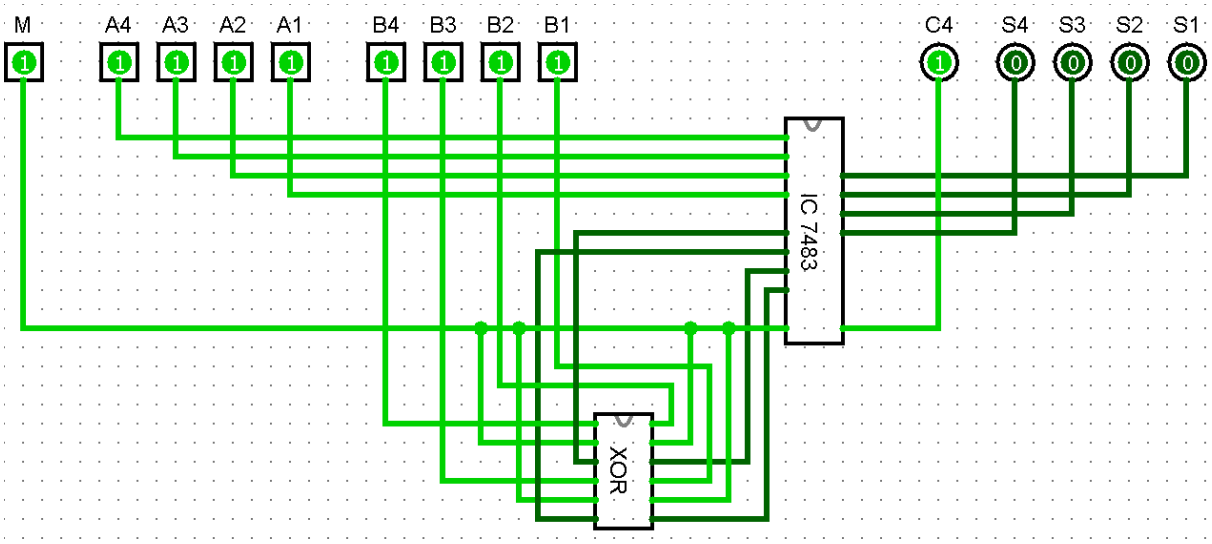
Md. Ishtiaq Ahamed Fahim
2012518642
Lab-5

C Binary Adder-Subtractor:

Logisum:

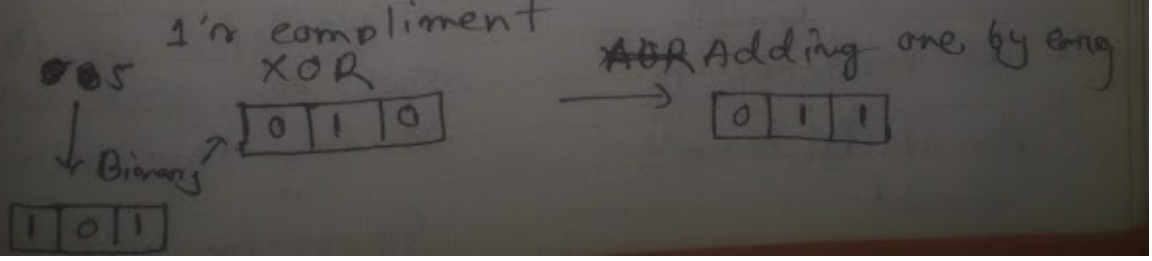


Proof and Discussion:

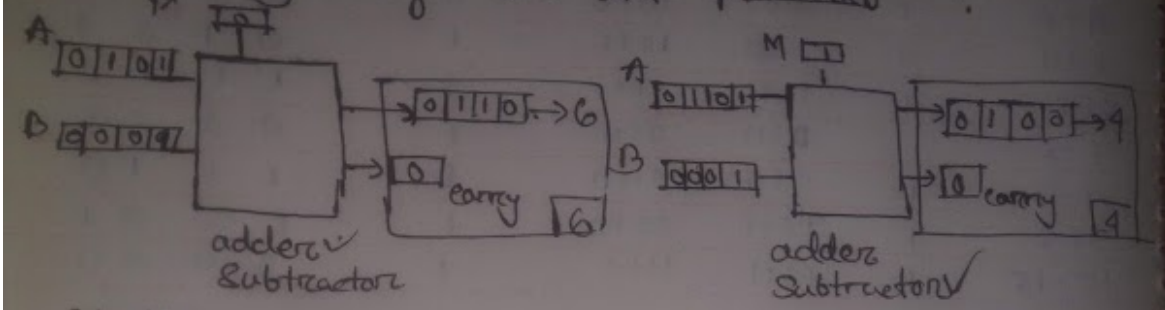
		8	4	2	1				
		0	0	0	0				
peration	M	A		B		C4	S_4	S_3	S_2 S_1
7+5	0	0111		0101		0	1	1	0 0
9+6	0	0100		0110		0	1	0	1 0
9+11	0	1001		1011		1	0	1	0 0
15+15	0	1111		1111		1	1	1	1 0
7-5	1	0111		0101		1	0	0	1 0
4-6	1	0100		0110		0	1	1	1 0
11-2	1	1011		0010		1	1	0	0 1
15-15	1	1111		1111		1	0	0	0 0

M is Responible for creating ~~cost~~ 1's compliment and cin which also input by M we get added one carry as input which makes 1's compliment to 2's.

So at end M is Responsible to creating 2's complement just by plugging 1 as M



Let's see a diagram And we add every thing in last diagram.



