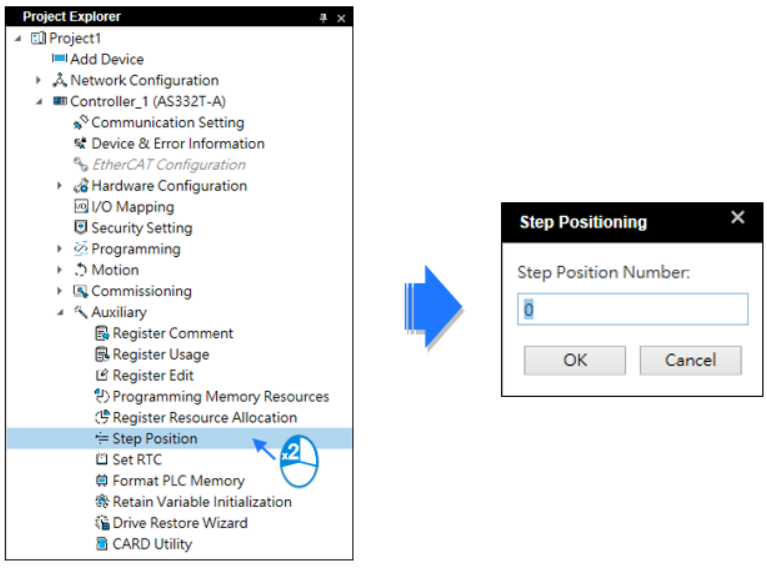
16.5 Step Position

When an error occurs during operation, use special registers (example SR1, SR2, the 32-bit registers for storing wrong addresses) to diagnose which step went wrong and then navigate to the specific program to check and fix it. Step Position helps you to understand which step number has gone wrong. Locate Step Position in **Controller** > **Auxiliary** > **Step Position** in DIADesigner **Project Explorer**.

Follow these steps to perform Step Position.

To perform Step Position

1. Double-click **Controller** > **Auxiliary** > **Step Position** in **Project Explorer** to open **Step Position** dialog.



2. Enter the step number in **Step Position Number** field.

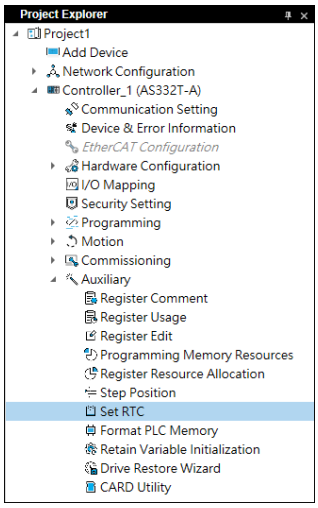
3. Click **OK**.

   System directs you to the corresponding POU and the position of the step in the program.

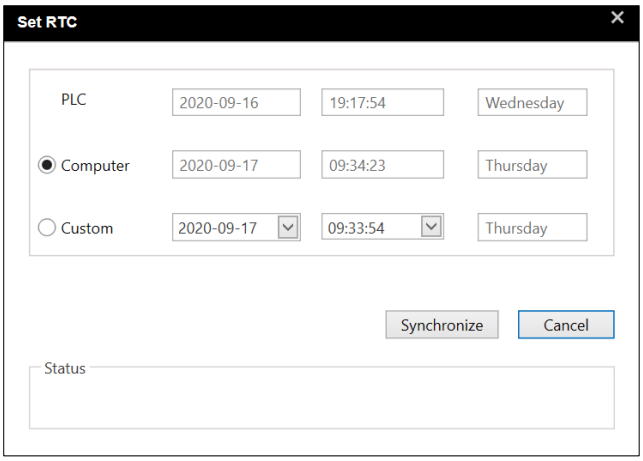
   Note: Be sure to program the project before using **Step Position** function.

16.6 Set RTC

Set the real time clock in the PLC from DIADesigner with the help of Set RTC. Make sure that DIADesigner is connected to the CPU module before performing this action. Locate the Set RTC in **Controller** > **Auxiliary** > **Set RTC** in the **Project Explorer**.

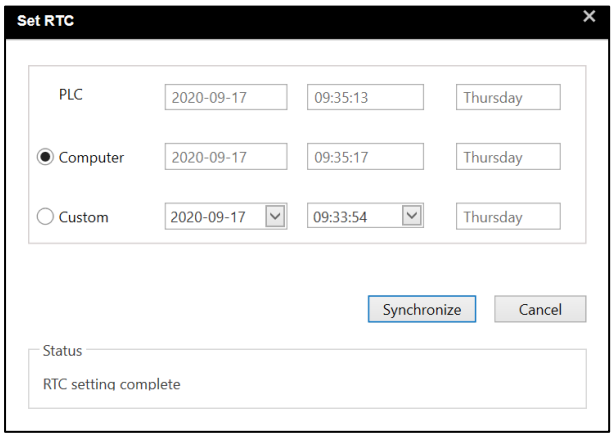


Double-click **Set RTC** to open **Set RTC** dialog.



16.6.1 Updating Computer Time to PLC Time

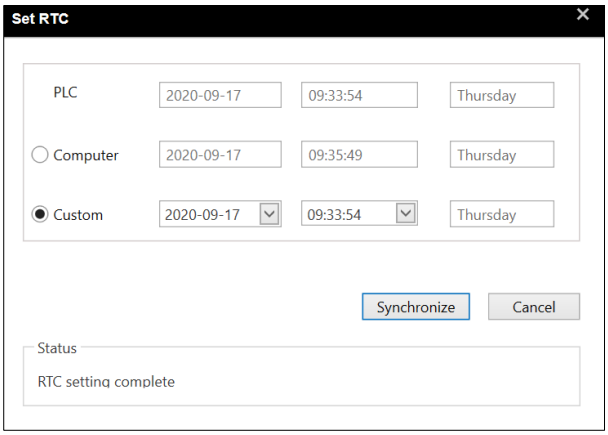
The **Set RTC** dialog shows the PLC date, time, and day information retrieved from the PLC. If you want to change the real-time clock in the PLC with the Computer's time, select Computer option and click **Synchronize**. Computer date, time, and day information are written to the PLC RTC. DIADesigner updates the status of action – **RTC setting complete** as shown in the following figure.



16.6.2 Updating Custom Time to PLC Time

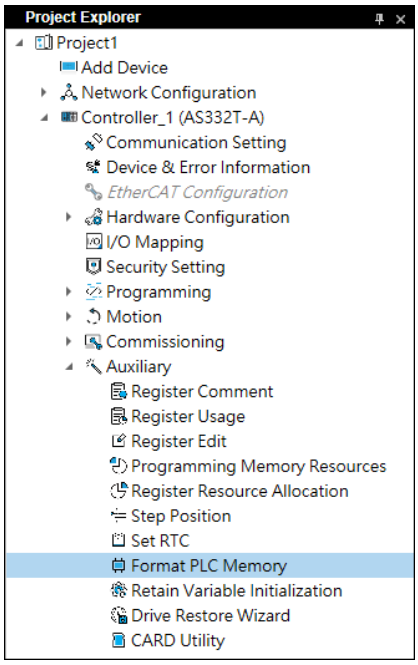
To set the real-time clock information in the PLC with a custom time, select the **Custom** option in the **Set RTC** dialog, and enter / select the date, time and day in the respective fields and click **Synchronize**.

DIADesigner updates the status of action – **RTC setting complete** as shown in the following figure.

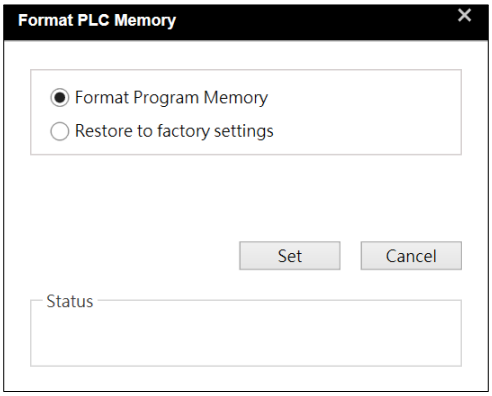


16.7 Format PLC Program Memory

Clear memory or restore the setting in the PLC to the factory setting from DIADesigner with the help of Format PLC Memory. Make sure that DIADesigner is connected to the CPU module before performing this action. Locate the Format PLC Memory function in **Controller** > **Auxiliary** > **Format PLC Memory** in the **Project Explorer**.



Double-click **Format PLC Memory** to open the **Format PLC Memory** dialog.



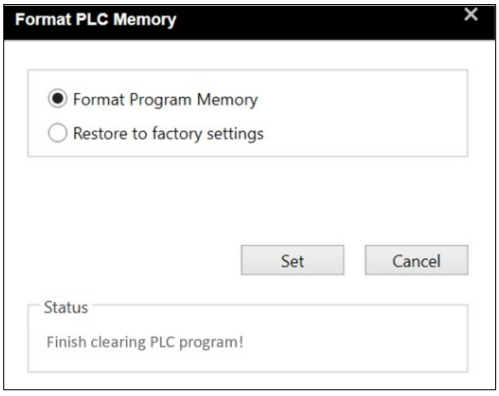
The following table describes the functions in **Format PLC Memory** dialog.

Function	Description
Format Program Memory	Click to format PLC program memory.

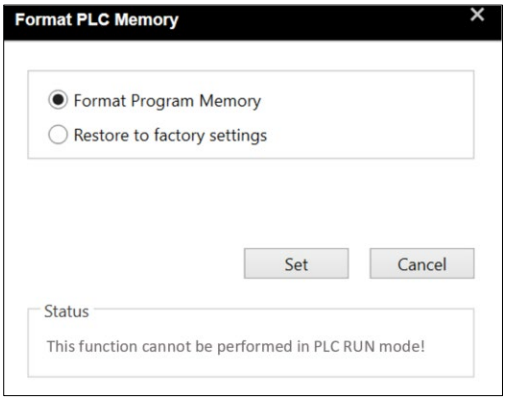
Function	Description
Restore to factory settings	Click to reset PLC to factory setting.
Set	Click to perform the action selected.
Cancel	Click to cancel the action.

16.7.1 Format Program Memory

To format program memory, select **Format Program Memory** option and click **Set**. DIADesigner updates the status of action – **Finish clearing PLC program!** as shown in the following figure.



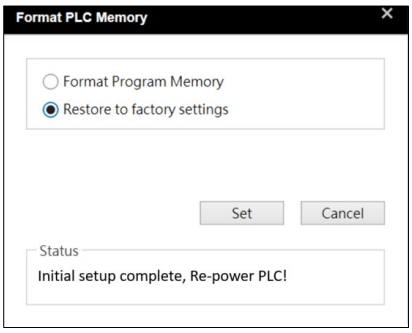
Note: You can format program memory when the controller is in **Offline** and **Stop** mode. In **Online** mode, **Format PLC Memory** function is disabled. When the PLC is in **Run** status, a message – **This function cannot be performed in PLC RUN mode** appears as shown in the following figure.



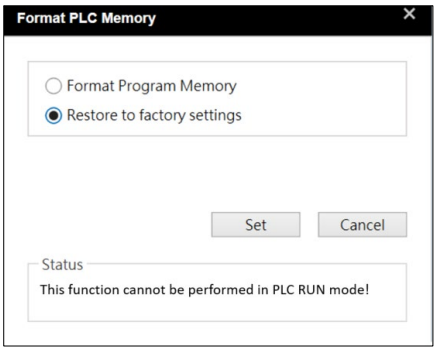
16.7.2 Reset PLC Memory (Factory Setting)

To reset the PLC memory to factory setting, select **Restore to factory setting** option in **Format PLC Memory** dialog and click **Set**.

DIADesigner updates the status of action – **Initial setup complete, Re-power PLC!** as shown in the following figure.



Note: Upon performing **Restore to factory settings**, restart the PLC power supply to proceed with the operations. Perform **Restore to factory settings** when the controller is in **Offline** and **Stop** mode. In **Online** mode**, Format PLC Memory** function is disabled. When the PLC is in **Run** status, a message – **This function cannot be performed in PLC RUN mode!** appears as shown in the following figure.



16.8 Backup & Restore

16.8.1 Parameter Backup & Restore

Note: Parameter Backup & Restore is supported only in the VFD C2000 series now. For other series, it may be supported in future version.

You can back-up the parameters in inverters, and restore it afterward. Navigate **Inverter Device > VFD Function > Parameter Backup & Restore** in **Project Explorer** to open the tab.

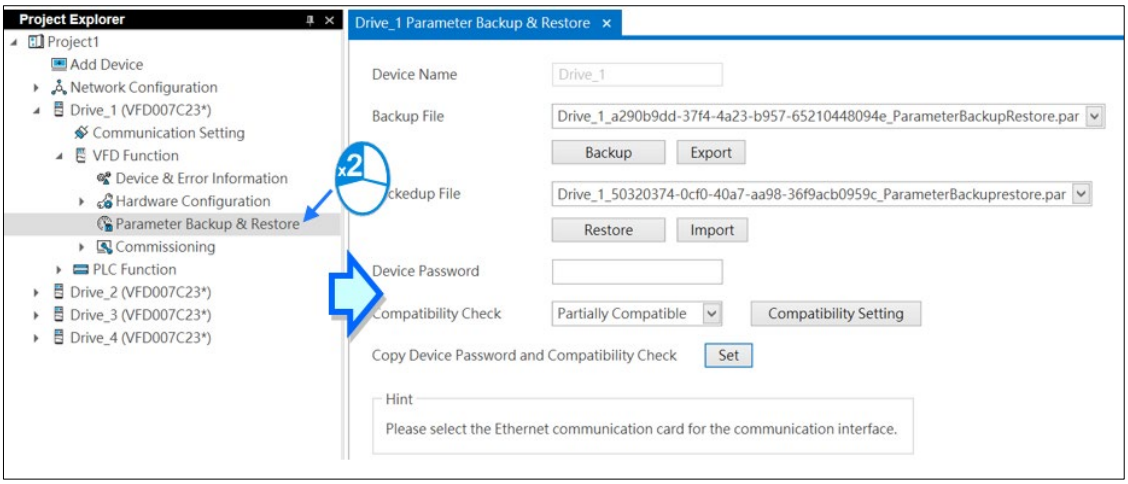
The following table describes the functions in **Parameter Backup & Restore**.

Function	Description
Device Name	Device name of target device.
Backup File	Select a file to export or overwrite.
Backup	Button to execute Backup.
Export	Export file to a specified directory.
Backed up File	Select a backed up file.
Restore	Button to execute Restore.
Import	Import file from a specified directory. (*.par file)
Device Password	Enter password which is matching with parameter 00-08 to proceed transformation, keep blank if the connected device has no password.
Compatibility Check	Select between **Partially Compatibility**, **Exactly Match**, and **Forced Transformation** before proceeding to restore.
Compatibility Setting	Available when **Partially Compatibility** is selected in Compatibility Check.
Copy Device Password and Compatibility Check	Copy current setting and password to other inverters.

Follow these steps to perform Parameter Backup & Restore.

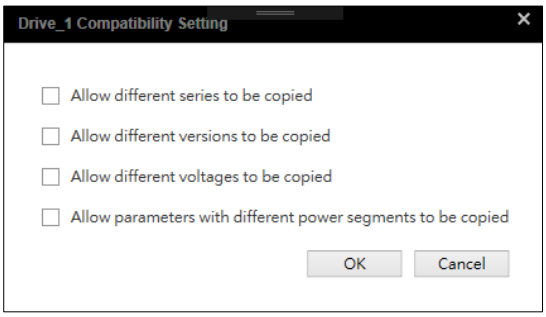
To perform Parameter Backup & Restore

1. Set up **Communication Setting**. (When connected, the backup, restore function in the **Parameter Backup & Restore** function will be turned on).
2. Click **Inverter Device** > **VFD Function** > double-click **Parameter Back-up & Restore** in **Project Explorer** to open the tab.

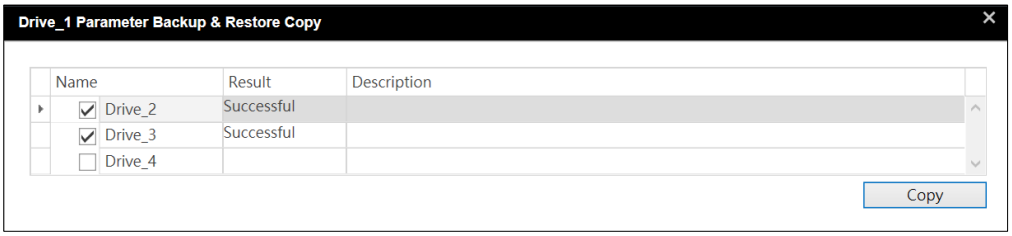


3. Backup: Click **Backup** to upload all parameters from current connected inverter and save it in a file in the Backup path directory.
4. Restore: Click **Restore** to overwrite all parameters from the file in Restore Path directory to the connected inverter.

In case of different application situation, DIADesigner allows you to restore the file to compatible devices, like restore the parameters into different series or version. Select **Partially Compatible** in **Compatibility Check**, the restore allows according to the setting of **Compatibility Setting** dialog.



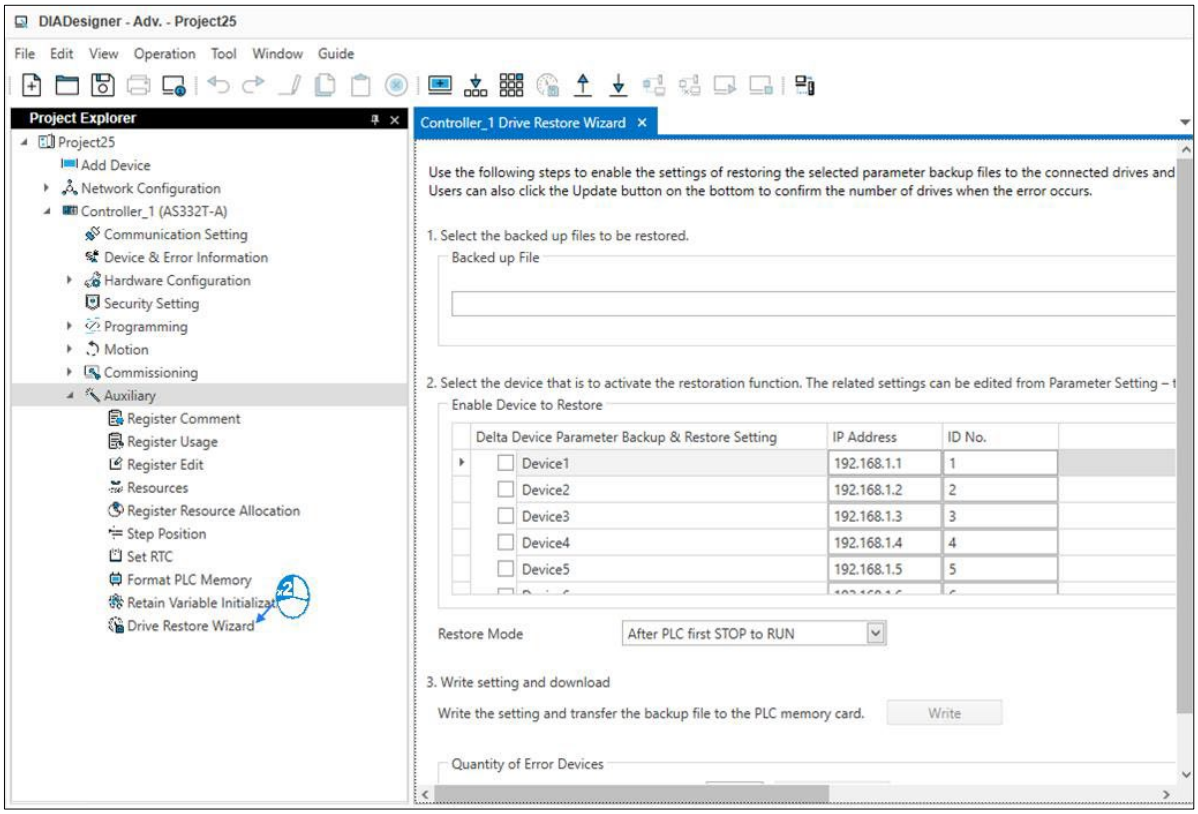
If you want to copy current setting to other inverters, click **Copy** button to copy Compatibility Setting and Password to the selected inverters in project.



16.8.2 Drive Restore Wizard

The content in this section is for the files of one / multiple inverters under different parameter setting to be back up to the PLC memory card and the restore mode is set to restore the backup parameters to other devices.

In **Project Explorer**, **Controller Device** > **Auxiliary** > **Drive Restore Wizard**, double-click **Drive Restore Wizard** to open **Drive Restore Wizard** dialog.

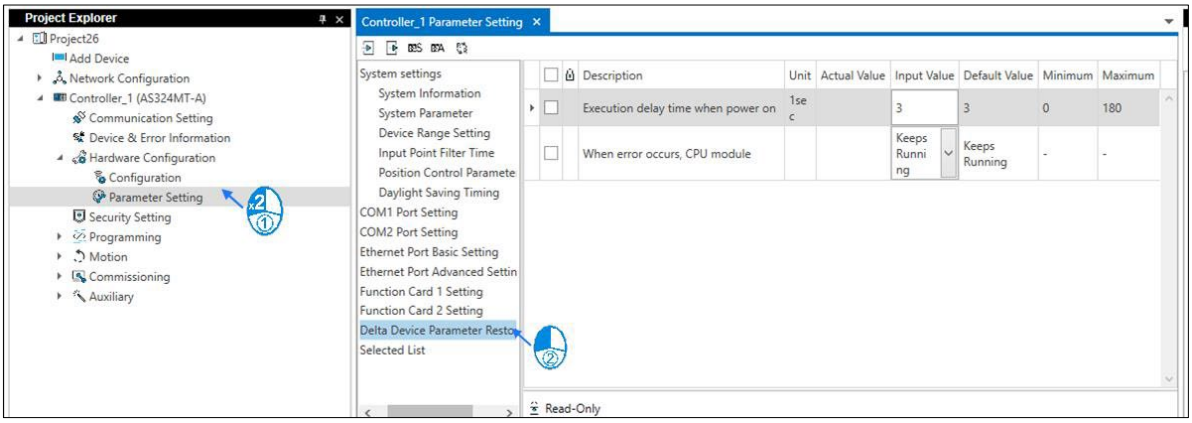


The following table describes the functions of **Drive Restore Wizard** dialog.

Function	Description
Files backup	This drop-down menu can select the backed-up file to be restored.
Import	Click this button to choose to import the backed-up file to be restored.
Enable restore device	Set the **IP Address** and **ID No** of the restoration device and select the device to enable the restoration function.
Restore mode	Select the restore mode. Choose from **After PLC first STOP to RUN** or other methods.
Write setting and move backup files to PLC memory card	Click **Write** button to write the setting and the contents of the moved backup file to the PLC memory card.

Function	Description
Number of error devices	Display the number of devices with errors in parameter restoration.

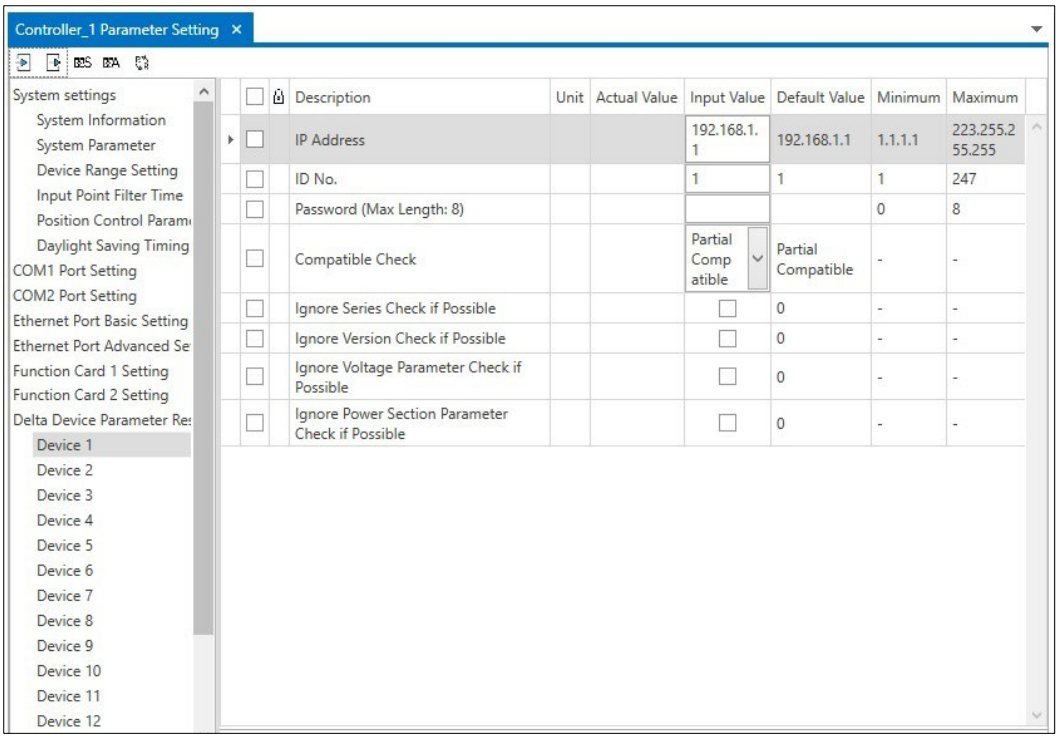
To set the inverter parameter restoration, go to **Controller Device** > **Hardware Configuration** > **Parameter Setting** in the **Project Explorer**. Double-click the parameter setting, and select **Delta Device Parameter Backup/Restore** on the left side of the **Parameter Setting** tab.



The following table describes the functions of **Delta Device Parameter Backup/Restore**.

Function	Description
Execution delay time when Power-on	Set the power-on delay time.
When error occurs, CPU Module	Set whether the CPU should enter the stop state when an error occurs.

To restore setting to the restored device, click the arrow on the left side of the **Delta Device Parameter Backup/Restore** setting option to expand the device, and click the device to open **Parameter Setting**.



The following table describes the functions of **Delta Device Parameter Backup/Restore** setting-device page.

Function	Description
IP address	Enter the IP address of the target device.
ID No	Enter the ID number of the target device.
Password (Max Length: 8)	Enter the password of the target device.
Compatibility Check	The input value selection is partially compatible and the following four restoration conditions can be cancelled. • Ignore Series Check if Possible: You can still try to restore when the series of devices are not the same. • Ignore Version Check if Possible: You can still try to restore when the device versions are not the same. • Ignore Voltage Parameter Check if Possible: You can still try to restore when the voltage parameters of the device are not the same. • Ignore Power Section Parameter Check if Possible: You can still try to restore when the power section parameters of the device are not the same. If the input value is completely compatible, the restoration must be performed when all the conditions are compatible. If the input value selection is maximum, it tries to force the restoration directly.

Function	Description
	If the target device does not support the compatibility check, even if the ignore condition is set, the restoration may not be performed. For the conditions supported by the target device, refer to the user manual of the connected target device.

Complete **Delta Device Parameter Backup/Restore** setting. Open **Drive Restore Wizard** setting and follow these steps to perform restore.

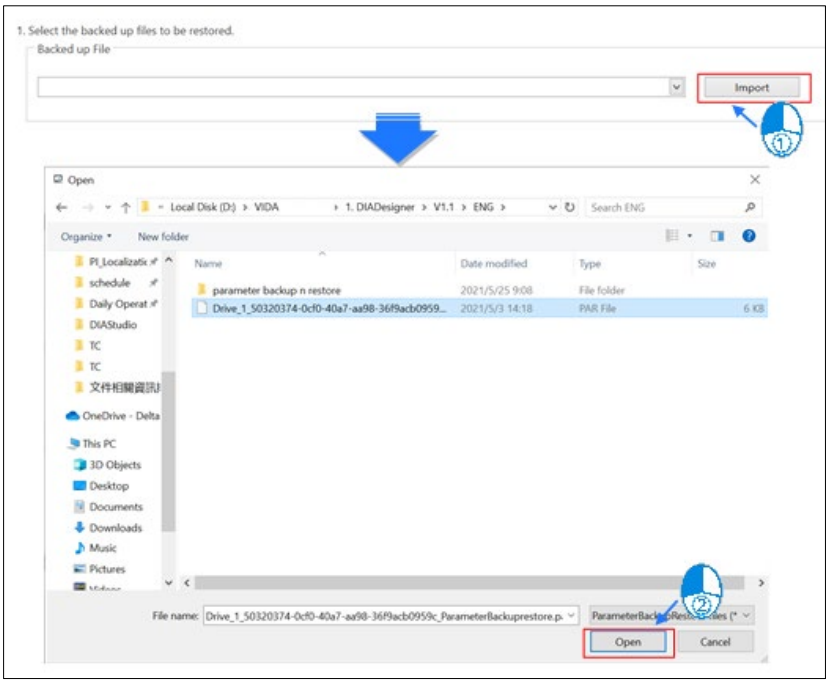
To perform restore

1. Select the backup file in the drop-down menu. The backup file method can be seen in the inverter. Refer to Parameter Backup & Restore.

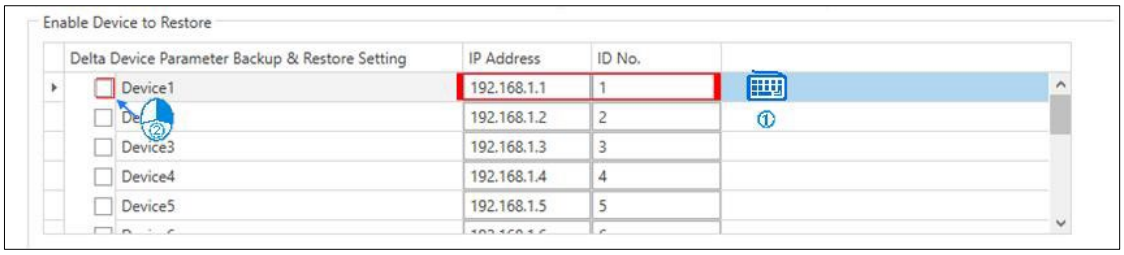


Or

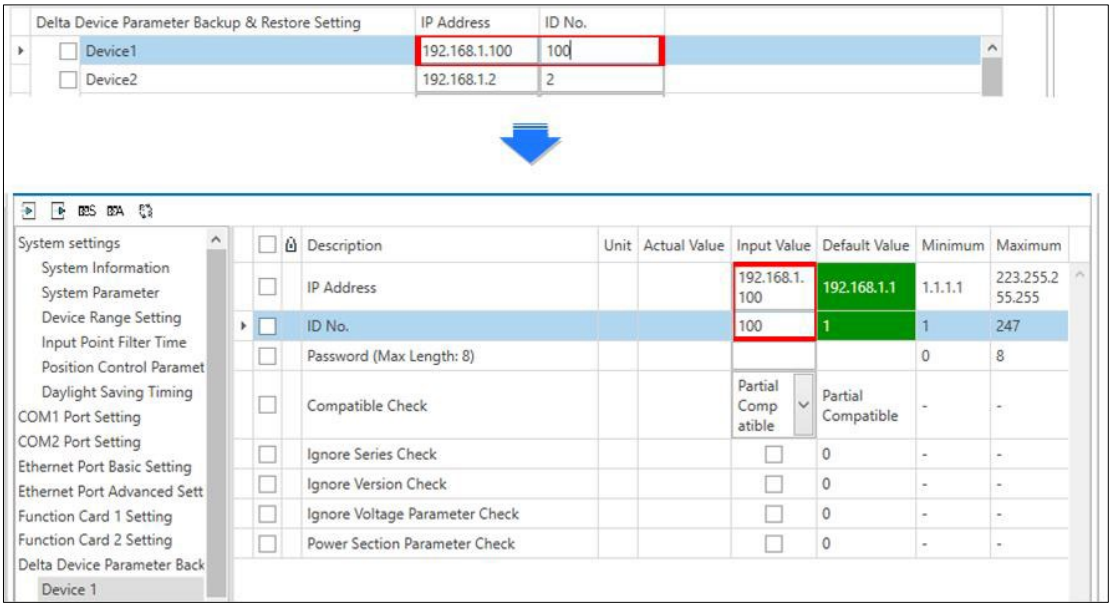
Click **Import** button to import the backup file.



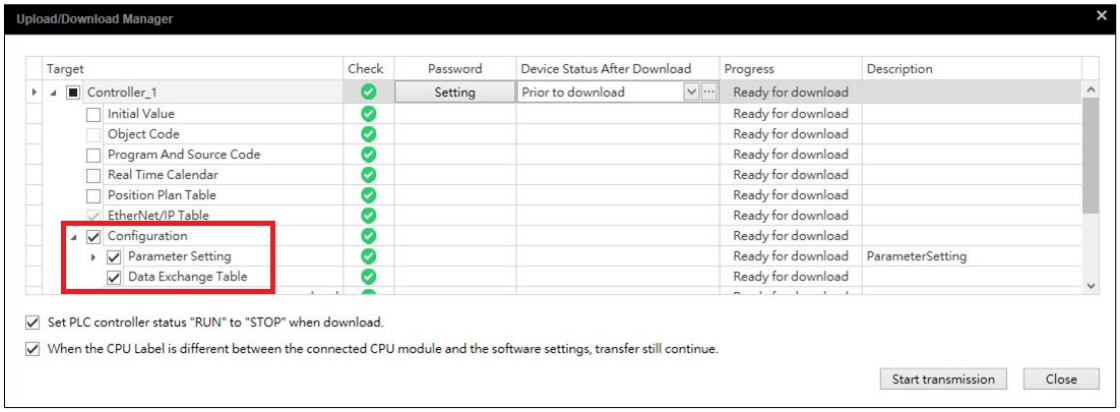
2. Enable the device name in the **Delta Device Parameter Backup & Restore Setting**, edit the IP address in **IP Address** field and edit the station number in **ID No.** field.



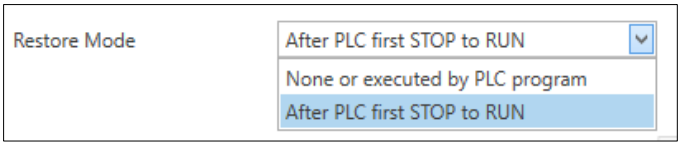
Note: The IP address and station number setting here will be synchronized with the contents of the device setting page of Delta Device Parameter Backup & Restore Setting.



