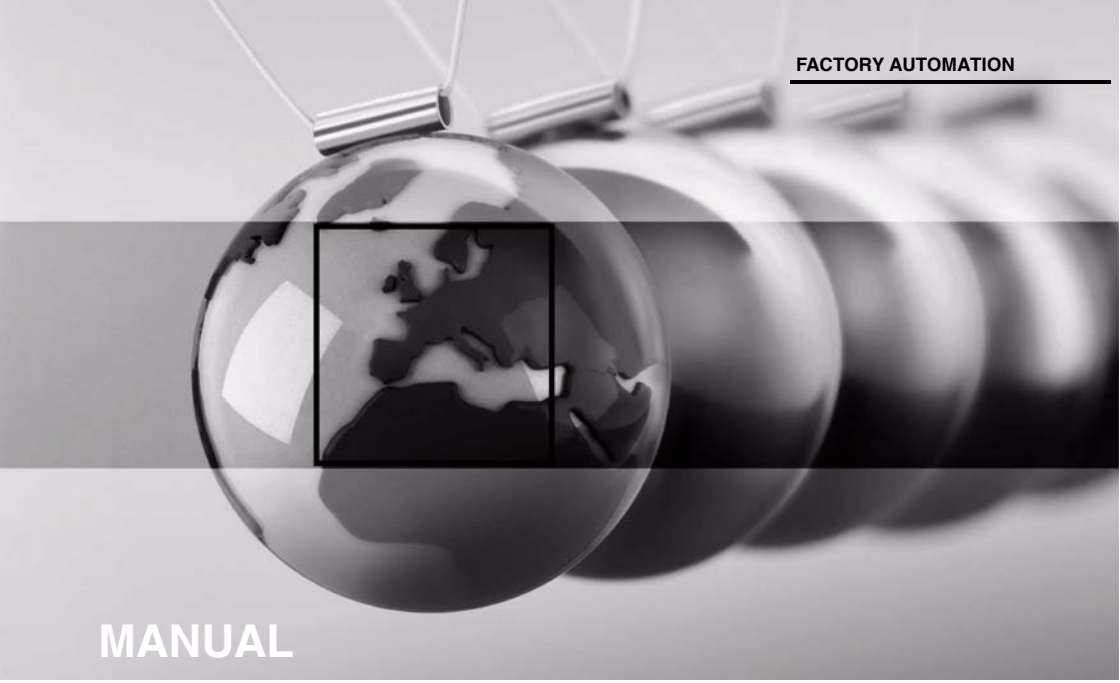
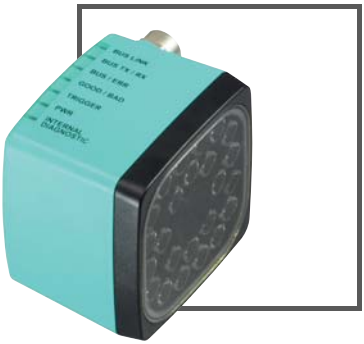


MANUAL

# OPC120P-F201-B17

## Integration into SIMATIC STEP 7



PEPPERL+FUCHS
SENSING YOUR NEEDS

With regard to the supply of products, the current issue of the following document is applicable: The General Terms of Delivery for Products and Services of the Electrical Industry, published by the Central Association of the Electrical Industry (Zentralverband Elektrotechnik und Elektroindustrie (ZVEI) e.V.) in its most recent version as well as the supplementary clause: "Expanded reservation of proprietorship"

2014-12

# 1 Integrating hardware

This section explains how to integrate the Optical Print Inspector via PROFINET. An example of integration is shown.

## 1.1 Installing the GSD file

You will find the current GSD file on our homepage http://www.pepperl-fuchs.com.

### Installing the GSD file

1. Before installing a GSD file, close all hardware configuration projects.
2. To install the GSD file, select **Options > Install GSD files** in the hardware configuration.

