Department of Computer Science & Engineering

Course No : CSE3110

Course Title : Digital System Design

Experiment No : 01

Experiment Name : ALU Implementation

Date of Submission : 3 December 2022

Submitted to : Md Raqibul Hasan & Anika Rahman

Submitted by –

Group No : 03(A2)

Name : Aurprita Mahmood

Sumaiya Shejin

Ziyan Shirin Raha

Meherin Sultana

Yasir Arafah Prince

Id : 20200104035, 20200104043, 20200104033, 20200104036, 20200104042

Section : A

Problem Statement: The problem statement is given below:-

S_2	S_1	S_0	Output	Function
1	0	1	Ai - Bi	Subtract
1	0	0	Ai + 1 + 1	Transfer A with Canny
0	0	1	Ai - 1	Decrement A
0	0	0	Ai - Bi - 1	Subtract with Borrow
1	1	X	Ai'	Complement A
0	1	X	Ai.Bi	AND

Introduction: The main purpose of this experiment is to design a ALU (Arithmetic and Logic Unit). Our ALU is 4 bit. This combinational circuit named ALU performs important functions like arithmetic and logic operators. Generating functions will be the goal of our experiment. For generating the function at first we will have to derive equation from the table given above using the selector bits for three inputs. After the k-map we have to simplify the equation then with the help of the software named proteus we will implement these functions. Thus the ALU will generate the output of the given functions.

Function Table: The function table is given below :-

S_2	S_1	S_0	X_i	Y_i	Z_i	Z_0/Cin	Output
0	0	0	A_i	$\overline{B_i}$	C_i	0	$A_i - B_i - 1$
0	0	1	A_i	All 1	C_i	0	$A_i - 1$
0	1	0	$A_i + \overline{B_i}$	$\overline{B_i}$	0	0	$A_i . B_i$
0	1	1	$A_i + \overline{B_i}$	$\overline{B_i}$	0	0	$A_i . B_i$
1	0	0	A_i	All 1	C_i	1	$A_i + 1 + 1$
1	0	1	A_i	$\overline{B_i}$	C_i	1	$A_i - B_i$
1	1	0	$\overline{A_i}$	0	0	0	A_i'
1	1	1	$\overline{A_i}$	0	0	0	A_i'

K-map: For Z_0,

	$\overline{S_1}\overline{S_0}$	$\overline{S_1}S_0$	S_1S_0	$S_1\overline{S_0}$
$\overline{S_2}$	0	0	0	0
S_2	1	1	0	0

$Z_0 = S_2\overline{S_1}C_i$

For Z_i,

	$\overline{S_1}\overline{S_0}$	$\overline{S_1}S_0$	S_1S_0	$S_1\overline{S_0}$
$\overline{S_2}$	C_i	C_i	0	0
S_2	C_i	C_i	0	0

$Z_i = C_i\overline{S_1}$

For Y_i,

S_2 \ S_1S_0	$\bar{S}_1\bar{S}_0$	$\bar{S}_1S_0$	S_1S_0	$S_1\bar{S}_0$
$\bar{S}_2$	$\bar{B}_i$	1	$\bar{B}_i$	$\bar{B}_i$
S_2	1	$\bar{B}_i$	0	0

$$Y_i = \bar{S}_2\bar{S}_0\bar{B}_i + \bar{S}_2\bar{S}_1S_0 + \bar{S}_2S_1\bar{B}_i + S_2\bar{S}_1\bar{S}_0 + S_2\bar{S}_1S_0\bar{B}_i$$
$$= \bar{S}_i\left(\bar{S}_2S_0 + S_2\bar{S}_0\right) + \bar{B}i\left(\bar{S}_2\bar{S}_0 + \bar{S}_2S_1 + S_2\bar{S}_iS_0\right)$$
$$= \bar{S}_i\left(S_2 \oplus S_0\right) + \bar{B}i\left(\bar{S}_2\bar{S}_0 + \bar{S}_2S_1 + S_2\bar{S}_1S_0\right)$$

