\# of inputs $= 2^n = 2^3 = 8$

\# of outputs $= 3$

CRN 22065

Riselda Kodra 040180935

Pg 1

1

x_7	x_6	x_5	x_4	x_3	x_2	x_1	x_0	y_2	y_1	y_0
1	0	0	0	0	0	0	0	1	1	1
0	1	0	0	0	0	0	0	1	1	0
0	0	1	0	0	0	0	0	1	0	1
0	0	0	1	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	1	0	0	0

$y_2 = \sum m(128, 64, 32, 16)$

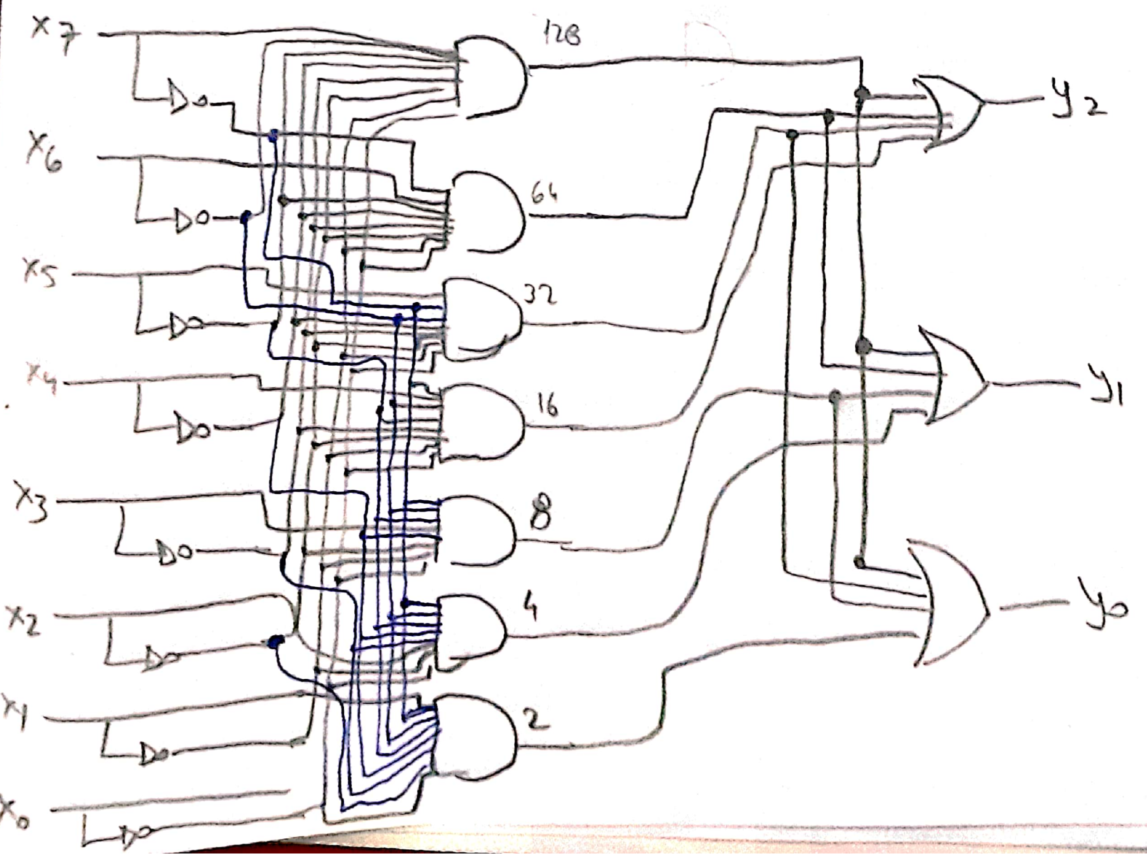
$y_1 = \sum m(128, 64, 8, 4)$

$y_0 = \sum m(128, 32, 8, 2)$

$y_2 = x_7\overline{x_6}\,\overline{x_5}\,\overline{x_4}\,\overline{x_3}\,\overline{x_2}\,\overline{x_1}\,\overline{x_0} + \overline{x_7}x_6\overline{x_5}\,\overline{x_4}\,\overline{x_3}\,\overline{x_2}\,\overline{x_1}\,\overline{x_0}$
$+ \overline{x_7}\,\overline{x_6}x_5\overline{x_4}\,\overline{x_3}\,\overline{x_2}\,\overline{x_1}\,\overline{x_0} + \overline{x_7}\,\overline{x_6}\,\overline{x_5}x_4\overline{x_3}\,\overline{x_2}\,\overline{x_1}\,\overline{x_0}$

