**Experiment / Assignment / Tutorial No. 2**

**Grade: AA / AB / BB / BC / CC / CD /DD**

**Signature of the Staff In-charge with date**

|  |
| --- |
| B**atch: A1 Roll No.: 1911004 Experiment / assignment / tutorial No.: 2** |

|  |
| --- |
| **Title:** Binary Adders and Subtractors |

**Objective:** To implement half and full adder–subtractor using gates and IC 7483

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**\_\_\_\_\_\_\_\_

**Expected Outcome of Experiment:**

**CO2:** Use different minimization technique and solve combinational circuits, synchronous & asynchronous sequential circuits.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

* **VLab Link:** [http://vlabs.iitb.ac.in/vlabs-dev/labs/dldesignlab/experimentlist.html](http://physics.niser.ac.in/labmanuals/sem5/elect/7_ADDER%20SUBTRACTOR%20CIRCUITS.pdf)
* R. P. Jain, “Modern Digital Electronics”, Tata McGraw Hill
* M .Morris Mano, “Digital Logic & computer Design”, PHI
* http://physics.niser.ac.in/labmanuals/sem5/elect/7\_ADDER%20SUBTRACTO  [R%20CIRCUITS.pd](http://vlabs.iitb.ac.in/vlabs-dev/labs/dldesignlab/experimentlist.html)f

**Pre Lab/ Prior Concepts:**

**Adder:** Addition of two binary digits is most basic operation performed by the digital computer. There are two types of adder:

* Half adder
* Full adder

**Half Adder:** Half adder is combinational logic circuit with two inputs and two outputs. It is the basic building block for addition of two single bit numbers.

**Full adder:** A half adder has a provision not to add a carry coming from the lower order bits when multi bit addition is performed. for this purpose a third input terminal is added and this circuits is to add A,B,C where A and B are the nth order bits of the number A and B respectively and C is the carry generated from the addition of (n-1) order bits. This circuit is referred to as full adder.

**Subtractor:** Subtraction of two binary digits is one of the most basic operations performed by digital computer .there are two types of subtractor:

* Half subtractor
* Full subtractor

**Half subtractor:** Logic circuit for the subtraction of B from A where A,B are 1 bit numbers is referred to as half subtract or .the subtract or process has two input and difference and borrow are the two outputs.

**Full subtractor:** As in the case of the addition using logic gates, a full subtractor is made by combining two half-sub tractors and an additional OR-gate. A full subtractor has the borrow in capability (denoted as BORIN) and so allows cascading which results in the possibility of multi-bit subtraction.

**IC 7483**

For subtraction of one binary number from another, we do so by adding 2’s complement of the former to the latter number using a full adder circuit.

IC 7483 is a 16 pin, 4-bit full adder. This IC has a provision to add the carry output to transfer and end around carry output using Co and C4 respectively.

**2’s complement:** 2’s complement of any binary no. can be obtained by adding 1 in 1’scomplement of that no.

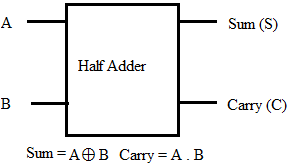
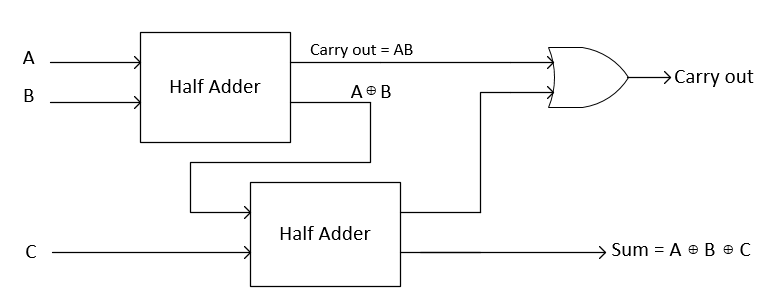
e.g. 2’s complement of +(10)10 =1010is

|  |  |  |  |
| --- | --- | --- | --- |
| 1C of 1010 |  | | 0101 |
|  |  | + | 1 |
| -(10)10 |  | | 0110 |

In 2’s complement subtraction using IC 7483, we are representing negative number in 2’s complement form and then adding it with 1st number.

