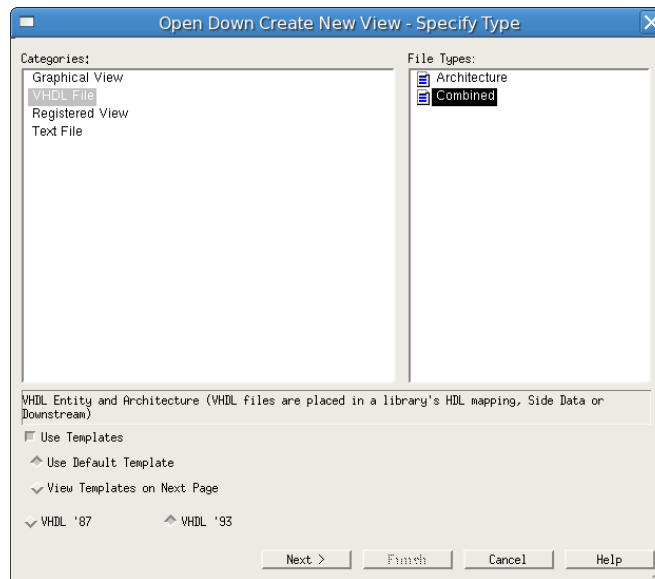


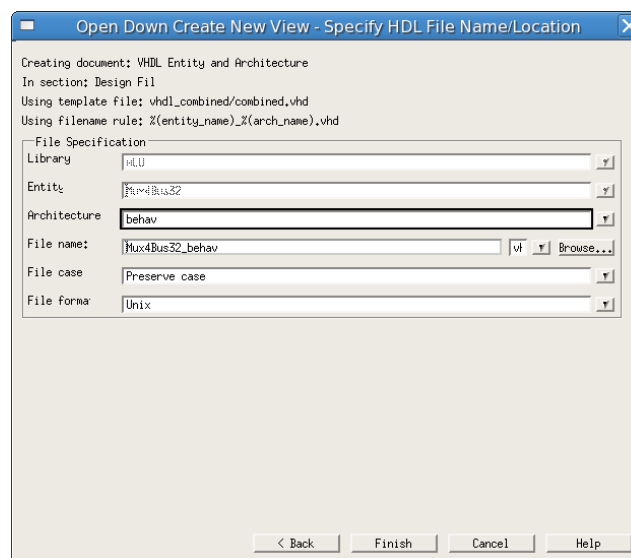
Creating the Mux4Bus32 Sub-Block

In the *Logical* sub-block, we accomplished this by placing an embedded block with a VHDL statement to multiplex the signals. For this sub-block, we will use similar VHDL statement, but will enter it directly as a text VHDL Architecture description.

