Finite State Machine

G V V Sharma*

Contents

***Abstract*—This manual explains state machines by deconstructing a decade counter.**

1 The Decade Counter

The block diagram of a decade counter (repeatedly counts up from 0 to 9) is available in Fig. 0. The *incrementing* decoder and *display* decoder are part of *combinational* logic, while the *delay* is part of *sequential* logic.

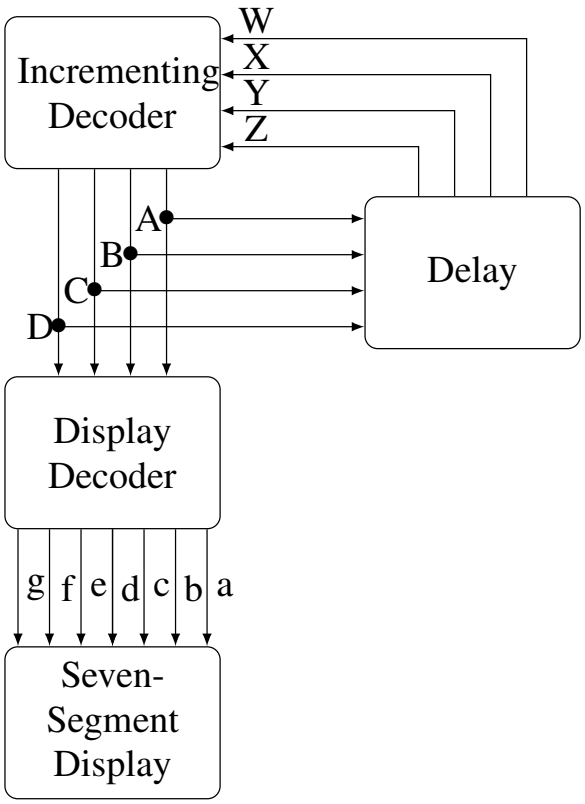

Fig. 0: The decade counter

2 Finite State Machine

1. Fig. 1 shows a *finite state machine* (FSM) diagram for the decade counter in Fig. 0. s_0 is the state when the input to the incrementing decoder is 0. The *state transition table* for the FSM is Table 0 in [1] where the present state is denoted by the variables W, X, Y, Z and the next state by A, B, C, D.
2. The FSM implementation is available in Fig. 2. The *flip-flops* hold the input for the time that is given by the *clock*. This is nothing but the implementation of the *Delay* block in Fig. 0.
3. The hardware cost of the system is given by

$$\text{No. of D Flip-Flops} = \lceil \log_2 (\text{No. of States}) \rceil \tag{2.1}$$

   For the FSM in Fig. 1, the number of states is 9, hence the number flipflops required = 4.
4. Draw the state transition diagram for a decade down counter (counts from 9 to 0 repeatedly) using an FSM.
5. Write the state transition table for the down counter.
6. Obtain the state transition equations with and without don't cares.
7. Verify your design using an arduino.
8. Repeat the above exercises by designing a circuit

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in.

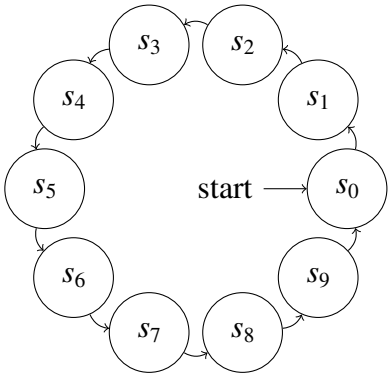

Fig. 1: FSM for the decade counter.

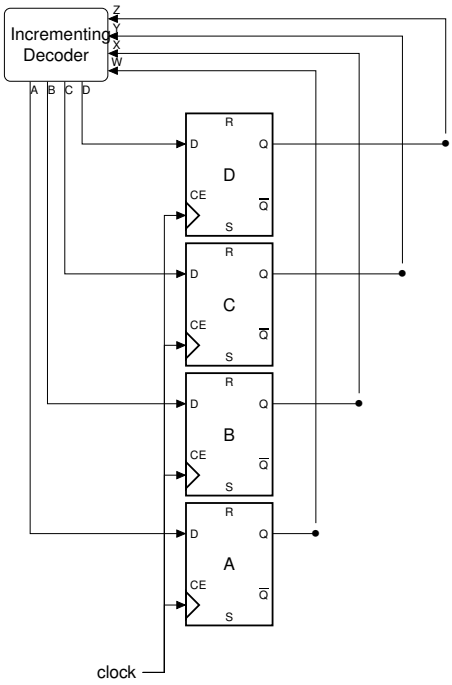

Fig. 2: Decade counter FSM implementation using D-Flip Flops.

that can detect 3 consecutive 1s in a bitstream.

References

[1] G. V. V. Sharma. Karnaugh Map. [Online]. Available: https://github.com/gadepall/arduino/raw/master/ide/kmap/gvv_kmap.pdf