**Implementation Details:**

**Half Adder Block Diagram Half Adder Circuit**

**A & B 🡺 INPUTS S & C 🡺 OUTPUTS**

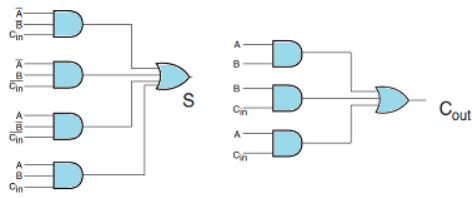
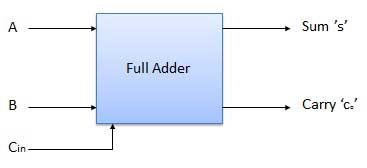
**Truth Table for Half Adder**

|  |  |  |  |
| --- | --- | --- | --- |
| **Inputs** | | **Outputs** | |
| **A** | **B** | **Sum** | **Carry** |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

**From the truth table (with steps):**

**S=A’.B+A.B’ C=A.B**

**Full Adder Block Diagram** **Full Adder Circuit**



**Truth Table for Full Adder**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **INPUT** | | | **OUTPUT** | |
| **A** | **B** | **Carry (in)** | **SUM** | **CARRY(C)** |
| **0** | **0** | **0** | **0** | **0** |
| **0** | **1** | **0** | **1** | **0** |
| **1** | **0** | **0** | **1** | **0** |
| **1** | **1** | **0** | **0** | **1** |
| **0** | **0** | **1** | **1** | **0** |
| **0** | **1** | **1** | **0** | **1** |
| **1** | **0** | **1** | **0** | **1** |
| **1** | **1** | **1** | **1** | **1** |

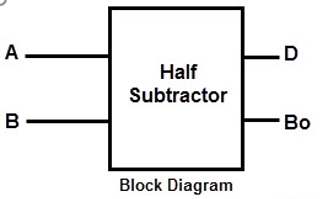
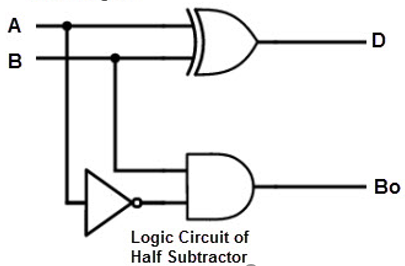
**From the truth table (with steps):**

**S= A’.B’.Cin + A’.B.Cin’ + A.B’.Cin’ + A.B.Cin**

**C= A’.B.Cin + A.B’.Cin + A.B.Cin’ + A.B.Cin**

**C = (A’.B.Cin + A.B.Cin) + (A.B’.Cin + A.B.Cin) + (A.B.Cin’ + A.B.Cin)**

**Half Subtractor Block Diagram** **Half Subtractor Circuit**

**A & B 🡺 INPUTS D & BO 🡺 OUTPUTS**

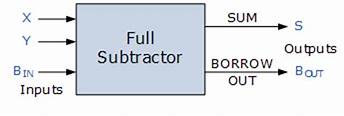
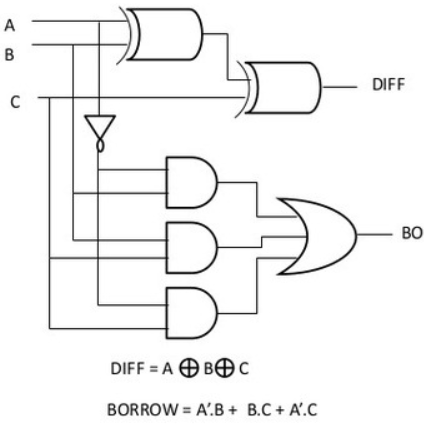
**Truth Table for Half Subtractor**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **INPUT** | | **OUTPUT** | |  |
| **A** | **B** | **DIFFERENCE(D)** | **BORROW(Bo)** |  |
|  |  |  |  |  |
| 0 | 0 | 0 | 0 |  |
|  |  |  |  |  |
| 1 | 0 | 1 | 0 |  |
|  |  |  |  |  |
| 0 | 1 | 1 | 1 |  |
|  |  |  |  |  |
| 1 | 1 | 0 | 0 |  |
|  |  |  |  |  |

