TP 6
Digital Electronics [ELEC-H-310]
Correction

v1.0.1

Question 1. Build the table of equivalence conditions for the following Huffman table (representing a Moore machine).

	ab				Z_1Z_2
1	**1**	2	3	10	00
2	7	**2**	9	4	11
3	7	6	**3**	4	10
4	7	2	5	**4**	01
5	11	8	**5**	12	10
6	11	**6**	5	10	11
7	**7**	8	5	4	00
8	1	**8**	3	4	11
9	7	8	**9**	12	10
10	11	6	9	**10**	01
11	**11**	8	9	10	00
12	7	2	5	**12**	11

Answer:

2	x										
3	x	x									
4	x	x	x								
5	x	x	7-11; 6-8; 4-12	x							
6	x	7-11; 5-9; 4-10	x	x	x						
7	2-8; 4-10 3-5	x	x	x	x	x					
8	x	1-7; 3-9	x	x	x	1-11; 3-5; 4-10	x				
9	x	x	6-8; 4-12	x	7-11;	x	x	x			
10	x	x	x	7-11; 5-9; 2-6;	x	x	x	x	x		
11	2-8; 3-9	x	x	x	x	x	5-9; 4-10;	x	x	x	
12	x	5-9; 4-12	x	x	x	11-7; 10-12; 2-6	x	1-7; 2-8; 3-5; 4-12	x	x	x
	1	2	3	4	5	6	7	8	9	10	11

Question 2. From this Huffman table, show the two possible automatons if we allow to merge lines for which the output is

a) the same

b) different

Y_1 Y_2	00	01	11	10	ab	Z
1	**1**	2	3	4		0
2	-	**2**	**2**	**2**		1
3	1	2	**3**	-		1
4	**4**	-	2	**4**		1

What are those two automatons called?

Answer:

a) If we cannot merge states with different outputs, we talk about a Moore machine. The equivalence table thus is:

2	x		
3	x	OK	
4	x	OK	2-3 1-4
	1	2	3

We can merge states 2 and 3, or 2 and 4. By merging 2 and 3, we get:

Y_1 Y_2	00	01	11	10	ab	Z
1	**1**	2	2	4		0
2	1	**2**	**2**	**2**		1
4	**4**	-	2	**4**		1

The next step is to choose a coding for the states. Let's take 1, 2 and 4 coded 00, 01 and 11[1].

Y_1 Y_2	00	01	11	10	ab	Z
00	**00**	01	01	11		0
01	00	**01**	**01**	**01**		1
11	**11**	-	01	**11**		1
10	-	-	-	-		-

$Y_1 = y_1\bar{b} + \overline{y_2}a\bar{b}$

y_1y_2 \ ab	00	01	11	10
00	0	0	0	1
01	0	0	0	0
11	1	-	0	1
10	-	-	-	-

$Y_2 = y_1 + a + b$

y_1y_2 \ ab	00	01	11	10
00	0	1	1	1
01	0	1	1	1
11	1	-	1	1
10	-	-	-	-

[1]Since the code 10 is not used, this state does not exist and all it's values are replaced with *don't care*.

y_1y_2 \ ab	00	01	11	10
00	0	-	-	-
01	-	1	1	1
11	1	-	1	1
10	-	-	-	-

$Z = y_2$

b) If we allow merging states with different outputs, we are working with a Mealy machine. However, we need to keep in mind that we still cannot merge stable states with different outputs. For example, let's take a look at states 1 and 4. For the input $ab = 00$, both future states are stable and their output is 0 and 1. Since the output are different for stable future states, we cannot merge 1 and 4. On the other hand, if we consider states 2 and 3 for the input $ab = 11$, both futur states are stable, but with the same output.

From this, we can build an equivalence table:

2	2-3 2-4		
3	OK	OK	
4	x	OK	2-3 1-4
	1	2	3

Let's merge states 1 and 3 into 1, and 2 and 4 into 2:

Y	00	01	11	10	ab	Z
1	**1**	2	**1**	2		?
2	**2**	**2**	**2**	**2**		?

Now, the output depends on the input and the current state. We thus need to define an output for each future state. Merged future states inheritate the output of their original state.

Output of merged states that were originally unstable is set so that we avoid glitches. The table 1 summarizes the rules to set the outputof "unstable" states.

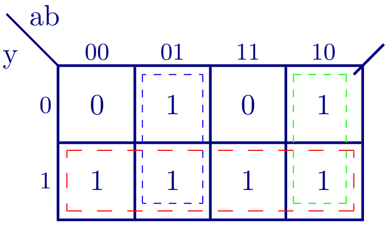
Z_p	Z_f	Z_t
0	0	0
0	1	-
1	0	-
1	1	1

Table 1 – Output of the transition state Z_t when we start at Z_p and end up at Z_f.

