SIEMENS
Ingenuity for life

Configuring Flexible F-Link Communication

SIMATIC Safety Integrated

https://support.industry.siemens.com/cs/ww/en/view/109768964

Siemens Industry Online Support

Legal information

Use of application examples

Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. You yourself are responsible for the proper and safe operation of the products in accordance with applicable regulations and must also check the function of the respective application example and customize it for your system.

Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. The application examples are not required to undergo the customary tests and quality inspections of a chargeable product; they may have functional and performance defects as well as errors. It is your responsibility to use them in such a manner that any malfunctions that may occur do not result in property damage or injury to persons.

Disclaimer of liability

Siemens shall not assume any liability, for any legal reason whatsoever, including, without limitation, liability for the usability, availability, completeness and freedom from defects of the application examples as well as for related information, configuration and performance data and any damage caused thereby. This shall not apply in cases of mandatory liability, for example under the German Product Liability Act, or in cases of intent, gross negligence, or culpable loss of life, bodily injury or damage to health, non-compliance with a guarantee, fraudulent non-disclosure of a defect, or culpable breach of material contractual obligations. Claims for damages arising from a breach of material contractual obligations shall however be limited to the foreseeable damage typical of the type of agreement, unless liability arises from intent or gross negligence or is based on loss of life, bodily injury or damage to health. The foregoing provisions do not imply any change in the burden of proof to your detriment. You shall indemnify Siemens against existing or future claims of third parties in this connection except where Siemens is mandatorily liable.

By using the application examples you acknowledge that Siemens cannot be held liable for any damage beyond the liability provisions described.

Other information

Siemens reserves the right to make changes to the application examples at any time without notice. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.

The Siemens terms of use (https://support.industry.siemens.com) shall also apply.

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 protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the Internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For additional information on industrial security measures that may be implemented, please visit https://www.siemens.com/industrialsecurity.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customer's exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed at: https://www.siemens.com/industrialsecurity.

Table of contents

1 Introduction

1.1 Overview

Flexible F-Link offers the possibility of fail-safe CPU-CPU communication. The failsafe data is exchanged as failsafe arrays between the F-CPUs via standard communication mechanisms.

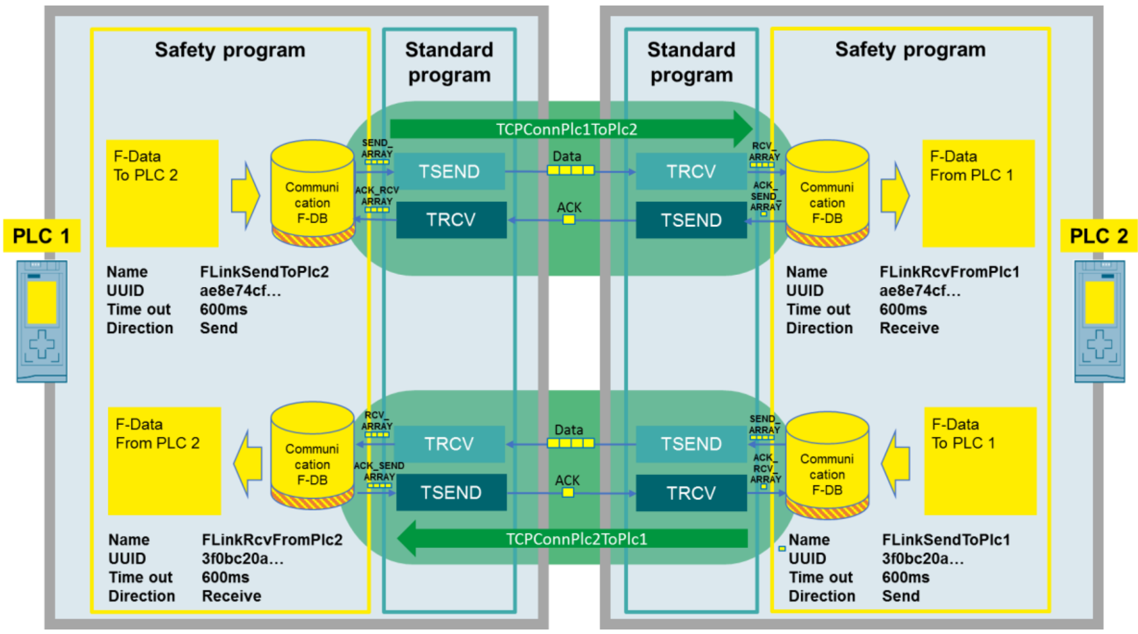
This application example demonstrates the use of Flexible F-Link communication between two controllers via a TCP connection. The example shows which points must be observed when configuring a Flexible F-Link connection. In addition, you will find a comparison of communication via Flexible F-Link and safety-related IO controller-IO controller communication or IO controller-I-device communication with SENDDP/RCVDP.

1.2 Operating principle

In this application example, safety-related data is transferred between two controllers (here: PLC 1 and PLC 2) in both directions using the Flexible F-Link. Two Flexible F-Link connections must be used for this purpose. It is recommended to use a separate TCP connection for each Flexible F-Link connection for the transmission of data on the standard channel. The data can be sent and acknowledged via the same TCP connection. The function modules "TSEND" and "TRCV" are used for this purpose.

