LineLidar class

Generated by Doxygen 1.9.1

2.1 Modules       3         3 Namespace Index       5         3.1 Packages       5         4 Hierarchical Index       7         4.1 Class Hierarchy       7         5 Class Index       9         5.1 Class List       9         6 Module Documentation       11         6.1 Base classes       11         6.1.1 Detailed Description       11         6.2 Enums       11         6.2.1 Detailed Description       11         6.3 Main classes       11         6.3 Nain classes       11         6.5 Routines       11         7 Namespace Documentation       11         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.default Namespace Reference       13         7.3.1 Function Documentation       14         7.3.1.1 discover()       14	1 Python3 LineLidar class	1
2 Module Index       3         2.1 Modules       3         3 Namespace Index       5         3.1 Packages       5         4 Hierarchical Index       7         4.1 Class Hierarchy       7         5 Class Index       9         5.1 Class List       5         6 Module Documentation       11         6.1 Detailed Description       11         6.2 Installed Description       11         6.2.1 Detailed Description       11         6.3 Main classes       11         6.3.1 Detailed Description       11         6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass default Namespace Reference       13         7.2 linelidarclass default Namespace Reference       14         7.3 linelidarclass. default Namespace Reference       14         7.3.1 runction Documentation       14         7.3.1 runction Documentation       15         8.1.2.1 mit_0       15         8.1.3 Member Function Documentation       16         8.1.3.1 mit_0       16         8.1.3.2 mash_0       16         8.1.3.3 setatr_0       16 <tr< th=""><th>1.1 Description</th><th> 1</th></tr<>	1.1 Description	1
2.1 Modules	1.2 Notes	1
3 Namespace Index 3.1 Packages 5.4 Hierarchical Index 4.1 Class Hierarchy 7.5 Class Index 5.1 Class List 5.6 Module Documentation 6.1 Base classes 6.1.1 Detailed Description 6.2 Enums 6.2.1 Detailed Description 6.2.1 Detailed Description 6.3 Main classes 6.3.1 Detailed Description 6.3.4 Detailed Description 6.4 Default parameters 6.5 Routines 7.1 linelidarclass Almespace Reference 7.2 linelidarclass Mamespace Reference 7.3 linelidarclass Mamespace Reference 7.3.1 inclidarclass Minelidar Namespace Reference 7.3.1 Function Documentation 7.3.1.1 discover() 8.1.2 Constructor & Destructor Documentation 8.1 LLChr Class Reference 8.1.2.1init() 8.1.3aq() 8.1.3aq() 8.1.3ata() 8.2	2 Module Index	3
3.1 Packages 5.5 4 Hierarchical Index 7.7 4.1 Class Hierarchy 7.7 5 Class Index 9.5 5 Class List 9.5 6 Module Documentation 11 6.1 Base classes 11 6.1.1 Detailed Description 11 6.2 Enums 11 6.2.1 Detailed Description 11 6.3 Main classes 11 6.3.1 Detailed Description 11 6.4 Default parameters 11 6.5 Routines 11 7 Namespace Documentation 13 7.1 linelidarclass Namespace Reference 15 7.2 linelidarclass default Namespace Reference 15 7.3 linelidarclass linelidar Namespace Reference 14 7.3.1 Function Documentation 14 7.3.1.1 discover() 14 8 Class Documentation 15 8.1_LLchr Class Reference 15 8.1.2 Linit_() 15 8.1.3 Member Function Documentation 16 8.1.2.1_init_() 15 8.1.3 Member Function Documentation 16 8.1.3.1_eq_() 16 8.1.3.2_hash_() 16 8.2_LLcmd Class Reference 17 8.2.1 Lotaled Description 16 8.1.3.3_setattr_() 16 8.2_LLcmd Class Reference 17 8.2.1 Lotaled Description 16 8.3_L2_LLcmd Class Reference 17 8.3_L1 Detailed Description 16 8.4_L2_hash_() 16 8.4_L2_LLcmd Class Reference 17 8.4_L1 Detailed Description 17	2.1 Modules	3
4 Hierarchical Index 4.1 Class Hierarchy 5 Class Index 5.1 Class List 6 Module Documentation 6.1 Base classes 6.1.1 Detailed Description 6.2 Enums 6.2.1 Detailed Description 6.3 Main classes 6.2.1 Detailed Description 6.3 Main classes 6.3.1 Detailed Description 6.4 Default parameters 6.5 Routines 11  7 Namespace Documentation 7.1 linelidarclass Namespace Reference 7.2 linelidarclass Namespace Reference 13 7.2 linelidarclass.linelidar Namespace Reference 13 7.3 Incurrence 14 7.3.1 Function Documentation 15 8.1_LLchr Class Reference 8.1.1 Detailed Description 8.1_L Constructor & Destructor Documentation 15 8.1_2 Constructor & Destructor Documentation 16 8.1.2.1init() 17 8.1.3 Member Function Documentation 18 8.1.3.3eat_in_() 19 8.2_LLcmd Class Reference 17 8.2.1 Detailed Description 17	3 Namespace Index	5
4.1 Class Hierarchy 7  5 Class Index 9  5.1 Class List 9  6 Module Documentation 111  6.1 Base classes 112  6.1.1 Detailed Description 113  6.2 Enums 116  6.2.1 Detailed Description 111  6.3 Main classes 113  6.3.1 Detailed Description 111  6.4 Default parameters 113  6.5 Routines 113  7 Namespace Documentation 113  7.1 linelidarclass Namespace Reference 113  7.2 linelidarclass.default Namespace Reference 113  7.3 linelidarclass.linelidar Namespace Reference 114  7.3.1 Function Documentation 114  7.3.1.1 discover() 115  8 Class Documentation 115  8.1.2 Constructor & Destructor Documentation 115  8.1.2 Constructor & Destructor Documentation 116  8.1.2.1init() 115  8.1.3 Member Function Documentation 116  8.1.3.1eq() 116  8.1.3.2hash() 116  8.2_LLcmd Class Reference 177  8.2.1 Detailed Description 177	3.1 Packages	5
5 Class Index       9         5.1 Class List       9         6 Module Documentation       11         6.1 Base classes       11         6.1 Detailed Description       11         6.2 Enums       11         6.2 In Detailed Description       11         6.3 Main classes       11         6.3 Detailed Description       11         6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass Junelidar Namespace Reference       13         7.3 linelidarclass Junelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1.2 Llchr Class Reference       15         8.1.2.1init()       15         8.1.2.1init()       15         8.1.3.3ebash()       16         8.1.3.3ebash()       16         8.2.1 Letailed Description       17	4 Hierarchical Index	7
5.1 Class List       9         6 Module Documentation       11         6.1 Base classes       11         6.1 Detailed Description       11         6.2 Enums       11         6.2.1 Detailed Description       11         6.3 Main classes       11         6.3.1 Detailed Description       11         6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1.2 Lbchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.3 Member Function Documentation       16         8.1.3.1 _eq_()       16         8.1.3.2 _hash_()       16         8.2.1 Lotted Class Reference       17         8.2.1 Detailed Description       17          8.2.1 Detailed Description       17     <	4.1 Class Hierarchy	
6 Module Documentation 6.1 Base classes 6.1.1 Detailed Description 6.2 Enums 6.2.1 Detailed Description 6.2.1 Detailed Description 6.3 Main classes 6.3.1 Detailed Description 6.3 Main classes 6.3.1 Detailed Description 6.4 Default parameters 6.5 Routines 7 Namespace Documentation 7.1 linelidarclass Namespace Reference 7.2 linelidarclass Namespace Reference 7.3 linelidarclass.linelidar Namespace Reference 7.3.1 Function Documentation 7.3.1.1 discover() 8 Class Documentation 15 8.1_LLchr Class Reference 8.1.1 Detailed Description 16 8.1_2 Constructor & Destructor Documentation 17 8.1.2 Constructor & Destructor Documentation 18 8.1.3 Member Function Documentation 19 8.1.3 Member Function Documentation 19 8.1.3.1eq_() 10 8.1.3.2hash() 11 8.2_LLcnd Class Reference 17 8.2.1 Detailed Description 17	5 Class Index	ç
6.1 Base classes       11         6.1.1 Detailed Description       11         6.2 Enums       11         6.2.1 Detailed Description       11         6.3 Main classes       11         6.3.1 Detailed Description       11         6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass. Namespace Reference       13         7.2 linelidarclass. Jefault Namespace Reference       13         7.3 linelidarclass. Jinelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17          8.2.1 Detailed Description       17	5.1 Class List	
6.1.1 Detailed Description       11         6.2 Enums       11         6.2.1 Detailed Description       11         6.3 Main classes       11         6.3.1 Detailed Description       11         6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.3 Member Function Documentation       15         8.1.3.1eq()       16         8.1.3.2hash()       16         8.2_LLcmd Class Reference       17         8.2_LLcmd Class Reference       17         8.2.1 Detailed Description       17	6 Module Documentation	11
6.2 Enums       11         6.2.1 Detailed Description       11         6.3 Main classes       11         6.3.1 Detailed Description       11         6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2_LLcmd Class Reference       17         8.2.1 Detailed Description       17	6.1 Base classes	11
6.2 Enums       11         6.2.1 Detailed Description       11         6.3 Main classes       11         6.3.1 Detailed Description       11         6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2_LLcmd Class Reference       17         8.2.1 Detailed Description       17	6.1.1 Detailed Description	11
6.2.1 Detailed Description	•	
6.3 Main classes       11         6.3.1 Detailed Description       11         6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass s.linelidar Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1_LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2_LLcmd Class Reference       17         8.2.1 Detailed Description       17		
6.3.1 Detailed Description       11         6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1_LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2LLcmd Class Reference       17         8.2_1 Detailed Description       17		
6.4 Default parameters       11         6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       14         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1_LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2_LLcmd Class Reference       17         8.2_LLcdd Description       17		
6.5 Routines       11         7 Namespace Documentation       13         7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1_LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2_LLcmd Class Reference       17         8.2_1 Detailed Description       17	·	
7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1 _LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17	·	
7.1 linelidarclass Namespace Reference       13         7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1 _LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17	7 Namespace Documentation	13
7.2 linelidarclass.default Namespace Reference       13         7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1 _LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2LLcmd Class Reference       17         8.2.1 Detailed Description       17	•	13
7.3 linelidarclass.linelidar Namespace Reference       14         7.3.1 Function Documentation       14         7.3.1.1 discover()       14         8 Class Documentation       15         8.1 _LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2LLcmd Class Reference       17         8.2.1 Detailed Description       17		
7.3.1 Function Documentation 14 7.3.1.1 discover() 14  8 Class Documentation 15 8.1_LLchr Class Reference 15 8.1.1 Detailed Description 15 8.1.2 Constructor & Destructor Documentation 15 8.1.2.1init() 15 8.1.3 Member Function Documentation 16 8.1.3.1eq() 16 8.1.3.2hash() 16 8.1.3.3setattr() 16 8.2_LLcmd Class Reference 17 8.2.1 Detailed Description 17	•	
7.3.1.1 discover()       14         8 Class Documentation       15         8.1 _ LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1 _ init _ ()       15         8.1.3 Member Function Documentation       16         8.1.3.1 _ eq _ ()       16         8.1.3.2 _ hash _ ()       16         8.1.3.3 _ setattr _ ()       16         8.2 _ LLcmd Class Reference       17         8.2.1 Detailed Description       17		
8.1 _ LLchr Class Reference       15         8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1 _ init _ ()       15         8.1.3 Member Function Documentation       16         8.1.3.1 _ eq _ ()       16         8.1.3.2 _ hash _ ()       16         8.1.3.3 _ setattr _ ()       16         8.2 _ LLcmd Class Reference       17         8.2.1 Detailed Description       17		
8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17	8 Class Documentation	15
8.1.1 Detailed Description       15         8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17	8.1 LLchr Class Reference	15
8.1.2 Constructor & Destructor Documentation       15         8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17		
8.1.2.1init()       15         8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17	·	
8.1.3 Member Function Documentation       16         8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17		
8.1.3.1eq()       16         8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17		
8.1.3.2hash()       16         8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17		
8.1.3.3setattr()       16         8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17		
8.2 _LLcmd Class Reference       17         8.2.1 Detailed Description       17		
8.2.1 Detailed Description		
·		
	•	

