**Arab Academy for Science and Technology**

**College of Language and Communication**

**Logo

Description automatically generated**

Computer Architecture

Single Cycle

(Report)

**Submitted by:**

Shady Mohamed Tarek

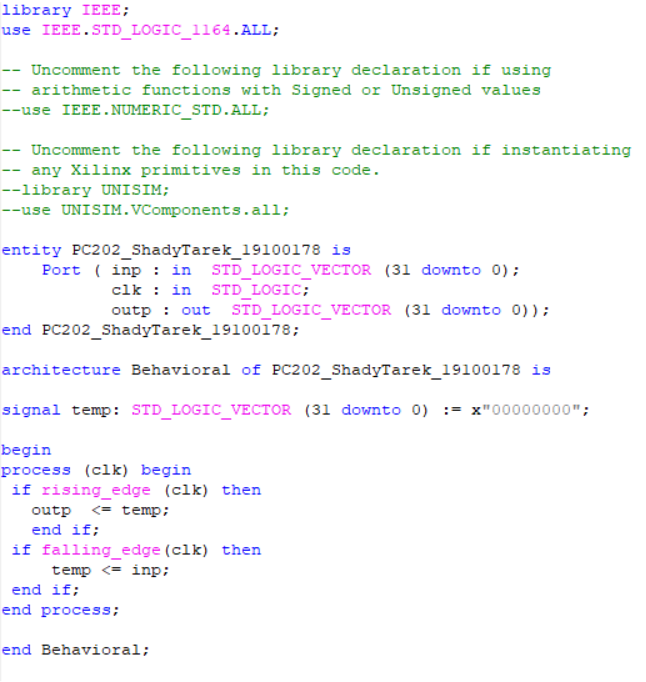
**Department:** computer engineering

**Reg #:** 19100178

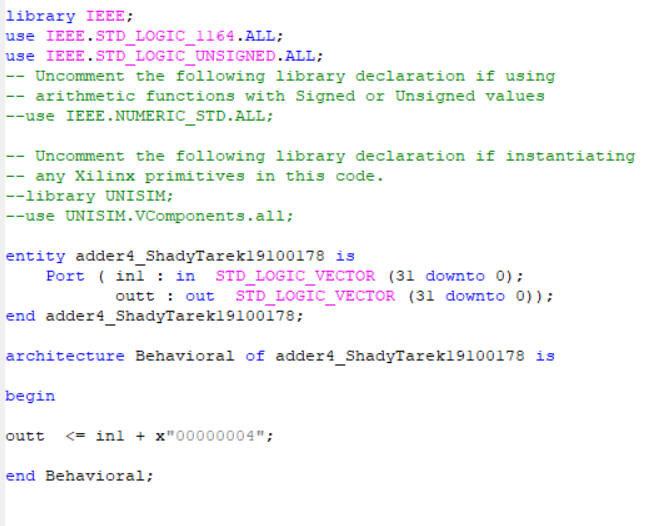
**Submitted to:** Dr. Marwa El Shenawy

Sources:

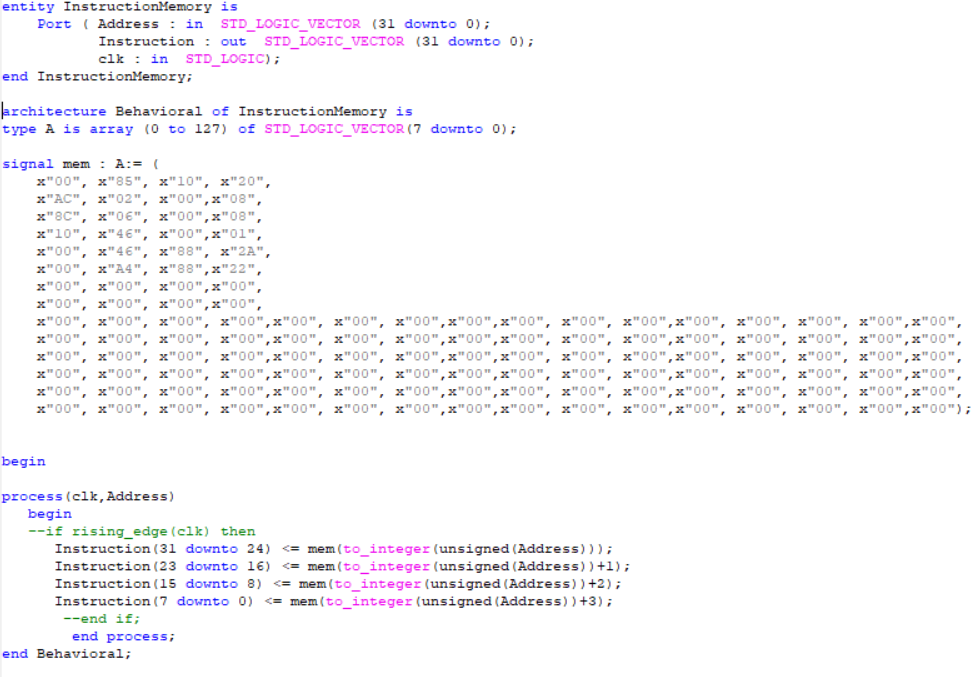
1-Program counter:



2-PC Adder:



3-Instruction memory:



**Corresponding Object Code**

**In Hexadecimal**

**1-00 85 10 20**

**2-AC 02 00 08**

**3-8C 06 00 08**

**4-10 46 00 01**

**5-00 46 882A**

**6-00 A4 88 22**

**Tested Instructions**

**1-add $v0, $a0, $a1**

**2-sw $v0, 8($zero)**

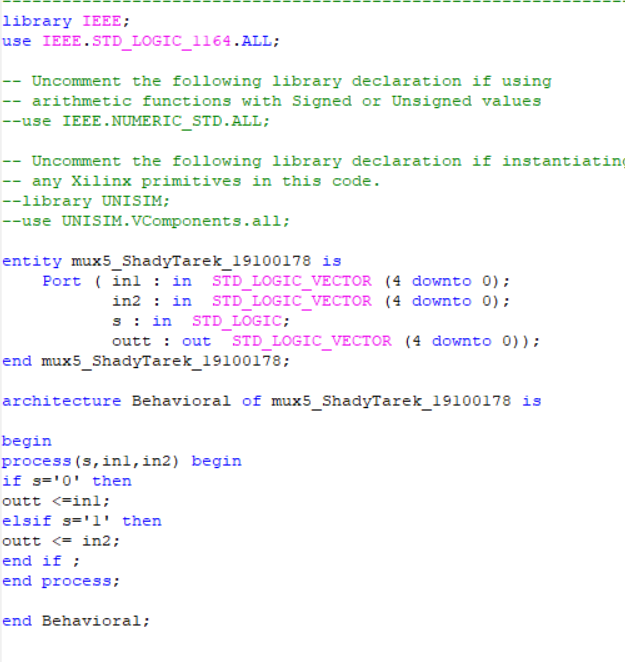
**3-lw $a2, 8($zero)**

**4-beq $v0, $a2, Good\_Processor**

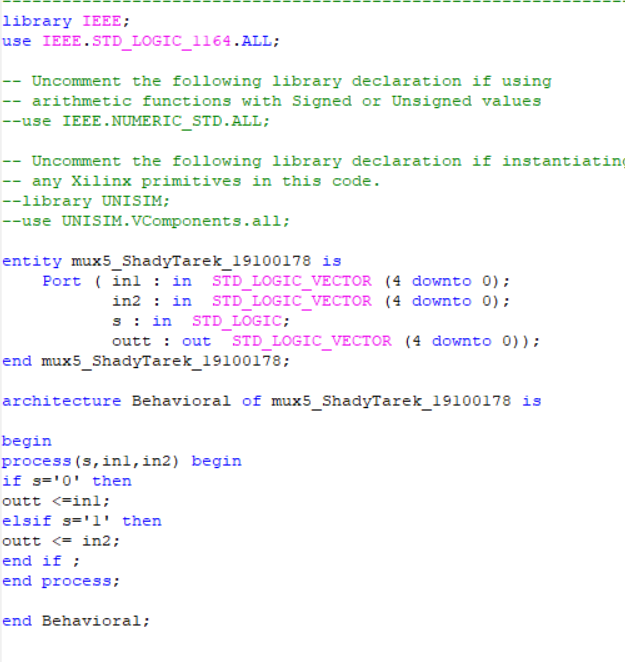
**5-slt $s1, $v0, $a2**

**6- Good\_Processor: sub $s1, $a1, $a0**

4-(2x1) 5-bit MUX:



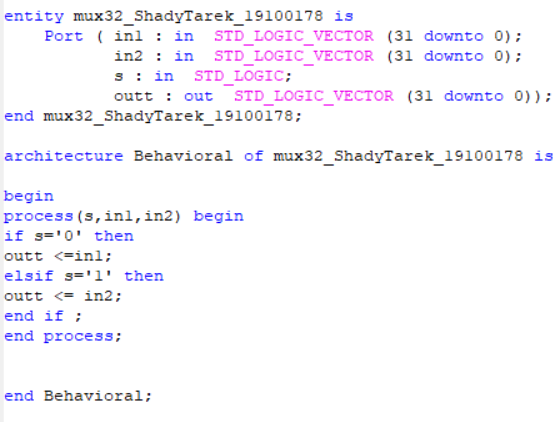
5-PC Mux (2x1):



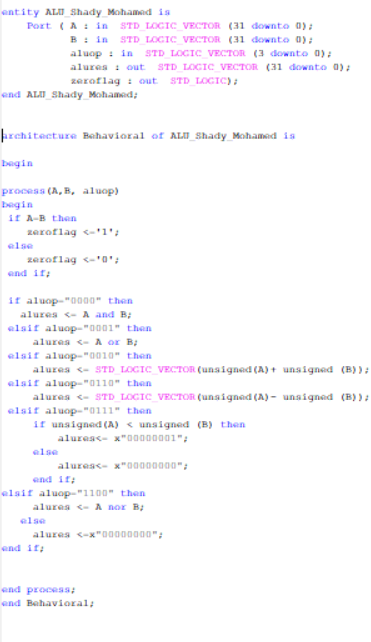
6-Registers :



