|  |  |  |  |
| --- | --- | --- | --- |
| **Course Name:** | **Microprocessors and Peripherals (2UXC404)** | **Semester:** | **IV** |
| **Date of Performance:** | 20/1/2021 | **Batch No:** | B2 |
| **Faculty Name:** | KCS | **Roll No:** | 1912052 |
| **Faculty Sign & Date:** |  | **Grade/Marks:** | \_\_\_/25 |

**Experiment No: 1**

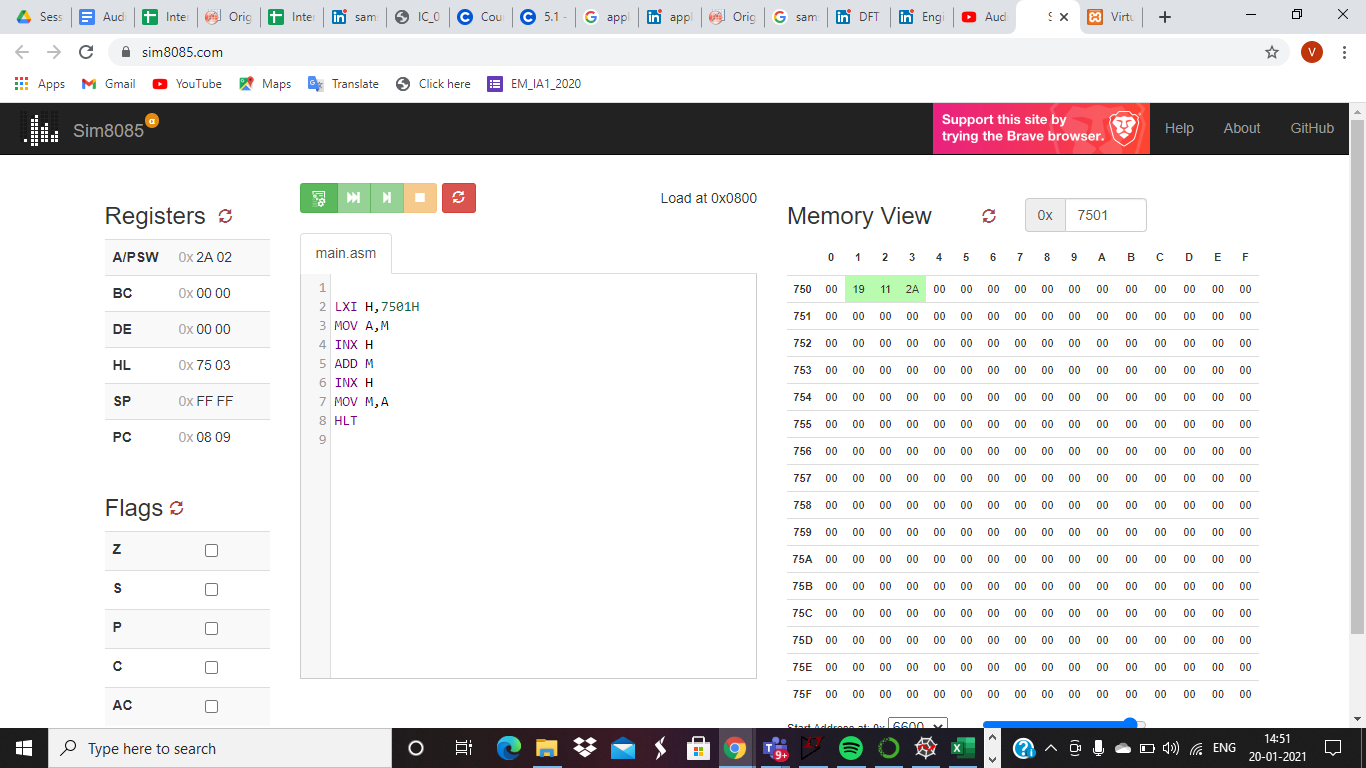
**Title:** Addition and Subtraction of two 8-bit and two 16-bit numbers

