جامعة بيرزيت

BIRZEIT UNIVERSITY

ANSWER BOOKLET

Student: Advanced	Number
Course: Department: asynch "3"	Number:
Division:	Instructor:
Date: Day	Month Year

For Instructor's Use

Question	Grade
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
Total	

Example

Design a negative Edge T-FF

Solution

① Design specification:

This circuit has 2 inputs (clk and T) and one output Q. The following are all the possible total states

T	clk	Q	stable??
0	0	0	yes
0	0	1	yes
0	↓	0	yes
0	↓	1	yes
1	0	0	yes
1	0	1	yes
1	1	0	yes
1	1	1	yes

② get the primitive flow table

State	Inputs T	Inputs clk	Output Q	Comments
a	1	1	0	after d or f
b	1	0	1	after a or g
c	1	1	1	after b or h
d	1	0	0	after c or e
e	0	0	0	after d or f
f	0	1	0	after e or a
g	0	0	1	after b or h
h	0	1	1	after g or c

⇒ Primitive flow table

	TC 00	01	11	10
a	-, -	f, -	(a), 0	b, -
b	g, -	-, -	c, -	(b), 1
c	-, -	h, -	(c), 1	d, -
d	e, -	-, -	a, -	(d), 0
e	(e), 0	f, -	-, -	d, -
f	e, -	(f), 0	a, -	-, -
g	(g), 1	h, -	-, -	b, -
h	g, -	(h), 1	c, -	-, -

③ reducing the number of states by merging the rows

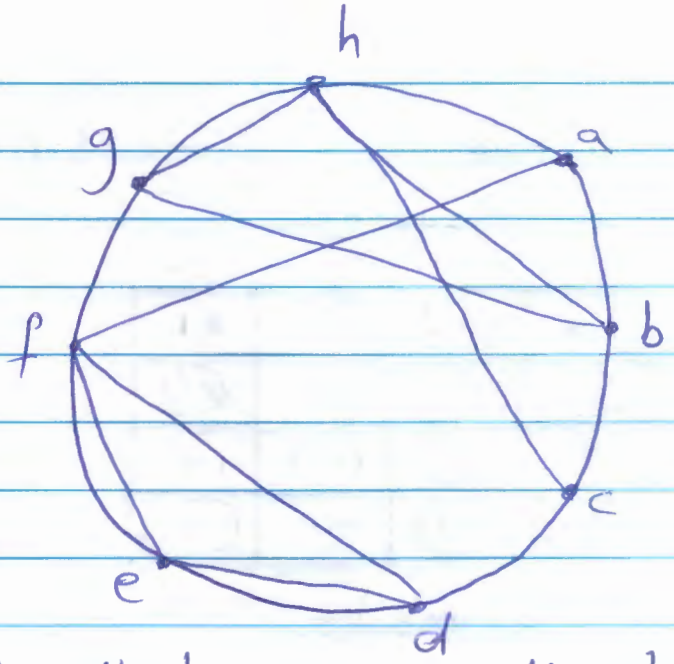
	a	b	c	d	e	f	g
b	a, c X						
c	X	b, d X					
d	b, d X	X	a, c X				
e	b, d X	e, g / b, d X	f, h X	✓			
f	✓	e, g / a, c X	a, c / f, h X	✓	✓		
g	f, h X	✓	b, d X	e, g / b, d X	X	e, g / f, h X	
h	f, h / a, c X	✓	✓	d, a / c, f X	e, g / f, h X	X	✓

~~pair~~ compatible pairs

(a, f), (b, g), (b, h), (c, h), (d, e), (d, f), (e, g), (g, h)

⇒ maximal compatibles using merger diagram

(d, e, f), (b, g, h),
(a, f), (e, h)



⇒ minimum collection that covers all states and class is

(a, f), (b, g, h), (c, h), (d, e, f)

