Exercises on state automata

Exercise 1

The simplified logic of a lift can be described as follows. The lift is waiting with the sliding door open (waiting state). When it receives a request to move to another floor, it starts closing the sliding door. If, during this operation, an impulse is received from the photocell located at the sliding door, for security reasons the sliding door is opened, and the lift returns in the waiting state. Otherwise, the lift moves according to the request. As soon as the destination floor is reached, the sliding door is opened, and then the lift is put in the waiting state. At initialization, the lift is in the waiting state.

1. Model the logic of the lift through a state automaton $(\mathcal{E}, \mathcal{X}, \Gamma, f, x_0)$.

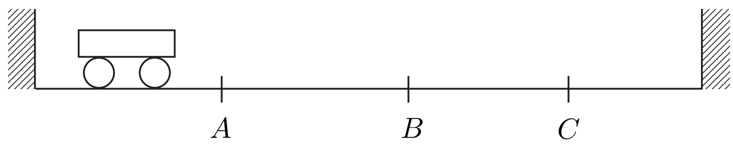
Exercise 2

A zero-order holder transforms an asynchronous sequence of bits 0 and 1 in a continuous-time binary signal $y(t)$. The zero-order holder implements the logical implication of the last two bits received. At initialization, assume that an indefinite sequence of bits 1 has been received.

1. Model the logic of the zero-order holder through a state automaton $(\mathcal{E}, \mathcal{X}, \Gamma, f, x_0)$.

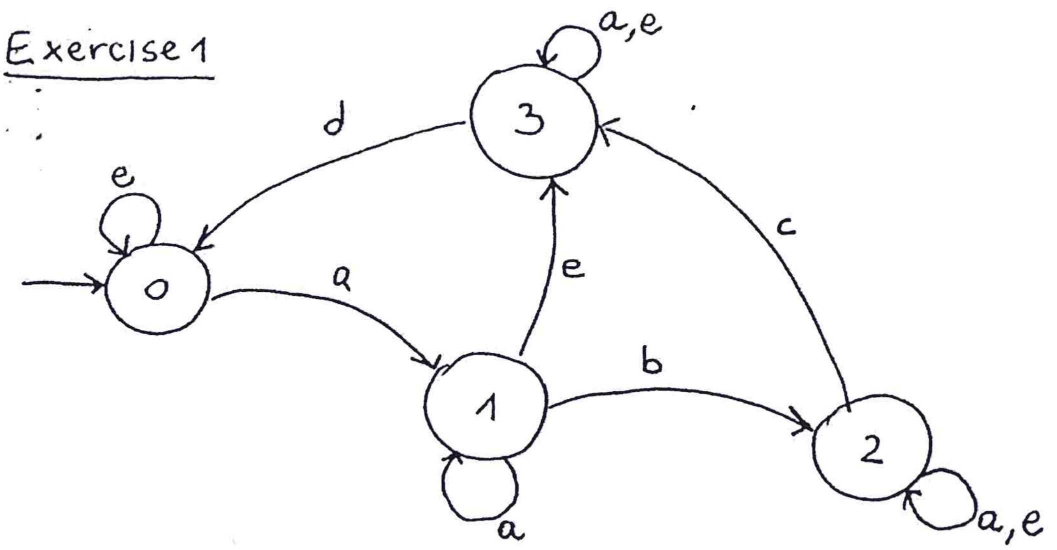
Exercise 3

A cart moves along a track. Sensors are located at three points on the track (they are denoted A, B and C in the figure). Each sensor sends an impulse when the cart crosses the corresponding point, in both directions. For the sake of simplicity, it is assumed that the cart never changes direction when it is across a sensor.



1. Provide a logical model of the cart position along the track.
2. Model a monitoring system which localizes the cart over the track, and detects possible failures of the sensors, by using only the signals it receives from the sensors.

Exercise 1



events:

a: arrival of a request

b: door closed

c: destination floor reached

d: door opened

e: impulse from photocell

states:

0: still with open door (waiting state) $\leftarrow$ initial state

1: door closing

2: moving to destination floor

3: door opening

$\Rightarrow$ state automaton $(\mathcal{E}, \mathcal{X}, \Gamma, f, x_0)$

$\mathcal{E} = \{a, b, c, d, e\}$

$\mathcal{X} = \{0, 1, 2, 3\}$

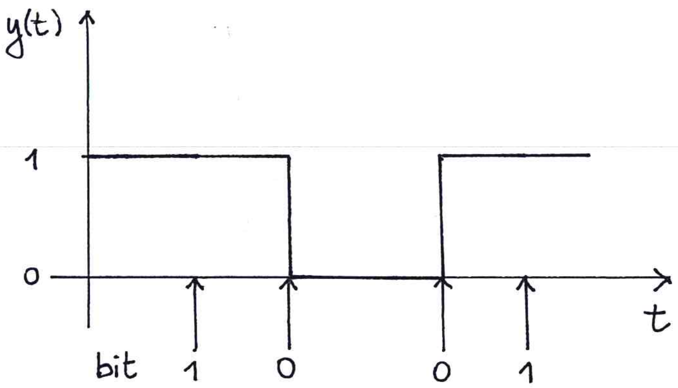
Γ, f, x_0: implicitly defined by the state transition diagram

