TECHLINE®
IMPROVING FLEXIBILITY

Installation guide

Modbus option for LA35, LA36 and LA37

To learn more about LINAK, please visit:

www.linak.com

LINAK®
WE IMPROVE YOUR LIFE

Contents

Introduction

The RS485 MODBUS RTU option is a serial communication interface between the actuators and a control system. The MODBUS interface can directly communicate with a PLC with a MODBUS module or a PC through an external USB to RS485 interface box.

This document describes how to install, configure and use an actuator with embedded MODBUS RTU serial communication.

Basic serial bus communication knowledge is a prerequisite for using and understanding the below documentation.

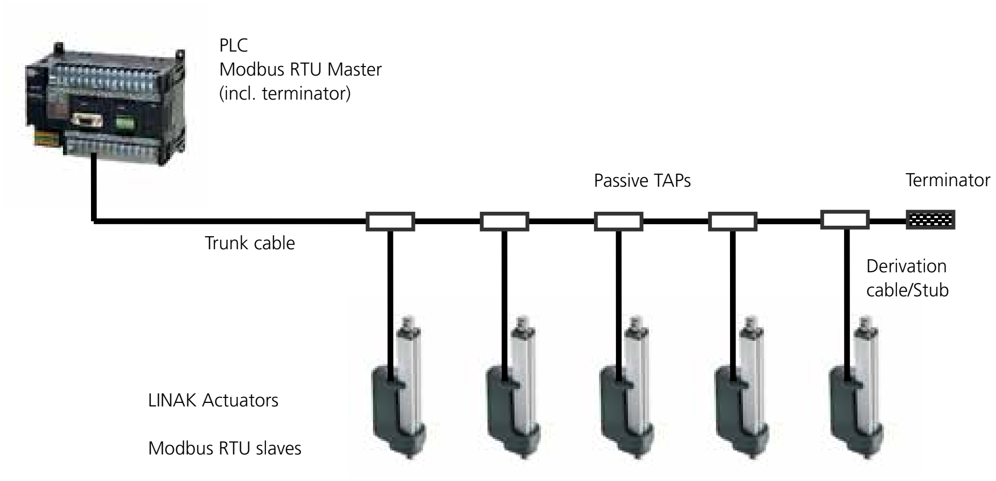
Supporting literature:
[1] MODBUS over serial line specification and implementation guide V1.02 (LINK)
[2] MODBUS Application Protocol Specification V1.1b (LINK)
[3] BusLink configuration software (if you do not have BusLink please contact your local LINAK office)

Concept

Modbus RTU System

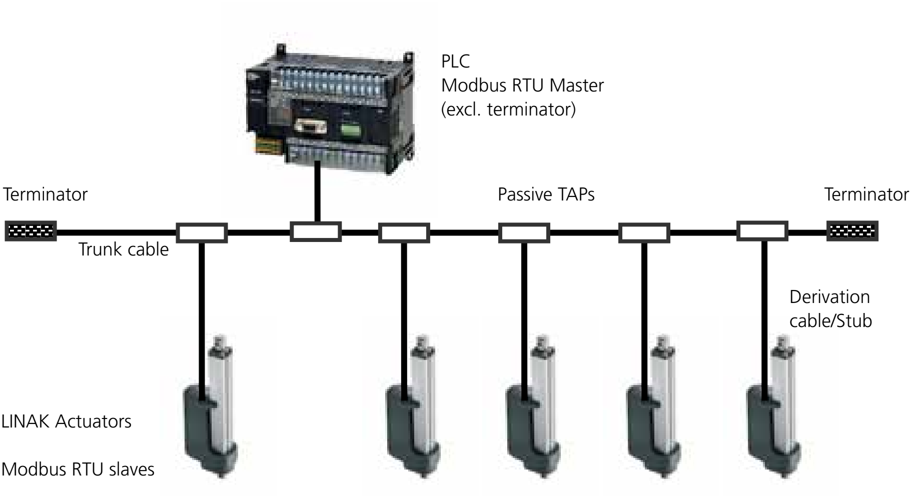
Universally recognised and widely used, the MODBUS RTU fieldbus is still an essential, open communications standard, supported by a large number of products on the market today.

In the MODBUS network any MODBUS Master can be connected to one or several LINAK LA35/LA36 Actuators with MODBUS interface. The topology is a serial bus-system with actuators including derivation cable connected to a trunk cable through passive TAPs. One master PLC or PC can be connected to the serial bus to control and supervise the actuator slaves. The actuators might potentially be mixed with other 3rd party MODBUS slaves.



At the physical protocol level, the RS-485 (TIA/EIA-485) two-wire interface is used, which supports half-duplex communication between the master and one or more slaves. Inter-communication between slaves is not possible and a slave will never transmit data without receiving a request from the master. The master (incl. terminator) can be connected at the end of the cable as shown in the figure above.

Alternatively connection of the master anywhere in-between two of the slaves is acceptable as well. As shown below the terminator is then moved to the end of the trunk cable.



Safety instructions

Be aware of the following symbols throughout the installation guide:

👉 **Recommendations**
Failing to follow these instructions can result in the actuator suffering damage or being ruined.

ⓘ **Additional information**
Usage tips or additional information that is important in connection with the use of the actuator.

ⓘ Be aware that a lot of test and quality activities have been performed to ensure the functionality and safe use of the product. As with other electronic equipment the MODBUS option has a finite failure rate. To ensure that one failure does not lead to an unsafe state the microcontroller monitors critical components (1. failure surveillance). It is still possible to run the actuator but it is of utmost importance that the user reacts upon these events to maintain 1. failure safety (i.e. polling of relevant MODBUS registers to identify the reason for last stop) – see Troubleshooting page 32 (Problem: The actuator does not move after a run-command).

Installation

Connection, cables and plugs

The actuator data and power cables are separate from each other and pre-mounted on the LA35/ LA36/LA37 Actuators.

Power cable
Standard LA35, LA36 and LA37 power cables are used. The power cable is an open-end flying leads version.

The following issues must be considered when extending the power cable:

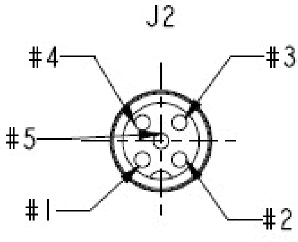
There are no specific requirements for the power cable itself other than it of course needs to be dimensioned to handle the maximum voltage drop for the actuator with the longest cable distance from the power supply. In terms of choosing the right power cable other issues must also be clarified, i.e.:

- The worst case current consumption for the individual actuator
- If it is specified that more than one actuator should run simultaneously

In other words the goal is to select cable dimensions, which ensure, that the input voltage of all bus actuators is within the specified limits under the selected worst case operating conditions.

Data cable
The data derivation cable ends in a standard M12 x 1 connector (Male, A-coded, 5-position). The data cable mounted from the factory must always be used.

The connector pin layout has been chosen to enable use of standard DeviceNet M12 cables, passive TAPs (T-pieces) and terminators.



Pin	Signal
1	Common RS485 signal reference + shield of the cable
2	Reserved for future use
3	Reserved for future use
4	RS485 non-inverting signal – RS485_B (aka TxD+/RxD+)
5	RS485 inverting signal – RS485_A (aka TxD-/RxD-)

☝ Even though the EIA-485 specification states that **A** is the **inverting** signal and **B** is the **non-inverting** signal, several RS485 transceiver manufacturers specify the opposite, which is a widespread misunderstanding.

As LINAK only supplies the derivation cabling from actuator to the nearest TAP in the trunk, the longer distance cabling that interconnects the individual TAPs is the responsibility of the customer/ system integrator.

The following cable characteristics must be fulfilled when selecting the trunk cabling:

Electrical requirements according to TSB89, Application Guidelines for TIA-EIA-485-A which is typical:

- 24 AWG shielded twisted pair
- 120 Ω characteristic impedance
- Capacity less than 40 pF/meter
- Drain wire as common signal reference

Installation rules

Grounding arrangements

The RS485 Signal Common must be connected directly to protective ground, at one point only for the entire bus. Generally this point is to be chosen on the master device or on its TAP, preferably using the power supply GND (negative supply). It is highly recommended that the RS485 interface on the master is chosen to be an isolated type.

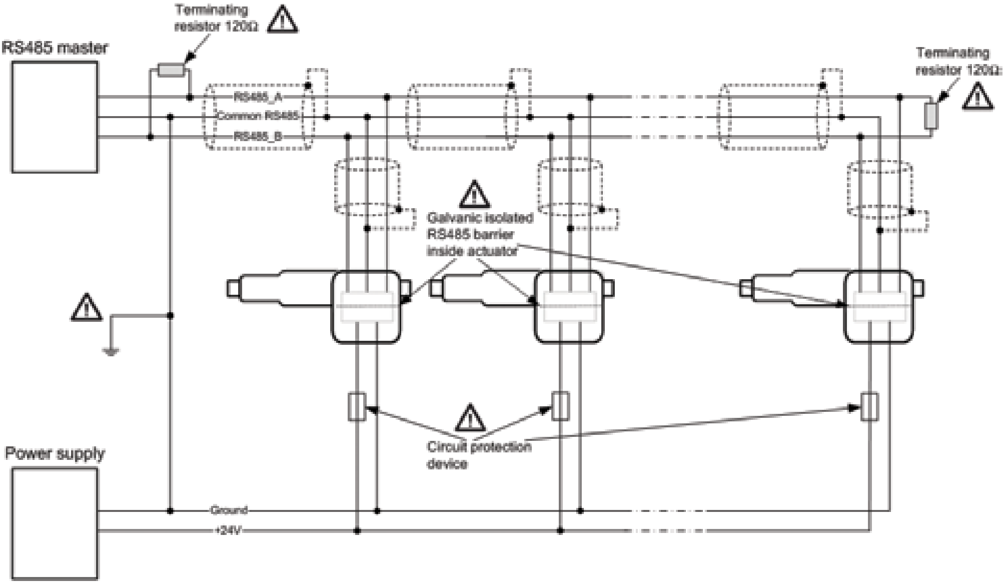
It is not allowed to connect the RS485 Signal Common to the power supply GND at the actuator nodes. Doing so will cause high ground currents to flow in the « Common » circuit.

Line termination

A reflection in a transmission line is the result of an impedance discontinuity that a travelling wave sees as it propagates down the line.

To minimise the reflections from the end of the RS485-cable it is required to place a Line Termination (LT) near each of the 2 ends of the Bus.