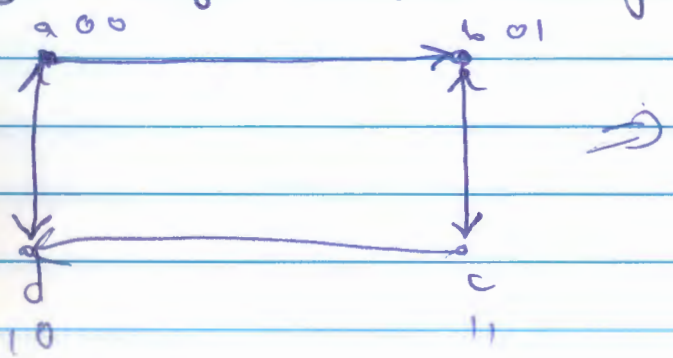
a b c d

TC	00	01	11	10
a, f	e, -	f, 0	a, 0	b, -
b, g, h	g, 1	h, 1	c, -	b, 1
c, h	g, -	h, 1	c, 1	d, -
d, e, f	e, 0	f, 0	a, -	d, 0

=

TC	00	01	11	10
a	d, -	a, 0	a, 0	b, -
b	b, 1	b, 1	c, -	b, 1
c	b, -	c, 1	c, 1	d, -
d	d, 0	d, 0	a, -	d, 0

(4) state assignment (race - free)
by using transition diagram



⇒

(5) get the transition table and output map

	y_1y_2 \ TC	00	01	11	10
a	00	10	(00)	(00)	01
b	01	(01)	(01)	11	(01)
c	11	01	(11)	(11)	10
d	10	(10)	(10)	00	(10)

Transition Table

y_1y_2 \ TC	00	01	11	10
00	0	0	0	X
01	1	1	1	1
11	1	1	1	X
10	0	0	0	0

$Q = y_2$ or $Q = Y_2$

y_1y_2 \ TC	00	01	11	10
00	1	0	0	0
01	0	0	1	0
11	0	1	1	0
10	1	1	0	1

$Y_1 = T'C'y_2' + T'Cy_1 + TCy_2 + TC'y_1$

y_1y_2 \ TC	00	01	11	10
00	0	0	0	1
01	1	1	1	1
11	1	1	1	0
10	0	0	0	0

$Y_2 = y_1'y_2 + T'y_2 + Cy_2 + TC'y_1'$

(Draw the logic Diagram)

Or By using SR latch

⟹

y_1y_2 \ TC	00	01	11	10
00	1	0	0	0
01	0	0	1	0
11	0	x	x	x
10	x	x	0	x

$S_1 = y_2 TC + y_2' T'C'$

y_1y_2 \ TC	00	01	11	10
00	0	x	x	x
01	x	x	0	x
11	1	0	0	0
10	0	0	1	0

$R_1 = y_2 T'C' + y_2' TC$

in the same way

$S_2 = y_1' TC'$

$R_2 = y_1 TC'$

Draw the logic diagram

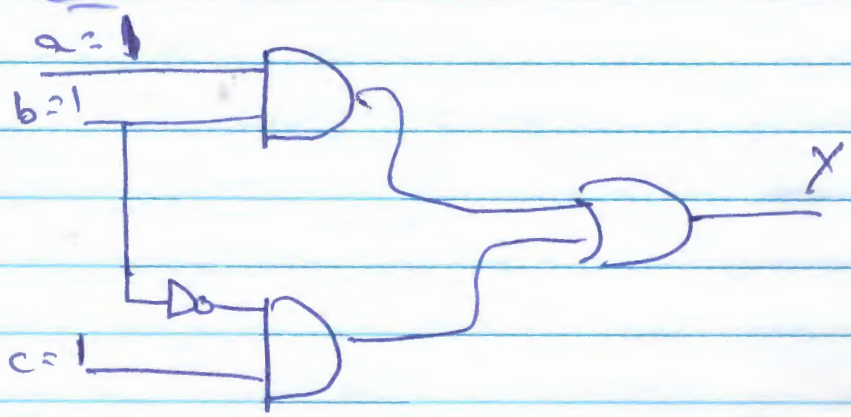
⊛ HAZARDS

Hazards are unwanted switching transients that may appear at the output of a circu because different paths exhibit different propagation delays.

