NTSC Video to VGA Tutorial

By Patrick Kuczera
Partners: Shahzeb Asif & Andrew Zhong
Group 7
April. 5th, 2016

Introduction

This tutorial will demonstrate how to interface NTSC standard video from the ADV7181B TV decoder to the ADV7123 VGA port on the Altera DE2 development board.

Hardware Requirements

- Altera DE2-115 Development Board
- NTSC standard video camera with RCA video adapter
- Monitor with 4:3 aspect ratio and VGA adapter

Software Requirements

- Quartus II 32-Bit Version 12.1 SP 1

Assumptions

You are:

- Able to setup a quartus project as detailed within the introductory labs.
- Familiar with Qsys and are comfortable navigating its interface.
- Capable of wiring the top level vhdl/verilog file with qsys system components.

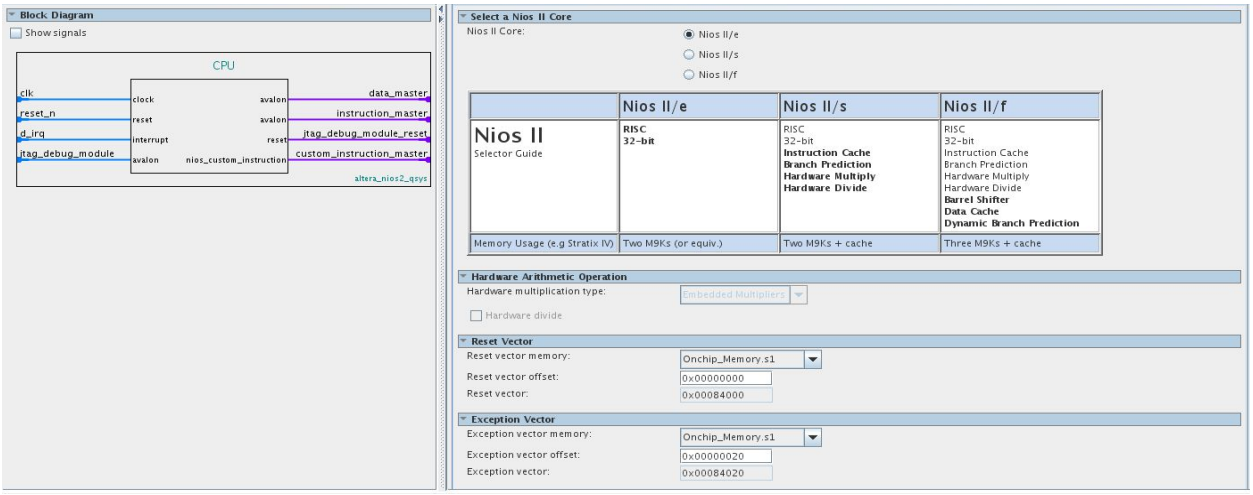
Provided Material

A demo quartus project is provided which was created following all the steps of this application note. This demo is obtained by downloading and unzipping *g7_video_app_notes.zip.* It includes the required Qsys system, top level VHDL, and DE2.qsf pin assignment file. This project is ready to be flashed to the Altera DE2-115 development board. Prior flashing the board, you must connect your camera and monitor via the RCA video and VGA ports, respectively. After flashing the board, powering on both camera and monitor, you should expect to see live stream video from the camera to monitor.

