Report on Calculator Design and Implementation

A Project of Digital Electronics Lab (EEE 4308)

Submitted by:

Group π

1. Md. Ashaduzzaman Niloy, ID: 160021002
2. Sakif Ahmed, ID: 160021028
3. Ahmad Azuad Yaseer, ID: 160021036
4. Tahmid Zaman Tahi, ID: 160021038
5. Md. Minhajul Islam, ID: 160021040
6. Elamin Muaz Hussein Elamin, ID: 160020002

Islamic University of Technology (IUT)
A subsidiary organ of OIC
Board Bazar, Gazipur-1704, Bangladesh

Acknowledgement

We tried our level best to complete the project. But due to our own lack of proper knowledge on specific areas like PCB designing and debugging of PCB, we couldn't complete cent percent. But we learnt a lot from this project which, we think, will help us to work properly in our upcoming works. We worked together as a team and everyone tried their best to bring the project this far. By the grace of Almighty Allah we have done what was in our capabilities.

Contents

Introduction:

The project was to implement a calculator using combinational circuit which would have the capability to carry out addition, subtraction, multiplication and division operations.

Problem Formulation:

The project should contain three units.

- Input Unit
- Arithmetic Unit
- Output Unit

Input Unit: Input must be taken from number keypad. So, input will be two 1 digit BCD numbers taken after one another.

Arithmetic Unit: The Arithmetic unit will take two BCD numbers $A=A_4A_3A_2A_1$ and $B=B_4B_3B_2B_1$ where the maximum possible value of A & B is $(9)_{10}$. Pressing specific functional key will carry out following operations.

- Addition operation between A & B (A+B), giving 2 digit BCD sum(S) as output; where max possible output can be 18.
- Subtraction operation for subtracting B from A (A-B), giving 1 digit BCD difference (D) as output & sign flag(SF) to indicate a (-) negative sign in case of B>A.
- Multiplication operation for multiplying A and B (A×B), giving a 2 digit BCD number (M) as output; where max possible output can be 81.
- Division operation for dividing A by B (A÷B),giving 1 digit BCD number(Q) as output; where max possible output can be 9 as only quotient should be displayed.

Output Unit: The output should be BCD numbers visualized using two 7-segment display.

Designing Procedure:

At first we discussed amongst ourselves, figured out the algorithm then worked out the design of subsequent logic circuits. Then we implemented the circuits in Proteus to check our logic. Finally after software debugging we made the PCB designs. After that we bought necessary equipment and completed hardware implementation.

- ***Adder & Subtractor designing:***

We used two 4 bit binary full adders (IC 74LS83) to accomplish this task. We used a switch input to perform addition and subtraction in same logic circuit. The switch connection was connected to the C_0(Carry In) of the first adder and with the switch gate 1 XOR gates (IC 74HC86) along with binary of the second decimal input ($B=B_4B_3B_2B_1$) whose purpose was if the value of switch gate was 0 then it would add without inverting the input B. Otherwise, it would make 1's complement of input B and perform 2's complement of B in first adder. Similarly the second adder was used-

When Switch=0:- To check if the output of first adder was invalid in BCD & if so then to make it valid 6 was added with output of first adder in the second one.

When Switch=1:- To check whether the C_4(Carry Out) of the first adder is 0 or 1, if 0 then the output of first adder was inverted by the help of another set of switch gate 2 (XOR gates) otherwise the outputs of first adder remained unchanged.

The output of second adder was the final output going into the first BCD to 7 segment converter(IC 7447) by output panel and the second converter was connected with the logic of adding 6 after the first adder if the number was invalid in BCD. And the inverted (by NOT gate IC 74HC04) AND(IC 74HC08) output of the switch and the inverted output of the C_4 of first adder. This is to give the negative sign in the 7 segment display so it is directly connected to it.

- ***Mechanism of checking if the BCD is invalid:***

BCD is represented by 4 binary bits. Now BCD only represents up to 9. But with 4 bits we can represent up to 15. So in case of BCD rest 6 (from 1010 to 1111) values are invalid. So to make them valid we have to add binary of 6 (0110) with the output.

C_4	S_4	S_3	S_2	S_1	Checker Output		C_4	S_4	S_3	S_2	S_1	Checker Output
0	0	0	0	0	0		1	0	0	0	0	1
0	0	0	0	1	0		1	0	0	0	1	1
0	0	0	1	0	0		1	0	0	1	0	1
0	0	0	1	1	0		1	0	0	1	1	×
0	0	1	0	0	0		1	0	1	0	0	×
0	0	1	0	1	0		1	0	1	0	1	×
0	0	1	1	0	0		1	0	1	1	0	×
0	0	1	1	1	0		1	0	1	1	1	×
0	1	0	0	0	0		1	1	0	0	0	×
0	1	0	0	1	0		1	1	0	0	1	×
0	1	0	1	0	1		1	1	0	1	0	×
0	1	0	1	1	1		1	1	0	1	1	×
0	1	1	0	0	1		1	1	1	0	0	×
0	1	1	0	1	1		1	1	1	0	1	×
0	1	1	1	0	1		1	1	1	1	0	×
0	1	1	1	1	1		1	1	1	1	1	×

S_4S_3 \ S_2S_1	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	0	0	1	1

$C_4 = 0$

S_4S_3 \ S_2S_1	00	01	11	10
00	1	1	×	1
01	×	×	×	×
11	×	×	×	×
10	×	×	×	×

$C_4 = 1$

Fig.1. Truth Table & K-Map Simplification of Checker Circuit

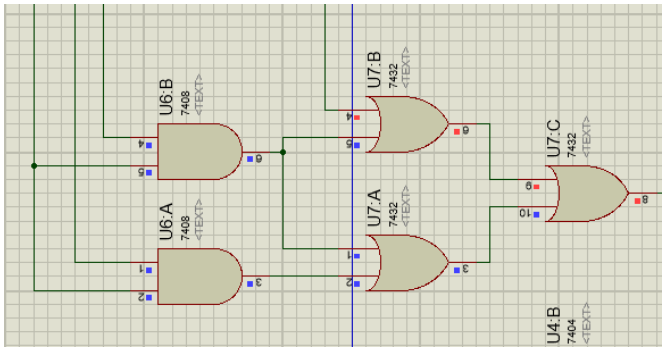


Fig.2: Checker Circuit

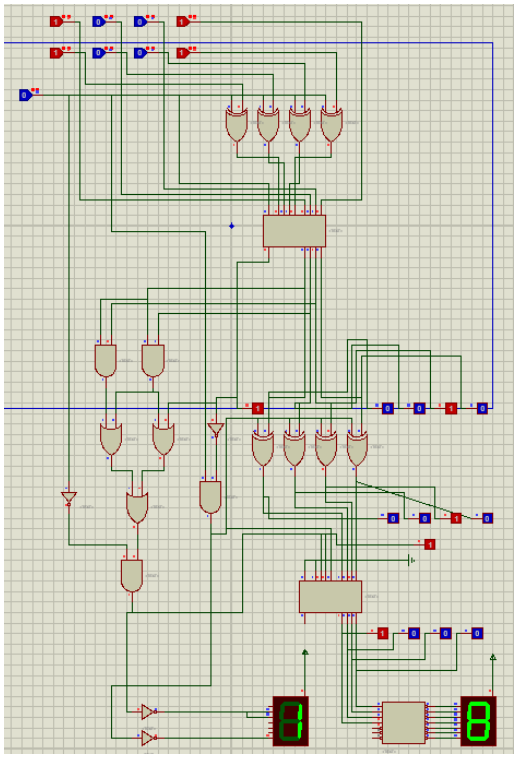


Fig.3: Adder Operation

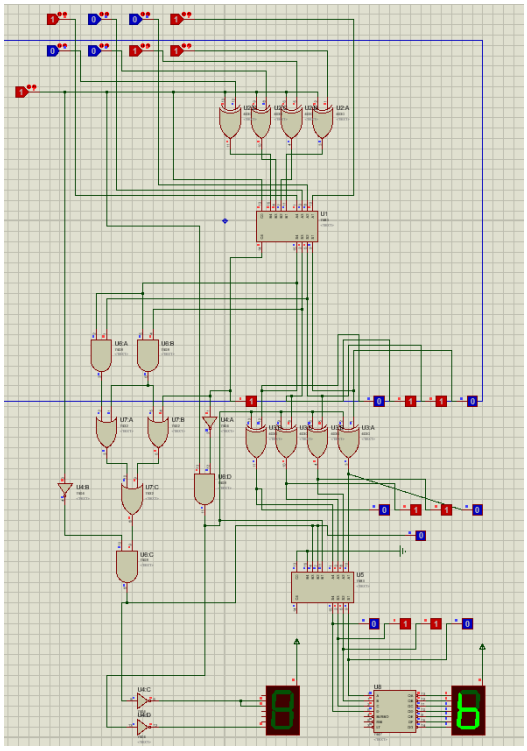


Fig.4: Subtraction Operation (A>B)

