National Exams May 2015

07-Elec-A4, Digital Systems & Computers

3 hours duration

NOTES:

1. If doubt exists as to the interpretation of any question, the candidate is urged to submit with the answer paper, a clear statement of any assumptions made.

2. This is a Closed Book exam.
   Candidates may use one of two calculators, the Casio or Sharp approved models.

3. FIVE (5) questions constitute a complete exam.
   Clearly indicate your choice of any five of the six questions given otherwise the first five answers found will be considered your pick.

4. All questions are worth 12 points.
   See below for a detailed breakdown of the marking.

Marking Scheme

1. (a) 2, (b) 2, (c) 2, (d) 3, (e) 3, total = 12
2. (a) 6, (b) 6, total = 12
3. (a) 4, (b) 4, (c) 4, total = 12
4. (a) 3, (b) 4, (c) 3, (d) 2, total = 12
5. (a) 6, (b) 6, total = 12
6. (a) 4, (b) 4, (c) 4, total = 12

The number beside each part above indicates the points that part is worth

1.- (a) Using only one NAND gate build a NOT gate. [2 pts]

(b) Using only NAND gates build a two-input OR gate. [2 pts]

Draw circuits for the function g given by

$$g = \overline{\left((A+B)\cdot\bar{C} + B\bar{C}D\right)\cdot E\cdot(A+B)}$$

(c) as written above using AND, OR & NOT gates (assume 2 & 3-input gates are available). [2 pts]

(d) using NAND gates only (assume literal complements as well as 2 and 3-input gates are available). [3 pts]

(e) using NOR gates only (assume literal complements as well as 2 and 3-input gates are available). [3 pts]

Note: Do not use Boolean algebra or K-map to simplify g in parts (c)-(e) above.

2.- The function of any flip-flop type can be obtained by using another type of flip-flop with suitable logic applied to the latter's inputs.

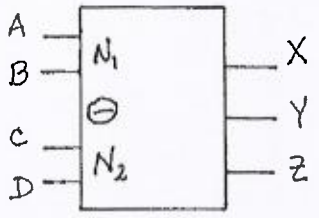
(a) Show how to implement a T flip-flop using a RS flip-flop. [6 pts]

(b) Show how to implement a RS flip-flop using a D flip-flop. [6 pts]

Include K-map work showing how to obtain the additional logic needed and draw the logic circuit for each case including flip-flops and additional logic.

Note: Use flip-flop excitation table attached at the end as needed.

3.- The logic box shown below performs the substraction of two 2-bit numbers N_1 =AB (minuend) & N_2 = CD (subtrahend). The result (difference) is a three bit number D = XYZ.