3. Click Download to download the configuration content to the PLC.



4. Select **Restore Mode**.

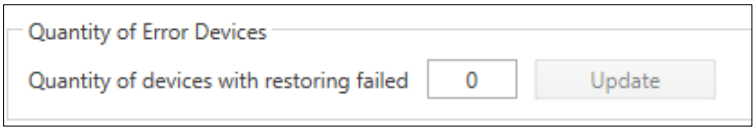


5. Click Write in **Write Setting and Download**. When the writing is successful, the success status appears in the description field of the error list.



6. If the restoration mode selects **After PLC first STOP to RUN**, after the writing is successful, when the PLC state changes from STOP to RUN for the first time, the restoration action will be executed.

   At this time, to check whether the restoration is successful, click **Update** to view the number of devices with errors in the parameter restoration.



16.8.3 Backup & Restore

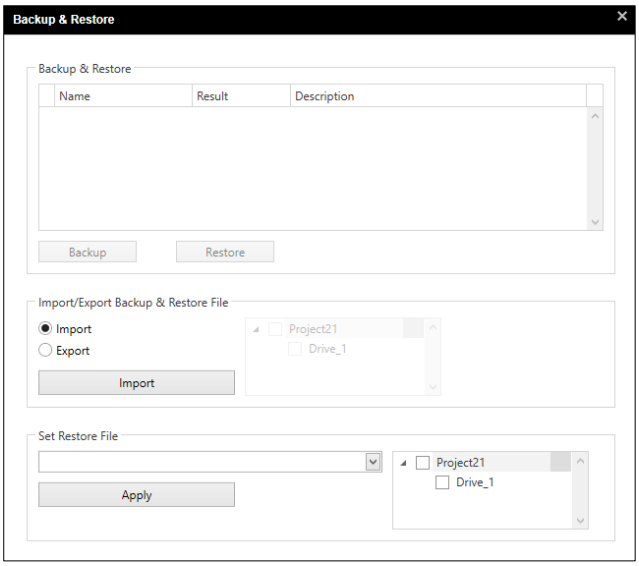
Note: Backup & Restore currently only supports the inverter C2000 series, and the remaining series will be supported in future versions.

The content in this section is for one or more inverters (Ethernet connection) in **Project Explorer** to perform backups of files under different parameter setting and restore the backup parameters to one or more inverters in **Project Explorer**.

The inverter provides a parameter backup function to allow you to back up parameters. Click  on development toolbar to open **Backup & Restore** dialog. Backup setting can be performed, and the restore parameter function of multiple inverters can be set to execute at the same time.



Note: If there is no inverter device in the project, the Backup & Restore icon is disabled and cannot be used.



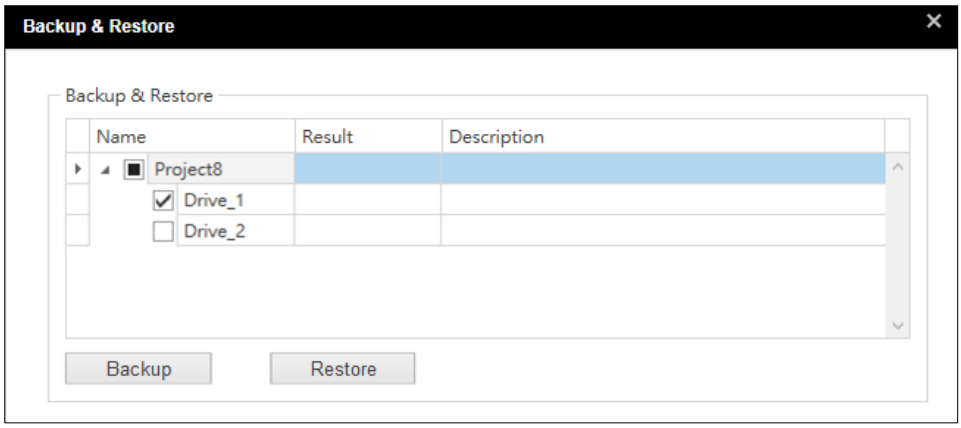
The following table describes the functions of **Backup & Restore** dialog.

Function	Description
Device Name	When the device has an Ethernet connection, the name of the device that can perform parameter backup and restoration appears in the list.
Backup	Backup can be performed after selecting the device.
Reduction	After setting the recovery file path, select the device to perform the recovery.
Import	Import files to a specific directory (*.par file).
Export	Select the device to export the backup parameters and export its files to a specific directory.
Set Restore File	Select the backup file you want to restore to the device, and tick the device you want to run.

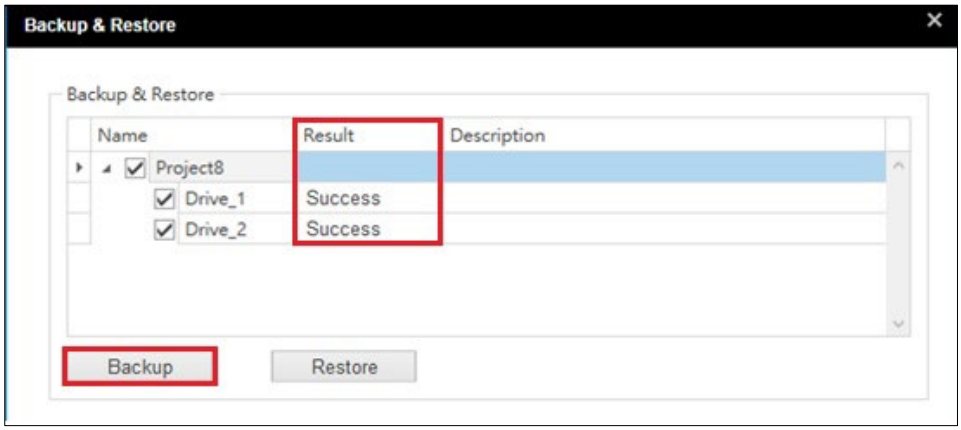
Follow these steps to Backup & Restore.

1. **Backup & Restore**

   a. All inverter devices need to perform Ethernet communication setting and connect. When there is a connection, the device appears in the **Backup & Restore** field, and the backup and restore functions can only be used when the device is selected.



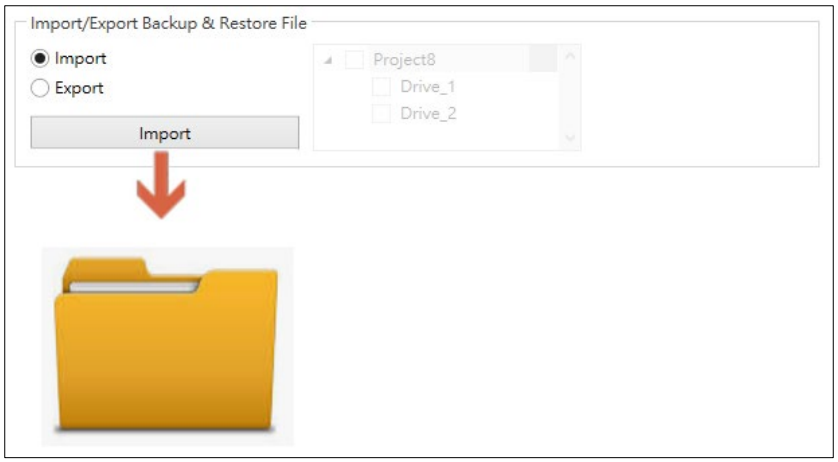
b. Click **Backup** to upload all the parameters of the currently selected inverter and store them in the parameter backup file folder. After the execution is completed, the success or failure appears in the **Result** column. You can also directly import the backup file to replace the action of generating the backup file.



2. Import / Export Backup & Restore Files

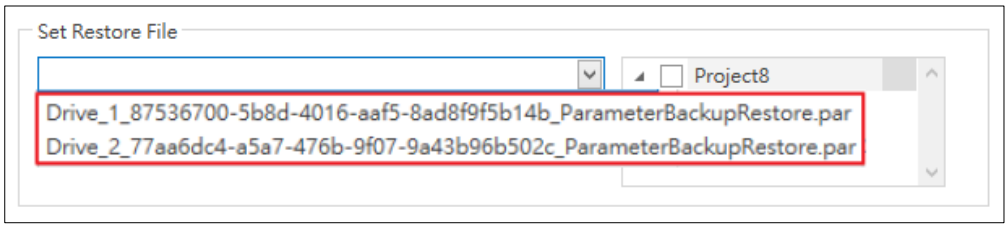
a. Import: The import function can replace the action of generating backup files in backup and restore.

b. Export: If necessary, export the backup files of one or more selected inverter devices to a specific directory.



3. Set the recovery file

a. Select the generated or imported backup file in the drop-down menu of **Set Restore File**.



b. In the device list on the right side of the setting recovery file field, select single, multiple or all devices to be recovered.

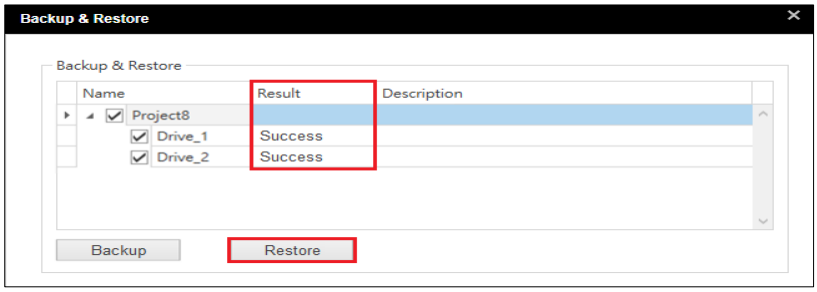
c. Click **Apply** to apply the restored file path to the selected device. After the application is completed, the application result appears in the error list.





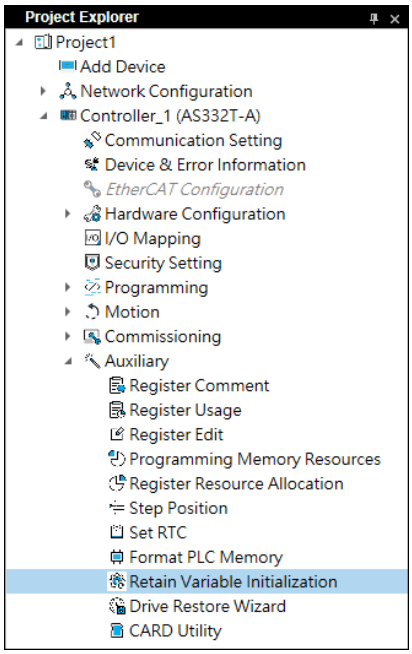
d. Restore: Click **Restore** to overwrite all the current parameters of the selected inverter with the values in the set restore file. After the restore is executed, the

execution result appears in the **Result** field.

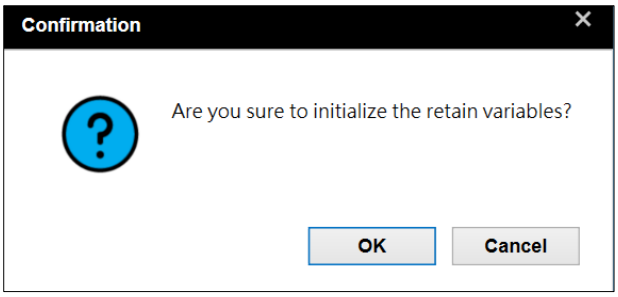


16.9 Retain Variable Initialization

Reset the retain variable to initialize value of a PLC. Back up the files of one / multiple inverters under different parameter setting to the PLC memory card and set the restore mode to restore the backup parameters to other one / multiple devices. Locate the Retain Variable Initialization in **Controller** > **Auxiliary** > **Retain Variable Initialization** in **Project Explorer**.



Click **OK** in **Confirmation dialog** to initialize the retain variables.



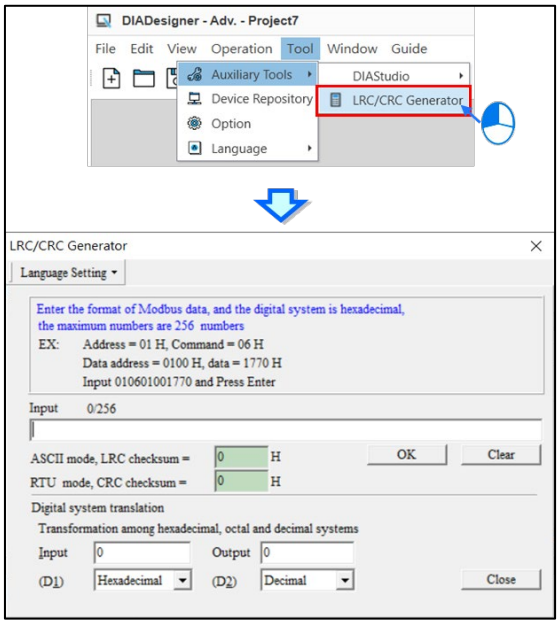
Note: Make sure the retain variables are downloaded, and the PLC is in STOP mode before executing the function.

16.10 LRC/CRC Generator

Follow these steps to calculate LRC / CRC checksum by this tool and to transform the numbers into different numeral systems.

To calculate LRC / CRC checksum

1. Click **Auxiliary Tools** > **LRC/CRC Generator** to launch the function.



2. Enter data in the **Input** box, then click **OK** to generate the LRC and CRC checksum.
3. Select the base of input and output in the Digital system translation area, and then enter number in the **Input** box, the transformation result appears in **Output** box.

16.11 Memory Card Tool – CARD Utility

CARD Utility mainly helps you to back up and restore data on memory cards, CPU modules, and DIADesigner projects, including program code, parameter settings, hardware configuration, network configuration, and the values in the devices on the CPU module. Refer to the operation manuals or technical documents for each model for more information about the SD card specifications.

The following functions are supported by **CARD Utility**.

- If you export data from the CPU module as a backup file (*.dup), the data exported can be saved in the memory card or in the computer. You can also export the values in the devices in the CPU module.
- Download the backup file saved in the inserted memory card back to the CPU module.
- Import a backup file (*.dup) saved in a computer back to the AH series CPU module.

Note: **CARD Utility** also supports the communication setting function. Toggle to **Communication Setting** tab by clicking **Tool** > **Communication Setting**. Refer to Communication Setting for the operation instructions.

Before performing a backup / restore using a memory card, refer to the following CARD Utility Backup and Restore table to check if the item to be backed up / restored is supported by your model type.

	AS	DVP-ES3 / ES3 (TEC) / EX3 / SV3 / SX3
Source Code	V	V
Actuating Code	V	V
Project Password / Program ID	V	V
HWCONFIG Configuration	V	V
CPU Modules Configuration / Modules Parameter settings	V	V
Registers (Note 1)	MD, C, HC, FR	M, D, C, HC, FR
Communication Data	COM1	

	AS	DVP-ES3 / ES3 (TEC) / EX3 / SV3 / SX3
Exchange Table	COM2 Ethernet FEN02 FunctionCard1 FunctionCard2 EIP I/O connection table CAN DS301 (Note 2)	
Positioning	V	V
Default Values	V	V
ECAM	V	V
Gcode	-	-
Axis Parameters	-	-
Read-only Area Settings / Comments for Backups	-	-
C Code	V	V
Data logger sampling setting	V	V
EIP TAG	V	V
OPC-UA TAG	V	V
HW TAG	V	V
IO Link Module	V	V
Webpage Setting	V (Note 2)	V (Note 2)

Note:

- Device backup and restore mechanism.
- CANopen DS301 and webpage setting only support backup and restore between CPU and SD cards.

	Backup		Restore	
	From CPU to PC	**From CPU to SD Card**	**From PC to CPU**	**From SD Card to CPU**
AS				

	Backup		Restore	
	From CPU to PC	From CPU to SD Card	From PC to CPU	From SD Card to CPU
DVP-ES3 / ES3 (TEC) / EX3 / SV3 / SX3	Optional for device backup	Optional for device backup	Restore device files when device data in backup files	Restore device files when device data in backup files

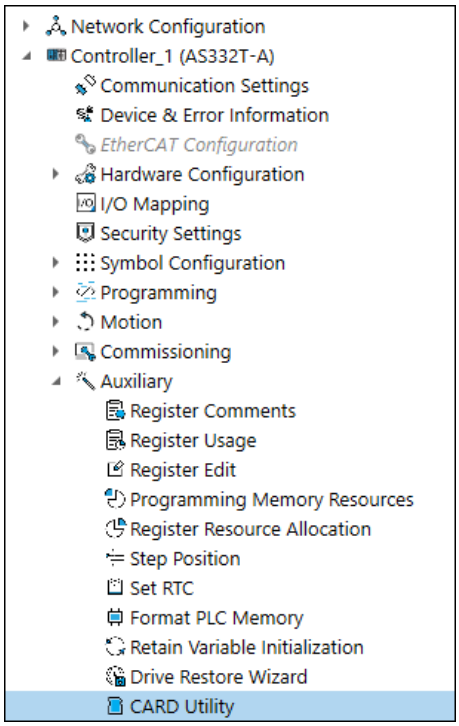
16.11.1 Backup

When the backup source / backup destination contains a CPU or memory card, make sure that there is no problem with the connection between DIADesigner and the CPU.

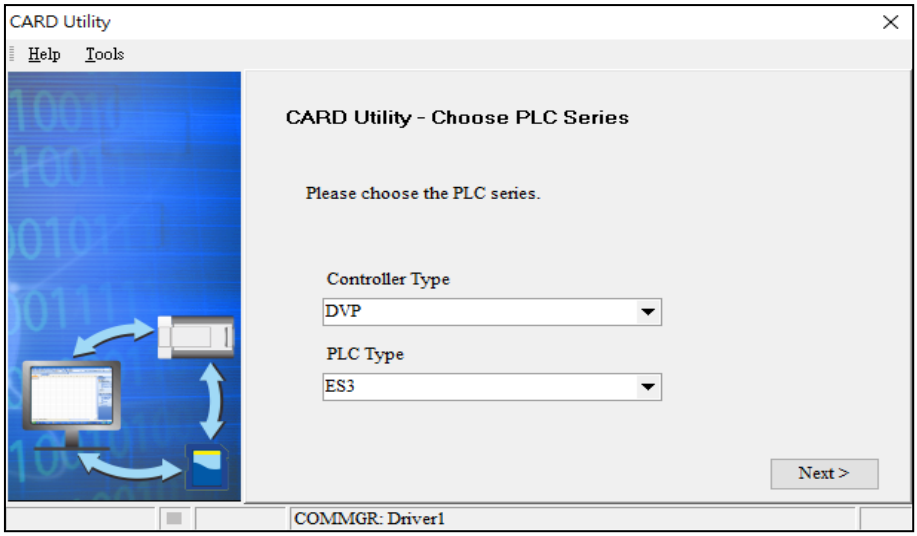
Follow these steps to back up the data.

To back up data

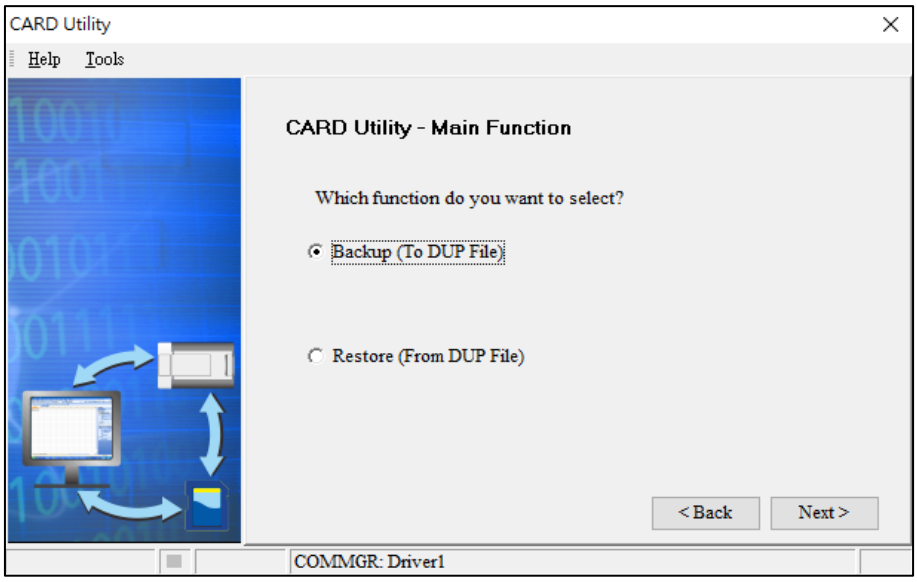
1. Create a device in **Project Explorer** and double-click **Device** > **Auxiliary** > **CARD Utility**.



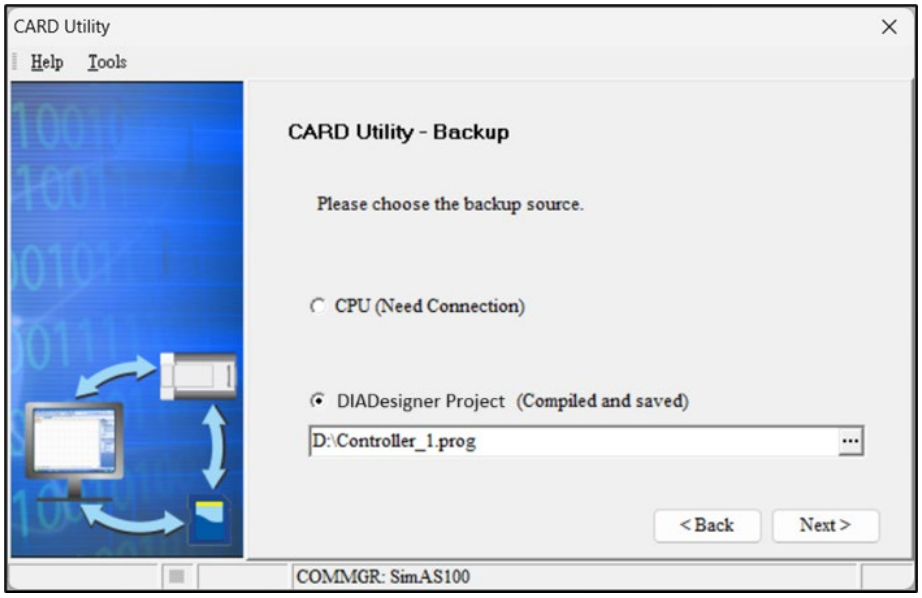
2. Click **Next** in the **CARD Utility - Choose PLC Series** dialog.



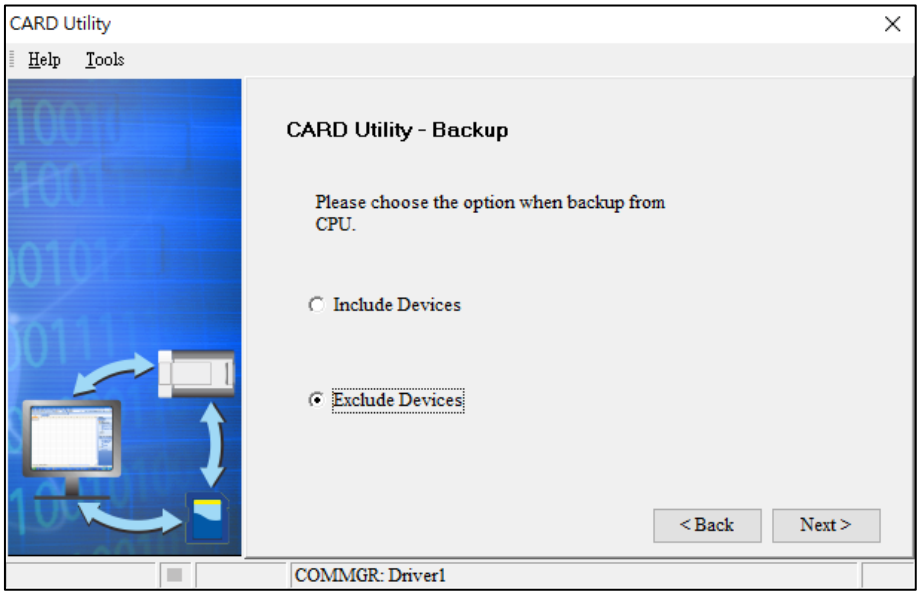
3. Select **Backup (To DUP File)** in the **CARD Utility – Main Function** dialog, and then click **Next**.



4. Choose the backup source and click **Next**.



5. When selecting **CPU (Need Connection)** as the backup source, a dialog will appear allowing you to choose whether the backup should include or exclude the device contents. If selecting **DIADesigner Project (Compiled and saved)**, a dialog will appear allowing you to choose whether the backup should include or exclude project resource code and initial values. Once selected, click **Next**.



6. Select a backup destination.

   a. **Memory Card (PLC side)** is selected.

      The filename and path of backup files for **AS** series models are as follows.

- AS300 series

  Root directory of the memory card\SDCard\PLC CARD\AS300\SysDup\AUTOEXEC.dup

- AS200 series

  Root directory of the memory card\SDCard\PLC CARD\AS200\SysDup\AUTOEXEC.dup

The filename and path of backup files for **DVP** series models are as follows.

- DVP-ES3 / EX3 / SV3 / SX3 series

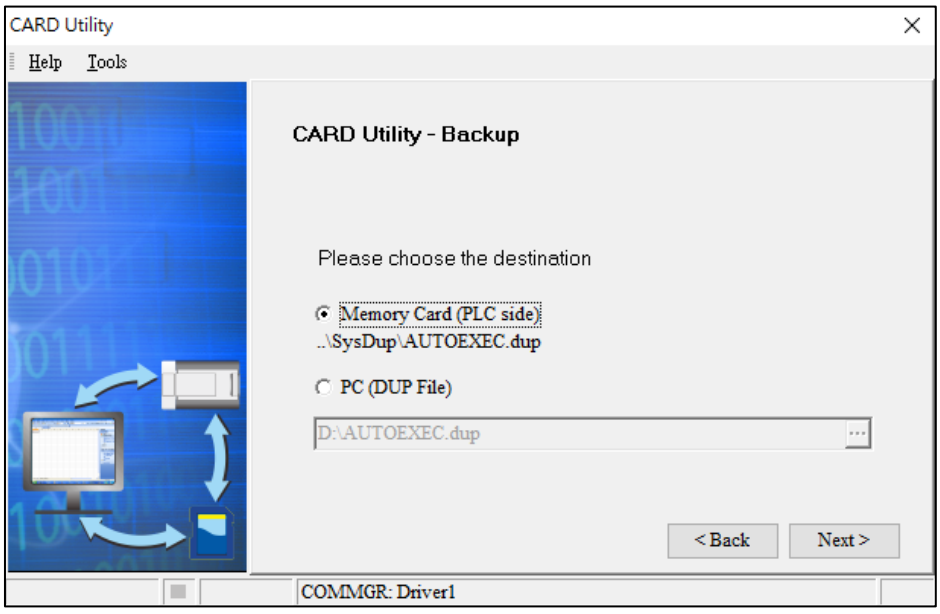
  Root directory of the memory card\SDCard\PLC CARD\ES3\SysDup\AUTOEXEC.dup

  Root directory of the memory card\SDCard\PLC CARD\ES3(TEC)\SysDup\AUTOEXEC.dup

  Root directory of the memory card\SDCard\PLC CARD\EX3\SysDup\AUTOEXEC.dup

  Root directory of the memory card\SDCard\PLC CARD\SV3\SysDup\AUTOEXEC.dup

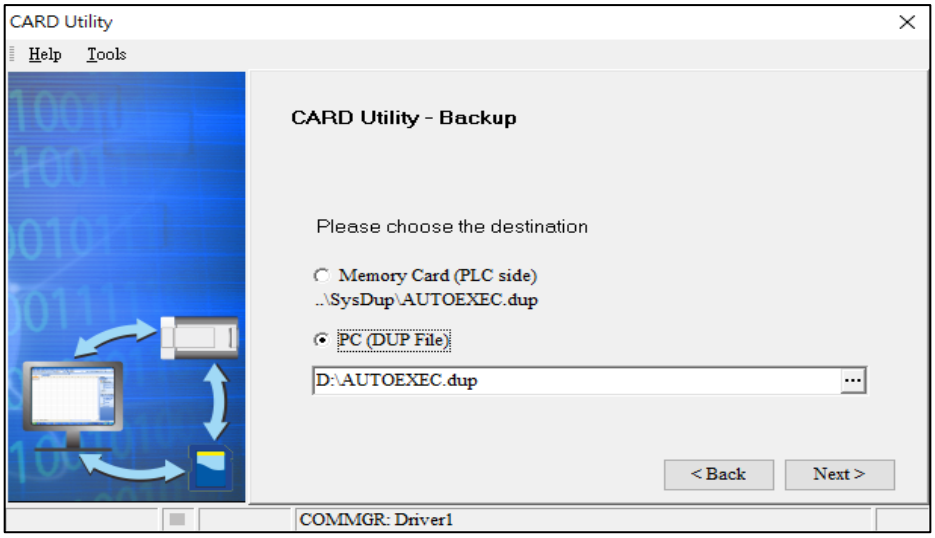
  Root directory of the memory card\SDCard\PLC CARD\SX3\SysDup\AUTOEXEC.dup



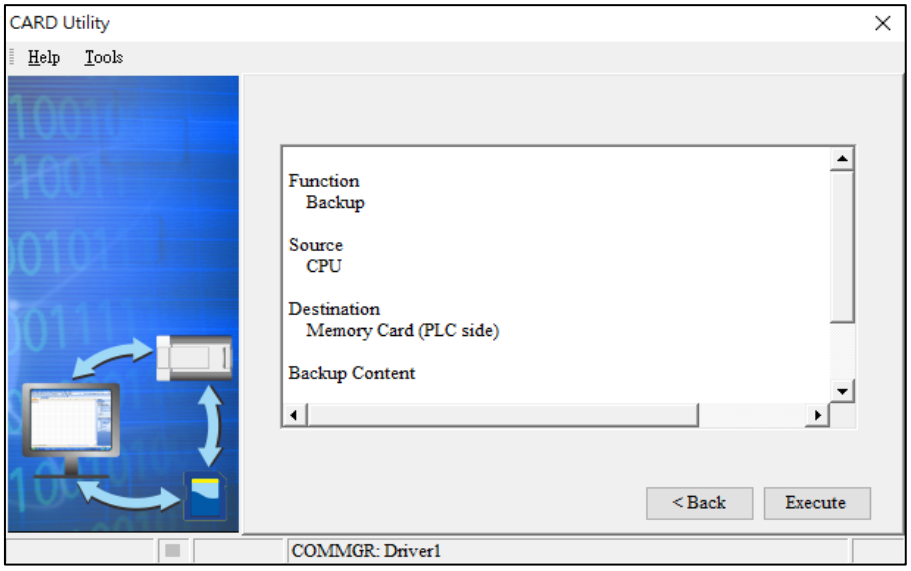
b. **PLC (DUP File)** is selected.

Click ··· to select a folder to set a filename and location, and then click **Next**.

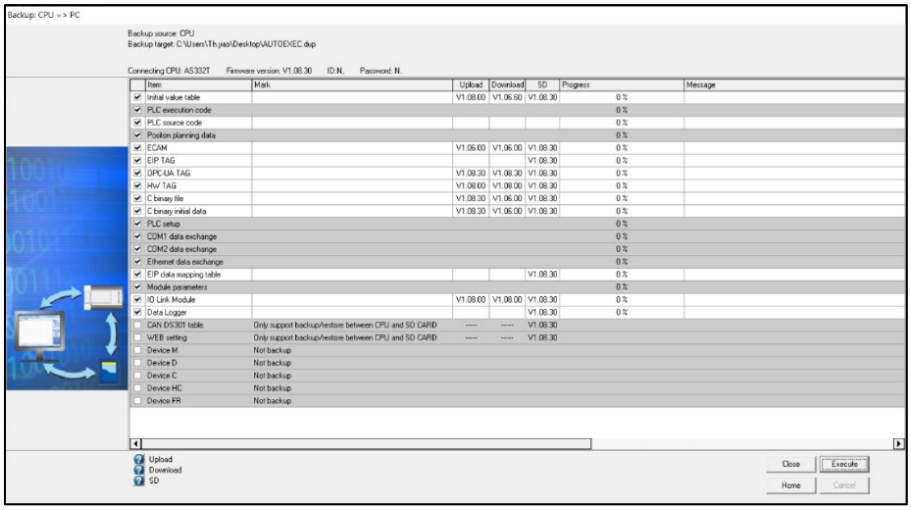
Note: If selecting the backup source as **DIADesigner Project (Compiled and Saved)** in step 4, you can set the PLC password in subsequent steps.



7. After confirming that the summary in **CARD Utility** dialog is consistent with the data backup to be performed, click **Execute** to start the backup.



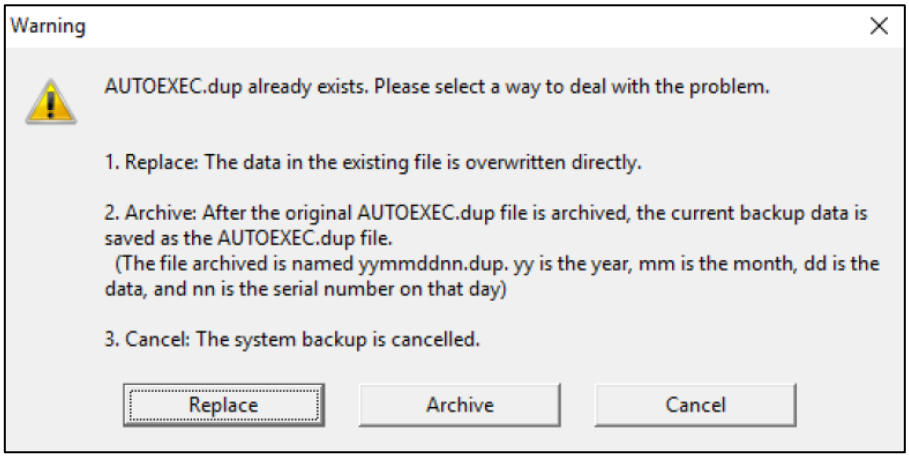
If **PC (DUP File)** is selected from the above step, the following dialog appears.



The items in a gray background cannot be unchecked, and the items in a white background can be checked upon the user's requirement. After the setting is complete, click **Execute** to start the data backup.

Even if you click **Cancel** to stop backing up data in the CPU module onto the memory card, the CPU module will still perform the data backup. At this time, you can power off the CPU module to stop the data backup. However, the produced backup file may be incomplete, so be sure to delete the backup file in the memory card afterwards.