8.3.1 Detailed Description	18
8.3.2 Constructor & Destructor Documentation	18
8.3.2.1init()	18
8.3.3 Member Function Documentation	19
8.3.3.1 <u>eq</u> ()	19
8.3.3.2hash()	19
8.3.3.3setattr()	19
8.4 _LLsrv Class Reference	20
8.4.1 Detailed Description	20
8.5 _LLsta Class Reference	21
8.5.1 Detailed Description	21
8.6 ipaddress Class Reference	21
8.6.1 Detailed Description	22
8.7 LineLidar Class Reference	22
8.7.1 Detailed Description	23
8.7.2 Constructor & Destructor Documentation	24
8.7.2.1init()	24
8.7.3 Member Function Documentation	25
8.7.3.1decode_msg()	25
8.7.3.2encode_cmd()	25
8.7.3.3highlighted_bytes()	26
8.7.3.4recv_ssh_line()	26
8.7.3.5retry_cmd()	27
8.7.3.6windows_reader_thread()	27
8.7.3.7 _get_cmd_response()	28
8.7.3.8 _recv_msg()	29
8.7.3.9 _send_cmd()	30
8.7.3.10 _send_data()	30
8.7.3.11 disable_all_notifications()	31
8.7.3.12 disable_notification()	31
8.7.3.13 enable_notification()	32
8.7.3.14 get_notification()	32
8.7.3.15 open()	33
8.7.3.16 read_chr()	34
8.7.3.17 report_zero_results()	34
8.7.3.18 reset()	35
8.7.3.19 restore_srv()	35
8.7.3.20 save_srv()	36
8.7.3.21 set_clean_state()	36
8.7.3.22 set_notification()	37
8.7.3.23 set_sampling_rate()	37
8.7.3.24 stop_sampling()	38

