# Newsletter

Industrial Automation Products

Edition 2022-Q4

## Contents





CODESYS

# 1 News

## 1.1 FTP-Site Link & Update

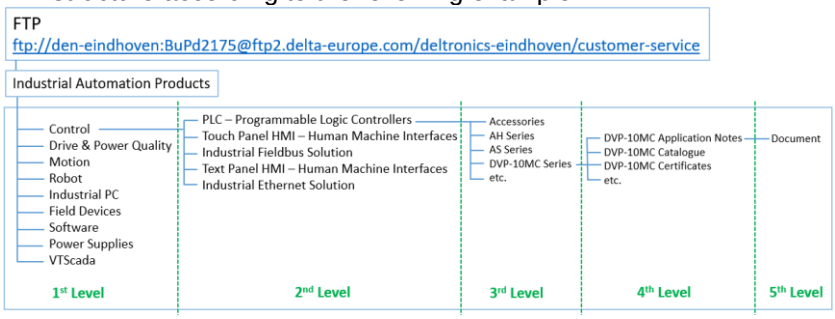
Our FTP server provides product information that is not available in Delta's Download Center on the global website, e.g. datasheets, technical notes, presentations, software, etc. Please visit our FTP site with below account info.

ftp://den-eindhoven:BuPd2175@ftp2.delta-europe.com/deltronics-eindhoven/customer-service

Name and password are included in the link.
Name: den-eindhoven
Password: BuPd2175

- **NOTE** It is only possible to access our FTP via TCP port 22/23. Therefore, please use common FTP clients such as FileZilla, Win SCP or Total Commander. Access with standard web browsers, like Edge, Chrome, Opera, etc. is not possible.

- **Update** To align the data categories with Delta's official Download Center, we adjusted the folder structure according to the following example.




## 1.2 NEW IABG EMEA HQ Officially Inaugurated

In October, Delta inaugurated a new facility at the Automotive Campus in Helmond, the Netherlands in the presence of Delta's top management and the Mayor of Helmond. The building will eventually house over 150 employees to support the expansion of Delta's industrial automation, industrial power supply and automotive business development, product testing and technical service in Europe, the Middle East & Africa (EMEA). Through the implementation of Delta's smart energy-saving solutions and innovative eco-friendly design, the new 4,055-square meter facility will consume an expected 56.84% less electricity than traditional buildings. In recognition of those efforts, the U.S. Green Building Council has awarded a LEED Gold green building certificate to the Delta Helmond office.


# 2 Product update

## 2.1 NEW – AX-3 Controllers, Programmable with CoDeSys

The AX-3 series is a range of professional PLCs and PLC-based motion controllers. It provides an advanced EtherCAT motion solution, based on the established AS PLC hardware platform. That allows expanding the system with AS series IO modules. The CoDeSys-based DIADesigner-AX software provides a convenient development environment and excellent product experience.

Delta releases two new AX-3N controllers to cover different application requirements.

The AX-3N possesses the prevalent Ethernet/IP and Modbus TCP fieldbuses as well as integrated OPC UA server functionality. The results are easy system integration and high-speed communication.

CPUs with or without built-in IO ensure that you always encounter the best matching controller for any application scenario.

### Features

AX-3N logic controller

- Min. execution time of basic instruction: 5 ns
- Built-in communication ports: Ethernet, RS-232, RS-485
- Ethernet port supports Ethernet/IP and Modbus TCP
- OPC UA server
- Serial ports support Modbus
- Built-in 16 DI (4 x high-speed 200 kHz) and 8 DO (4 x high-speed 200 kHz for pulse-train support up to 6 /15 axes), except AX-300N

### Electrical Specifications

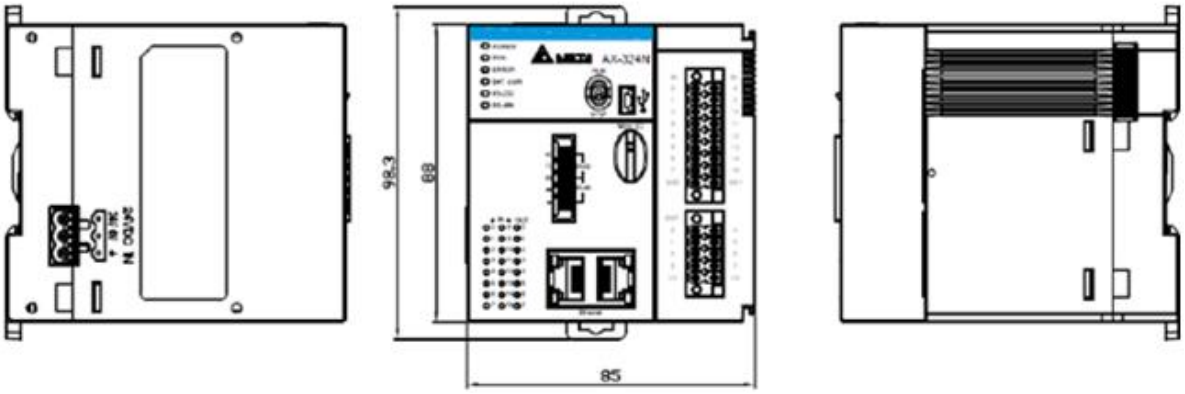
| | |
|---|---|
| Operating Temperature | -20 – +55 °C |
| Storage Temperature | -40 – +80 °C |
| Operating & Storage Humidity | 5 – 95%, non-condensing |
| Vibration | IEC 61131-2, IEC 60068-2-6 (TEST Fc);<br>5 Hz ≤ f ≤ 8.4 Hz, constant amplitude 3.5 mm;<br>8.4 ≤ f ≤ 150 Hz, constant acceleration 1 g |
| Shock | IEC 61131-2, IEC 60068-2-6 (TEST Ea); |

| | |
|---|---|
| | 15 g peak, 11 ms duration, half-sine |
| Operating Environment | Non-corrosive gas |
| Installation | Inside of control panel |
| Pollution Degree | 2 |
| Protection Rating | IP 20 |
| Conformal Coating | Yes |
| Conformity | CE, UL |

## Dimensions

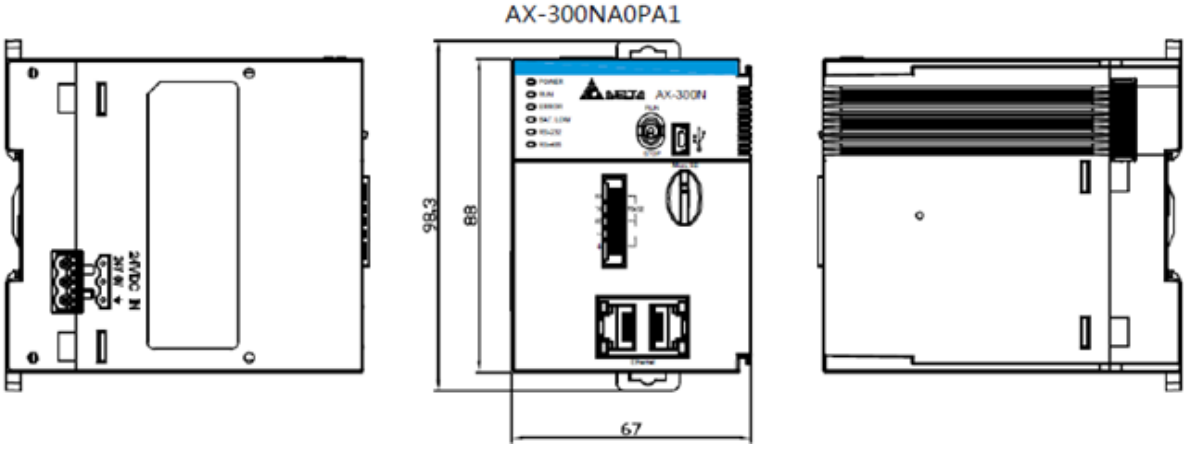
(Unit: mm)

AX-324NA0PA1P




(Unit: mm)

AX-300NA0PA1




## Ordering Information

| Model | Description |
|---|---|
| AX-308EA0MA1T | AX-3 series EtherCAT motion controller, 8-axis, built-in 16DI/8DO, NPN output, CODESYS |
| AX-324NA0PA1P NEW! | AX-3 series logic controller, built-in 16DI/8DO, PNP output, CODESYS |
| AX-300NA0PA1 NEW! | AX-3 series logic controller, no built-in IO, CODESYS |

## 2.2 UPDATE – AX3 Firmware Upgraded to Version 1.00.04

### Related Models

| Series | Model | Firmware Version | Release Date |
|---|---|---|---|
| AX-3 | AX-308EA0MA1P<br>AX-308EA0MA1T<br>AX-316EA0MA1T<br>AX-304ELA0MA1P<br>AX-304ELA0MA1T<br>AX-364ELA0MA1T<br>AX-324NA0PA1P<br>AX-300NA0PA1 | 1.00.04 | October 28, 2022<br>(Week 44/2022) |

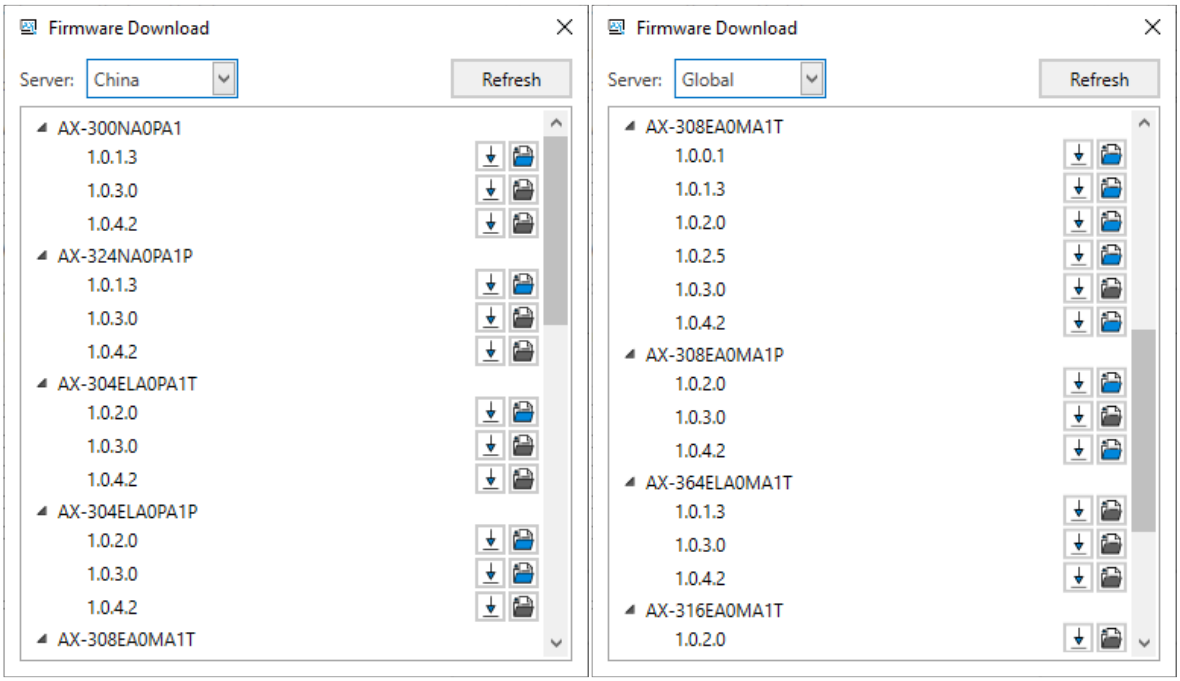
### Compatibility

Since the built-in memory card on the PCB board has been replaced with an eMMC IC, the hardware version is updated and the compatibility between the hardware and firmware is described below.

| Series | Model | Hardware Version*1 | Firmware Version*2 |
|---|---|---|---|
| AX-3 | AX-308EA0MA1T | A0 | V1.00.00 or later<br>Shown as 1.0.0 in DIADesigner-AX |
| | | A1 | V1.00.02.05 or later<br>Shown as 1.0.2.5 in DIADesigner-AX |
| | | B1 | V1.00.04 or later<br>Shown as 1.0.4.1 in DIADesigner-AX |
| | AX-300NA0PA1<br>AX-304ELA0PA1T<br>AX-304ELA0PA1P<br>AX-308EA0MA1P<br>AX-316EA0MA1T<br>AX-324NA0PA1P<br>AX-364ELA0MA1T | A0 | V1.00.00 or later<br>Shown as 1.0.0 in DIADesigner-AX |
| | | A1 | V1.00.03 or later<br>Shown as 1.0.3.0 in DIADesigner-AX |
| | | B1 | V1.00.04 or later<br>Shown as 1.0.4.2 in DIADesigner-AX |

*1 Since the built-in memory card has been replaced with an eMMC IC on the PCB board, the hardware version is updated from A1 to B1.

*2 Old hardware version A0 is compatible to all firmware versions, new and old. The hardware version A1 is only compatible to firmware version 1.00.03 or later. The hardware version B1 is only compatible to firmware version 1.00.04 or later. Use Firmware Package Upgrade function from DIADesigner-AX to upgrade your firmware.




## New functions

1. Now AS00SCM can be installed on the right side of AX-3 series PLC CPU. (works with DIADesigner-AX V1.4)

2. New memory mode, the device type “%M” is available for DDF V1.04.0 or later. Users can set a range for %M. and the variable type can be defined by the setting range of the %M. (works with DIADesigner-AX V1.4)

| **Variable and %M** | **Memory Mode = Retain** | **Memory Mode = %M** | |
|---|---|---|---|
| | **%M** | **%M in the setting range** | **%M out of the setting range** |
| **VAR (@ %M)** | VAR | Persistent retain | VAR |
| **Retain (@ %M)** | Retain | Persistent retain | Retain |
| **Persistent retain (@ %M)** | Persistent retain | Persistent retain | Persistent retain |
| **@ %M** | Persistent retain | Persistent retain | Persistent retain |

See the table below to check if the values for different types of variables are retainable after certain action is taken.

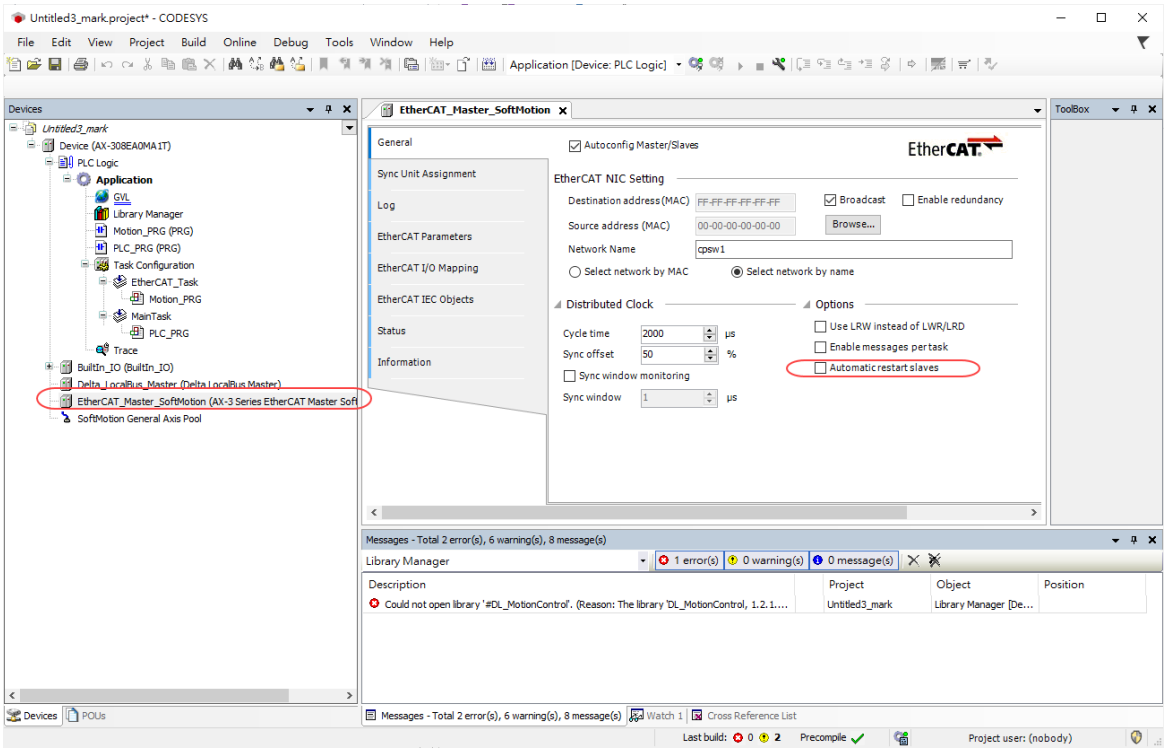
| **Action** | **VAR** | **Retain** | **Persistent retain** |
|---|---|---|---|
| Online change | ● | ● | ● |
| Reboot PLC | ○ | ● | ● |
| Reset Warm | ○ | ● | ● |
| Reset Cold | ○ | ○ | ● |
| Download | ○ | ○ | ● |
| Reset Origin | ○ | ○ | ● |

● = Value retained*; ○ = Clear to zero
* If the value is not retainable, default values will be used instead

### Improvements

1. Modified the RAM checking mechanism to minimize the chance of not able to turn the AX-3 series PLC CPU ON

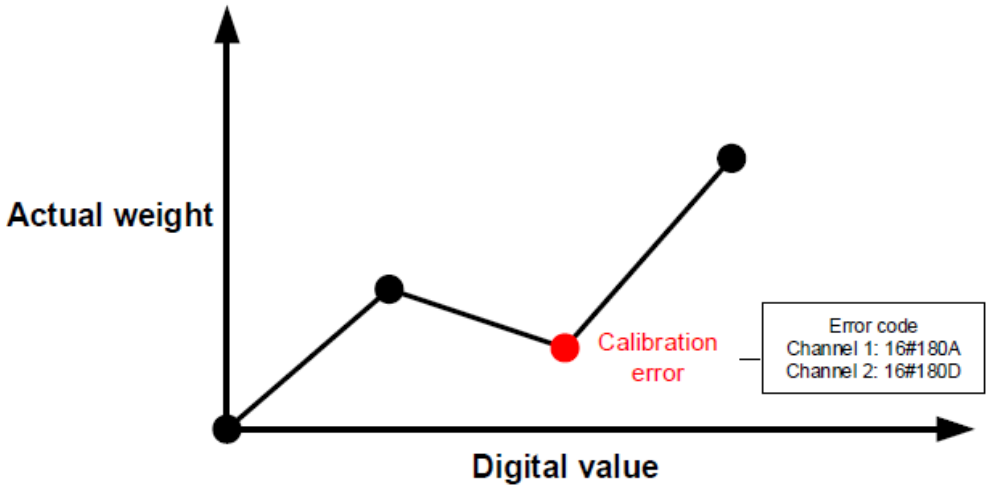
2. Fixed an issue that if the option “Automatics slave restart” is selected, it is possible that Ethernet connection may be lost and the connection to PLC cannot be established. Updated the procedures of turning on AX-308E CPU to eliminate the chance of NOT being able to turn AX-308E CPU on




## 2.3 UPDATE – AS02LC-A Firmware Update to Version 1.06

### Modified Functions

1. Fixed an issue that when the calibration curve goes into other direction, there is no error detected. The curve should go either upward or downward. If the calibration curve goes up and down as the image shown below, an error will occur




2. Fixed an issue that with no sensor connected to a channel, after closing and then opening the channel, there is no error detected. Error codes (16#1808 or 16#180B) should be generated, stating the signal received by channel 1 / 2 exceeds the range of analog inputs or the SEN voltage is abnormal

3. Fixed an issue that if executing the following commands on a closed channel, including calibration, tare weight measurement, net weight measurement, and clearing the weight to zero, after closing and then opening the channel, the values of the channel will be incorrect. Now you cannot execute the above-mentioned commands on a closed channel

4. Lower the sensitivity of reporting the driver board failure (16#1807)

## 2.4 UPDATE – DVP-ES3 Series CPU Firmware Update to Version 1.06.00

### Possible Issue and Solutions

| No. | API No. | Functions / Instructions | Description |
|---|---|---|---|
| 1 | 0709 | XCMP | When using the built-in X input points as trigger input points, it is possible that the software misjudges stable trigger signals as unstable and triggers no execution. |
| 2 | 2701 | DPLSR | Fixed an issue that for the even-numbered axes, when the number of pulses to output is set to 0, there is no limit on the number of pulses, but the pulse may stop output after a long time. The time the pulse stops outputting is different for different target output frequency. Some cases show that if the target output frequency is 200kHz, the pulse output may stop after around 12-hours of output. |
| 3 | 2704 | DZRN | Before: If the following conditions co-exist at the same time, the motion will not continue in the negative direction after reaching the positive limit.<br>- With zero return mode selected to leaving the zero point in the positive direction and then stops<br>- DOG point (contact A) is always ON due to a fault<br>- Reaching the positive limit |

| | | | |
|---|---|---|---|
| | | | After: If the above-mentioned conditions co-exist at the same time, the motion will continue in the negative direction after reaching the positive limit. |
| 4 | 2718 | TPO | - Fixed an issue that when in the single-axis point-to-point motion or 2-axis linear interpolation mode, if the same output axis and output parameters are set for the next segment, the actual output frequency will become abnormal<br>- Fixed an issue that when the 2-axis linear interpolation is in the absolute mode, if the output is resumed after a pause, the direction of the X axis will be inverted. |
| 5 | 2719<br>2720<br>2721 | DTPWS<br>DTPWL<br>DTPWC | Fixed an issue that when Symbol / Address is selected as the address mode of the position planning table, the target positions in the position planning table cannot be correctly modified via DTPWS / DTPWL / DTPWC. |
| 6 | 2809 | RSTD | Fixed an issue that when the RSTD instruction is frequently used, it is possible that the PLC stops for no reason and the error code 0x200A (invalid instruction) is recorded in the error log. |
| 7 | 2817<br>2818 | DTQC<br>DTQLC | Fixed an issue that other CAN positioning instructions cannot be executed after using online editing during the execution of DTQC or DTQLC instructions. |
| 8 | | COMMGR | Fixed an issue that if using the COMMGR Ethernet broadcast search function to search, the PLC name consisting of non-ASCII codes cannot be correctly displayed. (Should work with COMMGR V1.13.03 or later.) |
| 9 | | SFC | Fixed an issue that if the ACTIONs in the SFC program are written in the ST programming language, when the TMR instruction is used, the values in timer will not be cleared while STEPs are being converted. |

## New Instructions and Functions

| No. | API No. | Functions / Instructions | Description | Reference |
|---|---|---|---|---|
| 1 | | DVP02PU-E2 | New instructions for PU modules | Attachment B-1_PU Modules |
| 2 | 1016<br>1017<br>1018 | ATMR<br>ATMRH<br>ATMRM | Added new timer instructions, available for ISPSoft V3.16 or later | Attachment B-2_Timers |
| 3 | | Error Log | - Added relevant codes for CAN masters and slaves<br>- Added an RTC-related error code | Attachment B-3_ErroLog |
| 4 | | Status Log | Any RTC related changes will be added in the status log (Action No. 8). | |
| 5 | | HWCONFIG | - New 25 kbps baud rate option added for built-in CAN communication of AS-COPM card<br>- Added the position parameters for the execution of SFC STEP when PLC changes from STOP to RUN<br>Available for ISPSoft V3.16 or later | |

| | | | | |
|---|---|---|---|---|
| 6 | | MODBUS/TCP COM Port No. | Added following new SRs and SM flags.<br>- SR1092: COM port number of Modbus TCP slave; Default: 502 (when PLC acts as a slave)<br>- SR1093: COM port number of Modbus TCP data exchange table; Default: 502 (when PLC acts as a master<br>- SM1092: Error flag; Wrong COM port number of Modbus slave | Attachment B-6_PORT |

## Modified Instructions and Functions

| No. | API No. | Functions / Instructions | Descriptions | Reference |
|---|---|---|---|---|
| 1 | 1100<br>1101 | SFTR<br>SFTL | Increase the maximum length of the data to be shifted from 1024 to 4096. | |
| 2 | 1516 | FPOW | The base (source value) can be a negative number. When the base is a negative value, the exponent must be an integer number. | |
| 3 | 0600 | REF | - Supports refreshing the Ethernet communication<br>- Supports refreshing the PDO communication when acting as a slave | Attachment C-3_REF |
| 4 | 2712<br>2713<br>2714<br>2715 | DCICR<br>DCICA<br>DCICCR<br>DCICCA | Added options 20/21/22 in the function setting; users can use the basic angle of 10°, 5°, or 1° and the target coordinates X and Y to have a movement in a fixed 90° arc. | |
| 5 | 2720 | DTPWL | Supports using SM585 to modify the acceleration and deceleration time. | Attachment C-5_DTPWL |
| 6 | 2704<br>2724 | DZRN<br>DZRN2 | - Users can set alarm flags of positive and negative limits for the output axis; alarm flags appear, once positive and negative limits are reached during the search for the zero point | Attachment C-6_DZRN_DZRN2 |
| 7 | 1821 | DESO | Now Ethernet communication is supported. (COM1, COM2 and function card communications are already supported.) | |

## List of New PU Module Instructions

| API | Instruction Code | | Pulse Instruction | Function |
|---|---|---|---|---|
| | 16-Bit | 32-Bit | | |
| 1402 | | DPUCONF | ✓ | Setting output control parameters of PU module |
| 1403 | PUSTAT | | - | Reading PU module output state |
| 1404 | | DPUPLS | - | PU module pulse output (no acceleration) |
| 1405 | | DPUDRI | - | Relative position output of PU module (with acceleration and deceleration) |
| 1406 | | DPUDRA | - | Absolute addressing output of PU module |

| | | | | |
|---|---|---|---|---|
| | | | | (with acceleration and deceleration) |
| 1407 | | DPUZRN | - | PU module homing |
| 1408 | | DPUJOG | - | PU module jog output |
| 1409 | | DPUMPG | - | PU module MPG output |
| 1410 | | DPUCNT | - | High-speed counter function of PU module |
| 1411 | PUX | | ✓ | Setting PU module input point mode |
| 1412 | | DPULS | ✓ | Setting PU module software limits |

## B-1_PU Module

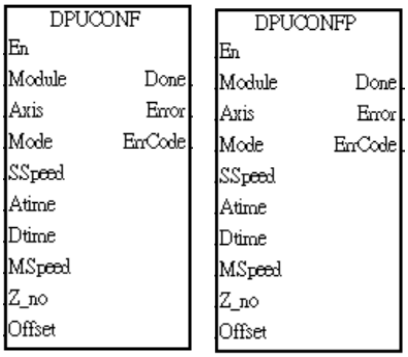
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1402 | D | PUCONF | P | **Module ~ Error, ErrCode** | Setting output control parameters of PU module |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| Axis | | | | | | | | ● | | | | | ○ | ○ | | |
| Mode | | | | | | | | ● | | | | | ○ | ○ | | |
| SSpeed | | | | | | | | ● | | | | | ○ | ○ | | |
| Atime | | | | | | | | ● | | | | | ○ | ○ | | |
| Dtime | | | | | | | | ● | | | | | ○ | ○ | | |
| MSpeed | | | | | | | | ● | | | | | ○ | ○ | | |
| Z_no | | | | | | | | ● | | | | | ○ | ○ | | |
| Offset | | | | | | | | ● | | | | | ○ | ○ | | |
| Done | | ● | ● | ● | | | | ● | | | | | | | | |
| Error | | ● | ● | ● | | | | ● | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Axis | | ● | | | | ● | | | | | | | |
| Mode | | ● | | | | ● | | | | | | | |
| SSpeed | | ● | | | | ● | | | | | | | |
| Atime | | ● | | | | ● | | | | | | | |
| Dtime | | ● | | | | ● | | | | | | | |
| Mspeed | | | ● | | | | ● | | | | | | |
| Z_no | | ● | | | | ● | | | | | | | |
| Offset | | ● | | | | ● | | | | | | | |
| Done | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| ES3 | — | ES3 |

## Symbol




**Module:** Module number
**Axis:** Output axis number
**Mode:** Output mode
**SSpeed:** Speed for starting / ending frequency
**Atime:** Acceleration time
**Dtime:** Deceleration time
**MSpeed:** Maximum output frequency
**Z_no:** Number of Z phases to look for after returning to the original point
**Offset:** Specify the number of outputs after returning to the original point
**Done:** Completion flag
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later. For ISPSoft, we recommend using software version 3.16 and above. The timing to set this instruction is when En changes from OFF to ON

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag **Error** will change to ON

3. **Module** sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and the module numbered as 2 is 02PU-E2

4. **Axis** sets the output axis number for the specified PU module. The setting values 1~2 represent the axis1~axis2 output of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag **Error** will change to ON
   See the following combination of axes numbers and corresponding output points of PU modules

| PU Module Name | Axis 1 | Axis 2 |
|---|---|---|
| DVP02PU-E2 | Y0 / Y1 | Y2 / Y3 |

5. **Mode** sets the output mode of an output axis and the setting values are explained in the following table

| Output Mode Value | Description | Remark |
|---|---|---|
| 0 | Single-point pulse output (An even-number point for output only) | Y0 or Y2 for output |
| 1 | Pulse (An even-number point) + direction (An odd-number point) | Y0 is for the pulse and Y1 is for the direction. Y1: ON, negative direction; Y1: OFF, positive direction |
| 2 | CW (An even-number point) + CCW (An odd-number point) | Y0 is for CW (positive direction) and Y1 is for CCW (negative direction) |
| 3 | Phase A (An even-number point) + Phase B (An odd-number point) | Y0 is for phase A and Y1 is for phase B. When phase A is leading phase B: positive direction; when phase B is leading phase A: negative direction |
| Others | Automatically switch to mode 1 (default value) | |

6. SSpeed~ Offset

See the explanation of the following non-latched parameters and setting values. If the setting values exceed the range, the instruction will automatically be executed at the minimum or maximum value

| Parameter | Function | Range | Default | Remark |
|---|---|---|---|---|
| SSpeed | Starting/ending frequency | 0 ~ 10,000 [Hz] | 100 | |
| Atime | Acceleration time | 0 ~ 10,000 [ms] | 100 | |
| Dtime | Deceleration time | 0 ~ 10,000 [ms] | 100 | |
| MSpeed | Maximum output frequency | 100 ~ 200,000 [Hz] | 100,000 | A 32-bit value |
| Z_no | Number of Z phase signals to seek after returning to origin | -100 ~ +100 | 0 | 0: disabled |
| Offset | Outputs the offset position after the homing is finished and Z phase seeking is done. | -10,000 ~ +10,000 [pulses] | 0 | 0: disabled |

7. **Done**, an output of the specified PU module has been set as the completion flag. When **Done** is On, it indicates that the parameter setting is successful. You can continue to perform positioning output based on the On state of the completion flag. The clearing of the **Done** flag need be conducted by manual. The **Done** flag changes to ON only when the setting is completed

8. **Error**, an output of the specified PU module is a parameter error flag. Most parameter ranges are filtered automatically by the PLC. Thus if the error flag is ON, it means that there is no specified PU module or the PU module number is wrong or the output axis number is incorrect

9. The instruction is a pulse instruction. Even if the A contact is adopted as the condition contact, PU module parameters are also set only when the instruction is started. Therefore, if a parameter value is to be updated, restart the instruction to make the parameter set again

10. Since the set parameters are delivered through the module communication command, confirm the state of the output **Done** or **Error** before a parameter value is modified and then proceed with relevant operations

11. **ErrCode** shows error codes. See the description as follows

| Error Code | Description |
|---|---|
| 16#1400 | The module does not support the function. |
| 16#1402 | There is no response from the module; communication timeout occurs. |

## Programming Example

Refer to the description of DPUDRI instruction (API1405) for more information.

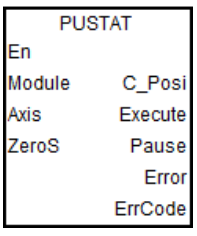
| API | Instruction code | Operand | Function |
|---|---|---|---|
| 1403 | PUSTAT | Module ~ ErrCode | Reading PU module output state |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| Axis | | | | | | | | ● | | | | | ○ | ○ | | |
| ZeroS | ● | ● | ● | ● | | | | ● | | | | | | | | |
| C_Posi | | | | | | | | ● | | | | | | | | |
| Execute | | ● | ● | ● | | | | ● | | | | | | | | |
| Pause | | ● | ● | ● | | | | ● | | | | | | | | |
| Error | | ● | ● | ● | | | | ● | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Axis | | ● | | | | ● | | | | | | | |
| ZeroS | ● | | | | | | | | | | | | |
| C_Posi | | | ● | | | | ● | | | | | | |
| Execute | ● | | | | | | | | | | | | |
| Pause | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| – | ES3 | – |

## Symbol




**Module:** Module number
**Axis:** Output axis number
**ZeroS:** Clear present output position to 0
**C_Posi:** Current output position
**Execute:** Execution flag
**Pause:** Pause flag
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later. For ISPSoft, we recommend using software version 3.16 and above. The En setting must be set to ON so as to update the status of the specified axis continuously

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag **Error** will change to ON

3. **Module** sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and the module numbered as 2 is 02PU-E2

4. **Axis** sets the output axis number for the specified PU module. The setting values 1~2 represent the axis1~axis2 output of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag **Error** will change to ON

5. **ZeroS** clears the present output position to 0. If the present axis position is to be cleared to 0, set **ZeroS** from OFF to ON when the instruction is started

6. **C_Posi** sets the present position of the output axis for the specified PU module. The parameter value is a latched value and stored in the PU module

7. **Execute** is an read-only flag which means the output axis of the specified PU module is outputting or not. When **Execute** is On, it means the output is being conducted. When **Execute** is Off, it means the output axis is unused and can accept the next output command

8. **Pause** is an read-only flag to control the output axis of the specified PU module to pause its output. When **Pause** is On, it means the output is paused, the present velocity is 0 and the present output has not reached the specified target output position. If you restore the output, the flag will be cleared automatically.
Note: While **Pause** is On, **Execute** is constantly On as well

9. **Error** is an read-only error flag which means an error occurs during the reading of the specified PU module. Refer to the explanation of error codes in **ErrCode**

10. After the PUSTAT instruction gives the pause command, the flags **Execute, Pause and Error** become read-only flags and at the moment, their states cannot be modified. The **Execute, Pause and Error** flags can be set or cleared only when the PUSTAT instruction is turned off

11. For PU module state, check out the data exchange function of the special extension module through SM228. Refer to Section 2.2.16 Additional Remarks on Special Auxiliary Relays and Special Data Registers in the DVP-ES3 Series Programming Manual for details

12. **ErrCode** shows error codes and the explanations are seen in the following table.

| Error Code | Description |
|---|---|
| 16#1400 | The module does not support the function. |
| 16#1402 | There is no response from the module; communication timeout occurs. |
| 16#1403 | There is no such output axis number in the PU module. |

## Programming Example

Refer to the description of DPUDRI (API1405) for more information.

