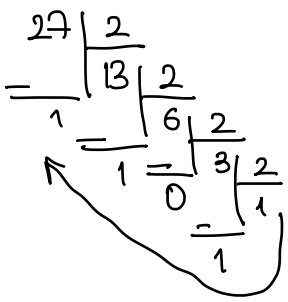
CS303 - HW2 - Deniz Cangi

Q1)

a)

Decimal	Binary	Hexadecimal
546	$(1000100010)_2$	$(222)_{16}$
128	$(1000\ 0000)_2$	$(80)_{16}$
27.375	$(11011.011)_2$	$(1B.6)_{16}$

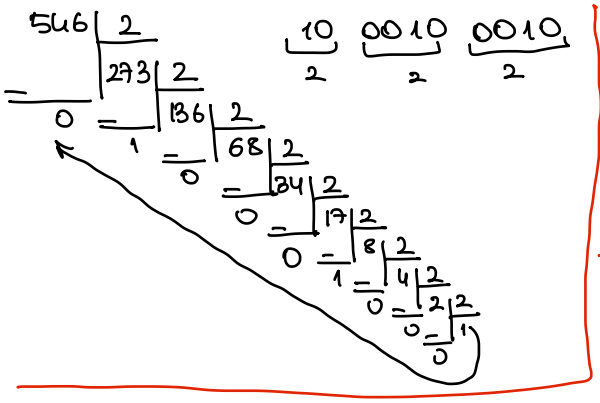
27.375:



$0.375 \times 2 = 0.75 \rightarrow 0$
$0.75 \times 2 = 1.5 \rightarrow 1$
$0.5 \times 2 = 1.0 \rightarrow 1$

in binary = 11011.011

1 1011 . 0110
1 B . 6



10 0010 0010
2 2 2

$128 = 2^7 = (1000\ 0000)_2$

1000 0000
1 1
8 0

Decimal	Signed 8-bit binary in 2's complement form
-6.375	$(1111\ 1001.101)_2$

6.375 in signed 8-bit binary in 2's complement form: 0000 0110.011

↓ 1's complement

1111 1001.100
\+ 1
1111 1001.101

$= -128 + 64 + 32 + 16 + 8 + 1 + 0.5 + 0.125$
$= -7 + 0.625 = -6.375$

Signed 8-bit binary in 2's complement form	Decimal
00000011.1101	$(3.8125)_{10}$
11111111.1111	$(-0.0625)_{10}$
10100110.0111	$(-89.5625)_{10}$

0000 0011 . 1101

$3 + 0.5 + 0.25 + 0.0625 = 0.8125 = 3.8125$

$11111111.1111 = -128+64+32+16+8+4+2+1+0.5+0.25+0.125+0.0625$

$= -1+0.5+0.25+0.125+0.0625$

$= -0.0625$

$10100110.0111 = -128+32+4+2+0.25+0.125+0.0625$

$= -90 + 0.25+0.125+0.0625$

$= -90 + 0.4375 = -89.5625$

b) (10 pt) Calculate the binary equivalent of 127/64. If we use only 4 bits in the fraction, what is the error in the binary representation? How many bits are needed to fully represent 127/64 in signed 2's complement binary number system? Show your work clearly.

$\frac{127}{64} = \frac{2^6+2^5+2^4+2^3+2^2+2^1+2^0}{2^6} = 2^0+2^{-1}+2^{-2}+2^{-3}+2^{-4}+2^{-5}+2^{-6}$

$2^0+2^{-1}+2^{-2}+2^{-3}+2^{-4}+2^{-5}+2^{-6} = (1.111111)_2 = 1+0.5+0.25+0.125+0.0625+0.03125+0.015625$

$=(1.984375)_{10}$

if we only use 4 bits in the fraction = the number becomes = $(1.1111)_2 = 1+0.5+0.25+0.125+0.0625$

$= 1.9375$

an error in binary fraction occurs. The difference between the numbers = $1.98 - 1.9375$

$= 0.046875$

How many bits needed to fully represent 27/64 in signed 2's complement binary number system?

2 bits needed for number part, as it is 01 and for fraction part 6 bits needed to fully represent the fraction part. So in total 2 for number + 6 for fraction = 8 bits needed

Q2) Assume a 5-bit, 2's complement scheme to represent signed integers.

a) (3 pt) Show the range of integers that can be represented.

Range is between $[(10000)_2, (01111)_2]$, in decimal it is equivalent to $[(-16)_{10}, (15)_{10}]$

b) (12 pt) Do the following arithmetic operations and detect overflows. Show your work clearly.

