



F. Bureau

# PS501 Control Builder Plus V2.2

## New Modbus Function codes 22 & 23

# New Modbus Function Codes 22 & 23

## Contents

- General presentation
  - Modbus Function code 23
  - Modbus Function code 22
- New Libraries structures
  - Ethernet support via Modbus TCP protocol
  - Serial support via Modbus RTU protocol
- Configuration & Programming example

# New Modbus Function Codes 22 & 23

## General Presentation

- These new Modbus function codes
  - Available since PS501 Control Builder Plus - version V2.2
    - Included in library Ethernet\_AC500\_V10.lib
    - Included in library Modbus\_AC500\_V10.lib
  - Used with CPU Firmware V2.2.0 (Onboard ETH and Serial COM)
  - Not possible with Communication coupleur CM577-ETH

# New Modbus Function Codes 22 & 23

## General Presentation – Code 22

- The Modbus function code 22 (0x16) **Mask Write Register**
  - This function code is used to modify the contents of a register using a combination of an **AND mask**, an **OR mask** and the register's current contents.
  - Application: This function can be used to **Set** or **Clear** individual bits in the register.

# New Modbus Function Codes 22 & 23

## General Presentation – Code 22

- The Modbus function code 22 (0x16) **Mask Write Register**
  - The function's algorithm is:  
Result = (Current Contents **AND** And\_Mask) **OR** (Or\_Mask **AND** And\_Mask)
  - If OR\_Mask =0 the result is simply the logical ANDing of the register and And\_Mask
  - If And\_Mask =0 the result is equal to the Or\_Mask value.

- For example:

	<b>Hex</b>	<b>Binary</b>
Current Contents =	12	0001 0010
And_Mask =	F2	1111 0010
Or_Mask =	25	0010 0101
<u>And_Mask</u> =	0D	0000 1101
Result =	17	0001 0111

# New Modbus Function Codes 22 & 23

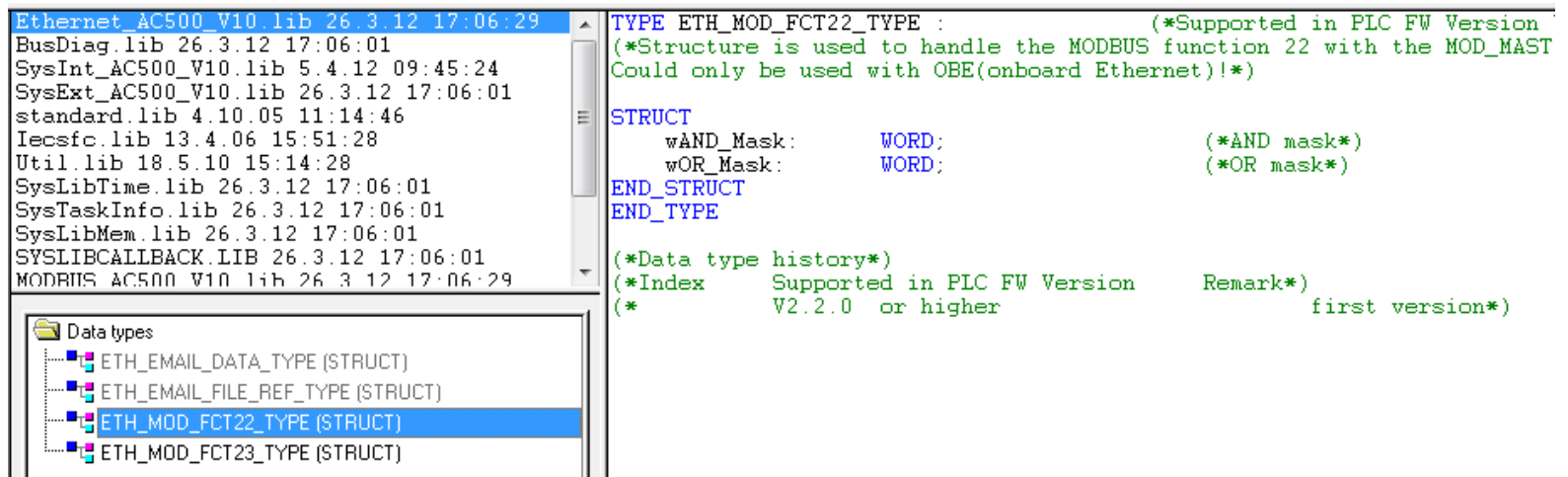
## General Presentation – Code 23

- The Modbus function code 23 (0x17) **Read/Write Multiple Registers**
  - This function code performs a combination of one **Read** operation and one **Write** operation in a single Modbus telegram.
  - The execution order of this telegram is the following:  
The **Write** operation is performed before the **Read**.