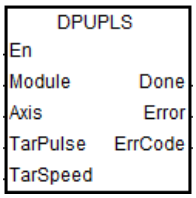
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1404 | D | PUPLS | | Module ~ ErrCode | PU module pulse output (no acceleration) |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| Axis | | | | | | | | ● | | | | | ○ | ○ | | |
| TarPulse | | | | | | | | ● | | | | | ○ | ○ | | |
| TarSpeed | | | | | | | | ● | | | | | ○ | ○ | | |
| Done | | ● | ● | ● | | | | ● | | | | | | | | |
| Error | | ● | ● | ● | | | | ● | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Axis | | ● | | | | ● | | | | | | | |
| TarPulse | | | ● | | | | ● | | | | | | |
| TarSpeed | | | ● | | | | ● | | | | | | |
| Done | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| — | — | ES3 |

## Symbol




**Module:** Module number
**Axis:** Output axis number
**TarPulse:** Target number of output pulses
**TarSpeed:** Target output frequency
**Done:** Done flag
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later. For ISPSoft, we recommend using software version 3.16 and above. When En setting is set to ON, this instruction would be effective. Once the setting changes to OFF, Output pulse would be terminated immediately

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag **Error** will change to ON

3. **Module** sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and the module numbered as 2 is 02PU-E2

4. **Axis** sets the output axis number for the specified PU module. The setting values 1~2 represent the axis1~axis2 output of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag **Error** will change to ON

5. **TarPulse** sets the number of output pulses. The pulse number is a positive signed 32-bit value. When the value is 0, it means the output is always being performed, the number of output pulses is not limited and the output is not stopped until the instruction is disabled. When the value is less than 0, the PLC automatically uses 2's complement to transform the value into a positive integer as the number of output pulses

6. **TarSpeed** sets the target output speed (Unit: Hz). The input value is a signed 32-bit value within the range of -200,000 (-200K) ~ 200,000 (200K). You can modify the target frequency any time after the instruction is enabled and the PU module will automatically switch to the newly set target frequency after outputting a full pulse

7. When **TarSpeed** is a positive number (>0), it means that the “positive direction” output point is Off. When **TarSpeed** is a negative number (<0), it means that the “negative direction” output point is On. When **TarSpeed** is 0, it means that the output will be paused after the being executed pulse is output fully

8. The instruction does not support the function of acceleration and deceleration. Use the DPUDRI instruction instead if you need the function of acceleration and deceleration

9. The instruction can be used for the speed change. While the instruction is being executed, you can change the value of **TarSpeed** so as to change the output speed. When the setting value exceeds the maximum frequency, the instruction would be executed at the maximum frequency. But changing the speed would not change the direction. If the direction is to be changed, set the value of **TarSpeed** to 0 first and then modify the target speed

10. When the outputs have reached the pulse number specified by **TarPulse**, the **Done** flag changes to ON. The **Done** flag need be cleared by manual. The instruction sets the completion flag to ON only when the output is completed

11. The instruction can be used with the software and hardware limit points. When the limits are triggered, the output stops immediately and the **Error** flag changes to ON

12. If any error occurs as the instruction is in process of the output, the **Error** flag changes to ON. Refer to the error codes **ErrCode** shows for the trouble shooting

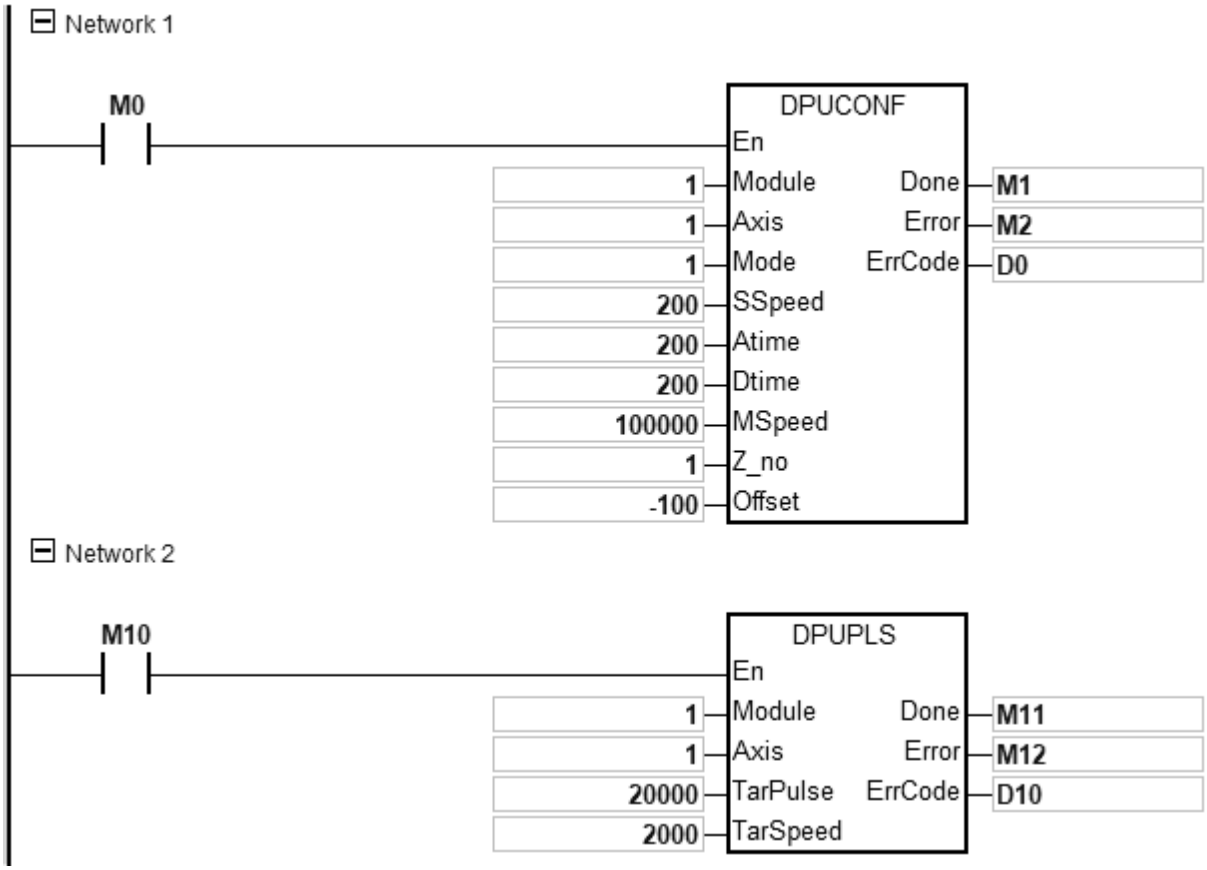
The error codes that **ErrCode** shows are listed in the following table.

| Error Code | Description |
|---|---|
| 16#1400 | The module does not support the function. |
| 16#1402 | There is no response from the module; communication timeout occurs. |
| 16#1403 | There is no such output axis number in the PU module. |
| 16#1405 | The output axis specified by the PU module is outputting data. It is not allowed to specify the output repeatedly. |
| 16#1406 | PU module stops Output pulse when the positive limit is reached. |
| 16#1407 | PU module stops Output pulse when the negative limit is reached. |

## Programming Example:

1. When M0 is ON, the DPUCONF instruction for axis 1 is executed to modify the parameters by setting **Mode** to 1 (Pulse Y0 + direction Y1), **SSpeed** to 200Hz, **Atime** to 200ms, **Dtime** to 200ms and **MSpeed** to 100kHz. After the output of **Done** is completed, M1 is ON

2. 2. When M10 is ON, the DPUPLS instruction for axis 1 starts to output 20,000 pulses from Y0 at the frequency of 2KHz (without acceleration and deceleration). Y1 is OFF, which indicates the positive direction and M11 is ON after the pulse output is finished


Network 1
M0
DPUCONF
En
1 Module
1 Axis
1 Mode
200 SSpeed
200 Atime
200 Dtime
100000 MSpeed
1 Z_no
-100 Offset
Done M1
Error M2
ErrCode D0
Network 2
M10
DPUPLS
En
1 Module
1 Axis
20000 TarPulse
2000 TarSpeed
Done M11
Error M12
ErrCode D10

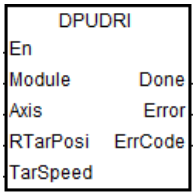
| PI | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1405 | D | PUDRI | | Module ~ ErrCode | Relative position output of PU module (with acceleration and deceleration) |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| Axis | | | | | | | | ● | | | | | ○ | ○ | | |
| RTarPosi | | | | | | | | ● | | | | | ○ | ○ | | |
| TarSpeed | | | | | | | | ● | | | | | ○ | ○ | | |
| Done | | ● | ● | ● | | | | ● | | | | | | | | |
| Error | | ● | ● | ● | | | | ● | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Axis | | ● | | | | ● | | | | | | | |
| RTarPosi | | | ● | | | | ● | | | | | | |
| TarSpeed | | | ● | | | | ● | | | | | | |
| Done | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| — | — | ES3 |

## Symbol




**Module:** Module number
**Axis:** Output axis number
**RTarPosi:** Number of output pulses for relative positioning
**TarSpeed:** Target output frequency
**Done:** Completion flag
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later. For ISPSoft, we recommend using software version 3.16 and above. When En setting is set to ON, this instruction would be effective. Once the setting changes to OFF, Output pulse would decelerate until it stops

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag **Error** will change to ON

3. **Module** sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is

numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and the module numbered as 2 is 02PU-E2

4. **Axis** sets the output axis number for the specified PU module. The setting values 1~2 represent the axis1~axis2 output of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag **Error** will change to ON

5. **RTarPosi** sets the position for relative positioning. The pulse number is a signed 32-bit value. When the value is greater than 0, the output will go in the positive direction (and the direction output point is off). When the value is less than 0, the output will go in the negative direction (and the direction output point is on). When the value is 0, the output completion flag **Done** changes to ON

6. **TarSpeed** sets the target output frequency (Unit: Hz). The frequency value is a positive signed 32-bit integer. When the value is less than 0, the instruction will automatically use 2's complement to transform the value into a positive integer. When the value is 0, the instruction will notify the module to enter the pause mode. The actual output is decelerated at the deceleration rate till the output speed is equal to 0 and the pause flag changes to ON. Refer to PUSTAT instruction for more details. See the setting range of **TarSpeed** for the module in the following table

| Module Name | **TarSpeed Setting Range** |
|---|---|
| DVP02PU-E2 | -200,000 ( -200K ) ~ 200,000 ( 200K ) |

7. After the output is started, the target frequency is allowed to change any time. In the actual frequency change, the PLC will automatically change the frequency based on the set acceleration and deceleration rate in the DPUCONF instruction. When the modified speed exceeds the allowed maximum frequency, the output will be performed at the maximum frequency

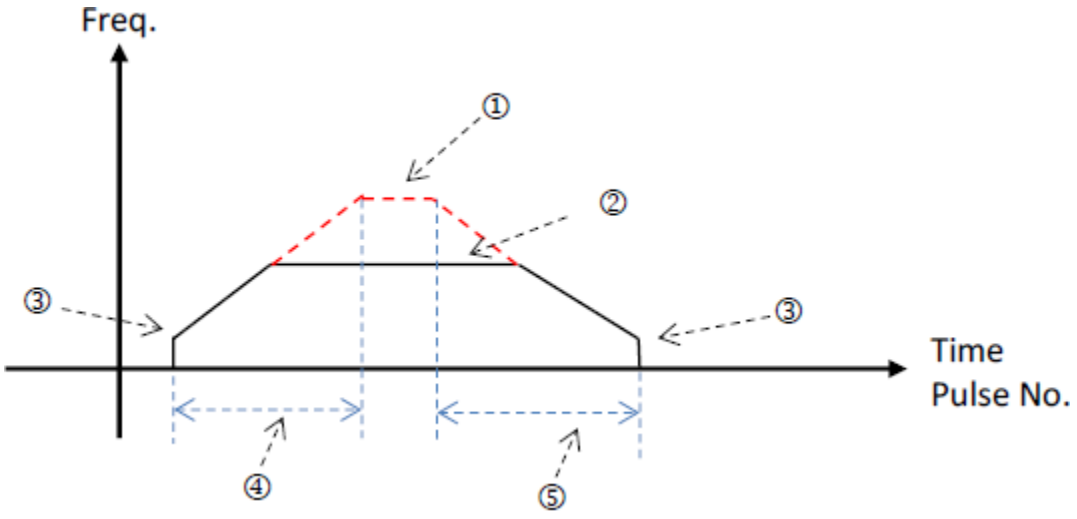
8. When the outputs have reached the pulse number for relative positioning specified by **RTarPosi**, the **Done** flag changes to ON. The **Done** flag need be cleared by manual. The instruction sets the completion flag to ON only when the output is completed

9. The instruction can be used with the software and hardware limit points. The output stops immediately and the **Error** flag changes to ON when the limits are triggered

10. If any error occurs as the instruction is in process of the output, the **Error** flag changes to ON. Refer to the error codes that **ErrCode** shows for the trouble shooting

11. The error codes that **ErrCode** shows

12. Illustration of the acceleration and deceleration curve of the DPUDRI instruction




① : Maximum output frequency value. Refer to the setting in the DPUCONF instruction for the parameter setting. Alternatively, set the parameter value through HWCONFIG.

②: The target frequency specified by the PU module output instruction. The target frequency output must not exceed the maximum output frequency. If the maximum output frequency is exceeded, the maximum output frequency is regarded as the output frequency.

③: Starting/ending output frequency value. Refer to the setting in the DPUCONF instruction for the parameter setting. Alternatively, set the parameter value through HWCONFIG.

④: The acceleration time value. Refer to the setting in the DPUCONF instruction for the parameter setting. Alternatively, set the parameter value through HWCONFIG.

⑤: The deceleration time value. Refer to the setting in the DPUCONF instruction for the parameter setting. Alternatively, set the parameter value through HWCONFIG.

The acceleration and deceleration that the PU module controls is performed according to the fixed slope. So the actual acceleration time and deceleration time change based on the output target frequency. The formula for calculation of acceleration rate and deceleration rate are respectively shown as follows.
(Max. output frequency - starting frequency)/acceleration time;
(Max. output frequency - ending frequency)/deceleration time.

## Programming Example:

1. When M0 is ON, the PUX instruction is executed to set the input parameters of DVP02PU-E2 module with “Axis 1, DOG, Rising-edge triggered” for input point X0, “Axis 1, Z phase, Rising-edge triggered” for input point X1 and “15m” for X0 / X1 filter time. When the output of **Done** is completed, M1/M3 changes to ON

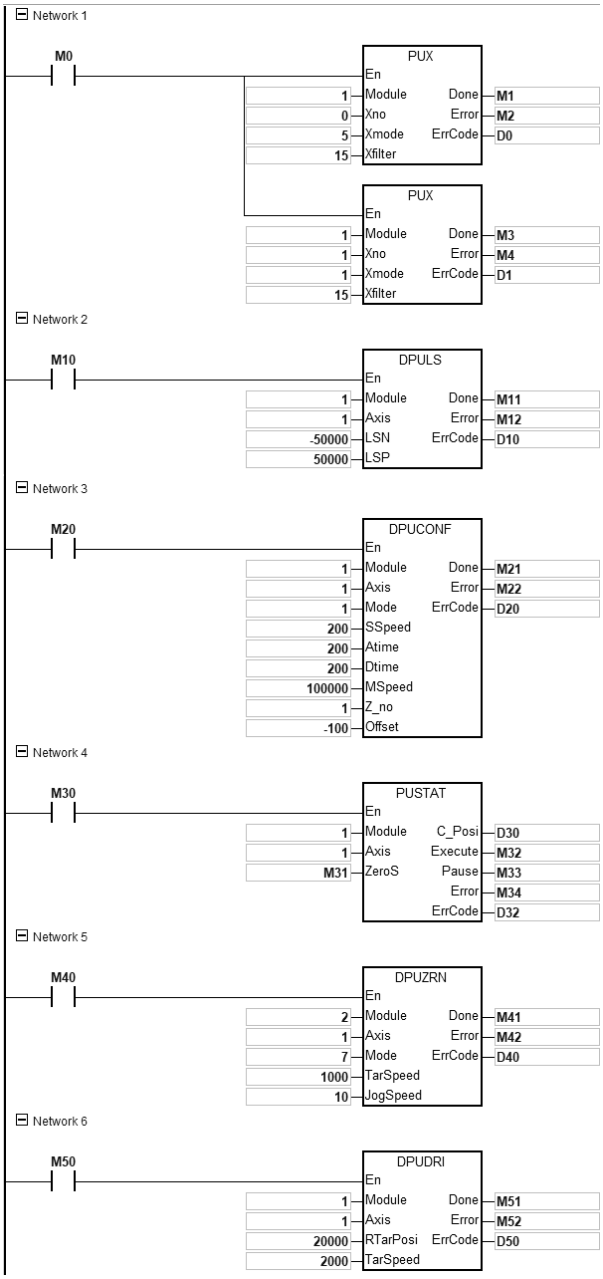
2. When M10 is ON, the DPULS instruction for axis 1 is executed to set the software limit points with “-50000” for LSN and “50000” for LSP. When the output of **Done** is completed, M11 changes to ON

3. When M20 is ON, the DPUCONF instruction for axis 1 is executed to change the parameters by setting **Mode** to 1 (Pulse Y0+ Direction Y1), **SSpeed** (start speed) to 200Hz, **Atime** (acceleration time) to 200ms, **Dtime** (deceleration time) to 200ms, **MSpeed** (maximum speed) to 100kHz, Z_NO (Number of Z phases to look for after returning to the home position) to 1 and **Offset** (number of outputs after homing is finished) to -100. And M21 changes to ON as the output of Done is completed

4. When M30=ON, the PUSTAT instruction for axis 1 is executed to read the PU module output state. To clear current output position, you can set M31 to ON so that the current position of axis 1 (rising-edge triggered) in D30 would be cleared to 0

5. When M40 is ON, the DPUZRN instruction for axis1 starts to perform homing and the PUSTAT instruction displays the current position in D30. The output point Y0 outputs pulses at the frequency of 1kHz and the search for the home starts in positive direction. Once the near home signal (DOG) is reached and X0 is ON, the axis starts to decelerate and then moves at the **Jogspeed** of 100Hz in the negative direction. When X0=OFF, the axis moves in the positive direction to search for Z phase until the first rising-edge triggered signal at X1 (Z phase) is detected, then it moves toward negative direction after 100 output pulses are completed. Finally, M41 changes to ON after the output of **Done** is finished

6. When M50 is ON, the DPUDRI instruction for axis 1 starts to perform relative positioning output. The PUSTAT instruction displays the current position in D0 and the output point Y0 outputs 20,000 pulses at the frequency of 2kHz (relative addressing). Y1 is OFF, which indicates that the direction is positive and the PUSTAT instruction displays the current position in D30. Finally, M51 changes to ON after the output of **Done** is completed.


Network 1
M0
PUX
En
1 Module
0 Xno
5 Xmode
15 Xfilter
Done M1
Error M2
ErrCode D0
PUX
En
1 Module
1 Xno
1 Xmode
15 Xfilter
Done M3
Error M4
ErrCode D1
Network 2
M10
DPULS
En
1 Module
1 Axis
-50000 LSN
50000 LSP
Done M11
Error M12
ErrCode D10
Network 3
M20
DPUCONF
En
1 Module
1 Axis
1 Mode
200 SSpeed
200 Atime
200 Dtime
100000 MSpeed
1 Z_no
-100 Offset
Done M21
Error M22
ErrCode D20
Network 4
M30
PUSTAT
En
1 Module
1 Axis
M31 ZeroS
C_Posi D30
Execute M32
Pause M33
Error M34
ErrCode D32
Network 5
M40
DPUZRN
En
2 Module
1 Axis
7 Mode
1000 TarSpeed
10 JogSpeed
Done M41
Error M42
ErrCode D40
Network 6
M50
DPUDRI
En
1 Module
1 Axis
20000 RTarPosi
2000 TarSpeed
Done M51
Error M52
ErrCode D50

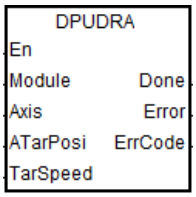
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1406 | D | PUDRA | | Module ~ ErrCode | Absolute addressing output of PU module (with acceleration and deceleration) |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| Axis | | | | | | | | ● | | | | | ○ | ○ | | |
| ATarPosi | | | | | | | | ● | | | | | ○ | ○ | | |
| TarSpeed | | | | | | | | ● | | | | | ○ | ○ | | |
| Done | | ● | ● | ● | | | | | | | | | | | | |
| Error | | ● | ● | ● | | | | | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Axis | | ● | | | | ● | | | | | | | |
| ATarPosi | | | ● | | | | ● | | | | | | |
| TarSpeed | | | ● | | | | ● | | | | | | |
| Done | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| – | – | ES3 |

## Symbol




**Module:** Module number
**Axis:** Output axis number
**ATarPosi:** Number of output pulses for absolute positioning
**TarSpeed:** Target output frequency
**Done:** Completion/pause flag
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later. For ISPSoft, we recommend using software version 3.16 and above. When En setting is set to ON, this instruction would be effective. Once the setting changes to OFF, Output pulse would decelerate until it stops

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag **Error** will change to ON

3. **Module** sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and

the module numbered as 2 is 02PU-E2

4. **Axis** sets the output axis number for the specified PU module. The setting values 1~2 represent the axis1~axis2 output of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag **Error** will change to ON

5. **RTarPosi** sets the position for relative positioning. The pulse number is a signed 32-bit value. When the value is greater than 0, the output will go in the positive direction (and the direction output point is off). When the value is less than 0, the output will go in the negative direction (and the direction output point is on). When the value is 0, the output completion flag **Done** changes to ON

6. **TarSpeed** sets the target output frequency (Unit: Hz). The frequency value is a positive signed 32-bit integer. When the value is less than 0, the instruction will automatically use 2's complement to transform the value into a positive integer. When the value is 0, the instruction will notify the module to enter the pause mode. The actual output is decelerated at the deceleration rate till the output speed is equal to 0 and the pause flag changes to ON. Refer to PUSTAT instruction for more details.
   See the setting range of **TarSpeed** for the module in the following table

| Module Name | TarSpeed Setting Range |
|---|---|
| DVP02PU-E2 | -200,000 ( -200K ) ~ 200,000 ( 200K ) |

7. After the output is started, the target frequency is allowed to change any time. In the actual frequency change, the PLC will automatically change the frequency based on the set acceleration and deceleration rate in the DPUCONF instruction. When the modified speed exceeds the allowed maximum frequency, the output will be performed at the maximum frequency

8. When the outputs have reached the pulse number for relative positioning specified by **RTarPosi**, the **Done** flag changes to ON. The **Done** flag need be cleared by manual. The instruction sets the completion flag to ON only when the output is completed

9. The instruction can be used with the software and hardware limit points. The output stops immediately and the **Error** flag changes to ON when the limits are triggered

10. If any error occurs as the instruction is in process of the output, the **Error** flag changes to ON. Refer to the error codes that **ErrCode** shows for the trouble shooting

11. The error codes that **ErrCode** shows are listed in the following table

| Error Code | Description |
|---|---|
| 16#1400 | The module does not support the function. |
| 16#1402 | There is no response from the module; communication timeout occurs. |
| 16#1403 | There is no such output axis number in the PU module. |
| 16#1405 | The output axis specified by the PU module is outputting data. It is not allowed to specify the output repeatedly. |
| 16#1406 | PU module stops Output pulse when the positive limit is reached. |
| 16#1407 | PU module stops Output pulse when the negative limit is reached. |

12. Illustration of the acceleration and deceleration curve of the DPUDRI instruction

① : Maximum output frequency value. Refer to the setting in the DPUCONF instruction for the parameter setting. Alternatively, set the parameter value through HWCONFIG.

②: The target frequency specified by the PU module output instruction. The target frequency output must not exceed the maximum output frequency. If the maximum output frequency is exceeded, the maximum output frequency is regarded as the output frequency.

③: Starting/ending output frequency value. Refer to the setting in the DPUCONF instruction for the parameter setting. Alternatively, set the parameter value through HWCONFIG

④: The acceleration time value. Refer to the setting in the DPUCONF instruction for the parameter setting.

Alternatively, set the parameter value through HWCONFIG

⑤: The deceleration time value. Refer to the setting in the DPUCONF instruction for the parameter setting.
Alternatively, set the parameter value through HWCONFIG
The acceleration and deceleration that the PU module controls is performed according to the fixed slope. So the actual acceleration time and deceleration time change based on the output target frequency. The formula for calculation of acceleration rate and deceleration rate are respectively shown as follows.
(Max. output frequency - starting frequency)/acceleration time;
(Max. output frequency - ending frequency)/deceleration time.

## Programming Example:

1. When M0 is ON, the PUX instruction is executed to set the input parameters of DVP02PU-E2 module with “Axis 1, DOG, Rising-edge triggered” for input point X0, “Axis 1, Z phase, Rising-edge triggered” for input point X1 and “15m” for X0 / X1 filter time. When the output of **Done** is completed, M1/M3 changes to ON

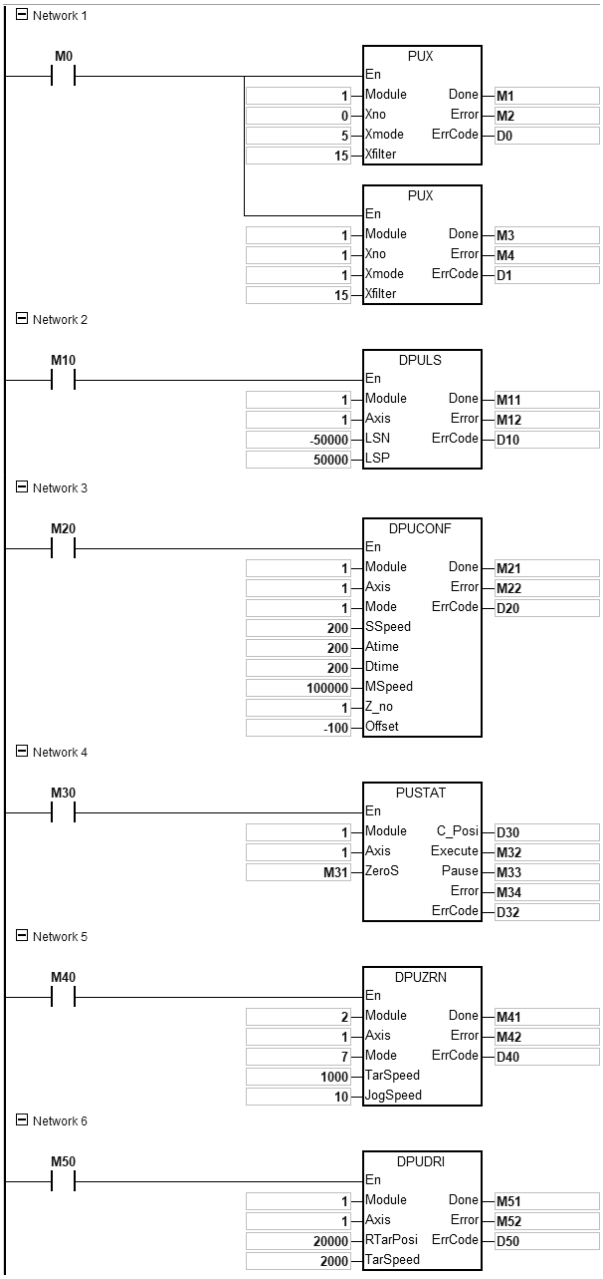
2. When M10 is ON, the DPULS instruction for axis 1 is executed to set the software limit points with “-50000” for LSN and “50000” for LSP. When the output of **Done** is completed, M11 changes to ON

3. When M20 is ON, the DPUCONF instruction for axis 1 is executed to change the parameters by setting **Mode** to 1 (Pulse Y0+ Direction Y1), **SSpeed** (start speed) to 200Hz, **Atime** (acceleration time) to 200ms, **Dtime** (deceleration time) to 200ms, **MSpeed** (maximum speed) to 100kHz, Z_NO (Number of Z phases to look for after returning to the home position) to 1 and **Offset** (number of outputs after homing is finished) to -100. And M21 changes to ON as the output of Done is completed

4. When M30=ON, the PUSTAT instruction for axis 1 is executed to read the PU module output state. To clear current output position, you can set M31 to ON so that the current position of axis 1 (rising-edge triggered) in D30 would be cleared to 0

5. When M40 is ON, the DPUZRN instruction for axis1 starts to perform homing and the PUSTAT instruction displays the current position in D30. The output point Y0 outputs pulses at the frequency of 1kHz and the search for the home starts in positive direction. Once the near home signal (DOG) is reached and X0 is ON, the axis starts to decelerate and then moves at the **Jogspeed** of 100Hz in the negative direction. When X0=OFF, the axis moves in the positive direction to search for Z phase until the first rising-edge triggered signal at X1 (Z phase) is detected, then it moves toward negative direction after 100 output pulses are completed. Finally, M41 changes to ON after the output of **Done** is finished

6. When M50 is ON, the DPUDRI instruction for axis 1 starts to perform relative positioning output. The PUSTAT instruction displays the current position in D0 and the output point Y0 outputs 20,000 pulses at the frequency of 2kHz (relative addressing). Y1 is OFF, which indicates that the direction is positive and the PUSTAT instruction displays the current position in D30. Finally, M51 changes to ON after the output of **Done** is completed.


Network 1
M0
PUX
En
1 Module
0 Xno
5 Xmode
15 Xfilter
Done M1
Error M2
ErrCode D0
PUX
En
1 Module
1 Xno
1 Xmode
15 Xfilter
Done M3
Error M4
ErrCode D1
Network 2
M10
DPULS
En
1 Module
1 Axis
-50000 LSN
50000 LSP
Done M11
Error M12
ErrCode D10
Network 3
M20
DPUCONF
En
1 Module
1 Axis
1 Mode
200 SSpeed
200 Atime
200 Dtime
100000 MSpeed
1 Z_no
-100 Offset
Done M21
Error M22
ErrCode D20
Network 4
M30
PUSTAT
En
1 Module
1 Axis
M31 ZeroS
C_Posi D30
Execute M32
Pause M33
Error M34
ErrCode D32
Network 5
M40
DPUZRN
En
2 Module
1 Axis
7 Mode
1000 TarSpeed
10 JogSpeed
Done M41
Error M42
ErrCode D40
Network 6
M50
DPUDRI
En
1 Module
1 Axis
20000 RTarPosi
2000 TarSpeed
Done M51
Error M52
ErrCode D50

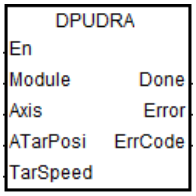
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1406 | D | PUDRA | | Module ~ ErrCode | Absolute addressing output of PU module (with acceleration and deceleration) |

| Device | X | Y | M | S | T | C | HC | D | FR | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | ○ | ○ | | |
| Axis | | | | | | | | ● | | | | ○ | ○ | | |
| ATarPosi | | | | | | | | ● | | | | ○ | ○ | | |
| TarSpeed | | | | | | | | ● | | | | ○ | ○ | | |
| Done | | ● | ● | ● | | | | | | | | | | | |
| Error | | ● | ● | ● | | | | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Axis | | ● | | | | ● | | | | | | | |
| ATarPosi | | | ● | | | | ● | | | | | | |
| TarSpeed | | | ● | | | | ● | | | | | | |
| Done | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| – | – | ES3 |

## Symbol




**Module:** Module number
**Axis:** Output axis number
**ATarPosi:** Number of output pulses for absolute positioning
**TarSpeed:** Target output frequency
**Done:** Completion/pause flag
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later. For ISPSoft, we recommend using software version 3.16 and above. When En setting is set to ON, this instruction would be effective. Once the setting changes to OFF, Output pulse would decelerate until it stops

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag **Error** will change to ON

3. **Module** sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and

the module numbered as 2 is 02PU-E2

4. **Axis** sets the output axis number for the specified PU module. The setting values 1~2 represent the axis1~axis2 output of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag **Error** will change to ON

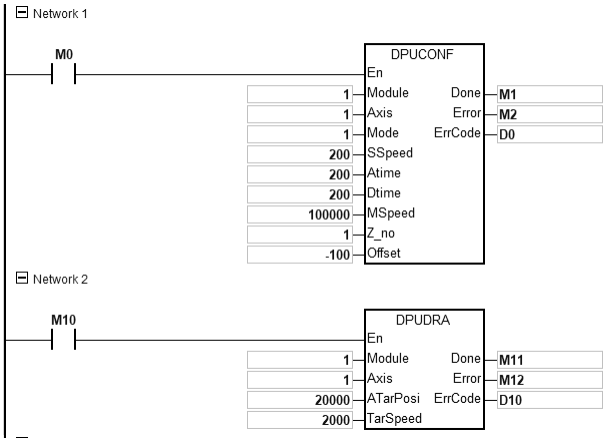
5. **ATarPosi** is the position for absolute addressing. The input pulse number is a signed 32 bit value. The PU module will automatically compare it with the present position. If the comparison result is greater than 0, the output will be conducted in the positive direction (and the direction output point is off). If the comparison result is less than 0, the output will be conducted in the negative direction and the direction output point is on). When the value is 0, the instruction sets the **Done** flag to ON

6. Refer to the DPUDRI instruction for the explanation of other parameters.

## Programming Example:

1. When M0 is ON, the DPUCONF instruction for axis 1 is executed to modify the parameters by setting **Mode** to 1 (Pulse Y0+ Direction Y1), **SSpeed** (the speed for starting) to 200Hz, **Atime** (acceleration time) to 200ms, **Dtime** (deceleration time) to 200ms and **MSpeed** (maximum output frequency) to 100kHz. And M1 changes to ON as the output of **Done** is completed

2. When M10 is ON, the DPUDRA instruction for axis 1 is executed to output pulses from Y0 at the frequency of 2kHz until the current position reaches 20,000 (absolute addressing). When Y1 is OFF, the direction is positive. And M11 changes to ON as the output of **Done** is completed



| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1407 | D | PUZRN | | Module ~ ErrCode | PU module homing |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| Axis | | | | | | | | ● | | | | | ○ | ○ | | |
| Mode | | | | | | | | ● | | | | | ○ | ○ | | |
| TarSpeed | | | | | | | | ● | | | | | ○ | ○ | | |
| JogSpeed | | | | | | | | ● | | | | | ○ | ○ | | |
| Done | | ● | ● | ● | | | | | | | | | | | | |
| Error | | ● | ● | ● | | | | | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Axis | | ● | | | | ● | | | | | | | |
| Mode | | ● | | | | ● | | | | | | | |
| TarSpeed | | | ● | | | | ● | | | | | | |
| JogSpeed | | ● | | | | ● | | | | | | | |
| Done | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| – | ES3 | ES3 |

## Symbol


**Module:** Module number
**Axis:** Output axis number
**Mode:** Homing mode selection
**TarSpeed:** Maximum output frequency for the homing
**JogSpeed:** The jog frequency for the homing
**Done:** Completion flag
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later. For ISPSoft, we recommend using software version 3.16 and above. When En setting is set to ON, this instruction would be effective. Once the setting changes to OFF, Output pulse would decelerate until it stops

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag **Error** will change to ON

3. **Module** sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is

numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and the module numbered as 2 is 02PU-E2

4. **Axis** sets the output axis number for the specified PU module. The setting values 1~2 represent the axis1~axis2 output of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag **Error** will change to ON

5. **Mode** sets a homing mode. The explanation of modes is shown in the following table

| Mode Value | Function | Use matching input points by PUX instruction | Remark |
|---|---|---|---|
| 0 | Directly clear the current position to 0. | **None** | |
| 1 | The axis starts to go toward the negative direction and then stops after leaving the DOG point position | **DOG** | |
| 2 | The axis starts to go toward the positive direction and then stops after leaving the DOG point position | **DOG** | |
| 3 | After Mode 1 is finished, seek the set number of Z phases. | DOG and Z phase input | Use DPUCONF instruction to set up the number of Z phases. |
| 4 | After Mode 2 is finished, seek the set number of Z phases. | DOG and Z phase input | |
| 5 | After Mode 1 is finished, output the offset position. | DOG | Use DPUCONF instruction to set up the offset position. |
| 6 | After Mode 2 is finished, output the offset position. | DOG | |
| 7 | After Mode 1 is finished, seek the set number of Z phases and then output the offset position. | DOG and Z phase input | Use DPUCONF instruction to set up the number of Z phases and offset position. |
| 8 | After Mode 2 is finished, seek the set number of Z phases and then output the offset position. | DOG and Z phase input | |
| 255 | Modify the current output position for the axis. | None | Use the setting value of **TarSpeed** |
| Other | Reserved | | |

6. **TarSpeed** sets the maximum output frequency for the homing. The setting value is a signed 32-bit value. When **Mode** value is between 1~8, the range of the setting value is -200,000 ~ -100 (Hz) and 100 ~ 200,000 (Hz). If **Mode** value is 255, **TarSpeed** value will become the present output position value of the PU module

7. **JogSpeed** is the jog frequency for reaching the home position and also represents the start/end frequency for homing. The setting value is a signed 16-bit value within the range of 1~10,000 (Hz)

8. When the specified home position is reached during the instruction is executed, the **Done** flag changes to ON. The **Done** flag need be cleared by manual. The instruction sets the completion flag to ON only when the output is completed

9. The instruction does not support software limit points. It can be used with hardware limit points only. When a hardware limit point is triggered during the output, the **Error** flag will be set to ON.
The following cases and corresponding axis actions occur when a hardware limit point is triggered

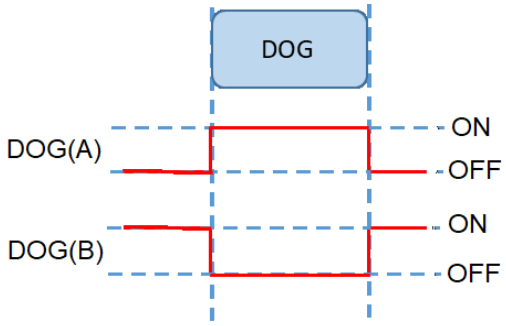
| Case | Action |
|---|---|
| DOG is not entered | The axis stops immediately, then speeds up toward the opposite direction from the frequency specified by JogSpeed until the frequency specified by TarSeed is reached and continues to seek the DOG signal. |
| At DOG | The axis stops immediately, then moves toward the opposite direction at the frequency specified by JogSpeed and continues to seek the DOG signal. |
| DOG is moved away from. | The axis stops immediately. |

10. If any error occurs as the instruction is in process of the output, the **Error** flag changes to ON. Refer to the error codes that **ErrCode** shows for the trouble shooting

11. The error codes that **ErrCode** shows are listed in the following table

| Error Code | Description |
|---|---|
| 16#1400 | The module does not support the function. |
| 16#1402 | There is no response from the module; communication timeout occurs. |
| 16#1403 | There is no such output axis number in the PU module. |
| 16#1405 | The output axis specified by the PU module is outputting data. It is not allowed to specify the output repeatedly. |
| 16#1406 | PU module stops Output pulse when the positive limit is reached. |
| 16#1407 | PU module stops Output pulse when the negative limit is reached. |

12. Explanation of DOG (A) and DOG (B) signals

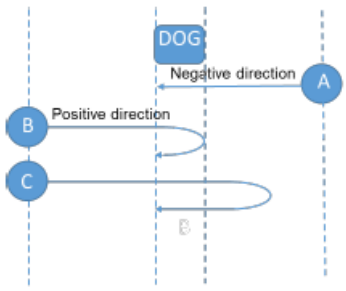
| Contact Type | | |
|---|---|---|
| DOG(A) | The DOG signal emerges and the axis enters DOG when the contact switches from OFF to ON.<br>The DOG signal disappears and the axis leaves DOG when the contact switches from ON to OFF. |  |
| DOG(B) | The DOG signal emerges and the axis enters DOG when the contact switches from ON to OFF.<br>The DOG signal disappears and the axis leaves DOG when the contact switches from OFF to ON. | |

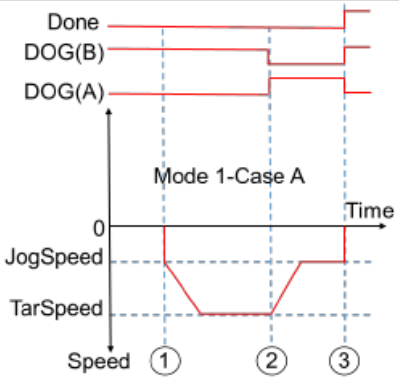
13. Explanation of homing modes

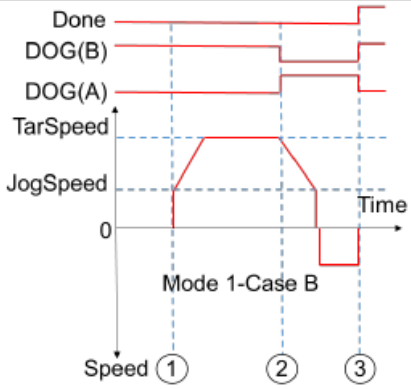
    Mode 0: Directly clear the position to 0.

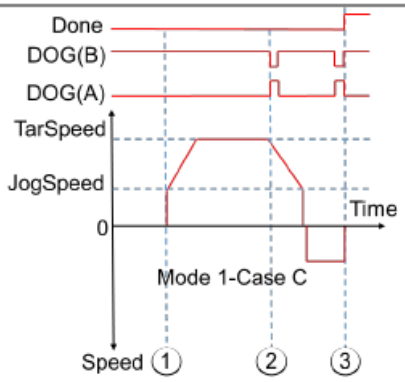
Mode 1: The axis starts to go toward the negative direction and then stops after leaving the DOG point position

| Description of cases under mode 1 | |
|---|---|
| Case A | The motion starts in the negative direction. As the DOG point is encountered, The axis starts to decelerate until the JOG speed is reached and then the axis stops while leaving the DOG point. |
| Case B | The motion starts in the positive direction; the deceleration time is less than the duration for the DOG signal. |
| Case C | The motion starts in the positive direction; the deceleration time is greater than the duration for the DOG signal. |




① The DPUZRN instruction is started; the axis accelerates from the speed specified by JogSpeed to the target speed specified by TarSpeed (positive direction: the value > 0 or negative direction: the value < 0) and then the axis keeps moving at the target speed.

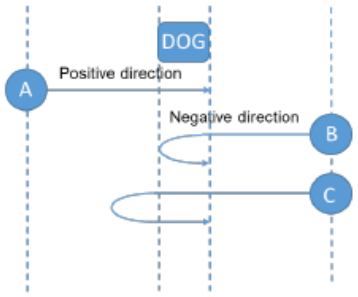
② After the DOG signal appears, the DOG signal is left in the following directions according to the selected mode.

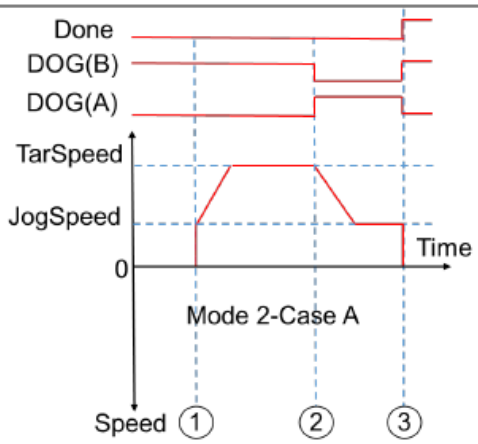
- In the previous direction as Case A shows, the axis decelerates to the JOG speed and then prepares for leaving the DOG signal.
- In the positive direction as CaseB/CaseC shows, the axis decelerates to the JOG speed and then stops. After that, it moves at the JOG speed in the opposite direction and prepares for moving away from the DOG signal.

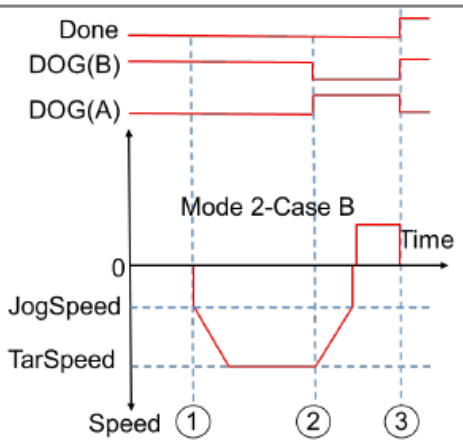
③ The DOG signal is moved away from and meanwhile the axis stops immediately.

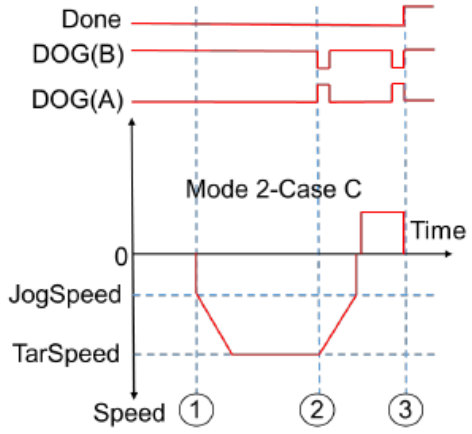
Mode 2: The axis starts to go toward the positive direction and then stops after leaving the DOG point position.

| Description of cases under mode 2 | |
|---|---|
| Case A | The motion starts in the positive direction. As the DOG point is encountered, the axis starts to decrease its speed until the JOG speed is reached and then stops after leaving the DOG point. |
| Case B | The motion starts in the negative direction, the deceleration time is less than the duration for the DOG signal. |
| Case C | The motion starts in the negative direction; the deceleration time is greater than the duration for the DOG signal. |




① The DPUZRN instruction is started and the axis accelerates from the speed specified by JogSpeed to the target speed specified by TarSpeed (positive direction: the value > 0 or negative direction: the value < 0) and then the motion continues at the target speed.

