

First lab assignment Report

Designer: Group 4

(TEAM MEMBERS: Qinghui. Liu, Zhili.Shao, Joseph. Fotso)

CONTENTS

1	Problem Statement	1
2	Design solution & Block Diagram	2
2.1	Brief block diagram as below figure.	2
2.2	Modules description.....	3
3	Modules Testbenchs	5
4	Instructions Manual	6
5	Appendix	7
5.1	Detail inside block diagram	7
5.2	Detail FSM diagrams.....	7
6	Reference	7

1 PROBLEM STATEMENT

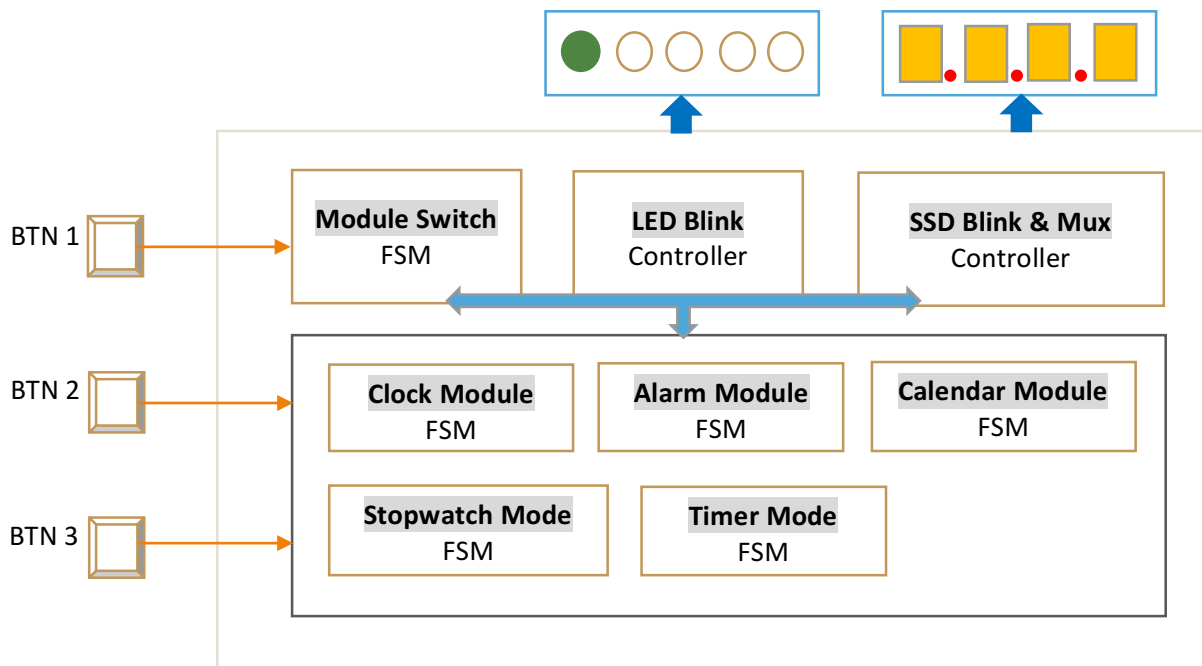
Using only the resources available in the NEXYS 3, namely switches, leds, press buttons, 7-segment displays, etc.... to design an alarm clock. Add other features at your will, like:

- 12h or 24h option;
- A simple calendar with day/month or perpetual calendar with day/month/year;
- Days of the week;
- A chronometer with stopwatch and lap timer;
- Incremental speed when setting clock, alarm clock or calendar;
- Any other creative features!

2 DESIGN SOLUTION & BLOCK DIAGRAM

We plan to use three buttons (**Btn1**, **Btn2**, and **Btn3**), that are used to change the mode, set the time, set the alarm, set date, start/stop the stopwatch, set/start timer and so on, to implement all required functions of the digital clock.