## 1.2 Integrating Optical Print Inspector

### Integrating Optical Print Inspector

1. To integrate an Optical Print Inspector into your PROFINET, double-click the PN-IO unit in the rack.

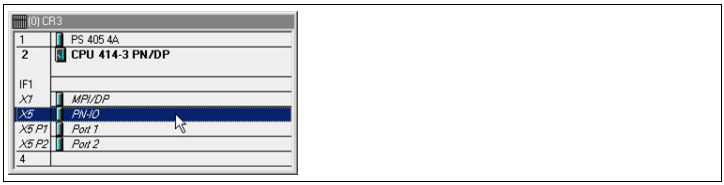



Figure 1.1 Assigned rack

↳ This opens the Properties window.

2. Click **Properties**.
3. To create a new Ethernet subnet, click **New**.
4. To insert the Ethernet subnet into the hardware configuration, right-click the PN-IO unit and select **Insert PROFINET-IO system**.

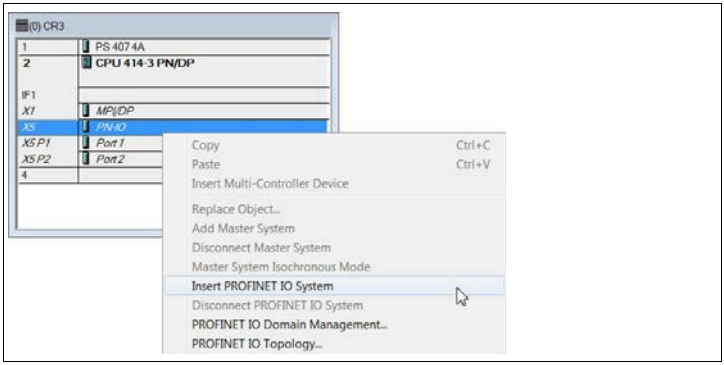



Figure 1.2 Inserting PROFINET-IO system

2014-12

↳ A PROFINET-IO system is now available to which you can connect new devices.

5. Drag the PROFINET module of the Optical Print Inspector from the catalog into the connection window and link it to the PROFINET-IO system.

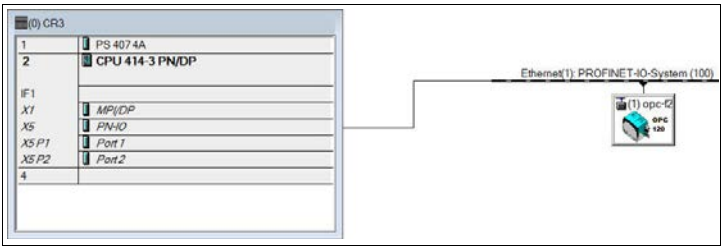



Figure 1.3 PROFINET topology

6. To assign the Optical Print Inspector to the PROFINET module just inserted, select **Destination system > Ethernet > Edit Ethernet device** from the menu bar. In the window that opens, click **Browse**.

↳ A list appears containing all accessible bus devices.

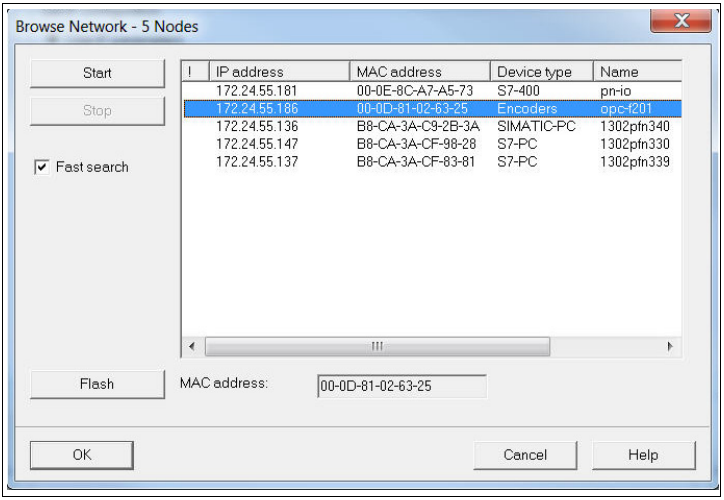



Figure 1.4 Browsing PROFINET

7. Select the Optical Print Inspector from the list (in this example **opc-f201**). To identify a device more easily, click on **Flash**. This causes the GOOD/BAD LED of the Optical Print Inspector to start flashing.
8. Click **OK**.

