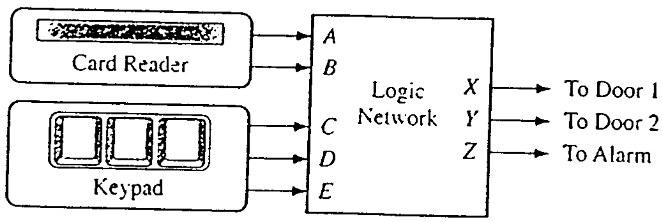
EEN 220	SPRING 2001	NDU ECE Dept.
Logic Design	Test 2	S.C. Elmurr

Note 1: Open book, open Notes, Closed Neighbours
Note 2: Show all work in order to receive full credit.
Note 3: Start each problem on a new page.

1:25 Pts A simple security system for two doors consists of a card reader and a keypad.



A person may open a particular door if he or she has a card containing the corresponding code, and enters an authorized keypad code for that card. The outputs from the card reader are as follows:

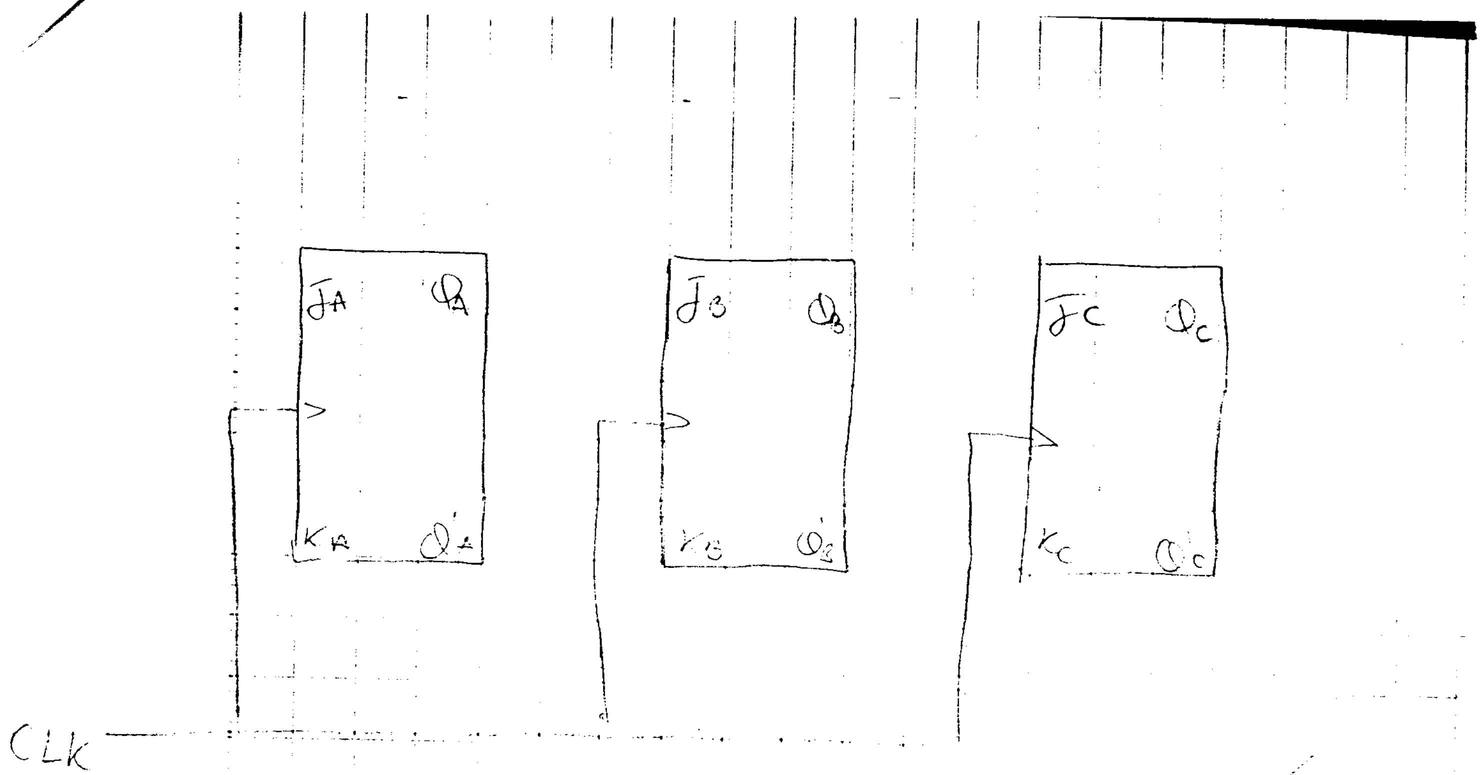
	$\underline{A}$	$\underline{B}$
No card inserted	0	0
Valid code for door 1	0	1
Valid code for door 2	1	1
Invalid card code	1	0

