

**Programmer's guide**  
last change on: 02.08.2016

# CAN - EDCP Programmers guide

Description of iseg Enhanced Device Control Protocol to access iseg hardware by CAN bus connection

## Document history

| Version | Date      | Major changes            |
|---------|-----------|--------------------------|
| 2.0     | 28.1.2016 | Relayouted documentation |

## Disclaimer / Copyright

Copyright © 2015-2016 iseg Spezialelektronik GmbH / Germany. All Rights Reserved.

It is forbidden to copy, extract parts, duplicate for any kind of publication without a written permission of iseg Spezialelektronik GmbH. This information has been prepared for assisting operation and maintenance personnel to enable efficient use.

## Important security information

Please read all security warnings and notices carefully. iseg declines responsibility for any kind of damage, injuries or other consequences that are caused by improper use negligence, whether or not under conditional intent.

**WARNING** advices in text indicate attention to hazards, that could lead to injuries or death.

**CAUTION** notes in text indicate information to avoid damages to the equipment.

**INFO** notes in text show important information and resources.

### SAFETY ADVICE



WARNING

This device generates high voltages or is part of or attached to high voltage supplying systems. High voltages are dangerous and may be fatal.

USE CAUTION WHILE WORKING WITH THIS EQUIPMENT.  
BE AWARE OF ELECTRICAL HAZARDS.

Always follow at the minimum these provisions:

- High voltages must always be grounded
- Do not touch wiring or connectors without securing
- Never remove covers or equipment
- Always observe humidity conditions
- Service must be done by qualified personnel only

### WARNING



WARNING

#### RAMP DOWN VOLTAGES !

Before insertion or removal of crate controller, please make sure, all voltages are ramped down, crates are switched off and power cord is disconnected.

### ATTENTION



CAUTION

#### CHECK COMPATIBILITY

Please use this device only in compatible MMS crates. Check compatibility list first.

### INFO

The information in this manual is subject to change without notice. We take no responsibility for any error in the document. We reserve the right to make changes in the product design without reservation and without notification to the users.

# Table of Contents

|                                                          |    |
|----------------------------------------------------------|----|
| Document history.....                                    | 1  |
| Disclaimer / Copyright.....                              | 2  |
| Important security information.....                      | 2  |
| 1 Crate Controller CC24/23.....                          | 6  |
| 1.1 Data types.....                                      | 6  |
| 1.2 Access rights.....                                   | 6  |
| 1.3 NMT CAN commands (broadcast messages).....           | 6  |
| 1.4 Crate Controller commands.....                       | 7  |
| 1.4.1 Crate Controller Uptime.....                       | 7  |
| 1.4.2 Crate Controller Serial Number.....                | 7  |
| 1.4.3 Crate Controller Firmware Release.....             | 7  |
| 1.4.4 Crate Controller Firmware Name.....                | 7  |
| 1.4.5 Crate Controller Article Description.....          | 7  |
| 1.4.6 Crate Controller Controller.....                   | 8  |
| 1.4.7 Crate Controller Status.....                       | 8  |
| 1.4.8 Crate Controller Event Status.....                 | 9  |
| 1.4.9 Crate Controller Event Mask.....                   | 9  |
| 1.4.10 Crate Controller Fan Speed Percent.....           | 9  |
| 1.4.11 Crate Power On/Off.....                           | 10 |
| 1.4.12 Crate Chassis Identification.....                 | 10 |
| 1.4.13 Crate Backplane Type.....                         | 10 |
| 1.4.14 Crate Temperature Sensor.....                     | 11 |
| 1.4.15 Crate Supply Measurement.....                     | 11 |
| 1.4.16 Crate Supply Nominal.....                         | 11 |
| 1.5 Statistic registers.....                             | 12 |
| 1.5.1 CAN Bus Received.....                              | 12 |
| 1.5.2 CAN Bus Receiver Overrun.....                      | 12 |
| 1.5.3 CAN Bus Transmitted.....                           | 12 |
| 1.5.4 CAN Bus Transmit Buffer Full.....                  | 13 |
| 1.5.5 CAN Bus Dropped.....                               | 13 |
| 1.5.6 CAN Bus Error.....                                 | 13 |
| 1.5.7 CAN Bus Throttle.....                              | 13 |
| 1.5.8 CAN Bus Status.....                                | 13 |
| 1.5.9 CAN Bus Disabled.....                              | 14 |
| 1.5.10 CAN Bus Bittate.....                              | 14 |
| 1.6 Examples.....                                        | 14 |
| 2 EDCP CAN for High Voltage Devices.....                 | 16 |
| 2.1 General information.....                             | 16 |
| 2.2 General settings and options.....                    | 16 |
| 3 Operation principle.....                               | 16 |
| 3.1 Remote interface control.....                        | 16 |
| 3.1.1 Operation modes.....                               | 17 |
| 3.1.2 Constant Voltage (CV).....                         | 17 |
| 3.1.3 Constant Current (CC).....                         | 17 |
| 3.1.4 Current trip.....                                  | 17 |
| 3.1.5 Function KillEnable.....                           | 17 |
| 3.1.6 Additional current measuring range (Option).....   | 17 |
| 3.2 Control and Status items.....                        | 18 |
| 3.2.1 Controls.....                                      | 18 |
| 3.2.2 Status and events.....                             | 18 |
| 3.2.3 Event status and event mask.....                   | 18 |
| 3.3 Summarizing channel characteristics into groups..... | 19 |
| 3.3.1 Set Group.....                                     | 19 |
| 3.3.2 Status Group.....                                  | 19 |
| 3.3.3 Monitor Group.....                                 | 19 |
| 3.3.4 Timeout Group.....                                 | 19 |
| 3.3.5 Responses on events (Soft-Kill features).....      | 20 |
| 4 Communication via Interface.....                       | 20 |

|                                                                                                          |    |
|----------------------------------------------------------------------------------------------------------|----|
| 4.1 Enhanced Device Control Protocol EDCP.....                                                           | 20 |
| 4.2 CAN-Bus Implementation.....                                                                          | 22 |
| 4.3 Summary of CAN data frame accesses via the NMT service identifier.....                               | 24 |
| 4.4 Summary of CAN data frame accesses via the Front-end-address identifier.....                         | 26 |
| 4.4.1 List to access of the EDCP made for HV boards up to 255 channels EDCP Single Channel Accesses..... | 26 |
| 4.5 Description of data information per DATA_ID in EDCP.....                                             | 34 |
| 4.5.1 EDCP Single Access.....                                                                            | 34 |
| 4.5.1.1 Channel status (single/multiple single read-write access).....                                   | 34 |
| 4.5.1.2 Channel control: (single write- and single/ multiple single read-write access).....              | 35 |
| 4.5.1.3 Channel event status (single write- and single/ multiple single read-write access).....          | 35 |
| 4.5.1.4 Channel event mask (single write- and single/ multiple single read-write access).....            | 36 |
| 4.5.1.5 Delayed trip action (single/ multiple single read-write access).....                             | 37 |
| 4.5.1.6 Delayed trip time.....                                                                           | 37 |
| 4.5.1.7 External channel inhibit.....                                                                    | 37 |
| 4.5.1.8 Set voltage (single write- and single/ multiple single read-write access).....                   | 38 |
| 4.5.1.9 Set current / trip (single write- and single/ multiple single read-write access).....            | 38 |
| 4.5.1.10 Voltage measurement (single/ multiple single read-write access).....                            | 39 |
| 4.5.1.11 Current measurement (single/ multiple single read-write access).....                            | 39 |
| 4.5.1.12 Voltage bounds (single write- / single/ multiple single read-write access).....                 | 39 |
| 4.5.1.13 Current bounds (single write- / single/ multiple single read-write access).....                 | 39 |
| 4.5.1.14 Nominal voltage (single/ multiple single read-write access).....                                | 40 |
| 4.5.1.15 Nominal current (single/ multiple single read-write access).....                                | 40 |
| 4.5.1.16 Current measurement range* (single/ multiple single read-write access).....                     | 40 |
| 4.5.1.17 VCT Coefficient* (single/ multiple single read-write access).....                               | 40 |
| 4.5.1.18 Temperature external* (single/ multiple single read-write access).....                          | 41 |
| 4.5.1.19 Group number (single/ multiple single read-write access).....                                   | 41 |
| 4.5.2 EDCP Module Accesses.....                                                                          | 42 |
| 4.5.2.1 ModuleStatus (module read-write access).....                                                     | 42 |
| 4.5.2.2 ModuleControl (module write- / read-write access).....                                           | 43 |
| 4.5.2.3 ModuleEventStatus (module write- / read-write access).....                                       | 43 |
| 4.5.2.4 ModuleEventMask (module write- / read-write access).....                                         | 44 |
| 4.5.2.5 ModuleEventChannelStatus (module write- / read-write access).....                                | 44 |
| 4.5.2.6 ModuleEventChannelMask (module write- / read-write access).....                                  | 45 |
| 4.5.2.7 ModuleEventGroupStatus (module write- / read-write access).....                                  | 45 |
| 4.5.2.8 ModuleEventGroupMask (module write- / read-write access).....                                    | 45 |
| 4.5.2.9 VoltageRampSpeed (module write- / read-write access).....                                        | 46 |
| 4.5.2.10 CurrentRampSpeed – current controlled modules only (module write- / read-write access).....     | 46 |
| 4.5.2.11 VoltageMax – OPTION (module read-write access).....                                             | 46 |
| 4.5.2.12 CurrentMax – OPTION (module read-write access).....                                             | 47 |
| 4.5.2.13 Supply24 (module read-write access).....                                                        | 47 |
| 4.5.2.14 Supply5 (module read-write access).....                                                         | 47 |
| 4.5.2.15 BoardTemperature (module read-write access).....                                                | 47 |
| 4.5.2.16 Threshold to arm the errors detection (module write / read- write access).....                  | 48 |
| 4.5.2.17 Serial number (module read-write access).....                                                   | 48 |
| 4.5.2.18 Firmware release (module read-write access).....                                                | 48 |
| 4.5.2.19 Bit rate (module write- / read-write access).....                                               | 48 |
| 4.5.2.20 Firmware Name (module read-write access).....                                                   | 49 |
| 4.5.2.21 ADC SamplesPerSecond SPS (module write- / read-write access).....                               | 49 |
| 4.5.2.22 DigitalFilter (module write- / read-write access).....                                          | 50 |
| 4.5.2.23 ModuleOption (module read access).....                                                          | 50 |
| 4.5.2.24 ModuleOptionSpec (module read access).....                                                      | 51 |
| 4.5.2.25 ModuleCommMode (module write access).....                                                       | 51 |
| 4.5.3 EDCP Group Accesses.....                                                                           | 52 |
| 4.5.3.1 Set group.....                                                                                   | 53 |
| 4.5.3.2 Status group.....                                                                                | 54 |
| 4.5.3.3 Monitoring group.....                                                                            | 55 |
| 4.5.3.4 Delayed Trip group.....                                                                          | 56 |
| 4.5.3.5 Set current (– trip) of all channels (group write access).....                                   | 57 |
| 4.5.3.6 Set ON / OFF of all channels (group write- / read-write access).....                             | 57 |
| 4.5.3.7 Set EMERGENCY of all channels (group write- / read-write access).....                            | 57 |
| 4.5.3.8 Event status voltage limit of all channels (group write- / read-write access).....               | 58 |
| 4.5.3.9 Event status current limit of all channels (group write- / read-write access).....               | 58 |
| 4.5.3.10 Event status trip of all channels (group write- / read-write access).....                       | 59 |

|                                                                                                                                       |    |
|---------------------------------------------------------------------------------------------------------------------------------------|----|
| 4.5.3.11 Event status inhibit of all channels (group write- / read-write access).....                                                 | 59 |
| 4.5.3.12 Set ON / OFF channels extender (group write- / read-write access).....                                                       | 60 |
| 4.5.3.13 Set EMERGENCY channels extender (group write- / read-write access).....                                                      | 60 |
| 4.5.4 Important DCP Module Accesses.....                                                                                              | 61 |
| 4.5.4.1 General status (group write- / read-write / active access).....                                                               | 61 |
| 4.5.4.2 Log-on / Log-off Front-end device at superior layer (module active- / write access).....                                      | 62 |
| 4.5.5 Events.....                                                                                                                     | 64 |
| 4.5.5.1 Channel events.....                                                                                                           | 64 |
| 4.5.5.2 Group events (in preparation).....                                                                                            | 64 |
| 4.5.5.3 Module events.....                                                                                                            | 65 |
| Appendix A – Shortcuts.....                                                                                                           | 66 |
| Appendix D – Programming flowchart to store the configurations of the module permanently with help of General state save bit<br>..... | 69 |

# 1 Crate Controller CC24/23

## 1.1 Data types

|       |                                                     |
|-------|-----------------------------------------------------|
| UI1:  | One byte unsigned integer (8 bit)                   |
| CHAR: | One ASCII character (8 bit)                         |
| UI4:  | Four byte unsigned integer (32 bit)                 |
| R4:   | Four byte floating point IEEE-754, single precision |

All data on the CAN bus is in big endian format, i.e. for UI4 and R4, the highest data byte is transmitted first.

## 1.2 Access rights

R = Read only: the register can only be read

R/W = Read/write: the register can be read and written

R/C = Read/clear: the register can be read and cleared

## 1.3 NMT CAN commands (broadcast messages)

The following NMT commands (CAN-ID 0x004) are handled by the Crate Controller:

- NMT\_DATAID\_START (0xC4)
  - Switches from stop mode to normal operation
  - Message is forwarded to the backplane (and therefore to the modules)
- NMT\_DATAID\_STOP (0xC8)
  - Switches from normal operation to stop mode to allow the command NMT\_RESET\_CAN
  - Message is forwarded to the backplane (and therefore to the modules)
- NMT\_DATAID\_RESET\_CAN (0xCC)
  - Re-initializes all CAN interfaces
  - Clears all transmit buffer
  - Clears all statistic registers
  - Message is forwarded to the backplane (and therefore to the modules)
  - Re-initializes the modules CAN interface
  - Modules starts to send their Log-On message

A short delay (10 milliseconds) should be kept between the commands NMT\_STOP and NMT\_RESET\_CAN, so that all messages can be forwarded correctly to the modules.

## 1.4 Crate Controller commands

### 1.4.1 Crate Controller Uptime

Crate Controller Uptime 0x1113 UI4 R

This register returns the crate controllers uptime in seconds.

| Access  | CAN-ID | DLC | DATA_ID | Data_3           | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|------------------|--------|--------|--------|
| Request | 0x601  | 2   | 0x1113  |                  |        |        |        |
| Answer  | 0x604  | 6   | 0x1113  | Crate Uptime UI4 |        |        |        |

### 1.4.2 Crate Controller Serial Number

Crate Controller Serial Number 0x1200 UI4 R

This register returns the crate controllers serial number.

| Access  | CAN-ID | DLC | DATA_ID | Data_3                  | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|-------------------------|--------|--------|--------|
| Request | 0x601  | 2   | 0x1200  |                         |        |        |        |
| Answer  | 0x604  | 6   | 0x1200  | Crate Serial Number UI4 |        |        |        |

### 1.4.3 Crate Controller Firmware Release

Crate Controller Firmware Release 0x1201 UI1[4] R

This register returns the crate controllers firmware release.

| Access  | CAN-ID | DLC | DATA_ID | Data_3    | Data_2    | Data_1    | Data_0    |
|---------|--------|-----|---------|-----------|-----------|-----------|-----------|
| Request | 0x601  | 2   | 0x1201  |           |           |           |           |
| Answer  | 0x604  | 6   | 0x1201  | Release 1 | Release 2 | Release 3 | Release 4 |

### 1.4.4 Crate Controller Firmware Name

Crate Controller Firmware Name 0x1203 CHAR[6] R

This register returns the crate controllers firmware name.

| Access  | CAN-ID | DLC | DATA_ID | Data_5 | Data_4 | Data_3 | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|--------|--------|--------|--------|--------|--------|
| Request | 0x601  | 2   | 0x1203  |        |        |        |        |        |        |
| Answer  | 0x604  | 8   | 0x1203  | 'E'    | 'C'    | 'H'    | '4'    | 'x'    | 'A'    |

### 1.4.5 Crate Controller Article Description

Crate Controller Article Description 0x1209 CHAR[] R

This register returns the crate controllers article description. Depending on the length of the article description, multiple CAN messages may be sent. The description is terminated by a zero character.

| Access  | CAN-ID | DLC   | DATA_ID | Data_5 | Data_4 | Data_3 | Data_2 | Data_1 | Data_0 |
|---------|--------|-------|---------|--------|--------|--------|--------|--------|--------|
| Request | 0x601  | 2     | 0x1209  |        |        |        |        |        |        |
| Answer  | 0x604  | 3...8 | 0x1209  | 0      | 'C'    | 'C'    | '2'    | '4'    | 0      |

## 1.4.6 Crate Controller Controller

Crate Controller Control

0x1A01

UI4

R/W

The register Crate Controller Control sets crate functions.

| Access  | CAN-ID | DLC | DATA_ID | Data_3            | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|-------------------|--------|--------|--------|
| Request | 0x601  | 2   | 0x1A01  |                   |        |        |        |
| Answer  | 0x604  | 6   | 0x1A01  | Crate Control UI4 |        |        |        |
|         |        |     |         |                   |        |        |        |
| Set     | 0x600  | 6   | 0x1A01  | Crate Control UI4 |        |        |        |

Bit 31

|         |         |         |         |         |         |                            |                         |
|---------|---------|---------|---------|---------|---------|----------------------------|-------------------------|
| res (0) | res (0) | res (0) | res (0) | res (0) | res (0) | do Set Crate Enable Active | set Crate Enable Active |
| res (0) | res (0) | res (0) | res (0) | res (0) | res (0) | res (0)                    | res (0)                 |
| res (0) | res (0) | res (0) | res (0) | res (0) | res (0) | res (0)                    | res (0)                 |
| res (0) | res (0) | res (0) | res (0) | res (0) | res (0) | do Clear Statistic         | do Clear Events         |

Bit 0

do Clear Events: Clears the crates event status register. This bit can only be written, it always reads zero.

doClear Statistic: Clears the crates statistic registers. This bit can only be written, it always reads zero.

do Set Crate Enable Active: This bit masks the bit set Crate Enable Active. If this bit is zero, the bit set Crate Enable Active is ignored. This bit can only be written, it always reads zero.

set Crate Enable Active: This bit can only be changed if the backplane is powered off and the bit do Set Crate Enable Active is set to one.

## 1.4.7 Crate Controller Status

Crate Controller Status

0x1A00

UI4

R

The Crate Controller Status contains the *actual* status. The bits will be set or reset depending on the crates status.

| Access  | CAN-ID | DLC | DATA_ID | Data_3           | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|------------------|--------|--------|--------|
| Request | 0x601  | 2   | 0x1A00  |                  |        |        |        |
| Answer  | 0x604  | 6   | 0x1A00  | Crate Status UI4 |        |        |        |

Bit 31

|                  |         |                    |                   |                   |                  |                  |                 |
|------------------|---------|--------------------|-------------------|-------------------|------------------|------------------|-----------------|
| res (0)          | res (0) | res (0)            | res (0)           | res (0)           | res (0)          | res (0)          | res (0)         |
| res (0)          | res (0) | Crate Fast Off     | Crate Enabled     | Shut Down         | High Voltage On  | Power Fail       | Power On        |
| res (0)          | res (0) | res (0)            | Sum Error         | High +3.3 CC      | Low +3.3 CC      | High +5 CC       | Low +5 CC       |
| High Temperature | Service | High +24 Backplane | Low +24 Backplane | High +5 Backplane | Low +5 Backplane | High +24 Battery | Low +24 Battery |

Bit 0

High-Voltage-On: Backplane is powered on and at least one channel within the crate has Status.isOn or measured voltage > 63 V The front panel LED HV-ON is derived from this bit.

Power-On: Backplane is powered on (modules are supplied with voltage)The front panel LED Status lights green when Power-On is set and no supply error exists.

Power-Fail: The front panel LED Status lights red when Power-On is set and a supply error exists.

AC line power fail detected.

Crate without UPS: high voltage is turned fast off

Crate with UPS: high voltage is turned off with ramp after the wait time

High Temperature: At least one modules or the crate controller have high temperature. The high voltage is turned off with the configured voltage ramp speed.

If the front button POWER ON is pressed for more than 10 seconds, this bit is set for approx. 1 minute. This is used to perform a shut down of the embedded crate computer.

This bit is set, whenever one of the bits Power Fail, High Temperature, Service, High Supply, Low Supply or Crate Fast Off is set or if Crate Enabled is cleared

A fatal error occurred. Contact service.

Measured voltage X exceeds the upper limit.

Measured voltage X exceeds the lower limit.

If the crate is enabled, it is possible to turn on high voltage for all channels. The crate is enabled, if the Crate Control bit set Crate Enable Active is not set, or if the CONTROL pin Crate Enable is pulled high.

If the CONTROL pin Crate Fast Off is turned high, the high voltages are shut down without ramp.

### 1.4.8 Crate Controller Event Status

|                               |        |     |     |
|-------------------------------|--------|-----|-----|
| Crate Controller Event Status | 0x1A02 | UI4 | R/C |
|-------------------------------|--------|-----|-----|

The Event Status bits are set together with the status bits. Unlike Status bits, Event Status bits are not reset automatically. The have to be reset by the user, by writing an 1 to this event bit. All Event Status bits are reset by the Crate Control bit do Clear.

|         |        |     |         |                        |        |        |        |
|---------|--------|-----|---------|------------------------|--------|--------|--------|
| Access  | CAN-ID | DLC | DATA_ID | Data_3                 | Data_2 | Data_1 | Data_0 |
| Request | 0x601  | 2   | 0x1A02  |                        |        |        |        |
| Answer  | 0x604  | 6   | 0x1A02  | Crate Event Status UI4 |        |        |        |
|         |        |     |         |                        |        |        |        |
| Clear   | 0x600  | 6   | 0x1A02  | Crate Event Status UI4 |        |        |        |

Bit 31

|                  |         |                    |                   |                   |                  |                  |                 |
|------------------|---------|--------------------|-------------------|-------------------|------------------|------------------|-----------------|
| res (0)          | res (0) | res (0)            | res (0)           | res (0)           | res (0)          | res (0)          | res (0)         |
| res (0)          | res (0) | res (0)            | res (0)           | Shut Down         | High Voltage On  | Power Fail       | Power On        |
| res (0)          | res (0) | res (0)            | Sum Error         | High +3.3 CC      | Low +3.3 CC      | High +5 CC       | Low +5 CC       |
| High Temperature | Service | High +24 Backplane | Low +24 Backplane | High +5 Backplane | Low +5 Backplane | High +24 Battery | Low +24 Battery |

Bit 0

### 1.4.9 Crate Controller Event Mask

|                             |       |     |     |
|-----------------------------|-------|-----|-----|
| Crate Controller Event Mask | 0x1A3 | UI4 | R/W |
|-----------------------------|-------|-----|-----|

The Event Mask is defined for compatibility to the module EDCP command set, but not used at the moment.

| Access  | CAN-ID | DLC | DATA_ID | Data_3               | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|----------------------|--------|--------|--------|
| Request | 0x601  | 2   | 0x1A03  |                      |        |        |        |
| Answer  | 0x604  | 6   | 0x1A03  | Crate Event Mask UI4 |        |        |        |
|         |        |     |         |                      |        |        |        |
| Set     | 0x600  | 6   | 0x1A03  | Crate Event Mask UI4 |        |        |        |

#### 1.4.10 Crate Controller Fan Speed Percent

|                                    |        |    |   |
|------------------------------------|--------|----|---|
| Crate Controller Fan Speed Percent | 0x1A04 | R4 | R |
|------------------------------------|--------|----|---|

This register returns the crates fan speed in percent (0...100). The fan speed is regulated according to the maximum crate temperature. The maximum temperature is collected over all modules and the crate controller. The maximum fan speed is reached at approx. 45° C.

| Access | CAN-ID | DLC | DATA_ID | Data_3 | Data_2 | Data_1 | Data_0 |
|--------|--------|-----|---------|--------|--------|--------|--------|
|--------|--------|-----|---------|--------|--------|--------|--------|



Bit meaning:

0 = Backplane type iseg: backplane is sequential

1 = Backplane type wiener: backplane is not sequential

Bit position:

0 = Master crate

1...5 = Slave crates on CAN line yellow (1 = Slave 0, 2 = Slave 1, ...)