$y_1 = x_7\overline{x_6}\,\overline{x_5}\,\overline{x_4}\,\overline{x_3}\,\overline{x_2}\,\overline{x_1}\,\overline{x_0} + \overline{x_7}x_6\overline{x_5}\,\overline{x_4}\,\overline{x_3}\,\overline{x_2}\,\overline{x_1}\,\overline{x_0} + \overline{x_7}\,\overline{x_6}\,\overline{x_5}\,\overline{x_4}x_3\overline{x_2}\,\overline{x_1}\,\overline{x_0}$
$+ \overline{x_7}\,\overline{x_6}\,\overline{x_5}\,\overline{x_4}\,\overline{x_3}x_2\overline{x_1}\,\overline{x_0}$

$y_0 = x_7\overline{x_6}\,\overline{x_5}\,\overline{x_4}\,\overline{x_3}\,\overline{x_2}\,\overline{x_1}\,\overline{x_0} + \overline{x_7}\,\overline{x_6}x_5\overline{x_4}\,\overline{x_3}\,\overline{x_2}\,\overline{x_1}\,\overline{x_0} + \overline{x_7}\,\overline{x_6}\,\overline{x_5}\,\overline{x_4}x_3\overline{x_2}\,\overline{x_1}\,\overline{x_0}$
$+ \overline{x_7}\,\overline{x_6}\,\overline{x_5}\,\overline{x_4}\,\overline{x_3}\,\overline{x_2}x_1\overline{x_0}$



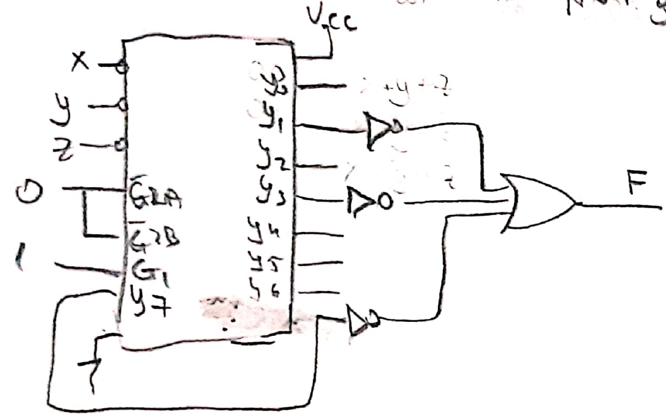
CRN 22665

Riselda Kodra
040180935

$F = x'y'z + yz$

$F = x'y'z + (x+x')yz = x'y'z + xyz + x'yz$

active 0 decoder ⇒ outputs are maxterms
to have minterms we will use NOT gates at output



$y_0 = x + y + z$
$y_1 = x + y + \bar{z}$
$y_2 = x + \bar{y} + z$
$y_3 = x + \bar{y} + \bar{z}$
$y_4 = \bar{x} + y + z$
$y_5 = \bar{x} + y + \bar{z}$
$y_6 = \bar{x} + \bar{y} + z$
$y_7 = \bar{x} + \bar{y} + \bar{z}$

Riselda Kodra
040180935
CRN
22665

EXP 4

a)

