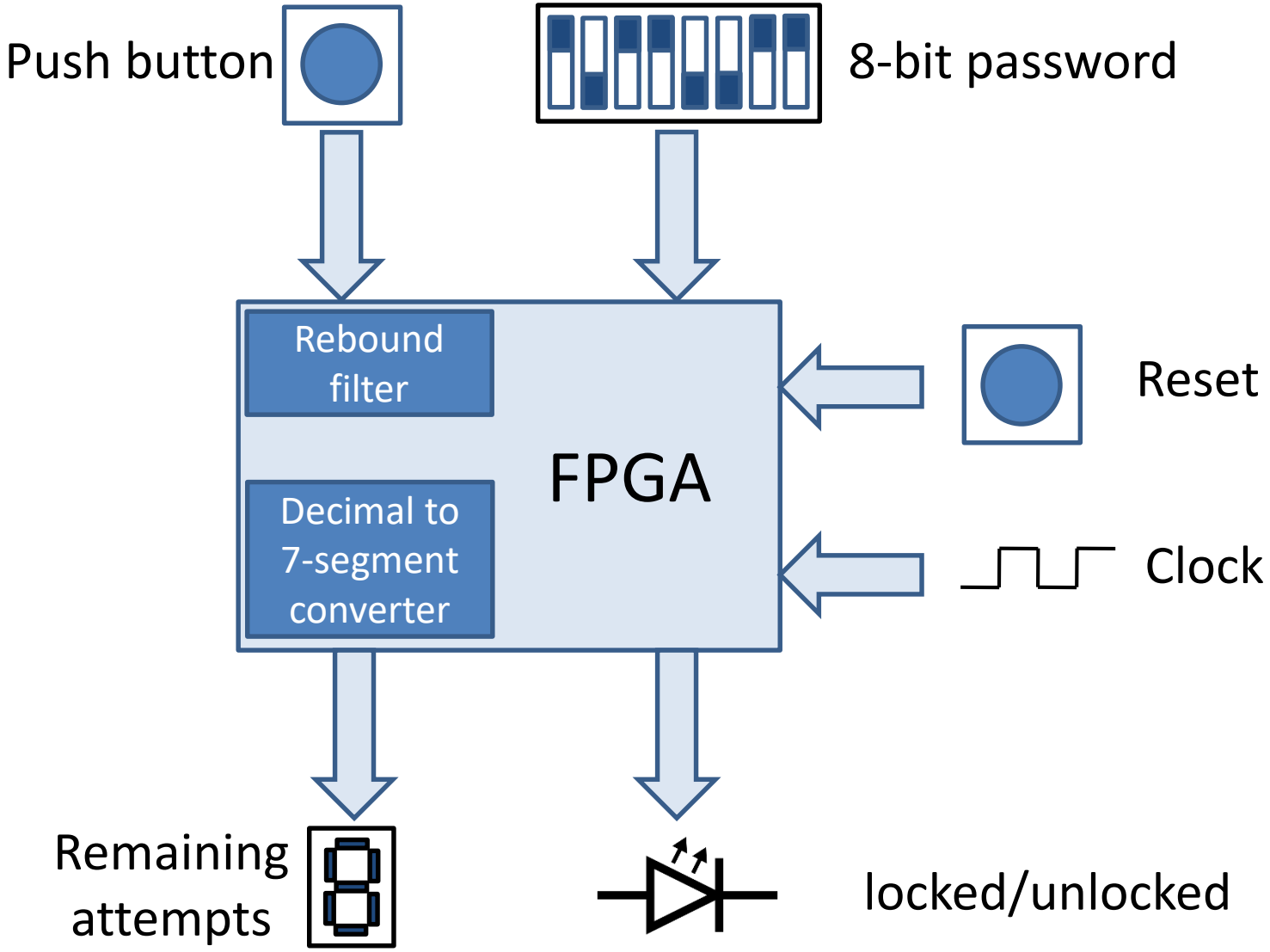
Laboratory practice 2

Finite State Machines

Objective

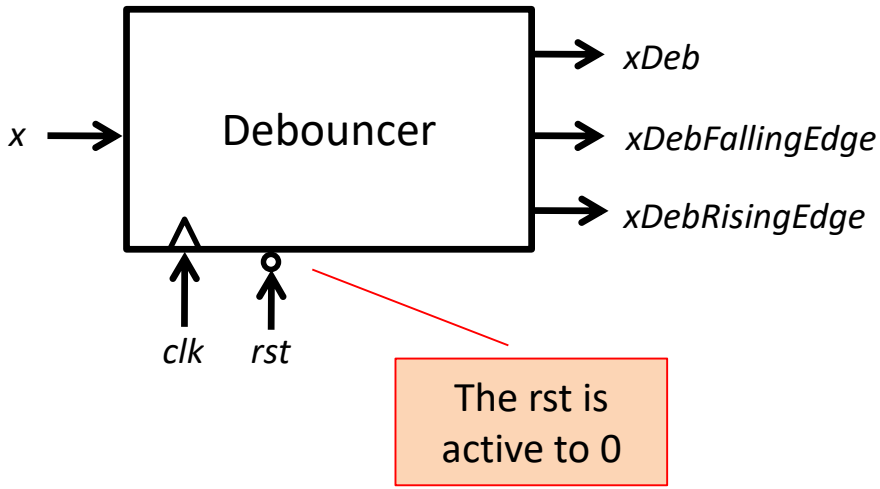
- Implement an 8 bit digital combination lock:
  - User inputs:
    - Reset, a push button and 8 switches
  - User outputs:
    - One of the 7-segment displays and a LED
  - The lock is initially open (LED on), when the push button is pressed, the key is internally stored in **a register** and the lock is closed (LED off)
  - After that, the user has 3 attempts (pressing the push button) to open the lock. The values of the register and the new input (8 switches) must be **compared** (signal **eq**)
  - The remaining attempts will be displayed in the 7-segment display
  - If the user exhausts the attempts, the lock will remain blocked indefinitely