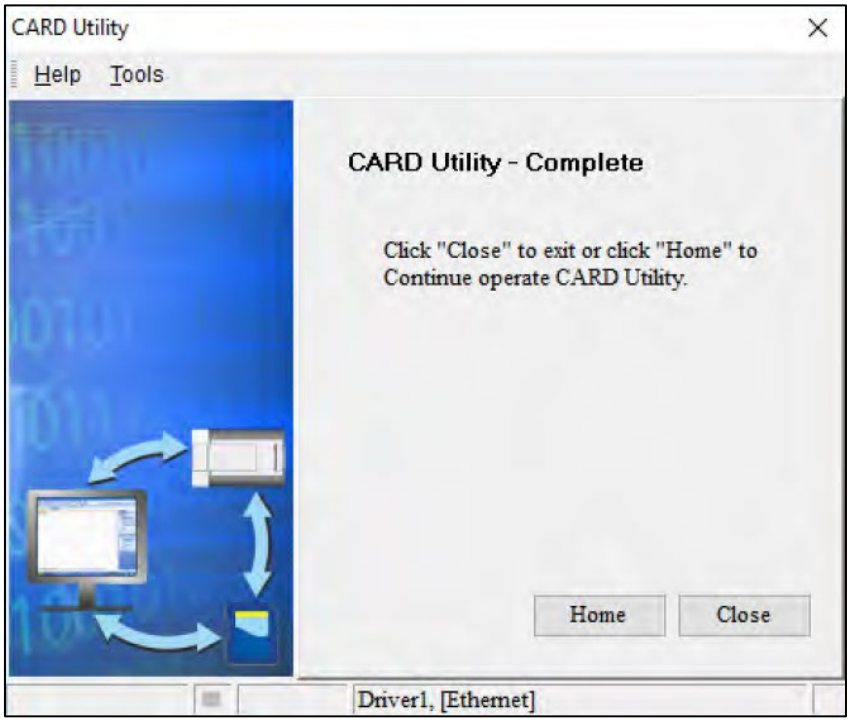
If **Memory Card (PLC side)** is selected from the above step, the filename and path will be **Root directory of the memory card\model type\SysDup\AUTOEXEC.dup**. If there is an old backup file in the memory card, a warning message appears. Follow the instructions in the dialog box.



If the backup source is protected by passwords, enter the password in the password field to continue the backup operations.

Data Backup	Description
CPU → Memory card	The **PLC ID** and **PLC password** set in the CPU module will be included in the backup file.
CPU → Computer	The system will ask you to enter the **PLC ID** and **PLC password** of the CPU module. The data backup will be performed once the information is confirmed by you. After the data backup is complete, the data of **PLC ID** and **PLC password** will be included in the backup file.

8. Once the data backup is complete, click **Home** to continue the operation or **Close** to close the dialog.



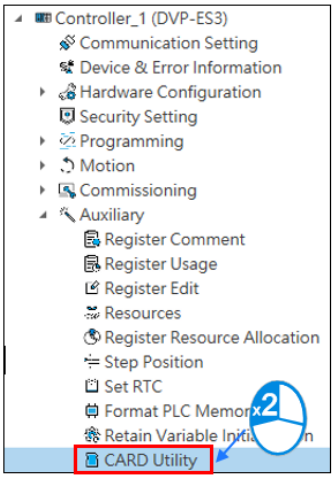
16.11.2 Restoration

When the restoration source / restoration destination contains a CPU or memory card, make sure that the connection is normal between DIADesigner and the CPU.

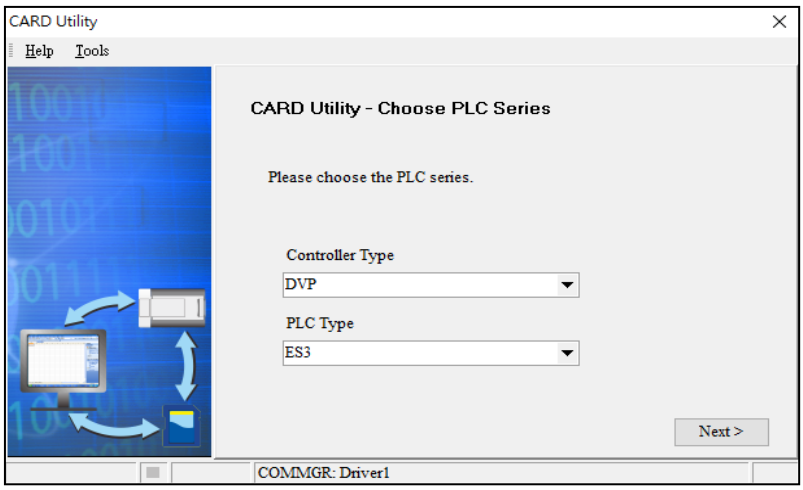
Follow these steps to restore the data.

To restore data

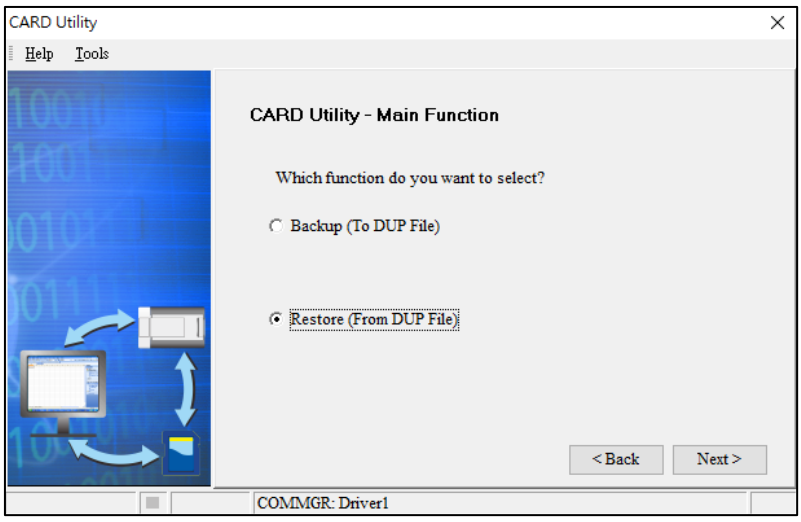
1. Create a device in **Project Explorer,** and double-click **Device** > **Auxiliary** > **CARD Utility**.



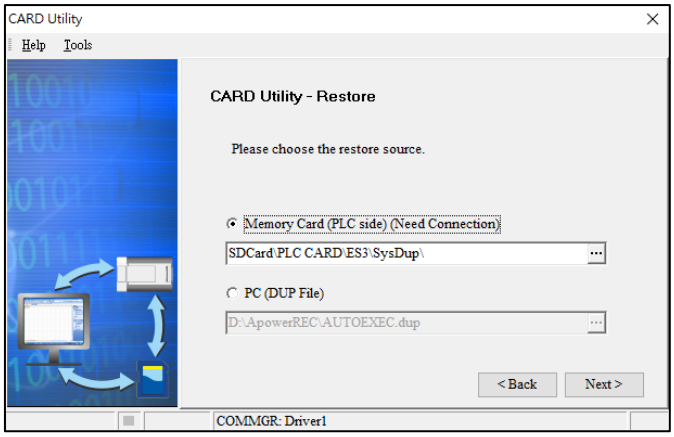
2. Click **Next** in the **CARD Utility - Choose PLC Series** dialog.



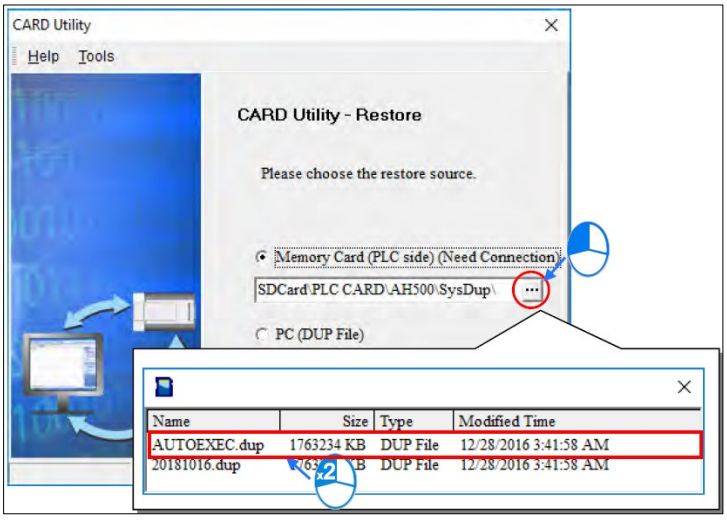
3. Select **Restore (From DUP File)** in the **CARD Utility – Main Function** dialog, and then click **Next**.



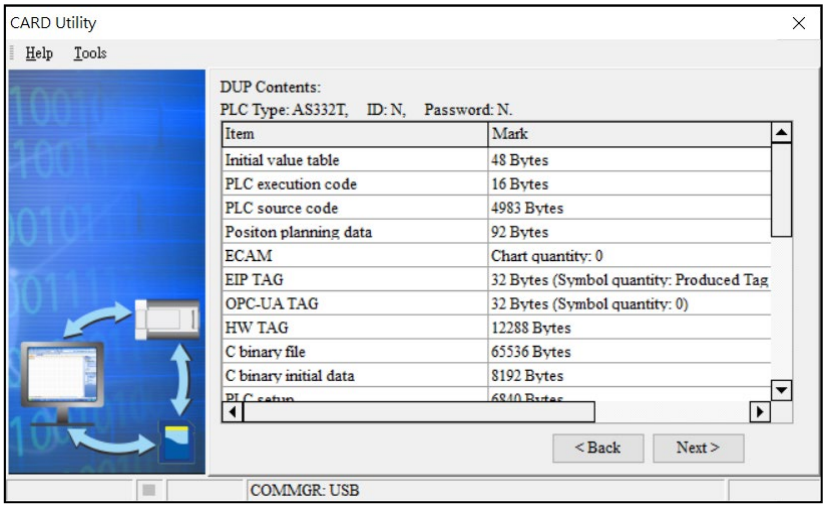
4. Click ··· to select a backup file to restore, and then click **Next**.



If **Memory Card (PLC side) (Need Connection)** is selected, the system will automatically connect to PLC and display all backup files in the memory card on the window, and then double-click the backup file to be restored.

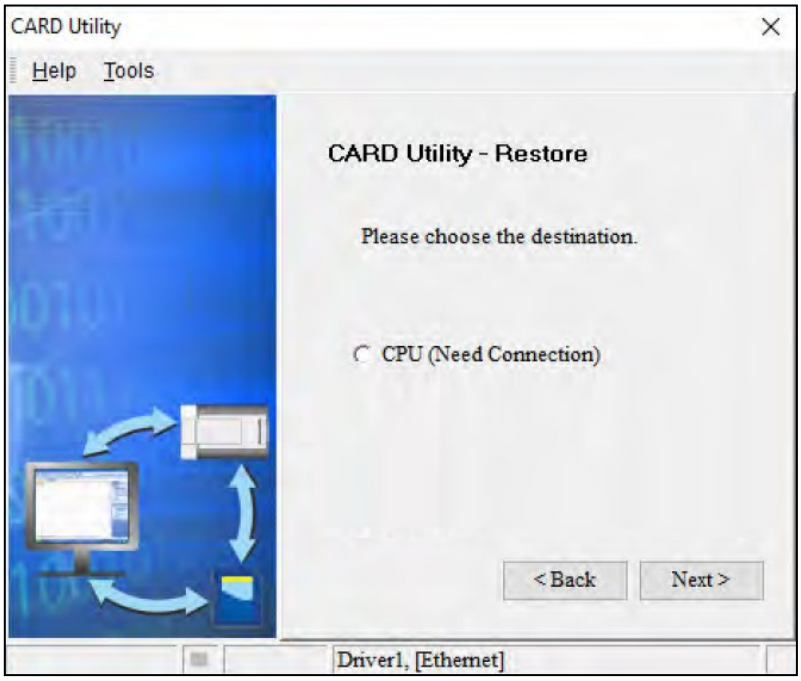


If **PC (DUP File)** is selected from the above step, the following dialog appears to check the restore content. Click **Next** to continue.

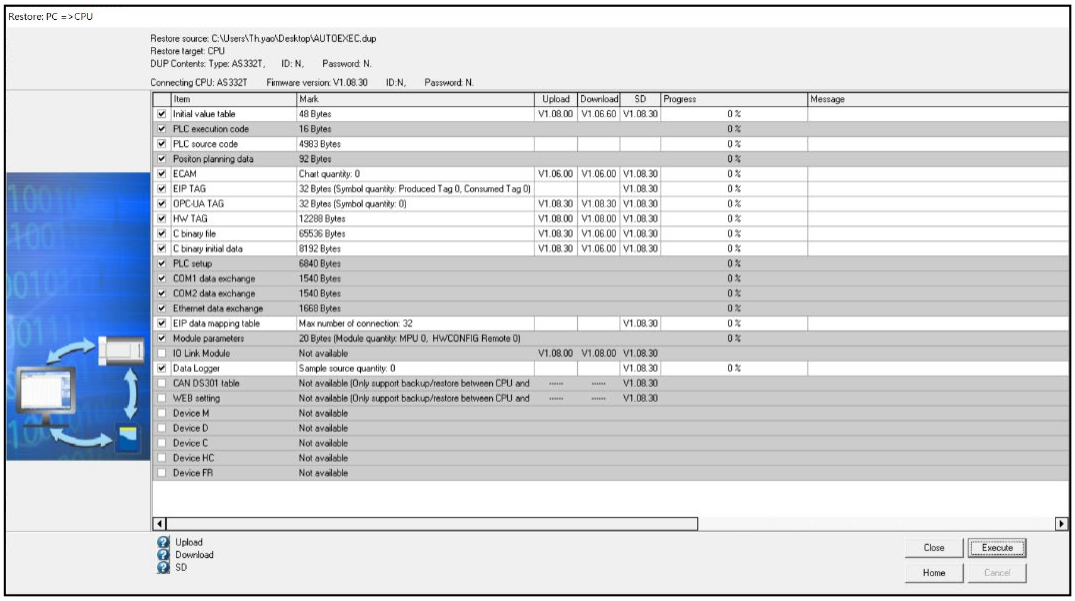


5. Choose the restoration destination, and then click **Next**.

   Note: The default restoration destination is only **CPU (Need Connection)**.



6. Select the checkbox of restoration items and click **Execute** to start the data restoration.



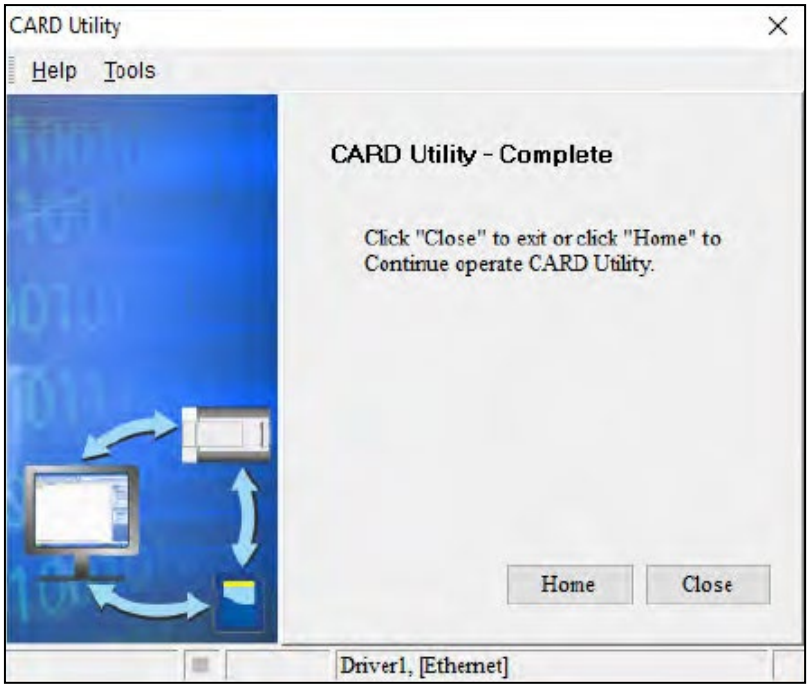
   If you click **Cancel** to stop restoring data from the memory card onto the CPU module, the restoration data may be incomplete. If you do not perform the data restoration anymore, be sure to factory reset the CPU module to avoid causing operation confusion. If the backup source is a memory card, even if you cancel

the process of data restoration, the CPU module will still perform the data restoration. At this time, you can power off the CPU module to stop the data restoration.

If the restore source is protected by passwords, follow the instructions below.

Data restoration	Description
Memory card → CPU module	• The **ID** in the backup file and the CPU module must be consistent, otherwise the data restoration will not be performed. • If there is a **PLC password** set in the CPU module, the **password** in the backup file and the CPU module must be consistent, otherwise the data restoration will not be performed. • If there is no **PLC password** set in the CPU module, and there is a **password** in the backup file, the system will perform the data restoration, and the **password** in the backup file will become the **PLC password** in the CPU module.
Computer → CPU module	• The **ID** in the backup file and the CPU module must be consistent, otherwise the data restoration will not be performed. • If there is a **PLC password** set in the CPU module, the **password** in the backup file and the CPU module must be consistent, otherwise the data restoration will not be performed and a prompt message will appear. • If there is no **PLC password** set in the CPU module, and there is a **password** in the backup file, the system will perform the data restoration, and the **password** in the backup file will become the **PLC password** in the CPU module.

7. Once the data restoration is complete, click **Home** to continue the operation or **Close** to close the dialog.



16.11.3 Command-line Instruction Execution

CARD Utility supports instructions execution in command-line mode. The execution file (PLC CARD.exe) can be found in the installation directory of DIADesigner. The default installation path is set as `C:\ProgramData\Delta Industrial Automation\DIAStudio\DIADesigner\Software\PLC_CARD`, and then continue to do the following the operation.

- Empty spaces are required to separate commands.
- /Lock and /UnLock cannot be used simultaneously as well as /C and /IP, /F and /V.
- /DLINK_USB, /DLINK_CDC, /DLINK_ETH cannot be used simultaneously.
- Commands and parameters are not case-sensitive; only password is set to be case-sensitive.
- Commands not required parameter inputs.

Command	Functions
/RUN or /R	Run PLC
/Stop	Stop PLC
/H	Hide all message dialogs, but still have return codes as execution finished.

Command	Functions
/SaveOFF	Disable the function of archiving returned codes.
/?	Display help messages. Other commands would be invalid while using this command.
/DLINK_USB	Use DirectLink function through Standard USB wire for communication. This command is only supported by DVP series and cannot be used with /C, /IP, /DLINK_ETH and /DLINK_CDC simultaneously.
/DLINK_CDC	Use DirectLink function through CDC port for communication. This command is only supported by DVP series and needs /C command to specify a communication port, which cannot be used with /IP, /DLINK_ETH and /DLINK_USB simultaneously.
/DLINK_ETH	Use DirectLink function through Ethernet for communication. This command is only supported by DVP series and needs /IP command to specify an IP address, which cannot be used with /C, /DLINK_CDC and /DLINK_USB simultaneously.

- Commands required parameter inputs

Command	Parameter	Functions
/PLC=P1	P1=PLC model type	Specify the PLC model type for connection. e.g./PLC=AS332T. Supported model types: AS300N, AS320T, AS320P, AS324MT, AS332T, AS332P AS218TX, AS218PX, AS218RX, AS228T, AS228P, AS228R, AS132T, AS132P, AS132R, AS148T, AS148P, AS1148R, AS164T, AS164P, AS164R, ES3, EX3, SX3, SV3
/F=P1	P1=file name of DUP	Download DUP file to PLC, e.g./F=test.dup
/C=P1	P1=COM Port Number	Assign the COM Port number of PC, e.g./C=1
/Lock=P1, P2	P1=password P2=the limited number of retries (Set 0 mean no limits to the number of retries)	If the PLC is without password protection and no password is in the DUP file, you can set the password with this command after the download is complete, e.g./Lock=1234, 0 **Note**: If there's a password in DUP file, it would be automatically set as the

Command	Parameter	Functions
		password for PLC after the download is complete.
/UnLock=P1	P1=password	If the PLC has been secured by password lock and a same password does not exist in the DUP file, you must remove PLC password before downloading, e.g./Unlock=1234 **Note**: You can set password again by using /Lock command or downloading a DUP file which includes a password.
/P=P1, P2, P3, P4	P1=Baud Rate P2=Data Bits P3=Parity P4=Stop Bits	Set the communication protocol. The default value is 9600, 7, e, 1. Leave blank for values remain unchanged, while "," is necessary to be input. **Example**: • Change all the setting values. /p=115200, 8, n, 2. • Only change the Baud Rate. /p=115200, , ,. • Must input ", ".
/Save=P1	P1=File name	Assign the path and the file name to save returned codes. If a specified path directory does not exist, it will be generated automatically. A default file name "Result.txt" will be taken as the current path when this command is not used or the parameter is empty. **Example**: /Save=C:\test\result1.txt Find the return codes table below for the corresponding messages.
/IP=P1	P1=IP address	Assign Ethernet IP address. When this command is used, it means to communicate via Ethernet, which cannot be used with /C command. **Example**: /IP=192.168.1.5
/T=P1,P2	P1=Source file P2=Target file	Convert the file format between DVZ and DUP. When using this command, other commands would be invalid. This command is supported by DVP series only.
/V=P1	P1=Name of the file to	Verify PLC program. This command is supported by DVP series only.

Command	Parameter	Functions
	verify with PLC program.	**Example**: /V=test.dup

Command	Parameter	Functions
/STN=P1	P1=Station number	Set the station number for communication with PCs. For example, set /STN=5 to communicate with the PLCs, which station number is 5. If the command is omitted, message would be sent via broadcast communication.
/SetPLCSTN=P1	P1=Station number	Change station number of a PLC. For example, set /SetPLCSTN=5 to change the station number of the PLC to 5. This command is supported by DVP series only.
/B=P1,P2,P3,P4	P1= The save as DUP file name P2= The password to verify P3= The verify ID P4= Whether to back up the device (N: No, Y: Yes)	Read the PLC program as well as the device data and save as DUP files. This command is supported by DVP series only. **Example**: • There's no password and ID in the PLC. Device backup is not required. /B=p1.dup,,,N • There's no ID in the PLC but with password-lock. Device backup is required. /B=p1.dup,pw99,,Y If P4 is blank, the default setting is to back up the device.
/BP=P1,P2,P3,P4	P1= The target DUP file name P2= The source project file name (dvp/isp) P3= Source register file (dvl) P4= Source file register (wft)	Backup the specified project file in DUP file format. The input values of P1 and P2 are required, but P3 and P4 can be omitted. This command is supported by DVP series only. For example, save test.dvp backup as out.dup. /BP=out.dup,test.dvp /BP=d:\dir\out.dup,d:\dir1\test.dvp The specified directory would be automatically created if it does not exist, while the existing file would be overwritten.

Example

- Download **test.dup** to **PLC.AS332T** and run PLC with the default value of communication setting (9600, 7, e, 1) via COM port1.
  - File directory under the current path: "PLC CARD.exe" /PLC=AS332T /F=test.dup /RUN /C=1
  - File directory under the appointed path: "d:\temp\PLC CARD.exe" /PLC=AS332T /F=d:\temp\test.dup
  - /RUN /C=1
- Stop PLC AS324MT via Ethernet.

  "PLC CARD.exe" /PLC=AS324MT /Stop /IP=192.168.1.1
- Ex3: Change the station number of PLC to 2 via COM port 1. The station number is originally set to 1.

  "\PLC CARD.exe" /PLC=SS2 /C=1 /STN=1 /SetPLCSTN=2

Return Codes and Corresponding Messages

Code	Message
1	Finish!
2	Timeout!
3	The PLC type does not match!
4	DUP File is error!
5	Communication port has been used or not this port!
6	Cannot open file
7	Command is incorrect!
8	Command H is incorrect!
9	Command P is incorrect!
10	Station number is incorrect!.
11	COM port number is incorrect!
12	Command R is incorrect!
13	Baud rate is incorrect!
14	Data bits is incorrect!

Code	Message
15	Parity is incorrect!
16	Stop bits is incorrect!
17	Command STOP is incorrect!
18	Command V is incorrect!
19	Password Locked!
20	Can't get correct program!
21	Can't get actuating code in the PLC for comparison!
22	Reserved.
23	Command C is incorrect!
24	Command B is incorrect!
25	Command Lock is incorrect!
26	Command UnLock is incorrect!
27	Password Verification Failure!
28	Failed to save the backup file!
29	Locked. Please remove first!
30	Command IP is incorrect!
31	Can't get actuating code in the PLC to read!
32	Can't get correct PLD ID status!
33	PLC ID compare failure!
34	Communication error!
35	Downloads have reached the limit!
36	Machine not support!
37	Read PLC failure!
38	Write PLC failure!
39	Command failure!
40	Command PLC is incorrect!
41	The command has been cancelled by users.
42	Command STN or SetPLCSTN is incorrect!
43	Command BP is incorrect!
44	The version of EasyDownload is incorrect!

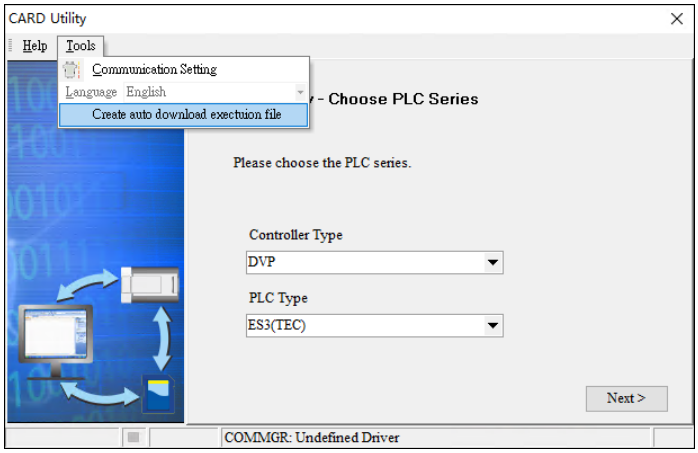
16.11.4 Create Auto Download Executable File

Transfer the backup file to another computer through the **Create auto download execution file** function of CARD Utility and directly execute the executable file for backup or restoration without installing or opening DIADesigner software.

Follow these steps to create an auto download-executable file.

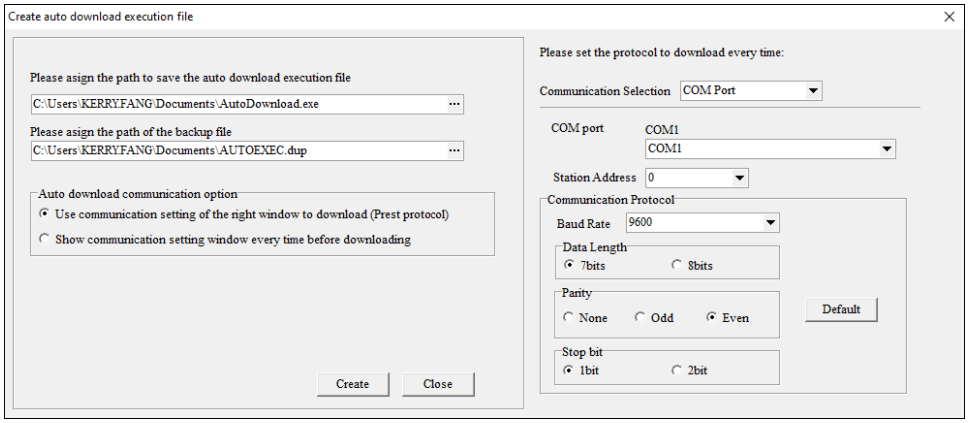
To create an auto-download executable file

1. In **Project Explorer**, add a device, and double-click **Device** > **Auxiliary** > **CARD Utility**.
2. In the **CARD Utility** dialog, click **Tools** > **Create auto download execution file**.



3. In the **Create auto download execution file** dialog, set **Please assign the path to save the auto download execution file**, **Please assign the path of the backup file**, and the **Auto download communication option**.

   Note: If selecting **Use communication setting of the right window to download (Preset protocol)** for **Auto download communication option**, the communication settings on the right are used for communication when restoring the backup file; if selecting **Show communication setting window every time before downloading**, the communication method can be set again according to the field situation when executing the executable file.



4. Click **Create**.

   The executable file is created on the specific path. Double-click the executable file to perform backup or restoration.

Chapter 17: Other Functions

17.1 Find and Replace

You can find text objects in DIADesigner with the help of **Find** feature and replace the texts with the help of **Replace** feature. There is also a batch replacement register, which can be used as a batch replacement register function.

The scope of implementation of batch replacement register function is as follows.

- Program POU (including LD, ST, SFC language) and local variable table
- Function block POU (including LD, ST language) and local variable table
- Global Variable Table
- DUT – Structure and Enumeration and Union.

The **Find** and **Replace** function can be used in following contexts.

- POU – LD, ST, SFC, C Programs and Local Variable Table
- POU – LD, ST, C Function Blocks and FB Variable Table
- Global Variable Table

17.1.1 Find – POU

Open any POU and click the POU working area or the local variable table to start Find operation.

Follow these steps to perform a find operation in DIADesigner POU.

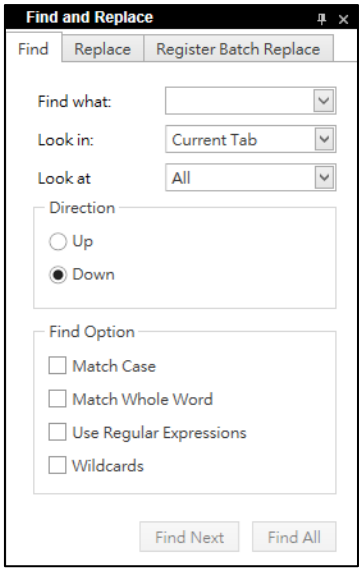
To perform a find operation in DIADesigner POU

1. Open a POU.
2. Click **Edit** > **Find and Replace** > **Find**.

   Or

   Press **Ctrl** + **F** keys simultaneously on the keyboard.

   Find and Replace dialog appears with **Find** tab opened by default.



The following table describes the functions available in the **Find** tab in **Find and Replace** dialog.

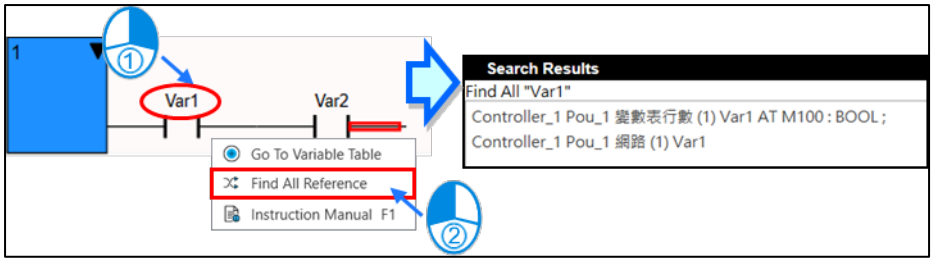
Function	Description
Find what	Enter the object name which is searched for, or select an object which has been searched.
Look in	Select the area to look into. Available options are: • **Current Tab**: Find current tab open in working area. • **Current Device**: Find all POUs that belong to current focus device in **Project Explorer**. • **Current Project**: Find POU of every device in the project. **Note**: The default value is **Current Tab**.
Look at	Search **All**, **Variable Name**, and **Instructions**.
Match Case	Click to select the matching case search.
Match Whole Word	Click to select the whole word case search.
Use Regular Expressions	Click to use the regular expressions. (Refer to the related technical documents for more information.)
Wildcards	Click to use the wildcard expressions. (Refer to the related technical documents for more information.)
Direction - Up	Click to select among previous instances.
Direction - Down	Click to select among instances after the present instance.
Find Next	Click to find the next instance.
Find All	Click to find all the instances.

Follow these steps to perform a search on the **Find and Replace** panel.

To perform a search on the Find and Replace panel

1. Enter the text to be searched in the **Find what** field, select the options and directions as per your requirements.
2. Click **Find Next** or **Find All** to open **Search Results** dialog that displays all the occurrences of the text in search.

You can also directly right-click the Variable in the program area, and then click **Find All Reference** in the context menu to find out all the locations where this variable has been used in the project.



17.1.2 Replace – POU

Follow these steps to perform a replace operation in DIADesigner POU.

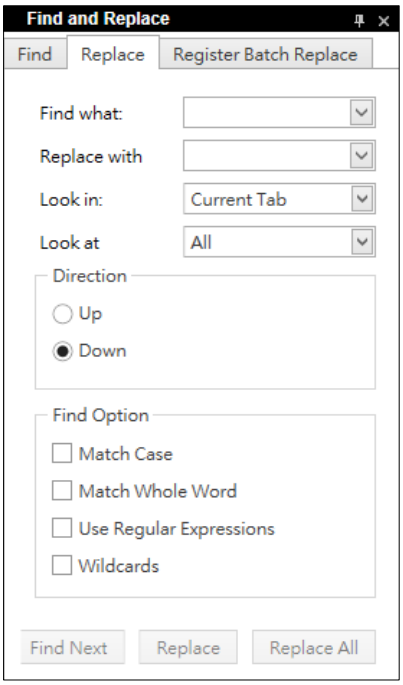
To perform a replace operation in DIADesigner POU

1. Open a POU.
2. Click **Edit** > **Find and Replace** > **Replace**.

   Or

   Press **Ctrl** + **H** keys simultaneously on the keyboard.

   The **Find and Replace** dialog appears with **Replace** tab.



The following table describes the functions available in **Replace** tab in the **Find and Replace** panel.

Function	Description
Find what	Enter the object name which is searched for, or select an object which has been searched.
Replaced with	Enter the object name which is replacing the searched text in this operation.
Look in	Select the area to look in. The available options include: • Current Tab: Find current tab open in working area. • Current Device: Find all POUs that belong to current focus device in Project Explorer. • Current Project: Find POU of every device in the project. **Note**: The default value is **Current Tab**.
Look at	Search **All**, **Variable Name**, and **Instructions**.
Match Case	Click to select the matching case search.
Match Whole Word	Click to select the whole word case search.
Use Regular Expressions	Click to use the regular expressions. (Refer to the related technical documents for more information.)
Wildcards	Click to use the wildcard expressions. (Refer to the related technical documents for more information.)
Direction – Up	Click to select among previous instances.
Direction – Down	Click to select among instances after the present instance.

Function	Description
Find Next	Click to find the next instance.
Replace	Click to replace the current instance.
Replace All	Click to replace all the instances.

Follow these steps to perform the replacement operation on the **Find and Replace** panel.

To perform the replacement operation on Find and Replace panel

1. Enter the text to be searched in the **Find what** field, select the options and directions as per your requirements.
2. Click **Replace** or **Replace All**.

