## Building a Debug

This portion of the experiment was dedicated to the construction of a debug unit for the ALU used in a previous lab. Once the example code was added to the project it was required that code used in previous labs needed to be added to the project as well, which can be observed in Figure 1.

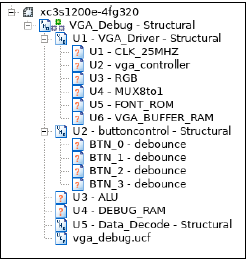


Figure : Required Source Code

The source code used to resolve this issue included the following:

|  |  |
| --- | --- |
| * alu\_arithmetic\_unit.vhd | * front\_rom\_ascii.vhd |
| * alu\_logic\_unit.vhd | * load\_store\_unit.vhd |
| * alu\_mux.vhd | * mux8to1.vhd |
| * alu\_shift\_unit.vhd | * rgb.vhd |
| * alu\_tb.vhd | * clk25MHz.vhd |
| * alu\_toplevel.vhd | * debounce.vhd |

It was also required that RAM needed to be generated for the VGA buffer and for the debugger. The Xilinx Core Generator tool was used to generate RAM for the VGA buffer and the debugger similar to how it was generated in the previous lab. A block diagram was then created to develop a better understanding of the functionality of the debug unit and can be observed in **Appendix XX**.

Once the required files were added to the project the Nexsys was then flashed and tested to test proper functionality. The switches were used as an input for a hexadecimal value that would be translated into the ascii character that would be outputted to the screen. The buttons were used to trigger flags in the code which determined what would be displayed on the computer monitor.

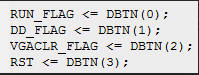


Figure : Flags triggered by buttons