OPTIONS FOR ABB DRIVES

FSCS-21 CIP Safety™ functions module

User's manual

FSCS-21 CIP Safety™ functions module

User's manual

3AXD50001065478 Rev A
EN
Original instructions
EFFECTIVE: 2024-05-28

Table of contents

14 EtherNet/IP – Communication profiles

15 EtherNet/IP – Communication protocol

16 CIP Safety

17 Verification and validation

18 Start-up

19 Fault tracing

20 Maintenance

23 Appendix B - Module configuration backup

Further information

1

Safety instructions

Contents of this chapter

This chapter contains the safety instructions which you must obey when you install, operate and do maintenance on the safety functions of a drive.

Safety messages

These safety messages help to prevent personal injury and damage to the equipment. The hazard levels comply with standard ANSI Z535.6.

The manual uses these warning symbols:

⚠DANGER Indicates a hazardous situation which, if not avoided, will result in death or serious injury.

⚠WARNING Indicates a hazardous situation which, if not avoided, could result in death or serious injury.

⚠CAUTION Indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.

NOTICE Is used to address practices not related to physical injury, but which can result in equipment damage.

Safety in installation and maintenance

⚠WARNING Obey the safety instructions of the drive. If you ignore them, injury or death to personnel, or damage to the equipment can occur.

Only a qualified electrical professional who has sufficient knowledge about functional, machine, and process safety is permitted to install, commission, and maintain the safety circuit.

This manual does not contain the complete safety instructions of the drive. It only includes the instructions related to the scope of this manual.

In addition to this manual:

- for ACS380, ACS580, ACS880 drives, refer to the drive hardware manual
- for ACS880 air-cooled multidrives, multidrive modules and single drive modules, refer to ACS880 multidrive cabinets and modules safety instructions (3AUA0000102301 [English])
- for ACS880 liquid-cooled multidrives, multidrive modules and single drive modules, refer to ACS880 liquid-cooled multidrive cabinets and modules safety instructions (3AXD50000048633 [English]).

2

Introduction to the manual

Contents of this chapter

This chapter describes the compatibility of the FSCS-21 module, gives a list of recommended reading and related manuals, and provides a link to product certificates.

Applicability

This manual is applicable to the FSCS-21 safety functions fieldbus module, revision B.

The revision of the module is shown on the type designation label. For more information, refer to section Type designation label (page 44).

Compatibility

Drives

The FSCS-21 safety fieldbus module is compatible with:

- ACS880 drives
  - ACS880 primary control program version 2.90 or later
- ACS580 drives
  - ACS580 standard control program version 2.07 or later
- ACS380 drives
  - ACS380 machinery control program version 2.07 or later

Tools

- Drive Composer: version 1.7 or later

Supported safety functions

This manual provides instructions for creating and using these safety functions (according to EN/IEC 61800-5-2):

- Safe torque off (STO)
- Safe stop 1 (SS1-t)

Target audience

The manual is intended for qualified persons who design the safety application, plan the installation as well as install and commission the safety application. Read the manual before starting work on the safety application. You must know the fundamentals of safety technology, electricity, wiring, electrical components and electrical schematic symbols.

Purpose of the manual

The manual explains how to install the FSCS-21 module and configure and commission the supported safety functions. It describes how to meet and maintain safety life cycle requirements of the FSCS-21 module to ensure required safety performance and specified safety integrity.

Recommended reading

This manual is based on the following standards. It is recommended that you are familiar with these standards before implementing safety-related systems.

- EN/IEC 61800-5-2: Adjustable speed electrical power drive systems – Part 5-2: Safety requirements – Functional. (Includes safety function definitions.)
- EN ISO 13849-1: Safety of machinery – Safety-related parts of control systems – Part 1: General principles for design
- EN/IEC 62061: Safety of machinery – Functional safety of safety-related electrical, electronic and programmable electronic control systems
- EN 60204-1: Safety of machinery – Electrical equipment of machines – Part 1: General requirements.
- IEC 61508: Functional safety of electrical/electronic/programmable electronic safety-related systems.
- The CIP Networks Library, Volume 5: CIP Safety. Edition: 2.25, April 2023. ODVA Inc.

Before starting the implementation of safety-related systems, it is highly recommended to read and understand the following manuals, which will also be referred to in the later chapters of this manual.

- Functional safety; Technical guide No. 10 (3AUA0000048753 [English])
- firmware and hardware manuals of the drive.

For a complete list of related standards and directives, refer to section Related standards and directives (page 228).

Exclusion of liability

ABB is not responsible for the implementation, verification and validation of the overall safety system. It is the responsibility of the system integrator (or other party) who is responsible for the overall system and system safety.

The system integrator (or other responsible party) must make sure that the entire implementation complies with the instructions in this manual, all relevant standards, directives and local electrical code, and that the system is tested, verified and validated correctly.

Cyber security disclaimer

This product is designed to be connected to and to communicate information and data via a network interface. It is Customer's sole responsibility to provide and continuously ensure a secure connection between the product and Customer network or any other network (as the case may be). Customer shall establish and maintain any appropriate measures (such as but not limited to the installation of firewalls, application of authentication measures, encryption of data, installation of anti-virus programs, etc.) to protect the product, the network, its system and the interface against any kind of security breaches, unauthorized access, interference, intrusion, leakage and/or theft of data or information.

ABB and its affiliates are not liable for damages and/or losses related to such security breaches, any unauthorized access, interference, intrusion, leakage and/or theft of data or information.

Terms and abbreviations

Term	Description
BCU	Type of control unit
Cat.	Classification of the safety-related parts of a control system in respect of their resistance to faults and their subsequent behavior in the fault condition, and which is achieved by the structural arrangement of the parts, fault detection and/or by their reliability. The categories are: B, 1, 2, 3 and 4. (EN ISO 13849-1)
CCF	Common cause failure (EN ISO 13849-1)

Term	Description
CIP	Common Industrial Protocol
Control unit	The part in which the control program runs.
Control word	16-bit or 32-bit word from a controller to the controlled device with bit-coded control signals (sometimes called the Command word).
DC	Diagnostic coverage (EN ISO 13849-1)
DI	Digital input
DLR	Device Level Ring. DLR network is a single-fault tolerant ring network topology intended for interconnection of automation devices.
EMC	Electromagnetic compatibility
FIT	Failure in time: 1E-9 hours (IEC 61508)
FSCS-21	Optional functional safety module
Inverter unit	Inverter module(s) under control of one control unit, and related components. One inverter unit typically controls one motor.
LLDP	Link Layer Discovery Protocol
LSB	Least significant bit
MAC address	Media Access Control address
MSB	Most significant bit
$MTTF_D$	Mean time to dangerous failure: (Total number of life units) / (Number of dangerous, undetected failures) during a particular measurement interval under stated conditions (EN ISO 13849-1)
Network control	With fieldbus protocols based on the Common Industrial Protocol (CIP™), such as DeviceNet and Ethernet/IP, denotes the control of the drive using the Control Supervisor and AC/DC drive objects of the ODVA AC/DC Drive Profile. For more information, see www.odva.org.
ODVA™	Open DeviceNet Vendor Association. ODVA is an independent organization that promotes interoperability between different manufacturers' EtherNet/IP products. ABB is an Associate Member at ODVA.
PFD_{avg}	Average probability of dangerous failure on demand (IEC 61508)
PFH	Average frequency of dangerous failures per hour (IEC 61508)
PL	Performance level. Levels a...e correspond to SIL (EN ISO 13849-1)
PLC	Programmable logic controller
Power drive systems (Safety-related), PDS (SR)	Adjustable speed electrical power drive system suitable for use in safety-related applications
Profile	Adaptation of a communication protocol for a certain application field (for example, drives)
Proof test	Periodic test performed to detect failures in a safety-related system so that, if necessary, a repair can restore the system to an “as new” condition or as close as practical to this condition. (IEC 61508, IEC 62061)
QoS	Quality of Service

Term	Description
Response time	The internal response time of the functional safety module, that is, the time in which the STO control output of the functional safety module reacts after receiving a request. (Usually this is not the same as the time from the request to the safe state of the machine application.) See also term Safety function response time (SFRT).
Safe state	STO is active. The drive STO circuit is open, modulation stops and motor coasts.
Safety module; functional safety module	Part of a safety system, a physical entity. For example: FSCS-21, FSPS-21, FSO-21.
Safety-related parameter	Parameters which have direct impact to the behavior of the safety function of the application, eg, trip limit parameter for a safely limited speed function. Unauthorized access to these parameters is prevented with a password.
SIL	Safety integrity level (1…3) (IEC 61508, IEC 62061, IEC 61800-5-2)
SS1	Safe stop 1 (IEC/EN 61800-5-2)
SS1-t	Safe stop 1, time controlled (IEC/EN 61800-5-2)
STO	Safe torque off (IEC/EN 61800-5-2)
Stop category	There are three categories of stop functions defined by IEC/EN 60204-1: • stop category 0: an uncontrolled stop where power to the machine actuators is removed immediately (for example, STO) • stop category 1: a controlled stop where the machine actuators have power for stopping, after which the power is removed (SS1) • stop category 2: a controlled stop where the machine actuators continue to have power (SS2).
Supply unit	Supply module(s) under control of one control unit, and related components.
T_1	Proof test interval. Defines the probabilistic failure rate (PFH or PFD_{avg}) for the safety function or subsystem. Performing a proof test at a maximum interval of T_1 is required to keep the SIL capability valid. The same interval must be followed to keep the PL capability (EN ISO 13849) valid. Note that any T_1 values given cannot be regarded as a guarantee or warranty.
Validation	Confirmation by, for example, analysis that the safety system meets the functional safety requirements of the specific application.
Verification	Confirmation by, for example, testing that the safety system meets the requirements set by the specification.
Zero speed	For safety functions, the zero speed limit indicates the completion of the safe stopping function.

CIP Safety terms and abbreviations

Term	Description
CFUNID	Configuration UNID

Term	Description
CPCRC	Connection Parameters CRC
CRC	Cyclic Redundancy Check
NodeID	Network-specific ID. The NodeID for EtherNet/IP is the TCP/IP address in hexadecimal format.
OCPUNID	Output Connection Point Owner UNID
OUNID	Originator UNID
SCCRC	Safety Configuration CRC
SCID	Safety Configuration ID
SCTS	Safety Configuration Timestamp
SNN	Safety Network Number. Uniquely identifies a network across all networks in the safety system.
TUNID	Target UNID
UCMM	Unconnected Message Manager
UNID	Unique Identifier. Consists of SNN and NodeID.

Certificates

The valid EtherNet/IP + CIP Safety and functional safety certificates are available in ABB Library (www.abb.com/drives/documents).

Related manuals

Name	Code
Drive hardware manuals	
ACS880-01 hardware manual	3AUA0000078093
ACS880-04 hardware manual	3AUA0000128301
ACS880-04 single drive module packages hardware manual	3AUA0000138495
ACS880-11 hardware manual	3AXD50000045932
ACS880-31 hardware manual	3AXD50000045933
ACS880-14 and -34 single drive module packages hardware manual	3AXD50000022021
ACS880-04XT drive modules (500 to 1200 kW) hardware manual	3AXD50000025169
ACS880-04FXT drive module packages hardware manual	3AXD50000274444
ACS880-07 (45 to 630 kW) hardware manual	3AUA0000105718
ACS880-07 (560 to 2800 kW) hardware manual	3AUA0000143261

Name	Code
ACS880-17 (160 to 3200 kW) hardware manual	3AXD50000020436
ACS880-37 (160 to 3200 kW) hardware manual	3AXD50000020437
ACS880-17 (132 to 355 kW) hardware manual	3AXD50000035158
ACS880-37 (132 to 355 kW) hardware manual	3AXD50000035159
ACS880-104 inverter modules hardware manual	3AUA0000104271
ACS880-107 inverter units hardware manual	3AUA0000102519
ACS880-07LC drives hardware manual	3AXD50000569786
ACS880-17LC drives hardware manual	3AXD50000250295
ACS880-37LC drives hardware manual	3AXD50000251407
ACS880-104LC inverter modules hardware manual	3AXD50000045610
ACS880-107LC inverter units hardware manual	3AXD50000196111
ACS580-01 drives (0.75 to 250 kW, 1.0 to 350 hp) hardware manual, frames R1-R9	3AXD50000044794
ACS580-04 hardware manual	3AXD50000015497
ACS580-07 drives (75 to 250 kW) hardware manual	3AXD50000045815
ACS580-07 (250 to 500 kW) hardware manual	3AXD50000032622
ACS380 hardware manual	3AXD50000029274
Drive firmware manuals	
ACS880 primary control program firmware manual	3AUA0000085967
ACS580 standard control program firmware manual	3AXD50000016097
ACS380 machinery control program firmware manual	3AXD50000029275
Option manuals	
ACS-AP-I, -S, -W and ACH-AP-H, -W Assistant control panels user's manual	3AUA0000085685
ACS-BP-S Basic control panel user's manual	3AXD50000032527
Drive PC tool manual	
Drive Composer start-up and maintenance PC tool user's manual	3AUA0000094606
General safety guides	
Functional safety; Technical guide No. 10	3AUA0000048753

You can find manuals on the Internet. Visit www.abb.com/drives/documents. For manuals not available in ABB Library, contact your local ABB representative.

For additional ABB safety information and solutions, visit http://www.abb.com/safety.

3

Safety system information and considerations

Contents of this chapter

This chapter contains general safety considerations and information you must take into account when you apply the safety functions.

Applying FSCS-21 safety functions

To make sure that the application where the FSCS-21 module and its safety functions are used is safe and fulfills all necessary safety requirements, it is necessary to take into account the requirements of the local (machinery) safety legislation (for example, Machinery Directive) and/or the applicable functional safety standards (for example, ISO 13849-1).

To make safe applications and implementing safety functions requires a systematic approach / process, where risk assessment is the basis of all safety requirements. These processes are described in global ISO and IEC machinery safety and/or functional safety standards (for example, ISO 12100, ISO 13849-1, IEC 62061). These are also introduced in Functional safety; Technical guide No. 10 (3AUA0000048753 [English]).

Before you take into use a system where the FSCS-21 module with its safety functions is used, and also every time you modify application parameters or the configuration related to safety, you must verify and validate the entire system according to the system safety verification plan. Refer to chapter Verification and validation (page 197).

Meeting the requirements of the Machinery Directive and UK Supply of Machinery (Safety) Regulations

If the machine, where the FSCS-21 module is part of a safety system, will be sold or taken into use in Europe, it is the responsibility of the machine builder / OEM / system integrator to make sure that the machine is safe to use and all applicable essential health and safety requirements (EHSR) of the Machinery Directive and/or UK Supply of Machinery (Safety) Regulations are fulfilled. The requirements in the applicable standards must also be met and the safety module must be used according to all instructions provided in this manual.

Responsibilities

ABB as a component manufacturer is only responsible for the safety and conformity of their manufactured products, not of the systems where these products are used.

If you detect a failure in safety functions, contact your local ABB representative.

Intentional misuse

Use the safety module according to the instructions given in the user's manual. ABB is not responsible for damage caused by the misuse of the module.

The module is not designed to protect a machine against intentional misuse or sabotage.

Safety-related parts

The safety module and the drive Safe torque off (STO) channel/function are safety relevant, and the rest of the drive is considered as not safety relevant, for example, the drive regular I/O cannot be used for requesting safety functions on the safety module.

Limitations of Safe torque off (STO) function

The Safe torque off function can be used for stopping the drive in the operational mode. When activated, the Safe torque off function disables the control voltage for the power semiconductors of the drive output stage, thus preventing the drive from generating the torque required to rotate the motor. If the motor is running when Safe torque off is activated, it coasts to a stop. The Safe torque off function can also be used for prevention of unexpected start-up according to the limitations of IEC 60204-1, 5.4 and ISO 14118.

⚠WARNING Do not use the Safe torque off function to do maintenance work on the electrical parts of the system (including drive and motor). The Safe torque off function does not disconnect the voltage of the main and auxiliary circuits from the drive.

If the motor shaft has external forces acting on it (for example, a hanging load), you must take this into account when you plan the safety system. The Safe torque off function does not prevent the motor shaft from turning.

For more information on the drive Safe torque off function, refer to the drive hardware manual.

Overall system stopping performance

A safety function always consists of other components in addition to FSCS-21. The overall safe stopping time of the entire system must meet the requirements of the risk assessment. This means that the dangerous movement of the machine must be stopped before a dangerous situation can occur.

Safety function response time and the FSCS-21 module response times are specified in section Response times (page 224).

4

Overview of the FSCS-21 module and the Ethernet network

Contents of this chapter

This chapter contains on overview of the Ethernet network and the FSCS-21 module.

Module overview

The FSCS-21 is an option module for ABB drives. The compatible drive types are given in section Compatibility (page 17). With the module, you can connect a drive to an Ethernet network and a safety PLC. The module supports the CIP Safety communication technology through the EtherNet/IP communication protocol.

The intended use of the FSCS-21 safety option module is to safely stop the drive when necessary. A safety PLC can activate the safety functions of the module. The module supports these safety functions:

- **Safe torque off (STO)**: The safety function activates the drive STO function. The motor coasts to a stop (stop category 0).
- **Safe stop 1 (SS1-t)**: The safety function decelerates the motor to zero speed (stop category 1). When the drive indicates that zero speed is reached, the safety function activates the drive STO function. If zero speed is not reached within the SS1-t monitored time limit, the safety function activates the drive STO function.

The module is installed into an option slot on the drive control unit. Refer to the applicable drive hardware manual for module placement options.

Through the FSCS-21 module you can:

- start and stop the drive and reset drive faults
- give speed or torque reference to the drive
- read drive status word bits
- write to drive control word bits
- activate safety functions.

The protocol used to access these functionalities over the Ethernet is:

- EtherNet/IP – Communication protocol
- CIP Safety over EtherNet/IP.

The module supports 10 Mbit/s and 100 Mbit/s Full and Half duplex data transfer rates and automatically detects the data transfer rate used in the network.

The configuration of the FSCS-21 safety functions module does not include safety-related parameters. For other parameters, refer to section Drive parameters related to FSCS-21 module configuration (ACS380, ACS580, and ACS880) (page 60).

Diagnostics

During power-up and operation, the FSCS-21 module does automatic diagnostics tests on:

- internal components
- power and signal connections between the drive and the module
- STO connection between the drive and the module.

Fault reaction function

The module activates STO and goes into Fail-safe mode if it detects a fault in:

- STO connection between the FSCS-21 module and the drive STO connector
- internal parts of the module (for example, CPU or memory).

The module activates STO and goes into the Passivation state if it detects a fault in the CIP Safety communication between the module and the safety PLC.

The fault reaction function activates the Fail-safe mode.

Automatic acknowledgement

FSCS-21 only has an automatic acknowledgement method for a safety function. This means that when the safety function request is removed, the FSCS-21

automatically enters the Operational state. Automatic acknowledgement also takes place during module start-up.

Manual acknowledgement can only be programmed into a safety PLC. For more information on FSCS-21 states, see section States (page 41).

⚠WARNING FSCS-21 module has automatic acknowledgement method. Make sure that the system is designed so that this does not cause unacceptable risk, for example, due to an automatic start of the drive.

Ethernet network

Ethernet standards support a variety of physical media (coaxial cable, twisted pair, fiber optics) and topologies (bus and star).

The FSCS-21 module supports:

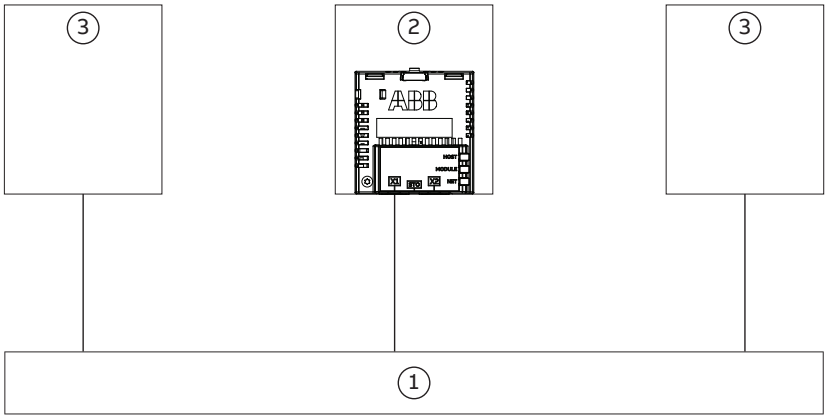
- twisted pair as the physical media
- star topology, daisy chain topology and ring topology (DLR).

The maximum length for an Ethernet segment on twisted pair media is 100 m (328 ft). All twisted pair media between the Ethernet node and the switch or router must be shorter than 100 m (328 ft), including media within patch panels.

Example topology of the Ethernet link

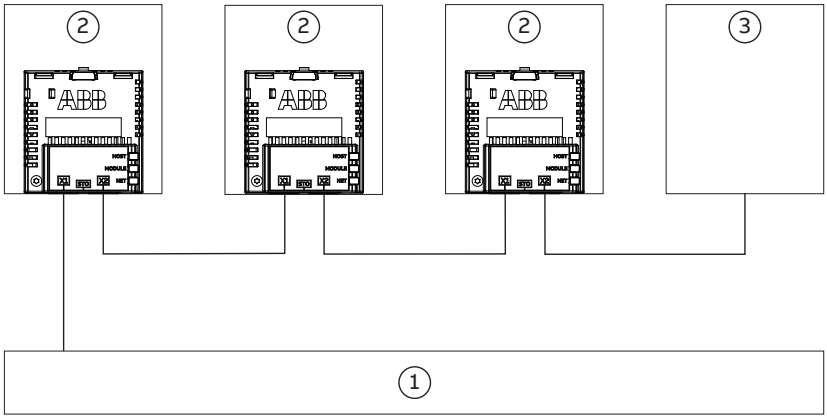
The figures below show example topologies for an Ethernet network with FSCS-21 module.

Star topology



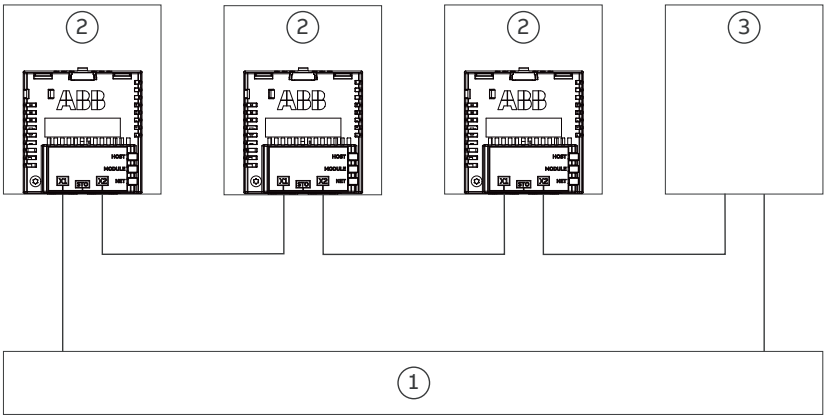
1	Switch or router
2	ABB drive with FSCS-21 module
3	Other device

Daisy chain topology



1	Switch or router
2	ABB drive with FSCS-21 module
3	Other device

Ring topology (DLR)



1	Active Ring Supervisor
2	ABB drive with FSCS-21 module
3	Other device

5

Safety functions

Contents of this chapter

This chapter gives information on the safety functions that the FSCS-21 module supports.

Safety functions

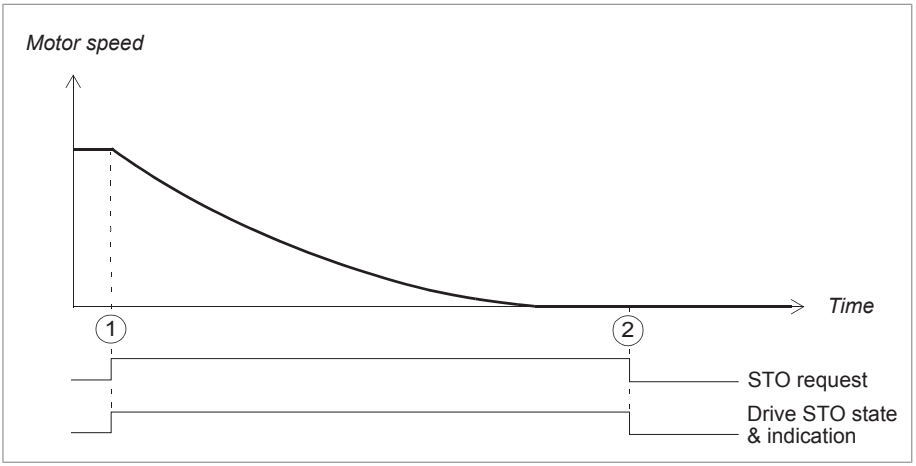
- **STO**

The STO function brings the machine safely into a no-torque state. When the STO request is active, it prevents the machine from starting accidentally. This functionality can be used as prevention of unexpected start-up. The STO function in the FSCS-21 module activates the drive STO function, that is, opens the STO circuit in the drive. This prevents the drive from generating the torque required to rotate the motor. If the motor is running when the STO function is activated, and there are no external forces (for example, hanging loads), it coasts to a stop.

Note: If the motor shaft has external forces acting on it, consider using Safe Brake Control (SBC) for stopping the motor. For more information, visit https://new.abb.com/drives/functional-safety

For more information on the STO function in the drive, refer to the drive hardware manual.

The operation of the STO function is described in the time diagram and table below.



Step	Description
1	The STO request is received from the safety PLC. The module activates the drive STO function. The STO indications in the drive and safety PLC go on.
2	The STO request is removed from the safety PLC. The STO function is deactivated and the indications go off.

SS1 with time monitoring

The SS1-t function stops the motor safely by ramping down the motor speed. The FSCS-21 module activates the drive STO function when the zero speed limit defined by parameter 21.06 Zero speed limit is reached.

The SS1 function uses drive parameter 23.23 Emergency stop ramp to define the stop ramp. For ACS880, ACS580 and ACS380, see section Drive parameters related to FSCS-21 module configuration (ACS380, ACS580, and ACS880) (page 60) for more information.

The FSCS-21 module monitors the stop ramp with a time limit. If the motor speed does not reach the user-defined zero speed limit within the specified time limit, the module activates the drive STO function, the motor coasts to a stop, and FSCS-21 creates a fault (0x7AA0).

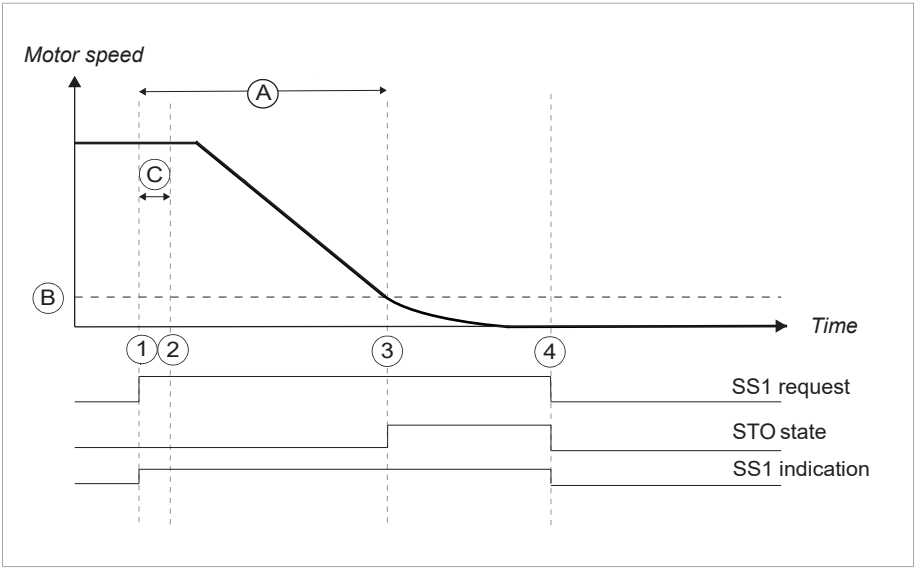
The SS1 time limit 0.0…3276.7 s is set by the safety PLC in PS3 telegram SS1-t_time variable.

The operation of the SS1 with time monitoring (SS1-t) is described in the timing diagrams below.

Note: The controlled stop of SS1-t can fail undetected, therefore SS1-t cannot be applied if this failure can cause a dangerous situation in the final application.

Timing diagrams

SS1-t function normal operation

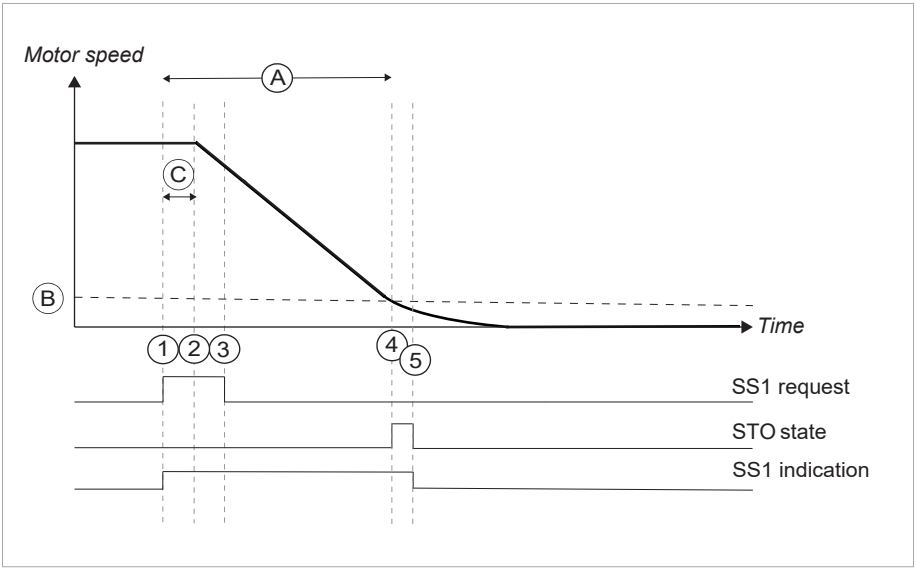


A	SS1-t delay for STO: Time after which the module activates the drive STO function regardless of the motor speed. The value is received from the safety PLC in each SS1-t activation request.
B	Zero speed (drive parameter 21.06): Speed limit for activating the drive STO function.
C	Safety function response time (depends on system configuration). For more information see chapter Technical data.

Step	Description
1	The SS1-t request is received from the safety PLC. The module starts a counter for time A. The SS1-t indication (Off3) in the drive and safety PLC goes on.
2	After time C has elapsed, the drive starts to ramp down the motor speed. Drive parameters define the deceleration ramp (emergency stop ramp).
3	The motor speed reaches the user-defined zero speed limit (B) and the module activates the drive STO function.

Step	Description
4	The SS1-t request is removed from the safety PLC. The STO and SS1 functions are automatically acknowledged and the indications go off.

SS1-t function, pulse request



A — SS1-t delay for STO: Time after which the module activates the drive STO function regardless of the motor speed. The value is received from the safety PLC in each SS1-t activation request.

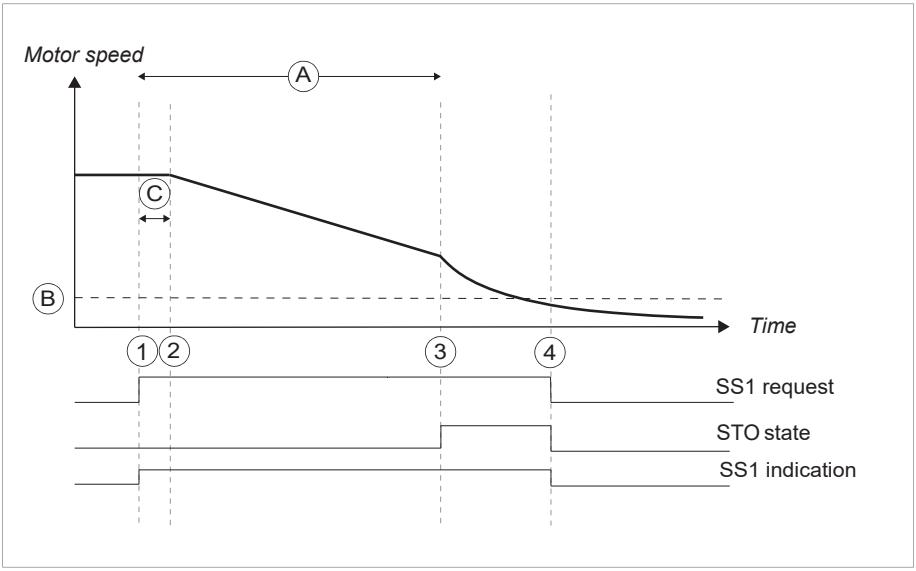
B — Zero speed (drive parameter 21.06): Speed limit for activating the drive STO function.

C — Safety function response time (depends on system configuration). For more information see chapter Technical data.

Step	Description
1	The SS1-t request is received from the safety PLC. The module starts a counter for time A. The SS1-t indication (Off3) in the drive and safety PLC goes on.
2	After time C has elapsed, the drive starts to ramp down the motor speed. Drive parameters define the deceleration ramp (emergency stop ramp).
3	The SS1-t request is removed from the safety PLC.

Step	Description
4	The motor speed reaches the user-defined zero speed limit (B) and the module activates the drive STO function.
5	The STO indications quickly go on and off, and SS1 indication is removed. **Note:** The time STO state was active might not be seen on PLC.

