SIEMENS

Application Example • 03/2017

Multiple Use IO System with Configuration Control for IO Systems

S7-1500, PRONETA, SIMATIC Automation Tool

https://support.industry.siemens.com/cs/ww/en/view/29430270

Warranty and Liability

Note

The Application Examples are not binding and do not claim to be complete with regard to configuration, equipment or any contingencies. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for the correct operation of the described products. These Application Examples do not relieve you of the responsibility of safely and professionally using, installing, operating and servicing the equipment. 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 and without prior notice. If there are any deviations between the recommendations provided in this Application Example and other Siemens publications – e. g. catalogues – the contents of the other documents shall have priority.

We assume no responsibility for information contained in the present 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 fundamental contractual obligations (“wesentliche Vertragspflichten”). The compensation for damages due to a breach of a fundamental contractual obligation is, 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 in 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 AG.

Security information

Siemens provides products and solutions with Industrial Security functions that support the secure operation of plants, systems, machines, and networks.

In order to secure plants, systems, machines, and networks against cyber-threats it is necessary to implement (and to maintain continuously) a holistic, state-of-the-art Industrial Security concept. With this in mind, Siemens’ products and solutions are only part of such a concept.

It is the client’s responsibility to prevent unauthorized access to his plants, systems, machines, and networks. Systems, machines, and components should only be connected with the company’s network or the Internet, when and insofar as this is required and the appropriate protective measures (for example, use of firewalls and network segmentation) have been taken.

In addition, Siemens’ recommendations regarding appropriate protective action should be followed. For further information on Industrial Security, please visit http://www.siemens.com/industrialsecurity.

Siemens’ products and solutions undergo continuous development to make them even more secure. Siemens explicitly recommends to carry out updates as soon as the respective updates are available and always only to use the current product versions. The use of product versions that are no longer supported, and failure to apply latest updates may increase customer’s exposure to cyber-threats.

To stay informed about product updates as they occur, subscribe to the Siemens Industrial Security RSS feed at http://www.siemens.com/industrialsecurity.

Table of Contents

Warranty and Liability **2**

Introduction **4**

1 Task **5**

2 Solution **6**

2.1 Overview 6
2.2 Hardware and software components 9
2.2.1 Validity 9
2.2.2 Components used 9

3 Principles **10**

3.1 Multiple Use IO system 10
3.2 Configuration Control for the IO System 10

4 Operation principle **11**

4.1 General overview 11
4.2 Functionality of the “Multiple Use IO system” 13
4.2.1 Overview 13
4.2.2 Program details of system block “T_CONFIG“ 14
4.2.3 Structure of the “CONF_DATA” data block 15
4.3 Functionality of the Configuration Control within the IO system 17
4.3.1 Overview 17
4.3.2 Program details of the “ConfigIOSys” function block 18
4.3.3 Structure of the control data record “CtrlRec” 19

5 Configuration and Engineering **21**

5.1 Engineering of the Configuration Control for IO systems 21
5.2 Configuring a “Multiple Use IO System” 23

6 Installation / Commissioning **27**

6.1 Installing the hardware 27
6.2 Installing the software 27
6.3 Commissioning 28

7 Links & Literature **31**

8 History **31**

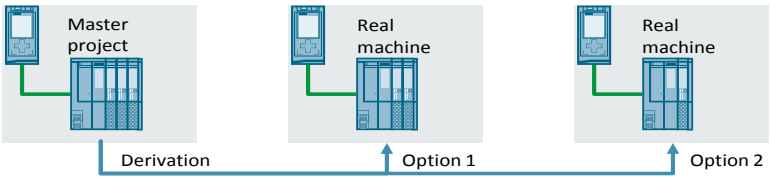
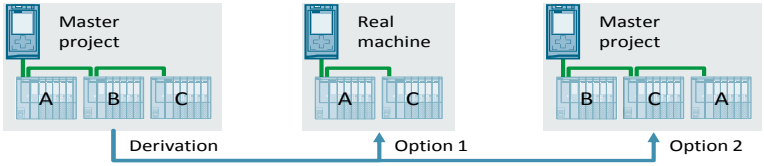
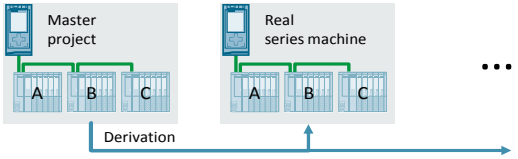
Introduction

The present engineering example is part of the documents dealing with the variants management with SIMATIC S7. This chapter gives you an overview of several applications.

Definition of the term “Variants Management”

“Variants Management” is a generic term for an innovative machine concept in the series production of modular machines that may be adapted to customer requirements as easy as possible. Yet this presupposes the possibilities to be just as flexible. An addressing of all the plant and machine parts performed at a time must be adaptable without great effort and changes in the engineering project.

The table below shows the fields of application of the variants management:

Term	Explanation of the fields of application	In this DOC
Module-level Configuration Control	The module-level Configuration Control allows for flexible configuration levels of the distributed and centralised I/O systems within a project. Thus a single STEP 7 project (maximum configuration) may be used for multiple configuration levels of I/O stations. 	✗
Configuration Control for IO systems	The Configuration Control for IO systems makes flexible configuration levels and interconnections of stations within an IO system possible. Thus a single STEP 7 project may be used for several concrete IO system versions as long as they can be derived from the maximum configuration. 	✓
Multiple Use IO systems	“Multiple Use IO systems” means that a single IO system is used for several machines. Thus a PROFINET IO system created in a STEP 7 project may be used for several machines as the IP addresses and the device names are fixed by the IO controller and not directly by the STEP 7 project. 	✓

1 Task

Description

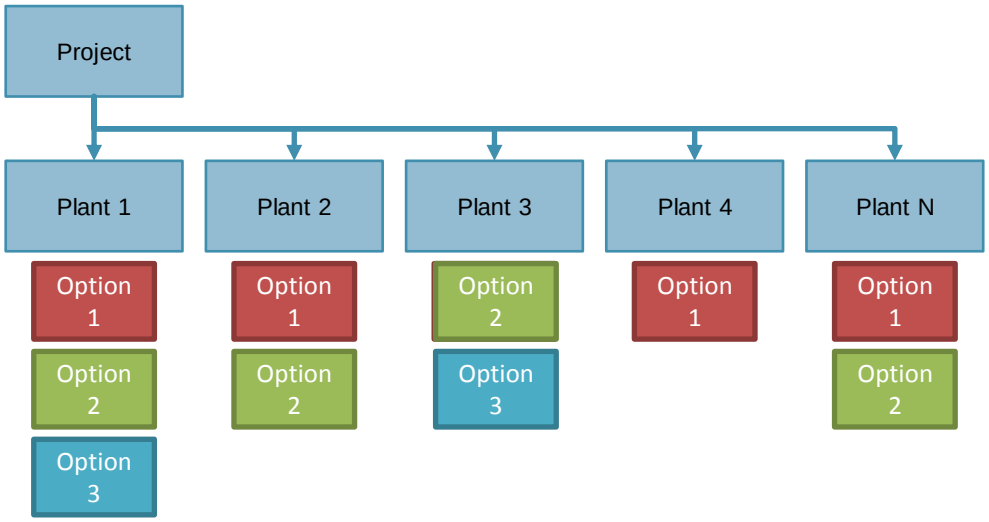
In the series machine production, it is common practice that the PROFINET IO system of a machine (consisting of an IO controller and the IO devices assigned to it) is used in identical or different configuration levels in several plants or even within the same automation network. The diverse systems are different only in their network address, their device name, and the related configuration level.

This is the reason why every machine required its own engineering project until now although the automation components were identical in all of them. Thus a lot of time and money has been spent for engineering and commissioning, added to the lack of flexibility.

Overview of the automation task

The figure below provides an overview of the automation task.

Figure 1-1



Requirements

Based on typical application cases, the automation solution must meet the following requirements:

- One project (configuration and program) shall be loadable without any change, or in several configuration levels to several machines of the same type.
- To connect the machine to an existing network infrastructure, the on-site commissioning shall only require some minor adjustments.
- The on-site commissioning shall also be possible without TIA Portal.

2 Solution

2.1 Overview

Description

The combined use of the functions “Multiple Use IO System” and “Configuration Control for IO Systems” provides a flexible automation solution in the series machine production where the individual machines have different configuration levels.

By setting the IO system to “Multiple Use IO System”, the STEP 7 project becomes a “Series Machine Project”. Combined with the Configuration Control for IO systems, different machines with several, concrete specifications may arise from this project.

In this setting, the IO controller can assign an IP address and a PROFINET device name to all the IO devices attributed to it. The Configuration Control for IO systems allows the IO system configuration to flexibly vary for a determined application as long as the real configuration can be derived from the engineered one.

Using the “Multiple Use IO System” and the Configuration Control for IO systems side-by-side, a single project may cover several automation systems with different configuration levels.

Advantages

A “Multiple Use IO System” has the following advantages:

- A single project for several machines with identical set-up.
- Less adjustments during the on-site commissioning (IP address, device name)
- No programming unit with STEP 7 needed for commissioning; commissioning can also be done with tools like SIMATIC Automation Tool and PRONETA.

Topics not covered in this application

This application does not include a description of the following:

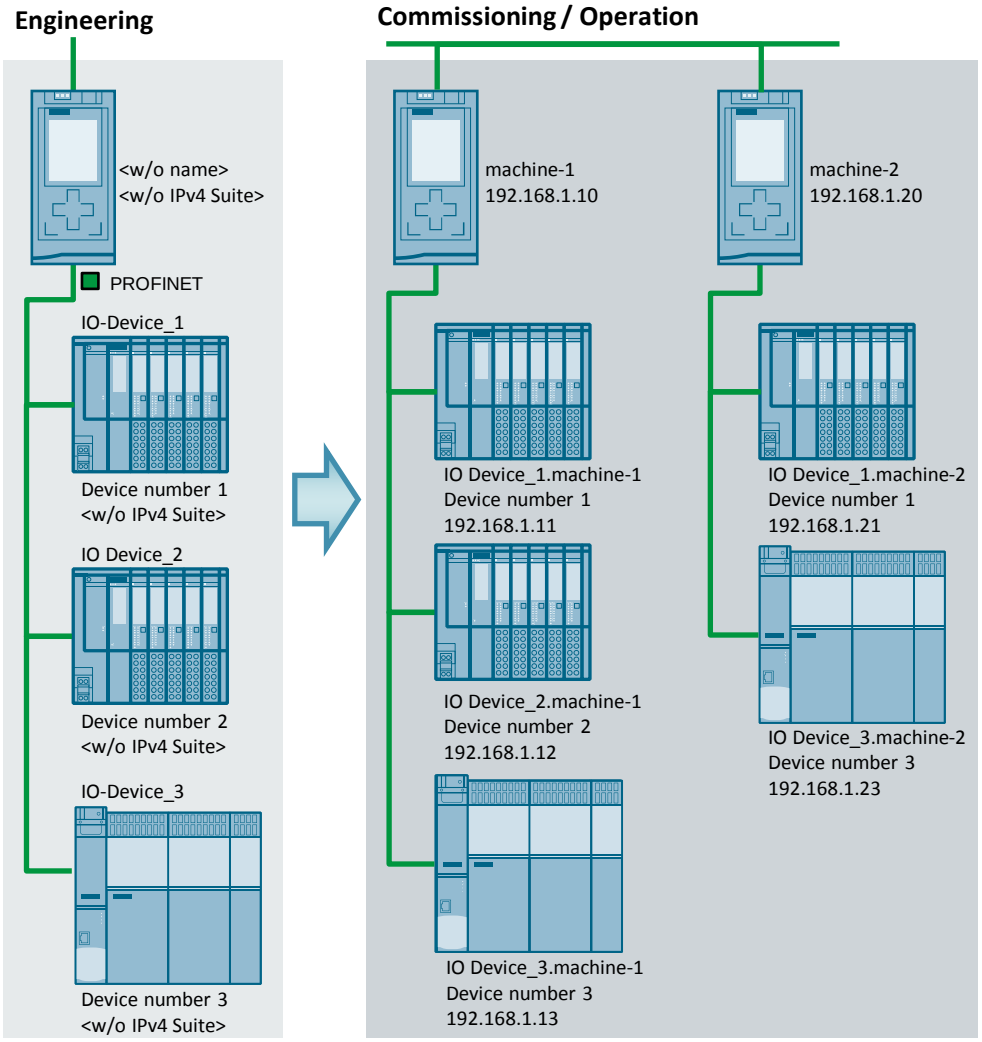
- PLCs (particularly S7-1500)
- STEP 7 V13 SP1, FUP, SCL

Basic knowledge of these topics is assumed.

Schematic diagram

The Figure below shows the principle of a “Multiple Use IO System” with Configuration Control for IO Systems as a graph.

Figure 2-1



Master Project:

- A single project
- Controller: Flexible address
- Devices: Addresses adaptable to controller

Maschine modules:

- Controller: Address assigned by external tool (SAT, PRONETA) or application program
- Devices: Addresses adapted from controller address

A “Multiple Use IO System” enables you to commission with a single project several machine modules of the same type with different network parameters. These “Multiple Use IO Systems” can be adapted to different application cases by means of optional IO devices and specified port interconnections using the Configuration Control for IO systems. The use of these two functions essentially reduces the engineering effort and the length of the commissioning for such kind of plants.

Process overview of the core functionality

The Figure below describes the process starting with the parametrisation of the data blocks in the “STOP” condition, over the assignment of IP addresses and device names until the ready PROFINET IO system in the “RUN” condition of the IO controller.

Figure 2-2

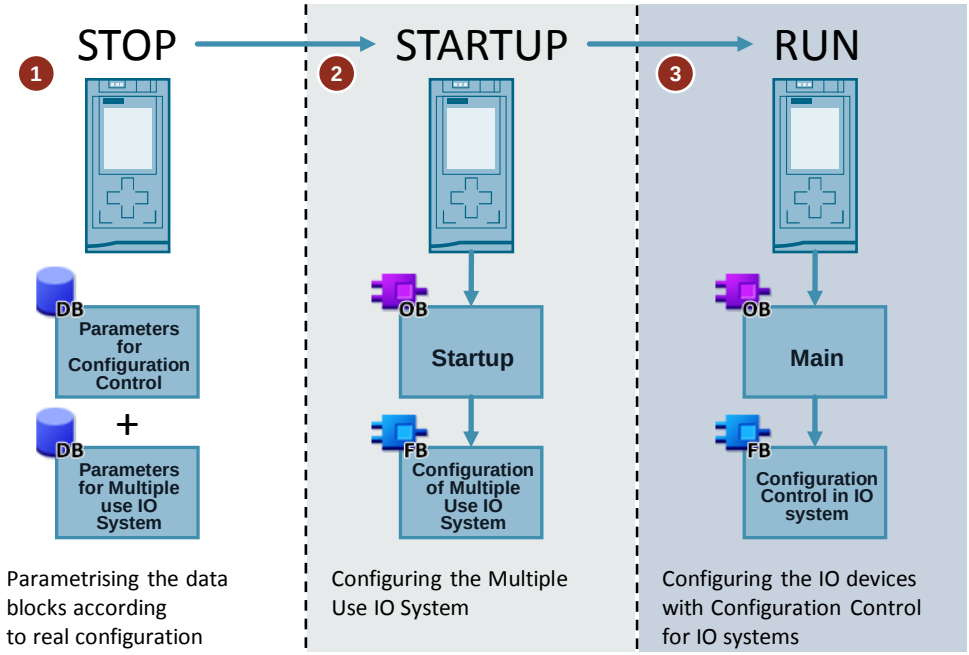


Table 2-1

No.	Action	Note
1.	The parameter data blocks are parametrised with the configuration used in the real machine.	The CPU is in “STOP” condition. The user himself must do the parametrisation.
2.	The start-up OB 100 is executed and calls up the system block which assigns the IP addresses and device names to the IO devices and the IO controller.	The CPU is in “STARTUP” condition.
3.	The “Main” OB 1 is executed and calls up the function block which activates the IO devices as established in step 1.	The CPU is in RUN condition.

