Countdown Timer

Rohit Timmagi, Syed Hussain, Tala Fakhouri, Matthew Buffa

Electrical and Computer Engineering Department
School of Engineering and Computer Science
Oakland University, Rochester, MI
e-mails: rtimmagi@oakland.edu, sdhussain@oakland.edu, tsfakhouri@oakland.edu, mjbuffa@oakland.edu

Abstract— **The design's purpose is to count backwards from a user defined time, which could be employed to notify the user after that specific amount of time has passed. VHDL code for the design includes an implementation of both combinational and synchronous circuitry. The design includes 5 counters: two modulo-6 counters, two BCD counters, and one generic counter. The BCD and modulo-6 counters count back from the entered time when their enable input is high. In order for the BCD and modulo-6 counters to update every second, a generic counter was used to slow the master clock signal, providing a signal that would enable of the BCD and modulo-6 counters every second. A serializer was also used to display different values on the seven-segment displays. Additionally, a series of 13 switches serve as the time input, allowing a user to set the amount of time from which the circuit counts down. Finally, a reset and load button were used to reset the display, and to load the user's desired time respectively.**

I. Introduction

Two drag racers awaiting launch, microwaving a frozen dinner, and watching the ball descend in New York to welcome the New Year all have one common theme: they necessitate the use of a countdown timer. The countdown timer's use is extremely broad. Given such a simple concept as counting down, it is easy to underestimate the complexity of a countdown timer's circuitry. This document will explore each facet of the design, including discussion on how the serializer, BCD, modulo-6 and generic counters each function in unison to create such a useful circuit that is often taken for granted.

II. Methodology

A. Generic Counter

A generic counter is not only useful to generate pulses of different frequencies, but can also count to arbitrary bounds. This generic counter was used to decrease the master clock signal to provide a logic high every one second, which allowed the BCD and modulo-6 counter to decrease their values every second. This was done by modifying the generic counter to count an upper bound of 10^8. With a master clock frequency of 100MHz, outputting a logic high upon reaching this upper bound would provide a signal acted as a one second clock pulse. This one second clock pulse was then connected to each of the BCD and modulo-6 counters. This generic counter was only active so long as the pause and reset buttons were not actuated.

B. BCD and Modulo-6 Counters

BCD counters can be used to count between zero and nine with every rising edge of a clock signal. Modulo-6 counters are used to count between zero and five. The BCD counters were used to count the least precise values of minutes and seconds (i.e. the boldface values - 5**9**:5**9**). Modulo-6 counters are used to count the tens place of the minutes and seconds (i.e. the boldface values - **5**9:**5**9**)**. Each of the BCD and modulo-6 counter's are enabled only when the previous counter reaches its minimum value and when the generic counter outputs its one second clock pulse. Additionally, the counters output a logic high when reaching their minimum values, thereby enabling the next counters to change their values. These four counters each have "load" and "start time" inputs which allows the counters to start counting from the user specified value. Finally, each of the counter's value is outputted to the inputs of the serializer.

C. Serializer

Because the the seven-segment displays are wired in such a way that allows only one value to be displayed at a time, a serializer must be used to create the appearance of four different values being displayed simultaneously. The value from each of the four counters are inputted into the serializer. These four values are multiplexed, fed through a hex decoder and then outputted to four seven-segment displays. These values change so quickly that the display appears to show four different continuously lit numbers.

D. Switches, Buttons and LEDs

In order to maximize the usefulness of the circuit, is important to make its design user friendly. A series of 13 switches are used to input the desired amount of time from which the circuit counts down. Switches 0-3 are used to change the first digit i.e. the second's ones place. Switches

4-6 are used to change the second digit i.e. the second's tens place. Switches 7-10 are used to change the third digit i.e. the minute's one place. Switches 11-13 are used to change the fourth digit i.e. the minute's tens place. The switches values are only loaded into the BCD and modulo-6 counter's when the user actuates the the load push button. Two additional push buttons are used to pause the timer, and reset the time. Pressing the pause button disables the generic counter, thereby also disabling the BCD and modulo-6 counters. The reset button is wired to all counters and the serializer. Finally, LEDs are illuminated when the time has reached zero, effectively alerting the user of the time's expiration.

III. Experimental Setup

The software that was used to create this timer was Xlinx ISE Design Suite 14.7. The hardware that was used to implement the design was the Nexus 4 DDR board. The specific inputs and outputs of the board can be seen in the constraints file, which includes: four seven-segment displays, a button to pause time, a button to reset time, switches to input time, and LEDs to notify the time has expired. The design's functionality was tested by loading onto the Nexus board. The device's accuracy was tested through its comparison to a smartphone timer. It was found that the design functions just as well as the smartphone, as their times remained synchronized through the course of their countdown.

IV. Results

The design successfully counts down from the user's inputted values.. The design notifies the user when the time has expired, pauses when the pause button is pressed, and resets when the reset button is pressed. All inputs and outputs function as desired. Attached below is a video of the design functioning.

https://www.youtube.com/watch?v=wEenC2-6kbc

Conclusions

Through the use of BCD, modulo-6 and generic counters, the task of creating a countdown timer was achieved. This was no simple feat, as pausing the time once the time expired, proved to be quite the challenge. This reinforced the importance of drawing the circuit, and using a karnaugh map to derive the boolean expression needed to pause the time. Improvements to the design that could be made include flashing the notification LED thereby drawing more attention to the board when the time has expired.

References

[1] Llmocca D VHDL Coding for FPGAs. In: VHDL Coding for FPGAs. http://www.secs.oakland.edu/~llamocca/vhdlforfpgas.html. Accessed 2 Nov 2016