⊛ Hazards in Combinational circuits

This occurs when a singl variable change prod a momentary output charge when no output should occur

Exc:

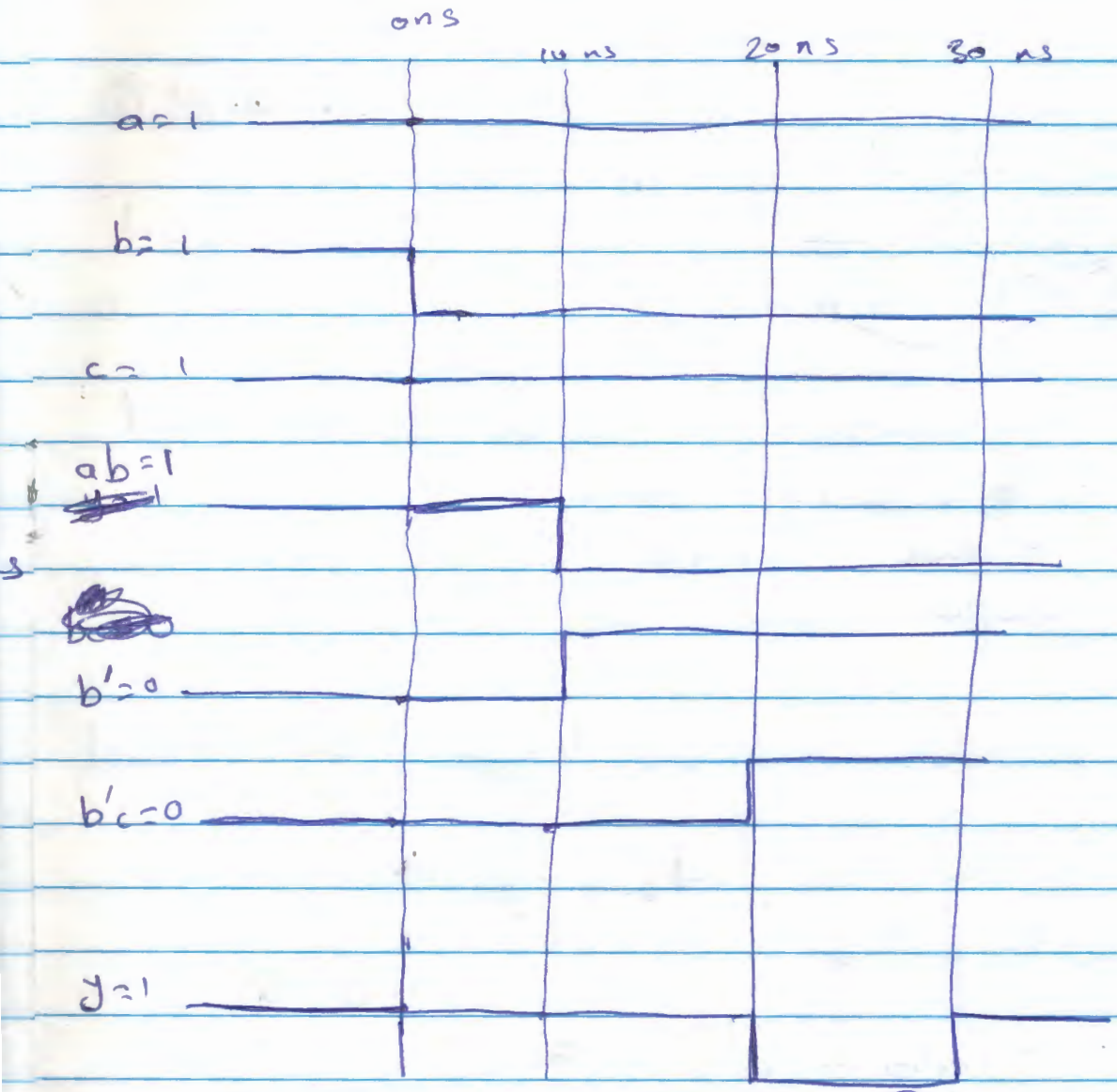


a=1, b=1, c=1 ⇒ y=1

if b changes to zero

⇒ a=1, b=0, c=1 ⇒ y=1

bab static hazard may occur because of differenb p-b^L delays

