NIOS Character HW

Last updated 10/12/20

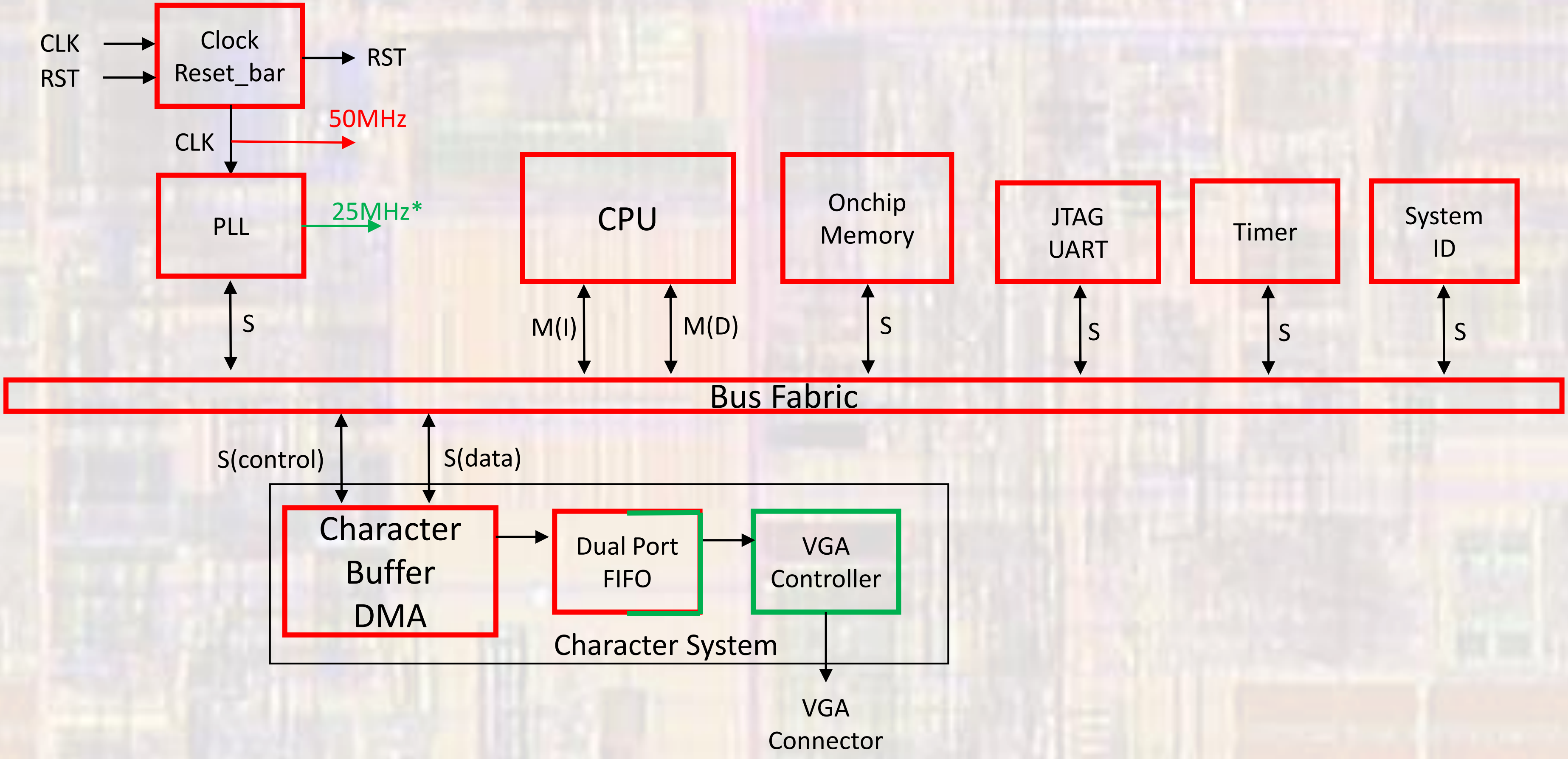
NIOS II Character Display - HW

These slides describe the development of a moderately complex NIOS Processor using the Character Buffer IP

Upon completion: You should be able implement your own NIOS processor using the Character Buffer IP

NIOS II Character Display - HW

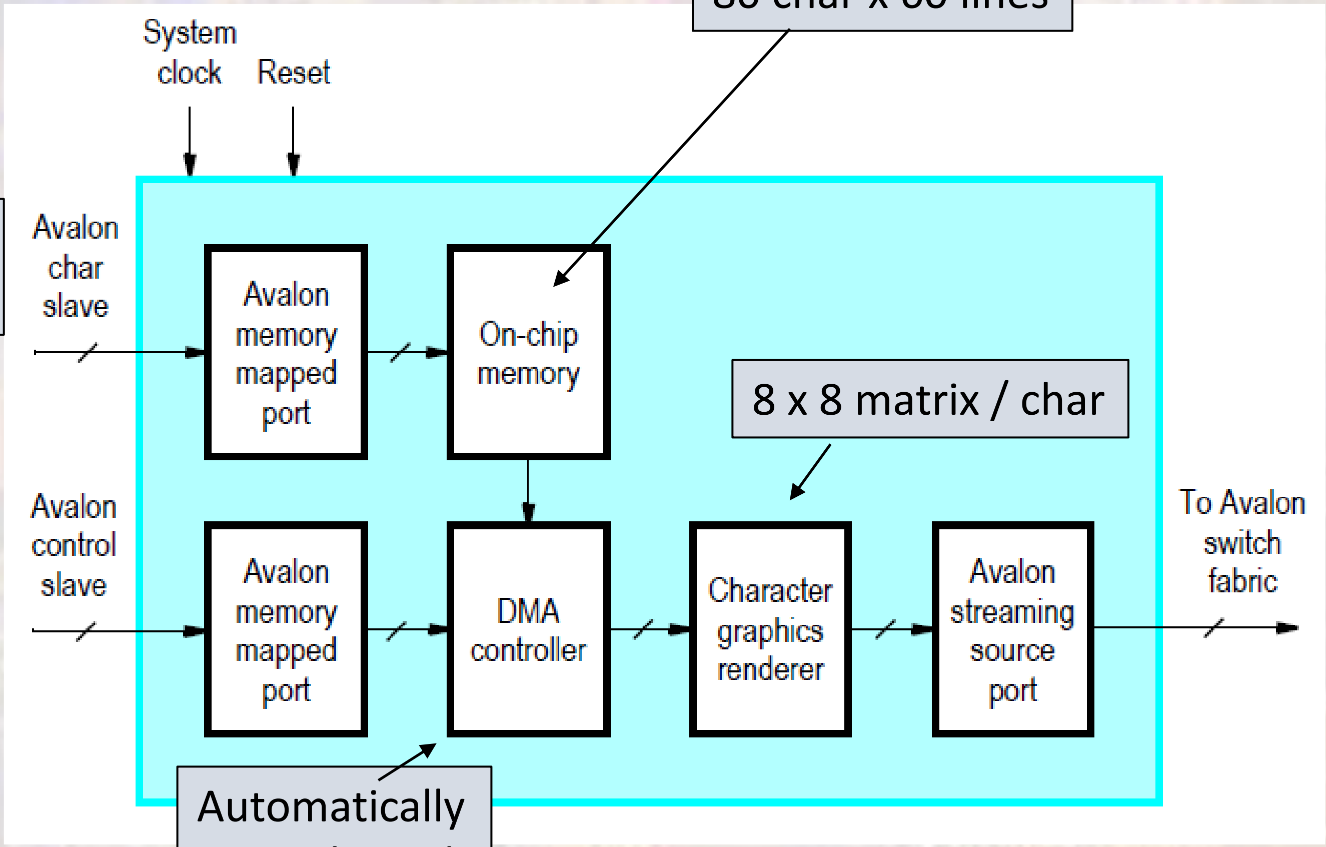
- Character Buffer Block Diagram



25MHz** - VGA clk

NIOS II Character Display - HW

- Character Buffer Block


System clock
Reset
Avalon char slave
Avalon memory mapped port
On-chip memory
Avalon control slave
Avalon memory mapped port
DMA controller
Character graphics renderer
Avalon streaming source port
To Avalon switch fabric

