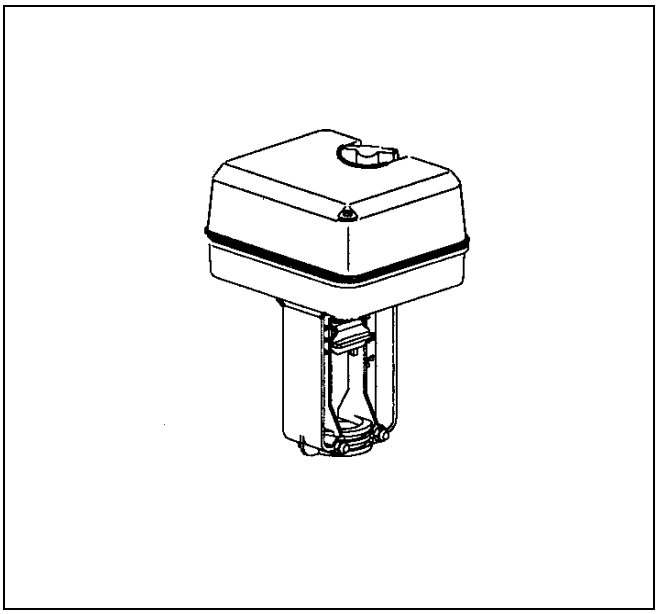
Honeywell

ML7420G

ELECTRIC LINEAR VALVE ACTUATORS WITH LON COMMUNICATION AND ADDITIONAL I/O FEATURES

SPECIFICATION DATA & USER GUIDE



GENERAL

The ML7420G actuators are application oriented large linear actuators and provide local operating network capabilities (LON) as well as additional digital inputs/outputs to integrate further objects. They operate Honeywell's standard valves in heating, ventilation and air conditioning (HVAC) applications.

The actuators adapt automatically the mechanical tolerances of the connected valves and offers features like sequential acting, stroke limitation and to operate floating devices from the LON based actuator.

FEATURES

- **Digital inputs for frost protection or filter supervision**
- **Digital outputs to operate floating devices**
- **Span / Offset adjustment**
- **Digital outputs to switch fan or pump**
- **Autoadaption of stroke in end positions**
- **Force limiting end switches**
- **Internal potentiometer feedback**
- **Open products for third party integration**
- **Direct / reverse action**
- **Stroke position on communication failure selectable**
- **Easy and quick installation**
- **No separate linkage required**
- **Low power consumption**
- **Simplifying complex HVAC plants**
- **Manual operation**
- **Corrosion resistant design**
- **Maintenance free**

Models

OS-No.	Description
ML7420G1004	Internal actuator
ML7420G1012	Internal actuator with 2 digital inputs
ML7420G1020	Internal actuator with 2 digital outputs

SPECIFICATIONS

Temperature Limits

Ambient operating limits	-10 to +50°C @ 5 to 95%rh
Ambient storage limits	-40 to +70°C @ 5 to 95%rh
Medium valve temperature	Max. 150°C

Communication (Actuator Position)

nviActPos[SNVT_lev_percent]	0...100%
Control	LON (FTT10A)
Hardware	3150 32K FLASH

Safety

Protection class	II in acc. with EN60730-1
Protection standard	IP54 in acc. with EN60529
Flame retardant housing	V0 in acc. with UL94 (with metal cable gland)

LONMARK®

CE

Wiring		**Weight**	1.3kg
Wiring terminals	1.5mm^2	**Material**	
Cable entry	2xPG13.5. One additional knock out PG11 for digital inputs/outputs or daisy chain options	Cover	ABS-FR
		Base	Glass fibre reinforced plastic
		Yoke	Aluminium diecast

Supply Voltage	24Vac +/-15% 50/60Hz
Power Consumption	5VA/6VA (50Hz/60Hz)
SNVT_lev_percent 0%	Actuator stem retracted. Two way valve:"open", three way valve port A-AB:"closed" *
SNVT_lev_percent 100%	Actuator stem extended. Two way valve:"closed", three way valve port A-AB:"open" *
Stroke	20mm
Run Time @ 50Hz/60Hz	60s/50s
Close-Off Force	≥ 600N
2 Digital Inputs	Potential free contacts
2 Digital Outputs	24Vac/17VA

* = Factory setting

OPERATION

General

The drive of a synchronous motor is converted into linear motion of the actuator stem by using a spur gear transmission. The actuator stem is connected with the valve stem by a button keyed retainer connection.

An integrated spring package limits the stem force to a factory adjusted value in either direction.
Installed microswitches switch off the actuator precisely when the specified stem force is reached.

Manual Operation

Actuators are equipped with a manual operator used in case of power failure. Manual operation is only possible after the power supply is switched off or disconnected.

To operate push the manual operator knob down and turn clockwise to move the stem downward and counter-clockwise to move the stem upward. Upon power restore, the actuator returns to automatic control, the manual operator knob unlocks automatically.

Electrical Installation

The actuators are delivered with two pre-installed cable glands PG13.5 and one separate included knock out for PG11.
Cable length/diameter for field mounting:

- Max. 200m/1.5mm^2

NOTE: To avoid malfunction it is necessary to connect 24Vac power and ground (see Fig. 2).

Communication Failure

If the communication between the actuator and the binded controller fails (watch dog) the actuator drives to the pre-configured position (see page 20, SCPTdefOutput).

Adaption of stroke

If the nviActPos[SNVT_lev_percent] is 0 or 100%, the actuator adapts the stroke and positions, see chapter 3.3.2.1.

Adjustable SPAN/Offset

This function defines the relation between nviActPos[SNVT_lev_percent] and the stroke travel (e. g. sequence control), see chapter 3.2.2 for internal actuator and 3.3.2 for external floating device.

Accessories

The actuators can be equipped on site with an auxiliary switch unit with two switches. Their switching points are adjustable over the full length of the actuator stroke. The switches can be used to switch pumps or provide remote indication of any stroke position.

Part number: 43191680-005

High Temperature Kit

The High Temperature Kits are necessary for applications between 150 and 220°C medium valve temperature.

Order Number High Temperature Kit	Valve	DN
43196000-001	V5011A/V5011K V5013A/V5013G V5011R/V5013R V5328A/V5329A	15 - 40 15 - 40 15 - 50 15 - 32
43196000-002	V5011A V5013A/V5013G V5328A/V5329A V5049A V5050A	50 50 40 - 80 15 - 65 15 - 80

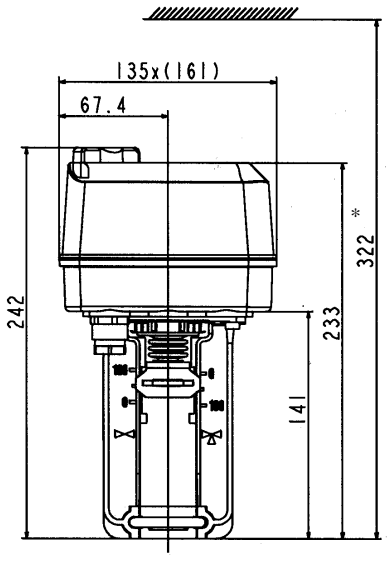
CLOSE-OFF PRESSURE RATINGS

	Valve Size							
	DN15 1/2"	DN20 3/4"	DN25 1"	DN32 1 1/4"	DN40 1 1/2"	DN50 2"	DN65 2 1/2"	DN80 3"
Valve	**Close-Off Pressure Ratings in kPa**							
V5011R,S	1600	1600	1000	700	460	260	--	--
V5013R	1600	1600	1000	700	460	260	--	--
V5328A	1600/1000	1000	1000	600	350	200	120	50
V5329A (PN16)	1000	1000	1000	790	480	260	160	100
V5329C (PN6)	600	600	600	600	480	260	160	100
V5049A	1600/1000	1000	1000	600	350	200	120	-
V5050A	1000	1000	1000	600	350	200	120	50
V5095A	--	1600	1600	1600	1600	1600	1600	1600
V5025B	1600	1600	1600	1600	1600	1600	1600	--
V176A,B	1500	1500	1500	--	1500	1500	1500	1500

For specific data of above listed valves see Specification Data No.:

V5011R,S EN0B-064
V5013R EN0B-065
V5328A EN0C-0432
V5329A,C EN0C-0434
V5049A EN0C-0433
V5050A EN0C-0435
V5095A EN0C-0490
V5025B EN0B-1017
V176A,B EN0B-0262

DIMENSIONS



*with High Temperature Kit 402mm

Fig. 1 Dimensions in mm

WIRING

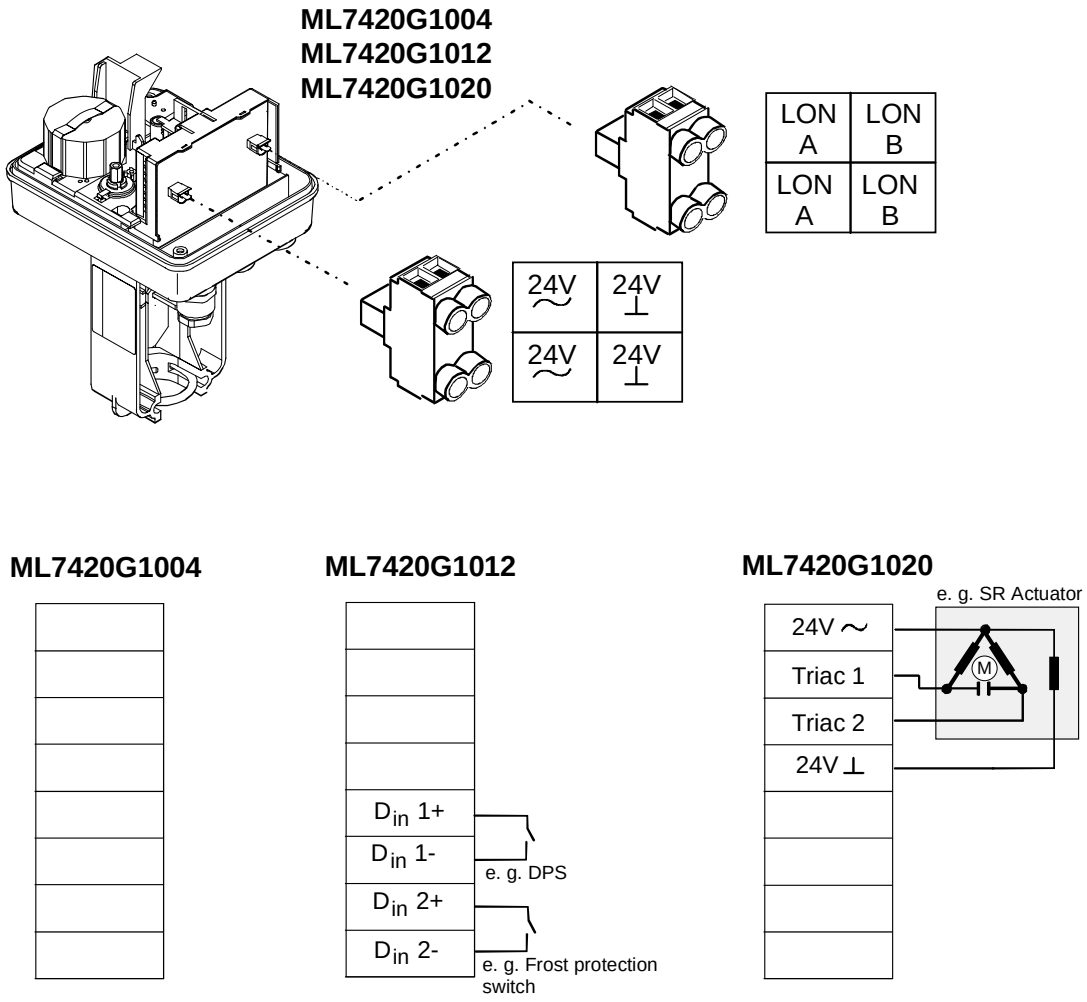


Fig. 2 Wiring

USER GUIDE

Contents

1 Definitions, Acronyms and Abbreviations

CREVAL	Cost Reduced Electric Valve Actuator Line
DI	Digital Input
DO	Digital Output
DPS	Digital Pump Switch
Echelon	Provider company of LonWorks
FTT10A	Free Topologie Transceiver
HW	Hardware
HVAC	Heating Ventilation Air Condition
I/O	Input/Output
IF	Interface
ILONA	Intelligent Lon Actuator
LON	Local Operating Network
Lon ID	Neuron ID, 48 Bit
LONMARK	Interoperability Association
LonWorks	Technology providing control networking
Node	LON-Device
NV	network variable
Object	Name of a complete Function inside of a object (e.g. a sensor-function)
OLAO1 internal	Open Loop Actuator Object #1 for internal Valve-Actuator
OLAO2 external	Open Loop Actuator Object #2 for external floating Valve-Actuator
OLSO1 external	Open Loop Sensor Object #1 for external connection of a digital sensor
OLSO2 external	Open Loop Sensor Object #2 for external connection of a digital sensor
SCPTs	Standard Configuration Parameter Types, defined by LONMARK
SLTA	Serial LonTalk Adapter
SNVTs	Standard network variable Types, defined by LONMARK
SR Actuator	Spring Return Actuator
SW	Software

2 INTERFACE FUNCTIONAL REQUIREMENTS

This chapter includes an overview and very special informations of the three ILONA-C actuators. A normal user only needs to read chapter 2.1 to know, which functionalities will be supported by the different actuators.