For X_i,

S_2 \ S_1S_0	$\bar{S}_1\bar{S}_0$	$\bar{S}_1S_0$	S_1S_0	$S_1\bar{S}_0$
$\bar{S}_2$	A_i	A_i	$A_i + \bar{B}_i$	$A_i + \bar{B}_i$
S_2	A_i	A_i	$\bar{A}_i$	$\bar{A}_i$

$$= A_i\bar{S}_1 + \bar{S}_2S_1\left(A_i + \bar{B}_i\right) + \bar{A}_iS_2S_i$$

Result:

For subtract with borrow operation ——

Selector			Input (A)				Input (B)				Output				
S_2	S_1	S_0	A_3	A_2	A_1	A_0	B_3	B_2	B_1	B_0	C_{out}	F_3	F_2	F_1	F_0
			1	1	0	0	1	0	0	0	1	0	0	1	1
0	0	0	1	1	1	0	1	0	0	1	1	0	1	0	0
			1	0	0	1	0	1	1	1	1	0	0	0	1

For decrement operation ——

Selector			Input (A)				Input (B)				Output				
S_2	S_1	S_0	A_3	A_2	A_1	A_0	B_3	B_2	B_1	B_0	C_{out}	F_3	F_2	F_1	F_0
			0	1	1	1	0	1	0	0	1	0	1	1	0
0	0	1	1	1	0	1	0	1	1	0	1	1	1	0	0
			1	0	1	1	1	1	0	0	1	1	0	1	0

For AND operation ——

Selector			Input (A)				Input (B)				Output				
S_2	S_1	S_0	A_3	A_2	A_1	A_0	B_3	B_2	B_1	B_0	C_{out}	F_3	F_2	F_1	F_0
			1	1	0	1	1	1	0	0	0	1	1	0	0
0	1	X	0	1	1	1	1	1	0	1	0	0	1	0	1
			1	0	1	1	1	0	1	0	0	1	0	1	0