The termination is made with a 120 Ohms (0.25 W) resistor (see figure below).


Terminating resistor 120Ω
RS485 master
Terminating resistor 120Ω
RS485_A
Common RS485
RS485_B
Galvanic isolated RS485 barrier inside actuator
Circuit protection device
Power supply
Ground
+24V

Line biasing

No line biasing is necessary for the LINAK MODBUS actuators to function, as their isolated RS485-transceivers are true fail-safe, meaning that they correctly handle an idle bus (no enabled driver on the bus).

Cable length and number of slaves

Even though a MODBUS master can logically address up to 247 nodes in a multipoint serial line there are several parameters that have an influence on the maximum number of slaves in a system.

The RS485 standard basically specifies a system consisting of 32 slaves without the use of repeaters. Modern transceivers allow for even more, as their bus load is less than the maximum allowed by the RS485 standard. The LINAK MODBUS actuators use such modern transceivers, theoretically supporting up to 256 nodes. In real life this number is reduced by the below listed operating conditions:

- The baud rate
- The total length of the trunk cable
- The total length of the derivation cables
- The quality of the cabling
- The functionality i.e. the poll rate

Based on our tests, a bus string with a master, 32 LINAK actuators and no 3’rd party slaves will work under the following conditions:

- Baud rate: 19200
- Data cable: Following the guidelines in the following section DATA cable
- Derivation cables: Standard LINAK cable supplied with MODBUS actuator, not extended
- Termination: In both ends of string
- Total string length: 1000 m

ⓘ If your MODBUS system either:

- includes more than 32 actuators
- the trunk cable exceeds 1000 m
- the derivation cables are extended
- includes 3rd party slaves

..then you have to test if the installation needs repeaters or other actions to provide a stable communication level.

Wiring in general

☝ Be careful not to run any AC cabling in parallel with the wiring for the actuator system as this will create a noisier environment.

Power up rules

☝

- It is preferable not to have power on the power line when you connect or disconnect the actuators.
- Fuse selection: It is recommended to install a fuse for each actuator in a string (prevents a dead string (series of actuators) in case of an error (e.g. short circuit)). Alternatively, a fuse for each power supply must be installed. It is also recommended to use a current limited power supply.

Reverse polarity protection

In the case of reversing the polarity of the power to the actuator, a protection diode gets into action (short circuiting the input voltage). The result is a very high current. As the protection diode has a limited current handling capability, the fuse mentioned above must open in order to limit the duration of the short circuit.

The following table shows the allowed reverse current at different current pulse durations. The duration of the pulse is determined by the time it takes for the fuse to open:

Pulse duration (ms)	Max allowable current (A)
10	120
50	100
100	90
500	80
1000	80
Infinite	7

If the values in the above table are exceeded, the actuator will take permanent damage.

This is used when selecting the power supply and the fuse types for a string of actuators. The power supply must be a current limiting type and the maximum operating time* for the selected fuse must be known at the current limit of the power supply. (For further explanation read the following examples):

Example 1
The selected power supply has a current limit of 40A. The selected fuse protecting each actuator has a nominal rating of 10A and a guaranteed opening time at 400% rated of maximum 400ms. If the actuator is connected with reverse polarity, the fuse (and the reverse polarity diode in the actuator) will see 40A of current. At this level, the current will be interrupted after 400ms maximum. We can see from the table that this is OK, as a current pulse of up to 80A with duration of 500ms is allowed.

Example 2

The selected power supply has a current limit of 100A. The selected fuse protecting each actuator has a nominal rating of 10A and a guaranteed opening time at 1000% rated of maximum 150ms. If the actuator is connected with reverse polarity, the fuse (and the reverse polarity diode in the actuator) will see 100A of current. At this level, the current will be interrupted after 150ms maximum. We can see from the table that this is not OK, as a current pulse of up to 100A is only allowed to have a duration of 50ms.

* Usually stated at 150%, 210%, 275%, 400% and 1000% of the rated current.

ⓘ When using soft-stop on a DC-motor, a short peak of higher voltage will be sent back towards the power supply. It is important when selecting the power supply, that it does not turn off the output, when this backwards load dump occurs.

Modbus RTU Functional Description

Implementation class

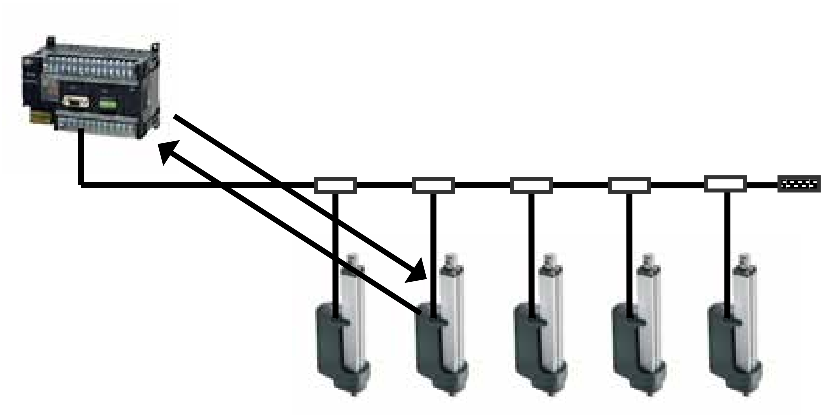
The LINAK MODBUS protocol is implemented conforming to the “basic slave” implementation class as described in document [1]. The following options have been implemented:

Parameter	**Options**	**Default value**	**Remarks**
Adressing	Configurable from 1 to 246	247 (= un-assigned)	Configured by BusLink software
Broadcast	Yes		
Baud rate	9.6 kBaud - 115.2 kBaud	19.2 kBaud	Configured by BusLink software
Parity	Even, odd, no-parity	Even	Configured by BusLink software
Stop Bits	1, 2	1	Configured by BusLink software. Notice that a No-parity setting requires 2 stop bits
Mode	RTU	-	Not configurable
Electrical Interface	RS485 2W-cabling	-	Not configurable
Connector Type	M12	-	Not configurable

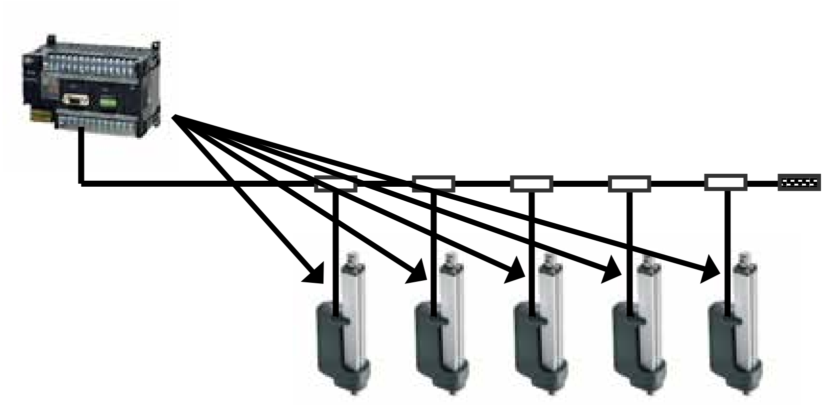
Unicast/broadcast

MODBUS is a single master system, which means that only one master can be connected at any single point in time. Two modes of communication are possible, Unicast and Broadcast.

Request/response (unicast)
The requests from the master are addressed to a given slave. The master then waits for the response from the slave which has been interrogated. In this mode the transaction consists of 2 messages: a request from the master and a response from the slave.



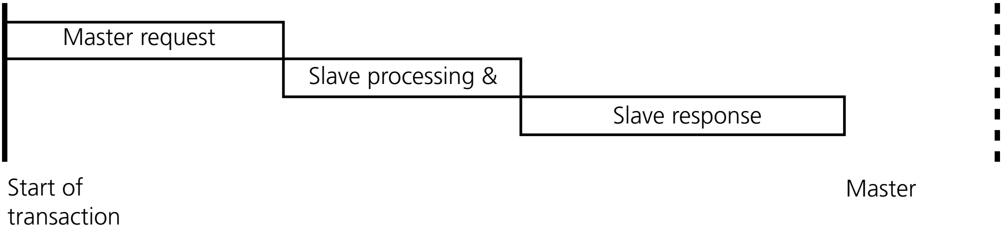
Broadcasting
The master broadcasts a message to address "0", which means that the information is for all slave devices on the network. These stations execute the order without transmitting a response.



Response time

The slave device will respond on each valid MODBUS request from the master within a time which is dependent on the setting of parameter 'MODBUS Response Delay' (Input Register 59). With a default parameter value (3 ms) the max. response time is 18 ms (3 + 15 ms) – when increasing the parameter value worst case is 115 ms (100 + 15 ms).

Modbus message timing:



The MODBUS response timeout setting of the master should be set to a value larger than the calculated max. response time.

Error checking

MODBUS RTU networks employ two methods of error checking:

1. Parity checking of each data character (even, odd, or no parity)
2. Frame checking within the message frame (Cyclical Redundancy Check)

Parity checking
A LINAK MODBUS device can be configured for even or odd parity, or for no parity checking. This determines how the parity bit of the character's data frame is set. If even or odd parity checking is selected, the number of 1 bits in the data portion of each character frame is counted. Each character in RTU mode contains 8 bits. The parity bit will then be set to a 0 or a 1, to result in an even (even parity), or odd (odd parity) total number of 1 bits.

The use of 'no parity' requires 2 stop bits.

Frame checking
RTU Mode message frames include an error checking method that is based on a Cyclical Redundancy Check (CRC). The error-checking field of a message frame contains a 16-bit value (two 8-bit bytes) that contains the result of a Cyclical Redundancy Check (CRC) calculation performed on the message contents.

Message format

Any MODBUS message consists of the basic fields shown below: Slave address (Addr), Function code (Function), up to 252 data bytes (Data) and a calculated 16 bit checksum (CRC).

Byte 0	Byte 1	Byte 2 .. N	Byte N+1, N+2
Addr	Function	Data	CRC

Messages start with a silent interval of at least 3.5 character times – at the actual communication baud rate. The first field transmitted is the device address. Following the last transmitted byte, a similar interval of at least 3.5 character times marks the end of the message. A new message can begin after this interval.

Address field

The address field Addr is one byte long. Valid slave addresses are 1 – 246. Value 0 and value 247 to 255 is reserved for special purposes. A master addresses a slave by placing the slave address in the Addr field of the message. When the slave responds it places its own address in the Addr field to let the master know which slave is responding.

