

EDK 6.2 System Design Example

Web Server Running on MicroBlaze

Introduction

This document details an Embedded System Example Design – a web server running on the MicroBlaze soft processor – designed using the Embedded Development Kit. It will also explain how to set up your system to become a web client and connect to the web server running on MicroBlaze.

Hardware Requirements:

Memec V2P4/7 Development Board with P160 Communications module (Rev 1)
[See notes at the end of this document for information on porting this design to the Memec V2P4/7 Rev 2, Rev 3, or Rev 4 boards]
JTAG Parallel 4 cable
Cross-over Ethernet Cable

Software Requirements:

Embedded Development Kit 6.2 SP2
ISE 6.2i SP2 or greater
NetScape, Mozilla, Internet Explorer

Design Description

This system contains a MicroBlaze connected to 8K of BRAM memory over the Local Memory Bus (LMB); a UARTLite, an external SRAM memory interface, an Ethernet 10/100 MAC and a GPIO are connected to the On-chip Peripheral Bus (OPB). The General Purpose I/O (GPIO) are connected to the four LEDs and to the 4 DIP switches (Dip1, Dip2, Dip3 and Dip4) on the board. There is a Fast Simplex Link interface that provides a fast connection from the OPB MicroBlaze Debug Module (MDM) to the MicroBlaze processor. This download mechanism supports file download speeds up to 500 KB/s. In the xmd.ini script, the ELF file is downloaded using this interface.

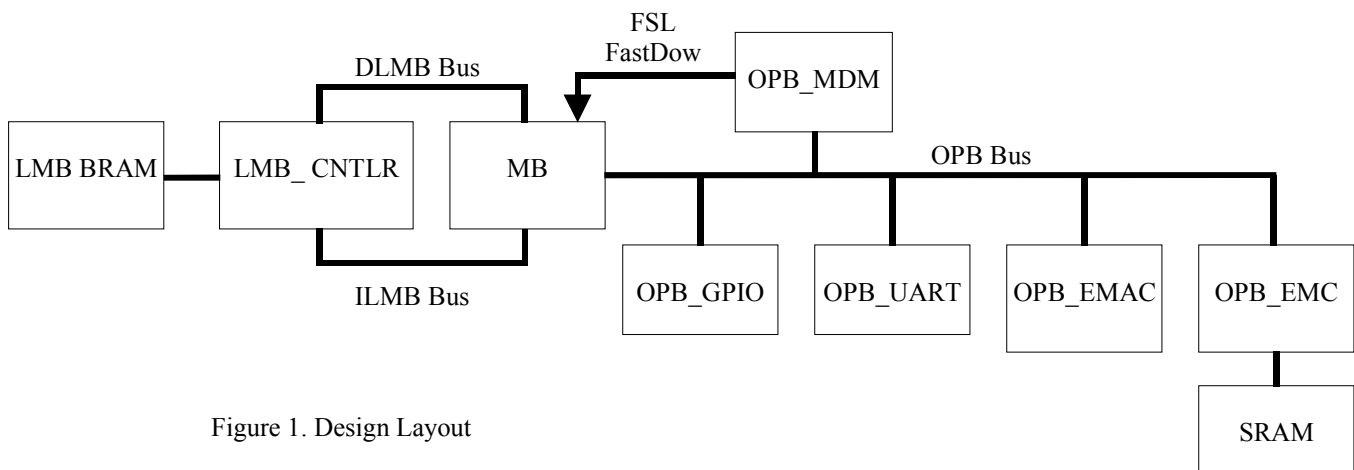


Figure 1. Design Layout

The web server source code is located in the /code directory. The LibXil Networking and File System libraries accessed by the web server design are included by libgen.

The web server is running HTTP 1.0 on this system and a file system (built using the LibXil Memory File System library) is used to store files. The server listens at port 8080 for requests. Every request is processed and replies are served up by the server to the client.

Different demos are present in this system. These are:

- Hex Digit display in LEDs - Hex digit typed in the web page shown, gets displayed as a 4 digit binary in LEDs on reload of page
- Changing colors - Background color of the pages keeps changing on every reload of web pages
- DIP switches - Set the dip switches 1, 2, 3, 4 and see it displayed as a binary value on reload of page

Here is the memory map for this design:

Device	Address		Size	Comment
	Min	Max		
LMB BRAM	0x0000_0000	0x0000_1FFF	8kB	Local Memory
OPB GPIO	0xFFFF_0100	0xFFFF_01FF	256B	DIP Switch Input
OPB_UARTLITE	0xFFFF_0200	0xFFFF_02FF	256B	Serial I/O
OPB_MDM	0xFFFF_0300	0xFFFF_03FF	256B	Debug Peripheral
OPB_EMAC	0x8000_0000	0x8000_3FFF	16kB	Ethernet MAC
OPB EMC	0x80F0_0000	0x80FF_FFFF	1MB	External SRAM Memory

Figure 2 Memory Map

Note on licensing: The 10/100 Ethernet MAC OPB peripheral is not a free core and an evaluation license is required to use the core. The evaluation version of the core includes built-in timeout circuitry that will disable the core after a period a time. Beginning in EDK 3.2, evaluation copies of the licenses are included in the EDK.

