



# PLS2024 Software Users Guide

## Version 3

### Draft 26 Feb. 2004

<b>1</b>	<b>Software functions.....</b>	<b>3</b>
1.1	Software function overview .....	4
1.2	General purpose function profiles:.....	5
1.2.1	Occupancy Sensor (1060), 2 objects.....	5
1.2.2	Occupancy Controller (3071), 2 objects .....	9
1.3	Lighting profiles:.....	12
1.3.1	Lamp Actuator (3040), 2 objects .....	12
1.3.2	Switch (3200), 2 objects.....	15
<b>2</b>	<b>Software History .....</b>	<b>18</b>
2.1	Lonbox PLS2024 Version 3.00.....	18
2.2	Lonbox PLS2024 Version 2.02.....	18
2.3	Lonbox PLS2024 Version 2.01.....	18
2.4	Lonbox PLS2024 Version 2.00.....	18
2.5	Lonbox PLS2024 Version 1.20.....	18
2.6	Lonbox PLS2024 Version 1.03.....	19
2.7	Lonbox PLS2024 Version 1.02.....	19
<b>3</b>	<b>Applications and bindings .....</b>	<b>20</b>
3.1	Manual Control, Light Bindings .....	20
3.1.1	Single on and off. ....	20
3.1.2	Siingle on and off with feed back. ....	20
3.1.3	Multi on and off with feed back.....	20
3.1.4	Single on and off with multi actuators 1. ....	20
3.1.5	Single on and off with multi actuators 2. ....	21
3.1.6	Multi on and off with multi actuators 1. ....	21
3.1.7	Multi on and off with multi actuators 2. ....	21
3.2	Occupancy automatic, Light Bindings.....	21
3.2.1	Occupancy automatic on off. ....	21
3.2.2	Occupancy on off, with manual override.....	22
3.2.3	Occupancy on off, manual override and manual control indicator.....	22
<b>4</b>	<b>Figure list.....</b>	<b>23</b>

# 1 Software functions.

The following section is describing the software function profiles implemented in the Lonbox® Light Controller model PLS2024.

The PLS2024 is designed to fulfil the requirement for controlling lighting in modern Intelligent Building Installations, using the LonWorks® network standard ANSI/EIA/CEN 709.1 or as a stand-alone unit.

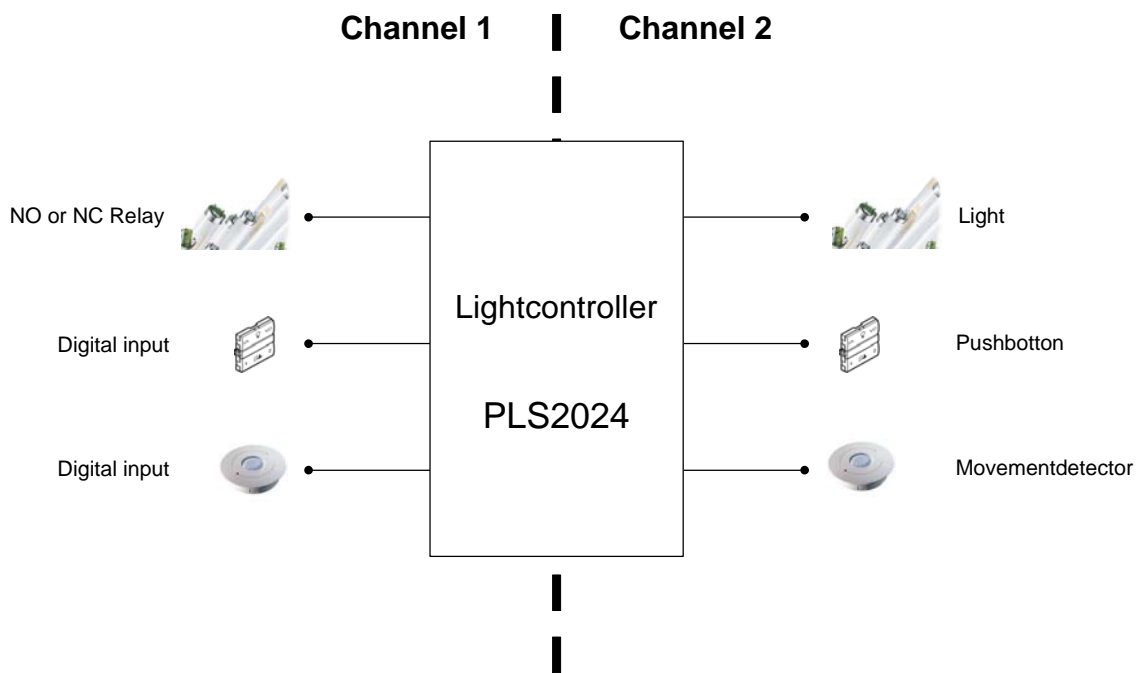
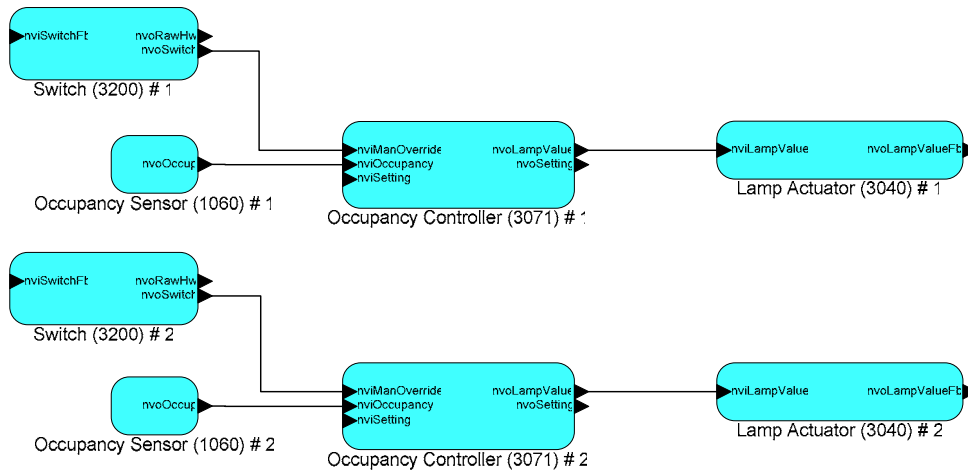


