S7-1200 as a PN/DP Gateway

S7-1200 / TIA Portal / PROFINET-IO / PROFIBUS DP

Warranty and Liability

Note

The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are used correctly. These Application Examples do not relieve you of the responsibility to use safe practices in application, installation, operation and maintenance. When using these Application Examples, you recognize that we cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice. If there are any deviations between the recommendations provided in these Application Examples and other Siemens publications – e.g. Catalogs – the contents of the other documents have priority.

We do not accept any liability for the information contained in this document.

Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc., described in this Application Example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act ("Produkthaftungsgesetz"), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract ("wesentliche Vertragspflichten"). The damages for a breach of a substantial contractual obligation are, however, limited to the foreseeable damage, typical for the type of contract, except in the event of intent or gross negligence or injury to life, body or health. The above provisions do not imply a change of the burden of proof to your detriment.

Any form of duplication or distribution of these Application Examples or excerpts hereof is prohibited without the expressed consent of Siemens Industry Sector.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, solutions, machines, equipment and/or networks. They are important components in a holistic industrial security concept. With this in mind, Siemens’ products and solutions undergo continuous development. Siemens recommends strongly that you regularly check for product updates.

For the secure operation of Siemens products and solutions, it is necessary to take suitable preventive action (e.g. cell protection concept) and integrate each component into a holistic, state-of-the-art industrial security concept. Third-party products that may be in use should also be considered. For more information about industrial security, visit http://www.siemens.com/industrialsecurity.

To stay informed about product updates as they occur, sign up for a product-specific newsletter. For more information, visit http://support.automation.siemens.com.
# Table of Contents

- **Warranty and Liability** ........................................................................................................... 2
- **Task** ........................................................................................................................................ 4
- **Solution** ................................................................................................................................ 5
  - 2.1 Overview .......................................................................................................................... 5
  - 2.2 Hardware and software components ................................................................................. 6
  - 2.2.1 Validity ....................................................................................................................... 6
  - 2.2.2 Components used ....................................................................................................... 6
- **Basics** ..................................................................................................................................... 8
  - 3.1 I-device / I-slave communication ...................................................................................... 8
  - 3.1.1 S7 CPU as an IO device (I-device) ............................................................................... 8
  - 3.1.2 S7 CPU as a DP slave (I-slave) .................................................................................... 8
  - 3.2 Consistency, time and diagnostics ..................................................................................... 9
  - 3.2.1 Consistency ................................................................................................................ 9
  - 3.2.2 Signal propagation delay ......................................................................................... 10
- **Principle of Operation** ........................................................................................................... 12
  - 4.1 Complete overview ........................................................................................................... 12
  - 4.2 The bus mechanisms ....................................................................................................... 13
- **Configuration and Project Engineering** ............................................................................... 15
  - 5.1 Configuration as a "single project" .................................................................................. 16
  - 5.1.1 Configuring and networking the devices ..................................................................... 16
  - 5.1.2 Configuring the I-device ........................................................................................... 17
  - 5.1.3 Configuring the I-slave ............................................................................................. 18
  - 5.2 Configuration as a "PROFIBUS project" ....................................................................... 19
  - 5.2.1 "S7-1200" project ..................................................................................................... 19
  - 5.2.2 "S7-1500" project ..................................................................................................... 22
- **Installation and Startup** ......................................................................................................... 25
  - 6.1 Installing the hardware ...................................................................................................... 25
  - 6.2 Installing the software (download) .................................................................................... 26
  - 6.3 Startup ............................................................................................................................. 27
- **References** ............................................................................................................................ 28
- **History** .................................................................................................................................... 28
1 Task

Introduction

PROFINET has become standard in industrial automation. However, older systems often still use PROFIBUS.

Expanding older automation systems by new system parts or replacing existing system parts frequently involves problems with communication.

This application deals with communication between the old (PROFIBUS) and new world (PROFINET) or between older and newer system parts.

Overview of the automation task

A gateway that is part of both automation networks is required to establish a connection between PROFINET and PROFIBUS.

Figure 1-1
2 Solution

2.1 Overview

Diagrammatic representation

The diagrammatic representation below shows the most important components of the solution:

Configuration

An S7-1200 with a communication module (2) that acts as a gateway between the PROFINET and PROFINET system part is the core component of this application. An S7-300 (1) represents the PROFINET system part, an S7-1500 (3) represents the PROFINET system part.

