LL5000 User Manual

Contents:

Figures:

Tables:

1. LL5000 Development System

1.1. Features

- Xilinx Spartan3 FPGA
- ARM LPC2144 Housekeeping Processor
- 16MB SDRAM on-board
- 16MB FLASH on-board
- High-speed SelectMAP FPGA configuration from Platform Flash In-System
- Programmable Configuration PROM
- On-board 10/100 Ethernet PHY device
- Silicon Serial Number for unique board identification
- SD/MMC card slot
- RS-232 DB9 serial port
- Two PS-2 serial ports
- Eight LEDs connected to Spartan3 I/O pins
- LCD 16 x 2 character display with backlight
- Eight switches connected to Spartan3 I/O pins
- Five push buttons connected to Spartan3 I/O pins
- Two high-speed mezzanine board connectors joined to 80 Spartan3 I/O pins
- AC-97 audio CODEC with audio amplifier and speaker/headphone output and line level output
- Microphone and line level audio input
- On-board VGA output, 640 x 480 at 60 Hz supported by software) with added signal generator capability
- On-board video decoder with CVBS (composite), Y/C (S-video) and YPrPb (component) video input support
- On-board video encoder with with CVBS (composite), Y/C (S-video), YPrPb (component) and EuroSCART RGB video output support
- On-board power supplies
- Power-on reset circuitry

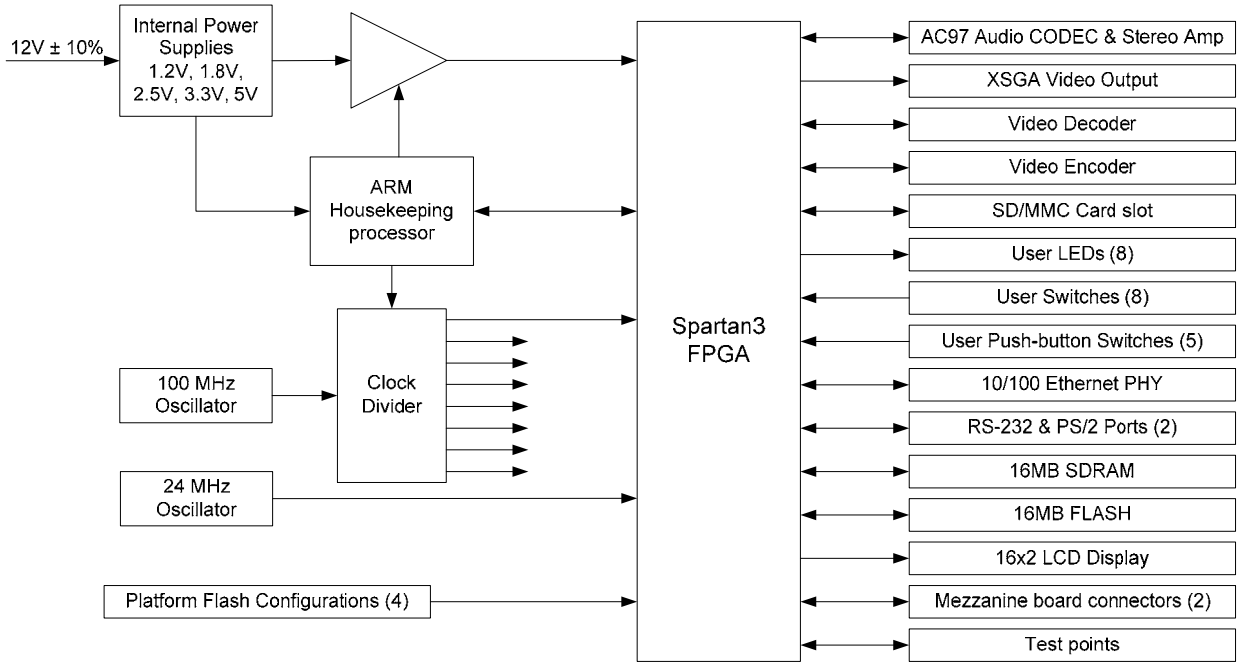
1.2. General Description

The LL5000 Development System provides an advanced hardware platform that consists of a low cost Spartan3 Platform FPGA surrounded by a comprehensive collection of peripheral components that can be used to create a complex system.

1.2.1. Block Diagram

Figure 1.1 shows a block diagram of the LL5000 Development System.

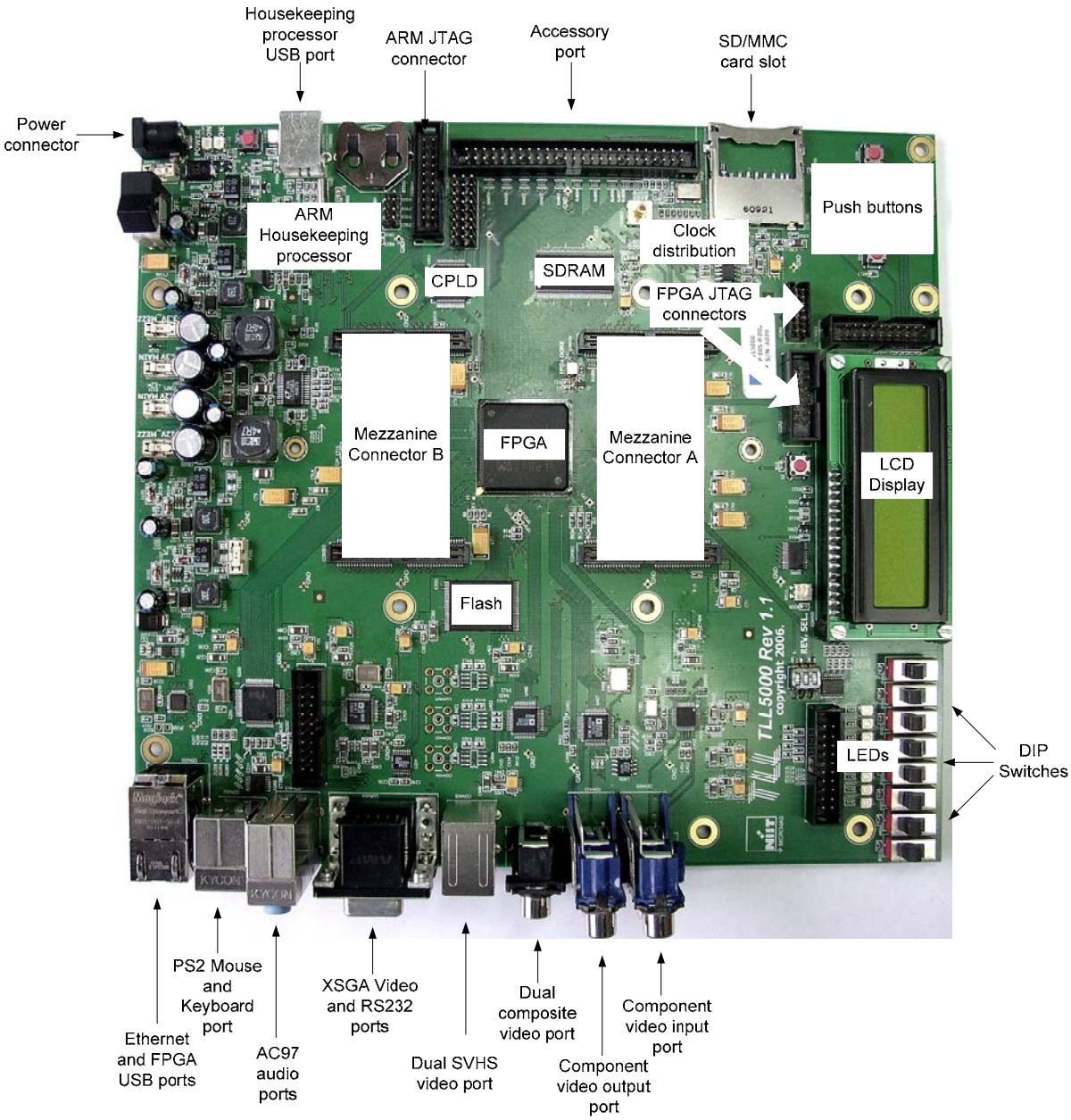
Figure 1.1: **LL5000 Development System Block Diagram**



1.2.2. Board Components

This section contains a concise overview of several important components on the Ll5000 Development System (see Figure 1.2).

Figure 1.2: **LL5000 Development System Board Photo**



1.2.3. Spartan3 FPGA

U1 is a Xilinx Spartan3 XC3S1500 FPGA device packed in 676-lead fine-pitch ball grid array package. All XC3S1500 key features can be seen in ***Table 1.1***.

