ENGR 2323 Digital Design Lab
Sequential Circuit Design

Sequential Circuits

Combinational circuits do not have memory. Sequential circuits (state machines) have memory elements combined with combinational logic. The memory elements store the state of the system and the combinational circuitry reacts to the inputs and current state to generate outputs and the next state. In response to change in the inputs, the system will generate outputs and update the state.

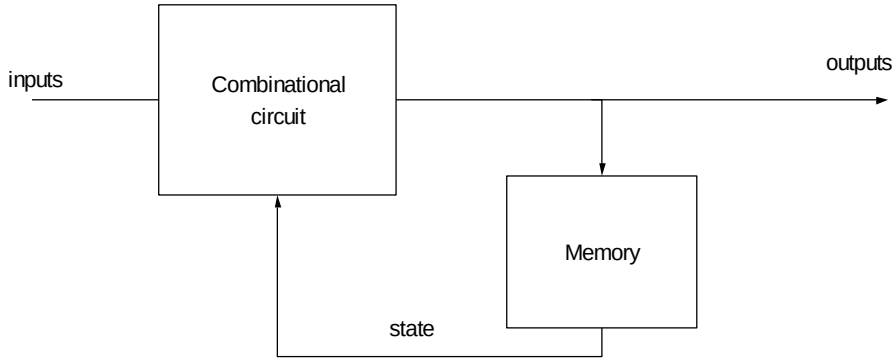

Figure 1. Sequential Circuit Block Diagram

Sequential circuits operate as follows: information is received through the inputs; the inputs combined with the current or present state of the system determine the outputs and the next state of the system. Thus, in response to a change in the input the system will generate outputs and update its state. The state change and outputs will depend on current state and current input.

Sequential circuits can be classified as either synchronous (clocked) or asynchronous (unclocked). Synchronous sequential systems use memory devices that are enabled with signal derived from a clock and the system behavior depends on inputs and outputs at discrete instants in time (system will not respond to change in input until get appropriate clock signal).

Sequential Circuit Design

Sequential circuit design involves realizing a circuit corresponding to a set of specifications. The initial information may include timing diagrams and sample input/output sequences.

The sequential circuit design procedure is:

- From the specifications construct the state diagram and state transition table.
- (Optional) Minimize the number of states using a state reduction (minimization) method.
- Perform state assignment (code the states).
- Determine the type and amount of state memory.
- Derive the flip-flop inputs for the state transitions in state table.
- Design the next state and output logic using combinational circuit design techniques.

- Realize the circuit and verify operation.

Design of Modulo-4 Counter (2-bit counter)
A modulo-4 counter is a 2-bit counter, it counts through the sequence 00, 01, 10, 11, 00 etc. The counter will count up when the control, U, is activated and hold at the current count when control is deactivated.

The state diagram of the counter, Figure 1, is a loop. State diagrams are graphical representations of a state machine. States are represented by circles and state transitions by directed arcs. States are labeled by the state name or state coding. The arcs are labeled with the inputs causing the transition. Outputs are placed inside the state for a Moore machine.

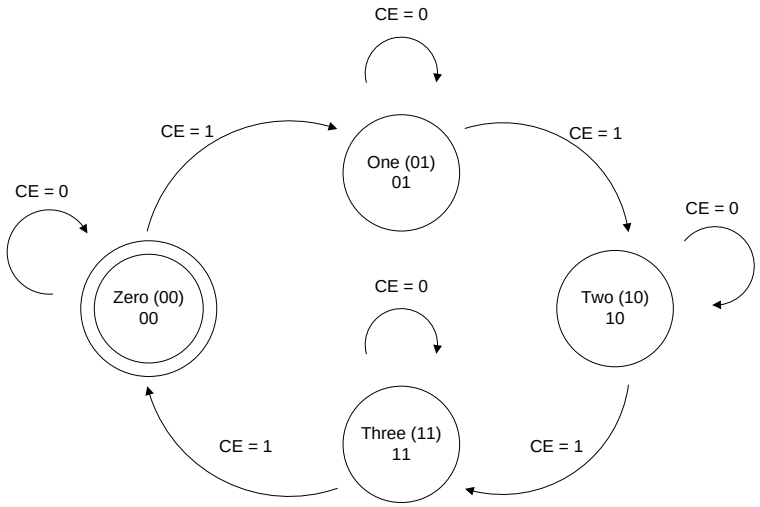

Figure 1. State Diagram of Modulo 4 Counter (2-bit counter)

The state of a system is the past information necessary to determine the future behavior of the system. The state information is typically stored as binary values in state variables. A circuit with n state-variables can have up to 2^n states. For the modulo-4 counter, there are four states corresponding to the count.

The state transition table for the modulo-4 counter is shown in Figure 2. State transition tables (state tables) contain a description of the sequential circuit behavior and are similar to function tables. A state table consists of columns for the present state, the inputs, the next state, and output. There is one row for each state-input combination and each row provides the output and next state for that state-input combination. For n state variables and m inputs there will be 2^{n+m} rows in state table.

For the module-4 counter, there are two state variables (4 states) and one input so 8 rows in the state table. Note that in both the state diagram of Figure 1 and state table of Figure 2 that the clock input is omitted. State transitions only occur during the active portion of the clock cycle (state does not change during inactive portions of clock cycle) so there is no loss of

information by omitting the clock signal from the state table and state diagram of a synchronous sequential circuit. Asynchronous inputs, such as a circuit reset, are also typically omitted from the state table and state diagram.