8...12 = Slave crates on CAN line green (8 = Slave 0, 9 = Slave 1, ...)

## 1.4.14 Crate Temperature Sensor

Crate Temperature Sensor

0x2001

R4

R

This register contains the actual temperature for different sensors.

Sensor 0 and 1 are placed at the crate controller. Sensor 2 is the maximum temperature in the crate (collected over all modules and the crate controller). The maximum temperature is used to control the crates fan speeds.

If the request is done with DLC = 2, all temperature sensors are returned in three CAN messages. If the request is done with DLC = 3 and a specific sensor, only this sensor is returned.

| Access  | CAN-ID | DLC | DATA_ID | Sensor | Data_3                 | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|--------|------------------------|--------|--------|--------|
| Request | 0x601  | 2   | 0x2001  |        |                        |        |        |        |
| Answer  | 0x604  | 7   | 0x2001  | 0x00   | Crate Temperature 0 R4 |        |        |        |
|         | 0x604  | 7   | 0x2001  | 0x01   | Crate Temperature 1 R4 |        |        |        |
|         | 0x604  | 7   | 0x2001  | 0x02   | Crate Temperature 2 R4 |        |        |        |
|         |        |     |         |        |                        |        |        |        |
| Request | 0x601  | 3   | 0x2001  | 0x01   |                        |        |        |        |
| Answer  | 0x604  | 7   | 0x2001  | 0x01   | Crate Temperature 0 R4 |        |        |        |

## 1.4.15 Crate Supply Measurement

Crate Supply Measurement

0x2002

R4

R

This register contains the measured supply voltages.

If the request is done with DLC = 2, all supply measurement values are returned in consecutive CAN messages. If the request is done with DLC = 3 and a specific supply number, only this measurement value is returned.

| Access  | CAN-ID | DLC | DATA_ID | Supply | Data_3                        | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|--------|-------------------------------|--------|--------|--------|
| Request | 0x601  | 2   | 0x2002  |        |                               |        |        |        |
| Answer  | 0x604  | 7   | 0x2002  | 0x00   | Crate Supply Measurement 0 R4 |        |        |        |
|         | 0x604  | 7   | 0x2002  | 0x01   | ...                           |        |        |        |
|         | 0x604  | 7   | 0x2002  | 0x08   | Crate Supply Measurement 8 R4 |        |        |        |

## 1.4.16 Crate Supply Nominal

Crate Supply Nominal

0x2003

R4

R

This register contains the nominal supply voltages.

If the request is done with DLC = 2, all supply nominal values are returned in consecutive CAN messages. If the request is done with DLC = 3 and a specific supply number, only this nominal value is returned.

| Access  | CAN-ID | DLC | DATA_ID | Supply | Data_3 | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|--------|--------|--------|--------|--------|
| Request | 0x601  | 2   | 0x2003  |        |        |        |        |        |



## 1.5.4 CAN Bus Transmit Buffer Full

CAN Bus Transmit Buffer Full

0x2043

UI4

R

This register counts the messages, that could not be sent to the given CAN bus because of full transmit buffer.

| Access  | CAN-ID | DLC | DATA_ID | CAN bus | Data_3                   | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|---------|--------------------------|--------|--------|--------|
| Request | 0x601  | 3   | 0x2043  | 00      |                          |        |        |        |
| Answer  | 0x604  | 7   | 0x2043  | 00      | Transmit Buffer Full UI4 |        |        |        |

## 1.5.5 CAN Bus Dropped

CAN Bus Dropped

0x2044

UI4

R

This register counts the messages that were received from the given CAN bus and could not be routed because of unclear destination.

This register should always be zero.

| Access  | CAN-ID | DLC | DATA_ID | CAN bus | Data_3               | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|---------|----------------------|--------|--------|--------|
| Request | 0x601  | 3   | 0x2044  | 00      |                      |        |        |        |
| Answer  | 0x604  | 7   | 0x2044  | 00      | Dropped Messages UI4 |        |        |        |

## 1.5.6 CAN Bus Error

CAN Bus Error

0x2045

UI4

R

This register is incremented every second when the given CAN bus is in error state.

| Access  | CAN-ID | DLC | DATA_ID | CAN bus | Data_3            | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|---------|-------------------|--------|--------|--------|
| Request | 0x601  | 3   | 0x2045  | 00      |                   |        |        |        |
| Answer  | 0x604  | 7   | 0x2045  | 00      | Error Seconds UI4 |        |        |        |

## 1.5.7 CAN Bus Throttle

CAN Bus Throttle

0x2046

UI4

R

This register counts the number of generated throttle messages for the given CAN bus.

| Access  | CAN-ID | DLC | DATA_ID | CAN bus | Data_3                | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|---------|-----------------------|--------|--------|--------|
| Request | 0x601  | 3   | 0x2046  | 00      |                       |        |        |        |
| Answer  | 0x604  | 7   | 0x2046  | 00      | Throttle Messages UI4 |        |        |        |

## 1.5.8 CAN Bus Status

CAN Bus Status

0x2047

UI4

R

This register holds the current status for the given CAN bus.

| Access  | CAN-ID | DLC | DATA_ID | CAN bus | Data_3         | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|---------|----------------|--------|--------|--------|
| Request | 0x601  | 3   | 0x2047  | 00      | 1              | 2      | 3      | 4      |
| Answer  | 0x604  | 7   | 0x2047  | 00      | Bus Status UI4 |        |        |        |

The CAN Bus Status register contains the following information:

- Receiver bus status: Ok (0), Warning (1), Error (2), Bus off (3)
- Transmitter bus status: Ok (0), Warning (1), Error (2), Bus off (3)
- CAN hardware activated

- CAN hardware synchronized with CAN bus
- CAN hardware in special mode (initializing, sleep, listen-only, loopback).  
These bits should not be set in normal operation conditions.

The CAN hardware status is refreshed at the time of the request.

Bit 31

|                                                                      |  |          |             |         |              |          |              |
|----------------------------------------------------------------------|--|----------|-------------|---------|--------------|----------|--------------|
|                                                                      |  |          |             |         |              |          |              |
|                                                                      |  | Loopback | Listen Only | Enabled | Synchronized | Sleeping | Initializing |
| Transmitter bus status (0 = Ok, 1 = Warning, 2 = Error, 3 = Bus off) |  |          |             |         |              |          |              |
| Receiver bus status (0 = Ok, 1 = Warning, 2 = Error, 3 = Bus off)    |  |          |             |         |              |          |              |

Bit 0

## 1.5.9 CAN Bus Disabled

CAN Bus Disabled

0x2048

UI4

R

This register counts the number of dropped messages because the given CAN bus is not enabled (e.g. backplane is off).

| Access  | CAN-ID | DLC | DATA_ID | CAN bus | Data_3                | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|---------|-----------------------|--------|--------|--------|
| Request | 0x601  | 3   | 0x2048  | 00      |                       |        |        |        |
| Answer  | 0x604  | 7   | 0x2048  | 00      | Disabled Messages UI4 |        |        |        |

## 1.5.10 CAN Bus Btrate

CAN Bus Btrate

0x2049

UI4

R

This register holds the current bit rate for the given CAN bus.

| Access  | CAN-ID | DLC | DATA_ID | CAN bus | Data_3     | Data_2 | Data_1 | Data_0 |
|---------|--------|-----|---------|---------|------------|--------|--------|--------|
| Request | 0x601  | 3   | 0x2049  | 00      |            |        |        |        |
| Answer  | 0x604  | 7   | 0x2049  | 00      | Btrate UI4 |        |        |        |

## 1.6 Examples

CC = Crate-Controller, PC = Controlling PC

| Message             | Direction | CAN-ID | DLC | Data ID | Data | Data | Data | Data |
|---------------------|-----------|--------|-----|---------|------|------|------|------|
| Log-On Request      | CC → PC   | 601    | 3   | D8 00   | 2E   |      |      |      |
| Log-On Confirmation | CC → PC   | 600    | 2   | D8 01   |      |      |      |      |
| Status Request      | PC → CC   | 601    | 2   | 1A 00   |      |      |      |      |
| Status Answer       | CC → PC   | 604    | 6   | 1A 00   | 00   | 00   | 00   | 00   |
| Crate Power On      | PC → CC   | 600    | 3   | 1A 05   | 01   |      |      |      |
| Crate Power Off     | PC → CC   | 600    | 3   | 1A 05   | 00   |      |      |      |
| Fan Speed Request   | PC → CC   | 601    | 2   | 1A 04   |      |      |      |      |
| Fan Speed Answer    | CC → PC   | 604    | 6   | 1A 04   | 40   | A0   | 00   | 00   |

Request Temperatures and Supplies (group request)

| Message                  | CAN-ID | DLC | DataID | Channel | Data | Data | Data | Data | Value   |
|--------------------------|--------|-----|--------|---------|------|------|------|------|---------|
| Request all Temperatures | 601    | 2   | 20 01  |         |      |      |      |      |         |
| Temperature 0            | 604    | 7   | 20 01  | 00      | 41   | EF   | 0D   | B0   | 29.9 °C |
| Temperature 1            | 604    | 7   | 20 01  | 01      | 41   | ED   | 66   | 30   | 29.7 °C |

|                               |     |   |       |    |    |    |    |    |         |
|-------------------------------|-----|---|-------|----|----|----|----|----|---------|
| Temperature 2                 | 604 | 7 | 20 01 | 02 | 41 | EF | 0D | B0 | 29,9 °C |
| Request all Measured Supplies | 601 | 2 | 20 02 |    |    |    |    |    |         |
| Backplane +24                 | 604 | 7 | 20 02 | 00 | 41 | BE | 75 | 98 | 23,8 V  |
| Reserved                      | 604 | 7 | 20 02 | 01 | 00 | 00 | 00 | 00 | 0 (res) |
| Backplane +5                  | 604 | 7 | 20 02 | 02 | 40 | A0 | 51 | EC | 5,01 V  |
| Reserved                      | 604 | 7 | 20 02 | 03 | 00 | 00 | 00 | 00 | 0 (res) |
| Reserved                      | 604 | 7 | 20 02 | 04 | 00 | 00 | 00 | 00 | 0 (res) |
| Supply +5                     | 604 | 7 | 20 02 | 05 | 40 | 9F | A2 | 1B | 4,99 V  |
| Supply +3.3                   | 604 | 7 | 20 02 | 06 | 40 | 53 | 2E | 1C | 2,99 V  |
| Reserved                      | 604 | 7 | 20 02 | 07 | 00 | 00 | 00 | 00 | 0 (res) |
| Battery + 24                  | 604 | 7 | 20 02 | 08 | 41 | D0 | CC | CD | 26,1 V  |

## 2 EDCP CAN for High Voltage Devices

### 2.1 General information

Each single HV channel is independently controllable. The units are software controlled via CAN-Interface through a PC or similar controller.

Using an iseg crate combined with a CC24 controller or an iCSmini as stand alone iseg communication server WEB-services, Remote-service, EPICS and SNMP are possible via Ethernet.

We offer a comfortable control program "isegCANHVControl" for up to 64 iseg modules with CAN interface under Windows and „iseg OPC Control" in connection with the „iseg OPC Server".

Using the w-i-e-n-e-r Mpod crate it is possible to control up to 10 modules via Ethernet-SNMP Interface.

### 2.2 General settings and options

Please note that there are additional hardware features for these devices in this manual called **OPTION**. The use of an access without the hardware implementation will be described under **OPTION** in manual.

Devices with different bit rate settings do not work on the same CAN bus.

The actual channel and module values (measurement and status) are refreshed approximately every 10ms x number of channels.

The board temperature values are refreshed approximately every 5 up to 10 s.

## 3 Operation principle

### 3.1 Remote interface control

The Multi-Channel HV modules are controlled via a remote interface. The communication between an application and the module is performed by the transmission of data items. A data item contains information to be submitted to and/or received from the module. It can represent a specific quantity or a union of single bits. The majority of the data items are standard for all Multi-Channel HV modules and are described in the interface manual in detail. Data items for optional functions are described in the interface options manual.

A general distinction can be made between data items to control individual HV channels and data items to control the HV modules with the sum of all contained channels.

The former group includes the following data items, which exist for every single HV channel:

- items to handle channel status, control and event's
- items to set the voltage or current, bounds, interlock maximum and minimum
- items to read the measured voltage and current
- items to read the nominal voltage and current

The following data items control the properties of the whole HV module. These items exist only once per module:

- items to handle module status, control and events
- voltage ramp speed (is the same for all HV channels)
- current ramp speed (is the same for all HV channels)
- restart time after recalling set values
- maximum set voltage
- maximum set current
- ADC samples per second
- digital filter setting
- power supply voltages
- temperature
- maximum voltage
- maximum current

### 3.1.1 Operation modes

There are three operation modes depending on the HV hardware and the module configuration.

### 3.1.2 Constant Voltage (CV)

In the mode Constant Voltage the module works as a constant voltage source. For this mode it is required that the value for current set ( $I_{set}$ ) or current trip ( $I_{trip}$ ) is greater than the resulting output current.

### 3.1.3 Constant Current (CC)

In the mode Constant Current the module works as a constant current source. For this mode it is required that the HV channel has implemented a current control and that the current set value  $I_{set}$  is smaller than the current that would result from set voltage and the load at the HV output.

### 3.1.4 Current trip

This is a special case of the Constant Voltage mode. The module usually provides a constant output voltage, where the value of the parameter  $I_{trip}$  defines a current limit. If this value is reached or exceeded (e.g. by arcs), in this mode the channel will be switched off immediately.

### 3.1.5 Function KillEnable

KillEnable is a global control signal that defines the behaviour of the module if a given voltage ( $V_{max}$ ) or current limit ( $I_{max}/I_{set}/I_{trip}$ ) is exceeded.

If KillEnable is active the violation of one of the limits will trigger a Kill-signal in the respective channel. This signal will switch off the channel immediately without ramp.

If KillEnable is inactive and one of the limits  $I_{max}/I_{set}$  or  $I_{trip}$  is exceeded the following will happen:

HV hardware with current control - switch the channel from voltage control into current control.

HV hardware without current control - a trip in the channel hardware will switch off the high voltage generation. Then the module automatically starts to restore the HV via a voltage ramp to the set voltage. If the HV is held during the trip, e.g. by an external capacity load, the recovery of the HV starts from the voltage at the output. The auto-recovery of the voltage is performed only once in a time span of 10 minutes. If the channel trips a second time within the 10 minutes the HV will be switched off.

### 3.1.6 Additional current measuring range (Option)

Some modules are equipped with a second current measuring range to capture small current values. The range is automatically detected. In the second range the values will be converted with a higher resolution. The value is in the same floating point format as in the first range. The device control protocol allows to request which range is active.

**INFO** The second range cannot be activated if:

- function KillEnable is on.
- a voltage ramp up is running.
- the module operates in CC mode.

## 3.2 Control and Status items

### 3.2.1 Controls

Control items encapsulate a number of bits which allow to switch On or Off specific functions. There is a control item for the module (ModuleControl) and one for each channel (ChannelControl). Control bits that are used to switch a function permanently are named "set..." (e.g. "setON" to switch a channel On or Off). Bits that initiate the execution of a task just once are named "do..." (e.g. "doClear" to clear all events).

### 3.2.2 Status and events

Status items contain a register that encapsulates bits that indicate the current status of the module or channel. Status bits are named starting with "is...". The status always displays only present conditions, if a condition has changed corresponding status bits will be updated.

Unlike the status, event items record previous conditions (e.g. exceeded limits, trips etc.). If an event is registered the corresponding event bit is set permanently to "1" and will keep the information until explicitly reset. Event bits are named starting with "E...".

|        |                                                                                       |
|--------|---------------------------------------------------------------------------------------|
| status | Summary of actual condition of module, channel or group                               |
| event  | Event, characterizes a former or actual special condition of module, channel or group |

### 3.2.3 Event status and event mask

To avoid the need for checking all event sources permanently for incoming events, the module provides a hierarchical chain for the combination of the events to a single status bit. The structure for the event processing allows a combination of events coming from the module status, the status of the channels and the group status. For each event status item a corresponding event mask item is provided. The event mask defines which event status bits contribute to the combined event status.

Event status Events that have been registered so far

Event mask Filter to define which individual events contribute to the summarized event

Between event status items and the corresponding mask is a bit by bit correspondence. The bits in the mask are named starting with "ME...". If the mask bit is set, the occurring of the respective event will activate the combined event. In turn these sum events are collected in an event status register and connected with an event mask register at this higher level.



If an event bit in the EventStatus is active and the corresponding bit in the EventMask is set, it is not possible to ramp up the voltage or to activate the HV generation if it has been switched off. To unblock this the EventStatus bits must be reset by writing "1" on the corresponding bit positions.

Individual events in the channel event status are starting point of the event combination logic.

First each event status bit for the channel is combined with the corresponding bit in the event mask using a logical AND. Then an event status bit for the channel is generated by combining all resulting bits with a logical OR. The full logical operation is given by

```

EventChannelStatus[n] = (Channel[n].EventVoltageLimit AND Channel[n].MaskEventVoltageLimit) OR
(Channel[n].EventCurrentLimit AND Channel[n].MaskEventCurrentLimit) OR
(Channel[n].EventCurrentTrip AND Channel[n].MaskEventCurrentTrip) OR
(Channel[n].EventExtInhibit AND Channel[n].MaskEventExtInhibit) OR
(Channel[n].EventVoltageBounds AND Channel[n].MaskEventVoltageBounds) OR
(Channel[n].EventCurrentBounds AND Channel[n].MaskEventCurrentBounds) OR
(Channel[n].EventControlledVoltage AND Channel[n].MaskEventControlledVoltage) OR
(Channel[n].EventControlledCurrent AND Channel[n].MaskEventControlledCurrent) OR
(Channel[n].EventEmergency AND Channel[n].MaskEventEmergency) OR
(Channel[n].EventEndOfRamp AND Channel[n].MaskEventEndOfRamp) OR
(Channel[n].EventOnToOff AND Channel[n].MaskEventOnToOff) OR
(Channel[n].EventInputError AND Channel[n].MaskEventInputError)

```

The result of the first step for all channels is stored in the register EventChannelStatus.

In the next step all bits of the EventChannelStatus are combined to a single status bit, using the corresponding mask (EventChannelMask). The logical operation is given by

```

EventChannelActive =      (EventChannelStatus[0] AND EventChannelMask[0]) OR
                          (EventChannelStatus[1] AND EventChannelMask[1]) OR
                          ...
                          (EventChannelStatus[n] AND EventChannelMask[n])

```

A second branch in the event processing logic treats events generated by the status of the module. The following scheme applies to these module events:

```

EventModuleActive =      (EventTemperatureNotGood AND MaskEventTemperatureNotGood) OR
                          (EventSupplyNotGood AND MaskEventSupplyNotGood) OR
                          (EventSafetyLoopNotGood AND MaskEventSafetyLoopNotGood)

```

A third branch combines events generated by groups (monitor group, timeout group, see chapter 3)

Group events are stored in the status register EventGroupStatus. The mask EventGroupMask is used to generate the combined bit EventGroupActive with the following operation:

```

EventGroupActive =      (EventGroupStatus[0] AND EventGroupMask[0]) OR
                          (EventGroupStatus[1] AND EventGroupMask[1]) OR
                          ...
                          (EventGroupStatus[32] AND EventGroupMask[32])

```

Finally the three branches are combined to the bit IsEventActive in the register ModuleStatus:

```

IsEventActive =          EventChannelActive OR EventModuleActive OR EventGroupActive

```

### 3.3 Summarizing channel characteristics into groups

The module provides a highly flexible group functionality. A group is a combination of all or a selection of channels with the ability to control or monitor a specified quantity or characteristic of all included channels. There are two classes of groups "Fix Groups" and "Variable Groups". The former are predefined groups that allow to set single specification values in all channels. The latter are configurable groups that allow to customize the logical structure of the module to the logical structure of the application. They allow an arbitrary assignment of channels and provide a wide range of functionality, structured in four predefined group types. Up to 32 Variable Groups can be defined. The predefined group types are:

#### 3.3.1 Set Group

sets a specified channel characteristic in all selected channels  
no event generation

#### 3.3.2 Status Group

represents the status (condition) of a channel characteristic for all channels  
no event generation

#### 3.3.3 Monitor Group

monitors the condition of a channel characteristic for selected channels  
event generation when the condition changes  
configurable response (e.g. switch off)

#### 3.3.4 Timeout Group

monitors the current trip in selected channels  
to employ this group the signal KillEnable must be turned off  
Event generation only after expiry of a predefined time within which the trip condition must be active  
configurable response (e.g. switch off)

### 3.3.5 Responses on events (Soft-Kill features)

Event generating groups can be configured to perform one out of four predefined responses if the event has been generated:

- shut down of the whole module without ramp
  - high voltage in all channels of the module is switched off
- switch off all channels that are members of the group without ramp
  - high voltage in all channels of the group is switched off
- switch off all channels that are members of the group with ramp
  - high voltage in all channels of the group is ramped down
- no response
  - no change

## 4 Communication via Interface

All modules are controlled via a serial CAN bus interface according to CAN bus specification 2.0A. The actual control protocol is the "Enhanced Device Control Protocol" and is explained more precisely in the following sections.

Furthermore it is implemented a second command set, which corresponds to the older standard protocol "Device Control Protocol". The description of the Device Control Protocol is carried out in the corresponding manual.

### 4.1 Enhanced Device Control Protocol EDCP

The communication between the controller and the module is working according to the Enhanced Device Control Protocol EDCP, which has been designed for instruments of Multi-Channel systems by iseg Spezialelektronik GmbH. This protocol is working according to the master slave principle. Therefore, the control of the HV device through a controller in the superior layer is the master in this system, while the module (as a Front-end device with intelligence) is the slave.

The data exchange between the controller and the HV device is working with help of data frames. These data frames are made out of one direction bit DATA\_DIR, one 16bit DATA\_ID and further data bytes. The direction bit DATA\_DIR defines whether the data frame is a write or read-write access. Write access means that the host writes data into the module, read-write access means that the host wants to read data from the module (this is the read access), and the module answers by a write access.

The DATA\_ID is characterized through the first bit of the data frame with DATA\_ID.b15=0 of EDCP frames (DATA\_ID.bit7=1 of standard DCP frames). In order to code the type of an access the bit14=1 for a single channel access (symbol S), b13=1 for a group access (symbol G) and b12=1 for a module access (symbol M).

