IDENT*Control*
Add-On Instructions for RSLogix 5000

Add-On Instructions for RSLOGIX 5000 and Pepperl+Fuchs IDENTControl

These second generation Add-On Instructions will replace the older style. In the previous series two copy (CPS) instructions were required for each read head or mode used in your ladder logic. These copy instruction though rather easy to setup added confusion. These copy instructions were eliminated for all Ethernet controllers. DeviceNet controllers will still require a copy but only one large copy is required rather then 4 smaller copy instructions like previous generations required. The next generation instructions added the following changes:

- Eliminated the copy instructions
- Unlocked all code
- Checked for new data on all execution counter changes
- Expanded the naming of the instructions for clarity
- Added code carrier instructions
- Rewrite of the read continuous instructions

Available Add-On instructions are:

1. **PF_RFID_SetTagTyp_1HD_1Time** – (Set Tag type) Should be done once initially so the controller knows exactly which tag will be in front of the read head. This setting is stored in non-volatile memory so it isn't necessary to send again unless a controller is replaced
2. **PF_RFID_ReadData_1HD_1Time** – (Single Read Data Carrier) Will read the requested amount of data at the specified start address one time and stop. Should be used any time data needs to be read from a tag one time.

3. **PF_RFID_ReadData_1HD_Cont** – (Enhanced Read Data Carrier) Will read up to 14 DINTs continuously. Stays running forever. Put rung high and leave it on. Use anytime you want to read data on the fly or if you want to read data all the time without sending more commands.
4. **PF_RFID_WriteData_1HD_1Time** – (Single Write Data Carrier) Will write the requested amount of data at the specified start address one time and stop. Should be used any time data needs to be written to a tag one time.
5. **PF_RFID_ClearData_1HD_1Time** – (Fill Data Carrier) Clears the entire tag with a 0x00(or data of choice) to every byte. Make sure tag type is correct so the read head knows how much data to write to the tag
6. **PF_RFID_ReadCode_1HD_1Time** - (Single Read Code Carrier) Reads a single fixed code from a tag and stops. This is the read only identifier that can't be changed on the tag. Tags produce 4 bytes, 5 bytes, 7 bytes or an 8 byte fixed code depending on tag type.
7. **PF_RFID_ReadCode_1HD_Cont** - (Enhanced Read Code Carrier) Reads a single fixed code from a tag and keep running. This is the read only identifier that can't be changed on the tag. Tags produce 4 bytes, 5 bytes, 7 bytes or an 8 byte fixed code depending on tag type.
8. **PF_RFID_WriteCode_1HD_1Time_21** – (Single Write Fixed Code) Write a fixed code to a tag. Typical tag used is the IPC11 type.

Setup

Before using the Add-On Instruction the Ethernet IDENTControl must be setup. A Generic Ethernet Module is used to map the Input and Output data or an EDS file is used to create an EDS-AOP. The EDS-AOP is only available in RSLogix 5000 V20 or higher. The Add-On Instruction will use this directly mapped data to issue commands to the ID system. Different Assembly instances are available depending on if you want to map each head individually and/or you want to map all data together. In general a larger Assembly instance means quicker I/O transfer if you want to read or write a lot of data. Also separated mode is faster then mixed mode if you want to talk to multiple read heads simultaneously.

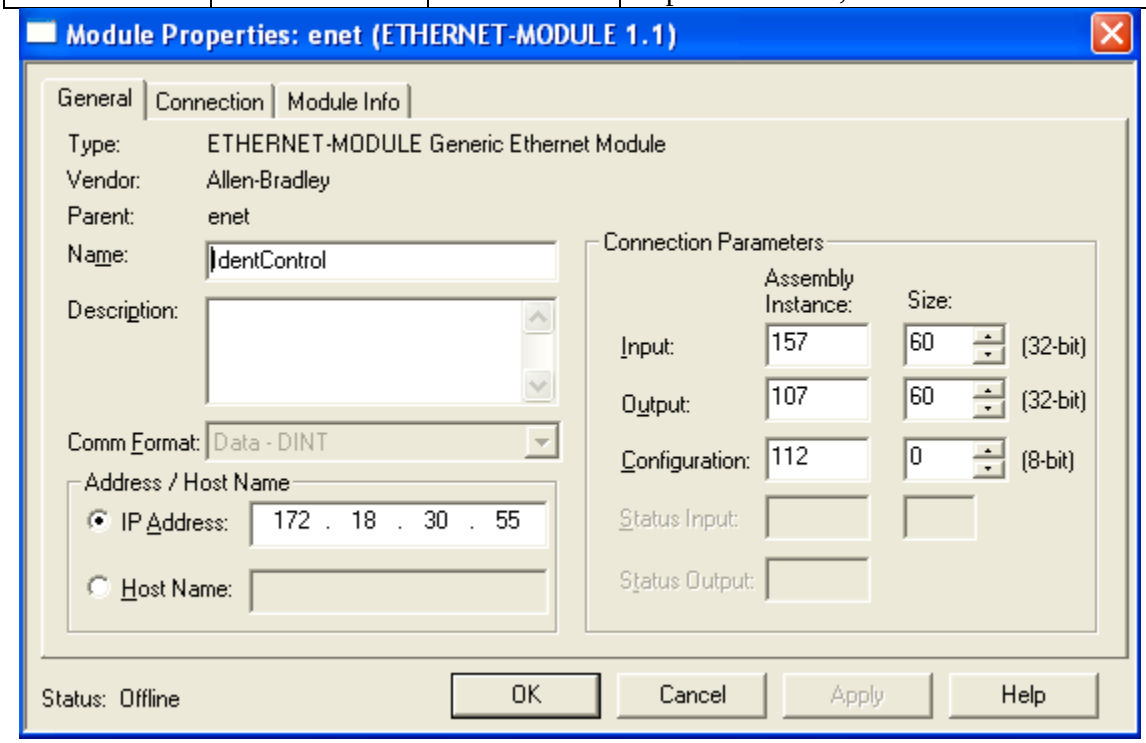
Separated mode means each read head has its own memory map. Commands can be issued simultaneously to all heads. Data mapping is done always for 4 heads even for a two head controller.

In **mixed** mode you must toggle between heads. Only one heads can have a command issued to it at one time.

Always use Comm. Format Data – DINT. You know when this is correctly set because the Data Size now has a (32-bit) next to it. The Configuration is always 112 and the length for the Configuration is always 0.

Assembly Instance Input	Assembly Instance Output	Length (DINTs)	Description

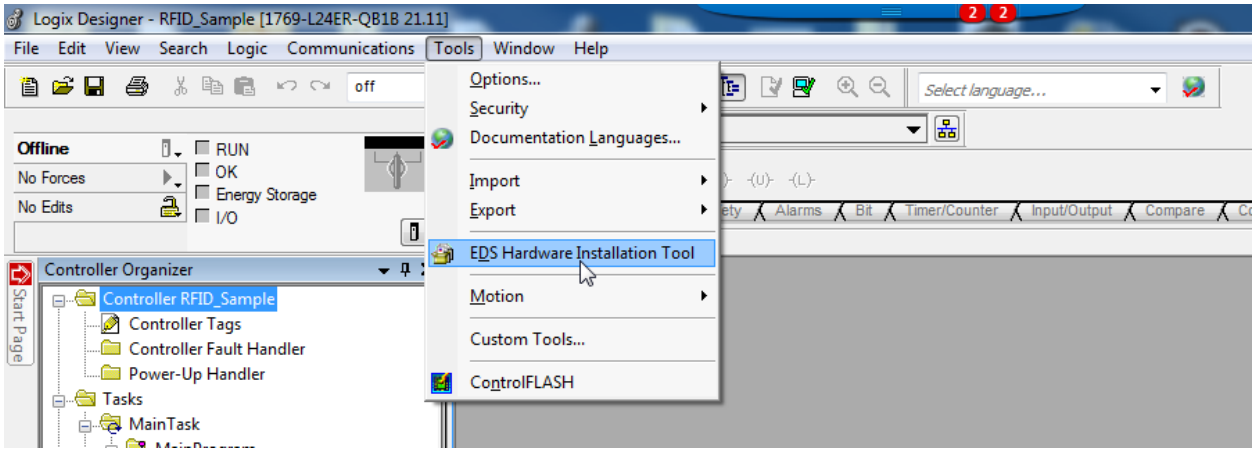
150	100	2	Mixes mode, 1 DINTs at a time
151	101	3	Mixed mode, 2 DINTs at a time
152	102	8	Mixed mode, 7 DINTs at a time
153	103	15	Mixed mode, 14 DINTs at a time
154	104	8	Separated mode, 1 DINTs at a time
155	105	12	Separated mode, 2 DINTs at a time
156	106	32	Separated mode, 7 DINTs at a time
157	107	60	Separated mode, 14 DINTs at a time