8.7.3.25 wait_device_quiet()	38
8.7.3.26 write_chr()	39
8.7.4 Member Data Documentation	39
8.7.4.1 int	39
8.8 LLchr Class Reference	40
8.8.1 Detailed Description	40
8.8.2 Notes	40
8.8.3 Member Data Documentation	40
8.8.3.1 AMPLITUDE_THRESHOLD	40
8.8.3.2 CALIBRATED_ANGLES	41
8.8.3.3 DEFAULT_NETWORK	41
8.8.3.4 FW_VERSION	41
8.8.3.5 MAC	41
8.8.3.6 MAX_ANGLE	42
8.8.3.7 MAX_DISTANCE	42
8.8.3.8 MEASURING_RATE	42
8.8.3.9 MIN_ANGLE	42
8.8.3.10 MIN_DISTANCE	43
8.8.3.11 NB_PEAKS	43
8.8.3.12 NETWORK	43
8.8.3.13 RANGE	43
8.8.3.14 REPORT_ZERO_RESULTS	44
8.8.3.15 RESET_DEVICE	44
8.8.3.16 SERIAL_NUMBER	44
8.8.3.17 TEMPERATURE	44
8.8.3.18 TIME	45
8.8.3.19 TRIGGER_DIVISION	45
8.8.3.20 TRIGGER_SOURCE	45
8.8.3.21 TRIGGER_TYPE	45
8.9 LLcmd Class Reference	46
8.9.1 Detailed Description	46
8.10 LLresponse Class Reference	46
8.10.1 Detailed Description	47
8.10.2 Member Function Documentation	47
8.10.2.1 <u>repr</u> ()	47
8.10.3 Member Data Documentation	47
8.10.3.1 reset	47
8.11 LLsrv Class Reference	48
8.11.1 Detailed Description	48
8.11.2 Notes	48
8.12 LLsta Class Reference	48
8.12.1 Detailed Description	48

Index		51
	8.13.1 Detailed Description	49
8.13	3 macaddress Class Reference	48

# **Python3 LineLidar class**

# 1.1 Description

Low-level class to communicate with a LineLidar device in Python

## 1.2 Notes

Copyright 2023 (c) Noptel Oy, Oulu Finland

Tested on:

- Linux
- Windows

Authors: PCo

# **Module Index**

# 2.1 Modules

Here is a list of all modules:

Base classes									 										 	11
Enums									 										 	11
Main classes									 										 	11
Default parameters									 										 	11
Routines																				11

4 Module Index

# Namespace Index

# 3.1 Packages

Here are the packages with brief descriptions (if available):

linelidarclass	
LineLidar Low-level communication class	 13
linelidarclass.default	
LineLidar Low-level communication class	 13
linelidarclass.linelidar	
LineLidar Low-level communication class	 14

6 Namespace Index

# **Hierarchical Index**

# 4.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

_LLchr											 												15
_LLintEnum											 												18
_LLcmd																			 				 . 17
_LLsrv .																			 				 20
_LLsta .																			 				 21
ipaddress .											 												21
LineLidar																							
LLchr																							
LLcmd																							
LLresponse																							
LLsrv																							
LLsta																							48
macaddress																							48

8 Hierarchical Index

# **Class Index**

# 5.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

_LLchr		15
_LLcmd		
	LineLidar command enum element	17
_LLintEn	ıum	
_LLsrv		
	LineLidar service enum element	20
_LLsta		
	LineLidar status code enum element	21
ipaddres		
	Simple IPv4 address class	21
LineLida		
LLchr	Main LineLidar class	22
LLCIII	Characteristics	40
LLcmd	Characteristics	40
LLCITIC	Commands / command responses	16
LLrespor		70
oopo.	Decoded response class	46
LLsrv		
	Services	48
LLsta		
	Status codes	48
macaddr	ress	
	Simple MAC address class	48

10 Class Index

# **Module Documentation**

### 6.1 Base classes

#### **Classes**

- class ipaddress
- · class macaddress

## 6.1.1 Detailed Description

### 6.2 Enums

#### **Classes**

- class LLcmd
- class LLsrv
- · class LLchr
- class LLsta

## 6.2.1 Detailed Description

### 6.3 Main classes

#### Classes

- class LLresponse
- class LineLidar

### 6.3.1 Detailed Description

Type definitions.

Main classes

## 6.4 Default parameters

## 6.5 Routines

12 Module Documentation

# **Namespace Documentation**

# 7.1 linelidarclass Namespace Reference

### **Namespaces**

- · default
- linelidar

#### **Classes**

- · class ipaddress
- class macaddress
- class \_LLintEnum
- class LLcmd
- class \_LLsrv
- class \_LLsta
- class LLchr
- class LLcmd
- class LLsrv
- class LLchr
- class LLsta
- · class LLresponse

#### **Variables**

• list \_\_all\_\_ = ["default", "linelidar"]

# 7.2 linelidarclass.default Namespace Reference

## **Variables**

- int \_ll\_default\_port = 9907
- float \_ll\_default\_udp\_timeout = 1.5
- int II default ssh timeout = 5
- int \_ll\_default\_reboot\_timeout = 5
- int II default retries = 1
- int \_ll\_default\_sent\_cmds\_log\_depth = 16
- string \_ll\_default\_ssh\_python\_path = "python"

## 7.3 linelidarclass.linelidar Namespace Reference

#### **Classes**

· class LineLidar

#### **Functions**

• List[str] discover (Optional[str] network=None, Optional[int] port=\_ll\_default\_port, Optional[str] sshcmd=None, Optional[str] sshpypath=None, int retries=None, bool debug=False, Optional[float] timeout=None)

### 7.3.1 Function Documentation

#### 7.3.1.1 discover()

Discover LineLidar devices on a network.

#### **Parameters**

network	Network / netmask to discover devices on, in xx.xx.xx/mm format, or None to discover on all
	interfaces
port	Port of devices to discover
sshcmd	If specified, ssh command to log into a shell account to use the host as a relay to talk to the
	LineLidar. Works with a Linux or Windows host with sshd and Python 2 or 3 installed.
sshpypath	Path of the Python executable on the SSH relay host
retries	How many more times the discovery packet should be sent for redundancy
debug	Enable or disable debugging messages
timeout	Communication timeout in seconds

#### Returns

Broadcast a read request for the NETWORK characteristic, wait for replies and return the list of addresses of the devices that replied.

Definition at line 2089 of file linelidar.py.

# **Class Documentation**

# 8.1 \_LLchr Class Reference

#### **Public Member Functions**

```
None __init__ (_LLchr self, str name, Tuple[_LLsrv, int] value)
bool __eq__ (_LLchr self, object other)
None __setattr__ (_LLchr self, str attr, val)
int __hash__ (_LLchr self)
```

#### **Private Attributes**

· \_hash

### 8.1.1 Detailed Description

```
Pseudo-enum element to store a (_LLsrv, int) tuple describing a characteristic
@ingroup BaseClasses

Definition at line 229 of file __init__.py.
```

### 8.1.2 Constructor & Destructor Documentation

Definition at line 244 of file \_\_init\_\_.py.

### 8.1.3 Member Function Documentation

```
8.1.3.1 __eq__()
```

```
bool __eq__ (
    __LLchr self,
    object other )
__eq__ method to compare two enum elements
```

Definition at line 257 of file \_\_init\_\_.py.

### 8.1.3.2 \_\_hash\_\_()

Definition at line 278 of file \_\_init\_\_.py.

