Subject:
Date

- به نام خدا
- پاسخنامه تمرین سری دوم مدار منطقی

- سؤال ①

الف) این مدار از دو Comparator و Full Adder تشکیل شده است.
در ابتدا در مقایسه کننده ورودی ۴ بیتی A با عدد B (که حاوی مقدار ۹ باینری است) مقایسه می‌شود.
اگر مقدار A>B بزرگ‌تر باشد (A>9) خروجی ۱ خواهد شد.
در مرحله دوم که با یک جمع‌کننده مواجه‌ایم با توجه به خروجی A>B مقایسه‌کننده ورودی A با عدد 0000 (0) و یا عدد 0110 (6) جمع خواهد شد.
(توجه شود که carry = 0 است)

اگر A>9 : A + 6
اگر A≤9 : A + 0 = A (آشکار می‌باشد)

با توجه به توضیحات جدول ضابطه‌ای که در بالا برای A نوشتیم می‌توانیم وضعیت carry out (C) و f را مشخص کنیم.

		C	f_3 f_2 f_1 f_0
اگر A>9	A + 6 > 15	1	____
اگر A≤9	A ≤ 9	0	____

حاصل f نیز به سادگی از معادله بالا به دست می‌آید.

کاربرد این مدار تبدیل عدد باینری به خاکستر BCD است.

Subject:
Date

ب) برای مشخص شدن تابع نهایی خروجی کافیست در هر مرحله بررسی کنیم چه تغییراتی بر A اعمال می‌شود.

Binary Adder ① : $A + 5 + 0 = A + 5$ (ورودی دوم: 5، carry: 0)

در مقایسه کننده دوم $A < 4$ باشد ← خروجی 1 و در غیر این حالت یعنی $A \geq 4$ خروجی 0 خواهد بود.

Binary Adder ② : اگر $A < 4$: $(A + 5) + 3 + 0$ (ورودی دوم: 3، carry: 0)

اگر $A \geq 4$: $(A + 5) + 2 + 0$ (ورودی دوم: 2، carry: 0)

$$g \begin{cases} A < 4 : A + 8 \\ A \geq 4 : A + 7 \end{cases}$$

این تابع دو ضابطه‌ای حاصل جمع بعد از جمع کننده دوم است.

حال به MUX 2:1 می‌رسیم ورودی‌های اول $A + 5$ است و ورودی‌های دوم خروجی جمع بعد از جمع کننده دوم است. اینکه کدام یک از این ورودی‌ها انتخاب شود به select بستگی دارد.

select : $A > 4$

نکته دیگری که باید توجه داشت این است که select به هر صورت انتخاب شود خروجی‌ها همواره به صورت دسته‌ای خواهد بود یعنی یا کل ورودی‌های I_0 یا کل ورودی‌های I_1

	خروجی (f)
if $A \leq 4$:	A
if $A > 4$:	g

Subject:
Date

• سؤال ②

الف)

a	b	c	d	x	y	z	w
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
0	1	0	1	0	1	1	0
0	1	0	0	0	1	1	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	0	0	1	1	1	1	0
1	0	0	0	1	1	1	1

$x = a$

$y = \sum(4, 5, 6, 7, 8, 9, 10, 11)$

$z = \sum(2, 3, 4, 5, 8, 9, 14, 15)$

$w = \sum(1, 2, 4, 7, 8, 11, 13, 14)$

CamScanner

Subject:
Date

select : c, d

	I_0	I_1	I_2	I_3	
$a'b'$	0	1	2	3	+4
$a'b$	(4)	(5)	(6)	(7)	+8
ab	12	13	14	15	
ab'	(8)	(9)	(10)	(11)	-4
	$a \oplus b$	$a \oplus b$	$a \oplus b$	$a \oplus b$	

$ab' + a'b = a \oplus b$

	I_0	I_1	I_2	I_3
$a'b'$	0	1	(2)	(3)
$a'b$	(4)	(5)	6	7
ab	12	13	(14)	(15)
ab'	(8)	(9)	10	11
	$a \oplus b$	$a \oplus b$	$a \odot b$	$a \odot b$

