Perfection in Automation
www.br-automation.com

POWERLINK IP-Core
Altera FPGA Documentation

Date: January 30, 2012

Project Number: AT-B0-000002

I Versions

Version	Date	Comment	Edited by
0.1	Dec 14, 2011	First Edition	Zelenka Joerg
0.2	Jan 11, 2012	Added FPGA Resource Utilization and Timing Consideration	Zelenka Joerg

Table 1: Versions

II Safety Notices

Safety notices in this document are organized as follows:

Safety notice	Description
Danger!	Disregarding the safety regulations and guidelines can be life-threatening.
Warning!	Disregarding the safety regulations and guidelines can result in severe injury or heavy damage to material.
Caution!	Disregarding the safety regulations and guidelines can result in injury or damage to material.
Information:	Important information used to prevent errors.
Example:	Functionality is described with an example to prevent errors.

Table 2: Safety notices

III Table of Contents

1 Introduction

The POWERLINK IP-Core is provided as ready-to-use in Altera Quartus II environment with SOPC version 10.1. This documentation presents the integration of the IP-Core into the SOPC environment, as well as timing considerations and interfacing to the software development environment.

If you require detailed information about the POWERLINK IP-Core itself, please refer to the "POWERLINK IP-Core Generic Documentation" (POWERLINK-IP-Core_Generic.pdf).

The POWERLINK IP-Core is implemented generically, which enables to use the same VHDL-sources in any FPGA environment. For instance memory components are given in extra VHDL-files ready to use in Cyclone III or IV devices. Experienced users can modify these files easily to port the IP-Core to other devices.
The IP-Core configuration is supported by an easy-to-use graphical user interface (GUI), which is called in Altera SOPC. This enables the user to define several parameters for the POWERLINK node, like the type of API (e.g. SPI or parallel interface), the number of process data objects (PDO) or enabling low-jitter synchronization feature.

It has to be emphasized that no VHDL or FPGA knowledge is required to setup a POWERLINK node on an FPGA successfully. All the necessary configuration is done depending on the settings defined in SOPC.

2 System-On-a-Programmable-Chip (SOPC) Integration

The POWERLINK IP-Core's "Internal Bus Interface" complies with the Altera Avalon Interface Specification [1], hence there is no extra VHDL wrapper file necessary for converting purpose. The IP-Core is fully compatible to the Avalon Memory Mapped interface, hence it can be used in combination with e.g. Altera Nios II processors.

