SOC-FPGA design

Manual

Course Real Time Embedded Systems

LAP – IC – EPFL

Version 0.0 (Preliminary)

René Beuchat

1 Introduction

Development of embedded systems based on chip containing one or more microprocessor and hardcore peripherals well as FPGA part is becoming more and more important. This technology allows the the designer a lot of freedom and powerful capabilities. Classical design as with microcontrollers is emphasized with the full power of the FPGAs.

Mixed design are becoming a reality with evolution for the design of specific accelerator the improve a lot of algorithms as well as specific programmable interface with the external world.

Two main HDL (**H**ardware **D**esign **L**anguage) languages are available for the design of the FPGA part VHDL and Verilog. Some tools allow the automatic translation from C to HDL. New technologies emerge as OpenCL to have compatibility between high level design of software and implementation in the hardware as:

- Compilation for a processor or multicore processors
- Compilation for GPU (Graphical Processing Unit)
- Translation and compilation for FPGA, for the last one, PCIe interface is mandatory or another way of parameters passing between the main mandatory processor and the FPGA part is necessary

This guide assumes that the users know how to use QuartusII, NIOSII, Qsys and ModelSim-Altera.

The used board is the DE1-soc from terasic: http://de1-soc.terasic.com

2 DE1-soc board

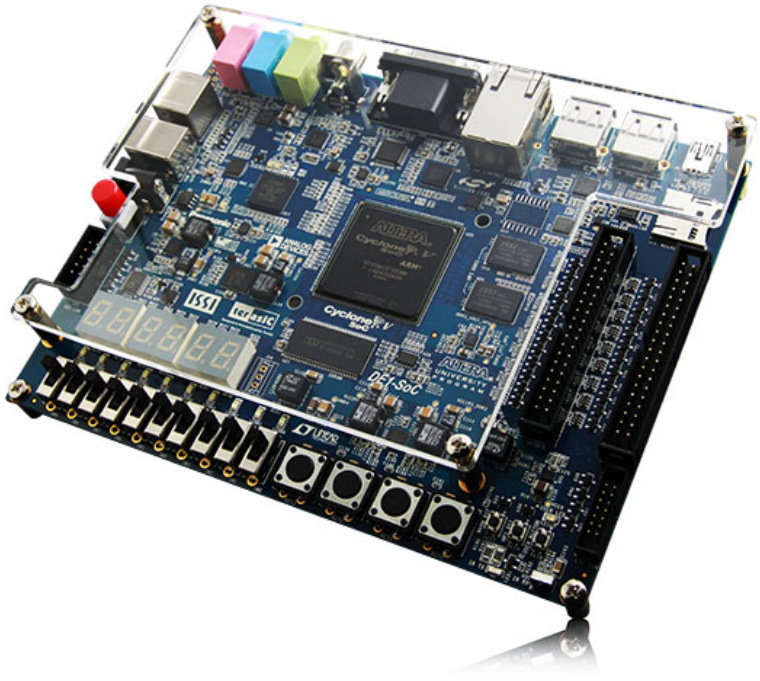


Fig. 1. de1-soc board from terasic

Characteristic of the board:

FPGA Device

- Cyclone V SoC **5CSEMA5F31C6** Device
- Dual-core **ARM Cortex-A9** (HPS)
- **85K** Programmable Logic Elements
- 4'450 Kbits embedded memory
- 6 Fractional PLLs
- Hard Memory Controllers

Configuration and Debug

- Quad Serial Configuration device – **EPCQ256** on FPGA
- On-Board **USB Blaster II** (Normal type B USB connector)

Memory Device

- **64MB** (32Mx16) SDRAM on FPGA
- **1GB** (2x256Mx16) DDR3 SDRAM on HPS
- **Micro SD** Card Socket on HPS

Communication

- Two Port USB 2.0 Host (ULPI interface with USB type A connector)
- USB to UART (micro USB type B connector)
- 10/100/1000 Ethernet
- PS/2 mouse/keyboard
- IR Emitter/Receiver

Connectors

- Two 40-pin Expansion Headers
- One 10-pin ADC Input Header
- One LTC connector (One Serial Peripheral Interface (SPI) Master ,one I2C and one GPIO interface)

Display

- 24-bit VGA DAC

Audio

- 24-bit CODEC, Line-in, line-out, and microphone-in jacks

Video Input

- TV Decoder (NTSC/PAL/SECAM) and TV-in connector

ADC

- Fast throughput rate: 1 MSPS
- Channel number: 8
- Resolution: 12 bits
- Analog input range : 0 ~ 2.5 V or 0 ~ 5V as selected via the RANGE bit in the control registe

