

Homework 12: Due Thursday May 3rd (10:30 am, beginning of class)

Write your name at the top of each page.

- Start a **new page for each problem**.
 - **Order** and **staple** your pages.
 - Always complete the reading assignments *before* attempting the homework problems.
 - Show all of your work. Use written English, where applicable, to provide a log of your steps in solving a problem. (For numerical homework problems, the writing can be brief.)
 - A solution which requires physical units is *incorrect* unless the units are listed in the result.
 - Underline, circle or box your result.
 - Always write neatly. Communication skills are essential in engineering and science. If neither the TA nor the instructor can read it, you will receive zero points.
-

1) State Machines

- a) Design a Mealy state machine to detect the sequence “1011” on its input. Every occurrence of this input sequence should be detected independent of other '0's or '1's at the beginning e.g. “111**1011**” and “0**10111011**”, but these sequences can not overlap e.g. “1**1011**011”. Use the 8 step state machine design process we used in class and label all steps. You can omit the *state minimization* step. Use **D Flip-Flops** in your design and choose binary encoding.
- b) Design a Moore state machine to detect the sequence “1011” on its input. Every occurrence of this input sequence should be detected independent of other '0's or '1's at the beginning e.g. “111**1011**” and “0**10111011**”, but these sequences can not overlap e.g. “1**1011**011”. Use the 8 step state machine design process we used in class and label all steps. You can omit the *state minimization* step. Use **JK Flip-Flops** in your design and choose binary encoding(See book chapter 8.7.4.).

2) State Machine and VHDL

Design a state machine for a traffic light with car detection. The traffic light is at the intersection of Main Street and High Street. Main Street is the default direction of traffic, i.e. when the traffic light is switched on Main Street will have a green light and High Street a red light. High Street has the car detection. Only when the traffic light detects a car on High Street, the traffic light switches to give High Street a green light. During this time, Main Street will have a red light. Then the traffic light will return to the default direction. Assume that the car detection shows that a car is present for the whole time a car is waiting at the light on High Street and that the traffic light reacts to this detection only when the light on High Street is red, i.e. the default.

The traffic light should have the standard light pattern for the US: Red -> Green -> Yellow -> Red. When one street has a green light or a yellow light the other must have a red light. Assume you have a system clock period of 5 seconds. The yellow light should be illuminated for 5 seconds (1 state), the green for 20 seconds (4 states) and therefore the red for a minimum of 25 seconds.

- Determine the inputs (1) and outputs (6) assuming that there is only one set of lights for each street.
- Design a Moore state machine for this traffic light. Follow the design process until you have described the State Table (Step 3) (you need 10 states).
- Compile the VHDL code from the state machine VHDL example from class and simulate it. Get familiar with it. You can find this handout on the class webpage.
- Use the state machine VHDL example from class as a basis to implement the traffic light state machine in VHDL. Turn in the source code and waveform showing one full iteration through all states.

3) State Minimization

Minimize the following state machines using the minimization through partitioning method we used in class.

- Figure 1
- Figure 2