80 char x 60 lines

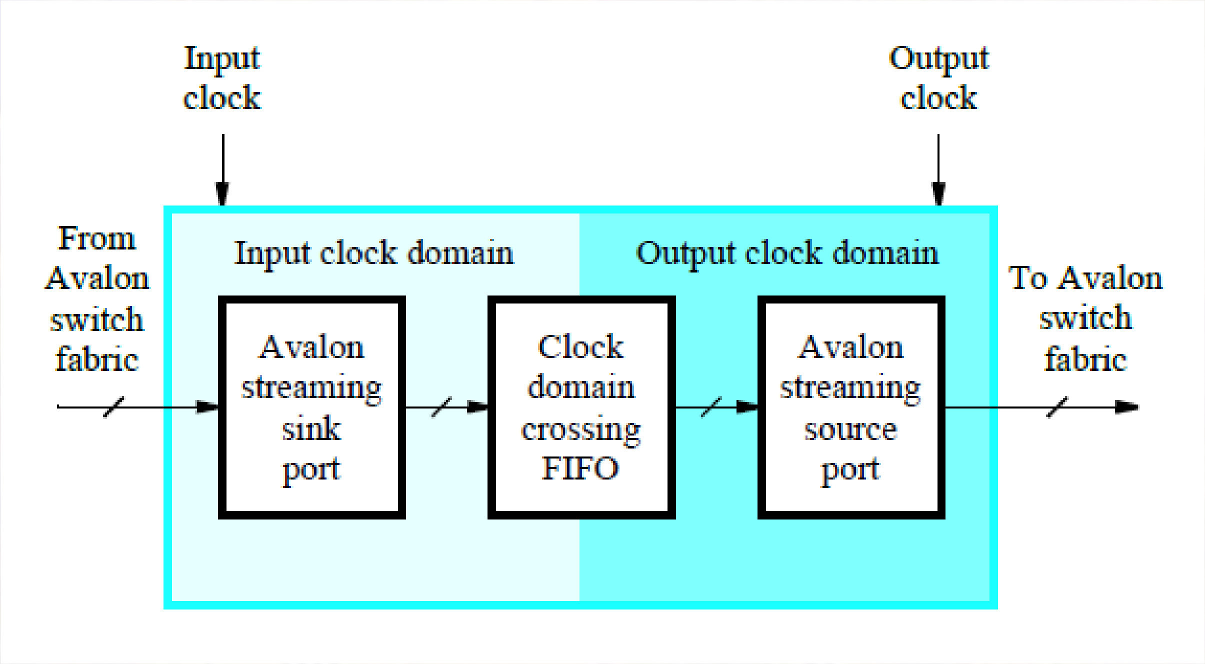
ASCII Character

8 x 8 matrix / char

Automatically scans through memory

NIOS II Character Display - HW

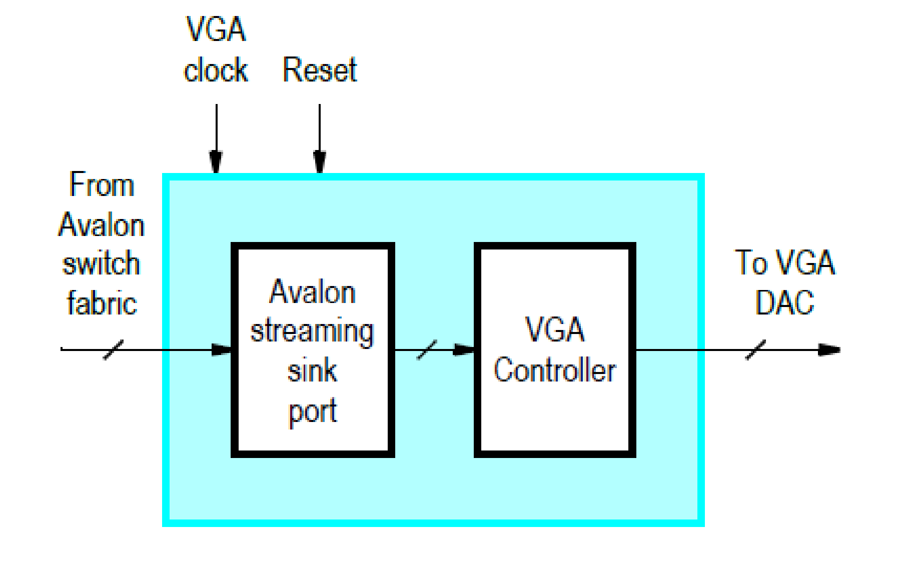
- Dual Clock FIFO


Input clock
Output clock
From Avalon switch fabric
Input clock domain
Output clock domain
To Avalon switch fabric
Avalon streaming sink port
Clock domain crossing FIFO
Avalon streaming source port

Allows different incoming and outgoing data rates

NIOS II Character Display - HW

- VGA Controller Block

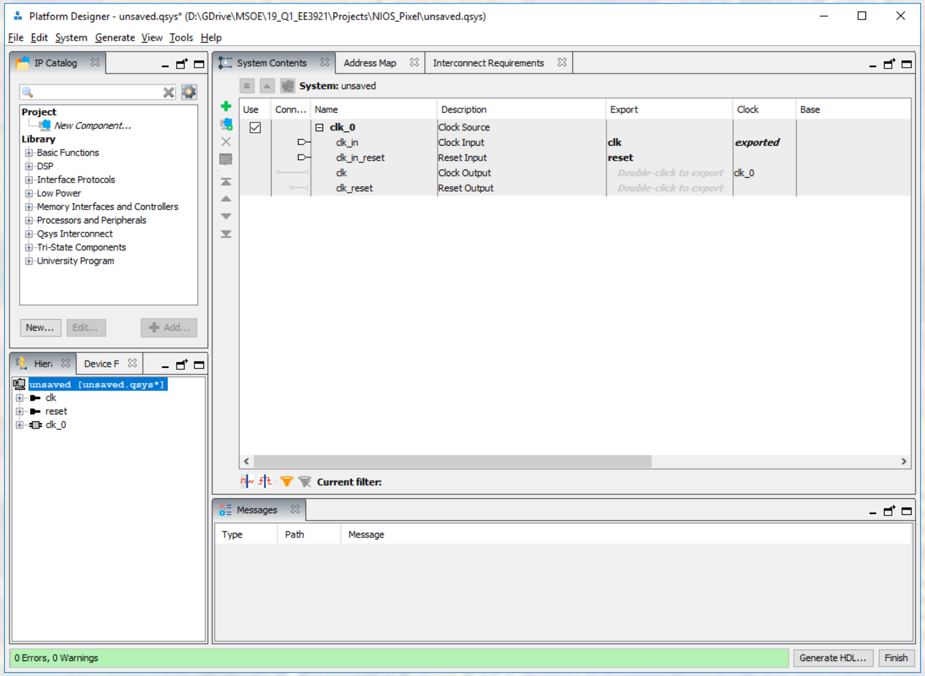


Creates and drives the required VGA signals

NIOS II Character Display - HW

- Create a new Quartus project
  - Do not select a Simulation Tool in EDA Tool Settings

- Open Tools → Platform Designer



System: unsaved

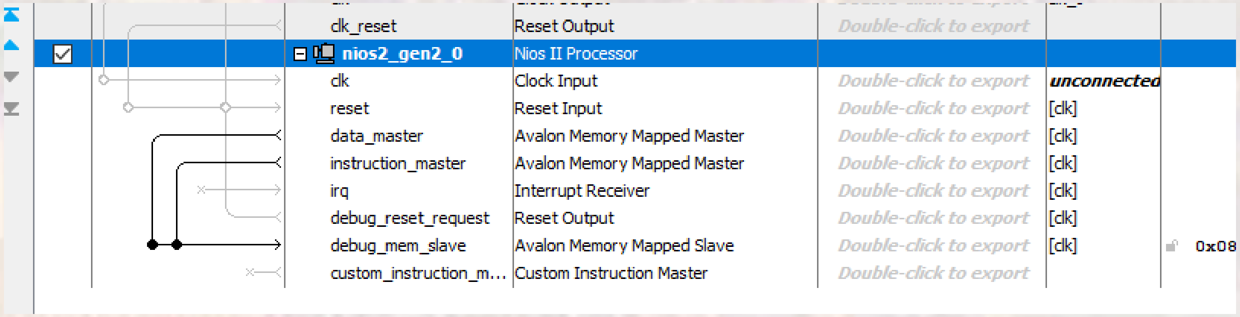
Use	Conn...	Name	Description	Export	Clock	Base
☑		**clk_0**	Clock Source			
		clk_in	Clock Input	**clk**	***exported***	
		clk_in_reset	Reset Input	**reset**		
		clk	Clock Output	*Double-click to export*	clk_0	
		clk_reset	Reset Output	*Double-click to export*		

Type	Path	Message

0 Errors, 0 Warnings

NIOS II Character Display - HW

- Add NIOS
  - Processors and Peripherals → Embedded Processors → NIOS II Processor
  - NIOS II/f

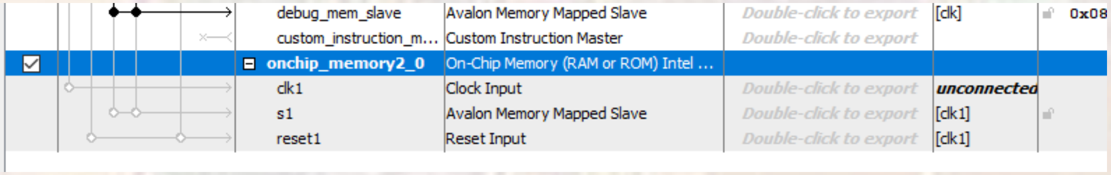


	clk_reset	Reset Output	Double-click to export		
☑	nios2_gen2_0	Nios II Processor			
	clk	Clock Input	Double-click to export	unconnected	
	reset	Reset Input	Double-click to export	[clk]	
	data_master	Avalon Memory Mapped Master	Double-click to export	[clk]	
	instruction_master	Avalon Memory Mapped Master	Double-click to export	[clk]	
	irq	Interrupt Receiver	Double-click to export	[clk]	
	debug_reset_request	Reset Output	Double-click to export	[clk]	
	debug_mem_slave	Avalon Memory Mapped Slave	Double-click to export	[clk]	0x08
	custom_instruction_m...	Custom Instruction Master	Double-click to export		

- Add On-chip Memory
  - Basic Functions→ On Chip Memory → On Chip Memory (RAM or ROM)…

RAM

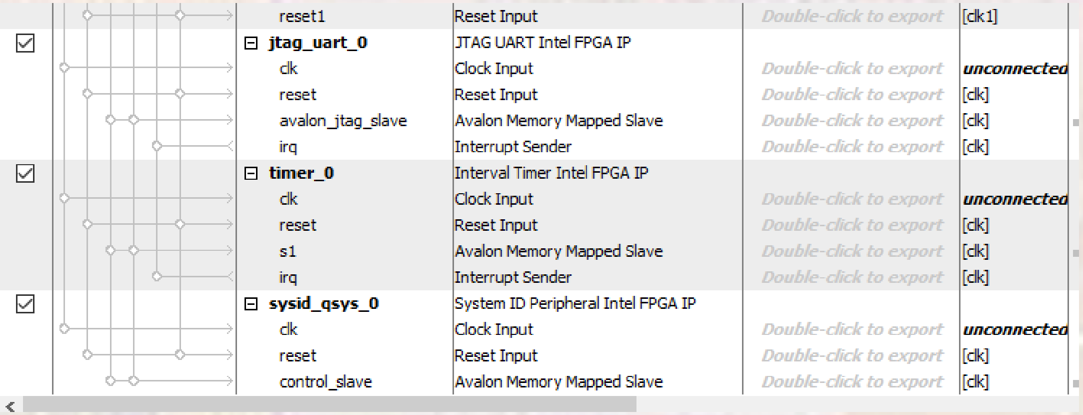
Size = 12,000 bytes



	debug_mem_slave	Avalon Memory Mapped Slave	Double-click to export	[clk]	0x08
	custom_instruction_m...	Custom Instruction Master	Double-click to export		
☑	onchip_memory2_0	On-Chip Memory (RAM or ROM) Intel ...			
	clk1	Clock Input	Double-click to export	unconnected	
	s1	Avalon Memory Mapped Slave	Double-click to export	[clk1]	
	reset1	Reset Input	Double-click to export	[clk1]	