ⓘ Each slave device must have assigned a unique address (from 1-246) so that it can be addressed independently from other nodes. Value 0 is reserved for broadcast messages which all slaves recognise.

Function field

The function field Function is one byte. Supported MODBUS functions are 3, 4, 6, 16. When a message is sent from the master to a slave, the function field code tells the slave what kind of action to perform.

When the slave responds to the master, it uses the function field to signal either a normal, error-free response or an exception response. For a normal response the slave simply echoes the original function code. For an exception response the slave returns a code that is equivalent to the original function code with the most significant bit set. In addition the slave adds a unique code into the data field of the message telling the master what kind of error occurred.

Data field

The data field is of varying length. The data field of the message sent from the master to the slave contains additional information which the slave must use to take the action requested by the function field. This can include items like register addresses, quantity of register to handle, and the count of actual data bytes in the message.

CRC field

The CRC field is 2 bytes long. The CRC value is calculated by the transmitting device, which appends it to the end of the message. The receiving device recalculates a CRC during receipt or the message and compares the calculated value to the actual value received in the CRC field. In the case of a difference an exception response is returned.

Register-parameter mapping

All data addresses in MODBUS messages to LINAK actuators are referenced to zero. The input register known as e.g. ‘Input Register 30002’ in a programmable controller is addressed as register 1 in the Addr field of a MODBUS message. The function code of the message already specifies an ‘Input Register’ operation and therefore the ‘3xxxx’ reference is implicit.

In the same way the holding register known as e.g. ‘Holding register 40008’ is addressed as register 7 in the Addr field of the message, together with a relevant Holding Register function code.

All MODBUS registers are default mapped to a specific actuator parameter. To optimise communication throughput when reading or writing non-contiguous register sets, it is possible to re-define this mapping (as described later in this guide).

Data formats

Application information communicated between MODBUS master and slave is organised as one or more 16-bit Registers. Different datatypes are mapped into theses addressable registers. The type of any parameter value embedded into the MODBUS message has to be recognised according to the register/parameter tables in Appendix A. LINAK MODBUS devices support the following datatypes (illustrated by single register write-message examples):

Short integer register (U8)

Status bytes and small integer values are stored in MODBUS registers where only half of the register is utilised. Integer values from 0 to 255 are stored in the least significant byte of the register.

Byte 0	Byte 1	Byte 2 .. 5				Byte 6, 7	
Addr	Function	Register address		Register data		CRC	
		MSB	LSB	N/A	U8		

Unsigned integer register (U16)

Integer values from 0 to 65,535 are stored in the normal 2-byte MODBUS register. The most significant byte of the value is sent first.

Byte 0	Byte 1	Byte 2 .. 5				Byte 6, 7	
Addr	Function	Register address		Register data		CRC	
		MSB	LSB	U16-MSB	U16-LSB		

Signed integer register (S16)

Integer values from -32,768 to 32,767 are stored in the normal 2-byte MODBUS register. The most significant byte of the value is sent first.

Byte 0	Byte 1	Byte 2 .. 5				Byte 6, 7	
Addr	Function	Register address		Register data		CRC	
		MSB	LSB	S16-MSB	S16-LSB		

Long integer registers (U32)
Some integer values used by the actuator are larger than 65,535, which is the largest number that can be stored in a single MODBUS register. In these cases, the unsigned value is stored in two consecutive integer registers enabling values up to 4,294,967,295. These “long integer” registers are potentially accessed using the MODBUS functions 3, 4, and 16 (see below). The most significant word is stored in the lower register (sent first), and the least significant word is stored in the higher register.

Byte 0	Byte 1	Byte 2 .. 7						Byte 8, 9	
Addr	Function	Register address		Register data				CRC	
		MSB	LSB	U32-MSB			U32-LSB		

Function codes

The LINAK actuator supports a subset of the standard MODBUS RTU function codes to provide access to the internal actuator parameters and functions.

The LINAK MODBUS protocol does not support the Diagnostic function (function code 08). As a more general approach, the MODBUS master – and the BusLink service tool – can read a large amount of service counter input registers.

Function code 3 – Read Holding Registers
Function code 3 is used to read one or more holding registers in the actuator, referenced in Appendix A. When the master access a register that is not supported by the slave it responds with an exception message.

The register address in this context is without ‘4xxxx’ identification. E.g. the ‘Target Position’ holding register is read via register address 1 – in the PLC often referenced as 40002.

Broadcast is not supported.

Request message:

Addr	Function	Register address		Register count		CRC	
Slave Address	Function Code (=3)	Starting Address High	Starting Address Low	No. of Registers High	No. of Registers Low		

In the response message the requested values are delivered to the master. The data bytes are organised according to data type as explained above.

Response message:

Addr	Function		Data	CRC	
Slave Address	Function Code (=3)	Byte Count	Data byte 1 .. N		

Function code 4 – Read Input Registers
Function code 4 is used to read one or more input registers in the actuator, referenced in Appendix A. When the master accesses a register that is not supported by the slave, it responds with an exception message.

The register address in this context is without ‘3xxxx’ identification. E.g. the ‘Distance from Target’ input register is read via register address 3 – in the PLC often referenced as 30004.

Broadcast is not supported.

Request message:

Addr	Function	Register address		Register count		CRC	
Slave Address	Function Code (=4)	Starting Address High	Starting Address Low	No. of Registers High	No. of Registers Low		

In the response message the requested values are delivered to the master. The data bytes are organised according to data type as explained above.

Response message:

Addr	Function			Data		CRC	
Slave Address	Function Code (=4)	Byte Count		Data byte 1 .. N			

Function code 6 – Write single Holding Register
Function code 6 is used to write a new value to a holding register of the actuator, referenced in Appendix A. When the master accesses a register that is not supported by the slave or when it tries to write a value outside defined boundaries the slave responds with an exception message.

The register address in this context is without ‘4xxxx’ identification, e.g. meaning address 0 equals register 40001.

Broadcast is supported (but any response – including exceptions - is discarded).

Request message:

Addr	Function	Register address		Register data		CRC	
Slave Address	Function Code (=6)	Address High	Address Low	New Value High	New Value Low		

The normal response is an echo of the request message.

Response message:

Addr	Function	Register address		Register data		CRC	
Slave Address	Function Code (=6)	Address High	Address Low	New Value High	New Value Low		

Function code 16 – Write Multiple Holding Registers
Function code 16 is used to write one or more holding registers in the actuator, referenced in Appendix A. When the master accesses a register that is not supported by the slave it responds with an exception message.

The register address in this context is without '4xxxx' identification, e.g. meaning address 0 equals register 40001.

Multiple registers are written as an entity with commands executed as the final stage. That means, if e.g. Holding Register 1, 2 and 3 (Target Position, Command-Remote and Max Speed) are written in one single command then the actuator will start running towards the new position with the new speed.

Broadcast is supported (but any response – including exceptions - is discarded).

Request message:

Addr	Function	Register address		Register count			Register Data	CRC	
Slave Address	Function Code (=3)	Starting Address High	Starting Address Low	No. of Registers High	No. of Registers Low	Byte Count	New Value 1 .. N		

The normal response message returns the slave address, function code, starting address and the quantity of registers written.

Response message:

Addr	Function	Register address		Register count		CRC	
Slave Address	Function Code (=16)	Starting Address High	Starting Address Low	Register Count High	Register Count Low		

Exceptions

When a MODBUS master transmits a request to a slave the expected behaviour is that the slave responds with a normal response. But several error scenarios are possible:

- The slave does not receive the request due to a communication error. No response is returned; the master will process a time-out and eventually repeat the request.
- The slave receives the request but detects a parity or CRC communication error. No response is returned; the master will process a time-out and eventually repeat the request.
- The slave receives the request without a communication error but cannot handle it. The slave will in that case return an exception response informing the master about the nature of the error.

Exception response message:

Addr	Function	Error	CRC	
Slave Address	Function Code +80 $_{\text{hex}}$	Error Code		

Exception codes are:

Exception Code	Definition	Description
01	Invalid Function	The message received is not an allowable action
02	Illegal Data Adress	The register address(es) referenced in the function-dependent data section of the message is not valid
03	Illegal Data Value	The data value of the referenced address is not within limits
04	Slave Device Failure	The addressed device is not able to process a valid message due to a bad device state

Typical Use Cases

In this section further descriptions of how to communicate with the LINAK LA35/LA36/LA37 Actuators are shown. The examples are typical user scenarios and application solutions. All examples include references to registers, which are further described in details in Appendix A.

Configuration

Before integration into a MODBUS system a few parameters of the actuator have to be checked and eventually changed. This preparation is done by use of the BusLink PC tool (the tool is described in details later) and guarantees that the actuator is able to execute basic functionality. Further fine-tuning may be required to fulfil system-or application requirements.

Parameters to be verified by BusLink:

Parameter	Description
Address	Set the MODBUS device address to a unique value between 1 and 246.
Baudrate	The baudrate parameter is set to the communication speed required by the PLC/system.
Parity	The parity parameter is set to the value required by the PLC/system.
Stopbits	The stopbits parameter is set to the value required by the PLC/system.
Current limit in	Depending on the load in the specific application it might be necessary to adjust this parameter. If the motor current exceeds this limit, the actuator will not move.
Current limit out	Depending on the load in the specific application it might be necessary to adjust this parameter. If the motor current exceeds this limit, the actuator will not move.
Max speed	Set the actuator speed according to application requirements.

Run to taget

Before you move the actuator to any new position you have to verify that some general prerequisites are fulfilled. Timing (e.g. when the actuator is still moving), environment conditions and errors might mean that the actuator is in a state where further operation is not possible.

General run-prerequisites:

Step	Register *	Action
1	IR 4	'Status Register 1, bit 4 (Ready)' has to be = 1.
2	IR 4	'Status Register 1, bit 9 (LOCAL connected)' has to be = 0.
3	IR 4	'Status Register 1, bit 10 (Position Valid)' has to be = 1.

*IR = Input Register, HR = Holding Register

When the actuator wakes up after power down and sees that the state of the two hall sensors has changed, the “position valid” bit will be set to “0”. On firmware versions 1.2 and lower, it is required to do an initialisation run in either inwards or outwards direction. On firmware versions 1.3 and higher, it is NOT required to do an initialisation run, but it is recommended, to be sure that the position is valid.