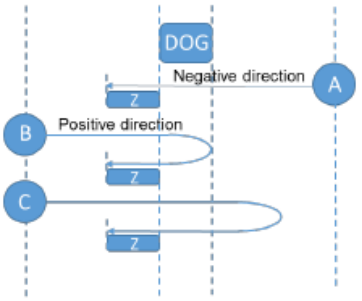
② After the DOG signal appears, the DOG signal is left in the following directions according to the selected mode.

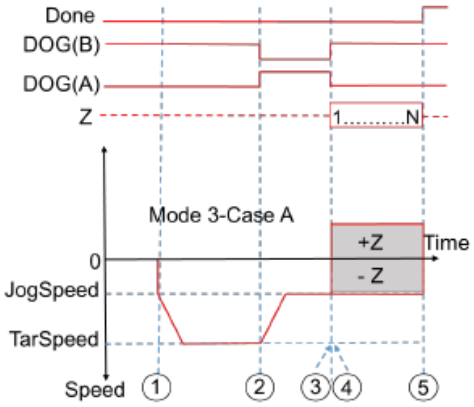
- In the previous direction as Case A shows, the axis decelerates to the JOG speed and then prepares for leaving the DOG signal.
- In the negative direction as CaseB/CaseC shows, the axis decelerates to the JOG speed and then stops. After that, it goes at the JOG speed in the opposite direction and prepares for leaving the DOG signal.

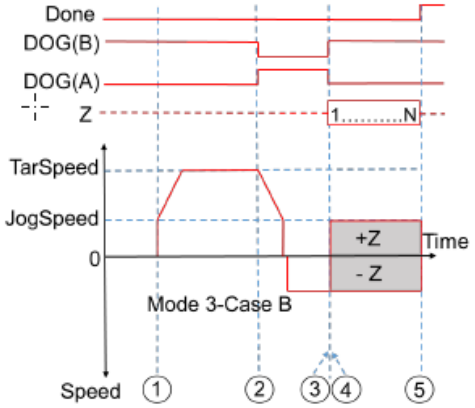
③ The DOG signal is left and meanwhile the axis stops immediately.

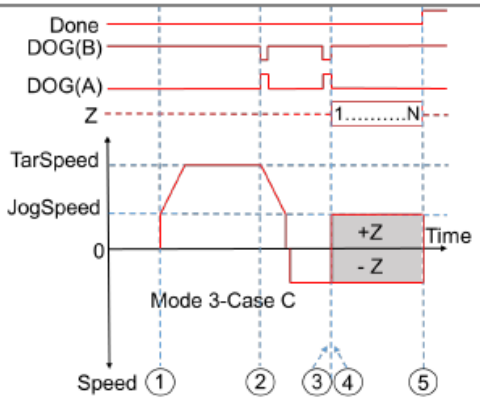
Mode 3: After Mode 1 is finished, seek the set number of Z phases.

| Description of cases under mode 3 | |
|---|---|
| Case A | The motion starts in the negative direction. As the DOG point is encountered, the axis starts to decelerate until the JOG speed is reached and then the search for a set number of Z phases begins before the axis stops immediately. |
| Case B | The motion starts in the positive direction; the deceleration time is less than the duration for the DOG signal. |
| Case C | The motion starts in the positive direction; the deceleration time is greater than the duration for the DOG signal. |




① The DPUZRN instruction is started and the axis accelerates from the speed specified by JogSpeed to the target speed specified by TarSpeed (positive direction: the value > 0 or negative direction: the value < 0) and then the motion continues at the target speed.

② After the DOG signal appears, the DOG signal is left in the following directions according to the selected mode.

- In the previous direction as Case A shows, the axis decelerates to the JOG speed and then prepares for leaving the DOG signal.
- In the positive direction as CaseB/CaseC shows, the axis decelerates to the JOG speed and then the axis stops. After that, the axis goes at the JOG speed in the opposite direction and prepares for leaving the DOG signal.

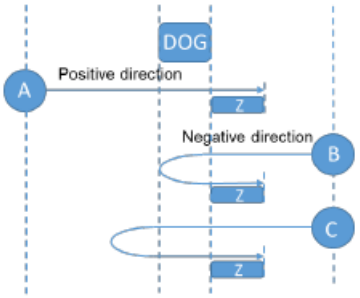
③ After the DOG signal is moved away from, the search for the set number of Z phase pulses begins (positive direction: the value > 0 or negative direction: the value < 0).

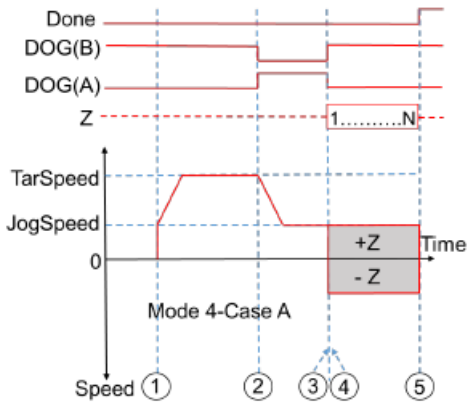
④ The first Z phase pulse is counted from.

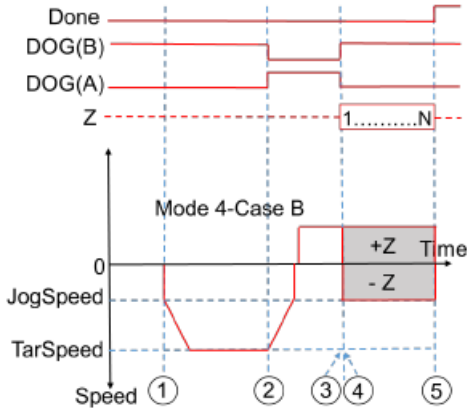
⑤ When counting to the $N^{th}$ Z phase pulse, the axis stops immediately.

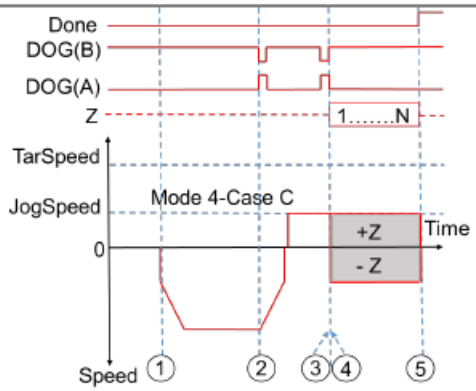
Mode 4: After Mode 2 is finished, seek the set number of Z phases.

| Description of cases under mode 4 | |
|---|---|
| Case A | The motion starts in the positive direction. As the DOG point is encountered, the axis starts to decelerate until the JOG speed is reached and then the search for a set number of Z phases begins before the axis stops immediately. |
| Case B | The motion starts in the negative direction; the deceleration time is less than the duration for the DOG signal. |
| Case C | The motion starts in the negative direction; the deceleration time is greater than duration for the DOG signal. |




① The DPUZRN instruction is started and the axis accelerates from the speed specified by JogSpeed to the target speed specified by TarSpeed (positive direction: the value > 0 or negative direction: the value < 0) and then the motion goes on at the target speed.

② After the DOG signal appears, the DOG signal is left in the following directions according to the selected mode.

- In the previous direction as Case A shows, the axis decelerates to the JOG speed and then prepares for leaving the DOG signal.
- In the negative direction as CaseB/CaseC shows, the axis accelerates to the JOG speed and then the motion stops. After that, the axis continues at the JOG speed in the opposite direction and prepares for leaving the DOG signal.

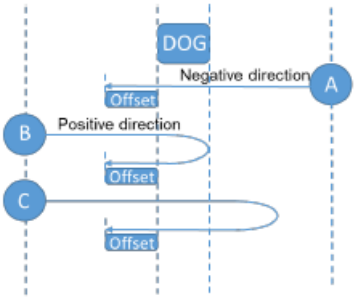
③ After the DOG signal is left, the search for a set number of Z phase pulses starts (positive direction: the value > 0 or negative direction: the value < 0)

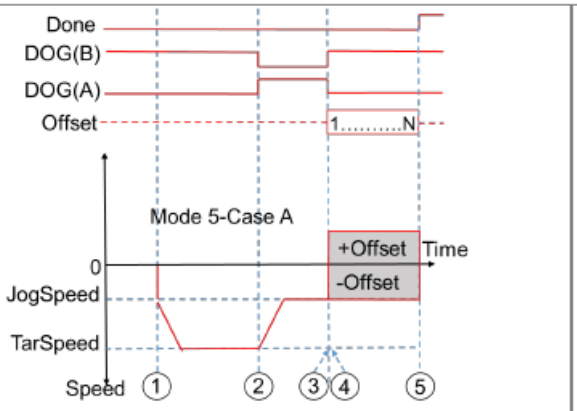
④ The first Z phase pulse is counted from.

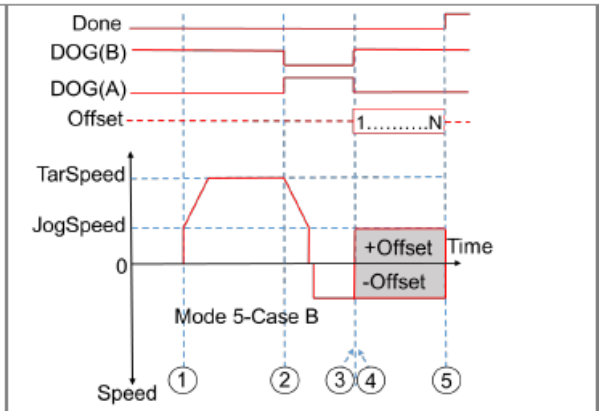
⑤ When counting to the $N^{th}$ Z phase pulse is completed, the axis stops immediately.

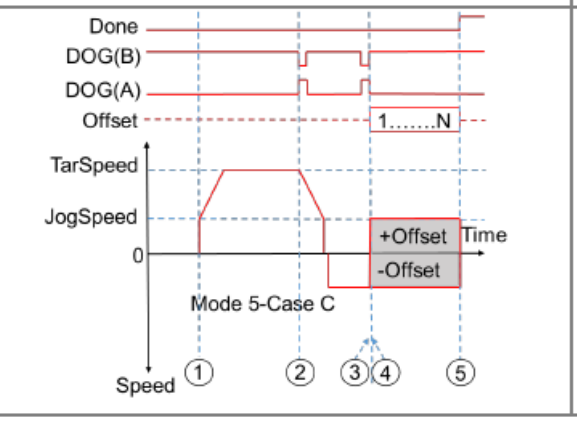
Mode 5: After Mode 1 is finished, output the offset position.

| Description of cases under mode 5 | |
|---|---|
| Case A | The motion starts in the negative direction. As the DOG point is encountered, the axis starts to decelerate until the JOG speed is reached, and then the offset position is output. When the offset outputs are completed, the axis stops right away. |
| Case B | The motion starts in the positive direction; the deceleration time is less than the duration for the DOG signal. |
| Case C | The motion starts in the positive direction; the deceleration time is greater than the duration for the DOG signal. |




① The DPUZRN instruction is started and the axis accelerates from the speed specified by JogSpeed to the target speed specified by TarSpeed (positive direction: the value > 0 or negative direction: the value < 0) and then the motion goes on at the target speed.

② After the DOG signal appears, the DOG signal is left in the following directions according to the selected mode.

- In the previous direction as Case A shows, the axis decelerates to the JOG speed and then prepares for leaving the DOG signal.
- In the positive direction as CaseB/CaseC shows, the axis decelerates to the JOG speed and then stops. After that, the axis continues at the JOG speed in the opposite direction and prepares for leaving the DOG signal.

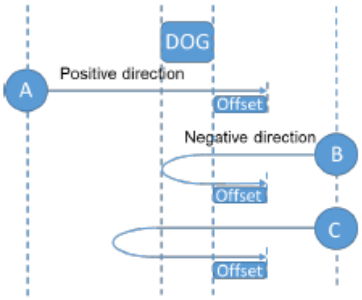
③ After the DOG signal is moved away from, the pulses of the number specified by Offset are output (positive direction: the value > 0 or negative direction: the value < 0).

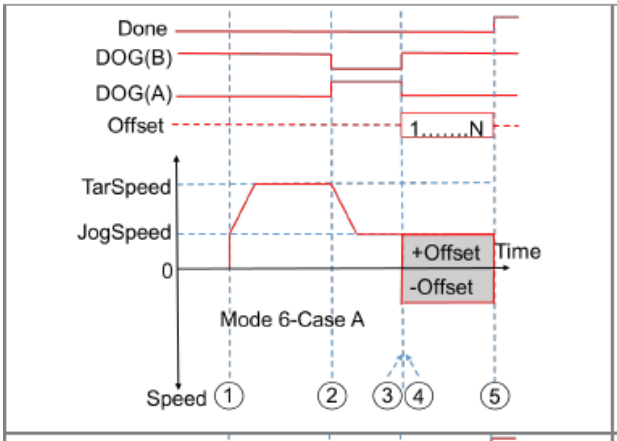
④ The first offset pulse is output.

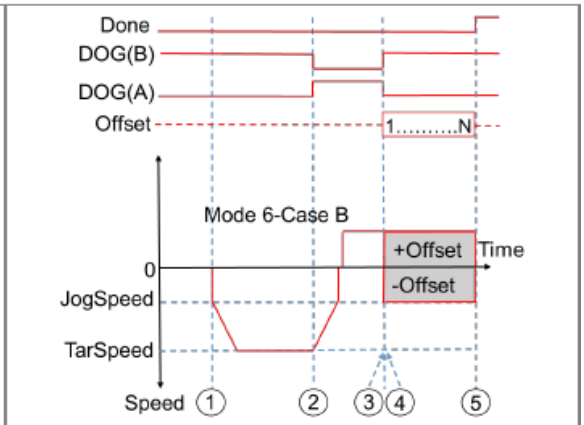
⑤ When the $N^{th}$ offset pulse output is completed, the axis stops immediately.

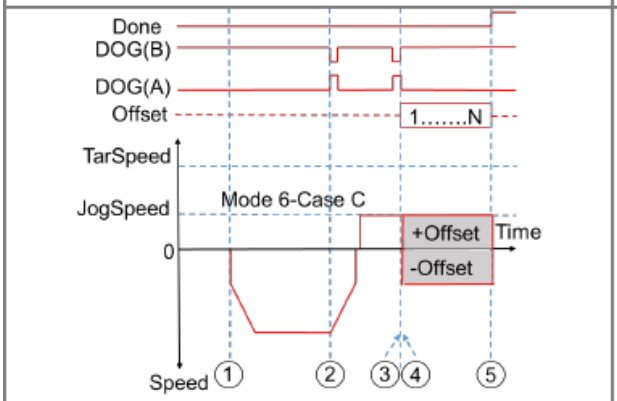
Mode 6: After Mode 2 is finished, output the offset position.

| Description of cases under mode 6 | |
|---|---|
| Case A | The motion starts in the positive direction. As the DOG point is encountered, the axis starts to decelerates until the JOG speed is reached and then the offset position is output. When the offset outputs are completed, the axis stops right away. |
| Case B | The motion starts in the negative direction; the deceleration time is less than the duration for the DOG signal. |
| Case C | The motion starts in the negative direction; the deceleration time is greater than the duration for the DOG signal. |




① The DPUZRN instruction is started and the axis accelerates from the speed specified by JogSpeed to the target speed specified by TarSpeed (positive direction: the value > 0 or negative direction: the value < 0) and then the motion goes on at the target speed.

② After the DOG signal appears, the DOG signal is left in the following directions according to the selected mode.

- In the previous direction as Case A shows, the axis decelerates to the JOG speed and then prepares for leaving the DOG signal.
- In the negative direction as CaseB/CaseC shows, the axis decelerates to the JOG speed and then it stops. After that, the axis continues at the JOG speed in the opposite direction and prepares for leaving the DOG signal.

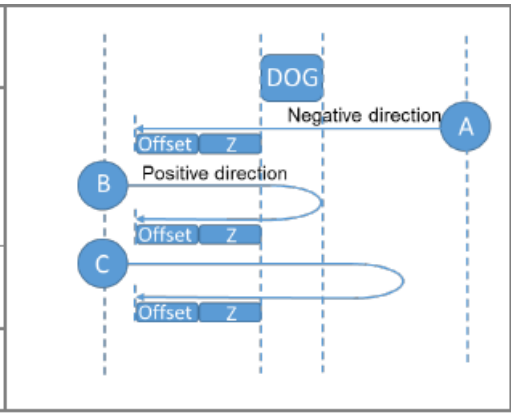
③ After the DOG signal is moved away from, the pulses of the number specified by Offset are output (positive direction: the value > 0 or negative direction: the value < 0).

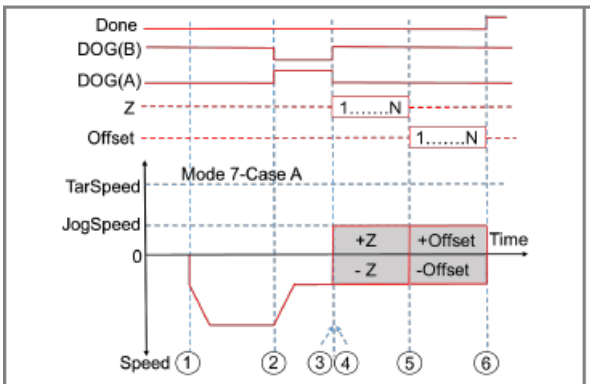
④ The first offset pulse is output.

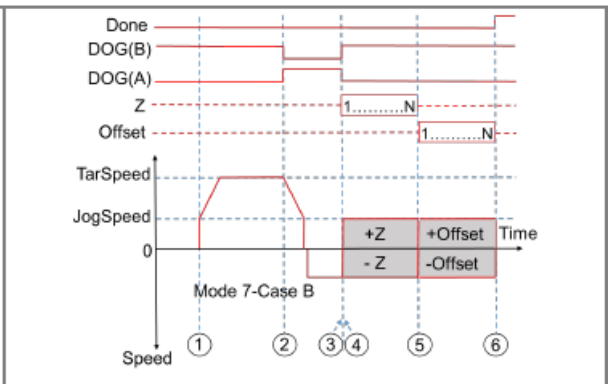
⑤ When the $N^{th}$ offset pulse output is completed, the axis stops immediately.

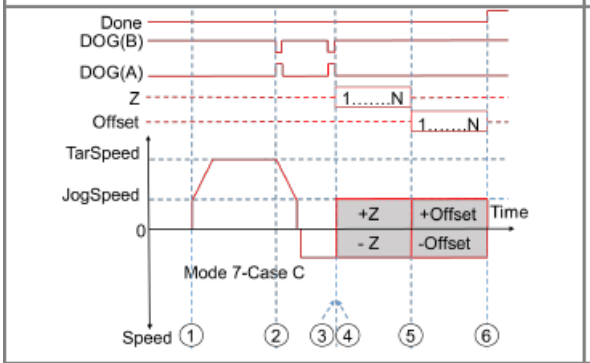
Mode 7: After Mode 1 is finished, seek the set number of Z phases and then output the offset position.

| Description of cases under mode 7 | |
|---|---|
| Case A | The motion starts in the negative direction. As the DOG point is encountered, the axis starts to decelerates until the JOG speed is reached and then the search for Z phases starts. After the last Z phase is counted to, the offset position output begins. When offset outputs are completed, the axis stops right away. |
| Case B | The motion starts in the positive direction; the deceleration time is less than the duration for the DOG signal. |
| Case C | The motion starts in the positive direction; the deceleration time is greater than the duration for the DOG signal. |




① The DPUZRN instruction is started and the motion accelerates from the speed specified by JogSpeed to the target speed specified by TarSpeed (positive direction: the value > 0 or negative direction: the value < 0) and then the motion goes on at the target speed.

② After the DOG signal appears, the DOG signal is left in the following directions according to the selected mode.

- In the previous direction as Case A shows, the axis decelerates to the JOG speed and then prepares for leaving the DOG signal.
- In the positive direction as CaseB/CaseC shows, the axis decelerates to the JOG speed and then stops. After that, the axis continues at the JOG speed in the opposite direction and prepares for moving away from the DOG signal.

③ After the DOG signal is moved away from, the search for a set number of Z phases (positive direction: the value > 0 or negative direction: the value < 0).

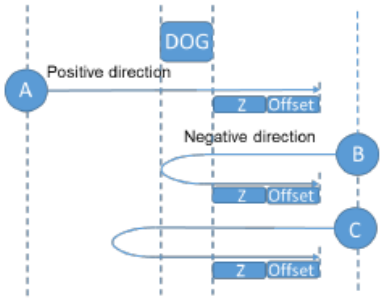
④ The first Z phase pulse is counted from.

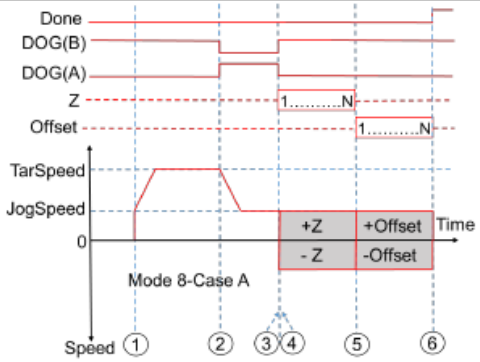
⑤ When counting to the $N^{th}$ Z phase pulse, the first offset pulse output starts (positive direction: the value > 0 or negative direction: the value < 0).

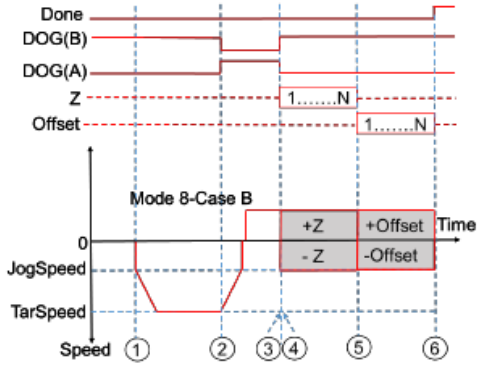
⑥ When the $N^{th}$ offset pulse output is completed, the axis stops immediately.

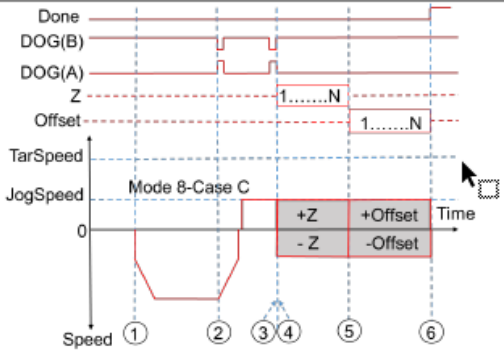
Mode 8: After Mode 2 is finished, seek the set number of Z phases and then output the offset position.

| Description of cases under mode 2 | |
|---|---|
| Case A | The motion starts in the positive direction. As the DOG point is encountered, the axis starts to decelerates until the JOG speed is reached and then the search for Z phases starts. After the last Z phase is counted to, the offset position output begins. When offset outputs are completed, the axis stops right away. |
| Case B | The motion starts in the negative direction; the deceleration time is less than the duration for the DOG signal. |
| Case C | The motion starts in the negative direction; the deceleration time is greater than the duration for the DOG signal. |




① The DPUZRN instruction is started and the axis accelerates from the speed specified by JogSpeed to the target speed specified by TarSpeed (positive direction: the value > 0 or negative direction: the value < 0) and then the motion goes on at the target speed.

② After the DOG signal appears, the DOG signal is left in the following directions according to the selected mode.

- In the previous direction as Case A shows, the axis decelerates to the JOG speed and then prepares for leaving the DOG signal.
- In the negative direction as CaseB/CaseC shows, the axis decelerates to the JOG speed and then stops. After that, the axis continues at the JOG speed in the opposite direction and prepares for moving away from the DOG signal.

③ After the DOG signal is moved away from, the search for a set number of Z phases (positive direction: the value > 0 or negative direction: the value < 0).

④ The first Z phase pulse is counted from.

⑤ When counting to the $N^{th}$ Z phase pulse, the first offset pulse output starts (positive direction: the value > 0 or negative direction: the value < 0).

⑥ When the $N^{th}$ offset pulse output is completed, the axis stops immediately.

## Programming Example

Refer to the description of DPUDRI instruction (API 1405) for more information.

| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1408 | D | PUJOG | | Module - ErrCode | PU module jog output |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| Axis | | | | | | | | ● | | | | | ○ | ○ | | |
| JogSpeed | | | | | | | | ● | | | | | ○ | ○ | | |
| Busy | | ● | ● | ● | | | | ● | | | | | | | | |
| Error | | ● | ● | ● | | | | ● | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Axis | | ● | | | | ● | | | | | | | |
| JogSpeed | | | ● | | | | ● | | | | | | |
| Busy | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| – | – | ES3 |

## Symbol

**Module:** Module number
**Axis:** Output axis number
**JogSpeed:** The jog frequency for the homing
**Busy:** Output in execution
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later. For ISPSoft, we recommend using software version 3.16 and above. When En setting is set to ON, this instruction would be effective. Once the setting changes to OFF, Output pulse would decelerate until it stops

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag **Error** will change to ON

3. **Module** sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and

the module numbered as 2 is 02PU-E2

4. **Axis** sets the output axis number for the specified PU module. The setting values 1~2 represent the axis1~axis2 output of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag **Error** will change to ON

5. **JogSpeed** sets the jog output frequency. The setting value is a signed 32 bit value within the range of -200,000 (-200K) ~200,000 (200K) (Hz). When the value is greater than 0, the output will go in the positive direction (and the direction output point is off). When the value is less than 0, the output will go in the negative direction (and the direction output point is on). When the value is 0, the output will stop

6. The instruction can be used for the speed change. While the instruction is being executed, you can change the value of **TarSpeed** so as to change the output speed. When the setting value exceeds the maximum frequency, the instruction would be executed at the maximum frequency. But changing the speed would not change the direction. To change the direction, set the value of **TarSpeed** to 0 first and then modify the target speed

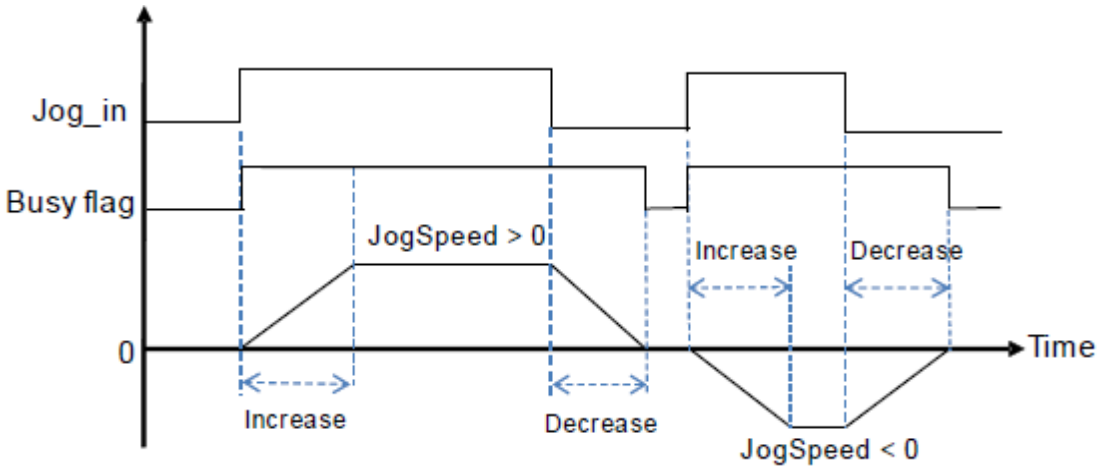
7. The instruction can be used with the software and hardware limit points. When the limits are triggered, the output stops immediately and the **Error** flag changes to ON

8. If any error occurs as the instruction is in process of the output, the **Error** flag changes to ON. Refer to the error codes that **ErrCode** shows for the trouble shooting

9. The error codes that **ErrCode** shows are listed in the following table

| Error Code | Description |
|---|---|
| 16#1400 | The module does not support the function. |
| 16#1402 | There is no response from the module; communication timeout occurs. |
| 16#1403 | There is no such output axis number in the PU module. |
| 16#1605 | The output axis specified by the PU module is outputting data. It is not allowed to specify the output repeatedly. |
| 16#1606 | PU module stops Output pulse when the positive limit is reached. |
| 16#1607 | PU module stops Output pulse when the negative limit is reached. |

10. See the output timing diagram as below. (Jog_in is the switch to start the instruction and the Busy flag is the **Busy** flag.)



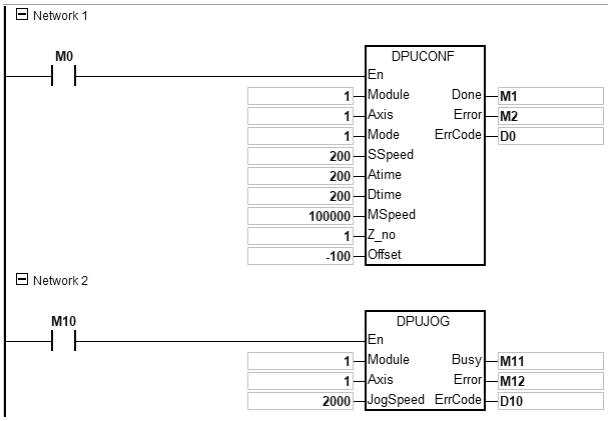

11. After the PUJOG instruction is disabled and the **Busy** flag is off, other output control can be carried out

## Programming Example

1. When M0 is ON, the DPUCONF instruction for axis 1 is executed to modify the parameters by setting **Mode** to 1 (Pulse Y0 + direction Y1), **SSpeed** to 200Hz, **Atime** to 200ms, **Dtime** to 200ms and **MSpeed** to 100kHz. After the output of **Done** is completed, M1 is ON

2. When M10 is ON, the DPUJOG instruction for axis 1 starts to perform jog outputs. The pulses are output from Y0 at the frequency of 2KHz. If Y1 is OFF, the direction is positive. And M11 is ON during the instruction

execution


Network 1
M0
DPUCONF
En
1
Module
Done
M1
1
Axis
Error
M2
1
Mode
ErrCode
D0
200
SSpeed
200
Atime
200
Dtime
100000
MSpeed
1
Z_no
-100
Offset
Network 2
M10
DPUJOG
En
1
Module
Busy
M11
1
Axis
Error
M12
2000
JogSpeed
ErrCode
D10

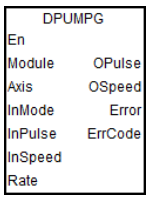
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1409 | D | PUMPG | | **Module ~ ErrCode** | PU module MPG output |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Module** | | | | | | | | ● | | | | | ○ | ○ | | |
| **Axis** | | | | | | | | ● | | | | | ○ | ○ | | |
| **InMode** | | | | | | | | ● | | | | | ○ | ○ | | |
| **InPulse** | | | | | | | | ● | | | | | | | | |
| **InSpeed** | | | | | | | | ● | | | | | | | | |
| **Rate** | | | | | | | | ● | | | | | | | | ○ |
| **OPulse** | | | | | | | | ● | | | | | | | | |
| **OSpeed** | | | | | | | | ● | | | | | | | | |
| **Error** | | ● | ● | ● | | | | ● | | | | | | | | |
| **ErrCode** | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Module** | | ● | | | | ● | | | | | | | |
| **Axis** | | ● | | | | ● | | | | | | | |
| **InMode** | | ● | | | | ● | | | | | | | |
| **InPulse** | | | ● | | | | ● | | | | | | |
| **InSpeed** | | | ● | | | | ● | | | | | | |
| **Rate** | | | | | | | | | ● | | | | |
| **OPulse** | | | ● | | | | ● | | | | | | |
| **OSpeed** | | | ● | | | | ● | | | | | | |
| **Error** | ● | | | | | | | | | | | | |
| **ErrCode** | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| – | – | ES3 |

## Symbol




**Module:** Module number
**Axis:** Output axis number
**InMode:** Encoder input mode and frequency multiplication for counting
**InPulse:** Number of pulses which have been input
**InSpeed:** Detected input frequency
**Rate:** Input/output rate (floating point number)
**OPulse:** Number of pulses which have been output
**OSpeed:** Frequency at which pulses are being output
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC CPU with FW V1.06.00 or later and module with FW V1.04.00 or later. For ISPSoft, we recommend using software version 3.16 and above. When En setting is set to ON, this instruction would be effective. Once the setting changes to OFF, Output pulse would decelerate until it stops

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag Error will change to ON

3. Module sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and the module numbered as 2 is 02PU-E2

4. Axis sets the output axis number for the specified PU module. The setting values 1~2 represent the axis1~axis2 output of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag Error will change to ON

5. InMode sets the input mode of the encoder source and the frequency multiplication for counting.
   See the explanation of InMode value in the following table

6. **InPulse** displays the number of already input pulses, which is a signed 32-bit value. Every time the instruction is started, the PU module will automatically clear the value to 0 and then starts counting

7. **InSpeed** displays the already detected input frequency which is a 32-bit value. The basic time for the frequency detection is 20ms. Therefore, the detected input frequency is 0 if there is no counting value within 20ms. If there is a counting value within 20ms, the output starts at the minimum frequency of 50Hz. Even if **OSpeed** value is lower than 50Hz through the **Rate-**value-based conversion, the output is still conducted at 50Hz

8. **Rate** is the input / output rate and the value is a floating point number. The number of actual output pulses and frequency are respectively equal to the input pulse number and frequency multiplied by the rate value

9. OPulse shows the number of pulses which have been output. OSpeed displays the frequency at which the output is being conducted. They are signed 32-bit values

10. When the DPUMPG instruction is disabled, check the frequency at which the output is being conducted and see if it has reached 0. If the instruction is disabled before the frequency reaches 0, the PU module will stop the output immediately and the output of the pulses which are counted based on the conversion rate will not continue any more

11. The error codes that ErrCode shows are listed in the following table

12. When the DPUMPG instruction is enabled or disabled, the PLC will have to notify the module to enable or disable the high-speed counter function. Thus the instruction can not be used with API1410 DPUCNT together. Otherwise it may occur that the two instructions enable or disable the counting of the module with each other

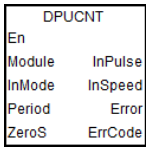
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1410 | D | PUCNT | | Module ~ ErrCode | High-speed counter function of PU module |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| InMode | | | | | | | | ● | | | | | ○ | ○ | | |
| Period | | | | | | | | ● | | | | | ○ | ○ | | |
| ZeroS | ● | ● | ● | ● | | | | ● | | | | | | | | |
| InPulse | | | | | | | | ● | | | | | | | | |
| InSpeed | | | | | | | | ● | | | | | | | | |
| Error | | ● | ● | ● | | | | ● | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| InMode | | ● | | | | ● | | | | | | | |
| Period | | ● | | | | ● | | | | | | | |
| ZeroS | ● | | | | | | | | | | | | |
| InPulse | | | ● | | | | ● | | | | | | |
| InSpeed | | | ● | | | | ● | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | | ● | | | | ● | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| — | — | ES3 |

## Symbol




**Module:** Module number
**InMode:** Encoder input mode and frequency multiplication for counting
**Period:** Period time for capturing the frequency
**ZeroS:** Clear the counter to 0
**InPulse:** Number of pulses which have been input
**InSpeed:** Detected input frequency
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later. For ISPSoft, we recommend using software version 3.16 and above. When En setting is set to ON, this instruction would be effective. Once the setting changes to OFF, the counting would be stopped immediately

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag Error will change to ON

3. Module sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and the module numbered as 2 is 02PU-E2

4. InMode sets the input mode of the encoder source and the frequency multiplication for counting See the explanation of InMode value in the following table.
Note: Phase A leads phase B, indicating counting in the positive direction.
Phase B leads phase A, indicating counting in the negative direction

| **Value** | **Input Modes**<br>Set as the following values, otherwise the module will use defaults |
|---|---|
| 16#0000 | Reserved |
| 16#0001 | Onefold frequency A/B phase input |
| 16#0002 | Twofold frequency A/B phase input |
| 16#0003 | Reserved |
| 16#0004 | Fourfold frequency A/B phase input (default) |
| 16#0005 | Pulse + directional input (A+/A-: pulse input; B+/B-: directional input)<br>Phase B ON: counting in the negative direction<br>Phase B OFF: counting in the positive direction<br>Phase A: counting is started by rising-edge trigger |
| 16#0006 | Pulse + directional input (A+/A-: pulse input; B+/B-: directional input)<br>Phase B ON: counting in the positive direction<br>Phase B OFF: counting in the negative direction<br>Phase A: counting is started by rising-edge trigger |
| 16#0007 | Single phase pulse input (A+/A-: pulse input)<br>Phase A: counting is started by rising-edge trigger. |
| Others | Reserved |

5. Period is the setting value of a cycle time for capturing the frequency within the range of 10ms ~ 1000ms. If the setting value exceeds the range, the maximum value or minimum value will be automatically taken as the setting value by the PLC

6. ZeroS clears the present output position to 0. If the present axis position is to be cleared to 0, set ZeroS from OFF to ON when the instruction is started

7. InPulse is the number of already input pulses, which is a signed 32-bit value. The counting value is a latched value. If the value need be cleared to 0, just set ZeroS from Off to ON while the instruction is running

8. InSpeed displays the counting value for every Period time, which is a signed 32-bit value. If you need convert it into the value with the unit of Hz, use the calculation formula for conversion by yourself

9. The error codes that ErrCode shows are listed in the following table

| **Error Code** | **Description** |
|---|---|
| 16#1400 | The module does not support the function. |
| 16#1402 | There is no response from the module; communication timeout occurs. |

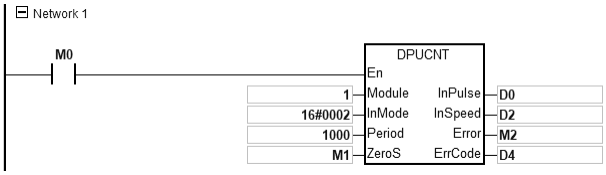
10. When the DPUCNT instruction is enabled or disabled, the PLC will have to notify the module to enable or disable the high-speed counter function. Thus the instruction can not be used with API1409 DPUMPG together. Otherwise it may occur that the two instructions enable or disable the counting of the module with each other

## Programming Example

1. When M0 is ON, the DPUCNT instruction is executed and InMode is set to “twofold frequency A/B phase input”

2. When M1 is ON, the counted number of pulses on axis 1 in InPulse is cleared

3. When the input number of pulses is 100 and frequency is 10Hz, InPulse and InSpeed show 200 pulses and 20Hz respectively



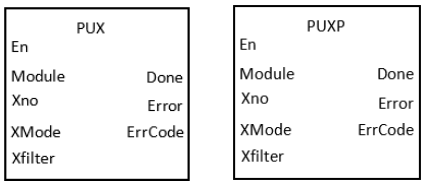
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1411 | | PUX | P | Module ~ Error、ErrCode | Setting PU module input point mode |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| Xno | | | | | | | | ● | | | | | ○ | ○ | | |
| XMode | | | | | | | | ● | | | | | ○ | ○ | | |
| Xfilter | | | | | | | | ● | | | | | ○ | ○ | | |
| Done | | ● | ● | ● | | | | ● | | | | | | | | |
| Error | | ● | ● | ● | | | | ● | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Xno | | ● | | | | ● | | | | | | | |
| XMode | | ● | | | | ● | | | | | | | |
| Xfilter | | ● | | | | ● | | | | | | | |
| Done | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| ES3 | ES3 | - |

## Symbol




**Module:** Module number
**Xno:** Input point number
**XMode:** Input point mode
**Xfilter:** Input point filter time
**Done:** Completion flag
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 or later and the firmware for the module must be V1.00.00 or above. For ISPSoft, we recommend using software version 3.16 and above. The timing to set this instruction is when the En setting changes from OFF to ON

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag Error will change to ON

3. Module sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module

numbered as 1 is 04AD-E2 and the module numbered as 2 is 02PU-E2

4. Xno sets the input point number for the PU module with one of the input values 0~4 which respectively represent the input points X0~X4. If there is no corresponding input point in the PU module, the error flag Error will change to ON. The value in Xno and the corresponding input point are listed in the following table

| **Setting in Xno** | **0** | **1** | **2** | **3** | **4** | **Others** |
|---|---|---|---|---|---|---|
| DVP02PU-E2 | X0 | X1 | X2 | X3 | X4 | No input point |

5. XMode selects an input mode for input points. Setting values are explained in the following table:

| **Setting in XMode** | **DVP02PU-E2** |
|---|---|
| 0 | General input (Default) |
| 1 | Axis 1, Z phase, Rising-edge triggered |
| 2 | Axis 2, Z phase, Rising-edge triggered |
| 3 | Axis 1, Z phase, Falling-edge triggered |
| 4 | Axis 2, Z phase, Falling-edge triggered |
| 5 | Axis 1, DOG, Rising-edge triggered |
| 6 | Axis 2, DOG, Rising-edge triggered |
| 7 | Axis 1, DOG, Falling-edge triggered |
| 8 | Axis 2, DOG, Falling-edge triggered |
| 9 | Axis 1, LSN, Rising-edge triggered |
| 10 | Axis 2, LSN, Rising -edge triggered |
| 11 | Axis 1, LSN, Falling-edge triggered |
| 12 | Axis 2, LSN, Falling-edge triggered |
| 13 | Axis 1, LSP, Rising-edge triggered |
| 14 | Axis 2, LSP, Rising-edge triggered |
| 15 | Axis 1, LSP, Falling-edge triggered |
| 16 | Axis 2, LSP, Falling-edge triggered |
| Others | Automatically switch to mode 0 (default) |

6. Xfilter is explained in the following table. The value in Xfilter is the default value if the setting is out of the allowed range

| Parameter | Function | Range | Default |
|---|---|---|---|
| Xfilter | Input point filter time | 0 ~ 25 [ms] | 10 |

7.

