**Design of 16-Bit Processor in Verilog**

**(by Ratan Abhinav Sharma)**

**DESIGN DESCRIPTION**

Implementation of 16-Bit simple instruction set Processor supporting 27 instructions including Arithmetic, Logical, Load & Store, Jump and Halt instructions in Verilog using Xilinx Vivado application.

**DESIGN ARCHITECTURE**

Program Memory

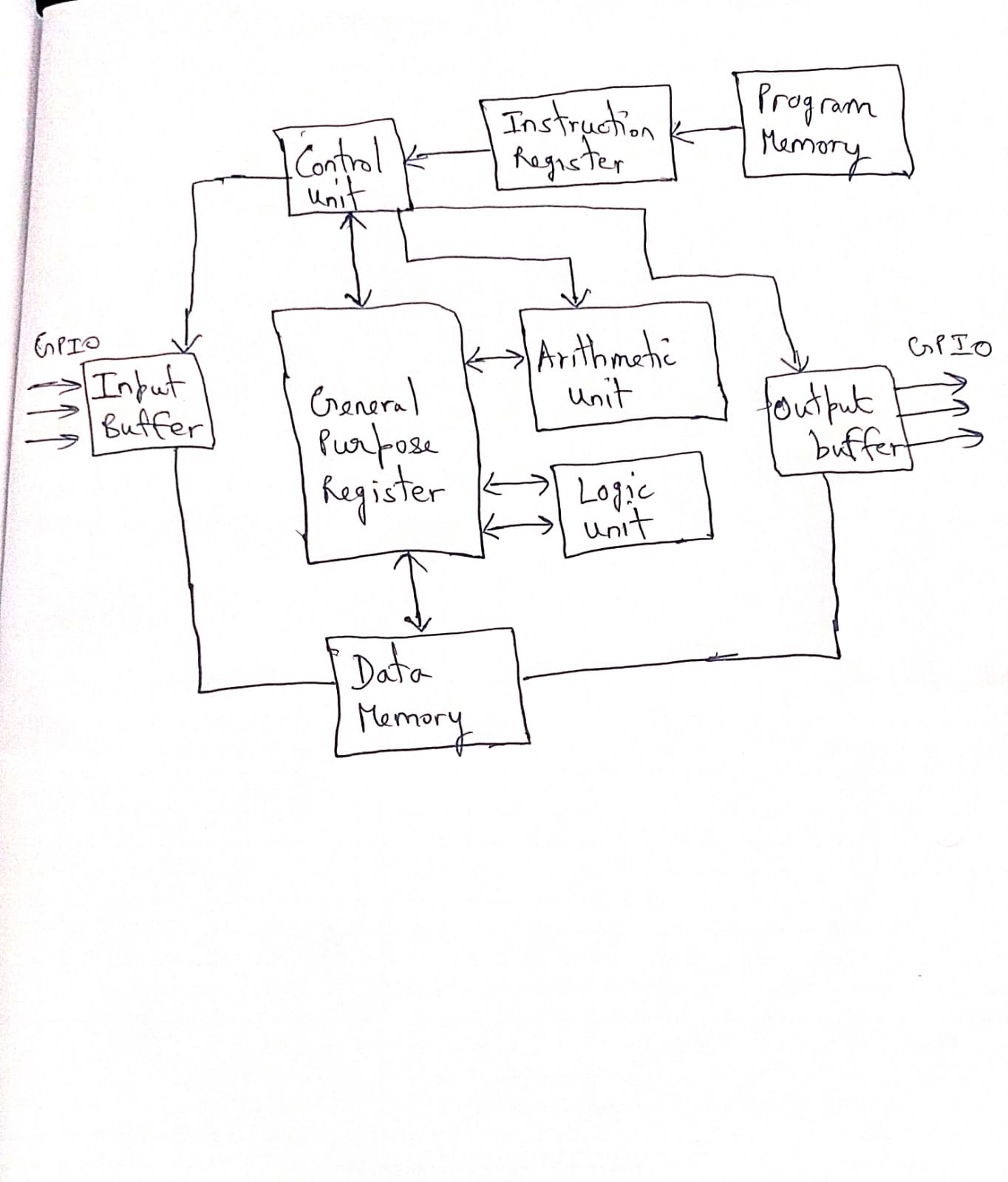
32-Bit Instruction Register

32 16-Bit General Purpose Registers