Figure 1 The main application for PLS2024

### 1.1 Software function overview



**Figure 2 Typically use of the Lonbox PLS2024 software, 2 channel light with occupancy and manual overwrite**

## 1.2 General purpose function profiles:

The following is describing the general purpose software functions implemented in the Light Controller model PLS2024.

### 1.2.1 Occupancy Sensor (1060), 2 objects

The following describes the function for the occupancy sensor object. This object is used with a hardware sensor whose output is either in an occupied or unoccupied state. The occupancy sensor object is used to detect occupancy in a room or an area and keep the occupied state until no occupancy can be detected. The output from the occupancy sensor object is typically connected to a controller. A occupancy controller can take care of the proper action and calculates application delay or hold times as appropriate.

PIR #	Connector #	Name in LonMaker list
1	I2	OccupSensor[0]
2	I4	OccupSensor[1]

Figure 1 The Occupancy hardware/software relation

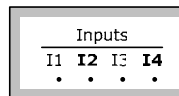


Figure 3 The occupancy hardware input

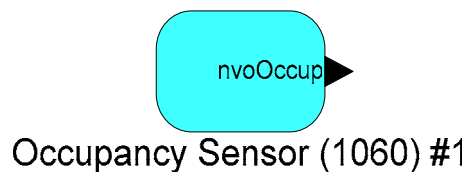


Figure 4 The occupancy sensor object

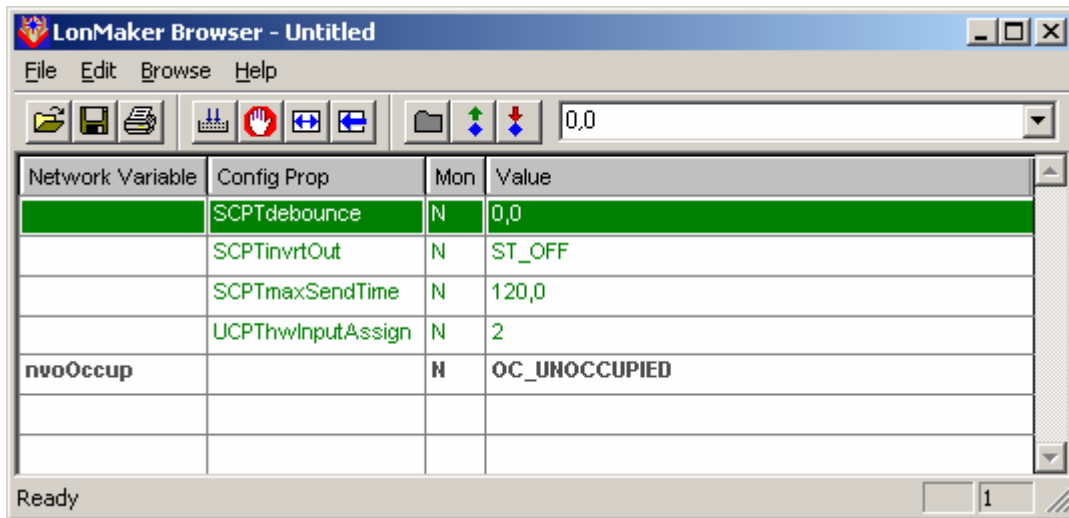


Figure 5 Browsing the occupancy sensor object

### 1.2.1.1 Network Variable Interface

Variable name	Default Value	Type	Description
nvoOccup	OC_UNOCCUPIED	SVNT_occupancy (109)	Sensor output

#### 1.2.1.1.1 nvoOccup

This output network variable provides the qualified state of the hardware sensor output connected to the Comfort and Ligth Controller. The sensor input can be inverted with the configuration property SCPTinvrtOut.