NIOS II Character Display - HW

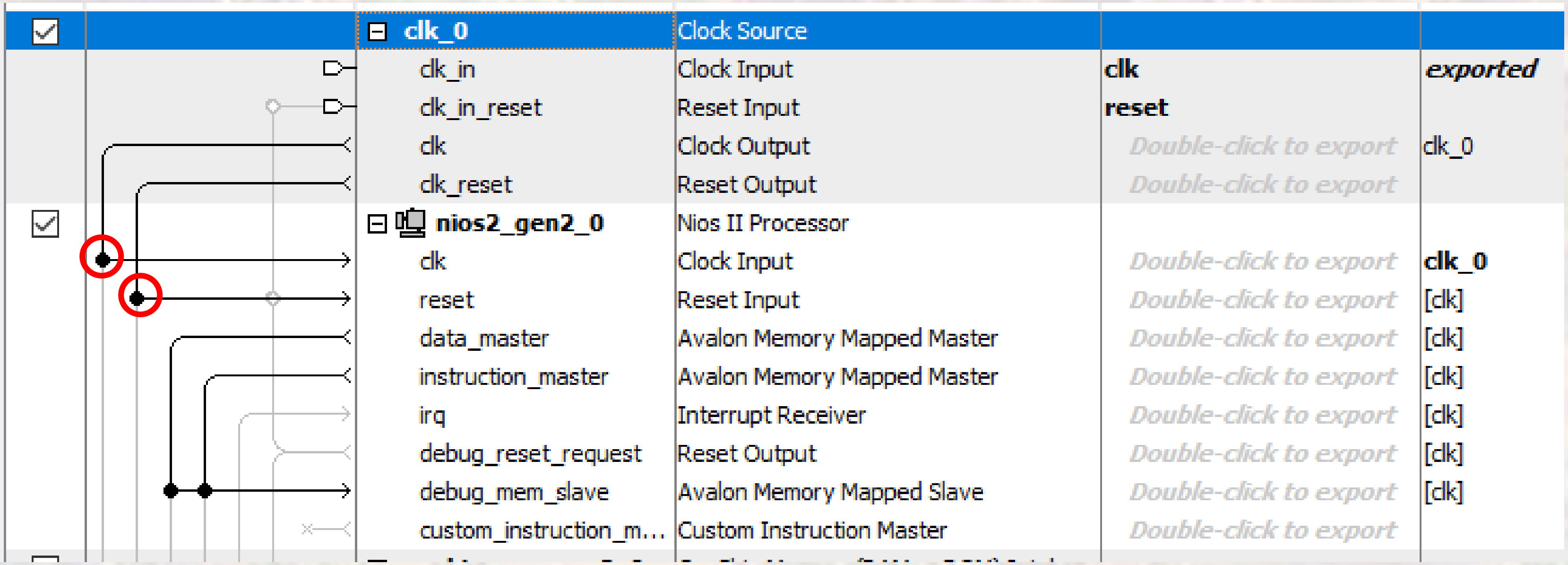
- Add JTAG
  - Interface Protocols → Serial→ JTAG Uart Intel FPGA IP
- Add Timer
  - Processors and Peripherals → Peripherals → Interval Timer Intel FPGA IP
- Add System ID
  - Basic Functions → Simulation; Debug and Verification → Debug and Performance → System ID Peripheral Intel FPGA IP



	reset1	Reset Input	Double-click to export	[clk1]
☑	⊟ **jtag_uart_0**	JTAG UART Intel FPGA IP		
	clk	Clock Input	Double-click to export	*unconnected*
	reset	Reset Input	Double-click to export	[clk]
	avalon_jtag_slave	Avalon Memory Mapped Slave	Double-click to export	[clk]
	irq	Interrupt Sender	Double-click to export	[clk]
☑	⊟ **timer_0**	Interval Timer Intel FPGA IP		
	clk	Clock Input	Double-click to export	*unconnected*
	reset	Reset Input	Double-click to export	[clk]
	s1	Avalon Memory Mapped Slave	Double-click to export	[clk]
	irq	Interrupt Sender	Double-click to export	[clk]
☑	⊟ **sysid_qsys_0**	System ID Peripheral Intel FPGA IP		
	clk	Clock Input	Double-click to export	*unconnected*
	reset	Reset Input	Double-click to export	[clk]
	control_slave	Avalon Memory Mapped Slave	Double-click to export	[clk]

NIOS II Character Display - HW

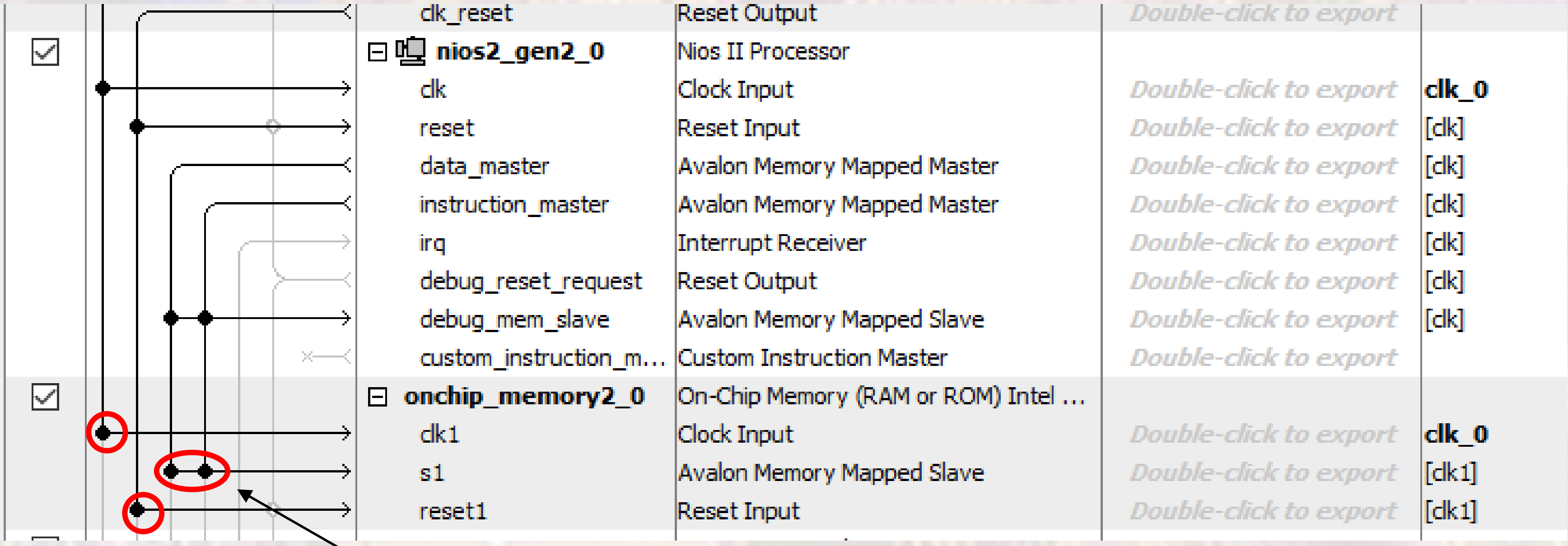
- Connect up basic NIOS system
  - NIOS Inputs



☑	⊟ clk_0	Clock Source		
	clk_in	Clock Input	clk	*exported*
	clk_in_reset	Reset Input	reset	
	clk	Clock Output	*Double-click to export*	clk_0
	clk_reset	Reset Output	*Double-click to export*	
☑	⊟ nios2_gen2_0	Nios II Processor		
	clk	Clock Input	*Double-click to export*	clk_0
	reset	Reset Input	*Double-click to export*	[clk]
	data_master	Avalon Memory Mapped Master	*Double-click to export*	[clk]
	instruction_master	Avalon Memory Mapped Master	*Double-click to export*	[clk]
	irq	Interrupt Receiver	*Double-click to export*	[clk]
	debug_reset_request	Reset Output	*Double-click to export*	[clk]
	debug_mem_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]
	custom_instruction_m...	Custom Instruction Master	*Double-click to export*	

NIOS II Character Display - HW

- Connect up basic NIOS system
  - On-chip Memory

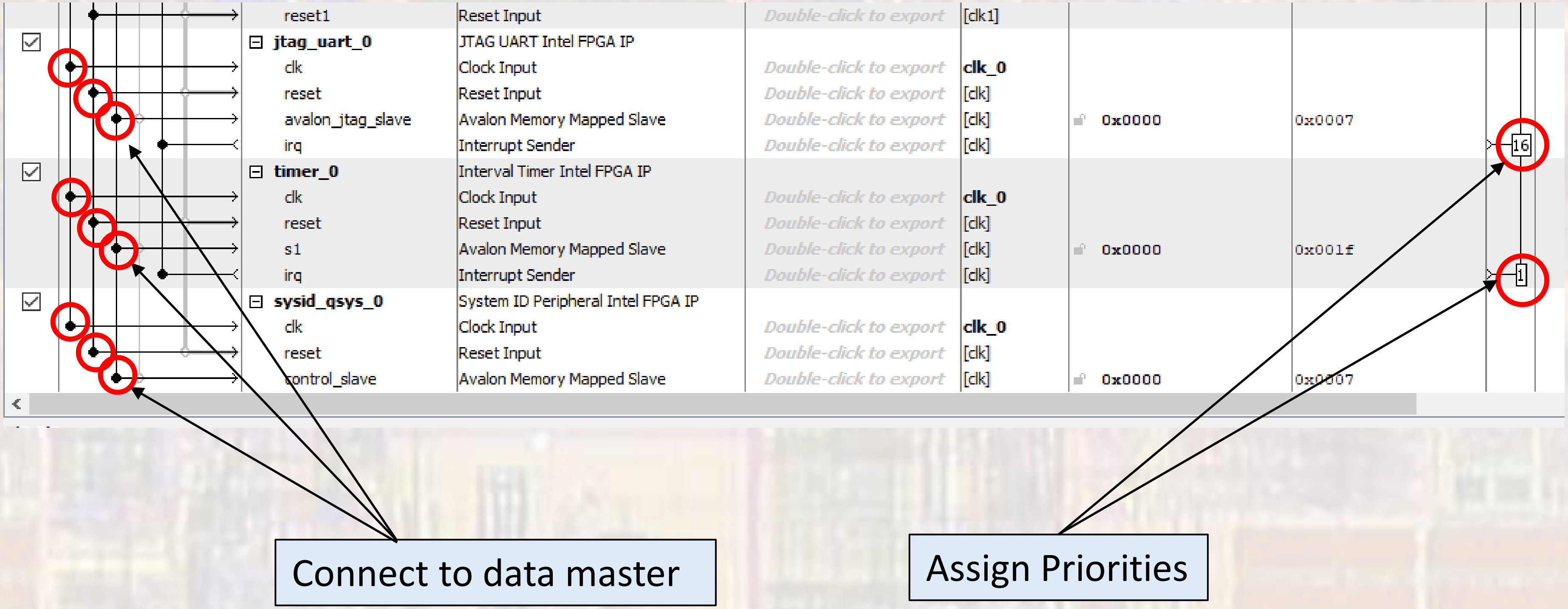


	clk_reset	Reset Output	Double-click to export	
☑	⊟ nios2_gen2_0	Nios II Processor		
	clk	Clock Input	Double-click to export	clk_0
	reset	Reset Input	Double-click to export	[clk]
	data_master	Avalon Memory Mapped Master	Double-click to export	[clk]
	instruction_master	Avalon Memory Mapped Master	Double-click to export	[clk]
	irq	Interrupt Receiver	Double-click to export	[clk]
	debug_reset_request	Reset Output	Double-click to export	[clk]
	debug_mem_slave	Avalon Memory Mapped Slave	Double-click to export	[clk]
	custom_instruction_m...	Custom Instruction Master	Double-click to export	
☑	⊟ onchip_memory2_0	On-Chip Memory (RAM or ROM) Intel ...		
	clk1	Clock Input	Double-click to export	clk_0
	s1	Avalon Memory Mapped Slave	Double-click to export	[clk1]
	reset1	Reset Input	Double-click to export	[clk1]

Connect to data and instruction masters

NIOS II Character Display - HW

- Connect up basic NIOS system
  - JTAG, Timer, SysID



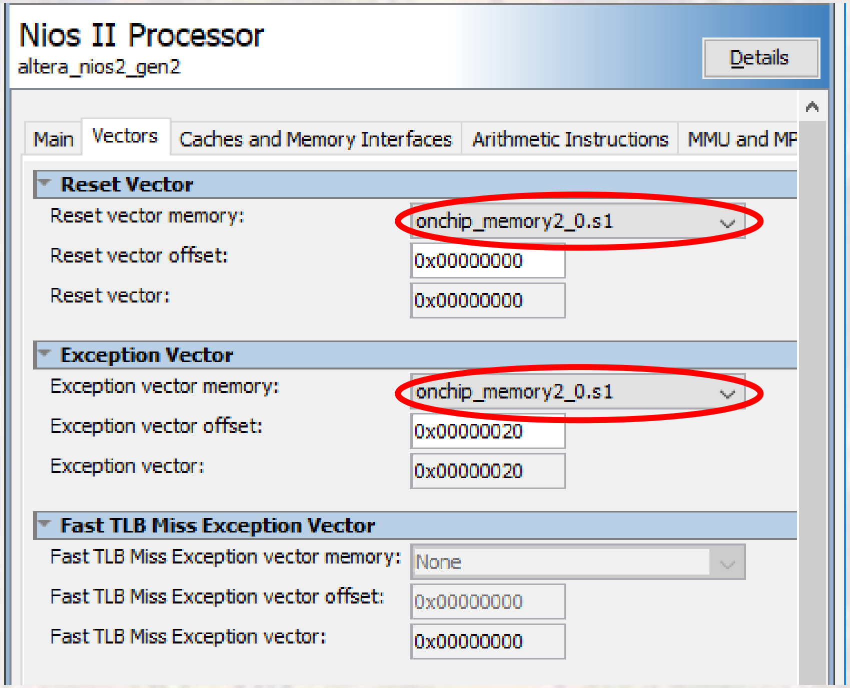
	reset1	Reset Input	*Double-click to export*	[clk1]			
☑	⊟ **jtag_uart_0**	JTAG UART Intel FPGA IP					
	clk	Clock Input	*Double-click to export*	**clk_0**			
	reset	Reset Input	*Double-click to export*	[clk]			
	avalon_jtag_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x0000	0x0007	
	irq	Interrupt Sender	*Double-click to export*	[clk]			16
☑	⊟ **timer_0**	Interval Timer Intel FPGA IP					
	clk	Clock Input	*Double-click to export*	**clk_0**			
	reset	Reset Input	*Double-click to export*	[clk]			
	s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x0000	0x001f	
	irq	Interrupt Sender	*Double-click to export*	[clk]			1
☑	⊟ **sysid_qsys_0**	System ID Peripheral Intel FPGA IP					
	clk	Clock Input	*Double-click to export*	**clk_0**			
	reset	Reset Input	*Double-click to export*	[clk]			
	control_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x0000	0x0007	

Connect to data master

Assign Priorities

NIOS II Character Display - HW

- Connect up basic NIOS system
  - Assign the NIOS II Reset and Exception vectors



Nios II Processor
altera_nios2_gen2

Reset Vector

Reset vector memory: onchip_memory2_0.s1

Reset vector offset: 0x00000000

Reset vector: 0x00000000

Exception Vector

Exception vector memory: onchip_memory2_0.s1

Exception vector offset: 0x00000020

Exception vector: 0x00000020

Fast TLB Miss Exception Vector

Fast TLB Miss Exception vector memory: None

Fast TLB Miss Exception vector offset: 0x00000000

Fast TLB Miss Exception vector: 0x00000000

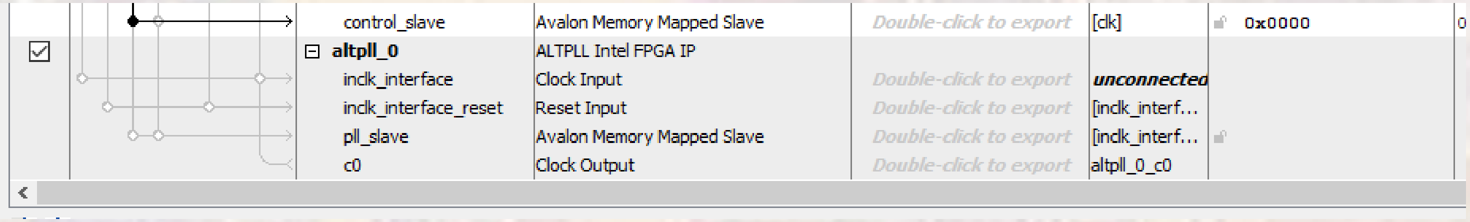
NIOS II Character Display - HW

- Create Character System
  - Add a PLL
    - Basic Functions → Clocks; PLLs and Resets → PLL → ALTPLL Intel FPGA IP

50MHz input clock

no areset or locked output

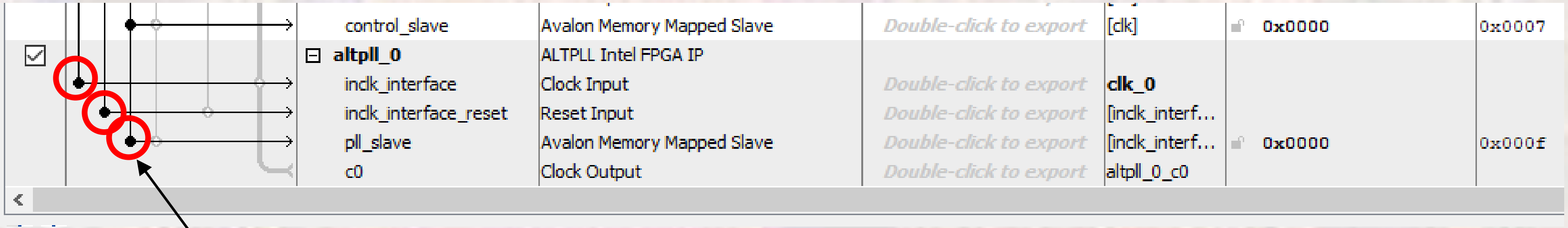
c0 → 25MHz : the VGA frequency



	control_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x0000
☑	⊟ **altpll_0**	ALTPLL Intel FPGA IP			
	inclk_interface	Clock Input	*Double-click to export*	***unconnected***	
	inclk_interface_reset	Reset Input	*Double-click to export*	[inclk_interf...	
	pll_slave	Avalon Memory Mapped Slave	*Double-click to export*	[inclk_interf...	
	c0	Clock Output	*Double-click to export*	altpll_0_c0	