Nothing Interesting Happens beause B doesn't change

Beause

B	M	B⊕M	
0	0	0	✓ M=0
0	1	1	
1	0	1	✓ M=0
1	1	0	

on M=0 There is No change in B⊕M ✓

It reflects B As if M doesn't exist

Something Interesting Happens beause B changes

Because

B	M	B⊕M	
0	0	0	
0	1	1	✓ M=1
1	0	1	
1	1	0	✓ M=1

on M=1 There is change in B⊕M ✓

It Reflects on B and A gets added on By the following Step in (3) pTo

KETO-A VET

Date / /

Title Page

(3) OK lets think now

tell you want to add 1 to a neratro

You have 5 pen

your friend have 5 pen

and you want to add them but

first you want to add another pen

So Answer is

You → A → 5 → +1 → 11 pen

friend → B → 5 →

It doesn't matter who you add

1 with

You → A → 6 → 11

friend → B → 5

You → A → 5 → 11

friend → B → 6

Both are Same

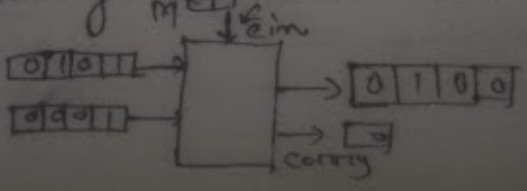
CRAZON VET

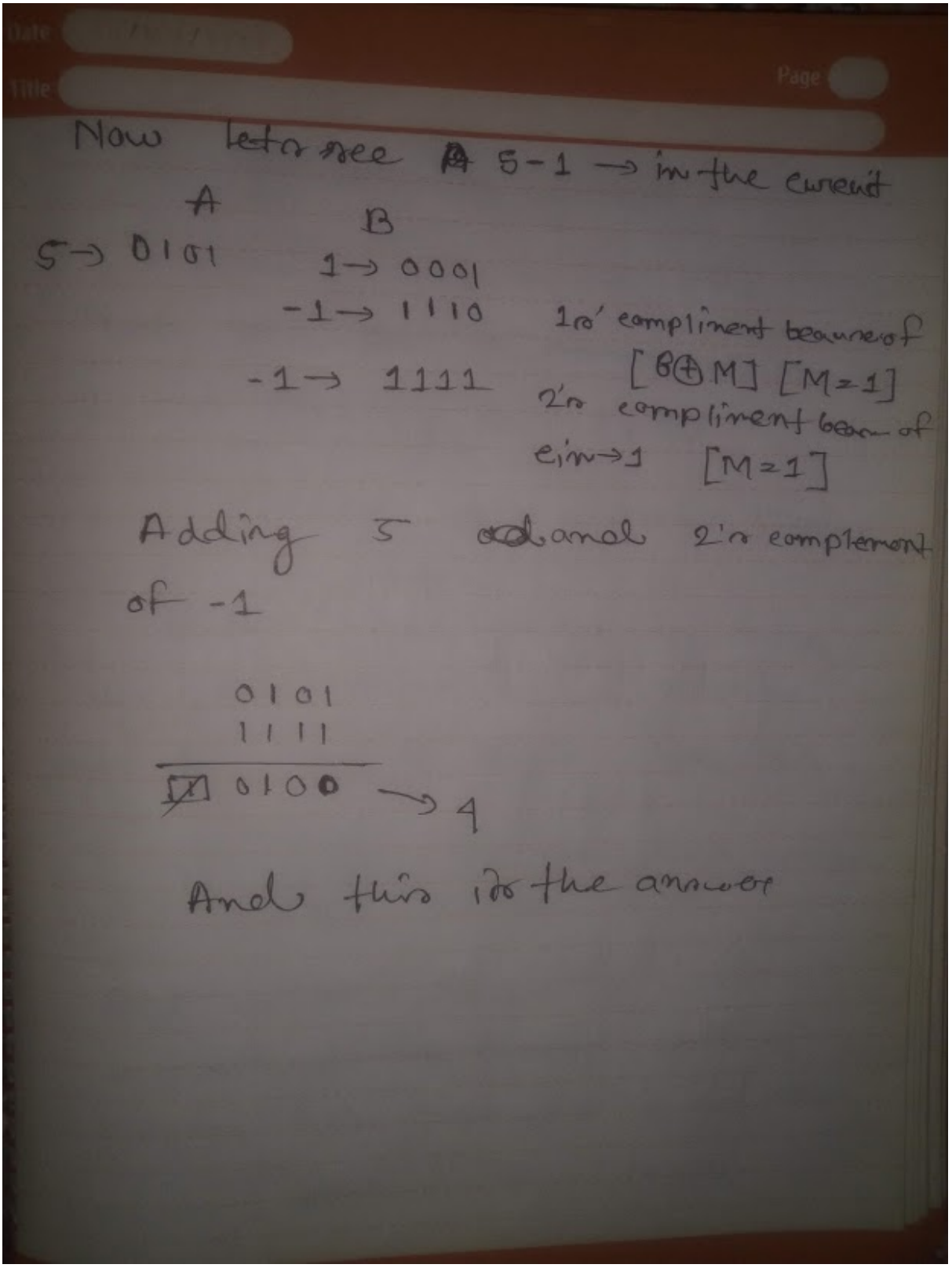
So for Subtracting opperation we put 1 in M and This Addn 1 in both of its addition but why ?

because XOR creates 1s complement of a number if M is (1)

B ⊕ M	B' 1s compliment
0 ⊕ 1 →	1
1 ⊕ 1 →	0

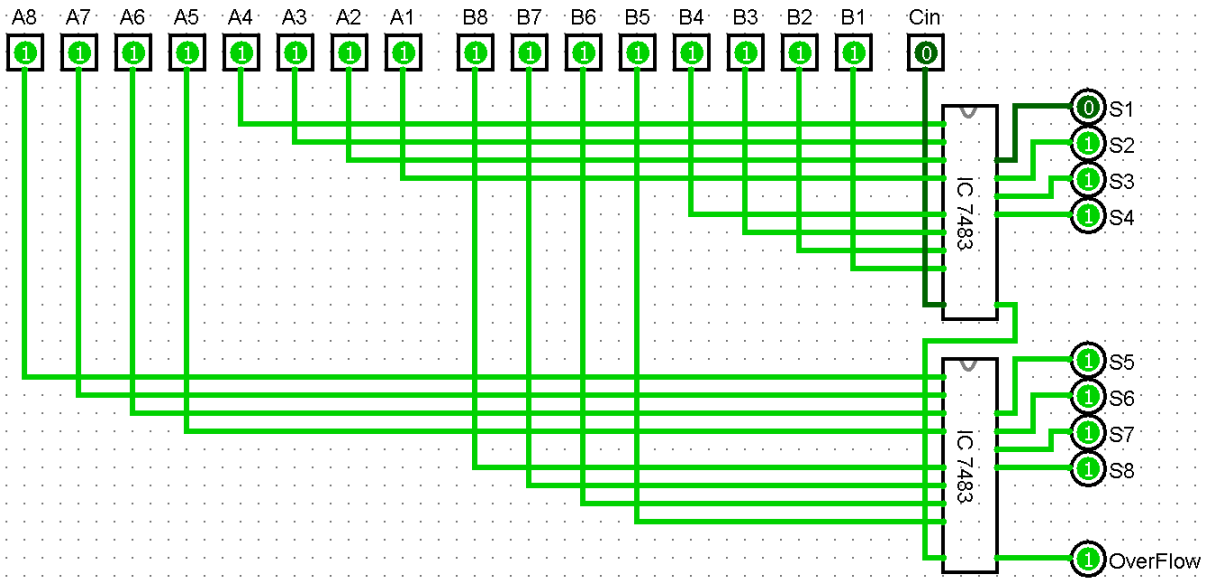
But To add a negative number we need to add 1 in ones compliment which is done by putting 1 in cin



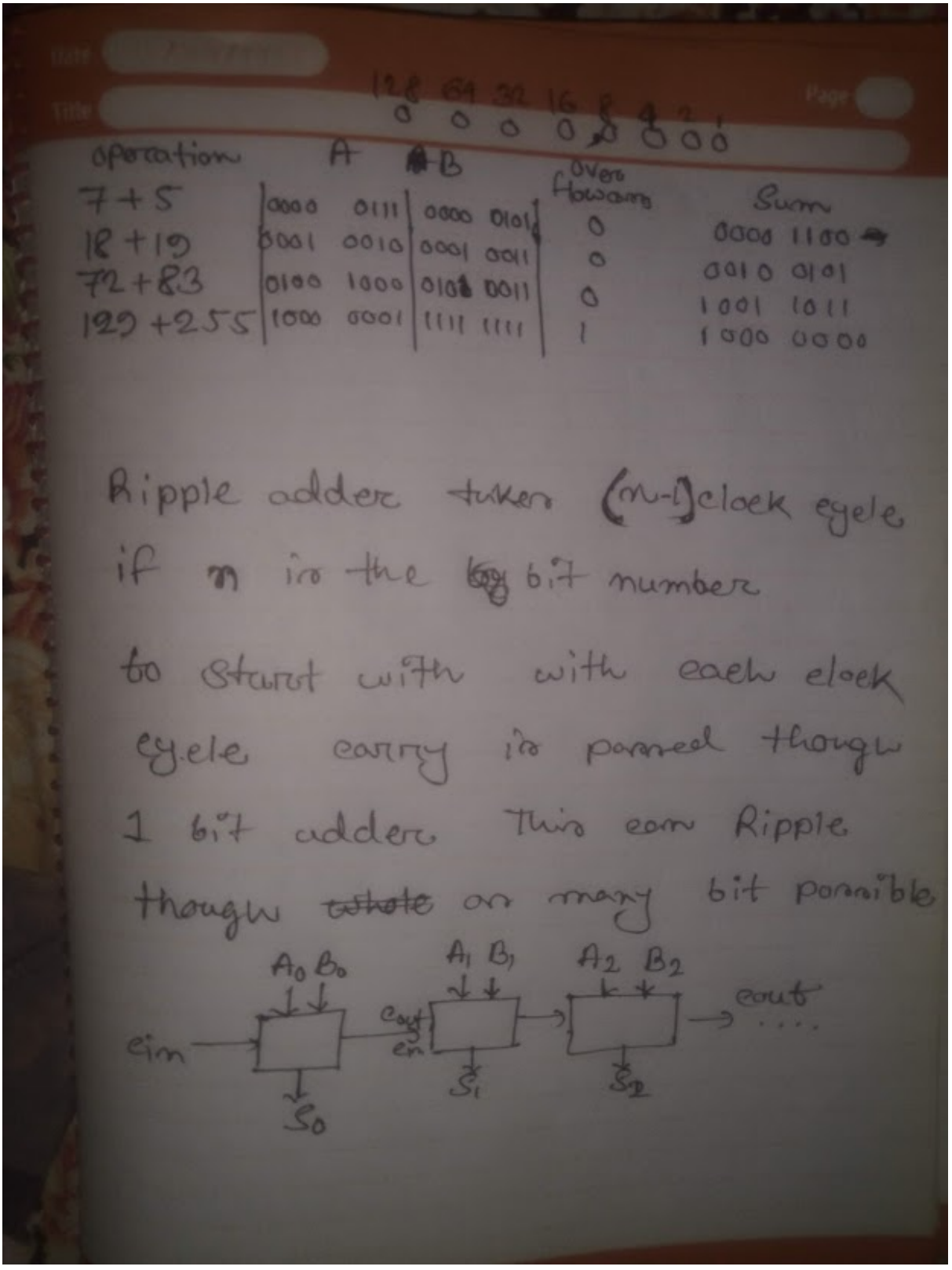


D 8-Bit-Ripple-Through-Carry Adder:

Logisum:

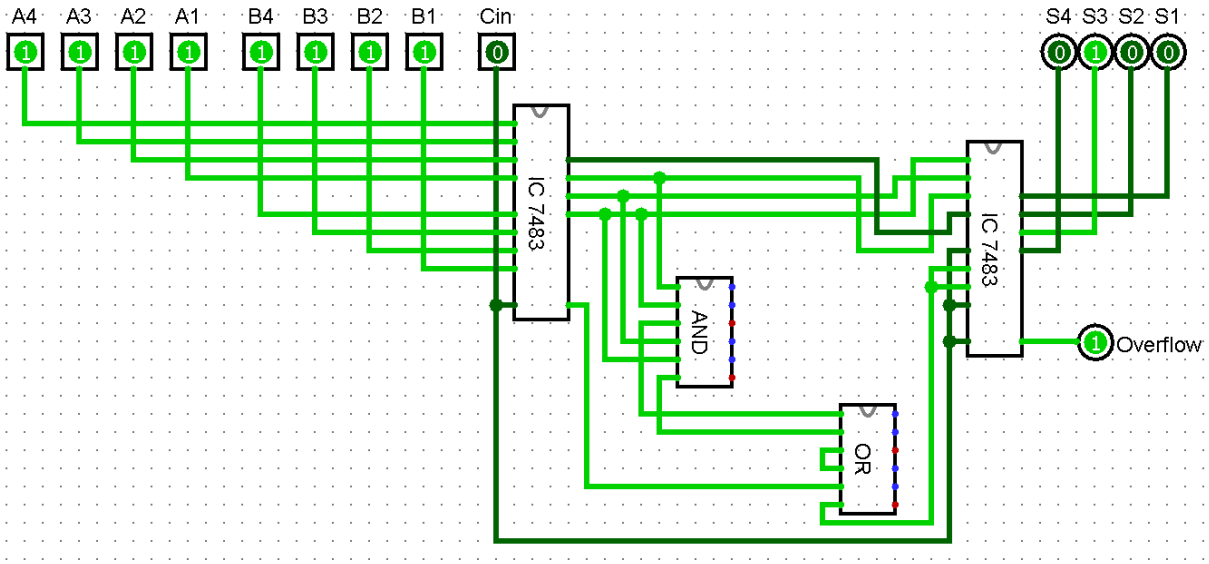


Proof and Discussion:



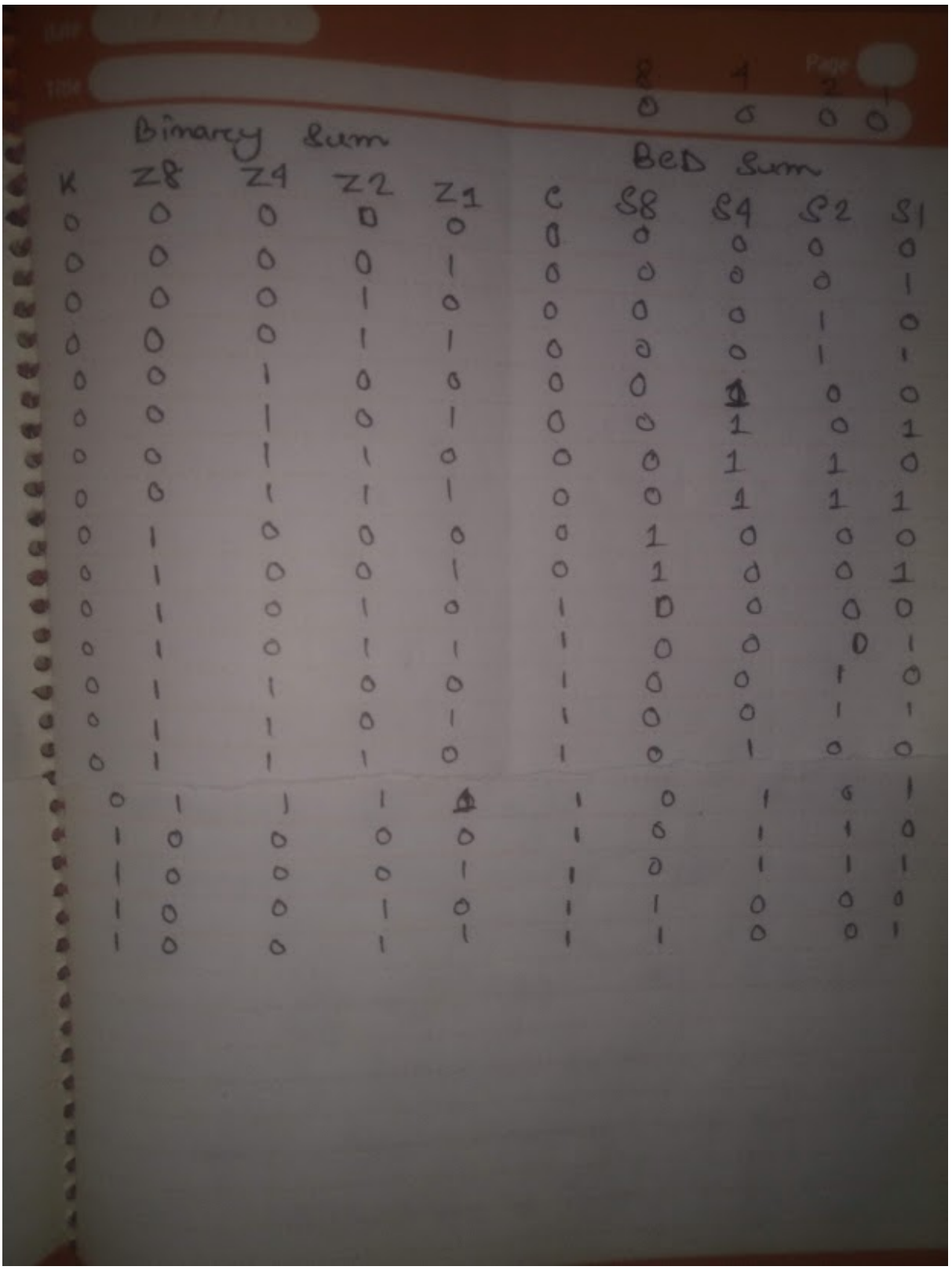
E Binary BCD sum:

Logisum:



E1:

Proof and Discussion:



E2:

Proof and Discussion:

	8	4	2	1	
	0	0	0	0	

operation	A	B	overflow carry	Sum
9+0	1001	0000	0	1001
9+1	1001	0001	1	0000
9+2	1001	0010	1	0001
9+3	1001	0011	1	0010
9+4	1001	0100	1	0011
9+5	1001	0101	1	0100
9+6	1001	0110	1	0101
9+7	1001	0111	1	0110
9+8	1001	1000	1	0111
9+9	1001	1001	1	1000

To understand How BCD works

lets play agame.

convert 80 minute in hh:mm format ; the naive approach would be

$$\frac{80}{60} = 1.333\dot{3}$$

1 : □□

→ beause 60 min in 1h

take the floating unit only

$.333\dot{3} \times 60 = 20$ → 1:20

$$\begin{array}{r} 80 \\ +\ 40 \\ \hline 120 \end{array}$$

Now lets put a ~~semicolon~~ colon after 1 → Making it 1:20

Then the answer right

BCD (Adder) also works in same way. Are think of a number dial for one digit like this

It can have number from 0-9

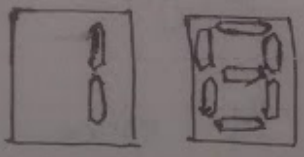
But to represent 9 we need 4 bit in binary number which has max of 15 counting 6 bit.

So using old rule

leta represent 12

$$\begin{array}{r} 12 \\ +\ 6 \\ \hline 18 \end{array}$$

and leta put a operator are ita impoible to put 2 number in one digit



To contact the corect for 6 step up after 9 we don't require S_1 in the input So we take S_2 S_3 S_4 and carry: