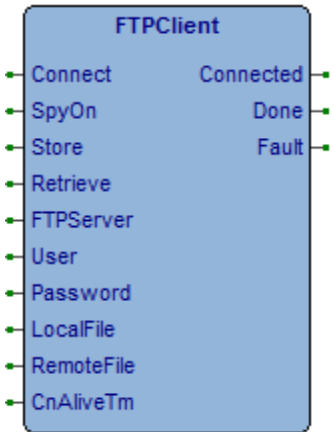
1.1.13 FTPClient, connect to a FTP server

Type	Library
FB	eLLabNetworkLib_A200



Questo blocco funzione permette di gestire la connessione ad un server FTP definito in ***FTPServer***. Con il comando ***Connect*** si forza la connessione al server utilizzando il username definito in ***User*** e la password definita in ***Password***. Se la connessione và a buon fine si attiva l'uscita ***Connected***. A connessione avvenuta ogni tempo definito in ***CnAliveTm*** viene inviato un comando di NOOP pe mantenere la connessione.

Attivando ***Store*** il file locale definito in ***LocalFile*** viene trasferito nel server FTP con il nome indicato in ***RemoteFile***. Terminato il trasferimento si attiva per un loop ***Done***.

Attivando ***Retrieve*** il file nel server FTP con il nome indicato in ***RemoteFile*** viene trasferito in locale con il nome definito in ***LocalFile***. Terminato il trasferimento si attiva per un loop ***Done***.

In caso di errore il trasferimento si interompe e s i attiva per un loop l'uscita ***Fault***.

Connect (BOOL)	Comando abilitazione connessione al server FTP.
SpyOn (BOOL)	Se attivo permette di spiare il funzionamento della FB.
Connect (BOOL)	Comando trasferimento file da locale a server FTP.
Retrieve (BOOL)	Comando trasferimento file da server FTP a locale.
FTPServer (@USINT)	Puntatore stringa definizione server FTP.
User (@USINT)	Puntatore stringa definizione nome utente.
Password (@USINT)	Puntatore stringa definizione password accesso.
LocalFile (@USINT)	Puntatore stringa definizione nome del file locale.
RemoteFile (@USINT)	Puntatore stringa definizione nome del file su server FTP.
CnAliveTm (REAL)	Tempo invio comando NOOP. Se “0” comando non viene inviato (S)
Connected (BOOL)	Si attiva se connessione al server FTP avvenuta.
Done (BOOL)	Si attiva per un loop al termine della esecuzione comando.
Fault (BOOL)	Attivo per un loop se errore esecuzione comando.

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	***Tx:*** Comandi inviati al server FTP.
16#00000002	***Rx:*** Stati risposta dal server FTP.
16#10000000	***Lg:*** Messaggio di log.
16#40000000	***Er:*** Messaggio di errore.

Codici di errore

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

Codice	Descrizione
10063020	FB eseguita in una task diversa dalla task di background.
10063050	Timeout esecuzione.
10063100~3	Errore ricezione indirizzo IP passive connection.
10063110~1	Errore ricezione porta passive connection.
10063200	Errore apertura file locale su comando Store.
10063300	Errore apertura file locale su comando Retrieve.
10063310	Errore scrittura file locale su comando Retrieve.
10063500	Ricezione stringa troppo lunga da server FTP.

Esempi

Nell'esempio attivando ***Di00CPU*** viene eseguito un comando di ***Store***, il file locale ***Storage/MyForm.bin*** viene trasferito sul server FTP con il nome ***MyForm.bin***.

Attivando ***Di01CPU*** viene eseguito un comando di ***Retrieve***, il file ***MyForm.bin*** viene trasferito dal server FTP verso il file locale ***Storage/MyForm.bin***.

Definizione variabili

	Name	Type	Address	Array	Init value	Attribute	Description
1	CaseNr	USINT	Auto	No		..	Program case
2	FTP	FTPClient	Auto	No		..	FTP client

Esempio LD *(PTP119B100, ST_FTPClient)*

```
(* Eseguo inizializzazioni. *)

IF (SysFirstLoop) THEN
  FTP.SpyOn:=TRUE; (* Spy active *)
  FTP.FTPServer:=ADR('myserver'); (* Server FTP *)
  FTP.User:='myuser'; (* User *)
  FTP.Password:='mypassword'; (* Password *)
  FTP.CnAliveTm:=10; (* Connection alive time (S) *)
END_IF;

(* Esecuzione FB, sono abilitate nei vari cases di programma. *)

FTP(); (* FTP client *)
IF (FTP.Fault) THEN CaseNr:=0; END_IF;

(* Cases gestione programma. *)

CASE (CaseNr) OF

  (* Disattivo le richieste in corso. *)

  0:
  FTP.Connect:=FALSE; (* Abilitazione connessione *)
  IF (Di00CPU) THEN CaseNr:=10; RETURN; END_IF;
  IF (Di01CPU) THEN CaseNr:=20; RETURN; END_IF;

  (* Connessione al server. *)

  10:
  FTP.Connect:=TRUE; (* Connect to server *)
  CaseNr:=CaseNr+1; (* Program case *)

  (* Attesa connessione al server. *)

  11:
  IF NOT(FTP.Connected) THEN RETURN; END_IF;
  FTP.LocalFile:=ADR('Storage/MyForm.bin');
```

```
   FTP.RemoteFile:=ADR('MyForm.bin');
   FTP.Store:=TRUE; (* Store command *)
   CaseNr:=CaseNr+1; (* Program case *)

   (* Attesa fine comando. *)

   12:
   IF NOT(FTP.Done) THEN RETURN; END_IF;
   FTP.Store:=FALSE; (* Store command *)
   CaseNr:=0; (* Program case *)

   (* Connessione al server. *)

   20:
   FTP.Connect:=TRUE; (* Connect to server *)
   CaseNr:=CaseNr+1; (* Program case *)

   (* Attesa connessione al server. *)

   21:
   IF NOT(FTP.Connected) THEN RETURN; END_IF;
   FTP.LocalFile:=ADR('Storage/MyForm.bin');
   FTP.RemoteFile:=ADR('MyForm.bin');
   FTP.Retrieve:=TRUE; (* Retrieve command *)
   CaseNr:=CaseNr+1; (* Program case *)

   (* Attesa fine comando. *)

   22:
   IF NOT(FTP.Done) THEN RETURN; END_IF;
   FTP.Retrieve:=FALSE; (* Retrieve command *)
   CaseNr:=0; (* Program case *)
END_CASE;
```