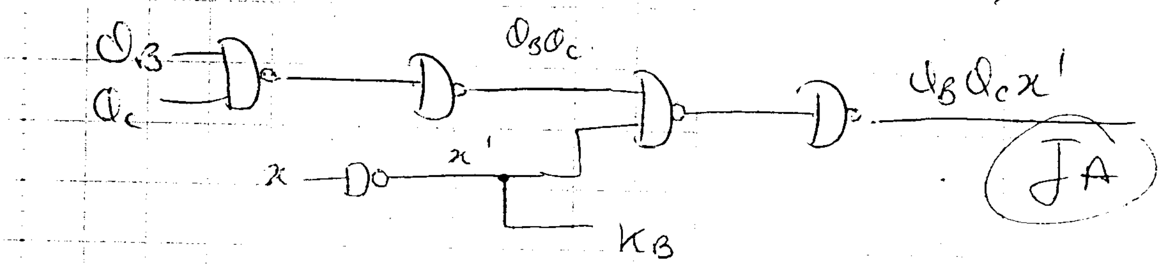
To unlock a door, a person must hold down the proper keys on the keypad and then insert the card in the reader. The authorized keypad codes for door 1 are 101 and 110, and the authorized keypad codes for door 2 are 101 and 011. If the card has an invalid code or if the wrong keypad code is entered, the alarm will ring when the card is inserted. If the correct keypad code is entered, the corresponding door will be unlocked when the card is inserted.
Design the logic network for this simple security system. Your network's inputs will consist of a card code AB, and a keypad code CDE. The network will have three outputs XYZ (if X or Y = 1, door 1 or 2 will be opened; if Z= 1, the alarm will sound). Design your network and implement in NAND/NAND form.