8. Done, an output of the specified PU module has been set as the completion flag. When Done is On, it indicates that the parameter setting is successful. You can continue to perform positioning output based on the state of the completion flag (ON). The clearing of the Done flag need be conducted by manual. The Done flag changes to ON only when the setting is completed. 8. Error, an output of the specified PU module is a parameter error flag. Most parameter ranges are filtered automatically by the PLC. Thus if the error flag is ON, it means that there is no specified PU module or the PU module number is wrong or the output axis number is incorrect

9. The instruction is a pulse instruction. Even if the A contact is adopted as the condition contact, PU module parameters are also set only when the instruction is started. Therefore, if a parameter value is to be updated, restart the instruction to make the parameter set again

10. Since the set parameters are delivered through the module communication command, confirm the state of the output Done or Error before a parameter value is modified and then proceed with relevant operations

11. For the state of PU module input points, check the data exchange function of the special extension module, SM228. Refer to Section 2.2.16 Additional Remarks on Special Auxiliary Relays and Special Data Registers in DVP-ES3 Series Programming Manual for details on SM228

12. The error codes that ErrCode shows are listed in the following table

| Error Code | Description |
|---|---|
| 16#1400 | The module does not support the function. |
| 16#1402 | There is no response from the module; communication timeout occurs. |

## Programming Example

Refer to the description of DPUDRI instruction (API 1405) for more information.

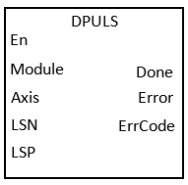
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 1412 | D | PULS | P | Module ~ Error、ErrCode | Setting PU module software limits |

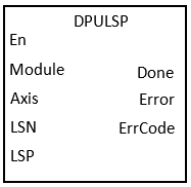
| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | | | | | | | ● | | | | | ○ | ○ | | |
| Axis | | | | | | | | ● | | | | | ○ | ○ | | |
| LSN | | | | | | | | ● | | | | | ○ | ○ | | |
| LSP | | | | | | | | ● | | | | | ○ | ○ | | |
| Done | | ● | ● | ● | | | | ● | | | | | | | | |
| Error | | ● | ● | ● | | | | ● | | | | | | | | |
| ErrCode | | | | | | | | ● | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Module | | ● | | | | ● | | | | | | | |
| Axis | | ● | | | | ● | | | | | | | |
| LSN | | | ● | | | | ● | | | | | | |
| LSP | | | ● | | | | ● | | | | | | |
| Done | ● | | | | | | | | | | | | |
| Error | ● | | | | | | | | | | | | |
| ErrCode | | ● | | | | ● | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| ES3 | - | ES3 |

## Symbol




**Module:** Module number
**Axis:** Output axis number
**LSN:** Software negative limit
**LSP:** Software positive limit
**Done:** Completion flag
**Error:** Error flag
**ErrCode:** Error code

## Explanation

1. This instruction is available for PLC with FW V1.06.00 and later and the firmware for the module must be V1.00.00 or above. For ISPSoft, we recommend using software version 3.16 and above. The timing to set this instruction is when the En setting changes from OFF to ON

2. The instruction is exclusive to the PU modules at the right of the PLC. If the specified module is not a PU module, the error flag Error will change to ON

3. Module sets the serial number of non-DIO modules at the right of the PLC. Only non-DIO modules at the right of the PLC will be numbered. The first non-DIO module is numbered as 1, the second non-DIO module is numbered as 2 and so forth. The maximum number is 8. For example, when a 32ES3 PLC CPU connects to modules 16XP2 + 08XM2 + 04AD-E2 + 02PU-E2 at its right size, the module numbered as 1 is 04AD-E2 and the module numbered as 2 is 02PU-E2. 4. Axis sets the output axis

number for the specified PU module. The setting values 1~2 represent the axis1~axis2 of the specified PU module respectively. If the PU module has no corresponding axis number for output, the error flag Error will change to ON

4. See the following combination of axis numbers and corresponding output points of PU modules

| PU Module Name | Axis 1 | Axis 2 |
|---|---|---|
| DVP02PU-E2 | Y0/Y1 | Y2/Y3 |

5. See the explanation of LSN and LSP and setting values in the following table. If the setting value is outside the range, the instruction will automatically be executed at the minimum or maximum value

| Parameter | Function | Range | Default | Remark |
|---|---|---|---|---|
| LSN | Software negative limit | -2,147,483,648 ~ +2,147,483,647 | 0 | Inactive when both are set to 0 |
| LSP | Software positive limit | -2,147,483,648 ~ +2,147,483,647 | 0 | |

6. Done, an output of the specified PU module has been set as the completion flag. When Done is On, it indicates that the parameter setting is successful. You can continue to perform positioning output based on the state of the completion flag (ON). The clearing of the Done flag need be conducted by manual. The Done flag changes to ON only when the setting is completed

7. Error, an output of the specified PU module is a parameter error flag. Most parameter ranges are filtered automatically by the PLC. Thus if the error flag is ON, it means that there is no specified PU module or the PU module number is wrong or the output axis number is incorrect

8. The instruction is a pulse instruction. Even if the A contact is adopted as the condition contact, PU module parameters are also set only when the instruction is started. Therefore, if a parameter value is to be updated, restart the instruction to make the parameter set again

9. Since the set parameters are delivered through the module communication command, confirm the state of the output Done or Error before a parameter value is modified and then proceed with relevant operations

10. The error codes that ErrCode shows are listed in the following table

| Error Code | Description |
|---|---|
| 16#1400 | The module does not support the function. |
| 16#1402 | There is no response from the module; communication timeout occurs. |

## Programming Example

Refer to the description of DPUDRI instruction (API 1405) for more information.

## Data Exchange of PU Modules

1. SM228 and D2800 - D28079 When SM228 is ON, it is to disable the data exchange among the CPU and its connected modules.
When SM228 is OFF, data exchange among the CPU and its connected modules is enabled and data is stored in D2800 - D28079. If the PLC is connected with a special extension module, the PLC uses registers in D28000-D28079 and the registers in this area correspond to CRs to update data. If you need to use this area, you need to pay attention not to use the same area repeatedly.
See the example below to learn how this works.

| Order Number | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Model | DVP02PU-E2 | DVP06XA-E2 | DVP04DA-E2 | DVP04TC-E2 | DVP04PT-E2 | DVP06PT-E2 |
| Reading data | Reading values from D28000: the input value of X point and D28001: the code of the axis state | Reading values from AD channels 1-4 (D28010 - D28013) | N(A | Reading values from TC channels 1-4 (D28030 - D28033) | Reading values from PT channels 1-4 (D28040 - D28043) | Reading values from PT channels 1-6 (D28050 - D28055) |
| Writing data | N/A | Writing values into DA channels 1-2 (D28014 - D28015) | Writing values into DA channels 1-4 (D28020 - D28023) | N/A | N/A | N/A |

When SM228 is ON, the PLC CPU will disable the data exchange function with the module.

| State Code Byte # | Description | Axis |
|---|---|---|
| 0 | Error flag | 1 |
| 1 | The output is active. | |
| 2 | The output has stopped working. | |
| 3 | The instruction execution is complete. | |
| 4 | The positive limit is reached. | |
| 5 | The negative limit is reached. | |
| 6 | Current position value overflow | |
| 7 | Pulse direction (positive or negative) | |
| 8 | Error flag | 2 |
| 9 | The output is active. | |
| 10 | The output has stopped working. | |
| 11 | The instruction execution is complete. | |
| 12 | The positive limit is reached. | |
| 13 | The negative limit is reached. | |
| 14 | Current position value overflow | |
| 15 | Pulse direction (positive or negative) | |

The corresponding error flag will be ON when the above mentioned incidents happened: 4/12, 5/13, 6/14. Once the error flag is ON, you need to use instruction to clear the shown error codes.

2. Descriptions of the values in SR1560-SR1568

| **SR** | **Description** |
|---|---|
| SR1560 | Number of the right-side modules connected to the CPU module |
| SR1561-SR1568 | Model code of the 1st – 8th right-side module connected to the CPU module |

Number of modules and the device codes

| Model | Device Code |
|---|---|
| DVP04AD-E2 | 16#0080 |
| DVP04DA-E2 | 16#0081 |

| | |
|---|---|
| DVP02DA-E2 | 16#0041 |
| DVP06XA-E2 | 16#00C4 |
| DVP04PT-E2 | 16#0082 |
| DVP04TC-E2 | 16#0083 |
| DVP06PT-E2 | 16#00C2 |
| DVP02PU-E2 | 16#0045 |

## List of Timer and Counter Instructions

The following table lists new Timer instructions.

| API | Instruction Code | | Pulse Instruction | Function |
|---|---|---|---|---|
| | 16-Bit | 32-Bit | | |
| 1016 | ATMR | - | - | 16-bit contact timer (Unit: 100ms) |
| 1017 | ATMRM | - | - | 16-bit contact timer (Unit: 1ms) |
| 1018 | ATMRH | - | - | 16-bit contact timer (Unit: 10ms) |

## BU-2_Timers

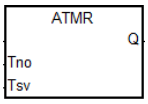
| API | Instruction code | Operand | Function |
|---|---|---|---|
| 1016 | ATMR | Tno、Tsv | 16-bit contact timer (Unit: 100ms) |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tno | | | | | ○ | | | | | | | | | | | |
| Tsv | | | | | | | | ○ | | | | ○ | ○ | ○ | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tno | | | | | | | | | | | ● | | |
| Tsv | | ● | | | | ● | | | | | | | |

| Pulse instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| - | ES3/SV3/SX3 | - |

## Symbol




**Tno:** Timer number
**Tsv:** Setting value for the timer

## Explanation

1. The instruction is available for ES3 with firmware V1.06.00 and later and for SV3/SX3 with firmware V1.00.00 or later

2. The ATMR instruction uses 100ms as the timing unit in the timer. Refer to the explanation of the ATMRH instruction (API 1018) for details

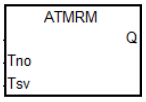
| API | Instruction code | Operand | Function |
|---|---|---|---|
| 1017 | ATMRM | Tno、Tsv | 16-bit contact timer (Unit: 10ms) |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tno | | | | | ○ | | | | | | | | | | | |
| Tsv | | | | | | | | ○ | | | | ○ | ○ | ○ | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tno | | | | | | | | | | | ● | | |
| Tsv | | ● | | | | ● | | | | | | | |

| Pulse instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| - | ES3/SV3/SX3 | - |

## Symbol




**Tno:** Timer number
**Tsv:** Setting value for the timer

## Explanation

1. The instruction is available for ES3 with firmware V1.06.00 and later and for SV3/SX3 with firmware V1.00.00 or later

2. The ATMRM instruction uses 10ms as the timing unit in the timer. Refer to the explanation of the ATMRH instruction (API 1018) for details

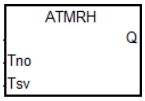
| API | Instruction code | Operand | Function |
|---|---|---|---|
| 1018 | ATMRH | Tno、Tsv | 16-bit contact timer (Unit: 1ms) |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tno | | | | | ○ | | | | | | | | | | | |
| Tsv | | | | | | | | ○ | | | | ○ | ○ | ○ | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tno | | | | | | | | | | | ● | | |
| Tsv | | ● | | | | ● | | | | | | | |

| Pulse instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| - | ES3/SV3/SX3 | - |

## Symbol




**Tno:** Timer number
**Tsv:** Setting value for the timer

## Explanation

1. The instruction is available for ES3 with firmware V1.06.00 and later and for SV3/SX3 with firmware V1.00.00 or later

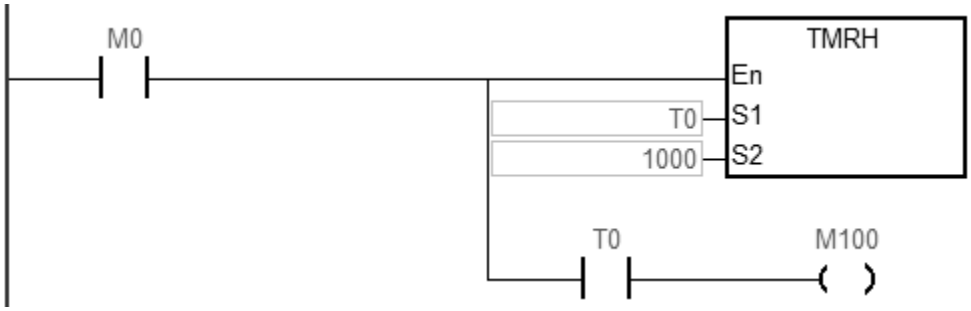
2. The ATMRH instruction is the same as the combination of AND and TMRH instructions.
   If the conditional contact is met, the coil for the specified timer is ON and the timer starts timing. When the specified timing value is reached (timing value >= setting value), its contact will act as the following table shows. If the condition for the AND contact action is met but the conditional contact is not met, the ATMRH instruction automatically clears the timing value in the timer.

| | |
|---|---|
| NO (Normally Open) contact | Continuity |
| NC (Normally Closed) contact | Discontinuity |

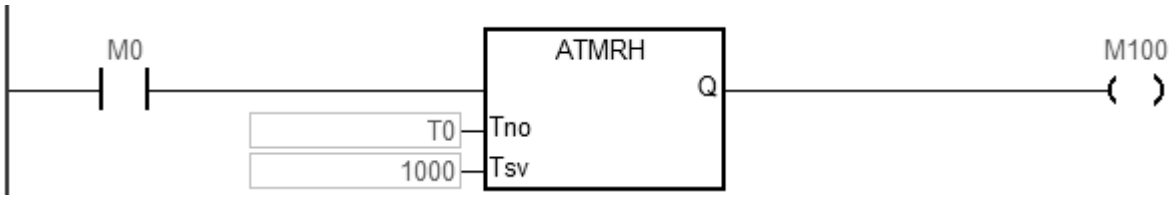
## Example

The program executes as the following shows:
When the normally open contact M0 is ON, T0 timer starts timing, when the timing value is greater than or equal to K1000, the normally open contact M100 is ON.

**Ladder diagram (Using TMRH instruction)**



**Ladder diagram (Using ATMRH instruction)**



## B-3_Errorlog

| Error Code | Description | Solution | Flag | Log |
|---|---|---|---|---|
| 16#0026 | RTC cannot keep track of the current time | If the problem persists, contact the local authorized distributors. | SM218 | ✓ |
| 16#0028 | No response from reading or writing RTC data | If the problem persists, contact the local authorized distributors. | SM217 | ✓ |
| 16#19B0 | Heartbeat timeout occurred in the slave of the CANopen communication module. | Check if the CANopen connection cable is well connected. | - | ✓ |
| 16#19B1 | The data length of PDO (process data object) in the slave mode is not matched with the setting. | Revise the PDO data length setting in the slave mode and download the setting again. | - | ✓ |
| Note: Check the values in SR830~SR893 to see which slave (1~64) experiences an error and refer to the following error codes 19E1~19E8 to check the details. | | | | |
| 16#19E1 | The data length of PDO (process data object) in the slave mode is not matched with the configuration in the scan list.<br>Refer to CANopen communication related descriptions in AS Series Hardware and Operation Manual for more details on the error codes 19E1 to 19E8. | Revise the PDO data length setting in the slave mode and download the setting again. | - | ✓ |
| 16#19E2 | PDO in the slave mode is not received. | Check if the configurations are correctly set. | - | ✓ |
| 16#19E3 | The function of auto downloading SDO fails at the first start-up. | Check if the SDO contents for auto downloading are correct. | - | ✓ |
| 16#19E4 | PDO configurations are set incorrectly. | Make sure to set the PDO configurations correctly. | - | ✓ |
| 16#19E5 | The main settings are not consistent with the ones set in connected slave. | Make sure the connected slaves are the ones configured in ISPSoft. | - | ✓ |
| 16#19E6 | The slave does NOT exist in the network. | Make sure the power supply of the slave is normal and the slave is correctly connected to the network. | - | ✓ |
| 16#19E7 | Timeout on the slave error control | Make sure the power supply of the slave is normal and the slave is correctly connected to the network. | - | ✓ |
| 16#19E8 | The node IDs of the master and slave are duplicated. | Set the node ID of the master or slave again and make sure their node IDs are unique. | - | ✓ |
| 16#19F3 | Error in the configuration | 1. Download the parameter configuration again<br>2. If the problem persists, contact the local authorized distributors | - | ✓ |
| 16#19F4 | CANopen communication is in the BUS-OFF state. | 1. Check if the start and end of the network cable are | - | ✓ |

| | | | | |
|---|---|---|---|---|
| | | both connected with a 121Ω terminal resistor<br>2. Check if all the node devices run at the same baud rate on the network | | |
| 16#19FB | Transmission of the to-be-sent data can NOT be complete during the set synchronization time. | Modify the synchronization time; suggested to prolong the time. | - | ✓ |
| 16#19FC | Transmission of the to-be-received data can NOT be complete during the set synchronization time. | Modify the synchronization time; suggested to prolong the time. | - | ✓ |

## B-6_PORT

### New Special Auxiliary Relay SM1092

| SM | Function | ES3 Series | SV3/SX2 Series | OFF ⇩ ON | STOP ⇩ RUN | RUN ⇩ STOP | Latched | Attribute | Default |
|---|---|---|---|---|---|---|---|---|---|
| SM1092 | Error flag for the COM port number of Modbus TCP slave Available for ES3 with firmware V1.04.30 and later and for SV3/SX3 with firmware V1.00.00 or later. | ○ | ○ | OFF | - | - | N | R/W | OFF |

### New Special Data Registers SR1092 & SR1093

| SR | Function | ES3 Series | SV3/SX2 Series | OFF ⇩ ON | STOP ⇩ RUN | RUN ⇩ STOP | Latched | Attribute | Default |
|---|---|---|---|---|---|---|---|---|---|
| SR1092 | COM port number of Modbus TCP slave (PLC acts as a slave) Available for ES3 with firmware V1.04.30 and later and for SV3/SX3 with firmware V1.00.00 or later. | ○ | ○ | 502 | - | - | N | R/W | 502 |
| SR1093 | COM port number of Modbus TCP for the data exchange table (PLC works as a master) Available for ES3 with firmware V1.04.30 and later and for SV3/SX3 with firmware V1.00.00 or later. | ○ | ○ | 502 | - | - | N | R/W | 502 |

### Additional Remarks

Special registers for storing the settings of data exchange via Ethernet communication port and a new special auxiliary relay is shown below. Available for ES3 with firmware V1.04.30 and later and for SV3/SX3 with firmware V1.00.00 or later.

| SM | Attribute | Description |
|---|---|---|
| SM1092 | R/W | Error flag for the COM port number of Modbus TCP slave |

| SR | Description |
|---|---|
| SR1092 | COM port number of Modbus TCP slave (PLC acts as a slave.) |
| SR1093 | COM port number of Modbus TCP for the data exchange table (PLC acts as a master.) |

### Explanation

1. When PLC CPU acts as Modbus TCP slave

    a. The default communication port number 502 is activated to connect to the remote master. According to the Modbus TCP specifications, the required default communication port number of the remote slave is 502

    b. If the target communication port of the remote master cannot use the port number 502, you should modify the value in SR1092 as the desired communication port number

    c. SR1092 is not a latched register and the value in the register will automatically change back to 502 once the power is ON again. Use ISPSoft to set the value in the PLC program

    d. If the communication port number set via SR1092 has been occupied by other functions (e.g. Socket or Web), the setting in SR1092 will not take effect and SM1092 will automatically change to ON

2. A data exchange table created for PLC CPU to use when it works as Modbus TCP master

    a. According to the Modbus TCP specifications, the required default communication port number of the remote slave is 502

    b. If the remote slave cannot use 502 as the communication port number, modify the value in SR1093 as the required communication port number

    c. SR1093 is not a latched register and the value in the register will automatically change back to 502 once the power is ON again. Use ISPSoft to set the value in the PLC program

    d. The setting value in SR1093 is the required communication port of the remote slave and it does not affect PLC CPU's own communication port

## C-3_REF

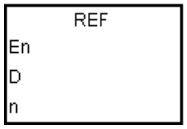
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 0600 | | REF | P | D、n | Refreshing the I/O |

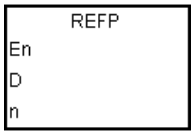
| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| D | ○ | ○ | | | | | | | | | | | | | | |
| n | | | | | | | | ● | ● | | ○ | ○ | ○ | ○ | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| D | ● | | | | | | | | | | | | |
| n | | ● | | | | ● | | | | | | | |

| Pulse instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| ES3/SV3/SX3 | ES3/SV3/SX3 | - |

## Symbol




**D:** Start device for I/O state refresh

**n:** Number of I/O points for state refresh

## Explanation

1. The I/O states are normally not refreshed until the PLC executes the END instruction. When the PLC starts scanning the program, it reads and stores the states of the external inputs in memory. After executing the END instruction, the PLC sends the states of the outputs in the memory to the output terminals. Therefore, when you need the latest I/O data during the operation process, you can use this instruction, or use the device DX/DY to refresh the input/output

2. The operand n must be a multiple of eight, e.g. 8, 16, 24 and so on. The maximum value is 256. If the value here is less than a multiple of eight, it will be seen as the next multiple of eight. For example, the value 20 will be seen as its next multiple of eight, 24

3. The number of the high-speed output point is stored in D device. If n is 1, it indicates to refresh the high-speed output value of the corresponding SR immediately. If n is 0, it indicates to stop high-speed output and refresh the SR current value
   For example, during the execution of this instruction, if n is 0 and the external interrupt input is received through X0, it indicates an external interrupt occurs in X0 and high-speed outputting through Y0 should be stopped immediately. The PLC sets the stop flag SM463 to ON and refresh the current corresponding output position in SR. Note: if the output completion auto-reset flag is set to ON, the PLC sets the output completion auto-reset flag to OFF and refresh the current corresponding output position in SR. But the PLC does not set the stop flag SM463 to ON.

4. Explanation of n and D operands

| Value in n | Device in D | Action Description |
|---|---|---|
| n = a multiple of 8 | X0 or Y0 | Refresh I/O immediately<br>See Example 1 and 2 |

| | | |
|---|---|---|
| n = 1 | High-speed output point | Refresh new pulse positon immediately<br>See Example 3 |
| n = 0 | High-speed output point<br>Without output completion auto-reset flag | Stop high-speed outputting, set the stop flag SM463 to ON and refresh the current corresponding output position.<br>Set the output completion auto-reset flag to OFF and refresh the current corresponding output position.<br>See Example 3 |
| n = -1*1 | Any X input point | Refresh the mapped area DS301 TxPDO (Master <= Slave) |
| n = -1*1 | Any Y output point | Refresh the mapped area DS301 RxPDO (Master => Slave) |
| n = -2*2 | Any X input point | Make a response to Modbus TCP command which is well received. (PLC is Server.) |

*1: This function is available for Master mode: ES3 with firmware V1.02.00 or later and SV3/SX3 with firmware V1.00.00 or later and for Slave mode: ES3 with firmware V1.06.00 or later and SV3/SX with firmware V1.00.00 or later. The function does not shorten the PDO data mapping time and so it is suggested to use this function when the PLC scan time is larger than the PDO refreshing time of DS301.
*2: This function is available for ES3 with firmware V1.06.00 or later and SV3/SX with firmware V1.00.00 or later. When the PLC program scan time is too long, using this function plus the time interrupt program, the PLC can regularly make a response to the Modbus TCP communication command which is received from the upper computer so as to speed up the communication between the upper computer and the PLC. It is suggested that the shortest time for an interrupt is 5ms. If the time for an interrupt is too short, it may slow down the speed of the PLC program scanning.

**Note:** The communication response function can process all data in real time, which means in one cycle of PLC program scan, reading or writing data may be interrupted.

## C-5_DTPWL

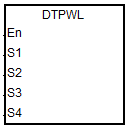
| API | Instruction code | | | Operand | Function |
|---|---|---|---|---|---|
| 2720 | D | TPWL | P | $S_1$、$S_2$、$S_3$、$S_4$ | Setting linear interpolation parameters in the position planning table |

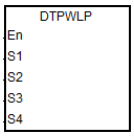
| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S_1$ | | | | | | | | ● | ● | | | | ○ | ○ | | |
| $S_2$ | | | | | | | | ● | ● | | | | ○ | ○ | | |
| $S_3$ | | | | | | | | ● | ● | | | | ○ | ○ | | |
| $S_4$ | | | | | | | | ● | ● | | | | ○ | ○ | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S_1$ | | | ● | | | | ● | | | | | | |
| $S_2$ | | | ● | | | | ● | | | | | | |
| $S_3$ | | | ● | | | | ● | | | | | | |
| $S_4$ | | | ● | | | | ● | | | | | | |

| Pulse instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| ES3/SV3/SX3 | - | ES3/SV3/SX3 |

### Symbol




**$S_1$:** A number listed in the position planning table
**$S_2$:** Target position of the X axis
**$S_3$:** Target position of the Y axis
**$S_4$:** Target speed

### Explanation

1. This instruction sets the 2-axis linear interpolation parameters in the position planning table. S1 is the number listed in the position planning table. If the number does not exist in the table or the output of the number does not belong to the 2-axis linear interpolation, the instruction is not executed, SM0 is ON and the error code is SR0=16#2027

2. S2 and S3 are respectively the target positions of the X and Y axes, which can only be 32-bit integers. If you use the mechanical unit conversion when editing the position planning table in ISPSoft, use the conversion instruction for modification first

3. S4 is the target speed. The range is between 1~200,000Hz. (Note: if the setting value is out of the range, the instruction automatically changes the setting into the minimum or maximum value.)

4. When the instruction executes the linear interpolation, the target frequency S4 automatically corresponds to the output of the axis which is farthest from its target position. If X axis and Y axis cannot simultaneously reach the target positions, the PLC automatically decelerates the frequency to make the two axes reach the target positions simultaneously

5. When the instruction is executed to modify parameters for the two axes (either of which is outputting), the modified parameters of the two axes are kept in the table and are not effective until the next 2-axis output starts

6. The parameters modified by the instruction can be modified only while the PLC is running. The last written parameter is not saved when the power turns OFF. The table that you edit in ISPSoft and download to the PLC is processed as the default position planning table when the power is ON

7. To modify the acceleration/deceleration time, you should modify the acceleration/deceleration time in SR that the specified output axis of X axis corresponds to, set the flag SM585 to ON, and then execute this instruction. When changing relevant parameters is complete, SM585 will be automatically cleared and change to OFF
   For example, if Y0 output axis is designated for X axis, you should modify the value in SR464 (acceleration time of Y0 output) to modify the acceleration/deceleration time. For details on operation steps, please refer to Example 1 of the TPWS instruction.
   Note:

   a. The acceleration and deceleration time is only for the PLC operation of the 2-axis synchronized motion. If the simultaneous arrival to the target positions cannot be achieved after calculation, the PLC will automatically execute the instruction with the most proper acceleration/deceleration time and no error information will occur

   b. The function is available for ES3 with firmware V1.06.00 or later and SV3/SX3 with firmware V1.00.00 or later

8. During the execution of TPO instruction, if you need to modify parameters through this instruction, it is suggested to avoid modifying the used number for the output that is being performed or is to be performed soon

## C-6_DZRN_DZRN2

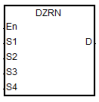
| API | Instruction | Operand | Description |
|---|---|---|---|
| 2704 | D ZRN | $S_1$、$S_2$、$S_3$、$S_4$、D | Zero return |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S_1$ | | | | | | | ● | ● | ● | | ○ | | ○ | ○ | | |
| $S_2$ | | | | | | | ● | ● | ● | | ○ | | ○ | ○ | | |
| $S_3$ | | | | | | | ● | ● | ● | | ○ | | ○ | ○ | | |
| $S_4$ | ○ | | | | | | | | | | | | | | | |
| D | | ○ | | | | | | | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S_1$ | | | ● | | | | ● | | | | | | |
| $S_2$ | | | ● | | | | ● | | | | | | |
| $S_3$ | | | ● | | | | ● | | | | | | |
| $S_4$ | ● | | | | | | | | | | | | |
| D | ● | | | | | | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| - | - | ES3/SV3/SX3 |

## Symbol




**$S_1$:** Target frequency for zero return
**$S_2$:** JOG frequency for DOG
**$S_3$:** Zero return mode
**$S_4$:** Input device for DOG
**D:** Pulse output device

## Explanation

1. This instruction causes the machine to return to the zero point. The range of the target frequency for zero return S1 is between 1 Hz–200 kHz. The JOG frequency S2 should be less than the target frequency S1. The JOG frequency S2 is the start frequency. If S1 is less than S2, S1 is automatically revised processed as equal to S2

2. The input point for S4 and output point for D must match. Do not change them during instruction execution. The input point for S4 is suggested to use the 16 high-speed input points X0–X7 and X10–X17. They will not be affected by PLC instruction scan time. If you use X20 successive input points or M devices, they will be affected by the PLC instruction scan time. Refer to the following table for the selection of D output point and direction output point. If D is not the preset Pulse+direction output (default: 0), change the mode to A/B phase output by setting SR to 1

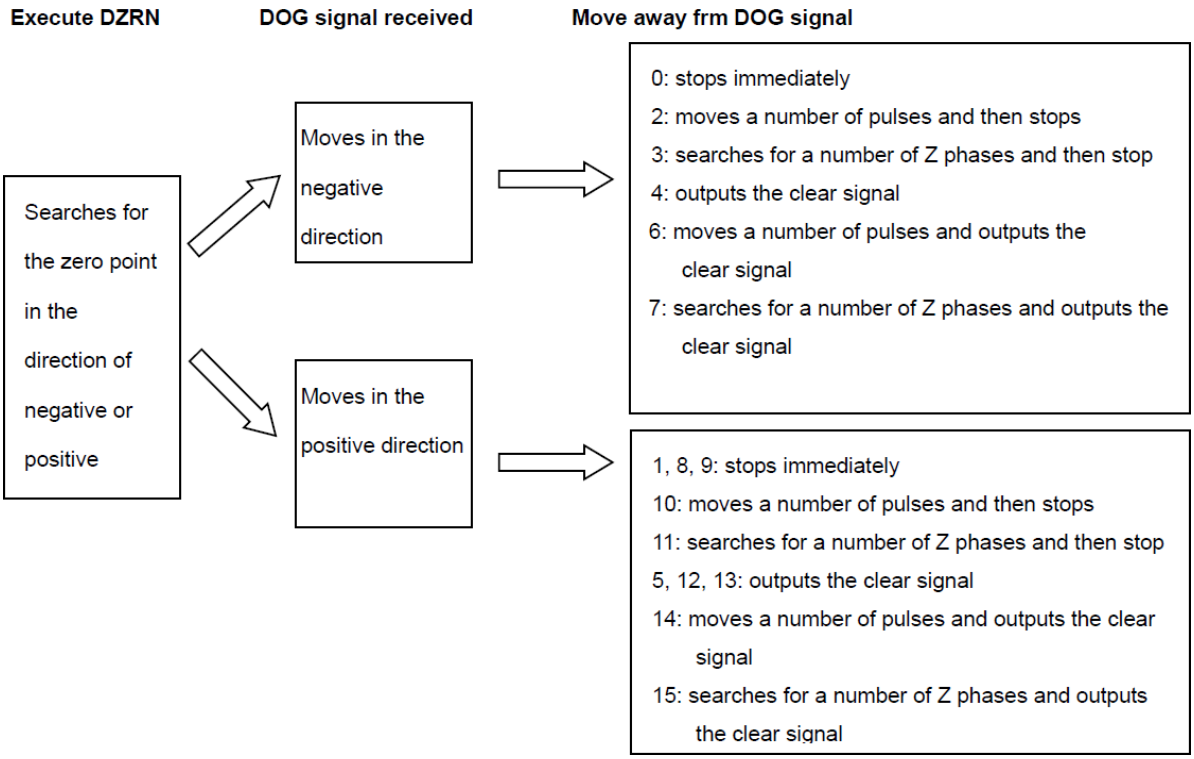
| Axis Number | Axis 1 | Axis 2 | Axis 3 | Axis 4 |
|---|---|---|---|---|
| Input point for **S4** | Can be any one of the input points X0–X7 and X10–X17. But the same input point cannot be selected for different axis output. If the DOG point shakes or the switch bounces, set the input point filter time in HWCONFIG. | | | |
| Output point for **D** | Y0 | Y2 | Y4 | Y6 |

| Direction output point | Y1 | Y3 | Y5 | Y7 |
|---|---|---|---|---|
| Output mode | SR462 | SR482 | SR502 | SR522 |
| Buys flag | SM460 | SM480 | SM500 | SM520 |
| Completion flag | SM461 | SM481 | SM501 | SM521 |
| Present output position | SR460<br>SR461 | SR480<br>SR481 | SR500<br>SR501 | SR520<br>SR521 |

3. Use S3 to select the zero return mode. The function code is set by the two high and low16-bit parameters. See the following table for details.

| **S3: select the zero return mode** | | | | |
|---|---|---|---|---|
| **High 16-bit** | **Low 16-bit** | | | |
| **b31~b16** | **b15~b6** | **b5** | **b4** | **b3~b0** |
| Number of pulses for motion | Reserved | Direction setting*1<br>0: in the negative direction<br>1: in the positive direction | Setting DOG signal<br>0: contact A<br>1: contact B | Mode setting<br>0~15 (F) |
| Number of Z phases | | | | |

See the diagram below for mode setting




4. Use S3 to select the zero return mode. The function code is set by the two high and low16-bit parameters. See the following table for details

| **Function** | **Code** | | **Explanation** |
|---|---|---|---|
| | **High 16-bit** | **Low 16-bit** | |
| Leaves the zero point in the negative direction and then stops (Mode 0) | 0 | 0 | When the instruction is executed, the search for the zero point is in the negative direction with the target frequency. When the zero point is ON |

| | | | |
|---|---|---|---|
| | | | (the zero point signal changes from OFF to ON), the frequency is decreased to the JOG speed and the motion in the negative direction continues, and does not stop until the zero point signal changes from ON to OFF. |
| Leaves the zero point in the positive direction and then stops (Mode 1) | 0 | 1 | When the instruction is executed, the search for the zero point is in the negative direction with the target frequency. When the zero point is ON (the zero point signal changes from OFF to ON), the frequency decreases to 0 immediately, and then the motion is in the positive direction at the JOG speed, and does not stop until the zero point signal changes from ON to OFF. |
| Mode 0 Moves again after returning to the zero point | Number of pulses for motion | 2 | Returning to the zero point is the same as that for the low 16-bit code. After the zero point is ON, the motion continues according to the number of specified pulses. When the high 16-bit code is a positive number, the search is in the positive direction. A negative value means that the search is in the negative direction. |
| Mode 0 Searches Z phase after returning to the zero point (Z phase input point is set in HWCONFIG ) | Number of Z phases | 3 | Returning to the zero point is the same as that for the low 16-bit code. After returning to the zero point, the motion continues according to the number of Z phases. When the high 16-bit code is a positive number, the search is in the positive direction. A negative value indicates that the search is in the negative direction. Suppose you specified that the rising-edge trigger of X0 as the condition for the Z phase input in HWCONFIG. The counting is performed once whenever the rising-edge trigger for X0 occurs. |
| Mode 0 Outputs the clear signal after returning to the zero point. (Output clear point is set in HWCONFIG ) | Number of pulses for motion or number of Z phases | 4+0=4<br>4+1=5<br>4+2=6<br>4+3=7<br>(bit 2=ON) | Choosing a value between 4–7 means selecting the functions codes 0–3 respectively, and the specified output point sends an ON signal that is about 20ms wide when the function execution completes. The range of the output point is Y14–Y17 and Y20–Y27. For example, if you specify Y22 as the output point in HWCONFIG, it indicates Y22 is for the output of clear signals. |
| Leaves the zero point in the positive direction and then stops (Mode 1) | 0 | 8+0=8<br>8+1=9<br>(bit 3=ON) | The operation for zero point return is the same as that for code 1 (mode 1). |
| Mode 1 outputs the number of pulses after returning to the zero point | Number of pulses for motion | 8+2=10<br>(bit 3=ON) | The operation for zero point return is the same as that for low 16-bit code 1. After returning to the zero point, the motion continues in accordance with the number of specified pulses. When the |

| | | | |
|---|---|---|---|
| | | | value of the high 16-bit code is a positive number, the motion is in the positive direction. A negative number indicates that the motion is in the negative direction. |
| Mode 1 Searches for Z phase after returning to the zero point<br>(Z phase input point is set in HWCONFIG ) | Number of Z phases | 8+3=11<br>(bit 3=ON) | The operation for zero point return is the same as that for low 16-bit code 1.<br>After returning to the zero point, the motion continues in accordance with the number of Z phases to seek. When the value of the high 16-bit code is a positive number, the motion is in the positive direction. A negative number indicates that the motion is in the negative direction.<br>If the rising edge trigger for X1 is the condition for Z phase input, counting is performed once when the rising-edge trigger for X1 occurs. |
| Mode 1 Outputs the clear signal after returning to the zero point<br>(Output clear point is set in HWCONFIG ) | 0 or number of pulses or number of Z phases | 12–15<br>(bit 3=bit 2 =ON) | After returning to the zero point in mode 1, the 20ms-width clear signal is output. |
| DOG point is B point | | +16<br>(bit 4=ON) | When in the low 16-bit code, bit 4 is ON, it means the zero point is ON as the DOG point changes from ON to OFF and the zero point is left as the DOG point changes from OFF to ON. |

5. The execution sequence is based on the value of the low 16-bit code in the table above, and described below

   → The direction of DOG signal is determined by the value of bit 5

   → The DOG signal is determined by the value of bit 4

   → Mode 0 or mode 1 for the zero point return, selected according to the value of bit 3

   → The operation of the zero point return is performed according to the values of bit 1 and bit 0

   → The operation of the clear signal specified by bit 2 is performed

6. Set the input point and the rising/falling edge trigger condition in HWCONFIG, when the position control system needs positive and negative limit input points. Note that the limit input points must not be the same as the zero point or Z phase input points
   For ES3 with firmware V1.04 or earlier, when the zero return motion is in process and the limit point check function is activated, the limit alarm flag is not set to ON once the limit point is reached.
   For ES3 with firmware V1.06 or later and SV3/SX3 with firmware V1.00.00 or later, the limit alarm flag will automatically set to ON for a reminder, and then you can decide the follow-up operation based on the alarm flag. (Note: After the limit alarm flag is ON, it must be cleared by yourself.)
7. The Completion flag is set to ON after the instruction finishes performing the specified function. For example, for function code 6, the PLC sets the Completion flag to ON only when the Z phases seeking completes

8. After the DZRN instruction is executed, an interrupt service program is not executed till the DZRN instruction is disabled, if the specified input point for the zero point is the same as that for the

external input interrupt in the program

9. When the limit switch is specified in HWCONFIG, and there is an external input interrupt service program, the interrupt program will be executed at the same time

10. It is not suggested to use this instruction in the ST programming language, interrupt tasks or function block which is called only once
For ES3 with firmware V1.06 or later and SV3/SX3 with firmware V1.00.00 or later, the ST programming language is supported. For details, refer to the description in the example

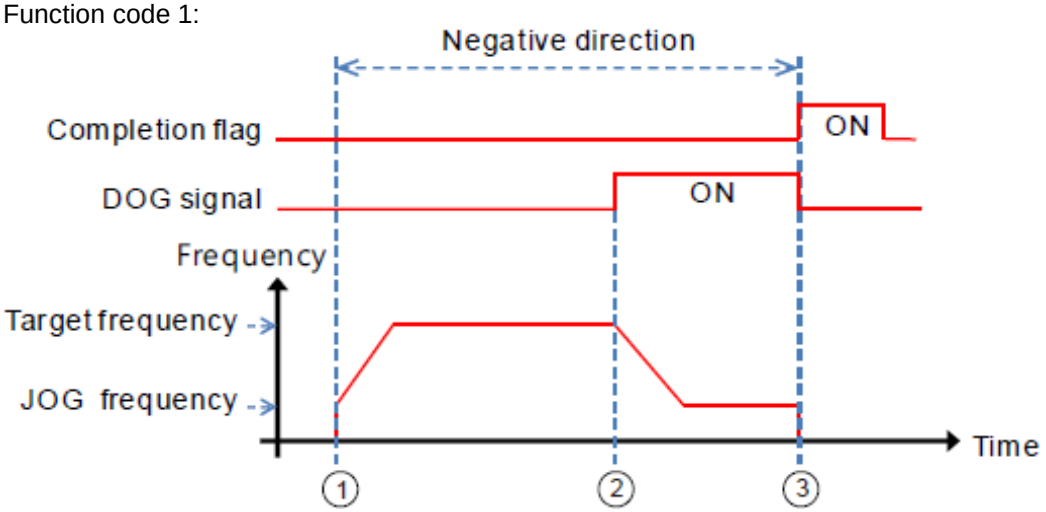
11. The steps for performing the functions are as below
Function code 0:

① The DZRN function is executed and the search for the zero point is in the negative direction with the target frequency.

