

SUBJECT:- NIOS PROCESSOR BOOT USING QSPI FLASH

This document follows all the steps to boot Nios from QSPI FLASH as mentioned in [AN 730: Nios II Processor Booting Methods in MAX 10 FPGA Devices \(eeweb.com\)](#) (OPTION-5 : Nios II processor application copied from QSPI flash to RAM using boot copier)

Device used:- MAX10 DEVELOPMENT KIT(REV C)

Quartus Software Version:- Quartus Prime 21.1

1- QSYS Design.

Use	Connections	Name	Description	Export	Clock	Base
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> clk_0 clk_in clk_in_reset clk clk_reset 	<ul style="list-style-type: none"> Clock Source Clock Input Reset Input Clock Output Reset Output 	<ul style="list-style-type: none"> clk reset Double-click to export Double-click to export 	exported	clk_0
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> altpll_0 indk_interface indk_interface_reset pll_slave c0 c1 	<ul style="list-style-type: none"> ALTPLL Intel FPGA IP Clock Input Reset Input Avalon Memory Mapped Slave Clock Output Clock Output 	<ul style="list-style-type: none"> Double-click to export Double-click to export Double-click to export Double-click to export Double-click to export 	<ul style="list-style-type: none"> clk_0 [indk_interf... altpll_0_c0 altpll_0_c1 	0x0801_1050
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> nios2_gen2_0 clk reset data_master instruction_master irq debug_reset_request debug_mem_slave custom_instruction_master 	<ul style="list-style-type: none"> Nios II Processor Clock Input Reset Input Avalon Memory Mapped Master Avalon Memory Mapped Master Interrupt Receiver Reset Output Avalon Memory Mapped Slave Custom Instruction Master 	<ul style="list-style-type: none"> Double-click to export Double-click to export Double-click to export Double-click to export Double-click to export Double-click to export Double-click to export 	<ul style="list-style-type: none"> altpll_0_c0 [clk] [clk] [clk] [clk] [clk] 	0x0801_0800
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> generic_quad_spi_controller_0 clock_sink reset avl_csr avl_mem interrupt_sender flash_dataout flash_dclk_out flash_ncs 	<ul style="list-style-type: none"> Generic QUAD SPI Controller Intel FPG... Clock Input Reset Input Avalon Memory Mapped Slave Avalon Memory Mapped Slave Interrupt Sender Conduit Conduit Conduit 	<ul style="list-style-type: none"> Double-click to export Double-click to export Double-click to export Double-click to export Double-click to export Double-click to export generic_quad_spi_contr... generic_quad_spi_contr... generic_quad_spi_contr... 	<ul style="list-style-type: none"> altpll_0_c1 [clock_sink] [clock_sink] [clock_sink] [clock_sink] 	<ul style="list-style-type: none"> 0x0801_1020 0x0400_0000
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> led0 clk reset s1 external_connection 	<ul style="list-style-type: none"> PIO (Parallel I/O) Intel FPGA IP Clock Input Reset Input Avalon Memory Mapped Slave Conduit 	<ul style="list-style-type: none"> Double-click to export Double-click to export Double-click to export 	<ul style="list-style-type: none"> altpll_0_c0 [clk] [clk] 	0x0801_1040
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> onchip_memory2_0 clk1 s1 reset1 	<ul style="list-style-type: none"> On-Chip Memory (RAM or ROM) Intel ... Clock Input Avalon Memory Mapped Slave Reset Input 	<ul style="list-style-type: none"> Double-click to export Double-click to export Double-click to export 	<ul style="list-style-type: none"> altpll_0_c0 [clk1] [clk1] 	0x0800_8000

Clock source to NIOS Processor – 50MHz

Clock source to PLL - 50MHz

Clock source to Generic Quad SPI Controller through PLL(c1)- 25MHz

Clock source to On-Chip Memory(c1)- 50MHz

Nios Processor Parameters:-

Nios II Processor

altera_nios2_gen2

Main Vectors Caches and Memory Interfaces Arithmetic Instructions MMU and MPU Settings JTAG Debug Advanced Features