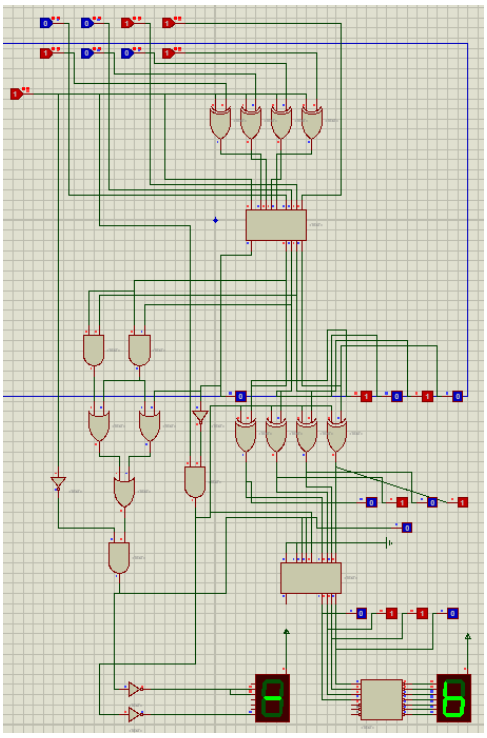


Fig.5: Subtraction Operation (A<B)

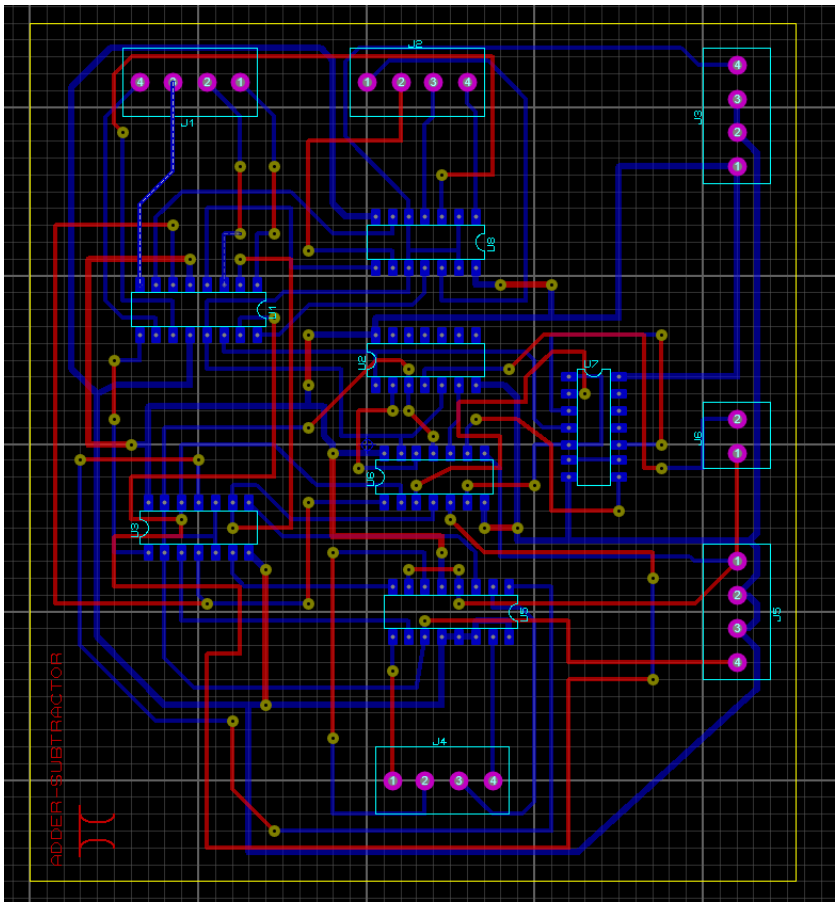


Fig.6: PCB Layout of Adder-Subtractor Unit

- ***Multiplier designing:***

For multiplier we took two BCD numbers, A ($A_3A_2A_1A_0$) and B($B_3B_2B_1B_0$). Then we performed the following operation using three 4 bit Binary Full Adders (IC 74LS83) and AND gates (IC 74HC08).

			AND Operation				
							A0
							B0
			GND	A0	A0	A0	
			GND	B3	B2	B1	
				First Adder			
		GND	A1	A1	A1	A1	
		GND	B3	B2	B1	B0	
			Second Adder				
	GND	A2	A2	A2	A2		
	GND	B3	B2	B1	B0		
		Third Adder					
	A3	A3	A3	A3			
	B3	B2	B1	B0			
Output	P6	P5	P4	P3	P2	P1	P0

Fig.7: Working Diagram of Multiplication

In each Adder the C_0 (Carry In) was grounded. In this way the binary result of the multiplication was obtained. In this case the maximum output that we would get is 81 (Binary= 1010001). We can represent this number by 7 bits, so the last bit will be always grounded.

As our display can only show BCD output. So we had to convert 7 bit Binary number to BCD. For that purpose we designed another circuit by which this operation can be happened. This converter circuit is separated from main multiplier PCB board. Output of Multiplier PCB Unit is considered as the input of converter circuit. This process is discussed later on.

