SIEMENS

SIMATIC

SIMATIC ECCSIMATIC ECC TIA Library for data exchange according to CHAdeMO specification V0.9.1 to V2.0

Function Manual

11/2020

A5E50072365-AB

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER

indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠ WARNING

indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠ CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Digital Industries
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E50072365-AB
Ⓟ 01/2021 Subject to change

Table of contents

Introduction

1

This document describes how to use the TIA library LIB_CHAdeMO.

The FB_CHAdeMO function block contained in this library handles the data exchange between a charging station (EVCS) and an electric vehicle (EV) according to the CHAdeMO specification V0.9.1 to V2.0. The communication itself is based on CAN Bus.

You can download the TIA library LIB_CHAdeMO from SIEMENS Customer Support (https://mall.industry.siemens.com/mall/en/WW/Catalog/Products/10090533?tree=CatalogTree#).

In addition to this TIA library, further standard TIA components (e.g. for CAN-BUS communication) are required.

Safety instructions

2

CHAdeMO specification

Note in particular chapter 4 "General requirements" and chapter 5 "Requirements about basic design and safety design for charger and vehicle" in CHAdeMO specifications V0.9.1 to V2.0.

Description

3

3.1 Properties

The TIA library can be used for 1200 controllers, 1500 controllers and open controllers.

The "FB_CHAdeMO" FB that is contained in the TIA library and to be used by the user is protected by a licensing procedure and permanently linked to the associated CAN communication module via its serial number. This means that a separate license is required for each charging output.

The included "FB_CHAdeMO" function block handles the data exchange between a charging station (EVCS) and a vehicle (EV) according to CHAdeMO specifications V0.9.1 to V2.0. For data communication via the CAN bus, you can only use the communication modules described in chapter "Required hardware components (Page 11)".

All other functions contained in the library cannot be used directly by the user. These are called by the function block and are relevant for proper functioning of the described functionality.

NOTICE
To ensure that the reaction times are maintained, the FB must be called every 10 milliseconds. Other time intervals are not possible.

In addition, the switches "d1" (for Charge sequence signal 1) and "d2" (for Charge sequence signal 2) are controlled. Proximity detection is monitored and reacts according to the CHAdeMO specification. The vehicle charge permission is evaluated accordingly.

Additional TIA modules are required for switching and evaluating the signals. These are described in the chapter "Required hardware components (Page 11)".

The basic structure of the hardware circuitry is shown in the following figure.

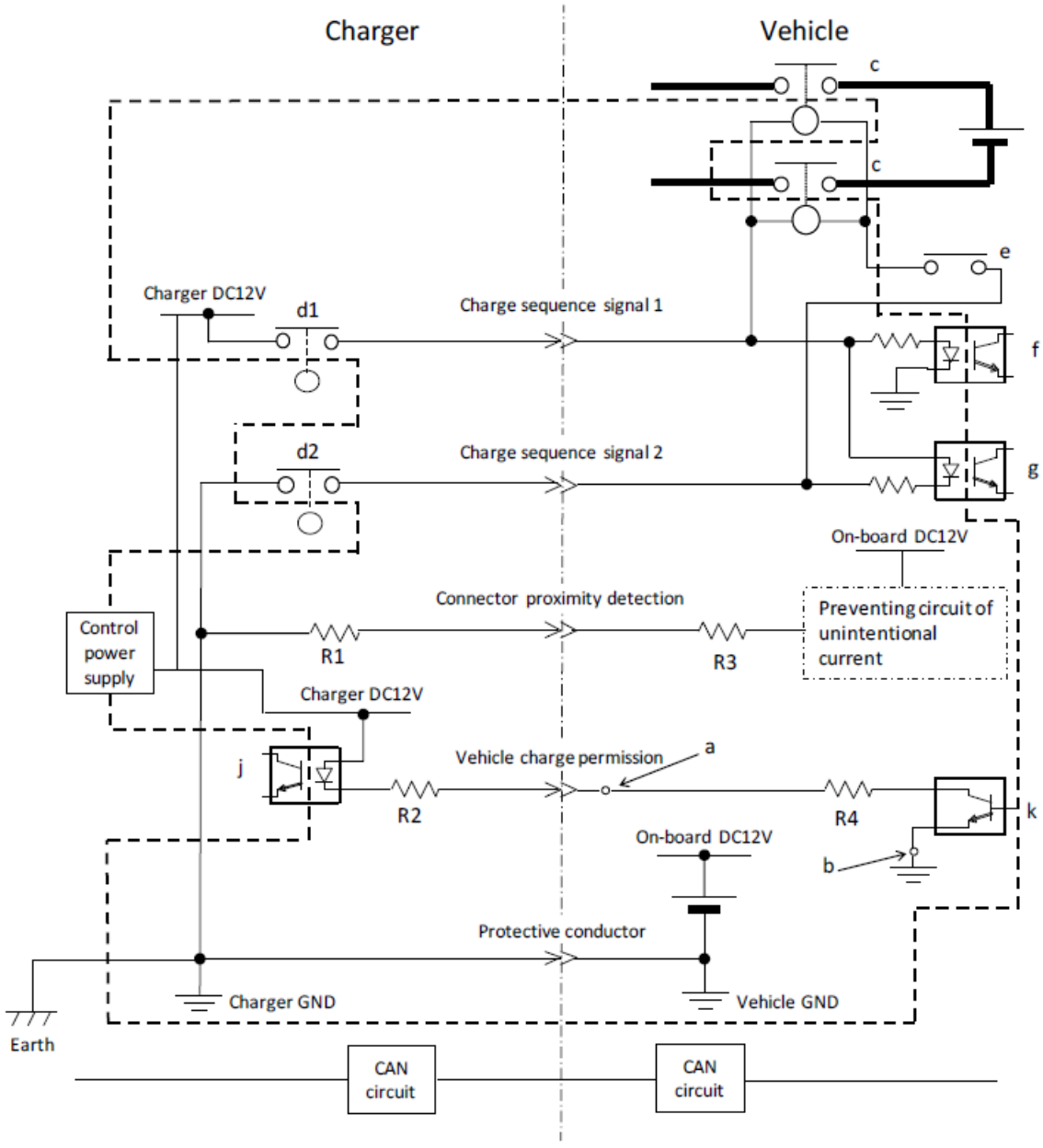

Figure 3-1 CHAdeMO schematic hardware wiring, according to V2.0 . Different wiring may apply for older versions.

Table 3- 1 Resistors in the above figure

Resistor	minimum	typical	Maximum
R1	190 ohms	200 ohms	210 ohms
R2	950 ohms	1000 ohms	1050 ohms
R3	950 ohms	1000 ohms	1050 ohms
R4	190 ohms	200 ohms	210 ohms

3.2 Required hardware components

The appropriate SIMATIC hardware components are needed to set up a charging station (EVCS) using the CHAdeMO library (LIB_CHAdeMO).

CPU

For example, SIMATIC ET 200SP CPU 1510SP-1 PN
Order number: 6ES7 510-1DK01-0AB0
See SIEMENS Industry Mall
(https://mall.industry.siemens.com/goos/WelcomePage.aspx?regionUrl=/&language=en)

For example, Open Controller
CPU 1515SP PC2
Order number: 6ES7677-2DB42-0GB0
See SIEMENS Industry Mall
(https://mall.industry.siemens.com/goos/WelcomePage.aspx?regionUrl=/&language=en)

CAN communication module (alternative)

SIMATIC PN/CAN LINK
Order number: 6BK1 620-0AA00-0AA0.
See SIEMENS Industry Mall
(https://mall.industry.siemens.com/goos/WelcomePage.aspx?regionUrl=/&language=en)

ET 200SP CM CAN
Order number: 6ES7137-6EA00-0BA0
See SIEMENS Industry Mall
(https://mall.industry.siemens.com/goos/WelcomePage.aspx?regionUrl=/&language=en)

Note

Only these two CAN modules are supported by the TIA library.

Switches "d1" and "d2" in the form of relays

(max. current to be switched: 2 ampere)

Relay module
for example: RQ NO 4x120VDC/230VAC/5A ST
Order number: 6ES7132-6HD01-0BB1

Monitoring of proximity detection

Analog input module
e.g.: ET 200SP, AI 2xU ST
Order number: 6ES7134-6FB00-0BA1

Evaluation of the vehicle charge permission

preferred:

Digital input module
e.g.: DI 8x24VDC BA
Order number: 6ES7131-6BF01-0AA0

+

Optocouplers
e.g.: 3RQ3052-2SM40

Alternative:

Analog input module
e.g.: ET 200SP, AI 2xU ST
Order number: 6ES7134-6FB00-0BA1
Note that "Overflow" diagnostics is **not** activated for this module.

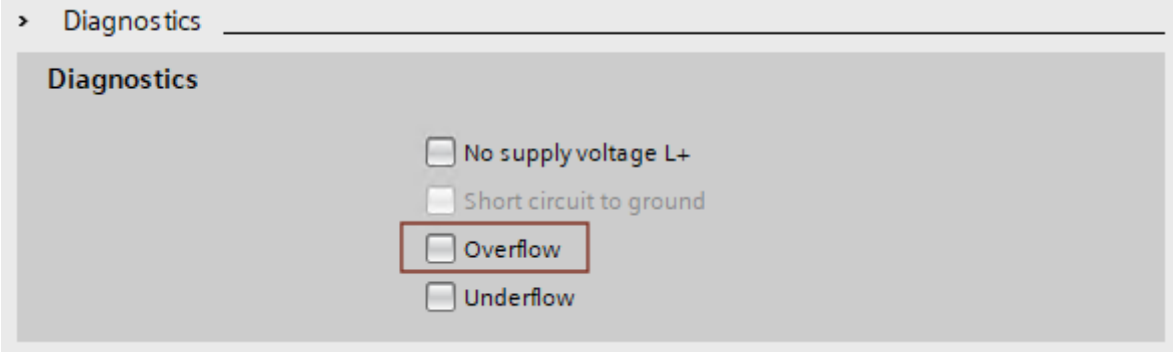

Figure 3-2 Diagnostics - overflow

Connector locking

Additional digital modules may be required to trigger and read back the connector lock.