# New Modbus Function Codes 22 & 23

## Structure Ethernet support via Modbus TCP - Code 22

- The Structure `ETH_MOD_FCT22_TYPE` is used with the function block `ETH_MOD_MAST` to define the input parameters
- This structure is included in library `Ethernet_AC500_V10.lib`



```
 Ethernet_AC500_V10.lib 26.3.12 17:06:29
 BusDiag.lib 26.3.12 17:06:01
 SysInt_AC500_V10.lib 5.4.12 09:45:24
 SysExt_AC500_V10.lib 26.3.12 17:06:01
 standard.lib 4.10.05 11:14:46
 Iecsfc.lib 13.4.06 15:51:28
 Util.lib 18.5.10 15:14:28
 SysLibTime.lib 26.3.12 17:06:01
 SysTaskInfo.lib 26.3.12 17:06:01
 SysLibMem.lib 26.3.12 17:06:01
 SYSLIBCALLBACK.LIB 26.3.12 17:06:01
 MODBUS AC500 V10 lib 26.3.12 17:06:29

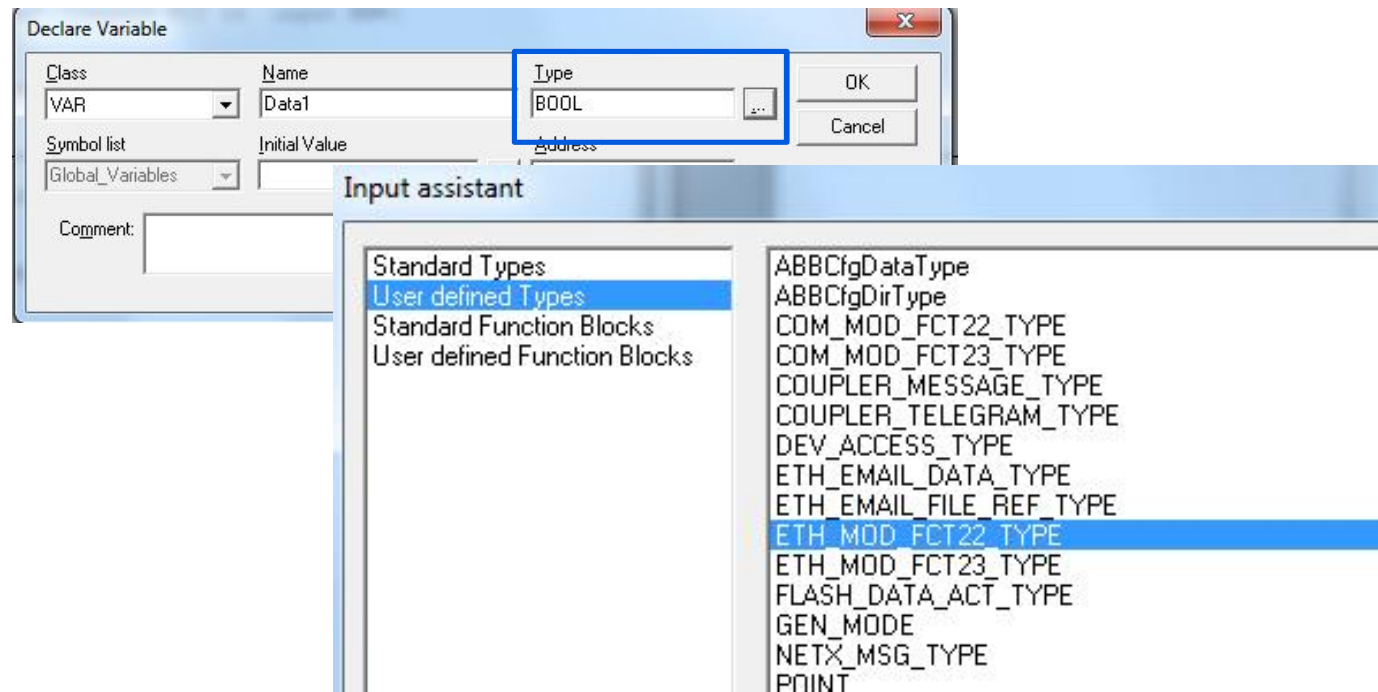
TYPE ETH_MOD_FCT22_TYPE : (*Supported in PLC FW Version
(*Structure is used to handle the MODBUS function 22 with the MOD_MAST
Could only be used with OBE(onboard Ethernet)!*)
STRUCT
    wAND_Mask: WORD: (*AND mask*)
    wOR_Mask: WORD: (*OR mask*)
END_STRUCT
END_TYPE

(*Data type history*)
(*Index Supported in PLC FW Version Remark*)
(* V2.2.0 or higher first version*)
```

# New Modbus Function Codes 22 & 23

## CoDeSys Programming - ETH/Modbus TCP - Code 22

- Using Input assistant, define a variable associated with the structure:  
User defined types / `ETH_MOD_FCT22_TYPE`