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$C = \sum m(3)$

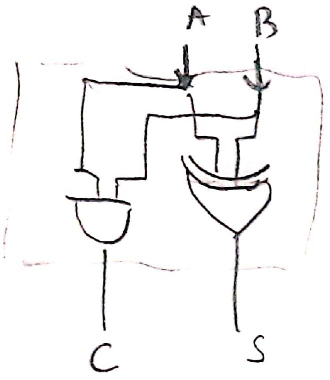
$C = AB$

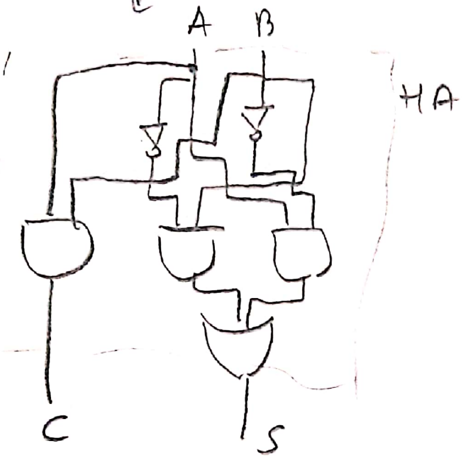
$S = \sum m(1,2)$

$S = \bar{A}B + A\bar{B}$ (SOP representation)

$S = A \oplus B$ (using less gates)

b)

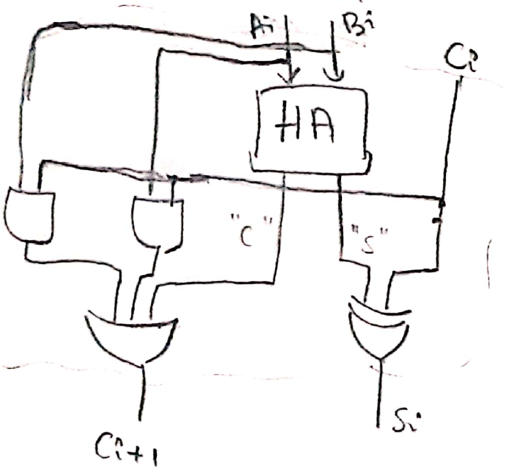




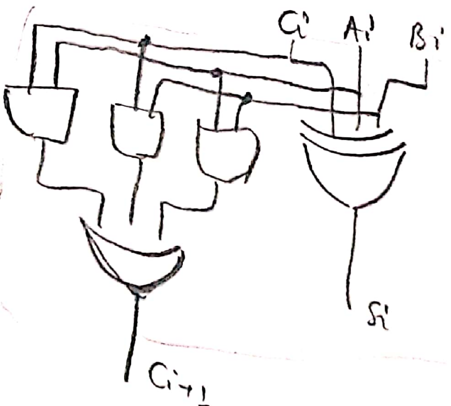
c) Full Adder

$S_i = A_i \oplus B_i \oplus C_i$

$C_{i+1} = A_i B_i + A_i C_i + B_i C_i$

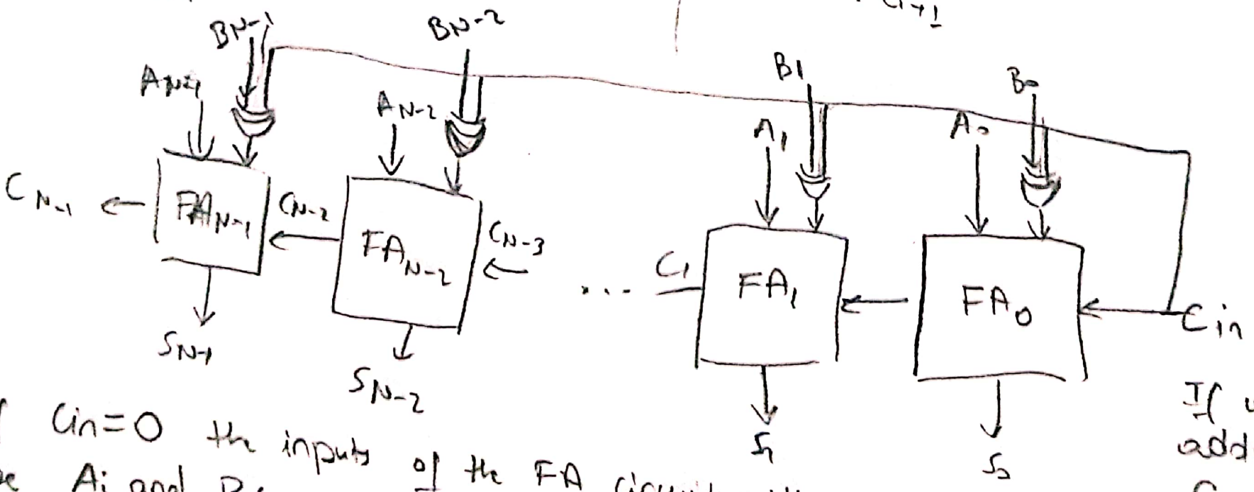


FA using HA



FA (not using HA) directly

d)



