1 Chapter Outline

1. Latches and Flip Flop

Latches and flip-flop are state element which store one bit of state. Unlike other element in previous chapters, latches and flip-flop uses memory.

In latches, there are SR latch, D latch. SR latch stores one bit of state in state variable Q or (~Q) by using two NOR gates. S represents Set and R represents Reset. When S is asserted, the output will be 1, and when R is asserted, the output will be 0. There are 4 possible binary inputs, 00, 01, 10, 11 of SR inputs. When the value of S and R is 0 or 1, the output is predictable except both value of S and R is 1. If both value of S and R is 1, the output is previous value of the output. This means that past state affects the However, SR latch has no input which controls the state. To solve this problem, D latch appeared.

D latch is a latch which has inputs control the state. D latch has two inputs which are Clock and D. Clock controls when the output changes, and D controls what the output changes to. It is implemented by using AND, NOT, and SR latch.

D flip-flop is also an element which uses memory. It controls the state when the Clock rises. When the Clock rises from 0 to 1, then D passes through Q. Otherwise, Q holds its previous value. Thus, Q changes output when the Clock rises.

By using D flip-flop, there are much complex flip-flops, which are enable flip-flop, and resettable flip-flop.

2. Synchronous Logic Design

To learn about synchronous logic design, leaning about sequential logic is required. The reason is that synchronous logic design is a solution of problematic situation in sequential logic.

First of all, sequential logic is all circuits that are not combinational. However, it can cause some problem when it is design in some sort of path. To be specific, in sequential logic design, there is a cyclic path which output fed back to the input, and makes the circuit astable and oscillate. To solve this problem, synchronous logic design is needed.

In solving the problematic circuit by using synchronous logic design, the cyclic path has to be broken by inserting registers. As a result, state changes at clock edge which means that system is synchronized to the clock because registers contain state of the system. In the process of designing synchronous logic, the rules of synchronous sequential circuit composition must be considered.

3. Finite State Machines

Finite state machine is a common synchronous sequential circuit. It is consisted of state register and combinational logic. To be specific, state register stores the current state and loads next state at clock edge. Moreover, combinational logic computes the next state and the output. By using state register and combinational logic, finite state

machines have the next step determined by current state and input.

In the difference of output logic, there are two types of FSM, Moore Machine and Mealy Machine. In case of Moore Machine, outputs only depend on the current state, but in case of Mealy Machine, outputs depend on the inputs and the current state.

After learning about Moore Machine and Mealy Machine, the transition of FSM will be handled, and the logic and the schematic of both machines will be introduced.

4. Timing of Sequential Logic

Timing of sequential logic is similar to the delay in Chapter 2. However, it is more complex than the previous chapter one because it is using much more complex element, D flip-flop. Because flip-flop samples D at clock edge and at that timing D has to be stable otherwise it can cause metastability.

Timing of sequential logic has input timing constrain, output timing constraint, and dynamic disciplines.

First of all, in input time constraint, there are setup time, hold time, and aperture time. Setup time is time before clock edge, hold time is time after clock edge, and aperture time is time around clock edge which is sum of setup time and hold time. In these time, data must be stable.

Secondly, in output time constraint, there are propagation delay and contamination delay. Propagation delay is time after clock edge which makes the output stable and contamination delay is time after clock edge which makes the output unstable.

Finally, in dynamic discipline, inputs must be stable at the period of clock edge, and delays have maximum and minimum depending on the delays of the circuit element.

Moreover, there is clock skew which is a gap between two clock edges. The role of clock skew is not to violate dynamic discipline for any register by performing worst case analysis.

5. Parallelism

In parallelism, there are two types of parallelism which are spatial parallelism and temporal parallelism. The spatial parallelism is performing multiple tasks at once by duplicating hardware, and the temporal parallelism is breaking tasks into multiple stages so called pipelining. Furthermore, there are some important definitions in parallelism: token, latency, throughput. Token is a set of inputs processed to produce the set of output, and latency is a period of one token to pass from start to the end. Throughput is a number of token produced per unit time. Parallelism increases throughput.

2 Research and Explore (Grey box side)

The first blurb is on page 123 of the textbook. According to page 123, Moore and Mealy developed automat theory. Edward F Moore his article in 1956, and George H Mealy published his article in 1955. The blurb is describing about two scholars Edward F Moore and George H Mealy, and how they had their career after publishing sequential machine and circuit.

