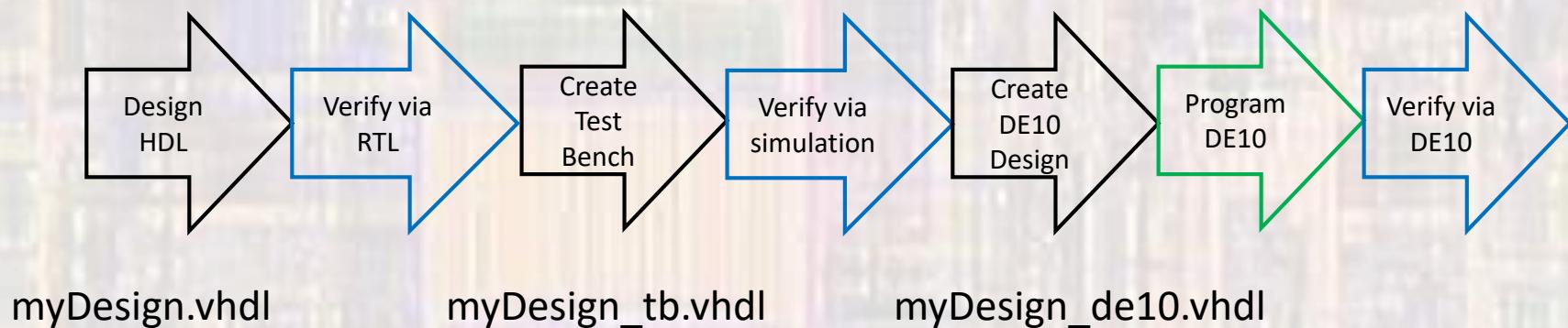


# HDL Project Example

Common - last updated 2/1/19

# HDL Project Example

- Project Flow

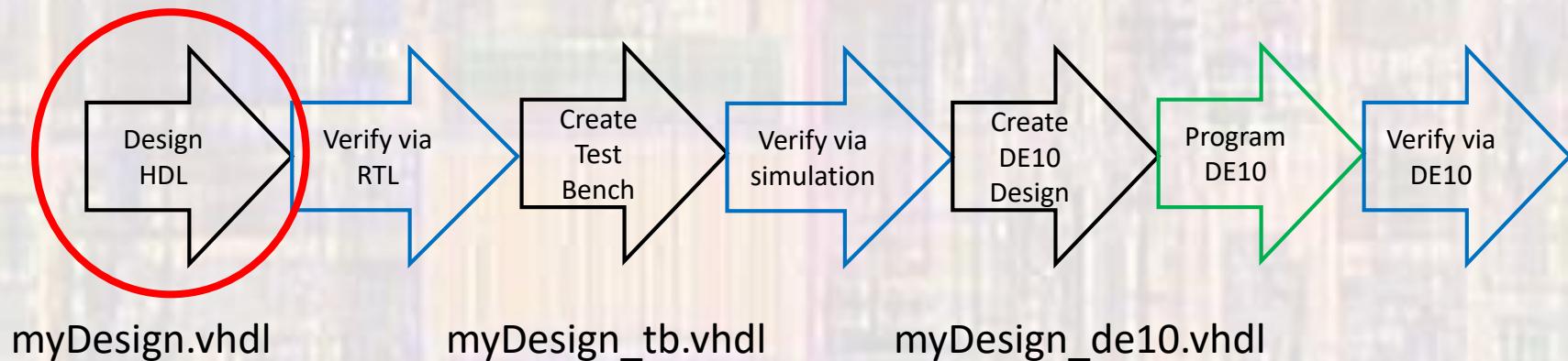


# HDL Project Example

- Create a new Project
  - Select **File -> New Project Wizard**
    - ...
    - See project setup slides if necessary