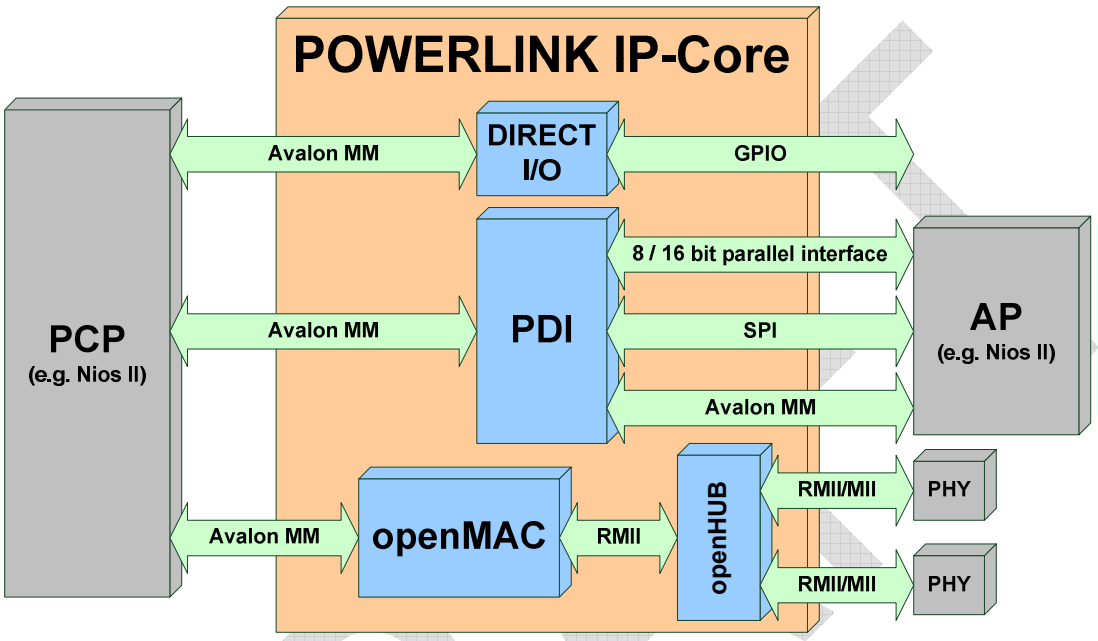

Fig. 1: POWERLINK IP-Core Block Diagram

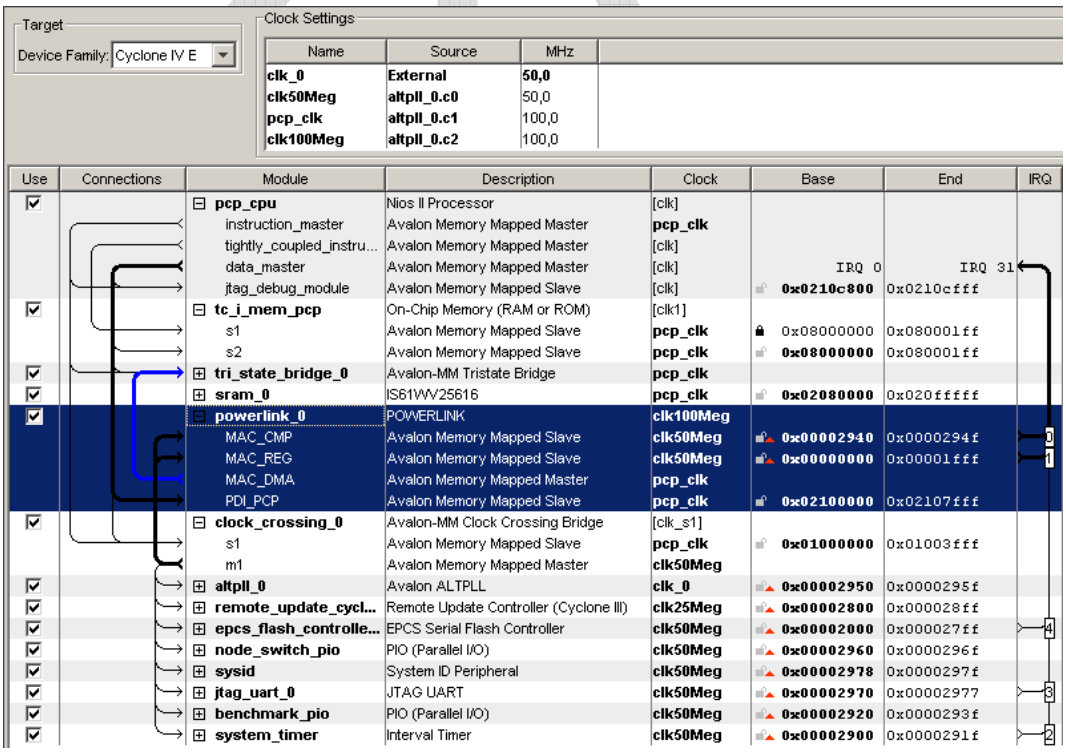

Fig. 2: Example SOPC with POWERLINK IP-Core

The SOPC allows easy configuration of the IP-Core, thus the user is able to define parameters depending on the application of the POWERLINK Node. Fig. 1 visualizes the block diagram of the IP-core, emphasizing the interfaces to the POWERLINK Communication Processor (PCP) and to the application with the optional Application Processor (AP). Please note that the PDI and DIRECT I/O blocks are mutual exclusive, depending on the SOPC configuration. However, the MAC-layer components openMAC and openHUB[1] are mandatory components of a POWERLINK node.

