University of California at Berkeley
College of Engineering
Department of Electrical Engineering and Computer Science

EECS 150   R. H. Katz
Fall 2000

Problem Set # 6 (Assigned 12 October, Due 20 October)

1. Design the logic for a 3-bit counter that follows the following sequence: 000, 001, 011, 010, 110, 111, 101, 100, 000 and repeats. Design the counter so when Reset is asserted, the counter enters the state 000.
2. Design the logic for a 3-bit counter that follows the following sequence: 001, 011, 111, 110, 100, 001, and repeat. Design the counter so that it is self-starting, i.e., whatever state it comes up in, it will eventually get into the sequence as shown above.
3. Consider the design of a simple elevator controller. The building has two floors, an up button on the first floor, a down button on the second floor, and two buttons inside the elevator indicating the floor to go to. While you can make assumptions, the behavior of the system has to be reasonable. For example, pressing the "Floor 2" button with the elevator on the second floor causes the elevator to remain there with its door open.
    (a) Identify your inputs, outputs, and name and describe your states.
    (b) Draw a symbolic state diagram for your design, labeling all state transitions.
4. Professor Katz has one of those incredibly complicated European Miele washing machines. The machine can advance through the following states: Extra Prewash, Prewash, Main Wash 1, Main Wash 2, Rinse 1, Rinse 2, Rinse 3, Starch, Rinse Hold, Graduated Spin, and Spin. The user manually starts the cycle by positioning a dial to Extra Prewash, Prewash, or Main Wash 1. When the Start button is press, the cycle begins in the indicated starting state. The machine has a "program control" to indicate the kind of fabrics being washed: Cotton Normal, Cotton Short, Permanent Press Normal, Permanent Press Short, Delicates Normal, Delicates Short, and Woolens. Normal cotton and permanent press programs cycle through every state. Short cotton and permanent press programs and the Delicates Normal program pass through Main Wash 1, skip Main Wash 2, enter Rinse 1 and 2, and skip Rinse 3. Delicates Short and Woolens are similar but also skip the second rinse. Finally if the Short Spin/Rinse Hold button is depressed, the program holds in the Rinse Hold until the button is released, and the program advances directly to Spin skipping the Graduated Spin.
    (a) Identify your inputs, outputs, and name and describe your states.
    (b) Draw a symbolic state diagram for your design, labeling all state transitions. Write down any additional assumptions you are making.
5. Consider the following variation on the classical traffic light controller problem. A North-South road intersects an East-West road. In addition to the Red/Yellow/Green traffic lights, each of the four corners of the intersection have *Walk/Don't Walk* signs facing N-S and E-W. In addition, there are push buttons for impatient pedestrians to force a quicker cycle of the lights so they may cross the street (separate buttons for N-S and E-W crossings). The *Walk/Don't Walk* signs cycle for the N-S road as follows. When the traffic lights turn green in the N-S direction, *Walk* is illuminated. Half way through the green light time, the *Don't Walk* sign becomes illuminated and starts to blink (*Walk* is no longer illuminated). When the traffic light turns yellow, the *Don't Walk* sign remains solidly illuminated (i.e., no blinking). It stays this way throughout the red light cycle. The behavior is the same in the E-W direction. The Pedestrian Button works as follows. If the facing light is red, and the button is pushed, the remaining green light time of the cross street is reduced by half (e.g., shift the time left by one bit to the right). Note that despite popular myth, it doesn't matter how many times this button is pushed! Further note that the Yellow light time on the cross street must never reduced (otherwise expect serious accidents!).
    (a) Assume that the standard Green light time is 56 seconds, the Yellow time is 8 seconds, and the Red time is 64 seconds. Draw a simple timing chart that shows the behavior of the N-S

and E-W traffic lights and *Walk/Don't Walk* lights. Show a complete cycle with and without a pedestrian button press.

(b) Identify your inputs and outputs. Think about issues like whether the state machine needs to see a normal time interval or a half time interval, as well as the need to reset the push button inputs once they have affected the state machine. What additional circuitry, like timers and flipflops, do you need outside of the state machine?

(c) Draw a symbolic state diagram. Make clear all of your assumptions about the problem consistent with the specification above.

(d) Choose a state encoding. What is the rationale for your choice?