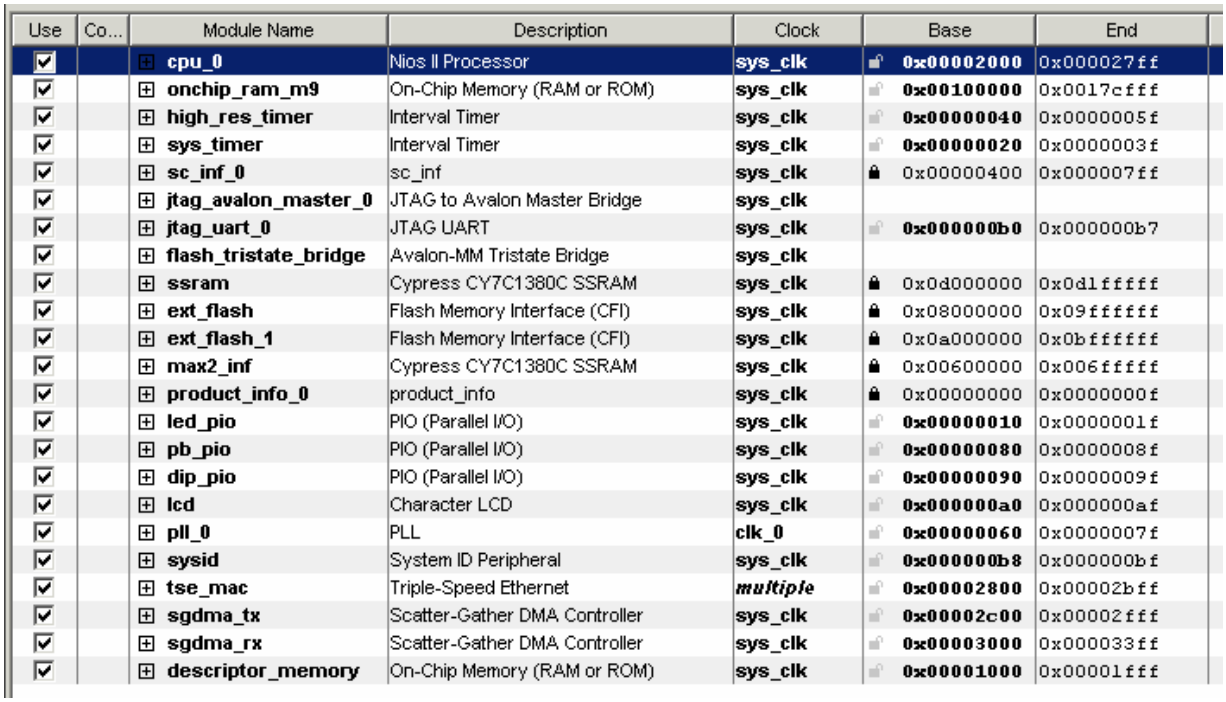
SSRAM_FLASH user guide



Use	Co...	Module Name	Description	Clock	Base	End
☑		cpu_0	Nios II Processor	sys_clk	0x00002000	0x000027ff
☑		onchip_ram_m9	On-Chip Memory (RAM or ROM)	sys_clk	0x00100000	0x0017cfff
☑		high_res_timer	Interval Timer	sys_clk	0x00000040	0x0000005f
☑		sys_timer	Interval Timer	sys_clk	0x00000020	0x0000003f
☑		sc_inf_0	sc_inf	sys_clk	0x00000400	0x000007ff
☑		jtag_avalon_master_0	JTAG to Avalon Master Bridge	sys_clk		
☑		jtag_uart_0	JTAG UART	sys_clk	0x000000b0	0x000000b7
☑		flash_tristate_bridge	Avalon-MM Tristate Bridge	sys_clk		
☑		ssram	Cypress CY7C1380C SSRAM	sys_clk	0x0d000000	0x0dlfffff
☑		ext_flash	Flash Memory Interface (CFI)	sys_clk	0x08000000	0x09ffffff
☑		ext_flash_1	Flash Memory Interface (CFI)	sys_clk	0x0a000000	0x0bffffff
☑		max2_inf	Cypress CY7C1380C SSRAM	sys_clk	0x00600000	0x006fffff
☑		product_info_0	product_info	sys_clk	0x00000000	0x0000000f
☑		led_pio	PIO (Parallel I/O)	sys_clk	0x00000010	0x0000001f
☑		pb_pio	PIO (Parallel I/O)	sys_clk	0x00000080	0x0000008f
☑		dip_pio	PIO (Parallel I/O)	sys_clk	0x00000090	0x0000009f