The following figure shows the principle of data transmission:

Figure 1-1: Principle of data transmission between PLC 1 and PLC 2



1.3 Components used

This application example was created with these hardware and software components:

Table 1-1

Components	Quantity	Article number	Note
CPU 1516F-3 PN/DP	1	6ES7 516-3FN01-0AB09	FW V2.6
CPU 1512SP F-1 PN	1	6ES7 512-1SK01-0AB0	FW V2.6
SIMATIC Memory Card	2	6ES7 954-8LF02-0AA0	Accessories
STEP 7 Professional V15.1	1	6ES7810-5CC12-0YA5	Engineering
Safety Advanced V15.1	1	6ES7 833-1FA15-0YA5	Engineering

Note The functionality was tested with the specified hardware components. You can also use similar products that differ from the above list. In such a case, note that changes may be necessary in the sample project (e.g. different addresses).

Requirements:

- F-CPUs S7-1500 Firmware V2.0 or higher
- F-CPUs S7-1200 Firmware V4.2 or higher
- Safety System Version V2.2 or higher

This application example consists of the following components:

Table 1-2

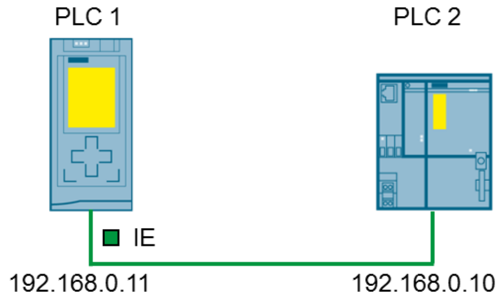
Component	File name	Note
Documentation	109768964_Flexible_F-Link_DOC_V10_en.pdf	
TIA Portal project	109768964_Flexible_F-Link_PROJ_V10.zip	

2 Engineering

2.1 Hardware setup

The following figure schematically shows the hardware structure of the application example:

Figure 2-1

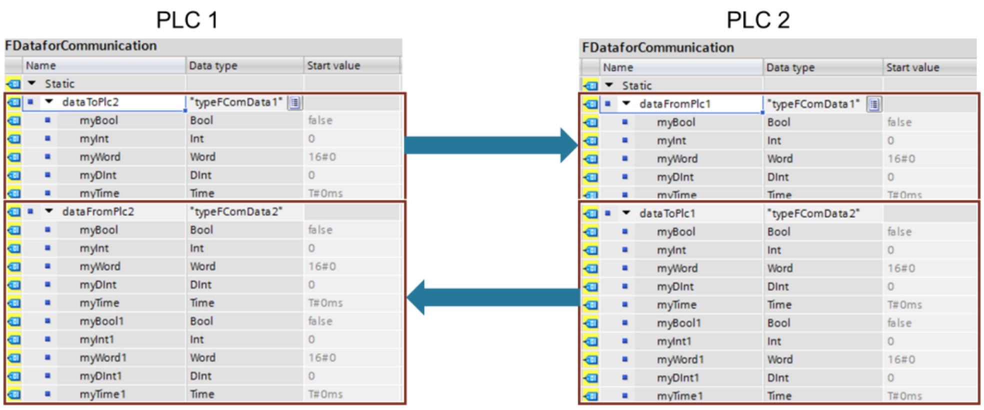


2.2 Configuring the F-Link connection

2.2.1 Configuration on both controllers

In this example application a bidirectional data exchange takes place. The following data is to be transmitted in a safety-related manner:

Figure 2-2



PLC 1

FDataforCommunication

Name	Data type	Start value
Static		
dataToPlc2	"typeFComData1"	
myBool	Bool	false
myInt	Int	0
myWord	Word	16#0
myDInt	DInt	0
myTime	Time	T#0ms
dataFromPlc2	"typeFComData2"	
myBool	Bool	false
myInt	Int	0
myWord	Word	16#0
myDInt	DInt	0
myTime	Time	T#0ms
myBool1	Bool	false
myInt1	Int	0
myWord1	Word	16#0
myDInt1	DInt	0
myTime1	Time	T#0ms

PLC 2

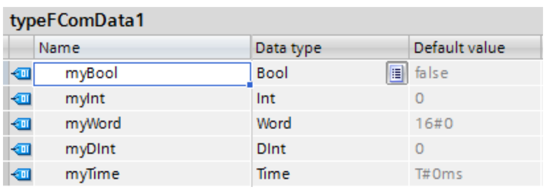
FDataforCommunication

Name	Data type	Start value
Static		
dataFromPlc1	"typeFComData1"	
myBool	Bool	false
myInt	Int	0
myWord	Word	16#0
myDInt	DInt	0
myTime	Time	T#0ms
dataToPlc1	"typeFComData2"	
myBool	Bool	false
myInt	Int	0
myWord	Word	16#0
myDInt	DInt	0
myTime	Time	T#0ms
myBool1	Bool	false
myInt1	Int	0
myWord1	Word	16#0
myDInt1	DInt	0
myTime1	Time	T#0ms

