NCS4000 User Manual

Content

1 Start Tutorial

Welcome to NCS4000 industrial control system!

1.1. Introduction

The NCS4000 control system is an advanced distributed fieldbus control system, which is applicable in large-scale industrial control environment. It can provide the functions of process control, logical control and batch control， and can be widely used in metallurgy, cement, coke, sewage treatment, oil and chemical industry.

NCS4000 has following features:

- Support large-scale industrial control application, maximum load 16348 points；
- Support multifarious fieldbus standards, such as FF, PROFIBUS DP, AS-I, HART；
- Support redundancy, include controller redundancy, network redundancy, I/O module redundancy and power redundancy；
- Support device management, diagnosis and maintenance；
- Support distributed configuration, offline configuration and simulation；
- Support IEC61131-3 programming standard；
- Open data access interface, support OPC standard；
- Integrate advanced process control technology, and provide industrial solutions.

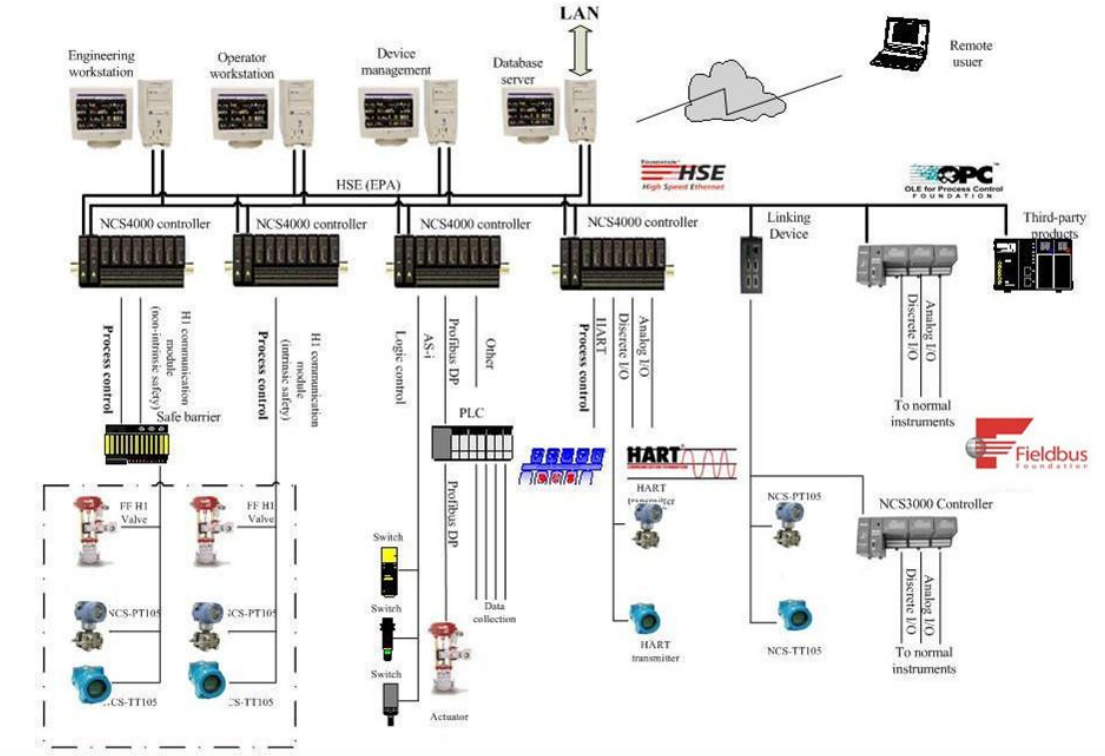


Figure 1 NCS4000 System Architecture

NCS4000 is a complete distributed fieldbus control system, which is composed of human-computer interface, control stations and field devices. The field device conforming to fieldbus communication protocol standard includes intelligent sensors, transmitter, actuator, and other instruments. The following figure depicts the architecture of the NCS4000.

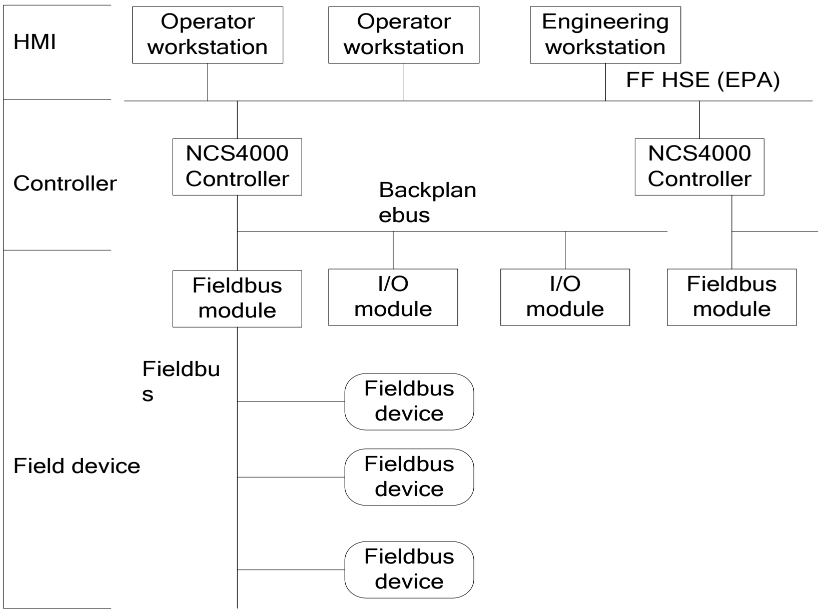


Figure 2. NCS4000 System Hierarchy

NCS4000 control system includes control stations, workstations, fieldbus devices and control networks. In NCS4000, a network can include 32 control stations at most. Workstation is made up of industrial computers, and the number of workstations in NCS4000 is not limitative.

The following figure describes a basic constitute of NCS4000:

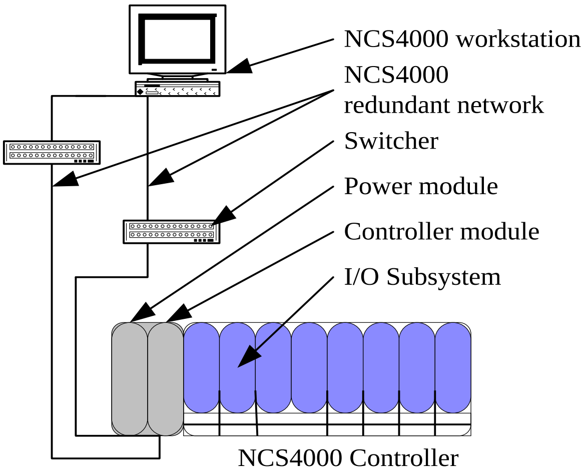


Figure3. NCS4000 control station

The NCS4000 system is consists of:

1. At least one NCS4000 control station. The control station completes the sampling of field data, computation control, network commutation, devices diagnosis and maintenance. It is made up of controller module, I/O subsystem and power module, and it is the kernel device of the NCS4000 control system.
2. At least one NCS4000 workstation. The workstation is maybe a normal industrial computer, or a normal PC. It is able to control the configuration of application, display HMI graphics and record data history.
3. Network connecting devices, such as switches.
4. Control network. The control network conforms to the definition of fast Ethernet of Foundation Fieldbus standard-FF HSE. It provides 100 Mbps data transferring rate with network redundancy.

1.2. Software installation and updates

To install the software system of NCS4000, i.e. software on NCS4000 workstation, it is requesting:

- Hardware requirements
- Software requirements

When the software and hardware requirements are both satisfied, the installation can be started. To execute NCS4000Setup.exe and go on continue，please follow the prompts.

1.3. Hardware requirements of NCS4000 software system

Hardware requirements of NCS4000 software system are:
x86 series computer

device	requirement	recommended
CPU	1 Ghz	P4 2.00 Ghz or upper
memory	256MB	512MB
Hard disk	50MB available	100MB available
monitor	800×600	1024×768
Network card	10/100M	10/100M

It is recommended to use higher configurations for the main engineer station or the computer running OPC server.

1.4. Software requirements of NCS4000 software system

Software requirements of NCS4000 software system are:

OS	WIN2000 + SP4 WIN XP + SP1/SP2
Python parser	2.4.3
MSXML parser	4.0 SP2
WinPcap (optional)	3.0 or later Used by data packages analysis tool

Other requirements for software system are explicated in “initial configurations” later.

1.5. NCS4000 initial configurations

Before use of NCS4000 software system, it is necessary to do some initial configurations.
NCS4000 system requires all the workstations and controllers to connect on the same network segment, and it’s recommended to connect them into the same switch or Hub.
The IP address of each of workstations and controllers is configured within the same network segment, such as 192.168.1.*.

By default, after the NCS4000 software is installed on a workstation, the workstation is automatically set as a main engineer station and runs system information database. When it needs multi workstations to cooperate with each other, users must set main engineer station’s IP address on the non-main engineer stations after installation. User can complete this operation by employing the system information database management tool.

In the case of cooperation, the main engineer station communicates with other workstations, and it also needs to configure the DCOM privileges for each workstation. Browse the details.

1.5.1. NCS4000 system information database DCOM configuration specification

In the NCS4000 system, when each workstation is cooperating with each other, all of them share a single system information database, and the workstation on which the system information database runs is called main engineer station. In order to successfully communicate between each of the engineer station that is the client of system information database and the main engineer station is the server, users need to configure the DCOM privileges of the system information database.

Note: It is assume that all the workstations in NCS4000 system are in workgroup mode, not in NT field.

(1) Configure OS user

In the simplest case that the security is not considered, all the workstations can be logged on with the same user name and password, and the user has Administrator privileges.

The more secure configuration:

It is assumed that the main engineer station is named P, and other workstations are named A, B, C…

First, build a user with the same user name and password on all of the workstations, and the user needn’t have Administrator privileges, such as *NCSCommonUser*. The main engineer station P is logged on with this user name.

If the user wants to log on the workstation A with the user name of *NCSUserA*, then the *NCSUserA* must be also built on the main engineer station P with the same password, but needn’t have Administrator privileges. Analogically, build the users of other workstations, such as *NCSUserB, NCSUserC,* and so on.

Build a workgroup named *NCSUserGroup* on the main engineer station P, and add the users *NCSUserA, NCSUserB, NCSUserC* to the workgroup.

(2) DCOM privileges configuration

First, set the DCOM privileges of the system information database on the main engineer station. Use the system tool dcomcnfg.exe to set the identification for the system information database service routine **SysDbSrv**, default is “**startup user**”, modified to “**interactive user**”.

Then, configure the access and startup privileges of the system information database on the main engineer station. There are two methods. The first one is use default security mechanism, i.e. the default access and startup privileges. In the default DCOM security mechanism of the main engineer station, the *NCSUserGroup* is allowed to **startup and access** the system information database. The second method is defining the system information database’s own access and startup privileges, also permits *NCSUserGroup*.

Finally, configure DCOM privileges on other workstations, and permits *NCSCommonUser* with access privileges.

Note: 1. For Windows XP operating system, particularly the installation of SP2, in the control panel to ensure that in the *local security strategy: Local Strategy -> security options* the *network access: the sharing of local accounts and safe mode* set to *classical - local users to verify their identity*.

2. The interface and usage of dcomcnfg.exe vary in different operating system. See “OPC DCOM privileges configuration” for details.

1.6. Example

Here we provide a simple example, in order to show the design flow of an application project based on NCS4000 control system.

1. Create system information database;
2. Design control strategy structure;
3. Edit Function Block Diagram module arithmetic;
4. Edit Ladder Diagram module arithmetic;
5. Configure inter-module communication ;
6. Download configuration information;
7. Online monitoring.

1.6.1. Create system information database

- In “Start” menu, select “Program->NCS4000->System information database management”, and open system information database management tool.
- In the NCS4000 system information database management tool，double click“New database”icon，this opens a “New database” dialog.
- In the “New database” dialog，type “SysDb1” in the name field，and leave other items set to default value，then click “OK”.

- Click “Yes” button in the “Database SysDb1 is created successfully. Set it to active database or not?” message box, and click “OK” in the message box “Active the database successful”.
- Close the NCS4000- system information database management tool.

1.6.2. Design control strategy structure

- In “Start” menu, select “Program->NCS4000->Resource manager”, and open NCS4000 resource manager.
- Create area object.
  - In the view of resource manager，click your right mouse button and select “System configuration->control strategy”.
  - Select “Create new area…” from the pop-up menu, and an area object node “Area1” will be added under the “control strategy” node.
- Create module object.
  - Select the “Area1” and right-click it, and select “Create new module…” from the pop-up menu, this opens a “New module” dialog.
  - In the “New module” dialog，type “Fdb1” in the name field，then select “Function
  - Block Diagram” in “New module object” item and leave other items set to default value，then click “OK”. A module object node “Fbd1” will be added under the “Area1” node.
  - Use the same method to add a “ladder diagram” object node “Ld1” under the “Area1” node.

1.6.3. Edit Function Block Diagram module arithmetic

- In the view of resource manager，click your right mouse button and select “System configuration->control strategy->Area1->Fbd1”.
- Select “Edit->Use configuration software offline edit” from the pop-up menu, this open the NCS4000 control strategy design environment.
- Add function block object
  - In the template view of control strategy design environment, select “function block library->input/output->DI”.
  - Use your mouse to “drag and drop” the graphics object of DI function block and add the module object “Fbd1” to the edit view. At meanwhile, a DI function block object node “DI_1” will be added under the “Area1” node in the module view.

- ■ Use the same method to add a DO function block object “DO_1”.
- Add parameter object
  - ■ In the module view, right-click the “Fbd1”.
  - ■ Select “Add parameter” from the pop-up menu，and a parameter object node “Param_1” will be added under the “Fbd1” node.
  - ■ In the attribute view, modify the name of “Param_1” to “In1” and modify the “logic I/O” field to “Input”.
  - ■ Use the same method to add another parameter object with the name “Out1” and
    set the “logic I/O” field to “Output”.
  - ■ Use your mouse to “drag and drop” the graphics objects of “In1” and “Out1”, and
    add the parameter objects to the edit view of “Fbd1”.
- Ensure connect relation.
  - ■ In the toolbar of the edit view, select “connect” icon.
  - ■ In the edit view，draw a wire from the output port “OUT_D” of function block “DI_1” to input port of parameter object “Out1”.
  - ■ Use the same method to draw another wire from the output port of parameter object
    “In1” to the input port “CAS_IN_D” of function block object “DO_1”.
- In the module view, right-click the “Fbd1”. Select “Validity check” from the pop-up menu. Then, in the output view, you can see success information.
- Save the configuration, and exit the control strategy design environment.

1.6.4. Edit Ladder Diagram module arithmetic

- In the view of resource manager，click your right mouse button and select “System configuration->control strategy->Area1->Fbd1”.
- Select “Edit->Use configuration software offline edit” from the pop-up menu, this open
  the NCS4000 control strategy design environment.
- Add graphics object
  - ■ In the edit view of control strategy design environment, select “add network” icon
    on the toolbar, and add two network objects. The network identifications for each network object is “000001” and “00002” .
  - ■ In the edit view, select horizontal network connect graphics object under the network object “000001”, then click the “add winding” icon on the toolbar to add a winding graphics object.
  - ■ Click the “add connecting point” icon on the toolbar to add a connecting point
    graphics object in the front of the winding graphics object.
  - ■ Use the same method to add a winding graphics object and a connecting point

graphics object for the network object “000002”.

- Add parameter object
  - In the module view, right-click the “Fbd1”.
  - Select “Add parameter” from the pop-up menu，and a parameter object node “Param_1” will be added under the “Fbd1” node.
  - In the attribute view, modify the name of “Param_1” to “In1” and modify the “logic I/O” field to “Input”.
  - Use the same method to add another parameter object with the name “Out1” and

    set the “logic I/O” field to “Output”; add a parameter object with the name “Local1” and the “logic I/O” field to “Interior”; add a parameter object with the name “Local2” and the “logic I/O” field to “Interior”.
- Associate parameter object with graphics object
  - In the edit view of module, select “Ld1->In1” node.
  - Use your mouse to “drag and drop” the parameter object “In1” and associate it with

    the connecting point graphics object under the network object “000001” in the edit view. If operate successfully, the name of the connecting point graphics object will be update to “In1”.
  - Use the same method to associate the parameter object “Local1” with the winding

    graphics object under the network object “000001”, associate the parameter object “Local2” with the connecting point graphics object under the network object “000002” and associate the parameter object “Out1” with the winding graphics object under the network object “000002”.
- In the module view, right-click the “Fbd1”. Select “Validity check” from the pop-up

  menu. Then, in the output view, you can see success information.
- Save the configuration, and exit the control strategy design environment.

1.6.5. Configure inter-module communication

- In “Start” menu, select “Program->NCS4000-> control strategy design environment”,

  and open control strategy design environment.
- In the control strategy design environment, select“Application->Inter-module communication”, then open the “Configure inter-module communication” dialog.
- Add communication configuration items
  - In the dialog of “Configure inter-module communication”，click “Add” button. A

    communication configuration item “Item_1” is added into the item list, and the attribute of the “Item_1” is displayed in the attribute list.
  - In the attribute list, modify the “Locality area” field of “Input parameter” to “Area1”; modify the “Locality module” field to “Fbd1”; modify the “Parameter

name” field to “In1”.

- In the attribute list, modify the “Locality area” field of “Output parameter” to “Area1”; modify the “Locality module” field to “Ld1”; modify the “Parameter name” field to “Out1”.
- Use the same method to add another communication configuration item. Its input parameter is “Area1->Ld1->In1”, output parameter is “Area1->Fbd1->Out1”.

- Save the communication configuration, close “Configure inter-module communication” dialog, and exit the control strategy design environment.

1.6.6. Download configuration information

- Create virtual console
  - In the view of resource manager，click your right mouse button and select “System configuration->control network”.
  - Select “Create a new controller…” from the pop-up menu, and a controller object “Controller1” is added under the “control network” .
- Map controller
  - Right-click the “Controller1”, and select “Map controller…” from the pop-up menu, then open a “Object browse” dialog.
  - In the “Object browse” dialog, select an available physics controller object, then click “OK”.
- Assign module
  - Right-click the “Controller-> control strategy module list”, and select “Assign…->Area…” from the pop-up menu, then open a “object browse” dialog.
  - In the “Object browse” dialog, select the area object “Area1”, and then click “OK”. Prompt the message that the “Area1” is assigned to the “controller1”.
- Download module
  - Right-click the “Controller1”, and select “Download…->Whole controller…” from the pop-up menu, then open a “Controller1 download configuration” dialog.
  - In the “Controller1 download configuration” dialog, click “Download” button and start download.
  - Download successfully, and click “Close” button, then the “Controller1 download configuration” dialog is closed.

1.6.7. Online monitoring

- Start online monitor
  - ■ In the view of resource manager，click your right mouse button and select “System configuration->control strategy->Area1->Fbd1”.
  - ■ Select “Edit->Use configuration software offline edit” from the pop-up menu, this opens the online monitor window of “Fbd1”.
  - ■ Use the same method to open the online monitor window of “Ld1”.
- Monitor online
  - ■ In the module view of the “Fbd1” online monitor window, select “Fbd1->DI_1”, then all the values of parameter objects of function block object are displayed in the monitor view.
  - ■ In the module view of the “Ld1” online monitor window, select “Ld1”, then all the values of parameter objects of function block object are displayed in the monitor view.
- Control online
  - ■ In the monitor view of the “Fbd1” online monitor window, modify the “VALUE” field of parameter object “OUT_D” under function block object “DI_1” to “Discrete state 1”; click “Write parameter” button，and write value to field device.
  - ■ In the monitor view of the “Ld1” online monitor window, modify the value of parameter object “Local2” under function block object “Ld1” to “1”; click “Write parameter” button，and write value to field device.
- Inter-module communication
  - ■ The current value of parameter object “In1” under module object “Ld1” changes along with the parameter object “Out1” under module object “Fbd1”.
  - ■ The current value of parameter object “In1” under module object “Fbd1” changes along with the parameter object “Out1” under module object “Ld1”.

1.6.8. Note

When using the current version of NCS4000 to configure project, it should be noted that:

Capability of system information database

In the system information database, it permits the user to build at most 255 areas in each project database, and it is allowed to build at most 255 modules in each area and add at most 255 standard function blocks (defined by H1 or users themselves) in each FBD module. But right now, it is recommended that the number of function block in a NCS4000 project database should be less than 10000.

Scale of ladder diagram arithmetic

By now, there is no specific limit of the length of ladder diagram arithmetic for NCS4000 controller, but if the ladder diagram arithmetic is too long, the resource may be not enough. If it prompts that program downloading fails, user need to check the length of ladder diagram arithmetic. Otherwise, during the time of configuring ladder diagram, there should be at most 1024 parameter objects, which include all the parameter objects under module objects and user-defined function block (instance, reference) objects.

Module loop cycle

By now, the loop cycle for each module (including FBD and LD) is 1000 ms by default. When the quantity of the configuration is large, the loop cycle of 1000 ms may be a little short. It leads to that the load of controller is too big to run.

At present, NCS4000 controller ensures 255 modules to run simultaneity and can include 3000 function blocks in total. In this case, set the loop cycle of each module to 3~5 seconds to ensure the controller work orderly. Users should consult this index.

In practice, if the load of controller is too large to work orderly, users can clear the configuration information in controller by eliminating static data and improve control strategy, then download again.

2. System configuration

2.1. System information database

2.1.1. Introduction

In the NCS4000 system information database (SysDB), it stores all the configuration information of project, such as control arithmetic module, virtual controller configuration, user-defined function blocks and template library.

The NCS4000 system information database is different from the traditional database based on files, further more, it supports distributed subsequently configuring. Several configuration engineers can exploit control strategy simultaneity based on the network, and this will increase design efficiency and consistency.

2.1.2. Master engineer station

For a distributed control system network, in order to ensure the consistency of configuration, there should be only one computer, which is called master engineer station, executing system information database server. Other workstation must be connected to the master engineer station to access the information of system configuration.

2.1.3. Subsequent control

In the NCS4000 system information database, it is allowed multi-procedure and multi-user to exploit control strategy subsequently. For example, engineer A and B work with different computers, and when engineer A is editing a module, at the same time, engineer B can modify another module in the project. Otherwise, the system information database is able to harmonize with multi-procedure. For example, when engineer A is editing the module M using the control strategy design environment, if engineer B wants to delete the module M in NCS4000 resource manager, the operation of deleting will fail and the system will warn engineer B that the module is locked.

2.1.4. Exception disposal

In practice, some exceptions maybe happens, such as system power off, network congestion. In the example above mentioned, when an exception happens, i.e. engineer A's computer powers off when he is editing the module M (the Computer engineer A used is not the master engineer station), at the same time, if engineer B wants to delete the module B, the system will also warn engineer B that the module is locked.

The NCS4000 system information database checks the connections of clients. After engineer A has disconnected form the database for 6 minutes, the database will release the user. If the engineer B doesn't want to wait for that time, he can stop the system information database server on the master engineer station and then restart the server.

2.2. Control strategy hierarchical structure

In the NCS4000 control system, the control strategy structure is based on module objects. A module object is a control entity in logic, and it can complete certain control tasks, such as a control loop. Modules don't influence each other, and engineers can modify a module without affecting other modules.

Module objects are assigned to several area objects. An area object is a higher abstract concept than the module objects. The area objects don't substantively influence the control strategy, and they only divide the whole control strategy into several portions in order to make users to manage the module objects of different types in different ranges.

According to the types of control arithmetic, the module object in NCS4000 control system can be sorted into two types: function block (FBD) module and ladder diagram (LD) module.

The control arithmetic of the FBD module is composed of standard function blocks, user-defined function blocks and the connecting relationship between the function blocks.

NCS4000 controllers support tens of standard function blocks, users can make use of these standard function blocks to design user-defined function blocks.

The control arithmetic of the LD module is made up of ladder diagrams. The ladder diagrams of LD module conform to the IEC61131-3 standard.

The NCS4000 system control strategy architecture is given in the following figure:

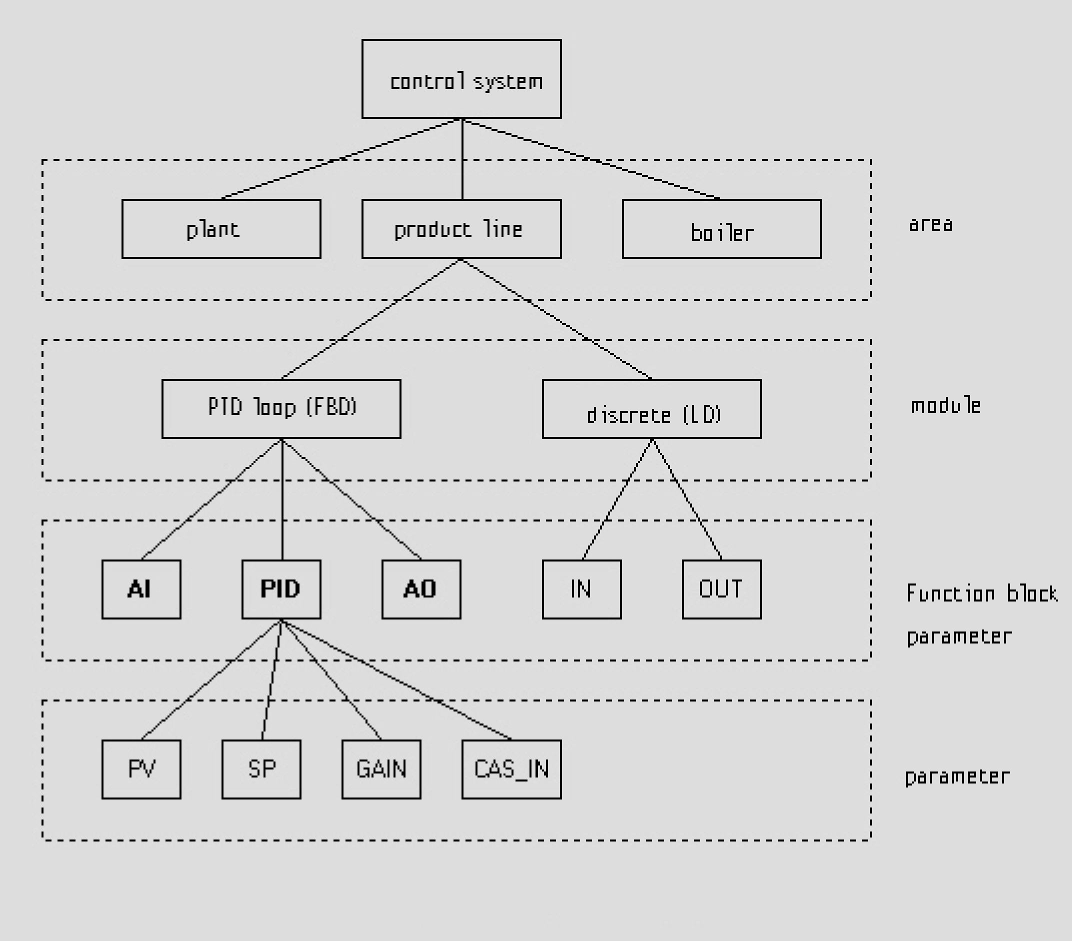


Figure The NCS4000 system control strategy architecture

2.2.1. Area

Area object

In the whole control system, the area object is a logical concept, and it needn't to be mapped to a physics object or a range. In practice, it usually corresponds to a field location, such as a plant, or an unattached process.