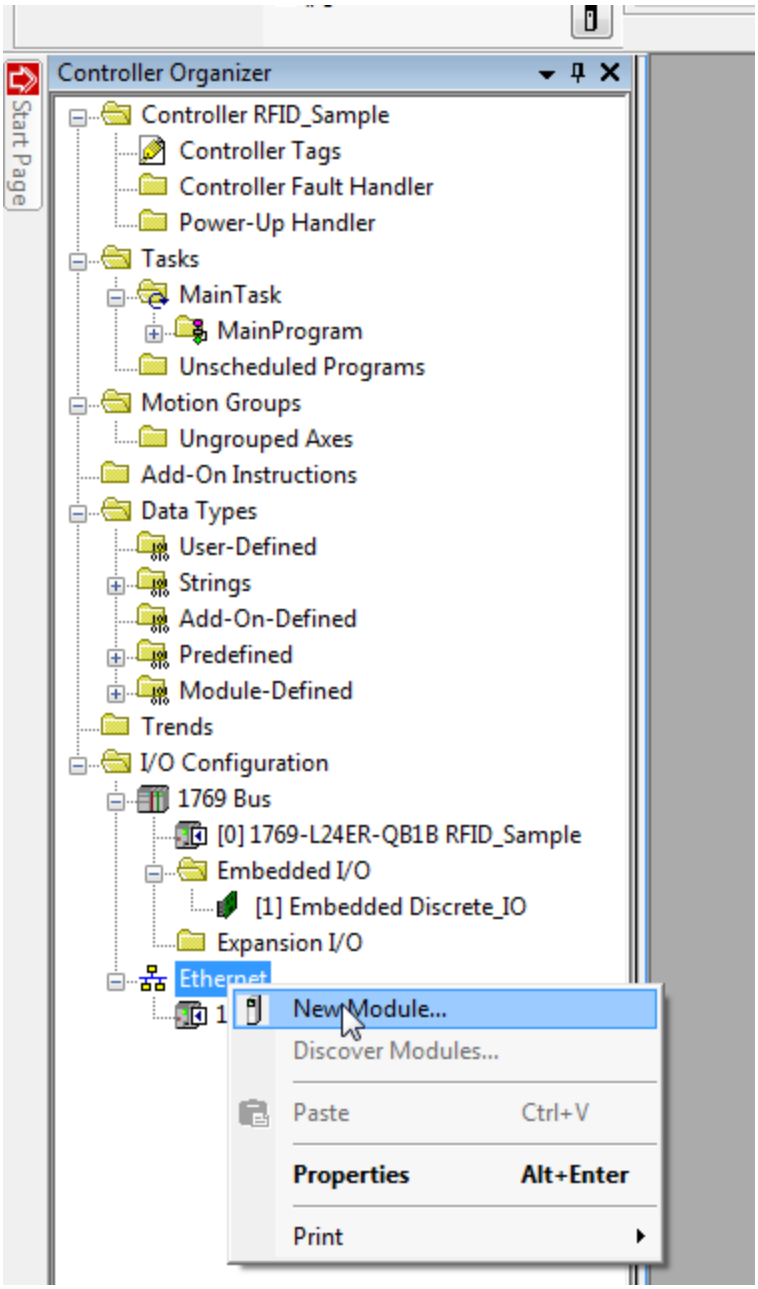
Setup using the EDS-AOP

The EDS files for all the controllers can be downloaded from the web site under their model number. The models are IC-KP2-1HB17-2V1D, IC-KP2-2HB17-2V1D and IC-KP-B17-AIDA1. Here is the procedure to install and use the EDS files.

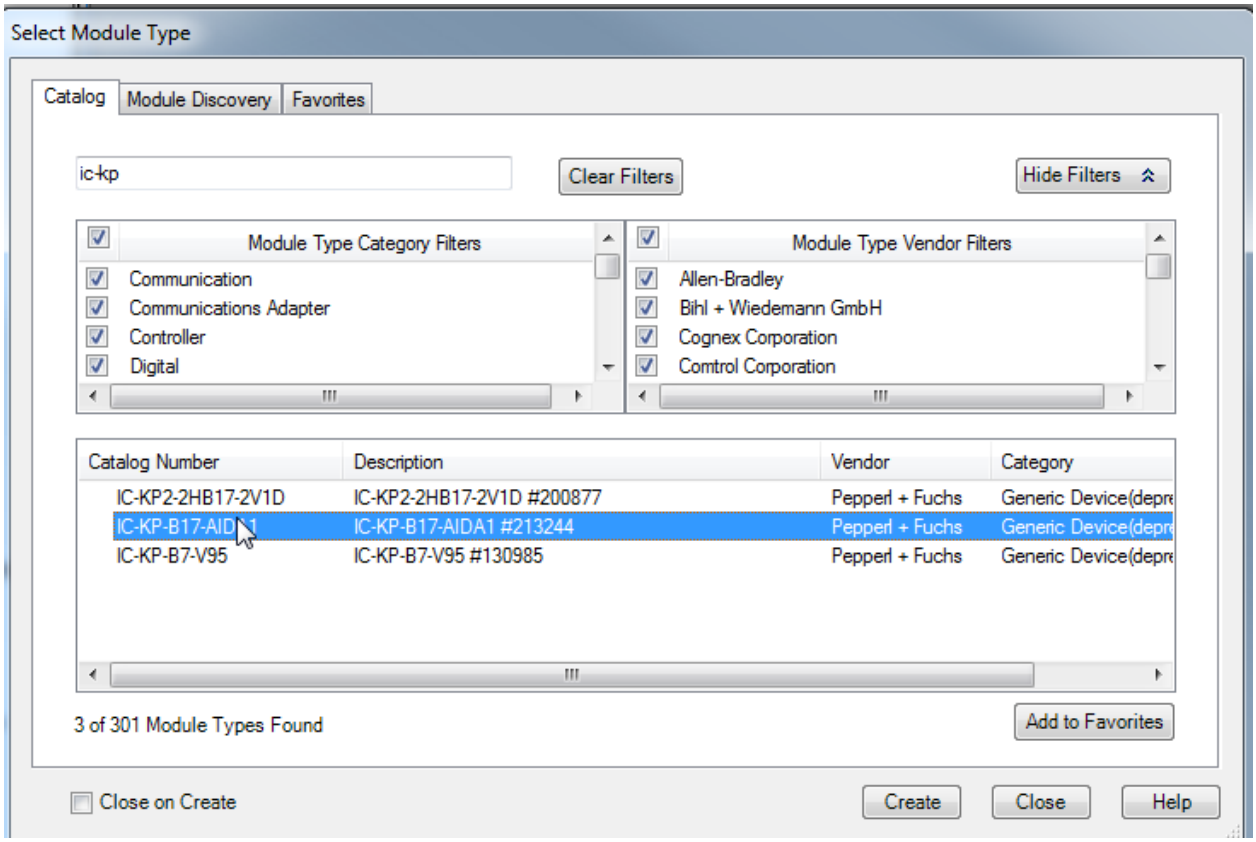
Step 1: Import the EDS file into your project.



