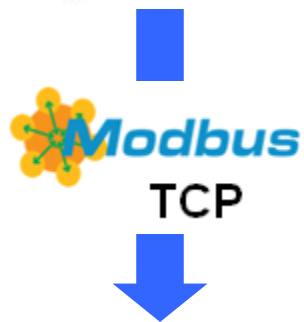


Quickstart Guide

ABB PLC and drives integration using Modbus TCP



Contents

| | |
|--|----|
| Introduction..... | 4 |
| Safety instructions..... | 4 |
| Limitations | 4 |
| Setup exmple..... | 5 |
| Hardware physical connection | 6 |
| Fieldbus modules for ABB drives..... | 6 |
| Common recommendations..... | 6 |
| RJ45 connector. | 6 |
| Connector pin assignment..... | 6 |
| Installation examples..... | 7 |
| Drive configuration..... | 8 |
| Starting up ACS355 drives with FENA-01 Ethernet adapter..... | 9 |
| Starting up ACS850, ACQ810 drives with FENA-11 Ethernet module | 11 |
| Starting up ACS880 drives with FENA-11 Ethernet adapter..... | 13 |
| Starting up ACSM1 drives with FENA-11 Ethernet adapter | 15 |
| Starting up ACS550 with RETA-01 Ethernet adapter..... | 17 |
| Starting up ACS800 drives with RETA-01 Ethernet adapter | 19 |
| Control Builder Plus for PLC and Drives..... | 21 |
| Install PS553-Drives library..... | 21 |
| Hardware configuration | 22 |
| IP configuration of CPU (for Ethernet connection between PC and PLC) | 23 |
| CODESYS | 24 |
| CODESYS tabs | 24 |
| Library Manager | 25 |
| PLC logic..... | 26 |
| Download program to PLC..... | 29 |
| Create boot project | 30 |
| Visualizations | 30 |
| Test the program via the visualization | 33 |
| Read and write more data between PLC and drive (optional) | 34 |

Quickstart Guide
ABB PLC and drives integration using Modbus TCP

Add more drives (optional)40

Other useful documentation41

Revision41

Introduction

This guide will take you through the hardware installation and configuration of ABB ACS Drives and PLC in order to prepare for Modbus TCP control. ABB-specific ready-made function blocks and visualizations from the PS553-DRIVES library will be used for the control of the drives. While the guide is applicable to all AC500 PLCs and all Modbus TCP compatible ABB ACS drives, the example is built up with an AC500-eCo PM554-ETH PLC and an ACS850 drive.

Note! *The material in this application might need to be adapted according to actual equipment and function before it is used. Testing of the equipment must always be performed by the responsible start-up person according to current legislation before the equipment is placed in service. ABB does not take any responsibility for possible damage caused by using the material in this application (shown examples, data, project tools, etc.).*

Safety instructions

Follow all safety instructions delivered with the used components:

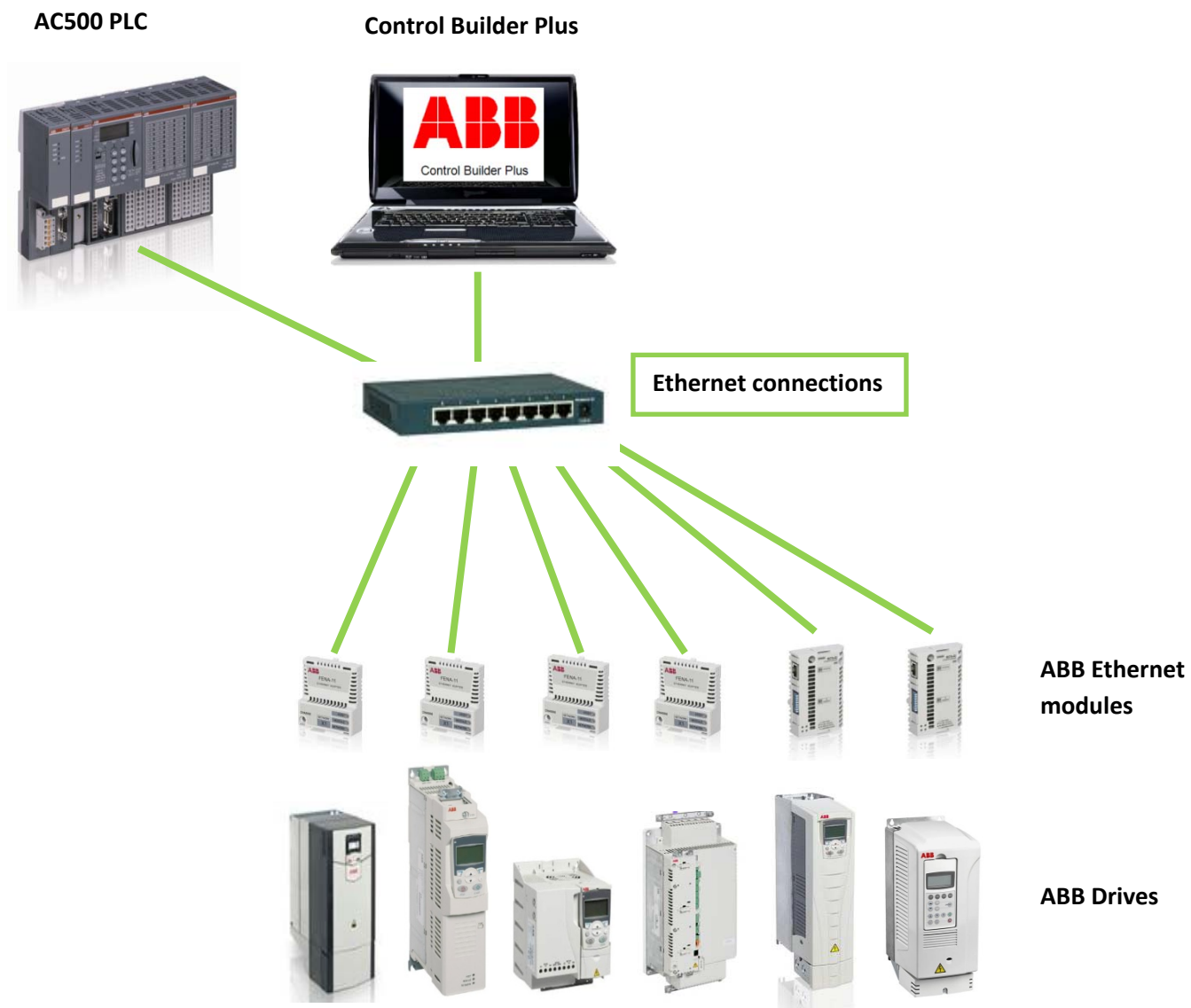
- Read the complete safety instructions for the used drive before you install, commission or use the drive. The complete safety instructions are given at the beginning of the drives user's manual (e.g. for ACS850: 3AUA0000045496 [English]).
- Read all safety and installation instructions of the used PLC. See the online help in CoDeSys (Help > Contents > Target System > AC500 / S500 > Introduction > Overview > Regulations).

Limitations

The following limitations are valid for this guide:

- General
 - All instructions in this guide are based on default settings in PLC and drives
 - PS501 Control Builder Plus version 2.2.0 or later
 - The drive must be equipped with a Modbus TCP adapter
- Function block and visualization library PS553-DRIVES compatibility
 - Communication profile: ABB Drives (classic) or ABB Drives enhanced
 - Drive types: All Modbus TCP compatible ABB ACS drives
 - Application types: Speed/Frequency control or Torque control

Setup exmple



Hardware physical connection

In the following chapter, physical connection of the hardware will be explained.

Fieldbus modules for ABB drives

- FENA-01 (Used with for instance ACS355)
- FENA-11 (Used with for instance for ACS355,ACSM1,ACS850,ACQ810 and ACS880)
- RETA-01 , RETA-02 (Used with for instance of ACS550 and ACS800)

Common recommendations

Note! The following points are only recommendations and no general rules.

- Use Switch or Router
- Cable type:
 - Use only cables with 4 pair CAT5 cables.

RJ45 connector.

Two general variants are distinguished for the pin assignment of the normally used RJ45 connectors. EIA/TIA 568 version A and version B, where the wiring according to EIA/TIA 568 version B is commonly used.

Connector pin assignment.

Today RJ45 connectors are accepted as a standard for the cabling of 10 Mbits/s networks as well for 100 Mbits/s. Generally the same pin configuration is used for both the variants.

Installation examples.

The topology of an Ethernet network is like a star or tree structure. Up to two stations can be connected to each segment where active distribution devices like hubs or switches are also considered as a station. The following figure shows an example of a simple Ethernet network.



Drive configuration

Drives need to be equipped with Modbus TCP adapter according to actual drive type.

The following drive configuration steps will adapt the drive to Modbus TCP control based on ABB-specific drives library PS553-DRIVES.

Drive parameters can be set from the drive's control panel or from drive-specific PC tool:

- DriveWindow Light for ACS355 and ACS550,
- DriveStudio for ACS850 and ACSM1,
- Drive Composer for ACS880 and DriveWindow for ACS800.

Note! All parameter settings are based on drive default settings. If the drive has been parameterized previously, return to default settings before continuing. It can be done by:

- Changing macro (and then changing back again) in parameter **99.02** for ACS310, ACS355 and ACS550.
- Setting parameter **99.03** APPLIC RESTORE to **YES** in ACS800.
- Setting parameter **16.04** Param restore to **Restore defs** in ACS850 and ACSM1.
- Setting parameter **96.06** Parameter restore to **Restore defaults** in ACS880.

Starting up ACS355 drives with FENA-01 Ethernet adapter

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter **98.02 COMM PROT SEL** to **EXT FBA**.
3. Set the FENA-01/-11 configuration parameters in group 51. At the minimum, select the communication protocol and profile with parameter **51.02** and configure the network settings with parameters **51.03...51.13**.
4. With parameter **30.18 COMM FAULT FUNC**, select how the drive reacts to a fieldbus communication break.
5. With parameter **30.19 COMM FAULT TIME**, define the time between communication break detection and the selected action.
6. Define the process data transferred to and from the drive in parameter groups 54 and 55.
Note! The adapter module sets the Status word and actual value automatically in parameters **54.01** and **54.02**, and Control word and reference in parameters **55.01** and **55.02**.
7. Validate the settings made in parameter groups 51, 54 and 55 by setting parameter **51.27 FBA PAR REFRESH** to **REFRESH**.
8. Set the relevant drive control parameters to control the drive according to the application.

Minimum required parameter settings (based on factory default settings)

| Parameter | Description | Setting | Comment |
|--------------|--------------------|----------------|--|
| 98.02 | COMM PROT SEL | EXT FBA | Activates fieldbus module |
| 51.01 | FBA TYPE | ETHERNET | Type of the connected fieldbus module. Read only |
| 51.02 | PROTOCOL / PROFILE | 0 or 1 | 0=ModbusTCP ABB Drives profile classic 1= ModbusTCP ABB Drives profile enhanced |
| 51.03 | COMMRATE | 0 | Bit rate for the Ethernet interface. |
| 51.04 | IP CONFIGURATION | 0 (Static IP) | Sets the method for configuring the IP address. 0 = Disable DCHP →static IP address |
| 51.05 | IP ADDRESS 1 | [IP address 1] | First part of IP address. E.g. 192 |
| 51.06 | IP ADDRESS 2 | [IP address 2] | Second part of IP address. E.g. 168 |
| 51.07 | IP ADDRESS 3 | [IP address 3] | Third part of IP address. E.g. 0 |
| 51.08 | IP ADDRESS 4 | [IP address 4] | Fourth part of IP address. E.g. 11 |
| 51.09 | SUBNET CIDR | [Subnet mask] | Subnet mask. E.g. 24 = 255.255.255.0 |
| 51.20 | MODBUS/TCP TIMEOUT | 20 *) | Timeout = (MODBUS/TCP Timeout value) * 100 milliseconds.*) e.g. 20 = 2 seconds |
| 51.21 | TIMEOUT MODE | 0 or 1 | 0 = none; 1 = any message, (NOT 2= control read/write as control and references are only written after changes) |
| 51.22 | WORD ORDER | 0 | Selects in which order 16 bit registers of 32 bit parameters are transferred. 0 = HILO |
| 51.27 | FBA PAR REFRESH | 1 | Updates fieldbus settings (groups 51 to 55) |

| Parameter | Description | Setting | Comment |
|--------------|-----------------|---------|---|
| 10.01 | EXT 1 COMMANDS | COMM | Fieldbus interface as source for start and stop |
| 11.02 | EXT1/EXT2 SEL | COMM | Fieldbus interface as source to switch to EXT2 |
| 11.03 | REF1 SELECT | COMM | Fieldbus interface as source for speed reference |
| 16.04 | FAULT RESET SEL | COMM | Fieldbus interface as source for fault reset |
| 11.05 | REF1 MAX | | Max speed/frequency scaling value . Must be less or equal to drive parameter max speed/frequency. |

Optional parameter settings

| Parameter | Description | Setting | Comment |
|--------------|-----------------|--------------|---|
| 30.18 | COMM FAULT FUNC | FAULT | How the drive reacts to a fieldbus communication break. |
| 30.19 | COMM FAULT TIME | Default:. 3s | Time between communication break detection and the selected action. |

More actual values or parameters to be read from drive to PLC if “ABB Drives profile ENHANCED” is used (optional)

| Parameter | Description | Setting | Comment |
|---------------------------|------------------------------------|---------|---|
| 54.03 .. 54.10 | Actual value or parameter of drive | GGii | GG = parameter group, ii = parameter index e.g. 0104 = actual current; 0145 = motor temp |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 51 to 55) |

More parameters to be written from PLC to drive if “ABB Drives profile ENHANCED” is used (optional)

| Parameter | Description | Setting | Comment |
|---------------------------|--------------------|---------|---|
| 55.03 .. 55.10 | Parameter of drive | GGii | GG = parameter group, ii = parameter index e.g. 2202 = acceleration time 1 |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 51 to 55) |

Starting up ACS850, ACQ810 drives with FENA-11 Ethernet module

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter **50.01 FBA enable** to **Enable**.
3. With parameter **50.02 Comm loss func**, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter **50.03 Comm loss t out**, define the time between communication break detection and the selected action.
5. Select application-specific values for parameters **50.04...50.11**. Examples of appropriate values are shown in the tables below.
6. Set the FENA-01/11 configuration parameters in group 51. At the minimum, select the communication protocol and profile with parameter **51.02** and configure the network settings with parameters **51.03...51.13**.
7. Define the process data transferred to and from the drive in parameter groups 52 and 53.
Note! The adapter module automatically sets the communication profile-specific virtual address for the Status word in parameter **52.01** and for the Control word in parameter **53.01**.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter **51.27 FBA par refresh** to **Refresh**.
9. Set the relevant drive control parameters to control the drive according to the application.

