« Real Time Embedded systems »
Cyclone V – SOC - FPGA

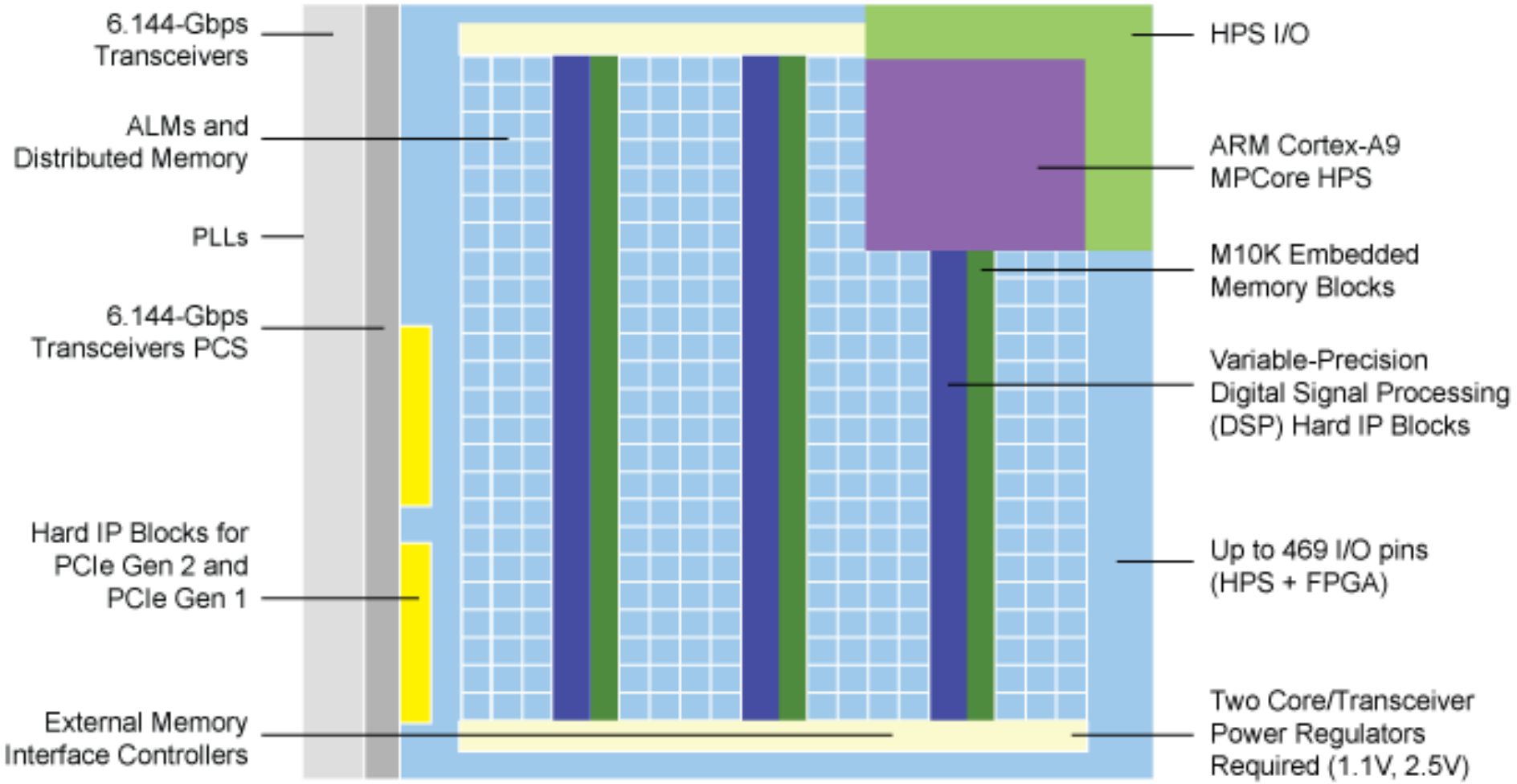
rene.beuchat@epfl.ch
LAP/ISIM/IC/EPFL
Chargé de cours
rene.beuchat@hesge.ch
LSN/hepia
Prof. HES

Ref: http://www.altera.com

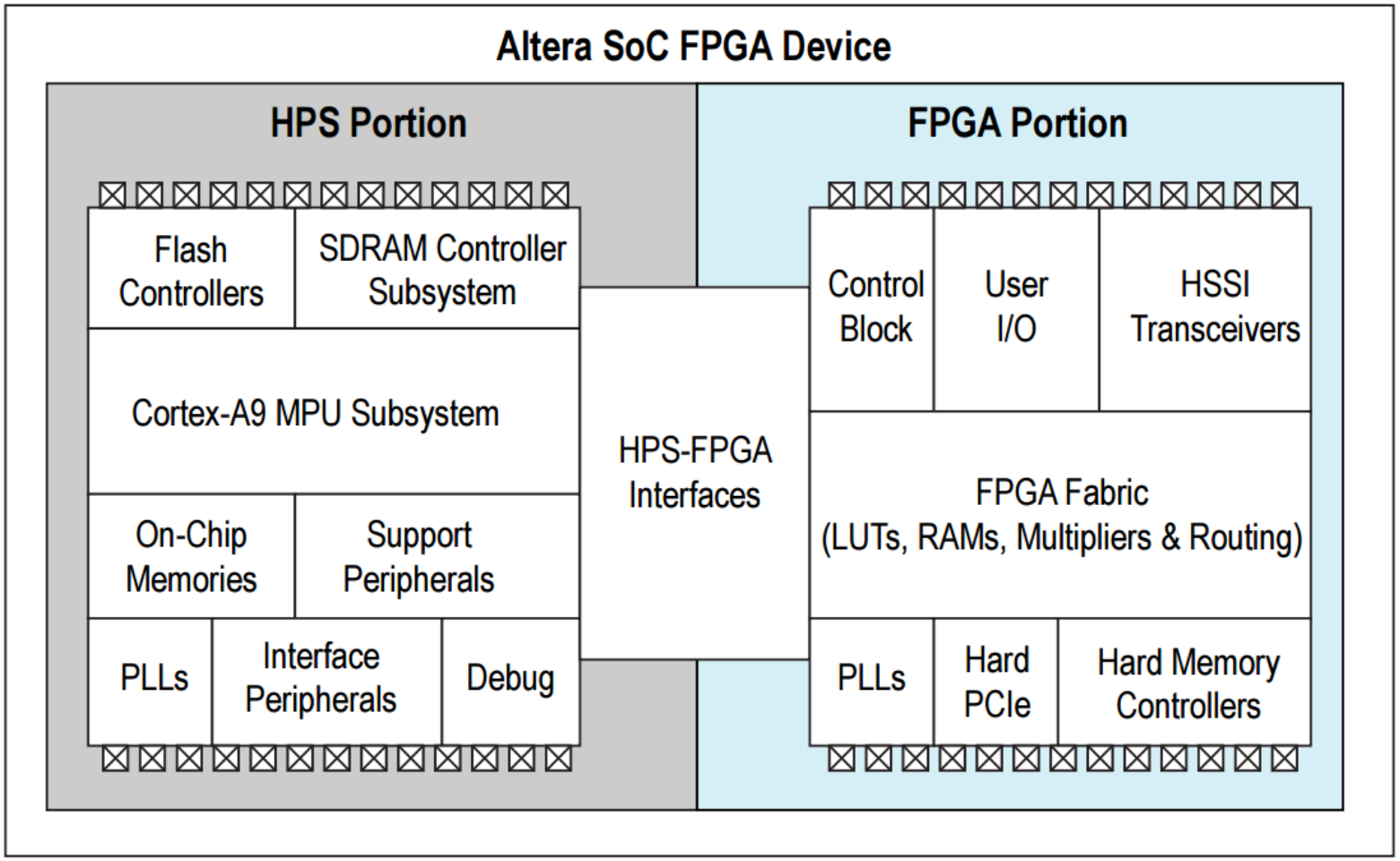
SOC + FPGA (ex. Cyclone V, Xilinx Zynq)

- New generation of FPGA include today
  - FPGA parts
  - Hardcore units
- System On Chip with programmable part for highly specialized systems on One Chip and high performance.