Proceed as follows to configure the Flexible F-Link communication:

1. PLC 1: Create an F-conform PLC data type (typeFComData1) for the data to be transferred from PLC 1 to PLC 2:

   Figure 2-3: typeFComData1



typeFComData1

Name	Data type	Default value
myBool	Bool	false
myInt	Int	0
myWord	Word	16#0
myDInt	DInt	0
myTime	Time	T#0ms

2. PLC 1: Create an F-conform PLC data type (typeFComData2) for the data to be transferred from PLC 2 to PLC 1:

Figure 2-4: typeFComData2

typeFComData2

Name	Data type	Default value
myBool	Bool	false
myInt	Int	0
myWord	Word	16#0
myDInt	DInt	0
myTime	Time	T#0ms
myBool1	Bool	false
myInt1	Int	0
myWord1	Word	16#0
myDInt1	DInt	0
myTime1	Time	T#0ms

3. Copy both PLC data types into the PLC 2 controller. If necessary, use the project library or global library.
4. Configure two TCP connections:
   a. **TcpConnPlc1ToPlc2** for the data transmission from PLC 1 to PLC 2 as well as the corresponding acknowledgement from PLC 2 back to PLC 1 (see Figure 2-6).
   b. **TcpConnPlc2ToPlc1** for the data transmission from PLC 2 to PLC 1 as well as the corresponding acknowledgement from PLC 1 back to PLC 2 (see Figure 2-7).

Figure 2-5: Overview of the configured TCP connections

Local connection n...	Local end point	Local ID (hex)	Partner ID ...	Partner	Connection type
TcpConnPlc1ToPlc2	PLC2 [CPU 1512SP F-1 PN]	100	100	PLC1 [CPU 151...	TCP connection
TCPConnPlc1ToPlc2	PLC1 [CPU 1516F-3 PN/DP]	100	100	PLC2 [CPU 1512SP ...	TCP connection
TCPConnPlc2ToPlc1	PLC2 [CPU 1512SP F-1 PN]	101	101	PLC1 [CPU 1516F-...	TCP connection
TCPConnPlc2ToPlc1	PLC1 [CPU 1516F-3 PN/DP]	101	101	PLC2 [CPU 1512SP ...	TCP connection

Figure 2-6: Details of the connection TcpConnPlc1ToPlc2

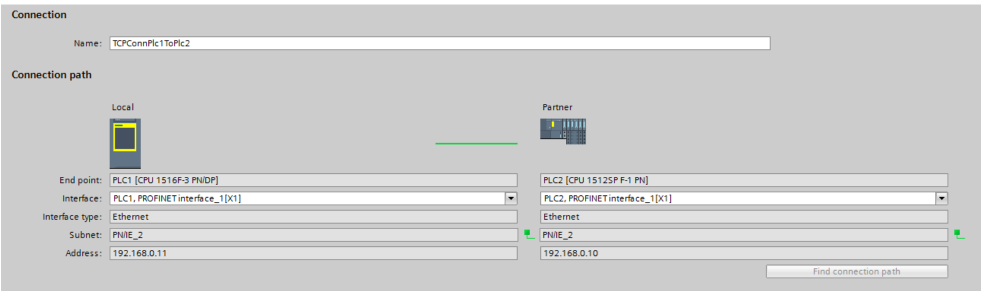
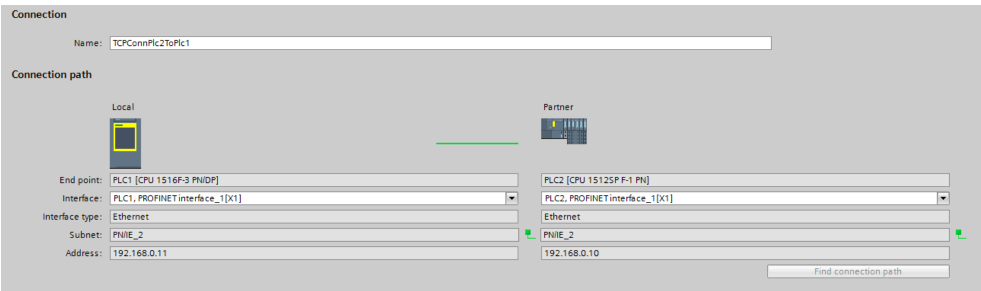


Figure 2-7 Details of the connection TcpConnPlc2ToPlc1



As an alternative to the permanently configured connections, you can also implement the connection management via the blocks "TCON" / "TDISCON".

2.2.2 Configuration PLC 1

1. Create an F-Link communication in the Safety Administration Editor with the direction "Send" (FLinkSendToPlc2) and one with the direction "Receive" (FLinkRcvFromPlc2). Select the appropriate data type for the data to be transferred and the transfer direction and set the F monitoring time. (see Figure 2-8). To help you determine the F monitoring time, you can use the table with the S7 reaction times. The table can be found under the following link:

   https://support.industry.siemens.com/cs/ww/en/view/93839056

   The largely unique F communication UUID is generated by the system itself when a communication is created.

