Application Note

Applicative System Redundancy for PLCnext Control AXC F 2152

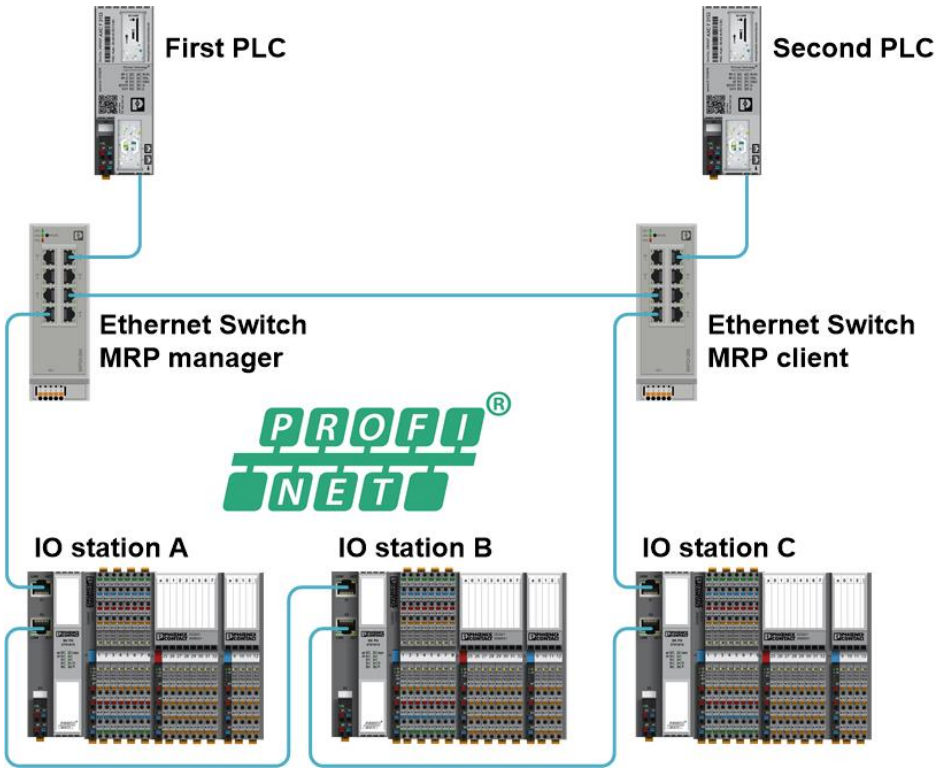
Established by:
Phoenix Contact Electronics GmbH
Friedrich Hackl
Phone: +49 52 81 9 46-2264
E-mail: fhackl@phoenixcontact.com

Revision Documentation			
Date	**Author**	**Comments**	**Revision**
27.01.20	Friedrich Hackl	Preliminary version	**00**
29.04.20	Friedrich Hackl	First released version	**01**
19.02.21	Friedrich Hackl	Documentation adapted to library version 2	**02**
03.11.21	Friedrich Hackl	Documentation adapted to library version 3	**03**
02.06.22	Friedrich Hackl	Documentation of HF1 for version 3 added	**04**
14.11.22	Friedrich Hackl	Documentation adapted to library version 4	**05**
16.01.23	Friedrich Hackl	Documentation of HF1 for version 4 added	**06**
02.10.24	Friedrich Hackl	Documentation adapted to library version 5	**07**

Table of Contents

1 Introduction

The Applicative System Redundancy (ASR) is realized by providing two redundant PLCs (First and Second) in the same network together with the responding IO stations. The communication between the PLCs and the IO stations is based on the PROFINET RT protocol.



On both controllers (First and Second PLC) the identical application program is in operation. One of the PLCs is the active controller (serves as Primary controller) and the other is the standby controller (serves as Backup controller). The Primary controller controls the process. The redundancy role (Primary or Backup) is specified by the application software. Therefore, the responding function block needs to be implemented and parameterized in the application program. If the Primary controller fails or some other user defined switch-over condition occurs, the Backup controller takes over the process control.

The use of function blocks for data synchronization allows selected data to be continuously synchronized between the Primary and the Backup controller during normal operation (as long as both controllers are available). By holding and processing identical application data, a seamless switch-over and a short take-over time is possible from the Primary to the Backup controller in case of malfunction of the Primary controller.

2 Change Notes

Library version / Library build / PLCnext Engineer	Change notes

PLCnext_ASR2152_5
20241009
2024.0.3 LTS
	Included libraries:
	• Update of the library “PLCnextBase” to the latest version 1.6.4.
	
	Function block group “ContrRed”:
	• Function block “ASR_ContrRed_3152_4”:
	  o Internal improvements
	• Function block for the integration of non-S2-capable PROFINET IO devices added:
	  o ASR_NonS2_AR_StartStop_2
	
	Function block group “DataSync”:
	• Function block “ASR_Data_LREAL_4”:
	  o Internal improvements
	• Data synchronization function block for data type “STRING” added:
	  o ASR_Data_STRING_2

PLCnext_ASR2152_4_HF1 20230118 2022.0 LTS	**Function block “ASR_GET_SNMP_UNI_4”:** • Internal improvements due to data type conflict with PLCnext Engineer version ≥ 2023.0 LTS. **Function block “ASR_PORT_STATE_PN_BK_2”:** • Internal improvements due to data type conflict with PLCnext Engineer version ≥ 2023.0 LTS.
PLCnext_ASR2152_4 20221123 2022.0 LTS	**Included libraries:** • Update of the library “PLCnextBase” to the latest version 1.5.1. **Function block “ASR_ContrRed_2152_4”:** • The input parameter “xInhibitPrimaryRole” has been eliminated in this version: The function block now automatically detects whether the Ethernet port of the Profinet IO Controller is connected to the network. That’s why this input parameter is no longer necessary. • Project comparison between the two redundant PLCs: When the function block is initialized and activated (wDiagCode = 16#8000), the least significant bit of the output parameter “wAddDiagCode” indicates whether the IEC61131 project code is identical on both PLCs. • Initialization of the PLCnext component for the PLCnextBaseService library: In case of an ASR project was downloaded to PLC the first time, the function block reported an error which indicated that the IP address settings of the PLC could not be read. This behavior was related to the missing initialization of the PLCnext component used by the PLCnextBaseService library. In this version the function block automatically checks whether this PLCnext component has been initialized proper. If the function block detects that this component has not been started, an automatic restart of the PLC will be executed. • Internal improvements.

PLCnext_ASR2152_3_HF1 20220603 2021.0 LTS	**Function block "ASR_SetupBaseServices_1":** This ASR library uses the PLCnextBaseService as a sub library in order to get access to some important system services. Usually, the responding PLCnext component, which makes these services available, starts up automatically when the controller is started. However, in some rare circumstances it could happen that the PLCnext component of the PLCnextBaseService library is not initialized proper and therefore the function block "ASR_ContrRed_2152" displays an error (wDiagCode = 16#C405 and wAddDiagCode = 16#0002). To resolve this issue, the function block "ASR_SetupBaseServices" can be used to check whether this PLCnext component has been initialized proper. If the function module detects that the component has not been started correctly, the function block either performs a warm restart or a warm reboot of the PLC and checks the proper initialization of the responding PLCnext component once again.
PLCnext_ASR2152_3 20220210 2021.0 LTS	**Included libraries:** • Update of the library "PLCnextBase" to the latest version 1.4.1. • New library "ProfinetServices" version 1.0.0 added. **Function block "ASR_ContrRed_2152_3":** • Via the input parameter "uiEthernetAdapter" the function block can be parameterized for the usage of the left-alignable Ethernet interface AXC F XT ETH 1TX. • The two output parameters "wDiagCode" and "wAddDiagCode" now indicate status messages in addition to the error messages. • Internal improvements. **Function block group "DataSync":** • Data synchronization function blocks for data types "LINT", "ULINT" and "LWORD" added: o ASR_Data_LINT_1 o ASR_Data_ULINT_1 o ASR_Date_LWORD_1 • Bistable function blocks "RS" and "SR" with integrated synchronization of the output parameter "xQ1" added: o ASR_Data_RS_1 o ASR_Data_SR_1 • Counter function blocks for counting up (CTU), counting down (CTD) and counting up/down (CTUD) added with integrated synchronization of the counter value: o ASR_Data_CTU_1 o ASR_Data_CTD_1 o ASR_Data_CTUD_1 • Timer function blocks for delayed switching on (TON), delayed switching off (TOF) and pulse (TP) added with integrated synchronization of the elapsed time: o ASR_Data_TON_1 o ASR_Data_TOF_1 o ASR_Data_TP_1 • For all other function blocks in this group, version 2 has been changed to version 3 due to an improvement in the internal communication structure. **Function block "ASR_GET_SNMP_UNI_3":** • Improvement of the internal communication structure. **Function block "ASR_PORT_STATE…":** • The function block "ASR_PORT_STATE_AXLBK_1" was replaced by the function block "ASR_PORT_STATE_PN_BK_1".

	With this new function block, it is now possible that the port status of various types of PROFINET IO devices can be queried. **Program "ASR_EventStop_2":** • The ARs (Application Relations) to all specified PROFINET IO devices are deactivated in case of the PLC changes its operating mode to Stop.
PLCnext_ASR_2 20210125 2021.0 LTS	First released version.

3 System requirements/overview

The following system requirements are necessary for the implementation and error-free execution of the software.

PROFINET System Redundancy

- The Applicative System Redundancy is based on the mechanism of the PROFINET System Redundancy. This PROFINET standard describes several types of System Redundancy.
- The PLCnext Technology control system supports PROFINET System Redundancy type S2 according to that PROFINET specification.

PROFINET IO devices

- PROFINET IO devices used in such a system must support PROFINET system redundancy type S2, including the “RT_InputOnBackupAR_Supported” option activated. This option ensures that the PROFINET IO devices provide valid input data also for the Backup controller.
- PROFINET IO devices that do not support this option should not be integrated into the PROFINET bus configuration when the PROFINET system redundancy type S2 is activated.

IO stations

The IO stations are directly integrated into a network ring structure via the Media Redundancy Protocol (MRP) by using two Ethernet switches. One switch must be configured as MRP manager and the other as MRP client. In such a network ring structure with Media Redundancy Protocol, a maximum network recovery time of 200ms can be expected.

- First and Second PLC (PROFINET IO controller):
  AXC F 2152 with firmware version ≥2024.0 LTS

- Ethernet switch MRP manager and client:
  FL SWITCH 22xx with firmware version ≥3.20

- IO stations (PROFINET IO device):
  AXL F BK PN TPS with firmware version ≥1.32
  AXL F BK PN TPS XC with firmware version ≥1.32
  AXL P BK PN with firmware version ≥1.0.7
  IL PN BK-PAC with firmware version ≥1.11
  AXC F 1152 with firmware version ≥2021.0 LTS
  AXC F 2152 with firmware version ≥2021.0 LTS