SS1-t function, time delay passed



A SS1-t delay for STO: Time after which the module activates the drive STO function regardless of the motor speed. The value is received from the safety PLC in each SS1-t activation request.

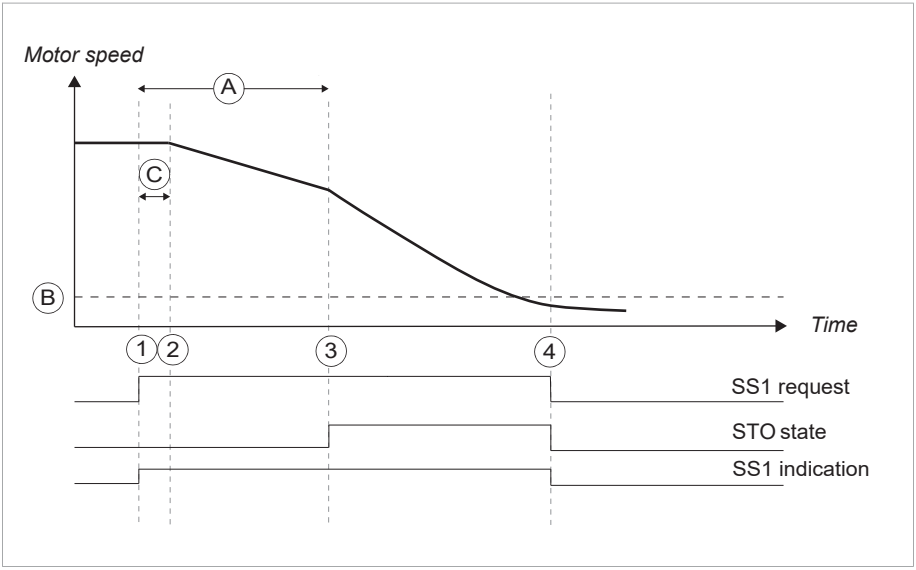
B Zero speed (drive parameter 21.06): Speed limit for activating the drive STO function.

C Safety function response time (depends on system configuration). For more information see chapter Technical data.

Step	Description
1	The SS1-t request is received from the safety PLC. The module starts a counter for time A. The SS1-t indication (Off3) in the drive and safety PLC goes on.
2	After time C has elapsed, the drive starts to ramp down the motor speed. Drive parameters define the deceleration ramp (emergency stop ramp).

Step	Description
3	The defined SS1-t delay time has elapsed and the module activates the drive STO function.
4	The SS1-t request is removed from the safety PLC. The STO and SS1 functions are automatically acknowledged and the indications go off.

SS1-t function drive modulation stopped



A SS1-t delay for STO: Time after which the module activates the drive STO function regardless of the motor speed. The value is received from the safety PLC in each SS1-t activation request.

B Zero speed (drive parameter 21.06): Speed limit for activating the drive STO function.

C Safety function response time (depends on system configuration). For more information see chapter Technical data.

Step	Description
1	The SS1-t request is received from the safety PLC. The module starts a counter for time A. The SS1-t indication (Off3) in the drive and safety PLC goes on.
2	After time C has elapsed, the drive starts to ramp down the motor speed. Drive parameters define the deceleration ramp (emergency stop ramp).

Step	Description
3	The modulation of the drive is stopped during ramp down (Eg. Drive enable signal removed, stop request from panel or from ATEX module, etc.) and the module activates the drive STO function.
4	The SS1-t request is removed from the safety PLC. The STO and SS1 functions are automatically acknowledged and the indications go off.

Priorities of safety functions

STO function will always override SS1-t function, should they occur within the same time frame.

Acknowledgement methods

The safety functions have one acknowledgement method for entering the Operational state (during start-up, or after a safety function request is removed):

- **Automatic**: The module grants the drive permission to restart after a safety function request is removed or the start-up is complete.

Note: In addition, you can create a manual acknowledgement method in the safety PLC program.

⚠WARNING FSCS-21 module only has the automatic acknowledgement method. Make sure that the drive system does not start accidentally.

States and modes

States

When the FSCS-21 module is up and running, it can be in one of the following states depending on the drive STO status:

- **Safe**: STO active, that is, the drive STO circuit is open, modulation stops, and motor coasts.
- **Operational**: STO inactive.

In the Operational and Safe states, the FSCS-21 module can execute the safety functions.

Modes

The FSCS-21 can be in one of the following modes:

- **Power down**: The power to the FSCS-21 is off. The drive STO circuit is open.

- **Start-up**: The FSCS-21 is starting up after power-up.
- **Run**: The FSCS-21 is up and running. It can be in different states (see previous section) depending on the status of safety functions and the safety fieldbus communication.
- **Fault-detected**: The FSCS-21 has detected a failure, drive STO is activated, and FSCS-21 enters the Fail-safe mode. It can be in different states (see previous section) depending on the status of safety functions and the safety fieldbus communication.
- **Fail-safe**: There is a failure in the FSCS-21 and Safe state has been reached. The drive STO is active. You must reboot the FSCS-21 to exit the Fail-safe mode.

Safety functions indications

You can see the status of safety functions from the CIP Safety message. Only CIP Safety information can be used for functional safety applications.

For the structure of the CIP Safety message, refer to section CIP Safety description (page 183).

In addition, the drive indicates the drive STO status according to parameter 31.22 STO indication run/stop. For more information, see the firmware manual.

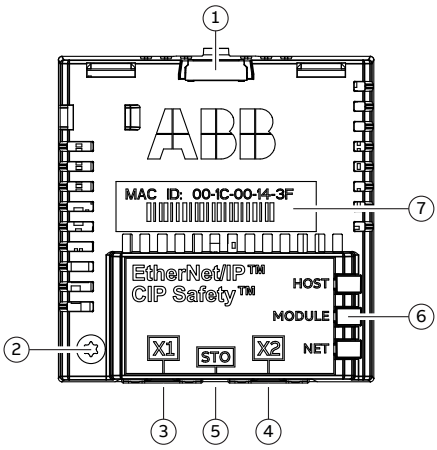
6

Hardware description

Contents of this chapter

This chapter gives a short description of the module.

Module layout



No.	Description
1	Lock
2	Mounting and grounding screw
3	RJ-45 connector [X1] to Ethernet
4	RJ-45 connector [X2] to Ethernet
5	STO connector (under connectors [X1] and [X2])
6	Diagnostic LEDs
7	MAC ID

Type designation label

1	Type
2	Serial number of format RYWWSSSSWS, where R: component revision; A, B, … Y: Last digit of the manufacturing year: 2, 3, … for 2022, 2023 WW: Manufacturing week: 01, 02, … for week 1, week 2, … SSSS: Integer starting every week from 0001 WS: Manufacturing location
3	ABB MRP code of the FSCS-21 module
4	Combined ABB MRP code, component revision, serial number and manufacturing location
5	Software version
6	RoHS mark
7	CE mark

7

Planning for installation

Contents of this chapter

This chapter gives instructions and references to instructions in other manuals for planning the safety system installation, as well as the requirements for installation in the applicable safety standards.

Requirements for designers and installers

- Designers and installers must be trained to understand the requirements and principles of designing and installing safety-related systems.
- Designers and maintainers must be trained to understand the causes and consequences of Common Cause Failures (CCF).

Installation site

The subsystem elements must always be likely to operate within the range of temperature, humidity, corrosion, dust, vibration, etc. for which they are specified, without the use of external environmental control.

Use the FSCS-21 module only in an environment where there are no conductive dust or contaminants. If necessary, use the module inside an enclosure with sufficient protection (for example, an IP54 enclosure). For more information on environmental limits, refer to section Ambient conditions (page 223) and the drive hardware manual.

⚠WARNING If you operate the drive system with a safety module in environmental conditions that are outside of the specified ranges for the safety module, loss or malfunction of a safety function can occur.

Electrical installation

General requirements

Obey these general requirements for the electrical installation:

- Do the installation of the safety system according to the requirements given in the drive hardware manual.
- Make sure that all cables are sufficiently protected, correctly routed, and attached where necessary.
- Make sure that there is no strain on the cables.

Ensuring the EMC compatibility

The system must only be used in the EMC environment it is designed for, or necessary mitigations must be applied.

8

Mechanical installation

Contents of this chapter

This chapter contains a delivery checklist and instructions on installing the module.

Necessary tools and instructions

Use a Torx TX10 screwdriver to attach the FSCS-21 module to the drive. For more information, refer to the drive hardware manual.

If you have an ACS380 drive and order FSCS-21 separately as a kit, you must replace the ACS380 standard cover with the optional deeper cover. The ordering code for the deeper cover is 3AXD50000190188.

Unpacking and examining the delivery

1. Open the option package.
2. Make sure that the package contains:
   - fieldbus module, type FSCS-21 (1)
   - STO cable (with an attached connector for ACS880 series drives) (2)
   - quick guide (not shown in the figure).
3. Make sure that there are no signs of damage.

Before you start

Plastic pins, a lock and one screw hold the module in place. The screw also makes an electrical connection between the FSCS-21 module and drive frame for cable shield grounding.

Drives with separate supply and inverter units: You can only install the FSCS-21 module to an inverter unit.

When you install the FSCS-21 module, it makes the signal and power connection to the drive through a 20-pin connector.

Installing option modules

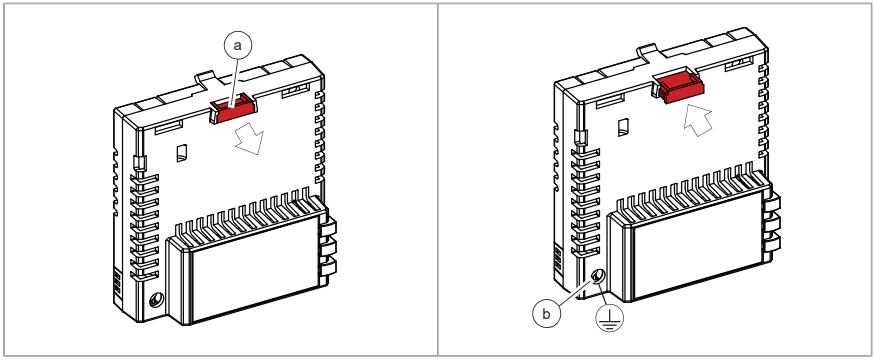
⚠WARNING Obey the safety instructions of the drive. If you ignore them, injury or death to personnel, or damage to the equipment can occur. If you are not a qualified electrical professional, do not do electrical installation or maintenance work.

Pay attention to the free space required by the cabling and terminals that connect to the option modules.

1. Stop the drive and do the steps in section **Electrical safety precautions** of the drive hardware manual.
2. Get access to the drive control unit. For the location of the control unit, refer to the drive hardware manual.
3. Pull out the lock on the module (a).
4. Install the module in a free option module slot on the control unit.
5. Push in the lock on the module (a).
6. Torque the grounding screw (b) to 0.8 N·m (7 lbf·in).

   Note: The screw tightens the connections and grounds the module. It is necessary for fulfilling the EMC requirements and for correct operation of the module.

   NOTICE Do not tighten the screw too much, or leave it loose. If you tighten the screw too much, damage to the threads or module can occur. A loose screw can cause a malfunction.



7. Connect the wiring to the module. Obey the instructions in this manual.

If you must remove the adapter module after it was installed into the drive, use a suitable tool (for example, small pliers) to carefully pull out the lock.

9

Electrical installation

Contents of this chapter

This chapter contains general cabling instructions and instructions on how to connect the FSCS-21 module to the Ethernet network and the drive.

Necessary tools and instructions

Refer to the drive hardware manual.

General cabling instructions

- Arrange the bus cables as far away from the motor cables as possible.
- Avoid parallel runs.
- Use grommets at cable entries.

Connecting the FSCS-21 module to the network

⚠WARNING Obey the safety instructions of the drive. If you ignore them, injury or death to personnel, or damage to the equipment can occur.

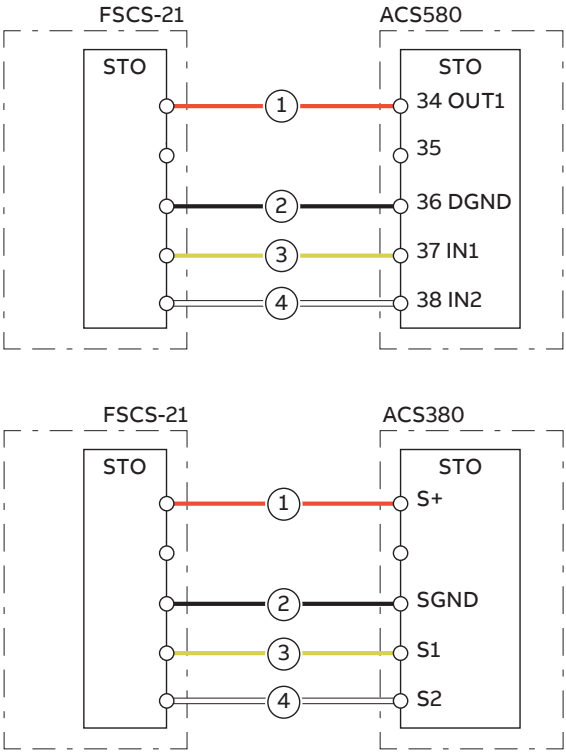
Select the correct cable for the application. CAT5e and CAT6 cables are recommended for industrial applications. ABB recommends to use double-shielded twisted pair cable, for example, "SF/FTP".

The cable shield is connected to the drive frame through an RC network.

For information on the module connectors, refer to section Module layout (page 43).

Connect the cables to the module as follows:

1. Connect the network cable to connector [X1] on the module.
2. If you want to create a daisy chain network with the modules, connect connector [X2] of the first module to connector [X1] on the next module, and so on.
3. Connect the STO cable to the module (black connector).
4. **ACS880:** Connect the yellow connector to the drive STO terminals.
   ACS380/580: Remove the yellow connector from the cable and connect the conductors to the drive STO terminals. Refer to the diagram that follows.



1	Red
2	Black
3	Yellow

4	White

Note: It is not permitted to install devices between the FSCS-21 module and the drive STO connector, unless otherwise guided. See safety related technical instructions in https://new.abb.com/drives/functional-safety.

Note: It is not permitted to use an external power supply for the STO circuit with the FSCS-21 module.

10

Installation checklists

Contents of this chapter

This chapter contains a checklist of the mechanical and electrical installation of the FSCS-21 module and refers to common cause failure checklists in standards.

Installation checklist

Use this checklist to make sure that the mechanical and electrical installation of the module is completed. Go through the checklist together with another person.

Make sure that ...	☑
The ambient operating conditions meet the requirements.	☐
Tools, debris, and unwanted materials are removed from the installation area.	☐
The module is correctly attached to the control unit.	☐
The drive and the module are correctly grounded to the same potential.	☐
The network cable is correctly connected.	☐
The STO cable is correctly installed.	☐

Common cause failure (CCF) checklists

Check measures against common cause failures (CCF). You can use, for example, the checklists given in EN ISO 13849-1 or EN/IEC 62061.

11

Commissioning

Contents of this chapter

This chapter gives information on commissioning the module.

Safety

Make sure that the safety of the machine is maintained during the commissioning.

Until all the safety functionality is validated, the system must not be considered safe.

Commissioning procedure

Commission the FSCS-21 module as follows:

1. Commission, configure and do the necessary ID runs for the drive. Refer to the drive hardware and firmware manuals.
2. Install the FSCS-21 module according to the instructions in this manual. Refer to chapters Planning for installation, Mechanical installation, and Electrical installation.
3. Configure EtherNet/IP and CIP Safety according to the application requirements. Refer to chapters Configuration and EtherNet/IP – Start-up.
4. Create a safety PLC project with CIP Safety communication to the drive. Refer to chapter CIP Safety.
5. Validate the safety functions. Refer to chapter Verification and validation.

Note: You can disable the FSCS-21 module with parameter 50.01.

12

Configuration

Contents of this chapter

This chapter lists the parameters related to the safety functions and gives configuration instructions.

Competence

The person who configures the safety functions must be a competent person as required by IEC 61508-1 clause 6. In this context, the person must have adequate expertise and knowledge of functional safety, the safety functions as well as the configuration of the module.

Tools

A control panel or the Drive Composer PC tool is necessary to configure the FSCS-21 module. For more information on the Drive Composer PC tool, see Drive composer start-up and maintenance PC tool user's manual (3AUA0000094606 [English]).

Drive parameters related to FSCS-21 module configuration (ACS380, ACS580, and ACS880)

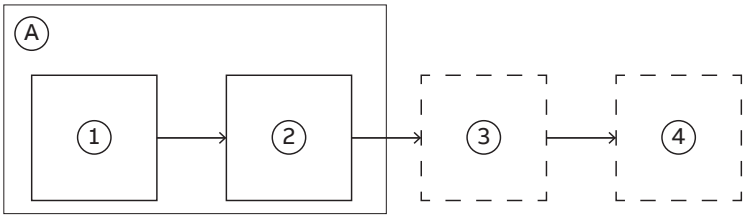
This table lists the drive parameters related to the FSCS-21 module and the safety functions for ACS380, ACS580, and ACS880 drives. For more information, see the firmware manual.

Index	Name	Description
21.06	Zero speed limit	Sets the zero speed limit for the drive. The drive will stop modulating when the zero speed limit is reached. Relevant for the SS1-t function.
21.07	Zero speed delay	Sets the delay for the zero speed delay function. The function is useful in applications where a smooth and quick restarting is essential. During the delay, the drive knows the rotor position accurately. Relevant for the SS1-t function.
23.23	Emergency stop time	In speed control mode, this parameter defines the deceleration rate for emergency stop Off3 as the time it would take for the speed to decrease from the value of parameter 46.01 Speed scaling to zero. This also applies to torque control because the drive switches to speed control on receiving an emergency stop Off3 command. In frequency control mode, this parameter specifies the time it would take for the frequency to decrease from the value of 46.02 Frequency scaling to zero. **Note:** With SS1 commanded from FSCS-21, emergency stop mode is always set as Ramp. **Note:** Parameters 21.04 Emergency stop mode and 21.05 Emergency stop source have no effect to the safety functions of the FSCS-21 module.
25.15	Proportional gain em stop	Defines the proportional gain for the speed controller when an emergency stop is active.

Index	Name	Description
31.22	STO indication run/stop	Selects which indications are given when one or both Safe torque off (STO) signals are switched off or lost. The indications also depend on whether the drive is running or stopped when this occurs. The tables at each selection below show the indications generated with that particular setting. **Note:** This parameter does not have an effect on the operation of the STO function.

Configuring the FSCS-21

This illustration shows an overview of the configuration process.



A	Configuration
1	Plan configuration
2	Configure
3	Do commissioning tests
4	Prepare the commissioning report

To configure the FSCS-21, do the steps shown below.

1. Plan the configuration according to the application requirements.
2. Set the drive parameters related to EtherNet/IP and CIP Safety communication. Refer to section EtherNet/IP connection configuration (page 63).
3. Set the drive parameters related to the safety functions. Refer to section Drive parameters related to FSCS-21 module configuration (ACS380, ACS580, and ACS880) (page 60).
4. Set up the EtherNet/IP communication. Refer to sections Starting up fieldbus communication for the drive (page 78) and Configuring the client (page 81).

5. Configure the safety PLC and set up CIP Safety communication between the module and the safety PLC. Refer to section Configuration (page 186).
6. Do the commissioning tests. Refer to chapter Verification and validation.

13

EtherNet/IP – Start-up

Contents of this chapter

This chapter contains:

- information on configuring the drive for operation with the adapter module
- drive-specific instructions on starting up the drive with the adapter module
- examples of configuring the client for communication with the adapter module.

Warnings

⚠WARNING Obey the safety instructions given in this manual and the drive documentation.

Drive configuration

The information in this section applies to all drive types compatible with the adapter module, unless otherwise stated.

EtherNet/IP connection configuration

After the adapter module has been mechanically and electrically installed according to the instructions in chapters Mechanical installation and Electrical installation, you must prepare the drive for communication with the module.

Normally, you must adjust a parameter to activate the communication. For detailed procedure of activating the module for EtherNet/IP communication with the drive, refer to section Starting up fieldbus communication for the drive (page 78).

Once communication between the drive and the adapter module has been established, several configuration parameters are copied to the drive. These parameters are shown in the tables below and must be checked first and adjusted where necessary. You can adjust the parameters via a drive control panel or a PC tool.

Note: The new parameter settings take effect only when you power up the module the next time or when you activate the fieldbus adapter refresh parameter.

FSCS-21 configuration parameters – group A (group 1)

Note: The actual parameter group number depends on the drive type. Group A (group 1) corresponds to:

- parameter group 51 in ACS380 and ACS580
- parameter group 51/54 (or 151/154 in some variants) in ACS880 if the adapter is installed as fieldbus adapter A/B.

No.	Name/ Value	Description	Default
01	FBA type	Read-only. Shows the fieldbus adapter type as detected by the drive. The value cannot be adjusted by the user. If the value is 0 = None, the communication between the drive and the module has not been established.	2222 = EtherNet/IP
02	Protocol/Profile	Selects the application protocol and communication profile for the network connection. The selections available for EtherNet/IP communication are listed below.	101 = EIP ABB Pro
	100 = EIP AC/DC	EtherNet/IP protocol: ODVA AC/DC drive profile	
	101 = EIP ABB Pro	EtherNet/IP protocol: ABB Drives profile	
	102 = EIP T16	EtherNet/IP protocol: Transparent 16-bit profile	
	103 = EIP T32	EtherNet/IP protocol: Transparent 32-bit profile	
03	Commrate	Sets the bit rate for the Ethernet interface. In FSCS-21 this parameter is used for configuring port 1. For configuring port 2, see parameter 14 Commrate port 2 (page 67).	0 = Auto
	0 = Auto	Autonegotiate	

No.	Name/ Value	Description	Default
	1 = 100 Mbps FD	100 Mbps, full duplex	
	2 = 100 Mbps HD	100 Mbps, half duplex	
	3 = 10 Mbps FD	10 Mbps, full duplex	
	4 = 10 Mbps HD	10 Mbps, half duplex	
04	IP configuration	Sets the method for configuring the IP address, subnet mask and gateway address for the adapter module.	1 = Dyn IP DHCP
	0 = Static IP	Configuration will be obtained from parameters 05…13.	
	1 = Dyn IP DHCP	Configuration will be obtained via DHCP.	
05	IP address 1	An IP address is assigned to each IP node on a network. An IP address is a 32-bit number that is typically represented in dotted decimal notation consisting of four decimal integers, on the range 0…255, separated by periods. Each integer represents the value of one octet (8-bits) in the IP address. Parameters 05…08 define the four octets of the IP address of the adapter module.	0
	0 … 255	IP address	
06	IP address 2	Second octet of adapter module IP address. See parameter IP address 1 (page 65).	0
	0 … 255	IP address	
07	IP address 3	Third octet of adapter module IP address. See parameter IP address 1 (page 65).	0
	0 … 255	IP address	
08	IP address 4	Fourh octet of adapter module IP address. See parameter IP address 1 (page 65).	0
	0 … 255	IP address	

No.	Name/ Value	Description	Default
09	Subnet CIDR	Subnet masks are used for splitting networks into smaller networks called subnets. A subnet mask is a 32-bit binary number that splits the IP address into a network address and host address. Subnet masks are typically represented in dotted decimal notation or CIDR notation.	0
	31	255.255.255.254	
	30	255.255.255.252	
	29	255.255.255.248	
	28	255.255.255.240	
	27	255.255.255.224	
	26	255.255.255.192	
	25	255.255.255.128	
	24	255.255.255.0	
	23	255.255.254.0	
	22	255.255.252.0	
	21	255.255.248.0	
	20	255.255.240.0	
	19	255.255.224.0	
	18	255.255.192.0	
	17	255.255.128.0	
	16	255.255.0.0	
	15	255.254.0.0	
	14	255.252.0.0	
	13	255.248.0.0	
	12	255.240.0.0	
	11	255.224.0.0	

No.	Name/ Value	Description	Default
	10	255.192.0.0	
	9	255.128.0.0	
	8	255.0.0.0	
	7	254.0.0.0	
	6	252.0.0.0	
	5	248.0.0.0	
	4	240.0.0.0	
	3	224.0.0.0	
	2	192.0.0.0	
	1	128.0.0.0	
10	GW address 1	IP gateways connect individual physical IP subnets into a unified IP network. When an IP node needs to communicate with an IP node on another subnet, the IP node sends the data to the IP gateway for forwarding. Parameters 10…13 define the four octets of the gateway IP address.	0
	0 … 255	GW address	
11	GW address 2	Second octet of the gateway IP address. See parameter GW address 1 (page 67).	0
	0 … 255	GW address	
12	GW address 3	Third octet of the gateway IP address. See parameter GW address 1 (page 67).	0
	0 … 255	GW address	
13	GW address 4	Fourth octet of the gateway IP address. See parameter GW address 1 (page 67).	0
	0 … 255	GW address	
14	Commrate port 2	Sets the bit rate for the Ethernet port 2. This parameter is used only with FSCS-21.	0 = Auto

No.	Name/ Value	Description	Default
	0 = Auto	Autonegotiate	
	1 = 100 Mbps FD	100 Mbps, full duplex	
	2 = 100 Mbps HD	100 Mbps, half duplex	
	3 = 10 Mbps FD	10 Mbps, full duplex	
	4 = 10 Mbps HD	10 Mbps, half duplex	
15	Service configuration	Disables services that are not required. Each service is represented by a single bit. Bit 0, Lock configuration, can be used to prevent accidental changing of this parameter. By default, all services are enabled and configuration is unlocked.	
	0	Lock configuration Changing this bit to one will lock service configuration and no bit can be changed. Only resetting the fieldbus configuration will unlock the parameter. To reset the fieldbus configuration, choose "Reset all fieldbus settings" or "Clear all" in parameter **96.06**.	
	1	Disable IP config tool When this bit is set, access from ABB IP Configuration tool is prevented.	
	2	Reserved	
	3	Disable ping response When this bit is set, response to ICMP (ping) message is prevented.	
	4…15	Reserved	
	0000b … 1111b	Service configuration	

No.	Name/ Value	Description	Default
18	Safety Network Number	Uniquely identifies a network across all networks in the safety system. The user should assign SNN numbers for each safety network or safety sub-net that are unique system-wide. To change a non-zero SNN: 1. Set the value of this parameter to 0. 2. Refresh the configuration of the module with drive parameter 51.27. 3. Set the value of this parameter to the wanted SNN. 4. Refresh the configuration of the module again with drive parameter 51.27.	0
	0	Reset Safety Network Number	
	1 … 9999	Safety Network Number	
19	T16 scale	Defines the scaling for reference 1 and actual 1 with Transparent 16 profile. (Protocol.Profile = EIP T16) Scaling also depends on the selected Reference type on 50.04 FBA A Ref 1 type and 50.34 FBA B Ref 1 type and 50.07 and 50.37 for the actual 1. Ref type = Transparent FBA_A/B_Ref1 = Ref1_from_PLC * (T16_Scale + 1) Ref type = General FBA_A/B_Ref1 = Ref1_from_PLC * (T16_Scale + 1) / 100	99
	0 … 65535	Reference multiplier/actual value divisor	
20	Control timeout	Defines the control timeout value. The EtherNet/IP protocol specifies connection timeout for I/O messaging (Class 1) and Connected explicit messaging (Class 3), but not Unconnected explicit messaging. This parameter provides a timeout for Unconnected explicit messaging and for instances of Connected explicit messaging (Class 3), where the client breaks the connection in between requests.	0

No.	Name/ Value	Description	Default
	0	**I/O messaging (Class 1) and Connected explicit messaging (Class 3):** (Requested Packet Interval) × (Connection Timeout Multiplier) **Note:** Timeout behavior can be modified by Watchdog Timeout Action attribute of Connection object. **Unconnected explicit messaging:** Always timeout Control timeout must be greater than zero to control drive with Unconnected Explicit Messaging.	
	1 … 65534	**I/O messaging (Class 1):** (Requested Packet Interval) × (Connection Timeout Multiplier) **Note:** Timeout behavior can be modified by Watchdog Timeout Action attribute of Connection object. **Connected explicit messaging (Class 3)** and **Unconnected explicit messaging:** 100ms × (Control Timeout Value) since last Control Event	
	65535	**I/O messaging (Class 1):** (Requested Packet Interval) × (Connection Timeout Multiplier) **Note:** Timeout behavior can be modified by Watchdog Timeout Action attribute of Connection object. **Connected explicit messaging (Class 3)** and **Unconnected explicit messaging:** Never Timeout.	

No.	Name/ Value	Description	Default
		Control timeout events: • Write of an output assembly object instance • Write of control bits (Run1, Run2, NetCtrl, NetRef and FaultReset) • Write Speed Reference • Write Torque Reference • Reset Control Supervisor object • Write Force Fault via Control Supervisor object If a timeout occurs, the adapter module signals the drive that communication with the client has been lost. The drive configuration then determines how to respond. Example: If the timeout is 250 ms and the drive is configured to trip on a fault on a communication failure with a delay of 500 ms, then the drive will trip on a fault 750 ms after communications is lost.	
21	Idle action	I/O connections may include a Run/Idle notification. This parameter determines the action the drive takes in response to an Idle notification.	0 = Off-line
	0 = Off-line	In the event of an Idle notification, the adapter module signals the drive that communication with the client has been lost. The drive configuration then determines how to respond. Example: If the timeout is 250 ms and the drive is configured to trip on a fault on a communication failure with a delay of 500 ms, then the drive will trip on a fault 750 ms after communications is lost.	
	1 = On-line	In the event of an Idle notification, the drive will continue to operate using the last command and references received.	
22	Stop function	Determines how the motor is to be stopped when a stop command is received via EtherNet/IP. This parameter only applies to the ODVA AC/DC drive profile.	0 = Ramp
	0 = Ramp	The motor decelerates along the active deceleration ramp.	
	1 = Coast	The motor comes to a stop by coasting.	

No.	Name/ Value	Description	Default
23	Speed scale	This parameter only applies to the ODVA AC/DC drive profile. The units of reference and actual speeds for the ODVA AC/DC drive profile are given by the formula below. Speed unit = RPM × $2^{(-1 \times \text{ODVA speed scale value})}$ Use the speed scale value of the drive parameter when reading/writing parameter **Speed scale** through the drive control panel, Drive parameter object, class 90h and Fieldbus configuration object, class 91h. When written through these methods, the new value takes effect after the drive control unit is restarted or a "Fieldbus Adapter Parameter refresh" is given. Use the ODVA speed scale value when reading/writing parameter Speed scale via AC/DC-drive object, class 2Ah. When written via the AC/DC drive object, the new value takes effect immediately. **Note:** While a wide range of resolutions can be configured, the actual performance is limited to the performance capabilities of the drive. The rows below show how the drive ODVA speed scale parameter values correspond to the ODVA speed scale units.	128
	123	ODVA speed scale value: -5 Unit: 32 RPM	
	124	ODVA speed scale value: -4 Unit: 16 RPM	
	125	ODVA speed scale value: -3 Unit: 8 RPM	
	126	ODVA speed scale value: -2 Unit: 4 RPM	
	127	ODVA speed scale value: -1 Unit: 2 RPM	
	128	ODVA speed scale value: 0 Unit: 1 RPM	
	129	ODVA speed scale value: 1 Unit: 0.5 RPM	

No.	Name/ Value	Description	Default
	130	ODVA speed scale value: 2 Unit: 0.25 RPM	
	131	ODVA speed scale value: 3 Unit: 0.125 RPM	
	132	ODVA speed scale value: 4 Unit: 0.0625 RPM	
	133	ODVA speed scale value: 5 Unit: 0.03125 RPM	
	0 … 255	Speed scale value of drive parameter	
24	Torque scale	This parameter only applies to the ODVA AC/DC drive profile. The units of reference and actual torques for the ODVA AC/DC drive profile are given by the formula below. Torque unit = N·m × $2^{(-1 \times \text{ODVA torque scale})}$ where: (N·m = Newton × Meter) Use the torque scale value of the drive parameter when reading/writing parameter Torque scale through the drive control panel, Drive parameter object, class 90h and Fieldbus configuration object, class 91h. When written through these methods, the new value takes effect after the drive control unit is restarted or a “Fieldbus Adapter Parameter refresh” is given. Use the ODVA torque scale value when reading/writing parameter Torque scale via AC/DC-drive object, class 2Ah. When written via the AC/DC drive object, the new value takes effect immediately. **Note:** While a wide range of resolutions can be configured, the actual performance is limited to the performance capabilities of the drive. The rows below show how the drive ODVA torque scale parameter values correspond to the ODVA torque scale units.	128
	123	ODVA torque scale value: -5 Unit: 32 N·m	
	124	ODVA torque scale value: -4 Unit: 16 N·m	

No.	Name/ Value	Description	Default
	125	ODVA torque scale value: -3 Unit: 8 N·m	
	126	ODVA torque scale value: -2 Unit: 4 N·m	
	127	ODVA torque scale value: -1 Unit: 2 N·m	
	128	ODVA torque scale value: 0 Unit: 1 N·m	
	129	ODVA torque scale value: 1 Unit: 0.5 N·m	
	130	ODVA torque scale value: 2 Unit: 0.25 N·m	
	131	ODVA torque scale value: 3 Unit: 0.125 N·m	
	132	ODVA torque scale value: 4 Unit: 0.0625 N·m	
	133	ODVA torque scale value: 5 Unit: 0.03125 N·m	
	0 … 255	Torque scale value of drive parameter	
25 … 26	Reserved		
27	FBA A/B par refresh	Validates changed adapter module configuration parameter settings. After refreshing, the value reverts automatically to 0 = Done. **Note:** This parameter cannot be changed while the drive is running.	0 = Done
	0 = Done	Refreshing done	
	1 = Refresh	Refreshing	

No.	Name/ Value	Description	Default
28	FBA A/B par table ver	Read-only. Displays the parameter table revision of the fieldbus adapter module mapping file stored in the memory of the drive, in format xyz, where x = major revision number y = minor revision number z = correction number OR in format axyz, where a = major revision number xy = minor revision numbers z = correction number or letter.	N/A
		Parameter table revision	
29	FBA A/B drive type code	Read-only. Displays the drive type code of the fieldbus adapter module mapping file stored in the memory of the drive.	N/A
		Drive type code of the fieldbus adapter module mapping file	
30	FBA A/B mapping file ver	Read-only. Displays the fieldbus adapter module mapping file revision stored in the memory of the drive in decimal format.	N/A
		Mapping file revision	
31	D2FBA A/B comm status	Read-only. Displays the status of the fieldbus adapter module communication. **Note:** The value names can vary by drive.	0 = Idle or 4 = Off-line
	0 = Idle	Adapter is not configured.	
	1 = Exec.init	Adapter is initializing.	
	2 = Time out	A timeout has occurred in the communication between the adapter and the drive.	
	3 = Conf.err	Adapter configuration error: The major or minor revision code of the common program revision in the fieldbus adapter module is not the revision required by the module or mapping file upload has failed more than three times.	
	4 = Off-line	Adapter is off-line.	