2.1 Main functionalities

Model	Node Object	Internal Actuator (OLAO1)	External Floating Actuator or 2 Digital Outputs (OLAO2)	Digital Input (OLSO1)	Digital Input (OLSO2)
ML7420G1004 (ILONA-C1)	•	•	-	-	-
ML7420G1012 (ILONA-C3)	•	•	-	•	•
ML7420G1020 (ILONA-C2)	•	•	•	-	-

2.2 Selfdocumentation strings

Model (Type)	String
ML7420G1004 (ILONA-C1)	"&3.2@;0,3;ILAC1_R100"
ML7420G1012 (ILONA-C3)	"&3.2@;0,3,1,2;ILAC3_R100"
ML7420G1020 (ILONA-C2)	"&3.2@;0,3,3;ILAC2_R100"

The selfdocumentation string includes very special informations, described by the following example "&3.2@;0,3;ILAC1_R100".
It means:

- &3.2: LonMark certification done on LonMark Guidelines 3.2
- 0: LonMark Object #0 = Node Object
- 3: LonMark Object #3 = Open Loop Actuator Object
- LAC1_R100: ILONA-C1 Softwareversion # R100

For first release. Later sd_strings changes to Revision R101..R102..R200, if needed.

2.3 Program IDs

The program ID's are needed by the Management-Tools to identify the LonMark certified interface (software).

Model (Type)	Program ID
ML7420G1004 (ILONA-C1)	80:00:0C:07:00:03:04:22
ML7420G1012 (ILONA-C3)	80:00:0C:07:00:03:04:24
ML7420G1020 (ILONA-C2)	80:00:0C:07:00:03:04:23