9. Activate the **Use IP parameter** option in the **Edit Ethernet device** window.

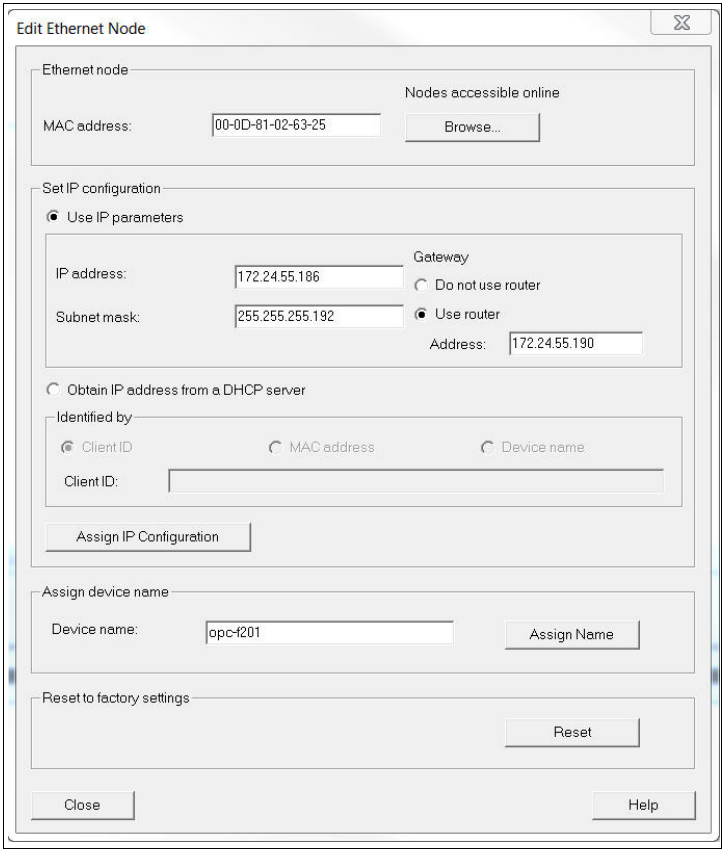



Figure 1.5 Editing Ethernet devices

10. If the name of the device from the list open previously is present in the area **Assign device name** (in this example **opc-f201**), click **Assign name**.
11. Click **Close**.
12. Double-click the PROFINET module in the connection window and check whether the device name has been successfully transferred. If the device name has not been transferred, enter the device name in the field **Device name**.

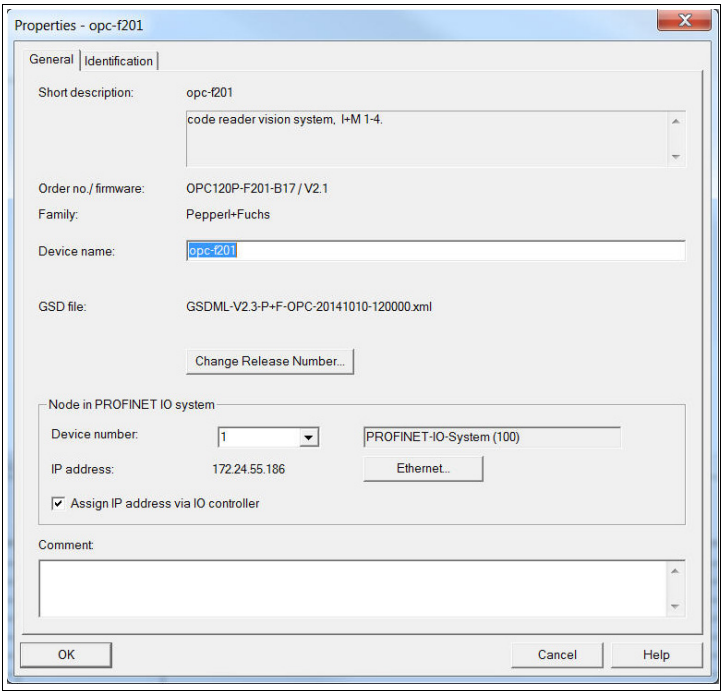



Figure 1.6 PROFINET module properties

13. Click **OK**.
14. To assign address areas for inputs and outputs, add the following modules from the catalog to the Optical Print Inspector:
    - Result counter: Good results
    - Result counter: Bad results
    - Read quality
    - Gray value
    - Software trigger
    - Result 64 byte

