Consider an FSM that outputs a 1 as long as there has been a (non-zero) even number of ones in the input stream since the last reset. The machine stops this behavior after the sequence 0101 (oldest to newest bit) is detected in the input stream. It will not output a one again until the machine is reset. This state machine has a simple datapath consisting of a 4-bit shift register, which is set to zero on reset, and a 4-bit comparator to detect the critical “stop outputting a 1” sequence.

To clarify the behavior of the machine, study the following examples (time advances from left to right, from first input bit to later input bits):

**INPUT:**
0 1 1 1 0 0 1 0 0 0 1 0 1 0 1 1 0 0 0 0 0 0 0 0 ...

**OUTPUT:**
0 0 1 0 0 0 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 ...

**INPUT:**
1 0 1 1 1 0 1 0 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 ...

**OUTPUT:**
0 0 1 0 1 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 ...

The block diagram of the machine is as follows. NOTE: new inputs are shifted into the shift register at the LEFT and old bits fall off at the right.

Assuming a Moore Machine implementation, complete the state diagram on the reverse side of this sheet. HINT: Think carefully about the relative timing of In being seen by the control and when the comparator can determine if the stopping sequence has been seen.

(Continues on back …)
Complete the state diagram, indicating what each of your states represent (e.g., seen odd ones, seen even ones, etc.). NOTE: Your instructor did it in FOUR states!