If we want adder Cin=0
If we want subtractor Cin=1

If Cin=0 the inputs of the FA circuit will be Ai and Bi ⇒ we will have a basic adder

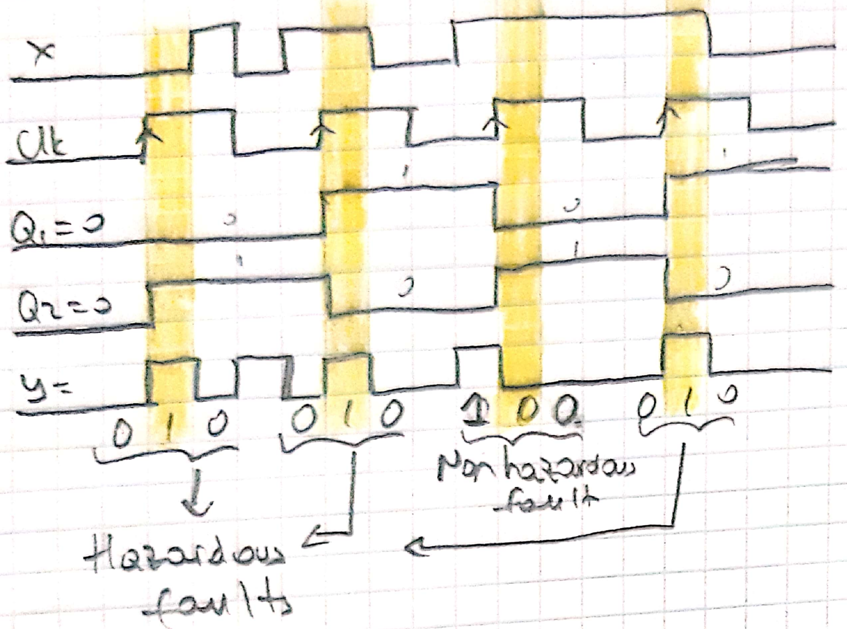
If Cin=1 the inputs will be Ai and $\overline{B_i}$ so we will have a subtractor ↳ because of XOR gate

If $B_i = 1$ $1 \oplus 1 = 0 = \bar{1} = \overline{B_i}$

$B_i = 0$ $0 \oplus 1 = 1 = \bar{0} = \overline{B_i}$

Rivaldo Kodra
040180935
CRN 22665
Pg 4

a)

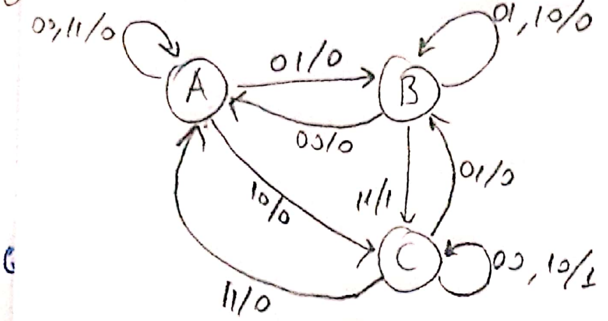


b) This circuit is a Mealy machine since we have such faults at output and output value depends on both the current state and current input values

EXP 7

EXP 6

Reselda Kodra
040180935
CRN 22665
Pg 5



a) 3 states

⇓

2 bits are enough to encode them

A ⇒ 00

B ⇒ 01

C ⇒ 10

JK FF

$Q = Jq' + K'q$

00 → 0k

11 → k0

01 → 1k

10 → k1

b)