☑		lcd	Character LCD	sys_clk	0x000000a0	0x000000af
☑		pll_0	PLL	clk_0	0x00000060	0x0000007f
☑		sysid	System ID Peripheral	sys_clk	0x000000b8	0x000000bf
☑		tse_mac	Triple-Speed Ethernet	*multiple*	0x00002800	0x00002bff
☑		sgdma_tx	Scatter-Gather DMA Controller	sys_clk	0x00002c00	0x00002fff
☑		sgdma_rx	Scatter-Gather DMA Controller	sys_clk	0x00003000	0x000033ff
☑		descriptor_memory	On-Chip Memory (RAM or ROM)	sys_clk	0x00001000	0x00001fff

This project is just a connection to a lot of components.
There are 2 Avalon masters, CPU and JTAG-Avalon-Memory mapped Master.
Bellow them, there are

Component	instance name	Start address	End address
Onchip_ram_m9		0x00100000	0x0017cfff
System Console Interface	(sc_inf_inst)	0x00000400	0x000007ff
LED interface	(led_out)	0x10	0x1f
Dipswitch interface	(dipsw_in)	0x90	0x9f
Push Button interface	(pb_in)	0x80	0x8f
LCD interface	(lcd_0)	0xa0	0xaf
CFI Flash interface	(cfi_flash_0)	0x08000000	0x0bffffff
SSRAM interface	(ssram_0)	0x0D000000	0x0D1fffff
MAX-II register interface	(max2_inf)	0x00600000	0x006fffff

Using System Console to access FPGA.

The system console can be called from SOPC Builder or command line.