SOC + FPGA (ex.CycloneV)



SOC + FPGA (ex.CycloneV)

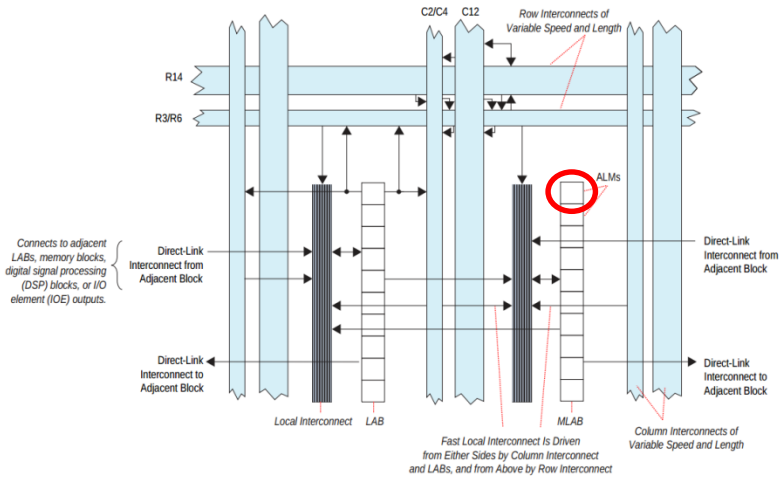


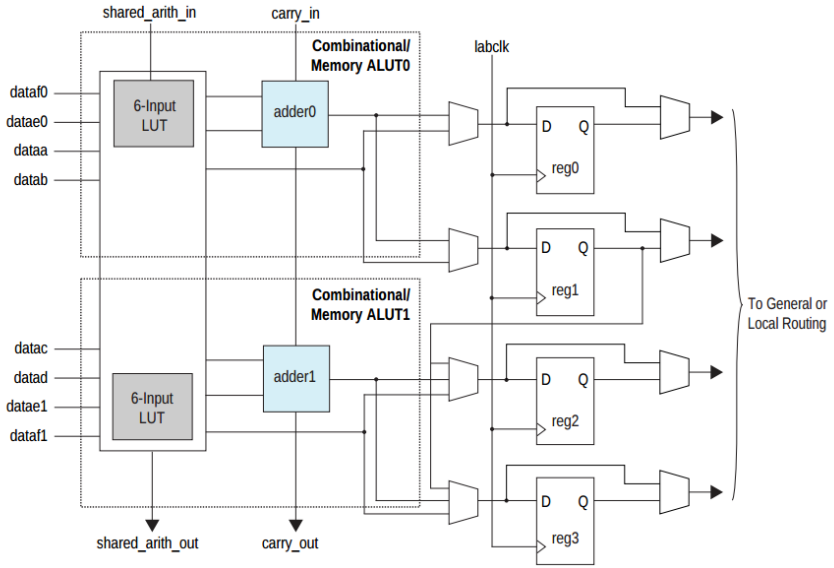
SOC + FPGA (ex. Cyclone V)

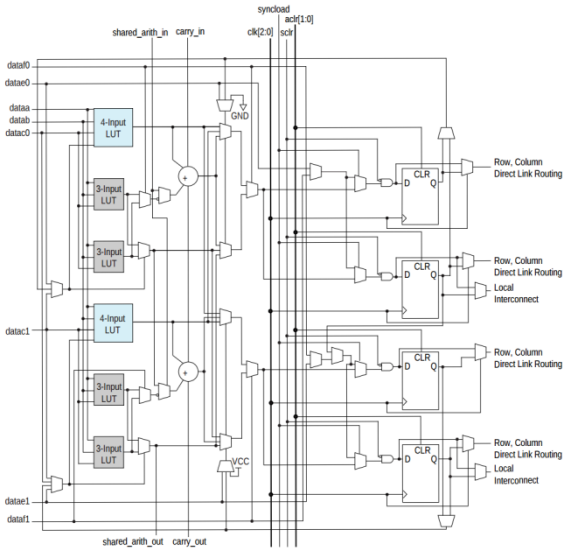
- FPGA part
  - ALM (Adaptative Logic Module)
    - 4 registers
    - Many modes of operations:
      - Normal mode
      - Extended LUT mode
      - Arithmetic mode
      - Shared arithmetic mode
  - Memory (M10k blocks)
  - DSP (Digital Signal Processing) blocks

ALM : Adaptative Logic Module

- ALM (Adaptative Logic Module)
- → 32 x 2 Memory Block
- 4 registers



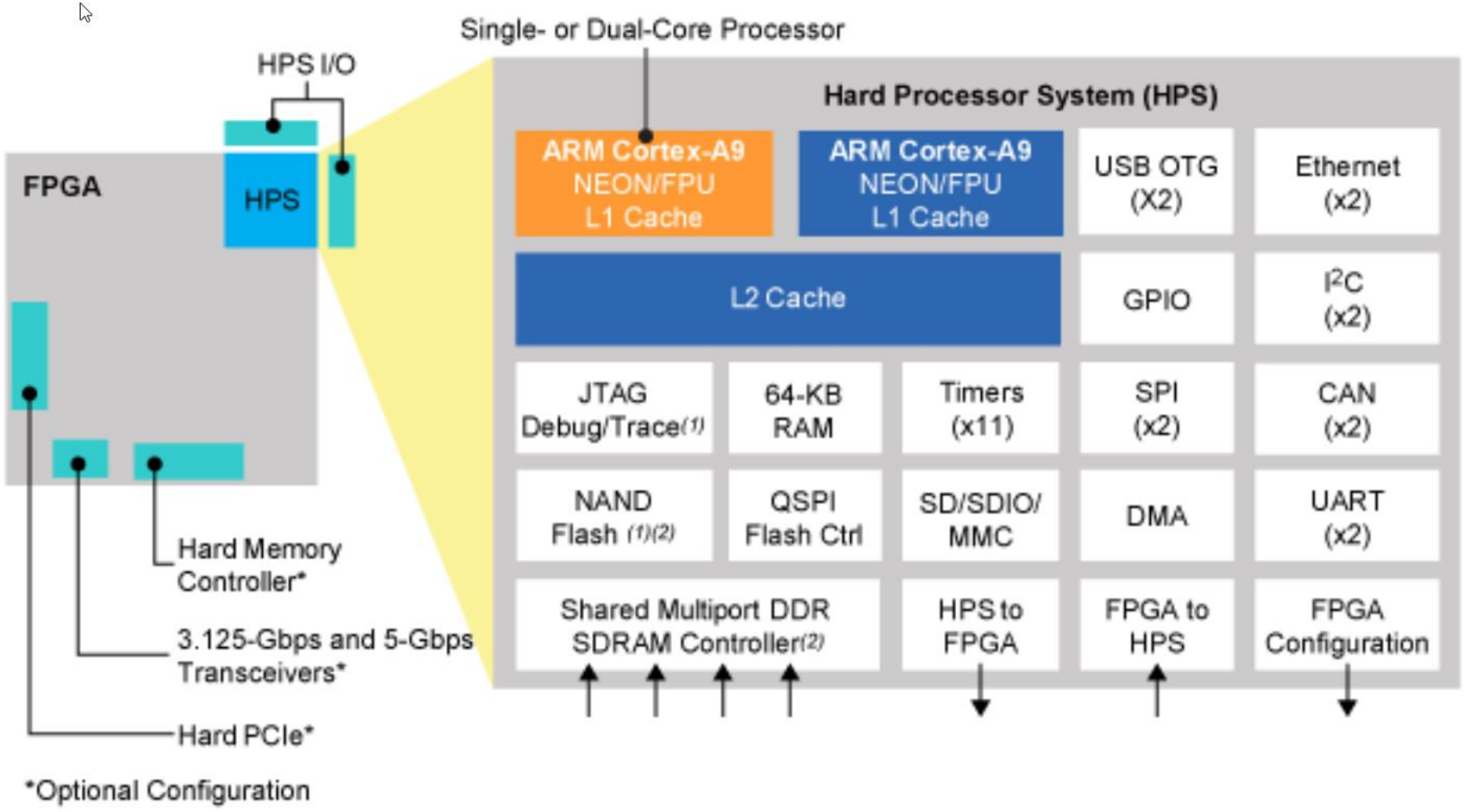




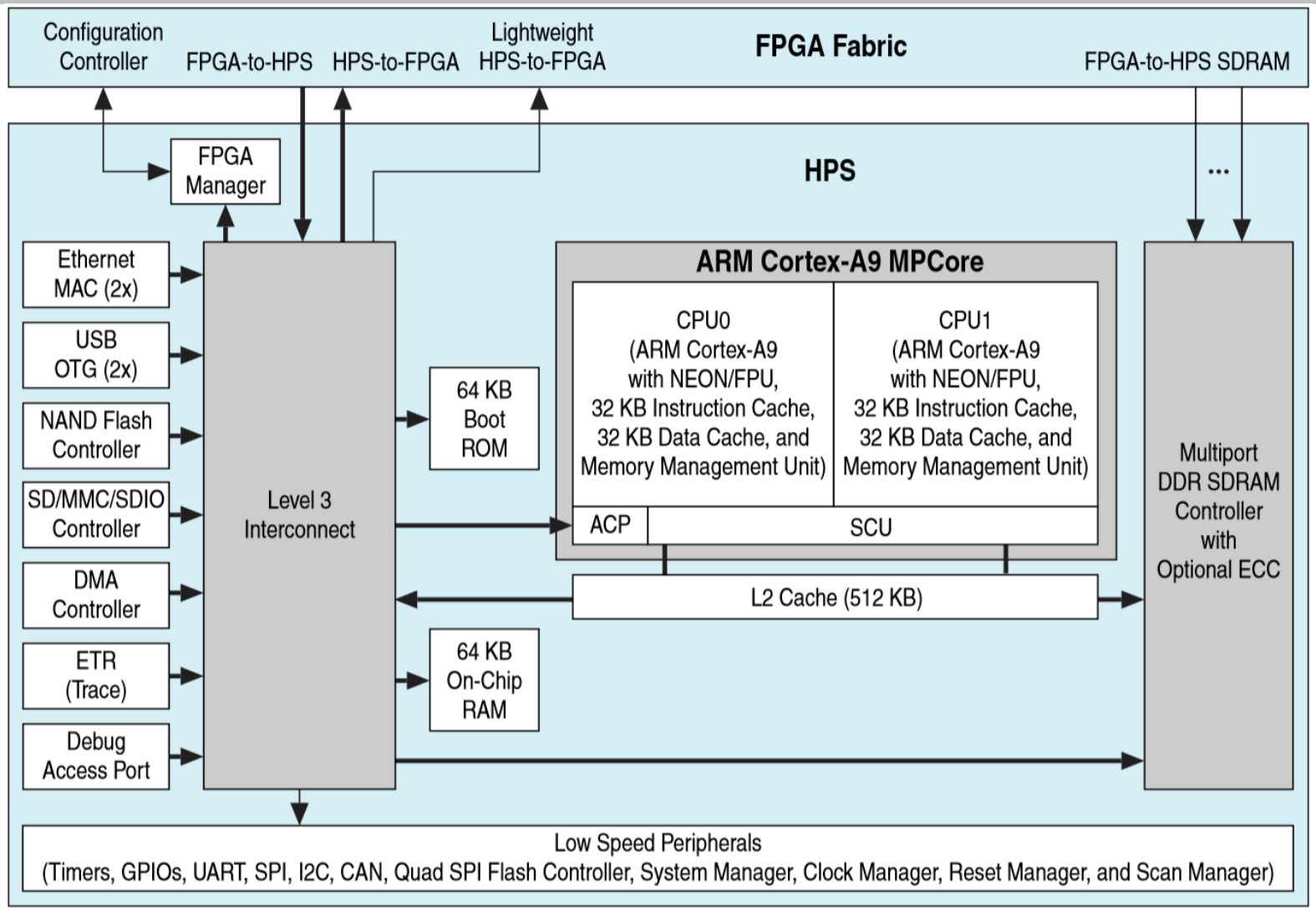
SOC + FPGA (ex. Cyclone V)

- Hardcore part
  - 2 x ARM Cortex-A9 core
    - + NEON™ SIMD coprocessor
    - +FPU
  - Snoop Control Unit (SCU)
  - Accelerator Coherency Port (ACP)
  - Many programmable interfaces
  - External memory ctrl (DDRx)
  - PCIe (opt.)
  - High speed link (6.144 Gbps) (opt.)
  - HPS I/O

SOC + FPGA (ex.CycloneV)

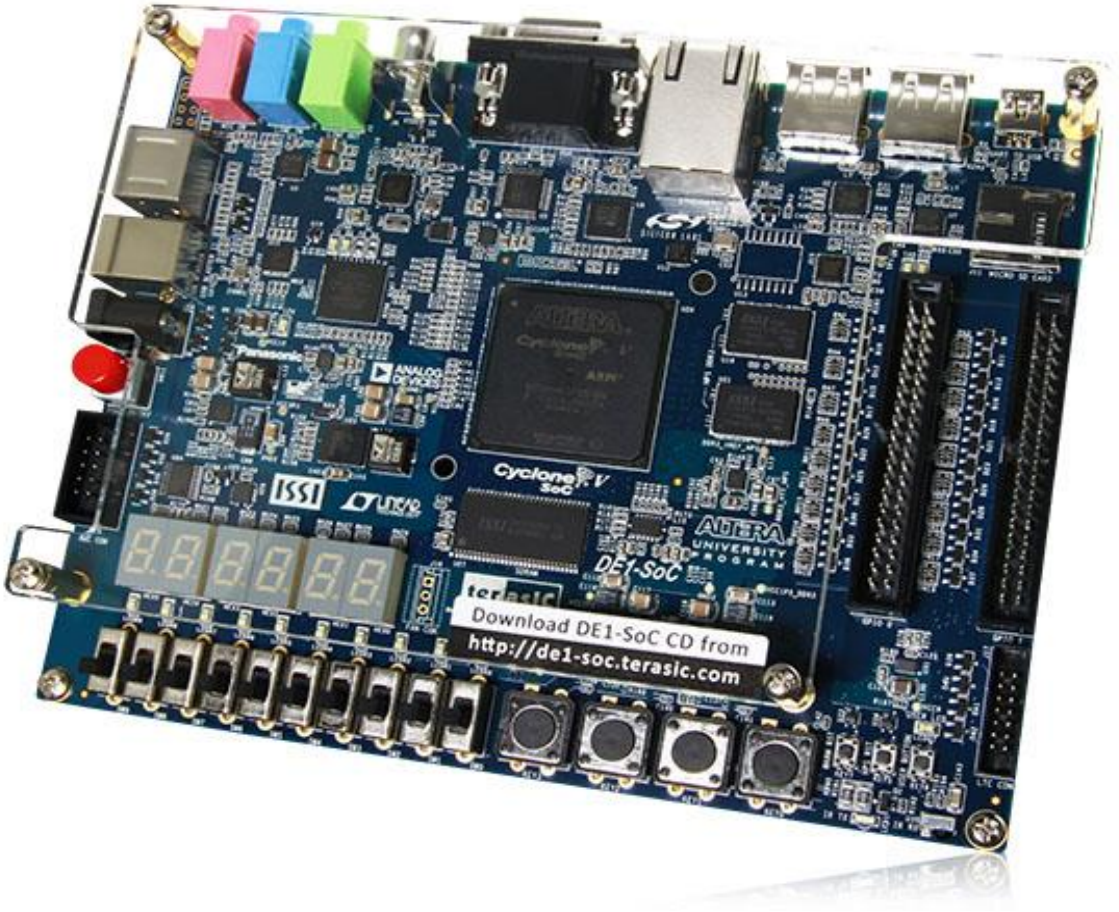


SOC + FPGA (ex.CycloneV)



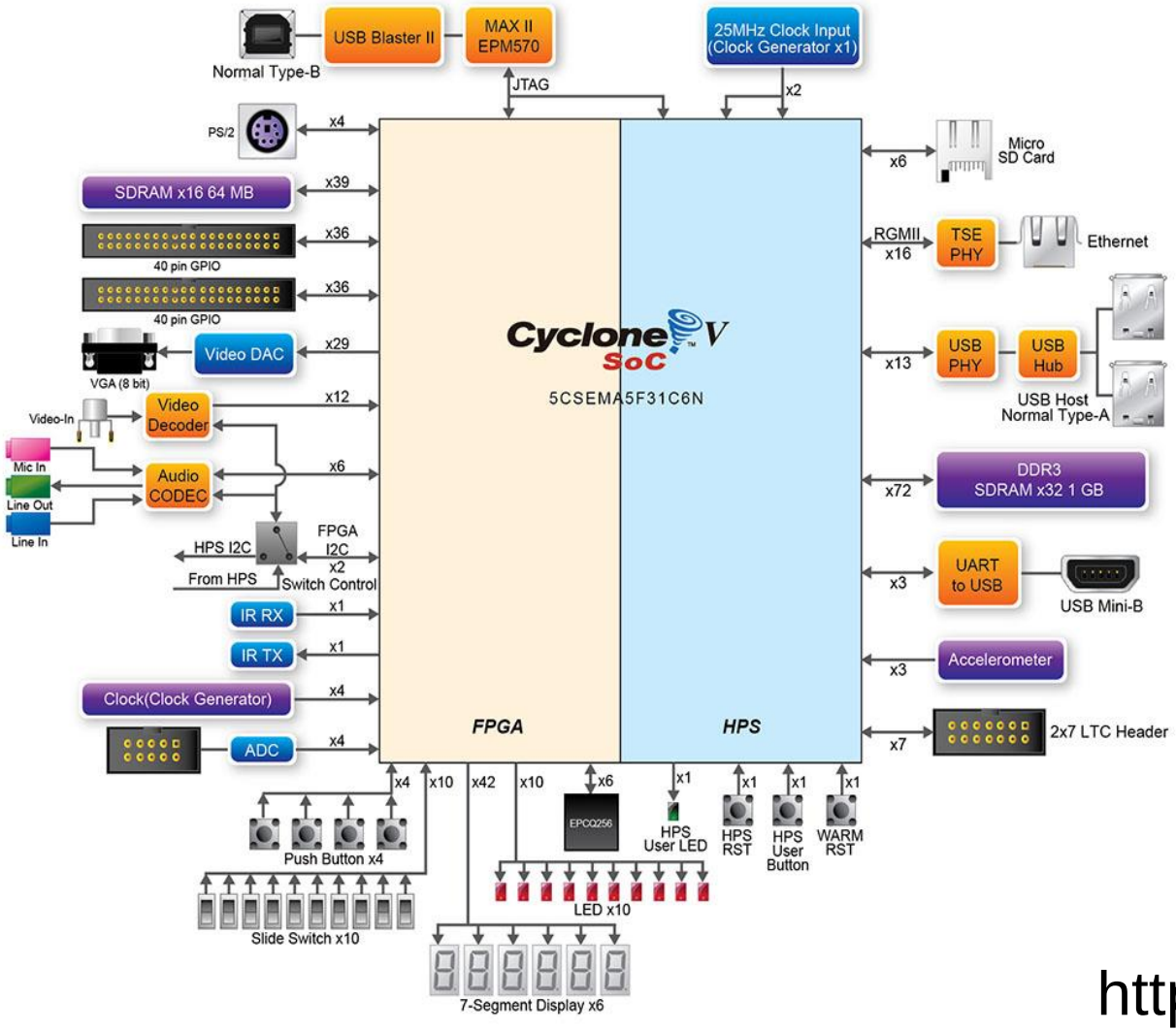
DE1-SOC (Terasic)

- Board for test and laboratories
- Cyclone V SoC ***5CSEMA5F31C6*** Device
- ***64 MB*** (32Mx16) **SDRAM** on FPGA
- ***1 GB*** (2x256Mx16) **DDR3** SDRAM on HPS
- ***MICRO SD*** Card Socket on HPS
- Ethernet 10/100 Mbps
- VGA
- USB, …



