TSX ETY 110 WS Premium PLCs

TCPIP Free Messaging System DFBs

Operating Modes Manual

1.1   General

The TCP communication DFB library used with version 2.8 or later of the TSX ETY 110 WS module for Premium PLCs is used to transfer blocks of data between a PLC application and a remote application via a TCPIP connection established at the initiative of the remote application.

The remote application is executed on a device that supports the TCPIP communication profile.

The communication DFB library is composed:

• of a connection management TCP_CNX DFB,

• of a TCP_SEND DFB for transmission of data blocks of a maximum size of 8 kilobytes,

• of a TCP_RECEIVE DFB for transmission of data blocks of a maximum size of 8 kilobytes.

Note:
TCP communication DFBs for Premium PLCs use the services from the "OPEN TCP for TSX Premium" range which is itself constituted of an TCP EF (Elementary Function) library: an EF enabling the PL7 application to manage TCP connections and to send and receive byte flows on its connections.
For further information please refer to the document "OPEN TCP for TSX Premium" (ref. 3500 1359 P04 000 Issue No.:07).

Installation of the product «DFBs of TCPIP free messaging system» completes a PL7 Pro programming workshop of V3.3 or later by installing:

- 3 DFBs: TCP_CNX, TCP_SEND, TCP_RECEIVE,

- and a TCPIP_DFB EF family.

2.1   Installation procedure

The CD-ROM TLXCDUNTCPB33F gives access to the installation procedure and documentation.

Insert the CD-ROM, confirm at the welcome screen and follow the menus.

2.2   Configuration of module TSX ETY 110 WS

The module TSX ETY 110 WS is configured by PL7 in accordance with the data entered by the user in the module configuration screen:

• Local IP address of the PLC on the network.

• The SubNet Mask.

• The IP address of the gateway by default.

• The use of Ethernet 802.3 frames with SNAP (SubNetwork Access Protocol).

The module TSX ETY 110 WS shall be installed in rack 0.

2.3   Programming DFBs in an application

2.3-1   Importing DFBs

In "Structure View" of PL7 Pro, under the "DFB types" heading, select "Import Binary".

Import each of the 3 DFBs TCP_CNX, TCP_RECEIVE, TCP_SEND into the application.

2.3-2   Create a DFB Instance

From within PL7 Pro select "Tools", "Library" then the "DFB" thumbnail tab.

Then choose the type of DFB that you wish to instantiate and confirm with "Create".

Next type the instance name of your DFB and confirm with "Create".

Your DFB is now instantiated. The instance name is the one which is used in the PL7 PLC program.

CAUTION

It is only possible to import the DFBs if the installation procedure of the CD-ROM TLXCDUNTCPB33F has been fully executed:

• installation of the TCP_CNX, TCP_RECEIVE, TCP_SEND DFBs,

• installation of the TCPIP_DFB EF family in the library.

3.1   TCP connection principles

The connection is established between the two applications asymmetrically, and is based on a client/server model.

In order for a connection to be established between the two applications, the server application must be awaiting an incoming connection on a TCP service port.

The TCP protocol uses the port numbers to identify destinations on a given machine. When the port number is identified by the server, the client can then send a connection request to the server application.

The server application accepts the incoming connection. Once the connection is open, a send buffer and a receive buffer are attributed to this TCP service port.

Data can be transferred on an established connection.

The data transfer is reliable, (data stored in buffer zone), non-structured (byte and not message flows) and simultaneously bidirectional (full-duplex).
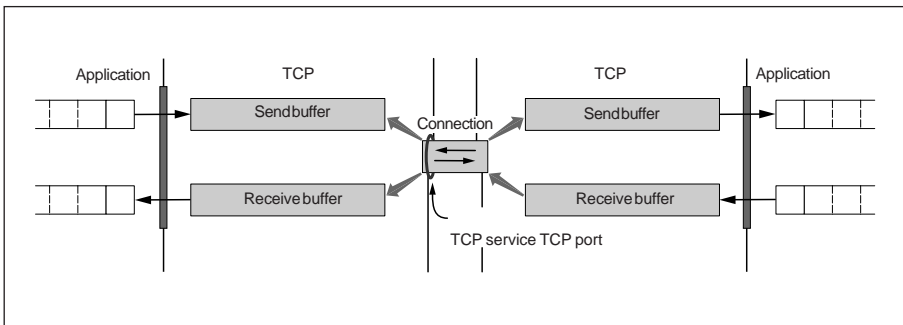
3.2   Basic principle of DFBs

3.2-1 Establishing connections with access security

The server PLC awaits incoming connections for each local TCP service port.

This connection queue is managed by the TCP_CNX DFB. All the remote machines declared in the configuration of the TCP_CNX DFB can then connect to a PLC. Immediately after an incoming connection is accepted on a local service port, the TCP_CNX DFB checks that the IP address of the remote machine is in the list of remote machines with authorization to connect.

If the remote IP address is given in this list, the connection to the local TCP service port is accepted. This connection is then stored in the list of open connections. The associated local TCP service port is busy and is not be accessible for other connection requests.
If the remote IP address is not included in the list of authorized addresses , the TCP_CNX DFB closes the connection and returns to awaiting connection to this local TCP service port.

Note

A connection will be closed down immediately by the TCP_CNX DFB for any one of the following reasons:

- Port used by another machine (one connection per port).
- Unauthorized IP address.

Each TCP connection opened with the module is thus characterized individually by the local TCP service port (1 port ↔ 1 connection).