Minimum required parameter settings (based on factory default settings)

| Parameter | Description | Setting | Comment |
|--------------|--------------------|----------------|--|
| 50.01 | FBA ENABLE | 1 | Enables communication between the drive and fieldbus adapter. 1 = Enable. |
| 51.01 | FBA TYPE | Ethernet | Displays the type of the connected fieldbus adapter type. Read only |
| 51.02 | PROTOCOL / PROFILE | 0 or 1 | 0 = Modbus TCP ABB Drives profile classic, 1 = Modbus TCP ABB Drives profile enhanced |
| 51.03 | COMMRATE | 0 *) | Bit rate for the Ethernet interface. *) e.g. 0 = Auto-negotiate. |
| 51.04 | IP CONFIGURATION | 0 (Static IP) | Sets the method for configuring the IP address. 0 = Disable DCHP →static IP address |
| 51.05 | IP ADDRESS 1 | [IP address 1] | First part of IP address. E.g. 192 |
| 51.06 | IP ADDRESS 2 | [IP address 2] | Second part of IP address. E.g. 168 |
| 51.07 | IP ADDRESS 3 | [IP address 3] | Third part of IP address. E.g. 0 |
| 51.08 | IP ADDRESS 4 | [IP address 4] | Fourth part of IP address. E.g. 11 |
| 51.09 | SUBNET CIDR | [Subnet mask] | Subnet mask. E.g. 24 = 255.255.255.0 |
| 51.27 | FBA PAR REFRESH | 1 | Updates fieldbus settings (groups 51 to 53) |

| Parameter | Description | Setting | Comment |
|-----------|-----------------|----------------|---|
| 10.01 | EXT1 START FUNC | FB | Fieldbus interface as source for start and stop |
| 10.10 | FAULT RESET SEL | P.02.22 bit 8 | Fieldbus interface as source for fault reset |
| 12.01 | EXT1/EXT2 SEL | P.02.22 bit 15 | Fieldbus interface as source to switch to EXT2 control location |
| 19.01 | SPEED SCALING | [Scale max] | Max speed/frequency scaling value . Must be less or equal to drive parameter max speed/frequency. |
| 21.01 | SPEED REF1 SEL | FBA ref1 | Fieldbus interface as source for speed reference |

Optional parameter settings

| Parameter | Description | Setting | Comment |
|-----------|--------------------|---------|---|
| 50.02 | COMM LOSS FUNC | FAULT | Defines the reaction of the drive upon a communication break. (0 =none, 1=fault,...) |
| 51.20 | MODBUS/TCP TIMEOUT | 20 *) | Timeout = (MODBUS/TCP Timeout value) * 100 milliseconds. *) e.g. 20 = 2 seconds |
| 51.21 | TIMEOUT MODE | 0 or 1 | 0 = none; 1 = any message, (NOT 2= control read/write as control and references are only written after changes) |
| 51.22 | WORD ORDER | 0 | Selects in which order 32 bit parameters are transferred. 0 = HILO |
| 51.27 | REFRESH SETTINGS | 1 | Updates fieldbus settings (groups 51 to 53) |

More actual values or parameters to be read from drive to PLC if “ABB Drives profile ENHANCED” is used (optional)

| Parameter | Description | Setting | Comment |
|-------------------|------------------------------------|---------|---|
| 52.01 .. 52.12 | Actual value or parameter of drive | GGii | GG = parameter group, ii = parameter index e.g. 0104 = actual current; |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 51 to 53) |

More parameters to be written from PLC to drive if “ABB Drives profile ENHANCED” is used (optional)

| Parameter | Description | Setting | Comment |
|-------------------|--------------------|---------|---|
| 53.01 .. 53.12 | Parameter of drive | GGii | GG = parameter group, ii = parameter index |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 51 to 53) |

Starting up ACS880 drives with FENA-11 Ethernet adapter

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter **50.01 FBA A Enable** to **Enable**.
3. With parameter **50.02 FBA A comm loss func**, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter **50.03 FBA A comm loss t out**, define the time between communication break detection and the selected action.
5. Select application-specific values for the rest of the parameters in group 50, starting from **50.04**. Examples of appropriate values are shown in the tables below.
6. Set the FENA-11 configuration parameters in group 51. At the minimum, select the communication protocol and profile with parameter **51.02 Protocol/Profile** and configure the network settings with parameters **51.03...51.13**.
7. Define the process data transferred to and from the drive in parameter groups 52 and 53. **Note!** The adapter module automatically sets the communication profile-specific virtual address for the Status word in parameter **52.01** and for the Control word in parameter **53.01**.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter **51.27 FBA par refresh** to **Refresh**.
9. Set the relevant drive control parameters to control the drive according to the application.

Minimum required parameter settings (based on factory default settings)

| Parameter | Description | Setting | Comment |
|--------------|--------------------|----------------|--|
| 50.01 | FBA A enable | Enable | Activates fieldbus module |
| 51.01 | FBA TYPE | Ethernet | Displays the type of the connected fieldbus adapter type. Read only |
| 51.02 | Protocol / Profile | 1 | Application protocol and communication profile. |
| 51.03 | COMMRATE | 0 | Bit rate for the Ethernet interface. |
| 51.04 | IP CONFIGURATION | 0 | Sets the method for configuring the IP address. |
| 51.05 | IP ADDRESS 1 | [IP address 1] | First part of IP address. E.g. 192 |
| 51.06 | IP ADDRESS 2 | [IP address 2] | Second part of IP address. E.g. 168 |
| 51.07 | IP ADDRESS 3 | [IP address 3] | Third part of IP address. E.g. 0 |
| 51.08 | IP ADDRESS 4 | [IP address 4] | Fourth part of IP address. E.g. 11 |
| 51.09 | SUBNET CIDR | [Subnet mask] | Subnet mask. E.g. 24 = 255.255.255.0 |
| 51.20 | MODBUS/TCP TIMEOUT | 20 *) | Timeout = (MODBUS/TCP Timeout value) * 100 milliseconds. *) e.g. 20 = 2 seconds |
| 51.22 | WORD ORDER | 0 | Selects in which order 16 bit registers of 32 bit parameters are transferred. 0 = HILO |
| 51.27 | FBA par refresh | Configure | Updates fieldbus settings (groups 50 to 57) |

| Parameter | Description | Setting | Comment |
|-----------|-----------------------|-------------------|---|
| 19.11 | Ext1/Ext2 selection | MCW Bit11 (06.01) | Fieldbus interface as source to switch to EXT2 control location |
| 20.01 | Ext 1 commands | Fieldbus A | Fieldbus interface as source for start and stop |
| 22.11 | Speed ref1 selection | FB A ref1 | Fieldbus interface as source for speed reference |
| 31.11 | Fault reset selection | P.06.01 bit 7 | Fieldbus interface as source for fault reset |
| 46.01 | Speed scaling | [Scale max] | Max speed/frequency scaling value . Must be less or equal to drive parameter max speed/frequency. |

Optional parameter settings

| Parameter | Description | Setting | Comment |
|-----------|--------------------------|------------|---|
| 50.02 | FBA A comm loss func | [optional] | Defines the reaction of the drive upon a communication break. (0 =none, 1=fault,...) |
| 50.03 | FBA A comm. loss timeout | 20 *) | Fieldbus A communication break supervision time. * 100 milliseconds. *) e.g. 20 = 2 seconds |
| 51.21 | TIMEOUT MODE | 0 or 1 | 0 = none; 1 = any message, (NOT 2= control read/write as control and references are only written after changes) |
| 51.22 | WORD ORDER | 0 | Selects in which order 16 bit registers of 32 bit parameters are transferred. 0 = HILO |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 50 to 53) |

More actual values or parameters to be read from drive to PLC if “ABB Drives profile ENHANCED” is used (optional)

| Parameter | Description | Setting | Comment |
|-------------------|------------------------------------|---------|--|
| 52.01 .. 52.12 | Actual value or parameter of drive | GGii | GG = parameter group, ii = parameter index e.g. 0104 = actual current; |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 51 to 53) |

More parameters to be written from PLC to drive if “ABB Drives profile ENHANCED” is used (optional)

| Parameter | Description | Setting | Comment |
|-------------------|--------------------|---------|---|
| 53.01 .. 53.12 | Parameter of drive | GGii | GG = parameter group, ii = parameter index |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 51 to 53) |

Starting up ACSM1 drives with FENA-11 Ethernet adapter

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter **50.01 FBA ENABLE** to **Enable**.
3. With parameter **50.02 COMM LOSS FUNC**, select how the drive reacts to a fieldbus communication break. Note that this function monitors both communication between the fieldbus master and the adapter module and communication between the adapter module and the drive.
4. With parameter **50.03 COMM LOSS T OUT**, define the time between communication break detection and the selected action.
5. Select application-specific values for parameters **50.04...50.11**. Examples of appropriate values are shown in the tables below.
6. Set the FENA-11 configuration parameters in group 51. At the minimum, select the communication protocol and profile with parameter **51.02** and configure the network settings with parameters **51.03...51.13**.
7. Define the process data transferred to and from the drive in parameter groups 52 and 53.
Note! The adapter module automatically sets the communication profile-specific virtual address for the Status word in parameter **52.01** and for the Control word in parameter **53.01**.
8. Validate the settings made in parameter groups 51, 52 and 53 by setting parameter **51.27 FBA PAR REFRESH** to **REFRESH**.
10. Set the relevant drive control parameters to control the drive according to the application.

Minimum required parameter settings (based on factory default settings)

| Parameter | Description | Setting | Comment |
|--------------|--------------------|----------------|--|
| 50.01 | FBA ENABLE | Enable | Activates fieldbus module |
| 51.01 | FBA TYPE | Ethernet | Displays the type of the connected fieldbus adapter type. Read only |
| 51.02 | PROTOCOL / PROFILE | 0 or 1 | 0 = Modbus TCP ABB Drives profile classic, 1 = Modbus TCP ABB Drives profile enhanced |
| 51.03 | COMMRATE | 0 *) | Bit rate for the Ethernet interface. *) e.g. 0 = Auto-negotiate. |
| 51.04 | IP CONFIGURATION | 0 (Static IP) | Sets the method for configuring the IP address. 0 = Disable DCHP →static IP address |
| 51.05 | IP ADDRESS 1 | [IP address 1] | First part of IP address. E.g. 192 |
| 51.06 | IP ADDRESS 2 | [IP address 2] | Second part of IP address. E.g. 168 |
| 51.07 | IP ADDRESS 3 | [IP address 3] | Third part of IP address. E.g. 0 |
| 51.08 | IP ADDRESS 4 | [IP address 4] | Fourth part of IP address. E.g. 11 |
| 51.09 | SUBNET CIDR | [Subnet mask] | Subnet mask. E.g. 24 = 255.255.255.0 |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 50 to 53) |

| Parameter | Description | Setting | Comment |
|-----------|-----------------|----------------|---|
| 10.01 | EXT1 START FUNC | FBA | Fieldbus interface as source for start and stop |
| 12.01 | EXT1/EXT2 SEL | P.02.12 bit 15 | Fieldbus interface as source to switch to EXT2 control location |
| 24.01 | SPEED REF1 SEL | FBA REF1 | Fieldbus interface as source for speed reference |
| 10.08 | FAULT RESET SEL | P.02.12 bit 8 | Fieldbus interface as source for fault reset |
| 25.02 | SPEED SCALING | [Scale max] | Max speed/frequency scaling value . Must be less or equal to drive parameter max speed/frequency. |

Optional parameter settings

| Parameter | Description | Setting | Comment |
|-----------|--------------------|---------|---|
| 50.02 | COMM LOSS FUNC | FAULT | Defines the reaction of the drive upon a communication break. (0 =none, 1=fault,...) |
| 51.20 | MODBUS/TCP TIMEOUT | 20 *) | Timeout = (MODBUS/TCP Timeout value) * 100 milliseconds. *) e.g. 20 = 2 seconds |
| 51.21 | TIMEOUT MODE | 0 or 1 | 0 = none; 1 = any message, (NOT 2= control read/write as control and references are only written after changes) |
| 51.22 | WORD ORDER | 0 | Selects in which order 32 bit parameters are transferred. 0 = HILO |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 50 to 53) |

More actual values or parameters to be read from drive to PLC if “ABB Drives profile ENHANCED” is used (optional)

| Parameter | Description | Setting | Comment |
|-------------------|------------------------------------|---------|--|
| 52.01 .. 52.12 | Actual value or parameter of drive | GGii | GG = parameter group, ii = parameter index e.g. 0104 = actual current; |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 51 to 53) |

More parameters to be written from PLC to drive if “ABB Drives profile ENHANCED” is used (optional)

| Parameter | Description | Setting | Comment |
|-------------------|--------------------|---------|---|
| 53.01 .. 53.12 | Parameter of drive | GGii | GG = parameter group, ii = parameter index |
| 51.27 | FBA PAR REFRESH | REFRESH | Updates fieldbus settings (groups 51 to 53) |

Starting up ACS550 with RETA-01 Ethernet adapter.