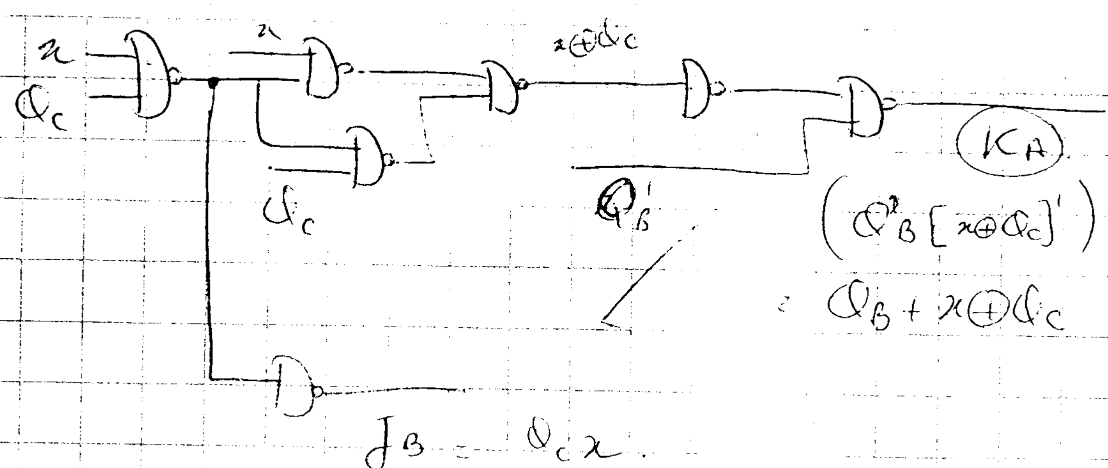
2. 20 Pts. Realize the function

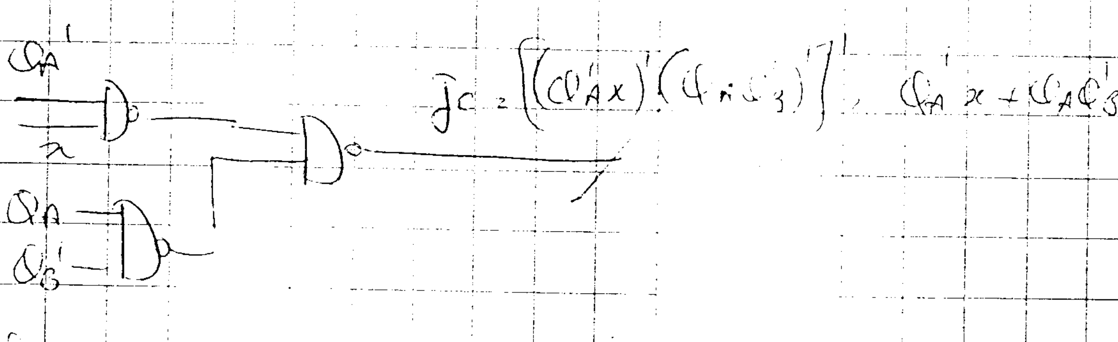
(a) using a 16-to-1 MUX with control inputs A,B,C, and D
(b) using an 8-to-1 MUX with control inputs A, B, and C and added gates

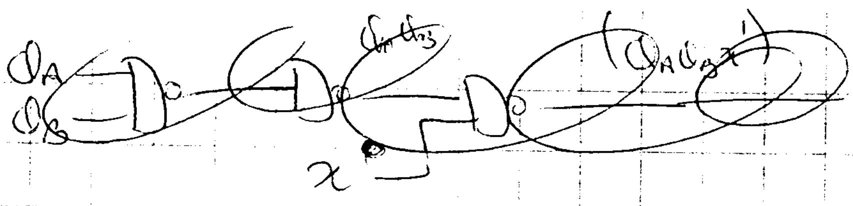
$$F(A,B,C,D,E) = \Sigma\, m(0,2,6,7,8,10,11,12,13,14,16,18,19,29,30) + \Sigma\, d(4,9,21)$$

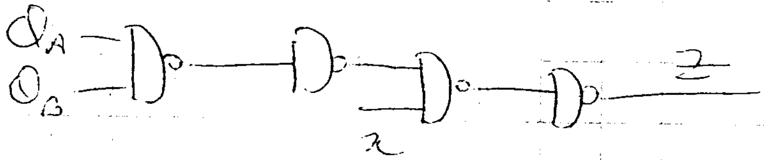

JA
QA
KA
Q'A
JB
QB
KB
Q'B
JC
QC
KC
QC
CLK


QB
QC
$Q_B Q_C$
x
x'
KB
$Q_B Q_C x'$
JA


x
QC
x
$x \oplus Q_C$
QC
Q'B
KA
$(Q'_B[x\oplus Q_C]')'$
$= Q_B + x \oplus Q_C$
$J_B = Q_C x$


Q'A
x
QA
Q'B
$J_C = [(Q'_A x)'(Q_A Q'_B)']' = Q'_A x + Q_A Q'_B$





Don't Care States

Q_A	Q_B	Q_C	x	J_A	K_A	J_B	K_B	J_C	K_C	Q_A^+	Q_B^+	Q_C^+
1	1	1	0	1	1	0	1	0	1	0	0	0
1	1	1	1	0	1	1	0	0	0	0	1	1

$\Rightarrow$ The circuit counts out of don't care states. ~~for 1110 →~~

if $x=0$ $111 \rightarrow 000$

$x=1$ $111 \rightarrow 011$

NOTRE DAME UNIVERSITY
Zouk Mosbeh - Lebanon

Name :
Course : A

(3)

