Don Bosco Institute of Technology,Kurla

EXPERIMENT NO:1

Title: A program to simulate the working of 4bit parallel adder (7483)

Name: **Hayden Cordeiro** Class: **SE Comps** Roll No: **05**

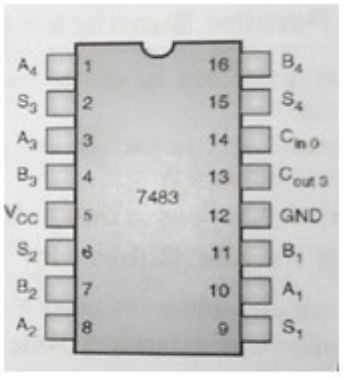
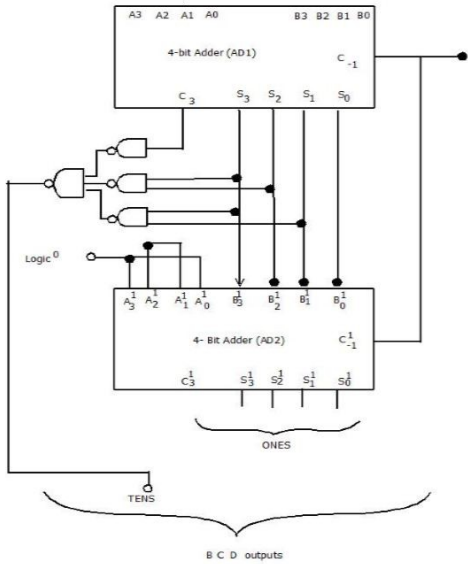
Class:S.E **Lecturer:Sejal.Ch**

Subject:PA

EXPERIMENT NO:1

Simulate the working of 4bit parallel adder

AIM	Write a program to simulate the working of 4bit parallel adder (7483)
LEARNING OBJECTIVE	To implement the operation of the arithmetic unit including the implementation of fixedpoint addition.
LEARNING OUTCOME	Students will be able to implement the simulation of adder circuit in higher level language .
LAB OUTCOME	CSL 403.1: Ability to compile a code for computer operations.
PROGRAM OUTCOME	PO11, PO52, PO83, PO93, PO122, PSO12
BLOOM'S TAXONOMY LEVEL	Remember, Understand

THEORY	 Fig1. Pin diagram of IC 7483 IC 7483 is a 4 bit parallel adder which consists of four interconnected full adders along with the look ahead carry circuit. The inputs to the IC are A, B and Cin while outputs are S and C_{out}. $A_3A_2A_1A_0$ is a 4 bit input word 'A' and $B_3B_2B_1B_0$ is the second 4 bit input word 'B'. C_{in} is the input carry. The IC adds the two four bit words along with input carry to produce a 4 bit sum and a one bit carry-out. C_{out} represents the output carry. S_3,S_2,S_1, S_0 represents sum output with S3 as the MSB. The 4-bit binary adder IC (7483) can be used to perform addition of BCD numbers. In this, if 4- bit sum output is not a valid BCD digit, or if carry C_3 is generated, then decimal 6 (0 1 1 0) is to be added to the sum to get the correct result.  Fig (a) 4- Bit B C D Adder
SOFTWARE USED	C/C++/Java/Python
STEPS TO EXECUTE THE PROGRAM	1.Take two 4-bit binary inputs and add them. 2.Consider the incoming carry to perform addition. 3.Reflect the output on sum pins and also the outgoing carry output.
CODE	**Part A:** #include<stdio.h> void dec_bin(int num, int a[], int size) {

```
	int i;
	for(i=size-1;i>=0;i--)
	{
		a[i]=num%2;
		num=num/2;
	}
}
void main()
{
	int num1=0,num2=0,cin=0,cout=0,intcarry=0;
	int a[4],b[4],s[4];
	printf("Enter number 1(0-15):");
	scanf("%d",&num1);
	printf("Enter number 2(0-15):");
	scanf("%d",&num2);
	printf("Enter Cin(0 or 1) :");
	scanf("%d",&cin);
	dec_bin(num1,a,4);
	dec_bin(num2,b,4);
int j=0;

	s[3]=a[3]+b[3]+cin;
	if(s[3]==2)
	{
		s[3]=0;
		intcarry=1;
	}
	else if(s[3]==3)
	{
		s[3]=1;
		intcarry=1;
	}
	else
		intcarry=0;
for(j=2;j>=0;j--)
{
	s[j]=a[j]+b[j]+intcarry;
	if(s[j]==2)
	{
		s[j]=0;
		intcarry=1;
	}
	else if(s[j]==3)
	{
		s[j]=1;
		intcarry=1;
	}
	else
		intcarry=0;
}
printf("Sum : ");
for(int k=0;k<4;k++)
```

```
printf("%d",s[k]);
printf("\nCout : %d",intcarry);

}
```

```
PS D:\COA\Exp 1> gcc .\Grejo_coa_exp_1.c
PS D:\COA\Exp 1> ./a.exe
Enter number 1(0-15):9
Enter number 2(0-15):9
Enter Cin(0 or 1) :0
Sum : 0010
Cout : 1
PS D:\COA\Exp 1>
```

```
PS D:\COA\Exp 1> ./a.exe
Enter number 1(0-15):15
Enter number 2(0-15):15
Enter Cin(0 or 1) :0
Sum : 1110
Cout : 1
PS D:\COA\Exp 1>
```

Part B:

```
#include<stdio.h>

void dec_bin(int num, int a[], int size)
{
        int i;
        for(i=size-1;i>=0;i--)
        {
                a[i]=num%2;
                num=num/2;
        }
}
void main()
{
        int num1=0,num2=0,cin=0,cout=0,intcarry=0;
        int a[4],b[4],s[4];
        printf("Enter number 1(0-9):");
        scanf("%d",&num1);
        printf("Enter number 2(0-9):");
        scanf("%d",&num2);
        printf("Enter Cin(0 or 1) :");
        scanf("%d",&cin);
        dec_bin(num1,a,4);
        dec_bin(num2,b,4);
        int j=0;

        s[3]=a[3]+b[3]+cin;
        if(s[3]==2)
        {
                s[3]=0;
```

```
                intcarry=1;
        }
        else if(s[3]==3)
        {
                s[3]=1;
                intcarry=1;
        }
        else
                intcarry=0;
for(j=2;j>=0;j--)
{
        s[j]=a[j]+b[j]+intcarry;
        if(s[j]==2)
        {
                s[j]=0;
                intcarry=1;
        }
        else if(s[j]==3)
        {
                s[j]=1;
                intcarry=1;
        }
        else
                intcarry=0;
}

printf("Sum : ");
int k;

int bcd[4],bcdcarry=0;
dec_bin(6,bcd,4);
int nsum[4];
//BCD Correction
if((num1+num2)>9)
{
intcarry=1;
for(j=3;j>=0;j--)
{
        nsum[j]=s[j]+bcd[j]+bcdcarry;
        if(nsum[j]==2)
        {
                nsum[j]=0;
                bcdcarry=1;
        }
        else if(nsum[j]==3)
        {
                nsum[j]=1;
                bcdcarry=1;
        }
        else
                bcdcarry=0;
}
```

```
for(k=0;k<4;k++)
printf("%d",nsum[k]);
printf("\nCout : %d",intcarry);
}
else
{
for(k=0;k<4;k++)
printf("%d",s[k]);
printf("\nCout : %d",intcarry);
}
}
```

```
Enter number 1(0-9):9
Enter number 2(0-9):9
Enter Cin(0 or 1) :0
Sum : 1000
Cout : 1
Process returned 9 (0x9)   execution time : 8.434 s
Press any key to continue.
```

```
Enter number 1(0-9):5
Enter number 2(0-9):5
Enter Cin(0 or 1) :1
Sum : 0001
Cout : 1
Process returned 9 (0x9)   execution time : 3.071 s
Press any key to continue.
```

CONCLUSION	Hence, we implemented the working of a 4-bit parallel adder . The program simulated the working of IC 7483. The inputs were taken from the user and the addition sum and carry generated was displayed as output. The additional scope of the experiment which was to implement BCD Correction in the output was also implemented.
REFERENCES	1. William Stallings, “Computer Organization and Architecture: Designing for Performance”,Pearson Publication, 10 th Edition, 2013 2. B. Govindarajulu, “Computer Architecture and Organization: Design Principles and Applications”, Second Edition, McGrawHill (India)