The University of Alabama in Huntsville
ECE Department
EE 202 – 02
Fall 2010
Sample Final Exam Solution

J	K	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	Q'(t)

D	Q(t+1)
0	0
1	1

T	Q(t+1)
0	Q(t)
1	Q'(t)

1. (1 point) A major trend in digital design methodology is the use of a _hardware description language_ to describe and simulate the functionality of a digital circuit.
2. (1 point) _Propagation delay_ is the average transition delay time for a signal to propagate from input to output.
3. (1 point) A _parity bit_ is an extra bit included with a binary message to make the number of 1's either odd or even.
4. (1 point) A multiplexer can be constructed with three-state gates: high, low and _high-impedance_.
5. (1 point) A register capable of transferring the binary information held in each cell to its neighboring cell, in a selected direction, is called a _shift register_
6. (5 points Convert (32445_6) to decimal:

$$3 * 6^4 + 2 * 6^3 + 4 * 6^2 + 4 * 6^1 + 5 * 6^0 = 3 * 1296 + 2 * 216 + 4 * 36 + 4 * 6 + 5 * 1$$
$$= 3888 + 432 + 144 + 24 + 5$$
$$= 4493$$

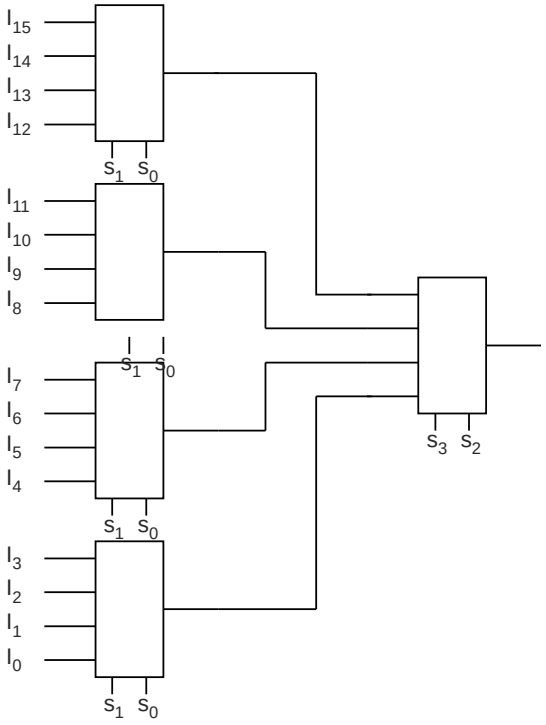
7. (10 points) Convert decimal +12 and +17 to binary, using the signed-2's-complement representation and enough digits to accommodate the numbers. Then perform the binary equivalent of (+12) + (-17). Convert the answer back to decimal and verify that it is correct.

2	12		2	17	
2	6	0	2	8	1
2	3	0	2	4	0
2	1	1	2	2	0
2	0	1	2	1	0
			2	0	1

+12 = 00 1100 +17 = 01 0001 -17 = -10 1111

+12 = 00 1100 12 - 17 = -5
+(-17)= 10 1111
11 1011 = $1 \times -2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
$= -32 + 16 + 8 + 0 + 2 + 1$
$= -5$ ✓

8. (10 points) Construct a 16 × 1 multiplexer with as many 4 × 1 multiplexers and any additional logic that you might need. Use block diagrams for the components.



9. (15 points) Design a circuit that implements the following code converter. You do not have to draw a circuit diagram.

x	y	z		A	B	C
0	0	0		1	1	1
1	0	0		1	1	0
0	1	0		1	0	1
1	1	0		1	0	0
0	0	1		0	1	1
1	0	1		0	1	0
0	1	1		0	0	1
1	1	1		0	0	0

A

x \ y	0	1
0	1	0
1	0	1

z

B

x \ y	0	1
0	1	1
1	0	0

z

C

x \ y	0	1
0	0	1
1	0	1

z

10. (15 points) An XY flip-flop has four operations, clear to 0, no change, set to 1, and complement, when inputs are X and Y are 00, 01, 10, and 11, respectively.

(a) Tabulate the characteristic table.

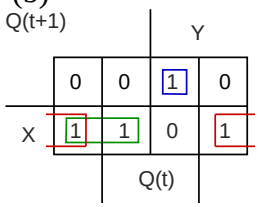
(b) Derive the characteristic equation.

(c) Tabulate the excitation table.

