Digital System Design and Implementation

Lab. 3

(Due on 05/17 PM 8:00)

Note: Please upload your codes and hand in the **hardcopy** of this experiment including

a. Verilog codes

b. Test bench

c. Simulation results.

Total points :150 points including 50 points for demo.

In this Lab., we will learn to use sequential logic to sense the pressing of a push button and the debouncing technique. Also, we will learn how to sensing keyboard input.

Assume that we are playing a baseball game

(a) Use the **RESET** button.

(b) Define the Keys "**F**"(**Fastball**), "**C**"(**Change up**), "**S**"(**Slider**).

(c) Define push button "S4" as "**Increase**", push button "S1" as "**Decrease**" for velocity. Initially, the default velocity is 135Km/hr, which is shown in the seven-segment of group 1. If button "Increase" is pressed, the velocity is increased by 5 Km/hr until 160 Km/hr. If button "Decrease" is pressed, the velocity is decreased by 5 Km/hr until 120 Km/hr. (Debouncing is necessary for these two buttons.)

(d) For even-numbered students, define push button "**S3**" as "**Pitch**". For odd-numbered students, define push button "**S0**" as "**Pitch**". (Debouncing may not be necessary for this button.) The shining of LED will be changed when the player presses "**Pitch**".

Initially, the seven-segment group 0 shows nothing, but the seven-segment group 1 shows the default velocity as given in Fig. 1.

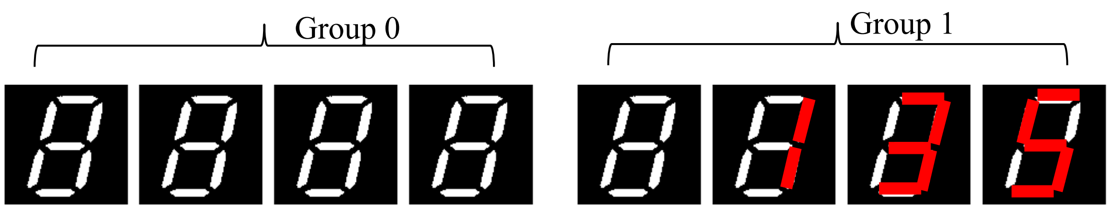
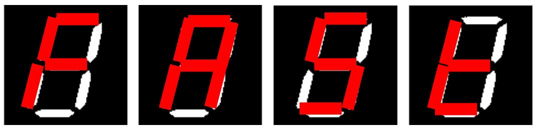
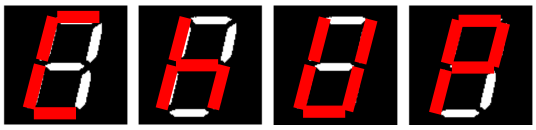


Fig. 1 Initial display

Then, we can use keyboard to set the pitch type. If the player pressed “**F**” or “**C**” or “ **S**” on the keyboard, the seven-segment group 0 shows the type as given in Fig. 2. You can change the pitch type arbitrarily before you pressed the button “**Pitch**”.


(a)


(b)

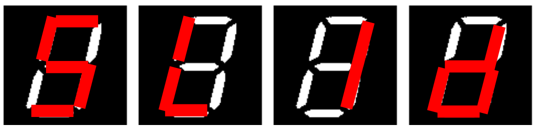
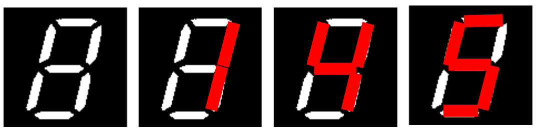

(c)

Fig. 2 The display for pitch type, (a)when “F” is pressed”, (b) when “C” is pressed, and (c) when “S” is pressed.

The speed can be changed by using the push buttons “**Increase**” and “**Decrease**”. The display of seven-segment group 1 will change accordingly. Fig. 3 shows the results when the button “**Increase**” is pressed twice.


(a)

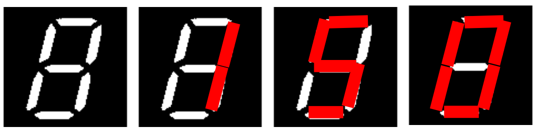

(b)

Fig. 3 The display of the speed (a) after the player presses “**Increase**” twice, and (b) after the player presses “**Increase**” three times.

After the player enters “**Pitch**”, we use LED to show the trajectory of the ball. For simplicity, only two changing rates for the LED pattern are used.

(a) For even-numbered students, if the speed is greater than or equal to 140Km/hr, 2 Hz changing rate is used and if the speed is smaller than 140 Km/hr, 1 Hz changing rate is used for the LED pattern. The shining pattern of the LEDs starts from the right side.

(b) For odd-numbered students, if the speed is greater than or equal to 140Km/hr, 2 Hz changing rate is used and if the speed is smaller than 140 Km/hr, 0.5 Hz changing rate is used for the LED pattern. The shining pattern of the LEDs starts from the left side.

(c) ***For all students, if the ball is change up, please let the last eight LEDs flash at 2Hz. If your student ID is even, the first eight LEDs should flash at 1Hz, and if it's odd, they should flash at 0.5Hz***

Basically, after the button "**Pitch**" is pressed, the first eight LEDs will be lit up one by one and only one LED is ON each time for three pitch types. The LED patterns for different pitch types vary only for the last eight LEDs. Here, we show the complete pattern for fastball ("**F**") of an even-numbered student in Fig. 4. Fig. 5 and Fig. 6 describe the LED patterns for change up ("**C**") and slider ("**S**"). For odd-number students, only the trajectory is changed from the left to the right side.

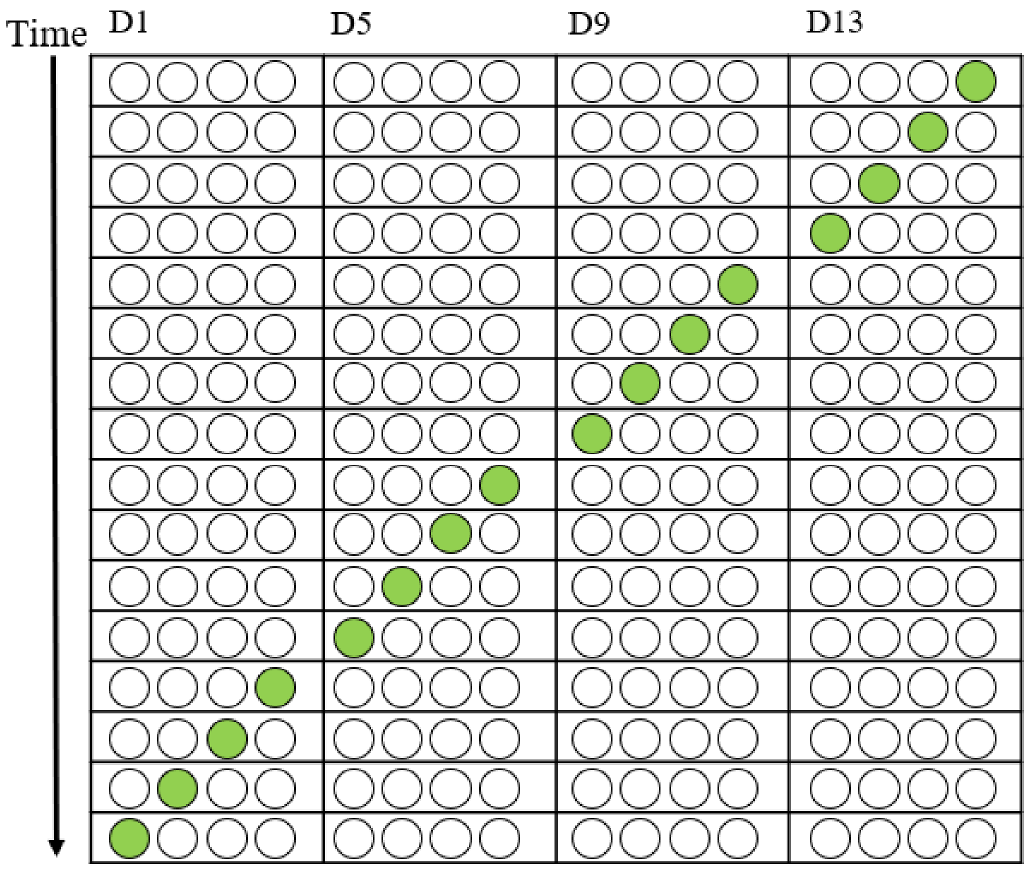


Fig. 4 LED pattern for fast ball.