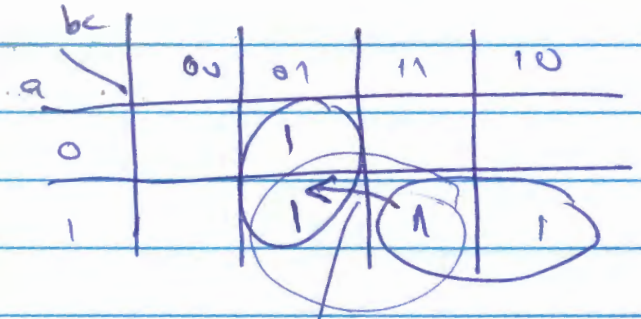


static hazard

- in this example we have

$$y = ab + b'c$$

⇒ k-map



$y = ab + b'c$

this change will cause static hazard

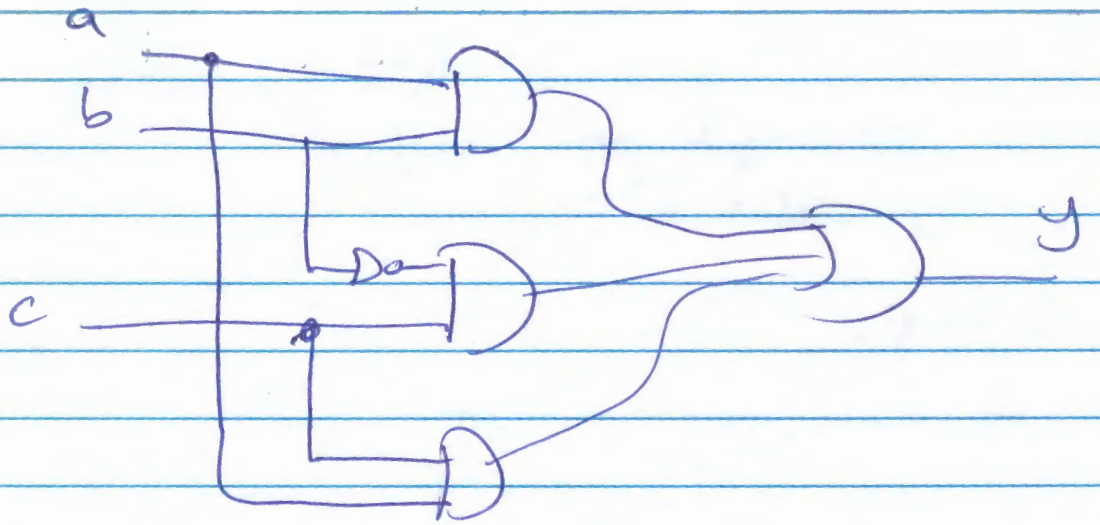
(a & c didn't change

b has changed ⇒ the circuit move from circle to another.

⇒ to remove this type of haz

include all the prime implicants

⇒ $y = ab + b'c + ac$

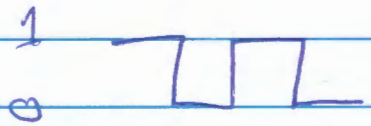


Hazard free circuit

Types of hazard

static 1-hazard

static-0-hazard



dynamic hazard

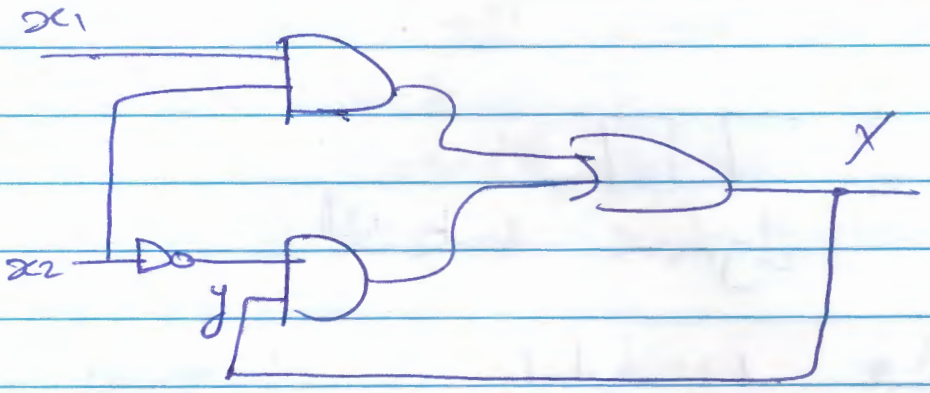
removal of static 1-hazard in (AND-OR, NAND-Circuits) will also remove 0-hazards and dynamic hazards.

(Problem in testing??).

⊛ Hazards in sequential circuits

- Hazards in asynchronous circuits may cause the circuit to go to the wrong stable state.

Ex.



$$Y = x_1x_2 + x_2'y$$

y \ x_1x_2	00	01	11	10
0	(0)	(0)	1	(0)
1	(1)	0	(1)	(1)

Transition table

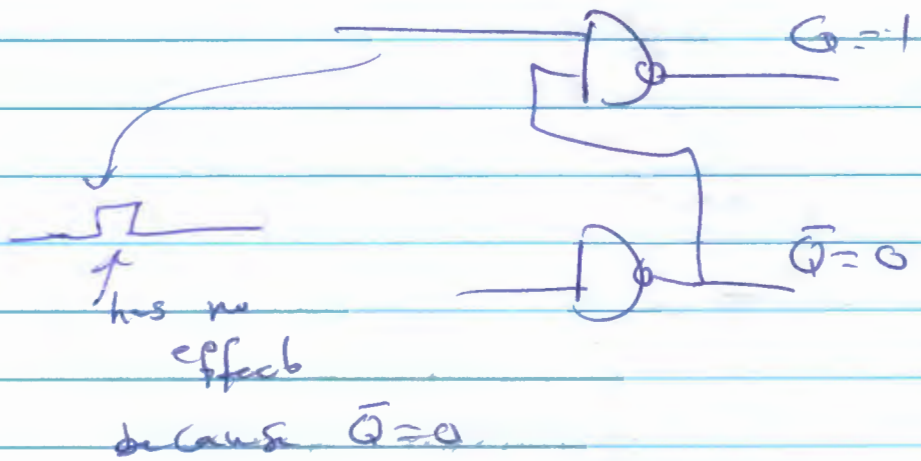
if we are in stable total state 111 then x_2 goes to 0 ⇒ next stable total state should be (y, x_1x_2) = 110

but if y changes to 0 (hazard) then total stable state is 010 error

⊛ Avoidance: the same way like combinational circuits: add a redundant terms for all prime implicants.