2.1 BRIEF BLOCK DIAGRAM AS BELOW FIGURE.

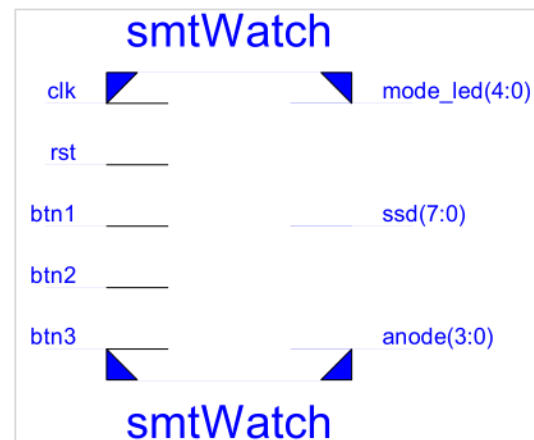


Input:

- **Pressing Btn1** changes the mode from *Clock* to *Alarm*, *Calendar*, *Stopwatch*, *Timer*, and then back to *Clock*.
- **The functions of Btn2 and Btn3** vary upon the mode selected by users.

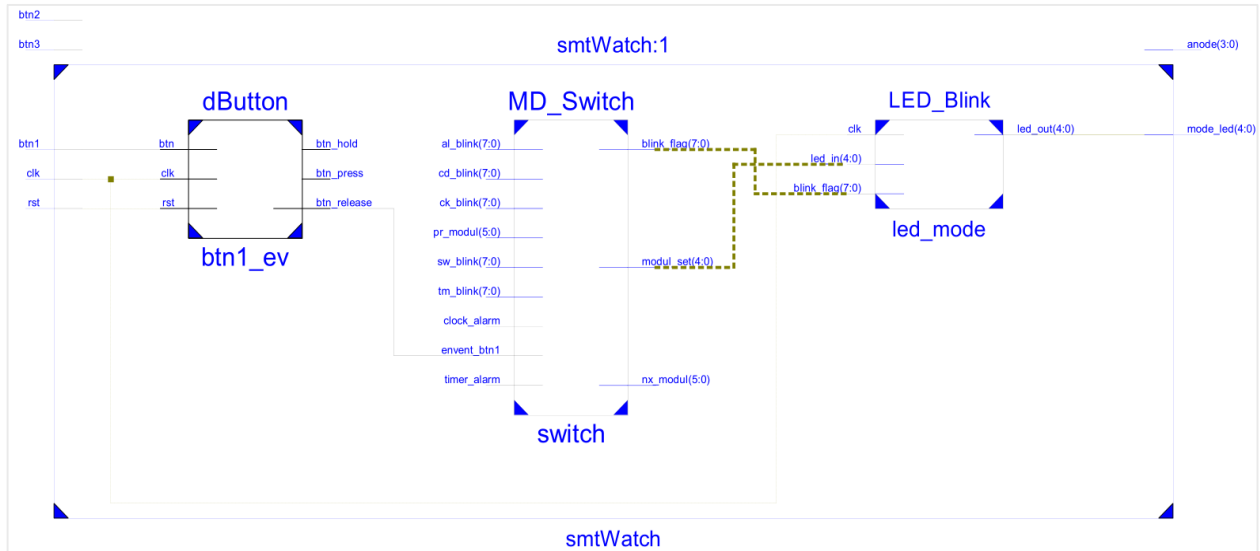
Output:

- **5 LEDs** will be used to indicate which mode is selected by lit on.
- **4-digit 7-segment LCD** will display kinds of information upon the module selected and its state.