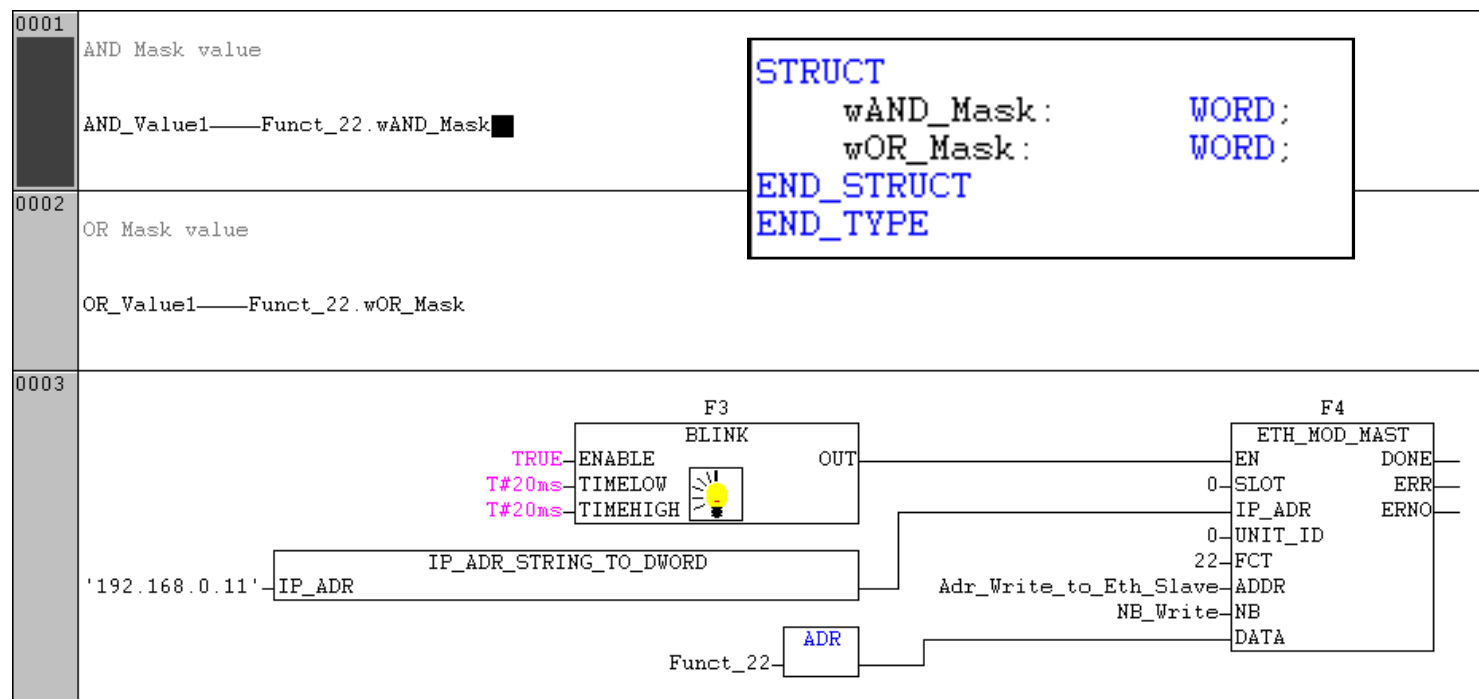




# New Modbus Function Codes 22 & 23

## Example - ETH/Modbus TCP - Code 22

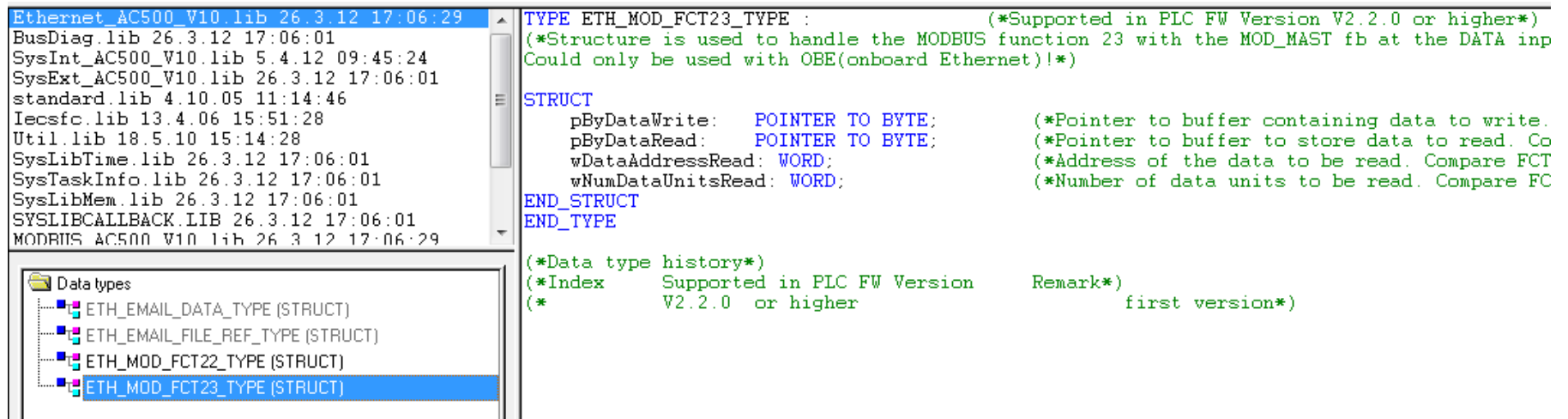
- Define the input parameters of Function block ETH\_MOD\_MAST
- The structure has to be used at the Data input with an **ADR** operator



# New Modbus Function Codes 22 & 23

## Structure Ethernet support via Modbus TCP - Code 23

- The Structure `ETH_MOD_FCT23_TYPE` is used with the function block `ETH_MOD_MAST` to define the input parameters
- This structure is included in library `Ethernet_AC500_V10.lib`



```
Ethernet_AC500_V10.lib 26.3.12 17:06:29
BusDiag.lib 26.3.12 17:06:01
SysInt_AC500_V10.lib 5.4.12 09:45:24
SysExt_AC500_V10.lib 26.3.12 17:06:01
standard.lib 4.10.05 11:14:46
Iecsfc.lib 13.4.06 15:51:28
Util.lib 18.5.10 15:14:28
SysLibTime.lib 26.3.12 17:06:01
SysTaskInfo.lib 26.3.12 17:06:01
SysLibMem.lib 26.3.12 17:06:01
SYSLIBCALLBACK.LIB 26.3.12 17:06:01
MODBUS_AC500_V10.lib 26.3.12 17:06:29

TYPE ETH_MOD_FCT23_TYPE : (*Supported in PLC FW Version V2.2.0 or higher*)
(*Structure is used to handle the MODBUS function 23 with the MOD_MAST fb at the DATA inp
Could only be used with OBE(onboard Ethernet)!*)

STRUCT
  pByDataWrite: POINTER TO BYTE; (*Pointer to buffer containing data to write.
  pByDataRead: POINTER TO BYTE; (*Pointer to buffer to store data to read. Co
  wDataAddressRead: WORD; (*Address of the data to be read. Compare FCT
  wNumDataUnitsRead: WORD; (*Number of data units to be read. Compare FC

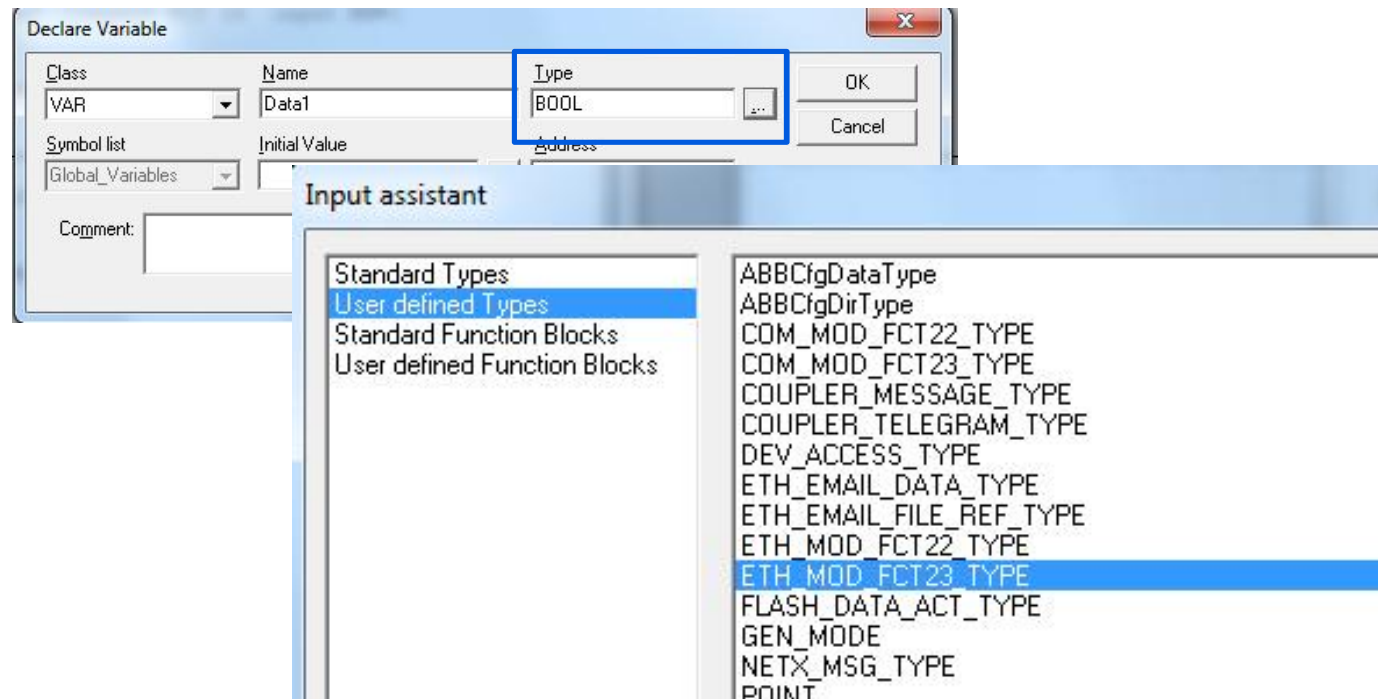
END_STRUCT
END_TYPE

(*Data type history*)
(*Index Supported in PLC FW Version Remark*)
(* V2.2.0 or higher first version*)
```