The next tables will give an overview of the parts of the EDCP:

| Access                               | DATA DIR | DATA_ID bits |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | CHN bits |                |   |   |   |    |   |   |
|--------------------------------------|----------|--------------|----|----|----|-----|-----|----|----|----|----|----|----|----|----|----|----|----------|----------------|---|---|---|----|---|---|
|                                      |          | 15           | 14 | 13 | 12 | 11  | 10  | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  | 7        | 6              | 5 | 4 | 3 | 2  | 1 | 0 |
| Enhanced DATA_ID                     | 1/0      | 0            | S  | G  | M  | x   | x   | x  | x  | x  | x  | x  | x  | x  | x  | x  | x  |          |                |   |   |   |    |   |   |
| Single channel CHN Write access      | 0        | 0            | 1  | 0  | 0  | S11 | S10 | S9 | S8 | S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 | C7       | CHN (0 to 255) |   |   |   | C0 |   |   |
| Single channel CHN Read-write access | 1/0      | 0            | 1  | 0  | 0  | S11 | S10 | S9 | S8 | S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 | C7       | CHN (0 to 255) |   |   |   | C0 |   |   |
| Module Write access                  | 0        | 0            | 0  | 0  | 1  | M11 | M10 | M9 | M8 | M7 | M6 | M5 | M4 | M3 | M2 | M1 | M0 |          |                |   |   |   |    |   |   |
| Module Read-write access             | 1/0      | 0            | 0  | 0  | 1  | M11 | G10 | M9 | M8 | M7 | M6 | M5 | M4 | M3 | M2 | M1 | M0 |          |                |   |   |   |    |   |   |

If the type of the data frame is a single channel access it will code the corresponding channel information with help of the next multiplex of channel byte (symbol CHN). If the type of the data frame is a module access then a DATA\_ID is necessary only.

| Access                            | DATA DIR | DATA_ID bits |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | CHN / MBR bits |     |   |   |   |    |        |   |
|-----------------------------------|----------|--------------|----|----|----|-----|-----|----|----|----|----|----|----|----|----|----|----|----------------|-----|---|---|---|----|--------|---|
|                                   |          | 15           | 14 | 13 | 12 | 11  | 10  | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  | 7              | 6   | 5 | 4 | 3 | 2  | 1      | 0 |
| Enhanced DATA_ID                  | 1/0      | 0            | S  | G  | M  | x   | x   | x  | x  | x  | x  | x  | x  | x  | x  | x  | x  |                |     |   |   |   |    |        |   |
| Single channel CHN of members MBR | 1<br>0   | 0            | 1  | 1  | 0  | S11 | S10 | S9 | S8 | S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 | M15            | MBR |   |   |   | M0 | OFFSET |   |
| Read-write access                 |          |              |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    |                | C7  |   |   |   |    | CHN    |   |

If the type of the data frame is a single channel and group access then it will code the corresponding channel members with help of the next 16bit word (symbol MBR, channel15=bit15, ... , channel0=bit0) followed by an OFFSET byte to have a channel start index in steps of 16. If a HV device has received such a request message it will answer with multiple CAN frames for all channels which are addressed as members (MBR).

**NOTE** MBR=0 will address all channels of the module!

| Access                                 | DATA DIR | DATA_ID bits |    |    |    |     |     |     |    |     | NBR / CHN bits |     |    |           |     |     |    |      |
|----------------------------------------|----------|--------------|----|----|----|-----|-----|-----|----|-----|----------------|-----|----|-----------|-----|-----|----|------|
|                                        |          | 15           | 14 | 13 | 12 | 11  | 10  | ... | 1  | 0   | 7              | ... | 0  |           |     |     |    |      |
| Group of members MBR Write access      | 0        | 0            | S  | G  | M  | x   | x   | ... | x  | x   | N7             | NBR | N0 | OFFSET    | M15 | MBR | M0 | Type |
| Group of members MBR Read-write access | 1        | 0            | 0  | 1  | 0  | G11 | G10 | ... | G1 | G0  | C7             | CHN | C0 | 0, 16, 32 | M15 | MBR | M0 |      |
|                                        | 0        |              |    |    |    |     |     |     | C7 | CHN | C0             |     |    |           |     |     |    |      |

If the type of the data frame is a group access than it will coded the corresponding group number symbol NBR, the channel members symbol MBR and the channel start index symbol OFFSET.

These data frames correspond to a transfer into layer 3 (Network Layer) and layer 4 (Transport Layer) of the OSI model of ISO. The transmission medium is the CAN Bus according to specification 2.0A, related to level1 (Physical Layer) and level 2 (Data Link Layer).

The Enhanced Device Control Protocol EDCP has been matched to the CAN Bus according to specification CAN 2.0A. Therefore specials of layer 1 and 2 are mentioned only if absolutely necessary and if misunderstandings of functions between the Transport Layer and functions of the Data Link Layer may be possible. The communication between the controller and a module on the same bus segment can be described as follows.

## 4.2 CAN-Bus Implementation

The data frame structure is matched to the message frame of the standard-format according to CAN specification 2.0A, whereas looking from the point of view of the CAN protocol a pure data transmission will be done, which is not applying to the protocol. The data frame of the EDCP will be transferred as data word with n bytes length in the data field of the CAN frame according to the specific demand of the related access. Therefore this results into a Data Length Code (DLC) of the CAN-protocol of n.

It is possible to transfer 8 data bytes that apply to the DLC field with decreasing values.

The addressing of the Front-end device is also made using the 11 bit identifier of the CAN protocol.

In order to keep the CAN segment open also for other protocols, the address room has been limited to 64 nodes.

ID10 is dominant.

ID9 When the Event structure of the module was configured and the bit isEvtActive in the ModuleStatus was triggered, then the module will send the DCP Module frame General status as an active message with higher priority (ID9 = 0) than normal messages.

ID8 to ID3

allow the addressing of 64 Front-end devices (ID3: A0 = 20 ;...; ID8: A5 = 25 ), see 3.2 Back Panel also.

ID2 is used for a special network management service (NMT).

ID1 is not used.

ID0 is used for defining the direction of the data transfer (DATA\_DIR). The controller therefore will start a read-write access for data with DATA\_DIR = 1 and will send data with DATA\_DIR=0. The Front-end device responds to the data request by sending the corresponding data with DATA\_DIR = 0.

That means all "even" CAN-ports (Identifier) are interpreted as 'Write ports' all "odd" CAN ports as 'Read ports'.

Only if the Front-end device is not registered at the controller or if it does not receive valid data during a longer time period (ca. 1 min), then it will actively send the registration frame with DATA\_DIR = 1 (see also item 4.3). The RTR Bit is always set to zero.

In one CAN segment modules with unequal identifier and equal bit rate are allowed only. The factory fixed bit rate is written on the sticker of the 96-pin connector.

Data formats:

The data format on the network is big endian, i.e. on Intel computers, the value is stored byte-wise reverse. To convert floating-point values to their hexadecimal representation, the online calculator <http://babbage.cs.qc.edu/IEEE-754/> can be used.

|     |                                                     |
|-----|-----------------------------------------------------|
| UI1 | unsigned character                                  |
| SI1 | signed character                                    |
| UI2 | unsigned short integer (16 bit)                     |
| UI4 | unsigned integer (32 bit)                           |
| R4  | float according to IEEE-754 single precision format |

Example Vset = 1000 V:

Data-Bytes on the network – 0x44 0x7a 0x00 0x00

Data-Bytes in computers using a little endian memory – 0x00 0x00 0x7a 0x44

Conventional CAN data frame to control of the HV modules, see ehq\_multi\_channel\_can also.

|   |            |   |         |    |         |                                               |              |              |              |           |     |    |     |    |     |     |        |
|---|------------|---|---------|----|---------|-----------------------------------------------|--------------|--------------|--------------|-----------|-----|----|-----|----|-----|-----|--------|
| S | Identifier | R |         |    | DLC     | n - data bytes (1 to 8)                       |              |              |              |           |     |    |     |    |     | CRC | ack    |
| O |            | T | 0       | 0  | (n=1-8) | DATA_ID<br>Single channel<br>access           | CHN          | DATA_(n-3) 0 | DATA_(n-4) 0 | DATA_ ... |     |    |     |    |     |     | F.     |
| F | b10 ... b0 | R | Reserve | b3 | b0      | b15=01 0 0 b0                                 | C7           | C0           | b7           | b0        | b7  | b0 | b7  | b0 | b7  | b0  | 15 bit |
| S | Identifier | R |         |    | DLC     | n - data bytes (1 to 8)                       |              |              |              |           |     |    |     |    |     | CRC | ack    |
| O |            | T | 0       | 0  | (n=1-8) | DATA_ID<br>Multiple single<br>channels access | MBR          | DATA_(n-4) 0 | DATA_(n-5) 0 | DATA_ ... |     |    |     |    |     |     | F.     |
| F | b10 ... b0 | R | Reserve | b3 | b0      | b15=01 1 0 b0                                 | M1           | M0           | b7           | b0        | b7  | b0 | b7  | b0 | b7  | b0  | 15 bit |
| S | Identifier | R |         |    | DLC     | n - data bytes (1 to 8)                       |              |              |              |           |     |    |     |    |     | CRC | ack    |
| O |            | T | 0       | 0  | (n=1-8) | DATA_ID<br>Group access                       | NBR          | OFFSET       | ChList       | Type ...  |     |    |     |    |     |     | F.     |
| F | b10 ... b0 | R | Reserve | b3 | b0      | b15=00 1 0 b0                                 | N7           | N0           | b7           | b0        | b15 | b0 | b15 | b0 | b15 | b0  | 15 bit |
| S | Identifier | R |         |    | DLC     | n - data bytes (1 to 8)                       |              |              |              |           |     |    |     |    |     | CRC | ack    |
| O |            | T | 0       | 0  | (n=1-8) | DATA_ID<br>Module access                      | DATA_(n-2) 0 | DATA_(n-3) 0 | DATA_ ...    |           |     |    |     |    |     |     | F      |
| F | b10 ... b0 | R | Reserve | b3 | b0      | b15=00 0 1 b0                                 | b7           | b0           | b7           | b0        | b7  | b0 | b7  | b0 | b7  | b0  | 15bit  |

|      |     |     |     |     |     |     |     |     |     |          |
|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|----------|
| ID10 | ID9 | ID8 | ID7 | ID6 | ID5 | ID4 | ID3 | ID2 | ID1 | ID0      |
| 0    | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 1   | 0   | DATA_DIR |

1. Acceptance-Filter of the CAN-Controller is set to NMT service identifier

|      |     |     |     |     |     |     |     |     |     |          |
|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|----------|
| ID10 | ID9 | ID8 | ID7 | ID6 | ID5 | ID4 | ID3 | ID2 | ID1 | ID0      |
| 0    | P   | A5  | A4  | A3  | A2  | A1  | A0  | 0   | 0   | DATA_DIR |

2. Acceptance-Filter of the CAN-Controller is set to FE-address A0 - A5

The Front-end device must do:

- Processing of NMT services via broadcast messages inside of the CAN segment
- Processing of the single accesses with direct channel values.
- Processing of group information of the module.
- Self-registration in the higher level through sending the module address.
- Building of status information.
- Send an active error message with higher priority if one of the bits - sum status, supply voltages or safety loop - in the group access "General status module" not has been set (the module must be configured as a CAN-node with an Active-CAN message function).

## 4.3 Summary of CAN data frame accesses via the NMT service identifier

| Access                               | DAT<br>A_D<br>IR | NMT DATA_ID<br>Bit |   |    |    |    |    |    |    | read<br>/<br>write<br>/<br>activ<br>e | DATA<br>-<br>Bytes | Page |
|--------------------------------------|------------------|--------------------|---|----|----|----|----|----|----|---------------------------------------|--------------------|------|
|                                      | ID0              | 7                  | 6 | 5  | 4  | 3  | 2  | 1  | 0  |                                       |                    |      |
| No NMT DATA_ID                       | x                | 0                  | x | x  | x  | x  | x  | x  | x  |                                       |                    |      |
| NMT service CAN segment:             | 0                | 1                  | 1 | N3 | N2 | N1 | N0 | R1 | R0 |                                       |                    |      |
| NMT Address (MICC only)              | 0/1              | 0                  | 0 | 1  | 1  | 0  | 0  | 0  | 0  | w                                     | 0/2/3              |      |
| NMT Start                            | 0                | 1                  | 1 | 0  | 0  | 0  | 0  | 1  | x  | x                                     | w                  | 1    |
| NMT Stop                             | 0                | 1                  | 1 | 0  | 0  | 1  | 0  | x  | x  | w                                     | 1                  | 18   |
| NMT Reset CAN                        | 0                | 1                  | 1 | 0  | 0  | 1  | 1  | x  | x  | w                                     | 1                  | 18   |
| NMT Reset hardware                   | 0                | 1                  | 1 | 0  | 1  | 0  | 0  | x  | x  | w                                     | 1                  | 18   |
| NMT set of Bit rate                  | 0                | 1                  | 1 | 0  | 1  | 0  | 1  | x  | x  | w                                     | 3                  | 18   |
| NMT temperature set                  | 0                | 1                  | 1 | 0  | 1  | 1  | 0  | x  | x  | w                                     | 3                  | 19   |
| NMT mode set                         | 0                | 1                  | 1 | 1  | 0  | 0  | 0  | x  | x  | w                                     | 2/6                | 19   |
| NMT set standard DCP or enhanced DCP | 0                | 1                  | 1 | 1  | 0  | 0  | 1  | x  | x  | w                                     | 2                  | 19   |
| NMT channel group set                | 0                | 1                  | 1 | 1  | 0  | 1  | 0  | x  | x  | w                                     | 8/6                | 19   |
| NMT module set                       | 0                | 1                  | 1 | 1  | 0  | 1  | 1  | x  | x  | w                                     | 8/6                | 19   |
| Ni:                                  | NMT access       |                    |   |    |    |    |    |    |    |                                       |                    |      |
| Ri:                                  | reserved         |                    |   |    |    |    |    |    |    |                                       |                    |      |

### NMT Address (MICC only)

| Access                   | CAN ID (NMT) | RTR | EXT_INSTR | DATA_DIR | DATA_ID | DATA_1      | DATA_0      |
|--------------------------|--------------|-----|-----------|----------|---------|-------------|-------------|
| NMT-RTR master           | 0x004        | 1   | 0         | 0        | -       | -           | -           |
| NMT-RTR board reply      | 0x003        | 0   | 0         | 0        | 0xC0    | address     | -           |
| NMT Address master write | 0x004        | 0   | 0         | 0        | 0xC0    | old address | new address |

NMT Start The state of all Front-end devices is going to OPERATIONAL (see [Appendix C](#))

NMT Stop The state of all Front-end devices is going to PREPARED

This is necessary before storing any information permanently in EEPROM or execute one of the following NMT services

NMT Reset CAN re - initialise all connected iseg Multi-Channel CAN devices.

NMT Reset hardware execute a hardware reset of all connected CAN devices.

NMT set of Bit rate set a new bit rate for all connected iseg Multi-Channel CAN devices (DATA\_1 / DATA\_0 see group access [Bit rate](#))

NMT set of temperature An offset for the calculation of the temperature will be calculated in all modules which receive this message.

all devices

DATA\_1 to DATA\_0 measured temperature in tenth parts of °C UI2

| DATA_1 | DATA_0 |
|--------|--------|
| MSB    | LSB    |

NMT mode

| Mode                | Data_0    | Description                                        |
|---------------------|-----------|----------------------------------------------------|
| NORMAL_MODE         | 000       | Operating Mode "Operational" without any condition |
| FACTORY MODES       | 001 - 005 | will used in production of the modules only!       |
| LIVE_INSERTION_MODE | 006       | Mode to avoid events during a live insertion       |

NMT set standard DCP or enhanced DCP: DATA\_0=0 standard DCP DCP  
DATA\_0=1 enhanced DCP EDCP

NMT broadcast messages

NMT channel group set frame:

| Access                   | DATA_DIR | NMT DATA_ID | GROUP | EDCP DATA_IDEDCP Multiple Single Channels Access | DATAN            | DATAN-1 | DATAN-2 | DATAN-3 |
|--------------------------|----------|-------------|-------|--------------------------------------------------|------------------|---------|---------|---------|
| NMT group voltage set    | 0        | 0xe8        | Group | 0x6100                                           | Voltage [R4]     |         |         |         |
| NMT group current set    | 0        | 0xe8        | Group | 0x6101                                           | Current [R4]     |         |         |         |
| NMT group control set    | 0        | 0xe8        | Group | 0x6001                                           | Control [UI2]    |         |         |         |
| NMT group event mask set | 0        | 0xe8        | Group | 0x6003                                           | Event mask [UI2] |         |         |         |

Group 0..255 (group = 0 after power on of the module)

NMT module set frame:

|                            |   |      |    |        |                         |  |  |  |
|----------------------------|---|------|----|--------|-------------------------|--|--|--|
| NMT voltage ramp speed set | 0 | 0xec | xx | 0x1100 | Voltage ramp speed [R4] |  |  |  |
| NMT current ramp speed set | 0 | 0xec | xx | 0x1101 | Current ramp speed [R4] |  |  |  |
| NMT control set            | 0 | 0xec | xx | 0x1001 | Control [UI2]           |  |  |  |
| NMT event mask             | 0 | 0xec | xx | 0x1003 | Event mask [UI2]        |  |  |  |
| NMT event channel mask     | 0 | 0xec | xx | 0x1005 | Event mask [UI2]        |  |  |  |

xx reserved

With one of the NMT channel group set or the NMT module set frames a message is sent to the corresponding data point of the table above in kind of a broadcast information for all channels, which have the same group number GROUP. The detailed description of the frames can be found by a click on the arrows of the tables. The EDCP Single Channel Access [GroupNumber](#) (described on page 30) handles the distribution of a group number for each channel.

The NMT channel group set and NMT module set frames has been implemented since following firmware releases:

| Name of firmware | Firmware release | Device class |
|------------------|------------------|--------------|
| E16D0            | 00.04.02.01      | 21           |
| E16D1            | 00.04.02.01      | 21           |
| E08C0            | 00.02.02.00      | 24           |
| E16C1            | 00.01.00.00      | 70           |
| E08C2            | 00.01.00.00      | 27           |
| E08F0            | 00.02.02.08      | 25           |
| E08F2            | 00.04.00.06      | 26           |

Example: Switch ON all channels of the whole system (group number after reset of all channels is zero)

| ID    | DLC | NMT DATA_ID | GROUP | EDCP DATA_ID Multiple Single Channel Access | Channel control |
|-------|-----|-------------|-------|---------------------------------------------|-----------------|
| 0x004 | 0x6 | 0xe6        | 0x00  | 0x6001                                      | 0x0008          |

## 4.4 Summary of CAN data frame accesses via the Front-end-address identifier

Multi-channel High Voltage CAN modules are made out of one or two PCBs (in order to double the number of HV channels) and one digital CAN Interface per PCB.

Each module board has to be controlled separately via its own CAN nodes identifier (see chapter above).

### 4.4.1 List to access of the EDCP made for HV boards up to 255 channels EDCP Single Channel Accesses

| Access                                       | DATA_DIR | DATA_ID |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | read / write / active | DATA-Bytes | Page |
|----------------------------------------------|----------|---------|-----|----|----|----|-----|-----|----|----|----|----|----|----|----|----|----|----|-----------------------|------------|------|
|                                              |          | Word    | Bit |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    |                       |            |      |
|                                              | ID0      | hex     | 15  | 14 | 13 | 12 | 11  | 10  | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |                       |            |      |
| DATA_ID                                      |          |         | 0   | S  | G  | M  | x   | x   | x  | x  | x  | x  | x  | x  | x  | x  | x  | x  |                       |            |      |
| Single channel access                        | 1/0      | 0x4xxx  | 0   | 1  | 0  | 0  | S11 | S10 | S9 | S8 | S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 |                       |            |      |
| <a href="#">ChannelStatus</a>                | 1        | 0x4000  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | r                     | 3/5        | 34   |
| <a href="#">ChannelControl</a>               | 0/1      | 0x4001  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | w/r                   | 3/5        | 35   |
| <a href="#">ChannelEventStatus</a>           | 1/0      | 0x4002  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | r/w                   | 3/5        | 35   |
| <a href="#">ChannelEventMask</a>             | 0/1      | 0x4003  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | w/r                   | 3/5        | 36   |
| <a href="#">DelayedTripAction</a>            | 0/1      | 0x4006  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | w/r                   | 3/4        | 37   |
| <a href="#">DelayedTripTime</a>              | 0/1      | 0x4005  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 0  | w/r                   | 3/5        | 37   |
| <a href="#">ExternalInhibitAction</a>        | 0/1      | 0x4007  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 1  | w/r                   | 3/4        | 37   |
| <a href="#">VoltageSet</a>                   | 0/1      | 0x4100  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | w/r                   | 3/7        | 38   |
| <a href="#">CurrentSet /<br/>CurrentTrip</a> | 0/1      | 0x4101  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | w/r                   | 3/7        | 38   |
| <a href="#">VoltageMeasure</a>               | 1        | 0x4102  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | r                     | 3/7        | 39   |
| <a href="#">CurrentMeasure</a>               | 1        | 0x4103  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | r                     | 3/7        | 39   |
| <a href="#">VoltageBounds</a>                | 0/1      | 0x4104  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 0  | w/r                   | 3/7        | 39   |
| <a href="#">CurrentBounds</a>                | 0/1      | 0x4105  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | w/r                   | 3/7        | 39   |
| <a href="#">VoltageNominal</a>               | 1        | 0x4106  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 0  | r                     | 3/7        | 40   |
| <a href="#">CurrentNominal</a>               | 1        | 0x4107  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 1  | r                     | 3/7        | 40   |
| <a href="#">CurrentMeasure Range</a>         | 1        | 0x4109  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 1  | 0  | 0  | 1  | r                     | 3/8        | 40   |

|                                                                                                                                                                                                                                                                                                                       |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |     |     |    |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|-----|-----|----|
| <u>VctCoefficient</u>                                                                                                                                                                                                                                                                                                 | 1/0 | 0x4120 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | r/w | 3/7 | 40 |
| <u>TemperatureExternal</u>                                                                                                                                                                                                                                                                                            | 1/0 | 0x4121 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | r/w | 3/7 | 41 |
| <u>GroupNumber</u>                                                                                                                                                                                                                                                                                                    | 1/0 | 0x4200 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | w/r | 3/4 | 41 |
| <p>S      DATA_ID type bit for a EDCP-frame of an access to a single channel</p> <p>G      DATA_ID type bit for a EDCP-frame of an access to a group of channels</p> <p>M      DATA_ID type bit for a EDCP-frame of an access to the whole module</p> <p>S<sub>i</sub>; (i=0..11)      single channel access bits</p> |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |     |     |    |

# EDCP Multiple Single Channels Access

| Access                                   | DATA_DIR | DATA_ID |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | read / write / active | DATA-Bytes | Page |
|------------------------------------------|----------|---------|-----|----|----|----|-----|-----|----|----|----|----|----|----|----|----|----|----|-----------------------|------------|------|
|                                          |          | Word    | Bit |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    |                       |            |      |
|                                          | ID0      | hex     | 15  | 14 | 13 | 12 | 11  | 10  | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |                       |            |      |
| DATA_ID                                  |          |         | 0   | S  | G  | M  | x   | x   | x  | x  | x  | x  | x  | x  | x  | x  | x  | x  |                       |            |      |
| Single channel access                    | 1        | 0x6xxx  | 0   | 1  | 1  | 0  | S11 | S10 | S9 | S8 | S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 |                       |            |      |
| <a href="#">ChannelStatus</a>            | 1        | 0x6000  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | r                     | 5/5        | 34   |
| <a href="#">ChannelControl</a>           | 1        | 0x6001  | 0   | 1  | 0  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | r                     | 5/5        | 35   |
| <a href="#">ChannelEventStatus</a>       | 1        | 0x6002  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | r                     | 5/5        | 35   |
| <a href="#">ChannelEventMask</a>         | 1        | 0x6003  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | r                     | 5/5        | 36   |
| <a href="#">DelayedTripAction</a>        | 0/1      | 0x6006  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | w/r                   | 3/4        | 37   |
| <a href="#">DelayedTripTime</a>          | 0/1      | 0x6005  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 0  | w/r                   | 3/5        | 37   |
| <a href="#">ExternalInhibitAction</a>    | 0/1      | 0x6007  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 1  | w/r                   | 3/4        | 37   |
| <a href="#">VoltageSet</a>               | 1        | 0x6100  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | r                     | 5/7        | 38   |
| <a href="#">CurrentSet / CurrentTrip</a> | 1        | 0x6101  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | r                     | 5/7        | 38   |
| <a href="#">VoltageMeasure</a>           | 1        | 0x6102  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | r                     | 5/7        | 39   |
| <a href="#">CurrentMeasure</a>           | 1        | 0x6103  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | r                     | 5/7        | 39   |
| <a href="#">VoltageBounds</a>            | 1        | 0x6104  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 0  | r                     | 5/7        | 39   |
| <a href="#">CurrentBounds</a>            | 1        | 0x6105  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | r                     | 5/7        | 39   |
| <a href="#">VoltageNominal</a>           | 1        | 0x6106  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 0  | r                     | 5/7        | 40   |
| <a href="#">CurrentNominal</a>           | 1        | 0x6107  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 1  | r                     | 5/7        | 40   |
| <a href="#">CurrentMeasure Range</a>     | 1        | 0x6109  | 0   | 1  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 1  | 0  | 0  | 1  | r                     | 3/8        | 40   |

|                                                                                                                                                                                                                                                                                                  |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |     |     |    |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|-----|-----|----|
| <u>VctCoefficient</u>                                                                                                                                                                                                                                                                            | 1/0 | 0x6120 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | r/w | 3/7 | 40 |
| <u>TemperatureExternal</u>                                                                                                                                                                                                                                                                       | 1/0 | 0x6121 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | r/w | 3/7 | 41 |
| <u>ChannelGroup</u>                                                                                                                                                                                                                                                                              | 0   | 0x6200 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | w   | 6   | 41 |
| S      DATA_ID type bit for a EDCP-frame of an access to single channel<br>G      DATA_ID type bit for a EDCP-frame of an access to a group of channels<br>M      DATA_ID type bit for a EDCP-frame of an access to the whole module<br>S <sub>i</sub> (i=0..11)      single channel access bits |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |     |     |    |

## Module Access

| Access                                   | DATA_DIR | DATA_ID |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | read / write / active | DATA-Bytes | Page |
|------------------------------------------|----------|---------|-----|----|----|----|-----|-----|----|----|----|----|----|----|----|----|----|----|-----------------------|------------|------|
|                                          |          | Word    | Bit |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    |                       |            |      |
|                                          | ID0      | hex     | 15  | 14 | 13 | 12 | 11  | 10  | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |                       |            |      |
| DATA_ID                                  |          |         | 0   | S  | G  | M  | x   | x   | x  | x  | x  | x  | x  | x  | x  | x  | x  | x  |                       |            |      |
| Module access                            | 1/0      | 0x1xxx  | 0   | 0  | 0  | 1  | M11 | M10 | M9 | M8 | M7 | M6 | M5 | M4 | M3 | M2 | M1 | M0 |                       |            |      |
| <a href="#">ModuleStatus</a>             | 1        | 0x1000  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | r                     | 2/4        | 42   |
| <a href="#">ModuleControl</a>            | 0        | 0x1001  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | w/r                   | 4/2        | 43   |
| <a href="#">ModuleEventStatus</a>        | 1/0      | 0x1002  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | r/w                   | 2/4        | 43   |
| <a href="#">ModuleEventMask</a>          | 0/1      | 0x1003  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | w/r                   | 4/2        | 44   |
| <a href="#">ModuleEventChannelStatus</a> | 1/0      | 0x1004  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 0  | r/w                   | 2/4        | 44   |
| <a href="#">ModuleEventChannelMask</a>   | 0/1      | 0x1005  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | w/r                   | 4/2        | 45   |
| <a href="#">ModuleEventGroupStatus</a>   | 0/1      | 0x1006  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 0  | r/w                   | 2/4        | 45   |
| <a href="#">ModuleEventGroupMask</a>     | 0/1      | 0x1007  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 1  | w/r                   | 4/2        | 45   |
| <a href="#">VoltageRampSpeed</a>         | 0/1      | 0x1100  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | w/r                   | 6/2        | 46   |
| <a href="#">CurrentRampSpeed</a>         | 0/1      | 0x1101  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | w/r                   | 6/2        | 46   |
| <a href="#">VoltageMax</a>               | r        | 0x1102  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | r                     | 2/6        | 46   |
| <a href="#">CurrentMax</a>               | r        | 0x1103  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | r                     | 2/6        | 47   |
| <a href="#">Supply24</a>                 | r        | 0x1104  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 0  | r                     | 2/6        | 47   |
| <a href="#">Supply5</a>                  | r        | 0x1105  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | r                     | 2/6        | 47   |
| <a href="#">BoardTemperature</a>         | 0/1      | 0x1106  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 0  | r                     | 2/6        | 47   |
| ThresholdArmErrorDetect.                 | 0/1      | 0x1107  | 0   | 0  | 0  | 1  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 1  | w/r                   | 6/2        | 48   |

|                             |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |     |     |    |
|-----------------------------|-----|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|-----|-----|----|
| <u>SerialNumber</u>         | 1   | 0x1200 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | r   | 2/6 | 48 |
| <u>FirmwareRelease</u>      | 1   | 0x1201 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | r   | 2/6 | 48 |
| <u>BitRate</u>              | 0/1 | 0x1202 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | r   | 4/2 | 48 |
| <u>NameOfFirmware</u>       | 1   | 0x1203 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | r   | 5/6 | 36 |
| <u>ADC SamplesPerSecond</u> | 0/1 | 0x1204 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | w/r | 4/2 | 37 |
| <u>DigitalFilter</u>        | 0/1 | 0x1205 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | w/r | 4/2 | 37 |

|                                                                         |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |         |         |   |
|-------------------------------------------------------------------------|-----|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---------|---------|---|
| <u>ModuleOption</u>                                                     | 1   | 0x1280 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | r | 6       | 38      |   |
| <u>ModuleOptionSpec</u>                                                 | 1   | 0x1290 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | r | 7       | 38      |   |
| <u>ModuleCommMode</u>                                                   | 1   | 0x12a0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | w | 4       | 38      |   |
| Factory settings                                                        | 1/0 | 0x140x | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x | r/<br>w | 4/<br>8 | - |
| S DATA_ID type bit for a EDCP-frame of an access to a single channel    |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |         |         |   |
| G DATA_ID type bit for a EDCP-frame of an access to a group of channels |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |         |         |   |
| M DATA_ID type bit for a EDCP-frame of an access to the whole module    |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |         |         |   |
| M <sub>i;i=0..11</sub> module access bits                               |     |        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |         |         |   |

# EDCP Group Accesses

| Access                                                                                                                                                                                                                                                                | DATA_DIR | DATA_ID |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | read / write / active | DATA-Bytes | Page |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|---------|-----|----|----|----|-----|-----|----|----|----|----|----|----|----|----|----|----|-----------------------|------------|------|
|                                                                                                                                                                                                                                                                       |          | Word    | Bit |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    |                       |            |      |
|                                                                                                                                                                                                                                                                       | ID0      | hex     | 15  | 14 | 13 | 12 | 11  | 10  | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |                       |            |      |
| DATA_ID                                                                                                                                                                                                                                                               |          |         | 0   | S  | G  | M  | G11 | G10 | G9 | G8 | G7 | G6 | G5 | G4 | G3 | G2 | G1 | G0 |                       |            |      |
| Groups<br>SetGroup<br>StatusGroup<br>MonitorGroup<br>TripGroup                                                                                                                                                                                                        | 0/1      | 0x2000  | 0   | 0  | 1  | 0  | 0   | 0   | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | w/r                   | 8/4        | 52   |
| VoltageSetAllChannels                                                                                                                                                                                                                                                 | 0        | 0x2100  | 0   | 0  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | w                     | 6          | 56   |
| CurrentSetAllChannels                                                                                                                                                                                                                                                 | 0        | 0x2101  | 0   | 0  | 1  | 0  | 0   | 0   | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | w                     | 6          | 57   |
| SetOnOffAllChs                                                                                                                                                                                                                                                        |          | 0x2200  |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | r/w                   | 6          | 44   |
| SetEmergencyAllChs                                                                                                                                                                                                                                                    |          | 0x2201  |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | r/w                   | 6          | 45   |
| EventStatusVLimtAllChs                                                                                                                                                                                                                                                |          | 0x2202  |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | r/w                   | 6          | 45   |
| EventStatusCLimtAllChs                                                                                                                                                                                                                                                |          | 0x2203  |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | r/w                   | 6          | 46   |
| EventStatusTrpAllChs                                                                                                                                                                                                                                                  |          | 0x2204  |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | r/w                   | 6          | 47   |
| EventStatusInhAllChs                                                                                                                                                                                                                                                  |          | 0x2205  |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    | r/w                   | 6          | 47   |
| SetOnOffChsExtender                                                                                                                                                                                                                                                   |          | 0x2280  | 0   | 0  | 1  | 0  | 0   | 0   | 0  | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | r/w                   | 6          | 48   |
| SetEmergencyChsExten.                                                                                                                                                                                                                                                 |          | 0x2290  | 0   | 0  | 1  | 0  | 0   | 0   | 0  | 1  | 1  | 0  | 0  | 1  | 0  | 0  | 0  | 0  | r/w                   | 6          | 48   |
| S DATA_ID type bit for a EDCP-frame of an access to a single channel<br>G DATA_ID type bit for a EDCP-frame of an access to a group of channels<br>M DATA_ID type bit for a EDCP-frame of an access to the whole module<br>G <sub>i</sub> (i=0..11) group access bits |          |         |     |    |    |    |     |     |    |    |    |    |    |    |    |    |    |    |                       |            |      |

## Important DCP Module Access

| Access                                          | EXT_I<br>NSTR | DATA_<br>DIR | DATA_ID |     |   |    |    |    |    |    |    |     | read<br>/<br>write<br>/<br>active | DAT<br>A-<br>Byte<br>s | Page |
|-------------------------------------------------|---------------|--------------|---------|-----|---|----|----|----|----|----|----|-----|-----------------------------------|------------------------|------|
|                                                 |               |              | Byte    | Bit |   |    |    |    |    |    |    |     |                                   |                        |      |
|                                                 | ID1           | ID0          |         | 7   | 6 | 5  | 4  | 3  | 2  | 1  | 0  |     |                                   |                        |      |
| Group access MODULE:                            | 0             | 1/0          |         | 1   | 1 | M3 | M2 | M1 | M0 | R1 | R0 |     |                                   |                        |      |
| <u>GeneralStatus</u>                            | 0             | 1/0          | 0xc0    | 1   | 1 | 0  | 0  | 0  | 0  | 0  | 0  | a   | 3                                 | 61                     |      |
| <u>LogOnOff</u> Front-end at the superior layer | 0             | 1/0          | 0xD8    | 1   | 1 | 0  | 1  | 1  | 0  | 0  | 0  | a/w | 3                                 | 62                     |      |

## 4.5 Description of data information per DATA\_ID in EDCP

### 4.5.1 EDCP Single Access

The single access describes the control of the channel properties. The range of the single access contains the accesses to the analog digital data items, to the status and the control words of the channels.

#### 4.5.1.1 Channel status (single/multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_1        | DATA_0 |
|-------------------------|----------|---------------|--------|---------------|--------|
| master single read-     | 1        | 0x4000        | Mx     |               |        |
| master single MBR read- | 1        | 0x6000        | Member |               | Offset |
| HV board write access   | 0        | 0x4000/0x6000 | Mx     | ChannelStatus |        |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... too access up to 255 channels  
ChannelStatus DATA\_0 to DATA\_1 UI2

| Bit15  | Bit14  | Bit13 | Bit12  | Bit11  | Bit10  | Bit9     | Bit8  | Bit7 | Bit6 | Bit5   | Bit4   | Bit3 | Bit2 | Bit1  | Bit0 |
|--------|--------|-------|--------|--------|--------|----------|-------|------|------|--------|--------|------|------|-------|------|
| isVLIM | isCLIM | isTRP | isEINH | isVBND | isCBND | isArcErr | isLCR | isCV | isCC | isEMCY | isRAMP | isON | IERR | isArc | res  |

The ChannelStatus register describes the actual status. Depending on the status of the module the bits will be set or reset. The bit InputError will be set if the given parameter is not plausible or it exceeds the module parameters (e.g. if the command Vset=4000V is given to a module with NominalVoltage=3000V). The bit InputError is not be set if the given values are temporarily not possible (e.g. Vset=2800 at a module with NominalVoltage=3000V, but HardwareLimitVoltage=2500V). A certain signature which kind of input error it is does not exists.

|          |                            |                                                                                                                                              |
|----------|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| isVLIM   | IsVoltageLimitExceeded     | voltage limit set by Vmax is exceeded                                                                                                        |
| isCLIM   | IsCurrentLimitExceeded     | current limit set by Imax is exceeded                                                                                                        |
| isTRP    | IsTripExceeded             | Trip is set when Voltage or Current limit or Iset has been exceeded (when KillEnable=1 )                                                     |
| isEINH   | IsExternalInhibit          | External Inhibit                                                                                                                             |
| isVBND   | IsVoltageBoundsExceeded    | Voltage out of bounds                                                                                                                        |
| isCBND   | IsCurrentBoundsExceeded    | Current out of bounds                                                                                                                        |
| isArcErr | IsArcError                 | maximum number of allowed arcs is exceeded, high voltage has been turned off                                                                 |
| isLCR    | IsLowCurrentRange          | Low or small current range of the current measurement                                                                                        |
| isCV     | IsControlledVoltage        | Voltage control active (evaluation is guaranteed when no ramp is running)                                                                    |
| isCC     | IsControlledCurrent        | Current control active (evaluation is guaranteed when no ramp is running)                                                                    |
| isEMCY   | IsEmergencyOff             | Emergency off without ramp                                                                                                                   |
| isON     | IsOn                       | On                                                                                                                                           |
| isRAMP   | IsRamping                  | Ramp is running                                                                                                                              |
| IERR     | InputError                 | Input error                                                                                                                                  |
| IsArc    | isArc or IsRegulationError | at least one electrical arc is active or faster error detection of the channel hardware is not in regulation (updated by firmware every 5ms) |
| res      | reserved                   |                                                                                                                                              |

|          |                                                                                                                                                                   |            |                                                                                                                                     |
|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|-------------------------------------------------------------------------------------------------------------------------------------|
| isVLIM=0 | channel is ok                                                                                                                                                     | isVBND=1   | Vmeas-Vset  > Vbounds                                                                                                               |
| isVLIM=1 | the hardware voltage limit is exceeded                                                                                                                            | isCBND=0   | channel is ok                                                                                                                       |
| isCLIM=0 | channel is ok                                                                                                                                                     | isCBND=1   | Imeas-Iset  > Ibounds (to detect a voltage or current out of bound flag the firmware has to ramp the channel voltage Vset at first) |
| isCLIM=1 | the hardware current limit is exceeded (to detect a hardware voltage or current error flag the firmware has to evaluate the channel voltage and current at first) | isArcErr=0 | no arc error                                                                                                                        |
| limit    |                                                                                                                                                                   | isArcErr=1 | maximum number of allowed arcs is exceeded                                                                                          |
| isTRP=0  | channel is ok                                                                                                                                                     | isLCR=0    | high or standard current range                                                                                                      |
| isTRP=1  | VO is shut off to 0V without ramp because the channel has been tripped.                                                                                           | isLCR=1    | low current range of the current measurement                                                                                        |
| isEINH=0 | channel is ok                                                                                                                                                     | sCV=1      | channel is in state of voltage control                                                                                              |
| isEINH=1 | External Inhibit was scanned                                                                                                                                      | isCC=1     | channel is in state of current control                                                                                              |
| isVBND=0 | channel is ok                                                                                                                                                     | isEMCY=1   | channel is in state of emergency off, VO has been shut off to 0V without ramp                                                       |

|          |                                                       |         |                                                                                       |
|----------|-------------------------------------------------------|---------|---------------------------------------------------------------------------------------|
| isON=0   | channel is off                                        | IERR=0  | no input-error                                                                        |
| isON=1   | channel voltage follows the Vset value                | IERR=1  | incorrect message to control the channel                                              |
| isRAMP=0 | no voltage is in change                               | isARC=0 | no arc active / normal error evaluation                                               |
| isRAMP=1 | voltage is in change with the stored ramp speed value | isARC=1 | at least one electrical arc is active / fast detection of a regulation error (OPTION) |

#### 4.5.1.2 Channel control: (single write- and single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_1         | DATA_0 |
|-------------------------|----------|---------------|--------|----------------|--------|
| master write access     | 0        | 0x4001        | Mx     | ChannelControl |        |
| master read-            | 1        | 0x4001        | Mx     |                |        |
| master single MBR read- | 1        | 0x6001        | Member | Offset         |        |
| HV board write access   | 0        | 0x4001/0x6001 | Mx     | ChannelControl |        |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... to access up to 255 channels

| ChannelControl |       |       | DATA_0 to DATA_1 |       |       |      |      |      | UI2  |         |      |       |      |      |      |
|----------------|-------|-------|------------------|-------|-------|------|------|------|------|---------|------|-------|------|------|------|
| Bit15          | Bit14 | Bit13 | Bit12            | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5    | Bit4 | Bit3  | Bit2 | Bit1 | Bit0 |
| res            | res   | res   | res              | res   | res   | res  | res  | res  | res  | setEMCY | res  | setON | res  | res  | res  |

The signals SetOn and SetEmergencyOff control are basic functions of the channel. The signal SetOn is switching ON the HV of the channel and is a precondition for giving voltage to the output. As far as a VoltageSet has been set and no event has occurred and is not registered yet (in minimum, bit 5 and bit 10 to 15 of the register Channel Event Status must be 0), a start of a HV ramp will be synchronized (a ramp is a software controlled, time proportionally increase / decrease of the output voltage ).

A SetEmergencyOff switch the channel in state isEmergencyOff and a new SetOn is only possible after SetEmergencyOff is reset to zero

|         |                 |                     |
|---------|-----------------|---------------------|
| setEMCY | SetEmergencyOff | Set "Emergency Off" |
| setON   | SetOn           | Set On              |
| res     | Reserved        |                     |

setEMCY=0 channel emergency cut-off works  
setEMCY=1 cut-off VO shut off to 0V without ramp  
setOn=0 switch the channel to OFF  
setOn=1 switch the channel to ON

#### 4.5.1.3 Channel event status (single write- and single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_1             | DATA_0 |
|-------------------------|----------|---------------|--------|--------------------|--------|
| master write access     | 0        | 0x4002        | Mx     | ChannelEventStatus |        |
| master read-            | 1        | 0x4002        | Mx     |                    |        |
| master single MBR read- | 1        | 0x6002        | Member | Offset             |        |
| HV board write access   | 0        | 0x4002/0x6002 | Mx     | ChannelEventStatus |        |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... too access up to 255 channels

| ChannelEventStatus |       |       | DATA_0 to DATA_1 |        |        |      | UI2  |      |      |       |      |         |      |      |      |
|--------------------|-------|-------|------------------|--------|--------|------|------|------|------|-------|------|---------|------|------|------|
| Bit15              | Bit14 | Bit13 | Bit12            | Bit11  | Bit10  | Bit9 | Bit8 | Bit7 | Bit6 | Bit5  | Bit4 | Bit3    | Bit2 | Bit1 | Bit0 |
| EVLIM              | ECLIM | ETRP  | EEINH            | EVBNDs | ECBNDs | EAE  | res  | ECV  | ECC  | EEMCY | EEOR | EOn2Off | EIER | EARC | res  |

|        |                      |                                                                                                                              |
|--------|----------------------|------------------------------------------------------------------------------------------------------------------------------|
| EVLIM  | EventVoltageLimit    | Event: Hardware- voltage limit has been exceeded                                                                             |
| ECLIM  | EventCurrentLimit    | Event: Hardware- current limit has been exceeded                                                                             |
| ETRP   | EventTrip            | Event: Trip is set when Voltage or Current limit or Iset has been exceeded (when KillEnable=1 or Delayed Trip is configured) |
| EEINH  | EventExternalInhibit | Event external Inhibit                                                                                                       |
| EVBNDs | EventVoltageBounds   | Event: Voltage out of bounds                                                                                                 |
| ECBNDs | EventCurrentBounds   | Event: Current out of bounds                                                                                                 |

|         |                        |                                                                        |
|---------|------------------------|------------------------------------------------------------------------|
| EAE     | EventArcError          | Event: Arc Error                                                       |
| ECV     | EventControlledVoltage | Event: Voltage control                                                 |
| ECC     | EventControlledCurrent | Event: Current control                                                 |
| EEMCY   | EventEmergencyOff      | Event: Emergency off                                                   |
| EEOR    | EventEndOfRamp         | Event: End of ramp                                                     |
| EOn2Off | EventOnToOff           | Event: Change from state "On" to "Off" without ramp                    |
| EIER    | EventInputError        | Event: Input Error                                                     |
| EARC    | EventArc               | Event: Arc active or Regulation Error (Option - fast error evaluation) |

An event bit is permanently set if the status bit is 1 or is changing to 1. Different to the status bit an event bit isn't automatically reset. A reset has to be done by the user by writing an 1 to this event bit.

#### 4.5.1.4 Channel event mask (single write- and single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_1           | DATA_0 |
|-------------------------|----------|---------------|--------|------------------|--------|
| master write access     | 0        | 0x4003        | Mx     | ChannelEventMask |        |
| master read-            | 1        | 0x4003        | Mx     |                  |        |
| master single MBR read- | 1        | 0x6003        | Member | Offset           |        |
| HV board write access   | 0        | 0x4003/0x6003 | Mx     | ChannelEventMask |        |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... to access up to 255 channels

| ChannelEventMask |        |         | DATA_0 to DATA_1 |         |         |       |      | UI2  |      |      |       |          |        |       |      |
|------------------|--------|---------|------------------|---------|---------|-------|------|------|------|------|-------|----------|--------|-------|------|
| Bit15            | Bit14  | Bit13   | Bit12            | Bit11   | Bit10   | Bit9  | Bit8 | Bit7 | Bit6 | Bit5 | Bit4  | Bit3     | Bit2   | Bit1  | Bit0 |
| MEVLIM           | MECLIM | MECTRIP | MEEINH           | MEVBNDs | MECBNDs | MEARC | res  | MECV | MECC | res  | MEEOR | MEOn2Off | MEIERR | MEARC | res  |

The function of the ChannelEventMask register is described in 4.5.5.1 Channel events

|          |                            |                                                                                          |
|----------|----------------------------|------------------------------------------------------------------------------------------|
| MEVLIM   | MaskEventVoltageLimit      | EventMask: Hardware- voltage limit has been exceeded                                     |
| MECLIM   | MaskEventCurrentLimit      | EventMask: Hardware- current limit has been exceeded                                     |
| METRIP   | MaskEventTrip              | EventMask: Voltage limit or Current limit or Iset has been exceeded (when KillEnable=1 ) |
| MEEINH   | MaskEventExtInhibit        | EventMask: External Inhibit                                                              |
| MEVBNDs  | MaskEventVoltageBounds     | EventMask: Voltage out of bounds                                                         |
| MECBNDs  | MaskEventCurrentBounds     | EventMask: Current out of bounds                                                         |
| MEARC    | MaskEventArcError          | EventMask: Arc error                                                                     |
| MECV     | MaskEventControlledVoltage | EventMask: Voltage control                                                               |
| MECC     | MaskEventControlledCurrent | EventMask: Current control                                                               |
| MEEMCY   | MaskEventEmergencyOff      | EventMask: Emergency off                                                                 |
| MEEOR    | MaskEventEndOfRamp         | EventMask: End of ramp                                                                   |
| MEOn2Off | MaskEventOnToOff           | EventMask: Change from state on to off without ramp                                      |
| MEIER    | MaskEventInputError        | EventMask: Input Error                                                                   |
| MARC     | MaskEventArc               | EventMask: Arc active                                                                    |
|          | MaskEventRegulationError   | EventMask: Regulation Error (Option - fast error evaluation)                             |

Module in mode KILL disable:



If a bit of the ChannelEventStatus register is set to "1" and the corresponding bit in the ChannelEventMask register is "0", it is not necessary to clear the ChannelEventStatus bit to switch on HV again.