The IP address of the remote machine is an attribute associated solely with this connection

Connection status graph for a port
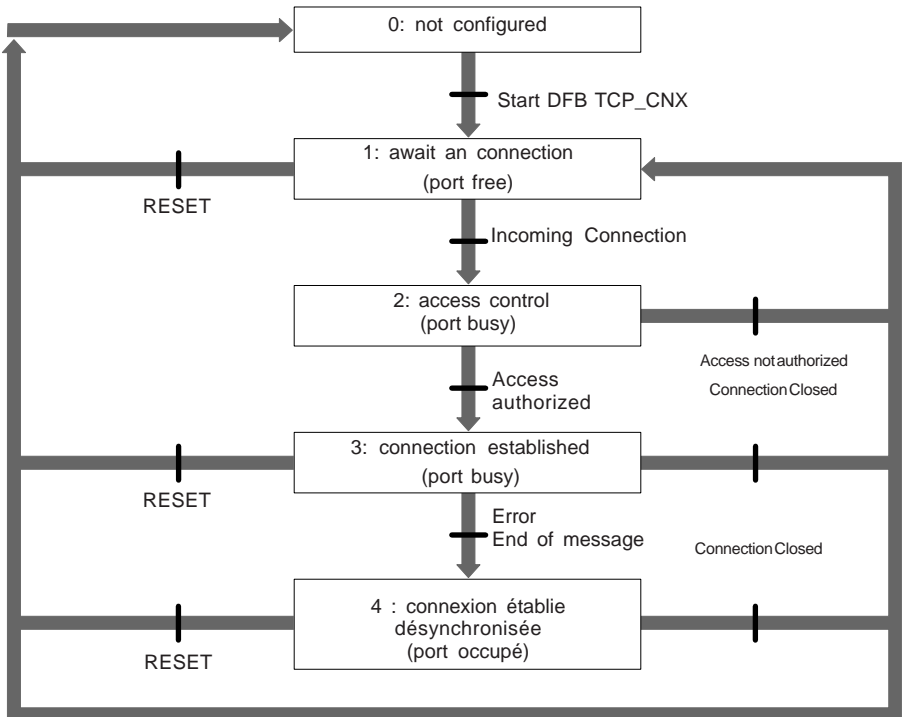
Note :
This status graph, managed by the TCP_CNX DFB is for a TCP service port. The status graph is the same for all the other TCP service ports.

The TCP_CNX must be started for every PLC cycle in order to manage the connections.

Note

An established connection will be closed:
• as a result of a request made by the PL7 application to cancel a transfer in progress.
• upon closure of the connection by the remote application.
• in the event of access by an unauthorized machine (temporary opening for inspection).

Detection of an "end of message" type error upon reception of a message by the module does not result in the connection being automatically closed by the module.

However, the connection will be marked as desynchronized, with the following characteristics:

- All data received on a desynchronized connection is lost.

- New receive requests on a desynchronized connection are immediately sent back (without any data) with a desynchronized connection error.

- It is possible to send data on a desynchronized connection.

All connections will be closed:

- as a result of a software RESET (CPU Reset, downloading of PL7 application).

- as a result of a module equipment RESET (power outage, change of address on terminal block, etc.).

- as a result of a TCP_CNX DFB RESET.

Note

> Interrupting connections suddenly without prior warning to the PLC (sudden stoppage of the remote application, power outage of the remote machine, connection closed by the remote machine with cable unplugged, etc.) can give rise to freezing of communication on the port attached to this connection. In practice, a receive request can remain frozen for a maximum of two hours (KEEP ALIVE time-out time). In addition, any new connection to this port during this time period will be refused by the PLC.
>
> In order to remedy this, it is recommended that the PLC application send out a maintain connection message (the message of length 0 will be accepted by the TCP_SEND DFB).

3.2-2 Transferring messages

Description of the transfer service

The transfer service is used to exchange **messages** (series of N bytes) on a connection established between a remote client and the PL7 application of the PLC on a given local port.

The transfer requests offer the PL7 application two possible ways of identifying the remote client:

1. By using a single parameter: local port number of the PLC (parameter sufficient for identification of a TCP connection of the module).
   The transfer will be made if a connection is established on this port, whatever remote machine is connected to this port.

2. By using the pair of parameters: local port number and IP address of the remote machine.
   The transfer will be made if a connection is established on this port with the remote machine specified by the PL7 application.

Transferring a message from application to application
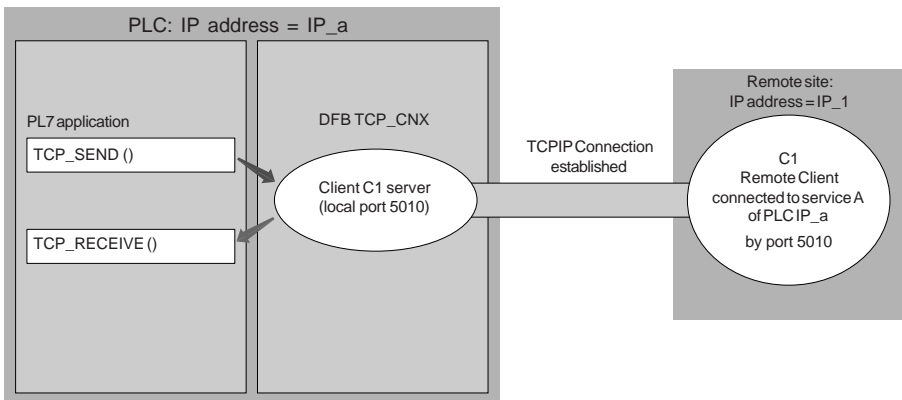
The following message transfer services are available to the PL7 application of the PLC:

- Send request for transmission of a message with a maximum size of 8 kilobytes to a remote client.
  A message transmission will remain active as long as the two following conditions are met:
  - the send buffer of the transmitter connection point (PLC) is full,
  - the receive buffer of the destination connection point is full,
  in other words if the destination is not using the data on the connection.