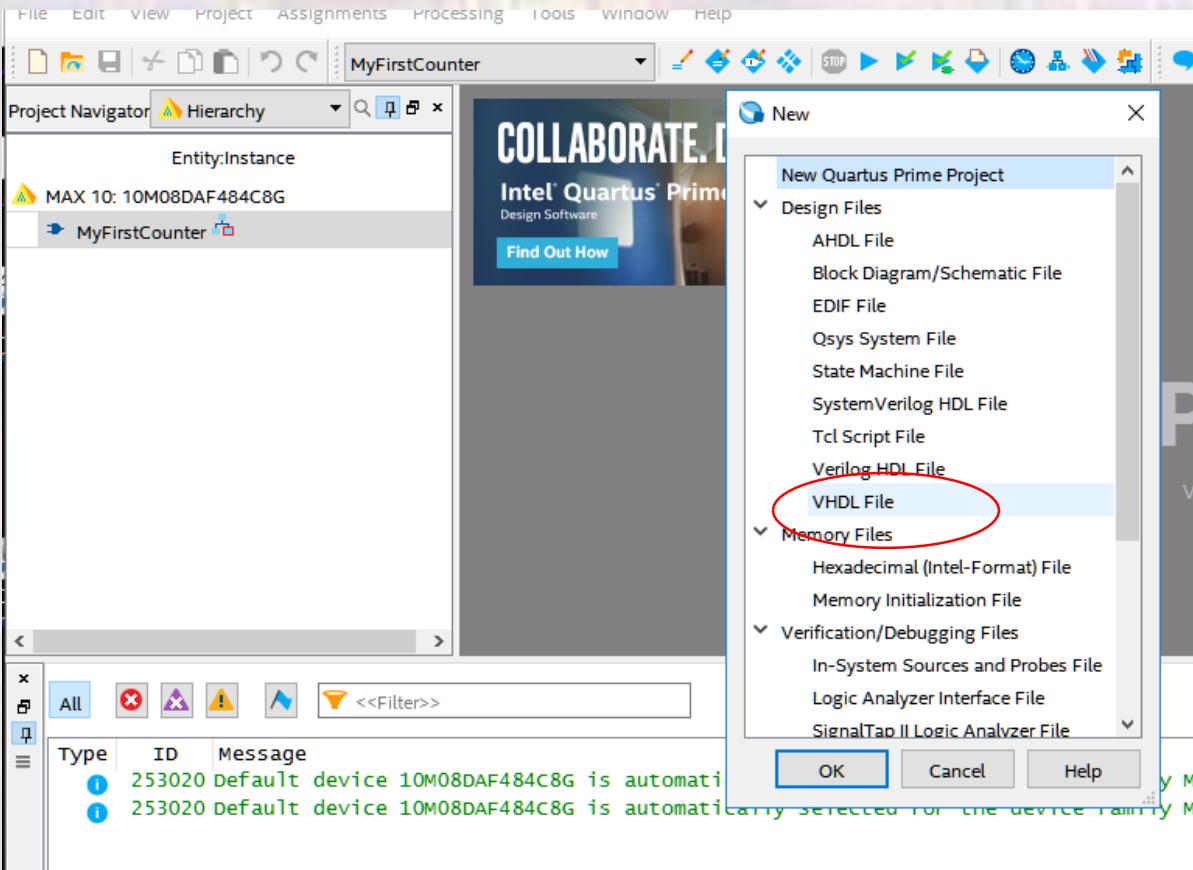
# HDL Project Example

- Project Flow



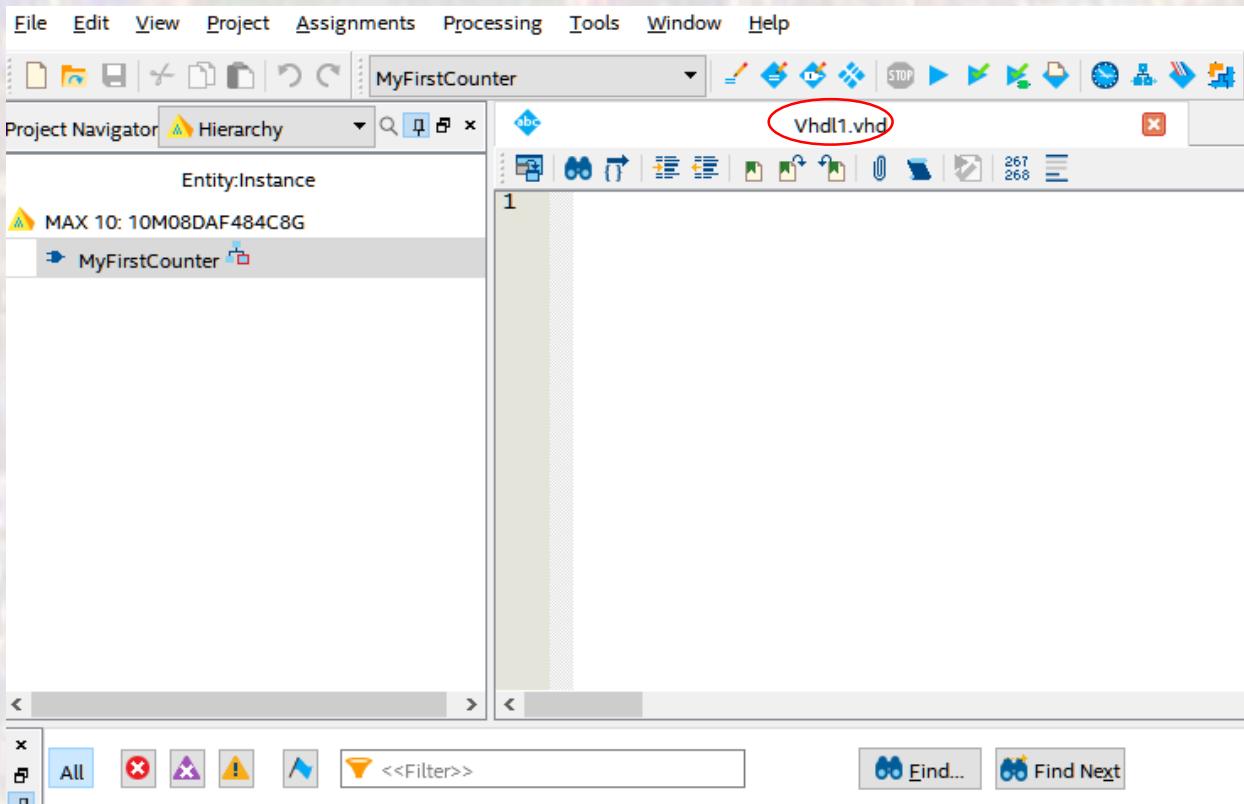
# HDL Project Example

- Create a new HDL file
  - Select **File -> New -> VHDL File**



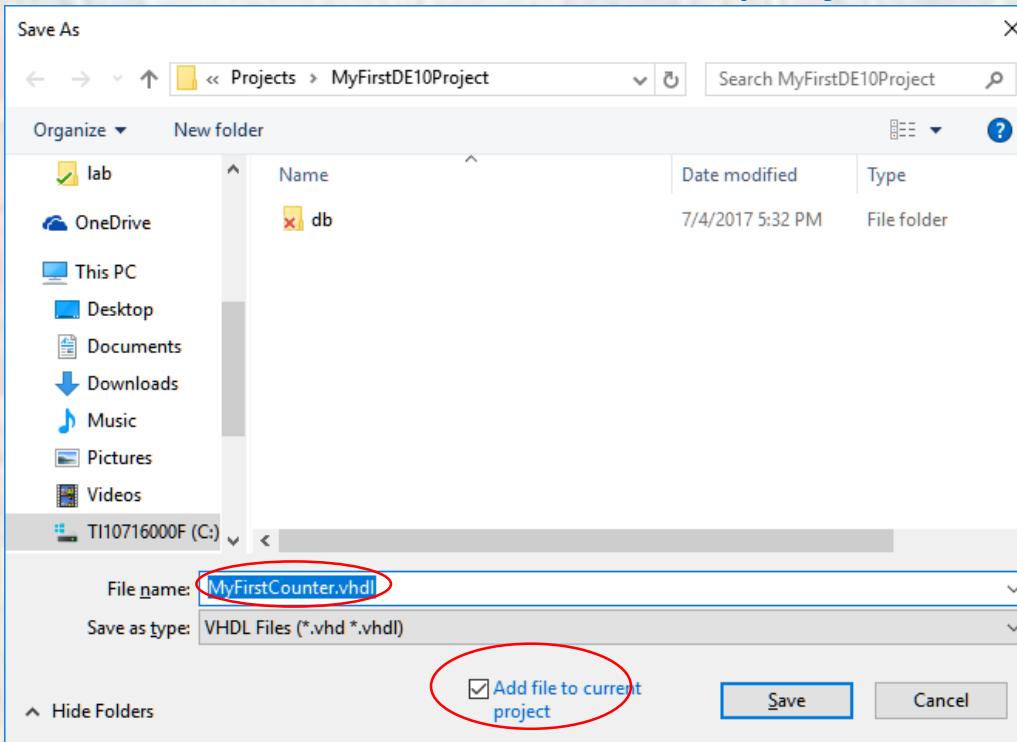
# HDL Project Example

- Create a new HDL file
  - Note the default file name
  - We want this to be our top level entity so must change the name