The communication sequence to position the actuator to a given position is:

Step	Register *	Action
1	-	Check that general run-prerequisites are fulfilled.
2	HR 1	Write a value into ‘Target Position’ parameter (unit is 1/10 mm relative to offset).
3	HR 2	Write a 1 (Run to Target Position) into ‘Command, Remote’ parameter. The actuator will now start to move towards the target position.
4	IR 3 IR 5	If you want the acknowledge of a successful positioning you should read and wait until ‘Run Status’ has become 0 (Idle) and then verify that ‘Reason for Last Stop’ = 0 (Target position reached). This step can alternatively be part of the general run-prerequisites above as verification of the previous positioning - to avoid inefficient busy-looping.

*IR = Input Register, HR = Holding Register

Run to predefined positions

It is possible to pre-define up to 4 different actuator positions and then have the actuator switch between these positions by simply sending one command.

Step	Register *	Action
1	-	Check that general run-prerequisites are fulfilled.
2	HR 6, 7, 8 or 9	Write one to four values into ‘Reference Position 1,2,3 ad/or 4’ (unit is 1/10 mm relative to offset).
3	HR 2	Write a 6, 7, 8 or 9 into ‘Command, Remote’ parameter. The actuator will now start to move towards either the pre-set position 1, 2, 3 or 4.

*IR = Input Register, HR = Holding Register

Ramp down before target

Depending on the application it may be convenient to slow down actuator speed, getting close to the target. By setting the ‘Ramp Down Before Target’ to a different distance (default value is 20, corresponding to 2 mm) you can control when this slow down takes place.

Step	Register *	Action
1	HR 17	Write a value into ‘Ramp Down Before Target’ parameter (unit is 1/10 mm).

*IR = Input Register, HR = Holding Register

Activation of multiple actuators

In some large MODBUS communication systems the sequential and time-consuming polling of the slaves by the master makes it hard to command multiple actuators to move at the same time. The LINAK LA35/LA36/LA37 MODBUS actuator offers a feature to improve this functionality. Below is shown the process to start 3 actuators.

Step	Register *	Action
1	-	Check that general run-prerequisites are fulfilled in all 3 actuators.
2	HR 1	Write a value into ‘Target Position’ parameter (unit is 1/10 mm relative to offset) of each actuator.
3	HR 10, 11	Clear the internal clock by writing a 0 (or set to a desired value) into ‘Current Time (s/m)’ parameters of all actuators by use of a broadcast request.
4	HR 4 and 5	Write the exact trigger time into ‘Next Time to Run (s/m)’ parameters of each actuator.
5	HR 2	Write a 3[1] (Run at “Next time to run”) into ‘Command, Remote’ parameter of each actuator. The actuators are now armed to start moving towards the target position when the specified absolute time arises (HR 4 = HR 10 and HR 5 = HR 11).[2]

*IR = Input Register, HR = Holding Register

- The steps, as described above, must be performed in the right sequence (i.e. write to HR 5 as the last step) in order to obtain the correct functional behaviour.
- The power-line (supply and cabling) has to be dimensioned for simultaneous run of all started actuators.

[1] The “Next time to run” command must be sent minimum 2 sec before the trigger (the trigger is HR4 = HR10 and HR5 = HR11)

[2] When waiting for next time to run (IR 1 = 3) the function will be interrupted if a new value is written to ‘Command, Remote’ (HR 2)

Retrieval of service data

If you want to supervise detailed working conditions of the actuator in an operational MODBUS system a number of input registers provide service counters and values. These registers are organised so that you can read them in a single MODBUS request. Notice that all counters are non-resetable and they display the number of events measured in the actuators total life-time.

Step	Register *	Action
1	IR 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47	Execute the command: Read input registers, Register Address = 26, Register Count = 22. Double register 26,27: Number of Power Fails Double register 28,29: Total Number of Starts Inwards Double register 30,31: Total Number of Starts Outwards Double register 32,33: Total Running Time [s] Register 34: Max Current Measured [mA] Register 35: Max FET Temperature Measured [°C] Register 36: Min Actuator Temperature Measured [°C] Double register 37,38: Performed Work [A*s] Double register 39,40: Number of Telegrams With Invalid Content Double register 41,42: Number of Telegrams With CRC Error Double register 43,44: Total Number of Current Cut-Offs out Double register 45,46: Total Number of Current Cut-Offs in Register 47: Number of Times Running with Actuator Temperature Exceeded

*IR = Input Register, HR = Holding Register

Customisation of MODBUS register mapping

To optimise efficiency of the MODBUS communication between a master and the LINAK LA35/LA36/LA37 Actuator(s) the device offers an opportunity to re-arrange the numbering of Input – and Holding Registers. Appendix A shows the default mapping between registers and internal actuator parameters.

It is possible to re-arrange the register mapping by use of the special holding registers 1001 to 1005 in the actuator. If you e.g. want to read parameter 4 ('Run Status', default mapped to input register 1) and parameter 56 ('Total Running Time', default mapped to input register 32, 33) in a single MODBUS request/response transaction, you have to change the mapping as e.g. shown in the example below.

In the example we will set up input registers 74, 75, 76 to read the two parameters 'Run Status' (16 bit) and 'Total Running Time (32 bit).

Step	Register *	Action
1	HR 1001	Specify input register 74, which you want to re-map by writing the number 74 to holding register 1001. If you now read holding register 1002 it will show the existing mapping to parameter 29.
2	HR 1002	Specify the new mapping of input register 74 by writing the number 4 into holding register 1002 - as new reference to parameter 4 ‘Run Status’
3	HR 1001	Specify the next input register 75, which you want to re-map by writing the number 75 to holding register 1001. If you now read holding register 1002 it will show the existing mapping to parameter 30.
4	HR 1002	Specify the new mapping of input register 75 by writing the number 32 into holding register 1002 - as new reference to parameter 56 ‘Total Running Time, most significant part’
5	HR 1001	Specify the input register 76, which you want to re-map by writing the number 76 to holding register 1001. If you now read holding register 1002 it will show the existing mapping to parameter 31.
6	HR 1002	Specify the new mapping of input register 76 by writing the number 33 into holding register 1002 - as new reference to parameter 56 ‘Total Running Time, least significant part’
7	HR 1005	If you want to save this new mapping into non-volatile memory of the device then you can write the command 1 (‘Save Modbus User Map’) into holding register 1005.
8	HR 1005	If you anytime – now or later – want to return to the factory default mapping of input- and holding registers then you can write the command 2 (‘Set Modbus User Map to Default’) into holding register 1005.

*IR = Input Register, HR = Holding Register

By re-defining the mapping of MODBUS registers, the original mapping of these registers is destroyed.

Writing to holding registers can be optimised in an exactly similar way by using another set of control registers HR 1003 and HR 1004 instead of HR 1001 and HR 1002.

BusLink Configuration Software

The Bus actuators from LINAK can be configured via the software tool named “BusLink.” (if you don’t have BusLink please contact your local LINAK office).

BusLink is a tool that you can use to set the parameters, run initialisation, or upload new firmware etc. The next pages will give you a guide on to how to use the software tool.

In order to connect your actuator to the PC-tool you will need 2 pcs of separate cables:

- 1 pc. USB2LIN cable and 1 pc. Interface cable (has to be ordered from LINAK).
  The cables can be ordered in one package with Axapta item number 0367998.

When changing the cables on a LINAK actuator, it is important that this is done carefully. In order to protect the plugs and pins, please be sure that the plug is in the right location and fully pressed in before the cable lid is mounted.

As power is supplied through the USB connection the actuator does not have to be separately powered up (24V) to be configured. However if you want to run with it in “manual run” then you need to supply the actuator with external power (24V).

USB2LIN cable

Interface cable

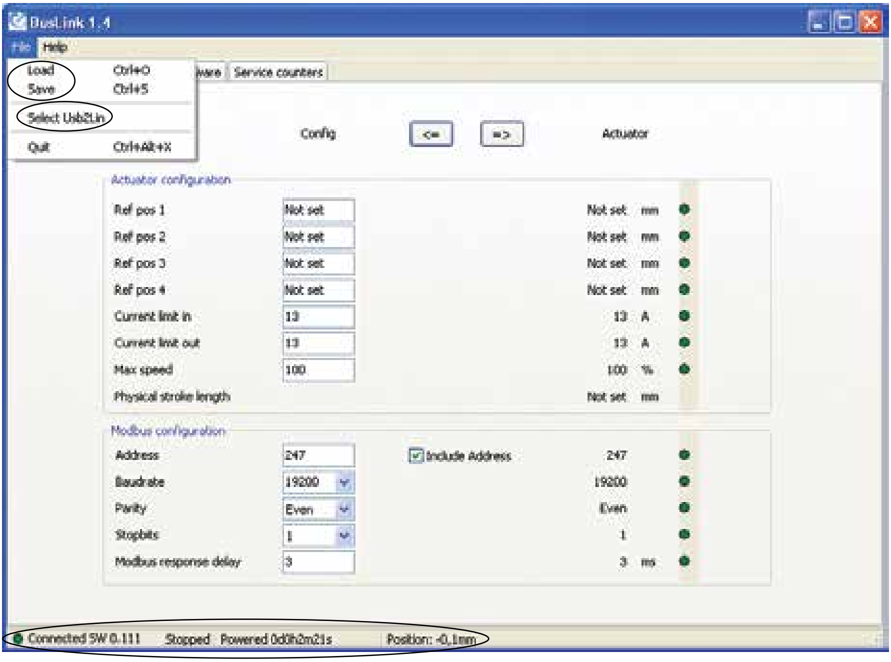
Screen nr. 1: ”Program initialisation”

In the “file” menu it is possible to “Save” and “Load” configurations on your PC. This can save you time if you e.g. have to set-up the same configuration in many actuators.

- “Save” means that the actual configuration (left side of tab “Configuration”) is saved to a user definable file on your PC.
- “Load” means you can select and read any previously saved configuration file.

- "Select Usb2Lin" can be used if you e.g. have several Usb2Lin cables connected to your PC.
- In the status area in the bottom of the screen you can se the actual connection status. If an actuator is connected you can additionally se the firmware version of the actuator, the actual run-status, clock value and position.

ⓘ 'Configurations' in this context mean the 12 parameters you see on the left part of the 'Configuration tab' – not the complete actuator setting.



Actuator configuration

	Config	Actuator	
Ref pos 1	Not set	Not set mm	●
Ref pos 2	Not set	Not set mm	●
Ref pos 3	Not set	Not set mm	●
Ref pos 4	Not set	Not set mm	●
Current limit in	13	13 A	●
Current limit out	13	13 A	●
Max speed	100	100 %	●
Physical stroke length		Not set mm	

Modbus configuration

	Config		Actuator	
Address	247	☑ Include Address	247	●
Baudrate	19200		19200	●
Parity	Even		Even	●
Stopbits	1		1	●
Modbus response delay	3		3 ms	●

● Connected SW 0.111 Stopped Powered 0d0h2m21s Position: -0,1mm

Screen nr. 2: "Configuration tab"

Generally the left part of this screen shows actual configuration data (loaded and may be changed from file or actuator). The right part of the screen shows the actual actuator values (if any actuator is connected). The green and red signals to the right indicates if actuator and PC values are equal or unequal.

With the two arrows in the top of the screen you can upload and download values between the BusLink configuration tool and the actuator (the arrows become active when an actuator is connected).

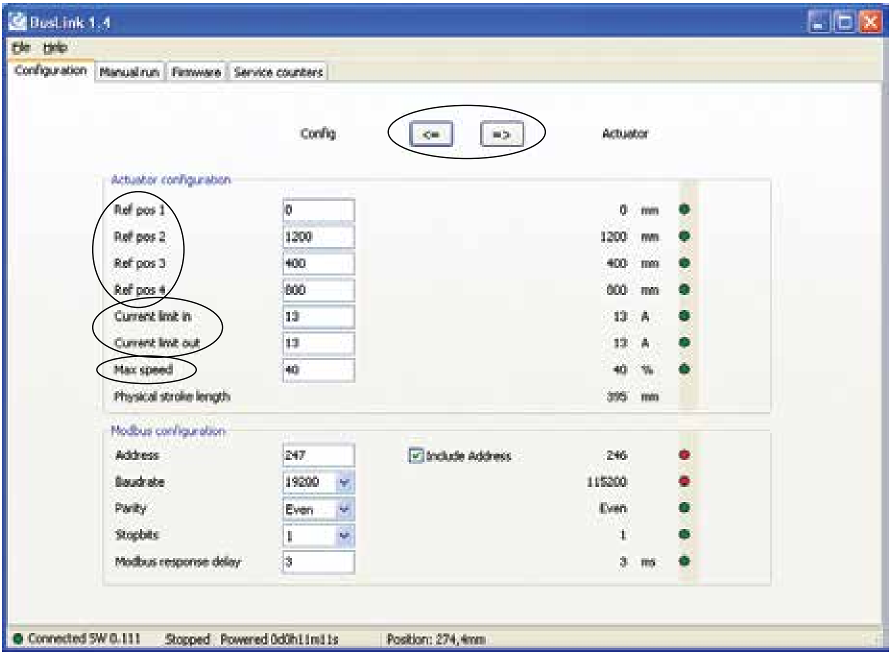
All values written to the actuator are validated according to the defined limits. If you try to write a value outside the limits it will be rejected.

In the “Modbus configuration” section of the screen you can set up the communication parameters of the actuator.

- ”Address” is from the factory set to 247. Remember that each unit must have its own unique address. Do not use 247 or 0 as an address. It is possible to avoid upload/download of the address value by unchecking the “Include Address” box.
- ”Baudrate”, “parity”, “stop bits” has to be set to specified, common values across the MODBUS system. No-parity requires 2 stop bits.
- ”Modbus response delay” is used in cases where the MODBUS master is unable to pickup too fast responses from the slave. In case of communication problems you can increase this value.

In the “Actuator configuration” section of the screen you can set up some basic parameter values.

- ”Ref pos 1 – 4” are used to define 4 different set points as you prefer. This could e.g. be a safe position in case of storm, or maintenance position for cleaning etc.
- ”Current limit” can be set in both directions. On the “Manual run” tab it is possible to see the max current measured in running mode; then you can set the current limit according to the measured max current.
- ”Max speed” may be specified from 10 - 100%.



Actuator configuration

	Config	Actuator
Ref pos 1	0	0 mm ●
Ref pos 2	1200	1200 mm ●
Ref pos 3	400	400 mm ●
Ref pos 4	800	800 mm ●
Current limit in	13	13 A ●
Current limit out	13	13 A ●
Max speed	40	40 % ●
Physical stroke length		395 mm

Modbus configuration

	Config		Actuator
Address	247	☑ Include Address	246 ●
Baudrate	19200		115200 ●
Parity	Even		Even ●
Stopbits	1		1 ●
Modbus response delay	3		3 ms ●

● Connected SW 0.111 Stopped Powered 0d0h11m11s Position: 274,4mm

Screen nr. 3: ”Manual run”

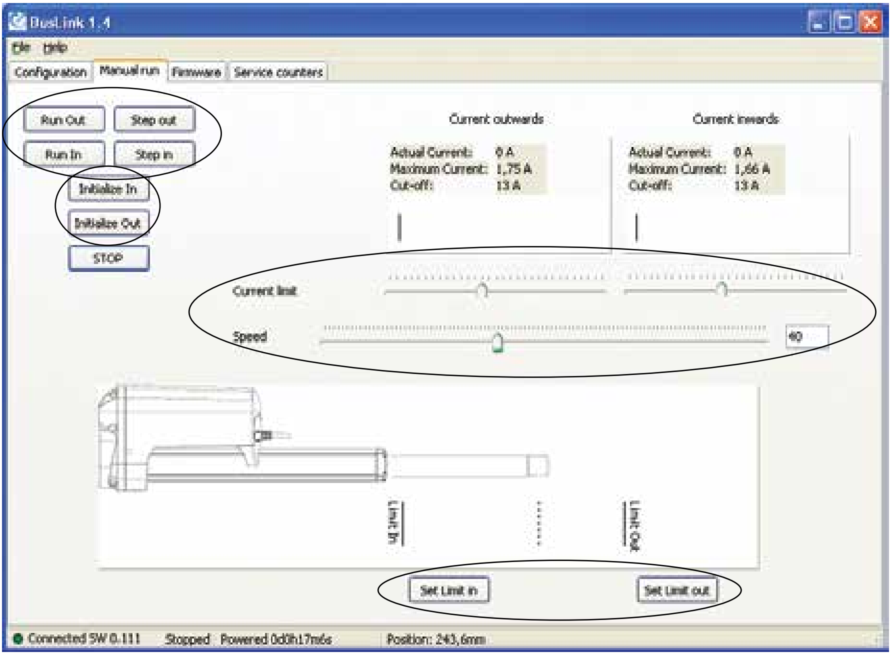
This tab is used to set the basic movement parameters of the actuator.

- “Run Out”/”Run In” will start the actuator in the selected direction and it will run until it is either stopped or until it reaches an end-stop.
- “Step Out”/”Step In” is used to adjust the position in small steps in either outgoing or ingoing direction.
- “Initialise In”/”Initialise Out” is used to repeat the initialisation as done in factory.
- “Current limit In/Out” and maximum “Speed” for the manual run can be controlled by moving the sliders. This speed setting is not persistent – after the manual run the setting is reset to previous value (except if you disconnect the data cable during the run).
- “Set Limit in”/”Set Limit out” is used to define the operational position-limits after having positioned the actuator to the position where you want these limits.
  Please be aware to use ONLY one virtual endstop (‘Limit in’ or ‘Limit out’) at a time, in order to ensure that one end is always open for initialisation of the actuator.

Initialise In/Out and Run Out/In will not respect the Limit in/out setting.
Be aware of potential mechanical destruction.

Actuator current is monitored in the upper-right part of the screen. Here you can see the actual and maximum measured current in both outgoing and ingoing direction - together with actual cut-off values.

The actuator figure in the lower part of the screen visually shows the actual position of the actuator (dashed line) together with the actual setting of Limit In (corresponding to off-set) and Limit Out (corresponding to Max. stroke length).



Current outwards

Actual Current: 0 A
Maximum Current: 1,75 A
Cut-off: 13 A

Current inwards

Actual Current: 0 A
Maximum Current: 1,66 A
Cut-off: 13 A

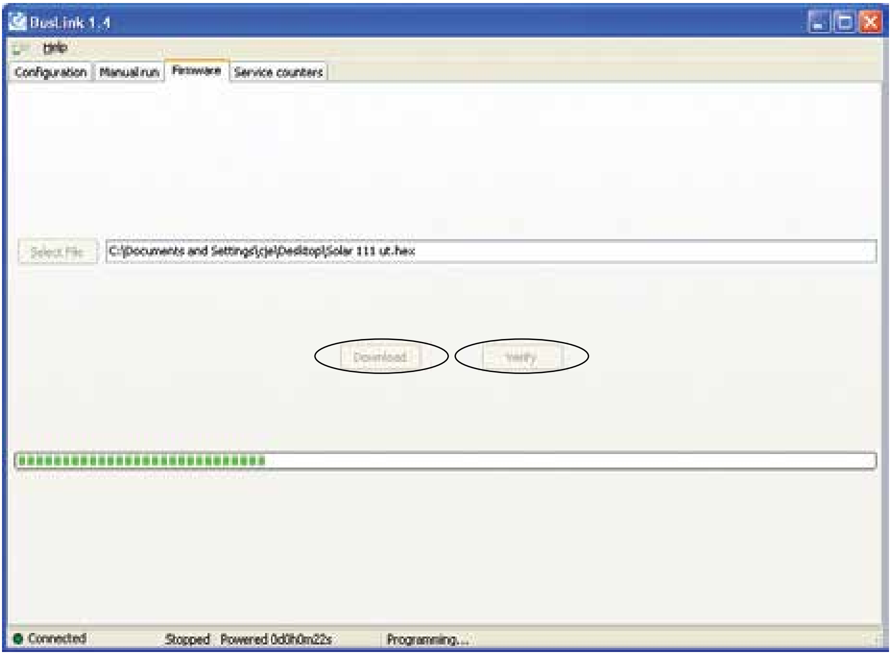
Speed: 40

Connected SW 0.111 Stopped Powered 0d0h17m6s Position: 243,6mm

Screen nr. 4: ”Firmware”

On this tab it is possible to install firmware into the actuator, if a new firmware version is released.

- All you need to do is to select the new firmware file and press “Download.”
- “Verify” means that you can verify the previously downloaded firmware. Verification is a time-consuming process and is normally not needed after a successful download.