- Request from a remote client for reception of a message with a maximum size of 8 kilobytes. Await reception is active as long as there is a connection and no complete message has been received on this connection. If necessary, the PL7 application can opt to use a timer (connected to the RST signal of the DFB) to cancel the current request at the end of a given period.

If the TCP connection is not established with the remote client at the time of the data transfer request (send or receive), the request will be refused immediately with the error indication: **connection not established**.

Similarly, if the connection is closed by the remote client or lost, all operations in progress on this connection will be canceled.

Multiple transfer on the same port

The PL7 application cannot activate several requests of the same type (send or receive) on the same port at the same time, without awaiting the report for the previous request.

However, there is no temporal correlation between transmissions and receptions. For one and the same connection, the transmission channel and the reception channel are independent and thus the PL7 application can simultaneously request both a transmission and a reception on the same port.

Format of messages exchanged between the module and the remote application

An important characteristic of communication on a TCP connection is the continuity of the information («byte flows»); the receiving machine cannot identify the structure of a message without the information and the particular protocol established between the sender and the destination.

It has been decided to structure the messages exchanged using the following format:

2 bytes	N octets	
Message length = N (network format)	Données du message utilisateur (N-1) octets	Caractère Fin de Message

At the PLC, the two additional fields (message length and end of message character) will be added (transmission) or deleted (reception) by the DFBs (TCP_SEND or TCP_RECEIVE).

Message length:

• This field specifies the total number of useful bytes in the message, constituted by the user data plus the optional end of message character.

• It is defined as a 16 bit signed value in network format. It is used to encode the values from 0 to 32 767 (32 K - 1).

"End of message" character:

• This 1 byte field is an optional character used to mark the end of the message.

• This end mark becomes redundant when used with the "message length" parameter which is sufficient to structure the flow of data into messages. It is only useful for controlling the coherence of a message when it is being received by the module and for avoiding the spread of an error generated by a remote application.

Procedure for transmission on the TCP connection

The procedure for transmission on an established TCP connection is triggered by the TCP_SEND DFB of the PL7 application.

The TCP_SEND DFB transmission procedure is carried out by block of 240 bytes sent to the TSX ETY 110 WS module which immediately transmits the block to the remote machine. The "length" and "end of message character" fields (if configured) are automatically transmitted by the TCP_SEND DFB.
The DFB signals the end of the exchange using its activity bit.

Procedure for reception by the module on the TCP connection

The data received on an established TCP connection is stored in the buffers of the TSX ETY 110 WS module. The data received on the connection will only be read if a TCP_RECEIVE DFB is active.

The size of the receive buffer is a maximum of 4096 bytes. This means that when a message of 8 K is sent by a remote machine, half of the data is stored in its send buffers until the PL7 applications has received the first half of the data.

The TCP_RECEIVE reception procedure is as follows:

• Await/Receive* 2 bytes indicating the length N of the message to be received.

• Receive* N bytes of the message and of the end of message character if the end of message character option is selected in the configuration. Check it and delete it.

• The DFB signals the end of the exchange using its activity bit.

Note

> If an "end of message character" error is detected, the receive buffer of the PL7 application contains the received message which can be analyzed.
>
> If there are characters missing relative to the length specified in the message header (sender application error), reception is put on infinite standby whilst awaiting the missing bytes. It will be unlocked upon reception of the following message, but with an "end of message character" error detection (if the end of message character error does not coincide with any useful message data).
>
> Reception is then desynchronized.
>
> An end of message error is not necessarily detected immediately with the first subsequent message (when the end of message character coincides with data in the following messaging, several messages may be received prior to detection).

* The Await/Receive operation on the connection also enables the indication of closure of the connection by the remote machine to be received.

Application to application protocol

No restriction is imposed on the exchange protocol used on a TCP connection established between the remote client and the PLC application. The exchanges can be defined as:

• Request/response at the initiative of the client application.
• Request/response at the initiative of the PLC application.
• Unsolicited messages at the initiative of the client application.
• Unsolicited messages at the initiative of the PLC application.

Characteristics of use of transfer by PL7 application

Number of TCP ports

The module is dimensioned for 16 TCP ports and thus for 16 connections.

User buffer associated with DFB

During CPU/module exchanges, the send or receive DFB directly uses the buffer specified by the user (no user data is copied into a system buffer). Thus, as long as the DFB is active (ACTIVITY bit at 1), the PL7 program shall under no circumstances modify the contents of the buffer associated with this DFB.

4.1   Points common to all DFBs

The function blocks use the messaging services implemented in the PREMIUM processor. Because of this,**all the communication function blocks can be executed over several PLC cycles**.

Because of the module/CPU exchange architecture, sent messages (8 Kbytes) or received messages (8 Kbytes) will be exchanged between the module and the PL7 application by 256 byte datagrams containing information bytes for checking the segmentation and 240 bytes of useful data.

The CPU/module exchange of a datagram is made per PLC cycle.

4.1-1   Maintained call

The execution of function blocks is explicitly maintained. The user must program maintenance of the DFB as long as its ACTIVITY bit is active.

It is strictly prohibited to execute the same DFB instance more than once in the same PLC cycle .

Maintenance of the DFBs is obligatory since all the communication function blocks are being executed over several PLC cycles.
As soon as the DFB is called, it is restarted for each turn of the PLC cycle as long as its activity bit is equal to 1.

4.1-2 Management table

The PL7 application must supply a 67 word management table to the DFBs in order to enable each DFB to be aware of the status of the connections in progress.
Each table entry is constituted by 16 bit words and includes the IP address, the port number and the status of connection with a remote machine.