b	x						
c	x	α					
d	x	α	~~hf bh~~				
e	x	df	x	x			
f	x	α	~~bg hj~~	hg	α		
g	hg	α	x	x	α	x	
h	✓	α	α	α	α	α	ga
	a	b	c	d	e	f	g

a = h
h = g
g = a

a = h = g

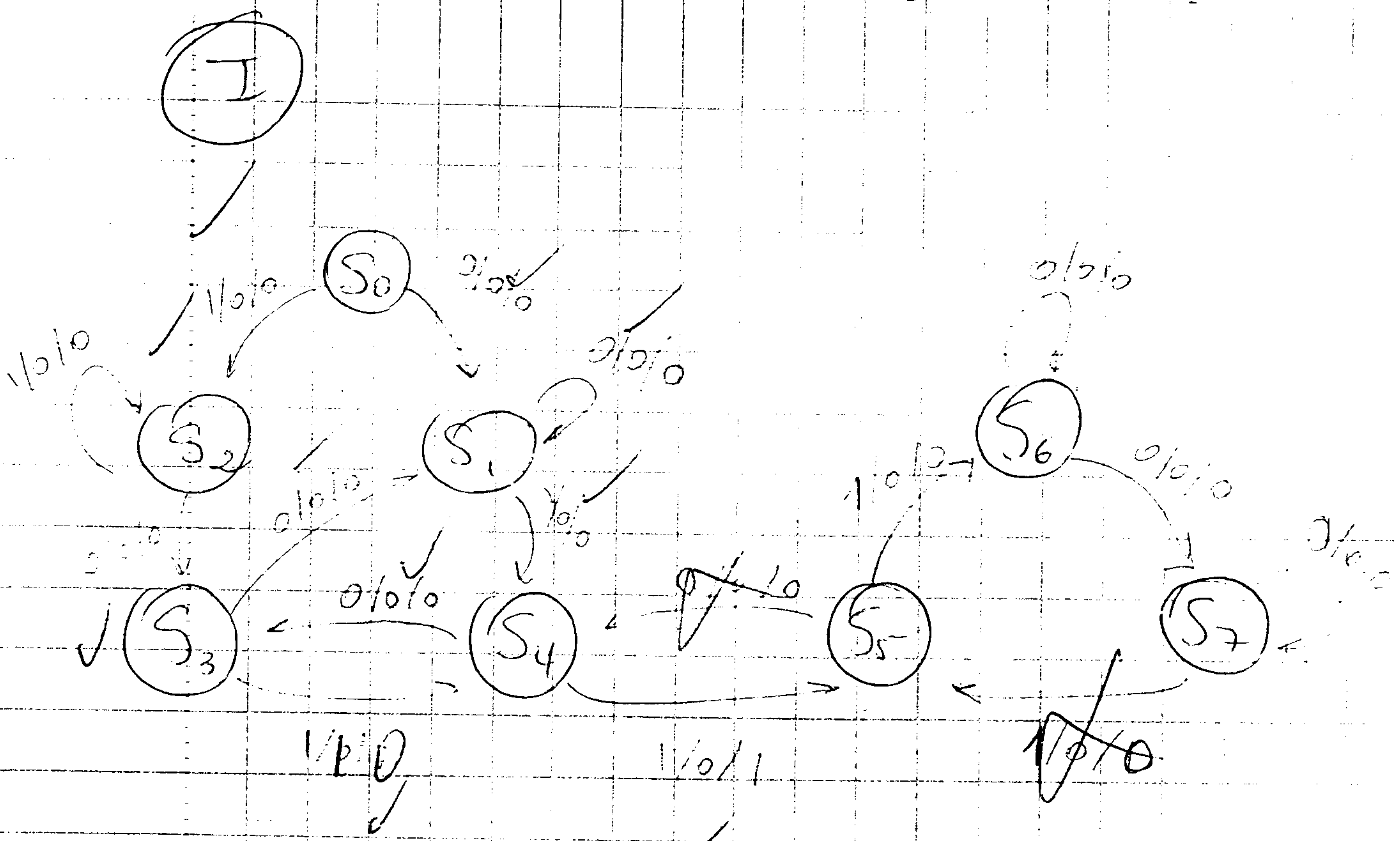
F = D
B = E

~~Reduce~~

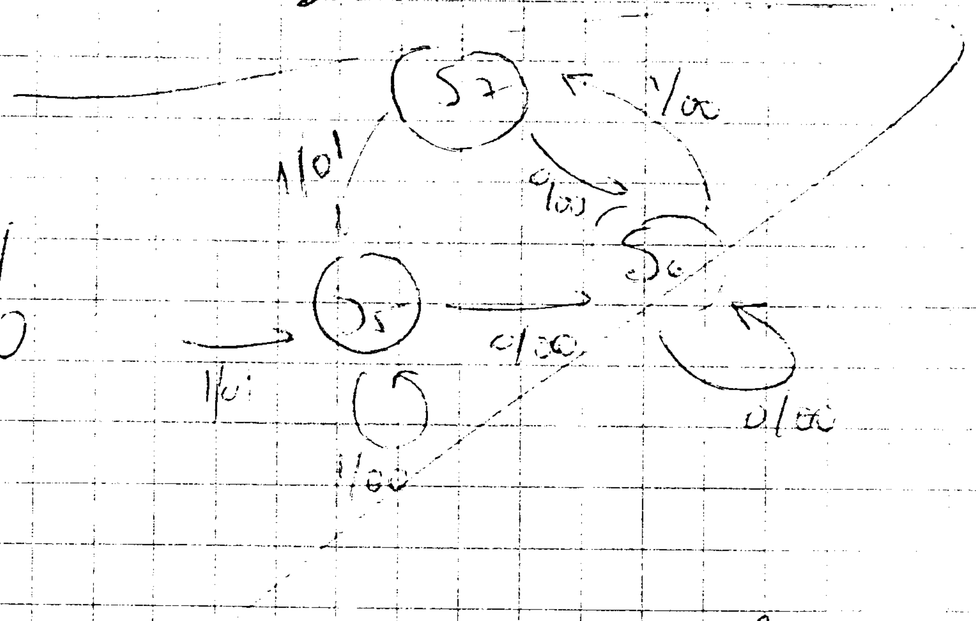
10

a	a/1	c/0
b	c/0	d/1
c	a/0	b/0
d	f/0	a/0
e	c/0	f/1
f	f/0	a/0

4 states



16



State table

NOTRE DAME UNIVERSITY
Zouk Mosbeh - Lebanon