In Fig. 2 an SOPC example is shown, which includes the PCP as a Nios II/s, SRAM memory controller and other components available in the SOPC. Depending on the configurations of the POWERLINK IP-Core the Avalon Memory Mapped Slave/Master interfaces may vary, however, MAC_REG and MAC_CMP are mandatory in any case.
Fig. 3 shows the GUI for the POWERLINK IP-Core configuration. The parameters are separated into subgroups (e.g. General Settings or Process Data Interface Settings), which are arranged in descending order (starting with general settings). The user has to set the intended POWERLINK Slave Design Configuration, which affects the subsequent parameters' visibility (e.g. in case of "openMAC only" no PDI settings are possible or meaningful).

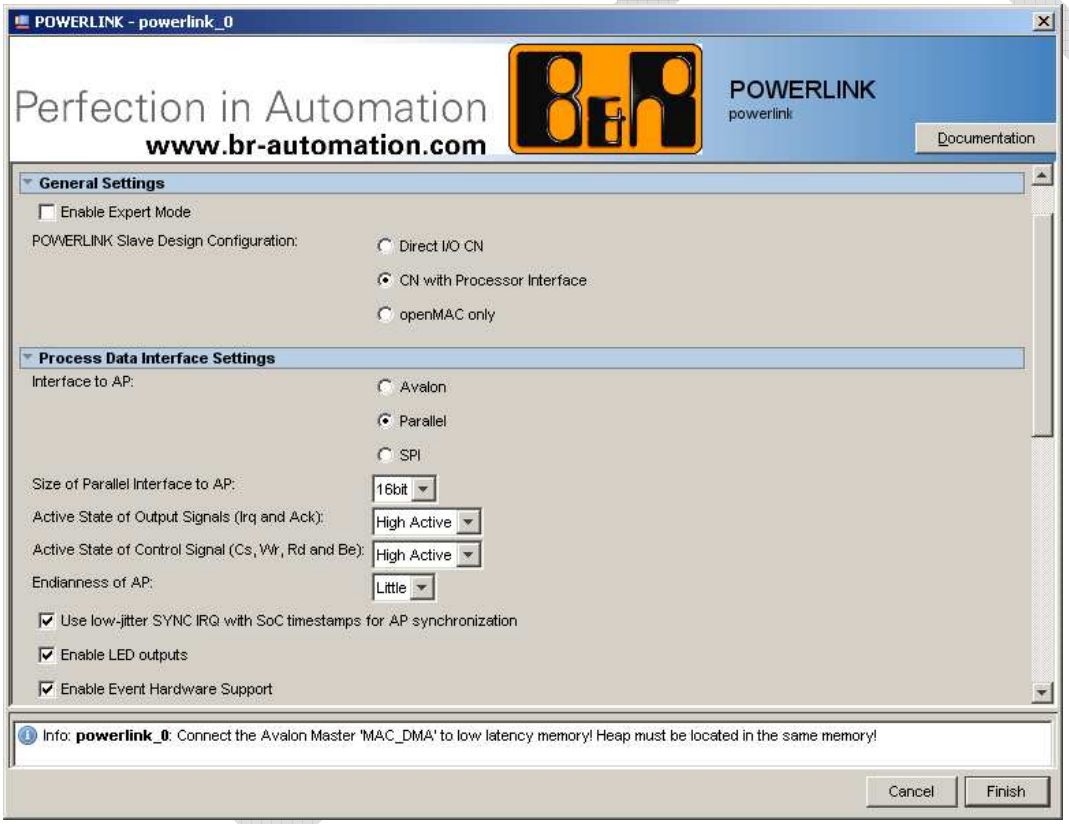

Fig. 3: GUI of POWERLINK IP-Core with example configuration

The configuration set in the GUI by the user is verified to test feasibility, and in addition messages are forwarded to SOPC in the bottom of the window (e.g. "Info: Powerlink_0: Consider to use RMII to reduce resource usage!").