| Slot | Module ... | Order number | I Address | Q address | Diagnostic Address |
|---|---|---|---|---|---|
| *0* | *opc-f201* | *OPC120P-F201-B17* | | | *8186** |
| *Interface* | *Interface* | | | | *8185** |
| *Port 1* | *Port 1* | | | | *8184** |
| *Port 2* | *Port 2* | | | | *8183** |
| *1* | *Configuration* | | | | *8182** |
| 2 | | | | | |
| 3 | | | | | |
| 4 | | | | | |
| 5 | counter value GOOD | | 514...515 | | |
| 6 | code quality | | 518...521 | | |
| 7 | BAD value counter | | 516...517 | | |
| 8 | | | | | |
| 9 | current grey value | | 512...513 | | |
| 10 | software trigger | | | 512...513 | |
| 11 | | | | | |
| 12 | | | | | |
| 13 | | | | | |
| 14 | | | | | |
| 15 | | | | | |
| 16 | | | | | |
| 17 | result 64 byte | | 556...619 | | |

Figure 1.7 PROFINET configuration tables

***Note!***

Only use the **Result 64 byte** module.

# 2 Inserting function block and data module

## Inserting function block and data module

1. Unzip the zip file.
2. In the module folder, mark the function block **OPC120P-F201**, the associated instance data block **iDB_OPC120P-F201**, and the UDT **I/O Address OPC_Modules**. Right-click the marked entries and select **Copy**.
3. Right-click the destination project and select **Insert**.

***Note!***

If the modules cannot be inserted because of a numbering conflict, change the numbers of the function blocks and data modules. See chapter 3

# 3 Changing the module number

If you want to assign a new number to the function block and the data module, follow the instructions in this chapter.

***Note!***

Create a backup copy of your project so that any unwanted change to the absolute address caused by changed icons can be canceled if necessary.

## Changing the module number

Before you begin changing the module number, close all open modules.

1. Right-click the module folder.
2. Internally, the code reader function block accesses the command datasets of the data module symbolically. In order not to change the symbolic assignments, select the combination **Recommended for symbolic programming/Symbol has priority** on the tab **Address priority**.

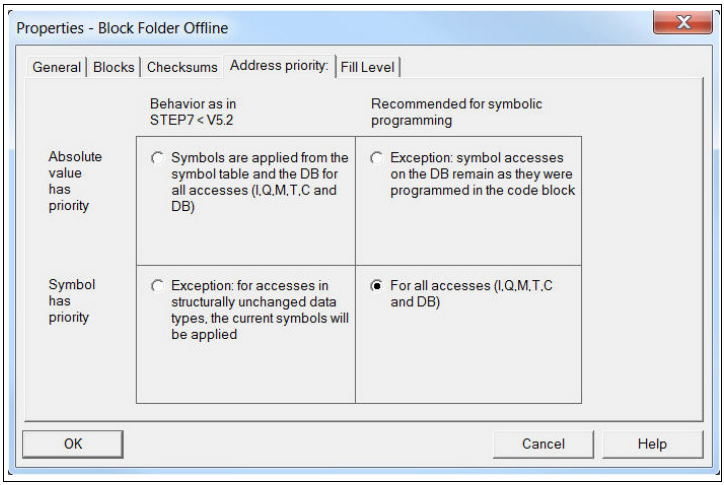



Figure 3.1 Setting the address priority

3. Click **OK**.
4. To change the number of a module, right-click **DB/FB** and select **Rename**.
5. To reestablish the symbolic assignment to the renamed module, open the symbols table.

6. Assign the new number to the function block symbol and the data module symbol so that the values in the lines **Address** and **Data type** are transferred.

| | | | | | |
|---|---|---|---|---|---|
| Codereader | FB | 64 | FB | 64 | ControlFB for P+F Codereader VB14N, MAH120, MAC335, MAC502 |
| Codereader-Data | DB | 53 | DB | 53 | Commands for VB14N, MAH120, MAC335, MAC502 |

7. Save the symbols table.
8. To update the module folder, press the **F5** key. The symbols are now assigned to the modules in the module folder.
9. Open the renamed function block.