# New Modbus Function Codes 22 & 23

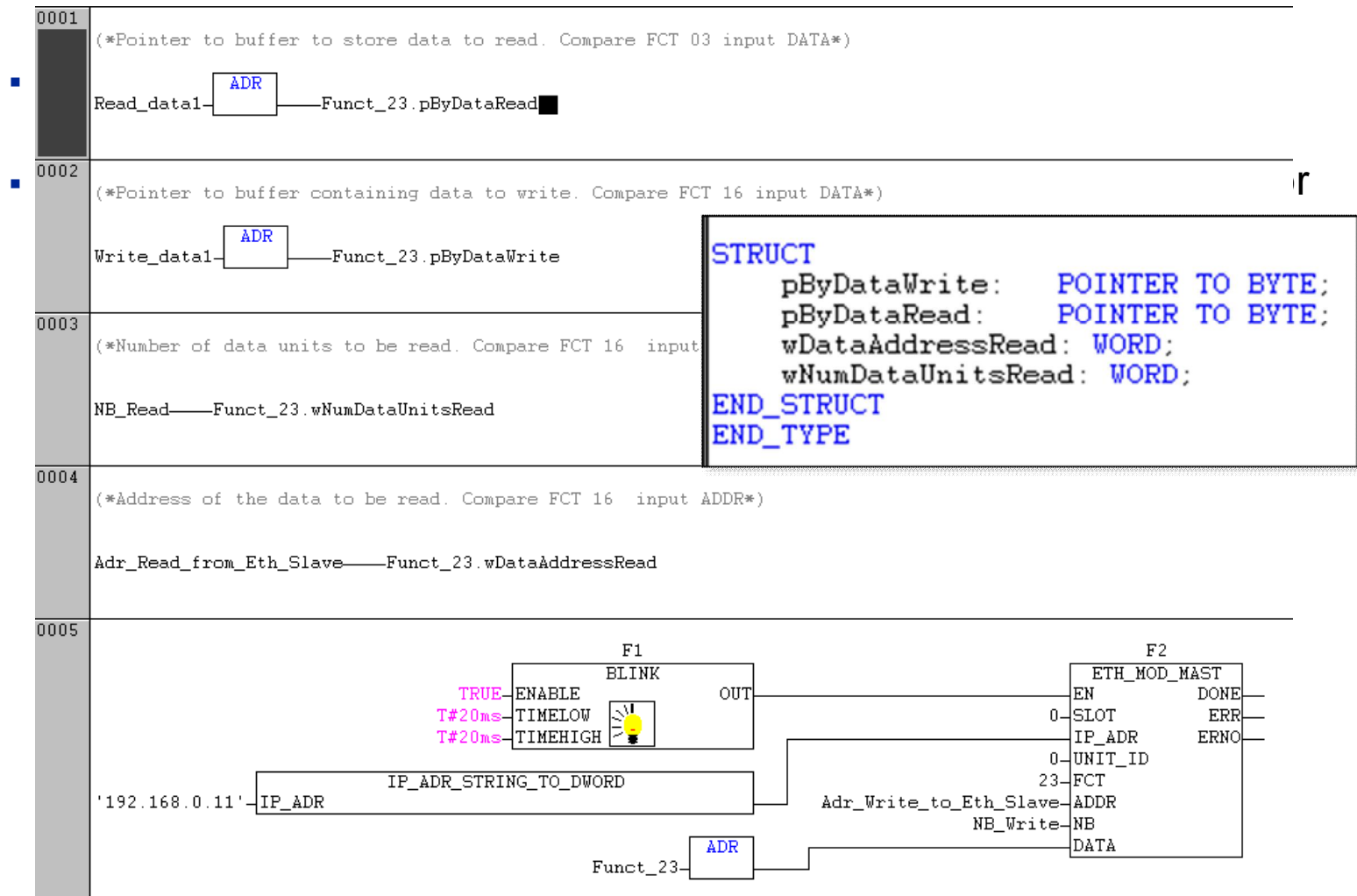
## CoDeSys Programming - ETH/Modbus TCP - Code 23

- Using Input assistant, define a variable associated with the structure:  
User defined types / `ETH_MOD_FCT23_TYPE`