(a)

X	Y	Q(t)	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

(b)



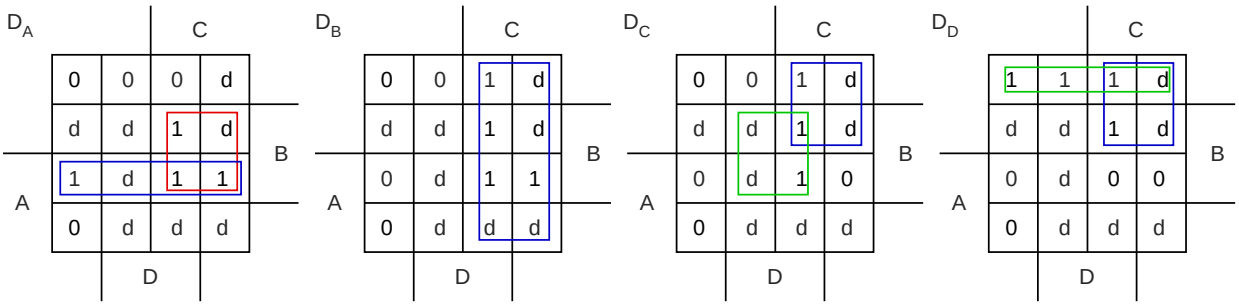
$Q(t + 1) = X'YQ + XY' + XQ'$

(c)

Q(t)	Q(t+1)	X	Y
0	0	0	d
0	1	1	d
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

11. (20 points) Design a 4-bit counter which counts in the sequence 0000, 0001, 0011, 0111, 1111, 1110, 1100, 1000, 0000 using clocked D flip-flops. You do not have to draw the circuit diagram. Is the counter self-correcting if it comes up in an unused state?

Current State	Next State	D_A	D_B	D_C	D_D
0000	0001	0	0	0	1
0001	0011	0	0	1	1
0011	0111	0	1	1	1
0111	1111	1	1	1	1
1111	1110	1	1	1	0
1110	1100	1	1	0	0
1100	1000	1	0	0	0
1000	0000	0	0	0	0



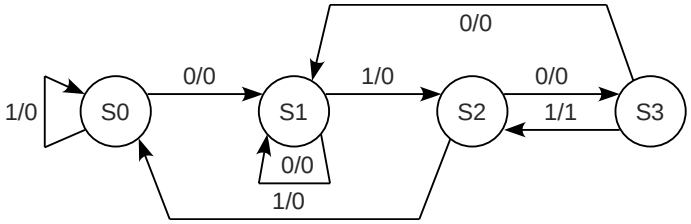
$D_A = AB + BC$ $D_B = C$ $D_C = BD + A'C$ $D_D = A'B' + A'C$

Unused States	D_A	D_B	D_C	D_D	Next States
0010	0	1	1	1	0111, correcting
0100	0	0	0	0	0000, correcting
0101	0	0	1	0	0010, 0111, correcting
0110	1	1	1	1	1111, correcting
1001	0	0	0	0	0000, correcting
1010	0	1	0	0	0100, 0000, correcting
1011	0	1	0	0	0100, 0000, correcting
1101	1	0	1	0	1010, 0100, 0000, correcting

12. (20 points) Design a Mealy sequential circuit that has one input and one output. This circuit has an output of 1 whenever its input string has the string 0101 in sequence and otherwise has an output of 0. Two sequences can overlap. Use JK flip-flops. You do not have to draw the circuit diagram.

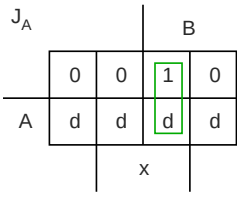
Input: 01101010111101101111010100101
Output: 00000010100000000000000100001

S0: no valid part of the sequence yet
S1: received 0
S2: received 01
S3: received 010

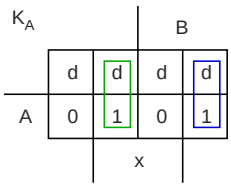


Current State	Input	Next State	Output
S0	0	S1	0
S0	1	S0	0
S1	0	S1	0
S1	1	S2	0
S2	0	S3	0
S2	1	S0	0
S3	0	S1	0
S3	1	S2	1

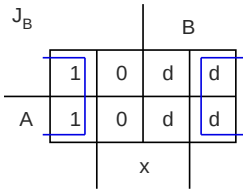
Current State	Input	Next State	Output	J_A	K_A	J_B	K_B
00	0	01	0	0	d	1	d
00	1	00	0	0	d	0	d
01	0	01	0	0	d	d	0
01	1	10	0	1	d	d	1
10	0	11	0	d	0	1	d
10	1	00	0	d	1	0	d
11	0	01	0	d	1	d	0
11	1	10	1	d	0	d	1



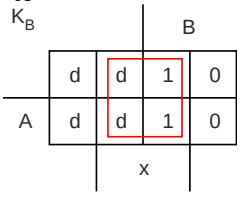
$J_A = Bx$



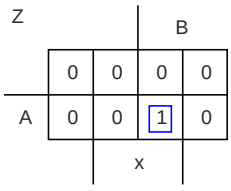
$K_A = B'x + Bx'$



$J_B = x'$



$K_B = x$



$Z = ABx$