



AARHUS  
UNIVERSITY  
SCHOOL OF ENGINEERING

PETER HØGH MIKKELSEN  
5. JANUARY, 2017

# SOPC INTRODUCTION

---

# INTRODUCING SOPC

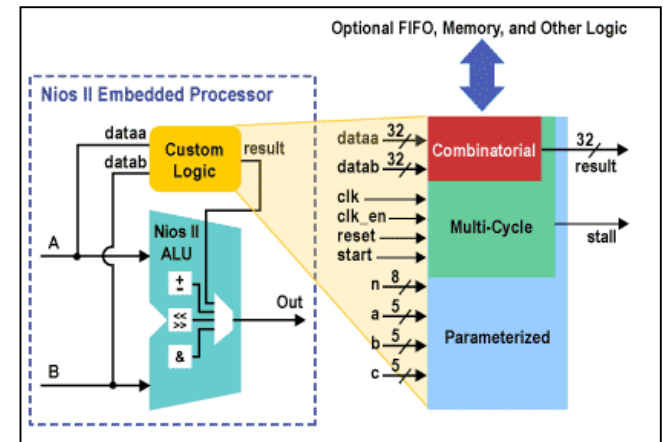
- › System on Programmable Chips comprises of FPGA + CPU
- › *CPU is good for*
  - › *Control logic, Managing network interfaces, UI*
- › *FPGA is good for*
  - › *Massive parallel data processing, custom interfaces, low latency interfaces*
- › Different setups for different applications
  - › *Hard- / Firm- / Soft Core setups*

# “HARD CORE” PROCESSOR

- › Processor implemented in silicon with an FPGA on the same die or bundled.
- › **Altera:**
  - › Cyclone V SoC: ARM Cortex-A9 (1-2x @ 900MHz)
  - › Stratix 10 SoC: ARM Cortex-A53 (4x @ 1500MHz)
- › **Xilinx:**
  - › Zynq 7000 : ARM Cortex-A9 (1-2x @ 900MHz)
  - › Ultrascale+ : ARM Cortex-A53 (4x @ 1500MHz)
- › **Actel:**
  - › ProASIC3 : ARM Cortex-M1 (1x @ 200MHz)
- › *Trend is hard core for high performance + low power*

# FIRM CORE PROCESSOR

- › Processor created from optimized programmable blocks and structures in an FPGA
- › Altera: *Nios II* and Xilinx: *MicroBlaze*
- › Very feature-rich and flexible
- › Can change following things compile time:
  - › ALU functionality,
  - › Number and types of peripherals,
  - › Memory width, and address space
  - › Size of instruction/data cache
  - › Level of debug logic
  - › Custom Instructions
  - › Hardware acceleration C2H
- › *Lower performance than Hard Cores*