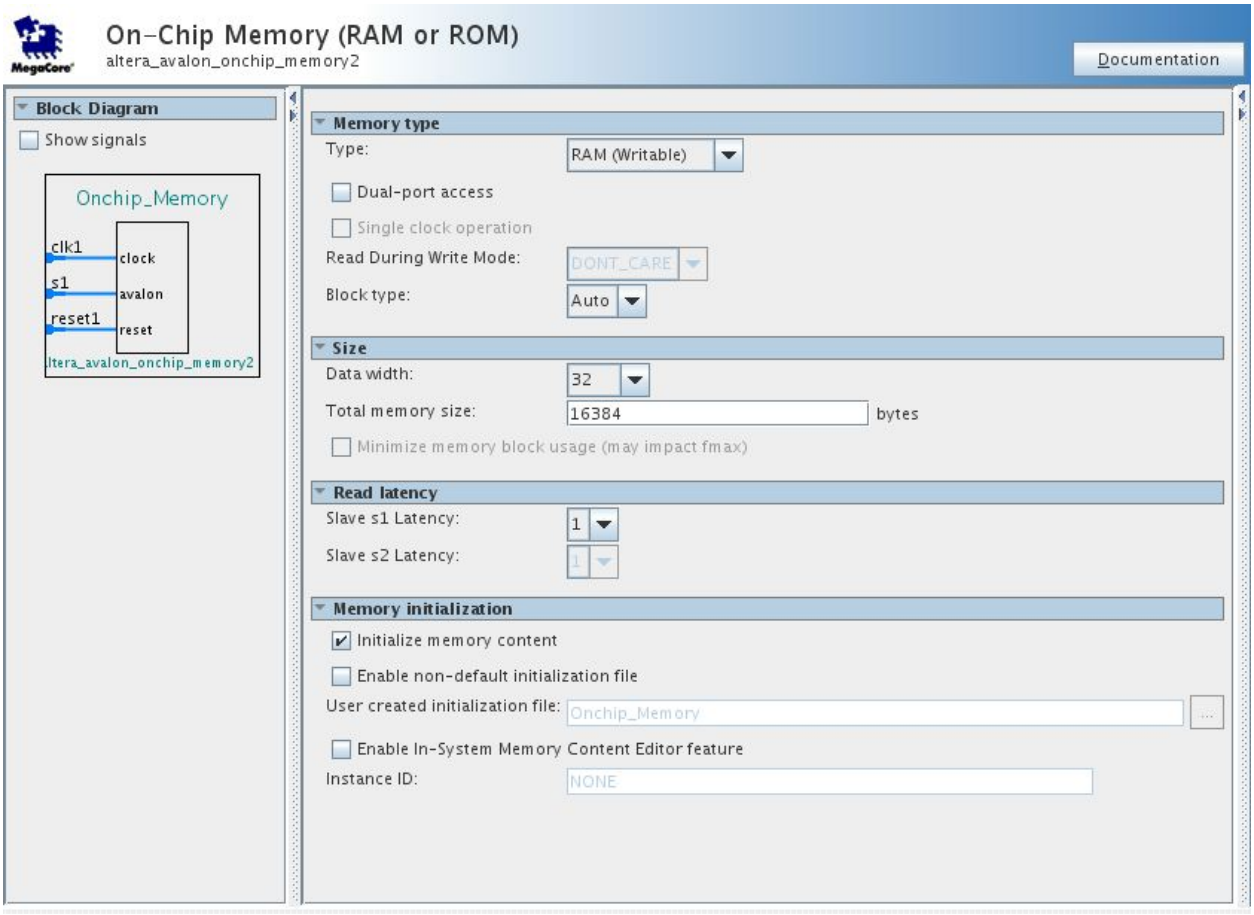
Procedure

The following is a list of components you must add to your Qsys system. Note that the component connections will be outlined at the end, as shown in Step 18.

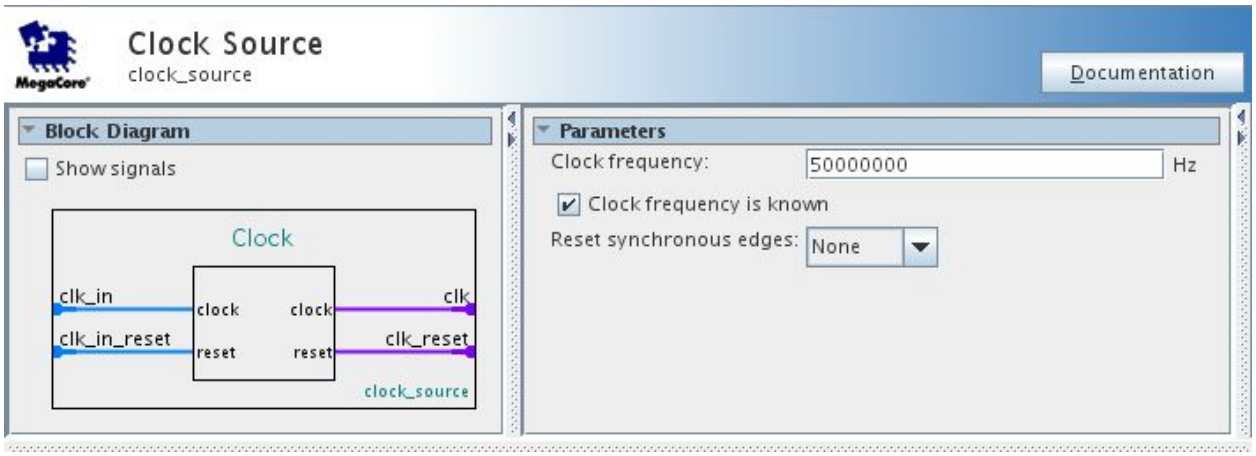
1. Nios II Processor



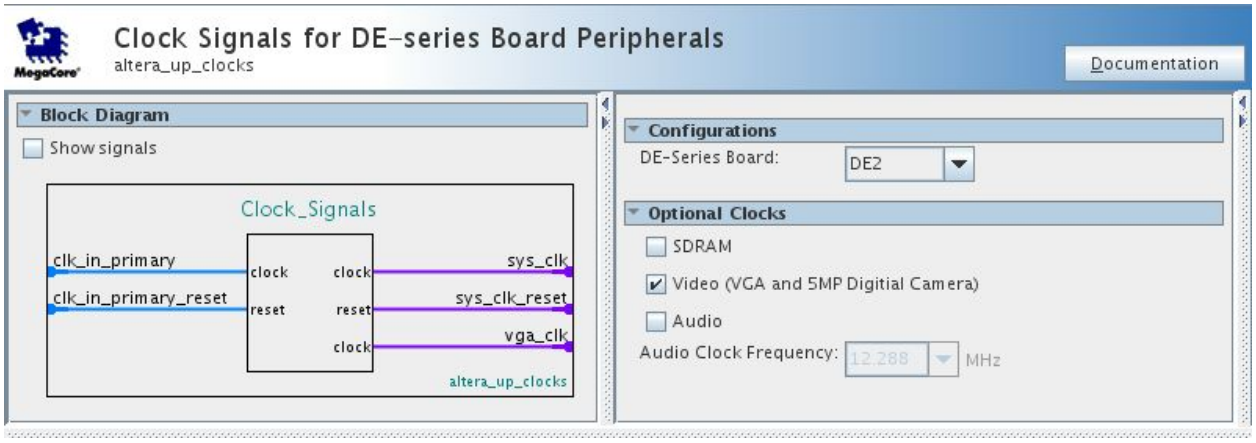
2. On-Chip Memory (RAM or ROM)