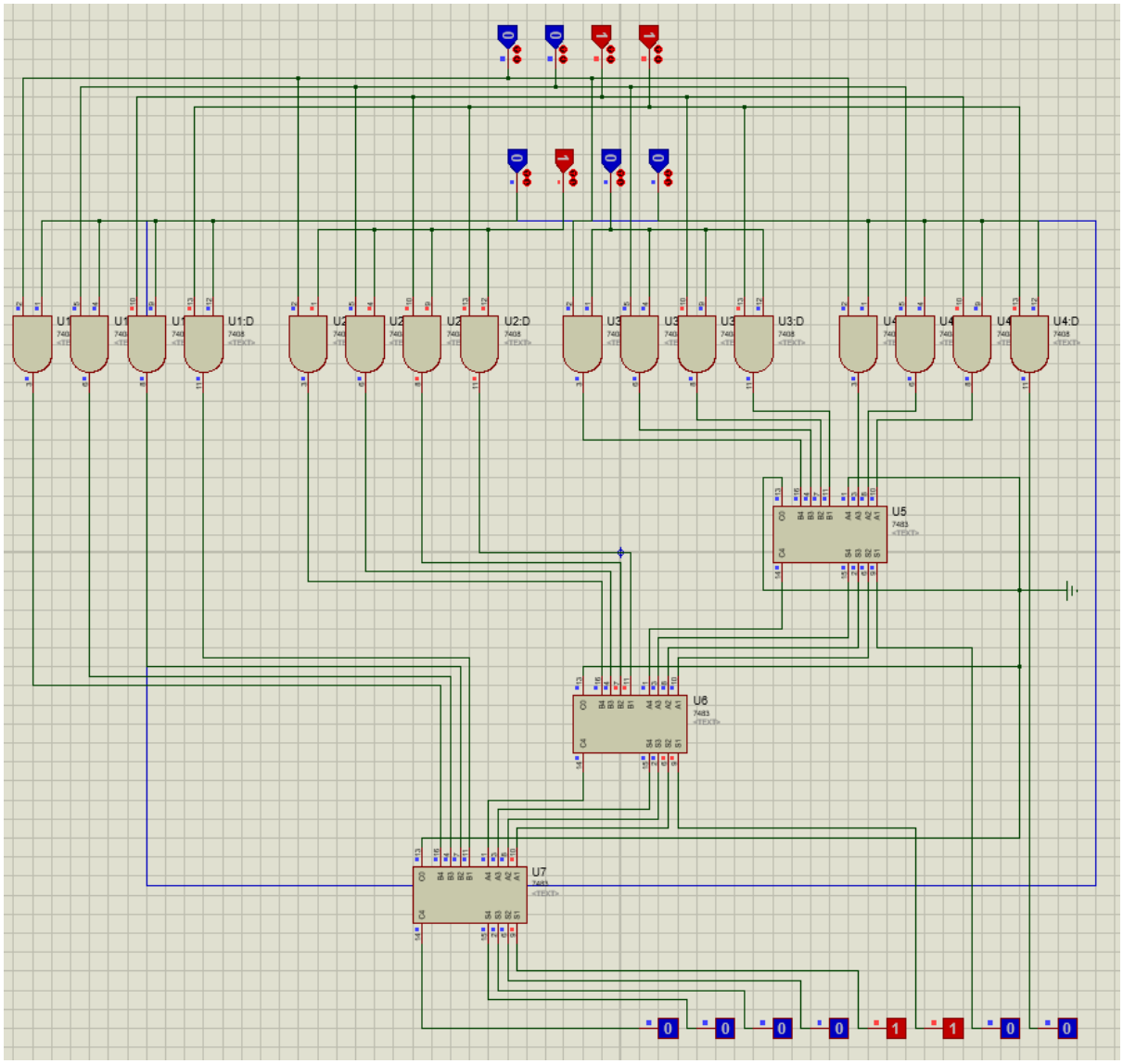


Fig.8: Multiplier Operation

- ***Binary to BCD Converter:***

For binary to BCD conversion, we used the shift add 3 logic. In this case, first the 7 bit number was taken and shifted such that the MSB (Most Significant Bit) was taken out at first and then the next bits simultaneously. If the binary was greater than binary of 4 (100) then the binary of 3 (011) was added and shifted. In this way, a total of 4 shifts would take place for 7 bits.

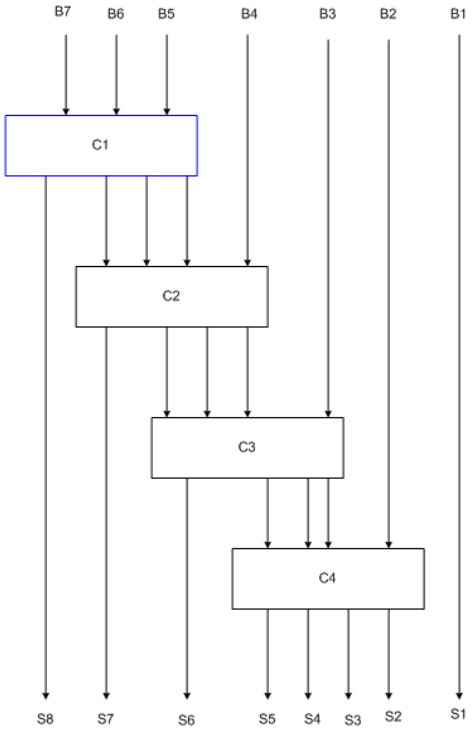


Fig.9: Flowchart Diagram

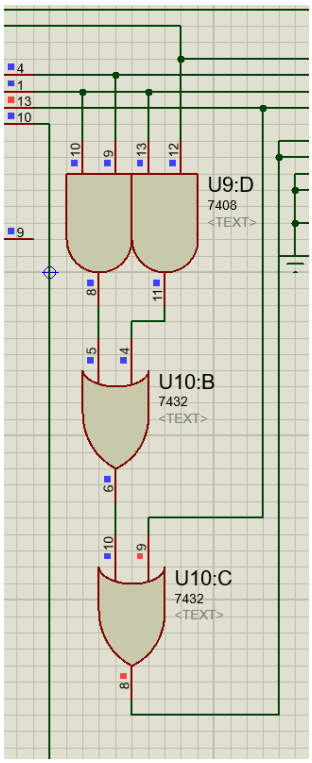


Fig.10: Checker Circuit to add 3

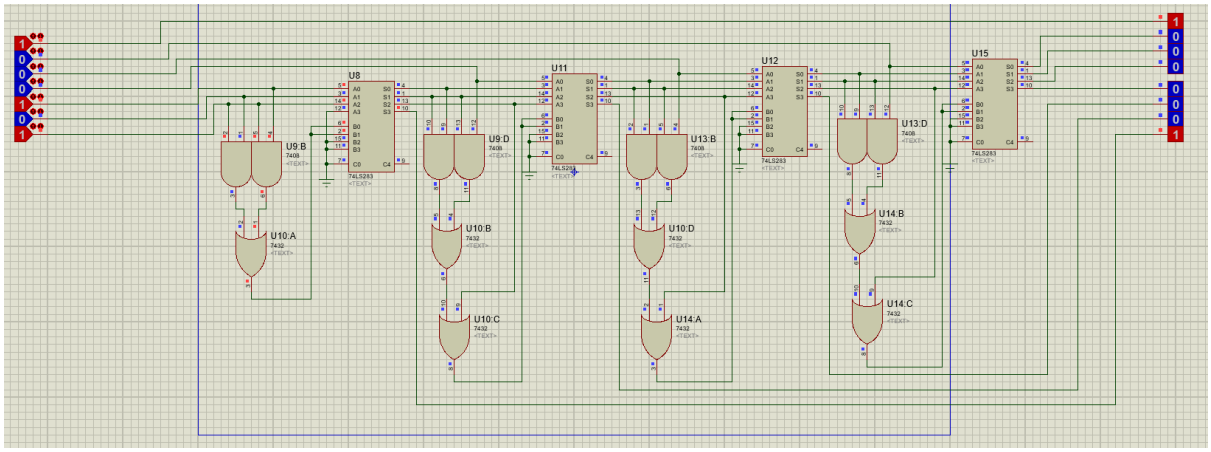


Fig.11: 7-bit Binary to BCD Converter

Normally the circuit would need Shift Left Registers to work according to the logic. But we were used to combinational logic at that time and since all our other circuits contained combinational logic, so we used combinational logic in this case too.