NIOS II Character Display - HW

- Create Character System
  - Connect PLL

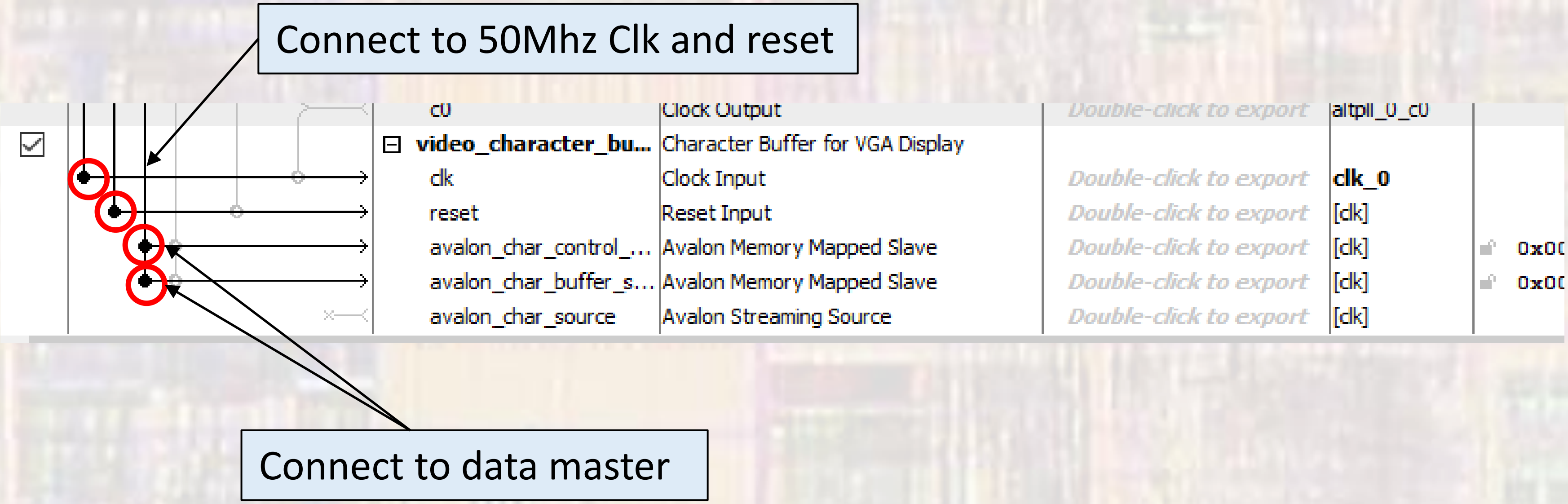


	control_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x0000	0x0007
☑	⊟ **altpll_0**	ALTPLL Intel FPGA IP				
	inclk_interface	Clock Input	*Double-click to export*	**clk_0**		
	inclk_interface_reset	Reset Input	*Double-click to export*	[inclk_interf...		
	pll_slave	Avalon Memory Mapped Slave	*Double-click to export*	[inclk_interf...	0x0000	0x000f
	c0	Clock Output	*Double-click to export*	altpll_0_c0		

Connect to the 50MHz Clk, reset and the data master

NIOS II Character Display - HW

- Create Character System
  - Character Buffer
    - University Program → Audio and Video → Video → Character Buffer for VGA Display



	c0	Clock Output	Double-click to export	altpll_0_c0	
☑	⊟ **video_character_bu...**	Character Buffer for VGA Display			
	clk	Clock Input	*Double-click to export*	**clk_0**	
	reset	Reset Input	*Double-click to export*	[clk]	
	avalon_char_control_...	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x00
	avalon_char_buffer_s...	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x00
	avalon_char_source	Avalon Streaming Source	*Double-click to export*	[clk]	

Connect to 50Mhz Clk and reset

Connect to data master

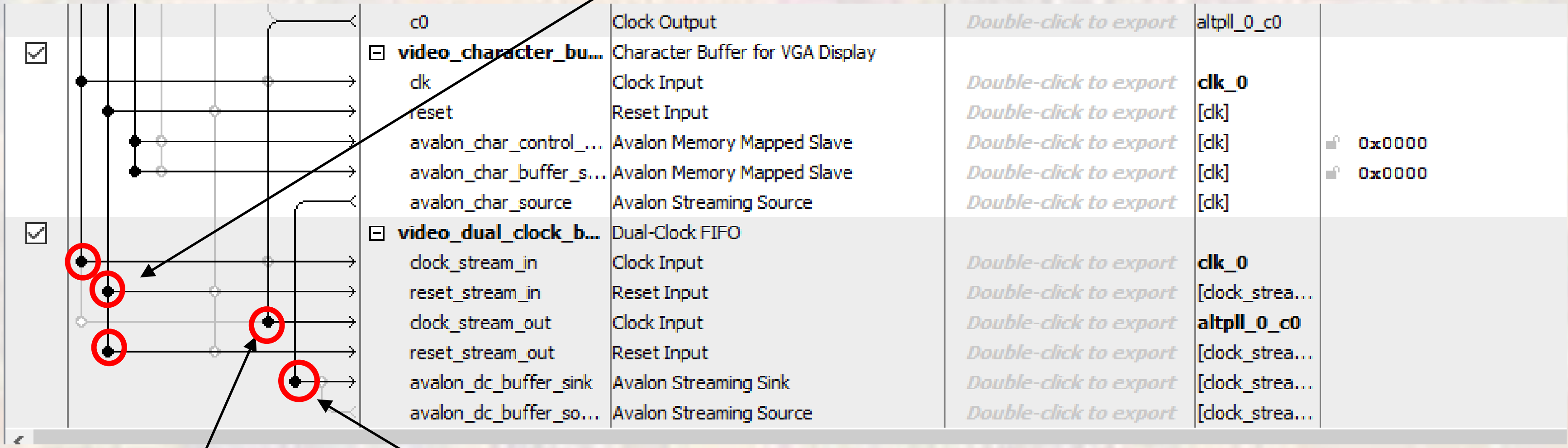
NIOS II Character Display - HW

- Create Character System
  - Dual Clock FIFO
    - University Program → Audio and Video → Video → Dual Clock FIFO

      Color Bits – 10

      Color Planes - 3

Connect to 50Mhz Clk and reset



	c0	Clock Output	*Double-click to export*	altpll_0_c0	
☑	⊟ **video_character_bu...**	Character Buffer for VGA Display			
	clk	Clock Input	*Double-click to export*	**clk_0**	
	reset	Reset Input	*Double-click to export*	[clk]	
	avalon_char_control_...	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x0000
	avalon_char_buffer_s...	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x0000
	avalon_char_source	Avalon Streaming Source	*Double-click to export*	[clk]	
☑	⊟ **video_dual_clock_b...**	Dual-Clock FIFO			
	clock_stream_in	Clock Input	*Double-click to export*	**clk_0**	
	reset_stream_in	Reset Input	*Double-click to export*	[clock_strea...	
	clock_stream_out	Clock Input	*Double-click to export*	**altpll_0_c0**	
	reset_stream_out	Reset Input	*Double-click to export*	[clock_strea...	
	avalon_dc_buffer_sink	Avalon Streaming Sink	*Double-click to export*	[clock_strea...	
	avalon_dc_buffer_so...	Avalon Streaming Source	*Double-click to export*	[clock_strea...	

Connect to c0
VGA – 25MHz

Connect to video character buffer streaming source

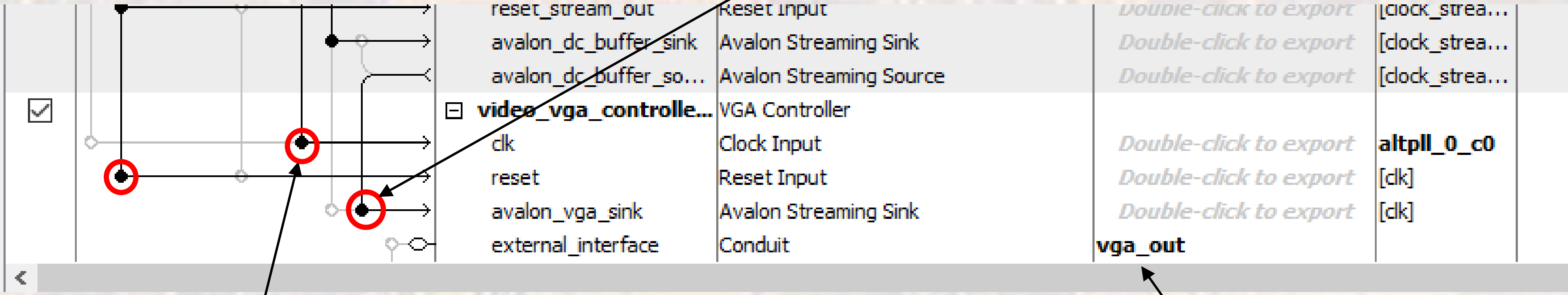
NIOS II Character Display - HW

- Create Character System
  - VGA Controller
    - University Program → Audio and Video → Video → VGA Controller

      DE10-Lite

      VGA Connector

      VGA 640x480



	reset_stream_out	Reset Input	Double-click to export	[clock_strea...
	avalon_dc_buffer_sink	Avalon Streaming Sink	Double-click to export	[clock_strea...
	avalon_dc_buffer_so...	Avalon Streaming Source	Double-click to export	[clock_strea...
☑	⊟ **video_vga_controlle...**	VGA Controller		
	clk	Clock Input	Double-click to export	**altpll_0_c0**
	reset	Reset Input	Double-click to export	[clk]
	avalon_vga_sink	Avalon Streaming Sink	Double-click to export	[clk]
	external_interface	Conduit	**vga_out**	