No.	Name/ Value	Description	Default
	5 = On-line	Adapter is on-line.	
	6 = Reset	Adapter is performing a hardware reset.	
32	FBA A/B comm SW ver	Read-only. Displays firmware patch and build number of the adapter module in the xxyy format, where: xx = patch number yy = build number Example: If the firmware version (<major>.<minor>.<patch>.<build>) is 3.10.200.13, the value C80D is displayed. If the version is 3.10.0.0, the value 0 is displayed. See also parameter 33 FBA A/B appl SW ver (page 76).	N/A
33	FBA A/B appl SW ver	Read-only. Displays firmware version of the adapter module in xxyy format, where: xx = major revision number yy = minor revision number Example: If the firmware version (<major>.<minor>.<patch>.<build>) is 3.10.200.13 or 3.10.0.0, the value 310 is displayed. See also parameter 32 FBA A/B comm SW ver (page 76).	N/A

FSCS-21 configuration parameters – group B (group 2)

Note: The actual parameter group number depends on the drive type. Group B (group 2) corresponds to:

- parameter group 53 in ACS380 and ACS580
- parameter group is typically 53/56 (153/156 in some variants) in ACS880 if the adapter is installed as fieldbus adapter A/B.

No.[1)]	Name/ Value	Description	Default
01	FBA A/B data out 1 (client to drive)	In output assembly instances that include drive parameters, this parameter specifies which parameter's value will be placed in location Data out 1 value received by the drive from the EtherNet/IP client.	0 = None
	0 = None	Not used	
	1 … 99	Virtual address area of drive control. Not used with the EtherNet/IP protocol.	

No.[1]	Name/ Value	Description	Default
	101 … 9999	Parameter area of the drive. Parameter index with format xxyy, where • xx is the parameter group number (1…99) • yy is the parameter number index within that group (01…99). **Note:** In ACS580 and ACS880, choose Other to display a list of mappable drive parameters.	
02 … 10	Data out 2 … Data out 10	See parameter 01 FBA A/B data out 1 (client to drive) (page 76).	0 = None

1) The number of parameters in this group may vary by drive type and drive firmware.

FSCS-21 configuration parameters – group C (group 3)

Note: The actual parameter group number depends on the drive type. Group C (group 3) corresponds to:

- parameter group 52 in ACS380 and ACS580
- parameter group is typically 52/55 (152/155 in some variants) in ACS880 and ACS880-M04 if the adapter is installed as fieldbus adapter A/B.

No.[1]	Name/ Value	Description	Default
01	FBA A/B data in1 (drive to client)	In input assembly instances that include drive parameters, this parameter specifies which parameter's value will be placed in location Data in 1 value sent by the drive to the EtherNet/IP client.	0 = None
	0 = None	Not used	
	1 … 99	Virtual address area of drive control. Not used with the EtherNet/IP protocol.	
	101 … 9999	Parameter area of the drive. Parameter index with format xxyy, where • xx is the parameter group number (1…99) • yy is the parameter number index within that group (01…99). **Note:** In ACS580 and ACS880, choose Other to display a list of mappable drive parameters.	

No.[1]	Name/ Value	Description	Default
02 … 10	Data in 2 … Data in 10	See parameter 01 FBA A/B data in1 (drive to client) (page 77).	0 = None

1) The number of parameters in this group may vary by drive type and drive firmware.

Control locations

ABB drives can receive control information from multiple sources including digital inputs, analog inputs, the drive control panel and a fieldbus adapter module. ABB drives allow the user to separately determine the source for each type of control information (Start, Stop, Direction, Reference, Fault reset, etc.).

To give the fieldbus client the most complete control over the drive, you must select the adapter module as the source of this information. The drive-specific parameter setting examples below contain the drive control parameters relevant in the examples. For a complete parameter list, see the drive documentation.

Starting up fieldbus communication for the drive

1. Power up the drive.
2. Enable the communication between the adapter module and the drive by selecting the correct slot number in parameter 50.01 FBA A enable.
   The selection must correspond to the slot where the adapter module is installed. For example, if the adapter module is installed in slot 2, you must select slot 2.
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 module 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 assigns the Control word, Status word, references 1…2 and actual values 1…2 automatically to cyclical communication according to the selected assembly instances.

8. Save the valid parameter values to permanent memory with parameter 96.07 Parameter save manually.
9. Validate the settings made in parameter groups 51, 52 and 53 with parameter 51.27 FBA A par refresh.
10. Set the relevant drive control parameters to control the drive according to the application.
    Examples of appropriate values are shown in the tables below.

Parameter setting examples – ACS380, ACS580 and ACS880

Speed control using the ODVA AC/DC drive profile, Extended speed control assembly

This example shows how to configure a speed control application that uses the ODVA AC/DC drive profile, Extended speed control assembly. In addition, some application-specific data is added to the communication.

The start/stop commands and reference scaling are according to the ODVA AC/DC drive profile. For more information, see section ODVA AC/DC drive profile (page 93).

When Reference 1 (REF1) is used for speed control and the value of parameter 51.23 is 128, an ODVA speed reference value of ±30000 (decimal) corresponds to an equal amount of rpm in the drive. The speed reference value sent from the PLC is limited by parameter 30.12 Maximum speed in the forward direction and 30.11 Minimum speed in the reverse direction.

The minimum and maximum 16-bit integer values that can be given through the fieldbus are -32768 and 32767 respectively.

Bytes	Instance 121	Instance 171
0…1	Control word	Status word
2…3	Speed reference	Speed actual value
4…7	Constant speed 1 [32] 1)	Output power [32] 1)
8…11	Constant speed [32] 1)	DC voltage [32] 1)

1) Example

The table below gives the recommended drive parameter settings.

Drive parameter	Setting for drives	Description
50.01 FBA A enable	1 = Option slot 1[1)]	Enables communication between the drive and the fieldbus adapter module.
50.02 FBA A comm loss func	1 = Fault[1)]	Enables fieldbus A communication fault monitoring.
50.03 FBA A comm loss t out	3.0 s[1)]	Defines the fieldbus A communication break supervision time.
50.04 FBA A ref1 type	4 = Speed	Selects the fieldbus A reference 1 type and scaling.
51.01 FBA A type	2222 = EtherNet/IP[2)]	Displays the type of the fieldbus adapter module.
51.02 Protocol/Profile	100 = EIP AC/DC	Selects the EtherNet/IP protocol and the ODVA AC/DC drive profile.
51.03 Commrate	0 = Auto[1)]	Ethernet communication rate is negotiated automatically by the device.
51.04 IP configuration	0 = Static IP[1)]	Configuration will be obtained from parameters 51.05…51.13.
51.05 IP address 1	192[1)]	First part of the IP address
51.06 IP address 2	168[1)]	Second part of the IP address
51.07 IP address 3	0[1)]	Third part of the IP address
51.08 IP address 4	16[1)]	Fourth part of the IP address
51.09 Subnet CIDR	24[1)]	Sets the network mask. In this example, the network mask is set as 255.255.255.0.
51.23 ODVA speed scale	128[1)]	Sets the scaling for the ODVA speed reference.
52.01 FBA data in1	01.14[1)]	Output power
52.03 FBA data in3	01.11[1)]	DC voltage
53.01 FBA data out1	22.26[1)]	Constant speed 1
53.03 FBA data out3	22.27[1)]	Constant speed 2

Drive parameter	Setting for drives	Description
51.27 FBA A par refresh	1 = Refresh	Validates the FSCS-21 configuration parameter settings.
20.01 Ext1 commands	12 = Fieldbus A	Selects the fieldbus A interface as the source of the start and stop commands for external control location 1.
22.11 Speed ref1 source	4 = FB A ref1	Selects the fieldbus A reference 1 as the source for speed reference 1.

1) Example
2) Read-only or automatically detected/set

The start sequence for the parameter example above is given below.

Control word:

- Reset the fieldbus communication fault (if active).
- Enter 0h (0 decimal) → READY.
- Enter 1h (1 decimal) → ENABLED (Running forward).
- Enter 2h (2 decimal) → ENABLED (Running reverse).

Configuring the client

After the adapter module has been initialized by the drive, you must prepare the client for communication with the module. Refer to the documentation of the client system.

The example applies to all drive types compatible with the module.

Before you start

Decide on these points before you start the client configuration.

Select profile

The communication profile determines which I/O assemblies and objects are available. See chapter EtherNet/IP – Communication profiles (page 91) for more information.

Select output and input assembly instances

EtherNet/IP devices implement multiple objects each with many attributes. While it is possible to write or read each attribute separately to control the drive, this is inefficient. Assembly object instances provide a means to group writes or reads of attributes. The selection of assembly objects is limited by the choice of the communication profile. This table gives a listing of the output and input assemblies.

Name	Output instance	Input instance	Size (bytes)	Size (16-bit words)	Profile
Basic Speed Control	20	70	4	2	ODVA AC/DC drive
Enhanced Speed Control	21	71	4	2	ODVA AC/DC drive
Basic Speed and Torque Control	22	72	6	3	ODVA AC/DC drive
Enhanced Speed and Torque Control	23	73	6	3	ODVA AC/DC drive
Basic Speed Control plus Drive Parameters	120	170	24	12	ODVA AC/DC drive
Enhanced Speed Control plus Drive Parameters	121	171	24	12	ODVA AC/DC drive
Basic Speed and Torque Control plus Drive Parameters	122	172	26	13	ODVA AC/DC drive
Enhanced Speed and Torque Control plus Drive Parameters	123	173	26	13	ODVA AC/DC drive
ABB Drives Profile w/ Set Speed	1	51	4	2	ABB Drives profile
ABB Drives Profile w/ Set Speed and Set Torque	2	52	6	3	ABB Drives profile
ABB Drives Profile w/ Set Speed plus Drive Parameters	101	151	24	12	ABB Drives profile
ABB Drives Profile w/ Set Speed and Set Torque plus Drive Parameters	102	152	26	13	ABB Drives profile
Transparent16 w/One	11	61	4	2	Transparent16 profile
Transparent16 w/Two	12	62	6	3	Transparent16 profile
Transparent16 w/One plus Drive Parameters	111	161	24	12	Transparent16 profile

Name	Output instance	Input instance	Size (bytes)	Size (16-bit words)	Profile
Transparent16 w/Two plus Drive Parameters	112	162	26	13	Transparent16 profile
Transparent32 w/One	41	91	8	4	Transparent32 profile
Transparent32 w/Two	42	92	12	6	Transparent32 profile
Transparent32 w/One plus Drive Parameters	141	191	28	14	Transparent32 profile
Transparent32 w/Two plus Drive Parameters	142	192	32	16	Transparent32 profile

Select connection method

EtherNet/IP provides a variety of connection methods to communicate between devices. Not all methods are supported by all devices. Refer to the client documentation to determine which method(s) are supported by the client.

Note: The choice of the connection method has a significant impact on the timeout behavior. Refer to configuration parameters 20 Control timeout and 21 Idle action for more information.

The FSCS-21 adapter module supports the following connection methods:

Connection method	Description
I/O connections	The adapter module supports Class 1 I/O connections. I/O connections are often also referred to as “Implicit Messaging”. I/O connections are typically established by configuring an I/O scanner to write and read assembly object instances.
Connected explicit messaging	The adapter module supports Class 3 connected explicit messaging. Class 3 connected explicit messages are typically established by using a “message instruction” to write or read an attribute. **Note:** When using Class 3 explicit messaging, some EtherNet/IP clients may close the connection after the MSG instruction is done. This will cause the module to behave as if it were controlled via unconnected explicit messaging.

Connection method	Description
Unconnected explicit messaging	The adapter module supports unconnected explicit messaging. Unconnected explicit messages are typically established by using a “message instruction” to write or read an attribute. **Note:** EtherNet/IP does not provide a timeout means for unconnected explicit messaging. To use unconnected explicit messaging for control, refer to configuration parameter 20 Control timeout (page 69).

EDS files

Electronic Data Sheet (EDS) files specify the properties of the device for the EtherNet/IP client. The client identifies the device by means of the product code, device type and major revision attributes.

FSCS-21 is configured with an EDS file. The EDS file is specific to the option module, and is compatible with different drive and firmware versions.

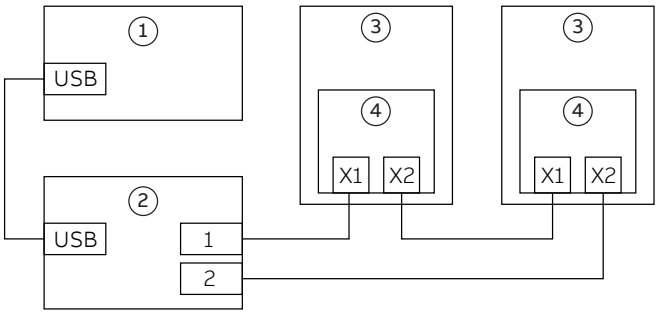
EDS files are available from ABB Library (https://www.abb.com/drives/documents).

Configuring DLR topology for FSCS-21

This example shows how to prepare an Allen-Bradley® CompactLogix™ PLC for DLR topology with FSCS-21 adapter modules. After installing the devices on the DLR network, at least one supervisor node must be configured. Configuration can be done by using the Studio 5000® Logix Designer or RSLinx® Classic Lite software.

Setup using Logix Designer

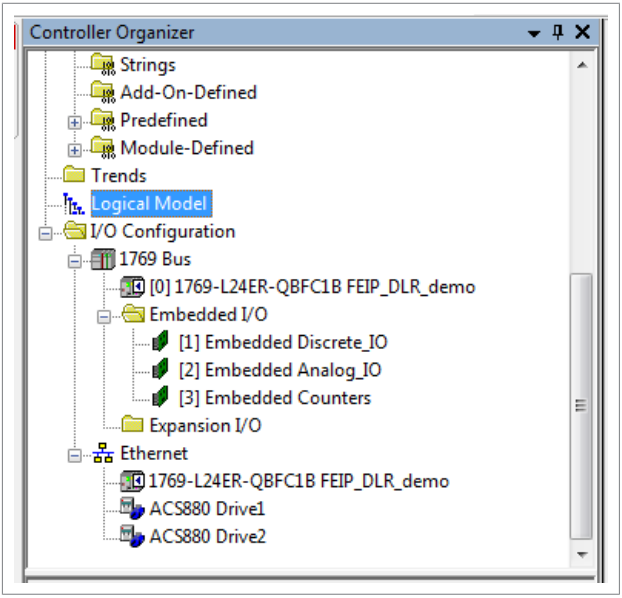
1. Open the Studio 5000® software. Test setup uses an Allen Bradley PLC connected in a ring topology with two FSCS-21 fieldbus Ethernet modules. The topology used in the example is shown below. More devices can be added, but the recommended maximum number of nodes on a single DLR network is 50.



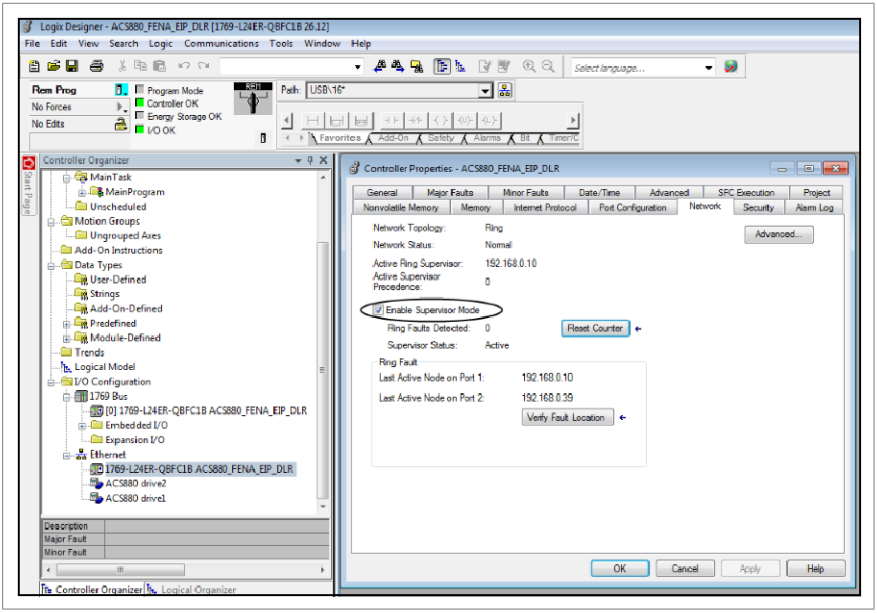
1	PC Studio 5000
2	PLC
3	Drive
4	FSCS-21 module

Note: It is assumed that the PLC configuration was already established in the Studio 5000® project and the EDS file(s) are installed and at least two FSCS-21 modules are added to the project.

For more information on adding modules to a project and installing EDS files, refer to section Configuration (page 186).



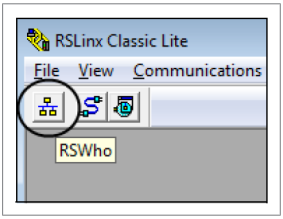
2. Download the project to the PLC.
3. Go online with the PLC and leave it in Program mode.
4. Double-click the module in the I/O Configuration. In the Controller Properties window, open Network tab and select Enable Supervisor Mode. Click OK.



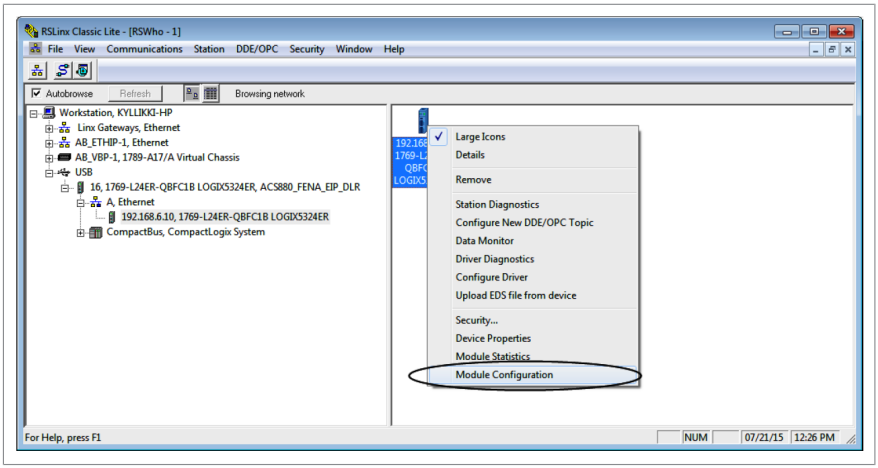
Setup using RSLinx® Classic

You can configure and enable DLR supervisor via RSLinx®Classic.

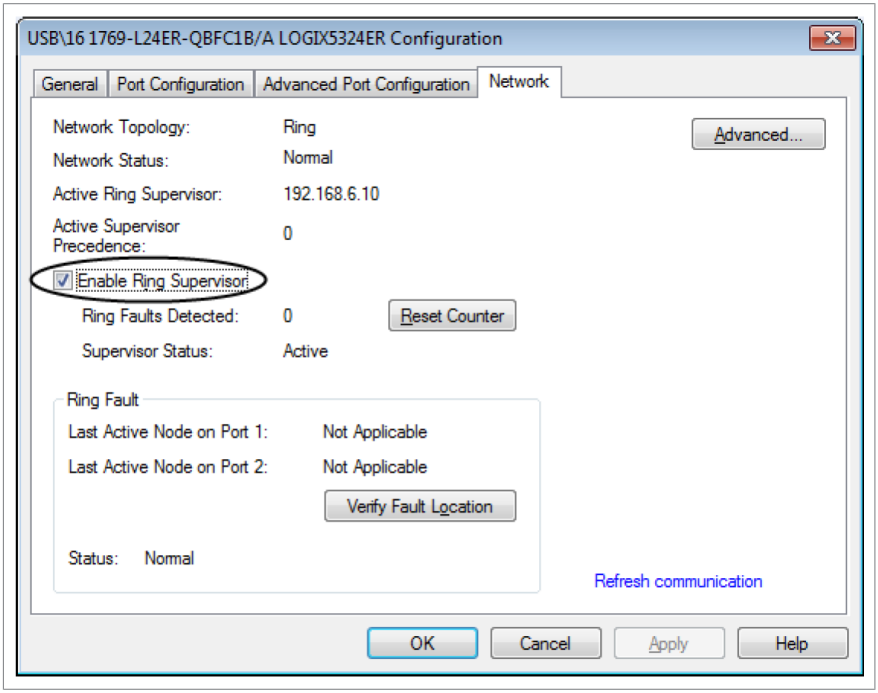
1. Open the RSLinx®Classic software.
2. Browse to the DLR network.



3. Open the Module Configuration by right-clicking on the ring supervisor in the list.



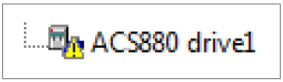
4. On the Network tab, select Enable Ring Supervisor, to enable DLR messages in the ring.



5. Click Advanced... to configure DLR parameters, such as Beacon Interval and Beacon Timeout.

Note: It is recommended to use the default values.

6. Go back to Logix Designer and make sure that none of the FSCS-21 modules are faulted (no warning symbols are shown).



14

EtherNet/IP – Communication profiles

Contents of this chapter

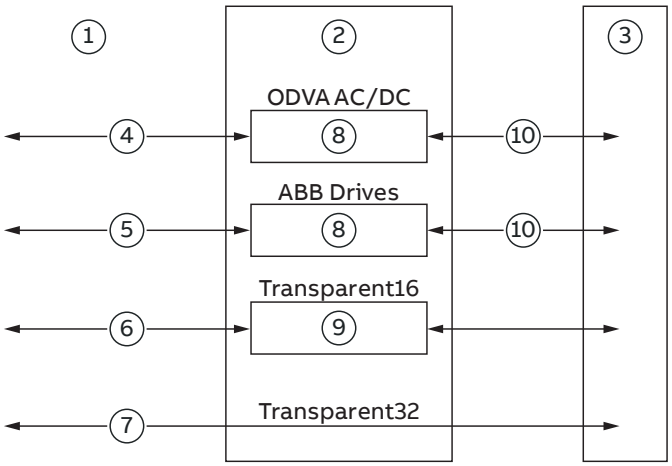
This chapter describes the communication profiles used in the communication between the EtherNet/IP client, the adapter module and the drive. It also describes the Control word, the Status word, references and actual values for the ODVA AC/DC drive and ABB Drives communication profiles. Refer to the drive manuals for details on the native profiles.

Communication profiles

Communication profiles are ways of conveying control commands (Control word, Status word, references and actual values) between the master station and the drive.

With the FSCS-21 adapter module, the EtherNet/IP network may employ either the ODVA AC/DC drive profile or the ABB Drives profile. Both are converted to the native profile (eg, DCU or FBA) by the adapter module. In addition, two Transparent modes – for 16-bit and 32-bit words respectively – are available. With the Transparent modes, no data conversion takes place.

The figure below illustrates the profile selection:



1	EtherNet/IP network
2	FSCS-21 module
3	Drive
4	ODVA AC/DC profile
5	ABB Drives profile
6	Drive-specific profile (with 16-bit words) [1)]
7	Drive-specific profile (with 32-bit words)[1)]
8	Data conversion
9	Optional reference/actual value scaling
10	Native profile (for example, DCU, FBA)

1) Can be used if the drive supports the native profile.

ODVA AC/DC drive profile

This section briefly describes the ODVA AC/DC drive profile. Additional information is available at www.odva.org.

An EtherNet/IP node is modeled as a collection of abstract objects. Each object represents the interface to and behavior of a component within the product. The ODVA AC/DC drive profile defines a collection of objects suitable for the control of AC and DC drives. The objects supported by the adapter module are listed in section Class objects (page 143).

Objects are defined by:

- Service
- Class
- Instance
- Attribute
- Behavior.

For example, to set the drive speed reference, the Set_Attribute_Single service can be requested for the SpeedRef attribute of the AC/DC drive object class. The resulting behavior is that the reference speed of the drive is set to the requested value.

This is an example of explicit messaging, where each attribute of a class is set individually. While this is allowed, it is inefficient. Instead, implicit messaging using input and output assembly Instances is recommended. Implicit messaging allows the EtherNet/IP client to set or get predefined groups of attributes in a single message exchange. The assembly instances supported by the adapter module are listed and defined in section Assembly objects (page 108).

ODVA output attributes

This section briefly describes the instances found in the output assemblies of the ODVA AC/DC drive profile. Note that all output assembly instances do not support all attributes listed here.

In the ODVA EtherNet/IP specification the word **output** is used to describe data flow from the network into a device (such as the adapter module).

Run Forward & Run Reverse (Control supervisor object)

These attributes are used to assert run and stop commands to the Control supervisor object state machine according to the following Run/Stop event matrix. See sectionState (Control supervisor object) (page 97).

RunFwd (Run1)	RunRev (Run2)	Trigger event	Run type
0	0	Stop	N/A
0 → 1	0	Run	RunFwd
0	0 → 1	Run	RunRev
0 → 1	0 → 1	No action	N/A
1	1	No action	N/A
0 → 1	1	Run	RunRev
1	1 → 0	Run	RunFwd

Fault Reset (Control supervisor object)

This attribute resets a drive fault on a transition from zero to one if the condition that caused the fault has been cleared.

Net Ctrl (Control supervisor object)

This attribute requests that the drive Run/Stop command be supplied locally (Net Ctrl = 0) or by the network (Net Ctrl = 1).

Net Ref (AC/DC drive object)

This attribute requests that the drive speed and torque references be supplied locally (Net Ref = 0) or by the network (Net Ref = 1).

Speed Reference (AC/DC drive object)

This attribute is the speed reference for the drive. The units are scaled by the Speed Scale attribute of the AC/DC drive object. See parameter 23 Speed scale (page 72) for details.

Scalar mode

When the drive is operating in the scalar mode, the adapter module provides the drive with a frequency reference. The ODVA AC/DC drive profile uses rpm units for the speed reference. The drive frequency reference is calculated as follows:

$$Dfr = \frac{Osr \times Us \times Mf}{Mss}$$

where

Dfr = Drive Frequency Reference in Hz

Osr = ODVA Speed Reference

Us = ODVA Speed Unit (see parameter 23 Speed scale (page 72))

Mf = Motor Nominal Frequency in Hz

Mss = Motor Synchronous Speed in rpm (not Motor Nominal Speed).

For example, for a 4-pole 60 Hz motor (Mss = 1800 rpm) with a unit of 1 rpm and an ODVA Speed Reference of 900, the drive frequency reference is:

$$Dfr = \frac{Osr \times Us \times Mf}{Mss} = \frac{900 \times 1rpm \times 60Hz}{1800rpm} = 30Hz$$

Vector mode

When the drive is operating in the vector mode, the adapter module provides the drive with a speed reference. The ODVA AC/DC drive profile uses rpm units for the speed reference. The drive speed reference is calculated as follows:

$$Dsr = Osr \times Us$$

where

Dsr = Drive Speed Reference in rpm

Osr = ODVA Speed Reference

Us = ODVA Speed Unit (see parameter 23 Speed scale (page 72)).

For example, for an ODVA Speed Reference of 900 rpm with a unit of 0.5 rpm, the drive speed reference is:

$$Dsr = Osr \times Us = 900 \times 0.5rpm = 450rpm$$

Torque Reference (AC/DC drive object)

This attribute is the torque reference for the drive. The units are scaled by the Torque Scale attribute of the AC/DC drive object. See parameter 24 Torque scale (page 73) for details.

The adapter module provides the drive with a torque reference in percent of the motor nominal torque. The ODVA AC/DC drive profile uses Newton-meter (N·m) units for the torque reference. The drive torque reference is calculated as follows:

$$Dtr = \frac{100 \times Otr \times Ut}{Mt}$$

where

Dtr = Drive Torque Reference in Percent of Motor Nominal Torque

Otr = ODVA Torque Reference

Ut = ODVA Torque Unit (see 24 Torque scale on page 128)

Mt = Motor Nominal Torque in N·m.

For example, for a 1000 N·m Motor Nominal Torque with a unit of 1 N·m and an ODVA Torque Reference of 500, the drive torque reference is:

$$Dtr = \frac{100 \times Otr \times Ut}{Mt} = \frac{100 \times 500 \times 1\,Nm}{1000\,Nm} = 50$$

ODVA input attributes

This section briefly describes the instances found in the ODVA AC/DC drive profile's input assemblies. Note that all input assembly instances do not support all attributes listed here.

In the ODVA EtherNet/IP specification the word input is used to describe data flow from a device (such as the adapter module) to the network.

Faulted (Control supervisor object)

This attribute indicates that the drive has experienced a fault. The fault code may be read from the FaultCode attribute of the Control supervisor object.

Warning (Control supervisor object)

This attribute indicates that the drive is experiencing a warning condition. The warning code may be read from the WarnCode attribute of the Control supervisor object.

Running Forward (Control supervisor object)

This attribute indicates that the drive is running in the forward direction.

Running Reverse (Control supervisor object)

This attribute indicates that the drive is running in the reverse direction.

Ready (Control supervisor object)

This attribute indicates that the Control supervisor object state machine is in the Ready, Running or Stopping state. See State (Control supervisor object) (page 97).

Ctrl From Net (Control supervisor object)

This attribute indicates if the Run/Stop command is being supplied locally (Ctrl From Net = 0) or by the network (Ctrl From Net = 1).

Ref From Net (AC/DC drive object)

This attribute indicates if the speed and torque references are being supplied locally (Ref From Net = 0) or by the network (Ref From Net = 1).

At Reference (AC/DC drive object)

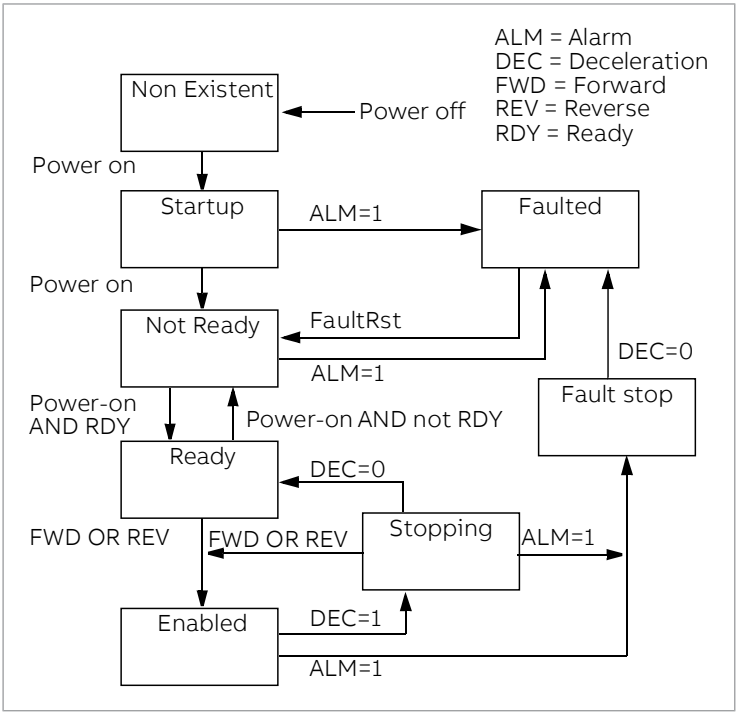
This attribute indicates that the drive is operating at the specified speed or torque reference.

State (Control supervisor object)

This attribute indicates the current state of the Control supervisor object.

State	Description	State	Description
0	Vendor-specific	4	Enabled
1	Startup	5	Stopping
2	Not ready	6	Fault stop
3	Ready	7	Faulted

The ODVA state transition diagram is shown below:



Speed Actual (AC/DC drive object)

This attribute indicates the actual speed at which the drive is operating. The units are scaled by the SpeedScale attribute of the AC/DC drive object. See parameter 23 Speed scale (page 72) for details.