The variable is transmitted when the occupancy state changes from unoccupied to occupied or when the occupancy state has changed from occupied to unoccupied at the frequency of the heartbeat. The maximum update rate is fixed by the heartbeat frequency and the minimum update rate is fixed by the debouncing time. The default service type is acknowledged.

### 1.2.1.2 Configuration Properties

Configuration	Default Value	Description
SCPTdebounce (139)	0,0	Debounce filter time
SCPTinvrtOut (16)	ST_OFF	Invert the output
SCPTmaxSendTime (49)	120,0	Heartbeat time
UCPThwInputAssign	2 or 4	Assigning hardware input

### 1.2.1.2.1 SCPTdebounce

This mandatory configuration network variable defines the debounce time to generate the detection envelop and the OCCUPIED and NON OCCUPIED messages. Valid range is 0.0 - 6553.4 by steps of 0.1s.

The following timing diagram defines the different timers used in the profile, it illustrate the debounce filter function.

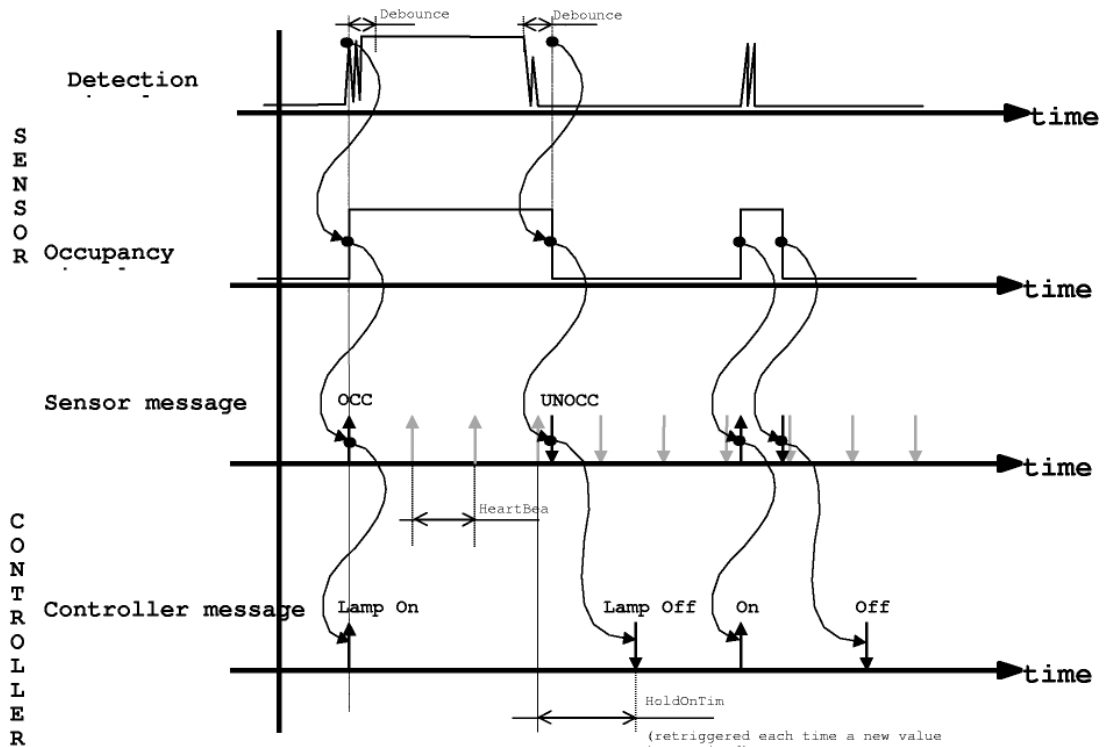


Figure 6 Occupancy timing overview

### 1.2.1.2.2 SCPTinvrtOut

The SCPTinvrtOut configuration property is used to invert the the input signa, so sensor with normal close also can be used.

The SCPTinvrtOut property is either:

SCPTinvrtOut	
ST_OFF	Not inverted input
ST_ON	Inverted input

### 1.2.1.2.3 SCPTmaxSendTime

SCPTmaxSendTime is a configuration property. This configuration defines the repeat period between to value update sent on the bus also called a heartbeat. The aims of the heartbeat is to be sure that the sensor is alive and to permit a controller to have multiple sensors on the same input

**Remark :** Even if the heartbeat is optional, the heartbeat functionality shall be implemented in the Occupancy Sensor with a 2 minutes repeat period, (default value).