$N_1 = A\ B$

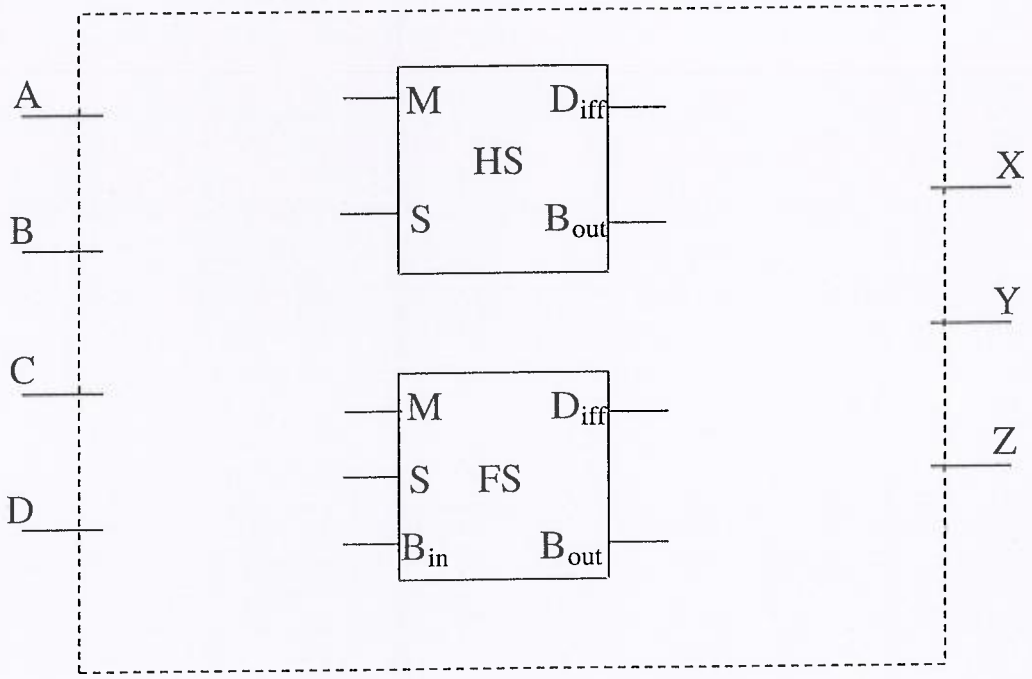
$-N_2 = -C\ D$

$D = X\ Y\ Z$

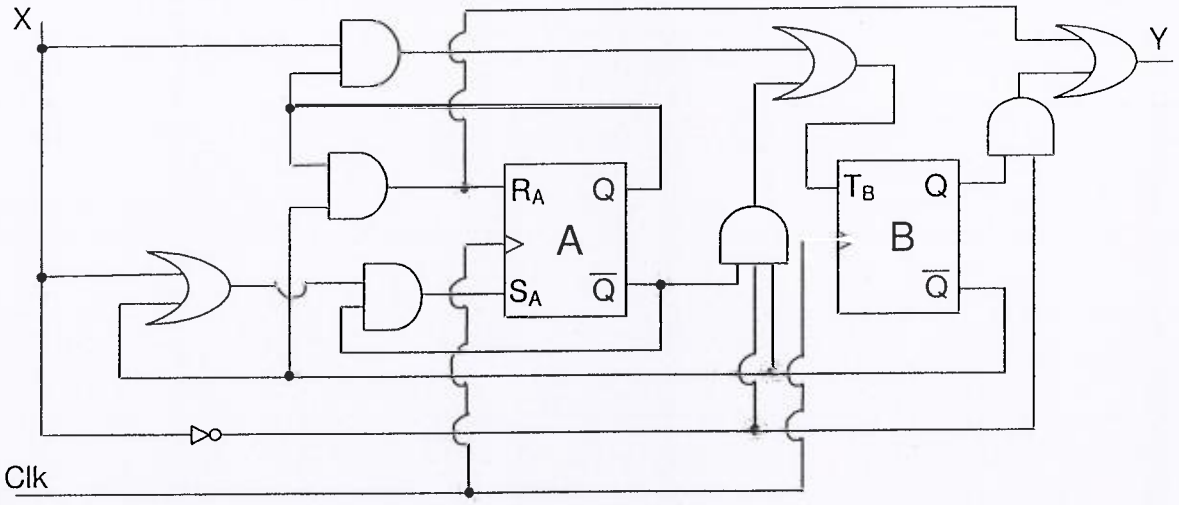
(a) Build the truth table for the functions X, Y & Z. [4 pts]
Note: Values of D result in 3-bit 2's complement.

(b) Use K-maps to minimize the functions X, Y & Z in sum of products form. [4 pts]

(c) You are given the half-subtractor (HS) and full-subtractor (FS) shown below.
Complete the diagram connections required to obtain the difference D = XYZ of $N_1 - N_2 = AB - CD$.
M - minuend, S - subtrahend, D_{iff} - difference, B_{in} - borrow-in, B_{out} - borrow-out [4 pts]



4.- The following circuit with input X and output Y uses one RS flip-flop and a T flip-flop.



(a) Write the logic expressions for R_A, S_A, T_B and Y. [3 pts]

(b) Obtain the state transition table for the circuit.
Include A, B, X, R_A, S_A, T_B, A^+, B^+, Y in this order. [4 pts]

(c) Sketch the state transition diagram for the circuit. [3 pts]

(d) Is this a Moore or a Mealy machine? Explain. [2 pts]

Note: Consult flip-flop excitation table attached at the end as needed.

5.- (a) Describe the algorithm of a short assembly program, including any additional routines needed, in charge of [6 pts]

1. Reading ONE (1) character (char) received through an asynchronous serial port,
2. Converting that char received to lowercase if uppercase, or viceversa (assume the char received is a letter), and
3. Transmitting the char obtained in Step 2 above through the asynchronous serial port using interrupts.

You can use a subroutine called `inchar` starting at address $FFCD_{hex}$. This subroutine loops until a character is received by the input serial port then returns the ASCII character in a CPU register called accumulator A (ACCA).

Assume the asynchronous serial port has been initialized already including a serial line speed of 9600 baud.

Assume standard serial port registers: status (SR), control (CR), transmit data (TDR) and receive data (RDR) are memory mapped and their addresses available.

(b) Both input and output serial channels support RS-232 levels, this is, the microprocessor board receive data (RxD) line is coupled to the channel through RS-232 receivers and the transmit data (TxD) line is coupled to the channel through RS-232 drivers.

Assume the character 'U' is received by your program through the serial port.

Once your program runs sketch the time waveforms for:

(i) the serial bit stream on the TxD line in the board before the RS-232 drivers, and

(ii) the same serial bit stream in (i) above on the transmit line of the RS-232 cable connected to the board serial port connector.

Label each bit appropriately according to their position and role in the frame.

Include time scale values and voltage scale values for each case. [6 pts]

Assume CMOS logic levels are used within the microprocessor board.