This table is defined in the PLC in the MW memory zone and according to the following structure:

%MWi	MODULE	Module number
%MWi + 1	NP	Number of listen ports
%MWi + 2	EOM	End of message character
%MWi + 3	IP 1	IP address of remote machine N° 1
%MWi + 5	P 1	Number of local port N° 1
%MWi + 6	Status 1	Status of connection N° 1
%MWi + 7	IP 2	
%MWi + 62	Status 15	
%MWi + 63	IP 16	IP address of remote machine N° 16
%MWi + 65	P 16	Number of local port N° 16
%MWi + 66	Status 16	Status of connection N° 16

The structure of this table is given for information purposes. This table is entirely managed by the DFBs and must not be modified without activating the RESET input of the connection management DFB.

• **Internal data**

 MODULE – Module number
 Number of the physical slot in which module TSX ETY 110 WS is located in the rack.

 NP - Number of listen ports
 This parameter, at between 1 and 16 inclusive, defines the NPs, the first elements of the table of the local port numbers to be used.

EOM – End of Message Character
This parameter defines the presence and value of an "End of Message" character to be used in message exchanges. It is updated by the TCP_CNX DFB when it first calls or after a RESET with the value parametered by the PLC application:

- EOM = 0: no end of message character.

- EOM from 0x01 to 0xFF : value of end of message character:
  - to be added during transmission,
  - to be checked and deleted during reception.

IP i – IP address of the client connected to local port No. i
This parameter is updated by the TCP_CNX DFB when a client is connected (value 0 if there is no connection on this port). It is then used by the TCP_SEND and TCP_RECEIVE DFBs to check the "IP address" input parameter when this is not set to zero.

P i – Number of local port No. i
Table of 16 local service ports to be listened to (signed value $\geq$ 5010).
For a given configuration, only the first number NPs will be used.

Status i – connection status No i
This parameter is used to provide information on the associated connection:

- Bits 0..3: Service socket number of between 0 and 15.
- Bit 8: 1 $\rightarrow$ Established connection 0 $\rightarrow$ no connection.
- Bit 9: 1 $\rightarrow$ Sending.
- Bit 10: 1 $\rightarrow$ Receiving.
- Bit 11: 1 $\rightarrow$ Error on last transmission.
- Bit 12: 1 $\rightarrow$ Error on last reception.

4.1-3   RST and ACTIVITY bits

Inputs		Outputs
RST bit		ERROR bit
	MANAGNT array of word	
Public Internal Data	ACTIVITY bit STATUS word	
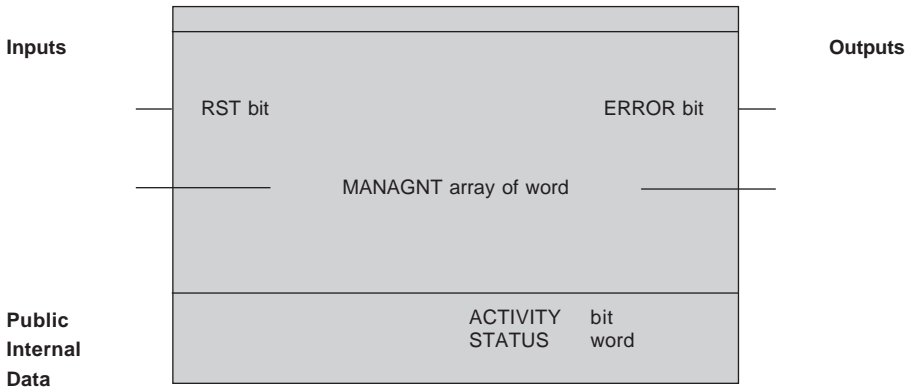
All the TCP function blocks use the messaging system and are executed over several PLC cycles. This is why the user must be able to interrupt the execution of a block (when awaiting a response to a message); he must also be in a position to know whether or not any blocks are currently being executed, in order to know whether or not to maintain its activity.

ACTIVITY bit

TCP function blocks have a BIT type variable which indicates their activity; in other words whether they have to be maintained or not.

RST action

TCP function blocks have a BIT type RST input which can be used to interrupt their execution when this bit is set to 1, which gives rise to the following actions (if the connection is open):

• Any messages in progress (send and receive) are lost for the connection.

• The connection is closed.

4.1-4   ERROR bit - STATUS WORD

This output bit is set to 1 if the DFB is not able to function correctly. The STATUS word then indicates the type of error.

• **STATUS word**

Bit	DFB	Meaning	Corrective action
Bit 0 = 1	C R S	The module is not an ETY 110 module	Check the value used for the DEVICE input parameter of the configuration. Check the hardware configuration of the PLC.
Bit 1 = 1	C R S	Syntax error	DFB TCP_CNX RESET. Check the parameters and restart.
Bit 2 = 1	R S	The port requested is used by a machine whose IP address is different from that specified in the input parameter (DEST for transmission, FRM for reception).	Check the configuration of the remote clients. Enter the DEST or FRM parameter to select the remote client correctly.
Bit 3 = 1	R S	Local port incorrect. The port number is not in the list of local ports given during module configuration.	Check the list of listen ports of the TCP_CNX DFB.
Bit 4 = 1	R S	Erroneous length	During transmission check that the SIZE parameter is less than 8192. During reception a message header with a field length of > 8192 has been received.
Bit 5 = 1	R S	The connection is not open	Check that there is a client.
Bit 6 = 1	R S	Loss of current connection functioned	Check that the client transfer has correctly.
Bit 7 to 9		Not used	

Bit	DFB	Meaning	Corrective action
Bit 10 = 1	R	Connection desynchronized during reception. A message with an "end of message" character error has been received.	Analyze the message in the receive buffer to identify the source of the error.
Bit 11= 1	R	End of Message Character error. An "end of message" character error has been detected on this message	Analyze the message in the receive buffer to identify the source of the error. RESET the DFB.
Bit 12 = 1	R	Message truncated. The receive buffer is too small to receive the whole of the message sent by the remote client	Provide a larger buffer. Check the maximum size of the messages exchanged between applications on this connection.
Bit 13 = 1	R	Not used	
Bit 14 = 1	R S	Processing interrupted. The function block has been interrupted during execution by an action RST ou une reprise à restart.	Restart the connection from the client. The server automatically goes into listen mode.
Bit 15 = 1		Not used	

5 TCP_CNX DFB connection management

5.1   Presentation

The TCP_CNX DFB is responsible for managing the connections with remote clients as well as the operating modes of these connections. It is informed of breaks in connections via the intermediary of the status of the management words by a bit set by a TCP_SEND or TCP_RECEIVE DFB instance.