2.4 LON-Interface for the ML7420G1004

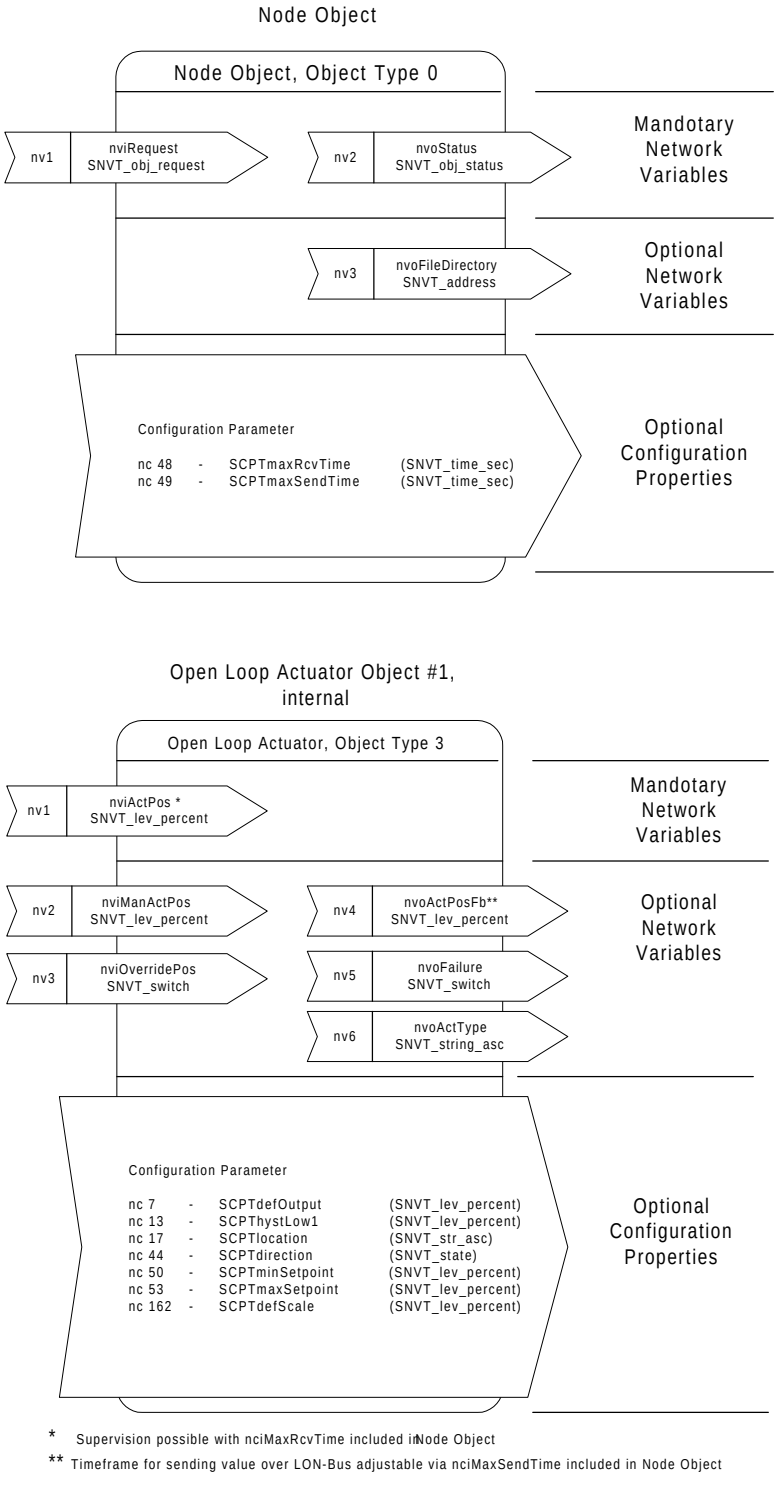


Fig. 3 LON-Interface for ML7420G1004

2.5 LON-Interface for the ML7420G1012

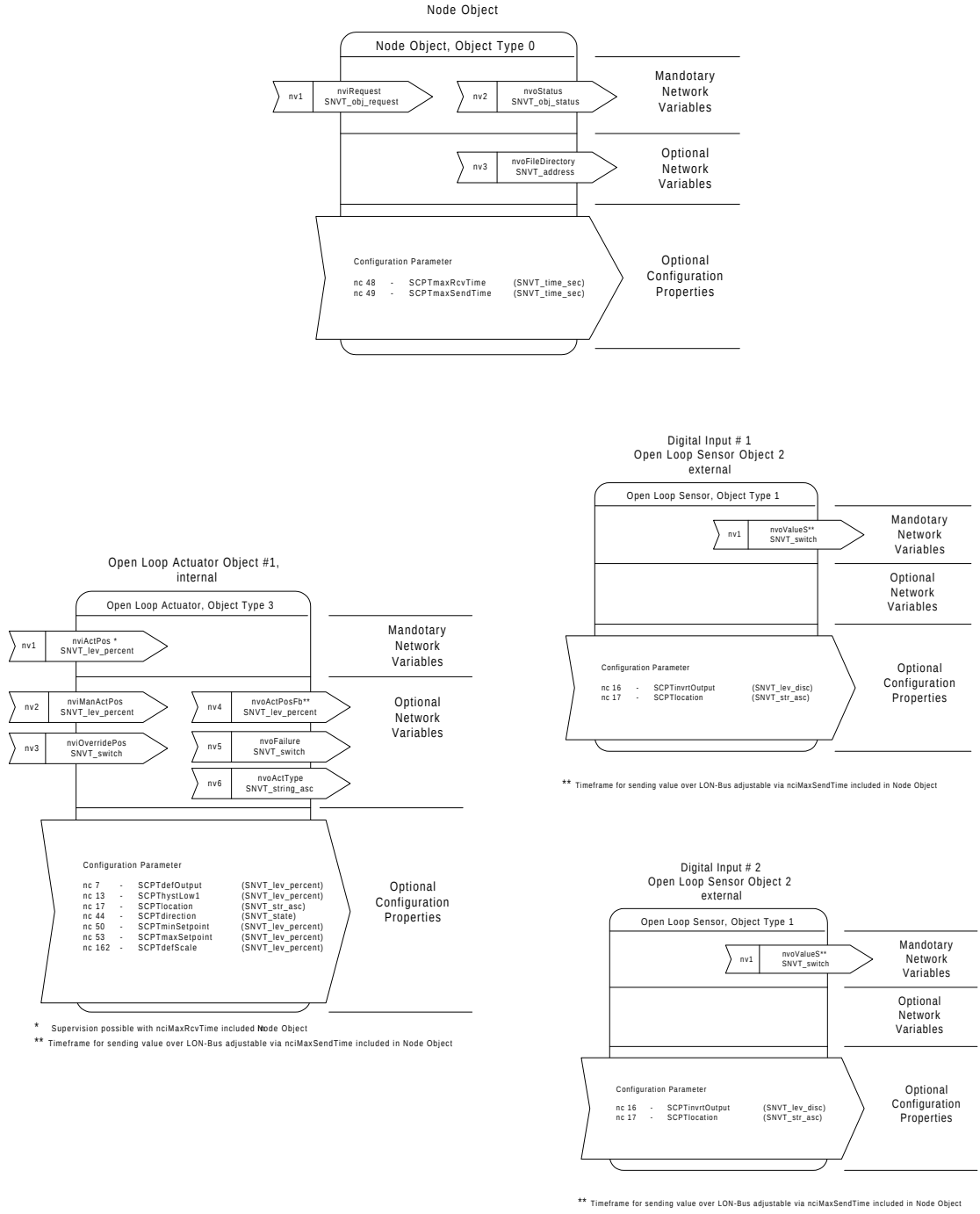


Fig. 4 LON-Interface for ML7420G1012

2.6 LON-Interface for the ML7420G1020

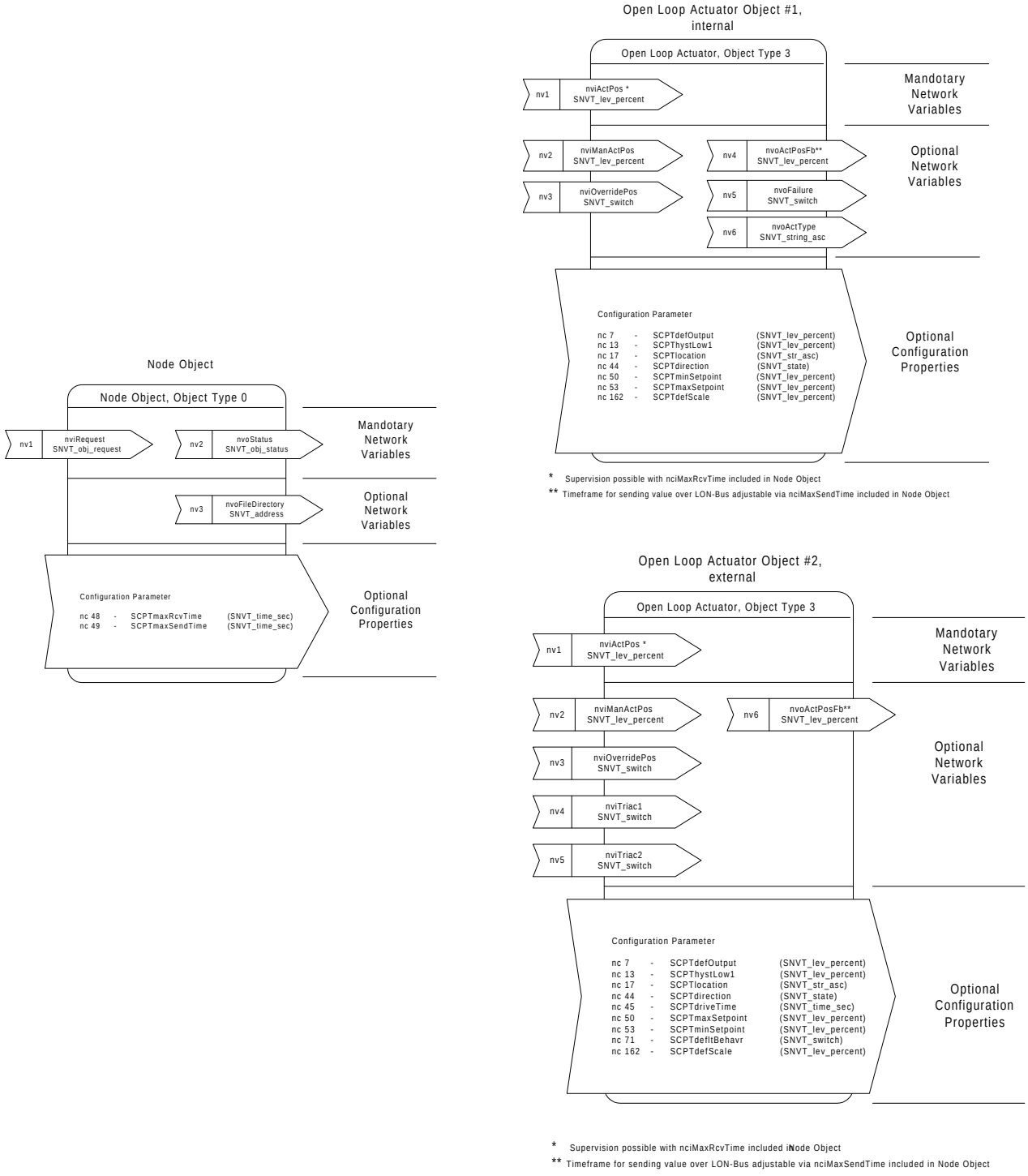


Fig. 5 LON-Interface for ML7420G1020

2.7 Example for a ML7420G1020 application for two valve sequence action with an internal actuator an external floating device.

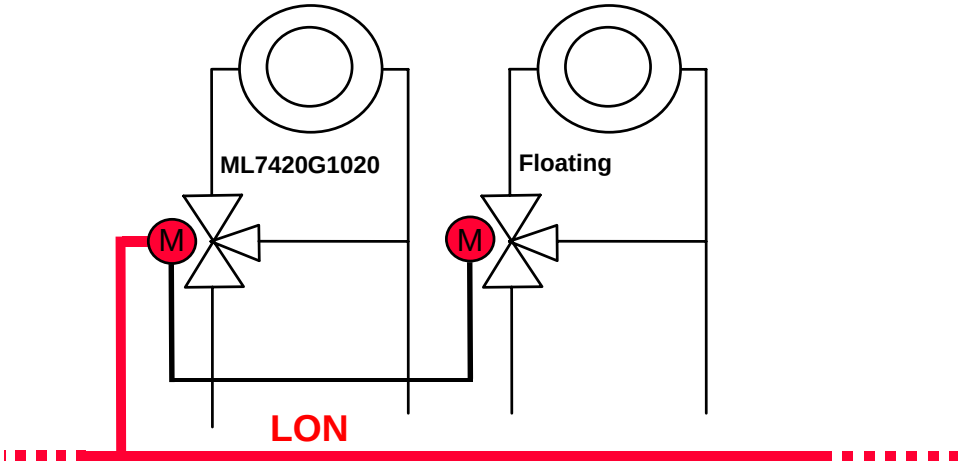


Fig. 6 Example for ML7420G1020 application

1) Make the installation of the devices in the field.

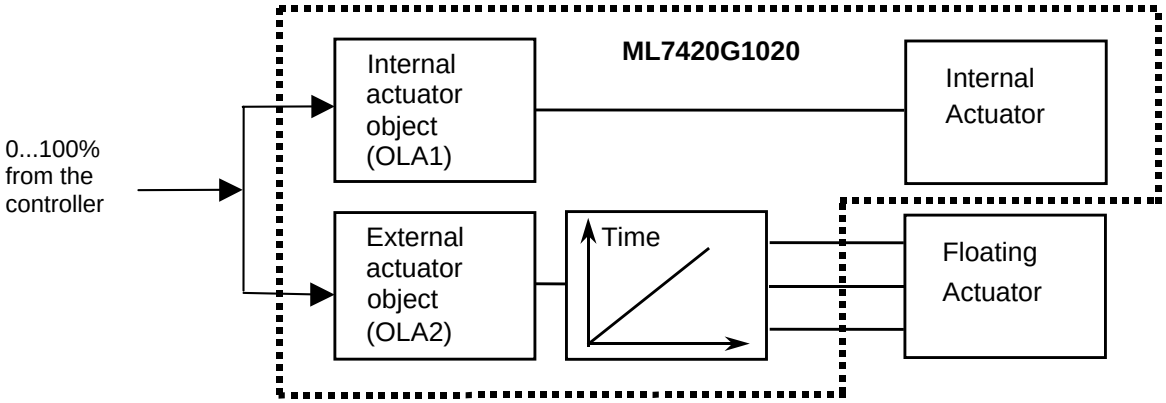


Fig. 7 Block diagram of ML7420G1020

2) After commissioning the ILONA-C2 device by a LON-Tool create the NV bindings.
The controller network variable for the setpoint has to be bound with
- nviActPos[0] of the internal actuator (OLA1) and
- nviActPos[1] of the external floating actuator (OLA2)

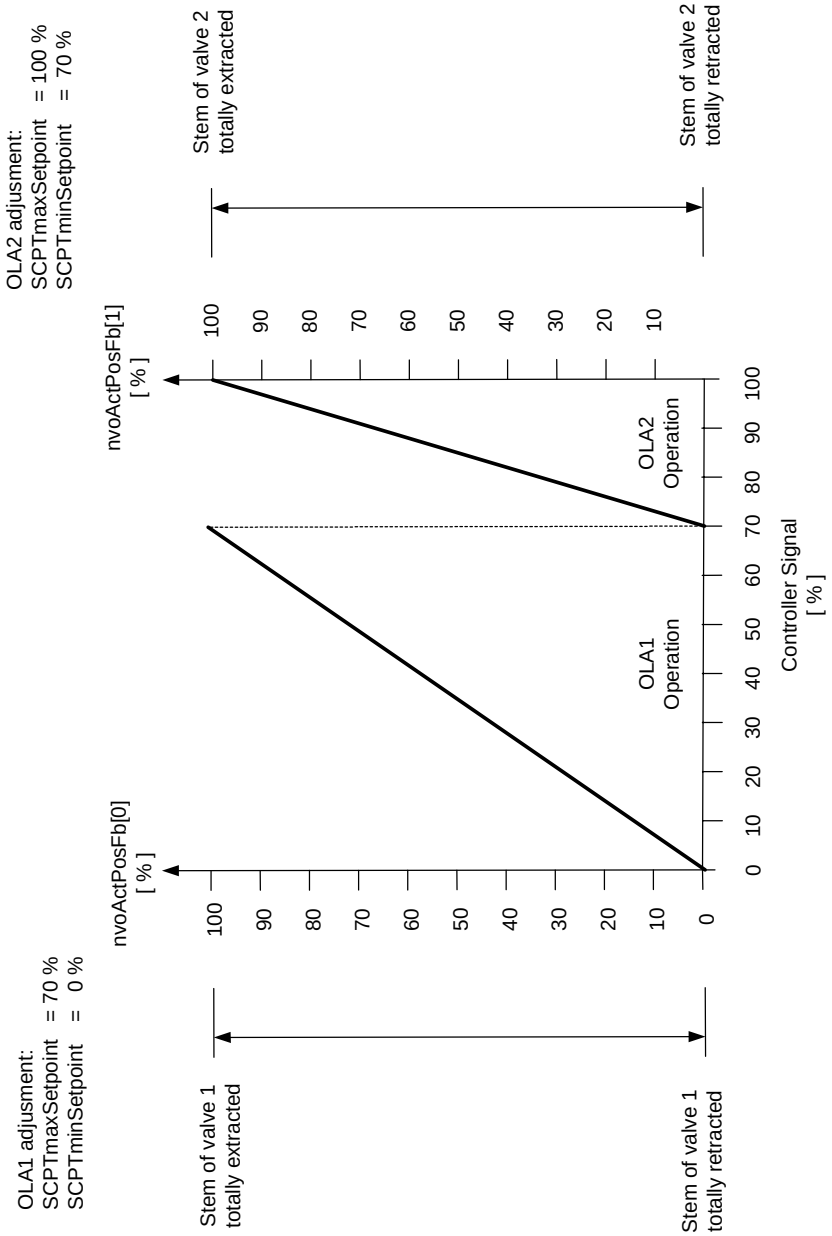


Fig. 8 OLA1 and OLA2 adjustments

3) Do the required adjustments for SCPTmaxSetpoint and SCPTminSetpoint of OLA1 and OLA2. There are a lot of different adjustments possible to get a two valve sequence action.

3 SOFTWARE INTERFACE (LON INTERFACE)

⚠ ATTENTION

For a correct view of the SCPTs by a LNS Network Management Tool, at least the "LonMark SNVTs & SCPTs-File" Version 10 has to be installed. You can download this from the LonMark-Webside: (http://www.lonmark.org/products/snvtfile.htm**)**

3.1 The "Node Object" (included in ML7420G1004/ML7420G1012/ML7420G1020)

The "Node Object" allows the function of objects within a node to be monitored. There are no applications included in the node object, but it is possible to analyze, to influence and to manage all objects within a node.