14 + 9 overflow
14 + (-9) no overflow
(-5) + (-9) no overflow
(-7) + (-10) overflow

14 = 01110
9 = 01001
+
*01011 = -16+4+2+1
= -9 X

overflow!

14 = 01110
9 = 01001 → -9 = 10111
111
01110
+ 10111
*00101 = 4+1 = 5 ✓

5 = 00101 9 = 01001
-5 = 11011 -9 = 10111
1 1111
11011
+ 10111
*10010 = -16+2 = -14 ✓

7 = 00111 → -7 = 11001
10 = 01010 → -10 = 10110
1 11001
+ 10110
*01111 = 8+4+2+1 = 15 X

Q3) a) (5 pt) Express the following function as a sum of minterms.

$$F(x,y,z,t) = (x \oplus y)(z+t)$$

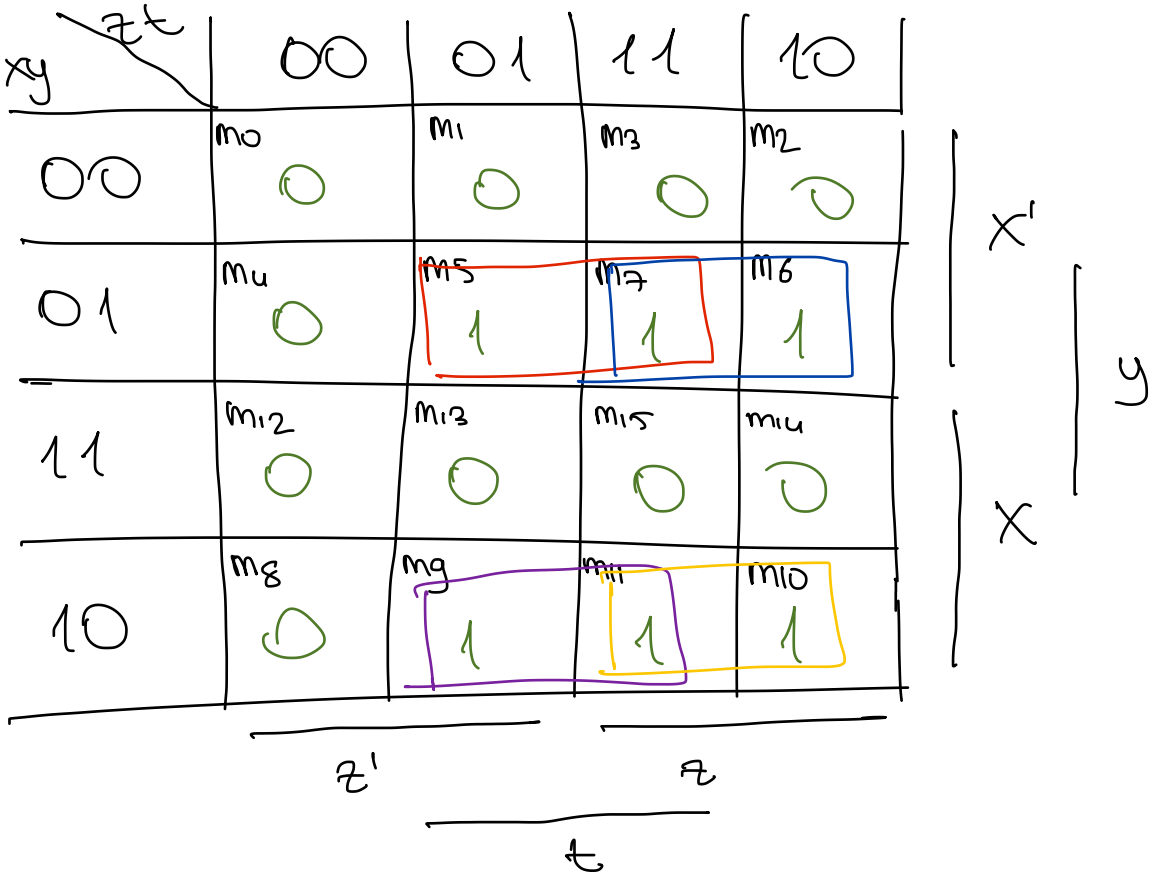
	x	y	z	t	x⊕y	z+t	F(x,y,z,t)
m_0	0	0	0	0	0	0	0
m_1	0	0	0	1	0	1	0
m_2	0	0	1	0	0	1	0
m_3	0	0	1	1	0	1	0
m_4	0	1	0	0	1	0	0
m_5	0	1	0	1	1	1	1
m_6	0	1	1	0	1	1	1
m_7	0	1	1	1	1	1	1
m_8	1	0	0	0	1	0	0
m_9	1	0	0	1	1	1	1
m_{10}	1	0	1	0	1	1	1
m_{11}	1	0	1	1	1	1	1
m_{12}	1	1	0	0	0	0	0
m_{13}	1	1	0	1	0	1	0
m_{14}	1	1	1	0	0	1	0
m_{15}	1	1	1	1	0	1	0