Contactor control

Digital input module
e.g.: DQ 4x24VDC/2A ST
Order number: 6ES7132-6BD20-0BA0

3.3 Description of the hardware setup

Various possible hardware setups are described below as examples.

Please note that the resistors shown, R1 and R2, are not included in any of the modules used. These must be supplied by the customer.

The first example describes the use of the SIMATIC PN/CAN Link and the evaluation of the charge permission via analog input:

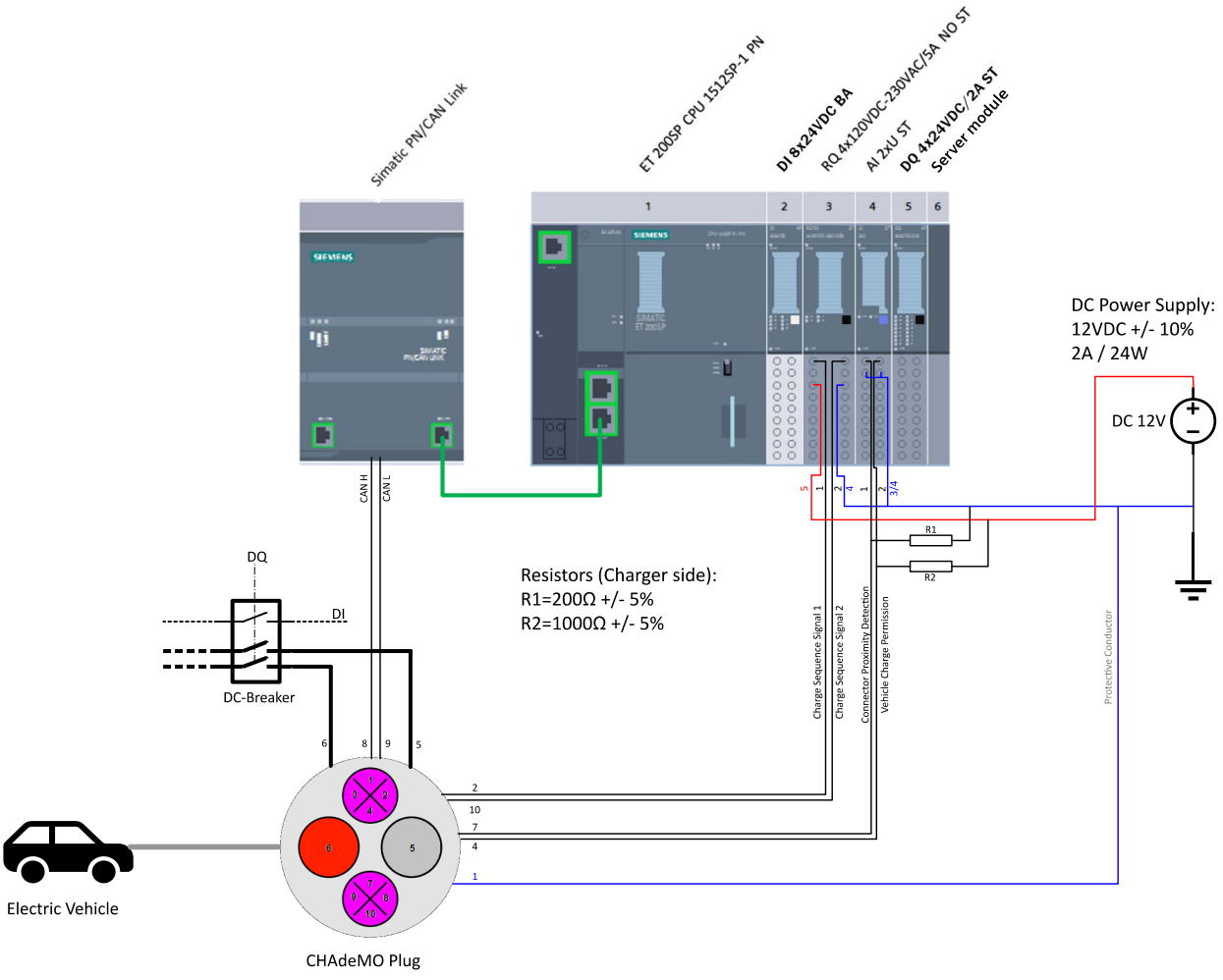

Figure 3-3 Typical circuit diagram with SIMATIC PN/CAN link and evaluation of the charge permission via analog input

An open controller with distributed I/O devices and CM CAN module is used in this typical circuit diagram.

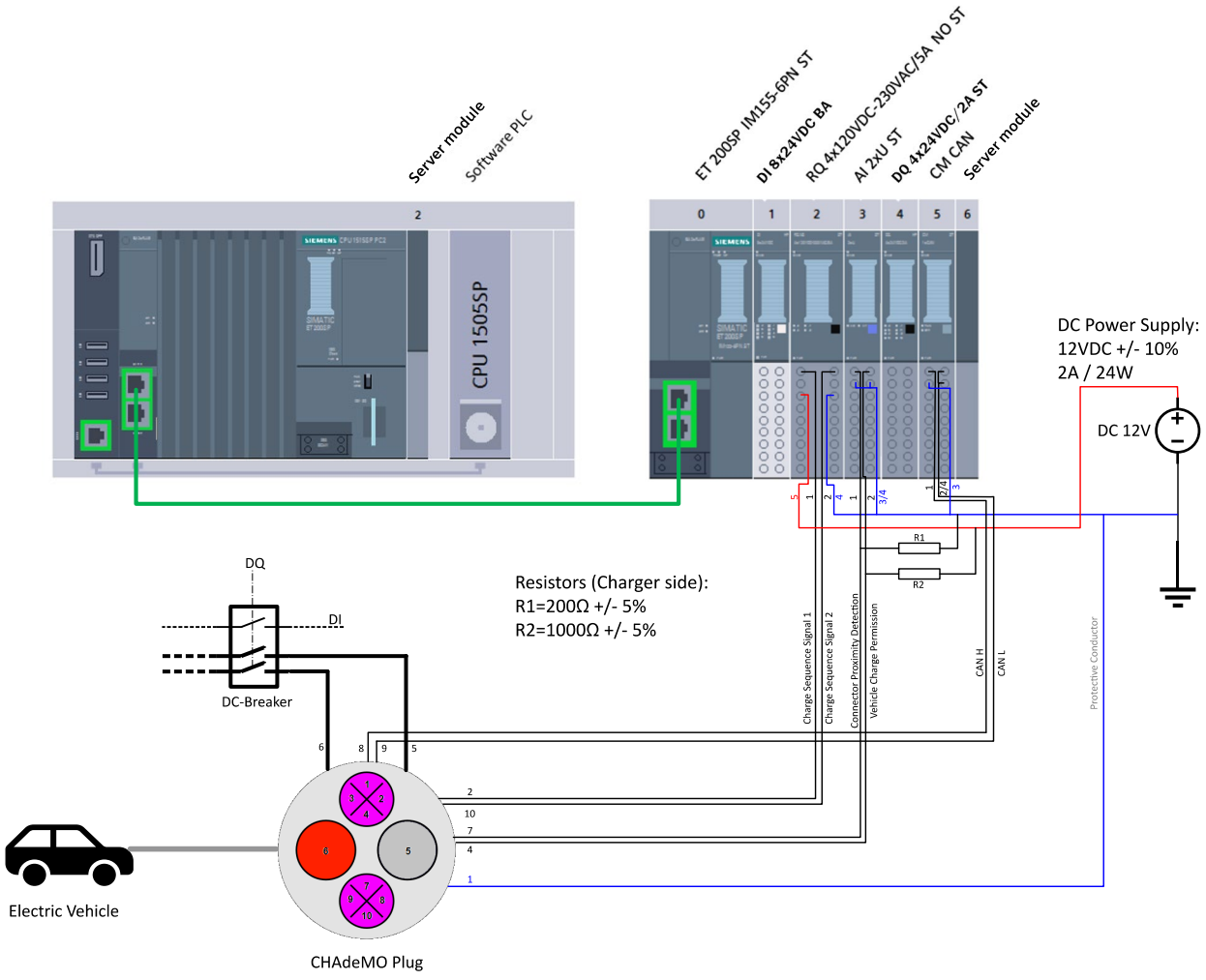

Figure 3-4 Typical circuit diagram with CM CAN module and open controller with distributed I/O devices

If you use an optocoupler, you must insert an additional resistor.

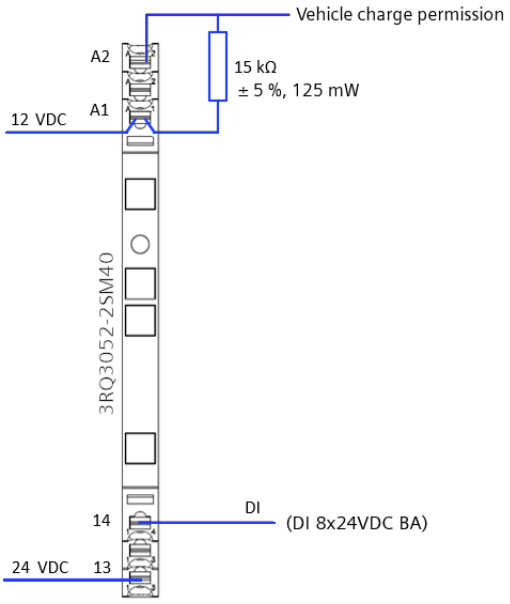

Figure 3-5 Circuit with additional resistor (15 kOhm, 0.125 W, 5%), when using the 3RQ3052-2SM40 optocoupler.

The following circuit diagram shows where the measured values "presentOutputVoltage" and "presentCurrent" are recorded.

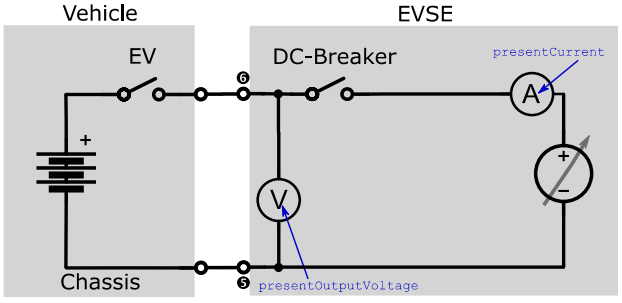

Figure 3-6 Positions of the measurement locations

The structure of the CHAdeMO charging plug is described below:

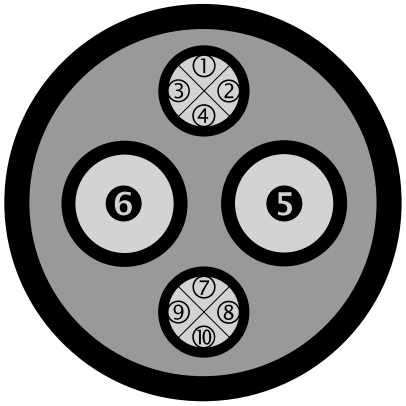

Figure 3-7 CHAdeMO charging plug

Table 3- 2 Contact assignment of CHAdeMO charging plug

Pin	Description
1	GND
2	Charge sequence signal 1
3	N/A
4	Vehicle charge permission
5	Power supply (-)
6	Power supply (+)
7	Connector proximity detection
8	CAN-H
9	CAN-L
10	Charge sequence signal 2

3.4 Charge communication

The CHAdeMO specification describes the digital communication between a DC charging station and an electric vehicle to control the DC charging. The charging application communicates with the vehicle via CAN bus (point-to-point) using the TIA library LIB_CHAdeMO. In addition, the charging station (EVCS) communicates with the electric vehicle (EV) via the specified enabling signals.

Communication between the vehicle (EV) and the charging station (EVCS) takes place on both sides by cyclic transmission of CAN PDUs.

The charging process is divided into the following stages:

- Start CAN communication
- Information exchange before the charging session
- Insulation test in the power circuit
- Battery connection (EV contactors are closed)
- Charging control process
- Termination of charging process with welding detection
- Disconnection of CAN communication

The figure below shows the individual steps of the communication between the charging station (EVCS) and the vehicle (EV) in chronological order.

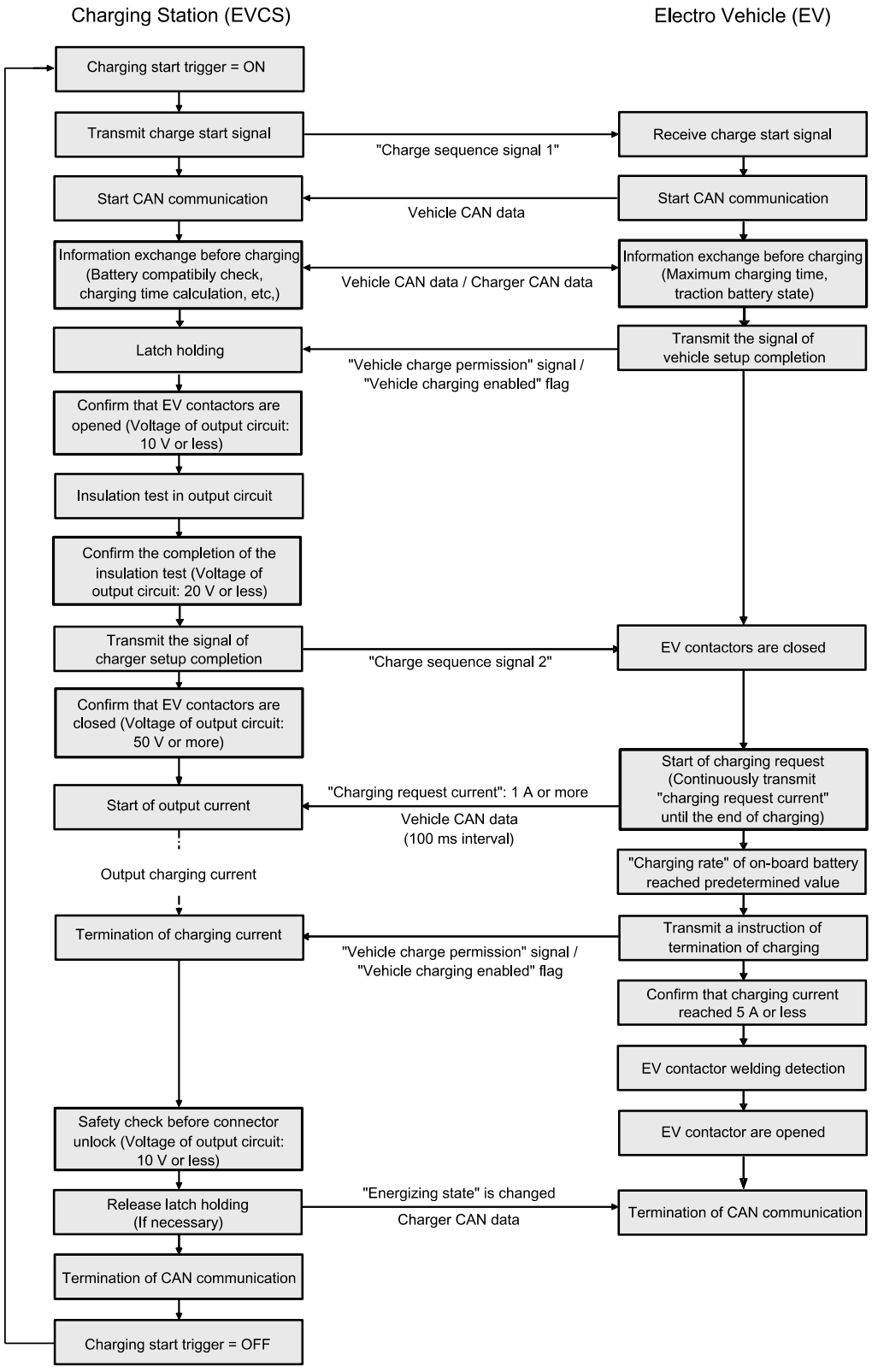

Figure 3-8 CHAdeMO V2.0 charging phases. Other schematics may apply for versions V0.9.1 to V2.0.

3.4.1 CAN communication setup

Recognition of the analog "Connector proximity detection" input signal by the charging station (EVCS) and the corresponding release by the charging application is a requirement. CAN communication starts after the EVCS has closed the switch "d1" (Charge sequence signal 1).

3.4.2 Information exchange before the charging process

The necessary parameters are exchanged here between EVCS and EV.

Among others, these include:

- Compatibility test of the battery parameters (maximum charging current, minimum/maximum charging voltage, maximum charging power, etc. during the entire charging process)
- Calculation of charging time (maximum charging time)

3.4.3 Isolation test in power circuit

In the insulation test phase, the check for the employed charging cable according to IEC 61851-23 must be initiated for DC charging.

The CHAdeMO specification requires an insulation resistance of at least 20 kOhm. If this value is undershot, the charging process is aborted with an error message.

3.4.4 Battery connection

The connection of the load voltage is done on the charging station side via the DC breaker schematically represented in the figure "Position of the measuring point (Page 15)". Since the control of the breaker can differ depending on the design of the charging station (EVCS), you need to configure it in the user program.

Note that the "close" command overwrites the "open" command.

The control of the contactor in the EV is not within the scope of the charging station application, but is instead performed by the CHAdeMO implementation in the vehicle using the corresponding CAN signals.

3.4.5 Charging control process

This stage begins when the EV requests a charging current of 1 ampere or more. The EVCS now supplies the required charging current (chargingCurrentReq).

3.4.6 Termination of charging process

Normally the charging process is terminated by the vehicle (EV).

Other criteria that lead to (normal) termination are, for example:

- Leaving the parking position (shift lever in the EV)
- Remaining charging time is zero (from the EVCS)
- Ending the charging process at the charging station (stop button)

The charging process is also terminated (abnormally) if, for example

- CAN communication fails,
- a HW error of the EVCS is detected (such as emergency stop, door open),
- the battery reports overvoltage/undervoltage/overtemperature,
- a ground fault is detected.

3.4.7 Welding Detection

After the charging process is completed, the DC disconnectors in the vehicle are checked for "welding". Within a defined period, the measured voltage (presentOutputVoltage) must fall below 10 V by disconnecting the battery from the EV (see also Positions of the measuring points (Page 15)).

3.4.8 Session Stop

After successful completion of the welding detection stage, the connector lock is opened.

After the EV has terminated communication, EVCS opens switches "d1" and "d2" and deactivates the CAN communication.

Configuring the CAN interface

4

4.1 Introduction

You must configure the CAN communication module (PN/CAN-Link or ET200SP CM CAN). The configuration of the PN/CAN link is described below. The configuration of the ET200SP CM CAN module works in a similar manner.

Documentation on

SIMATIC PN/CAN LINK (https://support.industry.siemens.com/cs/de/en/ps/6BK1620-0AA00-0AA0/man)

ET 200SP CM CAN (https://support.industry.siemens.com/cs/de/en/ps/6ES7137-6EA00-0BA0)

System environment

You can use the SIMATIC ECC CHAdeMO function block in the following system environments:

Table 4- 1 System environments

Applications	Required components	Configuration software	In your program
Distributed operation in an S7-1200 system	Automation system S7-1200 (from FW V4.0) ET 200SP Distributed I/O System	STEP 7 (TIA Portal): Device configuration and parameter settings with the hardware configuration (HWCN)	Access only via the function block call FB_CHAdeMO
Distributed operation in an S7-1500 system	Automation system S7-1500 (FW V2.0 or higher) ET 200SP Distributed I/O System	STEP 7 (TIA Portal): Device configuration and parameter settings with the hardware configuration (HWCN)	Access only via the function block call FB_CHAdeMO
Centralized or distributed operation in an ET 200SP system	ET 200SP automation system (FW V2.0 or higher)	STEP 7 (TIA Portal): Device configuration and parameter settings with the hardware configuration (HWCN)	Access only via the function block call FB_CHAdeMO

4.2 Configuring the CAN communication module (example PN/CAN link)

4.2.1 General setting

CAN operating mode

Select the operating mode "CAN Bus transparent"

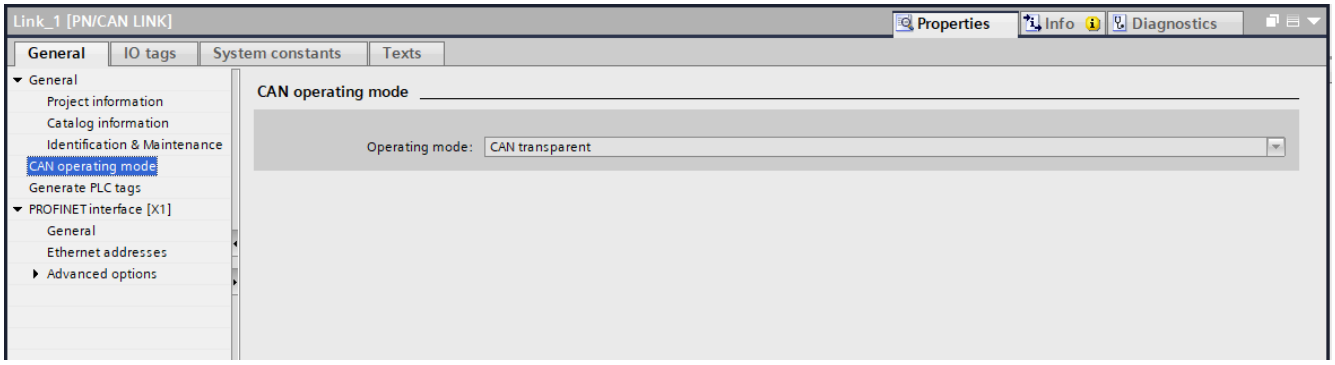

Figure 4-1 Operating mode: CAN Transparent

CAN baud rate

Set a baud rate (transmission rate) of 500 Kbps.

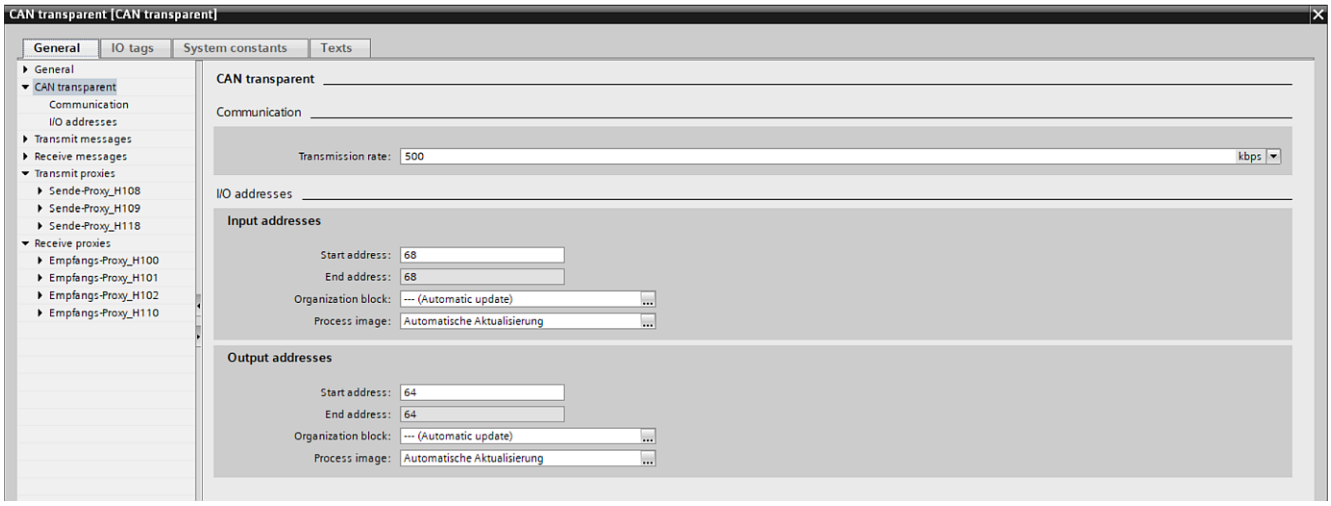

Figure 4-2 Setting the CAN baud rate

PROFINET interface update time

Set a update time of 2.0 ms here.

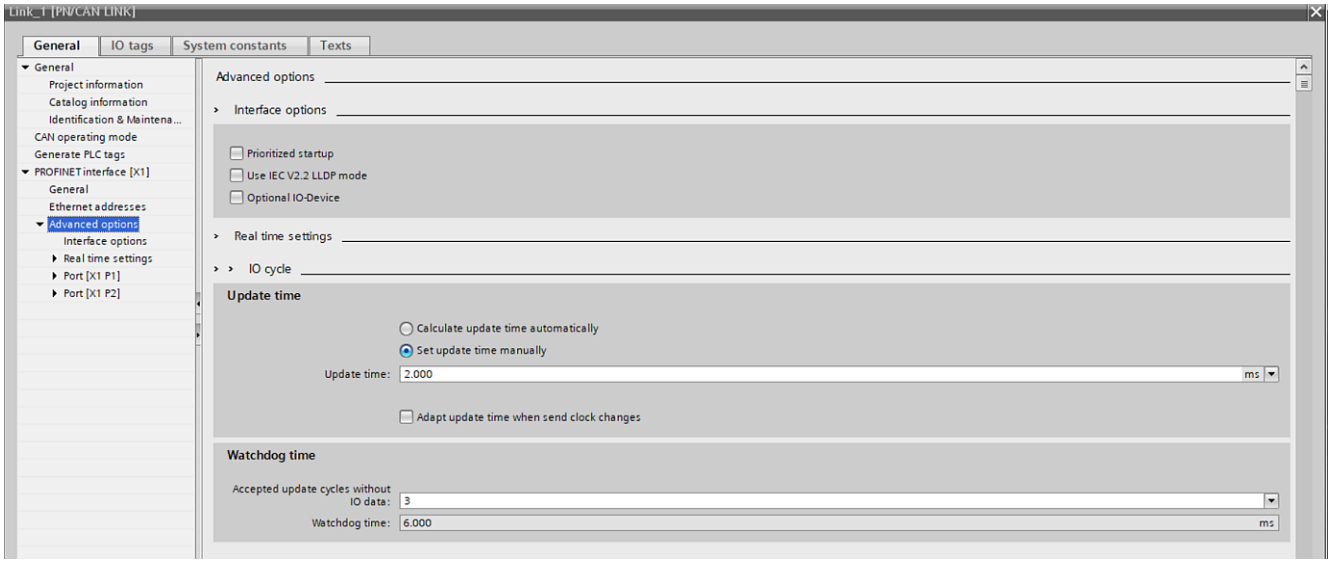

Figure 4-3 PN update time

4.2.2 Setting the transmit and receive proxies

No transmit and receive messages are configured. Communication takes place exclusively via transmit and receive proxies.

Transmit proxies

You must create three transmit proxies for the CAN IDs H'108, H'109 and H'118 with the "default settings".

