Answer sheet for CME341 October, 2016 Midterm

Name: _____

Exam Number: _____

Change `exam_dependent_seed` to `8'H`_____

Answers:

Q1: 16'H_____

Q2: 16'H_____

Q3a: 16'H_____

Q3b: 16'H_____

Q4: 16'H_____

CME341 - October, 2016 Midterm
Material up to end of Assignment 5

`Date: Wednesday, October 12, 2016`
`Time = 1.00 hours`
`Text Books, Notes and Computer Files Only`

Before starting the exam make sure you have modified the second "initial" procedure in the test bench so that `exam_dependent_seed` is assigned the value on your answer sheet.

(2) 1. Redesign prototype "student_circuit" to implement a combinational logic circuit that has `cct_output` equal to zero while `clear` is high and equal to `cct_input` plus the unsigned 8-bit constant that has a decimal value of 119 when `clear` is low.

   The accumulator output, i.e. `accumulator_output`, when `counter_full_bar` is low
   for `seed = 8'HAA` should be `16'HDB68`.
   Once `accumulator_output` is correct for `seed = 8'HAA` report it for the seed on your answer sheet.

(2) 2. Redesign prototype "student_circuit" again. This time design a combinational logic circuit that sets `cct_output` to zero while `clear` is high and while clear is low it does the following

   If the most significant 3 bits of input `cct_input` is either 3'b111 or 3'b000, then `cct_output` should equal `cct_input`.

   If the most significant 3 bits of `cct_input`, when treated as an unsigned number, are between 3 and 5, inclusive, then `cct_output` equals $255 -$ `cct_input`.

   Otherwise `cct_output` is that of `cct_input` with the positions of bits 0 and 7 exchanged, that is bit 7 of `cct_output` will equal bit 0 of `cct_input` and bit 0 of `cct_output` will equal bit 7 of `cct_input`. Bits 1 through 6 of `cct_output` will equal bits 1 through 6 of `cct_input`.

The accumulator output, i.e. `accumulator_output`, when `counter_full_bar` is low
for `seed = 8'HAA` should be `16'H049B`.
Once `accumulator_output` is correct for `seed = 8'HAA` report it for the seed on your answer sheet.

(2) 3. (a) Redesign prototype "student_circuit" again. This time the circuit must have an 8-bit counter called `count_down`. The counter is to be synchronously cleared when `clear==1'b1` other wise it should count down by 1 on each rising edge of input `clk`.

The output, `cct_output`, is to be the exclusive-or of `count_down` and `cct_input`.

The accumulator output, i.e. `accumulator_output`, when `counter_full_b` is low
for `seed = 8'HAA` should be `16'FCF2`.
Once `accumulator_output` is correct for `seed = 8'HAA` report it for the seed on your answer sheet.

(2) (b) Modify counter `count_down` to count up by 1 if the most significant bit of `cct_input` is 1'b1. It should still count down by 1 if the most significant bit of the input is 1'b0.

`cct_output` is still to equal the exclusive-or of `count_down` and `cct_input`.

The accumulator output, i.e. `accumulator_output`, when `counter_full_b` is low
for `seed = 8'HAA` should be `16'H4688`.
Once `accumulator_output` is correct for `seed = 8'HAA` report it for the seed on your answer sheet.

(2) 4. Redesign prototype "student_circuit" again. This time design a sequential circuit that loads the ones complement (i.e. all bits inverted) of the 8-bit input `cct_input` into a register called `delay` on the positive edge of input `clk` and then makes the output the "exclusive-or" of `cct_input` and `delay`. The register `delay` is to be synchronously cleared when `clear == 1'b1`.

The accumulator output, i.e. `accumulator_output`, when `counter_full_bar` is low for `seed = 8'HAA` should be `16'HFCF2`.

Once `accumulator_output` is correct for `seed = 8'HAA` report it for the seed on your answer sheet.