   ↳ The following message appears. This message is to inform you that absolute addresses have changed due to symbolic assignments.
10. Save and close the function block.
11. Right-click the module folder.
12. Reset the settings on the **Address priority** tab to their original values.
13. Click **OK**.

# 4 Function block description

The function block **OPC120P-F201-B17** and the associated instance data block are called via:

`CALL OPC120P-F201-B17, iDB_OPC120P-F201` (symbolic representation)

This module reads in a Data Matrix code and stores it in its instance data block. The I/O addresses of the individual communication modules are parameterized via the user-defined data type (UDT), and the parameters are then transferred to the function block as an input variable.

The following image shows the call of the function block and the variables to be parameterized.


Figure 4.1 Calling the function block

### Input/output variables

| Name | Data type | Input/output | Description |
|---|---|---|---|
| Start | BOOL | Input | Starts a read command (positive edge) |
| IO_Module_Address | UDT | Input | I/O addresses of the communication modules |
| Busy | BOOL | Output | Command is being processed |
| Done | BOOL | Output | Command terminated |
| GoodRead | BOOL | Output | New data present |
| BadRead | BOOL | Output | No data read |
| GoodReads | WORD | Output | Counter value: successful reads |
| BadReads | WORD | Output | Counter value: failed reads |
| Error | BOOL | Output | Error occurred during processing |
| ErrorStatus | WORD | Output | Status value: 0 = OK, -1 = Timeout |

2014-12

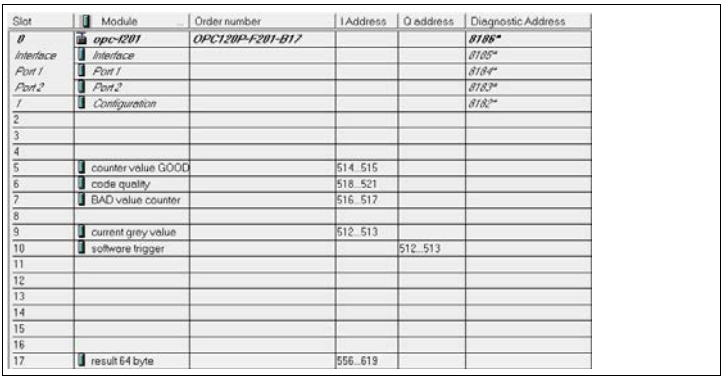
## 4.1 Setting communication parameters

***Note!***

Only the modules used in the example are required by the function block **OPC120P-F201-B17** for processing.

### Setting communication parameters

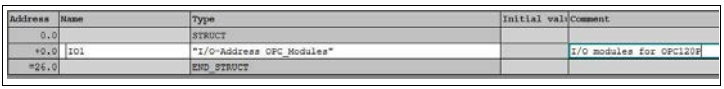
1. Transfer the I/O addresses of the individual modules defined in the hardware configuration to the UDT **I/O Address OPC_Modules**. You only need to enter the respective start address.



| Slot | Module | Order number | I Address | Q address | Diagnostic Address |
|---|---|---|---|---|---|
| *0* | *opc-f201* | *OPC120P-F201-B17* | | | *8186** |
| *Interface* | *Interface* | | | | *8185** |
| *Port 1* | *Port 1* | | | | *8184** |
| *Port 2* | *Port 2* | | | | *8183** |
| *1* | *Configuration* | | | | *8182** |
| 2 | | | | | |
| 3 | | | | | |
| 4 | | | | | |
| 5 | counter value GOOD | | 514...515 | | |
| 6 | code quality | | 518...521 | | |
| 7 | BAD value counter | | 516...517 | | |
| 8 | | | | | |
| 9 | current grey value | | 512...513 | | |
| 10 | software trigger | | | 512...513 | |
| 11 | | | | | |
| 12 | | | | | |
| 13 | | | | | |
| 14 | | | | | |
| 15 | | | | | |
| 16 | | | | | |
| 17 | result 64 byte | | 556...619 | | |

Figure 4.2 I/O addresses in the hardware configuration

2. Declare a new variable (e.g., in a global data module) as the UDT **I/O Address OPC_Modules** so that it can be transferred at the UDT input of the function block.



