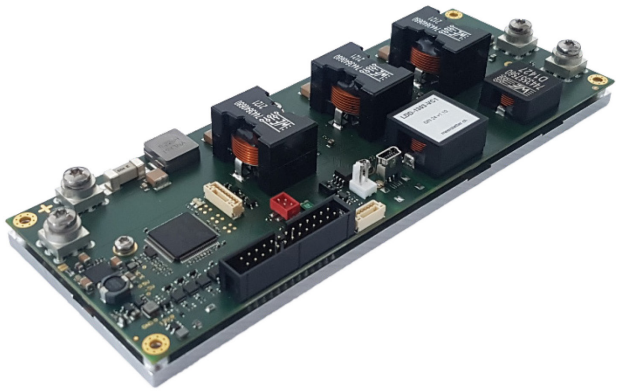
Communication Protocol – LDD-130x



LDD-130x:

LDD-1301
LDD-1303

meerstetter engineering

Member of Berndorf Group

Developed, assembled, and tested in Switzerland

Index

Meerstetter Engineering GmbH
Schulhausgasse 12
CH-3113 Rubigen
Switzerland

Phone: +41 31 529 21 00
Email: contact@meerstetter.ch

Document 5260B

Release date: 24 Feb 2022

1 General Description

If you have any questions, please do not hesitate to contact us.

1.1 Protocol Specifications

- General serial interface specifications: 8 data bits; no parity; 1 stop bit; no handshaking.
- The communication protocol used is based on the “MeCom Protocol Specification 5117D” document.
- Our demo applications and APIs can help you implement this specification. Please also check the example communication strings at the end of this document.
    - The **MeComAPI** (with demo application) shows the full implementation of this protocol for C and C++ applications. Tested on Windows, Linux, and microcontrollers.
    - The **MeComAPI for .Net** (with demo application) is a C# based library that enables communication over RS232/RS485, direct FTDI driver access or Ethernet.

1.2 Addressing

The package format contains an address field, to address up to 254 devices on the same bus. The device reacts in the following cases:

- The device receives a package that matches to the user configurable device address.
    - The device address can be configured and is nonvolatile.
    - An offset can be added to the configured address by using the GPIO pins. This allows the user to have the same configuration on more than one device on a bus. The individual address is then set by hardware-bridged pins on the GPIO pins.
- The device receives a package with the address 0 (broadcast).
- The device receives a package with the address 255. Similar to 0, but the device does not send an answer back to the host. This can for example be used to change the serial speed on all devices connected to the bus.

1.3 Connecting Service Software

- By default, the Service Software always sends to each FTDI USB interface the “?IF” command with 57600 Baud and checks if it gets the correct answer.
- If an answer has been received, then it checks the firmware version.
- If the firmware version matches, then it uses the CS (Change Speed) command to temporarily change the speed from 57600 to 1M Baud. Please have a look at the 2 tables on the next page.
- Using the factory default settings, it is possible to connect the Service Software on each interface (USB, RS232 TTL, RS485).
- If the Base Baud Rate has been changed to a different value, it is possible to change the Service Software behavior by passing some additional startup parameters. It is recommended to create a shortcut to the .exe file and then modify the shortcut parameters. Startup parameters:
    - Service_LDD-130x.exe /LoSpeed 115200
      Sets the initial speed to a certain Baud Rate. The default value is 57600
    - Service_ LDD-130x.exe /HiSpeed 921600
      Sets the high speed value to a certain Baud Rate. The default value is 1000000
    - Service_ LDD-130x.exe /Speed 115200:
      Changes both the LoSpeed and the HiSpeed to a certain Baud Rate.
    - Service_ LDD-130x.exe /IP 192.168.1.191 /DevAddr 3
      Sets the Mode to Ethernet and uses the given IP Address and Device Address to connect.
- To use RS485 we recommend the following interface cable:
  http://www.ftdichip.com/Products/Cables/USBRS485.htm
- To use RS232 TTL we recommend the 3.3V versions of the following interface cable:
  http://www.ftdichip.com/Products/Cables/USBTTLSerial.htm

1.4 Interfaces, Baud Rate

The following table shows the available interfaces for the different products All interfaces support the same commands. It is also possible to use several interfaces simultaneously, for example one interface to connect the Service Software and a second one to connect a custom application.

	LDD-1301, LDD-1303
Interface 1	USB UART MODE 3
Interface 2	RS485 Channel 1 UART MODE 2
Interface 3	RS232 Channel 2 UART MODE 2

UART MODEs

	Primary Base Baud Rate	**Secondary Baud Rate**	**Description**
UART MODE 1	Fixed to 57600	None	Fixed to 57600, but can temporarily be changed with the CS command to a different value. Falls back to 57600 if no data has been received for more than 5s. Is always compatible with the Service Software.
UART MODE 2	Configurable	None	Configurable Primary Base Baud Rate, but can temporarily be changed with the CS command to a different value. Falls back to Primary Base Baud Rate if no data has been received for more than 5s. Is only compatible with the Service Software when the Primary Base Baud Rate is set to 57600 or Startup Parameters are used.
UART MODE 3	Configurable	57600	Configurable Primary Base Baud Rate, but can temporarily be changed with the CS command to a different value. Falls back to Primary Base Baud Rate if no data has been received for more than 5s. A secondary UART is always listening on this interface with 57600 and therefore it is always compatible with the Service Software.

1.5 Flash Parameters (non-volatile) / RAM Parameters (volatile)

Most of the Parameters are saved to the flash after a delay of 0.5 seconds, after the last parameter has been modified. The flash can only be re-written about 100’000 times. This means it is not recommended to write regularly to a parameter that gets saved to the flash. There are 2 possibilities to solve this problem:

- Disabling the automatic save-to-flash mechanism: We recommend to setup the device how you desire, then switch off the saving to the flash. This way, the device has always the desired default configuration after startup. Please consult the User Manual for more information about this option.
- The use of special parameters that are only saved in the RAM. Search for “volatile parameter” in this document.

2 LDD-130x Commands

2.1 Set Commands

Here only the device-specific commands are listed. The standard commands are specified in the document 5117.

Command	Mnemonic	Arguments			
		Type	Min	Max	Description
Emergency Stop	ES	-	-	-	Disables all Power Outputs immediately and generates Error 11.
Change Speed	CS	INT32	4800	1M	Changes the speed of the communication interface where this command was received. If no communication is detected on the interface for more than 5s, the speed is changed back to the Base Baud Rate. The address is ignored when detecting communication on the interface, just a valid package frame is necessary.

2.2 Query Commands

Here only the device-specific commands are listed. The standard commands are specified in the document 5117.

The firmware identification string of LDD-130x devices is: "`8144-LDD-130X G1 `" (20 Chars).

Request	**Mnemonic**	**Description**	**Server Response**	
			Type	**Description**
Bootloader Control	?BC	Control the bootloader	UINT32	See 4 Bootloader for details
Bootloader Stream	?BS	Bootloader data stream		See 4 Bootloader for details
Settings Download	?SD	Can be used to download the exported Settings Dump (*.mepar) of the Service Software.		
		One line of the settings dump File (*.mepar)	UINT4	0: Parameter accepted 1: CRC incorrect. Possible causes: The *.mepar file has been modified The firmware version is not exactly the same as it was while the *.mepar file has been created The *.mepar file was created for another device.

3 Service Software Parameters

3.1 Payload Format Description

The parameter instance is used to control multiple instances of the same feature, if available on a device. If a feature is present only once, it will be controlled through instance 1. For example, the LDD-130x devices have a single current driver channel, so e.g. the target current only has one instance.

See document 5117 for more information about the MeCom protocol.

3.2 Parameter List

3.2.1 Common Product Parameters

3.2.1.1 Device Identification

ID	Name	Format	Unit/Range	Description
100	Device Type	INT32		E.g. 1303 → LDD-1303
101	Hardware Version	INT32		E.g. 123 → 1.23
102	Serial Number	INT32		
103	Firmware Version	INT32		E.g. 123 → 1.23
104	Device Status	INT32		0: Init 1: Ready 2: Run 3: Error 4: Bootloader 5: Device will reset within next 200ms
105	Error Number	INT32		
106	Error Instance	INT32		
107	Error Parameter	INT32		

3.2.1.2 Flash

ID	Name	Format	Unit/Range	Description
108	Save Data to Flash	INT32		0: Enabled 1: Disabled (all Parameters can then be used as volatile Parameters)
109	Parameter System: Flash Status	INT32		0: All Parameters are saved to Flash 1: Save to flash pending or in progress (please do not power off the device during this process). 2: Saving to Flash is disabled

Hint: Parameter 100-999 are read-only, except Parameter 108.

