

Latching Relay

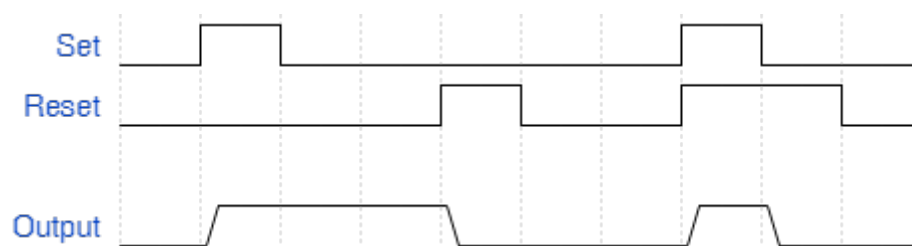


The **Latching Relay** node sends **1** to the output when the **Set** input is set to **1** and sends **0** to the output when the **Reset** input is set to **1**.

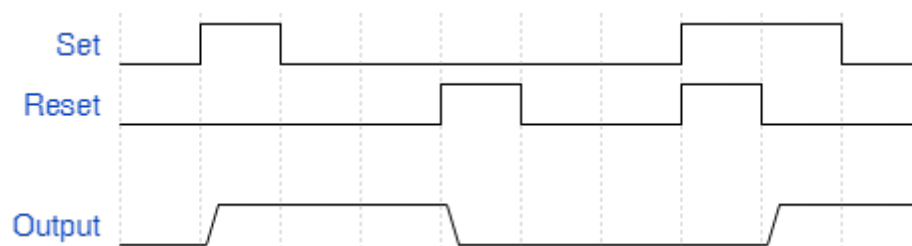
When both **Set** and **Reset** are equal to **0**, the previous state of the output is maintained.

The **Behaviour** property determines which input must be dominant when both **Set** and **Reset** are equal to **1**.

Set	Reset	Output
0	0	prev. status
0	1	0
1	0	1
1	1	Set or Reset



Behavior of node when Set input is dominant



Behavior of node when Reset input is dominant

Connectors

Connector	Type	Description
Set	Input	Sets the output to 1 when activated.
Reset	Input	Resets the output to 0 when activated.
Output	Output	Current latched state of the node.

Main properties

Property	Description
Behaviour	Defines which input is dominant when Set and Reset are both equal to 1 .

Operation

The Latching Relay node stores its output state until a new command changes it.

Its basic behaviour is:

- if **Set** = **1** and **Reset** = **0**, the output becomes **1**;
- if **Set** = **0** and **Reset** = **1**, the output becomes **0**;
- if **Set** = **0** and **Reset** = **0**, the previous output state is maintained;
- if **Set** = **1** and **Reset** = **1**, the output depends on the configured **Behaviour** property.

This means that the node acts as a memory element.



When both inputs are equal to 0, the output does not change.
The node keeps the last valid state that was previously set or reset.

Truth table

The behaviour of the Latching Relay depends on the values of **Set** and **Reset**.

When only one of the two inputs is active, the behaviour is always the same:

Set	Reset	Output
0	0	Maintains previous state
1	0	1
0	1	0
1	1	Depends on Behaviour

The last case depends on whether **Set** or **Reset** is configured as dominant.

Set dominant

If the **Behaviour** property is configured so that **Set** is dominant, when both inputs are equal to 1 the output becomes:

1

The complete logic is therefore:

Set	Reset	Output
0	0	Maintains previous state
1	0	1
0	1	0
1	1	1

This configuration gives priority to activation.

Reset dominant

If the **Behaviour** property is configured so that **Reset** is dominant, when both inputs are equal to 1 the output becomes:

0

The complete logic is therefore:

Set	Reset	Output
0	0	Maintains previous state
1	0	1
0	1	0
1	1	0

This configuration gives priority to deactivation.

Example

Assume the following sequence:

Event	Set	Reset	Output
Initial condition	0	0	0
Activation command	1	0	1
Inputs return idle	0	0	1
Reset command	0	1	0
Inputs return idle	0	0	0

The example shows that the node keeps its state even when both inputs return to 0.

Example - Set dominant

Assume that the node is configured with:

Behaviour = Set dominant

and the current output state is 0.

If both **Set** and **Reset** become 1 at the same time, the resulting output is:

1

because the Set input has priority.

Example - Reset dominant

Assume that the node is configured with:

Behaviour = Reset dominant

and the current output state is 1.

If both **Set** and **Reset** become 1 at the same time, the resulting output is:

0

because the Reset input has priority.

Typical applications

The Latching Relay node is useful whenever a command must be memorized until a separate command changes it.

Typical applications include:

- maintaining the ON/OFF state of a device;
- storing an enable or disable condition;
- managing start and stop commands;
- implementing alarm memory behaviour;
- creating bistable logic controls;
- separating activation and reset commands.

Verification through simulation

The Latching Relay node can be tested using the Logic Module simulation tools.

A typical test procedure is the following:

- start with **Set = 0** and **Reset = 0**;
- verify the initial value of **Output**;
- set **Set = 1** and verify that **Output = 1**;
- return **Set** to 0 and verify that the output remains at 1;
- set **Reset = 1** and verify that **Output = 0**;
- return **Reset** to 0 and verify that the output remains at 0;
- set both **Set = 1** and **Reset = 1**;

- verify that the output follows the configured **Behaviour** property.

Configuration notes

- Use the Latching Relay when the output must preserve its last state.
- The output is not forced to 0 when both inputs are 0; instead, the previous state is maintained.
- Configure **Behaviour** carefully according to whether activation or deactivation must have priority when both commands are present simultaneously.
- Use **Set dominant** when activation must prevail.
- Use **Reset dominant** when deactivation must prevail.
- The node works with digital logical values.

Related pages

- [Logic Module](#)
- [Logic Module - Simulation management](#)

From:

<http://www.thinknx.com/wiki/> - **Learning Thinknx**

Permanent link:

http://www.thinknx.com/wiki/doku.php?id=lm_logic_lrelay

Last update: **2026/09/21 16:01**