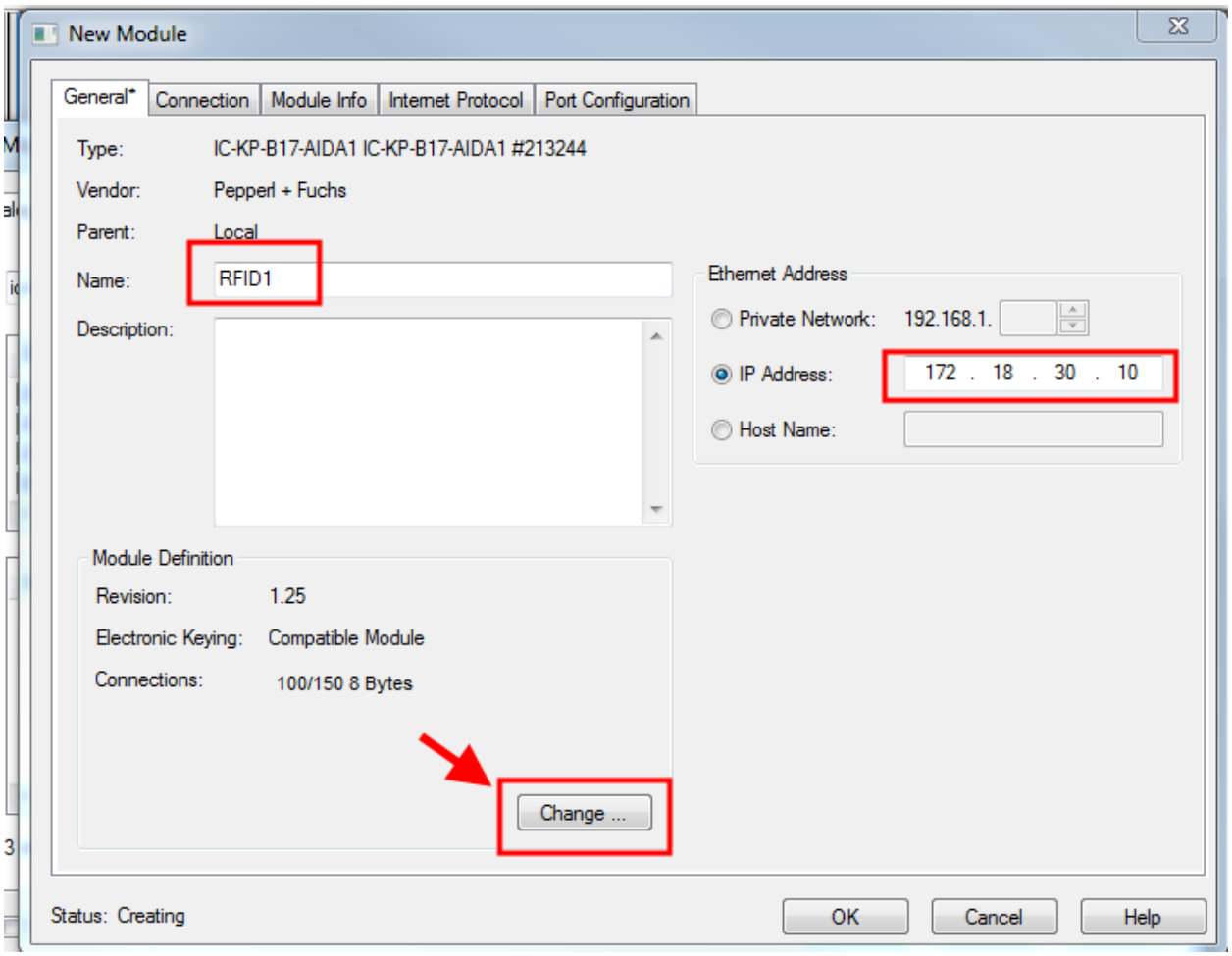
Step 2: Select your ethernet card or port and add a “New Module”



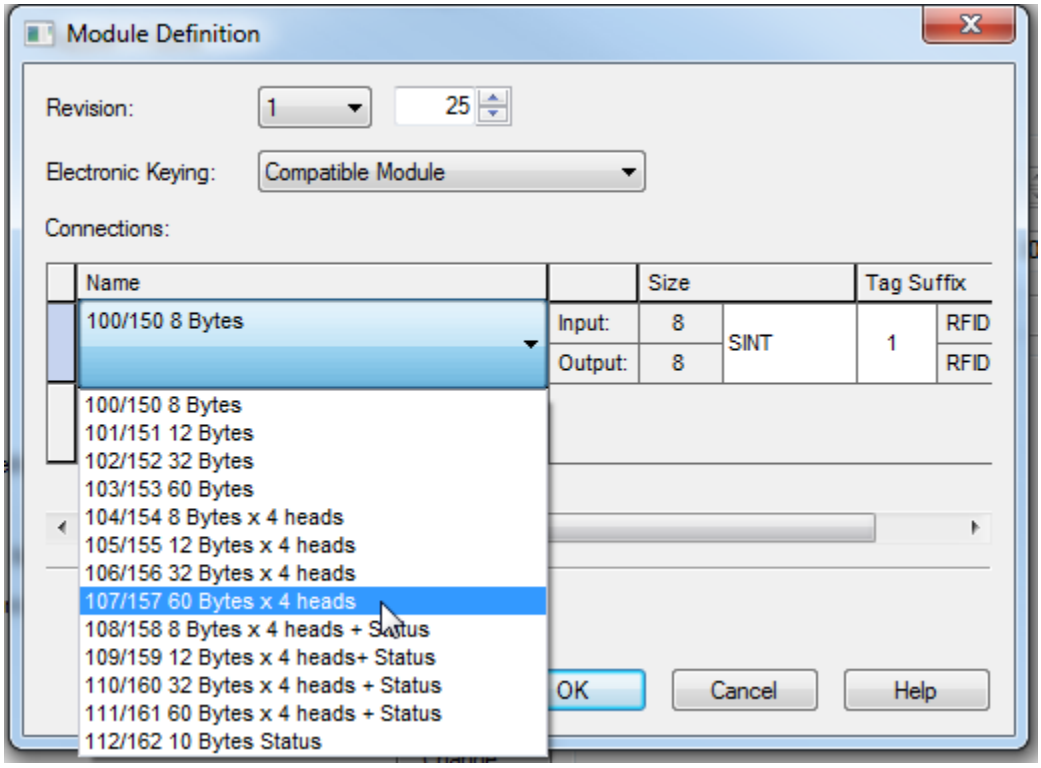
Step 3: Select the device/EDS file that you installed



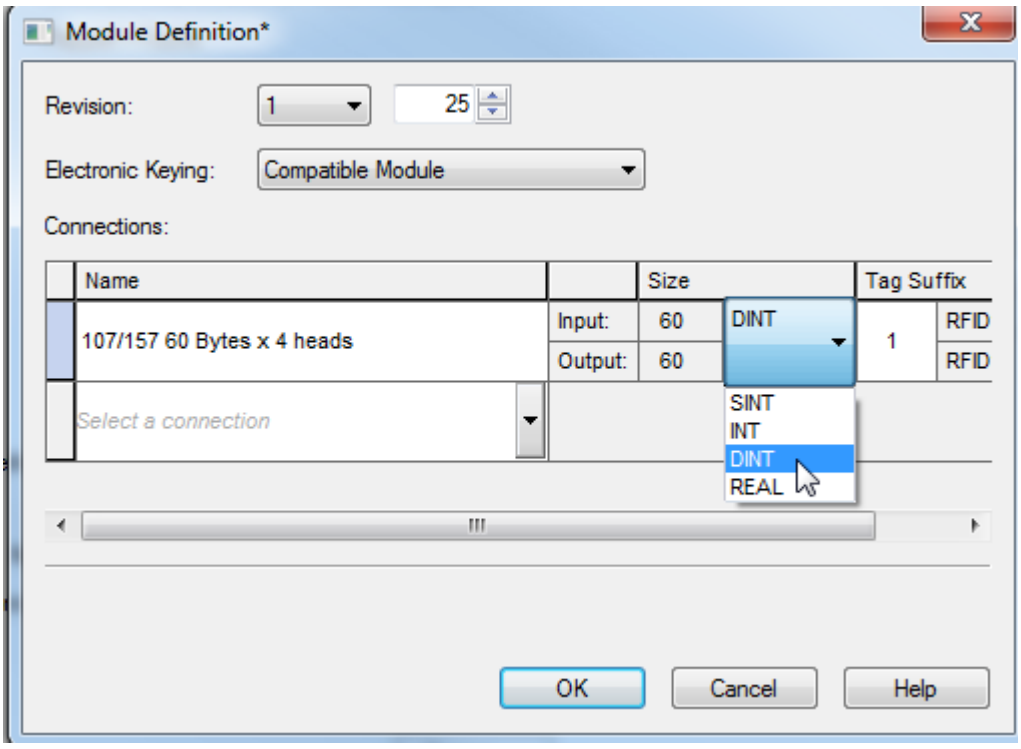
Step 4: Create a new name and configure the IP address of your RFID controller then click on “Change”



