Taylor Cowley

HW 12

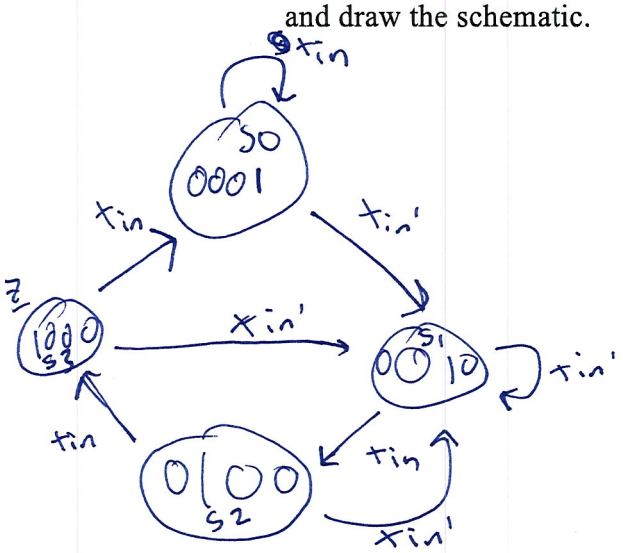
EE 220

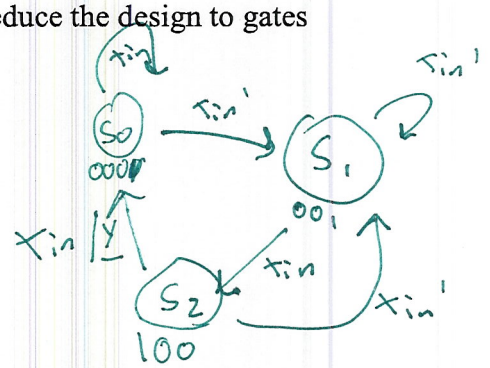
Jun07

ECEn 220
Chapter 17 Homework

1. Design the FSM of Figure 16.5 using a one-hot encoding and reduce the design to gates and draw the schematic.

2. Design the FSM of Figure 16.9 using a one-hot encoding and reduce the design to gates and draw the schematic.





$S_0 = S_0 \cdot X_{in} + S_3 \cdot X_{in}$

$S_1 = S_0 \cdot X_{in}' + S_1 \cdot X_{in}' + S_3 \cdot X_{in}' + S_2 \cdot X_{in}'$

$S_2 = S_1 \cdot X_{in}$

$S_3 = S_2 \cdot X_{in}$

$Z = S_3$

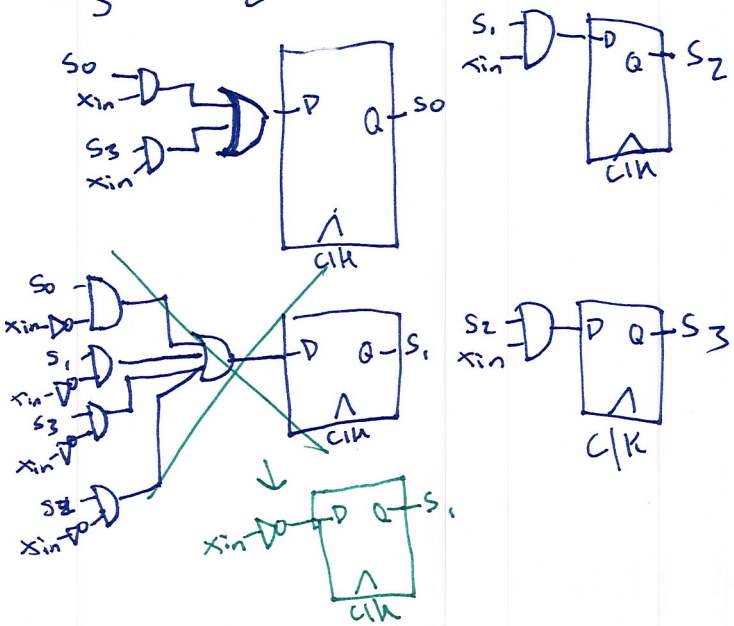
$S_0 = S_0 \cdot X_{in} + S_2 \cdot X_{in}$

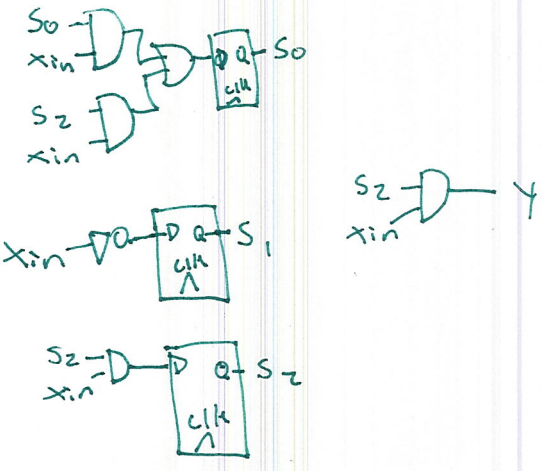
$S_1 = S_0 \cdot X_{in}' + S_1 \cdot X_{in}' + S_2 \cdot X_{in}'$

$= X_{in}'$

$S_2 = S_1 \cdot X_{in}$

$Y = S_2 \cdot X_{in}$

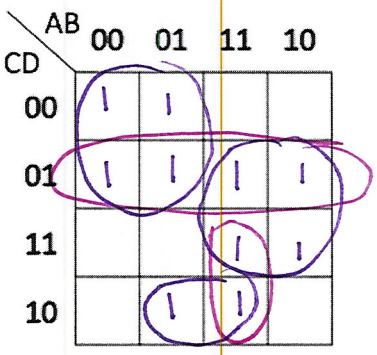




ECEn 220
Chapter 18 Homework

1. Plot the following function on a KMap and identify a minimum SOP solution for it. Then, solve it a second time using hazard-free minimization. Show your work.