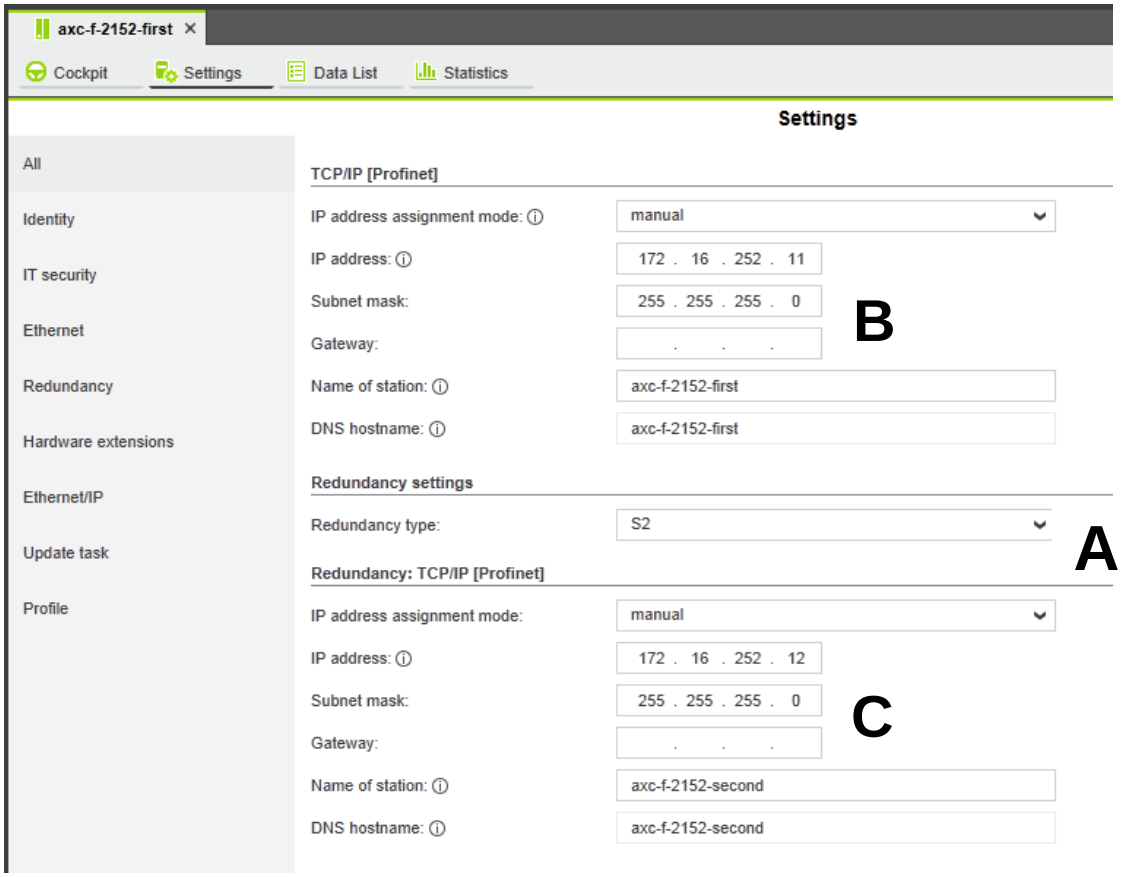
Firewall configuration

If the firewall is activated on the PLCs, appropriate rules have to be configured in order to allow communication between the PLCs via following ports:

Description	Protocol	Port
Controller Redundancy	UDP	3101
Data Synchronization	UDP	3102, 3103

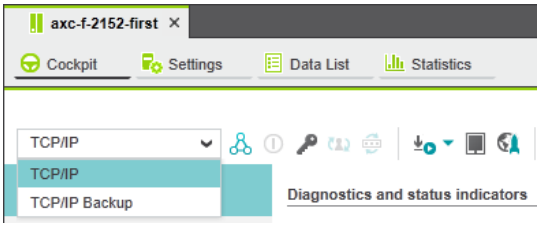
4 Engineering software PLCnext Engineer for redundant systems

The PROFINET System Redundancy needs to be activated and parameterized. For a detailed description of how to do this, please have a look at the online help of PLCnext Engineer.



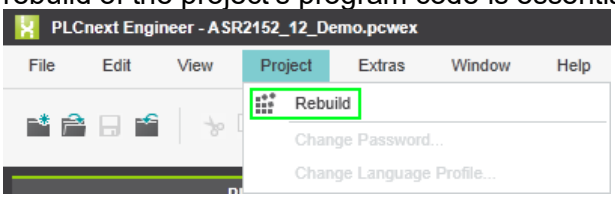
A Activation of the PROFINET System Redundancy
B IP-address and PROFINET name of the First PLC
C IP-address and PROFINET name of the Second PLC

Afterwards the selection with which controller you would like to connect to for the purpose of program download or debugging is subsequently possible via the dropdown menu in the Cockpit tab:

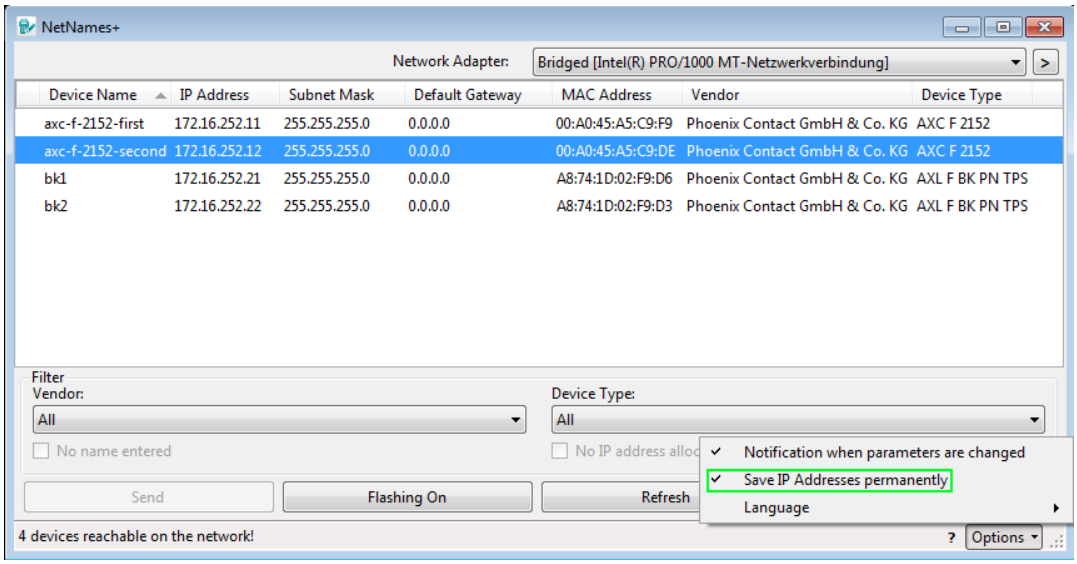


- TCP/IP Connect online services to the First PLC
- TCP/IP Backup Connect online services to the Second PLC

NOTE:
If you switch the communication path between the First and Second PLC, a rebuild of the project's program code is essential:



For the adjustments of the IP address and the PROFINET name on the PLCs, it is preferable to use the stand-alone software tool NetNames+. If the parameters are to be stored on the device permanently, the corresponding option must be selected:

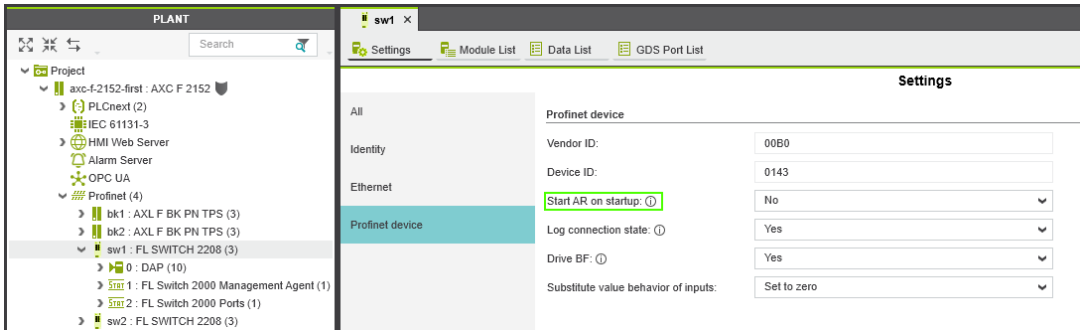


4.1 Configuration of the Media Redundancy network ring structure

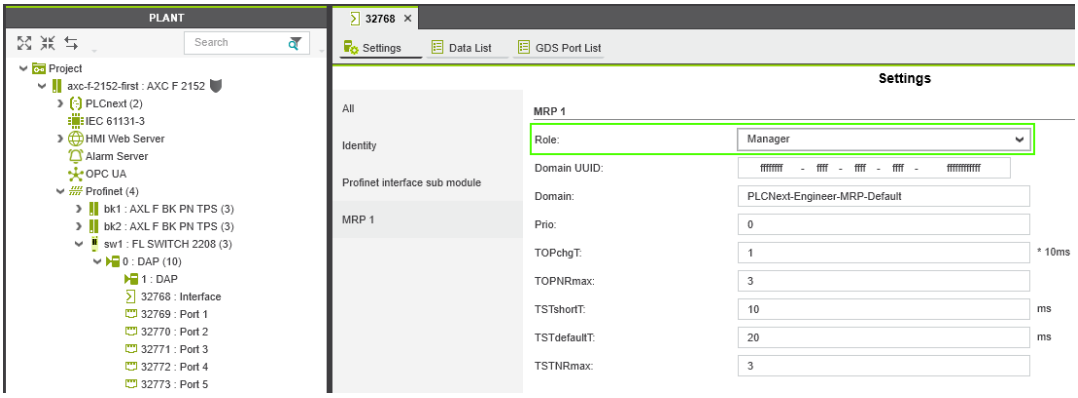
4.1.1 Network switches FL SWITCH 22xx (PN)

Network switches with firmware version ≥ 3.20 used in the Applicative System Redundancy can be integrated as PROFINET IO devices into the PROFINET configuration of the PLCnext Engineer project. Therefore, the “PROFINET” operating mode (which is the default setting of the PROFINET version) is correct for these network switches.

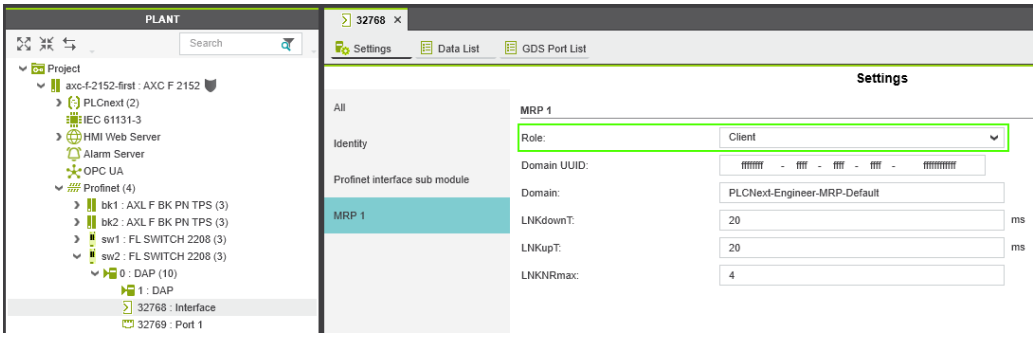
At the “Profinet device” category of the settings for the node of the PROFINET IO device, the “Start AR on startup” parameter has to be set deactivated:



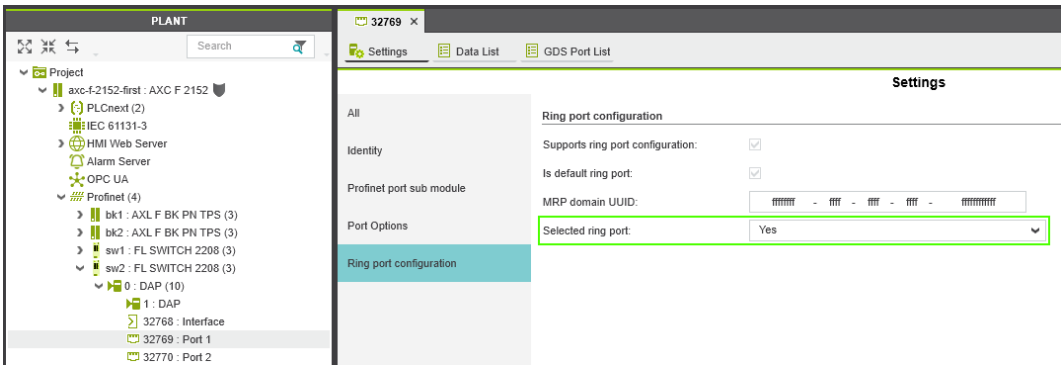
At the settings of the “interface” node the parameters for MRP can be adjusted. One switch of the MRP ring structure needs to be parametrized as MRP manager:



Since there may only be one manager in an MRP ring structure, all other switches need to be parameterized as MRP clients:



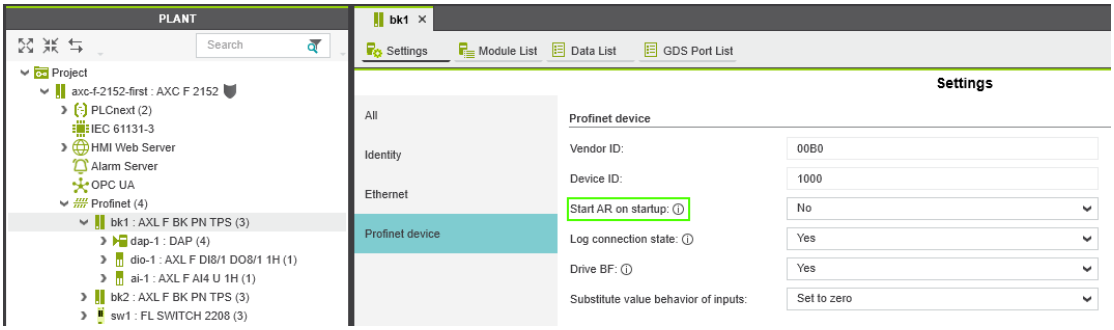
At the “Ring port configuration” category of the ports, the “Selected ring port” parameter must be set to “Yes” for those ports which are part of the MRP ring structure:



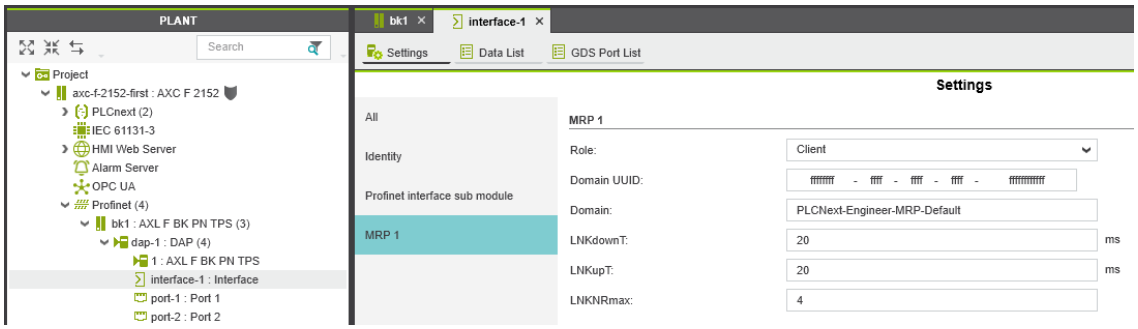
4.1.2 IO stations (PROFINET IO devices)

The IO stations can be directly integrated into the MRP network ring structure. In this case, the MRP client setting on the bus coupler must be activated and parameterized. For these devices, the necessary settings of the MRP client are directly done in PLCnext Engineer.

At the “Profinet device” category of the settings for the node of the PROFINET IO device the “Start AR on startup” parameter has to be set deactivated:

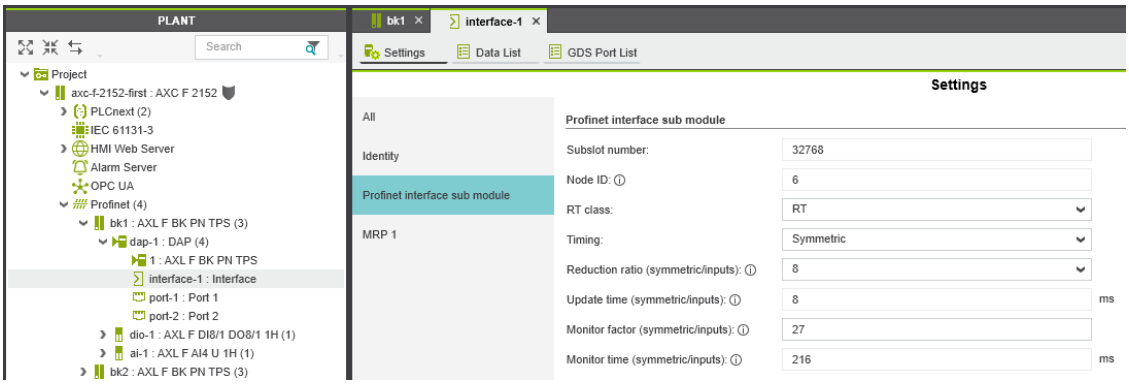


At the settings of the “interface” node the parameters for MRP can be set. For the use case of the Applicative System Redundancy, the default values can be maintained:



In general, PROFINET System Redundancy is independent of the network topology and, conversely, Media Redundancy is independent of the PROFINET System Redundancy.

Therefore, both redundancy levels can be combined or used independently of each other. However, if both redundancy mechanism are combined, the network recovery time of the Media Redundancy Protocol (MRP) must be considered in the adjustments of the PROFINET System Redundancy. Specifically, this concerns the value of the so-called “Monitor time” for the PROFINET IO devices. This parameter must be set in that way that a switch-over in the ring network does not result in a loss of communication between PLCs and IO stations. This means that the “Monitor time” value must be greater than the maximum network reconfiguration time to expect. The value for the “Monitor time” is the product of “Monitor factor” and “Update time” values and cannot be set directly. Since the “Update time” can be adjusted individually according to the needs of the application for each PROFINET IO device, the “Monitor factor” must also be adjusted to the proper value for each device in the settings of the “interface” node:

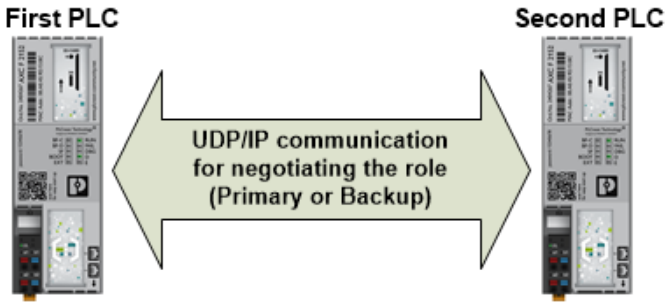


Since the maximum of the network recovery time for MRP is 200ms, at the example of an “Update time” of 8ms the “Monitor factor” needs to be set to a minimum of 27 which results in 8 x 27 = 216ms for the “Monitor time” attribute.

5 Function block for the Controller-Redundancy

5.1 General

The communication between the two PLCs necessary for the Controller-Redundancy is realised via an UDP/IP communication protocol.



For this purpose, the following rules for the assignment of the PROFINET IO controller IP addresses must be considered:

- The IP addresses of the First and Second PLC must use the same Net-IDs.
- The difference between the IP addresses of the First and the Second PLC must be limited to the fourth octet only.

Example:

	First PLC	Second PLC
IP Address	192.168.0.91	192.168.0.92
Subnet Mask	255.255.255.0	255.255.255.0

NOTE:
If the PLCnext controller “AXC F 2152” is extended with another separate network interface by adding the left-alignable Ethernet interface AXC F XT ETH 1TX it must be considered that the PROFINET IO controller function is automatically switched to this interface. Therefore, in this case, the rules listed above refer to the IP address settings of this left-alignable Ethernet interface.

5.2 Function block “ASR_ContrRed_2152”

On the two PLCs the same program is in operation. Therefore, during the runtime of the program is necessary to determine, which PLC is the First controller and which PLC is the Second controller. For this purpose, the fourth octet of the IP addresses (Host-ID) of the First and of the Second controller needs to be stored as a constant at the corresponding input parameters of this function block in the PLCnext Engineer program.

During the start-up of the program, the own IP address is read out by executing a firmware service. As a result of the comparison of the own IP address with these constants, it can be determined whether the controller is the First or the Second PLC.
Only one controller (either First or Second) controls the process. The PLC which controls the process is the Primary, the other PLC has the role of the Backup. The determination which PLC is the Primary, respectively Backup, is done by a comparison of the value of its own current state signals with the current state signals of the partner PLC.

For this purpose, this function block compares cyclically the current state signals of the partner PLC (xPartnerSignal...) with its own current state signals (xSignal...). The values of the signals are compared in ascended order beginning with 0.
If a difference of the values at a determined bit-number x is perceived the values of the current state signals with higher bit-numbers (>x) are not compared anymore. Therefore, the priority of a switch-over condition depends on the bit-number used for this current state signal.

Possible switch-over conditions are:

- Switching over the redundancy role of a PLC from Backup to Primary is carried out if the value of its own current state signal is TRUE and the value of the corresponding current state signal from the partner PLC is FALSE.

- Switching over the redundancy role of a PLC from Primary to Backup is carried out if the value of its own current state signal is FALSE and the value of the corresponding current state signal from the partner PLC is TRUE.

5.2.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
iIP4_First	INT	Input	Fourth octet of the IP address of the First PLC.
iIP4_Second	INT	Input	Fourth octet of the IP-address of the Second PLC.
uiEthernetAdapter	UINT	Input	Usage of an Ethernet extension module: 0: No Ethernet extension module is used. 1: The left-alignable Ethernet interface AXC F XT ETH 1TX is used.
tSwitchOverDelay	TIME	Input	Delay time for switching over the PLCs in case of a switch-over condition occurs.
xResetSignalDelayMax	BOOL	Input	A rising edge at this input signal resets the "uiSignalDelayMax" output signal.
xSignal...	BOOL	Input	Current state signals: 0: highest priority to 7: lowest priority.
arrAR_DevList	ASR_ARR_STR_PN_DEV_LIST	Input	“Name of station” list of all PROFINET IO devices.
xFirst	BOOL	Output	Indicates whether the PLC is the First PLC.
xSecond	BOOL	Output	Indicates whether the PLC is the Second PLC.
xPrimary	BOOL	Output	Indicates whether the PLC has the redundancy role Primary.
xBackup	BOOL	Output	Indicates whether the PLC has the redundancy role Backup.
xPartnerIsFirst	BOOL	Output	Indicates whether the partner PLC is the First PLC.
xPartnerIsSecond	BOOL	Output	Indicates whether the partner PLC is the Second PLC.

xPartnerIsPrimary	BOOL	Output	Indicates whether the partner PLC has the redundancy role Primary.
xPartnerIsBackup	BOOL	Output	Indicates whether the partner PLC has the redundancy role Backup.
xPartnerSignal...	BOOL	Output	Current state signals of the partner PLC: 0: highest priority to 7: lowest priority
strOwnIP	STRING	Output	Own IP address.
strPartnerIP	STRING	Output	IP address of the partner PLC.
strLastToPrimary	STRING	Output	Indicates time and date for the last switch-over of the PLC from Backup to Primary.
bLastToPrimarySig	BYTE	Output	Indicates the state of the signals when the last switch-over from Backup to Primary was performed.
bLastToPrimaryPartnerSig	BYTE	Output	Indicates the state of the signals from the partner PLC when the last switch-over from Backup to Primary was performed.
strLastToBackup	STRING	Output	Indicates time and date for the last switch-over of the PLC from Primary to Backup.
bLastToSlaveSig	BYTE	Output	Indicates the state of the signals when the last switch-over from Primary to Backup was performed.
bLastToSlavePartnerSig	BYTE	Output	Indicates the state of the signals from the partner PLC when the last switch-over from Primary to Backup was performed.
diSignalDelayMax	DINT	Output	Maximum measured delay time for the transmission of the current state signals. This output parameter can be reset by a rising edge at the input parameter “xResetSignalDelayMax”.
xError	BOOL	Output	This output indicates with a rising edge that an error has occurred. You can read the corresponding error codes at the output parameters "wDiagCode" and "wAddDiagCode" during the cycle in which the rising edge occurs.
wDiagCode	WORD	Output	Indicates the cause of the error.
wAddDiagCode	WORD	Output	Indicates additional information to the cause of the error.
iRole	INT	Output	Indicates the current redundancy role of the PLC: 0: Undefined 1: Primary 2: Backup
iSyncLink	INT	Output	Indicates the status for the transmission of the current state signals: 0: Undefined 1: Transmission of the signals is OK 2: Transmission of the signals is not OK
iSwitchCount	INT	Output	Number of switchovers to Primary since the last restart of the PLC.

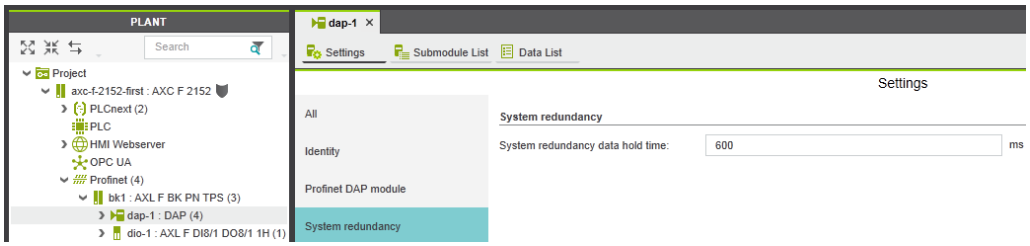
The valid range for the value of the “tSwitchOverDelay” input parameter is between 300ms and 1000ms. The achievable switch-over time of the system depends on the following parameters of the overall application and therefore cannot be generalized:

- The task interval in which this function block is used.
- The overall CPU load of the PLC.
- The Network Redundancy protocol which is used for the application specific network layout.