[1] OpenHUB is only required if a second Ethernet interface is available on your hardware platform.

3 Timing Consideration

The POWERLINK IP-Core and the mandatory components (e.g. PCP and memory controller) require further configuration if implemented in an FPGA. In order to work properly timing constraints have to be applied to the design, which is done by SDC-files directing the Altera TimeQuest timing analyzer tool. The files are delivered with the POWERLINK IP-Core (refer to Tab. 6) and have to be forwarded to TimeQuest (in Quartus II: Assignments – Settings – TimeQuest – "SDC files to include in project"). The timing constraints are considered by Quartus II during place & route, which ensures an optimized fitter result within short compilation times.

The delivered SDC-files include the configuration of the POWERLINK IP-Core (e.g. PHY interface and parallel interface) and the SRAM interface. If the user adds additional components to the SOPC (e.g. SDRAM or GPIO), timing constraints have to be added manually.

It is important that the design's top level pin-names match those defined in the SDC files (e.g. EXT_CLK or PHY0_TXEN). Otherwise TimeQuest is not able to apply constraints properly!

Further information about TimeQuest and basics on timing constraints are provided by Altera (www.altera.com).

4 Software Interfacing (system.h)

The POWERLINK IP-core is included into an SOPC (Altera) or any other processor environment. In order to provide vital information to the PCP a header file is used (e.g. system.h). The POWERLINK IP-core provides parameters depending on the settings done via the SOPC GUI as listed in Tab. 1. Note that several parameters are described in more detail in Tab. 2.

Tab. 1: System description parameters

Name	Macro	Value
MAC buffer size	MACBUFSIZE	Any integer (0 if PKTLOC = 2)
MAC RX buffer size	MACRXBUFSIZE	Any integer (0 if PKTLOC = 1 or 2)
MAC RX buffers	MACRXBUFFERS	1 … 16
MAC TX buffer size	MACTXBUFSIZE	Any integer (0 if PKTLOC = 2)
MAC TX buffers	MACTXBUFFERS	Any integer (0 if CONFIG = 5)
Number of MAC CMP timer	CMPTIMERCNT	1 or 2 (1 if CONFIG = 0 or 5)
Number of PDI RPDOs	PDIRPDOS	1, 2 or 3 (0 if CONFIG = 5)
Number of PDI TPDOs	PDITPDOS	1 (0 if CONFIG = 5)
POWERLINK IP-core configuration	CONFIG	0, 1, … , 5
AP interface endianness	CONFIGAPENDIAN	0 or 1
Number of phys	PHYCNT	1 or 2
MAC packet buffer location	PKTLOC	0, 1 or 2
Time synchronization feature enable	TIMESYNC	0 or 1 (false or true)
LED feature enable	LEDGADGET	0 or 1 (false or true)
MAC DMA observer feature enable	DMAOBSERVER	0 or 1 (false or true)

Tab. 2: System description parameters, details

Macro	Value	Description
CONFIG	0	Direct I/O (no AP intended)
	1	8bit parallel interface
	2	16bit parallel interface
	3	SPI
	4	Avalon bus
	5	openMAC only (no PDI available)
CONFIGAPENDIAN	0	Little endian
	1	Big endian
PKTLOC	0	TX and RX packets are stored in MAC-internal DPRAM
	1	TX packets are stored in MAC-internal DPRAM, RX buffers are located in system's heap
	2	TX and RX packets are stored in system's heap

5 FPGA Resource Utilization

The POWERLINK IP-Core is used in combination with other components in an FPGA, hence the resource utilization and the circuit's performance (f_{max}) given in this section is an estimation only. The M9K utilization is not given since it depends on several parameters defined in the SOPC. Note that the parameters not noted in the table are set to default values (not changed in GUI).