# New Modbus Function Codes 22 & 23

## Example - ETH/Modbus TCP - Code 23



# New Modbus Function Codes 22 & 23

## Structure Serial support via Modbus RTU - Code 22

- The Structure `COM_MOD_FCT22_TYPE` is used with the function block `COM_MOD_MAST` to define the input parameters
- This structure is included in library `Modbus_AC500_V10.lib`

The screenshot displays a software interface with a file list on the left and a code editor on the right. The file list includes:

- standard.lib 4.10.05 11:14:46
- Iecsfclib 13.4.06 15:51:28
- Util.lib 18.5.10 15:14:28
- SysLibTime.lib 26.3.12 17:06:01
- SysTaskInfo.lib 26.3.12 17:06:01
- SysLibMem.lib 26.3.12 17:06:01
- SYSLIBCALLBACK.LIB 26.3.12 17:06:01
- MODBUS\_AC500\_V10.lib 26.3.12 17:06:29**
- MODBUS\_EXT\_AC500\_V20.lib 26.3.12 17:06:29
- SysLibABBcfg.lib 26.3.12 17:06:27

The code editor shows the definition of the structure `COM_MOD_FCT22_TYPE`:

```
TYPE COM_MOD_FCT22_TYPE : (*Supported in PLC FW Version V2.2.0 or higher*)
(*Structure is used to handle the MODBUS function 22 with the MOD_MAST
Could only be used with the serial interface!*)
STRUCT
  wAND_Mask: WORD; (*AND mask*)
  wOR_Mask: WORD; (*OR mask*)
END_STRUCT
END_TYPE

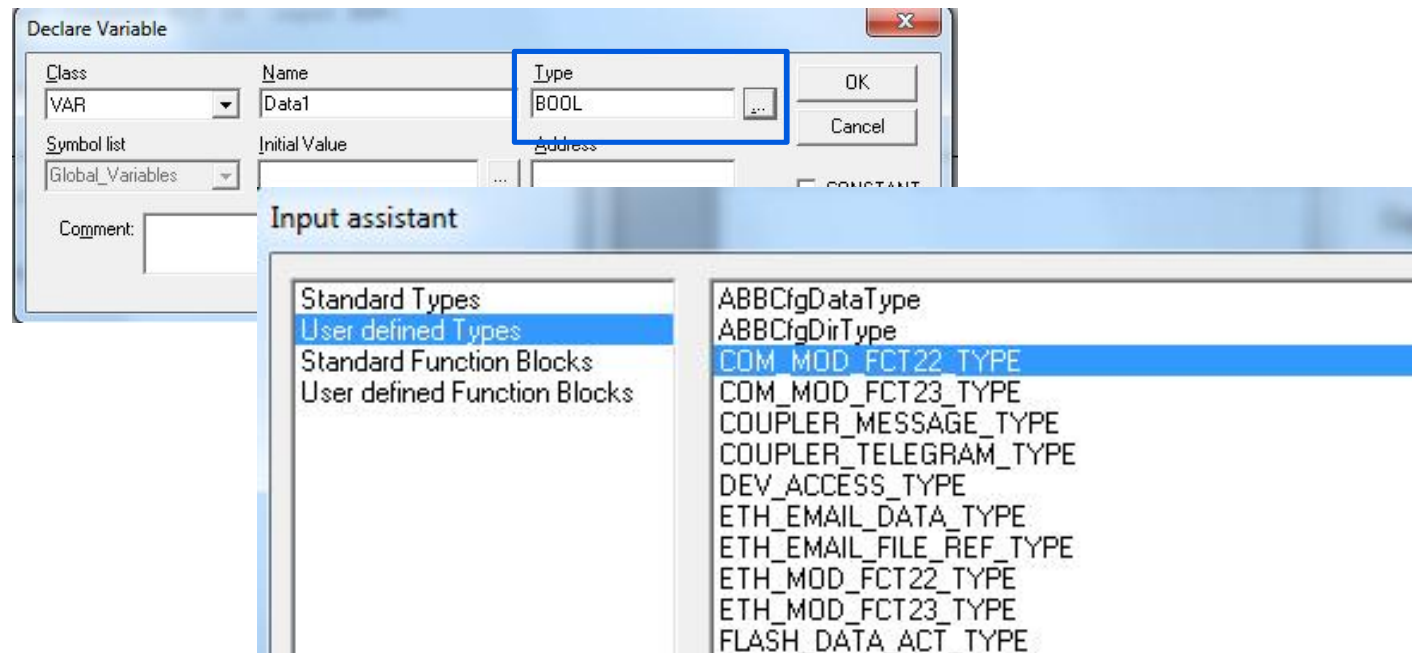
(*Data type history*)
(*Index Supported in PLC FW Version Remark*)
(* V2.2.0 or higher first version*)
```

A 'Data types' window is open, showing a tree view with 'COM\_MOD\_FCT22\_TYPE (STRUCT)' selected.