However, there is a recommended minimum switch-over time which can be adjusted depending on the adjusted task interval in which the function block is used:

Task interval	Minimum value for the input parameter „tSwitchOverDelay“
20ms	300ms
30ms	500ms

The switch-over time of the system must also be parameterized for the PROFINET IO devices. This is done with PLCnext Engineer at the settings of the “DAP” node via the “System redundancy data hold time”. The value entered here must not be lower than the sum of switch-over time set at the function block + ”Monitor time” set at the settings of the PROFINET IO devices:



5.2.2 Diagnostic

Status codes:

wDiagCode	Description
8000	Indicates that the function block is activated and initialized. The least significant bit of the output parameter “wAddDiagCode” indicates whether the IEC61131 project code is identical on both PLCs.
8100	Indicates that the function block is in the initialization phase. The “wAddDiagCode” output parameter indicates the relevant step of the internal state machine (High Byte) and some internal values for further diagnosis (Low Byte).

Error codes:

wDiagCode	Description
C401	Indicates that an error occurred while trying to stop the PROFINET ARs. The “wAddDiagCode” output parameter indicates the relevant error code from the responding function block.
C402	Indicates that an error occurred while trying to start the PROFINET ARs. The “wAddDiagCode” output parameter indicates the relevant error code from the responding function block.
C403	Indicates that an error occurred while reading the adjustments of the IP address. The “wAddDiagCode” output parameter indicates the relevant error code from the responding function block.
C404	Indicates that this PLC is neither the First nor the Second PLC, because the fourth octet of the IP address is neither equal with the value of the input parameter “iIP4_First” nor with “iIP4_Second”. The “wAddDiagCode” output parameter contains the fourth octet of the adjusted IP address.

C405	Indicates that during the start-up of the internal state machine a timeout occurred. The “wAddDiagCode” output parameter indicates the relevant step at which the timeout was detected.
C407	Indicates that an invalid value has been applied to the “uiEthernetAdapter” input parameter. At this input parameter only the values UINT#0 and UINT#1 are allowed. The “wAddDiagCode” output parameter indicates the value which was applied to this input parameter.
C408	Indicates that an error occurred while trying to check the proper initialization of the responding PLCnext component for the PLCnextBaseService library. The “wAddDiagCode” output parameter indicates the relevant error code from the responding function block.

5.2.3 Additional program instances in the ESM

For the controller redundancy to work properly, the following program instances must be created in addition to the cyclic call of the function block:

- Program instance for the system event task “Cold Start”:

Name	Component name	Task type	Event name	Program type	Interval (ms)	Priority	Threshold (ms)	Watchdog (ms)
ESM1								
COLDSTART		Event task	Cold start			0	0	100
ProgColdstart	Arp.Plc.Eclr			ASR_EventStart_1				

- Program instance for the system event task “Warm Start”:

Name	Component name	Task type	Event name	Program type	Interval (ms)	Priority	Threshold (ms)	Watchdog (ms)
ESM1								
WARMSTART		Event task	Warm start			0	0	100
ProgWarmStart	Arp.Plc.Eclr			ASR_EventStart_1				

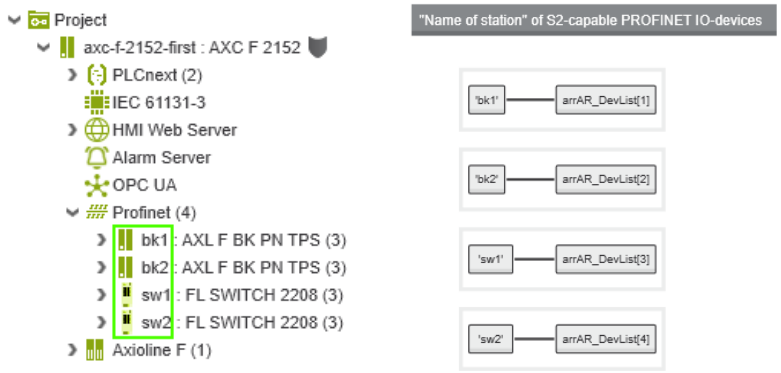
- Program instance for the system event task “Stop”:
  Note: The watchdog of this system event task must be set to 500ms.

Name	Component name	Task type	Event name	Program type	Interval (ms)	Priority	Threshold (ms)	Watchdog (ms)
ESM1								
STOP		Event task	Stop			0	0	500
ProgStop	Arp.Plc.Eclr			ASR_EventStop_2				

5.2.4 Recommendations for the configuration of the function block

InOut parameter “arrAR_DevList”:

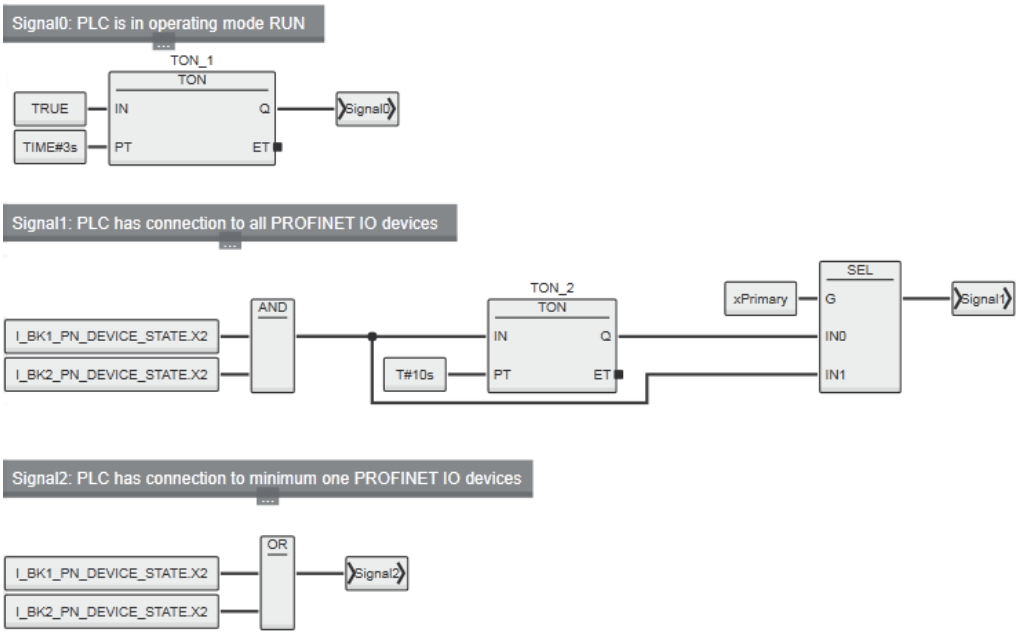
At start-up, the function block performs some PROFINET specific services on the PROFINET IO devices and therefore the function block needs to know the name of all connected IO devices. For this purpose, the “Name of Station” of all PROFINET IO devices (bus couplers) must be stored at the “arrAR_DevList” InOut parameter:



In this example program, four PROFINET IO devices with the names “bk1”, “bk2”, “sw1” and “sw2” are used. If there are more IO devices connected their names must be also added to this list.

“xSignal0” to “xSignal7” input parameters:

The switch-over conditions are defined by logical operations which are assigned to the “xSignal0” to “xSignal7” input parameters.
If there are no special requirements related to the switch-over conditions, the following configuration is recommended to trigger a switch-over in case of “standard failure” situations, like switching off the power supply of a PLC, setting a PLC into Stop operating mode, or disconnecting the Ethernet connection:



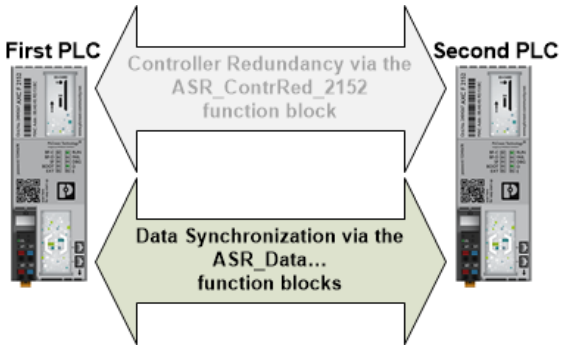
The “I_BKx_PN_DEVICE_STATE” variables are to be defined as External Variables and the responding Global Variables needs to be connected to the “PN_DEVICE_STATE” process data item of the PROFINET IO devices.
At this example program, two PROFINET IO devices are used. If there are more PROFINET IO devices in the system, the “AND” and the “OR” function needs to be extended accordingly.

5.2.5 OPC UA variables

To make it possible to display the current status of the redundant controllers on a visualization system, the function block provides corresponding information in the form of OPC UA variables via the embedded OPC UA server on the PLCs.

Name	Data type	Description
iRole	INT	Indicates the current role of the PLC: 0: Undefined 1: Primary 2: Backup
iSwitchCount	INT	Number of switchovers to Primary since the last restart of the PLC.
iSyncLink	INT	Indicates the status for the transmission of the current state signals: 0: Undefined 1: Transmission of the signals is OK 2: Transmission of the signals is not OK
uiPlcInRun	UINT	Cycle counter for the “PLC in Run” monitoring. The value of this counter is incremented each program cycle by 1 as soon as the application program is in operating mode RUN.
siFirstPlcCurrentRole	SINT	Indicates the current role of the First PLC: 0: Undefined 1: Primary 2: Backup
siFirstPlcState	SINT	Indicates the current state of the First PLC: 0: Undefined 1: Error 2: Run
siSecondPlcCurrentRole	SINT	Indicates the current role of the Second PLC: 0: Undefined 1: Primary 2: Backup
siSecondPlcState	SINT	Indicates the current state of the Second PLC: 0: Undefined 1: Error 2: Run

6 Function blocks for the Data Synchronization



At the Applicative System Redundancy, the same application program is in operation on the two redundant PLCs (First and Second PLC) but only one PLC (either First or Second) controls the process. By means of the function block “ASR_ContrRed_2152” the PLCs negotiate among themselves who controls the process. The PLC, which controls the process is the Primary, the other PLC has the role of the Backup.

The task of the function blocks described here is to synchronize values of selected variables between the PLCs. The library provides function blocks for common standard data types and IEC 61131-3 function blocks for which it is necessary to synchronize internal values. For this purpose, there are three different types of function blocks:

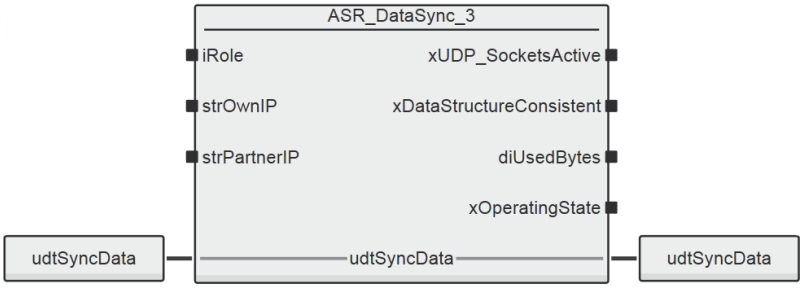
- Main function block for establishing the communication link between the PLCs.
- Function blocks for the synchronization of standard data types.
- IEC 61131-3 function blocks with integrated data synchronization.

NOTE:

- These function blocks are designed to be used only in one common cyclic task.

- The call of the function block must be active in each program cycle. This means that these function blocks must not be used inside any IF-, ELSE-, ELSIF-, REPEAT-, CASE-, WHILE-Statement or FOR-Loops.