The second blurb on page 129 of the textbook is about the story of Ben. In the story, Ben tried his best to make student have grasp about the knowledge, but student did not try to. Then the Dean suggest Ben to make catapult in order to make interaction. However, this suggestion was not that effective. The story implicitly describes about engineering students who don't want to have interaction from real world to the textbook. This may claim that engineer students need to interact with the real world.

3 Figure

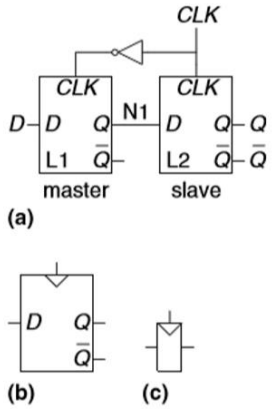


Figure 3.8 is about D flip-flop. (a) is its schematic design, (b) is the symbol of D flip-flop, and (c) is condensed symbol of D flip-flop. The reason why this figure is chosen is that D flip-flop will be integrated to much more complicated and complex flip-flop and machine such as resettable flip-flop or finite state machine. Because of this importance of D flip-flop, figure 3.8 is selected.

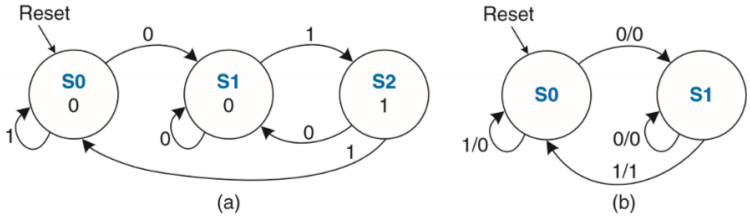


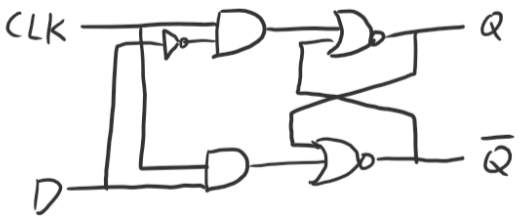
Figure 3.30 is FSM state transition diagram. (a) is Moore Machine, and (b) is Mealy Machine. To design much complex FSM, it is important how both FSM do transition. Moreover, this figure is related to encoding and the future state. Therefore, figure 3.30 is selected.

4 Example Problems

(Note: all of the problems have variation from the original problem.)

1. From EX 3.1

   How many transistors are needed to build the D latch?



   There are one NOT gate, two AND gates, and two NOR gates. NOT gate uses 2 transistors, AND gate uses 6 transistors, and NOR gate uses 4 transistors. Thus, it is 2 + 2*6 + 2*4 = 2 + 12+ 8 = 22.

   Therefore, 22 transistors are needed to build the D latch.

2. From EX 3.3

   In figure 3.16, the output of the third inverter is fed back to the first inverter. Each inverter has a propagation delay of 5ns. Determine when the second rise of X is.

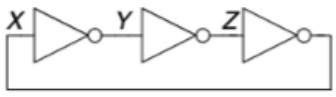
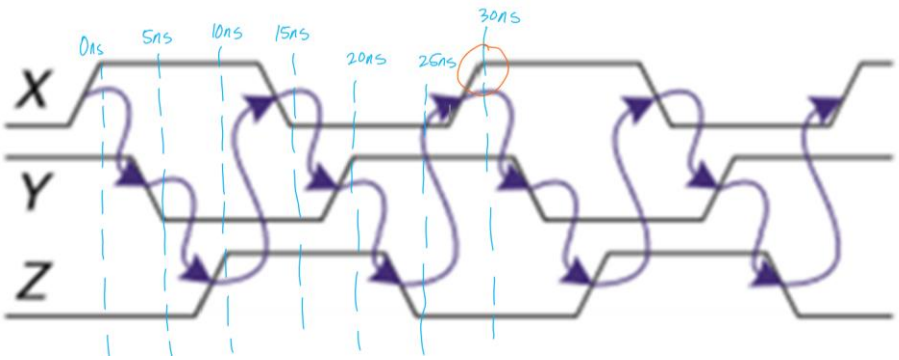