Step 5: Select the assembly instance that you want to use from the list



Step 6: Select “DINT” from the Size list



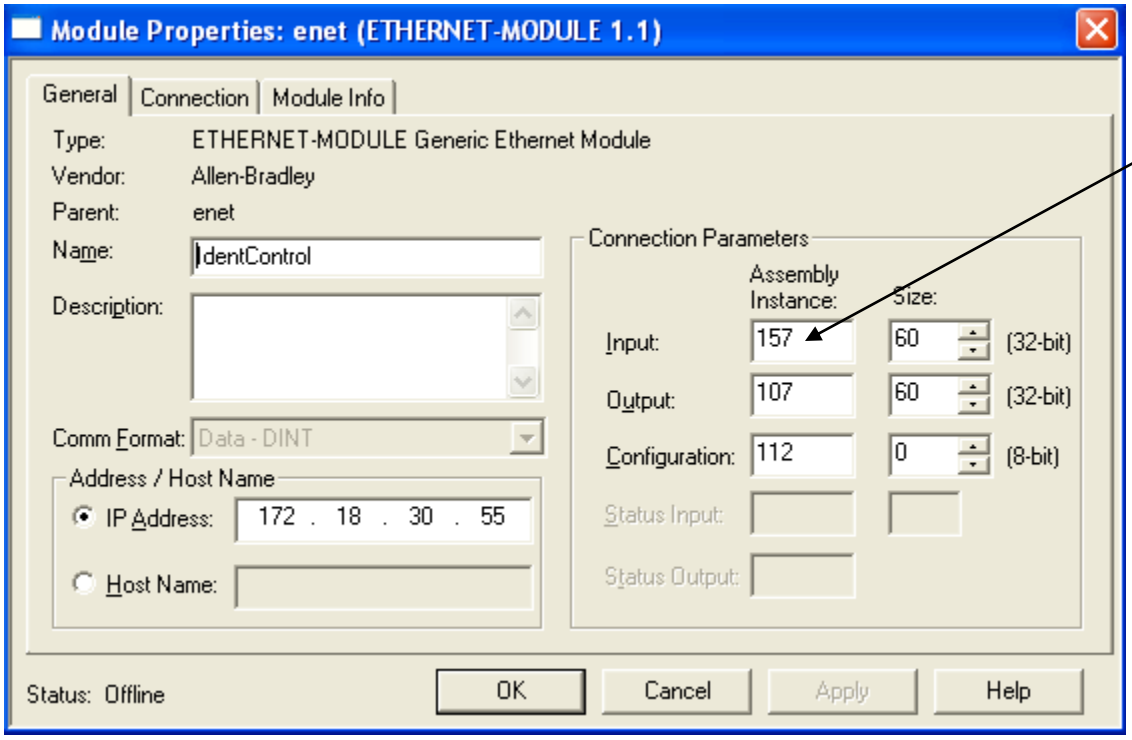
Using the instructions

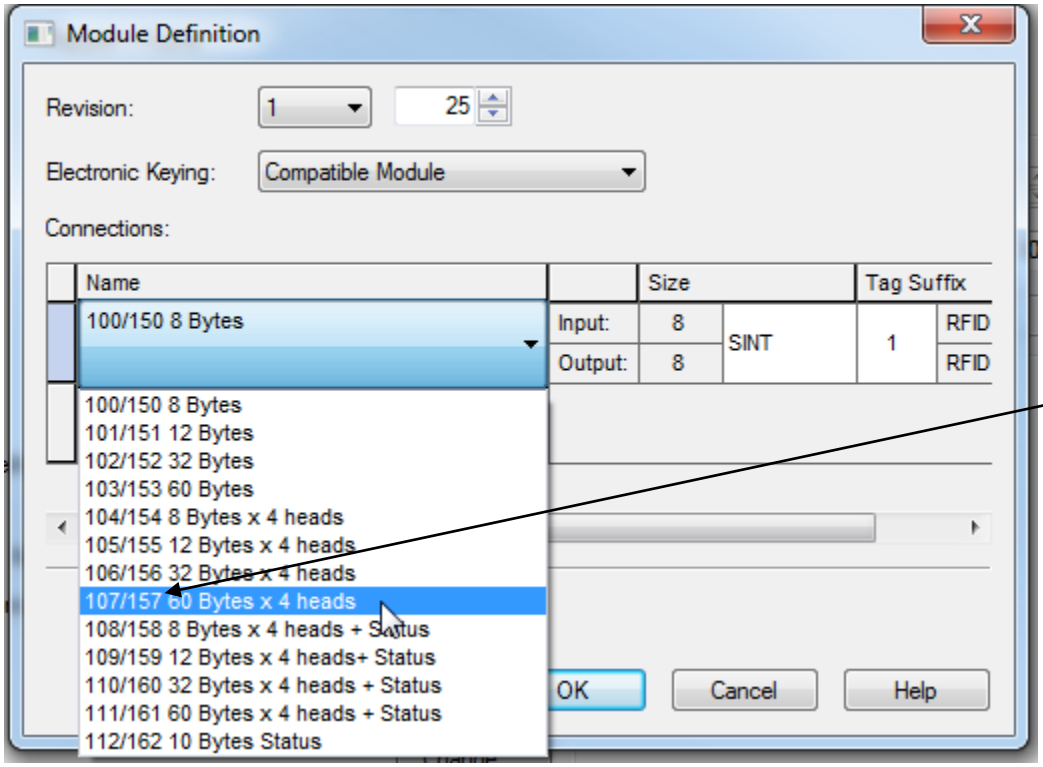
One Time Instructions
All of the instructions that end in "1Time" are enabled when the rung condition is made high. The instruction completes when the "DN" bit comes on. If the "DN" bit does not come on the instruction has not completed yet. If the "DN" bit comes on and the "ER" bit is not on then the instruction has completed without error, the status is 0 and if it was a read command the data can be read out of the "readdata" parameter. At this point the instruction can be made low again such that all of the bits DN, EN, ER will turn off. When this happens a new command can be reinitiated by raising the rung again. Also OneShots cannot be used to initiate these instructions. The entire rung must remain true for as long as the command is running.

Continuous Instructions
These instructions are those that end in "Cont". They internally run the enhanced read commands. These commands stay running for as long as the controller has power. Even if a read head is disconnected the command restarts when it is reconnected. These instructions run all the time and the rung must remain high continuously. When a tag shows up the read counter counts up. Also the tag present will go high. It is possible that the tag present bit statys on and a new tag counter counts up. This means that two tags passed without a "no tag, status 5" message in-between. Use the tag counter trigger to trigger on new tags.

Common Parameters

Certain parameters are common to all Add-on instruction. These parameters are listed here are are not repeated with the explanation of each individual add-on instruction.
Input Parameters
Assembly_Instance_Input – This is exactly the assembly instance that was used to setup your RFID system in the first place. This number must be exactly correct. Go to your I/O configuration of your PLC and then to the generic Ethernet module used for configuring the ID system





It is the number associated with the "Assembly Instance Input". Transfer this value to your instruction.

Tag_Type – The model number of tag you will be communicating with. See model number of tag or table below.

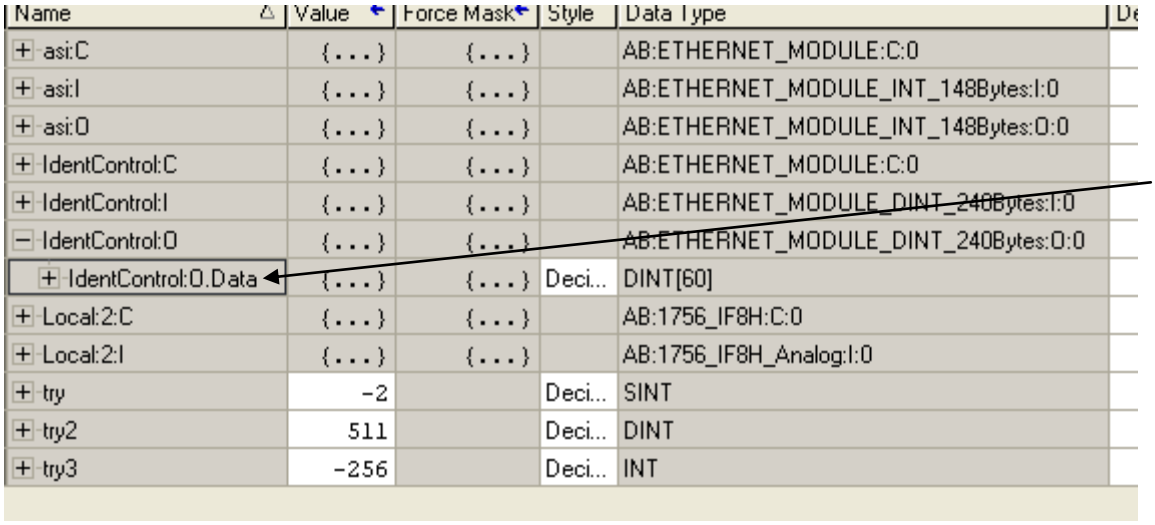
Supported tag types

Tag type		Description P+F	Chip type	Access	Bits	Fixcode length [byte]	Frequency range
High Byte	Low Byte						
'0'	'2'	IPC02	Unique, EM4102 (EM microelectronic)	Read only code	40	5	125 kHz
'0'	'3'	IPC03	EM4450 (EM microelectronic), Titan	Read/write read only code	928 32	4	125 kHz
'1'	'0'	IPC10	Nova (Sokymat)	Read/write	40	-	125 kHz
'1'	'1'	IPC11	Q5 (Sokymat)	Read/write	40	-	125 kHz
'1'	'2'	IPC12	P+F FRAM	Read/write read only code	64k 32	4	125 kHz
'1'	'4'	IPC14	T5557 (Atmel)	Read/write	40	-	125 kHz
'2'	'0'	IQC20[1)]	All ISO 15693 compliant tags	Read/write read only code		8	13.56 MHz
'2'	'1'	IQC21	I-Code SLI (NXP)	Read/write read only code	896 64	8	13.56 MHz
'2'	'2'	IQC22	Tag-it HF-I Plus (Texas Instruments)	Read/write read only code	2k 64	8	13.56 MHz
'2'	'3'	IQC23	my-D (Infinion) SRF55V02P	Read/write read only code	1792 64	8	13.56 MHz
'2'	'4'	IQC24	my-D (Infinion) SRF55V10P	Read/write read only code	7424 64	8	13.56 MHz
'3'	'3'	IQC33[2)]	Fujitsu FRAM MB89R118	Read/write read only code	16k	8	13.56 MHz
'3'	'5'	IQC35	I-Code SLI-S (NXP)	Read/write read only code	1280	8	13.56 MHz
'5'	'0'	IDC-...-1K	P+F	Read/write read only code	1k 32	4	250 kHz
'5'	'2'	ICC-...	P+F	Read only code	28	7	250 kHz
'6'	'0'	MVC-60	P+F RAM	Read/write read only code	59k	4	2.45 GHz

Read_Head_Number – This is the read head you will be sending the command to.

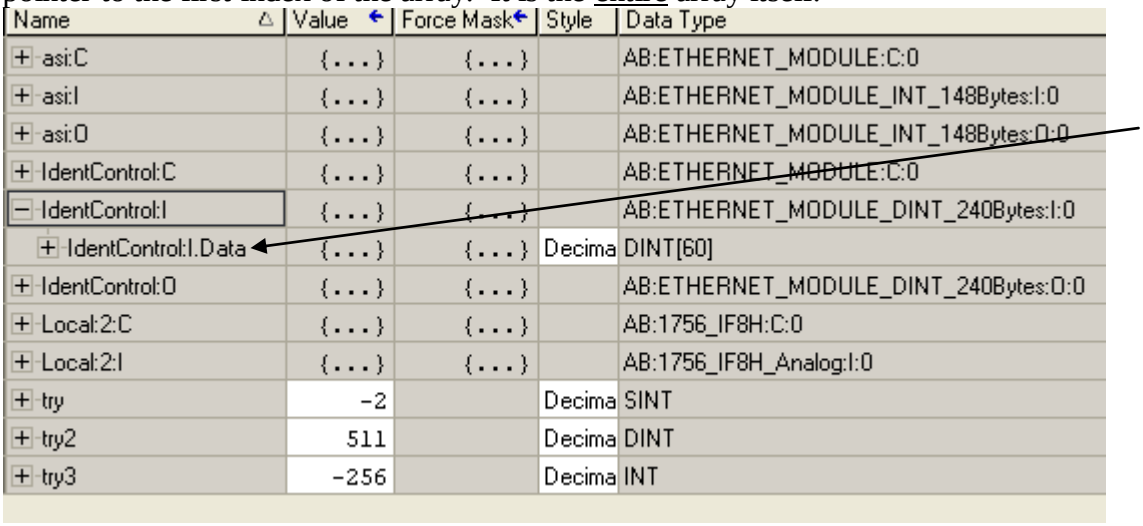
In/Out Parameters

Output – This is the controller tag associated with the output data for your ID system. It is not the pointer to the first index of the array. It is the entire array itself.



Name	Value	Force Mask	Style	Data Type
asi:C	{...}	{...}		AB:ETHERNET_MODULE:C:0
asi:I	{...}	{...}		AB:ETHERNET_MODULE_INT_148Bytes:I:0
asi:O	{...}	{...}		AB:ETHERNET_MODULE_INT_148Bytes:O:0
IdentControl:C	{...}	{...}		AB:ETHERNET_MODULE:C:0
IdentControl:I	{...}	{...}		AB:ETHERNET_MODULE_DINT_240Bytes:I:0
IdentControl:O	{...}	{...}		AB:ETHERNET_MODULE_DINT_240Bytes:O:0
IdentControl:O.Data	{...}	{...}	Deci...	DINT[60]
Local:2:C	{...}	{...}		AB:1756_IF8H:C:0
Local:2:I	{...}	{...}		AB:1756_IF8H_Analog:I:0
try	-2		Deci...	SINT
try2	511		Deci...	DINT
try3	-256		Deci...	INT

Input - This is the controller tag associated with the input data for your ID system. It is not the pointer to the first index of the array. It is the entire array itself.

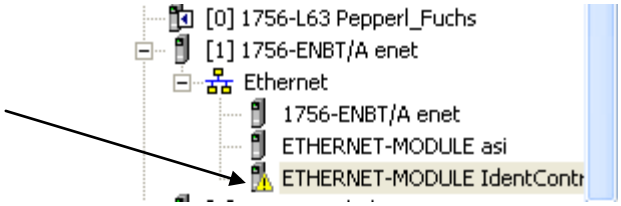


Name	Value	Force Mask	Style	Data Type
asi:C	{...}	{...}		AB:ETHERNET_MODULE:C:0
asi:I	{...}	{...}		AB:ETHERNET_MODULE_INT_148Bytes:I:0
asi:O	{...}	{...}		AB:ETHERNET_MODULE_INT_148Bytes:O:0
IdentControl:C	{...}	{...}		AB:ETHERNET_MODULE:C:0
IdentControl:I	{...}	{...}		AB:ETHERNET_MODULE_DINT_240Bytes:I:0
IdentControl:I.Data	{...}	{...}	Decima	DINT[60]
IdentControl:O	{...}	{...}		AB:ETHERNET_MODULE_DINT_240Bytes:O:0
Local:2:C	{...}	{...}		AB:1756_IF8H:C:0
Local:2:I	{...}	{...}		AB:1756_IF8H_Analog:I:0
try	-2		Decima	SINT
try2	511		Decima	DINT
try3	-256		Decima	INT

Output Parameters

Status – This byte will give you information about the command. Two additional status messages were defined to give you information about the execution of the Add-On Instruction itself

FE – Timeout. The command did not complete before the watchdog timer turned out. This status is caused by a missing IDENTControl interface. Check to see if it connected by looking at the connection status. The yellow warning triangle is a good indication of connection failure.



FD – Bad parameters. This is caused by an incorrect assembly instance number, a head number not 1,2,3, or 4 or a read or write data array that is smaller then the amount of data you want to read or write. For example if you want to read 14 DINTs the read_data_array must of size DINT[14] minimum.

Fault/Status messages

Status	Meaning
00h	Command has been executed without error.
FFh	Command is processing.

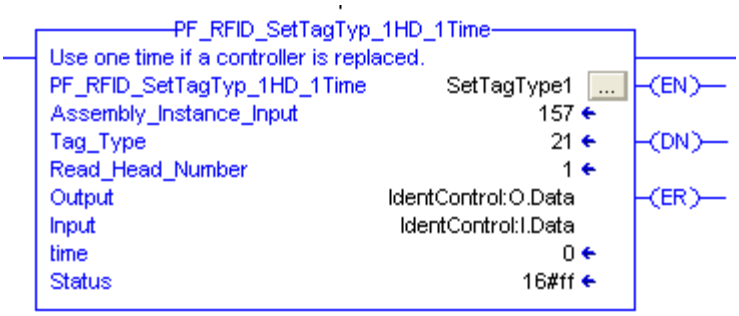
Fault messages triggered by the identification system

Status	Meaning
01h	Battery of the tag is weak.
02h	Reserved
03h	Reserved
04h	Command is incorrect or incomplete or parameter not in the valid range.
05h	No tag in the detection range.
06h	Hardware error, e.g. error during self-test or read/write head defect.
07h	Internal device error
08h	Reserved
09h	Programmed tag type is not compatible with the connected read/write head.
0Ah	Reserved
0Bh	Reserved
0Ch	Reserved
0Dh	Reserved
0Eh	Reserved
0Fh	Reserved

EN – When the rung is enabled this bit goes high. If the rung goes off this bit also turns off.
DN – When the command completes this bit is set. The value of the error bit determines if the command was successful or not. If the ER bit is low the command execution was a success. If the ER bit is high the command failed
ER – If this bit is high some sort of error occurred. Check the status for exact cause of the error.

Set tag type

PF_RFID_SetTagTyp_1HD_1Time – This command sets the tag type on one head one time. The command will then stop and DN will turn on. No error bit means the command was successful. Using this command is a better alternative to setting the tag type on the controller using Display or web page. If the controller becomes damaged and is swapped out, this command can be used to reset the tag type.



The tag type defines exactly what ASIC or chip is being used in the tag. For example the I-code SLI chip is our tag type 21. The different tag types support different commands so it is important for the read head to know exactly which one you are talking to. Also the different chips have different memory sizes. If you look at the Pepperl+Fuchs model number you can easily determine the tag type. It is a two-digit number directly after the prefix of IPC, IQC. Also IDCs have a tag type 50 and ICCs have a tag type 52 as stated in the table.

IPC03-50P > tag type is 3
IQC21-30P > tag type is 21
Tag Type Example

Parameters

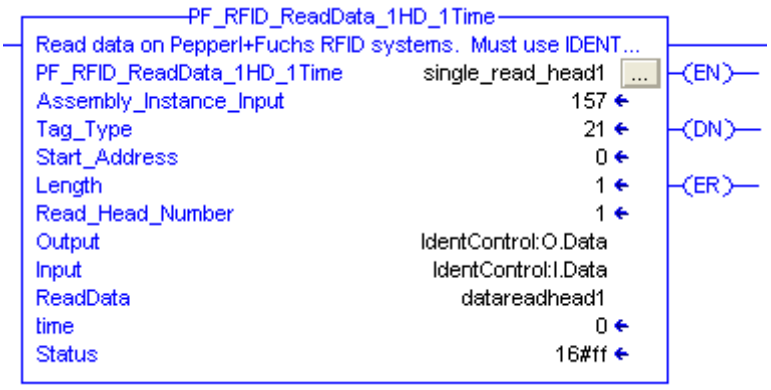
Input Parameters
No additional input parameters

In/Out Parameters
No additional input parameters

Output Parameters
Time – This is the recorded time is takes for the command to complete. The time stops when the DN bit goes high. If the command goes DN(done) with no ER(error) and you think this time is unusually long check the "Data hold time" This parameter can be found on the display of the controller or on the web page of the controller. This value is 500ms by default and can be reduced to a very low number or 0 in some instances speeding up your process.

Read Data

PF_RFID_ReadData_1HD_1Time – This command will read data from a specific start address on a tag from a single head one time. The command will then stop and DN will turn on. No error bit means the command was successful.



This command will read any number of 32-bit words or double integers(DINTs) from the tag. Make sure that the ReadData array is large enough to accept them.

Parameters

Input Parameters

Start_Address – This is the double integer (DINT) start location where the data will be read from the tag. This always starts at 0 and can be any number up to the last DINT on the tag. Make sure the combination of the start address and length do not exceed the maximum memory capacity defined by the tag type.

Length – This is the number of double integers (DINT) to be read from the tag. Always starts at one (IQC33 starts at 2) and can be any number up to the entire capacity of the tag. Make sure the combination of the start address and length do not exceed the maximum memory capacity defined by the tag type.

In/Out Parameters

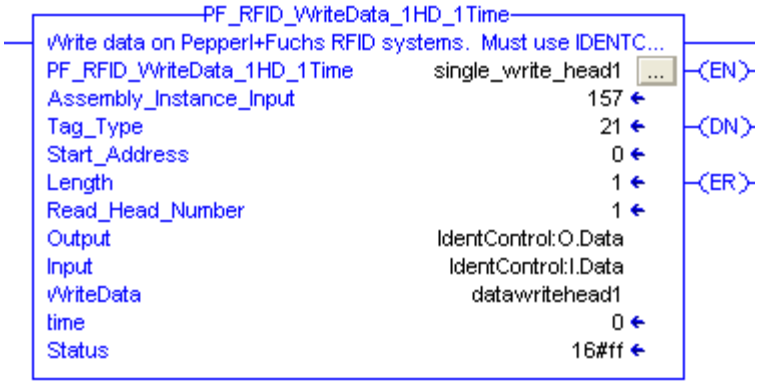
ReadData – This is type DINT[x] where is X is the size of the array/ Any length is allowed as long as it is larger then the length defined in the command.

Output Parameters

Time – This is the recorded time is takes for the command to complete. The time stops when the DN bit goes high. If the command goes DN(done) with no ER(error) and you think this time is unusually long check the "Data hold time" This parameter can be found on the display of the controller or on the web page of the controller. This value is 500ms by default and can be reduced to a very low number or 0 in some instances speeding up your process.

Write Data

PF_RFID_WriteData_1HD_1Time - This command will write data to a specific start address on a tag of a single head one time. The command will then stop and DN will turn on. No error bit means the command was successful.



This command will write any number of 32-bit words or double integers (DINT) to a tag. Make sure that the WriteData array is large enough and filled with data before executing the command.

Parameters

Input Parameters

Start_Address – This is the double integer(DINT) start location where the data will be written. This always starts at 0 and can be any number up to the last DINT on the tag. Make sure the combination of the start address and length do not exceed the maximum memory capacity defined by the tag type.

Length – This is the number of double integers(DINT) to be written to the tag. Always starts at one and can be any number up to the entire capacity of the tag. Make sure the combination of the start address and length do not exceed the maximum memory capacity defined by the tag type.

In/Out Parameters

WriteData – This is type DINT[x] where is X is the size of the array/ Any length is allowed as long as it is larger then the length defined in the command. Make sure the data is filled into this array and not being changed during the entire time the command is executing.

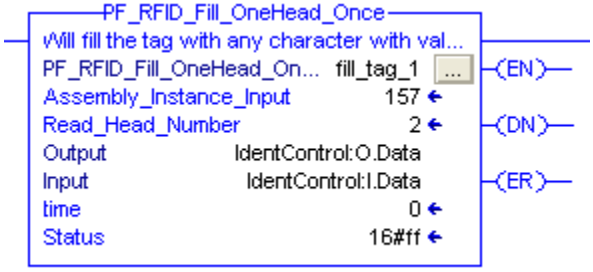
Output Parameters

Time – This is the recorded time is takes for the command to complete. The time stops when the DN bit goes high. If the command goes DN(done) with no ER(error) and you think this time is unusually long check the "Data hold time" This parameter can be found on the display of the controller or on the web page of the controller. This value is 500ms by default and can be reduced to a very low number or 0 in some instances speeding up your process.

Clear Data

PF_RFID_ClearData_1HD_1Time – The command will clear the entire contents of the tag. The exact tag type must have been defined previously using the Set Tag Type command. If not you will get an error because if the controller does not know what chip type is in front of the reader it can't clear all the memory within the tag. This is a much faster way to clear the

contents of a tag because the read head is putting the same byte value to every register on the tag. A write command could do the same job it would just take a lot longer.



Parameters

Input Parameters

Fill_Data – Normally hidden but can be made visible if desired. Defines that easy byte on the tag will be cleared with. By default this byte is 0 but it could be any value from 0x00 up to 0xFF.

Start_Address – Normally hidden but can be set if the entire tag should not be cleared but rather just a small section should be. The Length must also be a non-zero number if this parameter is used

Length – Normally hidden but can be set if the entire tag should not be cleared but rather just a small section should be.

In/Out Parameters

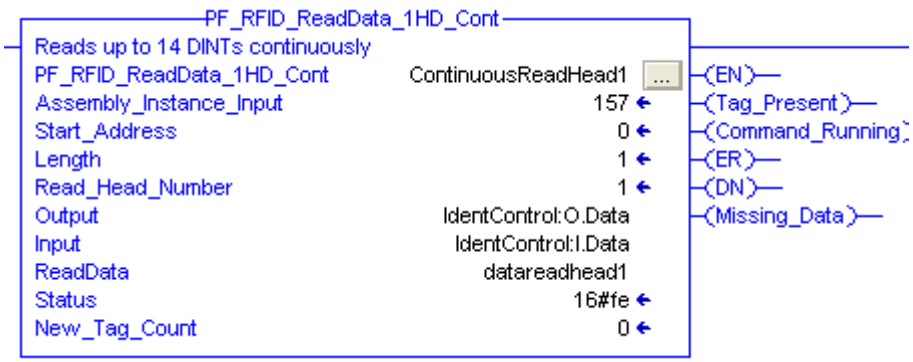
No additional In/Out parameters

Output Parameters

Time – This is the recorded time is takes for the command to complete. The time stops when the DN bit goes high. If the command goes DN(done) with no ER(error) and you think this time is unusually long check the "Data hold time" This parameter can be found on the display of the controller or on the web page of the controller. This value is 500ms by default and can be reduced to a very low number or 0 in some instances speeding up your process.

Read Continuously

PF_RFID_ReadData_1HD_Cont – Can read up to 14 double integers from a tag, depending on tag type, from any location on the tag. Once enabled and not done the command will run indefinitely. No other command should be executed while this command runs. These commands are often used to read data on the fly. Depending on the amount of data read tag-passing speeds can exceed 3m/s.



Parameters

Input Parameters

Start_Address – This is the double integer (DINT) start location where the data will be read from the tag. This always starts at 0 and can be any number up to the last DINT on the tag. Make sure the combination of the start address and length do not exceed the maximum memory capacity defined by the tag type.

Length – This is the number of double integers (DINT) to be read from the tag. Always starts at one and can be any number up to the entire capacity of the tag. Make sure the combination of the start address and length do not exceed the maximum memory capacity defined by the tag type.

In/Out Parameters

ReadData – This is type DINT[x] where is X is the size of the array/ Any length is allowed as long as it is larger then the length defined in the command.

Out Parameters

New_Tag_Count – Every time a new tag is read this double integer counts up by one. It will count up to 2,147483,647 and then roll to a negative number just like a counter would.

Tag_Present – When a tag is in front of the read head this bit is high otherwise it is low.

Command_Running – If the command was executed successfully and is currently running successfully then this bit will be high otherwise it is low.

Missing_Data – Every time a new status message is sent by the ID system an internal execution counter increases by 1. If this count every goes up by 2 or more then data was missed. This could mean that your "Data_Retention_Time" parameter is too low. Increase this time until you don't miss any tags. Once this bit turns on it will remain on until it is manually lowered or the rung goes low and high again.

Status_2_count – Normally hidden, will count how often that status was a 2

Status_4_count – Normally hidden, will count how often that status was a 4

Status_5_count – Normally hidden, will count how often that status was a 5

Status_6_count – Normally hidden, will count how often that status was a 6

Status_7_count – Normally hidden, will count how often that status was a 7

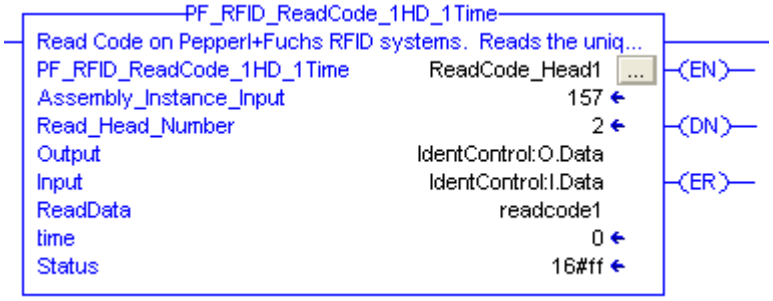
Status_8_count – Normally hidden, will count how often that status was an 8

Status_9_count – Normally hidden, will count how often that status was a 9

Status_255_count – Normally hidden, will count how often that status was a 255(-1)
Status_other_count - Normally hidden, will count how often a status occurs that is different then any of the above listed statuses.

Read Code

PF_RFID_ReadCode_1HD_1Time – This command will read data from a specific start address on a tag from a single head one time. The command will then stop and DN will turn on. No error bit means the command was successful.
This command will read any number of 32-bit words or double integers(DINTs) from the tag. Make sure that the ReadData array is large enough to accept them.



Parameters

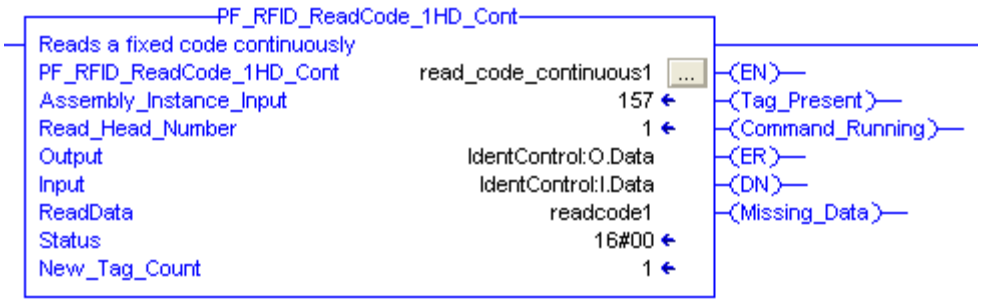
Input Parameters
No additional input parameters

In/Out Parameters
ReadData – This is type DINT[x] where is X is the size of the array/ Any length is allowed as long as it is larger than the length defined in the command.

Output Parameters
Time – This is the recorded time is takes for the command to complete. The time stops when the DN bit goes high. If the command goes DN(done) with no ER(error) and you think this time is unusually long check the "Data hold time" This parameter can be found on the display of the controller or on the web page of the controller. This value is 500ms by default and can be reduced to a very low number or 0 in some instances speeding up your process.

Read Code Continuously

PF_RFID_ReadData_1HD_Cont – Can read up to 14 double integers from a tag, depending on tag type, from any location on the tag. Once enabled and not done the command will run indefinitely. No other command should be executed while this command runs. These commands are often used to read data on the fly. Depending on the amount of data read tag-passing speeds can exceed 3m/s.



Parameters

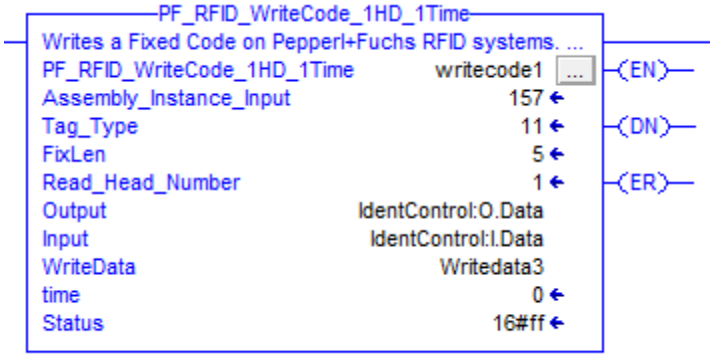
Input Parameters
No additional input parameters

In/Out Parameters
ReadData – This is type DINT[x] where is X is the size of the array/ Any length is allowed as long as it is larger than the length defined in the command.

Out Parameters
New_Tag_Count – Every time a new tag is read this double integer counts up by one. It will count up to 2,147,483,647 and then roll to a negative number just like a counter would.
Tag_Present – When a tag is in front of the read head this bit is high otherwise it is low.
Command_Running – If the command was executed successfully and is currently running successfully then this bit will be high otherwise it is low.
Missing_Data – Every time a new status message is sent by the ID system an internal execution counter increases by 1. If this count every goes up by 2 or more then data was missed. This could mean that your "Data_Retention_Time" parameter is too low. Increase this time until you don't miss any tags. Once this bit turns on it will remain on until it is manually lowered or the rung goes low and high again.
Status_2_count – Normally hidden, will count how often that status was a 2
Status_4_count – Normally hidden, will count how often that status was a 4
Status_5_count – Normally hidden, will count how often that status was a 5
Status_6_count – Normally hidden, will count how often that status was a 6
Status_7_count – Normally hidden, will count how often that status was a 7
Status_8_count – Normally hidden, will count how often that status was a 8
Status_9_count – Normally hidden, will count how often that status was a 9
Status_255_count – Normally hidden, will count how often that status was a 255(-1)
Status_other_count - Normally hidden, will count how often a status occurs that is different then any of the above listed statuses.

Write Code

This command will write a fixcode to an RFID tag one time. A typically used RFID system would be an IPH… read head and an IPC11… tag. When using this command the read head must be configured for the appropriate read/write tag like IPC11 or IDC. The read head cannot be configured for ICC or IPC02.



Parameters

Input Parameters
Tag_type- This parameter will determine if the tag is locked after programming or not. 11 = tag unlocked and still writeable, 2 is locked and can never be written again.
FixLen – This is the fixed code length in bytes, IPC11 = 5, IDC = 7

In/Out Parameters
WriteData – This is type DINT[2] and this is the data that will be written to the fixcode

Output Parameters
Time – This is the recorded time is takes for the command to complete. The time stops when the DN bit goes high. If the command goes DN(done) with no ER(error) and you think this time is unusually long check the "Data hold time" This parameter can be found on the display of the controller or on the web page of the controller. This value is 500ms by default and can be reduced to a very low number or 0 in some instances speeding up your process.

Addendum: DeviceNet

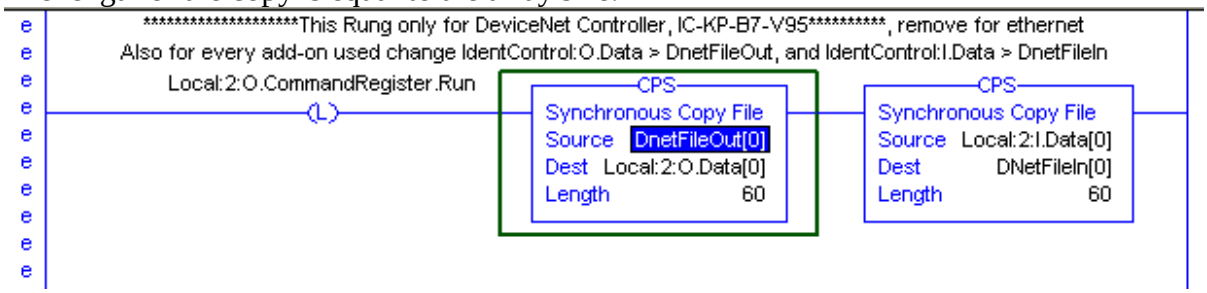
If a DeviceNet IDENTControl is being used, IC-KP-B7-V95, then an additional two copy instructions are required initially. The reason is simply because the way the add-on instructions work all the I/O data must be grouped in to one large array. If for some reason you only have one IDENTControl on your DeviceNet network and you made the DeviceNet mapping identical to the size of the IDENTControl this would not be necessary. This is highly unlikely because DeviceNet cards are used to connect to many types of I/O devices.

