Ripple Carry Adder

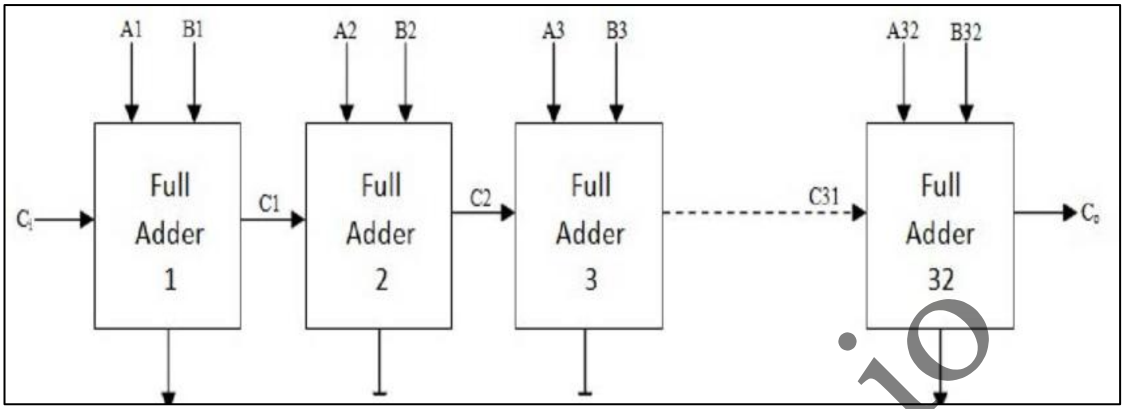


Fig. 1: 32 Bit Ripple Carry Adder

A Ripple Carry Adder is made of Full Adder Circuits connected in series where the subsequent adder is connected to the previous full adder and the carryout of the previous adder becomes the carryin of the next full adder. The Schematic of a 32-Bit Ripple Carry Adder is shown in figure 1. The worst-case delay of a Ripple Carry Adder circuit is **n*2D** where **n** is the number of bits and D is the delay of one logic gate level. In Fig. 2 we can observe the first logic gate delay occurs at **Sum = A ^ B ^ Cin** and **(A & B), (B & Cin), (Cin & A)**. The Second Logic Gate delay occurs at **Cout = (A & B) | (B & Cin) | (Cin & A)**, Hence the Full adder has a delay of 2D.

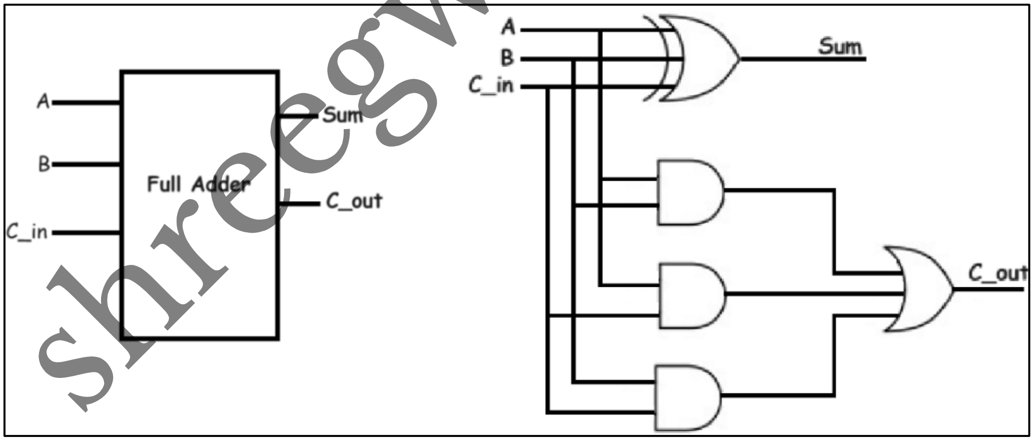


Fig. 2: Full Adder Schematic

1. Approach for Ripple Carry Adder

For this assignment, the Ripple Carry Adder is coded using C++. There are a total of 3 functions used for the program.

1. array<unsigned long int, 32> bcdConv (unsigned long int A)
   This function converts decimal number to binary number. The datatype used here is unsigned long int which has a range of 0 to 4294967295, therefore converting the decimal number to binary equivalent.

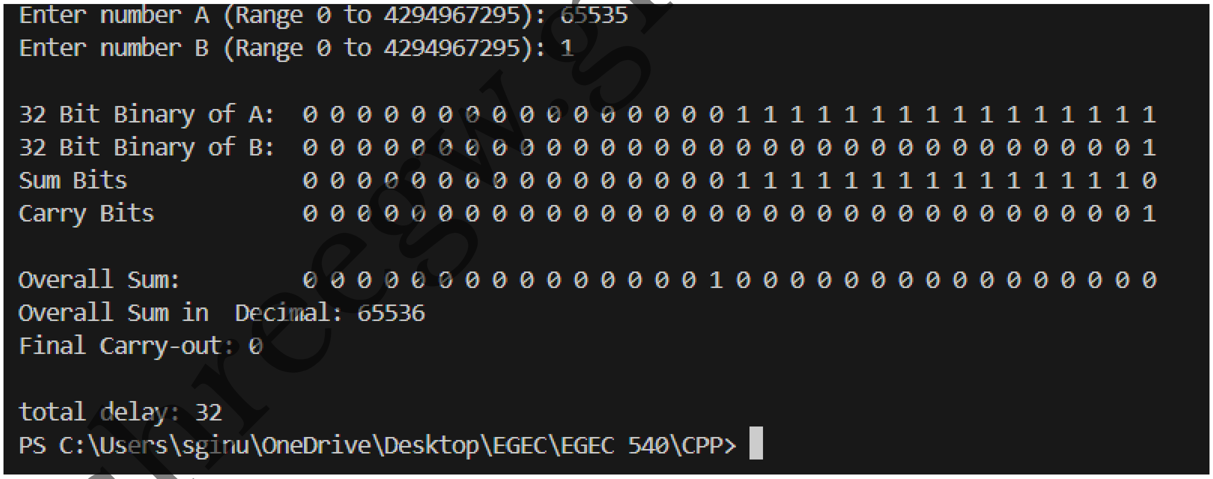
It returns an array of size 32 with the binary equivalent. The program utilizes this function twice to convert Number A and Number B to 32-bit binary equivalent by padding zeros in front of it if required.