If a bit of the ChannelEventMask register is set to "1" and if the corresponding bit in the ChannelEventStatus is set to "1" by the module firmware then a reset of the corresponding ChannelEventStatus bits is necessary before a switch on the HV of this channel is possible again.

Module in mode KILL enable: A reset of the ChannelEventStatus bits is necessary before switch on the HV of this channel again.

#### 4.5.1.5 Delayed trip action (single/ multiple single read-write access)

EDCP frame:

| Access                   | DATA_DIR | DATA_ID | CHN    | DATA_0 | DATA_1 | DATA_2 |
|--------------------------|----------|---------|--------|--------|--------|--------|
| master write access      | 0        | 0x4006  | Mx     | Action |        |        |
| master read -            | 1        | 0x4006  | Mx     |        |        |        |
| master single MBR write- | 1        | 0x6006  | Member |        | Offset | Action |
| HV board write access    | 0        | 0x4006  | Mx     | Action |        |        |

Mx Channel 0 ... 255

Member Members 1 ... 16

Offset Channel member offset 0, 16, 32 ... access up to 255 channels

Action Action if a trip event will be initiated

0 – no action

1 – ramp down high voltage of the channel

2 – shut down high voltage of the channel without ramp

3 – shut down the whole module without ramp

4 – switch off the delayed trip function

#### 4.5.1.6 Delayed trip time

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_1       | DATA_0 |
|-------------------------|----------|---------------|--------|--------------|--------|
| master single read-     | 1        | 0x4005        | Mx     |              |        |
| master single MBR read- | 1        | 0x6005        | Member |              | Offset |
| HV board write access   | 0        | 0x4005/0x6005 | Mx     | Timeout-time |        |

Mx Channel 0 ... 255

Member Members 1 ... 16

Offset Channel member offset 0, 16, 32 ... to access up to 255 channels

Timeout-time DATA\_0 to DATA\_1[ms] UI2 (Range 1 to 4095 ms)

Time in milliseconds until delayed trip action becomes active and channel is in current control state. Note special functionality for modules with a second low current range – see manual “Delayed trip EHS.pdf”.

#### 4.5.1.7 External channel inhibit

EDCP frame:

| Access                   | DATA_DIR | DATA_ID       | CHN    | DATA_0 | DATA_1 | DATA_2 |
|--------------------------|----------|---------------|--------|--------|--------|--------|
| master write access      | 0        | 0x4007        | Mx     | Action |        |        |
| master read -            | 1        | 0x4007        | Mx     |        |        |        |
| master single MBR write- | 1        | 0x6007        | Member |        | Offset | Action |
| HV board write access    | 0        | 0x4007/0x6007 | Mx     | Action |        |        |

Mx Channel 0 ... 255

Member Members 1 ... 16

Offset Channel member offset 0, 16, 32 ... access up to 255 channels

Action Action if an inhibit signal will be triggered

0 – no action

1 – ramp down high voltage of the channel

2 – shut down high voltage of the channel without ramp

3 – shut down the whole module without ramp

4 – switch off the external inhibit function

#### 4.5.1.8 Set voltage (single write- and single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3     | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|------------|--------|--------|--------|
| master write access     | 0        | 0x4100        | Mx     | VoltageSet |        |        |        |
| master read -           | 1        | 0x4100        | Mx     |            |        |        |        |
| master single MBR read- | 1        | 0x6100        | Member |            | Offset |        |        |
| HV board write access   | 0        | 0x4100/0x6100 | Mx     | VoltageSet |        |        |        |

|            |                       |                                            |
|------------|-----------------------|--------------------------------------------|
| Mx         | Channel               | 0 ... 255                                  |
| Member     | Members               | 1 ... 16                                   |
| Offset     | Channel member offset | 0, 16, 32 ... to access up to 255 channels |
| VoltageSet | DATA_0 to DATA_3 [V]  | R4                                         |

The VoltageSet value is the preset for voltage generation. Allowed values are between 0 and the actual hardware limit value. If written values are between the hardware limit and the nominal value, then the module reduces these values to the value of the actual hardware limit. If written values are higher than the nominal data or lower than 0 an input error is indicated by setting the bit InputError.

If the channel is switched 'ON' then the voltage will be ramped to the set value after the receipt of this access. Otherwise the set value will just be stored and only used for ramping to the set voltage after the channel will be switched 'ON'.

#### 4.5.1.9 Set current / trip (single write- and single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3                   | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|--------------------------|--------|--------|--------|
| master write access     | 0        | 0x4101        | Mx     | CurrentSet (CurrentTrip) |        |        |        |
| master read -           | 1        | 0x4101        | Mx     |                          |        |        |        |
| master single MBR read- | 1        | 0x6101        | Member |                          | Offset |        |        |
| HV board write access   | 0        | 0x4101/0x6101 | Mx     | CurrentSet (CurrentTrip) |        |        |        |

|                          |                       |                                            |
|--------------------------|-----------------------|--------------------------------------------|
| Mx                       | Channel               | 0 ... 255                                  |
| Member                   | Members               | 1 ... 16                                   |
| Offset                   | Channel member offset | 0, 16, 32 ... to access up to 255 channels |
| CurrentSet (CurrentTrip) | DATA_0 to DATA_3 [A]  | R4                                         |

Allowed values are between 0 and the actual hardware limit value. If written values are between the hardware limit and the nominal value, then the module reduces these values to the value of the actual hardware limit. If written values are higher than the nominal data or lower than 0 an input error is indicated by setting the bit InputError.

The mode of action of this item depends on the setting of the signal Kill Enable (KILEna) in the ModuleControl register (4.5.2.2). If Kill Enable is 0, the value is interpreted as CurrentSet. If Kill Enable is 1, the value is CurrentTrip.

CurrentSet:

The CurrentSet value is the preset for current regulation. If the output current reaches or exceeds the Current Set value, the channel goes into Current Regulation mode. In this mode the output current is regulated at the CurrentSet value, but the output voltage is going to a value between 0V and Vset, depending of the external load.

When Current Control mode is active the bit isCC of the ChannelStatus register and the bit EventControlledCurrent of the ChannelEventStatus are set, the bit isCV of the ChannelStatus is reset.

CurrentTrip:

In Current Trip mode this value will be used as software current trip. If exceeding this value a current trip event will be registered. The green LED on front panel will be switched off.

The bits isTrip in the ChannelStatus and ETRP in ChannelEventStatus are set, the bit isNoSumError in the ModuleStatus is reset.

#### 4.5.1.10 Voltage measurement (single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3         | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|----------------|--------|--------|--------|
| master read -           | 1        | 0x4102        | Mx     |                |        |        |        |
| master single MBR read- | 1        | 0x6102        | Member |                | Offset |        |        |
| HV board write access   | 0        | 0x4102/0x6102 | Mx     | VoltageMeasure |        |        |        |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... access up to 255 channels  
VoltageMeasure DATA\_0 to DATA\_3 [V] R4

#### 4.5.1.11 Current measurement (single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3         | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|----------------|--------|--------|--------|
| master read -           | 1        | 0x4103        | Mx     |                |        |        |        |
| master single MBR read- | 1        | 0x6103        | Member |                | Offset |        |        |
| HV board write access   | 0        | 0x4103/0x6103 | Mx     | CurrentMeasure |        |        |        |

Mx Channel 0 ... 255  
MBR Members 1 ... 16  
OFFSET Channel member offset 0, 16, 32 ... access up to 255 channels  
CurrentMeasure DATA\_0 to DATA\_3 [A] R4

#### 4.5.1.12 Voltage bounds (single write- / single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3        | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|---------------|--------|--------|--------|
| master write access     | 0        | 0x4104        | Mx     | VoltageBounds |        |        |        |
| master read -           | 1        | 0x4104        | Mx     |               |        |        |        |
| master single MBR read- | 1        | 0x6104        | Member |               | Offset |        |        |
| HV board write access   | 0        | 0x4104/0x6104 | Mx     | VoltageBounds |        |        |        |

Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... access up to 255 channels  
VoltageBounds DATA\_0 to DATA\_3 [V] R4  
(0 to VoltageNominal)

#### 4.5.1.13 Current bounds (single write- / single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3        | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|---------------|--------|--------|--------|
| master write access     | 0        | 0x4105        | Mx     | CurrentBounds |        |        |        |
| master read -           | 1        | 0x4105        | Mx     |               |        |        |        |
| master single MBR read- | 1        | 0x6105        | Member |               | Offset |        |        |
| HV board write access   | 0        | 0x4105/0x6105 | Mx     | CurrentBounds |        |        |        |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... access up to 255 channels  
CurrentBounds DATA\_0 to DATA\_3 [A] R4  
(0 to CurrentNominal)

#### 4.5.1.14 Nominal voltage (single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3         | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|----------------|--------|--------|--------|
| master read -           | 1        | 0x4106        | Mx     |                |        |        |        |
| master single MBR read- | 1        | 0x6106        | Member |                | Offset |        |        |
| HV board write access   | 0        | 0x4106/0x6106 | Mx     | VoltageNominal |        |        |        |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... access up to 255 channels  
VoltageNominal DATA\_0 to DATA\_3 [V] R4

#### 4.5.1.15 Nominal current (single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3         | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|----------------|--------|--------|--------|
| master read -           | 1        | 0x4107        | Mx     |                |        |        |        |
| master single MBR read- | 1        | 0x6107        | Member |                | Offset |        |        |
| HV board write access   | 0        | 0x4107/0x6107 | Mx     | CurrentNominal |        |        |        |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... access up to 255 channels  
CurrentNominal DATA\_0 to DATA\_3 [A] R4

#### 4.5.1.16 Current measurement range\* (single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_4         | DATA_3 | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|----------------|--------|--------|--------|--------|
| master read -           | 1        | 0x4109        | Mx     |                |        |        |        |        |
| master single MBR read- | 1        | 0x6109        | Member |                | Offset |        |        |        |
| HV board write access   | 0        | 0x4109/0x6103 | Mx     | CurrentMeasure |        |        |        | Range  |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... access up to 255 channels  
CurrentMeasure DATA\_1 to DATA\_4 [A] R4  
Range DATA\_0=0 – high range ( $\geq 20\mu\text{A}$ ) UI1  
DATA\_0=1 – low range ( $< 20\mu\text{A}$ )

(\* for device class 26, 27 and 39, E08F2, E08C2 and N06C2 only)

**NOTE** The information channel status bit isLowCurrentRange can be used also.

#### 4.5.1.17 VCT Coefficient\* (single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3         | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|----------------|--------|--------|--------|
| master read -           | 1        | 0x4120        | Mx     |                |        |        |        |
| master single MBR read- | 1        | 0x6120        | Member |                | Offset |        |        |
| HV board write access   | 0        | 0x4120/0x6120 | Mx     | VctCoefficient |        |        |        |

Mx Channel 0 ... 255  
Member Members 1 ... 16  
Offset Channel member offset 0, 16, 32 ... access up to 255 channels

VctCoefficient

DATA\_0 to DATA\_3 [V/K]

R4

(\* Option VCT only)

#### 4.5.1.18 Temperature external\* (single/ multiple single read-write access)

EDCP frame:

| Access                  | DATA_DIR | DATA_ID       | CHN    | DATA_3              | DATA_2 | DATA_1 | DATA_0 |
|-------------------------|----------|---------------|--------|---------------------|--------|--------|--------|
| master read -           | 1        | 0x4121        | Mx     |                     |        |        |        |
| master single MBR read- | 1        | 0x6121        | Member |                     | Offset |        |        |
| HV board write access   | 0        | 0x4121/0x6121 | Mx     | TemperatureExternal |        |        |        |

Mx Channel 0 ... 255

Member Members 1 ... 16

Offset Channel member offset 0, 16, 32 ... access up to 255 channels

TemperatureExternal DATA\_0 to DATA\_3 [°C] R4

(\* Option VCT only)

#### 4.5.1.19 Group number (single/ multiple single read-write access)

EDCP frame:

|                          |          |         |        |        |        |       |
|--------------------------|----------|---------|--------|--------|--------|-------|
| Access                   | DATA_DIR | DATA_ID | CHN    | DATA_0 |        |       |
| master write access      | 0        | 0x4200  | Mx     | GROUP  |        |       |
| master read -            | 1        | 0x4200  | Mx     |        |        |       |
| master single MBR write- | 1        | 0x6200  | Member |        | Offset | Group |
| HV board write access    | 0        | 0x4200  | Mx     | GROUP  |        |       |

Mx Channel 0 ... 255

Member Members 1 ... 16

Offset Channel member offset 0, 16, 32 ... access up to 255 channels

Group Group number of the channel members 0 .. 255

With a group number GROUP for each channel, channels can be combined to groups involving all connected modules. The [NMT channel group set](#) and the NMT module set frames (described on page 19) send broadcast information for all channels, which have the same group number.

## 4.5.2 EDCP Module Accesses

### 4.5.2.1 ModuleStatus (module read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_1       | DATA_0 |
|-----------------------|----------|---------|--------------|--------|
| master read-          | 1        | 0x1000  |              |        |
| HV board write access | 0        | 0x1000  | ModuleStatus |        |

ModuleStatus DATA\_0 to DATA\_1 UI2

| Bit15   | Bit14   | Bit13    | Bit12   | Bit11    | Bit10    | Bit9     | Bit8       | Bit7 | Bit6    | Bit5     | Bit4    | Bit3   | Bit2    | Bit1 | Bit0  |
|---------|---------|----------|---------|----------|----------|----------|------------|------|---------|----------|---------|--------|---------|------|-------|
| isKillE | isTmpGd | isSplyGd | isModGd | isEvtAct | isSflpGd | isNoRamp | isNoSumErr | res  | isInErr | isHwVIGd | needSrv | isHvOn | isLvIns | res  | isAdj |

The status bits as there are IsTemperatureGood, IsSupplyGood, IsModuleGood, IsEventActive, IsSafetyLoopGood, IsNoRamp and IsNoSumError indicate the single status for the complete module.

The status bit IsCommandComplete indicates whether all CAN commands given to the module have been executed.

|            |                            |                                                                                             |
|------------|----------------------------|---------------------------------------------------------------------------------------------|
| isKillE    | IsKillEnable               | Module state of kill enable                                                                 |
| isTmpGd    | IsTemperatureGood          | Module temperature good                                                                     |
| isSplyGd   | IsSupplyGood               | Power supply good                                                                           |
| isModGd    | IsModuleGood               | Module in state good                                                                        |
| isEvtAct   | IsEventActive              | Any event is active and mask is set                                                         |
| isSflpGd   | IsSafetyLoopGood           | Safety loop closed                                                                          |
| isNoRamp   | IsNoRamp                   | All channels stable, no ramp active                                                         |
| isNoSumErr | IsNoSumError               | All channels without failure                                                                |
| isInErr    | IsInputError               | Input error in connection with a module access                                              |
| isHwVIGd   | IsHardwareVoltageLimitGood | Hardware voltage limit in proper range, for HV distributor modules with current mirror only |
| needSrv    | IsServiceNeeded            | Hardware failure detected (consult iseg Spezialelektronik GmbH)                             |
| isHvOn     | IsHighVoltageOn            | At least one channel generates a high voltage                                               |
| isLvIns    | IsLiveInsertion            | Mode live insertion                                                                         |
| isAdj      | IsFineAdjustment           | Mode of the fine adjustment                                                                 |
| res        | Reserved                   |                                                                                             |

|             |                                                                                                                 |              |                                                                                                                                                                                                                      |
|-------------|-----------------------------------------------------------------------------------------------------------------|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| isKillEna=0 | Module in state kill disable                                                                                    | isSflpGd=0   | safety loop is broken -VO has been shut off,                                                                                                                                                                         |
| isKillEna=1 | Module in state kill enable                                                                                     | isSflpGd=1   | safety loop is closed                                                                                                                                                                                                |
| isTmpGd=0   | If module temperature is higher than 55°C then all channel are switched off permanently                         | isNoRamp=0   | VO is ramping in at least one channel                                                                                                                                                                                |
| isTmpGd=1   | module temperature is within working range                                                                      | isNoRamp=1   | no channel is ramping                                                                                                                                                                                                |
| isSplyGd=0  | supply voltages are out of range (range of 24V +/-10% and of 5V +/-5%)                                          | isNoSumErr=0 | voltage limit, current limit, trip, voltage bound or current bound has been exceeded in at least one of the channels or external INHIBIT<br>→ error, reset by reset of the corresponding flag of the channel status' |
| isSplyGd=1  | supply voltages are within range                                                                                | isnoSumERR=1 | evaluation of the 'Channel Status' over all channels to a sum error flag<br>→ LIM&CLIM&CTRP&EINH&VBND&CBND=0<br>→ no errors                                                                                          |
| isModGd=0   | module is not good, that means (isnoSERR AND (ETMPngd OR ESPLYngd OR ESFLPngd))==0                              | isInErr=1    | input error in connection with a module access                                                                                                                                                                       |
| isModGd=1   | module is good, that means (isnoSERR AND NOT(ETMPngd OR ESPLYngd OR ESFLPngd))==1(see module event status also) | isInErr=0    | no input error in connection with a module access                                                                                                                                                                    |
| isEvtAct=0  | no Event is active                                                                                              |              |                                                                                                                                                                                                                      |
| isEvtAct=1  | any Event is active                                                                                             |              |                                                                                                                                                                                                                      |

|            |                                                       |           |                                          |
|------------|-------------------------------------------------------|-----------|------------------------------------------|
| isHwVIGd=0 | hardware voltage limit not in proper range            |           | Modules with 7 digit serial number only. |
| isHwVIGd=1 | hardware voltage limit in proper range                | isLvIns=0 | no Live Insertion mode                   |
| needSrv=0  | Module is ready for working.                          | isLvIns=1 | Live Insertion mode                      |
| needSrv=1  | Module need a service.                                | isAdj=0   | Fine adjustment is off.                  |
| isHvOn=0   | No high voltage will be generated                     | isAdj=1   | Fine adjustment is on (default).         |
| isHvOn=1   | At least one channel generates a high voltage output. |           |                                          |

#### 4.5.2.2 ModuleControl (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_1        | DATA_0 |
|-----------------------|----------|---------|---------------|--------|
| master write          | 0        | 0x1001  | ModuleControl |        |
| master read-          | 1        | 0x1001  |               |        |
| HV board write access | 0        | 0x1001  | ModuleControl |        |

ModuleControl DATA\_0 to DATA\_1 UI2

| Bit15 | Bit14     | Bit13 | Bit12  | Bit11   | Bit10 | Bit9 | Bit8 | Bit7 | Bit6    | Bit5   | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-----------|-------|--------|---------|-------|------|------|------|---------|--------|------|------|------|------|------|
| res   | setKILena | res   | setADJ | setENDN | res   | res  | res  | res  | doCLEAR | setILK | res  | res  | res  | res  | res  |

|           |            |                                                                                      |
|-----------|------------|--------------------------------------------------------------------------------------|
| setKILena | KillEnable | Kill function                                                                        |
| setADJ    | Adjust     | Switch ON of fine adjustment                                                         |
| setENDN   | Endian     | Order of bytes in word: 0 = Little Endian (INTEL); 1 = Big Endian (MOTOROLA)         |
| doCLEAR   | ClearKill  | Hardware ClearKill signal and clear all event signals of the module and the channels |
| setILK    | Interlock  | Interlock signal CRATE_ENABLE (WIENER MPOD crate, active TTL high)                   |
| res       | Reserved   |                                                                                      |

|           |                                                                                      |           |                                                                                                  |
|-----------|--------------------------------------------------------------------------------------|-----------|--------------------------------------------------------------------------------------------------|
| setKILL=0 | kill function disable                                                                | doCLEAR=0 | no action                                                                                        |
| setKILL=1 | kill function enable                                                                 |           |                                                                                                  |
| setADJ=0  | fine adjustment OFF                                                                  | setILK= 1 | INTERLOCK signal in order to switch off HV and set bit EEINH of the EventStatus for all channels |
| setADJ=1  | fine adjustment ON                                                                   |           |                                                                                                  |
| setENDN=1 | big endian (MOTOROLA format)                                                         | setILK=0  | reset the INTERLOCK signal in order to clear the bit EEINH of the EventStatus for all channels   |
| doCLEAR=1 | Hardware ClearKill signal and clear all event signals of the module and the channels |           |                                                                                                  |

#### 4.5.2.3 ModuleEventStatus (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_1            | DATA_0 |
|-----------------------|----------|---------|-------------------|--------|
| master write          | 0        | 0x1002  | ModuleEventStatus |        |
| master read-          | 1        | 0x1002  |                   |        |
| HV board write access | 0        | 0x1002  | ModuleEventStatus |        |

ModuleEventStatus DATA\_0 to DATA\_1 UI2

| Bit15 | Bit14   | Bit13    | Bit12 | Bit11 | Bit10    | Bit9 | Bit8 | Bit7 | Bit6  | Bit5     | Bit4 | Bit3 | Bit2   | Bit1 | Bit0 |
|-------|---------|----------|-------|-------|----------|------|------|------|-------|----------|------|------|--------|------|------|
| res   | ETMPngd | ESPLYngd | res   | res   | ESFLPngd | res  | res  | res  | EIERR | EHwVLngd | ESrv | res  | ELVINS | res  | res  |

|          |                                  |                                                                               |
|----------|----------------------------------|-------------------------------------------------------------------------------|
| ETMPngd  | EventTemperatureNotGood          | Event: Temperature is above 55°C                                              |
| ESPLYngd | EventSupplyNotGood               | Event: at least one of the supplies is not good                               |
| ESFLPngd | EventSafetyLoopNotGood           | Event: Safety loop is open                                                    |
| EIERR    | EventInputError                  | Event: input error in connection with a module access                         |
| EHwVLngd | EventHardwareVoltageLimitNotGood | Event: Hardware voltage limit is not in proper range, only for HV distributor |

|        |                    |                                                                                                                                                                                 |
|--------|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|        |                    | modules with current mirror;                                                                                                                                                    |
| ESrvs  | EventService       | Event: A hardware failure of the HV module has been detected. The HV will switched off without a possibility to switch on again. Please consult the iseg Spzialelektronik GmbH. |
| ELVINS | EventLiveInsertion | Event live insertion to prepare a hot plug of a module                                                                                                                          |
| res    | Reserved           |                                                                                                                                                                                 |

#### 4.5.2.4 ModuleEventMask (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_1          | DATA_0 |
|-----------------------|----------|---------|-----------------|--------|
| master write          | 0        | 0x1003  | ModuleEventMask |        |
| master read-          | 1        | 0x1003  |                 |        |
| HV board write access | 0        | 0x1003  | ModuleEventMask |        |

ModuleEventMask DATA\_0 to DATA\_1 UI2

| Bit15 | Bit14    | Bit13     | Bit12 | Bit11 | Bit10     | Bit9 | Bit8 | Bit7 | Bit6   | Bit5      | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|----------|-----------|-------|-------|-----------|------|------|------|--------|-----------|------|------|------|------|------|
| res   | METMPngd | MESPLYngd | res   | res   | MESFLPngd | res  | res  | res  | MEIERR | MEHwVLngd | res  | res  | res  | res  | res  |

|           |                                      |                                                                                                                  |
|-----------|--------------------------------------|------------------------------------------------------------------------------------------------------------------|
| METMPngd  | MaskEventTemperatureNotGood          | MEEventMask: Temperature is above 55°C                                                                           |
| MESPLYngd | MaskEventSupplyNotGood               | MEEventMask: at least one of the supplies is not good                                                            |
| MESFLPngd | MaskEventSafetyLoopNotGood           | MEEventMask: Safety loop (SL) is open                                                                            |
| MEIERR    | MaskEventInputError                  | MEEventMask: Input error in connection with a module access                                                      |
| MEHwVLngd | MaskEventHardwareVoltageLimitNotGood | MEEventMask: Hardware voltage limit is not in proper range, only for HV distributor modules with current mirror; |
| res       | Reserved                             |                                                                                                                  |

All bits of the EventMask register are set to "0" after the power on reset.

