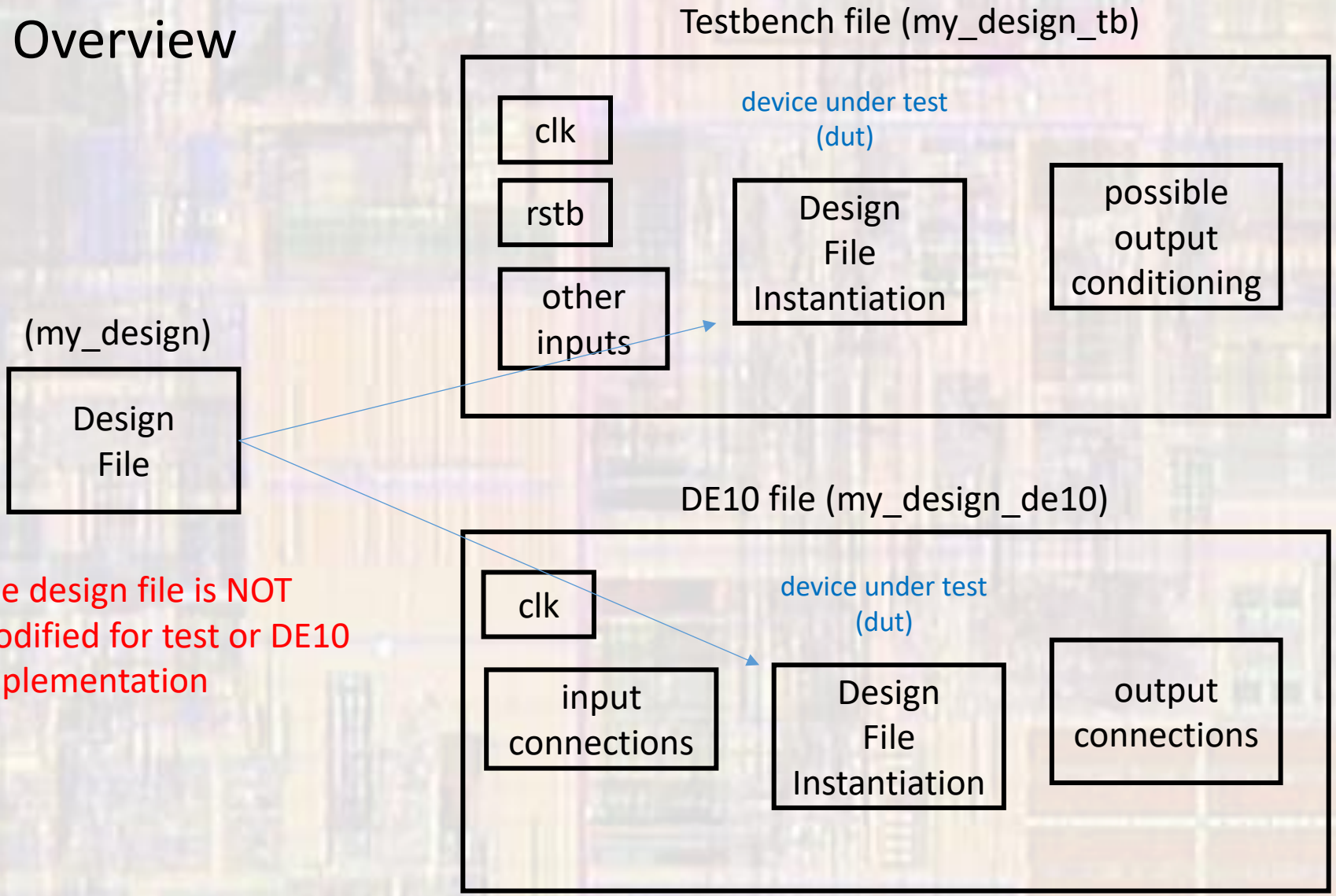
VHDL DE10 Implementation

Last updated 1/21/25

VHDL DE10 Implementation

- Overview



The design file is NOT modified for test or DE10 implementation

VHDL DE10 Implementation

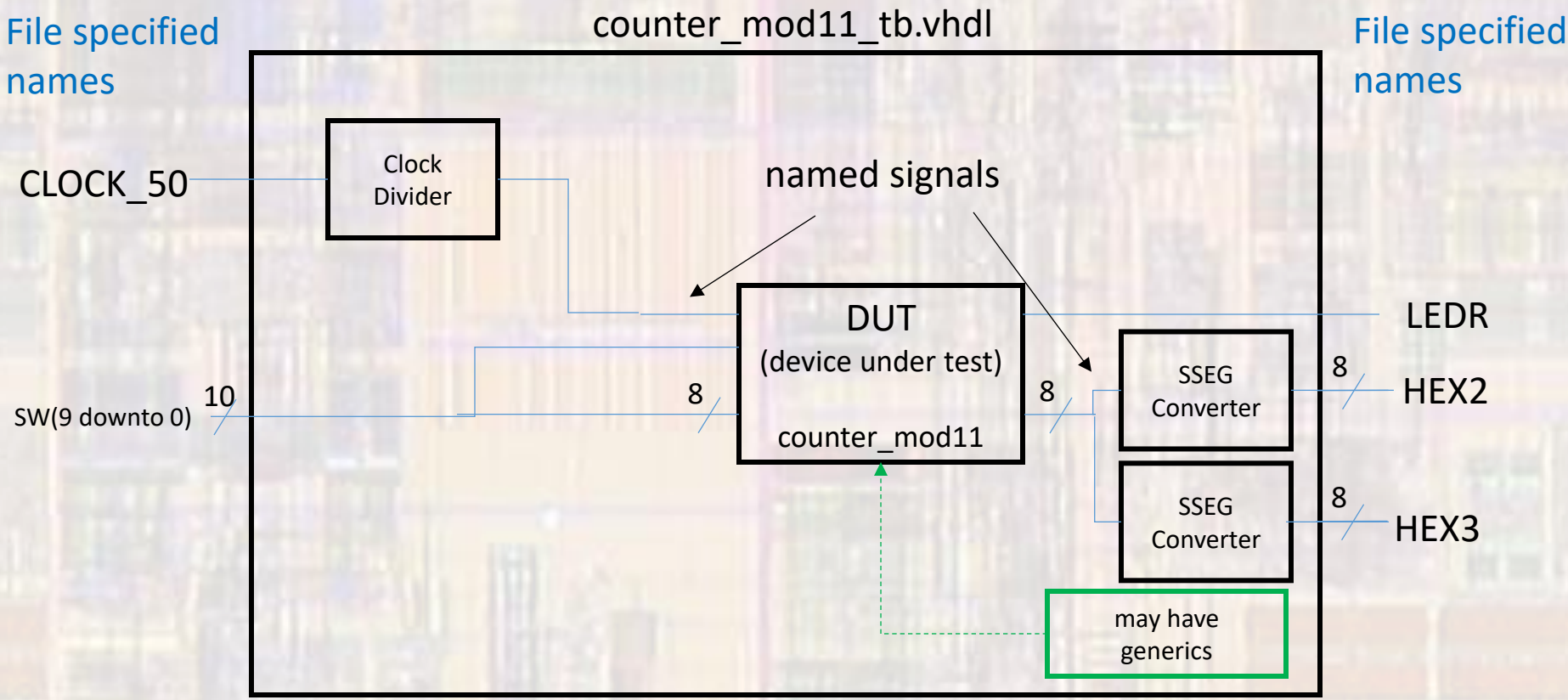
- Quartus Interaction and the DE10
  - Quartus provides a file-based interface to map a VHDL file to the DE10
    - This allows a VHDL design to map to different FPGAs by simply using a different mapping file
  - Each pin on the DE10 family of FPGAs is mapped to a unique name
    - Mapping is stored in the de10.qsf file
    - The de10.qsf file is imported to the project
  - The unique names are used as the port names in the DE10 VHDL implementation file
    - We will use the structure my_top_entity_de10 to name this file

VHDL DE10 Implementation

- DE10 Implementation File
  - The de10 implementation file must be synthesizable
    - Only synthesizable constructs allowed
  - Our best practices
    - Name the de10 file based on the top level entity name
      - add _de10 to the top level entity name
      - e.g. counter_mod11 → counter_mod11_de10
    - If extra processing is required, it is placed in the DE10 file
      - Never modify the top level entity to make it work on the DE10
      - Typical additional blocks include:
        - Clock dividers to allow is to slow the effective clock rate down from 50MHz to just a few Hz
        - Seven Segment Drivers

VHDL DE10 Implementation

- DE10 Graphical Structure



VHDL DE10 Implementation

- DE10 VHDL

```
--
-- counter_updown_signed_nbit_de10.vhdl
--
-- created: 3/30/18
-- by: johnsontimoj
-- rev: 0
--
-- de10 implementation for signed nbit up/down counter
-- of counter_updown_signed_nbit.vhdl
--
--------------------------------
library ieee;
use ieee.std_logic_1164.all;

entity counter_updown_signed_nbit_de10 is
      generic(
                NN:   positive := 4
             );
      port(
         CLOCK_50:   in std_logic;
         SW:         in std_logic_vector(9 downto 0);

         LEDR:       out std_logic_vector(2 downto 2);
         HEX1:       out std_logic_vector(7 downto 0)
      );end entity;

architecture de10 of counter_updown_signed_nbit_de10 is

   signal   CLK_SIG:    std_logic;
   signal   CNT:        std_logic_vector(3 downto 0);

   --------------------------------
   -- Component prototype
   --------------------------------
   component counter_updown_signed_nbit is
      generic(
                N: positive := 8
             );
      port(
         i_clk:   in std_logic;
         i_rstb:  in std_logic;
         i_dir:   in std_logic;

         o_cnt:   out std_logic_vector((N - 1) downto 0)
         );
   end component;

   component sseg is
      port(
            i_nibble_in:   in    std_logic_vector(3 downto 0);
            o_sseg_out:    out   std_logic_vector(7 downto 0)
          );
   end component;

   component clock_1hz is
      port(
            i_clk_50MHz :  in std_logic;
            I_rstb :       in std_logic;

            o_clk_1Hz :    out std_logic
          );
   end component;
   ------------------------------------
```

Description

DE10 generic value

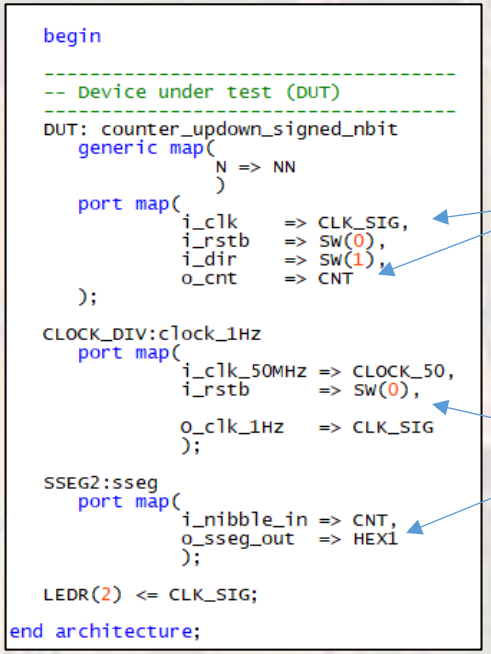
Names from de10.qsf file

Internal signal names

component declarations

VHDL DE10 Implementation

- DE10 VHDL structure



```
begin

-----------------------------------
-- Device under test (DUT)
-----------------------------------
DUT: counter_updown_signed_nbit
   generic map(
               N => NN
               )
   port map(
            i_clk    => CLK_SIG,
            i_rstb   => SW(0),
            i_dir    => SW(1),
            o_cnt    => CNT
   );

CLOCK_DIV:clock_1Hz
   port map(
            i_clk_50MHz => CLOCK_50,
            i_rstb      => SW(0),

            o_clk_1Hz   => CLK_SIG
            );

SSEG2:sseg
   port map(
            i_nibble_in => CNT,
            o_sseg_out  => HEX1
            );

LEDR(2) <= CLK_SIG;

end architecture;
```

map to internal signals

Block instantiations

map to qsf pin names

Bring 1Hz clk out to an LED (heartbeat)

VHDL DE10 Implementation

- DE10 VHDL structure
  - Set the DE10 file as the Top Level Entity