For transfer with carry operation:—

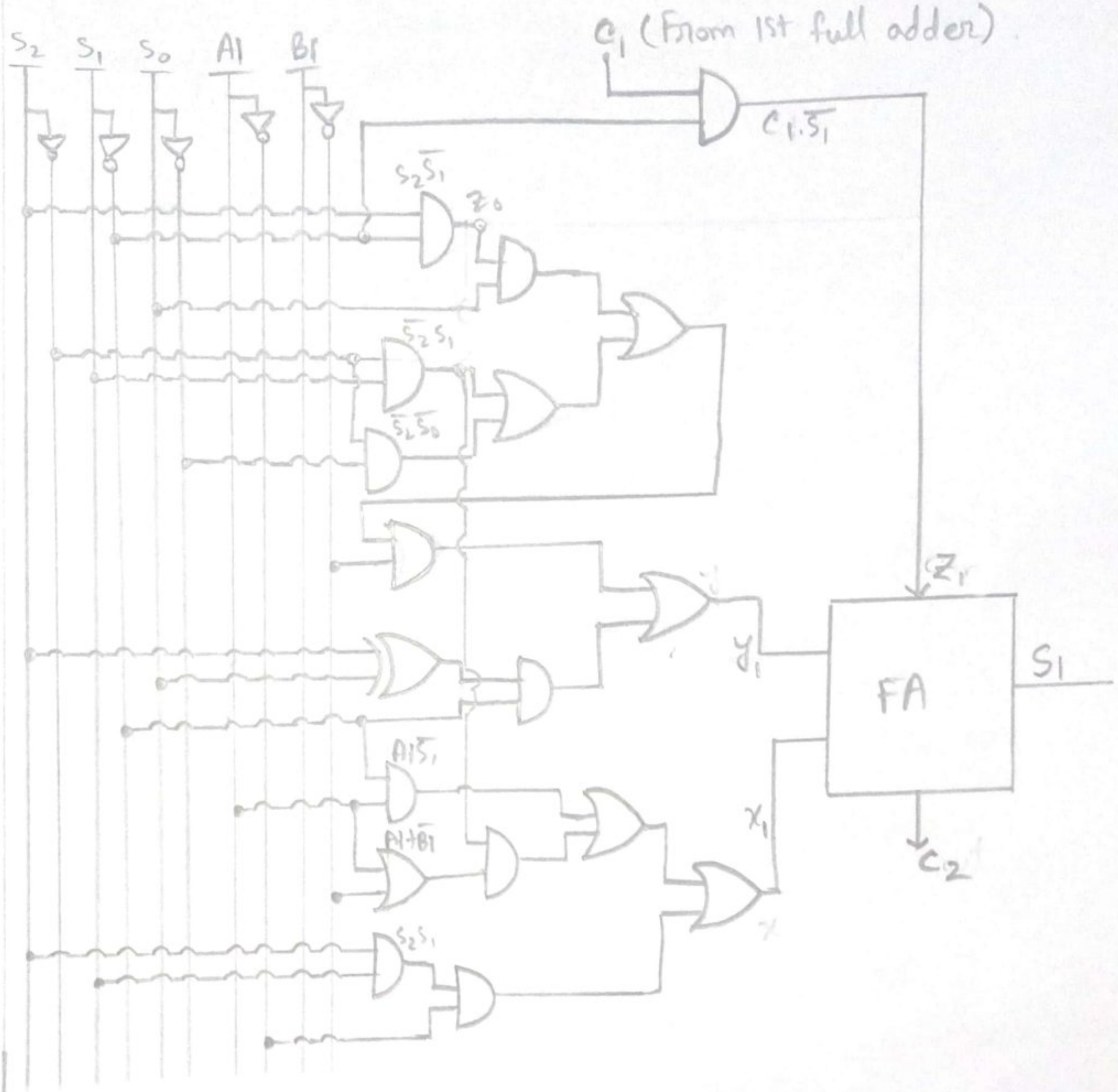
Selector			Input (A)				Input (B)				Output				
S_2	S_1	S_0	A_3	A_2	A_1	A_0	B_3	B_2	B_1	B_0	Cout	F_3	F_2	F_1	F_0
			0	1	1	0	0	0	0	1	1	0	1	1	0
1	0	0	1	0	1	0	0	0	0	1	1	1	0	1	0
			0	0	1	0	0	0	0	1	1	0	0	1	1

