```
1. -62981
                                      Xinshi Wang
   1. take absolute value
                                       661975305
     62981
   2. Substract 1
                           7 Proce dure
       62980
                              62990/2=3149000
   3. Binary representation
                              31490/2 = 15745
                            15745/2 = 7872 ...
       0111101100000010
                             787212 = 3936 - 0
   4. Flip all the bits
                              3936/2=1968-0
       1000 0100 1111 1101
                              1968/7 = 984 ...0
2. We can represent 34,90623 as 1117/32,
   which is equivalent
                  60
            0100 0101 1101/25
                                                  Prtra 0
           = 0100 0101 1101 17 5
             = 0100 010.1 1101 ·20
             - 1.000 IIII ol · 25
           Fraction: 0000 0000 0000 0111 101
           Exponent: 5 +127=132=1000 0100 -> Procedure
           Sign !
                                                132/2=66 ~~
                                                66/2=33-10
                   0 F B 0
            0
      1 . 0 0 15 11 0 1
                                                16/z=8 ...p
 8+4+2+1
        0 0 814+2+1 812+1 0
        1000 0000 1101 1011 0000 0000
                                                 2/2=1--0
```

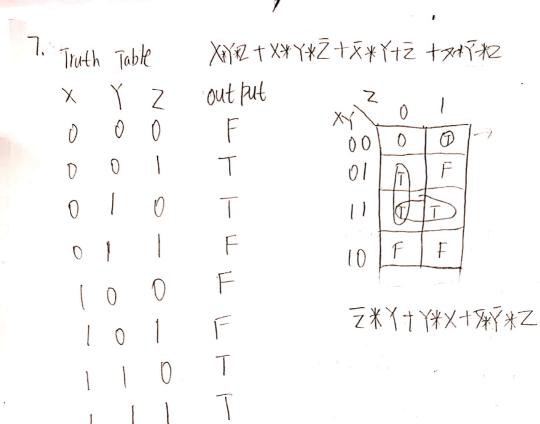
1/2=000

4. CAB005 E5

- 5 x 16+ 14 x 16+ 5 x 162+0+0+ 11 x 165+ 10x 16+ 12 x 167
- = 5+224+1280+11534336+167772160+3721225 472
 - = 3400 533477
- 5. We could use the same circuit for add and substration in this case, for example, We can calculate 5+3 using an addler, Similarly, we can use the same add gate to calculate 5-3 by taking the 2's complement of 3, which gives us -3, Thus we have 5+(-3)=5-3, Therefore it is efficient.

6. A*(ATB)+(B+AKA)*(A+B)

- = A*(A+B)+(B+A)*(A+B) > idempotent Law
- = AXA + AXB+ LBTA) X (A+B) -> Distribution
- = 0 T A*B+(13+A)*(A+B) -> Complement Law
- = A*B+(BtA)*(A+B) -) Identity Law
- = AXB+ (ATB)XB+(ATB)XA -> Distribution
- = AXB+BXA+BXB+CATB)XA -> Distribution
- = AXB TBXA + O+CA+B)XA > Complement Law
- = AXB + BXA + (A + B)XA -> Identity Law
- = B(A+A) + LA+B)*A-> Distributive Law
- = BXI+ CA+B)*A -> complement Law
 - = B+(A+B)*A>IdeMtHLaw
 - B+ AXA+ AX B > distributive
 - = B+A+AB -> idempotent = B+A -> absorption Law



8. ARBACTARBACTABACTABAC