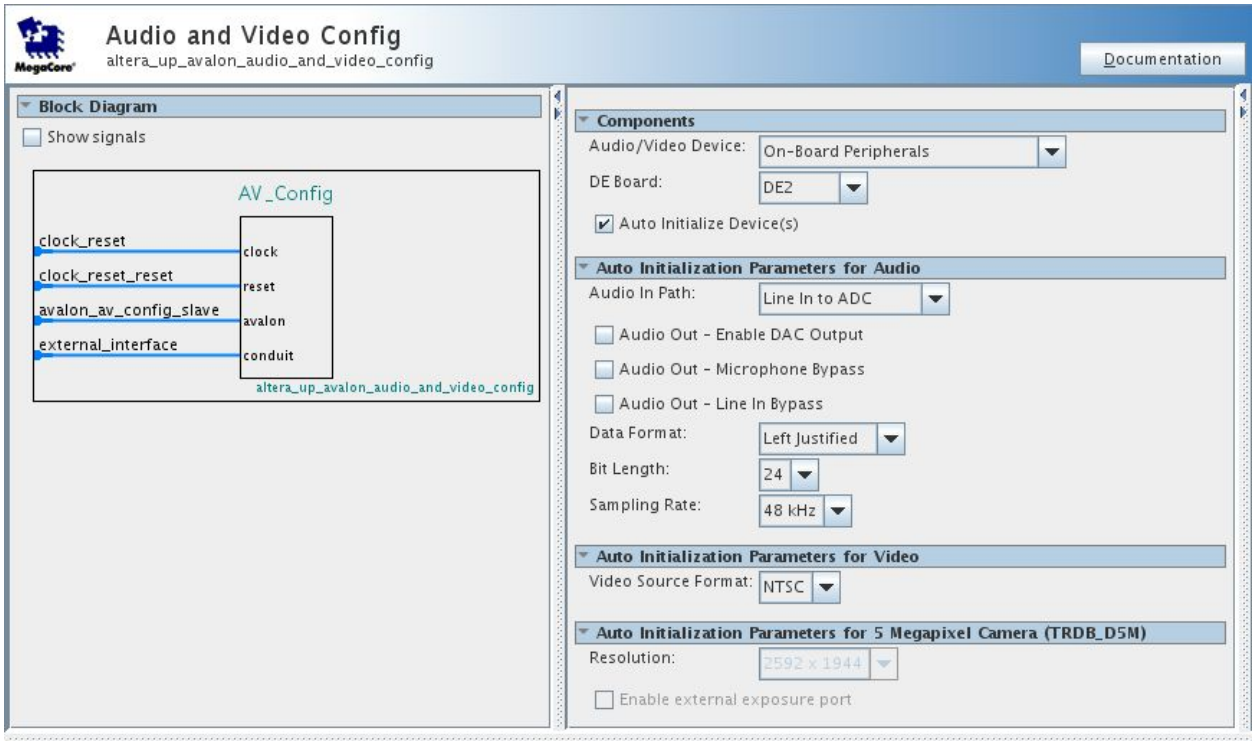
3. Clock Source



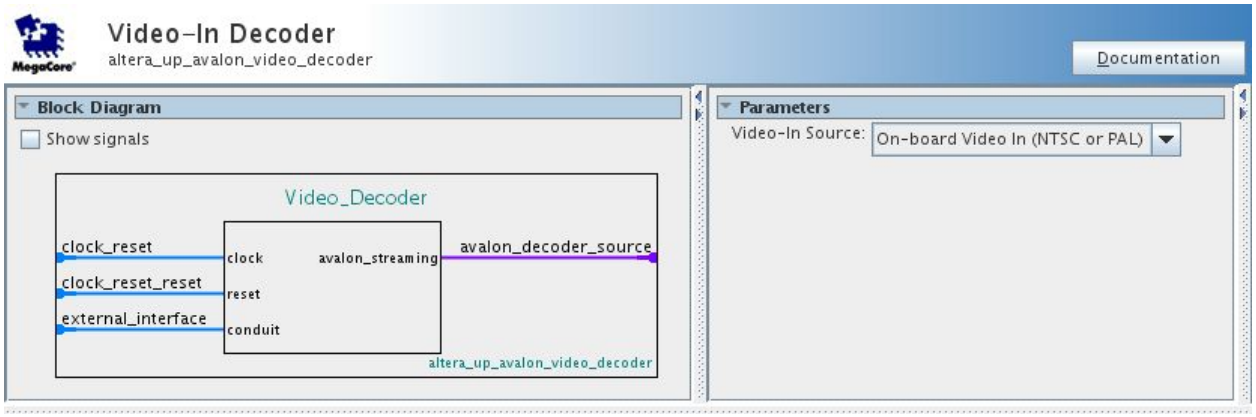
4. Clock Signals for DE-series Board Peripherals



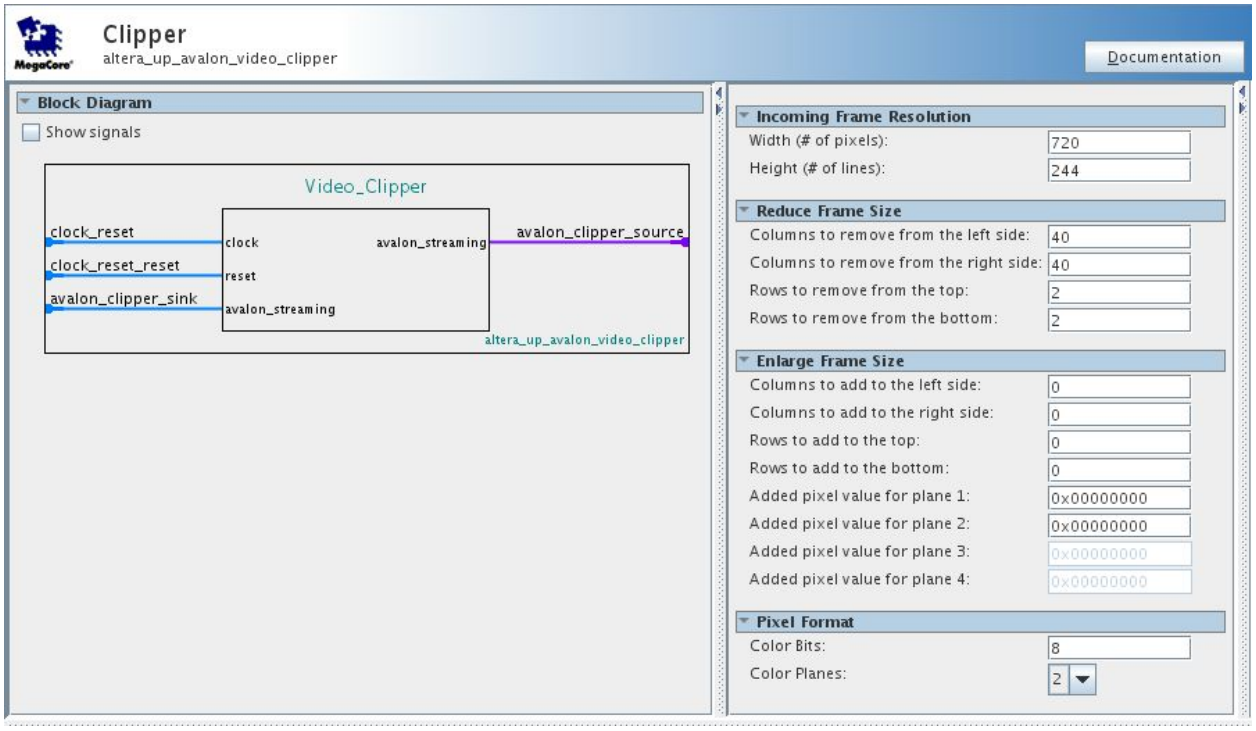
5. Audio and Video Config



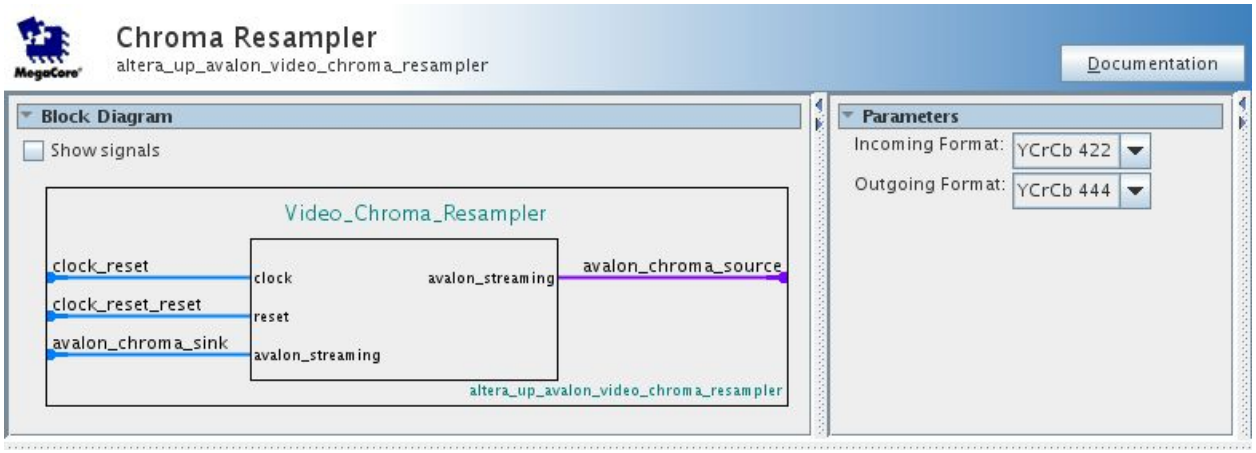
6. Video-In Decoder



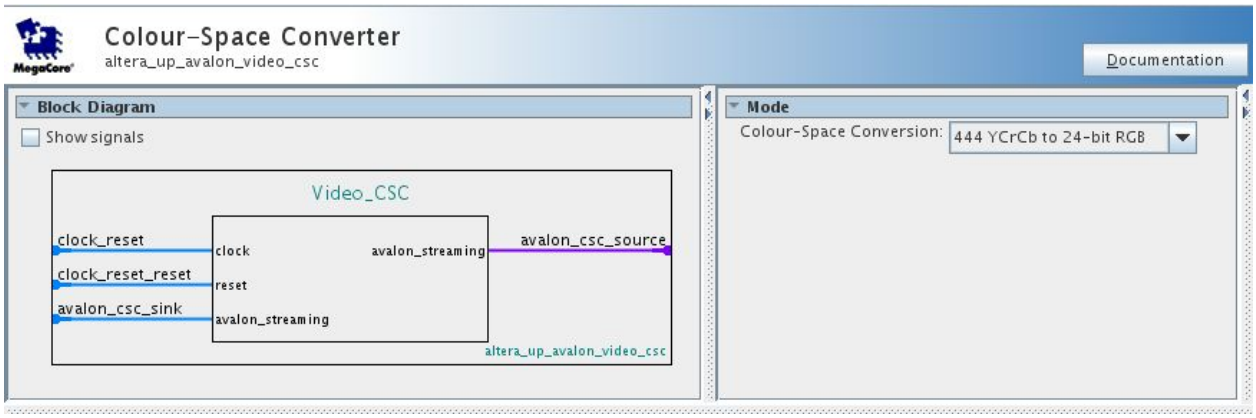
7. Clipper



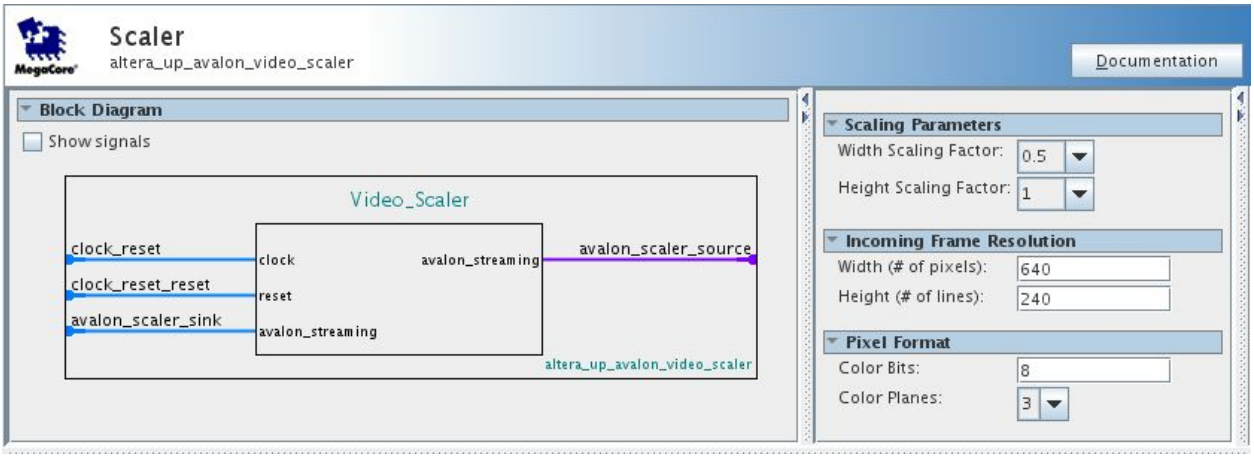
8. Chroma Resampler



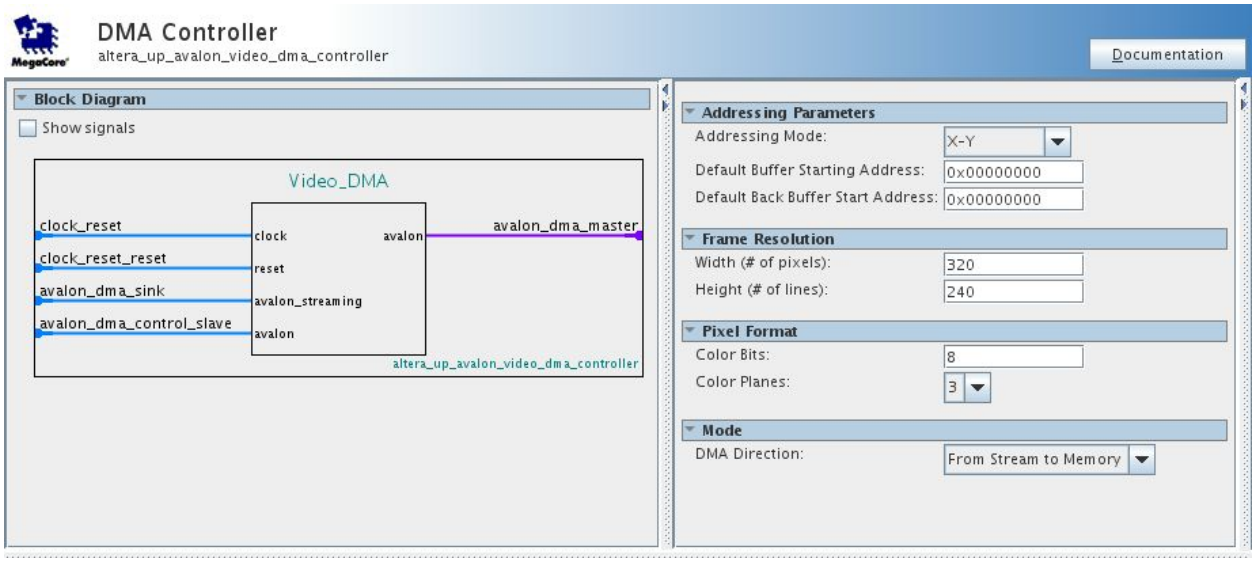
9. Colour-Space Converter



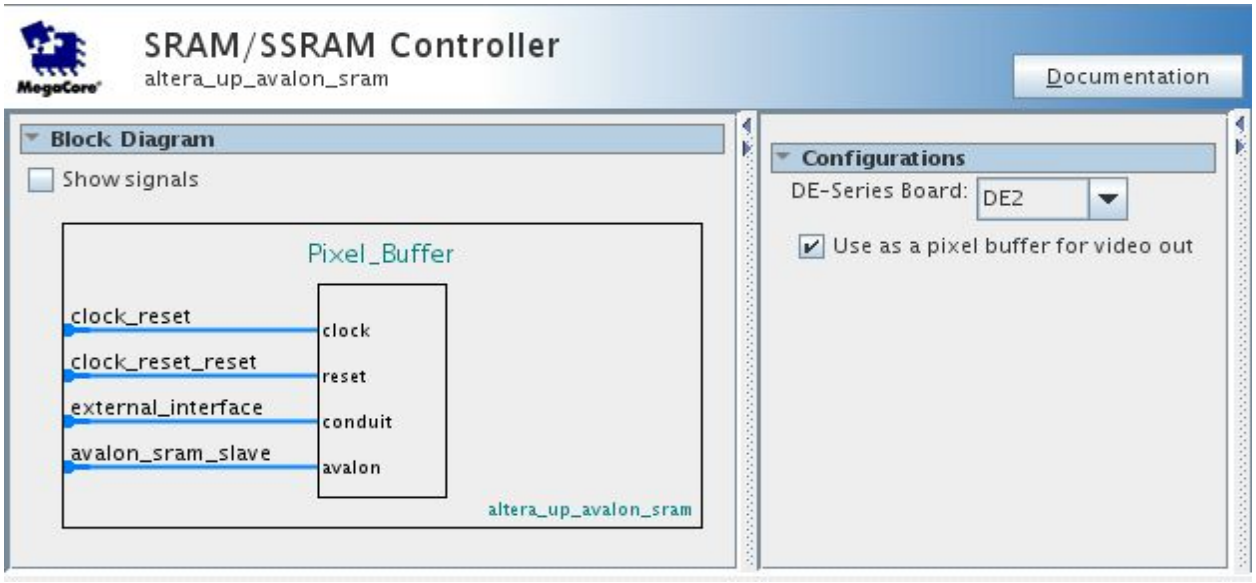
10. Scaler



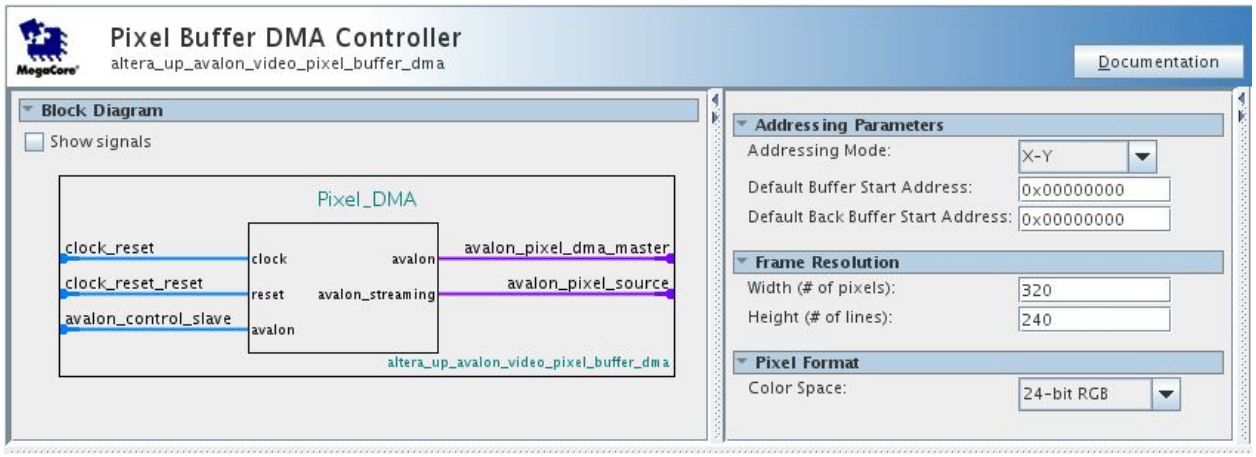
11. DMA Controller



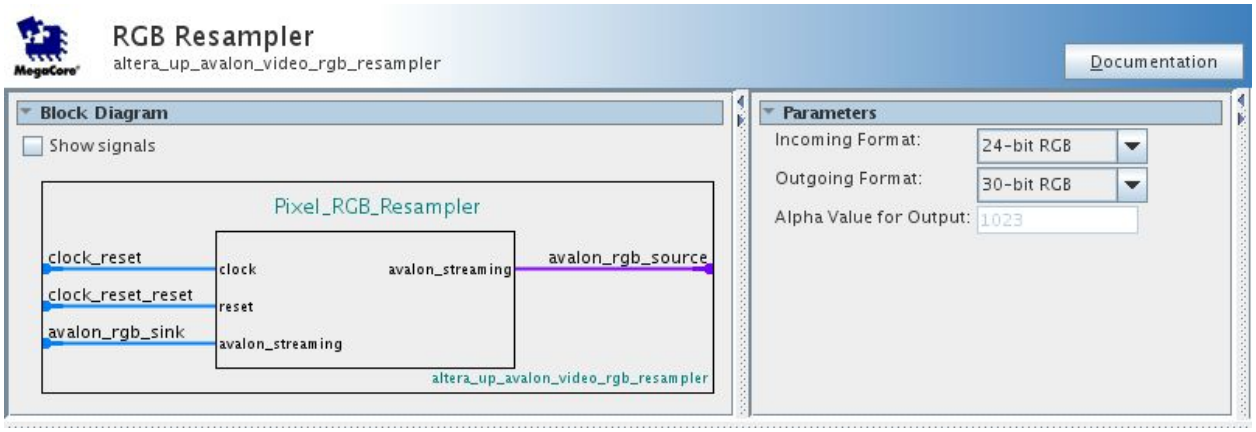
12. SRAM/SSRAM Controller



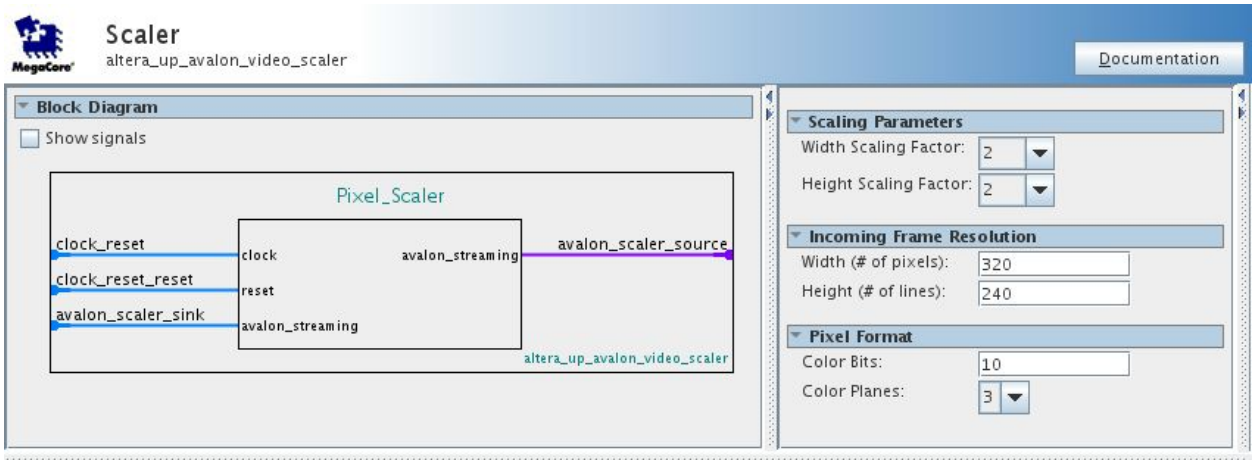
13. Pixel Buffer DMA Controller



14. RGB Resampler

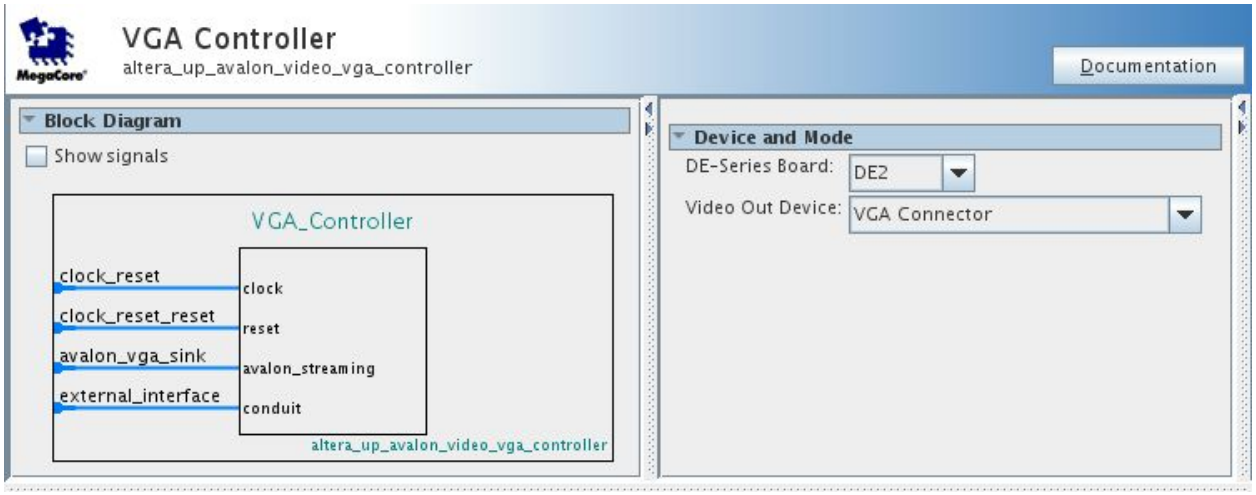


15. Scaler

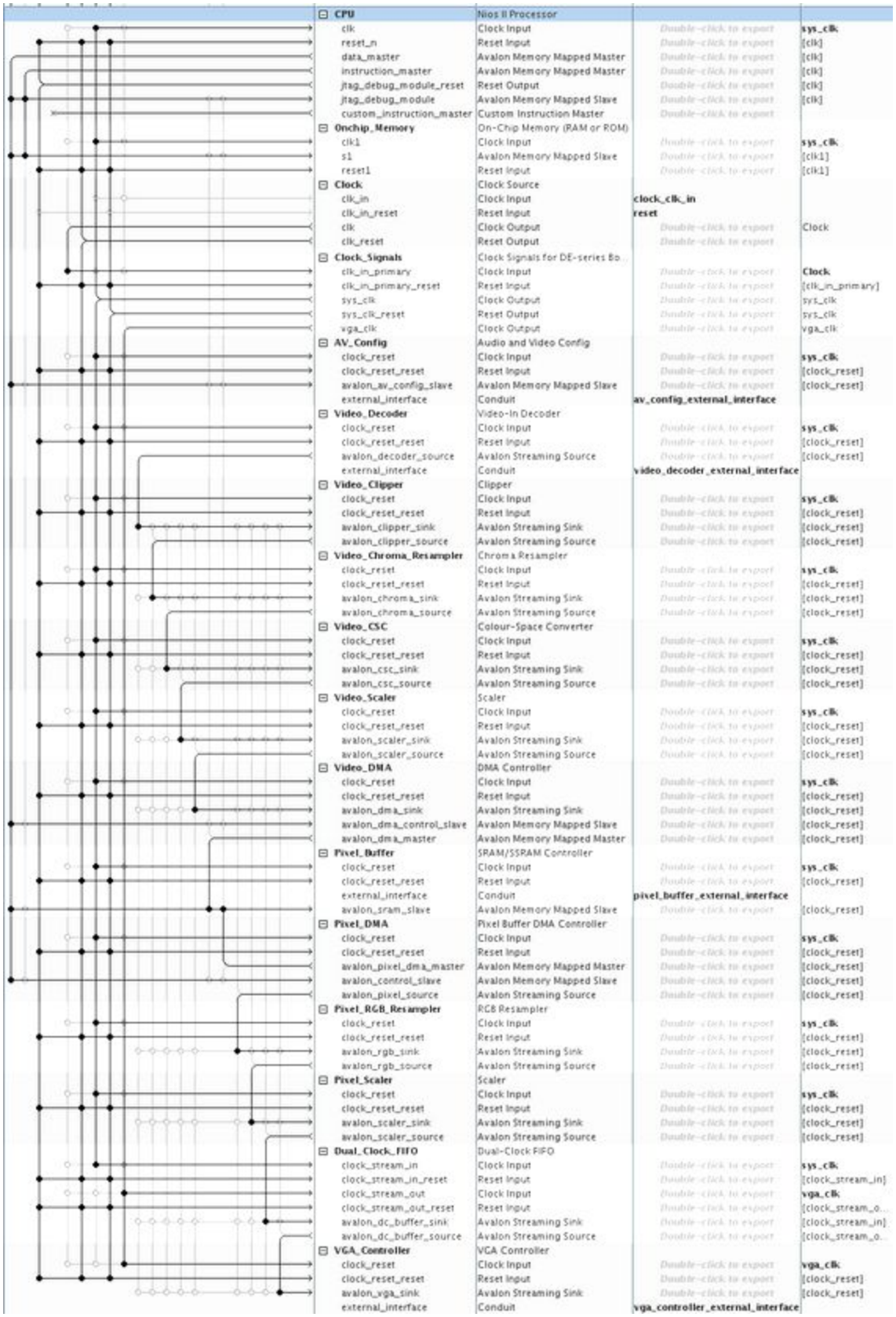


16. Dual-Clock FIFO

17. VGA Controller



18. Component Connections



CPU	Nios II Processor		
clk	Clock Input	Double-click to export	sys_clk
reset_n	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]
jtag_debug_module_reset	Reset Output	Double-click to export	[clk]
jtag_debug_module	Avalon Memory Mapped Slave	Double-click to export	[clk]
custom_instruction_master	Custom Instruction Master	Double-click to export	
Onchip_Memory	On-Chip Memory (RAM or ROM)		
clk1	Clock Input	Double-click to export	sys_clk
s1	Avalon Memory Mapped Slave	Double-click to export	[clk1]
reset1	Reset Input	Double-click to export	[clk1]
Clock	Clock Source		
clk_in	Clock Input	clock_clk_in	
clk_in_reset	Reset Input	reset	
clk	Clock Output	Double-click to export	Clock
clk_reset	Reset Output	Double-click to export	
Clock_Signals	Clock Signals for DE-series Bo		
clk_in_primary	Clock Input	Double-click to export	Clock
clk_in_primary_reset	Reset Input	Double-click to export	[clk_in_primary]