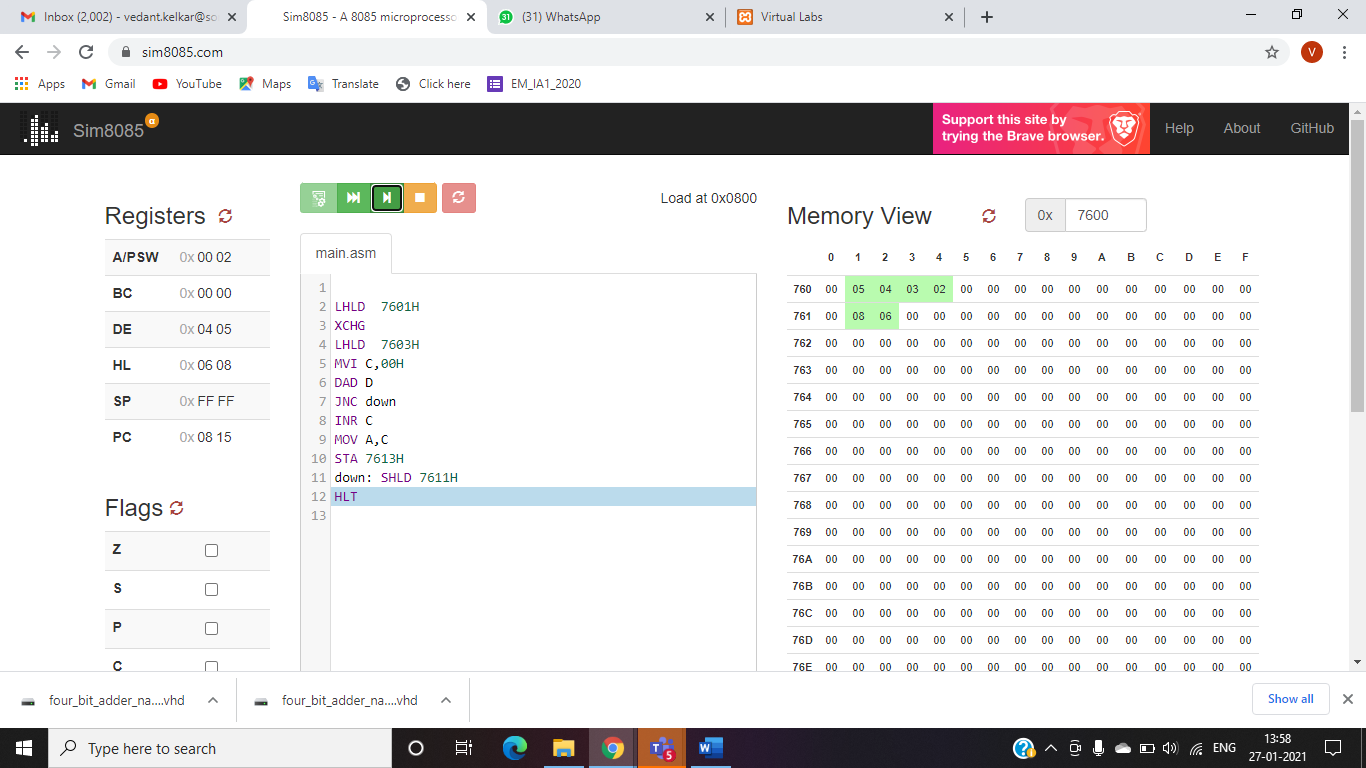
|  |
| --- |
| **Aim and Objective of the Experiment:** |
| **Aim:** Write an 8085 based ALP to   1. Add two 8 bit numbers 2. Add two 16 bit numbers with carry. 3. Subtract two 8 bit numbers with a display of borrow. 4. Subtract two 16 bit numbers with a display of borrow     **Objectives:**  To study basic instructions and addressing modes of 8085.  This experiment covers following instructions groups.   1. Data transfer 2. Arithmetic 3. Logical 4. Branch |