C:\Documents and Settings\cje\Desktop\Solar 111 ut.hex

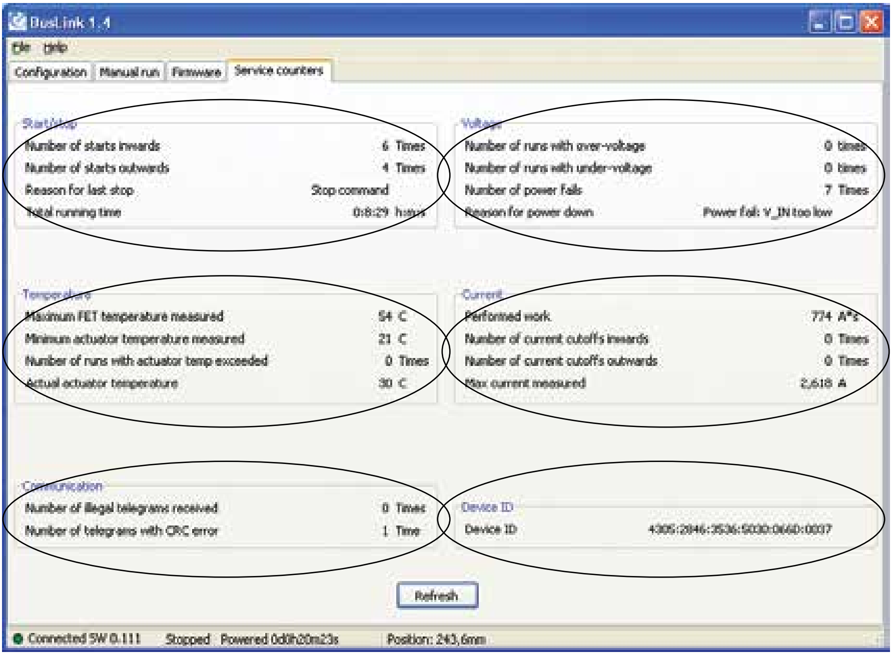
● Connected Stopped Powered 0d0h0m22s Programming...

Screen nr. 5: "Service counters"

The service counter tab is divided into 4 areas:

- Start/Stop data where it is possible to see the total running time, number of actuator starts and stops and the reasons.
- Temperatures, here it is possible to read the relevant temperatures measured on the FET transistors and the processor.
- In the current section it is possible to see the performed work, current cut-offs etc.
- In the voltage section the number of power fails and the reason for power down.
- In the communication section you can read if there has been any communication problems.
- For better traceability in an service situation a unique device ID can be utilised as reference

The service counters are not reset-able and will be valid during all actuator life-time.



Start/stop

Number of starts inwards 6 Times
Number of starts outwards 4 Times
Reason for last stop Stop command
Total running time 0:8:29 h:m:s

Voltage

Number of runs with over-voltage 0 times
Number of runs with under-voltage 0 times
Number of power fails 7 Times
Reason for power down Power fail: V_IN too low

Temperature

Maximum FET temperature measured 54 C
Minimum actuator temperature measured 21 C
Number of runs with actuator temp exceeded 0 Times
Actual actuator temperature 30 C

Current

Performed work 774 A*s
Number of current cutoffs inwards 0 Times
Number of current cutoffs outwards 0 Times
Max current measured 2,618 A

Communication

Number of illegal telegrams received 0 Times
Number of telegrams with CRC error 1 Time

Device ID

Device ID 4305:2846:3536:5030:066D:0037

Connected SW 0.111 Stopped Powered 0d0h20m23s Position: 243,6mm

Troubleshooting

Integration of a LINAK LA35/LA36/LA37 Actuator into a MODBUS system is fairly simple. But sometimes you may potentially encounter unexpected results or strange behaviour. In the following section you will find some potential problems, diagnosis suggestions, possible causes and corresponding solutions.

Problem: The master does not get any response from the actuator

Diagnosis: Check with BusLink: ‘Configuration / Modbus configuration’ setting.

Causes/solutions:

Cause: The power- or communication cabling is not implemented as specified.
Solution: Inspect cabling and repair.

Cause: Communication baudrate, parity, stop bits are not set correctly.
Solution: Set up communication parameters as required by use of BusLink.

Cause: The device does not have the expected slave address.
Solution: Set up the slave address between 1 and 246 by use of BusLink.

Problem: The master does not reliably get responses from the actuator(s)

Diagnosis: Check with BusLink: ‘Service counters / Communication’ status (should count < 5 errors per operational hour)

Causes/solutions:

Cause: The RS485 cabling is not implemented as specified.
Solution: Change cabling and/or grounding.

Cause: All devices on the bus do not have a unique slave address.
Solution: Set up unique addresses by use of BusLink.

Cause: The master is not fast enough to pick up fast slave responses.
Solution: Increase MODBUS response delay by use of BusLink.

Cause: The MODBUS is placed in an extremely hazardous industrial environment.
Solution: Improve cabling, grounding or lower communication speed.

Cause: The actuator is just powered up and not ready to answer the request.
Solution: Wait 5 seconds after a power up before starting MODBUS comunication.

Problem: The master receives a ‘Slave Device Failure’ from the actuator

Causes/solutions:

Cause: The parameter database of the actuator is disrupted.
Solution: Return the actuator for repair together with read IR 16 value.

Problem: The master receives an exception response from the actuator

Diagnosis: Register the exception code[3] received from the actuator to identify the cause.

Causes/solutions:

Cause: The master ask for a non-implemented function/address or an invalid value
Solution: Correct the PLC/PC program.

[3] Exception codes are explained in details in Exceptions page 20

Problem: The actuator does not move after a run-command

Diagnosis:

Check with BusLink: Current Cut-off limits (Inwards/Outwards) and Max. Amps measured.

Check with BusLink: Position is valid, i.e. <> 32767.

Check with MODBUS master: Read Input register 6 (Actual_VIN) to verify that power supply is within specified values.

Check with MODBUS master: Read Input register 4 (Status Register 1) to verify that the preconditions to run are fulfilled (Ready & Position Valid).

Check with MODBUS master: Read Input register 5 (Reason for Last Stop) to identify the cause (ref. Appendix A).

Causes/solutions:

Cause: Actuator is already moving when it receives the command.[4]
Solution: Check that preconditions[5] are fulfilled before sending new run.

Cause: Power supply voltage is below min. or above max. limits.
Solution: Adjust supply voltage.

Cause: Current limit exceeded.
Solution: Adjust current limits by use of BusLink or decrease actuator load.

Cause: Actuator lost its position.
Solution: Reinitialise In/Out by use of BusLink.

Cause: Temperature limit exceeded.
Solution: Verify that specified operation conditions are met.

Cause: Voltage or power supply error.
Solution: Return the actuator for repair if you consistently recognise stop 9 ‘Power Switch error’, 10 ‘H-bridge error’ or 11 ‘High side voltage error’.

Cause: Motor stall due to wrong load/speed relationship.
Solution: Increase Max Speed setting by use of BusLink or MODBUS master.

Problem: BusLink PC tool can not communicate with the actuator

Diagnosis: Errors announced during startup or no connection to actuator.

Causes/solutions:

Cause: .ini file (C:\Program Files\Linak\BusLink\parametertable_0.xxx.ini) is incompatible with the actuator.

Solution: Update .ini file to match the current actuator parameter table version.

Cause: The USB2LIN communication cable is too old and incompatible with BusLink.[6]

Solution: Update USB2LIN cable.

[4] If you really want to interrupt and start a new positioning, then execute a stop-command first

[5] See Run to target page 21

[6] See BusLink Configuration Software page 26 for details

Problem: Actuator firmware is corrupted e.g. due to upgrade problem

Diagnosis: After firmware upgrade any digital communication with actuator is impossible.

Causes/solutions:

Cause: Firmware is corrupted.

Solution: To activate the actuator bootloader and download new firmware using BusLink. This is done by strictly following the following procedure:

- Disconnect actuator communication cable and power cable
- Start BusLink without any cables connected to the actuator
- Go to ‘Firmware’ tab
- Connect ONLY actuator communication cable (USB2LIN)
- Download new firmware

DECLARATION OF INCORPORATION OF PARTLY COMPLETED MACHINERY

LINAK A/S
Smedevænget 8
DK - 6430 Nordborg

Herewith declares that LINAK TECHLINE ® products
as characterized by the following models and types:

Linear Actuators	LA12, LA14, LA22, LA23, LA25, LA30, LA35, LA36, LA37

comply with the following parts of the Machinery Directive 2006/42/EC, ANNEX I, *Essential health and safety requirements relating to the design and construction of machinery*:

1.5.1 Electricity supply

The relevant technical documentation is compiled in accordance with part B of Annex VII and that this documentation or part hereof will be transmitted by post or electronically to a reasoned request by the national authorities.

This partly completed machinery must not be put into service until the final machinery into which it is to be incorporated has been declared in conformity with the provisions of the Machinery Directive 2006/42/EC where appropriate.

Nordborg, 2014-10-20

John Kling

LINAK A/S
John Kling, B.Sc.E.E.
Certification and Regulatory Affairs
Authorized to compile the relevant technical documentation

Original Declaration

Appendix A - Register Map

All accessible Input- and Holding registers are listed below. Notice that the unit of the parameter values is found in column Remark'.

Read Only Registers (Input Registers)			**Parameter list version**	**1.1**			
Address	**Term**	**Para-meter**	**Remark**	**Type**	**Min. limit**	**Max. limit**	**High/ Low Order**
1	Run Status	4	0 = Idle 1 = Running outwards 2 = Running inwards 3 = Waiting for time = "Next Time To Run" 4 = Initialising outwards 5 = Initialising inwards	U8	0	5	
2	Actual Position	8	Resolution in 1/10mm relative to Offset	S16	-1000	32000	
3	Distance From Target	14	Resolution in 1/10mm	S16	-	-	
4	Status Register 1	51	See definition of bits below Bit 0: EOS switch, Inwards (Switch activated => 1) Bit 1: EOS switch, Outwards (Switch activated => 1) Bit 2: Hall A Signal Bit 3: Hall B Signal Bit 4: Ready (True when actuator is ready to run) Bit 5: FET Temperature Valid Bit 6: Reserved Bit 7: Current Time Valid Bit 8: Reserved Bit 9: LOCAL Connected (True when in service mode) Bit 10: Position valid Bit 11 -> 15: Reserved	U16	-	-	
5	Reason for Last Stop	50	0 = Target position reached 1 = Current cut-off out 2 = Current cut-off in 3 = Stop command received 4 = Undervoltage detected on V_IN 5 = Overvoltage detected on V_IN 6 = FET Temperature exceeded 7 = Actuator Internal Temperature exceeded 8 = Hall error 9 = Power Switch error 10 = H-brige error 11 = High side voltage error 12 = Not configured/initialised 13 = Endstop reached 14 = Motor Stall (Max Speed too low) 15 = System Stop	U8	0	15	
6	Actual V_IN	23	Average mV value, updated every 100ms	U16	0	65535	
7	Actual Current	28	Average mA value, updated every 100ms	U16	0	30000	
8	Actuator Internal Temperature	33	Pheudo ambient temperature [°C]	S16	-40	150	
9	FET Temperature	35	FET Temperature [°C] If set value is out of range, actuator will deny to run	S16	-40	225	
10	Offset	9	Number of pulses relative to EOS switch inwards. Can only be set via the BusLink service tool	U16	0	65000	
11	Max Stroke Length	11	Resolution in 1/10mm relative to Offset. Used to limit stroke length (i.e. Stroke length shorter than physical maximum). Can only be set via the BusLink service tool	U16	0	65000	