Only one TCP_CNX DFB instance shall exist for a given TSX ETY 110 WS module.

5.1-1 Characteristics

Inputs

Ouputs

DFB   TCP_CNX

RST bit

ERROR bit

MANAGNT array of word

Public Internal Data

DEVICE	word	ACTIVITY	bit
PORT	array of word	STATUS	word
IPACCEPT	array of dword	IPCNX0	string
EOM	word	…	
		IPCNX15	string
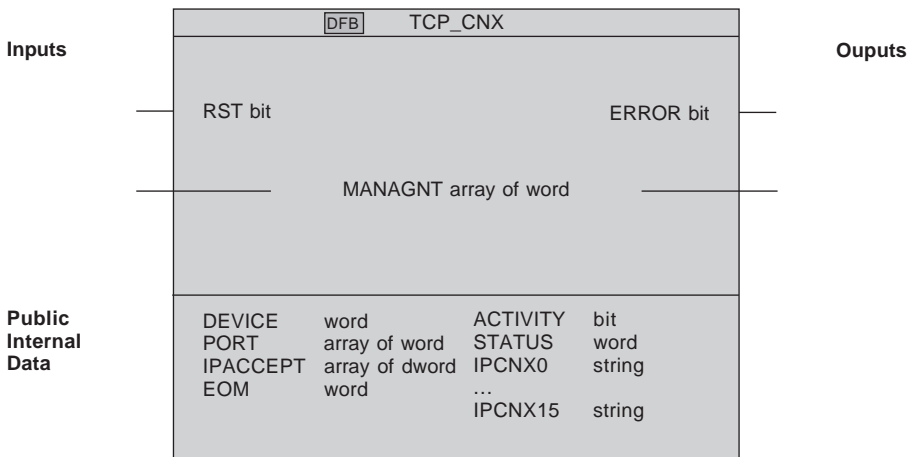
Input parameters

Parameters	TypeDescription	
RST	Bit	Setting this input to 1: - interrupts the exchanges in progress, - causes all connections to close.

Input/Output parameters

Parameters	Type	Description
MANAGNT	Word table	Management table common to all DFBs.

Output parameters

Parameters	Type	Description
ERROR	Bit	BitThis output bit is set to 1 if the DFB is not able to function correctly. The STATUS word indicates the type of error which occurred.

Internal public data

Parameters	Type	Write/Read Variables	Description
DEVICE	Word	E	Module number Number of the physical slot in which module TSX ETY 110 WS is located in he rack.
PORT	Word table	E	Local port number No. i. Table of 16 local service ports to be listened to (signed value $\geq$ 5010). The value 0 indicates a non-significant value.
IPACCEPT	Double	E	IP address of the remote machine No. i word 4 byte IP address table from table 8 remote machines authorized to connect to the PLC. The value 0 indicates a non-significant value.
EOM	Byte	E	"End of Message" character This parameter defines the presence and the value of an "End of Message" character to be used in the message exchanges: - EOM = 0: no end of message character, - EOM from 0x01 to 0xFF: value of end of message character: - added by the module during transmission - checked and deleted by the module during reception.
IPCNXi	Character string	L	IP address of the remote machine No. i Read-only character string to be used for maintenance only and indicating the IP address of the remote machine in the format aaa.bbb.ccc.ddd connected to the port with index i in the configuration. The value 0.0.0.0 indicates that no machine is connected to this port.
ACTIVITY	Bit	L	This output bit is at one when the DFB is in progress and needs to be maintained. It is set to 0 upon a warm or cold restart and upon a DFB RESET.
STATUS	Mot	L	This word is only meaningful if the ERROR output bit (erroneous exchange) is set to 1. It indicates the code of the error which occurred during the exchange (Each word bit set to 1 indicates an error). Please refer to the chapter "Points in common to all DFBs".

5.1-2 Operation

The TCP_CNX DFB must be called at each turn of the PLC cycle to ensure permanent management of the TCP ports.

Listening to the configured ports commences if the RST and ACTIVITY bits are set to 0. Then the ACTIVITY bit remains at 1 until the application allows the RST bit to return to 0.

If the dialog with the module is not correct, the output error bit ERROR (erroneous exchange) is set to 1.

At any time, setting the RST bit (priority input) to 1 enables all the connections in progress or on standby to be interrupted. The ACTIVITY bit (exchange finished) is set to 0 and the ERROR bit (erroneous exchange) is set to 1. The status word <TCP_CNX instance name>.STATUS indicates the type of error.

At a cold or warm restart the TCP_CNX DFB automatically returns to listen mode for the configured ports.