# SOFT CORE PROCESSORS

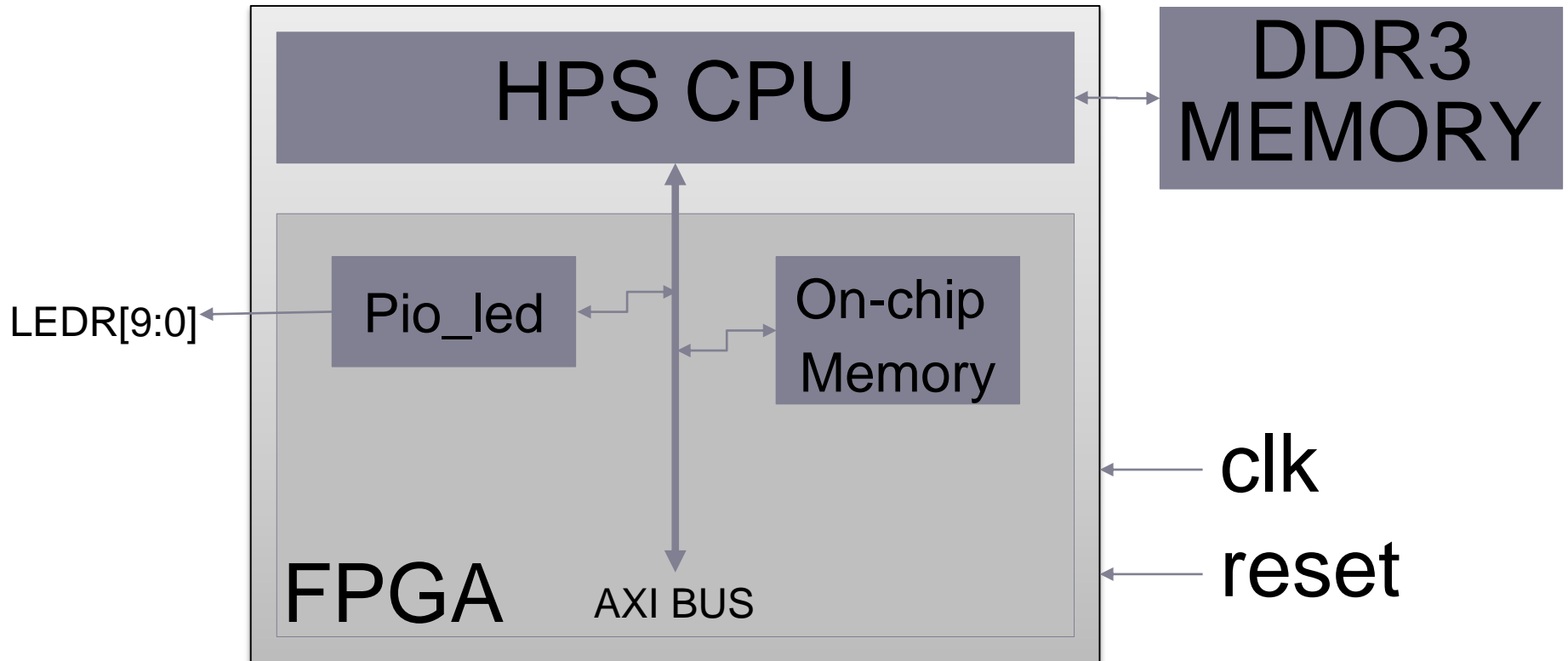
- › Processor purely based on general IP, can fit into most FPGAs
- › ESA: Leon II (MIPS based)
- › Opencores.org:
  - › 8051, MSP430, AVR..... And many more
- › Not architecture dependent
- › Higher power and resource consumption.
- › Software toolchain and support is not very mature
- › *Used for low complexity, low-cost high-volume projects that will evolve into asics*

# SOPC TOOLS

- › A specific EDA\* tool from every vendor to specify the SOPC system.
- › The EDA generates a synthesizable HDL-component for the softcore. Source code is often scrambled – but we got a licenses that unlocks the code.
- › We can instantiate the HDL-component along with user specific HDL code.
- › The Vendor normally supplies C/C++ tool chain for the Softcore.