Figure 3.16 Three-inverter loop



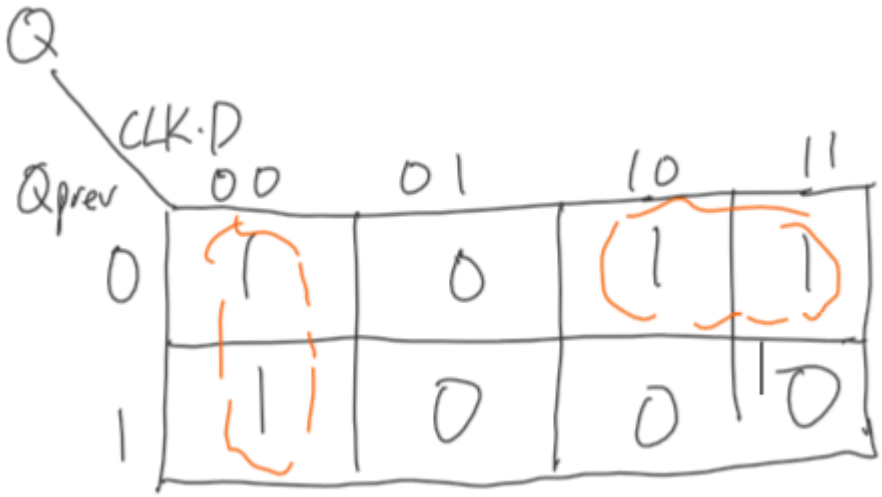
   This is a diagram is ring oscillator waveforms. Each inverter has a propagation delay of 5ns so that if we follow the diagram, the second rise of X is at 30ns.

3. From EX 3.4

Based on the truth table, implement the circuit and determine if it is synchronous or asynchronous.

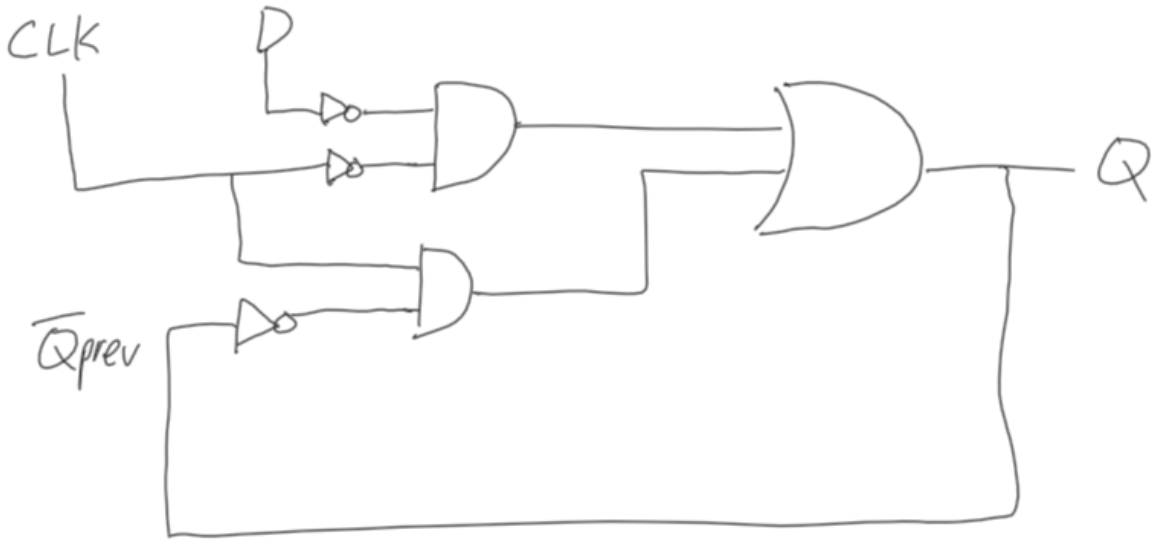
CLK	D	Qprev	Q
0	0	0	1
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0
1	0	1	1
0	1	1	0
1	1	1	1

By using K-map, we can find the logic of output.



$$Q = \overline{CLK} \cdot \overline{D}\,(Qprev + \overline{Qprev})$$
$$+ \overline{Qprev} \cdot (CLK \cdot \overline{D} + CLK \cdot D)$$
$$= \overline{CLK} \cdot \overline{D} + \overline{Qprev} \cdot CLK$$

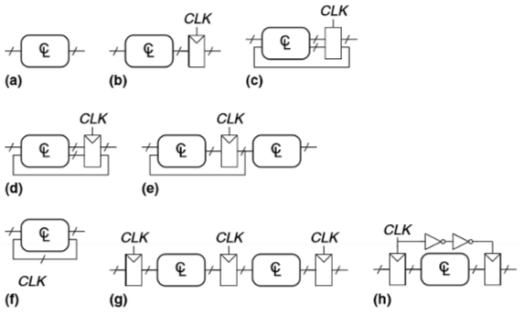
We can implement the circuit from the result of K-map.