Tab. 3: Resource utilization on Cyclone 4 FPGA (EP4CE115F29C7)

PARAMETERS								FITTER Results		
IP CORE MODE	PACKET BUFFER LOCATION [1]	USE RMII	USE 2ND PHY	NUMBER OF RPDO	LOW-JITTER SYNC	EVENT HW SUPPORT	PAP DATA WIDTH	LEs (POWERLINK [2])	LEs (SYSTEM [3])	fmax [MHz] [4]
Direct I/O	2	×	✓	3	-	-	-	2006	4988	105
Parallel Interface	2	×	✓	3	×	×	8	2351	5388	105
Parallel Interface	2	×	✓	1	×	×	16	2313	5306	105
Parallel Interface	2	×	✓	2	×	×	16	2392	5399	105
Parallel Interface	2	×	✓	3	×	×	16	2448	5464	105
Parallel Interface	2	×	✓	3	✓	×	16	2547	5574	105
Parallel Interface	2	×	✓	3	✓	✓	16	2572	5604	104
SPI	2	×	✓	3	×	×	-	2432	5460	106
Avalon	2	×	✓	3	×	×	-	2307	5429 [5]	105
openMAC only	0	×	✓	-	-	-	-			
openMAC only	1	×	✓	-	-	-	-			
openMAC only	2	×	✓	-	-	-	-			
openMAC only	2	✓	✓	-	-	-	-			

Table note:

1. The Packet Buffer Location determines the location of the openMAC's packet buffers (0 = "TX and RX into DPRAM", 1 = "TX into DPRAM and RX over Avalon" or 2 = "TX and RX over Avalon").
2. The fitter results are given for the POWERLINK IP-Core only.
3. The fitter results are given for the whole system design including the recommended component setup in SOPC (e.g. Nios II/s, clock-crossing bridge and memory controller).
4. The f_{max} values give the circuit's performance of the 100 MHz clock in slow corner (85℃).
5. The resource utilization does not include any application-specific components (e.g. second Nios II).
6. Unused parameters are identified with a hyphen ("-")

6 Files

The POWERLINK IP-Core for Altera FPGA is delivered with the necessary VHDL files and the SOPC component description (*_hw.tcl). In Tab. 4 to Tab. 7 the package directories are described.

Information:

It is essential that the package is stored in a subdirectory of your Quartus project. Otherwise SOPC cannot recognize the package, and in addition the mif-directory (Tab. 4) is created incorrectly!
Consider to apply the following path example: $QUARTUS_PRJ/POWERLINK

Tab. 4: Directory description

Directory name	Description
doc	The doc-directory includes all IP-Core documentation files.
img	In the img-directory images for the GUI in SOPC are stored.
sdc	The sdc-directory provides different timing-constraints file usable for TimeQuest.
src src/lib src/openMAC_DMAmaster	The VHDL-files are stored in the src-directory.
../mif	The mif-directory is created when inserting the POWERLINK IP-core successfully into SOPC. It provides memory initialization files necessary for the IP-core during compilation of Quartus II.

Tab. 5: Documentation file description (doc)

File name	Description
OpenMAC.pdf	The “openMAC & Components Documentation” introduces the IP-Cores openMAC, openFILTER and openHUB, as well as the software drivers (omethlib).
POWERLINK-IP-Core_Altera.pdf	This document.
POWERLINK-IP-Core_Generic.pdf	The “POWERLINK IP-Core Generic Documentation” provides a detailed description of the IP-Core.

Tab. 6: Timing Constraints file description (sdc)

File name	Description
PLK_MII_base.sdc	Defines the timing constraints for MII phys.
PLK_RMII_base.sdc	Defines the timing constraints for RMII phys.
PLK_RMII_base_PARPDI.sdc	Defines the timing constraints for RMII phys and a parallel asynchronous 8/16 bit interface (to the AP).