The area module contains module objects. The number of the module objects in each area object couldn't exceed 255.

The definition of the area is flexible. Engineers decide how to divide the whole control system into several areas according to the actual situation, so that they can effectively manage the configuration strategy in the control system.

At present, in each NCS4000 project database, it is able to build at most 255 area objects.

The name of area object can contain, but can't include special characters of "! @ $ ^ &?". The max length of area object's name is 32 bytes (A Chinese characters is 2 bytes).

2.2.2. Module

Module object

A module object is an application entity that can complete certain control tasks solely. A module object could be seen as an application process running on a NCS4000 controller. Tens of module objects are distributing to several controllers in the control network and make up of the whole NCS4000 control system.

By now, users can use two sorts of graphic languages, which are function block diagram (FBD) and ladder diagram (LD), to exploit module objects in the NCS4000 system. After exploiting arithmetic of module objects using the control strategy design environment, users can employ NCS4000 resource manager to download each module to controllers.

A NCS4000 controller supports at most 255 modules, and each module object has its own loop running cycle.

2.2.3. Virtual controller

Introduction

A virtual controller is abstract of physical controller, and it need not exist or be connected. But virtual controller has many features of physical controller, for example, it can be assigned control arithmetic (module), added and configured I/O modules and channels.

The motive of providing virtual controllers in NCS4000 system is that engineers can configure control strategy in despite of they don’t know the actual hardware devices. The offline configuration can prevent exploitation halt caused by deferred arrival of devices and provide a clear boundary between the control arithmetic and physical device, so that the users can easily maintain the system.

Virtual controller configuration
Users can add virtual controllers arbitrarily into the control network. The configuration of virtual controllers includes two aspects: control strategy module assignment and I/O mapping configuration.

- Control strategy module assignment
  Users can assign modules in arbitrary areas of current project to virtual controllers, but each module has to been assigned to only one virtual controller.

  The modules, not only FBD modules but also LD modules, in different areas can be assigned to a single virtual controller; the modules in a single area can also be assigned to different virtual controllers.

  Each virtual controller could contain at most 255 modules, and each physical controller dose.

- I/O mapping configuration
  Without physical devices, uses can configure virtual controllers’ I/O communication modules offline, i.e. directly add I/O communication modules that will be used in future. When physical devices are online, such as NCS4000 controller and I/O communication modules, NCS4000 resource manager will read the information of actual I/O modules and map them to the configured virtual I/O modules. The mapping is based on the location of slot in which the module is inserted.

  For example, if users insert an AI module in Slot 1 at the time of configuring the virtual controller offline, but a DI module is inserted in Slot 1 of the physical controller, then the resource manager will warn that types mismatch. At this time, users can change configuration or insert an AI module into Slot 1 of the physical controller instead.

  If users haven’t configured I/O modules for the virtual controller offline, after the physical controller updates, the resource manager will list all the actual I/O modules, so that the types of modules will not mismatch. However, in this case, users cannot map I/O modules and channels beforehand, and this part of the work has to start after devices are ready. User should decide how to configure I/O modules according to the actual situation.

2.3. Control strategy development

2.3.1. Control arithmetic

NCS4000 control arithmetic

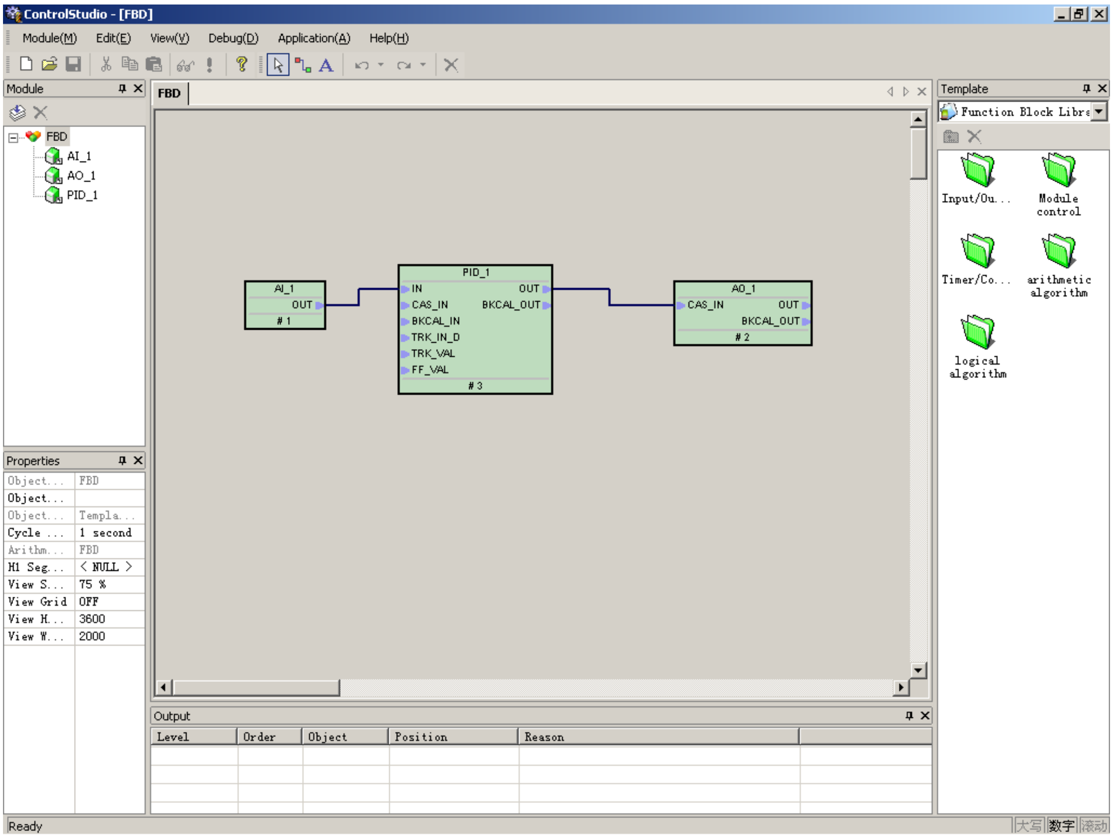
NCS4000 module control strategy supports two sorts of graphic languages: function block diagram (FBD) and ladder diagram (LD).

Function block diagram approaches Foundation Fieldbus standard, and all the I/O parameters have state information. It is mainly used for process control. Ladder diagram conforms to IEC61131-3 standard, and it is mainly used for discrete control. In the NCS4000 system, each module has only one type of arithmetic, but the modules with different types of arithmetic can transmit data with each other (See inter-module configuration for details).

Each arithmetic module has a loop cycle, which is 1000 ms by default. Note: when the arithmetic of the module is complex, or when too much arithmetic modules are assigned to a single controller, engineers need to prolong the loop cycle properly.

2.3.1.1. Function block diagram

In NCS4000, function block diagram is a kind of graphic control strategy design language, which is similar with FF. The NCS4000 provides nearly 40 standard function blocks including I/O, control, arithmetic, boolean calculation and so on. NCS4000 controllers support all these function blocks.



Properties

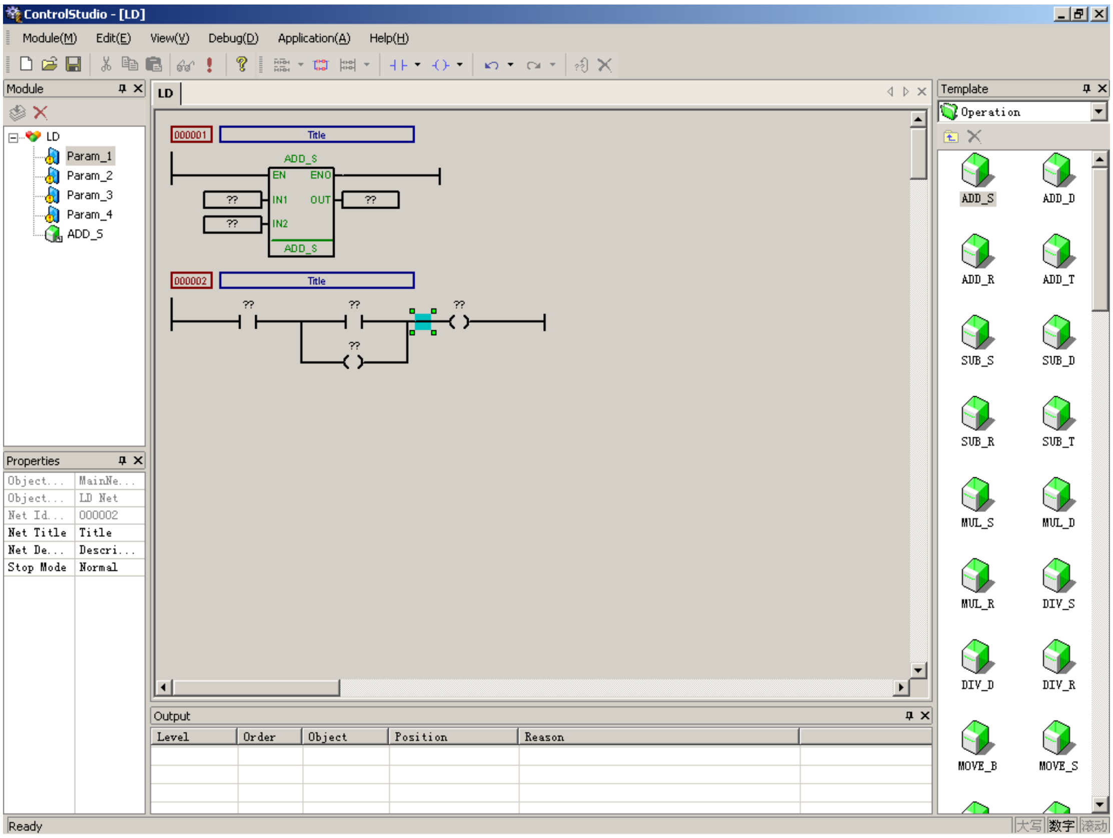
Object...	FBD
Object...	
Object...	Templa...
Cycle ...	1 second
Arithm...	FBD
H1 Seg...	< NULL >
View S...	75 %
View Grid	OFF
View H...	3600
View W...	2000

Level	Order	Object	Position	Reason

2.3.1.2. Ladder diagram

NCS4000 ladder diagram arithmetic conforms to IEC61131-3 standard.

For ladder diagram arithmetic modules, NCS4000 supports offline simulating and debugging.



Properties

Object...	MainNe...
Object...	LD Net
Net Id...	000002
Net Title	Title
Net De...	Descri...
Stop Mode	Normal

Level	Order	Object	Position	Reason

2.3.1.3. I/O mapping

I/O mapping builds corresponding relationship between input/output function blocks or parameters in control strategy and I/O communication modules or channels in controllers.

Users can employ I/O mapping tool to complete the I/O mapping, or directly configure the I/O modules or channels in the NCS4000 resource manager.
The rules of I/O mapping are:

- I/O modules and their channels can only be mapped to the function blocks and parameters in the modules of the virtual controller that the I/O modules have been assigned to.
- Each I/O modules' channel corresponds to a one-way I/O function block in the control strategy, such as: AI, AO, DI, DO, PI.
- The whole I/O module can be mapped to a multi-way I/O function block in the control strategy, and the type must match.
- For input I/O module (AI, DI, PI, TC and RTD), it can be mapped to multi function blocks or parameters.
- For output I/O module (AO, DO), it can only be one-to-one mapped. That means if the whole module has already been mapped to a multi-way I/O function block, the channels of the module can not be remapped; conversely, if certain channels have been mapped, the whole module can not be remapped.

I/O mapping table

I/O communication module	Function block mapped to module	Function block mapped to channel
AI-8 channels	MAI8	AI
HART AI-8 channels	-	AI
AI-16 channels	MAI16	AI
AO-8 channels	MAO8	AO
DI-8 channels	MDI8	DI
DI-16 channels	MDI16	DI
DO-8 channels	MDO8	DO
DO-16 channels	-	DO
PI-8 channels	-	PI
RTD-8 channels	MAI8	AI
RTD-4 channels	-	AI
TC-8 channels	MAI8	AI

I/O channel attribute

I/O channel attribute including:

- Channel enabling attribute
- Line fault detection attribute
- Channel filter attribute
- Channel sample attribute
- Channel mode selection
- Channel measurement range setup attribute
- RTD module and wire selection attribute

Attribute matching list:

Channel type	enable	Line fault detection	filter	Sample cycle	Mode selection	measurement range	Wire selection
AI	Support	Support	Support			Support	2-wire or 4-wire
AO	Support	Support				Support	
DI			Support				
DO							
PI	Support			Support	Count/ frequency	Support	

RTD	Support	Support	Support			Support	Support
TC	Support	Support	Support			Support	

Filter range:

Channel type	Filter range
AI channel	0~20 data sample points
DI channel	0~255 data sample points

Sample cycle range:

Channel type	Sample cycle range
Pulse input channel	10~2550 ms

Measurement range:

	AI voltage	AI current-8channels	AI current-16channels	PI -8channels	RTD	TC
SCALE_0_5VDC	Support					
SCALE_1_5VDC	Support					
SCALE_0_10VDC	Support					
SCALE_0_10mA		Support	Support (4-wire)			
SCALE_4_20mA		Support	Support			
SCALE_0_20mA		Support	Support (4-wire)			
SCALE_100mV						Support
SCALE_+/-5VDC	Support (only 16 channels)					
SCALE_+/-10VDC	Support (only 16 channels)					
SCALE_+/-10mA						
SCALE_+/-20mA			Support (4-wire)			
SCALE_0_500 Ohm					Support	

SCALE_0_4000 Ohm					Support	
SCALE_200Hz				Support		
SCALE_500Hz				Support		
SCALE_1KHz				Support		
SCALE_2KHz				Support		
SCALE_5KHz				Support		
SCALE_10KHz				Support		
SCALE_25KHz				Support		
SCALE_50KHz				Support		
SCALE_CU50					Support	
SCALE_CU100					Support	
SCALE_PT100					Support	
SCALE_PT1000					Support	
SCALE_B_TC						Support
SCALE_E_TC						Support
SCALE_J_TC						Support
SCALE_K_TC						Support
SCALE_N_TC						Support
SCALE_R_TC						Support
SCALE_S_TC						Support
SCALE_T_TC						Support
SCALE_DIO						
SCALE_COUNTER						
SCALE_PLUSE						
SCALE_PWM						

During the setup of I/O channels for HART AI modules, users can turn on/off the HART AI function. Turning on the HART AI function, HART instruments can obtain input signals; turning off the HART AI function, HART instruments can only be used as normal current type AI channels.

2.3.1.4. Template

Template function

For some configuration objects, NCS4000 system provides template function, and these objects are reusable. These objects include: user-defined function block objects for FBD modules, user-defined function block objects for LD modules and module objects.

There are some differences between these 3 types of template objects:

In template library, when the user-defined function block template objects (FBD and LD) are used by certain modules, they will be locked and cannot be deleted; while module template objects don’t have the limit.

In template library, the user-defined function block template objects cannot be edited and modified, while module template objects can.

Note: In despite of a module template object is modified; the module created previously based on the module template object doesn’t change.

Can it be understood that user-defined function block template adopts “link” mode while module template adopts “copy” mode.

Note: For user-defined function block object created from template library, its type is “reference” by default and its arithmetic inside cannot be modified. Users can convert the type of user-defined function block object to “instance” and then edit it.

2.3.2. Control strategy download

In the NCS4000 system, control strategy becomes effective after it is downloaded to physical controller. Completing control strategy exploitation, virtual controller configuration, I/O module and channel configuration and mapping virtual controller to physical controller, users can download the whole control strategy.

For NCS4000 controller, users can download the whole controller or just an arithmetic module; users can either clear up the old download information and download new information, or reserve the old download information and just update the changed part.

Download controllers after clear up

Clear up all the information which has been already downloaded in the controller, and then download the new information.

Download controllers without disturbance

Reserve the runtime information of the modules in the controller. If it isn't sure that the virtual controller is consistent with the physical controller, it is recommended to execute "Download controllers after clear up" before this operation.

Download modules after clear up

Clear up the information which was downloaded to the module in the last time, and then download the new information.

Download modules without disturbance

The values of the function blocks, parameters, I/O mapping relationship and communication relation of the module keep the same with the values of pre-download (or continued operation).

IF users have already downloaded an FBD module, then edit it in the control strategy design environment and download without disturbance, the objects which have already been downloaded keep the same and the objects which is deleted will delete from the physical controller, and new objects added will be downloaded.

If a parameter of an FBD module is modified online, its value will be reserved after download without disturbance; if the default value of the parameter is modified offline, its value will come up to the default value.

Note

Before download, the NCS4000 will check the validity of each module which is going to be downloaded. If error occurs, the NCS4000 stops the download immediately.

In order to ensure consistency and avoid conflict, before download, the NCS4000 will check whether the module that will be downloaded is opened by other procedures, if yes, the system will warn that the module is locked.

If a physical controller, which is downloaded under another environment, is going to be connected to current environment, the information of the physical controller is not consistent with the virtual controller. So that, some unpredictable problems will appear when the controller is directly downloaded without disturbance or edited online. In this case, it is recommended to execute once "Download controllers after clear up", and after that execute download without disturbance.

The information consistency checking function is added into PC software and physical controllers in the new version. If the information of the physical controller is not consistent with the virtual controller, or if it's the first time to download the controller after mapping, only the operations of "Download controllers without disturbance" and "edit offline" are permitted, while other download modes and "edit module online" function are prohibited. In the case that the consistency is unsure, before executing download or "edit online", the system will give the corresponding warnings and users decide whether the operation goes on or not.

3. Access the controller

3.1. I/O server

3.1.1. Introduction

NCS4000 I/O server serves as drivers of NCS4000 hardware device. Each of the software in the I/O server visits NCS4000 controller by the I/O server.

I/O server is one of background program, and I/O server can only run one instance for one computer.

3.1.2. Start and stop

When the client programs link the I/O server, the I/O server could run automatically.

In addition, the user also could start、stop the I/O server in the *NCS4000 diagnostic tools*.

3.1.3. Binding of local IP address

If more than one Network Interface Card is installed in the computer or if more than one IP address are settled aimed at certain Network Interface Card, then the binding of local IP address of the I/O server is needed set.

The way is to modify the item of IP Address in the configuration files of **ios_ip_setting.ini.**

For example：

IP Address=192.168.1.5

If the item is not set, the I/O server will bind the first IP address automatically.

Note：After the NCS4000 software was installed, there is a file named “ios_ip_setting.ini.tmpl”in installed catalogue. The user could change the name to ios_ip_setting.ini and modify it.

Controller filtering

The I/O server can filter the controller according to the IP address, namely, for the filter controller, the I/O server will not provide related visiting service.

Filter setting is also saved in the configuration files of ios_ip_setting.ini. When the default filer setting is initialized completely（**Init All=yes**）, the I/O server will provide visiting server of all the controllers which is out of the IP address in the **[Filter Exclude]** list**.**

Vice versa, when the default filter setting is non-initialized completely（**Init All=no**）, the I/O server will only provide the controller visiting services of the IP address in the [Filter Include] list.

For example, when the filters are installed as follow, the I/O server will provide all controller visiting services except the controller (192.168.3.5).

[Filter Default]
Init All=yes

[Filter Include]

[Filter Exclude]
192.168.3.5=

Controller Access Protocol parameters
Users can collocate some controller access protocol parameters in the configuration file of ios_ip_setting.ini. The detailed information is as below:

Max Session Buffer Size:
Size of the biggest conversation buffer. It is related to the controller version of the agreement, users cannot be amended in general. The value of new controller is 2048 (default), the old is 1460.

3.2. OPC server

3.2.1. Introduction

NCS4000 OPC server accords with the standards of OPC DA 2.0, and can provide client programs with real-time visiting services of the controller data. The ProgID of the server is:

Microcyber.NCS2IOServer

3.2.2. Start and stop

Users can start NCS4000 OPC server in the manner of direct connecting through OPC client program.

The server will automatically establish the address space corresponding to the configuration information. For the larger-scaled configuration system, the initialization may take some time (mainly depending on the system size and performance, not more than 10 seconds in general). In addition, users can also start or stop the NCS4000 OPC server in the *NCS4000* diagnostic tools.

Address Space Structure

NCS4000 OPC server organizes the address space according to
“Regional-modules-function block (parameters) “ in the control strategy. For the unassigned virtual controller module, it is not shown on OPC server. But for the assigned virtual controller module unrelated to the physical controller module, OPC will add it to the address space, but the data is invalid. Currently, NCS4000 objects support Chinese name, and the object names of function block is modified with "[]", which could distinguish with other objects. For example:

Region 2. FBDModule_01. [AI_1]. MODE_BLK.ACTUAL

region2.FBDModule_01.[AI_1].MODE_BLK.ACTUAL

For the FBD module, the parameters which will be displayed in the address space include:

- Self-definition module parameters
- Implementation cycle of Module
- Standard function block parameters
- Standard function block entity ID
- Implementation order of Standard function block
- Self -defined parameters of Self-definition function blocks (including examples and references)

Currently, the interior standard function block parameters of the Self-definition function blocks are not available.

For the unused standard function block in configuration strategy (has been added to the module), there is no implementation order.

For LD modules, the parameters which will be displayed in the address space include:

- Self-definition module parameters
- Self -defined parameters of Self-definition function blocks (including examples and references)

Address space initialization method

For the address space initialization of the NCS4000 OPC, there are two ways: read from the system information database and read from local store.

System information database read: When it starts, the OPC server automatically tries to link system information database and accesses configuration information from it and establishes address space. If the access of configuration information is successful, the OPC server will automatically store configuration information in the local files.

Reading from local storage: If users set the OPC server to read from the local storage address space structure, or if the read from system information database is failed, the OPC server will attempt to read the local storage files. Under normal conditions, when engineering system configuration and debugging are basically completed, and when the project is basically running, the way of reading from local storage is proper.

The initialization method of configuration mode address space is:
Editor the configuration file of ios_ip_setting.ini, which is stored in the installation directory of NCS4000 executable program. Found:

```
......
[OPC Server]
Load from SYSDB = no
......
```

Set the item of Load from SYSDB to “no”. After the configuration is modified,
the OPC server should re-run.
[Note: At present the local storage function are not provided.

3.2.3. OPC server accessing optimization

NCS4000 system provides a series of options for the configuration of the I/O server and OPC server. A reasonable configuration will enable NCS4000 OPC Data Access to achieve the best performance.

OPC server address space configuration
All of the control strategy module parameters of NCS4000 system can shown in the address space structure of the OPC server, but in fact a lot of parameters are rarely used. If these parameters are added to address space, and meanwhile there are lots of function blocks, it will waste the system resources of the computer and affect the performance.

In NCS4000 system, several configuration ways of address space are provided as follows:

- 0 not add any parameters;
- 1 Add I/O typed parameters and reserve the parameters in the list; (Default)
- 2 only add I/O typed parameters;
- 5 Self-defined; (Currently does not support)
- 10 All parameters are added;

In the file of ios_ip_setting.ini, the parameter （**[OPC Server]，AddrSpace Level**）can be set and it defaults to 1 if the parameter is not set.

"Reservations" list is one series of self-definition parameters name, and all of the parameters would be added to the address space. The definition way of retaining list is:

RPN[###]=[NAME] for example：

RPN000=MODE_BLK
RPN001=PV
RPN002=SP
... ...

Controller accessing optimization

NCS4000 system supports controller accessing optimization based on the buffer zone, currently only the OPC server uses this function. OPC server supports the buffer zone accessing optimization of FBD, LD modules. If the users want to activate the function, they can set the Cache Level in the file of ios_ip_setting.ini to:

0 – denotes not to use the functions of the buffer zone, all data are read directly from the controller device;

1 – use the functions of the buffer zone when FBD module parameters are read；

2-use the functions of the buffer zone when LD module parameters are read；

3- use the functions of the buffer zone when all module parameters are read；

Currently，the parameters access for LD must use the function of buffer zone.

Note: This function is still in the testing phase, we don’t recommend you using this function.

3.2.4. FF H1 subsystem

Support of OPC server to FF H1

NCS4000 OPC server provides the subsystem of FF H1 with the access server subsystem services.

In FBD module, the configured FF H1 function block can access the parameters through OPC server. When the OPC server starts it will establish address space related H1 automatically.

In the address space of OPC server, the prefix of FF H1 function block is ffh1, for example:

Area01.ffh1 [AI_FF_1]. OUT.VALUE

The FF H1 function block parameters can be loaded selectively according to FBD modules function block. And the configuration of ios_ip_setting.ini will also have efficiency for the FFH1 function block.

3.3. Configuration online monitor

Online Monitoring

NCS4000 system provides a kind of device data access mode based on graphic, namely, on-line monitoring of control strategy development environment.

In the control strategy development environment, users can view and modify the module data which is downloaded to the physical controller real-timely. When the development environment is in the state of on-line monitoring, the input and output data of function block and the parameters of the ladder diagram are all shown in the editing view. In this way, users can easily check the accuracy of control algorithm.

Before the control strategy development environment monitoring is switched to the on-line mode, it will check the consistency between the current module configuration information and download information. If it is not consistent (such as users modify the configuration strategy after they download it, but still have not re-download it), then the users can not switch to on-line monitoring mode.

Note: This function mainly uses for engineers to debug and diagnose, it is not recommend using as monitor screen in the formal running duration of the system.

3.4. NCS4000 system diagnose tools

NCS4000 system diagnostic tool has direct access function, and can modify the attribute of the device. It is mainly used for configuration engineers. The tools provided these functions:

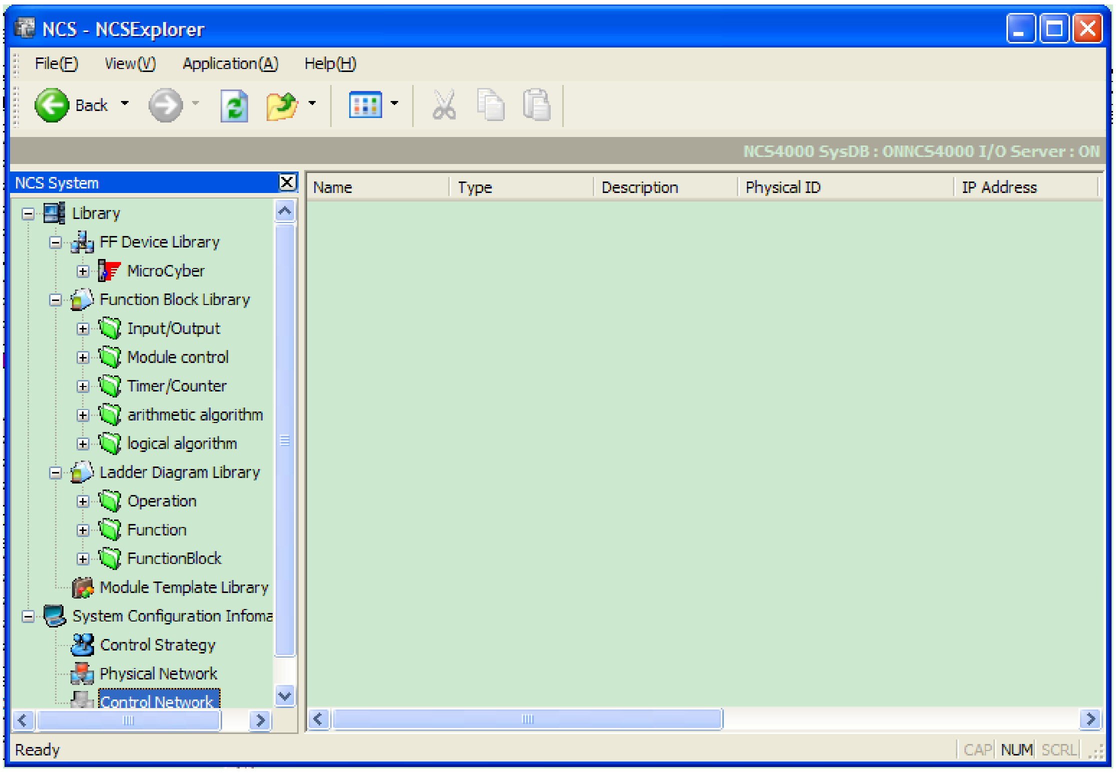
- Monitoring of online controller state
- Start and stop control of I/O server and OPC server
- Modifying of controller name and IP address
- Read, write the parameter, download and debug the controller

4. Using the Software

4.1. NCS4000 Explorer

Introduction

NCS4000 Explorer is a System-level configuration tool in the NCS4000 control system. We can browse and manage control strategy configuration of current application project, configuration object properties, I/O mapping relationship, and downloading the configuration information to physical controller by the NCS4000 Explorer.