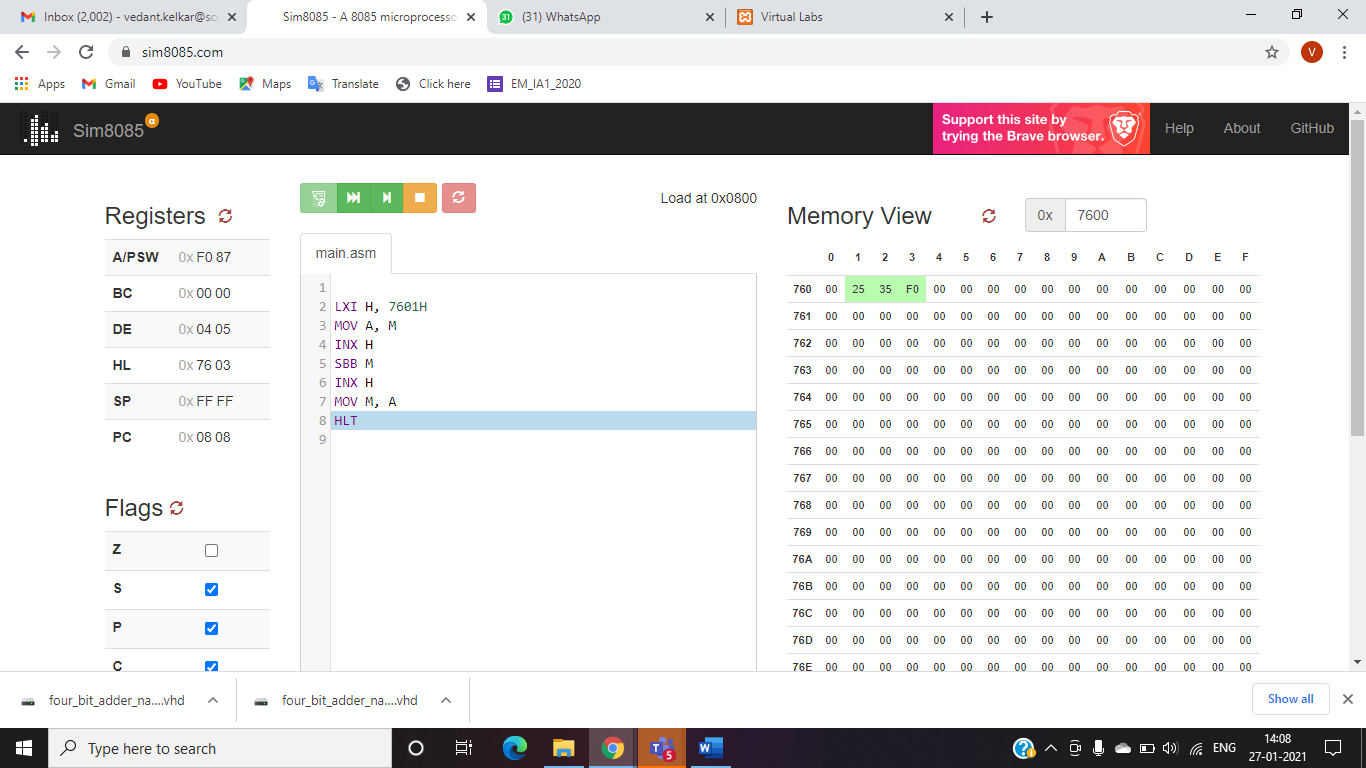
|  |
| --- |
| **COs to be achieved:** |
| **CO 1.** Describe basic operation of 8085 microprocessor system and explain its timing diagrams. |