Connect to dual_clock_buffer streaming source

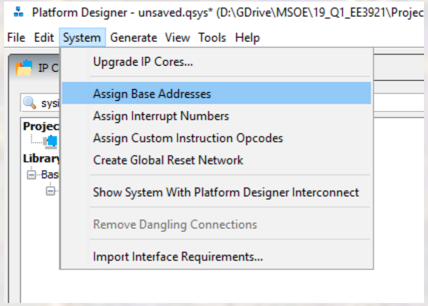
Connect to c0
VGA – 25MHz

Export and rename

NIOS II Character Display - HW

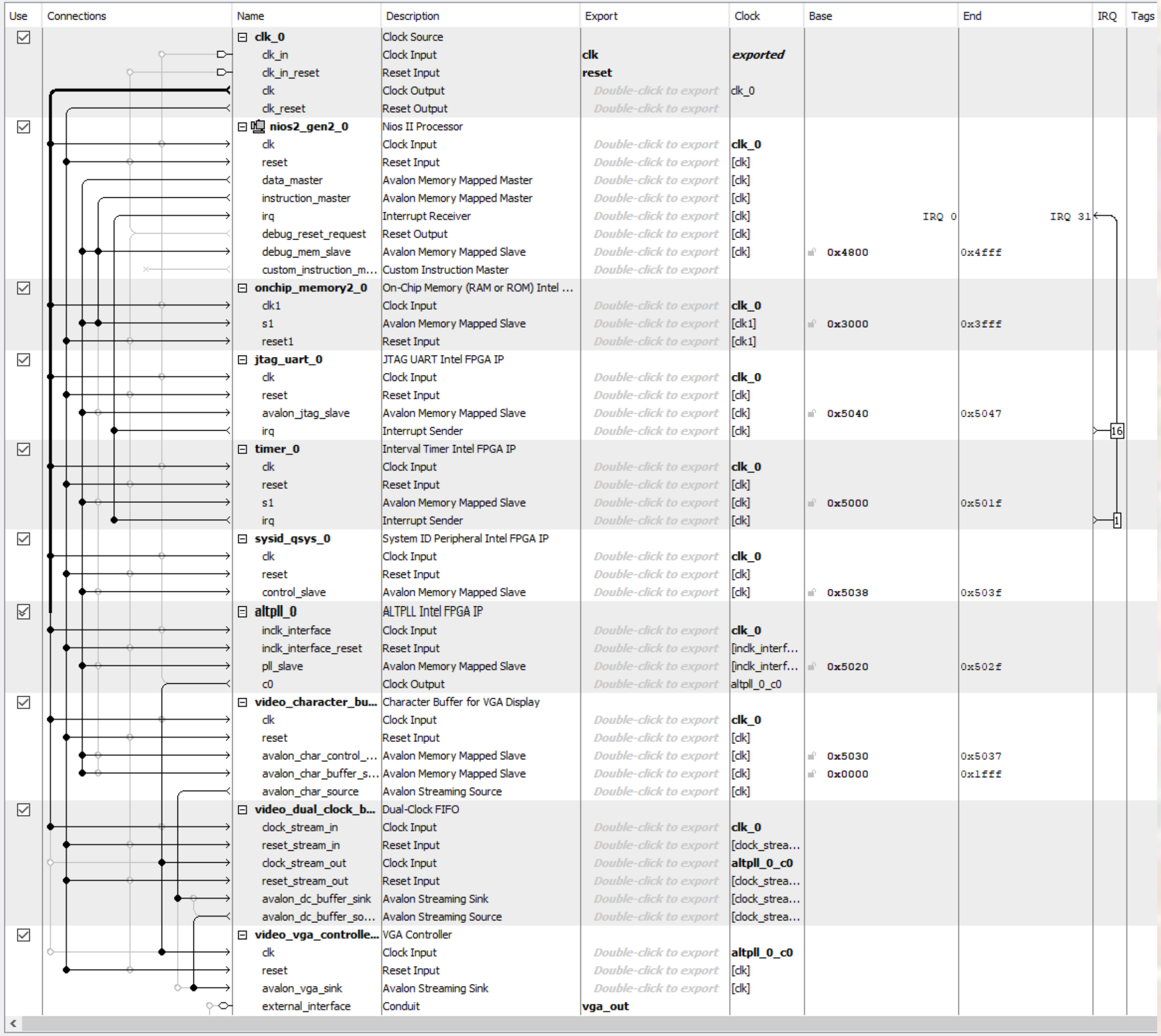
- Create Charac

Assign Base Addresses



Platform Designer - unsaved.qsys* (D:\GDrive\MSOE\19_Q1_EE3921\Projec

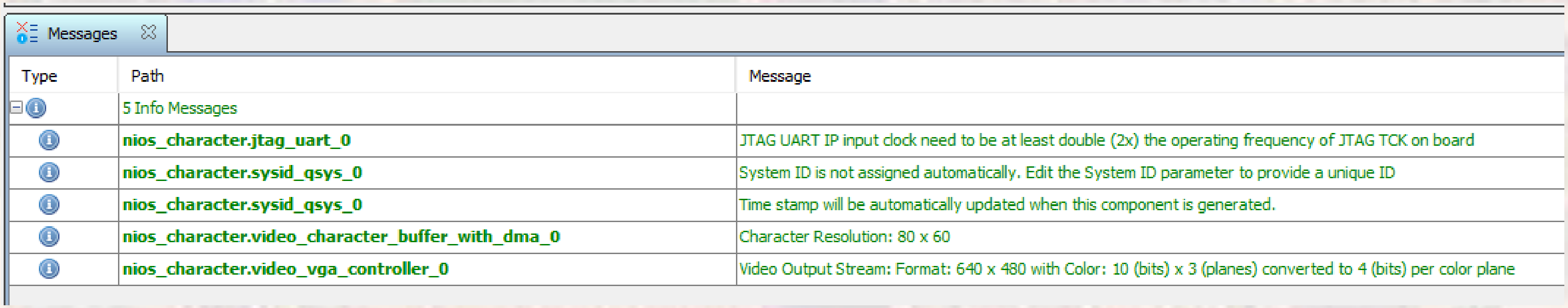
File Edit System Generate View Tools Help
IP C
sysi
Projec
Library
Bas
Upgrade IP Cores...
Assign Base Addresses
Assign Interrupt Numbers
Assign Custom Instruction Opcodes
Create Global Reset Network
Show System With Platform Designer Interconnect
Remove Dangling Connections
Import Interface Requirements...



Use	Connections	Name	Description	Export	Clock	Base	End	IRQ	Tags
☑		⊟ **clk_0**	Clock Source						
		clk_in	Clock Input	**clk**	***exported***				
		clk_in_reset	Reset Input	**reset**					
		clk	Clock Output	*Double-click to export*	clk_0				
		clk_reset	Reset Output	*Double-click to export*					
☑		⊟ **nios2_gen2_0**	Nios II Processor						
		clk	Clock Input	*Double-click to export*	**clk_0**				
		reset	Reset Input	*Double-click to export*	[clk]				
		data_master	Avalon Memory Mapped Master	*Double-click to export*	[clk]				
		instruction_master	Avalon Memory Mapped Master	*Double-click to export*	[clk]				
		irq	Interrupt Receiver	*Double-click to export*	[clk]	IRQ 0	IRQ 31		
		debug_reset_request	Reset Output	*Double-click to export*	[clk]				
		debug_mem_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x4800	0x4fff		
		custom_instruction_m...	Custom Instruction Master	*Double-click to export*					
☑		⊟ **onchip_memory2_0**	On-Chip Memory (RAM or ROM) Intel ...						
		clk1	Clock Input	*Double-click to export*	**clk_0**				
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk1]	0x3000	0x3fff		
		reset1	Reset Input	*Double-click to export*	[clk1]				
☑		⊟ **jtag_uart_0**	JTAG UART Intel FPGA IP						
		clk	Clock Input	*Double-click to export*	**clk_0**				
		reset	Reset Input	*Double-click to export*	[clk]				
		avalon_jtag_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x5040	0x5047		
		irq	Interrupt Sender	*Double-click to export*	[clk]			16	
☑		⊟ **timer_0**	Interval Timer Intel FPGA IP						
		clk	Clock Input	*Double-click to export*	**clk_0**				
		reset	Reset Input	*Double-click to export*	[clk]				
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x5000	0x501f		
		irq	Interrupt Sender	*Double-click to export*	[clk]			1	
☑		⊟ **sysid_qsys_0**	System ID Peripheral Intel FPGA IP						
		clk	Clock Input	*Double-click to export*	**clk_0**				
		reset	Reset Input	*Double-click to export*	[clk]				
		control_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x5038	0x503f		
☑		⊟ **altpll_0**	ALTPLL Intel FPGA IP						
		inclk_interface	Clock Input	*Double-click to export*	**clk_0**				
		inclk_interface_reset	Reset Input	*Double-click to export*	[inclk_interf...				
		pll_slave	Avalon Memory Mapped Slave	*Double-click to export*	[inclk_interf...	0x5020	0x502f		
		c0	Clock Output	*Double-click to export*	altpll_0_c0				
☑		⊟ **video_character_bu...**	Character Buffer for VGA Display						
		clk	Clock Input	*Double-click to export*	**clk_0**				
		reset	Reset Input	*Double-click to export*	[clk]				
		avalon_char_control_...	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x5030	0x5037		
		avalon_char_buffer_s...	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	0x0000	0x1fff		
		avalon_char_source	Avalon Streaming Source	*Double-click to export*	[clk]				
☑		⊟ **video_dual_clock_b...**	Dual-Clock FIFO						
		clock_stream_in	Clock Input	*Double-click to export*	**clk_0**				
		reset_stream_in	Reset Input	*Double-click to export*	[clock_strea...				
		clock_stream_out	Clock Input	*Double-click to export*	**altpll_0_c0**				
		reset_stream_out	Reset Input	*Double-click to export*	[clock_strea...				
		avalon_dc_buffer_sink	Avalon Streaming Sink	*Double-click to export*	[clock_strea...				
		avalon_dc_buffer_so...	Avalon Streaming Source	*Double-click to export*	[clock_strea...				
☑		⊟ **video_vga_controlle...**	VGA Controller						
		clk	Clock Input	*Double-click to export*	**altpll_0_c0**				
		reset	Reset Input	*Double-click to export*	[clk]				
		avalon_vga_sink	Avalon Streaming Sink	*Double-click to export*	[clk]				
		external_interface	Conduit	**vga_out**					