x	y	q_1	q_2	Q_1	Q_2	J_1	K_1	J_2	K_2	z
0	0	0	0	0	0	0	k	0	k	0
0	0	0	1	0	0	0	k	k	1	0
0	0	1	0	1	0	k	0	0	k	1
0	0	1	1	k	k	k	k	k	k	k
0	1	0	0	0	1	0	k	1	k	0
0	1	0	1	0	1	0	k	k	0	0
0	1	1	0	0	1	k	1	1	k	0
0	1	1	1	k	k	k	k	k	k	k
1	0	0	0	1	0	1	k	0	k	0
1	0	0	1	0	1	0	k	k	0	0
1	0	1	0	1	0	k	0	0	k	1
1	0	1	1	k	k	k	k	k	k	k
1	1	0	0	0	0	0	k	0	k	0
1	1	0	1	1	0	1	k	k	1	1
1	1	1	0	0	0	k	1	0	k	0
1	1	1	1	k	k	k	k	k	k	k

z

xy \ q_1q_2	00	01	11	10
00	0	0	k	1
01	0	0	k	0
11	0	1	k	0
10	0	0	k	1

→ $\bar{y}q_1$

→ xyq_2

$z = \bar{y}q_1 + xyq_2$

c)

J_1

xy \ q_1q_2	00	01	11	10
00	0	0	k	k
01	0	0	k	k
11	0	1	k	k
10	1	0	k	k

$J_1 = x\bar{y}\bar{q_2} + xyq_2$

K_1

xy \ q_1q_2	00	01	11	10
00	k	k	k	0
01	k	k	k	1
11	k	k	k	1
10	k	k	k	0

$K_1 = y$

J_2

xy \ q_1q_2	00	01	11	10
00	0	k	k	0
01	1	k	k	1
11	0	k	k	0
10	0	k	k	0

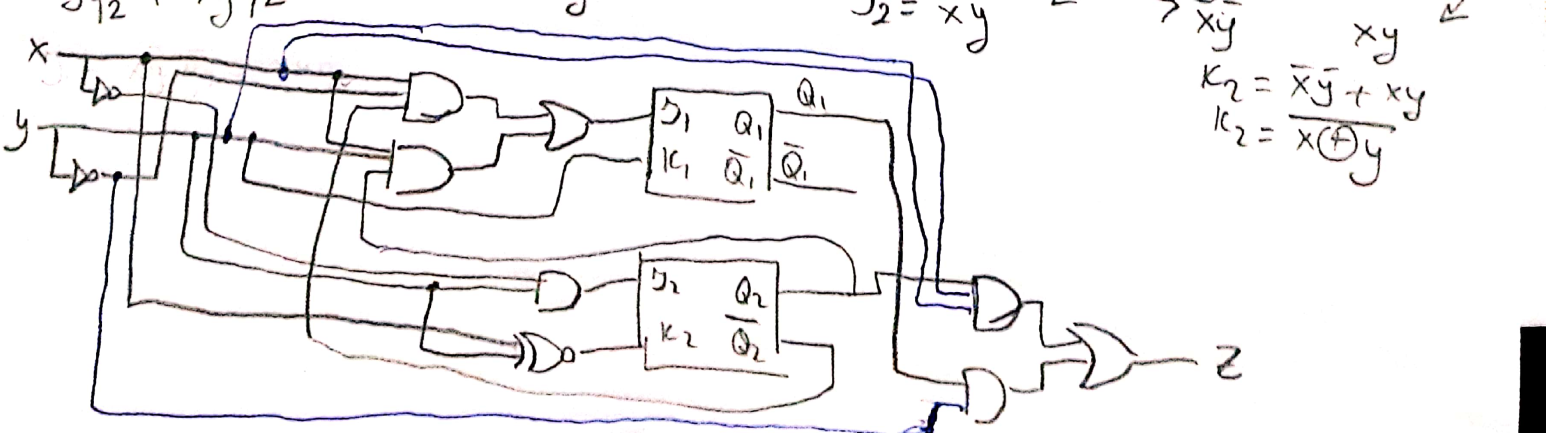
$J_2 = \bar{x}y$

xy \ q_1q_2	00	01	11	10
00	k	1	k	k
01	k	0	k	k
11	k	1	k	k
10	k	0	k	k

$\bar{x}\bar{y}$ $\quad xy$

$K_2 = \bar{x}\bar{y} + xy$

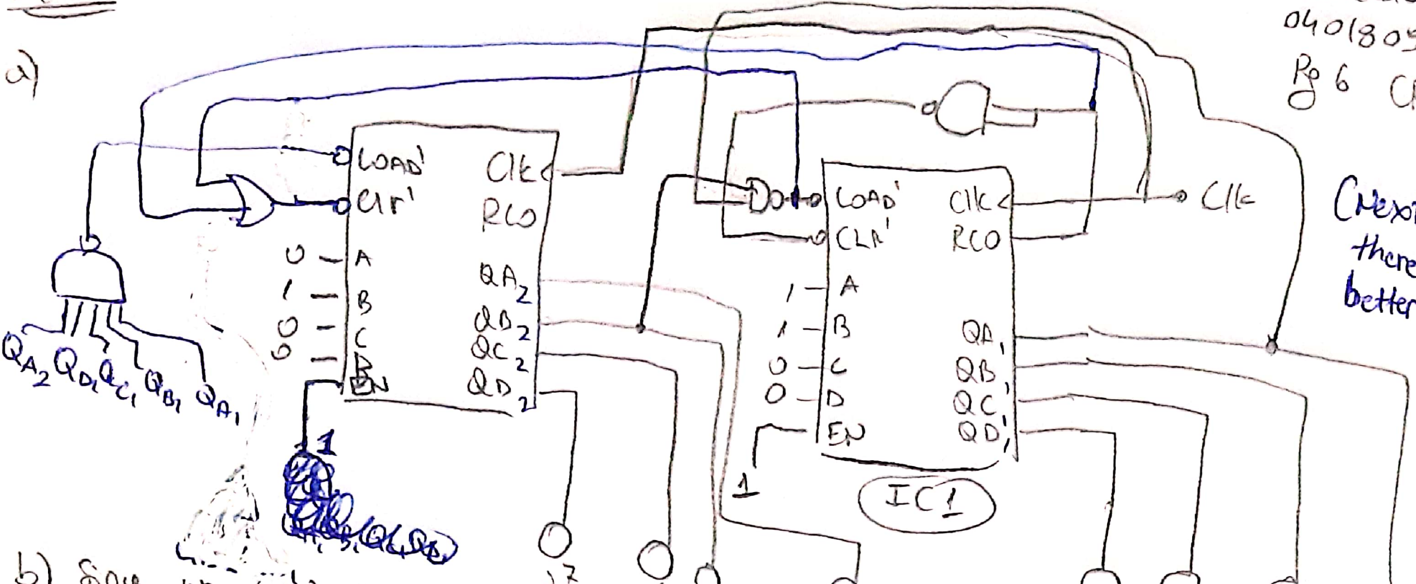
$K_2 = \overline{x \oplus y}$



EXP 7

Riselda Kodra
040180935 Rg
Pg 6 CRN 22665

a)



(next page there is a better figure)

