

General Statement: In this exam, we assume that every input comes with its complement.

I. Adders/Subtractors: A Half Subtractor inputs two bits x and subtrahend y , and outputs the difference d and borrow-out $bout$, while a Full Subtractor inputs three bits x , subtrahend y and borrow-in bin , and outputs the difference d and borrow-out $bout$.

I.1. Write the truth tables of the Half Subtractor and the Full Subtractor. Describe those logic using minimal sum of products expressions.

HS

x	y	d	$bout$
0	0	0	0
1	0	1	1
2	1	1	0
3	1	0	0

$d = x \oplus y$
 $bout = xy'$

FS

x	y	bin	d	$bout$
0	0	0	0	0
1	0	0	1	1
2	0	1	1	1
3	0	1	0	1
4	1	0	1	0
5	1	0	0	0
6	1	1	0	0
7	1	1	1	1

$d = x \oplus y \oplus bin$
 $bout = x'bin + x'y + ybin$

I.2. Implement a Full Subtractor with only one Half Subtractor and a minimal two level NOR gates. Draw the logic diagram.

Q. Use HS for $x \neq y$, rewrite truth table

x	y	bin	d_u	b_w	d	$bout$
0	0	0	0	0	0	0
1	0	0	1	0	1	1
2	0	1	1	1	1	1
3	0	1	1	1	0	1
4	1	0	1	0	1	0
5	1	0	1	0	0	0
6	1	1	0	0	0	0
7	1	1	0	1	1	1