Operating mode

The SIMATIC S7-1200 is additionally equipped with a communication module for PROFINET (CM 1242-5). This communication module can be addressed directly via the process image. No other aspects need be taken into consideration for communication via the communication module.

The S7-1200 is operated as follows:

- PROFIBUS DP slave as an I-slave.
- PROFINET IO device as an I-device.

Therefore, for both sides (S7-1500 and S7-300), the S7-1200 looks like a distributed IO device, which makes it unnecessary to program communication. This allows the two system parts to exchange data.
2 Solution

2.2 Hardware and software components

Advantages

The solution presented here offers the following advantages:

- Very cost-effective option to connect PROFIBUS DP to PROFINET IO.
- Very little configuration overhead.
- Very little programming overhead.
- Existing systems require hardly any changes.
- Additional automation tasks can be outsourced to the S7-1200 without modifying the existing systems.

Required knowledge

Good knowledge of STEP 7 V13 and PLC programming.

2.2 Hardware and software components

2.2.1 Validity

This application is valid for:

- STEP 7 V13 or higher
- S7-1200 V4.0 or higher
- S7-300, S7-400, S7-1500

2.2.2 Components used

This application was created with the following components:

Hardware components

Table 2-1

<table>
<thead>
<tr>
<th>Component</th>
<th>No.</th>
<th>Article no.</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU 1215C DC/DC/DC</td>
<td>1</td>
<td>6ES7215-1AG40-0XB0</td>
<td>Or other S7-1200 V4.0 or higher (I-device)</td>
</tr>
<tr>
<td>CM 1242-5</td>
<td>1</td>
<td>6GK7242-5DX30-0XE0</td>
<td>As a PROFIBUS DP slave</td>
</tr>
<tr>
<td>CPU 1511-1 PN</td>
<td>1</td>
<td>6ES7511-1AK00-0AB0</td>
<td>Or other PROFINET IO controller</td>
</tr>
<tr>
<td>CPU 315-2 DP</td>
<td>1</td>
<td>6ES7315-2AH14-0AB0</td>
<td>Or other PROFIBUS DP master</td>
</tr>
</tbody>
</table>

Software components

Table 2-2

<table>
<thead>
<tr>
<th>Component</th>
<th>No.</th>
<th>Article no.</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEP 7 Professional V13 Update 1</td>
<td>1</td>
<td>6ES7822-1...</td>
<td>Or more current version</td>
</tr>
</tbody>
</table>
2 Solution

2.2 Hardware and software components

Sample files and projects

The following list contains all files and projects that are used in this example.

Table 2-3

<table>
<thead>
<tr>
<th>Component</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>101495058_PN_DP_Gateway_CODE_v10_TIAP_V13_1.zip</td>
<td>This file contains the TIA Portal project.</td>
</tr>
<tr>
<td>101495058_S7-1200_PN_DP_Gateway_DOKU_v10_en.pdf</td>
<td>This document.</td>
</tr>
</tbody>
</table>
3 Basics

3.1 I-device / I-slave communication

Communication between the PROFINET IO controller and the I-device or the PROFIBUS DP master and the I-slave takes place via so-called transfer areas. The length, number, direction and addresses of these areas can be conveniently set in the hardware configuration of TIA Portal.

The individual parameters are:
- Number: Nearly any number of transfer areas can be created. Here, too, the limit is the quantity framework of the CPU.
- Addresses: The addresses can be freely assigned within the process image of the CPU.
- Direction: There are input and output transfer areas. In the partner CPU, an input transfer area corresponds to an output transfer area and vice versa.
- Length: The length of the transfer areas is set in bytes. The length is only limited by the quantity frameworks of the CPU and the bus system used.

3.1.1 S7 CPU as an IO device (I-device)

When an S7 CPU is used as an IO device (distributed IO device), this is referred to as an I-device (intelligent IO device). In this case, the CPU retains its intelligence but, "in the upward direction" - i.e. towards the PROFINET IO controller -, appears as a simple IO device with addresses in the process image. Therefore, it is not required to program communication. It is sufficient to configure the I-device communication in the hardware configuration.

