EE 4702 **Homework 3** **Due: 4 April 2001.**

Solve this problem by modifying a copy of `http://www.ee.lsu.edu/v/2001/hw03.html` *(or .v) which can also be found in* `/home/classes/ee4702/files/v/hw03.v`. *See* `http://www.ee.lsu.edu/v/proc.html` *for instructions on running the simulator. Alternate instructions can be found in Lesson 7 of the ModelSim Tutorial, linked to the references web page,* `http://www.ee.lsu.edu/v/ref.html`. *The links are clickable when this assignment is viewed with Acrobat Reader. The ModelSim tutorial and other documentation can also be accessed from the Help menu on the ModelSim GUI (started by the command vsim -gui).*

Problem 1: Write a Verilog behavioral description of a microwave oven controller in module `microwave_oven_controller`. The module has two inputs, `key_code` and `clk`. The user operates the oven through a keypad, the keypad has a six-bit output which is connected to the controller through the port named `key_code`. Values for `key_code` are given in the template. As with the calculator described in class, when no key is pressed `key_code` is `key_none` (see the template). The keypad is de-bounced and a user must release one key before pressing another. One-bit input `clk` is a 64 Hz clock.

The controller has six outputs, `beep`, `dmt dmu`, `dst`, `dsu`, and `mag_on`. When one-bit output `beep` is 1 the oven will emit a tone. Four-bit output `dmt` (display minute tens) is connected to the tens digit of the oven minutes display, output `dmu` (display minute units) is connected to the unit digit of the minutes display, `dst` is connected to the tens digit of the seconds display, and `dsu` is connected to the unit digit of the seconds display. The display will properly render digit values 0-9 and will display nothing for a digit value of 10.

When controller output `mag_on` is 1 the magnetron is on (and so the oven is heating).

The keypad has keys for each digit `key_0` - `key_9`, and keys `key_power`, `key_start`, and `key_reset`. There is no popcorn button. The oven operates as follows: When the oven is plugged in it should be placed in a reset state in which the magnetron is off and the display shows zero minutes and zero seconds. To cook at full power, the user enters 1 to 4 digits and presses start. (The digits indicate the cooking time in minutes and seconds. The number of seconds entered must be in $[0, 59]$, so if the user wants to cook for 90 seconds 130 must be entered, not 90.) To cook at some other power the user presses a digit, power, then 1 to 4 digits for the time, then start. Digit 9 indicates 90% of full power, 8 indicates 80% of full power, etc.

Once commanded to start, the oven turns the magnetron on and off until the set time has elapsed. For full power the magnetron stays on over the entire interval. To cook at partial power the magnetron is turned on for a part of each 2.5-second interval. For example, to cook at 30% power the magnetron would be on for 0.75 seconds, off for 1.75 seconds, on for 0.75 seconds, and so on.

The controller must update the display as the user is entering the power and time and while the oven is heating.

If the user presses reset once while the oven is heating the magnetron is turned off but the display should show the remaining time. If the user presses reset again the oven should reset, if the user presses start cooking should resume.

If reset is pressed when the oven is not heating then it will go to the reset state and so any partially entered time or power will be lost.

When cooking is complete the oven should go into the reset state and sound a 2-second beep.

Whenever an invalid key is pressed, even when heating, the oven should emit a 250 ms beep. A key is invalid if it has no meaning when pressed, for example, pressing a digit while heating or pressing start with more than 59 seconds.

Resist the urge to gold plate your submission, for example, by adding outputs for a power indicator or using the display for a clock when not cooking. This will only confuse the TA-bot. Instead, discuss any such ideas with the instructor.