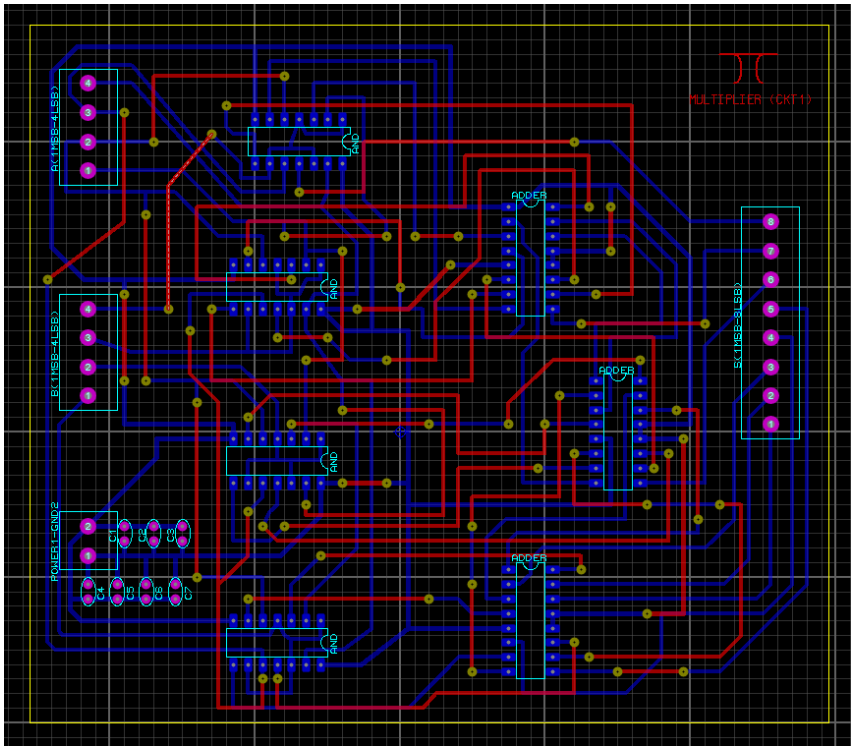


Fig.12: PCB Layout of Multiplier Unit

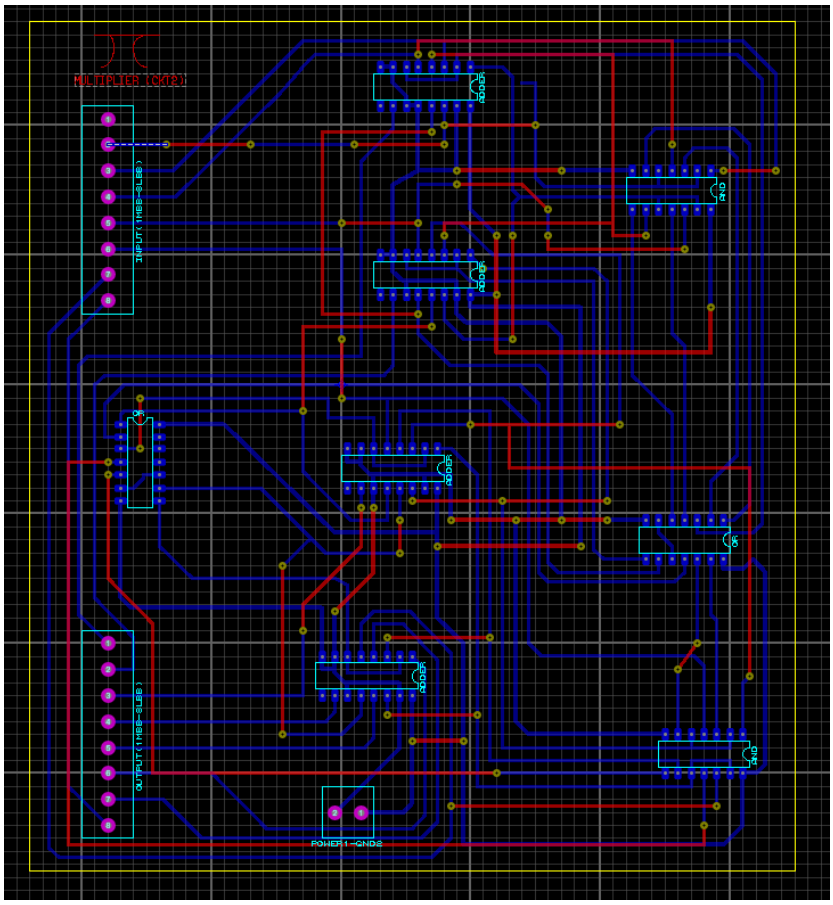


Fig.13: PCB Layout of Binary to BCD Unit

- ***Divider designing:***

In case of a divisor, we used Non-restoring method by the help of 4 bit Binary Full Adder(IC 74LS83) and XOR (IC 74HC86). We took two BCD numbers, first one as Dividend ($A=A_4A_3A_2A_1$) and the other one as Divisor ($B=B_4B_3B_2B_1$). The result was taken as Quotient ($Q=Q_4Q_3Q_2Q_1$).

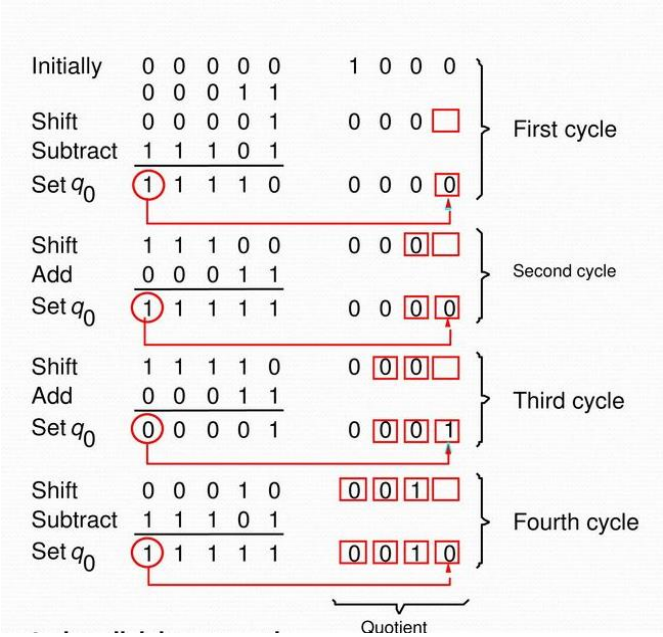


Fig.14: Working method of Division

According to this method, at first we set the quotient as 0 & shift the dividend 1 digit left aligned to divisor. If that portion of the dividend above the divisor is greater than or equal to the divisor, then we need to subtract divisor from that portion of the dividend and concatenate 1 to the right hand end of the quotient else we should concatenate 0 to the right hand end of the quotient. We should repeat the process 4 times as we are dealing with only 4-digit binary numbers. Then we will found desired Quotient.

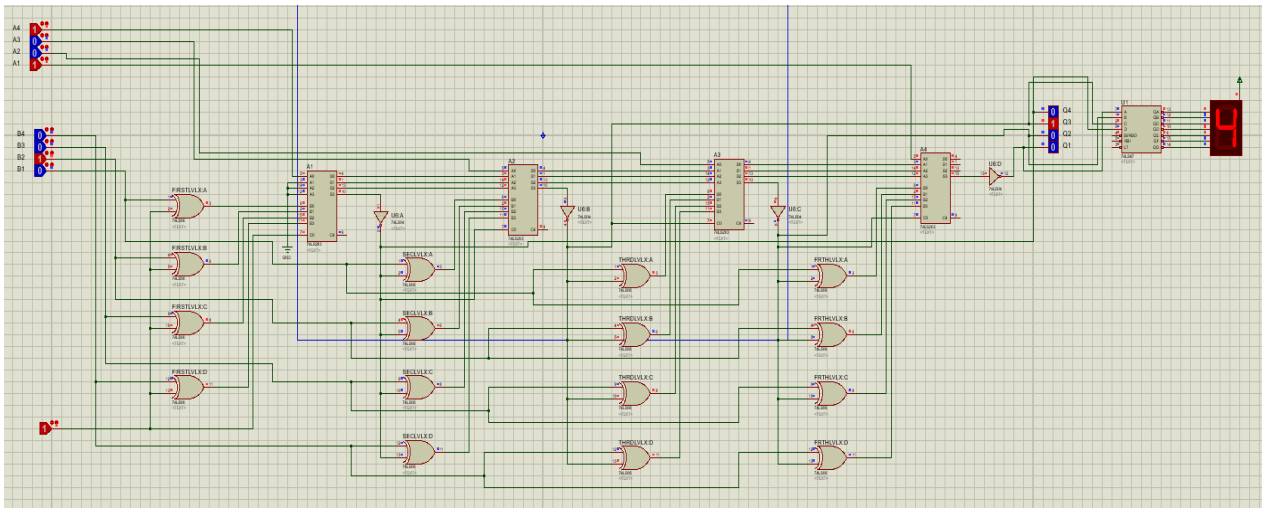


Fig.15: Divider Operation ($9_{10} \div 2_{10}=4_{10}$)

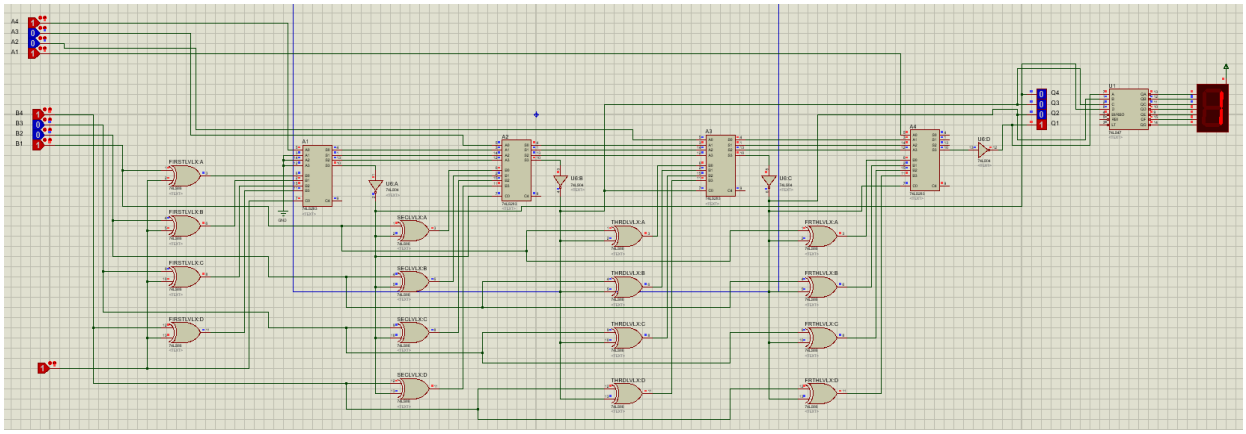


Fig.16: Divider Operation ($9_{10} \div 9_{10}=1_{10}$)

