**Experiment – 1**

**Problem statement**: Write an assembly language program to perform arithmetic operations.

**Algorithm:**

**Step 1**: Define the Base Register Address value during the program creation.

**Step 2**: Move the first operand in the General-Purpose Register R1.

**Step 3**: Move the second operand in the General-Purpose Register R2.

**Step 4**: Perform the arithmetic operation with the values in the registers.

**Step 5**: Result will be stored in the destination register.

**Step 6**: Store the resultant value in a data memory location.

**Step 7**: Terminate the program.

**Assembly Language Code:**

* **Addition**

**MOV #10, R01** //Store value of 10 in register R01

**MOV #5, R02** //Store value of 05 in register R02

**ADD R02, R01** //Add the register R01 and R02 values and store the resultant value in register R01

**STB R01, 00** //Store the resultant value of R01 in memory location 00

**HLT** //Stop the simulator

* **Subtraction**

**MOV #20, R03** //Store value of 20 in register R03

**MOV #15, R04** //Store value of 15 in register r04

**SUB R04, R03** //Subtract the register R03 and R04 values and store the resultant value in register R03

**STB R03, 08** //Store the resultant value of R03 in memory location 08

**HLT** //Stop the simulator

* **Multiplication**

**MOV #6, R05** //Store value of 06 in register R05

**MOV #3, R06** //Store value of 03 in register R06

**MUL R06, R05** //Multiplicate the register R05 and R06 values and store the resultant value in register R05

**STB R05, 16** //Store the resultant value of R05 in memory location 16

**HLT** //Stop the simulator

* **Division**

**MOV #8, R07** //Store value of 08 in register R07

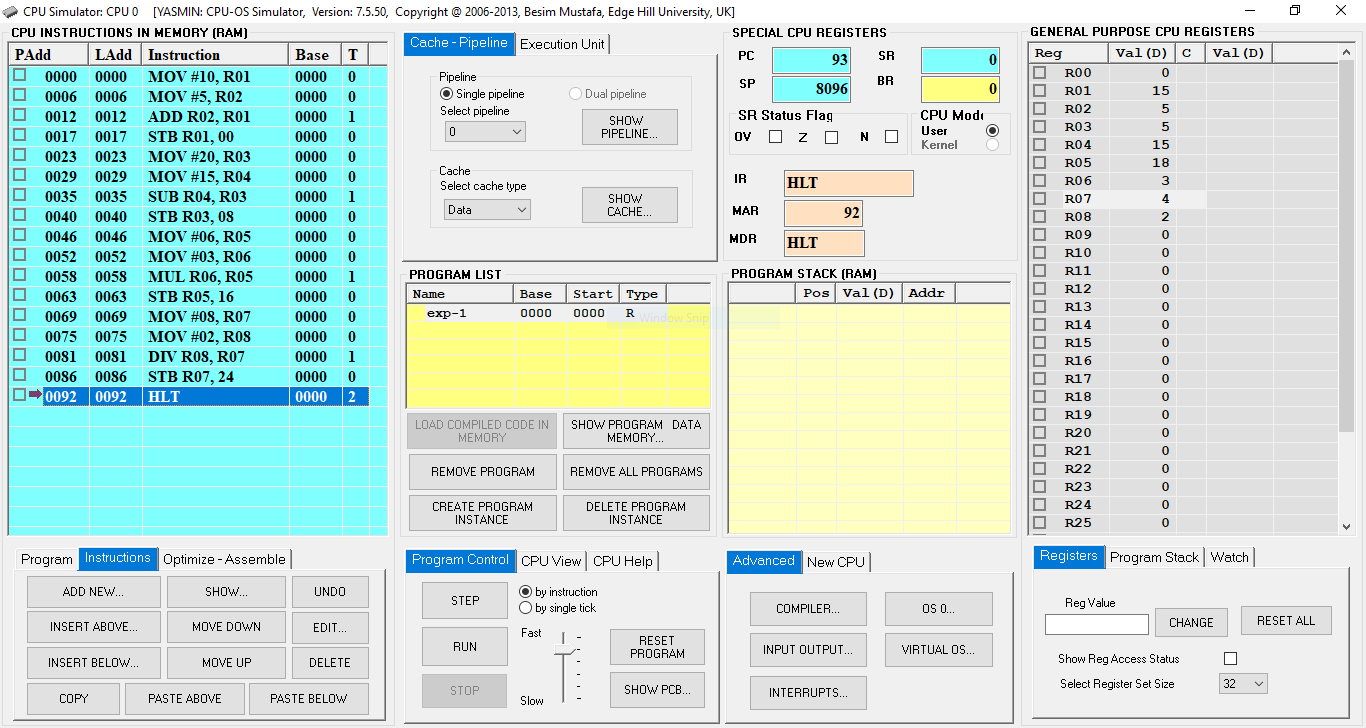
**MOV #2, R08** //Store value of 02 in register r08