sys_clk	Clock Output	Double-click to export	sys_clk
sys_clk_reset	Reset Output	Double-click to export	sys_clk
vga_clk	Clock Output	Double-click to export	vga_clk
AV_Config	Audio and Video Config		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_av_config_slave	Avalon Memory Mapped Slave	Double-click to export	[clock_reset]
external_interface	Conduit	av_config_external_interface	
Video_Decoder	Video-In Decoder		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_decoder_source	Avalon Streaming Source	Double-click to export	[clock_reset]
external_interface	Conduit	video_decoder_external_interface	
Video_Clipper	Clipper		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_clipper_sink	Avalon Streaming Sink	Double-click to export	[clock_reset]
avalon_clipper_source	Avalon Streaming Source	Double-click to export	[clock_reset]
Video_Chroma_Resampler	Chroma Resampler		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_chroma_sink	Avalon Streaming Sink	Double-click to export	[clock_reset]
avalon_chroma_source	Avalon Streaming Source	Double-click to export	[clock_reset]
Video_CSC	Colour-Space Converter		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_csc_sink	Avalon Streaming Sink	Double-click to export	[clock_reset]
avalon_csc_source	Avalon Streaming Source	Double-click to export	[clock_reset]
Video_Scaler	Scaler		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_scaler_sink	Avalon Streaming Sink	Double-click to export	[clock_reset]
avalon_scaler_source	Avalon Streaming Source	Double-click to export	[clock_reset]
Video_DMA	DMA Controller		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_dma_sink	Avalon Streaming Sink	Double-click to export	[clock_reset]
