Lab 1: Simple Hardware Design

Targeting MicroBlaze™ on Spartan™-3E Starter Kit
Lab 1: Simple Hardware Design Lab

Introduction

This lab guides you through the process of using Xilinx Platform Studio (XPS) to create a simple processor system targeting the Spartan-3E Starter Kit.

Objectives

After completing this lab, you will be able to:

- Create an XPS project by using the Base System Builder (BSB)
- Create a simple hardware design by using Xilinx IP cores available in the Embedded Development Kit

Procedure

The purpose of the lab exercises is to walk you through a complete hardware and software processor system design. Each lab will build upon the previous lab. The following diagram represents the completed design (Figure 1-1).
In this lab, you will use the BSB of the XPS system to create a processor system consisting of the following processor IP (Figure 1-2):

- MicroBlaze (version 7.1)
- PLB_MDM
- LMB BRAM controllers for BRAM
- BRAM
- UART for serial communication
- GPIO for LEDs
- MPMC controller for external DDR_SDRAM memory

**Figure 1-1. Completed Design**

In this lab, you will use the BSB of the XPS system to create a processor system consisting of the following processor IP (Figure 1-2):

- MicroBlaze (version 7.1)
- PLB_MDM
- LMB BRAM controllers for BRAM
- BRAM
- UART for serial communication
- GPIO for LEDs
- MPMC controller for external DDR_SDRAM memory
Figure 1-2. Processor IP

This lab comprises three primary steps:

1. Create a project using the Base System Builder
2. Analyze the created project
3. Test in hardware

For each procedure within a primary step, there are general instructions (indicated by the symbol). These general instructions only provide a broad outline for performing the procedure. Below these general instructions, you will find accompanying step-by-step directions and illustrated figures that provide more detail for performing the procedure. If you feel confident about completing a procedure, you can skip the step-by-step directions and move on to the next general instruction.
Creating the Project Using the Base System Builder  Step 1

Launch Xilinx Platform Studio (XPS) and create a new project. Use Base System Builder to generate a MicroBlaze system and memory test application targeting the Spartan-3E starter kit.

1. Open XPS by selecting Start → Programs → Xilinx ISE Design Suite 10.1 → EDK → Xilinx Platform Studio

2. Leave the default Base System Builder option and click OK to start the wizard (Figure 1-3). If you clicked cancel, you can select File → New Project and the same dialog box will appear.

Figure 1-3. New Project Creation Using Base System Builder

3. Browse to c:xup\embedded\labs directory, create a new folder called lab1 and select it, and click Open followed by click Save (Figure 1-4). Click <OK>.

Figure 1-4. Assigning Project Directory
Select the **I would like to create a new design option** in the Welcome to Base System Builder dialog box and click **Next**.

In the **Select Board** dialog box, specify the settings below (**Figure 1-5**) and click **Next** to continue.

- **Board Vendor:** Xilinx
- **Board Name:** Spartan™-3E Starter Board
- **Board Revision (Verify on board):** D

**Figure 1-5. Select Board Dialog Box**

In the **Select Processor** dialog, leave the default **MicroBlaze** option (**Figure 1-6**) and click **Next**.

**Figure 1-6. Select Processor Dialog Box**
In the **Configure Processor** dialog box (Figure 1-7), leave the default settings (see below) and click **Next**.

- Reference Clock Frequency: **50 MHz**
  - This is the external clock source on the board you are using. This clock will be used to generate the processor and bus clocks.

- Processor –bus Clock Frequency: **50 MHz**

- Debug Interface: **On-Chip H/W debug module**

- Local Data and Instruction Memory – **8 KB**

- Cache Setup: **Enable - unchecked**

![Figure 1-7. Configure Processor Dialog Box](image-url)
Select and configure the LEDs_8Bit, RS232_DCE, and DDR_SDRAM as the only external devices. Generate the memory test sample application and linker script.

- In the **Configure IO Interfaces** dialog, select and configure the **RS232_DCE**, **LEDs_8Bit** and **DDR_SDRAM** peripherals as shown below, leaving the rest of the peripherals unchecked.
  
  - **RS232_DCE**: XPS UARTLITE, 115200 baud rate, 8 Data bits, no interrupt, no parity (Figure 1-8)
  
  - **LEDs_8Bit**: XPS GPIO. No interrupt (Figure 1-9)
  
  - **DDR_SDRAM**: MPMC Controller (Multi-Port Memory Controller)

Note that the number of peripherals that appear on each window will depend on the resolution of your monitor.

---

**Figure 1-8. Configure RS-232 DCE**

**Figure 1-9. Configure XPS GPIO**

**Figure 1-10: Configure DDR_SDRAM with MPMC Controller**
Click **Next** until the *Add Internal Peripherals* dialog is displayed, making sure that none of the other devices are selected.

At this point you could click **Add Peripheral** to add additional internal peripherals, but you will see an alternative method in the next lab for adding internal peripherals to an existing project.

Click **Next** to display the **Software Setup** dialog box.

Unselect Peripheral selfTest (**Figure 1-11**) and click **Next**.

**Figure 1-11. Software Setup Dialog Box**

Leave the default selections in the **Configure Memory Test Application** dialog (**Figure 1-12**) and click **Next**.

**Figure 1-12. Configure Memory Test Application**
Verify the system summary in the **System Created** dialog (**Figure 1-13**) and click **Generate**.

![Figure 1-13. System Created Dialog Box](image)

- In the **Next Step** dialog box, ensure **Start Using Platform Studio** is checked and click **OK**.
- A Software Agreement dialog may appear if this is the first time the software is run.
- A System Assembly View will be displayed (**Figure 1-14**) showing peripherals and busses in the system, and the system connectivity.
Figure 1-14. System Created Dialog Box
Analyze the Hardware

Generate a block diagram of the system and study the system components and interconnections. Look in the System Assembly View and analyze the bus and port connections. Run PlatGen to generate the system netlists (NGC) and review the generated files.

1. Click on the **Block Diagram tab** to open a block diagram view (**Figure 1-15**) and observe the various components that are used in the design.

![Block Diagram View of the Generated Project](image)

**Figure 1-15. Block Diagram View of the Generated Project**

You can zoom in and out and use the scroll bars to navigate around the block diagram. You will see the MicroBlaze™ processor, LMB controller and PLB bus connected to the MicroBlaze processor. In addition, you will see the I/O ports on the sides and legend at the bottom of the diagram.
In the **System Assembly View** click on plus button and observe the expanded (detailed) bus connection view of the system (**Figure 1-16**)

![Click to expand and Contract](image)

**Figure 1-16. Detailed Bus Connections**

1. List the bus connection to the following peripherals:
   
   debug_module:
   dlnb_cntlr:
   RS232_DCE:

   ![Click on the Ports filter and have an expanded view similar to Figure 1-17. This is where you can make internal and external net connections.](image)
2. List the nets which are connected to the following ports:

- **RS232_DCE – RX:**
- **RS232_DCE – TX:**
- **LEDs_8Bit – GPIO_d_out:**

Click on the *Addresses* tab and have an expanded view similar to Figure 1-18. This is where you can assign base/high addresses to the peripherals in the system.
### Figure 1-18. Assign Base/High Addresses

3. Select Addresses filter and list the address for the following instances:

- **RS232_DCE** – Base address:
- **RS232_DCE** – High address:
- **LEDs_8Bit** – Base address:
- **LEDs_8Bit** – High address:
- **dlmb_cntlr** – Base address:
- **dlmb_cntlr** – High address:
- **ilmb_cntlr** – Base address:
- **ilmb_cntlr** – High address:
- **DDR_SDRAM** – Base address:
- **DDR_SDRAM** – High address:

4. Run PlatGen by selecting **Hardware → Generate Netlist** or click ![Generate Netlist](image) in the toolbar

5. Browse to the **Lab1** project directory using Windows Explorer

Several directories containing VHDL wrappers and implementation netlists have been created.

4. List the directories that were created.
Test in Hardware

Step 3

Generate bitstream and download to the board. Prior to download, the instruction memory (FPGA Block RAM) will be updated in the bitstream with the executable generated using the GNU compiler.

1. Connect and power up the Spartan-3E starter kit

2. Open a hyperterminal session (Figure 1-19)

![COM1 Properties](image)

Figure 1-19. HyperTerminal Settings

3. Select Device Configuration → Download Bitstream in XPS.

You should see the following output on hyperterminal

```
-- Entering main() --
Starting MemoryTest for DDR_SDRAM:
  Running 32-bit test...PASSED!
  Running 16-bit test...PASSED!
  Running 8-bit test...PASSED!
-- Exiting main() --
```

Figure 1-20. HyperTerminal Output
Conclusion

The Base System Builder can be used in XPS to quickly generate a MicroBlaze system and software test application. Several files—including an MHS file representing the processor system—are created. A System Assembly View, representing the hardware system, provides hardware system parameters information. After the system has been defined, the netlist of the processor system can be created. You can verify hardware operation by downloading a bitstream (configured with test application) to the FPGA.
### Answers