   Search result(s) is replaced by the button clicked in the **Replace** tab.

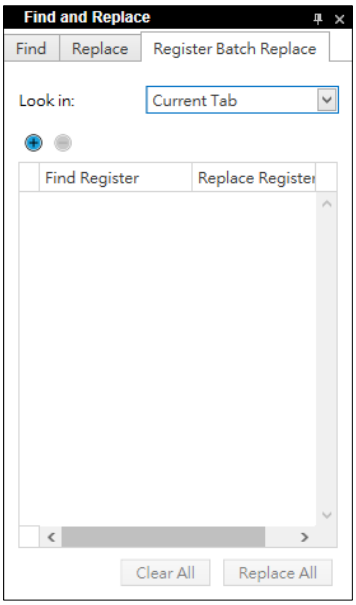
17.1.3 Batch Replacement Register

You can open the POU and click the editing workspace or the local variable table, and then perform the batch replacement operation.

Follow these steps to perform batch replacement register operation in the POU of DIADesigner.

To perform batch replacement register operation in the POU of DIADesigner

1. Open POU.
2. Open **Find and Replace** panel, and click **Register Batch Replace** tab to open **Find and Replace** dialog with **Register Batch Replace** tab.



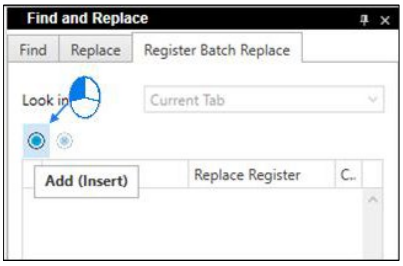
The following table describes the functions of the **Register Batch Replace** tab.

Function / Icon	Description
Inquire	The supported replacement ranges are as follows: • Current tab **Note**: The current tab is the default value.
	Click this item to add setting **Find Register** and **Replace Register**.
	Click this item to delete the set **Find Register** and **Replace Register**.
Find Register	Enter the search register.
Replace Register	Enter the register to replace.
Consecutive Number	Enter the number of registers to replace.
Clear All	Clear all **Find Register** and **Replace Register** items.
Replace All	Replace the **Find Register** content with the **Replace Register** content. (The replacement quantity varies according to the setting of **Continuous Counts**.)

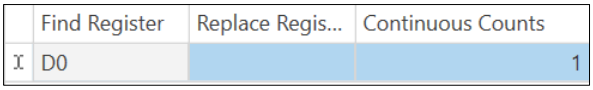
Follow these steps to perform the replacement operation on the **Batch Replace Register** tab.

To perform replacement operation on Batch Replace Register tab

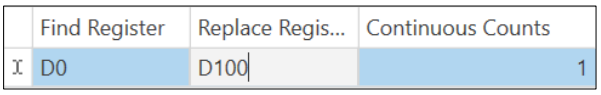
1. Click **Add**.



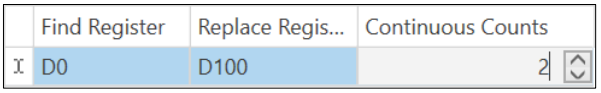
2. Enter the Register to search in the **Find Register** column.



3. Enter the Register to replace in the **Replace Register** column.



4. Set the number of Registers in the **Continuous Counts** column.

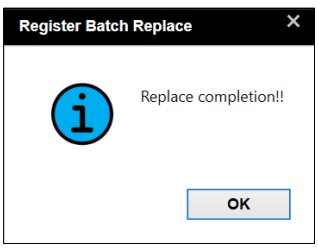


Note: The maximum input limit for the **Continuous Counts** field is 999.

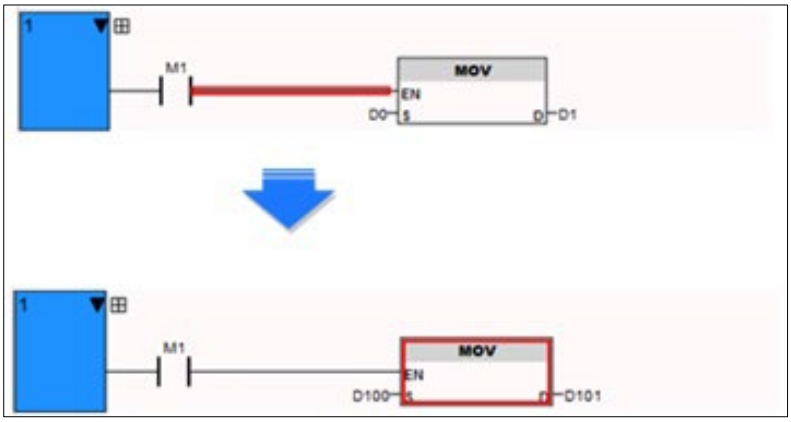
5. Click **Replace All**.



6. Click **OK** to complete the replacement.



If a batch is executed to replace the registers, all searchable registers will be replaced.



17.2 Bookmark

Bookmark is used in a POU to mark the location(s) – a line in a Structured Text or a Network in Ladder Diagram POU. Bookmark function is similar to a bookmark which a reader inserts in a book to remember the page the reader has to open or the pages where some important read out are located.

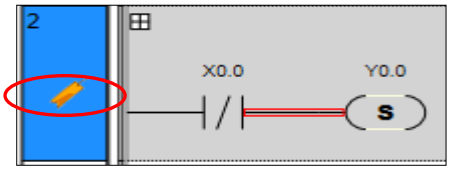
Perform the following actions with Bookmarks.

- Create Bookmark
- Go to Next Bookmark
- Go to Previous Bookmark
- Delete Bookmark
- Clear All Bookmarks

All the Bookmarks created appear in the **Bookmark** dialog. POU in which Bookmarks are created are shown in a tree structure and all the Network names with Bookmarks are shown in the branch level. Open **Bookmark** dialog from **View** > **Bookmark** in the menu bar.

17.2.1 Bookmark in Ladder Diagram (LD) POU

In Ladder Diagram POU, the bookmark is located at the line number header on the left (the blue area in the figure below). The following figure is an example of a bookmark in a LD Language.



The subsequent chapters will describe the actions, such as adding, moving, and deleting bookmarks.

17.2.1.1 Create Bookmark – LD POU

Follow these steps to create Bookmark in LD POU.

To create Bookmark in LD POU

1. Select the Network(s) where Bookmark has to be created.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Toggle Bookmark** to add the bookmark.

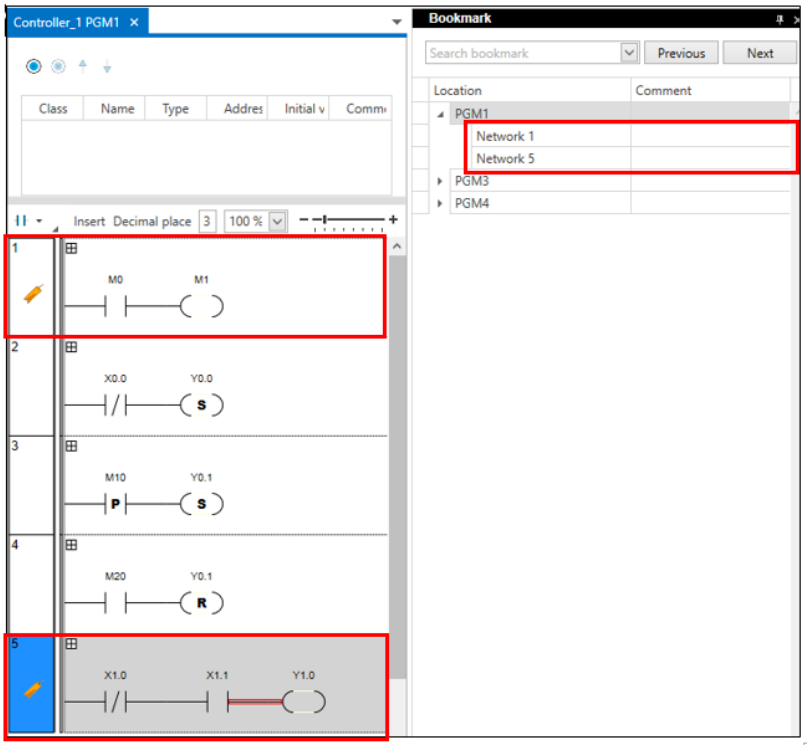
   Or

   Click **Edit** > **Bookmark** > **Toggle Bookmark** in the menu bar.

   Or

   Click **Toggle Bookmark**  on the toolbar of the POU dialog to add the bookmark. The hotkey for the toggling the bookmark is **CTRL** + **F12**.

   Example: The following figure shows Networks – Network 1 and Network 5 with Bookmark.



17.2.1.2 Next Bookmark in Tab – LD POU

Follow these steps to navigate to **Next Bookmark in Tab** in an LD POU.

To navigate to Next Bookmark in Tab in LD POU

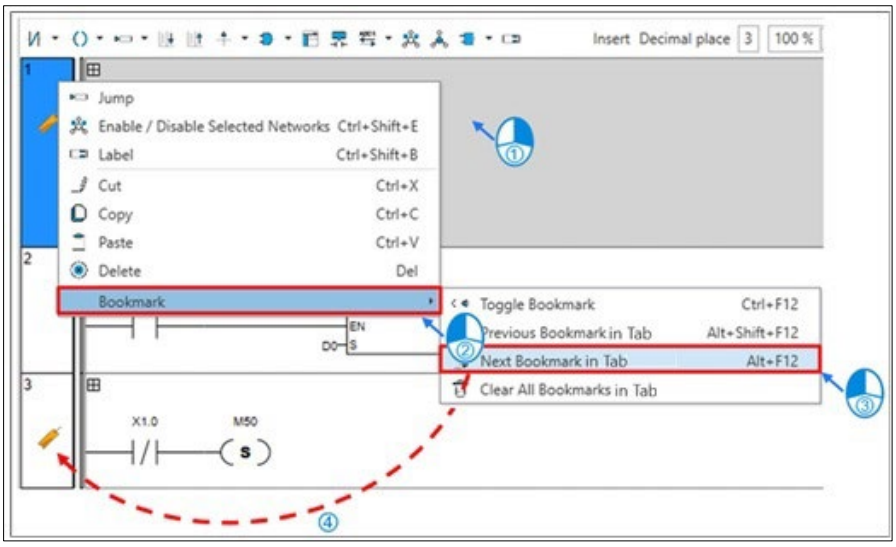
1. Select the Network with Bookmark.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Next Bookmark in Tab** to move to the next bookmark.

   Or

   Click **Edit** > **Bookmark** > **Next Bookmark in Tab** in the menu bar.

   Or

   Click **Next Bookmark in Tab**  on the toolbar of the POU dialog to move to the next bookmark location. The hotkey for moving to the next bookmark location is **ALT** + **F12**.



Note: Select the Bookmark in the Bookmark Manager, and click **Enter** on the keyboard to edit the Bookmark.

17.2.1.3 Previous Bookmark in Tab – LD POU

Follow these steps to navigate to **Previous Bookmark in Tab** in an LD POU:

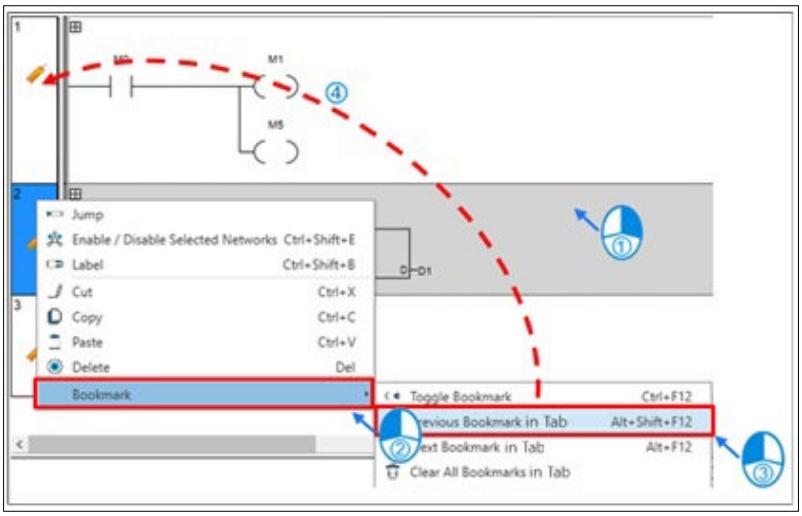
To navigate to Previous Bookmark in Tab in LD POU

1. Select the Network with Bookmark.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Previous Bookmark in Tab** to move to the previous bookmark. or

   Click **Edit** > **Bookmark** > **Previous Bookmark in Tab** in the menu bar.

   Or

   Click **Previous Bookmark in Tab**  on the toolbar of the POU dialog to move to the previous bookmark. The hotkey for moving to the previous bookmark location is **ALT** + **Shift** + **F12**.



17.2.1.4 Delete Bookmark – LD POU

Follow these steps to delete the Bookmark(s) in an LD POU.

To delete Bookmark(s) in LD POU

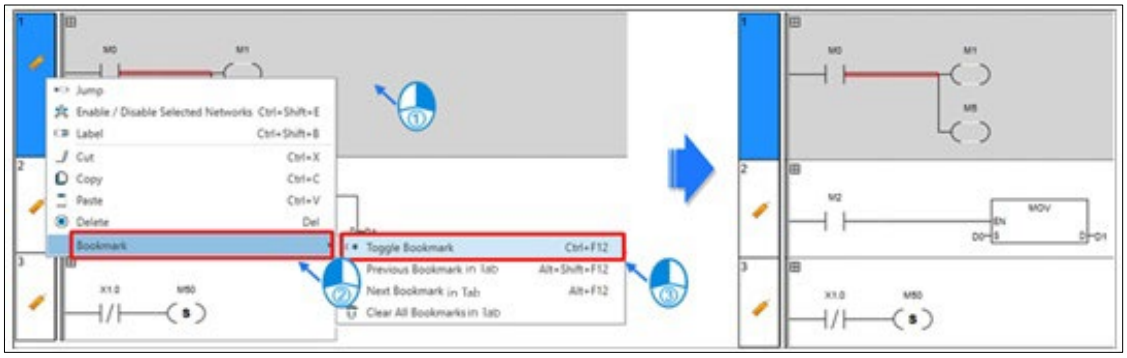
1. Select the Network(s) where Bookmark is / are created.
2. Right-click to open the context menu
3. Click **Bookmark** > **Toggle Bookmark** to delete the bookmark

   Or

   Click **Edit** > **Bookmark** > **Toggle Bookmark** in the menu bar.

   Or

   Click **Toggle Bookmark**  on the toolbar of the POU dialog to delete the bookmark. The hotkey for deleting the bookmark is **CTRL** + **F12**.



17.2.1.5 Clear All Bookmarks in Tab – LD POU

Follow these steps to delete all the Bookmarks in an LD POU.

To delete all Bookmarks in LD POU

1. Right-click to open the context menu
2. Click **Bookmark** > **Clear All Bookmarks in Tab** to delete all the bookmarks.

   Or

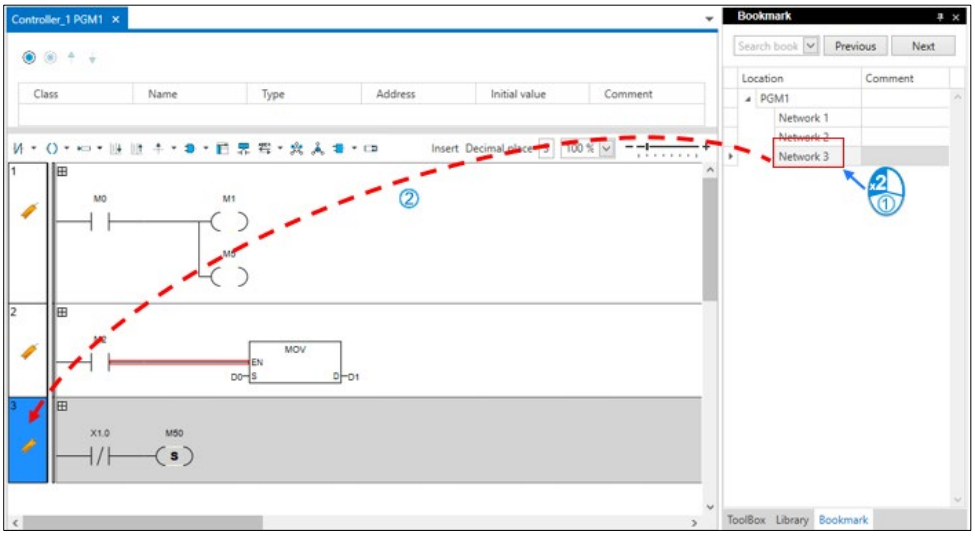
   Click **Edit** > **Bookmark** > **Clear All Bookmarks in Tab** in the menu bar.

   Or

   Click **Clear All Bookmarks in Tab** on the toolbar of the POU dialog to delete all bookmarks.

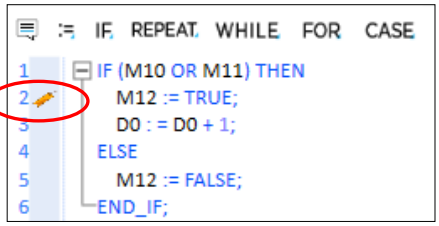


Note: You can click the Bookmark location in the **Bookmark** tab to navigate to the Bookmark as shown in the following figure. Double-click the Network name below **Location** in the **Bookmark** dialog to navigate to the respective Bookmark.



17.2.2 Bookmark in Structured Text (ST) POU

In Structured Text POU, the bookmark is located at the line number header on the left (the blue area in the figure below). The following figure is an example of a bookmark in ST Language.



The subsequent chapters will describe the actions, such as adding, moving, and deleting bookmarks.

17.2.2.1 Create Bookmark – ST POU

Follow these steps to create Bookmark(s) in an ST POU.

To create Bookmark(s) in ST POU

1. Select the line where Bookmark has to be created.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Toggle Bookmark** to add the bookmark.

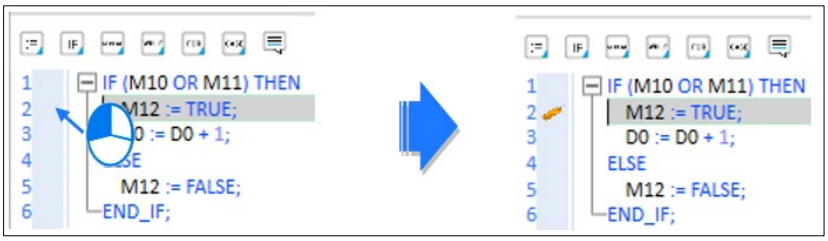
   Or

   Click **Edit** > **Bookmark** > **Toggle Bookmark** in the menu bar.

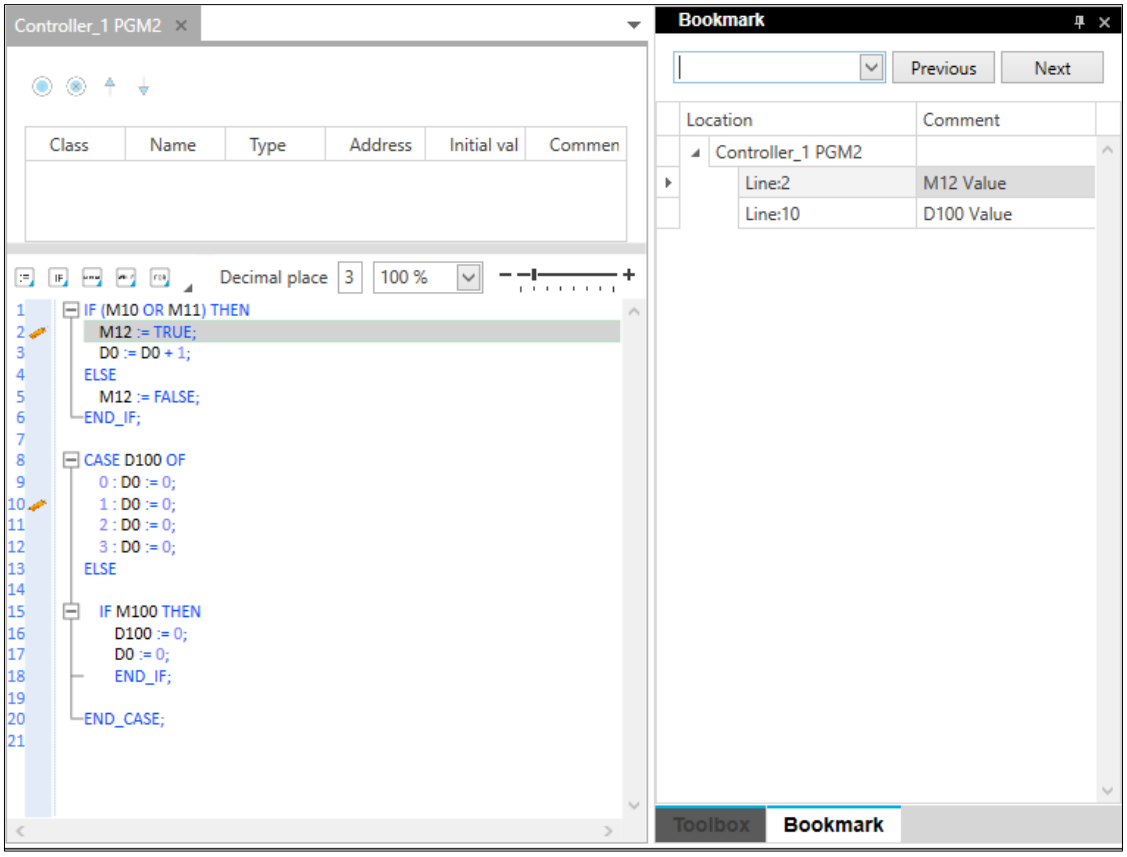
   Or

   Click **Toggle Bookmark**  on the toolbar of POU dialog to add the bookmark.

The hotkey for the toggling the bookmark is **CTRL** + **F12**.



Example: Line number 2 and line number 10 are bookmark positions. It can be observed that in addition to displaying the bookmark icon in the line number header of the structured language, the bookmark manager on the right side also displays all the bookmark positions marked by the POU.



Note: Select the Bookmark in the Bookmark Manager, and click **Enter** on the keyboard to edit the Bookmark.

17.2.2.2 Next Bookmark in Tab – ST POU

Follow these steps to navigate to **Next Bookmark in Tab** in an ST POU.

To navigate to Next Bookmark in Tab in ST POU

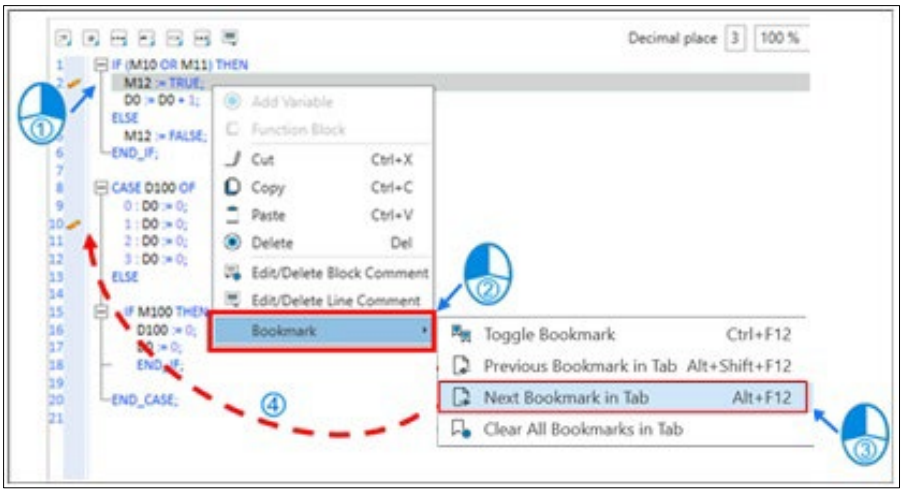
1. Select the line with Bookmark.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Next Bookmark in Tab** to move to the next bookmark.

   Or

   Click **Edit** > **Bookmark** > **Next Bookmark in Tab** in the menu bar.

Or

Click **Next Bookmark in Tab** on the toolbar of POU dialog to move to the next bookmark location. The hotkey for moving to the next bookmark location is **ALT** + **F12**.



17.2.2.3 Previous Bookmark in Tab – ST POU

Follow these steps to navigate to **Previous Bookmark in Tab** in an ST POU.

To navigate to Previous Bookmark in Tab in ST POU

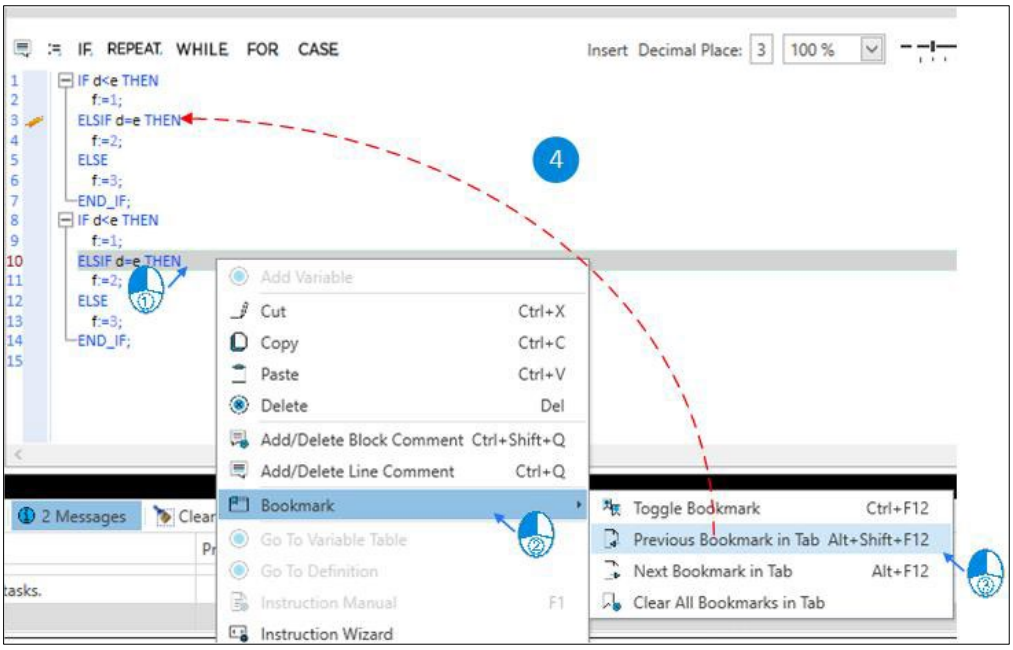
1. Select the line with Bookmark.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Previous Bookmark in Tab** to move to the previous bookmark.

   Or

   Click **Edit** > **Bookmark** > **Previous Bookmark in Tab** in the menu bar.

   Or

   Click **Previous Bookmark in Tab** on the toolbar of POU dialog to move to the previous bookmark. The hotkey for moving to the previous bookmark location is **ALT** + **Shift** + **F12**.



17.2.2.4 Delete Bookmark – ST POU

Follow these steps to delete the Bookmark in an ST POU.

To delete Bookmark in ST POU

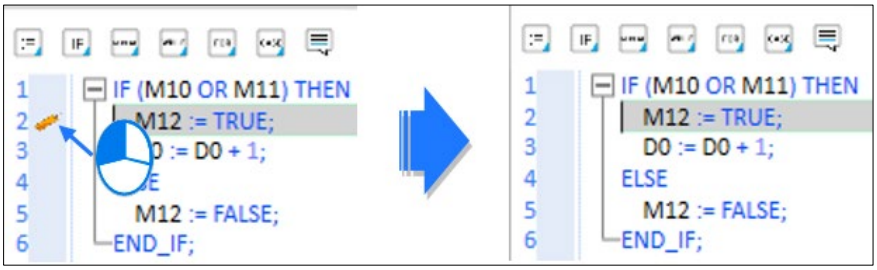
1. Select the line where Bookmark is created.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Toggle Bookmark** to delete the bookmark.

   Or

   Click **Edit** > **Bookmark** > **Toggle Bookmark** in the menu bar.

   Or

   Click **Toggle Bookmark**  on the toolbar of POU dialog to delete the bookmark. The hotkey for deleting the bookmark is **CTRL** + **F12**.



17.2.2.5 Clear All Bookmarks in Tab – ST POU

Follow these steps to delete all the Bookmarks in an ST POU.

To delete all Bookmarks in ST POU

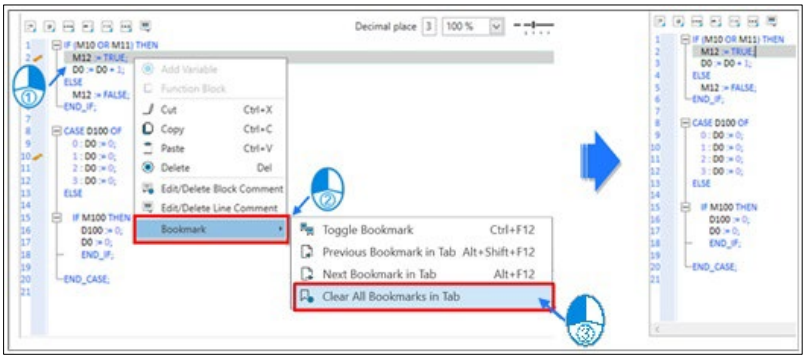
1. Right-click to open the context menu.
2. Click **Bookmark** > **Clear All Bookmarks in Tab** to delete all the bookmarks.

   Or

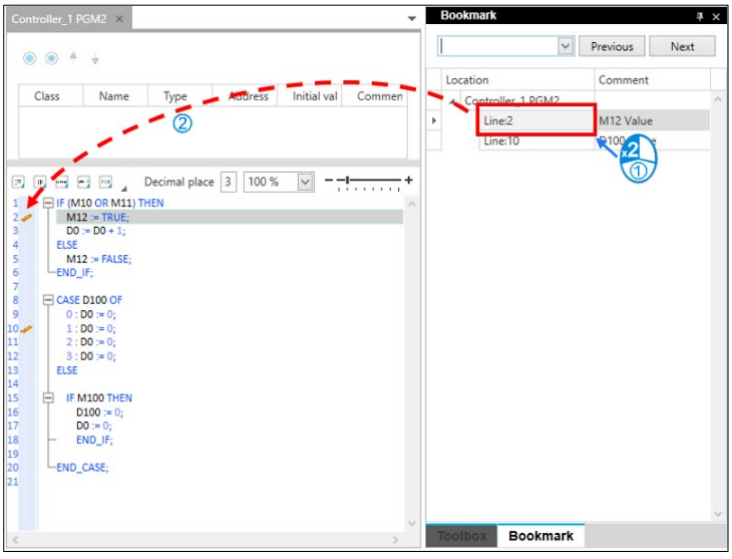
   Click **Edit** > **Bookmark** > **Clear All Bookmarks in Tab** in the menu bar.

   Or

   Click **Clear All Bookmarks in Tab**  on the toolbar of POU dialog to delete all the bookmarks.

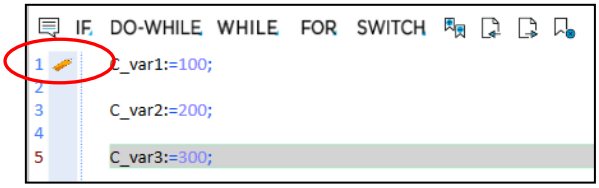


Note: You can click the Bookmark location in the **Bookmark** tab to navigate to the Bookmark as shown in the following figure. Double-click the **Line** name below **Location** in the **Bookmark** dialog to navigate to the respective Bookmark.



17.2.3 Bookmark in C Language

In C POU, the bookmark is located at the header of the line on the left (the blue area in the figure below). The following figure is an example of a bookmark in C Language.



The subsequent sections describe the actions, such as adding, moving, and deleting bookmarks.

17.2.3.1 Create Bookmark – C POU

Follow these steps to add bookmarks in C language.

To add bookmarks in C language

1. Select the row position to add a bookmark.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Toggle Bookmark** to add the bookmark.

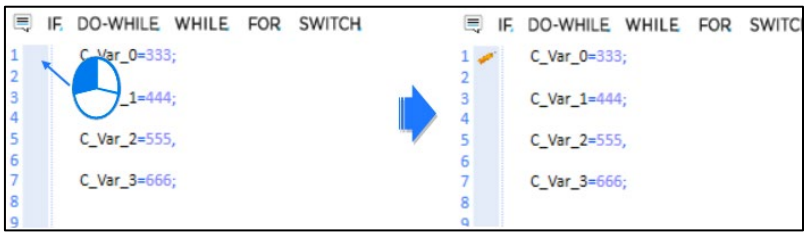
   Or

   Click **Edit** > **Bookmark** > **Toggle Bookmark** in the menu bar.

   Or

   Click **Toggle Bookmark**  on the toolbar of POU dialog to add the bookmark.

   The hotkey for the toggling the bookmark is **CTRL** + **F12**.



Example: Line number 1 and line number 7 are the bookmark locations. The **Bookmark Manager** on the right displays all the bookmark locations marked by the POU.



Note: Select the Bookmark in the Bookmark Manager, and click **Enter** on the keyboard to edit the Bookmark.

17.2.3.2 Next Bookmark in Tab – C POU

Follow these steps to move to **Next Bookmark in Tab** in C language.

To move to Next Bookmark in Tab in C language

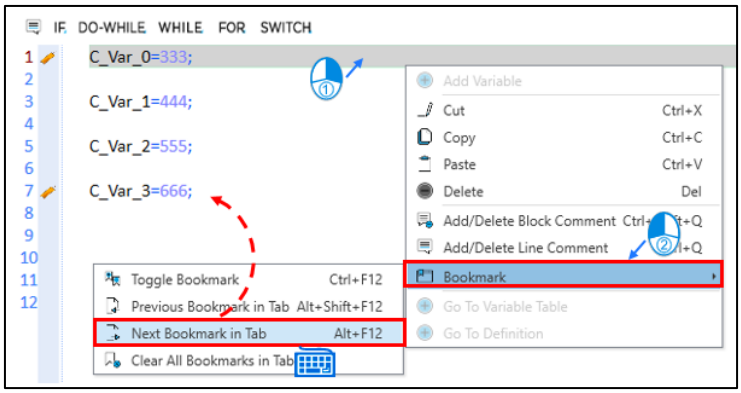
1. Select the line with Bookmark.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Next Bookmark in Tab** to move to the next bookmark.

   Or

   Click **Edit** > **Bookmark** > **Next Bookmark in Tab** in the menu bar.