Control Strategy Framework Management

Virtual Controller Management

Control Strategy Download

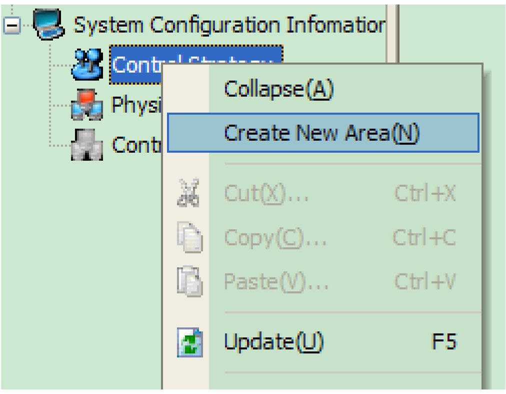
Library Manage

Startup Correlation Tools

4.1.1. Control Strategy Framework Management

Area Management

User can create an area object in Control Strategy. Click the right key of mouse on the item “Control Strategy”, then select “create new area”.

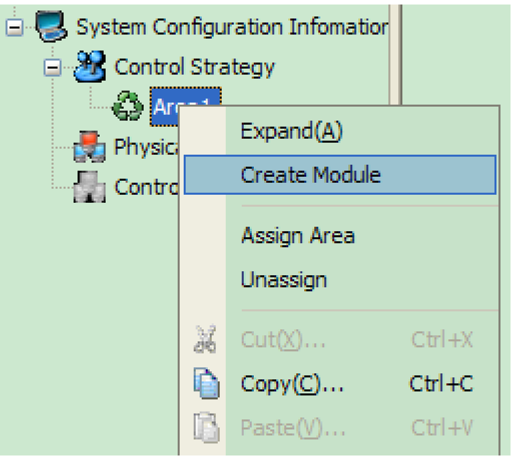


The new area object is directly added to Control Strategy and its name is in edited status. The default name of new area object is “area n” (n begins with 1).At present, there is no limit of the total number of area.

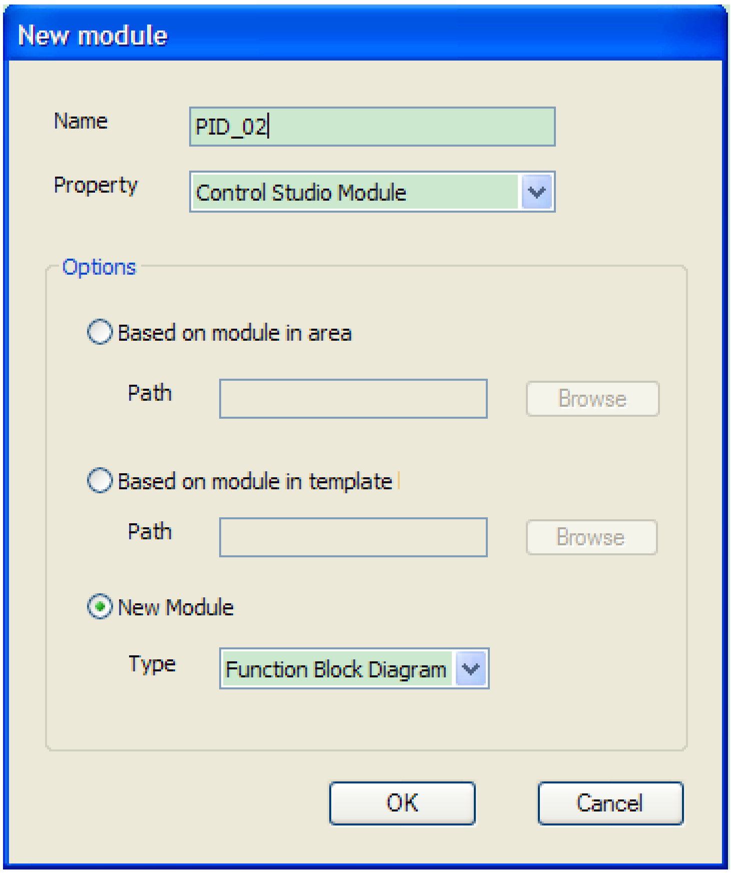
User can rename and delete the area object. We must pay attention to the deletion of area object, because the module of the area object will be deleted simultaneously. When other operation is working with the area object, the delete operation will fail. For example, the deletion of the area will be prompted that object being locked when control strategy development environment is editing a module object.

Module Management

Users can create a new algorithm module object for area object in NCS4000 Explorer. Click the right key of mouse on the item “Area” object, will pop-up “Create Module”.



In the dialog box, you can input module name firstly, but special characters don’t allow adding to the module name.



New module

Name: PID_02

Property: Control Studio Module

Options

○ Based on module in area

Path:

○ Based on module in template

Path:

● New Module

Type: Function Block Diagram

There are 3 mode of the module creation: base on the existed module, base on the module library and create directly. User needs to select the module manually if he selects "base on the existed module" or "base on the module library". The directly created module is blank, user needs to point out the sort of arithmetic: Function Block or Ladder Diagram. Select "OK", the creation of module will be end.

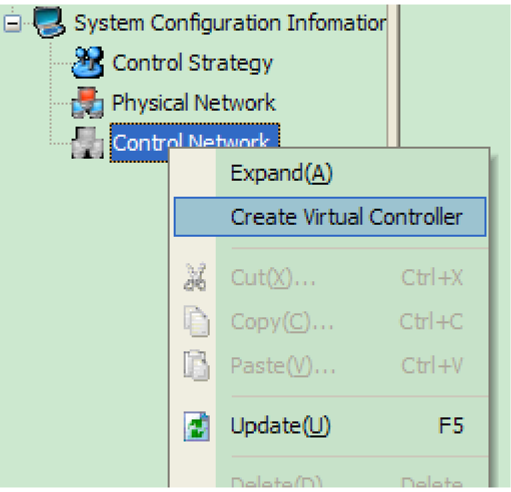
In addition, there is a convenient method of module creation, namely dragging. User can drag the module in module library to the Area; also can drag in the area or between areas. The sequence number sets the obtained modules from 1.The editing module can not be copy, delete and rename.

4.1.2. Virtual Controller Management

Virtual Controller Management

User can add Virtual Controller to current configuration project under "Control Network". Virtual Controller can be created less than 32.

Click the right key of mouse on the item "Virtual Controller Object", and can copy, delete and rename by the item.



In NCS4000 Explorer, user can download Virtual Controller, controller mapping and I/O module mapping configuration, etc.

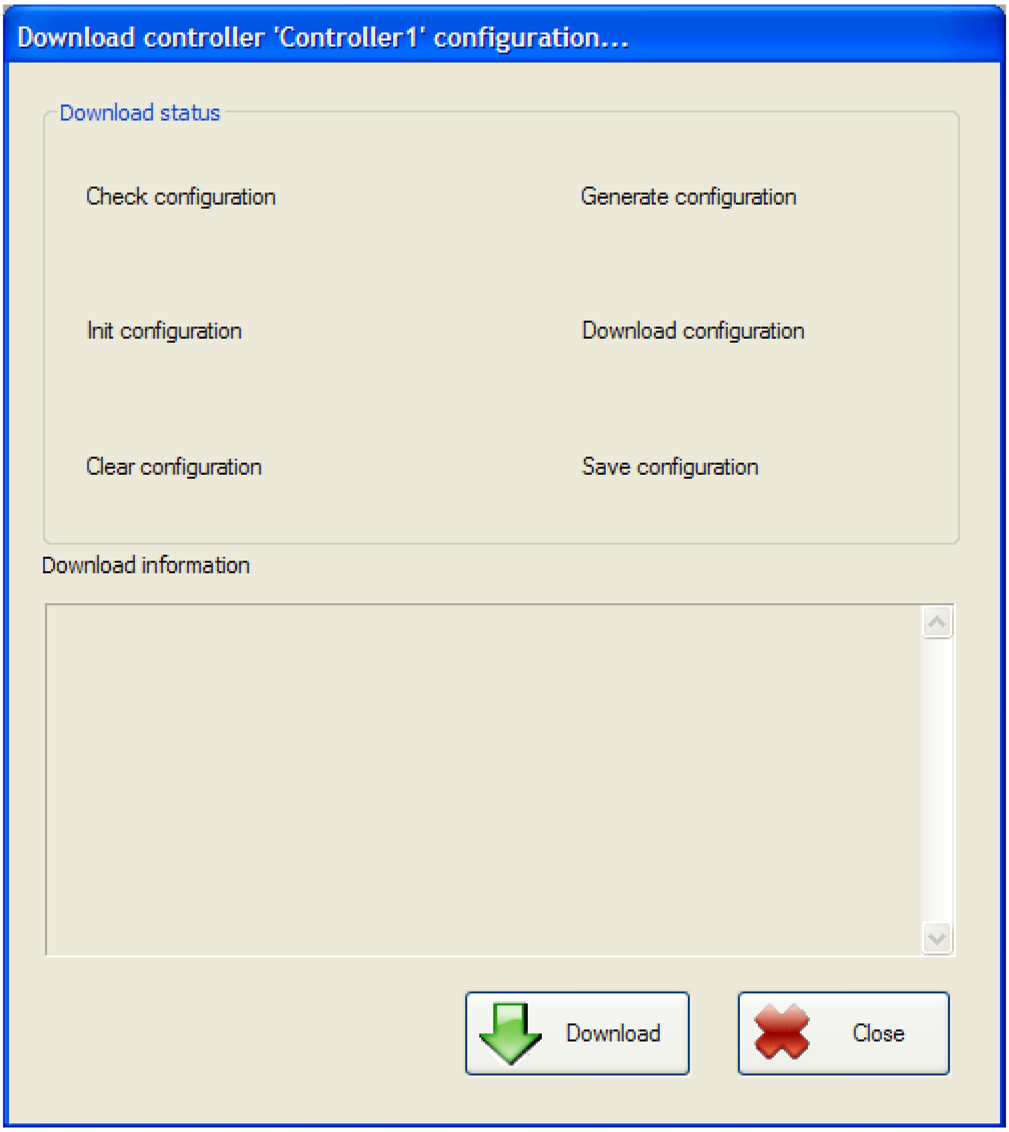
Controller Download

Controller download consist of two modes, one is downloading the entire controller, and the other is downloading the control strategy module. All of them can be selected in the right key item of Virtual Controller.

When downloading single controller, user can click the right key item of controller object, manually select the downloading module in the pop-up dialog of module object browse. User also can find out the downloading module directly from control strategy module list, then click the right key and select download menu.

After confirming the downloaded object, the download dialog will be popped up. Then click the “download” button to begin downloading. The downloaded information will be display in the download information column.

At present, user can not force to interrupt the download process because the download and close button will be invalided during downloading period. After the end of downloading, these buttons will be valid.

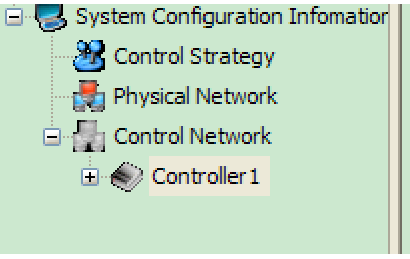


Controller Mapping

Controller mapping refers to the establishment of virtual controller and the physical relationship between the controllers.

In the item of "system configuration information – physical network", there are currently on-line controller physical controller included name, IP address, work status, etc.

Configuration engineer can drag physical controller to virtual controller directly, then can be obtained the mapping between the two objects. We also can execute the operation manually. Attention should be paid to that the mapping is one by one.



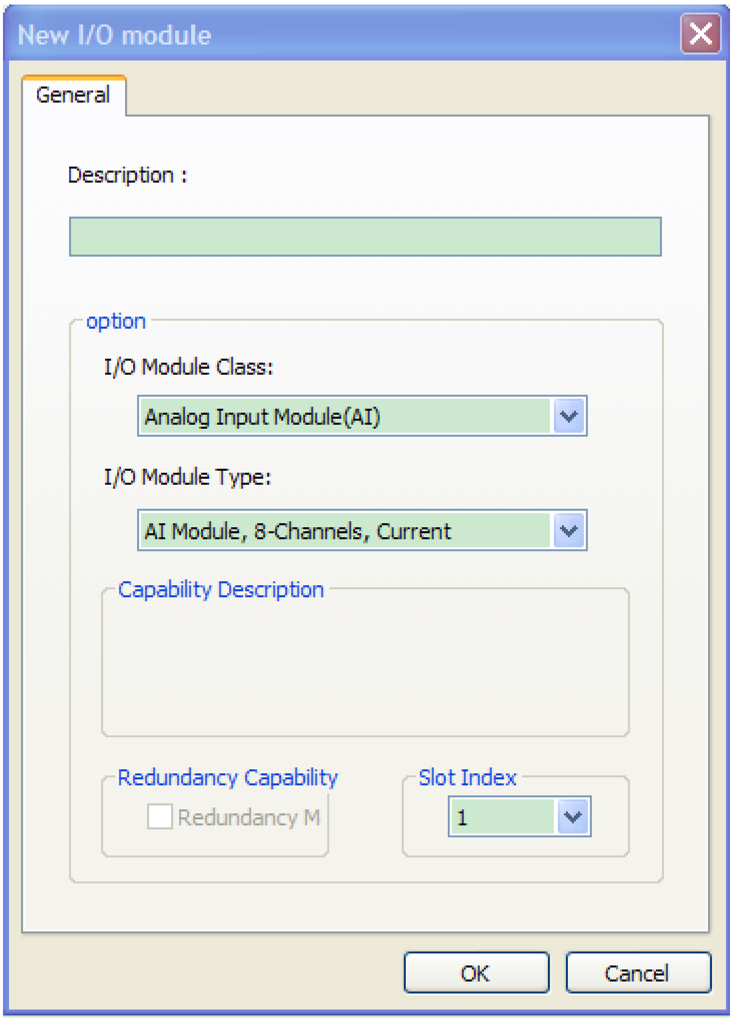
The mapped virtual controller will be darker. At the same time, in the information column of virtual control object we will obtain identifier of physical controller, work status and the information of corresponding physical controller; we will obtain the information of the mapping virtual controller in the information column of physical controller.

Before changing the mapping relationship of “virtual - physical”, user must cancel their mapping relationship in the item “Controller Unmapping”. The mapping relationship will be automatically cancelled once the virtual controller is deleted.

Controller I/O Module Configuration

In the off-line configuration process, the virtual I/O communications modules can be added to controller object. In the right mouse button menu of the virtual I/O modules list of virtual controller object, select "New I/O modules", can pop the new I/O module dialog. In the dialog box choose I/O module types, models, such as slot Index. Currently NCS4000 support seven kinds of I/O communications modules: AI, AO, DI, DO, TC, RTD, as well as Field bus modules. Each virtual controller can be up 64 new I/O communications modules.

It must be noted that for the added I/O communications modules, users can not modify its type or model, which can only amend the slots Index and the I/O module project description information. In view of I/O communications modules attributes, slots Index drop-down list only the unoccupied Index.



Description :

I/O Module Class: Analog Input Module(AI)

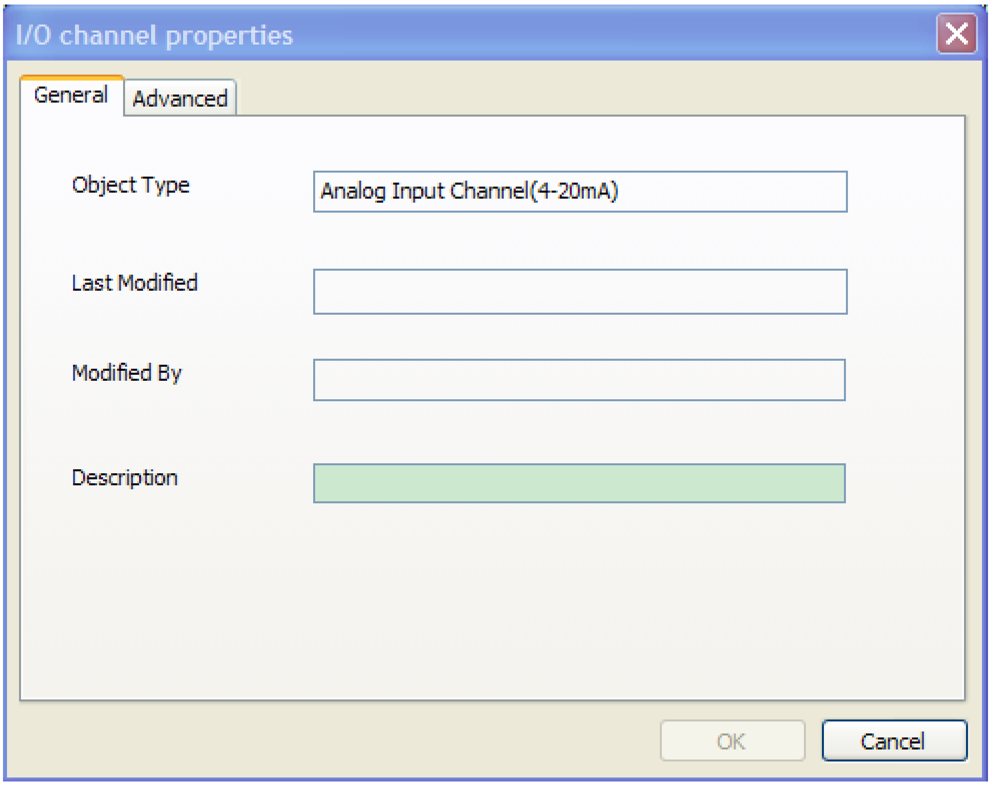
I/O Module Type: AI Module, 8-Channels, Current

☐ Redundancy M

Slot Index: 1

Controller I/O Channel Attribute Setting

In the off-line configuration process, you can set the I/O channel attribute information. These messages include: channel enabling information, line fault detection information, channel filtering information, range setting information and line choice information. In the right mouse button menu of the I/O channel, choose "Properties" or double-click the I/O channel objects can popped up I/O channel attributes dialog box.



Object Type: Analog Input Channel(4-20mA)

Last Modified:

Modified By:

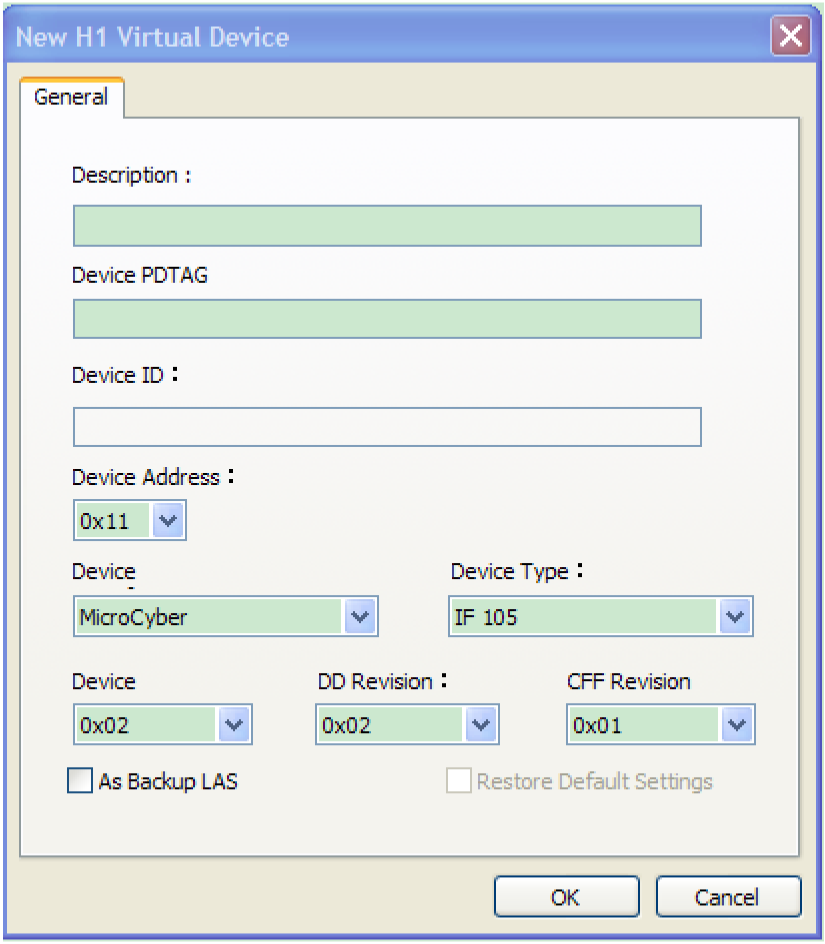
Description:

After the completion of user configuration, user can download the I/O channel configuration to the module through the " I/O modules download functions", the method is Right-click I/O modules nodes, in the pop-up menu choose the right "download" order, pop-up download dialog box, click " Download "button, explorers download the information of all channels to the controller, users can also use the" I/O modules List "Right menu" “download "order to download the configuration information of I/O channel to the controller.

H1 Device Configuration

New H1 device: Add a new target virtual H1 device

First we can create a H1 target through "new I/O modules" modules. I/O module types can be chosen "Field bus module (BUS)" or "H1 communication module, 2 - port" (NCS4000 current system only supports two ports H1 module) during creating the new I/O module. Then we would create a new H1 module object. There are two H1 network segment in the H1 module, right-click the H1 network segment, in the pop-up menu choose the right "new H1 device" orders, pop-up dialog new H1 device in the dialog box to choose device vendors, device type, device versions, as well as the information of versions of DD and CFF.



New H1 Virtual Device

General

Description :

Device PDTAG

Device ID：

Device Address：0x11

Device: MicroCyber

Device Type：IF 105

Device: 0x02

DD Revision：0x02

CFF Revision: 0x01

□ As Backup LAS

□ Restore Default Settings

Note (restrictions):

1. PDTAG device in the network segment and the address must be the only，address range (0x11 ~ 0x2e).

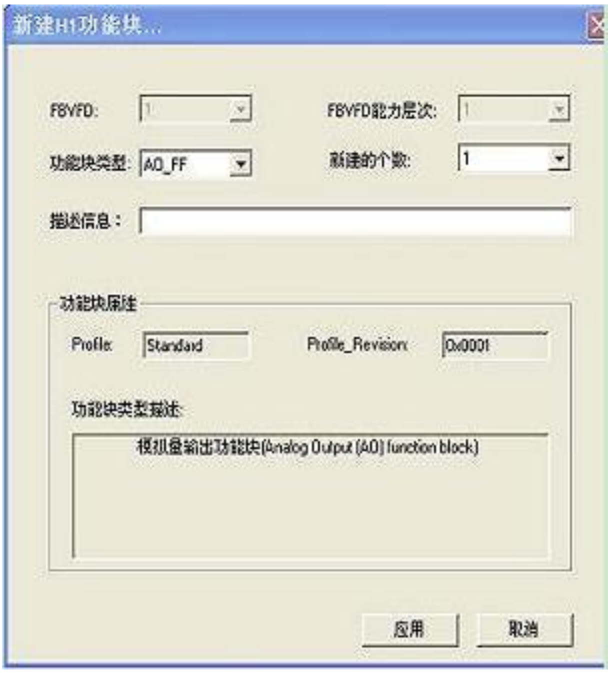
2. A H1 network segment supports up to 16 H1 device.

3. For virtual H1 device, we can change its address and PDTAG, but not modify the device vendors, device type, device versions, as well as DD and CFF versions in the virtual H1 device.

New H1 function block: user can create new H1 function block by the initialize function of H1 device.

NCS4000 explorer will confirm the ability of initialize function block by reading the information described in the CFF document.

User can right-click virtual H1 device nodes, in the pop-up menu choose the right "new H1 function block" orders, new H1 function block pop-up dialog box, users need to choose the dynamic creation of the type and number of function blocks. NCS4000 explorer can name automatically for each new function block by the naming rules "device addresses _ function block type and the order".



FBVFD: 1

FBVFD能力层次: 1

功能块类型: AO_FF

新建的个数: 1

描述信息：

功能块属性

Profile: Standard

Profile_Revision: 0x0001

功能块类型描述:

模拟量输出功能块(Analog Output (AO) function block)

Note: the name of the function block in H1 segment should be unique.

H1 Device Mapping: the physical H1 device will be mapped to the virtual H1 device.

The all online physical device note of current network segment will be listed in H1 network segment, the attributes of physical device including device physical marking, device addresses, device category, device manufacturers ID, device type ID, device, revised version, DD, and the amended version working conditions and so on. Users can directly drat the physical H1 device to the virtual one.

Note:

1. Operation can be carried out mapping the physical device and virtual H1 H1 device must have the same device vendors ID, device type ID, device, as well as revised versions of DD version of the amendment.
2. Virtual H1 device and physical one must map one to one.
3. Because a large number of communications, in order to increase the success rate of mapping, the proposed network, users will only need to access this mapping the physical H1 device, in the successful mapping, access to a need for physical mapping device H1.

H1 Device Download: H1 device will be downloaded to the device configuration information in the device.

H1 device download is different from the configuration of device download, the following functions:

1. Setting the address device of H1.
2. Setting the PDTAG device of H1.
3. Dynamic examples the H1 function block.
4. Delete dynamic examples of H1 function block.

5. Empty H1 device configuration information (optional).

User right-click virtual H1 device nodes, in the pop-up menu choose the right "download" order (If you choose "Download (clearance device)" order will be emptied H1 device configuration information), in the pop-up download dialog box, click on "download" button.

Note:

1. Download H1 device orders were not activated only in the physical and virtual H1 device for the mapping device.
2. When users choose emptied device configuration information, the network will lead paragraph of the circuit configuration that can not be used.

4.1.3. Control Strategy Download

This part can refer to **Virtual Controller Management**.

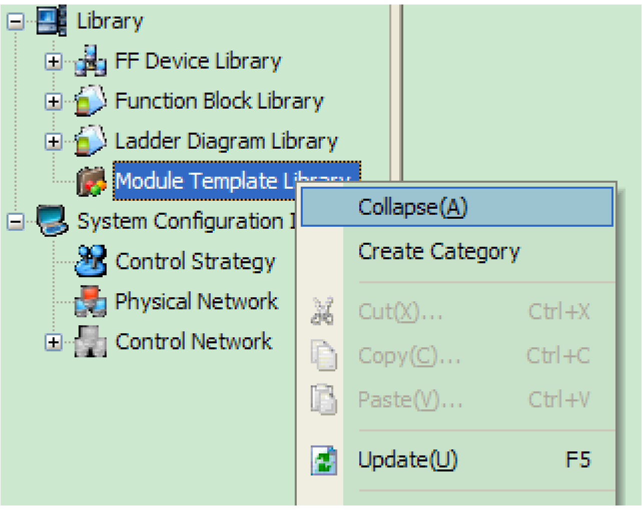
4.1.4. Library Manage

Explorer can visit all of library object, and manage part of them.