$$F(x,y,z,t) = m_5 + m_6 + m_7 + m_9 + m_{10} + m_{11}$$

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

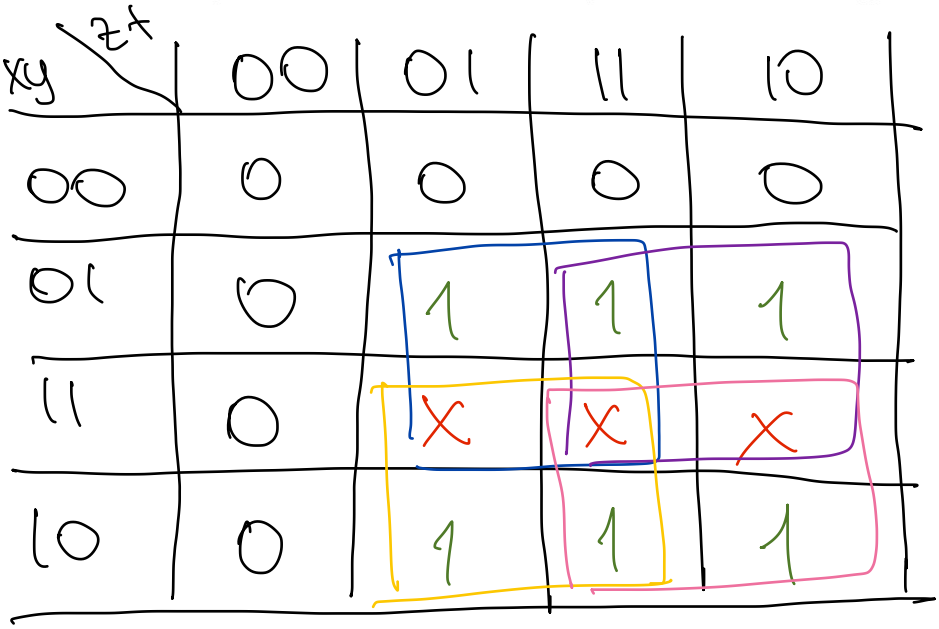
$$= x'yz't + x'yzt' + x'yzt + xy'z't + xy'zt' + xy'zt$$

b) (10 pt) Optimize F using Karnaugh map.



$$F(x,y,z,t) = x'yt + x'yz + xy't + xy'z$$

c) (5 pt) Assume we guarantee that $xyzt_2$ will never be larger than 12. Add necessary don't care conditions and optimize F further using K-map.

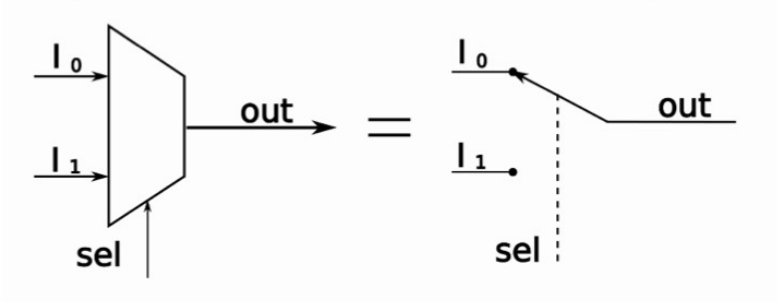


$$F(x,y,z,t) = yt + yz + xt + xz$$

can further optimize with ad-hoc:

$$= y(t+z) + x(t+z) = (t+z)(x+y)$$

Q4) Multiplexer is a logical unit that selects between several input signals.