### 8.1.3.3 \_\_setattr\_\_()

```
None __setattr__ (
   __LLchr self,
        str attr,
        val )

Disabled setter, as enums are immutable
```

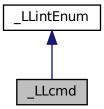
Definition at line 267 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

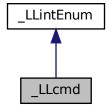
• linelidarclass/\_\_init\_\_.py

## 8.2 LLcmd Class Reference

Inheritance diagram for \_LLcmd:



Collaboration diagram for \_LLcmd:



## **Additional Inherited Members**

### 8.2.1 Detailed Description

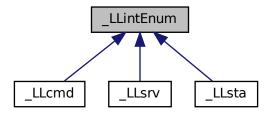
Definition at line 208 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

• linelidarclass/\_\_init\_\_.py

## 8.3 LLintEnum Class Reference

Inheritance diagram for \_LLintEnum:



### **Public Member Functions**

```
• None __init__ (_LLintEnum self, str name, int value)
```

- bool <u>eq</u> (<u>LLintEnum</u> self, object other)
- None <u>\_\_setattr\_\_</u> (<u>\_LLintEnum</u> self, str attr, val)
- int \_\_hash\_\_ (\_LLintEnum self)

### **Public Attributes**

value

### 8.3.1 Detailed Description

```
Pseudo-enum element to store an integer value 
@ingroup BaseClasses
```

Definition at line 153 of file \_\_init\_\_.py.

#### 8.3.2 Constructor & Destructor Documentation

### 8.3.2.1 \_\_init\_\_()

```
None __init__ (
    __LLintEnum self,
    str name,
    int value )
```

Initialize the enum element

Definition at line 166 of file \_\_init\_\_.py.

### 8.3.3 Member Function Documentation

```
8.3.3.1 __eq__()

bool __eq__ (
    __LLintEnum self,
    object other)

__eq__ method to compare two enum elements

Definition at line 178 of file __init__.py.

8.3.3.2 __hash__()

int __hash__ (
```

\_LLintEnum self )

Definition at line 199 of file \_\_init\_\_.py.

### 8.3.3.3 \_\_setattr\_\_()

\_\_hash\_\_ method

```
None __setattr__ (
    __LLintEnum self,
    str attr,
    val )

Disabled setter, as enums are immutable
```

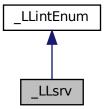
Definition at line 188 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

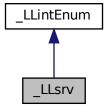
• linelidarclass/\_\_init\_\_.py

## 8.4 LLsrv Class Reference

Inheritance diagram for \_LLsrv:



Collaboration diagram for \_LLsrv:



## **Additional Inherited Members**

## 8.4.1 Detailed Description

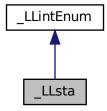
Definition at line 215 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

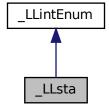
• linelidarclass/\_\_init\_\_.py

## 8.5 LLsta Class Reference

Inheritance diagram for \_LLsta:



Collaboration diagram for \_LLsta:



### **Additional Inherited Members**

## 8.5.1 Detailed Description

Definition at line 222 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

• linelidarclass/\_\_init\_\_.py

# 8.6 ipaddress Class Reference

#### **Public Member Functions**

- None \_\_init\_\_ (ipaddress self, Union[str, bytes] addr)
- bool <u>eq</u> (ipaddress self, object other)
- str \_\_repr\_\_ (ipaddress self)

#### **Public Attributes**

ip

### 8.6.1 Detailed Description

Definition at line 42 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

• linelidarclass/ init .py

#### 8.7 LineLidar Class Reference

#### **Public Member Functions**

- None \_\_init\_\_ (LineLidar self, Optional[str] addr=None, Optional[int] port=None, Optional[str] sshcmd=None,
   Optional[str] sshpypath=None, Optional[int] sent\_cmds\_log\_depth=None, int retries=None, bool set\_clean\_state=True,
   bool autostop\_ranging=True, bool debug=False, Optional[float] timeout=None)
- LineLidar \_\_enter\_\_ (LineLidar self)
- None \_\_exit\_\_ (LineLidar self, Optional[Type[BaseException]] exc\_type, Optional[BaseException] exc\_value, Optional[TracebackType] exc\_traceback)
- None del (LineLidar self)
- Union[socket.socket, Tuple[subprocess.Popen, Optional[ multiprocessing.queues.Queue]]] open (LineLidar self, str addr, Optional[int] port=None, Optional[str] sshcmd=None, Optional[str] sshpypath=None, Optional[int] sent\_cmds\_log\_depth=None, int retries=None, bool set\_clean\_state=True, bool autostop
   \_ranging=True, Optional[float] timeout=None)
- None close (LineLidar self)
- LLresponse read\_chr (LineLidar self, \_LLchr char, int retries=None, bool exc\_on\_nok=True, Optional[float] timeout=None)
- LLresponse write\_chr (LineLidar self, \_LLchr char, int retries=None, bool exc\_on\_nok=True, Optional[float] timeout=None, \*\*\_CHR\_ARGS\_TYPE kwargs)
- LLresponse set\_notification (LineLidar self, \_LLchr char, bool enabled, int retries=None, bool exc\_on\_
   —
   nok=True, Optional[float] timeout=None)
- LLresponse enable\_notification (LineLidar self, \_LLchr char, int retries=None, bool exc\_on\_nok=True, Optional[float] timeout=None)
- LLresponse disable\_notification (LineLidar self, \_LLchr char, int retries=None, bool exc\_on\_nok=True, Optional[float] timeout=None)
- LLresponse disable\_all\_notifications (LineLidar self, bool incl\_restricted=False, int retries=None, bool exc\_
  on\_nok=True, Optional[float] timeout=None)
- LLresponse report\_zero\_results (LineLidar self, bool on, int retries=None, bool exc\_on\_nok=True, Optional[float] timeout=None)
- LLresponse set\_sampling\_rate (LineLidar self, int frequency, int retries=None, bool exc\_on\_nok=True, Optional[float] timeout=None)
- LLresponse stop\_sampling (LineLidar self, int retries=None, bool exc\_on\_nok=True, Optional[float] time-out=None)
- LLresponse save\_srv (LineLidar self, \_LLsrv srv, int retries=None, bool exc\_on\_nok=True, Optional[float] timeout=None)
- LLresponse restore\_srv (LineLidar self, \_LLsrv srv, int retries=None, bool exc\_on\_nok=True, Optional[float] timeout=None)
- None flush\_notifications (LineLidar self)
- LLresponse get\_notification (LineLidar self, Union[List[\_LLchr], Tuple[\_LLchr], None] chrmask=None, Optional[float] timeout=None)
- None wait\_device\_quiet (LineLidar self, Optional[float] timeout=None)
- None set\_clean\_state (LineLidar self, bool incl\_restricted=False, int retries=None, Optional[float] time-out=None)
- None reset (LineLidar self, bool reconnect=True, Optional[float] reconnect\_timeout=None, int retries=None, bool exc on nok=True, Optional[float] timeout=None)

