**Homework #8b**

# Name: Section:

# submit via gradescope

Build a circuit to read in a 16-bit KEY using a two-line handshake; the circuit is a passive consumer.

The circuit should search an 18kx16 RAM, counting the number of words that match KEY.

Assume the RAM is preloaded with data.

You are given hw8b\_tb.vhdl,

You are given partial files for HW8b\_design\_template.pptx, hw8b\_cu.vhdl, and hw8b\_dp.vhdl

Which you will need to complete.

Here is the mini-C to implement:

1. while(1) {

2. while(REQ == 0);

3. KEY = data;

4. ACK = 1;

5. while(REQ == 1);

6. ACK = 0;

7. match = 0;

8. for(i=0; i<1024; i++) {

9. MBR = RAM[i];

10. if (MBR == KEY) {

11. match=match+1;

12. } // end if

13. } // end for

14. } // end while

You must use the mini-C design method taught in lesson 10.

Turn in:

1. Block diagram of your datapath design. (see HW8b\_design\_template.pptx and hw8b\_dp.vhdl)
2. State diagram for your control unit design (see HW8b\_design\_template.pptx and hw8b\_cu.vhdl)
3. The output control word table for your design. (see HW8b\_design\_template.pptx and hw8b\_cu.vhdl)
4. A testbench and partial datapath and partial control unit VHDL code are provided. The datapath has a partial BRAM instantiation with 1024 16-bit initial data values preloaded. Upload to bitbucket your final version of the control unit and datapath VHDL code.
5. Your testbench output simulation plots showing at least the following signals - remove all other signals. One plot should be zoomed in showing all the states of the statemachine executing for the 1st MBR value fetched from memory and incrementing “match” and another plot showing the entire run for the 1024 values, with the final value for “match”
   * + clk
     + reset
     + REQ
     + ACK
     + Data
     + Match
     + MBR
     + i (or address into BRAM)
     + control unit state
     + Status word
     + Control word
6. With the testbench providing an initial data value of 0x7FFF, how many matches did you get when you searched the 1024 values?