FSM

Q1. Kelvin is tasked with building an FSM which would detect **10 or 011 overlapping** sequences coming out of a communication channel. A sample is given below:

(w indicates input sequence values and z indicates output)

Clock cycle	t0	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10	t11	t12	t13
w	1	0	1	1	0	0	1	1	0	1	0	1	0	0
z	0	0	**1**	0	**1**	**1**	0	0	**1**	**1**	0	**1**	0	**1**

Note: Use the table below for **Gray coding** if needed.

Decimal Equivalent	**Binary Code**			**Gray Code**		
	B2	**B1**	**B0**	**G2**	**G1**	**G0**
0	0	0	0	0	0	0
1	0	0	1	0	0	1
2	0	1	0	0	1	1
3	0	1	1	0	1	0
4	1	0	0	1	1	0
5	1	0	1	1	1	1
6	1	1	0	1	0	1
7	1	1	1	1	0	0

(a)	**Determine** the type of **FSM** (Mealy/Moore) to be implemented	[1]
(b)	**Draw** the state diagram. Use no more than **6 states**. Using more states will result in deduction of marks.	[4]
(c)	**Draw** the state assignment table using **Binary encoding**.	[3]
(d)	**Determine** the expression of only the output, **z**. No need to draw circuit.	[2]

Q2. **Design** a finite state machine (FSM) for a car alarm system. The system has two binary inputs: x, indicating the lock status of the car door (**x = 1 if unlocked, x = 0 if locked**) and y, indicating the door status (**y = 1 if the door is open, y = 0 if closed**). The output z is used to control the alarm: z = 1 when the alarm is active, and z = 0 otherwise. The alarm should be triggered (z = 1) if the door is opened (y = 1) while it is still locked (x = 0) **for two consecutive clock cycles**. An example is shown below:

x	1	0	1	0	0	0	1	1	0	0	0	0	1	0
y	0	0	1	1	1	0	1	0	1	1	1	0	0	1
z	0	0	0	0	**1**	0	0	0	0	**1**	**1**	0	0	0

(a)	**Determine** the type of **FSM** that would be needed for the given problem and design the state diagram for the corresponding FSM.	[2]
(b)	**Derive** the state-assigned table using the **Gray encoding** from your state diagram in (a).	[2]
(c)	**Find** the expression for the next state and the output for the FSM using **KMAP**.	[3]
(d)	**Calculate** the number of flip-flops required for the circuit implementation and **draw** the circuit for this FSM.	[3]