

# CS 341- Fall 1997

## Laboratory Exercise #3

### Creation and Test of a 6-bit sequence detector

Assigned: 9/25

Due: 10/2(at the beginning of lab)

#### **Purpose:**

The purpose of this exercise is to create Finite State Machine (FSM) which can detect a certain six digit binary sequence in a continuous serial binary input stream. When the correct sequence is detected, the single output signal should be a logical true value, in all other cases it should be a logical false. This exercise is designed to allow a gradual transition from strictly combinational circuits to a simple sequential circuit.

#### **Procedure:**

- 1). Every group will have a unique six digit sequence, the first step to this lab is to speak with your lab instructor and get your six-digit sequence.
- 2). Create a state diagram describing the action of the circuit.
- 3). From the state diagram, draw a state table, without yet labeling each of the states.
- 4). Decide upon the number of flip-flops or memory devices required, and choose a binary encoding for each of the states.
- 5). Create a state transition table, an output table, and then a transition table for each of the state bits.
- 6). From the tables drawn, generate an output function for each of the outputs (both explicit and next-state outputs). These functions should be minimal, minimized through whatever method you choose.
- 7). You should now have all the information required to implement this sequence detector in Logicworks. Start by creating a sub-circuit for each of your outputs, from the functions created in step # 6. The explicit output (i.e. sequence detection signal) should be wired to a binary probe, while the next state outputs should be wired to the input of a D flip-flop. Then wire in the inputs. The state inputs should come from the Q signals of the D flip flops, and the explicit input should come from a binary switch. Using the switch and the probe, and carefully watching the timing in the scrolling timing window., verify the operation of the circuit.

NOTE: If you use the D flip-flops in the “primlogi.clf” LogicWorks library, which is recommended, you should be aware that they latch the incoming value on the positive clock edge. This is the opposite of the example done in class.

NOTE: In order to watch signals in the timing window it is required that you name these signals on the schematic.

## **Deliverables**

- 1). Lab report, format specified in lab guidelines
- 2). Truth Table, minimization work, and minimal boolean expression for all combinational logic outputs. This should include state assignments.
- 3). Print outs of each of your LogicWorks circuits. Each should be constrained to be a single sheet of paper, and have your group member names on it, as well as a title telling me which function the circuit is implementing, and the class name.
- 4). An Email to me, with your final (top level) circuit MIME attached. This can be done using the attach command in PINE. If you have questions, ask me in lab - this can be done during the lab period on the day this lab is turned in.