6.1 Main function block “ASR_DataSync”



The input parameters "iRole", “strOwnIP” and "strPartnerIP" must be connected to the corresponding output parameters of the function block "ASR_ContrRed_2152". Via the “xUDP_SocketsActive” output parameter, the function block indicates whether the UDP/IP sockets, necessary for the data synchronization, were successfully opened.

xDataStructureConsistent

The “xDataStructureConsistent” output parameter indicates with TRUE that the number and the sequence of calls of the function blocks for the synchronization of the standard data types and the IEC 61131-3 function blocks with built-in data synchronization are identical on both PLCs. Only if this bit indicates TRUE, the variables between the two controllers will be synchronized. Via the “xOperatingState” output parameter it is shown whether the communication between the two PLCs is in operation.

iUsed Bytes

The “iUsedBytes” output parameter shows the number of bytes which are necessary for the synchronization of the standard data type variables and the IEC 61131-3 function blocks. The system supports a maximum of 32766 bytes for the data synchronization.

udtSyncData

The data exchange between the individual function blocks for the data synchronization is done by the means of the “udtSyncData” in-/out-parameter. Therefore, at each individual function block the same variable must be applied to this parameter.

NOTE:

- This function block may only be instantiated once in the entire project.
  There must not be any other instance of this function block in the program.

6.1.1 Description of inputs and outputs:

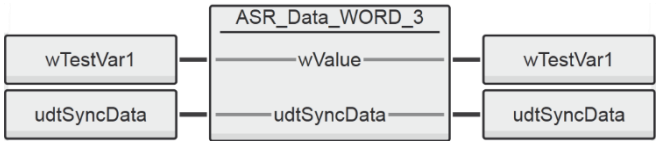
Name	Data type	Data direction	Description
iRole	INT	Input	Current role of the PLC: 0: Undefined 1: Primary 2: Backup
strOwnIP	STRING	Input	Own IP address. The proper syntax for this String is: ‘www.xxx.yyy.zzz’ (e.g. ‘192.168.0.91’)
strPartnerIP	STRING	Input	IP address of the partner PLC. The proper syntax for this String is: ‘www.xxx.yyy.zzz’ (e.g. ‘192.168.0.92’)
xUDP_SocketsActive	BOOL	Output	TRUE indicates that the UDP/IP sockets were successfully opened.
xDataStructureConsistent	BOOL	Output	TRUE indicates that the sequential arrangement of the function blocks for the data synchronization is identical on both PLCs.
iUsedBytes	INT	Output	Number of used bytes for the data exchange of the data synchronization function blocks.
xOperatingState	BOOL	Output	TRUE indicates that the data exchange for the data synchronization is in operation.
udtSyncData	ASR_UDT_SYNC_DATA	In/Out	Parameter for the data exchange with the data type specific function blocks.

6.2 Function blocks for standard data types “ASR_Data_*”

By the means of these function blocks, values of variables of standard data types can be synchronized between the PLCs. For this purpose, the active PLC (Primary) sends the values of the variables, which are applied to the “...Value” in-/out-parameter, to the passive PLC (Backup). The passive PLC receives these values and copies them to the variables connected to this in-/out-parameter.

‘*' is a placeholder for the supported data types. The FB is available for the data types BOOL, BYTE, WORD, DWORD, LWORD, SINT, INT, DINT, LINT, UDINT, UINT, USINT, ULINT, REAL, LREAL and STRING.

Example for the data type WORD:



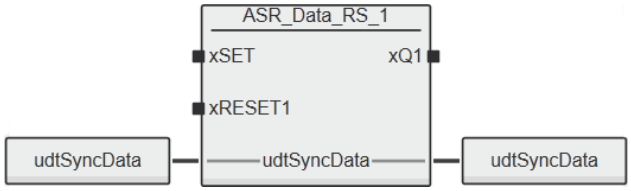
The variable, which should be synchronized between the PLCs, has to be applied to the “...Value” in-/out-parameter.
Via the “udtSyncData” in-/out-parameter the function block is connected to the responding “ASR_DataSync” main function block.

6.2.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
...Value	BOOL, BYTE, WORD, DWORD, LWORD, SINT, INT, DINT, LINT, USINT, UINT, UDINT, ULINT, REAL, LREAL, STRING	In/Out	Variable which should be synchronized.
udtSyncData	ASR_UDT_ SYNC_DATA	In/Out	Parameter for the data exchange with the "ASR_DataSync" function block.

6.3 Bistable function block “ASR_Data_RS”

This function block realizes a latching relay with resetting priority over setting. The value of the “xQ1” output parameter will be synchronized between the two redundant PLCs.

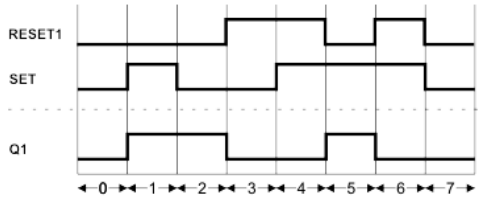


The “xQ1” output parameter is set to TRUE with a transition from FALSE to TRUE at the “xSET” input parameter. “xQ1” remains TRUE until the “xRESET1” input parameter switches from FALSE to TRUE.
If “xSET” and “xRESET1” are TRUE, the “xQ1” output parameter is reset to FALSE since the “xRESET1” input parameter has priority.

6.3.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xSET	BOOL	Input	If TRUE, output parameter “xQ1” is set.
xRESET1	BOOL	Input	If TRUE, output parameter “xQ1” is reset dominant.
xQ1	BOOL	Output	Output value
udtSyncData	ASR_UDT_SYNC_DATA	In/Out	Parameter for the data exchange with the "ASR_DataSync" function block.

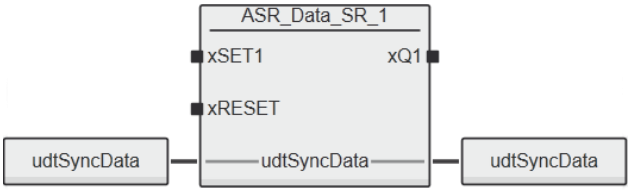
6.3.2 Timing diagram:



0	The input parameters “xSET” and “xRESET1” are FALSE. Output parameter “xQ1” is also FALSE.
1	“xSET” switches from FALSE to TRUE, “xQ1” is set to TRUE.
2	“xSET” is reset to FALSE, “xQ1” remains TRUE.
3	“xRESET1” switches from FALSE to TRUE. “xQ1” is reset to FALSE.
4	“xSET” switches from FALSE to TRUE while “xRESET1” remains TRUE. As “xRESET1” has priority over “xSET”, “xQ1” remains FALSE.
5	“xRESET1” is reset to FALSE while “xSET” remains TRUE. “xQ1” is set to TRUE.
6	“xRESET1” switches from FALSE to TRUE while “xSET” remains TRUE. As “xRESET1” has priority over “xSET”, “xQ1” is set to FALSE.
7	The input parameters “xSET” and “xRESET1” are FALSE. Output parameter “xQ1” is also FALSE.

6.4 Bistable function block “ASR_Data_SR”

This function block realizes a latching relay with setting priority over resetting. The value of the “xQ1” output parameter will be synchronized between the two redundant PLCs.

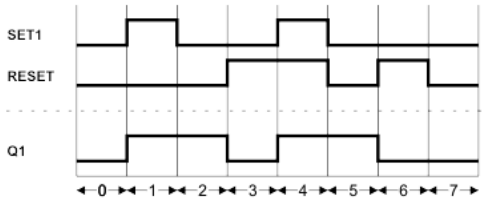


The “xQ1” output parameter is set to TRUE with a transition from FALSE to TRUE at the “xSET” input parameter. “xQ1” remains TRUE until the “xRESET1” input parameter switches from FALSE to TRUE.
If “xSET1” and “xRESET” are TRUE, the “xQ1” output parameter is set to TRUE since the “xSET1” input parameter has priority.

6.4.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xSET1	BOOL	Input	If TRUE, output parameter “xQ1” is set dominant.
xRESET	BOOL	Input	If TRUE, output parameter “xQ1” is reset.
xQ1	BOOL	Output	Output value
udtSyncData	ASR_UDT_ SYNC_DATA	In/Out	Parameter for the data exchange with the "ASR_DataSync" function block.

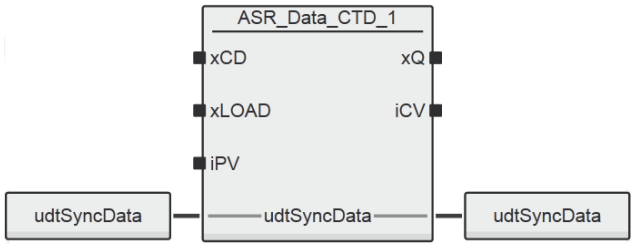
6.4.2 Timing diagram:



0	The input parameters “xSET1” and “xRESET” are FALSE. Output parameter “xQ1” is also FALSE.
1	“xSET1” switches from FALSE to TRUE, “xQ1” is set to TRUE.
2	“xSET1” is reset to FALSE, “xQ1” remains TRUE.
3	“xRESET” switches from FALSE to TRUE. “xQ1” is reset to FALSE.
4	“xSET1” switches from FALSE to TRUE while “xRESET1” remains TRUE. As “xSET1” has priority over “xRESET”, “xQ1” is set to TRUE.
5	The “xSET1” and “xRESET” input parameters are reset to FALSE. Output parameter “Q1” remains TRUE.
6	“xRESET” switches from FALSE to TRUE. “xQ1” is reset to FALSE.
7	The input parameters “xSET” and “xRESET1” are FALSE. Output parameter “xQ1” is also FALSE.

6.5 Counter function block “ASR_Data_CTD”

This counter function block counts down. The value of the “iCV” output parameter will be synchronized between the two redundant PLCs.



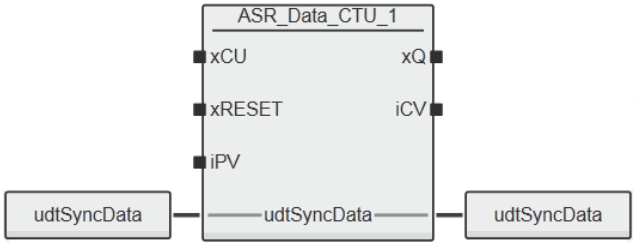
In case of a rising edge at the “xCD” input parameter and input parameter “xLOAD” is FALSE, the output parameter “iCV” decrements by one. When the value 0 is reached, output parameter “xQ” is set to TRUE and the function block stops counting.
If input parameter “xLOAD” is TRUE, the counter is initialized to the value at the “iPV” input parameter. To enable counting, the “xLOAD” input parameter must be FALSE. Otherwise, the counter will always be re-initialized.

6.5.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xCD	BOOL	Input	A rising edge decrements the “iCV” output parameter by one if input parameter “xLOAD” is FALSE.
xLOAD	BOOL	Input	If TRUE, the “iCV” output parameter is initialized with the value of the “iPV” input parameter. If FALSE, counting is enabled.
iPV	INT	Input	Preset value.
xQ	BOOL	Output	TRUE if the value of the “iCV” output parameter is 0.
iCV	INT	Output	Counter value.
udtSyncData	ASR_UDT_SYNC_DATA	In/Out	Parameter for the data exchange with the "ASR_DataSync" function block.

6.6 Counter function block “ASR_Data_CTU”

This counter function block counts up. The value of the “iCV” output parameter will be synchronized between the two redundant PLCs.



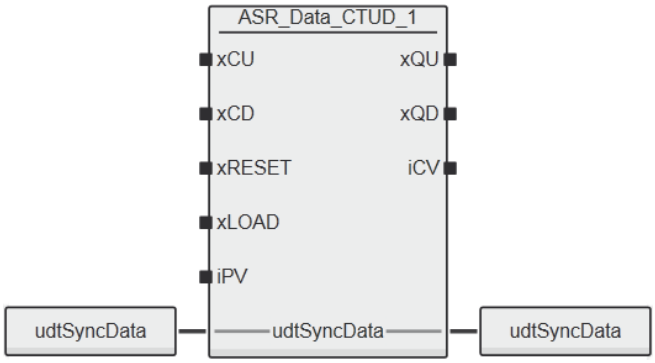
In case of a rising edge at the “xCU” input parameter and input parameter “xRESET” is FALSE, the output parameter “iCV” increments by one. When the preset value “iPV” is reached, output parameter “xQ” is set to TRUE and the function block stops counting.
If input parameter “xRESET” is TRUE, the counter is initialized with 0. To enable counting, the “xRESET” input parameter must be FALSE. Otherwise, the counter will always be re-initialized.