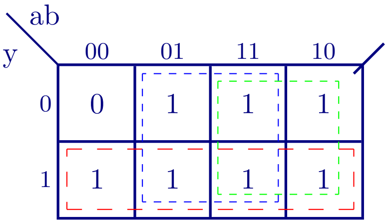
This way, we obtain this Huffman table:

Y	00	01	11	10	ab
1	**1/0**	2/1	**1/1**	2/1	
2	**2/1**	**2/1**	**2/1**	**2/1**	

By coding 1 with 0 and 2 with 1, we obtain the following:



$Y = y + \overline{a}b + a\overline{b}$



$Z = y + a + b$

Question 3. In the following Huffman table:

$Y_1\ Y_2$	00	01	11	10	ab
1	**1**	**1**	3	-	
2	**2**	**2**	3	4	
3	2	1	**3**	**3**	
4	1	-	**4**	**4**	

a) Find a coding without race problems. Write the retroaction functions Y_1 and Y_2.

Answer: A race problem happens when the machine needs to change more than one retroaction variable to transition. For example, when we go from 00 to 11, both y_1 and y_2 change. But if both signals do not travel at the same speed, we could end up in the wrong state.

To avoid this, it may be enough to choose an appropriate coding. For example, if we code 1 by 00, we need to code the state 3 so that it is at a Hamming distance of 1 from 1. Let's say we replace 3 by 01.

In the end, we can get this coded Huffman table:

$Y_1\ Y_2$	00	01	11	10	ab
00	**00**	**00**	01	-	
11	**11**	**11**	01	10	
01	11	00	**01**	**01**	
10	00	-	**10**	**10**	

Beware: when extracting the K-map from the Huffman table, don't forget to sort the states so that all neighbouring lines are at a Hamming distance of 1 from each other.

$$Y_1 = y_2\overline{a}\,\overline{b} + y_1\overline{a}b + y_1\overline{y_2}a + y_1a\overline{b}$$

y_1y_2 \ ab	00	01	11	10
00	0	0	0	-
01	1	0	0	0
11	1	1	0	1
10	0	-	1	1

$$Y_2 = y_2\overline{a}\,\overline{b} + y_1y_2b + \overline{y_1}a$$

y_1y_2 \ ab	00	01	11	10
00	0	0	1	-
01	1	0	1	1
11	1	1	1	0
10	0	-	0	0

b) Solve the race problems with this imposed coding: 1 = 00; 2 = 01; 3 = 11; 4 = 10.

Answer: By applying the coding, we get this Huffman table:

Y_1 Y_2	00	01	11	10	ab
00	**00**	**00**	11	-	
01	**01**	**01**	11	10	
11	01	00	**11**	**11**	
10	00	-	**10**	**10**	

We have several race problems. For example, when inputs change from 01 to 11 in state 00, the machine sends us to 11 where both retroaction variables changed. Here are all the race problems:

a) Transition from state 00 to 11 when inputs change from 01 to 11.
b) Transition from state 01 to 10 when inputs change from 00 to 10.
c) Transition from state 11 to 00 when inputs change from 11 to 01.

Since the coding is imposed, we need to use *don't care* and unstable states to solve the race problems, by addind intermediary transitions. Let's illustrate with partial Huffman tables:

a)

Y_1Y_2	01	11
00	**00**	11
01	**01**	11
11	00	**11**

becomes

Y_1Y_2	01	11
00	**00**	01
01	**01**	11
11	00	**11**

b)

Y_1Y_2	00	10
00	**00**	-
01	**01**	10
10	00	**10**

becomes

Y_1Y_2	00	10
00	**00**	10
01	**01**	00
10	00	**10**

c)

Y_1Y_2	01	11
00	**00**	01
11	00	**11**
10	-	**10**

becomes

Y_1Y_2	01	11
00	**00**	01
11	10	**11**
10	00	**10**

We end up with a new Huffman table without race problems:

Y_1 Y_2	00	01	11	10	ab
00	**00**	**00**	01	10	
01	**01**	**01**	11	00	
11	01	10	**11**	**11**	
10	00	00	**10**	**10**	

Question 4. By coding states 1, 2, 3 and 4 by y_1y_2 = 00, 01, 11 and 10, compute the excitation functions or memory modules for this automaton:

Y_1 Y_2	00	01	11	10	ab
1	**1**	**1**	2	-	
2	**2**	3	**2**	**2**	
3	4	**3**	2	-	
4	**4**	1	2	**-**	

Consider bistables D and SRc.