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter **98.02 COMM PROT SEL** to **EXT FBA**.
3. With parameter **30.18 COMM FAULT FUNC**, select how the drive reacts to a fieldbus communication break.
4. With parameter **30.19 COMM FAULT TIME**, define the time between communication break detection and the selected action.
5. Set the RETA-01 configuration parameters in group 51. At the minimum, set the required IP address in parameters **51.04...51.07**, Subnet mask in parameters **51.08...51.11** and **51.16** Protocol to **0 (Modbus TCP, ABB Drives profile)**.
6. Define the process data transferred to and from the drive in the RETA-01 configuration parameter group 51.
Note! The Status Word, Actual Speed, Control Word and Speed Reference are fixed in ACS550 and not necessary to set.
7. Validate the settings made in parameter group 51 by setting parameter **51.27 FBA PAR REFRESH** to **REFRESH**.
8. Set the relevant drive control parameters to control the drive according to the application.

Minimum required parameter settings (based on factory default settings)

| Parameter | Description | Setting | Comment |
|--------------|-----------------|-----------------|---|
| 98.02 | COMM PROT SEL | EXT FBA | Activates fieldbus communication |
| 51.01 | FBA TYPE | ETHERNET | Displays the type of the connected fieldbus adapter module. Read only |
| 51.02 | COMM RATE | 0 (Example) | Defines the baud rate of the Ethernet interface. 0 = Auto-negotiate. |
| 51.03 | DHCP | 0 (Example) | DHCP Enable / Disable . 0 = Disable |
| 51.04 | IP ADDRESS 1 | [IP address 1] | First part of IP address. E.g. 192 |
| 51.05 | IP ADDRESS 2 | [IP address 2] | Second part of IP address. E.g. 168 |
| 51.06 | IP ADDRESS 3 | [IP address 3] | Third part of IP address. E.g. 0 |
| 51.07 | IP ADDRESS 4 | [IP address 4] | Fourth part of IP address. E.g. 11 |
| 51.08 | SUBNET MASK 1 | [Subnet mask 1] | First part of Subnet mask. E.g. 255 |
| 51.09 | SUBNET MASK 2 | [Subnet mask 2] | Second part of Subnet mask. E.g. 255 |
| 51.10 | SUBNET MASK 3 | [Subnet mask 3] | Third part of Subnet mask. E.g. 255 |
| 51.11 | SUBNET MASK 4 | [Subnet mask 4] | Fourth part of Subnet mask. E.g. 0 |
| 51.16 | PROTOCOL | 0 | Selects the application protocol and communication profile. 0 = Modbus/TCP |
| 51.17 | MODBUS TIMEOUT | 20 *) | Timeout = (MODBUS/TCP Timeout value) * 100 milliseconds. *) e.g. 20 = 2 seconds |
| 51.27 | FBA PAR REFRESH | 1 | Validates any changed adapter module configuration parameter setting .After refreshing, the values reverts automatically to done. |

| Parameter | Description | Setting | Comment |
|--------------|-----------------|-------------|---|
| 10.01 | EXT 1 COMMANDS | COMM | Fieldbus interface as source for start and stop |
| 11.02 | EXT1/EXT2 SEL | COMM | Fieldbus interface as source to switch to EXT2 |
| 11.03 | REF1 SELECT | COMM | Fieldbus interface as source for speed reference |
| 11.05 | REF1 MAX | [Scale max] | Max speed/frequency scaling value . Must be less or equal to drive parameter max speed/frequency. |
| 16.04 | FAULT RESET SEL | COMM | Fieldbus interface as source for fault reset |

Optional parameter settings

| Parameter | Description | Setting | Comment |
|--------------|--------------------|---------------|---|
| 30.18 | COMM FAULT FUNC | FAULT | How the drive reacts to a fieldbus communication break. |
| 30.19 | COMM FAULT TIME | 3.0 s Example | Time between communication break detection and the selected action. |

Starting up ACS800 drives with RETA-01 Ethernet adapter

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by setting parameter **98.02 COMM. MODULE LINK** to **FIELD BUS**.
3. With parameter **30.18 COMM FLT FUNC**, select how the drive reacts to a fieldbus communication break.
4. With parameter **30.19 MAIN REF DS T-OUT**, define the time between communication break detection and the selected action.
5. Set the RETA-01 configuration parameters in group 51. At the minimum, set the required IP address in parameters **51.04...51.07**, Subnet mask in parameters **51.08...51.11** and **51.16** Protocol to **0 (Modbus TCP)**.
6. Define the process data transferred to and from the drive in the RETA-01 configuration parameter group 51.
Note! The Status Word, Actual Speed, Control Word and Speed Reference are fixed in ACS550 and not necessary to set.
7. Validate the settings made in parameter group 51 by setting parameter **51.27 FBA PAR REFRESH** to **REFRESH**.
8. Set the relevant drive control parameters to control the drive according to the application.

Minimum required parameter settings (based on factory default settings)

| Parameter | Description | Setting | Comment |
|--------------|-------------------|-----------------|--|
| 98.02 | COMM. MODULE LINK | FIELD BUS | Activates fieldbus module |
| 98.07 | COMM PROFILE | ABB DRIVES | Communication profile "ABB Drives" |
| 51.01 | FBA TYPE | ETHERNET | Displays the type of the connected fieldbus adapter module. Read only |
| 51.02 | COMM RATE | 0 *) | Defines the baud rate of the Ethernet interface. *) e.g. 0 = Auto-negotiate. |
| 51.03 | DHCP | 0 *) | DHCP Enable / Disable. 0 = Disable – static IP address |
| 51.04 | IP ADDRESS 1 | [IP address 1] | First part of IP address. E.g. 192 |
| 51.05 | IP ADDRESS 2 | [IP address 2] | Second part of IP address. E.g. 168 |
| 51.06 | IP ADDRESS 3 | [IP address 3] | Third part of IP address. E.g. 0 |
| 51.07 | IP ADDRESS 4 | [IP address 4] | Fourth part of IP address. E.g. 11 |
| 51.08 | SUBNET MASK 1 | [Subnet mask 1] | First part of Subnet mask. E.g. 255 |
| 51.09 | SUBNET MASK 2 | [Subnet mask 2] | Second part of Subnet mask. E.g. 255 |
| 51.10 | SUBNET MASK 3 | [Subnet mask 3] | Third part of Subnet mask. E.g. 255 |
| 51.11 | SUBNET MASK 4 | [Subnet mask 4] | Fourth part of Subnet mask. E.g. 0 |
| 51.16 | PROTOCOL | 0 | Selects the application protocol and communication profile. 0 = Modbus/TCP |
| 51.17 | MODBUS TIMEOUT | 20 *) | Timeout = (MODBUS/TCP Timeout value) * 100 milliseconds. *) e.g. 20 = 2 seconds |
| 51.27 | FBA PAR REFRESH | 1 | Validates any changed adapter module configuration parameter setting. After refreshing, the value reverts automatically to done. |

| Parameter | Description | Setting | Comment |
|--------------|-----------------------|-------------|---|
| 10.01 | EXT 1 STRT/STP/DIR | COMM.CW | Fieldbus interface as source for start and stop |
| 11.02 | EXT1/EXT2 SELECT | COMM.CW | Fieldbus interface as source to switch to EXT2 control location |
| 11.03 | EXT REF1 SELECT | COMM.REF | Fieldbus interface as source for speed reference |
| 11.05 | EXT REF1 MAXIMUM | [Scale max] | Max speed/frequency scaling value . Must be less or equal to drive parameter max speed/frequency. |
| 16.04 | FAULT RESET SEL | COMM.CW | Fieldbus interface as source for fault reset |
| 92.02 | MAIN DS ACT1 | 102 | Actual speed value taken from 01.02 |
| 92.03 | MAIN DS ACT2 | e.g. 105 | Actual value2. E.g. torque taken from 01.05 |

Optional parameter settings

| Parameter | Description | Setting | Comment |
|--------------|------------------|------------|---|
| 30.18 | COMM FLT FUNC | e.g. FAULT | How the drive reacts to a fieldbus communication break. |
| 30.19 | MAIN REF DS TOUT | e.g. 3.0 s | Time between communication break detection and the selected action. |

Control Builder Plus for PLC and Drives

Start the “Control Builder Plus” PC tool for PLC configuration and Drive Manager handling and go through the following steps.

Install PS553-Drives library

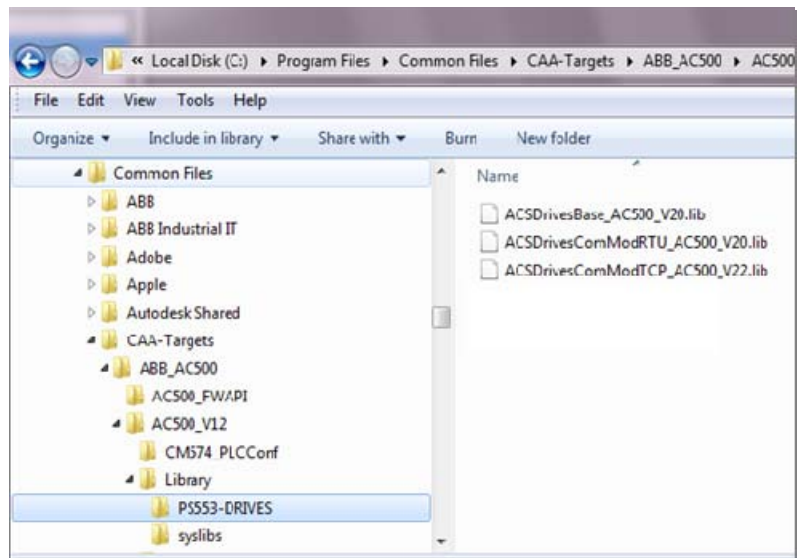
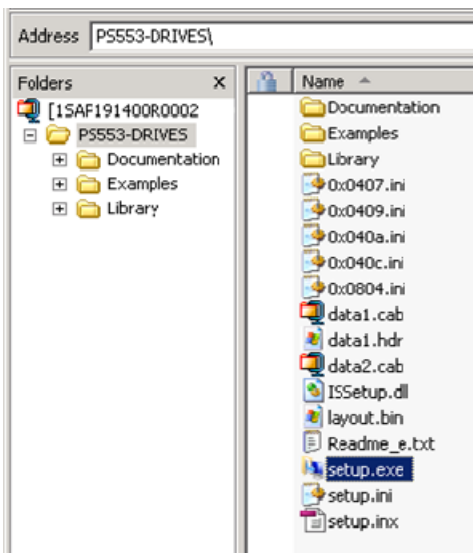
Note! In Control Builder Plus V2.3 and later, the ACSDrivesBase library is already installed. You can find it in the subfolder “PS553-DRIVES” of the default library folder (see right picture below)

Note! The version of the library can differ, nevertheless the name will stay the same! Creation date and main changes can be seen in the library manager of CODESYS only.

For older Versions of Control Builder Plus you can download the installation package in the following way from:

- www.abb.com/PLC:
- On right side menu under heading "Your preferences": Select "English" as language ... (country doesn't matter).
- On right side menu under "More Info Links" : Click on "PS501 Updates".
- Select "PS501-UPDA: PS553-DRIVES..." to download **1SAP191400R0002.zip** file.