1. List the bus connection to the following peripherals:

   - debug_module: mb_plb
   - dlmb_cntrl: dlmb
   - RS232_DCE: mb_plb

2. List the nets which are connected to the following ports:

   - RS232_DCE – RX: fpga_0_RS232_DCE_RX
   - RS232_DCE – TX: fpga_0_RS232_DCE_TX
   - LEDs_8Bit – GPIO_d_out: fpga_0_LEDs_8Bit_GPIO_d_out

3. Select Addresses filter and list the address for the following instances:

   - RS232_DCE – Base address: 0x84000000
   - RS232_DCE – High address: 0x8400ffff
   - LEDs_8Bit – Base address: 0x81400000
   - LEDs_8Bit – High address: 0x8140ffff
   - dlmb_cntrl – Base address: 0x00000000
   - dlmb_cntrl – High address: 0x00001fff
   - ilmb_cntrl – Base address: 0x00000000
   - ilmb_cntrl – High address: 0x00001fff
   - DDR_SDRAM – Base address: 0x8c000000
   - DDR_SDRAM – High address: 0x8fffffff

4. List the directories that were created.
   - __xps
   - blkdiagram
   - data
   - etc
   - hdl
   - implementation
   - pcores
   - microblaze_0
   - synthesis
   - TestApp_Memory
Completed MHS File

# #################################################################################################################
# Created by Base System Builder Wizard for Xilinx EDK 10.1.01 Build EDK_K(SP1.3)
# Thu Jun 05 12:21:01 2008
# Target Board: Xilinx Spartan-3E Starter Board Rev D
# Family: spartan3e
# Device: XC3S500e
# Package: FG320
# Speed Grade: -4
# Processor: microblaze_0
# System clock frequency: 50.00 MHz
# On Chip Memory :  8 KB
# Total Off Chip Memory :  64 MB
# - DDR_SDRAM =  64 MB
# #################################################################################################################
PARAMETER VERSION = 2.1.0

PORT fpga_0_RS232_DCE_RX_pin = fpga_0_RS232_DCE_RX, DIR = I
PORT fpga_0_RS232_DCE_TX_pin = fpga_0_RS232_DCE_TX, DIR = O
PORT fpga_0_LEDs_8Bit_GPIO_d_out_pin = fpga_0_LEDs_8Bit_GPIO_d_out, DIR = O, VEC = [0:7]
PORT fpga_0_DDR_SDRAM_DDR_DQS_Div_1_DDR_SDRAM_DDR_DQS_Div_O = fpga_0_DDR_SDRAM_DDR_DQS_Div_1_DDR_SDRAM_DDR_DQS_Div_O, DIR = IO
PORT fpga_0_DDR_SDRAM_DDR_Clk_pin = fpga_0_DDR_SDRAM_DDR_Clk, DIR = O
PORT fpga_0_DDR_SDRAM_DDR_Clk_n_pin = fpga_0_DDR_SDRAM_DDR_Clk_n, DIR = O
PORT fpga_0_DDR_SDRAM_DDR_Addr_pin = fpga_0_DDR_SDRAM_DDR_Addr, DIR = O, VEC = [12:0]
PORT fpga_0_DDR_SDRAM_DDR_BankAddr_pin = fpga_0_DDR_SDRAM_DDR_BankAddr, DIR = O, VEC = [1:0]
PORT fpga_0_DDR_SDRAM_DDR_CAS_n_pin = fpga_0_DDR_SDRAM_DDR_CAS_n, DIR = O
PORT fpga_0_DDR_SDRAM_DDR_CE_pin = fpga_0_DDR_SDRAM_DDR_CE, DIR = O
PORT fpga_0_DDR_SDRAM_DDR_CS_n_pin = fpga_0_DDR_SDRAM_DDR_CS_n, DIR = O
PORT fpga_0_DDR_SDRAM_DDR_RAS_n_pin = fpga_0_DDR_SDRAM_DDR_RAS_n, DIR = O
PORT fpga_0_DDR_SDRAM_DDR_WE_n_pin = fpga_0_DDR_SDRAM_DDR_WE_n, DIR = O
PORT fpga_0_DDR_SDRAM_DDR_DM_pin = fpga_0_DDR_SDRAM_DDR_DM, DIR = O, VEC = [1:0]
PORT fpga_0_DDR_SDRAM_DDR_DQS = fpga_0_DDR_SDRAM_DDR_DQS, DIR = IO, VEC = [1:0]
PORT fpga_0_DDR_SDRAM_DDR_DDR_Q = fpga_0_DDR_SDRAM_DDR_DDR_Q, DIR = IO, VEC = [15:0]
PORT sys_clk_pin = dcm_clk_s, DIR = I, SIGIS = CLK, CLK_FREQ = 5000000
PORT sys_rst_pin = sys_rst_s, DIR = I, RST_POLARITY = 1, SIGIS = RST