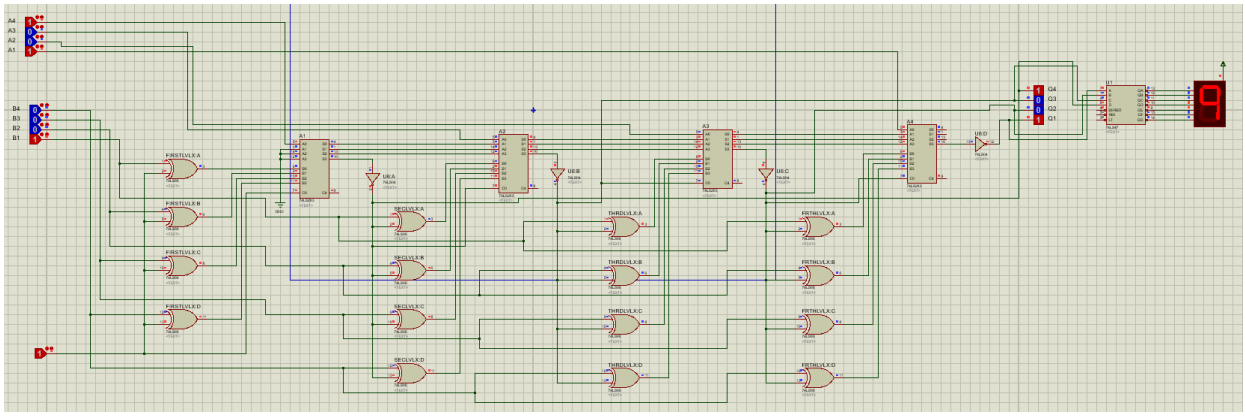


Fig.17: Divider Operation ($9_{10} \div 1_{10}=9_{10}$)

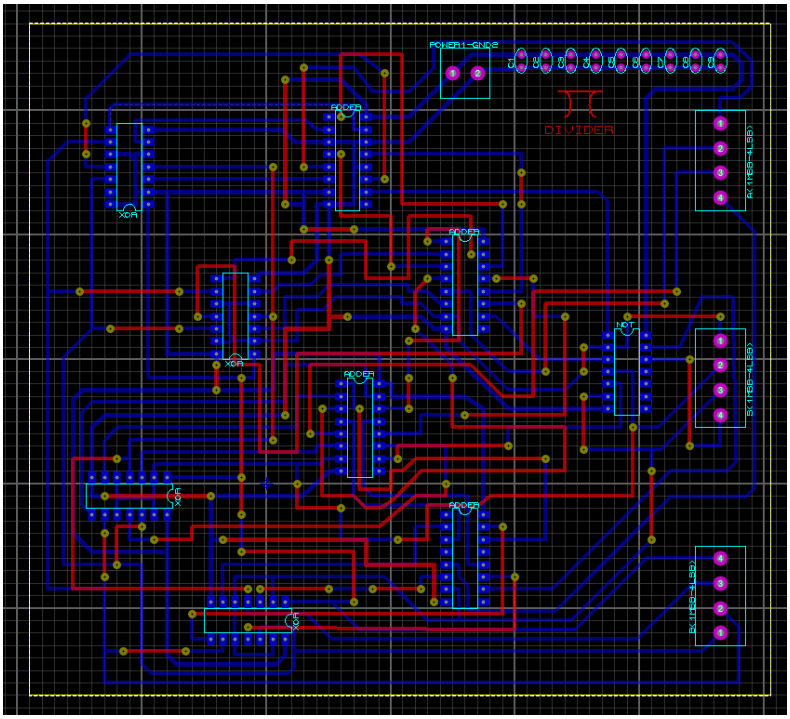


Fig.18: PCB Layout of Divider Unit

- ***Output Panel:***

In this unit, we have four 8 × 2 Multiplexers(IC 74153), two 4-bit Binary to BCD Converter (IC 7447) and two common anode 7-segment display. On each MUX we have 2 selection bit($J=J_1J_0$)= using which we can select which arithmetic unit's result will be shown on display.

Selection Bit	Secondary Selection Bit	Operation
$J_1=0; J_0=0$	Switch= 0	Adder
	Switch= 1	Subtraction
$J_1=0; J_0=1$		Multiplication
$J_1=1; J_0=0$		Division

Selection Bit ($J_1=1; J_0=1$) is not used here.

- ***Modification at the input side of Display:***

As all of our results are of 2-bits, we need two separate display. First one is for showing LSB of the results. Second one is for showing MSB of the results & Minus sign of subtraction operation when A-B=-R (as B>A). As first display is only showing numbers we need not to do any modification. But in second circuit , we need to modify because minus sign can not be implemented using binary to BCD converter.

On display there are 7 inputs for 7 segments. (a,b,c,d,e,f,g) . For minus sign only 'g' should on. As 7-segment display is active low device, to on a segment we need to provide zero. Now we designed our adder-subtractor unit such a way that a output pin will be active when minus result occurs.

Converter Output	Minus Output	Display Input
0	0	0
0	1	1
1	0	1
1	1	1

Fig.19: Truth table for each of 'a'-'f' pin

Converter Output	Minus Output	Display Input
0	0	0
0	1	0
1	0	1
1	1	0

Fig.20: Truth Table for 'g' pin

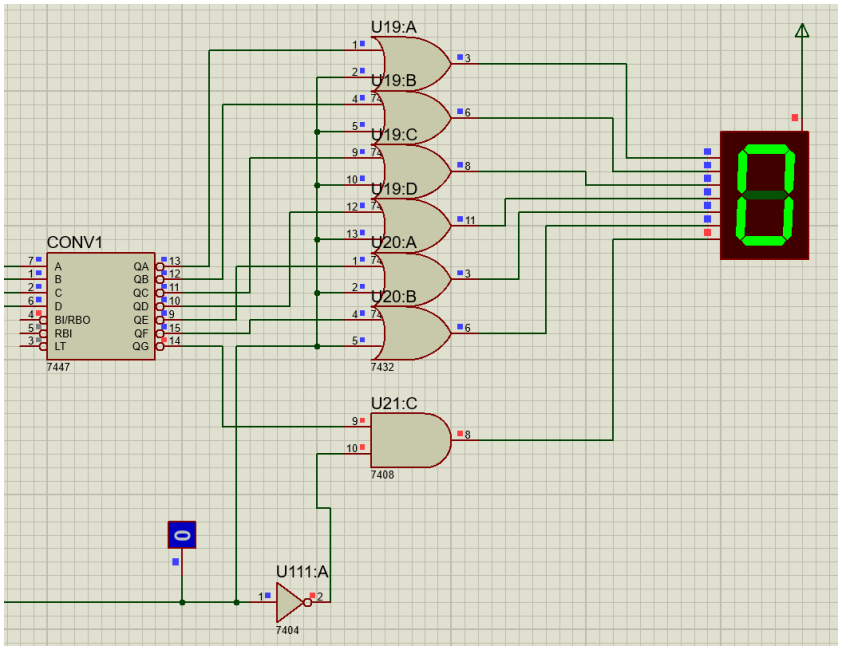


Fig.21: When Minus Output is 0

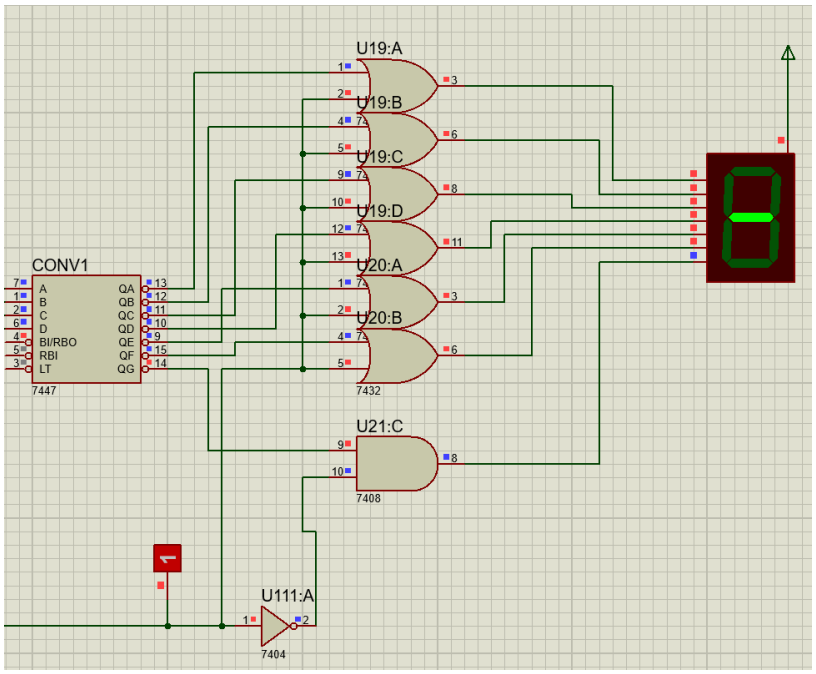


Fig.22: When Minus Output is 1

In Proteus, we constructed three user-made IC to check the whole calculator. Inside these ICs, one of them is Adder-Subtractor unit, one of them is Multiplier unit with converter & the last one is for Divider unit. In PCB , Some ports has been attached where output from each unit will enter.