2.2 Hardware and software components

2.2.1 Validity

This application is valid for

- STEP 7 as of V13 SP1 or higher
- S7-1500
- ET 200SP

2.2.2 Components used

This application was created using the following components:

Hardware components

Table 2-2

Component	Qty	Article Code	Note
CPU 1516-3 PN/DP	1	6ES7516-3AN01-0AB0	Alternatively, all the S7-1500 CPUs can also be used.
ET 200SP IM155-6PN ST	3	6ES7155-6AU00-0BN0	Alternatively, IO devices with PROFINET interface can also be used. The module set-up of the interface module is not relevant for the function shown.

Software components

Table 2-3

Component	Qty	Note
STEP 7 PROFESSIONAL V14	6ES7822-1AA04-0YA5	Alternatively, a smaller package is also possible.

Example files and projects

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

Table 2-4

Component	Note
29430270_MultipleIOSystemsKonfig _PROJ_v20.zip	This zip file contains the STEP 7 project.
29430270_MulitIOSysKonfig _DOCU_v20_de.pdf	The present document.

3 Principles

3.1 Multiple Use IO system

Description

"Multiple Use IO system" means that a single IO system is used for several machines. Thus a PROFINET IO system created in a STEP 7 project may be used for several machines since the IP addresses and the device names are fixed by the IO controller and not directly by the STEP 7 project.

When should I use a "Multiple Use IO System"?

- For commissioning of several machines with the same set-up.
- For commissioning without TIA Portal.
  The IP addresses and the device name can be defined on site using one of the Engineering Tools (PRONETA, SIMATIC Automation Tool), or directly at the device.

Note For further information on the "Multiple Use IO System", please refer to the entry sheet (2) in the "Multiple Use IO System" document. The function may also be used with IRT.

3.2 Configuration Control for the IO System

Description

The Configuration Control for the IO System is used to extend, upgrade, or modify complete stations without any further engineering effort. Using the target IP address at the S7-1500, lower-level PROFINET devices are at the same time automatically assigned the correct network parameters (automatic address adaption).
The decentralised Configuration Control enables you to prepare the set-up of an IO device for future extensions and modifications. Thus complex automation projects must be configured once only for the maximum configuration level. Afterwards you are free to define several variants and chose them to meet flexibility demands.

When should I use the Configuration Control in the IO System?

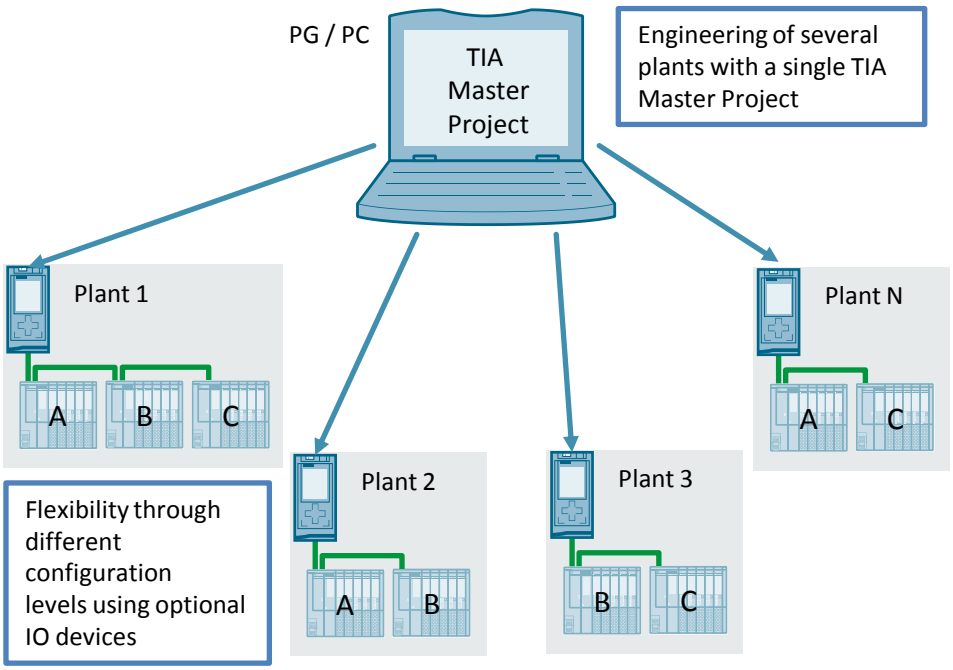
- Commissioning of several plants with different IO devices.
- For more flexibility in the plant set-up and allowing adaptions with the help of the application program.
- Commissioning of an incomplete plant planned to be extended at a later moment.

Note For further information on the Configuration Control for the IO system, please refer to the entry sheet (2) in the "Configuration Control in the IO System" document.
The function may also be used with IRT.

4 Operation principle

4.1 General overview

Figure 4-1

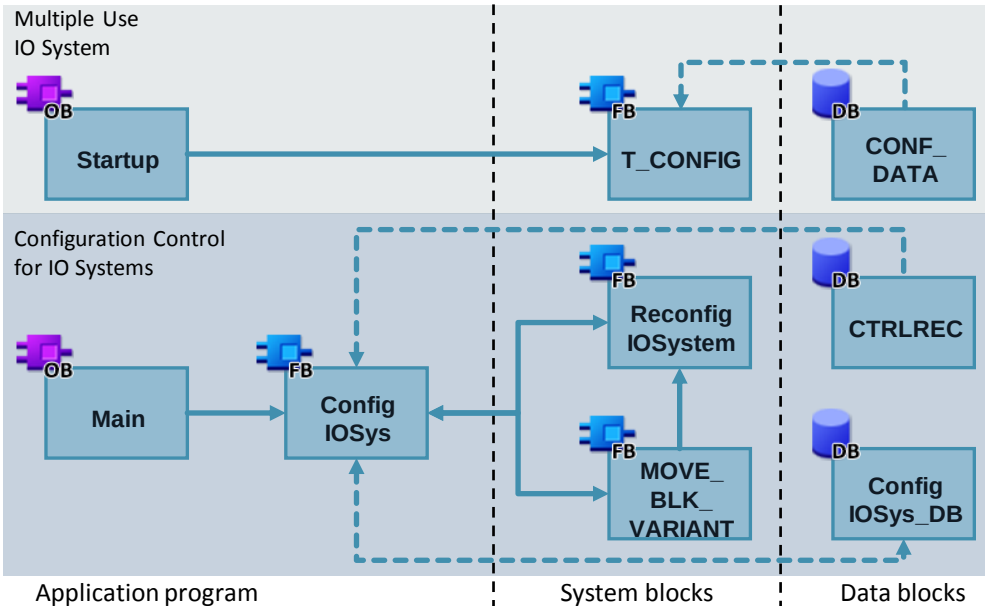


The application example serves as the Master Project on the basis of which N plants in different configuration levels can be created. This is done using the functions “Multiple Use IO System” and Configuration Control in the IO system.

Program overview

The Figure below shows the call structure of the application example.

Figure 4-2



Description

The present application example enables you to commission several plants in different configuration levels. It serves as the Master project for the plants involved and combines the functions “Multiple Use IO system” with “Configuration Control for IO Systems”.

For the “Multiple Use IO System”, the system block “T_CONFIG” is called up in OB 100 “Startup”. This block assigns to the IO controller the IP address and the device name communicated by the data block „CONF_DATA“.