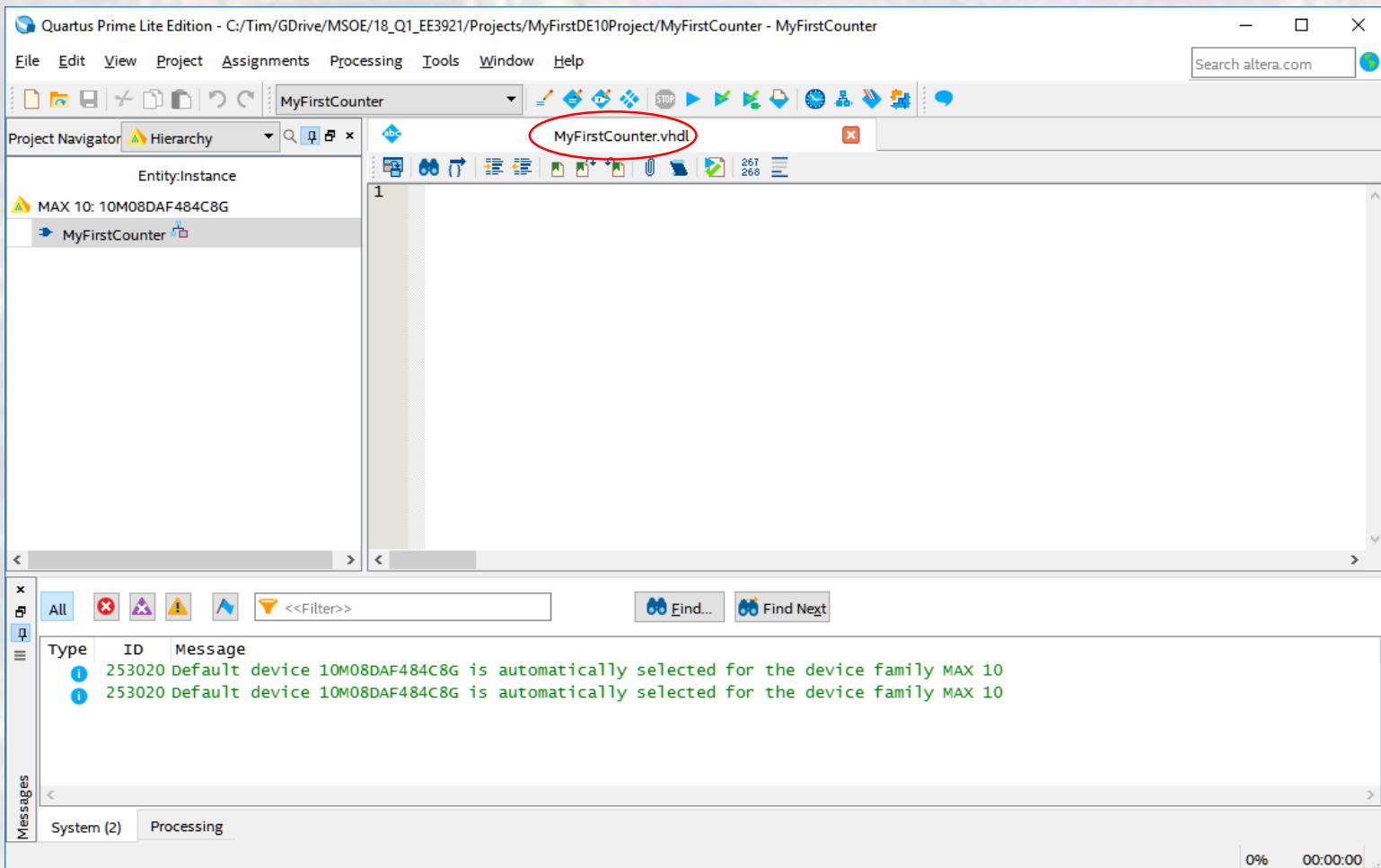
# HDL Project Example

- Create a new HDL file
  - Select **File -> Save As**
  - Set the file name to the our desired value
  - Change the file extension to vhdl (for VHDL files)
  - Make sure “**Add file to current project**” is checked



# HDL Project Example

- Create a new HDL file
  - Note the name change



# HDL Project Example

- Create a new HDL file
  - Enter your VHDL code

The screenshot shows the Quartus Prime Lite Edition software interface. The main window title is "Quartus Prime Lite Edition - C:/Tim/GDrive/MSOE/18\_Q3\_CE1911/Projects/Lab\_Projects/MyFirstDE10Project/MyFirstCounter - MyFirstCounter". The menu bar includes File, Edit, View, Project, Assignments, Processing, Tools, Window, and Help. A search bar at the top right says "Search altera.com". The interface has several panels: "Project Navigator" on the left showing files like "MyFirstCounter\_de10.vhd", "clk10Hz.vhd", "MyFirstCounter.vhd", and "MyFirstCounter\_tb.vhd"; a central "Code Editor" pane displaying VHDL code for a behavioral architecture; a "Tasks" panel showing compilation status; and a "Library" panel on the right listing various IP cores and libraries. The code in the editor is:

```
38  architecture behavioral of myfirstcounter is
39  --
40  -- Internal signals
41  --
42  signal cnt_sig:  unsigned(7 downto 0);
43  begin
44  --
45  -- update section
46  --
47  process(clk)
48  begin
49  --
50  if (rstb ='0') then
51  cnt_sig <= (others => '0');-- reset
52  elsif (rising_edge(clk)) then
53  if (dir = '0') then
54  cnt_sig <= (cnt_sig + 1); -- count up
55  else
56  cnt_sig <= (cnt_sig - 1); -- count down
57  end if;
58  end process;
59  --
60  --
61  --
62  --
```

# HDL Project Example

- Create a new HDL file
  - Enter your VHDL code

```
--Project Header---
-- MyFirstCounter
-- Created: 2/23/18
-- By: johnsontimoj
-- For: CE1911
--

---Project Overview---
-- This project creates a binary up/down
-- counter
--

--- Project Details ---
-- Inputs: rstb, clock, dir
-- outputs: cnt
-- dir determines up(0) or down(1)
--

-- Library inclusion
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity MyFirstCounter is
    port(
        i_rstb:      in  std_logic;
        i_clk:       in  std_logic;
        i_dir:       in  std_logic;
        o_cnt:       out std_logic_vector(7 downto 0)
    );
end entity;
```

```
architecture behavioral of myfirstcounter is
    -- Internal Signals
    signal cnt_sig:    unsigned(7 downto 0);

begin
    -- Update Section
    process(i_clk, i_rstb)
    begin
        if (i_rstb = '0') then
            cnt_sig <= (others => '0'); -- reset
        elsif (rising_edge(i_clk)) then
            if (i_dir = '0') then
                cnt_sig <= (cnt_sig + 1); -- count up
            else
                cnt_sig <= (cnt_sig - 1); -- count down
            end if;
        end if;
    end process;

    -- Output Section
    --
    o_cnt <= std_logic_vector(cnt_sig);
end architecture;
```

# Combinational Project Example

- Create VHDL file
  - Create a component template for your design (DUT)
    - Select [File → Create/Update → Create VHDL Component Declaration Files from Current File](#)