$a'b' + ab = a \odot b$

	I_0	I_1	I_2	I_3
$a'b'$	0	(1)	(2)	3
$a'b$	(4)	5	6	(7)
ab	12	(13)	(14)	15
ab'	(8)	9	10	(11)
	$a \oplus b$	$a \odot b$	$a \odot b$	$a \oplus b$

پیاده سازی های مورد نیاز با MUX ۲x۱ :

$a \oplus b$

$a \odot b$

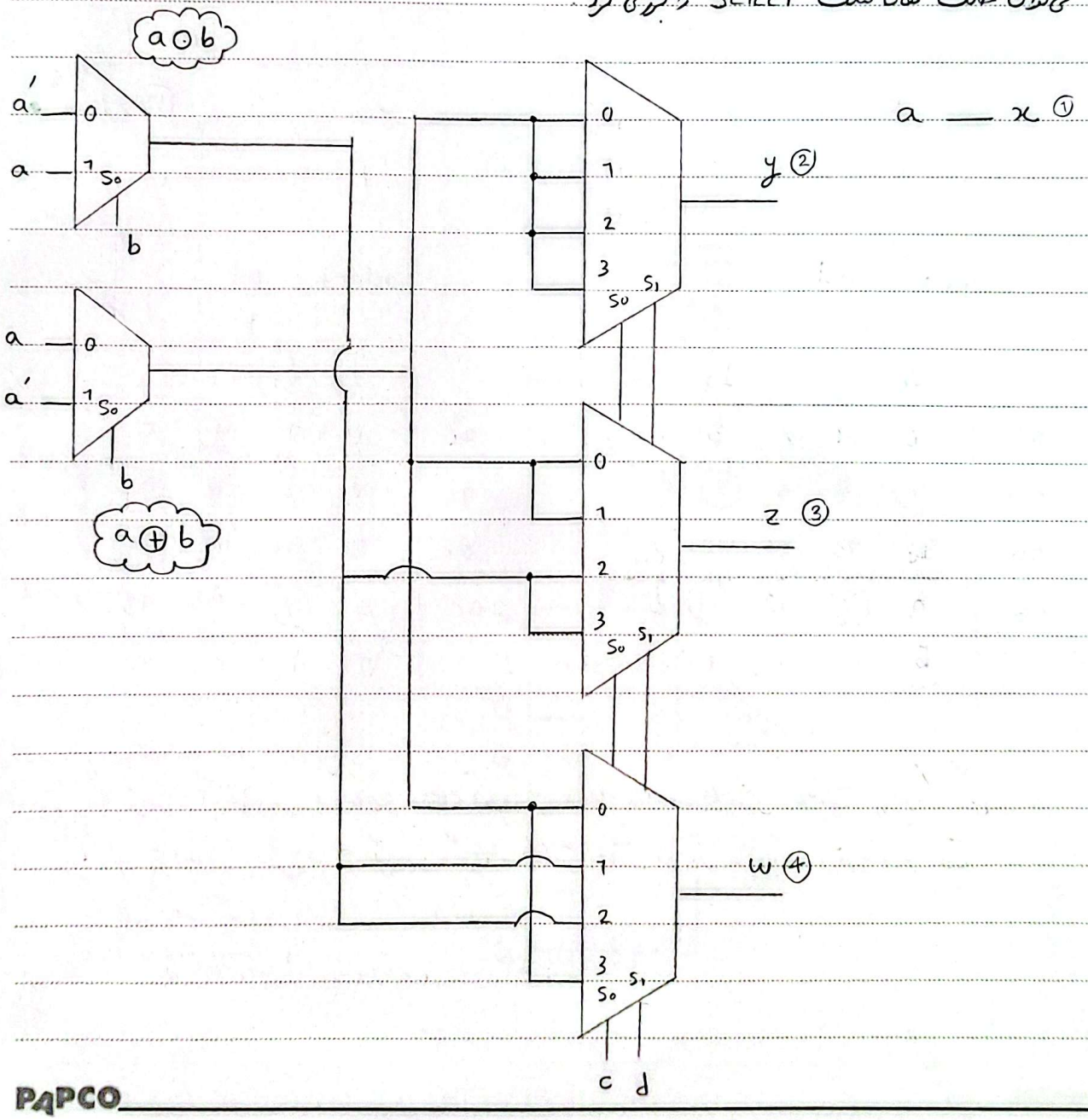
PAPCO

Subject:
Date

a⊙b	I_0	I_1
a′	⓪	1
a	2	③
	a′	a

a⊕b	I_0	I_1
a′	0	①
a	②	3
	a	a′

select های ترانه متفاوت باشد و با توجه به این موضوع جداول تغییر خواهد کرد. برای بهینه ترین حالت می‌توان حالت های مختلف select را بررسی کرد.



PAPCO