The Configuration Control for IO systems is managed using the function block „ConfigIOSys“. The data block „CtrlRec“ provides the function block „ConfigIOSys“ with information on optional IO devices and specified port interconnections. The block performs the modes of the system data record “ReconfigIOSystem“. The IO system is configured using the information from the data block “CtrlRec”.

Block list

The table below gives you an overview of the program blocks of the present application example:

Table 4-1

Element	Symbolic name	Description
Multiple Use IO system		
OB100	Startup	Start-up OB: The start-up OB is processed once when the CPU changes from the STOP to the RUN mode. The system block “T_CONFIG“ is called up here.
DB100	T_CONFIG_DB	Instance DB for the “T_CONFIG“ instruction.
DB110	CONF_DATA	Includes the IO controller’s IP address and the device name of the “Multiple Use IO System”.
Configuration Control for IO systems		
OB1	Main	Cyclic OB: Call of function block “ConfigIOSys“.
FB50	ConfigIOSys	Configures the IO devices in the IO system following the information from DB51 “CtrlRec“ by performing the three modes of the “ReconfigIOSystem” system block.
DB50	ConfigIOSys_DB	Instance DB of FB50 „ConfigIOSys“
DB51	CtrlRec	Control data record for the Configuration Control. Includes the data for the IO system to be configured.

4.2 Functionality of the “Multiple Use IO system”

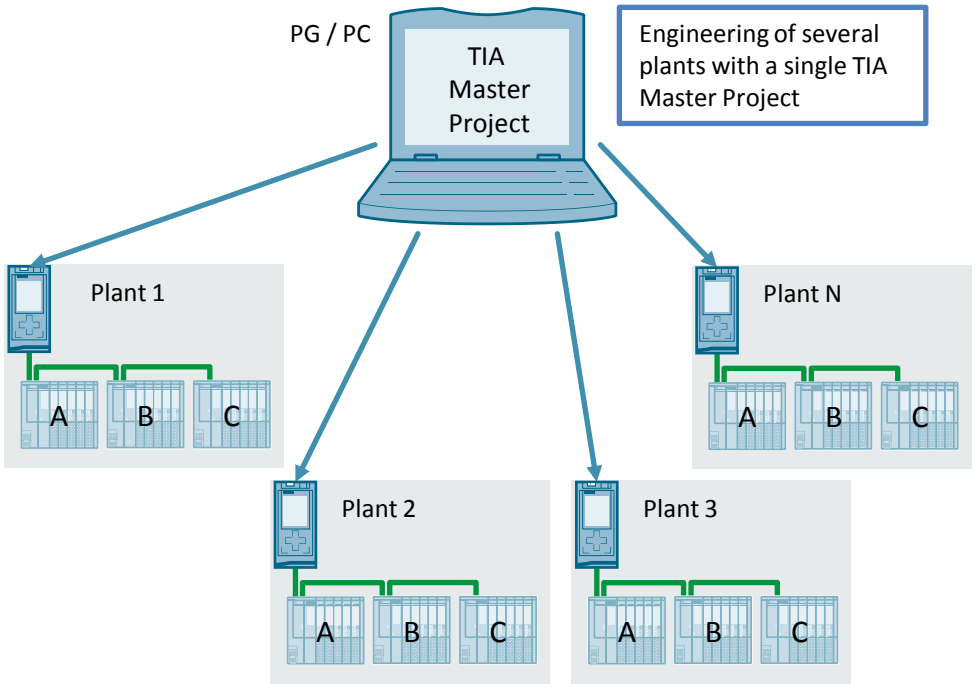
4.2.1 Overview

Thanks to the “Multiple Use IO System”, you can load one and the same project to several S7-1500 CPUs within the present application example. Afterwards, you can set for every CPU the related IP address and the device name.

The CPU then automatically starts, based on the topology engineered, assigning a unique device name as well as an IP address to any of the devices it is allocated to, starting from it’s own IP address.

The Figure below shows the “Multiiple Use IO System” function.

Figure 4-3



Call of “T_CONFIG“ in “Startup“ OB 100

In the application program, system block “T_CONFIG“ is called up in a loop in the “Startup” OB 100 as long as the configuration of the IO controller is successfully completed (output parameter “Done”).

Figure 4-4

```
1 WHILE NOT "T_CONFIG_DB".Done DO
2     "T_CONFIG_DB"(Req := true,
3                   Interface :="Local~PROFINET_interface_1",
4                   Conf_Data := "CONF_DATA".ConfData);
5 END_WHILE;
```

4.2.2 Program details of system block “T_CONFIG“

The block “T_CONFIG“ is used to assign the device name and the IP address to the IO controller. As a result, the IO controller assigns the IP address and the device name to the IO devices allocated to it.

General

The “T_CONFIG“ instruction is used for the program-controlled configuration of the integrated PROFINET interface of the CPU or of a CP/CM, respectively.

The following settings can be done using the “T_CONFIG“ instruction:

- IP protocol settings
  - IP address
  - Subnet mask
  - Router address
- PROFINET settings
  - PROFINET device name

The table below shows the input parameters of the “T_CONFIG“ block.

Table 4-2

Parameters	Data type	Description
REQ	Boolean	Edge-triggered control parameter
INTERFACE	HW_Interface	HW flag of the PROFINET interface to be configured
CONF_DATA	Variant	Pointer to the configuration data record of the “Multiple Use IO system”

The table below shows the output parameters of the “T_CONFIG“ block.

Table 4-3

Parameters	Data type	Description
DONE	Boolean	TRUE as soon as the instruction is completed.
BUSY	Boolean	TRUE if the instruction is active.
ERROR	Boolean	TRUE if the instruction is completed with error.
STATUS	DWord	Status of the instruction.
ERR_LOC	DWord	Information on the error location: 0: Error during execution or parametrisation. >0: Structure or content error in „CONF_DATA“

Mode of Operation

“T_CONFIG“ is an instruction working in asynchronous mode. The complete processing of the task covers several cycles. However, only one task can be active at any time.

The task starts as soon as a positive edge is acquired at input “REQ”.

The output parameters “STATUS“, “BUSY“, “DONE“, and “ERROR“ display the status of the task.

Note The CPU restarts (warm start) once the task successfully executed.

Requirements

Please, consider the following requirements for the use of the “T_CONFIG” instruction:

- In the hardware configuration, make sure that you have set that the IP addresses and the device names are assigned by the application program.
- Configuration of the PROFINET interface:
  - To change the IP address parameters, the option “IP address is set directly at device” must be activated.
  - To change the PROFINET device names, the option “PROFINET device name is set directly at the device” must be activated.
- The configuration data must be stored to the following system data types and delivered to the “CONF_DATA” parameter:
  - “IF_CONF_V4“: IP Address, subnet mask, router address.
  - “IF_CONF_NOS“: Device names of the IO devices pertaining to the IO system.

4.2.3 Structure of the “CONF_DATA” data block

The data block “CONF_DATA“ contains the PLC data type “ConfData“. This one is parametrised as follows:

- Header: „head“ with data type “IF_CONF_Header“:

  Table 4-4

Name	Data type	Value	Description
FieldType	UInt	0	Field type: Must always have the value "0".
FieldId	UInt	0	Field ID: Must always have the value "0".
SubfieldCount	UInt	2	Number of system data types used (“IF_CONF_V4“ and “IF_CONF_NOS“): 1. Only one of the system data types is used. 2. Both system data types are used.

- Structure “config“ with data type “IF_CONF_V4“:

  Table 4-5

Name	Data type	Value	Description
Id	UInt	30	System data type flag. This parameter must not be changed.
Length	UInt	18	Length of the system data type “IF_CONF_V4“. This parameter must not be changed.
Mode	UInt	1	1: Permanent configuration data. 2: Temporary configuration data (erasing the permanent configuration data)

Name		Data type	Value	Description
InterfaceAddress	1	Byte	192	IP address of the PROFINET interface. The IP address 192.168.0.10 is assigned to the IO controller.
	2	Byte	168	
	3	Byte	0	
	4	Byte	10	
SubnetMask	1	Byte	255	Subnet mask of the PROFINET interface. The subnet mask 255.255.255.0 s assigned to the IO controller.
	2	Byte	255	
	3	Byte	255	
	4	Byte	0	
DefaultRouter	1	Byte	0	Router address of the PROFINET interface. No router is assigned to the IO controller.
	2	Byte	0	
	3	Byte	0	
	4	Byte	0	

- Structure “nos“ with data type “IF_CONF_NOS“:

Table 4-6

Name		Data type	Value	Description
Id		UInt	40	System data type flag.
Length		UInt	246	Length of the system data type “IF_CONF_NOS“.
Mode		UInt	1	1: Permanent configuration data. 2: Temporary configuration data (erasing the permanent configuration data)
NOS	1	Byte	‚m‘	Device name of the IO controller (station name). The PN device name “machine_1” is assigned to the IO controller.
	2	Byte	‚a‘	
	3	Byte	'c'	
	4	Byte	‚h‘	
	5	Byte	'i'	
	6	Byte	‚n‘	
	7	Byte	'e'	
	8	Byte	‚_‘	
	9	Byte	‚1‘	

Note For further information on the “Multiple Use IO System”, please refer to the entry sheet (2) in the “Multiple Use IO System” document.

The structure “CONF_DATA” must be parametrised individually for any IO controller and thus for any plant.

4.3 Functionality of the Configuration Control within the IO system

4.3.1 Overview

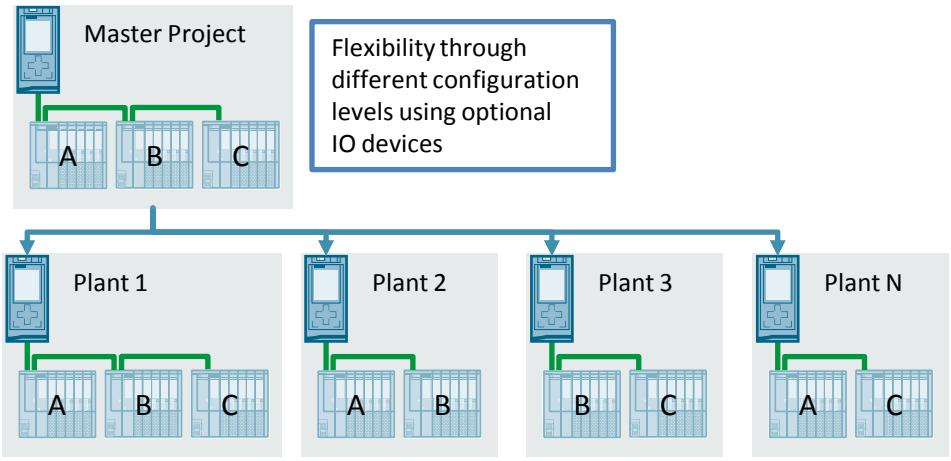
The Configuration Control within the IO system allows the flexible use of IO devices in the application example.

CPUs of type S7-1500 as of firmware version V1.7 or higher enable you to remove / add stations (IO devices, I devices) or to vary the sequence of the stations of a PROFINET IO system in a concrete plant.

It is possible to combine the Configuration Control for devices with the Configuration Control for IO systems; their functions are independent from each other.

The Figure below shows the function of the Configuration Control within the IO system.

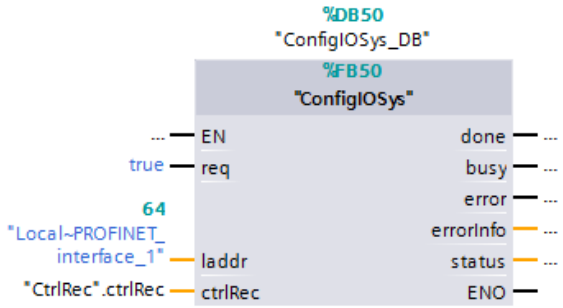
Figure 4-5



Call of “ConfigIOSys“ in “Main“ OB 1

In the application program, the function block “ConfigIOSys“ is called up as long as the configuration of the IO system has been successfully completed (output parameter “DONE”).

Figure 4-6



4.3.2 Program details of the “ConfigIOSys” function block

The “ConfigIOSys” function block is used to transmit the “CtrlRec” data record to the PROFINET interface of the CPU. This results in activating the optional IO devices and specified port interconnections set out in the “CtrlRec” data record for the relevant IO system.

Within the function block “ConfigIOSys”, the instruction “ReconfigIOSystem” is called up in all the three modes. The system deactivates at first all the IO devices networked with the IO controller, then transmits the data record “CtrlRec” to the IO controller’s PROFINET interface and finally, it re-activates all the IO devices.

The Figure below shows the mode of operation of the “ConfigIOSys” function block:

Figure 4-7

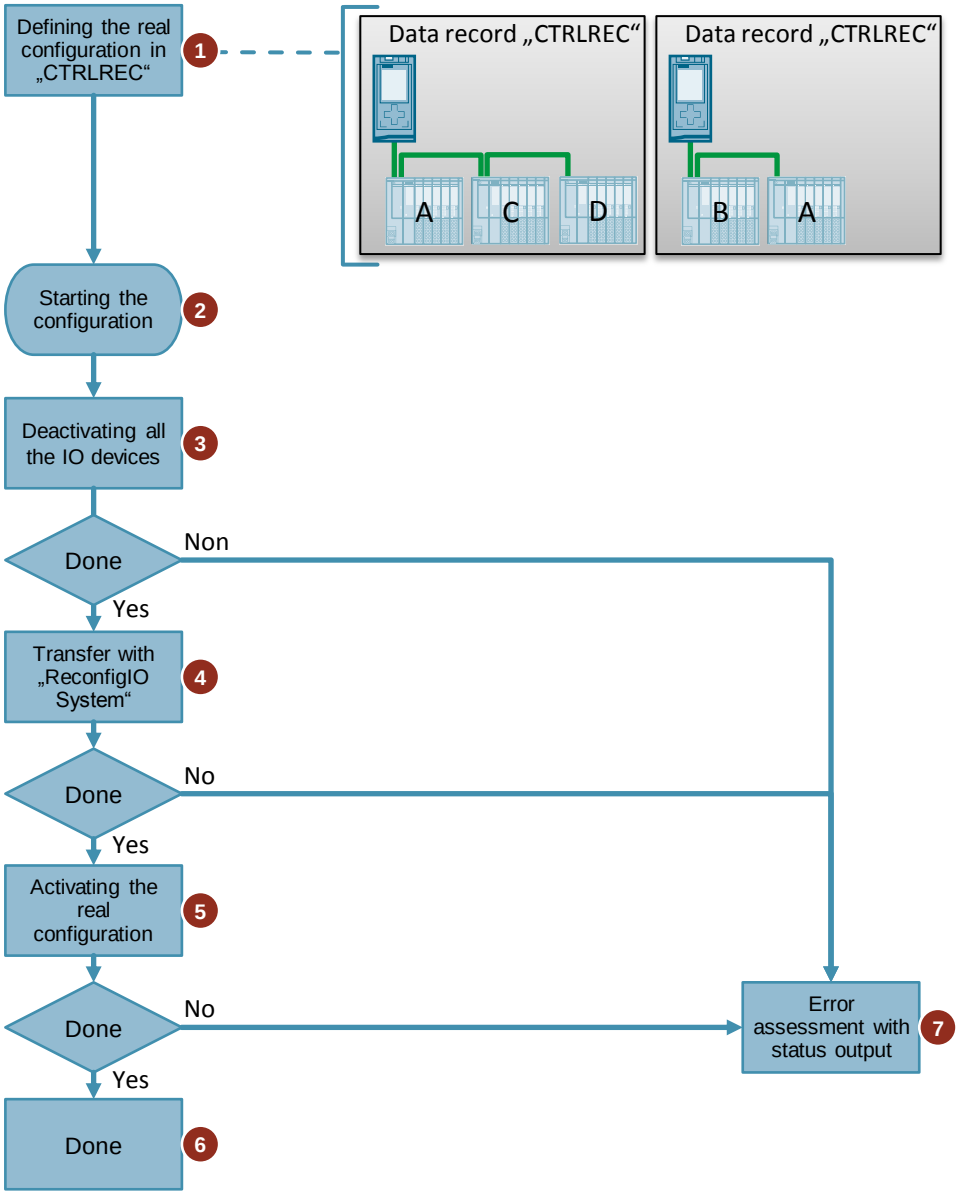


Table 4-7

No.	Event
1.	The data record “CtrlRec” determines the parametrisation to do for the the real IO system. For further information on the data record structure, please refer to the entry sheet (2) in the “Multiple Use IO System” document.
2.	The “Main” OB 1 is called up as soon as the controller is in the “RUN” mode. This block starts processing the function block “ConfigIOSys“.
3.	The instruction "ReconfigIOSystem" is called up in the 1st mode (“MODE“ = 1) in order to deactivate all the IO devices.
4.	As soon as all the IO devices are deactivated, the instruction "ReconfigIOSystem” is called up again in the 2nd mode (“MODE“ = 2) in order to transmit the data record “CtrlRec“ to the IO controller’s PROFINET interface.
5.	After the transmission of the data record was completed successfully, instruction “ReconfigIOSystem“ is called up a last time in the 3rd mode in order to re-activate all the IO devices involved in the current configuration.
6.	The function block “ConfigIOSys“ sets its output to “done” (“true”) to signal that all the modes were completed successfully.
7.	The function block “ConfigIOSys“ sets its output to “error“ (“true“) whenever one of the calls of the instruction “ReconfigIOSystem“ was not completed successfully. The “status” output displays the error message of the call of “ReconfigIOSystem“ that has caused the error.

4.3.3 Structure of the control data record “CtrlRec”

The control data record “CtrlRec” in the present application example is parametrised so that all the optional IO devices engineered in the Master project are activated.

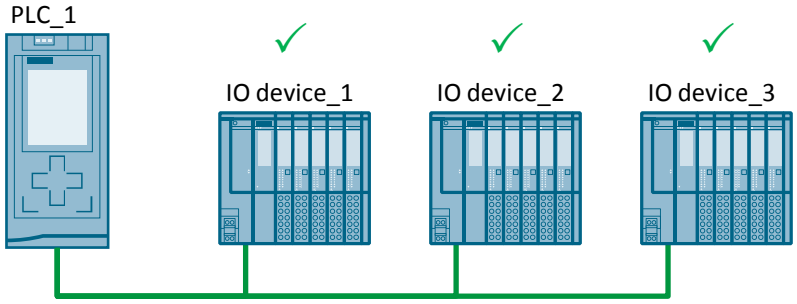
The table below shows the parametrisation of the data record “CtrlRec” in our application example:

Table 4-8

Name	Data type	Value	Description
ctrlRec[0]	Word	16#0100	Data record version.
ctrlRec[1]	Word	3	Number of optionl IO devices to be activated.
ctrlRec[2]	Word	272	HW flag of the first IO device to be activated.
ctrlRec[3]	Word	265	HW flag of the second IO device to be activated.
ctrlRec[4]	Word	262	HW flag of the third IO device to be activated.
ctrlRec[5]	Word	0	Number of specific port interconnections used.

The Figure below shows the structure activated through the allocation of “CtrlRec” and the Configuration Control within the IO system in the present application example.

Figure 4-8



Note	An IO device cannot be an optional IO device and possess a specified port interconnection at a time. Ports whose specified port interconnections are not defined in the application program are being declared as “Any partner”. The related IO devices are activated.

ATTENTION	**Reconfiguring may take up a lot of time, depending on the number of optional IO devices / flexible port interconnections.** **It might be necessary to increase the cycle time of the controller!**

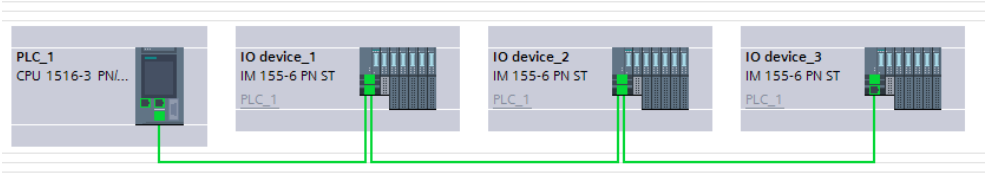
5 Configuration and Engineering

5.1 Engineering of the Configuration Control for IO systems

Topology

The Figure below shows the topology to be engineered for the present example.

Figure 5-1



All the IO devices are configured as optional IO devices; hence all the ports have to be firmly connected following the topology.

Configuring the IO controller

To configure the Configuration Control with optional IO devices, no parametrisation is required at the IO controller.

The IO controller must be correctly connected to the IO devices following the project topology.

Configuring the IO devices as optional IO devices

The following requirements must be satisfied:

- The interface modules used are equipped with a PROFINET interface.
- The distributed I/O stations are assigned to the same subnet as the IO controller.
- The IO devices have a fixed topology.

To configure optional IO devices, you have to parametrise those IO devices which shall be optional ones:

Proceed as follows:

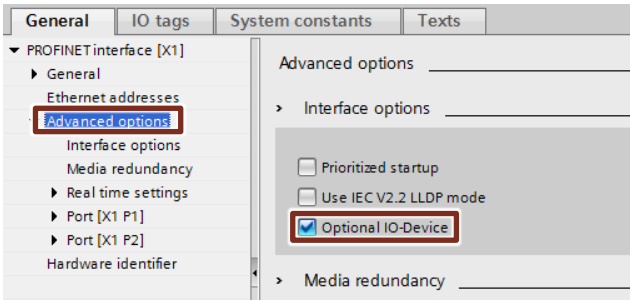
1. Access the “Devices & Networks” overview.

2. Double click the symbol of the corresponding IO device to open the related device view.

3. Click the PROFINET interface of the IO device. Access the “Advanced options”.

4. Under “Interface options” in the inspector window, check the “Optional IO Device” check box.

Figure 5-2



Note Repeat steps 1 thru 4 for any station to be used as an optional “IO Device”.

Result

The data record “CtrlRec” determines now the IO Devices which shall be activated for a given variant. The function block “ConfigIOSystem“ sends this data record to the IO controllers of the PROFINET IO system which activate the IO devices following the information contained in the data record.

In the present application example, all the optional “IO Devices” are activated.

Note For further information on configuring the Configuration Control for IO systems, please refer to the entry sheet (2) in the “Configuration Control for IO Systems” document.

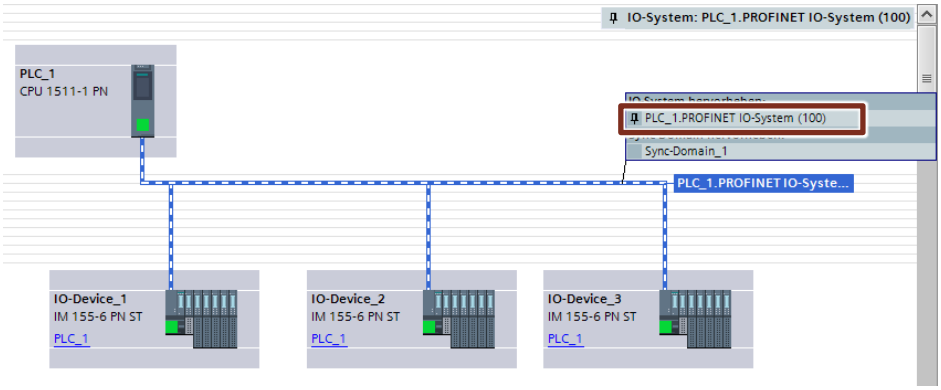
5.2 Configuring a “Multiple Use IO System”

Configuring the IO system

The steps below explain the settings to do at the PROFINET IO system:

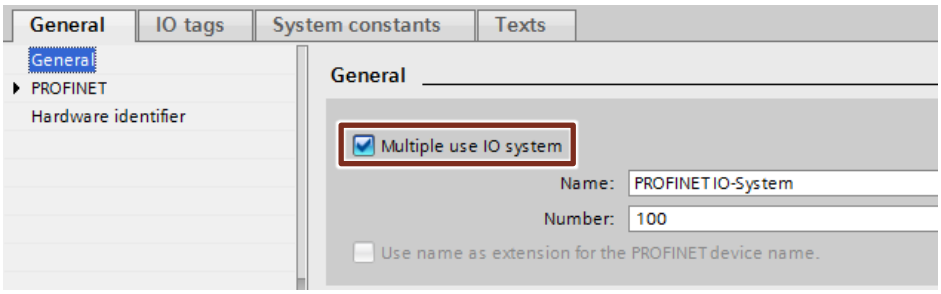
1. Select the IO system whose properties you wish to edit in the inspector window.

   Figure 5-3



2. Check the “Multiple use IO system” check box in the “General” area of the inspector window.

   Figure 5-4

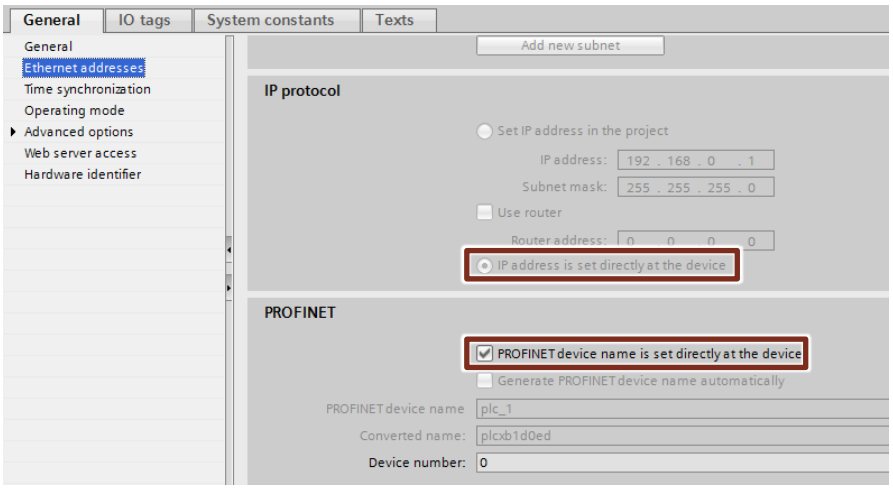


Result

Do the settings below at the devices in the STEP 7 IO system:

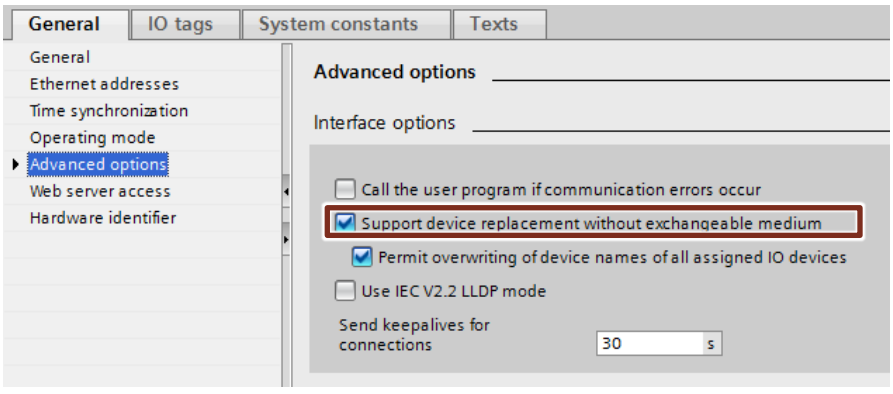
- IO controller:
  - The option “PROFINET device name is set directly at the device” is set. Initially, the IO controller does not have a PROFINET device name.
  - The option “IP address is set directly at the device” is set. Initially, the IO controller does not have an IP address.

    Figure 5-5



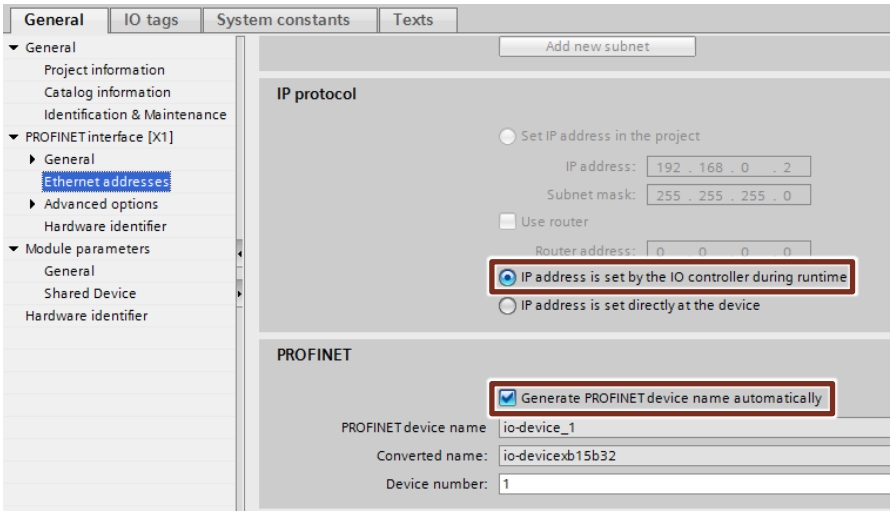
  - The option “Support device replacement without exchangeable medium” is activated. This facilitates an automatic commissioning. The commissioning engineer must no longer assign the device name and the IP address. Based on the target topology and the other settings, the IO controller assigns to the IO devices the device name and the IP address during start-up.

    Figure 5-6



- IO devices:
  - The option “Generate PROFINET device name automatically” is active.
  - The option “IP address is set by the IO controller during runtime” is active. Initially, the IO devices do not have an IP address.

    Figure 5-7



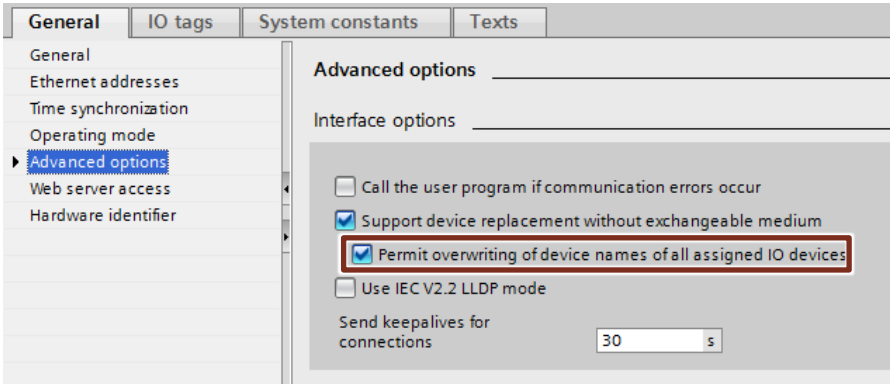
  - The device name is automatically assigned to the IO devices and is used on site to disambiguate the IP address.

Configuring the IO controller

Further settings are necessary at the IO controller to configure a “Multiple use IO system”:

1. Open the device view of the IO controller by double clicking “Device configuration”.
2. Click the PROFINET interface of the IO device to be connected. Access the “Advanced options”.
3. Check the option “Permit overwriting of device names of all assigned IO devices”.

   Figure 5-8



Result

The CPU is now able to overwrite the device names and the IP address of the IO devices assigned, using the application program.

Note For further information on configuring the “Multiple use IO system”, please refer to the entry sheet (2) in the “Multiple use IO Systems” document.

6 Installation / Commissioning

6.1 Installing the hardware

The figure below shows the hardware set-up of the application in maximum configuration:

Figure 6-1

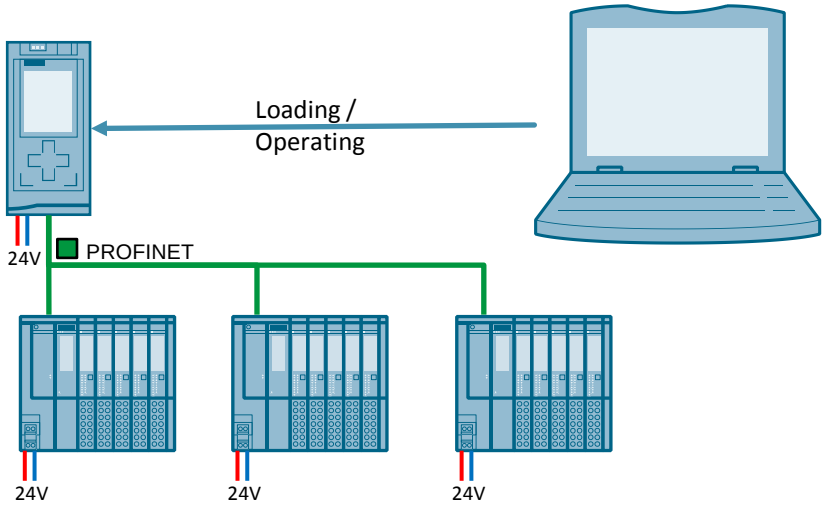


Table 6-1

No.	Action
1.	Connect the controller and the I/O systems to a 24 V power supply.
2.	Connect the components via an Ethernet cable (RJ45).

6.2 Installing the software

This chapter describes the steps required for installing the example code.

Note It is advisable to run the latest versions of any installed software.

Table 6-2

No.	Action	Note
1.	Install STEP 7 V14 on your programming unit.	To do this, follow the instructions of the program.
2.	Install WinCC V14 on your programming unit.	Follow the instructions of the program.

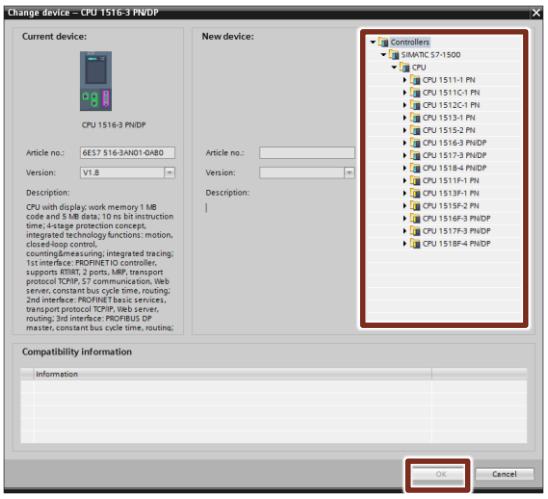
6.3 Commissioning

Note When assigning the IP addresses to your devices, make sure that all devices are located in the same subnet and each IP address is only assigned once within the subnet.

Controller

The table below shows how to commission the application example:

Table 6-3

No.	Action	Note
1.	Load the application example to your programming unit and unzip the archive.	You will find the entry under \2\ in “Links & Literature”.
2.	Open the example project	“29430270_MultipleIOSystemsKonfig_PROJ _v10.ap13“
3.	Open the “Device Configuration” of the “CPU 1516-3 PN/DP” controller. If you are using the same controller as in the example, proceed with step 5.	SIMATIC S7-1500 CPU 1516-3 PN
4.	Right-click the controller and click “Change device”. Select your S7-1500 from the tree and confirm with “OK”.	
5.	Now, the controller has been engineered. Adapt the control data record “CtrlRec” to your hardware set-up. In the control data record “CONF_DATA”, configure the desired IP address and the device name of the IO controller.	

Adapting the “CONF_DATA” structure

You have to enter the IP address and the device name of the IO controller into the data record “CONF_DATA”.

Enter the parameters at the following points:

- IP address: “CONF_DATA“ > “IF_CONF_V4“ > “InterfaceAddress“
- Subnet mask: “CONF_DATA“ > “IF_CONF_V4“ > “SubnetMask“
- Router address: “CONF_DATA“ > “IF_CONF_V4“ > “DefaultRouter“
- Device name: “CONF_DATA“ > “IF_CONF_NOS“ > “NOS[1..240]“

Note For further information on adapting the “CONF_DATA” data record, please refer to the entry sheet (2) in the “Multiple Use IO System” document.

Adapting the “CtrlRec” data record

An IO device declared as an optional IO device and used in the real configuration must be activated by the function block “ConfigIOSys”. For this purpose you must set the optional IO device in the “CtrlRec” data record by entering the IO device’s HW flag into the data record.

Use the CPU system tags for the interconnection in the “CtrlRec” data record. Details can be found here:

“PLC tags“ > “Show all tags“ > “System constants“

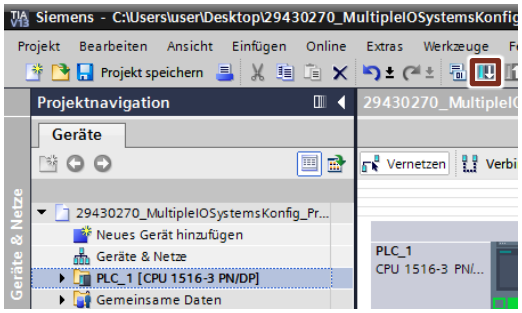
Adapt the “CtrlRec” data record to the Configuration Control in your IO system.

Note For further information on adapting the the “CtrlRec” data record, please refer to the entry sheet (2) in the “Configuration Control for IO Systems” document.

Loading

Load the program into your controller.

Figure 6-2

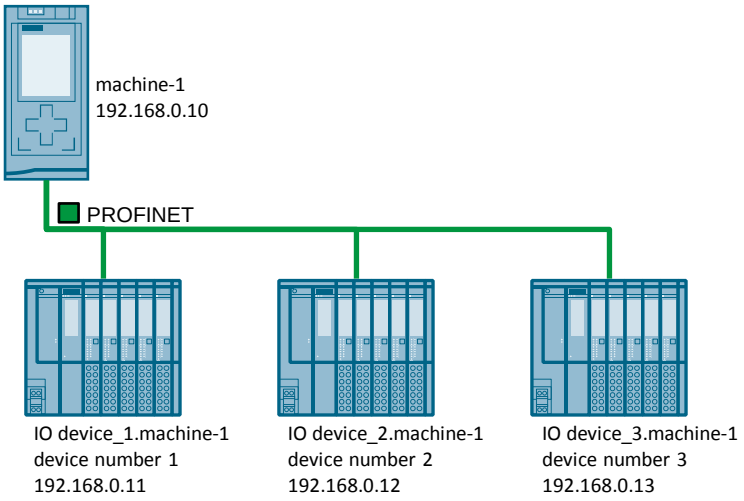


Once the program loaded into the controller and the controller restarted, the PROFINET IO system parametrises itself as described in the control data records “CtrlRec“ and “CONF_DATA“.

Note A tool like the SIMATIC Automation Tool may also be used for loading. For the link to the SIMATIC Automation Tool, please refer to \5\ in “Links & Literature”.

The Figure below shows how to activate the IO system in the delivered project of the application example.

Figure 6-3



7 Links & Literature

Table 7-1

	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Download page of the entry https://support.industry.siemens.com/cs/ww/en/view/29430270
\3\	Manual PROFINET with STEP 7 V14 https://support.industry.siemens.com/cs/ww/de/view/109742272
\4\	Entry regarding the SIMATIC Automation Tool https://support.industry.siemens.com/cs/ww/de/view/98161300
\5\	PRONETA download page https://support.industry.siemens.com/cs/ww/en/view/67460624

8 History

Table 8-1

Version	Date	Revision
V1.0	06/2016	First version
V2.0	03/2017	Update to STEP 7 V14