Install the drives library “PS553-DRIVES” with ready-made function blocks and visualization objects by unzipping to a temporary directory and clicking on the file “setup.exe” (see left picture below).

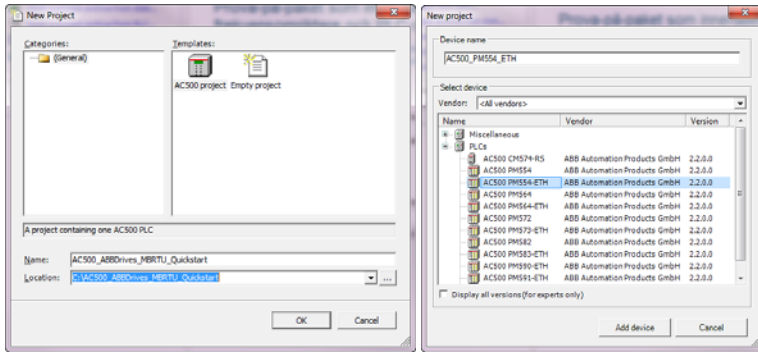


Quickstart Guide

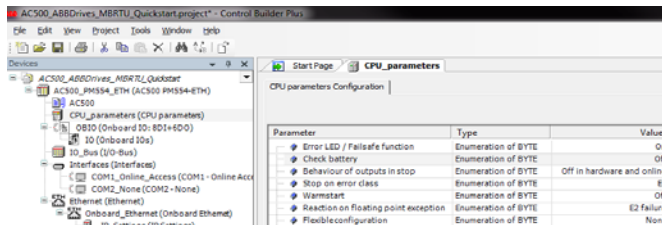
ABB PLC and drives integration using Modbus TCP

Hardware configuration

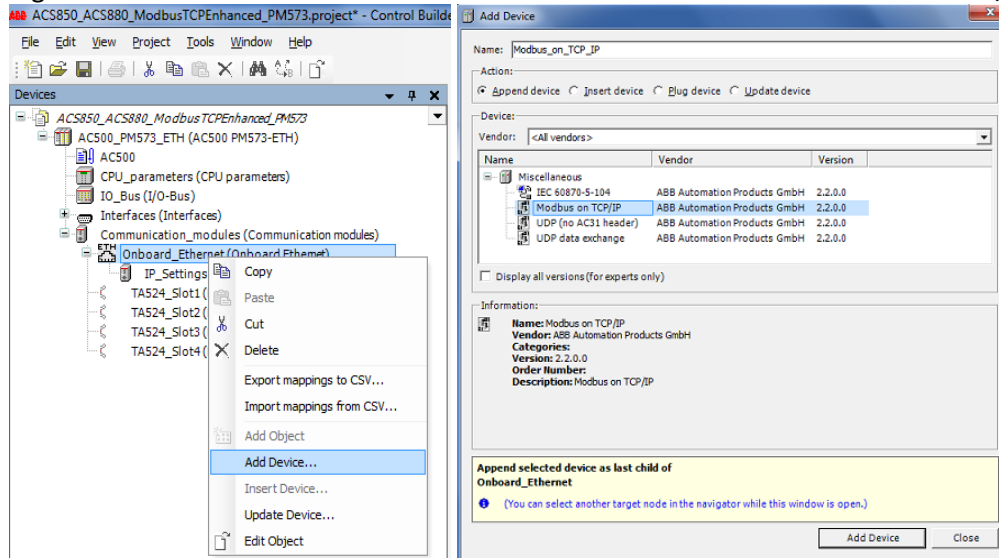
File -> New Project -> Choose “AC500 project” and name the project. Choose PLC device according to your equipment and then “Add device”.



Double-click “CPU_parameters” and configure according to your application. Set for instance “Check battery” = “Off” if your CPU does not have a battery installed.



Right click on the onboard Ethernet and click “Add device”. Select “Modbus on TCP/IP” and click “Add device”.



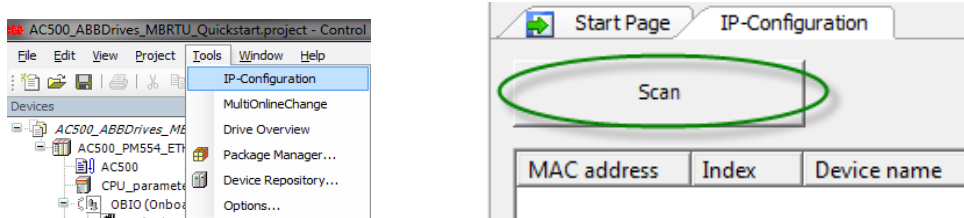
No further configurations for Modbus TCP has to be done.

IP configuration of CPU (for Ethernet connection between PC and PLC)

If the IP address has not already been configured, follow the steps below.

De-activate the firewall of the PC, or allow the connection if question comes up. Connect an Ethernet cable between PC and PLC (or via a switch).

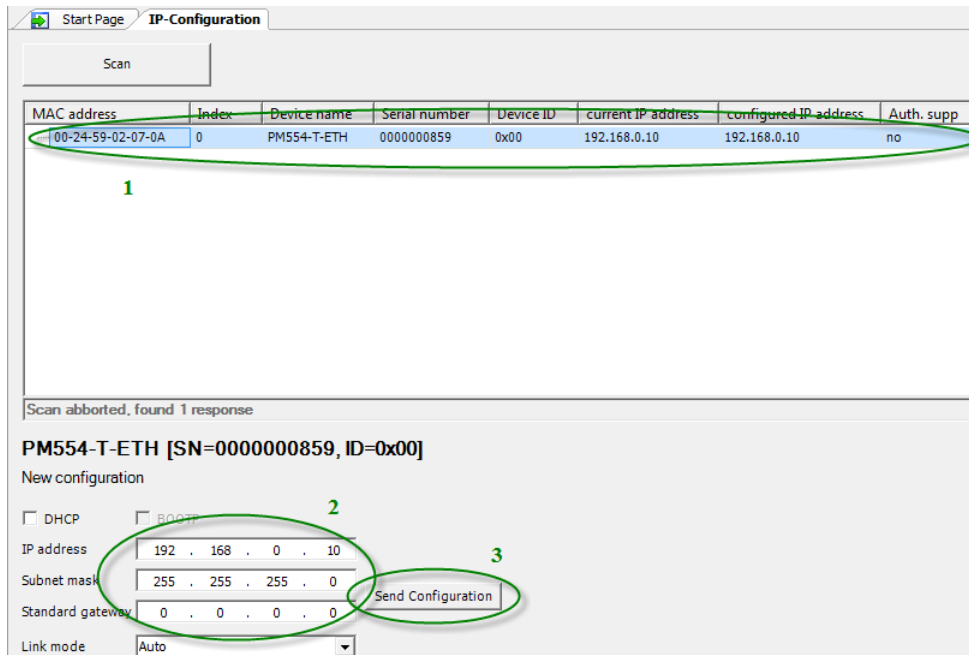
Open the IP configuration tool from the “Tools” menu and press the “Scan” button.



Select actual PLC from the scanned objects (1), make your network settings (2) and press the “Send Configuration” button. This needs to be done the first time you connect to the PLC even if the IP address is already correct.

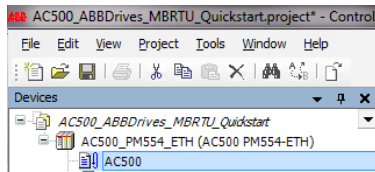
Note! The PLC needs to be in “STOP” mode for IP configuration.

After the new configuration was sent the PLC will automatically reboot.



CODESYS

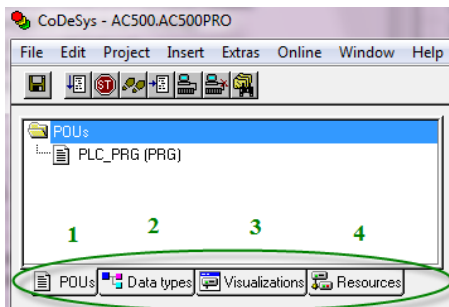
Double-click “AC500” from the Device tree in the Control Builder Plus project to open CODESYS.



CODESYS tabs

In CODESYS there are four different tabs for programming, configuration, etc.

1. POU's (Program Organization Units): This tab contains your Functions, Function blocks and Programs.
2. Data types: In this tab, along with the standard data types you can define your own user-specific data types.
3. Visualizations: In this tab you can create your own or use predefined visualizations to display your project variables graphically. In Online mode, these can then change their form, color, text, position, output, etc. in response to specified variable values.
4. Resources: In this tab you can configure and organize your project. It includes Global variables, Library manager, etc.

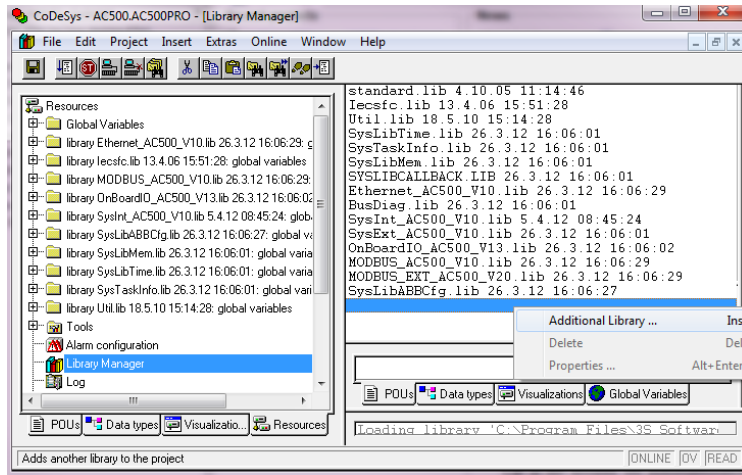


Quickstart Guide

ABB PLC and drives integration using Modbus TCP

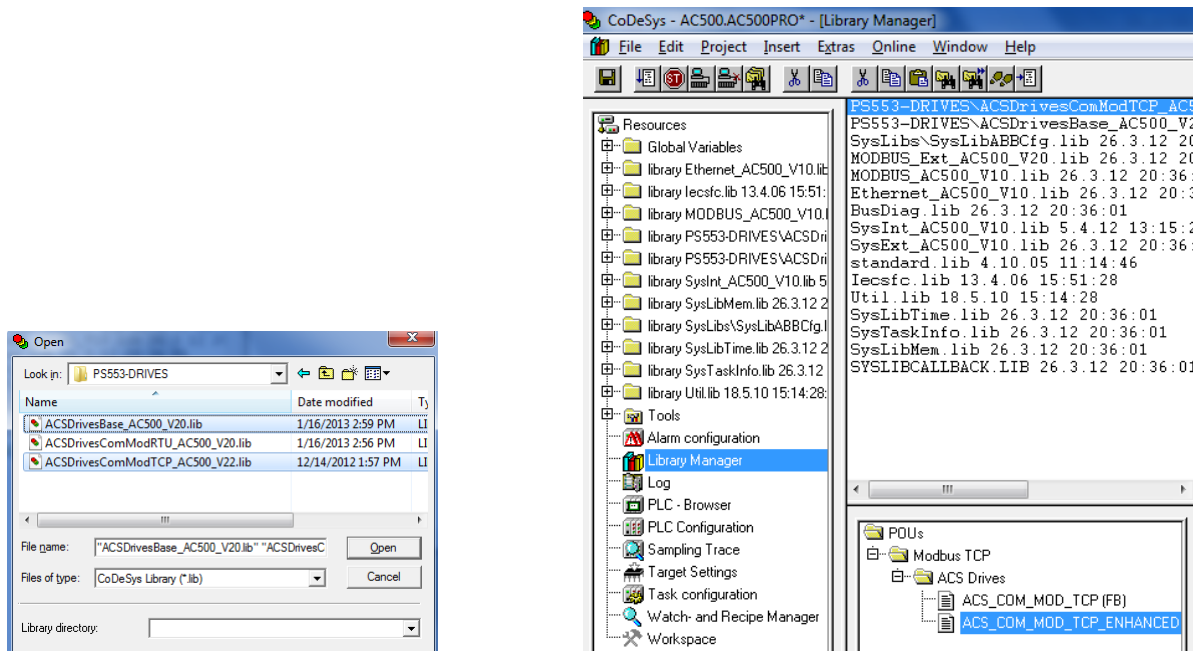
Library Manager

Open the Library Manager by double-clicking “Library Manager” from the “Resources” tab. Right-click in the library field and choose “Additional Library” (or select “Insert” from the menu).



Select the “ACSDrivesBase_AC500_V20.lib” and “ACSDrivesComModTCP_AC500_V22.lib” library files from the PS553-DRIVES folder (under the standard CODESYS library folder) and click “Open”. The libraries are added to the project.

Note! When copying projects the used libraries must be found on the same library path as on the original PC. Otherwise a warning will show that a library can not be found. If so, this red marked library has to be deleted and added newly from the now different path to the project before saving the project.



