**EE6361: Advanced concepts of VLSI**

**Assignment 2**

Problem Statement: Using Verilog code, define a simple Processor IIT6361-µP101 with following features:

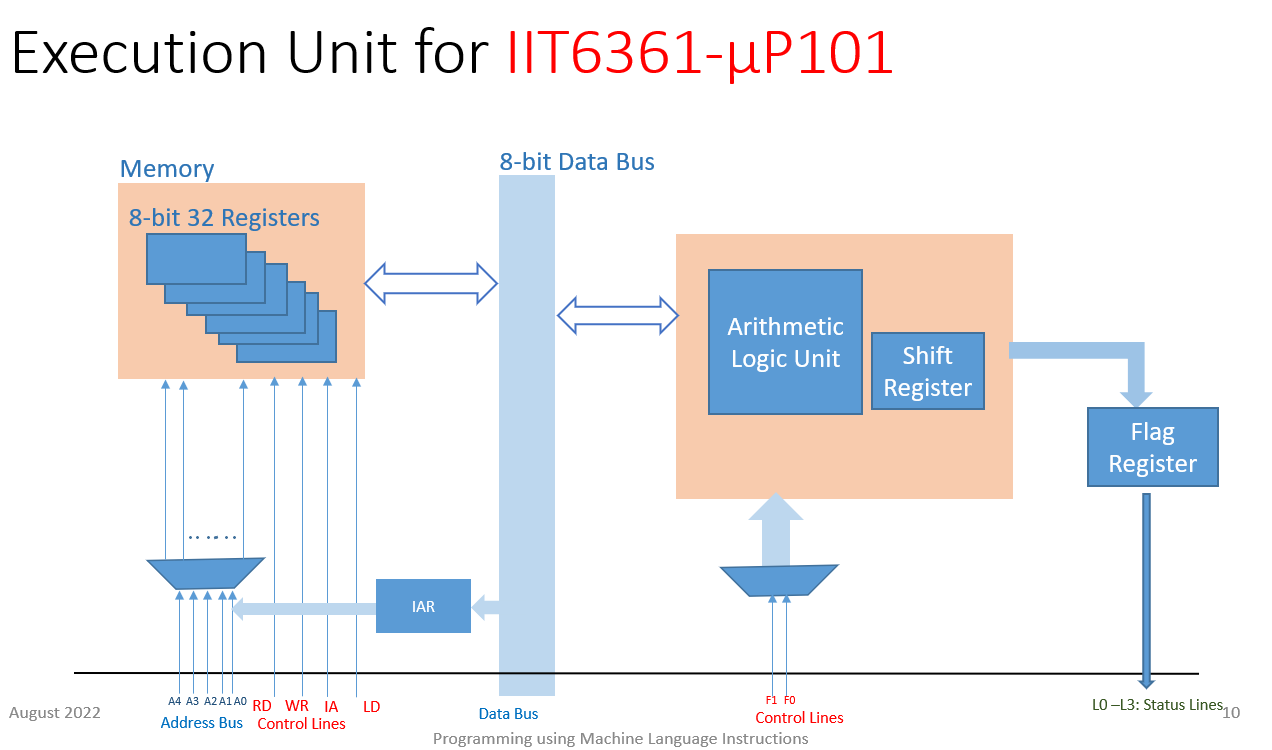
* Thirty-two 8 bit-Registers: R0-R31
  + 5 Address Lines (A4, A3, A2, A1, A0)
* 8 Bit Data Lines
* 4 Bit Flag Register
* 4 function ALU with
  + R0, R1 Mapped input for ADD and SUBTRACT
  + R4, R5 Mapped input for COMPARE: R4>R5
  + R0 Mapped input for INC
  + R2 Mapped Output
  + 2 ALU function select lines F1, F0

|  |  |
| --- | --- |
| **Function Select Line** | **Function** |
| 0 | 00 - ADD |
| 1 | 01 - SUBTRAT |
| 2 | 10 – COMPARE |
| 3 | 11 - INC |

Table 1: ALU Functions

|  |  |  |
| --- | --- | --- |
| **Bit number** | **Flag name** | **Description** |
| 0 | L0 | Carry/Borrow/Overflow/underflow |
| 1 | L1 | Shifted out bit |
| 2 | L2 | is set if any ALU operation results in Zero |
| 3 | L3 | 1 indicates result of compare is true, 0 otherwise |

Table 2: Flag Register



Note: Use the register memory built in assignment 1 for designing execution unit.

**Questions:**

Now use the Execution unit built to execute following instructions.

Write a testbench code for each of below problem statements and verify your simulation results.

Problem 1:

1. Load 141 to R4 check R4.
2. Load 208 to R6 check R6.
3. Load 32 to R8 check R8.

Problem 2:

1. Mov R4 to R5 check R5.
2. Mov R8 to R9 check R9.
3. Mov R6 to R7 check R7.

Problem 3: R4 + R6 => R10, check R10 and check if Carry.

Problem 4: R6 – R8 => R11, check R11and check if Barrow.

Problem 5: R8 – R4 => R12, check R12and check if Barrow.

Problem 6: INC R12 check R12.

Problem 7: R4 >= R6? check the flag.

Problem 8: R4 – R4 => R13, check R13 and check zero flag.

Problem 9: Add the numbers with carry using a Loop and store in R13.

R4 + R5 + R6 + R7 + R8 + R9 => R13 , check R13 with carry.

Problem 10: Find the largest number in R4, R5, R6, R8, R9, R10 and store in R13, check R13.