81/100

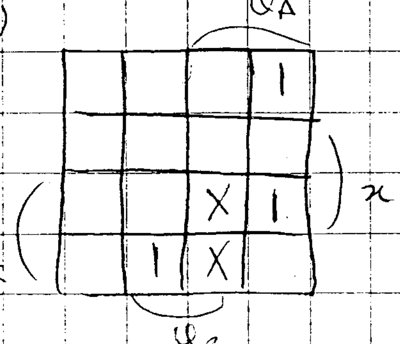
1) 16
2) 50
3) 15

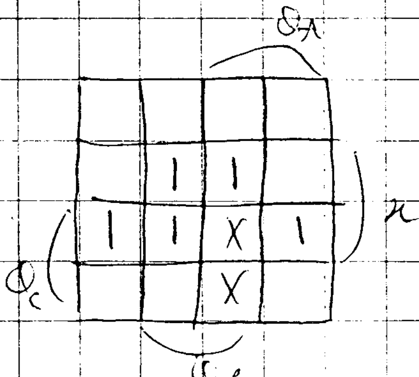
2) State Table.

	x = 0	x = 1
A	A/0	B/0
B	A/0	C/0
C	A/0	D/0
D	E/0	D/0
E	F/0	B/0
F	A/0	G/0
G	A/0	C/1

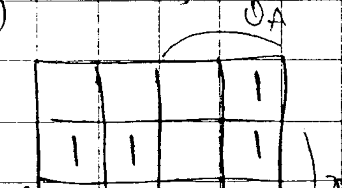
$Q_A Q_B Q_C$	x = 0	x = 1
000	000/0	001/0
001	000/0	010/0
010	000/0	011/0
011	100/0	011/0
100	101/0	001/0
101	000/0	110/0
110	000/0	010/1