Exercise 2

Truth table of the logical implication $(\rightarrow)$:

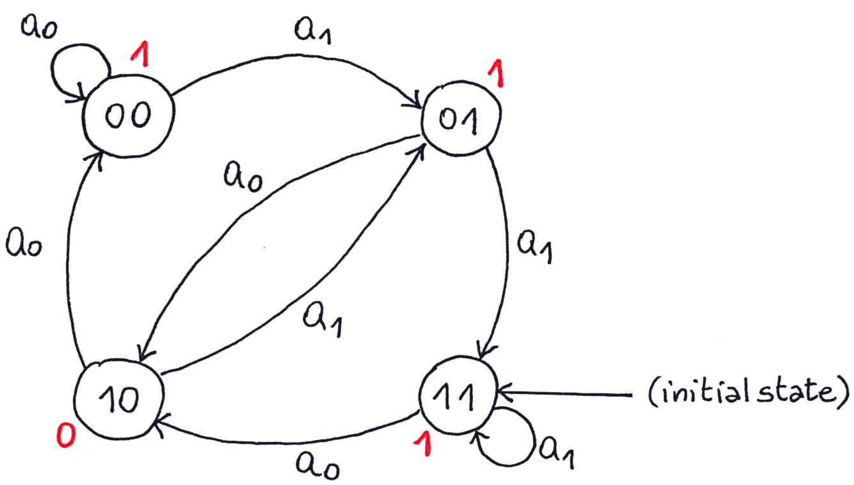
x_1	x_2	$x_1 \rightarrow x_2$
0	0	1
0	1	1
1	0	0
1	1	1

example of how the zero-order holder works



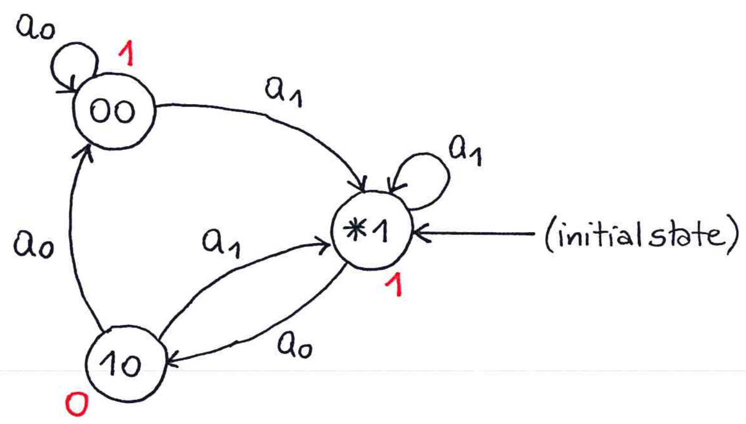
events $\mathcal{E} = \{ \underbrace{a_0}_{\text{arrival of bit 0}}, \underbrace{a_1}_{\text{arrival of bit 1}} \}$

state $x = \underbrace{x_1}_{2^{nd}\text{ last bit}} \underbrace{x_2}_{\text{last bit}} \in \{00, 01, 10, 11\}$ output: $y = x_1 \rightarrow x_2$



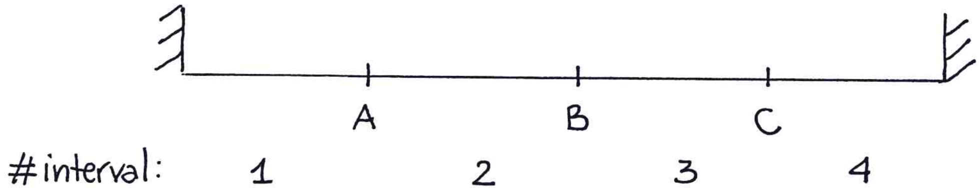
Noticing that the output is always 1 when the last bit is 1, the number of states can be reduced to three:

$$x = \begin{cases} *1 & \text{the last bit is 1} \\ 00 & \\ 10 & \end{cases} \quad \text{as before}$$



EXERCISE 3

1. The cart position is discretized:



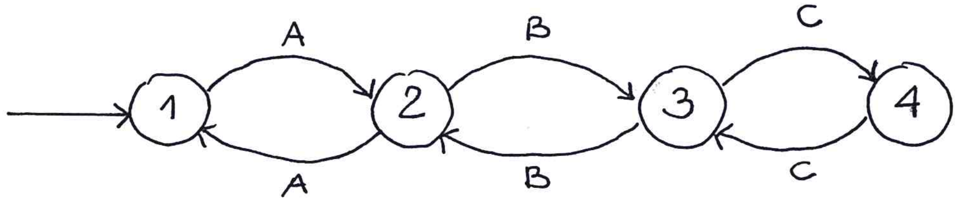
state $x = \#\text{interval} \in \underbrace{\{1,2,3,4\}}_{X}$

events $\mathcal{E} = \{A, B, C\}$

A: impulse from sensor A

B: impulse from sensor B

C: impulse from sensor C



2. We add a dummy state collecting possible failure situations:

state 5: failure

We also define the following output:

$$y = \begin{cases} 0 & \text{if a failure has been detected} \\ 1 & \text{otherwise} \end{cases}$$

Resulting model: