



# Logical states

## Exercises Digital Design

### Solution vs. Hints:



While not every response provided herein constitutes a comprehensive solution, some serve as helpful hints intended to guide you toward discovering the solution independently. In certain instances, only a portion of the solution is presented.

## 1 | LST - Single-state logic gates

### 1.1 Switch Circuits

$$y = \bar{a}$$

*lst/one-state-01-01*

### 1.2 Switch Circuits

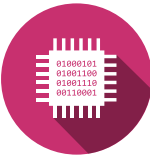
$$y = \overline{ab}$$

*lst/one-state-01-02*

### 1.3 Switch Circuits

Multiple possible solution. Minimal with 2 interruptors n-type and 2 interruptors p-type.

*lst/one-state-01-03*



### 1.4 Open-Source Circuit

Missing pull-down resistor on y.

<i>a</i>	<i>b</i>	<i>c</i>	<i>y</i>
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

*lst/one-state-02-01*

### 1.5 Alarm Circuit

Open-Drain and Open-Source Circuit possible.

*lst/one-state-02-02*

### 1.6 Collision Detection

You need either Open-Source or Open-Drain as well as a comparator. The priority depends on the Open-X you've chosen.

*lst/one-state-02-03*

### 1.7 Transmission of information on a single wire

Possible with Open-Drain and Open-Source. Only one information can be transmitted at a time.

$$\begin{cases} D_a = '1' \text{ if } a = '0' \text{ and } b = '1' \\ D_b = '1' \text{ if } a = '1' \text{ and } b = '0' \end{cases} \tag{1}$$

*lst/one-state-02-04*



## 2 | LST - Logic gates with high impedance output

### 2.1 Series linking of peripheral modules

Can be done with the help of a DEMUX-1to4.

*lst/hiz-01*

### 2.2 Creation of a function Tristate circuits

Two tri-state buffers are needed.

*lst/hiz-02*

### 2.3 Creation of a function with the help of tristate circuits

You need one tri-state inverter.

*lst/hiz-03*

### 2.4 Collision Detection

You need a comparator and a buffer with tristate functionality. Only one component is enabled at any given time.

*lst/hiz-04*

### 2.5 Register with bidirectional data bus

The register is written if **wr = '1' and cs = '1'** and the register is read if **rd = '1' and cs = '1'**. For the chip select get inspired by the exercises *lst\_hiz-01*.

*lst/hiz-05*