#### **Public Attributes**

- · retries
- · notifications

#### **Static Public Attributes**

• int

#### **Private Member Functions**

- str highlighted bytes (LineLidar self, bytes b, Optional[int] start=None, Optional[int] end=None)
- None <u>\_send\_data</u> (LineLidar self, Union[bytes, List[bytes]] data)
- Tuple[bytes, Optional[str]] \_\_encode\_cmd (LineLidar self, int msgid, \_LLcmd cmd, Union[\_LLsrv, \_LLchr] chr\_or\_srv, \*bool args, \*\*\_CHR\_ARGS\_TYPE kwargs)
- int \_send\_cmd (LineLidar self, \_LLcmd cmd, Union[\_LLsrv, \_LLchr] chr\_or\_srv, \*bool args, \*\*\_CHR\_←
   ARGS\_TYPE kwargs)
- None \_\_windows\_reader\_thread (LineLidar self, IO fd, multiprocessing.queues.Queue queue)
- str <u>recv\_ssh\_line</u> (LineLidar self, Optional[float] timeout=None)
- LLresponse <u>decode\_msg</u> (LineLidar self, bytes msg)
- LLresponse <u>\_recv\_msg</u> (LineLidar self, Optional[float] timeout=None)
- LLresponse \_get\_cmd\_response (LineLidar self, bool exc\_on\_cmd=True, bool exc\_on\_msgid=True, bool exc\_on\_chr=True, bool exc\_on\_nok=True, bool ignore\_sent\_cmds\_log=False, Optional[float] timeout=None)
- LLresponse \_\_retry\_cmd (LineLidar self, \_LLcmd cmd, Union[\_LLsrv, \_LLchr] chr\_or\_srv, int retries=None, bool exc\_on\_nok=True, Optional[float] timeout=None, \*bool args, \*\*\_CHR\_ARGS\_TYPE kwargs)

### **Private Attributes**

- debug
- \_\_default\_timeout
- \_\_timeout
- \_\_autostop\_ranging
- \_\_is\_ranging
- ssh recvbuf
- \_netmask
- \_\_addrport
- · \_sent\_cmds\_log\_depth
- \_\_sent\_cmds\_log
- \_\_msgid

#### **Static Private Attributes**

- socket = None
- \_\_sshproc = None
- sshproc win stdout = None
- \_\_win\_reader\_thread = None

#### 8.7.1 Detailed Description

Definition at line 183 of file linelidar.py.

### 8.7.2 Constructor & Destructor Documentation

### 8.7.2.1 \_\_init\_\_()

```
None __init__ (
    LineLidar self,
    Optional[str] addr = None,
    Optional[int] port = None,
    Optional[str] sshcmd = None,
    Optional[str] sshpypath = None,
    Optional[int] sent_cmds_log_depth = None,
    int retries = None,
    bool set_clean_state = True,
    bool autostop_ranging = True,
    bool debug = False,
    Optional[float] timeout = None)
```

SSH receive buffer.

Default communication timeout in seconds

Current communication timeout in seconds

Whether to automatically stop ranging upon closing the device

Whether the device is currently ranging

last sent message ID

Sent commands log depth

Sent commands log

How many times a failed command should be retried for fault-tolerance

Notifications stack

Debugging messages toggle

Enum-by-value mappings

\_\_init\_\_ method

#### **Parameters**

addr	IP address of the device. If unspecified, the device is not automatically opened.
port	Port of the device (if the device is automatically opened)
sshcmd	If specified, ssh command to log into a shell account to use the host as a relay to talk to the LineLidar(if the device is automatically opened). Works with a Linux or Windows host with sshd and Python 2 or 3 installed.
sshpypath	Path of the Python executable on the SSH relay host (if the device is automatically opened)
sent_cmds_log_depth	Number of sent commands tracked, to ignore out-of-order responses that have already generated a timeout (if the device is automatically opened). Set to 1 to disable filtering out out-of-order UDP packets.  Generated by Doxygen
retries	How many times a failed command should be retried (if the device is automatically opened)
set_clean_state	Set device in a known, stopped state after opening (if the device is automatically

Definition at line 266 of file linelidar.py.

### 8.7.3 Member Function Documentation

## 8.7.3.1 \_\_decode\_msg()

Decode a message from the LineLidar.

#### **Parameters**

msg Packet to decode
----------------------

### Returns

Received response

Definition at line 1152 of file linelidar.py.

### 8.7.3.2 \_\_encode\_cmd()

Encode a LineLidar command.

#### **Parameters**

msgid	Message ID
cmd	Command to encode
chr_or_srv	Characteristic or service
args	Positional arguments (here, only True or False allowed - see below)
kwargs	Keyworded arguments (see below)

If the command is SAVE\_SERVICE or RESTORE\_SERVICE, pass a service (\_LLsrv) in chr\_or\_srv. Otherwise pass a characteristic (\_LLchr)

If the command is SET\_NOTIFICATION, pass a single argument True or False to enable or disable notification - e.g. \_encode\_cmd(LLcmd.SET\_NOTIFICATION, LLchr.TEMPERATURE, True)

If the command is WRITE, pass the relevant parameters as keyworded arguments - e.g.  $\_$ encode $\_$  $\leftarrow$  cmd( $\_$ LLcmd.WRITE,  $\_$ LLchr.MIN $\_$ DISTANCE, distance = 3.5)

#### Returns

(Encoded command, optional debug message)

Definition at line 687 of file linelidar.py.

#### 8.7.3.3 \_\_highlighted\_bytes()

Generate a printable byte sequence in hex with a slice of it highlighted.

#### **Parameters**

b	Byte sequence
start	Start of the byte sequence slice to highlight
end	End of the byte sequence slice to highlight

#### Returns

Printable hex byte sequence

Definition at line 354 of file linelidar.py.

#### 8.7.3.4 \_\_recv\_ssh\_line()

Get a line of text from the SSH client.

#### **Parameters**

timeout	Communication timeout in seconds
unicoal	

#### Returns

Received text line

Definition at line 1061 of file linelidar.py.

#### 8.7.3.5 \_\_retry\_cmd()

Try to send a command to the device and get a response from it.

If the response times out, retry sending the command and getting a new response until the number of allowed retries runs out.

#### **Parameters**

cmd	Command to send
chr_or_srv	Characteristic or service
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK
timeout	Communication timeout in seconds
args	Positional arguments for _send_cmd
kwargs	Keyworded arguments for _send_cmd

#### Returns

Response to the command

Definition at line 1591 of file linelidar.py.

#### 8.7.3.6 \_\_windows\_reader\_thread()

Windows-only blocking file reader thread.

This thread reads data from a blocking file and transfer the data to a queue, which in turn can be read with a timeout.

This workaround is required because Windows doesn't provide select() on file descriptors.

Definition at line 1014 of file linelidar.py.

#### 8.7.3.7 \_get\_cmd\_response()

Get a response to the latest command (i.e.

not a notification).

#### **Parameters**

exc_on_cmd	Raise an exception if the response's command doesn't match that of the last sent command
exc_on_msgid	Raise an exception if the response's message ID doesn't match that of the last sent command
exc_on_chr	Raise an exception if the response's characteristic doesn't match that of the last sent command in responses to READ commands
exc_on_nok	Raise an exception on response not OK (see below)
ignore_sent_cmds_log	If asserted, incoming messages will not be checked against older entries in the sent commands log to catch out-of-order responses, and the last sent command will not be removed from the log
timeout	Communication timeout in seconds

if ignore\_sent\_cmds\_log isn't asserted, if a response to a command is received and its message ID, command and service or characteristic match one of the older entries in the sent commands log, the response is considered an out-of-order message, the corresponding entry is removed from the sent commands log and the response is silently discarded.

After discarding out-of-order messages, if either the message's command, message ID or characteristic (in the case of response to a READ command) don't match the last sent command's and either exc\_on\_cmd, exc\_on\_msgid or exc\_on\_chr are asserted (default), an exception is raised.

if ignore\_sent\_cmds\_log isn't asserted and the response matches the last sent command, it is removed from the log.

If exc on nok is asserted (default), an exception is raised when the response's status is not OK.

If a NOTIFICATION is received, it is pushed into the notifications stack and the function keeps on waiting for a response to a command.

#### Returns

Received response

Definition at line 1425 of file linelidar.py.

## 8.7.3.8 \_recv\_msg()

Get a message from the LineLidar.

#### **Parameters**

timeout	Communication timeout in seconds
timeout	Communication timeout in seconds

#### Returns

Received response

Definition at line 1312 of file linelidar.py.

### 8.7.3.9 \_send\_cmd()

```
int _send_cmd (
          LineLidar self,
          _LLcmd cmd,
          Union[_LLsrv, _LLchr] chr_or_srv,
          *bool args,
          **_CHR_ARGS_TYPE kwargs ) [private]
```

Send a command to the device.

#### **Parameters**

cmd	Command to send
chr_or_srv	Characteristic or service
args	Positional arguments (here, only True or False allowed - see below)
kwargs	Keyworded arguments (see below)

If the command is SAVE\_SERVICE or RESTORE\_SERVICE, pass a service (\_LLsrv) in chr\_or\_srv. Otherwise pass a characteristic (\_LLchr)

If the command is SET\_NOTIFICATION, pass a single argument True or False to enable or disable notification - e.g. \_send\_cmd(LLcmd.SET\_NOTIFICATION, LLchr.TEMPERATURE, True)

If the command is WRITE, pass the relevant parameters as keyworded arguments - e.g. \_send\_cmd(LLcmd.WRITE, LLchr.MIN\_DISTANCE, distance = 3.5)

#### Returns

Sent message ID

Definition at line 931 of file linelidar.py.

### 8.7.3.10 \_send\_data()

Send data to the LineLidar.

#### **Parameters**

ata to send in the form of a byte array or a list of byte arrays
--

If the data is sent to the LineLidar directly, the data in the form of a byte array is sent.

If the data is sent to the LineLidar through a SSH relay, if the data is in the form of a byte array, it is first encapsulated in a list. Then the list of byte array(s) is pickled, base-64 encoded and sent to the SSH relay stub.

Definition at line 646 of file linelidar.py.

#### 8.7.3.11 disable all notifications()

Disable notification on all characteristics that support it.

#### **Parameters**

incl_restricted	Also disable notification on characteristics where notifications are restricted
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
timeout	Communication timeout in seconds

Definition at line 1769 of file linelidar.py.

# 8.7.3.12 disable\_notification()

Disable notification on a characteristic.

## **Parameters**

char	Characteristic to disable notification on
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
timeout	Communication timeout in seconds

Generated by Doxygen

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

#### Returns

SET\_NOTIFICATION command response

Definition at line 1743 of file linelidar.py.

#### 8.7.3.13 enable notification()

Enable notification on a characteristic.

#### **Parameters**

char	Characteristic to enable notification on
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
timeout	Communication timeout in seconds

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

## Returns

SET\_NOTIFICATION command response

Definition at line 1718 of file linelidar.py.

## 8.7.3.14 get\_notification()

Get a notification, either from the notifications stack, or receive it from the device if the stack is empty.

#### **Parameters**

chrmask	List of expected notification characteristics (see below)
timeout	Communication timeout in seconds

If a message other than a notification is received from the device (i.e. a response to a command), an exception is raised

If a list of characteristics is specified in chrmask and the notification's characteristic isn't in that list, an exception is raised.

## Returns

Received notification.

Definition at line 1933 of file linelidar.py.

#### 8.7.3.15 open()

```
Union[socket.socket,Tuple[subprocess.Popen, Optional[ multiprocessing.queues.Queue]]] open (
    LineLidar self,
    str addr,
    Optional[int] port = None,
    Optional[str] sshcmd = None,
    Optional[str] sshpypath = None,
    Optional[int] sent_cmds_log_depth = None,
    int retries = None,
    bool set_clean_state = True,
    bool autostop_ranging = True,
    Optional[float] timeout = None)
```

Open communication with a LineLidar.

#### **Parameters**

addr	IP address of the device
port	Port of the device
sshcmd	If specified, ssh command to log into a shell account to use the host as a relay to talk to the LineLidar. Works with a Linux or Windows host with sshd and Python 2 or 3 installed.
sshpypath	Path of the Python executable on the SSH relay host
sent_cmds_log_depth	Number of sent commands tracked, to ignore out-of-order responses that have already generated a timeout. Set to 1 to disable filtering out out-of-order UDP packets.
retries	How many times a failed command should be retried
set_clean_state	Set device in a known, stopped state after opening
autostop_ranging	Enable or disable automatically stopping ranging upon closing the device
timeout	Communication timeout in seconds

## Returns

Open UDP socket if the communication with the device is direct, or the tuple (open SSH client process, Windows-only stdout queue) if the communication with the device goes through an SSH relay host. On Linux machines, the Windows-only stdout queue is always None. On Windows machine, it is a multiprocessing. — Queue() object from which the SSH process' stdout can be read in non-blocking mode instead of the process' stdout handle, which is always blocking.

Definition at line 379 of file linelidar.py.

## 8.7.3.16 read\_chr()

Read a characteristic.

#### **Parameters**

char	Characteristic to read
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
timeout	Communication timeout in seconds

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

## Returns

READ command response.

Definition at line 1640 of file linelidar.py.

# 8.7.3.17 report\_zero\_results()

Enable or disable zero results reporting.

#### **Parameters**

on	Whether to report zero results
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
timeout	Communication timeout in seconds

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

#### Returns

WRITE command esponse

Definition at line 1806 of file linelidar.py.

#### 8.7.3.18 reset()

Soft-reset the device.

#### **Parameters**

reconnect	Attempt to reconnect the device after reset
reconnect_timeout	Timeout in seconds when attempting to reconnect
retries	How many times a failed command should be retried. Only applies if reconnect is False.
exc_on_nok	Raise an exception on response not OK (see below). Only applies if reconnect is False.
timeout	Communication timeout in seconds. Only applies if reconnect is False.

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

Definition at line 2027 of file linelidar.py.

#### 8.7.3.19 restore\_srv()

Restore a service from non-volatile memory.

#### **Parameters**

srv	Service to restore from non-volatile memory
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
Getienaeuty DoxygerCommunication timeout in seconds	

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

#### Returns

RESTORE\_SERVICE command response

Definition at line 1900 of file linelidar.py.

#### 8.7.3.20 save srv()

Save a service to non-volatile memory.

#### **Parameters**

srv	Service to save to non-volatile memory
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
timeout	Communication timeout in seconds

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

## Returns

SAVE\_SERVICE command response

Definition at line 1876 of file linelidar.py.

#### 8.7.3.21 set\_clean\_state()

Stop any active ranging, disable all notifications and flush the notifications stack, so that the device is left in a known, stopped state.

