***LAB 3 DEMO* – Working with an SRAM, Reg File, ALU, I/O, and C Stuff Cont….**

Team _____ ***Reviewer***_____

Designing an ALU (70)

Design

ALU modeled first in behavioural Verilog then as dataflow model.

Comparison between two models using Netlist Viewer / RTL Viewer

Sixteen-bit data from SRAM sign extended to 32 bits for ALU

Arithmetic

1. Addition
   Implemented as carry look-ahead to meet timing constraints
2. Subtraction – 2's complement

Logical

1. AND
2. OR
3. XOR
4. SLT
5. SLL
   Implemented as barrel shifter - Selectable shift 0..3 positions

Flags

Z, V, C, N

Working with the ALU, SRAM, and Register File (80)

Setup

1. Load test ALU instructions into SRAM.
2. Load test data into SRAM.

Run

Configure signal tap to trigger on command and display following transactions:

On command…

1. Transfer data block from SRAM to register file.
2. Read and interpret ALU instruction from SRAM.
3. Execute ALU instruction and set proper flags.
4. Store result into register file.

Timing and Execution

Operations completed in single clock cycle

Verified using signal tap

A = 25
B = 16
C = 7
D = 4
E = 9

result = ((A – B)*(C+D))/E

Implemented using pointers to all variables

Fully functional