Implementation

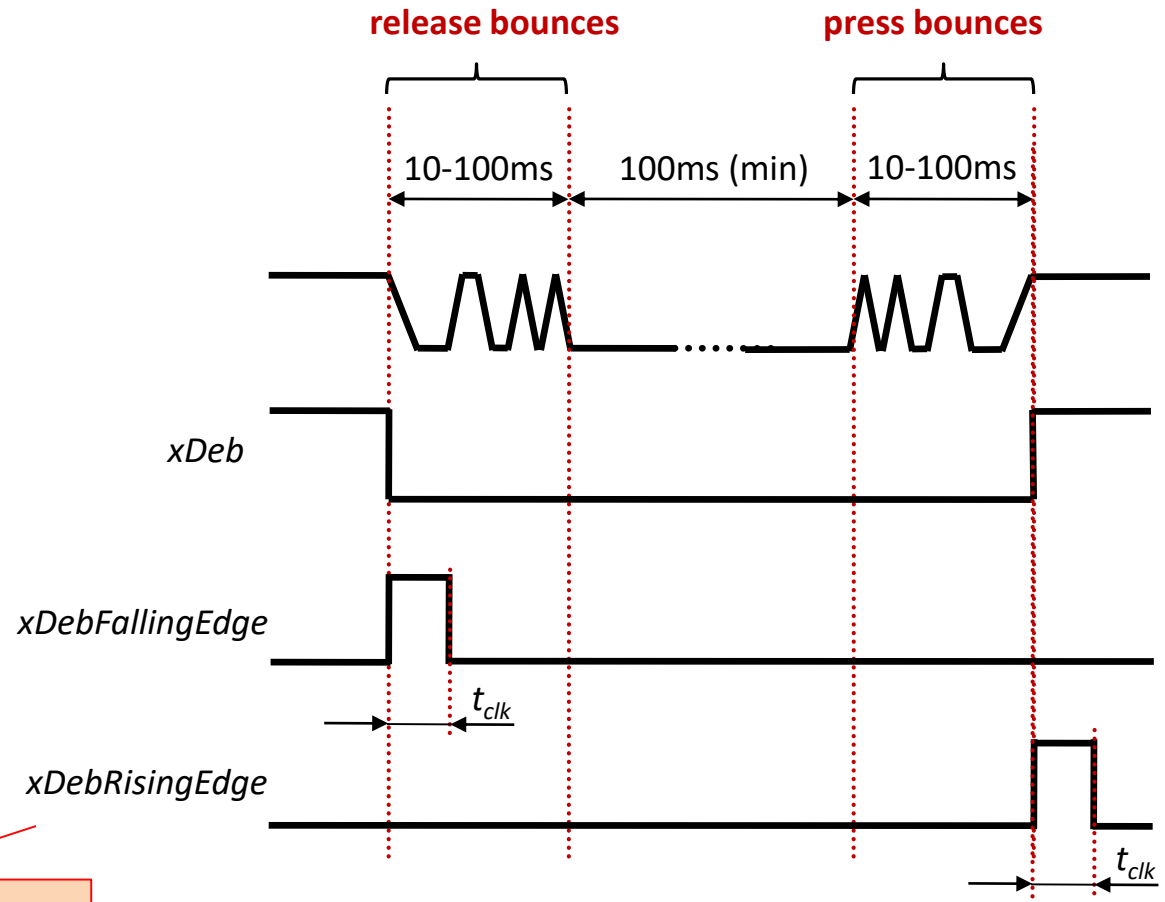


Debouncer

- Every digital signal that comes from a mechanical device (e.g. push button) has rebounds when its state changes
- A rebound filter is a circuit that removes the fast transitions that occur after each change in the state of the device
- You can find an implementation in 'debouncer.vhd'



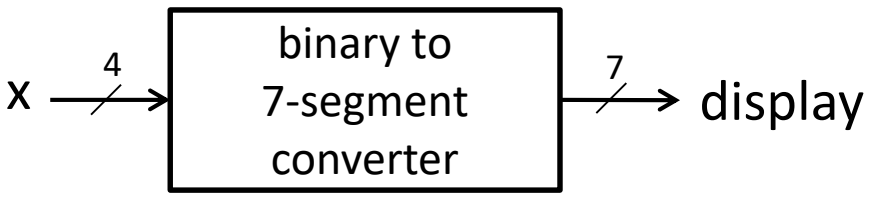
Rebound filter

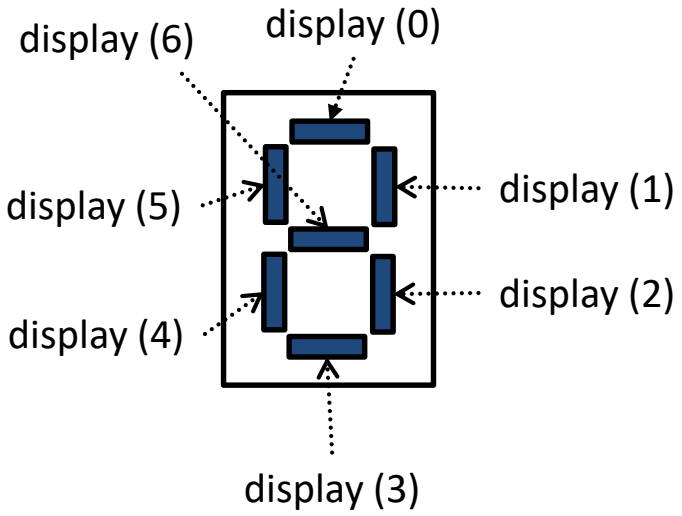


We should use this output

Decimal to 7-segment converter

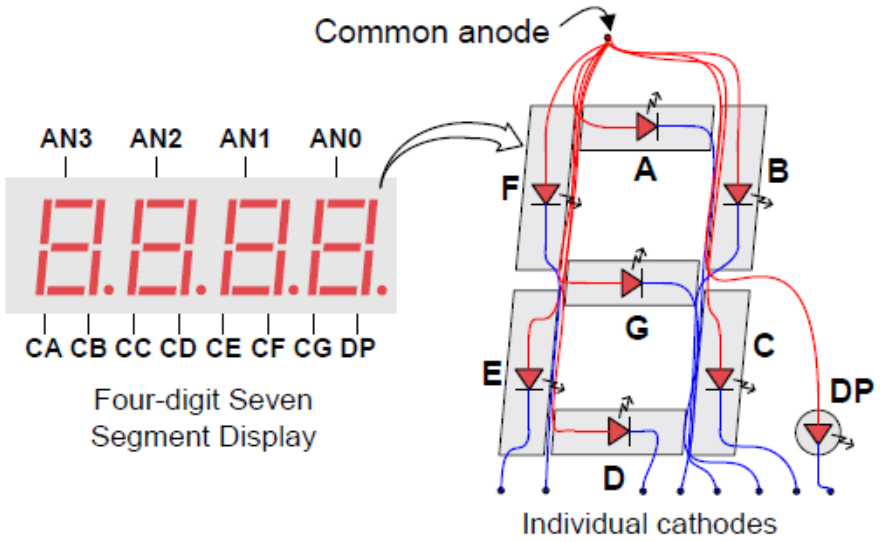
- The converter receives as input an unsigned binary value from 0 to 9 and produces the corresponding values for the 7-segment display, to show the value in a human readable form.
- You can find an implementation in 'conv_7seg.vhd'





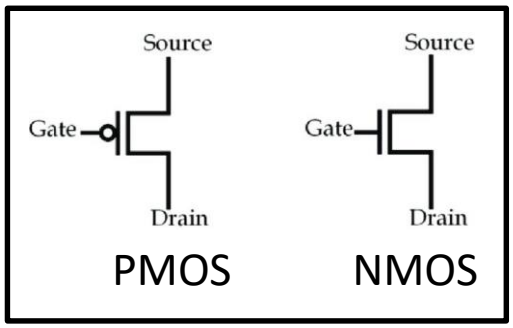
7-segment display: .xdc file

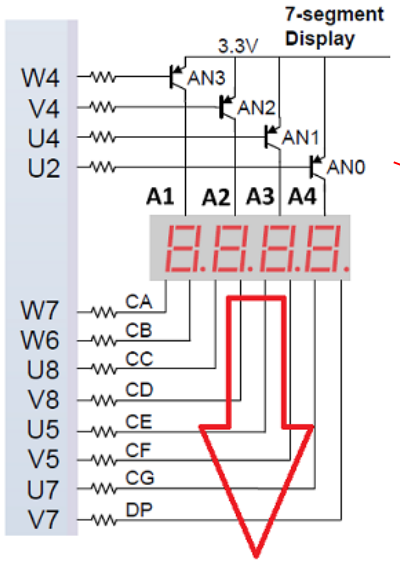
- In the Basys 3 FPGA, the 4 existing 7-segment displays can be lit separately, but they share the cathodes (CA through DP).
  - Pins AN3 – AN0 = '1' enable displays 3 – 0, respectively.
  - Pins CA-CG turn on/off the LEDs of all enabled displays.
- Therefore, when enabling several displays simultaneously, all of them will show the same value.



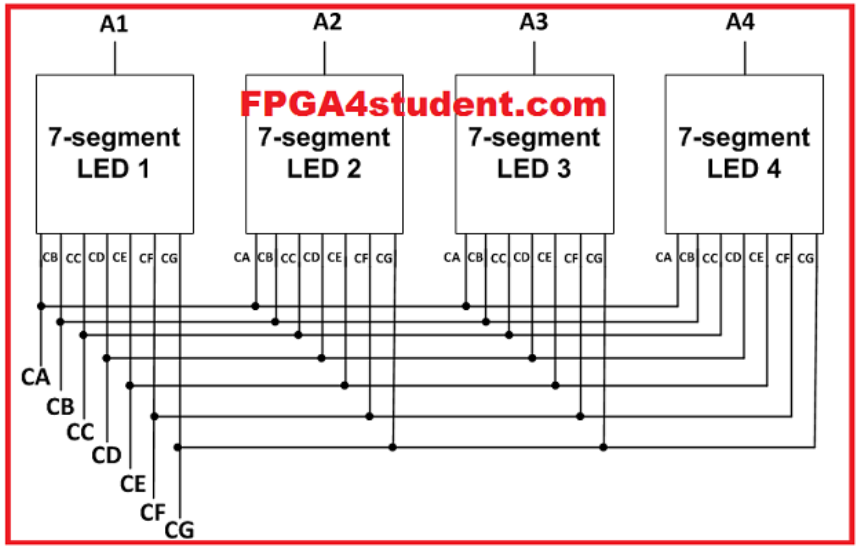
https://www.fpga4student.com/2017/09/seven-segment-led-display-controller-basys3-fpga.html

7-segment display: .xdc file





PMOS transistors: they conduct the logic level (3.3V) in the source (upper port) to the drain (lower port) if the gate (left port) is 0

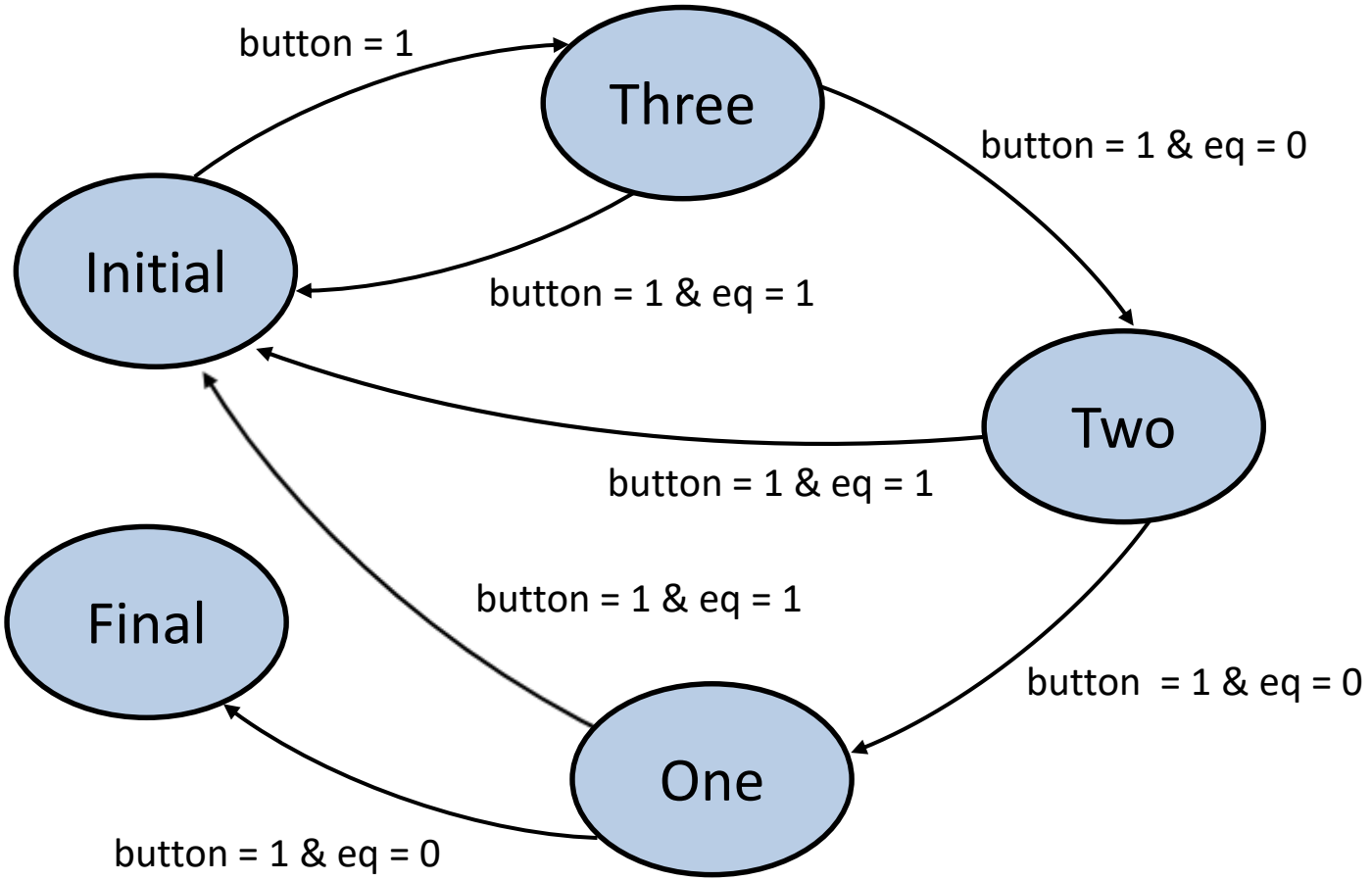


In other words, the 7-segment display is activated if the corresponding .xdc pin (W4, V4, U4 or U2) is 0

7-segment display: .xdc file

- In order to show different values in all displays simultaneously, we need to proceed as follows:
  1. Enable AN3 (**by setting pin W4 to '0' in the .xdc**), disable others, write value of Display 1 in CA-DP (pins W7, W6, U8, V8, U5, V5, U7, V7 in .xdc). Wait for 1ms-16ms.
  2. Enable AN2 (pin V4 in .xdc), disable others, write value of Display 2 in CA-DP. Wait for 1ms-16ms.
  3. Enable AN1 (pin U4 in .xdc), disable others, write value of Display 3 in CA-DP. Wait for 1ms-16ms.
  4. Enable AN0 (pin U2 in .xdc), disable others, write value of Display 4 in CA-DP. Wait for 1ms-16ms. Go back to Step 1.
- In other words, **we need to show a value at a given time in the corresponding display, then refreshing periodically**.
- We will not need this in the lab.
  - You can do it if you want. More info in the link that I made available in the Campus Virtual.

State diagram



Only the input values that lead to a transition are shown

Grading

- The student must go to the laboratory with the digital lock implemented and simulated as homework.
- The student must show the digital lock working, and have to understand the implementation and functionality:
  - If it works properly in the FPGA, 0.15 pts
  - In case of just simulation working, 0.10 pts
- Advanced part 0.15 pts (only sim. → 0.10 pts)
- Laboratory practice 2 is **NOT** recoverable