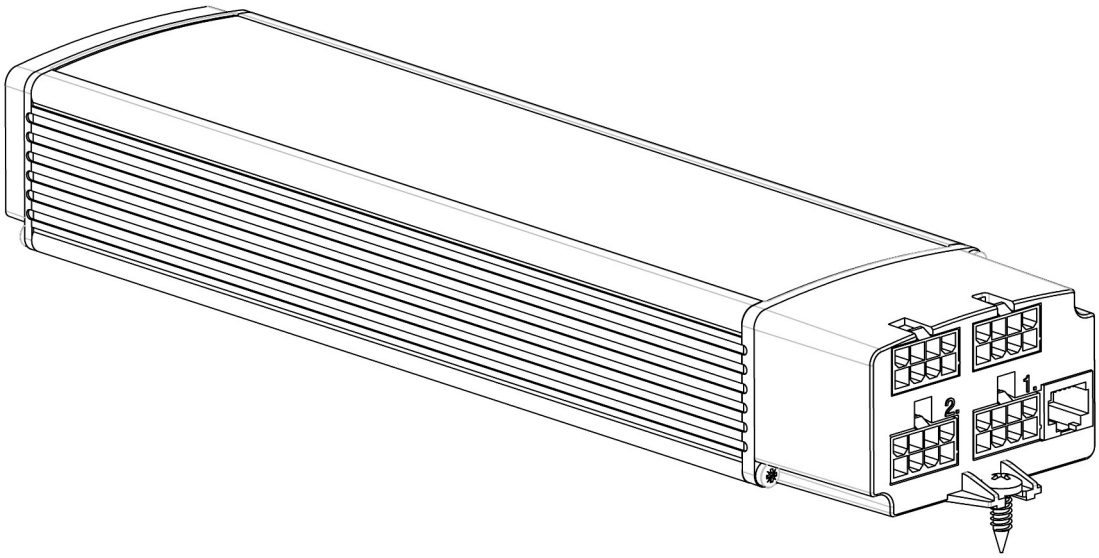


Laing Table Control

Modbus Interface Specification

Table of Contents

Index of Tables

Check for application updates: https://laing-controller.de/apps/

KZO@202208091417

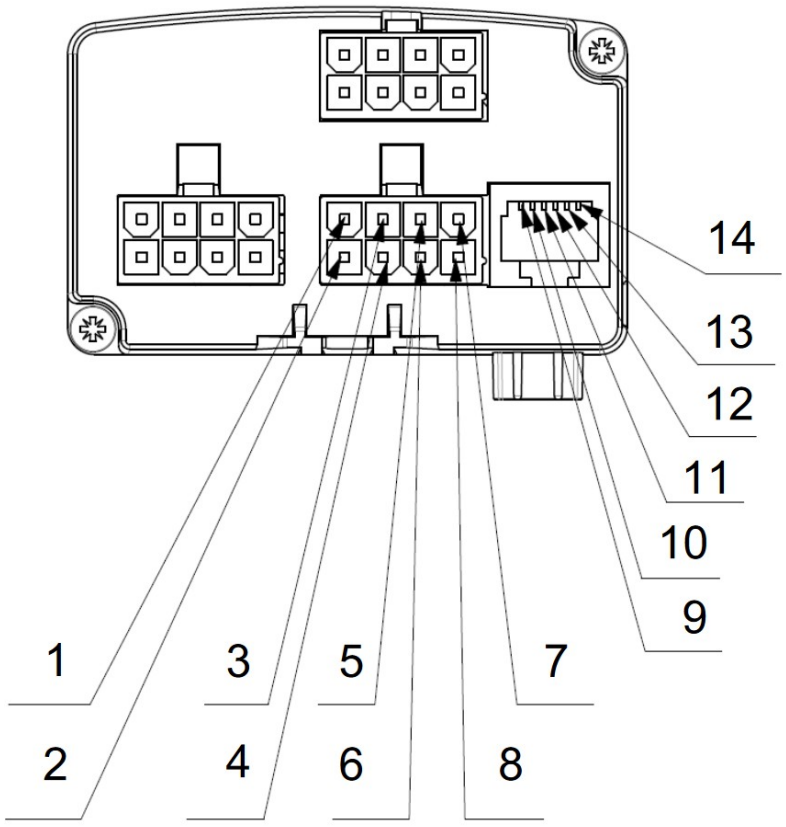
1. Connection

LTC are using Modbus (http://www.modbus.org/tech.php) RTU Slave with 2 wire RS485.
Connection parameters: 57600 baud 8n2. Slave address 1.