Subject:
Date:

ب) در طراحی ROM اگر یکی از خروجی ها همواره $\underline{0}$ یا $\underline{1}$ باشد یا دقیقاً مثل یکی از ورودی ها باشد ← از جدول حقیقت ROM کنار گذاشته می‌شود تا اندازه ROM کاهش پیدا کند.

ابعاد ROM : تعداد خروجی‌ها × $2^{\text{تعداد ورودی‌ها}}$

ابعاد ROM : $3 \times 2^4 = 48$

• سؤال ③

الف)

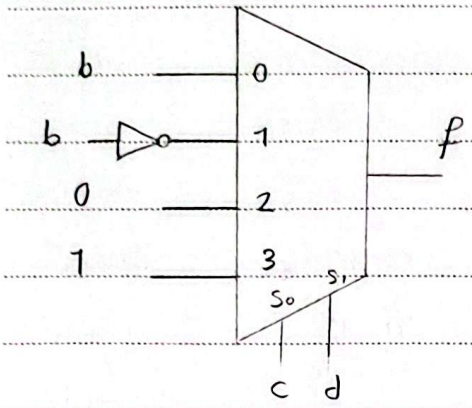
select : cd

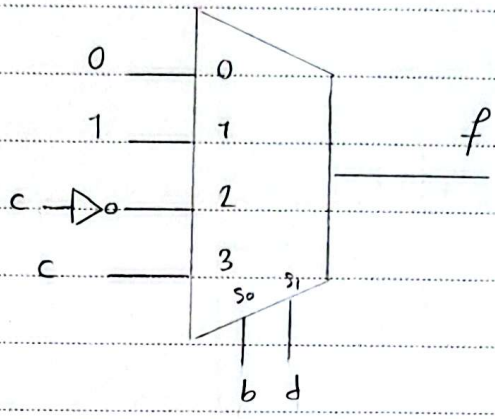
	I_0	I_1	I_2	I_3	
a'b'	0	(1)	2	(3)	+4
a'b	(4)	5	6	(7)	+8
ab	(12)	13	14	(15)	−8
ab'	8	(9)	10	(11)	
	b	b'	0	1	