④ Implementation with SR latch

we can avoid hazard if we implement the circuits with SR latches



note about implementation with SR latch

~~NOR~~

- SR with 2 nor gates

$$y = S + R'y$$

- SR with 2 nand gate

$$y = S' + Ry$$

Excitation table

this is for SR with nor gates

Q(t)	Q(t+1)	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

for example if we get

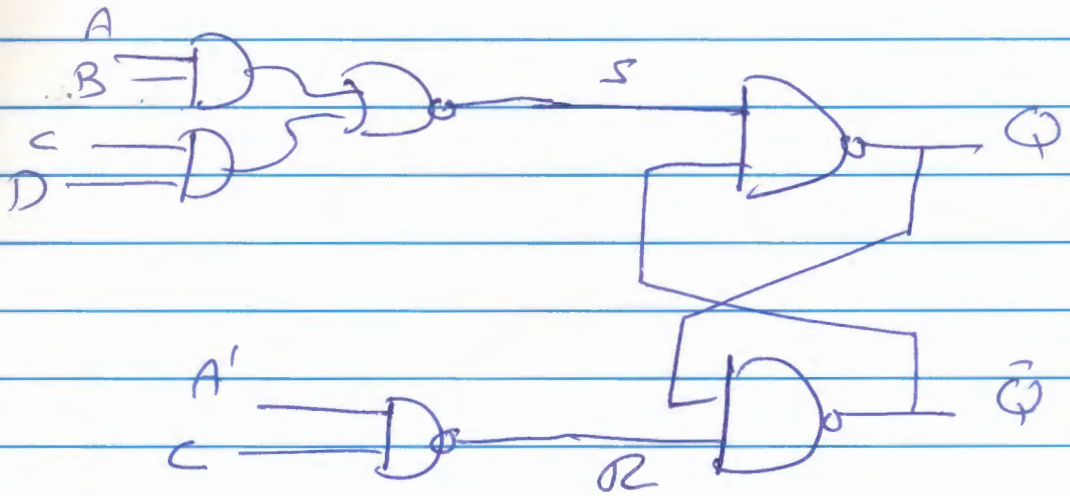
$$S = AB + CD$$

$$R = A'C$$

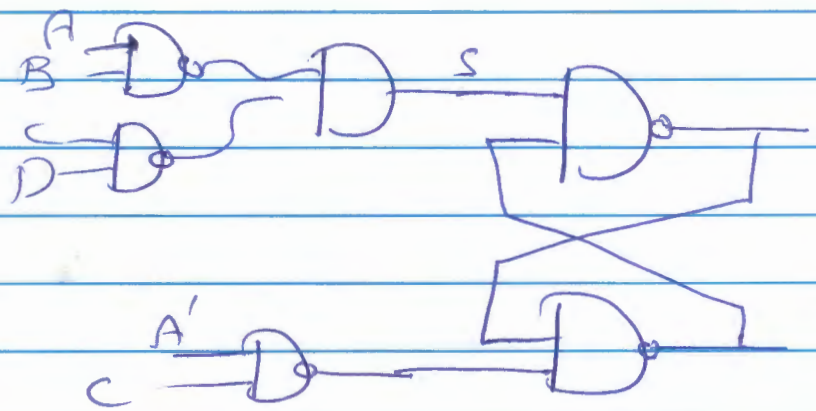
From this excitation table and we want to use SR with nand gates

$$\Rightarrow S_{nand} = S'_{nor} = (AB + CD)'$$

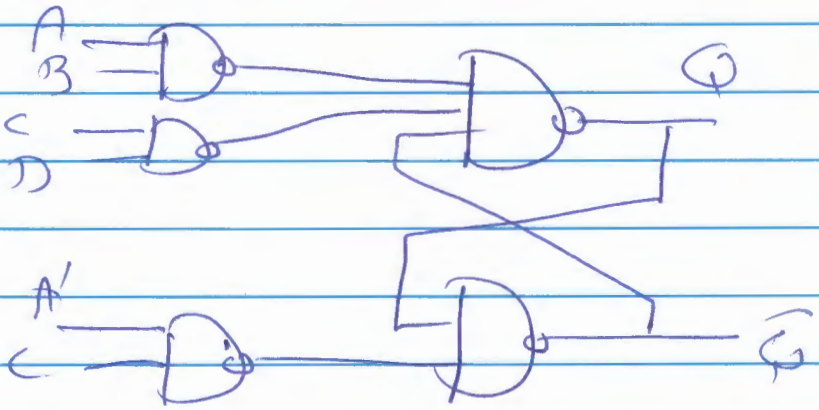
$$R_{nand} = R'_{nor} = (A'C)'$$


A
B
C
D
S
Q
A'
C
R
$\bar{Q}$

=


A
B
C
D
S
A'
C

=


A
B
C
D
Q
A'
C
$\bar{Q}$