Lab 3 (15 marks)

Files from eDimension:

- lab3.v
- labkit_nexys3.ucf

Checkoff List:

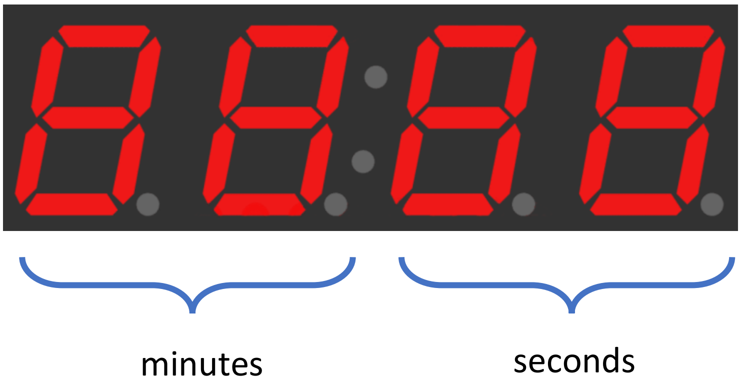
Please be ready with the following when checking off Lab:

1. Have your Verilog code ready to be examined on the computer monitor.
2. Demonstrate that the number displayed in the two rightmost seven segments increments by 1 with each second, and which becomes zero after 59.
3. Demonstrate that the number displayed in the two leftmost seven segments increments by 1 with each 60 seconds.
4. Demonstrate the extra features/functionalities that you implemented (exercise 2) in addition to the expected outcome in exercise 1.

During checkoff you may be asked to discuss one or more questions.

Exercise 1: Implementing a digital clock using Verilog (5 marks)

In this exercise, you will implement a Verilog module that takes the labkit's (NEXYS 3) 100 MHz clock (oscillator) as an input and displays seconds and minutes of a **digital clock** using four seven segments, as shown in the following figure. That is, the seconds are displayed in the two rightmost seven segments and the minutes are displayed in the two leftmost seven segments.



Download the source files from eDimension: **lab3.v** and **labkit_nexys3.ucf**. Then, create a new Xilinx project that includes lab3.v and labkit_nexys3.ucf. Compile and load the bit file into the labkit. You should be able to see that the static number "1234" is displayed in the four seven segments. Now, change the Verilog code (lab3.v) so as to achieve the functionality explained in the above.

Implementation Tips:

- Increment the value in “**displayed_number**” register by 1 with each second.
- **Modulo operator (“%”)** will be useful in extracting seconds and minutes from the “**displayed_number**” register. If x and y are integers (or registers), then the expression: x % y produces the remainder when x is divided by y. For example, let **displayed_number** = 125. Then, seconds = 125%60=5 and minutes = 125/60=2 (note that the outcome of any integer division is also an integer in Verilog).

Exercise 2: Adding extra functionalities to Exercise 1 (10 marks)

In this exercise, you are expected to add any extra features/functionalities to the above implemented design. Few ideas are as follows:

- Use switches/push buttons to control the digital clock, e.g., pause clock, change time.
- Implement an alarm/timer while the LEDs are used as indicators (instead of a buzzer/speaker).