select : bd

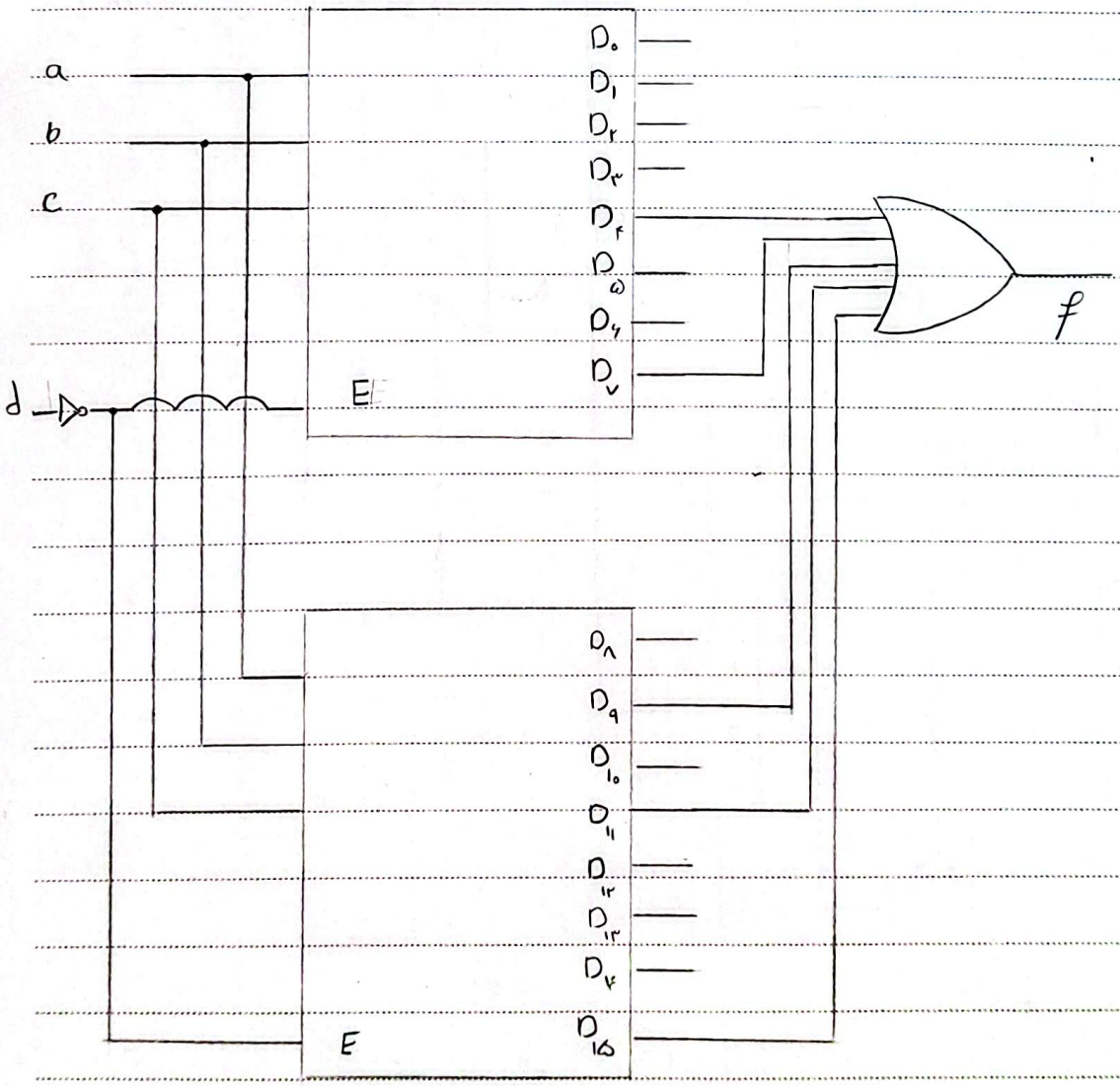
	I_0	I_1	I_2	I_3	
a'c'	0	(1)	(4)	5	+2
a'c	2	(3)	6	(7)	+8
ac	10	(11)	14	(15)	−2
ac'	8	(9)	(12)	13	
	0	1	c'	c	

با توجه به انتخاب select های مختلف روش های ساده سازی بدرستی می آید که هر دو درست اند و ممکن است یکی از دیگری ساده تر باشد اما باید به جایگاه و ارزش هر بیت توجه داشت.