Supported commands:

- Read holding registers (0x03)
- Write multiple registers (0x10)
- Read Write multiple registers (0x17)

Number format is unsigned short.



1. Motor connection 1
2. Hall sensor 1
3. Ground
4. Hallsensor +5 V
5. Optional 2
6. Optional 1
7. Hallsensor 2
8. Motor connection 2
9. +5V out
10. RS 485 A
11. RS 485 B
12. External +5V power
13. Analog control panel
14. Ground

Figure 1: Pin assignment

2. Keyboard control table

Address	Description	Unit
2000 (R)	Table height	SL[1]

Address	Description	Unit
2001 (R)	Last KeyPress Data	raw
2002 (R/W)	KeyPress Control	raw

- Register at 2000 is readable and MUST be polled regularly (suggested in every 500ms). This validates that the communication is up & running. (It must be ensured that there is no movement without user request/keypress!)
- The register value at 2000 provides the table height in millimeters. When a key is pressed, then the register at 2002 must be written with the keycode of the pressed button.
  - 0: no button is pressed
  - 1: button ‘1’ is pressed
  - 2: button ‘2’ is pressed
  - 3: button ‘3’ is pressed
  - 4: button ‘4’ is pressed
  - 5: button UP is pressed
  - 6: button DOWN is pressed.
- Suggestion: In normal usage, set 5/6 when UP/DOWN is pressed, set 0 when button is released.

3. Reference run settings

If limit switch turned off, current based reference searching activated. When motor current reached the limit reference found.

Table 1: Reference run settings

#	Meaning	Register	Action	Arg	Note
1	Stop current threshold at bottom[2]	29000, 29001, 29002, 29003	Write		Limit value for motor stop current in reference run [mA]
2	Stop current threshold at top[3]	29017, 29018, 29019, 29020			
3	Use limit switch	29012	Write	0/1	„1” when limit switch (in motor wire) is used for reference detection
4	Check motor presence	29015	Write	0/1	Check the motors on reference run start

1 SL: Scaled length
2 Usually same values for all motors
3 Usually same values for all motors

#	Meaning	Register	Action	Arg	Note
5	Two way reference	29016	Write	0/1	Detect the bottom in first run, and detect the top (height) in secound run

4. Error detection

Table 2: Error check

#	Meaning	Register	Action	Arg	Note
1	Check logic state periodically	14005	Read		Table 3: Logic states
2	In error state, read last error	14006	Read		Table 4: Error codes
3	In error, check ErrorStateTimer	14007	Read		Can move, when it is 0

Note:

- Read registers with start 14005, length 3….

5. Device configuration

The configuration container XML file can be downloaded by a provided PC downloader application. Additionally, user can fine-tune numerous parameters of the system via Modbus table values as follows according to the next chapters.

6. Table specification

Table 5: Table specification

Address	Description	Unit
20003 (R/W)	Top mechanical limit for movement	SL[4]
20004 (R/W)	Distance kept from highes position (Mechanical margin)	SL
20005 (R/W)	Distance kept form lowest position (Mechanical margin)	SL
20006 (R/W)	Distance from target position where slowdown starts	SL
20007 (R/W)	Distance of move back after collision event	SL
20008 (R/W)	Distance from target where the controller considers the target reached	SL
20011 (R/W)	Correction for the heigth indication in the display	SL
20012 (R/W)	Desired speed	SL/sec
20014 (R/W)	Acceleration time until target speed is reached	[ms]
20015 (R/W)	Time for speed ramp down to zero, after "move button" is released	[ms]

4 SL: Scaled length

Address	Description	Unit
	in sec	
20017 (R/W)	Target speed for the reference run	SL/sec
20020 (R/W)	Acceleration time until reference target speed is reached	[ms]
20023 (R/W)	Value by what the bottom limit switch will be overrun(negative margin)	SL
20022 (R/W)	Value by what the top limit switch will be overrun(negative margin)	SL

Notes

- The above table is called as “Specification” table. Used to hold and identify the high-level specification data to determine the basic behaviour of the table control
- After the values are set, the “ApplySpecification” (3012) command must be called. Then all parameters used by the system internally will be calculated (by the device) and get into effect immediately.