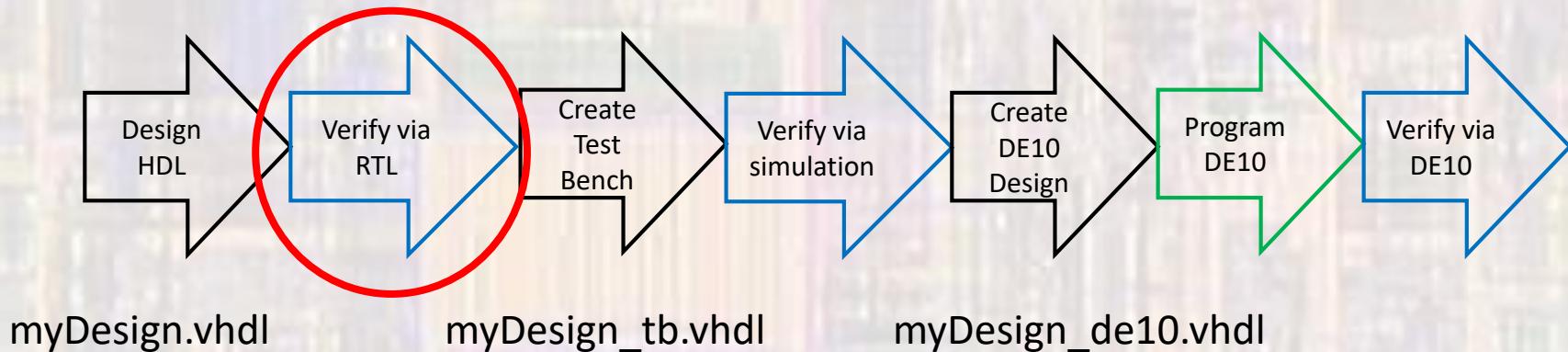
Copyright (C) 2017 Intel Corporation. All rights reserved.  
-- Your use of Intel Corporation's design tools, logic functions  
-- and other software and tools, and its AMPP partner logic  
-- functions, and any output files from any of the foregoing  
-- (including device programming or simulation files), and any  
-- associated documentation or information are expressly subject  
-- to the terms and conditions of the Intel Program License  
-- Subscription Agreement, the Intel Quartus Prime License Agreement,  
-- the Intel FPGA IP License Agreement, or other applicable license  
-- agreement, including, without limitation, that your use is for  
-- the sole purpose of programming logic devices manufactured by  
-- Intel and sold by Intel or its authorized distributors. Please  
-- refer to the applicable agreement for further details.

-- Generated by Quartus Prime Version 17.1 (Build Build 590 10/25/2017)  
-- Created on Fri Feb 23 09:52:29 2018

```
COMPONENT MyFirstCounter
PORT
(
    i_rstb      : IN STD_LOGIC;
    i_clk       : IN STD_LOGIC;
    i_dir       : IN STD_LOGIC;
    o_cnt       : OUT STD_LOGIC_VECTOR(7 DOWNTO
0)
);
END COMPONENT;
```

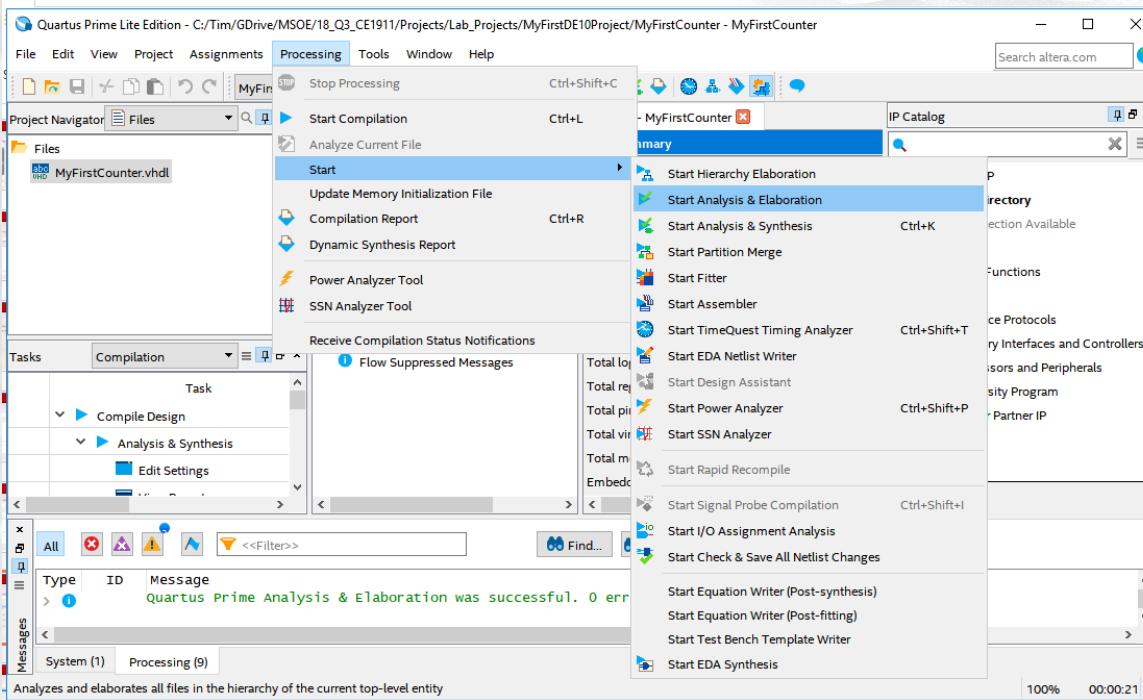
# HDL Project Example

- Project Flow



# HDL Project Example

- Create VHDL file
  - Verify your code is synthesizable
    - Set the VHDL file as the Top-Level Entity
      - In project Navigator – Files, Right click the VHDL file and select: **Set as Top-Level Entity**
    - Select **Processing -> Start -> Start Analysis & Elaboration**



# HDL Project Example

- Create VHDL file
  - Check all Warnings and Errors

The screenshot shows the Quartus Prime software interface. At the top, the menu bar includes File, Edit, View, Project, Assignments, Processing, Tools, Window, and Help. A search bar for 'altera.com' is also present. The main window displays a 'Project Navigator' on the left with a 'Hierarchy' view showing an Entity:Instance named 'MyFirstCounter' under a MAX 10: 10M08DAF484C8G device. The central area shows a 'Flow Summary' table with various parameters and their values. A red circle highlights the 'Flow Status' entry, which is 'Successful - Tue Jul 04 18:00:29 2017'. A blue box labeled 'Successful' is overlaid on the top right of the flow summary table. Below the table, a toolbar has several icons, with the 'All' icon circled in red. The bottom section shows a table of messages with columns for Type, ID, and Message. A yellow callout box points to the message table with the text: 'you will get a large number of warnings if you read in the qsf file – 1 for every unused pin'. On the far left, a blue box contains the text 'Highlight errors and warning'.