Tab. 7: VHDL source files (src)

File name	Description
lib/addr_decoder.vhd	Address decoder used in several components of the POWERLINK IP-Core
lib/edge_det.vhd	Component to detect level changes of a signal
lib/memMap.vhd	Package file used in pdi.vhd
lib/req_ack.vhd	Used to generate wait- or acknowledge signals
lib/sync.vhd	Two-stage FF synchronizer
lib/slow2fastSync.vhd	Synchronizer to transfer pulse signals
OpenMAC_Ethernet.vhd	MAC-layer top-level file
OpenFILTER.vhd	RMII filter that prevents distortions to propagate

OpenHUB.vhd	Ethernet hub
OpenMAC.vhd	Media Access Controller with dedicated hardware acceleration for POWERLINK
OpenMAC_16to32conv.vhd	Converter to access with a 32 bit CPU the openMAC register interface
OpenMAC_cmp.vhd	OpenMAC high-resolution timer component
OpenMAC_DMAFifo_Altera.vhd	Dual-clocked FIFO provided to the DMA master implementation
OpenMAC_DMAmaster.vhd	DMA master top-level file
OpenMAC_DMAmaster/dma_handler.vhd	DMA handler sub-component
OpenMAC_DMAmaster/master_handler.vhd	DMA master handler sub-component
OpenMAC_DPR_Altera.vhd	Dual Ported RAM description used in OpenMAC_*.vhdl files
OpenMAC_phyAct.vhd	Component to generate phy-activity LED signal
OpenMAC_PHYMI.vhd	Component to configure the phys via SMI
OpenMAC_rmii2mii.vhd	RMII to MII converter
pdi.vhd	Process Data Interface top-level file
pdi_apIrqGen.vhd	Sub-component to generate sync-interrupts to the AP
pdi_controlStatusReg.vhd	Sub-component implementing the Status/Control register
pdi_dpr_Altera.vhd	Dual Ported RAM description used in pdi.vhd
pdi_event.vhd	Sub-component providing hardware supported event handling
pdi_led.vhd	Sub-component providing LED interface
pdi_par.vhd	Asynchronous 8/16 bit parallel interface to AP
pdi_simpleReg.vhd	Sub-component implementing e.g. asynchronous buffer in PDI
pdi_spi.vhd	SPI interface to AP
spi.vhd	SPI slave implementation used by pdi_spi.vhd
spi_sreg.vhd	SPI shift register implementation used by spi.vhd
Pdi_tripleVBufLogic.vhd	Sub-component used to generate triple-buffer management of PDOs
portio.vhd	Direct I/O top-level file
portio_cnt.vhd	Counter for direct I/O data valid signal generator
powerlink.vhd	POWERLINK IP-Core top-level file

Table note: The highlighted rows refer to top-level files.

7 Definitions and Abbreviations

AP	Application Processor
BUF	MAC-internal packet buffer
CMD	Command
CMP	Compare Unit
CN	POWERLINK Controlled Node
DPR	Dual Ported RAM
DPRAM	Dual Ported RAM
FF	Flip Flop
FPGA	Field Programmable Gate Array
FSM	Finite State Machine
GUI	Graphical User Interface
IP	Intellectual Property
IRQ	Interrupt Request
MII	Media Independent Interface
MN	POWERLINK Managing Node
PCB	Printed Circuit Board
PCP	Powerlink Communication Processor
PDI	Process Data Interface
PDO	Process Data Object
PReq	Poll Request (POWERLINK frame type)
PRes	Poll Response (POWERLINK frame type)
RD	Read
RDY	Ready
RMII	Reduced Media Independent Interface
RO	Read Only
SDC	Synopsys Design Constraints
SDO	Service Data Object
SMI	Serial Management Interface (Ethernet phy register access)
SMP	Simple I/O Port
SoA	Start of Asynchronous (POWERLINK frame type)
SoC	Start of Cyclic (POWERLINK frame type)
SPI	Serial Peripheral Interface
SYNC	Synchronization (Clock Domain Crossing)
WR	Write

8 References

[1] Altera: “Avalon Interface Specification”
http://www.altera.com/literature/manual/mnl_avalon_spec.pdf
Version 11.0 (May 2011)

9 Figure Index

10 Table Index

11 Index