Scalar mode

When the drive is operating in the scalar mode, the drive provides the adapter module with a frequency actual. The ODVA AC/DC drive profile uses rpm units for the speed actual. The ODVA Speed Actual is calculated as follows:

$$Osa = \frac{Dfa \times Mss}{Mf \times Us}$$

where

Osa = ODVA Speed Actual

Dfa = Drive Frequency Actual in Hz

Us = ODVA Speed Unit (see parameter 23 Speed scale (page 72))

Mf = Motor Nominal Frequency in Hz

Mss = Motor Synchronous Speed in rpm (not Motor Nominal Speed).

For example, for a 4-pole 60 Hz motor (Mss = 1800 rpm) with a unit of 1 rpm and a Drive Frequency Actual of 30 Hz, the ODVA Speed Actual is:

$$Osa = \frac{Dfa \times Mss}{Mf \times Us} = \frac{30Hz \times 1800rpm}{60Hz \times 1rpm} = 900$$

Vector mode

When the drive is operating in the vector mode, the drive provides the adapter module with a speed actual. The ODVA AC/DC drive profile uses rpm units for the speed actual. The ODVA Speed Actual is calculated as follows:

$$Osa = \frac{Dsa}{Us}$$

where

Dsa = Drive Speed Actual in rpm

Osa = ODVA Speed Actual

Us = ODVA Speed Unit (see parameter 23 Speed scale (page 72)).

For example, for a Drive Speed Actual of 900 rpm with a unit of 0.5 rpm, the ODVA Speed Actual is:

$$Osa = \frac{Dsa}{Us} = \frac{450rpm}{0.5rpm} = 900$$

Torque Actual (AC/DC drive object)

This attribute indicates the actual torque at which the drive is operating. The units are scaled by the Torque Scale attribute of the AC/DC drive object. See parameter 24 Torque scale (page 73) for details.

The drive provides the adapter module with a torque actual in percent of the Motor Nominal Torque. The ODVA AC/DC drive profile uses Newton-meter (N·m) units for the torque actual. The ODVA Torque Actual is calculated as follows:

$$Ota = \frac{Dta \times Mt}{100 \times Ut}$$

where

Dta = Drive Torque Actual in Percent of Motor Nominal Torque

Ota = ODVA Torque Actual

Ut = ODVA Torque Unit (see parameter 24 Torque scale (page 73))

Mt = Motor Nominal Torque in N·m.

For example, for a 1000 N·m Motor Nominal Torque with a unit of 1 N·m and a drive torque actual of 50%, the ODVA Torque Actual is:

$$Ota = \frac{Dta \times Mt}{100 \times Ut} = \frac{50 \times 1000Nm}{100 \times 1Nm} = 500$$

ABB Drives communication profile

Control word and Status word

The Control word is the principal means for controlling the drive from a fieldbus system. It is sent by the fieldbus client station to the drive through the adapter module. The drive switches between its states according to the bit-coded instructions in the Control word and returns status information to the client in the Status word.

The contents of the Control word and the Status word are detailed below. The drive states are presented in section State machine (page 104).

Control word contents

The table below shows the contents of the Control word for the ABB Drives communication profile. The upper case boldface text refers to the states shown in section State machine (page 104).

Bit	Name	Value	STATE/Description
0	OFF1_CONTROL	1	Proceed to READY TO OPERATE.
		0	Stop along currently active deceleration ramp. Proceed to OFF1 ACTIVE; proceed to READY TO SWITCH ON unless other interlocks (OFF2, OFF3) are active.
1	OFF2_CONTROL	1	Continue operation (OFF2 inactive).
		0	Emergency OFF, coast to stop. Proceed to OFF2 ACTIVE, proceed to SWITCH-ON INHIBITED.
2	OFF3_CONTROL	1	Continue operation (OFF3 inactive).
		0	Emergency stop, stop within time defined by drive parameter. Proceed to OFF3 ACTIVE; proceed to SWITCH-ON INHIBITED. **⚠WARNING** Make sure that the motor and driven machine can be stopped using this stop mode.
3	INHIBIT_OPERATION	1	Proceed to OPERATION ENABLED. **Note:** Run enable signal must be active; see drive documentation. If the drive is set to receive the Run enable signal from the fieldbus, this bit activates the signal.
		0	Inhibit operation. Proceed to OPERATION INHIBITED.
4	RAMP_OUT_ZERO	1	Normal operation. Proceed to RAMP FUNCTION GENERATOR: OUTPUT ENABLED.
		0	Force Ramp Function Generator output to zero. Drive ramps to stop (current and DC voltage limits in force).
5	RAMP_HOLD	1	Enable ramp function. Proceed to RAMP FUNCTION GENERATOR: ACCELERATOR ENABLED.
		0	Halt ramping (Ramp Function Generator output held).
6	RAMP_IN_ZERO	1	Normal operation. Proceed to OPERATION. **Note:** This bit is effective only if the fieldbus interface is set as the source for this signal by drive parameters.
		0	Force Ramp Function Generator input to zero.

Bit	Name	Value	STATE/Description
7	RESET	0 → 1	Fault reset if an active fault exists. Proceed to SWITCH-ON INHIBITED. **Note:** This bit is effective only if the fieldbus interface is set as the source for this signal by drive parameters.
		0	Continue normal operation.
8 ... 9	Drive-specific (For information, see the drive documentation.)		
10	REMOTE_CMD	1	Fieldbus control enabled.
		0	Control word and reference not getting through to the drive, except for CW bits OFF1, OFF2 and OFF3.
11	EXT_CTRL_LOC	1	Select External Control Location EXT2. Effective if control location parameterized to be selected from fieldbus.
		0	Select External Control Location EXT1. Effective if control location parameterized to be selected from fieldbus.
12 ... 15	Reserved		

Status word contents

The table below shows the contents of the Status word for the ABB Drives communication profile. The upper case boldface text refers to the states shown in section State machine (page 104).

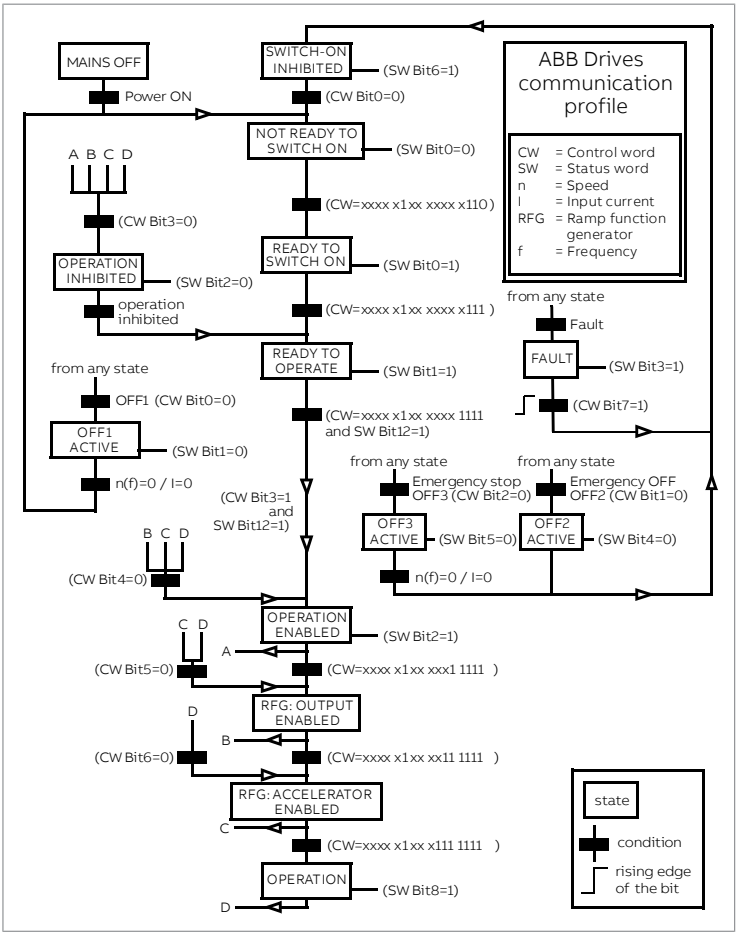
Bit	Name	Value	STATE/Description
0	RDY_ON	1	READY TO SWITCH ON
		0	NOT READY TO SWITCH ON
1	RDY_RUN	1	READY TO OPERATE
		0	OFF1 ACTIVE
2	RDY_REF	1	OPERATION ENABLED
		0	OPERATION INHIBITED

Bit	Name	Value	STATE/Description
3	TRIPPED	1	FAULT
		0	No fault
4	OFF_2_STA	1	OFF2 inactive
		0	OFF2 ACTIVE
5	OFF_3_STA	1	OFF3 inactive
		0	OFF3 ACTIVE
6	SWC_ON_INHIB	1	SWITCH-ON INHIBITED
		0	–
7	ALARM	1	Warning/Alarm
		0	No warning/alarm
8	AT_SETPOINT	1	OPERATION. Actual value equals reference (= is within tolerance limits, ie, in speed control, speed error is 10% max. of nominal motor speed).
		0	Actual value differs from reference (= is outside tolerance limits).
9	REMOTE	1	Drive control location: REMOTE (EXT1 or EXT2)
		0	Drive control location: LOCAL
10	ABOVE_LIMIT	1	Actual frequency or speed equals or exceeds supervision limit (set by drive parameter). Valid in both directions of rotation.
		0	Actual frequency or speed within supervision limit
11	EXT_CTRL_LOC	1	External Control Location EXT2 selected. **Note:** For ACS880: This bit is effective only if the fieldbus interface is set as the target for this signal by drive parameters. User bit 0 selection (06.33).
		0	External Control Location EXT1 selected

Bit	Name	Value	STATE/Description
12	EXT_RUN_ENABLE	1	External Run Enable signal received. **Note:** For ACS880: This bit is effective only if the fieldbus interface is set as the target for this signal by drive parameters. User bit 1 selection (06.34)
		0	No External Run Enable signal received
13 ... 14	Reserved		
15	FBA_ERROR	1	Communication error detected by fieldbus adapter module
		0	Fieldbus adapter communication OK

State machine

The state machine for the ABB Drives communication profile is shown below.



References

References are 16-bit words containing a sign bit and a 15-bit integer. A negative reference (indicating reversed direction of rotation) is formed by calculating the two's complement from the corresponding positive reference.

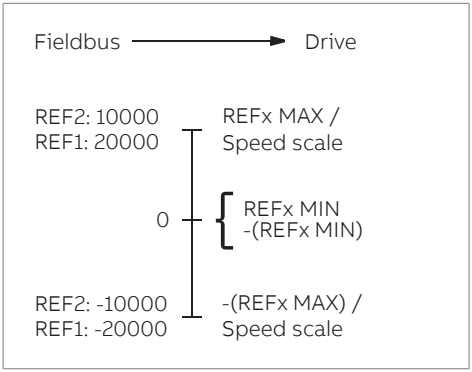
ABB drives can receive control information from multiple sources including analog and digital inputs, the drive control panel and a fieldbus adapter module (for example, FSCS-21). To have the drive controlled through the fieldbus, you must select the module as the source for control information, for example, reference.

Scaling

References are scaled as shown below.

Note: The values of REF1 MAX and REF2 MAX are set with drive parameters. See the drive manuals for further information.

In ACS380, ACS580 and ACS880, the speed reference (REFx) in decimal (0…20000) corresponds to 0…100% of the speed scaling value (as defined with a drive parameter).



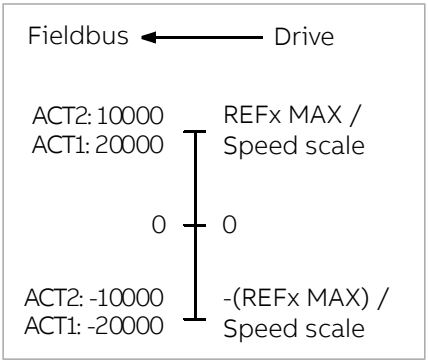
Actual values

Actual values are 16-bit words containing information on the operation of the drive. The functions to be monitored are selected with a drive parameter.

Scaling

Actual values are scaled as shown below.

Note: The values of REF1 MAX and REF2 MAX are set with drive parameters. See the drive manuals for further information.



15

EtherNet/IP – Communication protocol

Contents of this chapter

This chapter describes the EtherNet/IP communication protocol for the adapter module.

EtherNet/IP

EtherNet/IP is a variant of the Common Industrial Protocol (CIP) family of communication protocols intended for supervision and control of automation equipment. Specifically, it covers the use of CIP messaging over an IP network, typically using Ethernet as the media.

The FSCS-21 adapter module acts as a target on an EtherNet/IP network with support for the ODVA AC/DC drive, ABB Drives and Transparent profiles. Two simultaneous EtherNet/IP connections are supported, that is, two originators can be connected to the adapter module at a time.

Further information on the EtherNet/IP protocol is available at www.odva.org.

Object modeling and functional profiles

One of the main features of EtherNet/IP is object modeling. A group of objects can be described with a Functional Profile. The FSCS-21 adapter module realizes the ODVA AC/DC drive Functional Profile with additional features.

Assembly objects

I/O assembly instances may also be referred to as Block Transfer of data. Intelligent devices realizing a Functional Profile, such as FSCS-21, have several objects. Since it is not possible to transmit more than one object data through a single connection, it is practical and more efficient to group attributes from different objects into a single I/O connection using the assembly object. The assembly object acts as a tool for grouping these attributes.

The assembly selections described above are, in fact, instances of the assembly object class. The adapter module uses static assemblies (in other words, fixed groupings of different object data only). The following tables describe the assembly instances supported by the adapter module.

Basic speed control assembly

The Basic speed control assembly is defined by the ODVA AC/DC drive profile. The format of the output assembly is:

Instance 20 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0						Fault Reset		Run Fwd
1								
2	Speed Reference (Low Byte)							
3	Speed Reference (High Byte)							

The format of the input assembly is:

Instance 70 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0						Running1 (Fwd)		Faulted
1								
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							

Basic speed control plus drive parameters assembly

The Basic speed control plus drive parameters assembly, defined by ABB, adds configurable drive parameters to the Basic speed control assembly of the ODVA AC/DC drive profile.

The format of the output assembly is:

Instance 120 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0						Fault re-set		Run Fwd
1								
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							
4	DATA OUT 1 Value (Low Byte)							
5	DATA OUT 1 Value (High Byte)							
6	DATA OUT 2 Value (Low Byte)							
7	DATA OUT 2 Value (High Byte)							
8	DATA OUT 3 Value (Low Byte)							
9	DATA OUT 3 Value (High Byte)							
10	DATA OUT 4 Value (Low Byte)							
11	DATA OUT 4 Value (High Byte)							
12	DATA OUT 5 Value (Low Byte)							
13	DATA OUT 5 Value (High Byte)							
14	DATA OUT 6 Value (Low Byte)							
15	DATA OUT 6 Value (High Byte)							
16	DATA OUT 7 Value (Low Byte)							
17	DATA OUT 7 Value (High Byte)							
18	DATA OUT 8 Value (Low Byte)							

Instance 120 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
19	DATA OUT 8 Value (High Byte)							
20	DATA OUT 9 Value (Low Byte)							
21	DATA OUT 9 Value (High Byte)							
22	DATA OUT 10 Value (Low Byte)							
23	DATA OUT 10 Value (High Byte)							

The format of the input assembly is:

Instance 170 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0						Run-ning1 (Fwd)		Faulted
1								
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							
4	DATA IN 1 Value (Low Byte)							
5	DATA IN 1 Value (High Byte)							
6	DATA IN 2 Value (Low Byte)							
7	DATA IN 2 Value (High Byte)							
8	DATA IN 3 Value (Low Byte)							
9	DATA IN 3 Value (High Byte)							
10	DATA IN 4 Value (Low Byte)							
11	DATA IN 4 Value (High Byte)							
12	DATA IN 5 Value (Low Byte)							
13	DATA IN 5 Value (High Byte)							
14	DATA IN 6 Value (Low Byte)							

Instance 170 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
15	DATA IN 6 Value (High Byte)							
16	DATA IN 7 Value (Low Byte)							
17	DATA IN 7 Value (High Byte)							
18	DATA IN 8 Value (Low Byte)							
19	DATA IN 8 Value (High Byte)							
20	DATA IN 9 Value (Low Byte)							
21	DATA IN 9 Value (High Byte)							
22	DATA IN 10 Value (Low Byte)							
23	DATA IN 10 Value (High Byte)							

Extended speed control assembly

The Extended speed control assembly is defined by the ODVA AC/DC drive profile. The format of the output assembly is:

Instance 21 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0		NetRef	NetCtrl			Fault Reset	Run Rev	Run Fwd
1								
2	Speed Reference (Low Byte)							
3	Speed Reference (High Byte)							

The format of the input assembly is:

Instance 71 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	At Reference	Ref From Net	Ctrl From Net	Ready	Running2 (Rev)	Running1 (Fwd)	Warning	Faulted

Instance 71 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
1	Drive State. See section State (Control supervisor object) (page 97).							
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							

Extended speed control plus drive parameters assembly

The Extended speed control plus drive parameters assembly, defined by ABB, adds configurable drive parameters to the Extended speed control assembly of the ODVA AC/DC drive profile.

The format of the output assembly is:

Instance 121 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0		Net Ref	Net Ctrl			Fault re-set	Run Rev	Run Fwd
1								
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							
4	DATA OUT 1 Value (Low Byte)							
5	DATA OUT 1 Value (High Byte)							
6	DATA OUT 2 Value (Low Byte)							
7	DATA OUT 2 Value (High Byte)							
8	DATA OUT 3 Value (Low Byte)							
9	DATA OUT 3 Value (High Byte)							
10	DATA OUT 4 Value (Low Byte)							
11	DATA OUT 4 Value (High Byte)							
12	DATA OUT 5 Value (Low Byte)							

Instance 121 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
13	DATA OUT 5 Value (High Byte)							
14	DATA OUT 6 Value (Low Byte)							
15	DATA OUT 6 Value (High Byte)							
16	DATA OUT 7 Value (Low Byte)							
17	DATA OUT 7 Value (High Byte)							
18	DATA OUT 8 Value (Low Byte)							
19	DATA OUT 8 Value (High Byte)							
20	DATA OUT 9 Value (Low Byte)							
21	DATA OUT 9 Value (High Byte)							
22	DATA OUT 10 Value (Low Byte)							
23	DATA OUT 10 Value (High Byte)							

The format of the input assembly is:

Instance 171 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	At Reference	Ref From Net	Ctrl From Net	Ready	Running2 (Rev)	Running1 (Fwd)	Warning	Faulted
1	Drive State See section State (Control supervisor object) (page 97).							
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							
4	DATA IN 1 Value (Low Byte)							
5	DATA IN 1 Value (High Byte)							
6	DATA IN 2 Value (Low Byte)							
7	DATA IN 2 Value (High Byte)							

Instance 171 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
8	DATA IN 3 Value (Low Byte)							
9	DATA IN 3 Value (High Byte)							
10	DATA IN 4 Value (Low Byte)							
11	DATA IN 4 Value (High Byte)							
12	DATA IN 5 Value (Low Byte)							
13	DATA IN 5 Value (High Byte)							
14	DATA IN 6 Value (Low Byte)							
15	DATA IN 6 Value (High Byte)							
16	DATA IN 7 Value (Low Byte)							
17	DATA IN 7 Value (High Byte)							
18	DATA IN 8 Value (Low Byte)							
19	DATA IN 8 Value (High Byte)							
20	DATA IN 9 Value (Low Byte)							
21	DATA IN 9 Value (High Byte)							
22	DATA IN 10 Value (Low Byte)							
23	DATA IN 10 Value (High Byte)							

Basic speed and torque control assembly

The Basic speed and torque control assembly is defined by the ODVA AC/DC drive profile. The format of the output assembly is:

Instance 22 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0						Fault Reset		Run Fwd
1								

Instance 22 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
2	Speed Reference (Low Byte)							
3	Speed Reference (High Byte)							
4	Torque Reference (Low Byte)							
5	Torque Reference (High Byte)							

The format of the input assembly is:

Instance 72 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0						Run-ning1 (Fwd)		Faulted
1								
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							
4	Torque Actual (Low Byte)							
5	Torque Actual (High Byte)							

Basic speed and torque control plus drive parameters assembly

The Basic speed and torque control plus drive parameters assembly, defined by ABB, adds configurable drive parameters to the Basic speed and torque control assembly of the ODVA AC/DC drive profile.

The format of the output assembly is:

Instance 122 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0						Fault re-set		Run Fwd
1								
2	Speed Reference (Low Byte)							

Instance 122 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
3	Speed Reference (High Byte)							
4	Torque Reference (Low Byte)							
5	Torque Reference (High Byte)							
6	DATA OUT 1 Value (Low Byte)							
7	DATA OUT 1 Value (High Byte)							
8	DATA OUT 2 Value (Low Byte)							
9	DATA OUT 2 Value (High Byte)							
10	DATA OUT 3 Value (Low Byte)							
11	DATA OUT 3 Value (High Byte)							
12	DATA OUT 4 Value (Low Byte)							
13	DATA OUT 4 Value (High Byte)							
14	DATA OUT 5 Value (Low Byte)							
15	DATA OUT 5 Value (High Byte)							
16	DATA OUT 6 Value (Low Byte)							
17	DATA OUT 6 Value (High Byte)							
18	DATA OUT 7 Value (Low Byte)							
19	DATA OUT 7 Value (High Byte)							
20	DATA OUT 8 Value (Low Byte)							
21	DATA OUT 8 Value (High Byte)							
22	DATA OUT 9 Value (Low Byte)							
23	DATA OUT 9 Value (High Byte)							
24	DATA OUT 10 Value (Low Byte)							
25	DATA OUT 10 Value (High Byte)							

The format of the input assembly is:

Instance 172 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0						Run-ning1 (Fwd)		Faulted
1								
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							
4	Torque Actual (Low Byte)							
5	Torque Actual (High Byte)							
6	DATA IN 1 Value (Low Byte)							
7	DATA IN 1 Value (High Byte)							
8	DATA IN 2 Value (Low Byte)							
9	DATA IN 2 Value (High Byte)							
10	DATA IN 3 Value (Low Byte)							
11	DATA IN 3 Value (High Byte)							
12	DATA IN 4 Value (Low Byte)							
13	DATA IN 4 Value (High Byte)							
14	DATA IN 5 Value (Low Byte)							
15	DATA IN 5 Value (High Byte)							
16	DATA IN 6 Value (Low Byte)							
17	DATA IN 6 Value (High Byte)							
18	DATA IN 7 Value (Low Byte)							
19	DATA IN 7 Value (High Byte)							
20	DATA IN 8 Value (Low Byte)							
21	DATA IN 8 Value (High Byte)							
22	DATA IN 9 Value (Low Byte)							

Instance 172 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
23	DATA IN 9 Value (High Byte)							
24	DATA IN 10 Value (Low Byte)							
25	DATA IN 10 Value (High Byte)							

Extended speed and torque control assembly

The Extended speed and torque control assembly is defined by the ODVA AC/DC drive profile. The format of the output assembly is:

Instance 23 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0		Net Ref	Net Ctrl			Fault Reset	Run Rev	Run Fwd
1								
2	Speed Reference (Low Byte)							
3	Speed Reference (High Byte)							
4	Torque Reference (Low Byte)							
5	Torque Reference (High Byte)							

The format of the input assembly is:

Instance 73 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	At Reference	Ref From Net	Ctrl From Net	Ready	Running2 (Rev)	Running1 (Fwd)	Warning	Faulted
1	Drive State See section State (Control supervisor object) (page 97).							
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							

Instance 73 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
4	Torque Actual (Low Byte)							
5	Torque Actual (High Byte)							

Extended speed and torque control plus drive parameters assembly

The Extended speed and torque control plus drive parameters assembly, defined by ABB, adds configurable drive parameters to the Extended speed and torque control assembly of the ODVA AC/DC drive profile.

The format of the output assembly is:

Instance 123 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0		NetRef	NetCtl			Fault reset	Run Rev	Run Fwd
1								
2	Speed Reference (Low Byte)							
3	Speed Reference (High Byte)							
4	Torque Reference (Low Byte)							
5	Torque Reference (High Byte)							
6	DATA OUT 1 Value (Low Byte)							
7	DATA OUT 1 Value (High Byte)							
8	DATA OUT 2 Value (Low Byte)							
9	DATA OUT 2 Value (High Byte)							
10	DATA OUT 3 Value (Low Byte)							
11	DATA OUT 3 Value (High Byte)							
12	DATA OUT 4 Value (Low Byte)							
13	DATA OUT 4 Value (High Byte)							
14	DATA OUT 5 Value (Low Byte)							

Instance 123 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
15	DATA OUT 5 Value (High Byte)							
16	DATA OUT 6 Value (Low Byte)							
17	DATA OUT 6 Value (High Byte)							
18	DATA OUT 7 Value (Low Byte)							
19	DATA OUT 7 Value (High Byte)							
20	DATA OUT 8 Value (Low Byte)							
21	DATA OUT 8 Value (High Byte)							
22	DATA OUT 9 Value (Low Byte)							
23	DATA OUT 9 Value (High Byte)							
24	DATA OUT 10 Value (Low Byte)							
25	DATA OUT 10 Value (High Byte)							

The format of the input assembly is:

Instance 173 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	At Reference	Ref From Net	Ctrl From Net	Ready	Running2 (Rev)	Running1 (Fwd)	Warning	Faulted
1	Drive State See section State (Control supervisor object) (page 97).							
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							
4	Torque Actual (Low Byte)							
5	Torque Actual (High Byte)							
6	DATA IN 1 Value (Low Byte)							
7	DATA IN 1 Value (High Byte)							

Instance 173 (ODVA AC/DC profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
8	DATA IN 2 Value (Low Byte)							
9	DATA IN 2 Value (High Byte)							
10	DATA IN 3 Value (Low Byte)							
11	DATA IN 3 Value (High Byte)							
12	DATA IN 4 Value (Low Byte)							
13	DATA IN 4 Value (High Byte)							
14	DATA IN 5 Value (Low Byte)							
15	DATA IN 5 Value (High Byte)							
16	DATA IN 6 Value (Low Byte)							
17	DATA IN 6 Value (High Byte)							
18	DATA IN 7 Value (Low Byte)							
19	DATA IN 7 Value (High Byte)							
20	DATA IN 8 Value (Low Byte)							
21	DATA IN 8 Value (High Byte)							
22	DATA IN 9 Value (Low Byte)							
23	DATA IN 9 Value (High Byte)							
24	DATA IN 10 Value (Low Byte)							
25	DATA IN 10 Value (High Byte)							

ABB Drives profile with set speed assembly

The ABB Drives profile with set speed assembly is defined by ABB. The format of the output assembly is:

Instance 1 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Reset	Ramp in Zero	Ramp Hold	Ramp Out Zero	Inhibit Operation	Off 3 Control	Off 2 Control	Off 1 Control
1					Ext Ctrl Loc	Remote Cmd		
2	Set Speed (Low Byte)							
3	Set Speed (High Byte)							

The format of the input assembly is:

Instance 51 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Alarm	Swc On Inhib	Off 3 Sta	Off 2 Sta	Tripped	Rdy Ref	Rdy Run	Rdy On
1	Fieldbus Error			Ext Run Enable	Ext Ctrl Loc	Above Limit	Remote	At Set-point
2	Actual Speed (Low Byte)							
3	Actual Speed (High Byte)							

ABB Drives profile with set speed plus drive parameters assembly

The ABB Drives profile with set speed plus drive parameters assembly, defined by ABB, adds configurable drive parameters to the ABB Drives profile with set speed of the ABB Drives profile.

The format of the output assembly is:

Instance 101 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Reset	Ramp in Zero	Ramp Hold	Ramp Out Zero	Inhibit Operation	Off 3 Control	Off 2 Control	Off 1 Control
1					Ext Ctrl Loc	Remote Cmd		

Instance 101 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
2	Set Speed (Low Byte)							
3	Set Speed (High Byte)							
4	DATA OUT 1 Value (Low Byte)							
5	DATA OUT 1 Value (High Byte)							
6	DATA OUT 2 Value (Low Byte)							
7	DATA OUT 2 Value (High Byte)							
8	DATA OUT 3 Value (Low Byte)							
9	DATA OUT 3 Value (High Byte)							
10	DATA OUT 4 Value (Low Byte)							
11	DATA OUT 4 Value (High Byte)							
12	DATA OUT 5 Value (Low Byte)							
13	DATA OUT 5 Value (High Byte)							
14	DATA OUT 6 Value (Low Byte)							
15	DATA OUT 6 Value (High Byte)							
16	DATA OUT 7 Value (Low Byte)							
17	DATA OUT 7 Value (High Byte)							
18	DATA OUT 8 Value (Low Byte)							
19	DATA OUT 8 Value (High Byte)							
20	DATA OUT 9 Value (Low Byte)							
21	DATA OUT 9 Value (High Byte)							
22	DATA OUT 10 Value (Low Byte)							
23	DATA OUT 10 Value (High Byte)							

The format of the input assembly is:

Instance 151 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Alarm	Swc On Inhib	Off 3 Sta	Off 2 Sta	Tripped	Rdy Ref	Rdy Run	Rdy On
1	Fieldbus Error			Ext Run Enable	Ext Ctrl Loc	Above Limit	Remote	At Set-point
2	Actual Speed (Low Byte)							
3	Actual Speed (High Byte)							
4	DATA IN 1 Value (Low Byte)							
5	DATA IN 1 Value (High Byte)							
6	DATA IN 2 Value (Low Byte)							
7	DATA IN 2 Value (High Byte)							
8	DATA IN 3 Value (Low Byte)							
9	DATA IN 3 Value (High Byte)							
10	DATA IN 4 Value (Low Byte)							
11	DATA IN 4 Value (High Byte)							
12	DATA IN 5 Value (Low Byte)							
13	DATA IN 5 Value (High Byte)							
14	DATA IN 6 Value (Low Byte)							
15	DATA IN 6 Value (High Byte)							
16	DATA IN 7 Value (Low Byte)							
17	DATA IN 7 Value (High Byte)							
18	DATA IN 8 Value (Low Byte)							
19	DATA IN 8 Value (High Byte)							
20	DATA IN 9 Value (Low Byte)							
21	DATA IN 9 Value (High Byte)							
22	DATA IN 10 Value (Low Byte)							

Instance 151 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
23	DATA IN 10 Value (High Byte)							

ABB Drives profile with set speed and set torque assembly

The ABB Drives profile with set speed and set torque assembly is defined by ABB. The format of the output assembly is:

Instance 2 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Reset	Ramp in Zero	Ramp Hold	Ramp Out Zero	Inhibit Opera-tion	Off 3 Control	Off 2 Control	Off 1 Control
1					Ext Ctrl Loc	Remote Cmd		
2	Set Speed (Low Byte)							
3	Set Speed (High Byte)							
4	Set Torque (Low Byte)							
5	Set Torque (High Byte)							

The format of the input assembly is:

Instance 52 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Alarm	Swc On Inhib	Off 3 Sta	Off 2 Sta	Tripped	Rdy Ref	Rdy Run	Rdy On
1	Fieldbus Error			Ext Run Enable	Ext Ctrl Loc	Above Limit	Remote	At Set-point
2	Actual Speed (Low Byte)							
3	Actual Speed (High Byte)							
4	Actual Torque (Low Byte)							
5	Actual Torque (High Byte)							

ABB Drives profile with set speed and set torque plus drive parameters assembly

The ABB Drives profile with set speed and set torque plus drive parameters assembly, defined by ABB, adds configurable drive parameters to the ABB Drives profile with set speed and set torque of the ABB Drives profile.

The format of the output assembly is:

Instance 102 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Reset	Ramp in Zero	Ramp Hold	Ramp Out Zero	Inhibit Opera-tion	Off 3 Control	Off 2 Control	Off 1 Control
1					Ext Ctrl Loc	Remote Cmd		
2	Set Speed (Low Byte)							
3	Set Speed (High Byte)							
4	Set Torque (Low Byte)							
5	Set Torque (High Byte)							
6	DATA OUT 1 Value (Low Byte)							
7	DATA OUT 1 Value (High Byte)							
8	DATA OUT 2 Value (Low Byte)							
9	DATA OUT 2 Value (High Byte)							
10	DATA OUT 3 Value (Low Byte)							
11	DATA OUT 3 Value (High Byte)							
12	DATA OUT 4 Value (Low Byte)							
13	DATA OUT 4 Value (High Byte)							
14	DATA OUT 5 Value (Low Byte)							
15	DATA OUT 5 Value (High Byte)							
16	DATA OUT 6 Value (Low Byte)							
17	DATA OUT 6 Value (High Byte)							

Instance 102 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
18	DATA OUT 7 Value (Low Byte)							
19	DATA OUT 7 Value (High Byte)							
20	DATA OUT 8 Value (Low Byte)							
21	DATA OUT 8 Value (High Byte)							
22	DATA OUT 9 Value (Low Byte)							
23	DATA OUT 9 Value (High Byte)							
24	DATA OUT 10 Value (Low Byte)							
25	DATA OUT 10 Value (High Byte)							

The format of the input assembly is:

Instance 152 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Alarm	Swc On Inhib	Off 3 Sta	Off 2 Sta	Tripped	Rdy Ref	Rdy Run	Rdy On
1	Fieldbus Error			Ext Run Enable	Ext Ctrl Loc	Above Limit	Remote	At Set-point
2	Actual Speed (Low Byte)							
3	Actual Speed (High Byte)							
4	Actual Torque (Low Byte)							
5	Actual Torque (High Byte)							
6	DATA IN 1 Value (Low Byte)							
7	DATA IN 1 Value (High Byte)							
8	DATA IN 2 Value (Low Byte)							
9	DATA IN 2 Value (High Byte)							
10	DATA IN 3 Value (Low Byte)							
11	DATA IN 3 Value (High Byte)							

Instance 152 (ABB Drives profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
12	DATA IN 4 Value (Low Byte)							
13	DATA IN 4 Value (High Byte)							
14	DATA IN 5 Value (Low Byte)							
15	DATA IN 5 Value (High Byte)							
16	DATA IN 6 Value (Low Byte)							
17	DATA IN 6 Value (High Byte)							
18	DATA IN 7 Value (Low Byte)							
19	DATA IN 7 Value (High Byte)							
20	DATA IN 8 Value (Low Byte)							
21	DATA IN 8 Value (High Byte)							
22	DATA IN 9 Value (Low Byte)							
23	DATA IN 9 Value (High Byte)							
24	DATA IN 10 Value (Low Byte)							
25	DATA IN 10 Value (High Byte)							

Transparent 16 with one assembly

The Transparent 16 with one assembly, defined by ABB, provides unaltered 16-bit access to the configured drive profile.

The format of the output assembly is:

Instance 11 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 16-bit Control Word (Low Byte)							
1	Drive Profile 16-bit Control Word (High Byte)							
2	Drive Profile 16-bit Reference 1 Word (Low Byte)							
3	Drive Profile 16-bit Reference 1 Word (High Byte)							

The format of the input assembly is:

Instance 61 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 16-bit Status Word (Low Byte)							
1	Drive Profile 16-bit Status Word (High Byte)							
2	Drive Profile 16-bit Actual 1 Word (Low Byte)							
3	Drive Profile 16-bit Actual 1 Word (High Byte)							

Transparent 16 with one assembly plus drive parameters

The Transparent 16 with one assembly plus drive parameters, defined by ABB, adds configurable drive parameters to the Transparent 16 with one assembly.

The format of the output assembly is:

Instance 111 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 16-bit Control Word (Low Byte)							
1	Drive Profile 16-bit Control Word (High Byte)							
2	Drive Profile 16-bit Reference 1 Word (Low Byte)							
3	Drive Profile 16-bit Reference 1 Word (High Byte)							
4	DATA OUT 1 Value (Low Byte)							
5	DATA OUT 1 Value (High Byte)							
6	DATA OUT 2 Value (Low Byte)							
7	DATA OUT 2 Value (High Byte)							
8	DATA OUT 3 Value (Low Byte)							
9	DATA OUT 3 Value (High Byte)							
10	DATA OUT 4 Value (Low Byte)							
11	DATA OUT 4 Value (High Byte)							
12	DATA OUT 5 Value (Low Byte)							

Instance 111 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
13	DATA OUT 5 Value (High Byte)							
14	DATA OUT 6 Value (Low Byte)							
15	DATA OUT 6 Value (High Byte)							
16	DATA OUT 7 Value (Low Byte)							
17	DATA OUT 7 Value (High Byte)							
18	DATA OUT 8 Value (Low Byte)							
19	DATA OUT 8 Value (High Byte)							
20	DATA OUT 9 Value (Low Byte)							
21	DATA OUT 9 Value (High Byte)							
22	DATA OUT 10 Value (Low Byte)							
23	DATA OUT 10 Value (High Byte)							

The format of the input assembly is:

Instance 161 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 16-bit Status Word (Low Byte)							
1	Drive Profile 16-bit Status Word (High Byte)							
2	Drive Profile 16-bit Actual 1 Word (Low Byte)							
3	Drive Profile 16-bit Actual 1 Word (High Byte)							
4	DATA IN 1 Value (Low Byte)							
5	DATA IN 1 Value (High Byte)							
6	DATA IN 2 Value (Low Byte)							
7	DATA IN 2 Value (High Byte)							
8	DATA IN 3 Value (Low Byte)							
9	DATA IN 3 Value (High Byte)							

Instance 161 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
10	DATA IN 4 Value (Low Byte)							
11	DATA IN 4 Value (High Byte)							
12	DATA IN 5 Value (Low Byte)							
13	DATA IN 5 Value (High Byte)							
14	DATA IN 6 Value (Low Byte)							
15	DATA IN 6 Value (High Byte)							
16	DATA IN 7 Value (Low Byte)							
17	DATA IN 7 Value (High Byte)							
18	DATA IN 8 Value (Low Byte)							
19	DATA IN 8 Value (High Byte)							
20	DATA IN 9 Value (Low Byte)							
21	DATA IN 9 Value (High Byte)							
22	DATA IN 10 Value (Low Byte)							
23	DATA IN 10 Value (High Byte)							

Transparent 16 with two assembly

The Transparent 16 with two assembly, defined by ABB, provides unaltered 16-bit access to the configured drive profile.

The format of the output assembly is:

Instance 12 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 16-bit Control Word (Low Byte)							
1	Drive Profile 16-bit Control Word (High Byte)							
2	Drive Profile 16-bit Reference 1 Word (Low Byte)							
3	Drive Profile 16-bit Reference 1 Word (High Byte)							

Instance 12 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
4	Drive Profile 16-bit Reference 2 Word (Low Byte)							
5	Drive Profile 16-bit Reference 2 Word (High Byte)							

The format of the input assembly is:

Instance 62 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 16-bit Status Word (Low Byte)							
1	Drive Profile 16-bit Status Word (High Byte)							
2	Drive Profile 16-bit Actual 1 Word (Low Byte)							
3	Drive Profile 16-bit Actual 1 Word (High Byte)							
4	Drive Profile16-bit Actual 2 Word (Low Byte)							
5	Drive Profile 16-bit Actual 2 Word (High Byte)							

Transparent 16 with two assembly plus drive parameters

The Transparent 16 with two assembly plus drive parameters, defined by ABB, adds configurable drive parameters to the Transparent 16 with two assembly.

The format of the output assembly is:

Instance 112 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 16-bit Control Word (Low Byte)							
1	Drive Profile 16-bit Control Word (High Byte)							
2	Drive Profile 16-bit Reference 1 Word (Low Byte)							
3	Drive Profile 16-bit Reference 1 Word (High Byte)							
4	Drive Profile 16-bit Reference 2 Word (Low Byte)							
5	Drive Profile 16-bit Reference 2 Word (High Byte)							
6	DATA OUT 1 Value (Low Byte)							

Instance 112 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
7	DATA OUT 1 Value (High Byte)							
8	DATA OUT 2 Value (Low Byte)							
9	DATA OUT 2 Value (High Byte)							
10	DATA OUT 3 Value (Low Byte)							
11	DATA OUT 3 Value (High Byte)							
12	DATA OUT 4 Value (Low Byte)							
13	DATA OUT 4 Value (High Byte)							
14	DATA OUT 5 Value (Low Byte)							
15	DATA OUT 5 Value (High Byte)							
16	DATA OUT 6 Value (Low Byte)							
17	DATA OUT 6 Value (High Byte)							
18	DATA OUT 7 Value (Low Byte)							
19	DATA OUT 7 Value (High Byte)							
20	DATA OUT 8 Value (Low Byte)							
21	DATA OUT 8 Value (High Byte)							
22	DATA OUT 9 Value (Low Byte)							
23	DATA OUT 9 Value (High Byte)							
24	DATA OUT 10 Value (Low Byte)							
25	DATA OUT 10 Value (High Byte)							

Instance 162 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 16-bit Status Word (Low Byte)							
1	Drive Profile 16-bit Status Word (High Byte)							

Instance 162 (Transparent 16 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
2	Drive Profile 16-bit Actual 1 Word (Low Byte)							
3	Drive Profile 16-bit Actual 1 Word (High Byte)							
4	Drive Profile 16-bit Actual 2 Word (Low Byte)							
5	Drive Profile 16-bit Actual 2 Word (High Byte)							
6	DATA IN 1 Value (Low Byte)							
7	DATA IN 1 Value (High Byte)							
8	DATA IN 2 Value (Low Byte)							
9	DATA IN 2 Value (High Byte)							
10	DATA IN 3 Value (Low Byte)							
11	DATA IN 3 Value (High Byte)							
12	DATA IN 4 Value (Low Byte)							
13	DATA IN 4 Value (High Byte)							
14	DATA IN 5 Value (Low Byte)							
15	DATA IN 5 Value (High Byte)							
16	DATA IN 6 Value (Low Byte)							
17	DATA IN 6 Value (High Byte)							
18	DATA IN 7 Value (Low Byte)							
19	DATA IN 7 Value (High Byte)							
20	DATA IN 8 Value (Low Byte)							
21	DATA IN 8 Value (High Byte)							
22	DATA IN 9 Value (Low Byte)							
23	DATA IN 9 Value (High Byte)							
24	DATA IN 10 Value (Low Byte)							
25	DATA IN 10 Value (High Byte)							

Transparent 32 with one assembly

The Transparent 32 with one assembly, defined by ABB, provides unaltered 32-bit access to the configured drive profile.

The format of the output assembly is:

Instance 41 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 32-bit Control Word (Low Byte)							
1	Drive Profile 32-bit Control Word							
2	Drive Profile 32-bit Control Word							
3	Drive Profile 32-bit Control Word (High Byte)							
4	Drive Profile 32-bit Reference 1 Word (Low Byte)							
5	Drive Profile 32-bit Reference 1 Word							
6	Drive Profile 32-bit Reference 1 Word							
7	Drive Profile 32-bit Reference 1 Word (High Byte)							

The format of the input assembly is:

Instance 91 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 32-bit Status Word (Low Byte)							
1	Drive Profile 32-bit Status Word							
2	Drive Profile 32-bit Status Word							
3	Drive Profile 32-bit Status Word (High Byte)							
4	Drive Profile 32-bit Actual 1 Word (Low Byte)							
5	Drive Profile 32-bit Actual 1 Word							
6	Drive Profile 32-bit Actual 1 Word							
7	Drive Profile 32-bit Actual 1 Word (High Byte)							

Transparent 32 with one assembly plus drive parameters

The Transparent 32 with one assembly plus drive parameters, defined by ABB, adds configurable drive parameters to the Transparent 32 with one assembly.

The format of the output assembly is:

Instance 141 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 32-bit Control Word (Low Byte)							
1	Drive Profile 32-bit Control Word							
2	Drive Profile 32-bit Control Word							
3	Drive Profile 32-bit Control Word (High Byte)							
4	Drive Profile 32-bit Reference 1 Word (Low Byte)							
5	Drive Profile 32-bit Reference 1 Word							
6	Drive Profile 32-bit Reference 1 Word							
7	Drive Profile 32-bit Reference 1 Word (High Byte)							
8	DATA OUT 1 Value (Low Byte)							
9	DATA OUT 1 Value (High Byte)							
10	DATA OUT 2 Value (Low Byte)							
11	DATA OUT 2 Value (High Byte)							
12	DATA OUT 3 Value (Low Byte)							
13	DATA OUT 3 Value (High Byte)							
14	DATA OUT 4 Value (Low Byte)							
15	DATA OUT 4 Value (High Byte)							
16	DATA OUT 5 Value (Low Byte)							
17	DATA OUT 5 Value (High Byte)							
18	DATA OUT 6 Value (Low Byte)							
19	DATA OUT 6 Value (High Byte)							
20	DATA OUT 7 Value (Low Byte)							

Instance 141 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
21	DATA OUT 7 Value (High Byte)							
22	DATA OUT 8 Value (Low Byte)							
23	DATA OUT 8 Value (High Byte)							
24	DATA OUT 9 Value (Low Byte)							
25	DATA OUT 9 Value (High Byte)							
26	DATA OUT 10 Value (Low Byte)							
27	DATA OUT 10 Value (High Byte)							

The format of the input assembly is:

Instance 191 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 32-bit Status Word (Low Byte)							
1	Drive Profile 32-bit Status Word							
2	Drive Profile 32-bit Status Word							
3	Drive Profile 32-bit Status Word (High Byte)							
4	Drive Profile 32-bit Actual 1 Word (Low Byte)							
5	Drive Profile 32-bit Actual 1 Word (High Byte)							
6	Drive Profile 32-bit Actual 1 Word							
7	Drive Profile 32-bit Actual 1 Word (High Byte)							
8	DATA IN 1 Value (Low Byte)							
9	DATA IN 1 Value (High Byte)							
10	DATA IN 2 Value (Low Byte)							
11	DATA IN 2 Value (High Byte)							
12	DATA IN 3 Value (Low Byte)							
13	DATA IN 3 Value (High Byte)							

Instance 191 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
14	DATA IN 4 Value (Low Byte)							
15	DATA IN 4 Value (High Byte)							
16	DATA IN 5 Value (Low Byte)							
17	DATA IN 5 Value (High Byte)							
18	DATA IN 6 Value (Low Byte)							
19	DATA IN 6 Value (High Byte)							
20	DATA IN 7 Value (Low Byte)							
21	DATA IN 7 Value (High Byte)							
22	DATA IN 8 Value (Low Byte)							
23	DATA IN 8 Value (High Byte)							
24	DATA IN 9 Value (Low Byte)							
25	DATA IN 9 Value (High Byte)							
26	DATA IN 10 Value (Low Byte)							
27	DATA IN 10 Value (High Byte)							

Transparent 32 with two assembly

The Transparent 32 with two assembly, defined by ABB, provides unaltered 32-bit access to the configured drive profile.

The format of the output assembly is:

Instance 42 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 32-bit Control Word (Low Byte)							
1	Drive Profile 32-bit Control Word							
2	Drive Profile 32-bit Control Word							
3	Drive Profile 32-bit Control Word (High Byte)							

Instance 42 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
4	Drive Profile 32-bit Reference 1 Word (Low Byte)							
5	Drive Profile 32-bit Reference 1 Word							
6	Drive Profile 32-bit Reference 1 Word							
7	Drive Profile 32-bit Reference 1 Word (High Byte)							
8	Drive Profile 32-bit Reference 2 Word (Low Byte)							
9	Drive Profile 32-bit Reference 2 Word							
10	Drive Profile 32-bit Reference 2 Word							
11	Drive Profile 32-bit Reference 2 Word (High Byte)							

The format of the input assembly is:

Instance 92 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 32-bit Status Word (Low Byte)							
1	Drive Profile 32-bit Status Word							
2	Drive Profile 32-bit Status Word							
3	Drive Profile 32-bit Status Word (High Byte)							
4	Drive Profile 32-bit Actual 1 Word (Low Byte)							
5	Drive Profile 32-bit Actual 1 Word							
6	Drive Profile 32-bit Actual 1 Word							
7	Drive Profile 32-bit Actual 1 Word (High Byte)							
8	Drive Profile 32-bit Actual 2 Word (Low Byte)							
9	Drive Profile 32-bit Actual 2 Word							
10	Drive Profile 32-bit Actual 2 Word							
11	Drive Profile 32-bit Actual 2 Word (High Byte)							

Transparent 32 with two assembly plus drive parameters

The Transparent 32 with two assembly plus drive parameters, defined by ABB, adds configurable drive parameters to the Transparent 32 with two assembly.

The format of the output assembly is:

Instance 142 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 32-bit Control Word (Low Byte)							
1	Drive Profile 32-bit Control Word							
2	Drive Profile 32-bit Control Word							
3	Drive Profile 32-bit Control Word (High Byte)							
4	Drive Profile 32-bit Reference 1 Word (Low Byte)							
5	Drive Profile 32-bit Reference 1 Word							
6	Drive Profile 32-bit Reference 1 Word							
7	Drive Profile 32-bit Reference 1 Word (High Byte)							
8	Drive Profile 32-bit Reference 2 Word (Low Byte)							
9	Drive Profile 32-bit Reference 2 Word							
10	Drive Profile 32-bit Reference 2 Word							
11	Drive Profile 32-bit Reference 2 Word (High Byte)							
12	DATA OUT 1 Value (Low Byte)							
13	DATA OUT 1 Value (High Byte)							
14	DATA OUT 2 Value (Low Byte)							
15	DATA OUT 2 Value (High Byte)							
16	DATA OUT 3 Value (Low Byte)							
17	DATA OUT 3 Value (High Byte)							
18	DATA OUT 4 Value (Low Byte)							
19	DATA OUT 4 Value (High Byte)							
20	DATA OUT 5 Value (Low Byte)							

Instance 142 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
21	DATA OUT 5 Value (High Byte)							
22	DATA OUT 6 Value (Low Byte)							
23	DATA OUT 6 Value (High Byte)							
24	DATA OUT 7 Value (Low Byte)							
25	DATA OUT 7 Value (High Byte)							
26	DATA OUT 8 Value (Low Byte)							
27	DATA OUT 8 Value (High Byte)							
28	DATA OUT 9 Value (Low Byte)							
29	DATA OUT 9 Value (High Byte)							
30	DATA OUT 10 Value (Low Byte)							
31	DATA OUT 10 Value (High Byte)							

The format of the input assembly is:

Instance 192 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Drive Profile 32-bit Status Word (Low Byte)							
1	Drive Profile 32-bit Status Word							
2	Drive Profile 32-bit Status Word							
3	Drive Profile 32-bit Status Word (High Byte)							
4	Drive Profile 32-bit Actual 1 Word (Low Byte)							
5	Drive Profile 32-bit Actual 1 Word							
6	Drive Profile 32-bit Actual 1 Word							
7	Drive Profile 32-bit Actual 1 Word (High Byte)							
8	Drive Profile 32-bit Actual 2 Word (Low Byte)							
9	Drive Profile 32-bit Actual 2 Word							

Instance 192 (Transparent 32 profile)								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
10	Drive Profile 32-bit Actual 2 Word							
11	Drive Profile 32-bit Actual 2 Word (High Byte)							
12	DATA IN 1 Value (Low Byte)							
13	DATA IN 1 Value (High Byte)							
14	DATA IN 2 Value (Low Byte)							
15	DATA IN 2 Value (High Byte)							
16	DATA IN 3 Value (Low Byte)							
17	DATA IN 3 Value (High Byte)							
18	DATA IN 4 Value (Low Byte)							
19	DATA IN 4 Value (High Byte)							
20	DATA IN 5 Value (Low Byte)							
21	DATA IN 5 Value (High Byte)							
22	DATA IN 6 Value (Low Byte)							
23	DATA IN 6 Value (High Byte)							
24	DATA IN 7 Value (Low Byte)							
25	DATA IN 7 Value (High Byte)							
26	DATA IN 8 Value (Low Byte)							
27	DATA IN 8 Value (High Byte)							
28	DATA IN 9 Value (Low Byte)							
29	DATA IN 9 Value (High Byte)							
30	DATA IN 10 Value (Low Byte)							
31	DATA IN 10 Value (High Byte)							

Class objects

The following table lists the data types used in the class object descriptions of this manual.

Legend	Data type
UINT8	Unsigned Integer 8 bit
UINT16	Unsigned Integer 16 bit
SINT16	Signed Integer 16 bit
UINT32	Unsigned Integer 32 bit
BOOL	Boolean value
BYTE	Bit String 8 bit
WORD	Bit String 16 bit
DWORD	Bit String 32 bit

Note: The adapter module is designed to provide EtherNet/IP communications for a variety of drives with different capabilities. Default, minimum and maximum values for attributes necessarily vary based upon the capabilities of the drive to which the module is attached and are not documented herein. Default, minimum and maximum values for attributes may be found in the:

- drive manuals
- Electronic Data Sheet Files (EDS) for the drive.

Note that the units of the attributes may differ from those of the parameters documented elsewhere, and those differences must be considered when interfacing to the drive via the module.

The table below shows the service names of the class objects.

Service	Name
GET	0x0E Get_Attribute_Single
SET	0x10 Set_Attribute_Single
SET ALL	0x02 Set_Attribute_All
GET ALL	0x01 Get_Attribute_All

Identity object, class 01h

This object provides identification of and general information about the device.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of the identity object	Array of UINT8

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
1	Vendor ID	Get	Identification of the device vendor	UINT16
2	Device Type	Get	Identification of the general product type	UINT16
3	Product Code	Get	Assigned vendor code to describe the device	UINT16
4	Revision	Get	Revision of the item the identity object represents	Array [UINT8 UINT8]
5	Status	Get	Summary status of the device	UINT16
6	ODVA Serial Number	Get	Serial number of the EtherNet/IP module	UINT32
7	Product Name	Get	Product identification. Max 32 characters.	Short String

Attribute explanations

Vendor ID

Vendor IDs are managed by the Open DeviceNet Vendor Association, Inc. (ODVA). The ABB Vendor ID is 46.

Device Type

FSCS-21 Device Type is 102.

Product Code

FSCS-21 Product Code is 1.

Revision

Revision attribute, which consists of Major and Minor Revisions, identifies the revision of the item the identity object represents.

Status

This attribute represents the current status of the entire device. Its value changes as the state of the device changes. The Status attribute is a WORD, with the following bit definitions:

Bit(s)	Type/ Name	Definition
0	Owned	TRUE indicates the device (or an object within the device) has an owner. Within the Master/Slave paradigm the setting of this bit means that the Predefined Master/Slave Connection Set has been allocated to a master. Outside the Master/Slave paradigm the meaning of this bit is to be defined.
1		Reserved, set to 0
2	Configured	TRUE indicates that the application of the device has been configured to do something that differs from the “out–of–box” default. This does not include configuration of the communications.
3		Reserved, set to 0
4 … 7		Vendor-specific
8	Minor Recoverable Fault	TRUE indicates the device detected a recoverable problem. The problem does not cause the device fault state.
9	Minor Unrecoverable Fault	TRUE indicates the device detected a unrecoverable problem. The problem does not cause the device fault state.
10	Major Recoverable Fault	TRUE indicates the device detected a problem which caused the device to transfer into the “Major Recoverable Fault” state.
11	Major Unrecoverable Fault	TRUE indicates the device detected a problem which caused the device to transfer into the “Major Unrecoverable Fault” state.
12 … 15		Reserved, set to 0

ODVA Serial Number

This attribute is a number used in conjunction with the Vendor ID to form a unique identifier for each device on EtherNet/IP. The value of this attribute is 02000000h plus the SERNO value from the device label.

Product Name

This text string should represent a short description of the product/product family represented by the product code in attribute 3.

Motor data object, class 28h

The Motor data object can only be used if the ODVA AC/DC drive profile is in use.

The object serves as a database for motor parameters.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of the CIP Object Class Definition upon which the implementation is based	Array of UINT8

Instance attributes (Instance #1)

Different motor types require different data. The table below shows the data applicable to different motor types.

#	Attribute name	Services	Description	Motor type	Data type
3	Motor type	Get	Refer to section Attribute explanations (page 147).	AC or DC	UINT8
6	Rated Current	Get, Set	Rated Stator Current from motor name plate Units: [100mA]	AC or DC	UINT16
7	Rated Voltage	Get, Set	Rated Base Voltage from motor name plate Units: [V]	AC or DC	UINT16

#	Attribute name	Services	Description	Motor type	Data type
8	Rated Power	Get, Set	Rated Power at Rated Frequency Units: [W]	AC or DC	UINT32
9	Rated frequency	Get, Set	Rated Electrical Frequency Units: [Hz]	AC	UINT16
12	Pole Count	Get	Number of poles in the motor	AC	UINT16
15	Base Speed	Get, Set	Nominal speed at rated frequency from nameplate Units [RPM]	AC or DC	UINT16

Attribute explanations

Attribute	Value	Definition
Motor type	1	Permanent magnet DC motor
	2	FC DC motor
	3	Permanent magnet synchronous AC motor
	6	Wound rotor induction AC motor
	7	Squirrel cage induction AC motor

Control supervisor object, class 29h

The Control supervisor object can only be used if the ODVA AC/DC drive profile is in use.

The object models all the management functions for devices within the 'Hierarchy of Motor Control Devices'. The behavior of motor control devices is described by:

- AC/DC-drive object, class 2Ah (page 150) and
- Run/Stop event matrix under Run Forward & Run Reverse (Control supervisor object) (page 93).

See also section State (Control supervisor object) (page 97).

Note: If assembly instances are used, they override this object, for example, upon drive power-up.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of the CIP Object Class Definition upon which the implementation is based	Array of UINT8

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
3	Run 1 (RunFwd)	Get, Set	0 = Stop, 1 = Run See Run Forward & Run Reverse (Control supervisor object) (page 93).	BOOL
4	Run 2 (RunRev)	Get, Set	0 = Stop, 1 = Run See Run Forward & Run Reverse (Control supervisor object) (page 93).	BOOL
5	Net Control	Get, Set	0 = Local Control, 1 = Network Control	BOOL
6	State	Get	State of Object. See section State (Control supervisor object) (page 97).	UINT8
7	Running 1 (Fwd)	Get	0 = Stopped, 1 = Running	BOOL
8	Running 2 (Rev)	Get	0 = Stopped, 1 = Running	BOOL
9	Ready	Get	1 = Ready, Enabled or Stopping; 0 = Other state	BOOL

#	Attribute name	Services	Description	Data type
10	Faulted	Get	0 = Not faulted, 1 = Fault occurred	BOOL
11	Warning	Get	0 = No Warnings present, 1 = Warning	BOOL
12	FaultRst	Get, Set	0 → 1 Fault Reset	BOOL
13	Fault Code	Get	The fault that caused the last transition to the Faulted state. DRIVECOMM codes are reported. See the drive manual for further information on DRIVECOMM codes.	UINT16
14	Warning Code	Get	Code word indicating the warning present. If multiple warnings are present, the lowest code value is displayed. DRIVECOMM codes are reported. See the drive manual for further information on DRIVECOMM codes.	UINT16
15	CtlFromNet	Get	0 = NetControl disabled 1 = NetControl enabled	BOOL
16	DNFaultMode	Get, Set	2 = Vendor specified	UINT8
17	ForceFault	Get, Set	0 → 1 forces the drive to fault	BOOL

AC/DC-drive object, class 2Ah

The AC/DC-drive object can only be used if the ODVA AC/DC drive profile is in use.

The object models the functions specific to an AC or DC Drive.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of the CIP Object Class Definition upon which the implementation is based	Array of UINT8

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
3	At Reference	Get	Frequency arrival	BOOL
4	NetRef	Get, Set	Requests torque or speed reference to be local or from the network. 0 = Set Reference not DN Control 1 = Set Reference at DN Control Note that the actual status of torque or speed reference is reflected in attribute 29, RefFromNet.	BOOL
6	Drive mode	Get, Set	0 = Vendor-specific	UINT8
7	Speed Actual	Get	Units = See parameter 23 Speed scale (page 72).	SINT16
8	SpeedRef	Get, Set	Units = See parameter 23 Speed scale (page 72).	SINT16
11	TorqueActual	Get	Units = See parameter 24 Torque scale (page 73).	SINT16
12	TorqueRef	Get, Set	Units = See parameter 24 Torque scale (page 73).	SINT16
18	AccelTime	Get, Set	Units = milliseconds	UINT16
19	DecelTime	Get, Set	Units = milliseconds	UINT16

#	Attribute name	Services	Description	Data type
22	Speed Scale	Get, Set	Speed scaling factor. See parameter 23 Speed scale (page 72).	UINT8
24	Torque Scale	Get, Set	Torque scaling factor. See parameter 24 Torque scale (page 73).	UINT8
29	Ref From Net	Get	Reflecting attribute 4	BOOL

Drive parameter object, class 90h

With the FSCS-21 adapter module, drive parameters can also be accessed via Explicit Messaging. Explicit Messaging makes use of objects consisting of three parts: class, instance and attribute.

Note: When you use the drive parameter object to update the fieldbus configuration groups, changes to the fieldbus configuration will only take effect when the module is powered up the next time or when a 'Fieldbus Adapter parameter refresh' is given.

Class is always 144 (90h). Instance and attribute correspond to the drive parameter group and index in the following way:

- Instance = Parameter group (0…99) (ACS880/ACx580: 0...255)
- Attribute = Parameter index (01…99) (ACS880/ACx580: 0...255)

For example, parameter 99.01 is accessed as follows:

- Class = 144 = 90h
- Instance = 99 = 63h
- Attribute = 1 = 01h

Fieldbus configuration object, class 91h

The fieldbus configuration object allows you to configure the fieldbus configuration groups without the need to know the drive-specific groups associated with the configuration groups.

Note: When you use the fieldbus configuration object to update the fieldbus configuration groups, changes to the fieldbus configuration will only take effect when a reset service is requested of the Identity Object, the module is powered up the next time or when a 'Fieldbus Adapter parameter refresh' is given.

Class attributes

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of the Configuration Object	Array of UINT8

Instance #1: FSCS-21 configuration parameters group A (group 1)

The actual parameter group number depends on the drive type. Group A (group 1) corresponds to:

- parameter group 51 in ACS380 and ACS580
- parameter group is typically 51/54 (group 151/154 in some variants) in ACS880 if the adapter is installed as fieldbus adapter A/B.

#	Attribute name	Services	Description	Data type
1	Configuration Group A (Group 1) - Parameter 1	Get	See FBA type (page 64) .	UINT16
2	Configuration Group A (Group 1) - Parameter 2	Get, Set	See Protocol/Profile (page 64).	UINT16
3	Configuration Group A (Group 1) - Parameter 3	Get, Set	See Comm-rate (page 64).	UINT16
4	Configuration Group A (Group 1) - Parameter 4	Get, Set	See IP configuration (page 65).	UINT16
5	Configuration Group A (Group 1) - Parameter 5	Get, Set	See IP address 1 (page 65).	UINT16
6	Configuration Group A (Group 1) - Parameter 6	Get, Set	See IP address 1 (page 65).	UINT16
7	Configuration Group A (Group 1) - Parameter 7	Get, Set	See IP address 1 (page 65).	UINT16
8	Configuration Group A (Group 1) - Parameter 8	Get, Set	See IP address 1 (page 65).	UINT16
9	Configuration Group A (Group 1) - Parameter 9	Get, Set	See Subnet CIDR (page 66).	UINT16
10	Configuration Group A (Group 1) - Parameter 10	Get, Set	See GW address 1 (page 67).	UINT16

#	Attribute name	Services	Description	Data type
11	Configuration Group A (Group 1) - Parameter 11	Get, Set	See GW address 1 (page 67).	UINT16
12	Configuration Group A (Group 1) - Parameter 12	Get, Set	See GW address 1 (page 67).	UINT16
13	Configuration Group A (Group 1) - Parameter 13	Get, Set	See GW address 1 (page 67).	UINT16
14	Configuration Group A (Group 1) - Parameter 14	Get, Set	See Commrate port 2 (page 67).	UINT16
15	Configuration Group A (Group 1) - Parameter 15	Get, Set	See Service configuration (page 68).	UINT16
18	Configuration Group A (Group 1) - Parameter 18	Get, Set	See Safety Network Number (page 69).	UINT16
19	Configuration Group A (Group 1) - Parameter 19	Get, Set	See T16 scale (page 69).	UINT16
20	Configuration Group A (Group 1) - Parameter 20	Get, Set	See Control timeout (page 69).	UINT16
21	Configuration Group A (Group 1) - Parameter 21	Get, Set	See Idle action (page 71).	UINT16
22	Configuration Group A (Group 1) - Parameter 22	Get, Set	See Stop function (page 71).	UINT16
23	Configuration Group A (Group 1) - Parameter 23	Get, Set	See Speed scale (page 72).	UINT16
24	Configuration Group A (Group 1) - Parameter 24	Get, Set	See Torque scale (page 73).	UINT16
25	Configuration Group A (Group 1) - Parameter 25	Get, Set	See Reserved (page 74).	UINT16
26	Configuration Group A (Group 1) - Parameter 26	Get, Set	See Reserved (page 74).	UINT16
27	Configuration Group A (Group 1) - Parameter 27	Get, Set	See FBA A/B par refresh (page 74).	UINT16

#	Attribute name	Services	Description	Data type
28	Configuration Group A (Group 1) - Parameter 28	Get	See FBA A/B par table ver (page 75).	UINT16
29	Configuration Group A (Group 1) - Parameter 29	Get	See FBA A/B drive type code (page 75).	UINT16
30	Configuration Group A (Group 1) - Parameter 30	Get	See FBA A/B mapping file ver (page 75).	UINT16
31	Configuration Group A (Group 1) - Parameter 31	Get	See D2FBA A/B comm status (page 75).	UINT16
32	Configuration Group A (Group 1) - Parameter 32	Get	See FBA A/B comm SW ver (page 76).	UINT16
33	Configuration Group A (Group 1) - Parameter 33	Get	See FBA A/B appl SW ver (page 76).	UINT16

Instance #2: FSCS-21 configuration parameters group B (group 2)

The actual parameter group number depends on the drive type. Group B (group 2) corresponds to:

- parameter group 53 in ACS380 and ACS580
- parameter group is typically 53/56 (group 153/156 in some variants) in ACS880 if the adapter is installed as fieldbus adapter A/B.

#	Attribute name	Services	Description	Data type
1	Configuration Group B (Group 2) - Parameter 1	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580
2	Configuration Group B (Group 2) - Parameter 2	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580
3	Configuration Group B (Group 2) - Parameter 3	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580
4	Configuration Group B (Group 2) - Parameter 4	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580

#	Attribute name	Services	Description	Data type
5	Configuration Group B (Group 2) - Parameter 5	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580
6	Configuration Group B (Group 2) - Parameter 6	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580
7	Configuration Group B (Group 2) - Parameter 7	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580
8	Configuration Group B (Group 2) - Parameter 8	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580
9	Configuration Group B (Group 2) - Parameter 9	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580
10	Configuration Group B (Group 2) - Parameter 10	Get, Set	See FBA A/B data out 1 (client to drive) (page 76).	UINT16 / UINT32 ACS880 / ACx580

Instance #3: FSCS-21 configuration parameters group C (group 3)

The actual parameter group number depends on the drive type. Group C (group 3) corresponds to:

- parameter group 52 in ACS380 and ACS580
- parameter group is typically 52/55 (group 152/155 in some variants) in ACS880 if the adapter is installed as fieldbus adapter A/B.