② After the DOG signal is received, the output frequency decreases to the JOG frequency. The output continues in the negative direction and does not stop until the zero point signal changes from ON to OFF.

③ The output stops when the signal changes from ON to OFF and the axis moves away from the DOG signal.

Function code 1:




① The DZRN function is executed and the search for the zero point is in the negative direction with the target frequency.

② After the DOG signal is received, the output is in the positive direction with the JOG frequency after the output frequency decreases, and the motion direction reverses. The output does not stop until the zero point signal changes from ON to OFF.

③ The axis moves away from the DOG signal and PLC stops when the signal changes from ON to OFF.

Function code 2:

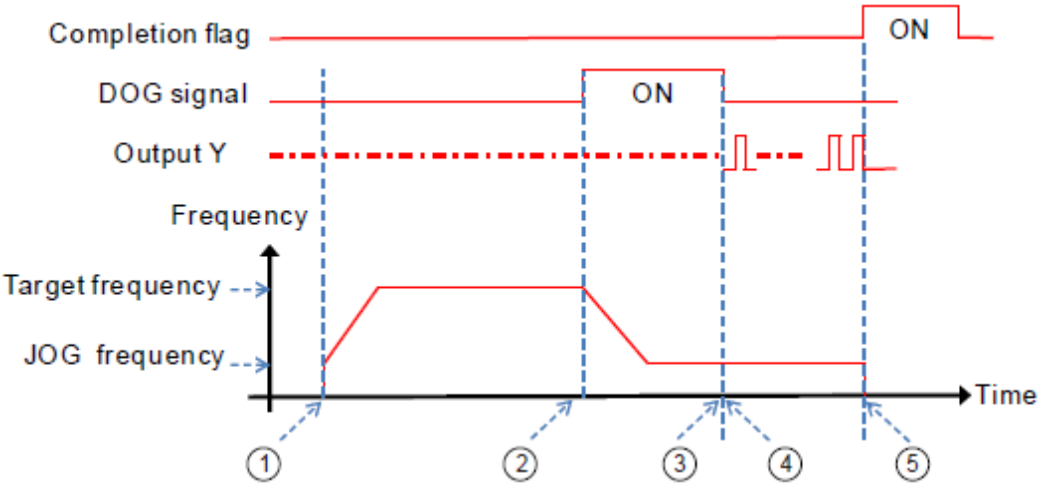
① The DZRN function is executed and the search for the zero point is in the negative direction with the target frequency.

② After the DOG signal is received, the output decreases the frequency to the JOG frequency and continues in the negative direction.

③ When the DOG signal is left and the signal changes from ON to OFF, the specified number of pulses are output.

④ The first pulse output starts.

⑤ When the 100th pulse output completes, the PLC stops and the Completion flag is ON.




Function code 3:

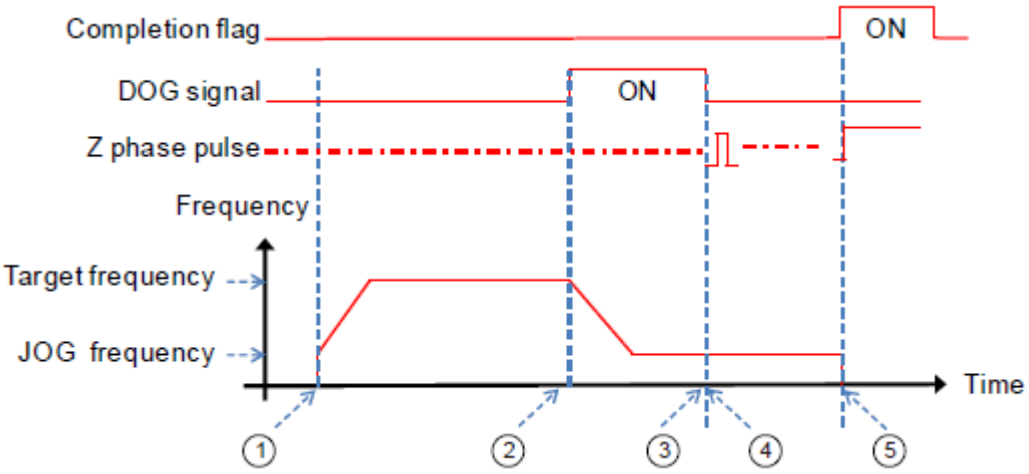
① DZRN function is executed and the search for the zero point is in the negative direction with the target frequency.

② After the DOG signal is received, the output frequency decreases to the JOG frequency and the motion continues in the negative direction.

③ The motion goes on according to the number of Z phases when the signal changes from ON to OFF after the axis moves away from the DOG signal.

④ The first Z phase pulse

⑤ The motion stops after the 2nd Z phase completes and the Completion flag is ON.



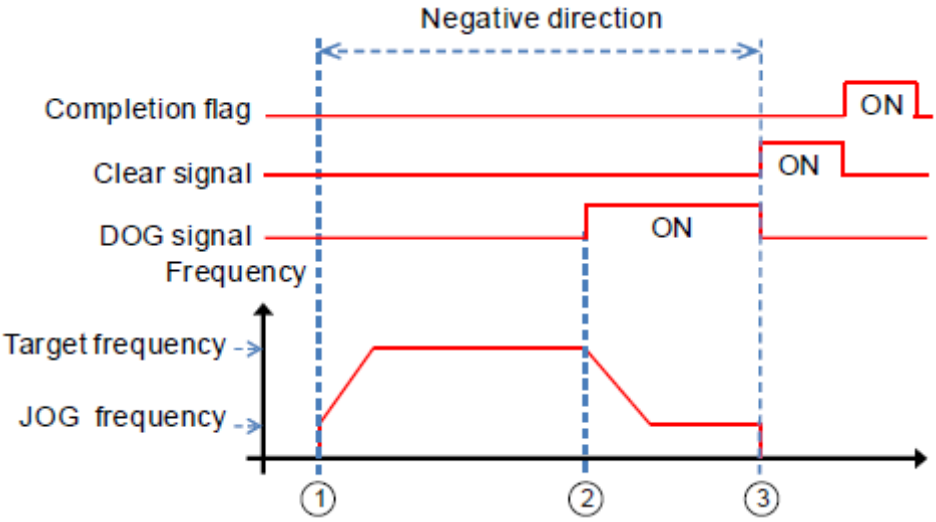

Function code 4:

① DZRN function is executed and the search for the zero point is in the negative direction with the

target frequency.

② After the DOG signal is received, the output frequency decreases to the JOG frequency and continues in the negative direction. The output does not stop until the zero point signal changes from ON to OFF.

③ After the axis moves away from the DOG signal, the output stops when the signal changes from ON to OFF and the clear signal is ON for about 20 milliseconds.




Function code 0+ the negative limit function enabling:

⓪ Set the negative limit input point in HWCONFIG, and then download the setting to the PLC. The PLC automatically calculates the negative limit function when the instruction is executed.

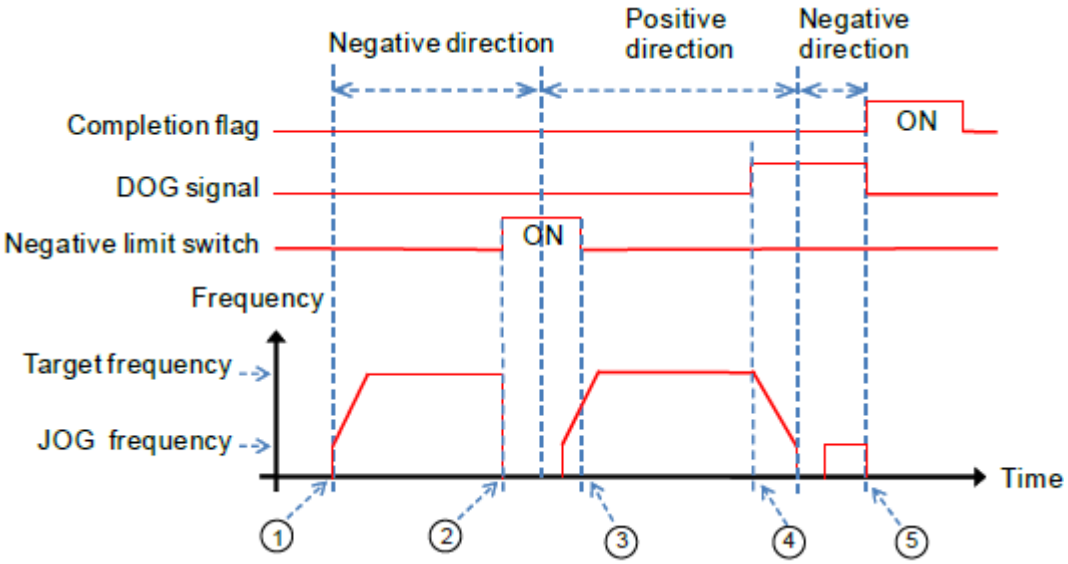
① DZRN function is executed and the search for the zero point is in the negative direction with the target frequency.

② After the negative limit switch is ON, the motion stops and then goes in the positive direction after reversing direction.

③ The motion continues in the positive direction after leaving the negative limit switch.

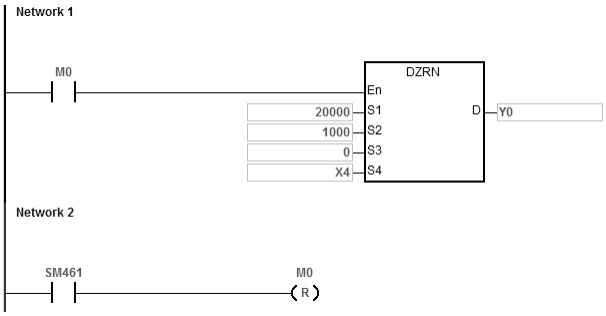
④ The output frequency ramps down after receiving the DOG signal. The reverse output is performed with the JOG frequency after reversing direction.

⑤ The output stops when the signal changes from ON to OFF after the axis moves away from the DOG signal.




## Example 1

When M0 is ON, outputting the pulse from Y0 with a frequency of 20 kHz to search for the zero point in the negative direction. When the DOG signal is received and X4 is ON, it keeps moving in the negative direction with the JOG frequency of 1 kHz. The output stops immediately after X4 changes from ON to OFF.




## Explanation

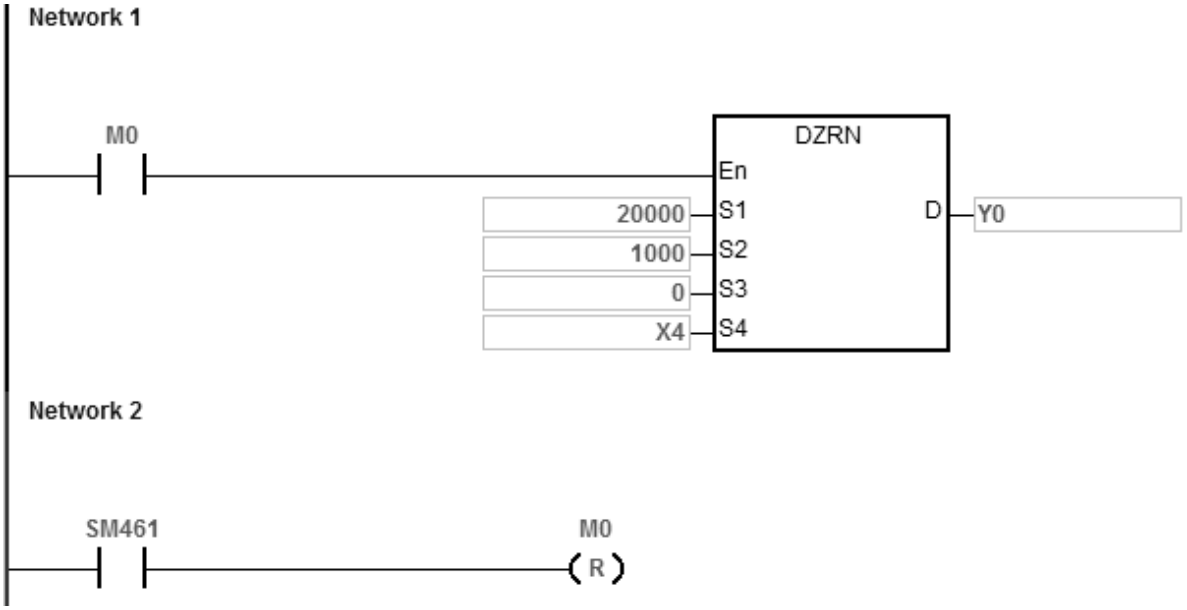
S3 is set to 0.

High 16-bit [0000] is to disable the function to move a number of pulses or to search for Z phases.

Low 16-bit [0000] is Mode 0; when the DOG signal is received, the axis moves in the negative direction; after the axis moves away from the DOG signal, it stops immediately.

## Example 2

When M0 is ON, outputting the pulse from Y0 with a frequency of 20 kHz to search for the zero point in the negative direction. When the DOG signal is received and X4 is ON, the PLC decreases the frequency to the JOG frequency of 1 kHz and keeps moving in the positive direction with the JOG frequency of 1 kHz. When X4 is OFF, the PLC starts to seek the Z phase pulse in the positive direction. When X5 receives two pulses, the PLC stops and Y14 outputs a 20ms-width pulse.




## Explanation

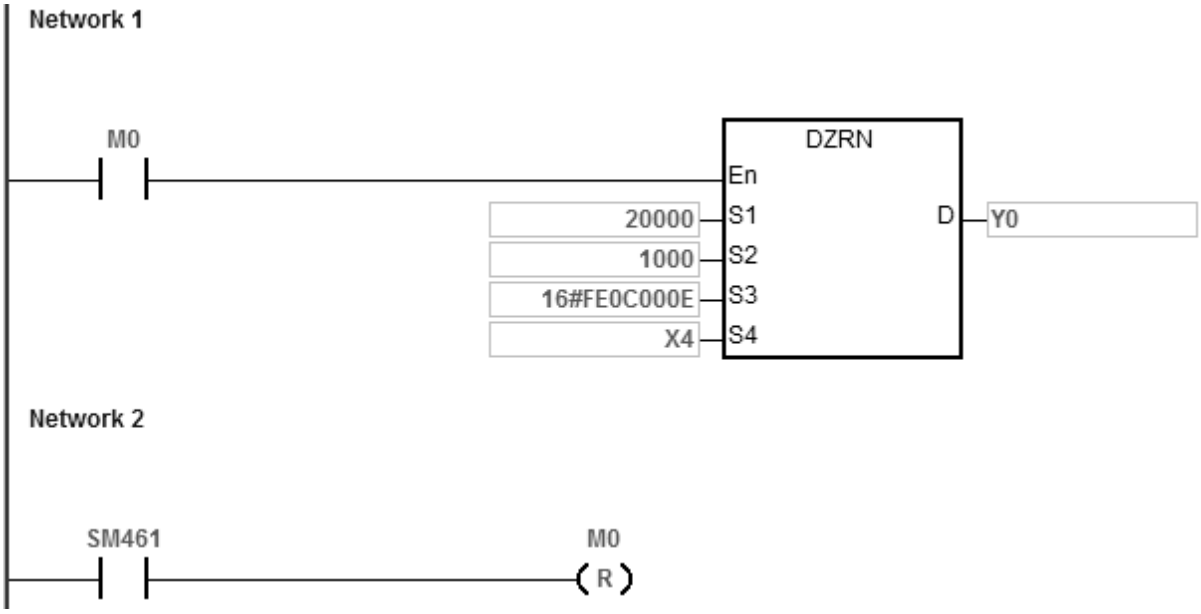
1. If you specify the rising-edge trigger at X5 as the condition for Z phase input in HWCONFIG, the count is incremented once whenever the rising-edge trigger at X5 occurs

2. Y14 is specified as the output point for outputting the clear signal in HWCONFIG

3. S3 is set as 16#00020007. High 16-bit [0002] is to search for the Z phase twice, once the axis moves away from the DOG signal.
   Low 16-bit [0007] is Mode 7; when the DOG signal is received, the axis moves in the negative direction; after the axis moves away from the DOG signal, and found the Z phase, a clear signal is outputted (20ms).

## Example 3

When M0 is ON, outputting the pulse from Y0 with a frequency of 20 kHz to search for the zero point in the negative direction. When the DOG signal is received and X4 is ON, the PLC decreases the frequency to the JOG frequency of 1 kHz and keeps moving in the positive direction with the JOG frequency of 1 kHz. When X4 is OFF, the axis starts to move after 500 pulse output completes in the negative direction. Y14 outputs a 20ms-width pulse and then stops outputting.




### Explanation

1. Y14 is specified as the output point for outputting the clear signal in HWCONFIG

2. S3 is set as 16#FE0C000E. High 16-bit [FE0C=-500] once the axis moves away from the DOG signal. The axis starts to move after 500 pulse output completes in the negative direction. Low 16-bit [000E] is Mode 14; when the DOG signal is received, the axis moves in the positive direction; after the axis moves away from the DOG signal, a clear signal is outputted (20ms).

### Example 4 (in ST programming language)

Generally, it is suggested to use the ladder diagram language to edit the program when using this instruction. If you use ST language (which is only available for ES3 with firmware V1.06.00 or later and SV3/SX3 with firmware V1.00.00), you must wait until the zero-return motion is completed (SM461 completion flag is ON) and then disable the instruction. Besides, you must use the auto reset flag together to notify the PLC to release the high-speed output control right.

Taking Y0 output for example, the program writing is shown in the figure below

```
0001 IF M0 THEN
0002     DZRN(1000,100,0,X0,Y0);
0003     M1:=TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
```

## C-6_DZRN_DZRN2

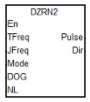
| API | Instruction | Operand | Description |
|---|---|---|---|
| 2724 | D ZRN2 | TFreq、JFreq、Mode、DOG、NL、Pulse、Dir | Zero return 2 (directional output can be defined) |

| Device | X | Y | M | S | T | C | HC | D | FR | SM | SR | E | K | 16# | "$" | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TFreq | | | | | | | ● | ● | ● | | ○ | | ○ | ○ | | |
| JFreq | | | | | | | ● | ● | ● | | ○ | | ○ | ○ | | |
| Mode | | | | | | | ● | ● | ● | | ○ | | ○ | ○ | | |
| DOG | ○ | | ○ | | | | | | | | | | | | | |
| NL | ○ | | ○ | | | | | | | | | | | | | |
| Pulse | | ○ | | | | | | | | | | | | | | |
| Dir | | ○ | ○ | | | | | | | | | | | | | |

| Data type | BOOL | WORD | DWORD | LWORD | UINT | INT | DINT | LINT | REAL | LREAL | TMR | CNT | STRING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TFreq | | | ● | | | | ● | | | | | | |
| JFreq | | | ● | | | | ● | | | | | | |
| Mode | | | ● | | | | ● | | | | | | |
| DOG | ● | | | | | | | | | | | | |
| NL | ● | | | | | | | | | | | | |
| Pulse | ● | | | | | | | | | | | | |
| Dir | ● | | | | | | | | | | | | |

| Pulse Instruction | 16-bit instruction | 32-bit instruction |
|---|---|---|
| - | - | ES3/SV3/SX3 |

## Symbol




**TFreq:** Target frequency for zero return
**JFreq :** JOG frequency for DOG
**Mode:** Zero return mode
**DOG:** Input device for DOG
**NL:** Input device for Negative Limit point
**Pulse:** Pulse output device
**Dir:** Output direction device

## Explanation

1. This instruction causes the machine to return to the zero point. The range of the target frequency (TFreq) for zero return is between 1 Hz–200 kHz. The JOG frequency (JFreq) should be less than the target frequency (TFreq). The JOG frequency (JFreq) is the start frequency. If the target frequency (TFreq) is less than (JFreq), the target frequency (TFreq) is automatically revised processed as equal to (JFreq)

2. Do not change the operands DOG, NL, Pulse, or Dir during instruction execution. The input point for DOG and NL is suggested to use the 16 high-speed input points X0–X7 and X10–X17. They will not be affected by PLC instruction scan time. If you use X20 successive input points or M devices, they will be affected by the PLC instruction scan time

3. See the below descriptions for DOG, NL and Dir

| | |
|---|---|
| DOG point | You can use X or M device but do NOT choose the same input point for different axis.<br>When the DOG point uses X0–X7 and X10–X17 for inputs, go to HWCONFIG to set up the filtering time for the inputs if the switch starts to rattle. |
| NL point | You can use X or M device but do NOT choose the same input point for different axis. |
| Dir contact | You can use Y or M device but do NOT choose the same input point for different axis. |

For ES3 with firmware V1.04 or earlier, when the zero return is in process and the limit point check function is activated, the limit alarm flag is not set to ON once the limit point is reached.
For firmware ES3 with firmware V1.06.00 or later and SV3/SX3 with firmware V1.00.00, the limit alarm flag will automatically set to ON for a reminder, and then you can decide the follow-up operation based on the alarm flag. (After the alarm flag is ON, it must be cleared by yourself.)
Note: DZRN2 does not support the positive limit function.

4. See the below SR and SM table for pulse outputting.

| Pulse Output Points | Y0 | Y2 | Y4 | Y6 |
|---|---|---|---|---|
| Setting up the time for directional outputting goes first before pulse outputting (unit: 1 ms) | SR640 | SR642 | SR644 | SR646 |
| Busy flag | SM460 | SM480 | SM500 | SM520 |
| Completion flag | SM461 | SM481 | SM501 | SM521 |
| Present output position | SR460<br>SR461 | SR480<br>SR481 | SR500<br>SR501 | SR520<br>SR521 |

| Pulse Output Points | Y1 | Y3 | Y5 | Y7 |
|---|---|---|---|---|
| Setting up the time for directional outputting goes first before pulse outputting (unit: 1 ms) | SR641 | SR643 | SR645 | SR647 |
| Busy flag | SM472 | SM492 | SM512 | SM532 |
| Completion flag | SM473 | SM493 | SM513 | SM533 |
| Present output position | SR474<br>SR475 | SR494<br>SR495 | SR514<br>SR515 | SR534<br>SR535 |

*1: Not available for SX3 series PLC CPU

5. Select the zero return mode. The function code is set by the two high and low16-bit parameters. See the following table for details

| Select the zero return mode | | | | | |
|---|---|---|---|---|---|
| High 16-bit | Low 16-bit | | | | |
| b31~b16 | b15~b7 | b6 | b5 | b4 | b3~b0 |

| | | | | | |
|---|---|---|---|---|---|
| Number of pulses for motion | Reserved | Setting DOG signal mask<br>0: disabled<br>1: enabled | Setting NL signal<br>0: contact A<br>1: contact B | Setting DOG signal<br>0: contact A<br>1: contact B | Mode setting<br>0~2;<br>8~10 |

6. The function code is set by the two high and low 16-bit parameters. See the following table for details

| Function | Code | | Description |
|---|---|---|---|
| | **High 16-bit** | **Low 16-bit** | |
| Leaves the zero point in the negative direction and then stops<br>(Mode 0) | **0** | **0** | When the instruction is executed, the search for the zero point is in the negative direction with the target frequency. When the zero point is ON (the zero point signal changes from OFF to ON), the frequency is decreased to the JOG speed and the motion in the negative direction continues, and does not stop until the zero point signal changes from ON to OFF. |
| Leaves the zero point in the positive direction and then stops<br>(Mode 1) | **0** | **1** | When the instruction is executed, the search for the zero point is in the negative direction with the target frequency. When the zero point is ON (the zero point signal changes from OFF to ON), the frequency decreases to 0 immediately, and then the motion is in the positive direction at the JOG speed, and does not stop until the zero point signal changes from ON to OFF. |
| Mode 0 Moves again after returning to the zero point | Number of pulses for motion | **2** | Returning to the zero point is the same as that for the low 16-bit code. After the zero point is ON, the motion continues according to the number of specified pulses. When the high 16-bit code is a positive number, the search is |

| | | | |
|---|---|---|---|
| | | | in the positive direction. A negative value means that the search is in the negative direction. |
| Leaves the zero point in the positive direction and then stops (Mode 1) | 0 | 8+0=8<br>8+1=9<br>(bit3=ON) | Refer to mode 1. The operation for zero point return is the same as that for code 1 (mode 1). |
| Mode 1 outputs the number of pulses after returning to the zero point | Number of pulses for motion | 8+2=10<br>(bit3=ON) | The operation for zero point return is the same as that for low 16-bit code 1.<br>After returning to the zero point, the motion continues in accordance with the number of specified pulses. When the value of the high 16-bit code is a positive number, the motion is in the positive direction. A negative number indicates that the motion is in the negative direction. |
| DOG point is B point | | +16<br>(bit4=ON) | When in the low 16-bit code, bit 4 is ON, it means the zero point is ON as the DOG point changes from ON to OFF and the zero point is left as the DOG point changes from OFF to ON. |
| NL point is B point | | +32<br>(bit5=ON) | When in the low 16-bit code, bit 5 is ON, it means the zero point is ON as the NL point changes from ON to OFF and the zero point is left as the DOG point changes from OFF to ON. |

7. The execution sequence is based on the value of the low 16-bit code in the table above, and described below

→ The direction of DOG signal is determined by the value of bit 5

→ The NL negative limit and DOG signal are determined by the value of bit 5 and bit 4

→ Mode 0 or mode 1 for the zero point return, selected according to the value of bit 3

→ The operation of the zero point return is performed according to the values of bit 1 and bit 0

8. The DOG signal mask function is only available for ES3 with firmware V1.06.00 or later and SV3/SX3 with firmware V1.00.00 or later. When the mask function is enabled (setting bit6 to 1), Mode+1 is the number of DOG signals to be masked. When the value is less than or equal to 0, the DOG point mask function is regarded as disabled
   Task section is determined by the number of output pulses (regardless of the direction positive or negative) after the DZRN2 instruction is executed. Since the start position for the search for the zero position may be different each time, the setting of the mask section may lead to the case that the zero position is missed. So be sure to evaluate whether to set the mask function or set the negative limit function based on the actual application
   For example, set Mode to D0=64 (bit6 set to 1) and set D2=120000 (the number of pulses to be masked). After DZRN2 is executed, the DOG signals for the number of output pulses less than 120,000 will all be masked and thus the DOG signals are not functioning

9. The Completion flag is set to ON after the instruction finishes performing the specified function. For example, for function code 2, the PLC sets the Completion flag to ON only when the specified number of pulses completes outputting

10. After the DZRN2 instruction is executed, an interrupt service program is not executed till the DZRN2 instruction is disabled, if the specified input point for the zero point is the same as that for the external input interrupt in the program

11. It is not suggested to use the instruction in the ST programming language, interrupt tasks or function block which is called only once

12. ST programming language is supported for ES3 with firmware V1.06.00 or later and SV3/SX3 with firmware V1.00.00 or later. For details, refer to the description in the example

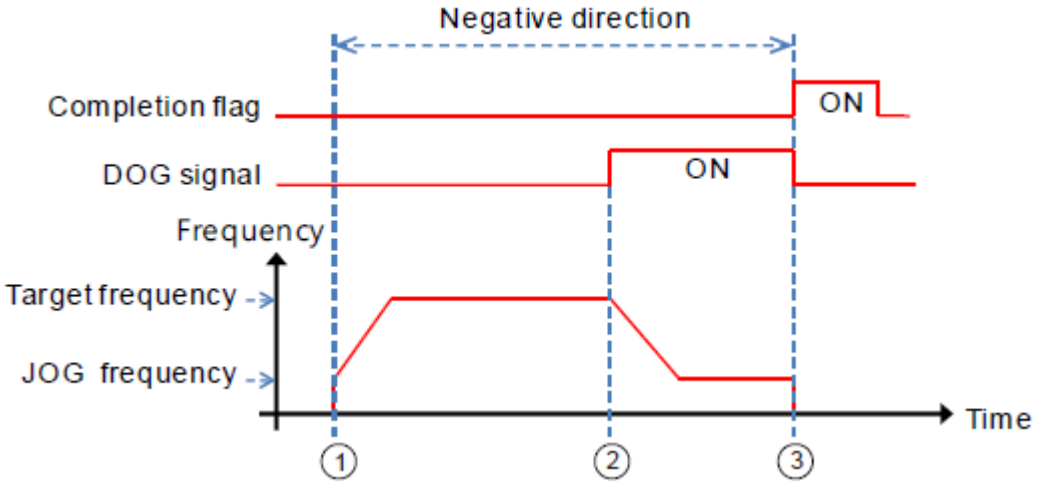
13. The steps for performing the functions are as below

    Function code 0:

    ① The DZRN2 function is executed and the search for the zero point is in the negative direction with the target frequency TFreq

    ② After the DOG signal is received, the output frequency decreases to the JOG frequency JFreq. The output continues in the negative direction and does not stop until the zero point signal changes from ON to OFF

    ③ The output stops when the signal changes from ON to OFF and the axis moves away from the DOG signal.

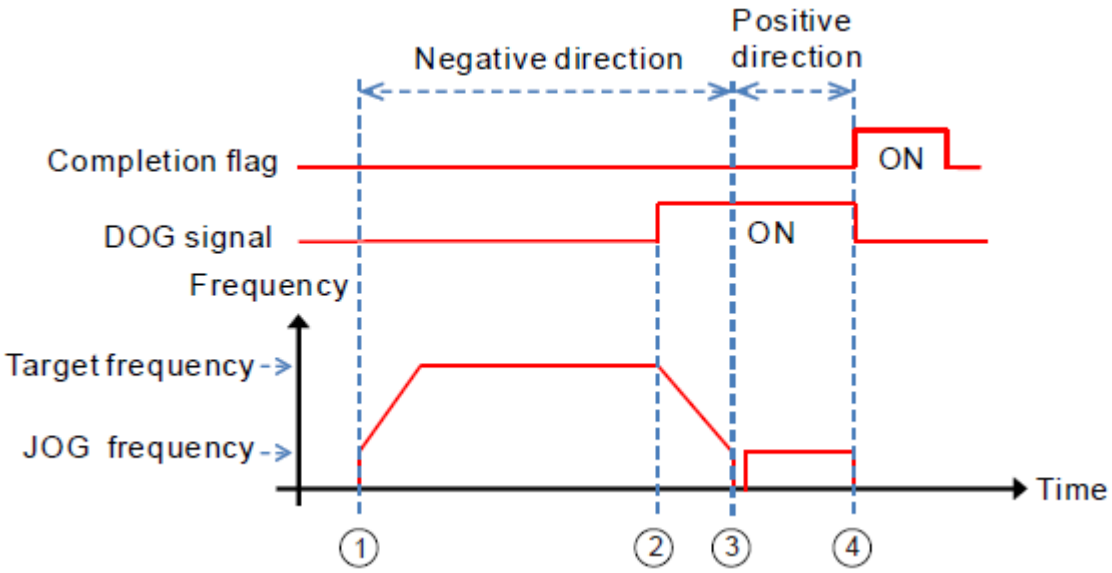


Function code 1:

① The DZRN2 function is executed and the search for the zero point is in the negative direction with the target frequency TFreq

② After the DOG signal is received, the output frequency decreases until the output stops. After that, the output is in the positive direction with the JOG frequency JFreq. The output does not stop until the DOG signal changes from ON to OFF

③ The axis moves away from the DOG signal and the output stops when the signal changes from ON to OFF




Function code 2:

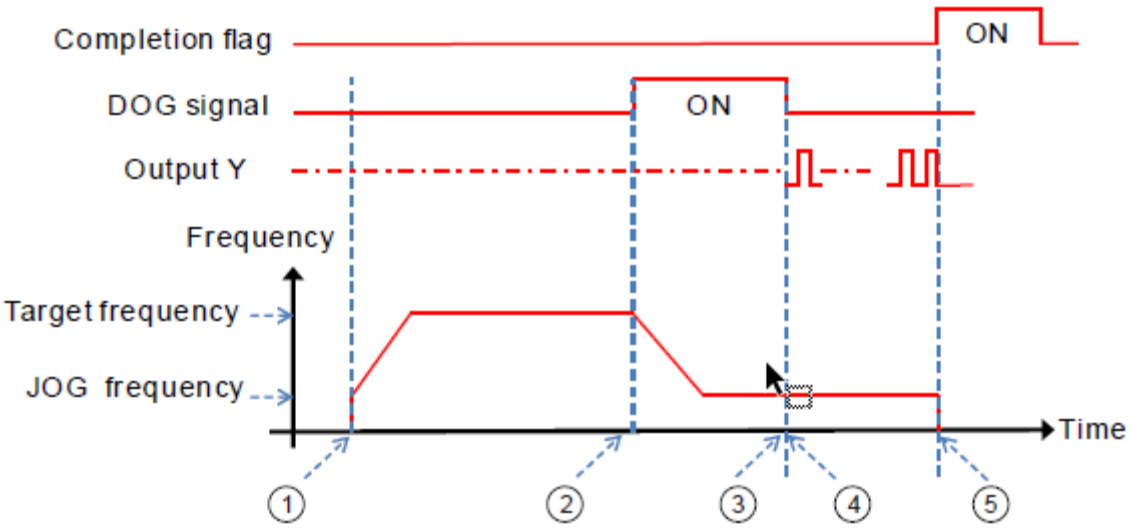
① The DZRN2 function is executed and the search for the zero point is in the negative direction with the target frequency TFreq

② After the DOG signal is received, the output frequency decreases to the JOG frequency JFreq and continues in the negative direction

③ When the DOG signal is left and the signal changes from ON to OFF, the specified number of pulses are output

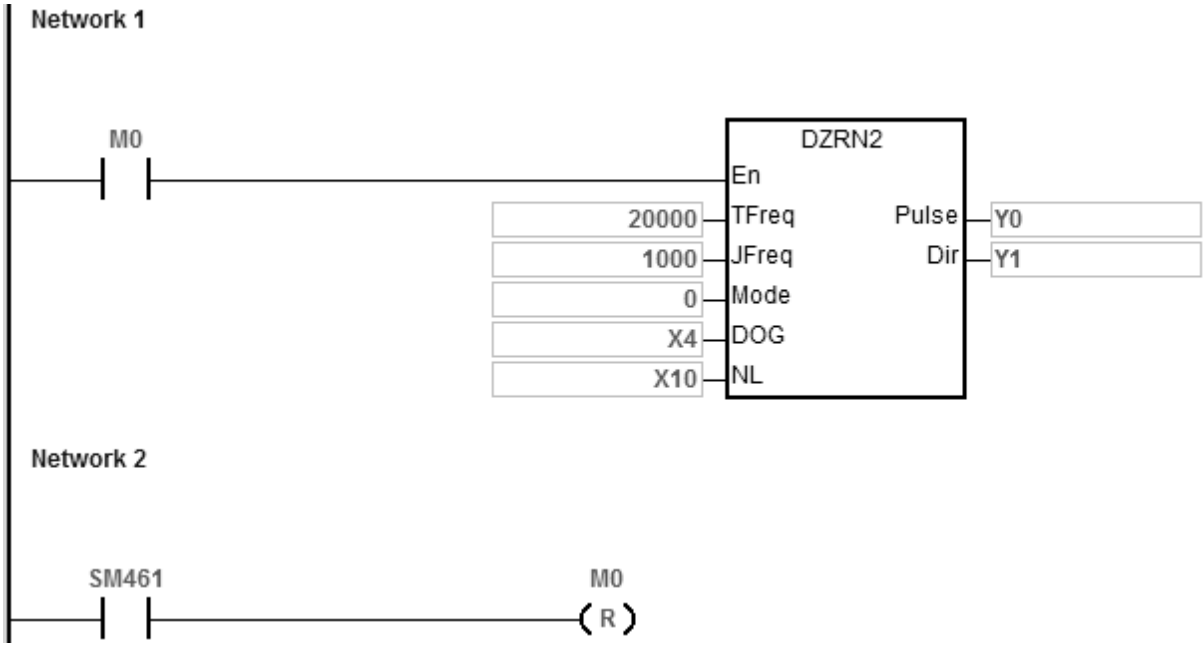
④ The first pulse output starts

⑤ When the 100th pulse output completes, the output stops and the Completion flag is ON




## Example

When M0 is ON, outputting the pulse from Y0 with a frequency of 20 kHz to search for the zero point in the negative direction. When the DOG signal is received and X4 is ON, it keeps moving in the negative direction with the JOG frequency of 1 kHz. The output stops immediately after X4 changes from ON to OFF.




## Explanation

Mode is set to 0.
High 16-bit [0000] is to disable the function to move a number of pulses..
Low 16-bit [0000] is Mode 0; when the DOG signal is received, the axis moves in the negative direction; after the axis moves away from the DOG signal, it stops immediately.

## Example 2 (in ST programming language)

Generally, it is suggested to use the ladder diagram language for editing the program when using this instruction. If you use ST language (which is available for ES3 with firmware V1.06.00 or later and SV3/SX3 V1.00.00 or later), you must wait until the zero-return motion is completed (SM461 completion flag is ON) and then disable the instruction. Besides, you must use the auto reset flag together to notify the PLC to release the high-speed output control right.
Taking Y0 output for example, the program writing is shown in the figure below

```
0001 IF M0 THEN
0002     DZRN2(1000,100,0,X0,X20,Y0,Y1);
0003     M1:=TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
```

## 2.5 UPDATE – MS300 Firmware Update to Version 2.01

### Function Correction

| Version 2.00 Issue | Version 2.01 |
|---|---|
| When copying parameters, the maximum operating frequency (Pr.01-00) will be copied abnormally. | When copying parameters, the maximum operating frequency (Pr.01-00) will be copied normally. |
| When the instantaneous power failure restart (Pr.07-06) or abnormal restart action selection (Pr.07-10) is set, when the Reset command is triggered continuously, it will cause abnormal action. | When the instantaneous power failure restart (parameter 07-06) or abnormal restart action selection (parameter 07-10) is set, when the Reset command is triggered continuously normally. |
| When PTC motor overheating (OH3) error is triggered, there is a chance that it cannot be reset (Reset). | When the PTC motor overheat (OH3) error is triggered, it can be reset normally (Reset). |
| When the multi-function display chooses to set the main frequency value display (00-04=47) or the frequency value display after the addition and subtraction of the main and auxiliary frequencies (00-04=48), the numerical display is abnormal. | When the multi-function display chooses to set the main frequency value display (00-04=47) or the frequency value display after the addition and subtraction of the main and auxiliary frequencies (00-04=48), the display is normal. |
| After the Jog command is executed in the FOC torque mode, it will stay in the speed mode. | After the Jog command is executed in the FOC torque mode, the torque mode is maintained. |
| Use MS300 Keypad, MODE left shift key function, input minus sign invalid. | Using MS300 Keypad, MODE left shift key function, input minus sign is normal. |
| Using the MI external terminal (Pr.00-21=1), the running function may fail. | Using the MI external terminal (Pr.00-21=1), the running function is normal. |

### Function Modification

1. External terminal signal optimization

2. IMSVC slip compensation calculation optimization: Correct the power calculation sequence to make the slip calculation correct

3. OC Stall optimizes the processing method of the power generation area to avoid triggering overvoltage

### Switching Period

| Firmware Version | Switching Period | |
|---|---|---|
| V2.01 | Taoyuan | T2226 |
| | Wujiang (Frame A-C) | W2218 |
| | Wujiang (Frame D-F) | W2217 |
| | Hosur | H2220 |

Note: S/N code T2226 means Taoyuan plant, year 2022, week 26.