5.1-3 Programming example - Connection to 1 single TCP port

- **Data used**

```
%MW0 :                  Management table
%KD200 : H'5A000001'    IP address (90.0.0.1) of the remote machine authorized to
                        connect
%KW300 : 5010           Service port
%KW0 : 2                Physical slot in which module ETY110 is located
%KW1 : 15               End of message character
%M0                     Error bit
%MW100 :                Working variable for indexing of tables
TCP_COX                 TCP_CNX DFB instance name
```

```
! < TRANSMISSION OF MESSAGE >
  IF NOT TCP_COX.ACTIVITY THEN

        (* Initializing configuration *)
        TCP_ COX.DEVICE := %KW0 ;

        (* prior setting to 0 of table *)
        FOR %MW100 :=0 TO 15 DO
            TCP_ COX.PORT[%MW100]:=0 ;
        END_FOR;
        (* List of listen ports *)
        TCP_ COX.PORT[0]:= %KW300;

        (* prior setting to 0 of table *)
        FOR %MW100 :=0 TO 7 DO
            TCP_ COX.IPACCEPT[%MW100]:=0;
        END_FOR;
        (* List of IP addresses authorized to connect *)
        TCP_ COX.IPACCEPT[0]:= %KD200;

        (* End of message character *)
        TCP_ COX.EOM:= %KW1;

        (* DFB start *)
        TCP_ COX (0, %MW0:67,%M0 );
  ELSE
        (* previous message REPORT PROCESSING *)
        IF %M0 THEN JUMP %L3;
          (* Maintenance of DFB *)
          TCP_ COX(0, %MW0:67 , %M0 );
          END_IF;
  END_IF;
```

Note:
Parameters passed on to TCP_CNX DFB
< DFB TCP_CNX instance name > (RST, MANAGNT, ERROR)

6.1 Presentation

The TCP_SEND DFB enables a data message to be sent to a remote "client" application via a TCPIP connection. The maximum size of the message to be sent is 8 K bytes.

6.1-1 Characteristics

				DFB	TCP_SEND			

Inputs   **Ouputs**

RST bit

BUF array of word   ERROR bit

MANAGNT array of word

Public
Internal
Data

DEST	dword	ACTIVITY	bit
PORT	word	STATUS	word
SIZE	word		
INHIB	bit		

Input parameters

Parameters	Type	Description
RST	Bit	Setting this bit to 1: - interrupts the exchange in progress (if the ACTIVITY bit was at 1) - causes the connection to close (if it is open) - sets the ACTIVITY information bit to 0 and the ERROR output bit to 1. The error code is then contained within the word STATUS.
BUF	word table	This variable defines the address of the first word %MWi of the message to be sent

Input/Output parameters

Parameters	Type	Description
MANAGNT	Word table	Management table common to all DFBs.

Output parameters

Parameters	Type	Description
ERROR	Bit	This output bit is set to 1 if the exchange is not carried out correctly. The STATUS word indicates the type of error that has occurred.

Internal public data

Parameters	Type	Write/Read Variable	Description
DEST	Double word	E	This variable defines the IP address of the remote machine on which the client is connected to the local port PORT. By default DEST = 0, the message is sent to the only remote client station connected to the local service port of the PLC.
PORT	Word	E	This variable defines the local port number to which the remote client is connected.
SIZE	Word	E	This variable defines the size in octets from the message to be sent. From 0 to 8192. Does not take account of the end character.
ACTIVITY	Bit	L	This bit is at one when an exchange is in progress. It is set to 0 when the exchange is finished.
INHIB	Bit	L	This bit enables the error warning to be inhibited: the ERROR output bit and the STATUS word remain at 0 (execution of the block is not interrupted).
STATUS	Word	L	This word is only significant if the ERROR output bit (erroneous exchange) is set at 1. It indicates the code of the error that occurred during the exchange (each word bit set to 1 indicates an error). Please refer to the chapter "Points common to all DFBs".

6.1-2 Operation

The transfer of data is triggered when the TCP_SEND DFB is called if the RST bit is at 0 and if the internal ACTIVITY bit is at 0 (no exchange in progress). During the exchange, the ACTIVITY bit is at 1. At the end of transmission, the ACTIVITY bit is set to 0. In addition, if the exchange is not correct, the ERROR output bit (erroneous exchange) is set to 1.

At any time, setting the RST bit (priority input) to 1 allows the exchange in progress to be interrupted. The ACTIVITY bit (exchange finished) is set to 0 and the ERROR bit (erroneous exchange) is set to 1. The status word <TCP_SEND instance name>.STATUS indicates the type of exchange error. If the connection is open, it closes and the module returns to awaiting the incoming connection. Reestablishing the connection to this TCP service port, reinitiates the transmission of messages. This reconnection is managed by the TCP_CNX DFB.

On triggering of transmission of a message, processing is as follows:

- Checking of the input parameters: IP address of the remote machine, number of local port.

- Search for the open connection with the remote client requested.

- Transmission of the message to be sent per block of 240 bytes (adding the "end of message" character, if the option is requested by default) on the TCP connection.

6.1-3 Programming example

• **Data used**

```
%MW0:               Management table
%KD200 : H'5A000001'  IP address (90.0.0.1) of the remote machine (in 4 bytes)
%KW300 : 5010       Service port

%W200 : 400         Size in words of the message to be sent
%W300 to W700       Message to be sent
%M0                 Error bit
TCP_EMIS            TCP_SEND DFB instance name

! < TRANSMISSION OF MESSAGE >
  IF NOT TCP_EMIS.ACTIVITY THEN
        (* previous message REPORT PROCESSING *)
        IF %M0 THEN JUMP %L3;

           (* Transmission of next message *)
           TCP_ EMIS.PORT:=%KW300;
           TCP_ EMIS.DEST:=%KD200;
           TCP_ EMIS.SIZE:=%MW200*2;
           (* start of DFB first parameter which represents RST at 0 *)
           TCP_ EMIS(0, %MW300:400, %MW0:67, %M0);
           END_IF;
  ELSE
        (* maintenance of DFB first parameter which represents RST at 0 *)
        TCP_ EMIS(0, %MW300:400, %MW0:67, %M0);
  END_IF;
```
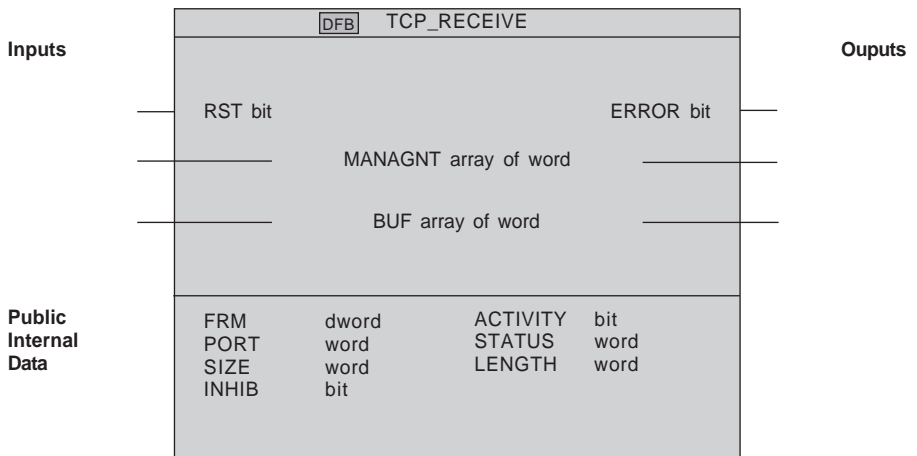
Note:
Parameters passed on to DFB
<TCP_SEND> instance name (RST, MANAGNT, BUF, ERROR)

7.1 Presentation

The TCP_DFB enables a data message to be received from a remote "client" application via a TCPIP connection. The maximum size of the message to be sent can be up to 8 K bytes.

7.1-1 Characteristics

Inputs   **Ouputs**

	DFB TCP_RECEIVE	
RST bit		ERROR bit
	MANAGNT array of word	
	BUF array of word	

Public Internal Data

FRM	dword	ACTIVITY	bit
PORT	word	STATUS	word
SIZE	word	LENGTH	word
INHIB	bit		

Input parameters

Parameters	Type	Description
RST	Bit	Setting this bit to 1: - interrupts the exchange in progress (if the ACTIVITY bit was at 1) - causes the connection to close (if it is open) - sets the ACTIVITY variable to 0 and the ERROR output bit to 1. The error code is then contained within the STATUS word.

Input/output parameters

Parameters	Type	Description
BUF	Word table	This input and output parameter defines the address of the first %MWi word of the buffer in order to receive the message.
MANAGNT	Word table	Management table common to all DFBs.

Output parameters

Parameters	Type	Description
ERROR	Bit	This output bit is set to 1 if the exchange is not finished correctly. The STATUS word indicates the type of error which occurred**.**

internal public data

Parameters	Type	Write/Read Variable	Description
FRM	Double word	E	This variable defines the IP address of the remote machine on which the client is connected to the local port PORT. By default FRM = 0 and the DFBawaits the message sent by the uniqueremote client connected to the local service port of the PLC.
PORT	Word	E	The variable defines the number of the local port to which the remote client is connected.
SIZE	Word	E	Cette variable définit la taille en octets du tampon pour recevoir le message. From 0 to 8192.
ACTIVITY	Bit	L	This variable is set to 0 by the DFB when the exchange is finished. If the output ERROR bit (erroneous exchange) is at 0, the ACTIVITY variable indicates that the message has been correctly received. However, if the ERROR bit is status 1, the ACTIVITY variable indicates that the exchange is finished but erroneous
INHIB	Bit	E	This bit enables the error warning to be inhibited: the ERROR output bit and the STATUS word remain at 0 (execution of the block is not interrupted)
STATUS	Word	L	This word is only significant if the ERROR output bit (erroneous exchange) is set at 1. It indicates the code of the error that occurred during the the exchange (each word bit set to 1 indicates an error). Please refer to chapter "Points in common to all DFBs"
LENGTH	Word	L	This variable contains the number of bytes received if the ERROR output bit (wrong exchange) is in status 0.

7.1-2 Operation

The transfer of data is triggered when the TCP_RECEIVE DFB is called if the RST input is at 0 and if the ACTIVITY bit is at 0 (no exchange in progress). During the exchange, the ACTIVITY bit is at 1. At the end of reception, the ACTIVITY bit is set to 0. In addition, if the exchange is not correct, the ERROR output bit (erroneous exchange) is set to 1.

At any time, setting the RST input (priority input) to status 1 allows the exchange in progress to be interrupted. The ACTIVITY bit (exchange finished) is set to 0 and the ERROR bit (erroneous exchange) is set to 1. The status word <TCP_RECEIVE instance name>.STATUS indicates the type of exchange error. If the connection is open, it closes and the module returns to awaiting the incoming connection. If one or more messages are in the module's receive buffers, they are lost.

On triggering the reception of a message, processing is as follows:

• Check the input parameters: identification of the remote machine, number of local port,...

• Search for the open connection with the remote application requested.

• Reception EF queued to await message on this connection.

As soon as an DFB is active for a given connection, the module awaits the reception of the data on this connection and the first two bytes received indicate the length of the message to be received. The data received is then transferred into the PLC's words in frames of 240 bytes.

As soon as the full message is copied onto the PLC:

1. The DFB checks the end of character message (if this option has been configured).

2. The STATUS, LENGTH parameters of the DFB are filled in and the ACTIVITY variable and the ERROR bit are then set for the PL7 application.