Table 1.1: **XC3S1500 Features**

Feature	Value
Equivalent Logic Cells	29952
Array Size	64 x 52
Distributed RAM	208 Kbits
Block RAM	576 Kbits
Dedicated Multipliers	32
DCMs	4

1.2.4. ARM Housekeeping Processor

ARM Housekeeping processor takes care of power up sequencing and monitoring of the supply regulators. It also sets up and manages communications with host computer via high USB connection.

1.2.5. Power Supplies

The LL5000 Development System is powered from a 12V regulated power supply. On-board switching power supplies generate 5V, 3.3V, 2.5V, 1.8V and 1.2V for the FPGA and peripheral components.

ARM housekeeping processor is powered all the time and it has control over power distribution to the rest of the board. All generated voltage levels are constantly monitored by the housekeeping processor.

1.2.6. FPGA Configuration

FPGA can be programmed directly using JTAG chain or indirectly using Platform Flash. Used Platform Flash device (XCF32PVO48C) can contain up to 4 FPGA code revisions.

1.2.7. System RAM

LL5000 Development System has on board mounted Micron 1Meg x 32 x 4 Banks SDRAM module.

1.2.8. System Flash

LL5000 Development System has on board mounted 128Mbit Spansion S29GL128N Flash module.

1.2.9. Ethernet interface

The LL5000 Development System provides an IEEE-compliant Fast Ethernet transceiver that supports both 100BASE-TX and 10BASE-T applications. It supports full duplex operation at 10 Mb/s and 100 Mb/s, with auto-negotiation and parallel detection. The PHY provides a Media Independent Interface (MII) for attachment to the 10/100 Media Access Controller (MAC) implemented in the FPGA. Each board is equipped with a Silicon Serial Number that uniquely identifies it with a 48-bit serial number. This serial number is retrieved using "1-Wire" protocol. This serial number can be used as the system MAC address.

1.2.10. Serial interfaces

The LL5000 Development System provides three serial ports: a single RS-232 port and two PS/2 ports. The RS-232 port supports hardware handshake and it uses a standard DB-9 serial connector. This connector is typically used for communications with a host computer using a standard 9-pin serial cable connected to a COM port. The two PS/2 ports could be used to attach a keyboard and mouse to the LL5000 Development System.

1.2.11. User LEDs, Switches, and Push Buttons

A total of eight LEDs are provided for user-defined purposes. Turning LED on is done by driving a logic 0 to the corresponding FPGA pin. A single eight-position DIP switch and five push buttons are provided for user input. Middle push button is used as FPGA reset button.

1.2.12. VGA Output

The LL5000 Development System includes a video DAC and 15-pin highdensity D-sub connector to support XSGA output. The video DAC can operate with a pixel clock of up to 180 MHz. Only VESA-compatible output of 640 x 480 at 60 Hz refresh is supported by software. It can also be used as a 3-channel signal generator.

1.2.13. Video decoder

The LL5000 Development System includes a video decoder with CVBS (composite), Y/C (S-video) and YPrPb (component) video input support. It supports NTSC/PAL/SECAM video standards with additional video standard autodetection feature.

1.2.14. Video encoder

The LL5000 Development System offers composite, Y/C (S-video) and YPrPb (component) video output support provided by Analog Devices ADV7173 video encoder.

1.2.15. AC97 Audio CODEC

An audio CODEC and stereo power amplifier are included on the LL5000 Development System to provide a high-quality audio path and provide all of the analog functionality in a PC audio system. It features a full-duplex stereo ADC and DAC, with an analog mixer, combining the line-level inputs, microphone input, and PCM data.

1.2.16. Expansion Connectors

LL5000 Development System includes two 80 pin (2 x 40) mezzanine board connectors. Every connector provides 40 Spartan3 I/O pins, JTAG signals and 3.3V and 12V power supplies.

1.2.17. SD-CARD interface

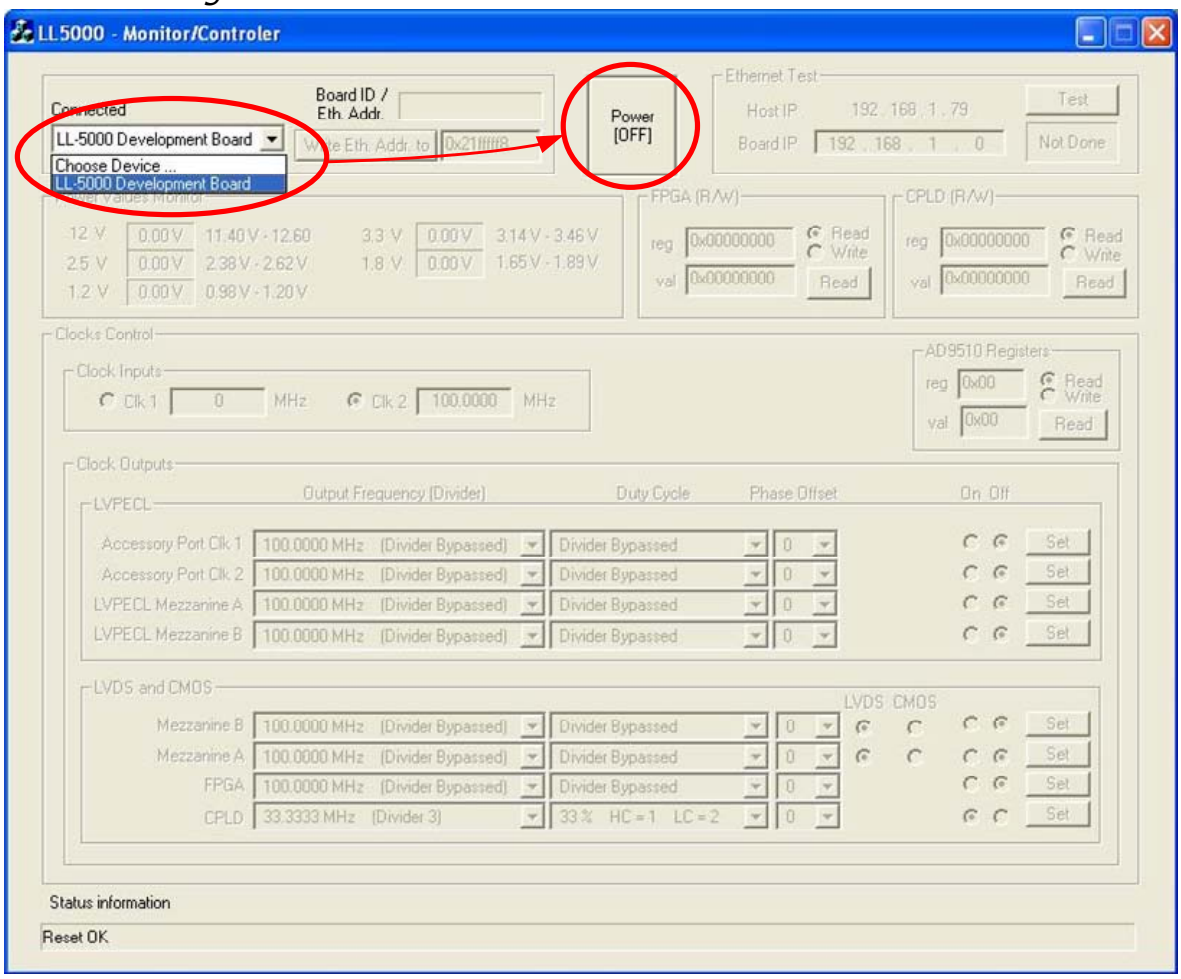
LL5000 system includes a header for SD and MMC cards which enable users to store their data on a removable media.