3.1.2 S7 CPU as a DP slave (I-slave)

When an S7 CPU is used as a DP slave, this is referred to as an I-slave (intelligent DP slave). In this case, the CPU retains its intelligence but, "in the upward direction" - i.e. towards the PROFIBUS DP master -, appears as a simple DP slave with addresses in the process image. Therefore, it is not required to program communication. It is sufficient to configure communication in the hardware configuration.
3 Basics

3.2 Consistency, time and diagnostics

3.2 Consistency, time and diagnostics

3.2.1 Consistency

When data is exchanged between multiple CPUs, consistency of the transmitted data plays a special role. In this application example, even three CPUs exchange data with each other.

Transmission path

The data is transmitted via PROFINET IO and PROFIBUS DP. These field buses ensure data consistency within certain limits. When configuring normally using TIA Portal, these limits cannot be exceeded.

CPU-internal

In the "gateway CPU" (S7-1200), the process input image (the data) is loaded, processed and the process output image is written. Within the process image, the data is always consistent.

Conclusion

There is no risk of loss of consistency - neither on the transmission path nor within the gateway CPU.

3.2.2 Signal propagation delay

This solution is a high-performance option to exchange data across multiple bus systems. In this solution, the bottleneck is the S7-1200 with its cycle time.

Formula

The time the data requires across the two bus systems and the gateway CPU can be calculated using the following formula.

\[
\text{Signal propagation delay} = \text{cycle time}_{\text{PN CPU}} + \text{update time}_{\text{PN}} + \text{cycle time}_{\text{gateway CPU}} + \text{update time}_{\text{DP}} + \text{cycle time}_{\text{DP CPU}}
\]

Sample calculation

The following maximum times generally occur for the configuration described in this application:

<table>
<thead>
<tr>
<th>Title</th>
<th>Maximum time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cycle time (PN CPU)</td>
<td>2 ms</td>
</tr>
<tr>
<td>Update time (PN)</td>
<td>1 ms</td>
</tr>
<tr>
<td>Cycle time (gateway CPU)</td>
<td>5 ms</td>
</tr>
<tr>
<td>Update time (DP)</td>
<td>2 ms</td>
</tr>
<tr>
<td>Cycle time (DP CPU)</td>
<td>3 ms</td>
</tr>
<tr>
<td>Total signal propagation delay</td>
<td>13 ms</td>
</tr>
</tbody>
</table>
3 Basics

3.2 Consistency, time and diagnostics

**Note**

The formula considers only the times from "process image to process image". Regarding the times, you may have to consider the response times/update times of your modules or possibly configured distributed IO devices.

**Measured values**

A measurement with the hardware configuration described in this application produced measured values for the signal propagation delay between 7 ms and 10 ms. This means that the calculated worst-case values are not attained in reality.

**3.2.3 Diagnostics**

When three CPUs communicate across different bus systems, diagnostics are a demanding challenge. Diagnostics must be programmed individually for each use case; for this reason, this application does not contain program code for diagnostics.

The aim of this chapter is to give you a brief overview of things you need to consider and how you can proceed. Basically, the diagnostic mechanisms for PROFINET and PROFIBUS are easy to handle. In the case of this application, you cannot rely on the system behavior - the relevant diagnostic events have to be transferred via the user program. This requires modifications in the programs of all CPUs.

**Behavior of the bus systems**

The most frequent fault scenario is the failure of a CPU or bus. In this case, your plant has to respond to the data that is no longer supplied or possibly faulty. The response of the bus systems differs depending on the fault location. In some cases, the last value is retained; in other cases, a reset to zero takes place. If communication at this location is necessary for secure continued operation of the plant, such a failure must be reliably detected.

**Example**

1. The bus connection to the S7-1500 fails.
2. The S7-1200 reports an error but continues to supply (corrupt) data to the S7-300.
3. The S7-300 continues to run, it has not noticed the error. The data it receives is now only from the S7-1200 and no longer up to date.
3 Basics
3.2 Consistency, time and diagnostics

Solution

Figure 3-2

One solution to this is monitoring (2) a predictable auxiliary value (1). For example, monitoring a clock memory bit or a comparable predictable value.

In the case of the clock memory bit, the CPU knows that a signal edge can be expected within a known time. If this signal edge does not occur (4), it can be assumed that communication is faulty (3).

To ensure the check in both directions, the clock memory bit must be transmitted in both directions.

For the implementation, it is useful to configure another transfer area pair. Alternatively, when planning the application, you should provide for appropriate spare spaces when setting up the transfer areas.
4 Principle of Operation

4.1 Complete overview

Data exchange between the PROFIBUS CPU (S7-300) and the PROFINET CPU (S7-1500) takes place via the S7-1200. Therefore, the S7-1200 acts as a gateway and features both a PROFINET and a PROFIBUS interface.

The I-device (PROFINET) or I-slave (PROFIBUS) mechanism with its transfer areas is used to exchange data.

In the S7-1200, the user program ensures that the inputs are assigned to the appropriate outputs on the respective opposite side. In this application example, the clock memory byte is transferred in place of real data.

4.2 The bus mechanisms

Data exchange between the plants or the three CPUs used here takes place via the I-device or I-slave mechanism.

Programming is not required. You configure the transfer areas in the hardware configuration.

In this application, the S7-1200 simultaneously acts as an I-device and an I-slave.

The following table provides an overview of the input/output addresses of the transfer areas of the individual CPUs.

<table>
<thead>
<tr>
<th>Direction</th>
<th>S7-300 (DP)</th>
<th>S7-1200 (DP)</th>
<th>S7-1200 (PN)</th>
<th>S7-1500 (PN)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S7-300 - S7-1200 - S7-1500</td>
<td>Q 300..331</td>
<td>I 300..331</td>
<td>Q 150..181</td>
<td>I 150..181</td>
</tr>
<tr>
<td></td>
<td>Transfer area 1</td>
<td>Transfer area 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S7-1500 - S7-1200 - S7-300</td>
<td>I 300..331</td>
<td>Q 300..331</td>
<td>I 150..181</td>
<td>Q 150..181</td>
</tr>
<tr>
<td></td>
<td>Transfer area 2</td>
<td>Transfer area 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.3 The user program

The S7-300 and the S7-1500 do not have to be programmed in terms of communication. In the S7-1200, the respective inputs must be linked to the respective outputs in the user program.

The inputs are linked to the outputs in the "Transfer" function (FC). This function contains two identical statements, one for copying the inputs of the PROFIBUS side and one for the PROFINET side.

The data is copied using the "UMOVE_BLK" statement (Move block uninterruptible).

For an overview of the parameters of the statement, please refer to the TIA Portal online help.
Tag tables

Each CPU provides a tag table that allows you to monitor the simulated values/data.

Figure 4-4

Section from the tag table of the S7-1200. It contains a set of 128 tags:

- PN_IN_X (32 bytes, input via PN)
- PN_OUT_X (32 bytes, output via PN)
- PB_IN_X (32 bytes, input via PB)
- PB_OUT_X (32 bytes, output via PB)

Each tag table of the other CPUs contains the clock memory byte and a set of 64 additional tags (only PB/PN).

Note

This application only describes setting up communication.

The data to be exchanged using the described mechanism must be written at the end points, i.e. within the user program of the S7-1500/S7-300.
This chapter tells you how to configure the individual CPUs in order to exchange data across PROFINET and PROFIBUS using an S7-1200 as a gateway.

The following configuration options are available:

Table 5-1

<table>
<thead>
<tr>
<th>No.</th>
<th>Option</th>
<th>Description</th>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Single project</td>
<td>All CPUs are included in a single project.</td>
<td>Easy handling. I-device/I-slave can be directly loaded.</td>
<td>Existing projects have to be integrated.</td>
</tr>
<tr>
<td>2</td>
<td>PROFIBUS project</td>
<td>All PROFIBUS CPUs in a single project.</td>
<td>Only the I-device GSD file has to be integrated into the project of an existing PROFINET system.</td>
<td>The GSD file for the I-device has to be generated.</td>
</tr>
</tbody>
</table>

From the configuration options described in the following, you can select the one best suited for you.

