Lab 5 Report

Design Specification

Lab5_1

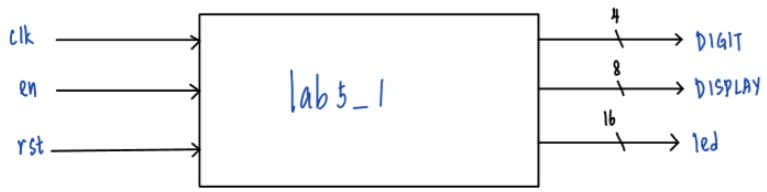
Module lab5_1:
input clk
input rst
input en
output[3:0] DIGIT
output[7:0] DISPLAY
output[15:0] led

Module clk_divider_27:
input clk
output clk_div_27

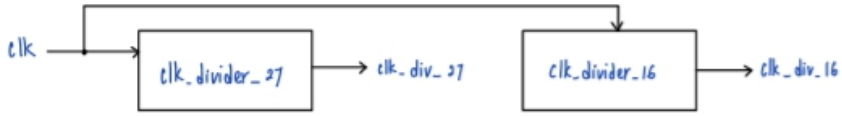
Module clk_divider_16:
input clk
output clk_div_16

Module debounce:
input pb
input clk
output pb_debounced

Module one_pulse:
input pb_debounced
input clk
output pb_one_pulse



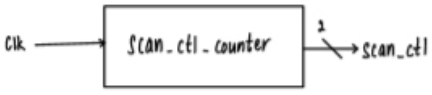
clock divider :

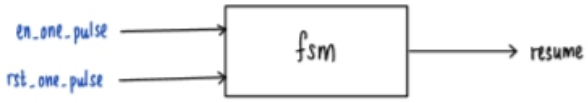


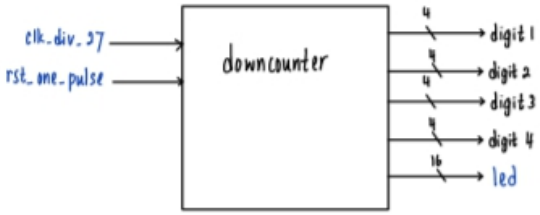
debounces :

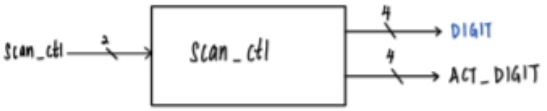
one pulses :

main parts : (always blocks)









Lab5_2

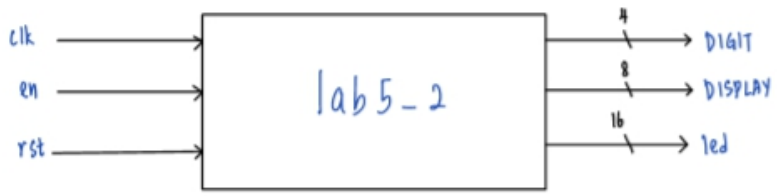
Module lab5_1:
input clk
input rst
input en
output[3:0] DIGIT
output[7:0] DISPLAY
output[15:0] led

Module clk_divider_27:
input clk
output clk_div_27

Module clk_divider_16:
input clk
output clk_div_16

Module debounce:
input pb
input clk
output pb_debounced

Module one_pulse:
input pb_debounced
input clk
output pb_one_pulse

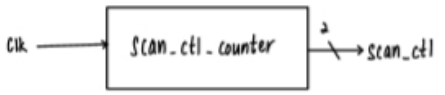


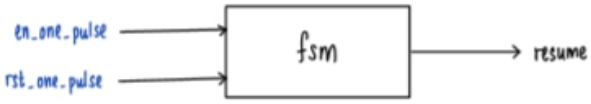
clock divider :

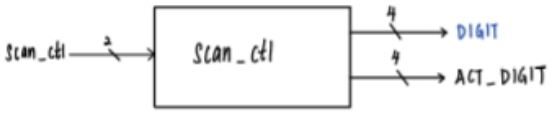
debounces :

one pulses :

main parts : (always blocks)