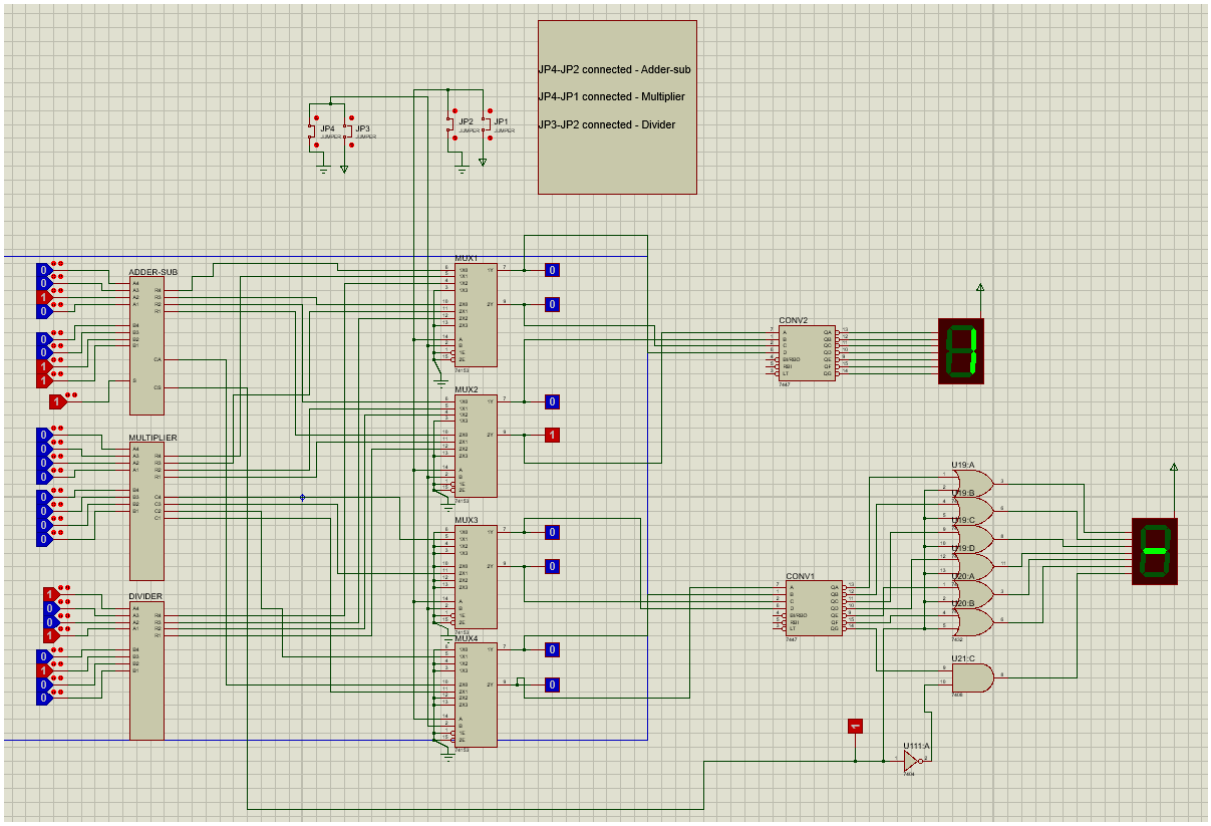


Fig.23: Working Schematic of Output Panel

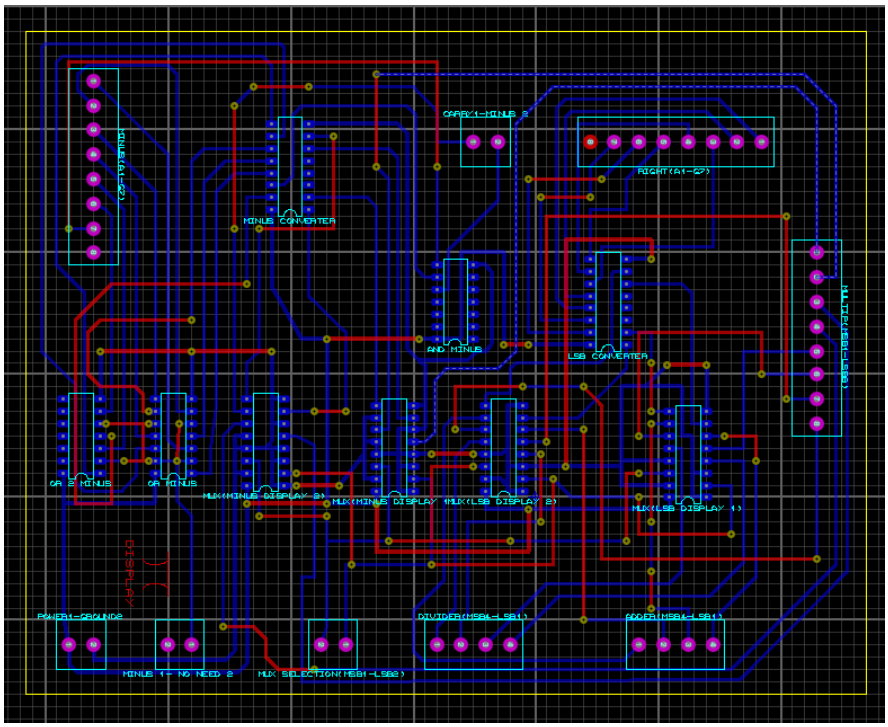
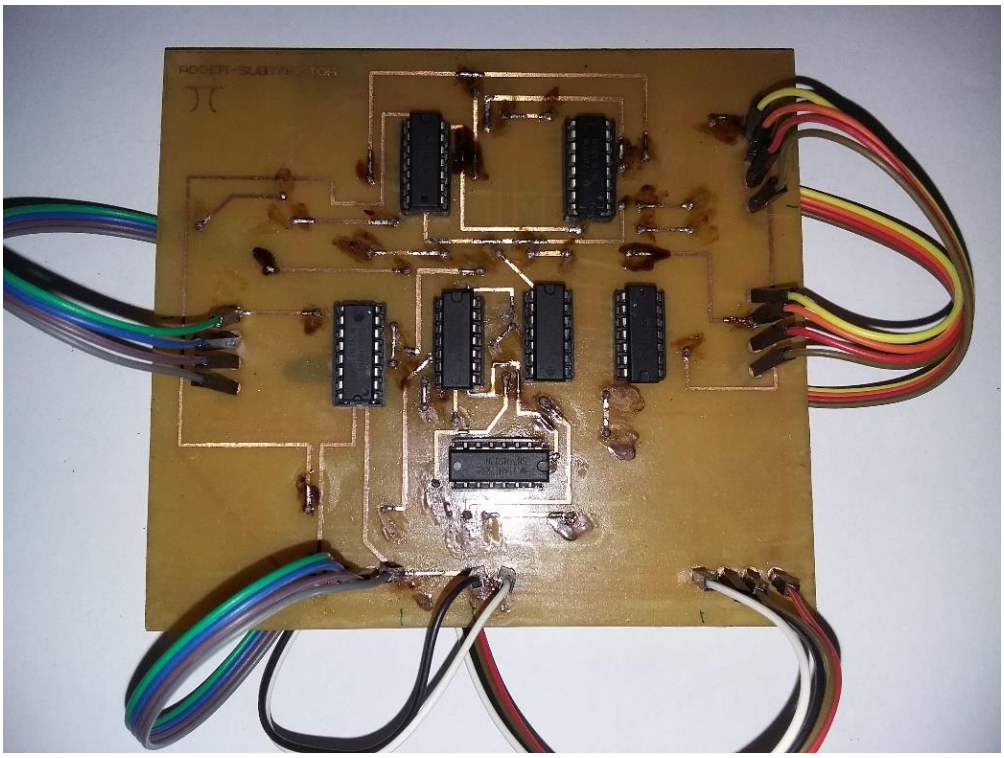