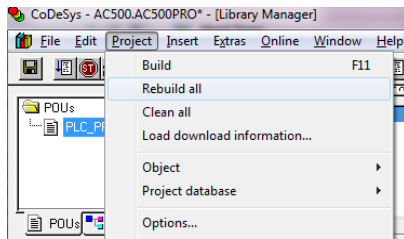
Quickstart Guide

ABB PLC and drives integration using Modbus TCP

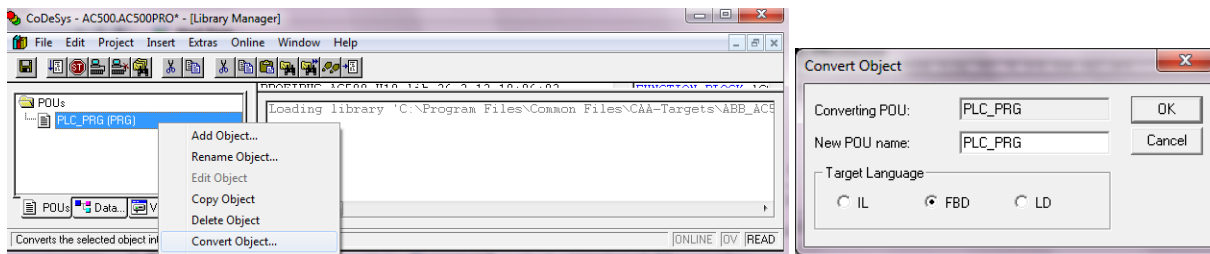
PLC logic

The following program communicates with ACS 850 drive over ModbusTCP, using the ABB Drives enhanced profile.

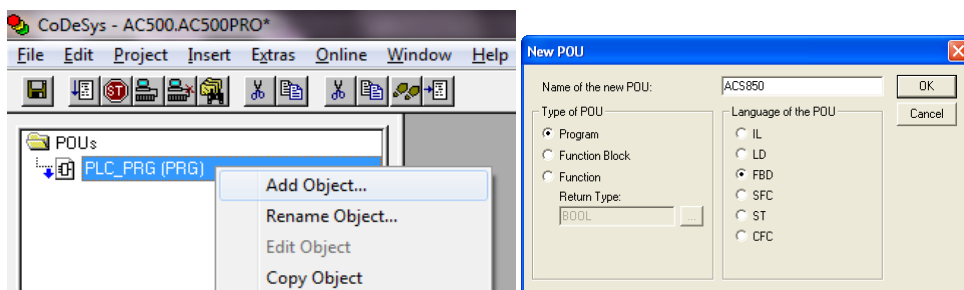
Compile your project, choose “Rebuild all” from the “Project” menu.



Right-click “PLC_PRG” in the “POUs” tab and choose “Convert Object”. Choose Target Language “FBD” and click “OK”.



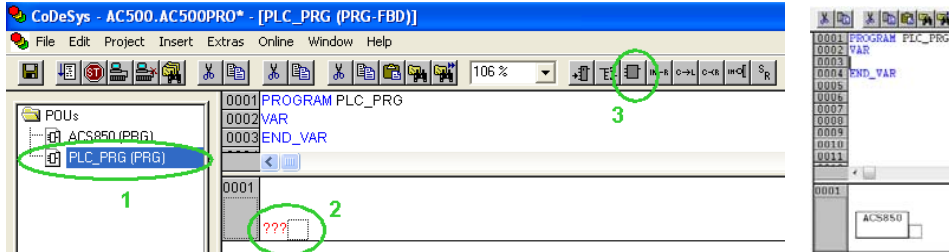
Right-click in the POU's field and choose “Add Object”. Set Type of POU to “Program” and Language of the POU to “FBD”, give the new Program a suitable name and click “OK”.



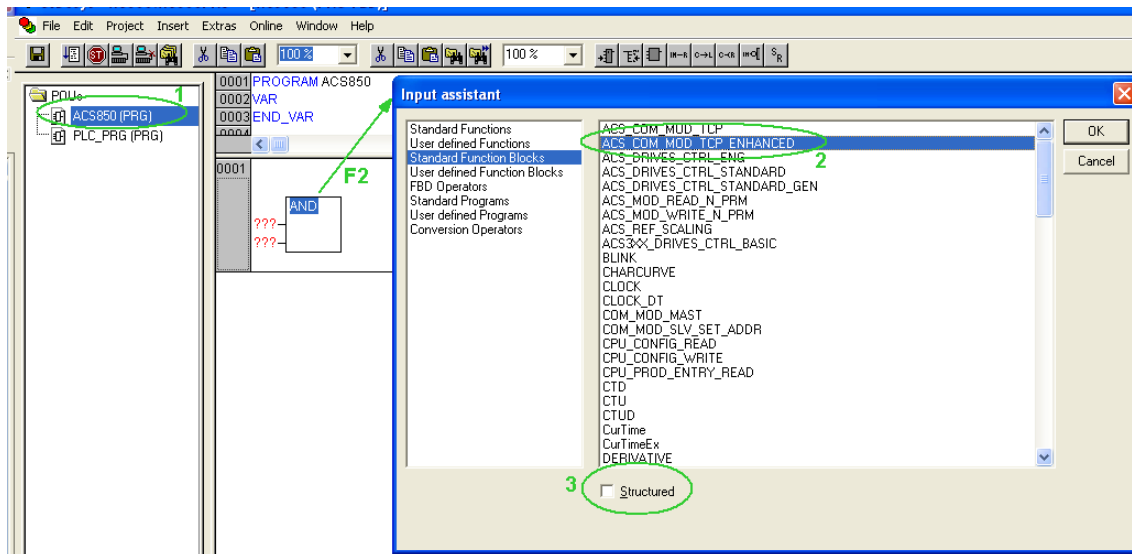
Quickstart Guide

ABB PLC and drives integration using Modbus TCP

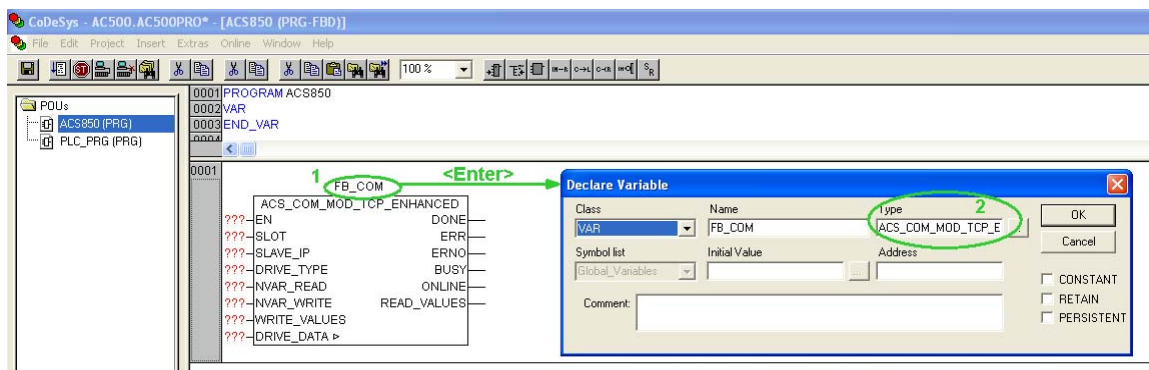
Double-click “PLC_PRG” (1) to open the main program. Select the dotted box (2) in Network 0001 and insert a box (3). Write the name of your new Program (ACS850 (PRG) in the example) to call for it from the main program.



Double-click your new program (1) and add a box as described above. This time, press F2 while the block title is selected and choose “ACS_COM_MOD_TCP_ENHANCED” (2) from “Standard Function Blocks”, then click “OK” (Tip: untick the “Structured” box in the Input assistant (3)).



Give the instance of the drive communication block a suitable name (1) and declare it as of type “ACS_COM_MOD_TCP_ENHANCED” (2) (automatic suggestion).



Quickstart Guide

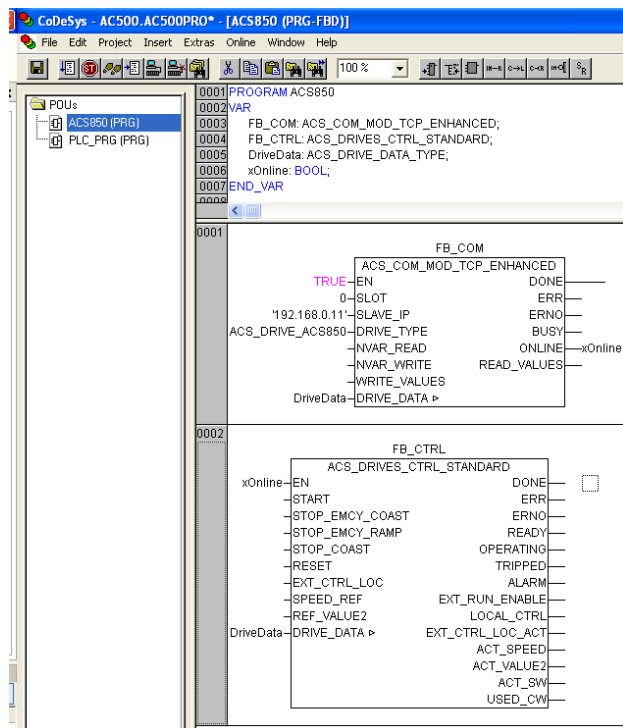
ABB PLC and drives integration using Modbus TCP

Open the program again from the “POUs” tab and connect the function block inputs and outputs according to your needs. In the example below, the block will always be enabled. Slot 0 of the PLC is used, the drive has IP Address 192.168.0.11, drive type is ACS850, no extra parameters besides Status word and Actual speed/torque are read from the drive.

Tip! Click inside the name of the function block and press F1 to open the description of the block in the online help

Create a second network (Ctrl +T) in the same program and add the block “ACS_DRIVES_CTRL_STANDARD” in the same way as for “ACS_COM_MOD_TCP_ENHANCED”. Connect the function block inputs and outputs according to your needs. In the example below:

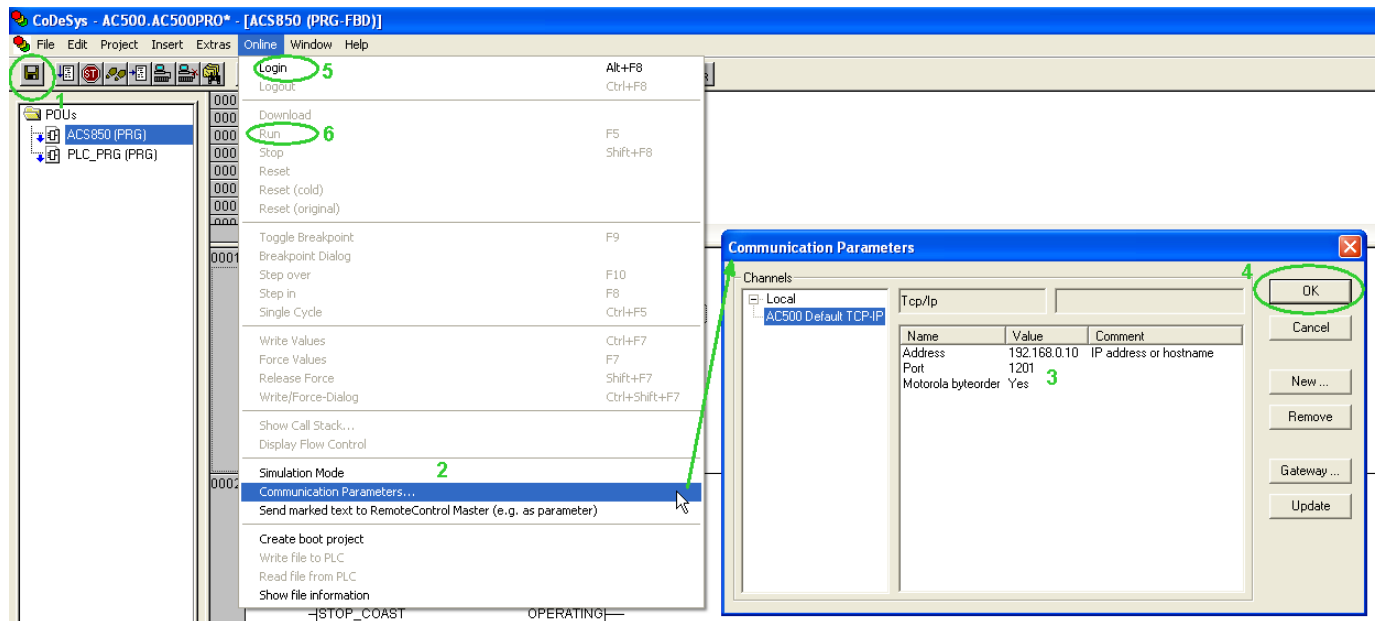
- The block will be enabled once the communication block is online.
- Other than Enable input the complete block will be controlled by using the Visualization.
- The variable connected to “ACS_DRIVES_CTRL_STANDARD” -> “DRIVE_DATA” must be the same as the one connected to “ACS_COM_MOD_TCP_ENHANCED” -> “DRIVE_DATA” and must be of type “ACS_DRIVE_DATA_TYPE”.



The variables declared between “VAR” and “END_VAR” in the picture below are local variables from the actual program (PRG). To access them from another program or a visualization the program name has to be used followed by a dot and the variable name. E.g. “ACS850.DriveData”.

Download program to PLC

Save the program (1) and choose “Communication Parameters” (2) from the “Online” menu. Set communication Parameters according to your online connection (3) and close the dialog with “OK” (4). In the example below, the “Address” 192.168.0.10 corresponds to the IP address of actual CPU and is also the default IP address of AC500 CPUs. Note that in case of Ethernet connection, the IP address of your PC port or Ethernet adapter should belong to the same subnet as the CPU, 192.168.0.XXX (XXX between 1 and 255).



Choose “Login” (5) from the “Online” menu and in the following pop-up window, click “Yes”. Then choose “Run” (6) from the “Online” menu to start the PLC. Check that the PLC goes to “RUN” mode in the PLC display, led or status bar of the CODESYS window.



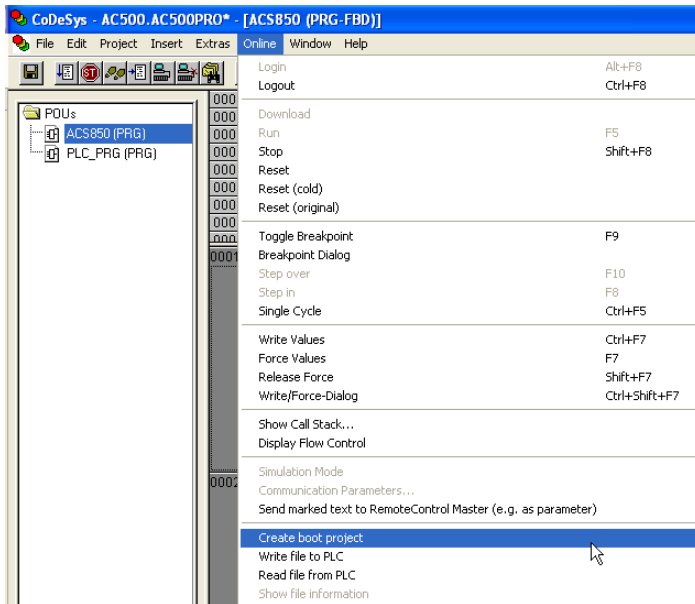
Note! If the PLC does not go to “RUN” mode, you might have some PLC errors that you need to reset. You can do that either by pressing the “DIAG” button on the PLC itself (not in the AC500-eCo series) followed by the “OK” button until all errors are reset. You can also do it in online mode by using the “diagreset” command from the “PLC Browser” in the CODESYS “Resources” tab.

Quickstart Guide

ABB PLC and drives integration using Modbus TCP

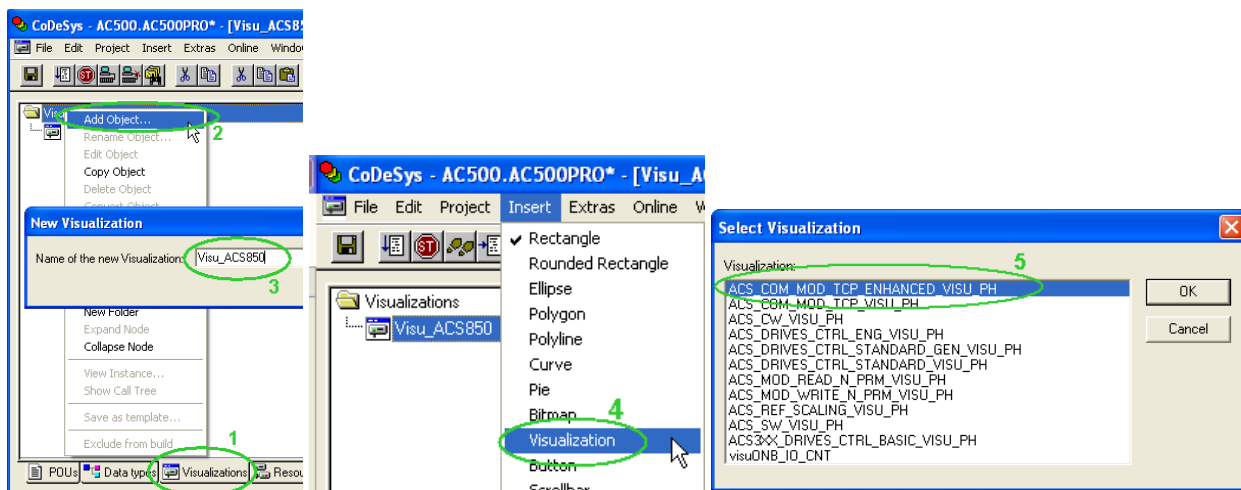
Create boot project

In “online” mode (Login), choose “Create boot project” from the “Online” menu. With this command, the compiled project is stored to the flash in such a way that the PLC will load it automatically when restarted.



Visualizations

Visualizations are **optional** and can be a good way to test the communication between PLC and drive. Go offline (“Logout” from the “Online” menu). Right-click “Visualizations” in the “Visualizations” tab (1) and choose “Add Object” (2), give the visualization page a suitable name (3). From the new page, choose “Visualization” from the “Insert” menu (4) and draw a box. Select Visualization “ACS_COM_MOD_TCP_ENHANCED_VISU_PH” (5).



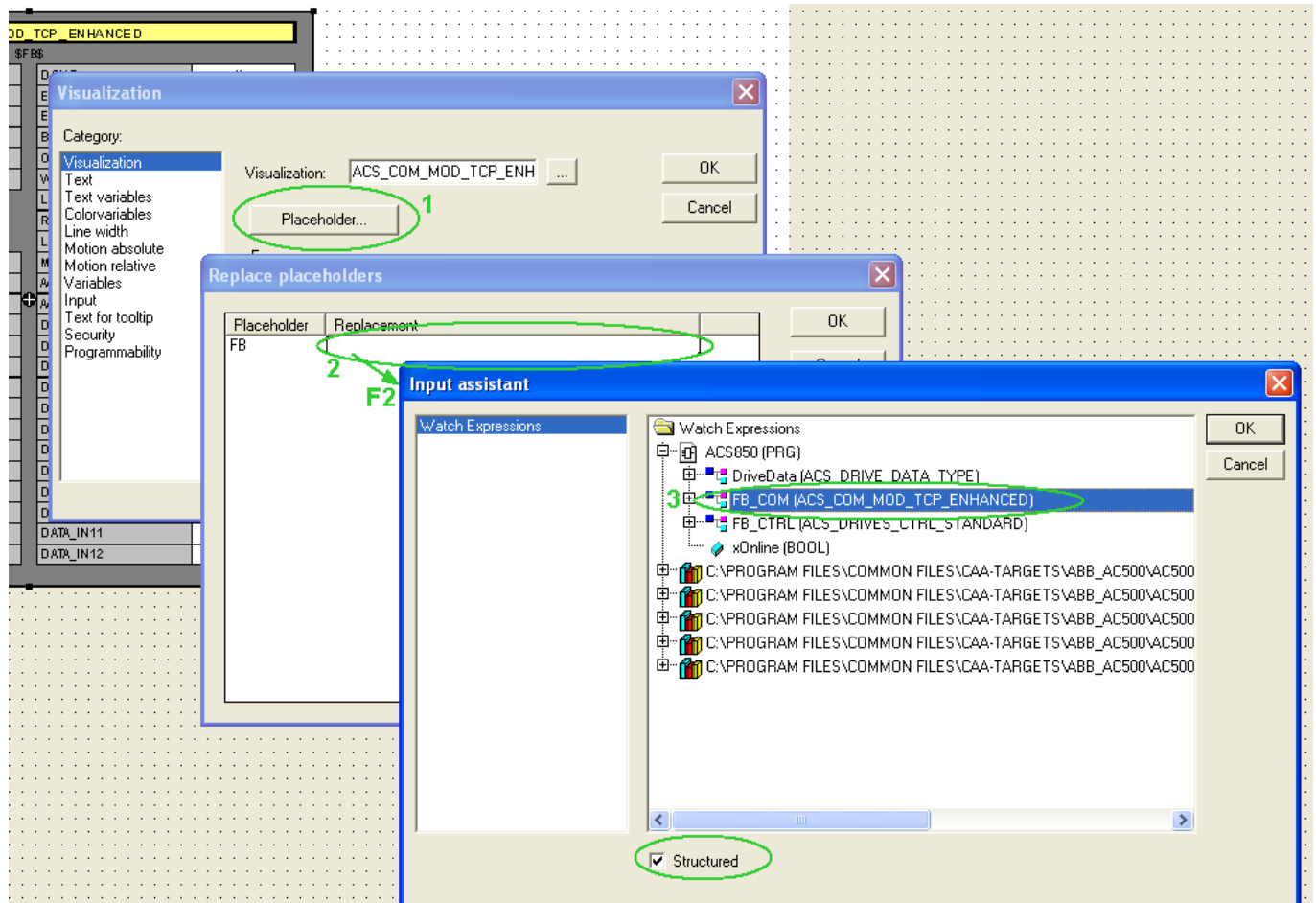
Quickstart Guide

ABB PLC and drives integration using Modbus TCP

Double-click the new visualization object for Settings and click “Placeholder” (1). Select the “Replacement” field (2) and press F2.

To make a connection to the instance of the function block “ACS_COM_MOD_TCP_ENHANCED”, choose the instance from actual program (PRG) (3) and click “OK” until all pop-up windows are closed.

Tip: Tick the “Structured” box in the Input assistant window for a better overview.



On the same page, create a visualization element for the drive control. Insert -> Visualization -> Select “ACS_DRIVES_CTRL_STANDARD_VISU_PH” -> double-click the new object and click “Placeholder” -> Press F2 from the “Replacement” field -> Choose the instance of function block “ACS_DRIVES_CTRL_STANDARD” -> click “OK” until all pop-up windows are closed.

Go online again by choosing “Login” from the “Online” menu and then “Run” from the same menu (if the PLC is not already in run mode).

The status of the function block inputs and outputs are now displayed in the visualization window.

Quickstart Guide

ABB PLC and drives integration using Modbus TCP

Note! Only Block inputs which have not been connected to the function blocks in the program “PRG” can be changed from the visualization window.

Some examples of what can be seen in the visualization:

- READY: if the drive is ready to start.
- OPERATING: if the drive is operating (magnetizing the motor), if remote control form PLC is active..
- TRIPPED: if the drive has an active fault.
- ACT_SPEED: actual speed value.
- ACT_SW: actual 16-bit Status Word from the drive.
- USED_CW: actual 16-bit Control Word sent from the PLC to the drive.
- MESSAGE field: information about the actual state, e.g. “Operation” or “Wait for rising edge of START”.

| ACS_COM_MOD_TCP_ENHANCED | | | |
|--------------------------|------------|--------------|-------|
| ACS850.FB_COM | | | |
| TRUE | EN | DONE | TRUE |
| 0 | SLOT | ERR | FALSE |
| 192.168.0.11 | IP-ADDRESS | ERNO | 0 |
| 9 | DRIVE_TYPE | BUSY | TRUE |
| 0 | NVAR_READ | ONLINE | TRUE |
| 0 | NVAR_WRITE | Write ErrCnt | 0 |
| | | LastWriteErr | 0 |
| | | ReadErrCnt | 0 |
| | | LastReadErr | 0 |
| 1151 | MCW | MSW | 5943 |
| 10000 | RefValue1 | ActValue1 | 9933 |
| 0 | RefValue2 | ActValue2 | 715 |
| 0 | DATA_OUT1 | DATA_IN1 | 0 |
| 0 | DATA_OUT2 | DATA_IN2 | 0 |
| 0 | DATA_OUT3 | DATA_IN3 | 0 |
| 0 | DATA_OUT4 | DATA_IN4 | 0 |
| 0 | DATA_OUT5 | DATA_IN5 | 0 |
| 0 | DATA_OUT6 | DATA_IN6 | 0 |
| 0 | DATA_OUT7 | DATA_IN7 | 0 |
| 0 | DATA_OUT8 | DATA_IN8 | 0 |
| 0 | DATA_OUT9 | DATA_IN9 | 0 |
| 0 | DATA_OUT10 | DATA_IN10 | 0 |
| 0 | DATA_OUT11 | DATA_IN11 | 0 |
| 0 | DATA_OUT12 | DATA_IN12 | 0 |

| ACS_DRIVES_CTRL_STANDARD | | | |
|--------------------------|--------------|--------------|-------|
| ACS850.FB_CTRL | | | |
| TRUE | EN | DONE | TRUE |
| TRUE | START | ERR | FALSE |
| TRUE | EMCY_COAST | ERNO | 0 |
| TRUE | EMCY_RAMP | READY | TRUE |
| FALSE | STOP_COAST | OPERATING | TRUE |
| FALSE | RESET | TRIPPED | FALSE |
| | | ALARM | FALSE |
| | | EXT_RUN_EN | TRUE |
| | | LOCAL_CTRL | FALSE |
| FALSE | EXT_CTRL_LOC | EXT_CTRL_LOC | FALSE |
| 10000 | SPEED_REF | ACT_SPEED | 9933 |
| 0 | REF_VALUE2 | ACT_VALUE2 | 715 |
| | | ACT_SW | 5943 |
| | | USED_CW | 1151 |
| MESSAGE | | Operation | |

Go offline by choosing “Logout” from the “Online” menu.