The sample project associated with this application example was configured according to configuration option 1.
5.1 Configuration as a "single project"

5.1.1 Configuring and networking the devices

1. Open TIA Portal and create a new project with a name of your choice.
2. Add the following CPUs to your project and assign CPU names:

<table>
<thead>
<tr>
<th>TYPE</th>
<th>Version</th>
<th>Article no.</th>
<th>CPU name</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU 1215C DC/DC/DC V4.0</td>
<td>6ES7215-1AG40-0XB0</td>
<td>S7-1200</td>
<td></td>
</tr>
<tr>
<td>CPU 1511-1 PN V1.5</td>
<td>6ES7511-1AK00-0AB0</td>
<td>S7-1500</td>
<td></td>
</tr>
<tr>
<td>CPU 315-2 DP V3.3</td>
<td>6ES7315-2AH14-0AB0</td>
<td>S7-300</td>
<td></td>
</tr>
</tbody>
</table>

You are free to select any names you desire, the application example uses the names listed in the table.

3. Enable the clock memory byte for the S7-300 and S7-1500. Enter address "0" for the clock memory byte.
4. Open the device configuration of the S7-1200.
5. Configure a CM 1242-5 communication module (V1.0).
6. Open the network view and connect the individual CPUs. To do this, click the PROFINET interface on the S7-1200 and, while holding down the mouse button, drag a line to the PROFINET interface of the S7-1500.
7. Repeat this process for the PROFIBUS interface of the S7-1200; this time, drag the line to the PROFINET interface of the S7-300.

Result

You have successfully configured and networked the components. PROFIBUS addresses were automatically assigned and PROFINET device names were generated from the CPU names.
5.1.2 Configuring the I-device

1. Open the device configuration of the S7-1200.
2. Navigate to "PROFINET interface > Operating mode".
3. Check the "IO device" check box to activate the I-device functionality.
4. Use the "Assigned IO controller" drop-down list to assign an IO controller to the I-device (in this case, S7-1500.PROFINET interface_1).
5. In the "I-device communication" section, add two transfer areas: To do this, click "Add new" for each transfer area.
6. Now customize the transfer area settings as follows:

![Configuration Screenshot](image)

The example shows an input transfer area and an output transfer area with a length of 32 bytes. In each case, the start address is 150.

You can freely select the names, length and start addresses of the transfer areas.

**Note**

**Addresses in the S7-1500**

The settings you make in the "Address in IO controller" column are the addresses used in the process image of the S7-1500. These addresses can only be set here.

**Result**

You have successfully configured the I-device communication, i.e. the PROFINET part, between the S7-1500 and the S7-1200.
5 Configuration and Project Engineering

5.1 Configuration as a "single project"

5.1.3 Configuring the I-slave

1. Open the device configuration of the CM 1242-5.
2. Navigate to "DP interface > Operating mode".
3. The CM 1245-5 can only be operated as a DP slave. Therefore, the assignment to the DP master has already been made automatically during networking.
4. In the "I-slave communication" section, add two transfer areas: To do this, click "Add new" for each transfer area.
5. Now customize the transfer area settings as follows:

The example shows an input transfer area and an output transfer area with a length of 32 bytes. In each case, the start address is 300.

You can freely select the names, length and start addresses of the transfer areas.

Result

You have successfully configured the I-slave communication, i.e. the PROFIBUS part, between the S7-300 and the S7-1200.

In the S7-1200, you now have to load the user program for data exchange.

Note

Process image of the S7-300

For the address configuration selected in this application example to work, you have to expand the process image of the S7-300 so that the highest address (331) is still in the process image.

To do this, open the "Cycle" property in the device configuration of the S7-300 and increase the "process input image/process output image" to a value greater than or equal to the highest address of your configuration (in the example: 331).
5.2 Configuration as a "PROFIBUS project"

The configuration as a "PROFIBUS project" involves the creation of two projects; in each of these projects, you define the interfaces to the other project. In the project of the S7-1500, the S7-1200 acts as a distributed IO device that is integrated as a GSDML file.

5.2.1 "S7-1200" project

Configuring the S7-1200

1. Open TIA Portal and create a new project with a name of your choice.
2. Add the following CPUs to your project and assign CPU names:

<table>
<thead>
<tr>
<th>TYPE</th>
<th>Version</th>
<th>Article no.</th>
<th>CPU name</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU 1215C DC/DC/DC</td>
<td>V4.0</td>
<td>6ES7215-1AG40-0XB0</td>
<td>S7-1200</td>
</tr>
<tr>
<td>CPU 315-2 DP</td>
<td>V3.3</td>
<td>6ES7315-2AH14-0AB0</td>
<td>S7-300</td>
</tr>
</tbody>
</table>

You are free to select any names you desire, the application example uses the name listed in the table.