### 1.2.1.2.4 UCPThwInputAssign

The configuration property UCPThwInputAssign, is shorted from hardware input assignment. The property is used to make the relation between the input hardware and the software object. The 4 hardware input on the PLS2024 are electrically identically to each other and you can select which hardware input you are using in your software object.

The default hardware assignment to the software objects:

Input	Connector #	Default Assignment#	Object Name	Function
1	I1	1	Switch[0]	Switch
2	I2	2	<i>OccupSensor[0]</i>	<i>PIR</i>
3	I3	3	Switch[1]	Switch
4	I4	4	<i>OccupSensor[1]</i>	<i>PIR</i>

Possibilities for the UCPThwInputAssign property are the following:

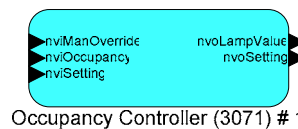
UCPThwInputAssign	
Value	Description
1	Using input 1
2	Using input 2
3	Using input 3
4	Using input 4



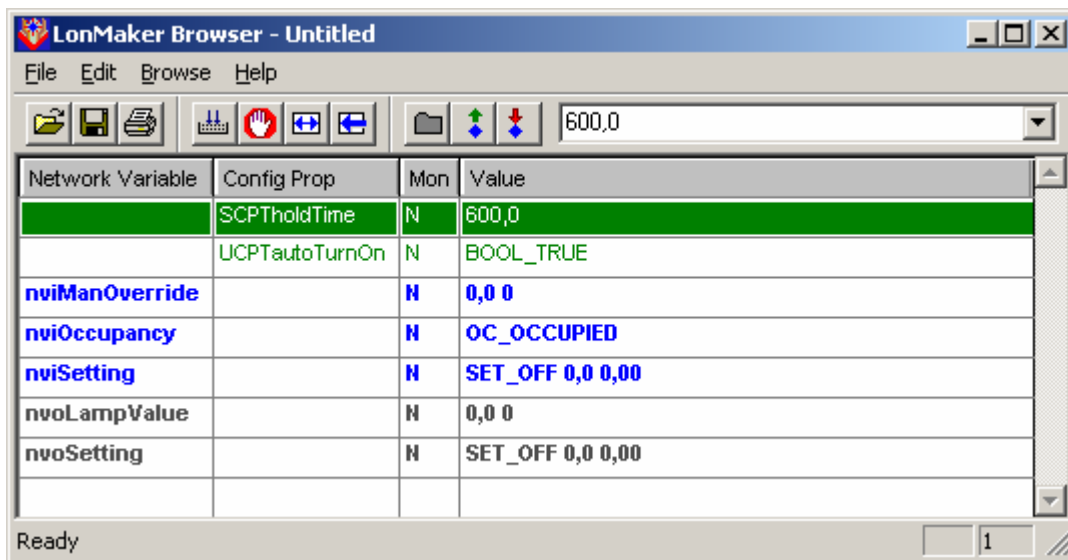
### 1.2.2 Occupancy Controller (3071), 2 objects

The following section is describing the occupancy controller. Typically the occupancy controller input is connected to the occupancy sensor and the output to a lamp. A switch can be used to turn the occupancy controller into ON and OFF mode. An additional switch can override the controller and directly control the lamp. The occupancy controller can also be driven by an occupancy sensor from a neighboring area, secondary input. The secondary function gives the users a secure feeling, due to the light around. The occupancy controller can also operate another controller, eg. a constant light controller.

Controller#	Name in LonMaker list
1	OccupCtrl[0]
2	OccupCtrl[1]



**Figure 7 TheOccupancy controller object**



**Figure 8 Browsing the occupancy controller object**

### 1.2.2.1 Network Variable Interface

Variable name	Default Value	Type	Description
nviManOverride	0,0 0	SNVT_switch (95)	Manual Operation
nviOccupancy	OC_NUL	SVNT_occupancy (109)	Occupancy input
nviSetting	SET_OFF 0,0 0,00	SNVT_setting (117)	Setting control input
nvoLampValue	0,0 0	SNVT_switch (95)	Controller output
nvoSetting	SET_OFF 0,0 0,00	SNVT_setting (117)	Setting control output

#### 1.2.2.1.1 nviManOverride

Manual override input, enables the local and manual control of the lamp value output.

#### 1.2.2.1.2 nviOccupancy

Occupancy status input value, provides the occupancy status for the area.

#### 1.2.2.1.3 nviSetting

Setting input (Auto/Off). This mode can be either be ON(AUTO) or OFF.

#### 1.2.2.1.4 nvoLampValue

Lamp value output. State for the lamp actuator ON or OFF, and the percentage level of intensity.

#### 1.2.2.1.5 nvoSetting

Setting output. Select the operating mode for another controller, e.g., constant light controllers.

### 1.2.2.2 Configuration Properties

Configuration	Default Value	Description
SCPTHoldTime (91)	600	Occupancy delay timer
UCPTautoTurnOn	BOOL_TRUE	Turn on selection, true or false

#### 1.2.2.2.1 SCPTHoldTime

Hold time for occupied state after there is no occupancy detected. This timer is used for both the main occupancy input and for the secondary input.