2. void fullAdder (array<unsigned long int, 32>& A,
                    array<unsigned long int, 32>& B
                    array<unsigned long int, 32>& Summ,
                     array<unsigned long int, 32>& Summstore,
                     array<unsigned long int, 32>& Carstore,
                     unsigned long int& Lcout, int& delayCount)

   This function is a Full Adder that performs addition on two 32-bit arrays by taking one pair of binary numbers at a time and storing the sum bits of each pair in Summstore and carry bits in Carstore array. Along with that it also counts delay for each iteration
3. int main ()
   This function contains code to display the outputs and is also used in Part 2 to generate 1600 pairs of random numbers.

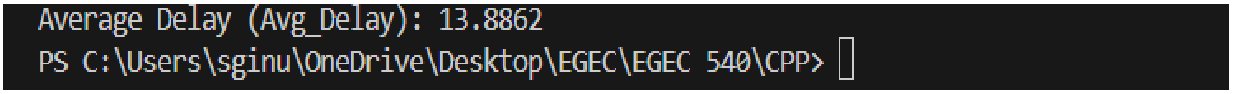
2. Result and Conclusion

For decimal number pair entered, the program converts the number to binary and performs addition. For each iteration the binary pairs generate a sum bit and carry bit which are stored in two 32-bit arrays, **Summstore** and **Carstore.** After the first iteration the delay is set to 2D and with each iteration until **Carstore** array becomes an all-zeroes array, and the function calculates the delay by adding 2D. Fig. 3 shows the output of the program where inputs are A = 65535 and B = 1 which converts the numbers to binary, adds them and converts back to decimal. Fig 4. Consists of output from Part 2.



```
Enter number A (Range 0 to 4294967295): 65535
Enter number B (Range 0 to 4294967295): 1

32 Bit Binary of A:  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
32 Bit Binary of B:  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
Sum Bits             0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0
Carry Bits           0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

Overall Sum:         0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Overall Sum in  Decimal: 65536
Final Carry-out: 0

total delay: 32
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
```

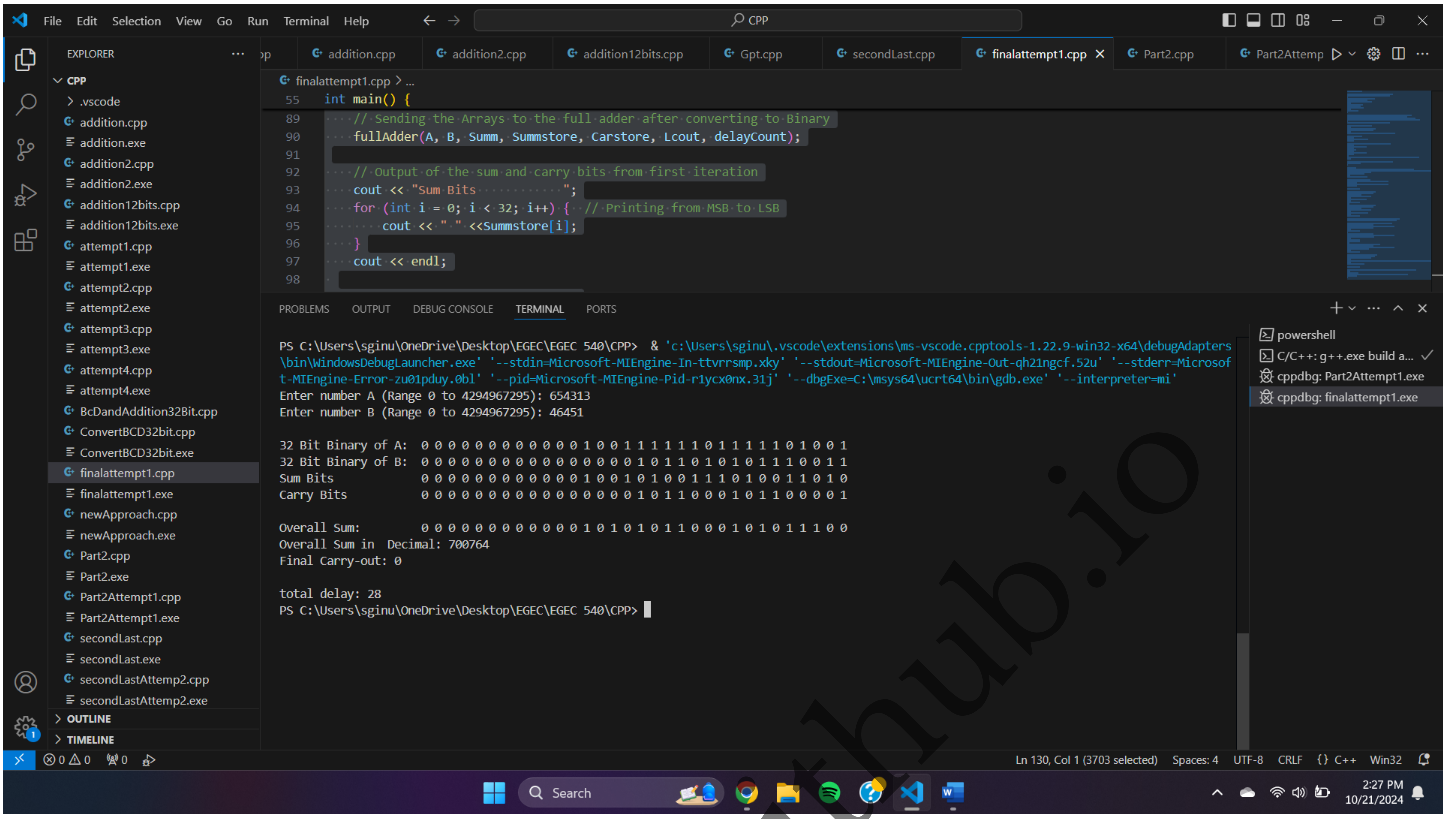
Fig. 3: Code Output of Part 1



```
Average Delay (Avg_Delay): 13.8862
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
```

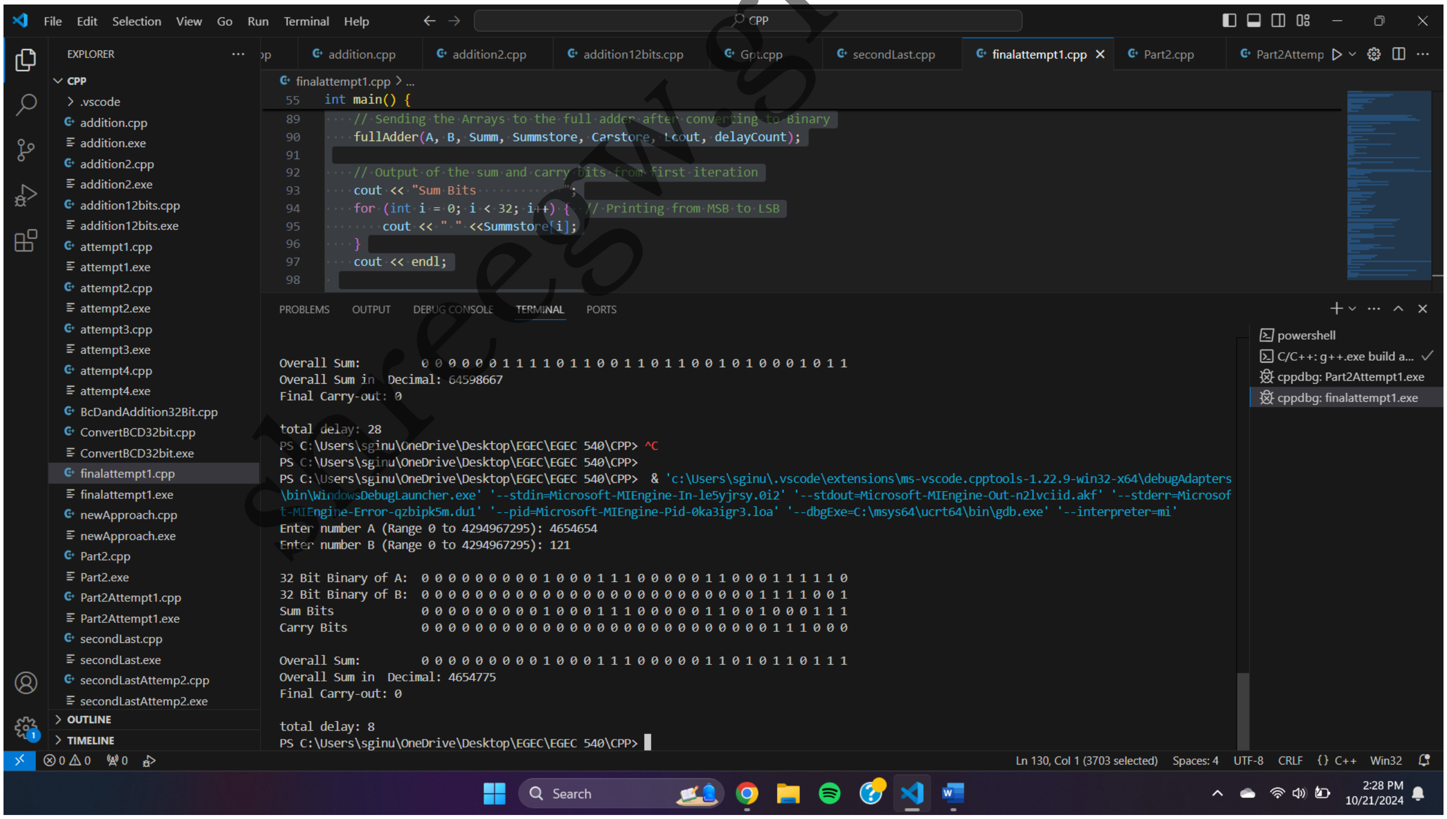
Fig. 4: Code Output of Part 2

Results Contd.



```cpp
int main() {
    // Sending the Arrays to the full adder after converting to Binary
    fullAdder(A, B, Summ, Summstore, Carstore, Lcout, delayCount);

    // Output of the sum and carry bits from first iteration
    cout << "Sum Bits              ";
    for (int i = 0; i < 32; i++) {  // Printing from MSB to LSB
        cout << " " <<Summstore[i];
    }
    cout << endl;
```

```
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-ttvrrsmp.xky' '--stdout=Microsoft-MIEngine-Out-qh21ngcf.52u' '--stderr=Microsof
t-MIEngine-Error-zu01pduy.0bl' '--pid=Microsoft-MIEngine-Pid-r1ycx0nx.31j' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Enter number A (Range 0 to 4294967295): 654313
Enter number B (Range 0 to 4294967295): 46451

32 Bit Binary of A:  0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1
32 Bit Binary of B:  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 0 1 0 1 1 1 0 0 1 1
Sum Bits             0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 0 1 1 1 0 1 0 0 1 1 0 1 0
Carry Bits           0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 0 0 1 0 1 1 0 0 0 0 1

Overall Sum:         0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 1 1 0 0 0 1 0 1 0 1 1 1 0 0
Overall Sum in  Decimal: 700764
Final Carry-out: 0

total delay: 28
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
```



```cpp
int main() {
    // Sending the Arrays to the full adder after converting to Binary
    fullAdder(A, B, Summ, Summstore, Carstore, Lcout, delayCount);

    // Output of the sum and carry bits from first iteration
    cout << "Sum Bits              ";
    for (int i = 0; i < 32; i++) {  // Printing from MSB to LSB
        cout << " " <<Summstore[i];
    }
    cout << endl;
```

```
Overall Sum:         0 0 0 0 0 0 1 1 1 1 0 1 1 0 0 1 1 0 1 1 0 0 1 0 1 0 0 0 1 0 1 1
Overall Sum in  Decimal: 64598667
Final Carry-out: 0

total delay: 28
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-le5yjrsy.0i2' '--stdout=Microsoft-MIEngine-Out-n2lvciid.akf' '--stderr=Microsof
t-MIEngine-Error-qzbipk5m.du1' '--pid=Microsoft-MIEngine-Pid-0ka3igr3.loa' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Enter number A (Range 0 to 4294967295): 4654654
Enter number B (Range 0 to 4294967295): 121

32 Bit Binary of A:  0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 1 0 0 0 0 0 1 1 0 0 0 1 1 1 1 1 0
32 Bit Binary of B:  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 1
Sum Bits             0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 1 0 0 0 0 0 1 1 0 0 1 0 0 0 1 1 1
Carry Bits           0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0

Overall Sum:         0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 1 0 0 0 0 0 1 1 0 1 0 1 1 0 1 1 1
Overall Sum in  Decimal: 4654775
Final Carry-out: 0

total delay: 8
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
```

CPP Code

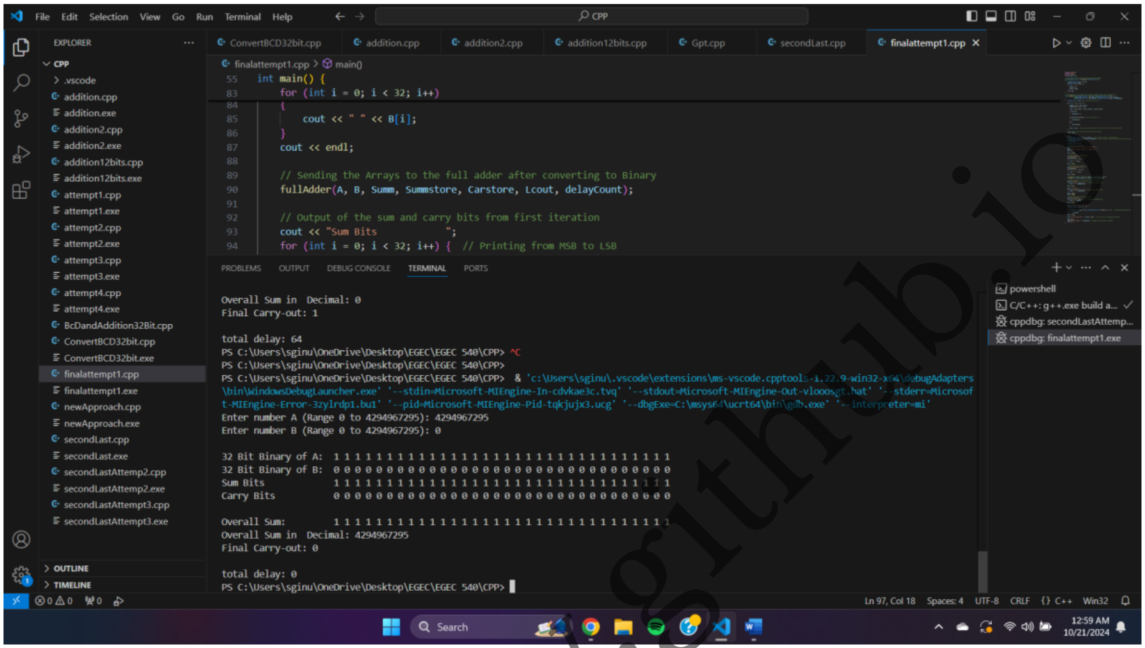
```cpp
#include <iostream>
#include <array>
#include <math.h>
using namespace std;

// This Function Converts Decimal to Binary Equivalent Number
array<unsigned long int, 32> bcdConv(unsigned long int A)
{
    array<unsigned long int, 32> bcd;
    unsigned long int temp;
    for (int i = 31; i >= 0; i--)
    {
        temp = A % 2;
        A = A / 2;
        bcd[i] = temp;
    }
    return bcd;
}

// This Function is a full adder which stores the value of each step for sum and carry
void fullAdder(array<unsigned long int, 32>& A, array<unsigned long int, 32>& B,
               array<unsigned long int, 32>& Summ, array<unsigned long int, 32>& Summstore,
               array<unsigned long int, 32>& Carstore, unsigned long int& Lcout, int& delayCount)
{
    unsigned long int Carin = 0;  //Initial Carry in of 0

    for (int i = 31; i >= 0; i--)
    {
        Summ[i] = A[i] ^ B[i] ^ Carin;
        Summstore[i] = A[i] ^ B[i];

        Lcout = (A[i] & B[i]) | (B[i] & Carin) | (Carin & A[i]);

        if(Lcout!=0)
        {
            delayCount += 2;
        }

        if(A[i]==B[i] && A[i]==1) //if A=1, B=1 Carry is 1
        {
            Carstore[i]=1;
        }
        else
        {
            Carstore[i]=0;
        }

        Carin = Lcout;  //Lcout is the carryout which will be set to carry in for next iteration
    }

    //the value of Carin will be used for the final carry-out
    Lcout = Carin;
}

int main() {
    array<unsigned long int, 32> A, B, Summ; // 32 bit arrays to store A, B and Final Sum
    array<unsigned long int, 32> Carstore, Summstore; // 32 Bit arrays to store values of Carry and Sum for each pair of A and B
    unsigned long int Lcout = 0;
    int delayCount = 0;

    unsigned long int a, b;

    // Input A and B in Decimal
    cout << "Enter number A (Range 0 to 4294967295): ";
    cin >> a;
    cout << "Enter number B (Range 0 to 4294967295): ";
    cin >> b;

    cout << endl;

    // Convert Decimal to Binary for A
    cout << "32 Bit Binary of A: ";
    A = bcdConv(a);
    for (int i = 0; i < 32; i++)
    {
        cout << " "<<A[i];
    }
    cout << endl;

    // Convert Decimal to Binary for B
    cout << "32 Bit Binary of B: ";
    B = bcdConv(b);
    for (int i = 0; i < 32; i++)
    {
        cout << " " << B[i];
    }
    cout << endl;

    // Sending the Arrays to the full adder after converting to Binary
    fullAdder(A, B, Summ, Summstore, Carstore, Lcout, delayCount);

    // Output of the sum and carry bits from first iteration
    cout << "Sum Bits            ";
    for (int i = 0; i < 32; i++) {  // Printing from MSB to LSB
        cout << " " <<Summstore[i];
    }
    cout << endl;

    cout << "Carry Bits          ";
          for (int i = 0; i < 32; i++) {  // Printing from MSB to LSB
              cout << " "<< Carstore[i];
          }
          cout << endl;
          cout << endl;

          // Output the result (Sum and Carry-out)
          cout << "Overall Sum:        ";
          for (int i = 0; i < 32; i++) {  // Printing from MSB to LSB
              cout <<" " <<Summ[i];
          }
          cout << endl;

          cout << "Overall Sum in  Decimal: ";
          unsigned long int decimalSum = 0;
          for (int i = 0; i < 32; i++)
          {
              decimalSum += Summ[i] * static_cast<unsigned long int>(pow(2, 31-i));  // Converting binary addition value to decimal
          }
          cout << decimalSum;

          cout << endl;

          cout << "Final Carry-out: " << Lcout << endl;  // Last Carry Out to display any overflow

          cout<<endl;
          cout << "total delay: " << delayCount; // Displays overall Delay of the adder
          return 0;
      }

```

Code Verification

(i) 4294967295 + 0

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0



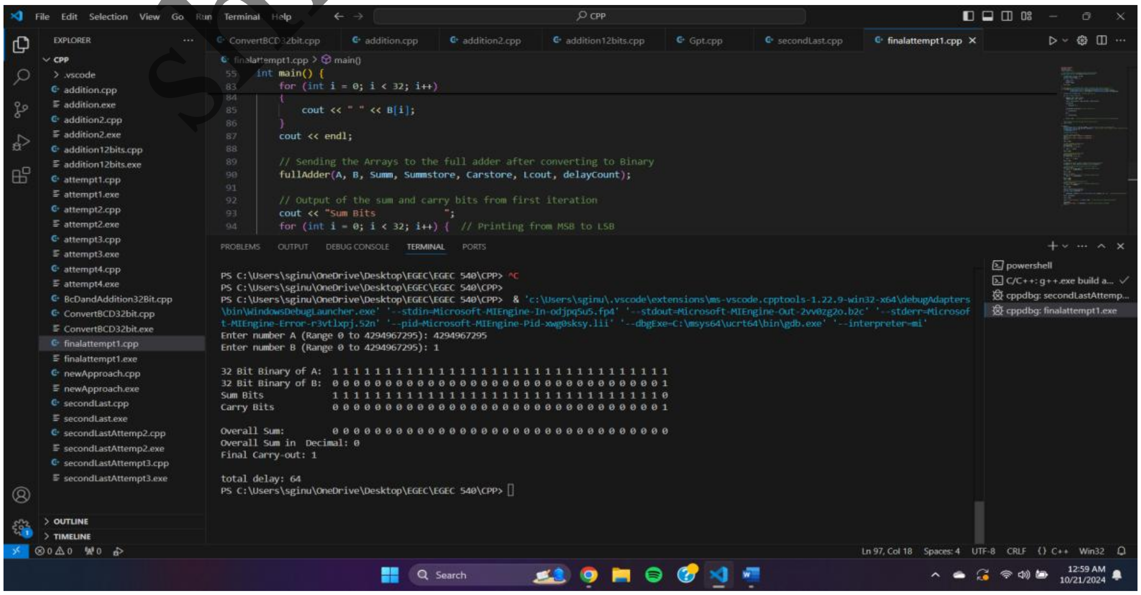
```
int main() {
    for (int i = 0; i < 32; i++)
    {
        cout << " " << B[i];
    }
    cout << endl;

    // Sending the Arrays to the full adder after converting to Binary
    fullAdder(A, B, Summ, Summstore, Carstore, Lcout, delayCount);

    // Output of the sum and carry bits from first iteration
    cout << "Sum Bits              ";
    for (int i = 0; i < 32; i++) {  // Printing from MSB to LSB
```

```
Overall Sum in  Decimal: 0
Final Carry-out: 1

total delay: 64
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-cdvkae3c.tvq' '--stdout=Microsoft-MIEngine-Out-vlooosgt.hat' '--stderr=Microsof
t-MIEngine-Error-3zylrdp1.bu1' '--pid=Microsoft-MIEngine-Pid-tqkjujx3.ucg' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Enter number A (Range 0 to 4294967295): 4294967295
Enter number B (Range 0 to 4294967295): 0

32 Bit Binary of A:  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
32 Bit Binary of B:  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Sum Bits             1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Carry Bits           0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Overall Sum:         1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Overall Sum in  Decimal: 4294967295
Final Carry-out: 0

total delay: 0
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> 
```

(ii) 4294967295 + 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1



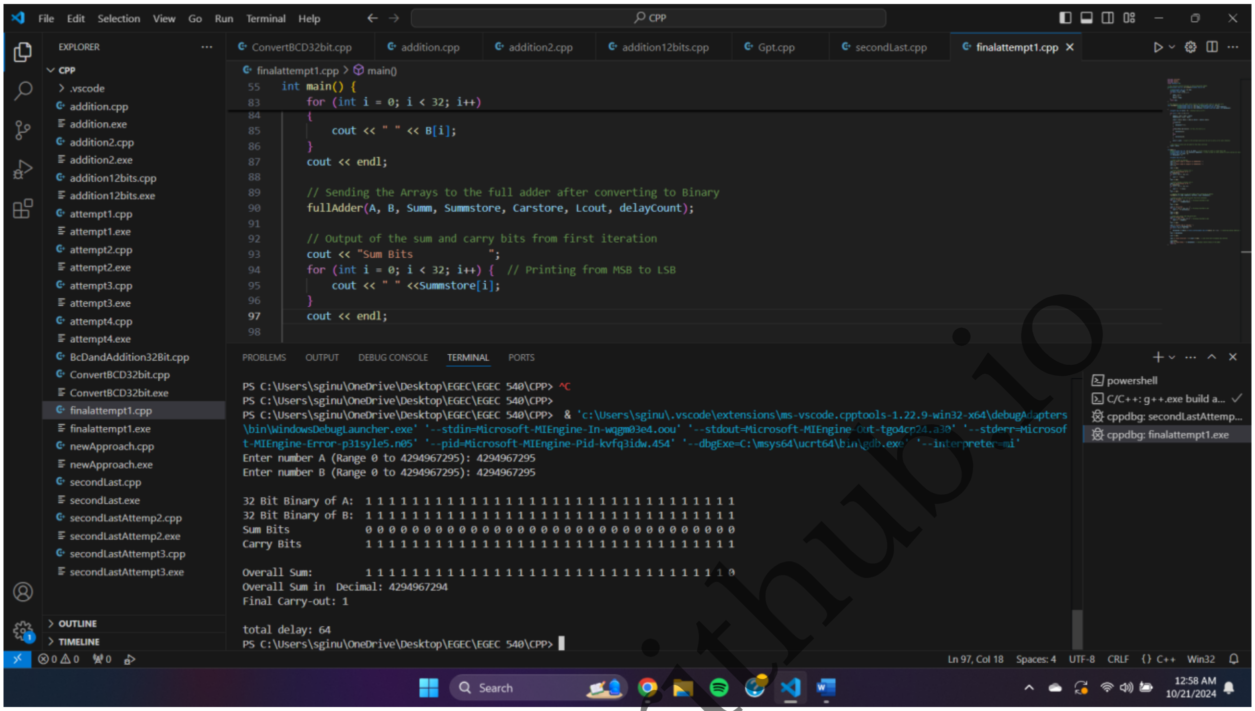
```
int main() {
    for (int i = 0; i < 32; i++)
    {
        cout << " " << B[i];
    }
    cout << endl;

    // Sending the Arrays to the full adder after converting to Binary
    fullAdder(A, B, Summ, Summstore, Carstore, Lcout, delayCount);

    // Output of the sum and carry bits from first iteration
    cout << "Sum Bits              ";
    for (int i = 0; i < 32; i++) {  // Printing from MSB to LSB
```

```
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-odjpq5u5.fp4' '--stdout=Microsoft-MIEngine-Out-2vv0zg2o.b2c' '--stderr=Microsof
t-MIEngine-Error-r3vtlxpj.52n' '--pid=Microsoft-MIEngine-Pid-xwg0sksy.lii' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Enter number A (Range 0 to 4294967295): 4294967295
Enter number B (Range 0 to 4294967295): 1

32 Bit Binary of A:  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
32 Bit Binary of B:  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
Sum Bits             1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0
Carry Bits           0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

Overall Sum:         0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Overall Sum in  Decimal: 0
Final Carry-out: 1

total delay: 64
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> 
```

(iii) 4294967295 + 4294967295

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

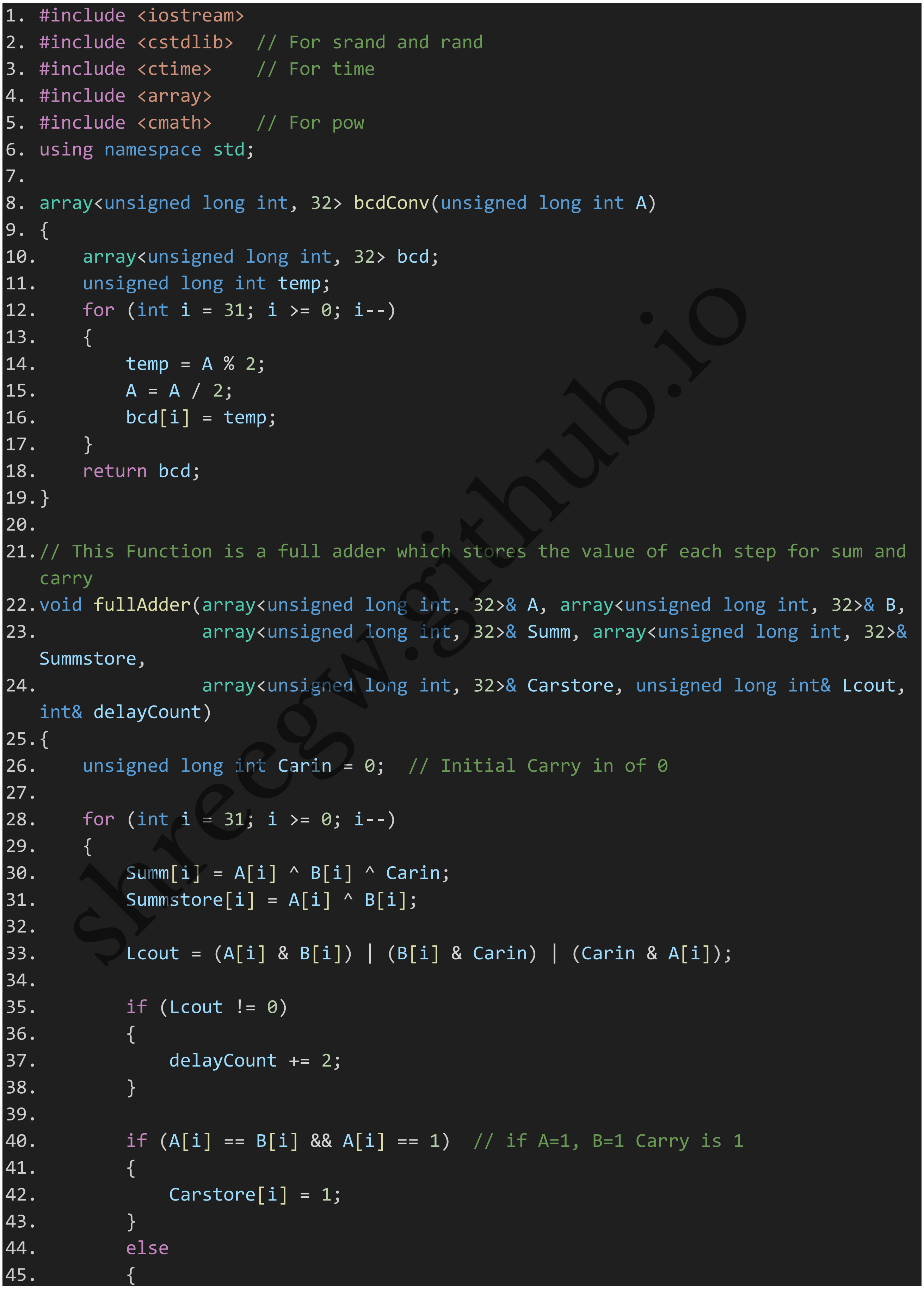
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1



```cpp
int main() {
    for (int i = 0; i < 32; i++)
    {
        cout << " " << B[i];
    }
    cout << endl;

    // Sending the Arrays to the full adder after converting to Binary
    fullAdder(A, B, Summ, Summstore, Carstore, Lcout, delayCount);

    // Output of the sum and carry bits from first iteration
    cout << "Sum Bits             ";
    for (int i = 0; i < 32; i++) {  // Printing from MSB to LSB
        cout << " " <<Summstore[i];
    }
    cout << endl;

```

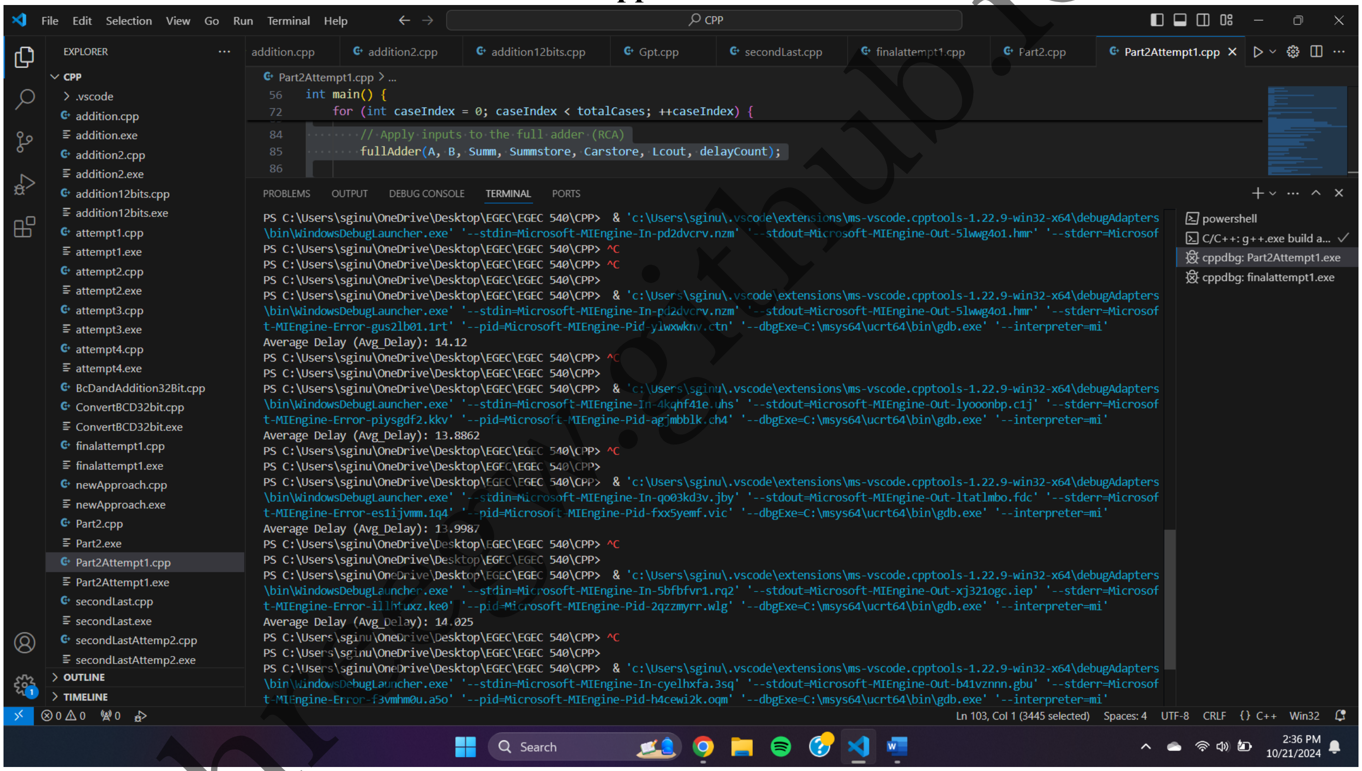
```
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-wqgm03e4.oou' '--stdout=Microsoft-MIEngine-Out-tgo4cp24.a30' '--stderr=Microsof
t-MIEngine-Error-p31syle5.n05' '--pid=Microsoft-MIEngine-Pid-kvfq3idw.454' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Enter number A (Range 0 to 4294967295): 4294967295
Enter number B (Range 0 to 4294967295): 4294967295

32 Bit Binary of A:  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
32 Bit Binary of B:  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Sum Bits             0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Carry Bits           1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Overall Sum:         1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0
Overall Sum in  Decimal: 4294967294
Final Carry-out: 1

total delay: 64
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
```

