

## **Problem 1 Bridge Crossing:**

In this model it finds the shortest time plan in 17 minutes. By checking several values starting from 30 it's the last time where the model provides a counter example. So, when 16 minutes is given as time it takes too long to sldv to prove the property. As the model is constructed in a bit complicated way that's why sldv took long time for 16 minutes.

**Model Construction:** In the model there are several states for implementation of the rules. There are 4 input data and ok is one output data which is compared to time and sent to property proving. The logic of the model is every state it can take two people and one can go alone, and one can go back to initial state alone. Also, while passing from one state to goal it takes the slowest time. So, when everyone reached other side from goal state they go to complete state where ok=1.

Note: Due to shortage of time the model is implemented in a complex way and looks a bit ugly.

## **Problem 2 Requirements for Elevator**

**a) R1:** To implement R1 it takes position as input signal and compares the position among them. It also takes the door and checks with position, if they are true then it fulfills the requirement that door will not be open when it's not at some floor.

**R2:** To model the second requirement a detector block is used with a synchronized value of 5-time steps which takes door signal as input to check doors are open for 5 seconds when they are open.

**R3:** To implement this requirement lamp signal for 4 floors is used which are connected to an extender block that acts as a button press here and is compared with position of the elevator. Extender block with inf value is used as this controller just shows the lamp for a small time.

**R4:** This is just the same as R3 just a small change in extender block with 5-time steps.

**R5:** This requirement is checked for floor 1 and 2 here. So, when floor 1 is pressed after that if floor 2 is pressed the logic checks that elevator stops at floor 2 once before going to floor 1. Two extender blocks are used here. One extender block is resetting itself when it reaches to floor 2. So that it can be compared with other one where it takes both floor 1 and floor 2 (lamp) as input.

**R6:** To implement this a small statechart is used in which a matlab function is used inside the chart where it finds the value for d, then it compares with given value of d.

**b)** For every requirements property proving Sldv runs forever. This model ran for 10.5 hours straight and doesn't show any result. The reason for not giving any result is the main model is having different logic with one matlab function also inside the state chart there are some logic. That could make the model a complex one.