Solutions for HW #4 CS W4825 04s

1.(2.9 in Asynch Bk)

Note that, in some of the handout copies, the reproduction may have obscured the fact that the entry for the third row, 2nd column is 6,1.

In constructing the pair chart, we must insert X's initially not only where a 1 conflicts with a 0, but also where a – conflicts with a 0 or 1. The final pair chart is as shown below:

	1	2	3	4	5	6	7
2	XX						
3	XX	XX					
4	XX	XX	XX				
5	XX	XX	XX	XX			
6	XX	XX	XX	17	XX		
7	38	XX	XX	XX	XX	XX	
8	XX	XX	46	XX	XX	XX	XX

The MC's (which are actually equivalence sets) are 18, 38, 46

This leads to the following minimal equivalent flow table:

(17)	1	2,0	-,1	3,-	2,0
(2)	2	3,0	5,1	2,0	-,-
(38)	3	3,0	4,1	-,-	5,0
(46)	4	-,-	1,1	2,-	-,-
(5)	5	-,-	1,1	1,1	-,-

2.10.If a & b are equivalent, then, for each row of a, there must be an equivalent row of b. Because of the output patterns, only 1b and 3b can be equivalent to 1a. (I will use = to mean equivalent here and p->q to mean p implies q.) But 1a=1b ->1a->2b (column 00). Since 1a and 2b have different output patterns, it is clear that they are not equivalent, and so 1a=1b is false. So, if a=b, it follows that 1a=3b. We could draw an implication graph (or use a pair chart) that would show that 1a3b->2a4b->3a1b->4a2b->1a3b and that none of these pairs implies any other pair. Since all these pairs are output equivalent (have the same output patterns, it follows that they are all valid. So states 1, 2, 3, 4 of table (a) are equivalent, respectively to states 3, 4, 1, 2 of table (b).

2.11

(a) If a covers b, then every row of b must be covered by at least one row of a. Because of output patterns, only 3a can cover 2b. We trace the implications of this, looking for any implications of covering failures due to immediate outputs (output patterns). We see the chain 3a2b->1a->3b->1a1b, where 1a does NOT cover 1b because of the don't care entry for the output of state 1B if table a. Therefore table a does NOT cover table b.

(b) Because of output patterns, only 2b might cover 3a. Tracing the implication paths we find that 2b3a->3b2a. Since 3b does NOT cover 2a due to the fact that in column B 3b has a – where 2a has a 0, it follows that 2b does NOT cover 3a and therefore that table b does NOT cover table a.

2.12. We can find a table covering both a and b by combining the two tables, simply stacking one on top of the other, with the rows of the second table renamed so as to have different names than the rows of the first table. Treat the result as a single table, even tho there are no transitions between the states of the top table and the states of the bottom table. Then we find a minimal cover of the combined table. This new table will cover ALL rows of both parts and so will cover both of the original tables.

Carrying out this procedure below, the rows of table b are transformed by adding 3 to each row number, so the rows 1, 2, and 3 are relabeled 3, 4, and 5, respectively. So the combined table is as shown below.

	A	B	C
1	2,-	-,-	1,0
2	1,-	3,1	2,0
3	2,-	1,0	1,0
4	5,1	5,1	4,0
5	4,-	5,-	6,0
6	4,-	4,0	6,0

If we apply our standard technique for reducing this table, we will find MC's {135, 24, 12, 16}. Using the implication graph, we will find that {135, 16, 24} constitutes a minimal closed cover, which leads to the table below, which covers both a and b.

		A	B	C
(1a3a2b)	1	3,-	1,0	2,0
(1a3b)	2	3,-	3,0	2,0
(2a1b)	3	1,1	1,1	3,0

In text.

5.9.

	X = 0	X = 1	
1	1,0	2,0	Last input 0, or in even 1-block
2	1,1	1,0	In odd 1-block

5.11. We will have 2 outputs, Z1 and Z2. If A>B, Z1Z2=10, if B>A, Z1Z2=01, if A=B, Z1Z2=00. (We could also have chosen to have 3 output variables, one to be set to 1 for each case.) The flow tables are as shown below.

	AB = 00	01	11	10	
1	1,00	2,01	1,00	3,10	Most Sig. bits equal so far
2	2,01	2,01	2,01	2,01	B is greater
3	3,10	3,10	3,10	3,10	A is greater

(a)

	AB = 00	01	11	10	
1	1,00	2,01	1,00	3,10	Least Sig. bits equal so far
2	2,01	2,01	2,01	3,10	Least Sig. bits of B greater
3	3,10	2,01	3,10	3,10	Least Sig. bits of A greater

(b)