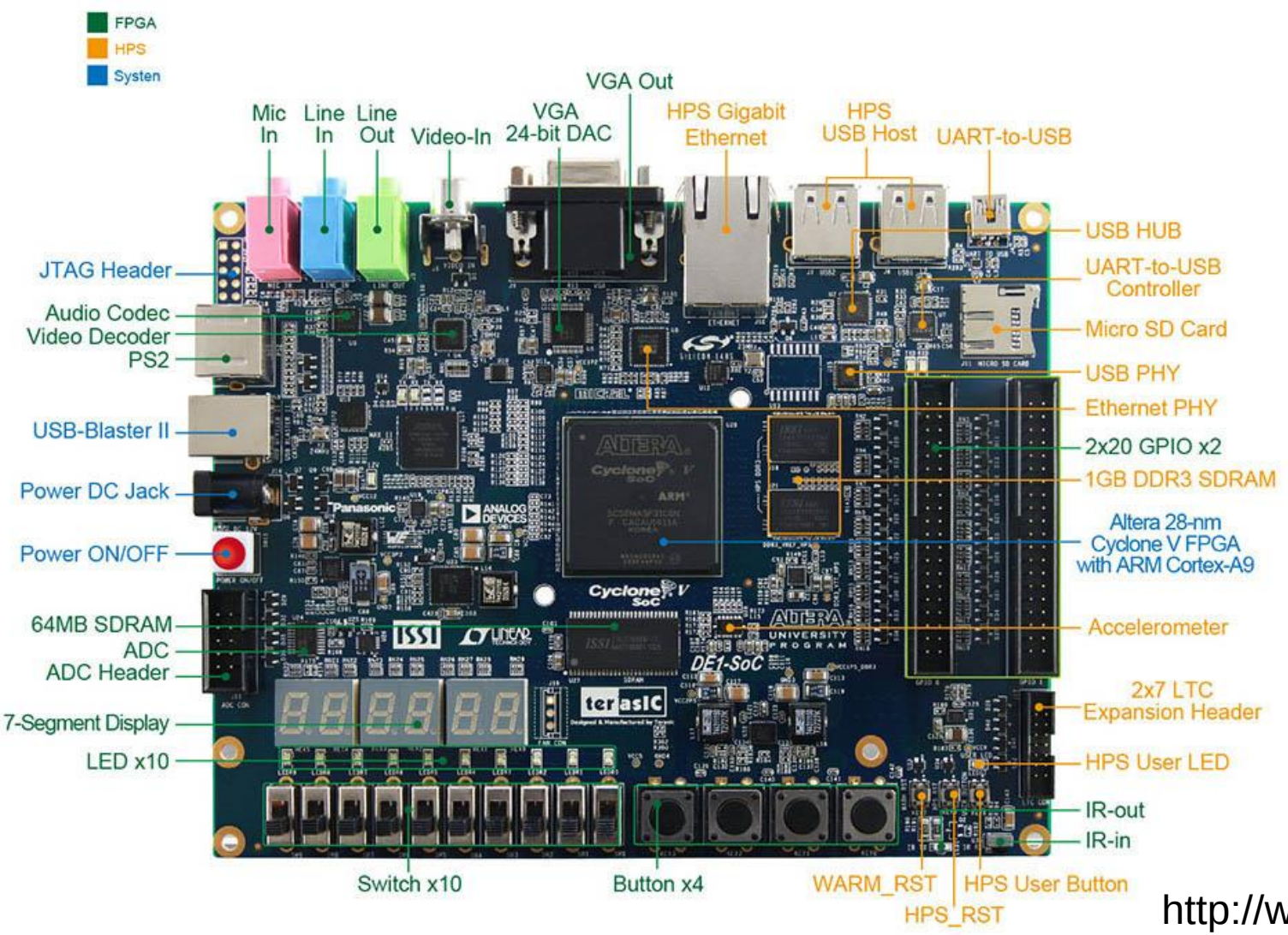
http://www.terasic.com

DE1-SOC (Terasic)



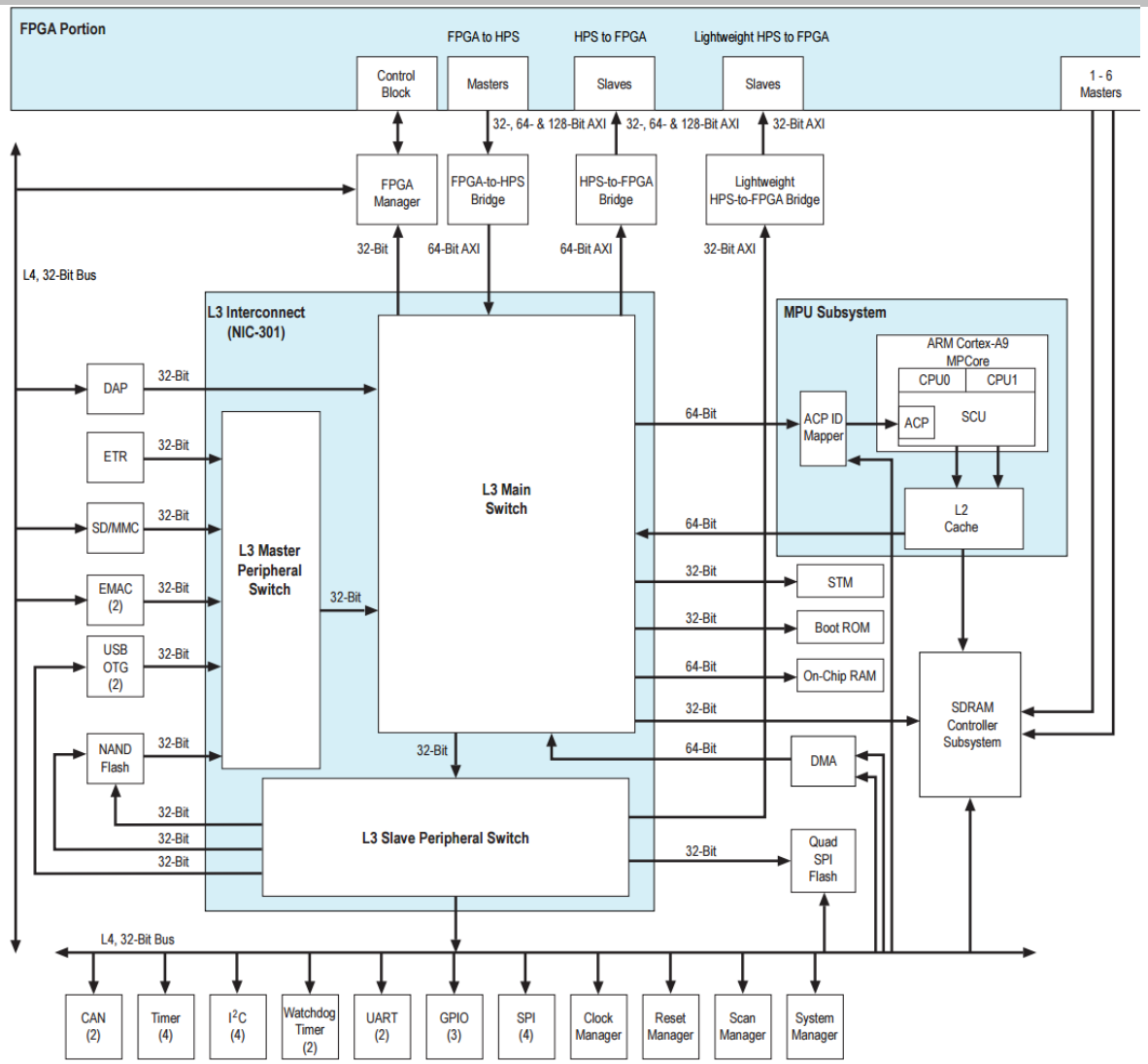
http://www.terasic.com

DE1-SOC (Terasic)



http://www.terasic.com

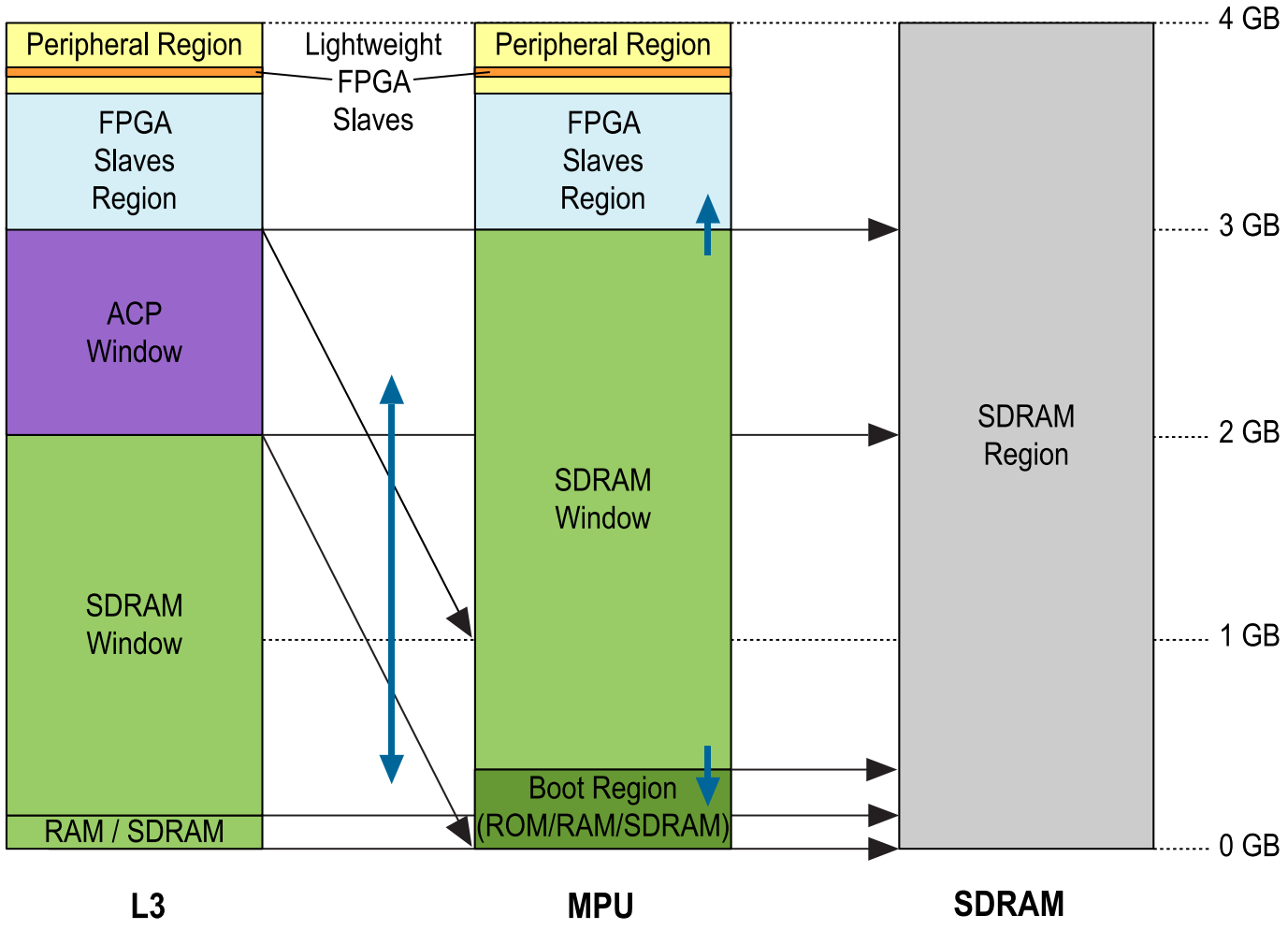
SOC + FPGA (ex.CycloneV)



HPS Address Space

HPS 3 address spaces

- L3
- MPU
- SDRAM



HPS L3/MPU Address Space

Region Name	Description	Base Address	Size
FPGA slaves	FPGA slaves connected to the HPS-to-FPGA bridge	0xC000 0000	960 MB
HPS peripherals	Slaves directly connected to the HPS	0xFC00 0000	64 MB
Lightweight FPGA slaves	FPGA slaves connected to the lightweight HPS-to-FPGA bridge	0xFF20 0000	2 MB

HPS Peripheral Region Address Map

Slave Identifier	Slave Title	Base Address	Size
STM	STM	0xFC00 0000	48 MB
DAP	DAP	0xFF00 0000	2 MB
LWFPGASLAVES	FPGA slaves accessed with lightweight HPS-to-FPGA bridge	0xFF20 0000	2 MB
LWHPS2FPGAREGS	Lightweight HPS-to-FPGA bridge GPV	0xFF40 0000	1 MB
HPS2FPGAREGS	HPS-to-FPGA bridge GPV	0xFF50 0000	1 MB
FPGA2HPSREGS	FPGA-to-HPS bridge GPV	0xFF60 0000	1 MB
EMAC0	EMAC0	0xFF70 0000	8 KB
EMAC1	EMAC1	0xFF70 2000	8 KB
SDMMC	SD/MMC	0xFF70 4000	4 KB
QSPIREGS	Quad SPI flash controller registers	0xFF70 5000	4 KB
FPGAMGRREGS	FPGA manager registers	0xFF70 6000	4 KB
ACPIDMAP	ACP ID mapper registers	0xFF70 7000	4 KB
GPIO0	GPIO0	0xFF70 8000	4 KB
GPIO1	GPIO1	0xFF70 9000	4 KB
GPIO2	GPIO2	0xFF70 A000	4 KB
L3REGS	L3 interconnect GPV	0xFF80 0000	1 MB
NANDDATA	NAND controller data	0xFF90 0000	1 MB
QSPIDATA	Quad SPI flash data	0xFFA0 0000	1 MB
USB0	USB0 OTG controller registers	0xFFB0 0000	256 KB
USB1	USB1 OTG controller registers	0xFFB4 0000	256 KB
NANDREGS	NAND controller registers	0xFFB8 0000	64 KB
FPGAMGRDATA	FPGA manager configuration data	0xFFB9 0000	4 KB

HPS Peripheral Region Address Map

Slave Identifier	Slave Title	Base Address	Size
CAN0	CAN0 controller registers	0xFFC0 0000	4 KB
CAN1	CAN1 controller registers	0xFFC0 1000	4 KB
UART0	UART0	0xFFC0 2000	4 KB
UART1	UART1	0xFFC0 3000	4 KB
I2C0	I2C0	0xFFC0 4000	4 KB
I2C1	I2C1	0xFFC0 5000	4 KB
I2C2	I2C2	0xFFC0 6000	4 KB
I2C3	I2C3	0xFFC0 7000	4 KB
SPTIMER0	SP Timer0	0xFFC0 8000	4 KB
SPTIMER1	SP Timer1	0xFFC0 9000	4 KB
SDRREGS	SDRAM controller subsystem registers	0xFFC2 0000	128 KB
OSC1TIMER0	OSC1 Timer0	0xFFD0 0000	4 KB
OSC1TIMER1	OSC1 Timer1	0xFFD0 1000	4 KB
L4WD0	Watchdog0	0xFFD0 2000	4 KB
L4WD1	Watchdog1	0xFFD0 3000	4 KB
CLKMGR	Clock manager	0xFFD0 4000	4 KB
RSTMGR	Reset manager	0xFFD0 5000	4 KB
SYSMGR	System manager	0xFFD0 8000	16 KB
DMANONSECURE	DMA nonsecure registers	0xFFE0 0000	4 KB
DMASECURE	DMA secure registers	0xFFE0 1000	4 KB
SPIS0	SPI slave0	0xFFE0 2000	4 KB
SPIS1	SPI slave1	0xFFE0 3000	4 KB
SPIM0	SPI master0	0xFFF0 0000	4 KB
SPIM1	SPI master1	0xFFF0 1000	4 KB
SCANMGR	Scan manager registers	0xFFF0 2000	4 KB
ROM	Boot ROM	0xFFFD 0000	64 KB
MPUSCU	MPU SCU registers	0xFFFE C000	8 KB
MPUL2	MPU L2 cache controller registers	0xFFFE F000	4 KB
OCRAM	On-chip RAM	0xFFFF 0000	64 KB

HPS Peripheral Region Address Map

Two directories contain all ***HPS***-related ***HEADER FILES***:

- *"<altera_install_directory>/<version>/embedded/ip/altera/hps/altera_hps/hwlib/include"*
  - Contains ***HIGH-LEVEL*** header files that typically contain a few ***FUNCTIONS*** which facilitate control over the HPS components. These functions are all part of Altera's ***HWLIB***, which was created to make programming the HPS easier.

- *"<altera_install_directory>/<version>/embedded/ip/altera/hps/altera_hps/hwlib/include/socal"*
  - Contains ***LOW-LEVEL*** header files that provide a peripheral's ***BIT-LEVEL REGISTER DETAILS***. For example, any bits in a peripheral's register that correspond to undefined behavior will be specified in these header files.

HPS Peripheral Region Address Map

"…/hwlib/include/alt_fpga_manager.h"

- *ALT_STATUS_CODE alt_fpga_reset_assert(void);*
- *ALT_STATUS_CODE alt_fpga_configure(const void* cfg_buf, size_t cfg_buf_len);*

"…/hwlib/include/socal/alt_fpgamgr.h"

- */* The width in bits of the ALT_FPGAMGR_CTL_EN register field. */*
- *#define ALT_FPGAMGR_CTL_EN_WIDTH 1*
- */* The mask used to set the ALT_FPGAMGR_CTL_EN register field value. */*
- *#define ALT_FPGAMGR_CTL_EN_SET_MSK 0x00000001*
- */* The mask used to clear the ALT_FPGAMGR_CTL_EN register field value. */*
- *#define ALT_FPGAMGR_CTL_EN_CLR_MSK 0xfffffffe*

HPS Peripheral Region Address Map

An *important* header file is "…/hwlib/include/socal/hps.h". It contains the HPS component's full ***REGISTER MAP***

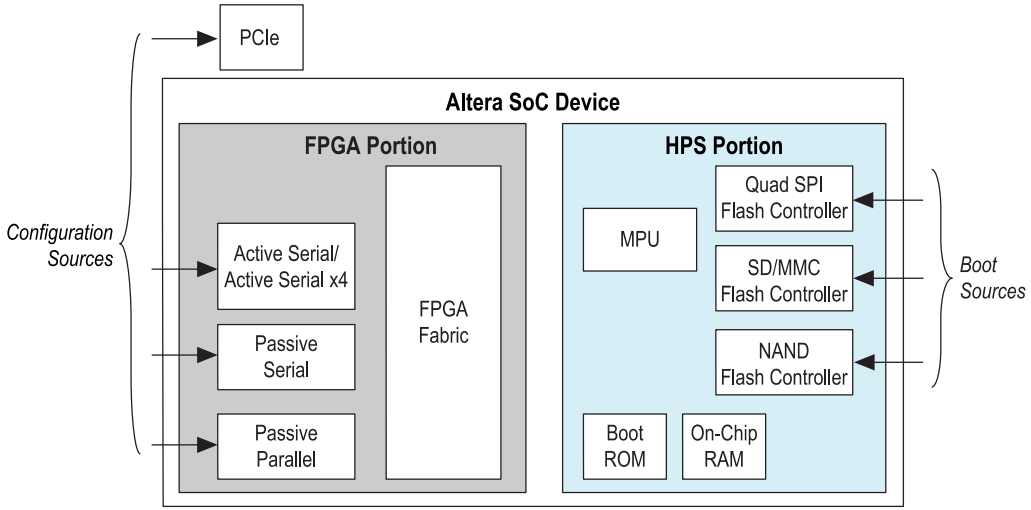
*Note that HWLIB can only be **DIRECTLY** used in a bare-metal application, or in a Linux device driver, as it directly references physical addresses.*

The linux kernel would terminate a user process that tries to access any of these addresses.

HPS Boot and FPGA Configuration

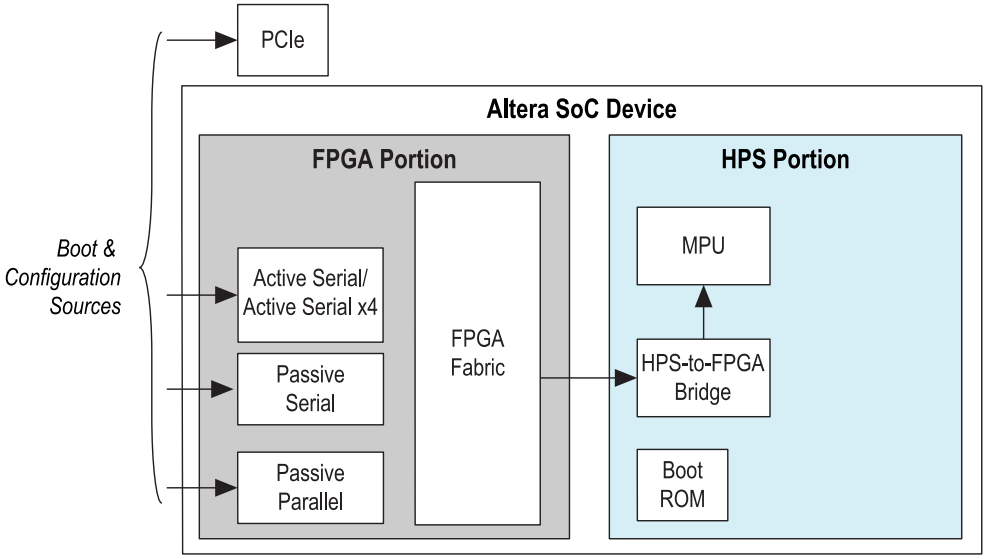
The processor can boot from the following sources:

- **NAND** flash memory through the NAND flash controller
- **SD/MMC** flash memory through the SD/MMC flash controller (***DE1-SOC board)***
- **SPI** and **QSPI** flash memory through the QSPI flash controller using *Slave Select 0*
- **FPGA** fabric on-chip memory



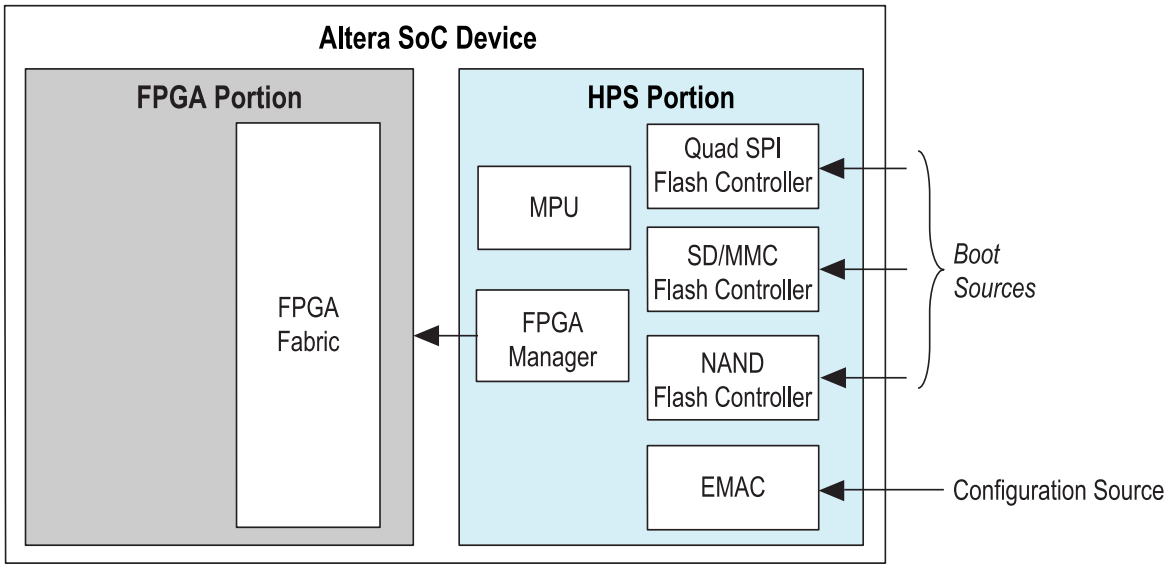
Independent FPGA Configuration and HPS Booting

HPS Boot and FPGA Configuration



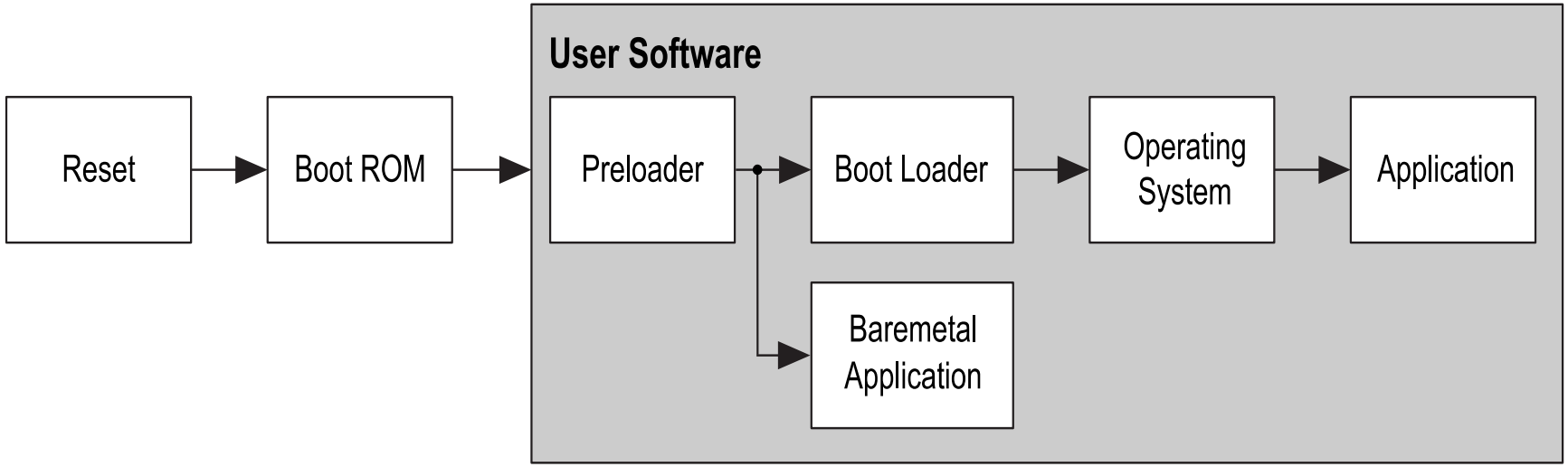
FPGA Configuration before HPS Booting (HPS boots from FPGA)

HPS Boot and FPGA Configuration



HPS Boots and Performs FPGA Configuration

HPS Boot Flows



HPS Boot Flows

- *Although the DE1-SoC has a **DUAL**-processor HPS:*
  - *CPU1 is under reset, and the boot flow only executes on CPU0.*
  - *To use both processors of the DE1-SoC, then **USER SOFTWARE** executing on CPU0 is responsible for releasing CPU1 from reset.*

HPS Boot Flows

The preloader typically performs the following actions:

- Initialize the SDRAM interface
- Configure the HPS I/O through the scan manager
- Configure pin multiplexing through the system manager
- Configure HPS clocks through the clock manager
- Initialize the flash controller (NAND, SD/MMC, QSPI) that contains the next stage boot software
- Load the next boot software into the SDRAM and pass control to it
- The preloader does ***NOT*** release CPU1 from reset.

