OpenRISC development board

Startup guide

Brought to You By ORSoC / OpenCores

Legal Notices and Disclaimers

Copyright Notice

General Disclaimer

The Publisher has strived to be as accurate and complete as possible in the creation of this ebook, notwithstanding the fact that he does not warrant or represent at any time that the contents within are accurate due to the rapidly changing nature of information.

The Publisher will not be responsible for any losses or damages of any kind incurred by the reader whether directly or indirectly arising from the use of the information found in this ebook.

This ebook is not intended for use as a source of legal, business, accounting, financial, or medical advice. All readers are advised to seek services of competent professionals in the legal, business, accounting, finance, and medical fields.

No guarantees of any kind are made. Reader assumes responsibility for use of the information contained herein. The Publisher reserves the right to make changes without notice. The Publisher assumes no responsibility or liability whatsoever on the behalf of the reader of this report.

Distribution Rights

The Publisher grants you the following rights for re-distribution of this ebook.

[YES] Can be given away.
[YES] Can be packaged.
[YES] Can be offered as a bonus.
[NO] Can be edited completely and your name put on it.
[YES] Can be used as web content.
[NO] Can be broken down into smaller articles.
[NO] Can be added to an e-course or auto-responder as content.
[NO] Can be submitted to article directories (even YOURS) IF at least half is rewritten!
[NO] Can be added to paid membership sites.
[NO] Can be added to an ebook/PDF as content.
[NO] Can be offered through auction sites.
[NO] Can sell Resale Rights.
[NO] Can sell Master Resale Rights.
[NO] Can sell Private Label Rights.

Table of Contents

Quick start

The board is shipped with a preinstalled OR1200 system inside the FPGA.

To get the system up and running follow these easy steps

1. connect the JTAG cable to the USB connector on your PC
2. install the FTDI USB driver to enable the JTAG cable
3. start a terminal emulator (such as TeraTerm)
   choose the USB serial port
   set speed to 115200 baud
   turn on local echo
4. connect the flat cable to JP3
   pin 1 has a marking at the board and a color marking in the cable
5. connect 3.3V supply to the board

You should get a prompt on the terminal. Pressing enter will start readout of data starting at address 0x0.

If it does not work

1. No text in the terminal
   1. Type at the keyboard and check LED indicator on the JTAG cable
      No indication – check serial port setup in terminal emulator
   2. check serial port speed. Should be 115200 baud 8n1
2. No serial port
   1. try disconnecting the JTAG cable and the insert again
      verify that a new serial port turns up
3. I accidentally connected to a supply larger than 3.3V
   1. order a new development board

Standalone usage

This board can be used standalone. Normally the IO voltage for the IO banks connected to JP1 and JP6 are supplied over the connectors.

For standalone usage connect VIOB (JP1 pin 40 and JP6 pin 40).

Standalone usage also requires the 1.5V DC/DC converter for the ACTEL core supply.

Debug environment

Boot code

A small boot loader in internal FLASH copies a simple monitor application from the SPI FLASH to SDRAM. Upon completion application is executed.

If a terminal emulator is connected to JP3 a short message and a prompt should appear.

The application accepts some simple command to read and write to memory

Command summary

Command			
G	GO	Hexadecimal 32 bit address as 8 ASCII characters	G00000100
M	MODIFY	Hexadecimal 32 bita address as 8 ASCII characters followed by ":" and 32 bit data modifies memory content	M00000000:12345678
S	S-record	Memory content transferred as Motorola S-record	

JTAG debug

The processor can be stalled with the use of the JTAG debugger. When the COU is stalled application program can be downloaded over the JTAG interface and executed under control of the GNU debugger.

To start JTAG debugger a proxy server must first be started to handle the JTAG communication.

To start application type:

jp_usb 12345

This start the proxy with socket number 12345. Next start GDB with GUI.

ddd –debugger /path-to-binaries/or32-uclinux-gdb

From within GDB type:

target jtag jtag://localhost:12345

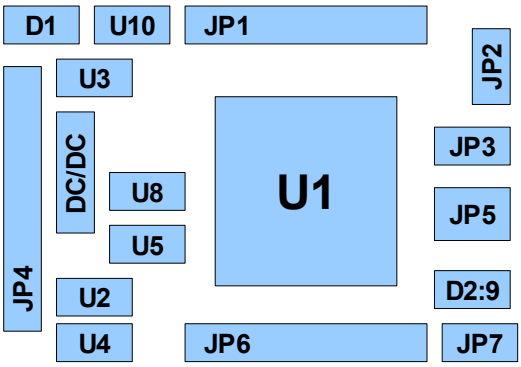
Some useful commands

CMDs	Description	Example
x	Display memory content	x/100 0x100
break	Set breakpoint	break *0x100
delete	Delete breakpoint(s)	delete
set	Modify memory content	set *0x100=0x0
load	Load application code	load vmlinux
c	Continue execution	
si	Single step	
ESC	Interrupt execution	