$$F(A,B,C,D) = m0 + m1 + m4 + m5 + m6 + m9 + m11 + m13 + m14 + m15$$

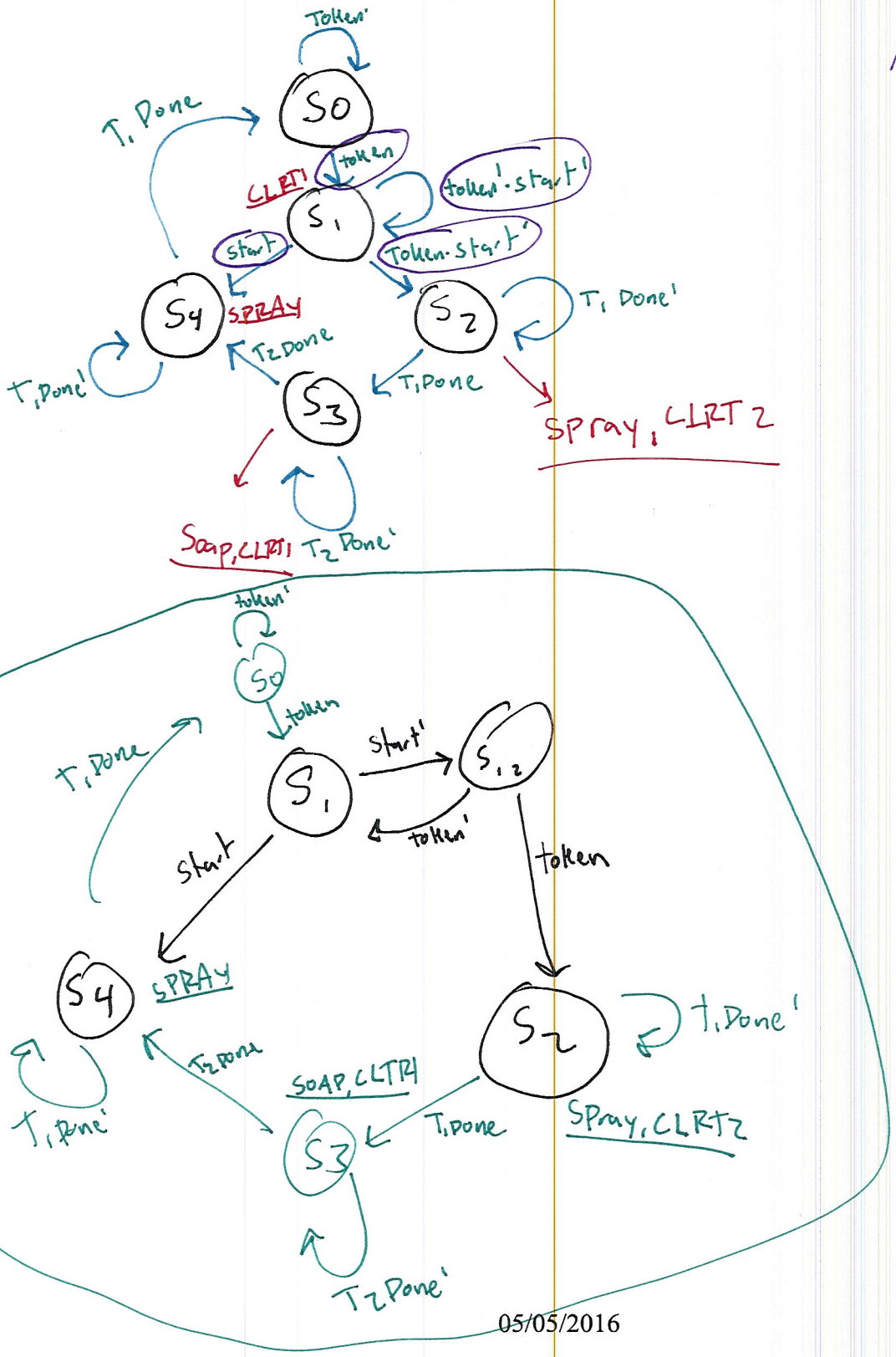


Minimum POS F = $A'C' + AD + BCD'$

Hazard Free F= $A'C' + AD + BCD' + C'D + ABC$

2. Make a copy of the state graph of Figure 16.14. On it, list the asynchronous inputs. Circle the transitions dependent on those asynchronous inputs. Do any of these transitions depend on more than one asynchronous input? If so, redraw the state graph by adding additional states and transitions so that no transition is dependent on more than one asynchronous input. Assume that the timers (T1 and T2) run on the same clock as the state machine.

Async inputs:
Token
Start

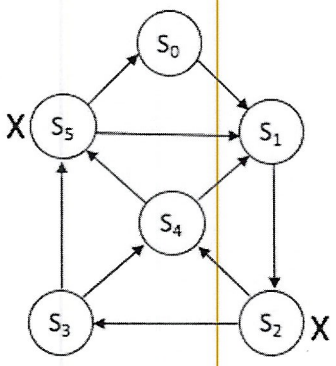


3. This problem is not directly discussed in the text or the lecture slides, but it is related to the topic of chapter 18. A brief explanation of why this problem is relevant follows. This chapter discusses how to handle asynchronous inputs. Another circuit has to generate these asynchronous signals. This problem is about how to generate good outputs that could be used as the asynchronous inputs to another circuit. In order to be a good asynchronous signal, it should be free of false outputs.

One way to make sure our Moore outputs will be hazard free is to use hazard fee logic minimization when designing the output forming logic. However, because of our choice of state encoding, this may not be possible. For example, there may be state transitions in our state machine that would involve multiple state bits changing at the same time. As a result, the hazard free minimization technique we've studied may not work.

The other alternative is to carefully choose our state encodings so that false outputs are eliminated. This is what the problem asks us to do. We can do this either by choosing state encodings that will make our hazard free logic design technique viable, or by choosing a state encoding that eliminates the output forming logic for X altogether (not require external gates).

Consider the FSM state graph shown below. No expressions have been shown for the transition conditions since they are not material to this problem. What is important is the set of state transitions and the X output.



S_0: 000
S_1: 001
S_2: 011 → X
S_3: 100
S_4: 101
S_5: 111 → X

As can be seen, the X output is asserted in both state $\mathbf{S_2}$ and state $\mathbf{S_5}$. Further, it is desired that signal X contain no false outputs on state transitions. Plot how you would assign state encodings using a K-Map and explain the conditions required to avoid false outputs on signal X. Then, list the state encoding chosen.

Let's pick one not requiring OFL

N_1N_0 \ N_2	0	1
00	0 S_0	0 S_3
01	0 S_1	0 S_4
11	1 S_2	1 S_5
10	X	X

$X = N_1$. No OFL required.

05/05/2016