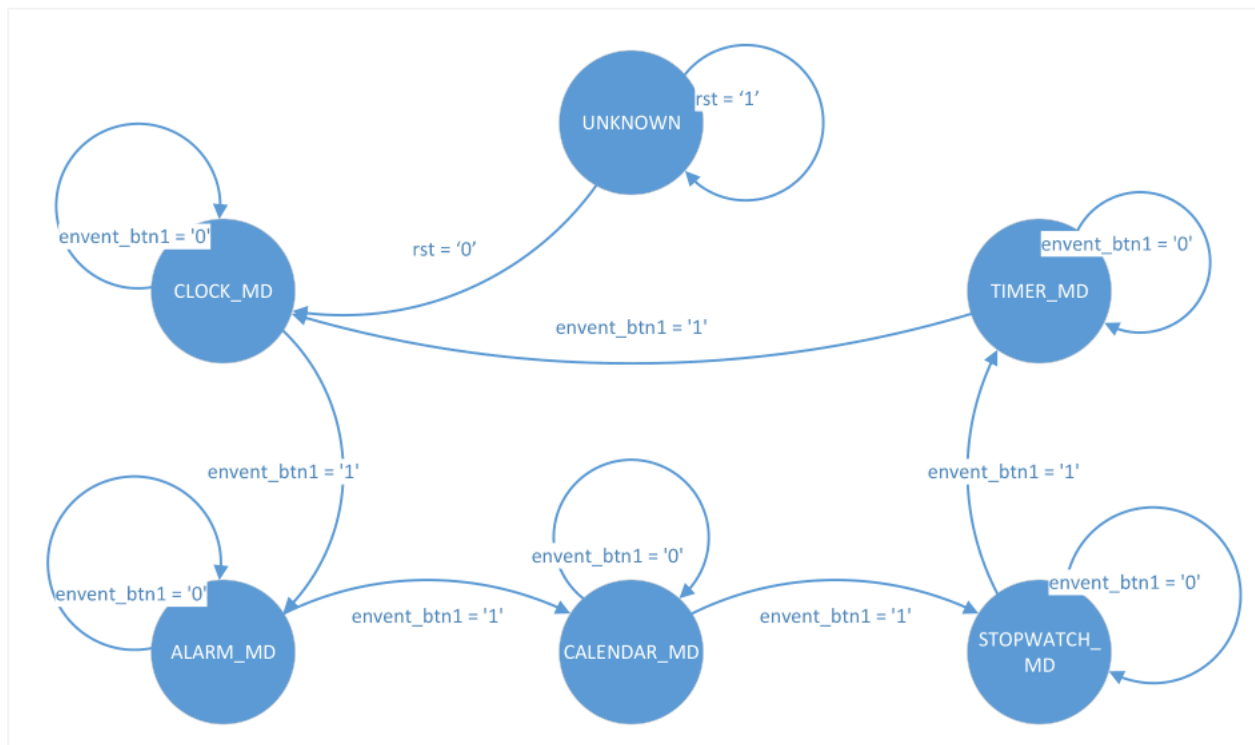


2.2 MODULES DESCRIPTION.

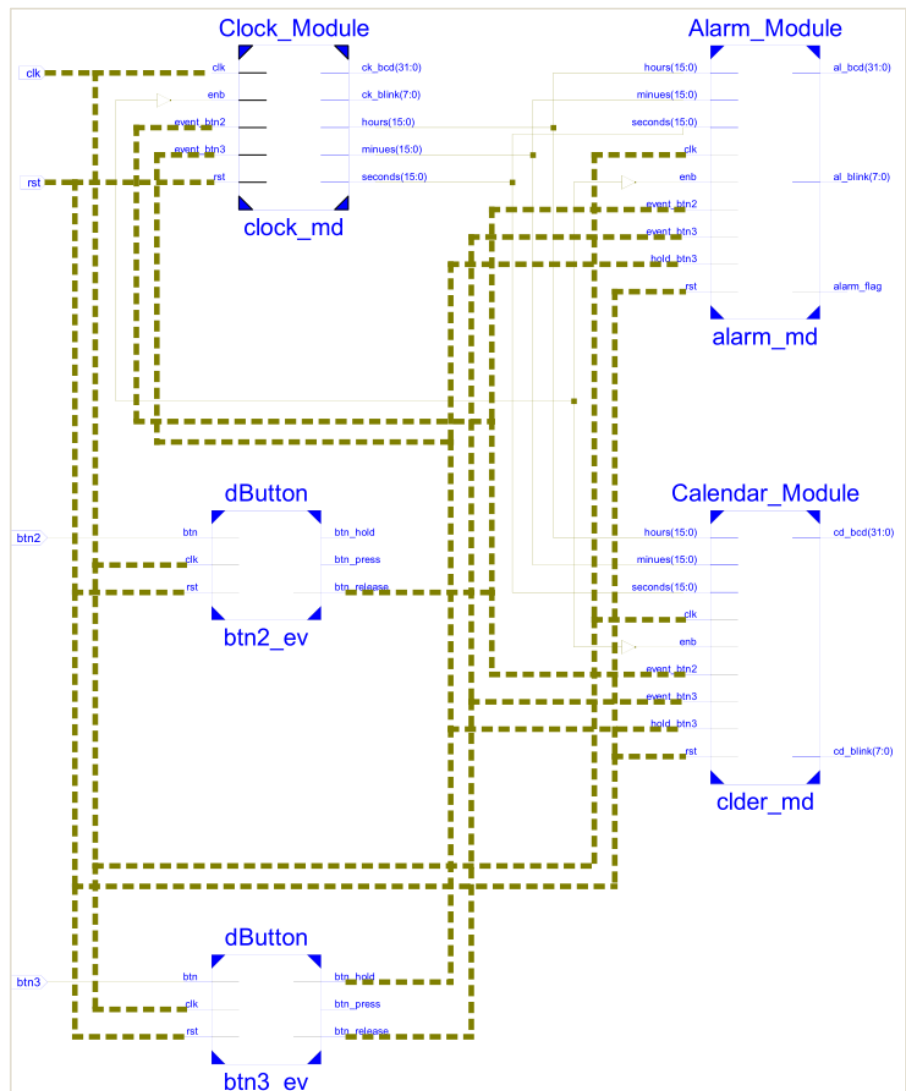
- A. **Module Switch** is designed to process the event of button 1, for each button releasing event, the module will change and loop among these five inside modules: clock, alarm, calendar, stopwatch and timer, and enable relevant module blink flag and lit on relevant LED. As show below figures.



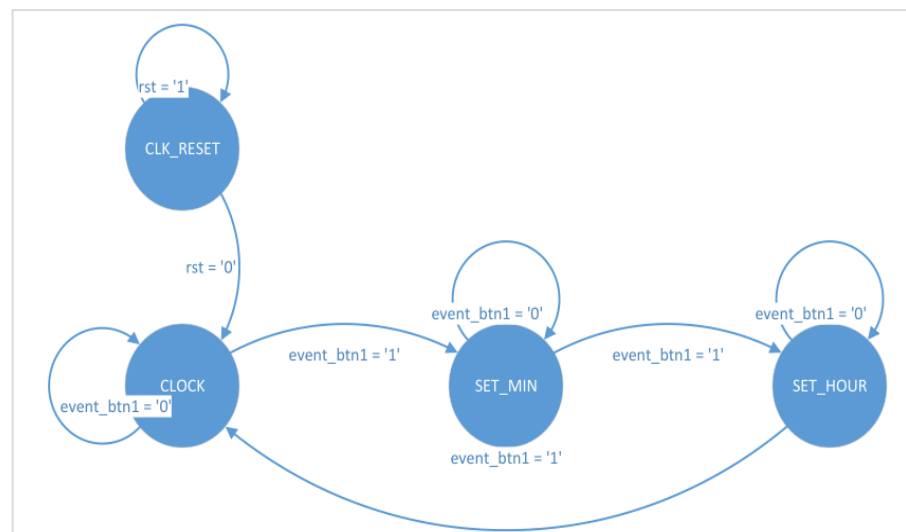
The finite state machine of module switch as shown in below figure.



B. **Clock Module**, **Alarm Module** and **Calendar Module**, are designed to process real-clock related functions. When each of them enabled by button 1, the events of button 2 & 3 will be processed inside of the module enabled. Basically, holding button3 during setting state, will change the value selected including the values of minutes, hours (in Clock or Alarm modules), day, months, years, and days of weeks (Calendar module). Each module will maintain its own data inside and output proper BCD data, blink flag and or alarm flag (in Alarm module) when they are enabled.



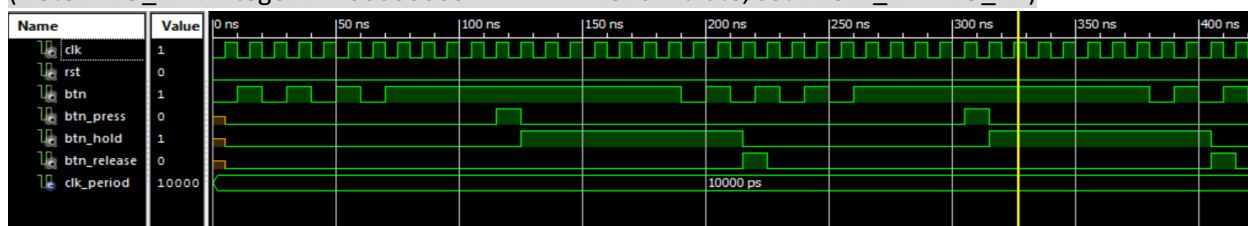
The Finite State Machine of Clock module is shown in the right side figure. For other modules' FSM diagram, please refer to the appendix. For Timer and Stopwatch modules, the behind principle are same with the above 3 modules. More information refer to source code and FSM.



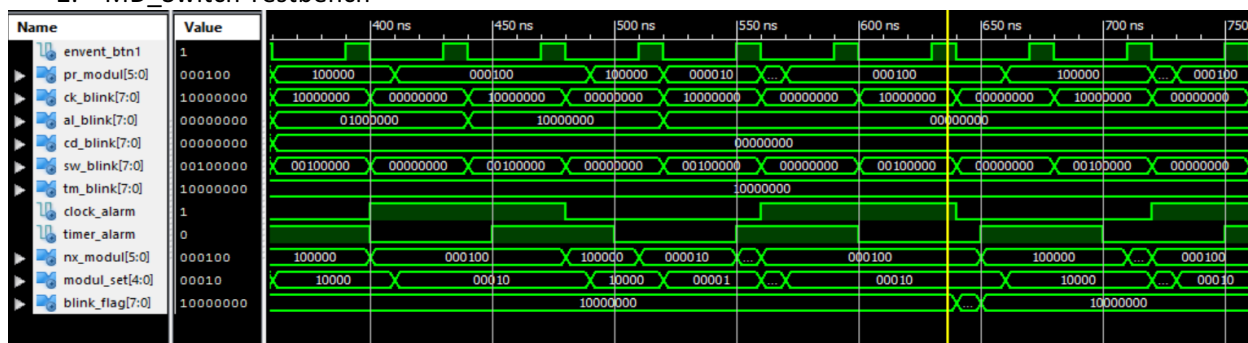
3 MODULES TESTBENCHS

1. dButton Testbench

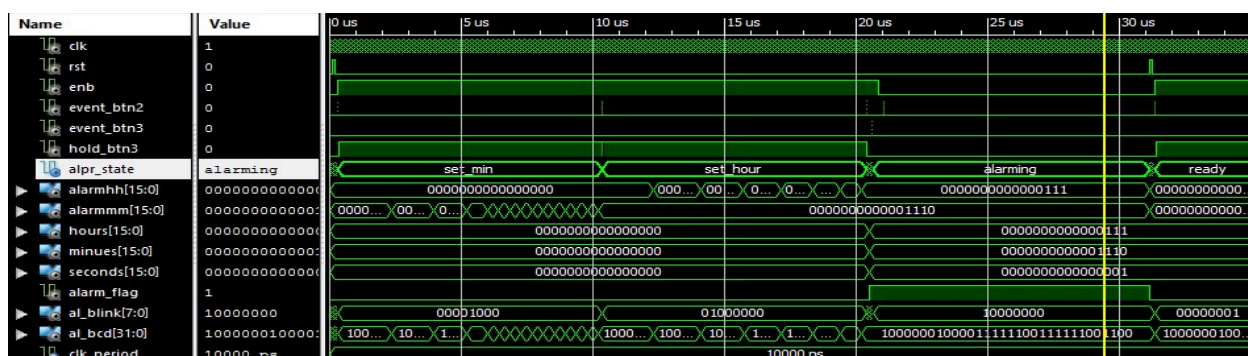
(Note: --TO_HZ : integer := 100000000 -- for simulate, set FROM_HZ = TO_HZ)



2. MD_Switch Testbench



3. Alarm_MD testbench



4. MD_Blink testbench



Other testbenches please refer to our source code.

4 INSTRUCTIONS MANUAL

We will use three buttons (**Btn1**, **Btn2**, and **Btn3**), that are used to change the mode, set the time, set the alarm, set date, start/stop the stopwatch, set/start timer and so on, to implement all required functions of the digital clock. **4-digit 7-segment LCD** will be use to display kinds of information, like date, clock, timer etc. upon the mode selected.

Pushing **Btn1** changes the mode from *Clock* to *Alarm*, *Calendar*, *Stopwatch*, *Timer*, and then back to *Clock*. **5 LEDs** will be used to indicate which mode is selected by lit on. The functions of **Btn2** and **Btn3** vary upon the mode selected by users, as below...

Operation in *Clock* mode:

The display will show the time using the format hh.mm (24hour). During this mode, Pushing Btn2 changes the state to Set Hours or Set Minutes and back to Clock Mode. When in Set Hours or Set Minutes state, a long press (hold) of Btn3 increases the hour or the minutes by 1 with incremental speed from per 1 sec to 0.2 sec.

Operation in *Alarm* mode:

The display will show the alarm time format hh.mm (24H). Pushing Btn2 changes the state to Set Alarm Hours or Set Alarm Minutes and then back to Alarm. When in the Set Alarm Hours or Set Minutes state, each long press (hold) of Btn3 increases the hour or the minutes by 1 with incremental speed from per 1 sec to 0.2 sec. When in the Alarm state, Pressing Btn3 resets the alarm. Once the alarm starts “ringing”, alarm LED will flicker for 50 seconds and then stop itself. It can also be stopped manually by pressing B3.

Operation in *Calendar* mode:

The display will show the date with the format mm.dd. During this mode. Pushing Btn2 will switch between the following states: Month, Days, Years, days of week, and back to Calendar Mode. When in Set Months or Set Days or Set Years state, or Set days of week, each long press (hold) of Btn3 increases the hour or the minutes by 1 with incremental speed from per 1 sec to 0.2 sec.

Operation in *Stopwatch* mode:

The display indicates stopwatch time in the format mm.ss. Pressing Btn2 starts the time counter (Counting state), Pressing Btn2 again stops it (Pausing state), and then pressing Btn2 resumes it (Counting state), and so on. However, during Counting state, Press Btn3, a lap time will be stored and “frozen” on the display (Lap time mode, Stopwatch continues running in the background or can be stopped) until pressed Btn3 again. During other modes, Pressing Btn3 will reset the stopwatch to zero. Once the stopwatch is started, it will keep running even when in Clock or Alarm or Calendar, and Timer mode.

Operation in *Timer* mode:

The display will show the timer format hh.mm. Pushing Btn3 changes the state to Set Hours or Set Minutes and then back to Timer. When in the Set Hours or Set Minutes state, each long press (hold) of Btn3 increases the hour or the minutes by 1 with incremental speed from per 1 sec to 0.2 sec. When in the Timer state, Pressing Btn3 start or cancel the timer. Once the timer starts, it can also be pause or resume by pressing Btn2.

5 APPENDIX

5.1 DETAIL INSIDE BLOCK DIAGRAM



SmartWatch Block Diagram V2.pdf

5.2 DETAIL FSM DIAGRAMS



smtWatch FSM for each module.pdf

5.3 VHDL SOURCE CODE



smtWatch_VHDL_Code.rar

5.4 TESTBENCH SOURCE CODE



smtWatch_testBench_Code.rar

6 REFERENCE

- [1] Hints: <https://drive.google.com/file/d/0B4tlwM6TYaj3Uml0TnZvQ0ZoRFE/view?usp=sharing>
- [2] Lesson 74 - Example 47: Debounce Pushbuttons - YouTube
https://www.google.no/url?sa=t&rct=j&q=&esrc=s&source=web&cd=4&cad=rja&uact=8&ved=0ahUKEwi360ThkuDKAhWKCywKHejbBqAQtwIIMzAD&url=https%3A%2F%2Fwww.youtube.com%2Fwatch%3Fv%3D8ISfNm9zv18&usg=AFQjCNGSWBtPzrCdZilYWGeKoQbRA7stkQ&sig2=q2AkvjzsOXiN8Se33X_wTA&bvm=bv.113370389,d.bGg
- [3] XST User Guide for Virtex-6, Spartan-6, and 7 Series Devices.
<https://drive.google.com/file/d/0B4tlwM6TYaj3Mzh3dXhKbXdQZkU/view?usp=sharing>