# SOC SYSTEM OVERVIEW

## System On Chip



# CLOCKS & RESET

System: soc\_system Path: pio\_led

Use	Connections	Name	Description	Export	Clock	Base	End	IRQ	Tags
<input checked="" type="checkbox"/>		<b>clk_0</b>	Clock Source						
<input checked="" type="checkbox"/>		clk_in	Clock Input						
<input checked="" type="checkbox"/>		clk_in_reset	Reset Input						
<input checked="" type="checkbox"/>		clk	Clock Output						
<input checked="" type="checkbox"/>		clk_reset	Reset Output						
<input checked="" type="checkbox"/>		<b>hps_0</b>	Arria V/Cyclone V Hard Processor System						
<input checked="" type="checkbox"/>		f2h_cold_reset_req	Reset Input	hps_0_f2h_cold_reset_req			multiple		
<input checked="" type="checkbox"/>		f2h_debug_reset_req	Reset Input	hps_0_f2h_debug_reset_req					
<input checked="" type="checkbox"/>		f2h_warm_reset_req	Reset Input	hps_0_f2h_warm_reset_req					
<input checked="" type="checkbox"/>		h2f_reset	Reset Output	hps_0_h2f_reset					
<input checked="" type="checkbox"/>		h2f_axi_dock	Clock Input		clk_0	# 0x0000_0000	0xffff_ffff		
<input checked="" type="checkbox"/>		f2h_axi_dock	Clock Input		clk_0				
<input checked="" type="checkbox"/>		h2f_lw_axi_dock	Clock Input		clk_0				
<input checked="" type="checkbox"/>		<b>hps_only_master</b>	JTAG to Avalon Master Bridge						
<input checked="" type="checkbox"/>		clk	Clock Input		clk_0				
<input checked="" type="checkbox"/>		clk_reset	Reset Input						
<input checked="" type="checkbox"/>		master_reset	Reset Output						
<input checked="" type="checkbox"/>		<b>sysid_qsys</b>	System ID Peripheral						
<input checked="" type="checkbox"/>		clk	Clock Input		clk_0	# 0x0001_0000	0x0001_0007		
<input checked="" type="checkbox"/>		reset	Reset Input		[clk]				
<input checked="" type="checkbox"/>		<b>onchip_memory2_0</b>	On-Chip Memory (RAM or ROM)						
<input checked="" type="checkbox"/>		clk1	Clock Input		clk_0	# 0x0000_0000	0x0000_ffff		
<input checked="" type="checkbox"/>		reset1	Reset Input		[clk1]				
<input checked="" type="checkbox"/>		<b>jtag_uart</b>	JTAG UART						
<input checked="" type="checkbox"/>		clk	Clock Input		clk_0	# 0x0002_0000	0x0002_0007		
<input checked="" type="checkbox"/>		reset	Reset Input		[clk]				
<input checked="" type="checkbox"/>		<b>fpga_only_master</b>	JTAG to Avalon Master Bridge						
<input checked="" type="checkbox"/>		clk	Clock Input		clk_0				
<input checked="" type="checkbox"/>		clk_reset	Reset Input						
<input checked="" type="checkbox"/>		master_reset	Reset Output						
<input checked="" type="checkbox"/>		<b>intr_capturer_0</b>	INTR Capturer						
<input checked="" type="checkbox"/>		clock	Clock Input		clk_0	# 0x0003_0000	0x0003_0007		
<input checked="" type="checkbox"/>		reset_sink	Reset Input		[clock]				
<input checked="" type="checkbox"/>		<b>pio_led</b>	PIO (Parallel I/O)						
<input checked="" type="checkbox"/>		clk	Clock Input		clk_0	# 0x0000_0000	0x0000_000f		
<input checked="" type="checkbox"/>		reset	Reset Input		[clk]				
<input checked="" type="checkbox"/>		<b>mm_bus_seven_seg...</b>	mm_bus_seven_seg_four_digit						
<input checked="" type="checkbox"/>		clockreset	Clock Input		clk_0	# 0x0001_0200	0x0001_03ff		
<input checked="" type="checkbox"/>		clockreset_reset	Reset Input		[clockreset]				