# New Modbus Function Codes 22 & 23

## CoDeSys Programming - Serial/Modbus RTU - Code 22

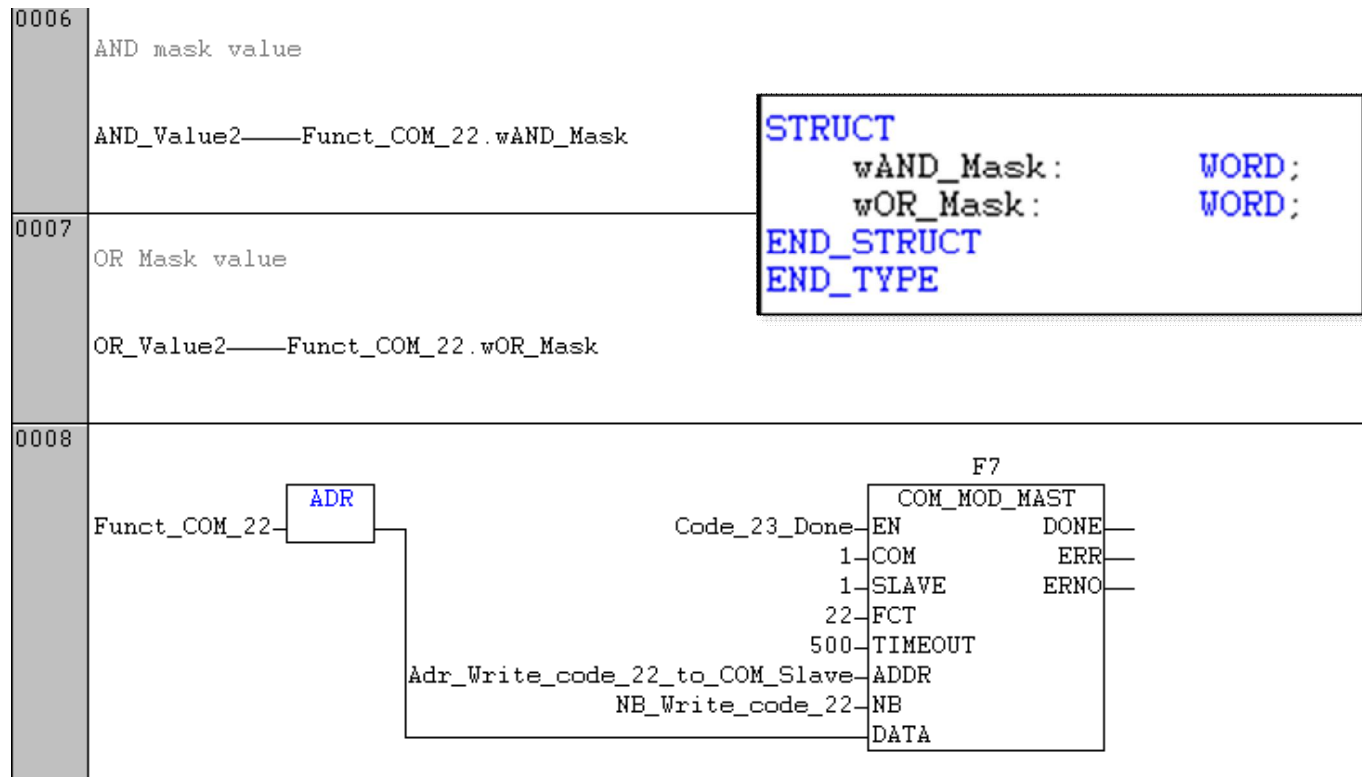
- Using Input assistant, define a variable associated with the structure:  
User defined types / `COM_MOD_FCT22_TYPE`



# New Modbus Function Codes 22 & 23

## Example - Serial/Modbus RTU - Code 22

- Define the input parameters of Function block COM\_MOD\_MAST
- The structure has to be used at the Data input with an **ADR** operator



# New Modbus Function Codes 22 & 23

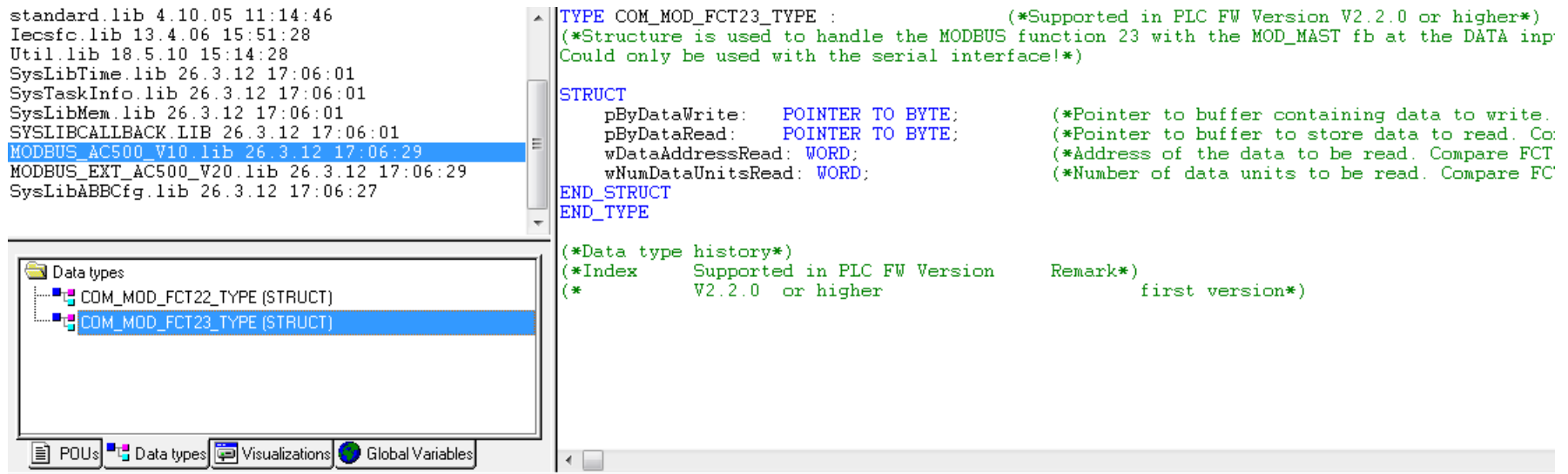
## Structure Serial support via Modbus RTU - Code 23

- The Structure `COM_MOD_FCT23_TYPE` is used with the function block `COM_MOD_MAST` to define the input parameters
- This structure is included in library `Modbus_AC500_V10.lib`

```
standard.lib 4.10.05 11:14:46
Iecsfc.lib 13.4.06 15:51:28
Util.lib 18.5.10 15:14:28
SysLibTime.lib 26.3.12 17:06:01
SysTaskInfo.lib 26.3.12 17:06:01
SysLibMem.lib 26.3.12 17:06:01
SYSLIBCALLBACK.LIB 26.3.12 17:06:01
MODBUS_AC500_V10.lib 26.3.12 17:06:29
MODBUS_EXT_AC500_V20.lib 26.3.12 17:06:29
SysLibABBcfg.lib 26.3.12 17:06:27
```

```
TYPE COM_MOD_FCT23_TYPE : (*Supported in PIC FW Version V2.2.0 or higher*)
(*Structure is used to handle the MODBUS function 23 with the MOD_MAST fb at the DATA inp
Could only be used with the serial interface!*)
STRUCT
  pByDataWrite: POINTER TO BYTE; (*Pointer to buffer containing data to write.
  pByDataRead: POINTER TO BYTE; (*Pointer to buffer to store data to read. Co
  wDataAddressRead: WORD; (*Address of the data to be read. Compare FCT
  wNumDataUnitsRead: WORD; (*Number of data units to be read. Compare FC
END_STRUCT
END_TYPE

(*Data type history*)
(*Index Supported in PIC FW Version Remark*)
(* V2.2.0 or higher first version*)
```