BEGIN microblaze
PARAMETER INSTANCE = microblaze_0
PARAMETER C_INTERCONNECT = 1
PARAMETER HW_VER = 7.10.b
PARAMETER C_DEBUG_ENABLED = 1
PARAMETER C_AREA_OPTIMIZED = 1
BUS_INTERFACE DLMB = dlmb
BUS_INTERFACE ILMB = ilmb
BUS_INTERFACE DPLB = mb_plb
BUS_INTERFACE IPLB = mb_plb
BUS_INTERFACE DEBUG = microblaze_0_dbg
PORT MB_RESET = mb_reset
END
BEGIN plb_v46
PARAMETER INSTANCE = mb_plb
PARAMETER HW_VER = 1.02.a
PORT PLB_Clk = sys_clk_s
PORT SYS_Rst = sys_bus_reset
END

BEGIN lmb_v10
PARAMETER INSTANCE = ilmb
PARAMETER HW_VER = 1.00.a
PORT LMB_Clk = sys_clk_s
PORT SYS_Rst = sys_bus_reset
END

BEGIN lmb_v10
PARAMETER INSTANCE = dlmb
PARAMETER HW_VER = 1.00.a
PORT LMB_Clk = sys_clk_s
PORT SYS_Rst = sys_bus_reset
END

BEGIN lmb_bram_if_cntlr
PARAMETER INSTANCE = dlmb_cntlr
PARAMETER HW_VER = 2.10.a
PARAMETER C_BASEADDR = 0x00000000
PARAMETER C_HIGHADDR = 0x00001fff
BUS_INTERFACE SLMB = dlmb
BUS_INTERFACE BRAM_PORT = dlmb_port
END

BEGIN lmb_bram_if_cntlr
PARAMETER INSTANCE = ilmb_cntlr
PARAMETER HW_VER = 2.10.a
PARAMETER C_BASEADDR = 0x00000000
PARAMETER C_HIGHADDR = 0x00001fff
BUS_INTERFACE SLMB = ilmb
BUS_INTERFACE BRAM_PORT = ilmb_port
END

BEGIN bram_block
PARAMETER INSTANCE = lmb_bram
PARAMETER HW_VER = 1.00.a
BUS_INTERFACE PORTA = ilmb_port
BUS_INTERFACE PORTB = dlmb_port
END

BEGIN xps_uartlite
PARAMETER INSTANCE = RS232_DCE
PARAMETER HW_VER = 1.00.a
PARAMETER C_BAUDRATE = 115200
PARAMETER C_DATA_BITS = 8
PARAMETER C_ODD_PARITY = 0
PARAMETER C_USE_PARITY = 0
PARAMETER C_SPLB_CLK_FREQ_HZ = 50000000
PARAMETER C_BASEADDR = 0x84000000
PARAMETER C_HIGHADDR = 0x8400ffff
BUS_INTERFACE SPLB = mb_plb
PORT RX = fpga_0_RS232_DCE_RX
PORT TX = fpga_0_RS232_DCE_TX
END

BEGIN xps_gpio
PARAMETER INSTANCE = LEDs_8Bit
PARAMETER HW_VER = 1.00.a
PARAMETER C_GPIO_WIDTH = 8
PARAMETER C_IS_DUAL = 0
PARAMETER C_IS_BIDIR = 0
PARAMETER C_ALL_INPUTS = 0
PARAMETER C_BASEADDR = 0x81400000
PARAMETER C_HIGHADDR = 0x8140ffff
BUS_INTERFACE SPLB = mb_plb
PORT GPIO_d_out = fpga_0_LEDs_8Bit_GPIO_d_out
END