$Q_A(t+1)$





$Q_C(t+1)$



J_A (Q_A, Q_B, Q_C, x)

		X	X
		X	X
		X	X
	1	X	X

K_A (Q_A, Q_B, Q_C, x)

X	X	1	
X	X	1	1
X	X	X	
X	X	X	1

$J_A = Q_B Q_C x'$ $\qquad$ $K_A = Q_B + x \oplus Q_C$

J_B (Q_A, Q_B, Q_C, x)

	X	X	
	X	X	
1	X	X	1
	X	X	

K_B (Q_A, Q_B, Q_C, x)

X	1	1	X
X			X
X		X	X
X	1	X	X

$J_B = Q_C x$ $\qquad$ $K_B = x'$

J_C (Q_A, Q_B, Q_C, x)

			1
1	1		1
X	X	X	X
X	X	X	X

K_C (Q_A, Q_B, Q_C, x)

X	X	X	X
X	X	X	X
1		X	1
1	1	X	1

$J_C = Q_A' x + Q_A Q_B'$ $\qquad$ $K_C = Q_B' + x' = (Q_B x)'$

Z (Q_A)

		1	

$Z = Q_A Q_B x$

3. 25 pts. Reduce the following state table to a minimum number of states.

Present State	Next State X = 0	Next State X = 1	Present Output X = 0	Present Output X = 1
a	h	c	1	0
b	c	d	0	1
c	~~h~~ a	b	0	0
d	f	~~h~~ a	0	0
e	c	~~f~~ d	0	1
~~f~~	~~f~~	~~g~~ a	~~0~~	~~0~~
~~g~~	g h a	c	1	0
h	a	c	1	0

a g h

d f

b	α						
c	α	α					
d	α	α	~~th bb~~				
e	α	df	α	~~[illegible]~~			
f	α	~~be gf~~	~~hf bg~~	hg	α		
g	hg	α	α	α	α	α	
h	–	α	α	α	α	α	ga
	a	b	c	d	e	f	g

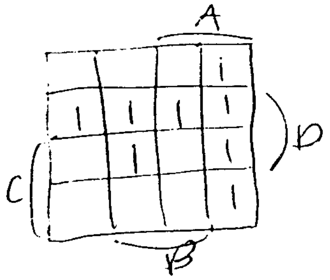
h = a = g

(d, f, b, e)

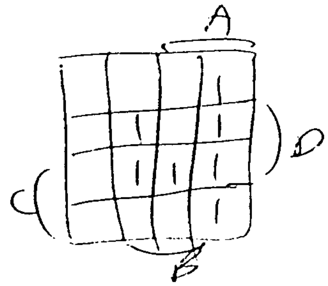
3) $x = A'BD + C'D + AB' + AB'C'D'$

$y = A'BD + BCD + AB'$

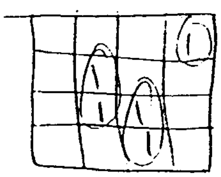
$z = A'BD + BCD + ABC + AB'C'D'$



$x = AB' + C'D + A'BD$



$y = AB' + BCD + ($



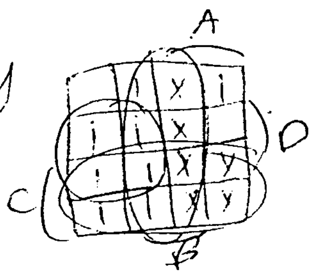
$z = A'BD + ABC + AB'C'D'$

$x = \Sigma m(1, 5, 7, 8, 9, 10$

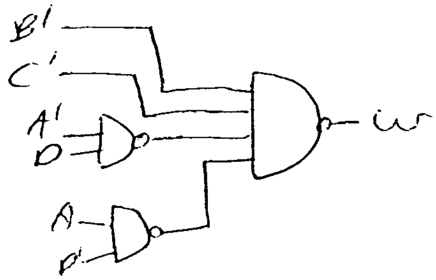
$y = \Sigma m(5, 7, 8, 9, 10, 11, 13,$

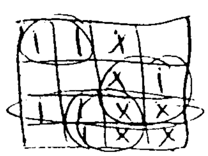
$z = \Sigma m(5, 7, 8, 14, 15)$

5)