Current filter: Clock and Reset Interfaces

## Qsys Clock/Reset View



# BUSSES - AXI (64/128-BIT)

System Contents Interconnect Requirements

System: soc\_system Path: hps\_0.h2f\_axi\_master

Use	Connections	Name	Description	Export	Clock	Base	End	IRQ	Tags
<input checked="" type="checkbox"/>		<b>hps_0</b>	Arria V/Cyclone V Hard Processor System						
		f2h_stm_hw_events	Conduit	<b>hps_0_f2h_stm_hw_events</b>					
		memory	Conduit	<b>memory</b>					
		hps_io	Conduit	<b>hps_0_hps_io</b>					
<input checked="" type="checkbox"/>		<b>h2f_axi_master</b>	AXI Master	<i>Double-click to export</i>	<b>clk_0</b>				
		f2h_axi_slave	AXI Slave	<i>Double-click to export</i>	<b>clk_0</b>	0x0000_0000	0xffff_ffff		
		h2f_hw_axi_master	AXI Master	<i>Double-click to export</i>	<b>clk_0</b>				
		f2h_irq0	Interrupt Receiver	<i>Double-click to export</i>				IRQ 0	
		f2h_irq1	Interrupt Receiver	<i>Double-click to export</i>				IRQ 0	
<input checked="" type="checkbox"/>		<b>hps_only_master</b>	JTAG to Avalon Master Bridge	<i>Double-click to export</i>	<b>clk_0</b>				
		master	Avalon Memory Mapped Master	<i>Double-click to export</i>	<b>clk_0</b>				
<input checked="" type="checkbox"/>		<b>sysid_qsys</b>	System ID Peripheral	<i>Double-click to export</i>	<b>clk_0</b>				
		control_slave	Avalon Memory Mapped Slave	<i>Double-click to export</i>	<b>clk_0</b>	0x0001_0000	0x0001_0007		
<input checked="" type="checkbox"/>		<b>onchip_memory2_0</b>	On-Chip Memory (RAM or ROM)	<i>Double-click to export</i>	<b>clk_0</b>				
		s1	Avalon Memory Mapped Slave	<i>Double-click to export</i>	<b>clk_0</b>	0x0000_0000	0x0000_ffff		
<input checked="" type="checkbox"/>		<b>jtag_uart</b>	JTAG UART	<i>Double-click to export</i>	<b>clk_0</b>				
		avalon_jtag_slave	Avalon Memory Mapped Slave	<i>Double-click to export</i>	<b>clk_0</b>	0x0002_0000	0x0002_0007		
		irq	Interrupt Sender	<i>Double-click to export</i>	<b>clk_0</b>				
<input checked="" type="checkbox"/>		<b>fpga_only_master</b>	JTAG to Avalon Master Bridge	<i>Double-click to export</i>	<b>clk_0</b>				
		master	Avalon Memory Mapped Master	<i>Double-click to export</i>	<b>clk_0</b>				
<input checked="" type="checkbox"/>		<b>intr_capturer_0</b>	intr_capturer	<i>Double-click to export</i>	<b>clk_0</b>				
		avalon_slave_0	Avalon Memory Mapped Slave	<i>Double-click to export</i>	<b>clk_0</b>	0x0003_0000	0x0003_0007		
		interrupt_receiver	Interrupt Receiver	<i>Double-click to export</i>	<b>clk_0</b>			IRQ 0	
<input checked="" type="checkbox"/>		<b>pio_led</b>	PIO (Parallel I/O)	<i>Double-click to export</i>	<b>clk_0</b>				
		s1	Avalon Memory Mapped Slave	<i>Double-click to export</i>	<b>clk_0</b>	0x0000_0000	0x0000_000f		
		external_connection	Conduit	<b>pio_led_external_connection</b>	<b>clk_0</b>				
<input checked="" type="checkbox"/>		<b>mm_bus_seven_seg...</b>	mm_bus_seven_seg_four_digit	<i>Double-click to export</i>	<b>clk_0</b>				
		s1	Avalon Memory Mapped Slave	<i>Double-click to export</i>	<b>clk_0</b>	0x0001_0200	0x0001_03ff		
		hex	Conduit	<b>hex</b>	<b>clk_0</b>				

