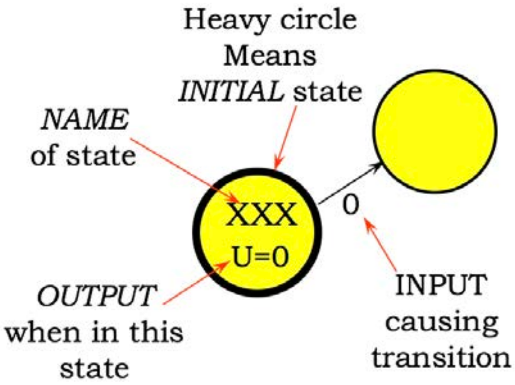
Computation Structures

Finite State Machines Worksheet

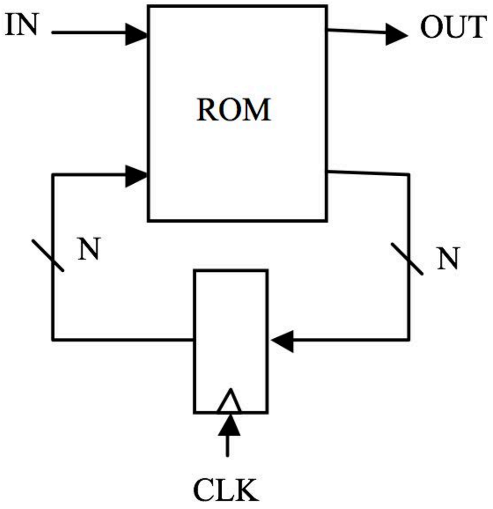
Concept Inventory:

- State transition diagrams & FSM truth tables
- Register & ROM implementation
- Equivalent FSMs; equivalent state reduction
- Metastability: causes and cures

Notes:



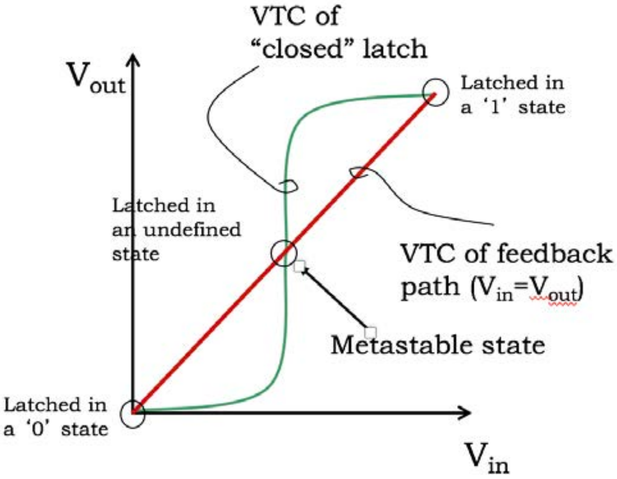
Arcs leaving a state must be (1) mutually exclusive, and (2) collectively exhaustive.



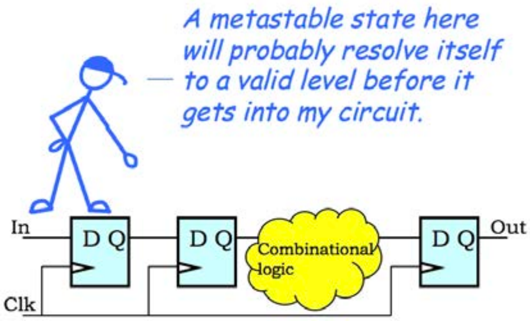
FSMs are EQUIVALENT if and only if every inputs sequence yields identical output sequences.

Two states are equivalent if
1. both states have identical outputs, AND
2. every input transitions to equivalent states.

Metastability:

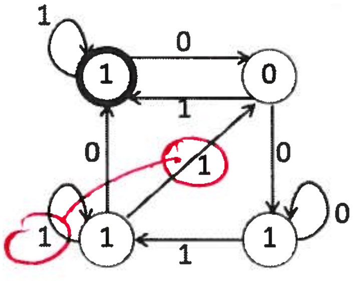


Quarantine time reduces p(metastable)

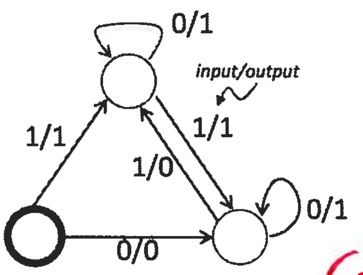


Problem 1.