Module in mode KILL enable: If a bit of the EventStatus register is set to "1" and the corresponding bit in the EventMask register is "0" no reset of the EventStatus bits is necessary before switch on the HV of any channel again.

If a bit of the EventMask register is set to "1" and if the corresponding bit in the EventStatus is set to "1" by the module firmware a reset of the corresponding EventStatus bits is necessary before a switch on the HV of any channel is possible.

Module in mode KILL enable: A reset of the EventStatus bits is necessary before switch on the HV of any channel is possible.

#### 4.5.2.5 ModuleEventChannelStatus (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_2 | DATA_1                   | DATA_0 |
|-----------------------|----------|---------|--------|--------------------------|--------|
| master write access   | 0        | 0x1004  | OFFSET | ModuleEventChannelStatus |        |
| master read-          | 1        | 0x1004  | OFFSET |                          |        |
| HV board write access | 0        | 0x1004  | OFFSET | ModuleEventChannelStatus |        |

OFFSET DATA\_2Channel member offset 0, 16, 32 ... access up to 255 channels EventChannelStatus DATA\_0 to DATA\_1 UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| CH15  | CH14  | CH13  | CH12  | CH11  | CH10  | CH9  | CH8  | CH7  | CH6  | CH5  | CH4  | CH3  | CH2  | CH1  | CH0  |

The n-th bit of the register is set, if an event is active in the n-th channel and the associated bit in the EventMask register of the n-th channel is set too.

$$CHn = EventStatus[n] \& EventMask[n]$$

Reset of a bit is done by writing a 1 to this bit.

#### 4.5.2.6 ModuleEventChannelMask (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_2 | DATA_1                 | DATA_0 |
|-----------------------|----------|---------|--------|------------------------|--------|
| master write access   | 0        | 0x1005  | OFFSET | ModuleEventChannelMask |        |
| master read-          | 1        | 0x1005  | OFFSET |                        |        |
| HV board write access | 0        | 0x1005  | OFFSET | ModuleEventChannelMask |        |

OFFSET DATA\_2Channel member offset 0, 16, 32 ... access up to 255 channels

EventChannelMask DATA\_0 to DATA\_1 UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| CH15  | CH14  | CH13  | CH12  | CH11  | CH10  | CH9  | CH8  | CH7  | CH6  | CH5  | CH4  | CH3  | CH2  | CH1  | CH0  |

This register decides whether a pending event leads to the sum event flag of the module or not. If the n-th bit of the mask is set and the n-th channel has an active event in the ModuleEventChannelStatus the bit isEventActive in the ModuleStatus register is set

#### 4.5.2.7 ModuleEventGroupStatus (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3                 | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|------------------------|--------|--------|--------|
| master write access   | 0        | 0x1005  | ModuleEventGroupStatus |        |        |        |
| master read-          | 1        | 0x1005  |                        |        |        |        |
| HV board write access | 0        | 0x1005  | ModuleEventGroupStatus |        |        |        |

EventGroupStatus DATA\_0 to DATA\_3 UI4

| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| GR31  | GR30  | GR29  | GR28  | GR27  | GR26  | GR25  | GR24  | GR23  | GR22  | GR21  | GR20  | GR19  | GR18  | GR17  | GR16  |

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| GR15  | GR14  | GR13  | GR12  | GR11  | GR10  | GR9  | GR8  | GR7  | GR6  | GR5  | GR4  | GR3  | GR2  | GR1  | GR0  |

The n-th bit of this double word register is set, if an event is active in the n-th group.

Reset of a bit is done by writing a 1 to this bit.

#### 4.5.2.8 ModuleEventGroupMask (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3               | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|----------------------|--------|--------|--------|
| master write access   | 0        | 0x1006  | ModuleEventGroupMask |        |        |        |
| master read-          | 1        | 0x1006  |                      |        |        |        |
| HV board write access | 0        | 0x1006  | ModuleEventGroupMask |        |        |        |

EventGroupMask DATA\_0 to DATA\_3 UI4

| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| GR31  | GR30  | GR29  | GR28  | GR27  | GR26  | GR25  | GR24  | GR23  | GR22  | GR21  | GR20  | GR19  | GR18  | GR17  | GR16  |

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| GR15  | GR14  | GR13  | GR12  | GR11  | GR10  | GR9  | GR8  | GR7  | GR6  | GR5  | GR4  | GR3  | GR2  | GR1  | GR0  |

This register decides whether a pending event leads to the sum event flag of the module or not. If the n-th bit of the mask is set and the n-th group has an active event in the ModuleEventGroupStatus the bit isEventActive in the ModuleStatus register is set.

#### 4.5.2.9 VoltageRampSpeed (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3           | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|------------------|--------|--------|--------|
| master write access   | 0        | 0x1100  | VoltageRampSpeed |        |        |        |
| master read -         | 1        | 0x1100  |                  |        |        |        |
| HV board write access | 0        | 0x1100  | VoltageRampSpeed |        |        |        |

VoltageRampSpeed DATA\_0 to DATA\_3 [%] R4

Voltage ramp speed range (disable Kill):  $1\text{mV/s} \leq \text{Ramp speed} \leq 20\%$  of  $V_{O\text{ max/s}}$

Option: fast ramp

|   |                                                                   |
|---|-------------------------------------------------------------------|
| 1 | $1\text{mV/s} \leq \text{Ramp speed} \leq 25\%$ of VoltageNominal |
| 2 | $1\text{mV/s} \leq \text{Ramp speed} \leq 50\%$ of VoltageNominal |
| 3 | $1\text{mV/s} \leq \text{Ramp speed} \leq 75\%$ of VoltageNominal |

Voltage ramp speed range (enable Kill):  $1\text{mV/s} \leq \text{Ramp speed} \leq 1\%$  of  $V_{O\text{ max/s}}$

The speed of the voltage ramp in percent of the nominal voltage of the channel per second.

#### 4.5.2.10 CurrentRampSpeed – current controlled modules only (module write- / read-write access)

EDCP frame

| Access                | DATA_DIR | DATA_ID | DATA_3           | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|------------------|--------|--------|--------|
| master write access   | 0        | 0x1101  | CurrentRampSpeed |        |        |        |
| master read -         | 1        | 0x1101  |                  |        |        |        |
| HV board write access | 0        | 0x1101  | CurrentRampSpeed |        |        |        |

CurrentRampSpeed DATA\_0 to DATA\_3 [%] R4

Current ramp speed range:  $2\% \text{ IO max/s} \leq \text{Ramp speed} \leq \text{IO max/s}$

The speed of the current ramp in percent of the nominal current of the channel per second.

#### 4.5.2.11 VoltageMax – OPTION (module read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3     | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|------------|--------|--------|--------|
| master read -         | 1        | 0x1102  |            |        |        |        |
| HV board write access | 0        | 0x1102  | VoltageMax |        |        |        |

HardwareVoltageLimit DATA\_0 to DATA\_3 [%] R4

HV Modules with the OPTION hardware voltage limit can adjust  $V_{O\text{ max}}$  via the potentiometer  $V_{\text{max}}$ .

For HV Modules without this OPTION VoltageMax equals to  $V_{O\text{ max}}$ .

The exceeding of the hardware voltage limit results in a limitation of the voltage when the KILL-enable.

The absolute value of the hardware voltage limit will compute by following:

$$\text{Voltage limit of the channel } x (\text{Chx}) = \text{VoltageNominal}[\text{Chx}] * \text{VoltageMax}$$

The module responds after the hardware voltage limit has been exceeded:

The green LED on front panel is off.

Depends of the kind of module:

Hardware KILL function controlled by the bit 'KILena' of the ModuleControl word:

KILL-enable = 1: The voltage will be switched off permanently without ramp. ChannelEventStatus flag 'EVLIM' will be set.

KILL-enable = 0: The voltage will be reduced to the value of the actual hardware voltage limit. ChannelStatus flag 'isVLIM' and ChannelEventStatus flag 'EVLIM' will be set.

Software KILL function controlled by the bit 'KILena' of the ChannelControl word:

- KILL-enable = 1: Voltage will be switched off permanently without ramp. ChannelEventStatus flag 'EVLIM' will be set.
- KILL-enable = 0: Voltage will be switched off without ramp. If the output voltage arrives at 0 V the ramping to set voltage will be restarted automatically. ChannelStatus flag 'isVLIM' and ChannelEventStatus flag 'EVLIM' will be set.

#### 4.5.2.12 CurrentMax – OPTION (module read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3     | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|------------|--------|--------|--------|
| master read -         | 1        | 0x1103  |            |        |        |        |
| HV board write access | 0        | 0x1103  | CurrentMax |        |        |        |

HardwareCurrentLimit DATA\_0 to DATA\_3 [%] R4

HV Modules with the OPTION CurrentMax can adjust the IO max via the potentiometer I<sub>max</sub>.

HV Modules without this OPTION deliver IO max.

The absolute value of the hardware current limit will compute by following:

$$\text{Current limit of the channel } x \text{ (Chx)} = \text{CurrentNominal[Chx]} * \text{CurrentMax}$$

The module responds after the hardware current limit has been exceeded:

The green LED on front panel is off.

Depends of the kind of module:

Hardware KILL function controlled by the bit 'KILena' of the ModuleControl word:

- KILL-enable = 1: Voltage will be switched off permanently without ramp. ChannelEventStatus flag 'ECLIM' will be set.
- KILL-enable = 0: Current will be reduced to the value of the actual hardware current limit. ChannelStatus flag 'isCLIM' and ChannelEventStatus flag ECLIM will be set.

Software KILL function controlled by the bit 'KILena' of the ChannelControl word:

- KILL-enable = 1: Voltage will be switched off permanently without ramp. ChannelEventStatus flag ECLIM will be set.
- KILL-enable = 0: Voltage will be switched off without ramp. If the output voltage arrives at 0 V the ramping to set voltage will be restarted automatically. ChannelStatus flag 'isCLIM' and ChannelEventStatus flag ECLIM will be set.

#### 4.5.2.13 Supply24 (module read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3   | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|----------|--------|--------|--------|
| master read -         | 1        | 0x1104  |          |        |        |        |
| HV board write access | 0        | 0x1104  | Supply24 |        |        |        |

Supply24 DATA\_0 to DATA\_3 [V] R4

An 'out of range error' (see DCP group access: General status) will be generated if deviation of voltage is more than  $\pm 10\%$ .

#### 4.5.2.14 Supply5 (module read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3  | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|---------|--------|--------|--------|
| master read -         | 1        | 0x1105  |         |        |        |        |
| HV board write access | 0        | 0x1105  | Supply5 |        |        |        |

Supply5 DATA\_0 to DATA\_3 [V] R4

An 'out of range error' (see DCP group access: General status) will be generated if deviation of voltage is more than  $\pm 5\%$ .

#### 4.5.2.15 BoardTemperature (module read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3                | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|-----------------------|--------|--------|--------|
| master read -         | 1        | 0x1106  |                       |        |        |        |
| HV board write access | 0        | 0x1106  | BoardTemperature      |        |        |        |
| BoardTemperature      |          |         | DATA_0 to DATA_3 [°C] |        | R4     |        |

An 'out of range error' (see group access: General status) will be generated if the temperature is higher than +55°C.

#### 4.5.2.16 Threshold to arm the errors detection (module write / read- write access)

EDCP frame:

| Access                     | DATA_DIR | DATA_ID | DATA_3                     | DATA_2 | DATA_1 | DATA_0 |
|----------------------------|----------|---------|----------------------------|--------|--------|--------|
| master write access        | 0        | 0x1107  | ThresholdArmErrorDetection |        |        |        |
| master read -              | 1        | 0x1107  |                            |        |        |        |
| HV board write access      | 0        | 0x1107  | ThresholdArmErrorDetection |        |        |        |
| ThresholdArmErrorDetection |          |         | DATA_0 to DATA_3 [%]       |        | R4     |        |

Factory setting for different kinds of HV modules is between 1V and to VO max/10 in percent to the nominal voltage of the channel.

The arming of the error detection is started while the actual voltage exceeds these value which has been stored before.

Exception: At the start of a ramp from zero the firmware evaluates that the feedback control will look in. If not, because the channel has a short or the hardware current limit is near to zero, then the channel will be switched off and a current error will be generated before the actual voltage is exceeding these threshold.

#### 4.5.2.17 Serial number (module read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3           | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|------------------|--------|--------|--------|
| master read -         | 1        | 0x1200  |                  |        |        |        |
| HV board write access | 0        | 0x1200  | SerialNumber     |        |        |        |
| SerialNumber          |          |         | DATA_0 to DATA_3 |        | UI4    |        |

serial number e.g. 471212

#### 4.5.2.18 Firmware release (module read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3           | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|------------------|--------|--------|--------|
| master read -         | 1        | 0x1201  |                  |        |        |        |
| HV board write access | 0        | 0x1201  | FirmwareRelease  |        |        |        |
| FirmwareRelease       |          |         | DATA_0 to DATA_3 |        | UI1[4] |        |

release e.g. 01.00.00.00

#### 4.5.2.19 Bit rate (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID                   | DATA_1  | DATA_0 |
|-----------------------|----------|---------------------------|---------|--------|
| master read-          | 1        | 0x1202                    |         |        |
| HV board write access | 0        | 0x1202                    | BitRate |        |
| BitRate               |          | DATA_0 to DATA_1 [kbit/s] |         | UI2    |

Following bit rates are possible: 20, 50, 100, 125, 250 kbit/s

The new bit rate gets active after RESET or POWER OFF/ON. The bit rate of all modules in the system must be the same before a RESET or POWER/ON is made.



#### 4.5.2.22 DigitalFilter (module write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_1        | DATA_0 |
|-----------------------|----------|---------|---------------|--------|
| master write          | 0        | 0x1205  | NumberOfSteps |        |
| master read-          | 1        | 0x1205  |               |        |
| HV board write access | 0        | 0x1205  | NumberOfSteps |        |

NumberOfSteps DATA\_0 to DATA\_1 [Steps] UI2 (possible steps are 1, 16, 64, 256, 512 and 1024)

The digital filter in the firmware of the processor reduces the white noise of the analog values of channel VoltageMeasure, channel CurrentMeasure. The digital filtering gives the possibility to get a higher precision and to react fast on changes of the measured values. The filter is not used during a voltage ramp. The filter is restarted after a significant change of the signal.

Factory settings: 64

#### 4.5.2.23 ModuleOption (module read access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3       | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|--------------|--------|--------|--------|
| master read -         | 1        | 0x1280  |              |        |        |        |
| HV board write access | 0        | 0x1280  | ModuleOption |        |        |        |

ModuleOption DATA\_0 to DATA\_3 UI4

The requested value of the module option is not valid when all bits are set to '1'!

| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23   | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|-------|-------|-------|-------|-------|-------|-------|-------|---------|-------|-------|-------|-------|-------|-------|-------|
| EDCP  | -     | -     | -     | -     | HVBPM | CLIM  | VLIM  | INHIBIT | RELAY | FRAMP | -     | -     | -     | -     | -     |

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| -     | -     | -     | -     | -     | -     | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    |

| BIT   | OPTION | DESCRIPTION SPECIFICATION        |
|-------|--------|----------------------------------|
| Bit31 | EDCP   | Enhanced Device Control Protocol |
| Bit26 | HVBM   | HV boards per (CAN nodes) module |
| Bit25 | CLIM   | hardware current limit           |
| Bit24 | VLIM   | hardware voltage limit           |
| Bit23 | INHIB  | external INHIBIT signals         |
| Bit22 | RELY   | discharge relay                  |
| Bit21 | FRMP   | fast ramp                        |

#### 4.5.2.24 ModuleOptionSpec (module read access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID          | DATA_4       | DATA_3 | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|------------------|--------------|--------|--------|--------|--------|
| master read -         | 1        | 0x1290           | DATA_4       | DATA_3 | DATA_2 | DATA_1 |        |
| HV board write access | 0        | 0x1290           | ModuleOption |        |        |        | Spec   |
| ModuleOption          |          | DATA_1 to DATA_4 |              |        |        |        | UI4    |
| Specification         |          | DATA_0           |              |        |        |        | UI1    |

The requested value of the module option specification is not valid or do not exist when all bits are set to '1' or '0'!

| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23   | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|-------|-------|-------|-------|-------|-------|-------|-------|---------|-------|-------|-------|-------|-------|-------|-------|
| EDCP  | -     | -     | -     | -     | HVBPM | CLIM  | VLIM  | INHIBIT | RELAY | FRAMP | -     | -     | -     | -     | -     |

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| -     | -     | -     | -     | -     | -     | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    |

To request a specification the corresponding bit of the module option word has to be set to '1'.

|                          |   |                       |
|--------------------------|---|-----------------------|
| Specification: fast ramp | 1 | 25% of VoltageNominal |
|                          | 2 | 50% of VoltageNominal |
|                          | 3 | 75% of VoltageNominal |

#### 4.5.2.25 ModuleCommMode (module write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID          | DATA_1         | DATA_0 |
|-----------------------|----------|------------------|----------------|--------|
| HV board write access | 0        | 0x12a0           | ModuleCommMode |        |
| ModuleOption          |          | DATA_0 to DATA_3 |                | UI2    |

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| -     | -     | -     | -     | -     | -     | -    | -    | -    | -    | -    | -    | -    | -    | -    | FAST |

Bit0 FAST 1 Fast communication mode (supported from isegHVOPCServer, isegCANHVControl)  
0 common mode

### 4.5.3 EDCP Group Accesses

The Multi Channel CAN module offers an extended and flexible range of group functions. There exist both predefined (so called fix) groups and variable groups.

Each group definition consists of 2 words each of 16 bits. In fix groups these 2 words are the value to be set into all channels (in float format) or they are a logical information. In variable groups one word carries the information about type and characteristics of the group, the other word carries the information about the members of the group or gives an overview about a selected situation in all channels.

Four different group types for variable groups have been established:

- Set group
- Status group
- Monitoring group
- Trip group

### 4.5.3.1 Set group

Set groups will be used in order to set channels to a same value, which happen to carry the identical channel value. Therefore within the group following will be defined:

- Member of the group: Each member will be activated in the channel setting list ChSetLst
- Type of the group: Set group type TypeSet
- Channel characteristics: Coding of characteristics, which have to be set commonly
- Control mode: Divides between a one-time setting of the slave channel property and a permanently copying of the Master channel's property to the slave channels
- Master channel: Number of the channel, which characteristics will be transferred to the other channels. Is just necessary for Set groups which set a value. If functions have to be initialized e.g. start of ramp then there is no Master channel

EDCP frame:

| Access                | DATA_DIR | DATA_ID | NBR | OFFSET | DATA_3   | DATA_2 | DATA_1  | DATA_0 |
|-----------------------|----------|---------|-----|--------|----------|--------|---------|--------|
| master group write    | 0        | 0x2000  | Nx  | 0x     | ChSetLst |        | TypeSet |        |
| master group read-    | 1        | 0x2000  | Nx  | 0x     |          |        |         |        |
| HV board write access | 0        | 0x2000  | Nx  | 0x     | ChSetLst |        | TypeSet |        |

Nx Group number 0 ... 31

0x Channel member offset 0, 16, 32 ... too access up to 255 channels

ChSetLst DATA\_2 to DATA\_3 ChannelSettingList members 0x1 ... 0xffff UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| CH15  | CH14  | CH13  | CH12  | CH11  | CH10  | CH9  | CH8  | CH7  | CH6  | CH5  | CH4  | CH3  | CH2  | CH1  | CH0  |

TypeSet DATA\_0 to DATA\_1 TypeSet UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| TYPE1 | TYPE0 | res   | res   | res   | res   | res  | MOD0 | SET3 | SET2 | SET1 | SET0 | MCH3 | MCH2 | MCH1 | MCH0 |

| TYPE1 | TYPE0 | Value        |                               |
|-------|-------|--------------|-------------------------------|
| 0     | 0     | SetGroupType | Group is defined as Set group |

| MOD0 | Value |                                        |
|------|-------|----------------------------------------|
| 0    | 0     | The group function is done one time    |
| 1    | 1     | The group function is done permanently |

| SET3 | SET2 | SET1 | SET0 | Value         |                                                                    |
|------|------|------|------|---------------|--------------------------------------------------------------------|
| 0    | 0    | 0    | 1    | SetVset       | Copy Vset from MCH to all members                                  |
| 0    | 0    | 1    | 0    | SetIset       | Copy Iset from MCH to all members                                  |
| 0    | 1    | 0    | 0    | SetVbnds      | Copy Vbounds from MCH to all members                               |
| 0    | 1    | 0    | 1    | SetIbnds      | Copy Ibounds from MCH to all members                               |
| 1    | 0    | 1    | 0    | SetOn         | Switch ON/OFF all members depending on setON in MCH                |
| 1    | 0    | 1    | 1    | SetEmrgCutOff | Switch OFF all members ( Emergency OFF )                           |
| 1    | 1    | 1    | 1    | Cloning       | Set all properties of members like MCH properties (in preparation) |

| MCH3 | MCH2 | MCH1 | MCH0 | Value |                                     |
|------|------|------|------|-------|-------------------------------------|
| 0    | 0    | 0    | 0    | 0     | 1: Channel 0 is MasterChannel MCH   |
| 0    | 0    | 0    | 1    | 1     | 1: Channel 1 is MasterChannel MCH   |
| ...  | ...  | ...  | ...  | ...   | ...                                 |
| 1    | 1    | 1    | 1    | 15    | 1: Channel 15 ist MasterChannel MCH |

### 4.5.3.2 Status group

Status groups are used to report the status of a single characteristic of all channels simultaneously. No action is foreseen. Therefore within the group following has to be defined :

- Members of the group: Each member will be activated in the channel status list ChStatLst.
- Type of the group: Status group type TypeStat
- Channel characteristics: Coding of characteristics which is to be reported.