In the explorer can visit the library including: (FBD) standard function block library, ladder diagram library (operator, function, function block), as well as module template library. One of the standard function block library and ladder diagram library are read-only, their modification must be done in the development environment of control strategy; Template Library can be amended.

Module Template Library Management

The module template library store module objects of predefined and user-defined, these modules can be dragged directly to control strategy. Module Objects can be classified in the storage, users can define their own categories. Right click on the mouse is modular template library, select "New types of directory" menu item, and then name for the new category of directory.



Users can drag the edited module in the development of control strategy, so the module can be such a reuse. Users can direct the editorial changes in the module object. Attention should be paid to reusable modules is "copy paste" mode instead of "link" model, the original module with the new module no link between the party and even delete the changes will not affect the other party.

Note: Since the definition of all the objects is only effective within the framework of the project, in other databases can not be shared.

FF Device Library Import

NCS4000 system support agreement with FF H1 standardized device, the device stored in the FF device. Users should import FF NCS4000 device information into the system information in the database before the configuration of H1 device:

1. Right-click the mouse "FF device library", then select "import FF device" order in the pop-up menu
2. In the " import FF device " dialog box, click on "Import device" button
3. In the "browser folder" option in the dialog box import into the needs of the CFF directory of FF device documents
4. Click on the "OK" button, explorer begin to import into the directory information of device description
5. After the completion of importing, by watching "importing detailed information" box combination to access the information into detailed information

The CFF Document Directory Requirements Are As Follows:

1. Path formal requirements.. \ vendor ID (six characters 16 mechanism) \ device version ID (four characters 16 mechanism) \ CFF file name (16 characters mechanism 6)
2. A CFF corresponding documents, in the meantime, some suffixes documents called .sym and .ffo, the names of these two documents counterpart CFF file name of the former four characters, and in the same directory.
3. CFF file format need accord with FF 1.5 or 1.7 format documents requested.

Note: In accordance with the following steps user can amend device manufacturers, the types of device and function block Name:

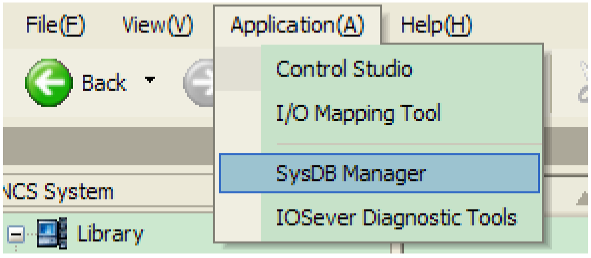
1. In the installation directory to find DD Contents
2. Open the Support.ini in DD directory
3. Find
   [Manufacturers By Id]
   ID = name
   To amend the relevant manufacturers name or add names of new vendors to participate in the paragraph
4. Find
   [Vendor ID Devices By Code]
   ID = name
   Amend the name or type of device will be new types of device added to the corresponding paragraph

5. Find
   [Block Type Mnemonics]
   Device manufacturers ID type ID device amended version of the revised version of the DD function block DDItem = name

To amend the relevant function block to add a new name or names of function blocks (including device amended version of CFF file name is the first and the second character, DD revised version of the file name is CFF third and fourth characters, DDItem can be found documents in CFF)

4.1.5. Load Related Tools

In NCS4000 explorer can activate other NCS4000 tools.



Applications menu item:
Control Strategy Development Environment:
Start a control strategy development environment program, without opening any module.

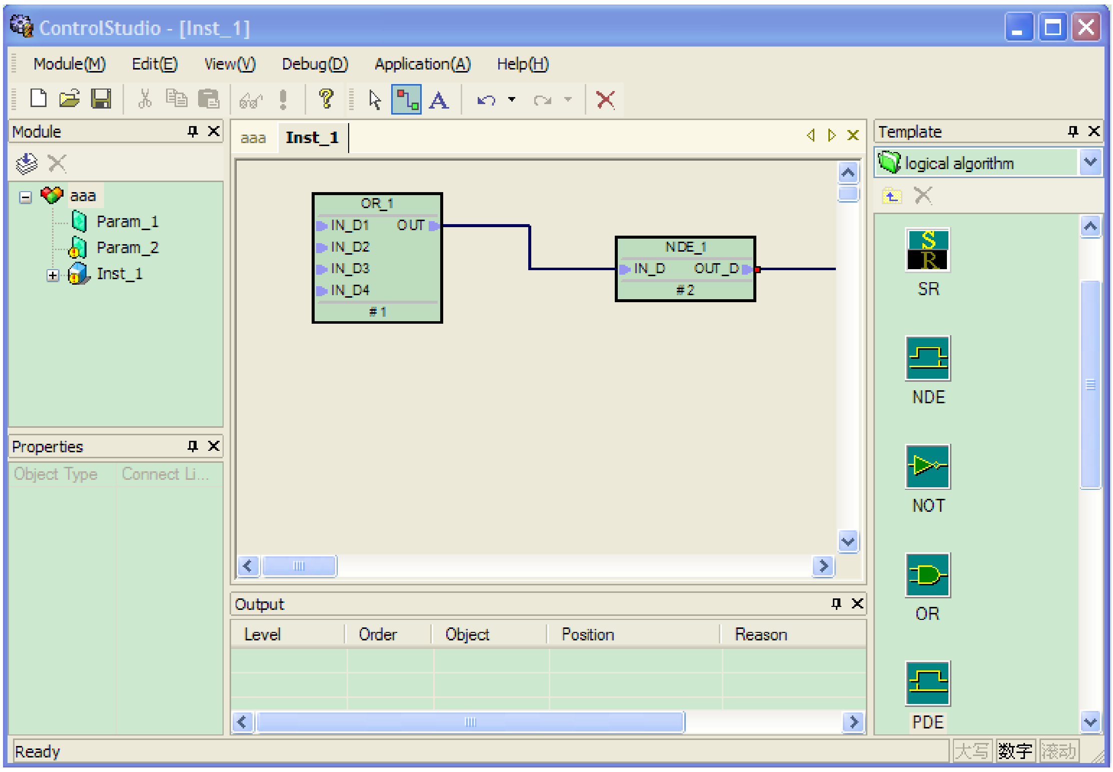
I/O mapping tools:
Start I/O mapping tools that all controllers and I/O modules, channel will be displayed.

Information System database management tools:
Start-up System database management tools.

4.2. Control Strategy Development Environment

4.2.1. Introduction

NCS4000 control strategy development environment is a NCS4000 control system module (templates or instance) object configuration tool.



Control strategy development environment mainly includes the following components:

Module Management

Module management targets for the completion of the various tasks, which include the modules object’s creation open, preservation, save, close and attribute editing.

Data Management

Data management targets for completion of managing various data object under module objects. Including the parameters object, the examples of user-defined object function block, standard function block reference object, the user-defined function block reference object’s add, delete, edit, and other functional attributes.

Template Management

Targeted for completion of various management tasks related with function block template. It includes adding, deleting, and attributes editing the custom directory objects and the user-defined function block template object, and automatic updating the template function blocks.

Arithmetic Editing

Targeted for completion of module (or user-defined function block instance object) control algorithm editorial work. It includes the operations of opening, closing and attributes editing Edit view, and adding, deleting, moving, attribute editing graphic objects, and withdrawing and redoing the functions graphics editing operations.

Arithmetic Check-Up

Targeted for completion of module (or user-defined function block instance object) control algorithm inspection work. This part includes editor syntax check, editing error positioning, and other functions.

Online Monitoring

It is targeted for the online monitoring and controlling of various data . This part includes the start-up, stop of the on-line monitoring, the circularly reading data and real-time data displaying, online revising parameters, and other functions.

Simulation Debug

It is targeted for the completion of Ladder Diagram (LD) module offline simulation debugging tasks. Including the start-up, stop of simulation debugging, the recycling read data and real-time data display, modify simulation data, and other functions.

Communication between Modules

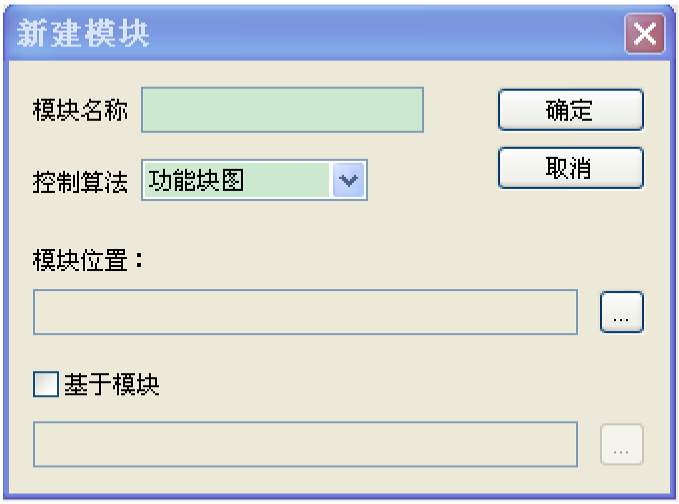
It is targeted for completion of communication module configuration. Including the open, refresh, preserve communication configuration information, and add, remove, attribute edit the communications objects, and other functions.

4.2.2. Module Management

Module management targets for the completion of the management of modules, including the following types of functions:

The New Module Object: creating a new module object.

Use the menu or toolbar and input "modules/new", pop-up "new module" dialog box. In this dialog box, enter the name of module; select module position; if based on the other modules, either to select a module object based on, or to choose a control algorithm.



控制算法 功能块图

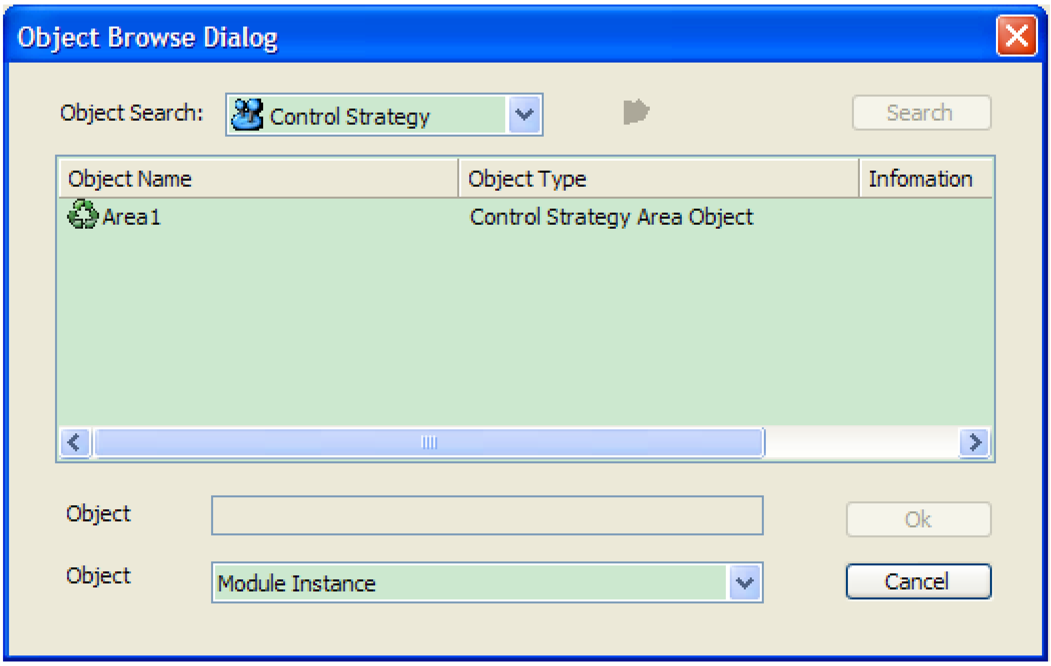
Object Name: It can not be empty; not including “/\:*?"<>|.” and other special symbols; If it is a module instance object, the whole object is under the sole control strategy, otherwise the object under his father only.

Module Location: It is used to decide the types (templates or examples) of module object.
Unable
to be empty and use "object browser" dialog box to choose.

Control Algorithm: it includes function block diagram (FBD) and Ladder Diagram (LD)
based on
the modules: using "Object View" dialog box to choose. (Optional)

After confirm that the input and choice are effective, the original "active" object modules will be closed, then create and open new module object. For the "module-based" object modules, their "content" is the same with the module object based on.

Open Module Object: Open an exited module.
Use the menu or toolbar and input "modules/Open" orders, pop-up "targets of View" dialog box. In this dialog box, select a module object.



Object Search: Control Strategy

Object Name	Object Type	Infomation
Area1	Control Strategy Area Object	

Object

Object: Module Instance

Confirm the selection, then close the existing "active" module object and open the chosen module: re-initializing modules, templates, attributes, output, etc. And automatically open the edit view of module object's control algorithm.

In addition, if the module object ID acts as a start-up parameter, then user can start the NCS4000 control strategy development environment with after the module object being opened directly.

Save Module Object: save the current configuration information.
Use the menu or toolbar and input "modules / Save" command.
The order only is valid when configuration information of the current module (or under its data objects) changes.

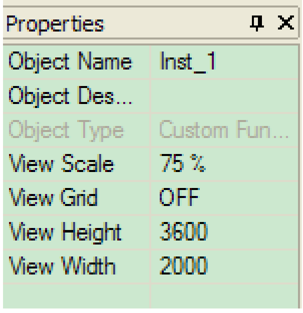
Save Module Object As: Save the current module as a new module object.
Use the menu, enter "modules/Save as" command, pop-up "targets Save" dialog box. In this dialog box, select storage location of the new module object, and enter the name of the new module object.

Storage locations: deciding the type of module object (templates or examples).
Object Name: It can not be empty; Not including "/ \:*?"<>|." and other special symbols; If it is a module instance object, the whole object is under the sole control strategy, otherwise the object under his father only.

After confirm that the input and choice are effective, the current object modules will be closed, then create and open a new module object. The new module object's "content" is the same with the module object based on.

Close Module Object: close the current module object.
Use the menu; enter "module on/off" order.
For the current module (or its Data Objects), if there is a change in configuration information, the system will suggest to preserve the module object before close it. After the module object is closed, all the edit views close, and the contents in other views are cleared up.

Edit Module Object Properties: amend the current module object attributes.
In the module view, select the module object node; in the properties view its associated attributes is shown, including the names of objects, object descriptions, object type, cycle, and control algorithm.



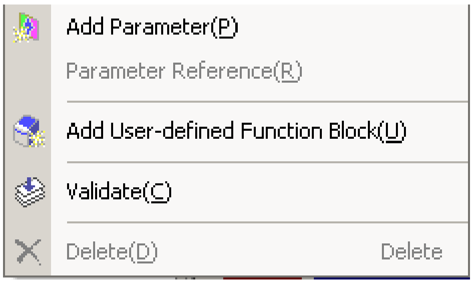
Object Name	Inst_1
Object Des...	
Object Type	Custom Fun...
View Scale	75 %
View Grid	OFF
View Height	3600
View Width	2000

Amend Object Description attribute.
Modify cycle attributes. Effective range: 100 milliseconds to 60 seconds.
Amend H1 network segment attributes. The controller has been assigned to the FBD module object, allowing the establishment of its associated H1 network segment, used in the H1 function block configuration. If the designated H1 net is effective, then all the network devices and H1 function blocks in the H1 are shown in the H1 network segment view. The relationship between H1 network segment and FBD modules is one-to-many, that is, under the same network segment one H1 function block can be configured in a number of FBD module, but a FBD module can only be configured a H1 function block under the same network.

4.2.3. Data Management

Data Management module target for the completion of various management tasks for data object, including the following types of functions:

Add A Custom Function Block instance Objects: Add a new custom function block instance object.



In the module view, right click on the target module object node; in the pop-up menu, select "Add-defined function block" order. After the success of the creation, in the module view, a new user-defined the function block instance sub-node is added under the module object node; in the properties view display its associated attributes; at the same time open the edit view of the user-defined function block instance object control arithmetic. The type of the user-defined function block instance object control arithmetic is the same with the module object it is belonged to.

In the template view, select a user-defined function block template object; then use the "drag and drop" method, put it to the module object nodes in the module view. "Drag and drop" successful, a new self-defined function block instance object node is added under the module object node; In the properties view display its associated attributes; at the same time open the edit view of the user-defined function block instance object control arithmetic.

Edit User-Defined Function Block instance Object Properties: amend the designated User-Defined Function Block instance Object attributes.

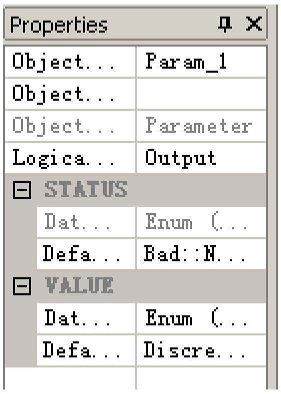
In the module View, user can select a user-defined function block instance object node; in the properties view display its associated attributes, including the names of objects, object descriptions, and other types of objects.

Amend object name attribute. The name of user-defined function block instance object can not be empty; Up to 32 characters; In alphabetical beginning, and only in English letters, numbers and underscores; The only object of the father. After the success of revision, automatically update the object in the module, and display in the edit view.
Amend Object Description attribute. It is up to 100 characters.

Added Parameter Object: add a new parameter object.
In the module view, right click on the object module (or user-defined function block

instance object) node; in the pop-up menu, select "Add parameters" order. After the success of the creation, a new parameter object node is added under the designated object node in the module view; in the properties view display its associated attributes. If FBD user-defined function block instance object participates in graphic configuration, then the function of adding parameter object will be prohibited.

Edit Parameters Object Properties: amend specified parameters object properties.
In the module view, select a parameter object node; in the properties view display its associated attributes, including the names of objects, object descriptions, object types, logical I/O, data type, default values, physical I/O, range type, the limits of such limits .



Object...	Param_1
Object...	
Object...	Parameter
Logica...	Output
⊟ STATUS	
Dat...	Enum (...
Defa...	Bad::N...
⊟ VALUE	
Dat...	Enum (...
Defa...	Discre...

Amend object name attribute. The function is only effective for the parameter object under the module object (or user-defined function block instance object). The name of the Parameter object can not be empty; up to 32 characters; In alphabetical beginning, and only in English letters, numbers and underscores; The only object of the father. The name of the parameter object of user-defined LD function block instance objects can not be replaced by "EN" and "ENO"; if user-defined function block instance object which contains input (or output) parameter objects participates in graphic configuration, then the function of modifying the name of parameter object will be prohibited. After the success of revision, automatically update the object in the module and edit views.

Amend object description attribute. The function is only effective for the parameters object under the module object (or user-defined function block instance object). It is up to 100 characters.

Amend FBD parameter object logic I/O attributes. The functional is only effective for the parameter object under FBD module object (or user-defined function block instance object). If FBD parameter object (or user-defined function block instance object which contains the parameter object) participates in the graphics configuration, the function of modifying logic I/O attributes will be prohibited. After the success of revision, automatically update the object in the module view.

Amend LD parameter object logic I/O attributes. The functional is only effective for the parameter object under modules LD object (or user-defined function block instance object). If user-defined function block instance object that contains the LD parameter object participates in the graphics configuration, the function of modifying logic I/O attributes will be prohibited. After the success of revision, automatically update the object in the module and edit view.

Amend FBD parameter object data type attributes. The functional is only effective for the parameter object under FBD module object (or user-defined function block instance object).

Amend LD parameters object data type attributes. The functional is only effective for the parameter object under LD module object (or user-defined function block instance object). If user-defined function block instance object that contains the LD parameter object participates in the graphics configuration, the function of modifying logic I/O attributes will be prohibited.   After the success of revision, for the type of boolean and float parameter objects under the LD module object, increase physical I/O attributes in the display attribute view.

Amend LD targeting physical parameters of I/O attributes. The functional is only effective for the type of boolean and float parameter objects under the LD module object.

Amend default attributes. The change of parameter object and effective scope is based on its data type.

Edit Standards (or H1) Function Block Reference Object Properties: amend specified standards (or H1) function block reference object properties.

In the module view, select a standard (or H1) function block reference object nodes; in the properties view display its associated attributes, including the names of objects, object descriptions, object types, templates name; double-click on the target node, in the properties view, that the standards (or H1) function block objects invoked under all parameters associated object attributes is shown.

Amend object name attribute. The function is only effective for standard (or H1) function block reference object under the module object (or user-defined function block instance object). The name of object standard function block can not be empty; Up to 32 characters; In alphabetical beginning, and only in English letters, numbers and underscores; The only object of the father. After the success of revision, automatically update the object in the module, the relevant editorial view show.

Amend parameter default attributes. The change of parameter object and effective scope is based on its data type.

Edit User-Defined Function Block Reference Object Properties: amend specified user-defined function block reference object properties.

In the module view, select a custom function block reference object node; in the properties view display its associated attributes, including the names of objects, object descriptions, object type, such as the name template.

Amend object name attribute. The name of user-defined function block reference object can not be empty; Up to 32 characters; In alphabetical beginning, and only in English letters, numbers and underscores; The only object of the father. After the success of revision, automatically update the relevant show of the object in the module and edit view. Object Description attribute amended. It is up to 100 characters.

Delete Data Object: to delete the specified data object.
In the module view, right click a data object (parameter object, user-defined function block instance object, user-defined function block reference object, the standard function block reference object, H1 function block reference object) node; In the pop-up menu, select "Delete" command.

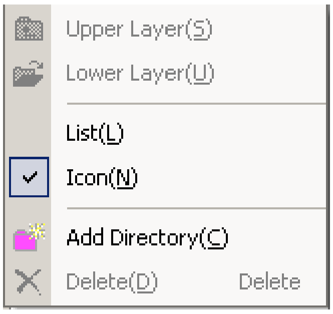
Generally, the order is only effective for the data objects which don't participate in graphical configuration. The object parameters under special and custom function block can only be deleted with his father; If user-defined function block instance object participates in graphic configuration, the input (output) target parameters can not be erased. Delete Data Objects, in the module view, the corresponding data object nodes were deleted; if it is user-defined function block instance (or reference) object, the editorial control algorithm view will also shut down at the same time.

4.2.4. Template Management

Template management for the completion of the various management tasks for related function block template objects, including the following types of functions:

Add A Custom Directory Object Classification: Add a new user-defined classification directory objects.

In the template view, if the current choice is the "function block library," then in the right menu, select "Add Directory" order.



In the object list of the template view, automatically add a new custom catalog objects; in the properties view to show its associated attributes.

Edit User-Defined Classification Directory Object Properties: amend specified user-defined classification directory objects properties.

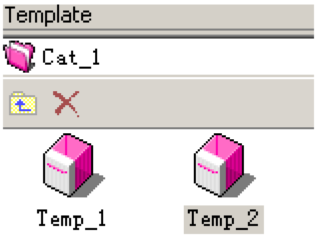
In the template view, select a custom classification directory object; in the properties view display its associated attributes, including the names of objects, object type, and so on.

Amend object name attribute. The name of definition of custom classification directory object can not be empty; up to 32 characters; In alphabetical beginning, and only in English letters, numbers and underscores; The only object of the father. After the success of revision, automatically update the related shows of the object in the template view.
Object Description attribute amended. It is up to 100 characters.

Delete User-Defined Classification Directory Object: to delete specified user-defined classification directory object.
In the template view, right click a user-defined classification directory object; in the pop-up menu, select "Delete" command.

The function is only effective when there is no-defined function block template object under the designated directory object-defined classification.

Add a custom function block template object: Add a new user-defined function block template object.



In the module view, select a custom function block instance object; then use the "drag and drop" method to put the custom function block template to the object list under a user-defined classification directory object in the template view.

Before "Drag and drop", it should confirm that the choice of user-defined function block instance control algorithms have been passed through effective inspection. "Drag and drop" success, in the object list of the template view, automatically add a new user-defined function block template objects; In view attributes to show its associated attributes.

Edit User-Defined Function Block Template Object Properties: amend specified user-defined function block template object properties.

In the template view, select a user-defined function block template object; in the properties view to show its associated attributes, including the names of objects, object type, and so on.

Amend object name attribute. The name of user-defined function block template object can not be empty; up to 32 characters; in alphabetical beginning, and only in English letters, numbers and underscores; the only object of the father. After the success of revision, automatically update the relevant shows of the object in the template and edit view.
Object Description attribute amended. It is up to 100 characters.

Delete User-Defined Function Block Template Object: to delete specified user-defined function block template objects.

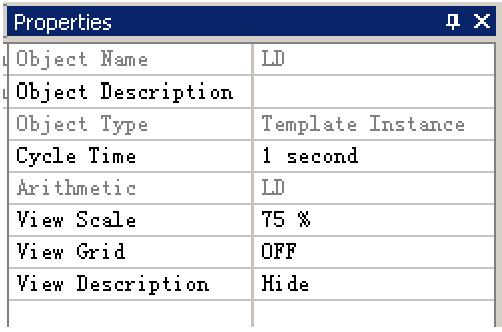
In the template view, right click a user-defined function block template object; in the pop-up menu, select "Delete" command.
The function is only effective when the designated user-defined function block template object was not used.

4.3. Arithmetic Editing

Algorithm editing includes the following types of functions:

Open View Edit: Open the edit view for a specified object control algorithm.
In the module view, double-click the object module, user-defined function block instance (reference) object node, open the edit view of specified object control algorithm; if the designated object control algorithm edit view already opened, then activate it. User-defined function block reference object control algorithm related with the edit view is read-only.

Amend Attribute Editor View: amend designated edit view object properties.
In an edit view, click button, and not selected any graphic object; associated attributes is shown in its properties view, including view proportion, view grid, view description, view height, view width, etc.



Object Name	LD
Object Description	
Object Type	Template Instance
Cycle Time	1 second
Arithmetic	LD
View Scale	75 %
View Grid	OFF
View Description	Hide

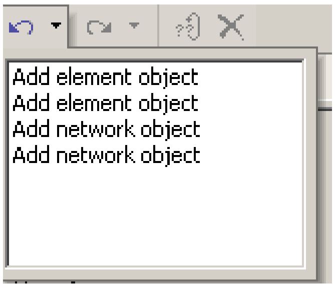
Revise view proportion attribute. After revision, the update shows the scaling of the designated editor view.

Revise view grid attribute. After revision, the update shows the grid state of the designated editing view.

Revise view height (or width) attributes. The editing function is effective only for FBD view. Edit View height (or width) of the effective range: 1,000-30,000 in multiples of 10. If the amended is effective, designated update shows that the high degree of editorial View (or width).

Revise View description attributes. The editing function is effective only for LD view. The revised, updated description of the designated editor is view information display state.
Function Block Configuration
Ladder Configuration

The Revocation of Graphics Editing and Duplication: withdrawal and duplication in the implementation of the various editorial operations in the edit view.


Add element object
Add element object
Add network object
Add network object

In the edit view, use the "edit/revocation (repeat Enforcement)," order in the toolbar or menu. Editorial resume to a specific steps the graphical interface shows. After the resumption of graphical interface, if there is no a graphic object with the relevant data objects matching, they will automatically be deleted (or home state for not related).

4.3.1. Function Block Diagram Configuration

Added Parameter Graphics Object: add a new parameter graphics objects.
In the module view, select a parameters object not involved in the graphical configuration; Then use the "drag and drop" method, the, drag and drop the associated graphic objects to the parameters of its father’s edit view.

"Drag and drop" successful, a graphic object associated with the specified parameters object is added in the designated edit view; in the properties view to show its associated attributes. Only the parameter graphics objects associated with the parameters object can be "dragged and dropped" to its father’s edit view, and the relationship between the two is one-to-one; The addition of new parameters and graphics objects can have parameters (or function block) graphical objects overlap.

Add Function Block Graphic Objects: adding a new function block graphics objects.
Method 1: In the module view, select a function block objects (user-defined function block instance object, user-defined function block reference object, the standard function block reference object, H1 function block reference object) had not participated in a

graphical configuration, then use the "drag and drop" method and drag and drop its associated function block graphic objects to its father’s edit view. After "Drag and drop" successful, a function block graphic object associated with the specified function blocks object is added in the designated edit view; in the properties view to show its associated attributes. Only the function block graphics objects associated with the function block object can be "dragged and dropped" to its father’s edit view, and the relationship between the two is one-to-one; the addition of new function block graphic object can not overlap with the existing parameter (or function block) graphical objects. If the function block object is a user-defined function block instance object, only in its effective control algorithm circumstances, the "drag and drop" operation can be successful.

Method 2: In the template view, select a standard function block template object (or user-defined function block template object), then use the "drag and drop" method drag and drop its associated function block graphic objects to a specific edit view. After "Drag and drop" successful, a function block object associated with the specified function blocks template object (standard function block reference object, user-defined function block reference object) is added under its object node in the designated module view. In the designated edit view, a function block graphic object associated with the specified function blocks object is added; in the properties view to show its associated attributes. The addition of new function block graphic object can not overlap with the existing parameter (or function block) graphical objects; the function block graphic objects corresponding to user-defined function block template object can only be "dragged and dropped" to the edit view of the module object.

Method 3: In the view of the H1 network, select a H1 device function block object without configuration, and then use the "drag and drop" method, and drag and drop its corresponding function block graphic object to its edit view. After "Drag and drop" successful, a H1 function block reference object associated with the specified H1 device function blocks object is added under its object node in the module view; a function block graphic object associated with the specified H1 function blocks reference object is added; in the properties view to show its associated attributes. The addition of new function block graphic object can not overlap with the existing parameter (or function block) graphical objects. The method is effective only for the designated FBD module object in H1 network segment.

H1 Device Function Block Replacement: amend object information of the designated H1 device function block object associated with H1 function block object.

In the view of the H1 network, select a H1 device function block object without configuration, and then use the "drag and drop" method, and drag and drop it to the designated function block graphic object in the edit view of module object. After "replacement" successful, update and show the H1 function block graphic objects and related H1 device function block object configuration identification. The function is only effective for the designated FBD module object in H1 net, and only in the same types of H1 device function block conducted.

Edit Function Block Graphic Object Properties: amend designated function block graphic object properties.
In the edit view, select a function block graphic object; in the properties view to show its associated attributes, including the names of objects, object descriptions, object type, such as the implementation of ID.

Amending Executive ID attribute. After revision, in the edit view, automatic update function block graphic object and object related to the function block graphics display.

Add Link Graphical Objects: add a new line connecting graphical objects.
In the Edit View toolbar, select "Link" tool. Then in the edit view, drawing link graphical object.
During process of the drawing link graphical object, connecting endpoint may be able to be caught. After the success of drawing, automatically add a connection line graphics object between the two endpoints; in the properties view to show its associated attributes. Link object can only connect from the parameters (or function block) graphical object output endpoints to the parameters (or function block) graphical object input endpoint. Output can be connected several times and input can connect only once.

Add Text Graphics Object: add a new text graphics object.
Edit View in the toolbar, select "Text" tool. Then in the edit view, the text graphics is rendering objects.

Drawing success, in view of the designated spots editorial add a new text graphics object; in the properties view to show its associated attributes. Text graphic objects can be overlapped with other graphic objects.

Editing text graphics object properties: amend specified text graphics object properties.
In the edit view, select a text graphics object; in the properties view to show its associated attributes, including the names of objects, object type, location graphics, text color, text content.
Revised text color attributes. After revision, in the edit view, automatically update and display the color of the text graphics object.

Revise text of the attribute. After revision, in the edit view, automatically update and display the content of the text graphics object.

Amend Graphical Object Location And Size:
In the edit view, select the "Choice" tool in toolbar; then select some graphic objects, and use the "Drag" method to change its position. The parameter (or function block) graphics object revised can not overlap with other parameter (or function block) graphical object. After amend successfully, the object graphics automatically update and display (the link object not selected will also be updated).

In the edit view, select the "Choice" tool in toolbar; then select a parameter (or function block) graphic object and use the "Drag" method to change its size. The parameter (or function block) graphics object revised can not overlap with other parameter (or function block) graphical object. After amend successfully, the object graphics automatically update and display (the link object not selected will also be updated). The height of parameter (or function block) graphic object can not be changed.

In the edit view, select the "Choice" tool in toolbar; then select a link graphic object, and use the "Drag" method to change the position of a particular endpoint. During the "Drag" processing, the endpoint can be captured. The property of connecting Endpoint can not be changed. After amend successfully, the object graphics automatically update and display (the link object not selected will also update).

Delete Graphic Objects: delete the specified graphics object.
In the edit view, select some graphic objects; then use the "Delete" shortcut (or use the tool of the "delete" command).

After deletion of graphic objects, the edit view is updated automatically:
Delete parameter (or function block) graphic objects, then related links object automatically be deleted; in the module view, the related parameter (or function block) graphic objects' configuration identifications are reset.

Delete function block graphic objects, the implementation of identifications of other related function block graphics objects subsequent updates.

4.3.2. Ladder Configuration

The Addition of Network Objects: Add new network objects.
In the editorial view, use the command "add network at the front" (or "add network at the back ") in the tools bar (or right-click menu).

This command can only take effect when there are no selected graphical objects and no selected master network objects in the editing view. In the editing view, if a backbone network object has been selected, the new network object will be added before or after it; Otherwise, the new network object will be added at the head (or at the end) of the backbone network objects queue. The new backbone network object includes the following basic graphic objects: network Index, network title, network head, network tail and horizontal network connection line.

The Addition of the Branch Network Objects: Add new branch network objects.
In the editing view, select a horizontal network connection graphical object, and then use the "add network branch” command in the tools bar (or right-click menu).

This command can only take effect when a horizontal network connection line is selected in the editing view. When a specific horizontal network connection line is selected, the new branch network objects can be added. The new network consists of two subnet objects; each subnet object contains three basic graphic objects: network heads, network tails and horizontal networks connection lines.

The Addition of Subnet Objects: Add new subnet objects.

In the editing view, select a subnet objects (network head graphical object), then use the "add subnet at the front" (or "add subnet at the back") command in the tools bar (or right-click menu).

This command can only take effect when a subnet object is selected. The new subnet object includes the following basic graphical objects: network head, network tail and horizontal networks connection line.

The Addition of Contact (Coil): Add new contact (coil) graphical objects.

In the editing view, select a horizontal network connection line graphical object, and then use the "added XXX contact" (or "Add XXX coil") command in the tools bar (or right-click menu).

This command can only take effect when a horizontal network connection line graphical object is selected. The contact graphical objects can be added includes four types: always open, always closed, the rising-edge, and the falling-edge; the coil graphical objects can be added includes six types: positive, negative, positive conversion, negative conversion, bit-set, and reset.

The Addition of Function Blocks Group: add new function blocks groups.

Method 1: in the module view, select a function block Objects (self-defined function block instance objects, standard function block reference object, self-defined function block instance objects), then use the "drag and drop" method, drag the correlation function blocks group object and drop it to the designated horizontal network graphical object in the editing view of its belonged parent. After the "Drag and drop" operation successfully finishes, after the designated horizontal network connection line object in the editing view; add a function block group object. The new function block group object includes a function block graphical object and a number of parameter graphical objects. Only the function block group object which is associated with the function block object can be dragged and dropped to the editing view of its parent object, the relation of the two are one-to-many. If the function block object is a self-defined function block instance objects, only when the control algorithm is valid can the "drag and drop" operation succeed.

Method 2: In the template view, select a standard function block template object (or self-defined function block template objects), then use the "drag and drop" method, drag the corresponding function block group object, and drop it to the designated horizontal network graphical object in the designated editing view. After the "Drag and drop" operation successfully finishes, beneath the object nodes belong to designated editing view in the module view, add a function block object (standard function block reference object, self-defined function block reference object) node corresponding to the designated function template object; After the designated horizontal network connection line graphical objects in the editing view, add a function block group object corresponding to this function block object. The function block group object corresponding to the self-defined function block template object can only be dragged and dropped to the editing view of the template object.

Note: Method 1 calls the same function block object (including references and instances) for many times, that is in a cycle period, the data of the same function block is processed for many times; Method 2 refers the same function block object for many times, although the algorithms of these function block reference objects are the same, but they have their own data area, and can make independent computing and processing. The user should have a good understanding of the differences between the two methods, and avoid the possible editing mistakes.

The Association of Parameter Object and the Graphical Object: associate the parameter objects and the graphical objects.

- In the module view, select a parameter view beneath the module object (or self-defined function block instance object) node, and then use the "drag and drop" method, drag and drop it to the designated contact (coil) graphical object (or parameter graphical object) in the editing view of its parent object. After the "drag and drop" operation successfully finishes, display the name of the associated parameter object on the contact (coil) graphical object in the editing view, and in the module view, and update the graphical configuration flag of the designated parameter object. The parameter object can only be associated to the contact (coil) graphical object (or parameter graphical object) which is matched with its data type. The requirement of contact (coil) graphical object (or parameter graphical object) for the data type of the parameter object can be obtained from the prompt information of the graphical object in the editing view.
- In the editing view, select a contact (coil) graphical object (or parameter graphical object), then use the "revocation of parameters association" command in the tools bar (or right-click menu). On the designated contact (coil) graphical object (or parameter graphical object) in the editing view, revocate the name display of the associated parameter objects; In the module view, update and display the graphical configuration flag of the designated parameter object.
- Note: The self-defined function block (include instance and reference) object and the parameter graphical object related to the standard function block (not including operators and functions) are allowed not to be associated with the parameter object.

Editing the attributes of the contact (coil) graphical object: to edit the attributes of the designated contact (coil) graphical object.

In the editing view, select a contact (coil) graphical object, and display the related attributes in the attributes view, including object type, data type, and element type.

- Editing element type attributes. After modification, the element styles of the contact (coil) graphical object is automatically updated and displayed.

Edit the Attributes of the Backbone Network Object: Modify the Attributes of the Backbone Network Object**.**

In the editing view, select a network graphical object beneath the backbone network object, display the related attributes of the backbone network object in the attribute view, including the object name, the object type, the network identification, the network title, the network description and the end pattern.

- Modify the network title attributes. After modification, in the editing view, the network title graphical object of the backbone network object is automatically updated and displayed.
- Modify the network description attributes. After modification, if the view description attribute of the current editing view is "display", the network description graphical object of the backbone network object is automatically updated and displayed.
- Modify the end style attributes. After modification, in the editing view, the network end graphical object's end style is automatically updated and displayed. If the end style is "Jump", its Jump location attributes is added in the attributes view.
- Modify the jump location attributes. After modification, in the editing view, the network end graphical object's jump location is automatically updated and displayed.

Deletion of the Graphical Objects: to delete the specified graphical object.

- In the editing view, select a contact (coil) graphical object; then use the "Delete" shortcuts or use the "Delete" command in the tools bar (or right-click menu). In the editing view, delete the designated contact (coil) graphical object; If the contact (coil) graphical object is associated with parameter object, then in the module view, update the graphical configuration flags of the related parameter object.
- In the editing view, select a function block graphical object; then use the "Delete" shortcuts or use the "Delete" command in the tools bar (or right-click menu),. In the editing view, delete all graphical objects (including function block graphical objects and related parameter graphical objects) beneath the specified function blocks; In the module view, update the graphical configuration flags of the related function block objects. If the deleted parameter graphical object is associated with parameter object, then in the module view, update the graphical configuration flags of the related parameter object.
- In the editing view, select the network head graphical object of the non-first subnet object beneath the network branch object, then use the "Delete" shortcuts or use the "Delete" command in the tools bar (or right-click menu). In the editing view, delete all the graphical objects under the specified subnet object If there is only one subnet object left in the network branch object which the deleted subnet object is affiliated to, then all the graphical objects under the left subnet object are automatically merged into the network objects affiliated to this network branch object.
- In the editing view, select the network head graphical object of the first subnet object beneath the network branch object, then use the "Delete" shortcuts or use the "Delete" command in the tools bar (or right-click menu). The prompt will come and ask if you want to delete the entire branch network object. If the answer is “yes”, then

delete all the graphical objects under this network branch in the editing view; otherwise, just delete the selected subnet object, and the expected results as mentioned above.

- In the editing view, select the network head graphical objects of some backbone network objects, then use the "Delete" shortcuts or use the "Delete" command in the tools bar (or right-click menu). In the editing view, delete all the graphical objects under these backbone network objects. The network index of other related backbone network objects will update automatically.

Move Graphical Objects: move the designated graphical objects.

- In the editing view, select a contact (coil) graphical object, then press the "Alt" key, and use the "drag and drop" method at the same time, move it to the designated horizontal network connection line graphical object. After the "drag and drop" operation successfully finishes, in the editing view, move the designated contact (coil) graphical object to the back of the designated horizontal network connection line graphical object.
- In the editing view, select a function block graphical object; then press the "Alt" key, and use the "drag and drop" method at the same time, move it to the designated horizontal network connection line graphical object. After the "drag and drop" operation successfully finishes, in the editing view, move all the graphical objects (including function block graphical objects and the related parameter graphical objects) under the designated function block group to the back of the designated network connection line graphical object.
- In the editing view, select the network head graphical object of some subnet object, then press the "Alt" key, and use the "drag and drop" method at the same time, move it to the network head graphical objects of other designated subnet objects. After the "drag and drop" operation successfully finishes, in the editing view, move the entire graphical object under the designated source subnet object to the front of the designated destination subnet object. Subnet object can not be moved to the subnet objects under it. If the source subnet object and the destination subnet object are under the same network branch object, and the destination subnet object is the last subnet object under the network branch object, the source subnet will be moved to the back of the destination subnet object. If there is only one subnet object left in the network branch object which the source subnet object is affiliated to, then all the graphical objects under the left subnet object are automatically merged into the network objects affiliated to this network branch object.
- In the editing view, select the network head graphical object of the first subnet object of some network branch object, then press the "Alt" key, and use the "drag and drop" method at the same time, move it to the designated horizontal network connection line graphical object. After the "drag and drop" operation successfully finishes, in the editing view, the editing view, move all the graphical object to the back of the designated horizontal network connection line graphical object. The network branch object can not be moved to back of the horizontal network connection line graphical object.

- In the editing view, select the network head graphical object of some backbone network object, then press the "Alt" key, and use and use the "drag and drop" method at the same time, move it to the network head graphical object of other backbone network object. After the "drag and drop" operation successfully finishes, in the editing view, if the designated source backbone network object is at the front of the designated destination backbone network object, then all the graphical object under the source backbone network object will be moved to the back of the destination backbone network object; and vice versa. Related network index of the backbone network object will automatically update.

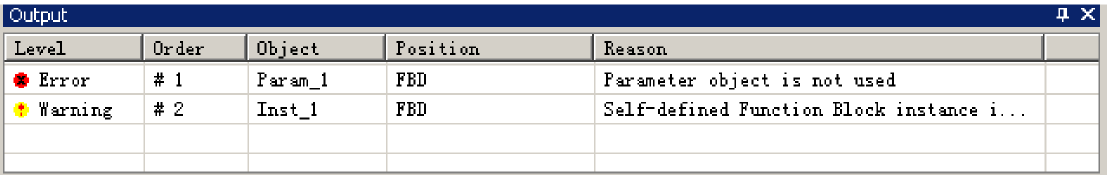
Copy, Paste Graphical Object: copy, paste all graphical objects under the designated backbone network object.

- In the editing view of the ladder diagram, select the network head graphical object of some backbone network object, then use the "edit -> copy" command in the menu, copy all the graphical objects under the backbone network object to the system clipboard.
- In the editing view of a arbitrary ladder diagram, use "edit -> paste" command in the menu, you can paste all the copied graphical object into it. If a backbone network object has been chosn before the paste, then the paste is performed at the front of the selected backbone network object, otherwise, the paste is performed at the queue tail of the backbone network object.
- Note：if the graphical object copied contains the graphical object of the self-defined function blocks (including reference and instances) object, then it can only be copied to the editing view of the module object; If the current self-defined function block (including reference and instances) object does not match with the graphical object pasted , the paste can not be performed; If the current parameter object does not match with the graphical object pasted, then the original association relations will not be kept.

4.3.3. Algorithm Check-Up

Algorithm check-up is used to check the control algorithm of the module object (or the self-defined function block instance object), it mainly includes the following types of functions:

The Syntax Check-Up: check syntax of the specific object’s control algorithm.
In the module view, Right click on the module object (or self-defined function block instance object) node; in the pop-up menu, select the "validity check-up" command.



Level	Order	Object	Position	Reason
Error	# 1	Param_1	FBD	Parameter object is not used
Warning	# 2	Inst_1	FBD	Self-defined Function Block instance i...

The command is only effective to the module objects (or self-defined function block instance object). It is used to perform the syntax validity check-up of the specified object's control algorithm. The control algorithm syntax check-up is performed on all the self-defined function block objects that the module object contains. The result of the check-up is displayed in the output view, including: level, serial number, object name, location, reason, and so forth.

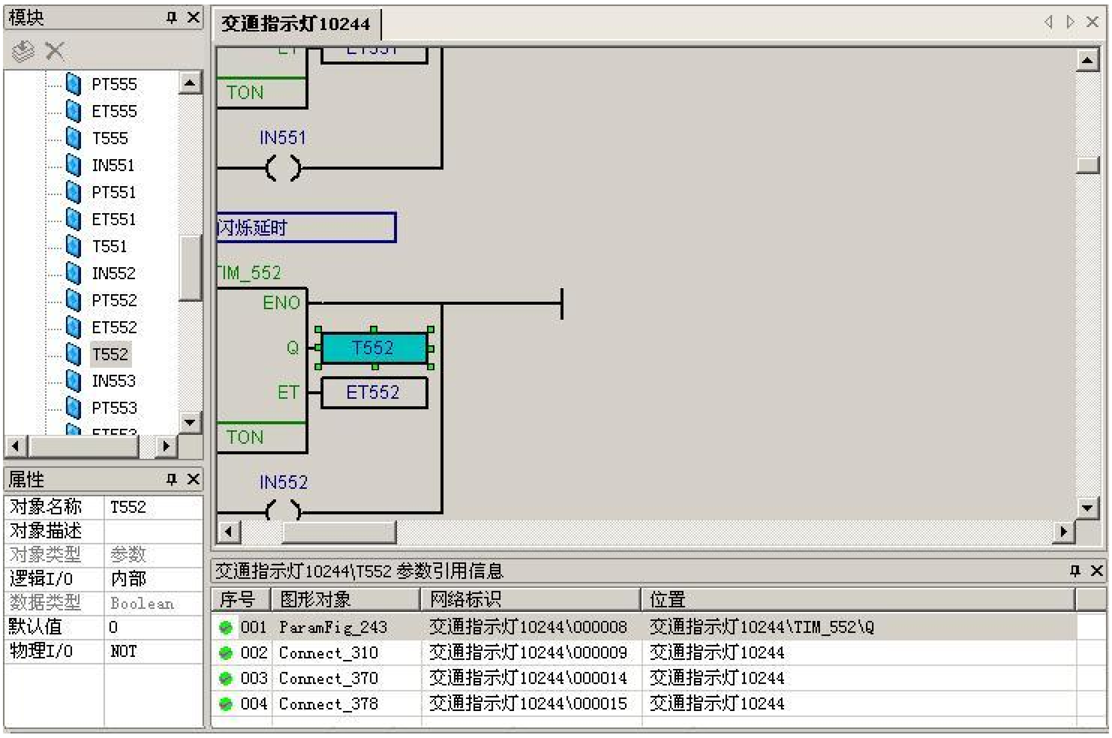
Editing faults Location: According to syntax check-up results, locate the data (or graphical) object witch has editing faults (or warnings).

In the output view, double-click the check-up result message of the specific algorithm.

In the module (or editing) view, select the relevant data (or graphical) objects. If the related editing view is not opened, then open it; if the editing view is not the current active window, and then activate it.

LD Parameter Reference Check-Up: check-up the graphical objects of the specific LD parameters.

In the module view, select the "parameter reference" command of the specific LD parameter object's right-click menu, then output the message of the graphical objects which reference the parameter.



对象名称	T552
对象描述	
对象类型	参数
逻辑I/O	内部
数据类型	Boolean
默认值	0
物理I/O	NOT

交通指示灯10244\T552 参数引用信息

序号	图形对象	网络标识	位置
001	ParamFig_243	交通指示灯10244\000008	交通指示灯10244\TIM_552\Q
002	Connect_310	交通指示灯10244\000009	交通指示灯10244
003	Connect_370	交通指示灯10244\000014	交通指示灯10244
004	Connect_378	交通指示灯10244\000015	交通指示灯10244

The command is only effective to the LD parameter object, the output information of the graphical object includes: the name of the graphical object, the network it belongs to and the network identification, its concrete location in the network. Users can locate the specific graphical objects by double-clicking the corresponding entry.

This function can be used in conjunction with a syntax check-up.

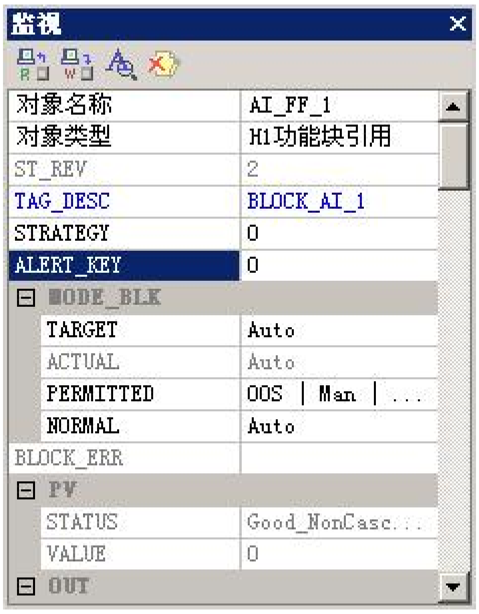
(1) The validity check-up, double-click the LD parameters object which has errors or warnings in the output window, and locate the parameter object in the module window.

(2) In the module window, select the "parameter reference" command in the right-click menu of the parameter object, and output the parameter reference information;
(3) Double-click the reference information entry, locate the graphical objects which referred the parameters.

4.3.4. Online Monitoring

The online monitoring function is used to perform the on-line monitor and control task of the data objects under the module object, and it has the following functions:

Start online monitoring: Start the on-line monitoring of the designated module object.
In the main menu, select "debugging / launch online monitoring" command.



监视

对象名称	AI_FF_1
对象类型	H1功能块引用
ST_REV	2
TAG_DESC	BLOCK_AI_1
STRATEGY	0
ALERT_KEY	0
⊟ MODE_BLK	
TARGET	Auto
ACTUAL	Auto
PERMITTED	OOS | Man | ...
NORMAL	Auto
BLOCK_ERR	
⊟ PV	
STATUS	Good_NonCasc...
VALUE	0
⊟ OUT	

This command is only effective for module object which is in editing state. If the control algorithm validity check-up of the module object is not successful, the on-line monitoring state (FBD module object allows warnings) can be entered; Module object which has not started the download, or the download is not successful, or the editing or modification has been made after the last successful download, can not entering the on-line monitoring state. After a successful launch, the operations such as module management, data management, template management, algorithm editing, and algorithm check-up are all prohibited. Re-initialize the module view, the monitor view, and the output view, and close control algorithm’s editing view of all the self-defined function block template objects, update and display the graphical object in all editing view which is in the open-state; the data reading thread is automatically started and entering the on-line monitor state.

Stop On-line Monitoring: stop on-line monitoring of specific module objects.
In the main menu, select the "debug / stop online monitoring" command.
This command is only effective for the module object which is in the on-line monitoring state. After the on-line monitoring is stopped, the data reading thread is automatically stopped; Reinitialize the module view and the monitoring view, close the control

algorithm's editing view of all self-defined function block reference object, update and display the graphical objects in all the editing view which is in the open state; the module object restores to the editing state.

On-line Data Reading And Real-Time Display

After successfully launching online monitoring, in accordance with predetermined cycles, real-time updates and display the following contents:

- In the FBD editing view, the current value of the related parameter object in all the parameters graphical objects, and the function block graphical objects.
- In the LD editing view, the current value of the related parameter object in all the parameters graphical objects, and the contact (coil) graphical objects.
- In the monitoring view, the current value of all the parameter objects under the designated objects.

In the module view, select a module object or a function block object (standard function block reference object, self-defined function block instance object, self-defined function block reference object) node; in the monitoring view, switch to display the current value of the parameter object under the designated object. In the monitoring view, the current value of the parameter object under the FBD module object, the self-defined function block instance object, and the self-defined function block reference object is read-only; the current value of parameter objects under the LD standard function block reference object is not displayed.

In the editing view, select a function block graphical object; In the monitoring view, switch to display the current value of the parameter objects under the function block objects associated with the designated function block graphical object.

On-line Writing Parameters: to modify the current value of the parameter object on line. In the monitoring view, select a writable parameter object, modify its current value. If the modification is valid and the value after modification is different with the current value, then it is showed in a remarkable blue color (e.g. the TAG_DESC parameter in the above figure).

If there are parameter objects in the monitoring view is in the “Modification” state, users can select the “write parameter” command in the tools bar of the monitoring view, and perform on-line writing operations. If the modification is successful, the relevant parameters object automatically to refresh its state; Multiple Modifications are allowed to be performed at the same time.

On-line Reading Parameters: reading the current value of the parameter objects on-line under the H1 function blocks.

In the monitoring view, switch and display a H1 function block object; select a parameter object under it, and select the “read parameter” command in the tools bar of the monitoring view. If the reading is successful, the current value of the relevant parameter is refreshed.

4.3.5. Simulation Debug

Simulation debug is used for the completion of Ladder Diagram (LD) module’s offline simulation debugging tasks, including the following type’s functions:

Start Simulation Debug: Start designated LD module’s simulation debug.

In the main menu, select "debug / start simulation debug" order.
The order is effective only for an edited state LD module objects. The check of LD module’s control algorithm validity is unsuccessful, failing for calling LD simulator; the state of simulation debug could not be entered. After successful start-up, it will prohibit from any module management, data management, template management, editing algorithm, the algorithm inspection operation; Re-initialize the module, monitor, output view, closing all-defined function block control algorithms template object editors view, update all in the current show state opened the editorial view graphic object; automatically activated cycle threads read data into the simulation debug state.

Stop Simulation Debug: stop simulation debug of designated LD module.
In the main menu, select "debug / simulation debug stop" order.
The order is effectively only for a state of LD simulation debugging module. After stopping simulation debug, the thread of read cycle data will be automatically terminated. Re-initialize the module, surveillance view, closing all user-defined function block reference the editorial control algorithm object view, update show the current opened state of all the editorial view graphic object; restore Editor module object to the state.