   Figure 2-8: Overview of the created F-Link communication (PLC 1)

   Flexible F-Link settings

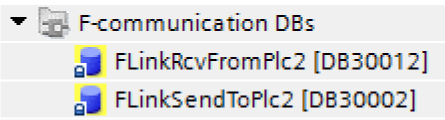
	Name	PLC Data Type	Direction	F-monitoring time: (ms)	F-communication UUID	Output data variable
1	FLinkSendToPlc2	typeFComData1	Send	600	ae8e74cf-2764-4b87-a68c-ab130f8f945a	"FLinkSendToPlc2".SE
2	FLinkRcvFromPlc2	typeFComData2	Receive	600	3f0bc20a-f26a-4d51-96aa-2f990ab8d825	"FLinkRcvFromPlc2".A
3	<Add new>					

WARNING

When a new communication is created with Flexible F-Link in the Safety Administration Editor, the system provides the unique F-communication UUID for the communication. If communications are copied within the parameter assignment table in the Safety Administration Editor or copied to another F-CPU, the F-communication UUIDs are not newly generated and are thus no longer unique. If the copy is used to configure a new communication relationship, you must take steps to ensure the uniqueness yourself. To do so, select the affected UUIDs and generate new UUIDs with "Generate UUID" in the shortcut menu. The uniqueness must be ensured in the safety printout of the safety program for acceptance of the safety program.

2. For F-Link communication, two new F-Communication DBs are created under "Program Modules\System Modules\STEP 7 Safety\F-Communication DBs":

   Figure 2-9: F-Communication DBs Generated by the System

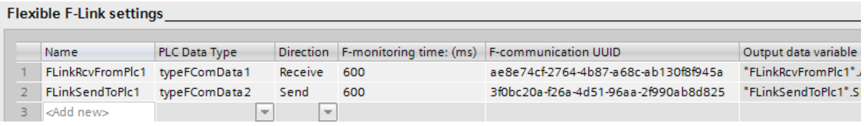


2.2.3 Configuration PLC 2

1. Create an F-Link communication in the Safety Administration Editor with the direction "Receive" (FLinkRcvFromPlc1) and one with the direction "Send" (FLinkSendToPlc1). Select the appropriate data type for the data to be transferred and the desired transfer direction.
2. Set the same F monitoring time as for PLC 1.
3. Copy the F communication UUIDs of the F-Link communication from PLC 1. Make sure that the UUIDs are assigned correctly. This ensures the uniqueness of the transmitter/receiver recognition:
   UUID from PLC 1 Transmit (FLinkSendToPlc2) = UUID from PLC 2 Receive (FLinkRcvFromPlc1)

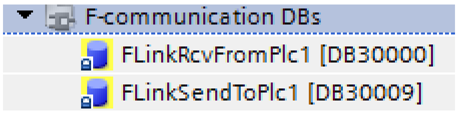
   UUID from PLC 1 Receive (FLinkRcvFromPlc2) = UUID from PLC 2 Send (FLinkSendToPlc1)

   Figure 2-10: Overview of the created F-Link communication (PLC 2)

4. For F-Link communication, two new F-Communication DBs are created under "Program Modules\System Modules\STEP 7 Safety\F-Communication DBs":