NIOS II Character Display - HW

- Create Character System
  - Check for errors

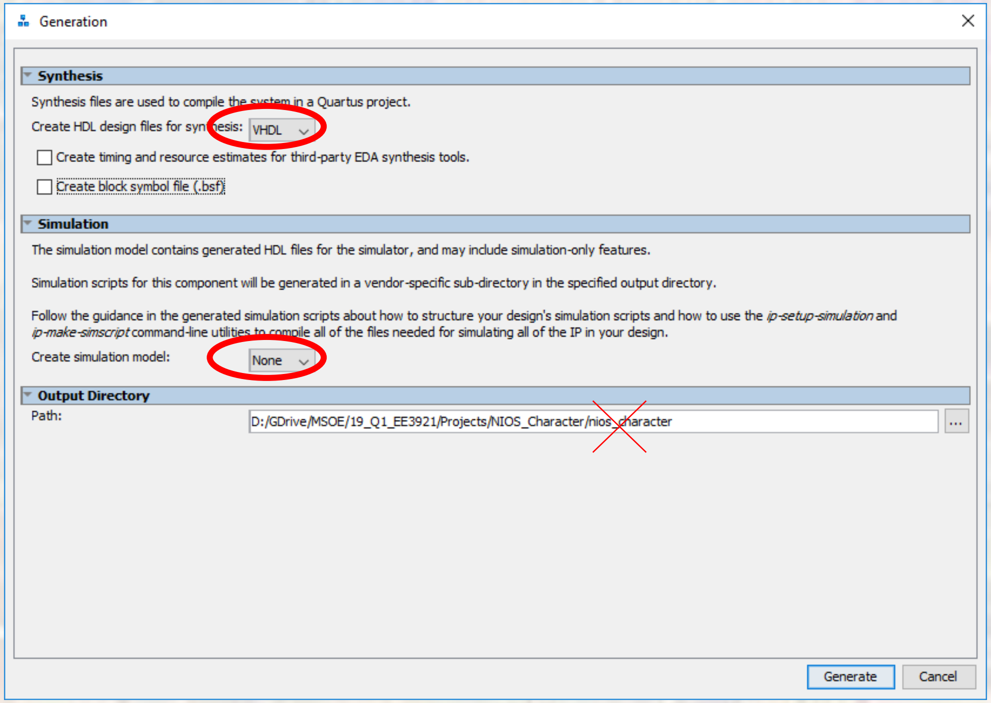


Messages

Type	Path	Message
⊟ ⓘ	5 Info Messages	
ⓘ	nios_character.jtag_uart_0	JTAG UART IP input clock need to be at least double (2x) the operating frequency of JTAG TCK on board
ⓘ	nios_character.sysid_qsys_0	System ID is not assigned automatically. Edit the System ID parameter to provide a unique ID
ⓘ	nios_character.sysid_qsys_0	Time stamp will be automatically updated when this component is generated.
ⓘ	nios_character.video_character_buffer_with_dma_0	Character Resolution: 80 x 60
ⓘ	nios_character.video_vga_controller_0	Video Output Stream: Format: 640 x 480 with Color: 10 (bits) x 3 (planes) converted to 4 (bits) per color plane

NIOS II Character Display - HW

- Create Character System
  - Save the Platform Designer system
  - Generate the Platform Designer system
    - The first time you generate you must delete the last directory in the path – don’t use the ‘…’



Generation

Synthesis

Synthesis files are used to compile the system in a Quartus project.

Create HDL design files for synthesis: VHDL

□ Create timing and resource estimates for third-party EDA synthesis tools.

□ Create block symbol file (.bsf)

Simulation

The simulation model contains generated HDL files for the simulator, and may include simulation-only features.

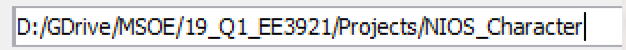
Simulation scripts for this component will be generated in a vendor-specific sub-directory in the specified output directory.

Follow the guidance in the generated simulation scripts about how to structure your design's simulation scripts and how to use the *ip-setup-simulation* and *ip-make-simscript* command-line utilities to compile all of the files needed for simulating all of the IP in your design.

Create simulation model: None

Output Directory

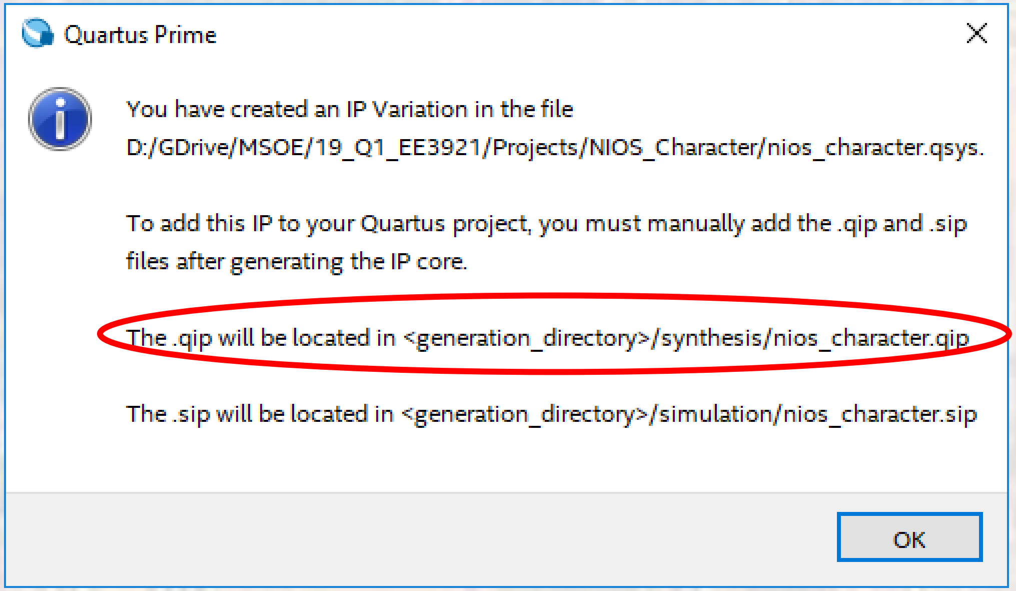
Path: D:/GDrive/MSOE/19_Q1_EE3921/Projects/NIOS_Character/~~nios_character~~



D:/GDrive/MSOE/19_Q1_EE3921/Projects/NIOS_Character

NIOS II Character Display - HW

- Create Character System
  - Add the .qip file to the project



You have created an IP Variation in the file D:/GDrive/MSOE/19_Q1_EE3921/Projects/NIOS_Character/nios_character.qsys.

To add this IP to your Quartus project, you must manually add the .qip and .sip files after generating the IP core.

The .qip will be located in <generation_directory>/synthesis/nios_character.qip

The .sip will be located in <generation_directory>/simulation/nios_character.sip

NIOS II Character Display - HW

- Create DE10 Design
  - Instantiate into a VHDL file
    - Open a new VHDL design
    - In Platform Designer: Generate → Show Instantiation Template
    - Copy and Paste into the new design where appropriate

```
component nios_char is
   port (
      clk_clk            : in  std_logic                  := 'X'; -- clk
      reset_reset_n :    in  std_logic                  := 'X'; -- reset_n
      vga_out_CLK      : out std_logic;                         -- CLK
      vga_out_HS       : out std_logic;                         -- HS
      vga_out_VS       : out std_logic;                         -- VS
      vga_out_BLANK  : out std_logic;                         -- BLANK
      vga_out_SYNC   : out std_logic;                         -- SYNC
      vga_out_R        : out std_logic_vector(3 downto 0);       -- R
      vga_out_G        : out std_logic_vector(3 downto 0);       -- G
      vga_out_B        : out std_logic_vector(3 downto 0)        -- B
   );
end component nios_char;
```

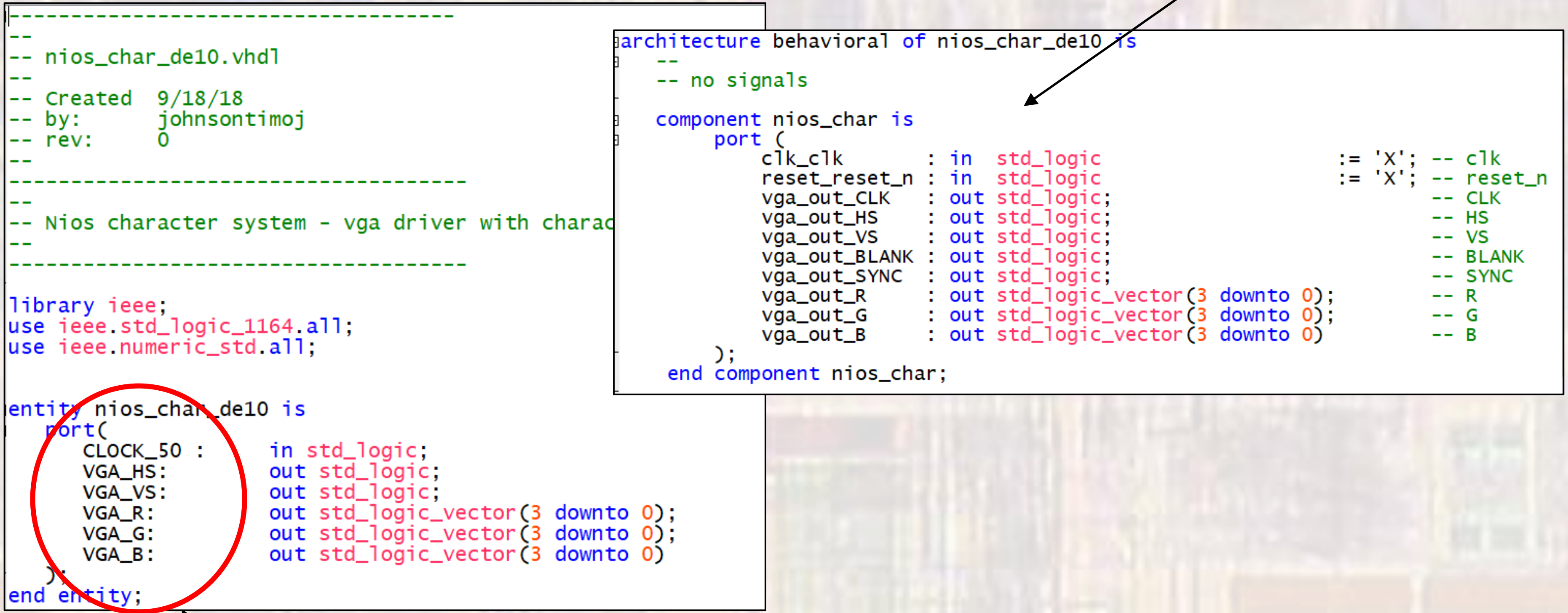
```
u0 : component nios_char
     port map (
        clk_clk              => CONNECTED_TO_clk_clk,        --     clk.clk
        reset_reset_n     => CONNECTED_TO_reset_reset_n, --   reset.reset_n
        vga_out_CLK       => CONNECTED_TO_vga_out_CLK,  -- vga_out.CLK
        vga_out_HS        => CONNECTED_TO_vga_out_HS,   --        .HS
        vga_out_VS        => CONNECTED_TO_vga_out_VS,   --        .VS
        vga_out_BLANK   => CONNECTED_TO_vga_out_BLANK, --        .BLANK
        vga_out_SYNC    => CONNECTED_TO_vga_out_SYNC,  --        .SYNC
        vga_out_R          => CONNECTED_TO_vga_out_R,     --        .R
        vga_out_G          => CONNECTED_TO_vga_out_G,     --        .G
        vga_out_B          => CONNECTED_TO_vga_out_B      --        .B
     );
```