Answer: First, coding the Huffman table:

Y_1 Y_2	00	01	11	10	ab
00	**00**	**00**	01	-	
01	**01**	11	**01**	**01**	
11	10	**11**	01	-	
10	**10**	00	01	**-**	

The excitation terms are in table 2 and the excitation tables in 3.

Q	Q^+		
0	0	μ_0	Maintien à 0
0	1	ε	Enclenchement
1	0	δ	Désenclenchement
1	1	μ_1	Maintien à 1

Table 2 – Excitation terms, Q is the current state, Q^+ the future state.

	D
μ_0	0
ε	1
δ	0
μ_1	1

(a) Flip-flop D

	S	R
μ_0	0	-
ε	1	0
δ	0	1
μ_1	-	0

(b) Flip-flop SR

Table 3 – Excitation tables of (a) D and (b) SR.

The first step is to build the state table from the Huffman table. We simply need to replace the stable states with their proper holding term. 00 becomes $\mu_0\mu_0$, 01 becomes $\mu_0\mu_1$, etc. As for the transition states, it depends on the variable that is changed. For example, if we go from state 00 to 01, the first term of the future state is held to 0, whilst the second changes from 0 to 1, hence becoming $\mu_0\varepsilon$.

We finally get this state table:

$Q_1^+\ Q_2^+$	00	01	11	10	ab
00	$\mu_0\mu_0$	$\mu_0\mu_0$	$\mu_0\varepsilon$	-	
01	$\mu_0\mu_1$	$\varepsilon\mu_1$	$\mu_0\mu_1$	$\mu_0\mu_1$	
11	$\mu_1\delta$	$\mu_1\mu_1$	$\delta\mu_1$	-	
10	$\mu_1\mu_0$	$\delta\mu_0$	$\delta\varepsilon$	-	

Since all future states are composed of two variables, we will need two bistables, one for each "sub-state-table":

Q_1^+	00	01	11	10	ab
00	μ_0	μ_0	μ_0	-	
01	μ_0	ε	μ_0	μ_0	
11	μ_1	μ_1	δ	-	
10	μ_1	δ	δ	-	

Q_2^+	00	01	11	10	ab
00	μ_0	μ_0	ε	-	
01	μ_1	μ_1	μ_1	μ_1	
11	δ	μ_1	μ_1	-	
10	μ_0	μ_0	ε	-	

a) Let's first work with a bistable D. Using the excitation table 3a, we can build two state tables for the two bistables.

D_1	00	01	11	10	ab
00	0	0	0	-	
01	0	1	0	0	
11	1	1	0	-	
10	1	0	0	-	

D_2	00	01	11	10	ab
00	0	0	1	-	
01	1	1	1	1	
11	0	1	1	-	
10	0	0	1	-	

Hence the K-maps and excitation functions:

y_1y_2 \ ab	00	01	11	10
00	0	0	0	-
01	0	1	0	0
11	1	1	0	-
10	1	0	0	-

$D_1 = y_1\overline{b} + y_2\overline{a}b$

y_1y_2 \ ab	00	01	11	10
00	0	0	1	-
01	1	1	1	1
11	0	1	1	-
10	0	0	1	-

$D_2 = y_2b + \overline{y_1}y_2 + a$

b) We then do the same for the SR by using table 3b.

S_1R_1	00	01	11	10	ab
00	0-	0-	0-	–	
01	0-	10	0-	0-	
11	-0	-0	01	–	
10	-0	01	01	–	

S_2R_2	00	01	11	10	ab
00	0-	0-	10	–	
01	-0	-0	-0	-0	
11	01	-0	-0	–	
10	0-	0-	10	–	

From which we find K-maps and excitation functions.

y_1y_2 \ ab	00	01	11	10
00	0	0	0	-
01	0	1	0	0
11	-	-	0	-
10	-	0	0	-

$S_1 = y_2\overline{a}b$

y_1y_2 \ ab	00	01	11	10
00	-	-	-	-
01	-	0	-	-
11	0	0	1	-
10	0	1	1	-

$R_1 = a + \overline{y_2}b$

y_1y_2 \ ab	00	01	11	10
00	0	0	1	-
01	-	-	-	-
11	0	-	-	-
10	0	0	1	-

$S_2 = a$

y_1y_2 \ ab	00	01	11	10
00	-	-	0	-
01	0	0	0	0
11	1	0	0	-
10	-	-	0	-

$R_2 = y_1\bar{b}$