

DigSim Assignment 2: A Finite State Machine

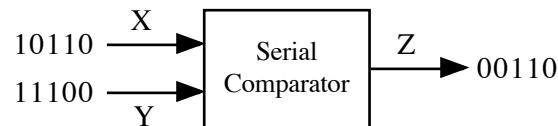
Due: Thursday November 21, 2002

Objective

The objective of this assignment is to design and implement a simple finite state machine.

Assignment

Design and implement in DigSim the serial comparator described in Problem A.31 of Murdocca & Heuring (p. 497). The machine has two inputs (X and Y) and 1 output (Z). The inputs are binary numbers given 1 bit at a time with the least significant bit given first. The output Z is 1 if the bits of X seen so far is larger (as a binary number) than the bits of Y seen so far. Consider the sample input and output sequence below:



The first output bit of Z is 0 because the first bit of X is 0 and the first bit of Y is 0. The second bit of Z is 1 because we have seen 10 for X and 00 for Y and $10 > 00$. Similarly the third bit of Z is 1 because $110 > 100$. The fourth bit of Z is 0 because $0110 < 1100$. The fifth bit of Z is 1 because $10110 < 11100$.

Implementation Notes

1. You may assume that when the machine starts that the “bits of X seen so far” is equal to the “bits of Y seen so far”.
2. You should be able to design a finite state machine for the serial comparator using only one D flip-flop.
3. Write out the truth table for a Boolean function with 3 inputs X, Y and Z_t and 1 output Z_{t+1} . Here Z_t represents the output of the comparator at time t and Z_{t+1} represents the output of the comparator at time t+1.
4. Produce the sum-of-products Boolean formulas for Z_{t+1} .
5. Simplify the Boolean formulas for Z_{t+1} using the properties of Boolean Algebra in Table A-1 of Murdocca & Heuring.
6. Implement a combinational circuit in DigSim using the simplified Boolean formulas.
7. Transform the combinational circuit into a finite state machine by connecting the output Z_{t+1} to the input Z_t through a D flip-flop.
8. Connect manual switches (powered & grounded) to the inputs X and Y. Use another manual switch to simulate a clock for the D flip-flop.
9. Test your finite state machine to make sure that it conforms to the specifications above.

Turning in your program

Save your circuit as you did in DigSim Assignment 1. Submit the circuit file using the Unix submit command as in previous assignments. The submission name for this assignment is: `digsim2`. The UNIX command to do this should look something like:

```
submit cs313-0101 digsim2 comparator.sim
```