   Figure 2-11: F-Communication DBs Generated by the System

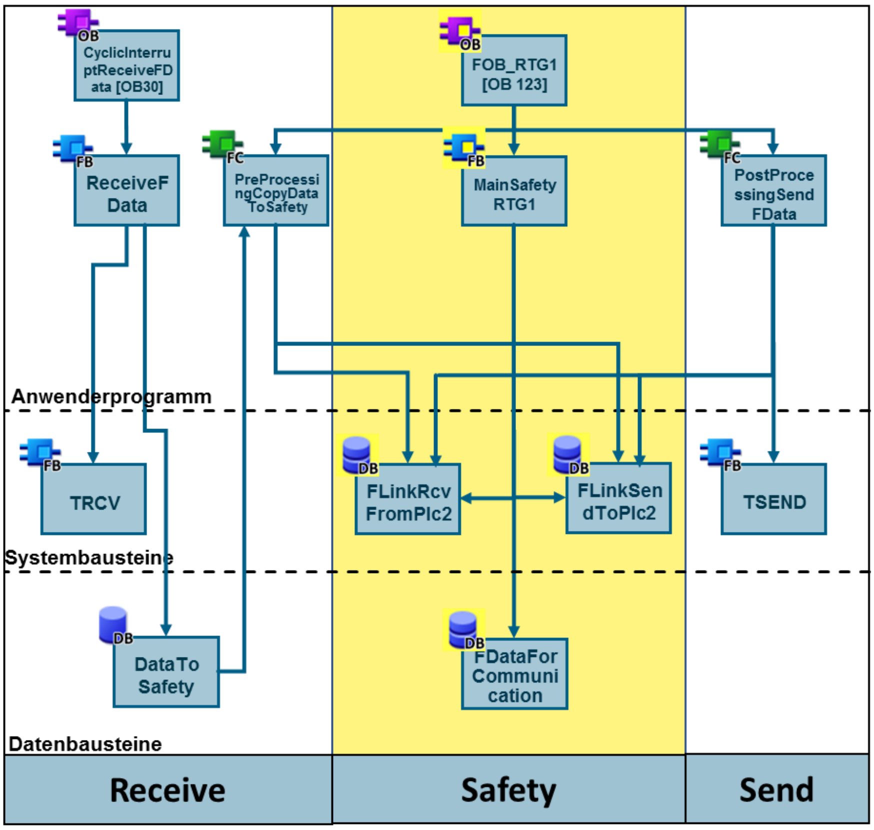
   

2.2.4 Programming

The programs in both controllers are structured according to the same principle. They are subdivided into the three areas "Safety", "Send" and "Receive". The following description explains how the PLC 1 works as an example.

The following graphic shows the program structure:

Figure 2-12 Program structure



1. The **"Safety"** area contains the safety program (MainSafetyRTG1). This block reads the received data from the F communication DB (FLinkRcvFromPlc2.RCV_DATA) or writes the data to be sent to the F communication DB (FLinkSendToPlc2.SEND_DATA).

   Figure 2-13: Reading the received data from the F-Communication-DB

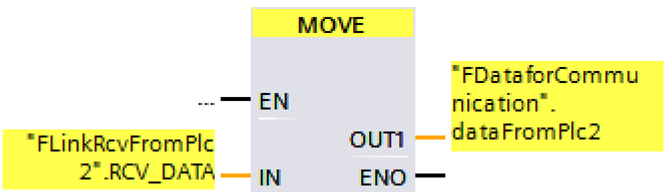
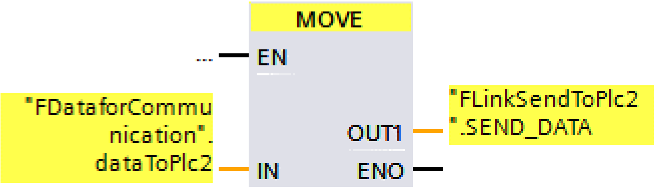
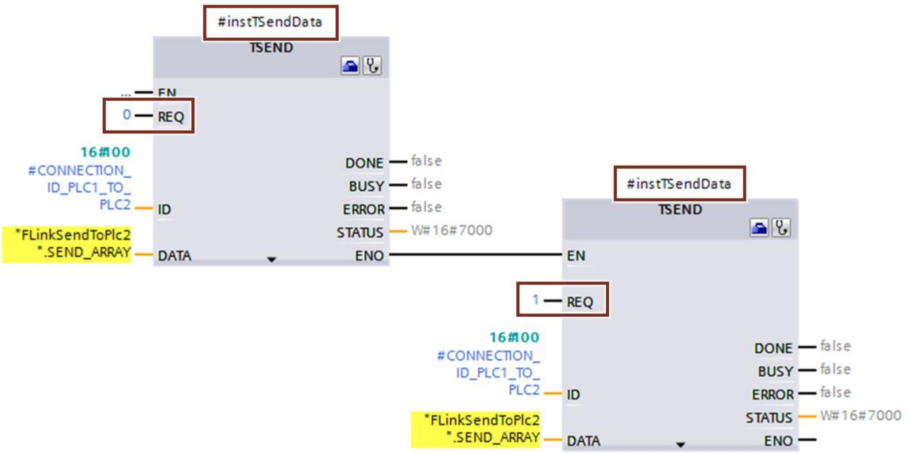


Figure 2-14: Write the data to be sent to the F communication DB.



2. In the **"Send"** area, the data is transferred to the PLC 2.
   The data to be sent is written by the system to a coded failsafe array in the F-Communication-DB (FLinkSendToPlc2.SEND_ARRAY). This is done analogously for the acknowledgement of the received data (FLinkRcvFromPlc2.ACK_SEND_ARRAY).
   These two arrays are transferred to the other controller using the communication blocks "TSEND". Transmission takes place in the postprocessing of the F sequence group and is thus called directly after the F user program (cycle time of the F-OB with PLC 1 = 100 ms or PLC 2 = 150 ms).
   The block "TSEND" is first called with REQ = 0 and then with REQ = 1. This generates a positive edge at the input in each cycle and enables faster transmission. It is important that the same instance is used for each of the two calls.

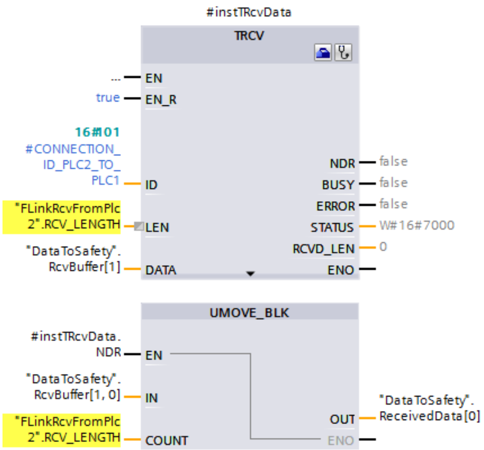
   Figure 2-15: Send the coded F-array



   The transmission of the coded F-array to acknowledge the received data is analog.