Further Information about the visualization elements can be found in the Help. Search for e.g. “ACS_COM_MOD_TCP_ENHANCED_VISU_PH”

Test the program via the visualization

If the drive has an active fault ("TRIPPED" output is colored orange), then try to reset by clicking on the "RESET" input = TRUE (1).

When the "READY" output is TRUE (green), then set the "SPEED_REF" input (2) to desired speed value and "START" input = TRUE (3).

Note! Speed reference is set in the block input "SPEED_REF" as a value between -20000 and 20000, where 20000 corresponds to a parameter in the drive reflecting the maximum speed of the drive (see chapter "Drive configuration").

| ACS_DRIVES_CTRL_STANDARD | | | | |
|--------------------------|-------|-----------------------|--------------|-------|
| AC 8850.FB_CTRL | | | | |
| 3 | TRUE | EN | DONE | TRUE |
| | FALSE | START | ERR | FALSE |
| | TRUE | EMCY_COAST | ERNO | 0 |
| | TRUE | EMCY_RAMP | READY | FALSE |
| | FALSE | STOP_COAST | OPERATING | FALSE |
| | FALSE | RESET | TRIPPED | TRUE |
| | | | ALARM | FALSE |
| | | | EXT_RUN_EN | TRUE |
| | | | LOCAL_CTRL | FALSE |
| 2 | FALSE | EXT_CTRL_LOC | EXT_CTRL_LOC | FALSE |
| | 10000 | SPEED_REF | ACT_SPEED | 0 |
| | 0 | REF_VALUE2 | ACT_VALUE2 | 0 |
| | | | ACT_SW | 4664 |
| | | | USED_CW | 1150 |
| MESSAGE | | TRIPPED - Reset drive | | |

This action can of course also be set by real variables directly connected to the corresponding inputs in the program.

Read and write more data between PLC and drive (optional)

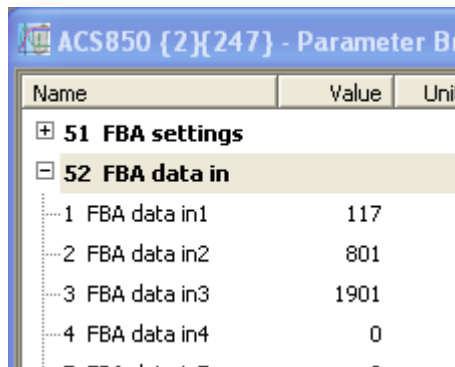
While the already implemented function block “ACS_COM_MOD_TCP_ENHANCED” has built-in functionality to read more data/parameters which must be configured in the drive at “FBA Data IN” and “FBA Data Out” groups, the function blocks “ACS_MOD_READ_N_PRM” and “ACS_MOD_WRITE_N_PRM” can read and write data/parameters from the drive according to user choice in the AC500 program.

Example - Read drive data with “ACS_COM_MOD_TCP_ENHANCED”

Values are read from “FBA Data IN” group in drive. The number of read parameters depends on the function block input “NVAR_READ”. Configuration is made in parameter group “FBA Data IN” in the drive. The supported number of parameters that can be read in this way depend on actual drive type, for instance 10 parameters can be read from ACS355 or 12 parameters can be read from ACS850, ACSM1 or ACS880.

Additional parameter settings in the drive are necessary, see example below with an ACS850 drive:

- 52.01 = 117 -> actual value 01.17 Motor Temperature will be written to READ_VALUES[1]
- 52.02 = 801 -> actual value 08.01 Actual Fault will be written to READ_VALUES[2]
- 52.03 = 1901 -> parameter 19.01 Speed Scaling will be written to READ_VALUES[3]



The screenshot shows a software window titled "ACS850 {2}{247} - Parameter Browser". It contains a table with three columns: "Name", "Value", and "Unit". The table is organized into two main sections: "51 FBA settings" and "52 FBA data in". The "52 FBA data in" section is expanded, showing four rows of data:

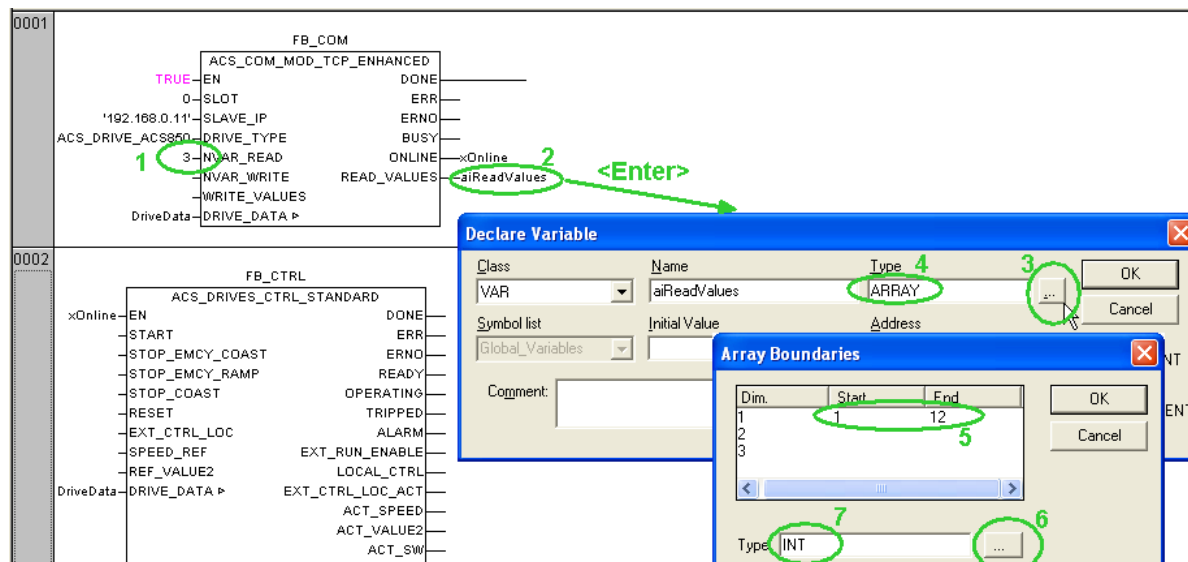
| Name | Value | Unit |
|----------------|-------|------|
| 1 FBA data in1 | 117 | |
| 2 FBA data in2 | 801 | |
| 3 FBA data in3 | 1901 | |
| 4 FBA data in4 | 0 | |

Note! After these changes of fieldbus parameter settings, the parameter 51.27 has to be set to 1 (Refresh) to activate the new settings (the fieldbus coupler FENA-11 will restart).

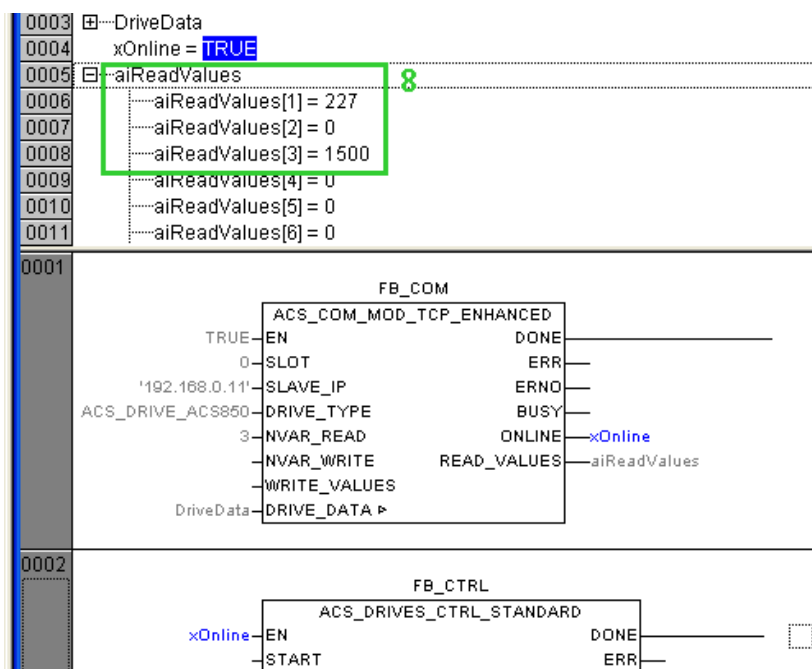
Quickstart Guide

ABB PLC and drives integration using Modbus TCP

Set “ACS_COM_MOD_TCP_ENHANCED” block input “NVAR_READ” according to number of parameters to be read (1). Create a new variable connected to block output “READ_VALUES” (2), press <Enter> and the “...” button (3) to set the variable of type “ARRAY” (4). Give the array Start index 1 and End index 12 (5), set the type to INT (6)(7) and press OK.



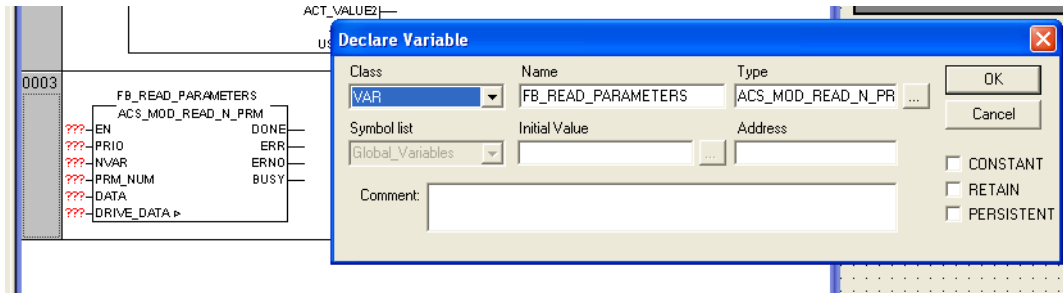
The parameter values from the drive are then written to the new array variable (8). Check this in online mode.



Example - Read drive data with "ACS_MOD_READ_N_PRM"

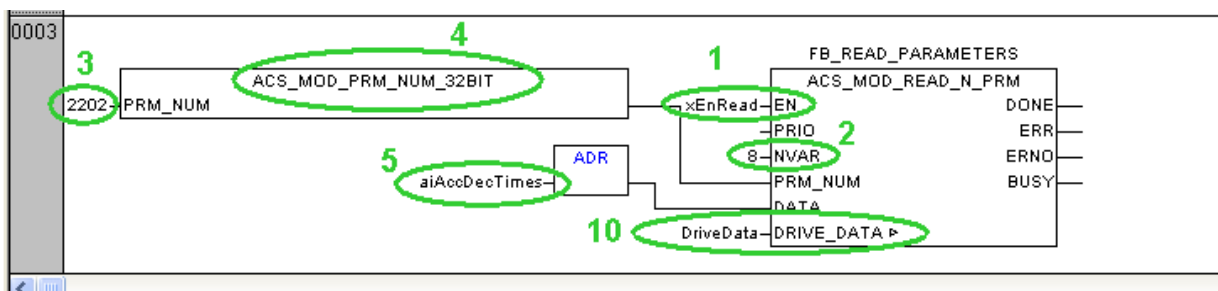
Values are read from any Modbus register address in the drive according to user choice. No additional parameter settings in the drive are necessary.

Create a new network in the same program (POU) and add the block "ACS_MOD_READ_N_PRM".



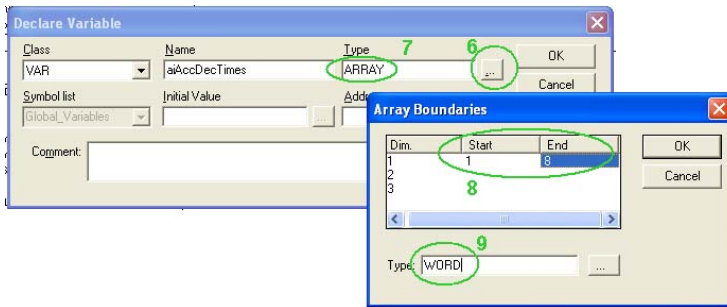
Set function block inputs according to:

- EN = xEnRead, which is a new variable (1) -> The block will be enabled if xEnRead is set to TRUE
- PRIO = Left unconnected.
- NVAR = Number of parameters to be read (2) -> 8. In this example 4 parameters of 32bit will be read, this means 8 words have to be configured.
- PRM_NUM = Start address of parameters to be read (3) -> parameters 22.02 to 22.05 (Acceleration and Deceleration times) will be read in this example. Because these parameters are 32bit parameters the conversion function "ACS_MOD_PRM_NUM_32Bit" (4) has to be used. If normal 16bit parameters are addressed this function must not be used.
- DATA is connected via an "ADR" block to a new array variable (5). Press the "..." button (6) and choose type "ARRAY" (7), set Start index to 1 and End index to the number of values to be read, 8 in this example (8). Set type to "WORD" (9), (because for 32bit values you have to consider the High and Low Word) and press "OK".
- DRIVE_DATA = the same variable that is connected to "ACS_COM_MOD_TCP_ENHANCED"-> "DRIVE_DATA" (10).