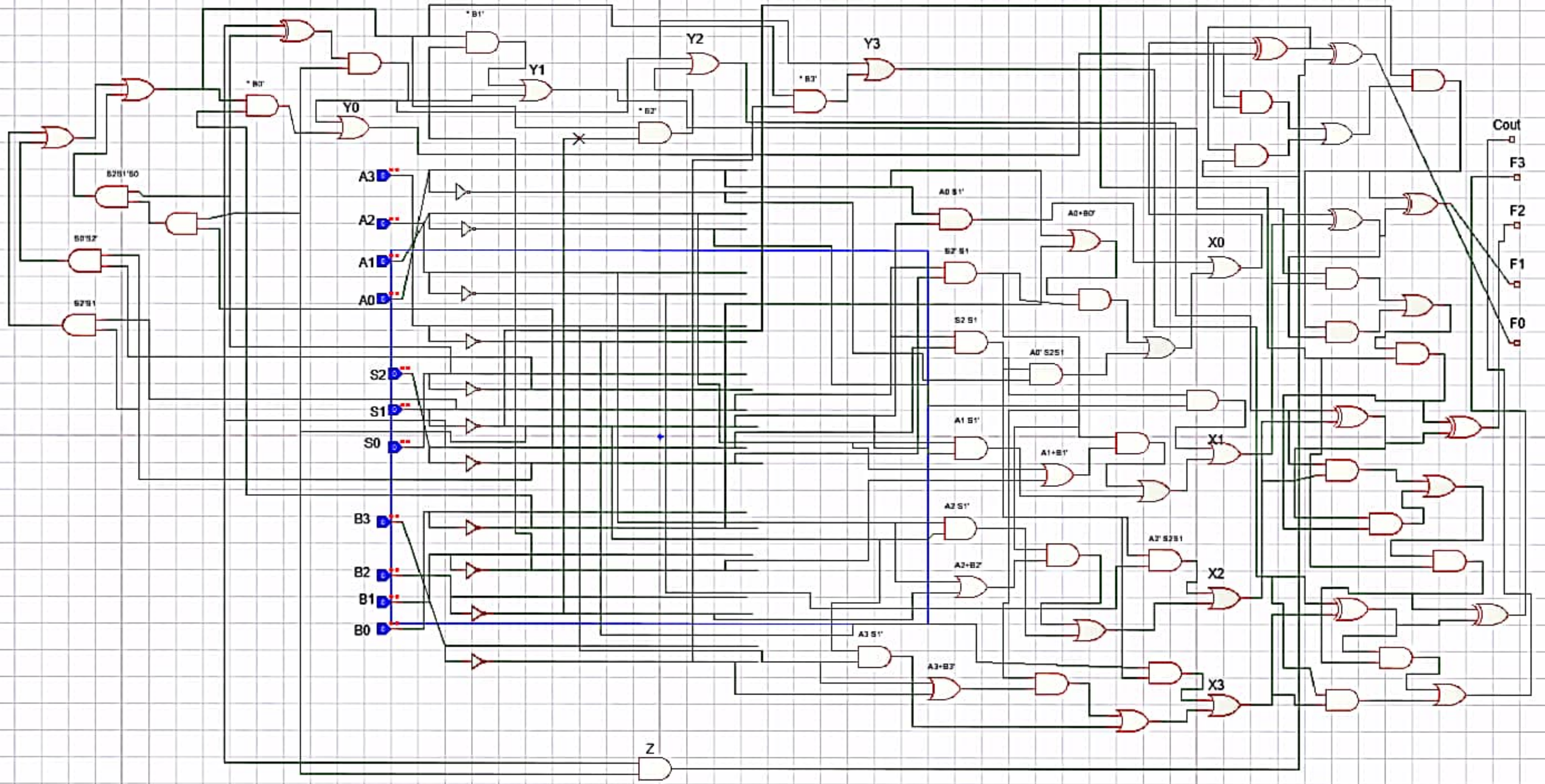
For subtract operation—

Selector			Input (A)				Input (B)				Output				
S_2	S_1	S_0	A_3	A_2	A_1	A_0	B_3	B_2	B_1	B_0	Cout	F_3	F_2	F_1	F_0
			1	0	1	1	0	1	0	1	1	0	1	1	0
1	0	1	1	1	1	1	0	0	1	1	1	1	1	0	0
			0	1	0	1	0	1	0	0	1	0	0	0	1

For complement operation—

Selector			Input (A)				Input (B)				Output				
S_2	S_1	S_0	A_3	A_2	A_1	A_0	B_3	B_2	B_1	B_0	Cout	F_3	F_2	F_1	F_0
			1	0	1	0	0	0	0	0	0	0	1	0	1
1	1	X	0	0	1	1	0	0	0	0	0	1	1	0	0
			1	0	0	1	0	0	0	0	0	0	1	1	0


S_2 S_1 S_0 A1 B1
C_1 (From 1st full adder)
$C_1.\overline{S_1}$
$S_2\overline{S_1}$
Z_0
$\overline{S_2}S_1$
$\overline{S_2}\overline{S_0}$
Z_1
Y_1
FA
S_1
$A1\overline{S_1}$
$A1+\overline{B1}$
X_1
C_2
S_2S_1


Y0
Y1
Y2
Y3
A3
A2
A1
A0
S2
S1
S0
B3
B2
B1
B0
Z
X0
X1
X2
X3
Cout
F3
F2
F1
F0

Conclusion : The proteus software that we use to execute the stimulation have some features, and on that account the instrumental errors occur. At first our ALU (Arithmetic and Logical Unit was giving us some inaccurate results on outputs for logical opercations however for some reason it was working accurately for ~~out~~ our arithmetic opercations. Thus we had to implement the circuit using direct equations without ~~simplyf~~ simplifying it and this time the functions were working correctly. But after practical implementation there might be an errcor in fact fact the IC's we use can ~~be~~ also become morce prone to errcor.