3. The data is received from PLC 2 in the "Receive" program area.
   A cyclic OB (CyclicInterruptReceiveFData) is used for this purpose. This OB must be called faster than the send cycle of the PLC 2. This reduction ensures that there is no accumulation of data or overflow in the CPU-internal communication stack (for further details see chapter 3.1 Basics Open User Communication / TCP).
   A ratio of Send is recommended for this: Receiving from 1 : 5 (Send cycle PLC 2 = 150 ms → OB for receiving in PLC 1 = 30 ms).

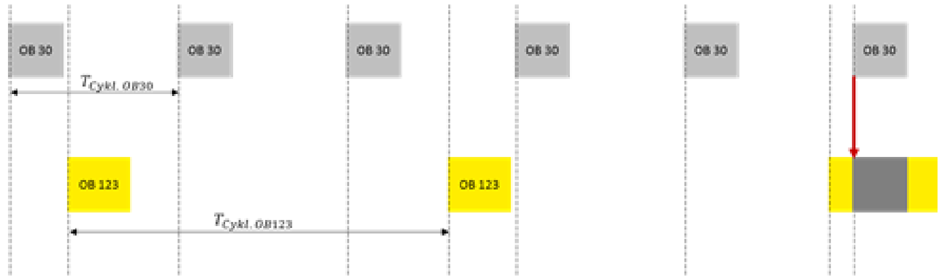
Figure 2-16: Reception of the coded F-Array



The coded F-array for acknowledging the transmitted data is received analogously.

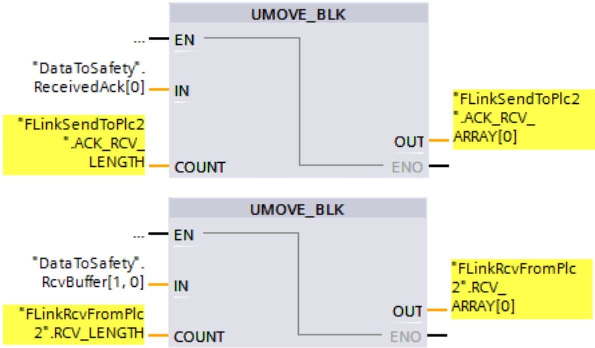
Data inconsistencies may occur due to different priorities of the cyclic OB and the F-process group: If the cyclic OB has a higher priority than the F-process group, the F-program can be interrupted by the cyclic OB.

Figure 2-17: Interruption of the F-program by OB30 with higher priority



To ensure data consistency, the data is therefore first written to a receive buffer (DataToSafety.RcvBuffer). At this point, the receive buffer is a byte array with the length of the maximum expected data of a Flexible F-Link communication (100 bytes user data + 22 bytes for UUID and checksum). The data is then transferred to the security program in the preprocessing of the F process group using the non-breakable statement "UMOVE_BLK".

Figure 2-18: PreprocessingCopyDataToSafety: Consistent data transfer into the F-program



Note

Further information on data transfer from the standard user program to the safety program can be found in the Programming and Operating Manual "SIMATIC Safety - Configuration and Programming":

https://support.industry.siemens.com/cs/ww/en/view/54110126

2.3 Operation

If the two controllers are started at different times, the Flexible F-Link communication must be acknowledged once.

When communication is running, an interruption of the connection is detected at the latest after expiry of the configured timeout. The Flexible F-Link communication is then passivated and the set replacement values are output.

After restoring the connection or correcting the error (ACK_REQ = TRUE), the Flexible F-Link communication must be reintegrated manually. In this example, the reintegration is carried out using the acknowledgement button connected to input %I0.0.

Figure 2-19: Reintegration of communication

After restoring the connection after a connection interruption, the status bit "ACK_REQ" may be toggled for a short time. The reason for this is the system properties of the standard communication and the intermediate buffering of not yet sent data in the TCP stack. See further information in the following section "Basics Open User Communication / TCP". When the communication buffer has been completely emptied again, the status bit "ACK_REQ" is permanently present and the communication can be reintegrated again.

3 Useful information

3.1 Basics Open User Communication / TCP

The following description refers to the TCP protocol that detects and automatically resolves data loss.

Certain properties of the TCP protocol must be considered when used as a transport protocol for Flexible F-Link communication. When "TSEND" is called, the data to be sent are transferred to the CPU-internal TCP stack (Figure 3-1).

The TCP stack tries to send the data to the receiver. From the sender's point of view, the data is sent when the recipient acknowledges receipt. If a data packet is not acknowledged by the receiving TCP stack due to a fault, the sender's TCP stack independently attempts to resend the packet within a defined time.

As long as the transmission has not been acknowledged by the other side, it remains in the transmit buffer of the TCP stack. Meanwhile further data packets are added by further calls of "TSEND", so that the data accumulates in the TCP stack. This takes place until an internal monitoring time of the acknowledgement of the opposite side has elapsed.

After this monitoring time has expired, the TCP connection is terminated. The TCP stack is emptied and data that has not yet been sent is discarded.