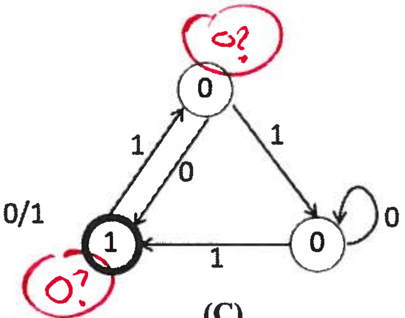
(A) For each of the following FSMs please indicate if they are or are not well formed. Note that the state names have been omitted for clarity; you may assume the state names are unique.



(A)



(B)



(C)

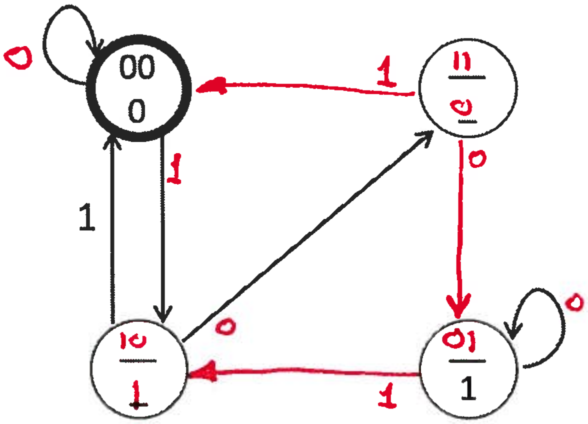
FSM A (circle one): Well Formed / [✓ Not Well Formed]

FSM B (circle one): [✓ Well Formed] / Not Well Formed

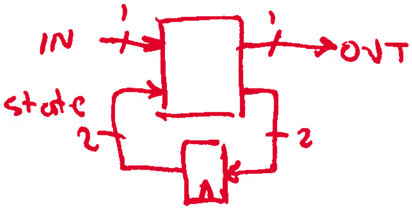
FSM C (circle one): Well Formed / [✓ Not Well Formed]

(B) Given the partially completed truth table and FSM diagram below. **Complete all the missing entries** in the truth table and the FSM diagram. The FSM is a Moore machine, i.e., the Out signal is determined only by the current state. In each state circle, the top entry is S_1S_0 and the bottom entry is the value of Out. Make sure that you have labeled all missing states, inputs, and outputs, and that you have added and labeled any missing transitions in the FSM.

S1	S0	In	S1'	S0'	Out
0	0	0	0	0	0
0	0	1	1	0	0
0	1	0	0	1	1
0	1	1	1	0	1
1	0	0	1	1	1
1	0	1	0	0	1
1	1	0	0	1	0
1	1	1	0	0	0



(C) If this FSM is implemented using a 2-bit state register and a ROM, what size ROM would be needed? Please specify the number of locations (entries) of the ROM, and the width of each entry.

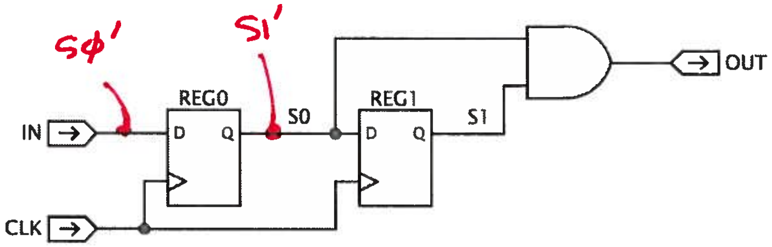


Number of locations in ROM: $2^3 = 8$

Width of each ROM entry (bits): $2+1 = 3$

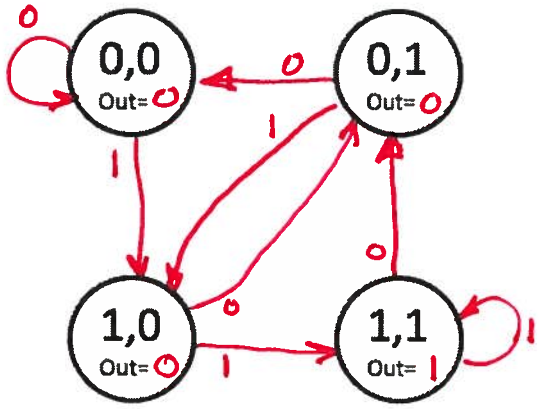
Problem 2.

Consider the sequential logic circuit to the right, which implements an FSM with a single data input IN and single data output OUT. Assume that all signal transitions are timed so that the dynamic discipline is satisfied at each register.



Please describe the operation of the FSM by filling in both the state transition diagram and the truth table shown below. The two-digit state names in the state transition diagram are S0,S1, the logic values present at the outputs of REG0 and REG1 after the rising edge of the clock. In the truth table, S0' and S1' are the values that will loaded into REG0 and REG1 at the next rising clock edge.