b
b
0
1
0
1
2
3
S0
S1
f
c
d

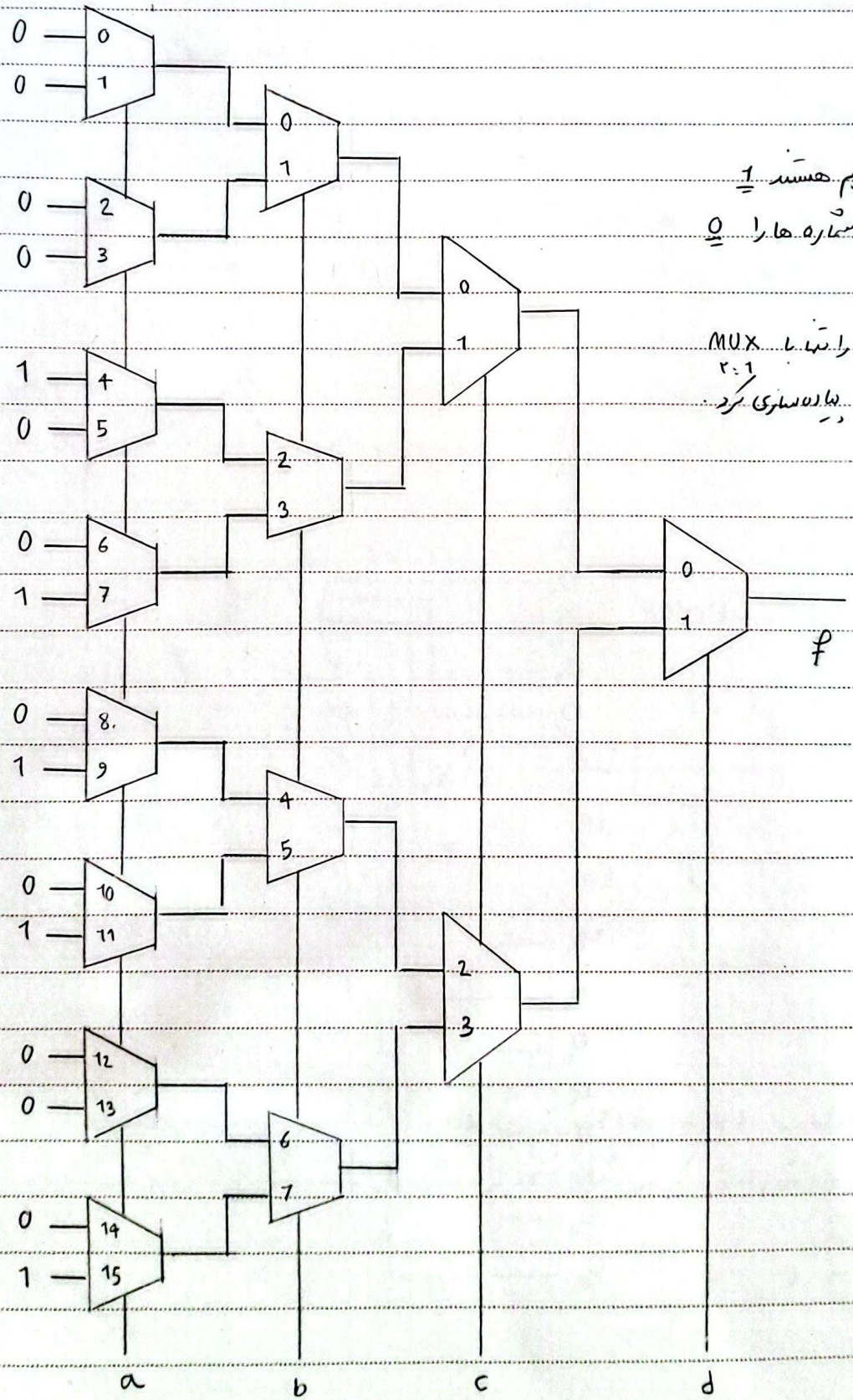

0
1
c
c
0
1
2
3
S0
S1
f
b
d

(e


a
b
c
d
E
E
D0
D1
D2
D3
D4
D5
D6
D7
D8
D9
D10
D11
D12
D13
D14
D15
f

Subject:
Date

ب)



شماره هایی که بیشترین هستند 1
می دهیم و بقیه شماره ها را 0
می دهیم.
می توان هر تابعی را تنها با MUX ۲:۱
و بدون گیت کمکی پیاده سازی کرد.