ASCII value for 'U' is 55_{hex}

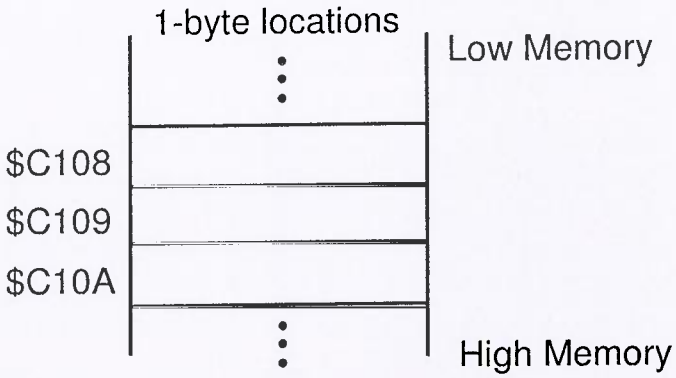
ASCII value for 'u' is 75_{hex}

hex stands for hexadecimal

6.- For microprocessors, such as Motorola's, that use big-endian order for storing multiple-byte variables, describe the results of executing the following two assembly instructions by answering parts (a)-(c) below.

(a) Fill the memory block below with the result of the instruction: [4 pts]

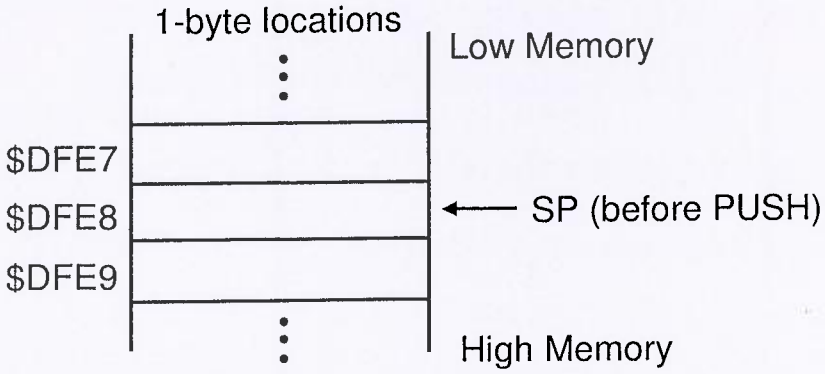
"Store the 16-bit number $10B2 to address $ C109"



(b) The stack pointer register SP = $DFE8 before the following instruction is executed.

Fill the memory block below with the result of the instruction: [4 pts]

"Push the 16-bit number $2EA4 onto the stack"



(c) What is the value of the stack pointer register SP after the PUSH? [4 pts]

Note: $XXXX = $XXXX_{hex}$ is a hexadecimal value

Excitation Table

Q	Q+	R	S	J	K	T	D
0	0	X	0	0	X	0	0
0	1	0	1	1	X	1	1
1	0	1	0	X	1	1	0
1	1	0	X	X	0	0	1

Basic Boolean Identities

	Identity	Comments
1.	$A+0=A$	Operations with 0 and 1
2.	$A+1=1$	Operations with 0 and 1
3.	$A+A=A$	Idompotent
4.	$A+\overline{A}=1$	Complementarity
5.	$A\cdot 0=0$	Operations with 0 and 1
6.	$A\cdot 1=A$	Operations with 0 and 1
7.	$A\cdot A=A$	Idompotent
8.	$A\cdot \overline{A}=0$	Complementarity
9.	$\overline{\overline{A}}=A$	Involution
10.	$A+B=B+A$	Commutative
11.	$A\cdot B=B\cdot A$	Commutative
12.	$A+(B+C)=(A+B)+C=A+B+C$	Associative
13.	$A\cdot(B\cdot C)=(A\cdot B)\cdot C=A\cdot B\cdot C$	Associative
14.	$A\cdot(B+C)=(A\cdot B)+(A\cdot C)$	Distributive
15.	$A+(B\cdot C)=(A+B)\cdot(A+C)$	Distributive
16.	$A+(A\cdot B)=A$	Absorption
17.	$A\cdot(A+B)=A$	Absorption
18.	$(A\cdot B)+(\overline{A}\cdot C)+(B\cdot C)=(A\cdot B)+(\overline{A}\cdot C)$	Consensus
19.	$\overline{A+B+C+\ldots}=\overline{A}\cdot\overline{B}\cdot\overline{C}\cdot\ldots$	De Morgan
20.	$\overline{A\cdot B\cdot C\cdot\ldots}=\overline{A}+\overline{B}+\overline{C}+\ldots$	De Morgan
21.	$(A+\overline{B})\cdot B=A\cdot B$	Simplification
22.	$(A\cdot\overline{B})+B=A+B$	Simplification