

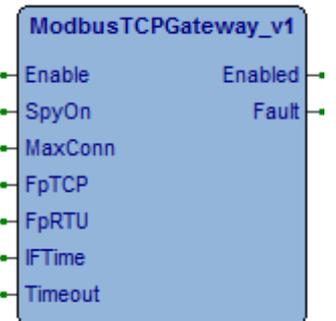
### 1.1.9 ModbusTCPGateway\_v1, modbus TCP gateway

Type	Library
FB	eLLabNetworkLib_A700

Questo blocco funzione opera come gateway Modbus tra una connessione modbus TCP ed una connessione seriale Modbus RTU. Il FB può gestire più streams Modbus TCP, per questo in **FpTCP** occorre passare l'array di streams di I/O da cui arrivano le richieste Modbus TCP. In **FpRTU** occorre passare lo stream verso cui sono inviate le richieste convertite in Modbus RTU.

In **MaxConn** occorre definire il numero di connessioni massime gestite sullo stream Modbus TCP.

In **IFTime** occorre definire il tempo di pausa su ricezione caratteri da porta seriale (Modbus RTU). In **Timeout** si definisce il tempo massimo di esecuzione di un comando Modbus TCP (Da quando il comando viene ricevuto a quando dopo la conversione in RTU viene ritornata la risposta).



L'ingresso **SpyOn** se attivo permette di spiare il funzionamento della FB. In caso di errore esecuzione o tempo di esecuzione superiore al tempo definito in **Timeout**, viene attivata per un loop di programma l'uscita **Fault**.

- Enable** (BOOL)      Abilitazione blocco funzione.
- SpyOn** (BOOL)      Se attivo permette di spiare il funzionamento della FB.
- MaxConn** (USINT)    Numero massimo di **streams** di I/O connessioni Modbus TCP gestiti.
- FpTCP** (@FILEP)      Pointer ad array **streams** di I/O delle connessioni Modbus TCP.
- FpRTU** (FILEP)      Flusso dati **stream** della connessione Modbus RTU.
- IFTime** (UDINT)      Tempo ricezione caratteri (µS), deve essere definito in base al baud rate.

Baud rate	Tempo
300	112000
600	56000
1200	28000
2400	14000
4800	7000
9600	3430

Baud rate	Tempo
19200	1720
38400	860
57600	573
76800	429
115200	286

- Timeout** (UINT)      Tempo massimo esecuzione comando espresso in mS. Se il comando non termina nel tempo definito viene abortito ed attivata l'uscita **Fault**.
- Enabled** (BOOL)      Blocco funzione abilitato.
- Fault** (BOOL)      Attivo per un loop di programma in caso di errore.

#### Trigger di spy

Se **SpyOn** attivo viene eseguita la funzione [SysSpyData](#) che permette di spiare il funzionamento della FB. Sono previsti vari livelli di triggers.

TFlags	Descrizione
16#00000001	<b>Tx</b> : Invio frame comando modbus RTU.
16#00000002	<b>Rx</b> : Ricezione frame risposta modbus RTU.

## Codici di errore

In caso di errore si attiva l'uscita **Fault**, con [SysGetLastError](#) è possibile rilevare il codice di errore.

10053050 Timeout esecuzione.

10053100~1 Errori gestione frame Modbus TCP.

10053200~1 Errori gestione frame Modbus RTU.

## Esempi

Semplice gateway Modbus TCP/RTU, accetta 1 connessione TCP su porta 2000 e utilizza la porta seriale COM2.

### Definizione variabili

```
VAR
  Fp : FILEP; (* File pointer *)
  MTCPGw : ModbusTCPGateway_v1; (* Modbus TCP gateway *)
  TCPServer : SysTCPServer; (* TCP server *)
  SPort : SysSerialPort; (* Serial port *)
END_VAR
```

### Esempio ST (PTP119B500, ST\_ModbusTCPGateway\_v1)

```
(* ----- *)
(* INITIALIZATIONS *)
(* ----- *)
(* Program initializations. *)

IF (SysFirstLoop) THEN

  (* Initialize serial port. *)

  SPort.COM:=ADR('COM2'); (* Serial port *)
  SPort.Baudrate:=19200; (* Baud rate *)
  SPort.Parity:='E'; (* Parity *)
  SPort.DataBits:=8; (* Data bits *)
  SPort.StopBits:=1; (* Stop bits *)
  SPort.DTRManagement:=DTR_AUTO_WO_TIMES; (* DTR management *)

  (* Initialize socket server. *)

  TCPServer.FilesArr:=ADR(Fp); (* File array *)
  TCPServer.MaxConn:=1; (* Number of connections accepted *)
  TCPServer.LocalAdd:=ADR('0.0.0.0'); (* Local address *)
  TCPServer.LocalPort:=2000; (* Local port *)
  TCPServer.LifeTm:=60; (* TSocket life time (S) *)
  TCPServer.FlushTm:=10; (* Socket flush time (mS) *)
  TCPServer.RxSize:=512; (* Rx size buffer *)
  TCPServer.TxSize:=512; (* Tx size buffer *)

  (* Initialize Modbus TCP gateway. *)

  MTCPGw.IFTime:=1720; (* Interframe time *)
  MTCPGw.Timeout:=500; (* Command execution timeout (mS) *)
END_IF;

(* ----- *)
(* MODBUS TCP/RTU GATEWAY *)
(* ----- *)
(* Execute the Modbus TCP/RTU gateway. *)

SPort(Open:=TRUE); (* Serial port management *)
TCPServer(Enable:=TRUE); (* TCP server management *)
MTCPGw.MaxConn:=TCPServer.MaxConn; (* Number of connections accepted *)
MTCPGw.FpRTU:=SPort.File; (* File pointer (Modbus RTU) *)
MTCPGw.FpTCP:=ADR(Fp); (* File pointer (Modbus TCP) *)
MTCPGw(Enable:=(TCPServer.ConnPeers <> 0)); (* Modbus TCP gateway *)

(* [End of file] *)
```

## Esempi

Semplice gateway Modbus TCP/RTU, accetta 3 connessioni TCP su porta 1000 e utilizza la porta seriale COM0.

### Definizione variabili

```

VAR
  TCPServer : SysTCPServer; (* TCP server *)
  SPort : SysSerialPort; (* Serial port *)
  MTCPGw : ModbusTCPGateway_v1; (* Modbus TCP gateway *)
  Fp : ARRAY[ 0..2 ] OF FILEP; (* File pointer *)
END_VAR
    
```

### Esempio FBD (PTP119B500, FBD\_ModbusTCPGateway\_v1)

