

Altera DE1-SoC GHRD

[Introduction](#)[Hardware](#)[Architecture](#)[HPS-to-FPGA Addr. Map](#)[Lightweight HPS-to-FPGA Addr. Map](#)[SDRAM assignments](#)[Programming the EPCQ Device](#)[How to debug?](#)[Download](#)

(2) J-Link Debugger (C5SoC)

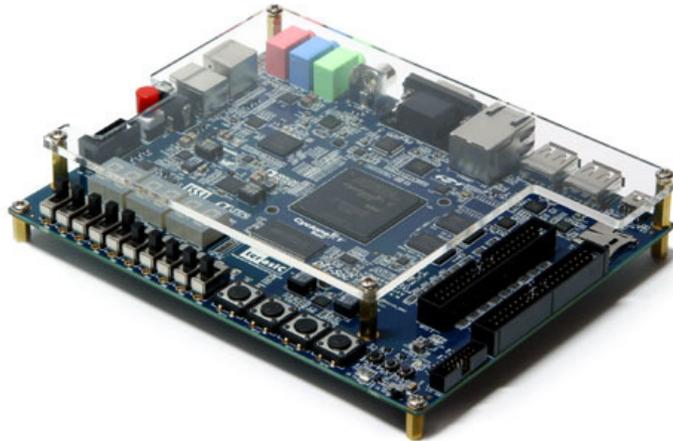
Introduction

Inspired by the Golden Hardware Reference Design from [RocketBoards](#) I have created my own design for the Altera DE1-SoC Board. It is based on the Terasic Verilog example with some modifications and ported to VHDL. The design was created with the Quartus II Web Edition [15.0.2.153](#)

[^ top](#)

Hardware

Here an [Altera DE1-SoC Board](#) was used:



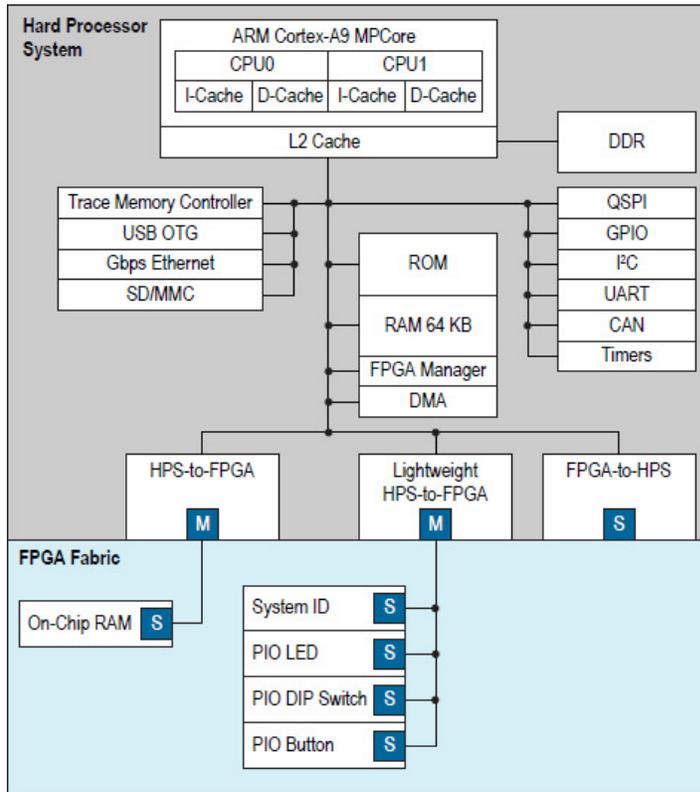
The DE1-SoC board provides a lot of functionality. But only the following subset was used:

- Cyclone V SoC (5SCEMA5F31C6)
- ARM Cortex-A9 (HPS)
- 1GB (2x256Mx16) DDR3 SDRAM on HPS
- USB to UART (micro USB type B connector)
- 4 User Keys (FPGA x4)
- 10 User switches (FPGA x10)
- 11 User LEDs (FPGA x10, HPS x 1)

[^ top](#)

Architecture

The architecture of the hardware looks like:



(Picture source: [RocketBoards](#))

The architecture looks very similar like the design from RocketBoards. But here I have removed the two JTAG Masters which was needed for the System Console.

[^] top

HPS-to-FPGA Address Map

From the perspective of the MPU (Cortex-A9) the HPS-to-FPGA address starts at **0xC0000000** and the map looks like:

Peripheral	Address Offset	Size (bytes)	Attribute
onchip_memory2_0	0x0	64K	On-chip RAM as scratch pad

[^] top

Lightweight HPS-to-FPGA Address Map

From the perspective of the MPU (Cortex-A9) the Lightweight HPS-to-FPGA address starts at **0xFF200000** and the map looks like:

Peripheral	Address Offset	Size (bytes)	Attribute
sysid_qsys	0x10000	8	Unique system ID
led_pio	0x10040	32	LED output
dipsw_pio	0x10080	32	DIP switch input
button_pio	0x100C0	32	Push button input

[^] top

SDRAM assignments

After the **very first run** of Qsys "Generate HDL..." the SDRAM assignments must be executed. After the Qsys code generation start "Analysis & Synthesis". It takes a few minutes (depending on host machine speed) before the "Analysis & Synthesis" will finish.

Now the SDRAM assignments can be created. In Quartus, go to "Tools > Tcl Scripts..." and select the **hps_sdram_p0_pin_assignments.tcl** file. Click "Run" to execute. A notification window will appear when the script is completed. Now the full "Compile Design" process can be used.

