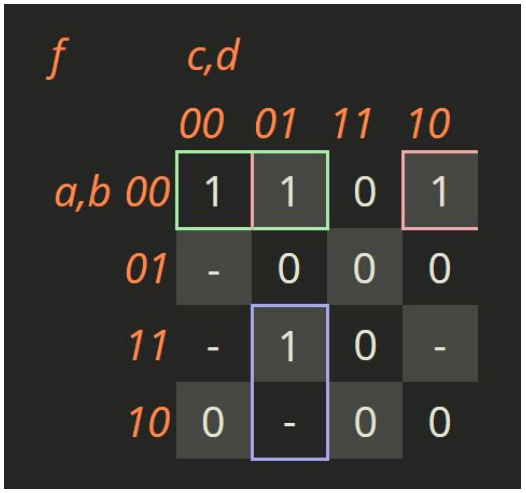
Tutorial T21
Toh Zhen Yu, Nicholas
A0201406Y

1.
lw $t6 -24($sp)
Binary: 100011 11101 01110 1111111111101000
Hex: 0x8FAEFFE8

Field	remarks	value
RegDst	I type instruction	0
MemRead	true	1
Branch	false	0
MemtoReg	Read data	1
MemWrite	false	0
ALUSrc	Use immediate	1
RegWrite	true	1
Instr[31-26]	lw	0x23
Instr[25-21]	$t6	29
Instr[20-16]	$sp	14
Instr[15-11]	-24 in hex is FFE8	0b11111
1	Value in $sp + (-24)	0x7FFFEFDC
2	Value in $t6	0x00006200
3		0b101000
4	-24, sign extended	0xFFFFFFE8
5	PC+4-24*4	108

2.
a.
A.B'.E.L'.O.P'.T'.W.H' + A.L + A.L'.T'
= A.L'.T'.B'.E.O.P'.W.H' + A.L + A.L'.T' (commutative law)
= A.L'.T'.B'.E.O.P'.W.H' + A.L'.T' + A.L (commutative law)
= A.L'.T' + A.L (absorption 1)
= A.(L'.T' + L) (distributive law)
= A.(T' + L) (absorption 2)
= A.T' + A.L (distributive law)

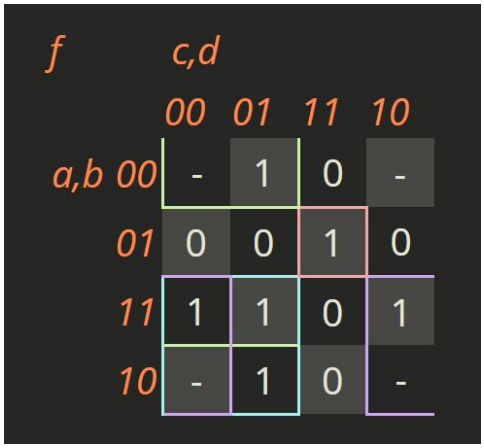
b.
Number of PIs: 6
Number of EPIs: 1



[1]

Simplest SOP: a'.b'.d' + a'.b'.c' + a.b.c'
Simplest POS: (c' + d').(a + b').(a' + b)

c.
Minterms are 1,7,9,12,13,14



Number of PIs: 4
Number of EPIs: 4
Simplest SOP: b'.c' + a'.b.c.d + a.c' + a.d'
Simplest POS: (b + c').(a + b' + c).(a + d).(a' + c' + d')

3.
74LS83 4-Bit Binary Adder with Fast Carry.
It takes in two 4-bit numbers A and B, and carry-in C_0, and outputs a 5-bit number VWXYZ, which can be treated as a 4-bit number WXYZ and a carry-out V. A and B are represented by $A_4A_3A_2A_1$ and $B_4B_3B_2B_1$ respectively.

4.
a.
Relevant prime numbers: 2,3,5,7,11,13,17,19,23
Don't care:
0,1,2,3,4,5,6,7
Minterms: 10,11,13,15,19,21,25,27,31

[1] Generated from https://www.charlie-coleman.com/experiments/kmap/

Map

	$\overline{D}.\overline{E}$	$\overline{D}.E$	$D.E$	$D.\overline{E}$
$\overline{A}.\overline{B}.\overline{C}$	x	x	x	x
$\overline{A}.\overline{B}.C$	x	x	x	x
$\overline{A}.B.C$	0	1	1	0
$\overline{A}.B.\overline{C}$	0	0	1	1
$A.B.C$	0	0	1	0
$A.\overline{B}.C$	0	1	0	0
$A.B.C$	0	0	1	0
$A.B.\overline{C}$	0	1	1	0

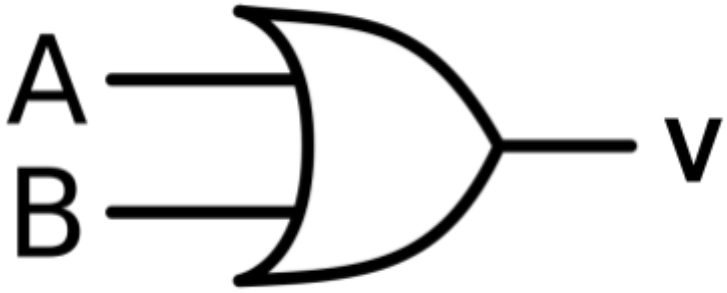
[2]

b.

B.D.E + A'.C'.D + C'.D.E + A'.C.E + A.B.C'.E + B'.C.D'.E

c.

To be valid, the decimal equivalent of the raw input must be greater than or equal to 8. It is sufficient to check that A=1 or B=1. Hence we use a single OR gate.



5.

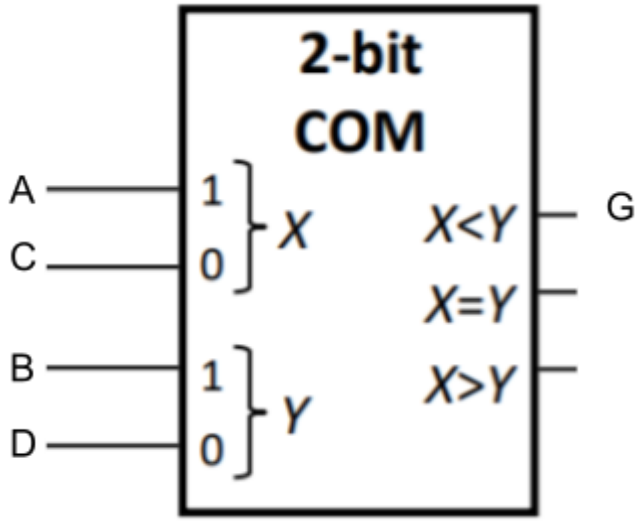
a.

K-map of G:

[2] Generated from http://www.32x8.com/sop5_____A-B-C-D-E_____m_10-11-13-15-19-21-25-27-31_____d_0-1-2-3-4-5-6-7_____option-0_____88878887697885789766 9

f	c,d			
	00	01	11	10
a,b 00	0	1	0	0
01	1	1	1	1
11	0	1	0	0
10	0	0	0	0

Let X1=A, X0=C, Y1=B, Y0=D, then G is the X<Y output



b.