Lab5_3

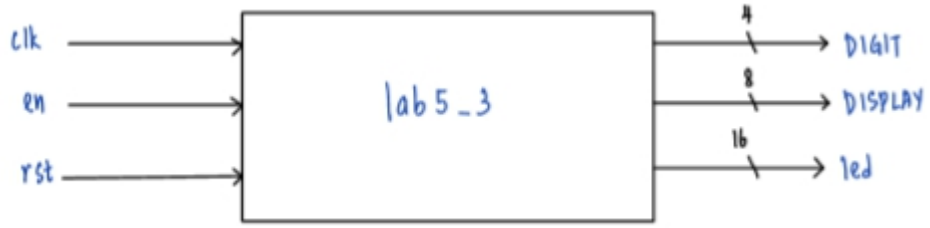
Module lab5_1:
input clk
input rst
input en
output[3:0] DIGIT
output[7:0] DISPLAY
output[15:0] led

Module clk_divider_27:
input clk
output clk_div_27

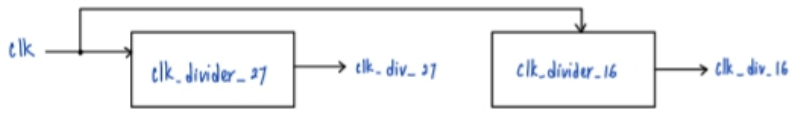
Module clk_divider_16:
input clk
output clk_div_16

Module debounce:
input pb
input clk
output pb_debounced

Module one_pulse:
input pb_debounced
input clk
output pb_one_pulse



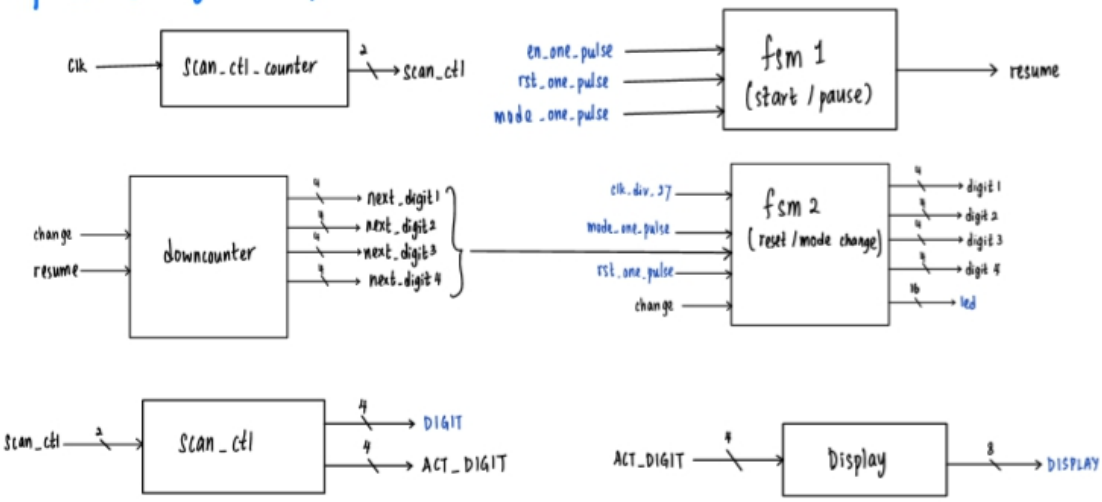
clock divider :



debounces :