3.2.2 Tab: Monitor (read-only)

3.2.2.1 Output Stage Monitoring

ID	Name	Format	Unit/Range	Description
1100	Actual Output Current	FLOAT32	A	
1101	Actual Output Voltage	FLOAT32	V	

3.2.2.2 External Temperature Measurement x

ID	Name	Format	Unit/Range	Description
1200	Temperature	FLOAT32	°C	
1201	Resistance	FLOAT32	Ω	
1202	Raw ADC Value	FLOAT32		

3.2.2.3 Power Stage Phase Monitoring

ID	Name	Format	Unit/Range	Description
1300	Phase Current x	FLOAT32	A	Only used in LDD-1303, always NA or 0 in LDD-1301
1301	Phase Symmetrization Factor x	FLOAT32		Only used in LDD-1303, always NA or 0 in LDD-1301

3.2.2.4 Internal Parameters

ID	Name	Format	Unit/Range	Description
50000	Volatile Output Enable	FLOAT32		
50001	Volatile Nominal Output Current	FLOAT32	A	
1402	Nominal Output Current (Ramp)	FLOAT32	A	
1403	Output Level	FLOAT32	%	
1404	Calculated Input Current	FLOAT32	A	
1405	Calculated Output Current	FLOAT32	A	

3.2.2.5 Power Stage Temperature Monitoring

ID	Name	Format	Unit/Range	Description
1302	Temperature Phase x Buck/Boost	FLOAT32	°C	Only one value used in LDD-1301, other values are NA. All values used in LDD-1303

3.2.2.6 Analog Input

ID	Name	Format	Unit/Range	Description
1500	Analog Voltage Input	FLOAT32	V	
1501	Photodiode Input	FLOAT32	mA	

3.2.2.7 Firmware and Hardware Versions

ID	Name	Format	Value Range	Description
1050	Firmware Version	INT32		123 → 1.23
1051	Firmware Build Number	INT32		
1054	Min Version for Firmware Downgrade	INT32		123 → 1.23
1052	Hardware Version	INT32		123 → 1.23
1053	Serial Number	INT32		

3.2.2.8 Power Supplies and Temperature

ID	Name	Format	Unit/Range	Description
1060	Device Input Voltage	FLOAT32	V	
1061	12V Internal Supply	FLOAT32	V	
1062	5V Internal Supply	FLOAT32	V	
1063	3.3V Internal Supply	FLOAT32	V	
1064	-5V Internal Supply	FLOAT32	V	
1065	Device Temperature	FLOAT32	°C	

3.2.2.9 Error Status

ID	Name	Format	Unit/Range	Description
1070	Error Number	INT32		
1071	Error Instance	INT32		
1072	Error Parameter	INT32		

3.2.2.10 Driver Status

ID	Name	Format	Unit/Range	Description
1080	Driver Status	INT32		0: Init 1: Ready 2: Run 3: Error 4: Bootloader 5: Device will Reset within the next 200ms
1081	Parameter System Flash Status	INT32		0: All Parameters are saved to flash 1: Save to flash pending or in progress (please do not power off the device during this process). 2: Saving to flash is disabled

3.2.3 Tab: Operation

3.2.3.1 Input Source Selection

ID	Name	Format	Unit/Range	Description
2100	Output Enable	INT32		0: Static OFF 1: Static ON 2: Volatile 3: GPIO
2101	Nominal Output Current	INT32		0: Set Current 1: Volatile

3.2.3.2 Nominal Output Current Values

ID	Name	Format	Unit/Range	Description
2102	Set Current	FLOAT32	A	

3.2.3.3 Current Controller Settings

ID	Name	Format	Unit/Range	Description
2110	PID Kp	FLOAT32	%/A	
2111	PID Ti	FLOAT32	s	
2112	PID Td	FLOAT32	s	
2113	Slope Limit	FLOAT32	A/s	

3.2.3.4 Output Stage Limits

ID	Name	Format	Unit/Range	Description
2122	Max Nominal Current	FLOAT32	1303: 0 … 20A	
2123	Min Nominal Current	FLOAT32	1303: 0 … 20A	
2120	Current Error Threshold	FLOAT32	A	
2121	Voltage Error Threshold	FLOAT32	V	

3.2.3.5 Laser Diode Characteristics

ID	Name	Format	Unit/Range	Description
2130	Slope Compensation Factor	FLOAT32	0 … 1	
2131	Max Diode Current	FLOAT32	0 … 100A	Max Current from Diode Datasheet

3.2.3.6 Device Address

ID	Name	Format	Unit/Range	Description
2051	Device Address	INT32	0 … 254	

3.2.3.7 UART Interface Settings

ID	Name	Format	Unit/Range	Description
2050	Base Baud Rate	INT32	4800 … 1M	Instance 1 = Interface 1 etc.
2052	Response Delay	INT32	0µs … 1E6µs	Instance 1 = Interface 1 etc.

3.2.3.8 Communication Watchdog

ID	Name	Format	Unit/Range	Description
2060	Timeout	FLOAT32	0.1 … 600s	0: Disable the watchdog

3.2.4 Tab: External Temperature

3.2.4.1 External Temperature Measurement Settings x

ID	Name	Format	Unit/Range	Description
5001	Temperature Offset	FLOAT32	°C	
5002	Temperature Gain	FLOAT32	°C/°C	

3.2.4.2 External Temperature Error Limits x

ID	Name	Format	Unit/Range	Description
5011	Upper Error Threshold	FLOAT32	°C	
5010	Lower Error Threshold	FLOAT32	°C	

3.2.4.3 External Temperature Errors Enable x

ID	Name	Format	Unit/Range	Description
5030	ADC Limit Errors	INT32		0: None 1: Upper Only 2: Lower Only 3: Both
5031	Temperature Limit Errors	INT32		0: None 1: Upper Only 2: Lower Only 3: Both

3.2.4.4 External Temperature Measurement Limits x

ID	Name	Format	Unit/Range	Description
5040	Lowest Resistance	FLOAT32	Ω	
5041	Highest Resistance	FLOAT32	Ω	
5042	Temperature at Lower Resistance	FLOAT32	°C	
5043	Temperature at Highest Resistance	FLOAT32	°C	

3.2.5 Tab: Analog Interfaces

ID	Name	Format	Unit/Range	Description
7000	Signal Source	INT32		0: Set Value 1: Sync
7001	Set Value	FLOAT32	-0.5V … 10.5V	
7002	Sync Scaling	FLOAT32	V/A	

3.2.6 Tab: Advanced / External Temperature Measurement

3.2.6.1 External Temperature ADC Calibration x

ID	Name	Format	Unit/Range	Description
5100	Offset	FLOAT32		
5101	Gain	FLOAT32		

3.2.6.2 External NTC Sensor Characteristics x

ID	Name	Format	Unit/Range	Description
5020	Upper Point: Temperature	FLOAT32		
5021	Upper Point: Resistance	FLOAT32		
5022	Middle Point: Temperature	FLOAT32		
5023	Middle Point: Resistance	FLOAT32		
5024	Lower Point: Temperature	FLOAT32		
5025	Lower Point: Resistance	FLOAT32		

3.2.7 Tab: Advanced / LDD Measurement

3.2.7.1 Current Calibration

ID	Name	Format	Unit/Range	Description
8000	Offset	FLOAT32		
8001	Gain	FLOAT32		

3.2.7.2 Voltage Calibration

ID	Name	Format	Unit/Range	Description
8002	Offset	FLOAT32		
8003	Gain	FLOAT32		

3.2.8 Tab: Advanced / Analog Interfaces

3.2.8.1 Analog Output DAC Calibration

ID	Name	Format	Unit/Range	Description
9000	Offset	FLOAT32		
9001	Gain	FLOAT32		

3.2.9 Tab: Advanced / GPIO

3.2.9.1 GPIO General / GPIO Configuration (GPIO1 … GPIO10)

The instance number corresponds to the GPIO number.