2. Using the System

2.1. Power and Clock Distribution

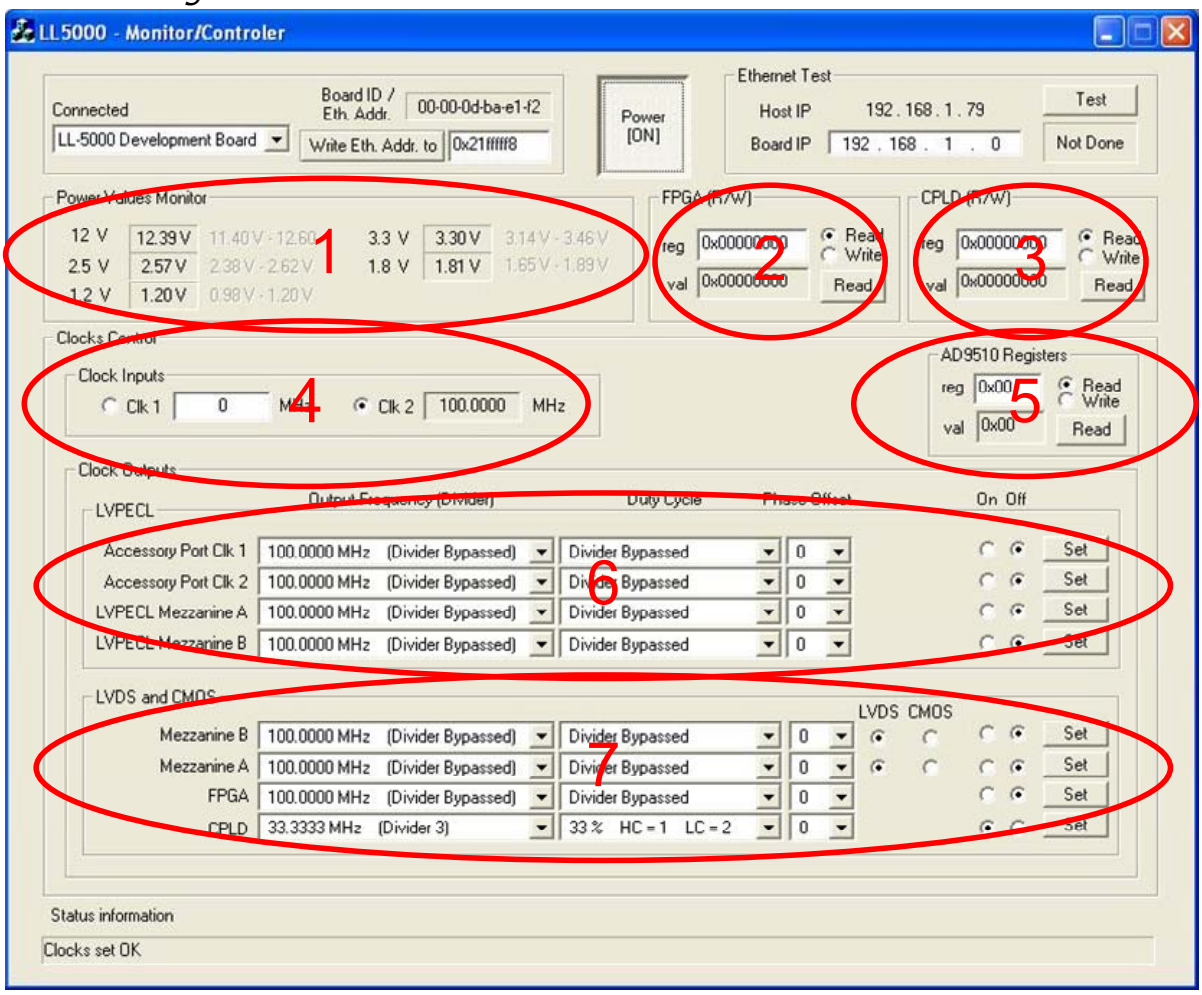
Before starting LL5000 Monitor/Controller application LL5000 Development System should be connected to the PC using ARM USB port. LL5000 Monitor/Controller application initial screen is shown in ***Figure 2.1***.

Figure 2.1: **LL5000 Monitor/Controller Initial Screen**



First LL5000 Development Board device should be selected using circled drop-down box. After that board can be powered up using Power button. After powering up, LL5000 Monitor/Controller screen should look like it is shown in ***Figure 2.2***.

Figure 2.2: **LL5000 Monitor/Controller Power On Screen**



1. Power Values Monitor shows current power supply levels with their valid ranges.
2. FPGA (R/W) provides FPGA OPB read/write functionality. It can be used for debug purposes.
3. CPLD (R/W) provides CPLD registers read/write functionality. It can be used for debug purposes.
4. AD9510 Clock Divider input clock selection. External clock (Clk 1) or on-board generated 100MHz clock (Clk 2) can be selected as Clock Divider input clock.
5. AD9510 Clock Divider manual register access. It should be used if desired clock distribution can not be obtained using controls 6 and 7.
6. Four LVPECL Clock Divider outputs settings. Divide value and phase offset can be individually adjusted for each channel. Channel setting is implemented by pressing the corresponding set button.
7. Two selectable CMOS/LVDS and two CMOS Clock Divider outputs settings. Divide value and phase offset can be individually adjusted for each channel. For Mezzanine clock outputs LVDS or CMOS signal standards can be selected. Channel setting is implemented by pressing the corresponding set button.

2.2. Configuring the FPGA

The FPGA is programmed from Xilinx EDK software during FPGA software development. This is done by selecting menu entry Device configuration/ Download bitstream.

During power-up of the board, the FPGA can be programmed by one of four available revisions inside Xilinx Platform flash XCF32P. To enable multiple revisions, place switch 1 from DIP switch pack S1 to position "1". To select the desired revision, use switches 2 and 3 from the same pack

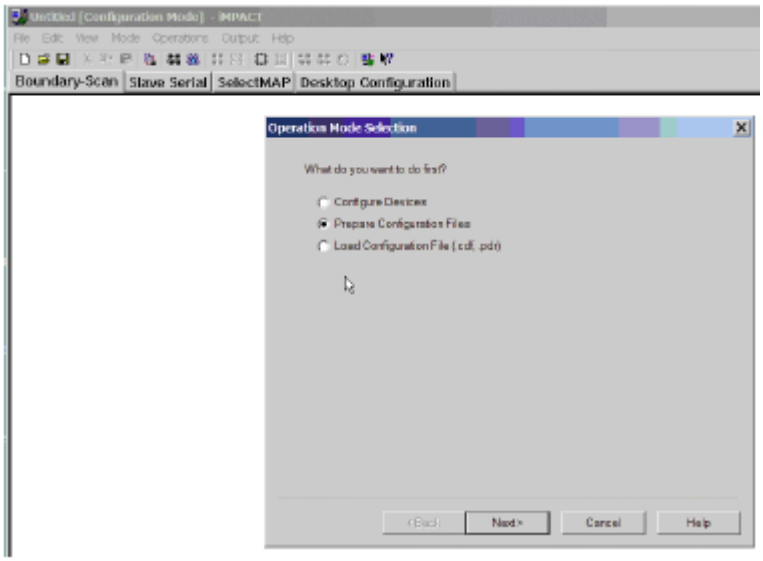
Table 2.1: **FPGA revisions**

Position of switch 2	Position of switch 3	Revision number	Default function
1	1	0	Peripheral tests except video and Ethernet
0	1	1	Memory tests
1	0	2	Video and Ethernet tests
1	1	3	No default function

To prepare a PROM file with multiple revisions, proceede as follows:
The **.bit** file created by the Xilinx implementation tools must be converted to an **.MCS** file before it can be programmed into the Platform FLASH PROM.

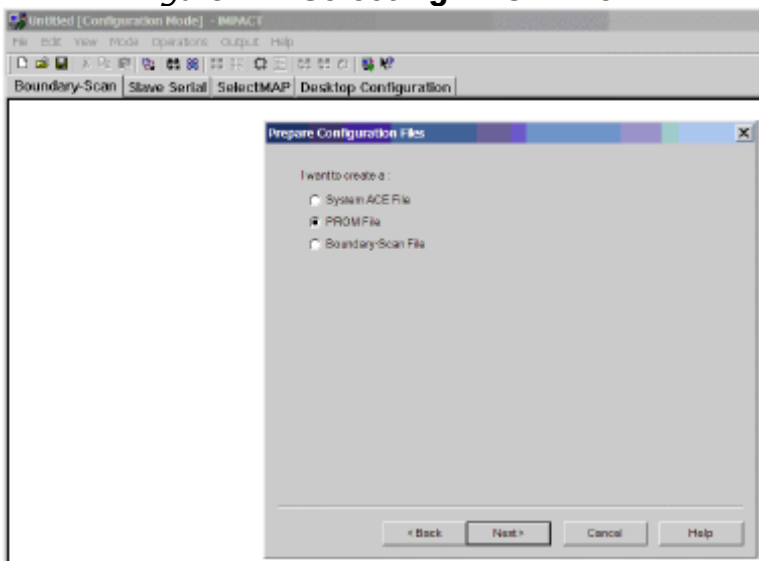
1. Start iMPACT and select Prepare Configuration Files as shown in Figure 2.3.