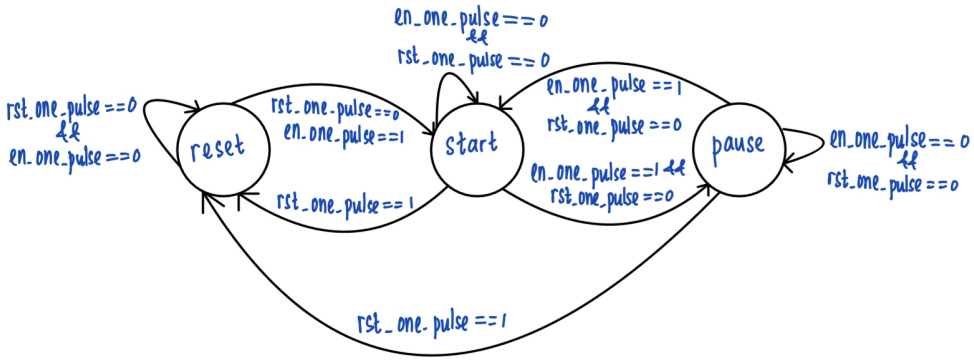
one pulses :

main parts : (always blocks)



Design Implementation

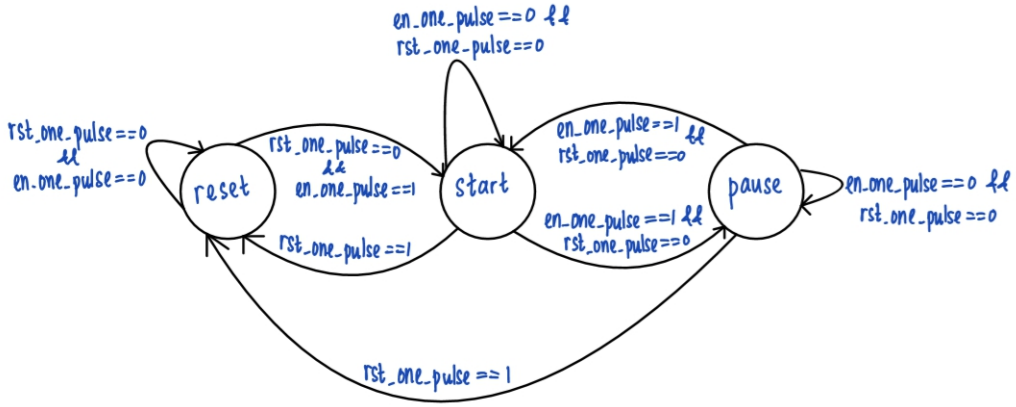
Lab5_1



This experiment is done by using 2 buttons. Function of the first button is the start and pause, and the second one is for reset. The first button is called as en(input), and the second button is called as rst(input). The en_one_pulse is the one pulse from en button, while the rst_one_pulse is the one pulse from rst button. Both of the one pulse is connected to a clock divider of 100Hz, and the downcounter is connected to a 1Hz clock divider. The initial state of the program is at start, which is at 25 second of downcounter as requested. When the en button is pressed, the program will start counting, and if it is pressed again, the program will pause. If the rst button is pressed, the downcounter will go back to initial state.

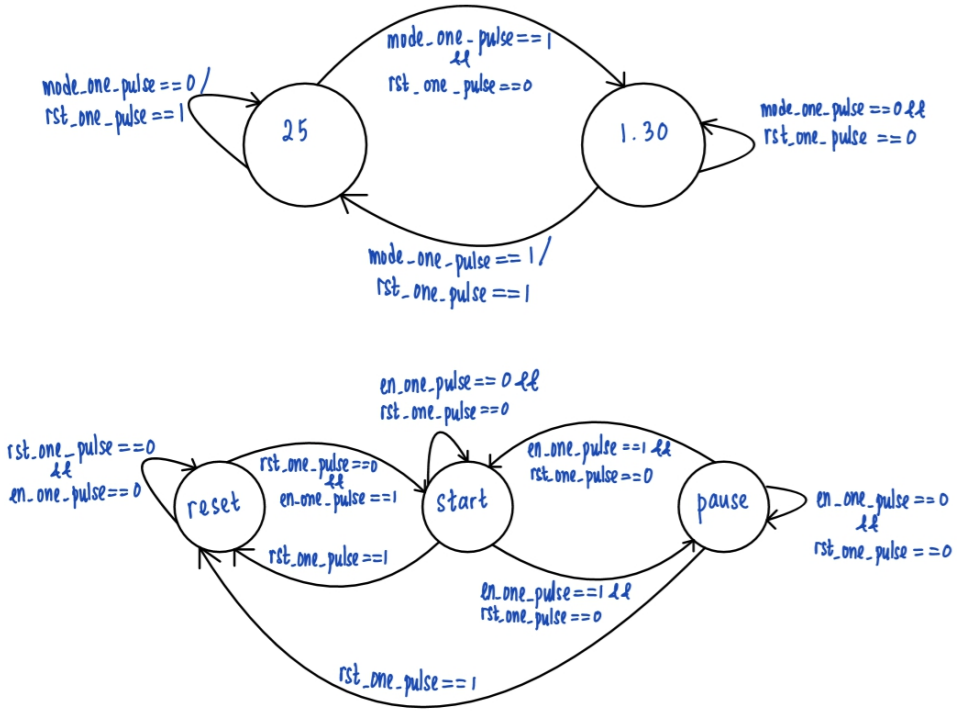
The scan_ctl_counter is a 16 bits counter used to generate the scan_ctl to use for the seven-segment-decoder. The ACT_DIGIT is used to determine the number should be displayed on the seven segment display.

