[CS M51A F18] Practice Homework 6 - do not turn in

Solutions posted: 12/6/18

Problem 1

We would like to put together a 12-input decoder using 4-input decoders. Let us consider the following two methods.

1. Using a tree structure, how many levels would we need? How many 4-input decoders are used in total?
2. The textbook has an example of using a coincident structure which divides the n-inputs into two groups using $\frac{n}{2}$-input decoders. This can be expanded to the case where we divide the n inputs into k groups, using r-input decoders. In this case, what would be the value of k in terms of n and r? How many AND gates would we need?
3. Now, for our coincident 12-input decoder, how many 4-input decoders do we need? How many AND gates? How many inputs do we need for the AND gates?

Problem 2

We want to use multiplexers to implement the function $f(a, b, c, d) = \sum m(1, 2, 4, 7, 10, 11, 14)$ using the following methods.

For both cases, show your work, either the minterm expressions or K-map implementations. Assume that complements to inputs are available.

1. Implement the function using 8-to-1 multiplexers. Use a, b, c as the selection bits.
2. Repeat the same process using 4-to-1 multiplexers. Use a, b as the selection bits, and for the additional logic needed, only use XOR gates and NOR gates.

Problem 3

Complete the following table for the given representations in each part.

	Signed integer x (in decimal)	Representation x_R (in decimal)	Bit-vector $\underline{x}$
a	-25		
b		37	
c			110110

The 'Signed integer' column shows the actual value of the signed integer, and the 'Representation' column holds the value of the bit-vector representation. Assume that any bit vector that is shorter than the given bit length has leading 0s.

1. Two's complement, $n = 7$ bits
2. Two's complement, $n = 6$ bits
3. Ones' complement, $n = 6$ bits

Problem 4

We would like to identify the working signals of an arithmetic unit at work, namely the two's-complement arithmetic unit shown in the textbook at Figure 10.12 (page 297). For the system, fill in the table below according to the given computation in each part. c_0, K_x and K_y are control signals, and z, c_{out}, ovf, $zero$ and sgn are result signals.

1. $z = x + y$

x	y	c_0	K_x	K_y	z	c_{out}	ovf	$zero$	sgn
00000000	00000000								
10101010	10100101								
10110110	00110011								

2. $z = x - y$

x	y	c_0	K_x	K_y	z	c_{out}	ovf	$zero$	sgn
00100010	00100010								
10010010	01000011								
00010011	00110011								

Problem 5

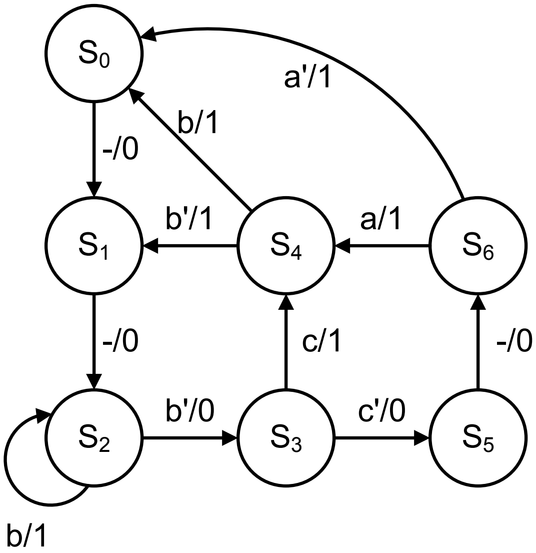
Using a modulo-16 binary counter with parallel inputs explained in the textbook (p. 321) along with logic gates, we would like to implement the following counters.

For each design, minimize the gate logic for the input signals and load signal as much as possible using K-maps.

1. A modulo-10 counter.
2. A 5-to-14 counter.
3. A counter which counts in the following sequence: 0, 1, 2, 3, 4, 8, 9, 10, 13, 14, 15

Problem 6

Implement a sequential system using the standard modulo-16 binary counter with parallel input. The system has three binary inputs a, b, c and one binary output z. The transition and the output functions are specified by the state diagram shown below.



1. Setting $LD = CNT'$, obtain the expression for the input CNT in terms of states (S_0 to S_6) and input signals.

2. Find all load input combinations that we need for this system. For each combination, write the input conditions in terms of states (S_0 to S_6) and input signals. Simplify the conditions as much as possible.

3. Using K-maps, obtain the minimal expressions for I_3 to I_0 and the output z, in terms of counter outputs (Q_3 to Q_0) and input signals.