   Or

   Click **Next Bookmark in Tab**  on the toolbar of POU dialog to move to the next bookmark location. The hotkey for moving to the next bookmark location is **ALT** + **F12**.



17.2.3.3 Previous Bookmark in Tab – C POU

Follow these steps to move to **Previous Bookmark in Tab** in C language.

To move to Previous Bookmark in Tab in C language

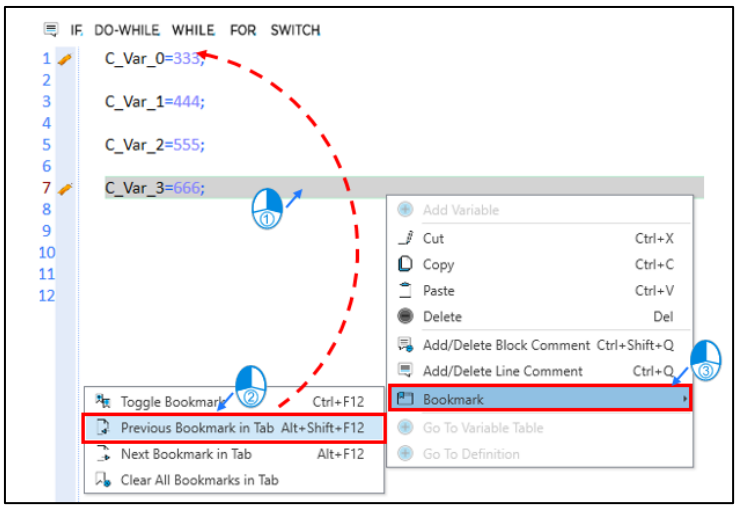
1. Click the line with Bookmark.
2. Right-click to open the context menu.
3. Click **Bookmark** > **Previous Bookmark in Tab** to move to the previous bookmark.

   Or

   Click **Edit** > **Bookmark** > **Previous Bookmark in Tab** in the menu bar.

   Or

   Click **Previous Bookmark in Tab**  on the toolbar of POU dialog to move to the previous bookmark. The hotkey for moving to the previous bookmark location is **ALT** + **Shift** + **F12**.



17.2.3.4 Delete Bookmark

Follow these steps to delete bookmarks in C language.

To delete bookmarks in C language

1. Select the line position and delete the bookmark.
2. Right-click to open the context menu.

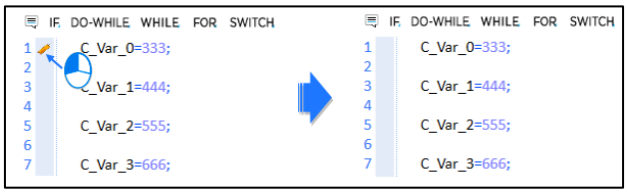
3. Click **Bookmark** > **Toggle Bookmark** to delete the bookmark.

   Or

   Click **Edit** > **Bookmark** > **Toggle Bookmark** in the menu bar.

   Or

   Click **Toggle Bookmark**  on the toolbar of POU dialog to delete the bookmark. The hotkey for deleting the bookmark is **CTRL** + **F12**.



17.2.3.5 Clear All Bookmarks

Follow these steps to delete all the bookmarks in C language.

To delete all bookmarks in C language

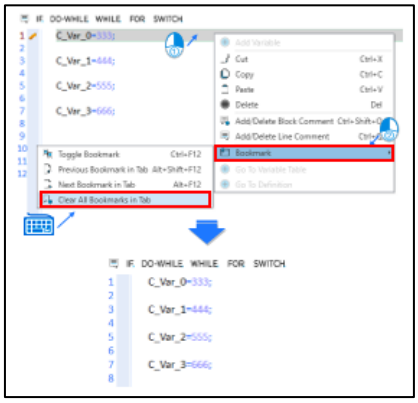
1. Right-click to open the context menu.
2. Click **Bookmark** > **Clear All Bookmarks in Tab** to delete all the bookmarks.

   Or

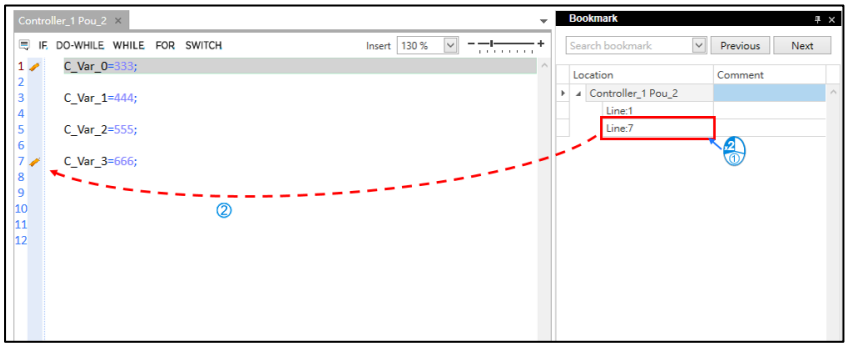
   Click **Edit** > **Bookmark** > **Clear All Bookmarks in Tab** in the menu bar.

   Or

   Click **Clear All Bookmarks in Tab**  on the toolbar of POU dialog to delete all the bookmarks.

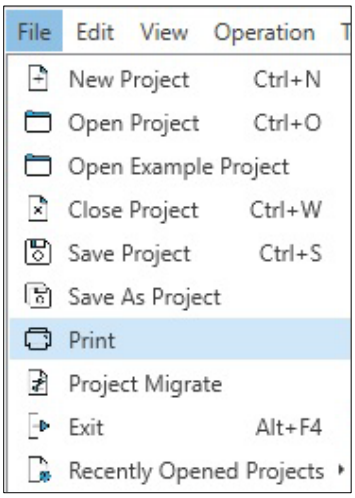


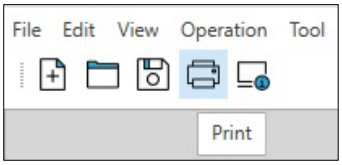
Note: In the Bookmark Manager dialog, double-click any bookmark location, and it will immediately move to the bookmark location.



17.3 Print

Go to the function menu **File** > **Print**, or click  on the standard toolbar to execute the print function of the current page.

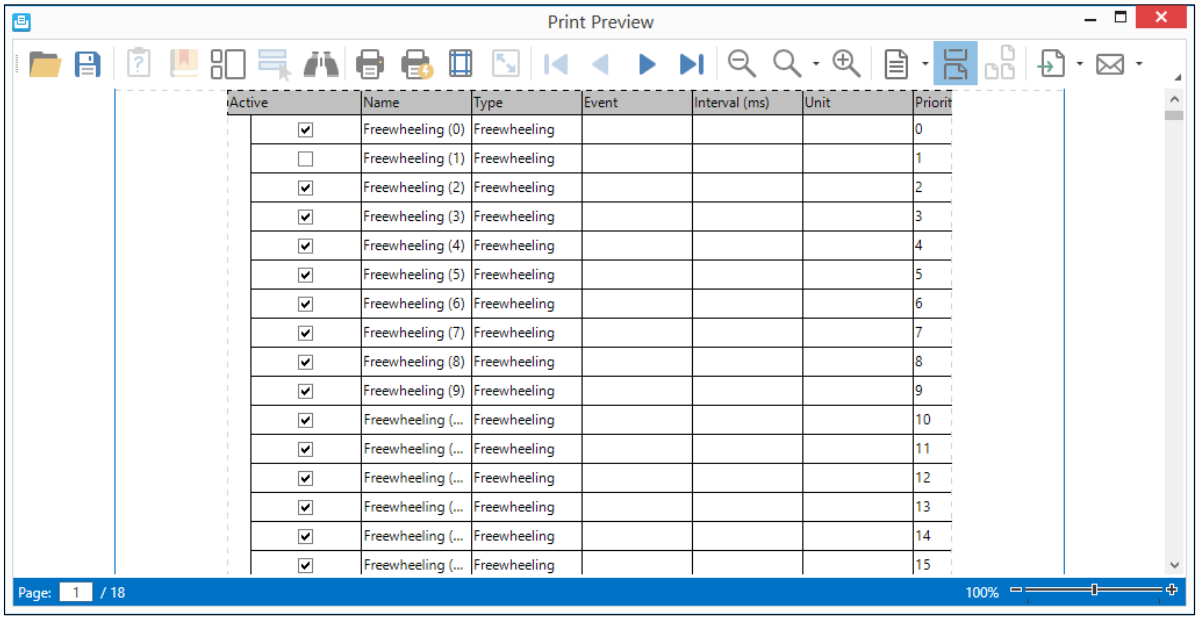




Note: When the execution page does not support print function, the print item will be grayed out and cannot be clicked.

17.3.1 Print Form Page

When the page used is the global / regional variable table, job management, positioning planning table, register notes, register usage status, etc., click **Print** and the following preview print window appears.



The following figure shows an icon toolbar in the **Print Preview** dialog to edit the print screen.



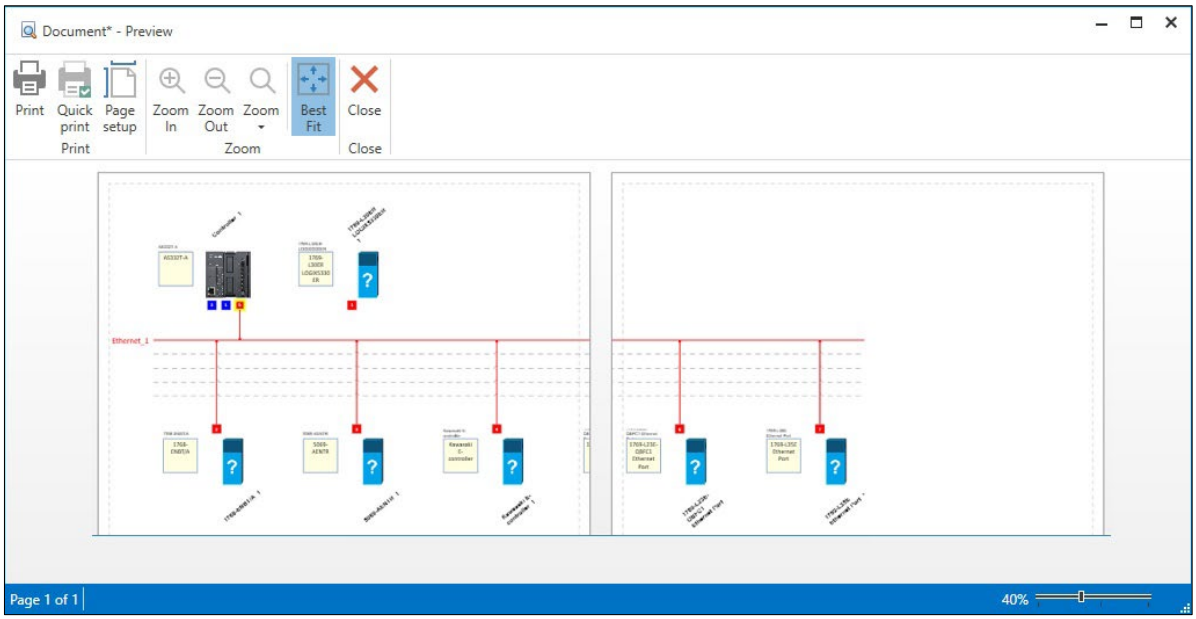
The following table describes the toolbar functions.

Icon	Description
	Open print file.
	Save this printed page as a file.
	Show file thumbnails.
	Search the content of the printed document.
	Confirm page setting and print.
	Use the system default printer and the default layout to print directly.

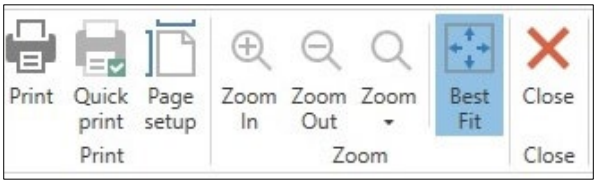
Icon	Description
	Document layout adjustment setting.
	The screen jumps to the first page of the document.
	The screen jumps to the previous page of the document.
	The screen jumps to the next page of the document.
	The screen jumps to the last page of the document.
	Reduce the display screen.
	Select the display aspect ratio.
	Enlarge the display screen.
	Choose how many pages to display at the same time.
	Set whether to display different pages at the same time when scrolling the screen.
	Export files in set format.
	Use E-mail to send the formatted file.
	Insert watermark.

17.3.2 Print Graphics Page

When the page used is network view, hardware configuration, custom data type, LD program, ST program, SFC program, C POU, etc., click **Print** to open **Print Preview** dialog.



The following figure shows an icon toolbar in the **Print Preview** dialog to edit the print screen.



The following table describes the toolbar functions.

Function	Description
Print	Specify page setting and print.
Quick print 列印	Use the system default printer and the default layout to print directly.
Page setup	Adjust the file page setting.
Zoom In	Enlarge the display screen.
Zoom Out	Reduce the display screen.

Function	Description
	Select the display aspect ratio.
	Click to automatically zoom to display the full page.
	Close the screen and choose whether to save the setting.

17.4 Download the Project with Delta HMI via USB or SD Card

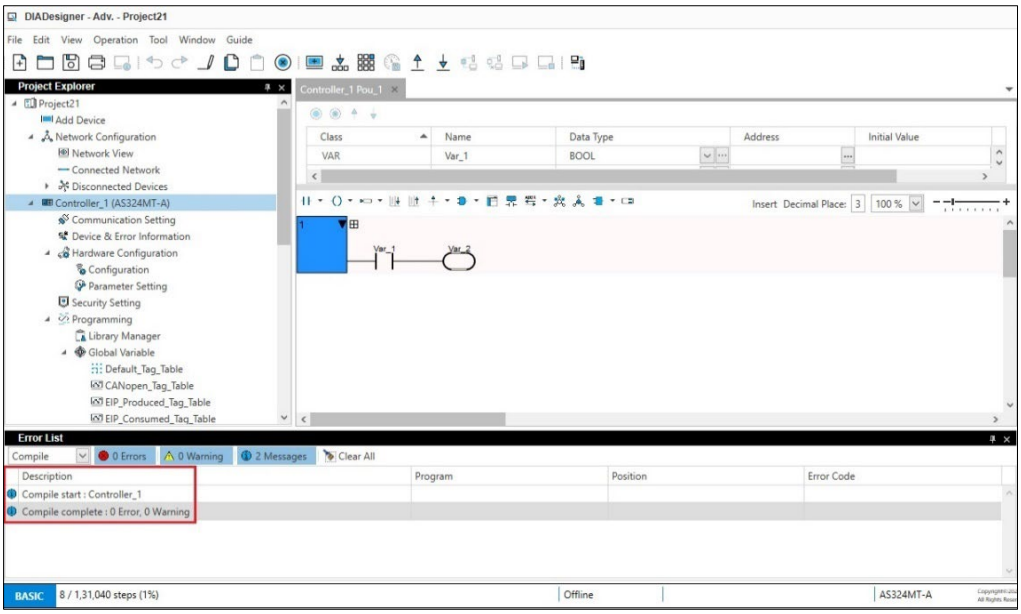
Note: This function requires a specific firmware version. For more information, contact the relevant contact window of DIAStudio.

In addition to downloading the project through the software interface, save the project file in a USB or SD card, and use the HMI to download the project file.

The file format supported by the USB or SD card is "device name.diaobj", and the generation method is as follows.

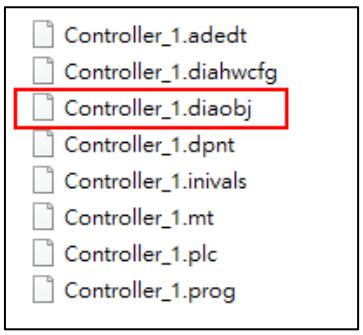
1. Compile Project

   Compile the project before archiving and confirm that the compile is error-free.



2. Project Archive

After compiling, do the project archive action to generate a file with the extension name diaobj in the corresponding device folder under the project archive path (currently only supports AS100, AS200, and AS300 series controllers), and place the file on the USB or SD card. Use DIAScreen with HMI to do project download actions. Refer to *DIAScreen User Manual* for the detailed process of downloading the project via USB or SD card via Delta HMI.



17.5 Option

To open Option dialog

- Click **Tool** > **Option** on the toolbar.

The **Option** dialog has the following setting pages.

- Project Setting

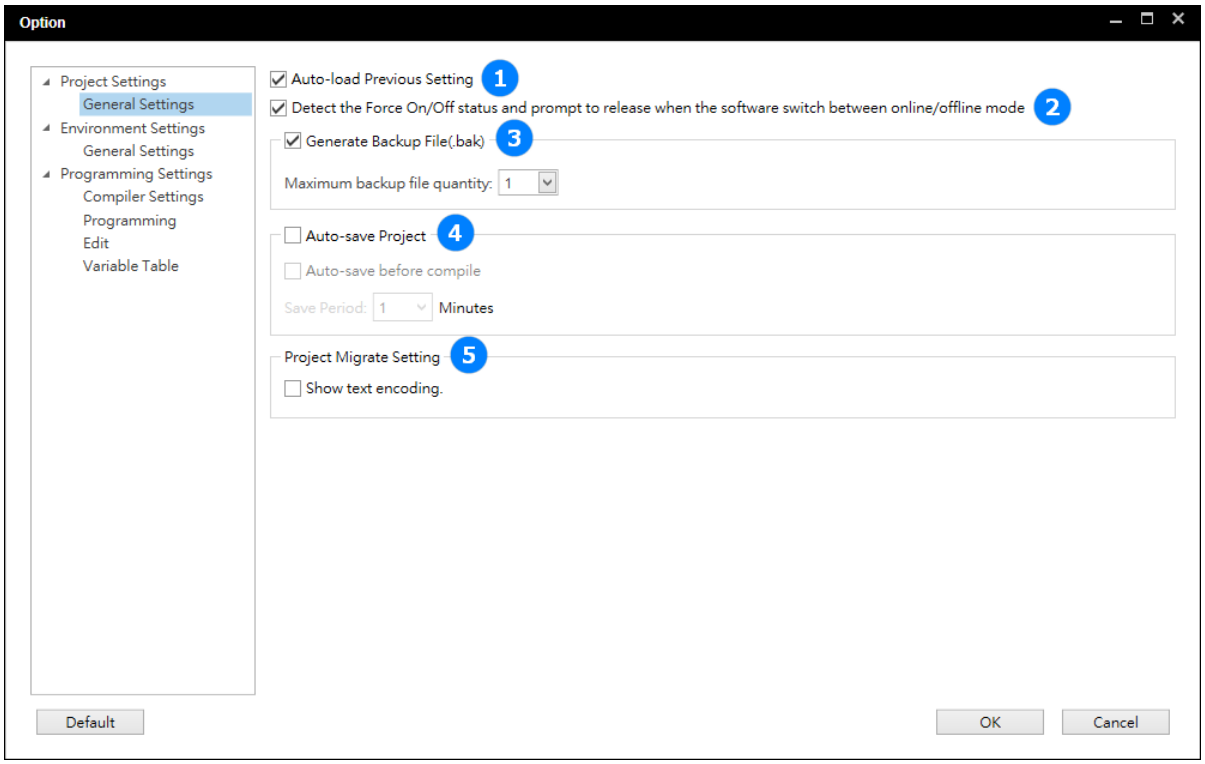
- Environment Setting
- Programming Setting

After making any changes,

- Click **OK** to save the changes.
- Click **Close** to discard the changes.
- Click **Default** to restore the default value.

17.5.1 Project Setting

The following table lists the settings with their description in **Project Settings** page.



1 Auto-load Previous Setting

Function	Description
Auto-load Previous Setting	If selected, a project which was edited last and which was not closed is opened automatically whenever the software is started.

② Detect the Force On/Off status and prompt it to release when the software switch between online/offline mode

Function	Description
Detect the Force On / Off status and prompt to release when the software switch between online / offline mode	If selected, when switching to the online or offline mode, if will force do display the On / Off status and a release prompt message pops up.

③ Generate Backup File (.bak)

Function	Description
Generate Backup File (.bak)	If selected, a backup file is created when saving a project. You can restore it when the disk drive is damaged.
Maximum backup file quantity	Set the maximum backup file quantity. The value range is 1–10, and the default value is 1.

④ Auto-save Project

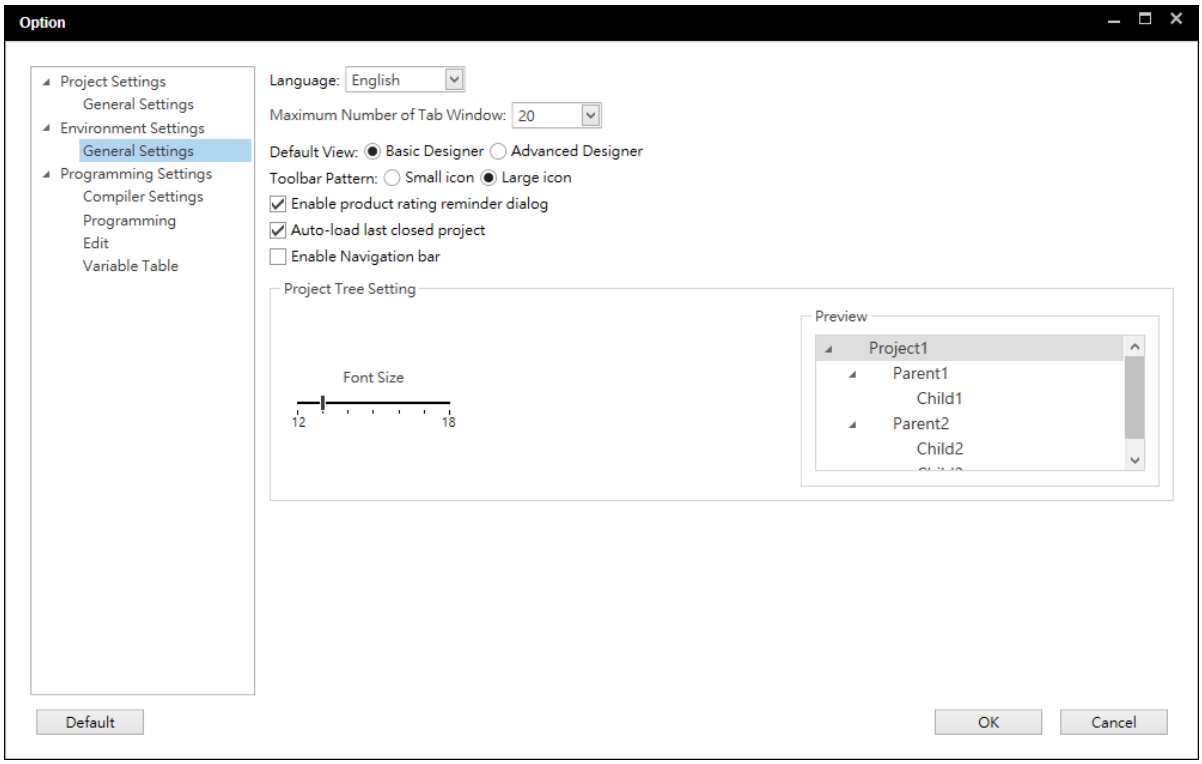
Function	Description
Auto-save Project	If selected, the project is saved automatically as per the set **Save Period**.
Auto-save before compile	If selected, the project is automatically saved before compiling.
Save Period	Set an interval time for the system to automatically save the project. The default value is 1.

⑤ Project Migrate Setting

Function	Description
Show text encoding	If selected, the text encoding is displayed when migrating a project.

17.5.2 Environment Settings

The following table lists the settings with their description in the **Environment Settings** page.

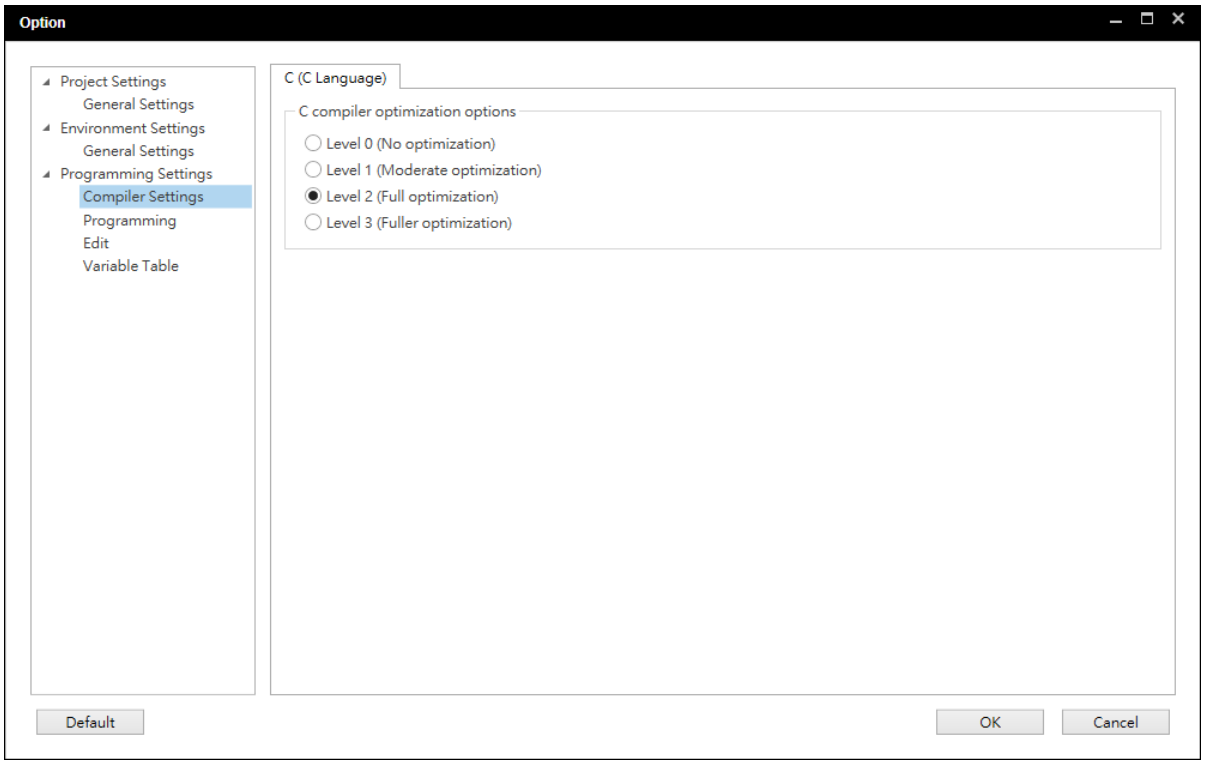


Function	Description
Language	Select the interface language of the software. The default is the operating system language.
Maximum Number of Tab Window	Select the maximum numbers of tabs to display simultaneously. The value range is 2–20, and the default value is **20**.
Default View	Choose the default starting interface. The default is **Basic Designer**.
Toolbar Pattern	Choose the display size of the toolbar icon. The default is **Small Icon**.
Enable product rating reminder dialog	Set whether to display the product rating reminder dialog when closing the software. The default is **enabled**.
Auto-load last closed project	Set whether to automatically load the last closed project when opening the software. The default is **enabled**.
Enable Navigation bar	Set whether to display the BASIC navigation bar when switching to the ADV page.

Function	Description
	The default is not enabled.
Project Tree Setting	Set the font size in **Project Explorer**. The value range is 12–18, and the default value is **13**.
Preview	Preview the effect of the set font size.

17.5.3 Programming Settings

Set the font, color, and so on through the Programming Settings function. The following table lists the settings with their description in the **Compiler Settings** page.



Function	Description
C compiler optimization options	Choose to spend more compile time optimizing the compiled code.

The following table lists the functions with their description in the **Programming** page.

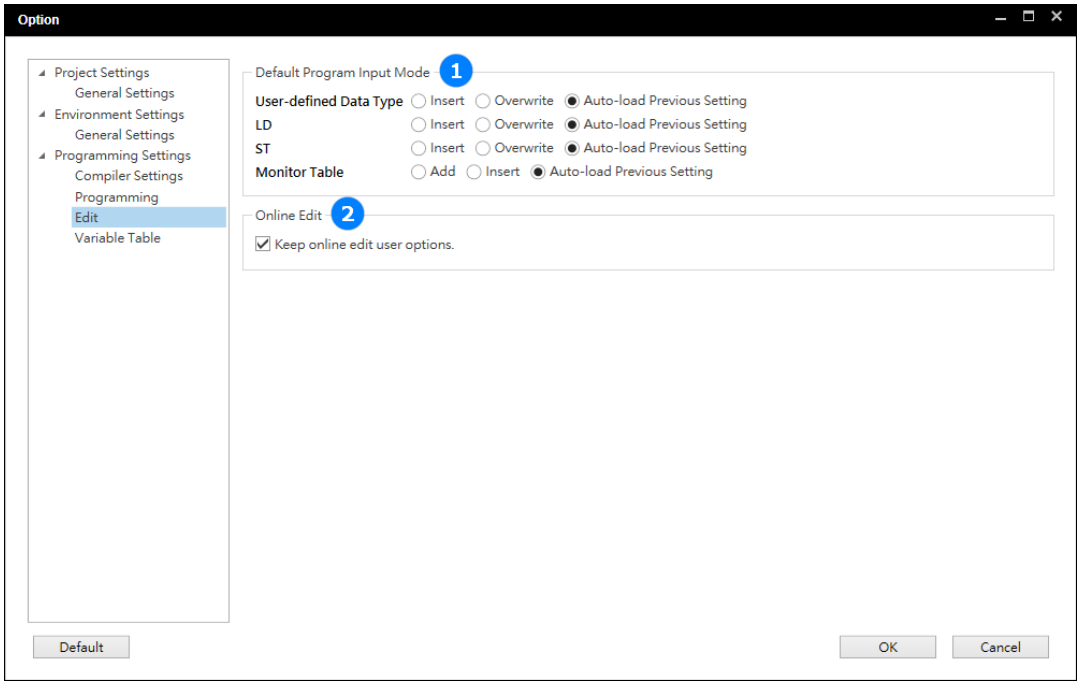
LD (Ladder Diagram)

Function	Description
Color Setting	Set the color of text, background, and other content in the LD diagram.
Default	Click to restore to the default color.
Preview (Offline)	Preview the effect of the offline mode color setting.
Preview (Online)	Preview the effect of the online mode color setting.
Operand Fields	Set the width of the operand field.
Block Name	Set the width of the block.
Preview	Preview the effect of the operand and block width setting.

Text Editor

Function	Description
Preview (Offline)	Preview the effect of the setting for the text editor.
Set	Perform the settings of the text editor, including all text, background color, and so on in the DUT declaration dialog, ST and C POU

The following table lists the functions with their description in the **Edit** page.



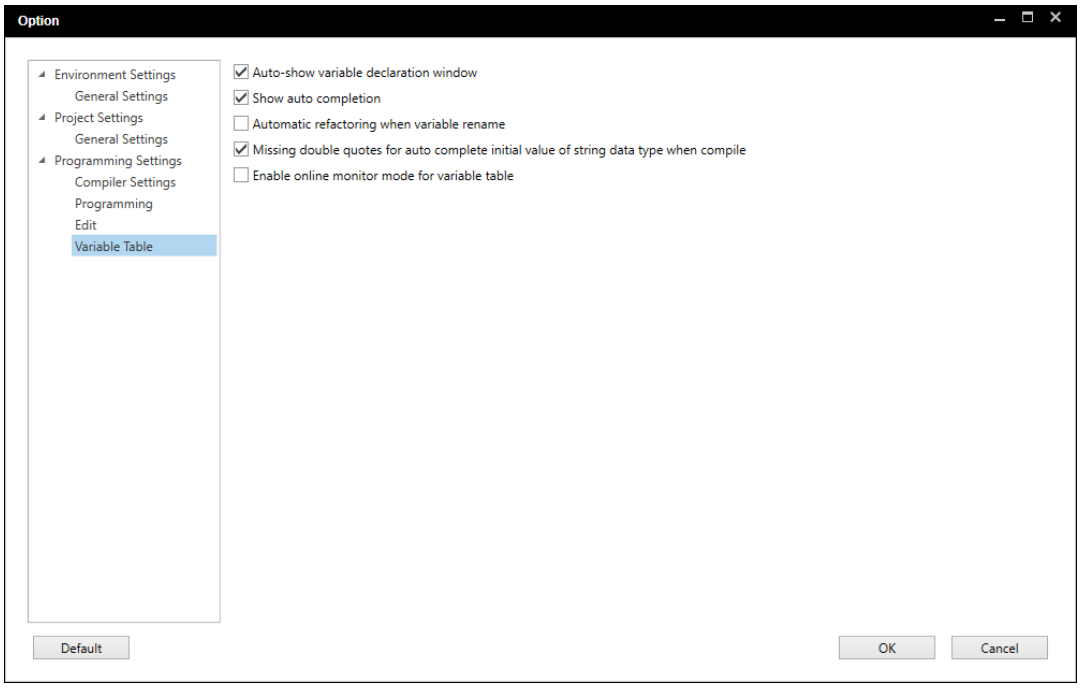
1 Default Program Input Mode

Function	Description
User-defined Data Type	Select the edit configuration for keyboard input.
LD	
ST	
Monitor Table	

② Online Edit

Function	Description
Keep online edit user options	If selected, you can choose actuating code or actuating code & program when editing in online mode.

The following table lists the settings with their description in the **Variable Table** page.



Function	Description
Auto-show variable declaration window	If selected, the **Add Variable to POU** dialog appears whenever a variable is created.
Show auto-completion	If selected, when inputting the variables in the program, the system will list all the set items for the variable, and you can use the mouse or keyboard to select from the list.

Function	Description
Automatic refactoring when variable rename	If selected, when the variable is changed, the variable in the program will be replaced automatically.
Missing double quotes for auto complete initial value of string data type when compile	If selected, double quotes will be automatically added to strings that do not have double quotes in the initial value field during compilation.
Enable online monitor mode for variable table	If selected, the variable table can read and write values and status.

17.6 Password Management

17.6.1 DIADesigner Project

For projects developed by you, DIADesigner also provides multiple protection mechanisms. The purpose is to ensure that the programs developed by you can be fully protected. The following is the password protection structure provided by DIADesigner, in which each password can be set, and you can adopt different data protection mechanisms according to their needs.

