ECE2020 Test 2 Summer 2012 GTL Name: KEY
June 27, 2013
Calculators not allowed. **Show your work for any possible partial credit or in some cases for any credit at all.** Note the last page shows you some blank K maps just in case you need them on this exam. 8 pages, possible 100 points.

Problem 1 Part A (10 points) Convert the following notations:

Binary representation	Hexidecimal representation
1010110010010101	AC95
Signed Binary two's complement represented	Decimal
11100.01	− 3.75
Decimal	Octal
673	1241
Decimal Notation	Binary notation
7.75	111.11

11100.01 => 00011.10 + 1 => 00011.11 => 3 + 1/2 + 1/2² = 3.75

673

$$\begin{array}{ll} 2\overline{)673} = 336 & R\ 1 \\ 2\overline{)336} = 168 & R\ 0 \\ 2\overline{)168} = 84 & R\ 0 \\ 2\overline{)84} = 42 & R\ 0 \\ 2\overline{)42} = 21 & R\ 0 \\ 2\overline{)21} = 10 & R\ 1 \\ 2\overline{)10} = 5 & R\ 0 \\ 2\overline{)5} = 2 & R\ 1 \\ 2\overline{)2} = 1 & R\ 0 \\ 2\overline{)1} = 0 & R\ 1 \end{array}$$

→ 1010100001

7.75 => 111.11

1×2^{-1} 1×2^{-2}

Problem 1 Part B (15 points) For the 24 (and 20) bit representations below, determine the most negative value, most positive value, and step size (difference between sequential values). All answers must be expressed in decimal notation. Fractions (for example 3/16ths) may be used. All signed representations are two's complement signed numbers.

representation	most negative value	most positive value	step size
unsigned integer (24 bits). (0 bits)	0	16M − 1	1
Signed integer (24 bits). (0 bits)	−8M	8M − 1	1
unsigned fixed-point (17 bits). (7 bits)	0	128k − 1/128	1/128
signed fixed-point (15 bits). (5 bits)	−16k	16k − 1/32	1/32

UNSIGNED : $\frac{2^N - 1}{2^k}$

$$\frac{2^{24}-1}{2^0} = 2^{24}-1 = 2^{10}2^{10}2^4 - 1 = 16M - 1$$

SIGNED ·

$$\frac{-2^{N-1}}{2^k} \text{ to } \frac{2^{N-1}-1}{2^k} \Rightarrow \frac{-2^{23}}{2^0} \text{ to } \frac{2^{23}-1}{2^0} \Rightarrow -2^{10}2^{10}2^3 \text{ to } 2^{10}2^{10}2^3-1$$

$-8M$ to $8M - 1$

UNSIGNED fixed point

$$\frac{2^{24}-1}{2^7} \Rightarrow 2^{17} - \frac{1}{2^7} \Rightarrow 2^{10}2^7 - \frac{1}{2^7} \Rightarrow 128k - \frac{1}{128}$$

SIGNED

$$\frac{-2^{19}}{2^5} \text{ to } \frac{2^{19}-1}{2^5} \Rightarrow -2^{14} \text{ to } 2^{14} - \frac{1}{2^5} \Rightarrow -2^{10}2^4 \text{ to } 2^{10}2^4 - \frac{1}{2^5}$$

$$\Rightarrow -16K \text{ to } 16k - \frac{1}{32}$$

Problem 2 Part A Arithmetic (5 points) For each problem below, compute the operations using the rules of arithmetic, and indicate whether an overflow occurs assuming all numbers are expressed using a six bit **unsigned** representation

	110110 +100110 1)011100	011001 +010110 101111
result	011100	101111
unsigned error?	Y	N

Problem 2 Part B Arithmetic (10 points) For each problem below, compute the operations using the rules of arithmetic, and indicate whether an overflow occurs assuming all numbers are expressed using a six bit **signed** two's complement representations.

	011111 +000100 100011	111111 +000010 1)000001	011000 -111011	001111 -010101
result	100011	000001	011101	111010
signed error?	Y	N	N	N

111011

000100
+1
000101

011000
000101
011101

010101

101010
+1
101011

001111
101011
111010

3) Given the circuit below:

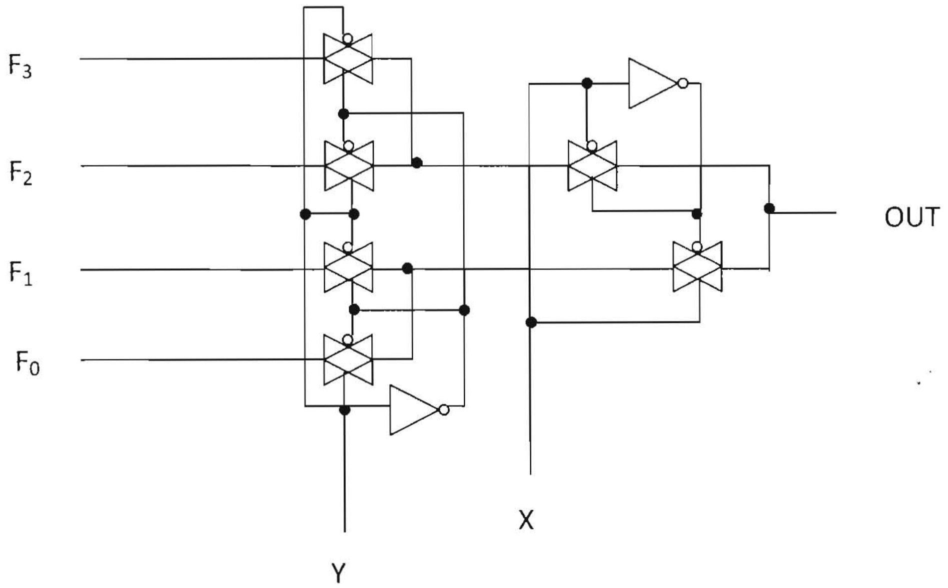
a) (5 points) The designer wants to implement an exclusive-or function where out = Y exclusive-or X. Fill in the values for $F_3\ F_2\ F_1\ F_0$ to implement the desired exclusive-or function.

b) (5 points) The designer wants to implement the NAND function where out = Y NAND X. Fill in the values for $F_3\ F_2\ F_1\ F_0$ to implement the desired NAND function.

C) (5 points) The designer wants to implement the NOR function where out = Y NOR X. Fill in the values for $F_3\ F_2\ F_1\ F_0$ to implement the desired NOR function.

function desired:	F_3	F_2	F_1	F_0
a) exclusive-or	0	1	1	0
b) NAND	1	1	1	0
c) NOR	1	0	0	0

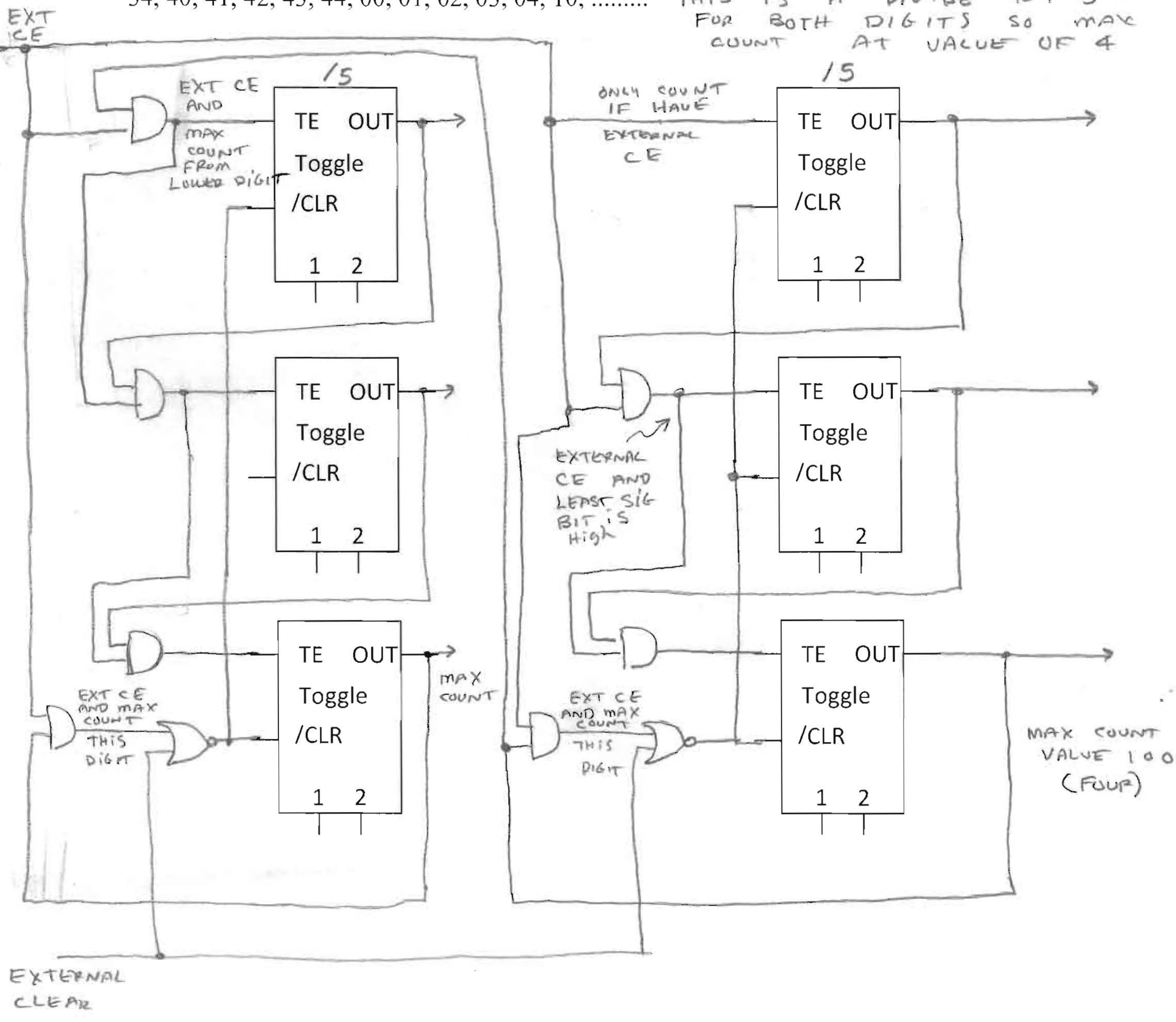
Y	X		EX-OR	NAND	NOR
0	0	F3	0	1	1
0	1	F1	1	1	0
1	0	F2	1	1	0
1	1	F0	0	0	0