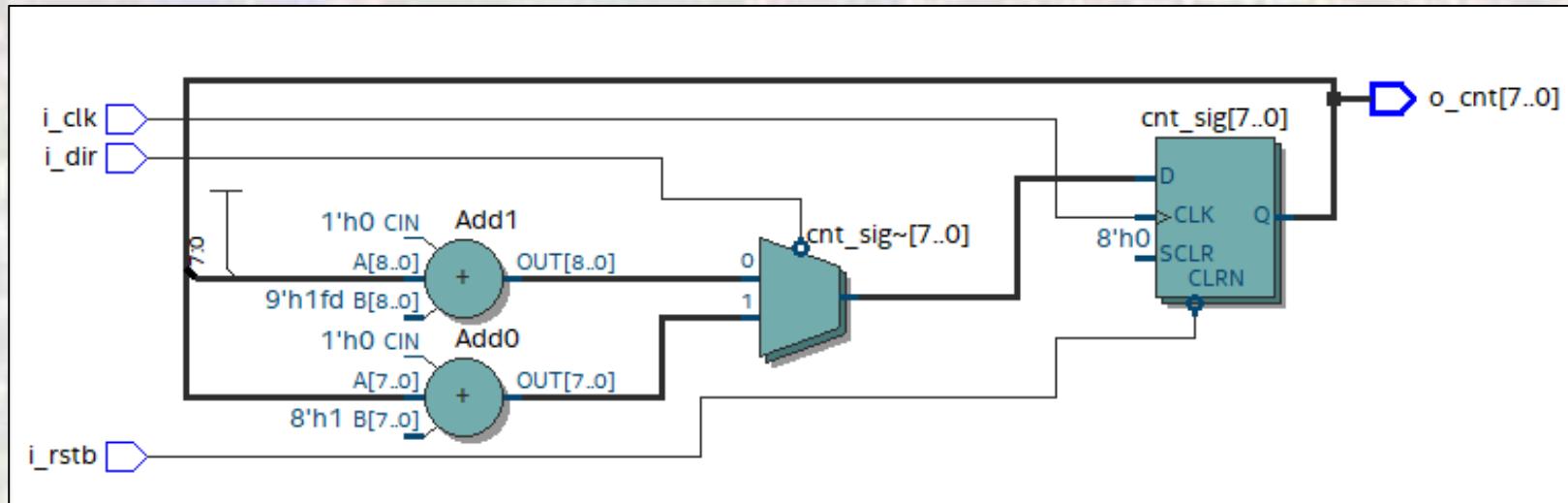
Type	ID	Message
!	18236	Number of processors has not been specified which may cause overloading on shared machines. Set the global assignment N
!	20030	Parallel compilation is enabled and will use 2 of the 2 processors detected
!	12021	Found 2 design units, including 1 entities, in source file myfirstcounter.vhd
!	12127	Elaborating entity "MyFirstCounter" for the top level hierarchy
		Quartus Prime Analysis & Elaboration was successful. 0 errors, 1 warning

you will get a large number of warnings if you read in the qsf file – 1 for every unused pin

Highlight errors and warning

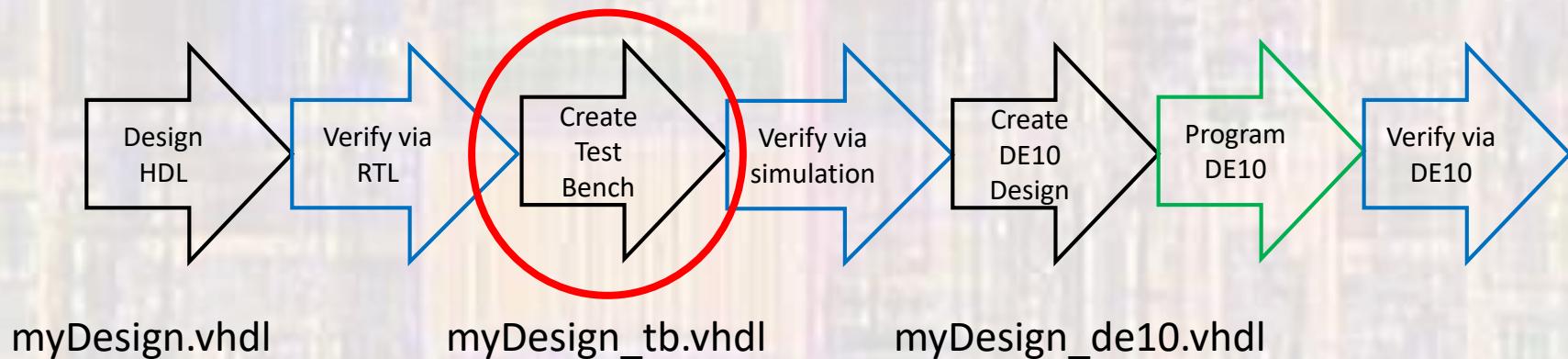
# HDL Project Example

- Create VHDL file
  - **ALWAYS** check your RTL to make sure it makes sense
  - Select **Tools** → **Netlist Viewers** → **RTL Viewer**



# HDL Project Example

- Project Flow



# Combinational Project Example

- Project Verification
  - Create a Test Bench
  - Open a new vhdl design file with the name MyFirstCounter\_tb
  - Create a test bench entity and instantiate your design at the device under test (DUT)
  - Create test inputs using the non-synthesizable `wait` instructions

# HDL Project Example

- Project Verification
  - Create a Test Bench

```
-- MyFirstCounter_tb.vhd1
-- created: 2/23/18
-- by: johnsontimoj
-- rev: 0
--
-- testbench for 8 bit counter
-- of MyFirstCounter.vhd1
-- brute force implementation
--

library ieee;
use ieee.std_logic_1164.all;

entity MyFirstCounter_tb is
    -- no entry - testbench
end entity;

architecture testbench of MyFirstCounter_tb is
    signal CLK: std_logic;
    signal RSTB: std_logic;
    signal DIR: std_logic;
    signal CNT: std_logic_vector(7 downto 0);
    constant PER: time := 20 ns;

    --
    -- Component prototype
    --
COMPONENT MyFirstCounter
    PORT
    (
        i_rstb : IN STD_LOGIC;
        i_clk  : IN STD_LOGIC;
        i_dir  : IN STD_LOGIC;
        o_cnt  : OUT STD_LOGIC_VECTOR(7 DOWNTO 0)
    );
END COMPONENT;
```

```
begin
    -----
    -- Device under test (DUT)
    -----
    DUT: MyFirstCounter
        port map(
            i_rstb  => RSTB,
            i_clk   => CLK,
            i_dir   => DIR,
            o_cnt   => CNT
        );

    -----
    -- Brute force test process
    -----
    -- Clock process
    clock: process    -- note - no sensitivity list allowed
        begin
            CLK <= '0';
            wait for PER/2;
            infinite: loop
                CLK <= not CLK; wait for PER/2;
            end loop;
        end process;

    -- Run process
    brute: process    -- note - no sensitivity list allowed
        begin
            --
            -- Initialize inputs
            RSTB <= '0';
            DIR <= '0';

            --
            -- reset
            RSTB <= '0'; wait for 2*PER;
            RSTB <= '1'; wait for 2*PER;

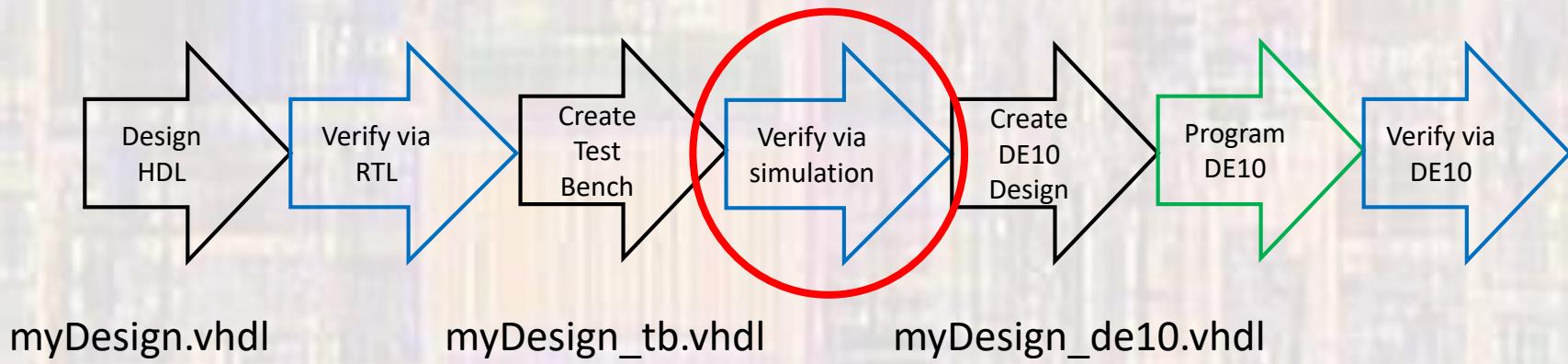
            --
            -- verify count up and roll over
            DIR <= '0'; wait for 258*PER;
            -- verify count down and roll over
            DIR <= '1'; wait for 262*PER;
        end process brute;
    end architecture;
```

# HDL Project Example

- Project Verification
  - Elaborate the design
    - With the original vhdl design set as the top level entity (**not the xxxx\_tb.vhdl design**)
    - Select **Processing → Start → Start Analysis and Elaboration**
  - This causes Quartus to check the Test Bench code along with the original vhdl design

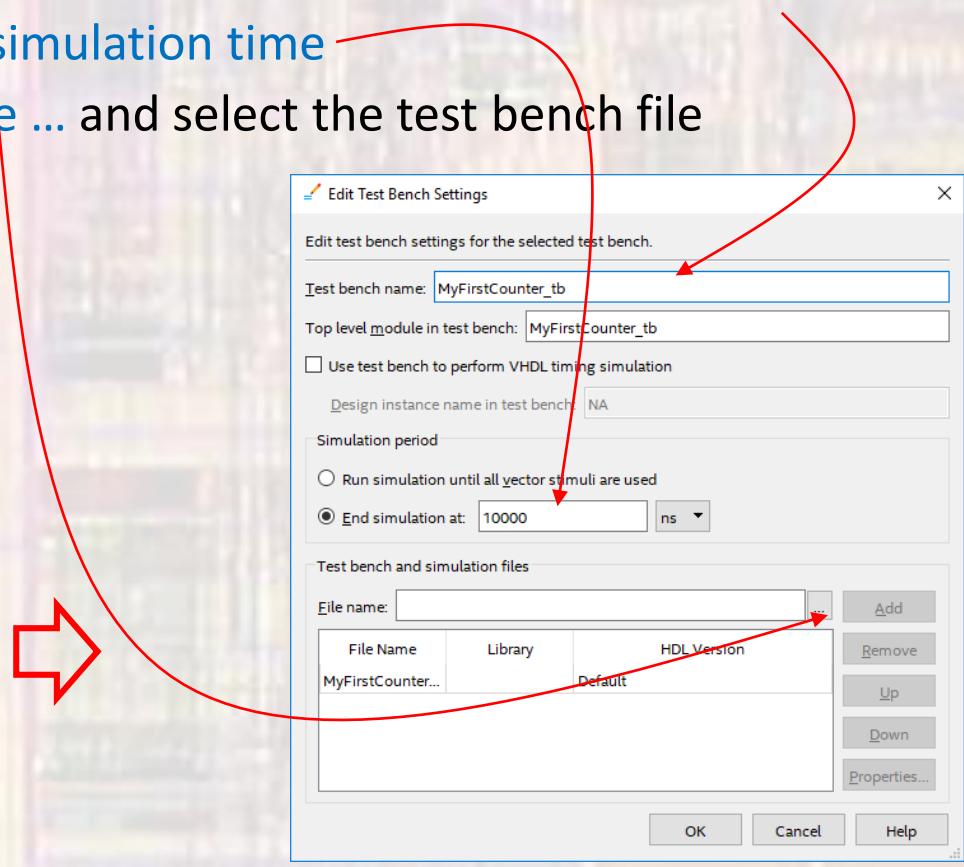
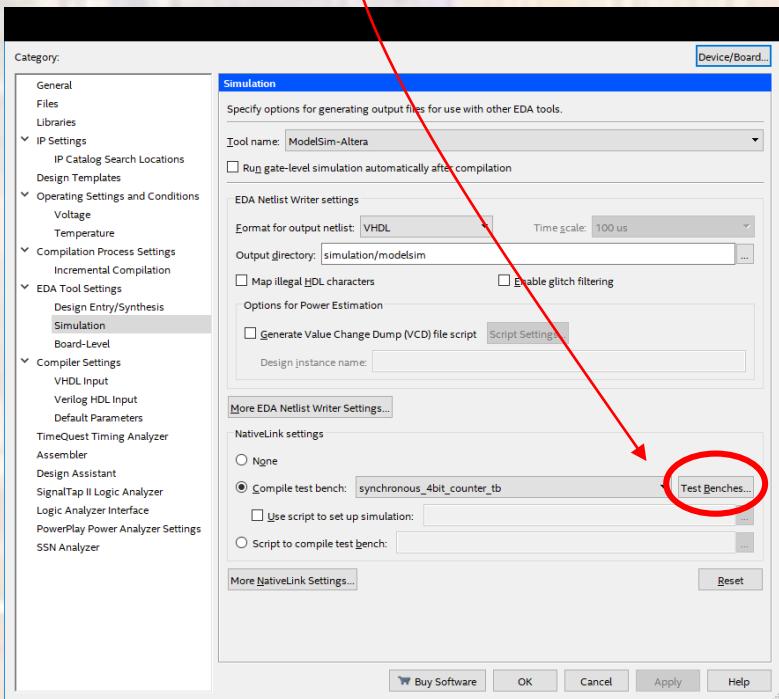
# HDL Project Example