Quickstart Guide

ABB PLC and drives integration using Modbus TCP



Go online and double-click to the input variable xEnRead. A TRUE is shown besides (1). Write this value to the variable by “Online” -> “Write Values” (2) or pressing <CTRL + F7>. Then the input is set to TRUE (3) and the parameters are continuously read from the drive until you reset the input variable in the same way.



Now you can click on the “+” of the “awAccDecTime” array (1) in the variable declaration area to see all array values.

The parameter 22.02 is split into Low and High Word. The Low Word is put to array index [1] while the High Word is available in the array index [2] (2). (As the High Word is zero there is actually no need to consider the High Word.) A screenshot of a connected Drive Studio tool shows the values in the drive besides (3).

Variable Declaration:

```

0058  awAccDecTimes
0059  awAccDecTimes[1] 5000
0060  awAccDecTimes[2] 0
0061  awAccDecTimes[3] 5000
0062  awAccDecTimes[4] 0
0063  awAccDecTimes[5] 60000
0064  awAccDecTimes[6] 0
0065  awAccDecTimes[7] 60000
0066  awAccDecTimes[8] 0
        
```

2: Low Word -> Index 1, High Word -> Index 2

Values in DriveStudio:

| 22 Speed ref ramp | | |
|-------------------|-----------------|----------|
| 1 | Acc/Dec sel | C.False |
| 2 | Acc time1 | 5.000 s |
| 3 | Dec time1 | 5.000 s |
| 4 | Acc time2 | 60.000 s |
| 5 | Dec time2 | 60.000 s |
| 6 | Shape time acc1 | 0.100 s |

PLC Ladder Logic:

```

0003  ACT_VALUE2 ---
      ACT_SW ---
      USED_CW ---

      2202-PRM_NUM --- ACS_MOD_PRM_NUM_32BIT
      awAccDecTimes --- ADR --- FB_READ_PARAMETERS
      DriveData --- DRIVE_DATA
        
```

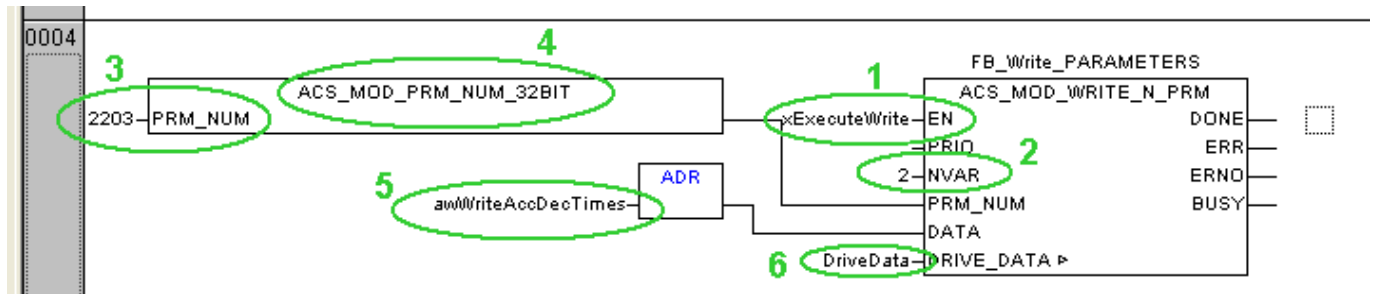
FB_READ_PARAMETERS block inputs: EN (xEnRead), Prio, NVAR (8), PRM_NUM, DATA, DRIVE_DATA.

Note! Several “ACS_MOD_READ_N_PRM” blocks can be activated at the same time.

Example - Write drive data with "ACS_MOD_WRITE_N_PRM"

Values are written to any Modbus register addresses of the drive according to user choice. No additional parameter settings in the drive are necessary.

Create a new network in the same program (POU) and add the block "ACS_MOD_WRITE_N_PRM".



Set function block inputs according to:

- EN is connected to a new variable "xExecuteWrite" of type "BOOL" (1). At rising edge the values are written once.
- PRIQ = Left unconnected.
- NVAR = Number of parameters to write (2) -> 2. Low and High Word of an 32bit parameter will be written in this example.
- PRM_NUM = Start address of parameter to be written (3) -> parameter 22.03 (Dec time1) will be written in this example. As parameter 22.03 is a 32bit parameter the Function "ACS_MOD_PRM_NUM_32BIT" (4) has to be used before the input PRM_NUM of the Function block.
If a normal 16bit parameter shall be written this function must not be used.
- DATA is connected via an "ADR" block to a new variable (5) of type "ARRAY [1..X] OF WORD" according to instructions for the "ACS_MOD_READ_N_PRM" block in the previous chapter.
- DRIVE_DATA = The same variable that is connected to "ACS_COM_MOD_TCP_ENHANCED" -> "DRIVE_DATA" (6).

To write the parameter the the following steps have to be performed:

Open the awWriteAccDecTimes array in the variable declaration part (1)

Double-click to the Low Word index of the array (2) and fill in a new value for the parameter (3)

Double-click on the variable xExecuteWrite to preset to TRUE. (4) and Press <CTRL + F7> (5) or "Online" -> "Write Values" to set the new values.

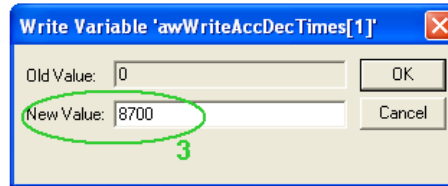
The new values are written to the variables, the write job is performed at the rising edge of the EN input.(6)
In the awAccDecTimes array the new value can be observed (7).

Quickstart Guide

ABB PLC and drives integration using Modbus TCP

```
0027 FB_READ_PARAMETERS
0028 xEnRead = TRUE
0029 awAccDecTimes
0030   awAccDecTimes[1] = 5000
0031   awAccDecTimes[2] = 0
0032   awAccDecTimes[3] = 5000
0033   awAccDecTimes[4] = 0
0034   awAccDecTimes[5] = 60000
0035   awAccDecTimes[6] = 0
0036   awAccDecTimes[7] = 60000
0037   awAccDecTimes[8] = 0
0038 FB_Write_PARAMETERS
0039 xExecuteWrite = FALSE
0040 awWriteAccDecTimes
0041   awWriteAccDecTimes[1] = 0
0042   awWriteAccDecTimes[2] = 0
0043   awWriteAccDecTimes[3] = 0
0044   awWriteAccDecTimes[4] = 0
0045   awWriteAccDecTimes[5] = 0
0046   awWriteAccDecTimes[6] = 0
0047   awWriteAccDecTimes[7] = 0
```

5: <Enter + F7>

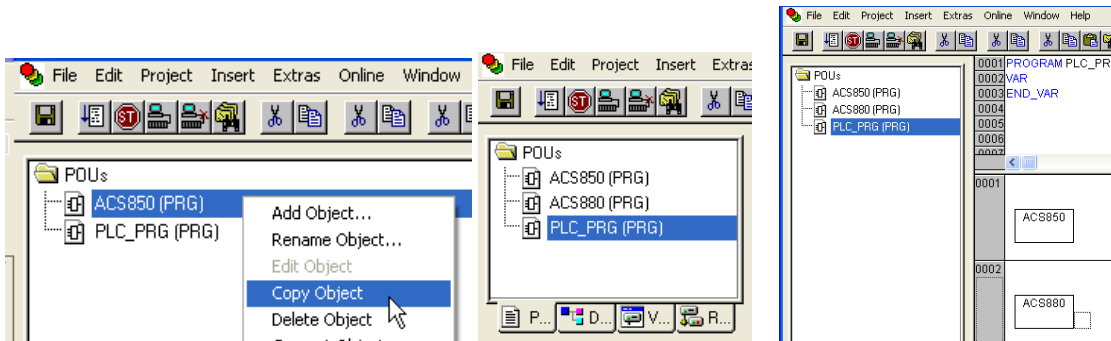


```
28 xEnRead = TRUE
29 awAccDecTimes
30   awAccDecTimes[1] = 5000
31   awAccDecTimes[2] = 0
32   awAccDecTimes[3] = 8700
33   awAccDecTimes[4] = 0
34   awAccDecTimes[5] = 60000
35   awAccDecTimes[6] = 0
36   awAccDecTimes[7] = 60000
37   awAccDecTimes[8] = 0
38 FB_Write_PARAMETERS
39 xExecuteWrite = TRUE
40 awWriteAccDecTimes
41   awWriteAccDecTimes[1] = 8700
42   awWriteAccDecTimes[2] = 0
43   awWriteAccDecTimes[3] = 0
```

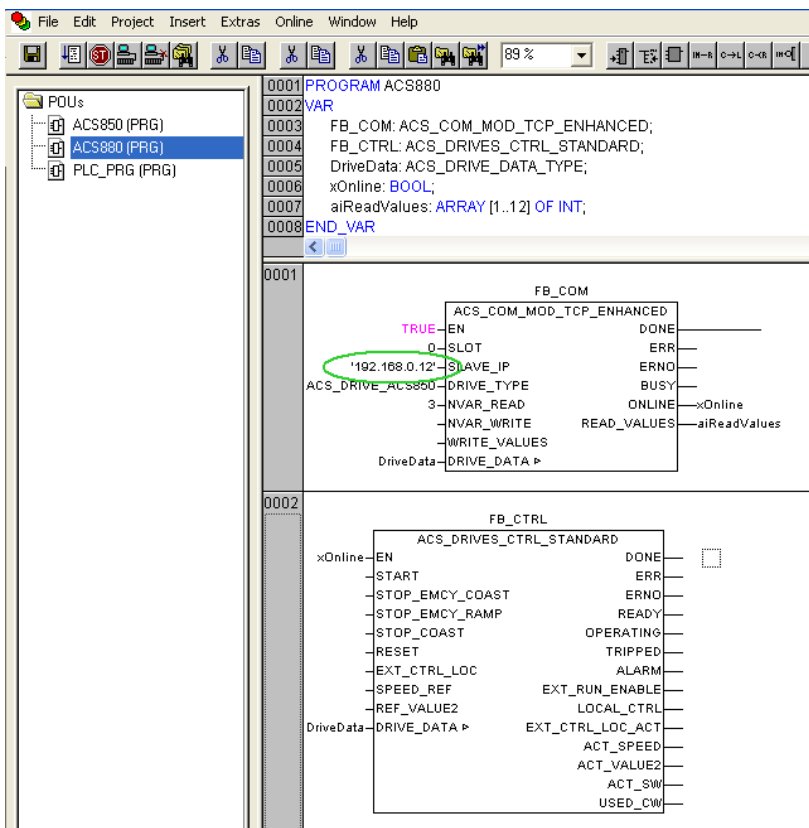
Note! Several “ACS_MOD_WRITE_N_PRM” blocks can be used in the program. The variable is only written at a rising edge of the EN input.

Add more drives (optional)

In the CODESYS “POUs” tab, right-click the program for your earlier drive and choose “Copy Object”, give the new program a suitable name and double-click “PLC_PRG”. Add the new program to the main program by copying Network 0001 to Network 0002 and rename the block in Network 0002 according to your new program.



Double-click and open the new program. Change “IP ADDRESS” input of the “ACS_COM_MOD_TCP_ENHANCED” block according to the IP ADDRESS of actual drive.



Continue to add as many drives as you need. If you want to create visualization pages/objects for the additional drives, make sure that you connect the placeholders to the function block instances for the correct program (PRG), “ACS880” in this example.

Other useful documentation

- CODESYS Help (Contents -> Target system -> AC500 / S500 -> ACS Drives Libraries)
- User's manual Ethernet Adapter FENA-01/11 [3UA0000093568]
- User's manual Ethernet Adapter RETA 01 [3AFE64539736]
- User's manual Ethernet Adapter RETA 02 [3AFE668895383]

- User's manual ACS355 drives [3AUA0000066143]
- ACS850 Firmware Manual Standard Control Program [3AUA0000045497]
- ACQ810 Firmware Manual Standard Pump Control Program [3AUA0000055144]
- Firmware manual ACS880 primary control program [3AUA0000085967]
- ACSM1 Firmware Manual Speed and Torque Control [3AFE68848270]
- ACSM1 Firmware Manual Motion Control [3AFE68848270]
- User's Manual ACS550-01 Drives [3AUA0000001418]
- Firmware Manual ACS800 Standard Control Program [3AFE64527592]

Revision

| Rev | Page (P) Chap (C) | Description | Date |
|-----|----------------------|--------------|------------|
| A | - | New document | 2013-05-16 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

For more information please contact your local ABB
representative or visit:
www.abb.com/drives
www.abb.com/drivespartners

© Copyright 2012 ABB. All rights reserved.
Specifications subject to change without notice.