When using a configured connection, the controller now attempts to re-establish the connection at regular intervals. Once the connection is restored, data can be sent and received again.

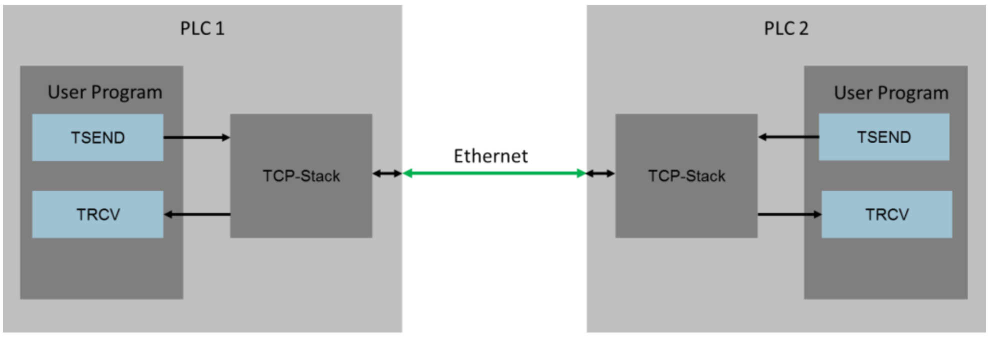
This has the following consequences for Flexible F-Link communication:

If no acknowledge data is received within the set F monitoring time of the Flexible F-Link connection (Figure 2-8), the Flexible F-Link connection is passivated and the parameterized error replacement values are output.

If there is a temporary interruption (i.e. the TCP connection still exists), the data accumulation that has occurred in the meantime from the TCP stack must first be processed. One way to implement this is to reduce the transmission/reception ratio as in this example. The block "TRCV" of the receiver is called faster than the block "TSEND" of the transmitter. A reduction of TRCV is recommended: TSEND at a ratio of 5 : 1.

Until the TCP stack is completely emptied, the bit "ACK_REQ" of the F communication DB may be toggled in the meantime. As soon as the "ACK_REQ" bit is permanently present, the F-Link communication can be reintegrated.

Figure 3-1



3.2 Possible impairments of Open User Communication (OUC) and effects on the Flexible F-Link

In safety-related communication Flexible F-Link, the data is transmitted via a standard protocol. The associated benefits, such as routing data across IP subnet boundaries, are offset by limitations.

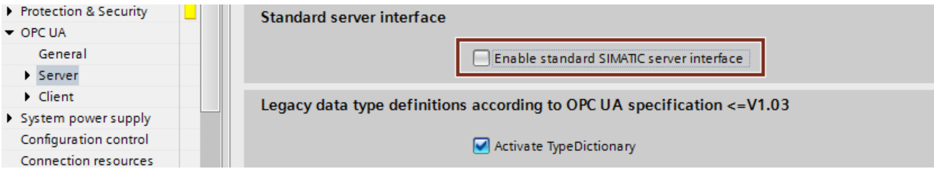
Compared to PROFINET, Open User Communication (e.g. TCP, S7 communication, etc.) is not deterministic. Data transfer may be affected by external factors, resulting in delays in transmission. If this causes the parameterized F monitoring time to be exceeded, the Flexible F-Link communication is passivated and the configured error replacement values are output. The following are possible interferences that can lead to a delay and thus to a timeout:

- Influences on the transmission path:
  Open User Communication shares the available network bandwidth with other communications. Too high a load on the transmission channel can therefore also affect the transmission speed.
- The Open User Communication is processed in the controller with priority 15. A high CPU utilization due to the use of OBs with the same or higher priority can lead to delays in the transmission of data.
- High communication load in the CPU:
  In the S7 CPU all Open User Communication runs with the same priority. A (temporarily) high communication load can lead to delays in transmission and consequently to a timeout at the Flexible F-Link. Open User Communication includes TCP/IP, UDP, S7 communication, OPC UA communication and the CPU-internal web server. In particular, the use of encrypted communication (e.g. HTTPS, Secure Open User Communication) can lead to a high load.

Typically, all these interferences result in a delayed transmission of telegrams, which has a negative effect on availability and possible response times. The following measures are recommended to reduce timeout times:

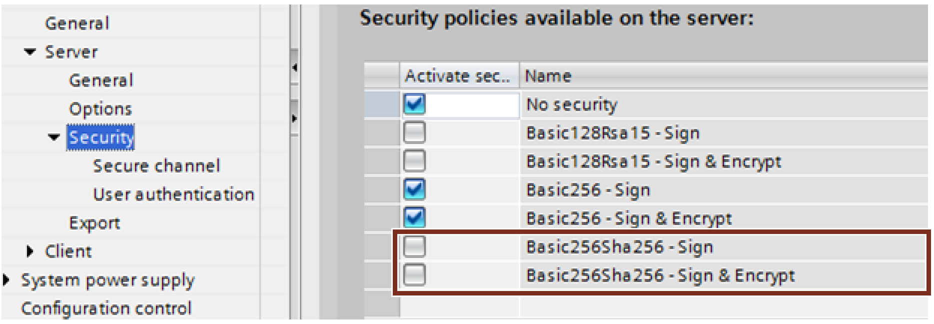
- Reduce the communication burden caused by other open user communication.
- Avoid web server access via HTTPS.
- Avoid Secure Open User communication.
- When using OPC UA communication:
  - Deactivate the standard SIMATIC OPC UA server interface and use only server interfaces instead:

Figure 3-2 Deactivation of the standard SIMATIC OPC UA server interface



  - Reduce the number of sessions and access to large arrays and structures.
  - Avoid connections with high encryption (e.g. Basic256Sha256 has a high impact on performance during the connection and when renewing the secure connection).