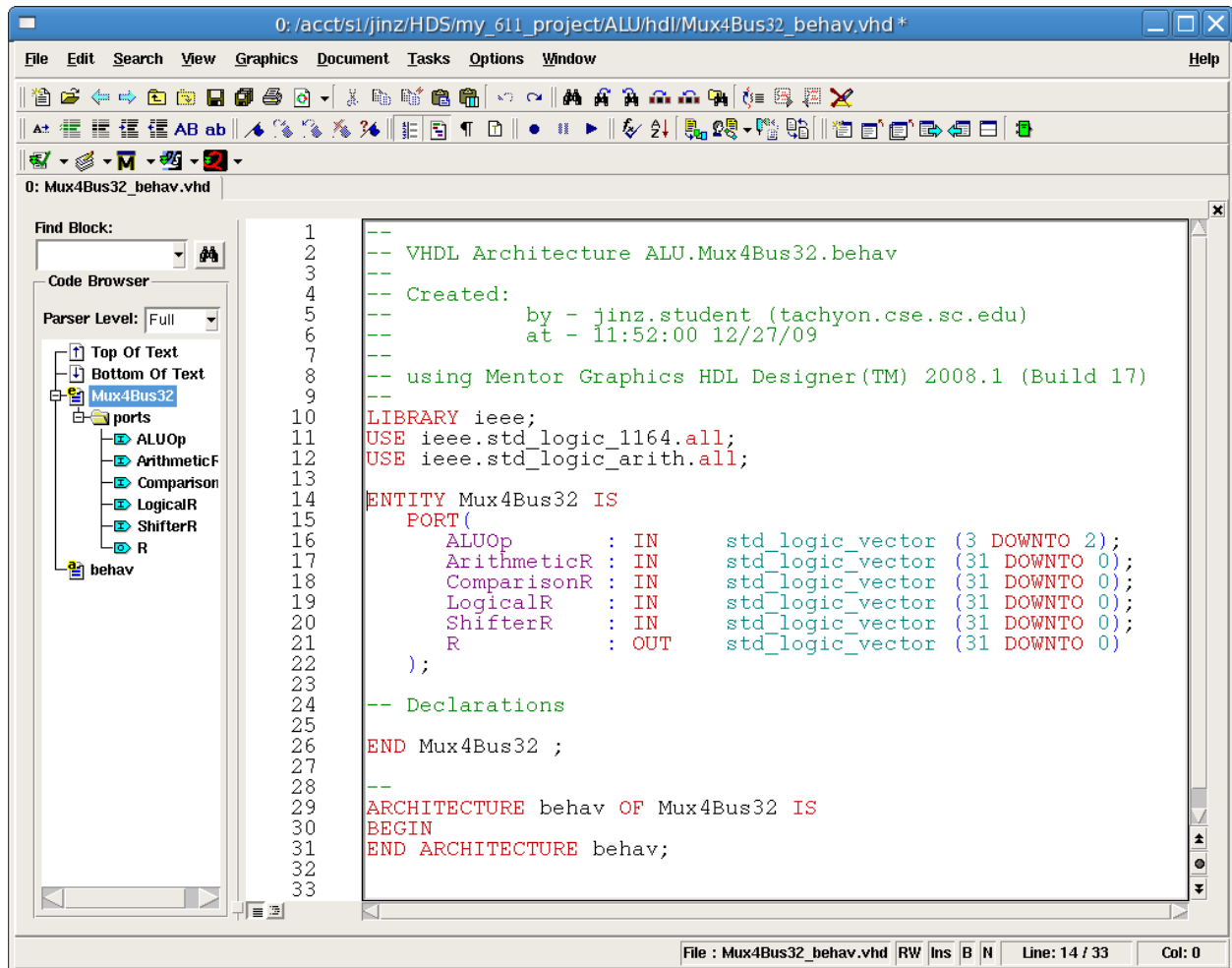
Open the *ALU* block diagram and double-click on the *Mux4Bus32* sub-block. From the window which appears select **VHDL File** from the left pane. The right pane will appear with two file types, select **Combined** and click the **Next** button.



Enter *behav* as the name of the view and then click the **Finish** button.



This will bring up a **VHDL Editor** with the skeleton VHDL Architecture declaration seen in the diagram below:



Now, just as in the embedded block, we will be using a conditional assignment statement to infer the multiplexor. The conditional assignment statement has the syntax:

```

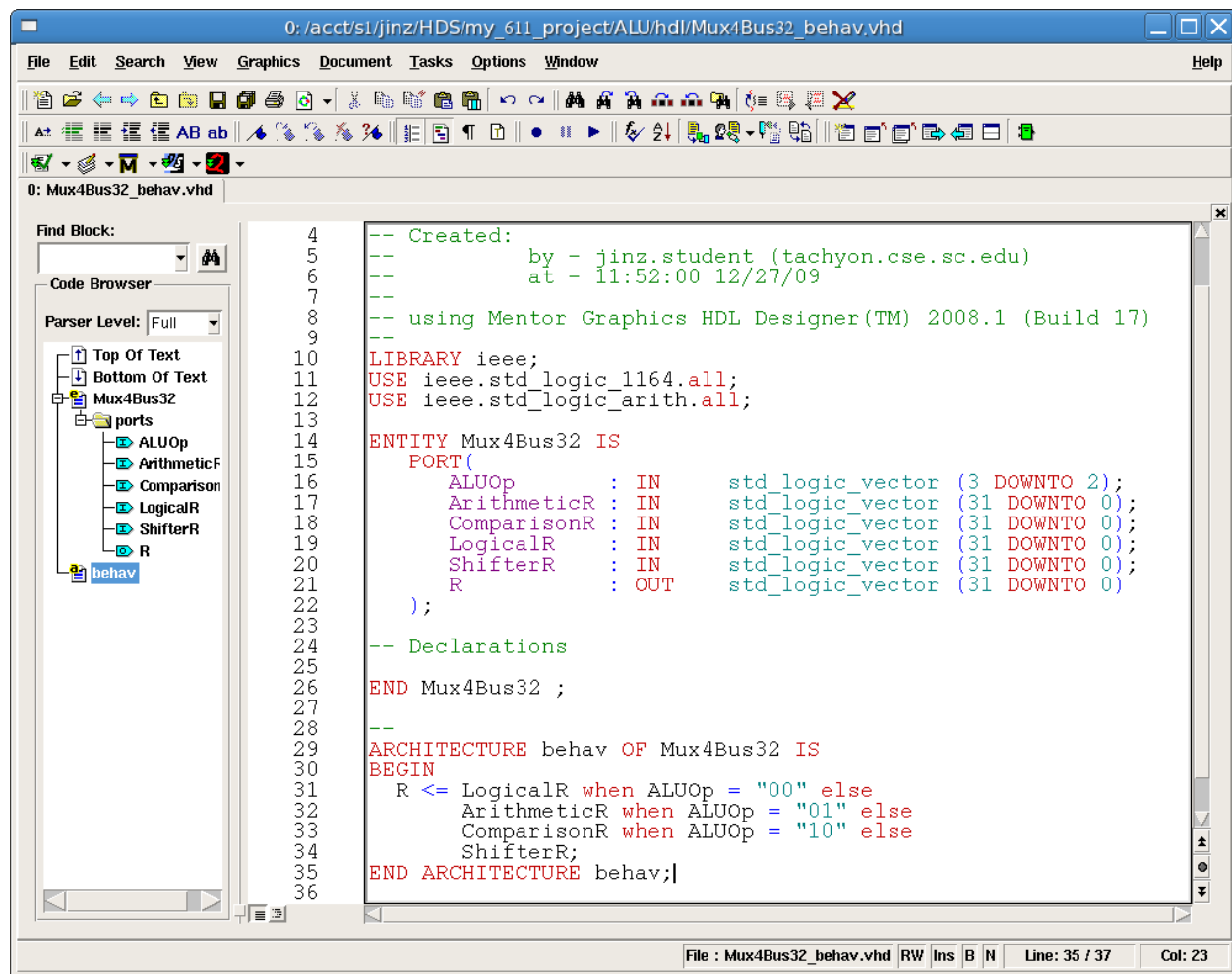
TargetSignal <= SourceSignalA when (ConditionA) else
                SourceSignalB when (ConditionB) else
                ...
                SourceSignalZ;