#### ***1.2.2.2.2 UCPTautoTurnOn***

This configuration property is used to select if the controller automatically switch on the output when occupancy is detected.

<b>UCPTautoTurnOn</b>	<b>Description</b>
BOOL_TRUE	Automatic on activated
BOOL_FALSE	No automatic on activated

### 1.3 Lighting profiles:

The following is describing the lightning software functions implemented in the light controller Lonbox PLS2024.

#### 1.3.1 Lamp Actuator (3040), 2 objects

The following describes the lamp actuator object. The lamp actuator object is used to control the illumination level of a hardware lamp output. The lamp actuator is used with switch and controller devices such as the constant light controller and scene controller. A switch object output nvoSwitch is connected to the input nviLampValue of the lamp actuator. Controller objects can be used between switch type sensors and lamp actuators. In cases of multiple sensors the feedback connection can be used to synchronize a group of switches.

Lamp #	Connector #	Name in LonMaker list
1	Out 1	LampActuator[0]
2	Out 2	LampActuator[1]

Figure 2 The lamp actuators software hardware relation

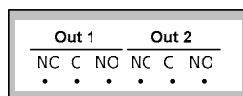


Figure 3 The lamp actuator hardware outputs

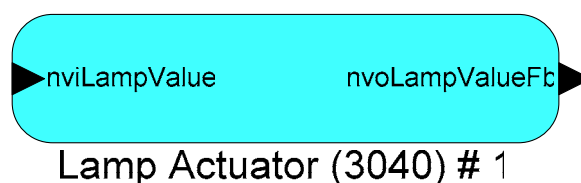


Figure 9 The lamp actuator object

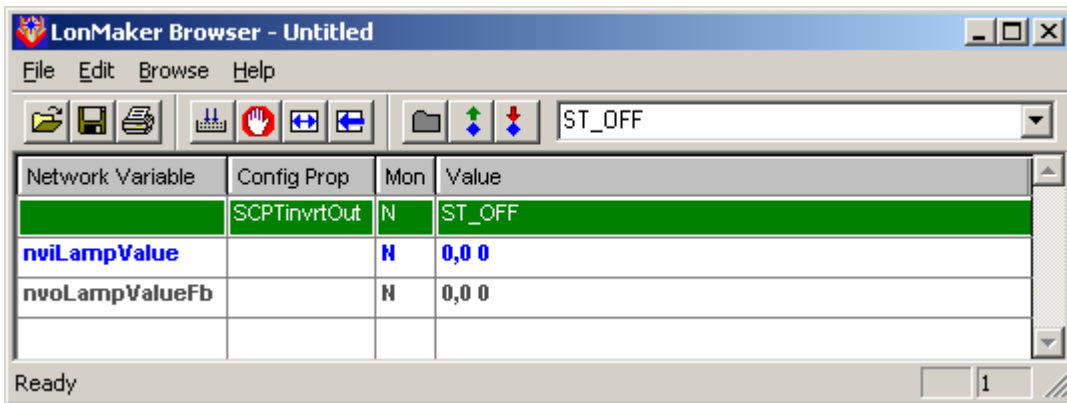


Figure 10 Browsing the lamp actuator object

### 1.3.1.1 Network Variable Interface

Variable name	Default Value	Type	Description
nviLampValue	0,0 0	SNVT_switch (95)	Control input
nvoLampValueFb	0,0 0	SNVT_switch (95)	Lamp actuator feed back

#### 1.3.1.1.1 nviLampValue

This variable is the control input. Permits other devices to control the Lamp Actuator level and state.

#### 1.3.1.1.2 nvoLampValueFb

Lamp feedback output variable. State of the Lamp Actuator (ON or OFF) and the percentage level of intensity.

### 1.3.1.2 Configuration Properties

Configuration	Default Value	Description
SCPTInvertOut (16)	ST_OFF	Setting if output is inverted

#### 1.3.1.2.1 SCPTInvertOut

This configuration property set the parameter, if the lamp output is inverted or not. This possibility is used when an external relay is used and the electrical installation shall be fail-safe.