The size of the variables DnetFileOut and DnetFileIn need to be exactly correct and depend on the assembly instances used. Check this table for the correct array size to use:

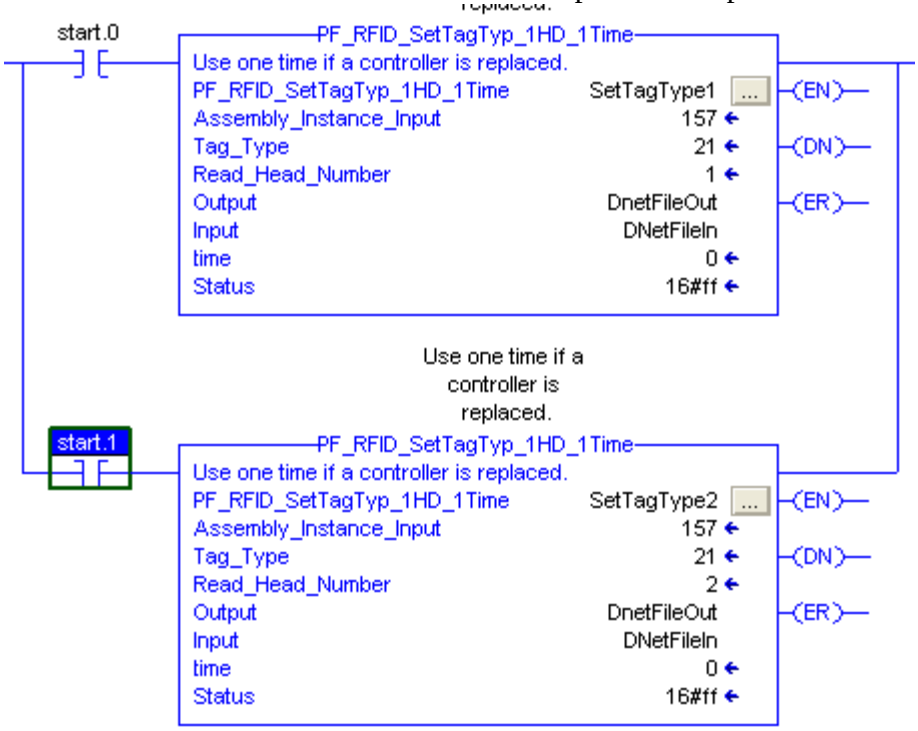
Assembly Instance Input	Assembly Instance Output	Array Size	Description
150	100	DINT[2]	Mixes mode, 1 DINTs at a time

151	101	DINT[3]	Mixed mode, 2 DINTs at a time
152	102	DINT[8]	Mixed mode, 7 DINTs at a time
153	103	DINT[15]	Mixed mode, 14 DINTs at a time
154	104	DINT[8]	Separated mode, 1 DINTs at a time
155	105	DINT[12]	Separated mode, 2 DINTs at a time
156	106	DINT[32]	Separated mode, 7 DINTs at a time
157	107	DINT[60]	Separated mode, 14 DINTs at a time

Make a rung that looks like this to move the DeviceNet data to the new DeviceNet arrays. The length of the copy is equal to the array size.

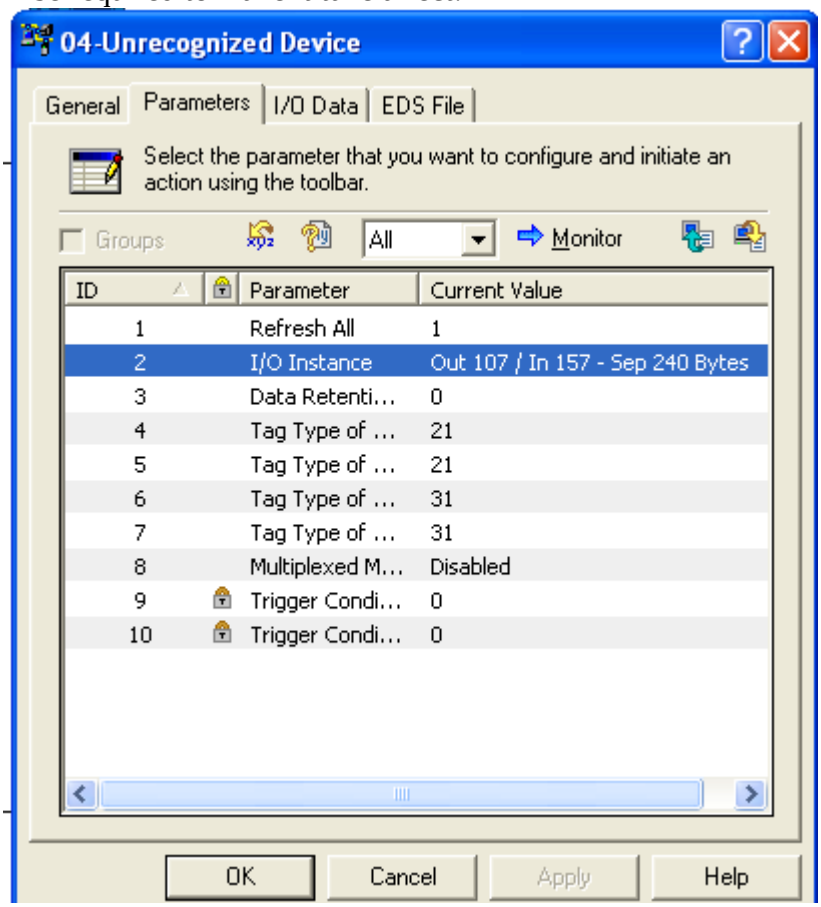


Then edit the instructions with the new names for the Input and Output variables like this

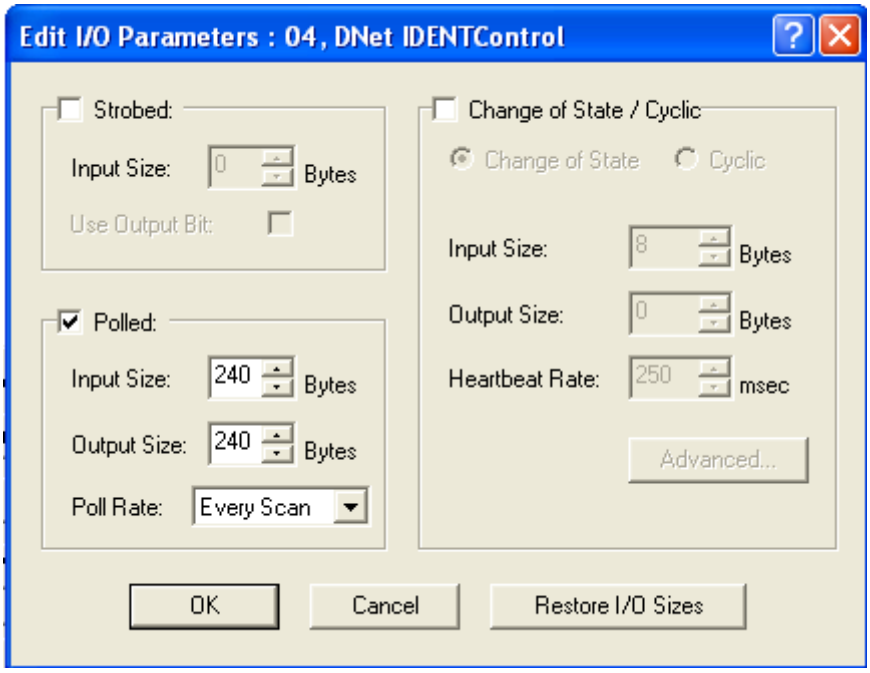


When setting up the DeviceNet IDENTControl verify that the correct Assembly instance has been set. You can check/set this on the graphical display or DeviceNet parameter. In the below example the assembly instance of 107/157 was selected. This means that an array size

of 60 DINTs(240 bytes/4) is required. If this setting is done using DeviceNet parameter a power cycle will be required to make it take affect.



Make sure that after you set the assembly instance of the controller the Polled Input and Output sizes are manually. Unfortunately RSNetworx does not do this automatically. This can be found by adding the controller to the scan list and then highlighting the device and choosing Edit I/O Parameters. Once this size has been changes download the changes to the DeviceNet controller and auto map the new data.


Edit I/O Parameters : 04, DNet IDENTControl
Strobed:
Input Size:
0
Bytes
Use Output Bit:
Polled:
Input Size:
240
Bytes
Output Size:
240
Bytes
Poll Rate:
Every Scan
Change of State / Cyclic
Change of State
Cyclic
Input Size:
8
Bytes
Output Size:
0
Bytes
Heartbeat Rate:
250
msec
Advanced...
OK
Cancel
Restore I/O Sizes