TEMPLATES

NIOS II Character Display - HW

- Create DE10 Design
  - Instantiate into a VHDL file

Instantiation template component



```
------------------------------------
--
-- nios_char_de10.vhdl
--
-- Created  9/18/18
-- by:      johnsontimoj
-- rev:     0
--
-------------------------------------
--
-- Nios character system - vga driver with charac
--
-------------------------------------

library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;


entity nios_char_de10 is
    port(
        CLOCK_50 :     in std_logic;
        VGA_HS:        out std_logic;
        VGA_VS:        out std_logic;
        VGA_R:         out std_logic_vector(3 downto 0);
        VGA_G:         out std_logic_vector(3 downto 0);
        VGA_B:         out std_logic_vector(3 downto 0)
    );
end entity;
```

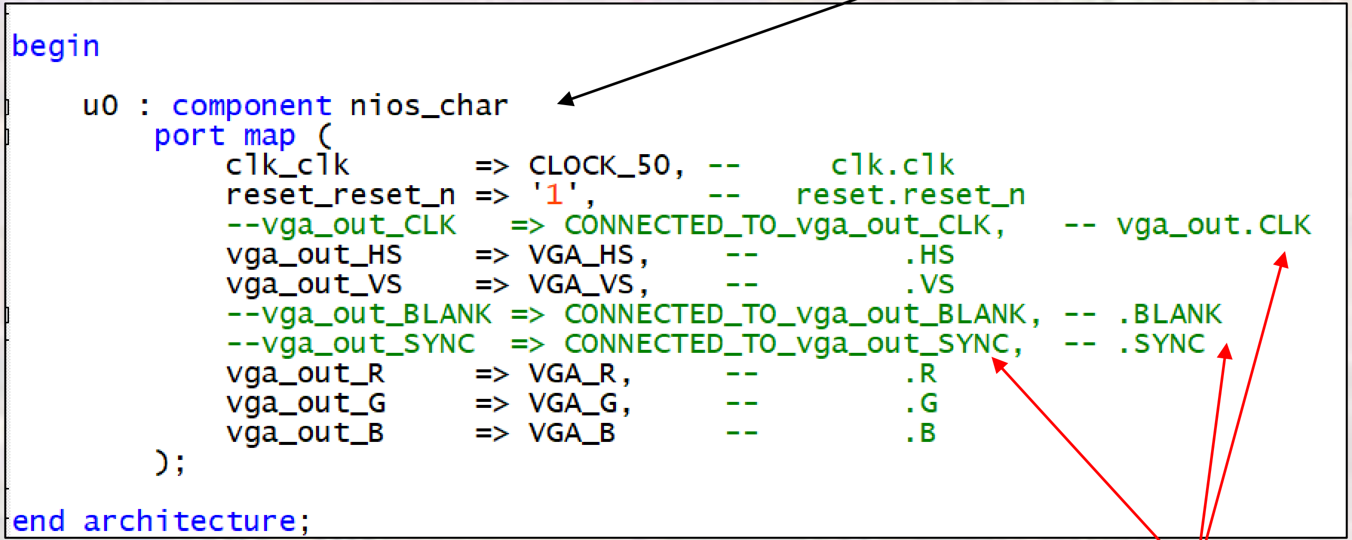
```
architecture behavioral of nios_char_de10 is
    --
    -- no signals

    component nios_char is
        port (
            clk_clk         : in  std_logic                    := 'X'; -- clk
            reset_reset_n   : in  std_logic                    := 'X'; -- reset_n
            vga_out_CLK     : out std_logic;                          -- CLK
            vga_out_HS      : out std_logic;                          -- HS
            vga_out_VS      : out std_logic;                          -- VS
            vga_out_BLANK   : out std_logic;                          -- BLANK
            vga_out_SYNC    : out std_logic;                          -- SYNC
            vga_out_R       : out std_logic_vector(3 downto 0);       -- R
            vga_out_G       : out std_logic_vector(3 downto 0);       -- G
            vga_out_B       : out std_logic_vector(3 downto 0)        -- B
        );
     end component nios_char;
```

DE10 pin aliases from .qsf file

NIOS II Character Display - HW

- Create DE10 Design
  - Instantiate into a VHDL file

Instantiation template instance mapped to DE10 qsf pin aliases



```
begin

    u0 : component nios_char
        port map (
            clk_clk       => CLOCK_50, --      clk.clk
            reset_reset_n => '1',      --    reset.reset_n
            --vga_out_CLK   => CONNECTED_TO_vga_out_CLK,   -- vga_out.CLK
            vga_out_HS    => VGA_HS,    --         .HS
            vga_out_VS    => VGA_VS,    --         .VS
            --vga_out_BLANK => CONNECTED_TO_vga_out_BLANK, -- .BLANK
            --vga_out_SYNC  => CONNECTED_TO_vga_out_SYNC,  -- .SYNC
            vga_out_R     => VGA_R,     --         .R
            vga_out_G     => VGA_G,     --         .G
            vga_out_B     => VGA_B      --         .B
        );

end architecture;
```

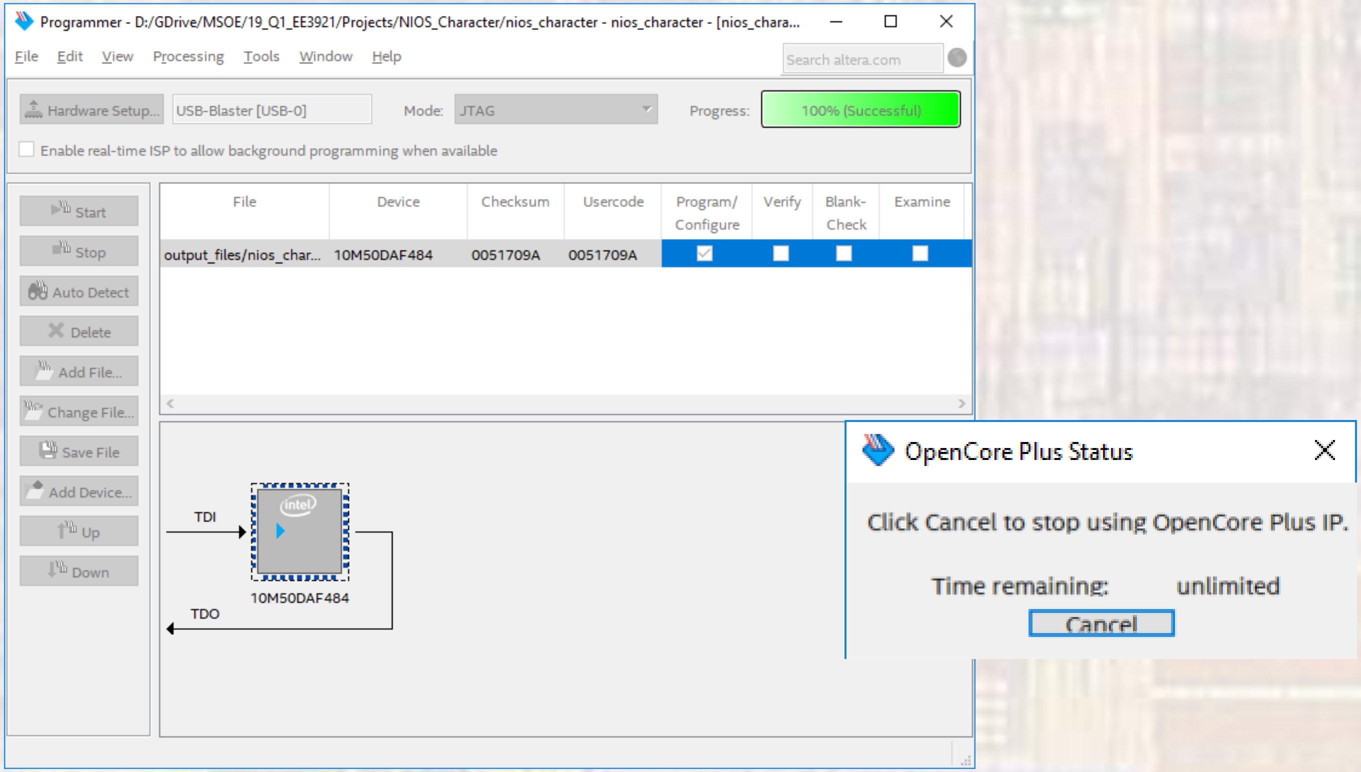
Note: these 3 signals are not used
- comment out or remove

NIOS II Character Display - HW

- Create DE10 Design
  - Prepare to synthesize
    - If you did not do these when you created the project be sure to do them now
      - assignments → device → device and Pin options
        - Single Uncompressed with memory initialization
      - Import the pin aliases (qsf file)
      - Setup the SDF file
  - Be sure to set your top level entity
  - Start Compilation

NIOS II Character Display - HW

- Create DE10 Design
  - Complete the HW setup
    - Download the HW project onto the board
    - DO NOT CLOSE either of these windows



Programmer - D:/GDrive/MSOE/19_Q1_EE3921/Projects/NIOS_Character/nios_character - nios_character - [nios_chara...

Hardware Setup... USB-Blaster [USB-0] Mode: JTAG Progress: 100% (Successful)

□ Enable real-time ISP to allow background programming when available

File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank-Check	Examine
output_files/nios_char...	10M50DAF484	0051709A	0051709A	☑	□	□	□

OpenCore Plus Status

Click Cancel to stop using OpenCore Plus IP.

Time remaining: unlimited