Read Only Registers (Input Registers)			**Parameter list version**	**1.1**			
Address	**Term**	**Para-meter**	**Remark**	**Type**	**Min. limit**	**Max. limit**	**High/ Low Order**
12	EOS Out Position	10	Pulses relative to EOS In. Set during production	U16	0	65000	
13	Actual Position (Pulses)	7	Pulses relative to EOS In switch	S16	-1000	32000	
14	-	24	For development/production purposes	U16	0	65535	
15	-	25	For development/production purposes	U16	0	65535	
16	-	52	For development/production purposes	U16	-	-	
17	.	45	For development/production purposes	U16	1	65000	
18	-	34	For development/production purposes	S16	40	225	
19	-	36	For development/production purposes	S16	40	225	
20	-	37	For development/production purposes	S16	40	225	
21	-	38	For development/production purposes	S16	-100	300	
22	-	19	For development/production purposes	U16	10000	60000	
23	-	20	For development/production purposes	U16	10000	40000	
24	-	21	For development/production purposes	U16	10000	60000	
25	-	22	For development/production purposes	U16	10000	40000	
26	Number of Power Fails	63	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(31:16)
27	Number of Power Fails	63	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(15:0)
28	Total Number of Starts Inwards	54	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(31:16)
29	Total Number of Starts Inwards	54	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(15:0)
30	Total Number of Starts Outwards	55	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(31:16)
31	Total Number of Starts Outwards	55	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(15:0)
32	Total Running Time	56	Service counter (value in seconds). Also available via the BusLink service tool	U32	0	4294967295	(31:16)
33	Total Running Time	56	Service counter (value in seconds). Also available via the BusLink service tool	U32	0	4294967295	(15:0)
34	Maximum Current Measured	57	Also available via the BusLink service tool (value in mA)	U16	0	65535	
35	Maximum FET Temperature Measured	58	Also available via the BusLink service tool (value in °C)	U16	0	65535	
36	Minimum Actuator Temperature Measured	59	Also available via the BusLink service tool (value in °C)	S16	-40	150	
37	Performed Work	60	Also available via the BusLink service tool (value is current multiplied by time [A *s])	U32	0	4294967295	(31:16)
38	Performed Work	60	Also available via the BusLink service tool (value is current multiplied by time [A *s])	U32	0	4294967295	(15:0)
39	Number of Telegrams With Invalid Content	61	Service counter (invalid function code, invalid data etc.) used for debugging on system level. Also available via the BusLink service tool	U32	0	4294967295	(31:16)

Read Only Registers (Input Registers)			Parameter list version	1.1			
Address	**Term**	**Para-meter**	**Remark**	**Type**	**Min. limit**	**Max. limit**	**High/ Low Order**
40	Number of Telegrams With Invalid Content	61	Service counter (invalid function code, invalid data etc.) used for debugging on system level. Also available via the BusLink service tool	U32	0	4294967295	(15:0)
41	Number of Telegrams With CRC Error	62	Service counter (total of corrupted telegrams) used for debugging on system level. Also available via the BusLink service tool	U32	0	4294967295	(31:16)
42	Number of Telegrams With CRC Error	62	Service counter (total of corrupted telegrams) used for debugging on system level. Also available via the BusLink service tool	U32	0	4294967295	(15:0)
43	Total Number of Current Cut-Offs out	78	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(31:16)
44	Total Number of Current Cut-Offs out	78	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(15:0)
45	Total Number of Current Cut-Offs in	79	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(31:16)
46	Total Number of Current Cut-Offs in	79	Service counter. Also available via the BusLink service tool	U32	0	4294967295	(15:0)
47	Number of Times Running With Actuator Temperature Exceeded	80	Service counter. Also available via the BusLink service tool	U16	0	65535	
48	-	94	For development/production purposes	U8	0	225	
49	-	95	For development/production purposes	U16	0	65535	
50	-	74	For development/production purposes	U16	0	65535	
51	-	75	For development/production purposes	U16	0	65535	
52	-	1	For development/production purposes	U16	0	65535	
53	Remote Data Bus Baud Rate	6	Baud rate in 100's (e.g. 1152 => 115200bps) Can only be set via the BusLink service tool	U16	0	1152	
54	Parity	81	Parity setting for Remote interface: 0 = No parity 1 = Even parity 2 = Odd parity Can only be set via the BusLink service tool	U8	0	2	
55	Number of Stop Bits	82	Stop bit setting for Remote interface: 1 = 1 stop bit 2 = 2 stop bits Can only be set via the BusLink service tool	U8	1	2	
56	-	98	For development/production purposes	U16	0	65535	
57	-	47	For development/production purposes	U8	0	250	
58	-	48	For development/production purposes	U8	0	250	
59	Modbus Response Delay	99	Delay in ms from reception of last character in request frame before response is sent Default: 19200-1*11bit*3,5char = 2,005ms => 3ms Can only be set via the BusLink service tool	U8	0	100	
60	Command, Local	3	Can only be set via the BusLink service tool	U8	0	137	
61	-	64	For development/production purposes	U32	0	4294967295	(31:16)
62	-	64	For development/production purposes	U32	0	4294967295	(15:0)
63	-	65	For development/production purposes	U16	0	65535	
64	-	66	For development/production purposes	U16	0	65535	

Read Only Registers (Input Registers)			Parameter list version	1.1			
Address	**Term**	**Para-meter**	**Remark**	**Type**	**Min. limit**	**Max. limit**	**High/ Low Order**
65	-	67	For development/production purposes	U16	0	65535	
66	-	68	For development/production purposes	U16	0	65535	
67	-	69	For development/production purposes	U16	0	65535	
68	-	70	For development/production purposes	U16	0	65535	
69	-	71	For development/production purposes	U16	0	65535	
70	-	72	For development/production purposes	U16	0	4095	
71	-	73	For development/production purposes	U8	0	255	
72	-	96	For development/production purposes	U16	20000	29999	
73	-	97	For development/production purposes	U8	0	15	
74	-	29	For development/production purposes	U16	100	4095	
75	-	30	For development/production purposes	U16	100	4095	
76	-	31	For development/production purposes	U16	0	50000	
77	-	32	For development/production purposes	U16	0	50000	
78	-	12	For development/production purposes	U8	0	200	
79	-	53	For development/production purposes	U8	0	3	
80	-	83	For development/production purposes	U16	1614	1923	
81	-	84	For development/production purposes	S8	0	40	
82	-	89	For development/production purposes	U8	0	255	
83	-	90	For development/production purposes	U8	0	255	
84	-	91	For development/production purposes	U8	0	255	
85	-	108	For development/production purposes	U8	0	255	
86	-	101	For development/production purposes	U32	0	4294967295	(31:16)
87	-	101	For development/production purposes	U32	0	4294967295	(15:0)
88	-	102	For development/production purposes	U32	0	4294967295	(31:16)

Internal registers (Holding registers)			**Parameter list version**	**1.1**			
Address	**Term**	**Para-meter**	**Remark**	**Type**	**Min. limit**	**Max. limit**	**High/ Low Order**
1	Target Position	13	Resolution in 1/10mm relative to Offset	U16	0	65000	
2	Command, Remote	2	0 = No Command 1 = Run to target position 2 = Stop 3 = Run at "Next time to run" 4 = Initialise outwards 5 = Initialise inwards 6 = Run to Reference position 1 7 = Run to Reference position 2 8 = Run to Reference position 3 9 = Run to Reference position 4 10 = Production test passed	U8	0	10	
3	Max Speed	46	% PWM	U8	10	100	
4	Next Time to Run (s)	43	Value between in the range from 0 to 59	U8	0	59	
5	Next Time to Run (m)	44	Value between in the range from 0 to 59	U8	0	59	
6	Reference position 1	15	Resolution in 1/10mm relative to Offset	U16	0	65000	
7	Reference position 2	16	Resolution in 1/10mm relative to Offset	U16	0	65000	
8	Reference position 3	17	Resolution in 1/10mm relative to Offset	U16	0	65000	
9	Reference position 4	18	Resolution in 1/10mm relative to Offset	U16	0	65000	
10	Current Time (s)	39	Value between in the range from 0 to 59. Written to by broadcast	U8	0	59	
11	Current Time (m)	40	Value between in the range from 0 to 59. Written to by broadcast	U8	0	59	
12	Current Time (h)	41	Value between in the range from 0 to 23. Written to by broadcast	U8	0	23	
13	Current Time (date)	42	Date counted as days since 1/1-2000 Written to by broadcast	U16	0	65000	
14	Reason for Power Down	49	0 = Power Fail, V_IN too low 1 = Power Fail, V_IN too high 2 = No Power Fail It is the Remote Master who sets to 2 after a power fail - the system itself doesn't do this!	U8	0	2	
15	Current Cut-Off Limit Inwards	26	Value in mA	U16	2000	30000	
16	Current Cut-Off Limit Outwards	27	Value in mA	U16	2000	30000	
17	Ramp Down Before Target	92	Resolution in 1/10mm	U8	10	255	
18	-	76	For development/production purposes	U32	0	4294967295	(31:16)
19	-	76	For development/production purposes	U32	0	4294967295	(15:0)
20	-	77	For development/production purposes	U16	0	65535	
21	Remote Data Bus Slave Address	5	Address 0 is broadcast address Address 247 is 'Unassigned'	U8	1	247	
22	-	93	For development/production purposes	U8	0	255	