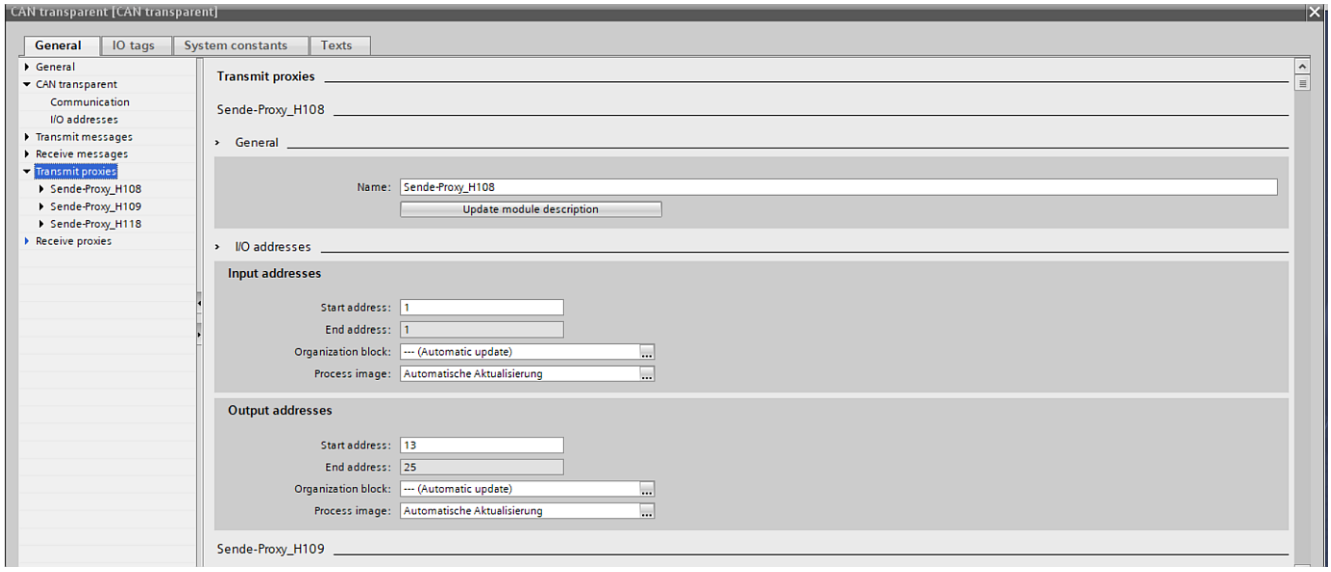

Figure 4-4 Transmit proxies

Receive proxies

You must configure four receive proxies. Use H'100 for the CAN ID with the settings shown here. For the CAN IDs H'101, H'102 and H'110 you have to similarly adapt the bit check of the ID.

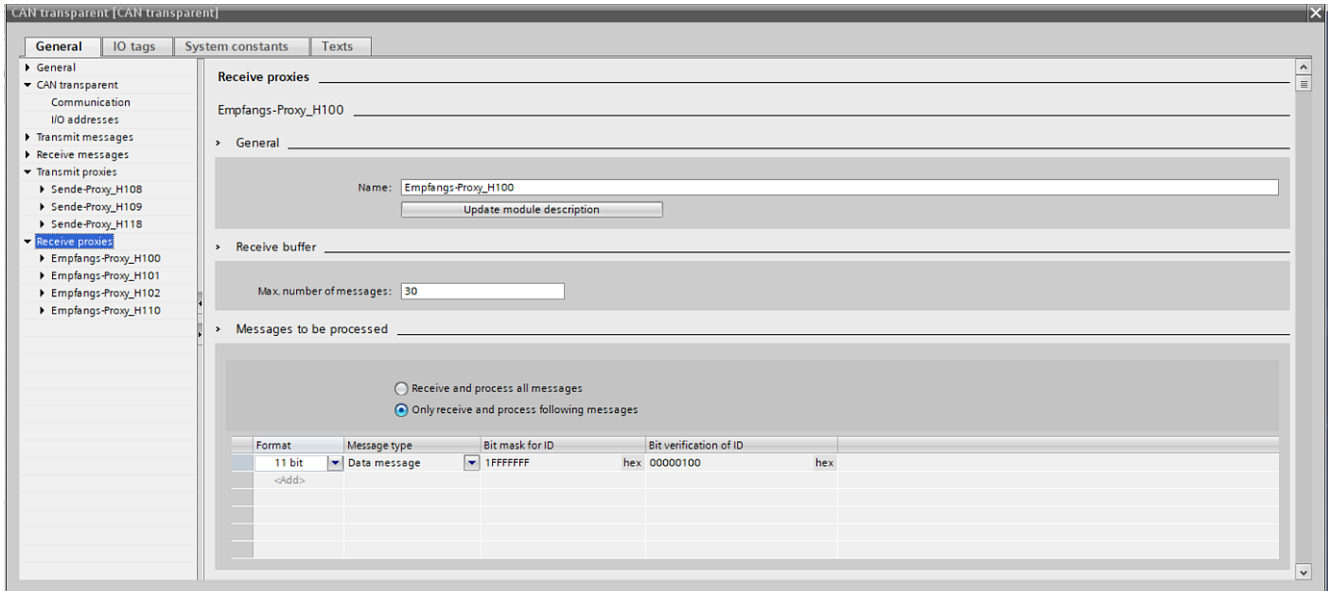

Figure 4-5 Configuration of the receive proxies (for example, H'100)

Configuration of the AI module

Measuring range or value range: 0 .. 10 V DC

Signal name	Values
Analog signal "Connector proximity detection" as connection detection between EV and EVCS.	• < 1.2 V: not connected • ≥ 1.2 V: connected
"Vehicle charge permission" of the vehicle via analog signal. **Note:** If you use the "vehicleChargePermissionDI" input for the charge release, you must set "vehicleChargePermission-AI" to a value > 3 V.	• ≤ 3 V: Charge permission from vehicle • > 3 V: No charge permission from vehicle

Programming

5

5.1 General information

This TIA library cannot be used without a valid license key.

TIA Library supports CHAdeMO versions V0.9.1 to V2.0.

Ordering the license key

Compile and load your program into the CPU and start it. The output parameter relevant for the order information is "CSInputData.identCode".

The content of this parameter is the order information for the license key.

After receiving the license key, enter it as value at the "HWConfig.licenceKey" input parameter of the "FB_CHAdeMO" function block.

Restart the CPU. The output parameter "CSInputData.FBAuthentificated" should now have the value "True". The function block is then fully functional.

Note

The license is linked to the associated CAN module.

If the CAN module is replaced by another one, a new license key must also be ordered, since the output parameter "CSInputData.identCode" changes according to the ident code at the output parameter.

5.2 The function block "FB_CHAdeMO"

Introduction

Below you can see the inputs and outputs of the function block "FB_CHAdeMO".

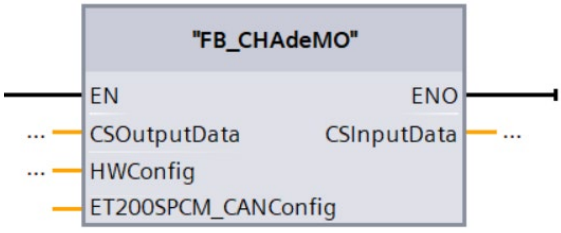

Figure 5-1 Function block FB_CHAdeMO

The function block "FB_CHAdeMO" can be found in the TIA library "LIB_CHAdeMO". For the 1500 controllers you must use the FB from the folder "LIB_CHAdeMO → Types → CHAdeMO".

For the 1200 controllers you must use the FB from the folder "LIB_CHAdeMO → Types → CHAdeMO_1200".

If you are using an Open Controller, you must use the function block from the folder "LIB_CHAdeMO → Types → CHAdeMO_OC".

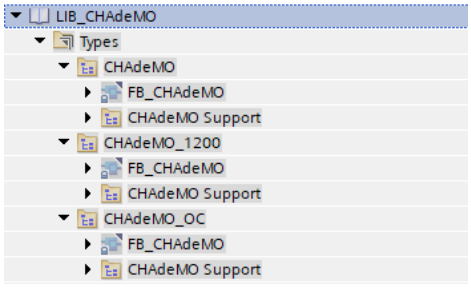

Figure 5-2 Structure of the TIA library

For programming you only have to use the function block "FB_CHAdeMO" itself. The "FB_CHAdeMO Support" folder contains internal help functions and is not relevant for the user.

5.3 Mode of operation of the FB_CHAdeMO function block

The function module contains the main CHAdeMO functionality. It serves as a link between the charging station application and an electric vehicle which can be charged according to the CHAdeMO specifications V0.9.1 to V2.0.

It comprises the following functionalities:

- Initialization (authentication, PN-CAN LINK HW-IO addresses, ...)
- Reading inputs from the charging application
- Updating outputs to the charging application
- Updating the CAN communication status
- Reading CAN input data from EV
- Transmitting CAN output data to EV
- Updating the error status
- Performing P-N insulation test (once before charging)
- Update CHAdeMO connector status
- Updating charging status of machine (the actual charging process)

Responding to errors

See also the Error codes (Page 39) output parameters "CSInputData.EVCSError".

There are two error cases

- Errors of the charging station can be signaled via the input parameter "CSOutputData.CSError".
  For example: Emergency stop, door open, ...
- Error during the charging process
  An error from the EV or an error detected by the FB is reported via the output parameter "CSInputData.EVCSError" and the charging process is terminated according to the CHAdeMO specification.

CAN communication setup

The first step is to check whether the charging cable is connected to the vehicle. This is detected according to the CHAdeMO specification using "Connector proximity detection". The output parameter "CSInputData.connProximityDetection" is set to "True" when the connection is detected. Otherwise this parameter has the value "False".

After the connection has been detected, CAN communication can be activated by setting the input parameter "CSOutputData.chargeOn" to "True".

CAN communication starts after the EVCS has closed the switch "d1" (Charge sequence signal 1). The switch is closed by setting the output parameter "CSInputData.chargeSeqSignal_1RQ" to "True".

Information exchange before the charging process

For example, the following parameters are exchanged between EVCS and EV.

- Protocol number check
- Battery Compatibility
  The target voltage of the battery must not be greater than the maximum permitted battery voltage.
  The target battery voltage must not be greater than the maximum available output voltage of the converter.
  The minimum available output voltage of the converter must not be greater than the minimum battery voltage
  The minimum charging current must not be greater than the maximum available output current of the converter.
- Evaluation of the "Extended Functions" (only for CHAdeMO specification V2.0) by the EV and comparison with EVCS
- Charging time calculation:
  The maximum charging time is the minimum of the maximum charging times of the EV and the charging station (EVCS). However, it may not exceed 4.25 h.

Isolation test in power circuit

In the insulation test stage, the check of the charging cable used (e.g. insulation resistance) must be initiated for DC charging. The check is controlled by the function block "FB_CHAdeMO". The charging application provides the test voltage and supplies the current and voltage measured values for the resistance calculation in the FB.

First, the value of the "presentOutputVoltage" is determined. This must be ≤ 20 V. Then the application activates the DQ of the ET200SP which in turn triggers and closes the DC breaker.

The test is activated by the output parameter "CSInputData.PNInsulationTestReq".

The test voltage to be set on the converter is specified via the output parameter "CSInputData.PNInsulationTestVoltage".

The insulation resistance is calculated based on the "presentOutputVoltage" and the measured current ("CSOutputData.presentCurrent" input parameter).

The resistance is determined after a configurable delay time (input parameter "CSOutputData.PNInsulationTestDelayTime").

The end of the test is signaled by setting the output parameter "CSInputData.PNInsulationTestReq" to "False". Then the voltage at the converter must be reset to the minimum voltage and the DQ must be reset (DC breaker = open).

If an error occurs during the measurement of the insulation, the output parameter "CSInputData.PNInsulationFault" is set to "True" and the charging process is terminated.

Battery connection

The connection of the load voltage is done on the charging station side via the DC breaker schematically represented in the figure"Position of the measuring point (Page 15)". Since the control of the breaker can differ depending on the design of the charging station (EVCS), you need to configure it in the user program.

The position of the DC breaker and the target voltage to be applied when the DC breaker is closed can be derived from the two following output parameters of FB_CHAdeMO.

Signal	Value	DC breaker	Voltage
PNInsulationTestReq	TRUE	Close	Insulation test voltage
	FALSE	Open	-
chargeSeqSignal_2RQ	TRUE	Close	Charging voltage
	FALSE	Open	-

Note that the "close" command overwrites the "open" command.

The control of the contactor in the EV is not within the scope of the charging station application, but is instead performed by the CHAdeMO implementation in the vehicle using the corresponding CAN signals.

Charging control process

This stage begins when the EV requests a charging current of 1 ampere or more. The "presentOutputVoltage" parameter is then checked. A voltage ≥ 50 V must be measured there. Now the DQ of the ET200SP (DQ4x24VDC/2A ST) is activated in the application. This closes the DC breaker. The EVCS now supplies the required charging current according to the CHAdeMO specification.

This charging current must be specified at the input parameter "CSOutputData.presentCurrent". The same applies to the current charging voltage ("CSOutputData.presentOutputVoltage").

Charging with dynamic response ("Dynamic Control" enhancement)

If dynamic charging is supported by EVCS and EV (output parameter "CSInputData.extendedFunction1"), EVCS may vary e.g. the maximum current (input parameter "CSOutputData.maxAvailableOutputCurrent"). If the level falls below the current charging current, the EV adjusts the current request (output parameter "CSInputData.chargingCurrentReq").

Once the maximum current of the EVCS remains constant, the EVCS must/can initiate a recalculation of the remaining charging time in the EV (input parameter "CSOutputData.CMDResEVMaxChargingTime"). The recalculation of the remaining charging time takes approx. 10 s and is signaled as status (input parameter "CSInputData.ResEVMaxChargingTimeActive").

Termination of charging process

An error from EV or EVCS always terminates the charging process immediately and starts the corresponding termination sequence. Any errors that have occurred are stored in the output parameter "CSInputData.EVCSError".

However, the charging process is normally terminated by the EV or EVCS by the following events.

Termination by the EV:

- Current request from EV = 0 A
- Charge permission withdrawn by EV
- Shift lever position not in "parking" position

Termination by the EVCS:

- Stop button pressed (the input parameter "CSOutputData.chargeOn" is set to "False" during the charging process)
- The remaining charging time has expired (0 s).
- The minimum charging current from the EV is permanently higher than the maximum available charging current of the EVCS

The charging application must then set the charging current to "0 A" ("CSOutput.presentCurrent" input parameter), the DQ is reset (DC breaker = open) and the EV in turn opens the battery contacts ("CSInputData.EVContactorsClosed" output parameter goes to "False").

As a check, the "presentOutputVoltage" parameter is now read out. A voltage ≥ 10 V must be measured there.

In this state, the EVCS opens the connector lock ("CSInputData.latchHoldingRQ" output parameter = "False") and the CAN communication is terminated (see also figure of the Charging phases (Page 18)).

The EVCS opens the switches "d1" (output parameter "CSInputData.chargeSeqSignal_1RQ" is set to "False") and "d2" (output parameter "CSInputData.chargeSeqSignal_2RQ" is set to "False").

CAN communication is deactivated after the EV has finished communication.

The input parameter "CSOutputData.chargeOn" must be reset to "False".

Note

Any pending errors must be evaluated beforehand, as resetting the input parameter "CSOutputData.chargeOn" also clears pending errors.

5.4 FB_CHAdeMO parameter

The FB has the following input parameters

- CSOutputData (Data type: typeCHAdeMORead),
- HWConfig (Data type: typeHWConfig),
- ET200SPCM_CANConfig (Data type: Variant)

and the following output parameters

- CSInputData (data type: typeCHAdeMOWrite).

These parameters are described in detail below.

5.4.1 Input parameter ET200SPCM_CANConfig: Variant

When using the ET 200SP CAN module "ET 200SP CM CAN", a data block with the following name is created as soon as the corresponding module is configured in TIA:

<PLC name>.<Module name>

For an example, see the following figure (data block name "<IODeviceName>.<ModuleName>"):

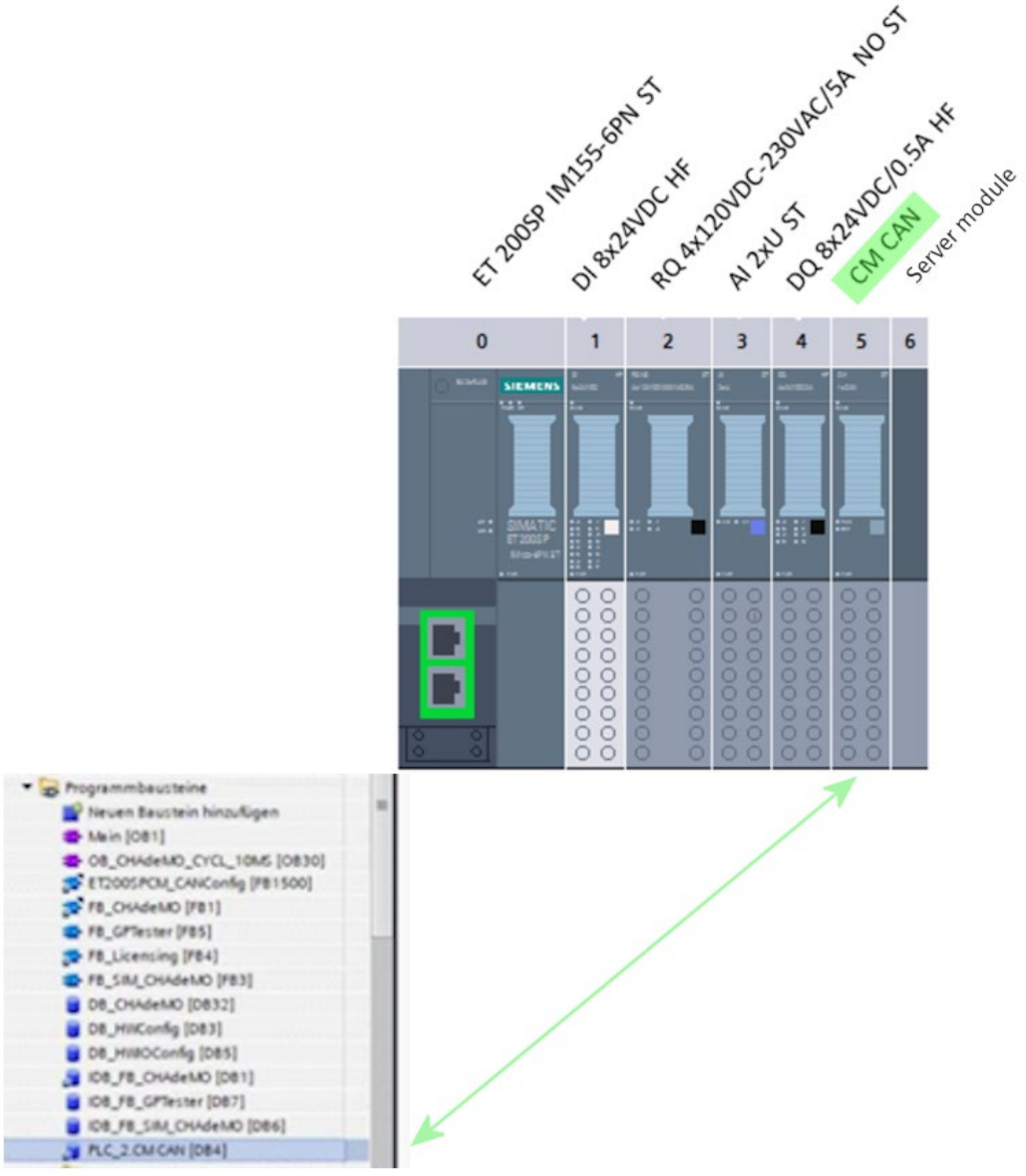

Figure 5-3 Data block name

This automatically generated data block must then be transferred to this input parameter.

If the CAN module "SIMATIC PN/CAN Link" is used, "Zero" must be transferred

5.4.2 Input parameter CSOutputData: typeCHAdeMORead

The typeCHAdeMORead is structured as follows:

Table 5- 1 List of input parameters for the typeCHAdeMORead

Parameter	Data type	Description
chargeOn	Bool	A charging process can be started with an edge change from 0 to 1. When a charging process is finished, this parameter must be reset to 0. Resetting this parameter also acknowledges and resets all errors. If this parameter is reset to 0 during an ongoing charging process, the charging process is immediately terminated normally.
CMDResEVMaxChargingTime	Bool	Recalculation of the remaining charging time of the electric vehicle (EV). To recalculate the charging time, the input must be set to "True" for at least 200 ms. It must then be reset to "False".
connectorLatchHoldingStatusDI	Bool	Feedback of the connector lock False: Unlocked True: Locked This signal may only differ from "latchHoldingRQ" for a maximum of 3 s, otherwise an error is output.
connectorLatchMalfunctionDI	Bool	Malfunction of the connector lock (optional) True: Malfunction
connectorProximityDetectionAI	Real	Analog signal "Connector proximity detection" (in V) Connection detection between EV and EVCS • < 1.2 V: not connected • ≥ 1.2 V: connected
CSError	Byte	Error message from the charging station > 0: Fault in the charging station (the charging process is terminated abnormally)
extendedFunction1	Byte	Extended functions of the charging station (True = selected; False = not selected) • Bit 0 = Dynamic current control is supported • Bit 1 = High current charging (201A – 400 A) is supported • Bit 2 = High voltage charging (601V – 1200 V) is supported
featureFlagsExt	Byte	Operating properties for the extension to high voltage charging (True = selected; False = not selected) • Bit 0 = Current charging current > Rated cable current (dyn.) • Bit 1 = Cooling function to protect cable temperature available • Bit 2 = Cable charging current protection available • Bit 3 = Connector temperature protecting cooling function available • Bit 4 = Connector for charging current protection available • Bit 5 = Temperature controlled connector lock available • Bit 6 = Safe charging current cut-off and error stop on detection of cable or connector errors

GNDInsulationOK	Bool	Status of the insulation monitoring of the charging station (ground fault) True: Insulation test passed False: Insulation fault **Note:** As long as the output parameter "GNDFaultTestDisable" is set to "True", no insulation monitoring may take place. The value of this parameter must be set to "False"
maxAvailableOutputCurrent	UInt	Maximum possible charging current of the converter (in A)
maxAvailableOutputPower	UInt	Maximum possible charging power of the converter (in W)
maxAvailableOutputVoltage	UInt	Maximum possible output voltage of the converter (in V)
maxChargingTime	UInt	Maximum charging time in seconds The default value is 3600 (corresponds to 3600 seconds) The value must be ≤ 15300. If it is greater, error bit 15 is set at the "EVCSError" parameter. A charging process is not possible.
minAvailableOutputVoltage	UInt	Minimum possible output voltage of the converter (in V)
PNInsulationTestDelayTime	USInt	Delay time until the insulation resistance is measured in steps of 0.1 s. The default value is 50 (corresponds to 5 s) After this delay time the resistance is calculated from "presentOutputVoltage" and "presentCurrent". In order to start charging, this must be at least 20 Kilohm.
presentCurrent	DInt	Current charging current of the converter (in mA)
presentOutputVoltage	UInt	Actual output voltage at the converter (in V)
tempSensorMalfunctionDI	Bool	Malfunction of the temperature sensor (optional) True: Malfunction
vehicleChargePermissionAI	Real	"Vehicle charge permission" of the vehicle via analog signal (in V) • ≤ 3 V: Charge permission from vehicle • > 3 V: No charge permission from vehicle **Note:** If you use the "vehicleChargePermissionDI" input for the charge release, you must set "vehicleChargePermissionAI" to a value > 3 V.
vehicleChargePermissionDI	Bool	"Vehicle charge permission" of the vehicle via digital signal If you use an optocoupler, the unlocking digital value of "Vehicle charge permission" must be set here. False: No charge permission from vehicle True: Charge permission from vehicle **Note**: If you use the input "vehicleChargePermissionAI" for the charge permission, you must set "vehicleChargePermissionDI" to False.

5.4.3 Input parameters HWConfig: typeHWConfig

The typeHWConfig is structured as follows:

Table 5- 2 List of input parameters for the typeHWConfig

Parameter	Data type	Description
ET200SPCM_CAN	HW_IO	HW ID of the CAN communication module used. Only relevant when using the "ET200SP CM CAN" module. If the "SIMATIC PN/CAN Link" CAN module is used, this parameter must be set to 0.
licenseKey	String	License key for activating the function module.
PNCLCANHead	HW_IO	<Module name>~Head
PNCLCANTransparentNetwork	HW_IO	<Module name>~CAN_transparent~CAN-Network
PNCLsendProxyH108	HW_IO	<Module name>~CAN_transparent~Send-Proxy_H108
PNCLsendProxyH109	HW_IO	<Module name>~CAN_transparent~Send-Proxy_H109
PNCLsendProxyH118	HW_IO	<Module name>~CAN_transparent~Send-Proxy_H118
PNCLreceiveProxyH100	HW_IO	<Module name>~CAN_transparent~Receive-Proxy_H100
PNCLreceiveProxyH101	HW_IO	<Module name>~CAN_transparent~Receive-Proxy_H101
PNCLreceiveProxyH102	HW_IO	<Module name>~CAN_transparent~Empfangs-Proxy_H102
PNCLreceiveProxyH110	HW_IO	<Module name>~CAN_transparent~Receive-Proxy_H110

The corresponding values (HW identifier) can be transferred from the "Device view of the CAN module → Properties".

The figure below shows an example of a configuration. The module name is "Link_1"

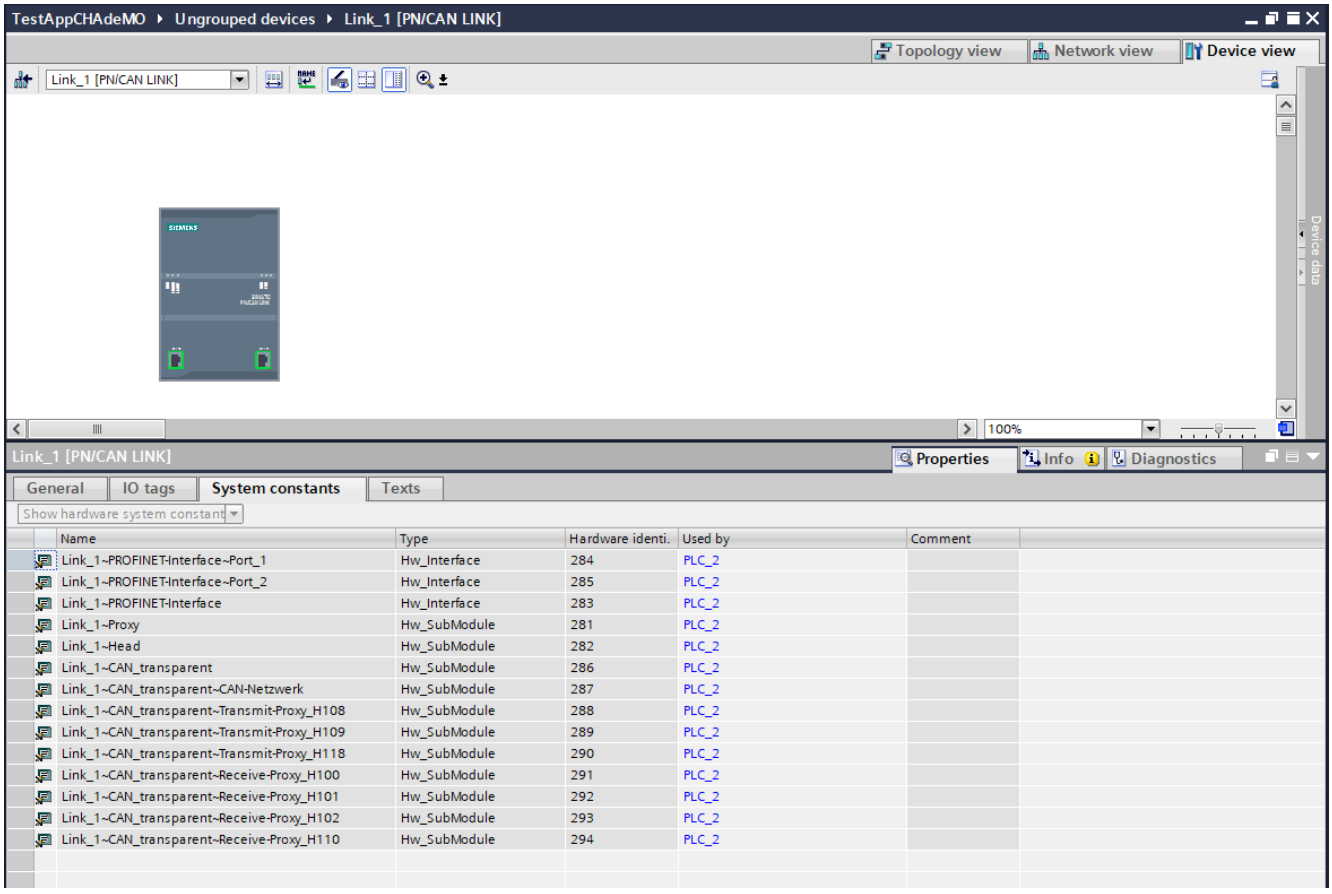

Figure 5-4 Device view of the CAN module

5.4.4 Output parameter CSInputData: typeCHAdeMOWrite

The typeCHAdeMOWrite is structured as follows:

Table 5- 3 Listing of the output parameters for typeCHAdeMOWrite

CHAdeMO versions	**Parameter**	**Data type**	**Description**
V0.9.1 .. V2.0	CHAdeMOProtNum	Byte	Employed protocol version: default 255 or 0xFF • 1: V0.9.1 • 2: V1.0.0, V1.0.1, V1.1, V1.2 • 3: V2.0
V0.9.1 .. V2.0	chargeSeqSignal_1RQ	Bool	Switching the switch "d1" (Charge sequence signal 1). This output parameter is used to open or close switch "d1". False: Open switch True: Close switch
V0.9.1 .. V2.0	chargeSeqSignal_2RQ	Bool	Switching the switch "d2" (Charge sequence signal 2). This output parameter is used to open or close switch "d2". False: Open switch True: Close switch
V0.9.1 .. V2.0	chargingActive	Bool	Charging process active. "Batch" according to d2 (completion of insulation test) False: No True: Yes
V0.9.1 .. V2.0	chargingCurrentReq	UInt	Charging current requested by the vehicle (in A)
V0.9.1 .. V2.0	chargingTerminated	UInt	Information about the charging process: • 0: Charging is not (yet) completed. • 1: Charging was completed normally by the EV. • 2: Charging was terminated normally by the charging station (EVCS). • 3: Charging was terminated due to an error by the EV. • 4: Charging was terminated by the EVCS due to an error. Case A e.g.: CSError >0 • 5: Charging was terminated by the EVCS due to an error. Case B Negative edge on ChargeOn before chargingActive = True
V0.9.1 .. V2.0	connProximityDetection	Bool	Indicates whether the charging plug is connected to the vehicle: False: No True: Yes
V1.0 .. V2.0	estimatedChargingTime	USInt	Estimated charging time (in minutes). This value comes from the vehicle. If this value is 255, the EV cannot specify a charging time.
V0.9.1 .. V2.0	EVContactorsClosed	Bool	The battery in the vehicle is connected to the charging cable False: No True: Yes

V0.9.1 .. V2.0	EVCSError	Byte	**Error messages** Info: Only the first error is output • 16#00 No error (OK). • 16#01 Invalid license key Battery fault: • 16#02 Overvoltage • 16#03 Undervoltage • 16#04 Current deviation • 16#05 Overtemperature • 16#06 Voltage deviation EV status • 16#07 System error • 16#08 Shift lever position • 16#09 Logical inconsistency EVCS error • 16#10 Incompatible battery • 16#11 Overvoltage (> threshold voltage) • 16#12 Too high charging current requested (by EV) • 16#13 Available output voltages (min/max values) • 16#14 Available output current > 200 A • 16#15 Available output power > 120 kW • 16#16 Maximum charging time exceeded (> 4.25 h) • 16#17 Timeout in the control flow of the charging process • 16#18 Fault of the connector lock (feedback status, error) • 16#19 Temperature sensor fault • 16#1A P-N Insulation fault • 16#1B Ground fault on EVCS • 16#1C Connector proximity signal fault • 16#1D CAN controller status • 16#1E CAN receive timeout from EV • 16#1F External error (CSError ≠ 0) • 16#20 Logical inconsistency (for example: Measurement voltage on the charging cable before the insulation test) • 16#21 CS P-N Isolation test delay > 15 s

V0.9.1 .. V2.0	extendedFunction1	Byte	Subset of the advanced functions which are possible from the point of view of EVCS and EV. True = Possible for both False = Not possible for EVCS or EV • Bit 0 = Dynamic current control is supported • Bit 1 = High current charging (201A - 400 A) is supported • Bit 2 = High voltage charging (601 V - 1200 V) is supported
V0.9.1 .. V2.0	FBAuthentificated	Bool	Indicates whether the FB has been correctly licensed. True: Yes False: No
V0.9.1 .. V2.0	GNDFaultTestDisable	Bool	Deactivation of the insulation monitoring is required False: No True: Yes If this parameter is set to "True", no insulation monitoring may be active. The value of input parameter "GNDInsulationOK" must be set to "False".
V0.9.1 .. V2.0	identCode	String	Ident code of the CAN module. Is required for licensing the function block.
V0.9.1 .. V2.0	latchHoldingRQ	Bool	Control signal for the charging plug lock. False: Unlocking required True: Locking required.
V0.9.1 .. V2.0	maxBatVoltage	UInt	Maximum battery voltage (in V)
V0.9.1 .. V2.0	maxChargingTime	UInt	Maximum charging time (in s)
V1.0 .. V2.0	minBatVoltage	UInt	Minimum battery voltage (in V)
V1.0 .. V2.0	minChargeCurrent	USInt	Minimum possible charging current from the perspective of the EV (in A)
V0.9.1 .. V2.0	PNInsulationFault	Bool	Insulation fault occurred False: No True: Yes
V0.9.1 .. V2.0	PNInsulationTestReq	Bool	Signaling that the insulation measurement is running As long as the value is set to "True", an insulation measurement must carried out. If the value is "False", no insulation measurement is active, according to the CHAdeMO specification.
V0.9.1 .. V2.0	PNInsulationTestVoltage	UInt	Voltage required by the vehicle to perform the insulation measurement (in V). This must be set at the converter.
V0.9.1 .. V2.0	remChargingTime	UInt	Remaining charging time (in s)
V0.9.1 .. V2.0	ResEVMaxChargingTime-Active	Bool	Status for recalculation of charging time for dynamic charging from EV False: No calculation True: Recalculation is running (approx. 10 s)
V0.9.1 .. V2.0	stateInfo	UDInt	Status information (see also Table stateInfo (Page 42)) This value is updated as long as the charging process is in progress. In particular, it provides additional information in case of premature termination of the charging process due to errors.
V0.9.1 .. V2.0	stateOfCharge	USInt	Charge level of the battery (in % or for V0.9.1 in 0.1 kWh)
V0.9.1 .. V2.0	targetBatVoltage	UInt	Target voltage of the battery (in V)

V1.0 .. V2.0	totCapTractBat	UInt	Charge capacity of the battery (in 0.1 kWh increments; for CHAdeMO V0.9.1 in %)
V0.9.1 .. V2.0	weldingDetection	Byte	Result of the check for welding of the contacts in EV "Welding detection" • 1: OK • Otherwise: not OK

Note

Some parameters of this table may have invalid values in the "CHAdeMOProtNum" parameter with protocol version V0.9.1.

EVCS input debug status information

Values of the EVCS output parameter "stateInfo" for analysis during support requests.

This value is updated as long as the charging process is running (see above "EVCS input charge termination codes"). In particular, it provides additional information in case of premature termination of the charging process due to errors.

Table 5- 4 stateInfo

XX FX XX XX	XX XF XX XX	XX XX FX XX	XX XX XF XX	XX XX XX FF
Error state	**Connection state**	**Connection state**	**P-N Insulation Test State**	**Charging State**
SM_ERR_ NO_ERRO (0)	SM_COM_OFF (0)	SM_CO_OFF_A (0)	SM_IT_PENDING (0)	SM_CS_OFF_A (0)
SM_ERR_ERROR (1)	SM_COM_REC (1)	SM_CO_CONNECTED_B_I (1)	SM_IT_RUNNING (1)	SM_CS_ESTABLISHING_B_C (1)
	SM_COM_EST (2)	SM_CO_LATCHED_C_G_H (2)	SM_IT_OK (2)	SM_CS_INSULATION_TEST_D (2)
		SM_CO_TB_CONNECTED_D_E_F (3)	SM_IT_ERROR (3)	SM_CS_CHARGING_STOP_D (3)
				SM_CS_CHARGING_E (4)

5.5 Required memory when using the TIA library

Memory requirement

The memory requirement of the CHAdeMO function block is as follows:

Table 5- 5 Memory requirement

	Load memory	**Code work memory**	**Data work memory**
First instance	431.18 KiB	23.55 KiB	2.78 KiB
In addition for each additional instance	16.49 KiB	0	2.78 KiB

List of abbreviations

A

Term / Abbreviation	Description
AI	Analog input
AQ	Analog input / analog output
EVCS	Electric Vehicle Charging Station
CS	Charging station
DI	Digital input
DQ	Digital input / digital output
EV	Electric Vehicle
PN	PROFINET
RQ	Relay output

Index

T

U

V

W