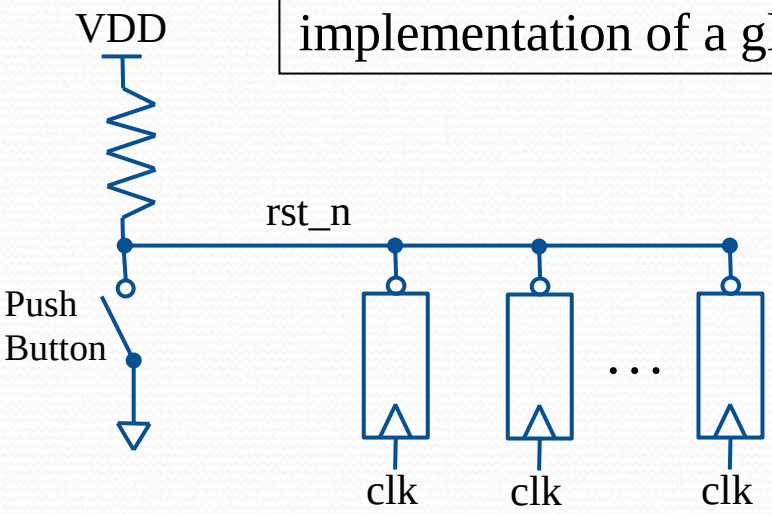
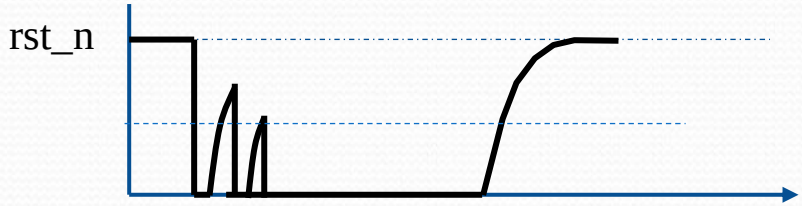
Exercise 15: Testing nonoverlap on DE-0 Na...

Please work in groups of 2

- On the FPGA board we have a couple of push buttons. We will use one as the source for our asynchronous reset.
- It is simply a momentary push button switch to ground with a pull-up resistor.

Imagine what a disaster this implementation of a global reset would be!!





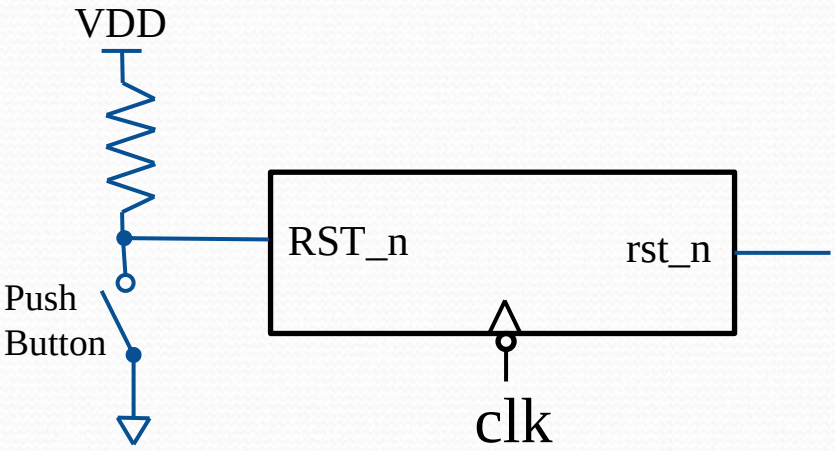
Button pushed (has bounce)

Button released (RC approach)

When does reset deassert relative to clock?

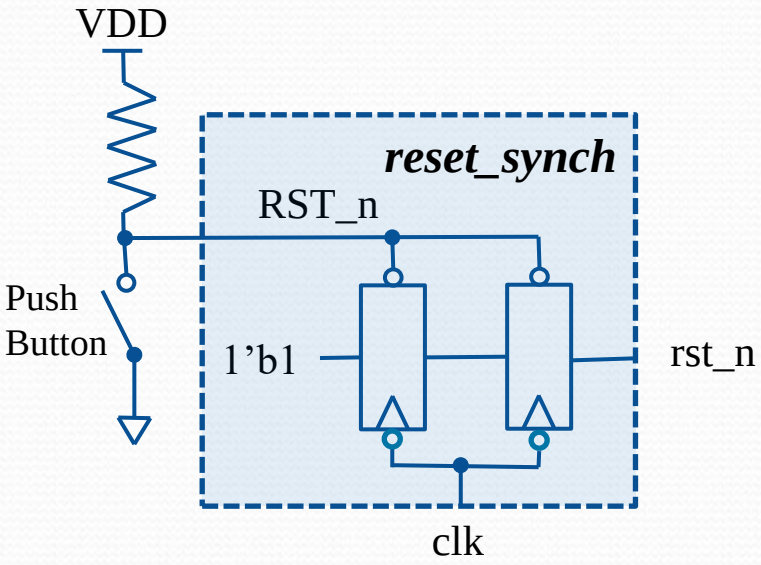
Remember…you want your reset de-asserted on the opposite edge of clock that your other flops are active on. This means we want our reset to de-assert (rise) on negative edge of clock.

Exercise 15: Testing nonoverlap on DE-0 Nano



We want to build a reset synchronizer that takes in the raw push button signal and creates a signal that is deasserted at the negative edge of clock.

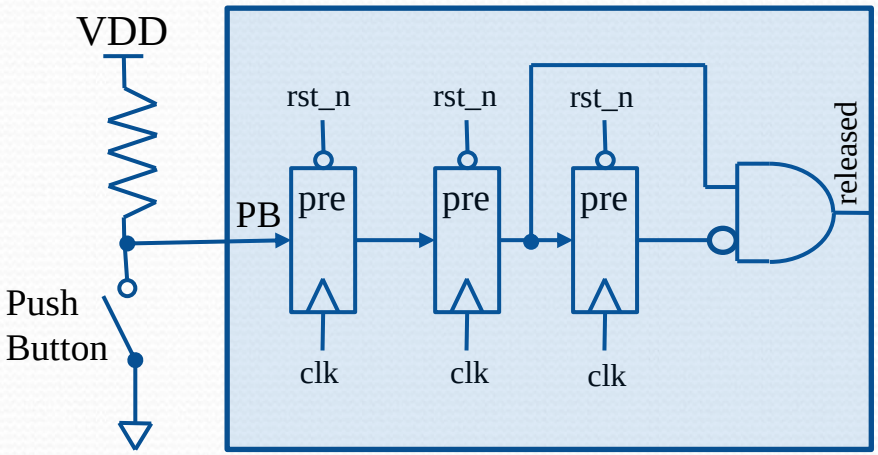
It will have an interface of:
RST_n = raw input from push button
clk = clock, and we use negative edge
rst_n = our syncrhronized output which will form the global reset to the rest of our chip.



A push of the button will asynch reset the two flops. When button is released we have a double flopping *(meta-stability reasons)* to produce our global **rst_n**. The flops are **negative** edge triggered so our global reset will deassert on the opposite edge of all our other flops.

Code this reset synch unit (**reset_synch.sv**)

Exercise 15 (Synching & Edge Detecting a PB input):



PB_release.sv

NOTE: these flops are asynch preset not reset

Study this…does it make sense?

- The first two flops are just double flopping for meta-stability purposes. PB is asych input right?
- Third flop is used to implement a rising edge detector. If the input to 3rd flop is high, but it output is low then a rising edge must be coming through (i.e. the release of the button.)

Implement this in system Verilog. Call it: **PB_release.sv**

Don't worry about testing it yet

Exercise 15: Testing of nonoverlap on DE0

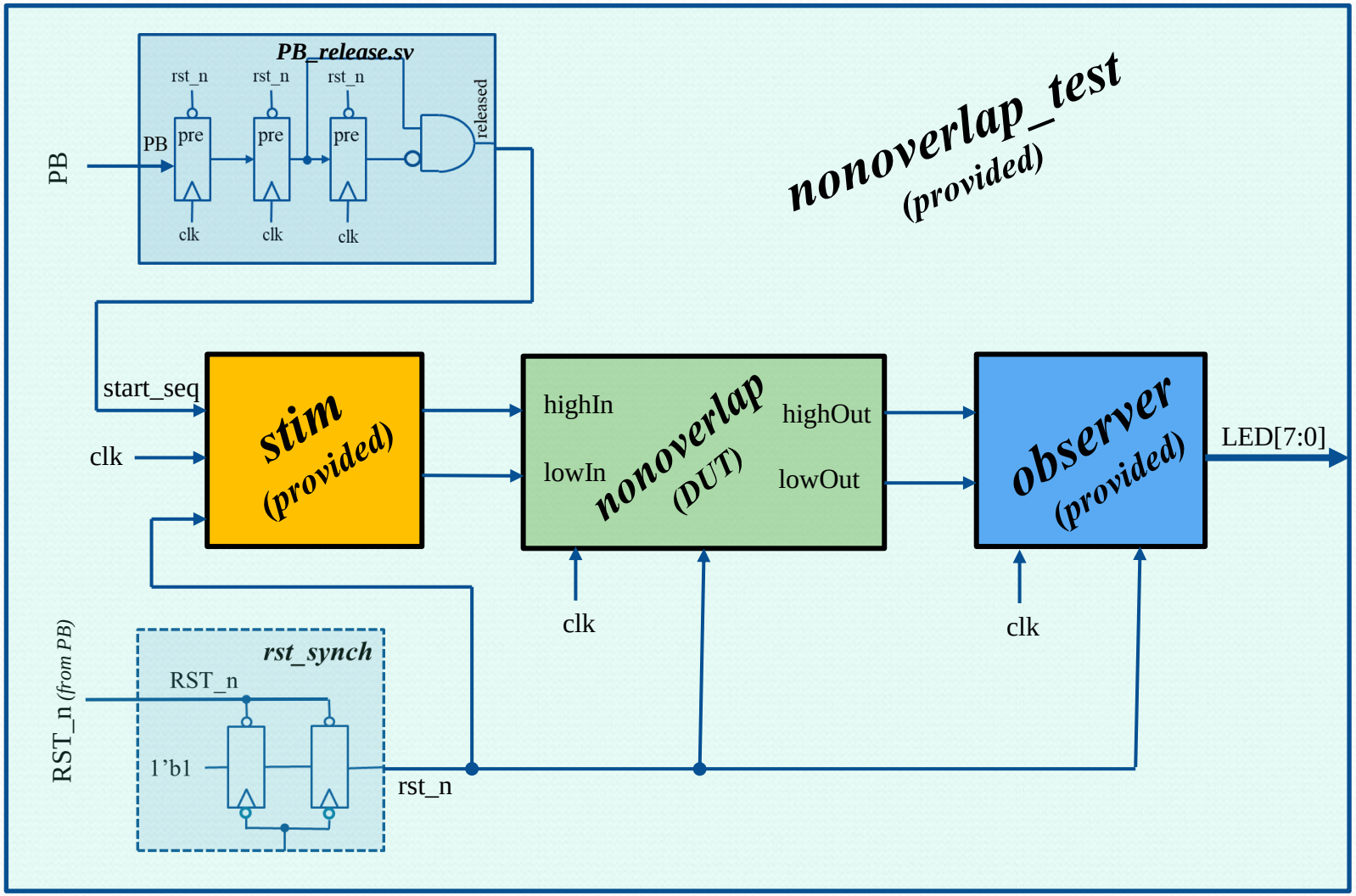
Please work in groups of 2

- In Exercise10 you coded **nonoverlap.sv** which has the interface shown:

Signal:	Dir:	Description:
clk, rst_n	in	50MHz clock, and asynch active low reset
highIn	in	Control for high side FET
lowIn	in	Control for low side FET
highOut	Out	Control for high side FET with ensured non-overlap
lowOut	Out	Control for low side FET with ensured non-overlap

- Now you are going to build a test wrapper (**non_overlap_test.sv**) for this that will be mapped to a DE0-Nano board and used to test it.
- The test wrapper (**nonoverlap_test.sv**) will contain two provided sub-modules (***stim*** & ***observer***).
- The ***stim*** block will simply wait for a button push and then start generating stimulus into your ***nonoverlap*** block (driving signals **highIn** & **lowIn**).
- The ***observer*** block will monitor the signals **highOut** & **lowOut** and ensure they have no overlap and have the specified deadtime (32 clk cycles). It will drive the LEDs on the board to let you know its "observations"
  - LED = 0x99 after reset
  - LED = 0xAA if no errors found, otherwise 0xEE

Exercise 15: Testing of nonoverlap on DE0

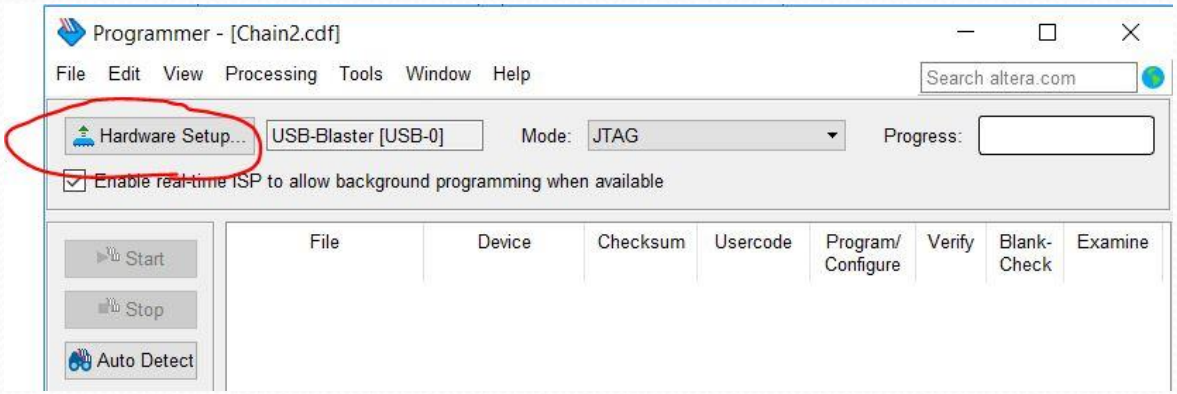


Once you are successfully compiling in Quartus call Winor or Prof. Ramanathan over to demo

Exercise 15: Testing of nonoverlap on DE0

- Download **nonoverlap_test.qpf** (Quartus Project File) and **nonoverlap_test.qsf** (Quartus Settings File) from the website and store in your Exercise15 directory

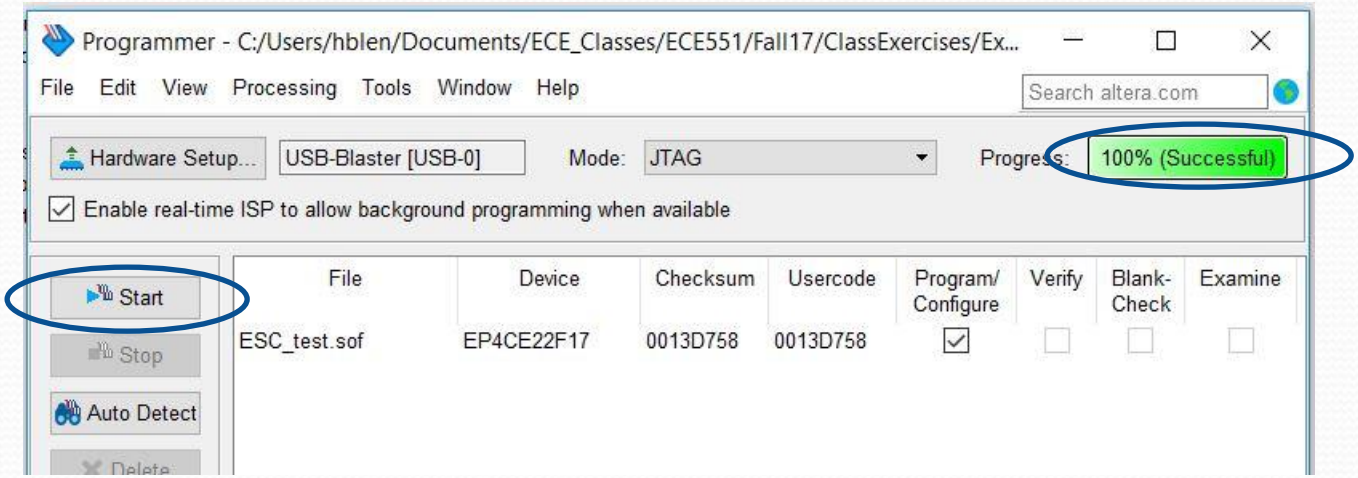
- Open up Quartus
  - Do a: **File → Open Project** and open up the **nonoverlap_test.qpf**
  - Compile the design and fix any errors
  - Call Winor or Prof. Ramanthan over to demo
  - Do a: **Tools → Programmer** and check that the USB Blaster shows up (see below) (you may have to wait a while on these CAE machines for it to enumerate)



Might have to go under "Hardware Setup" to get it to choose USB-Blaster

Exercise 15: Testing of nonoverlap on DE0

- Program the DE0-Nano

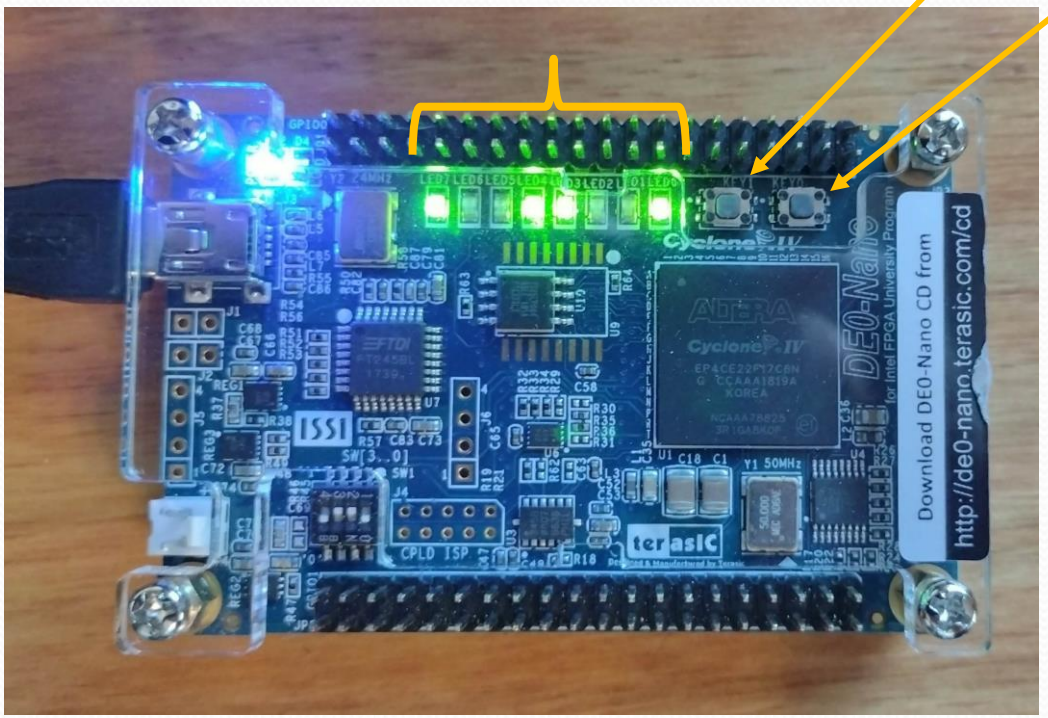


- Hit “Start” and look for 100% Success

- See next page for mapping of functions to DE0-Nano

Exercise 15: Testing of nonoverlap on DE0

LED will display:
0x99 after reset
0xAA after PB if all good
0xEE after PB if overlap issue found



"PB" push button

"RST_n" push button

Test your design.

LEDs should reset to 0x99 as shown here

When you push and release the PB it should display either 0xAA (all good) or 0xEE (overlap found).