Reset Vector

Reset vector memory: generic_quad_spi_controller_0.avl_mem ▾

Reset vector offset: 0x00000000

Reset vector: 0x04000000

Exception Vector

Exception vector memory: onchip_memory2_0.s1 ▾

Exception vector offset: 0x00000020

Exception vector: 0x08008020

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

QUAD SPI Controller Parameters:-

Parameters

System: sys_A Path: generic_quad_spi_controller_0

Generic QUAD SPI Controller Intel FPGA IP

altera_generic_quad_spi_controller

Parameters

Configuration device type: N25Q512A83GSF40F ▾

Choose I/O mode: QUAD ▾

Number of Chip Selects used: 1 ▾

On-Chip RAM:-

Parameters

System: sys_A Path: onchip_memory2_0

On-Chip Memory (RAM or ROM) Intel FPGA IP

altera_avalon_onchip_memory2

Memory type

Type: RAM (Writable) ▾

Dual-port access

Single clock operation

Read During Write Mode: DONT_CARE ▾

Block type: AUTO ▾

Size

Enable different width for Dual-port access

Slave s1 Data width: 32 ▾

Total memory size: 32000 bytes

Minimize memory block usage (may impact fmax)

Read latency

Slave s1 Latency: 1 ▾

Slave s2 Latency: 1 ▾

ROM/RAM Memory Protection

Reset Request: Enabled ▾

ECC Parameter

Extend the data width to support ECC bits: Disabled ▾

Memory initialization

Initialize memory content

Enable non-default initialization file

Type the filename (e.g: my_ram.hex) or select the hex file using the file browser button.

User created initialization file: onchip_mem.hex

Enable Partial Reconfiguration Initialization Mode

Enable In-System Memory Content Editor feature

Instance ID: NONE

This memory is not initialized during device programming.

The Configuration Mode was set to **Single Uncompressed Image(3584 KBits UFM)**.

Device and Pin Options - max10_top

Category:

- General
- Configuration**
- Programming Files
- Unused Pins
- Dual-Purpose Pins
- Capacitive Loading
- Board Trace Model
- I/O Timing
- Voltage
- Pin Placement
- Error Detection CRC
- CvP Settings

Configuration

Specify the device configuration scheme and the configuration device.

Configuration scheme: Internal Configuration

Configuration mode: Single Uncompressed Image (3584Kbits UFM)

Configuration device

Use configuration device: Auto

Device Options ...

The HDL was generated of this QSYS Design and Full Compilation was done.

Advanced.hal.linker Settings in BSP EDITOR

SOPC Information file: D:\boot_max10\sys_A.sopcinfo
 CPU name: nios2_gen2_0
 Operating system: Altera HAL Version: default
 BSP target directory: D:\boot_max10\software\led_blink_bsp

hal.linker

- allow_code_at_reset
- enable_alt_load
- enable_alt_load_copy_rodata
- enable_alt_load_copy_rwdata
- enable_alt_load_copy_exceptions

BSP Editor - D:\boot_max10\Software\led_blink_bsp\settings.bsp

File Edit Tools Help

Main Software Packages Drivers Linker Script Enable File Generation Target BSP Directory

Linker Section Mappings

Linker Section Name	Linker Region Name	Memory Device Name
.bss	onchip_memory2_0	onchip_memory2_0
.entry	reset	generic_quad_spi_controller_0_av1_mem
.exceptions	onchip_memory2_0	onchip_memory2_0
.heap	onchip_memory2_0	onchip_memory2_0
.rodata	onchip_memory2_0	onchip_memory2_0
.rwdata	onchip_memory2_0	onchip_memory2_0
.stack	onchip_memory2_0	onchip_memory2_0
.text	onchip_memory2_0	onchip_memory2_0

Linker Memory Regions

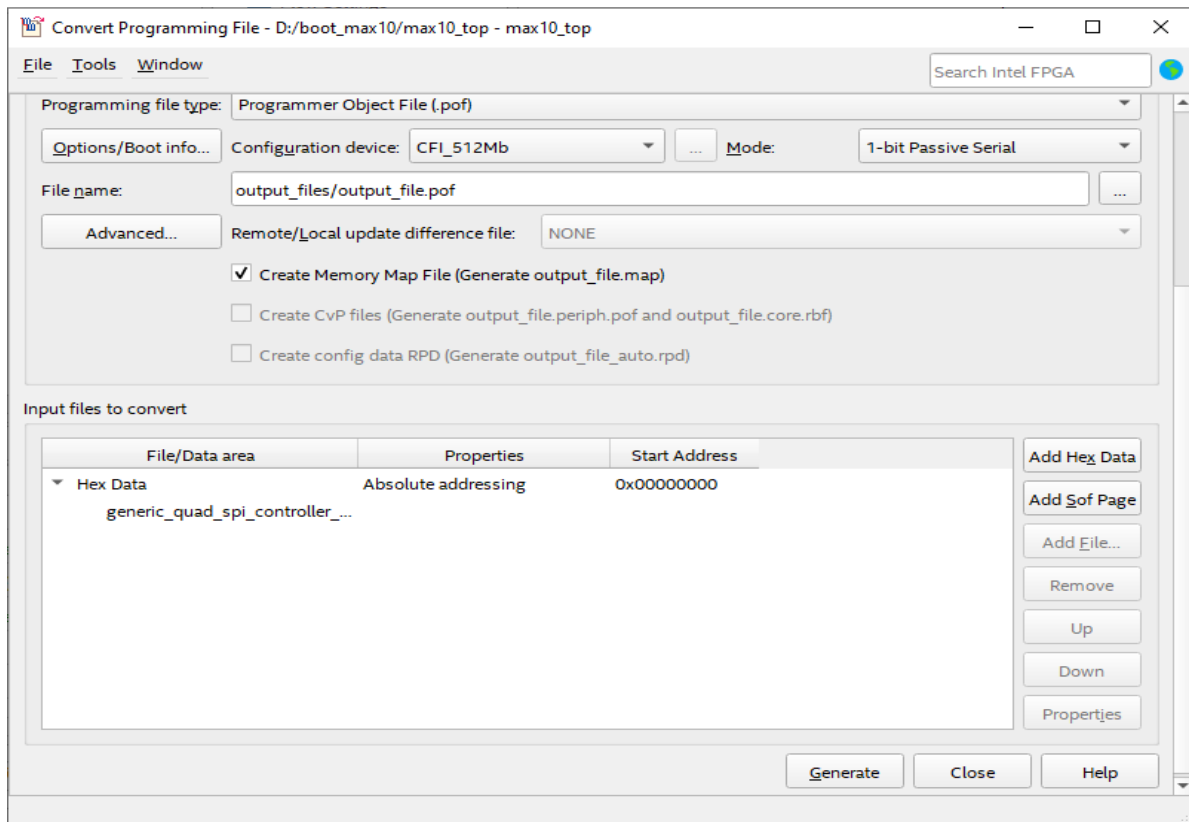
Linker Region Name	Address Range	Memory Device Name	Size (bytes)	Offset (bytes)
onchip_memory2_0	0x08008020 - 0x0800...	onchip_memory2_0	31968	32
onchip_memory2_0_BEFORE_EXCEPTION	0x08008000 - 0x0800...	onchip_memory2_0	32	0
generic_quad_spi_controller_0_av...	0x04000020 - 0x07FF...	generic_quad_spi_controller_0_av...	67108832	32
reset	0x04000000 - 0x0400...	generic_quad_spi_controller_0 av...	32	0

The hex file was then generated for the above settings.

2- Converting the hex file generated into POF File.