7. User settings table

Address	Description	Unit
26000 (R/W)	UserLowLimitPosition	
26002 (R/W)	UserHighLimitPosition	
26004 (R/W)	User position 1	SL
26006 (R/W)	User position 2	SL
26008 (R/W)	User position 3	SL
26010 (R/W)	User position 4	SL
26014 (R/W)	"0" metric (mm) "1" imperial (inch)	
26015 (R/W)	Display indication: "0" tenth are shown, "1" rounded to .5 "2" rounded to .0	
26017 (R/W)	"1" button mode active, drive will run after a short key press. "0" drive will only run as long as key is pressed	

8. Table control settings

Table 6: Table control settings

Address	Description	Unit
30000 (R/W)	Path Length	SL
30001(R/W)	Path Raw	raw
30003 (R/W)	Height Offset (additional offset to path length)	SL
30018 (R/W)	Speed control Proportional Gain	raw

30019 (R/W)	Speed control Integral Gain	raw
30020 (R/W)	Speed control Differential Gain	raw
30021 (R/W)	Speed control time-base prescaler	raw
30022 (R/W)	Speed control Output saturation range	mA
30028 (R/W)	Stop current threshold M1	mA
30029 (R/W)	Stop current threshold M2	mA
30030 (R/W)	Stop current threshold M3	mA
30031 (R/W)	Stop current threshold M4	mA
21013 (R/W)	Set Reference mode @ Power Up	
21015 (R/W)	Enable Brake (apply software-controlled brake when motor stopped)	

Notes

- The ratio of Path Length and Path Raw defines the conversion rate of a table's raw increments to scaled value (mm / length)

9. Enumerations

Table 7: Table modes

ID	Description
0	IDLE mode
3	PWM test mode (Usually for test and debug)
4	CURRENT test mode (Usually for test and debug)
5	SPEED test mode (Usually for test and debug)
6	Long Term test mode (Usually for test)
7	Reference mode
8	NORMAL mode

Table 8: Logic states

ID	Description
0	IDLE state
1	MOVE state
2	ERROR state

Table 9: Error codes

ID	Description	Default wait time [s]
0	No error	-

1	Non-volatile memory initialization error	-
2	Non-volatile memory write error	-
3	Non-volatile memory read error	-
4	Generic non-volatile memory error	-
5	Collision Event	1
6	Stop overcurrent error	60
7	Error overcurrent error	Same as Error #6
8	Hardware fault overcurrent error	Same as Error #6
9	I2t error	60
10	Power supply I2t error	60
11	Power supply overtemperature error	60
12	Motor drive overtemperature error	Same as Error #11
13	M3 motor drive overtemperature error	Same as Error #11
14	Position error	1
15	Motor blocked error	Same as Error #14
16	Motor presence error	Same as Error #14
17	Power Supply overload error	60
18	Hardware error	-
19	Motor sequence error	Same as Error #5.
20	Safety adapter missing	Same as Error #5.
21	Safety adapter active	No wait time, except in sync: Same as Error #5
22	Hall sensor error	Same as Error #14
23	Hub: controller config error	-
24	hub: controller count error	-
25	Battery overcurrent	Same as Error #6
26	Battery shortcircuit	Same as Error #6
27	Battery overvoltage	Same as Error #6
28	Battery undervoltage	Same as Error #6
29	Battery devicefault	Same as Error #6
30	Battery lowpower	Same as Error #6
31	Battery temp	Same as Error #6
32	Hub: firmware error	-

Table 10: Reference mode states

ID	Description
0	RM_IDLE

ID	Description
1	RM_PRE_POSITION
2	RM_FINAL_POSITION
3	RM_FINAL_POSITION_RUN
4	RM_REACHED

10. System status

Table 11: System status

Address	Description
14000 (R)	Reset count
14002 (R)	Cycle Count
14005 (R)	Logic State
14006 (R)	“Last Error” code
14008 (R)	Hardware overcurrent (fault) error detected

11. Error notebook

Table 12: Error notebook

Address	Description	Min	Max	Unit
1000 (R)	Error Index. Indicates the next error entry (Queue) index where the next error code will be written to	0	65535	-
1001 (R)	Error notebook Queue [0]	0	16	ERROR CODE
1002 (R)	Error notebook Queue [1]	0	16	ERROR CODE
1016 (R)	Error notebook Queue [15]	0	16	ERROR CODE

12. Motor Descriptor

Table 13: Motor Descriptor

M1/M2/M3/M4 Address	Description	Unit
11000/11022/11044/11066	MotorX.StopCurrentThreshold	mA
11003/11025/11047/11069	MotorX.ActualDirection	