Subject:
Date

• سؤال ④

① ابتدا جدول حالت را برای هر ۳ حالت تابع خروجی رسم می‌کنیم:

A		B				
A_0	A_1	B_0	B_1	f_1	f_2	f_3
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	0	0	0
0	1	0	1	0	1	0
0	1	1	0	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	0	0
1	0	0	1	0	1	0
1	0	1	0	1	0	0
1	0	1	1	0	0	0
1	1	0	0	0	0	0
1	1	0	1	0	1	1
1	1	1	0	1	1	0
1	1	1	1	1	1	0

$f_1 = \Sigma(7, 10, 14, 15)$

$f_2 = \Sigma(2, 3, 5, 6, 9, 13, 14, 15)$

$f_3 = \Sigma(1, 3, 6, 13)$

Subject:
Date

۲) رسم جدول MUX برای هر سه تابع خروجی:

در اینجا select را برای هر ۳ یکسان و $A_1 B_0 B_1$ انتخاب کردیم.

f_1:

	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
A_0'	0	1	2	3	4	5	6	(7)
A_0	8	9	(10)	11	12	13	(14)	(15)
	0	0	A_0	0	0	0	A_0	1

(+8)

f_2:

	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
A_0'	0	1	(2)	(3)	4	(5)	(6)	7
A_0	8	(9)	10	11	12	(13)	(14)	(15)
	0	A_0	A_0'	A_0'	0	1	1	A_0

(+8)

f_3:

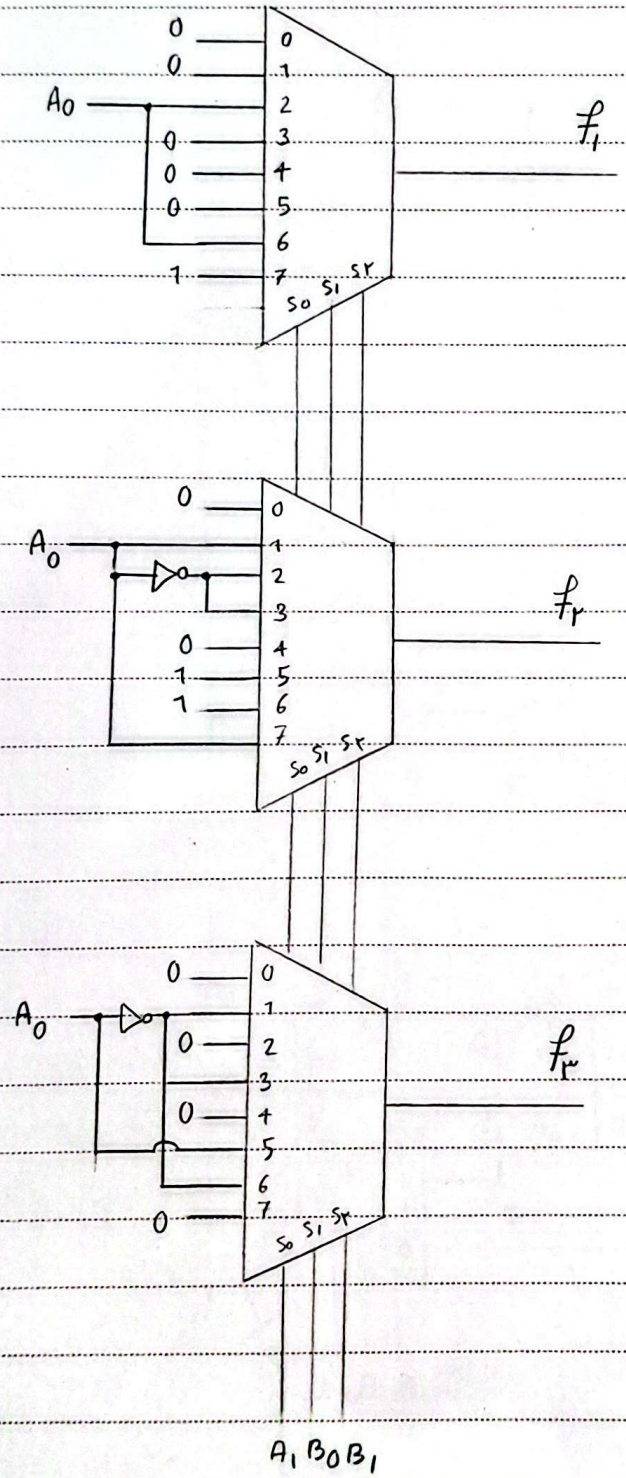
	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
A_0'	0	(1)	2	(3)	4	5	(6)	7
A_0	8	9	10	11	12	(13)	14	15
	0	A_0'	0	A_0'	0	A_0	A_0'	0