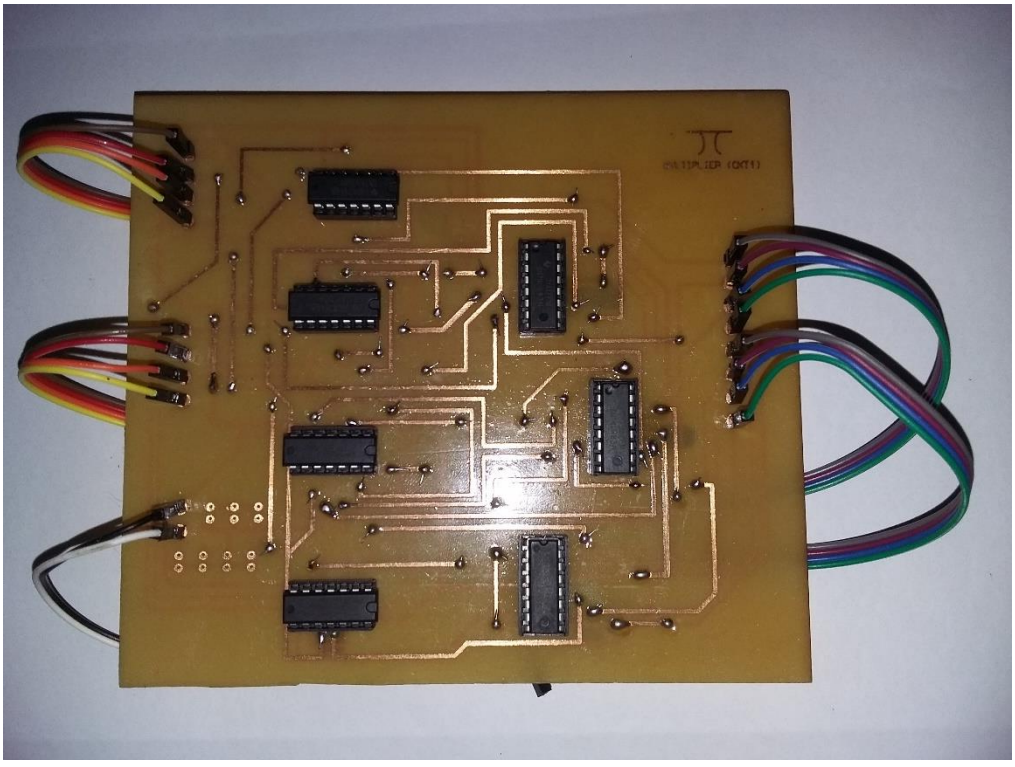
Fig.24: PCB Layout of Output Panel

Hardware Implementation:

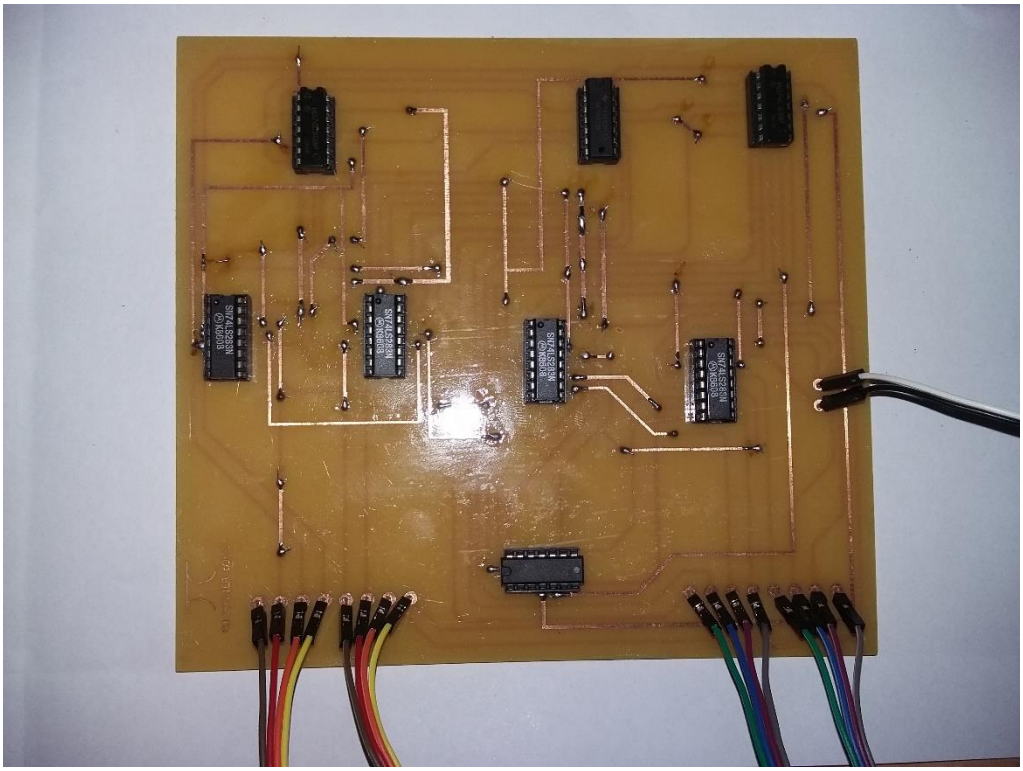
- ***Adder-Subtractor Unit:***



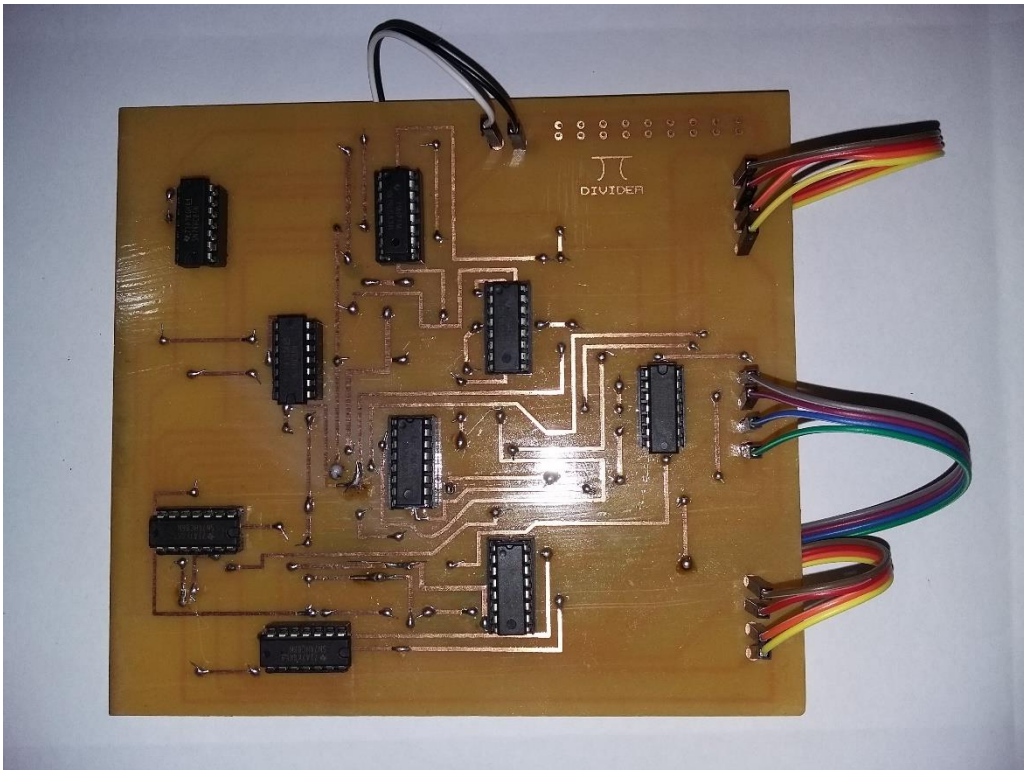
- ***Multiplier Unit:***



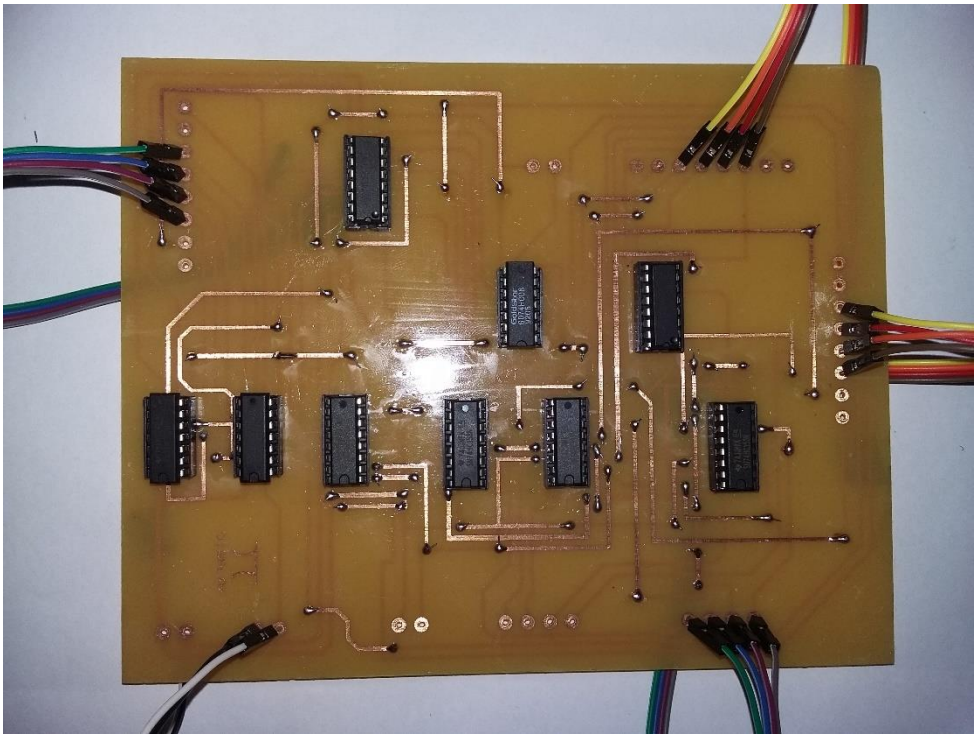
- ***7-bit Binary to BCD Converter:***



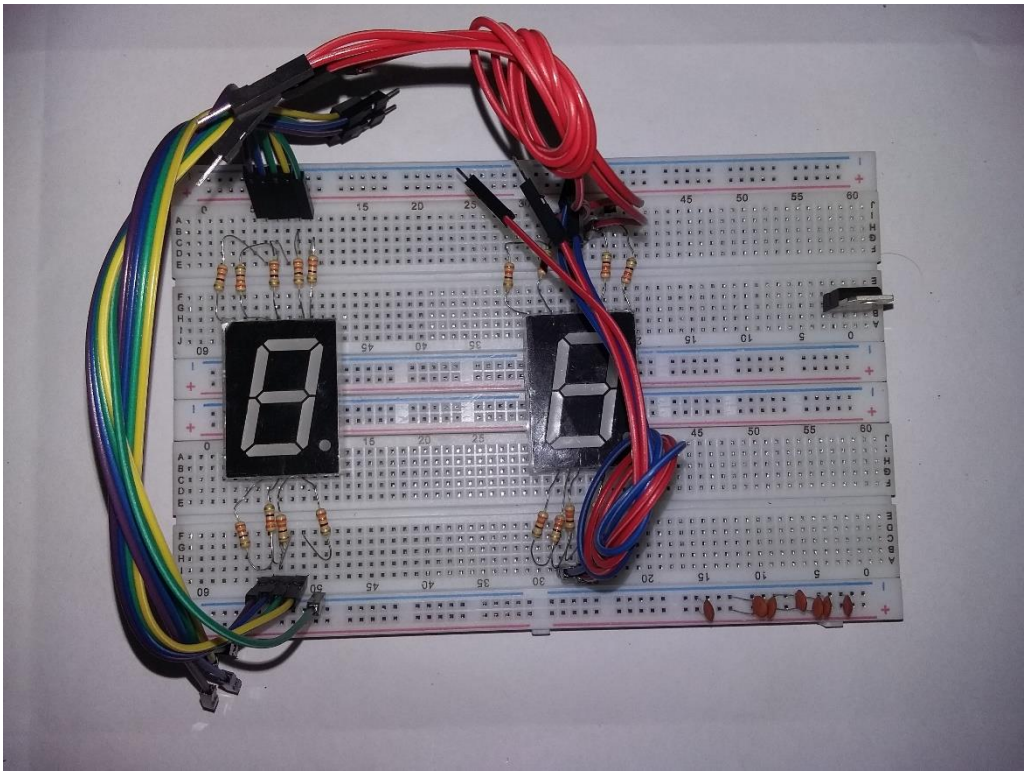
- ***Divider Unit:***



- ***Output Panel:***



- ***Display:***



Problem Faced:

- The first problem we faced is selecting proper IC. Our IC's are mostly of Quadruple 2 input. We were in need of 3-input ICs. As we couldn't find this IC, we had to cascade two 2-input gate to construct one 3-input gate. This results more IC's than required.
- Some IC's were unavailable in Bangladesh. Like 6-bit Binary to BCD converter IC 74185 or 8-bit Binary to BCD or 4×1 multiplexer etc. So we had to redesign some units like Binary to BCD converter in Multiplier Unit.
- As most of the IC's were Quadruple we had to design such a way that 4 gates are placed most efficiently. Like if 3 gates of a IC is used in left most part of the circuit, the fourth one should be as close to it as possible.
- According to our design Displays should be placed on Output Unit. But available IC package for display was 0.56 inch and our 7-segment display's size was 1.5 inch by 1 inch . So we had to place our displays separately.

Cost:

Component	***IC Code***	***Total Pins***	***Price per Unit***	***Quantity***	***Cost***
4-bit Binary Adder	74LS83	16	45	13	585
Quadruple 2-input AND	74HC08	14	15	8	120
Quadruple 2-input OR	74HC32	14	15	5	75
Quadruple 2-input XOR	74HC86	14	25	6	150
Hex Inverter	74HC04	14	15	2	30
BCD to 7-seg Decoder	74LS47	16	45	2	90
8×2 Multiplexer	74HC153	16	30	4	120
LED			1	10	10
7-segment Display	Common Anode	10	15	4	60
Battery(9V)	Rechargeable		400	1	400
	Non-rechargeable		25	2	50
Multimeter			360	1	360
Soldering Iron & Leads			270	1	270

Pumper & Rozon			110	1	110
Resistor	330 ohm			100	15
Capasitor	110 uF		1	40	40
Wire Cutter				1	150
Breadboard			70	4	280
IC Base		14	5	21	105
		16	10	19	190
5V Voltage Regulator	7805	3	25	5	125
PCB			400+780+1000		2180
Jumper	M to M		40	3	120
	M to F		40	2	80
	F to F		40	2	80
Total					5795

Conclusions:

- ***Outcomes:***

The outcome of this project is not cent percent satisfactory. Some PCB's are not working as they should. We couldn't complete the Input Panel. There will be two BCD input. But here input should be in Binary. These two inputs should be connected to each arithmetic unit. Unfortunately, we have different inputs for different arithmetic units. We couldn't introduce Keypad Input Process. Besides we are dealing with IC's where input voltage should be 5V. As we had 9V battery, so we had to use 5V voltage converter.

- ***Limitations:***

This project has some limitations. We can only do 1-bit arithmetic operation. Highest possible operation in Adder is A=9,B=9 then S=18; in Subtraction is A=9,B=0 then S=9 & A=0,B=9 then S=-9; in Multiplication is A=9,B=9 then S=81 and in Division is A=9,B=1 then Q=9 & A=9,B=2 then Q=4. In Division we could only showed Quotient result whether Remainder result couldn't be shown.

- ***Future Development:***

There are a lot of scopes to develop this project. It can be modified such a way that it can do 2-bit or 3-bit Decimal input operations. We can introduce memory unit here i.e. Input value will be shown on display until next input is given. We can introduce Single display where all bits of results can be shown.