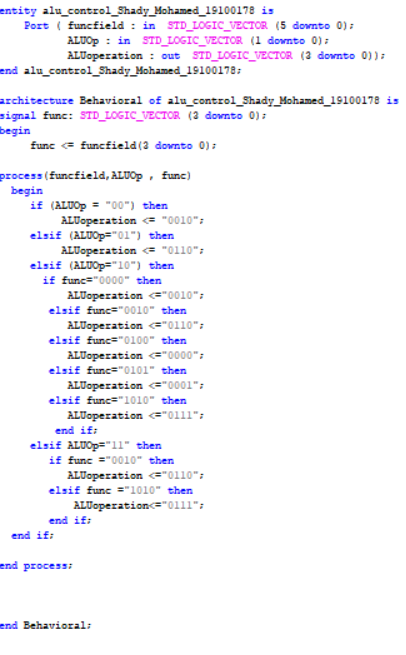
7-Mux 32-bit (2x1):



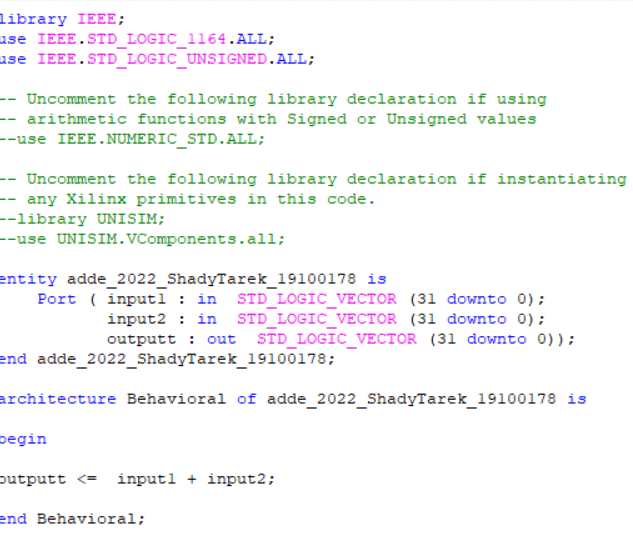
8-ALU:



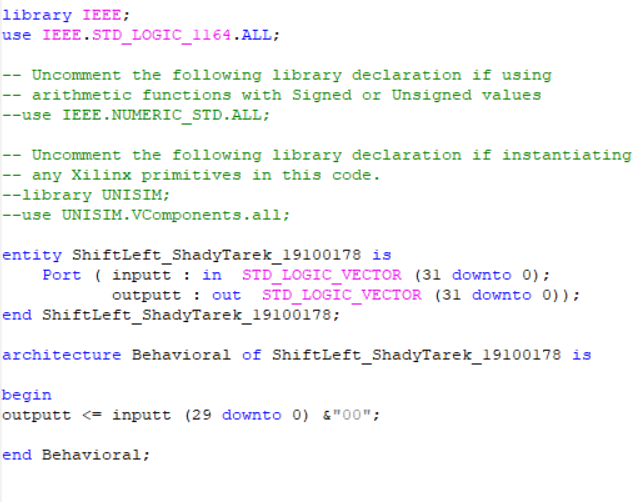
9-ALU control:



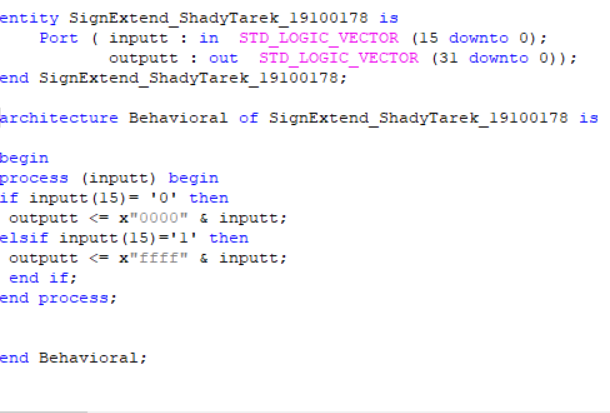
10-Adder:



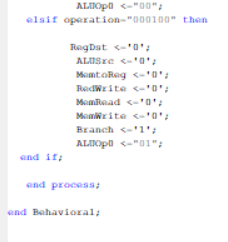
11-Shift left:

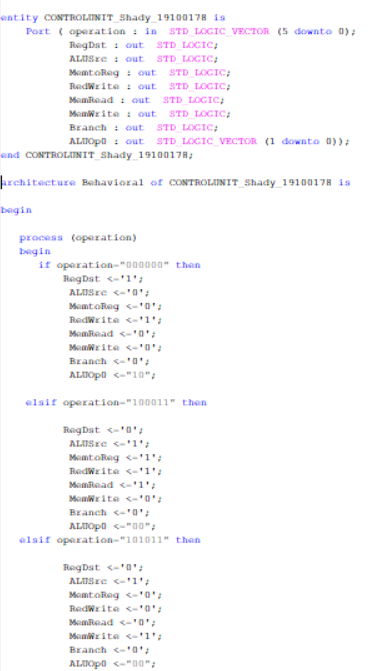


12-Sign Extend:

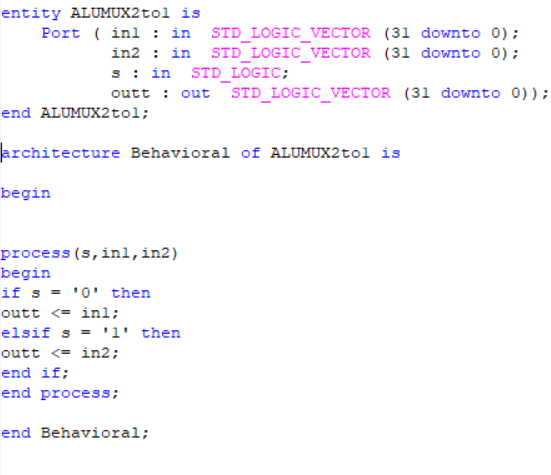


13-Control Unit:

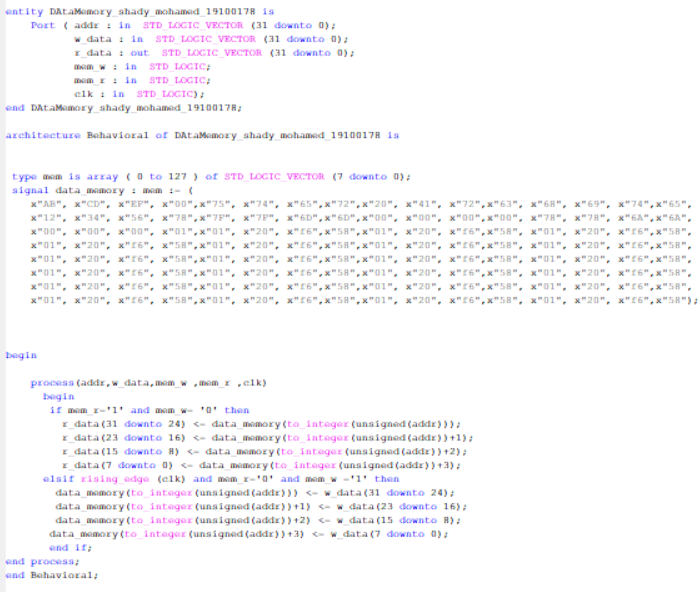




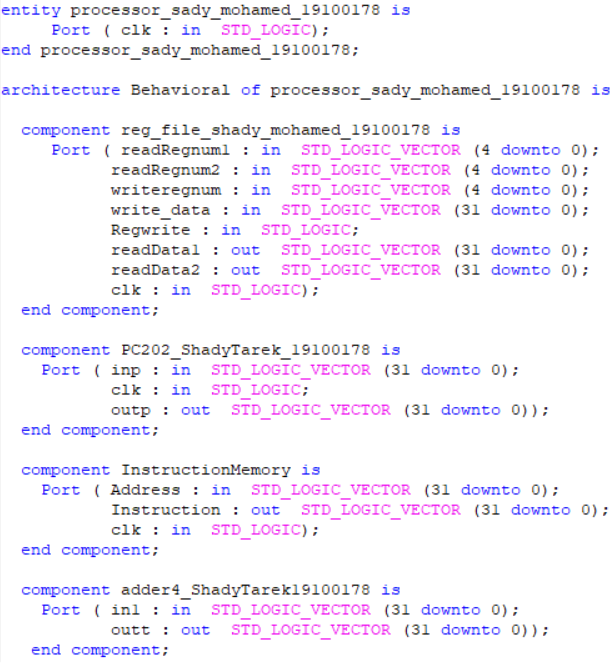
14-ALU Mux (2x1):

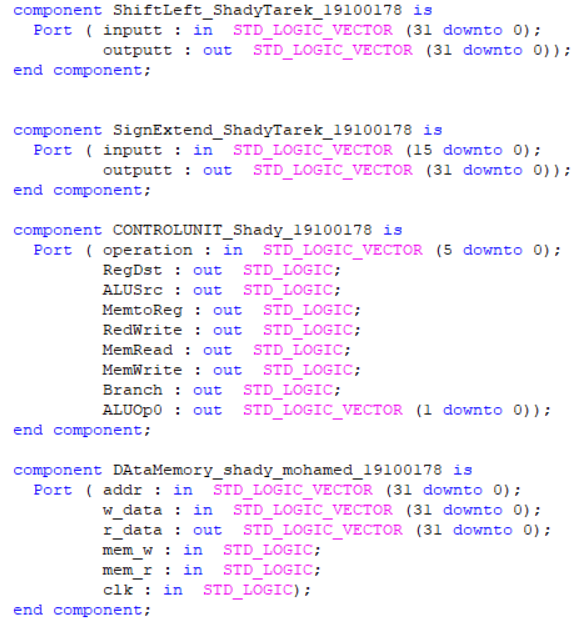


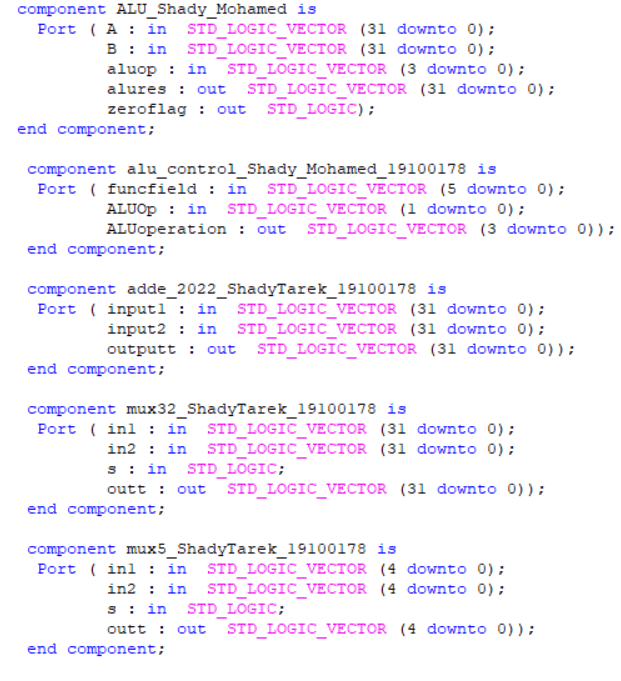
15-Data Memory:

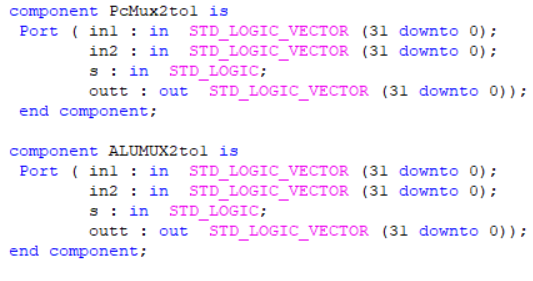


Components:

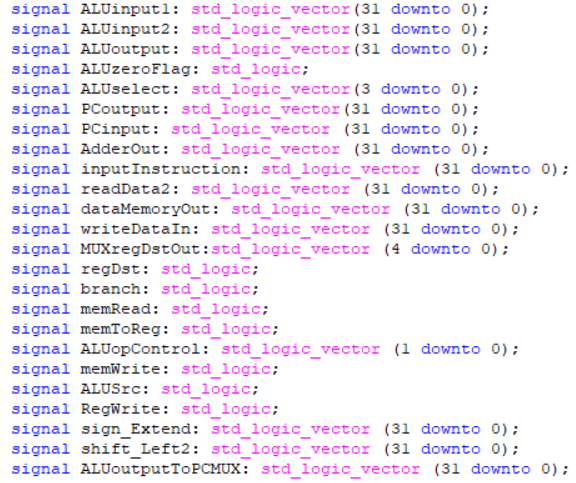




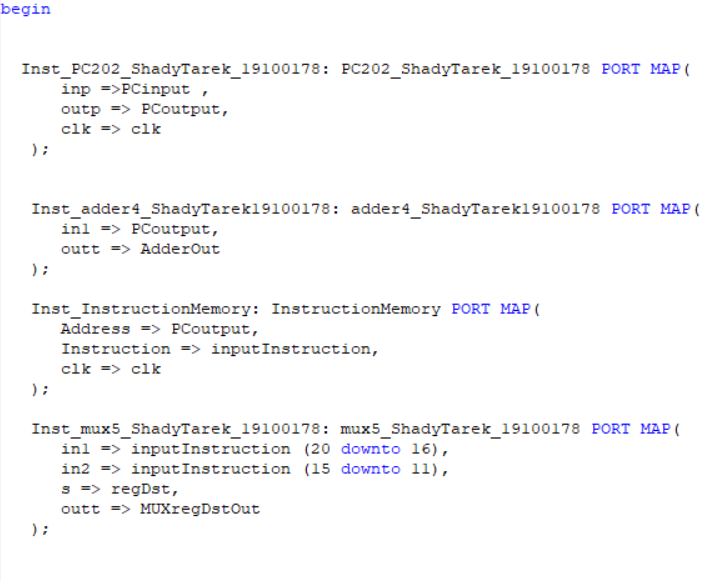


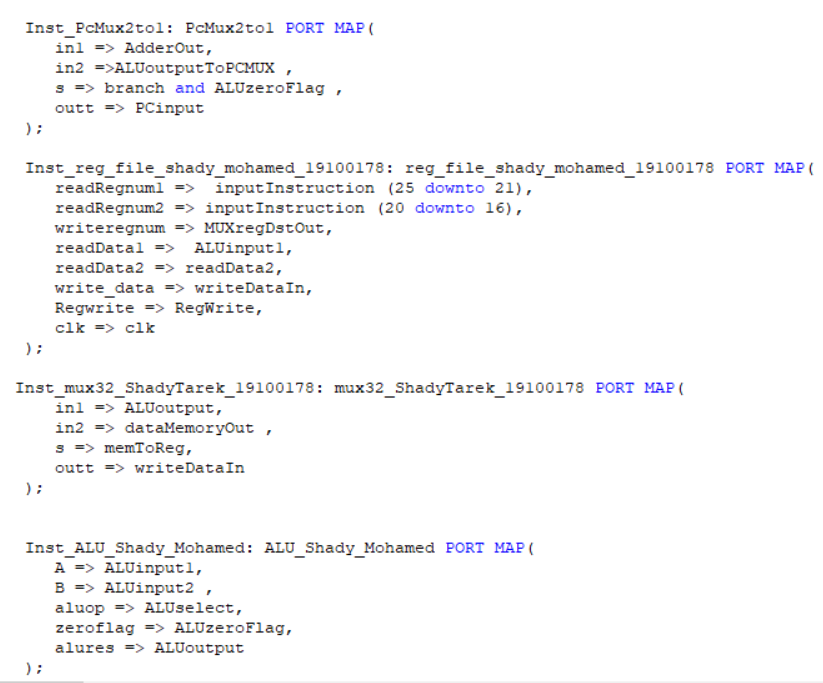


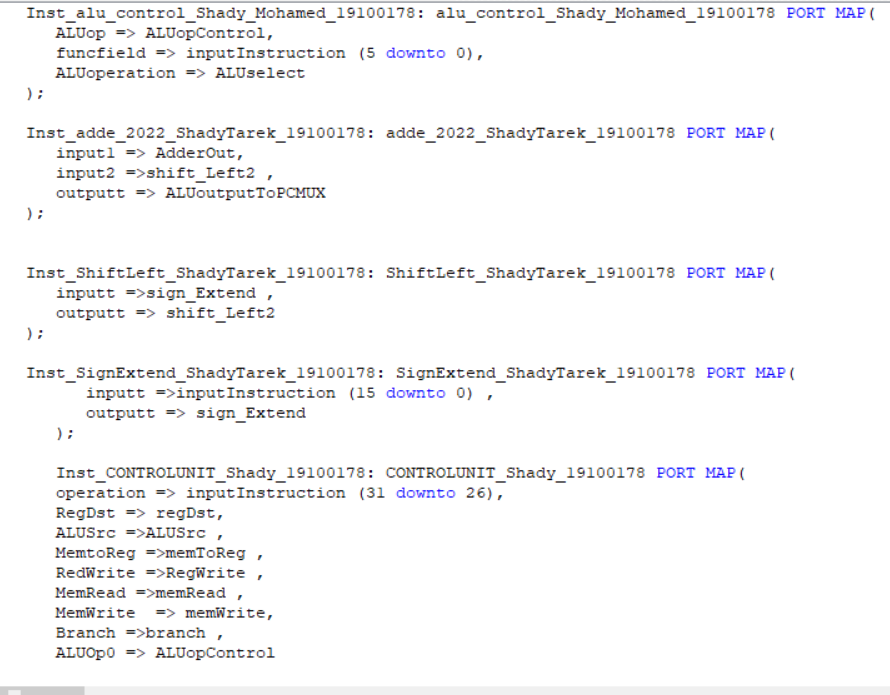
Signals used:

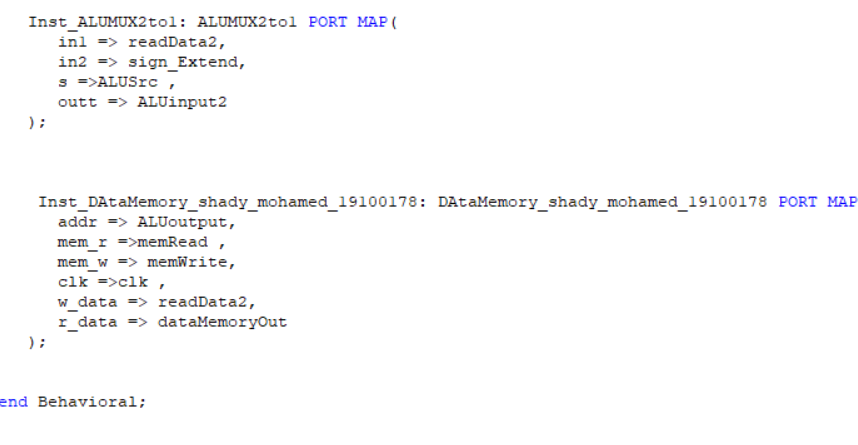


Port Mapping:



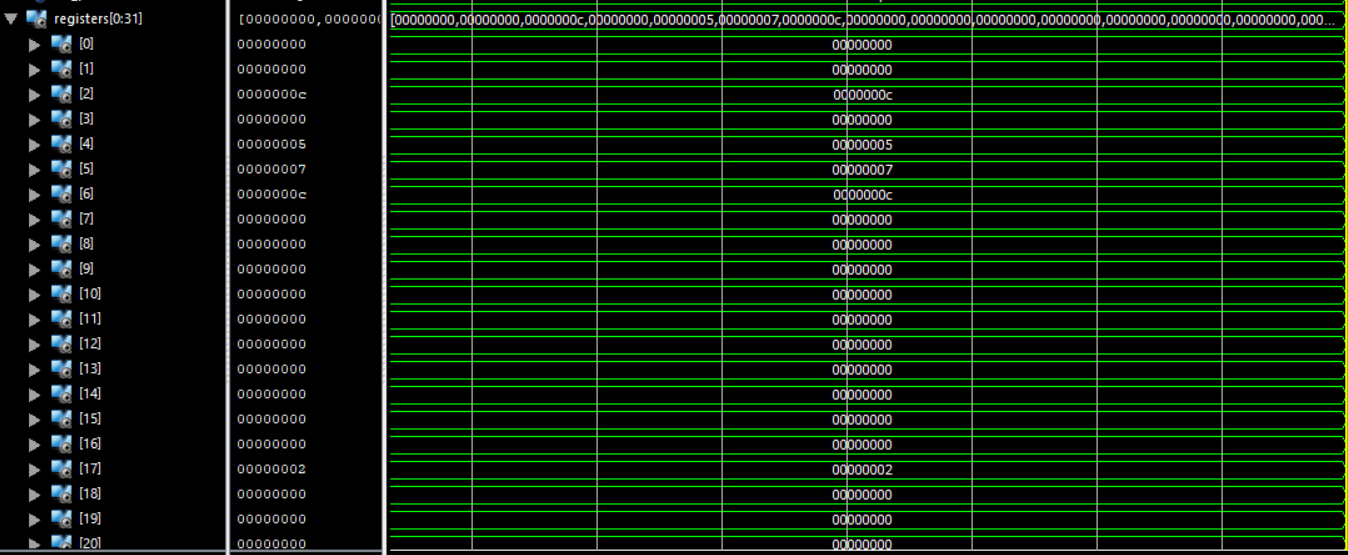






Simulation:

1-Registers:



2-Instruction memory:

Tested instructions:

add $v0, $a0, $a1

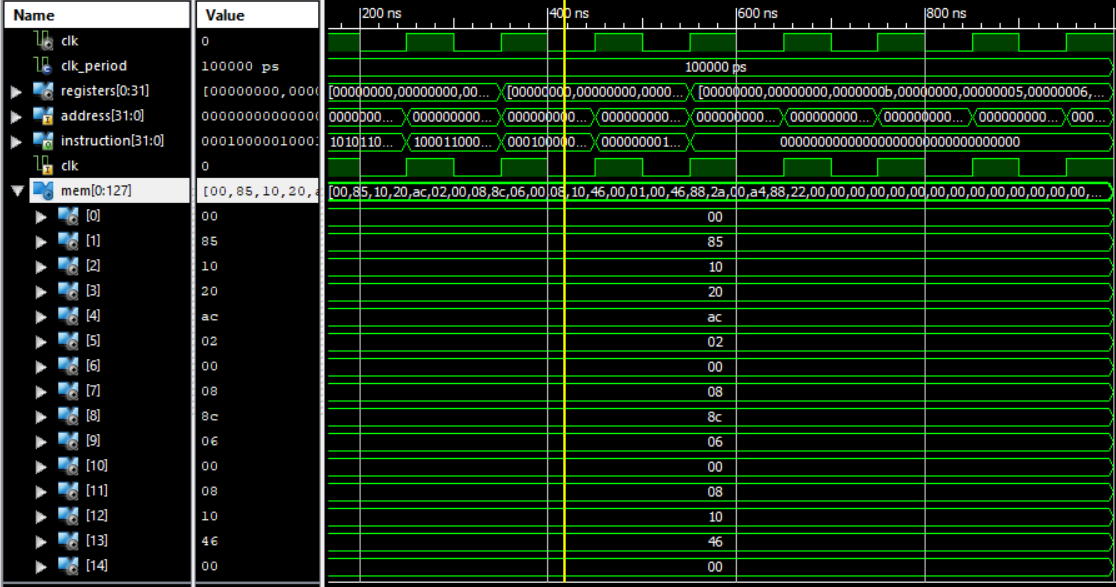
sw $v0, 8($zero)

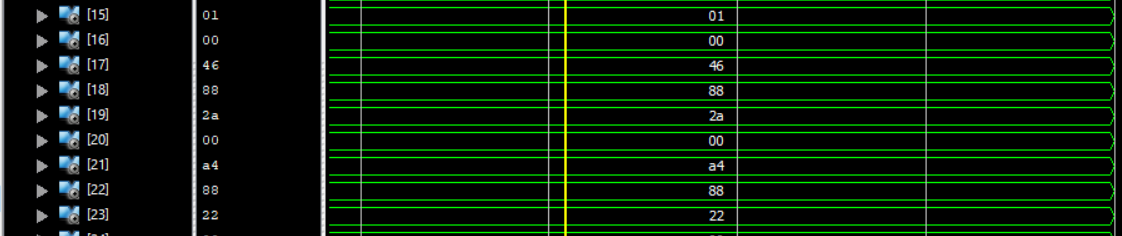
lw $a2, 8($zero)