Switches, Buttons and Indicators

- 4 User Keys (FPGA x4)
- 10 User switches (FPGA x10)
- 11 User LEDs (FPGA x10 ; HPS x 1)
- HPS Reset Buttons (HPS_RST_n and HPS_WARM_RST_n)
- Six 7-segment displays

Sensors

- G-Sensor on HPS

Power

- 12V DC input

Block Diagram of the DE1-SOC Board

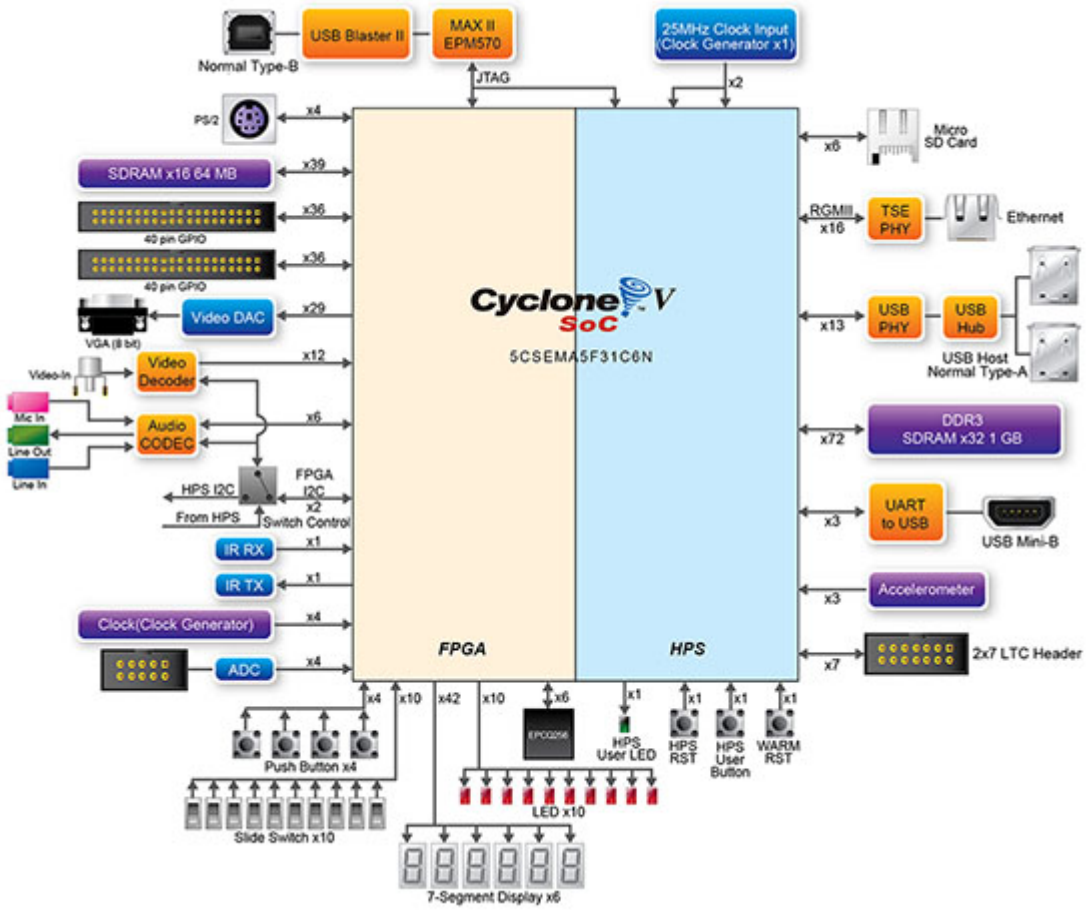


Fig. 2. Block Diagram of the DE1-SOC Board

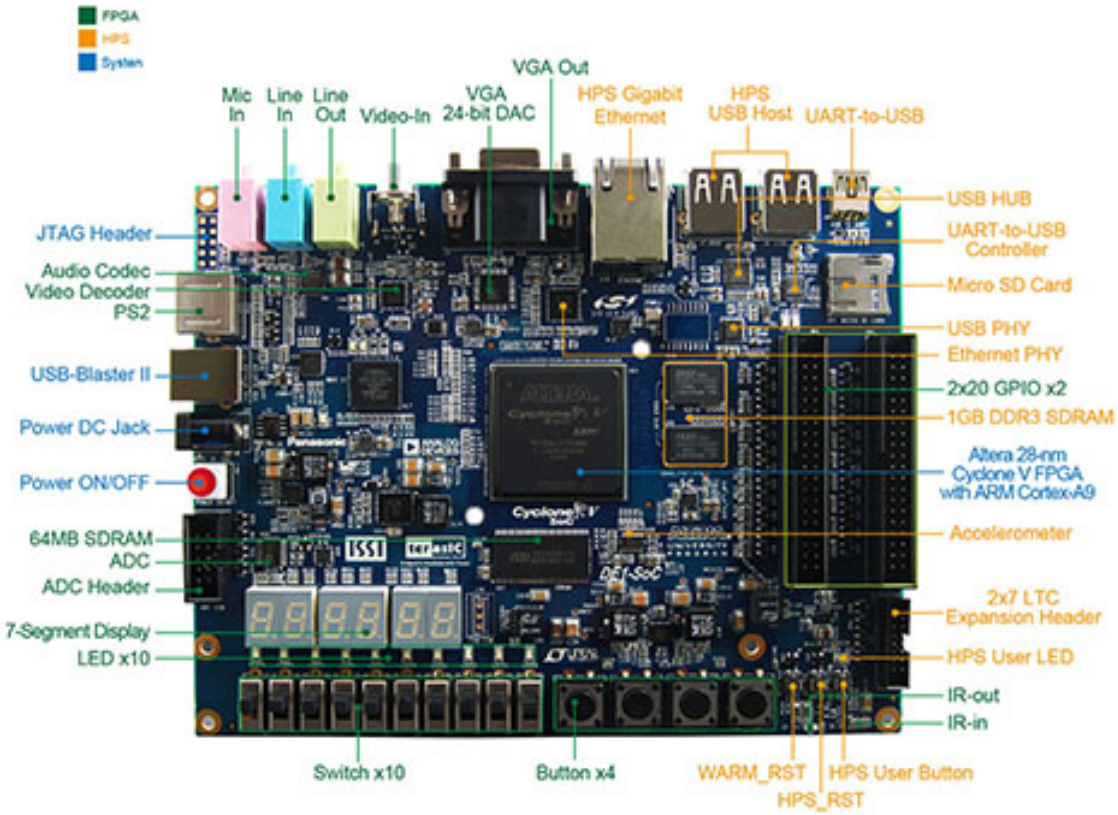


Fig. 3. Elements on the de1-soc boards
http://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=165&No=836&PartNo=3

- Green for FPGA part
- Orange for HPS part
- Blue for control

Manuals and resources are available at:

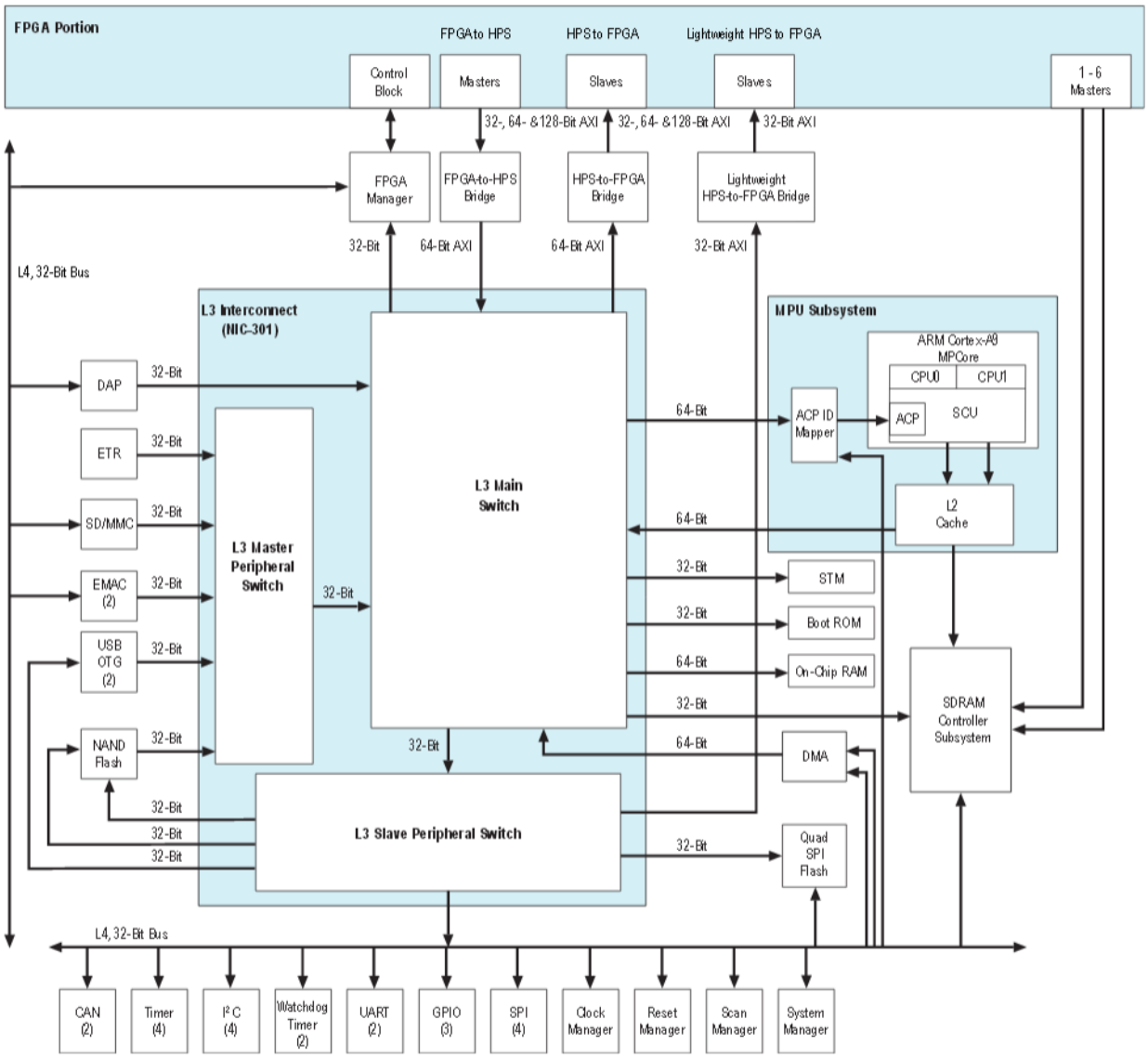
http://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=165&No=836&PartNo=4ParagrapheTexte

- Liste

3 SOC part test

3.1 HPS Architecture

To be able to program the ARM9's processors it is almost necessary to have the global view of the HPS architecture.



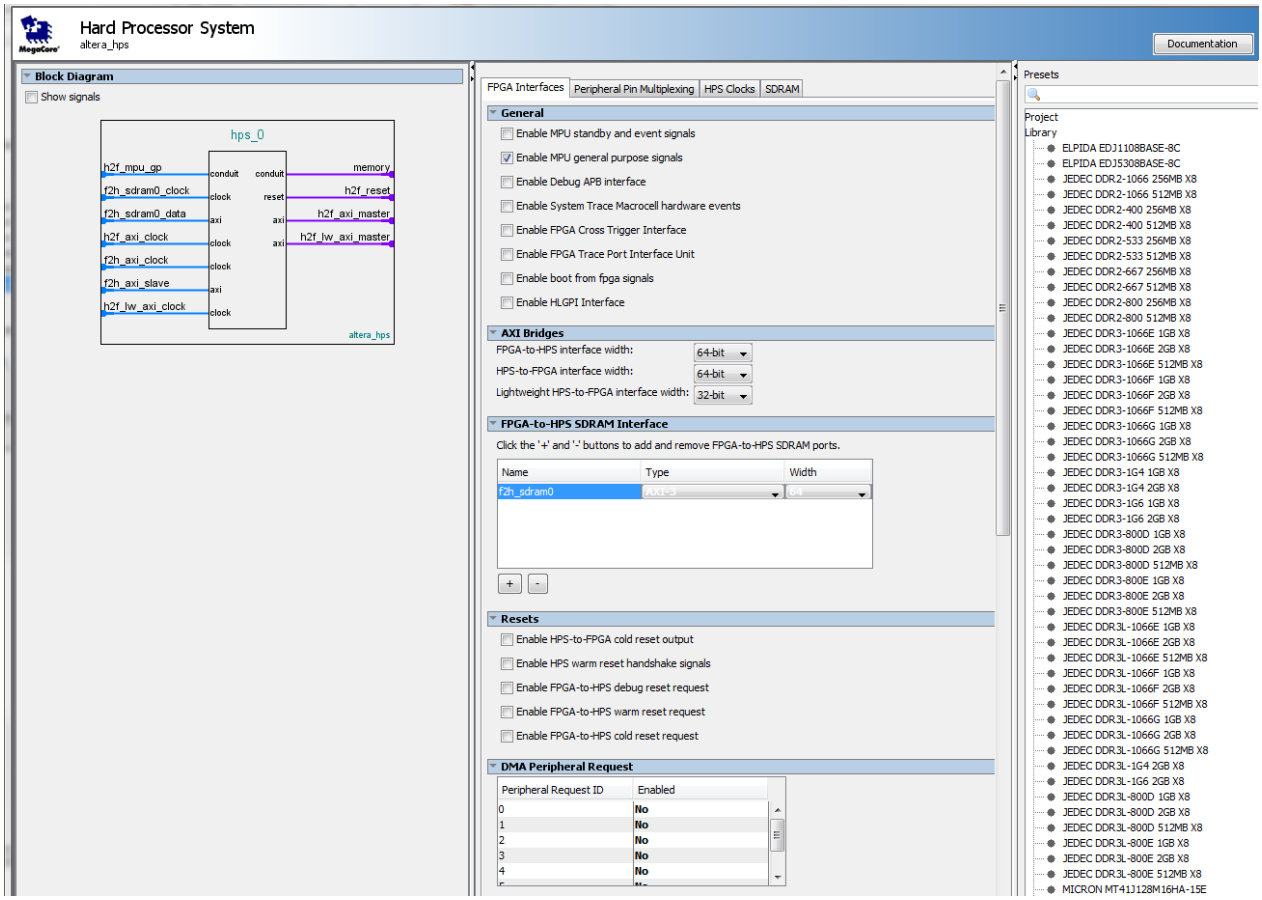
3.2 Hardware development

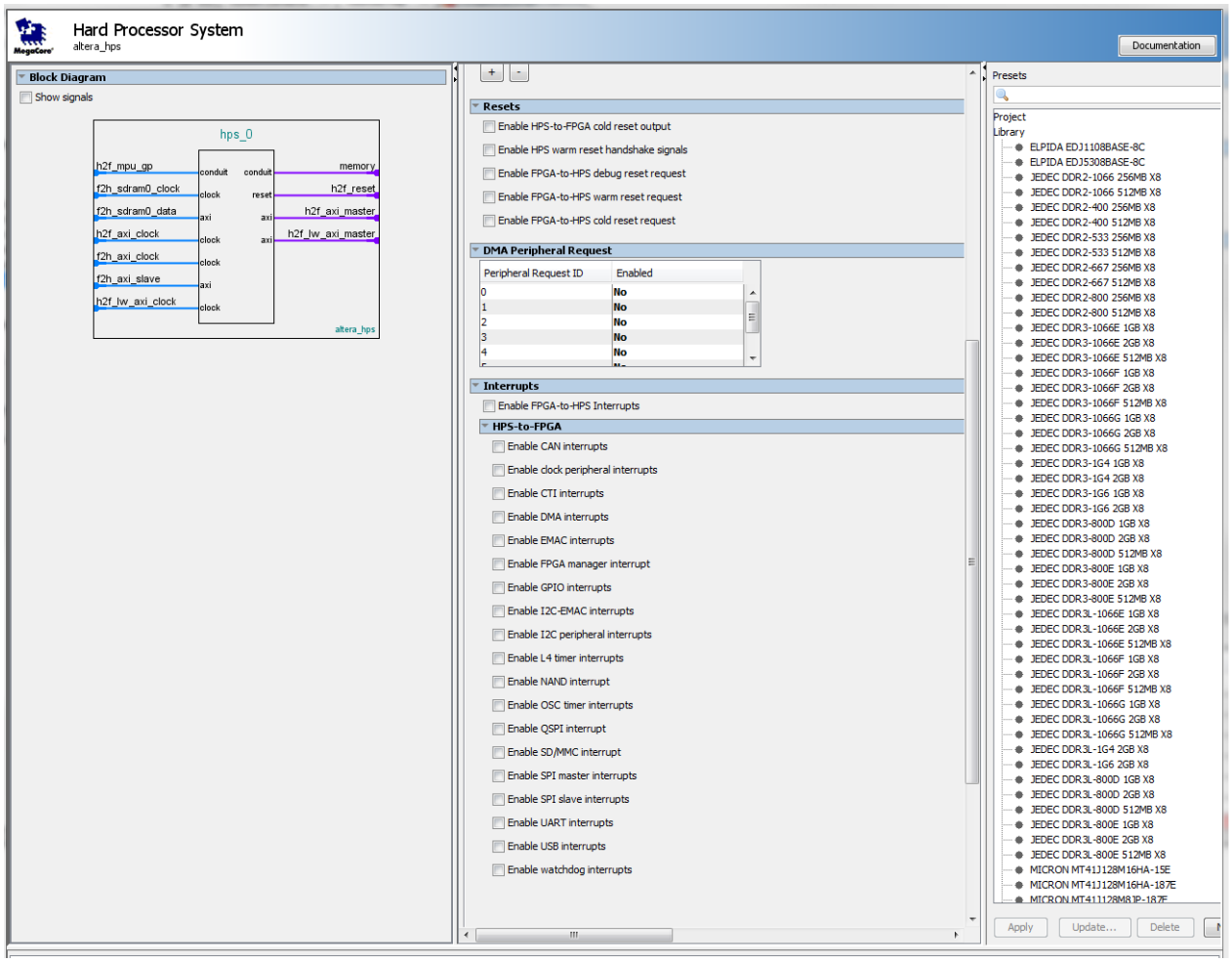
3.2.1 Qsys integration

Starting with **QuartusII** and after creating a project, select ***Tools → Qsys***

In **Qsys**, open ***Library → Embedded Processors → Hard Processor System*** the window with description of the parameters for the HPS is open.

The ***FPGA Interface*** tab allows the access from to the FPGA part with the HPS part.


Hard Processor System
altera_hps
Documentation
Block Diagram
Show signals
hps_0
h2f_mpu_gp
f2h_sdram0_clock
f2h_sdram0_data
h2f_axi_clock
f2h_axi_clock
f2h_axi_slave
h2f_lw_axi_clock
memory
h2f_reset
h2f_axi_master
h2f_lw_axi_master
altera_hps
FPGA Interfaces
Peripheral Pin Multiplexing
HPS Clocks
SDRAM
General
Enable MPU standby and event signals
Enable MPU general purpose signals
Enable Debug APB interface
Enable System Trace Macrocell hardware events
Enable FPGA Cross Trigger Interface
Enable FPGA Trace Port Interface Unit
Enable boot from fpga signals
Enable HLGPI Interface
AXI Bridges
FPGA-to-HPS interface width:
HPS-to-FPGA interface width:
Lightweight HPS-to-FPGA interface width:
64-bit
32-bit
FPGA-to-HPS SDRAM Interface
Click the '+' and '-' buttons to add and remove FPGA-to-HPS SDRAM ports.
Name
Type
Width
f2h_sdram0
Resets
Enable HPS-to-FPGA cold reset output
Enable HPS warm reset handshake signals
Enable FPGA-to-HPS debug reset request
Enable FPGA-to-HPS warm reset request
Enable FPGA-to-HPS cold reset request
DMA Peripheral Request
Peripheral Request ID
Enabled
Presets
Project
Library


Hard Processor System
altera_hps
Documentation
Block Diagram
Show signals
hps_0
h2f_mpu_gp
f2h_sdram0_clock
f2h_sdram0_data
h2f_axi_clock
f2h_axi_clock
f2h_axi_slave
h2f_lw_axi_clock
memory
h2f_reset
h2f_axi_master
h2f_lw_axi_master
altera_hps
Resets
Enable HPS-to-FPGA cold reset output
Enable HPS warm reset handshake signals
Enable FPGA-to-HPS debug reset request
Enable FPGA-to-HPS warm reset request
Enable FPGA-to-HPS cold reset request
DMA Peripheral Request
Peripheral Request ID
Enabled
Interrupts
Enable FPGA-to-HPS Interrupts
HPS-to-FPGA
Enable CAN interrupts
Enable clock peripheral interrupts
Enable CTI interrupts
Enable DMA interrupts
Enable EMAC interrupts
Enable FPGA manager interrupt
Enable GPIO interrupts
Enable I2C-EMAC interrupts
Enable I2C peripheral interrupts
Enable L4 timer interrupts
Enable NAND interrupt
Enable OSC timer interrupts
Enable QSPI interrupt
Enable SD/MMC interrupt
Enable SPI master interrupts
Enable SPI slave interrupts
Enable UART interrupts
Enable USB interrupts
Enable watchdog interrupts
Presets
Project
Library
Apply
Update...
Delete

With the ***PeripheralPin Multiplexing***, some I/O interface can be used by the HPS part or the FPGA part. The selection is done here.

3.3 Software development

3.3.1 ARM DS-5 tools

They are some differences between the versions of DS-5.

The one installed for the test is:

```
ARM DS-5 (DS-5 Altera Edition (Evaluation))
Version: 5.18.0
Build number: 5180018
```

3.3.2 Hello World on ARM HPS part

Copy the directory from Altera examples:

C:\altera\13.1\embedded\examples\software

And un-gz the file: Altera-SoCFPGA-HelloWorld-Baremetal-ARMCC.tar.gz

Then un-tar it.

The directory **Altera-SoCFPGA-HelloWorld-Baremetal-ARMCC** can then be copied in the Eclipse WorkSpace and Imported as a new project. The files inside are:

- .cproject used by Eclipse
- . project used by Eclipse
- ****.launch ??
- Makefile for the Compiler/Assembler/Linker
  An important info is the flag for the cpu: --cpu=Cortex-A9.no_neon.no_vfp
- scatter.scat Info for the compiler for the Code, Data, Stack and Heap addresses
  in this case in the internal SRAM

3.3.2.1 Scatter.scat

```
;*************************************************
; Copyright (c) 2013 Altera  All Rights Reserved.
;*************************************************

; Scatter-file for OnChip RAM based example
; This scatter-file places application code, data, stack and heap at suitable addresses in the memory map.

; Altera SoC-FPGA has 64kB of internal OnChip RAM

OCRAM 0xFFFF0000 0x10000
{
  APP_CODE +0
```

```
  {
    * (+RO, +RW, +ZI)
  }

  ARM_LIB_STACKHEAP  0xFFFF8000 EMPTY  0x8000 ; Application heap and stack
  { }
}
```

3.3.2.2 Makefile

Makefile for the ARM compiler

```
# Copyright (C) ARM Limited, 2011. All rights reserved.
#
# This example is intended to be built with the ARM Compiler armcc

TARGET=Altera-SoCFPGA-HelloWorld-Baremetal-ARMCC.axf

CC=armcc
AS=armasm
LD=armlink
AR=armar

# Select build rules based on Windows or Unix
ifdef WINDIR
DONE=@if exist $(1) echo Build completed.
RM=if exist $(1) del /q $(1)
SHELL=$(WINDIR)\system32\cmd.exe
else
ifdef windir
DONE=@if exist $(1) echo Build completed.
RM=if exist $(1) del /q $(1)
SHELL=$(windir)\system32\cmd.exe
else
DONE=@if [ -f $(1) ]; then echo Build completed.; fi
RM=rm -f $(1)
endif
endif


all: $(TARGET)
	$(call DONE,$(TARGET))

rebuild: clean all

clean:
	$(call RM,*.o)
	$(call RM,$(TARGET))

hello.o: hello.c
	$(CC) -c -g --cpu=Cortex-A9.no_neon.no_vfp -O0 hello.c

$(TARGET): hello.o scatter.scat
	$(LD) hello.o -o $(TARGET) --cpu=Cortex-A9.no_neon.no_vfp --scatter=scatter.scat
```

3.3.3 GPIO access

The references for gpio are:

- http://www.altera.com/literature/hb/cyclone-v/cv_54022.pdf
- http://www.altera.com/literature/hb/cyclone-v/hps.html
- Supports up to 71 I/O pins and 14 input-only pins depend on device variant

On de1-soc:

- Only 1 Button for HPS GPIO 1
- Only 1 LED for HPS GPIO 1

Pin Name	HPS GPIO	Register [bit]	Function	Address	Dir
HPS_KEY	GPIO54	GPIO1[25]	I/O	0xFF20 9000	In
HPS_LED	GPIO53	GPIO1[24]	I/O	0xFF20 9000	Out

HPS peripherals are mapped to HPS base address space 0xFC00 0000 with 64KB size.
Registers of GPIO0 controller are mapped to the base address 0xFF20 8000 - 0xFF20 8FFF (4KB size)
Registers of GPIO1 controller are mapped to the base address 0xFF20 9000 - 0xFF20 9FFF (4KB size)
Registers of GPIO2 controller are mapped to the base address 0xFF20 A000 - 0xFF20 8FFF (4KB size)

		http://www.altera.com/literature/hb/cyclone-v/cv_5v4.pdf		
GPIO0	0xFF20 8000 - 0xFF20 8FFF	0xFF70 8000		
GPIO1	0xFF20 9000 - 0xFF20 9FFF	0xFF70 9000		
GPIO2	0xFF20 A000 - 0xFF20 8FFF	0xFF70 A000		
LWFPGASLAVES		0xFF20 0000		

gpio0	0xFF70 8000	HPS_GPIO0_ADDRESS	HPS_GPIO0_OFFSET		
gpio_swporta_dr	0	HPS_GPIO0_GPIO_SWPORTA_DR_ADDRESS	GPIO_GPIO_SWPORTA_DR_OFFSET		
gpio_swporta_ddr	0x04	HPS_GPIO0_GPIO_SWPORTA_DDR_ADDRESS	GPIO_GPIO_SWPORTA_DDR_OFFSET		
gpio_inten	0x30	HPS_GPIO0_GPIO_INTEN_ADDRESS	GPIO_GPIO_INTEN_OFFSET		
gpio_intmask	0x34	HPS_GPIO0_GPIO_INTMASK_ADDRESS	GPIO_GPIO_INTMASK_OFFSET		
gpio_inttype_level	0x38	HPS_GPIO0_GPIO_INTTYPE_LEVEL_ADDRESS	GPIO_GPIO_INTTYPE_LEVEL_OFFSET		
gpio_int_polarity	0x3c	HPS_GPIO0_GPIO_INT_POLARITY_ADDRESS	GPIO_GPIO_INT_POLARITY_OFFSET		
gpio_intstatus	0x40	HPS_GPIO0_GPIO_INTSTATUS_ADDRESS	GPIO_GPIO_INTSTATUS_OFFSET		
gpio_raw_intstatus	0x44	HPS_GPIO0_GPIO_RAW_INTSTATUS_ADDRESS	GPIO_GPIO_RAW_INTSTATUS_OFFSET		
gpio_debounce	0x48	HPS_GPIO0_GPIO_DEBOUNCE_ADDRESS	GPIO_GPIO_DEBOUNCE_OFFSET		
gpio_porta_eoi	0x4c	HPS_GPIO0_GPIO_PORTA_EOI_ADDRESS	GPIO_GPIO_PORTA_EOI_OFFSET		
gpio_ext_porta	0x50	HPS_GPIO0_GPIO_EXT_PORTA_ADDRESS	GPIO_GPIO_EXT_PORTA_OFFSET		
gpio_ls_sync	0x60	HPS_GPIO0_GPIO_LS_SYNC_ADDRESS	GPIO_GPIO_LS_SYNC_OFFSET		
gpio_id_code	0x64	HPS_GPIO0_GPIO_ID_CODE_ADDRESS	GPIO_GPIO_ID_CODE_OFFSET		
gpio_ver_id_code	0x6c	HPS_GPIO0_GPIO_VER_ID_CODE_ADDRESS	GPIO_GPIO_VER_ID_CODE_OFFSET		
gpio_config_reg2	0x70	HPS_GPIO0_GPIO_CONFIG_REG2_ADDRESS	GPIO_GPIO_CONFIG_REG2_OFFSET		
gpio_config_reg1	0x74	HPS_GPIO0_GPIO_CONFIG_REG1_ADDRESS	GPIO_GPIO_CONFIG_REG1_OFFSET		

3.3.3.1 Library installation

C:\altera\13.1\embedded\ip\altera\hps\altera_hps\hwlib

HERE

3.3.3.2 Reference files

hps.h		

3.3.3.2.1 Titre5

3.3.3.2.1.1 Titre6

3.3.3.2.1.1.1 Titre7

3.3.3.2.1.1.1.1 Titre8

3.3.3.2.1.1.1.1.1 Titre9

References

- Altera, Cyclone V Devices documentation, http://www.altera.com/literature/lit-cyclone-v.jsp?ln=devices_fpga&l3=Low-Cost%20FPGAs-Cyclone%20V%20%28E,%20GX,%20GT,%20SE,%20SX,%20ST%29&l4=Documentation
- Cyclone V Device Handbook Volume 3: Hard Processor System Technical Reference Manual http://www.altera.com/literature/hb/cyclone-v/cv_5v4.pdf
- Cyclone V Hard Processor System User Guide http://www.altera.com/literature/hb/cyclone-v/cv_5v4_08.pdf
- Cyclone V, Device Datasheet http://www.altera.com/literature/hb/cyclone-v/cv_51002.pdf
- Cylone V HPS addresses http://www.altera.com/literature/hb/cyclone-v/hps.html
- Cyclone V Device Handbook Volume 1: Device Interfaces and Integration http://www.altera.com/literature/hb/cyclone-v/cyclone5_handbook.pdf

- Cyclone V, Device Overview
  http://www.altera.com/literature/hb/cyclone-v/cv_51001.pdf
- SoCAL documentation (html), The Altera SoC Abstraction Layer (SoCAL) API Reference Manual
  file:///C:/altera/13.1/embedded/ip/altera/hps/altera_hps/doc/socal/html/index.html
- Altera HWLIB, The Altera HW Manager API Reference Manual
  file:///C:/altera/13.1/embedded/ip/altera/hps/altera_hps/doc/hwmgr/html/index.html
- Cyclone V, A Bare-Metal Debugging using ARM DS-5 Altera Edition
  http://www.youtube.com/watch?v=CJ0EHJ9oQ7Y
- Linux Kernel Debug using ARM DS-5 Altera Edition
  http://www.youtube.com/watch?v=QcA39O6ofGw
- FPGA-adaptive debug on the Altera SoC using ARM DS-5
  http://www.youtube.com/watch?v=2NBcUv2TxbI
- A Look Inside: SoC FPGAs Introduction (Part 1 of 5)
  http://www.youtube.com/watch?v=RVM-ESUMOMU (Part 1 of 5)
  http://www.youtube.com/watch?v=Ssxf8ggmQk4 (Part 2 of 5)
  http://www.youtube.com/watch?v=cWIaqt2RU84 (Part 3 of 5)
  http://www.youtube.com/watch?v=gUE669XKhUY (Part 4 of 5)
  http://www.youtube.com/watch?v=NxZznvf5EKc (Part 5 of 5)
- DS-5 Altera Edition: Bare-metal Debug and Trace
  http://www.youtube.com/watch?v=u_xKybPhcHI
- OpenCL on FPGAs Accelerating Performance and Design Productivity — Altera
  http://www.youtube.com/watch?v=M6vpq6s1h_A
-

Summary

List of Figures