## 2.6 UPDATE – VFD-EL Firmware Update to Version 1.18

### New Functions

New function to identify the analog conversion chip as main material or the substitute material.

**Note:**
There is no material shortage on analog conversion chip material, so the action of the substitute material will not be executed in short term.
If there is a need to switch to replacement material in the future, another notification will be issued.
Although the firmware has been upgraded to V1.18 in this revision, its functions and performance can be regarded as the same as firmware version V1.17

### Switching Period

| Firmware Version | Switching Period | |
|---|---|---|
| V1.18 | Taoyuan | T2314 |
| | Wujiang | W2238 |
| | Dongguan | D2238 |
| | Hosur | H2240 |

Note: S/N code T2314 means Taoyuan plant, year 2023, week 14.

## 2.7 UPDATE – DVP-S Load Cell Modules Firmware Update to Version 1.14

### Related Products

| Series | Model |
|---|---|
| DVP-S Slim PLC | DVP201LC-SL<br>DVP202LC-SL<br>DVP211LC-SL |

## New Functions

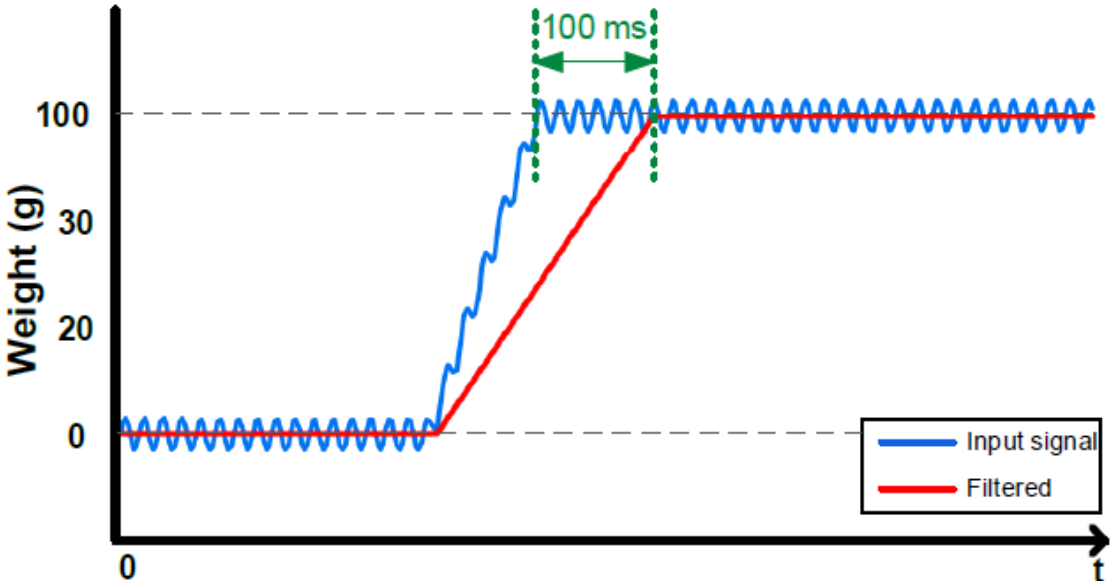
| CR# | Address | Attribute | | Register Name | Description |
|---|---|---|---|---|---|
| #43 | H102B | O | R/W | Filtering mode setting for channel 1 | K0: No filtering (default)<br>K1: Filtering out the maximum weight measured<br>K2: Averaging weights<br>**K3：LPF 5Hz**<br>**K4：LPF 10Hz**<br>**K5：LPF 20Hz**<br>**K6：LPF 50/60Hz**<br>Note: K3-K6 is only available for modules with firmware V1.14 or later. The transition time for low pass filter modes is fixed to 10 ms |
| #44 | H102C | O | R/W | Filtering mode setting for channel 2 | |

- The number behind LPF indicates the cut-off frequency point. For example, the mode **K3** is LPF 5 Hz and that means the low pass filter only allows low frequency signals from 0 Hz to 5 Hz to pass. Any frequencies applied above this cut-off point will be attenuated. The higher the frequency signals applied, the greater the attenuated will be. The mode **K6**, LPF 50/60 Hz, is designed specifically for filtering out the frequency of 50/60 Hz which is often seen as disturbance

- The transition time (CR3) in the low pass filter modes is fixed to 10 ms. Once it is not in the low pass filter mode, the transition time 10 ms can be modified

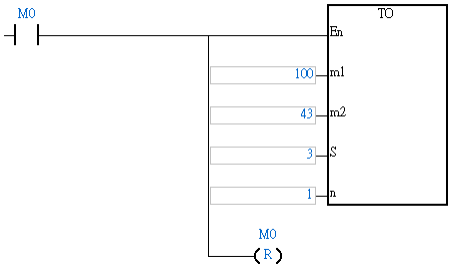
- The input delay times and output values in various low pass filter modes are shown below

- **Example of using mode K6, LPF 50/60**
  Input signal: Signals received from the load cell and a frequency of 50 Hz is also attached. See the blue sinusoidal waveform below.
  After applying the filtering mode, K6, LPF 50/60Hz, the frequency of 50 Hz is significantly attenuated, as shown in the red sinusoidal waveform below




- Methods to set the low pass filter

- When the module is connected to the left side of the PLC CPU, you can use To instruction to write the setting values K3-K6 into CR43 for channel 1 or CR44 for channel 2.
  The programming example below shows how to set the filtering mode K3, LPF 5 Hz, for the channel 1 of the first connected module on the left side




- Whether the module is connected to the left side of the PLC CPU or NOT, you can use the software, LCSoft V1.14 to set up the low pass filter mode and download the parameters through the RS-485 or RS-232 serial communication ports. LCSoft V1.14 will be available from November 2022.

- Whether the module is connected to the left side of the PLC CPU or NOT, with the remote PLC CPU or HMI, you can use Modbus protocols to write the setting values K3-K6 into CR43 for channel 1 or CR44 for channel 2 through the RS-485 or RS-232 serial communication ports.

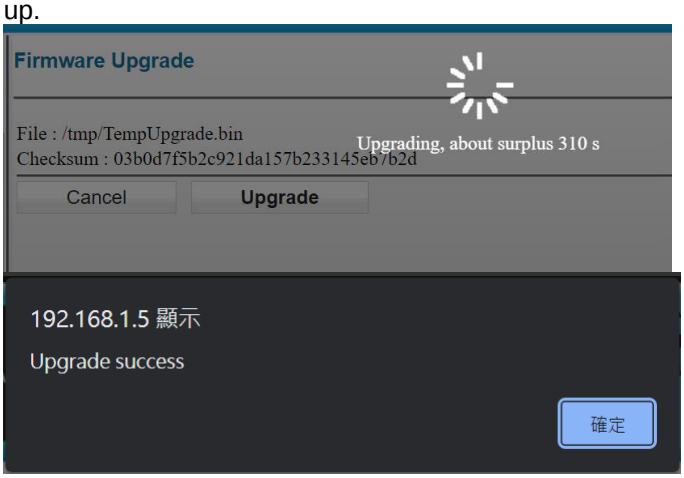
## 2.8 UPDATE – DVW-W01I2-E1 Wireless Access Point Firmware Update to Version 1.20

### Description

1. To minimize the risk of triggering RESET by mistake, increased the pressing button time for the RESTART and RESET to get started. And updated the corresponding DI/ALARM LED status

| | **Restart** | **Reset** |
|---|---|---|
| **OLD** | Press the RESET button for **<3** seconds and the DI/ALARM LED will be ON for 3 seconds and then OFF again. Wait for 120 seconds and the device restarts. | Press the RESET button for **>3** seconds and the DI/ALARM LED will be ON for 10 seconds and then OFF again. Wait for 120 seconds and the device resets to its default values. |
| **NEW** | Press the RESET button for **<=5** seconds and the DI/ALARM LED starts blinking. After releasing the button for 5 seconds, the DI/ALARM LED goes OFF. Wait for 120 seconds and the device restarts. | Press the RESET button for **>5** seconds and the DI/ALARM LED will be ON. After releasing the button for 5 seconds, the DI/ALARM LED goes OFF. Wait for 120 seconds and the device resets to its default values. |

2. Added updating information for the execution of firmware update

| | |
|---|---|
| **OLD** | No updating information during firmware update. |
| **NEW** | Added updating information to inform users how long it will take for the firmware update to complete. After the firmware update is complete, a message of “Upgrade success” shows up.<br>  |

3. Updated the APR packet transmission rules for a FIT AP

| | |
|---|---|
| **OLD** | For a FIT AP, if the MAC address belongs to the DVW device, but the IP address does NOT belong to the DVW device, the APR packets will be dropped. |
| **NEW** | For a FIT AP, even if the MAC address belongs to the DVW device, but the IP address does NOT belong to the DVW device, the APR packets will still be transmitted to the LAN port. |

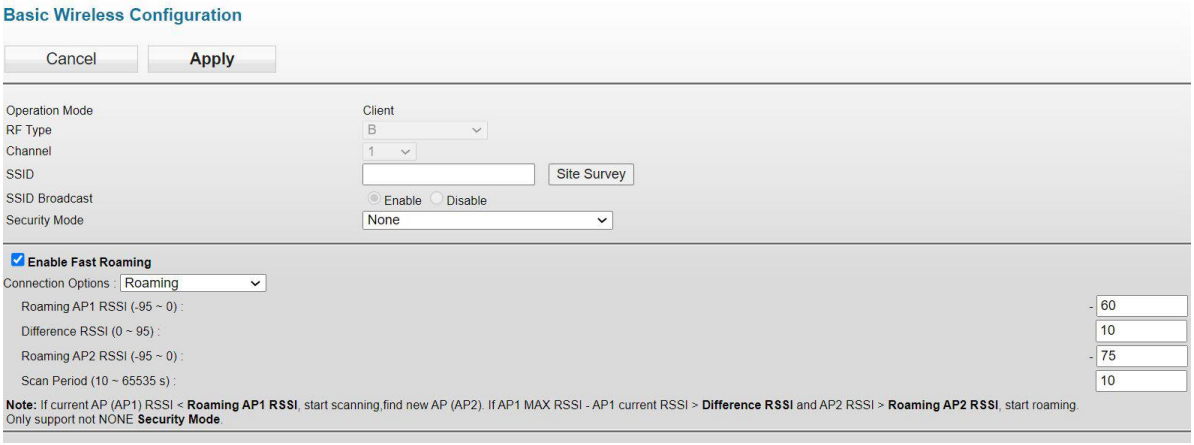
## Download Link

DVW-W01I2-E1 V1.20

## 2.9 UPDATE – DVW-W02W2-E2, DVW-W02W2-E2-EU Wireless Access Points Firmware Update to Version 2.5.2

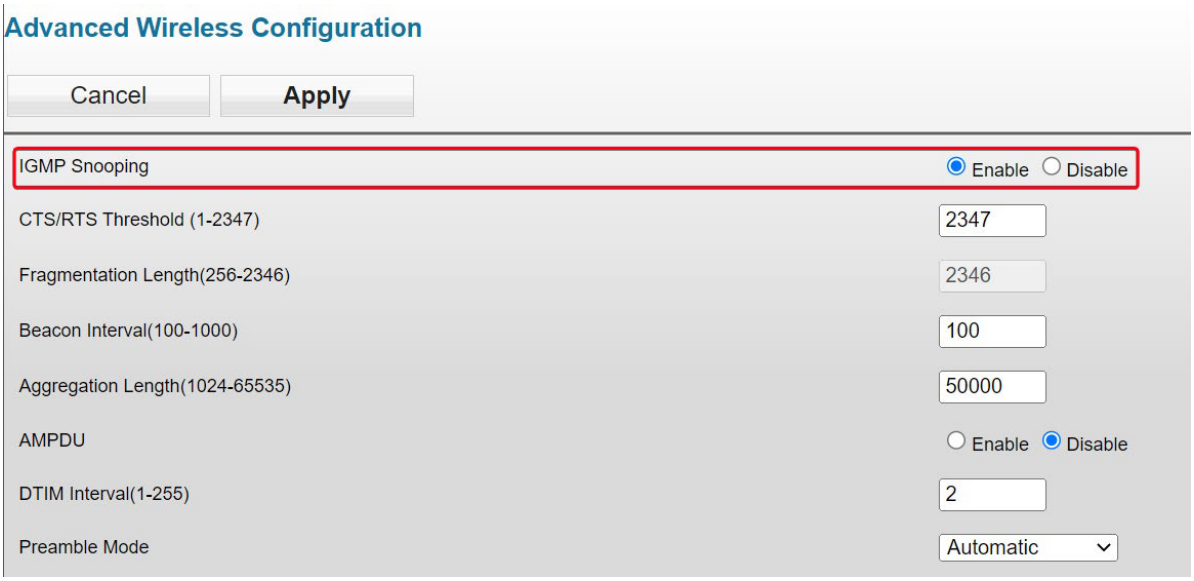
Description

1. Added Fast Roaming function in the Basic Wireless Configuration section. You can enable or disable this function. The default value is Disable

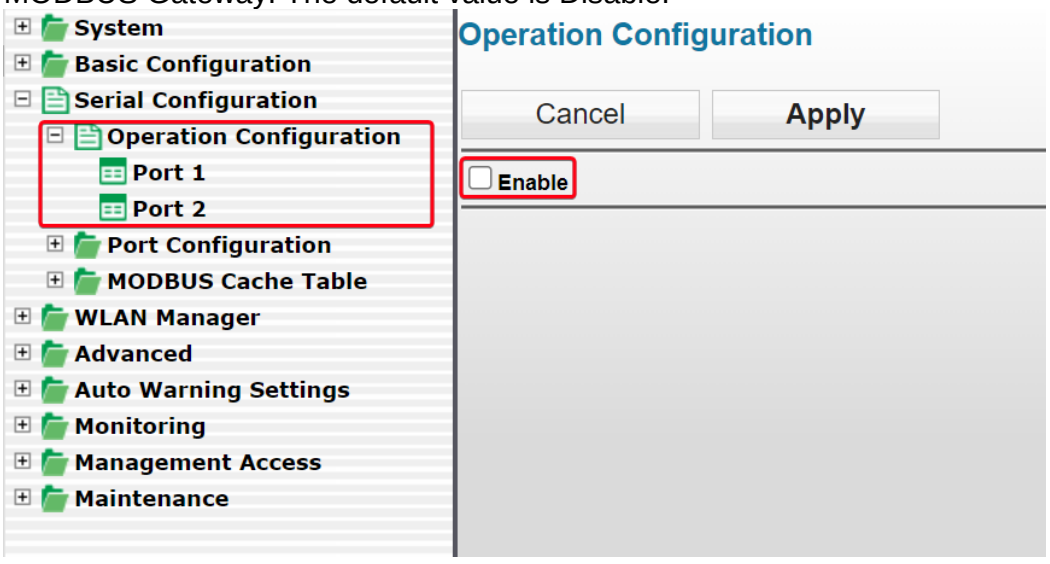



2. Added AMPDU function in the Advanced Wireless Configuration section. You can enable or disable this function. The default value is Disable


3. Added IGMP Snooping function in the Advanced Wireless Configuration section. You can enable or disable this function. The default value is Disable



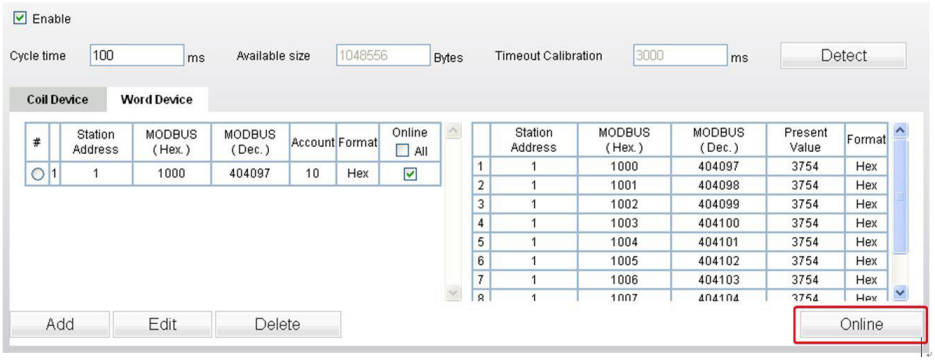
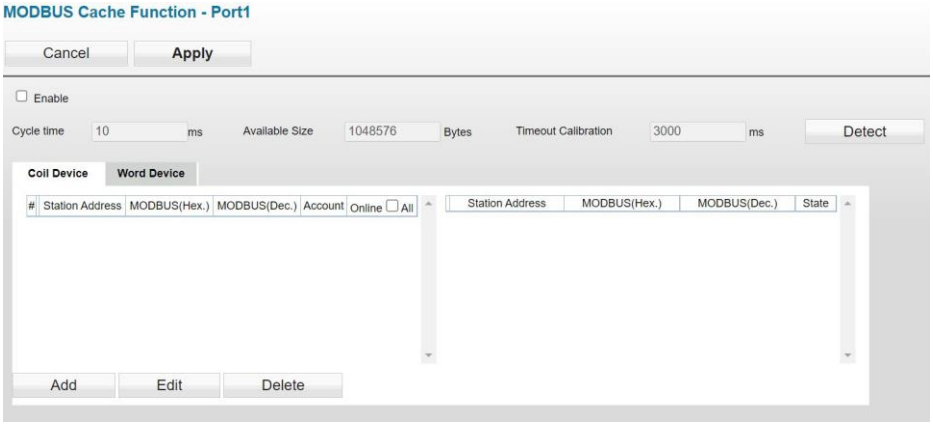

4. Added Operation Configuration page and users can enable or disable the MODBUS Gateway function. The default value is Disable

| | |
|---|---|
| **OLD** | No Operation Configuration setting page. MODBUS Gateway is enabled by default and users cannot change this setting. |
| **NEW** | Added Operation Configuration setting page. Users can enable or disable MODBUS Gateway. The default value is Disable.<br>  |

5. Fixed the security vulnerability issue of the special language input method for the webpage

6. Fixed an issue that the Modbus Slave App crashed from time to time

7. Removed the ONLINE button (no function) on the MODBUS Cache page

| | |
|---|---|
| **OLD** | The Online button is on the lower right corner but with no function.<br>  |
| **NEW** | No Online button.<br>  |

8. Improved the stability of FT Roaming

9. Fixed an issue that if trying to power off the device while it is in the process of turning on, the parameters will be restored back to default values

**Download Link**

DVW-W02W2-E2(-EU) V2.5.2

## 2.10 UPDATE – DX2100L1-CN Cloud Router Firmware Update to Version 1.5.0.12

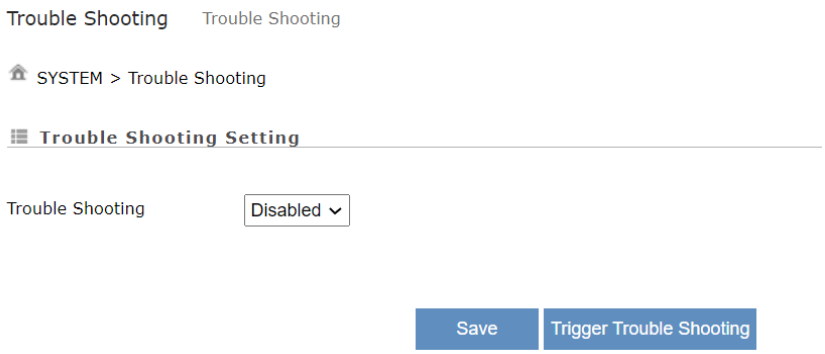
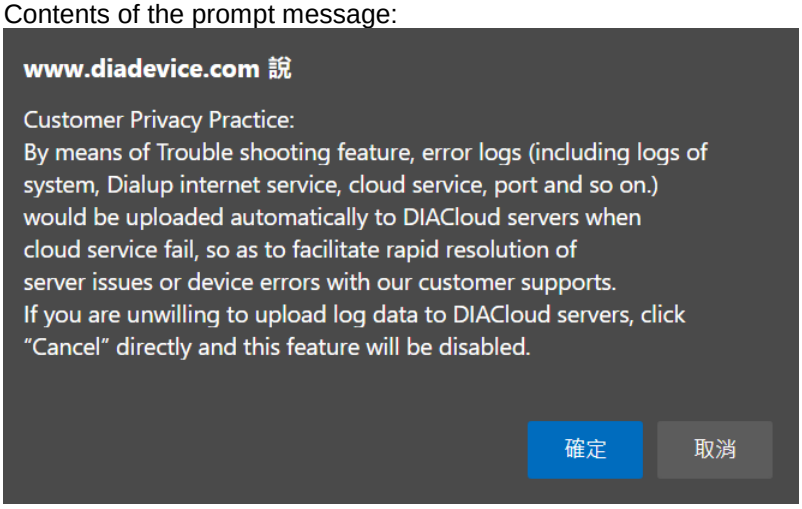
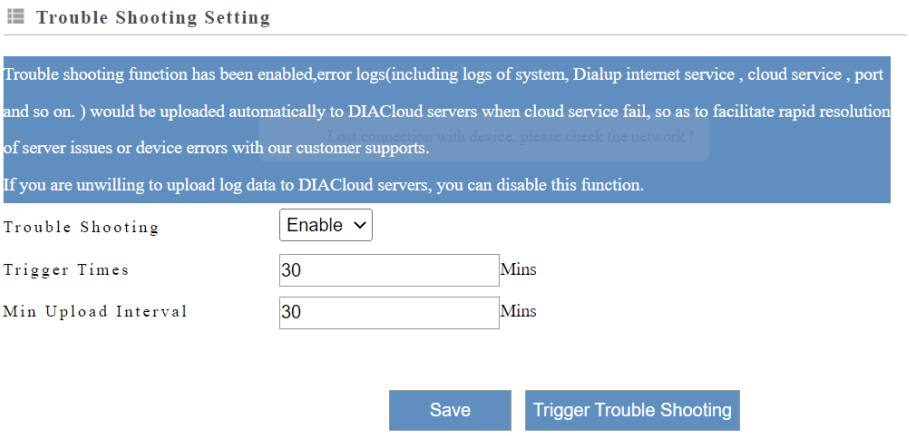
Description

1. Fixed the security vulnerability issue of the special language input method for the webpage

2. To minimize the risk of triggering RESET by mistake, increased the pressing button time for the RESTART and RESET to get started. And updated the corresponding READY LED status

| | **Restart** | **Reset** |
|---|---|---|
| **OLD** | Press the RESET button for **<=3** seconds and then release the button. The device restarts and then the READY LED is ON. | Press the RESET button for **>3** seconds and then release the button. All the LEDs start blinking. The device resets and then the READY LED is ON. |
| **NEW** | Press the RESET button for **<=5** seconds and the READY LED starts blinking. After releasing the button, the READY LED goes OFF. The device restarts and then the READY LED is ON. | Press the RESET button for **>=5** seconds and then release the button. The READY LED goes OFF. The device resets and then the READY LED is ON. |

3. Changed the default value of Trouble Shooting function from Enable to Disable and an explanation message will be prompted to inform users how this function works if users select Enable

<table>
<tr><td>OLD</td><td>The default value of Trouble Shooting function is Enable.</td></tr>
<tr><td rowspan="3">NEW</td><td>The default value of Trouble Shooting function is Disable. Once this function is enabled, a prompt message informs users that error logs will be uploaded automatically to DIACloud.<br> 
</td></tr>
<tr><td>Contents of the prompt message:<br> 
</td></tr>
<tr><td>Troubleshooting Setting Page:<br> 
</td></tr>
</table>

## Download Link

DX-2100L1-CN V1.5.0.12

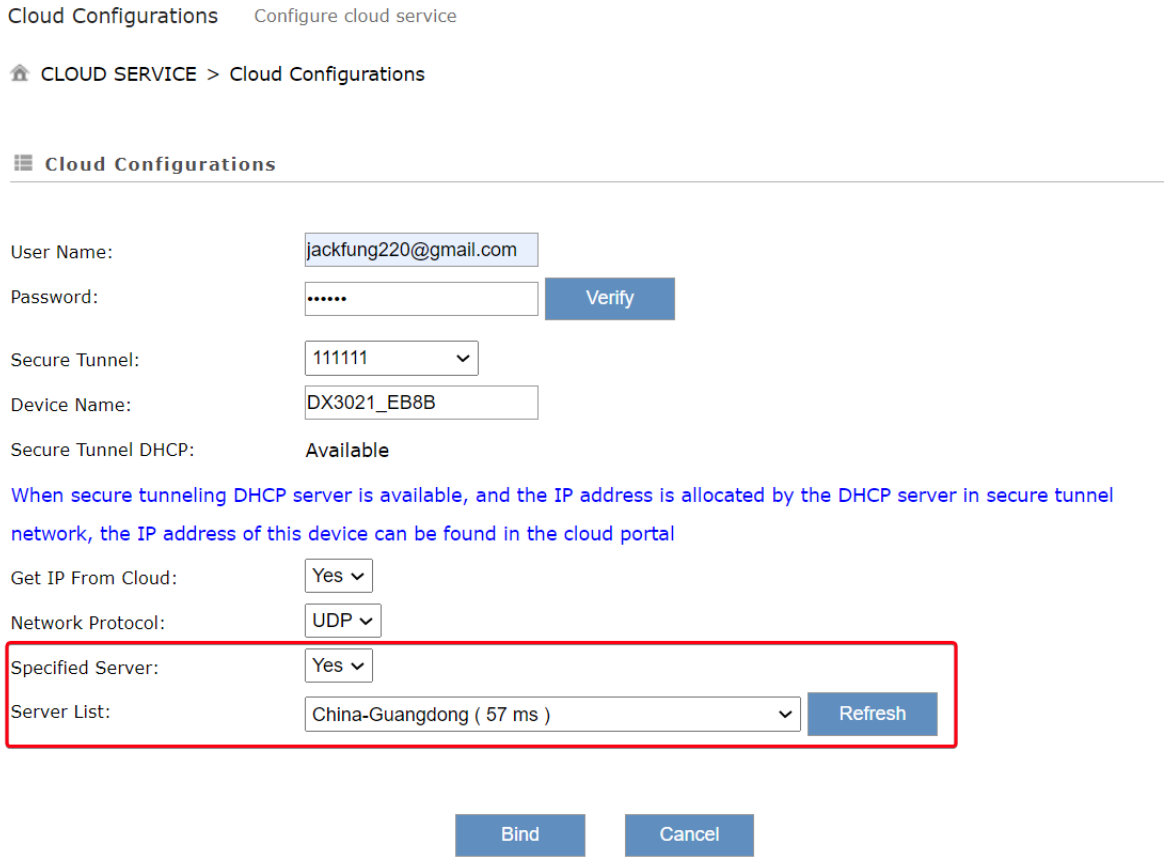
## 2.11 UPDATE – DX2300LN-WW, DX2300LN-CN Cloud Router Firmware Update to Version 1.5.2.0

### Description

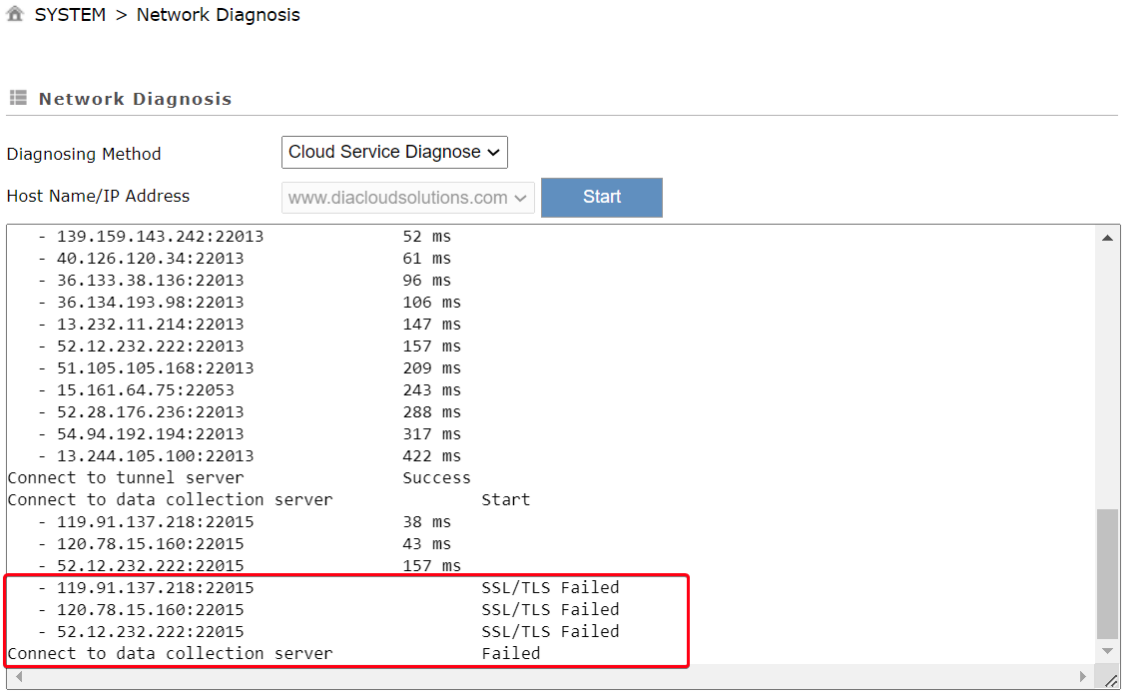
1. Added internal registers $900~$914 (the communication status of the lower device) as the ones to be uploaded to DIACloud. So that users can check the communication status of the lower device on DIACloud

2. This series supports MODBUS and MODBUS TCP function code 16

3. Added setting options Specified Server and Server List on the Cloud Configuration page. Once the Specified Server option is enabled, users can select a specific DIACloud Server from the Server List to establish a connection with DX-2300LN



4. Added a check mechanism to verify the permission to visit DIACloud SSL/TLS Server when executing the Diagnosing and Cloud Service Diagnose on the Network Diagnosis page



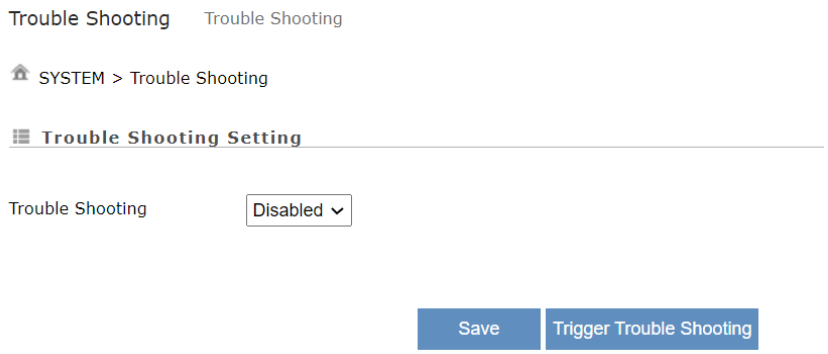
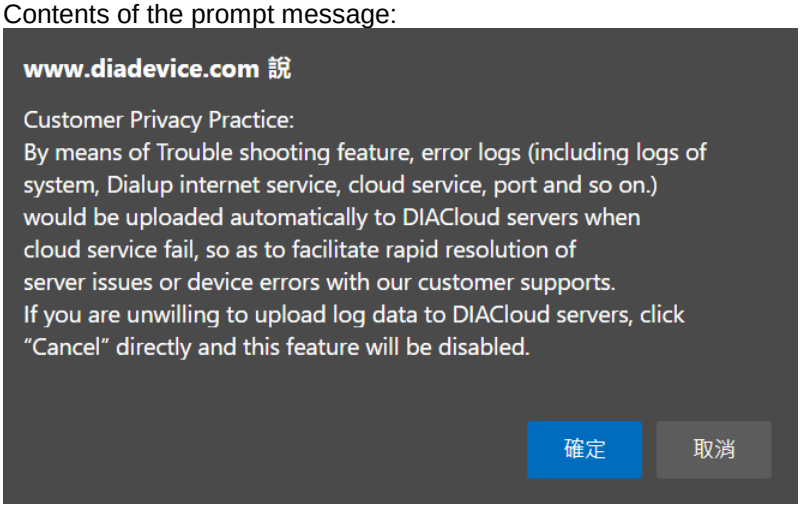
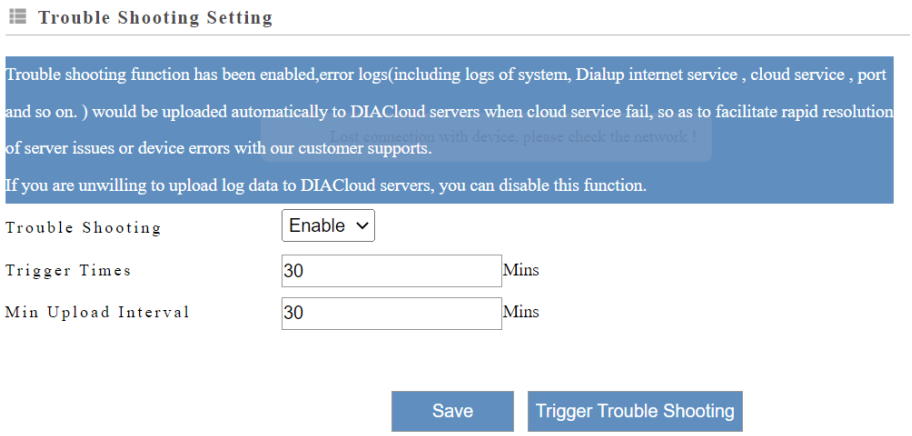

5. Fixed an issue that if the delay time of the DIACloud connection exceeds 1000 ms, the delay time record on DX-2300LN will be incorrect

6. Fixed an issue that the NTP synchronization cannot succeed

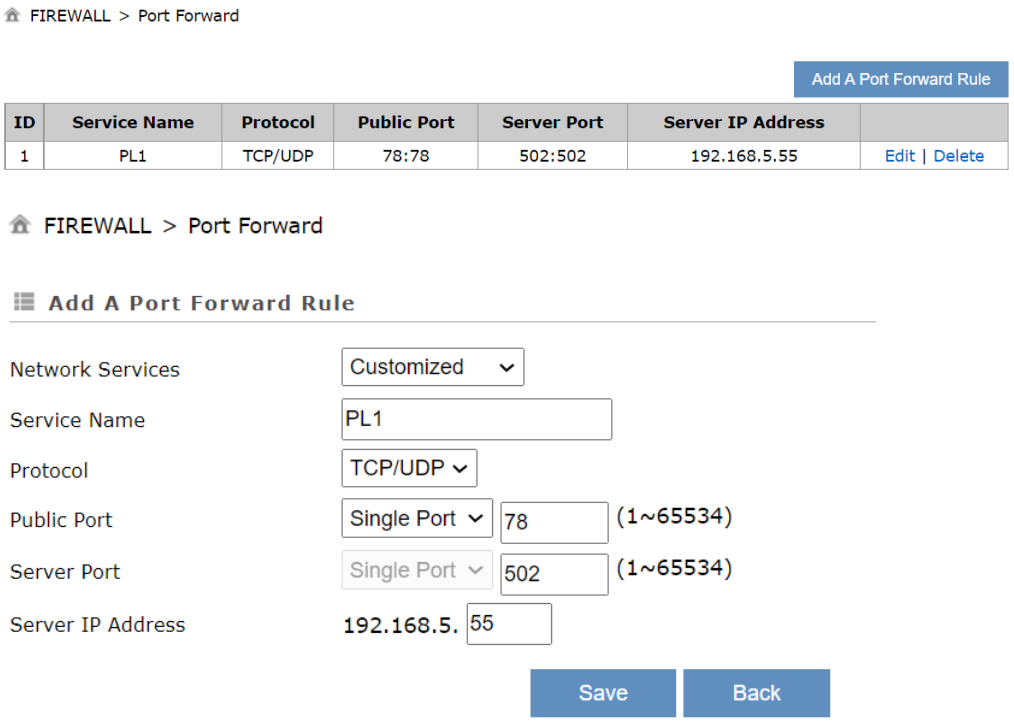
7. To minimize the risk of triggering RESET by mistake, increased the pressing button time for the RESTART and RESET to get started. And updated the corresponding READY LED status

| | **Restart** | **Reset** |
|---|---|---|
| **OLD** | Press the RESET button for **<=3** seconds and then release the button. The device restarts and then the READY LED is ON. | Press the RESET button for **>3** seconds and then release the button. All the LEDs start blinking. The device resets and then the READY LED is ON. |
| **NEW** | Press the RESET button for **<=5** seconds and the READY LED starts blinking. After releasing the button, the READY LED goes OFF. The device restarts and then the READY LED is ON. | Press the RESET button for **>=5** seconds and then release the button. The READY LED goes OFF. The device resets and then the READY LED is ON. |

8. Changed the default value of Trouble Shooting function from Enable to Disable and an explanation message will be prompted to inform users how this function works if users select Enable

<table>
<tr><td>OLD</td><td>The default value of Trouble Shooting function is Enable.</td></tr>
<tr><td rowspan="3">NEW</td><td>The default value of Trouble Shooting function is Disable. Once this function is enabled, a prompt message informs users that error logs will be uploaded automatically to DIACloud.<br> 
</td></tr>
<tr><td>Contents of the prompt message:<br> 
</td></tr>
<tr><td>Trouble Shooting Setting Page:<br> 
</td></tr>
</table>

9. Increased the maximum number of port forwards to 20

| | |
|---|---|
| **OLD** | Up to 10 sets of port forward can be used. |
| **NEW** | Up to 20 sets of port forward can be used.<br>  <br> |

### Download Link

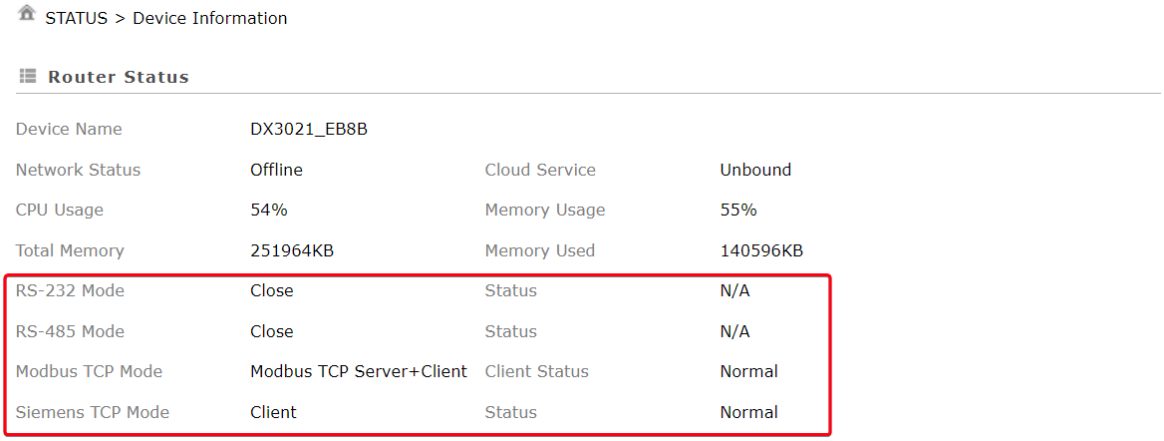
DX-2300LN-WW, DX2300LN-CN V1.5.2.0

## 2.12 UPDATE – DX3021L9 Cloud Router Firmware Update to Version 1.24

### Description

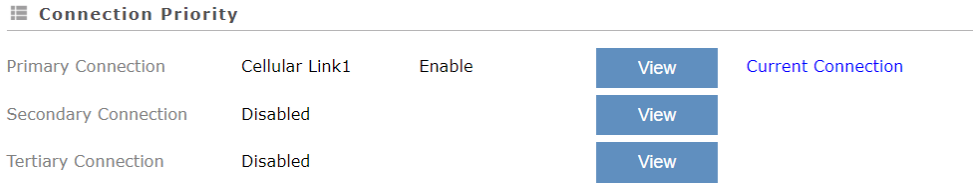
1. Added internal registers $900~$914 (the communication status of the lower device) as the ones to be uploaded to DIACloud. So that users can check the communication status of the lower device on DIACloud