**DIV R08, R07** //Divide the register R07 and R08 values and store the resultant value in register R07

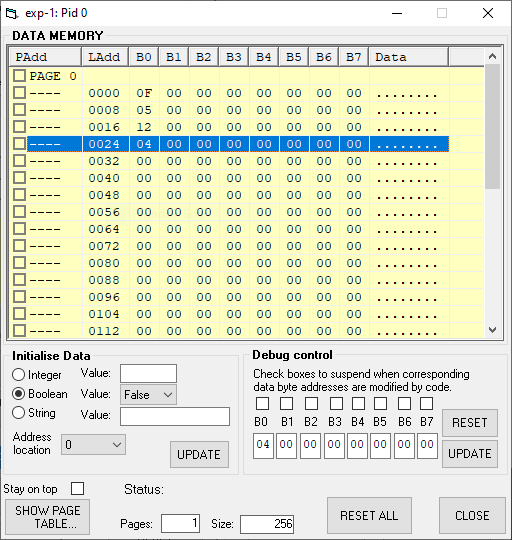
**STB R07, 24** //Store the resultant value of R07 in memory location 24

**HLT** //Stop the simulator

**Result:**



**Fig. 1**: CPU Simulator Window



**Fig. 2:** Data Memory Window

|  |  |
| --- | --- |
| **Step 01 - Addition Program Starts** | |
| PC | 6 |
| IR | MOV #10, R01 |
| MAR | 0 |
| MDR | MOV #10, R01 |
| R01 | 10 |
| **Step 02** | |
| PC | 12 |
| IR | MOV #5, R02 |
| MAR | 6 |
| MDR | MOV #5, R02 |
| R01 | 10 |
| R02 | 5 |
| **Step 03** | |
| PC | 17 |
| IR | ADD R02, R01 |
| MAR | 12 |
| MDR | ADD R02, R01 |
| R01 | 15 |
| R02 | 5 |
| **Step 04 - Addition Program Ends** | |
| PC | 23 |
| IR | STB R01,00 |
| MAR | 0 |
| MDR | 15 |
| R01 | 15 |
| R02 | 5 |
| 00 | 0F |
| **Step 05 - Subtraction Program Starts** | |
| PC | 29 |
| IR | MOV #20, R03 |
| MAR | 23 |
| MDR | MOV #20, R03 |
| R01 | 15 |
| R02 | 5 |
| R03 | 20 |
| 00 | 0F |
| **Step 06** | |
| PC | 35 |
| IR | MOV #15, R04 |
| MAR | 29 |
| MDR | MOV #15, R04 |
| R01 | 15 |
| R02 | 5 |
| R03 | 20 |
| R04 | 15 |
| 00 | 0F |
| **Step 07** | |
| PC | 40 |
| IR | SUB R04, R03 |
| MAR | 29 |
| MDR | SUB R04, R03 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| 00 | 0F |
| **Step 08 - Subtraction Program Ends** | |
| PC | 46 |
| IR | STB R03, 08 |
| MAR | 8 |
| MDR | 5 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| 00 | 0F |
| 08 | 05 |
| **Step 09 - Multiplication Program Starts** | |
| PC | 52 |
| IR | MOV #06, R05 |
| MAR | 46 |
| MDR | MOV #06, R05 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| R05 | 6 |
| 00 | 0F |
| 08 | 05 |
| **Step 10** | |
| PC | 58 |
| IR | MOV #03, R06 |
| MAR | 52 |
| MDR | MOV #03, R06 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| R05 | 6 |
| R06 | 3 |
| 00 | 0F |
| 08 | 05 |
| **Step 11** | |
| PC | 63 |
| IR | MUL R06, R05 |
| MAR | 58 |
| MDR | MUL R06, R05 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| R05 | 18 |
| R06 | 3 |
| 00 | 0F |
| 08 | 05 |
| **Step 12 - Multiplication Program Ends** | |
| PC | 69 |
| IR | STB R05, 16 |
| MAR | 16 |
| MDR | 18 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| R05 | 18 |
| R06 | 3 |
| 00 | 0F |
| 08 | 05 |
| 16 | 12 |
| **Step 13 - Division Program Starts** | |
| PC | 75 |
| IR | MOV #08, R07 |
| MAR | 69 |
| MDR | MOV #08, R07 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| R05 | 18 |
| R06 | 3 |
| R07 | 8 |
| 00 | 0F |
| 08 | 05 |
| 16 | 12 |
| **Step 14** | |
| PC | 81 |
| IR | MOV #02, R08 |
| MAR | 75 |
| MDR | MOV #02, R08 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| R05 | 18 |
| R06 | 3 |
| R07 | 8 |
| R08 | 2 |
| 00 | 0F |
| 08 | 05 |
| 16 | 12 |
| **Step 15** | |
| PC | 86 |
| IR | DIV R08, R07 |
| MAR | 81 |
| MDR | DIV R08, R07 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| R05 | 18 |
| R06 | 3 |
| R07 | 4 |
| R08 | 2 |
| 00 | 0F |
| 08 | 05 |
| 16 | 12 |
| **Step 16 - Division Program Ends** | |
| PC | 92 |
| IR | STB R07, 24 |
| MAR | 24 |
| MDR | 4 |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| R05 | 18 |
| R06 | 3 |
| R07 | 4 |
| R08 | 2 |
| 00 | 0F |
| 08 | 05 |
| 16 | 12 |
| 24 | 04 |
| **Step 17 - Simulation is Terminated** | |
| PC | 93 |
| IR | HLT |
| MAR | 92 |
| MDR | HLT |
| R01 | 15 |
| R02 | 5 |
| R03 | 5 |
| R04 | 15 |
| R05 | 18 |
| R06 | 3 |
| R07 | 4 |
| R08 | 2 |
| 00 | 0F |
| 08 | 05 |
| 16 | 12 |
| 24 | 04 |