avalon_dma_control_slave	Avalon Memory Mapped Slave	Double-click to export	[clock_reset]
avalon_dma_master	Avalon Memory Mapped Master	Double-click to export	[clock_reset]
Pixel_Buffer	SRAM/SSRAM Controller		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
external_interface	Conduit	pixel_buffer_external_interface	
avalon_sram_slave	Avalon Memory Mapped Slave	Double-click to export	[clock_reset]
Pixel_DMA	Pixel Buffer DMA Controller		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_pixel_dma_master	Avalon Memory Mapped Master	Double-click to export	[clock_reset]
avalon_control_slave	Avalon Memory Mapped Slave	Double-click to export	[clock_reset]
avalon_pixel_source	Avalon Streaming Source	Double-click to export	[clock_reset]
Pixel_RGB_Resampler	RGB Resampler		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_rgb_sink	Avalon Streaming Sink	Double-click to export	[clock_reset]
avalon_rgb_source	Avalon Streaming Source	Double-click to export	[clock_reset]
Pixel_Scaler	Scaler		
clock_reset	Clock Input	Double-click to export	sys_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_scaler_sink	Avalon Streaming Sink	Double-click to export	[clock_reset]
avalon_scaler_source	Avalon Streaming Source	Double-click to export	[clock_reset]
Dual_Clock_FIFO	Dual-Clock FIFO		
clock_stream_in	Clock Input	Double-click to export	sys_clk
clock_stream_in_reset	Reset Input	Double-click to export	[clock_stream_in]
clock_stream_out	Clock Input	Double-click to export	vga_clk
clock_stream_out_reset	Reset Input	Double-click to export	[clock_stream_o...
avalon_dc_buffer_sink	Avalon Streaming Sink	Double-click to export	[clock_stream_in]