6.6.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xCU	BOOL	Input	A rising edge increments the “iCV” output parameter by one if input parameter “xRESET” is FALSE.
xRESET	BOOL	Input	If TRUE, the “iCV” output parameter is initialized with 0. If FALSE, counting is enabled.
iPV	INT	Input	Preset value.
xQ	BOOL	Output	TRUE if the value of the “iCV” output parameter is equal to the value of the “iPV” input parameter.
iCV	INT	Output	Counter value.
udtSyncData	ASR_UDT_SYNC_DATA	In/Out	Parameter for the data exchange with the "ASR_DataSync" function block.

6.7 Counter function block “ASR_Data_CTUD”

This counter function block counts up or down. The value of the “iCV” output parameter will be synchronized between the two redundant PLCs.



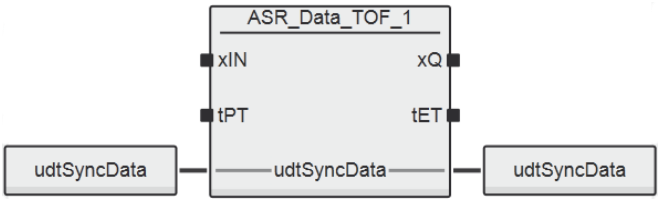
In case of a rising edge at the “xCU” input parameter and both input parameters “xRESET” and “xLOAD” are FALSE, the output parameter “iCV” increments by one. When the preset value “iPV” is reached, output parameter “xQU” is set to TRUE and the function block stops counting up.
In case of a rising edge at the “xCD” input parameter and both input parameters “xRESET” and “xLOAD” are FALSE, the output parameter “iCV” decrements by one. When the value 0 is reached, output parameter “xQD” is set to TRUE and the function block stops counting down.
If input parameter “xRESET” is TRUE, the counter is initialized with 0. If input parameter “xLOAD” is TRUE, the counter is initialized to the value at the “iPV” input parameter. To enable counting, the input parameters “xRESET” and “xLOAD” must be FALSE. Otherwise, the counter will always be re-initialized.

6.7.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xCU	BOOL	Input	A rising edge increments the “iCV” output parameter by one if input parameters “xRESET” and “xLOAD” are FALSE.
xCD	BOOL	Input	A rising edge decrements the “iCV” output parameter by one if input parameters “xRESET” and “xLOAD” are FALSE.
xRESET	BOOL	Input	If TRUE, the “iCV” output parameter is initialized with 0. If FALSE, counting is enabled.
xLOAD	BOOL	Input	If TRUE, the “iCV” output parameter is initialized with the value of the “iPV” input parameter. If FALSE, counting is enabled.
iPV	INT	Input	Preset value.
xQU	BOOL	Output	TRUE if the value of the “iCV” output parameter is equal to the value of the “iPV” input parameter.
xQD	BOOL	Output	TRUE if the value of the “iCV” output parameter is 0.
iCV	INT	Output	Counter value.
udtSyncData	ASR_UDT_SYNC_DATA	In/Out	Parameter for the data exchange with the "ASR_DataSync" function block.

6.8 Timer function block “ASR_Data_TOF”

This function block provides a timer for delayed switching off. The values of the “xQ” and “tET” output parameters will be synchronized between the two redundant PLCs. During a switch-over of the redundancy roles, the elapsed time of the delay remains frozen, consequently the elapsed time will be extended by the duration of the switch-over procedure.

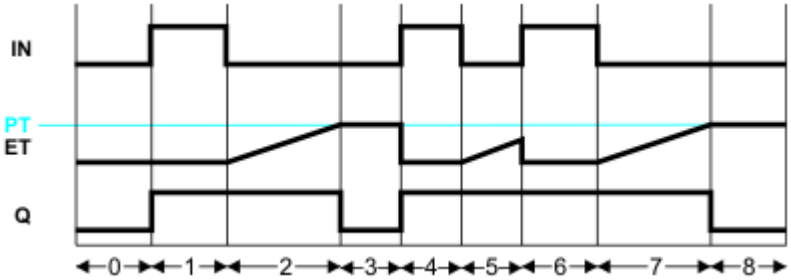


When the “xIN” input parameter is TRUE, the “xQ” output parameter is set to TRUE. If “xIN” changes from TRUE to FALSE, switching off is delayed for the duration of the time interval applied the “tPT” input parameter. After “tPT” has elapsed, output parameter “xQ” is set to FALSE. The elapsed time interval is indicated at the “tET” output parameter.

6.8.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xIN	BOOL	Input	If a falling edge is detected, the off-delay timing is started.
tPT	TIME	Input	Preset time interval for the delay.
xQ	BOOL	Output	TRUE if “xIN” is TRUE or “tET” is less than “tPT”. FALSE if “xIN” is FALSE and “tET” is equal to “tPT”.
tET	TIME	Output	Elapsed time interval.
udtSyncData	ASR_UDT_ SYNC_DATA	In/Out	Parameter for the data exchange with the "ASR_DataSync" function block.

6.8.2 Timing diagram:

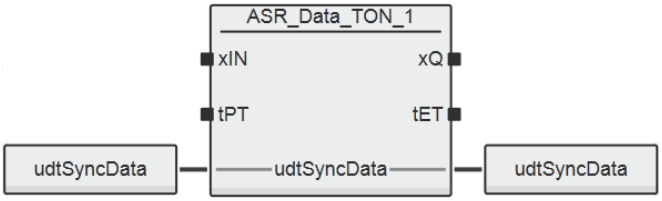


0	Input parameter “xIN” is FALSE. As a result, output parameter “xQ” is also set to FALSE.
1	“xIN” switches from FALSE to TRUE. “xQ” is set to TRUE immediately.
2	“xIN” switches from TRUE to FALSE. The measurement of the elapsed time “tET” is started.
3	When the delay time “tPT” has elapsed (“tET” >= “tPT”), “xQ” is reset to FALSE.
4	“xIN” switches from FALSE to TRUE. “xQ” is set to TRUE immediately.

5	“xIN” switches from TRUE to FALSE. The measurement of the elapsed time “tPT” is started once more.
6	“xIN” switches back to TRUE before the preset time “tPT” has elapsed. As a result, “xQ” remains TRUE.
7	“xIN” switches from TRUE to FALSE again. The measurement of the elapsed time “tPT” is started once more.
8	When the delay time “tPT” has elapsed (“tET” >= “tPT”), “xQ” is reset to FALSE.

6.9 Timer function block “ASR_Data_TON”

This function block provides a timer for delayed switching on. The value of “tET” output parameter will be synchronized between the two redundant PLCs. During a switch-over of the redundancy roles, the elapsed time of the delay remains frozen, consequently the elapsed time will be extended by the duration of the switch-over procedure.

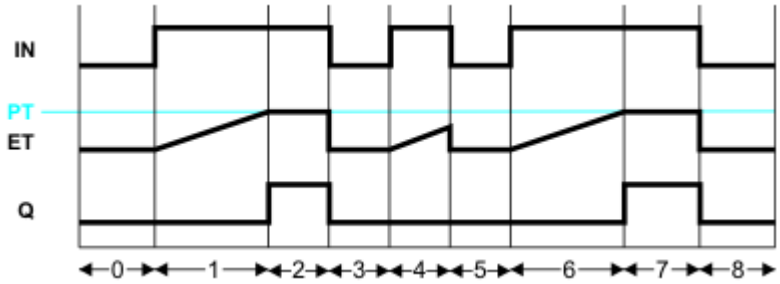


When the “xIN” input parameter is TRUE, the “xQ” output parameter is set to TRUE after the delay time has elapsed.
If “xIN” changes from FALSE to TRUE, switching on is delayed for the duration of the time interval applied the “tPT” input parameter. After “tPT” has elapsed, output parameter “xQ” is set to TRUE. The elapsed time interval is indicated at the “tET” output parameter.

6.9.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xIN	BOOL	Input	If a rising edge is detected, the on-delay timing is started.
tPT	TIME	Input	Preset time interval for the delay.
xQ	BOOL	Output	TRUE if “xIN” is TRUE and “tET” is equal to “tPT”. FALSE if “xIN” is FALSE or “tET” is less than “tPT”.
tET	TIME	Output	Elapsed time interval.
udtSyncData	ASR_UDT_ SYNC_DATA	In/Out	Parameter for the data exchange with the "ASR_DataSync" function block.

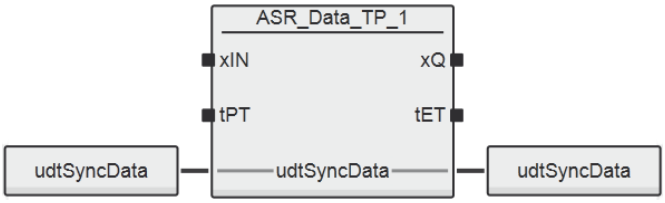
6.9.2 Timing diagram:



0	Input parameter “xIN” is FALSE. As a result, output parameter “xQ” is also set to FALSE.
1	“xIN” switches from FALSE to TRUE. The measurement of the elapsed time “tET” is started.
2	When the delay time “tPT” has elapsed (“tET” >= “tPT”), “xQ” is set to TRUE.
3	“xIN” switches from TRUE to FALSE. “xQ” is reset to FALSE immediately.
4	“xIN” switches from FALSE to TRUE. The measurement of the elapsed time “tPT” is started once more.
5	“xIN” switches back to FALSE before the preset time “tPT” has elapsed. As a result, “xQ” remains FALSE.
6	“xIN” switches from FALSE to TRUE. The measurement of the elapsed time “tPT” is started once more.
7	When the delay time “tPT” has elapsed (“tET” >= “tPT”), “xQ” is set to TRUE.
8	“xIN” switches from TRUE to FALSE. “xQ” is reset to FALSE immediately.

6.10 Timer function block “ASR_Data_TP”

This function block generates a pulse with a configurable pulse duration. The values of the “xQ” and “tET” output parameters will be synchronized between the two redundant PLCs. During a switch-over of the redundancy roles, the elapsed time of the pulse remains frozen, consequently the elapsed time will be extended by the duration of the switch-over procedure.

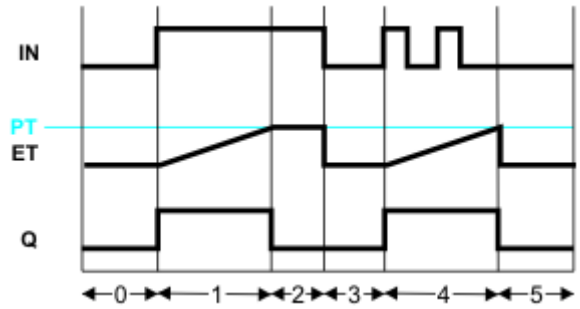


When the “xIN” input parameter switches from FALSE to TRUE, the “xQ” output parameter is set to TRUE and remains set until the delay time specified at the “tPT” input parameter has elapsed. The “xQ” output parameter remains unchanged, if the “xIN” input parameter switches from TRUE to FALSE before the specified delay time has elapsed or if this input parameter becomes TRUE for a second time before the delay time has elapsed.
The “tPT” input parameter specifies the time after which “xQ” is switched from TRUE to FALSE. The elapsed time interval is indicated at the “tET” output parameter.

6.10.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xIN	BOOL	Input	If a rising edge is detected, a pulse is generated.
tPT	TIME	Input	Preset time interval for the pulse.
xQ	BOOL	Output	TRUE if “tET” is less than “tPT”. FALSE if “tET” is equal to “tPT”.
tET	TIME	Output	Elapsed time interval.
udtSyncData	ASR_UDT_SYNC_DATA	In/Out	Parameter for the data exchange with the "ASR_DataSync" function block.