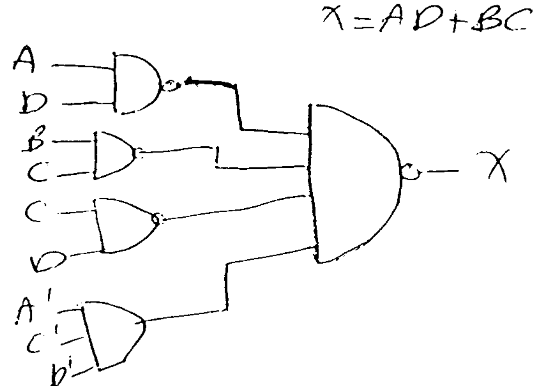


$w = C + B + A'D + AD' = (B'C'(A'D)'(AD')')'$

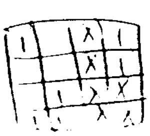




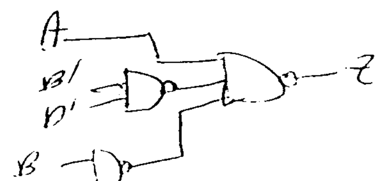
$x = AD + BC + CD + A'C'D'$

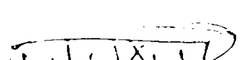


4)



$z = A + B'D' + BC$





Solution SPRING 2001

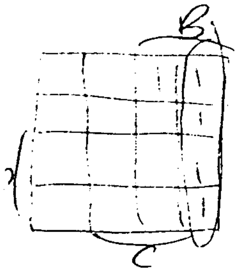
Logic Design Test 2

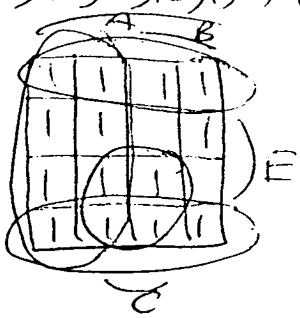
1) $x = A'BCD'E + A'BCDE'$

$y = ABCD'E + ABC'DE$

$z = AB' + A'B(\Sigma m(0,1,2,3,4,7)) + AB(\Sigma m(0,1,2,4,6,7))$

$z = AB' + \Sigma m(8,9,10,11,12,15) + \Sigma m(24,25,26,28,30,31)$





$z = AB' + AE' + A'BC' + ACD + \underbrace{AC'D'}_{BC'D'} + BD'E'$

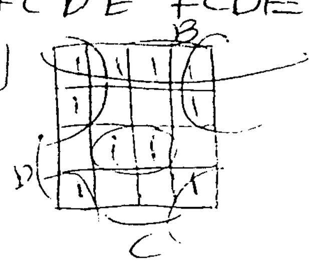
$z = ((AB')'(AE')'(A'BC')'(ACD)'(AC'D')'(BD'E'))'$

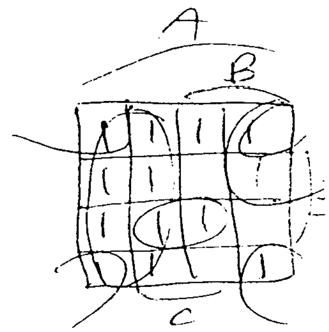
$x = ((A'BCD'E)'(A'BCDE')')'$

$y = ((ABCD'E)'(ABC'DE)')'$

$z = AB' + C'D'E' + C'D'E + C'DE' + CD'E' + CDE$

$\boxed{z = AB' + C'D' + C'E' + D'E' + CDE}$





2) $F(A,B,C,D,E) = \Sigma m(0,2,6,7,8,10,11,12,13,14,16,18,19,29,30) + \Sigma d(4,9,$