Password Type		Description	Model Character Limit	
			AS	DVP
DIADesigner Project	Program ID	As the authentication information for the first check, and when downloading the project, the "Program ID" set in the DIADesigner project and the "PLC ID" set in the host must be the same before the operation can be performed, so that it can be used to restrict specific projects from being downloaded to specific hosts only.	4~16 character	4~8 character
	POU Password	It protects the password function of the POU program, and each POU can independently set an	4~16 character	4~16 character

Password Type		Description	Model Character Limit	
			AS	DVP
		exclusive password, which can be used by you to protect specific project.		
	Design Tool Password	To protect various design data, different models also provide password protection for supported design tools, including E-CAM password and positioning planning table password, etc., which will be introduced in the chapters of each function. **Note**: Refer to Password for the password of the Position Plan Table.	4~16 character	-
PLC	PLC ID	As the authentication information for the first check, and when downloading the project, the "Program ID" set in the DIADesigner project and the "PLC ID" set in the host must be the same before the operation can be performed, so that it can be used to restrict specific projects from being downloaded to specific hosts only.	4~16 character	4~8 character
	PLC Password	The password function is used to protect the PLC internal data. You must input the correct password before uploading and downloading the data protected by "PLC password".	4~16 character	4 character

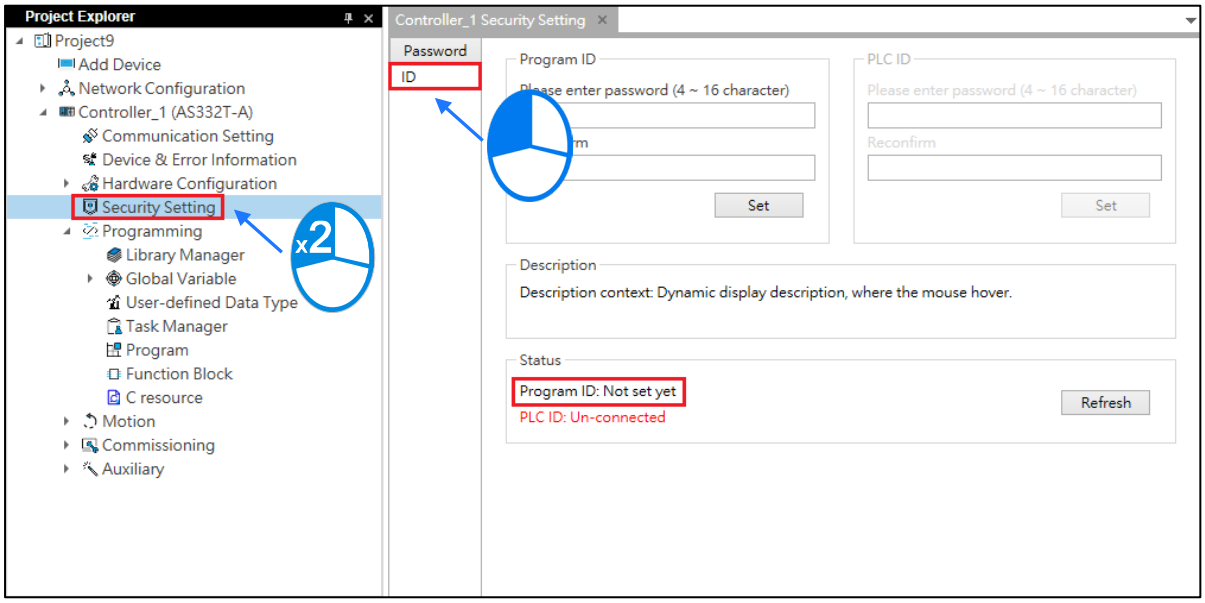
17.6.2 Program ID and PLC ID

The ID is the authentication information used as the first check. When downloading, it is set in the DIADesigner project. **Program ID** and **PLC ID** set in the host must be the same for operation. Therefore, the protection mechanism of the ID can be used

to restrict the project with **Program ID** set to be downloaded to a specific host. And the host must set the same **PLC ID**. When uploading the project, if the host has set **PLC ID**, the system requires you to input the correct ID for verification before uploading the data; and when opening a project with a set **Program ID**, the system also requires the correct ID to be entered before allowing it to open, so the ID can be used as your authorization to access data.

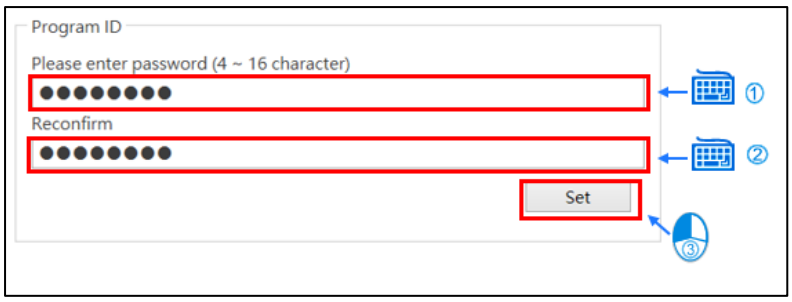
17.6.2.1 Program ID Setting

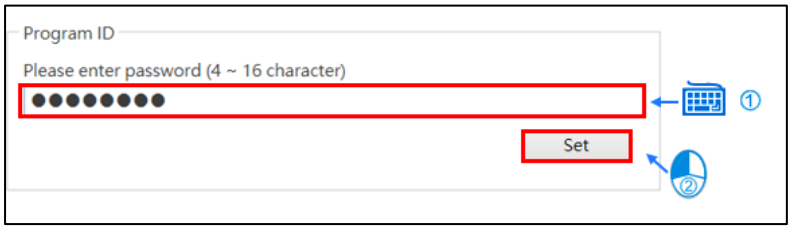
Select the device you want to confirm in the **Project Explorer**. Double-click the **Security Setting**, and select **ID** tab. The ID setting dialog appears, and from the bottom of the window, you can know whether for the currently opened project the password is set or not.



If the password for the current project is not set, there will be two fields for setting the password in the dialog. To set the password, input the same password in the two fields, and click **Set**. If the function of program ID is already set, there will only be a field for unlocking the password in the dialog. To unlock the password, directly input the correct password in this field, and click **Set**.

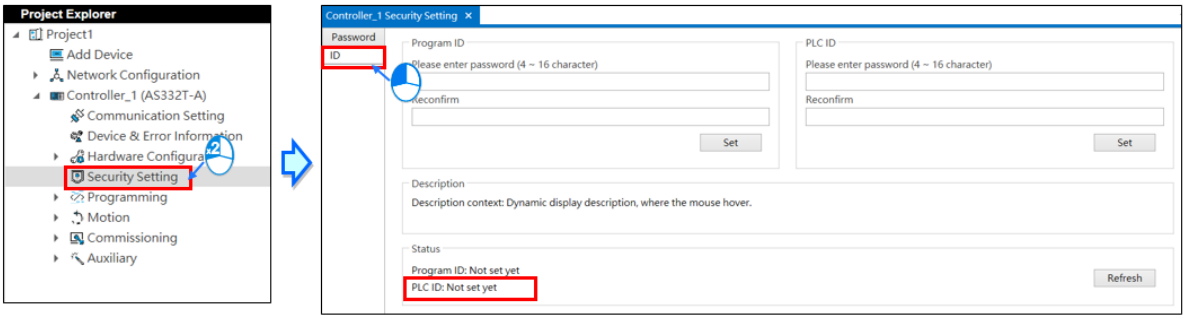
The password can only contain English letters, numbers and blanks. English letters must be case-sensitive; the character limit of the password may be different for different models. Follow the prompts in the field to set password. When the setting is completed, the state of the dialog automatically switches.



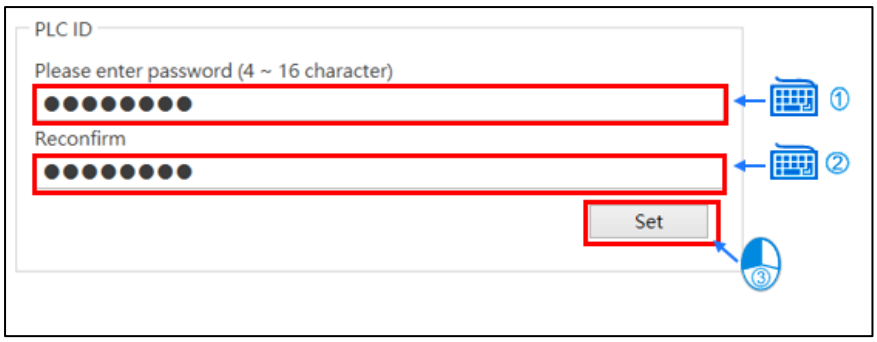


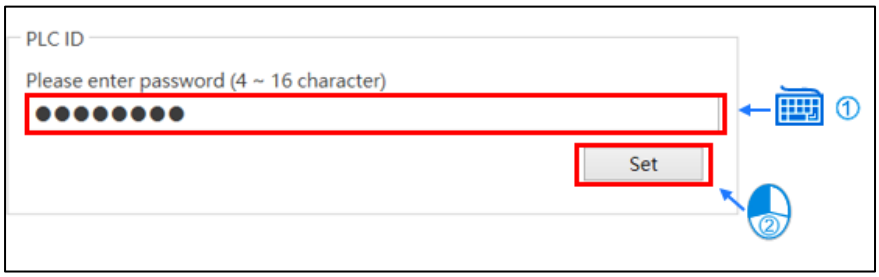
17.6.2.2 PLC ID Setting

Before setting PLC ID, make sure that DIADesigner is connected to the host normally. Select the device you want to confirm in **Project Explorer**. Double-click **Security Setting**, and select **ID** tab. The ID setting dialog appears. At the bottom of the window, you can know whether the current host has set PLC ID.



If the current PLC ID of the project is not set, there will be two fields for setting the ID in the dialog. To set the ID, input the same ID in the two fields, and click **Set**. If the function of PLC ID code is set, there will only be a field for unlocking the ID in the dialog. To unlock the ID, directly input the correct ID in this field, and click **Set**. The ID can only contain English letters, numbers and blanks. English letters must be case-sensitive; the character limit of the ID for different models may be different. Follow the prompts in the field to set ID. When the setting is completed, the state of the dialog automatically switches.



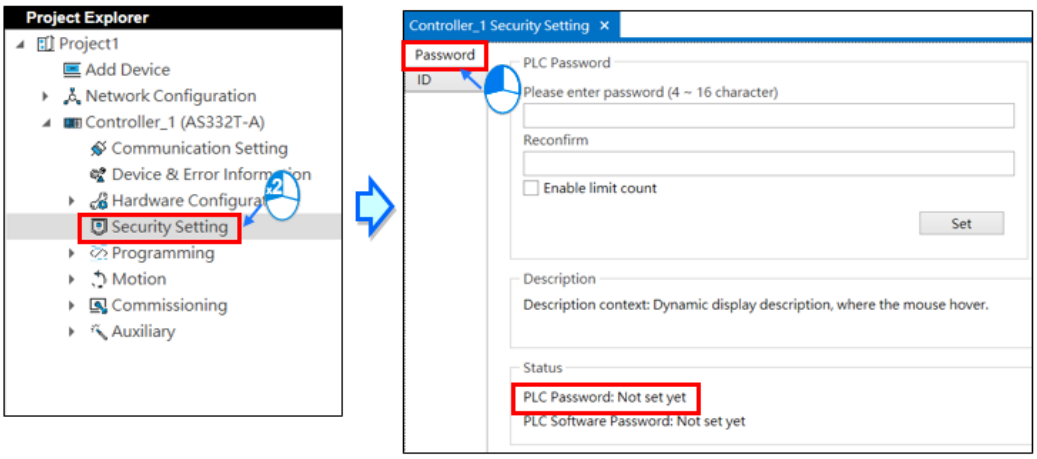


17.6.3 POU Password

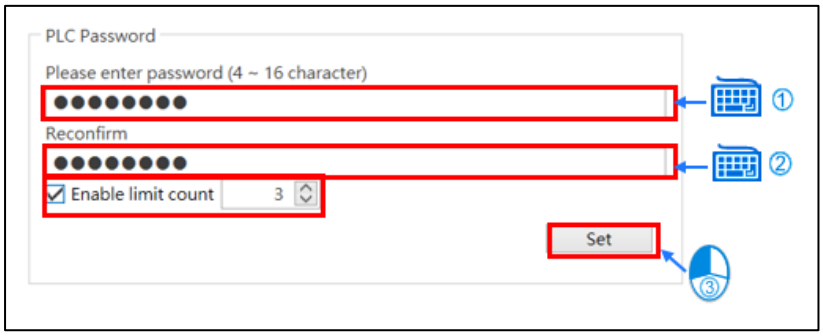
DIADesigner allows you to set a unique set of passwords for different programs or function block POUs. When you want to open a POU protected by a password, the system asks for the correct password before it can be opened. This function can be used to add an additional protection mechanism to the project. Refer to Managing a Program Organization Unit (POU) for the method of setting and unlocking the POU password.

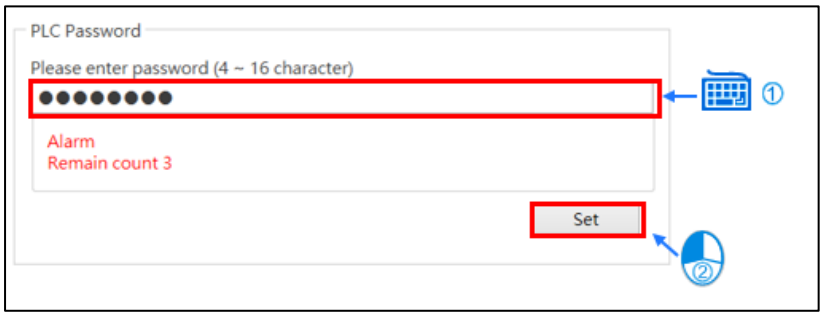
17.6.4 PLC Password

Before setting the PLC Password, confirm that DIADesigner is connected to the host normally. Click the device to be confirmed in **Project Explorer**. Double-click **Security Setting** and select **Password** to open password setting dialog.



If the current PLC password of the project is not set, there will be two fields for setting the password in the dialog. To set the password, input the same password in the two fields, and click **Set**. To set the limit count, check **Enable limit count** first and set the number in the field and click **Set**. If the function of the PLC password is currently set, there will only be a field for unlocking the password in the dialog. To unlock the password, directly input the correct password in this field and then click **Set**. The set password can only contain English letters, numbers and blanks. English letters must be case-sensitive; the password limit may be different for different models. Follow the prompts in the field to set password. When the setting is completed, the state of the dialog automatically switches.



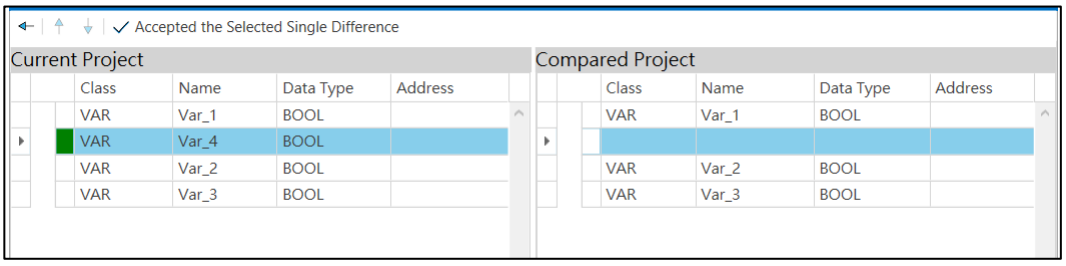


17.7 Comparison Function

17.7.1 Comparison Function Rules

DIADesigner provides a comparison tool for users to compare two DIADesigner projects on the computer or compare devices in the project. The rules for comparison are as follows.

- Comparable content includes global variable tables, local variable tables, DUTs, LD POU, ST POU, C language POU, and C resource files.
- Even if the variable symbol tables have the same contents, they will be judged as different if the order of variable symbols is different.
- Variables or programs will compare skipped lines of content, as shown below. The comparison result will determine that there is a difference in the second variable, and the others are the same.



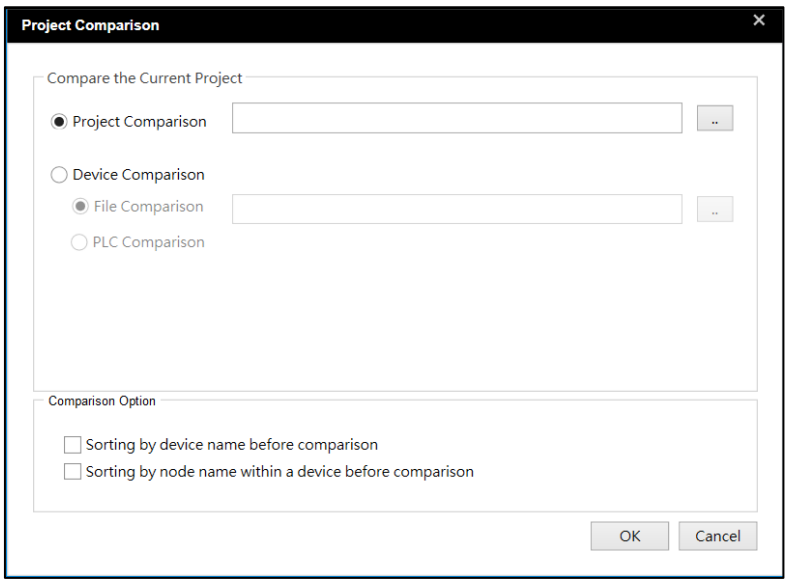
- If no device is added to the current project or the target comparison project, comparison is not allowed.
- There may be multiple devices in the project. If the devices in the current project and the compared project have the same name but different models, they cannot be compared.
- If the project or content is password protected, you will need to enter the password during the comparison process before continuing to comparison.
- For read-only nodes, such as CANopen variable tables, differences will be displayed, but changes are not allowed.
- While the comparison function is in progress, any other editing functions and online functions cannot be used.
- In **Comparison Differences** dialog, you can only operate the toolbar buttons to change the current project, and you cannot directly edit the content.

17.7.2 Project Comparison

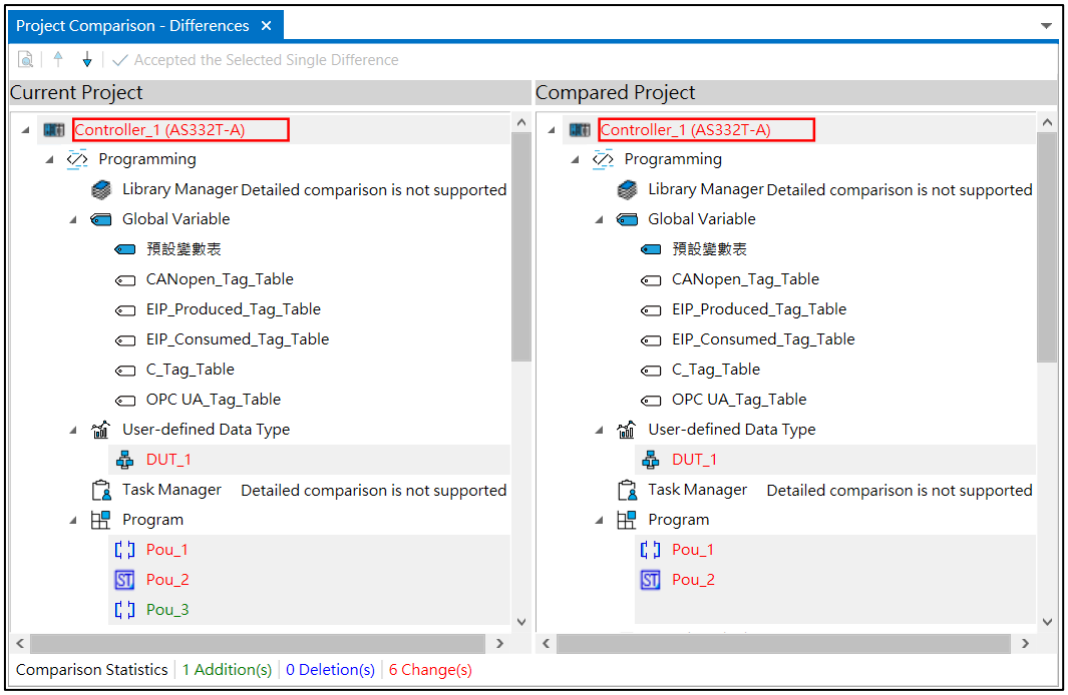
When you want to compare the currently open project with other saved projects, follow these steps to compare.

To compare projects

1. Click **Tool** > **Comparison** in the menu of the current project.
2. Select **Project Comparison** in **Project Comparison** dialog, and then click to select the path of the target project. Check the comparison options if necessary to decide the sorting method for comparison and click **OK** to start project comparison.

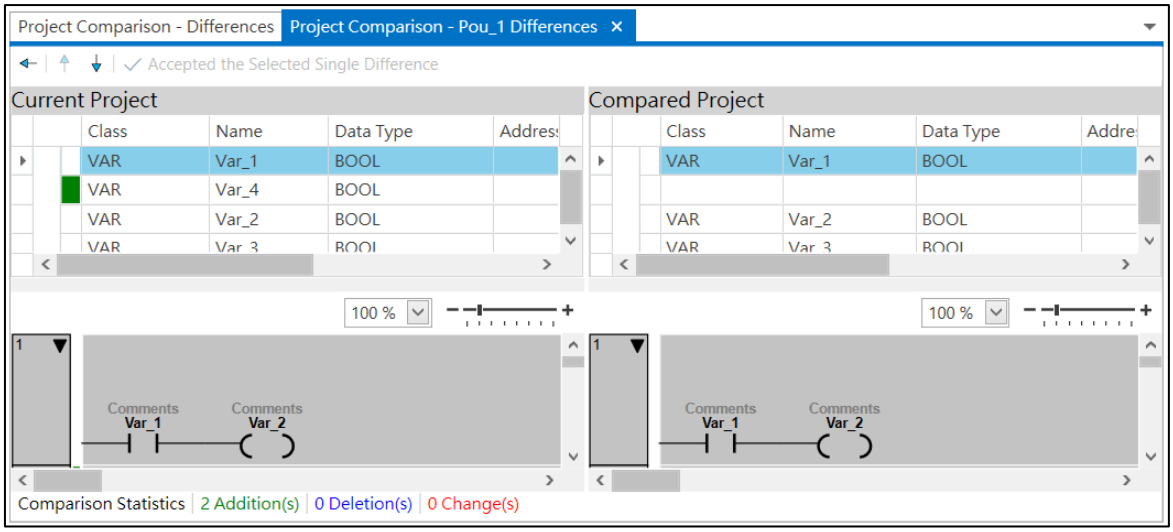


3. Open the **Compare Differences** dialog. The left side is the current project, and the right side is the compared project. The difference between the two is represented by color. Red is the node with differences between the two projects, green is the node added to the current project, and blue is the node deleted from the current project.

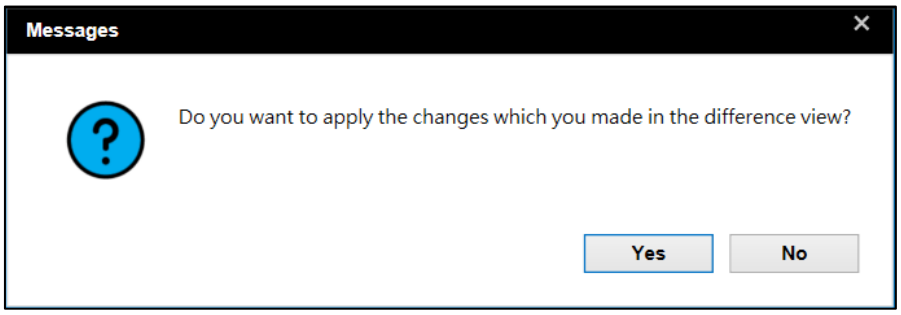


4. Click **Next** and **Previous** buttons on the toolbar to jump back and forth between nodes with differences. After selecting a node, click **Accepted the Selected Single Difference** to copy the contents of the compared project and paste into the current project.

5. If there is a node in the comparison dialog that you want to confirm in detail, double-click the node or select the node first and then click **Show the Differences Between the Selected Pair of Objects** button on the toolbar to enter the detailed comparison dialog. The detailed comparison dialog scrolls up and down between the current project and the compared project to confirm the differences between the two. Click **Next** and **Previous** buttons on the toolbar to jump back and forth between rows with differences. Select a row and click **Accepted the Selected Single Difference** button to copy the contents of the compared project and paste into the current project. Click **Accepted the Selected Single Difference** button again to restore the contents before the change.



6. After completing the detailed comparison, click **Show the Project Differences** button on the toolbar to return to the project tree comparison dialog. After completing all comparisons and changes, close the comparison dialog and select whether to save these changes in the confirmation dialog.

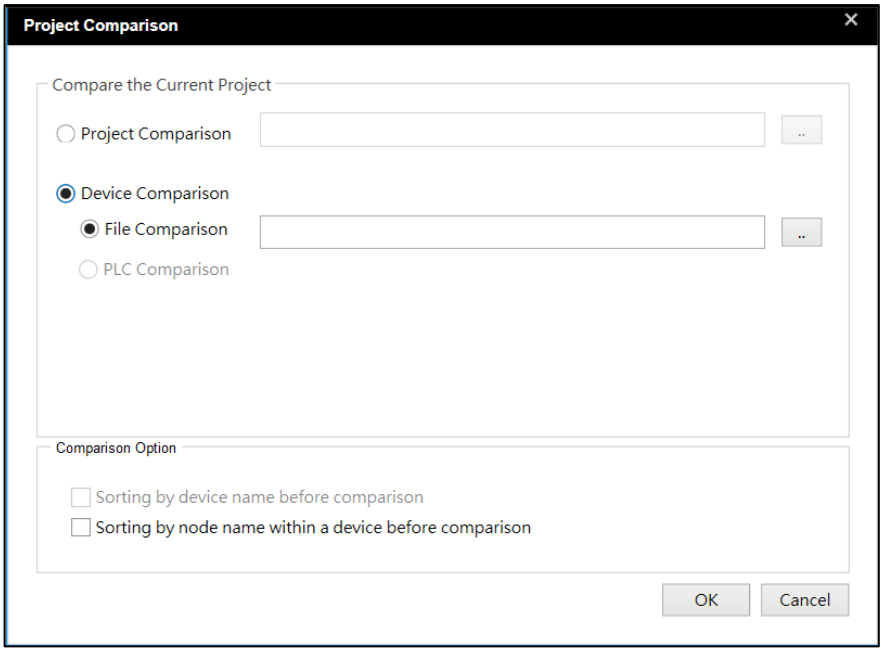


17.7.3 Device Comparison

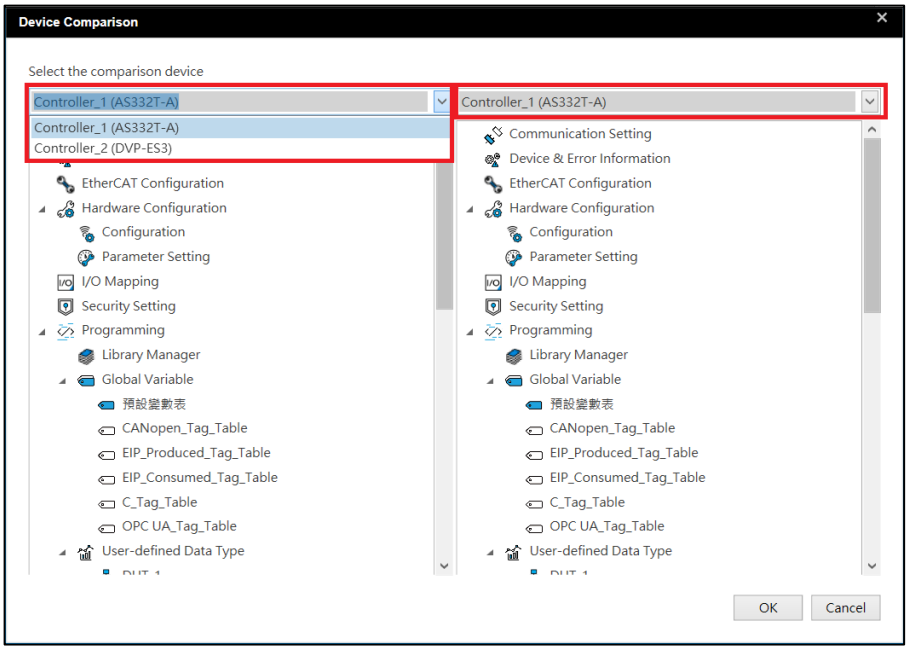
When you want to compare a single device in the currently open project to a single device in another saved project, follow these steps to compare.

To compare devices in open project to a device in another saved project

1. Click **Tool** > **Comparison** in the menu of the current project.
2. Select **Device Comparison** in **Project Comparison** dialog. Click to select the path of the target project, check the comparison options if necessary to decide the sorting method for comparison, and click **OK** to start project comparison.



3. Then specify a single device in the current project to compare with a single device in the compared project and select the device in each project from the drop-down menus on both sides. Devices of different models will not be able to compare each other. Press **OK** to start device comparison.



4. Then open the **Comparison Differences** dialog. For subsequent operations, please refer to Project Comparison and start with step 3.

Chapter 18: Troubleshooting

18.1 Error Log

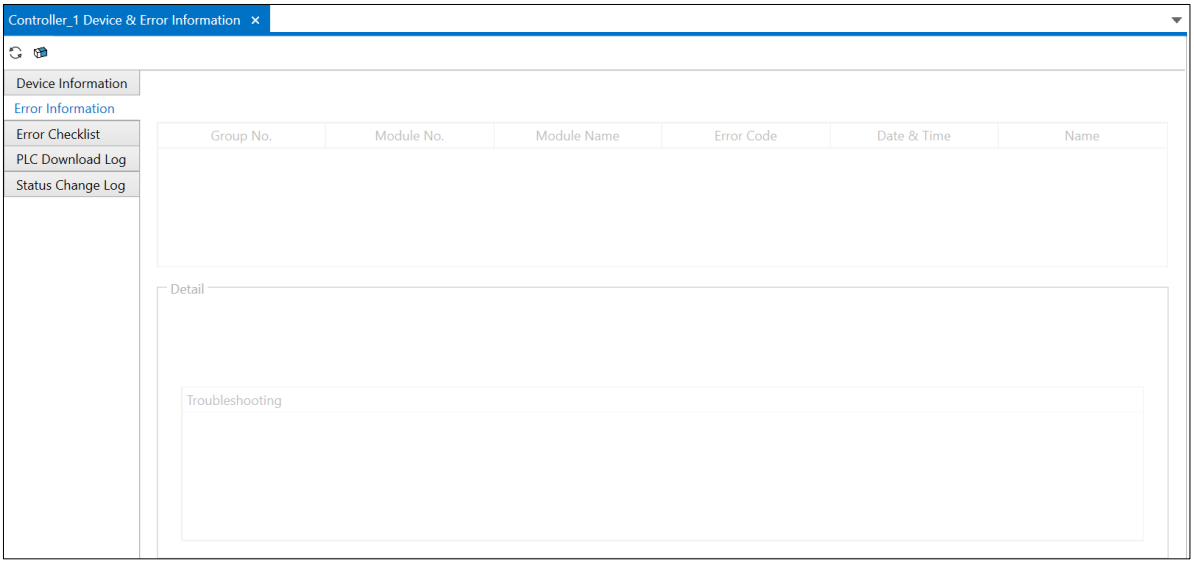
Under normal connection with the hosts, select **Controller** > **Device & Error Information** to view logs regarding the modifications and errors.

There are three type of logs.

- Error Information
- PLC Download Log
- Status Change Log

18.1.1 Error Information

The **Error Information** tab displays the errors in devices along with the date and time information. **Error Information** tab for different device series may be different. The following figure shows **Error Information** tab for PLC.



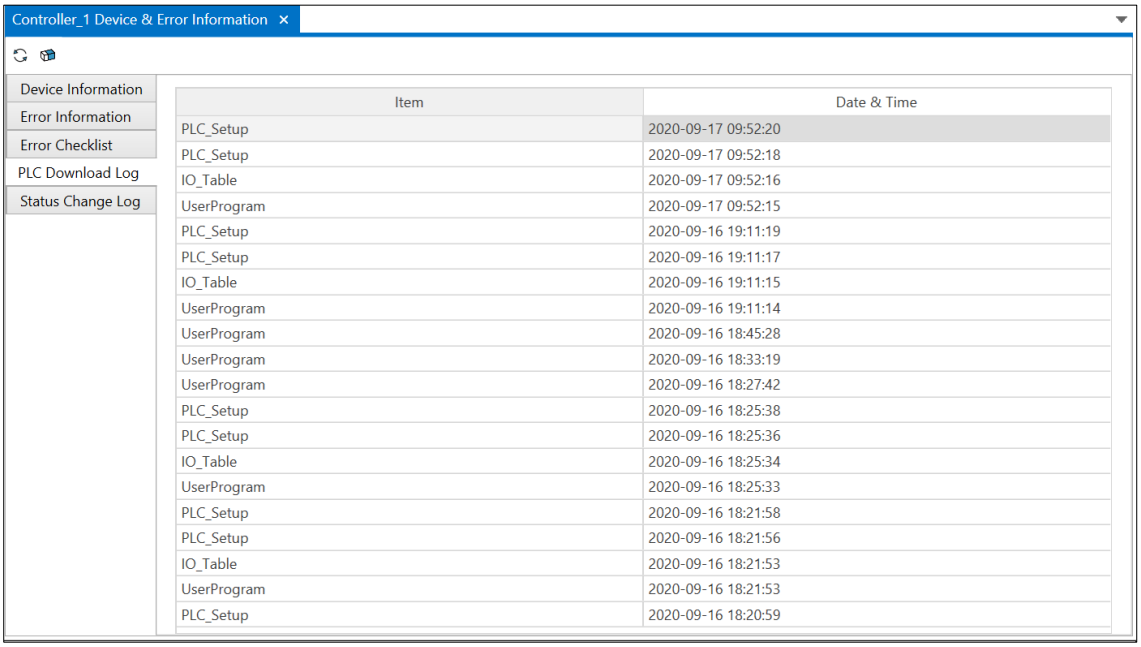
The following table describes the functions in **Error Information** tab.

Function / Icon	Description
Group No.	Display the group number in which the error is active.
Module No.	Display the module number in which the error is active.
Module Name	Display the module name in which the error is active.
Error Code	Display the error code.

Function / Icon	Description
Date & Time	Display the date and time of the error code.
Name	Display the error message.
Detail	Display the details of error.
Troubleshooting	Display how to troubleshooting and fix the error.
	Click **Refresh** to let the system retrieve data from the host and update the data in the log.
	Click **Clear** to clear the table and internal host log as well as the host reset status. **Note**: If the error is still active, the entry remains in the error status, and new error log is created.