Cycle Reading Simulation Data and Real-Time Display:
After the success of LD simulation debug launched, it will accordant with predetermined cycles, and real-time updates the following content:

- LD edit view, all parameters of graphics objects, contact (coil) graphical objects related to the current value of target parameters.
- Surveillance view, all parameters specified under the target object to the current value.

In the module view, selected a module object or function block (examples or cited) target node; In surveillance view, switching parameters shows that under the designated target object to the current value.
In the editorial view, select a function block graphic object; In surveillance view, switching function block graphics display designated targets linked to the function block target object parameters to the current value.

Simulation Write Parameters: LD write parameters in simulation debugging state.
In surveillance view, select a parameter of LD that could be written, modify its current value. If the revised parameter value is effective and changes with the current value are

not the same, the blue eye-catching logo is used, in order to show that in the "modification" status.

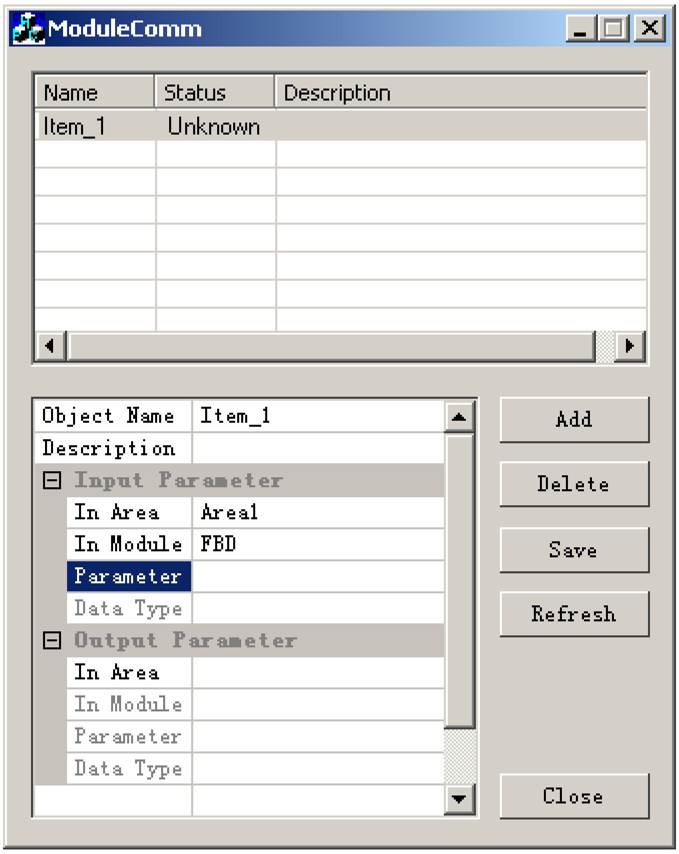
If there is the "modified" state parameters object in view surveillance that currently displayed in the List object, user can select "wrote parameters" orders in view of the tool, carry through simulation write. Once write parameters are successful, the relevant parameters objects will automatically update for the restoration state; and allow writing object of the amendment of many parameters.

4.3.6. Communication between Modules

Communication between modules is target for the completion of communication module configuration; several major functions are as shown in the following:

Open Communication Configuration Information: Open the original communication module configuration information.

Use the menu, type "application / Communication between Modules” order.



Name	Status	Description
Item_1	Unknown	

Object Name	Item_1
Description	
⊟ Input Parameter	
In Area	Area1
In Module	FBD
Parameter	
Data Type	
⊟ Output Parameter	
In Area	
In Module	
Parameter	
Data Type	

Open the “Communication between Modules configuration" dialog box, initialize interface display. If the original configuration information has the failure contents, it will replace attribute values of related communications.

Refresh Communications Configuration Information: inspect and update the communication module configuration information.

in the “Communication between Modules configuration" dialog box, click "refresh" button.

Checks validity of the current configuration information; If the current configuration information in the contents have failure content, replace attribute values of related communications object; Update interface display.

Keep Communication Configuration Information: preserve the current configuration information of communication module.
In the “Communication between Modules configuration" dialog box, user can click on the "Save" button.
After preservation, suggest that the success or failure of the relevant information.

Add Communications Targets: add a new module communication objects.
In the “Communication between Modules configuration" dialog box; user can click on "Add" button.
In the communications List box, add a new communications target list; in the Attribute List box indicate their associated attributes.

Delete Communications Targets: to delete the specified communication module object.
In the Communications between modules configuration dialog box List, select the communications object, and then click on "delete" button.
After deleting designated communications Object; in the communications list box, and the list of recent targets will be set to the current options; in the attribute List box indicate their associated attributes.

Edit Communications Object Properties: modify attribute information of designated communication module object
In the configuration dialog box of communications List object communications between modules, select a list of targeted communications; in the attribute List box indicate their associated attributes, including the names of objects, object descriptions, input parameters, output parameters, and other attributes of that input and output parameters respectively, and including their regional, owned module, parameter names, data types, and other attributes of items.

- Amend object name attribute. The name of the Communications object can not be empty, not including the "/ \:*?"<>|." and other special symbols, not duplicating names of other communications targets. If the change is effective, updating the list box displayed objects related communications List.
- Amend Object Description attribute. After revised, update target list box shown that communication of relevant list.
- Amend input parameters attributes:

- Amend the respective regional attributes. After revision, replace input parameters of the respective module, parameter name, data type attributes.
- Amend module attributes. After revision, reset input parameters parameter name, data type attributes. If the output parameters modules and the input one is same, reset the module output parameters, parameter name, data type attributes.
- Amend parameter name attribute. After revision, update automatically displaying the

data type of input parameters attributes. If the output data types and parameters and the input one do not match, reset output parameter name and data type parameter attributes. Data type matching conditions: In addition to the same type, the Boolean-type under LD module with the FBD module (UINT 8) is in terms of matching.

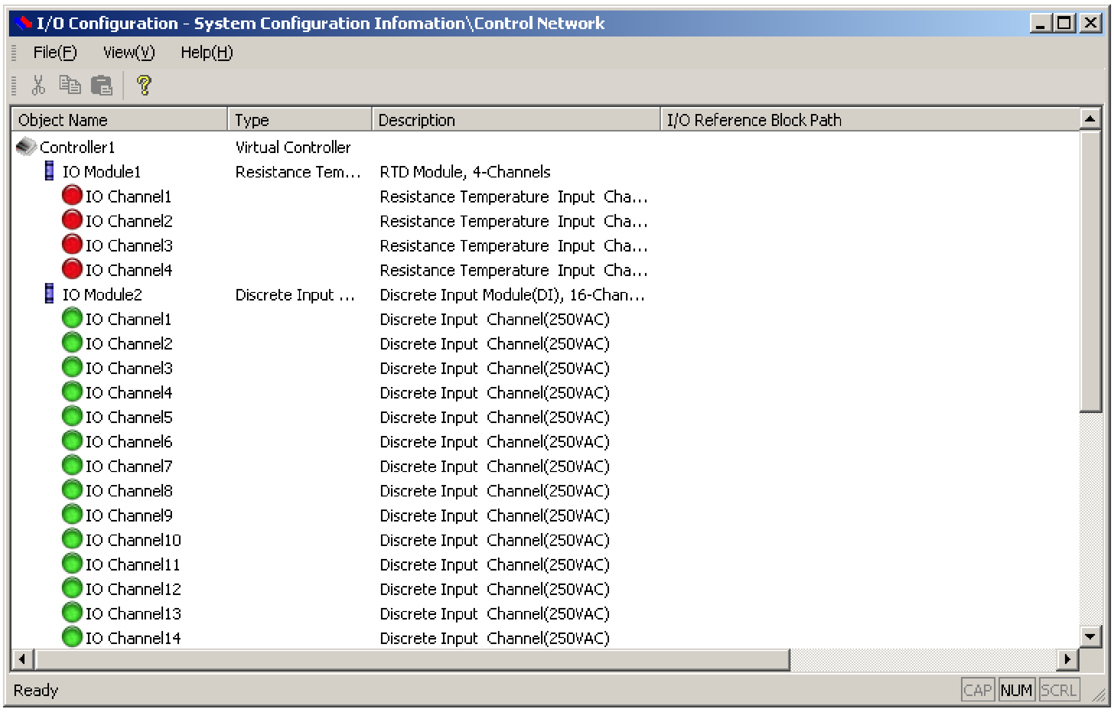
- Revise output parameters attributes:

- Amend the respective attributes. After revision, reset the respective output parameters module, parameter name, and data type attributes.
- Amend the respective module attributes. After revision, reset input parameters parameter name, data type attributes. Output parameters module optional item should not contain input parameters of their module.
- Amend parameter name attribute. After revision, updated automatically displaying the output parameter attribute data types. If the data type of input parameters has been identified, Output parameters module optional item should be the matched parameter object name.

4.3.7. I/O Mapping Tools

Introduction

NCS4000 I/O mapping tools, also called I/O configuration tools, are mainly used for the establishment of virtual controller I/O modules and communication with the control channel strategy and block the function of the parameter mapping relations.



I/O Configuration - System Configuration Infomation\Control Network

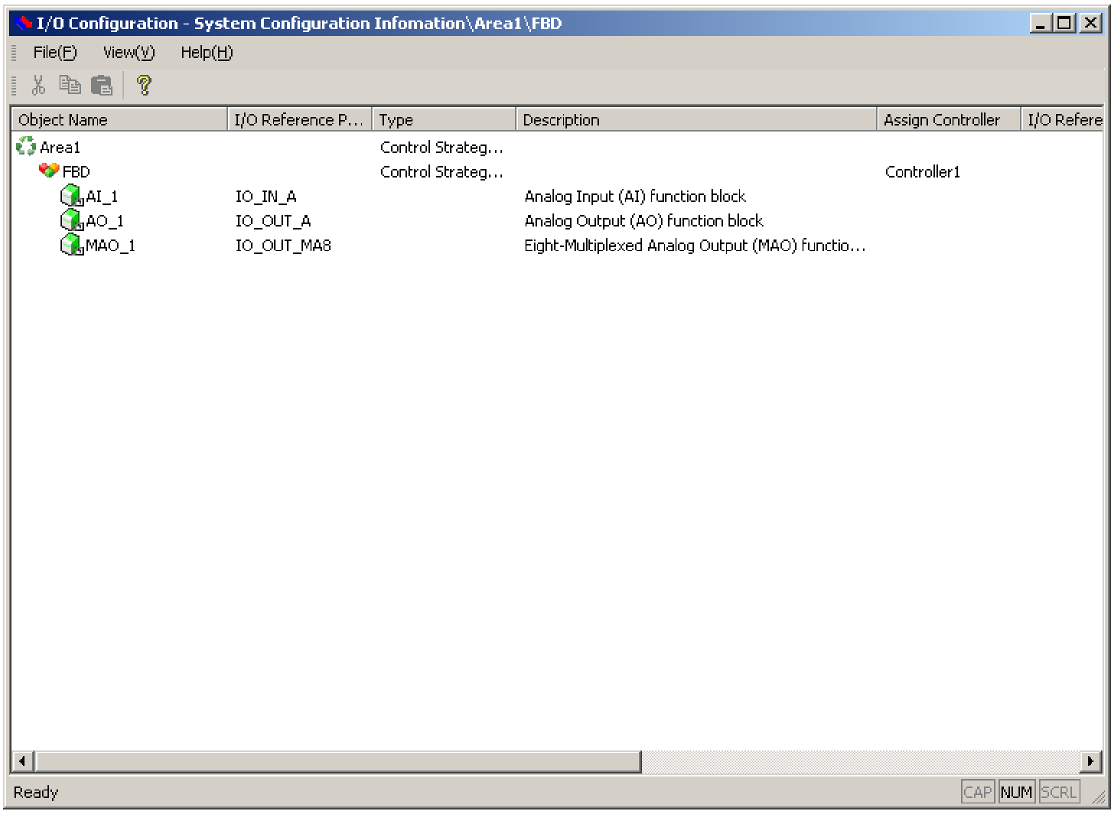
Object Name	Type	Description	I/O Reference Block Path
Controller1	Virtual Controller		
IO Module1	Resistance Tem...	RTD Module, 4-Channels	
IO Channel1		Resistance Temperature Input Cha...	
IO Channel2		Resistance Temperature Input Cha...	
IO Channel3		Resistance Temperature Input Cha...	
IO Channel4		Resistance Temperature Input Cha...	
IO Module2	Discrete Input ...	Discrete Input Module(DI), 16-Chan...	
IO Channel1		Discrete Input Channel(250VAC)	
IO Channel2		Discrete Input Channel(250VAC)	
IO Channel3		Discrete Input Channel(250VAC)	
IO Channel4		Discrete Input Channel(250VAC)	
IO Channel5		Discrete Input Channel(250VAC)	
IO Channel6		Discrete Input Channel(250VAC)	
IO Channel7		Discrete Input Channel(250VAC)	
IO Channel8		Discrete Input Channel(250VAC)	
IO Channel9		Discrete Input Channel(250VAC)	
IO Channel10		Discrete Input Channel(250VAC)	
IO Channel11		Discrete Input Channel(250VAC)	
IO Channel12		Discrete Input Channel(250VAC)	
IO Channel13		Discrete Input Channel(250VAC)	
IO Channel14		Discrete Input Channel(250VAC)	

I/O Configure Information Browsing and Management

This software provides two basic modes of browsing I/O configuration information: Based on the control strategies and control network.

Based on the control strategy, the users can see configuration information of region, module, function block, parameter, etc. List file names with the related I/O modules or channel; Contrary to Network-based control, what the users see is that the controller, I/O modules, I/O modules channel, and so on. List file names with their associated function block or parameters.

In I/O mapping tool, select the **control network** or **control strategy** item of the **View** menu, then user can switch between them.


I/O Configuration - System Configuration Infomation\Area1\FBD
File(F)
View(V)
Help(H)
Ready
CAP
NUM
SCRL

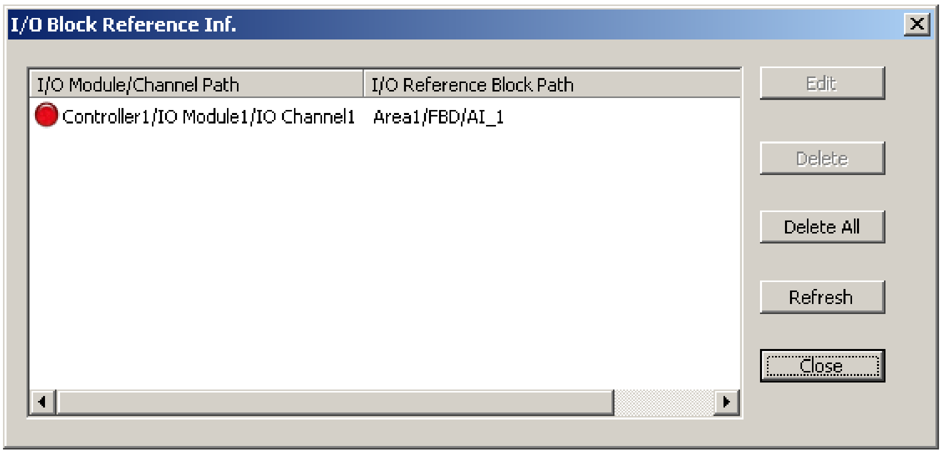
Object Name	I/O Reference P...	Type	Description	Assign Controller	I/O Refere
Area1		Control Strateg...			
FBD		Control Strateg...		Controller1	
AI_1	IO_IN_A		Analog Input (AI) function block		
AO_1	IO_OUT_A		Analog Output (AO) function block		
MAO_1	IO_OUT_MA8		Eight-Multiplexed Analog Output (MAO) functio...		

Filtering

When the project is larger, we can use this software filtering to reduce browsing scope. There are four kinds of filtering options: filtering by module or region, filtering by controller or I/O module. The menus of these four functions are all in the document menu.
When choosing type of filtering by XX, the software browser window will pop-up and suggest users to choose XX types of objects.

I/O function blocks referenced Information Management

Under control network mode, the mouse right click on the I/O modules or channel object, select I/O function block information menu items, then pop-up dialog function block of cited information from the I/O channel. Browsing the cited information from right key menu is shown as follows:



I/O Module/Channel Path	I/O Reference Block Path
● Controller1/IO Module1/IO Channel1	Area1/FBD/AI_1

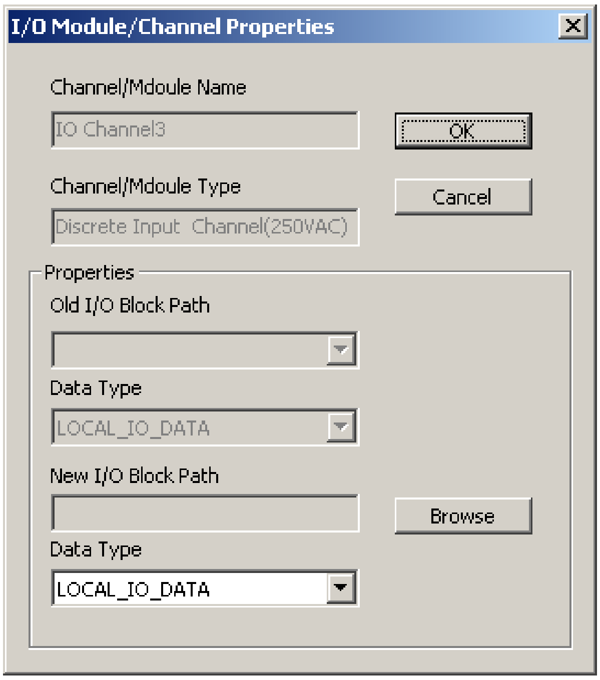
In the dialog box user can delete the configured citer information to manage, select "Edit" button will activate the control strategy development environment, edit the module object of current function block.

In the control strategy mode, the mouse right click on the function block or parameters object, select *Cancel I/O configuration information* menu item will deleted the I/O configuration information related the current function block.

Mapping From I/O Modules and Channel Perspective

Users can map from I/O module or a channel perspective. In the network-based control browser mode, user can select function blocks mapping from I/O module or channel menu. Attribute dialog box of I/O module or channel will be pop up. From I/O channel right view menu attribute dialog box is shown in the next figure.

Click on the Browse button and select the function blocks or ladder diagram parameters, View dialog function block can be displayed and parameters will be automatically filtered by type. If there is no channel mapping I/O function block, the original I/O function block path will be empty.



I/O Module/Channel Properties

Channel/Mdoule Name: IO Channel3

Channel/Mdoule Type: Discrete Input Channel(250VAC)

Properties

Old I/O Block Path:

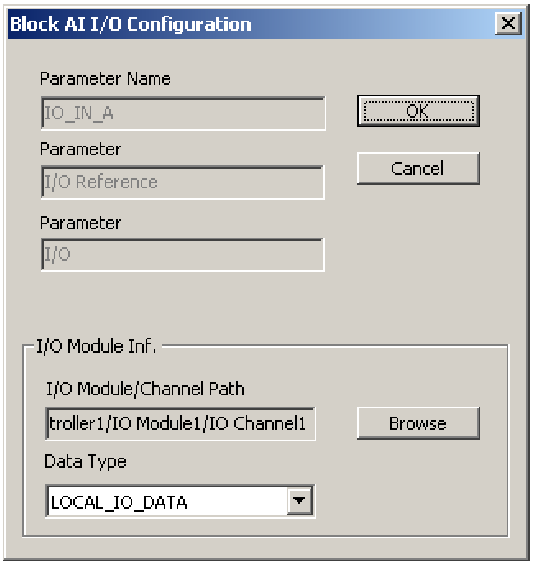
Data Type: LOCAL_IO_DATA

New I/O Block Path:

Data Type: LOCAL_IO_DATA

Mapping From the Perspective of Function Blocks and Parameter

Users can also map from the angle of I/O function block or parameters. Under the browser mode of control strategy based, the mouse double-click function block or parameters objects will pop up I/O function block configuration dialog box as next figure:



Block AI I/O Configuration

Parameter Name: IO_IN_A

Parameter: I/O Reference

Parameter: I/O

I/O Module Inf.

I/O Module/Channel Path: troller1/IO Module1/IO Channel1

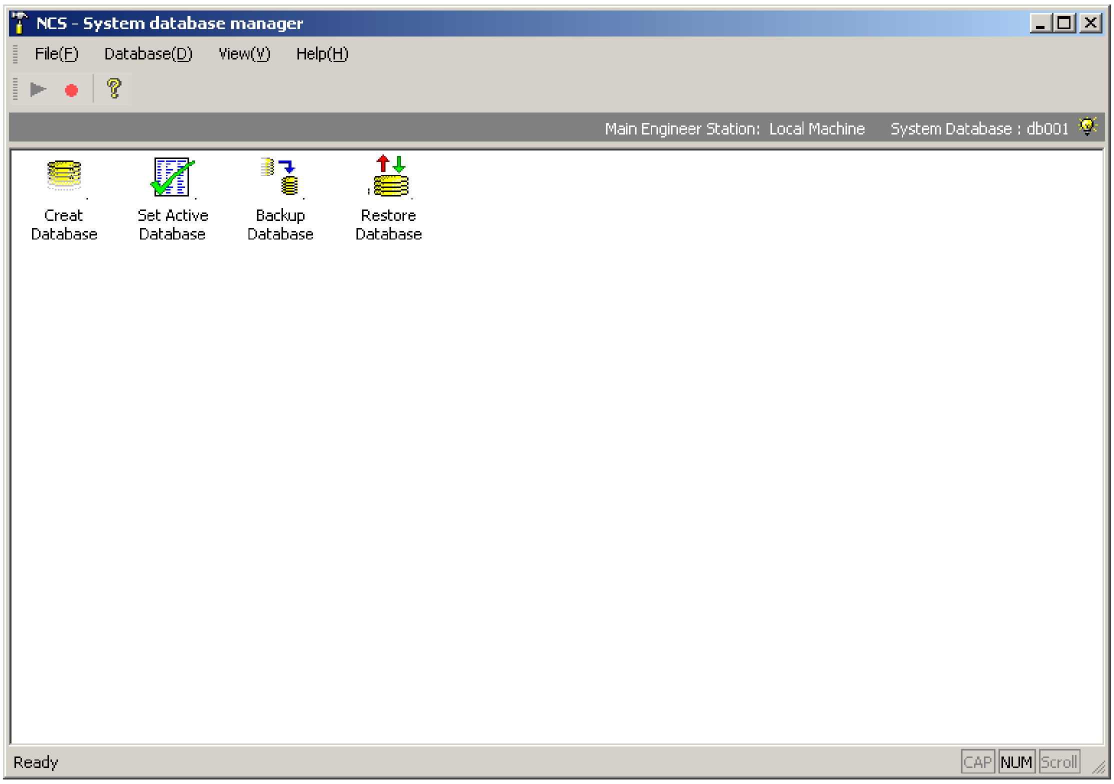
Data Type: LOCAL_IO_DATA

Click on the Browse button and select the available I/O modules or channel. Browse dialog box will be automatically filtered by type.

4.4. System Information Database Manage Tools

Introduction

NCS4000 system information database manage tools (NCS4000 DBManager) integrates all the relevant information database management functions of NCS4000 system, can control system information database services start, stop, the system can create a new database, and database information system designated for activation, backup, recovery operations. The engineers can set up stations.



Note:

1. Please do not hand information on the system directory and database files operation, as this will cause system information database objects to information chaos, causing the collapse of NCS4000 system, information system database for any operating system information requested through the database management tools.

2. For non main engineers station, the tools only provide engineers to install at the main function.

3. Before launching the database, the directory database files can not be other applications that visit, otherwise the database can not be activated. For example, compression software open the document in the database, operating system or other resource management tool for browsing folder contents, the database will lead to launch failures.

4. When database is operating, the database file folders are locked, unable to open the folder or modify existing documents. If you copy files to the folder, you can be different from the copy success, but the "revocation of copies" or "revoke mobile" in the database operation can only be carried out after the cessation of operations. After the cessation of operation of the database server, folders lock automatic discharge.

5. When database server is preserving data, and if the server was shut down, the next time will be automatically activated before the resumption of the depositary state; Data

such as the preservation cause of the accident led to the detection of invalid object ID, automatic cancellation last preservation results and alarm user to be prompted to stop operating, reopen databases, for a period of time (about a few minutes) users will not be able to continue to preserve data. If you ignore errors, before the next launch, the database server will no longer detect errors.

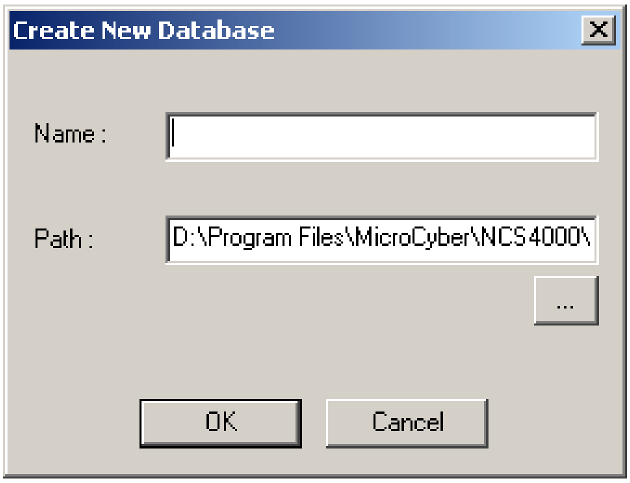
6. If a user operations (database) has abnormal status, once possible, follow-up recommendations to stop the preservation of data operation, first backup database, to try to solve the problem.

Database Start and Stop Services

Users may through a menu item or toolbar button to achieve control of the database services. Services are run by the state toolbar button to judge. Stop database services need a few seconds of time.

4.4.1. New Database

Selecting *database* -> *new* menu items, or Double click on the *new database* icon, the pop-up dialog os new database. The window can be imported and the name of the new database storage path.



Create New Database

Name:

Path: D:\Program Files\MicroCyber\NCS4000\

By default, database storage path is NCS4000 install directory, users can choose. New database files stored in the directory with the same name in the database. For example, the new database called DbDemo, for the storage path c: \ NCS4000 \ Bin, the new database will be created to c: \ NCS4000 \ Bin \ DbDemo directory.

After the success of the creation of the new database, management tools will be asked whether the new database will be set activation.

Note: Following is the name of the reserved NCS4000 system:

1. Crash
2. Dd

3. Log
4. Modules
5. baseproj

4.4.2. Configuration and Activation of the Database

Selecting *database -> activation database* set up menu items, or set up the mouse double-click *activation database* icon. Management tools directly pop up a folder browser dialog, suggesting that users choose to activate the database. The initial directory dialog default position for the NCS4000 installation directory. Choose to enable database directory and mouse click OK button.
The main tool in the management of the title of the window will show the current (activated) database storage paths and names.

Note: Users need to identify a directory of your choice is correct database directory, NCS4000 information system database directory service procedures are not the effectiveness of the inspections.

4.4.3. Backup Database

Selecting *database -> backup database* menu items, or Double click on the *database backup* icon. Management tools directly pop up a folder browser dialog, suggesting that users choose to backup database. The initial directory dialog default position for the NCS4000 installation directory. Option to backup database directory and mouse click OK button.

If the database is the current backup activated database, and database services are running, then backup is impossible. Under such circumstances, management tools will prompt users that database is in the open state, and asked whether the closure (that is to stop database services). Users choose the answer is yes, database services will be stopped and then proceed to backup work.