ID	Name	Format	Unit/Range	Description
6100	GPIO Function	INT32		0: No Function 1: Signal Control (see 3.2.12.2) 2: LDD OK (1 when Ready or Running) 3: No Function 4: HW Enable 5: No Function 6: No Function 7: No Function 8: LDD Run 9: No Function 10: No Function 11: No Function 12: No Function 13: Adr +1 (Device Address) 14: Adr +2 (Device Address) 15: Adr +4 (Device Address) 16: No Function 17: No Function 18: No Function 19: No Function 20: No Function 21: No Function 22: Pulse Input
6101	GPIO Level Assignment	INT32		0: Positive 1: Negative
6102	GPIO Hardware Configuration	INT32		0: IN WeakNo 1: IN WeakUp 2: IN WeakDown 3: OUT PushPull 4: OUT OD NoPull 5: OUT OD WeakUp
6103	GPIO Channel	INT32		1: Channel 1 … 10: Channel 10

3.2.9.2 GPIO Detail

3.2.10 Tab: Advanced / Temperature Correction

3.2.10.1 Temperature Correction Settings

ID	Name	Format	Unit/Range	Description
6110	Source	INT32		0: OFF 1: Ext Temp 1 2: Ext Temp 2
6111	Offset [°C]	FLOAT32	°C	
6112	Gain [A/°C]	FLOAT32	A/°C	

3.2.11 Tab: Advanced / Misc

3.2.11.1 Parameter System Save to Flash Configuration

ID	Name	Format	Unit/Range	Description
108	Save Data to Flash	INT32		0: Enabled 1: Disabled

3.2.11.2 Error State Auto Reset Delay

ID	Name	Format	Unit/Range	Description
6310	Delay until Reset	FLOAT32	0s … 86400s	0: No Auto Reset

3.2.12 Other Parameters (not displayed in thet Service Software)

3.2.12.1 Driver Parameters

ID	Name	Format	Unit/Range	Description
50000	Volatile Output Enable	INT32		0: Static OFF 1: Static ON
50001	Volatile Set Current	FLOAT32		

3.2.12.2 GPIO Signal Control

The following parameters are volatile parameters.

This feature can be used to control the GPIO signals. The pins are addressed by a bit field.

Example:
To configure GPIO3 / GPIO4 as output pins, and to set GPIO3 to high level and GPIO4 to low level, use the following commands:

Set ID 52102 to 4 (sets bit #2 to '1')
Set ID 52101 to 12 (sets bits #2 and #3 to '1')
Set ID 52100 to 1 (enables the function)

Bit Field Description

Bit Number	Output Signal
0	GPIO1
⋮	⋮
10	GPIO2

This command order has been chosen to avoid spikes. After reset, all values are set to 0.

ID	Name	Format	Unit/Range	Description
52100	Enable Function	INT32	0 … 1	Enable the output signal control function.
52101	Set Output to Push-Pull	INT32	0 … 255	If a bit is set to 0, the output signal is at high impedance s(set as input). If a bit is set to 1, the output signal is driven.
52102	Set Output States	INT32	0 … 255	Set the output states of driven signals.
52103	Read Input States	INT32	0 … 255	Read back the (input) states of all signals.

4 Bootloader

- The firmware can be downloaded using all communication interfaces.
- The Bootloader can be controlled with the Control and Stream commands.
- The whole new firmware will be cached and verified onboard, before the old firmware gets erased.
- Do not interrupt the power after the ReBoot command, until the device answers to your queries again. Interrupting the power after the ReBoot command can cause irreversible memory errors.
- Do not worry about communication failures, wrong command sequences or downloading the wrong firmware using this bootloader. Everything is verified, before the bootloader enters the critical process. If the power is not interrupted after the ReBoot command, no irreversible errors are possible.

It's necessary to use the correct command sequence:

1. Activate the bootloader using the **BootloaderActivate** command.
2. Query the status using **NoOperation** and wait until it reports the "activated" status.
3. Clear the update memory using the **ClearMemory** command**.**
4. Query the status using **NoOperation** and wait until it reports the "memory is cleared" status.
5. Send the whole *.hex* file using the Bootloader Stream command.
6. Query the status using **NoOperation** and wait until it reports the "valid application" status.
7. Start replacing the running application by sending the **ReBoot** command. Attention:
    a. The device will typically not answer your queries for 10s (it may take longer).
    b. Do not interrupt the power during this process!
    c. Constantly send queries (e.g. ?IF) to the device, to see if the new firmware is already running. If the device is answering your queries again, the critical process has ended.
8. Verify if the firmware version is as expected.

