Finite State Machine & Memories

Design the following circuits using Verilog **and create a testbench** for each design to check its functionality

1)

Requirements:

1- Design Moore FSM that detects "010" pattern. (Draw state transition diagram)
2- Store how many time the pattern was detected.

Ports:

Name	Type	Size	Description
x	Input	1 bit	Input sequence
clk			Clock
rst			Active high asynchronous reset
y	Output	1 bit	Output that is HIGH when the sequence 010 is detected
count		10 bits	Outputs the number of time the pattern was detected

2) Suppose that you are working as a Digital design Engineer in Tesla Company. It is required to design a control unit using Moore FSM for Self-driving cars on highways that controls the acceleration of the car as well as the door locking mechanism with the following specifications.

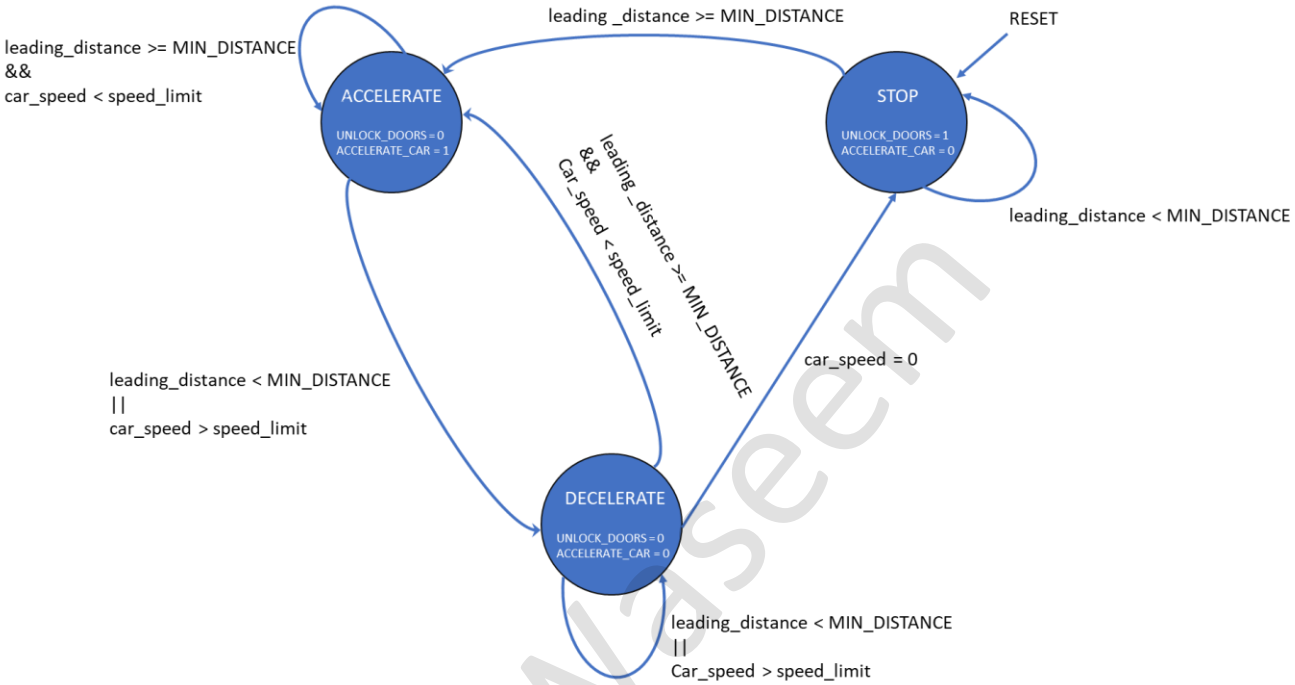
>> Consider creating realistic testbench for this design <<

Ports

Name	Type	Size	Description
speed_limit	Input	8 bits	Allowable speed limit of the highway
car_speed		8 bits	Current car speed
leading_distance		7 bits	Distance between the car and the vehicle/object in front of it
clk		1 bit	Clock
rst		1 bit	Active high asynchronous reset
unlock_doors	Output	1 bit	Signal that unlock the car doors when HIGH
accelerate_car			Signal that control the flow of the fuel to the engine to accelerate the car when HIGH

Parameters

- MIN_DISTANCE: Minimum distance between two vehicles, default = 7'd40 //40 meters

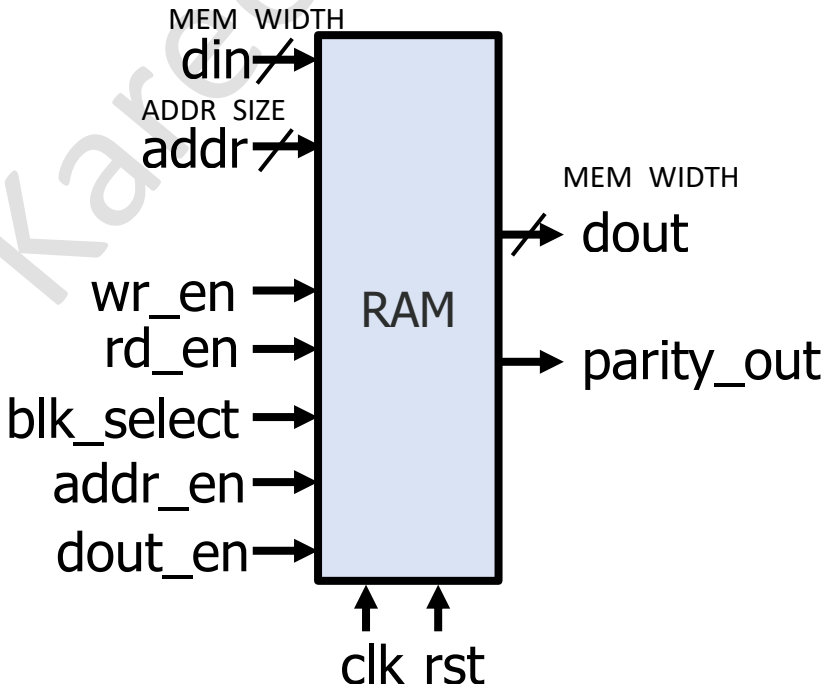


3) Implement the following single port synchronous write/read and create a testbench for it

- Parameters

Name	Description	Default values
MEM_WIDTH	Data in/out and memory word width	16
MEM_DEPTH	Memory depth	1024
ADDR_SIZE	Address size based upon the memory depth	10
ADDR_PIPELINE	If “TRUE” then the address should be pipelined before writing/reading the RAM, if ”FALSE” then the address input will be assigned directly to the RAM’s address port	FALSE
DOUT_PIPELINE	If “TRUE” then the data out should be pipelined, if ”FALSE” then the output will be out of the RAM directly	TRUE
PARITY_ENABLE	If the parameter value is 1 then the parity should be calculated and assigned to parity_out port, if the parameter is 0 then the parity_out port should be tied to 0	1

- Added ports functionality
  - addr_en: enable signal for the flipflop that pipelines the address
  - dout_en: enable signal for the flipflop that pipelines the data out
  - parity_out: calculates the parity on the dout bus



The internal pipelining done inside of the design is demonstrated by the following snippet