Lab session #11

Activity #1: 2-request arbiter
Consider the specifications for activity #2 in Lab. 7.

Two subsystems may access the same resource. An **arbiter** is a circuit that resolves any conflict and coordinates the access to the shared resource. The arbiter has 2 inputs *req0* and *req1* coming from the 2 subsystems and generates 2 outputs *grant0* and *grant1* that are inputs to the 2 subsystems. When a subsystem needs the resource, it activates its request signal. The arbiter monitors the use of the resource and the requests: if this is the only request, the arbiter grants access to the subsystem by activating the corresponding grant signal. Once its grant signal is activated, a subsystem has permission to access the resource. Any request coming from the other subsystem is ignored. After the task has been completed, the subsystem releases the resource and deactivates the request signal. If both subsystems issue a simultaneous request, 2 strategies are possible:

1. One of the subsystems has priority, e.g., subsystem 0
2. The arbiter keeps track of which subsystem had the resource last time and gives priority to the other subsystem.

For strategy #1 design 2 versions of the Moore FSM in Verilog using:

- 1 always block
- 2 always blocks.

Reset is synchronous. Create a suitable testbench to test the arbiter on relevant values.