18.1.2 PLC Download Log

The **PLC Download Log** tab displays the history log related to Download / Upload of programs and parameters for PLC devices.



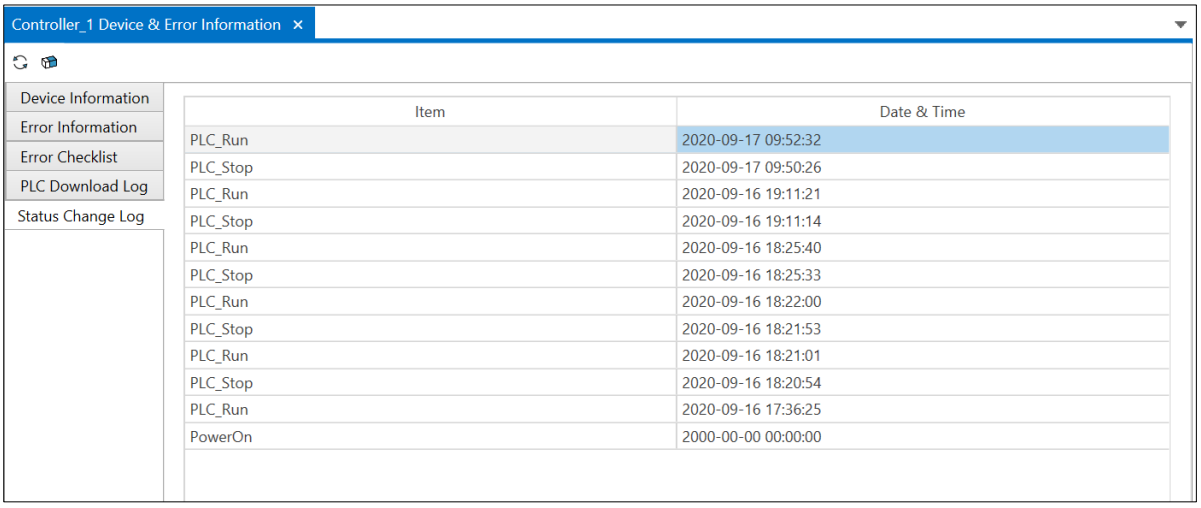
The functions in the **PLC Download Log** tab are described in the following table.

Function / Icon	Description
Item	Display the name of the item in Download or Upload action. **Example**: User Program, PLC Setup, IO Table etc.
Date & Time	Display the date and time of the upload or download item.

Function / Icon	Description
	Click **Refresh** to let the system retrieve data from the PLC and update the data in the log.
	Click **Clear** to clear the table and internal host log as well as the host reset status.

18.1.3 Status Change Log

The **Status Change Log** tab displays the log related to the change of PLC CPU status.

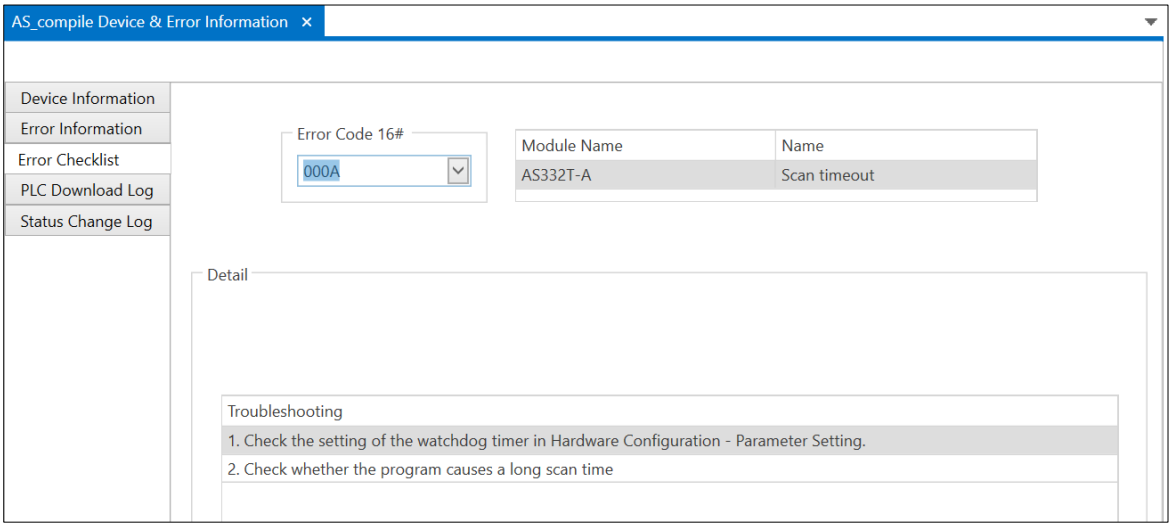


The functions in **Status Change Log** tab are described in the following table.

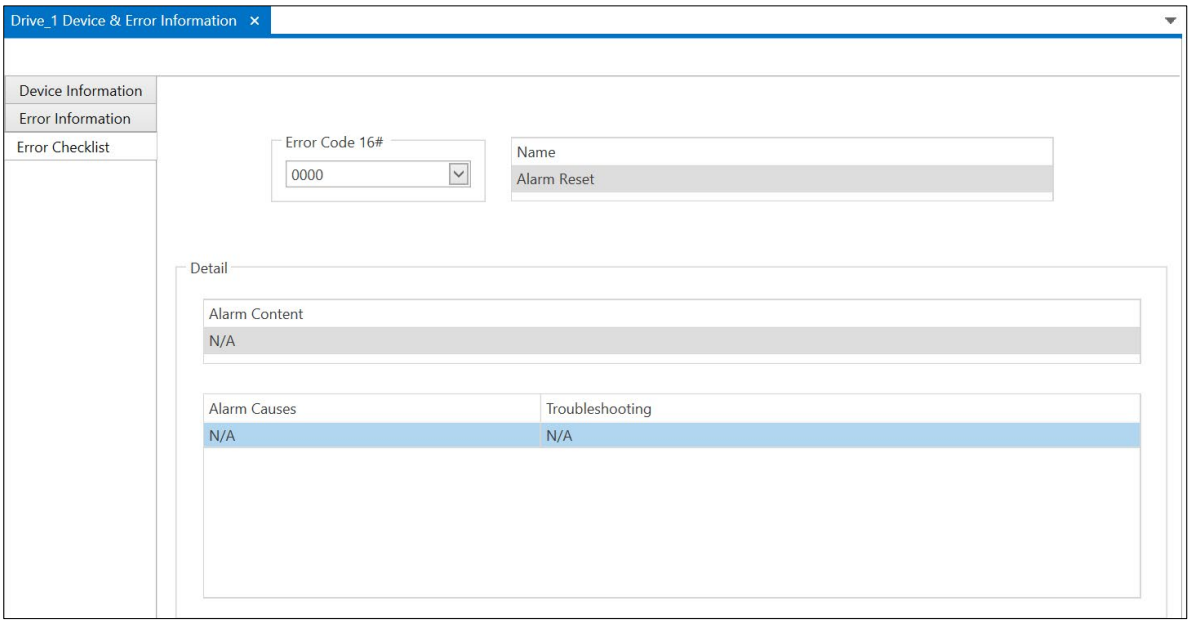
Function / Icon	Description
Item	Display the name of the item in PLC CPU status. **Example**: PLC RUN, PLC STOP etc.
Date & Time	Display the date and time of the PLC CPU status change.
	Click **Refresh** to let the system retrieve data from the PLC and update the data in the log.
	Click **Clear** to clear the table and internal host log as well as the host reset status.

18.2 Error Checklist

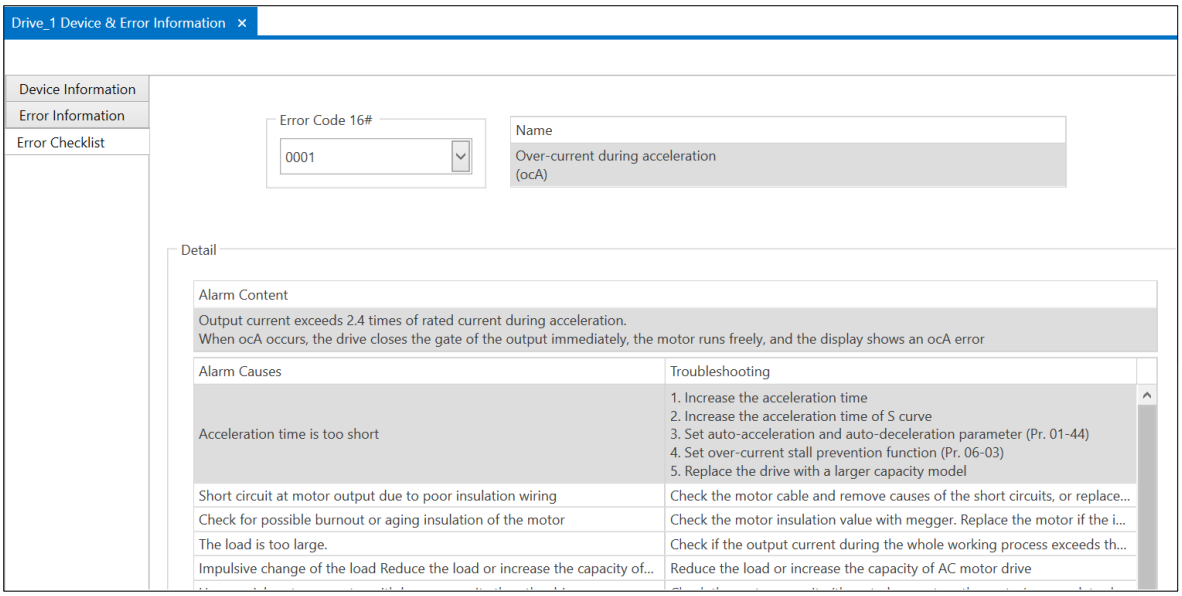
Check the description of the errors and error codes offline with the help of device > **Device & Error Information** > **Error Checklist**. **Error Checklist** tab displays all the error codes, names, description, alarm causes, fix methods, corrective actions etc.



The following figure shows **Error Checklist** tab of a servo.



The following figure shows **Error Checklist** tab of VFD.


Drive_1 Device & Error Information
Device Information
Error Information
Error Checklist
Error Code 16#
0001
Name
Over-current during acceleration
(ocA)
Detail
Alarm Content
Output current exceeds 2.4 times of rated current during acceleration.
When ocA occurs, the drive closes the gate of the output immediately, the motor runs freely, and the display shows an ocA error
Alarm Causes
Troubleshooting
Acceleration time is too short
1. Increase the acceleration time
2. Increase the acceleration time of S curve
3. Set auto-acceleration and auto-deceleration parameter (Pr. 01-44)
4. Set over-current stall prevention function (Pr. 06-03)
5. Replace the drive with a larger capacity model
Short circuit at motor output due to poor insulation wiring
Check the motor cable and remove causes of the short circuits, or replace...
Check for possible burnout or aging insulation of the motor
Check the motor insulation value with megger. Replace the motor if the i...
The load is too large.
Check if the output current during the whole working process exceeds th...
Impulsive change of the load Reduce the load or increase the capacity of...
Reduce the load or increase the capacity of AC motor drive

Chapter 19: Appendix

19.1 Installing USB Driver for AS Series CPU Module

19.1.1 Installing USB Driver in Windows 7

The installation of USB driver for an AS series PLC CPU module on Windows 7 operating system is introduced below. To install USB driver for an AS series PLC CPU module on another operating system, refer to the instructions in that operating system for more information about the installation of new hardware.

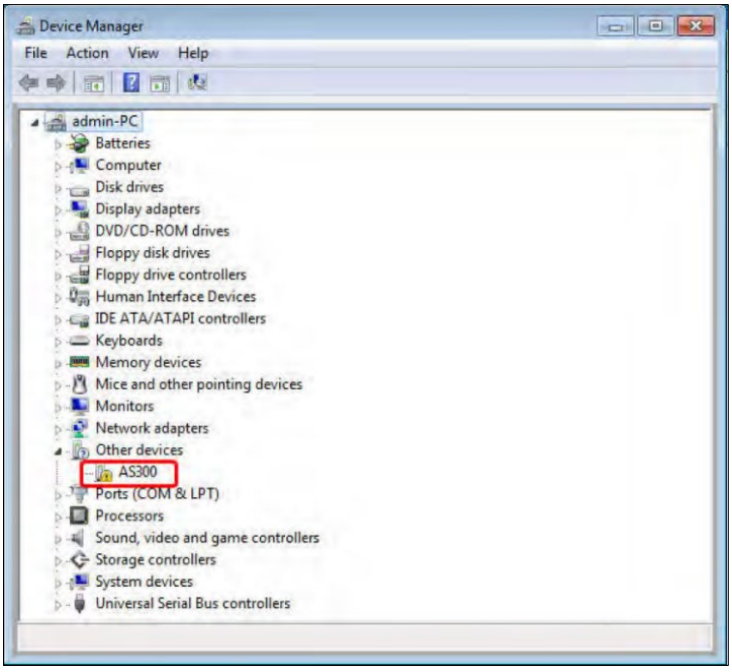
Follow these steps for the installation of USB driver in Windows 7 operating system.

To install USB driver

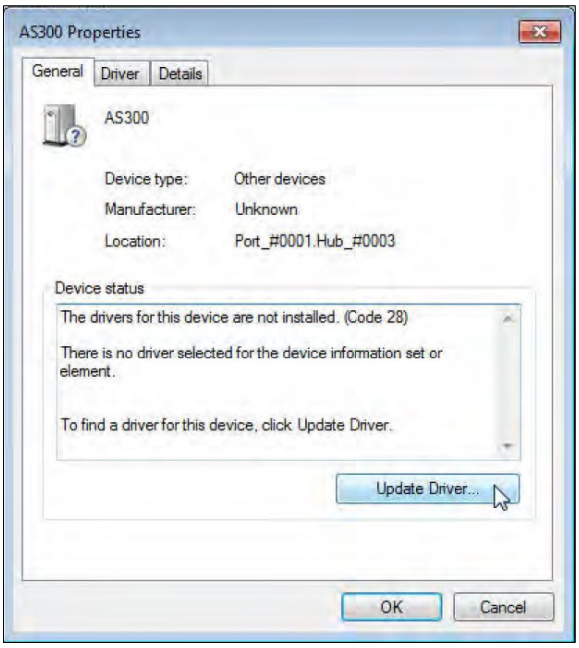
Pre-requisite

Make sure that the controller is powered up. Connect the controller to a USB port on the computer with a USB cable.

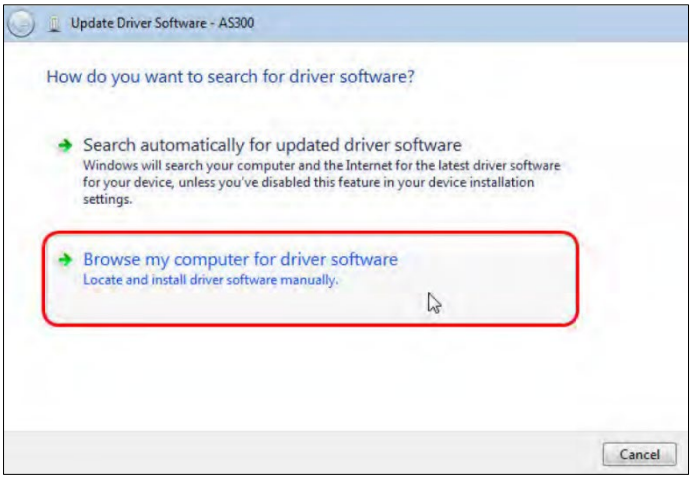
1. Open **Control Panel** > **Device Manager**.



2. Select and double-click **AS100/AS200/AS300** to open **AS100 Properties/AS200 Properties/AS300 Properties** dialog.

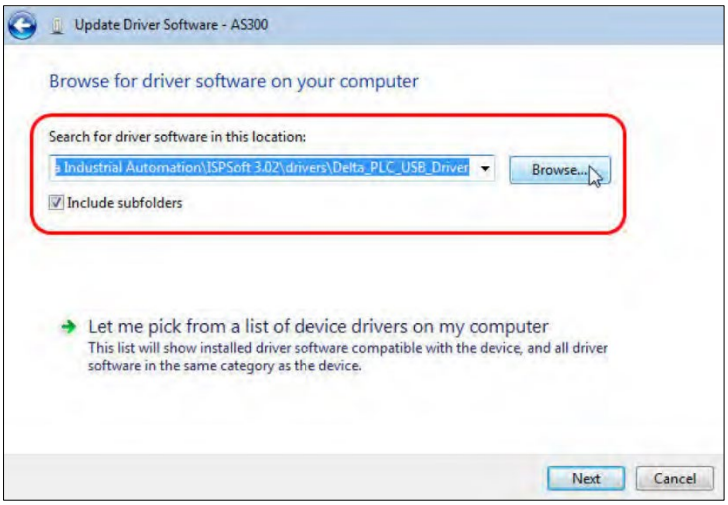


3. Click **Update Driver** to open **Update Driver Software – AS100/AS200/ Update Driver Software – AS300** dialog.

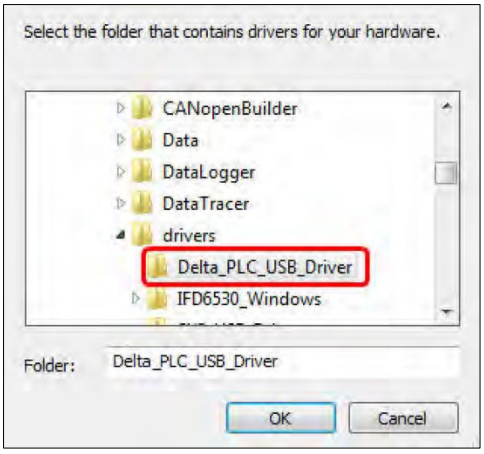


4. Click **Browse my computer for driver software** to open **Update Driver Software – AS300** dialog, showing the browse path of the driver.

   Note: After the installation of DIADesigner, the driver for AS series CPU module is available in the folder of installation path - `DIAStudio\DIADesigner\DIADesigner version\drivers\Delta_PLC_USB_Driver\`



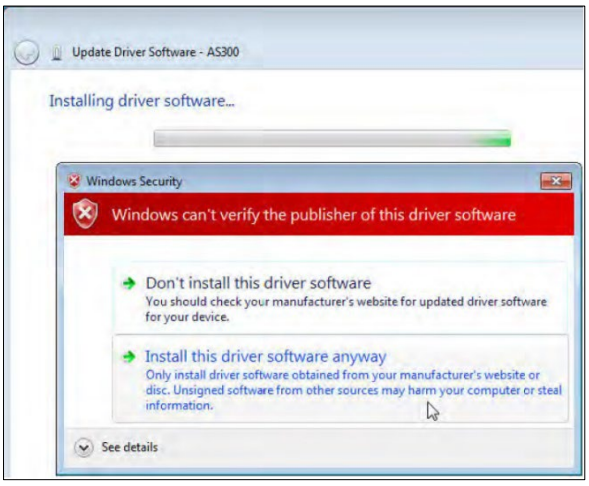
5. Click **Browse** and locate the correct path as per installation.



6. Click **OK** to select path.

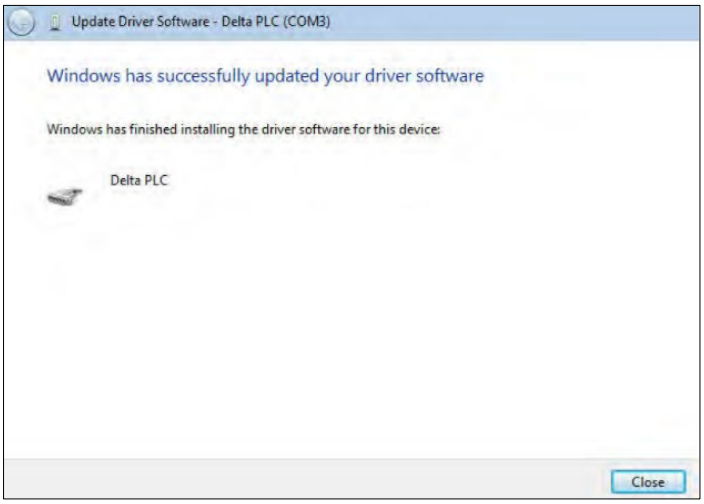
7. Click **Next** in **Update Driver Software – AS300** dialog.

   The system will start installing the driver. The **Windows Security** dialog appears during the installation.



8. Click **Install this driver software anyway**.

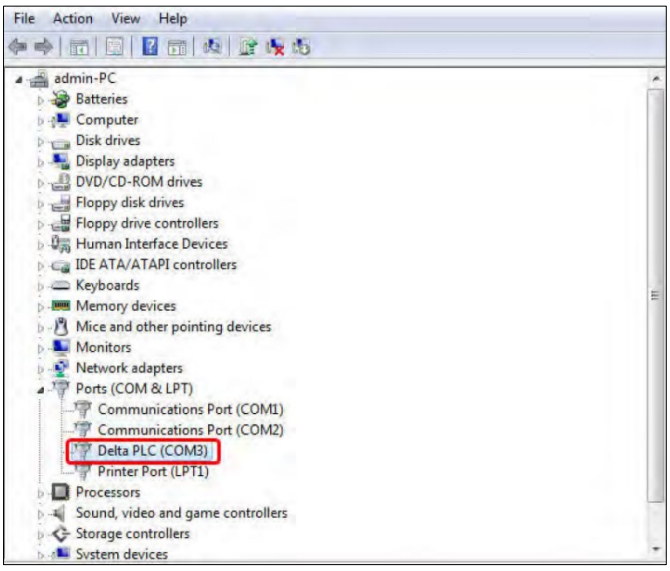
   After the installation is complete, the **Update Driver Software** dialog displays the status as shown in the following figure.



9. Click **Close**.

10. Open the **Device Manager** dialog after the installation is finished.

    If the name of the USB device connected is under **Ports (COM and LPT)**, the installation of the USB driver is successful. The operating system assigns a communication port number to the USB device as shown in the following example.



Note: If the PLC is connected to another USB port on the computer, the system can ask you to install the driver again. Follow the same steps as mentioned above, and

install the driver again. After the driver is installed, the communication port number that the operating system assigns to the USB device can be different.

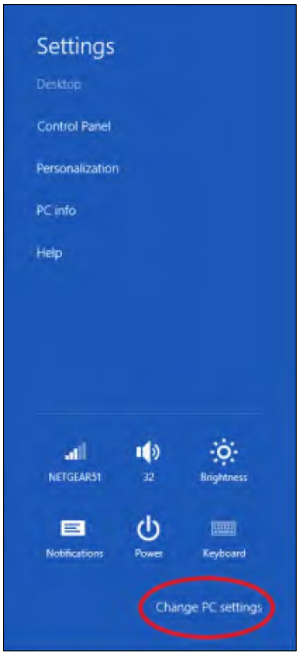
19.1.2 Installing USB Driver in Windows 8.1

Windows 8.1 driver signature enforcement provides a way to improve the security of the operating system by validating the integrity of a driver or system file each time it is loaded into memory. However, since Delta PLC USB driver does not include the driver signature, this section will help you to disable driver signature enforcement functionality in Windows 8.1 to ensure successful installation of USB driver.

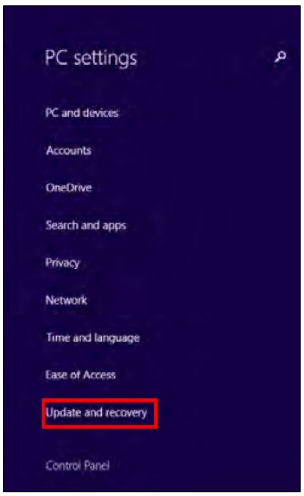
Follow these steps to disable driver signature enforcement in Windows 8.1.

To disable driver signature enforcement in Windows 8.1

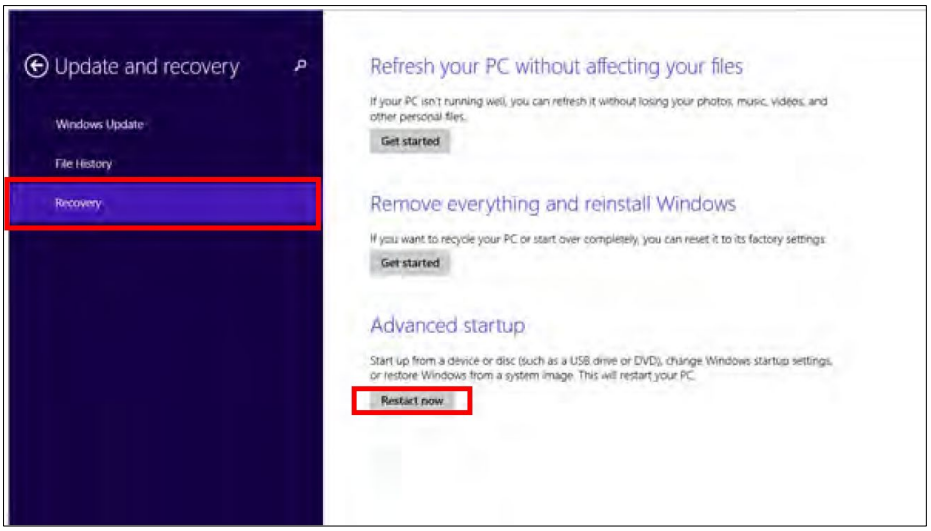
1. Press **Windows** () and **I** key simultaneously on the keyboard to open **Settings** interface.



2. Click **Change PC settings** in the **Settings** interface.

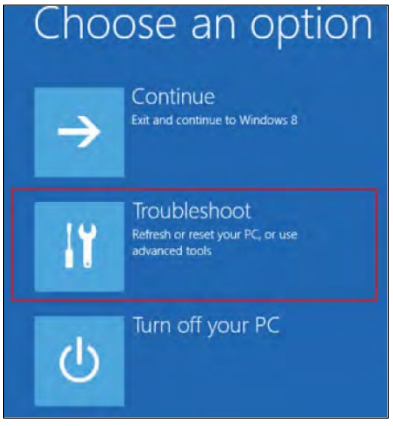


3. Click **Update and recovery** in the **PC settings** interface.



4. Click **Recovery** in the **Update and recovery** interface.
5. Click **Restart now** in the **Advanced startup** section in the **Refresh your PC without affecting your files**.

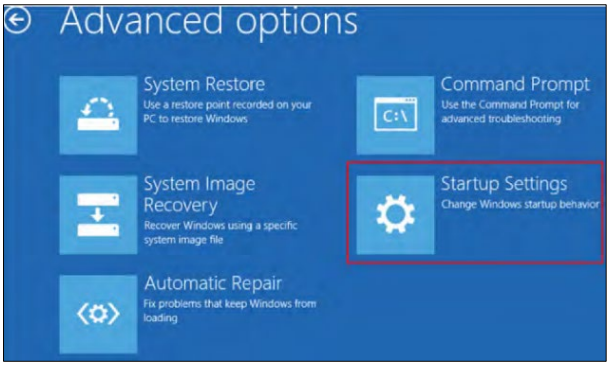
   Computer restarts, and it displays **Choose an option** interface.



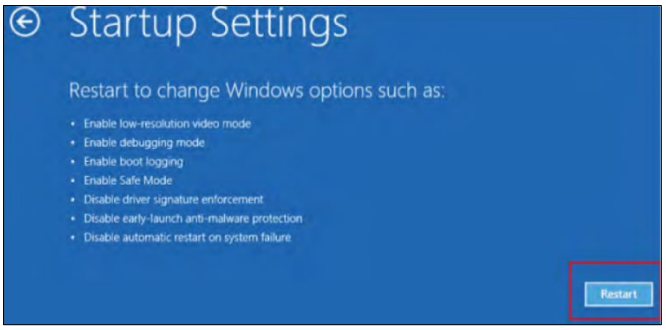
6. Click **Troubleshoot** in **Choose an option** interface.



7. Click **Advanced options** in **Troubleshoot** interface.



8. Click **Startup Settings** in the **Advanced options** interface.



9. Click **Restart** to restart computer.

10. After computer restarts, press **7** or **F7** to choose **Disable driver signature enforcement**.

    System directs you to Windows 8.1 operating page. Now you can install the Delta PLC USB driver. Procedure is same as the one mentioned in Windows 7.

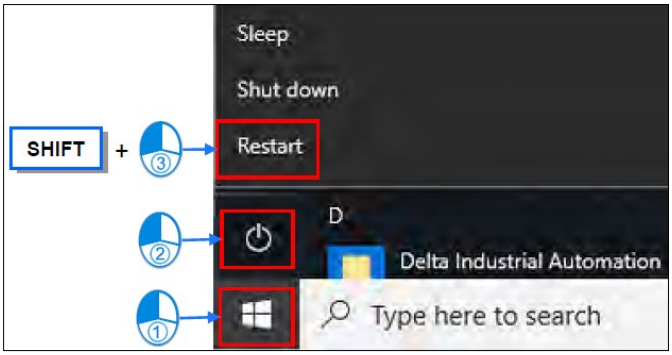
19.1.3 Installing USB Driver in Windows 10

Windows 10 driver signature enforcement provides a way to improve the security of the operating system by validating the integrity of a driver or system file each time it is loaded into memory. However, since Delta PLC USB driver does not include the driver signature, this section will help you to disable driver signature enforcement functionality in Windows 10 to ensure successful installation of USB driver. This operation is only valid for a single time. The setting will return to its original state after restarting.

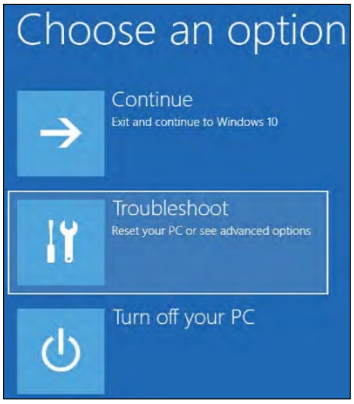
Follow these steps to disable driver signature enforcement in Windows 10 operating system.

To disable driver signature enforcement in Windows 10

1. Click **Windows** icon > **Power** icon and press **Shift** key and click **Restart**.



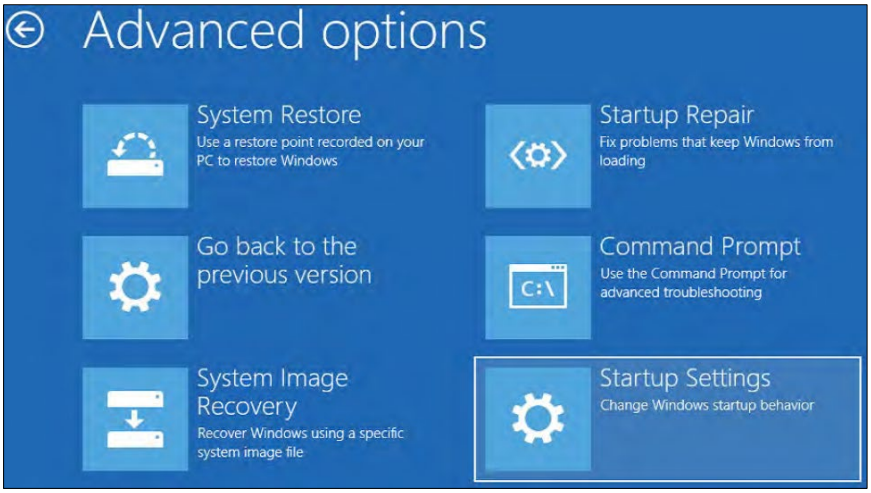
Computer restarts. **Choose an option** dialog appears.



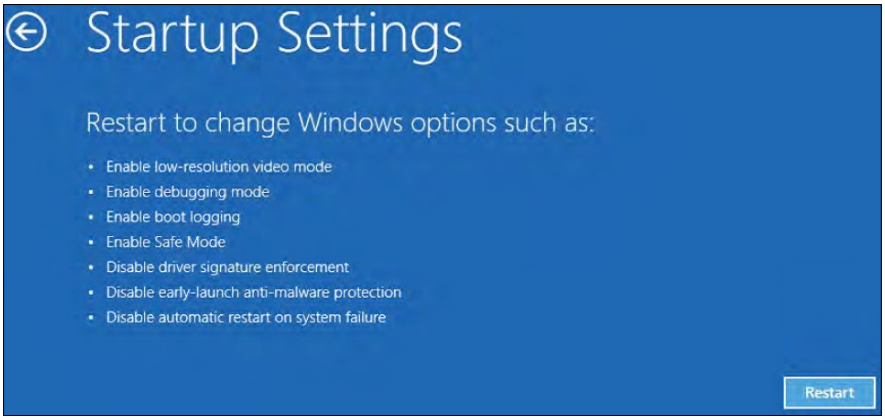
2. Click **Troubleshoot** in **Choose an option** dialog to open **Troubleshoot** dialog.



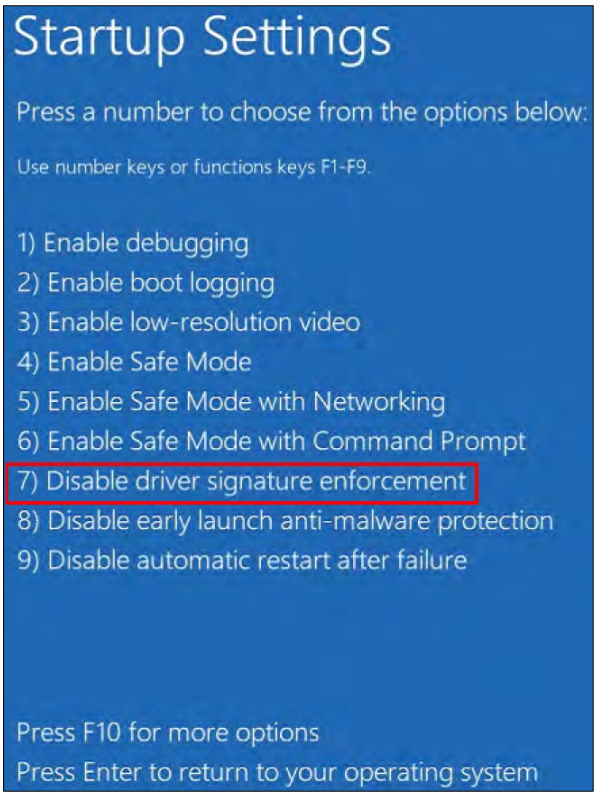
3. Click **Advanced options** in the **Troubleshoot** dialog to open **Advanced options** dialog.



