No spaces allowed in paths or file names

- No simulation can be selected
- The Time-Limited warning box must be left open



- To debug your C code
 - Select "debug as" instead of "run as"
 - You have access to variables, registers, stepping, breakpoints, ...

- Can't find ... (you built the BSP without the DE10 programmed and the time-limited window open)
 - Select the Target Connection tab
 - Scroll to the far right
 - Select refresh connections

	Name:	lios II Hardware connegration							
type filter text	Project Larget Connection & Deburger Common & Source								
C++ Application C++ Remote Application	Project name:	nios_pixel_sw	~						
inch Group	Project ELF file name:	D:\GDrive\MSOE\20_Q1_EE39	21\Projects\VIOS_Pixel\software	:\nios_pixel_sw\ ∨					
nios_pixel_sw Nios II Hardware c	Enable browse for f	ile system ELF file							
is II Hardware v2 (beta) is II ModelSim	File system ELF file nar	ne:	-						
ıs II ModelSim v2 (beta)				Name: nios_	pixel_sw N	ios II Hardware	configuratio	n	
				Project	Target (Connection	🏇 Debugge	r 🔲 <u>C</u> ommon 🖏	Source
						1			
				e Dev	ce ID	Instance ID	Name	Architecture	Refresh Connections
									Resolve Names
									System ID Properties

Cannot generate BSP

emplates	_	Template description	
Blank Project Board Diagnostics Count Binany	^	Blank Project creates an empty project to which you can A add your code.	
Float2 Functionality Float2 GCC Float2 Performance Hello Freestanding Hello MicroC/OS-II Hello World Hello World Small		For details, click Finish to create to readme.txt file in the project dire The BSP for this template is based operating system. To use a BSP b operating system, click Next and BSP projects list.	u-name nios2_gen2_0
Memory Test Memory Test Small	~	For information about how this surveyer example regime to the	OK
	_	Tor monitorior about not any sortific cautific reacts to	

- You failed to correct the path name when generating the platform designer HDP file
 - Remove the qip file from the project
 - Re-generate HDL with the corrected path
 - Compile, program
 - Open Eclipse ...

- ELF errors
 - Executable and Linkable Format
 - Contains the executable portions of your code along with information as to how to link, order and debug the sections

MOST ELF errors are CLOCK errors

- 1. Look at your Quartus RTL
 - 1. Make sure you compiled the correct TOP LEVEL design
 - 2. Make sure your pin NAMES are correct (capitalization MATTERS in pin names)
- 2. Check the pin planner and look for any pins that have not been assigned (they will appear white)
- 3. Check your Platform Designer (qsys) design
 - 1. Make sure all blocks have a clock and reset
 - 2. Check PLL frequencies (and phase shifts if using external DRAM)

- NIOS Pixel specific issues
 - My processor starts (prints out "entered main" or something similar) but nothing is going to the VGA display
 - 1. The notes clearly say that alt_up_pixel_buffer_dma_draw() draws to the back buffer, which means it is not going directly to the VGA
 - 2. Since the pixel buffer is in the external SDRAM but your code is in the FPGA SRAM you probably have a connection issue with the SDRAM
 - 1. Make sure the PLL has the correct phase offset
 - 2. Make sure the PLL drives the SDRAM CLK input

-- continued --

- NIOS Pixel specific issues
 - My processor starts (prints out "entered main" or something similar but nothing is going to the VGA display)
 - 3. Edit the BSP and open the Linker Script tab.
 - 1. .txt and the exceptions should point to on-chip-memory
 - 2. All other sections should point to the SDRAM

If not – recheck the wiring on the Pixel DMA buffer and the front and back buffer memory addresses

- NIOS Pixel specific issues
 - My processor does nothing (no elf error)
 - Make sure you checked the "small C library" and "reduced device driver" boxes in the BSP editor
 - Watch for "processor OK" and "processor paused" in the Nios console window
 - Check the memory allocation during the build project

CDT Build Console [nios_pixel_sw]	
<pre>nios2-elf-g++ -T'/nios_pixel_sw_bsp//linker.x' -msys-crt0='/nios_pixel_sw_bsp//obj/HAL/su</pre>	(C) A
nios2-elf-insert nios_pixel_sw_elfthread_model halcpu_name hros2_gen2_0gsys truesi	imi
Info: (nios_pixel_sw.elf) 16 KBytes program size (code + initialized data).	
Info: 65534 KBytes free for stack + heap.	
Info: Creating nios_pixel_sw.objdamp	
<pre>nios2-elf-objdumpdisassemblesymsall-headersource nios_pixel_sw.elf >nios_pixel_sw.</pre>	. oł
[nios_pixel_sw build complete]	
15:07:18 Build Finished (took 19s.296ms)	~
<	>

- NIOS Pixel specific issues
 - My display has a fixed colorful pattern
 - SDRAM is in it's default state you are not writing to it or it is not connected properly
 - My display has a fixed colorful pattern except the top has some ransom flickering pixels
 - Your program is running out of SDRAM and overlapping your pixel buffer
 - This usually is the result of a connection error between the pixel buffer and the SDRAM controller