Current filter: Hide Clocks and Resets

Qsys

# BUSSES – AXI LIGHT (32-BIT)

System Contents Interconnect Requirements

System: soc\_system Path: hps\_0.h2f\_lw\_axi\_master

Use	Connections	Name	Description	Export	Clock	Base	End	IRQ	T
<input checked="" type="checkbox"/>		<b>hps_0</b>	Arria V/Cyclone V Hard Processor System						
		f2h_stm_hw_events	Conduit	<b>hps_0_f2h_stm_hw_events</b>					
		memory	Conduit	<b>memory</b>					
		hps_io	Conduit	<b>hps_0_hps_io</b>					
		h2f_axi_master	AXI Master	<i>Double-click to export</i>	clk_0				
		f2h_axi_slave	AXI Slave	<i>Double-click to export</i>	clk_0	0x0000_0000	0xffff_ffff		
		<b>h2f_lw_axi_master</b>	AXI Master	<i>Double-click to export</i>	clk_0				
		f2h_irq0	Interrupt Receiver	<i>Double-click to export</i>				IRQ 0	IRQ 31
		f2h_irq1	Interrupt Receiver	<i>Double-click to export</i>				IRQ 0	IRQ 31
<input checked="" type="checkbox"/>		<b>hps_only_master</b>	JTAG to Avalon Master Bridge	<i>Double-click to export</i>	clk_0				
		master	Avalon Memory Mapped Master	<i>Double-click to export</i>	clk_0				
<input checked="" type="checkbox"/>		<b>sysid_qsys</b>	System ID Peripheral	<i>Double-click to export</i>	[clk]				
		control_slave	Avalon Memory Mapped Slave	<i>Double-click to export</i>	clk_0	0x0001_0000	0x0001_0007		
<input checked="" type="checkbox"/>		<b>onchip_memory2_0</b>	On-Chip Memory (RAM or ROM)	<i>Double-click to export</i>	[clk1]				
		s1	Avalon Memory Mapped Slave	<i>Double-click to export</i>	clk_0	0x0000_0000	0x0000_ffff		
<input checked="" type="checkbox"/>		<b>jtag_uart</b>	JTAG UART	<i>Double-click to export</i>	[clk]				
		avalon_jtag_slave	Avalon Memory Mapped Slave	<i>Double-click to export</i>	clk_0	0x0002_0000	0x0002_0007		
		irq	Interrupt Sender	<i>Double-click to export</i>	[clk]				
<input checked="" type="checkbox"/>		<b>fpga_only_master</b>	JTAG to Avalon Master Bridge	<i>Double-click to export</i>	clk_0				
		master	Avalon Memory Mapped Master	<i>Double-click to export</i>	clk_0				
<input checked="" type="checkbox"/>		<b>intr_capturer_0</b>	intr_capturer	<i>Double-click to export</i>	[clock]				
		avalon_slave_0	Avalon Memory Mapped Slave	<i>Double-click to export</i>	clk_0	0x0003_0000	0x0003_0007		
		interrupt_receiver	Interrupt Receiver	<i>Double-click to export</i>	[clock]			IRQ 0	IRQ 31
<input checked="" type="checkbox"/>		<b>pio_led</b>	PIO (Parallel I/O)	<i>Double-click to export</i>	[clk]				
		s1	Avalon Memory Mapped Slave	<i>Double-click to export</i>	clk_0	0x0000_0000	0x0000_000f		
		external_connection	Conduit	<b>pio_led_external_connection</b>					
<input checked="" type="checkbox"/>		<b>mm_bus_seven_seg...</b>	mm_bus_seven_seg_four_digit	<i>Double-click to export</i>	[clockreset]				
		s1	Avalon Memory Mapped Slave	<i>Double-click to export</i>	clk_0	0x0001_0200	0x0001_03ff		
		hex	Conduit	<b>hex</b>	[clockreset]				

Current filter: Hide Clocks and Resets

# BUSSES – ADDRESSING

System: soc\_system Path: clk\_0

pio_led.s1	...	hps_0.h2f_lw_axi_master	hps_only_master.master
sysid_qsys.control_slave	--	0x0001_0000 - 0x0001_0007	
mm_bus_seven_seg_four_digit_0.s1		0x0001_0200 - 0x0001_03ff	
jtag_uart.avalon_jtag_slave	--	0x0002_0000 - 0x0002_0007	
hps_0.f2h_axi_slave			0x0000_0000 - 0xffff_ffff
intr_capturer_0.avalon_slave_0	--		
onchip_memory2_0.s1	--		