11004/11026/11048/11070	MotorX.ActualPosition	raw
11005/11027/11049/11071	MotorX.ActualSpeed	raw
11006/11028/11050/11072	MotorX.ActualPwm	
11007/11029/11051/11073	MotorX.ActualMotorPower	W
11008/11030/11052/11074	MotorX.ActualMotorSignalState	bool
11009/11031/11053/11075	MotorX.CurrentReference	mA
11010/11032/11054/11076	MotorX.CurrentFeedback	mA
11011/11033/11055/11077	MotorX.MaximumCurrent	mA
11015/11027/11059/11078	MotorX.I2t	

13. Move Descriptor

Table 14: Move Descriptor

Address	Description	Unit
12000	Movement in Progress	bool
12001	Actual direction	0: DOWN, 1: UP
12002	Actual position	raw
12003	Position target	raw
12004	Distance from target	raw
12005	Position error	raw
12006	In Position (target reached)	bool
12015/12021/12027/12033	MotorX.SpeedFeedback (motor speed)	raw

14. Collision Settings

Table 15: Dynamic overcurrent based collision settings

Address	Description	Unit
26012 (UserSetting)	Collision Level	1: disabled, 2: default, 3: lower sensitivity, 4 lowest sensitivity
25000/25005/25010/25015	Collision Detection High Pass Filter	raw
25001/25006/25011/25016	Collision Detection Low Pass Filter	raw
25002/25007/25012/25017	Collision Detection Limit	raw
26013 (UserSetting)	Collision Level Step	raw

15. Gyro

Table 16: Gyro settings

Address	Description	Unit
25024 (R/W)	VelocityLPFilter	raw
25025 (R/W)	VelocityHPFilter	raw
25026 (R/W)	VelocityLimit_X	raw
25027 (R/W)	VelocityLimit_Y	raw
25028 (R/W)	VelocityLimit_Z	raw

Table 17: Gyro information

Address	Description	Unit
15009 (R)	Velocity_HPX	raw
15010 (R)	Velocity_HPY	raw
15011 (R)	Velocity_HPZ	raw
15006 (R)	Velocity_LPX	raw
15007 (R)	Velocity_LPY	raw
15006 (R)	Velocity_LPZ	raw

15.1 Gyro parameter calculation

```
Velocity_LPX = Velocity_LPX + ((gyro_X_value - Velocity_LPX) / VelocityLPFilter)
Velocity_LPY = Velocity_LPY + ((gyro_Y_value - Velocity_LPY) / VelocityLPFilter)
Velocity_LPZ = Velocity_LPZ + ((gyro_Z_value - Velocity_LPZ) / VelocityLPFilter)

vlpf_X = vlpf_X + (Velocity_LPX - vlpf_X) / VelocityHPFilter
vlpf_Y = vlpf_Y + (Velocity_LPY - vlpf_Y) / VelocityHPFilter
vlpf_Z = vlpf_Z + (Velocity_LPZ - vlpf_Z) / VelocityHPFilter

Velocity_HPX = Velocity_LPX - vlpf_X
Velocity_HPY = Velocity_LPY - vlpf_Y
Velocity_HPZ = Velocity_LPZ – vlpf_Z

if ( Velocity_HPX >= VelocityLimit_X ) collision();
if ( Velocity_HPY >= VelocityLimit_Y ) collision();
if ( Velocity_HPZ >= VelocityLimit_Z ) collision();
```

16. Self descriptor table

Table 18: Selfdescriptor

Address	Field name	Description
100	Validity	Just a number to identificate the internal database type (typically never change in same parameter set)
101	WhoAmI	"0" LTC, "1" LMC, "2" Hub
102	FirmwareVersion	Firmware version number
103	HardwareVersion	Hardware version number
104	SerialNumberLow	Lower part serial number
105	SerialNumberMid	Higher part serial number
106	SerialNumberHigh	Produktion year, month (2003)
107	ParameterSetID	ID of the Parameter set suitable for the firmware
108	VendorID	Optional customer specific code
109	VendorProductID	Numeric product designation 1 what can be used by the customer when saving the configuration
110	VendorParam1	Numeric product designation 2 what can be used by the customer when saving the configuration
111	VendorParam2	Numeric product designation 3 what can be used by the customer when saving the configuration
112	InternalBarcodeLow	Lower part serial number
113	InternalBarcodeHigh	Higher part serial number
114	ProductionData0	Device voltage
115	ProductionData1	Now it is empty
116	M1CurrentCalibration	Calibration value for Current measurement motor 1
117	M2CurrentCalibration	Calibration value for Current measurement motor 2
118	M3CurrentCalibration	Calibration value for Current measurement motor 3
119	M4CurrentCalibration	Calibration value for Current measurement motor 4
120	GyroOrientation	Orientation of gyro sensor (parallel to main board or perpendicular)
121	ParameterFileID	ID of the Parameter file (setted by user)
122	MaxMotorCount	Number of connected motors the controller is configured for
148	Config Checksum HI	Check sum high for configuration (can be used to detect changes in the configuration)
149	Config Checksum LO	Check sum low for configuration (can be used to detect changes in the configuration)