4) Counters (15 points) Connect the needed toggle cells below to build a multiple digit counter (like in an alarm clock type application) that counts in the following strange 2 digit sequence. Include any circuitry needed to allow this to work. The toggle cells in the right column are for the least significant digit (right value), the toggle cells in the left column are for the most significant digit (left value). Include an active high count enable (CE) and an active high reset (RESET) inputs.

COUNT SEQUENCE
000
001
010
011
100

desired count sequence: 00, 01, 02, 03, 04, 10, 11, 12, 13, 14, 20, 21, 22, 23, 24, 30, 31, 32, 33, 34, 40, 41, 42, 43, 44, 00, 01, 02, 03, 04, 10, ... THIS IS A DIVIDE BY 5 FOR BOTH DIGITS SO MAX COUNT AT VALUE OF 4



NOTE: ALL CLK 1'S CONNECTED
ALL CLK 2'S CONNECTED

5) Priority Encoders (15 points) Given the truth table for the following priority encoder:

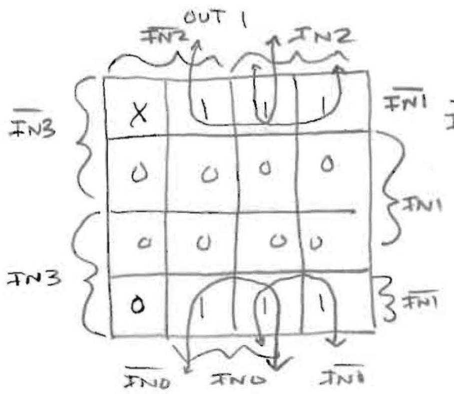
IN3	IN2	IN1	IN0	OUT1	OUT0	VALID
0	0	0	0	X	X	0
1	0	0	0	0	1	1
X	1	0	X	1	0	1
X	X	1	X	0	0	1
X	0	0	1	1	1	1

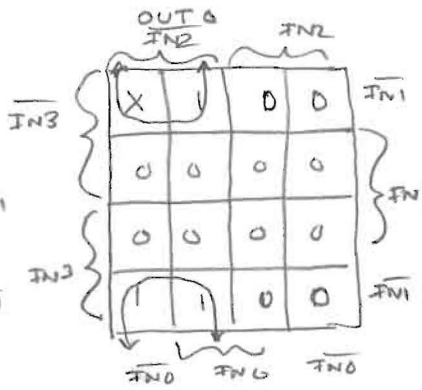
a) List the priority order of the inputs IN3, IN2, IN1, IN0:

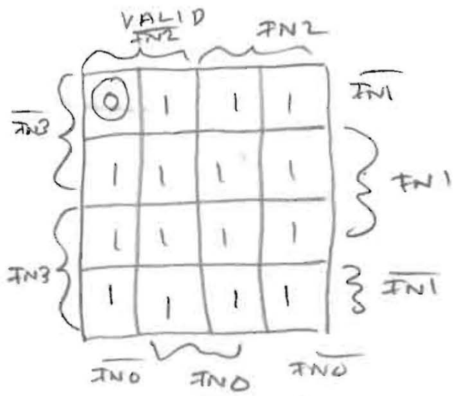
IN1 > IN2 > IN0 > IN3

highest ... lowest

b) Using basic gates (AND, OR, NAND, NOR, NOT) show the gate level implementation for this priority encoder.