#### **Parameters**

incl_restricted	Also disable notification on characteristics where notifications are restricted
retries	How many times a failed command should be retried
timeout	Communication timeout in seconds

Definition at line 2000 of file linelidar.py.

# 8.7.3.22 set\_notification()

```
LLresponse set_notification (
    LineLidar self,
    _LLchr char,
    bool enabled,
    int retries = None,
    bool exc_on_nok = True,
    Optional[float] timeout = None)
```

Enable or disable notification on a characteristic.

#### **Parameters**

char	Characteristic to enable or disable notification on	
enabled	Whether notification is enabled	
retries	How many times a failed command should be retried	
exc_on_nok	Raise an exception on response not OK (see below)	
timeout	Communication timeout in seconds	

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

## Returns

SET\_NOTIFICATION command response

Definition at line 1692 of file linelidar.py.

#### 8.7.3.23 set\_sampling\_rate()

Set the sampling rate.

#### **Parameters**

frequency	Sampling frequency
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
timeout	Communication timeout in seconds

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

## Returns

WRITE command esponse

Definition at line 1830 of file linelidar.py.

# 8.7.3.24 stop\_sampling()

Stop sampling.

# **Parameters**

retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
timeout	Communication timeout in seconds

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

## Returns

WRITE command esponse

Definition at line 1854 of file linelidar.py.

## 8.7.3.25 wait\_device\_quiet()

Wait until the device becomes silent.

#### **Parameters**

timeout   Communication timeout in seconds
--

Definition at line 1980 of file linelidar.py.

## 8.7.3.26 write\_chr()

Write a characteristic.

#### **Parameters**

char	Characteristic to write
retries	How many times a failed command should be retried
exc_on_nok	Raise an exception on response not OK (see below)
timeout	Communication timeout in seconds
kwargs	Keyworded arguments (see below)

Pass the relevant parameters as keyworded arguments - e.g. write\_chr(LLchr.MIN\_DISTANCE, distance = 3.5)

If exc\_on\_nok is asserted (default), an exception is raised when the response's status is not OK.

# Returns

WRITE command esponse

Definition at line 1663 of file linelidar.py.

#### 8.7.4 Member Data Documentation

#### 8.7.4.1 int

```
int [static]
```

Address and port of the device or network.

Netmask to filter incoming UDP replies by IP - Default: /32 (single host)

Definition at line 194 of file linelidar.py.

The documentation for this class was generated from the following file:

linelidarclass/linelidar.py

# 8.8 LLchr Class Reference

## **Static Public Attributes**

- SERIAL\_NUMBER
- FW\_VERSION
- NETWORK
- TIME
- TEMPERATURE
- MAC
- DEFAULT\_NETWORK
- MIN DISTANCE
- MAX\_DISTANCE
- AMPLITUDE\_THRESHOLD
- MIN\_ANGLE
- MAX\_ANGLE
- TRIGGER\_SOURCE
- NB PEAKS
- REPORT\_ZERO\_RESULTS
- TRIGGER\_TYPE
- TRIGGER\_DIVISION
- CALIBRATED\_ANGLES
- RANGE
- MEASURING\_RATE
- RESET\_DEVICE

# 8.8.1 Detailed Description

# 8.8.2 Notes

Underscore in a name indicates a private characteristic

Definition at line 356 of file \_\_init\_\_.py.

#### 8.8.3 Member Data Documentation

## 8.8.3.1 AMPLITUDE\_THRESHOLD

Amplitude threshold.

Definition at line 416 of file \_\_init\_\_.py.

8.8 LLchr Class Reference 41

#### 8.8.3.2 CALIBRATED\_ANGLES

# 8.8.3.3 DEFAULT\_NETWORK

Definition at line 399 of file \_\_init\_\_.py.

# 8.8.3.4 FW\_VERSION

Definition at line 374 of file \_\_init\_\_.py.

## 8.8.3.5 MAC

Definition at line 394 of file \_\_init\_\_.py.

## 8.8.3.6 MAX\_ANGLE

# 8.8.3.7 MAX\_DISTANCE

Maximum ranging distance.

Definition at line 411 of file \_\_init\_\_.py.

# 8.8.3.8 MEASURING\_RATE

Definition at line 470 of file \_\_init\_\_.py.

# 8.8.3.9 MIN\_ANGLE

Definition at line 421 of file \_\_init\_\_.py.

8.8 LLchr Class Reference 43

## 8.8.3.10 MIN\_DISTANCE

# 8.8.3.11 NB\_PEAKS

Number of peaks.

Definition at line 436 of file \_\_init\_\_.py.

# 8.8.3.12 NETWORK

Definition at line 379 of file \_\_init\_\_.py.

NETWORK: network settings.

# 8.8.3.13 RANGE

Definition at line 465 of file \_\_init\_\_.py.

# 8.8.3.14 REPORT\_ZERO\_RESULTS

Definition at line 441 of file \_\_init\_\_.py.

# 8.8.3.15 RESET\_DEVICE

Reset device.

Definition at line 477 of file \_\_init\_\_.py.

# 8.8.3.16 SERIAL\_NUMBER

Serial number.

Definition at line 369 of file \_\_init\_\_.py.

# 8.8.3.17 TEMPERATURE

Temperature of the device.

Definition at line 389 of file \_\_init\_\_.py.

8.8 LLchr Class Reference 45

## 8.8.3.18 TIME

## 8.8.3.19 TRIGGER\_DIVISION

## 8.8.3.20 TRIGGER\_SOURCE

Definition at line 451 of file \_\_init\_\_.py.

Definition at line 431 of file \_\_init\_\_.py.

# 8.8.3.21 TRIGGER\_TYPE

The documentation for this class was generated from the following file:

linelidarclass/\_\_init\_\_.py

Definition at line 446 of file \_\_init\_\_.py.

# 8.9 LLcmd Class Reference

## **Static Public Attributes**

- ERROR = LLcmd("ERROR", 0x0000)
- READ = LLcmd("READ", 0x0001)
- WRITE = \_LLcmd("WRITE", 0x0002)
- NOTIFICATION = LLcmd("NOTIFICATION", 0x0005)
- SET\_NOTIFICATION = \_LLcmd("SET\_NOTIFICATION", 0x0006)
- SAVE\_SERVICE = \_LLcmd("SAVE\_SERVICE", 0x0007)
- RESTORE\_SERVICE = \_LLcmd("RESTORE\_SERVICE", 0x0008)

# 8.9.1 Detailed Description

Definition at line 289 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

• linelidarclass/\_\_init\_\_.py

# 8.10 LLresponse Class Reference

## **Public Member Functions**

- None \_\_init\_\_ (LLresponse self, Optional[\_LLcmd] cmd=None, Optional[int] msgid=None, Optional[\_LLsta] status=None, Optional[\_LLchr] char=None, Optional[datetime] recv\_local\_timestamp=None)
- str \_\_repr\_\_ (LLresponse self)

## **Public Attributes**

- · recv\_local\_timestamp
- cmd
- msgid
- status
- char
- value
- major
- minor
- bugfix
- addr
- · defaultgw
- · netmask
- port
- time
- · temperature
- mac
- distance
- angle
- source

- type
- div
- frequency
- threshold
- · angles
- · peaks
- timestamp
- measurementid
- · nbresults
- reset

# 8.10.1 Detailed Description

Definition at line 517 of file \_\_init\_\_.py.