17. System control

Table 19: System control registers

Address	Description
3000 (W)	Reset controller processor, processor will load saved configuration [5]
3000 (R)	Controller uptime
3002 (W)	Save configuration to flash
3003 (W)	Load configuration from flash
3004 (R/W)	System mode. See Table 7
3005 (W)	Reset position, controller goes to reference mode
3011 (W)	Clear the errors from log
3012 (W)	Convert the real units to raw data
3021 (W)	Modify the position target when controller moving

18. Remote control interface

18.1 BLE (Nordic UART service)

RX characteristic: 6E400003-B5A3-F393-E0A9-E50E24DCCA9E

TX characteristic: 6E400002-B5A3-F393-E0A9-E50E24DCCA9E

18.2 Wifi

The LTC device acts as a TCP server where the ASCII based communication interface is provided. After establishing a connection to an existing WIFI network, a SOCKET client must connect to the SOCKET server of the LTC (that is listening at the specified TCP/IP port).

18.3 Plain text protocol

Then at the application level, pure text based communication protocol is used. This WIFI/BLE communication channel accesses the Modbus tables of the system. Additional configuration keywords / status parameters are not defined/introduced. Generic structure of the protocol on the application level is structured as follows:

HEADER CMD PAR <LF>

The generic concept is to allow 100% capability through this channel too. Consequently, there is pure access to the existing MODBUS tables. For this reason, the protocol is optimized for SET (register) and GET (register/table) commands, that accesses the MODBUS tables. SET register is used to set the value of a single 16 bit MODBUS register. GET register is used to obtain the

5 Controller will answer to reset command. You have to wait 100 ms after this command

value of a single 16 bit MODBUS register. GET table is used to obtain all registers of a MODBUS table, identified by the address of the first item. The generic format of the SET (with regular expression format) as follows:

```
Message: ^#R[0-9]+=[0-9]+\n$
Answer: ^OK\n$
```

The generic format of the GET Register (with regular expression format) as follows:

```
Message: ^#GR=[0-9]+\n$
Answer: ^#R[0-9]+=[0-9]+\n$
```

The generic format of the GET Table (with regular expression format) as follows:

```
Message: ^#GT=[0-9]+\n$
Answer: (^#R[0-9]+=[0-9]+\n$)+
```

For safe usage: sending a message and waiting for the always mandatory answer is advised.

If no response received, repeat the message.

„#OK” is the end of response.

Table 20: Plain text commands

Command	Sample	Response	Description
#Rx=y	#R30003=650	#OK	Set the register 30003 to 650
#GT=x	#GT=10000	#R10039=0 #R10038=0 …... #R10000=0 #OK	Query the table which started with 10000
#GR=x	#GR=30003	#R30003=650 #OK	Query the register 30003
#CMD idle	#CMD idle	#R10005=756 #R12000=0 #OK	Idle (poll) or stop command. Response: the table height and moveprogress
#CMD up	#CMD up	#R10005=766 #R12000=1 #OK	Up command Response: the table height and moveprogress
#CMD down	#CMD down	#R10005=766 #R12000=1 #OK	Down command Response: the table height and moveprogress
#CMD y	#CMD 1	#R10005=766 #R12000=1 #OK	Go to user position 1 command for M1. Response: the motor positions.
#CMD stop	#CMD stop	#R10005=766 #R12000=1 #OK	All motor stop. Response: the motor positions.

Command	Sample	Response	Description
#CMD … :z	#CMD up:13011	#R13011=142 #OK	Up command. Response: the value of register 13011

18.4 Logic

You have to poll at least every 1500ms. This value stored in register 23010. If this time expired movement will stop!

For eg:
Go up
Send #CMD up in every 400 ms.
Send #CMD idle will stop
Send #CMD idle in every 400 ms to receive motor positions.