**From the truth table (with steps) :**

**D= A’.B+A.B’ BO=A’.B**

**Full Subtractor Block Diagram Full Subtractor Circuit**

**X , Y & BIN 🡺 INPUTS D(S) & BO 🡺 OUTPUTS**

**Truth Table for Full subtractor**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **B** | **BIN** | **D** | **BOROUT** |
|  |  |  |  |  |
| **0** | **0** | **0** | **0** | **0** |
|  |  |  |  |  |
| **0** | **0** | **1** | **1** | **1** |
|  |  |  |  |  |
| **0** | **1** | **0** | **1** | **1** |
|  |  |  |  |  |
| **0** | **1** | **1** | **0** | **1** |
|  |  |  |  |  |
| **1** | **0** | **0** | **1** | **0** |
|  |  |  |  |  |
| **1** | **0** | **1** | **0** | **0** |
|  |  |  |  |  |
| **1** | **1** | **0** | **0** | **0** |
|  |  |  |  |  |
| **1** | **1** | **1** | **1** | **1** |
|  |  |  |  |  |

**From the truth table (with steps):**

**D= A’.B’.Bin + A’.B.Bin’ + A.B’.Bin’ + A.B.Bin**

**B= A’.B’.Bin + A’.B.Bin’ + A’.B.Bin + A.B.Bin**

**B= (A’.B’.Bin + A’.B.Bin) + (A’.B.Bin’ + A’.B.Bin) + (A.B.Bin + A’.B.Bin)**

**B= A’.Bin + A’.B + B.B**

**IC 7483**

**Procedure:**

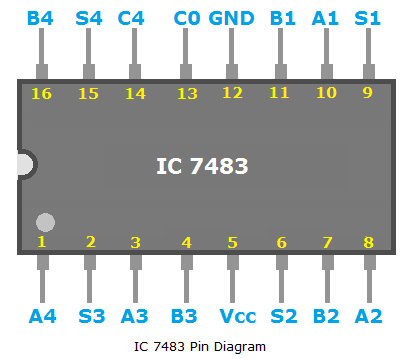
1. **Locate the IC 7483 and 4-not gates block on trainer kit.**
2. **Connect 1st i/p no. to A4-A1 i/p slot & 2nd (negative) no. to B4-B1 through 4-not gates (1C of 2nd no.)**
3. **Connect high input to Co so that it will get added with 1C of 2nd no. to get 2C.**
4. **Connect 4-bit output to the output indicators.**
5. **Switch ON the power supply and monitor the output for various input combinations.**

**Example:**

|  |  |  |
| --- | --- | --- |
| **1) 710 -210 = 510** | |  |
| **7** |  | **0111** |
| **2** |  | **0010** |
| **1’C of 2** | | **1101** |
|  |  | **+ 1** |
| **2’C of 2** | | **1110** |

0111 + 1110 1 0101

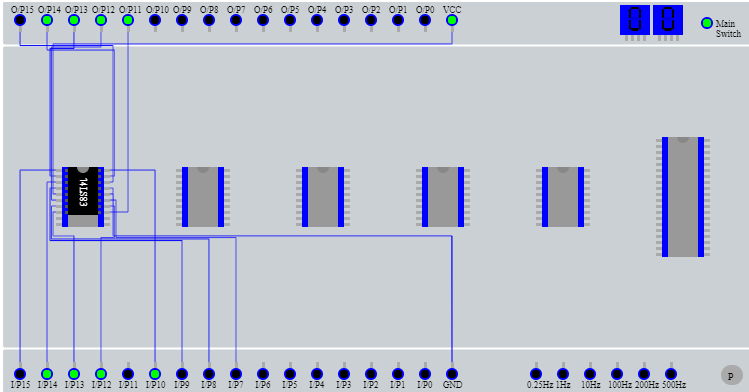
**Pin Diagram IC7483**



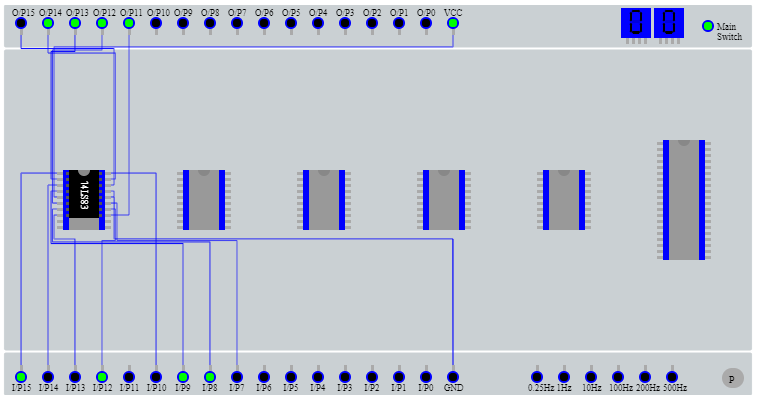
**Addition (Adder)**

**VIRTUAL LAB – IMPLEMENTATION OF BINARY ADDER USING MSI ICs**

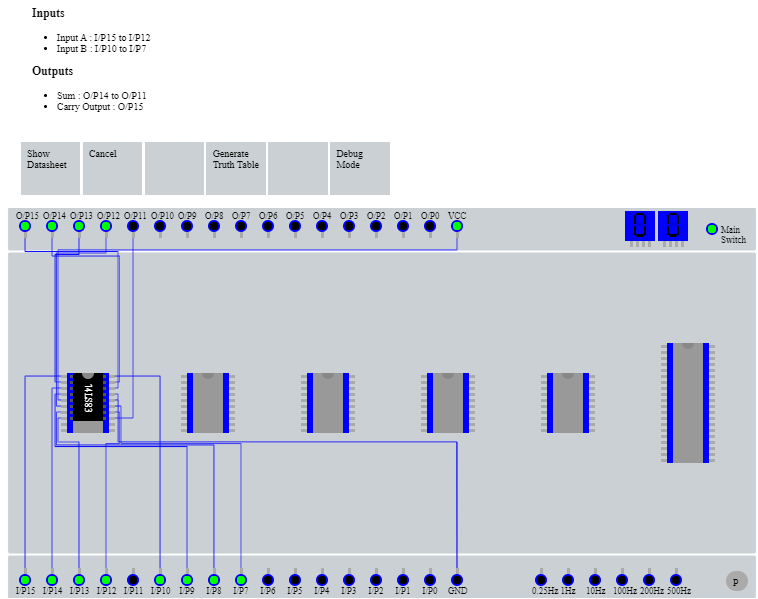
**A)**

****

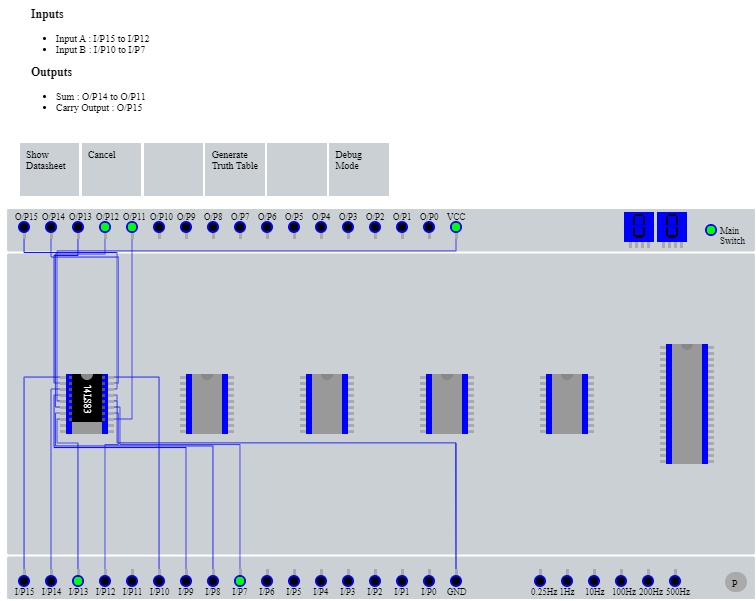
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **I/P 15** | **I/P14** | **I/P13** | **I/P12** |
| **0** | **1** | **1** | **1** |
| **B** | **I/P 10** | **I/P 9** | **I/P 8** | **I/P 7** |
| **1** | **0** | **0** | **0** |
| **A+B(SUM)** | **O/P 14** | **O/P 13** | **O/P 12** | **O/P 11** |
| **1** | **1** | **1** | **1** |
| **C(CARRY)**  **O/P 15** | **0** | | | |

**B)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **I/P 15** | **I/P14** | **I/P13** | **I/P12** |
| **1** | **0** | **0** | **1** |
| **B** | **I/P 10** | **I/P 9** | **I/P 8** | **I/P 7** |
| **0** | **1** | **1** | **0** |
| **A+B(SUM)** | **O/P 14** | **O/P 13** | **O/P 12** | **O/P 11** |
| **1** | **1** | **1** | **1** |
| **C(CARRY)**  **O/P 15** | **0** | | | |

**C)**

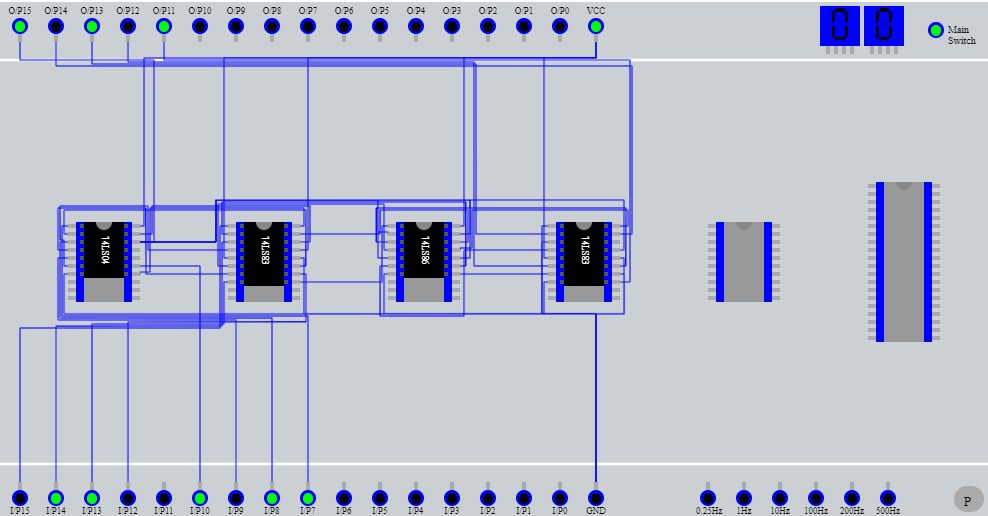
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **I/P 15** | **I/P14** | **I/P13** | **I/P12** |
| **1** | **1** | **1** | **1** |
| **B** | **I/P 10** | **I/P 9** | **I/P 8** | **I/P 7** |
| **1** | **1** | **1** | **1** |
| **A+B(SUM)** | **O/P 14** | **O/P 13** | **O/P 12** | **O/P 11** |
| **1** | **1** | **1** | **0** |
| **C(CARRY)**  **O/P 15** | **1** | | | |

**D)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **I/P 15** | **I/P14** | **I/P13** | **I/P12** |
|  | **0** | **1** | **0** |
| **B** | **I/P 10** | **I/P 9** | **I/P 8** | **I/P 7** |
| **0** | **0** | **0** | **1** |
| **A+B(SUM)** | **O/P 14** | **O/P 13** | **O/P 12** | **O/P 11** |
| **0** | **0** | **1** | **1** |
| **C(CARRY)**  **O/P 15** | **0** | | | |

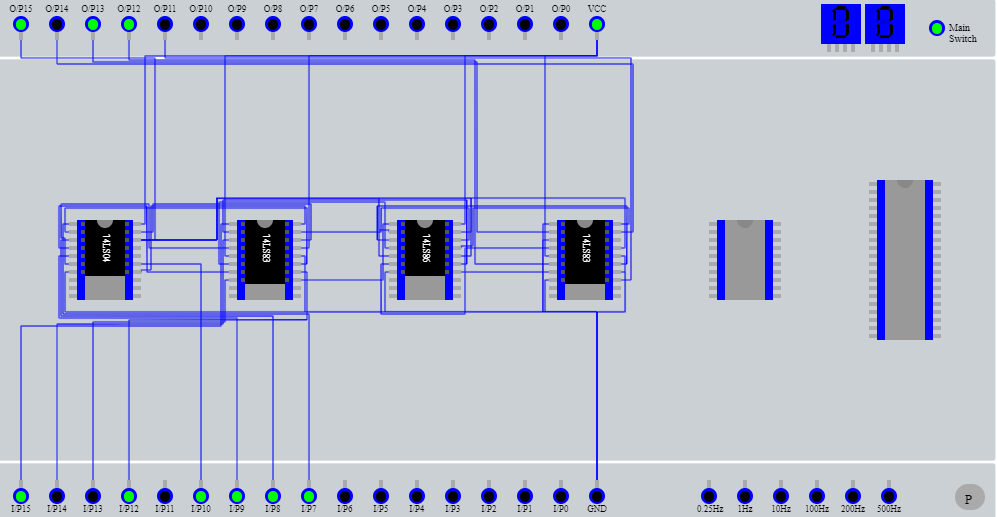
**SUBSTRACTION ( SUBTRATOR )**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **I/P 15** | **I/P14** | **I/P13** | **I/P12** |
| **0** | **1** | **1** | **0** |
| **B** | **I/P 10** | **I/P 9** | **I/P 8** | **I/P 7** |
| **1** | **0** | **1** | **1** |
| **A-B(DIFFERENCE)** | **O/P 14** | **O/P 13** | **O/P 12** | **O/P 11** |
| **0** | **1** | **0** | **1** |
| **C**  **(CARRY/BORROW)**  **O/P 15** | **1** | | | |

