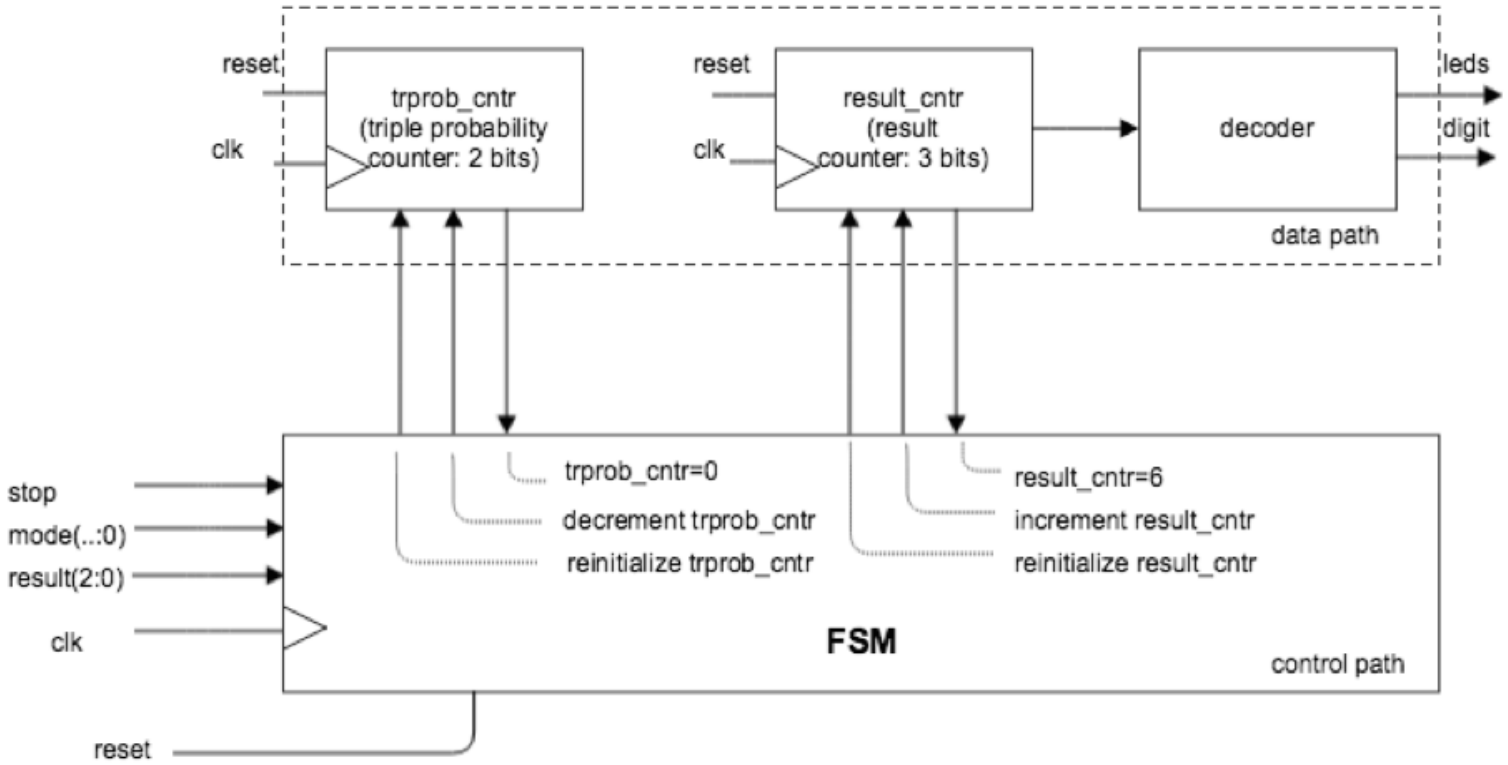
W02 Jan 15 (D2) The cheating e-dice as a FSMD