#	Attribute name	Services	Description	Data type
1	Configuration Group C (Group 3) - Parameter 1	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580
2	Configuration Group C (Group 3) - Parameter 2	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580
3	Configuration Group C (Group 3) - Parameter 3	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580

#	Attribute name	Services	Description	Data type
4	Configuration Group C (Group 3) - Parameter 4	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580
5	Configuration Group C (Group 3) - Parameter 5	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580
6	Configuration Group C (Group 3) - Parameter 6	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580
7	Configuration Group C (Group 3) - Parameter 7	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580
8	Configuration Group C (Group 3) - Parameter 8	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580
9	Configuration Group C (Group 3) - Parameter 9	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580
10	Configuration Group C (Group 3) - Parameter 10	Get, Set	See FBA A/B data in1 (drive to client) (page 77).	UINT16 / UINT32 ACS880 / ACx580

TCP/IP interface object, class F5h

This object provides the mechanism to configure the TCP/IP network interface of the device.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of the TCP/IP Interface Object Class Definition upon which the implementation is based	Array of UINT8

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
1	Interface status	Get	Refer to section Attribute explanations (page 158).	DWORD
2	Configuration Capability	Get	Refer to section Attribute explanations (page 158).	DWORD
3	Configuration Control	Get	Refer to section Attribute explanations (page 158).	DWORD
4	Physical Link Object	Get	Path to physical link object	STRUCT of:
	Path Size		Path size	UINT
	Path		Logical segments identifying the physical link object	Padded EPATH
5	Interface Configuration	Get		STRUCT of:
	IP Address		IP Address	UINT32
	Network Mask		Network Mask	UINT32
	Gateway Address		Gateway Address	UINT32
	Unused			UINT32
	Unused			UINT32
	Default Domain Name		Default Domain Name for unqualified host names.	STRING
6	Host Name	Get / Set	Host name	STRING
8	TTL Value	Get / Set	TTL value for EtherNet/IP multi cast packets	UINT8 1 … 255

#	Attribute name	Services	Description	Data type
13	NV Encapsulation Inactivity	Get / Set	Encapsulation inactivity timeout	UINT16 0 = disabled 1 … 3600 timeout in seconds

Attribute explanations

Interface Status attribute (#1) bits			
Bit	**Name**	**Value**	**Description**
0 … 3	Interface configuration status	0	The Interface Configuration attribute has not been configured.
		1	The Interface Configuration attribute contains valid configuration obtained from BOOTP, DHCP or non-volatile storage.
		2	The IP address member of the Interface Configuration attribute contains valid configuration obtained from hardware settings (e.g., push-wheel, thumbwheel).
		3 … 15	Reserved

Interface Status attribute (#1) bits		
Bit	**Name**	**Description**
4	Mcast pending	Indicates a pending configuration change in the TTL Value and/or Mcast Config attributes. This bit is set when either the TTL Value or Mcast Config attribute is set, and cleared the next time the device starts.
5 … 31		Reserved, set to 0

Configuration Capability attribute (#2) bits		
Bit	**Name**	**Description**
0	BOOTP client	1 (True) = The device is capable of obtaining its network configuration via BOOTP.
1	DNS client	1 (True) = The device is capable of resolving host names by querying a DNS server.

Configuration Capability attribute (#2) bits		
Bit	**Name**	**Description**
2	DHCP client	1 (True) = The device is capable of obtaining its network configuration via DHCP.
3	DCHP-DNS update	1 (True) = The device is capable of sending its host name in the DHCP request as documented in Internet draft <draft-ietf-dhc-dhcp-dnc-12.txt>
4	Configuration settable	1 (True) = The Interface Configuration attribute is settable. Some devices, e.g., a PC or workstation, may not allow interface configuration to be set via the TCP/IP interface object.
5	Hardware configurable	1 (True) = The IP address member of the Interface Configuration attribute can be obtained from hardware settings (eg, push-wheel, thumb-wheel).
		0 (False) = The Status instance attribute (1) Interface configuration status field value shall never be 2. (The Interface configuration attribute contains valid configuration obtained from hardware settings.)
6 … 31		Reserved, set to 0

Configuration Control attribute (#3) bits			
Bit	**Name**	**Value**	**Description**
0 … 3	Start-up configuration	0	The device uses the interface configuration values previously stored (eg, in non-volatile memory or via hardware switches).
		1	The device obtains its interface configuration values via BOOTP.
		2	The device obtains its interface configuration values via DHCP upon startup.
		3 … 15	Reserved

Configuration Control attribute (#3) bits		
Bit	**Name**	**Description**
4	DNS enable	1 (True) = The device resolves host names by querying a DNS server.

Configuration Control attribute (#3) bits		
Bit	**Name**	**Description**
5 … 31		Reserved, set to 0.

Ethernet link object, class F6h

This object maintains link-specific counters and status information for the Ethernet communication interface.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of the Ethernet Link Object Class Definition upon which the implementation is based	Array of UINT8

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
1	Interface Speed	Get, Set	10 or 100 Mbps	UDINT
2	Interface Flags	Get, Set	Interface status flags. For details, refer to section Attribute explanations.	DWORD
3	Physical Address	Get	Ethernet MAC address of the module	ARRAY of 6 UINT8s

Attribute explanations

Attribute	Description	Bit	Definition
Interface Flags	Interface status flags	0	Link status
		1	Half/Full duplex
		2 … 4	Negotiation status
		5	Manual setting requires reset
		6	Local hardware fault
		7 … 31	Reserved

Connection object, class 05h

Do not modify this object. This object is only used while establishing the connection between the adapter module and the PLC.

The connection class allocates and manages the internal resources associated with both I/O and explicit messaging connections. The specific instance generated by the connection class is referred to as connection instance or connection object.

Connection object states			
State	**Description**	**State**	**Description**
00	Non-Existent	03	Established
01	Configuring	04	Timed Out
02	Waiting for Connection ID	05	Deferred Delete

Class attributes

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of the connection object	Array of UINT8

Instance attributes

Instance number	Description
1	Explicit messaging connection

Instance number	Description
2	Polled I/O connection
4	Change-of-State / Cyclic I/O connection

#	Attribute name	Services	Description	Data type
1	State	Get	State of the object. See the Connection object states (page 161).	UINT8
2	Instance Type	Get	Indicates either I/O (1) or messaging connection (0).	UINT8
3	Transport Class Trig-ger	Get	Defines the behavior of the connection.	UINT8
4	Produced Cnxn Id	Get	Placed in CAN Identifier Field when the connection transmits.	UINT16
5	Consumed Cnxn Id	Get	CAN Identifier Field value that denotes the message to be received	UINT16
6	Comm Character-istics	Get	Defines the Message Group(s) across which productions and consumptions are associated in this connection.	UINT8
7	Produced Connection Size	Get	Maximum number of bytes transmitted across this connection	UINT16
8	Consumed Connection size	Get	Maximum number of bytes received across this connection	UINT16
9	Expected Packet Rate	Get, Set	Defines the timing associated with this connection in milliseconds. A value of 0 deactivates the associated timers.	UINT16
12	Watchdog Timeout Ac-tion	Get, Set	Defines how to handle Inactivity/Watch-dog timeouts.	UINT8
13	Produced Connection Path Length	Get	Number of bytes in the produced_con-nection_path length attribute	UINT16

#	Attribute name	Services	Description	Data type
14	Produced Connection Path	Get	Application object producing data on this connection	Array of UINT8
15	Consumed Connection Path Length	Get	Number of bytes in the consumed_connection_path length attribute	UINT16
16	Consumed Connection Path	Get	Specifies the application object(s) that are to receive the data consumed by this connection object.	Array of UINT8
17	Production Inhibit Time	Get	Defines the minimum time between new data production in milliseconds.	UINT16

DLR object, class 47h

The Device Level Ring (DLR) Object provides the configuration and status information interface for the DLR protocol. The DLR protocol is a layer 2 protocol that enables the use of an Ethernet ring topology. The DLR Object provides the CIP application-level interface to the protocol.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of this object	UINT16

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
1	Network Topology	Get	Current network in topology mode	UINT8
2	Network Status	Get	Current status of network	UINT8
10	Active Supervisor Address	Get	IP and/or MAC address of the active ring supervisor	STRUCT of: UINT32 – IP address ARRAY – MAC of 6 UINT8s

#	Attribute name	Services	Description	Data type
11	Active Supervisor Precedence	Get	Precedence value of the active ring supervisor	UINT8
12	Capability Flags	Get	Describes the DLR capabilities of the device	DWORD

Attribute explanations

Attribute	Description	Value	Definition
Network Topology	The Network Topology attribute indicates the current network topology mode.	0	Linear topology
		1	Ring topology
Network Status	The Network Status attribute provides current status of the network based the device's view of the network.	0	Normal operation in Ring and Linear Network Topology modes.
		1	Ring Fault. A ring fault has been detected. Valid only when Network Topology is Ring.
		2	Unexpected Loop Detected. A loop has been detected in the network. Valid only when the Network Topology is Linear.
		3	Partial Network Fault. A network fault has been detected in one direction only. Valid only when Network Topology is Ring and the node is the active ring supervisor.
		4	Rapid Fault/Restore Cycle. A series of rapid ring fault/restore cycles has been detected, per the criteria in Chapter 9. Similar to the Partial Network Fault status, the supervisor remains in a state with forwarding blocked on its ring ports. The condition must be cleared explicitly via the "Clear Rapid Faults" service.

Attribute	Description
Active Supervisor Address	This attribute contains the IP address and/or Ethernet MAC address of the active ring supervisor. The initial values of IP address and Ethernet MAC address are 0, until the active ring supervisor is determined.

Attribute	Description
Active Supervisor Precedence	This attribute contains the precedence value of the active ring supervisor. The initial value is 0, until the active ring supervisor is determined.

Attribute	Description	Bit	Called	Definition
Capability Flags	The Capability Flags describe the DLR capabilities of the device.	0	Announce-based Ring Node	Is set if the ring node implementation of the device is based on processing of Announce frames
		1	Beacon-based Ring Node	Is set if the ring node implementation of the device is based on processing of Beacon frames
		2…4	Reserved	Must be set to zero
		5	Supervisor Capable	Is set if device is capable of providing the supervisor function
		6	Redundant Gateway Capable	Is set if device is capable of providing the redundant gateway function
		7	Flush_Table frame Capable	Is set if device is capable of supporting the Flush_Tables frame
		8…31	Reserved	Must be set to zero

QoS object, class 48h

Quality of Service (QoS) is a general term that is applied to mechanisms used to treat traffic streams with different relative priorities or other delivery characteristics. Standard QoS mechanisms include IEEE 802.1D/Q (Ethernet frame priority) and Differentiated Services (DiffServ) in the TCP/IP protocol suite.

The QoS Object provides a means to configure certain QoS-related behaviors in EtherNet/IP devices.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of this object	UINT16

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
1	802.1Q Tag Enable	Get, Set	Enables or disables sending 802.1Q frames on CIP and IEEE 1588 messages	UINT8
2	DSCP PTP Event	Get, Set	DSCP value for PTP (IEEE 1588) event messages	UINT8
3	DSCP PTP General	Get, Set	DSCP value for PTP (IEEE 1588) general messages	UINT8
4	DSCP Urgent	Get, Set	DSCP value for CIP transport class 0/1 Urgent priority messages	UINT8
5	DSCP Scheduled	Get, Set	DSCP value for CIP transport class 0/1 Scheduled priority messages	UINT8
6	DSCP High	Get, Set	DSCP value for CIP transport class 0/1 High priority messages	UINT8
7	DSCP Low	Get, Set	DSCP value for CIP transport class 0/1 Low priority messages	UINT8
8	DSCP Explicit	Get, Set	DSCP value for CIP explicit messages (transport class 2/3 and UCMM) and all other EtherNet/IP encapsulation messages	UINT8

Attribute explanations

Attribute	Description	Value	Definition
802.1Q Tag Enable	The 802.1Q Tag Enable attribute enables or disables sending 802.1Q frames on CIP and IEEE 1588 messages. When the attribute is enabled, the device sends 802.1Q frames for all CIP and IEEE 1588 messages. A change to the value of the attribute takes effect the next time the device restarts.	0	Disabled (default)
		1	Enabled

Attribute	Description
DSCP Value Attributes	Attributes 2…8 contain the DSCP values that must be used for the different types of EtherNet/IP traffic.

LLDP Management object, class 109h

The LLDP Management object contains administrative information for the LLDP protocol.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of this object	UINT16

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
1	LLDP Enable	Get, Set	Enable generation of LLDP Frames both Globally and per Port and the processing of received LLDP frames globally	Struct of: UINT16 – array length ARRAY of BYTE
2	msgTxInterval	Get, Set	From 802.1AB-2016. The interval in seconds at which LLDP frames are transmitted from this device.	UINT16
3	msgTxHold	Get, Set	From 802.1AB-2016. A multiplier of msgTxInterval to determine the value of the TTL TLV sent to neighboring devices.	UINT8
4	LLDP Datastore	Get	An indication of the retrieval methods for the LLDP database supported by the device	WORD
5	Last Change	Get	The value of sysUpTime taken the last time any entry in the local LLDP database (ignoring TTL) changed	UINT32

Attribute explanations

Attribute	Description	Bit	Name	Values	Description
LLDP Enable	Enable generation of LLDP Frames both Globally and per Port and the processing of received LLDP frames globally. The bit definitions of the LLDP Enable Array are shown on the right.	0	Global Enable	0 = LLDP Tx & Rx Disabled 1 = LLDP Tx & Rx Enabled (default)	If Global Enable (Bit 0) is set to 0, transmitting will be stopped for all ports (other bits must be ignored) and received LLDP frames will be ignored. When Global Enable (Bit 0) is set from 1 to 0, all existing table entries are removed.
		1…N	Port Tx Enable	0 = LLDP Tx Disabled 1 = LLDP Tx Enabled (default)	The values of elements corresponding to non-existent Ethernet Link instances or instances (eg, internal) that do not support LLDP must be set to 0 and are ignored
		>N	Reserved	Must be 0 and are ignored	Pad bits included as necessary to fill the last BYTE

Attribute	Description	Value	Definition
msgTxInterval	From 802.1AB-2016. The interval in seconds at which LLDP frames are transmitted from this device. The recommended default value is 30.	0	Reserved
		1…3600	Message Transmission Interval for LLDP frames
		3601… 65535	Reserved

Attribute	Description	Value	Definition
msgTxHold	From 802.1AB-2016. A multiplier of msgTxInterval to determine the value of the TTL TLV sent to neighboring devices. The recommended default value is 4.	0	Reserved
		1…100	Message Transmission Multiplier for LLDP Frames
		101…255	Reserved

Attribute	Description	Bit	Definition
LLDP Datastore	An indication of the retrieval methods for the LLDP database supported by the device.	0	LLDP Data Table Object
		1	SNMP
		2	NETCONF YANG
		3	RESTCONF YANG
		4…15	Reserved

LLDP Data Table Object, class 10Ah

The LLDP Data Table object displays a record of all adjacent LLDP implementing devices that are currently active according to the receive state machine of the LLDP protocol. If neighboring devices have not been detected, only class attributes may be reachable.

One instance of the LLDP Data Table object is implemented for each adjacent device detected. Only 8 instances are supported. Instances are created and removed as neighboring devices change. The same instance number is maintained for each neighboring device until the next power cycle of the device implementing this object.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of this object	UINT16
2	Max Instance	Get	Maximum instance number of an object currently created in this class level of the device.	UINT16

#	Attribute name	Services	Description	Data type
3	Number of Instances	Get	Number of object instances currently created at this class level of the device.	UINT16

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
1	Ethernet Link Instance Number	Get	The local instance number of the Ethernet Link Object that matches the physical Ethernet port the LLDP frame populating this instance was received on, if known.	UINT16
2	MAC Address	Get	The neighboring MAC Address received from the CIP MAC Address, Chassis ID, or Port ID TLV	ETH_MAC_ADDR
3	Interface Label	Get	The neighboring Interface Label received from the CIP Interface Label, Chassis ID or Port ID TLV	SHORT_STRING
4	Time to Live	Get	The number of seconds the neighboring information is to be considered valid	UINT16
5	System Capabilities TLV	Get	The capabilities which the neighboring device supports based on currently loaded firmware.	STRUCT of: WORD – System Cap WORD – Enabled Cap
6	IPv4 Management Addresses	Get	A list of IPv4 encoded management addresses as defined by one or more received Management Address TLVs (TLV Type = 8)	STRUCT of: UINT8 – num of addr ARRAY of UINT32

#	Attribute name	Services	Description	Data type
7	CIP Identification	Get	The CIP Identification TLV of the neighboring device, if present.	STRUCT of: UINT16 – Vendor ID UINT16 – DevType UINT16 – ProductCode BYTE – Major Rev UINT8 – Minor Rev UINT32 – CIP Ser Num
8	Additional Ethernet Capabilities	Get	A TLV for Ethernet Preemption Support from the neighboring device	STRUCT of: BOOL – Preemption support BOOL – Preemption status BOOL – Preemption active
9	Last Change	Get	The value of sysUpTime taken the last time an attribute in this instance changed.	UINT32

Attribute explanations

Attribute	Description	Value	Definition
Ethernet Link Instance Number	The local instance number of the Ethernet Link Object that matches the physical Ethernet port the LLDP frame populating this instance was received on, if known.	0	Unknown
		1… 65535	Ethernet Link Object (0xF6) Instance Number

Attribute	Description	Value	Definition
Time to Live	The number of seconds the neighboring information is to be considered valid. **Note:** A received TTL TLV value of 0 means that the table entry must be removed according to IEEE 802.1AB-2016.	0	Reserved
		1… 65535	Time To Live (in seconds)

Attribute	Description	Bit	Definition
System Capabilities TLV	The system capabilities TLV is a structure that contains bitmaps of both the supported and enabled capabilities of the neighboring device.	0	Other
		1	Repeater
		2	Bridge
		3	Access Point
		4	Router
		5	Telephone
		6	DOCSIS Cable Device
		7	End Station
		8	C-VLAN component
		9	S-VLAN component
		10	Two-port MAC relay component
		11…15	Reserved by IEEE

Safety Supervisor Object, class 39h

The Safety Supervisor is the core object of CIP Safety Devices.

The Safety Supervisor object centralizes application object state definitions and related status information, exception status indications (alarms and warnings), and defines a behavior model which is assumed by objects identified as belonging to safety devices. That is, if a reset is requested of the Safety Supervisor object instance, it is performed by this object instance as well as all of its associated application objects.

The Identity object get its state information from the Safety Supervisor object. A reset request to the Identity object is not supported by CIP Safety devices and must be replaced by a reset request to the Safety Supervisor object, which has additional qualifiers to execute the request.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of this object	UINT16

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
11	Device Status	Get	Represents the current state of the device. Its value changes as the state of the device changes. For details, refer to section Attribute explanations (page 174).	UINT8
12	Exception Status	Get	Indicates the status of the alarms and warnings for the device For details, refer to section Attribute explanations (page 174).	BYTE
15	Alarm Enable	Get, Set	Enables (1) or disables (0) the Safety Supervisor object's process of setting Exception bits	BOOL
16	Warning Enable	Get, Set	Enables (1) or disables (0) the Safety Supervisor object's process of setting Exception bits	BOOL
25	Configuration UNID	Get	CFUNID - Identifies the owner of a Device Configuration. all 0xFF = Tool-only configuration 0 = un-owned, accept any owner	10 octets
26	Safety Configuration Identifier	Get	The SCID is comprised of the Safety Configuration CRC + Safety Configuration Time Stamp. This is the signature for the Configuration	10 octets
27	Target UNID	Get	The current UNID of the device	10 octets

#	Attribute name	Services	Description	Data type
28	Output Connection Point Owners	Get	Safety connections to outputs define an owner to prevent errant connections from hijacking an output resource in a validated safety system. This owner Id identifies which originator safety device has been given ownership rights to this safety output connection. For details, refer to section Attribute explanations (page 174).	STRUCT

Attribute explanations

Attribute	Description	Value	State
Device Status	This attribute represents the current state of the device. Its value changes as the state of the device changes.	0	Undefined
		1	Self-Testing
		2	Idle
		3	Self-Test Exception
		4	Executing
		5	Abort
		6	Critical Fault
		7	Configuring
		8	Waiting for TUNID
		9… 50	Reserved by CIP
		51	Waiting for TUNID with Torque Permitted
		52	Executing with Torque Permitted
		53… 99	Device Specific
		100… 255	Vendor Specific

Attribute	Description	Bit	Definition
Exception Status	A single byte attribute whose value indicates the status of the alarms and warnings for the device.	0	ALARM/Device-common
		1	ALARM/Device-specific
		2	ALARM/Manufacturer-specific
		3	Reserved, 0
		4	WARNING/Device-common
		5	WARNING/Device-specific
		6	WARNING/Manufacturer-specific
		7	0

Attribute	Description
Alarm Enable	These Boolean attributes are used to enable (1) or disable (0) the Safety Supervisor object's process of setting Exception bits. When disabled, corresponding bits are never set, and if they were set, disabling clears them. Also, alarm and warning states are not retained: when enabled, bits will be set only if the corresponding condition is true.
Warning Enable	

Output Connection Point Owner	Safety connections to outputs define an owner to prevent errant connections from hijacking an output resource in a validated safety system. This owner Id identifies which originator safety device has been given ownership rights to this safety output connection.
	It is defined as a STRUCT of:
	• Number of Array Entries: UINT16
	• Output Owners: ARRAY of STRUCT:
	  • OCPUNID: 10 octets (The owner UNID for the output resource; 0 = unowned, accept any owner)
	  • EPATH size: UINT8 (Path size, number of bytes)
	  • Application Resource: Packed EPATH (The path to owned resource [eg, 20 04 24 01 -Assembly class, instance 1])

Safety Supervisor Object Specific Services

Service Code	Service Name	Description
54h	Safety reset	Resets the device in a manner similar to the Identity object, except that a password is required to execute.

Safety Validator Object, class 3Ah

The Safety Validator Object is designed for use in Safety devices on all CIP Networks. The Safety Validator contains the information necessary to coordinate and maintain reliable safety connections between client and server safety applications. The primary role of the Safety Validator function is to act as a safety transport manager of multiple low-level CIP connections that together form a complete safety connection.

Class attributes (Instance #0)

#	Attribute name	Services	Description	Data type
1	Revision	Get	Revision of this object	UINT16
8	Safety Connection Fault Count	Get	Diagnostic Counter that is a running count of Safety Connection Faults	UINT16

Instance attributes (Instance #1)

#	Attribute name	Services	Description	Data type
1	Safety Validator State	Get	State of the Safety Connection	UINT8
2	Safety Validator Type	Get	Safety Validator type used in this instance	UINT8
3	Ping Interval EPI Multiplier	Get	Number that defines the Ping_Count_Interval for a particular connection	UINT16
4	Time Coord Msg Min Multiplier	Get	This attribute is used as part of the Safety Validator function. For details, refer to section Attribute explanations (page 178).	STRUCT
5	Network Time Expectation Multiplier	Get	This attribute defines the connection reaction time associated with this connection. For details, refer to section Attribute explanations (page 178).	STRUCT

#	Attribute name	Services	Description	Data type
6	Timeout Multiplier	Get	This attribute defines the tolerance the Safety Validator will have for lost messages before faulting a connection and taking a safety action. For details, refer to section Attribute explanations (page 178).	STRUCT
7	Max Consumer Number	Get	Maximum number of consumers permitted for the connection	UINT8
12	Max Data Age	Get	Diagnostic which holds the largest Data Age detected in 128 microsecond increments. Attribute only updated by Safety Consumers.	UINT16
13	Application Data Path	Get	Points to the application data attached to this safety connection	Packed EPATH
14	Error Code	Get	Reason for error within this instance	UINT16
15	Producer/Consumer Fault Counters	Get	This attribute reflects the internal Producer or Consumer Fault Counters. For details, refer to section Attribute explanations (page 178).	STRUCT

Attribute explanations

Attribute	Description	Value	State Name	Definition
Safety Validator State	This attribute defines the current state of the Safety Validator Object instance. Transitions in this state attribute reflect the current state of the object. This attribute can be used to determine the condition of established safety connections.	0	Unallocated	The Safety Validator object is not allocated to a connection or is closed
		1	Initializing	The Safety Validator object is in the process of exchanging time coordination information across the connection. The “safety” connection is not yet fully established.
		2	Established	In this state the Safety Validator instance is fully established and producing/consuming safety data on behalf of the safety application.
		3	Connection Failed	This state is entered when all connections associated with this validator have failed for any reason. In producers, all multicast consumer connections must have failed. As long as a consumer is still operating, the producer remains in the Established state.
		4…255	Reserved	

Attribute	Description	Bit 7	Bit 6
Safety Validator Type	This attribute defines the type of Safety Validator transport that is being used. Targets must derive this value from the Safety Connection parameters in the Safety Open.	Producer(client) = 0 Consumer(server) = 1	Safety Connection Type: 0 = unallocated 1 = Single-cast 2 = Multicast 3-127 = Reserved

Attribute	Description
Time Coord Msg Min Multiplier	This attribute is used as part of the Safety Validator function. This parameter is obtained from the Safety Segment Parameters in the SafetyOpen: STRUCT of: • Time Coord Msg Min Multiplier array size: UINT8 (Size of array equals Max Consumer number for multicast producers and 1 for single-cast and multicast consumer) • Time Coord Msg Min Multiplier: ARRAY of: UINT16 (Minimum number of 128 microsecond increments it could take for a Time Coordination Message to traverse from the consumer to the producer)
Network Time Expectation Multiplier	This attribute defines the connection reaction time associated with this connection. This parameter is obtained from the Safety Segment Parameters in the SafetyOpen: STRUCT of: • Network Time Expectation Multiplier array size: UINT8 (Size of array equals Max Consumer number for multicast producers and 1 for single-cast and multicast consumer) • Network Time Expectation Multiplier: ARRAY of: UINT16 (Maximum number of 128 microsecond increments that a consumer should allow the age of the safety data to reach)
Timeout Multiplier	This attribute defines the tolerance the Safety Validator will have for lost messages before faulting a connection and taking a safety action. This parameter is obtained from the Safety Segment Parameters in the SafetyOpen: STRUCT of: • Timeout Multiplier array size: UINT8 (Size of array equals Max Consumer number for multicast producers and 1 for single-cast and multicast consumer) • Timeout Multiplier: ARRAY of: UINT8 (Determines the number of messages that can be lost before declaring a connection error)
Producer/Consumer Fault Counters	This attribute reflects the internal Producer or Consumer Fault Counters: STRUCT of: • Producer/Consumer Counter Array Size: UINT8 (Size of array equals Max Consumer number for multicast producers and 1 for single-cast and multicast consumer) • Producer/Consumer Fault Counter: ARRAY of UINT8 (Number of Faults detected this hour)

16

CIP Safety

Contents of this chapter

This chapter describes the safety system when the FSCS-21 module is communicating with a safety PLC using the CIP Safety protocol. It gives information on the module states and transitions, and the contents of CIP Safety messages. The chapter also includes installation instructions, configuration instructions for a safety PLC, and fault tracing tips.

Introduction

When the drive is controlled from a safety PLC, the reliability of the fieldbus communication must be secured. This can be done with the CIP Safety technology. The CIP Safety technology includes safety measures that minimize the effect of transmission errors that can occur when messages are transferred in a complex network. CIP Safety is an application layer (protocol) that describes the safety communication between fail-safe devices. It is an additional layer on top of standard CIP protocols (for example, DeviceNet® and EtherNet/IP™). The CIP Safety protocol can be used for safety applications up to SIL 3 according to IEC 61508 / IEC 62061, Category 4 according to EN 954-1 or PL e according to ISO 13849-1. For more information on CIP Safety and EtherNet/IP, see www.odva.com.

General safety information for CIP Safety devices

- The replacement of safety devices requires that the replacement device is configured properly and operation of the replacement device must be user verified.

- If you choose to configure safety connections with an SCID of 0, you are responsible for ensuring that originators and targets have the correct configurations.
- SNN numbers for each safety network or safety sub-net must be unique system-wide.
- When a safety device is configured directly from a workstation, compare the transferred SCID and configuration data with the SCID and configuration data that is originally viewed in the workstation.
- User testing is the means by which all downloads are validated.
- The signature can only be considered verified (and configuration locked) after user testing.
- Configuring an originator with connection data and/or target configuration data must be downloaded to the target so it can be tested and verified. Only then can SCIDs from the target be confirmed.
- Device operation must be completely tested before setting the Lock Attribute.
- A pre-existing configuration must be cleared from the safety device before installing it onto a safety network.
- All safety devices must be commissioned with an IP address (and Ethernet link setting if necessary) before installing them onto a safety network.
- The persons who implement safety functions must carefully consider the implications of mixing devices with different SIL levels on the network.
- Safety connection configurations must be tested after they are applied in an originator to make sure that the target connection is operating as intended.
- You must make sure visually that all configuration data was downloaded correctly.

⚠WARNING Do not attempt to use LEDs as operational indicators. LEDs are not reliable indicators and cannot be guaranteed to give accurate information. Use them only for general diagnostics during commissioning and troubleshooting.

⚠WARNING Originators that have an “automatic” SNN setting feature must only use that feature when the safety system is not being relied upon.

System description

Required components

- FSCS-21 module, revision A or later

- compatible safety PLC system (for example, Allen-Bradley GuardLogix or Compact GuardLogix controller).

CIP Safety description

Safety data format

Safety Input Assembly = 131, Size = 2 bytes				
Byte	**Bit**	**Name**	**Value**	**Definition**
0	0	Device state	0	Operational state
			1	Safe state
	1	Mode.0	0	Refer to section Description of modes (page 185).
			1	
	2	Mode.1	0	
			1	
	3	STO signals active	0	STO signals not at 24 V
			1	STO signals at 24 V
	4	STO active	0	STO function not active
			1	STO function active
	5	SS1 active	0	SS1-t function not active
			1	SS1-t function active
	6…7	Reserved [1)]	0	Reserved for future use
1	0…7	Reserved [1)]	0	Reserved for future use

1) The safety PLC must set reserved bits to value 0 and ignore the value of the reserved bits. This ensures the compatibility with future versions.

Safety Output Assembly = 31, Size = 4 bytes				
Word[1)]	**Bit**	**Name**	**Value**	**Definition**
0	0	STO request	0	Deactivate STO function
			1	Activate STO function
	1	SS1 request	0	Deactivate SS1-t function
			1	Activate SS1-t function
	2…15	Reserved[2)]	0	Reserved for future use
1	0…15	SS1-t time	0	Safe stop 1 monitoring time limit (SS1-t) in 100 ms. Value range is 0…32767. For example, value 21 = 2.1 s. Negative values result in 0.0 s.
			1	

1) Data type Word is transferred in Little Endian byte order.
2) The safety PLC must set reserved bits to value 0. This ensures the compatibility with future versions.

FSCS-21 module modes

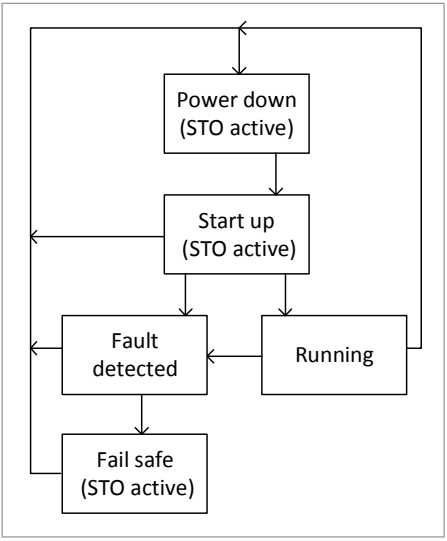
FSCS-21 modes

The FSCS-21 module can be in these modes:

- Start-up mode

  Note: CIP Safety communication is not used in this mode.

- Fault-detected mode
- Fail-safe mode
- Run mode.



Description of modes

The table describes the FSCS-21 module modes and how they are shown in the CIP Safety messages.

The table refers to several variables that are available to the programmer of an F-Host program.

Mode	Description
Run	The application runs without fatal failures. Bits in the Safety Assembly data for the FSCS-21 module: • Mode.0 = 0 • Mode.1 = 0
Fault-detected	This mode is reached if a fatal error (for example, CPU test, RAM test, STO channel test, etc., failed) takes place. In the end, the drive STO is activated and transition to fail-safe mode is performed. Bits in the Safety Assembly data for the FSCS-21 module: • Mode.0 = 1 • Mode.1 = 0 • Device state = 1 **Note:** Time spent in this mode may be so short that device_mode-bits may not be updated to Safety PLC.

Mode	Description
Fail-safe	The application keeps the system in the safe state. CIP Safety communication is up and running. This mode is reached after safe stopping function is completed in fault-detected mode. The drive STO is kept permanently active. Bits in the Safety Assembly data for the FSCS-21 module: • Mode.0 = 0 • Mode.1 = 1 To exit the fail-safe mode, reboot the FSCS-21 module.

Transitions between modes

Transition	Description
Start-up - Run	Initial tests done successfully.
Start-up - Fault-detected	Fatal failure detected during start-up.
Run - Fault-detected	Fatal failure detected during run.
Fault-detected - Fail-safe	FSCS-21 reached Safe state. Refer to section States (page 41).