SCPTInvertOut	Description
ST_OFF	Output not inverted
ST_ON	Output inverted

### 1.3.1.3 Lamp object typically use

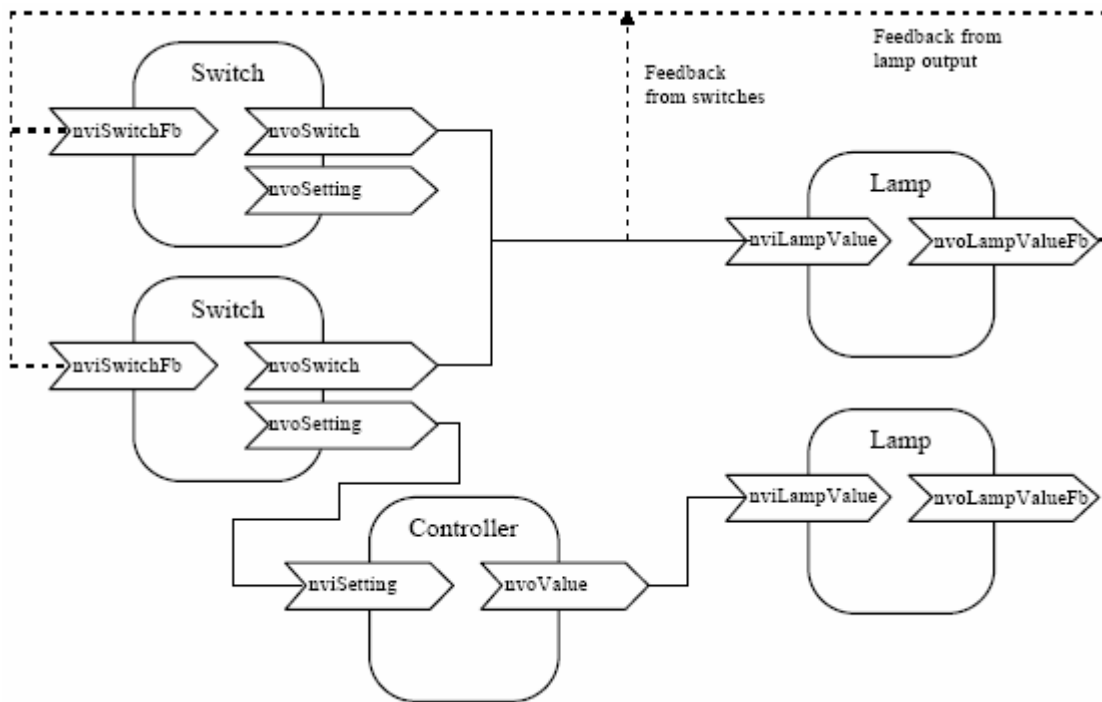


Figure 11 Typically use of the lamp actuator object

### 1.3.2 Switch (3200), 2 objects

The following describes the function for the switch input object. This object is used with a hardware switch. The switch object is used to detect if a hardware switch is shorted or open and hereafter to operate an actuator. The output from the switch object is typically connected to a lamp actuator object or other objects using at switch value for ON of OFF purposes.

Switch #	Connector #	Name in LonMaker list
1	I1	Switch[0]
2	I3	Switch[1]

Figure 12 The switch hardware/software relation

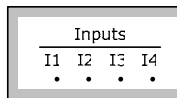
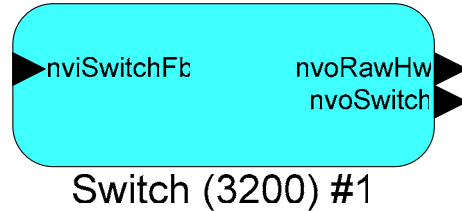


Figure 13 The switch hardware input



Switch (3200) #1

Figure 14 The switch object

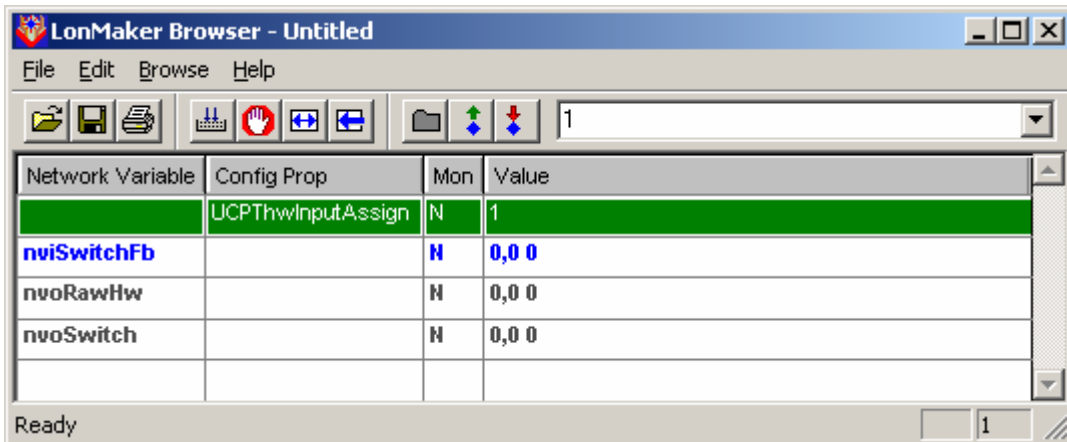


Figure 15 Browsing the switch object

### 1.3.2.1 Network Variable Interface

Variable name	Default Value	Type	Description
nviSwitchFb	0,0 0	SNVT_switch (95)	Actuator feed back
nvoSwitch	0,0 0	SNVT_switch (95)	Control output
nvoRawHw	0,0 0	SNVT_switch (95)	Raw hardware status

#### 1.3.2.1.1 nviSwitchFb

This variable is the feed back input on the object, giving the object a feed back from the device of object it is operating.