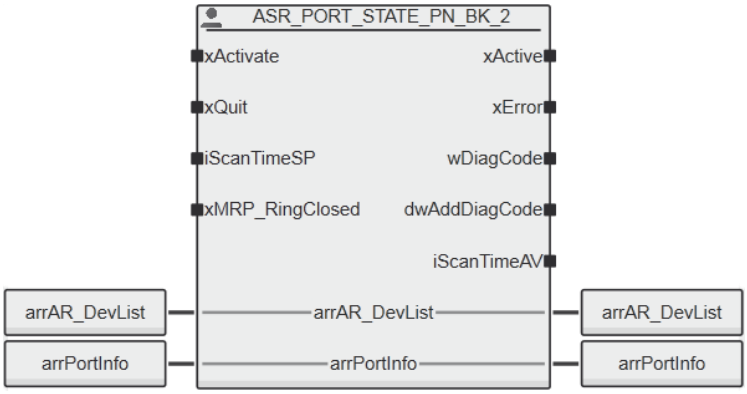
6.10.2 Timing diagram:



0	Input parameter “xIN” is FALSE. As a result, output parameter “xQ” is also set to FALSE.
1	“xIN” switches from FALSE to TRUE. “xQ” is set to TRUE immediately and the measurement of the elapsed time “tET” is started.
2	The delay time “tPT” has elapsed (“tET” >= “tPT”), “xQ” is reset to FALSE independent of the “xIN” input parameter.
3	“xIN” switches from TRUE to FALSE. “xQ” remains FALSE.
4	“xIN” switches from FALSE to TRUE. “xQ” is set to TRUE immediately and the measurement of the elapsed time “tET” is started. Before the delay time “tPT” has elapsed, “xIN” switches repeatedly from TRUE to FALSE and vice versa. “xQ” remains TRUE as the delay time “tPT” has not expired yet.
5	The delay time “tPT” has elapsed (“tET” >= “tPT”), “xQ” is reset to FALSE immediately.

7 Function block for gathering port states from PROFINET IO devices

By the means of this function block the port states from PROFINET IO devices can be read directly by the PLC via PROFINET acyclic services.



arrAR_DevList

The Name of station and the corresponding Device ID of the PROFINET IO devices must be applied to the “arrAR_DevList” in-/out-parameter. This in-/out-parameter is an Array of Struct with 64 elements, so that a maximum number of 64 PROFINET IO devices can be used.

xActivate

“TRUE” at the “xActivate” input parameter activates the function block. After the activation the function block starts with the initialization phase. During this initialization phase the applied PROFINET IO device names are converted into unique IDs by the means of special PROFINET function blocks. This process can take several seconds. The successful termination of the initialization phase is indicated with “TRUE” at the “xActive” output parameter. As soon as this output parameter is set to “TRUE”, the function block requests cyclically the port states from the specified PROFINET IO devices. “FALSE” at the “xActivate” input parameter deactivates the function block and requesting of the port states is stopped.

xError

The appearance of an error is indicated by the "xError" output parameter. The "wDiagCode" and "dwAddDiagCode" output parameters display the cause of the error. The values are valid within the cycle, in which the rising edge of the "xError" output parameter occurred. The acknowledgment of an error message, which occurred during requesting the port states (“xActive” output parameter is “TRUE”), is carried out by a rising edge of the "xQuit" input parameter. The acknowledgment of an error message during the initialization phase of the function block (“xActive” output parameter is “FALSE”) is only possible by deactivating the function block.

xMRP_RingClosed

If the bus couplers are integrated into an MRP network ring structure and a signal is available indicating that the ring structure is closed, reading of the port states can be skipped by setting the “xMRP_RingClosed” input parameter to “TRUE”. As soon as this input parameter is set to

"TRUE" the reading of port states via PROFINET acyclic services is stopped and the values of the elements "xDataValid", "xLinkPort1" and "xLinkPort2" for all PROFINET IO devices are forced to "TRUE".

7.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xActivate	BOOL	Input	With TRUE the function block is activated and reading of the port states is started. FALSE deactivates the function block and reading of the port states is stopped.
xQuit	BOOL	Input	With a rising edge at this input parameter, error messages that occurred during requesting the port states are acknowledged.
iScanTimeSP	INT	Input	Desired scanning cycle time in seconds.
xMRP_RingClosed	BOOL	Input	With TRUE the reading of the port states via PROFINET acyclic services is stopped and the values of the elements "xDataValid", "xLinkPort1" and "xLinkPort2" for all PROFINET IO devices are set to TRUE.
arrAR_DevList	ASR_ARR_PORT_DEV_LIST	Input	List of Name of stations and the corresponding Device ID of the PROFINET IO devices from which the port states should be collected. This input parameter is an Array of Struct with 64 elements. The Name of station has to be applied to the element "strStationName" and the corresponding Device ID to the element "wDeviceID".
xActive	BOOL	Output	TRUE at this output parameter indicates that the initialization phase of the function block has been successfully terminated and the function block is reading the port states cyclically.
xError	BOOL	Output	This output indicates with a rising edge that an error has occurred. You can read the corresponding error codes at the output parameters "wDiagCode" and "dwAddDiagCode" during the cycle in which the rising edge occurs.
wDiagCode	WORD	Output	Indicates the cause of the error.
dwAddDiagCode	DWORD	Output	Indicates additional information to the cause of the error.
iScanTimeAV	INT	Output	Actual scanning cycle time in seconds.
arrPortInfo	ASR_ARR_PORT_INFO	Output	Port states from the specified PROFINET IO devices. This output parameter is an Array of Struct, in which each field contains the port states of a PROFINET IO device corresponding to the PROFINET IO device names at the in-/out-parameter "arrAR_DevList". For each PROFINET IO device the following values are available: - xDataValid: This element indicates whether the last reading of the port states from the PROFINET IO device was successful (TRUE) or unsuccessful (FALSE). The following values are only valid if this bit is TRUE. - xLinkPort1 and xLinkPort2: Link state of the port.

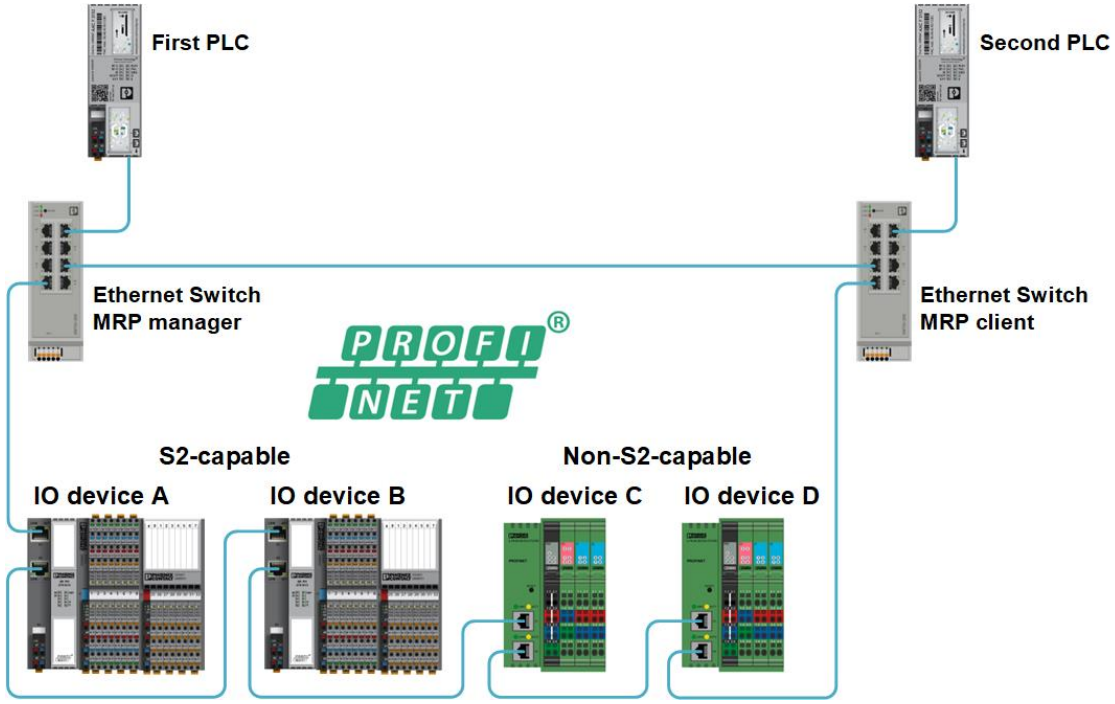
7.2 Error Codes

Error Code	Description
C401	Indicates that an error occurred at the initialization phase during the conversion of the PROFINET IO device names into unique IDs. One possible reason for such an error could be that the appropriate PROFINET IO device name doesn't exist in the bus configuration of the project. The "dwAddDiagCode" output parameter indicates the error code of the conversion function (High Word) and the index of the relevant PROFINET IO device (Low Word). The acknowledgement of this error is done by deactivating the function block.
C402	Indicates that the function block for reading the port states reported an error while a request was sent to a PROFINET IO device. The "dwAddDiagCode" output parameter contains the error code of the function block (High Word) and the index of the relevant PROFINET IO device (Low Word). The acknowledgment of this error message is carried out at a rising edge of the "xQuit" input parameter.
C403	Time-out detection of the internal state machine. The "dwAddDiagCode" output parameter indicates the step of the internal state machine (High Word) and the index of the relevant PROFINET IO device (Low Word) at which this error has occurred. The acknowledgment of this error message is carried out at a rising edge of the "xQuit" input parameter.

8 Function block for integration of non-S2-capable PROFINET IO devices

The Applicative System Redundancy (ASR) is based on the mechanism of the PROFINET S2 System Redundancy. For this reason, it is required that PROFINET IO devices used in this system support PROFINET System Redundancy type S2. The S2 System Redundancy layer ensures that both redundant controllers (First and Second PLC) are allowed to establish PROFINET communication to the IO devices simultaneously and that redundancy switch-over of the PLC takes place seamlessly in regard to PROFINET communication with the IO devices. If PROFINET IO devices do not support PROFINET S2 System Redundancy, only one of the two controllers can establish PROFINET communication with these IO devices. The attempt to establish a PROFINET connection by the second controller would then be rejected by these non-S2-capable IO devices.

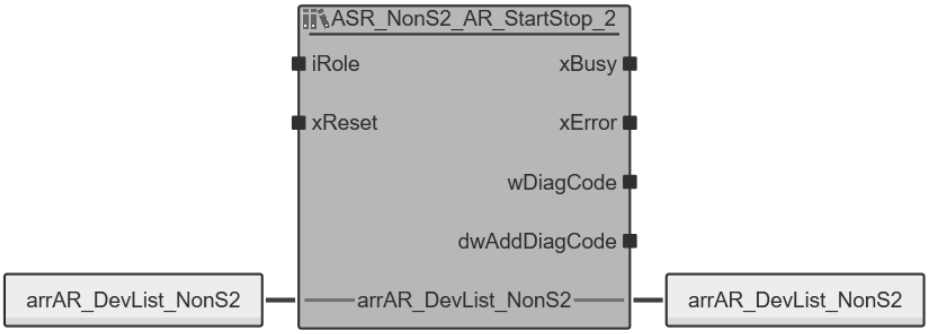
For some applications it may be necessary and useful to integrate non-S2-capable PROFINET IO devices into an ASR system. However, as only one controller is allowed to establish PROFINET communication to such IO devices, the function block described below makes sure that only the controller with the Primary redundancy role establishes PROFINET connection to the non-S2-capable IO devices.



Significant restrictions when using non-S2-capable PROFINET IO devices:

- Since only the Primary controller establishes PROFINET communication to non-S2-capable IO devices, the Backup controller does not receive input process data from these IO devices.
- In the event of a redundancy switch-over of the controllers, this switch-over process is not seamless for the non-S2-capable IO devices, since the “old” Primary controller first has to release its PROFINET connection before the “new” Primary controller can establish PROFINET communication with these IO devices.

8.1 Function block “ASR_NonS2_AR_StartStop”



8.1.1 Description of inputs and outputs

Name	Data type	Data direction	Description
iRole	INT	Input	Current role of the PLC: 0: Undefined 1: Primary 2: Backup
xReset	BOOL	Input	With a rising edge at this input parameter, error messages that occurred during activation/deactivation of the PROFINET ARs are acknowledged.
arrAR_DevList_NonS2	ASR_ARR_STR_PN_DEV_LIST	Input	“Name of station” list of all non-S2-capable PROFINET IO devices.
xBusy	BOOL	Output	TRUE indicates that the function block is busy with the activation / deactivation of the requested PROFINET ARs. A falling edge indicates that the operation is completed.
xError	BOOL	Output	This output indicates with a rising edge that an error has occurred. You can read the corresponding error codes at the output parameters "wDiagCode" and "dwAddDiagCode" during the cycle in which the rising edge occurs.
wDiagCode	WORD	Output	8xxx: Indicates status messages. Cxxx: Indicates the responding error code in case of an error.
dwAddDiagCode	DWORD	Output	Indicates additional information on the status messages or the cause of an error.