Figure 2.3: **Operation Mode Selection: Prepare Configuration Files**



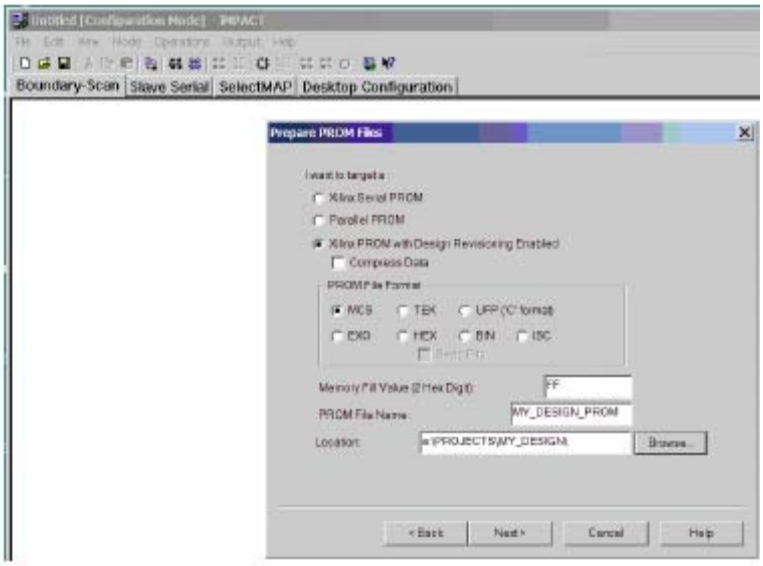
2. Click on Next and select PROM File in the Prepare Configuration Files option menu shown in Figure 2.4.

Figure 2.4: **Selecting PROM file**



3. Click on Next and then select Xilinx PROM with Design Revisioning Enabled using the MCS PROM File Format.

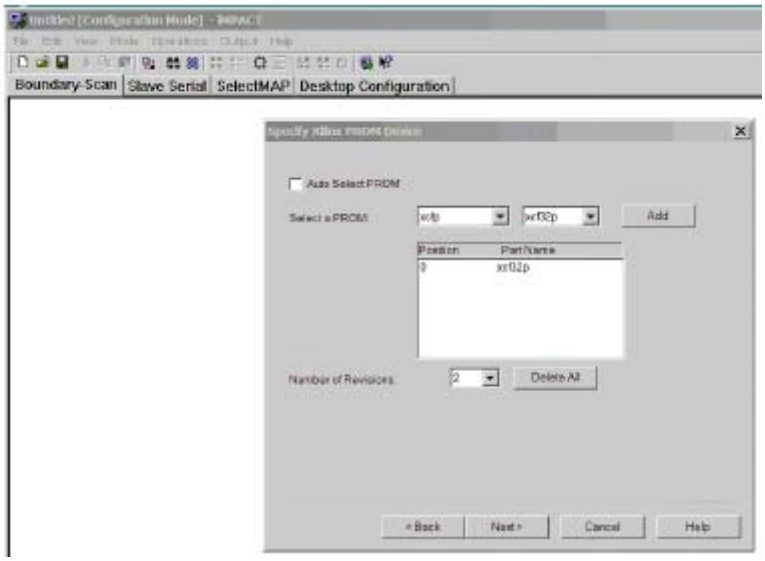
Figure 2.5: **Selecting a PROM with Design Revisioning Enabled**



4. Give the PROM File a name of your choice in the location of your choice as shown in Figure 2.5.

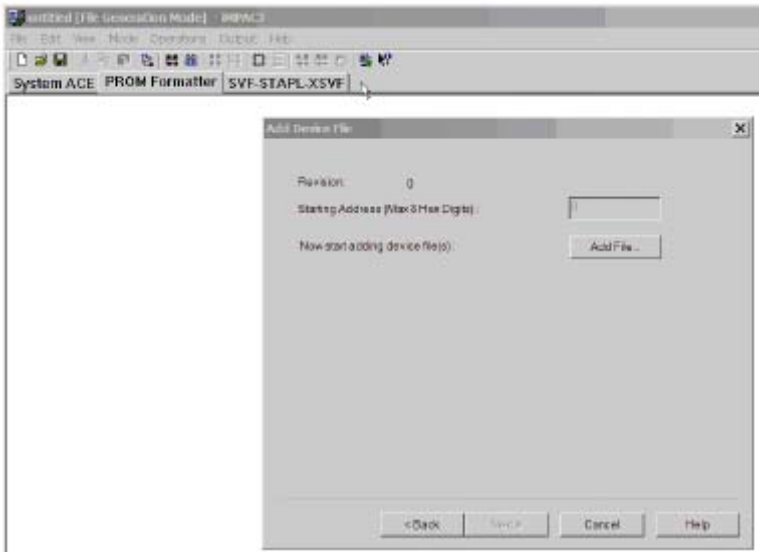
   Note: Do NOT select Compress Data, because the LL5000 development System hardware does not support this option.

Figure 2.6: **Selecting an XCF32P PROM with 3 Revisions**



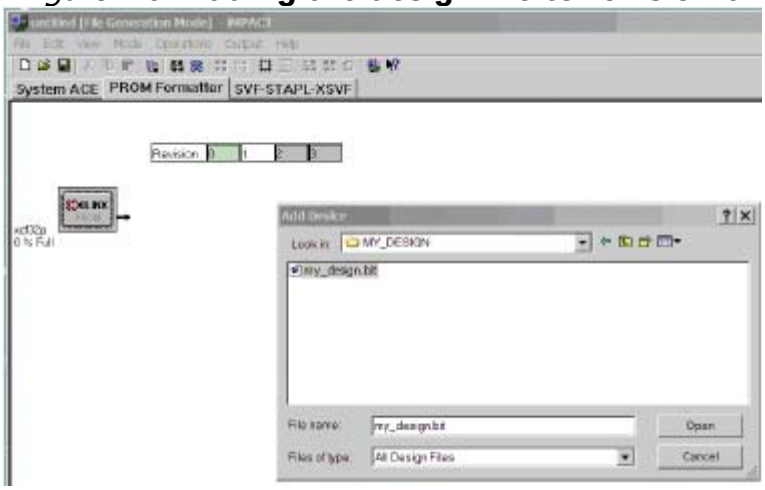
5. Click on Next to bring up the option screen where the type of PROM is specified.
6. Select the XCF32P PROM from the drop down men. Click on the "Add" button and specify "3" from the Number of Revisions drop down menu as shown in Figure 2.6.
7. Click on Next twice to bring up the Add Device File screen shown in Figure 2.7.

Figure 2.7: **Adding a device file**



8. Click on Add File and navigate to your design directory and select the .bit file for your design as shown in Figure 2.8.

Figure 2.8: **Adding the design file to revision 0**



9. Click on Open and answer No when prompted to add another design file to Revision 0.
10. Note that Revision 0 is highlighted in green; this is where the "known" configuration will be placed in the PROM. By selecting your design file for Revision 0, you are just reserving space in the PROM for the known configuration.

    If the design file was created with the Startup Clock set to JTAG, iMPACT will issue a warning that the Startup Clock will be changed to CCLK in the bitstream programmed into the PROM. This warning can be safely ignored.
11. Once you answer No when prompted to add another design file to Revision 0, the green revision highlight will move to Revision 1. You will be prompted to add your design file to Revision 1. By selecting your design file for Revision 0, you are just reserving space in the PROM for the known configuration.
12. Click on Open and answer No when prompted to add another design file to Revision 1.
13. Once you answer No when prompted to add another design file to Revision 1, the green revision highlight will move to Revision 2. You will be prompted to add you design file to Revision 2.
14. Click on Open and answer No when prompted to add another design file to Revision 1. Click on Finish to start the generation of the MCS
15. After iMPACT successfully creates the MCS file, select Configuration Mode from the Mode menu.
16. Make sure that LL5000 is powered up and that a PC4 cable connects the board to the PC that is running the iMPACT software.
17. Select the Initialize Chain command. The iMPACT software then interrogates the system and reports that there are three devices in the JTAG chain. The first device is the XCF32P PROM; the second device is the CPLD; and the third device is the Spartan 3 FPGA.

18. Select the MCS file that you created earlier as the configuration file for the XCF32P PROM and click Open, as shown in Figure 7.
19. Select BYPASS as the configuration files for the CPLD and the Spartan 3 FPGA.
20. Right mouse click on the icon for the XCF32P PROM and select Program from the drop down menu as shown in Figure.
21. The iMPACT software responds with a form that allows the user to specify which design revisions are to be programmed and the programming options for the various revisions. De-select Design Revision Rev 0 and Rev 1 and all of the options for these revisions to minimize the programming time.

