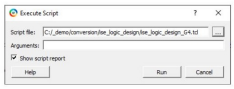
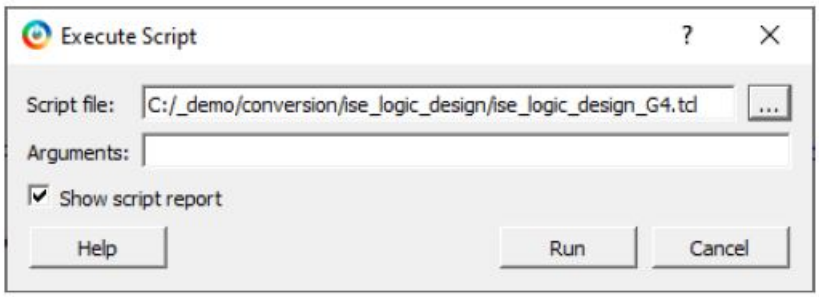


MICROCHIP Xilinx Spartan 6 Example Conversion User Guide

Contents

1 MICROCHIP Xilinx Spartan 6 Example Conversion
2 Create Libero® SoC Design Suite Project
3 Continued
4 Create Shim
4.1 Update Design Hierarchy
5 Constraints
5.1 Create Derived Constraints“
6 Assign Pins
6.1 Implement Design
7 Documents / Resources
7.1 References
8 Related Posts

MICROCHIP Xilinx Spartan 6 Example Conversion

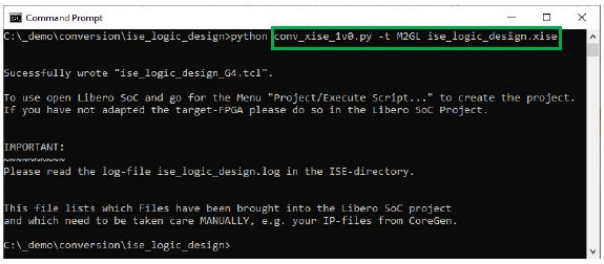


A Leading Provider of Smart, Connected and Secure Embedded Control Solutions

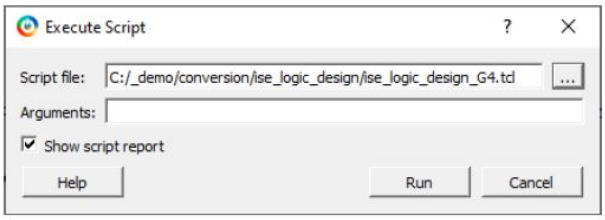
Create Libero® SoC Design Suite Project

Place conversion-script into ISE® project directory

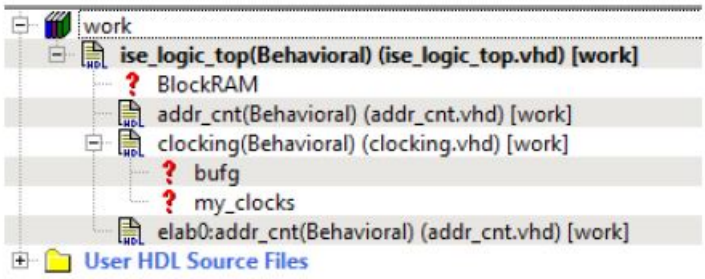
python conv_xise_1v0.py -t <arch> <design>.xise



Open Libero SoC Design Suite and run created TCL-script



Project is created but missing:



- **IP:** BlockRAM, my_clocks
- **Architectural base-blocks:** bufg

Continued

Supported target architectures for conversion

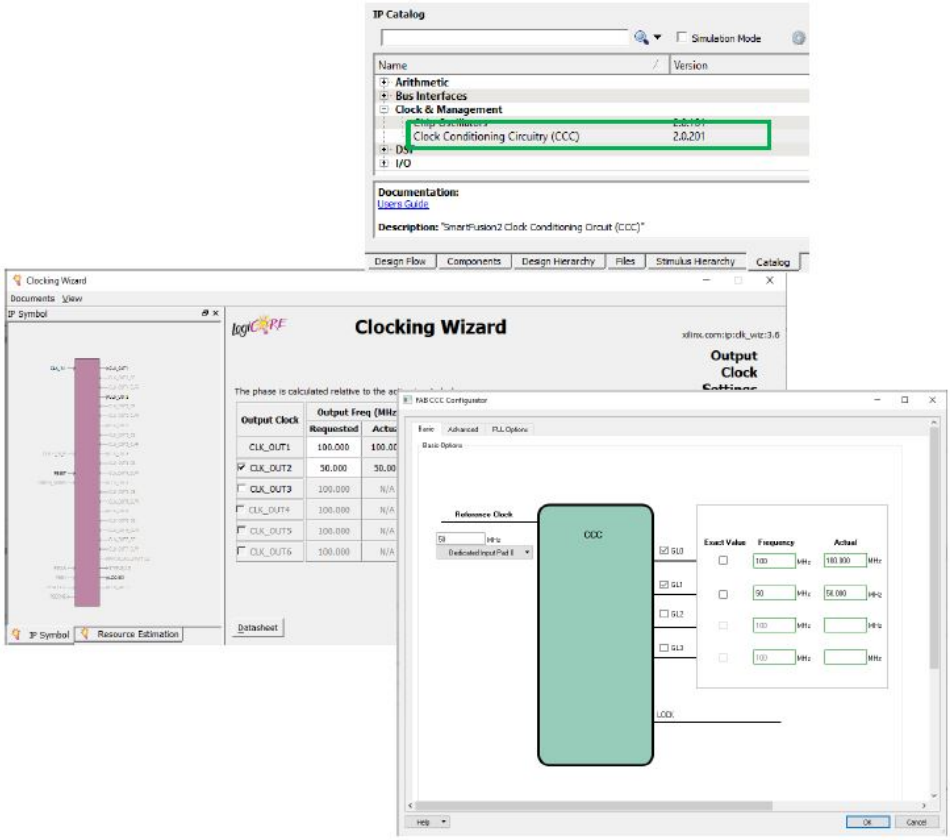
- **MPFS:** PolarFire® SoC
- **MPF:** PolarFire FPGA
- **M2S:** SmartFusion®2
- **M2GL:** IGLOO®2
- **AGL:** IGLOO
- **A3P:** ProASIC®3

IGLOO and ProASIC3 devices require Libero SoC version 11.9 or earlier

Other architectures supported in latest version of Libero SoC

Replace PLLs and DCMs

- Select IP catalog in Libero ® SoC Design Suite



- Create Clock Conditioning Circuit (CCC) for required frequencies
- Choose Advanced“ tab for reset

Replace Individual Clock Buffers

Designs often contain instantiated clock buffers (BUFG)

```
--library UNISIM;
--use UNISIM.VComponents.all;

library smartfusion2;
use smartfusion2.all;
```

Spartan®-6

```
begin
    --main_clocks : my_clocks
    --port map(-- Clock in ports
        --CLK_IN1 => CLK_50M,
        ---- Clock out ports
        --CLK_100M => CLK_100M_o,
        --CLK_50M => CLK_50M_o,
        ---- Status and control signals
        --RESET  => RESET,
        --LOCKED => LOCKED);
--
    main_clocks: FCCC_C0 port map(
        PLL_ARST_N      => not RESET,
        PLL_POWERDOWN_N => '1',
        CLK0_PAD => CLK_50M,
        GL0      => CLK_100M_o,
        GL1      => CLK_50M_o,
        LOCK     => LOCKED);


    --side_clk: BUFG port map(
        --I=> CLK_50M_a,
        --O=> CLK_50M_ind_o
    --);

    side_clk: CLKINT port map(
        A  => CLK_50M_a,
        Y  => CLK_50M_ind_o
    );
```

IGLOO2

Spartan-6

IGLOO2

- Vendor specific libraries
- Unisim => smartfusion, smartfusion2,polarfire

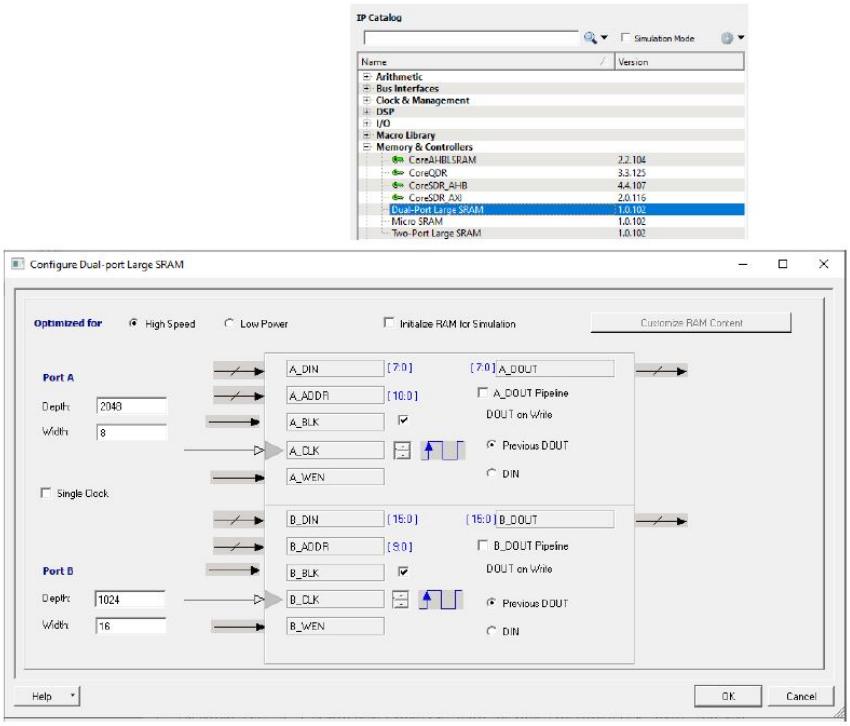
Change of instantiations

- BUFG => CLKINT

Documentation: Macro Library Guide

- SmartFusion®, IGLOO® and ProASIC®3
- SmartFusion2 and IGLOO2
- PolarFire ®

Replace Block RAM



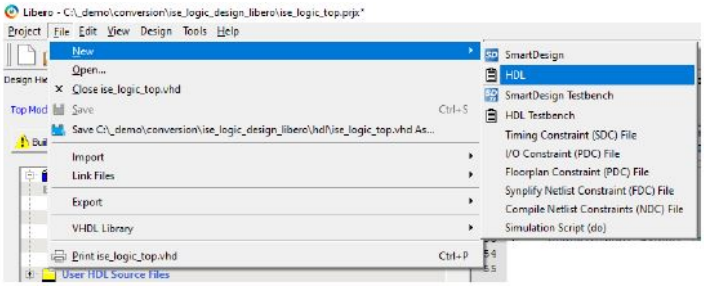
- Create new LSRAM from IP catalog
- Configure LSRAM

Create Shim

- Take existing port map of Block RAM

```
COMPONENT BlockRAM
  PORT (
    clka : IN STD_LOGIC;
    ena : IN STD_LOGIC;
    wea : IN STD_LOGIC_VECTOR(0 DOWNTO 0);
    addra : IN STD_LOGIC_VECTOR(10 DOWNTO 0);
    dina : IN STD_LOGIC_VECTOR(7 DOWNTO 0);
    douta : OUT STD_LOGIC_VECTOR(7 DOWNTO 0);
    clkb : IN STD_LOGIC;
    enb : IN STD_LOGIC;
    web : IN STD_LOGIC_VECTOR(0 DOWNTO 0);
    addrb : IN STD_LOGIC_VECTOR(9 DOWNTO 0);
    dinb : IN STD_LOGIC_VECTOR(15 DOWNTO 0);
    doutb : OUT STD_LOGIC_VECTOR(15 DOWNTO 0)
  );
END COMPONENT;
```

- Create new HDL file

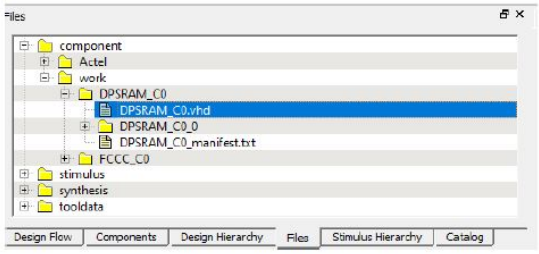



```
library IEEE;

use IEEE.std_logic_1164.all;

entity BlockRAM is
port (
    --<port_name> : <direction> <type>;
    port_name1 : IN  std_logic; -- example
    port_name2 : OUT std_logic_vector(1 downto 0)  -- example
    --<other_ports>;
);
end BlockRAM;
```

- Adapt port map of shim

Instantiate LSRAM into Shim

- Take entity declaration from IP file



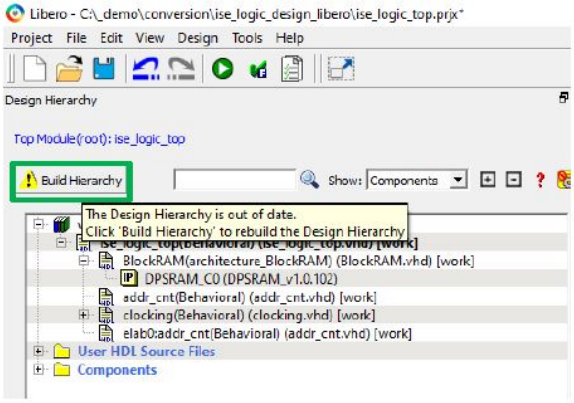

```
BRAM_SHIM: DPSRAM_C0 port map(
    -- Inputs
    A_ADDR => addra,--: in  std_logic_vector(10 downto 0);
    A_BLK  => ena,--: in  std_logic;
    A_CLK  => clka,--: in  std_logic;
    A_DIN  => dina,--: in  std_logic_vector(7 downto 0);
    A_WEN  => wea,--: in  std_logic;
    B_ADDR => addrb,--: in  std_logic_vector(9 downto 0);
    B_BLK  => enb,--: in  std_logic;
    B_CLK  => clkb,--: in  std_logic;
    B_DIN  => dinb,--: in  std_logic_vector(15 downto 0);
    B_WEN  => web,--: in  std_logic;
    -- Outputs
    A_DOUT => douta,--: out std_logic_vector(7 downto 0);
    B_DOUT => doutb--: out std_logic_vector(15 downto 0)
```

- Connect shim ports with instance

Update Design Hierarchy

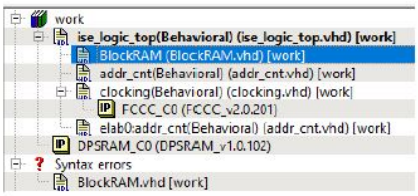
Click Build Hierarchy“

Integration of sources under root design



Correct errors in HDL

```
Reading file 'C:\_demo\conversion\ise_logic_design_libero\hdl\ise_logic_top.vhd'.
Reading file 'C:\_demo\conversion\ise_logic_design_libero\stimulus
\tb_ise_logic_top.vhd'.
The following file(s) have syntax errors:
Error:syntax error near ')': C:/_demo/conversion/ise_logic_design_libero/hdl/
BlockRAM.vhd(75)
```