```

In this case, the target signal is *R* and there are four source signals: *LogicalR* when *ALUOp*(3 DOWNTO 2) is "00"; *ArithmeticR* when *ALUOp*(3 DOWNTO 2) is "01"; *ComparisonR* when *ALUOp*(3 DOWNTO 2) is "10"; and *ShifterR* otherwise (when *ALUOp*(3 DOWNTO 2) is "11").

The conditional signal assignment statement can be a concurrent statement, meaning it can be at the top level of the architecture outside of any processes. Since this is the only statement necessary to describe the functionality of the *Mux4B32* sub-block, the VHDL Architecture

should look something like Figure below. Note that if you are assigning the value of one `std_logic_vector` to another of the same size (*R* and *LogicalR* for example) then you can omit the slice information.



```
0: Mux4Bus32_behav.vhd
Find Block:
Code Browser
Parser Level: Full
Top Of Text
Bottom Of Text
Mux4Bus32
  ports
    ALUOp
    ArithmeticF
    Comparison
    LogicalR
    ShifterR
    R
  behav
4  -- Created:
5  --       by - jinz.student (tachyon.cse.sc.edu)
6  --       at - 11:52:00 12/27/09
7
8  -- using Mentor Graphics HDL Designer(TM) 2008.1 (Build 17)
9
10 LIBRARY ieee;
11 USE ieee.std_logic_1164.all;
12 USE ieee.std_logic_arith.all;
13
14 ENTITY Mux4Bus32 IS
15     PORT(
16         ALUOp       : IN     std_logic_vector (3 DOWNTO 2);
17         ArithmeticR : IN     std_logic_vector (31 DOWNTO 0);
18         ComparisonR : IN     std_logic_vector (31 DOWNTO 0);
19         LogicalR    : IN     std_logic_vector (31 DOWNTO 0);
20         ShifterR    : IN     std_logic_vector (31 DOWNTO 0);
21         R           : OUT    std_logic_vector (31 DOWNTO 0)
22     );
23
24 -- Declarations
25
26 END Mux4Bus32 ;
27
28 --
29 ARCHITECTURE behav OF Mux4Bus32 IS
30 BEGIN
31     R <= LogicalR when ALUOp = "00" else
32         ArithmeticR when ALUOp = "01" else
33         ComparisonR when ALUOp = "10" else
34         ShifterR;
35 END ARCHITECTURE behav;|
36
File : Mux4Bus32_behav.vhd RW Ins B N Line: 35 / 37 Col: 23
```

Save the file and exit the **VHDL Editor** after you have entered the text properly. In the next tutorial, we will test the ALU design by creating a testbench.