8.1.2 Diagnostic

Status messages:

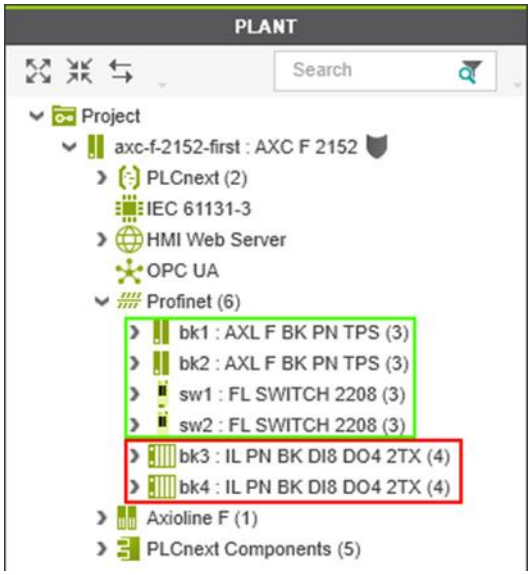
wDiagCode	Description
8000	Indicates that the function block is activated and initialized.
8601	The function block is busy with the activation of the requested PROFINET ARs. The “dwAddDiagCode” output parameter indicates the index of the PROFINET IO device for which the command is currently executed.
8602	The function block is busy with the deactivation of the requested PROFINET ARs. The “dwAddDiagCode” output parameter indicates the index of the PROFINET IO device for which the command is currently executed.

Error codes:

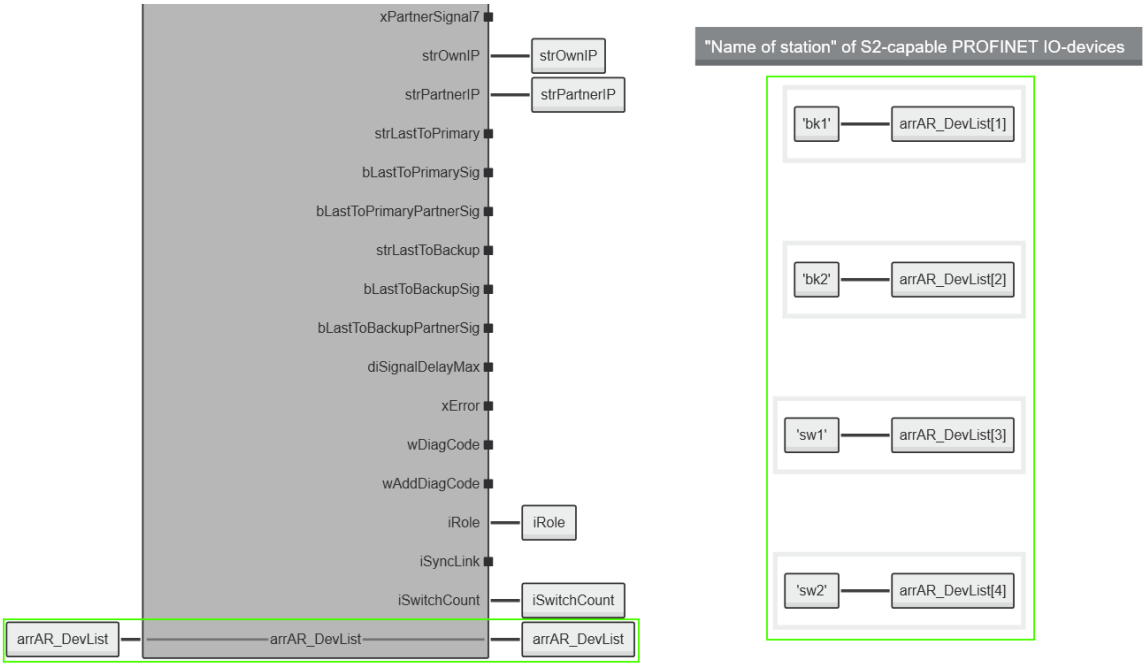
wDiagCode	Description
C403	An error occurred while trying to activate / deactivate a PROFINET AR. The “dwAddDiagCode” output parameter indicates the error code of the relevant firmware service (High Word) and the index of the relevant IO device (Low Word) at which this error has occurred.
C404	Time-out detection of the internal state machine. The “dwAddDiagCode” output parameter indicates the step of the internal state machine (High Word) and the index of the relevant IO device (Low Word) at which this error has occurred.
C405	The sum of S2-capable and non-S2-capable PROFINET IO devices exceeds the maximum allowed number of IO devices.

8.1.3 Example of use

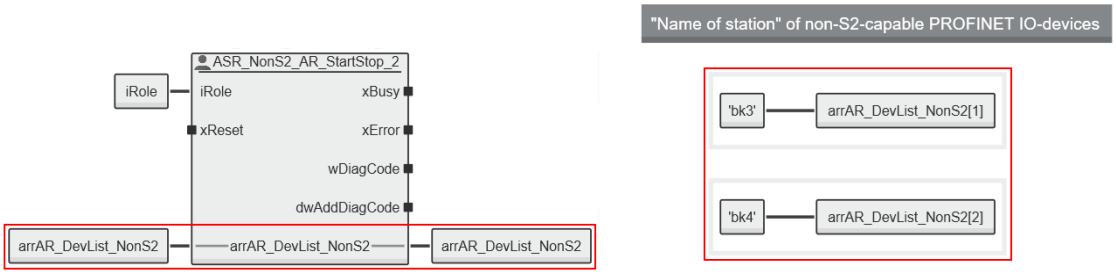
At the following PROFINET configuration example six IO devices are used in total. Four of them support PROFINET S2 system redundancy and two of them does not:



The “Name of Station” of the S2-capable PROFINET IO devices are stored at the “arrAR_DevList” InOut parameter of the “ASR_ContrRed_” function block:

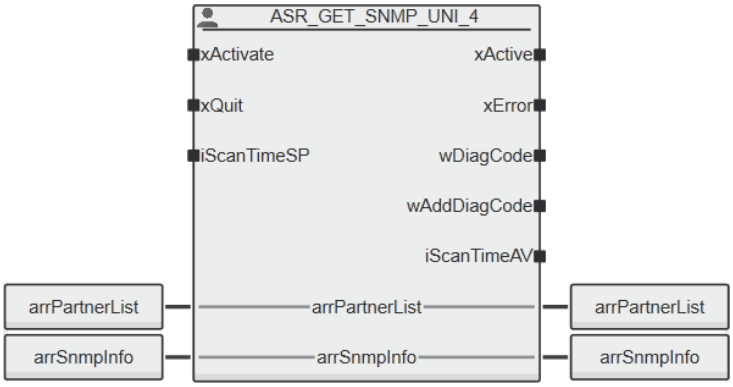


The “iRole” output parameter of the “ASR_ContrRed_” function block needs to be connected to the “iRole” input parameter of the “ASR_NonS2_AR_StartStop” function block. The “Name of Station” of the non-S2-capable PROFINET IO devices have to be stored at the “arrAR_DevList_NonS2” InOut parameter of the “ASR_NonS2_AR_StartStop” function block:



9 Function block for gathering SNMP values from managed switches

By the means of this function block SNMP values from managed switches with up to 16 ports can be read directly by the PLC via the SNMP v2 protocol.



arrPartnerList

The IP addresses and the number of ports (maximum 16) of the managed switches must be applied to the “arrPartnerList” in-/out-parameter. This in-/out-parameter is an Array of Struct with 30 elements, so that a maximum number of 30 IP addresses can be used. The proper syntax for an IP address is: ‘/IP=www.xxx.yyy.zzz’ (e.g. ‘/IP=192.168.0.21’).

xActivate

“TRUE” at the “xActivate” input parameter activates the function block. After the activation, the function block requests cyclically the SNMP values from the specified managed switches. The desired scanning time can be adjusted via the “iScanTimeSP” input parameter. The received SNMP values are stored at the “arrSnmpInfo” output parameter. “FALSE” at the “xActivate” input parameter deactivates the function block and requesting the SNMP values is stopped.

xError

The appearance of an error is indicated by the "xError" output parameter. The "wDiagCode" and "wAddDiagCode" output parameters display the cause of the error. The values are valid within the cycle, in which the rising edge of the "xError" output parameter occurred. The acknowledgment of an error message is carried out by a rising edge of the "xQuit" input parameter.

9.1 Description of inputs and outputs:

Name	Data type	Data direction	Description
xActivate	BOOL	Input	With TRUE the function block is activated and reading of the SNMP values is started. FALSE deactivates the function block and reading of the SNMP values is stopped.

xQuit	BOOL	Input	With a rising edge at this input parameter error messages, which occurred during requesting the SNMP values, are acknowledged.
iScanTimeSP	INT	Input	Desired scanning cycle time in seconds.
arrPartnerList	ASR_ARR_ PARTNER_ LIST	Input	List of IP addresses and the number ports of the managed switches from which the SNMP values should be collected. This in-/out-parameter is an Array of Struct with 30 elements. The IP address has to be applied to the element “IpAddress” and the number of ports of the switch to the element “NumberOfPorts”. The proper syntax for an IP address is: '/IP=www.xxx.yyy.zzz' (e.g. '/IP=192.168.0.21').
xActive	BOOL	Output	TRUE at this output parameter indicates that the function block is requesting the SNMP values cyclically.
xError	BOOL	Output	This output parameter indicates with a rising edge that an error has occurred. You can read the corresponding error codes at the "wDiagCode" and "wAddDiagCode" output parameters during the cycle in which the rising edge occurs.
wDiagCode	WORD	Output	Indicates the cause of the error.
wAddDiagCode	DWORD	Output	Indicates additional information to the cause of the error.
iScanTimeAV	INT	Output	Actual scanning cycle time in seconds.
arrSnmpInfo	ASR_ARR_ SNMP_ INFO	Output	SNMP values from the specified managed switches. This in-/out-parameter is an Array of Struct, in which each field contains the SNMP values of a managed switch corresponding to the IP addresses at the in-/out-parameter “arrPartnerList”. The element “DataValid” indicates whether the last request of SNMP values to this managed switch was successful (TRUE) or unsuccessful (FALSE). The meaning of the individual information read from the managed switches is explained separately in the table below.

Description of the elements of the structure “SnmpInfo”:

Most elements are available for each port and therefore presented in an Array.

Name	Data type	Description
SysUpTime	UDINT	Time in hundreds of seconds since the last reset of the managed switch.
PortSpeed	UDINT	Current bandwidth of the port in bits per second.
OperStatus	USINT	Current operational state of the port. Typical possible values: 1…LinkUp, 2…LinkDown, 3…Testing, 4…Unknown, 5…Dormant, 6…Not present
InOctets	UDINT	Total number of received octets.
InUcasts	UDINT	Total number of received unicast packets.
InNUcasts	UDINT	Total number of received broad and multicast packets.
InDiscards	UDINT	Total number of discarded inbound packets.
InErrors	UDINT	Total number of inbound packets with errors.
InUtilization	REAL	Calculated inbound utilization in percent.
InNUcastsPerSec	UINT	Calculated number of received broad and multicast packets per second.
InCorrupt	REAL	Calculated error rate of received packets in percent.
OutOctets	UDINT	Total number of transmitted octets.

OutUcasts	UDINT	Total number of transmitted unicast packets.
OutNUcasts	UDINT	Total number of transmitted broad and multicast packets.
OutDiscards	UDINT	Total number of discarded outbound packets.
OutErrors	UDINT	Total number of outbound packets with errors.
OutUtilization	REAL	Calculated outbound utilization in percent.
OutNUcastsPerSec	UINT	Calculated number of transmitted broad and multicast packets per second.
OutCorrupt	REAL	Calculated error rate of transmitted packets in percent.

9.2 Error Codes

Error Code	Description
C401	Indicates that the syntax of an applied string for the IP address at the "arrPartnerList" in-/out-parameter is wrong (proper syntax: '/IP=www.xxx.yyy.zzz'). The “wAddDiagCode” output parameter indicates the index of the relevant switch.
C402	Indicates that the SNMP-Client function block reported an error while a SNMP request was sent to a managed switch. The "wAddDiagCode" output parameter contains the error code of the SNMP-Client function block.
C403	Time-out detection of the internal state machine. The "wAddDiagCode" output parameter indicates the step of the internal state machine (High Byte) and the index of the relevant managed switch (Low Byte) at which this error has occurred.

The acknowledgment of these error messages is carried out at a rising edge of the "xQuit" input parameter.