Installation

Installation procedure:

1. Install the FSCS-21 safety functions fieldbus module to the drive. Refer to chapters Planning for installation, Mechanical installation, Electrical installation.
2. Connect the FSCS-21 module to the safety PLC through a EtherNet/IP network with CIP Safety.

Configuration

This section shows how to configure an Allen-Bradley® safety PLC for FSCS-21 with the Rockwell Automation® Studio 5000®.

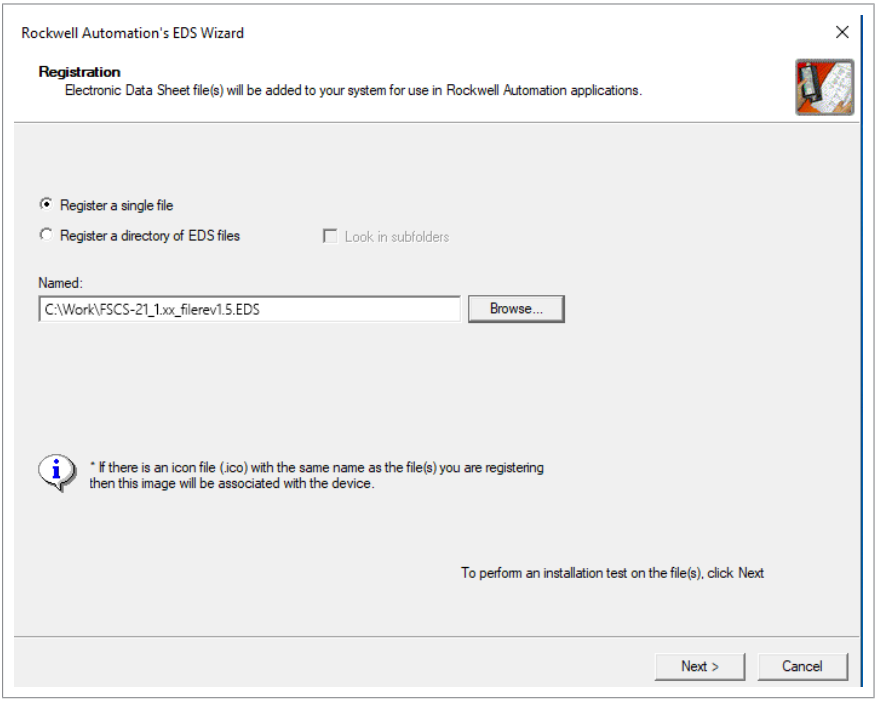
Registering EDS file

Register the EDS file as follows:

1. Start the Rockwell Automation EDS Hardware Installation Tool.
2. If you have installed an older version of the EDS file before, it is recommended to remove it first. Remove the file as follows:

a. Click the **Remove** button.
b. When the list of known devices is populated, click the **Found Device** button and type **fscs**.
c. Select **FSCS-21**, click the **Next** button and obey the on-screen instructions to complete the removal.

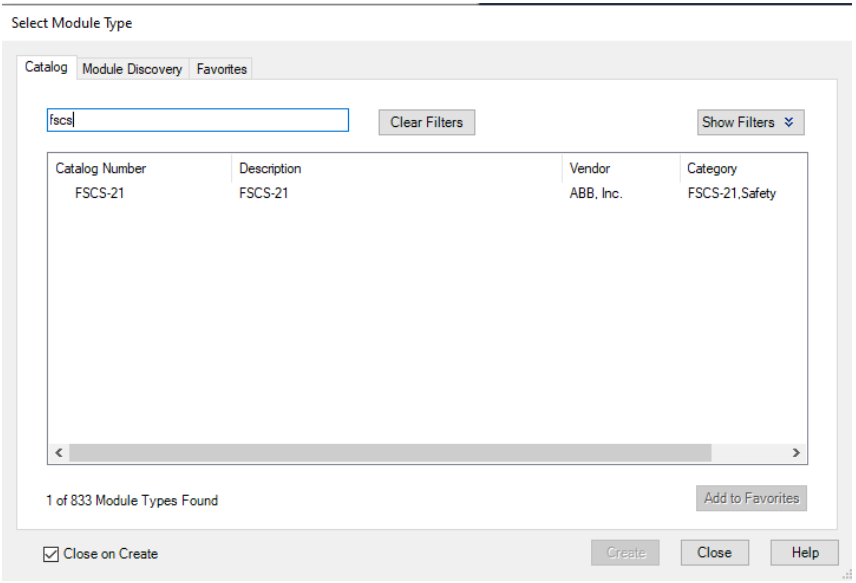
3. To add the EDS file, click on the **Add** button.
4. Select **Register a single file** and fill in the path to the correct EDS file (for example, FSCS-21_1.xx_filerev1.5.EDS) in the field **Named**. Obey the on-screen instructions to complete the installation.



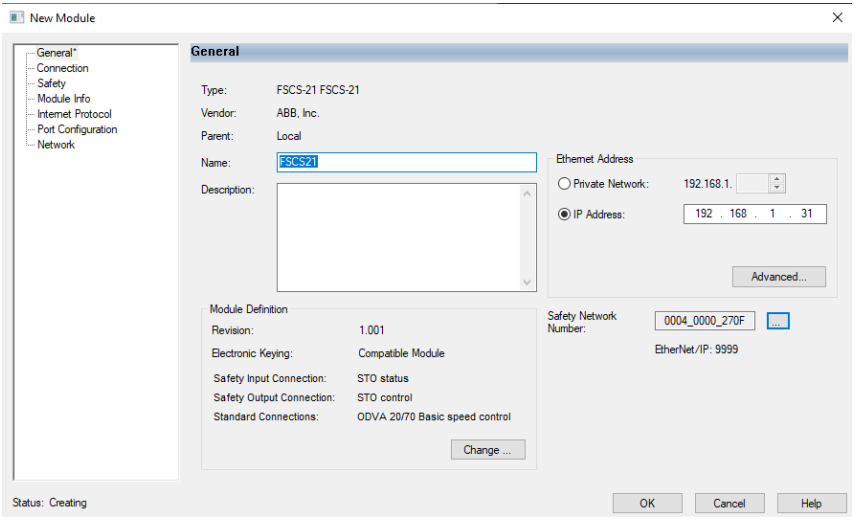
Adding FSCS-21 from EDS file

Add the FSCS-21 from EDS file as follows:

1. Under **Controller Organizer**, open **I/O Configuration** and right click on **Ethernet**. Select **New module**.
2. In the **Select Module Type** dialog, type **fscs** in the filter field. Select **FSCS-21** and click **Create**.



3. The **New Module** dialog opens. In the **General** tab, fill in the **Name** and **IP Address** fields.

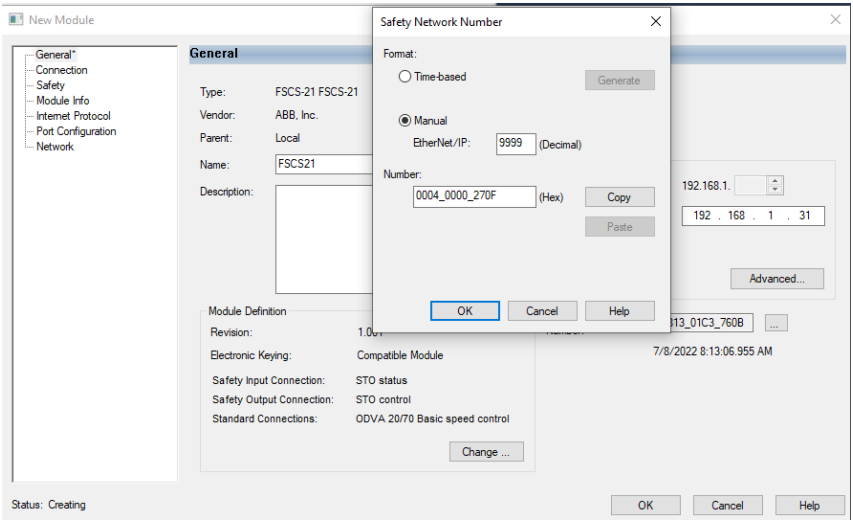


4. Click on the button adjacent to the field **Safety Network Number**. In the **Safety Network Number** window, make sure that **Manual** is selected. Set the Safety Network Number to a value between 1…9999 (default value is 9999). Set the

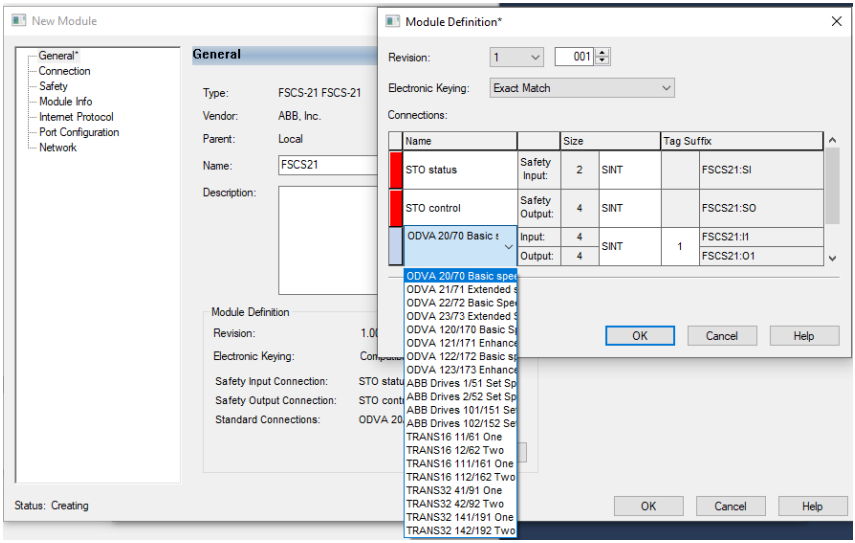
drive parameter 51.18 to the same value. Refresh FSCS-21 parameters with parameter 51.27 after you change parameter 51.18.

Note: The parameter group depends on the drive type. For more information, refer to section FSCS-21 configuration parameters – group A (group 1) (page 64).

Note: FSCS-21 supports only the manual format for the Safety Network Number.

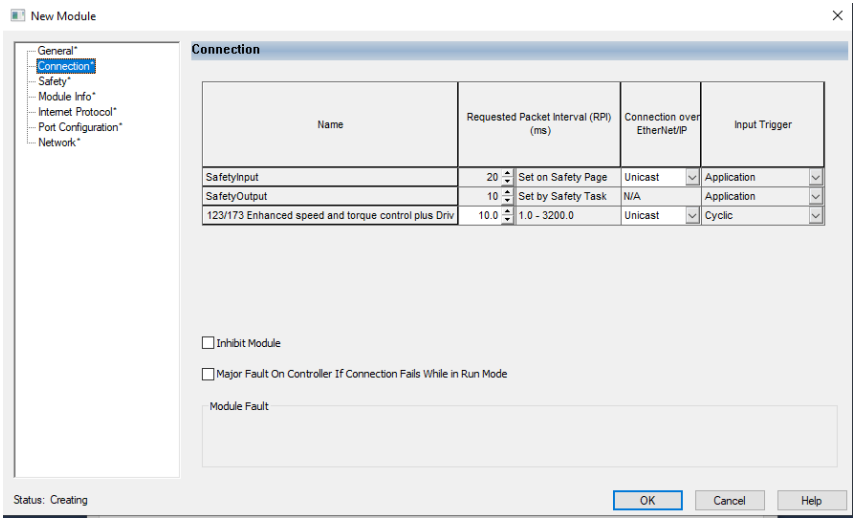


5. Click the **Change** button in the **Module Definition** box. Select the applicable non-safety assemblies and click **OK**.

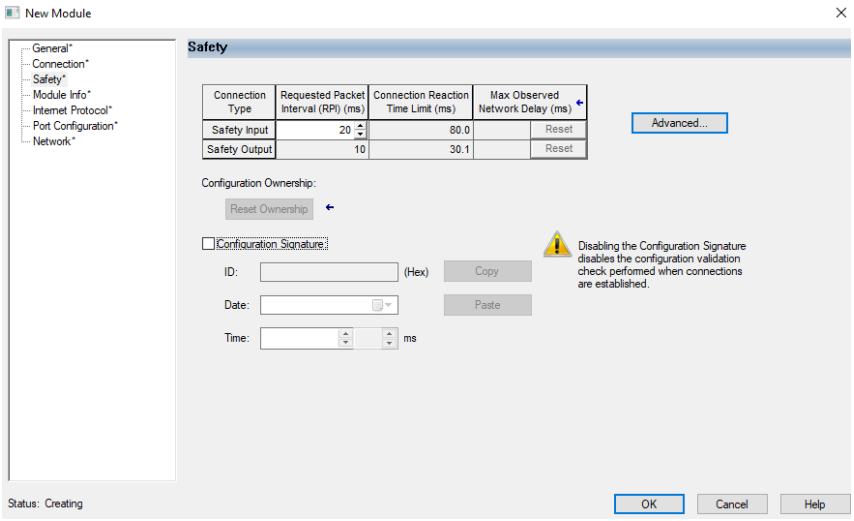


Note: To improve compatibility, you can change the data type of the assemblies from **SINT** to **INT**.

6. In the **Connection** tab, set the requested packet interval (RPI) value for the non-safety connection.

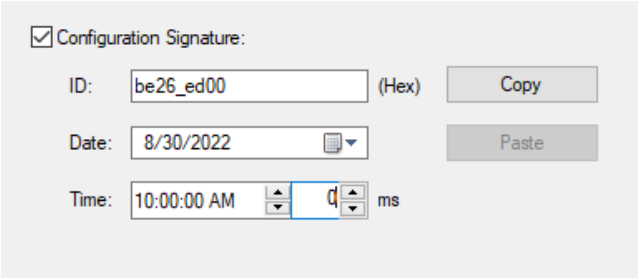


7. In the **Safety** tab, set the necessary Safety-related settings.



If the drive does not communicate with the PLC, it is possible that it is owned by another device. In this case, click the **Reset Ownership** button.

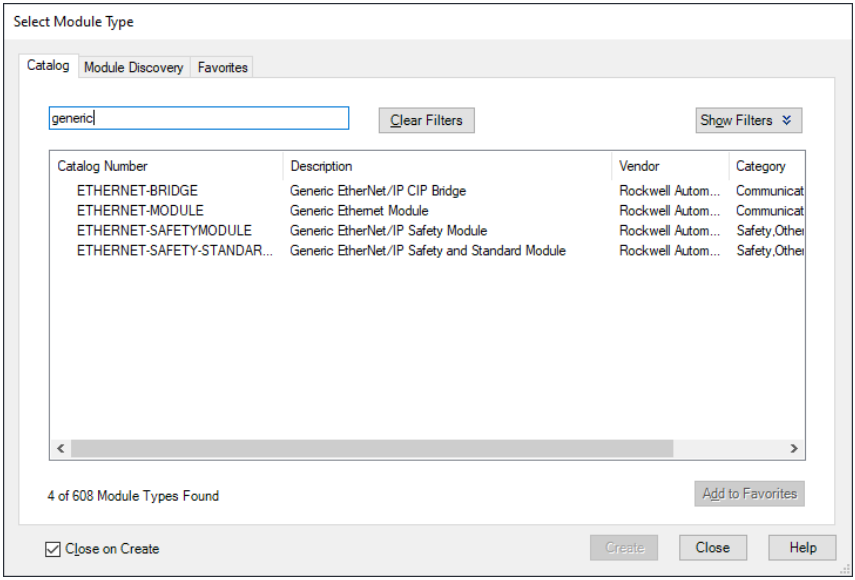
8. If necessary, enable the configuration signature. The timestamp shown in the figure below is valid only for the UTC time zone. If the time zone is different, adjust the time accordingly.



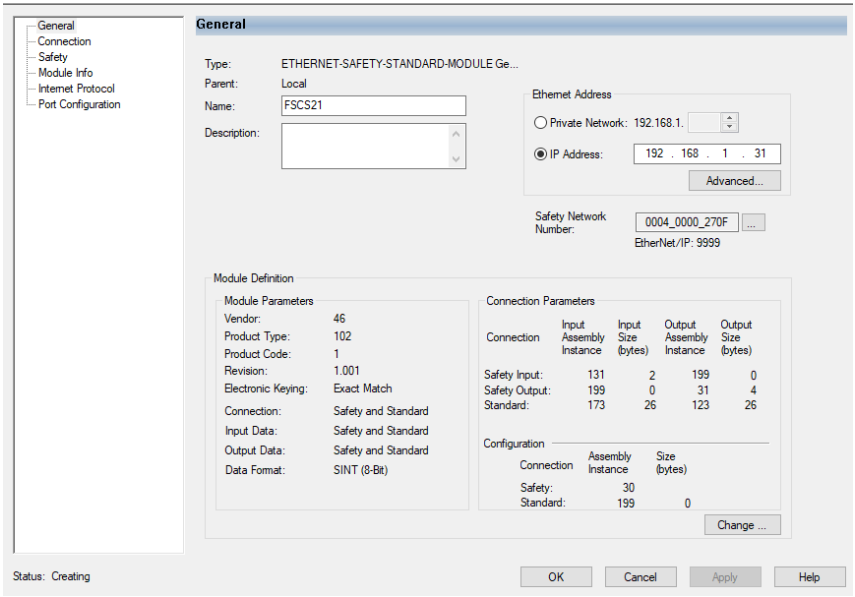
9. Click **OK** to exit the **New Module** window.

Adding FSCS-21 manually

1. Under **Controller Organizer**, open **I/O configuration** and right-click on **Ethernet**. Select **New module**.
2. In the **Select Module Type** dialog, type **generic** in the filter field. Select **ETHERNET-SAFETY-STANDARD-MODULE** from the list and click on **Create**.



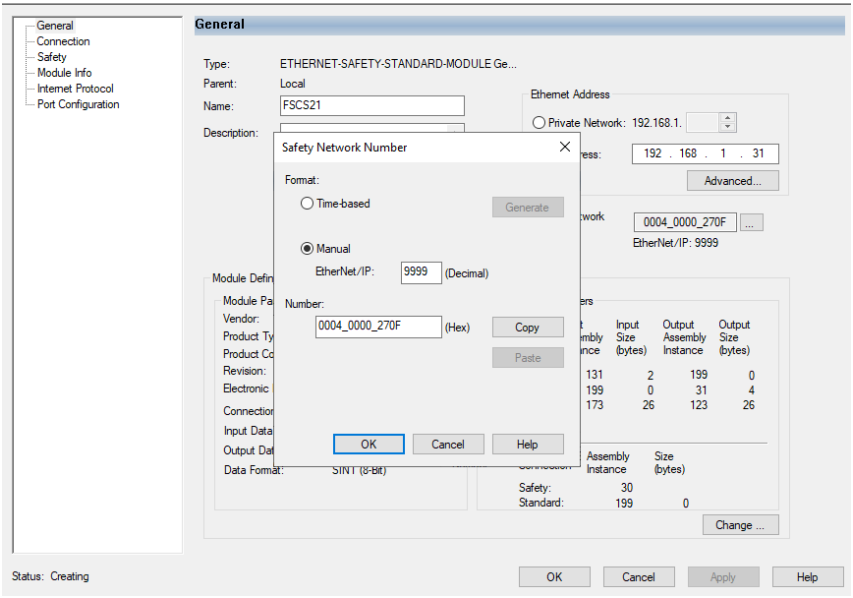
3. The **New Module** dialog opens. In the **General** tab, fill in the **Name** and **IP Address** fields.



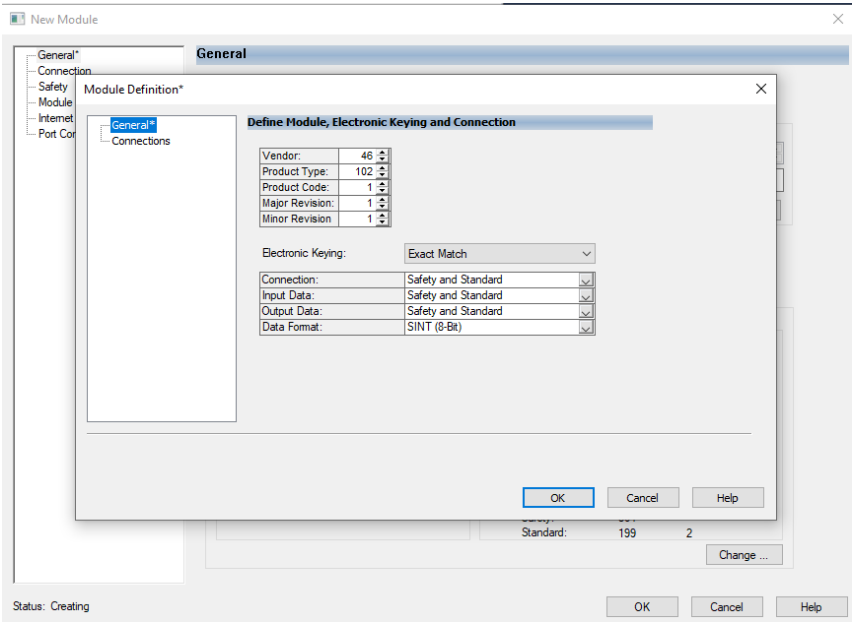
4. Click on the button adjacent to the field **Safety Network Number**. In the **Safety Network Number** window, make sure that **Manual** is selected. Set the Safety Network Number to a value between 1…9999 (default value is 9999). Set the drive parameter 51.18 to the same value. Refresh FSCS-21 parameters with parameter 51.27 after you change parameter 51.18.

   Note: The parameter group depends on the drive type. For more information, refer to section FSCS-21 configuration parameters – group A (group 1) (page 64).

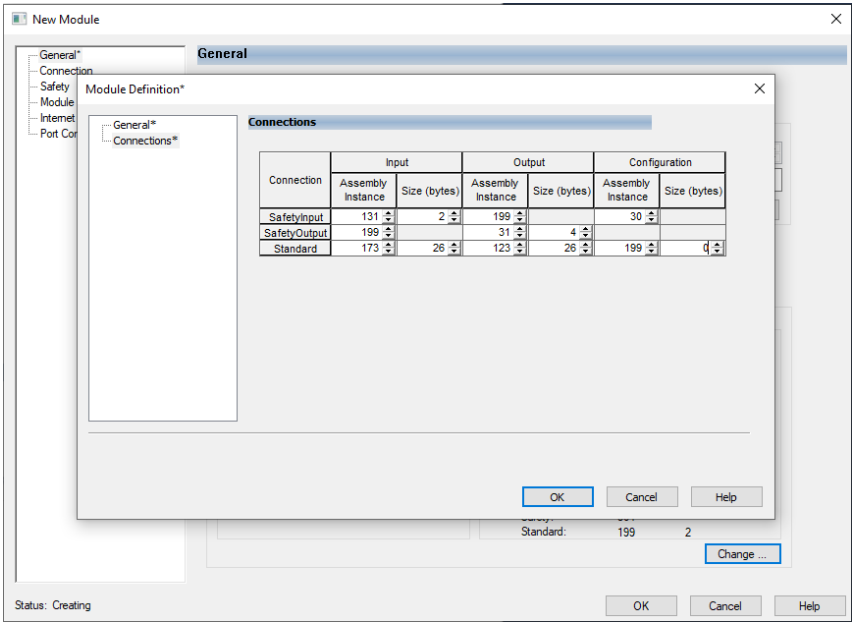
   Note: FSCS-21 supports only the manual format for the Safety Network Number.

5. 

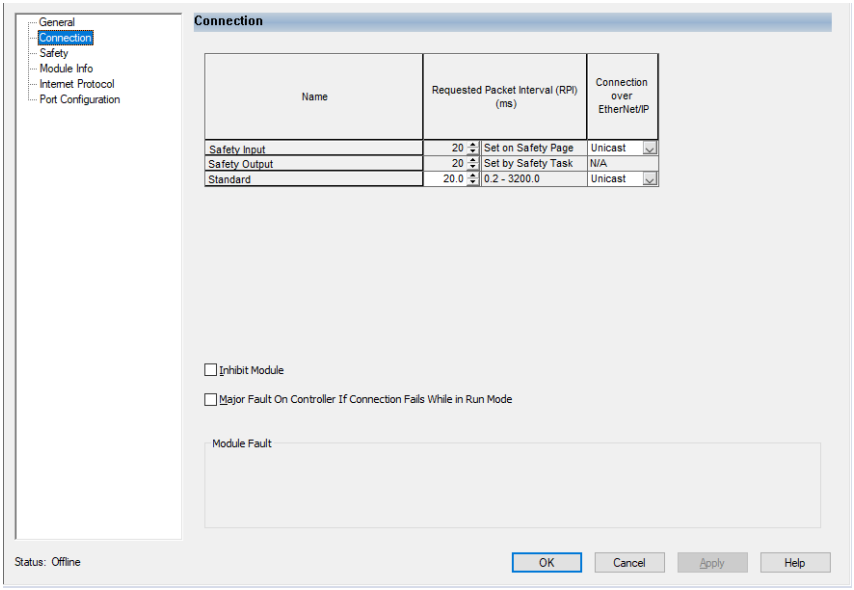
6. Under section Module Definition, click on **Change ...**.
7. In the **Module Definition** window, fill in the values as shown in the figure below.



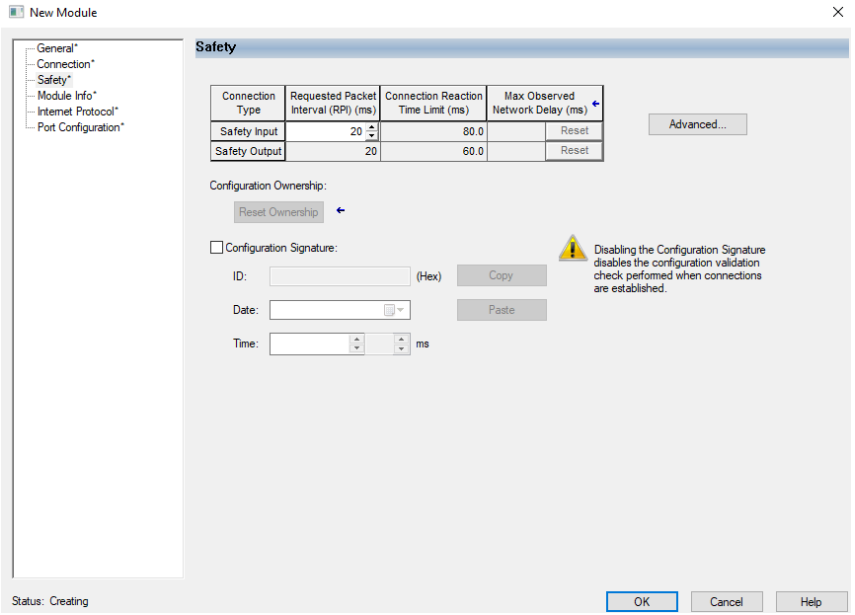
8. Go to the **Connections** tab. In the **Standard** row, fill in the values for **Input** and **Output** according to the table in section Select output and input assembly instances (page 81). Make sure that all other values are the same as shown in the figure below. Then click on **OK**.



9. In the **Connection** tab, set the requested packet interval (RPI) value for the non-safety connection.

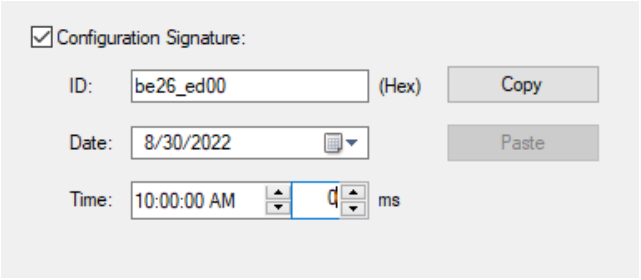


10. In the **Safety** tab, set the necessary Safety-related settings.



If the drive does not communicate with the PLC, it is possible that it is owned by another device. In this case, click the **Reset Ownership** button.

11. If necessary, enable the configuration signature. The timestamp shown in the figure below is valid only for the UTC time zone. If the time zone is different, adjust the time accordingly.



12. Click **OK** to exit the **New Module** window.

17

Verification and validation

Contents of this chapter

This chapter gives information on the verification and validation of the safety functions.

Verifying the achieved SIL/PL level

Verification of the functional safety system ensures that the implemented safety system meets the requirements specified for the system in the safety requirements specification phase.

The most convenient way to verify the required SIL/PL level reached with the implemented system is to use a specific safety calculator software.

Validation of the safety functions

You must do a validation test to make sure that the safety function operates correctly and according to the safety requirements.

Competence

The person who does the validation test of the safety function must be a competent person with expertise and knowledge of the safety function and functional safety, as required by IEC 61508-1 clause 6. This person must document and sign the test procedures and report.

Validation procedure

You must do the validation test using the checklist given in this manual and the validation test plan of the complete safety system:

- at the initial start-up of the safety function
- after changes related to the safety function (wiring, components, safety function -related parameter settings, etc.)
- after changes related to the power unit or its circuit boards
- after maintenance work related to the safety function
- at the proof test of the safety function.

The validation test must include at least the following steps:

- you must have a validation test plan
- you must test all commissioned functions for correct operation, from each operation location
- you must document all validation tests
- you must sign and store the validation test report for further reference.

Validation test reports

You must store the signed validation test reports in the logbook of the machine. The report must include, as required by the referred standards:

- a description of the safety application (including a figure)
- a description and revisions of safety components that are used in the safety application
- a list of all safety functions that are used in the safety application
- a list of all safety-related parameters and their values
- documentation of start-up activities, references to failure reports and resolution of failures
- the test results for each safety function, checksums, date of the tests, and confirmation by the test personnel.

You must store any new validation test reports done due to changes or maintenance in the logbook of the machine.

Checks before the start-up

Use this checklist to make sure that the system is prepared for the first start-up.

Make sure that ...	☑
The installation is checked according to individual product checklists (drive, safety component) and the checklists given in this manual.	☐
All necessary configuration steps are completed.	☐
All tools are removed from the installation area.	☐
It is not dangerous to start the system.	☐

Validation checklists for start-up

Validation of the STO function

Do these steps to validate the STO function:

1. Make sure that the parameters related to the STO function are correctly set. Refer to chapter Configuration.
2. Make sure that it is safe to start, run and stop the motor during the test.
3. Start the drive and set it to a typical motor speed for the application.
4. Activate the STO function from the safety PLC. Make sure that the motor coasts to a stop.
5. Make sure that the drive STO is active.
6. Try to start the drive. Make sure that it does not start while the STO function is active.
7. Deactivate the STO function from the safety PLC.
8. If the safety PLC is configured for manual acknowledgement of the safety function: Try to start the drive. Make sure that it does not start.
9. If the safety PLC is configured for manual acknowledgement of the safety function: Acknowledge the safety function from the PLC.
10. If parameter 31.22 is set to value Fault/Fault: Try to start the drive. Make sure that it does not start.
11. If parameter 31.22 is set to value Fault/Fault: Reset the faults from the drive.
12. Make sure that there are no unwanted errors in the drive.
13. Start the drive and make sure that the motor runs normally.

Validation of the SS1-t time limit hit

Do these steps to validate the SS1-t time limit hit:

1. Make sure that the SS1-t function is configured correctly.
2. Mak sure that parameter 25.15 Proportional gain em stop is set according to application needs.
3. Make sure that you can run and stop the motor freely.
4. Start the drive and set it to a motor speed typical for the application.
5. Activate the SS1-t function from the safety PLC.
6. Set drive emergency ramp time to be longer than SS1-t time (drive parameter groups 23 and 46 depending on operation mode, refer to drive firmware manual).
7. After SS1-t time runs out and the drive STO is activated, make sure that the system behaves in a safe way.
8. Restore drive emergency ramp time value to match your application requirement.
9. Remove SS1-t request/activation from the safety PLC.
10. Reset safety ramp fault from the drive.
11. If parameter 31.22 is set to value Fault/Fault: Reset the faults from the drive.
12. Continue to the SS1-t function validation procedure.

Validation of the SS1-t function

Do these steps to validate the SS1-t function:

1. Make sure that the SS1-t function is configured correctly.
2. Make sure that you can run and stop the motor freely.
3. Make sure that parameter 25.15 Proportional gain em stop is set according to application needs.
4. Start the drive and set it to a motor speed typical for the application.
5. Activate the SS1-t function from the safety PLC.
6. Make sure that the motor speed ramps down properly and the time monitoring delay is set correctly.
7. Make sure that the drive STO is activated when the drive reaches zero speed.
8. Make sure that the drive cannot be restarted while the SS1-t function request is on.
9. Remove SS1-t function request/activation from the safety PLC.

10. If safety function acknowledgement is configured to be manual in the safety PLC, make sure that the drive cannot be restarted.
11. Acknowledge the safety function from the safety PLC.
12. If parameter 31.22 is set to value Fault/Fault: Try to start the drive. Make sure that it does not start.
13. Reset the faults from the drive.
14. Make sure that there are no unwanted errors in the drive.
15. Restart the drive and make sure that the motor runs normally.

18

Start-up

Contents of this chapter

This chapter gives general precautions that you must take before you start up the safety system for the first time.

Safety

⚠WARNING Obey the safety instructions of the drive. If you ignore them, injury or death to personnel, or damage to the equipment can occur.

Only a qualified electrical professional who has sufficient knowledge about functional, machine, and process safety is permitted to start up the safety circuit.