2. Added more communication statuses on Router Status page for users to check, including RS-232 Mode, RS485 Mode, MODBUS TCP Mode and Siemens TCP Mode

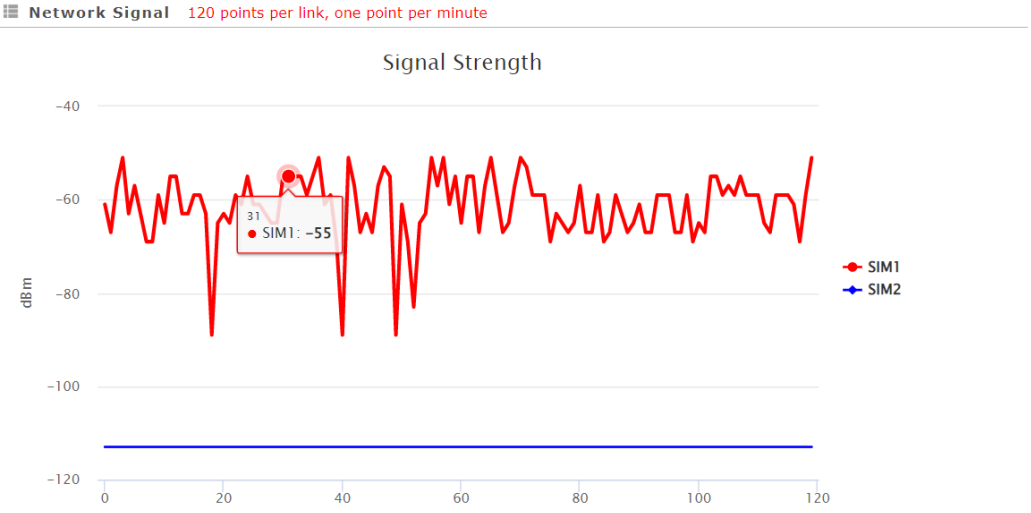


3. Added more connection information on Uplink Network Status page, including information concerning Connection Priority, Network Signal and Network Record

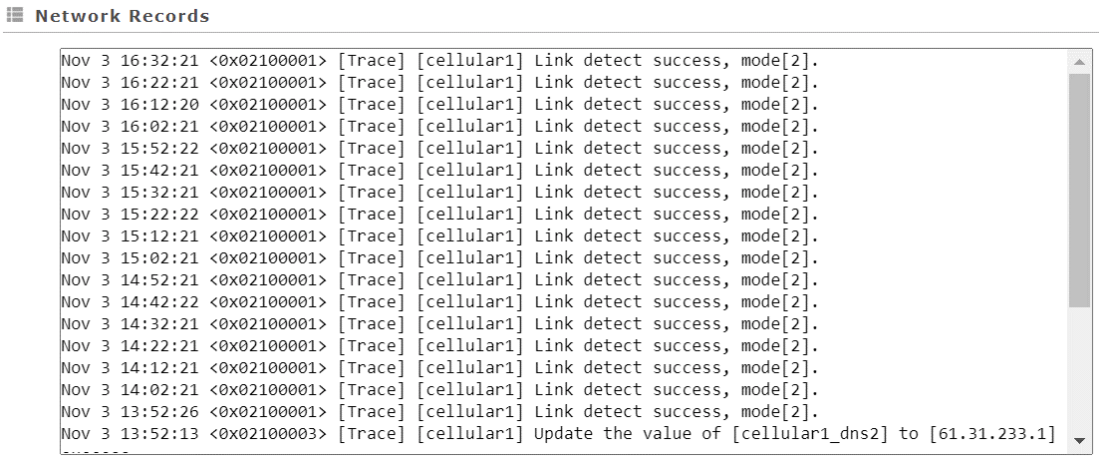
| Connection Priority |
| --- |
| Users can see the connection priority and the connection status in this section. Click View to see more details. |



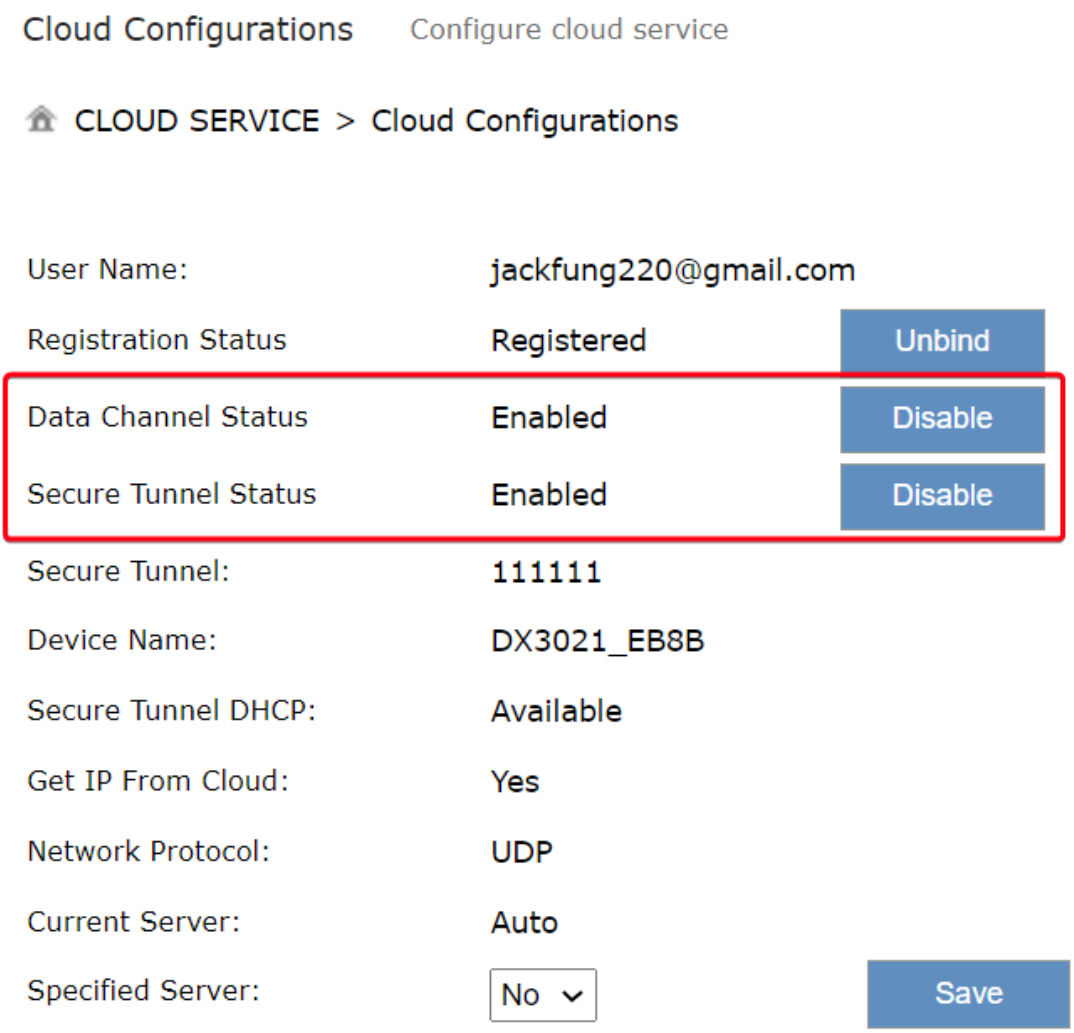

| Network Signal |
| --- |
| The network signal strength of the installed SIM card is recorded per minute. And the last 120 minutes of the network signal strength are shown in the network signal chart. |



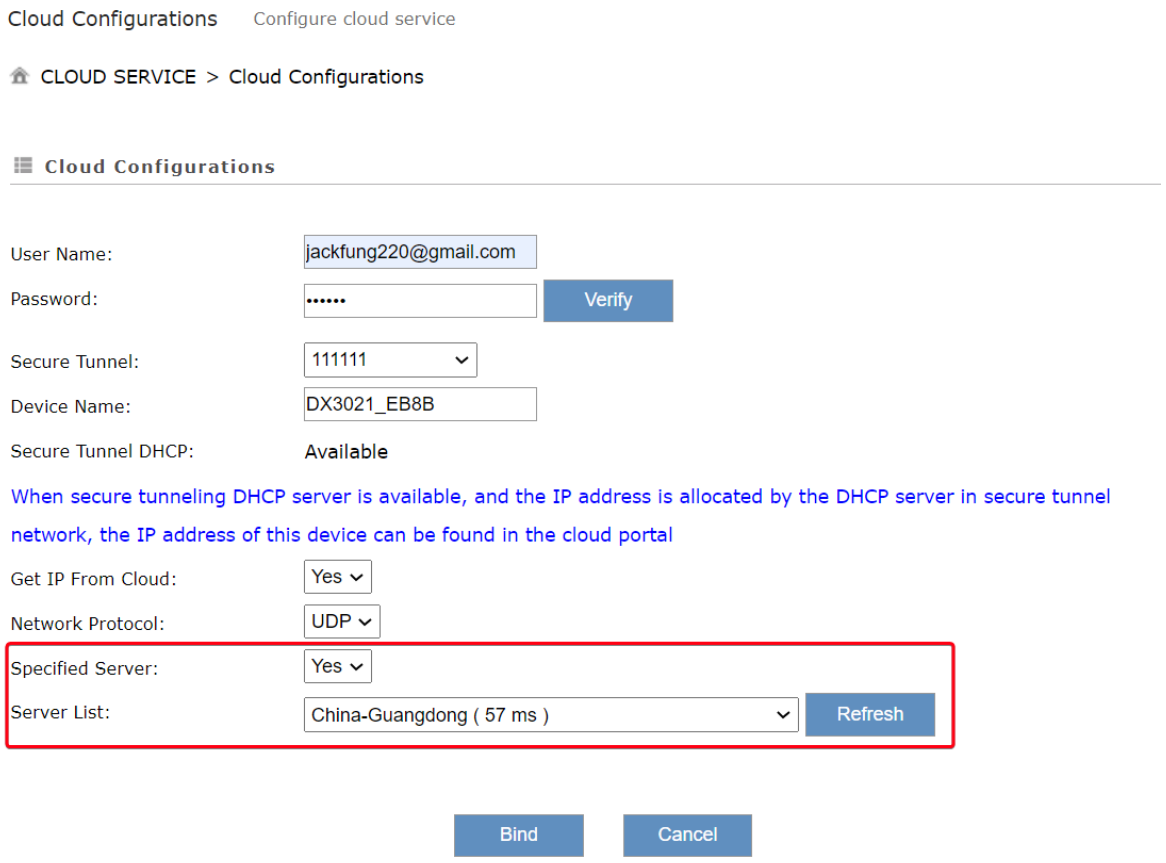

| Network Record |
| --- |
| Network record includes MAC address, IP address, DNS server, proxy and so forth. |



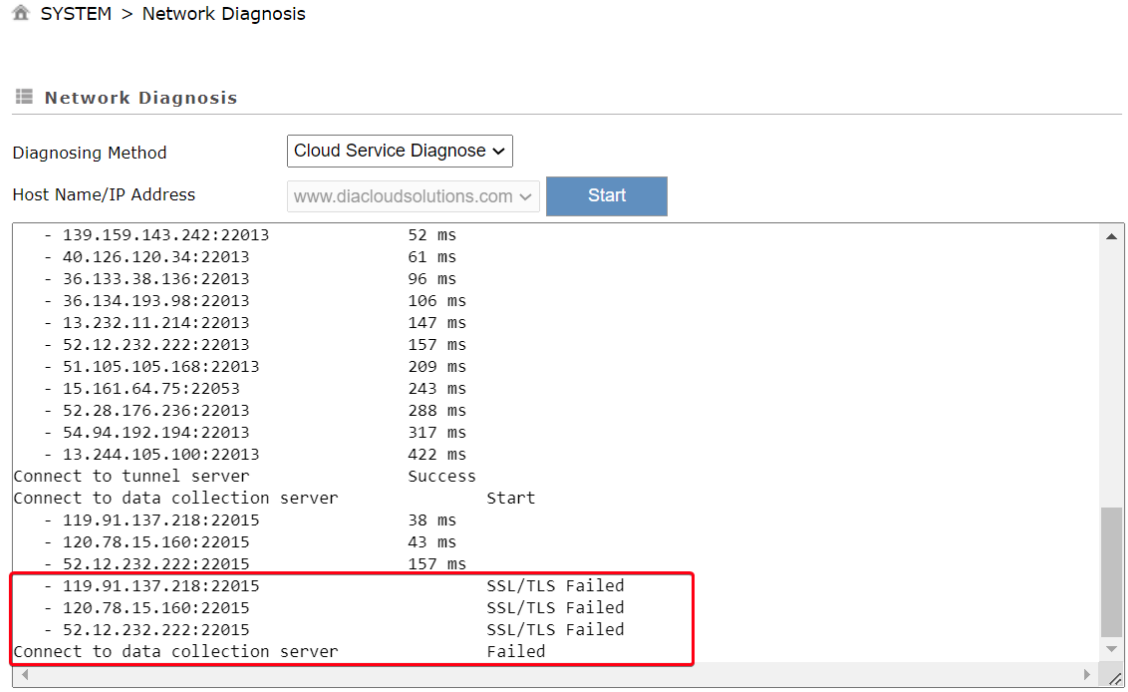
4. Added setting options Data Channel Status and Secure Tunnel Status on the Cloud Configuration page. You can enable or disable the functions



5. Added setting options Specified Server and Server List on the Cloud Configuration page. Once the Specified Server option is enabled, users can select a specific DIACloud Server from the Server List to establish a connection with DX-3021L9




6. Added a check mechanism to verify the permission to visit DIACloud SSL/TLS Server when executing the Diagnosing and Cloud Service Diagnose on the Network Diagnosis page



7. Fixed the security vulnerability issue of the special language input method for the webpage

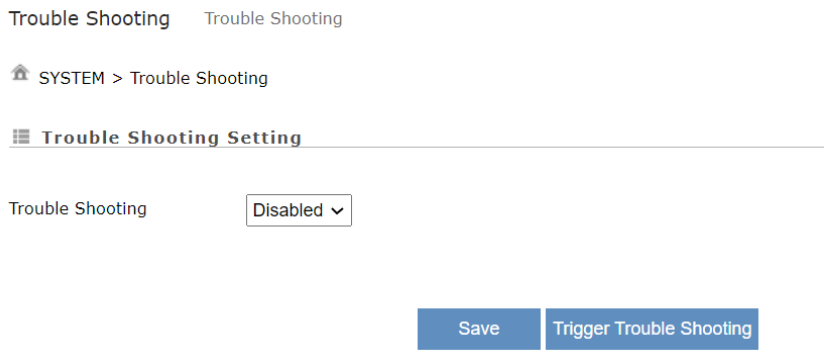
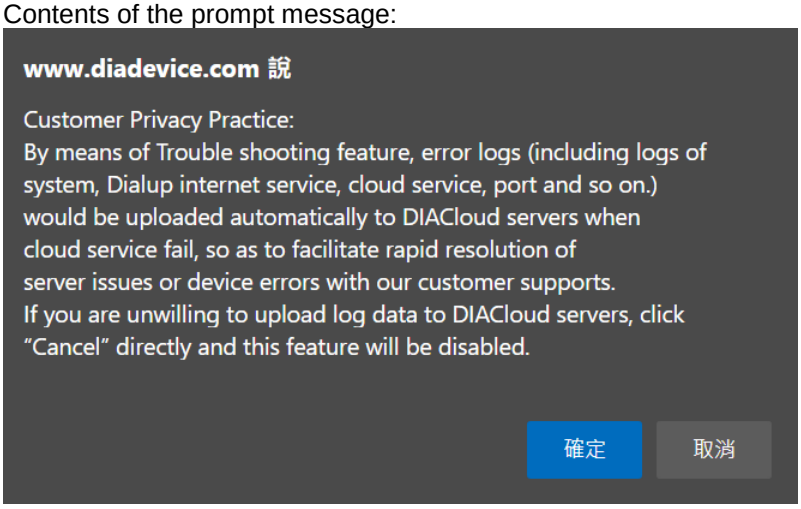
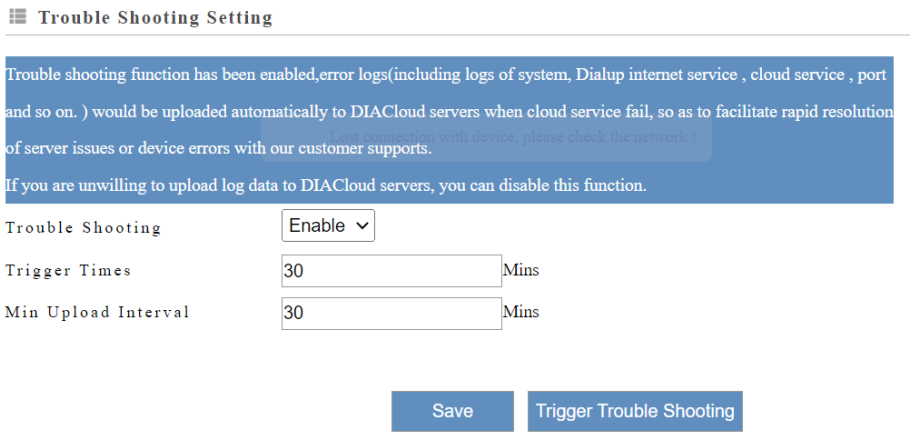
8. Fixed an issue that if the delay time of the DIACloud connection exceeds 1000 ms, the delay time record on DX-2300LN will be incorrect

9. Fixed an issue that the Siemens TCP Mode on the Router Status page cannot be shown correctly

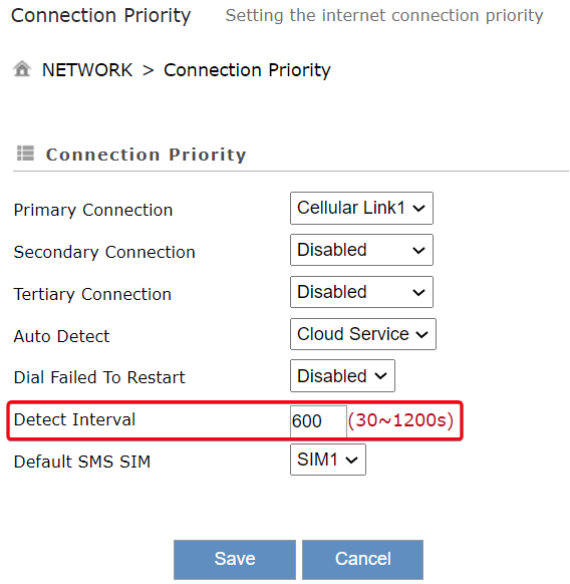
10. To minimize the risk of triggering RESET by mistake, increased the pressing button time for the RESTART and RESET to get started. And updated the corresponding READY LED status

| | **Restart** | **Reset** |
|---|---|---|
| **OLD** | Press the RESET button for **<=3** seconds and then release the button. The device restarts and then the READY LED is ON. | Press the RESET button for **>3** seconds and then release the button. All the LEDs start blinking. The device resets and then the READY LED is ON. |
| **NEW** | Press the RESET button for **<=5** seconds and the READY LED starts blinking. After releasing the button, the READY LED goes OFF. The device restarts and then the READY LED is ON. | Press the RESET button for **>=5** seconds and then release the button. The READY LED goes OFF. The device resets and then the READY LED is ON. |

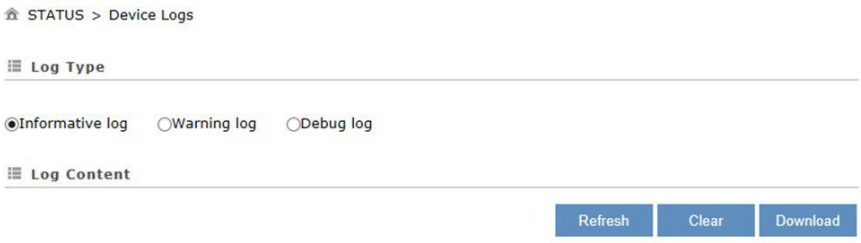
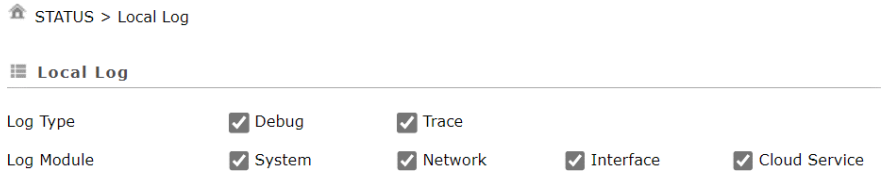
11. Changed the default value of Trouble Shooting function from Enable to Disable and an explanation message will be prompted to inform users how this function works if users select Enable

<table>
<tr><td>OLD</td><td>The default value of Trouble Shooting function is Enable.</td></tr>
<tr><td rowspan="3">NEW</td><td>The default value of Trouble Shooting function is Disable. Once this function is enabled, a prompt message informs users that error logs will be uploaded automatically to DIACloud.<br> 
</td></tr>
<tr><td>Contents of the prompt message:<br> 
</td></tr>
<tr><td>Troubleshooting Setting Page:<br> 
</td></tr>
</table>

12. Updated the default value of the Detect Interval from 60 to 600 seconds. And the setting range has changed from 30~300 seconds to 30~1200 seconds

| | |
|---|---|
| **OLD** | Default value: 60 seconds; setting range: 30~300s |
| **NEW** | Default value: 600 seconds; setting range: 30~1200s<br>  |

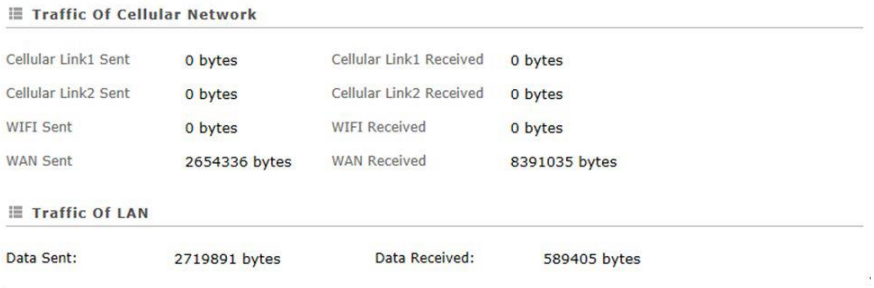
13. Updated the options under the Local Log

| | |
|---|---|
| **OLD** | Log Type: Informative log, Warning log, Debug log<br>  |
| **NEW** | Updated Log Type: Debug, Trace<br>Added Log Module: System, Network, Interface, Cloud Service<br>  |

14. Updated the content on the Traffic Statistics page

**OLD**

Only traffics of cellular network and LAN are shown.



Traffic Of Cellular Network

| | | | |
|---|---|---|---|
| Cellular Link1 Sent | 0 bytes | Cellular Link1 Received | 0 bytes |
| Cellular Link2 Sent | 0 bytes | Cellular Link2 Received | 0 bytes |
| WIFI Sent | 0 bytes | WIFI Received | 0 bytes |
| WAN Sent | 2654336 bytes | WAN Received | 8391035 bytes |

Traffic Of LAN

| | | | |
|---|---|---|---|
| Data Sent: | 2719891 bytes | Data Received: | 589405 bytes |

**NEW**

The traffics are categorized to 4G, Wi-Fi, WAN and LAN. And the traffics are calculated according to the groups, Today, Yesterday, This Week and This Month.

Traffic Of Cellular (Bytes)

| | Today | Yesterday | This Week | This Month |
|---|---|---|---|---|
| Cellular Link1 Sent | 7340522 | 0 | 7340522 | 7340522 |
| Cellular Link1 Received | 3946706 | 0 | 3946706 | 3946706 |
| Cellular Link2 Sent | 0 | 0 | 0 | 0 |
| Cellular Link2 Received | 0 | 0 | 0 | 0 |
| Total | 11287228 | 0 | 11287228 | 11287228 |

Traffic Of WAN (Bytes)

| | Today | Yesterday | This Week | This Month |
|---|---|---|---|---|
| WAN Sent | 0 | 0 | 0 | 0 |
| WAN Received | 0 | 0 | 0 | 0 |

Traffic Of Wi-Fi (Bytes)

| | Today | Yesterday | This Week | This Month |
|---|---|---|---|---|
| WIFI Sent | 0 | 0 | 0 | 0 |
| WIFI Received | 0 | 0 | 0 | 0 |

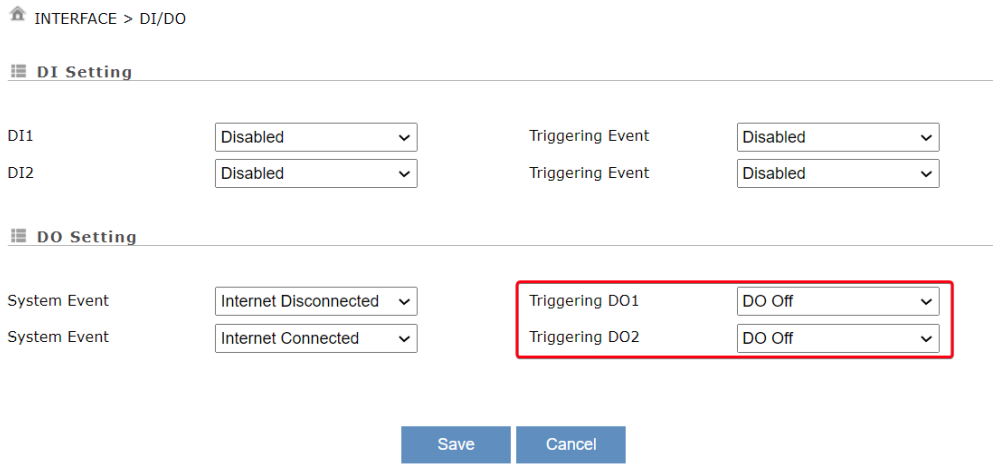
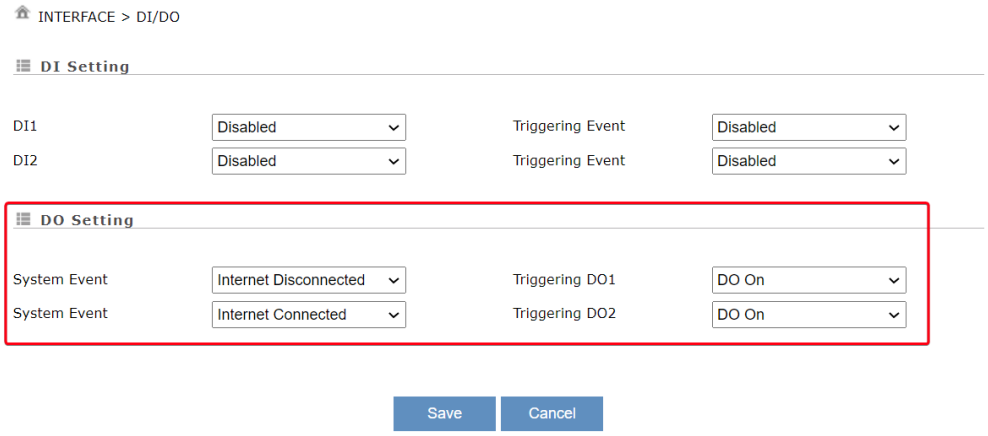
Traffic Of LAN (Bytes)

| | Today | Yesterday | This Week | This Month |
|---|---|---|---|---|
| LAN Sent | 10326264 | 4708900 | 15035164 | 15035164 |
| LAN Received | 4362761 | 2891859 | 7254620 | 7254620 |

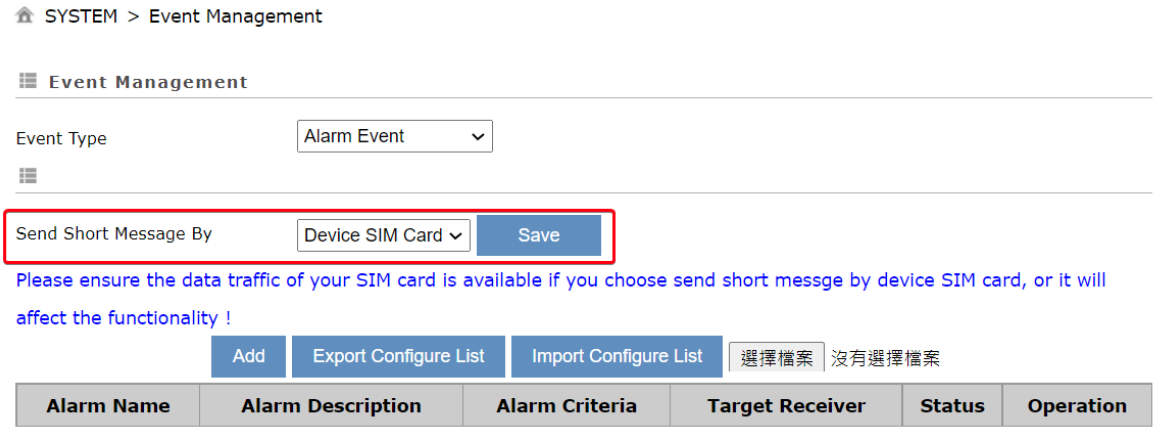
15. Updated the items shown on the Cloud Status page

| | |
|---|---|
| **OLD** | Only Registration Status, Service Status, Activated Time are shown.<br>STATUS > Cloud Status<br><br>**Cloud Status**<br>Registration Status: Not registered<br>Service Status: Disabled<br>Activated Time: N/A |
| **NEW** | 1. The following statuses are added, Data Channel Status, Secure Tunnel Status and Cloud Records.<br>2. Deleted Activated Time.<br>STATUS > Cloud Status<br>**Cloud Status**<br>Registration Status: Not registered<br>Registration Time: N/A<br>Data Channel Status: Disabled<br>Secure Tunnel Status: Disabled<br>**Cloud Records**<br>Nov 3 14:08:26 <0x05010004> [Debug] Vidagrid disabled.<br>Nov 3 14:08:26 <0x05020001> [Debug] Data channel not connected.<br>Nov 3 14:08:26 <0x05030001> [Debug] Secure tunnel not connected.<br>Nov 3 14:08:08 <0x05010002> [Debug] Join domain failed.<br>Nov 3 14:08:08 <0x05010003> [Debug] User logout.<br>Nov 3 13:58:28 <0x05020002> [Debug] Data channel connected.<br>Nov 3 13:58:10 <0x05020001> [Debug] Data channel not connected.<br>Nov 3 13:57:46 <0x05030002> [Debug] Secure tunnel connected.<br>Nov 3 13:57:35 <0x05010001> [Debug] Join domain success, register time: 2022-11-03 05:57:35 UTC. |

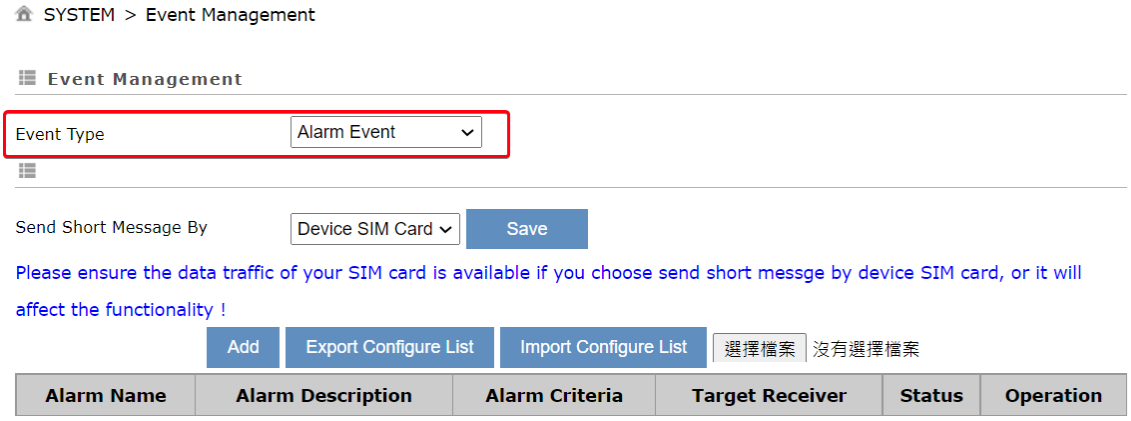
16. Updated options under DO Setting

| | |
|---|---|
| **OLD** | When the System Event is detected, the option Triggering DO1 is triggered and DO LED is ON. After the System Event is restored, the DO LED stays ON. |
| **NEW** | 1. Removed the setting option, DO OFF, under Triggering DO1 and Triggering DO2.<br>  <br>2. When the System Event is detected, the option Triggering DO1 is triggered and DO LED is ON. After the System Event is restored, the DO LED is OFF.<br>  |

17. Removed the Cloud SMS Gateway funciton from the Send Short Message on Event Management page



18. Removed the Communication verification funciton from the Event Type on Event Management page




### Download Link

DX-3021L9 V1.24

## 2.13 UPDATE – DCT-MV Wiring Length Increase

### Objective

The wiring length of DCT-MV series has changed from 1,200 mm to 2,000 mm to meet more application requirements.

### Related Products

DCT-MV005-3
DCT-MV060-3
DCT-MV100-3
DCT-MV200-3
DCT-MV300-3
DCT-MV400-3

With production dates after week 42, 2022. The information can be obtained from the serial number on the product label, e.g. IDCTxxxxW2242xxxx was produced in 2022, week 42.

## 2.14 UPDATE – DIAVH Core-i IPC/PPC Upgrade to 7th Generation Core™ i CPU

### Description

| Item | Before | After |
|---|---|---|
| Core-i3 CPU Number | i3-6100U | i3-7100U |
| Core-i5 CPU Number | i5-6200U | i5-7200U |

### Applicable Models

| | |
|---|---|
| IPC/PPC | DIAVH-IPC0031xx<br>DIAVH-IPC0051xx<br>DIAVH-IPC0031xxA<br>DIAVH-IPC0051xxA<br>DIAVH-PPC1531xx<br>DIAVH-PPC1551xx<br>DIAVH-PPC1531xxA<br>DIAVH-PPC1551xxA<br>DIAVH-PPC1731xx<br>DIAVH-PPC1751xx<br>DIAVH-PPC1731xxA<br>DIAVH-PPC1751xxA<br>DIAVH-PPC1931xx<br>DIAVH-PPC1951xx<br>DIAVH-PPC1931xxA<br>DIAVH-PPC1951xxA<br>DIAVH-PPC2131xx<br>DIAVH-PPC2151xx<br>DIAVH-PPC2131xxA<br>DIAVH-PPC2151xxA |

Release Date: Around September 05, 2022 (Week 2236), stock-dependent, running change.

You can find the serial number on the Core-i series product label of the IPC/PPC. If the serial number is IPCxxxxxxxW2236xxxx or PPCxxxxxxxW2236xxxx (indicating year 2022, week 36) or later, it means the package is equipped with Intel® 7th Gen Core™ i CPU.

## 2.15 UPDATE – DIAVH Core-i IPC/PPC Upgrade to 8th Generation Core™ i CPU

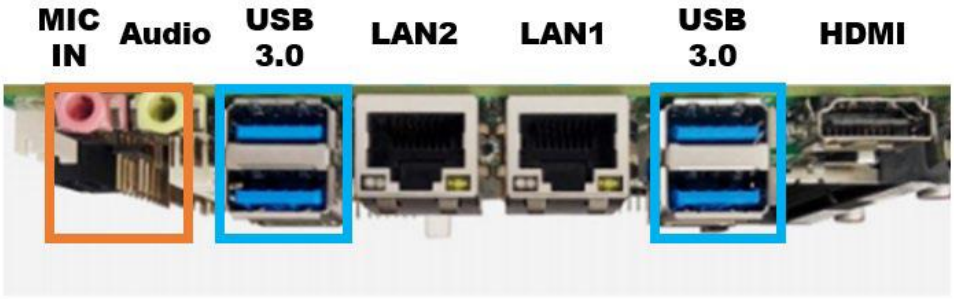
### Description

| Item | Before | After |
|---|---|---|
| Core-i3 CPU Number | i3-6100U | i3-8145U |
| Core-i5 CPU Number | i5-6200U | i5-8265U |

### Changed I/O placement on IPC/PPC series accordingly

1. Changed the placement of USB ports
2. Removed MIC IN port

**Before**




**After**


## Applicable Models

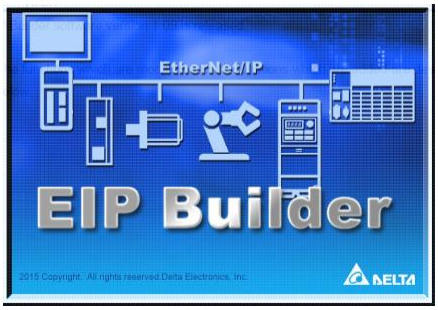
| | |
|---|---|
| IPC/PPC | DIAVH-IPC0031xx<br>DIAVH-IPC0051xx<br>DIAVH-IPC0031xxA<br>DIAVH-IPC0051xxA,<br>DIAVH-PPC1531xx<br>DIAVH-PPC1551xx<br>DIAVH-PPC1531xxA<br>DIAVH-PPC1551xxA<br>DIAVH-PPC1731xx<br>DIAVH-PPC1751xx<br>DIAVH-PPC1731xxA<br>DIAVH-PPC1751xxA<br>DIAVH-PPC1931xx<br>DIAVH-PPC1951xx<br>DIAVH-PPC1931xxA<br>DIAVH-PPC1951xxA<br>DIAVH-PPC2131xx<br>DIAVH-PPC2151xx<br>DIAVH-PPC2131xxA<br>DIAVH-PPC2151xxA |

Release Date: Around September 12, 2022 (Week 2237), stock-dependent, running change.

You can find the serial number on the Core-i series product label of the IPC/PPC. If the serial number is IPCxxxxxxxW2237xxxx or PPCxxxxxxxW2237xxxx (indicating year 2022, week 37) or later, it means the package is equipped with Intel® 8$^{th}$ Gen Core™ i CPU.

## 2.16 UPDATE – EIP Builder Version 1.09 Release

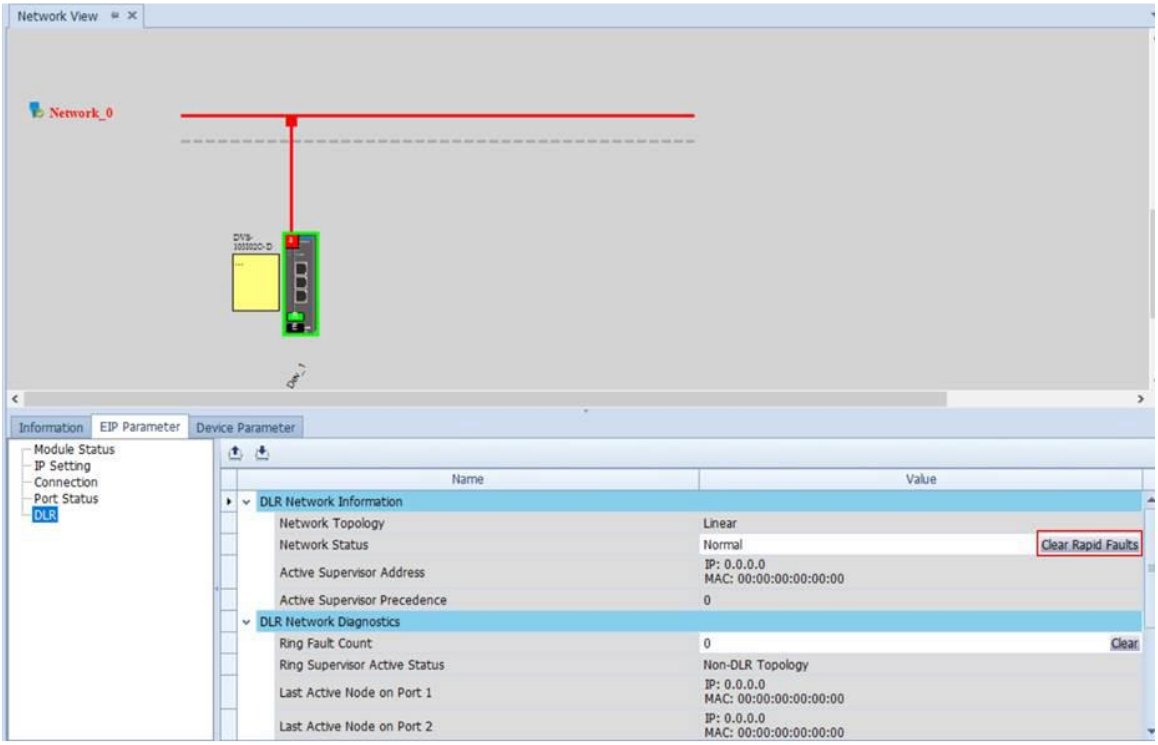
### Corrections




Delta releases the newest version of the EIP Builder software. This software tool supports Ethernet/IP network planning, data exchange planning, data upload and download as well as diagnosis for networks that include Delta products.
It facilitates IP address setting and import of EDS files.