Fill in state transition diagram and truth table



S0	S1	IN	S0'	S1'	OUT
0	0	0	0	0	0
0	0	1	1	0	0
0	1	0	0	0	0
0	1	1	1	0	0
1	0	0	0	1	0
1	0	1	1	1	0
1	1	0	0	1	1
1	1	1	1	1	1

IN SØ

Problem 3.

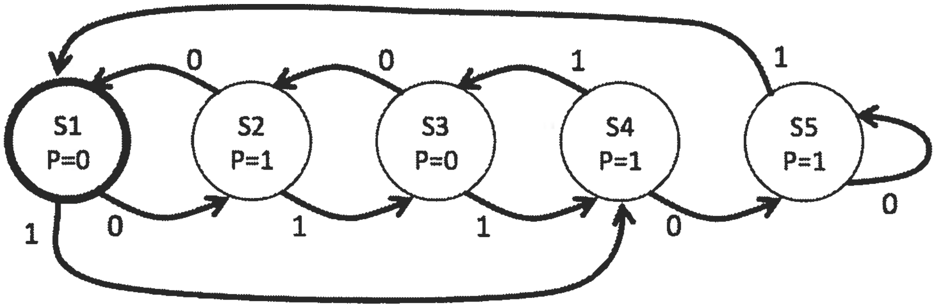
A "Thingee" is a clocked device built out of 3 interconnected components, each of which is known to be a 4-state FSM. What bound, if any, can you put on the number of states of a Thingee?

Each FSM might be in one of its 4 states.

Max # of states, or "Can't Tell": 4·4·4 = 64

Problem 4.

Consider the 1-input, 1-output finite state machine with the state transition diagram shown below. Note that the single output P only depends on the current state of the FSM.



(A) (1 Point) The FSM has been processing inputs for a while and we would like to determine its current state. After entering three additional inputs "000", we observe that we have reached a state where P=0. Please circle the possible values for the state *before* the additional three inputs were entered.

Possible values for state before: S1 [✓ S2] S3 S4 S5

(B) (2 Points) Assume that the states are represented by the 3-bit binary values given on the left below. Please fill in the appropriate entries for the partial truth table shown on the right where S is the current state, I is the input value, S' is the next state, P is the output value

State	*Encoding*
S1	001
S2	010
S3	011
S4	100
S5	101

S	*I*	*S'*	*P*
011	0	010	0
011	1	100	0
100	0	101	1
100	1	011	1

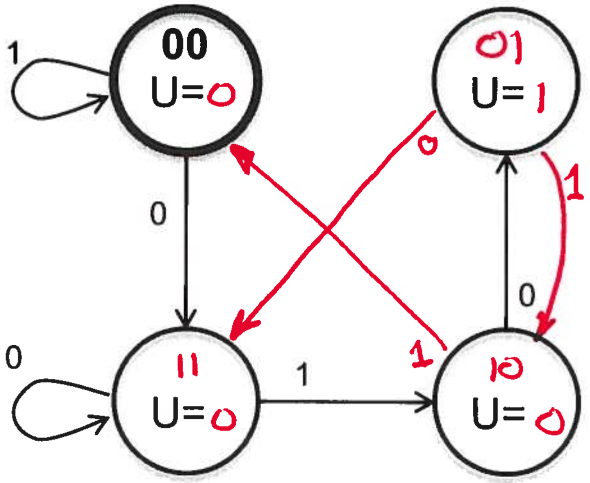
Fill in partial truth table

(C) Please identify which, if any, states are equivalent. For example, if states S1, S2, and S4 are equivalent, please write "(S1,S2,S4)". You may need multiple parenthesized lists if more than one set of states is equivalent.

Equivalent states: (S1,S3) (S4,S5)

Problem 5.

Perfectly Perplexing Padlocks makes an entry-level electronic lock, the P3b, built from an FSM with two bits of state. The P3b has two buttons ("0" and "1") that when pressed cause the FSM controlling the lock to advance to a new state. In addition to advancing the FSM, each button press is encoded on the B signal (B=0 for button "0", B=1 for button "1"). The padlock unlocks when the FSM sets the UNLOCK output signal to 1, which it does **whenever—and only whenever—the last 3 button presses correspond to the 3-digit combination.** The combination is unique, and will open the lock independently of the starting state. Unfortunately the design notes for the P3b are incomplete.