Development board description

Functional description

The development board has the following functions



U1 - FPGA

The board contains an ACTEL ProASIC3 FPGA in a PQ208 package. The board can optionally be used with other ACTEL FPGA devices from ACTEL A3P or A3PE families.

The following variants are available:

FPGA	System gates	Versa Tiles DFF	RAM kbits	4608 bit blocks	FLASH ROM bits	PLLs	Global nets	I/O banks	IO SE / Diff
A3P1000	1M	24576	144	32	1024	1	18	4	154/35
A3PE3000	3M	75264	504	112	1024	6	18	8	147/65

For configuration of the FPGA use JTAG connector JP2.
For configuration use ACTEL FLASH Pro 3 JTAG download cable.

U5, U8, JP5 - Memory

The board has 32 Mbyte of SDRAM and a 1 Mbit SPI FLASH device.
There is also a SD connector, JP5, for high density FLASH cards.

The SDRAM is a Micron SDR SDRAM memory, MC48LC16M16-7E.

The memory has the following address table:

	Configuration	Refresh count	Row addressing	Bank addressing	Column addressing
16 Meg x 16	4 Meg x 16 x 4 banks	8K	8K (A0-A12)	4 (BA0, BA1)	512 (A0-A8)

The board has a M25P10 serial FLASH memory. The memory is organized as 4 sectors, each containing 128 pages. Each page is 256 bytes wide. Thus, the whole memory can be viewed as consisting of 512 pages, or 131,072 bytes.

The whole memory can be erased using the Bulk Erase instruction, or a sector at a time, using the Sector Erase instruction.

The FLASH PROM uses SPI communication. Input data is latched in on the rising edge of

Serial Clock (C), and output data is available from the falling edge of Serial Clock (C).

In addition there is also a SD FLASH card connector on the bottom side of the board. This connector can be used for high capacity FLASH cards for mass storage.

JP1, JP6, JP7 - IO

There are two 2x20 pin 2 mm pitch headers for external connections. Each connector can be configured for any IO supply voltage.

A 10 pin 0.1" pitch header can be used for low density IO over a 10 pin flat cable.

JP3 - SoC debug

The OpenRISC processor has hardware support for JTAG debugging. On this board there is also a LVTTL serial port used as console. Both are connected to the debug connector. A JTAG debug cable with two independent channels can be used for both JTAG debug and a console.

This JTAG debug cable is available from ORSoC.

On board is also 8 LEDs that can be used for debugging purposes.

U2, U3, U4 - Oscillators

The board has three separate oscillators;

1. SDRAM and SoC , 25 MHz
2. Ethernet SMII channels @ 125 MHz.
   The SMII clock has a clock driver supporting up to 4 SMII channels
3. audio or general purpose

Only the first oscillator is mounted. Depending on needs the other two can optionally be mounted.

The first oscillator is connected to both the SDRAM CLK pin and a clock input on the FPGA. The SDRAM will run at the oscillator clock speed. The internal SoC could be running on any multiple of this clock through the PLL.

Second clock is intended for multiple Ethernet SMII channels. The board support up to four channels. This oscillator connects through a zero delay buffer to both the FPGA and to JP1. On A3PE equipped board this clock can use the PLL inside the FPGA.

Third clock could be used for VGA, audio or USB.

DC/DC converter

The board requires a single 3.3 volt supply if a local 1.5V DC/DC converter is mounted as a module. A linear converter sources the PLLs in the FPGA. All supply is available in JP4.

If a local 1.5V converter is not present this supply must be connected to JP4.

Recommended Resources

ORSoC – http://www.orsoc.se

ORSoC is a fabless ASIC design & manufacturing services company, providing RTL to ASIC design services and silicon fabrication service. **ORSoC** are specialists building complex system based on the OpenRISC processor platform.

Open Source IP – http://www.opencores.org

Your number one source for open source IP and other FPGA/ASIC related information.