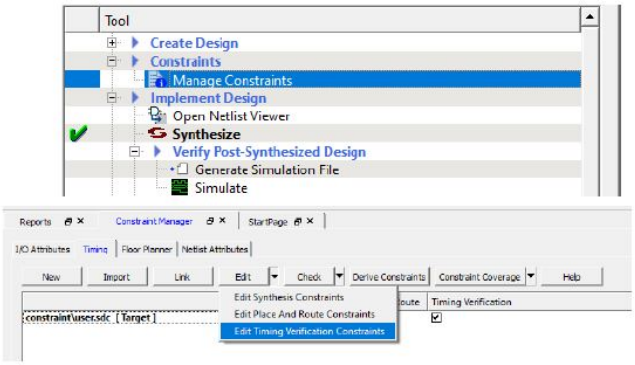


Run synthesis

- Correct potential typos reported by tools

Constraints

Double click Manage Constraints“

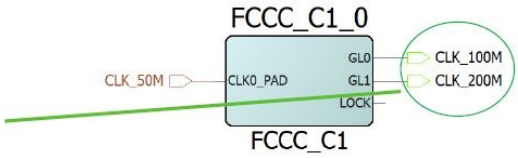


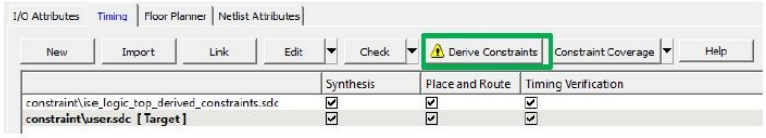
Enter timing constraints

Create Derived Constraints“

Derived constraints:

- Take PLL functionality (multiply/phase shift)
- Constraints “b ehind“ clock modification





Click on “Derive Constraints”

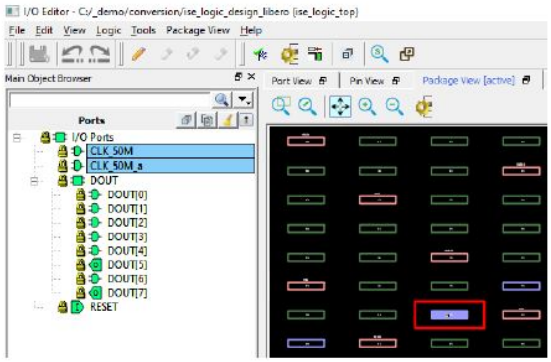
- Populates additional SDC file

Constrain clock domain crossings

Assign Pins

- Constraints manager

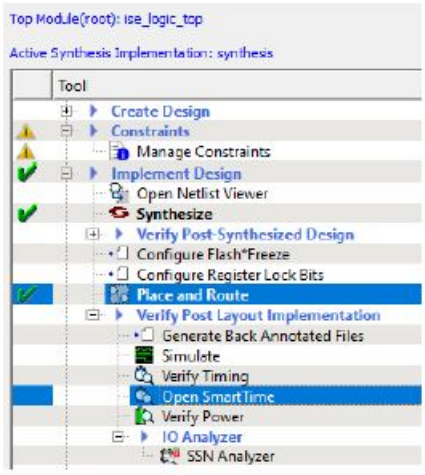
- Pin assignment via table



- Pin assignment via package

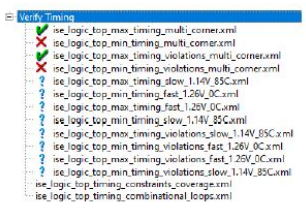
Implement Design

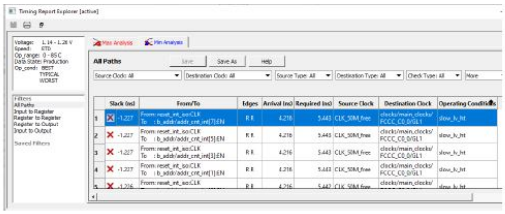
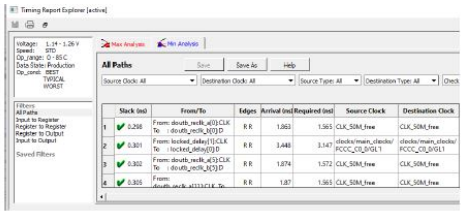
- **Place and route design**



- **Check timing and do timing closure**

  (set_false_path on clock domain



- Create bitstream

Done
Enjoy longevity of your new FPGA design

Documents / Resources

	MICROCHIP Xilinx Spartan 6 Example Conversion [pdf] User Guide Xilinx Spartan 6 Example Conversion, Xilinx, Spartan 6 Example Conversion, Example Conversion

References

- microsemi.com/document-portal/doc_download/1245848-smartfusion2-and-igloo2-macro-library-guide-for-libero-soc-v2021-2
- microsemi.com/document-portal/doc_download/130887-igloo-proasic3-smartfusion-and-fusion-macro-library-guide-for-software-v10-1
- microsemi.com/document-portal/doc_download/136918-polarfire-fpga-macro-library-guide