(+8)

۴) رسم توابع

CamScanner

Subject:
Date



خروجی از دو کنار هم قرار گرفتن ۳ تابع f_1, f_2, f_3 تشکیل می‌شود.

Subject:
Date

• سؤال ⑤

xy \ z	0	1
00	0	(x)
01	(1	0
11	1)	0
10	0	(1)

$A = yz' + y'z$

xy \ z	0	1
00	(1)	x
01	0	(1
11	0	1)
10	(1)	0

$A' = yz + y'z'$

xy \ z	0	1
00	0	(1
01	(1	x)
11	0	1
10	0	1)

$B = x'y + z$

xy \ z	0	1
00	(1)	0
01	0	x
11	(1	0
10	1)	0

$B' = xz' + y'z'$

xy \ z	0	1
00	0	(1
01	(1	1)
11	0	0
10	x	0

$C = x'y + x'z$

xy \ z	0	1
00	(1)	0
01	0	0
11	(1	1
10	x)	1)

$C' = x + y'z'$

CamScanner

Subject:
Date

xy \ z	0	1
00	1	0
01	1	x
11	1	x
10	0	0

$D = x'z' + y$

xy \ z	0	1
00	0	1
01	0	x
11	0	x
10	1	1

$D' = z + xy'$

3	2	1		1	2	3
			$A = yz' + y'z$	$A' = yz + y'z'$	☑	☑	☑
☑			$B = x'y + z$	$B' = xz' + y'z'$	☑	☑	
			$C = x'y + x'z$	$C' = x + y'z'$	☑	☑	☑
		☑	$D = x'z' + y$	$D' = z + xy'$		☑	☑

① حالت :

	x	y	z	A	B	C	D
yz	–	1	1	1	–	–	–
y'z'	–	0	0	1	1	1	–
xz'	1	–	0	–	1	–	–
x	1	–	–	–	–	1	–
x'z'	0	–	0	–	–	–	1
y	–	1	–	–	–	–	1
				C	C	C	T

PAPCO

Subject:
Date

حالت ② :

	x	y	z	A	B	C	D
yz	-	1	1	1	-	-	-
y'z'	-	0	0	1	1	1	-
xz'	1	-	0	-	1	-	-
x	1	-	-	-	-	1	-
z	-	-	1	-	-	-	1
xy'	1	0	-	-	-	-	1
				C	C	C	C

حالت ③ :

	x	y	z	A	B	C	D
yz	-	1	1	1	-	-	-
y'z'	-	0	0	1	-	1	-
x'y	0	1	-	-	1	-	-
z	-	-	1	-	1	-	1
x	1	-	-	-	-	1	-
xy'	1	0	-	-	-	-	1
				C	T	C	C

هرکدام از این ۳ حالت از نظر تعداد گیت های AND با هم برابرند و مساوی با ۴ است. هرکدام از این وضعیت ها بهینه ترین حالت برنامه ریزی PLA است.