NISHITH SHAH (SJSU ID:011478948) **DATE:28TH SEP'16**

5 – STAGE UNPIPELINED (IEEE 754) FLOATING POINT ADDER

CODE:

```
#include<stdio.h>
#include<conio.h>
void main()
 {
        int A_sign_48[6]={ 0 , 0 , 1 , 1 , 0 , 0},                  //Array of Signs of A
            B_sign_48[6]={ 0 , 1 , 0 , 1 , 0 , 1},                  //Array of Signs of B
            Sum_sign_48[6],                                         //Array of Signs of Sum
            A_exp_48[6]={ 0x85 , 0x85 , 0x84 , 0x87 , 0x00 , 0x00 }, //Array of Exponents of A
            B_exp_48[6]={ 0x86 , 0x84 , 0x84 , 0x85 , 0x00 , 0x86 }, //Array of Exponents of B
            Sum_exp_48[6],                                          //Array of Exponents of Sum
            shift,                                                  //Shifts required to align Mantissa
            i=0;                                                    //Index of elements of array

                //Array of Mantissa's of A
unsigned long int A_mantissa_48[6]={ 0x480000,0x480000,0x540000,0x160000,0x000000,0x000000 },
                //Array of Mantissa's of B
                B_mantissa_48[6]={ 0x480000,0x480000,0x0c0000,0x460000,0x000000,0x2b0000 },
                //Array of Mantissa's of Sum
                Sum_mantissa_48[6],
                //Masks for Normalization
                mask1=0x400000,
                mask2=0x7fffff,
                //Array of Normalized Sum
                x[6];

        for(i=0;i<6;i++)                    //Loop to access each element of the array sequentially
        {
                clrscr();
                printf("Testbench %d\n",i+1);
                printf("A is %d %x %lx\n",A_sign_48[i],A_exp_48[i],A_mantissa_48[i]);
                printf("B is %d %x %lx\n",B_sign_48[i],B_exp_48[i],B_mantissa_48[i]);

                printf("\n----------STAGE - 1----------");

                //Loop to determine the shift and the greater number

                if(A_exp_48[i]>B_exp_48[i])
                {
                        shift=A_exp_48[i]-B_exp_48[i];
                        printf("\nA is greater\n");
                }
                else if(A_exp_48[i]<B_exp_48[i])
                {
                        shift=B_exp_48[i]-A_exp_48[i];
                        printf("\nB is greater\n");
                }
```

```
else
{
        shift=0;
}

//Including the 1 from "1.m" into the Mantissa
A_mantissa_48[i]=A_mantissa_48[i]>>1;
B_mantissa_48[i]=B_mantissa_48[i]>>1;
A_mantissa_48[i]=A_mantissa_48[i]|mask1;
B_mantissa_48[i]=B_mantissa_48[i]|mask1;
A_exp_48[i]+=1;
B_exp_48[i]+=1;
printf("\nThe extra shifted no's are \n%d %x %lx\n%d %x %lx", A_sign_48[i], A_exp_48[i],
         A_mantissa_48[i], B_sign_48[i], B_exp_48[i], B_mantissa_48[i]);
printf("\nThe no. of shifts = %d\n",shift);

printf("\n----------STAGE - 2----------");

//Aligning the Mantissa based on the number of shifts required and making the exponents
  same
if(A_exp_48[i]<B_exp_48[i])
{
        A_mantissa_48[i]=A_mantissa_48[i]>>shift;
        A_exp_48[i]=A_exp_48[i]+shift;
        Sum_sign_48[i]=B_sign_48[i];
}
else if(A_exp_48[i]>B_exp_48[i])
{
        B_mantissa_48[i]=B_mantissa_48[i]>>shift;
        B_exp_48[i]=B_exp_48[i]+shift;
        Sum_sign_48[i]=A_sign_48[i];
}
else
{
        A_mantissa_48[i]=A_mantissa_48[i];
        B_mantissa_48[i]=B_mantissa_48[i];
        Sum_sign_48[i]=A_sign_48[i];
}
Sum_exp_48[i]=A_exp_48[i];
printf("\nThe alligned no. A is%d %x %lx\n",A_sign_48[i],A_exp_48[i],A_mantissa_48[i]);
printf("\nThe alligned mantissa of B is %d %x %lx\n", B_sign_48[i], B_exp_48[i],
         B_mantissa_48[i]);
```

printf("\n----------STAGE - 3 ----------");

```
//Determining the smaller mantissa and taking it's 2's complement
if(A_sign_48[i]!=B_sign_48[i])
{
        if(A_mantissa_48[i]<B_mantissa_48[i])
        {
                A_mantissa_48[i]=(~A_mantissa_48[i])+1;
        }
        else
        {
                B_mantissa_48[i]=(~B_mantissa_48[i])+1;
        }
        printf("\nThe mantissa's after 2's complement %lx %lx\n", A_mantissa_48[i],
                 B_mantissa_48[i]);
}

else
{
        A_mantissa_48[i]=A_mantissa_48[i];
        B_mantissa_48[i]=B_mantissa_48[i];
        printf("\n2's complement is not required as signs are same\n");
}
```

printf("\n----------STAGE - 4----------");

```
//Addition of the Mantissa's
Sum_mantissa_48[i]=A_mantissa_48[i]+B_mantissa_48[i];
printf("\nThe Sum is %d %x %lx\n" , Sum_sign_48[i] ,Sum_exp_48[i],Sum_mantissa_48[i]);
```

printf("----------STAGE - 5----------");

```
//Normalization of the Mantissa's

if(A_sign_48[i]!=B_sign_48[i])                  //For Sign(A) != Sign(B)
{
        if(A_sign_48[i]==0&B_sign_48[i]==1)     //For A=+ve & B=-ve
        {
                if(A_mantissa_48[i]==0)         //For A=0
                {
                        Sum_mantissa_48[i]=Sum_mantissa_48[i]<<2;
                        //printf("%lx\n\n",Sum_mantissa_48);
                        x[i]=Sum_mantissa_48[i]&mask2;
                        x[i]=x[i]>>1;
                        Sum_exp_48[i]=Sum_exp_48[i]-1;
                        printf("\nThe Normalized Value of Sum is %d %x %lx\n" ,
                                Sum_sign_48[i], Sum_exp_48[i], x[i]);
                }
```

```
                        Else                                    //For A !=0
                        {
                                Sum_mantissa_48[i]=Sum_mantissa_48[i]<<2;
                                //printf("%lx\n\n",Sum_mantissa_48);
                                x[i]=Sum_mantissa_48[i]&mask2;
                                Sum_exp_48[i]=Sum_exp_48[i]-2;
                                printf("\nThe Normalized Value of Sum is %d %x %lx\n",
                                        Sum_sign_48[i], Sum_exp_48[i],x[i]);
                        }
                }

                Else                                    //For Sign(A) != Sign(B)

                {
                        Sum_mantissa_48[i]=Sum_mantissa_48[i]<<2;
                        //printf("%lx\n\n",Sum_mantissa_48);
                        x[i]=Sum_mantissa_48[i]&mask2;
                        Sum_exp_48[i]=Sum_exp_48[i]-2;
                        printf("\nThe Normalized Value of Sum is %d %x %lx\n", Sum_sign_48[i],
                                Sum_exp_48[i], x[i]);
                }
        }
        Else                                            //For Sign(A) = Sign(B)
        {
                if(A_sign_48[i]==0&B_sign_48[i]==0)     //Both A & B = +ve
                {
                        if(A_mantissa_48[i]==0)         //For A=0
                        {
                                x[i]=Sum_mantissa_48[i]&mask2;
                                Sum_exp_48[i]=Sum_exp_48[i]-1;
                        }
                        Else                            //For A != 0
                        {
                                x[i]=Sum_mantissa_48[i]&mask2;
                        }
                }
                else if(A_sign_48[i]==1&B_sign_48[i]==1)    //Both A & B = -ve
                {
                        Sum_mantissa_48[i]=Sum_mantissa_48[i]<<1;
                        Sum_exp_48[i]=Sum_exp_48[i]-1;
                        x[i]=Sum_mantissa_48[i]&mask2;
                }
                else{}
                printf("\nThe Normalized Value of Sum is %d %x %lx\n", Sum_sign_48[i],
                        Sum_exp_48[i],x[i]);
        }
        getch();
    }
}
```

NISHITH SHAH (SJSU ID:011478948) **DATE:28TH SEP'16**

OUTPUT:

FOR A=100 & B=200 :

```
Testbench 1
A is 0 85 480000
B is 0 86 480000

----------------------------STAGE - 1-------------------------------
B is greater

The extra shifted no's are
0 86 640000
0 87 640000
The no. of shifts = 1

----------------------------STAGE - 2-------------------------------
The alligned no. A is0 87 320000

The alligned mantissa of B is 0 87 640000

----------------------------STAGE - 3 -------------------------------
2's complement is not required as signs are same

----------------------------STAGE - 4-------------------------------
The Sum is 0 87 960000
----------------------------STAGE - 5-------------------------------
The Normalized Value of Sum is 0 87 160000
```

FOR A=100 & B=-50 :

```
Testbench 2
A is 0 85 480000
B is 1 84 480000

----------------------------STAGE - 1-------------------------------
A is greater

The extra shifted no's are
0 86 640000
1 85 640000
The no. of shifts = 1

----------------------------STAGE - 2-------------------------------
The alligned no. A is0 86 640000

The alligned mantissa of B is 1 86 320000

----------------------------STAGE - 3 -------------------------------
The mantissa's after 2's complement 640000 ffce0000

----------------------------STAGE - 4-------------------------------
The Sum is 0 86 320000
----------------------------STAGE - 5-------------------------------
The Normalized Value of Sum is 0 84 480000
```

NISHITH SHAH (SJSU ID:011478948) **DATE:28TH SEP'16**

FOR A=-53 & B=35 :

```
Testbench 3
A is 1 84 540000
B is 0 84 c0000

----------------------------STAGE - 1-------------------------------
The extra shifted no's are
1 85 6a0000
0 85 460000
The no. of shifts = 0

----------------------------STAGE - 2-------------------------------
The alligned no. A is1 85 6a0000

The alligned mantissa of B is 0 85 460000

----------------------------STAGE - 3 -------------------------------
The mantissa's after 2's complement 6a0000 ffba0000

----------------------------STAGE - 4-------------------------------
The Sum is 1 85 240000
----------------------------STAGE - 5-------------------------------
The Normalized Value of Sum is 1 83 100000
```

FOR A=-300 & B=-99 :

```
Testbench 4
A is 1 87 160000
B is 1 85 460000

----------------------------STAGE - 1-------------------------------
A is greater

The extra shifted no's are
1 88 4b0000
1 86 630000
The no. of shifts = 2

----------------------------STAGE - 2-------------------------------
The alligned no. A is1 88 4b0000

The alligned mantissa of B is 1 88 18c000

----------------------------STAGE - 3 -------------------------------
2's complement is not required as signs are same

----------------------------STAGE - 4-------------------------------
The Sum is 1 88 63c000
----------------------------STAGE - 5-------------------------------
The Normalized Value of Sum is 1 87 478000
```

NISHITH SHAH (SJSU ID:011478948) **DATE:28TH SEP'16**

FOR A=0 & B=0 :

```
Testbench 5
A is 0 0 0
B is 0 0 0

----------------------------STAGE - 1-------------------------------
The extra shifted no's are
0 1 400000
0 1 400000
The no. of shifts = 0

----------------------------STAGE - 2-------------------------------
The alligned no. A is0 1 400000

The alligned mantissa of B is 0 1 400000

----------------------------STAGE - 3 -------------------------------
2's complement is not required as signs are same

----------------------------STAGE - 4-------------------------------
The Sum is 0 1 800000
----------------------------STAGE - 5-------------------------------
The Normalized Value of Sum is 0 1 0
```

FOR A=0 & B=-171 :

```
Testbench 6
A is 0 0 0
B is 1 86 2b0000

----------------------------STAGE - 1-------------------------------
B is greater

The extra shifted no's are
0 1 400000
1 87 558000
The no. of shifts = 134

----------------------------STAGE - 2-------------------------------
The alligned no. A is0 87 0

The alligned mantissa of B is 1 87 558000

----------------------------STAGE - 3 -------------------------------
The mantissa's after 2's complement 0 558000

----------------------------STAGE - 4-------------------------------
The Sum is 1 87 558000
----------------------------STAGE - 5-------------------------------
The Normalized Value of Sum is 1 86 2b0000
```