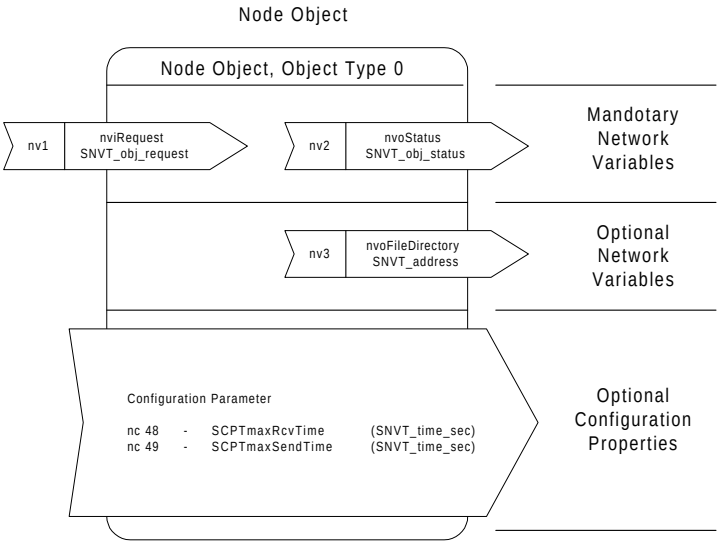


Fig. 9 The "Node Object"

Network Variable	**Type {Ranges / States} {Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
Mandatory Network Variables		Definition	Explanation
nviRequest			
	SNVT_obj_request {RQ_NORMAL, RQ_REPORT_MASK, RQ_UPDATE_STATUS, RQ_SELF_TEST} {RQ_NORMAL}	*RQ_NORMAL:* Same function as RQ_UPDATE_STATUS *RQ_REPORT_MASK:* The status bits that are supported by the object (you have to choose the right object ID) are send to the output network variable *nvoStatus*. All supported status bits will be set to a ONE. All Status bits which were not supported are zero. *RQ_UPDATE_STATUS:* The status (status bits) of the choosen object is sent to the output network variable *nvoStatus*. The state of the object is unchanged. The status bits of the node object (ID 0) (with the exception of *invalid_request* and *invalid_id*) are defined to be the inclusiv OR of the status bits of all the other objects in the node.	The input network variable *nviRequest* provides the mechanism to request a particular mode for a particular object within a node (device). The most important funcionallity for the installation of the device is the RQ_SELF_TEST request. If the RQ_SELF_TEST was executed an automatically adaption of the internal actuator will be done. This function can also be executed by pressing the service PIN for 5 seconds (see also chapter 3.2.3).

Network Variable	**Type** **{Ranges / States}** **{Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
Mandatory Network Variables		Definition	Explanation
nviRequest			
		RQ_SELFTEST: The internal Actuator makes an automatically adaption to the connected valve. This means that the Actuator runs from one limit switch to the other and then the adaption is done. During this operation the *manual_control* bit is set to ONE. If an error occurs the *out_of_limit* bit in the SNVT nvoStatus is set to ONE. After the operation is completed the *manual_control* bit is set to ZERO (see also chapter 3.2.3)	
nvoStatus			
	SNVT_obj_status supported status bits: {All objects *invalid_request*, *report_mask*, only node object *invalid_id*, only OLAO1 *out_of_limits,* *mechanical_fault,* *unable_to_measure* only OLAO1+OLAO2 *comm_failure*, *manual_control,*}	*invalid_id:* A Status of *invalid_id* is reported whenever a request is received for an object ID that is not defined in the node. If the object ID is invalid and *invalid_id* is reported, then no further checking of the other status bits are performed. *invalid_request:* Reporting a status of *invalid_request* is required if an unsupported request code is received on the object request input network variable *nviRequest*. *report_mask:* If the RQ_REPORT_MASK request-function is supported, the *report_mask* bit is set to ONE. *mechanical_fault:* • At least one limit switch is defect. • both limit switches were activated at the same time *out_of_limits:* If an synchronization (adaption) delivers new detected limit values which are far out of the norm values, the out_of_limit bit is set to ONE *out_of_limit* could be caused by a limit switch which is activated all the time (jams) too. *unable_to_measure:* The position feedback potentiometer value of the internal Actuator can not be measured because of an internal hardware defect. *comm_failure:* One or more received heartbeats are missing ⇒ [SCPTmaxRcvTime]	The output network variable *nvoStatus* reports the status for any object on a node. All status bits will be set to zero if the RQ_CLEAR_STATUS request is received or the reset event is occurred.

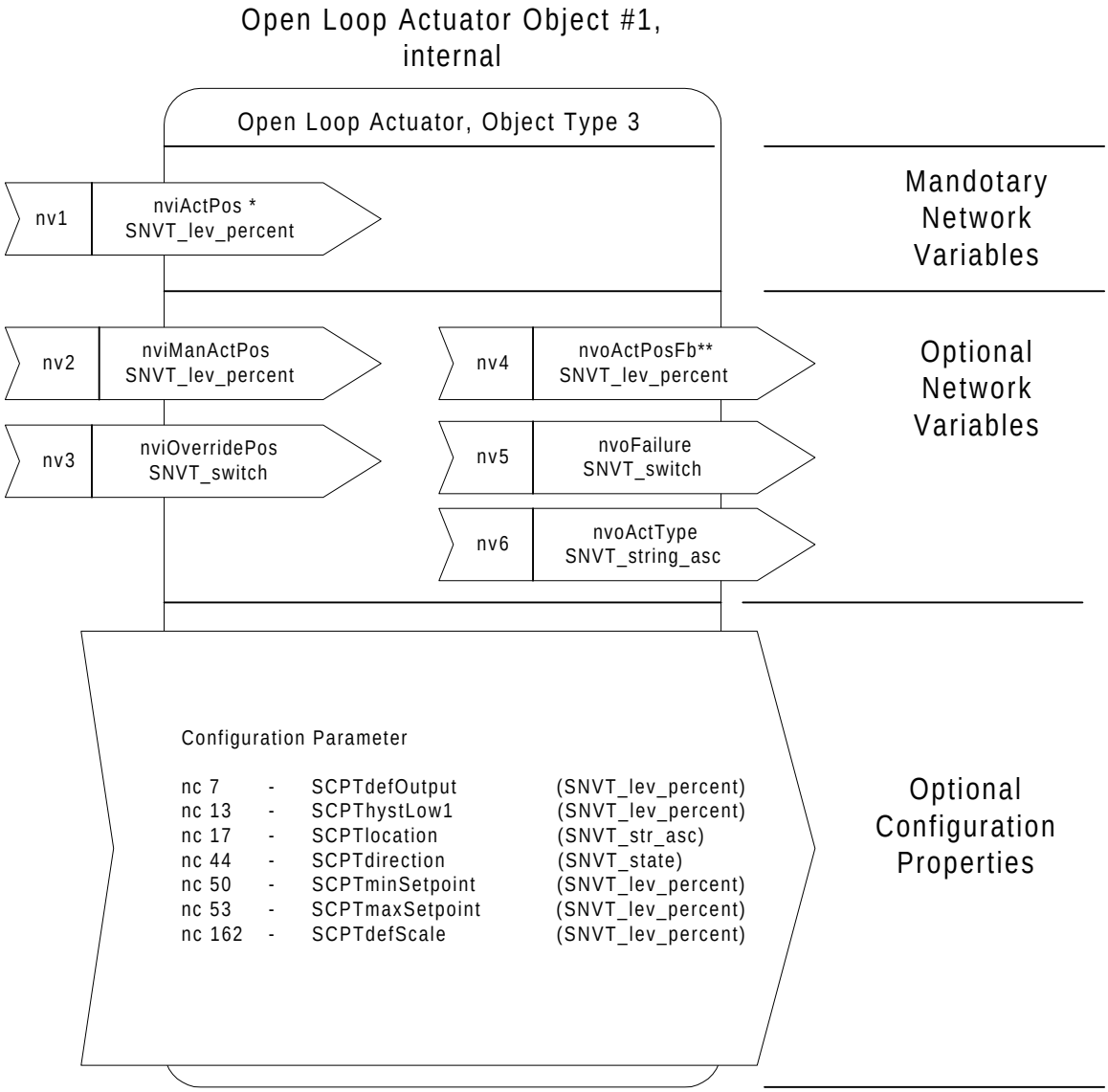
Network Variable	**Type {Ranges / States} {Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
Mandatory Network Variables		Definition	Explanation
nviRequest			
		manual_control: If the *manual_control* bit is ONE two situations are possible • The SNVT *nviManActPos* for the manual controlling of an actuator is activated (≠ 163.835). The *manual_control* bit will be set to zero again, if the *nviManActPos* will be set to INVALID • a manual caused synchronization is in process (see also chapter 3.2.3)	
Optional Network Variables		Definition	Explanation
nvoFileDirectory			
	SNVT_address {The valid range for the file directory address is any value within the user data memory space of the Neuron Chip}		The SNVT *nvoFileDirectory* is needed to get an access to the Configuration Parameter File which includes all SCPTs.

SCPT Master List - names from Echelon	**Type {Range} {Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
		Definition	Explanation
SCPTmaxRcvTime			
	SNVT_time_sec {0...6553s} {0} = OFF Only integer values are accepted and numbers after a comma will be ignored.	Is no value [nviActPos] in a certain timeframe [SCPTmaxRcvTime] received, the related actuator will run to the position defined in [SCTPdefOutput]. Supervision of the SNVTs OLAO1: [nviActPos[0]] OLAO2: [nviActPos[1]]	Heartbeat for the Controller / Actuator supervision [SCPTmaxRcvTime] = 0 Supervision is deactivated [SCPTmaxRcvTime] = for example 60 [sec] Heartbeat timer expires, if for 60s no new value is received on [nviActPos], the actuator will run to the position, defined in the SCPT [SCTPdefOutput].
SCPTmaxSendTime			
	SNVT_time_sec {0...6553s} {0} = OFF Only integer values are accepted and numbers after a comma will be ignored.	Normally the output network variable values will only be updated if change in value. But the values will be updated too if the timeframe defined in [SCPTmaxSendTime] is passed from at least one variable. Supervision of the SNVTs: OLAO1: [nvoActPosFb[0]] OLAO2: [nvoActPosFb[1]] OLSO1: [nvoValueS[0]] OLSO2: [nvoValueS[1]]	In the SCPT [SCPTmaxSendTime] the maximum timeframe for the cyclic sending of the output network variables values can be adjusted. [SCPTmaxSendTime] = 0 means deactivated [SCPTmaxSendTime] = for example 60 [sec] Timeframe is passed, if for 60s a SNVT was not updated. In this case an update of this SNVT will be done.

3.1.1 Behaviour of the Node Object after Online

- The Timer which counts the [SCPTmaxRcvTime] down will be reset to the adjusted value.
- The Timer which counts the [SCPTmaxSendTime] down will be reset to the adjusted value

3.2 The Open Loop Actuator #1 for the internal Actuator (included in ML7420G1004, ML7420G1012 and ML7420G1020)



* Supervision possible with nciMaxRcvTime included inNode Object

** Timeframe for sending value over LON-Bus adjustable via nciMaxSendTime included in Node Object

Fig. 10 Open Loop Actuator #1 for the internal Actuator

Network Variable	**Type** **{Ranges / States}** **{Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
Mandatory Network Variables		Definition	Explanation
nviActPos[0]t			
	SNVT_lev_percent {0...100% ; 163.835} {163.835} = INVALID	**nviActPos[0] = 163.835** [INVALID] ⇒ After Power on or Reset ⇒ Controller is sending [INVALID]. The actuator will run to the safety position defined in [SCPTdefOutput]. **[SCPTminSetpoint] < nviActPos[0] < [SCPTmaxSetpoint]** ⇒ The actuator runs to the position received from the controller. **[SCPTminSetpoint] ≥ nviActPos[0] ≥ [SCPTmaxSetpoint]** ⇒ The actuator synchronizes (see chapter 3.2.3)	The actuating signal received from a controller defined in the range of 0...100% which defines the stroke position of the connected valve. Received values below 0% are calculated like 0% and values over 100% will be calculated with 100%. Is the value 163.835 =[INVALID] received, the actuator drives to the safety position which is defined in the [SCPTdefOuput]. The received values will be never overwritten by the software of the actuator during normal operation. The variable [nviActPos[0]] is only be used, if the variable with higher priority [nviManActPos[0]] for manual operation is set to [INVALID] = 163.835. The cyclic received actuator signal can be supervised. In this case, the maximum timeframe for the cyclic received values can be defined in the SCPT [SCPTmaxRcvTime]. Is this timeframe passed, the actuator will be running to the safety position defined in [SCPTdefOuput].
Optional Network Variables			
nvoActPosFb[0]			
	SNVT_lev_percent {0...100% ; 163.835} {actual stem position}	**nvoActPosFb[0] = 163.835** (INVALID) ⇒ No defined stroke position feedback possible in case of fault. **0% ≤ nvoActPosFb[0] ≤ 100%** ⇒ The actual stem position	This value shows the actual stem position.
nvoFailure			
	SNVT_switch {value=0/state=0 or value=100,state=1} {value=0/state=0}	**{ value = 0 / state = 0 }** Means none of the three failure status bits is set = no errors in OLA1 deteced = normal operation **{ value = 100 / state = 1 }** At least one of the three failure status bits is set = error detected. Take the nviRequest {update_status} in the node object to check, which failure bit is set.	This SNVT could be bound e.g. to a warning light or a warning horn to make a online “actuator fails” detection possible. If one of the three bits {unable_to_measure, out_of_limits or mechanical_fault} occurs nvoFailure reacts. nvoFailure will be updated every 60s.

Network Variable	**Type** **{Ranges / States}** **{Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
Mandatory Network Variables		Definition	Explanation
nviManActPos[0]			
	SNVT_lev_percent {0...100% ; 163.835} {163.835} = INVALID	**nviManActPos[0] = 163.835** (INVALID) ⇒ Manual operation is switched off **0% ≤ nviManActPos[0] ≤ 100%** ⇒ The actuator is running to the manual adjusted value. [nviManActPos[0]] is not depending on [SCPTmaxSetpoint] and [SCPTminSetpoint]	The actuator can be operated manually. The heartbeat timer [SCPTmaxRcvTime] for [nviActPos[0]] is operating in the background (if selected). The manual operation has in each case a higher priority than the received actuator signal [nviActPos[0]]. After changing the value for manual operation [nviManActPos[0]] to [INVALID] the actuator is running according to the received values of [nviActPos[0]]. Received values below 0% are calculated like 0% and values over 100% will be calculated with 100%. The only exception is the value 163.835 = [INVALID]. Is this value received, the actuator drives to the safety position which is defined in the [SCPTdefOutput].
nviOverridePos[0]			
	SNVT_switch (see table 3.2b)	[nviOverridePos[0]] is not depending on [SCPTmaxSetpoint] and [SCPTminSetpoint] (see also table 3.2b)	
nvoActType			
sd_string ("@1|4;")	SNVT_string_asc {31 asci-signs } { “600N, 20mm stroke, 60s [50Hz]” }	Needed for future use.	This string shows the actuator type, its stroke and its runtime at 50Hz mains frequency.

SNVT	**Priority #**
nviManActPos[0]	1
nviOverridePos[0]	2
nviActPos[0]	3

Table 3.2a Priorities of the control signals for the internal actuator

State	**nviOverride Pos[0]**		**Description**
	State	**Value**	
ON-State	TRUE	> 0	**Override state**- The motor runs to the position defined in the SCPT [SCPTdefScale] if this one is not 163.835 [=INVALID]
OFF-State	TRUE	= 0	No override state - normal operation
OFF-State	FALSE	= x	No override state - normal operation **Default: OFF-State = {0 / 0} after power on / reset**

x **=** no functionality = no use = no need to define

Table 3.2b nviOverride Pos[0]

SCPT Master List - names from Echelon	**Type {Ranges / States} {Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
		Definition	Explanation
SCPTdefOutput			
	SNVT_lev_percent {0...100%} {0%}	This SCPT defines the condition which will happen if the timeframe [SCPTmaxRcvTime] of the watchdog is passed. For example: The controller fails and no values are received. In this case the actuator can run to the adjusted safety position. The selection should be done according to the application requirements.	Safety position in case of controller failure (only if [SCPTmaxRcvTime] >0).
SCPThystLow1			
	SNVT_lev_percent {0.5...100%} but 0-100% is allowed ⇒ {0.5%}	Only for [nviActPos[0]] For a higher accuracy it is possible to choose 0% instead of 0.5%. In this special case only the accuracy is higher, but the range is nevertheless 0.5...100%.	The actuator only moves if the change in value of [nviActPos[0]] compared to the last value which moved the actuator is larger than [SCPThystLow1].
SCPTlocation			
	SNVT_str_asc {31 ascii-signs} {Internal Actuator}	The location relates to the object and not to the node.	[SCPTlocation] can be used to provide descriptive physical location information of the internal Actuator.
SCPTdirection			
	SNVT_state {Bit 0 = 1 or 0} {1000000000000000}	**Direct Mode of the Actuator** ⇒ nviActPos[0] = 0% (Stem retracted) nviActPos[0]= 100% (Stem extended) Status Led blinking : **4s on/1s off** **Reverse Mode of the Actuator** ⇒ nviActPos[0] = 0%(Stem extended) nviActPos[0] = 100% (Stem retracted) Status LED is blinking : **1s on/4s off**	Inverse the direction of action **1**000000000000000 = Direct **0**000000000000000 = Reverse
SCPTmaxSetpoint			
	SNVT_lev_percent {-163.84...163.835%} {100%}	Defines the maximum network input value according to the corresponding stroke end value. Input values equal or above [SCPTmaxSetpoint] causes the Actuator to run to the end position of the valve. [SCPTmaxSetpoint] is only valid for nviActPos[0].	[SCPTmaxSetpoint] and [SCPTminSetpoint] can be used to adjust the acctivity of the proportional range concerning the SNVT [nviActPos[0]] and to adjust limitations of this range. [SCPTmaxPosition] must be higher than [SCPTminSetpoint].

SCPT Master List - names from Echelon	Type {Ranges / States} {Factory Settings}	Adjustment of the different functionalities and the explanation of the shown values.	
		Definition	Explanation
SCPTminSetpoint			
	SNVT_lev_percent {-163.84...163.834%} {0%}	Defines the minimum network input value according to the corresponding stroke end value. Input values equal or below [SCPTminSetpoint] causes the actuator to run to the end position of the valve. [SCPTminSetpoint] is only valid for nviActPos[0].	[SCPTmaxSetpoint] and [SCPTminSetpoint] can be used to adjust the activity of the proportional range concerning the SNVT [nviActPos[0]] and to adjust limitations of this range. [SCPTmaxSetpoint] has to be higher than [SCPTminSetpoint].
SCPTdefScale			
	SNVT_lev_percent {0...100%; 163.835%} {0%}	Every Position between 0 and 100% can be chosen. 163.835 (=INVALID) could be chosen to make the overwrite inactive without deleting the binding.	If [nviOverridePos[0]] turns to {TRUE, >0} the actuator runs to the position defined in [SCPTdefScale] (see table 3.2b)

3.2.1 Behaviour of the Actuator after Power On, Reset, Offline / Online

Power On behaviour and Reset

The actuator needs few seconds before normal run is active. The actuator is verifying the stem position of the actuator by using the internal potentiometer. The value is send to the network via [nvoActPosFb[0]].
All input variables will be set to INVALID. After that 1 x polling of all input SNVTs occurs. If no result from polling and no new value was received within 10s, the actuator runs to the safety position defined in [SCPTdefOutput]. During the max. 10s wait time the actuator makes no movement.

Offline / Online

Offline

If the actuator is taken offline by a network management tool the application program of the device stops and the following actions will be done before the offline mode takes place respectively the online mode is set again.
The motor stops, if it is running. The SNVT nvoActPosFb[0] will be set to INVALID (163.835), to indicate that the actuator is in an undefined situation.

Online

- 1 x polling of all input SNVTs. Till a result from polling or a new value will be received the actuator (motor model) works with the old SNVT values.
- The Timer which counts the [SCPTmaxRvcTime] down will be reset to the adjusted value (included in node object).
- The Timer which counts the [SCPTmaxSendTime] down will be reset to the adjusted value (included in node object).

3.2.2 Diagrams of the relation between nviActPos[0], nvoActPosFb[0], SCPTminSetpoint, SCPTmaxSetpoint and SCPTdirection

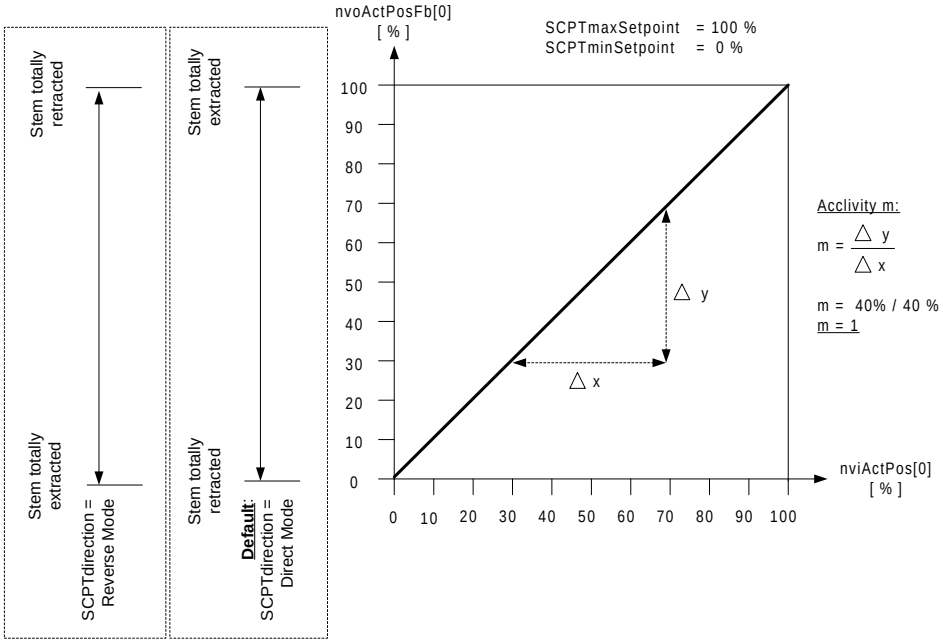


Fig. 11 Diagram# 1, Default Situation

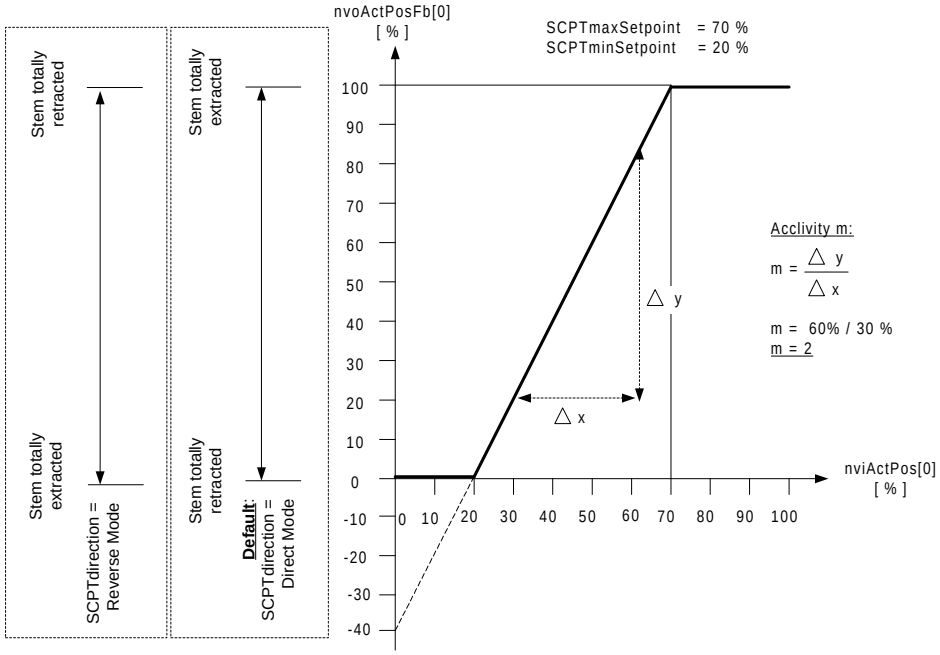


Fig. 12 Diagram# 2, Example for customer adjustment acclivity = 2

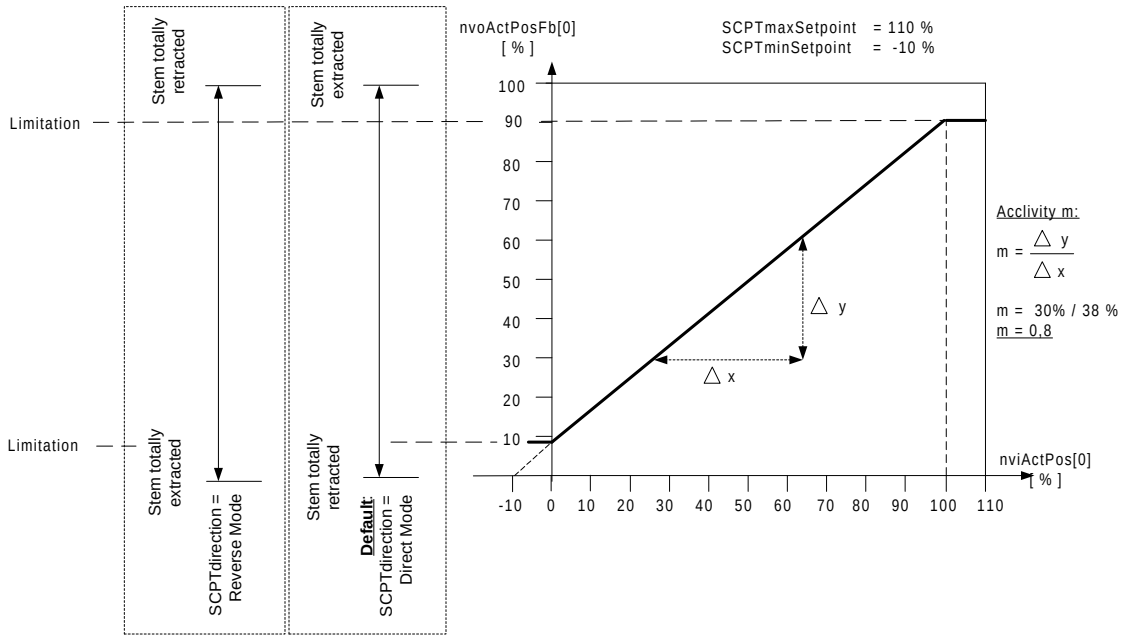


Fig. 13 Diagram# 3, Example for customer adjustment limitation for motor stroke 10%...90%

3.2.3 Adaption of the Internal Actuator

3.2.3.1 Automatically adaption during normal operation

The automatic synchronization works the whole time during normal operation.

Direct Mode

If the limit switch stem retract is reached and the requested motor position is 0% the actuator automatically makes an adaption.

If the limit switch stem extract is reached and the requested motor position is 100% the actuator automatically makes an adaption too.

Reverse Mode

If the limit switch stem retract is reached and the requested motor position is 100% the actuator automatically makes an adaption.

If the limit switch stem extract is reached and requested motor position is 0% the actuator automatically makes an adaption too.

If an adaption was done at both limit switch positions the highest accuracy will be raised.

3.2.3.2 Manual caused adaption over Service Pin in housing of actuator

By pressing the Service Pin permanently for 5s the actuator moves from one end position to the other and an adaption will be done. During this operation no other functions can be served. This function can be used to have most accuracy from the beginning of work. Manual caused synchronization can't be interrupted.

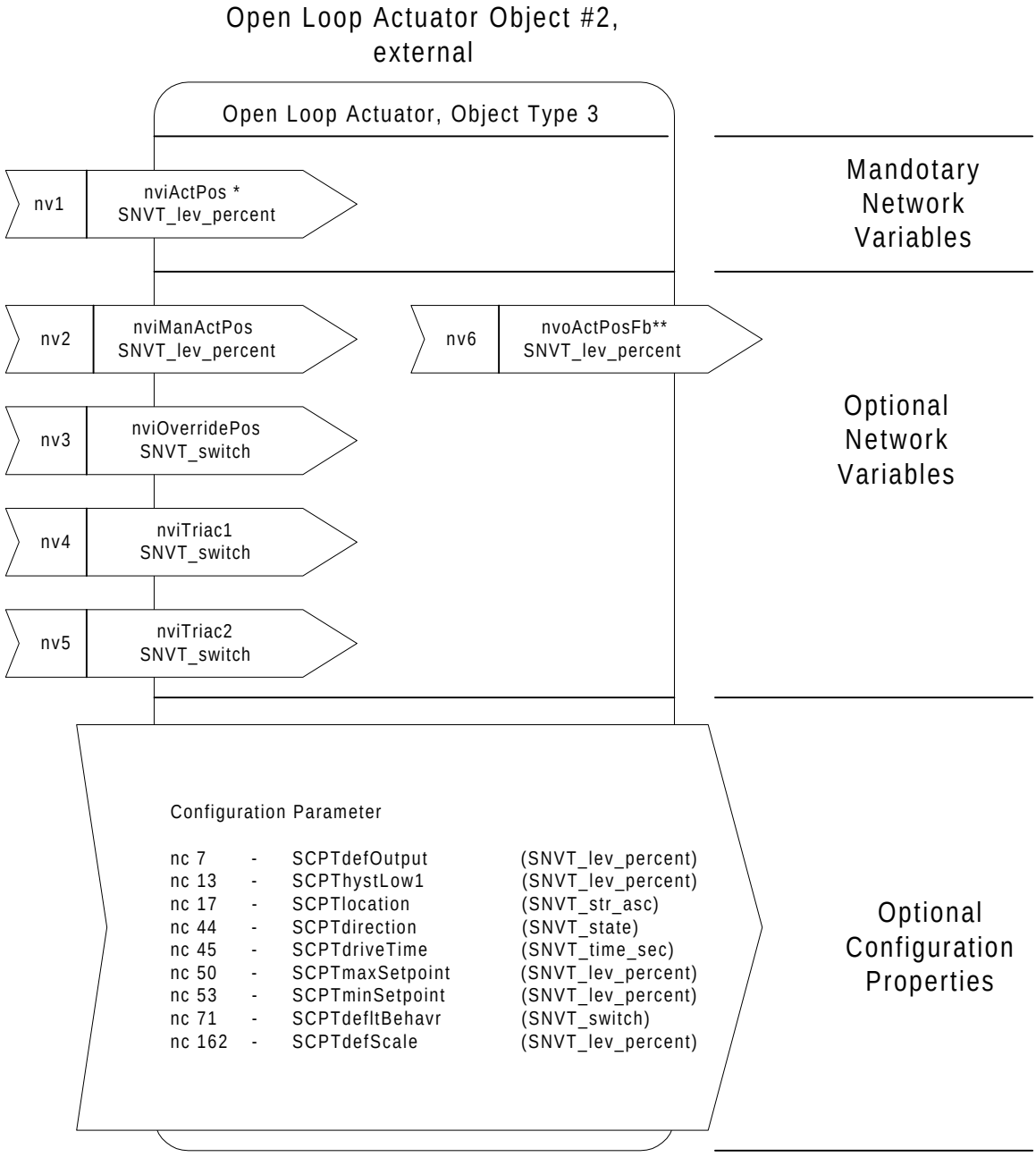
3.2.3.3 Manual caused adaption over Network Management Tool

If the RQ_SELF_TEST Mode will be requested over the network (node object) the actuator moves from one end position to the other and an adaption will be done. During this operation no other functions can be served. This function can be used to have most accuracy from the beginning of work. Manual caused synchronization can't be interrupted.

3.2.3.4 Error detection in case of adaption fails

NOTE: The stroke adaption is limited to factory defined norm values. If the deviation of the new detected values is far out of acceptable values the out_of_limit Bit in the node object will be set to 1. Additionally the nvoFailure network variable will be set to {value =100, state =1}.

3.3 The Open Loop Actuator #2 for an external Actuator or two Digital Outputs (included in ML7420G1020)



* Supervision possible with nciMaxRcvTime included inNode Object

** Timeframe for sending value over LON-Bus adjustable via nciMaxSendTime included in Node Object

Fig. 14 Open Loop Actuator #2

Network Variable / selfdocumentation string	**Type {Ranges / States} {Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
Mandatory Network Variables		Definition	Explanation
nviActPos[1]			
	SNVT_lev_percent {0...100% ; 163.835} {163.835} = INVALID	**nviActPos[0] = 163.835** [INVALID] ⇒ After Power on or Reset ⇒ Controller is sending [INVALID]. The actuator will run to the safety position defined in [SCPTdefOutput]. **[SCPTminSetpoint] < nviActPos[0] < [SCPTmaxSetpoint]** ⇒ The actuator runs to the position received from the controller. **[SCPTminSetpoint] ≥ nviActPos[0] ≥ [SCPTmaxSetpoint]** ⇒ Overtravel Situation takes place (see chapter 3.3.3)	The actuating signal received from a controller defined in the range of 0...100% which defines the stroke position of the connected valve. Received values below 0% are calculated like 0% and values over 100% will be calculated with 100%. If the value 163.835 = INVALID is received, the actuator drives to the safety position which is defined in the [SCPTdefOuput]. The variable [nviActPos[1]] is only used, if the variable with higher priority [nviManActPos[1]] for manual operation is adjusted to [INVALID] = 163.835. The cyclic received actuating signal can be supervised. In this case, the maximum timeframe for the cyclic received values can be defined in the SCPT [SCPTmaxRcvTime]. Is this timeframe passed, the actuator will run to the safety position defined in [SCPTdefOuput].
Optional Network Variables			
nvoActPosFb[1]			
	SNVT_lev_percent {0...100% ; 163.835} {163.835} = INVALID	**nvoActPosFb[1] = 163.835** (INVALID) ⇒ After power on or reset ⇒ No defined stroke position feedback possible in case of fault. ⇒ The SNVT [nviTriac1], [nviTriac2] or both are used as two independent Digital Outputs and the motor model is deactivated. **0% ≤ nvoActPosFb[1] ≤ 100%** ⇒ The actual calculated position of the stem.	This value shows the actual **calculated** stem position. The physical position and the calculated stem position can be different because the connected motor is a synchronous motor without a physical feedback.

Network Variable / selfdocumentation string	**Type {Ranges / States} {Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
Mandatory Network Variables		Definition	Explanation
nviManActPos[1]			
	SNVT_lev_percent {0...100% ; 163.835} {163.835} = INVALID	**nviManActPos[1] = 163.835** (INVALID) ⇒ After Power on or Reset ⇒ Manual operation is switched off **0% < nviManActPos[1] < 100%** ⇒ The actuator is running to the manual adjusted value. **0% ≥ nviManActPos[1] ≥ 100%** ⇒ Overtravel situation takes place (see chapter 3.3.3) [nviManActPos[1]] is not depending on [SCPTmaxSetpoint] and [SCPTminSetpoint]	The actuator can be operated manually. If you want most accuracy to set the actuator to a position, you first have to drive it to the position where the stem is totally retracted (0% or 100%). After that you can set the motor to the desired position. The heartbeat timer [SCPTmaxRcvTime] for [nviActPos[1]] is operating in the background (if selected). The manual operation has in each case a higher priority than the received actuating signal [nviActPos[1]]. After changing the value for manual operation [nviManActPos[1]] to [INVALID] the actuator is running according to the received values of [nviActPos[1]]. Received values below 0% are calculated like 0% and values over 100% will be calculated with 100%. The only exception is the value 163.835 = [INVALID]. Is this value received, the actuator drives to the safety position which is defined in the [SCPTdefOutput].
nviOverridePos[1]]			
	SNVT_switch {see table 3.3a)	[nviOverridePos[1]] is not depending on of [SCPTmaxPosition] and [SCPTminPosition] (see table 3.3a)	Can be used for example to connect an emergency switch. If the emergency situation takes place the motor runs to the position defined in [SCPTdefScale].
nviTriac1			
	SNVT_switch {see table 3.3b)	See table 3.3b (Hardware connection see page 4)	Can be used, for example to control a physical connected pump.
nviTriac2			
	SNVT_switch (see table 3.3b)	See table 3.3b (Hardware connection see page 4)	Can be used, for example to control a physical connected pump.

State	**nviOverride Pos[1]**		**Description**
	State	**Value**	
ON-State	TRUE	> 0	**Override state**- The motor runs to the position defined in the SCPT [SCPTdefScale] if this one is not 163.835 [=INVALID]
OFF-State	TRUE	= 0	No override state - normal operation
OFF-State	FALSE	= x	No override state - normal operation **Default:** **OFF-State = {FALSE / 0}after power on / reset**

x **=** no functionality = no use = no need to define

Table 3.3a nviOverride Pos[1]

Triac States	**nviTriac1**		**nviTriac2**	
	State	**Value**	**State**	**Value**
Triac1 ON	TRUE	> 0	-	-
Triac1 OFF	TRUE	= 0	-	-
	FALSE	= x	-	-
Triac2 ON	-	-	TRUE	> 0
Triac2 OFF	-	-	TRUE	= 0
	-	-	FALSE	= x
deleted				

x **=** no need to define

Table 3.3b Triac functions

SCPT direction- Bit 2	Motor Model Active	SVNT	Priority #
1	Yes	nviManActPos[1]	1
		nviOverride Pos[1]	2
		nviActPos[1]	3
0	No (Triac Mode active)	nviTriac1 and nvi Triac2 can be used for e.g. controlling of two pumps	-

Table 3.3c Selection between motor model for an external floating actuator or alternative two digitaloutputs. Additionally the priorities of the control signal for the external floating actuator

⚠ ATTENTION

In the Triac Mode it has to be ensured that no actuator is connected to the screwterminal otherwise it is possible that the actuator will be destroyed.

SCPT Master List - names from Echelon	Type {Ranges / States} {Factory Settings}	Adjustment of the different functionalities and the explanation of the shown values.	
		Definition	Explanation
SCPTdefOutput			
	SNVT_lev_percent {0 ; 100% ; 163.835} {163.835} = INVALID	This SCPT defines the condition hat will happen if the timeframe in [SCPTmaxRcvTime] is passed. For example : The controller fails and no values are received. In this case the actuator can run to a safety position (0 or 100%) or stops immediately (INVALID). The selection should be done according to the application requirements.	Safety position in case of controller failure (only if SCPTmaxRcvTime >0)
SCPThystLow1			
	SNVT_lev_percent {0.5...100%} {2%}	Only for [nviActPos[1]]	The actuator only moves if the change in value of [nviActPos[1]] compared to the last value which moved the actuator is larger than [SCPThystLow1].
SCPTlocation			
	SNVT_str_asc {31 ascii-signs} {External Actuator}	The location relates to the object and not to the node.	[SCPTlocation] can be used to provide descriptive physical location information of the External Actuator.

SCPT Master List - names from Echelon	**Type {Ranges / States} {Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
		Definition	Explanation
SCPTdirection			
	SNVT_state {Bit 0 = 1 or 0} for Direct / Reverse adjustment {Bit 2 = 1 or 0} for selection between MotorModel or DigitalOutput-Mode Bit 1 and Bits 3-15 are not used {1010000000000000}	**Direct Mode of the Actuator** ⇒ a) nviActPos requires a movement to the direction 100% (rising) ⇒ Triac1 = ON / Triac2 = OFF b) nviActPos requires a movement to the direction 0% (falling) ⇒ Triac1 = OFF / Triac2 = ON **Reverse Mode of the Actuator** ⇒ a) nviActPos requires a movement to the direction 100% (rising) ⇒ Triac1 = OFF / Triac2 = ON b) nviActPos requires a movement to the direction 0% (falling) ⇒ Triac1 = ON / Triac2 = OFF **MotorModel activated** ⇒ Motormodel active = a floating actuator has to be connected to the srewterminal **Triac Mode activated** ⇒ Single Digital Ouput usage ⚠ **ATTENTION:** **In this mode it has to be ensured that no actuator is connected to the screwterminal otherwise it is possible that the actuator will be destroyed.** **In this mode both Digital Outputs can be used for e. g. connect and control two different pumpss independent from each other (nviTriac1 and nviTriac2)**	Inverse the direction of action **1**010000000000000 = Direct **0**010000000000000 = Reverse 10**1**0000000000000 = Motormodel 10**0**0000000000000 = Single Digital Output usage ⚠ see table 3.3c
SCPTdriveTime			
	SNVT_time_sec {10...6553.5s} {60s}	Example: If 60s are adjusted the motor runs for 60s. (The installation team must recognize that a synchronous motor runs 16.6% faster if the main frequency is 60Hz instead of 50Hz).	To adapt the valve stroke to the motor runtime of the actuator the SCPTdriveTime has to be used. The shortest runtime is 10s. For runtimes < 20s the sensitivity defined in [SCPThystLow1] should not be smaller than 2%.
SCPTmaxSetpoint			
	SNVT_lev_percent {0...100%} {100%}	Defines the maximum network input value according to the correspond stroke end value. Input values equal or above [SCPTmaxSetpoint] causes the Actuator to run to the end position of the valve.	The [SCPTmaxSetpoint] and [SCPTminSetpoint] can be used to adjust the proportional range of the [nviActPos[1]] and [nviManActPos[1]]. [SCPTmaxSetpoint] must be higher than [SCPTminSetpoint].
SCPTminSetpoint			
	SNVT_lev_percent {0...100%} {0%}	Defines the minimum network input value according to the corresponding stroke end value. Input values equal or below [SCPTminSetpoint] causes the Actuator to run to the end position of the valve.	The [SCPTmaxSetpoint] and [SCPTminSetpoint] can be used to adjust the proportional range of the [nviActPos[1]] and [nviManActPos[1]]. [SCPTmaxSetpoint] must be higher than [SCPTminSetpoint].

SCPT Master List - names from Echelon	Type {Ranges / States} {Factory Settings}	Adjustment of the different functionalities and the explanation of the shown values.	
		Definition	Explanation
SCPTdefBehavr			
	SNVT_switch (see table 3.3d)	see table 3.3d	Defines the behavior of the actuator after power on.
SCPTdefScale			
	SNVT_lev_percent {0; 100%; 163.835%} {0%}	Position 0 or 100% can be chosen. 163.835 (=INVALID) could be chosen to make the overwrite inactive without deleting the binding	If [nviOverridePos[1]] turns to {TRUE, >0} the actuator runs to the position defined in [SCPTdefScale] (see table 3.3a)

⚠ ATTENTION

The change of the [SCPTdefBehavr] will be done during offline/online. This action will be executed only after a power on situation (power fail) or a reset. But be careful: a software-reset causes a reset of the internal actuator too.

Variant	SCPTdefBehavr		Meaning
	State	Value	
Default: Variant 1	x	0	The actuator makes no movement and waits. The first received value from the controller or manual will be saved as the actual position of the motor - no movement. The second received value will be compared with the first and the motor runs to the new required position (and so on...). If the first received value is 0 or 100% immediately an overtravel situation occurs and the motor runs to 0 or 100%.
Variant 2	x	1	After Power On the actuator assumes that its position is CLOSE. The actuator makes no movement and waits. The [nviActPosFb] is set to 0%. The first received value will be compared with 0%, and the motor runs to the new required position (and so on...). If the first received value is 0%, in this special situation the motor starts an overtravel to 0.
Variant 3	x	2	After Power On the actuator assumes that its position is OPEN. The actuator makes no movement and waits. The [nviActPosFb] is set to 100%. The first received value will be compared to 100%, and the motor runs to the new required position (and so on...). If the first received value is 100%, in this special situation the motor starts an overtravel to 100%.
Variant 4	x	3	After Power On the actuator assumes that its position is 50% between OPEN and CLOSE. The actuator makes no movement and waits. The [nviActPosFb] is set to 50%. The first received value will be compared with 50% and the motor runs to the new required position (and so on...).
Variant 5	x	4	The actuator synchronizes for 100% motor runtime to the position refers to 0% and runs to the position according to the controller/manual value, if one is received. If not, the actuator waits.
Variant 6	x	≥ 5	The actuator synchronizes for 100% motor runtime to the position refers to 100% and runs to the position according to the controller/manual value, if one is received. If not, the actuator waits.

x...no functionality = no use = no need to define

Table 3.3d Different Power On behaviours for the external floating actuator

3.3.1 Behaviour of the Actuator after Power On, Reset, Offline / Online

Power On behaviour and Reset

The Actuator needs a few seconds before normal run is active
The power on / reset behaviour of the external Actuator is defined in the SCPTdefBehavr
All Input Variables will be set to INVALID
The actuator is acting like described in the selected [SCPTdefBehavr]. [nvoActPosFb[1]] will be set to:

Variant 1: [nvoActPosFb[1]] = 163.835% = INVALID
Variant 2: [nvoActPosFb[1]] = 0%
Variant 3: [nvoActPosFb[1]] = 100%
Variant 4: [nvoActPosFb[1]] = 50%
Variant 5: [nvoActPosFb[1]] = 0% and synchronization to 0% starts; no interruption possible
Variant 6: [nvoActPosFb[1]] = 100% and synchronization to 100% starts; no interruption possible

After that...

...1 x polling of all input SNVTs occurs. If no result from polling and no new value was received within 10s, the actuator runs to the safety position defined in [SCPTdefScale]. During the max 10 seconds wait time the actuator makes no movement.

Offline / Online

If the actuator is taken offline by a network management tool the application program of the device stops
and the following actions will be done before the offline mode takes place respectively the online mode is set again.

Case 1, Motormodel active before and after offline / online (SCPTdirection.Bit_2 = 1)

Offline
The motor stops, if it is running, the SNVT *nviActPosFb[1]* will be set to INVALID (163.835), to show that the actuator is in an undefined situation.

Online
1 x polling of all input SNVTs. Until a result from polling or a new value will be received the actuator (motor model) works with the old SNVT values. Only if the [**SCPTdriveTime]** and [**SCPTdirection.bit0]** were changed, the actuator (motor-stroke model) will be reset, because working with the old values doesn't make sense.

The Timer which counts the [SCPTmaxRvcTime] down will be reset to the adjusted value included in node object)

The Timer which counts the [SCPTmaxSendTime] down will be reset to the adjusted value in node object).

Case 2, Motormodel active before offline ⇒ Triac (DO) – mode is active after online

Offline: SCPTdirection.Bit2 = 1 before
The motor stops, if it is running, The SNVT *nviActPosFb[1]* will be set to INVALID (163.835), to show that the actuator is in an undefined situation.

Offline: SCPTdirection.Bit2 = 0 after
1 x polling of all input SNVTs. Until a result from polling or a new value will be received the actuator (motor model) works with

Motor model will be reset. The Triac mode will be set, so the Triac I/Os can be served with the SNVTs nviTriac1 and nviTriac2.

⚠ ATTENTION

In this mode it has to be ensured that no actuator is connected to the screwterminal otherwise it is possible that the actuator will be destroyed.

Case 3, Triac (DO) – mode is active before offline ⇒ Motormodel is active after online

Offline: SCPTdirection.Bit2 = 0 before
The Triac DO's will be set to the OFF-State

Offline: SCPTdirection.Bit2 = 1 after
The motor model starts with the Power On/Reset behaviour adjusted in [SCPTdefBehavr]

The heartbeat timer [SCPTmaxRvcTime] for nviActPos[1] will be activated (included in node object)

3.3.2 Diagrams of the relation between nviActPos[1], nvoActPosFb[1], SCPTminSetpoint, SCPTmaxSetpoint and SCPTdirection

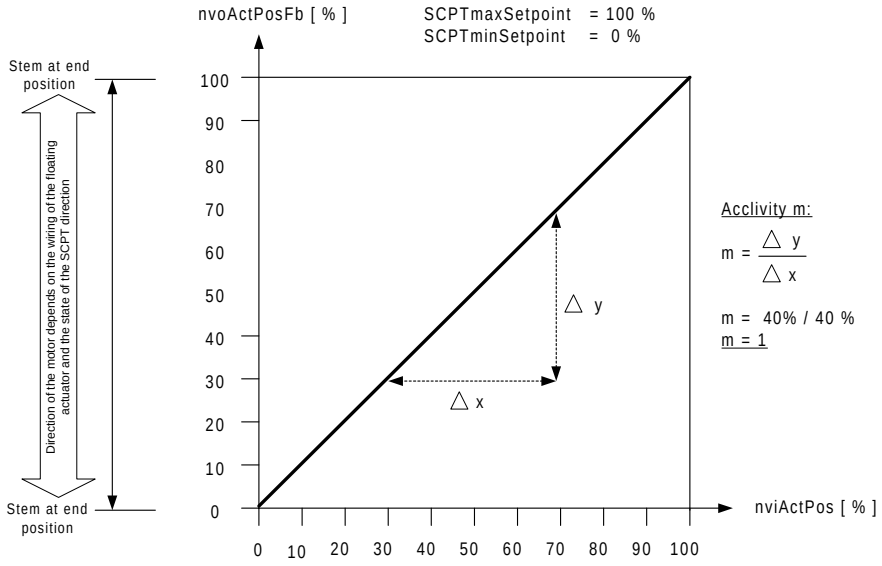


Fig. 15 Diagram# 1 Default Situation

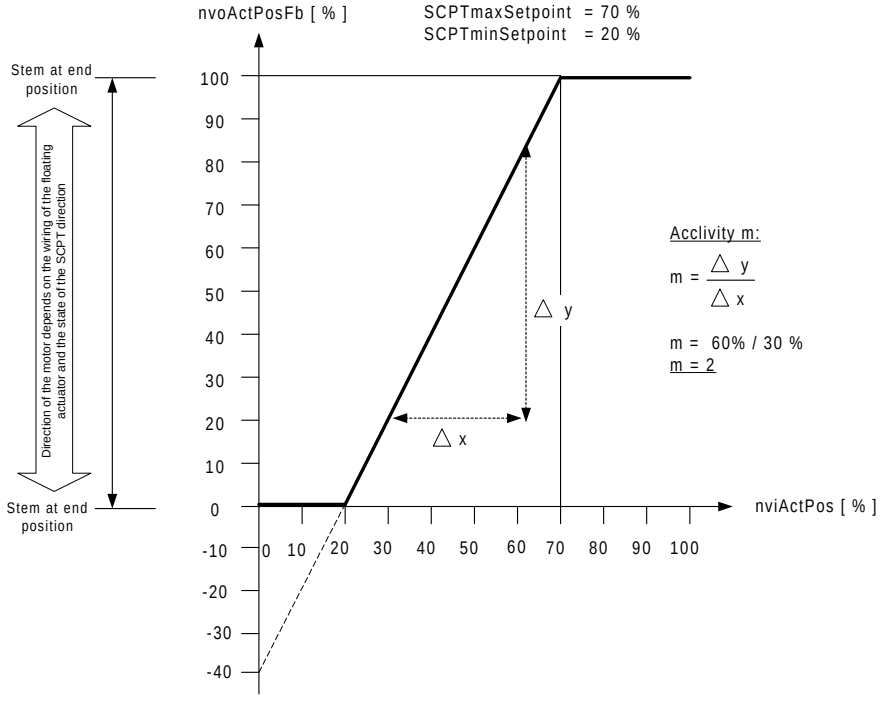


Fig. 16 Diagram# 2 Example for customer adjustment acclivity = 2

3.3.3 The Overtravel and the Pulsate situation of the External Actuator

If the nviActPos[1] is outside of its Overtravel threshold the Overtravel situation occurs (same situation if nviManActPos[1] or nviOverridePos[1] / SCPTdefScale active).

Overtravel situation

Overtravel situation occurs if:

- [SCPTminSetpoint] ≥ nviActPos[1] ≥ [SCPTmaxSetpoint]
- 0% ≥ nviManActPos[1] ≥ 100%
- 0% ≥ SCPTdefScale ≥ 100%
- The SNVT[nviActPos[1] (or if activated [nviManActPos[1]) is reached and NV changes now to the Overtravel threshold the actuator will first run for the difference of its actual position and position CLOSE /OPEN in the direction, where the Overtravel threshold was met, and then an additional 150% motor runtime in the same direction.

Interrupt of Overtravel situation

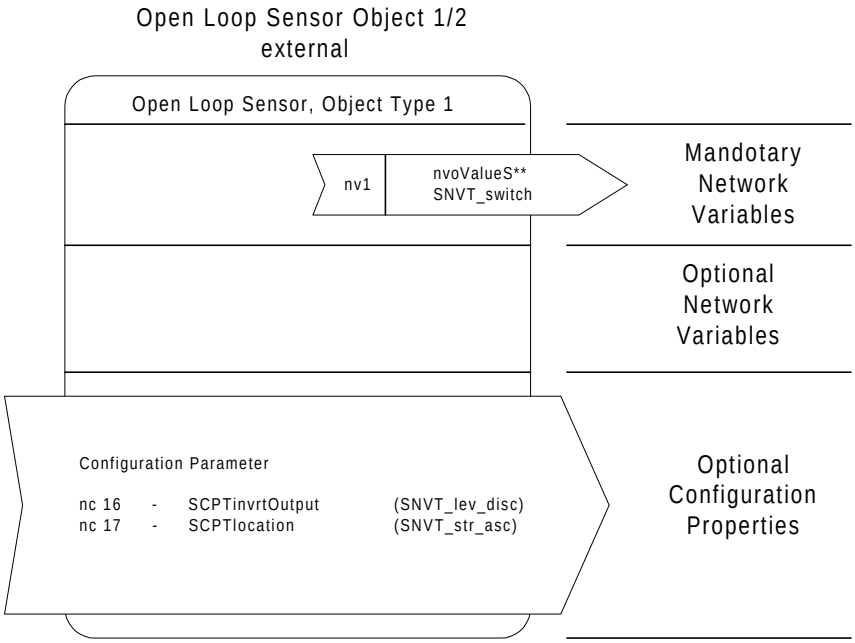
Overtravel situation is interrupted when new NV [nviActPos[1]] or [nviManActPos[1]] exceeds the Overtravel thresholds for the [SCPThystLow1] value. Overtravel is interrupted if:

- new value is ≥ (SCPTminSetpoint + SCPThystLow1) or
- new value is ≤ (SCPTmaxSetpoint – SCPT

Pulsate situation

After the Overtravel Motor Run Time is over, the motor is pulsed again for a fixed 5s time and a pause time specified to 120s. This is necessary, if the external actuator has a manual adjustment knob which is served by the user out of its actual position.

3.4 The Open Loop Sensor Objects #1 and #2 for Digital Inputs (included in ML7420G1012)



** Timeframe for sending value over LON-Bus adjustable via nciMaxSendTime included in Node Object

Fig. 17 Open Loop Sensor Objects #1 and #2

The following descriptions are guilty for nvoValueS[0] and nvoValueS[1].

Network Variable / selfdocumentation string	**Type {Ranges / States} {Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
Mandatory Network Variables		Definition	Explanation
nvoValueS			
	SNVT_switch (see table 3.4 a)	Description see table 3.4a	Contains the actual sensor value in the usual HVAC digital format.
SCPT Master List - names from Echelon	**Type {Ranges / States} {Factory Settings}**	**Adjustment of the different functionalities and the explanation of the shown values.**	
		Definition	Explanation
SCPTinvrtOut			
	SNVT_lev_disc {ST_OFF, ST_LOW, ST_MED, ST_HIGH ST_ON, ST_NUL } {ST_OFF}	Description see table 3.4a	Inversion of the SNVT [nvoValueS]
SCPTlocation			
	SNVT_str_asc {31 asci-signs} {Ext. Digital Switch #1} {Ext. Digital Switch #2}	The location relates to the object and not to the node.	[SCPTocation] can be used to provide descriptive physical location information of the External Sensor.

Configuration Parameter (SCPT...SCPTinvrtOut)		Network Variable (SNVT...nvoValueS)		Meaning
		Value	State	
Default: [SCPTinvrtOut]	ON	100.0	1	Switch Closed
= ST_OFF	OFF	0	0	Switch Opened
[SCPTinvrtOut]	OFF	100.0	1	Switch Opened
= ST_ON, ST_LOW, ST_MED, ST_HIGH	ON	0	0	Switch Closed
[SCPTinvrtOut] = ST_NUL (0xFF)		0	0xFF (= 255 = -1)	Undefined

Table 3.4a nvoValueS[0] and nvoValueS[1]

4 DOWNLOAD, CONFIGURATION AND COMMISSIONING

All configurations of the ILONA-C devices can be made with the Standard Configuration Parameter Types (SCPTs) over the Network (LON-Bus). The descriptions are included in the Software Interface, see chapter 3.

4.1 Download files for application updates

The Application-Software can be downloaded to the neuron chip with the following files:

⚠ ATTENTION

Please make a reset after the download because it is possible that undefined effects of the echelon chip occur after a download

File Extension	Usage	User
.NXE	Downloading the application over the network.	**For customers.** Download of the Application via an API-based LON-Tool (like LonMaker for Windows, NL220...)

4.2 XIF-File for offline configuration

The device interface file can be used for offline configuration. For offline configuration you don't need hardware and so you can create a system architecture comfortable in your office. Commissioning of the LON devices can be done later in the field.

File Extension	Usage	User
.XIF	External Interface File	Needed for offline configuration via an API-based LON-Tool (like LonMaker for Windows, NL220...)

4.3 Domain ID factory setting

	Domain ID lenght (bytes)	Domain ID value (hex)
Domain ID	1	00

If you are searching for a new device in the network via a network management tool, it could be possible that you need the domain ID to search in the right domain.

4.4 Commissioning of the device

4.4.1 Scanning the network for unconfigured devices

Some network management tools support a function which scans the whole network for unconfigured devices. So this is an easy way to find and install the device in the network.

Model (Factory Setting)	Unconfigured
ML7420G1004	x
ML7420G1012	x
ML7420G1020	x

4.4.2 Installation by service pin message

After Power On Reset the three LON actuators are automatically sending a service pin message over the network. Later it is possible to send every time a service pin message by pressing the service pin in the housing. So the devices can be found and installed.

4.4.3 Installation by neuron ID via bar code laser

Every LON actuator will be delivered with 3 neuron ID bar code labels. So the bar code is used to find and to comission the device via a bar code laser, if the used LON Tool supports "scanning the network by neuron ID".

Honeywell

Honeywell Regelsysteme GmbH
Honeywellstr. 2-6
D-63477 Maintal
Phone: (06181) 401-1
Fax: (06181) 401-400

Manufacturing location is certified according to DIN EN ISO 9001

EN2B-1011GE02 R0700