Finite State Machine

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

1) Suppose you're working for Vodafone, and they wish to store the mobile number of Vodafone users calling from Vodafone number "starting with 010 sequence" asking about internet service. These users are just calling to ask about the price of home internet packages so Vodafone will store their mobile numbers and will offer them discounts in the future to have Home internet service installed for free since they are Vodafone mobile number holders.

Requirements:

1- Detect that the calling number is a Vodafone number, so we need to design Moore FSM that detects "010" pattern. (Draw state transition diagram)
2- Store how many Vodafone users calling the internet service customer service
3- [BONUS] Instantiate the RAM done in the previous assignment to store all Vodafone numbers in the RAM

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
users_count		10 bits	Outputs the number of Vodafone users called the customer service till now

[BONUS PART – RAM Storage]

Assume that we receive the following sequence 0011111**010**00011100 then we shall store the 8 bits following 010 sequence which are "00011100" in the RAM and then increment the address and wait for the sequence to be detected again and so store the next 8 bits and so on

Kareem Waseem

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.

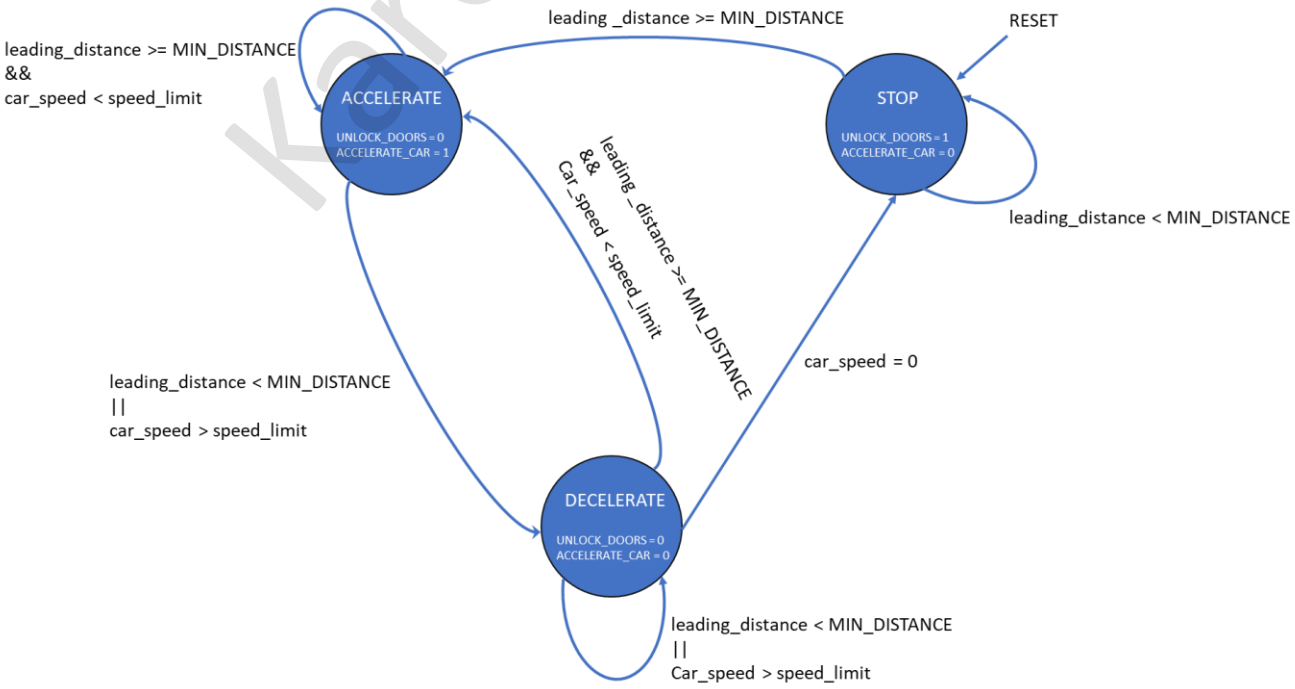
>> Consider creating realistic testbench for this design <<

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



Kareem Waseem