#### 1.3.2.1.2 nvoSwitch

This variable is the output from the object and is used to operate other object. The function on the switch object is the toggle type, meaning the every time the hardware input to the object has been shorted, the switch out variable will change between 100% state 1 (ON) and 0% state 0 (OFF).

#### 1.3.2.1.3 nvoRawHw

This variable is the raw output from the object and is giving the true status from the hardware input. If the hardware input is shorted the value on the variable is 100% state 1 (ON) and if the input is open the variable value is 0% state 0 (OFF).

### 1.3.2.2 Configuration Properties

Configuration	Default Value	Description
UCPThwInputAssign	1 or 3	Assigning hardware input



### 1.3.2.2.1 UCPThwInputAssign

The configuration property UCPThwInputAssign, is shorted from hardware input assignment. The property is used to make the relation between the input hardware and the software object. The 4 hardware input on the PLS2024 are electrically identical to each other and you can select which hardware input you are using in your software object.

The default hardware assignment to the software objects:

<b>Input</b>	<b>Connector #</b>	<b>Default Assignment#</b>	<b>Object Name</b>	<b>Function</b>
<i>1</i>	<i>I1</i>	<i>1</i>	<i>Switch[0]</i>	<i>Switch</i>
2	I2	2	OccupSensor[0]	PIR
<i>3</i>	<i>I3</i>	<i>3</i>	<i>Switch[1]</i>	<i>Switch</i>
4	I4	4	OccupSensor[1]	PIR

Possibilities for the UCPThwInputAssign property in the software object:

<b>UCPThwInputAssign</b>	
<b>Value</b>	<b>Description</b>
1	Using input 1
2	Using input 2
3	Using input 3
4	Using input 4



## 2 Software History

### 2.1 *Lonbox PLS2024 Version 3.00*

-----

Added missing functionality for setting commands to occupancy controllers. It does now correctly turn off the lamp when receiving SET\_OFF, SET\_ON cancels manual override and other functions are forwarded to the setting output.

Removed nvoEffectOccup output variable from occupancy controllers.

### 2.2 *Lonbox PLS2024 Version 2.02*

-----

New configuration property for occupancy sensor function that support inverted input making it possible to use "active open" security style PIR sensors. Changed configuration properties to be in a LonMark configuration file instead of in configuration network variables.

### 2.3 *Lonbox PLS2024 Version 2.01*

-----

Added nvoEffectOccup output variable to occupancy controllers. The output from the controller is heartbeated with a fixed 120 second interval. The heartbeat is active for nvoLampValue and nvoEffectOccup.

### 2.4 *Lonbox PLS2024 Version 2.00*

-----

Same as version 1.20, this is released to prevent problems with self documentation in version 1.20 saying that it is actually 2.00.

### 2.5 *Lonbox PLS2024 Version 1.20*

-----

This release is compatible with the Lonbox resource files.

This release is not compatible with version 1.03.

The internal control of relays has been modified to save power and allow for higher current usage on the 12Vdc outlet.

A New configuration is implemented for the 4 inputs. It is now possible to select which hardware input that is used for each sensor functional block. The configuration is implemented on each sensor functional block.



## **2.6 Lonbox PLS2024 Version 1.03**

-----  
This release is compatible with version 1.02 and uses same program id.

The control of relays have been modified to save power and allow for higher current usage on the 12Vdc outlet.

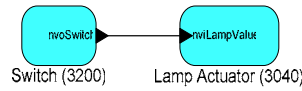
## **2.7 Lonbox PLS2024 Version 1.02**

Initial release.

### 3 Applications and bindings

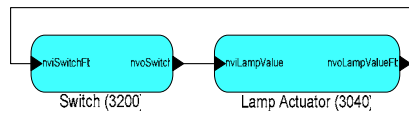
#### 3.1 Manual Control, Light Bindings

##### 3.1.1 Single on and off.



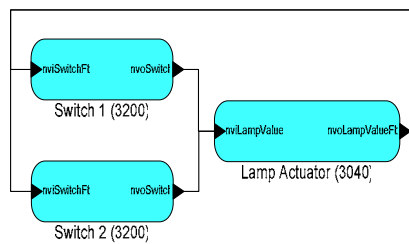
Figur 16 Single on and off

##### 3.1.2 Single on and off with feed back.



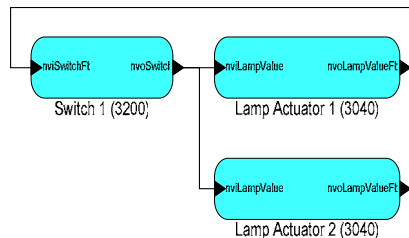
Figur 17 Single on and off with feed back

##### 3.1.3 Multi on and off with feed back.



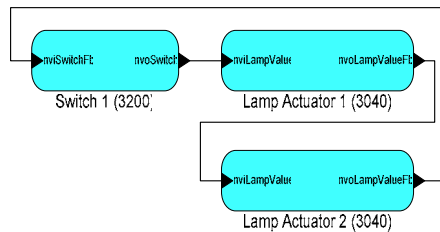
Figur 18 Multi on and off with feed back

##### 3.1.4 Single on and off with multi actuators 1.



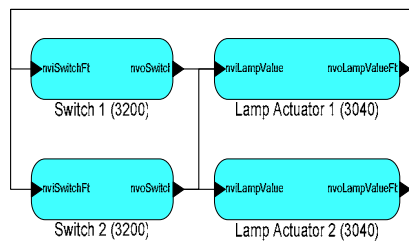
Figur 19 Single on and off with multi actuators

### 3.1.5 Single on and off with multi actuators 2.



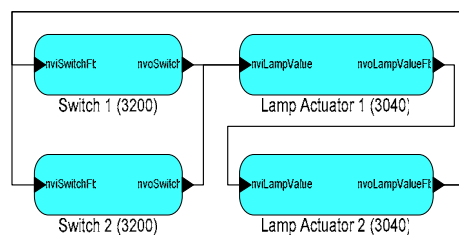
Figur 20 Single on and off with multi actuators, 2

### 3.1.6 Multi on and off with multi actuators 1.



Figur 21 Multi on and off with multi actuators, 1

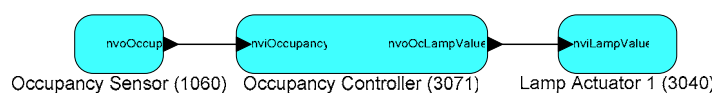
### 3.1.7 Multi on and off with multi actuators 2.



Figur 22 Multi on and off with multi actuators, 2

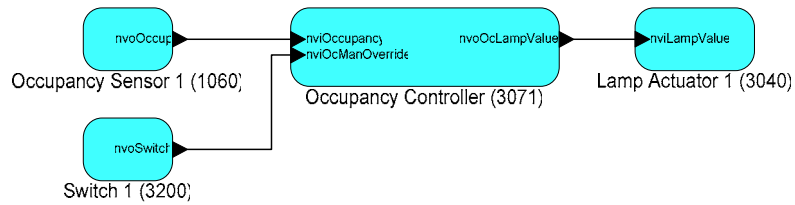
## 3.2 Occupancy automatic, Light Bindings

### 3.2.1 Occupancy automatic on off.



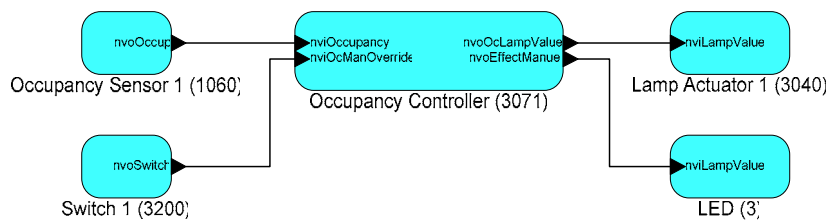
Figur 23 Occupancy automatic on off

### 3.2.2 Occupancy on off, with manual override.



**Figur 24 Occupancy automatic on off, with manual override**

### 3.2.3 Occupancy on off, manual override and manual control indicator.



**Figur 25 Occupancy automatic on off, manual override and manual control indicator**

## 4 Figure list

Figure 1 The main application for PLS2024	3
Figure 2 Typically use of the Lonbox PLS2024 software	4
Figure 3 The occupancy hardware input	5
Figure 4 The occupancy sensor object	5
Figure 5 Browsing the occupancy sensor object	6
Figure 6 Occupancy timing overview	7
Figure 7 TheOccupancy controller object	9
Figure 8 Browsing the occupancy controller object	9
Figure 9 The lamp actuator object	12
Figure 10 Browsing the lamp actuator object	13
Figure 11 Typically use of the lamp actuator object	14
Figure 12 The switch hardware/software relation	15
Figure 13 The switch hardware input	15
Figure 14 The switch object	15
Figure 15 Browsing the switch object	15
Figure 16 Single on and off	20
Figure 17 Single on and off with feed back	20
Figure 18 Multi on and off with feed back	20
Figure 19 Single on and off with multi actuators	20
Figure 20 Single on and off with multi actuators, 2	21
Figure 21 Multi on and off with multi actuators, 1	21
Figure 22 Multi on and off with multi actuators, 2	21
Figure 23 Occupancy automatic on off	21
Figure 24 Occupancy automatic on off, with manual override	22
Figure 25 Occupancy automatic on off, manual override and manual control indicator	22