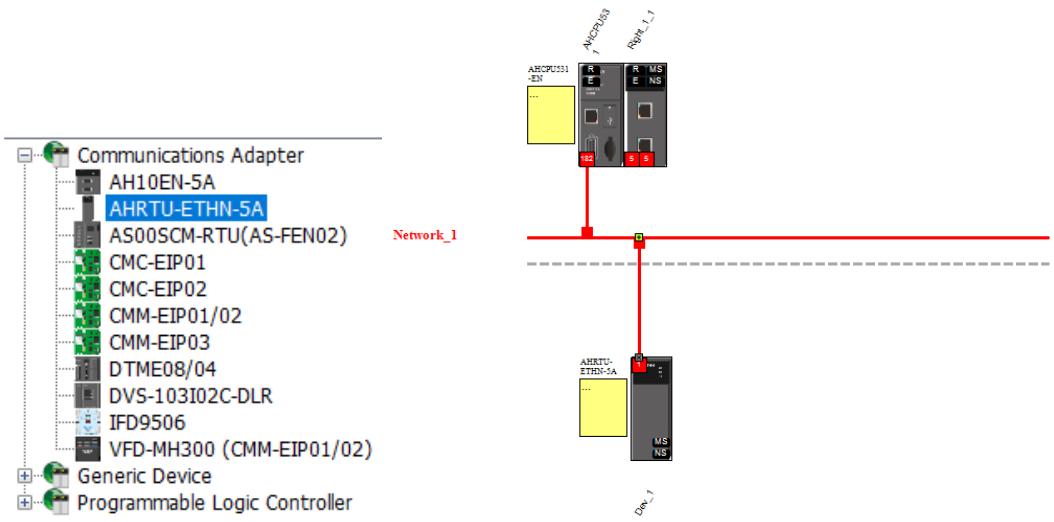
Description

1. EIP Builder added “Clear Rapid Faults” button; with one click, users can clear rapid faults of the network status by employing the explicit message “Clear_Rapid_Faults” (code: 16#4C)

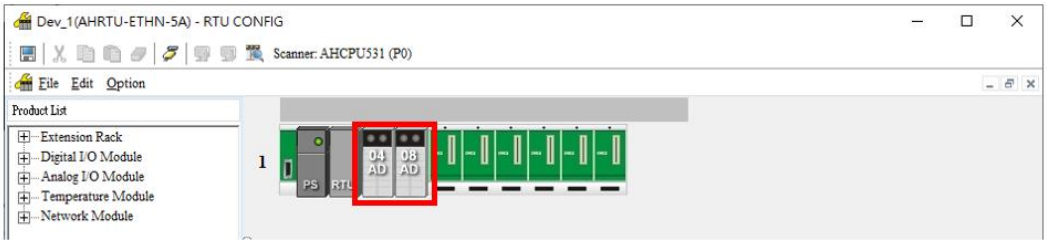



2. The Merge AIO Module function is available, when AHCPU5X1-EN (FW: V2.04 or later) or AHCPU560-EN2 (FW V1.10 or later) works with the remote modules, e.g. AHRTU-ETHN-5A (FW V1.04.0 or later) and AS00SCM-RTU (AS-FEN02) (FW V1.00.0 or later); all the data exchange connections can be combined to one connection. The number of CPU data exchange connections can be reduced and the number of remotely connected modules can be increased

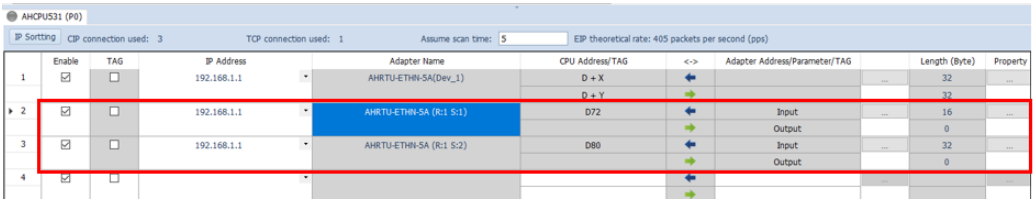
   2.1. When working with the remote module AHRTU-ETHN-5A (FW V1.04.0 or later): The combined connection for data exchange (CIP) can be used by AHRTU-ETHN-5A and its connected right-side AIO modules on the extension backplanes.



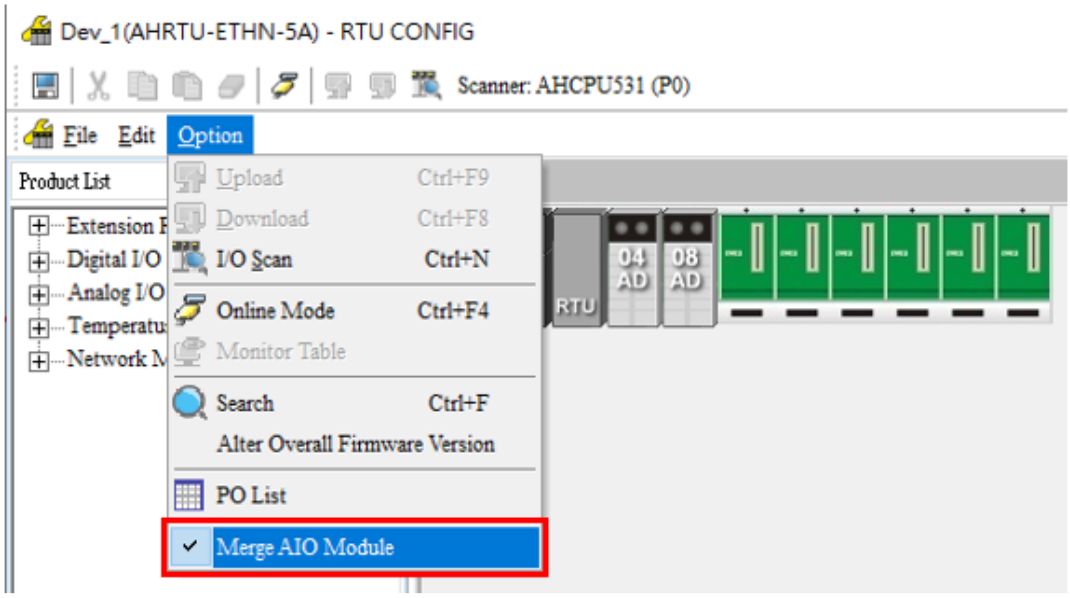

Example: The AIO modules AH04AD-5A and AH08AD-5A are connected to the right-side backplane of AHRTU-ETHN-5A for data exchange.



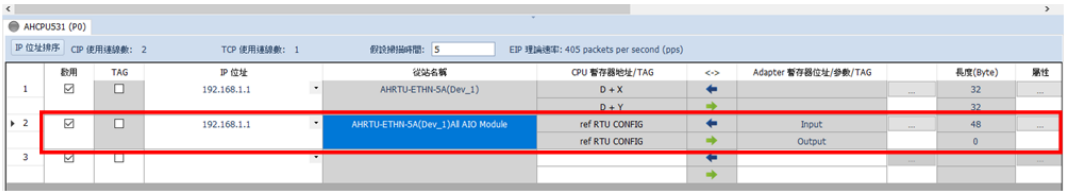

Previous versions allocated separate data exchange areas to the AIO modules.



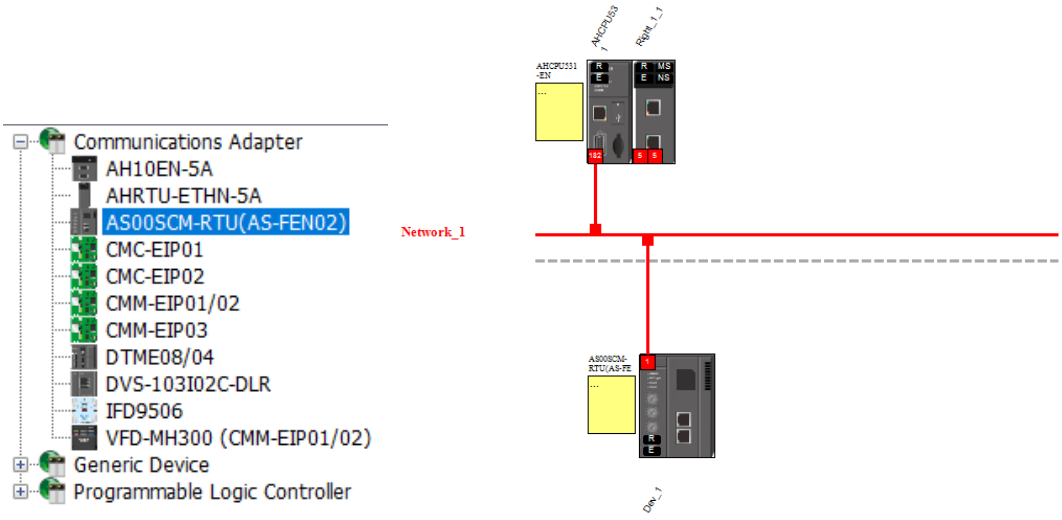
Version 1.09 provides the “Merge AIO Module” option.



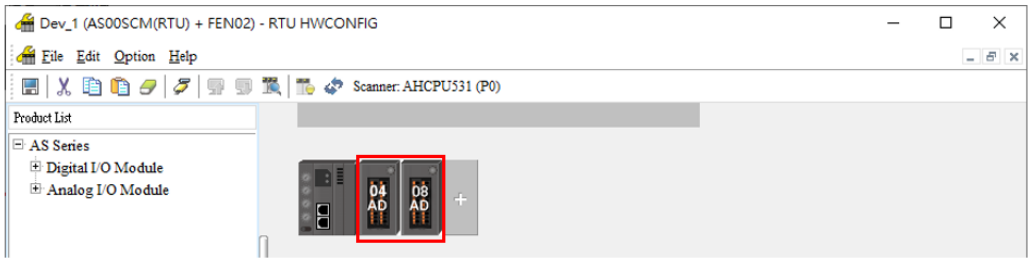

After activating that function, the AIO modules work like a single, larger module



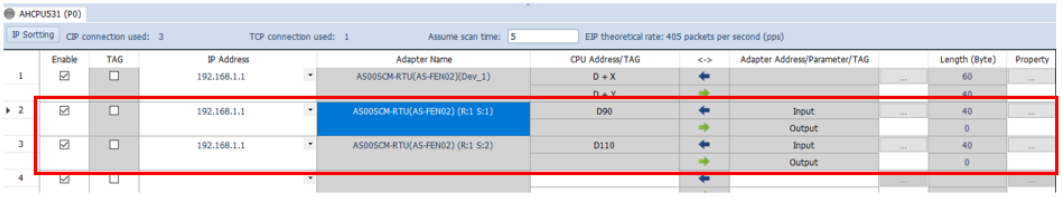
2.2. When working with the remote module AS00SCM-RTU(AS-FEN02) (FW V1.00.0 or later): The combined connection for data exchange (CIP) can be used by AS00SCM-RTU & AS-FEN02 and its connected right-side AIO modules



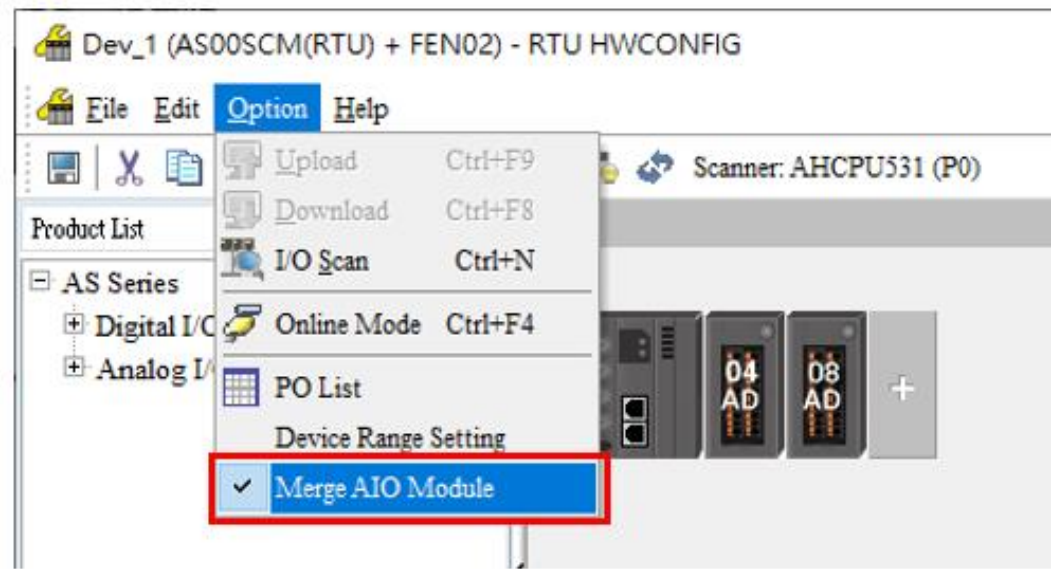
Example: The AIO modules AS04AD-A and AS08AD-C are connected to the right side of AS00SCM-RTU & AS-FEN02 for data exchange.



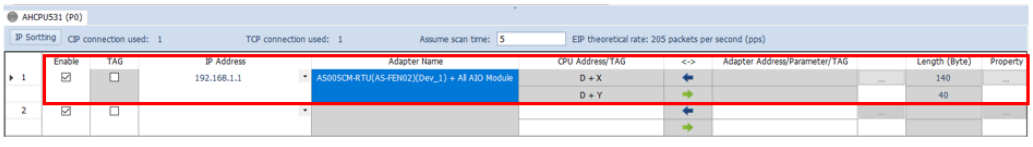

Previous versions allocated separate data exchange areas to the AIO modules.



Version 1.09 provides the “Merge AIO Module” option.



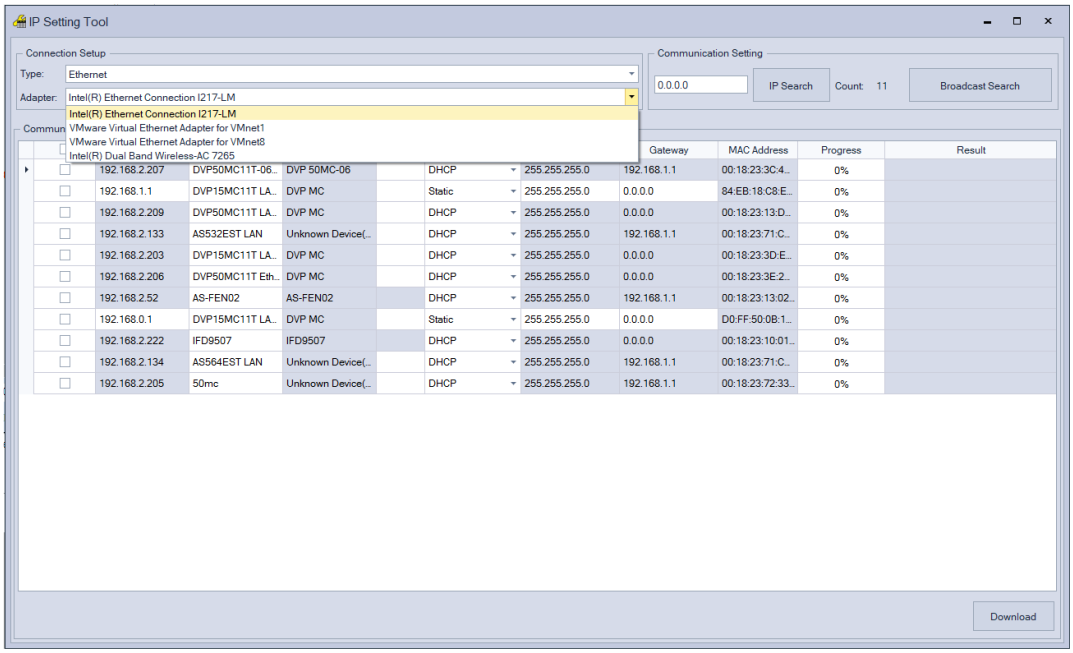

After activating that function, the AIO modules work like a single, larger module



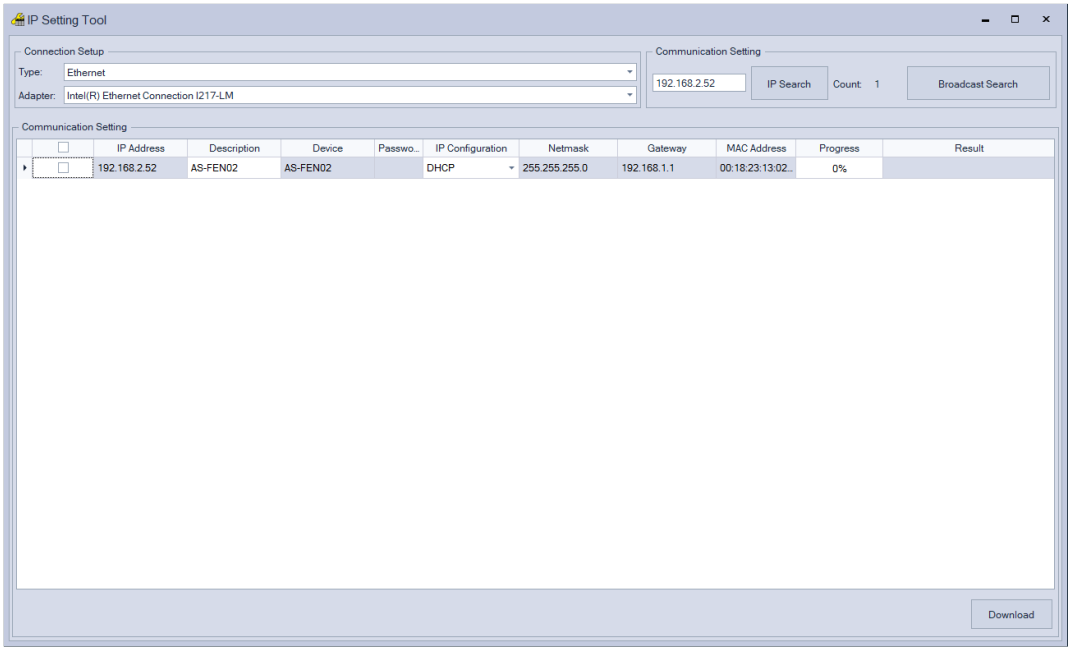
3. Added IP Manager Tool, available for the following Ethernet/IP devices. You can broadcast search, download and edit the Ethernet basic settings, including IP address, description, password, and IP configuration (Static/DHCP)

| Series | Modules |
|---|---|
| DVP | DVP-ES2-E, DVP-SE, DVP-MC, DVPEN01-SL, DVP-FEN01, RTU-EN01 |
| AS | AS500, AS-FEN02, AS-FOPC02, AS-FFTP01 |
| Others | IFD9506, DVS-103I02C-DLR, DTM-E, CMC/CMM communication cards |

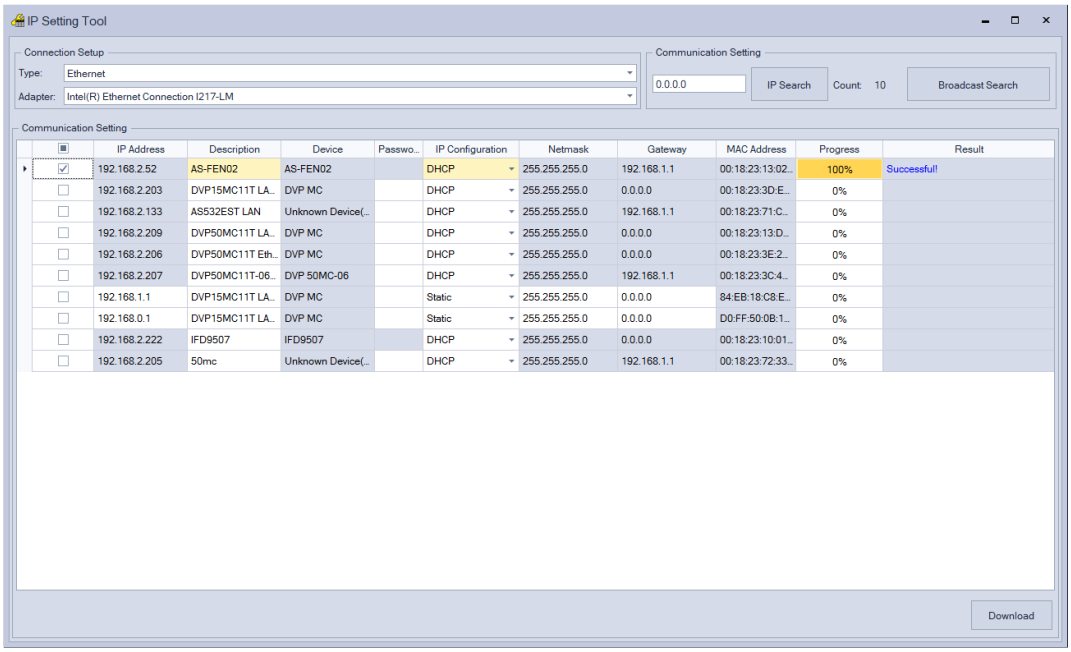
3.1. After selecting the corresponding adapter, you can use broadcast search to find a suitable Ethernet/IP device



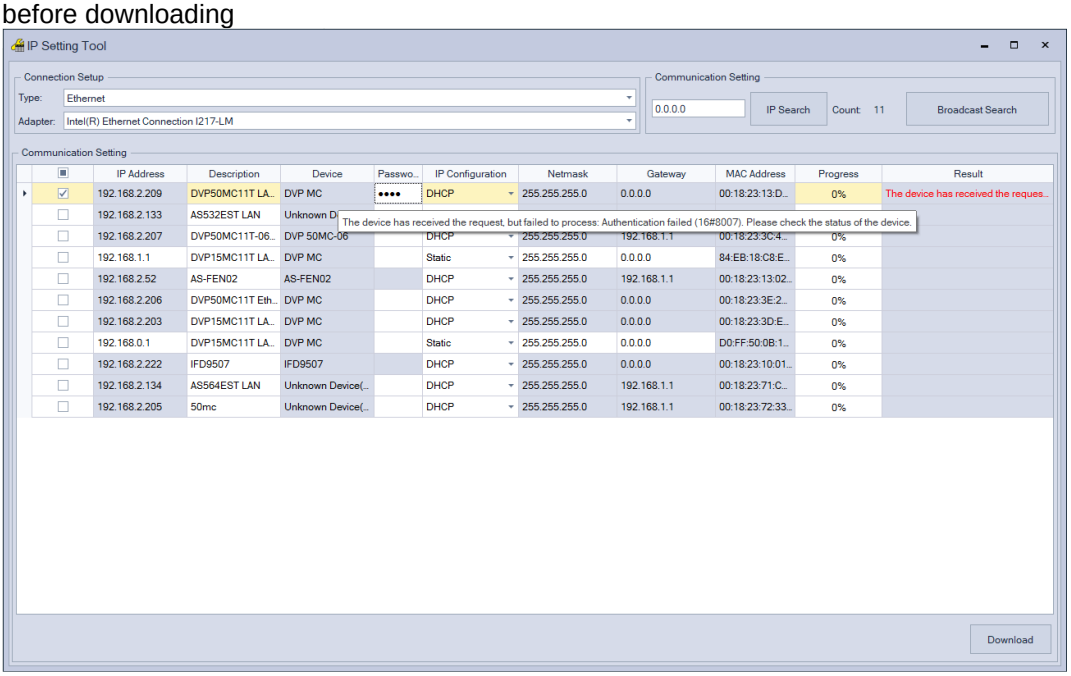

3.2. Enter an IP address to search for the specific one



3.3. Select the one you'd like to download and then click the Download button. After clicking the button, the result will be shown




3.4. If the one you selected is protected by a password, you will need to enter the password before downloading




## Download Link

[EIP Builder 1.09]

## 2.17 UPDATE – DIAEnergie Version 1.9 Release

### Added Functions

1. Tag Mapping -> Modbus Device -> Modbus Device Tag Add Dialog And Edit Dialog: added 4 Properties: Enable AFDD (Automated Fault Detection and Diagnosis), AFDD Classification, AFDD Suggestion, and AFDD Abnormal Condition
2. UI Design -> Real-time Data: added a new tool “AFDD Display”
3. UI Design -> Graph -> Comprehensive Chart Types: added a new tool “Single Bar”
4. UI Design -> Basic: added a new tool “Rectangle”
5. UI Design -> Real-time Data -> Meter: added a new property “Enable Disconnection Condition Status”
6. Device Topology -> Field device -> Gauge -> DELTA Power Meter: added more models, including DPM-DA510 and DPM-DA530
7. Device Topology -> Control device -> Programmable Logic Controller -> DELTA PLC: added one more model, AS-200

### Fixed Issues

1. Fixed the security vulnerability issues -ICS-VU-806371
    a. Clear text HTTP
    b. Cross-Site Request Forgery
2. Fixed the security vulnerability issues -ICS-VU-521908
    a. Arbitrary File Upload Remote Code Execution (ZDI-CAN-15580)
    b. SQL Injection Information Disclosure (ZDI-CAN-15581)
3. Fixed the security vulnerability issues - ICS-VU-032369
    a. Incorrect Default Permissions
    b. Blind SQL Injection #5 (unauthenticated)
    c. Blind SQL Injection #7 (unauthenticated)
    d. Blind SQL Injection #10 (unauthenticated)
    e. Blind SQL Injection #12 (unauthenticated)
    f. Blind SQL Injection #18 (unauthenticated)
    g. Blind SQL Injection #21 (unauthenticated)
    h. Blind SQL Injection #24 (unauthenticated)
    i. Blind SQL Injection #28 (unauthenticated)

j. Blind SQL Injection #29 (unauthenticated)

k. Blind SQL Injection #30 (unauthenticated)

l. Blind SQL Injection #31 (unauthenticated)

m. Blind SQL Injection #32 (unauthenticated)

4. UI Design → Right Click Context Menu: fixed the display issue found in align top, align bottom, align left, and align right
5. UI Design → Real-time Data → Alarm List: the updated data is incorrect
6. UI Design → Real-time Data → Event List: the updated data is incorrect
7. UI Design → Graph → Comprehensive Chart Types → Item Comparison: after sorting, the data shown is incorrect
8. Alarm → Alarm Group → Line notification: the notification cannot be sent

**Download Link**

DIAEnergie 1.9

## 2.18 TECHNICAL ANNOUNCEMENT – Avoidance of overheating EMI filters of REG2000

### Description

The capacitors in REG2000's EMI filter may heat up abnormally due to aging after exceeding its service life, resulting in the risk of fire. To avoid that risk, follow the instructions below to deal with different filter types

### Production period and service life of REG2000 EMI filter

If the EMI filter exceeds the service life, there may be safety problems. Please notify the customer to replace the filter with a new one immediately


Serial number is explained as follows:

xxxxxxxxW1915xxxx

19 : Year, 2019

15 : Week, 15th week

| Type | Production Period | Service Life |
|---|---|---|
| **Type A** | Produced before W1812 | 4 Years |
| **Type B** | Already recalled and changed to Type D | |
| **Type C** | RG-EF300A4, RG-EF550A4: W2009 ~ W2123<br>RG-EF150A4: W2013 ~ W2123<br>RG-EF110A2, RG-EF220A2, RG-EF370A2: W2050 ~ W2123 | 5 Years |
| **Type D** | Produced after W2124 | 8 Years |

## Suggested modification for REG2000 & EMI Filter in existing installations

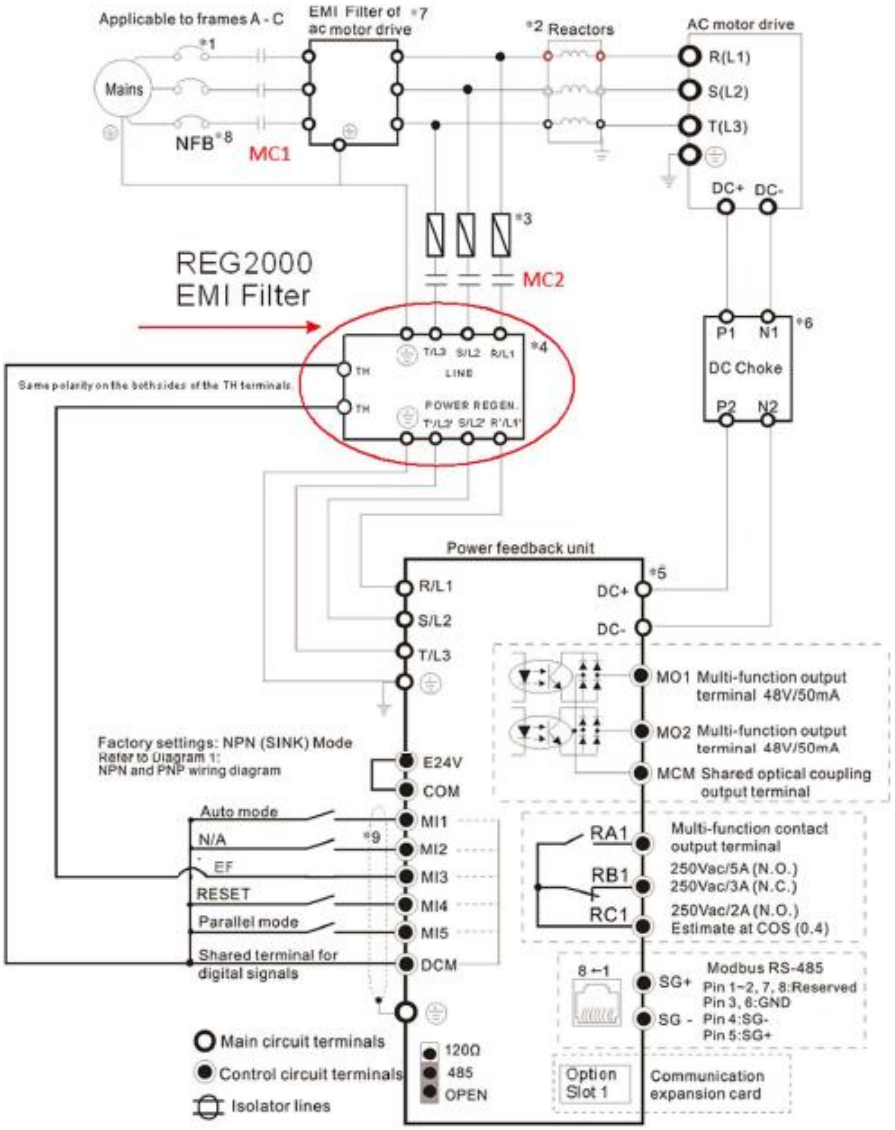
After confirming the Type of the EMI filter on the site, please follow the suggestions below to improve the situation

1) Type A (already out of warranty):
   Type A has exceeded its service life and the capacitor has begun aging. Please replace it with type D filter as soon as possible. And follow the suggested scheme from (3) below for improvement

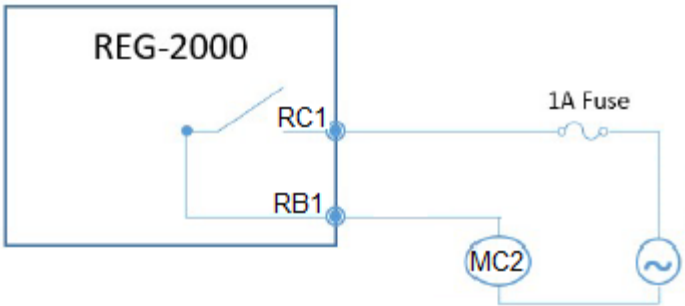
2) Type C (under warranty):
   When Type B end of service life is approaching, it should be replaced with Type D according to the suggested scheme from (3) below for improvement

3) Type D (under warranty) installation mechanisms:

    1. Install a contactor at the position MC2 in the figure below, to ensure that as long as the REG2000 detects an external fault (EF), REG2000 and filter remain disconnected




2. Connect contactor MC2 to the internal relay, set to Pr02-07 = 4



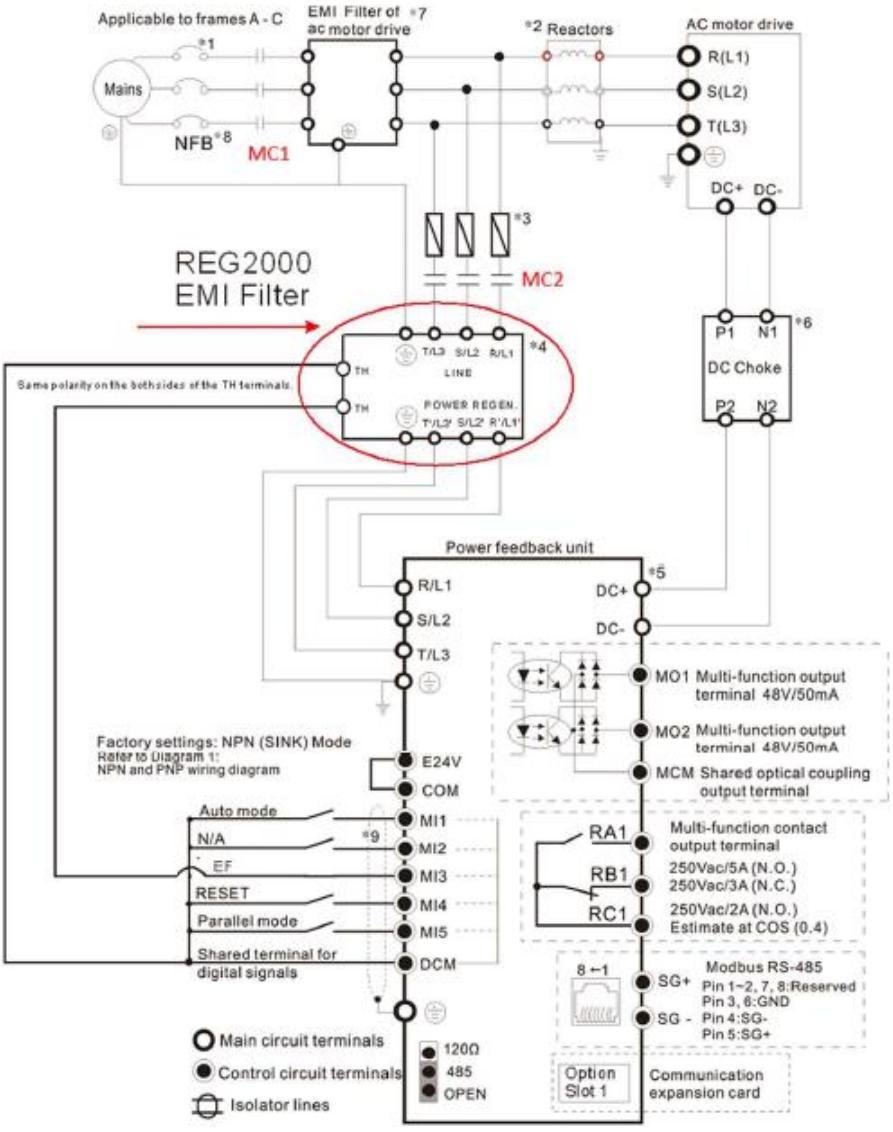
Suggested schematic for new installation

<Option-1>

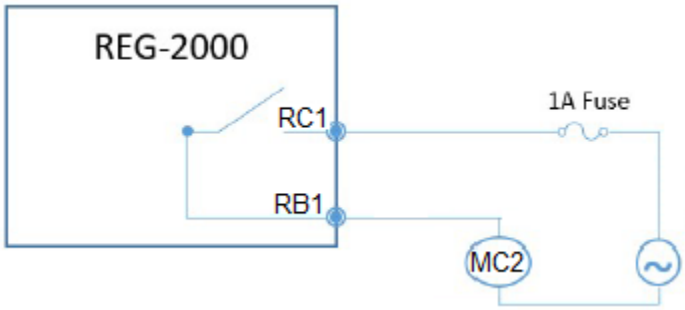
Connect the EIM filter's TH contact to the elevator controller. As long as the TH signal is high, the contactor (MC1 or MC2) should remain open. If MC1 is open, the elevator might shut down. Therefore, we suggest using MC2 instead.

<Option-2>

1. Add Contactor (MC2) like in the diagram below. As long as REG2000 detects the TH signal, it will show EF (External Fault) and shut down. The power to EMI Filter won't turn on again



2. Connect contactor MC2 to the internal relay, set to Pr02-07 = 4




# 3 Application

## 3.1 NEW – Technical Notes

**Drives**

DEN_IA_VFD_Control_by_Physical_Unit_TN_EN_20220915.pdf

DEN_IA_VFD_Motor_Control_Modes_TN_EN_20220914.pdf

DEN_IA_VFD_Sleep_Mode_TN_EN_20220915.pdf

DEN_IA_VFD_Tank_Level_Control_TN_EN_20220912.pdf

## 3.2 Update – Technical Videos, Tips and Trainings on Our YouTube Channel


https://www.youtube.com/c/DeltaIndustrialAutomationEMEA

Subscribe and enable notifications in order to get notifications on all our new videos.

# 4 FAQ

## 4.1 AC Motor Drives

**Variable Frequency Drives**

**Q How to match the technical data of a solar panel with a variable frequency drive?**

*A Multiply the AC voltage specifications of the drive with √2 to find the voltage specification of the solar panel. E.g. a 230 V drive has an operating range of 170 – 265 V AC, which means the solar panel needs to provide a nominal 240 V – 375 V DC peak voltage.*

*Find out the required maximum AC current of the motor at the available voltage. Since it is 3-phase AC current, multiply it with √3 x √2, i.e. ~2.45, to find the required DC current. E.g. if a motor draws 5 A of AC current, the panel needs to be able to provide ~12.3 A of DC current.*

*Assuming that the panel provides 325 V DC, which corresponds to 230 V AC, the calculation above leads to a 4 kW solar panel to drive a typical 0.75 kW motor.*

*Most of the time, the panel will operate at a fraction of its nominal power. This seems to be an exaggerated over-dimensioning. However, it is necessary due to the need to provide the peak DC voltage and DC current that corresponds to the required AC amplitudes, while the motor actually uses rms voltage and rms current.*

*It is possible to reduce the size of the solar panel by*

1. *Employing a DC/DC boost inverter with MPPT between solar panel and drive. While this may reduce the solar panel size, it adds cost and introduces losses into the system. It is necessary to assess whether the MPPT's positive effects and the reduced solar panel size level those disadvantages out*

2. *Adding batteries or large capacitors to buffer the difference between peak and rms current. However, such a battery or capacitor pack with its management system and additional maintenance requirement may cost more than the savings of the smaller solar panel. They need to be calculated beforehand*

## 4.2 Servo Systems

**ASDA-A2 and ASDA-A3/B3 Series**

**Q How to maintain all parameter settings after power cycling?**

*A Set bit Z of parameter 3-12 in the –A2 series or 3.012 in –A3/B3 series to 1, i.e.*

*3-12 or 3.012 = x1xx*

*Then cycle the power and change the parameters you require for the application.*