$OUT1 = \overline{IN1}\cdot IN0 + \overline{IN1}\, IN2$

$= \overline{IN1}\,(IN0 + IN2)$

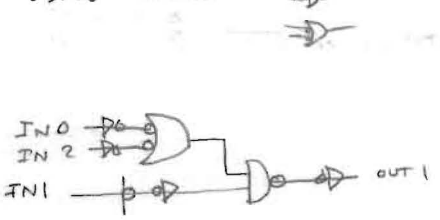
$OUT0 = \overline{IN2}\cdot\overline{IN1}$

$VALID = IN3 + IN2 + IN1 + IN0$

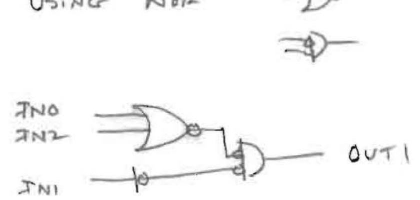
SAME AS

$VALID = \overline{\overline{IN3}\cdot\overline{IN2}\cdot\overline{IN1}\cdot\overline{IN0}}$

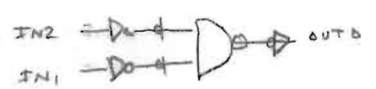
USING NAND



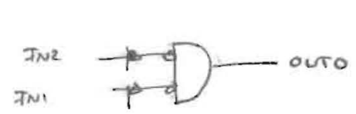
USING NOR



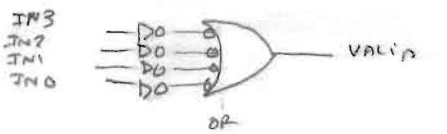
USING NAND

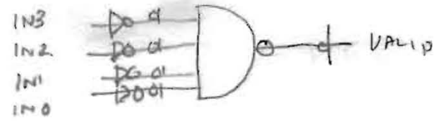


USING NOR

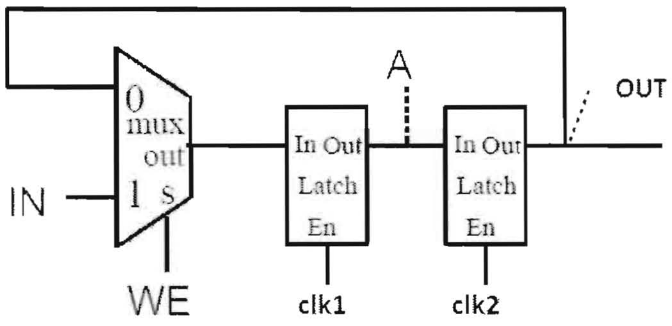


USING NAND

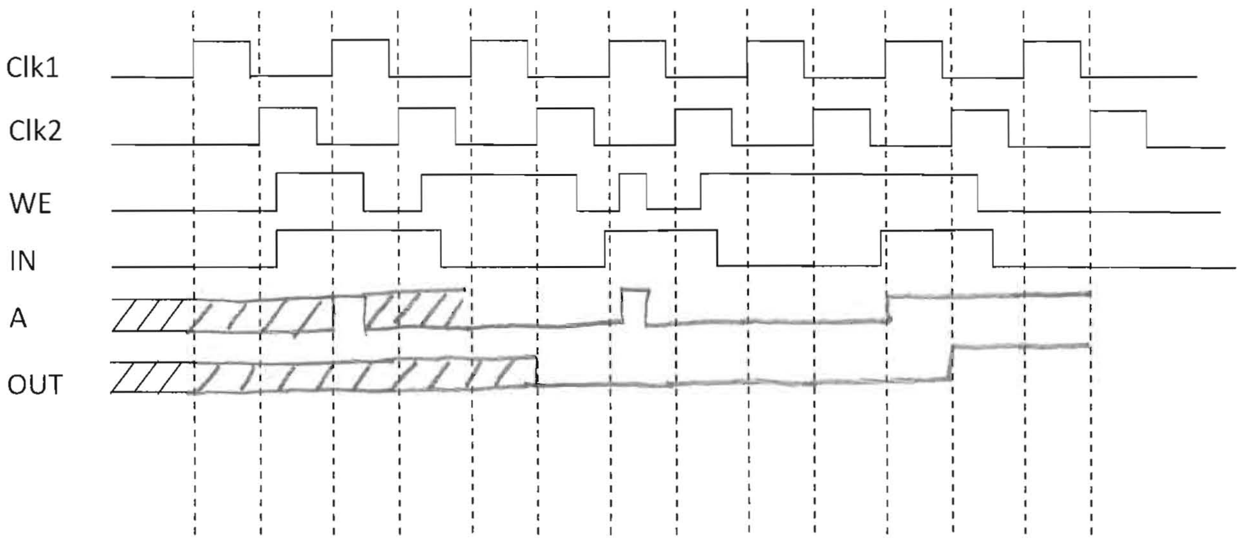




6) Registers (15 points) Consider the register implementation below.



Assume the following signals are applied to your register. Draw the signal at point A (output of the first latch), the signal at point OUT (output of second latch). Assume A and OUT start at unknown values.



K MAPS: