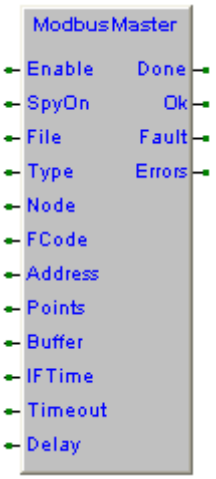
### 1.1.3 ModbusMaster, modbus master

| Type | Library |
|---|---|
| FB | eLLabUtyLib_C030 |




This function performs the management of Modbus RTU, Ascii, TCP master protocol (Selectable by ***Type***). You can define the ***File*** I/O terminal on which to communicate. Setting the ***Enable***, on the terminal I/O is sent a frame to execute the modbus function defined in the ***FCode***. Once the command execution is finished the ***Done*** output is activated. If the command execution is successful the Ok output will be set for a program loop. By disabling ***Enable*** the ***Done*** and ***Fault*** output are reset, to run again the command the ***Enable*** input must re-enabled.

If ***FCode*** is a read function, the value of variables starting from ***Address*** for the number of variables defined by ***Points***, is read from the slave system and transferred into the variables addressed by ***Buffer***. The ***SpyOn*** input allows to spy the FB working.

If ***FCode*** is a write function, the value of variables addressed by ***Buffer*** for the number of variables defined by ***Points***, is written to the slave system starting from ***Address*** address.

On error executing command or command execution time bigger than the defined ***Timeout***, the ***Fault*** output will be activated for a program loop and the ***Errors*** counter increased.

**Enable** (BOOL) Command that enables Modbus command execution. To re-execute the command, disable and then re-enable this input.

**SpyOn** (BOOL) Active allows to spy the FB working.

**File** (FILEP) Stream returned from ***Sysfopen*** function.

**Type** (USINT) Modbus type selector. 0:RTU, 1:Ascii, 2:TCP

**Node** (USINT) Modbus node number on which to execute command (Range from 0 to 255).

**FCode** (USINT) Modbus code function to execute (Range from 0 to 255).

| Code | Description |
|---|---|
| 16#01 | Read coil status (Max 255 coils) |
| 16#02 | Read input status (Max 255 inputs) |
| 16#03 | Read holding registers (Max 32 registers) |
| 16#04 | Read input registers (Max 32 registers) |
| 16#06 | Preset single register |
| 16#0F | Force multiple coils (Max 255 coils) |
| 16#10 | Preset multiple registers (Max 32 registers) |

**Address** (UINT) Address allocation of variables on slave system. According to the Modbus specifications, the address sent in the frame is (***Address-1***) (Range from 16#0001 to 16#FFFF).

**Points** (USINT) Number of consecutive variables on which the command operates.

**Buffer** (@USINT) Address of buffer of read or written data.

**IFTime** (UDINT) Time between frame (µS).
If you use the serial port, this time must be related to baud rate.

| Baud rate | Time |
|---|---|
| 300 | 112000 |
| 600 | 56000 |
| 1200 | 28000 |
| 2400 | 14000 |
| 4800 | 7000 |
| 9600 | 3430 |

| Baud rate | Time |
|---|---|
| 19200 | 1720 |
| 38400 | 860 |
| 57600 | 573 |
| 76800 | 429 |
| 115200 | 286 |
| | |

On ethernet connection set the minimum time value.

**Timeout** (UINT) Maximum command execution time expressed in mS. If the command does not end in the defined time, the command is aborted and the ***Fault*** output is activated.

**Delay** (UINT) Pause time after the execution of the command modbus expressed in mS.

**Done** (BOOL) Active at the end of command execution.

**Ok** (BOOL) Active for a program loop if command correctly executed.

**Fault** (BOOL) Active for a program loop if execution error.

**SpyTrg** (UDINT) Trigger for the SpyDataFile FB.

**SpyPtr** (@USINT) Pointer to data to be spy.

**Errors** (UDINT) Error counter. Increased at every error. If it reach the maximum value, it restarts from 0.

## Spy trigger

If ***SpyOn*** is active the ***SysSpyData*** function is executed this allows to spy the FB operations. There are various levels of triggers.

| TFlags | Description |
|---|---|
| 16#00000001 | ***Tx:*** Modbus command frame sent. |
| 16#00000002 | ***Rx:*** Modbus answer frame received. |

## Error codes

If an error occurs, the ***Fault*** output is activated and ***SysGetLastError*** can detect the error code.

| | |
|---|---|
| 10007010 | ***File*** value not defined. |
| 10007050 | Execution timeout. |
| 10007060 | Execution error. |
| 10037080 | ***Type*** definition error. |
| 10007100 | Function code defined in ***Function*** is not managed. |
| 10007120 | Wrong ***Points*** value. |
| 10007200~2 | Error transmitting the command frame. |
| 10007500~7 | Error receiving the answer frame (Wrong character, wrong length, won CRC). |