4. Click **Startup Settings** in the **Advanced options** interface.



5. Click **Restart** to restart computer. **Startup Settings** dialog appears.



6. Press **7** or **F7** to choose **Disable driver signature enforcement**.

   System directs you to Windows 10 operating page. Now you can install the Delta PLC USB driver. Procedure is same as the one mentioned in Windows 7.

19.2 Important Points related to CPU Modules

Read the manuals of different models before you develop projects. The points to which you must pay attention when you develop a project for an AS series CPU module are discussed in this section.

19.2.1 Addresses in AS Series CPU Modules

19.2.1.1 Address Types of AS Series CPU Modules

The following table describes address type of AS series CPU modules.

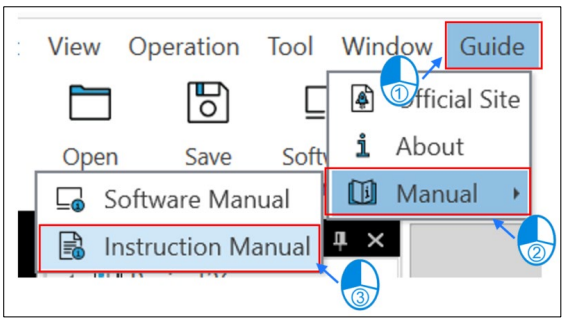
Address Type	Access Property	Description
X	Bit/Word	Input Relay It represents the state of a Digital Input. (*1)
Y	Bit/Word	Output Relay It represents the state of a Digital Output. (*1)
M	Bit	Auxiliary Relay The state of a bit is stored in an auxiliary relay.
S	Bit	Stepping Relay It is used as a state flag for a step in a Sequential Function Chart (SFC).
D	Bit/Word	Data Register Operation data is stored in a data register. (*1)
T	Bit	Timer If a timer is used as a bit address, the timer is ON when the time interval which is measured conforms to the setting value, and the timer is OFF when the time interval which is measured does not conform to the setting value.
	Word	Timer If a timer is used as a word address, the values stored in the timer is the present value of the timer.
C	Bit	Counter If a counter is used as a bit address, the counter is ON when the number of times a particular event or process has occurred conforms to the setting value, and the counter is OFF when the number of times a particular event or process has occurred does not conform to the setting value.
	Word	If a counter is used as a word address, the value stored in the counter is the present value of the counter.
HC	Bit	32-bit counter If a 32-bit counter is used as a bit address, the 32-bit counter is ON when the number of times a particular event or process has occurred conforms to the setting value, and the 32-bit counter is OFF when the number of times a particular event or process has occurred does not conform to the setting value
	Word	32-bit counter

Address Type	Access Property	Description
		If a 32-bit counter is used as a word address, the value stored in the 32-bit counter is the present value of the 32-bit counter.
E	Word	Index Register The value stored in an index register indicates the offset for an address modified. (*2)
SM	Bit	Special Auxiliary Register It is used as a state flag for a special function of the system. (*3)
SR	Word	Special Data Register It is used as a data register for a special function of the system (*3)

Note:

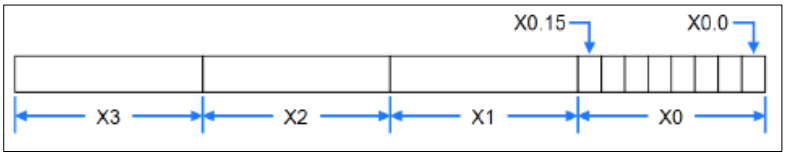
1. AS series CPU modules support the manipulation of the bits in the input relays, the output relays, the data registers, and the link registers. For example, X0.1 indicates that bit 1 in X0 is manipulated, Y0.1 indicates that bit 1 in Y0 is manipulated, D0.1 indicates that bit 1 in D0 is manipulated. Refer to the following sections for more information.
2. The Index Registers that AS series CPU modules have E address.
3. Different from the DVP model, the AS model separates the special auxiliary relay and the special data register into two types of devices, SM and SR.

Refer to the instruction manual in a project for a CPU module in DIADesigner **Help** > **Manual** > **Instruction Manual** in the menu bar.

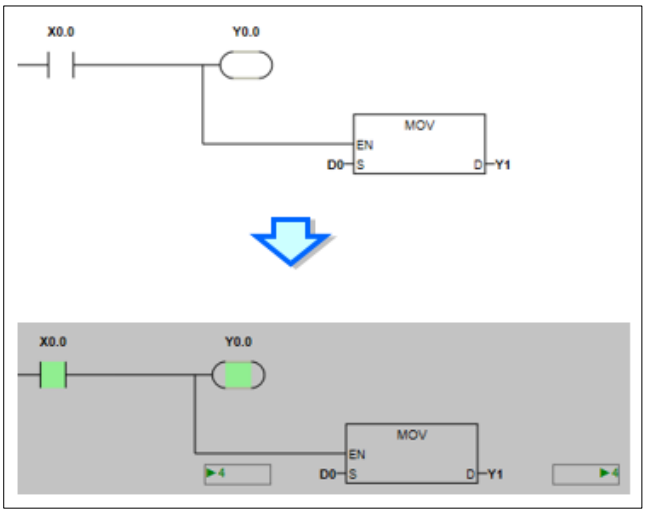


19.2.1.2 Manipulating the Bits in the X / Y / D / L Address for AS Series CPU Modules

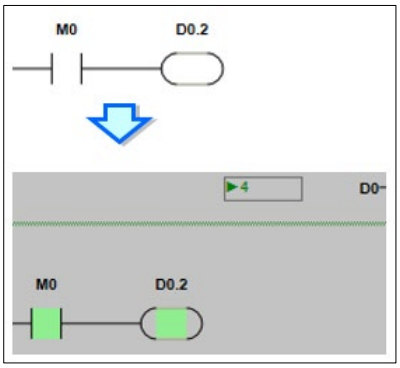
The X / Y addresses, which are used as bit, address in AS series CPU modules are represented by X0.0, Y0.0 etc. The AS model uses X0.0 to represent the first bit (Low Significant Bit, LSB) in X0 and X0.15 represents the last bit (Highest Significant Bit, HSB) in X0. The following figure shows bit-wise structure.



Example: In the following scenario, when X0.0 is set, value in D0 is moved to Y1 and the states of Y1.0 ~ Y1.15 will change.



The bit's operation can be used except for X / Y address in AS series CPU modules and can be manipulated. In the scenario shown in the following figure, the state of M0 is sent to D0.2. Suppose the initial value in D0 is 0. When M0 is ON, the value of D0 becomes 4.



19.2.2 Address Resources in AS Series CPU Modules

19.2.2.1 AS100 / AS200 / AS300 Address Resources

The following table lists address resources in AS series PLC – AS100 / AS200 / AS300.

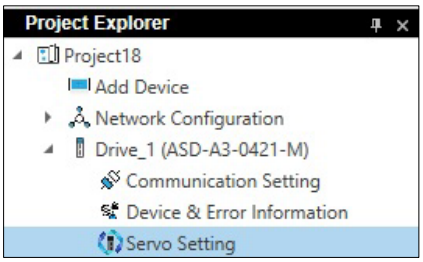
Address Type	Address Range	
X	Word: X0 ~ X63	Bit: X0.0 ~ X63.15
Y	Word: Y0 ~ Y63	Bit: Y0.0 ~ Y63.15
D	Word: D0 ~ D29999	Bit: D0.0 ~ D29999.15
M	M0 ~ M8191	
S	S0 ~ S2047	
C	C0 ~ C511	
HC	HC0 ~ HC255	
T	T0 ~ T511	
E	E0 ~ E14	
SR	SR0 ~ SR2047	
SM	SM0 ~ SM4095	

19.3 Servo Setting

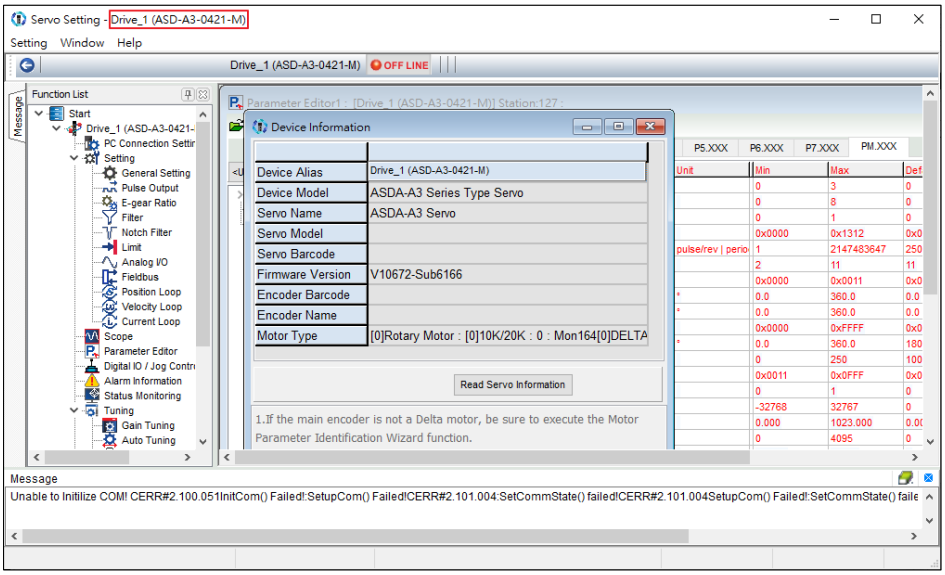
Open ASD-A3 / ASD-B3 Servo Setting through clicking **Servo Setting** node or click the context menu of the device in **Network View**. The respective descriptions are as follows.

- Servo Setting node

  Add an ASD-A3/ASD-B3 device, and expand the device.

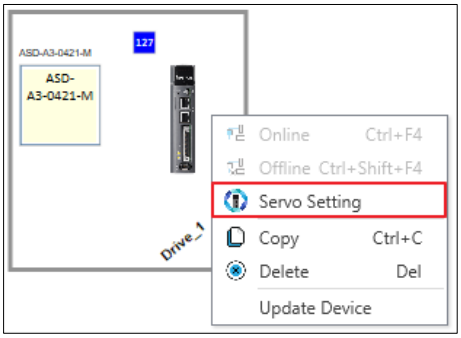


Double-click **Servo Setting** to open **Servo Setting** dialog. You can see the device model that is selected in Window title bar.

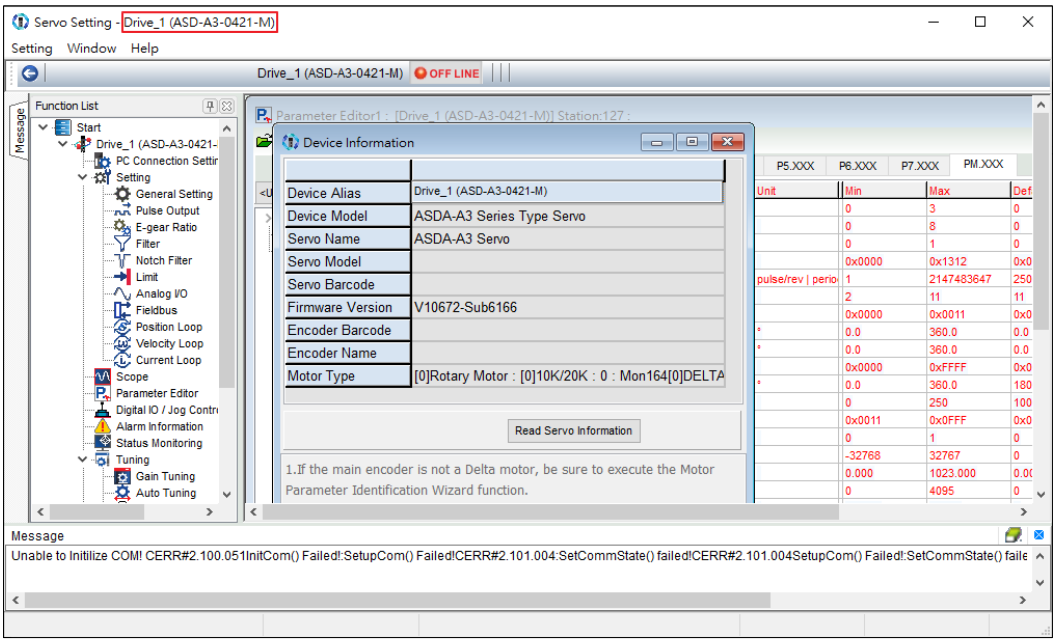


- Context menu of Network View

After adding ASD-A3 / ASD-B3 device, open Network View. Right-click the device to open the context menu.



Click **Servo Setting** to open the **Servo Setting** dialog. You can see the device model that is selected in Window Title Bar.

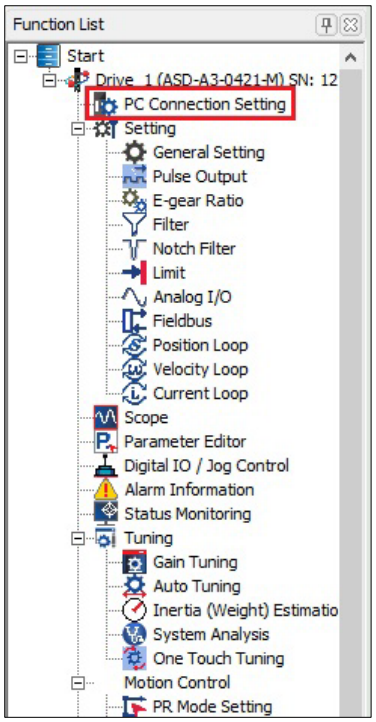


The **Servo Setting** dialogs of different devices can be opened at the same time. For more information on **Servo Setting** dialog usage, refer to the servo software manual.

Note:

- ASD-A3/ASD-B3 devices only support Serial Port connection (Connection through COMMGR is not supported).
- **Servo Setting** function cannot be used when the built-in oscilloscope is on.
- When **Servo Setting** dialog is open, the functions of project editing, device editing, and device upload / download cannot be executed.
- You cannot exit DIADesigner when the **Servo Setting** dialog is open.
- If device connection has been established before, click 

in Servo Setting to automatically execute the connection.
- To change the way to establish connection, click **PC Connection Setting** to open the **Communication Setting** dialog.



19.4 Common Hot Keys

Check the list and instructions below.

Function Description	Hot Key
New Project	Ctrl + N
Open Project	Ctrl + O
Close Project	Ctrl + W
Save Project	Ctrl + S
Exit	Alt + F4
Undo	Ctrl + Z
Redo	Ctrl + Y
Cut	Ctrl + X
Copy	Ctrl + C
Paste	Ctrl + V
Delete	Del
Toggle Bookmark	Ctrl + F12
Find	Ctrl + F

Function Description	Hot Key
Replace	Ctrl + H
Add Device	Ctrl + D
Project Compile	Ctrl + F11
Compile	F11
Download	Ctrl + U
Upload	Ctrl + L
Online	Ctrl + F4
Offline	Ctrl + F4
Run	F5
Stop	Shift + F8

19.5 Data Exchange Table Online Mode Error Code Table

19.5.1 EtherNet/IP Error Code

Error Code	Error Description
0x0005	Target reply unknown error code
0x0103	Transport class not supported.
0x0114	Vendor ID or product code mismatch.
0x0115	Device Type mismatch.
0x0116	Revision mismatch.
0x011E	Direction not supported.
0x011F	Invalid O->T network connection fixed/variable flag.
0x0120	Invalid T->O network connection fixed/variable flag.
0x0121	Invalid O->T network connection priority.
0x0122	Invalid T->O network connection priority.
0x0123	Invalid O->T network connection type.
0x0124	Invalid T->O network connection type.
0x0125	Invalid O->T network connection redundant owner.
0x0126	Invalid configuration size.

Error Code	Error Description
0x0129	Invalid configuration application path.
0x0132	Null forward open function not supported.
0x0106	Ownership conflict.
0x0113	Out of connections.
0x0302	Network bandwidth (PPS) not available for data.
0x0111	RPI not supported.
0x0100	Connection in use or duplicate forward open.
0x0110	Target for connection not configured.
0x0119	Non-Listen only connection not opened.
0x0204	Unconnected request timed out.
0x0128	Invalid T->O network connection size
0x0127	Invalid O->T network connection size
0x012D	Consuming symbol does not exist.
0x012E	Producing symbol does not exist.
0x0315	Invalid segment in connection path.
0x0203	The connection timeout.
0x0319	Secondary resources unavailable.
0x012A	Invalid O -> T path
0x012B	Invalid T -> O path
0x0107	Target connection not found
0x0108	Invalid network connection parameter
0x0109	Invalid connection size
0x0112	RPI value not acceptable
0x0117	Invalid produced or consumed application path
0x0118	Invalid or inconsistent configuration application path
0x011A	Target object out of connections
0x011B	The Production Inhibit Time is greater than the Target to Originator RPI
0x011C	Transport class not supported
0x011D	T->O production trigger not supported
0x012C	Configuration symbol does not exist
0x012F	Inconsistent application path combination
0x0130	Inconsistent consume data format

Error Code	Error Description
0x0131	Inconsistent produce data format
0x0134	Mismatched T->O network connection size
0x0135	Mismatched T->O network connection Fixed/Variable bit
0x0136	Mismatched T->O network connection priority
0x0137	Mismatched transport class
0x0138	Mismatched T->O production trigger
0x0139	Mismatched T->O production inhibit time segment
0x0205	Parameter error in Unconnected request service
0x0206	Message too large for Unconnected send service
0x0207	Unconnected acknowledge without reply
0x0301	No buffer memory available
0x0303	No consumed connection id filter available
0x0304	Not configured to send scheduled priority data
0x0305	Schedule signature mismatch
0x0306	Schedule signature validation not possible
0x0311	Port not available
0x0312	Link address not valid
0x0316	Forward close service connection path mismatch
0x0317	Schedule not specified
0x0318	Link address to self-invalid
0x031A	Rack connection already established
0x031B	Module connection already established
0x031C	Miscellaneous
0x031D	Redundant connection mismatch
0x031E	No more user configurable link consumer resources an available in the producing module
0x031F	No user configurable link consumer resources configured in the producing module
0x0800	Network link offline
0x0810	No target application data available
0x0811	No originator application data available
0x0812	Node address has changed since the network was scheduled
0x0813	No configured for off-subnet multicast

Error Code	Error Description
0x0814	Invalid produce/consume data format
0xFFEA	Invalid T->O data address
0xFFEB	Invalid T->O RPI
0xFFF3	Invalid O->T data address
0xFFF4	Invalid O-> RPI
0xFFFA	I/O Connection header not valid
0xFFFC	Unconnected send service is busy
0xFFFD	Invalid T->O connection parameter
0xFFFE	Originator CIP out of connections
0xFFFF	Originator TCP out of connections
-	Other Exception

19.5.2 Modbus Error Code

Error Code	Error Description
0x0001	The function code received in the query is not an allowable action for the server (or slave). This may be because the function code is only applicable to newer devices, and was not implemented in the unit selected. It could also indicate that the server (or slave) is in the wrong state to process a request of this type, for example because it is unconfigured and is being asked to return register values.
0x0002	The data address received in the query is not an allowable address for the server (or slave). More specifically, the combination of reference number and transfer length is invalid. For a controller with 100 registers, the PDU addresses the first register as 0, and the last one as 99. If a request is submitted with a starting register address of 96 and a quantity of registers of 4, then this request will successfully operate (address-wise at least) on registers 96, 97, 98, 99. If a request is submitted with a starting register address of 96 and a quantity of registers of 5, then this request will fail with Exception Code 0x02 “Illegal Data Address” since it attempts to operate on registers 96, 97, 98, 99 and 100, and there is no register with address 100.
0x0003	A value contained in the query data field is not an allowable value for server (or slave). This indicates a fault in the structure of the remainder of a complex request, such as that the implied length is incorrect. It specifically does NOT mean that a data item submitted for storage in a register has a value outside the expectation of the

Error Code	Error Description
	application program, since the MODBUS protocol is unaware of the significance of any particular value of any particular register.
0x0004	An unrecoverable error occurred while the server (or slave) was attempting to perform the requested action.
0x0005	Specialized use in conjunction with programming commands. The server (or slave) has accepted the request and is processing it, but a long duration of time will be required to do so. This response is returned to prevent a timeout error from occurring in the client (or master). The client (or master) can next issue a Poll Program Complete message to determine if processing is completed.
0x0006	Specialized use in conjunction with programming commands. The server (or slave) is engaged in processing a long–duration program command. The client (or master) should retransmit the message later when the server (or slave) is free.
0x0007	Checksum error or data format error
0x0008	Specialized use in conjunction with function codes 20 and 21 and reference type 6, to indicate that the extended file area failed to pass a consistency check.
0x0009	Communication timeout
0x000A	Specialized use in conjunction with gateways, indicates that the gateway was unable to allocate an internal communication path from the input port to the output port for processing the request. Usually means that the gateway is misconfigured or overloaded.
0x000B	Specialized use in conjunction with gateways, indicates that no response was obtained from the target device. Usually means that the device is not present on the network.

19.5.3 Modbus TCP Error Code

Error Code	Error Description
0x0001	The function code received in the query is not an allowable action for the server (or slave). This may be because the function code is only applicable to newer devices, and was not implemented in the unit selected. It could also indicate that the server (or slave) is in the wrong state to process a request of this type, for example because it is unconfigured and is being asked to return register values.
0x0002	The data address received in the query is not an allowable address for the server (or slave). More specifically, the combination of reference number and transfer length is invalid. For a controller with 100 registers, the PDU addresses the first register

Error Code	Error Description
	as 0, and the last one as 99. If a request is submitted with a starting register address of 96 and a quantity of registers of 4, then this request will successfully operate (address-wise at least) on registers 96, 97, 98, 99. If a request is submitted with a starting register address of 96 and a quantity of registers of 5, then this request will fail with Exception Code 0x02 “Illegal Data Address” since it attempts to operate on registers 96, 97, 98, 99 and 100, and there is no register with address 100.
0x0003	A value contained in the query data field is not an allowable value for server (or slave). This indicates a fault in the structure of the remainder of a complex request, such as that the implied length is incorrect. It specifically does NOT mean that a data item submitted for storage in a register has a value outside the expectation of the application program, since the MODBUS protocol is unaware of the significance of any particular value of any particular register.
0x0004	An unrecoverable error occurred while the server (or slave) was attempting to perform the requested action.
0x0005	Specialized use in conjunction with programming commands. The server (or slave) has accepted the request and is processing it, but a long duration of time will be required to do so. This response is returned to prevent a timeout error from occurring in the client (or master). The client (or master) can next issue a Poll Program Complete message to determine if processing is completed.
0x0006	Specialized use in conjunction with programming commands. The server (or slave) is engaged in processing a long–duration program command. The client (or master) should retransmit the message later when the server (or slave) is free.
0x0007	Checksum error or data format error
0x0008	Specialized use in conjunction with function codes 20 and 21 and reference type 6, to indicate that the extended file area failed to pass a consistency check.
0x0009	Communication timeout
0x000A	Specialized use in conjunction with gateways, indicates that the gateway was unable to allocate an internal communication path from the input port to the output port for processing the request. Usually means that the gateway is misconfigured or overloaded.
0x000B	Specialized use in conjunction with gateways, indicates that no response was obtained from the target device. Usually means that the device is not present on the network.
0xF000	Ethernet is not connected.
0xF001	There is no response from remote device.
0xF003	TCP connection is communication timeout.

Error Code	Error Description
0xF007	Device responds command error.
0xF009	The remote device disconnected.

19.6 Project Migrate and Unsupported Functions

This section describes the unsupported AS / DVP series functions by DIADesigner and the project migration effect.

- **AS series**
  - The related functions of IO-Link are not supported. During project conversion, the AS04SIL module can be configured, however, the IO-Link settings cannot be converted.
  - During project conversion, the AS-FCOPM function card can be configured, however, the data exchange settings will be cleared. You need to reset the data exchange of CANopen Builder in DIADesigner.
- **DVP series**

  During project conversion, users will be navigated to convert the old DVP series models to the third-generation DVP series models, and the third-generation DVP series models no longer support some functions of the old series models.
  - Programming-related function
    - The KxMx instruction structure is not supported.
    - STL instruction is not supported.
    - The special D / special M addresses will be automatically converted to the SR / SM addresses of the corresponding function. The special D / special M addresses that the third-generation DVP series model cannot support will be displayed on the project conversion report.

    Note: The unsupported functions will be displayed on the project conversion report and the compilation result.
  - Configuration-related function

    For example, the settings of PLCLink and EtherLink will be cleared. You need

to reconfigure the settings in **Network View** of DIADesigner.

19.7 Device and Software Version Compatible Table

Controller

Type	Version					
	V1.7	V1.8	V1.9	V1.10	V1.11	V1.13
AS100 Series						
AS132P-A	V	V	V	V	V	V
AS132R-A	V	V	V	V	V	V
AS132T-A	V	V	V	V	V	V
AS148P-A	V	V	V	V	V	V
AS148R-A	V	V	V	V	V	V
AS148T-A	V	V	V	V	V	V
AS164P-A	V	V	V	V	V	V
AS164R-A	V	V	V	V	V	V
AS164T-A	V	V	V	V	V	V
AS200 Series						
AS218PX-A	V	V	V	V	V	V
AS218RX-A	V	V	V	V	V	V
AS218TX-A	V	V	V	V	V	V
AS228P-A	V	V	V	V	V	V
AS228R-A	V	V	V	V	V	V
AS228T-A	V	V	V	V	V	V
AS300 Series						
AS300N-A	V	V	V	V	V	V
AS320P-B	V	V	V	V	V	V
AS320T-B	V	V	V	V	V	V
AS324MT-A	V	V	V	V	V	V
DVP-ES3 Series	V	V	V	V	V	V
DVP-ES3(TEC) Series	V	V	V	V	V	V
DVP-EX3 Series	V	V	V	V	V	V
DVP-SV3 Series	V	V	V	V	V	V

Type	Version					
	V1.7	V1.8	V1.9	V1.10	V1.11	V1.13
DVP-SX3 Series	V	V	V	V	V	V

Note: V with gray background color represents new series supported in this version.

AC Motor Drive

Type	Version					
	V1.7	V1.8	V1.9	V1.10	V1.11	V1.13
C2000 Family						
C2000 Series	V	V	V	V	V	V
C2000 Plus Series	V	V	V	V	V	V
C2000-HS Series	V	V	V	V	V	V
CFP2000 Series	V	V	V	V	V	V
CH2000 Series	V	V	V	V	V	V
CP2000 Series	V	V	V	V	V	V
M300 Family						
ME300 Series	V	V	V	V	V	V
ME300-HS Series	V	V	V	V	V	V
MH300 Series	V	V	V	V	V	V
MH-300-HS Series	V	V	V	V	V	V
MS300 Series	V	V	V	V	V	V
MS300-HS Series	V	V	V	V	V	V
MX300-C Series		V	V	V	V	V
MX300-E Series		V	V	V	V	V
MX300-H Series					V	V
VFD-EL Family						
VFD-EL Series	V	V	V	V	V	V
VFD-EL-C Series	V	V	V	V	V	V
Industry-specific						
EB3000	V	V	V	V	V	V

Type	Version					
	V1.7	V1.8	V1.9	V1.10	V1.11	V1.13
MH300-L Series	V	V	V	V	V	V
MPD / MP300 Series	V	V	V	V	V	V
VFD-PA	V	V	V	V	V	V
AFE3000					V	V
V3000 Family						
VP3000 Series	V	V	V	V	V	V
VP3000-E Series			V	V	V	V
VP3000-EHS Series			V	V	V	V

Note: V with gray background color represents new series supported in this version.

Servo Drive

Type	Version					
	V1.7	V1.8	V1.9	V1.10	V1.11	V1.13
ASD-A2 Series						
ASD-A2-L Series	V	V	V	V	V	V
ASD-A2-M Series	V	V	V	V	V	V
ASD-A3 Series						
ASD-A3-L Series	V	V	V	V	V	V
ASD-A3-M Series	V	V	V	V	V	V
ASD-B2 Series						
ASD-B2-B Series	V	V	V	V	V	V
ASD-B2-F Series	V	V	V	V	V	V
ASD-B3 Series						
ASD-B3-L Series	V	V	V	V	V	V
ASD-B3-M Series	V	V	V	V	V	V

Note: V with gray background color represents new series supported in this version.

Temperature Controller

Type	Version					
	V1.7	V1.8	V1.9	V1.10	V1.11	V1.13
DTC Series	V	V	V	V	V	V

Note: V with gray background color represents new series supported in this version.

19.8 Function and Software / Firmware Version Compatible Table

The following table lists functions that need to be used with specific software and firmware versions.

Function	Software Version	Firmware Version
DIO module supported configured by DVP series. Refer to Adding Module.	DIADesigner version 1.9.0 or above	DVP series: Version 1.10 or above
Transcode optimization. Refer to Programming Properties Setting.	DIADesigner version 1.9.0 or above	AS series: Version 1.16 or above DVP series: Version 1.10 or above
EtherNet/IP exchange table assigns X, Y, and W registers.	DIADesigner version 1.9.0 or above	AS series: Version 1.16 or above DVP-SV3/SX3/ES3 series: Version 1.10 or above DVP-EX3/ES3 (TEC) series: Version 1.08 or above

Smarter. Greener. Together.

Industrial Automation Headquarters

Delta Electronics, Inc.
Taoyuan Technology Center
No.18, Xinglong Rd., Taoyuan District,
Taoyuan City 33068, Taiwan
TEL: 886-3-362-6301 / FAX: 886-3-371-6301

Asia

Delta Electronics (Shanghai) Co., Ltd.
No.182 Minyu Rd., Pudong Shanghai, P.R.C.
Post code : 201209
TEL: 86-21-6872-3988 / FAX: 86-21-6872-3996
Customer Service: 400-820-9595

Delta Electronics (Japan), Inc.
Tokyo Office
Industrial Automation Sales Department
2-1-14 Shibadaimon, Minato-ku
Tokyo, Japan 105-0012
TEL: 81-3-5733-1155 / FAX: 81-3-5733-1255

Delta Electronics (Korea), Inc.
Seoul Office
1511, 219, Gasan Digital 1-Ro., Geumcheon-gu,
Seoul, 08501 South Korea
TEL: 82-2-515-5305 / FAX: 82-2-515-5302

Delta Energy Systems (Singapore) Pte Ltd.
4 Kaki Bukit Avenue 1, #05-04, Singapore 417939
TEL: 65-6747-5155 / FAX: 65-6744-9228

Delta Electronics (India) Pvt. Ltd.
Plot No.43, Sector 35, HSIIDC Gurgaon,
PIN 122001, Haryana, India
TEL: 91-124-4874900 / FAX : 91-124-4874945

Delta Electronics (Thailand) PCL.
909 Soi 9, Moo 4, Bangpoo Industrial Estate (E.P.Z),
Pattana 1 Rd., T.Phraksa, A.Muang,
Samutprakarn 10280, Thailand
TEL: 66-2709-2800 / FAX : 662-709-2827

Delta Electronics (Australia) Pty Ltd.
Unit 20-21/45 Normanby Rd., Notting Hill Vic 3168, Australia
TEL: 61-3-9543-3720

Americas

Delta Electronics (Americas) Ltd.
Raleigh Office
P.O. Box 12173, 5101 Davis Drive,
Research Triangle Park, NC 27709, U.S.A.
TEL: 1-919-767-3813 / FAX: 1-919-767-3969

Delta Greentech (Brasil) S/A
São Paulo Office
Rua Itapeva, 26 – 3˚ Andar - Bela Vista
CEP: 01332-000 – São Paulo – SP - Brasil
TEL: 55-11-3530-8643 / 55-11-3530-8640

Delta Electronics International Mexico S.A. de C.V.
Mexico Office
Gustavo Baz No. 309 Edificio E PB 103
Colonia La Loma, CP 54060
Tlalnepantla, Estado de México
TEL: 52-55-3603-9200

EMEA

Headquarters: Delta Electronics (Netherlands) B.V.
Sales: Sales.IA.EMEA@deltaww.com
Marketing: Marketing.IA.EMEA@deltaww.com
Technical Support: iatechnicalsupport@deltaww.com
Customer Support: Customer-Support@deltaww.com
Service: Service.IA.emea@deltaww.com
TEL: +31(0)40 800 3900

BENELUX: Delta Electronics (Netherlands) B.V.
De Witbogt 20, 5652 AG Eindhoven, The Netherlands
Mail: Sales.IA.Benelux@deltaww.com
TEL: +31(0)40 800 3900

DACH: Delta Electronics (Netherlands) B.V.
Coesterweg 45, D-59494 Soest, Germany
Mail: Sales.IA.DACH@deltaww.com TEL:
+49(0)2921 987 0

France: Delta Electronics (France) S.A.
ZI du bois Challand 2, 15 rue des Pyrénées,
Lisses, 91090 Evry Cedex, France
Mail: Sales.IA.FR@deltaww.com
TEL: +33(0)1 69 77 82 60

Iberia: Delta Electronics Solutions (Spain) S.L.U
Ctra. De Villaverde a Vallecas, 265 1º Dcha Ed.
Hormigueras – P.I. de Vallecas 28031 Madrid
TEL: +34(0)91 223 74 20

Carrer Llacuna 166, 08018 Barcelona, Spain
Mail: Sales.IA.Iberia@deltaww.com

Italy: Delta Electronics (Italy) S.r.l.
Via Meda 2–22060 Novedrate(CO)
Piazza Grazioli 18 00186 Roma Italy Mail:
Sales.IA.Italy@deltaww.com
TEL: +39 039 8900365

Russia: Delta Energy System LLC
Vereyskaya Plaza II, office 112 Vereyskaya str.
17 121357 Moscow Russia
Mail: Sales.IA.RU@deltaww.com
TEL: +7 495 644 3240

Turkey: Delta Greentech Elektronik San. Ltd. Sti. (Turkey)
Şerifali Mah. Hendem Cad. Kule Sok. No:16-A
34775 Ümraniye – İstanbul
Mail: Sales.IA.Turkey@deltaww.com
TEL: + 90 216 499 9910

GCC: Delta Energy Systems AG (Dubai BR)
P.O. Box 185668, Gate 7, 3rd Floor, Hamarain Centre
Dubai, United Arab Emirates
Mail: Sales.IA.MEA@deltaww.com
TEL: +971(0)4 2690148

Egypt + North Africa: Delta Electronics
Unit 318, 3rd Floor, Trivium Business Complex, North 90 street,
New Cairo, Cairo, Egypt
Mail: Sales.IA.MEA@deltaww.com

DIAS-Manual-0003-EN

*We reserve the right to change the information in this catalogue without prior notice.