3. Enable the clock memory byte for the S7-300. Enter address "0" for the clock memory byte.
4. Open the device configuration of the S7-1200.
5. Configure a CM 1242-5 communication module (V1.0).
6. Open the network view and establish the PROFIBUS connection between the CPUs. To do this, click the PROFIBUS interface on the S7-1200 and, while holding down the mouse button, drag a line to the PROFIBUS interface of the S7-300.

**Figure 5-4**

**Result**

You have successfully configured and networked the components, PROFIBUS addresses have been automatically assigned.
5 Configuration and Project Engineering

5.2 Configuration as a "PROFIBUS project"

**Configuring the I-slave**

1. Open the device configuration of the CM 1242-5.
2. Navigate to "DP interface > Operating mode".
3. The CM 1245-5 can only be operated as a DP slave. Therefore, the assignment to the DP master has already been made automatically during networking (previous step).
4. In the "I-slave communication" section, add two transfer areas: To do this, click "Add new" for each transfer area.
5. Now customize the transfer area settings as follows:

![Diagram](image)

The example shows an input transfer area and an output transfer area with a length of 32 bytes. In each case, the start address is 300.

You can freely select the names, length and start addresses of the transfer areas.

**Result**

You have successfully configured the I-slave communication, i.e. the PROFIBUS part, between the S7-300 and the S7-1200.

**Note**

**Addresses in the S7-300**

The settings you make in the "Master address" column are the addresses used in the process image of the S7-300. These addresses can only be set here.

**Note**

**Process image of the S7-300**

For the address configuration selected in this application example to work, you have to expand the process image of the S7-300 so that the highest address (331) is still in the process image.

To do this, open the "Cycle" property in the device configuration of the S7-300 and increase the "process input image/process output image" to a value greater than or equal to the highest address of your configuration (in the example: 331).
5 Configuration and Project Engineering

5.2 Configuration as a "PROFIBUS project"

Configuring the I-device and generating the GSDML file

1. Open the device configuration of the S7-1200.
2. Navigate to "PROFINET interface > Operating mode".
3. Check the "IO device" check box to activate the I-device functionality.
4. Use the "Assigned IO controller" drop-down list to assign no IO controller to the I-device ("Not assigned").
5. In the "I-device communication" section, add two transfer areas: To do this, click "Add new" for each transfer area.
6. Now customize the transfer area settings as follows:

Figure 5-6

The example shows an input transfer area and an output transfer area with a length of 32 bytes. In each case, the start address is 150.

You can freely select the names, length and start addresses of the transfer areas.
5 Configuration and Project Engineering

5.2 Configuration as a "PROFIBUS project"

7. Click the "Export" button to export the generic station description file (GSD).

![Figure 5-7](image)

8. A dialog box opens. Select a name and storage location and use the "Export" button to confirm the settings.

Result

You have successfully configured the GSD file for the I-device communication.

5.2.2 "S7-1500" project

1. Open TIA Portal and create a new project with a name of your choice.
2. Add the following CPU to your project and assign a CPU name:

<table>
<thead>
<tr>
<th>TYPE</th>
<th>Version</th>
<th>Article no.</th>
<th>CPU name</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU 1511-1 PN</td>
<td>V1.5</td>
<td>6ES7511-1AK00-0AB0</td>
<td>S7-1500</td>
</tr>
</tbody>
</table>

3. Enable the clock memory byte and enter address "0" for it.
4. Click "Options > Install general station description file (GSD)".
5. Navigate to the storage location of the previously generated GSD file and check the check box of the appropriate file. Select "OK" to exit the dialog. If necessary, confirm the information in the following dialog boxes.
6. Open the device configuration and go to the network view.
5.2 Configuration as a "PROFIBUS project"

7. Open the hardware catalog and navigate to the just imported GSD file.

Figure 5-8
8. Add the device to your project.
9. Establish the PROFIBUS connection between the CPU and the "DP standard slave" (the "GSD IO device"). To do this, click the PROFINET interface on the S7-1500 and, while holding down the mouse button, drag a line to the PROFINET interface of the DP standard slave.