② Derive $d, bout$ as functions of b_u, d_u, b_w

③ From K-map. $d = d_u \oplus b_{in}$
 $bout = b_w + b_{in} d_u$

$x \rightarrow d$

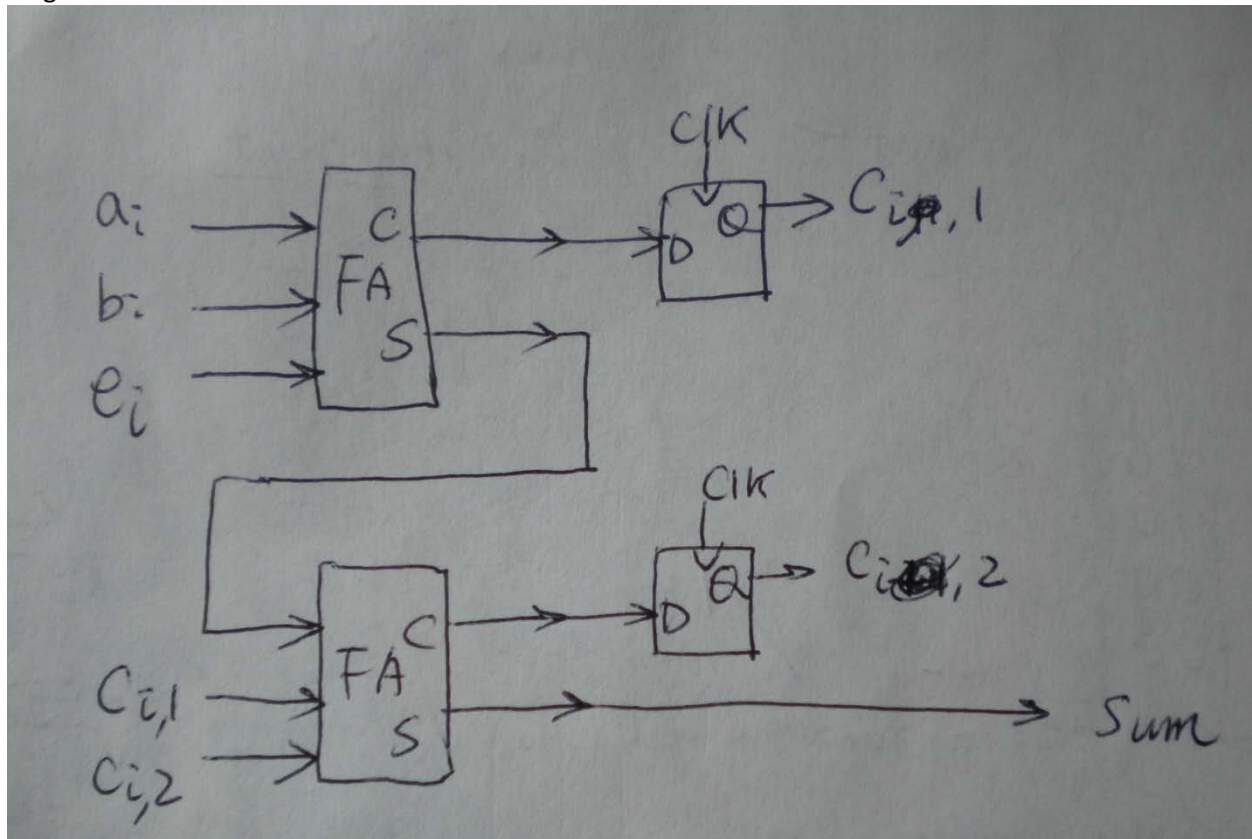
$y \rightarrow bout$

II. Serial Adders:

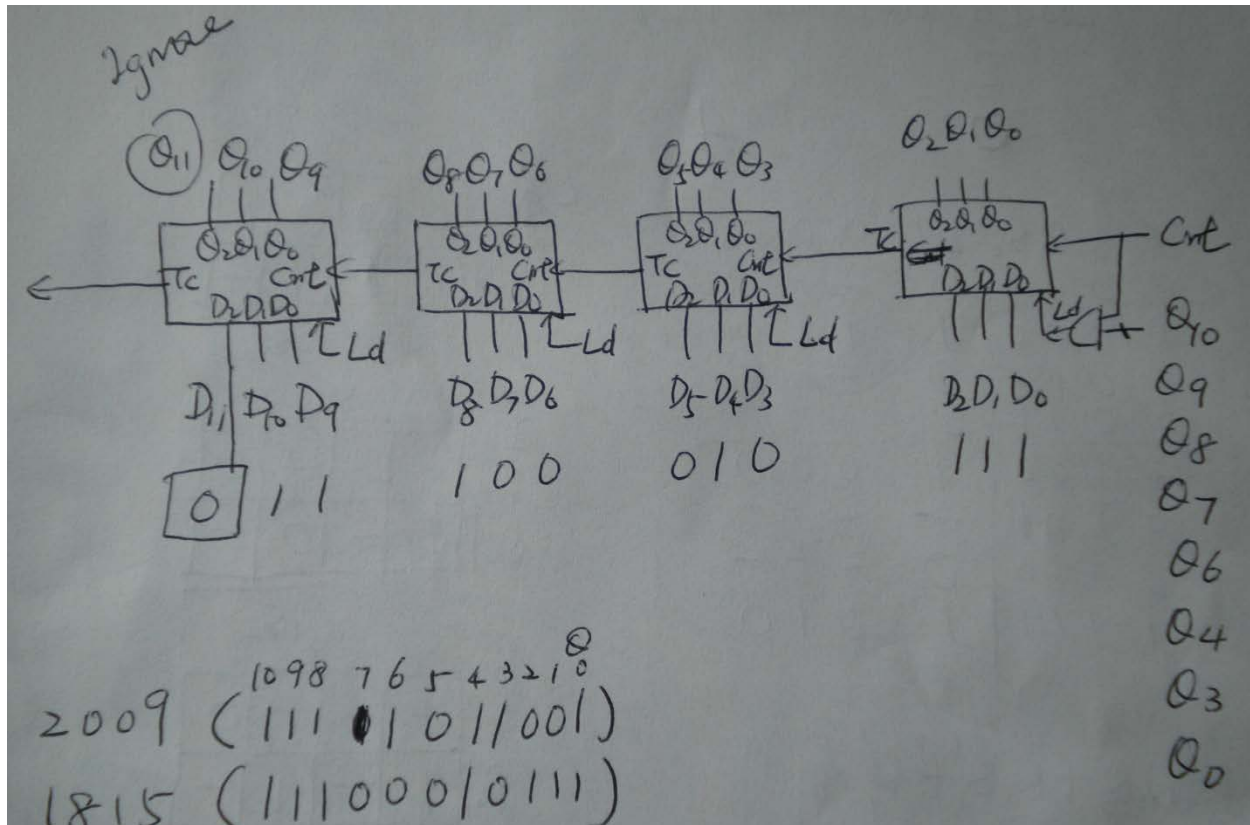
II.1. A sequential adder inputs a_i , b_i , the i 'th bit of two binary numbers in each clock cycle for $i = 0$ to $n - 1$ and outputs the sum s_i . Implement the adder with a T flip-flop, and a minimal AND .OR-NOT network (if the network is needed). Write the state table to describe the next states and outputs. Draw the schematic diagram.