This circuit is asynchronous circuit. The reason is that the output is negatively fed back to the input and there' s a delay which cause a race condition.

4. From EX 3.5

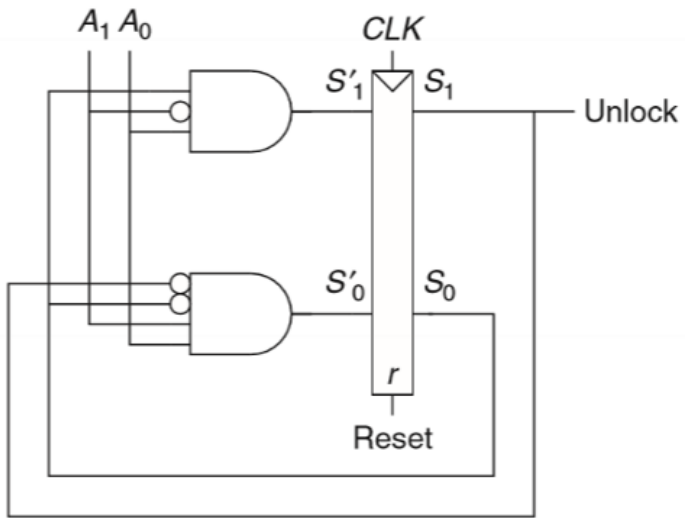
Categorize the circuits.



- Combinational: (a),
- Synchronous: (d), (e), (g)
- Asynchronous: none

5. From EX 3.9

Is this FSM? If it is, what type is it. Why?



This circuit is finite state machine. The type of this finite state machine is Moore Machine. In the type of finite state machine, the machine which only depends on the current state is Moore Machine, and the output “Unlock” is only depending on the state bits. By these reasons, this circuit is Moore FSM.

5 Glossary

1. Reminiscent

   Adjective:

   awakening memories of something similar; suggestive (usually followed by of)

   characterized by or of the nature of reminiscence.

   Given to reminiscence.

   1-1 Reminiscence:

      Noun:

      The act or process of recalling past experiences, events, etc.

      A mental impression retained and revived

      Something that recalls or suggests something else.

2. Synchronous

   Adjective:

   Occurring at the same time; coinciding in time; contemporaneous; simultaneous

   Relating to, or operating using fixed time intervals coordinated by a clock, as in paired data transmission.

   Relating to or being a computer operation that must complete before another event can begin

3. Asynchronous

   Adjective:

   Not occurring at the same time

   Relating to or being a computer operation that can occur independently, without waiting for another event.

   Relating to, or operating without the use of fixed-time intervals coordinated by a clock, as data transmitted one byte or character at time.

4. Encode

   Verb:

   To convert (a message, information, etc.) into code

5. Transition

   Noun:

   Movement, passage, or change from one position, state, stage, subject, concept, etc.

   A passage from one scene to another by sound effects, music, etc., as in a television program, theatrical production, or the like.

   Verb:

   To make a transition

6 Interview Question

What is the difference between a latch and a flip-flop? Under what circumstances is each one preferable?

The difference between a latch and a flip-flop is that a latch has no input which can control the state, and a flip-flop has inputs which can control the state.

In case of a latch, a latch is preferred in asynchronous sequential circuit system.

In case of a flip-flop, a flip-flop is preferred in synchronous sequential circuit system.

7 Reflection

The easiest section was first section of this chapter, Latches and Flip Flop. However, when it is compared to my past learning experience, it was relatively harder to learn about them. I am still confused about entire content of chapter 3. I think this reason is that I am not used to the information, and the examples of the content are not explicitly seen in my everyday life. Reading more about the content of the textbook, and having research on how the materials in

the textbook applicated may solve this problem. Thus, I am trying to do this in this week.

8 Study Question

1. Are there more types of FSM instead of Moore Machine and Mealy Machine?

2. How do we make large memory device such as 8GB memory from latch and flip-flop?

3. Is there any correlation between token and latency?