Jose Ferreira

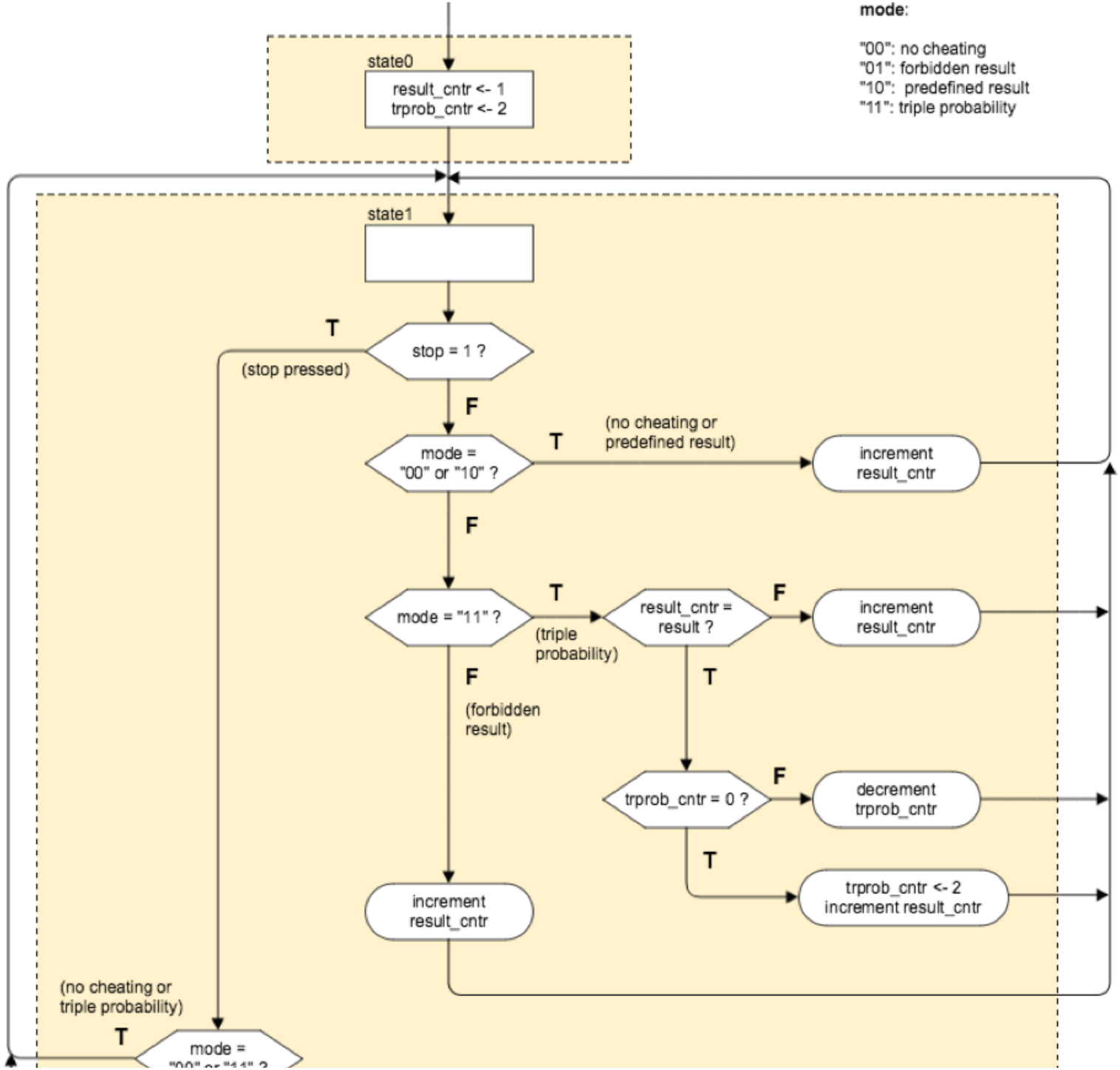
These questions are presented under the following assumptions:

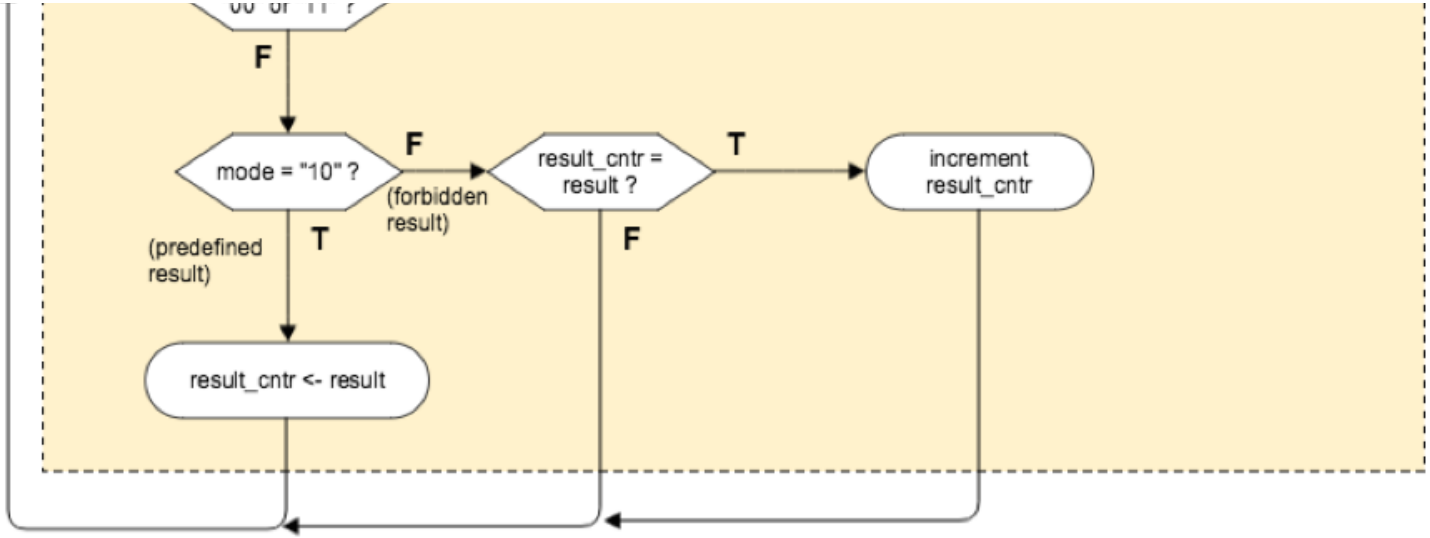
- They may be selected to be part of the final exam
- Responses must be posted by the students (not me)
- I will call your attention to any mistakes or wrong content posted in response

The block diagram represented below corresponds to a FSMD architecture that implements the functional requirements of a cheating electronic dice.



1. Assume that only two operating modes are supported: no cheating, and triple probability for the result specified by inputs “result(2:0)”. Present an ASMD chart that illustrates the behaviour of this FSMD. **N.B.**: In this case, “mode” is a single input that defines the no-cheating mode when at ‘0’, and the triple probability mode when at ‘1’.
2. Make all appropriate corrections and / or simplifications to the ASMD chart represented below, which is supposed to illustrate the behaviour of the architecture represented above, when supporting four operating modes (“00”: no-cheating; “01”: forbidden result; “10”: predefined result; “11”: triple probability).
3. Build the corresponding VHDL description and prove that your solution works by showing simulation results in Vivado.


mode:
"00": no cheating
"01": forbidden result
"10": predefined result
"11": triple probability
state0
result_cntr <- 1
trprob_cntr <- 2
state1
stop = 1 ?
T
(stop pressed)
F
mode = "00" or "10" ?
T
(no cheating or predefined result)
increment result_cntr
F
mode = "11" ?
T
(triple probability)
result_cntr = result ?
F
increment result_cntr
T
F
(forbidden result)
trprob_cntr = 0 ?
F
decrement trprob_cntr
T
trprob_cntr <- 2 increment result_cntr
increment result_cntr
(no cheating or triple probability)
T
mode =



Search entries or author | Unread | Subscribe

Reply

Deivydas Kazokas

Hey, here is my solution to this discussion. L…

2 3