avalon_dc_buffer_source	Avalon Streaming Source	Double-click to export	[clock_stream_o...
VGA_Controller	VGA Controller		
clock_reset	Clock Input	Double-click to export	vga_clk
clock_reset_reset	Reset Input	Double-click to export	[clock_reset]
avalon_vga_sink	Avalon Streaming Sink	Double-click to export	[clock_reset]
external_interface	Conduit	vga_controller_external_interface	

After adding all components as listed above, refresh memory addresses by System->Assign Base Addresses. Generate the system and copy the VHDL example from Qsys to your top level file.
Your top level VHDL should be wired similarly(names may differ):

```
u0 : component niosII_system
   port map (
      reset_reset_n                                => KEY(0),
      clock_clk_in_clk                             => CLOCK_50,

      pixel_buffer_external_interface_DQ           => SRAM_DQ,
      pixel_buffer_external_interface_ADDR         => SRAM_ADDR,
      pixel_buffer_external_interface_LB_N         => SRAM_LB_N,
      pixel_buffer_external_interface_UB_N         => SRAM_UB_N,
      pixel_buffer_external_interface_CE_N         => SRAM_CE_N,
      pixel_buffer_external_interface_OE_N         => SRAM_OE_N,
      pixel_buffer_external_interface_WE_N         => SRAM_WE_N,

      vga_controller_external_interface_CLK        => VGA_CLK,
      vga_controller_external_interface_HS         => VGA_HS,
      vga_controller_external_interface_VS         => VGA_VS,
      vga_controller_external_interface_BLANK      => VGA_BLANK,
      vga_controller_external_interface_SYNC       => VGA_SYNC,
      vga_controller_external_interface_R          => VGA_R,