Internal registers (Holding registers)			**Parameter list version**	**1.1**			
Address	**Term**	**Parameter**	**Remark**	**Type**	**Min. limit**	**Max. limit**	**High/ Low Order**
1001	Modbus User Map Input Register Address	85	Hardcoded to Modbus Holding register 1001 Used to set the target cell for the Parameter Address set by Modbus User Map Parameter Reference (I). When written to, Modbus User Map Parameter Reference (I) is updated with the current value from the Modbus Input Register part of the Modbus User Map table (used for reading out the current Modbus User Map)	U16	1	150	
1002	Modbus User Map Parameter Reference (I)	86	Hardcoded to Modbus Holding register 1002 When this parameter is written to, the content is passed to the ‘Parameter Address’ cell in the Modbus User Map table specified by Modbus User Map Input Register Reference	U8	0	255	
1003	Modbus User Map Holding Register Address	87	Hardcoded to Modbus Holding register 1003 Used to set the target cell for the Parameter Address set by Modbus User Map Parameter Reference (H). When written to, Modbus User Map Parameter Reference (H) is updated with the current value from the Modbus Holding Register part of the Modbus User Map table (used for reading out the current Modbus User Map)	U16	1	50	
1004	Modbus User Map Parameter Reference (H)	88	Hardcoded to Modbus Holding register 1004 When this parameter is written to, the content is passed to the ‘Parameter Address’ cell in the Modbus User Map table specified by Modbus User Map Holding Register Reference	U8	0	255	
1005	Modbus User Map Command	100	Hardcoded to Modbus Holding register 1005 0 = No command 1 = Save Modbus User Map 2 = Set Modbus User Map to Default	U8	0	2	

FACTORIES

CHINA
LINAK (Shenzhen) Actuator Systems, Ltd.
Phone: +86 755 8610 6656
Fax: +86 755 8610 6990
E-mail: sales@linak.cn
www.linak.cn

DENMARK
LINAK A/S - Group Headquarters, Guderup
Phone: +45 73 15 15 15
Fax: +45 74 45 80 48
Fax: +45 73 15 16 13 (Sales)
E-mail: info@linak.com
www.linak.com

SLOVAKIA
LINAK Slovakia s.r.o.
Phone: +421 51 75 63 414
Fax: +421 51 75 63 410
E-mail: jp@linak.sk
www.linak.com

USA
LINAK U.S. Inc.
North and South American Headquarters
Phone: +1 502 253 5595
Fax: +1 502 253 5596
E-mail: info@linak-us.com
www.linak-us.com

SUBSIDIARIES

AUSTRALIA
LINAK Australia Pty. Ltd
Phone: +61 3 8796 9777
Fax: +61 3 8796 9778
E-mail: sales@linak.com.au
www.linak.com.au

AUSTRIA
LINAK Repräsentanz Österreich (Wien)
Phone: +43 (1) 890 7446
Fax: +43 (1) 890 744615
E-mail: info@linak.de
www.linak.at

BELGIUM & LUXEMBOURG
LINAK Actuator-Systems NV/SA
Phone: +32 (0)9 230 01 09
Fax: +32 (0)9 230 88 80
E-mail: beinfo@linak.be
www.linak.be

BRAZIL
LINAK Do Brasil Comércio De Atuadores Ltda.
Phone: +55 (11) 2832 – 7070
Fax: +55 (11) 2832 – 7060
E-mail: info@linak.com.br
www.linak.com.br

CANADA
LINAK Canada Inc.
Phone: +1 502 253 5595
Fax: +1 416-255-7720
E-mail: info@linak.ca
www.linak-us.com

CZECH REPUBLIC
LINAK C&S S.R.O.
Phone: +420581741814
Fax: +420581702452
E-mail: ponizil@linak.cz
www.linak.cz

DENMARK
LINAK Danmark A/S
Phone: +45 86 80 36 11
Fax: +45 86 82 90 51
E-mail: linak@linak-silkeborg.dk
www.linak.dk

FINLAND
LINAK OY
Phone: +358 10 841 8700
Fax: +358 10 841 8729
E-mail: linak@linak.fi
www.linak.fi

FRANCE
LINAK France E.U.R.L
Phone: +33 (0) 2 41 36 34 34
Fax: +33 (0) 2 41 36 35 00
E-mail: linak@linak.fr
www.linak.fr

GERMANY
LINAK GmbH
Phone: +49 6043 9655 0
Fax: +49 6043 9655 60
E-mail: info@linak.de
www.linak.de

INDIA
LINAK A/S India Liaison Office
Phone: +91 120 4734613
Fax: +91 120 4273708
E-mail: info@linak.in
www.linak.in

IRELAND
LINAK UK Limited - Ireland
Phone: +44 (0)121 544 2211
Fax: +44 (0)121 544 2552
+44 (0)796 855 1606 (UK Mobile)
+35 387 634 6554 (Republic Of Ireland Mobile)
E-mail: sales@linak.co.uk
www.linak.co.uk

ITALY
LINAK Italia S.r.l.
Phone: +39 02 48 46 33 66
Fax: +39 02 48 46 82 52
E-mail: info@linak.it
www.linak.it

JAPAN
LINAK K.K.
Phone: 81-45-533-0802
Fax: 81-45-533-0803
E-mail: linak@linak.jp
www.linak.jp

MALAYSIA
LINAK Actuators Sdn. Bhd.
Phone: +60 4 210 6500
Fax: +60 4 226 8901
E-mail: info@linak-asia.com
www.linak.my

NETHERLANDS
LINAK Actuator-Systems B.V.
Phone: +31 76 5 42 44 40
Fax: +31 76 5 42 61 10
E-mail: info@linak.nl
www.linak.nl

NEW ZEALAND
LINAK New Zealand Ltd.
Phone: +64 9580 2071
Fax: +64 9580 2072
E-mail: nzsales@linak.com.au
www.linak.co.nz

NORWAY
LINAK Norge AS
Phone: +47 32 82 90 90
Fax: +47 32 82 90 98
E-mail: info@linak.no
www.linak.no

POLAND
LINAK Polska
Phone: +48 (22) 500 28 74
Fax: +48 (22) 500 28 75
E-mail: dkreh@linak.dk
www.linak.pl

REPUBLIC OF KOREA
LINAK Korea Ltd.
Phone: +82-(0)2-6231-1515
Fax: +82-(0)2-6231-1516
E-mail: scully@linak.kr
www.linak.kr

RUSSIAN FEDERATION
000 LINAK
Phone: +7 495 280 14 26
Fax: +7 495 687 14 26
E-mail: info@linak.ru
www.linak.ru

SPAIN
LINAK Actuadores, S.L.u
Phone: +34 93 588 27 77
Fax: +34 93 588 27 85
E-mail: esma@linak.es
www.linak.es

SWEDEN
LINAK Scandinavia AB
Phone: +46 8 732 20 00
Fax: +46 8 732 20 50
E-mail: info@linak.se
www.linak.se

SWITZERLAND
LINAK AG
Phone: +41 43 388 31 88
Fax: +41 43 388 31 87
E-mail: info@linak.ch
www.linak.ch

TAIWAN
LINAK A/S Taiwan Representative Office
Phone: +886 2 27290068
Fax: +886 2 27290096
Mobile: +886 989292100
E-mail: michael.chen@linak.com.tw
www.linak.com.tw

TURKEY
LINAK İth. İhr. San. ve Tic. A.Ş.
Phone: + 90 312 4726338
Fax: + 90 312 4726635
E-mail: info@linak.com.tr
www.linak.com.tr

UNITED KINGDOM
LINAK UK Limited
Phone: +44 (0)121 544 2211
Fax: +44 (0)121 544 2552
E-mail: sales@linak.co.uk
www.linak.co.uk

DISTRIBUTORS

ARGENTINA
Novotec Argentina SRL
Phone: 011-4303-8989/8900
Fax: 011-4032-0184
E-mail: info@novotecargentina.com
www.novotecargentina.com

AUSTRALIA
Ballarat Industrial Supplies
www.ballind.com.au

BL Shipways & Co
www.blshipway.com.au

Gas Strut Marine and Industrial
www.gasstrutmarine.com.au

Prime Motion & Control
www.primehyd.com.au

West Vic Industrial Supplies
www.westvicindustrial.com.au

COLOMBIA
MEM Ltda
Phone: +[57] (1) 334-7666
Fax: +[57] (1) 282-1684
E-mail: servicioalcliente@memltda.com.co
www.memltda.com.co

INDIA
Mechatronics Control Equipments India Pvt Ltd
Phone: +91-44-28558484/85,
E-mail: bala@mechatronicscontrol.com
www.mechatronicscontrol.com

INDONESIA
Pt. Himalaya Everest Jaya
Phone: +6 221 544 8956/8965,
Fax: +6 221 619 4658/1925
E-mail: hejplastic-div@centrin.net.id
www.hej.co.id

IRAN
Bod Inc.
Phone: +98 2188998635-6
Fax: +98 2188954481
E-mail: info@bod.ir
www.bod.ir

RUSSIAN FEDERATION
000 FAM
Phone: +7 812 3319333
Fax: +7 812 3271454
E-mail: purchase@fam-drive.ru
www.fam-drive.ru

SINGAPORE
Servo Dynamics Pte. Ltd.
Phone: +65 6844 0288
Fax: +65 6844 0070
E-mail: servodynamics@servo.com.sg
www.servo.com.sg

SOUTH AFRICA
Industrial Specialised Applications CC
Phone: +27 11 312 2292 or
+27 11 2077600 (Switch Board)
Fax: +27 11 315 6999
E-mail: gartht@isagroup.co.za
www.isaza.co.za

UNITED ARAB EMIRATES
Mechatronics
Phone: +971 4 267 4311
Fax: +971 4 267 4312
E-mail: mechtron@emirates.net.ae
www.mechatronics.ae

For contact details on other countries
please visit *www.linak.com* or contact:

LINAK INTERNATIONAL
Phone: +45 73 15 15 15
Fax: +45 74 45 90 10
Fax: +45 73 15 16 13 (Sales)
E-mail: info@linak.com
www.linak.com

Terms of use
The user is responsible for determining the suitability of LINAK products for specific application. LINAK takes great care in providing accurate and up-to-date information on its products.
However, due to continuous development in order to improve its products, LINAK products are subject to frequent modifications and changes without prior notice. Therefore, LINAK cannot guarantee the correct and actual status of said information on its products.
While LINAK uses its best efforts to fulfil orders, LINAK cannot, for the same reasons as mentioned above, guarantee the availability of any particular product. Therefore, LINAK reserves the right to discontinue the sale of any product displayed on its website or listed in its catalogues or other written material drawn up by LINAK.
All sales are subject to the Standard Terms of Sale and Delivery for LINAK. For a copy hereof, please contact LINAK.

LINAK®
WE IMPROVE YOUR LIFE