b) Since we want the counter to start counting from 3 we first make the load at IC1 as 3. So the IC1 will be counting from 3 to 15. When it reaches 15 it should restart from 0. Clr is an active-0 input so we should connect $\overline{RCO}$ at the input of Clr. To get the NOT of RCO I'm using a NAND gate

After 15 we will need the 2nd IC to represent our numbers in binary.

Since the counter will be from 3 to 33 we will be needing only QA and QB of 2nd IC to represent our numbers.

When IC_1 reaches 15 it should "notify" IC_2 to make QA = 1 and it should stay like that for 15 Clk pulses. (We use Clr to make the IC_2 restart each time)
Then QA should become 0 and QB should become 1 and stay like that for 2 pulses where IC_1 counts from 0 to 1

Unless RCO_1 is 1 then clr should be active

Then IC_1 should restart again from 3. (We connect at the load of IC_1 and clr of IC_2 $\overline{QB_2\, QA_1}$

To sum up

IC_1 counts from 3 to 15
When it reaches 15 using AND gate it "tells" IC_2 to start counting
QA_2 stays 1 for 15 pulses since I'm making Clr_2 command active by applying RCO_1 there.
When counter reaches 31 IC_2 loads 0010 and IC_1 counts from 0 to 1
We reach 33 ($QB_2 = 1$, $QA_1 = 1$) we load 3 at IC_1 and restart from there.

EXP 7

Riselda Kodra
040180935 Rg

EXP 7 (pg 7)

3 to 33 counter

Riselda Kodra
040180935 Rg
CRN 22665

Circuit