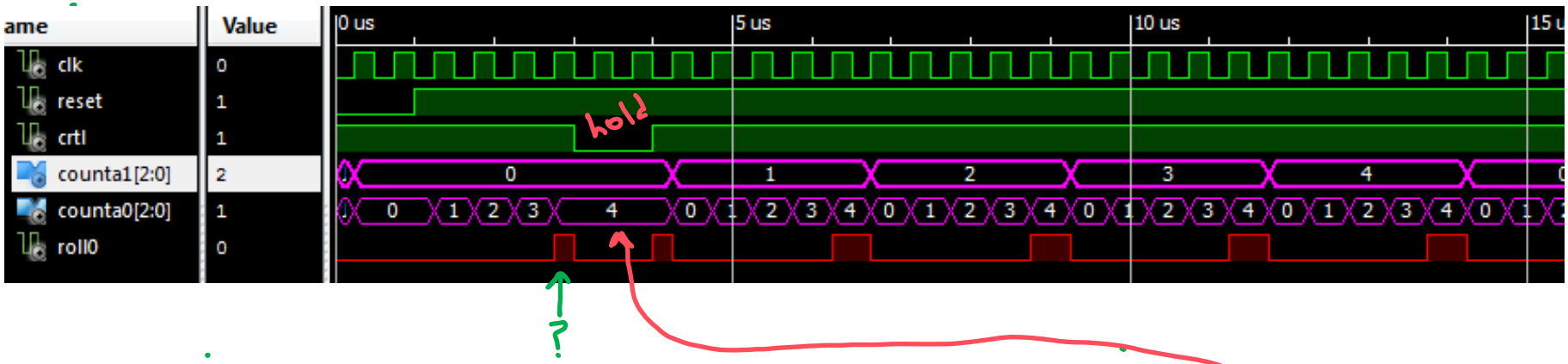
-Run lab#1 on 2nd monitor

UNITED STATES AIR FORCE ACADEMY

ECE 383 – Embedded Computer Systems II
Lecture 5 – Combination of Elements and Lab Intro

Official Grades?
Copy Lab 1 files?
HDMI Monitor?

Homework 3 and 4?



- HW4: count 00 to 44, then roll over

roll

00 01 02 03 04 10 11 12 13 14 20 21

40 41 42 43 44 00

- hold works?
- reset works?

00 01 02 03 04 hold 04 10 11

UNITED STATES AIR FORCE ACADEMY

Lesson Outline

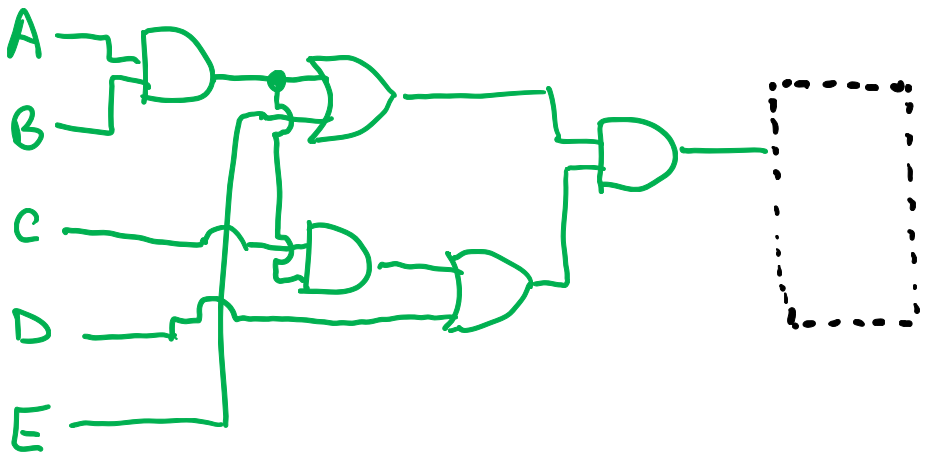
1. **Comparator Construction**
2. **Gated and Non-Gated Circuit**
3. **Lab 1 Intro**

L5	Combinations of elements, lab intro	7.2	HW #5	BOC L6
L6	Lab1 - VGA Synchronization		Gate Check 1	COB L6
L7	Lab1 - VGA Synchronization		Gate Check 2	COB L7
L8	Lab1 - VGA Synchronization		Lab1 Functionality	COB L8
L9	Finite State Machines	10.2.1, 10.3.2, 10.4, 10.6.1	Lab1 Write-up HW #6	COB L9 BOC L10

Glitches

Datasheet

	T_{CD}	T_{PD}
AND	20ps	45ps
OR	15ps	30ps



Suppose A and D change simultaneously, but logically the output should not change

output | WANT:

GET:

Contamination Delay?

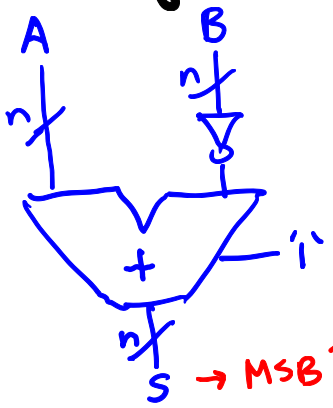
Propagation Delay?

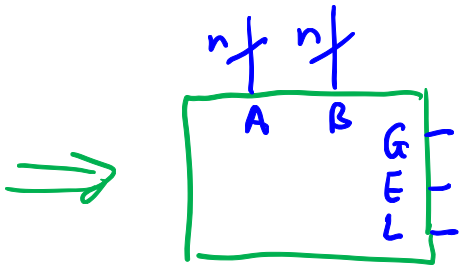
Solution?

Comparator Construction

E <= '1' when A = B else '0';

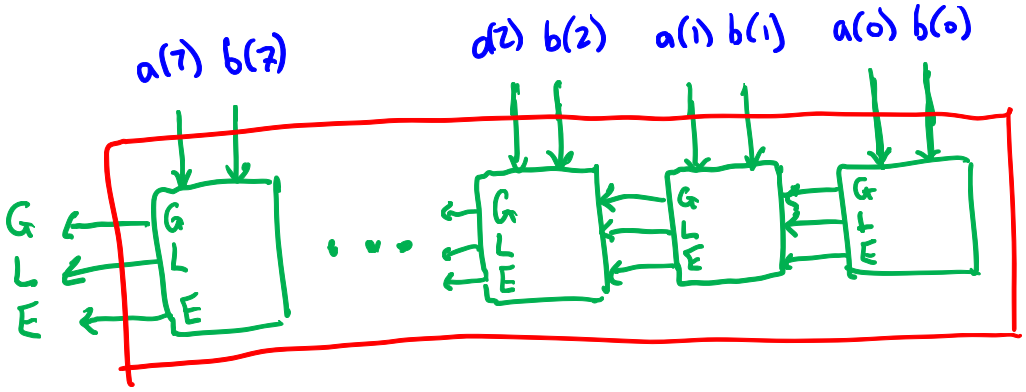
- How do you make a comparator?





→ MSB → 0 positive: A>B
→ 1 negative: A<B
→ ALL Zeros: A=B

- Bit SLICE → Cascade



RIPPLE

UNITED STATES AIR FORCE ACADEMY

Comparator Construction

- You will generate a signal similar to this CSA in Lab 1:

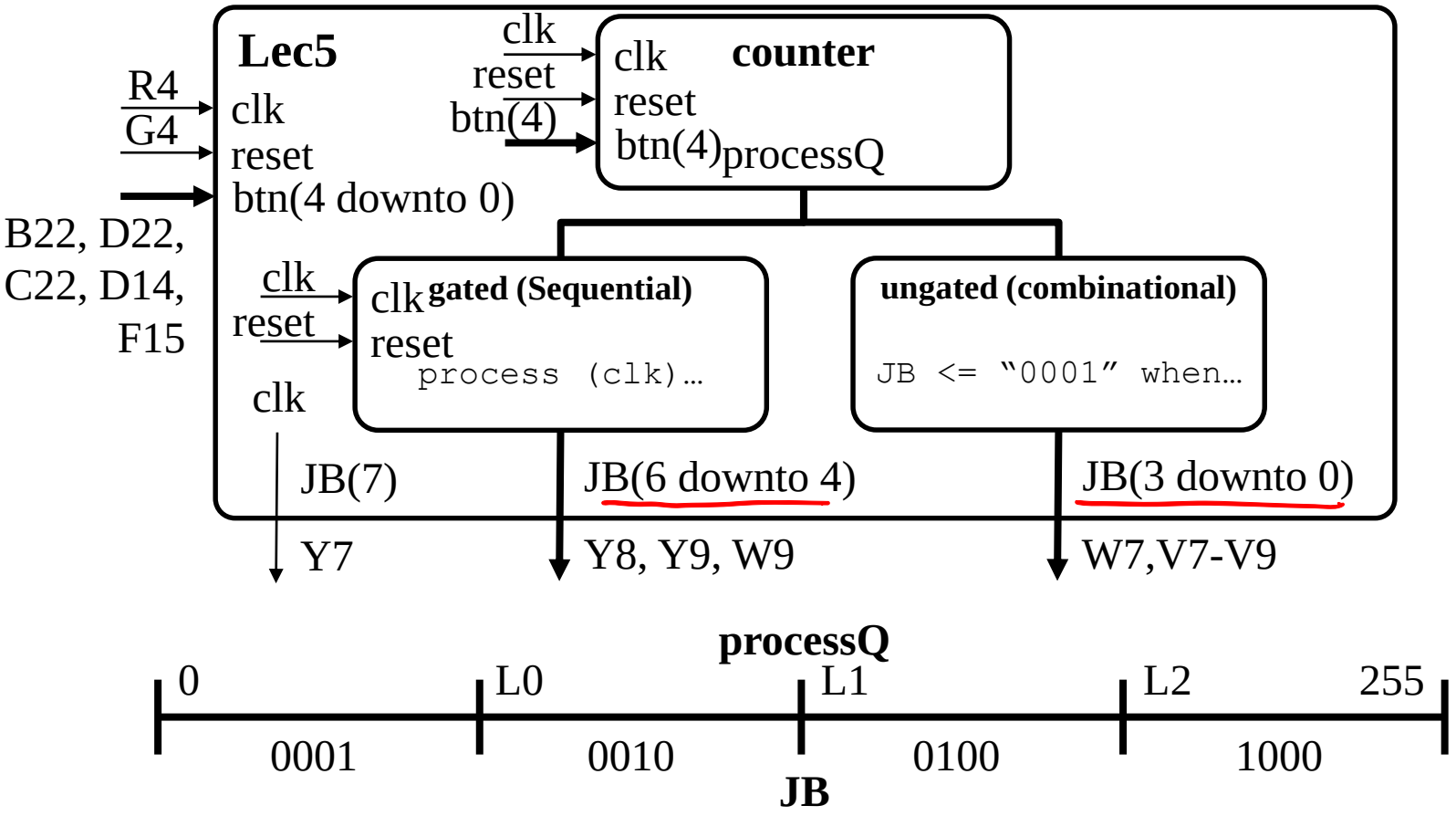
  h_blank <= '1' when ((h_count >= 100) and (h_synch < 200)) else '0';

  - This is a Non-Gated output Signal
- Non-Gated signals Generate glitches on the output!
- ~~How is a comparator constructed?~~

Gated and Non-Gated Circuit

Gated and Non-Gated Circuit

Take a look at lec05.vhdl



UNITED STATES AIR FORCE ACADEMY

Gated and Non-Gated Circuit – PMOD Connector

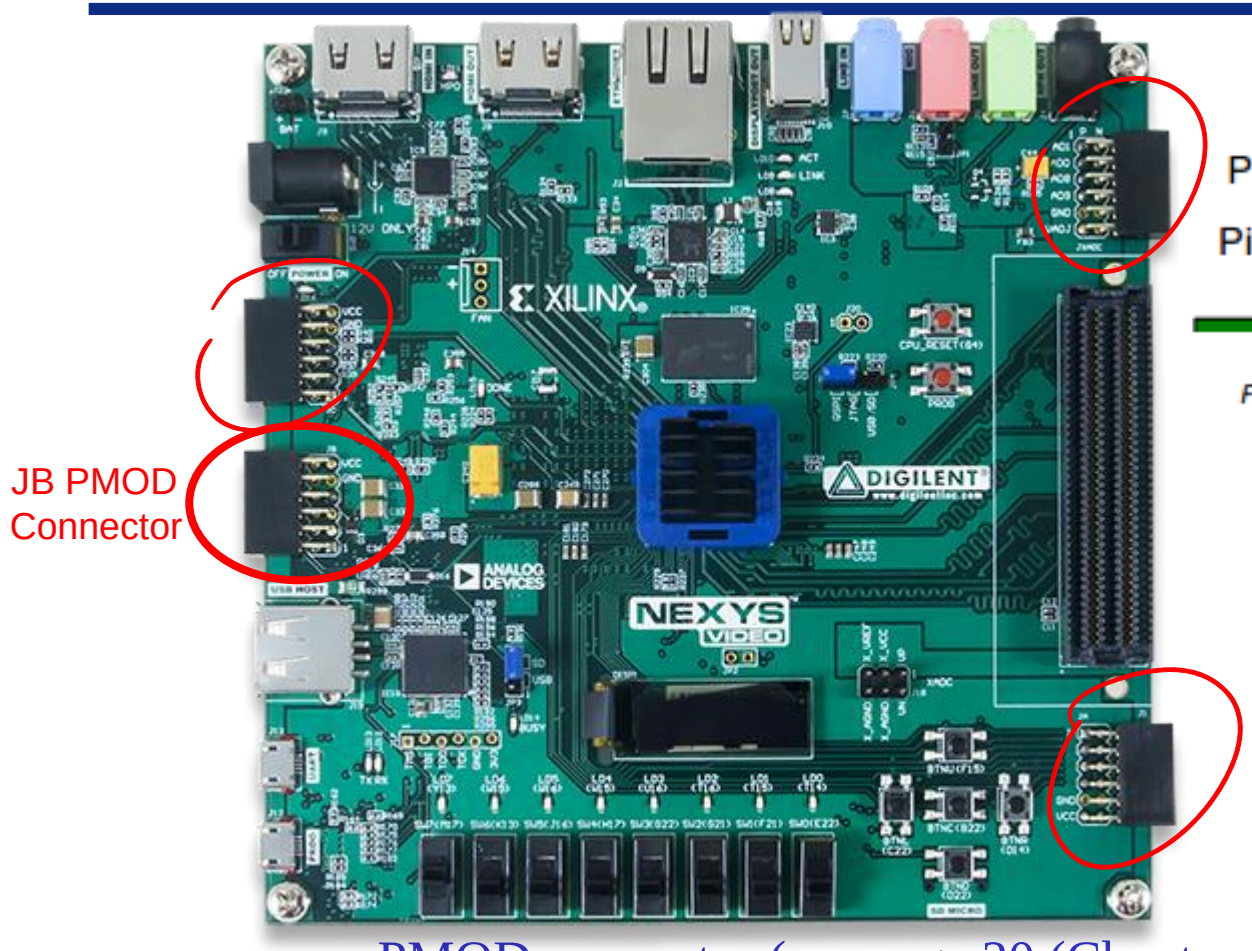


Figure 12. Pmod ports: front view as loaded on PCB.

PMOD connector (see page 20 (Chapter 10) of the Nexys Video Board Reference Manual)

Gated and Non-Gated Circuit – PMOD Connector

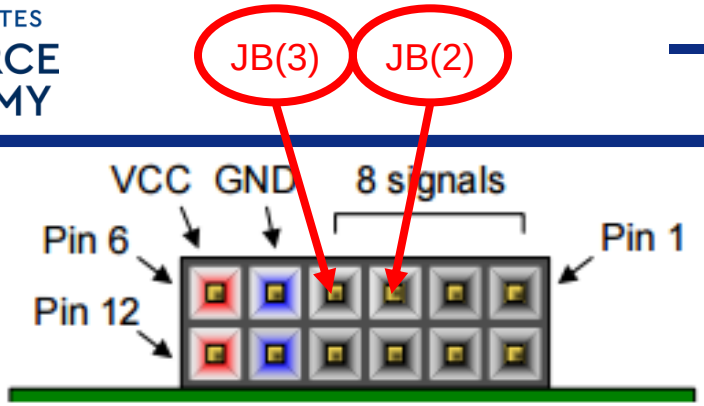


Figure 12. Pmod ports: front view as loaded on PCB.

PMOD Pinouts			
JA1: AB22	JB1: V9	JC1: Y6	JXADC1: J14
JA2: AB21	JB2: V8	JC2: AA6	JXADC2: H13
JA3: AB20	JB3: V7	JC3: AA8	JXADC3: G15
JA4: AB18	JB4: W7	JC4: AB8	JXADC4: J15
JA7: Y21	JB7: W9	JC7: R6	JXADC7: H14
JA8: AA21	JB8: Y9	JC8: T6	JXADC8: G13
JA9: AA20	JB9: Y8	JC9: AB7	JXADC9: G16
JA10: AA18	JB10: Y7	JC10: AB6	JXADC10: H15

JB(2)
JB(3)

Table 10. Nexys Video Pmod pin assignments.

PMOD connectors (see page 20 (Chapter 10) of the Nexys Video Board Reference Manual) corresponding to JB(3) and JB(2) (the most 2 significant bits of the non-gated comparator outputs)

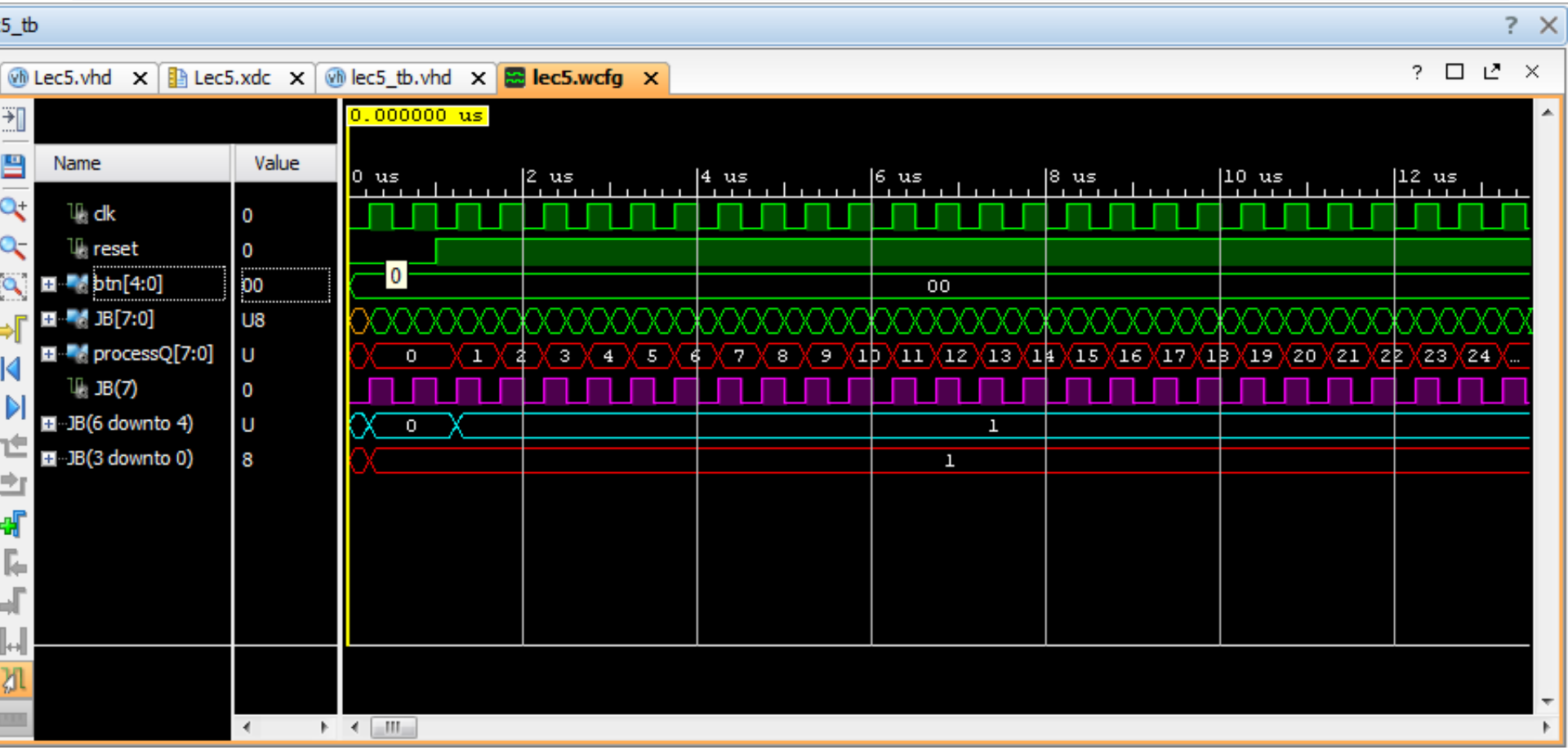
Gated and Non-Gated Circuit

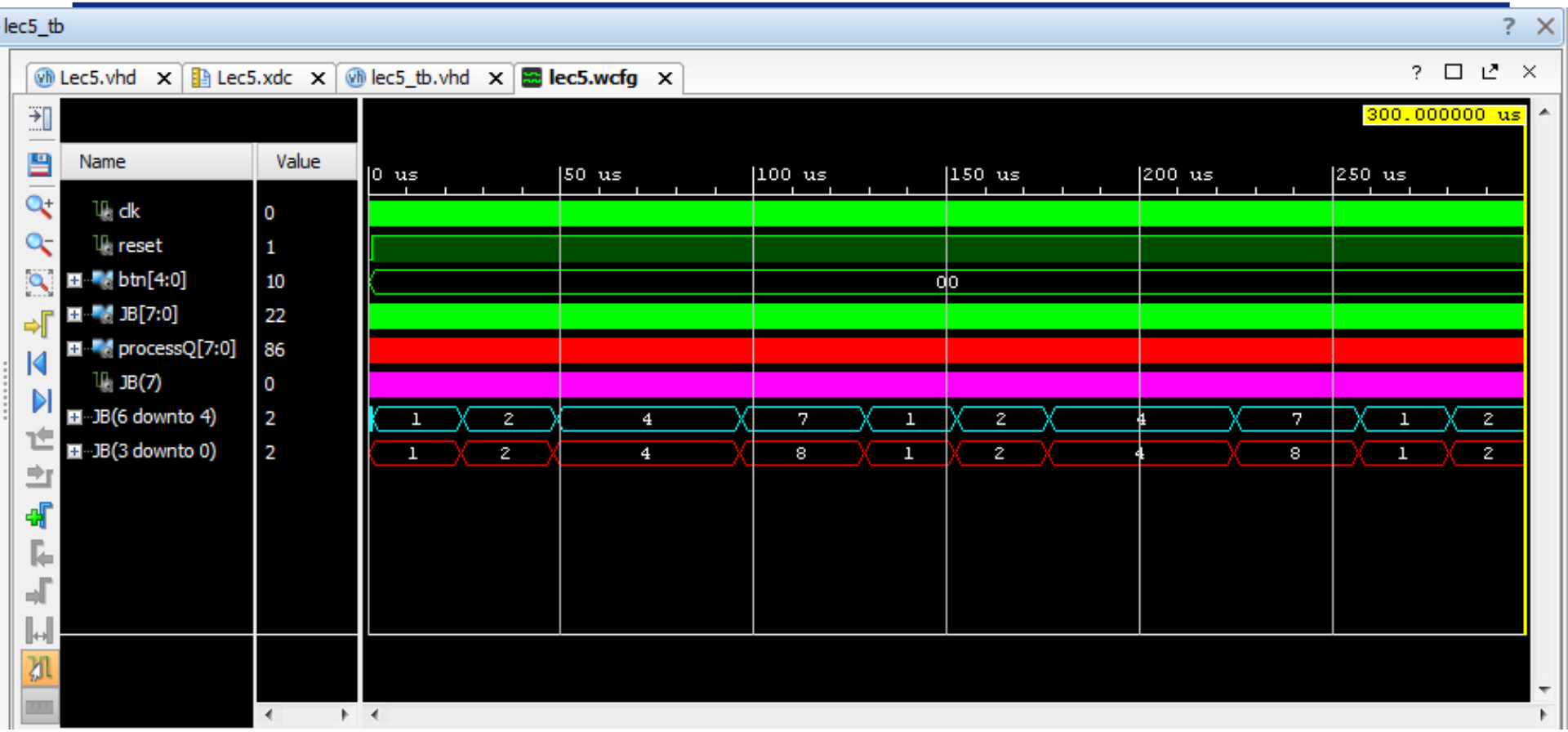
Combinational Realization – Non-gated

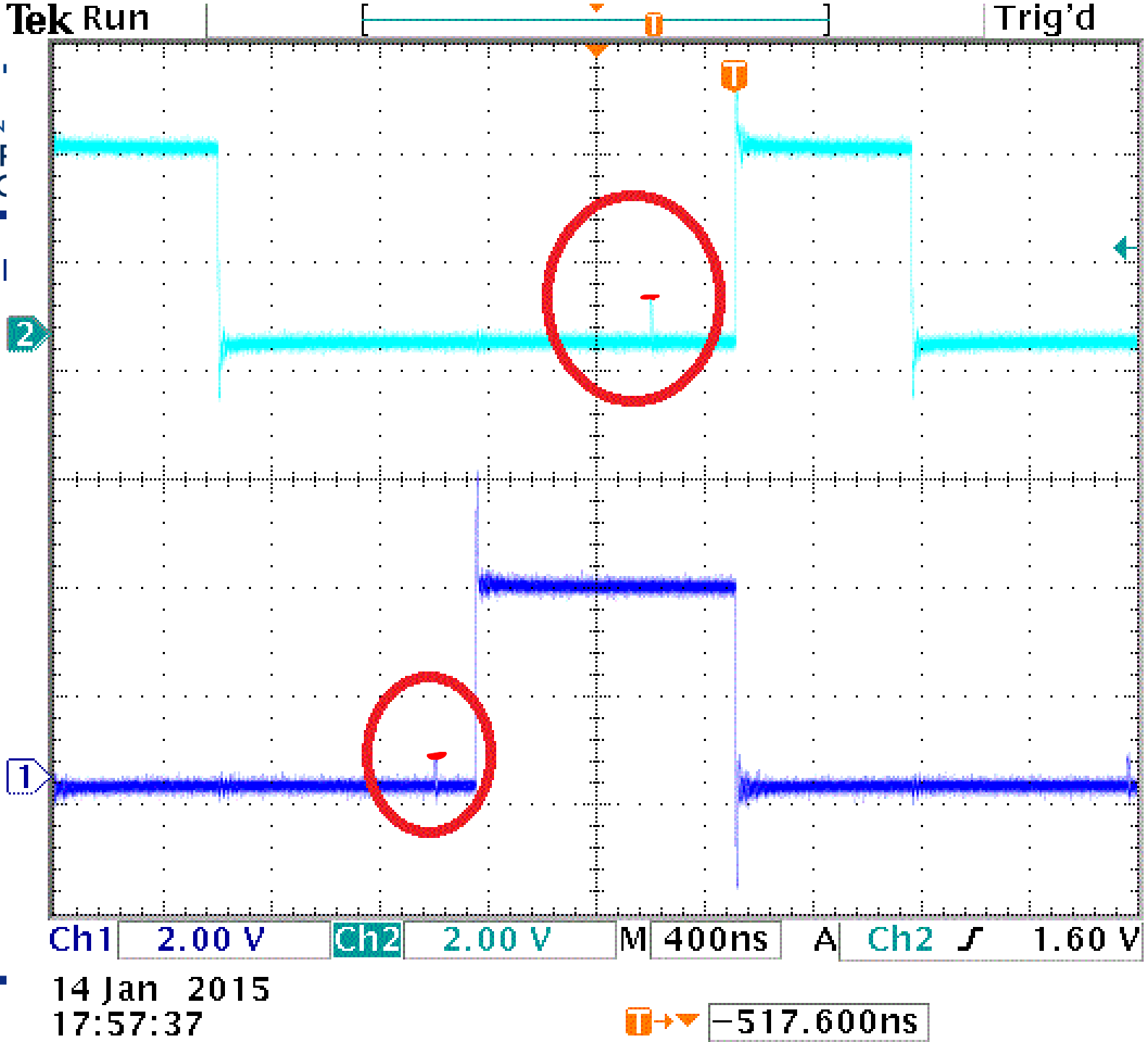
```
JB(3 downto 0) <=   "0001" when ((processQ >= 0) and (processQ < L0)) else
                    "0010" when ((processQ >= L0) and (processQ < L1)) else
                    "0100" when ((processQ >= L1) and (processQ < L2)) else
                    "1000";
```

Sequential Realization – Gated

```
process(clk)
begin
   if (rising_edge(clk)) then
            if (reset = '0') then
               JB(6 downto 4) <= "000";
            elsif ((processQ >= 0) and (processQ < L0)) then
               JB(6 downto 4) <= "001";
            elsif ((processQ >= L0) and (processQ < L1)) then
               JB(6 downto 4) <= "010";
            elsif ((processQ >= L1) and (processQ < L2)) then
               JB(6 downto 4) <= "100";
            elsif (processQ >= L2) then
               JB(6 downto 4) <= "111";
            end if;
```

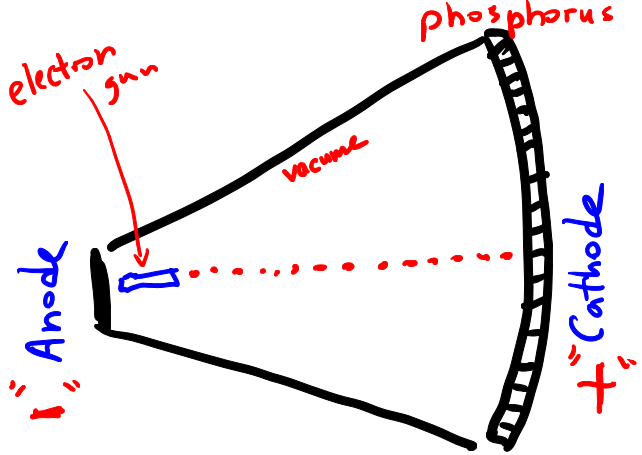

5_tb
Lec5.vhd
Lec5.xdc
lec5_tb.vhd
lec5.wcfg
0.000000 us
Name
Value
clk
0
reset
0
btn[4:0]
00
JB[7:0]
U8
processQ[7:0]
U
JB(7)
0
JB(6 downto 4)
U
JB(3 downto 0)
8
0 us
2 us
4 us
6 us
8 us
10 us
12 us
0
00
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
0
1
1

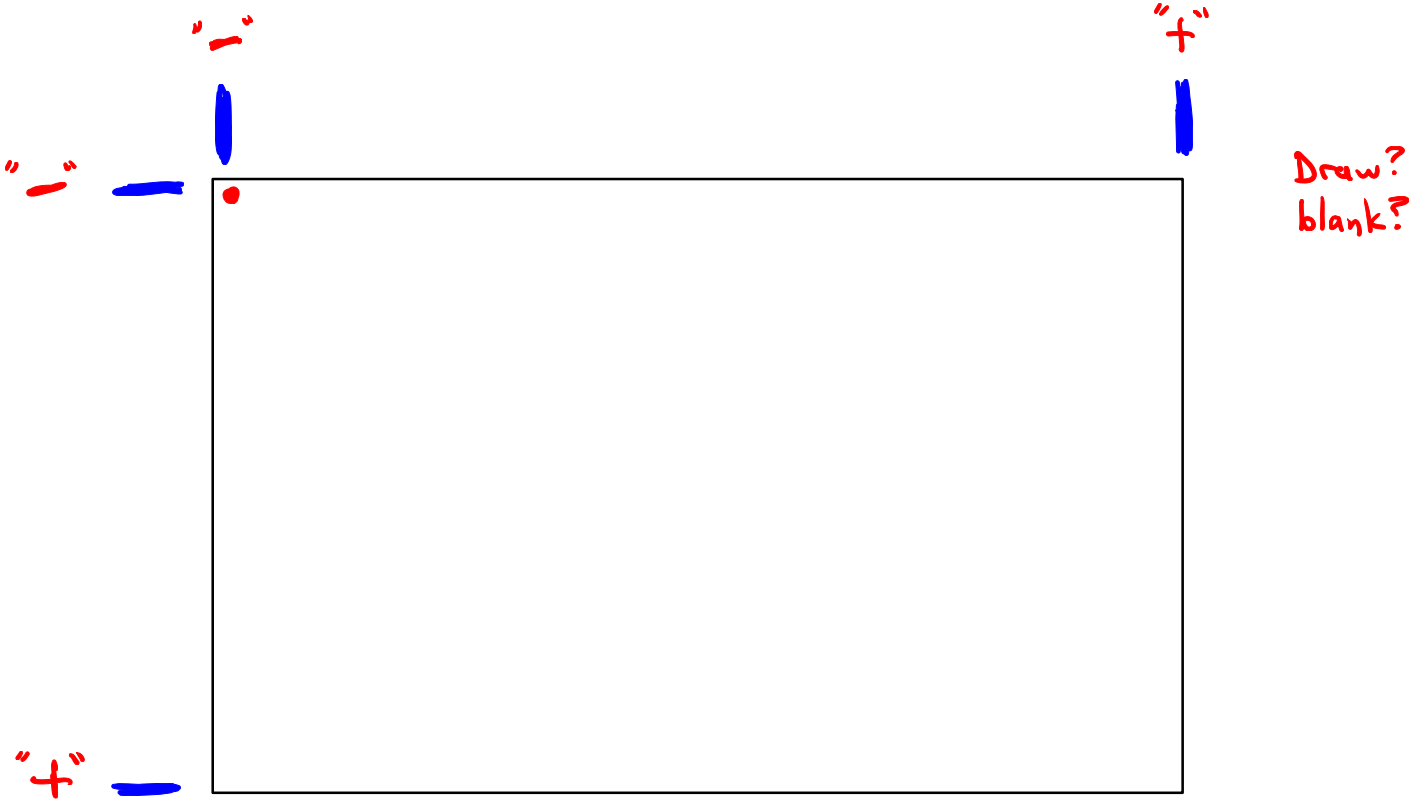

lec5_tb
Lec5.vhd
Lec5.xdc
lec5_tb.vhd
lec5.wcfg
Name
Value
clk
0
reset
1
btn[4:0]
10
JB[7:0]
22
processQ[7:0]
86
JB(7)
0
JB(6 downto 4)
2
JB(3 downto 0)
2
300.000000 us
0 us
50 us
100 us
150 us
200 us
250 us
00
1 2 4 7 1 2 4 7 1 2
1 2 4 8 1 2 4 8 1 2


Tek Run
Trig'd
Ch1 2.00 V
Ch2 2.00 V
M 400ns
A Ch2 1.60 V
14 Jan 2015
17:57:37
-517.600ns

Lab 1 Intro

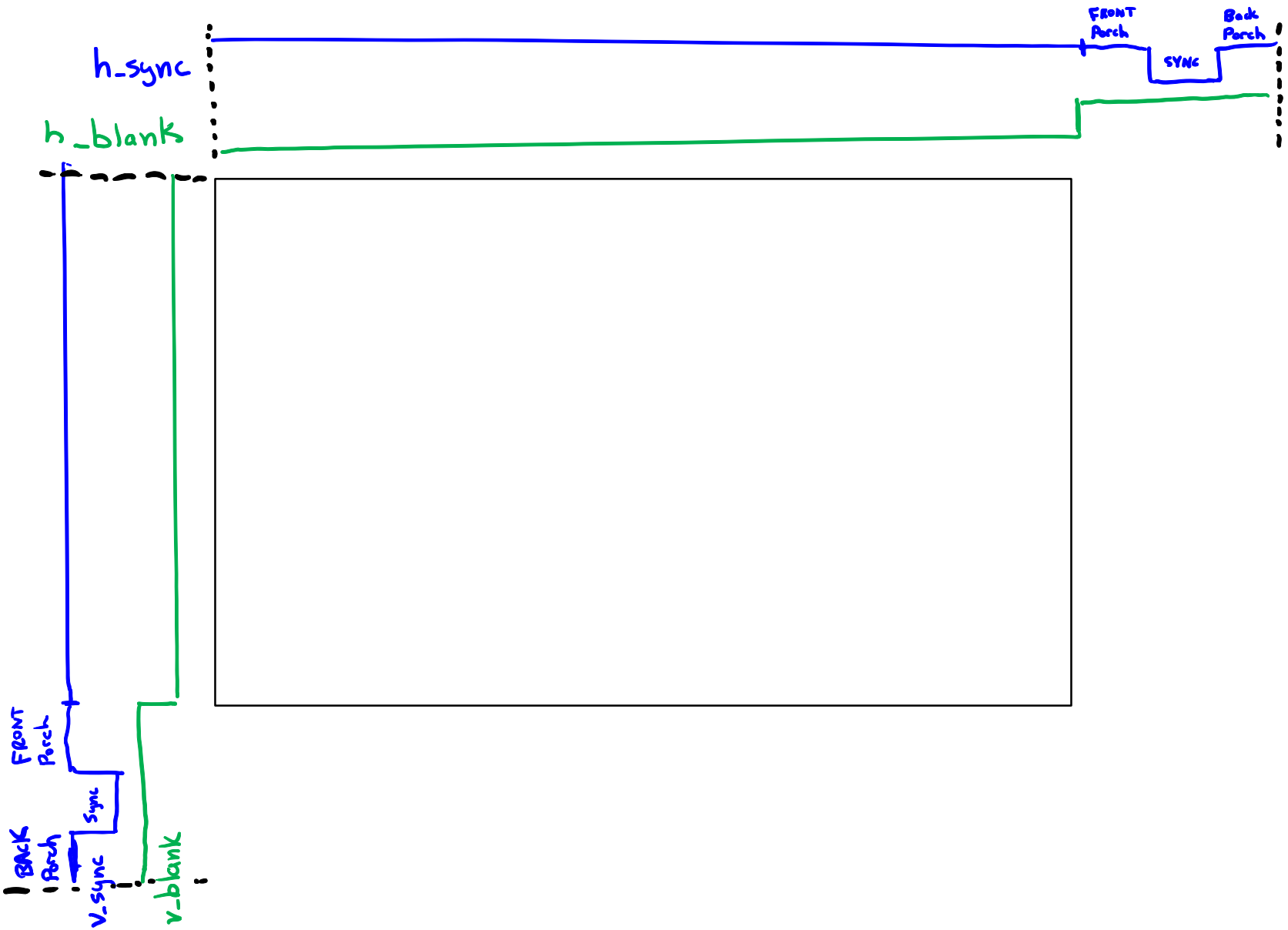
Analog TV -- CRT



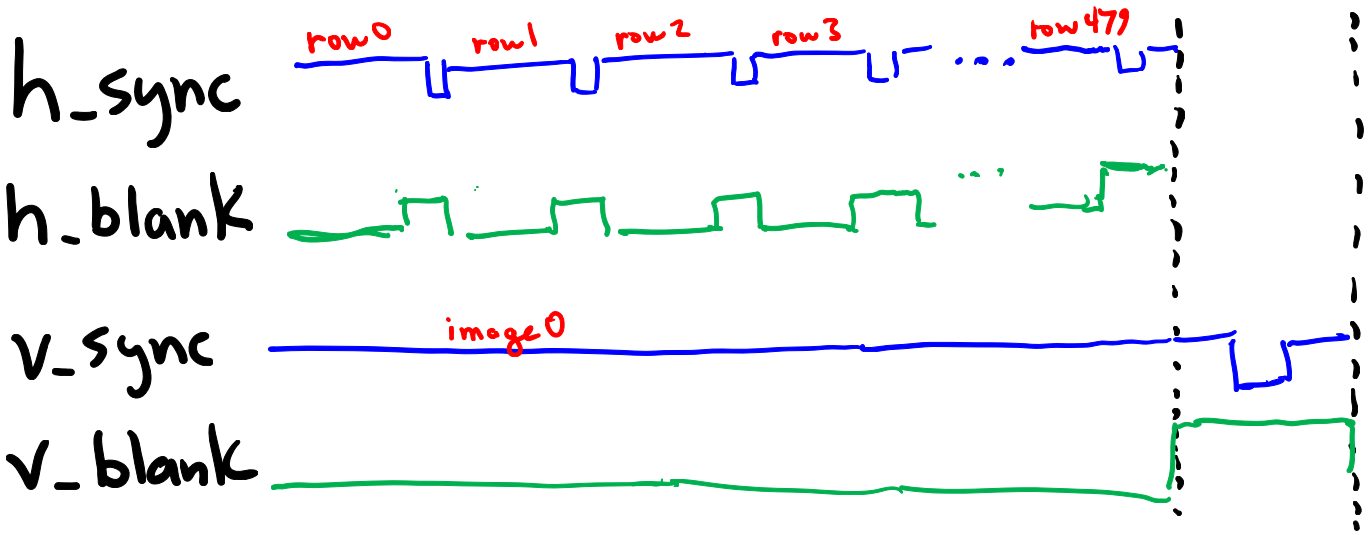


H_sync and V_sync

VGA: 640x480 pixels



H_sync relative to V_sync?



what about columns? pixels?

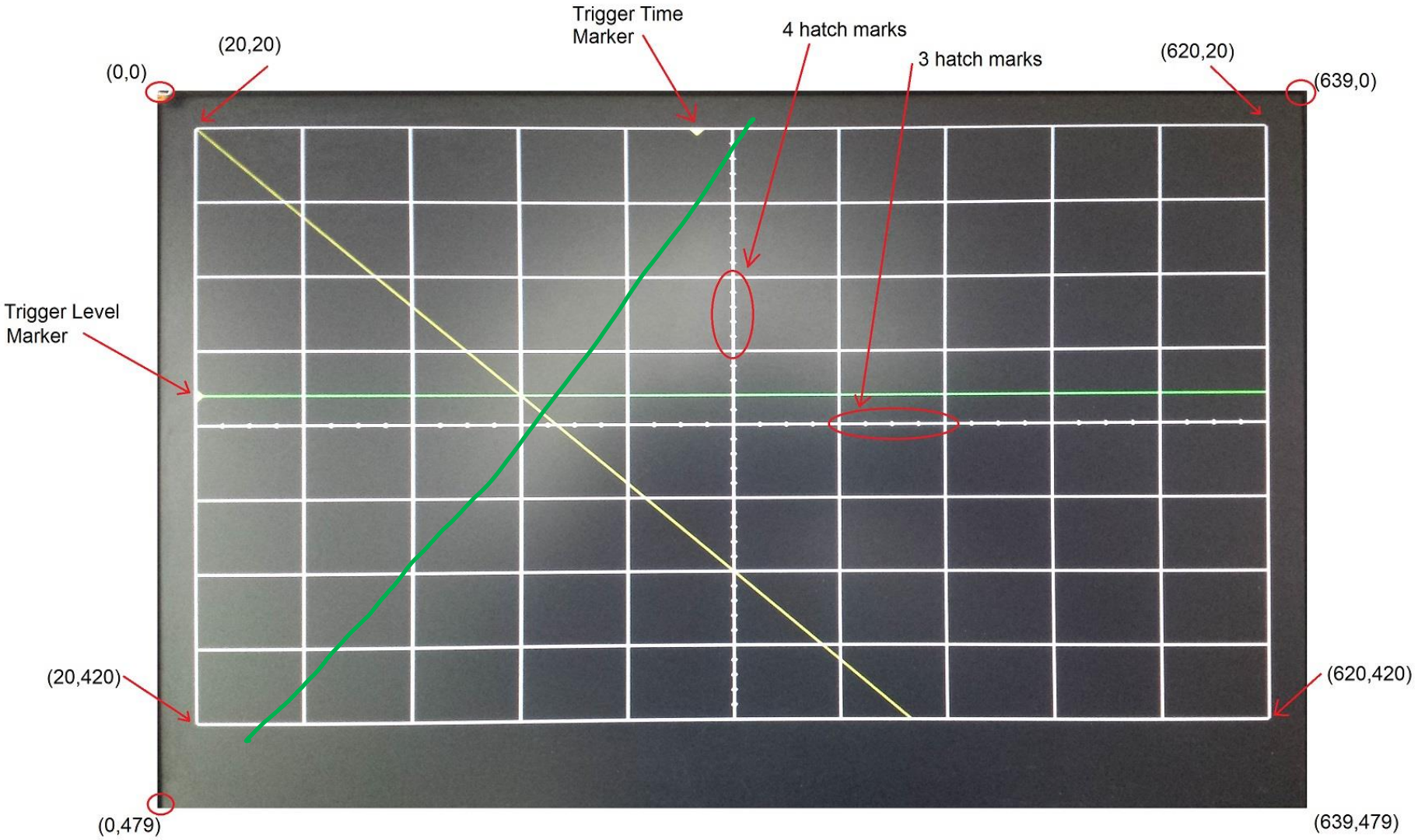
UNITED STATES AIR FORCE ACADEMY

HW 5 – Lab 1 Prelab

1. Draw a detailed diagram of the oscilloscope grid required for Lab1. A detailed diagram must be drawn on green engineering paper (or computer) and include
   - (x,y) corners of the monitor.
   - (x,y) each of the four major corners (already given).
   - y-coordinates for all the major horizontal grid lines.
   - (x,y) coordinates for one set of three horizontal of hatch marks. Indicate with an arrow which set of three.
   - x-coordinates for all the major vertical grid lines.
   - (x,y) coordinates for one set of four vertical of hatch marks. Indicate with an arrow which set of four.

VGA: 640x480 pixels

HW 5 – Lab 1 Prelab



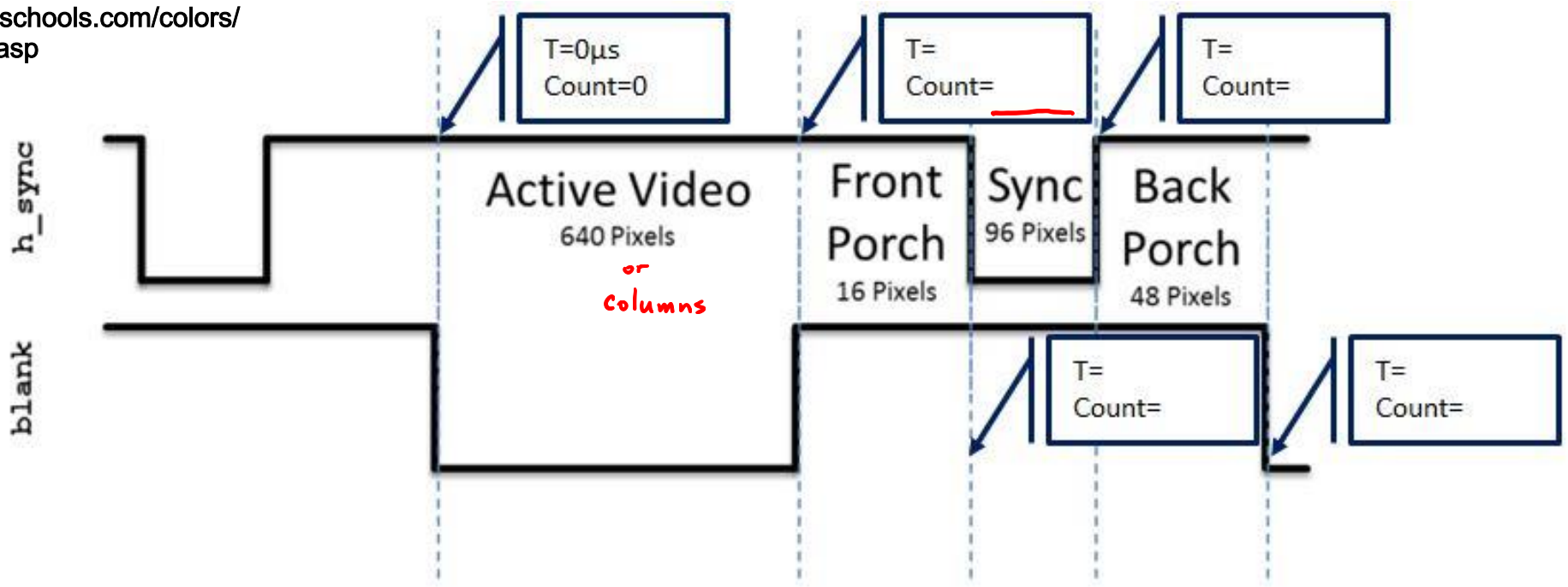
HW 5 – Lab 1 Prelab

T = 1/F = _____

2. Given that the pixel clock is running at 25Mhz, add the durations of the h_synch and v_synch signals show in Lab1. Set time=0 on the blue dashed line on the left side of the region labeled "Active Video".

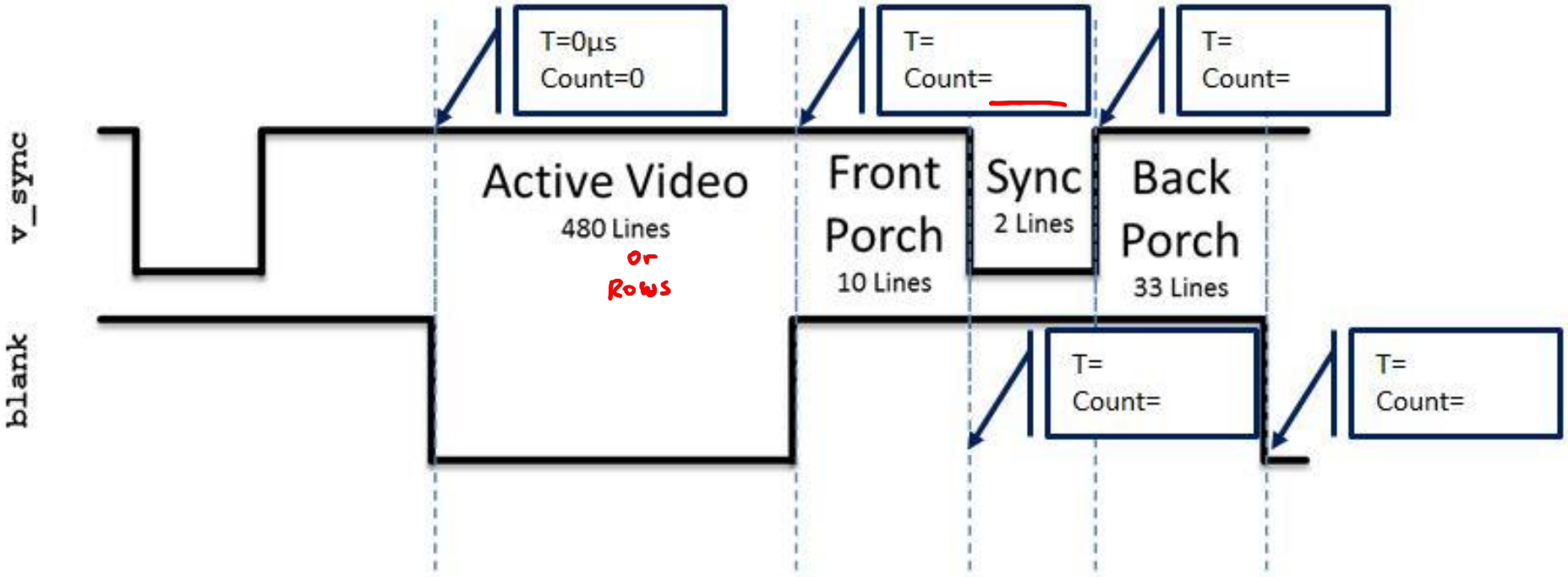
F=

w.w3schools.com/colors/
mes.asp



HW 5 – Lab 1 Prelab

2. Given that the pixel clock is running at 25Mhz, add the durations of the h_synch and v_synch signals show in Lab1. Set time=0 on the blue dashed line on the left side of the region labeled "Active Video".



Lab 1 Intro – VGA Overview

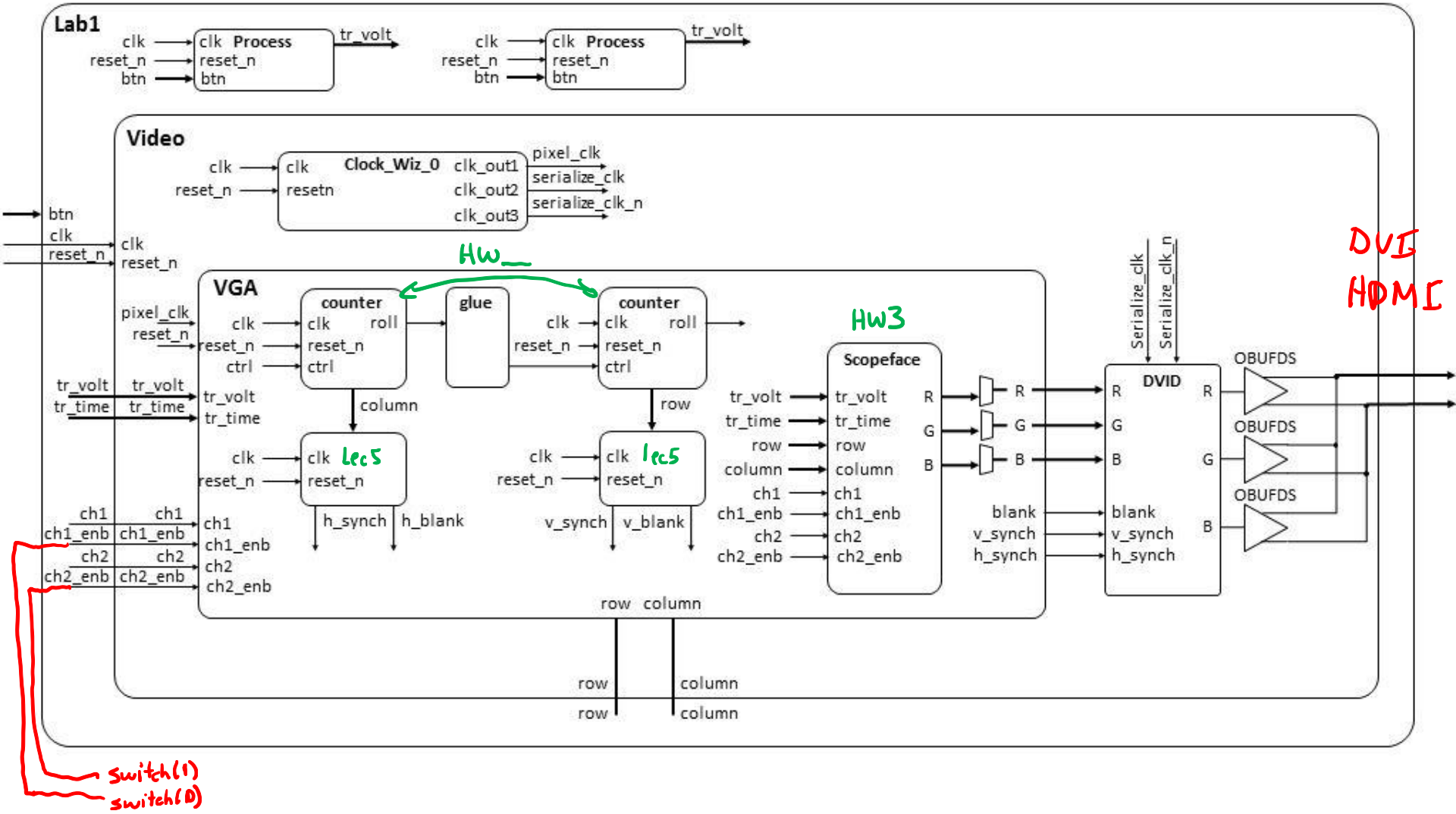
- https://www.w3schools.com/colors/colors_names.asp

RGB

Black	R =	G =	B =
White	R =	G =	B =
Green	R =	G =	B =
Blue	R =	G =	B =
Yellow	R =	G =	B =

Not complete!

Lab 1 Intro – Architecture



Gated logic?

blank <= h_blank ___ v_blank;

VGA Module

```
entity vga is
        Port(   clk: in  STD_LOGIC;
                reset_n : in  STD_LOGIC;
                h_sync : out  STD_LOGIC;
                v_sync : out  STD_LOGIC;
                blank : out  STD_LOGIC;
                r: out STD_LOGIC_VECTOR(7 downto 0);
                g: out STD_LOGIC_VECTOR(7 downto 0);
                b: out STD_LOGIC_VECTOR(7 downto 0);
                trigger_time: in unsigned(9 downto 0);
                trigger_volt: in unsigned (9 downto 0);
                row: out unsigned(9 downto 0);
                column: out unsigned(9 downto 0);
                ch1: in std_logic;
                ch1_enb: in std_logic;
                ch2: in std_logic;
                ch2_enb: in std_logic);
end vga;
```

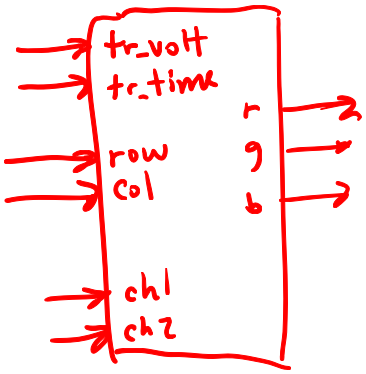
UNITED STATES AIR FORCE ACADEMY

VGA Module

clk	This is the 25Mhz pixel clock generated by the DCM in the video module.
reset_n	This is the same active low reset signal passed into the top level Lab1 module.
tr_volt	This is a 10-bit unsigned value representing the trigger voltage. This value is passed to the scopeFace module so that a yellow arrow (see Trigger Level Marker in the screen show) on the vertical axis.
tr_time	This is a 10-bit unsigned value representing the trigger time. This value is passed to the scopeFace module so that a yellow arrow (see Trigger Time Marker in the screen show) on the horizontal axis.
ch1	This 1-bit signal signals the VGA module to draw the channel 1 signal on the scope for this row, column pixel. When the value is 1, draw a yellow pixel on the display at the current row,colum position. When 0, do not draw a pixel.
ch1_enb	This 1-bit signal enable the ch1 signal to be drawn.
ch2	This 1-bit signal signals the VGA module to draw the channel 2 signal on the scope for this row,column pixel. When the value is 1, draw a green pixel on the display at the current row, column position. When 0, do not draw a pixel.
ch2_enb	This 1-bit signal enable the ch2 signal to be drawn.
R	The 8-bit red intensity for this row,column pixel on the screen.
G	The 8-bit green intensity for this row,column pixel on the screen.
B	The 8-bit blue intensity for this row,column pixel on the screen.
Row	The current row being drawn on the display.
Column	The current row being drawn on the display.
blank	The blank signal for the current row,column position. Its the logical OR of the h_blank and v_blank signals.
h_synch	The h_synch signal for the current row,column position.
v_synch	The v_synch signal for the current row,column position.
Behavior	The VGA component contains a pair of cascaded counters which generate the row and column values of the current pixel being displayed. The row and column values are used to generate the blank, h_synch and v_synch signals according to the Figures above. The scopeFace component (more on this below), takes the row and column values (along with some other information) and generates the R,G,B color of that pixel. The three muxes on the output of the R,G,B output of the scopeFace component output the scopeFace R,G,B values for row,column values within the 640x480 displayable region, or 0's for values outside this region.

scopeFace Module

```
entity scopeFace is
    Port ( row : in  unsigned(9 downto 0);
           column : in  unsigned(9 downto 0);
           trigger_volt: in unsigned (9 downto 0);
           trigger_time: in unsigned (9 downto 0);
           r : out  std_logic_vector(7 downto 0);
           g : out  std_logic_vector(7 downto 0);
           b : out  std_logic_vector(7 downto 0);
           ch1: in std_logic;
           ch1_enb: in std_logic;
           ch2: in std_logic;
           ch2_enb: in std_logic);
end scopeFace;
```



UNITED STATES AIR FORCE ACADEMY

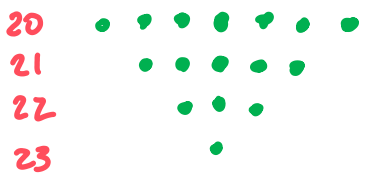
ScopeFace Module

clk	This is the 25Mhz pixel clock generated by the DCM in the video module.
reset_n	This is the same active low reset signal passed into the top level Lab1 module.
tr_volt	This is a 10-bit unsigned value representing the trigger voltage. This value is passed to the scopeFace module so that a yellow arrow (see Trigger Level Marker in the screen show) on the vertical axis.
tr_time	This is a 10-bit unsigned value representing the trigger time. This value is passed to the scopeFace module so that a yellow arrow (see Trigger Time Marker in the screen show) on the horizontal axis.
ch1	This 1-bit signal signals the VGA module to draw the channel 1 signal on the scope for this row, column pixel. When the value is 1, draw a yellow pixel on the display at the current row,column position. When 0, do not draw a pixel.
ch1_enb	This 1-bit signal enable the ch1 signal to be drawn.
ch2	This 1-bit signal signals the VGA module to draw the channel 2 signal on the scope for this row,column pixel. When the value is 1, draw a green pixel on the display at the current row, column position. When 0, do not draw a pixel.
ch2_enb	This 1-bit signal enable the ch2 signal to be drawn.
R	The 8-bit red intensity for this row,column pixel on the screen.
G	The 8-bit green intensity for this row,column pixel on the screen.
B	The 8-bit blue intensity for this row,column pixel on the screen.
Row	The current row being drawn on the display.
Column	The current row being drawn on the display.
Behavior	The scopeFace component takes in the current row,column coordinates of the display and generates the R,G,B values at that screen coordinate. For example, if row,column = 20,20 then the R,G,B output should be 0xFF,0xFF,0xFF (white) because the upper left corner of the O'scope grid display is white. Note, you can get the RGB values for common colors at this web site.

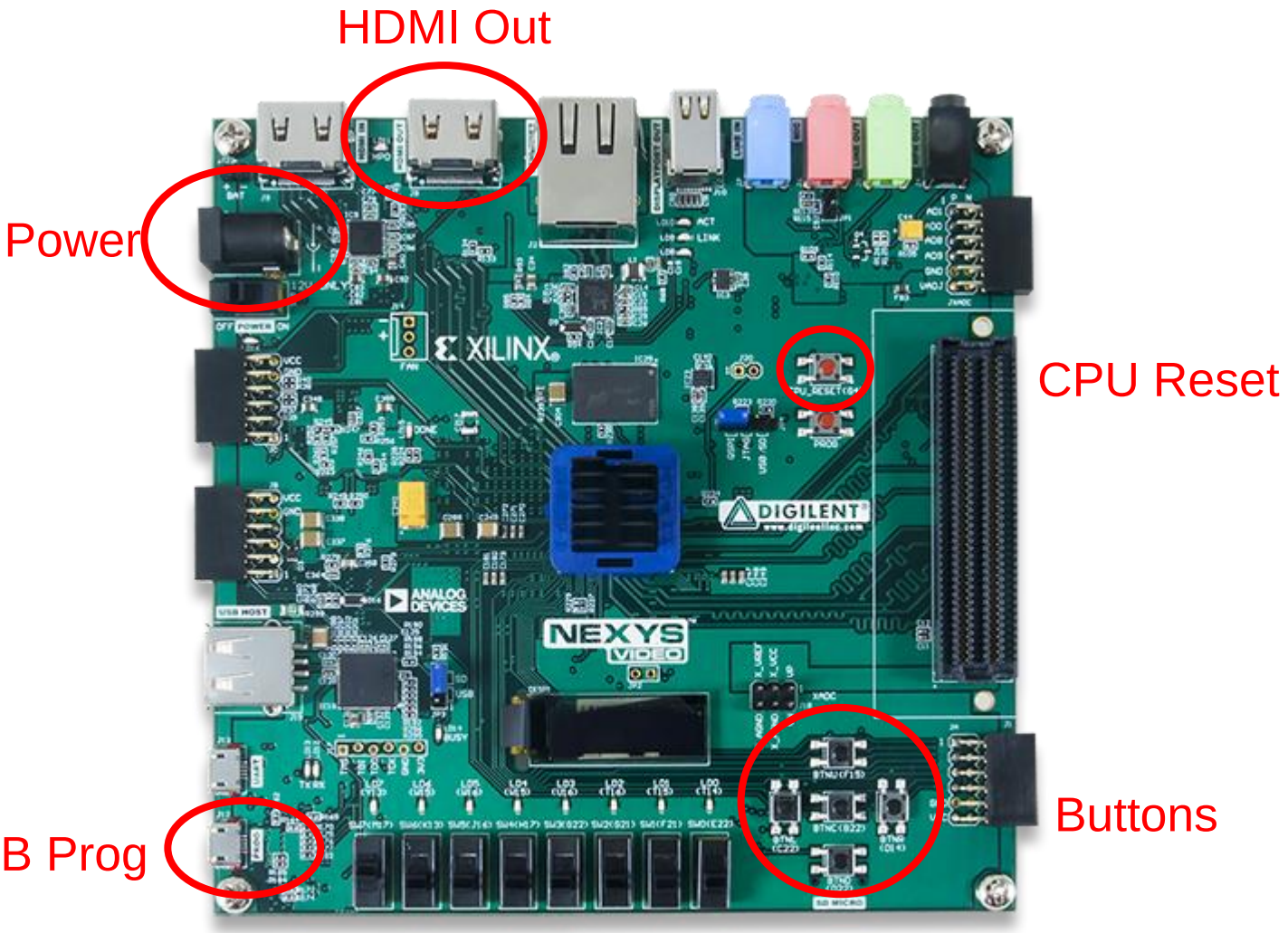
Scopeface: just CSA code

- how to make an X
  - ch1 <= '1' when _____
  - ch2 <= '1' when _____
- gridH <= '1' when row = __ or row = __ or …
- gridV <= '1' when col = __ or col = ___ or …
- white <= '1' when gridH = '1' or gridV = '1' or …
- r <= "11111111" when white = '1' or ch1 = '1' or… and ch1_enable = '1'
- g <= "11111111" when white = '1' or ch1 = '1' or ch2 = '1' or … and ch2_enable = '1' ?
- b <= "11111111" when white = '1' or …

Triangle <= '1' when _____



Lab 1 Connections



UNITED STATES AIR FORCE ACADEMY

Setup code for lab1

- Create a lab 1 project in Vivado
- Copy code from
  - https://georgeyork.github.io/ECE383_web/lab/lab1/lab1.html (middle of the webpage) or Teams → Lab1_cadet.zip
  - Add files to the project
- Save to bitbucket repo

UNITED STATES AIR FORCE ACADEMY

Digital Clocking Wizard

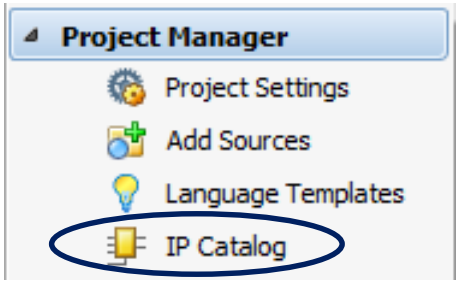
How to convert a 100 MHz clock
to a 25 MHz clock and a 125 MHz clock?

- **Xilinx Clocking Wizard Page:**
  - https://www.xilinx.com/products/intellectual-property/clocking_wizard.html
- **Clocking Wizard v5.3 - LogiCORE IP Product Guide:**
  - https://www.xilinx.com/support/documentation/ip_documentation/clk_wiz/v5_3/pg065-clk-wiz.pdf

UNITED STATES AIR FORCE ACADEMY

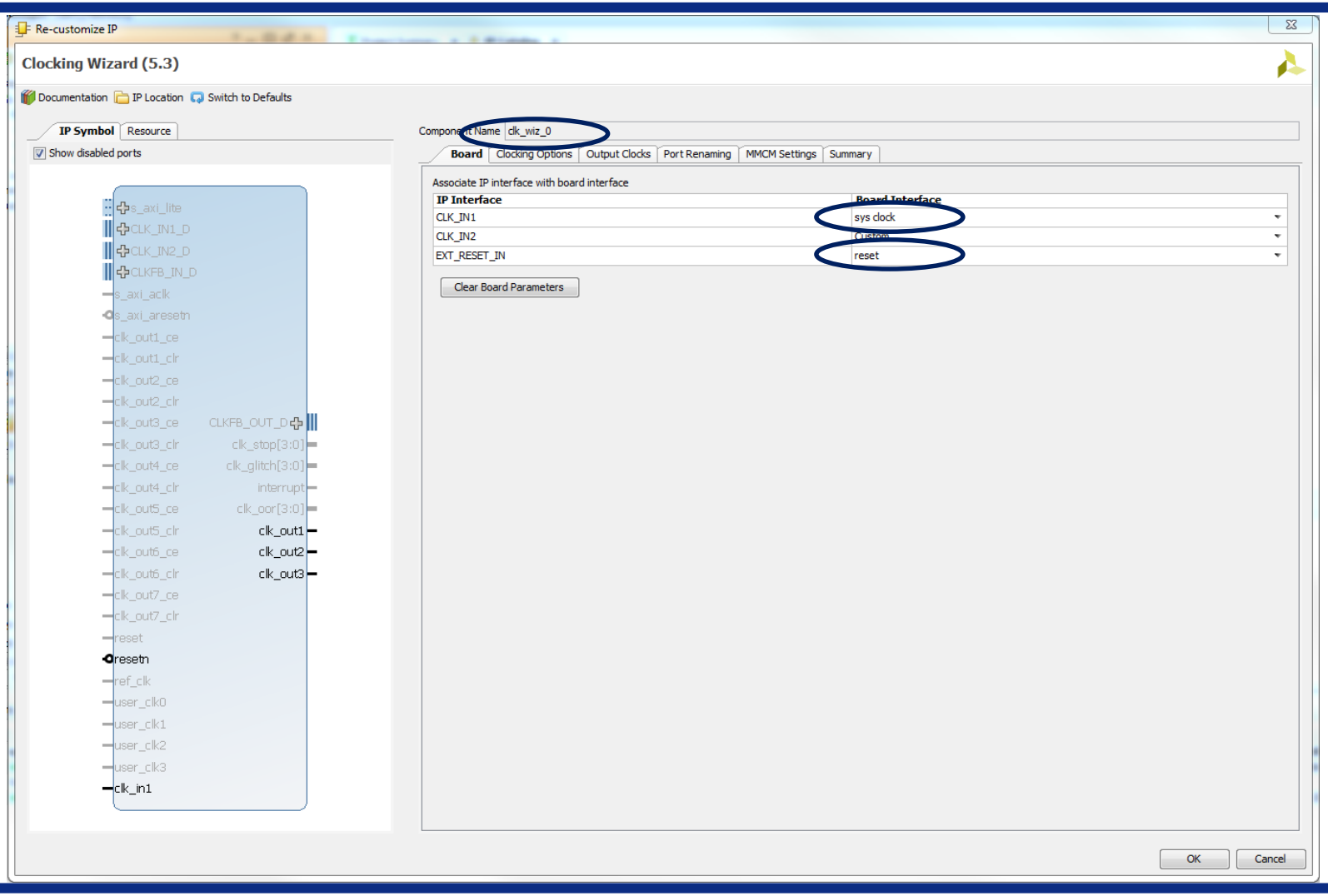
Digital Clocking Wizard

- Click on IP Catalog
- Search for Clocking Wizard IP



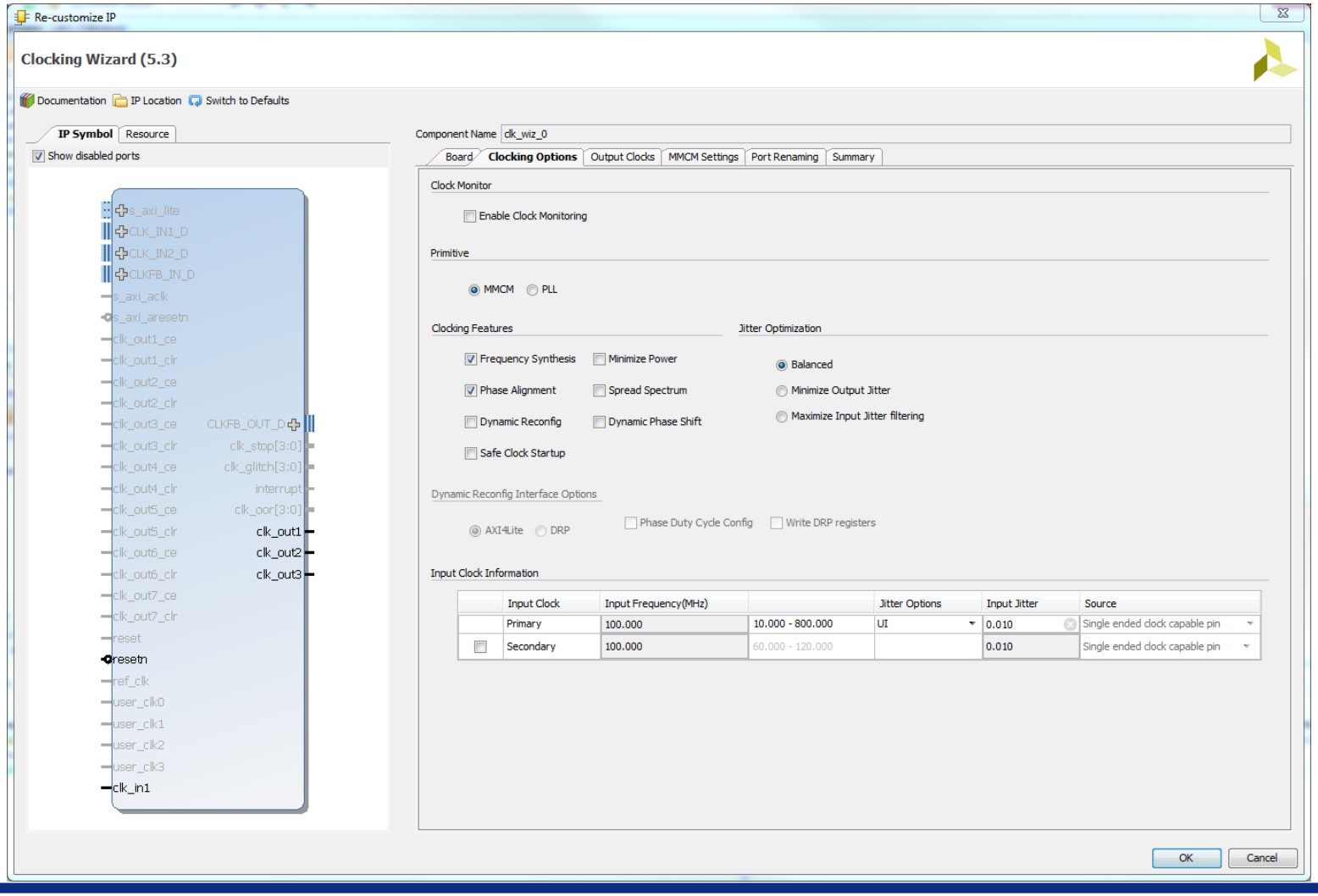
UNITED STATES AIR FORCE ACADEMY

Digital Clocking Wizard



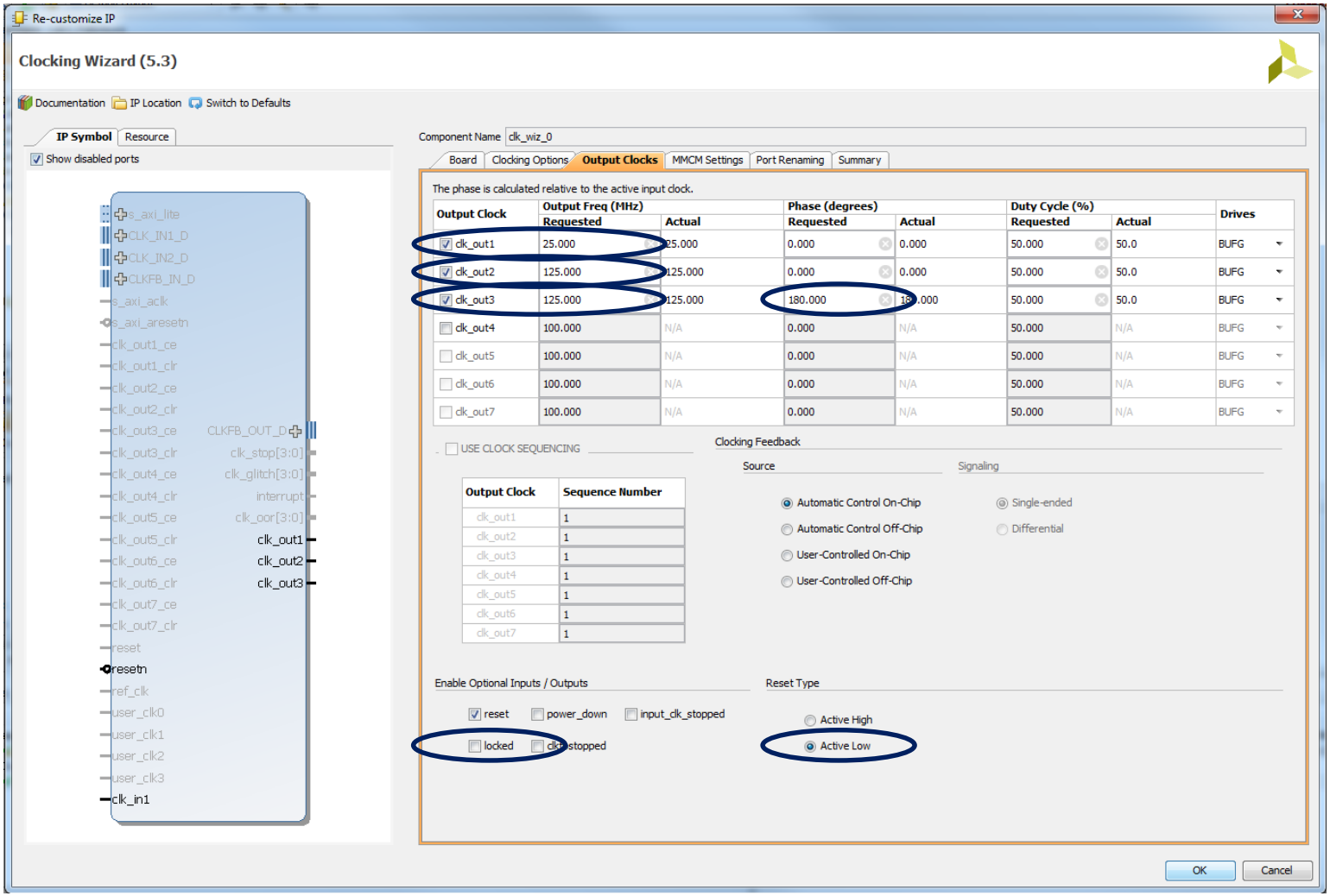
UNITED STATES AIR FORCE ACADEMY

Digital Clocking Wizard



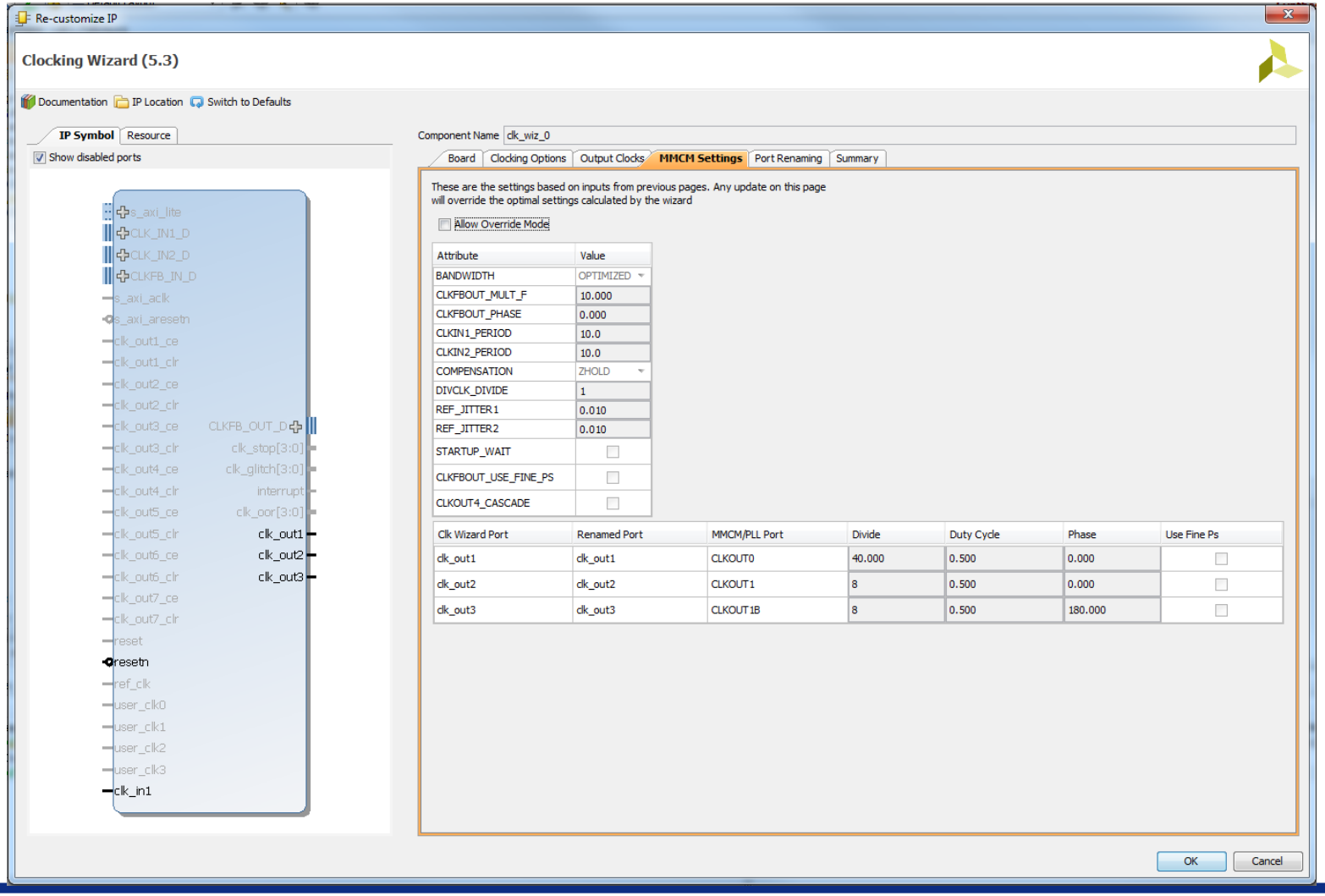
UNITED STATES AIR FORCE ACADEMY

Digital Clocking Wizard



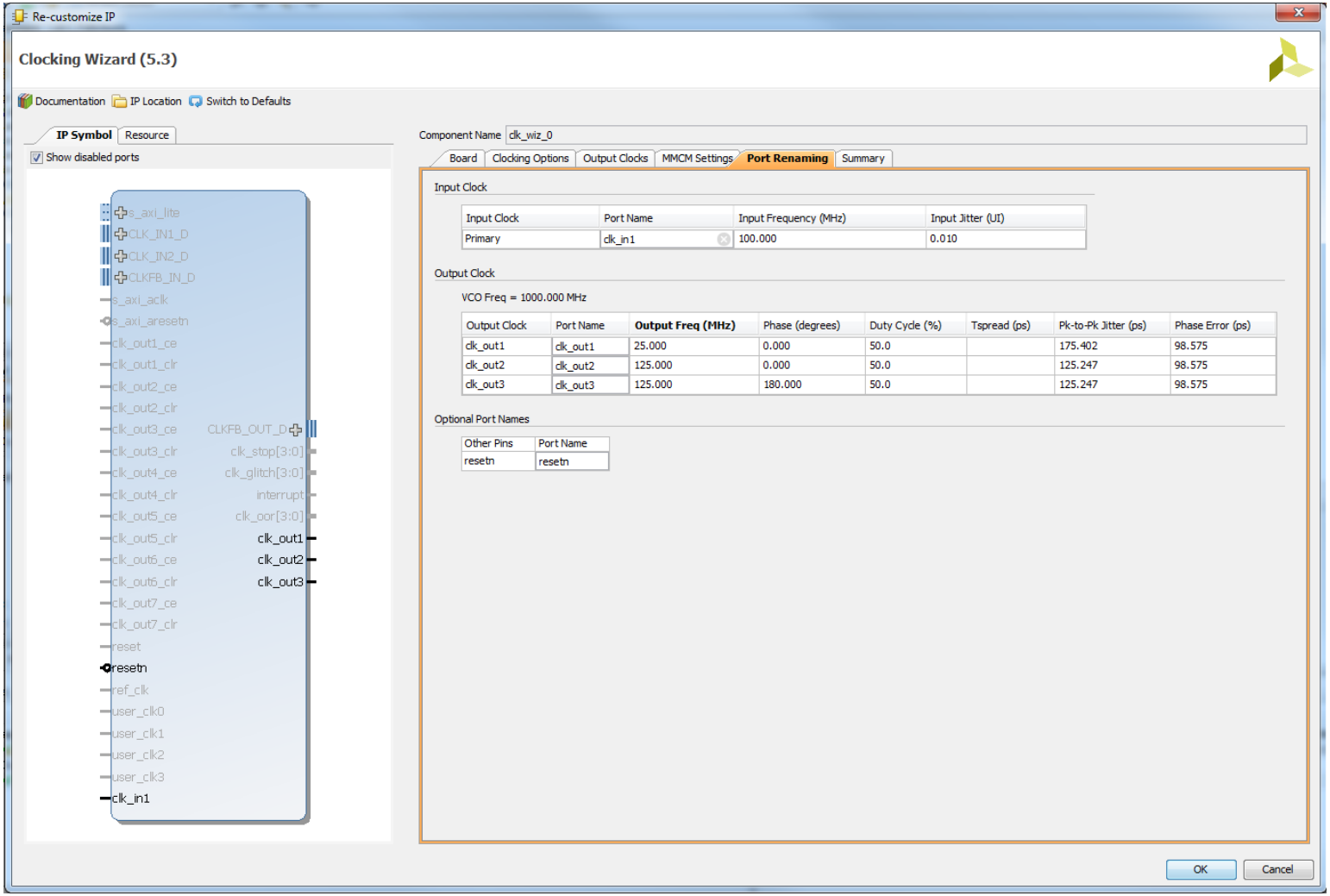
UNITED STATES AIR FORCE ACADEMY

Digital Clocking Wizard



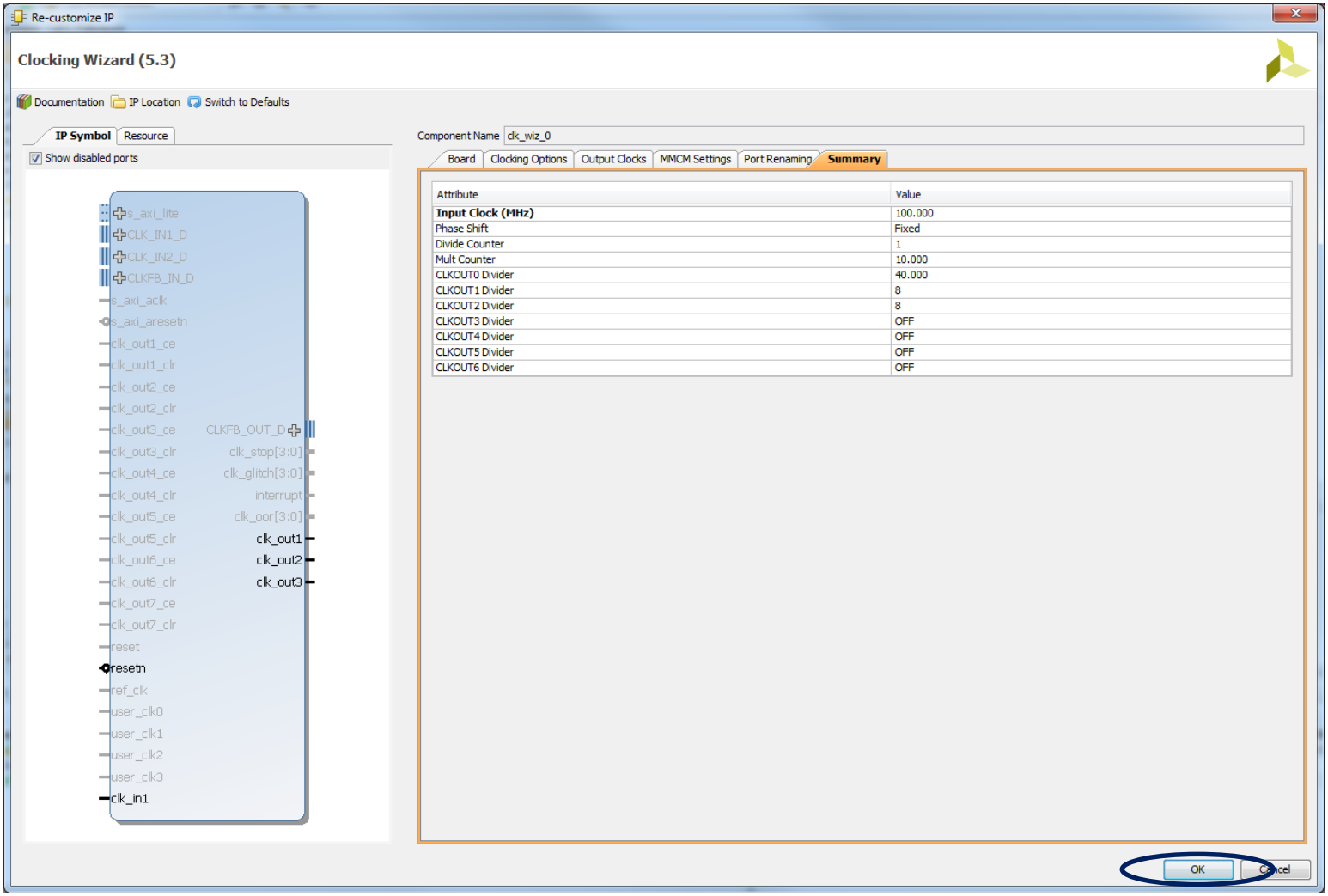
UNITED STATES AIR FORCE ACADEMY

Digital Clocking Wizard



UNITED STATES AIR FORCE ACADEMY

Digital Clocking Wizard



UNITED STATES AIR FORCE ACADEMY

Digital Clocking Wizard

- **Verify Component Declaration in video.vhd**

```
------------------------------------------------------------------------
-- Clock Wizard Component Instantiation Using Xilinx Vivado
------------------------------------------------------------------------
component clk_wiz_0 is
Port (
    clk_in1 : in STD_LOGIC;
    clk_out1 : out STD_LOGIC;
    clk_out2 : out STD_LOGIC;
    clk_out3 : out STD_LOGIC;
    resetn : in STD_LOGIC);
 end component;
```

- **Verify Component Instantiation in video.vhd**

UNITED STATES AIR FORCE ACADEMY

Digital Clocking Wizard

- **Verify Component Instantiation in video.vhd**

```
------------------------------------------------------------------
-- Digital Clocking Wizard using Xilinx Vivado creates 25Mhz pixel clock and
-- 125MHz HDMI serial output clocks from 100MHz system clock. The Digital
-- Clocking Wizard is in the Vivado IP Catalog.
------------------------------------------------------------------
mmcm_adv_inst_display_clocks: clk_wiz_0
   Port Map (
      clk_in1 => clk,
      clk_out1 => pixel_clk, -- 25Mhz pixel clock
      clk_out2 => serialize_clk, -- 125Mhz HDMI serial output clock
      clk_out3 => serialize_clk_n, -- 125Mhz HDMI serial output clock 180
                                      degrees out of phase
      resetn => reset_n);  -- active low reset for Nexys Video
```

UNITED STATES AIR FORCE ACADEMY

Lesson Outline

1. **Comparator Construction**
2. **Gated and Non-Gated Circuit**
3. **Lab 1 Intro**

See Gradescope GC1

-- State what was achieved?

achieved → details

partially achieved → details

not achieved