Figure 2.9: **Selecting configuration file for XCF32P**

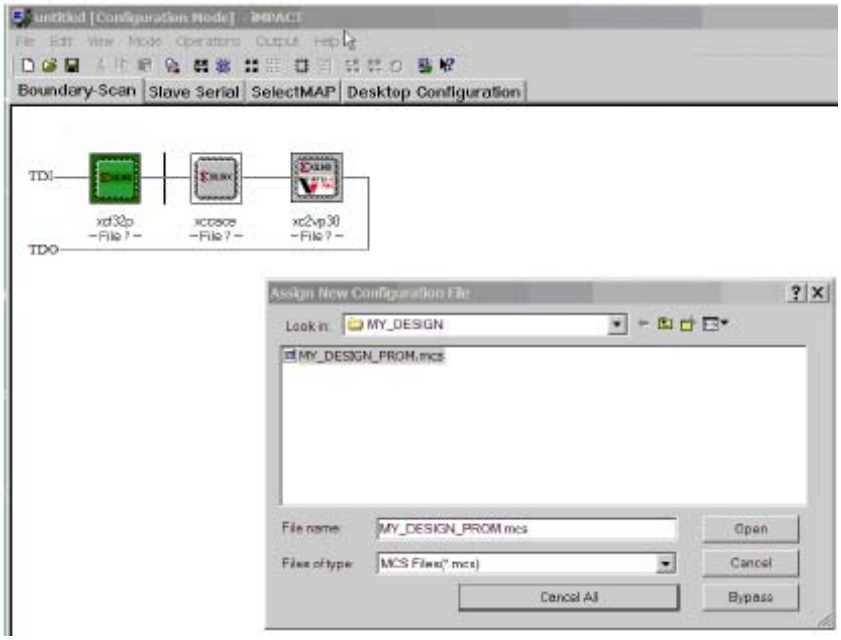
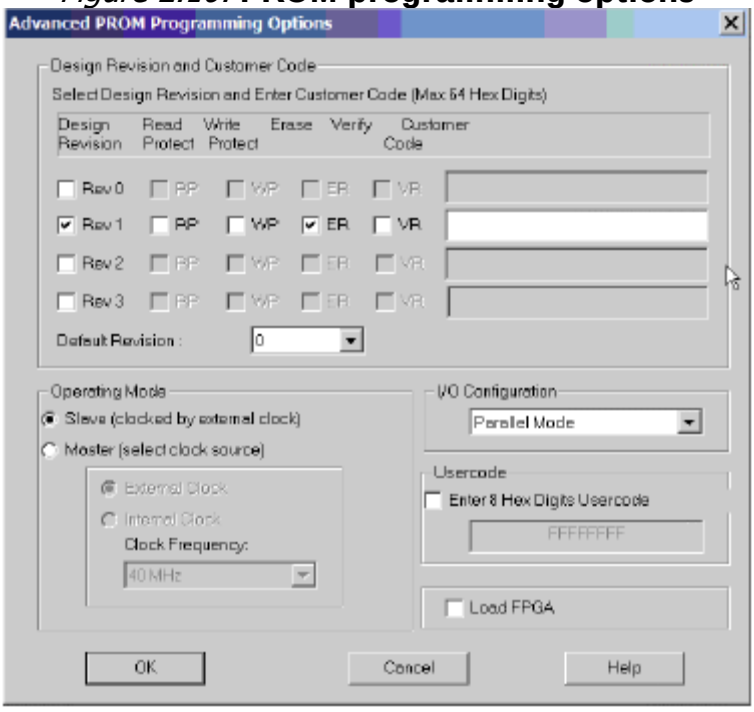


Figure 2.10: **PROM programming options**



22. Select Design Revision Rev 2, and set the Erase (ER) bit to erase any previous "User" design. Make sure that the Write Protect (WP) bit is **not** set.
23. Verify that the Operating Mode is set to Slave and the I/O Configuration is set to Parallel Mode as shown in Figure 2.10.
24. Click on OK to begin programming the PROM

2.3. Using the SDRAM

The LL5000 board is equipped with a single 16Mb SDRAM IC, Micron MT48LC4M32B2P which is connected directly to Bank 3 of the FPGA. FPGA to SDRAM connection list is shown in Table 2.2.

Table 2.2: **SDRAM to FPGA connection**

FPGA BALL	SCH Net Name	Interface Signal Name	FPGA BALL	SCH Net Name	Interface Signal Name
P26	RAM_IO0	DQ1	V25	RAM_IO28	CS#
P25	RAM_IO1	DQ0	V24	RAM_IO29	DQ29
P24	RAM_IO2	DQ13	V23	RAM_IO30	DQ28
P23	RAM_IO3	DQ12	V22	RAM_IO31	A0
P22	RAM_IO4	DQ14	U20	RAM_IO32	A6
P21	RAM_IO5	DQ15	V20	RAM_IO33	DQM3
P20	RAM_IO6	DQ2	W26	RAM_IO34	BA1
P19	RAM_IO7	CLK	W25	RAM_IO35	A11
R26	RAM_IO8	DQ4	W24	RAM_IO36	DQ27
R25	RAM_IO9	DQ3	W23	RAM_IO37	DQ26

R24	RAM_IO10	DQ11	V21	RAM_IO38	A4
T23	RAM_IO11	DQ9	W22	RAM_IO39	DQM2
R22	RAM_IO12	DQ10	Y26	RAM_IO40	A11
R21	RAM_IO13	DQ5	Y25	RAM_IO41	A10
R20	RAM_IO14	A8	W21	RAM_IO42	A3
R19	RAM_IO15	CKE	W20	RAM_IO43	DQ31
T26	RAM_IO16	DQ7	AA26	RAM_IO44	DQ16
T25	RAM_IO17	DQ6	AA25	RAM_IO45	A2
T22	RAM_IO18	RAS#	Y23	RAM_IO46	DQ25
T21	RAM_IO19	DQM0	Y22	RAM_IO47	DQ18
T20	RAM_IO20	A7	AA24	RAM_IO48	DQ25
T19	RAM_IO21	A9	AA23	RAM_IO49	DQ23
U26	RAM_IO22	CAS#	AB26	RAM_IO50	DQ19
U25	RAM_IO23	WE#	AB25	RAM_IO51	DQ17
U24	RAM_IO24	DQ8	Y21	RAM_IO52	DQ21
U23	RAM_IO25	DQM1	Y20	RAM_IO53	DQ30
U22	RAM_IO26	BA0	AC26	RAM_IO54	DQ22
U21	RAM_IO27	A5	AC25	RAM_IO55	DQ20

The memory has 1 Meg x 32 x 4 banks. It uses 12 address lines (A0-A11) for row addressing and 8 address lines (A0-A7) for column addressing. It has 4 banks, maximum operating frequency of 143MHz and CAS latency CL=3. A generic Xilinx SDRAM controller (OPB_SDRAM) IP is used to enable access to SDRAM in the design.

2.4. Using the System Flash

The LL5000 board is equipped with a Spansion S29GL-N MirrorBit™ Flash Family device. The provided IP enables access to the flash with the following commands: read, erase sector, erase chip, write word, write buffer.

The connection between the Flash and the FPGA is given in Table 2.3.

Table 2.3: **Flash to FPGA connection**

FPGA BALL	SCH Net Name	Interface Signal Name	FPGA BALL	SCH Net Name	Interface Signal Name
F6	FLASH_IO0	WE#	H3	FLASH_IO23	A5
F5	FLASH_IO1	A21	H2	FLASH_IO24	DQ6
E4	FLASH_IO2	A0	H1	FLASH_IO25	DQ13
E3	FLASH_IO3	CE#	J7	FLASH_IO26	A14
D2	FLASH_IO4	DQ2	K7	FLASH_IO27	A10
D1	FLASH_IO5	DQ9	J5	FLASH_IO28	A18
G7	FLASH_IO6	A22	J4	FLASH_IO29	A4
G6	FLASH_IO7	RESET#	J3	FLASH_IO30	A3
E2	FLASH_IO8	DQ3	J2	FLASH_IO31	DQ14
E1	FLASH_IO9	DQ10	K6	FLASH_IO32	A11
F4	FLASH_IO10	OE#	K5	FLASH_IO33	A17
F3	FLASH_IO11	DQ0	K4	FLASH_IO34	A2

G5	FLASH_IO12	WP#
G4	FLASH_IO13	DQ8
F2	FLASH_IO14	DQ4
F1	FLASH_IO15	DQ11
H7	FLASH_IO16	A15
H6	FLASH_IO17	A13
G2	FLASH_IO18	DQ5
G1	FLASH_IO19	DQ12
H5	FLASH_IO20	RY/BY#
J6	FLASH_IO21	A12
H4	FLASH_IO22	DQ1

K3	FLASH_IO35	A1
K2	FLASH_IO36	DQ15/A_1
K1	FLASH_IO37	DQ7
L8	FLASH_IO38	A19
L7	FLASH_IO39	A9
L6	FLASH_IO40	A6
L5	FLASH_IO41	A7
L2	FLASH_IO42	A16
L1	FLASH_IO43	BYTE#
M8	FLASH_IO44	A20
M7	FLASH_IO45	A8

2.5. Using the XSGA output

The LL5000 board is equipped with a triple 8-bit DAC ADV7125 (U402), a high density 15-pin D-Sub connector (CON203B), and IP placed in the FPGA. The TTL data inputs and control signals are converted into analog current outputs that can drive 25Ω to 37.5Ω loads, corresponding to a doubly-terminated 50Ω to 75Ω load. The BLANK input overrides the RGB inputs and blanks the display output.

Three additional buffered SMB connectors are provided which can be used as a rudimentary signal generator. The provided FPGA IP enables reading the desired output pattern from a memory area inside the SDRAM and outputting it to the D-Sub connector. In case the signal generator is used, no video will be visible on the monitor attached to the D-Sub connector.

The connections with the FPGA are given in Table 2.4.

Table 2.4: VGA DAC connections to FPGA

Table 2.4: **VGA DAC connections to FPGA**

FPGA BALL	Schematic Net Name	Interface Signal Name
W1	VDAC_IO0	B1
V6	VDAC_IO1	#SYNC
U7	VDAC_I02	#BLANK
V5	VDAC_IO3	G0
V4	VDAC_IO4	R6
V3	VDAC_IO5	R5
V2	VDAC_IO6	B2
U6	VDAC_IO7	G2
U5	VDAC_IO8	G1
U4	VDAC_IO9	R4
U3	VDAC_IO10	R3
U2	VDAC_IO11	B3
U1	VDAC_IO12	B4
T8	VDAC_IO13	G6
T7	VDAC_IO14	G7

T6	VDAC_IO15	G3
T5	VDAC_IO16	R1
T2	VDAC_IO17	B5
T1	VDAC_IO18	B6
R8	VDAC_IO19	G5
R7	VDAC_IO20	R7
R6	VDAC_IO21	G4
R5	VDAC_IO22	#PSAVE
T4	VDAC_IO23	R2
R3	VDAC_IO24	R0
R2	VDAC_IO25	B7
R1	VDAC_IO26	CLOCK
P8	VDAC_IO27	B0

2.6. Using the Video Encoder

The board has an ADV7173 video encoder. It is an integrated Digital Video Encoder that converts digital CCIR-601 4:2:2 8-bit component video data into a standard analog baseband television signal compatible with world wide standards.

There are six DACs available on the ADV7173. In addition to the Composite output signal there is the facility to output S-VHS Y/C Video, RGB Video and YUV Video. The on-board SSAF (Super Sub-Alias Filter), with extended luminance frequency response and sharp stopband attenuation, enables studio quality video playback on modern TVs, giving optimal horizontal line resolution. An additional sharpness control feature allows extra luminance boost on the frequency response. A PC'98-Compliant autodetect feature has been added to allow the user to determine whether or not the DACs are correctly terminated. If not, the ADV7173 flags that they are not connected through the Status bit and provides the option of automatically powering them down, thereby reducing power consumption. The ADV7173 also supports both PAL and NTSC square pixel operation. The parts also incorporate WSS and CGMS-A data control generation.

The output video frames are synchronized with the incoming data Timing Reference Codes. Optionally the encoder accepts (and can generate) *HSYNC*, *VSYNC*, and FIELD timing signals. These timing signals can be adjusted to change pulsewidth and position while the part is in the master mode. The Encoder requires a single two times pixel rate (27 MHz) clock for standard operation. Alternatively the Encoder requires a 24.5454 MHz clock for NTSC or 29.5 MHz clock for PAL square pixel mode operation. All internal timing is generated on-chip.

HSO/CSO and VSO TTL outputs, synchronous to the analog output video, are also available. A programmable CLAMP output signal is also available to enable clamping in either the front or back porch of the video signal. A separate teletext port enables the user to directly input teletext data during the vertical blanking interval.

Table 2.5 shows the connection list between the FPGA and the video encoder. The provided FPGA IP enables reading out a region of SDRAM memory containing CCIR-601 data stream and sending it to the encoder circuitry. The video encoder can be used stand-alone or together with the video decoder circuit.

Table 2.5: **Video encoder connection to FPGA**

FPGA BALL	Schematic Net Name	Interface Signal Name
AB2	VENC_IO0	P1
AB1	VENC_IO1	P2
Y7	VENC_IO2	TTX
Y6	VENC_IO3	#CSO/#HSO
AA4	VENC_IO4	P0
AA3	VENC_IO5	#VSO
Y5	VENC_IO6	FIELD/#VSYNC
Y4	VENC_IO7	#RESET
AA2	VENC_IO8	P3
AA1	VENC_I09	P4
Y2	VENC_IO10	P5
Y1	VENC_IO11	P6
W7	VENC_IO12	TTXREQ
W6	VENC_IO13	#HSYNC
W5	VENC_IO14	#BLANK
V7	VENC_IO15	SCRESET/RTC
W4	VENC_IO16	PAL/#NTSC
W3	VENC_IO17	CLAMP
W2	VENC_IO18	P7

2.7. Using the Video Decoder

The LL5000 contains a video decoder circuitry, ADV7180, enabling conversion from standard analog PAL/NTSC video signal to CCIR-601 data.

The ADV7180 automatically detects and converts standard analog baseband television signals compatible with worldwide NTSC, PAL, and SECAM standards into 4:2:2 component video data compatible with the 8-bit ITU-R BT.656 interface standard.

External HS, VS, and FIELD signals provide timing references for LCD controllers and other video ASICs, if required The accurate 10-bit analog-to-digital conversion provides professional quality video performance for consumer applications with true 8-bit data resolution. Three analog video input channels accept standard composite, S-video, or component video signals, supporting a wide range of consumer video sources. AGC and clamp-restore circuitry allow an input video signal peak-to-peak range up to 1.0 V. Alternatively, these can be bypassed for manual settings.

The provided IP core enables setting ADV7180 registers using I^2C and receiving CCIR-601 data and storing it to SDRAM memory from where it can be read by the video encoder or a user application. Connections to the FPGA are given in Table 2.6.

Table 2.6: **Video Decoder Schematic Connection**

FPGA BALL	Schematic Net Name	Interface Signal Name
AD2	VDEC_IO0	#RESET

AD1	VDEC_IO1	P1
AB4	VDEC_IO2	P2
AB3	VDEC_IO3	LCC
AC2	VDEC_IO4	P0
AC1	VDEC_IO5	#PWRDWN
AA6	VDEC_IO6	P3
AB6	VDEC_IO7	P4
AD5	VDEC_IO8	P5
AC6	VDEC_IO9	P6
AD6	VDEC_IO10	P7
AC7	VDEC_IO11	SFL
AC8	VDEC_IO12	HS
AD8	VDEC_IO13	#INTRQ
AC9	VDEC_IO14	VS/FIELD

2.8. Using the AC97 Audio Codec

LL5000 has an AC97 rev 2.3 compliant IC, AD1981BL. It can be used to record an play audio, but also as a rudimentary oscilloscope with.

AD1981BL characteristics include:

- S/PDIF output, 20-bit data format, supporting 48 kHz and 44.1 kHz sample rates
- Integrated stereo headphone amplifier
- Variable sample rate audio
- External audio power-down control
- >90 dB dynamic range
- Stereo full-duplex codec
- 20-bit PCM DAC
- 3 analog line-level stereo inputs for line-in, AUX, and CD
- Mono line-level phone input
- Dual MIC input with built-in programmable preamplifier
- High quality CD input with ground sense
- Mono output for speakerphone or internal speaker
- power management support
- Stereo MIC preamplifier support
- Built-in digital equalizer function for optimized
- speaker sound
- Full-duplex variable sample rates from 7040 Hz to
- 48 kHz with 1 Hz resolution
- Jack sense pins for automatic output switching
- Software-programmed V_{REFOUT} output for biasing
- microphone and external power amplifier
- Low power 3.3 V operation for analog and digital supplies
- Multiple codec configuration options

The provided IP is a Xilinx IP enabling access to the registers of AD1981BL. Additional software functions enable the user to record a 10sec long recording to the SDRAM

memory and play it back through the headphones, as well as set input and output gains.

Table 2.7: **AC97 codec connection with the FPGA**

FPGA BALL	Schematic Net Name	Interface Signal Name
P7	AC97_IO0	SDATA_IN
P6	AC97_IO1	BIT_CLK
P5	AC97_IO2	SDATA_OUT
P4	AC97_IO3	#ID_1
P3	AC97_IO4	#ID_0
P2	AC97_IO5	SYNC
P1	AC97_IO6	#RESET

2.9. Using the LEDs, Switches and Buttons

The LL5000 board has 8 surface-mounted LED diodes, 5 push-buttons and 8 DIP-switches.

The LED diodes are illuminated when the user outputs a logical "1" to the appropriate pin of the FPGA (Table 2.9), and they are off when a logical "0" is outputted to a pin. Series resistors limit the current to about 4mA when the Led is illuminated.

5 pushbuttons are in placed on tips of an imaginary diamond-form shape, giving the user the form of a joystick. When a button is pressed, it generates a logic "0" o the pin of the FPGA, while logic "1" is generated when a button is not pressed. Table 2.8 shows which pins of the FPGA are connected to which push-button.

Table 2.8: **Pushbutton connections**

FPGA BALL	Schematic Net Name	Interface Signal Name
AF13	UI_JOY0	UI_JOY0
AB24	UI_JOY1	UI_JOY1
AB23	UI_JOY2	UI_JOY2
AA22	UI_JOY3	UI_JOY3
AA21	UI_JOY4	UI_JOY4

Table 2.9: **LED connected to the FPGA**

FPGA BALL	Schematic Net Name	Interface Signal Name
AB7	UI_LED0	UI_LED0
AB8	UI_LED1	UI_LED1
AB9	UI_LED2	UI_LED2
AA9	UI_LED3	UI_LED3
AA10	UI_LED4	UI_LED4
AA11	UI_LED5	UI_LED5
AD12	UI_LED6	UI_LED6
Y13	UI_LED7	UI_LED7

8 DIP-switches are in placed in line on the board. When the switch is on, it generates logic "0" on the pin of the FPGA to which it is connected, while logic "1" is generated when a switch is off. Table 2.10 shows which pins of the FPGA are connected to which DIP switch.

Table 2.10: **DIP switches connected to the FPGA**

FPGA BALL	Schematic Net Name	Interface Signal Name
Y8	UI_SW0	UI_SW0
Y9	UI_SW1	UI_SW1
Y10	UI_SW2	UI_SW2
Y11	UI_SW3	UI_SW3
Y12	UI_SW4	UI_SW4
W11	UI_SW5	UI_SW5
W12	UI_SW6	UI_SW6
W13	UI_SW7	UI_SW7

A generic Xilinx IP core, opb_gpio is used to access all these peripherals.

2.10. Using the LCD Display

The LCD display onboard LL5000 consists of a Display electronic GmbH LCD module SYH 16216 SYH-LY which uses a Hitachi HD44780 LCD controller in 4 bit mode. A Xilinx IP core, opb_gpio_v3_01_b is used to access this peripheral, while additional software functions are given which enable resetting the LCD controller and writing on line or line 2 of he LCD.

Table 2.11 shows FPGA to LCD connection list.

Table 2.11: **LCD connection with FPGA**

FPGA BALL	Schematic Net Name	Interface Signal Name
J22	LCD_IO0	LCD_D3
K22	LCD_IO1	LCD_D2
K21	LCD_IO2	LCD_D1
L21	LCD_IO3	LCD_D0
M21	LCD_IO4	LCD_EN
N21	LCD_IO5	LCD_R/#W
M20	LCD_IO6	LCD_RS
N20	LCD_IO7	LCD_L+

2.11. Using the Serial Port

The LL5000 Development System has a single RS-232 port. The RS-232 port is configured as a Data Communication Equipment (DCE) with hardware handshake using a standard DB-9 serial connector. Considering +/-12V logic levels on RS232 connectors, ADM3202 high speed rs232/v.28 interface from Analog Devices is used for coupling.

The FPGA IP core which is used for RS232 is Xilinx opb_uartlite_v1_00_a core. Functions are provided which enable sending messages from the LL5000 to the PC, but also receiving user input from the PC.

Table 2.12: **RS232 connection to the FPGA**

FPGA BALL	Schematic Net Name	Interface Signal Name
M1	RS232_RX	RS232_RX
M2	RS232_TX	RS232_TX
N1	RS232_CTS	RS232_CTS
M6	RS232_RTS	RS232_RTS

2.12. Using the Ethernet Network Interface

The Ethernet interface on LL5000 is centered on Intel's PHY LXT972A. It's a single-Port 10/100 Mbps PHY Transceiver which directly supports both 100BASE-TX and 10BASE-T applications. It supports full-duplex operation at 10Mbs and 100Mbs. Operating conditions for the LXT972A Transceiver can be set using auto-negotiation, parallel detection, or manual control. The transceiver requires only a single 2.5 or 3.3 V power supply with 2.5 V MII interface support.

The Ethernet interface uses a standard Xilinx IP core, opb_ethernetlite which ships with Xilinx EDK software with a hardware evaluation license. Additional functions are given which enable verifying the interface functionality using the LL5000 desktop software.

There are a number of jumpers around the Ethernet PHY, the explanation is given in ***Table 2.13***.

Table 2.13: **Jumper settings for the Ethernet interface**

Jumper designator	Default position	Description
JS200	Open	TX output slew rate setting
JS201	Open	TX output slew rate setting
JS202	Open	Pause capability advertising during negotiation
JS203	Connected to GND	Device address setting

2.13. Using the SD-Card

The LL5000 Development Board has single SD Card slot directly connected to the FPGA. Electrical interface specification along with communication SPI (serial bus standard established by Motorola) access protocol for SD cards is accomplished with "SD Card Physical Layer System Specification, Version 1.01" defined by SD Card Association.

Communication is implemented with Xilinx SPI IP core opb_spi_v1_00_d. Additional functions are provided which enable initializing the card, reading it's size and block read and write. The connection to the FPGA is given in ***Table 2.14***.

Table 2.14: **SD-card interface connection to FPGA**

FPGA BALL	Schematic Net Name	Interface Signal Name
J21	MMC_IO0	DI/CD/CMD
K20	MMC_IO1	SW_CI

H21	MMC_IO2	CD/DAT3
J20	MMC_IO3	DAT2
L19	MMC_IO4	Do/DAT0
L20	MMC_IO5	CLK
M19	MMC_IO6	DAT1
N19	MMC_IO7	SW_WP

2.14. Using the Mezzanine Expansion Connectors

The LL-5000 is a composed of a BASE BOARD and a number of MEZZANINE BOARDS and is intended to provide the prototyping capability for designing the hardware and the software of an embedded computing system. For that purpose two pair of high speed connectors (for mezzanine board A and B) is implemented on the board. Each of the interface consist of 80 bits wide bus connected to FPGA, I^2C bus, two differential (or selectable one single ended and one LVPECL) clocks from clock generator subsystem and dedicated JTAG lines.

Connectors are symmetrical both in electrical and physical sense, giving the opportunity to use both pair of connectors equally.

12V and 3.3V power supplies are brought to connectors for powering up the mezzanine boards. 80 bit interface is routed as high speed 50ohm lines, with equalized line length to achieve signal integrity for a wide range of possible mezzanine boards.

High speed connectors QTE-40-02-L-D-A-K from Samtec are used for the mezzanine boards interconnecting. Mating connectors on the mezzanine boards should be QSE-40-01-L-D-A-K. Please refer to manufacturer resources for more detailed information on the connectors.

FPGA to mezzanine connectors A and B connection list is shown in Table 2.15 and Table 2.16 respectively.