7.1-3 Programming example

• Data used

```
%MW0 :                  Management table
%KD200 : H'5A000001'    IP address (90.0.0.1) of the remote machine (in 4 bytes)
%KW300 : 5010           Service port

%W200 : 400             Size of the receive buffer in words
%W300 à W700            Receive buffer
%M0                     Error bit
TCP_RECEP               DFB TCP_RECEIVE instance name

! < TRANSMISSION OF MESSAGE >
  IF NOT TCP_RECEP.ACTIVITY THEN
        (* previous message REPORT PROCESSING *)
        IF %M0 THEN JUMP %L3;

           (* Transmission of next message *)
           TCP_ RECEP.PORT:=%KW300;
           TCP_ RECEP.FRM:=%KD200;
           TCP_ RECEP.SIZE:=%MW200*2;
           (* DFB start*)
           TCP_ RECEP(0, %MW0:67,%MW300:400,%M0);
           END_IF;
  ELSE
        (* maintenance of DFB first parameter which represents RST at 0 *)
        TCP_ RECEP(0, %MW0:67 , %MW300:400,%M0);
  END_IF;
```

Note :
Parameters passed on to DFB
<TCP_RECEIVE> instance name (RST, MANAGNT, BUF, ERROR)

8.1 Points relevant to the performance of a TCPIP-OPEN communication

The performances of a TCPIP-OPEN communication depend on the following points:

• The PLC cycle time

  This takes account of the process handling time and the processing time reserved for the TCPIP-OPEN communication.

• The periodic cycle time

  This shall take into account the processing time of the TSX ETY 110 WS module in order for the response to the request made to the module to arrive during the following PLC cycle.

• The processing time of the TSX ETY 110 WS. module. This is less than 30 ms for 4 configured TCP ports.

• The maximum number of EFs started during the same PLC cycle. This is linked to the type of processor.
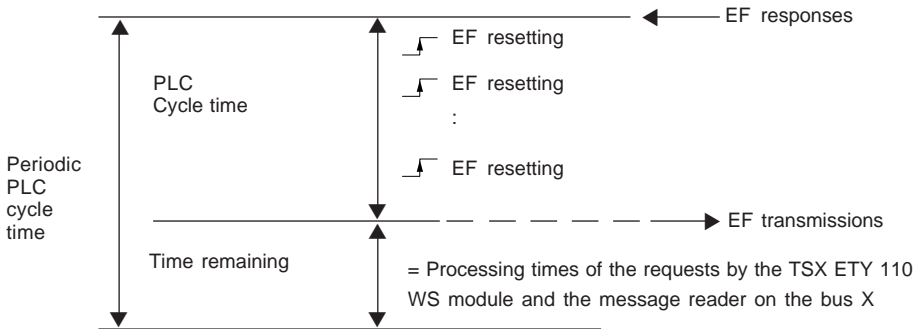
Note

> The best performances are obtained with a regular communication transmission and reception flow. This operation is optimal when a DFB can send an EF during each PLC cycle.
>
> As the EF is sent at the end of the PLC cycle, it is essential that its response arrives at the start of the following period.
> For this, the read time of the message on the X bus and the processing time of the ETY110WS module shall be less than the time remaining between the end of the PLC cycle and the start of the following period.

8.2 Diagram of EF processing during a PLC cycle

Note:
The TCPIP-OPEN EFs are used by the TCP_CNX, TCP_SEND, TCP_RECEIVE DFBs.
Please refer to the document "OPEN TCP for TSX Premium" relating to the structure and operation of the TCPIP-OPEN EFs.

8.3 Performance measurements

The performance measurements were made using a TCPIP-OPEN server PLC application on Premium TSXP57352 and a client TCPIP-OPEN application on a PC running Windows NT.

Periodic cycle time of a PLC	50 ms	70 ms	80 ms	90 ms
Response time for a 8 K octets message sent to 4 TCP service ports	3,7 s	3,33 s	3 s	3,75 s
Response time for a 8 K octets message sent to 2 TCP service ports	3 s			

9.1 Differences with OFB PL7-3

The offer is similar on the Series 7 range with ETH110. The differences are mainly:

- **Performance:** The ETH110 module only supports the TCPIP_OPEN communication profile. The performances with the TSX ETY 110 WS module will depend on the use of the other module functions (UNITE messaging system, MODBUS, HTTP, SNMP, FTP messaging system).

  The execution of several **simultaneous** DFBs (during the same PLC cycle) for making exchanges of the same nature with the same remote machine (identified by the port number and/or IP address) is not possible as it cannot be managed in the TCPOPEN architecture.

- **Operating modes: The connections are managed** by the module in the ETH110 offer whereas it **is managed by the PLC application** in the current approach with TCPOPEN.

- **Configuration:** On ETH110, the configuration consists of the module parameters, the list of ports used as well as the list of authorized remote IP addresses. In the TCPOPEN offer, the module parameters are configured by PL7 and managed by the system. The list of ports used and authorized IP addresses is managed by the TCP_CNX DFB (or EF).

- **PL7 programming:** The programming method is different: The **DFBs must be** maintained by the application, whereas the OFBs of the ETH110 are maintained automatically.
  The **Echo function does not** exist in the TCPOPEN architecture.

- **Network and remote external view:** The message entity is recognized by the ETH110 module whereas only the PLC recognizes it in the TCPOPEN offer. When sending a message this involves the **fragmentation into 240-byte units on the network. Transmission interruption** by the Premium of a message in progress may lead to the end of the message being lost.
  When receiving a message, the network view is identical to the behavior of the ETH110.

- **Diagnosis:** The diagnostic tools used for the OFBs in the 7 series (Applidiag) must be replaced by standard DFB operating tools built into the PL7.

9.2   Reminder on the TCPIP communication

Please refer to the following documents:

- Préparation au MCSE TCPIP, S & SM publications.

- Préparation au MCSE TCPIP, CAMPUSPRESS publications.

- TCPIP by Douglas Comer, InterEditions publications.