Figure 3-3



3.3 Alternative solutions / general information

1. **Handshake:**
   Instead of the reduction of "TSEND" / "TRCV", a handshake procedure can also be implemented: It must be ensured that only new data is sent as soon as the acknowledgement of the last data sent has been received. This ensures that there is no accumulation of data in the transmit buffer. The implementation of a handshake may require an extended F monitoring time for F-Link communication, as data may not be sent at each cycle.

2. **TCP connections:**
   Instead of the configured TCP connections, programmed connections can also be used. For this use the blocks "TCON" / "TDISCON". Note that if the TCP connection is aborted, the connection must be re-established manually using these blocks.

3. **Protocol:**
   For this example, TCP was selected as the underlying standard communication. In principle, the Flexible F-Link can also be used with other protocols (e.g. UDP, S7 communication, etc.) as long as consistent data transmission is guaranteed.

3.4 Flexible F-Link communication comparison

The following table shows a comparison of Flexible F-Link communication and safety-related IO controller-I-device or IO controller-IO controller communication. The table is intended to assist you in selecting the appropriate communication method:

Table 3-1: Comparison Flexible F-Link

Flexible F-Link via TCP	IO controller-I-device communication with SENDDP/RCVDP:	IO controller-IO controller communication with PN/PN coupler and SENDDP/RCVDP
Transmission of up to 100 bytes	Transmission of 6 bytes: (16 Bool + 2 Int or 1 DInt)	Transmission of 6 bytes: (16 Bool + 2 Int or 1 DInt)
Use of all F data types possible (Bool, Int, Word, DInt, Time)	Possible data types: Bool, Int and DInt	Possible data types: Bool, Int and DInt
Structuring of data possible	No structuring possible	No structuring possible
Cross-network communication possible (IP routing)	Communication only possible in the same subnet	Cross-network communication possible
No additional HW required	No additional HW required	PN/PN coupler required
Simple project planning	More complex project planning, especially when splitting into several TIA Portal projects	More complex project planning, especially when splitting into several TIA Portal projects
Not deterministic	Deterministic	Deterministic
Possible impairment of standard communication due to external interference (high network load, high CPU utilization due to higher-priority OBs, etc.)	No influence	No influence
Higher timeout times → longer response times	Short timeout times → short reaction times possible	Short timeout times → short reaction times possible
Fixed assignment of connection partners (fixed UUID)	Dynamically changing connection partners possible (variable DP_DP_ID)	Dynamically changing connection partners possible (variable DP_DP_ID)

4 Appendix

4.1 Service and support

Industry Online Support

Do you have any questions or need assistance?

Siemens Industry Online Support offers round the clock access to our entire service and support know-how and portfolio.

The Industry Online Support is the central address for information about our products, solutions and services.

Product information, manuals, downloads, FAQs, application examples and videos – all information is accessible with just a few mouse clicks:
https://support.industry.siemens.com

Technical Support

The Technical Support of Siemens Industry provides you fast and competent support regarding all technical queries with numerous tailor-made offers – ranging from basic support to individual support contracts. Please send queries to Technical Support via Web form:
www.siemens.com/industry/supportrequest

SITRAIN – Training for Industry

We support you with our globally available training courses for industry with practical experience, innovative learning methods and a concept that's tailored to the customer's specific needs.

For more information on our offered trainings and courses, as well as their locations and dates, refer to our web page:
www.siemens.com/sitrain

Service offer

Our range of services includes the following:

- Plant data services
- Spare parts services
- Repair services
- On-site and maintenance services
- Retrofitting and modernization services
- Service programs and contracts

You can find detailed information on our range of services in the service catalog web page:
https://support.industry.siemens.com/cs/sc

Industry Online Support app

You will receive optimum support wherever you are with the "Siemens Industry Online Support" app. The app is available for Apple iOS, Android and Windows Phone:
https://support.industry.siemens.com/cs/ww/en/sc/2067

4.2 Links and literature

Table 4-1

No.	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link to this entry page of this application example https://support.industry.siemens.com/cs/ww/en/view/109768964
\3\	Programming and Operating Manual SIMATIC Safety - Project Planning and Programming https://support.industry.siemens.com/cs/ww/en/view/54110126
\4\	SIMATIC STEP 7 Reaction time table https://support.industry.siemens.com/cs/ww/en/view/93839056

4.3 Change documentation

Table 4-2

Version	Date	Modifications
V1.0	08/2019	First version