Figure 5-9

10. Now use the "Network overview" task card to configure the IO addresses of the transfer areas in the "s7-1200" section.

Figure 5-10

Result
You have successfully configured the I-device communication between the S7-1200 and the S7-1500.
**6 Installation and Startup**

**6.1 Installing the hardware**

The figure below shows the hardware configuration of the application.

1. Following the manual's instructions, mount the devices onto the appropriate DIN rails.
2. Connect the components via PROFIBUS/PROFINET.
   a. Connect the DP interface (X2) of the S7-300 to the PROFIBUS interface of the CM 1242.
   b. Connect the PN interface (X1P1) of the S7-1200 to the PN interface (X1P1 R) of the S7-1500.
3. Connect the PG to the hardware.
   a. Connect the MPI (X1) of the S7-300 to the MPI/DP interface of the PG.
   b. Connect the PN interface (X1P2 R) of the S7-1500 to a PN interface of the PG (the connection to the S7-1200 is established via the integrated switch of the S7-1500).
4. Supply power to the hardware.
6.2 Installing the software (download)

The project associated with this application example contains:
- The hardware configuration as described in Configuration as a "single project".
- The user programs of the individual CPUs.

Contents of the user programs:
- The user programs of the S7-1500/S7-300 contain only blocks for simulating communication and catching bus errors.
- The user program of the S7-1200 contains the gateway block in which the gateway functionality has been programmed.

You can either download the entire project directly to the just installed hardware or copy only the required blocks or block calls, data types and tag tables to your own project.

Required objects:

**S7-1200**
- Program block – Main[OB1] (falls vorhanden darin den FC 1 aufrufen)
- Program block – Transfer[FC1]
- Tag table – TransferArea
- Data type – TransferArea

**S7-300**
- Program block – Main[OB1]
- If necessary, program block – I/O_FLT1[OB82]
- If necessary, program block – RACK_FLT[OB86]

**S7-1500**
- Program block – Simulation[OB123]

---

### Address overview

Table 6-1

<table>
<thead>
<tr>
<th>Component</th>
<th>Connection</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>PG</td>
<td>PROFINET (S7-1500)</td>
<td>192.168.0.1</td>
</tr>
<tr>
<td>PG</td>
<td>MPI (S7-300)</td>
<td>1</td>
</tr>
<tr>
<td>S7-1200</td>
<td>PROFINET (S7-1500)</td>
<td>192.168.0.2</td>
</tr>
<tr>
<td>S7-1200</td>
<td>PROFIBUS (S7-300)</td>
<td>2</td>
</tr>
<tr>
<td>S7-1500</td>
<td>PROFINET (S7-1200 / PG)</td>
<td>192.168.0.3</td>
</tr>
<tr>
<td>S7-300</td>
<td>PROFIBUS (S7-1200)</td>
<td>1</td>
</tr>
<tr>
<td>S7-300</td>
<td>MPI (PG)</td>
<td>2</td>
</tr>
</tbody>
</table>
6.3 Startup

1. Start the CPUs.
2. The watch tables allow you to follow the transmission of the values:

The figure shows the data transfer in the direction S7-1500 > S7-1200 > S7-300:

The left screenshot shows the values of the clock memory byte ("Clock_Byte") of the S7-1500 that are copied to the output addresses of the transfer area ("PN_OUT_X").

The center-left screenshot is the counterpart to the just viewed transfer area; this time from the perspective of the S7-1200 ("PN_IN_X").

The center-right screenshot shows the values that have been filled, this time in the direction of the S7-1200, into the next transfer area ("PB_OUT_X").

The right screenshot again shows the counterpart to the just viewed transfer area; this time within the S7-300 ("PB_IN_X").
7 References

Table 7-1

<table>
<thead>
<tr>
<th>Subject</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>\2\ Download page of this entry</td>
<td><a href="http://support.automation.siemens.com/WW/view/en/101495058">http://support.automation.siemens.com/WW/view/en/101495058</a></td>
</tr>
</tbody>
</table>

8 History

Table 8-1

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Modifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>V1.0</td>
<td>09/2014</td>
<td>First version</td>
</tr>
</tbody>
</table>