Class Notes

II.2. A sequential three-at-a-time adder inputs a_i , b_i , e_i , the i 'th bit of three binary numbers in each clock cycle for $i = 0$ to $n - 1$ and produces the sum s_i . Implement the adder with a minimal number of Full Adders and two D flip-flops. Other logic components are not needed, nor available. Draw the logic diagram



III. Counters: A Modulo-8 Counter reads data (D_2, D_1, D_0), load LN , count enable CNT , and generates state (Q_2, Q_1, Q_0) and terminal count TC . Design a counter that counts from 1815 to 2009 using four Modulo-8 Counters and minimal AND-OR-NOT networks. Draw the schematic diagram



IV. Design a counter with a repeated output sequence 0, 1, 6, 7, 2, 3, 4 with a Modulo-8 Counter and a minimal AND-OR-NOT network. Write the Boolean expression and draw the schematic diagram.

Load 6 when state = 1
 " 2 " " 7
 " 6 " " 4

Q ₂ Q ₁ Q ₀	Ld	D ₂ D ₁ D ₀
0 0 0	0	- - -
1 0 0	1	1 1 0
2 0 1	0	- - -
3 0 1	0	- - -
4 1 0 0	1	0 0 0
5 1 0 1	-	- - -
6 1 1 0	0	- - -
7 1 1 1	1	0 1 0

$L_d = Q_2 Q_0 + Q_2 Q_1' + Q_1' Q_0$
 $D_2 = Q_2'$
 $D_1 = Q_0$
 $D_0 = 0$

Karnaugh maps for Ld, D₂, D₁, and D₀ are shown on the right.

Schematic diagram of the counter:

V. System Designs:

Implement the following algorithm:

Alg(X,Y,Z,start,U,done);

Input X[7:0], Y[7:0], start;

Output U[7:0], done;

Local-object A[7:0], B[7:0], C[7:0];

S1: If start' goto S1;

S2: $A \leftarrow X \parallel B \leftarrow Y \parallel C \leftarrow (00001111) \parallel done \leftarrow 0$;

S3: $C \leftarrow Inc(C) \parallel$ If $C'[7]$ then $A \leftarrow Add(A, B)$ else $A \leftarrow Add(A, -B)$;

S4: $B \leftarrow Inc(B) \parallel$ If $B'[7]$ goto S3;

S5: $U \leftarrow A \parallel done \leftarrow 1 \parallel$ goto S1;

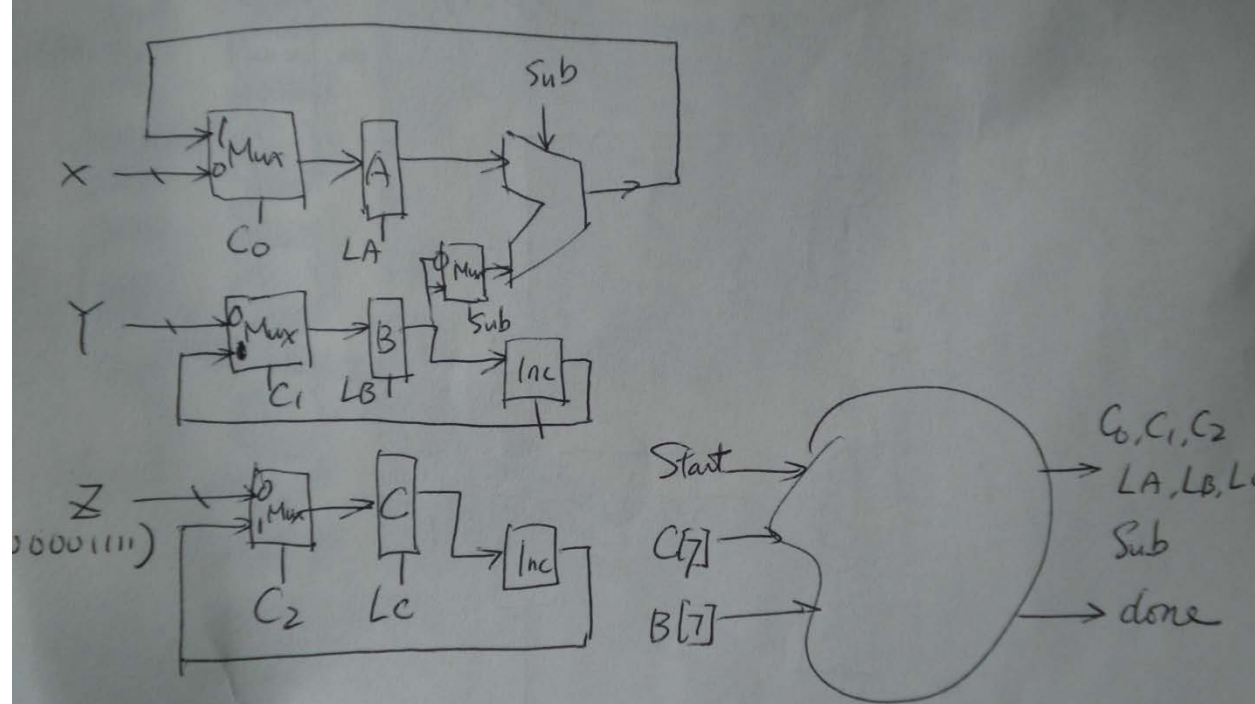
End Alg

V.1. Design a data subsystem that is adequate to execute the algorithm.

V.1.1. Use a table to list the instructions and corresponding components for data path subsystem.

Instruction	Comp
$A \leftarrow X$	Reg
$B \leftarrow Y$	Reg
$C \leftarrow (00001111)$	Reg
$C \leftarrow Inc(C)$	Inc
$A \leftarrow A+B$	Add
$A \leftarrow A-B$	Add(2's comp)
$B \leftarrow Inc(B)$	Inc
$U \leftarrow A$	Wires

V.1.2. Draw the schematic diagram to show the data path subsystem. Label the inputs, outputs, and control signals of all components.

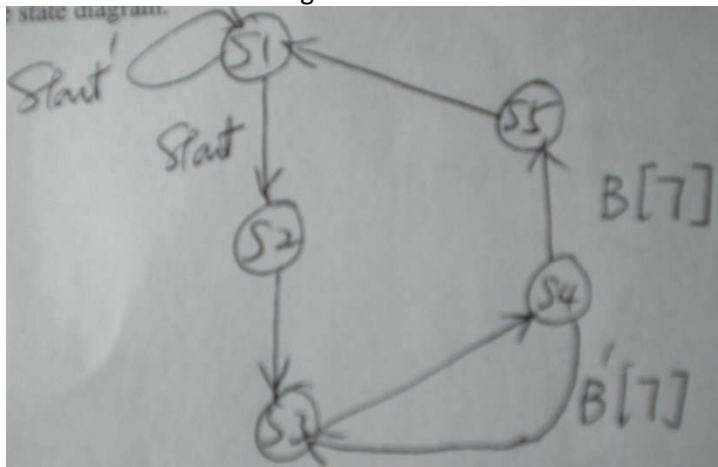


V.2. Design the control subsystem.

V.2.1. Use a table to list the value of control signals for every state.

	C_0	C_1	C_2	LA	LB	LC	Sub	done
S1	~~0~~	~~0~~	~~0~~	D	D	D	—	1
S2	0	0	0	1	1	1	—	0
S3	1	—	1	1	0	1	$C[7]$	0
S4	—	1	—	0	1	0	—	0
S5	—	—	—	0	0	0	—	1

V.2.2. Draw the state diagram.



V.2.3. Implements the control subsystem with a one hot encoding design. Draw the logic diagram.

