Assignment 3.2

Submission Instructions

Read the questions carefully and create a PDF document with solutions. Put code and output in this document, wherever necessary for proper evaluation. Upload the final document on Moodle.

Question 1. A monitor node for a property S takes as input specified set of flows to observe (e.g. P,Q:bool). It outputs a single boolean flow OK. The idea is that at every cycle OK is true if the property S holds for the past sequence of inputs (including the current cycle). For example

Property S: "P is continuously true in the past"
has the monitor node

```
node SMONITOR(P:bool) returns (OK:bool)
let
  OK = P -> (pre(OK) and P)
tel
```

Answer the following:

1. Give the output of the above SMONITOR for the input flow
   P=true true true false true false true true
2. Give a monitor node for the property:
   "Everytime A occurs C will remain true until a B occurs".
   Specify additional assumptions that you make in your design (E.g. what happens if A and B occur simultaneously).
3. Give a monitor node for the property:
   "If REQ has been true for the last 3 cycles (including the current cycle) then ACK must be currently true."

Question 2:
Model a Gas Burner Controller as a Heptagon node
node CONTROLLER(flame: bool; sec: bool) returns (gas: bool; spark: bool)
to meet the following requirements. Preferably Use the automaton construct.

"Controller keeps gas on/off using the output gas and strikes a flame using output spark. It can detect whether flame is on/off using input flame. Flame will not occur unless the gas has accumulated for at least 15 seconds. Not every spark results into flame. Flame also occasionally goes off due to wind. Making a spark after Gas has leaked for more than 1 minute causes an explosion. After a leakage longer than 1 Minute, the Gas must be allowed to dissipate for 2 minutes to reach a safe state. The controller should keep the flame on as much as possible."

Please explain your design. Simulate your code for various sample inputs and submit the output produced.