Code to test delay for 1600 Cases with random inputs



```cpp
#include <iostream>
#include <cstdlib>  // For srand and rand
#include <ctime>    // For time
#include <array>
#include <cmath>    // For pow
using namespace std;

array<unsigned long int, 32> bcdConv(unsigned long int A)
{
    array<unsigned long int, 32> bcd;
    unsigned long int temp;
    for (int i = 31; i >= 0; i--)
    {
        temp = A % 2;
        A = A / 2;
        bcd[i] = temp;
    }
    return bcd;
}

// This Function is a full adder which stores the value of each step for sum and carry
void fullAdder(array<unsigned long int, 32>& A, array<unsigned long int, 32>& B,
               array<unsigned long int, 32>& Summ, array<unsigned long int, 32>& Summstore,
               array<unsigned long int, 32>& Carstore, unsigned long int& Lcout, int& delayCount)
{
    unsigned long int Carin = 0;  // Initial Carry in of 0

    for (int i = 31; i >= 0; i--)
    {
        Summ[i] = A[i] ^ B[i] ^ Carin;
        Summstore[i] = A[i] ^ B[i];

        Lcout = (A[i] & B[i]) | (B[i] & Carin) | (Carin & A[i]);

        if (Lcout != 0)
        {
            delayCount += 2;
        }

        if (A[i] == B[i] && A[i] == 1)  // if A=1, B=1 Carry is 1
        {
            Carstore[i] = 1;
        }
        else
        {
            Carstore[i] = 0;
        }

        Carin = Lcout;  // Lcout is the carryout which will be set to carry in for next iteration
    }

    // the value of Carin will be used for the final carry-out
    Lcout = Carin;
}

int main() {
    const int size = 40;  // size of the array for random numbers
    const int totalCases = 1600;  // Total pairs of inputs (1600)
    unsigned int randomNumbersA[size];  // Array to store the pseudorandom numbers A
    unsigned int randomNumbersB[size];  // Array to store the pseudorandom numbers B
    array<unsigned long int, 32> A, B, Summ;  // 32 bit arrays to store A, B and Final Sum
    array<unsigned long int, 32> Carstore, Summstore;  // 32 Bit arrays to store values of Carry and Sum for each pair of A and B
    unsigned long int Lcout = 0;
    float Avg_Delay = 0.0f;  // Variable to store the average delay
    int delayCount = 0;
    int D_array[totalCases];  // Array to store delay values for 1600 cases

    // Seed the random number generator with current time
    srand(static_cast<unsigned int>(time(0)));

    // Generate and store 1600 pseudorandom unsigned int numbers
    for (int caseIndex = 0; caseIndex < totalCases; ++caseIndex) {
        // Generate two pseudorandom numbers
        unsigned int randomA = rand() % 4294967295;
        unsigned int randomB = rand() % 4294967295;

        // Convert to 32-bit binary arrays (BCD format)
        A = bcdConv(randomA);
        B = bcdConv(randomB);

        // Reset delay counter
        delayCount = 0;

        // Apply inputs to the full adder (RCA)
        fullAdder(A, B, Summ, Summstore, Carstore, Lcout, delayCount);

        // Store the delay for this pair in D_array
        D_array[caseIndex] = delayCount;
    }

    // Calculate the average delay by summing all delays and dividing by 1600
    int totalDelay = 0;
    for (int i = 0; i < totalCases; i++) {
        totalDelay += D_array[i];
    }
    Avg_Delay = static_cast<float>(totalDelay) / totalCases;

    // Display the average delay
    cout << "Average Delay (Avg_Delay): " << Avg_Delay << endl;

          return 0;
      }

```

Appendix D



```
int main() {
    for (int caseIndex = 0; caseIndex < totalCases; ++caseIndex) {
        // Apply inputs to the full adder (RCA)
        fullAdder(A, B, Summ, Summstore, Carstore, Lcout, delayCount);
```

```
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-pd2dvcrv.nzm' '--stdout=Microsoft-MIEngine-Out-5lwwg4o1.hmr' '--stderr=Microsof
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-pd2dvcrv.nzm' '--stdout=Microsoft-MIEngine-Out-5lwwg4o1.hmr' '--stderr=Microsof
t-MIEngine-Error-gus2lb01.1rt' '--pid=Microsoft-MIEngine-Pid-ylwxwknv.ctn' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Average Delay (Avg_Delay): 14.12
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-4kqhf41e.uhs' '--stdout=Microsoft-MIEngine-Out-lyooonbp.c1j' '--stderr=Microsof
t-MIEngine-Error-piysgdf2.kkv' '--pid=Microsoft-MIEngine-Pid-agjmbblk.ch4' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Average Delay (Avg_Delay): 13.8862
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-qo03kd3v.jby' '--stdout=Microsoft-MIEngine-Out-ltatlmbo.fdc' '--stderr=Microsof
t-MIEngine-Error-es1ijvmm.1q4' '--pid=Microsoft-MIEngine-Pid-fxx5yemf.vic' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Average Delay (Avg_Delay): 13.9987
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-5bfbfvr1.rq2' '--stdout=Microsoft-MIEngine-Out-xj321ogc.iep' '--stderr=Microsof
t-MIEngine-Error-illhtuxz.ke0' '--pid=Microsoft-MIEngine-Pid-2qzzmyrr.wlg' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Average Delay (Avg_Delay): 14.025
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP> ^C
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>
PS C:\Users\sginu\OneDrive\Desktop\EGEC\EGEC 540\CPP>  & 'c:\Users\sginu\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters
\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-cyelhxfa.3sq' '--stdout=Microsoft-MIEngine-Out-b41vznnn.gbu' '--stderr=Microsof
t-MIEngine-Error-f3vmhm0u.a5o' '--pid=Microsoft-MIEngine-Pid-h4cewi2k.oqm' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
```