EDCP frame:

| Access                | DATA_DIR | DATA_ID | NBR | OFFSET | DATA_3     | DATA_2 | DATA_1   | DATA_0 |
|-----------------------|----------|---------|-----|--------|------------|--------|----------|--------|
| master group write    | 0        | 0x2000  | Nx  | 0x     | ChStatList |        | TypeStat |        |
| master group read-    | 1        | 0x2000  | Nx  | 0x     |            |        |          |        |
| HV board write access | 0        | 0x2000  | Nx  | 0x     | ChStatList |        | TypeStat |        |

Nx Group number 0 ... 31

0x Channel member offset 0, 16, 32 ... to access up to 255 channels

ChStatLst DATA\_2 to DATA\_3 ChannelStatusList members 0x1 ... 0xffff UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11  | Bit10  | Bit9  | Bit8  | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
|-------|-------|-------|-------|--------|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| CH15  | CH14  | CH13  | CH12  | CHST11 | CHST10 | CHST9 | CHST8 | CHST7 | CHST6 | CHST5 | CHST4 | CHST3 | CHST2 | CHST1 | CHST0 |

TypeStat DATA\_0 to DATA\_1 TypeStatus UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7  | Bit6  | Bit5  | Bit4  | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|-------|-------|-------|-------|------|------|------|------|
| TYPE1 | TYPE0 | res   | res   | res   | res   | res  | res  | STAT3 | STAT2 | STAT1 | STAT0 | res  | res  | res  | res  |

| TYPE1 | TYPE0 | Value           |                                       |
|-------|-------|-----------------|---------------------------------------|
| 0     | 1     | StatusGroupType | Group will be defined as Status group |

| STAT3 | STAT2 | STAT1 | STAT0 | Value                  |                                                         |
|-------|-------|-------|-------|------------------------|---------------------------------------------------------|
| 0     | 0     | 1     | 1     | ChkIsOn                | check channel Status.isON (is on)                       |
| 0     | 1     | 0     | 0     | ChkIsRamping           | check channel Status.isRAMP (is ramping)                |
| 0     | 1     | 1     | 0     | ChkIsControlledCurrent | check channel Status.isCC (is current control)          |
| 0     | 1     | 1     | 1     | ChkIsControlledVoltage | check channel Status.isCV (is voltage control)          |
| 1     | 0     | 1     | 0     | ChkIsCurrentBounds     | check channel Status.isCBNDs (is current bounds)        |
| 1     | 0     | 1     | 1     | ChkIsVoltageBounds     | check channel Status.isVBNDs (is voltage bounds)        |
| 1     | 1     | 0     | 0     | ChkIsExternalInhibit   | check channel Status.isEINH (is external inhibit)       |
| 1     | 1     | 0     | 1     | ChkIsTrip              | check channel Status.isTRIP (is trip)                   |
| 1     | 1     | 1     | 0     | ChkIsCurrentLimit      | check channel Status.isCLIM (is current limit exceeded) |
| 1     | 1     | 1     | 1     | ChkIsVoltageLimit      | check channel Status.isVLIM (is voltage limit exceeded) |

### 4.5.3.3 Monitoring group

Monitoring groups are used to observe a single characteristic of selected channels simultaneously and in case of need take action. Therefore the group has to be defined :

- Members of the group: Each member will be activated in the channel monitoring list ChMonLst.
- Type of the group: Monitoring group type TypeMon
- Channel characteristics: Coding of characteristics which is to be monitored.
- Control mode: Coding of the control function, i.e. which kind of change in the group-image shall cause a signal.
- Activity: Define which activity has to happen after the event.

EDCP frame:

| Access                | DATA_DIR | DATA_ID | NBR | OFFSET | DATA_3   | DATA_2 | DATA_1  | DATA_0 |
|-----------------------|----------|---------|-----|--------|----------|--------|---------|--------|
| master group write    | 0        | 0x2000  | Nx  | 0x     | ChMonLst |        | TypeMon |        |
| master group read-    | 1        | 0x2000  | Nx  | 0x     |          |        |         |        |
| HV board write access | 0        | 0x2000  | Nx  | 0x     | ChMonLst |        | TypeMon |        |

Nx Group number 0 ... 31

0x Channel member offset 0, 16, 32 ... to access up to 255 channels

ChMonLst DATA\_2 to DATA\_3 ChannelMonitoringList members 0x1 ... 0xffff UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| CH15  | CH14  | CH13  | CH12  | CH11  | CH10  | CH9  | CH8  | CH7  | CH6  | CH5  | CH4  | CH3  | CH2  | CH1  | CH0  |

TypeMon DATA\_0 to DATA\_1 TypeMonitoring UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| TYPE1 | TYPE0 | ACT1  | ACT0  | res   | res   | res  | MOD0 | MON3 | MON2 | MON1 | MON0 | res  | res  | res  | res  |

| TYPE1 | TYPE0 | Value               |                                           |
|-------|-------|---------------------|-------------------------------------------|
| 1     | 0     | MonitoringGroupType | Group will be defined as Monitoring group |

| ACT1 | ACT0 | Value |                                                                  |
|------|------|-------|------------------------------------------------------------------|
| 0    | 0    | 0     | No special action ; EventGroupStatus[grp] will be set            |
| 0    | 1    | 1     | Ramp down of group EventGroupStatus[grp] will be set             |
| 1    | 0    | 2     | Shut down group without ramp; EventGroupStatus[grp] will be set  |
| 1    | 1    | 3     | Shut down module without ramp; EventGroupStatus[grp] will be set |

| MOD0 | Value |                                                |
|------|-------|------------------------------------------------|
| 0    | 0     | event will happen if at least one Channel == 0 |
| 1    | 1     | event will happen if at least one Channel == 1 |

| MON 3 | MON 2 | MON 1 | MON 0 | Value                      |                                                            |
|-------|-------|-------|-------|----------------------------|------------------------------------------------------------|
| 0     | 0     | 1     | 1     | MonitorIsOn                | monitor channel Status.isON (is on)                        |
| 0     | 1     | 0     | 0     | MonitorIsRamping           | monitor channel Status.isRAMP (is ramping)                 |
| 0     | 1     | 1     | 0     | MonitorIsControlledCurrent | monitor channel Status.isCC (is Constant Current)          |
| 0     | 1     | 1     | 1     | MonitorIsControlledVoltage | monitor channel Status.isCV (is Constant Voltage)          |
| 1     | 0     | 1     | 0     | MonitorIsCurrentBounds     | monitor channel Status.isCBNDs (is Current Bounds)         |
| 1     | 0     | 1     | 1     | MonitorIsVoltageBounds     | monitor channel Status.isVBNDs (is Voltage Bounds)         |
| 1     | 1     | 0     | 0     | MonitorIsExternalInhibit   | monitor channel Status.isEINH (is External Inhibit)        |
| 1     | 1     | 0     | 1     | MonitorIsTrip              | monitor channel Status.isTRIP (is Trip)                    |
| 1     | 1     | 1     | 0     | MonitorIsCurrentLimit      | monitor channel Status.isCLIM (is Current Limit exceeded.) |
| 1     | 1     | 1     | 1     | MonitorIsVoltageLimit      | monitor channel Status.isVLIM (is Voltage Limit exceeded.) |

### 4.5.3.4 Delayed Trip group

Trip timeout groups are necessary to keep the timing for the time controlled delayed Trip function and to define the action which has to happen after a Trip.

Therefore in the group following will be defined:

- Members of group: Each member will be activated in a word channel trip timeout list ChTrpTotLst.
- Type of the group: Time out group type TypeTime
- Activity: Define which activity has to happen after time controlled Trip
- Timeout: Coding of Timeout-time as integer value.

Timeout groups have to stay unchanged for the whole time as long they are used.

An overwriting will cause the definition of a new group. An overlay of the channels of multiple Trip groups is not allowed.

EDCP frame:

| Access                | DATA_DIR | DATA_ID | NBR | OFFSET | DATA_3      | DATA_2 | DATA_1   | DATA_0 |
|-----------------------|----------|---------|-----|--------|-------------|--------|----------|--------|
| master group write    | 0        | 0x2000  | Nx  | Ox     | ChTrpTotLst |        | TypeTime |        |
| master group read-    | 1        | 0x2000  | Nx  | Ox     |             |        |          |        |
| HV board write access | 0        | 0x2000  | Nx  | Ox     | ChTrpTotLst |        | TypeTime |        |

Nx Group number 0 ... 31

Ox Channel member offset 0, 16, 32 ... too access up to 255 channels

ChTrpTotLst DATA\_2 to DATA\_3 ChannelTripTimeoutList members 0x1 ... 0xffff UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| CH15  | CH14  | CH13  | CH12  | CH11  | CH10  | CH9  | CH8  | CH7  | CH6  | CH5  | CH4  | CH3  | CH2  | CH1  | CH0  |

TypeTime DATA\_0 to DATA\_1 TypeTimeOut UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| TYPE1 | TYPE0 | ACT1  | ACT0  | TOT11 | TOT10 | TOT9 | TOT8 | TOT7 | TOT6 | TOT5 | TOT4 | TOT3 | TOT2 | TOT1 | TOT0 |

| TYPE1 | TYPE0 | Value            |                                        |
|-------|-------|------------------|----------------------------------------|
| 1     | 1     | TimeoutGroupType | Group will be defined as Timeout group |

| ACT1 | ACT0 | Action |                                                                      |
|------|------|--------|----------------------------------------------------------------------|
| 0    | 0    | 0      | No special action; EventGroupStatus[grp] will be set.                |
| 0    | 1    | 1      | Ramp down group with ramp; EventGroupStatus[grp] will be set         |
| 1    | 0    | 2      | Shut down the group without ramp; EventGroupStatus[grp] will be set  |
| 1    | 1    | 3      | Shut down the module without ramp; EventGroupStatus[grp] will be set |

| TOT[11..0]: | Timeout-time in ms (8..4088ms) resolution is 8ms (different values to 8ms resolution will be rounded) |
|-------------|-------------------------------------------------------------------------------------------------------|
|-------------|-------------------------------------------------------------------------------------------------------|

Set voltage of all channels (group write access)

EDCP frame:

| Access              | DATA_DIR | DATA_ID | DATA_3                | DATA_2 | DATA_1 | DATA_0 |
|---------------------|----------|---------|-----------------------|--------|--------|--------|
| master write access | 0        | 0x2100  | VoltageSetAllChannels |        |        |        |

VoltageSetAllChannels DATA\_0 to DATA\_3 [V] R4

(see [VoltageSet](#) Single access also)

#### 4.5.3.5 Set current (– trip) of all channels (group write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID              | DATA_3                | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|----------------------|-----------------------|--------|--------|--------|
| master write access   | 0        | 0x2101               | CurrentSetAllChannels |        |        |        |
| CurrentSetAllChannels |          | DATA_0 to DATA_3 [A] |                       |        |        | R4     |

(see [CurrentSet](#) Single access also)

#### 4.5.3.6 Set ON / OFF of all channels (group write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3         | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|----------------|--------|--------|--------|
| master group write    | 0        | 0x2200  | SetOnOffAllChs |        |        |        |
| master group read-    | 1        | 0x2200  |                |        |        |        |
| HV board write access | 0        | 0x2200  | SetOnOffAllChs |        |        |        |

SetOnOffAllChs DATA\_0 to DATA\_3 for channel 0 to 31 (channel 16 to 31 are reserved at the moment) UI4

SetOnOffAllChs DATA\_2 to DATA\_3 for channel 16 to channel 31 (reserved at the moment) UI2

| Bit31                                                           | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|-----------------------------------------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| CH31                                                            | CH30  | CH29  | CH28  | CH27  | CH26  | CH25  | CH24  | CH23  | CH22  | CH21  | CH20  | CH19  | CH18  | CH17  | CH16  |
| SetOnOffAllChs DATA_0 to DATA_1 for channel 0 to channel 15 UI2 |       |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| Bit15                                                           | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| CH15                                                            | CH14  | CH13  | CH12  | CH11  | CH10  | CH9   | CH8   | CH7   | CH6   | CH5   | CH4   | CH3   | CH2   | CH1   | CH0   |

CHm = 1 Channel ON

CHm = 0 Channel OFF

The SetOnOffAllChs represents a 32 bit field to control the channel property setON for each channel member of the bit field with one bit in the bit-field. The data point is one of the EDCP group functions with advantage to setON for all channel members simultaneously with on instruction.

#### 4.5.3.7 Set EMERGENCY of all channels (group write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3             | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|--------------------|--------|--------|--------|
| master group write    | 0        | 0x2201  | SetEmergencyAllChs |        |        |        |
| master group read-    | 1        | 0x2201  |                    |        |        |        |
| HV board write access | 0        | 0x2201  | SetEmergencyAllChs |        |        |        |

SetEmergencyAllChs DATA\_0 to DATA\_3 for channel 0 to 31 (channel 16 to 31 are reserved at the moment) UI4

SetEmergencyAllChs DATA\_2 to DATA\_3 for channel 16 to channel 31 (reserved at the moment) UI2

| Bit31                                                               | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|---------------------------------------------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| CH31                                                                | CH30  | CH29  | CH28  | CH27  | CH26  | CH25  | CH24  | CH23  | CH22  | CH21  | CH20  | CH19  | CH18  | CH17  | CH16  |
| SetEmergencyAllChs DATA_0 to DATA_1 for channel 0 to channel 15 UI2 |       |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
| Bit15                                                               | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| CH15                                                                | CH14  | CH13  | CH12  | CH11  | CH10  | CH9   | CH8   | CH7   | CH6   | CH5   | CH4   | CH3   | CH2   | CH1   | CH0   |

CHm = 1 Channel EMERGENCY set

CHm = 0 Channel EMERGENCY reset

The SetEmergencyAllChs represents a 32 bit field to control the channel property setEMCY for each channel member of the bit field with one bit in the bit-field. The data point is one of the EDCP group functions with advantage to setEMCY for all channel members simultaneously with on instruction.

#### 4.5.3.8 Event status voltage limit of all channels (group write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3                  | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|-------------------------|--------|--------|--------|
| master group write    | 0        | 0x2202  | EventStatusVLimitAllChs |        |        |        |
| master group read-    | 1        | 0x2202  |                         |        |        |        |
| HV board write access | 0        | 0x2202  | EventStatusVLimitAllChs |        |        |        |

EventStatusVLimitAllChs DATA\_0 to DATA\_3 for channel 0 to 31 (channel 16 to 31 are reserved at the moment) UI4

EventStatusVLimitAllChs DATA\_2 to DATA\_3 for channel 16 to channel 31 (reserved at the moment) UI2

| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| CH31  | CH30  | CH29  | CH28  | CH27  | CH26  | CH25  | CH24  | CH23  | CH22  | CH21  | CH20  | CH19  | CH18  | CH17  | CH16  |

EventStatusVLimitAllChs DATA\_0 to DATA\_1 for channel 0 to channel 15 UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| CH15  | CH14  | CH13  | CH12  | CH11  | CH10  | CH9  | CH8  | CH7  | CH6  | CH5  | CH4  | CH3  | CH2  | CH1  | CH0  |

CHm = 1 Channel event status voltage limit

CHm = 0 nothing

The EventStatusVLimitAllChs represents a 32 bit field to control the channel property EVLIM for each channel with one bit in the bit-field. The data point is one of the EDCP group functions with advantage to reset EVLIM for all channel members simultaneously with on instruction.

If the voltage limit was exceeded or an external over voltage occurs at the channel output (i.e. Output voltage > Set voltage) then the channel will be switched off and the according bit will be set. The error bits will be canceled and the voltage of the corresponding channel can be switched on again only after writing 'EventStatusVLimitAllChs' with the bits, which are corresponding to the channel errors are set to "1".

#### 4.5.3.9 Event status current limit of all channels (group write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3                  | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|-------------------------|--------|--------|--------|
| master group write    | 0        | 0x2203  | EventStatusCLimitAllChs |        |        |        |
| master group read-    | 1        | 0x2203  |                         |        |        |        |
| HV board write access | 0        | 0x2203  | EventStatusCLimitAllChs |        |        |        |

EventStatusCLimitAllChs DATA\_0 to DATA\_3 for channel 0 to 31 (channel 16 to 31 are reserved at the moment) UI4

EventStatusCLimitAllChs DATA\_2 to DATA\_3 for channel 16 to channel 31 (reserved at the moment) UI2

| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| CH31  | CH30  | CH29  | CH28  | CH27  | CH26  | CH25  | CH24  | CH23  | CH22  | CH21  | CH20  | CH19  | CH18  | CH17  | CH16  |

EventStatusCLimitAllChs DATA\_0 to DATA\_1 for channel 0 to channel 15 UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| CH15  | CH14  | CH13  | CH12  | CH11  | CH10  | CH9  | CH8  | CH7  | CH6  | CH5  | CH4  | CH3  | CH2  | CH1  | CH0  |

CHm = 1 Channel event status current limit

CHm = 0 nothing

The EventStatusCLimitAllChs represents a 32 bit field to control the channel property ECLIM for each channel with one bit in the bit-field. The data point is one of the EDCP group functions with advantage to reset ECLIM for all channel members simultaneously with on instruction.

The module responds to the exceeding of the hardware current limit which has been set in the channel in dependence on the according setKILena bit of module control as follows:

- setKILena = 1: Voltage will be switched off permanently without ramp, green LED on front panel is off until a write of EventStatusCLimitAllChs with the bits, which are corresponding to the channel errors set to "1". The error bits will be cancelled and the voltage of the corresponding channels can be switched on again.
- setKILena = 0: HV modules without a current control E16D0, E16D1 and E08B0 will be switched off voltage without ramp, green LED on front panel is off. If the output voltage arrives at 0 V the ramping to set voltage will be started automatically again. The green LED again flash only after writing the EventStatusCLimitAllChs with the respective bits.

HV modules with a current control will not switch off high voltage and operate in constant current mode, green LED on front panel is off. The output current will be limited. The green LED flashes only after writing of EventStatusCLimitAllChs with the respective bits and removing of the limitation of current before.

#### 4.5.3.10 Event status trip of all channels (group write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3               | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|----------------------|--------|--------|--------|
| master group write    | 0        | 0x2204  | EventStatusTrpAllChs |        |        |        |
| master group read-    | 1        | 0x2204  |                      |        |        |        |
| HV board write access | 0        | 0x2204  | EventStatusTrpAllChs |        |        |        |

EventStatusTrpAllChs DATA\_0 to DATA\_3 for channel 0 to 31 (channel 16 to 31 are reserved at the moment) UI4

EventStatusTrpAllChs DATA\_2 to DATA\_3 for channel 16 to channel 31 (reserved at the moment) UI2

| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| CH31  | CH30  | CH29  | CH28  | CH27  | CH26  | CH25  | CH24  | CH23  | CH22  | CH21  | CH20  | CH19  | CH18  | CH17  | CH16  |

EventStatusTrpAllChs DATA\_0 to DATA\_1 for channel 0 to channel 15 UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| CH15  | CH14  | CH13  | CH12  | CH11  | CH10  | CH9  | CH8  | CH7  | CH6  | CH5  | CH4  | CH3  | CH2  | CH1  | CH0  |

CHm = 1 Channel event status trip

CHm = 0 nothing

The EventStatusTrpAllChs represents a 32 bit field to control the channel property ETRP for each channel with one bit in the bit-field. The data point is one of the EDCP group functions with advantage to reset ETRP for all channel members simultaneously with on instruction.

If the output current exceeds the programmed current trip value then the corresponding bits will be set:

setKILena = 1: Voltage will be switched off permanently without ramp, green LED on front panel is off until a write of EventStatusTrpAllChs with the bits, which are corresponding to the channel errors set to "1". The error bits will be cancelled and the voltage of the corresponding channels can be switched on again.

setKILena = 0: HV will not be switched but the green LED on front panel is off. The output current will be limited if there is a current control. The green LED flashes only after writing of EventStatusTrpAllChs with the respective bits and removing of the limitation of current before.

#### 4.5.3.11 Event status inhibit of all channels (group write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3               | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|----------------------|--------|--------|--------|
| master group write    | 0        | 0x2205  | EventStatusInhAllChs |        |        |        |
| master group read-    | 1        | 0x2205  |                      |        |        |        |
| HV board write access | 0        | 0x2205  | EventStatusInhAllChs |        |        |        |

EventStatusInhAllChs DATA\_0 to DATA\_3 for channel 0 to 31 (channel 16 to 31 are reserved at the moment) UI4

EventStatusInhAllChs DATA\_2 to DATA\_3 for channel 16 to channel 31 (reserved at the moment) UI2

| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| CH31  | CH30  | CH29  | CH28  | CH27  | CH26  | CH25  | CH24  | CH23  | CH22  | CH21  | CH20  | CH19  | CH18  | CH17  | CH16  |

EventStatusInhAllChs DATA\_0 to DATA\_1 for channel 0 to channel 15 UI2

| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| CH15  | CH14  | CH13  | CH12  | CH11  | CH10  | CH9  | CH8  | CH7  | CH6  | CH5  | CH4  | CH3  | CH2  | CH1  | CH0  |

CHm = 1 Channel event status inhibit

CHm = 0 nothing

The EventStatusInhAllChs represents a 32 bit field to control the channel property ETRP for each channel with one bit in the bit-

field. The data point is one of the EDCP group functions with advantage to reset ETRP for all channel members simultaneously with on instruction.

Voltage will be switched off permanently without ramp, if the INHIBIT is active, the green LED on front panel is off. When the INHIBIT is going back from active to passive state then the INHIBIT flag have to be erased by write of the EventStatusInhAllChs before the voltage can be switched on again. The INHIBIT flags are reset with set of the corresponding channel bit to "1".

#### 4.5.3.12 Set ON / OFF channels extender (group write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3                     | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|----------------------------|--------|--------|--------|
| master group write    | 0        | 0x2280  | <u>SetOnOffChsExtender</u> |        |        |        |
| master group read-    | 1        | 0x2280  |                            |        |        |        |
| HV board write access | 0        | 0x2280  | <u>SetOnOffChsExtender</u> |        |        |        |

SetOnOffAllChs DATA\_0 to DATA\_3 for channel 0 to 31 (channel 16 to 31 are reserved at the moment) UI4

SetOnOffAllChs DATA\_2 to DATA\_3 for channel 16 to channel 31 (reserved at the moment) UI2

|       |       |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| CH63  | CH62  | CH61  | CH60  | CH59  | CH58  | CH57  | CH56  | CH55  | CH54  | CH53  | CH52  | CH51  | CH50  | CH49  | CH48  |

SetOnOffAllChs DATA\_0 to DATA\_1 for channel 0 to channel 15 UI2

|       |       |       |       |       |       |      |      |      |      |      |      |      |      |      |      |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| CH47  | CH46  | CH45  | CH44  | CH43  | CH42  | CH41 | CH40 | CH39 | CH38 | CH37 | CH36 | CH36 | CH34 | CH33 | CH32 |

CHm = 1 Channel ON

CHm = 0 Channel OFF

The SetOnOffChsExtender represents a 32 bit field to control the channel property setON for each channel member of the bit field with one bit in the bit-field. The data point is one of the EDCP group functions with advantage to setON for all channel members simultaneously with on instruction.

#### 4.5.3.13 Set EMERGENCY channels extender (group write- / read-write access)

EDCP frame:

| Access                | DATA_DIR | DATA_ID | DATA_3                         | DATA_2 | DATA_1 | DATA_0 |
|-----------------------|----------|---------|--------------------------------|--------|--------|--------|
| master group write    | 0        | 0x2201  | <u>SetEmergencyChsExtender</u> |        |        |        |
| master group read-    | 1        | 0x2201  |                                |        |        |        |
| HV board write access | 0        | 0x2201  | <u>SetEmergencyChsExtender</u> |        |        |        |

SetEmergencyChsExtender DATA\_0 to DATA\_3 for channel 0 to 31 (channel 16 to 31 are reserved at the moment) UI4

SetEmergencyChsExtender DATA\_2 to DATA\_3 for channel 16 to channel 31 (reserved at the moment) UI2

|       |       |       |       |       |       |       |       |       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| CH63  | CH62  | CH61  | CH60  | CH59  | CH58  | CH57  | CH56  | CH55  | CH54  | CH53  | CH52  | CH51  | CH50  | CH49  | CH48  |

SetEmergencyChsExtender DATA\_0 to DATA\_1 for channel 0 to channel 15 UI2

|       |       |       |       |       |       |      |      |      |      |      |      |      |      |      |      |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| CH47  | CH46  | CH45  | CH44  | CH43  | CH42  | CH41 | CH40 | CH39 | CH38 | CH37 | CH36 | CH36 | CH34 | CH33 | CH32 |

CHm = 1 Channel EMERGENCY set

CHm = 0 Channel EMERGENCY reset

The SetEmergencyChsExtender represents a 32 bit field to control the channel property setEMCY for each channel member of the bit field with one bit in the bit-field. The data point is one of the EDCP group functions with advantage to setON for all channel members simultaneously with on instruction.