BEGIN mpmc
PARAMETER INSTANCE = DDR_SDRAM
PARAMETER HW_VER = 4.01.a
PARAMETER C_NUM_PORTS = 1
PARAMETER C_MEM_PARTNO = MT46V32M16-6
PARAMETER C_SPECIAL_BOARD = S3E_STKIT
PARAMETER C_MEM_BANKADDR_WIDTH = 2
PARAMETER C_MEM_DATA_WIDTH = 16
PARAMETER C_MEM_DQS_WIDTH = 2
PARAMETER C_MEM_DM_WIDTH = 2
PARAMETER C_MEM_TYPE = DDR
PARAMETER C_PIM0_BASETYPE = 2
PARAMETER C_MPMC_CLK0_PERIOD_PS = 10000
PARAMETER C_MPMC_BASEADDR = 0x8c000000
PARAMETER C_MPMC_HIGHADDR = 0x8fffffff
PARAMETER C_SPLB0_NATIVE_DWIDTH = 32
BUS_INTERFACE SPLB0 = mb_plb
PORT DDR_Clk = fpga_0_DDR_SDRAM_DDR_Clk
PORT DDR_Clk_n = fpga_0_DDR_SDRAM_DDR_Clk_n
PORT DDR_Addr = fpga_0_DDR_SDRAM_DDR_Addr
PORT DDR_BankAddr = fpga_0_DDR_SDRAM_DDR_BankAddr
PORT DDR_CAS_n = fpga_0_DDR_SDRAM_DDR_CAS_n
PORT DDR_CE = fpga_0_DDR_SDRAM_DDR_CE
PORT DDR_CS_n = fpga_0_DDR_SDRAM_DDR_CS_n
PORT DDR_RAS_n = fpga_0_DDR_SDRAM_DDR_RAS_n
PORT DDR_WE_n = fpga_0_DDR_SDRAM_DDR_WE_n
PORT DDR_DM = fpga_0_DDR_SDRAM_DDR_DM
PORT DDR_DQS = fpga_0_DDR_SDRAM_DDR_DQS
PORT DDR_DQ = fpga_0_DDR_SDRAM_DDR_DQ
PORT DDR_DQS_Div_O = fpga_0_DDR_SDRAM_DDR_DQS_Div_O
PORT DDR_DQS_Div_I = fpga_0_DDR_SDRAM_DDR_DQS_Div_I
PORT MPMC_Clk0 = DDR_SDRAM_mpmc_clk_s
PORT MPMC_Clk90 = DDR_SDRAM_mpmc_clk_90_s
PORT MPMC_Rst = sys_periph_reset
END
BEGIN clock_generator
PARAMETER INSTANCE = clock_generator_0
PARAMETER HW_VER = 2.01.a
PARAMETER C_EXT_RESET_HIGH = 1
PARAMETER C_CLKIN_FREQ = 50000000
PARAMETER C_CLKOUT0_FREQ = 50000000
PARAMETER C_CLKOUT0_BUF = TRUE
PARAMETER C_CLKOUT0_PHASE = 0
PARAMETER C_CLKOUT0_GROUP = NONE
PARAMETER C_CLKOUT1_FREQ = 100000000
PARAMETER C_CLKOUT1_BUF = TRUE
PARAMETER C_CLKOUT1_PHASE = 0
PARAMETER C_CLKOUT1_GROUP = DCM0
PARAMETER C_CLKOUT2_FREQ = 100000000
PARAMETER C_CLKOUT2_BUF = TRUE
PARAMETER C_CLKOUT2_PHASE = 90
PARAMETER C_CLKOUT2_GROUP = DCM0
PORT CLKOUT0 = sys_clk_s
PORT CLKOUT1 = DDR_SDRAM_mpmc_clk_s
PORT CLKOUT2 = DDR_SDRAM_mpmc_clk_90_s
PORT CLKIN = dcm_clk_s
PORT LOCKED = Dcm_all_locked
PORT RST = net_gnd
END

BEGIN mdm
PARAMETER INSTANCE = debug_module
PARAMETER HW_VER = 1.00.b
PARAMETER C_MB_DBG_PORTS = 1
PARAMETER C_USE_UART = 1
PARAMETER C_UART_WIDTH = 8
PARAMETER C_BASEADDR = 0x84400000
PARAMETER C_HIGHADDR = 0x8440ffff
BUS_INTERFACE SPLB = mb_plb
BUS_INTERFACE MBDEBUG_0 = microblaze_0_dbg
PORT Debug_SYS_Rst = Debug_SYS_Rst
END

BEGIN proc_sys_reset
PARAMETER INSTANCE = proc_sys_reset_0
PARAMETER HW_VER = 2.00.a
PARAMETER C_EXT_RESET_HIGH = 1
PORT Slowest_sync_clk = sys_clk_s
PORT Dcm_locked = Dcm_all_locked
PORT Ext_Reset_In = sys_rst_s
PORT MB_Reset = mb_reset
PORT Bus_Struct_Reset = sys_bus_reset
PORT MB_Debug_SYS_Rst = Debug_SYS_Rst
PORT Peripheral_Reset = sys_periph_reset
END