S_1	S_0	B	S'_1	S'_0	U
0	0	0	1	1	0
0	0	1	0	0	0
0	1	0	1	1	1
0	1	1	1	0	1
1	0	0	0	1	0
1	0	1	0	0	0
1	1	0	1	1	0
1	1	1	1	0	0

(A) (1 Point) What is the 3-bit combination for the lock?

lock combination: 010

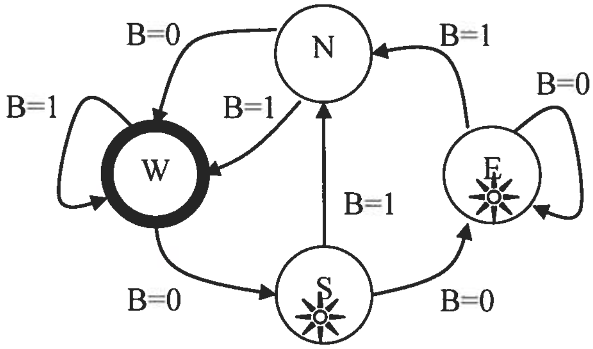
(B) (5 Points) Using the specification and clues from the partially completed diagrams above fill in the information that is missing from the state transition diagram and its accompanying truth table. When done:

- each state in the transition diagram should be assigned a 2-bit state name S_1S_0 (note that in this design the state name is *not* derived from the combination that opens the lock),
- the arcs leaving each state should be mutually exclusive and collectively exhaustive,
- the value for **U** should be specified for each state, and
- the truth table should be completed.

(complete above transition diagram and table)

Problem 6.

Below is a state transition diagram for a 4-state FSM with a single binary input B. The FSM has single output – a light that is "on" when the FSM is in states "E" or "S". The starting state, "W", is marked by the heavy circle.



(A) (1 Point) Does this FSM have a set or sets of equivalent states that can be merged to yield an equivalent FSM with fewer states?

List set(s) of states that can be merged or write NONE: S, E

(B) (5 Points) Please fill in as many entries as possible in the following truth table for the FSM. The *light* output is a function of the current state and should be 1 when the light is "on" and 0 when it's "off."

	S1	S0	B	S1'	S0'	light
E	0	0	0	0	0	1
	0	0	1	1	0	1
S	0	1	0	0	0	1
	0	1	1	1	0	1
N	1	0	0	1	1	0
	1	0	1	1	1	0
W	1	1	0	0	1	0
	1	1	1	1	1	0

LIGHT=1 implies either S or E. only S transitions to two different states.

by process of elimination

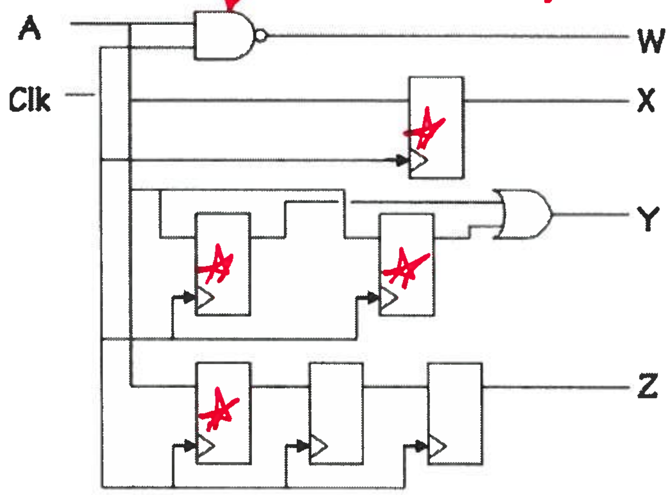
Problem 7.

The following circuit has two inputs (A, CLK) and four outputs (W, X, Y Z). The CLK signal is square wave with a period t_{CLK}=1us. The A signal makes a single 0→1 transition but the timing of the transition is close to (within a few ns of) the active CLK edge, ignoring dynamic discipline. All the devices are lenient and have the same propagation delay t_{PD}=10ns.

In a test involving a large number of trials, the **Z** output has been examined 100ns after an active CLK edge (and when *both CLK and A have been stable for many propagation delays*); at this time, **Z** was found to be invalid P times. In the same test, what would you expect to observe at the other outputs 100ns after the CLK edge? For each output, circle one of

LESS RELIABLE if you would expect the output to be **invalid** appreciably **more** than P times;
EQUALLY RELIABLE if you would expect the output to be **invalid** about P times; or
MORE RELIABLE if you would expect the output to be **invalid** appreciably **less** than P

combinational logic never goes metastable!



W reliability: LESS EQUAL [✓ MORE]

X reliability: [✓ LESS] EQUAL MORE

Y reliability: [✓ LESS] EQUAL MORE

Z was found to be invalid P times............

★ These registers are particularly susceptible to metastability since they are connected to input A.