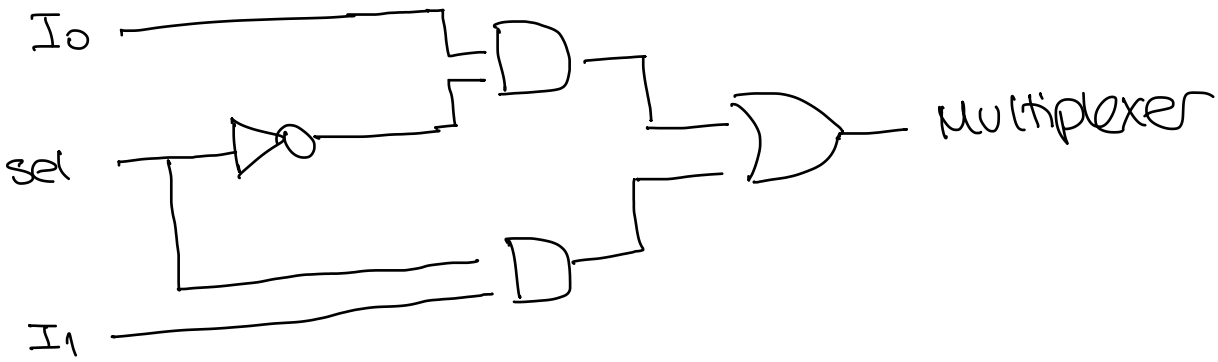


For a 2-to-1 multiplexer, if the sel input is 0, out output is equal to I_0. If the sel input is 1, out output is equal to I_1. Design this 2-to-1 multiplexer and draw its circuit.

Multiplexer = $I_0 \cdot sel' + I_1 \cdot sel$

if sel = 0 → sel' = 1 → Multiplexer = I_0

if sel = 1 → sel' = 0 → Multiplexer = I_1



Q5) You are asked to design a 2-bit comparator circuit. Operation of this circuit is defined as follows:

Inputs: A=(A1,A0)$_2$ and B=(B1,B0)$_2$, 2-bit signed numbers in 2's complement form
Outputs: EQ, G, L

if (A==B)
EQ=1
else
EQ=0
	if (A<B)
	L=1
	else
	L=0
		if (A>B)
		G=1
		else
		G=0

Design circuit for this comparator, however you would like to design it and draw its circuit. You can use deductions as we used in class. For example, if you have a circuit for EQ output, and a circuit for L output, you can use those two outputs to derive G output. Show your work clearly.

Truth Table:

	A_1	A_0	B_1	B_0	EQ	L	G
m_0	0	0	0	0	1	0	0
m_1	0	0	0	1	0	1	0
m_2	0	0	1	0	0	0	1
m_3	0	0	1	1	0	0	1
m_4	0	1	0	0	0	0	1
m_5	0	1	0	1	1	0	0
m_6	0	1	1	0	0	0	1
m_7	0	1	1	1	0	0	1
m_8	1	0	0	0	0	1	0
m_9	1	0	0	1	0	1	0
m_{10}	1	0	1	0	1	0	0
m_{11}	1	0	1	1	0	1	0
m_{12}	1	1	0	0	0	1	0
m_{13}	1	1	0	1	0	1	0
m_{14}	1	1	1	0	0	0	1
m_{15}	1	1	1	1	1	0	0

$EQ = \Sigma(0, 5, 10, 15)$

$L = \Sigma(1, 8, 9, 11, 12, 13)$

$G = \Sigma(2, 3, 4, 6, 7, 14)$

$G = EQ' \cdot L' = (EQ + L)'$

Karnaugh-map for EQ:

A_1A_0 \ B_1B_0	00	01	11	10
00	1	0	0	0
01	0	1	0	0
11	0	0	1	0
10	0	0	0	1

$EQ = A_1'A_0'B_1'B_0' + A_1'A_0B_1'B_0 + A_1A_0B_1B_0 + A_1A_0'B_1B_0'$

$= A_1'B_1'(A_0'B_0' + A_0B_0) + A_1B_1(A_0B_0 + A_0'B_0')$

$= (A_0B_0 + A_0'B_0')(A_1B_1 + A_1'B_1')$

(XNOR, XNOR)

$= \overline{(A_0 \oplus B_0)}\,\overline{(A_1 \oplus B_1)}$

Karnaugh-map for L:

A_1A_0 \ B_1B_0	00	01	11	10
00	0	1	0	0
01	0	0	0	0
11	1	1	0	0
10	1	1	1	0

$L = A_1B_1' + A_1A_0'B_0 + A_0'B_1'B_0$

$= A_1B_1' + A_0'B_0(A_1 + B_1')$

Karnaugh Map for G:

A_1A_0 \ B_1B_0	00	01	11	10
00	0	0	1	1
01	1	0	1	1
11	0	0	0	1
10	0	0	0	0

$G = A_1'B_1 + A_0B_1B_0' + A_1'A_0B_0'$

$= A_1'B_1 + A_0B_0'(A_1' + B_1)$

Circuits