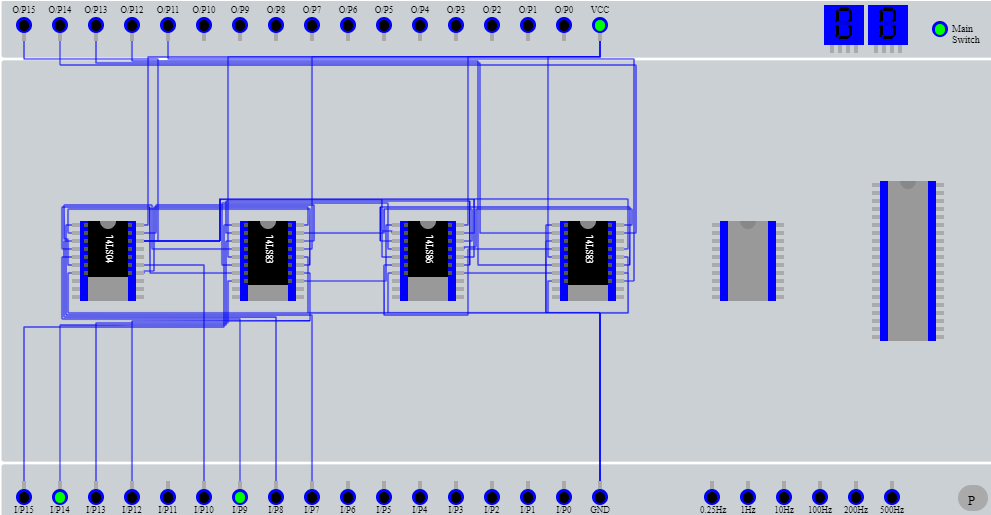




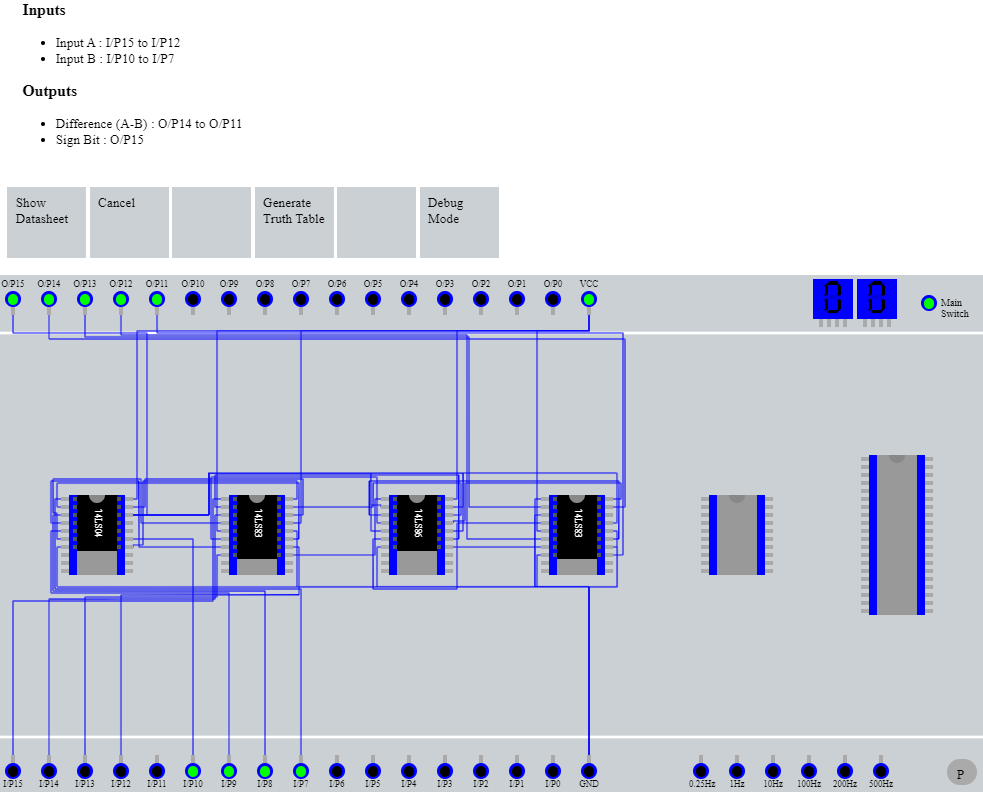
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **I/P 15** | **I/P14** | **I/P13** | **I/P12** |
| **1** | **0** | **0** | **1** |
| **B** | **I/P 10** | **I/P 9** | **I/P 8** | **I/P 7** |
| **1** | **1** | **1** | **1** |
| **A-B(DIFFERENCE)** | **O/P 14** | **O/P 13** | **O/P 12** | **O/P 11** |
| **0** | **1** | **1** | **0** |
| **C**  **(CARRY/BORROW)**  **O/P 15** | **1** | | | |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **I/P 15** | **I/P14** | **I/P13** | **I/P12** |
| **0** | **1** | **0** | **0** |
| **B** | **I/P 10** | **I/P 9** | **I/P 8** | **I/P 7** |
| **0** | **1** | **0** | **0** |
| **A-B(DIFFERENCE)** | **O/P 14** | **O/P 13** | **O/P 12** | **O/P 11** |
| **0** | **0** | **0** | **0** |
| **C**  **(CARRY/BORROW)**  **O/P 15** | **0** | | | |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **I/P 15** | **I/P14** | **I/P13** | **I/P12** |
| **0** | **0** | **0** | **0** |
| **B** | **I/P 10** | **I/P 9** | **I/P 8** | **I/P 7** |
| **1** | **1** | **1** | **1** |
| **A-B(DIFFERENCE)** | **O/P 14** | **O/P 13** | **O/P 12** | **O/P 11** |
| **1** | **1** | **1** | **1** |
| **C**  **(CARRY/BORROW)**  **O/P 15** | **1** | | | |



**Conclusion:**

**The circuit Diagrams with working ,Block Diagram & respective Truth tables of Half Adder, Full Adder, Half Subtractor & Full subtractor were understood & were implemented in Virtual Lab Successfully .**

**Post Lab Descriptive Questions**

1. **What is difference between half and full adder, half and full subtractor?**

|  |  |
| --- | --- |
| **HALF ADDER** | **FULL ADDER** |
| **Half adder is combinational logic circuit with 2 inputs and 2 outputs** | **Full adder is combinational logic circuit with 3 inputs and 2 outputs.** |
| **It is the basic building block for addition of 2 single bit.** | **It is suitable for n bit addition.** |
| **A half adder does not add up a carry coming from the lower order bits when multi bit addition is performed** | **In full Adder a third input terminal is added and this circuit adds A,B,Cin where A and B are the nth order bits of the number A and B respectively and C is the carry generated from the addition of (n-1) order bit.** |

|  |  |
| --- | --- |
| **HALF SUBTRACTOR** | **FULL SUBTRACTOR** |
| **Half subtractor is combinational logic circuit with 2 inputs and 2 outputs** | **Full adder is combinational logic circuit with 3 inputs and 2 outputs** |
| **It is the basic building block for subtraction of 2 single bits.** | **It is suitable for n bit subtraction.** |
| **A half subtractor does not consider a borrow coming from the lower order bits when multi bit subtraction is performed.** | **In full subtractor a third input terminal is added in the full subtractor and this circuit has A,B,C where A and B are the nth order bits of the number A and B respectively and C is the borrow generated from the subtraction of (n-1) order bits.** |

1. **Perform the following Binary subtraction with the help of appropriate ICs:**
2. **7-5**

|  |  |  |
| --- | --- | --- |
| **710 -510 = 210** | |  |
| **7** |  | **0111** |
| **5** |  | **0101** |
| **1’C of 5** | | **1010** |
|  |  | **+ 1** |
| **2’C of 5** | | **1011** |

**7+(-5) = 0111 + 1011 = ( 0010 ) 2 = 210**

1. **5-7**

|  |  |  |
| --- | --- | --- |
| 1. **510 -710 = -210** | |  |
| **5** |  | **0101** |
| **7** |  | **0111** |
| **1’C of 7** | | **1000** |
|  |  | **+ 1** |
| **2’C of 7** | | **1001** |

**5+(-7) = 0101 + 1001 = ( 1110 ) 2 = -210  2s’ complement of 1110 = 0001+1 = (0010) = 210**

1. **9-4**

|  |  |  |
| --- | --- | --- |
| 1. **910 -410 =5 10** | |  |
| **9** |  | **1001** |
| **4** |  | **0100** |
| **1’C of 4** | | **1011** |
|  |  | **+ 1** |
| **2’C of 4** | | **1100** |

**9+(-4) = 1001 + 1100 = ( 0101 ) 2 = 510**