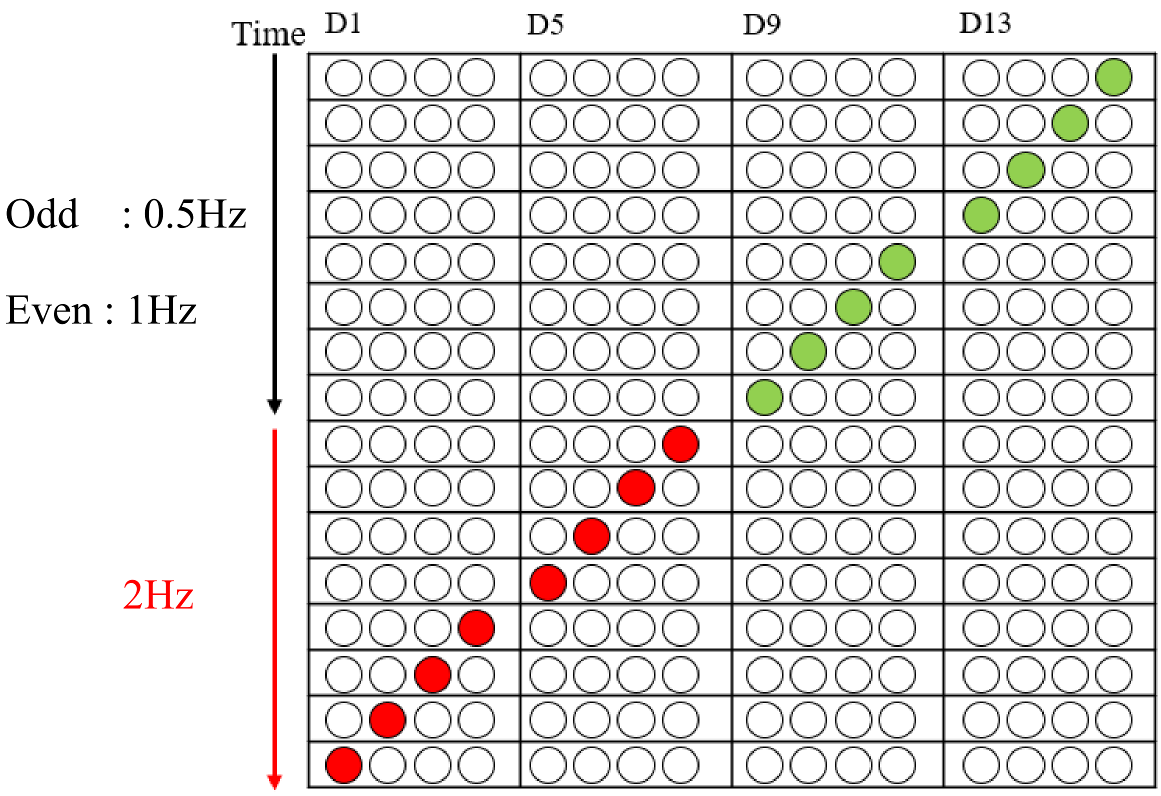


Fig. 5 LED pattern for change up.

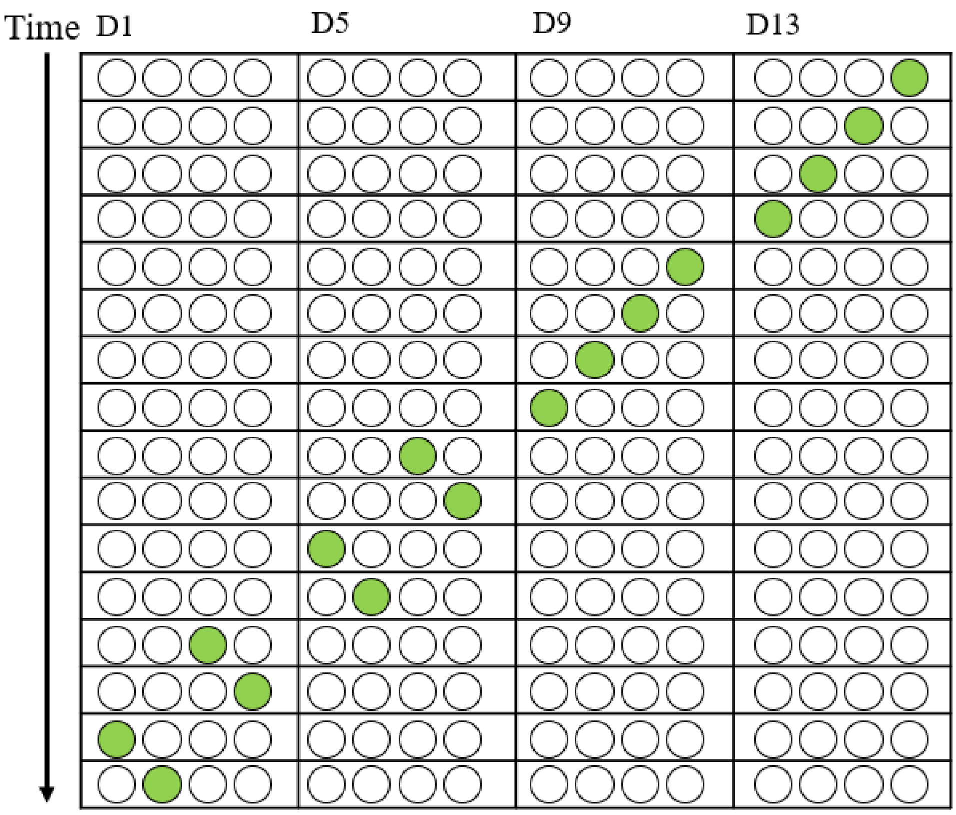


Fig. 6 LED pattern for slider.

1. Write verilog codes for the required functions in the lab.
2. Write the test bench for the cases of curve ball with speed **130Km/hr** for even-numbered students. So, you need to press "**Decrease**" once and assume you receive "21(hex)" with the start, parity, stop bits from PS/2 keyboard sequentially. Write the test bench for the cases of slider with speed **140 Km/hr** for odd-numbered students. So, you need to press "**Increase**" once and assume you receive "1B(hex)" with the start, parity, stop bits from PS/2 keyboard sequentially. Because of the long interval of two adjacent key-pressing operations compared to the clock frequency of 100 MHz, you can simply **a proper period** for your key-pressing operation in your test bench to observe the related output waveform. Note that you don't need to insert the bouncing phenomenon by yourselves. However, the existence of the debouncing module should not influence the result.
3. Show the behavior simulation results.
4. Demo in the lab time. Note the proper setting of the shining frequency and the debounce period.