# 8.10.2 Member Function Documentation

```
8.10.2.1 __repr__()
str __repr__ (
```

Generate a printable description of the response.

LLresponse self )

Returns

printable response

Definition at line 691 of file \_\_init\_\_.py.

#### 8.10.3 Member Data Documentation

## 8.10.3.1 reset

reset

Attribute present in notifications or read command responses for characteristic RANGE.

Attribute present in read command responses for characteristic RESET

Definition at line 687 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

linelidarclass/\_\_init\_\_.py

## 8.11 LLsry Class Reference

#### **Static Public Attributes**

- DEVICE\_INFO = \_LLsrv("DEVICE\_INFO", 0x0001)
- DEVICE\_CONFIG = \_LLsrv("DEVICE\_CONFIG", 0x0002)
- HW CONFIG = LLsrv("HW CONFIG", 0x0003)
- RESULTS = LLsrv("RESULTS", 0x0004)
- DEBUG = LLsrv("DEBUG", 0x0005)

# 8.11.1 Detailed Description

#### 8.11.2 Notes

Underscore in a name indicates a private service

Definition at line 325 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

• linelidarclass/\_\_init\_\_.py

# 8.12 LLsta Class Reference

## **Static Public Attributes**

- OK = LLsta("OK", 0x0000)
- CHAR DECODING FAILED = LLsta("CHAR DECODING FAILED", 0x0001)
- CHAR\_OUT\_OF\_RANGE = \_LLsta("CHAR\_OUT\_OF\_RANGE", 0x0002)
- PROCEDURE\_IN\_PROGRESS = \_LLsta("PROCEDURE\_IN\_PROGRESS", 0x0003)
- PROCEDURE FAILED = LLsta("PROCEDURE FAILED", 0x0004)
- NOT\_ALLOWED = \_LLsta("NOT\_ALLOWED", 0x0006)

# 8.12.1 Detailed Description

Definition at line 482 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

• linelidarclass/\_\_init\_\_.py

# 8.13 macaddress Class Reference

# **Public Member Functions**

- None \_\_init\_\_ (macaddress self, Union[str, bytes] addr)
- bool eq (macaddress self, object other)
- str \_\_repr\_\_ (macaddress self)

# **Public Attributes**

• mac

# 8.13.1 Detailed Description

Definition at line 96 of file \_\_init\_\_.py.

The documentation for this class was generated from the following file:

• linelidarclass/\_\_init\_\_.py

# Index

_LLchr, 15	LLchr, 40
eq, 16	
hash, 16	Base classes, 11
init, 15	
setattr, 16	CALIBRATED_ANGLES
_LLcmd, 17	LLchr, 40
_LLintEnum, 18	Defectly a superstant of
eq, 19	Default parameters, 11
hash, 19	DEFAULT_NETWORK
init, 18	LLchr, 41
setattr, 19	disable_all_notifications
LLsrv, 20	LineLidar, 31
_LLsta, 21	disable_notification
decode_msg	LineLidar, 31
LineLidar, 25	discover
encode_cmd	linelidarclass.linelidar, 14
LineLidar, 25	
eq eq	enable_notification
LLchr, 16	LineLidar, 32
LLintEnum, 19	Enums, 11
_hash	EW VEDOLONI
LLchr, 16	FW_VERSION
_LLintEnum, 19	LLchr, 41
highlighted bytes	get_notification
• • _ ·	LineLidar, 32
LineLidar, 26 init	LineLidal, 32
<del></del>	int
_LLchr, 15	LineLidar, 39
_LLintEnum, 18	ipaddress, 21
LineLidar, 24	1padd1633, 21
recv_ssh_line	LineLidar, 22
LineLidar, 26	decode_msg, 25
repr	encode_cmd, 25
LLresponse, 47	shighlighted_bytes, 26
retry_cmd	init, 24
LineLidar, 27	recv_ssh_line, 26
setattr	retry_cmd, 27
_LLchr, 16	ion y_ond, 27 windows_reader_thread, 27
_LLintEnum, 19	windows_redder_tiredd, 27
windows_reader_thread	get_cind_response, 27 _recv_msg, 28
LineLidar, 27	send cmd, 30
_get_cmd_response	:
LineLidar, 27	_send_data, 30
_recv_msg	disable_all_notifications, 31
LineLidar, 28	disable_notification, 31
_send_cmd	enable_notification, 32
LineLidar, 30	get_notification, 32
_send_data	int, 39
LineLidar, 30	open, 33
	read_chr, 34
AMPLITUDE THRESHOLD	report_zero_results, 34

52 INDEX

reset, 35	NETWORK
restore_srv, 35	LLchr, 43
save_srv, 36	
set_clean_state, 36	open
set_notification, 37	LineLidar, 33
set_sampling_rate, 37	DANIOE
stop_sampling, 38	RANGE
wait_device_quiet, 38	LLchr, 43
write_chr, 39	read_chr
linelidarclass, 13	LineLidar, 34
linelidarclass.default, 13	REPORT_ZERO_RESULTS
linelidarclass.linelidar, 14	LLchr, 43
discover, 14	report_zero_results
LLchr, 40	LineLidar, 34
AMPLITUDE_THRESHOLD, 40	reset
CALIBRATED_ANGLES, 40	LineLidar, 35
DEFAULT_NETWORK, 41	LLresponse, 47
FW VERSION, 41	RESET_DEVICE
MAC, 41	LLchr, 44
MAX ANGLE, 41	restore_srv
MAX DISTANCE, 42	LineLidar, 35
MEASURING_RATE, 42	Routines, 11
MIN ANGLE, 42	
MIN DISTANCE, 42	save_srv
NB PEAKS, 43	LineLidar, 36
NETWORK, 43	SERIAL_NUMBER
RANGE, 43	LLchr, 44
REPORT_ZERO_RESULTS, 43	set_clean_state
RESET DEVICE, 44	LineLidar, 36
SERIAL NUMBER, 44	set_notification
— · · · · · · · · · · · · · · · · · · ·	LineLidar, 37
TEMPERATURE, 44	set_sampling_rate
TIME, 44	LineLidar, 37
TRIGGER_DIVISION, 45	stop_sampling
TRIGGER_SOURCE, 45	LineLidar, 38
TRIGGER_TYPE, 45	
LLcmd, 46	TEMPERATURE
LLresponse, 46	LLchr, 44
repr, 47	TIME
reset, 47	LLchr, 44
LLsrv, 48	TRIGGER_DIVISION
LLsta, 48	LLchr, 45
MAC	TRIGGER_SOURCE
-	LLchr, 45
LLchr, 41	TRIGGER_TYPE
macaddress, 48	LLchr, 45
Main classes, 11	
MAX_ANGLE	wait_device_quiet
LLchr, 41	LineLidar, 38
MAX_DISTANCE	write_chr
LLchr, 42	LineLidar, 39
MEASURING_RATE	
LLchr, 42	
MIN_ANGLE	
LLchr, 42	
MIN_DISTANCE	
LLchr, 42	
NR DEAKS	
NB_PEAKS LLchr, 43	
LLGIII, 40	