Cyclone V, possible configurations

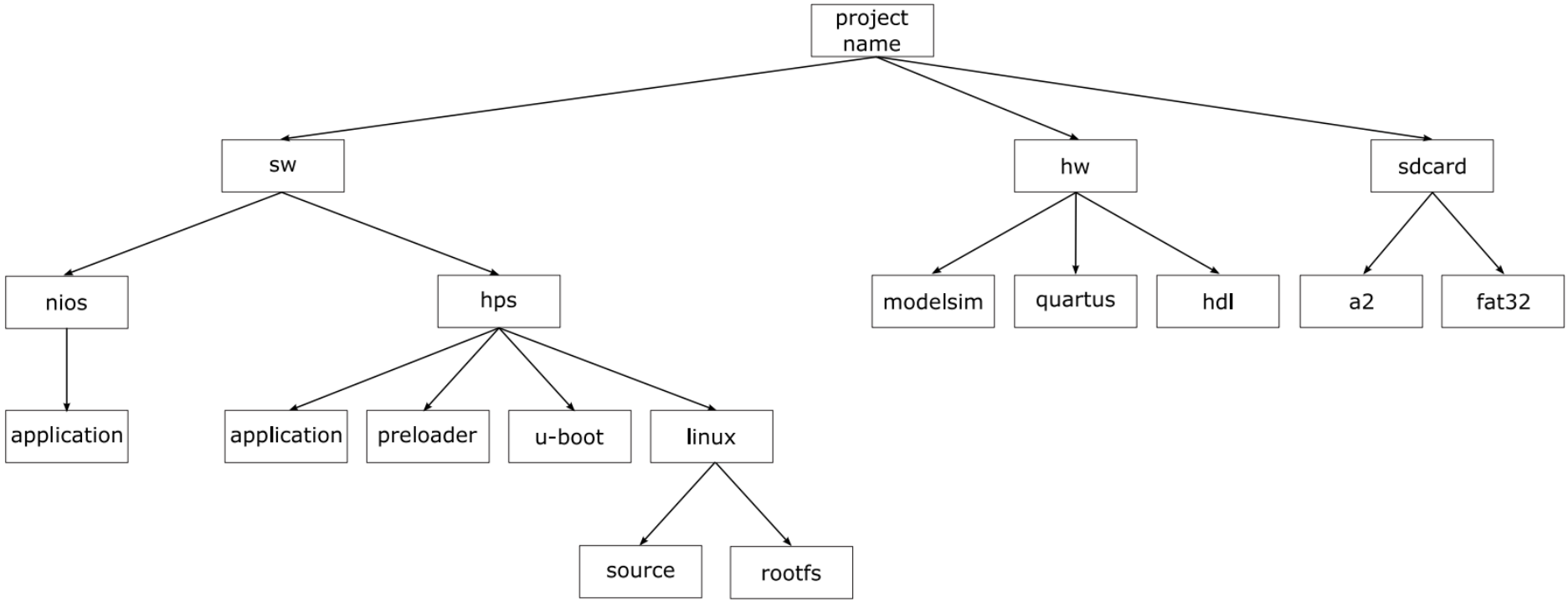
the Cyclone V SoC can work in 3 different configurations:

- FPGA-only
- HPS-only
- HPS & FPGA

- For HPS applications:
  - **Bare-metal** Application without OS
  - Application over an Operating System as **Linux**

Cyclone V, project structure

To help in the development process, suggested project structure:

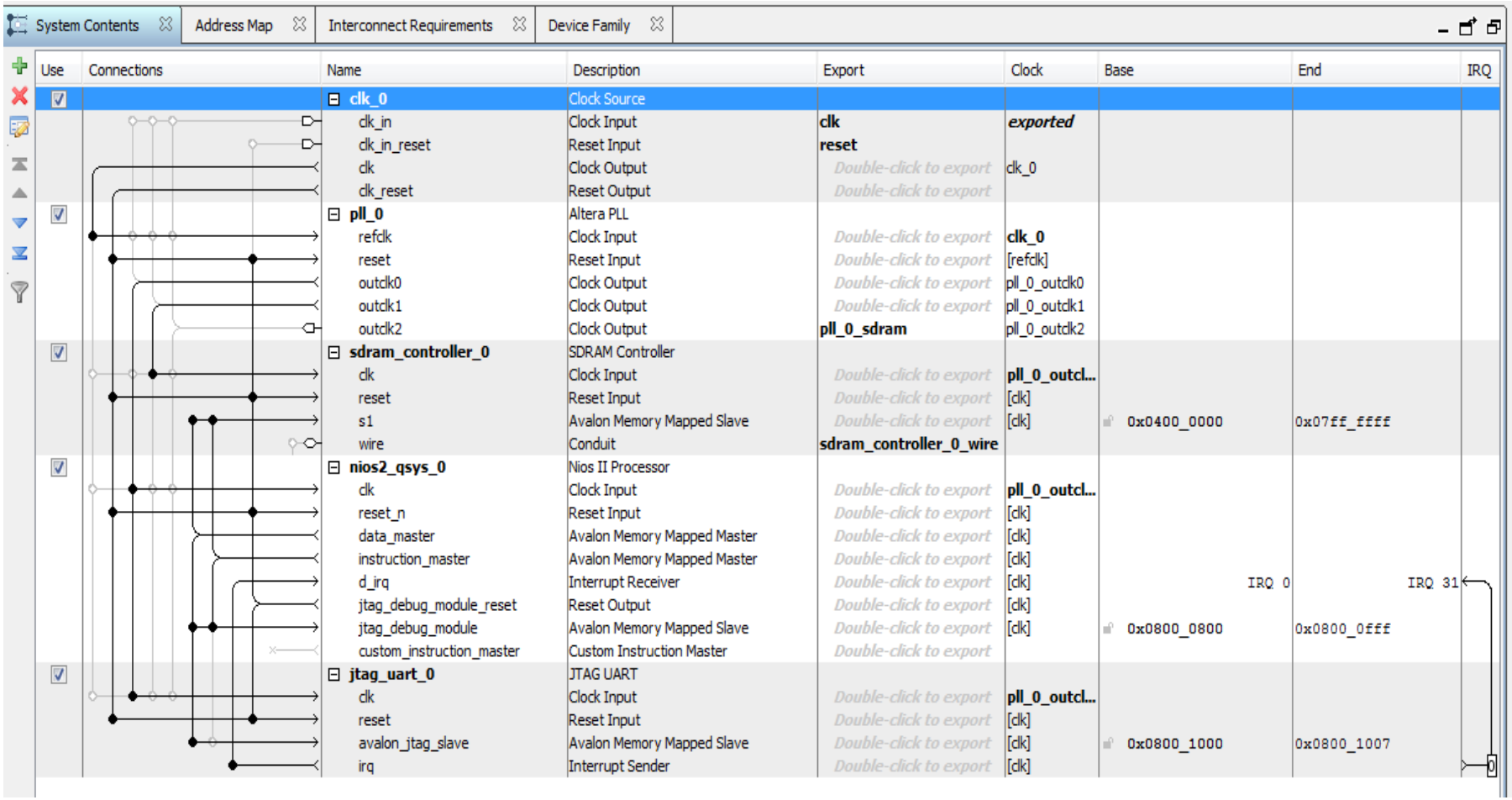


Cyclone V, FPGA development process

For the FPGA part, it's the same as for the others FPGA, **Quartus II** and **Qsys** tools

- NIOS II processor
- SDRAM Ctrl as softcore module
- Programmable Interface on Avalon Bus
- PLL for Clk and external SDRAM Clk

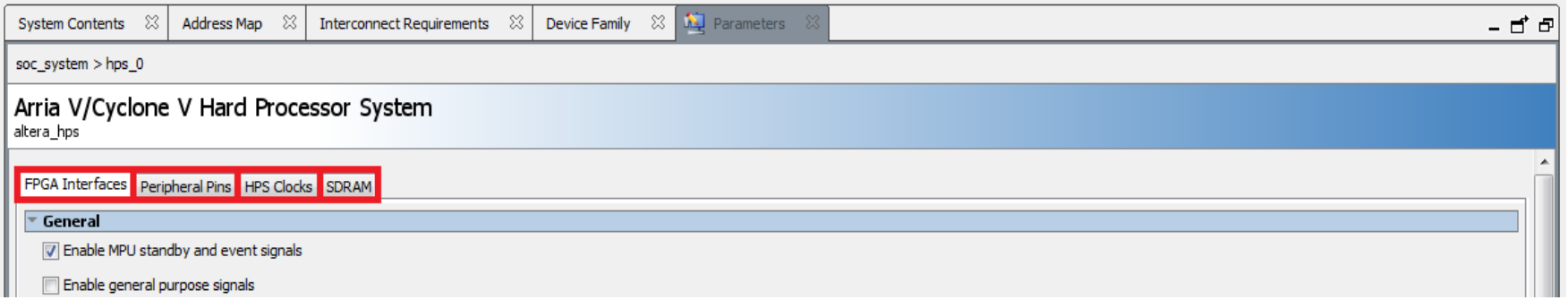
Cyclone V, FPGA development process



Cyclone V, HPS development process

For the HPS part, it's the same as for the others FPGA, **Quartus II** and **Qsys** tools

- HPS configuration with Qsys
- I/O pins association



Cyclone V, HPS I/O selection

Ex.: **HPS_KEY** & **HPS_LED**

From schematics:

In Qsys selection for the specifics pins:

TRACE_D4	CAN1.RX (Set0)	SPIS1.CLK (Set0)	TRACE.D4 (Set0)	GPIO53	LOANIO53
TRACE_D5	CAN1.TX (Set0)	SPIS1.MOSI (Set0)	TRACE.D5 (Set0)	GPIO54	LOANIO54

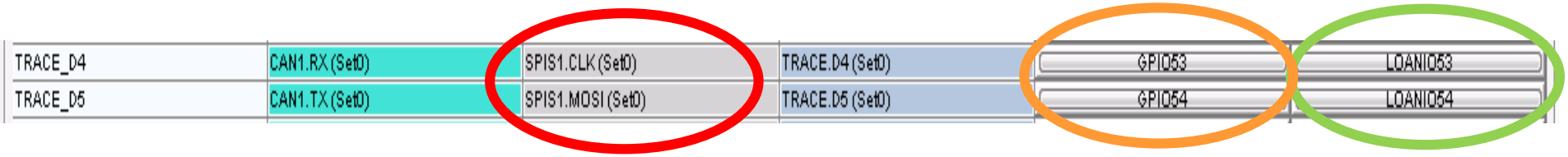
Mode GPIO:

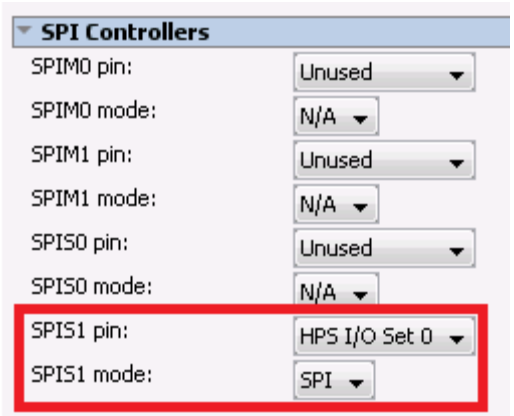
TRACE_D3	I2C1.SCL (Set0)	SPIS0.SS0 (Set0)	TRACE.D3 (Set0)	GPIO52	LOANIO52
TRACE_D4	CAN1.RX (Set0)	SPIS1.CLK (Set0)	TRACE.D4 (Set0)	**GPIO53**	LOANIO53
TRACE_D5	CAN1.TX (Set0)	SPIS1.MOSI (Set0)	TRACE.D5 (Set0)	**GPIO54**	LOANIO54
TRACE_D6	I2C0.SDA (Set0)	SPIS1.SS0 (Set0)	TRACE.D6 (Set0)	GPIO55	LOANIO55

Cyclone V, HPS I/O selection

Ex.: **HPS_KEY** & **HPS_LED**

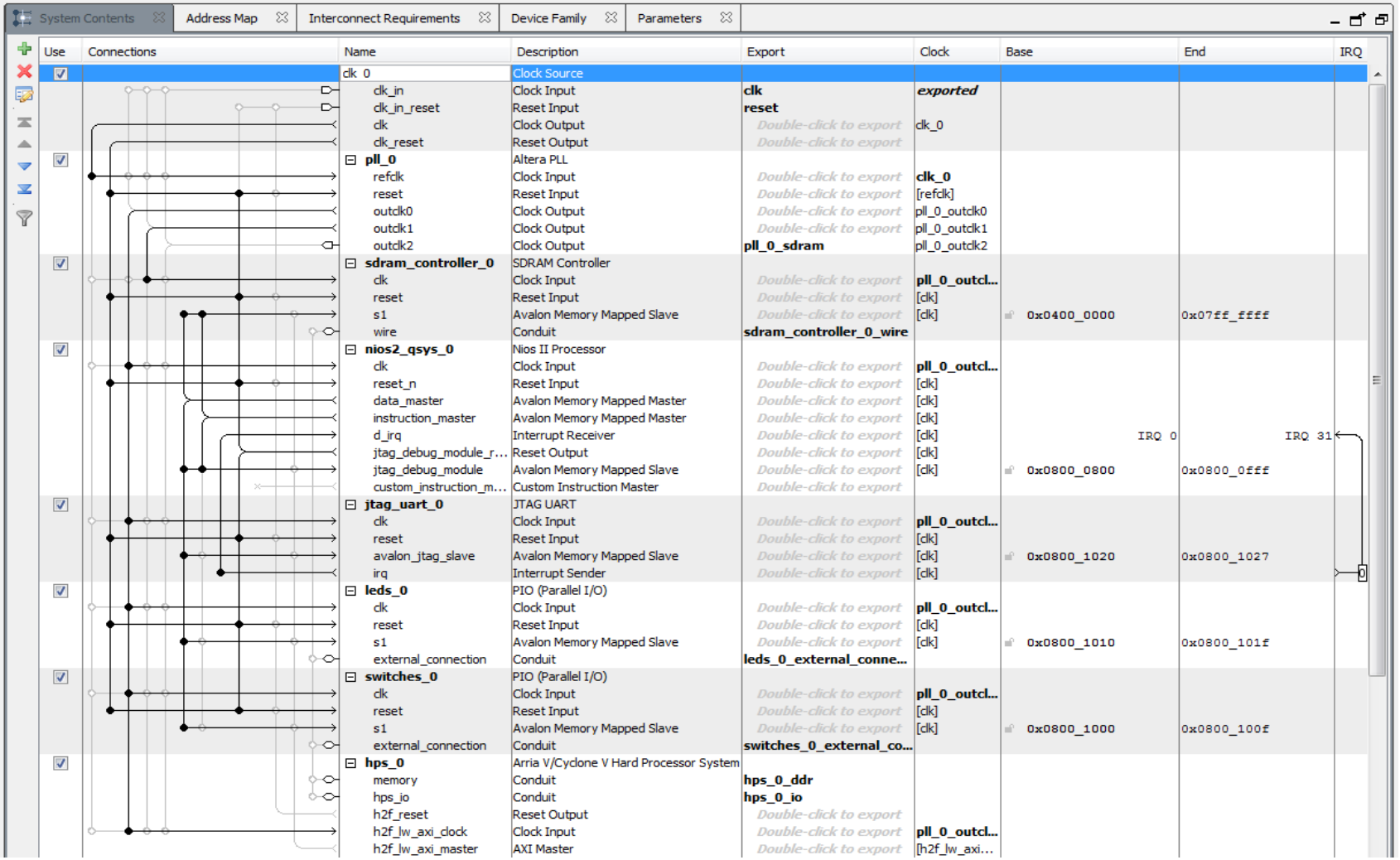
- ***GPIO****XY*: Configures the pin to be connected to the ***HPS' GPIO*** peripheral.



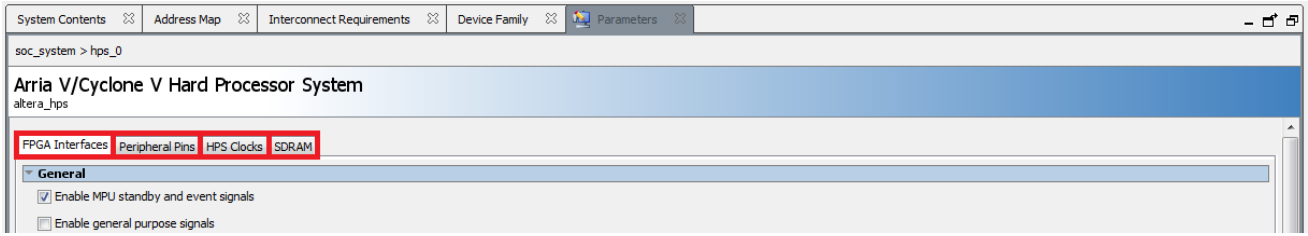


- ***LOANIO****XY*: Configures the pin to be connected to the ***FPGA*** fabric. This pin can be exported from *Qsys* to be used by the FPGA.

Cyclone V, HPS – FPGA development process

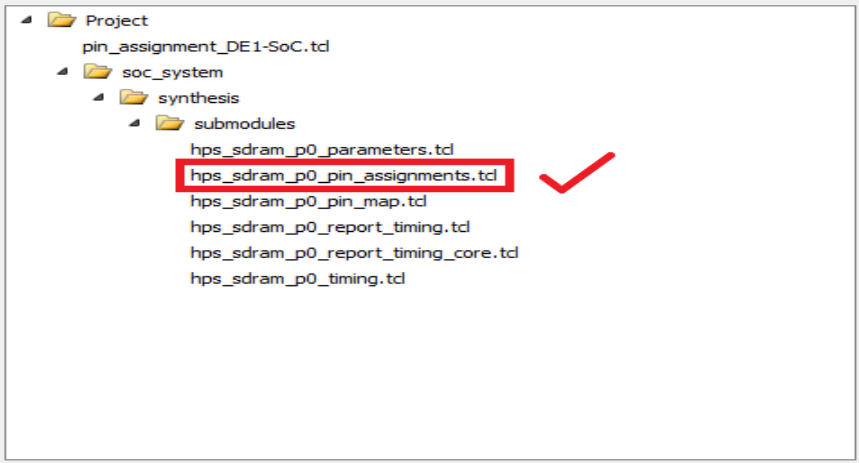


Cyclone V, HPS – DDR3 configuration



SDRAM Tab:

- In fact DDR3 memories
- Put the right parameters
- Assign the pins with tcl file



Cyclone V, Compilation

- Compilation
- Download through JTAG
- NIOS-SBT for NIOS programming
- ARM-DS-5 for ARM programming

Exercice → Follow the tutorial

SOC-FPGA Design Guide