| Address | Name | Type | Initial val | Comment |
|---|---|---|---|---|
| 0.0 | | STRUCT | | |
| +0.0 | IO1 | "I/O-Address OPC_Modules" | | I/O modules for OPC120P |
| =26.0 | | END_STRUCT | | |

Figure 4.3 Variables declaration in global data module

3. You can then assign the I/O addresses to the UDT variable set up previously.

```
// Move address to DB - Scanner 1
      L     514
      T     "MyUDT".IO1.GoodRead_Counter
      L     518
      T     "MyUDT".IO1.ReadingQuality
      L     516
      T     "MyUDT".IO1.BadRead_Counter
      L     512
      T     "MyUDT".IO1.GreyScale
      L     512
      T     "MyUDT".IO1.SoftwareTrigger
      L     556
      T     "MyUDT".IO1.Result64Byte
```

Figure 4.4 Assigning the I/O addresses to the UDT variable

## 4.2 Scanning Data Matrix code

***Note!***

Evaluate **Busy** and **Done** before you start a new read command.

To start a read process, the input variable **Start** must be triggered. This variable reacts to a positive edge change. **Busy** then changes to HIGH. **Done** and **Error** change to LOW.

### GoodRead

After a successful read, **Busy** changes to LOW. **Done** and **GoodRead** change to HIGH. In addition, the output **GoodReads** increases by 1.

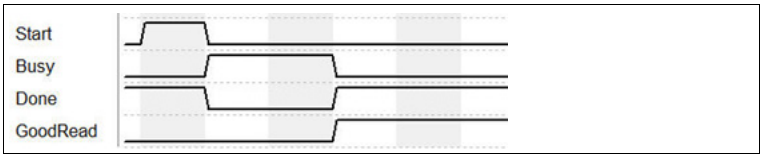



Figure 4.5 GoodRead signal curve

### BadRead

After a failed read, **Busy** changes to LOW. **Done**, **BadRead** and **Error** change to HIGH. In addition, the output **BadReads** increases by 1.

For an accurate error analysis, you can evaluate the output **ErrorStatus**.

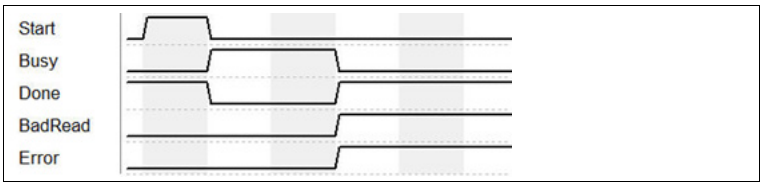



Figure 4.6 BadRead signal curve

# 5 Fault repair

| Fault pattern | Possible cause and remedy |
|---|---|
| Device does not respond to trigger command (no PLC error, no bus error) | Communication not initialized<br>■ Regenerate and reload the instance data block. |
| Inserting the function block causes PLC errors | Hardware configuration not consistent with function block circuitry<br>■ Check the input/output address and the length specified.<br>■ Check the PROFINET device name and the IP address. |
| Bus error during communication via PROFINET | Faulty hardware configuration<br>■ Check whether you are using only the **Result 64 byte** module.<br>■ Check the PROFINET device name and the IP address. |
| Function block status **Busy** is permanently HIGH | Consequence of a communication error<br>■ Regenerate and reload the instance data block. |
| **ErrorStatus** displays value **-1** | Timeoutoccurred<br>■ Check the connection between the PLC and the bus devices.<br>■ Check the device power supply. |

# FACTORY AUTOMATION – SENSING YOUR NEEDS


**Worldwide Headquarters**
Pepperl+Fuchs GmbH
68307 Mannheim · Germany
Tel. +49 621 776-0
E-mail: info@de.pepperl-fuchs.com

**USA Headquarters**
Pepperl+Fuchs Inc.
Twinsburg, Ohio 44087 · USA
Tel. +1 330 4253555
E-mail: sales@us.pepperl-fuchs.com

**Asia Pacific Headquarters**
Pepperl+Fuchs Pte Ltd.
Company Registration No. 199003130E
Singapore 139942
Tel. +65 67799091
E-mail: sales@sg.pepperl-fuchs.com


**www.pepperl-fuchs.com**

PEPPERL+FUCHS
SENSING YOUR NEEDS




TDOCT-3901AENG
12/2014