- Project Flow



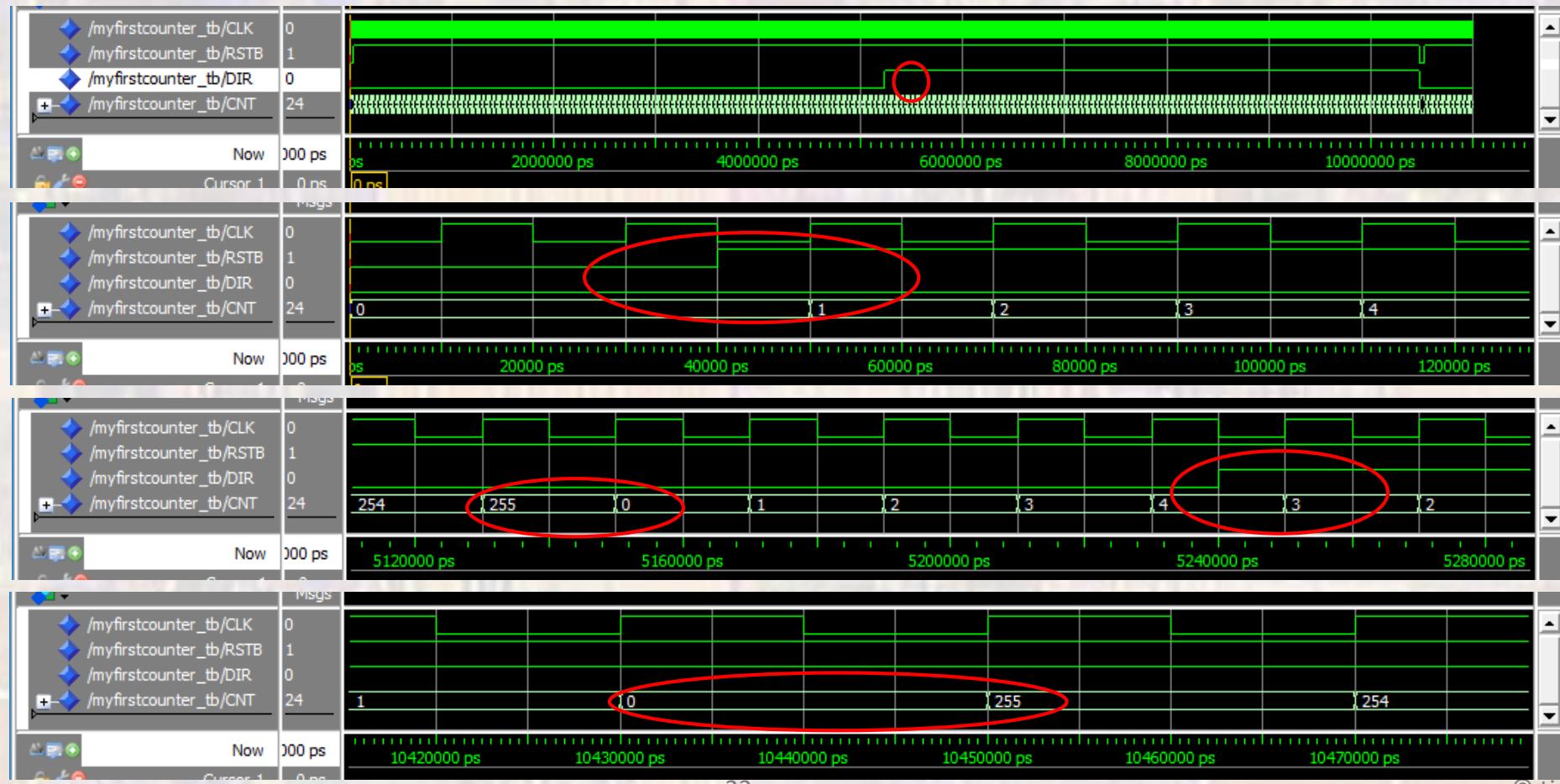
# HDL Project Example

- Project Verification
  - Setup the test bench
    - Select **Assignments** → **Settings** → **EDA Tool Settings** → **Simulation** → **Test Benches** : enter the **test bench file**
    - : select the **end simulation time**
    - : select **File name ...** and select the test bench file



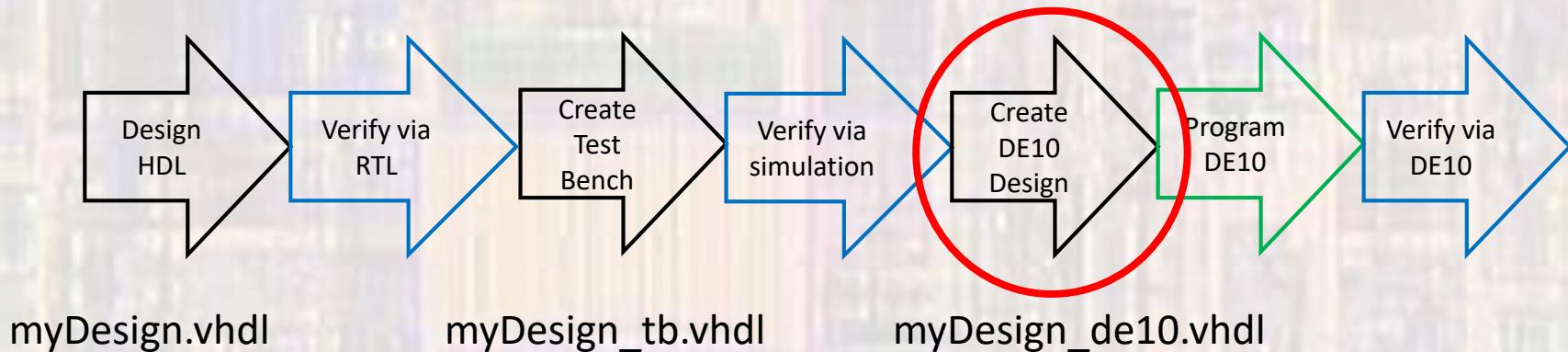
# HDL Project Example

- Project Verification
  - Run the simulation
    - Select Tools → Run Simulation Tool → RTL Simulation



# HDL Project Example

- Project Flow



# HDL Project Example

- Project Implementation
  - Prepare for DE10
  - Create a new VHDL File
    - MyFirstCounter\_de10.vhdl
  - Use the DE10 pin names from the qsf file
    - or you can use the pin planner
  - Instantiate your Design
  - Instantiate a clock divider
    - slow down the operation so a human can see it

# HDL Project Example

- Project Implementation
  - Prepare for DE10

```
-- MyFirstCounter_de10.vhd
-- created: 2/23/18
-- by: johnsontimoj
-- rev: 0
-- DE10 file for 8 bit counter
-- of MyFirstCounter.vhd
-- using qsfp pin names
-- SW[0] --> reset
-- SW[1] --> Dir
-- CLOCK_50 --> clock
-- LEDR --> cnt
--

library ieee;
use ieee.std_logic_1164.all;

entity MyFirstCounter_de10 is
  PORT(
    SW:      in std_logic_vector(1 downto 0);
    CLOCK_50: in std_logic;
    LEDR:      out std_logic_vector(7 downto 0)
  );
end entity;
```

```
architecture testbench of MyFirstCounter_de10 is
  signal clock10Hz: std_logic;

  -- Component prototypes
  COMPONENT MyFirstCounter
    PORT
    (
      i_rstb : IN STD_LOGIC;
      i_clk : IN STD_LOGIC;
      i_dir : IN STD_LOGIC;
      o_cnt : OUT STD_LOGIC_VECTOR(7 DOWNTO 0)
    );
  END COMPONENT;
  COMPONENT clk10Hz
    PORT
    (
      i_clk_50MHz : IN STD_LOGIC;
      i_rstb : IN STD_LOGIC;
      o_clk_10Hz : OUT STD_LOGIC
    );
  END COMPONENT;
```

```
-- Design placement
DESIGN: MyFirstCounter
port map(
  i_rstb  => SW(0),
  i_clk   => clock10Hz,
  i_dir   => SW(1),
  o_cnt   => LEDR
);

clk_divider: clk10Hz
port map(
  i_clk_50MHz => CLOCK_50,
  o_clk_10Hz  => clock10Hz,
  i_rstb     => SW(0)
);

end architecture;
```

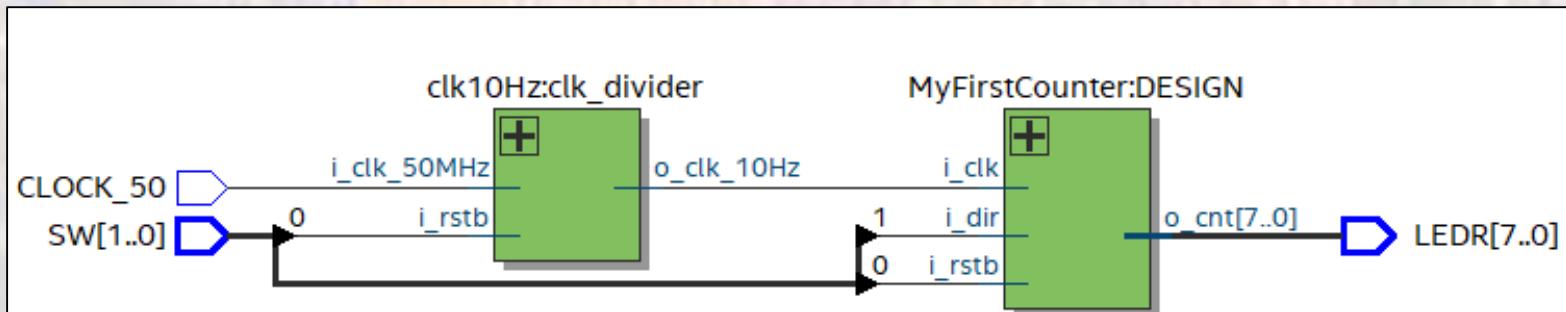
# Combinational Project Example

- Project Implementation
  - Prepare for DE10
  - Compile your design
  - Check the Compilation Summary

Flow Summary	
	<<Filter>>
Flow Status	Successful - Fri Feb 23 20:28:40 2018
Quartus Prime Version	17.1.0 Build 590 10/25/2017 SJ Lite Edition
Revision Name	MyFirstDE10Project
Top-level Entity Name	MyFirstCounter_de10
Family	MAX 10
Device	10M50DAF484C7G
Timing Models	Final
Total logic elements	48 / 49,760 (< 1 %)
Total registers	31
Total pins	11 / 360 (3 %)
Total virtual pins	0
Total memory bits	0 / 1,677,312 (0 %)
Embedded Multiplier 9-bit elements	0 / 288 (0 %)
Total PLLs	0 / 4 (0 %)
UFM blocks	0 / 1 (0 %)
ADC blocks	0 / 2 (0 %)

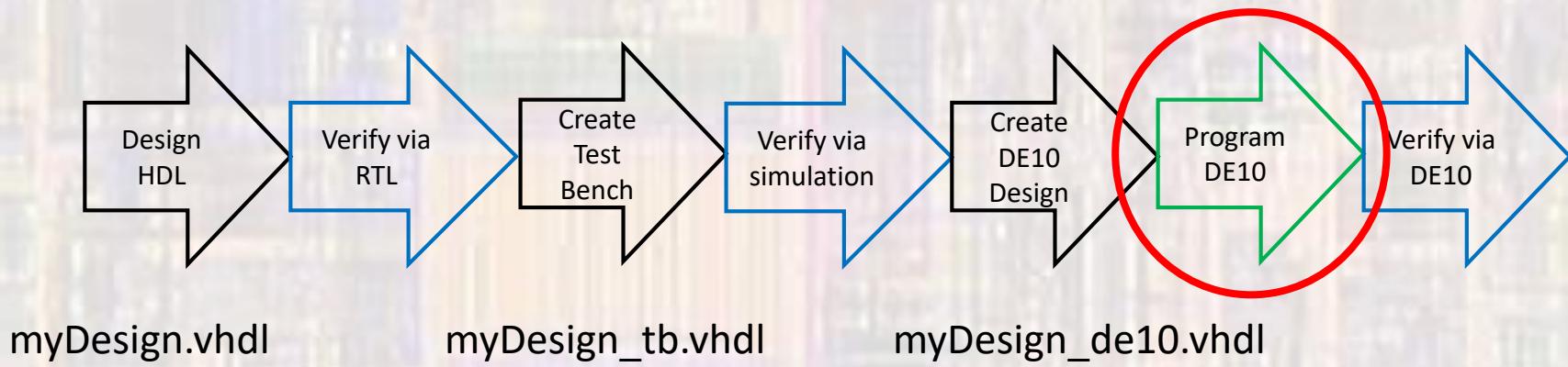
# HDL Project Example

- Project Implementation
  - Prepare for DE10
  - Check the RTL



# HDL Project Example

- Project Flow

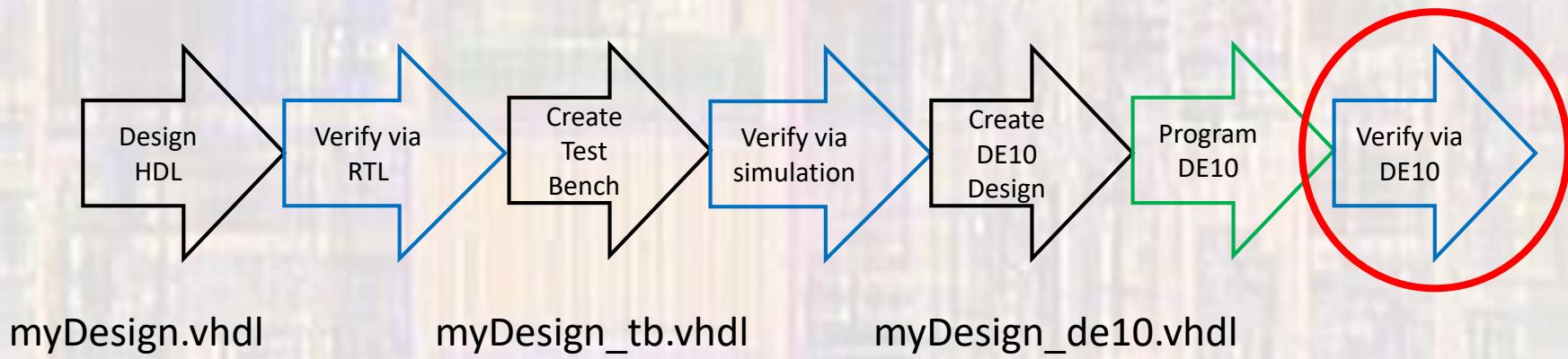


# HDL Project Example

- Project Implementation
  - Configure the DE10
  - Select **Tools -> Programmer**
    - point to the compiled output file
    - start

# HDL Project Example

- Project Flow



# HDL Project Example

- Project Implementation
  - Configure the DE10
  - Validate the design on the DE10 board