Each database backup file will has a response, and backup success will prompt users. By default backup file name and the database name the same as the expansion called tar.bz2, stored in the database where the same directory. For example, backup database c: \ NCS4000 \ Bin \ DbDemo, while backup documents for the c: \ NCS4000 \ Bin \ DbDemo.tar.bz2, users can pop up in the "backup database" dialog box installed backup file name, as well as document storage backup path.

4.4.4. Resume Database

Selecting *database -> resume database* menu item, or double-click the Mouse icon *resume database*. Management tools will directly choose a document dialog, the user to be prompted to choose the database backup file (tar.bz2).

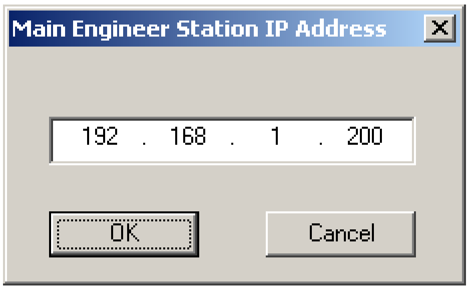
When restoration is successful, it will automatically build management tool to restore the directory database and the data.

Users can set the names and the storage directory path of the restoration database in the pop-up dialog "resume database".

During restoring the database, the restored directory database can not be browser and applicator, such as the use, it could cause loss of data, and other unpredictable problems. If there is a resumption errors, please check reasons, and then resumed. If there is a browser window looking at the catalog or its sub-folder contents, directory can be switched on or off window; If other applications using directory in the file, closed the corresponding applications.

4.4.5. The Engineers Station

Selecting database -> main engineer setting menu item will pop up the dialog of setting IP address of main engineer station.


Main Engineer Station IP Address

192 . 168 . 1 . 200

The default is the machine address. Once setting up the server IP address, other NCS4000 software to link database information system will automatically connect to the main station engineers.

Note:

1. System information database management tools can only monitor and control the machine database services. If the non-stop main engineer is of the machine, the management tools in the start-up and stop database services menu and toolbar will automatically lapse.

2. If NCS4000 other system components (resource management, configuration control strategy development environment, etc.) has been launched and not closed before revising the "main engineers IP address" and these components are still links before the main engineers Station Information System database. Therefore, to prevent unnecessary differences, please revise the "main engineers IP address" before determining NCS4000 other components in the system has stopped running.

4.5. System Information Database Upgrade Tool

Introduction

NCS4000 information system database upgrade tool (NCS4000 DBUpgrade) is mainly used for upgrading NCS4000 information database system, including upgrading the original database software compatible to the current database, the database add / reduce operational controller, and other functions.

The Original Database Upgrade

DBUpgrade only upgrade the current system to the updated database. Upgrade mode is divided into two types, manual updates and automatic updates, manual upgrades are as follows:

1. Click on the "select database" button, the dialog box in the pop-up choices need to upgrade the system directory information database

2. Click on the "upgrade" button, DBUpgrade began upgrading the system to the choosen Information Database

Automatic update is a new version of the user NCS4000 software (resource management, control strategy development environment, as well as I/O mapping tools), users will be asked whether to upgrade the database, click on "is", DBUpgrade will activate, automatic update databases; Click on the "no", automatic withdrawal, users can not pass the new version of the original database software operation.

Increase / Decrease Operational Controller

NCS4000 controller has version information; different versions of the controller have different functions. Controller has two versions of firmware version, as well as versions of the agreement can be observed in the explorer, which firmware version number associated with the PC. The NCS4000 each released new software package, will be limited database operated controller, the limited information in the directory database information system under the version.ini paper records.

The adoption of the following operating systems can provide information databases increase / decrease operational controller (such as the directory database for db001):

1. Under version.ini documents open db001

2. Find
[CONTROLLERVERSIONHEADER]
Number = XX
If the new controller will XX - plus-one; If reduce controller, will XX subtract one.

3. If there is a new controller, it added
[CONTROLLERVERSION n]

Version = X. XX

N is replacement by number of the above numerical, X. XX is the controller firmware version number in the major version number.

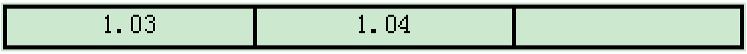
If reduce controller, then delete the relevant paragraph [CONTROLLERVERSION n] information, and update other [CONTROLLERVERSION n] n in the numerical order to ensure n 1 as the base, order growth

4. Preservation version.ini document, launched DBUpgrade tools, the upgrade operation db001

The current version of the database

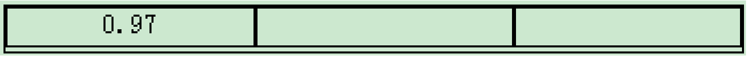
1.05

Compatible with the original version of the database



1.03	1.04	

Operational Controller major version number

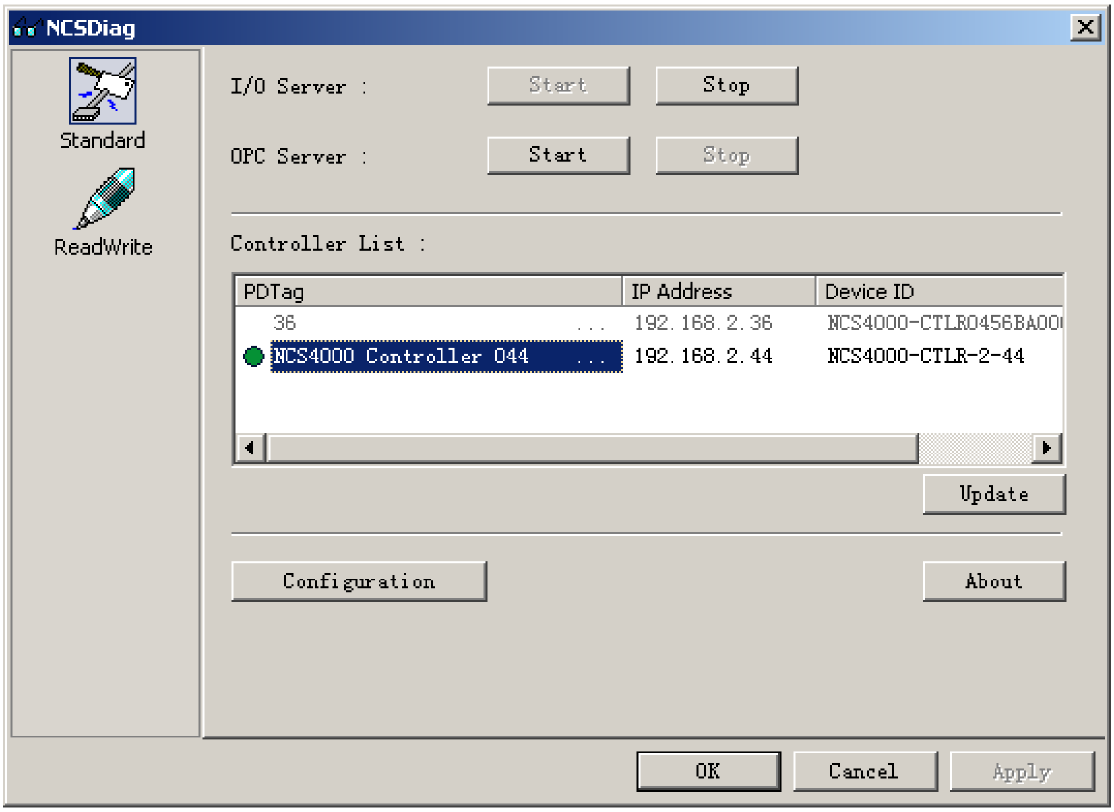


0.97		

4.6. NCS4000 system diagnose tools

Introduction

NCS4000 system diagnostic tools integrate some management and monitoring functions, in addition to direct access to a controller function of this software system.



	PDTag	IP Address	Device ID
	36 ...	192.168.2.36	NCS4000-CTLR0456BA00
●	NCS4000 Controller 044 ...	192.168.2.44	NCS4000-CTLR-2-44

Start Diagnostic Tools

Users can start NCS4000 system diagnostic tool from the process group, the tool automatically activated to connect NCS4000 I/O server. If the I/O server did not start, diagnostic tools users will be asked whether to activate.

I/O Server Management

In diagnostic tools we can control I/O server, the OPC server operations, including start, stop, state detection.

State Controller Detection

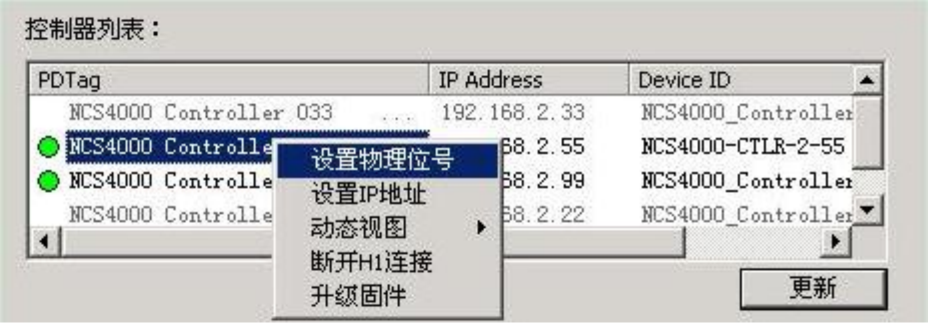
There is List controller information of Diagnostic tools in the "standard" pages. The list shows that all known (that is, once online) controller state information, including the controller that has been off the assembly line. The controller work before normal green icon; Offline or no detected before the controller icons and text information for the controller is Gray; The controller is initializing during no icon.

Shown information including: physical-controller, (PDTag), IP address, device ID (DeviceID).

Once the diagnostic tools and the I/O server connection established, they will automatically update the information.

Controllers Attribute Set

Users can use the on-line diagnostic tool for the physical attributes controller settings.



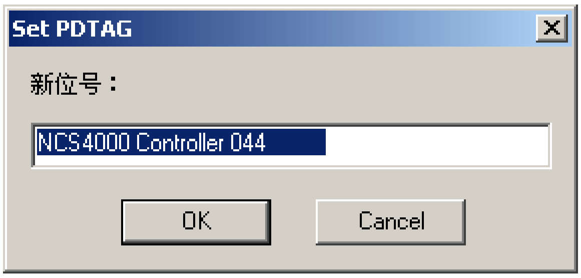
控制器列表：

PDTag	IP Address	Device ID
NCS4000 Controller 033 ...	192.168.2.33	NCS4000_Controller
● NCS4000 Controlle	68.2.55	NCS4000-CTLR-2-55
● NCS4000 Controlle	68.2.99	NCS4000_Controller
NCS4000 Controlle	68.2.22	NCS4000_Controller

4.6.1. Setting Of Physics Number

In the pages of the standard diagnostic tools, to amend the mouse right click on the physics of the controller, in the pop-up menu of choices, the physical set up, set up a physical place of the dialog box. In the dialog box input the new position, to select the button. If the installation is successful, the display controller column list will be updated automatically.

Controller of physics number at the maximum length can not be more than 32 characters.


Set PDTAG

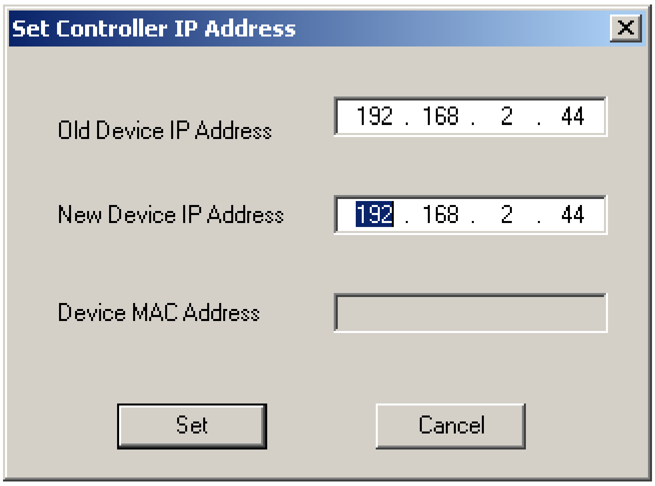
新位号：

NCS4000 Controller 044

4.6.2. Set IP Addresses

In the pages of the standard diagnostic tools, to amend the mouse right click on the IP address of the controller, in the pop-up menu select set of IP addresses, IP addresses controller installed in the dialog box. In the dialog box enter the new IP address, select Settings button.

After the success of installation users are prompted to restart the controller. Controller restart after 10 ~ 20 seconds after automatically on the line, controller list will be updated automatically.


Set Controller IP Address

Old Device IP Address: 192 . 168 . 2 . 44

New Device IP Address: 192 . 168 . 2 . 44

Device MAC Address:

-- Links To Disconnect H1

In the pages of the standard diagnostic tools, mouse right click on the selected controller, in the pop-up menu choose disconnect the link of the H1, the current controller IOServer with selected links between H1 was disconnected.

Attention: IOServer without informing its client, H1 link has been disconnected, if the "disconnect H1 link" after the controller under the H1 device operation, and the controller IOServer H1 link was re-established relations.

System Parameters Set

The standard page of the controller can also filter system parameters such as configuration; click "Configuration ..." button will show a configuration file (ini format).

The document can be assigned parameters include:

Bundled Local IP Addresses

Controller Substituted Filter

5. Appendix

5.1. HART instrument unit specification

Each channel of a HART module contains 4 variables, and each variable is composed of value and unit. The code of HART instrument unit is one byte data, and the unit is divided into several categories, such as temperature, length, capacity, speed, pressure and so on. HART instrumentation unit comparison table is in the reference documentation below Table 2. Unit Codes.
HART® Communication Foundation Document Number: HCF_SPEC-183
HART® - SMART Communications Protocol, Common Tables Document Revision: 9.0

Enumerations 240-249 are reserved for manufacturer specific definitions. Therefore, Host applications should not display meaningful information for these enumerations unless the Host has the Device Description for that specific Field Device. Some Host applications have used a string of the character ‘?’ for this application (e.g. Pressure:10.5????)
Any enumerations in the range from 250-255 are ‘Reserved’ for a single definition if not defined. Any enumeration not covered by another "Design Note" is ‘Undefined’.

HART instrumentation units comparison table

Category	Code	Unit	Description
Temperature	32	degC	degrees celsius
	33	degF	degrees fahrenheit
	34	degR	degrees rankine
	35	Kelvin	degrees kelvin
Pressure	1	InH2O	inches of water 68 degrees F, Note 1
	2	InHg	inches of mercury 0 degrees C, Note 1
	3	FtH2O	feet of water 68 degrees F, Note 1
	4	mmH2O	millimeters of water 68 degrees F, Note 1
	5	mmHg	millimeters of mercury 0 degrees C, Note 1
	6	psi	pounds per square inch, Note 1
	7	bar	bars, Note 1
	8	mbar	millibars, Note 1
	9	g_SqCm	grams per square centimeter, Note 1
	10	kg_SqCm	kilograms per square centimeter, Note 1
	11	PA	pascals, Note 1
	12	kPA	kilopascals, Note 1

	13	torr	torr, Note 1
	14	ATM	atmospheres, Note 1
	237	mega_pascals	megapascals, Note 1
	238	in_H2O_4_degrees_C	inches of water 4 degrees C, Note 1
	239	mm_H2O_4_degrees_C	millimeters of water 4 degrees C, Note 1
Volumetric Flow	15	CuFt_min	cubic feet per minute
	16	gal_min	gallons per minute
	17	l_min	liters per minute
	18	ImpGal_min	imperial gallons per minute
	19	CuMtr_hr	cubic meter per hour
	22	gal_s	gallons per second
	23	MillionGal_day	million gallons per day
	24	l_s	liters per second
	25	MilL_day	million liters per day
	26	CuFt_s	cubic feet per second
	27	CuFt_day	cubic feet per day
	28	CuMtr_s	cubic meters per second
	29	CuMtr_day	cubic meters per day
	30	ImpGal_hr	imperial gallons per hour
	31	ImpGal_day	imperial gallons per day
	121	normal_cubic_meter_per_hour	MKS System
	122	normal_liter_per_hour	normal liter per hour, MKS System
	123	standard_cubic_feet_per_minute	US System
	130	CuFt_hr	cubic feet per hour
	131	CuMtr_min	cubic meters per minute
	132	bbl_s	barrels per second, 1 barrel = 42 U.S. gallons
	133	bbl_min	barrels per minute, 1 barrel = 42 U.S. gallons
	134	bbl_hr	barrels per hour, 1 barrel = 42 U.S. gallons
	135	bbl_day	barrels per day, 1 barrel = 42 U.S. gallons
	136	gal_hr	gallons per hour
	137	ImpGal_s	imperial gallons per second
	138	l_hr	liters per hour
	235	gallons_per_day	gallons per day
Velocity	20	ft_s	feet per second
	21	mtr_s	meters per second
	114	in_s	inches per second
	115	in_min	inches per minute
	116	ft_min	feet per minute
	120	mtr_hr	meters per hour
Volume	40	gal	gallons
	41	liter	liters
	42	ImpGal	imperial gallons
	43	CuMtr	cubic meters

	46	bbl	Barrels, 1 barrel = 42 U.S. gallons
	110	bush	bushels
	111	CuYd	cubic yards
	112	CuFt	cubic feet
	113	CuIn	cubic inches
	124	bbl_liq	bbl liq, 1 liquid barrel = 31.5 U.S. gallons. Note that the abbreviation is not from a reference text
	166	normal_cubic_meter	normal cubic meter, MKS System
	167	normal_liter	normal liter, MKS System
	168	standard_cubic_feet	standard cubic feet, US System
	236	hecto_liter	hectoliters
Length	44	ft	feet
	45	meter	meters
	47	in	inches
	48	cm	centimeters
	49	mm	millimeters
Time	50	min	minutes
	51	sec	seconds
	52	hr	hours
	53	day	days
Mass	60	gram	grams
	61	kg	kilograms
	62	MetTon	metric tons
	63	lb	pounds
	64	ShTon	short tons
	65	LTon	long tons
	125	ounce	ounce
Mass Flow	70	g_s	grams per second
	71	g_min	grams per minute
	72	g_hr	grams per hour
	73	kg_s	kilograms per second
	74	kg_min	kilograms per minute
	75	kg_hr	kilograms per hour
	76	kg_day	kilograms per day
	77	MetTon_min	metric tons per minute
	78	MetTon_hr	metric tons per hour
	79	MetTon_day	metric tons per day
	80	lb_s	pounds per second
	81	lb_min	pounds per minute
	82	lb_hr	pounds per hour
	83	lb_day	pounds per day
	84	ShTon_min	short tons per minute
	85	ShTon_hr	short tons per hour

	86	ShTon_day	short tons per day
	87	LTon_hr	long tons per hour
	88	LTon_day	long tons per day
Mass per Volume	90	SGU	specific gravity units
	91	g_CuCm	grams per cubic centimeter
	92	kg_CuMtr	kilograms per cubic meter
	93	lb_gal	pounds per gallon
	94	lb_CuFt	pounds per cubic foot
	95	g_ml	grams per milliliter
	96	kg_l	kilograms per liter
	97	g_l	grams per liter
	98	lb_CuIn	pounds per cubic inch
	99	ShTon_CuYd	short tons per cubic yard
	100	degTwad	degrees twaddell
	102	degBaum_hv	degrees baume heavy
	103	degBaum_lt	degrees baume light
	104	degAPI	degrees API
Viscosity	54	centi_stokes	centistokes
	55	cpoise	Centipoise,Electromagnetic unit of Electric Potential
	36	mV	millivolts
	58	v	Volts, Electrostatic unit of Current
	39	mA	Milliamperes, Electromagnetic unit of Resistance
	37	Ohm	ohms
	163	kOhm	kohms
Energy (includes Work)	69	newton_meter	newton meter
	89	deka_therm	deka therm
	126	foot_pound_force	foot pound force
	128	kilo_watt_hour	kilo watt hour
	164	mega_joule	mega joule
	165	british_thermal_unit	british thermal unit, (1Btu=0.2519958kcal) Energy
	162	mega_calorie	mega calorie, 1 calorie = 4.184 Joules
Power	127	kilo_watt	kilo watt
	129	horsepower	horsepower
	140	mega_calorie_per_hour	mega calorie per hour, 1 calorie = 4.184 Joules
	141	mega_joule_per_hour	mega joule per hour
	142	british_thermal_unit_per_hour	british thermal unit per hour, 1Btu=0.2519958kcal
Radial Velocity	117	deg_s	degrees per second
	118	rev_s	revolutions per second
	119	rpm	revolutions per minute
Miscellaneous	38	Hz	hertz
	56	uMho	microsiemens
	57	Percent	percent

	59	pH	pH
	66	mSiemen_cm	milli siemens per centimeter
	67	uSiemen_cm	micro siemens per centimeter
	68	newton	newton
	101	degBrix	degrees brix
	105	Percent_sol_wt	percent solids per weight
	106	Percent_sol_vol	percent solids per volume
	107	degBall	degrees balling
	108	proof_vol	proof per volume
	109	proof_mass	proof per mass
	139	parts_per_million	parts per million
	143	degrees	degrees
	150	percent_StmQual	percent steam quality
	151	Ftin16	Note 151
	152	CuFt_lb	cubic feet per pound
	153	pico_farads	picofarads
	160	Percent_plato	percent plato
Generic	240 ～ 249	????	Enumeration may be used for manufacturer specific definitions
Reserved	250	not_used	
	251	none	
	252	unknown	
	253	special	

Note 1: The reference pressure is ambiguous with respect to absolute or gage.

Note 151: There must be 6 digits to the left of the decimal point of the associated numeric value. The formats of the most significant two of these digits indicate the number of feet. The adjacent two lesser significant digits are indicate the number of additional sixteenths (i.e. 16 sixteenths = 1 inch). If the numeric value is in the floating point format, any digits to the right of the decimal point are discarded by the host.

5.2. Common questions and answers

5.2.1. Controller redundancy failure

1. During inserting controller, it should be better inserted straight down, otherwise it could cause a controller can power but not communications. In this case, multi mutually redundant controllers are all likely to be the master device at the same time.

2. During switching, it should ensure that the slave controller obtain the configuration

information from master device before manual switch.

3. See **controller redundancy switching** specification

5.2.2. Controller IP address/PDTAG/DeviceID conflict

In use controller IP address/PDTAG/DeviceID conflict situation may be encountered, such conflicts can be very clear to see in the resource manager and I/O Sever diagnostic tool.

The most common conflict situation is that the two mutually redundant controllers are respectively plugged into two different control stations without any disposal; in addition, plugging in a new controller with the same IP address of an existing device will also cause conflict. When conflicts occur, neither of the controllers in conflict can normal work. Users should resolve the conflict before access the controllers.

Method of conflict resolution:

1. Use resource manager or I/O server diagnostic tool to determine the conflict device's DeviceID.

2. View the controller's shell and find out the conflict controller according to the DeviceID on the shell. Disconnect all the conflict controllers and leave only one of them connecting on the network. If there is DeviceID conflict, first check shell, as indeed DeviceID conflict, please contact suppliers.

3. Check the I/O server diagnostic tool and confirm that there is no conflict device. Then set IP address or PDTAG for the current devices, and restart controller. After that observe the I/O Server diagnostic tool and check if devices have new IP address or PDTAG on the line.

４． Repeat Step 3 until all conflicts resolved.

5.2.3. Can not modify the controller's IP address

During configuring controller IP address, if the controller IP address is going to be modified to another network segment, sometimes it prompted to set up successfully, but it does not really succeed. This is due to that there are multi network cards in the host. The package used to setup IP address is sent from other network interface but not from the network interface connecting to the controller, and it results in the controller fail to receive the package. Solutions are:

1. Use link controllers with only one network card.

2. Set IP address in the range of network segment for the network card connecting to controller.

5.2.4. The H1 devices can not go online

In FF protocol, the transmission operation of each device is strictly scheduled, a device on the bus can only perform transmission when it holds the token, and thus the collision can be avoided.

In the current protocol, however, there is still one case may lead to collisions, that is when the LAS finishes the transmission of PN, if there are two devices on the bus have the same address, and receive the PN at the same time, and have the same execution time, then the two device will send PR at the same time, which lead to collision. Under this condition, the collision results in that the PR can not be correctly received, and the LAS can not recognize the devices, thus the devices can not join the network.

When one or more device can not join the network, perform the following operations:

- Let the H1 device join the network serially, that is before the current device join the network, the other devices should not be connected on the network.
- Perform mapping on the devices that has temporary address, modify the address, let the temporary address unoccupied, and then connect other devices on the network.
- if the reason that lead to the H1 device can not join the network is the H1 device can not initialize successfully, just reconnect this device, and perform re-initialization.

Note: The total number of H1 devices in one physical subnet should not exceed 16, the suggestion value is 8.

5.2.5. Failure to set the address of an H1 device

1. Check if the H1 device is disconnected from the network.
2. Modify the T1 and T3 timer of the entire interface devices belong to the H1 device, then reconfigure the address.

5.2.6. Illustration of the time configuration related to the device address setting:

When performing address configuration (the bit number configuration also include the address configuration), the T1, T2, T3 timer in the SMIB should be correctly set to guarantee the successful address configuration, among which, the T1 and T3 timer is needed by the master device (the device used to perform address configuration, usually is the interface device in the subnet), the T2 timer is needed by the slave device （the devices whose addresses are configured, usually are the instruments and I/O devices）. These timers should be set before the address configuration.

The three timers are unsigned 32 bit integer, which denote time unit of 1/32 ms. The configuration of the three timers is related to the current network parameter, the devices number, and the network load. In order to guarantee the valid timer configuration, when performing the configuration, there should be no other packets transmission on the network.

T1: the value of T1 should be set according to the network parameters and the devices number on the network, and it should be larger than the total transmission time that a system management service request packet and the corresponding response packet needed. Its default value is 160000 that are 5 seconds.

T3: This timer is used to guarantee that a device could join the network using its new address before it is timeout, it value is related to the network parameters and the devices number on the network, the parameters needed to be considered is:

SlotTime (ST), MaxResponseDelay (MRD), MinInternalDelay (MID), FirstUnpolledAddr.

The maximum time a device needed to join the network is:
(FirstUnpolledAddr - 0x10 – NumberOfDevice +4) * ((15 + MRD * MID) * NumberOfDevice + 30 + 3 * MID + MRD * ST)
The time unit is 1/4ms.

To avoid the troublesome computing, the T3 can be configured with a maximum value to guarantee the success of the address configuration. The probability that the new device could join the network successfully increases with T3 value, but it should be noted that the time needed by the address configuration also increases with the T3 value, so a better solution to set the T3 value is to add an excess (the suggestion value is 2 seconds) to the result computed by the above formula.

T2: Its value should be larger than the sum of T1 and T3.

5.2.7. The I/O server can not obtain the list of field controllers

1. Check is the host computer is at the same subnet as the controller;

2. Check the firewall configuration of the host computer, and make sure that the NCSIOServer can access the network correctly.

3. Check the filtering configuration of NCSIOServer, make sure that the local IP address is not filtered by NCSIOServer. For further information, please refer to local IP address binding.

5.2.8. The data of online monitoring is incorrect

When performing on-line monitoring of the data generated by the control policy module using the control policy development environment, the situation that all the data are incorrect may occur. The reason is the inconsistence of the downloaded data in the controller and the data stored in the system information database, and the inconsistence comes from the fault that the database is changed, but the control policy in the controller is not updated according to the new database.

When the problem occurs, there are two solutions:
1. Re-download the data from the new database;
2. Restore the database which is consistent with the data in the controller.