Lab5_2



This experiment is done by using only 1 button. Function of button is the start or pause and reset. The first button is called as en(input). There are 2 one pulse generated from the en button, which is en_one_pulse and rst_one_pulse. The experiment is designed as the en_one_pulse is connected to a clock divider of 100Hz, and the rst_one_pulse is connected to a 1Hz clock divider. Which means, when the en button is pressed for once only, it will be paused or started to count down, while the en button is pressed for approximate to 1 seconds long, the program will be reset to the initial state. The downcounter is connected to a 1Hz clock divider. The initial state of the program is at start, which is at 25 second of downcounter as requested. When the en button is pressed, the program will start counting, and if it is pressed again, the program will pause. If the en button is pressed for approximate 1 seconds long, the downcounter will go back to initial state, which is the 25 seconds.

Lab5_3



This experiment is done by using 2 buttons. Function of the first button is the start or pause and reset, and the second one is for changing the mode of the downcounter. The first button is called as en(input), and the second button is called as mode(input). There are 2 one pulse generated from the en button, which is en_one_pulse and rst_one_pulse. The experiment is designed as the en_one_pulse is connected to a clock divider of 100Hz, and the rst_one_pulse is connected to a 1Hz clock divider. Which means, when the en button is pressed for once only, it will be paused or started to count down, while the en button is pressed for approximate to 1 seconds long, the program will be reset to the initial state.

The mode button is used to change the mode of the downcounter, whether 25 seconds to count down or 1 minutes 30 seconds to countdown. The mode button is connected to a 100Hz clock divider, when the mode button is pressed for once, it will change from initial state (25 seconds) to the other state (1 minutes and 30 seconds), and if it is being pressed again, it will change from 1 minutes and 30 seconds back to 25 seconds. When the state is at 1 minutes and 30 seconds, and receive the rst_one_pulse, the program will go back to 25 seconds countdown.

The downcounter is connected to a 1Hz clock divider. The initial state of the program is at start, which is at 25 second of downcounter as requested.

The next digit is the state digit recorded for the digit to display at the next clock cycle.

Discussion

There is one thing went wrong for my program is that my result is not counting down exactly 1 seconds. After testing for the accuracy, my results shown that there is a 0.3 second to 0.4 second delay for my downcounter. This is because the clock divider (frequency divider) I used is the 27 bits clock divider, which is approximately to 1 seconds but not accurately 1 second. There is no any mistake for my experiment except this small time delay.

Conclusion

I had learned about how to design the 2 different functions for a button, and learned about the one pulse and debounce.

I did realize my small mistake from this experiment, which is, instead of using the 50M counter (1Hz clock divider) for an accurate 1 second clock divider, I'm using 27bits of clock divider, and it did affect my result, which is a little time delay of 0.3 second to 0.4 second. I already improve this problem before the deadline of the submission.

*remark : when demo this whole lab 5, my result shown is 0.3 second to 0.4 second for every second counting down, but after that I did fix it.