      vga_controller_external_interface_G          => VGA_G,
      vga_controller_external_interface_B          => VGA_B,

      av_config_external_interface_SDAT            => I2C_SDAT,
      av_config_external_interface_SCLK            => I2C_SCLK,

      video_decoder_external_interface_TD_CLK27    => TD_CLK27,
      video_decoder_external_interface_TD_DATA     => TD_DATA,
      video_decoder_external_interface_TD_HS       => TD_HS,
      video_decoder_external_interface_TD_VS       => TD_VS,
      video_decoder_external_interface_TD_RESET    => TD_RESET
   );
```

Next, make sure to update your pin assignments by setting TD_CLK27 to PIN_C16 in the DE2.qsf file. A proper video output will not be achievable without this.

```
444 set_location_assignment PIN_V24 -to GPIO_1[32]
445 set_location_assignment PIN_V23 -to GPIO_1[33]
446 set_location_assignment PIN_W25 -to GPIO_1[34]
447 set_location_assignment PIN_W23 -to GPIO_1[35]
448 set_location_assignment PIN_C16 -to TD_CLK27
449
450 set_global_assignment -name PARTITION_NETLIST_TYPE SOURCE -section_id Top
```

Finally, compile your project and flash it to the DE2 board. After connecting your monitor and camera via VGA and RCA adapters respectively, you should obtain streamed 640x480 30-bit RGB video.

References

[1]	Altera Corporation. (August 2012) “Video IP Cores” [Online] Last accessed: Feb 28, 2016 ftp://ftp.altera.com/up/pub/Altera_Material/12.1/University_Program_IP_Cores/Audio_Video/Video.pdf
[2]	Gerry Finlay. (2013, March 27) “NTSC to VGA using Altera University Program IP Cores” [Online] Last Accessed: Feb 19, 2015 https://www.ualberta.ca/~delliott/local/ece492/appnotes/2013w/G15_Video_Out/G15_SOPC_Video