5.2.9. The OPC data is incorrect

When using the OPC clients to monitor the whole project through the NCS4000 OPC server, the situation that the data display of the OPC client is inconsistent with the actual control data. The reason is the refresh time of OPC server can not make in time response to the control modules whose execution period is less than 1s. To solve the problem, the execution period of the control policy module should be set to a value larger than 1s.

5.2.10. OPC server can not be stopped

When you are downloading the configuration information or when you trying to stop the OPC server using the diagnostic tools of I/O server, if the OPC server can not be stopped, please check the TCP 5001 port of the host computer and see if it is occupied by other program. If this port is occupied, the packets containing the “stop” command can not be delivered to OPC server, which results in the OPC server can not be stopped.

In another case, if there are clients continuously connecting the OPC server, the situation that the server be launched immediately after it was stopped may occur. In this situation, the user should close all the clients programs, and then stop the OPC server.

5.2.11. Abrupt output jump occurs when hot-swappable the AO modules

When a I/O module is hot-swappable, the output will jump to its maximum value immediately, the reason is when the module is inserted, the 24V DC power is conducted first, which lead to the DA in the AO module starts working, and after it output a maximum value, the 5v module is conducted, then the AO module resumes to the normal state, and the output resumes also. For the sake of safety, we suggest that the user should pull out the 24V power first, and then hot-swappable the AO module.

5.2.12. The operation time (such as download) is too long

When performing the control policy configuration or downloading, the operation time may last for a very long time. The reason is some program exit abnormally. When this situation encountered, it is strongly recommend that the user restart the I/O server and the database.

5.3. NCS4000 controller

5.3.1. Introduction

The NCS4000 controller is also called the control station, is the device which actually carry out the control policy. The controller has Ethernet NICs, can use the TCP/IP protocol to communicate with other controllers within the same subnet, or communicate with the work station(computer) in the network.

The I/O modules, i.e. I/O communication cards, can be installed into the controller, each controller can hold 32 I/O modules.

At current time, the NCS4000 controller can support the following I/O modules:

- 8-channel 4-20mA current input module (NCS4000-AI8-I)
- 16-channel current input module (NCS4000-AI16-I)
- 8-channel voltage input module (NCS4000-AI8-V)
- 16-channel voltage input module (NCS4000-AI16-V)
- 8-channel HART input module (NCS4000-AI8-HART)
- 8-channel pulse input module (NCS4000-PI8)
- 8-channel hot couple input module
- 4-channel hot resistor input module (NCS4000-RTD4)
- 8-channel hot resistor input module (NCS4000-RTD8)
- 8-channel 4-20mA current output module (NCS4000-AO8)
- 8-channel 24V DC digital input module (NCS4000-DI8-24VDC)
- 16-channel 24V DC input module (NCS4000-DI16-24VDC)
- 16-channel 24V DC dry-node input module (NCS4000-DI16-24VDC-DRY)
- 16-channel 24V DC voltage output module (NCS4000-DO16-24VDC-V)
- 8-channel 220V AC digital input module (NCS4000-DI8-220VAC)
- 16-channel 220V AC digital input module (NCS4000-DI16-220VAC)
- 8-channel 24V DC digital output module (NCS4000-DO8-24VDC)
- 8-channel 220V AC solid-relay output module (NCS4000-DO8-220VAC)
- 2-channel FF fieldbus H1 protocol module (NCS4000-FF-H1)

5.3.2. The Performance of NCS4000 Controller

I/O Related Indexes

Metrics	Value
the maximum number of I/O modules	32
the maximum channels number of each I/O module	16
The I/O data refresh time	

Communication Related Indexes

Metrics	Value
The maximum number of VCR (the communication relations between controllers, or between controller and workstation)	125
The maximum link objects number （the connection between function blocks）	8192
FBD Input or Output parameter number	255

Control Policy Execution Indexes

Metrics	Value
The minimum scan period of modules (application process)	100 ms
The maximum self-defined parameters number of ladder diagram module	1024
The maximum self-defined parameters number of function block diagram module	N/A
The types number of FBD function block	37
The maximum function blocks number of every controller	3000 (the period of each module is 5s)
The maximum modules (application process) number	255

Time Related Indexes

Device start time	5.20s (without static data) 5.34s (with static data)*
The execution time of function blocks (in AUTO mode)	See the appendix
The maximum offset of time synchronization	5Kb
The switch time of controllers communication redundancy	2s
The output switching time of the input controller redundancy	1s
CPU load	

* Related to the actual configuration policy

Appendix: The Execution Time of Function Blocks (in AUTO mode, unit: ms)

Function block type	Execution time
AI	0.09
AO	0.09
DI	0.05
DO	0.05
MAI8	0.05
MAO8	0.13
MDI	0.05
MDO	0.09
PID	0.31
RA(RTO)	0.15
DT	0.11
IS	0.06
ISE	0.07
CS	0.06
SS	0.16
LL	0.06
SPLT	0.09
AR	0.09
CNT	0.015
ABS	0.006
ADD	0.008
CMP	0.018
DIV	0.011
MUL	0.009
SUB	0.008
AND	0.016
BDE	0.008
BFI	0.14

BFO	0.06
MULT	0.03
NDE	0.007
NOT	0.006
OR	0.016
PDE	0.007
RS	0.009
SR	0.009

5.3.3. The Controller Redundancy

Introduction

The initialization process of the main controller includes the static data restoration and the initialization of other tasks. When the slave controller starts, it need to send request packets to the master controller to ask for the static data, this results in the slave controller needs more time to finish the initialization process. During the initialization process, the third LED (the loading LED) blinks, and this LED stops blinking when the initialization process finishes. The LED's status (ON or OFF) is consistent with the master controller.

Switching Operations

If the master controller fails at the runtime, the switching operations are performed. If the master controller needs to be substituted, it can be removed from the network. The redundancy switching should be avoided when the slave controller is in its initialization stage, because if the switching occurred at this time, the slave controller will change into the master controller immediately, but it can only run normally after the wait timeout (1 minute at the largest), and the configuration information downloaded at last time will be totally lost.

Download Operation

The download operation should not be performed when the slave controller is initializing, because during the initialization stage, the data from the workstation can not be received by the slave controller, which results in the inconsistence between the master controller and the slave controller. If the switching is occurred at this time, the correctness can not be guaranteed.

When the master controller is downloading (the downloading LED is blinking), the slave controller can be plugged into the network, but it will not request static data from the master controller until the latter finishes downloading. That is, the initialization time of the slave controller is enlarged, but the data consistence is guaranteed.

At the downloading period, the master controller should not be removed from the network.

The following operation process is recommended:

- If the master controller and the slave controller starts at the same time, the downloading operation should not be performed until all the LEDs' status of the master controller and the slave controller is the same (except the master/slave identification LED).
- If the slave controller is started after the master controller, it should not be plugged into the network until the downloading LED blinks. It is suggested that the slave controller should be plugged into the network after the master controller totally finishes its initialization.

The writing operations

The writing operations include operations such as configuring parameters of function blocks. The writing operations should not be performed until all the LEDs' status of the master controller and the slave controller is the same (except the master/slave identification LED)

5.4. The ladder diagram (LD) language reference

The standard operators, functions, and function blocks in LD

The standard operators, functions, and function blocks in the LD diagram configuration is the program organization unit which is compatible with the IEC61131-3 international standard.

Standard Operators and Functions

With many input parameters and one output parameter, the same input always produce the same output.

Standard Function Blocks

With many input and output parameters and internal memory, the output value is determined by the internal value.

Data Types

Types	Valid scope	Illustration	Overflow processing
Boolean	0,1	boolean, single byte	Non-zero to 1
Int8	-128-127	short, single byte	Turn into the corresponding value in the valid scope
Int32	-2147483648-2147483647	long, four bytes	Turn into the corresponding value

			in the valid scope
Float	positive: (1E-35, 1E+35) negative: (-1E+35, -1E-35)	Float, four bytes, 7 valid numbers	The result is undeterminable
Time	0-86399999	Time, four bytes	Turn into the corresponding value in the valid scope

The valid scope of the float type is: positive (3.4E-38, 3.4E+38), negative (-3.4E-38, -3.4E+38).

Auxiliary Parameters EN and ENO

The auxiliary parameters EN and ENO is only used to integrate the operators, functions, and function blocks into the LD diagram networks, pass over the “energy stream”, and determine the control stream.

EN	ENO	illustration
0	0	do not execute the related functions
1	1	execute the related functions

5.4.1. Operators

Standard Operators in LD

ADD_S、ADD_D、ADD_R、ADD_T：Sum, OUT=IN1+IN2

Parameters	Data types				Description
	ADD_S	ADD_D	ADD_R	ADD_T	
IN1	short	long	float	time	addend 1
IN2	short	long	float	time	addend 2
OUT	short	long	float	time	sum

SUB_S、SUB_D、SUB_R、SUB_T：subtraction. OUT=IN1-IN2

Parameters	Data types				Description
	SUB_S	SUB_D	SUB_R	SUB_T	
IN1	short	long	float	time	minuend
IN2	short	long	float	time	subtrahend
OUT	short	long	float	time	difference

MUL_S、MUL_D、MUL_R：multiplication。OUT=IN1×IN2。

Parameters	Data types			Description
	MUL_S	MUL_D	MUL_R	
IN1	short	long	float	multiplicator 1
IN2	short	long	float	multiplicator 2
OUT	short	long	float	product

DIV_S、DIV_D、DIV_R：division。OUT=IN1÷IN2。If IN2=0，the OUT is unchanged.

Parameters	Data types	Description

	DIV_S	DIV_D	DIV_R	
IN1	short	long	float	dividend
IN2	short	long	float	divisor
OUT	short	long	float	quotient

MOVE_B、MOVE_S、MOVE_D、MOVE_R、MOVE_T：set value. OUT=IN。

Parameters	Data types					Description
	MOVE_B	MOVE_S	MOVE_D	MOVE_R	MOVE_T	
IN	boolean	short	long	float	time	input value
OUT	boolean	short	long	float	time	output value

AND、AND_MASK：and operation. OUT=IN1 AND IN2。

Parameters	Data types		Description
	AND	AND_MASK	
IN1	boolean	short	input value 1
IN2	boolean	short	input value 2
OUT	boolean	short	output value

OR、OR_MASK：or operation. OUT=IN1 OR IN2。

Parameters	Data types		Description
	OR	OR_MASK	
IN1	boolean	long	input value 1
IN2	boolean	long	input value 2
OUT	boolean	long	output value

XOR、XOR_MASK：xor operation. OUT=IN1 XOR IN2。

Parameters	Data types		Description
	XOR	XOR_MASK	
IN1	boolean	long	input value 1
IN2	boolean	long	input value 2
OUT	boolean	long	output value

NOT、NOT_MASK：not operation. OUT=NOT IN。

Parameters	Data types		Description
	NOT	NOT_MASK	
IN	boolean	long	input value
OUT	boolean	long	output value

CNV_BS、CNV_BD、CNV_BR、CNV_BT：type conversion（boolean） OUT=IN。

Parameters	Data types				Description
	CNV_BS	CNV_BD	CNV_BR	CNV_BT	
IN	boolean	boolean	boolean	boolean	input value

OUT	short	long	float point	time	output value

CNV_SB、CNV_SD、CNV_SR、CNV_ST：type conversion（short）。OUT=IN。

Parameters	Data types				Description
	CNV_SB	CNV_SD	CNV_SR	CNV_ST	
IN	short	short	short	short	input value
OUT	boolean	long	float point	time	output value

CNV_DB、CNV_DS、CNV_DR、CNV_DT：type conversion（long） OUT=IN。

Parameters	Data types				Description
	CNV_DB	CNV_DS	CNV_DR	CNV_DT	
IN	long	long	long	long	input value
OUT	boolean	long	float point	time	output value

CNV_RB、CNV_RS、CNV_RD、CNV_RT：type conversion（float point）OUT=IN

Parameters	Data types				Description
	CNV_RB	CNV_RS	CNV_RD	CNV_RT	
IN	float point	float point	float point	float point	input value
OUT	boolean	short	long	time	output value

CNV_TB、CNV_TS、CNV_TD、CNV_TR：type conversion（time）. OUT=IN。

Parameters	Data types				Description
	CNV_TB	CNV_TS	CNV_TD	CNV_TR	
IN	time	time	time	time	input value
OUT	boolean	short	long	float point	output value

LT_S、LT_D、LT_R、LT_T：less than. OUT=IN1<IN2。

Parameters	Data types				Description
	LT_S	LT_D	LT_R	LT_T	
IN1	short	long	float point	time	input value 1
IN2	short	long	float point	time	input value 2
OUT	boolean	boolean	boolean	boolean	output value

GT_S、GT_D、GT_R、GT_T：bigger than. OUT=IN1>IN2。

Parameters	Data types				Description
	GT_S	GT_D	GT_R	GT_T	
IN1	short	long	float point	time	input value 1
IN2	short	long	float point	time	input value 2
OUT	boolean	boolean	boolean	boolean	output value

LE_S、LE_D、LE_R、LE_T：less equal. OUT=IN1≤IN2。

Parameters	Data types				Description
	LE_S	LE_D	LE_R	LE_T	
IN1	short	long	float point	time	input value 1
IN2	short	long	float point	time	input value 2
OUT	boolean	boolean	boolean	boolean	output value

GE_S、GE_D、GE_R、GE_T：bigger equal. OUT=IN1≥IN2。

Parameters	Data types				Description
	GE_S	GE_D	GE_R	GE_T	
IN1	short	long	float point	time	input value 1
IN2	short	long	float point	time	input value 2
OUT	boolean	boolean	boolean	boolean	output value

EQ_B、EQ_S、EQ_D、EQ_R、EQ_T：all equal. OUT=IN1≡IN2。

Parameters	Data types					Description
	EQ_B	EQ_S	EQ_D	EQ_R	EQ_T	
IN1	boolean	short	long	float point	time	input value 1
IN2	boolean	short	long	float point	time	input value 2
OUT	boolean	boolean	boolean	boolean	boolean	output value

NE_B、NE_S、NE_D、NE_R、NE_T：not equal. OUT=IN1≠IN2。

Parameters	Data types					Description
	NE_B	NE_S	NE_D	NE_R	NE_T	
IN1	boolean	short	long	float point	time	input value 1
IN2	boolean	short	long	float point	time	input value 2
OUT	boolean	boolean	boolean	boolean	boolean	output value

NEG_S、NEG_D、NEG_R：negate. OUT=-IN。

Parameters	Data types			Description
	NEG_S	NEG_D	NEG_R	
IN	short	long	float point	input value
OUT	short	long	float point	output value

5.4.2. Functions

Standard Functions in LD

SIN、COS、TAN、ASIN、ACOS、ATAN。Q=Fx（IN）。

Parameters	Data types	Description
IN	float point	input value
Q	float point	output value

Function	Valid Scope	Exception Handling
SIN	0 ～ 6.283185	the IN value exceed valid scope，the result is meaningless。
COS	0 ～ 6.283185	the IN value exceed valid scope，the result is meaningless。
TAN	0 ～ 6.283185	the IN value exceed valid scope，the result is meaningless。 if COS（IN）＜1E-35 and COS（IN）＞-1E-35，Q＝1E35。
ASIN	-1 ～ 1	the IN value exceed valid scope，Q＝0。
ACOS	-1 ～ 1	the IN value exceed valid scope，Q＝0。

ABS、SQRT、LOG、TRUNC：absolute value、square、logarithm、rounding. Q＝Fx（IN）。

Parameters	Data types	Description
IN	float point	input value
Q	float point	output value

function	Exception Handling
SQRT	IN≤0，Q＝0。
LOG	IN≤0，Q＝-1E35。

EXPT、POW：power. Q＝Fx（IN，EXP）。

Parameters	Data types		Description
	EXPT	POW	
IN	float point	float point	input
EXP	long	float point	exponent input
Q	float point	float point	output value

MAX、MIN：maximum value 、minimum value。Q＝Fx（IN1，IN2）。

Parameters	Data types	Description
IN1	long	input value 1
IN2	long	input value 2
Q	long	output value

LIMIT：limited value . if IN≤MIN，Q＝MIN；else if IN≥MAX，Q＝MAX；else Q＝IN。

Parameters	Data types	Description
MIN	long	minimum value
IN	long	input value
MAX	long	maximum value
Q	long	output value

MOD：residual. if BASE≤0，Q＝-1；else Q＝IN－IN÷BASE×BASE。

Parameters	Data types	Description
IN	long	input value
BASE	long	base value

Q	long	output value

RAND：random number. if IN≤0，Q＝0；else，Q is a random number between 0 and IN。

Parameters	Data types	Description
IN	long	input value
Q	long	output value

ODD：parity. if IN is odd, Q＝1；else，Q＝0。

Parameters	Data types	Description
IN	long	input value
Q	boolean	output value

ROL、ROR：circular left shift、circular right shift。if NBR≤0，Q＝IN；else，IN left（or right）shift NBR bits circularly , and let Q=IN。

Parameters	Data types	Description
IN	long	input value
NBR	long	the bits number to shift circularly
Q	long	output value

SHL、SHR：left shift、right shift。if NBS≤0，Q＝IN；else，IN left（or right）shift (NBS mod 32)，and the right（or left）is stuffed with zero，then let Q=IN。

Parameters	Data types	Description
IN	long	input value
NBS	long	the bits number to shift
Q	long	output value

SEL、MUX4、MUX8：two-value selection、four-value selection、eight-value selection。MUX4：the valid scope of SEL is 0～3；MUX8：the valid scope of SEL is 0～7；if the value of SEL is invalid，Q＝0。

Parameters	Data types			Description
	SEL	MUX4	MUX8	
SEL	boolean	long	long	select input
IN1	long	long	long	input value 1
IN2	long	long	long	input value 2
IN3	—	long	long	input value 3
IN4	—	long	long	input value 4
IN5	—	—	long	input value 5
IN6	—	—	long	input value 6
IN7	—	—	long	input value 7
IN8	—	—	long	input value 8
Q	long	long	long	output value

5.4.3. Standard Function Blocks

Standard Function Blocks in LD

SR：double stable（set operation first）.If SET=1，Q=1；if RESET=1，Q=0；if SET and RESET both equal to 1，Q=1.

Parameter	Data Type	Description
SET	boolean	set input
RESET	boolean	reset input
Q	boolean	output result

RS：double stable（reset operation first）.If SET=1，Q=1；if RESET=1，Q=0；if SET and RESET both equal to 1，Q=0.

Parameter	Data Type	Description
SET	boolean	set input
RESET	boolean	reset input
Q	boolean	output result

R_TRIG：edge dection（rising edge）.If CLK detect the rising edge，Q=1；else，Q=0.

Parameter	Data Type	Description
CLK	boolean	input detection
Q	boolean	output result

F_TRIG：edge dection（falling edge）.If CLK detect the falling edge，Q=1；else，Q=0.

Parameter	Data Type	Description
CLK	boolean	input detection
Q	boolean	output result

TON：Start the delay timer. When IN detects the rising edge, the timer is started；if IN=1，when the timer value ≥ PT，Q=1，and the timer is stopped；if IN＝0，Q=0，the timer is reset and stopped。Note：in the non-executing state（EN＝0），if the timer is started, the time counting persists.

Parameter	Data Type	Description
IN	boolean	input switch

PT	time	the delay time assigned
Q	boolean	output switch
ET	time	time counter

TOF：Stop the delay timer. When IN detects the falling edge, the timer is started; if IN=0, when the timer value ≥ PT，Q=0, and the timer is stopped; if IN=1, Q=1, and the timer is reset and stopped. Note: in the non-executing state（EN=0），if the timer is started, the time counting persists.

Parameter	Data Type	Description
IN	boolean	input switch
PT	time	the delay time assigned
Q	boolean	output switch
ET	time	time counter

TP：pulse timer.When IN detects the rising edge, the timer is started, and Q=1;if Q=1, when the timer value ≥ PT，Q=0，and the timer is stopped；if IN=0 and Q=0，the timer is reset and stopped。in the non-executing state（EN＝0），if the timer is started, the time counting persists.

Parameter	Data Type	Description
IN	boolean	input switch
PT	time	the delay time assigned
Q	boolean	output switch
ET	time	time counter

BLINK：blink timer.When RUN detects the rising edge，the timer is started；if RUN=1，when the timer value ≥ CYCLE/2，Q is negated and output，and the timer is reset；if RUN=0，Q=0，and the timer is stopped. Note: in the non-executing state（EN=0），if the timer is started, the time counting persists.

Parameter	Data Type	Description
RUN	boolean	input switch
CYCLE	time	assigned cycle period
Q	boolean	output switch

CTU：counter（plus）.If RESET=1，CV=0；else if CU=1 and CV < PV，CV increase 1.If CV ≥ PV，Q=1，else Q=0.

Parameter	Data Type	Description
CU	boolean	input switch
RESET	boolean	reset switch
PV	long	assigned counter
Q	boolean	output switch

CV	long	current counter

CTD：counter（minus）.If LOAD=1，CV=PV；else if CD=1 and CV>0，CV decrease 1. If CV≤0，Q=1；else Q=0.

Parameter	Data Type	Description
CD	boolean	input switch
LOAD	boolean	load switch
PV	long	assigned counter
Q	boolean	output switch
CV	long	current counter

CTUD：counter（plus/minus）。If RESET=1，CV=0；else LOAD=1，CV=PV；else if CU=1and CV<PV，CV increase 1；else if CD=1 and CV>0，CV decrease 1. If CV≥PV，QU=1，else QU=0. If CV≤0，QD=1；else QD=0.

Parameter	Data Type	Description
CU	boolean	input（plus）switch
CD	boolean	input（minus）switch
RESET	boolean	reset switch
LOAD	boolean	load switch
PV	long	assigned counter
QU	boolean	output（plus）switch
QD	boolean	output（minus）switch
CV	long	current counter

CMP：comparer。If VAL1<VAL2，LT=1；if VAL1≡VAL2，EQ=1；if VAL1>VAL2，GT=1。

Parameter	Data Type	Description
VAL1	long	input value 1
VAL2	long	input value 2
LT	boolean	less flag
EQ	boolean	equal flag
GT	boolean	big flag

SCALE：measure scope conversion。If |IN_EU100 - IN_EU0|<1E-7，OUT=0；else，OUT=（IN-IN_EU0）*（OUT_EU100 - OUT_EU0）/（IN_EU100 - IN_EU0）+OUT_EU0. If OUT>OUT_EU100，OUT=OUT_EU100；if OUT<OUT_EU0，OUT=OUT_EU0。

Parameter	Data Type	Description
IN	float	input value
IN_EU0	float	input low limit

IN_EU100	float	input high limit
OUT_EU0	float	output low limit
OUT_EU100	float	output high limit
OUT	float	output value

AVERAGE：average value。If RUN=0，OUT=XIN；the modification of N is valid，if N<1，average times is 1；if N>128，average timers is 128。If RUN=1,the average operation is valid，that is OUT equals the average value of last N XIN; When the computation is started, the default initialization value is averaged N according to XIN.

Parameter	Data Type	Description
RUN	boolean	enable average value
XIN	float point	input value
N	long	average times
OUT	float point	output value

LIM_ALRM：warning limit。If QH=1and X<（H-EPS），QH=0；if QH=0 and X>H，QH=1；if QL=1and X>（L+EPS），QL=0；if QL=0 and X<L，QL=1；if QH or QL is 1，Q=1. Note：when H≤L，there is no special processing.

Parameter	Data Type	Description
H	float point	high limit value
X	float point	input value
L	float point	low limit value
EPS	float point	dead scope
QH	boolean	high limit warning
Q	boolean	warning
QL	boolean	low limit warning

INTEGRAL：integral. If R1=1，XOUT=X0；else，Q=1，if RUN=1，XOUT equals the integral of XIN.

Parameter	Data Type	Description
RUN	boolean	enable integral
R1	boolean	enable reset
XIN	float point	integral input value
X0	float point	reset input value
CYCLE	time	integral time
Q	boolean	integral flag
XOUT	float point	integral output

DERIVATE：Difference. If RUN=1，XOUT equals the differential of XIN.

Parameter	Data Type	Description
RUN	boolean	enable difference
XIN	float point	difference input value
CYCLE	interger	difference time
XOUT	float point	difference output value

PID：PID function blocks.

Parameter	Data Type	Description
INPUT	float	input value
SETPOINT	float	set value
GAIN	float	proportional coefficient
RESET	float	integrate coefficient
RATE	float	difference coefficient
PV_EU0	float	zero measure scope of procedure value
PV_EU100	float	full measure scope of procedure value
OUT_EU0	float	zero measure scope of output value
OUT_EU100	float	full measure scope of output value
OUTPUT	float	output value
INTEGRAL	float	integral value
DIFFERENCE	float	differential value
E	float	offset value
PREDIFF	float	last time integral value
PREOUT	float	last time output value

SIG_GEN：The signal generator. When RUN detects the rising edge, the timer is started. If RUN=1, and the timer value ≥ PERIOD, then the timer is reset and the PULSE is negated; if UP<MAXIMUM, UP increases 1, and END=0, or else, UP=0, END=1. If RUN=0, the timer is stopped, and PULSE is reset, UP=0, END=0. SINE=sin(6.283*UP/MAXIMUM). Note: in the non-executing state（EN=0），if the timer is started, the time counting persists.

Parameter	Data Type	Description
RUN	boolean	enable the signal generation
PERIOD	time	cycle period
MAXIMUN	long	the maximum counter
PULSE	boolean	output of the pulse signal
UP	long	current pulse counter
END	boolean	the up-limit flag of pulse counter

SINE	float point	sine signal output

CURR_TIME：current time。Get the current time of the physical controller.

Parameter	Data Type	Description
YEAR	long	year
MONTH	short	month
DAY	short	day
HOUR	short	hour
MINUTE	short	minute
SECOND	short	second

The contact and coil in LD

The contact and coil is the basic unit in the LD diagram configuration. The contact is used to control the value passing in the LD network, and the coil is used to set the computed values to the related variables.

Contact: Execute a logical computation according to the left connection and the status of the related variables, and pass over the computed value to the right connection.

Name	Diagram object	Explanation
always open contact	variables -| |-	If the variable is TRUE，the status of the left connection is copied to the right connection; or else, the FALSE value is copied to the right connection.
always close contact	variables -|/|-	If the variable is FALSE, the status of the left connection is copied to the right connection; or else, the FALSE value is copied to the right connection.
rising-edge contact	variables -|P|-	When the variable changes from FALSE to TRUE, then the status of the left connection is copied to the right connection; or else, the FALSE value is copied to the right connection.
falling-edge contact	variables -|N|-	When the variable changes from TRUE to FALSE, then the status of the left connection is copied to the right connection; or else, the FALSE value is copied to the right connection.

Coil: according to the status value of the left connection, change the status value of the related variables, and always pass over the status value of the left connection to the right connection.

Name	Diagram object	Explanation
coil	variable -( )-	The status of the left connection is copied to the variable

negating coil	variable -(/)-	The status of the left connection is negated and copied to the variable
rising-edge coil	variable -(P)-	When the left connection changes from FALSE to TRUE, the TRUE value is copied to the variable; or else, the variable is unchanged.
falling-edge coil	variable -(N)-	When the left connection changes from TRUE to FALSE, the TRUE value is copied to the variable; or else, the variable is unchanged.
bit-setting coil	variable -(S)-	If the status of the left connection is TRUE, the TRUE value is copied to the variable; or else, the variable is unchanged.
reset coil	variable -(R)-	If the status of the left connection is TRUE, the FALSE value is copied to the variable; or else, the variable is unchanged.