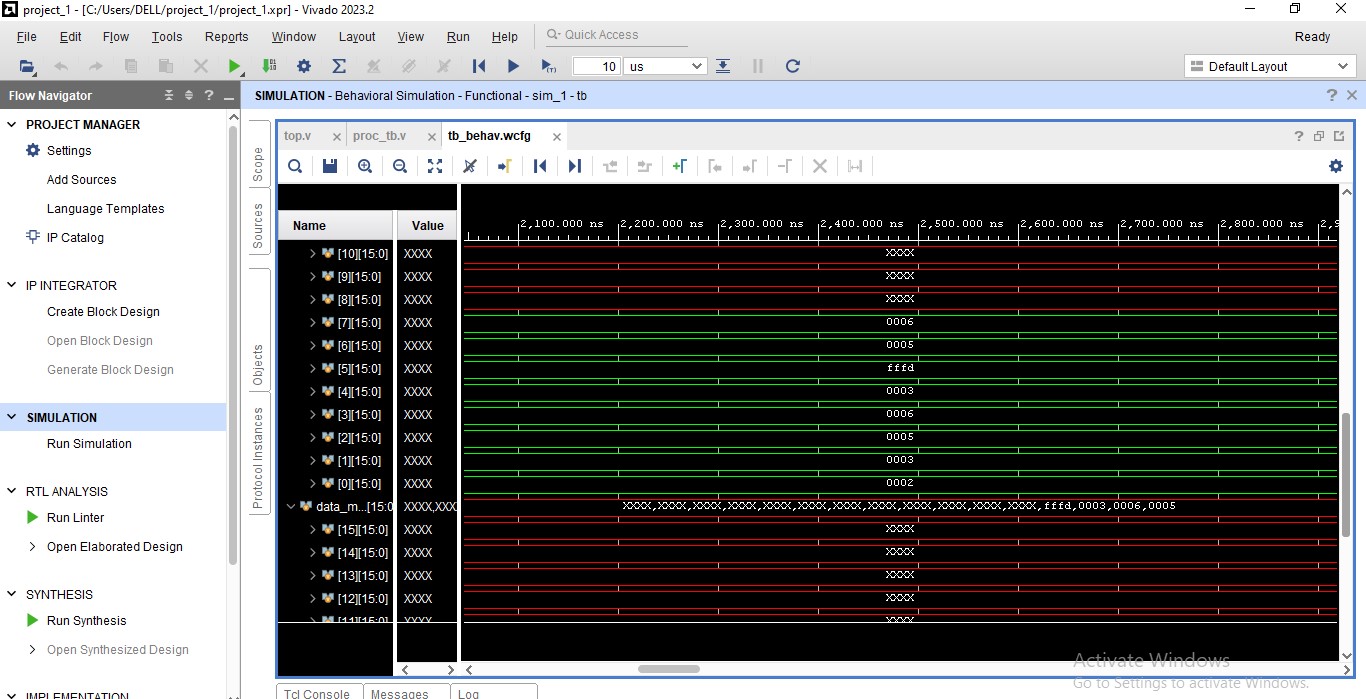
Data Memory (16-Bit)

Arithmetic & Logical Unit (ALU)

Input & Output buffers din and dout (16-Bit)

****

**Simulation Results for Arithmetic, Logical and Store & Load Inst.**



MOV R0, #2

MOV R1, #3;

ADD R2, R0, R1;

MUL R3, R0, R1;

ROR R4, R0, R1;

RNAND R5, R0, R1;

STOREREG 0, R2;

STOREREG 1, R3;

STOREREG 2, R4;

STOREREG 3, R5;

SENDREG R6, 0;

SENDREG R7, 1;

**Corresponding OpCode in Program Memory:**

00001\_00000\_00000\_1\_0000\_0000\_0000\_0010

00001\_00001\_00000\_1\_0000\_0000\_0000\_0011

00010\_00010\_00000\_0\_0000\_1000\_0000\_0000

00100\_00011\_00000\_0\_0000\_1000\_0000\_0000

00101\_00100\_00000\_0\_0000\_1000\_0000\_0000

01001\_00101\_00000\_0\_0000\_1000\_0000\_0000

01101\_00000\_00010\_0\_0000\_0000\_0000\_0000

01101\_00000\_00011\_0\_0000\_0000\_0000\_0001

01101\_00000\_00100\_0\_0000\_0000\_0000\_0010

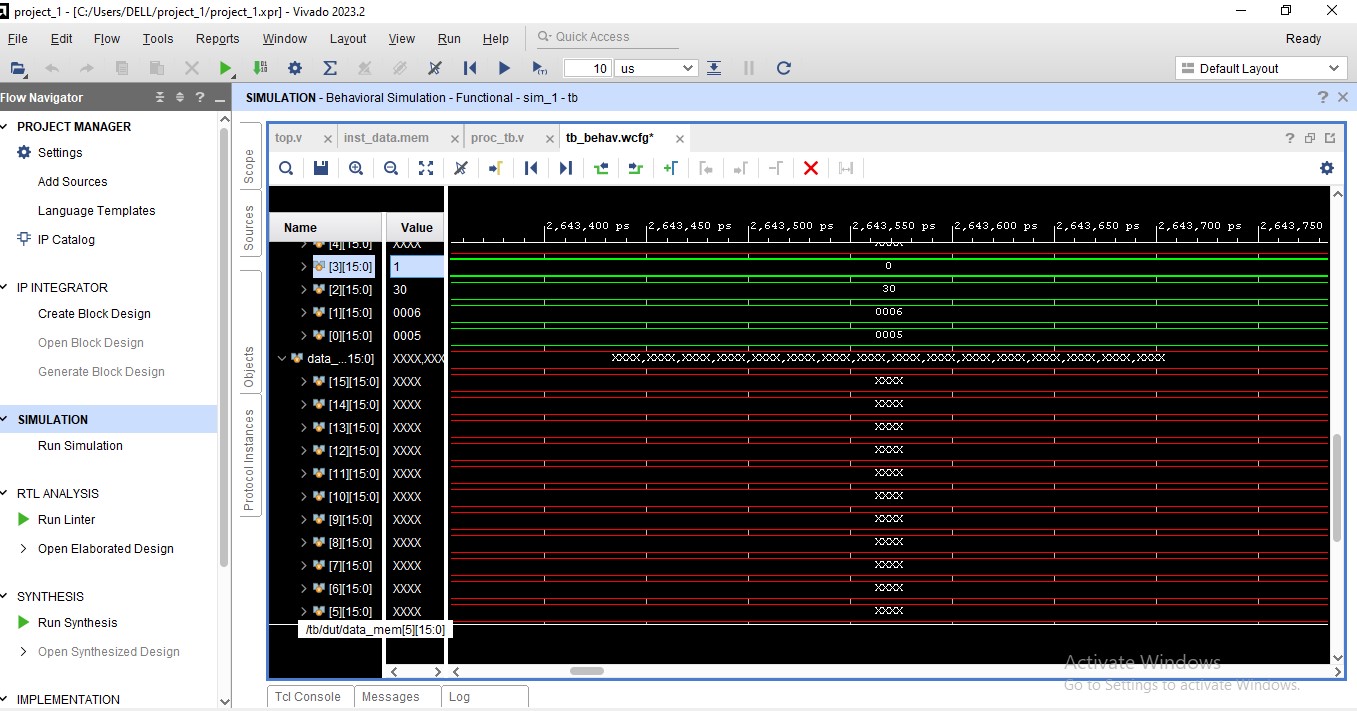
01101\_00000\_00101\_0\_0000\_0000\_0000\_0011

10001\_00110\_00000\_0\_0000\_0000\_0000\_0000

10001\_00111\_00000\_0\_0000\_0000\_0000\_0001

**SIMULATION RESULTS FOR JUMP INSTRUCTIONS**

**(Multiplication of 5 and 6 without MUL inst)**



MOV R0, #5;

MOV R1, #6;

MOV R2, #0;

MOV R3, #6;

ADD R2, R2, R0;

SUB R3, R3, #1;

JNZ @ 4;

HALT;

**CORRESPONDING OPCODE IN PROGRAM MEMORY**

00001\_00000\_00000\_1\_0000\_0000\_0000\_0101

00001\_00001\_00000\_1\_0000\_0000\_0000\_0110

00001\_00010\_00000\_1\_0000\_0000\_0000\_0000

00001\_00011\_00000\_1\_0000\_0000\_0000\_0110

00010\_00010\_00010\_0\_0000\_0000\_0000\_0000

00011\_00011\_00011\_1\_0000\_0000\_0000\_0001

11000\_00000\_00000\_0\_0000\_0000\_0000\_0100

11011\_00000\_00000\_0\_0000\_0000\_0000\_0000

**TESTBENCH CODE**