Table 2.15: **Mezzanine A connections with the FPGA**

FPGA BALL	SCH Net Name	Interface Signal Name	FPGA BALL	SCH Net Name	Interface Signal Name
AF15	MZ_IO_A0	CON900.25	AD22	MZ_IO_A40	CON901.21
AE15	MZ_IO_A1	CON900.27	AC21	MZ_IO_A41	CON901.23
AF16	MZ_IO_A2	CON900.31	AD21	MZ_IO_A42	CON901.25
AE16	MZ_IO_A3	CON900.33	AC20	MZ_IO_A43	CON901.27
AF17	MZ_IO_A4	CON900.35	AC19	MZ_IO_A44	CON901.31
AE17	MZ_IO_A5	CON900.37	AD19	MZ_IO_A45	CON901.33
AE18	MZ_IO_A6	CON900.41	AC18	MZ_IO_A46	CON901.35
AF19	MZ_IO_A7	CON900.43	AD18	MZ_IO_A47	CON901.37
AE19	MZ_IO_A8	CON900.45	AC17	MZ_IO_A48	CON901.41
AF20	MZ_IO_A9	CON900.47	AD17	MZ_IO_A49	CON901.43
AE20	MZ_IO_A10	CON900.51	AC16	MZ_IO_A50	CON901.45
AF21	MZ_IO_A11	CON900.53	AB16	MZ_IO_A51	CON901.47
AE21	MZ_IO_A12	CON900.55	AD15	MZ_IO_A52	CON901.51

AF22	MZ_IO_A13	CON900.57
AE22	MZ_IO_A14	CON900.61
AF23	MZ_IO_A15	CON900.63
AE23	MZ_IO_A16	CON900.65
AF24	MZ_IO_A17	CON900.67
AE24	MZ_IO_A18	CON900.71
AD25	MZ_IO_A19	CON900.73
AB20	MZ_IO_A20	CON900.75
AA20	MZ_IO_A21	CON900.77
W15	MZ_IO_A22	CON900.26
W16	MZ_IO_A23	CON900.28
Y16	MZ_IO_A24	CON900.32
Y17	MZ_IO_A25	CON900.34
Y18	MZ_IO_A26	CON900.36
AA18	MZ_IO_A27	CON900.38
AA19	MZ_IO_A28	CON900.42
Y19	MZ_IO_A29	CON900.44
AC22	MZ_IO_A30	CON900.46
AD23	MZ_IO_A31	CON900.48
AB21	MZ_IO_A32	CON900.52
AB22	MZ_IO_A33	CON900.54
AA15	MZ_IO_A34	CON900.56
AA16	MZ_IO_A35	CON900.58
AA17	MZ_IO_A36	CON900.62
AB17	MZ_IO_A37	CON900.64
AB18	MZ_IO_A38	CON900.66
AB19	MZ_IO_A39	CON900.68

AB15	MZ_IO_A53	CON901.53
AB14	MZ_IO_A54	CON901.55
AC11	MZ_IO_A55	CON901.57
AB10	MZ_IO_A56	CON901.61
AD10	MZ_IO_A57	CON901.63
AC10	MZ_IO_A58	CON901.65
AD9	MZ_IO_A59	CON901.67
AA8	MZ_IO_A60	CON901.71
AA7	MZ_IO_A61	CON901.73
AF4	MZ_IO_A62	CON901.75
AD4	MZ_IO_A63	CON901.77
AE12	MZ_IO_A64	CON901.22
AF12	MZ_IO_A65	CON901.24
AE11	MZ_IO_A66	CON901.26
AF11	MZ_IO_A67	CON901.28
AE10	MZ_IO_A68	CON901.32
AF10	MZ_IO_A69	CON901.34
AE9	MZ_IO_A70	CON901.36
AE8	MZ_IO_A71	CON901.38
AF8	MZ_IO_A72	CON901.46
AE7	MZ_IO_A73	CON901.48
AF7	MZ_IO_A74	CON901.50
AE6	MZ_IO_A75	CON901.52
AF6	MZ_IO_A76	CON901.56
AE5	MZ_IO_A77	CON901.58
AF5	MZ_IO_A78	CON901.60
AE4	MZ_IO_A79	CON901.62

Table 2.16: **Mezzanine B connection with the FPGA**

FPGA BALL	SCH Net Name	Interface Signal Name
F21	MZ_IO_B0	CON902.25
F20	MZ_IO_B1	CON902.27
E21	MZ_IO_B2	CON902.31
E20	MZ_IO_B3	CON902.33
E19	MZ_IO_B4	CON902.35
E18	MZ_IO_B5	CON902.37
F18	MZ_IO_B6	CON902.41
F17	MZ_IO_B7	CON902.43
F16	MZ_IO_B8	CON902.45
F15	MZ_IO_B9	CON902.47
G14	MZ_IO_B10	CON902.51
G13	MZ_IO_B11	CON902.53
G12	MZ_IO_B12	CON902.55
F13	MZ_IO_B13	CON902.57

FPGA BALL	SCH Net Name	Interface Signal Name
B3	MZ_IO_B40	CON903.21
A3	MZ_IO_B41	CON903.23
B4	MZ_IO_B42	CON903.25
E5	MZ_IO_B43	CON903.27
E6	MZ_IO_B44	CON903.31
C4	MZ_IO_B45	CON903.33
D5	MZ_IO_B46	CON903.35
C5	MZ_IO_B47	CON903.37
D6	MZ_IO_B48	CON903.41
C6	MZ_IO_B49	CON903.43
D7	MZ_IO_B50	CON903.45
D8	MZ_IO_B51	CON903.47
C8	MZ_IO_B52	CON903.51
D9	MZ_IO_B53	CON903.53

F12	MZ_IO_B14	CON902.61
F11	MZ_IO_B15	CON902.63
F10	MZ_IO_B16	CON902.65
E10	MZ_IO_B17	CON902.67
E9	MZ_IO_B18	CON902.71
E8	MZ_IO_B19	CON902.73
E7	MZ_IO_B20	CON902.75
F7	MZ_IO_B21	CON902.77
G19	MZ_IO_B22	CON902.26
F19	MZ_IO_B23	CON902.28
G18	MZ_IO_B24	CON902.32
G17	MZ_IO_B25	CON902.34
G16	MZ_IO_B26	CON902.36
G15	MZ_IO_B27	CON902.38
H16	MZ_IO_B28	CON902.42
H15	MZ_IO_B29	CON902.44
H14	MZ_IO_B30	CON902.46
H13	MZ_IO_B31	CON902.48
H12	MZ_IO_B32	CON902.52
H11	MZ_IO_B33	CON902.54
G11	MZ_IO_B34	CON902.56
G10	MZ_IO_B35	CON902.58
G9	MZ_IO_B36	CON902.62
F9	MZ_IO_B37	CON902.64
F8	MZ_IO_B38	CON902.66
G8	MZ_IO_B39	CON902.68

C9	MZ_IO_B54	CON903.55
D10	MZ_IO_B55	CON903.57
C10	MZ_IO_B56	CON903.61
D11	MZ_IO_B57	CON903.63
E11	MZ_IO_B58	CON903.65
C12	MZ_IO_B59	CON903.67
E12	MZ_IO_B60	CON903.71
C13	MZ_IO_B61	CON903.73
D13	MZ_IO_B62	CON903.75
E13	MZ_IO_B63	CON903.77
A4	MZ_IO_B64	CON903.22
B5	MZ_IO_B65	CON903.24
A5	MZ_IO_B66	CON903.26
B6	MZ_IO_B67	CON903.28
A6	MZ_IO_B68	CON903.32
B7	MZ_IO_B69	CON903.34
A7	MZ_IO_B70	CON903.36
B8	MZ_IO_B71	CON903.38
A8	MZ_IO_B72	CON903.46
B9	MZ_IO_B73	CON903.48
B10	MZ_IO_B74	CON903.50
A10	MZ_IO_B75	CON903.52
B11	MZ_IO_B76	CON903.56
A11	MZ_IO_B77	CON903.58
B12	MZ_IO_B78	CON903.60
A12	MZ_IO_B79	CON903.62

2.15.Enabling LL5000 Board in the Base System Builder Wizard

For enabling LL-5000 BOARD to be visible among the other boards, like predefined Xilinx and other vendor boards its necessary to copy Board Definition File NIT_LL5000_v2_2_0.xbd in the proper directory. Generally position of those files must be in board subdirectory of the EDK working folder. Typical example is: c:\EDK\board\Xilinx\boards\NIT_LL5000\data\ NIT_LL5000_v2_2_0.xbd. That practically means that user first must create directory NIT_LL5000 in c:\EDK\board\Xilinx\boards. Next step is to create data subfolder in NIT_LL5000 folder and finally xbd file from installation CD must be copied to it. After completion of this procedure LL-5000 is prepared to be chosen from Base System Builder Wizard in which automated instantiation of board peripherals is enabled.

3. Document history

Date	Version	Author	Reamrks
23.10.2006.	1.0	Dusan Majstorovic	Initial version.
03.11.2006	1.1	Nebojša Pjevalica	How to use xbd file explained
20.03.2007.	1.2	Dusan Majstorovic	Board layout and schematic connections updated to match board revision 1.1