Generate HDL

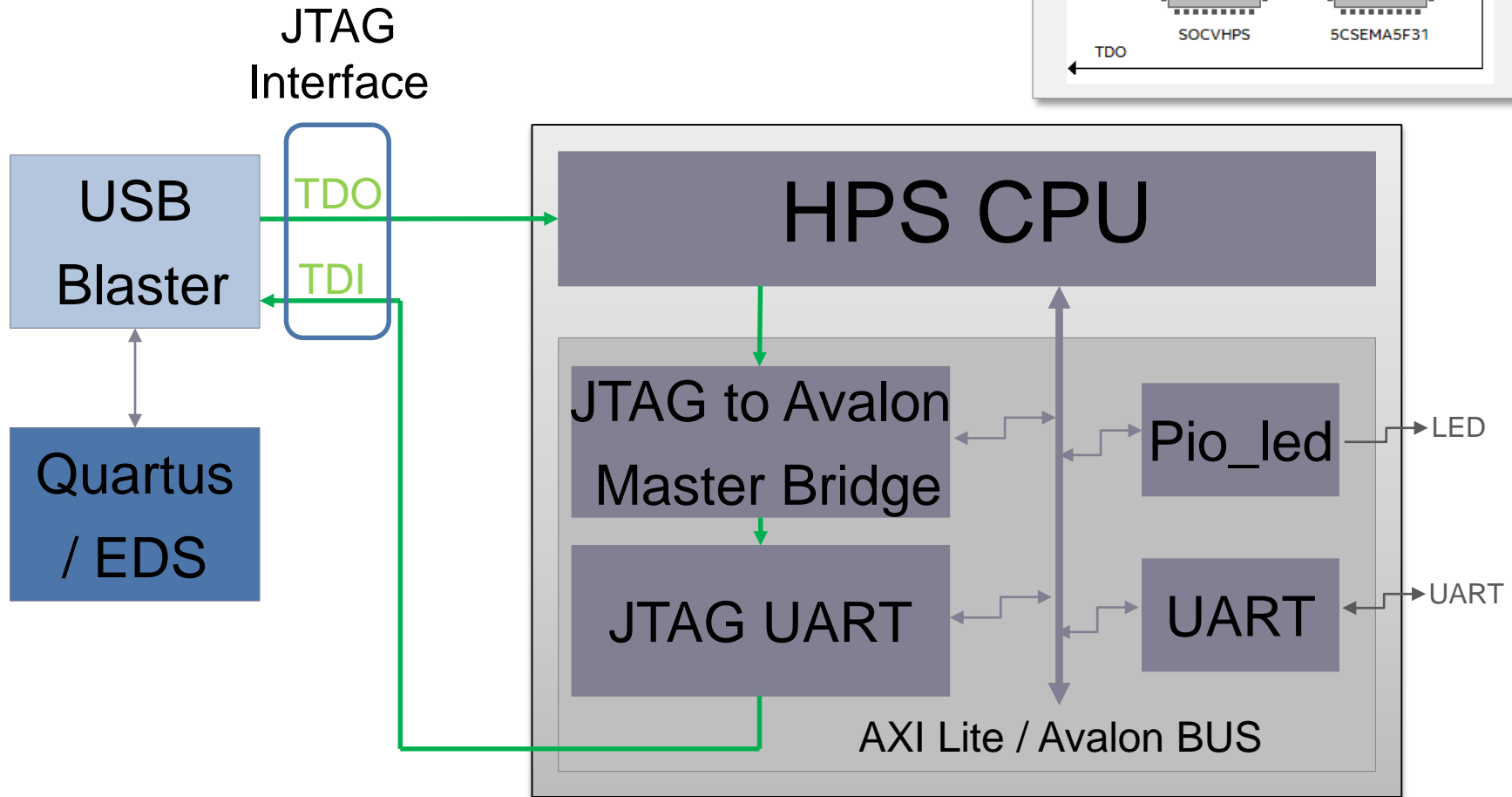
./generate\_hps\_qsys\_header.sh

```
//hps_0.h
#define PIO_LED_COMPONENT_TYPE
altera_avalon_pio
#define PIO_LED_COMPONENT_NAME pio_led
#define PIO_LED_BASE 0x0
#define PIO_LED_END 0xf
#define PIO_LED_SPAN 16
```

System: soc\_system Path: hps\_0.h2f\_lw\_axi\_master

Name	Description	Export	Clock	Base	End	IRQ
hps_0	Arria V/Cyclone V Hard Processor System					
f2h_stm_hw_events	Conduit	hps_0.f2h_stm_hw_events				
memory	Conduit	memory				
hps_io	Conduit	hps_0_hps_io				
h2f_axi_master	AXI Master	Double-click to export	clk_0	# 0x0000_0000	0xffff_ffff	
f2h_axi_slave	AXI Slave	Double-click to export	clk_0			
h2f_lw_axi_master	AXI Master	Double-click to export	clk_0			
f2h_irq0	Interrupt Receiver	Double-click to export				IRQ 0
f2h_irq1	Interrupt Receiver	Double-click to export				IRQ 0
hps_only_master	JTAG to Avalon Master Bridge	Double-click to export	clk_0			
sysid_qsys	System ID Peripheral	Double-click to export	clk_0	# 0x0001_0000	0x0001_0007	
control_slave	Avalon Memory Mapped Slave	Double-click to export	clk_0	# 0x0000_0000	0x0000_ffff	
onchip_memory2_0	On-Chip Memory (RAM or ROM)	Double-click to export	clk_0	# 0x0000_0000	0x0000_ffff	
s1	Avalon Memory Mapped Slave	Double-click to export	clk_0	# 0x0002_0000	0x0002_0007	
jtag_uart	JTAG UART	Double-click to export	clk_0	# 0x0002_0000	0x0002_0007	
avalon_jtag_slave	Avalon Memory Mapped Slave	Double-click to export	clk_0	# 0x0002_0000	0x0002_0007	
irq	Interrupt Sender	Double-click to export	clk_0			
fpqas_only_master	JTAG to Avalon Master Bridge	Double-click to export	clk_0			
avalon_slave_0	Avalon Memory Mapped Master	Double-click to export	clk_0	# 0x0003_0000	0x0003_0007	
intr_capturer_0	Interrupt Receiver	Double-click to export	clk_0	# 0x0003_0000	0x0003_0007	IRQ 0
intr_capturer	Avalon Memory Mapped Slave	Double-click to export	clk_0	# 0x0003_0000	0x0003_0007	IRQ 0
pio_led	PIO (Parallel I/O)	Double-click to export	clk_0	# 0x0000_0000	0x0000_000f	
s1	Avalon Memory Mapped Slave	Double-click to export	clk_0	# 0x0000_0000	0x0000_000f	
external_connection	Conduit	Double-click to export	clk_0	# 0x0001_0200	0x0001_03ff	
mm_bus_seven_seg_four_digit	Avalon Memory Mapped Slave	Double-click to export	clk_0	# 0x0001_0200	0x0001_03ff	
s1	hex	Double-click to export	clk_0	# 0x0001_0200	0x0001_03ff	
hex	hex	Double-click to export	clk_0	# 0x0001_0200	0x0001_03ff	

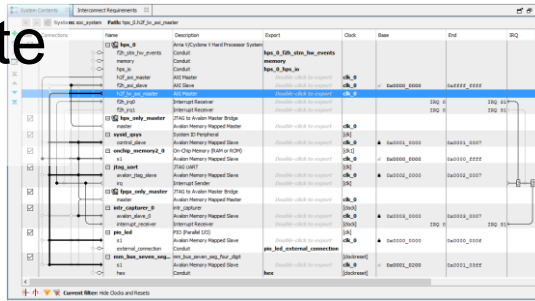
# JTAG



› JTAG is a shift-register interface for programming, test and debugging

# WORKFLOW

Generate  
HDL



Instantiate

```
-- main.vhd
begin
  u0 : component soc_system
    port map (
      clk_clk      => CLOCK_50,
      pio_led_export => LEDR, ...)
```

```
D22502+phm@D22502 /cygdrive/c/projects/dspc/d
$ ./generate_hps_qsys_header.sh
swinfo2header: Creating macro file 'hps_0.h'
```

```
//hps_0.h
#define PIO_LED_BASE 0x0
#define PIO_LED_END 0xf
#define PIO_LED_SPAN 16
```

```
D22502+phm@D22502 /cygdriv
$ arm-altera-eabi-gcc
```

```
D22502+phm@D22502 /cygdrive/c/pr
$ scp hello root@192.168.10.2: _
```

```
#include "hps_0.h"
int main() {
  vbase = mmap( NULL,...);
  led_addr=vbase\
+ ALT_LWFPGASLVS_OFST\
+ PIO_LED_BASE + ...;
  *led_addr = 0x55;
```