beq $v0, $a2, Good\_Processor

slt $s1, $v0, $a2

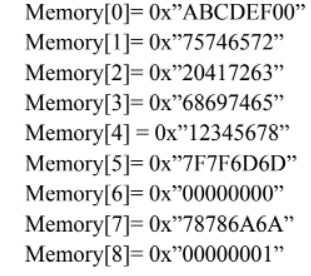
Good\_Processor: sub $s1, $a1, $a0

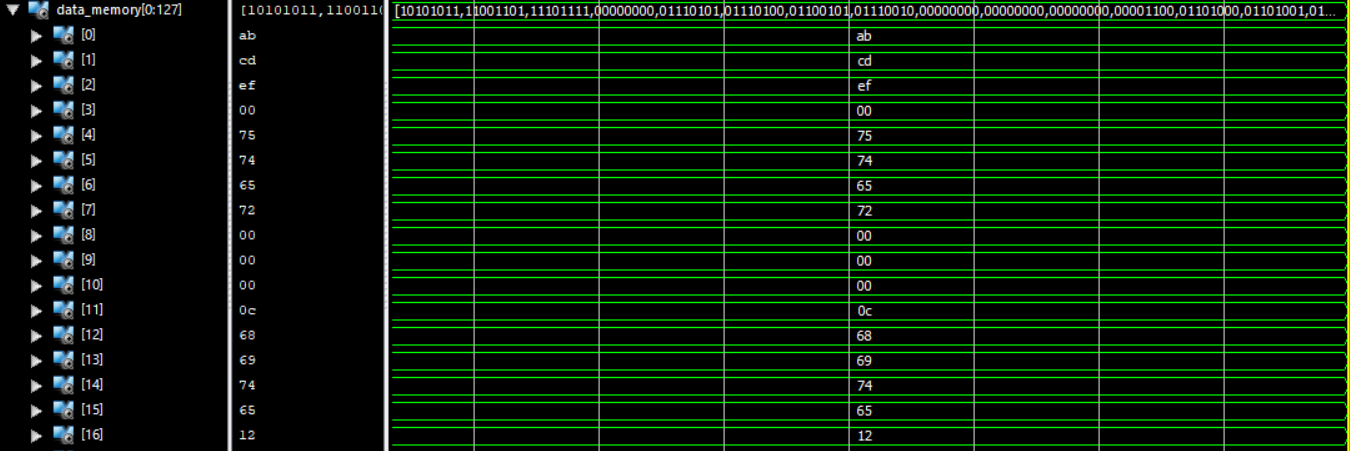


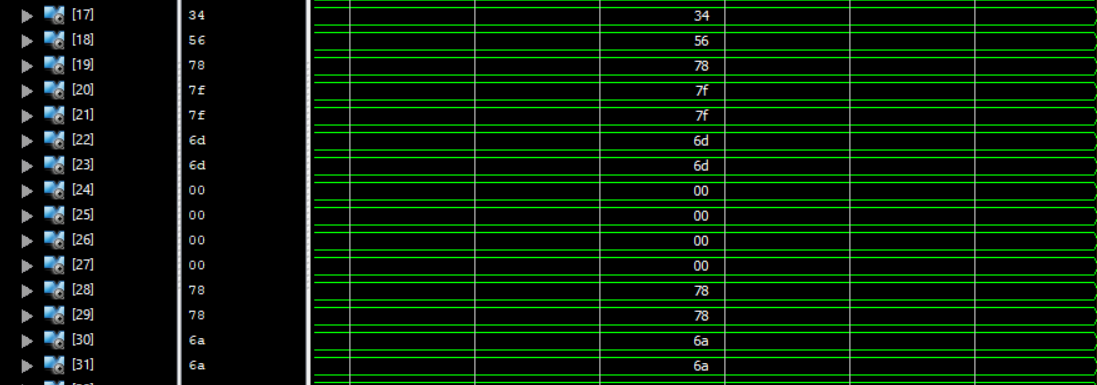


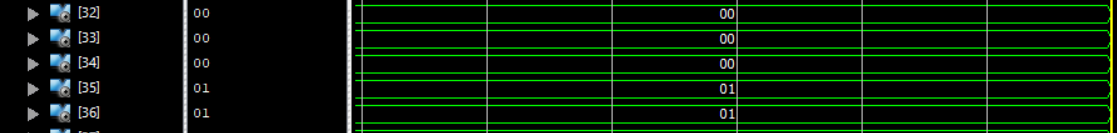
3-Data Memory:

Data in Data memory









So to explain what happened data in registers a0 and a1 (4 & 5) were added and saved in register v0 (2) then data in v0 was loaded from register to memory at address 8. After that data is load from memory to register a2 (6). Furthermore, beq (Branch on equal) is used to see if data in register v0 is equal to data is register a2 and jump to label Good\_processor and execute it. Subtraction occurs between data in a1 and data in a0 and answer stored in register s1 (17). On the other hand, if data in v0is not equal data in a2 slt (Set less than) instruction will be executed if data in v0 is less than data in a2 set s1 equal to 1 else equal to 0.