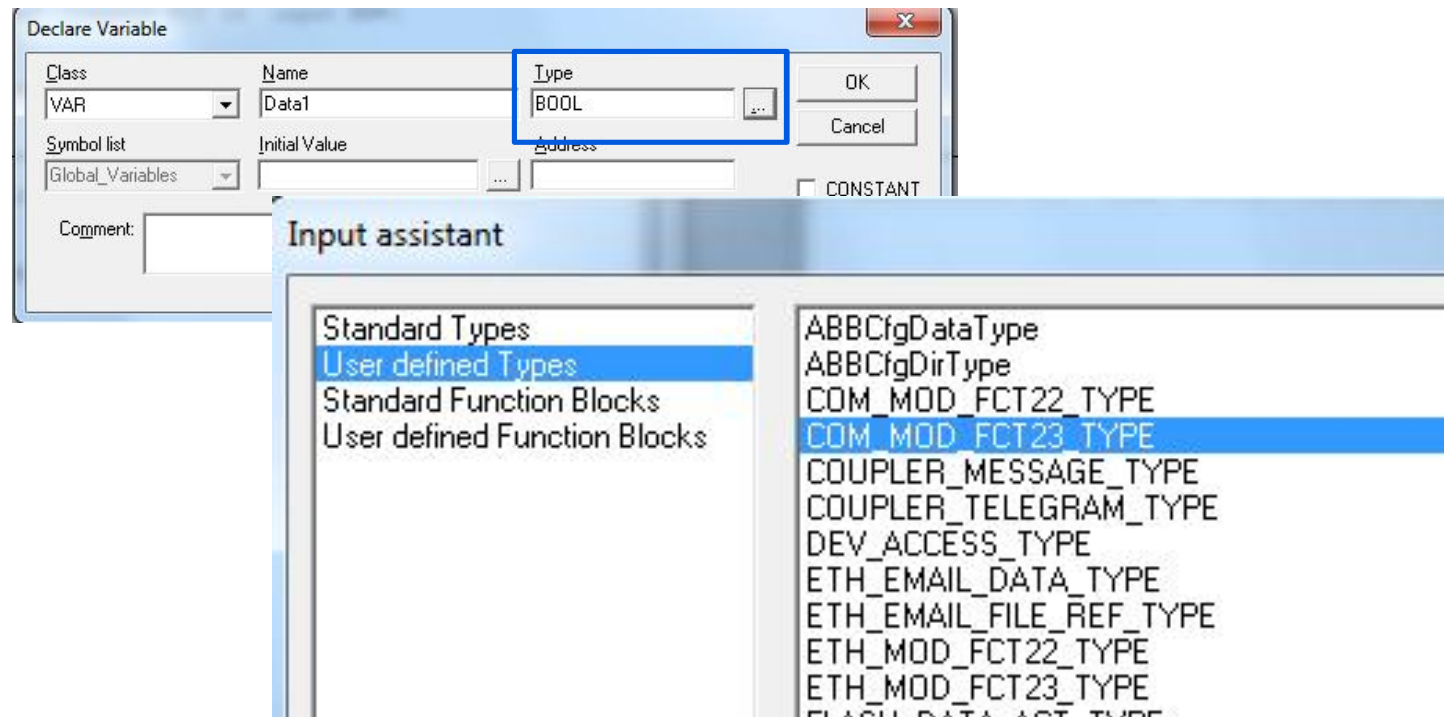




# New Modbus Function Codes 22 & 23

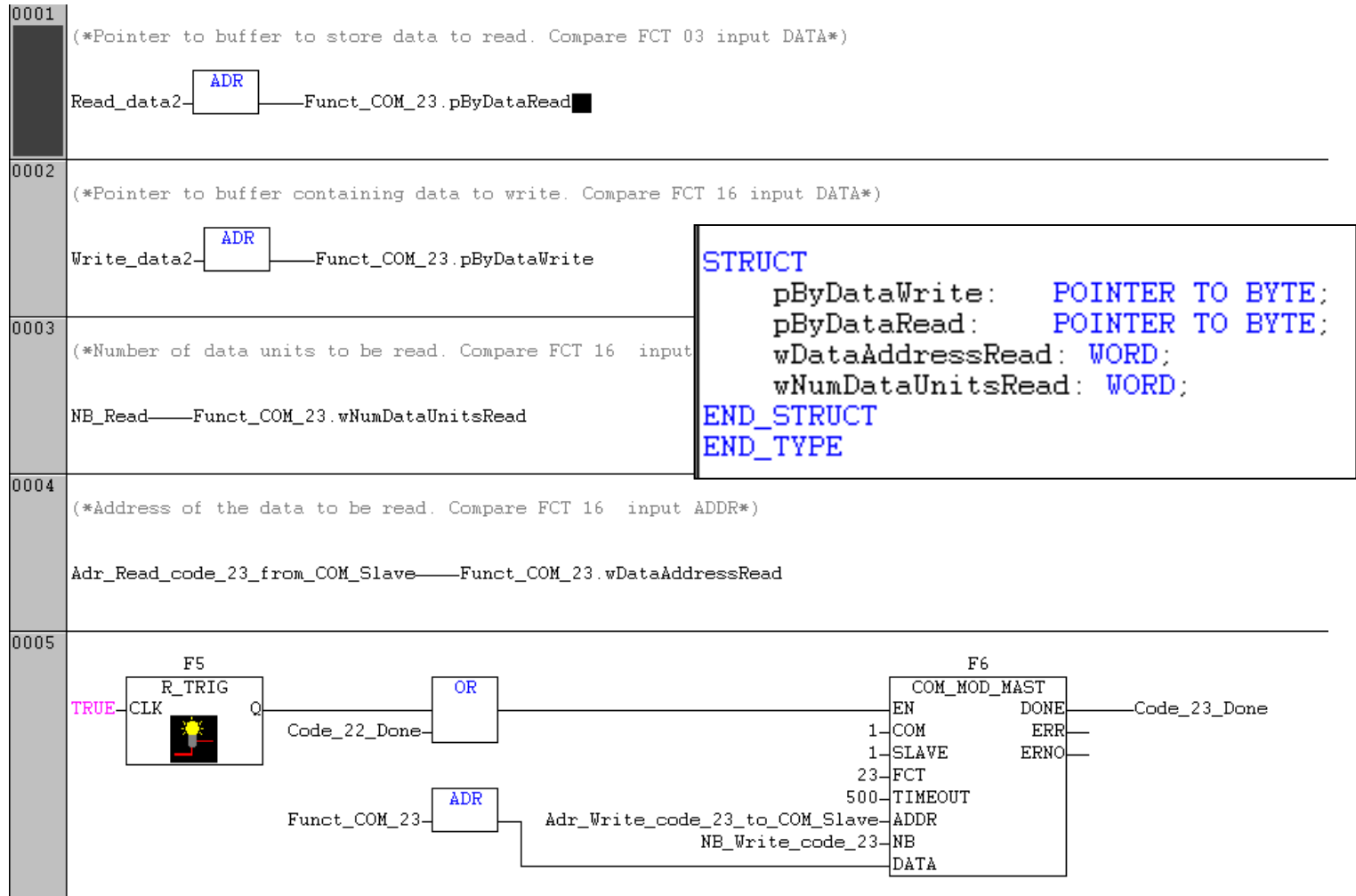
## CoDeSys Programming - Serial/Modbus RTU - Code 23

- Using Input assistant, define a variable associated with the structure:  