Q1 Q0 CE	F1 F0	Q1+ Q0+
0 0 0	0 0	0 0
0 0 1	0 0	0 1
0 1 0	0 1	0 1
0 1 1	0 1	1 0
1 0 0	1 0	1 0
1 0 1	1 0	1 1
1 1 0	1 1	1 1
1 1 1	1 1	0 0

Figure 2. State Transition Table of Modulo 4 Counter (2-bit counter)

From the state table, Boolean expressions for next state (flip-flop input) and outputs are generated as functions of the current state and circuit inputs (just like for combinational circuits). For D flip-flops, $Q^+ = D$, and the next state equations and the flip-flop input equations are the same. Next state equations describe the next state for the current state and input. Output equations describe the output for the current state and input. The next state is not entered until the next active clock edge. The expressions are simplified if they will be realized using SSI logic.

Q1/Q0 CE	00	01	11	10
0	0	0	1	0
1	1	1	0	1

Figure 3a. Karnaugh Map for D1

Q1/Q0 CE	00	01	11	10
0	0	1	0	1
1	0	1	0	1

Figure 3b. Karnaugh Map for D0

The flip-flop input (D flip-flop, $D = Q^+$) and output equations are

$$D1 = Q1 \cdot \overline{Q0} + Q1 \cdot \overline{CE} + \overline{Q1} \cdot Q0 \cdot CE$$

$$D0 = \overline{Q0} \cdot CE + Q0 \cdot \overline{CE}$$

$$F1 = Q1$$

$$F0 = Q0$$

The modulo-4 counter circuit, Figure 4, requires 2 flip-flops (1 flip-flop per state variable). The next state logic connects the flop-flop outputs (current state) and circuit inputs to the flip-flop input. The output logic connects the flip-flop outputs and circuit input to the circuit output. A common clock is typically connected to each flip-flop. Asynchronous inputs, such as presets and clears, are typically used to set up the initial state of the circuit. Figure 5 shows a simulation of the modulo-4 counter.

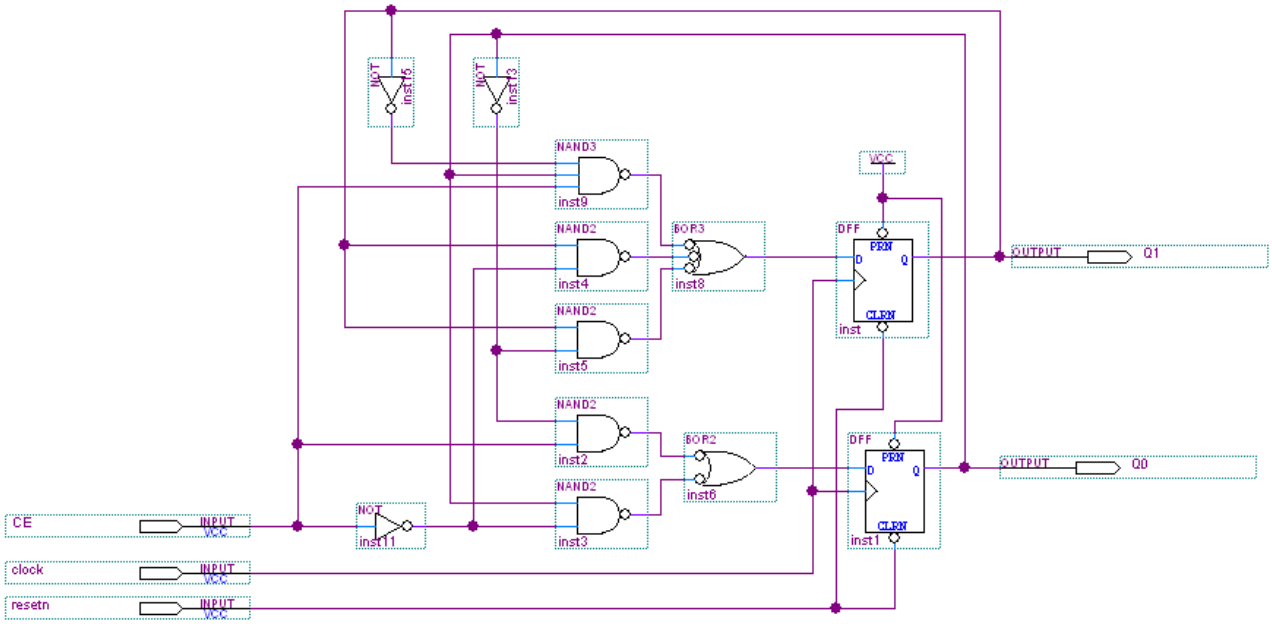

Figure 4. Modulo-4 Counter (2-bit counter) Circuit

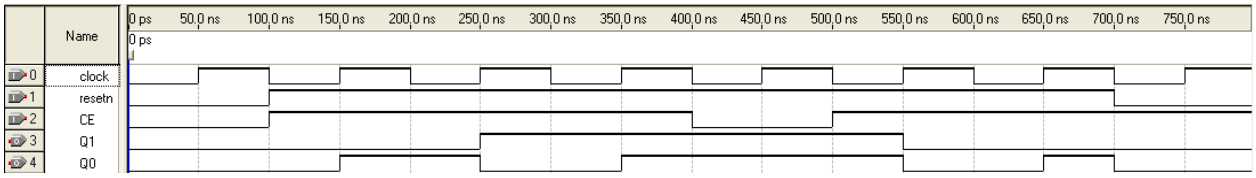

Figure 5. Modulo-4 Counter (2-bit counter) Simulation

References

None

Last modified Thursday, September 15, 2022