4.1 Bootloader Control (BC?)

Type	Mnemonic	Field 1
Query	?BC	UINT32 Bootloader Command

Type	Field 1
Response	UINT32 Bootloader Status or Server Error Code

4.1.1 Bootloader Command

Bit	Value	Description
NoBit	0x00	(No bit set) **NoOperation**. Can be used to read the Bootloader Status.
0	0x01	**BootloaderActivate**. Starts enabling the update memory.
1	0x02	**ClearMemory**. Starts clearing the update memory.
2	0x04	**ReBoot**. Starts replacing the running application. Will only be accepted if the Bootloader Status reports “Valid Application”.

4.1.2 Bootloader Status

Bit	Value	Description
0	0x0001	Bootloader is activated.
1	0x0002	Memory is cleared.
2	0x0004	Valid application. There is a valid application in the update memory.
3	0x0008	Bootloader error. This flag will always be set if an error occurs. See additional flags below (FW >= v5.00).
4	0x0010	Error: CRC error detected in the downloaded file.
5	0x0020	Error: the firmware identification string does not match. The downloaded firmware is not intended for this device.
6	0x0040	Error: this firmware is not made for this firmware branch.
7	0x0080	Error: the firmware is too old for this device.
8	0x0100	Error: AES decryption failure. The code was encrypted with a different key.
9	0x0200	Error: this firmware is too new for the currently installed firmware version. Please check the Software Release Notes. An intermediate firmware might be needed.
10	0x0400	Error: unencrypted firmware file detected. It is not possible to install this firmware anymore. Please contact the manufacturer.
11	0x0800	Error: firmware update limit reached. The downloaded file contains a firmware that is too old for the device.
12	0x1000	Error: firmware update limit reached. The downloaded file contains a firmware that is too new for the device.

4.2 Bootloader Stream (BS?)

Type	Mnemonic	Field 1	Field 2
Query	?BS	UINT32 Length in bytes of field 2	Data Stream Part of the Hex File

Type	Field 1
Response	UINT32 Bootloader Status or Server Error Code

4.2.1 Data Stream

The Data Stream command is used to send the hex file content to the microcontroller.

Add a few hex file lines to the payload field of the communication protocol frame and remove all '\n' and '\r' from the stream (the hex file lines are then only separated by the double dot).

The maximum size of the Payload Field is 512 Bytes.

It is recommended to send 10 hex file lines in one package. This will not exceed the 512 Byte limit.

4.2.2 Bootloader Status

See 4.1.2 Bootloader Status.

5 Example Communication Strings

- The following Example Communication Strings have been captured with the MeComAPI ComLog.txt file.
- It shows the Serial Communication Data as it would appear on a normal Serial Terminal Program. Only the “OUT:” and “IN:” tags have been added by the MeComAPI. The End-of-Frame Byte is not shown, because it is an ASCII <CR> (Carriage Return, 0x0D).
- The Checksum is calculated using the CRC16/XMODEM algorithm over the previous frame data as ASCII String.
- All the Frame data is colored to better understand what is going on:
  - Control
  - Address (using address 0, the device will always answer independently from its address.)
  - Sequence Number
  - Payload / Other Payload
  - Checksum

Get Firmware Identification String

OUT: #001EF8?IFF1E4
IN: !001EF88144-LDD-130X G1 CED8

→ Result: “8144-LDD-130X G1 “

Get Device Type (Using Parameter Value Read)

Parameter ID: 100 (0x0064); Instance 1

OUT: #000F24?VR0064012B1A
IN: !000F2400000517EABE

→ Result: 0x00000517 → 1303

Get Serial Number (Using Parameter Value Read)

Parameter ID: 102 (0x0066); Instance 1

OUT: #0015AC?VR0066018125
IN: !0015AC00000706F2C

→ Result: 0x00000070 → Interpreted as an INT32: decimal value 112

Querying a not available Parameter ID (Using Parameter Value Read)

Parameter ID: 1234 (0x04D2); Instance 1

OUT: #0015AC?VR04D2017BFE
IN: !0015AC+0532DA

→ As a result we get the Server Error Code 0x05 which means that this parameter is not available.

You can use the tool: http://www.h-schmidt.net/FloatConverter/ for tests involving FLOAT32 parameters.
Usually, microcontrollers support float according to IEEE754 by Hardware or Software FPU.

A Change History

Date of change	Version	Changed/ Approved	Change / Reason
20 December 2021	5260A	RS/HS	• First edition
24 February 2022	5260B	HS/PV	• Comments added to phase current, balancing factors, and power stage temperatures because some values are not used in the LDD-1301