User defined types / [COM\\_MOD\\_FCT23\\_TYPE](#)



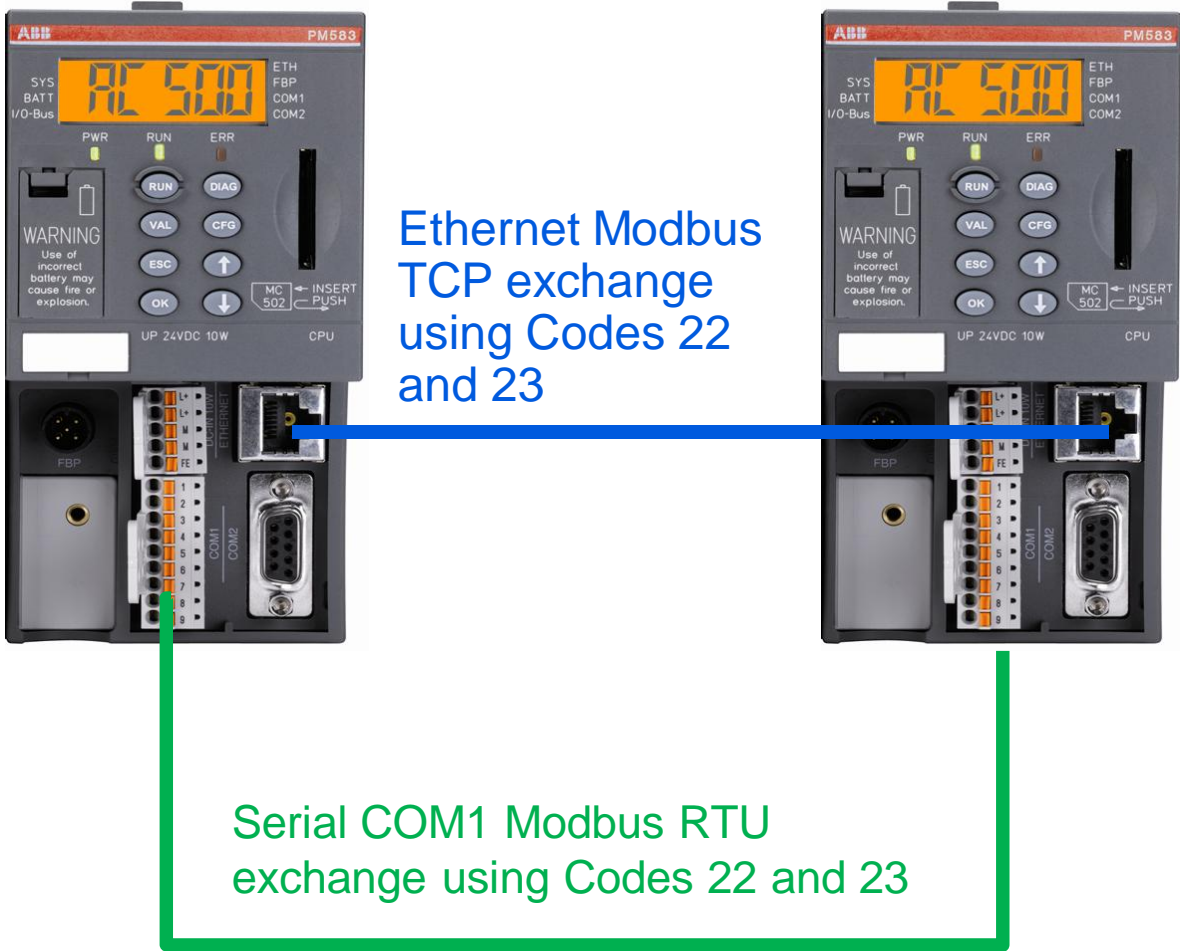
# New Modbus Function Codes 22 & 23

## Example - Serial/Modbus RTU - Code 22



# New Modbus Function Codes 22 & 23

## Example Description



Power and productivity  
for a better world™