## 4.5.4 Important DCP Module Accesses

### 4.5.4.1 General status (group write- / read-write / active access)

DCP frame:

| Access                 | EXT_INSTR | DATA_DIR | DATA_ID | DATA_1        | DATA_0  |
|------------------------|-----------|----------|---------|---------------|---------|
| HV board active access | 0         | 0        | 0xc0    | GeneralStatus | Details |

GeneralStatus DATA\_1 UI1

Details DATA\_0 UI1

| b15  | b14                | b13       | b12  | b11  | b10   | b9     | b8       | b7   | b6        | b5  | b4  | b3   | b2   | b1   | b0  |
|------|--------------------|-----------|------|------|-------|--------|----------|------|-----------|-----|-----|------|------|------|-----|
| Save | KILLena/HwVLnotLow | SPLYTMPgd | AvAd | Stbl | SFLPg | noRamp | noSumErr | INHb | BoardTemp | res | res | VLIM | CLIM | RERR | TRP |

|            |                            |                                                                                          |
|------------|----------------------------|------------------------------------------------------------------------------------------|
| Save       | Save                       | save function bit stored permanently the current set values (takes some seconds ca. 10s) |
| KILLena    | KillEnable                 | kill function bit                                                                        |
| HwVLnotLow | HardwareVoltageLimitNotLow | hardware voltage limit is not to low bit, for device class 21 only                       |
| SPLYTMPgd  | SupplyGoodTemperatureGood  | supply good and board temperature good bit                                               |
| AvAd       | AverageAdjust              | average and fine adjustment bit                                                          |
| SFLPg      | SafetyLoop                 | safety loop bit                                                                          |
| noRamp     | noRamp                     | flag to display that no voltage is ramping                                               |
| noSumErr   | NoSumError                 | displays that there has been built a sum error flag by VLIM&ILIM&TRP over all channels   |
| INHb       | Inhibit                    | an external INHIBIT at least one of the channels (device class 25)                       |
| BoardTemp  | BoardTemperatureGood       | board temperature is good                                                                |
| VLIM       | VoltageLimit               | hardware voltage limit has been exceeded                                                 |
| CLIM       | CurrentLimit               | hardware current limit has been exceeded                                                 |
| RERR       | RegulationError            | regulation error, for device class 21 only                                               |
| TRP        | Trip                       | voltage or current trip                                                                  |
| res        | reserved                   |                                                                                          |

|                |                                                                                                                             |              |                                                                                                                                             |
|----------------|-----------------------------------------------------------------------------------------------------------------------------|--------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| Save=0         | no write access to EEPROM                                                                                                   | Stbl = 0     | all channels are stable with program ADC filter frequency fN. (ADC conversion time =1/fN, see 'Set ADC filter frequency', default fN=50 Hz) |
| Save=1         | store all set values to EEPROM (time to ca. 10s) for device classes 24 and 25 only                                          | Stbl = 1     | at least one channel is ramping Vo or not yet stable after ramping (ramping - with ADC filter frequency fN=100 Hz)                          |
| KILLena=0      | kill function disable                                                                                                       | SFLPg = 0    | safety loop is broken -VO has been shut off, reset by a write of the 'General status' with sloop flag is set to "1"                         |
| KILLena = 1    | kill function enable, for device classes 21 only                                                                            | SFLPg = 1    | safety loop is closed                                                                                                                       |
| HwVLnotLow = 0 | HW Vlimit voltage limit is to low, it is not possible to switch on the HV, reset by write with HwVLToLow flag is set to "1" | noRamp = 0   | VO is ramping in at least one channel                                                                                                       |
| HwVLnotLow = 1 | HW Vlimit in proper range                                                                                                   | noRamp = 1   | no channel is ramping                                                                                                                       |
| SPLYTMPgd = 0  | supply voltages are out of range or module temperature > 55°C                                                               | noSumErr = 0 | voltage limit, current limit or trip has been exceeded in at least one of the channels (error)                                              |
| SPLYTMPgd = 1  | supply voltages are in range and module temperature <=55°C                                                                  | noSumErr = 1 | status channel flags v & c & t = 0 for all channels (no errors)                                                                             |
| AvAd = 0       | fine adjustment and average of voltage, current measurement OFF                                                             | INHb = 0     | no external INHIBIT signal                                                                                                                  |
| AvAd = 1       | fine adjustment and average of voltage, current measurement ON                                                              | INHb = 1     | external INHIBIT signal                                                                                                                     |

BoardTemp = 0    temperature <= 55°C                      RERR=0        hardware current hasn't been exceeded  
 BoardTemp = 1    temperature > 55°C                      RERR=1        voltage has been exceeded  
 VLIM=0    hardware voltage limit hasn't been exceeded    TRP=0        no trip  
 VLIM=1    hardware voltage limit has been exceeded       TRP=1        voltage or current trip  
 CLIM=0    hardware current limit hasn't been exceeded  
 CLIM=1    hardware current limit has been exceeded

If one of the bits noHwVLtoLow, SPLYTMPgd, SFLPg, noSumErr in the modul access "General status module" has not been set, the module will send this access as an active error message with higher priority (ID9=0). An additional 2nd data byte offers more information about the NoSumError flag of the first byte. OPC

Example of an active error message

| access                 | identifier | length code | DATA_ID | DATA_1 | DATA_0 |
|------------------------|------------|-------------|---------|--------|--------|
| HV board active access | 0x180      | 3           | 0xc0    | 0x57   | 0x01   |

TRP=1 noSumErr=0 etc.

#### 4.5.4.2 Log-on / Log-off Front-end device at superior layer (module active- / write access)

DCP frame:

| Access                 | DATA_DIR | DATA_ID | DATA_1        | DATA_0      |
|------------------------|----------|---------|---------------|-------------|
| HV board active access | 1        | 0xD8    | GeneralStatus | DeviceClass |
| master write access    | 0        | 0xD8    | LogOnOff      |             |

GeneralStatus DATA\_1 – refer chapter 4.    UI1

DeviceClass    DATA\_0        UI1

| device class<br>EDCP   DCP               | label       | firmware                            | description                                                                       | associated<br>serial numbers           |
|------------------------------------------|-------------|-------------------------------------|-----------------------------------------------------------------------------------|----------------------------------------|
| 21    0                                  | EDS         | E16D0_xxx<br>E16D1_xxx<br>E24D1_xxx | EDS 16 channels per PCB<br>EDS 16 channels per PCB<br>EDS 24 channels per PCB     | 471xxx /<br>71xxxx                     |
| 22    -                                  | HPS / LPS   | H101C0_5xx                          | HPS 19", 1 channel HV Power Supply (300W, 800W)                                   | 68xxxx /<br>70xxxx                     |
| 23    -                                  | HPS / LPS   | H201C0_2xx                          | HPS compact, 1 channel HV Power Supply (350W)                                     | 68xxxx /<br>70xxxx                     |
| 24    6<br>24    6<br>25    7<br>25    7 | EHS/EMS/ELS | E08C0_xxx                           | ExS 8 channel per PCB, standard common GND                                        | 73xxxx<br>73xxxxx<br>74xxxx<br>74xxxxx |
| 25    7                                  | EHS/EMS/ELS | E08F0_xxx                           | ExS 8 channel per PCB, standard, floating GND                                     | 474xxx                                 |
| 26    2                                  | EHS/EMS/ELS | E08F2_xxx                           | ExS 8 channel per PCB, floating GND<br>2 ranges for measurement of current        | 72xxxx<br>72xxxxx                      |
| 27    -                                  | EHS/EMS/ELS | E08C2_xxx                           | ExS 8 channel per PCB, common floating GND<br>2 ranges for measurement of current | 78xxxx<br>78xxxxx                      |
| 41    -                                  | MICC        | MICC_3xx                            | MICC Multichannel Interface Crate Controller for MMC Crates                       | 458xxx                                 |
| 42    -                                  | MICP        | MICP_3xx                            | MICP Multichannel Interface Crate PHQ Controller for MMC Crates                   | 458xxx                                 |
| 60    -                                  | HPS         | H101C1_1xx                          | HPS 19", 1 channel HV Power Supply (1.5kW – 10 kW)                                | 90xxxxx                                |
| 70    -                                  | EHS         | E16C1_1xx                           | ExS 16 channels per PCB or 32 channels per module                                 | 79xxxx<br>79xxxxx                      |

LogOnOff    DATA\_1=1 superior layer send a "Log-on" at Front-end device to registration    UI1  
 DATA\_1=0 superior layer send "Log-off" to Front-end device

xxx and xxxx are running numbers

After POWER ON the Front-end device - up to a number of two per module - will give this module access cyclically on the bus (ca. 1 sec). If a controller of superior layer identifies this access then it is possible to register this as a Front-end device and is possible to address it with FE\_ADR. (see also description 11bit-Identifier)

After the successful registration the Front-end device will not send further 'Log-on' accesses as long as:

- it receives accesses from the external CAN Bus in periods shorter than one minute or
- until the superior controller will send a 'Log-off' access to the Front-end device.

## 4.5.5 Events

The module provides an extended event collecting logic. This is necessary to monitor extraordinary events and forward them to the host.

### 4.5.5.1 Channel events

These event-bits in the channel event status register are related to mask bits in the channel event mask register. With help of an AND function (bit-wise) between an event bit and the according mask bit a result only occurs where the mask bit has been set. A following logic OR function of all of these results leads to the event status of the channels.

```
ModuleEventChannelStatus[ch]    =    (ChannelEventStatus.EVLIM[ch] AND ChannelEventMask.MEVLIM[ch]) OR
                                     (ChannelEventStatus.ECLIM[ch] AND ChannelEventMask.MECLIM[ch]) OR
                                     (ChannelEventStatus.ETRP[ch] AND ChannelEventMask.METRP[ch]) OR
                                     (ChannelEventStatus.EEINH[ch] AND ChannelEventMask.MEEINH[ch]) OR
                                     (ChannelEventStatus.EVBNDs[ch] AND ChannelEventMask.MEVBNDS[ch]) OR
                                     (ChannelEventStatus.ECBNDs[ch] AND ChannelEventMask.MECBNDs[ch]) OR
                                     (ChannelEventStatus.ECV[ch] AND ChannelEventMask.MECV[ch]) OR
                                     (ChannelEventStatus.ECC[ch] AND ChannelEventMask.MECC[ch]) OR
                                     (ChannelEventStatus.EEMCY[ch] AND ChannelEventMask.MEEMCY[ch]) OR
                                     (ChannelEventStatus.EEOR[ch] AND ChannelEventMask.MEEOR[ch]) OR
                                     (ChannelEventStatus.EOn2Off[ch] AND ChannelEventMask.MEOn2Off [ch]) OR
                                     (ChannelEventStatus.EIER[ch] AND ChannelEventMask.MEIER[ch])
                                     ch={0..n}
```

The status of all channel events is collected in the register EventChannelStatus of the module items.

For a selection or filtering of the channel events a related mask register has been provided ([ModuleEventChannelMask](#)). With help of the AND or OR function (see channel) the event active signal of the channels EventChannelActive will be generated:

```
EventChannelActive    =    (EventChannelStatus[0] AND EventChannelMask[0]) OR
                             (EventChannelStatus[1] AND EventChannelMask[1]) OR
                             ...
                             (EventChannelStatus[n] AND EventChannelMask[n])
```

### 4.5.5.2 Group events (in preparation)

Like written before groups are also able to generate Events. These events will be collected in the status word EventGroupStatus of the GroupData. With help of the mask register EventGroupMask the event active signal of the groups EventGroupActive will be generated.

```
EventGroupActive    =    (EventGroupStatus[0] AND EventGroupMask[0]) OR
                             (EventGroupStatus[1] AND EventGroupMask[1]) OR
                             ...
                             (EventGroupStatus[23] AND EventGroupMask[24])
```

#### 4.5.5.3 Module events

With help of the NOT, AND or OR function the event active signal of the module EventModuleActive will be generated:

EventModuleActive = (NOT(ModuleEventStatus.ETMPngd) AND ChannelEventMask.METMPngd) OR  
(NOT(ModuleEventStatus.ESPLYngd) AND ChannelEventMask.MESPLYngd) OR  
(NOT(ModuleEventStatus.ESFLPngd) AND ModuleEventMask.MESFLPngd) OR

From both signals EventChannelActive and EventModuleActive the global event active signal of the module IsEventActive of the ModuleStatus register will be generated.

IsEventActive = EventChannelActive OR EventGroupActive OR EventModuleActive

This global signal 'IsEventActive' triggers a fast message on the CAN bus with the DCP Module frame of [General status](#).

Example:

The event flag ECC of the ChannelEventStatus for channel 2 or the event flag EventTemperatureNotGood of the ModuleEventStatus should release a fast CAN frame:

- Channel[2].ChannelEventMask.Bit.MECC = 1
- Module.EventChannelMask.Bit.2 = 1
- Module.EventMask.Bit.METMPngd = 1

The signal isEvtActive is triggered and release a fast CAN frame of General status when:

( Channel[2].ChannelEventStatus.Bit.ECC & Channel[2].ChannelEventMask.Bit.MECC & Module.ModuleEventChannelMask.Bit2  
OR  
Module.ModuleEventStatus.Bit.ETMPngd & Module.ModuleEventMask.Bit.METMPngd  
OR  
(Module.ModuleEventChannelStatus.Bit2 & Module.ModuleEventChannelMask.Bit2 )

Fast CAN frame in case of Channel[2].ChannelEventStatus.Bit.ECC == 1:

0x190 3 0xc0 0x37 0x00 (ID=0x190, ID9=0; Len=3; DATA\_ID=0xc0; Data=0x3700)  
(Channel[2].ChannelEventStatus.Bit.ECC & Channel[2].ChannelEventMask.Bit.MECC)==1 →  
ModuleEventChannelStatus.Bit2=1

Fast CAN frame in case of Module.ModuleEventStatus.Bit. ETMPngd == 1:

0x190 3 0xc0 0x17 0x40 (ID=0x190, ID9=0; Len=3; DATA\_ID=0xc0; Data= 0x1740)

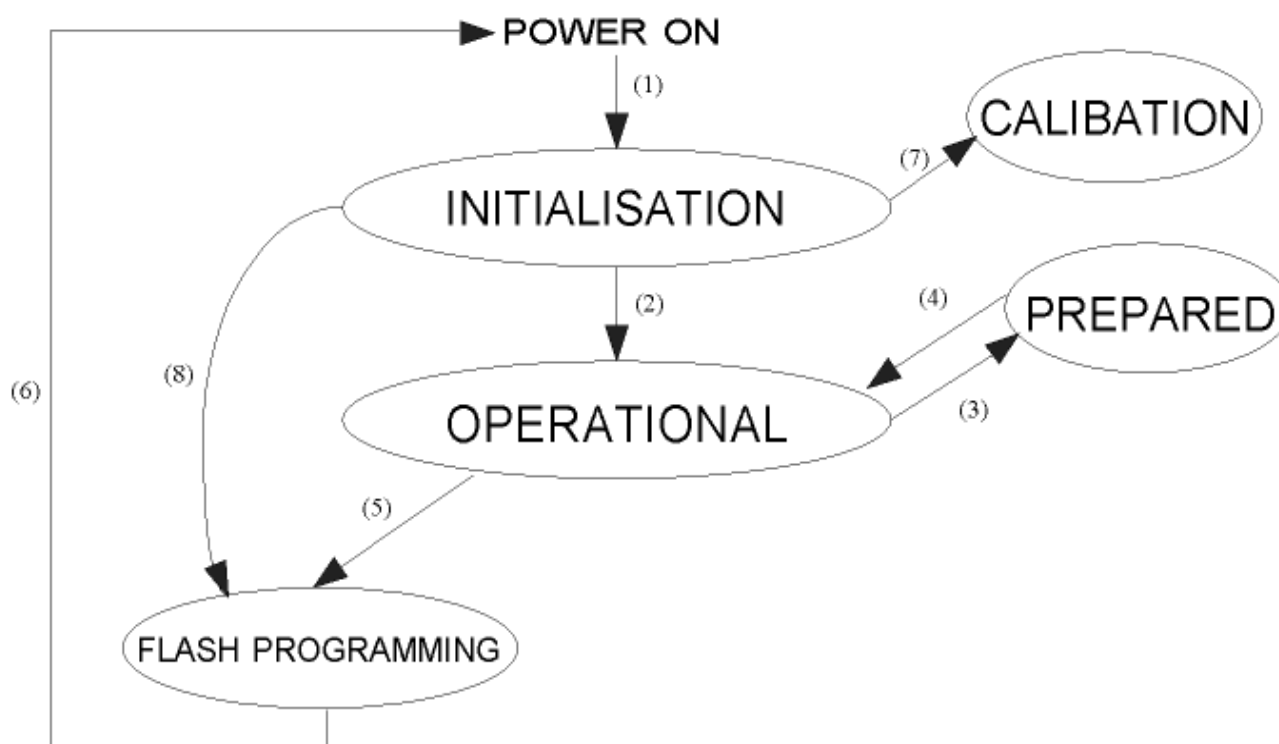


Please note that, a release of a fast CAN frame is different in handling depending on EDCP or DCP mode!

## Appendix A – Shortcuts

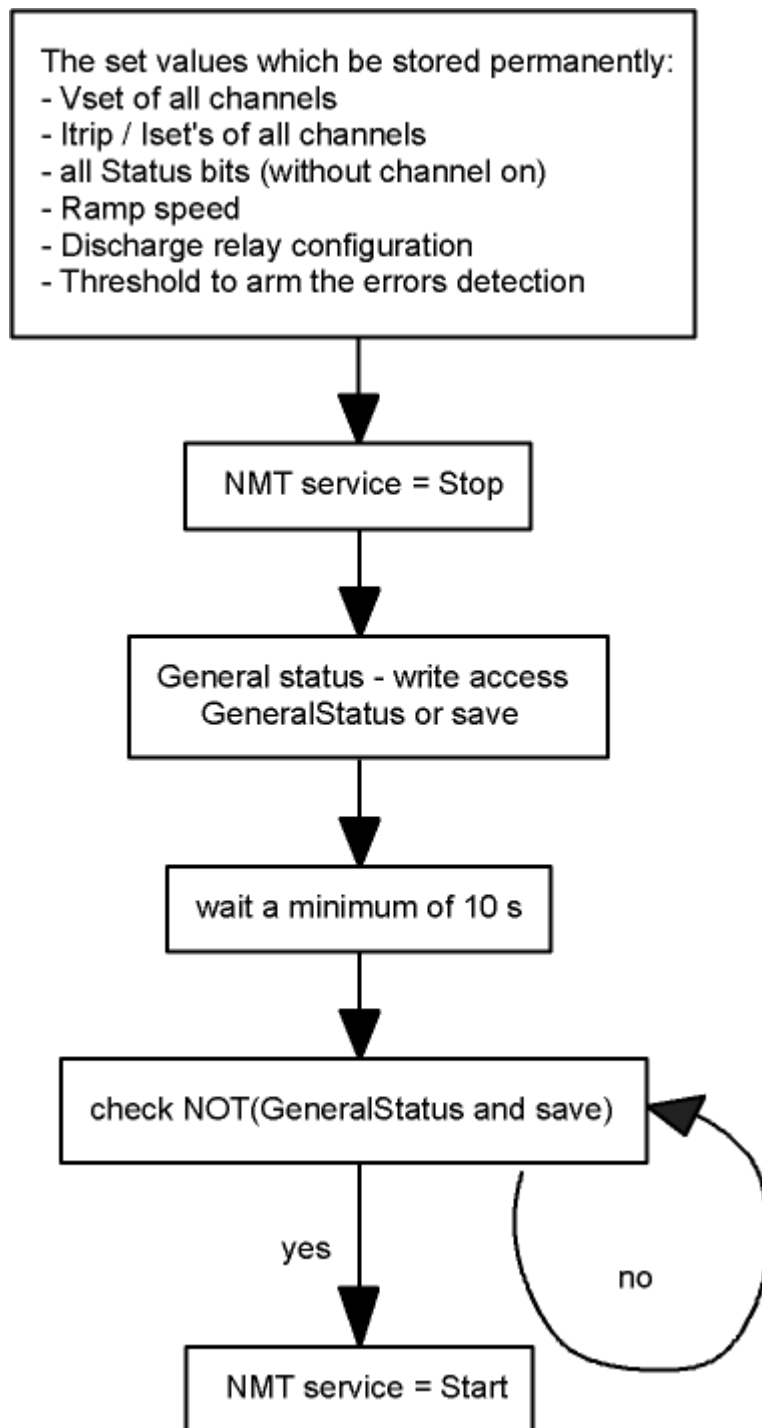
|                   |                                                     |
|-------------------|-----------------------------------------------------|
| BCD               | binary coded decimal format                         |
| CAN               | controller area network                             |
| Chm               | channel m=0..15                                     |
| CHN               | channel                                             |
| DCP               | device control protocol                             |
| DATA_ID           | data identifier of DCP                              |
| fN                | first filter notch frequency                        |
| HV                | High voltage                                        |
| HW                | hardware                                            |
| I <sub>meas</sub> | Actual current                                      |
| I <sub>max</sub>  | Hardware current limit                              |
| IO max            | Nominal current                                     |
| I <sub>set</sub>  | Set current                                         |
| I <sub>trip</sub> | Trip current                                        |
| ISO               | International Standard Organization                 |
| LSB               | least significant bit                               |
| MBR               | channel members                                     |
| MSB               | most significant bit                                |
| NBR               | group number                                        |
| NMT               | network management service                          |
| OSI               | Open System Interconnect                            |
| PCB               | printed circuit board                               |
| p/a               | passive / active                                    |
| SN.               | serial number                                       |
| UI1               | unsigned character                                  |
| SI1               | signed character                                    |
| UI2               | unsigned short integer (16 bit)                     |
| UI4               | unsigned integer (32 bit)                           |
| R4                | float according to IEEE-754 single precision format |
| V <sub>meas</sub> | Actual voltage                                      |
| V <sub>max</sub>  | Hardware voltage limit                              |
| VO max            | Nominal voltage                                     |
| V <sub>set</sub>  | Set voltage                                         |
| SW                | software                                            |

## Appendix B – Diagram of operating modes



- (1) The INITIALISATION follows after the POWER ON reset of the device hardware. It can be differ between different device classes.
- (2) The state OPERATIONAL will be obtained by the device itself if all initializations are ready or the state PREPARED runs in time out.
- (3) NMT Stop switches the devices of the CAN segment into the state PREPARED. In this state the permanent settings of the devices can be changed (per device Bit rate, Set voltage, Set current, Ramp speed, General status, Threshold to arm the errors detection, Discharge relay configuration, CAN message configuration and additional the Bit rate as a broadcast message).
- (4) NMT Start takes the devices of the CAN segment back to the OPERATIONAL state.
- (5) With the special Flash programming access the device runs into the state FLASH PROGRAMMING. The high voltage will be switched off automatically before.
- (6) The device will execute a POWER ON reset itself at the end of FLASH PROGRAMMING.
- (7) The state CALIBRATION will be obtained by setting of the corresponding switches at the Calibration Crate.
- (8) The state FLASH Programming will be obtained also if the corresponding switch at the Calibration Crate / Flash Programming Slot are set.

Appendix C – Programming flowchart to store the settings permanently with help of General state save bit



## Appendix D – Programming flowchart to store the configurations of the module permanently with help of General state save bit