⚠WARNING Until all the safety functionality is validated, the system must not be considered safe. Refer to chapter Verification and validation.

⚠WARNING FSCS-21 module has automatic acknowledgement method. Make sure that the system is designed so that this does not cause unacceptable risk, for example, due to an automatic start of the drive.

Checks

Before starting the system for the first time, make sure that:

- the installation is checked according to individual product checklists (drive, safety component) and the checklists given in this manual
- all necessary configuration steps are completed
- all tools are removed from the installation area
- it is not dangerous to start the system.

19

Fault tracing

Contents of this chapter

This chapter describes the status LEDs and provides generic diagnostics and troubleshooting tips.

Event types

The module generates three types of events to the drive: pure events, warnings, and faults. Only faults will cause the drive to stop. All event types are shown in the Event logger.

Fault messages

Code	Fault	Description	AUX code(s)	What to do
0x7A9A	FSx undefined fault	The fault is not recognized by the drive firmware.		Record the AUX codes from this fault and contact ABB.
0x7A9B	FSx internal fault	FSX internal faults (for example, CPU, memory, or peripherals)		Hardware can be defective. Contact ABB.

Code	Fault	Description	AUX code(s)	What to do
0x7A9C	FSx STO diagnostics fault	FSX STO fault. There is a fault in the STO circuit (for example, in the cabling) or inside the drive. **Note:** Initial STO test faults can occur in normal operation, depending on the drive type and possible attached devices into STO cable. In that case, they are signs that STO is kept open until the initial STO test is succesfully passed.	**0000 000A** Initial STO test failed since STO hi line is in 0 V	Examine the STO cable. If the fault continues, record the AUX codes from this fault and contact ABB.
			0000 000B Initial STO test failed in channel 1	Examine the STO cable. If the fault continues, record the AUX codes from this fault and contact ABB.
			0000 000C Initial STO test failed in channel 2	Examine the STO cable. If the fault continues, record the AUX codes from this fault and contact ABB.
			0000 000D Run time STO pulsing failed in channel 1	Examine the STO cable. If the fault continues, record the AUX codes from this fault and contact ABB.
			0000 000E Run time STO pulsing failed in channel 2	Examine the STO cable. If the fault continues, record the AUX codes from this fault and contact ABB.
0x7A9D	FSx temperature fault	FSX temperature fault (eg, overtemperature).	**0000 0003** Overtemperature, sensor 1	Make sure that the temperature of the installation agrees with the requirements for ambient conditions. Record the AUX codes from this fault and contact ABB.
			0000 0004 Overtemperature, sensor 2	Make sure that the temperature of the installation agrees with the requirements for ambient conditions. Record the AUX codes from this fault and contact ABB.

Code	Fault	Description	AUX code(s)	What to do
0x7A9F	FSx communication fault	FSX communication fault: Internal error in safety communication.		Record the AUX codes from this fault and contact ABB.
0x7AA0	FSx safety ramp fault	FS safety ramp fault: The drive did not stop in the monitoring time limit of SS1-t function.		Check if the drive can decelerate the load using the ramp time.

Note: The descriptions for AUX codes can be seen in Drive composer event log.

Warning messages

Code	Warning	Description	AUX code(s)	What to do
0xA7DC	FSx undefined warning	The fault is not recognized by the drive firmware.		Record the AUX codes from this warning and contact ABB.

Code	Warning	Description	AUX code(s)	What to do
0xA7DD	FSx safety bus warning	CIP Safety configuration problem.	**0000 0021** Waiting for CIP Safety connection	This warning is always shown after drive or module reboot. If the warning persists, the PLC cannot establish a CIP Safety connection to the drive. Examine the PLC program (for example, make sure that it is configured correctly, it is in RUN state, etc).
			0000 0022 Time coordination related issue	Possible causes: • A parity or CRC error was detected. Make sure that there are no network communication problems. • No time coordination messages were received during Ping interval (Timeout Multiplier × RPI). Make sure that the Timeout Multiplier and RPI values are sufficient (in advanced safety connection settings). If the network operates correctly and the Timeout Multiplier and RPI values are sufficient, but this error persists, the cause can be defective hardware. Contact ABB.
			0000 0023 Uncategorized error in CIP Safety connection	An unrecognized error was detected.

Code	Warning	Description	AUX code(s)	What to do
			0000 0024 Consumer activity monitor	No messages were received during Network Time Expectation timeout. Make sure that the connection is correctly configured and that there are no network communication problems.
			0000 0025 Timeout due to faulty messages	There are too many faulty messages or no valid messages during Network Time Expectation timeout. Make sure that there are no network communication problems.
			0000 0026 Timeout due to valid messages	The age of the last message is larger than Network Time Expectation. Make sure that there are no problems with the PLC.
			0000 0027 Connection set to idle	Set PLC to Run mode.

Event messages

Code	Event	Description
0xB795	FSx undefined event	FSX undefined event.
0xB796	FSx diagnostics	If AUX = 0000 0000, no errors exist. In any other case, gather the AUX codes from this event and contact ABB.

LEDs

LEDs are not reliable indicators and cannot be guaranteed to provide accurate information. They should only be used for general diagnostics during

commissioning or troubleshooting. Do not attempt to use LEDs as operational indicators.

Host status LED

LED status	Indication
Flashing green	Establishing communication to drive control unit.
Green	Connection to drive control unit OK.
Flashing red	Communication to drive control unit lost temporarily.
Red, together with Module LED red	The device is in a fail-safe state.
Flashing orange, alternating with the Module LED flashing orange	Internal file system error. The error may be cleared by cycling drive power. If the error persists, contact your local ABB representative.
Orange	STO is active. STO signal is 0 V (Shown when connection to drive control unit is OK and no fail-safe is indicated.)

Module status LED

LED status	Safety Supervisor state	Indication
Off	No Power	There is no power supplied to the device.
Green	Executing	The device is operating in a normal condition.
Flashing green	Idle state	The device is in the Idle or Standby State. Reference the Identity Object or Safety Supervisor Object.
Flashing red	Abort	Recoverable fault.
Red, together with Host LED red	Critical Fault	The device is in fail-safe state.
Flashing red/green	Device Self Testing	The device is in Self Test.

Network status LED

LED status	Network Connection Status	Indication
Off	Not Powered/Not On-line	Device is not on–line. The device has not completed the Dup_MAC_ID test yet. The device is not powered, look at the Module LED.
Flashing green	On-line, Not Connected	Device is on–line but has no connections in the established state. The device has passed the Dup_MAC_ID test, is on–line, but has no established connections to other nodes. For a Group 2 Only device it means that this device is not allocated to a controller. For a UCMM capable device it means that the device has no established connections. For Safety Devices, a connection may be established, but the Validator has not completed an initial Time Coordination exchange.
Green	Link OK, Online, Connected	The device is on–line and has connections in the established state.
Flashing red	Connection Time-out	One or more I/O Connections are in the Timed–Out state.
Red	Critical Link Failure	Failed communication device. The device has detected an error that has rendered it incapable of communicating on the network.

Status and control words

This table gives a list of the FSCS-21 module and drive status and control words. You can view these in the parameter window of Drive Composer pro.

Note: The status and control words in group 200 are non-safety critical information. Use these for fault tracing and status information only.

Index	Name	Description	Bit	Name	Values
200.05	FS control word 1	Shows the states of the FS commands.	0	STO request	0 = Off 1 = On
			2	SS1 request	0 = Off 1 = On

Index	Name	Description	Bit	Name	Values
200.06	FS control word 2	Shows the states of the FS commands.	6	SS1 ramp selection	0 = Off 1 = On
200.07	FS status word 1	Shows the FS status word 1.	0	FSx mode bit 1	0 = Boot 1 = Running 2 = Failure detected 3 = Fail-safe 5 = Undefined
			1	FSx mode bit 2	
			2	FSx mode bit 3	
			3	FSx state bit 1	0 = Safe state 1 = Operational
			4	FSx state bit 2	
			5	FSx STO active	0 = STO not active 1 = STO function active
			9	SS1 monitoring	0 = SS1 not completed 1 = SS1 completed
200.08	FS status word 2	Shows the FS status word 2.	15	STO completed	0 = STO not completed 1 = STO completed

Index	Name	Description	Bit	Name	Values
200.09	Drive status word 1	Shows the drive status word 1.	0	Drive status bit 1	0 = Disabled 1 = Ready on 2 = Ready run 3 = Starting 4 = Ready ref 5 = Stopping 6 = Faulted
			1	Drive status bit 2	
			2	Drive status bit 3	
			3	Drive status bit 4	
			6	Modulation [1)]	0 = Off 1 = On
			7	STO circuit 1	0 = Off 1 = On
			8	STO circuit 2	0 = Off 1 = On
			9	SS1 active	0 = Off 1 = On

1) This bit only shows the status of drive modulation, not the Safe state of the drive. Safe state of the drive is achieved only when STO is activated.

Reporting problems and failures

If you detect any failure in the safety module or safety functions, always contact your local ABB representative.

20

Maintenance

Contents of this chapter

This chapter gives maintenance instructions, and also information on decommissioning and proof tests.

Planning

All maintenance and the repair actions on a safety critical system are safety critical. You must plan and do them accordingly.

Before making changes, back up the drive parameters.

Tools

A control panel or the Drive Composer PC tool is necessary for the maintenance tasks.

FSCS-21 configuration back-up

Contents of this chapter

This chapter presents the settings for FSCS-21 configuration backup.

Description

FSCS-21 settings are stored in the drive parameters and also in the configuration files. FSCS-21 configuration parameters are included in the backup of the drive parameters.

Settings for backup

Consider the following points:

- Backup is not slot-specific. For example, backup in FBA A, slot 1 of the control unit can be restored to FBA A, slot 2.
- Backup depends on the fieldbus channel. For example, backup of FSCS-21 in FBA A is not restored to FSCS-21 in FBA B.

Configuration backup for FSCS-21

The settings are saved to the drive after 10 seconds. If a Refresh command is given to FSCS-21 using parameter 51.27, the pending backup is transferred to the drive immediately and FSCS-21 reboots after the transfer is completed.

Note: The new setting is not saved to drive if the drive was powered off or the adapter was disconnected from the drive within 10 seconds of changing a setting.

FSCS-21 module replacement

If the FSCS-21 module fails to operate, you must replace it with a new one. Do not try to repair the module.

Replacing the FSCS-21 module

⚠WARNING Obey the safety instructions of the drive. If you ignore them, injury or death to personnel, or damage to the equipment can occur. If you are not a qualified electrical professional, do not do electrical installation or maintenance work.

1. Stop the driven machinery and prevent an unexpected start-up.

   Note: Module parameters are automatically saved in the drive. It is not necessary to make a manual backup.

2. Isolate the drive from the main supply voltage and from all dangerous external voltages: do the steps in section **Electrical safety precautions** of the drive hardware manual. Make sure that it is safe to start the work.
3. Remove the wiring and the FSCS-21 module.
4. Mark clearly on the FSCS-21 module that it is decommissioned.
5. Install the FSCS-21 module and wiring to the drive according to chapters Mechanical installation and Electrical installation.
6. Re-configure the HW configuration in the PLC project.
7. Make sure that the installation is complete. Refer to the checklist in chapter Start-up.

8. Do the validation procedure for each safety function according to chapter Verification and validation.
9. Update the revision of the new FSCS-21 module to the logbook of the driven machine.

Drive replacement

If you must replace the drive where the FSCS-21 module is installed, for example, because of a serious drive failure, follow the procedure below.

Reinstalling the FSCS-21 module to another drive

⚠WARNING Obey the safety instructions of the drive. If you ignore them, injury or death to personnel, or damage to the equipment can occur. If you are not a qualified electrical professional, do not do electrical installation or maintenance work.

1. Stop the driven machinery and prevent an unexpected start-up.
2. Make a backup file of the drive parameters before drive replacement. You can make a backup of the drive configuration with the control panel or the Drive Composer PC tool. Refer to the applicable drive firmware manual, control panel manual, or Drive Composer manual.
3. Isolate the drive from the main supply voltage and from all dangerous external voltages: do the steps in section **Electrical safety precautions** of the drive hardware manual. Make sure that it is safe to start the work.
4. Remove the wiring and the FSCS-21 module.
5. Install the new drive. Refer to the drive hardware manual.
6. Install the FSCS-21 module and wiring to the new drive according to chapters Mechanical installation and Electrical installation.
7. Make the necessary parameter settings according to chapter Configuration, or restore the parameters from the backup file. Refer to the applicable drive firmware manual, control panel manual, or Drive Composer manual.
8. Make sure that the installation is complete. Refer to the checklist in chapter Start-up.
9. Do the validation procedure for each safety function according to chapter Verification and validation.

   Note: The STO function is the basic safety function and it must be validated first.
10. Update hardware and firmware versions of the new drive to the logbook of the driven machine.

Drive firmware update

If you must update the firmware of the drive where the FSCS-21 module is installed, follow the procedure below.

Updating the firmware of the drive where the FSCS-21 module is installed

1. Stop the driven machinery and prevent an unexpected start-up.
2. Make a back-up of the drive parameters. You can make a backup of the drive configuration with the control panel or the Drive Composer PC tool. Refer to the applicable drive firmware manual, control panel manual, or Drive Composer manual.
3. Update the firmware of the drive.
4. Make the necessary parameter settings according to chapter Configuration, or restore the parameters from the backup file. Refer to the applicable drive firmware manual, control panel manual, or Drive Composer PC tool manual.
5. Boot the FSCS-21 module with parameter 51.27 FBA A par refresh.
6. Do the validation procedure for each safety function according to chapter Verification and validation.

   Note: The STO function is the basic safety function and it must be validated first.

7. Update hardware and firmware versions of the new drive to the logbook of the driven machine.

Note: You can restore default fieldbus configuration to the drive and module with parameter 96.06 Parameter restore using the selection Reset all fieldbus settings. This also clears other parameters from the drive. Refer to the drive firmware manual for more information.

Updates

After changes in the safety application or the safety system configuration, you must do the validation tests to make sure that the safety functionality is maintained. See chapter Verification and validation (page 197).

Proof tests

If periodic proof testing is necessary based on the safety calculations, you must include proof tests in the maintenance plan and do them periodically.

After the operation of the circuits is validated at start-up, safety functions shall be maintained by periodic proof testing. In high demand mode of operation, proof

testing is not necessary. The module must be replaced after 20 years. In low demand mode of operation, the maximum proof test interval is 5 or 2 years. Refer to section Safety data (page 226). The proof test is done by rebooting the module.

Decommissioning

When you decommission the FSCS-21 module, make sure that the safety of the machine is maintained until the decommissioning is completed. Mark clearly on the module that it is decommissioned.

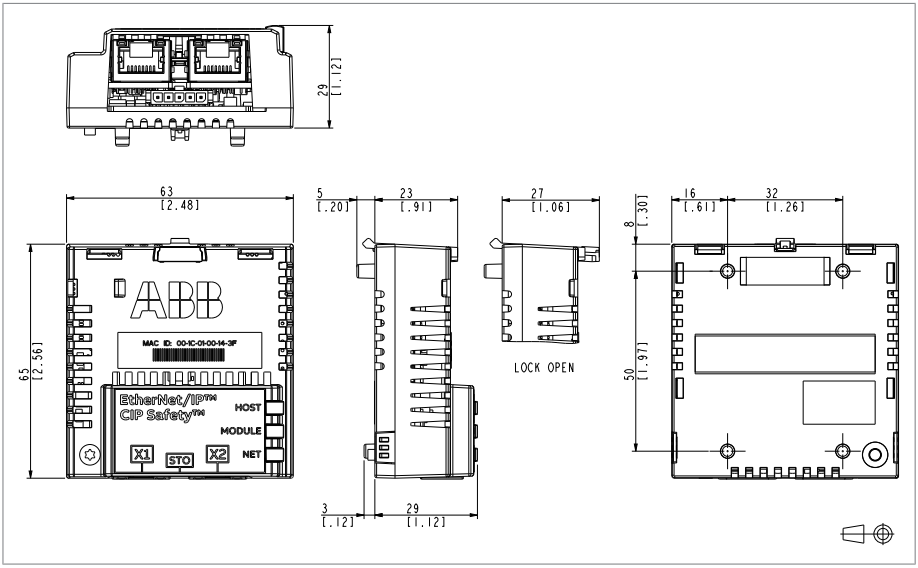
21

Technical data

Contents of this chapter

This chapter contains the technical specifications of the FSCS-21 module.

Dimension drawing



General data

Installation	Into an option slot on the drive control unit
Degree of protection	IP20
Power supply	Internal (3.3 V)
Power consumption	2 W continuous
Package	Cardboard. Plastic wrapping: Antistatic air bubble sheet (PE).
Indicators	Three bicolor LEDs (HOST, MODULE, NETWORK/NET)
General	Printed circuit board conformal coated
Weight	72 g (0.16 lbs)
EMC	Complies with EMC standards EN 61800-3:2004, IEC 61000-6-7:2014 and IEC 61800-5-2:2016. Maximum length of STO cable: 3 m (9.8 ft)
Disposal	For the end-of-life recycling instructions, see the drive hardware manual.

Connection data

Connectors	RJ-45 connector to Ethernet (X1) RJ-45 connector for chaining another adapter module (X2) STO connector

Ambient conditions

	Operation installed for stationary use	Storage in the protective package	Transportation in the protective package
Altitude (installation site)	0…1000 m (0…3300 ft) above sea level, no derating required 1000…2000 m (3300…6600 ft) above sea level, air outside the module derated to -15…+49 °C (+5…+120 °F) 2000…4000 m (6600…13200 ft) above sea level, air outside the module derated to -15…+40 °C (+5…+104 °F)	-	-
Air temperature	-15…+65 °C (+5…+149 °F)	-40…+70 °C (-40…+158 °F)	-40…+70 °C (-40…+158 °F)
Relative humidity	5…95%. No condensation permitted. If corrosive gases are present, the maximum permitted humidity is 60%.		
Contamination levels IEC 60721-3-x	No contaminants, conductive dust or corrosive dust permitted. Use at least IP54 enclosure in an environment where conductive dust or corrosive dust exists.		
Chemical gases	Class 3C2	Class 1C2	Class 2C2
Solid particles	Class 3S2 No conductive dust permitted.	Class 1S3	Class 2S2
Vibration IEC 60068-2-6, Test Fc (2007-12)	Frequency range: 10…150 Hz Amplitude ±0,075 mm, 10…57,56 Hz Constant peak acceleration 10 m/s^2 (1 g_n), 57,76…150 Hz		
Shock IEC 60068-2-27 Test Ea (2008-02)	Peak acceleration 50 m/s^2 (5 g_n) Pulse duration 30 ms 3 pulses in each direction with STO and SS1 functions activated		

	Operation installed for stationary use	Storage in the protective package	Transportation in the protective package
Atmospheric pressure	70…106 kPa (0.7…1.05 atmospheres)		

Response times

FSCS-21 response time	10 ms (for both safety functions and associated fault reaction function)
Total response time	To calculate the total response time, include the drive STO response time, FSCS-21 response time, PLC response time, and the response time of all other components included in the safety function.

Ethernet link

Compatible devices	Ethernet Standard IEEE 802.3 and IEEE 802.3u devices.
Medium	10BASE-TX or 100Base-TX with Auto-negotiation and Auto-MDIX (Auto-crossover) • Wiring: CAT5e/6 S/FTP, CAT5e/6 S/STP, CAT5e/6 SF/FTP • Connector: RJ-45 • Termination: Internal • Maximum segment length: 100 m (328 ft)
Topology	Bus, star or ring. A maximum of 50 nodes is permitted for FSCS-21 module in a ring topology. In a chain the maximum recommended number of nodes is 50.
Transfer rate	10 Mbps or 100 Mbps
Serial communication type	Half-duplex or full-duplex communication
Protocol	EtherNet/IP CIP Safety over EtherNet/IP

TCP and UDP service ports

There are multiple in-bound and out-bound network services running on the module. Some ports are protocol-specific and are not used when other protocols are selected.

Port	Service	Purpose
68 (UDP)	DHCP	DHCP client **Note:** Used only when IP configuration method is selected as "Dyn IP DHCP".
24576 (UDP)	ABB Netconfig	• Auto discovery protocol • Used by ControlBuilder plus (IP Configuration tool) and Drive Composer pro tools • Discovers ABB-specific Ethernet devices in a local network segment, by listening to and responding to UDP broadcasts.
44818 (TCP)	EtherNet/IP	EtherNet/IP, explicit messaging. **Note:** Used only when EtherNet/IP protocol is selected
2222 (UDP)	EtherNet/IP	EtherNet/IP, implicit messaging. **Note:** Used only when EtherNet/IP protocol is selected

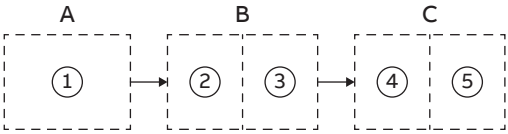
Safety functions

Stopping functions	
STO	Safe torque off
SS1-t	Safe stop 1, time-controlled

Safety data

General

To determine the SIL/PL capability of the whole safety function where the FSCS-21 is included, the failure rates (PFD_{avg}/PFH) of all components implementing the safety function must be added. See the example figure below.



A	Sensor (for example, safety light curtain or emergency push button)
B	Logic (for example, safety PLC)
C	Actuator (for example, the STO function in a drive)
1	Input device
2	PLC digital input
3	PLC [1)]
4	FSCS-21 [1)]
5	Drive STO

1) Each CIP Safety device shall incorporate the worst-case network PFH value 5.00E-10 per hour.

After calculating the total PFD_{avg}/PFH for the safety function, it must be verified that the PFD_{avg}/PFH of the safety function fulfills the requirement for the targeted SIL/PL. For more information on safety calculations, refer to standards ISO 13849-1, IEC 62061, IEC 61508, IEC 61511, or Technical guide No. 10, Functional safety (3AUA0000048753 [English]).

Basic safety data

The FSCS-21 data related to safety standards IEC 61508, IEC 61800-5-2, ISO 13849-1, IEC 61511 and IEC 62061 is listed below for the subsystem of the FSCS-21 module.

Maximum mission time (T_m) for the FSCS-21 is 20 years. After 20 years, the module must be replaced.

The given safety data applies with proof test interval T1 = 2, 5 or 20 years (low demand mode of operation). Make sure that the proof test is performed within this time (see also section Proof tests (page 218)).

This safety data applies to both safety functions, STO and SS1-t. For drive-specific STO safety data, see the drive hardware manual.

IEC 61508/62061 (high demand)	
SIL	3
SC	3
PFH (1/h) [1)]	1.15E-08
SFF (%)	>99
DC (%)	≥90
HFT	0
Component type	Type B
	3AXD10001627195 D

1) Includes worst-case extended format network PFH according to CIP Safety Specification 5.00E-10 per hour.

IEC 61508 (low demand)	
SIL	3
SC	3
PFD_{avg} (T_1=2a)	1.62E-04
PFD_{avg} (T_1=5a)	3.13E-04
PFD_{avg} (T_1=20a)	1.07E-03
SFF (%)	>99
DC (%)	≥90
HFT	0
Component type	Type B
	3AXD10001627195 D

ISO 13849	
PL	e
Cat.	3
PFH (1/h) [1)]	1.15E-08
$MTTF_D$ (a)	89
DC_{avg} (%)	≥90
	3AXD10001627195 D

1) Includes worst-case extended format network PFH according to CIP Safety Specification 5.00E-10 per hour.

Relevant failure modes

Failure modes have been considered in the design of FSCS-21. The only relevant dangerous failure mode is that the STO output is not activated on command because of an internal random hardware failure of the FSCS-21 module.

FSCS-21 implements many different diagnostics to detect internal random hardware failures. The cycle time of these diagnostics is 10 ms or less, with the exception of monitoring of the internal temperature of the device, for which the cycle time is 1 s or less. The relevant failure mode of the diagnostics is that there is a detectable fault in the safety function, but the fault reaction function is not activated because of a random hardware failure in the diagnostic system.

The estimated failure rate of the diagnostics for the FSCS-21 module:

- 232 λ_s [FIT] the safe failure rate
- 1279 λ_d [FIT] the dangerous failure rate
- 1279 $MTTF_D$ [a].

For the FSCS-21 module, there are no dangerous failures that are not detected by the diagnostics.

Related standards and directives

Related standards

Standard	Name
ISO 12100:2010	Safety of machinery – General principles for design – Risk assessment and risk reduction
ISO 13849-1:2023	Safety of machinery – Safety-related parts of control systems – Part 1: General principles for design
ISO 13849-2:2012	Safety of machinery – Safety-related parts of control systems – Part 2: Validation
IEC 60204-1:2016	Safety of machinery – Electrical equipment of machines – Part 1: General requirements
IEC 61000-6-7:2014	Generic standards - Immunity requirements for equipment intended to perform functions in safety-related system (functional safety) in industrial locations
IEC 61010-1:2010+A1:2016	Safety requirements for electrical equipment for measurement, control, and laboratory use - Part 1: General requirements
IEC 61508:2010 (Parts 1…3)	Functional safety of electrical/electronic/programmable electronic safety-related systems

Standard	Name
IEC 61511-1:2016	Functional safety – Safety instrumented systems for the process industry sector – Part 1: Framework, definitions, system, hardware and application programming requirements
IEC 61784-3-3:2016	Industrial communication networks - Profiles - Part 3-3: Functional safety fieldbuses - Additional specifications for CPF 3
IEC 61784-5-3:2018	Industrial communication networks - Profiles - Part 5-3: Installation of fieldbuses - Installation profiles for CPF 3
IEC 61800-3:2022	Adjustable Speed Electrical Power Drive Systems - Part 3: EMC requirements and specific test methods for PDS and machine tools
IEC 61800-5-1:2007	Adjustable speed electrical power drive systems - Part 5-1: Safety requirements - Electrical, thermal and energy
IEC 61800-5-2:2016	Adjustable speed electrical power drive systems – Part 5-2: Safety requirements – Functional
IEC 61918:2018	Industrial communication networks – Installation of communication networks in industrial premises
IEC 62061:2021	Safety of machinery – Functional safety of safety-related electrical, electronic and programmable electronic control systems

Related directives

Directive	Name
2006/42/EC	European Machinery Directive
2008 No. 1597	UK Supply of Machinery (Safety) Regulations 2008
2011/65/EU	European RoHS Directive
2014/30/EU	European EMC Directive

Other technical specifications

Specification name	Version
The CIP Networks Library, Volume 2: EtherNet/IP adaption of CIP. ODVA Inc.	Edition: 1.30, April 2023
The CIP Networks Library, Volume 5: CIP Safety. ODVA Inc.	Edition: 2.25, April 2023

22

Appendix A - ABB IP configuration tool

Contents of this chapter

This chapter shows how to use the ABB IP configuration tool to:

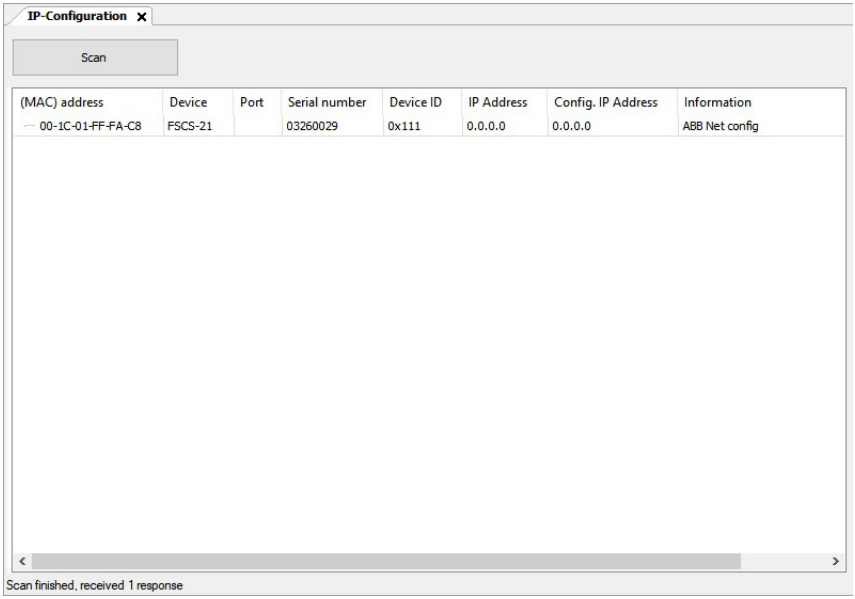
- find configured and unconfigured FSCS-21 adapter modules in the network
- rewrite the IP configuration of the adapter modules.

Installation

The ABB IP configuration tool is part of the ABB Automation Builder software. No separate installation is needed.

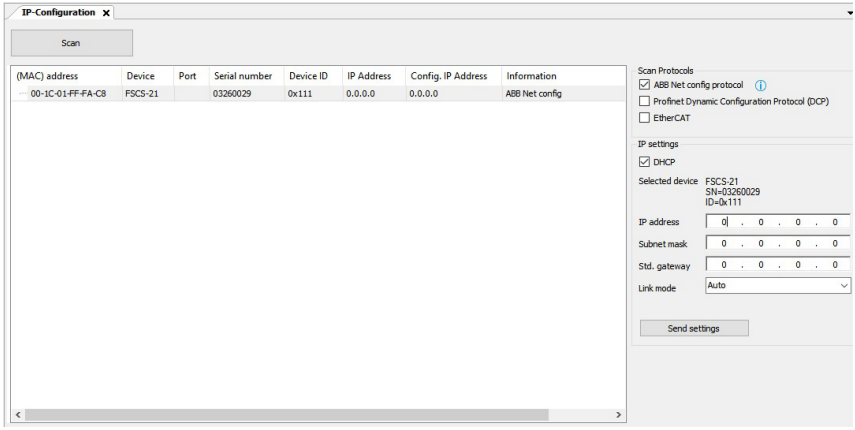
Finding adapter modules in the network

1. Open the ABB IP configuration tool.
2. Click the **Scan** button.
   The FSCS-21 adapter modules in the network are shown in the results list.

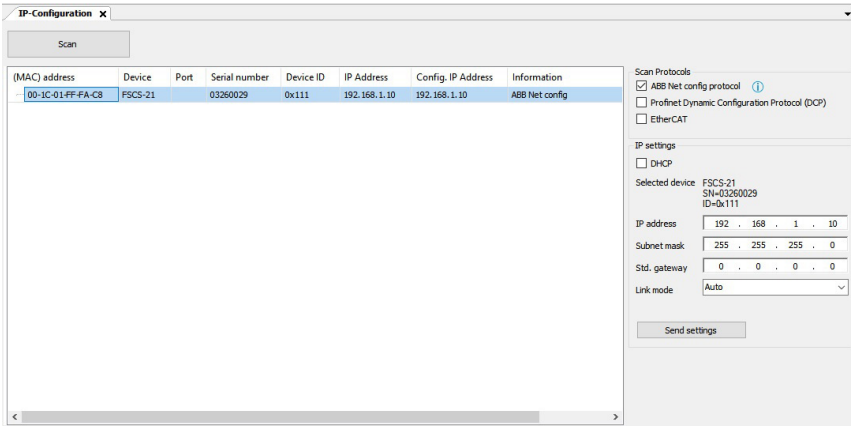


Rewriting the IP configuration of adapter modules

1. Scan the network for adapter modules.
   For instructions, refer to section Finding adapter modules in the network (page 231).
2. In the results list, select the adapter module.



3. Below **IP settings**, define the IP configuration settings according to your network configuration. If you want the adapter module to use DHCP for the IP configuration, tick the **DHCP** checkbox. If you want the adapter module to use a static IP address, clear the **DHCP** checkbox and set the IP configuration manually.
4. To apply the new settings, click the **Send settings** button.
   The new current IP address and configured IP address appear on the results list.



23

Appendix B - Module configuration backup

Contents of this chapter

This chapter presents the settings for FSCS-21 configuration backup.

Compatibility

FSCS-21 settings are stored in the drive parameters and also in the configuration files. FSCS-21 adapter module supports backup of all settings to the drive. These settings are now also included in any backups made of the drive using the Drive composer PC tool or the control panel.

Settings for backup

Consider these points:

- Backup is not slot-specific. For example, backup of FSCS-21 in FBA A, slot 1 can be restored to FSCS-21 FBA A, slot 2.
- Backup depends on the fieldbus channel. For example, backup of FSCS-21 in FBA A is not restored to FSCS-21 in FBA B.
- FSCS-21 configuration parameters are included in the backup when drive parameters are saved.

Configuration backup for all protocols in FSCS-21

The settings are saved to the drive after 10 seconds. If a Refresh command is given to FSCS-21 using parameter 51.27, the pending backup is transferred to drive immediately and FSCS-21 is rebooted after the transfer is completed.

Note: The new setting is not saved to drive if the drive was powered off or the adapter was disconnected from the drive within 10 seconds of changing a setting.

Further information

Product and service inquiries

Address any inquiries about the product to your local ABB representative, quoting the type designation and serial number of the unit in question. A listing of ABB sales, support and service contacts can be found by navigating to www.abb.com/contact-centers.

Product training

For information on ABB product training, navigate to new.abb.com/service/training.

Providing feedback on ABB manuals

Your comments on our manuals are welcome. Navigate to forms.abb.com/form-26567.

Document library on the Internet

You can find manuals and other product documents in PDF format on the Internet at www.abb.com/drives/documents.

a1 (frozen)
PDF-A5
Created 2024-05-28, 14:24:13

www.abb.com/drives

3AXD50001065478A

3AXD50001065478 Rev A (EN)
2024-05-28