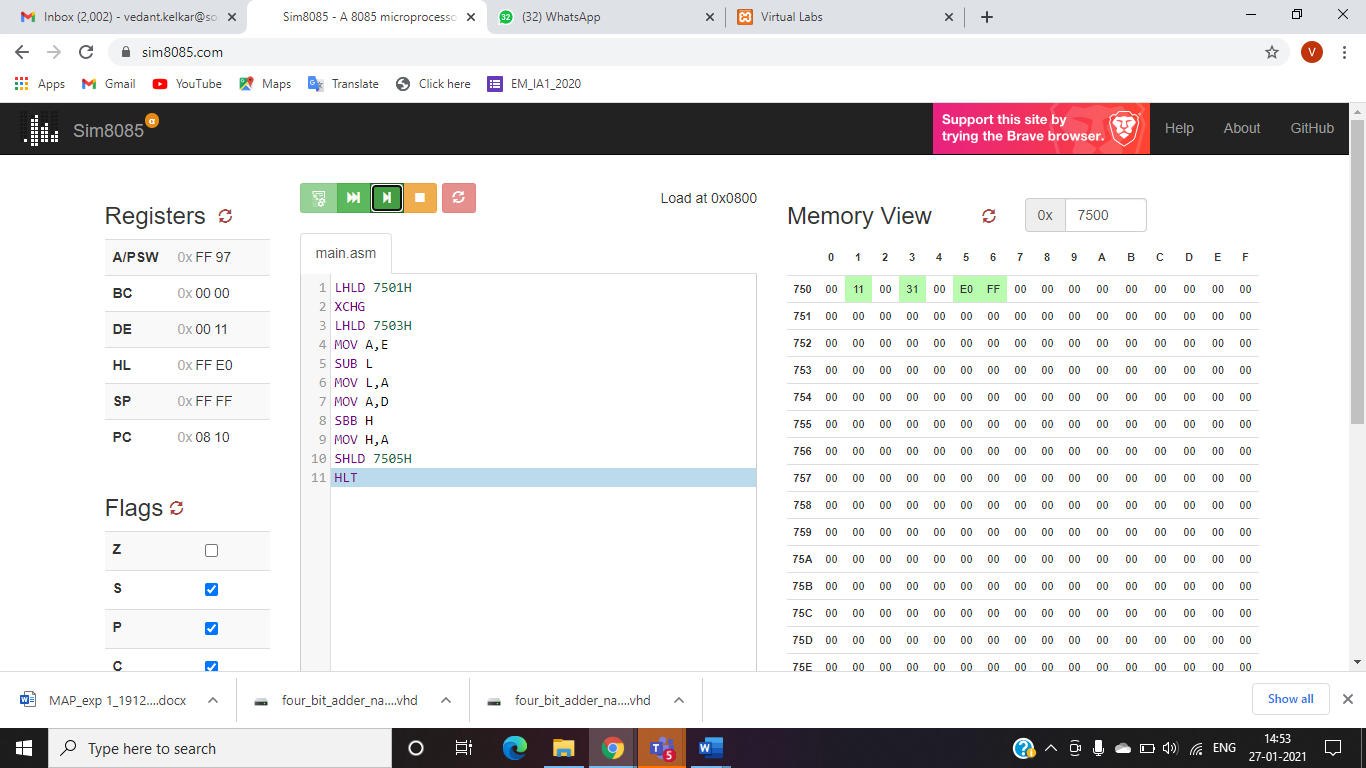
|  |
| --- |
| **Useful links** |
| Virtual Lab:  [http://vlabs.iitb.ac.in/vlabs-dev/labs\_local/microprocessor/labs/explist.php](%20http://vlabs.iitb.ac.in/vlabs-dev/labs_local/microprocessor/labs/explist.php)  Simulator:  <https://www.sim8085.com/> |

|  |
| --- |
| **Work to be done** |
| 1. Upload codes for addition of 8 bit and 16 bit addition and 8-bit and 16 bit subtraction and screenshots of virtual lab implementation. 2. Upload scanned image of handwritten algorithm/flowchart and code and results for post lab questions. |









|  |
| --- |
| **Post Lab Subjective/Objective type Questions:** |
| Q1. Write on 8085 based ALP to find 16 bit sum for an array of umbers.Assume length of the array in ML C020 and array actually starts from C021H. Store the 16-bit sum in Memory locations C030H and C031H      Q2. Write on 8085 based ALP to unpack a packed BCD number. Assume a packed BCD number in memory C500H, Disassemble the word into two nibbles. Store the lower unpacked BCD digit into the LSB position of C501H and the upper nibble (BCD digit) in the LSB postion of C502H    Q3. What are the different addressing modes of 8085? |
| * **Immediate addressing mode:**   In this mode, the source operand is always a data. Like LXI, MVI etc   * **Register addressing mode:**   The data that is to be operated is available in the register and the operand is also a register. For eg. MOV A,B: moves data stored in register B to register A   * **Register Indirect addressing mode:**   The data to be operated is available inside a memory location and that memory location is indirectly specified by a register pair. For eg. LDAX B: moves contents of B-C register to the accumulator   * **Direct addressing mode:**   The data to be operated is available inside a memory location and that memory location is directly specified as an operand. The operand is directly available in the instruction itself.  For eg. LHLD addr: loads the data from memory location to HL pair |

|  |
| --- |
| **Table to be used for Writing the code for Postlab Questions** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Address** | **Opcode** | **Label** | **Mnemonics** | **Comments** | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |

|  |
| --- |
| **Format for writing result for postlab questions** |
| |  |  |  |  | | --- | --- | --- | --- | | **Memory Location** | **Contents** | **Memory Location** | **Contents** | | Before execution | | After execution | | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |

**Conclusion:**

We achieved Addition and Subtraction of two 8-bit and two 16-bit numbers using simulator 8085.

**Signature of faculty in-charge with Date:**