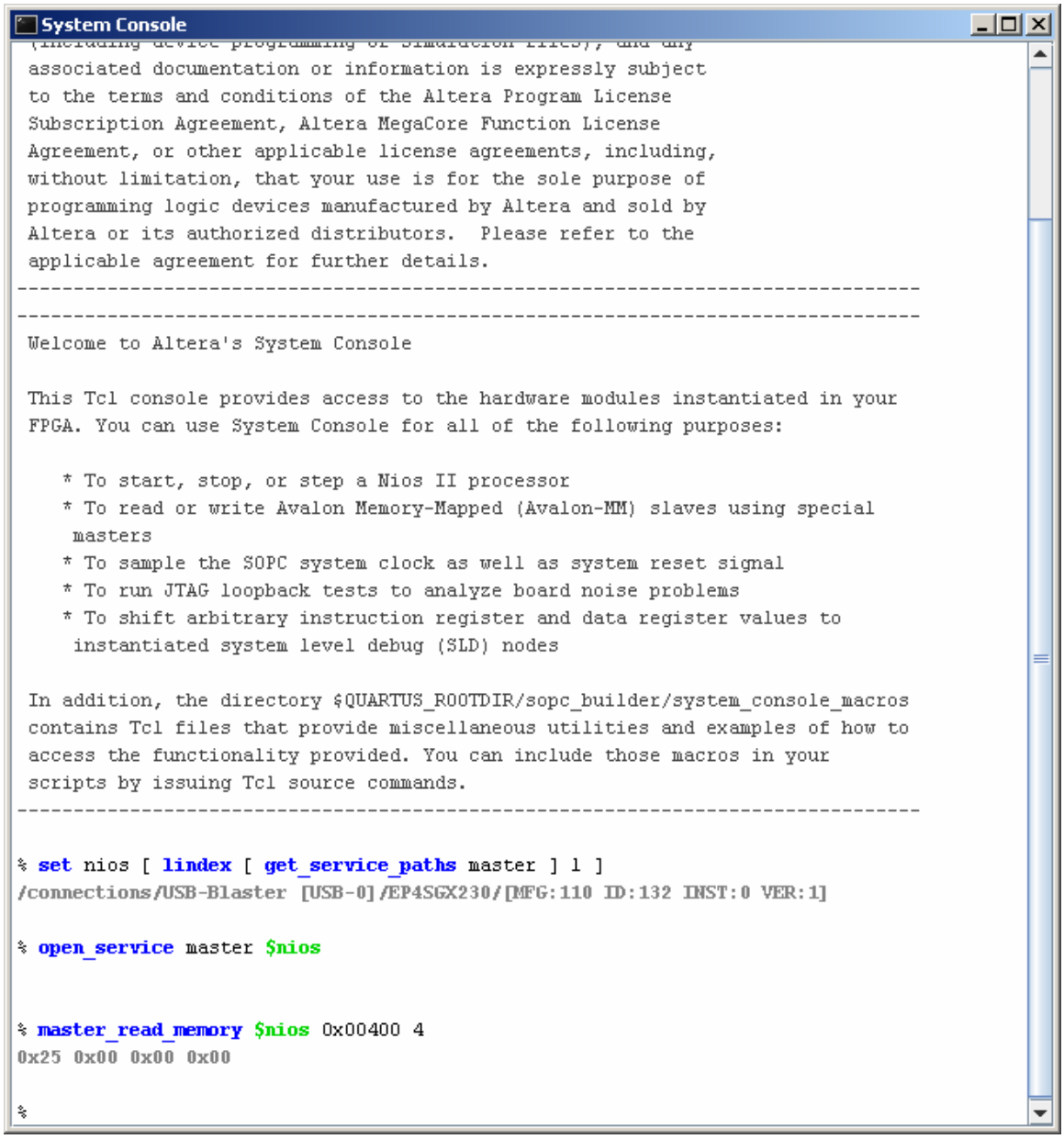
Here are the sample commands to use

1. set nios [lindex [get_service_paths master] 1]
2. open_service master $nios
3. master_read_memory $nios 0x00400 4
4. master_write_memory $nios 0x0040C [list 0x54]

3 and 4 is accessing the component.
Master_read_memory $nios [target_read_address] [byte_to_read]
So, the example is reading address 0x400 for 4 bytes.

master_write_memory $nios [target_write_address] [list write_data]
write_data is the hex data to write.
If you want to write 32 bits data, it will be [list 0x00 0x01 0x02 0x03]

For more information about the system console, please check online document for the system console.


System Console
(including device programming or simulation files), and any
associated documentation or information is expressly subject
to the terms and conditions of the Altera Program License
Subscription Agreement, Altera MegaCore Function License
Agreement, or other applicable license agreements, including,
without limitation, that your use is for the sole purpose of
programming logic devices manufactured by Altera and sold by
Altera or its authorized distributors. Please refer to the
applicable agreement for further details.
Welcome to Altera's System Console
This Tcl console provides access to the hardware modules instantiated in your
FPGA. You can use System Console for all of the following purposes:
* To start, stop, or step a Nios II processor
* To read or write Avalon Memory-Mapped (Avalon-MM) slaves using special
masters
* To sample the SOPC system clock as well as system reset signal
* To run JTAG loopback tests to analyze board noise problems
* To shift arbitrary instruction register and data register values to
instantiated system level debug (SLD) nodes
In addition, the directory $QUARTUS_ROOTDIR/sopc_builder/system_console_macros
contains Tcl files that provide miscellaneous utilities and examples of how to
access the functionality provided. You can include those macros in your
scripts by issuing Tcl source commands.
% set nios [lindex [get_service_paths master] 1]
/connections/USB-Blaster [USB-0]/EP4SGX230/[MFG:110 ID:132 INST:0 VER:1]
% open_service master $nios
% master_read_memory $nios 0x00400 4
0x25 0x00 0x00 0x00
%