The following settings were applied for generating the POF file as mentioned in [AN 730](#)

1. In Quartus II, click on **Convert Programming Files (.pof)** from the **File** tab.
2. Choose **Programmer Object File** as **Programming file type**.
3. Set **Mode** to **1-bit Passive Serial**.
4. Set **Configuration device** to **CFI_512Mb**.
5. Change the **File name** to the desired path and name.
6. Remove the **SOF Page_0**.
7. Click on **Add HEX Data**, choose the HEX file generated in **HEX Generation section**. Select **Absolute Addressing** and click **OK**.
8. Click **Generate** to create the .pof file.



3- Programming the above generated POF into QSPI Flash.

--> To detect the **QSPI Flash** in the Programmer, a **sof of Parallel Flash Loader** was uploaded into the MAX 10 device before programming the QSPI flash.(As mentioned in [AN 730](#))

Programmer - D:/boot_max10/max10_top - max10_top - [max10_top.cdf]*

Hardware Setup... USB-BlasterII [USB-1] 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	Security Bit	Erase	Reaccks	ISP CLAMP	IPS File
<none>	10M50DAF484C6...	00000000	00000000	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
output_files/output...	QSPI_512Mb	FBE88E9C		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
generic_quad_s...				<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Start Stop Auto Detect Delete Add File... Change File... Save File Add Device... Up Down

QSPI_512Mb

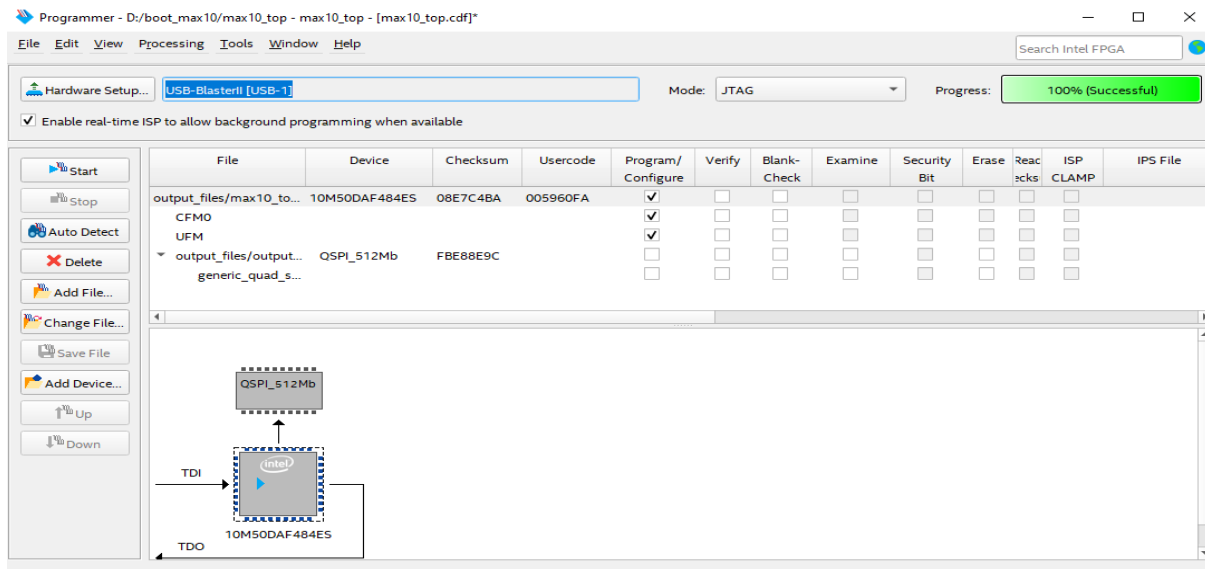
TDI

10M50DAF484ES

TDO

4- Programming the POF generated during Compilation into MAX10 Device

After programming the **POF File** (containing the Hex File) successfully, the POF File generated during the **Compilation Process** was programmed into the MAX10 Device.



Result: After completing all the steps, on Power Cycle of MAX10 Kit the LED[0] was not blinking according to the software code. NIOS Processor does not boot using QSPI Flash.