Setting Up the Web Server Design

- Unzip the included design files.
- Open XPS (**Start->Programs->Xilinx Embedded Development Kit->Xilinx Platform Studio**)
- In XPS, select **File->Open Project**.
- **Browse to the system.xmp file in the memec_v2p/microblaze directory. Open the system.xmp file with XPS.**

Generating the Netlist and Implementing the Design

- Generate the netlist by selecting **Tools > Generate Netlist** from within XPS. You will need the evaluation license for the 10/100 Ethernet MAC core in order to generate the netlist for this system.
- Implement the design by selecting **Tools > Generate Bitstream** from within XPS. The UCF file for the Memec V2P4 board is included in the /data directory. If XPS is used to implement the system, XPS will use the UCF constraints file residing in that directory. Additionally, the XPS implementation flow will use the OPT option file included in the /etc directory.

Configuring the Web Server

The Ethernet MAC requires IP and MAC addresses to be initialized. They must be set in the web server code. In the demo included, the **IP address must be set** in order for the web server to work properly.

- Open the source file **web.c** by double-clicking **code/web.c** located under the Sources in XPS.
- **Browse to line containing “xilnet_ip_init(“1.2.3.4”);”**;
- **You can leave this set to “1.2.3.4” (this will require you to change your host’s IP address) or change the arguments of the xilnet_ip_init function call to match the local IP address of your host.**
- Save changes to **web.c**, if any.
- Select **Tools > Compile Program Sources** to compile the web server design.

Downloading the Web Server Code and Configuring the Web Client

Once the web server code web.c has been updated with a valid IP value, the code must be compiled and the program can be updated.

- Select **Tools > Update Bitstream and Download**. (iMPACT needs to be closed to program the device via XPS.)
- While the bitstream is getting updated, unplug any ethernet cable you may have and **connect the crossover ethernet cable to your machine** and to the **P160 Ethernet module**.
- Make sure that the JTAG Parallel 4 cable is connected from your machine to the Memec V2P4 board.
- **Modify your host IP address** so that it is in the same subnet as the web server. To change your PC’s IP address, select **Start > Settings > Control Panel**.
- Double-Click on **Networking and Dialup**.
- Select your **LAN connection**, right-click and select Properties.
- LAN settings window similar to the one below in Figure 3 should appear.

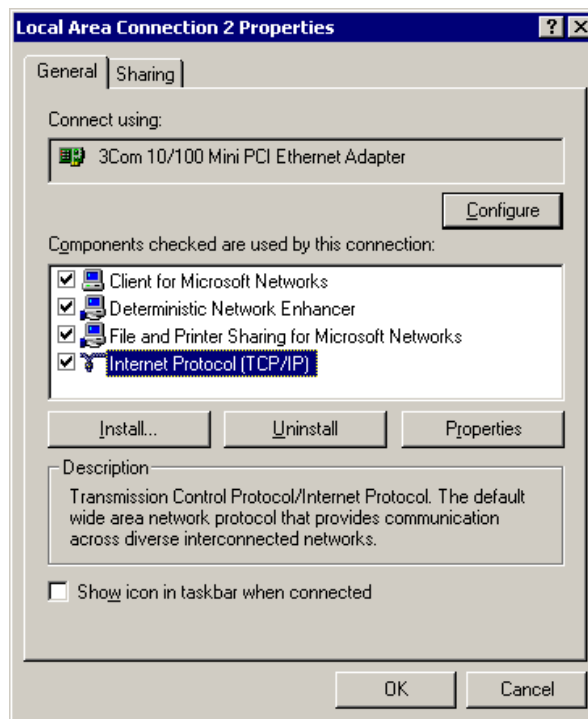


Figure 3. LAN settings

- Highlight Internet Protocol and select **Properties**.
- Change your PC's IP address by selecting the **Use the following IP Address** Radio button.
- Select a **unique IP address** that is in the **same subnet** as the one specified for the eb server. (If you changed the above code to match your host IP, then you won't have to make any changes.) For example, if the IP address you selected for the web server was 1.2.3.4, you should select an IP address of 1.2.3.9.

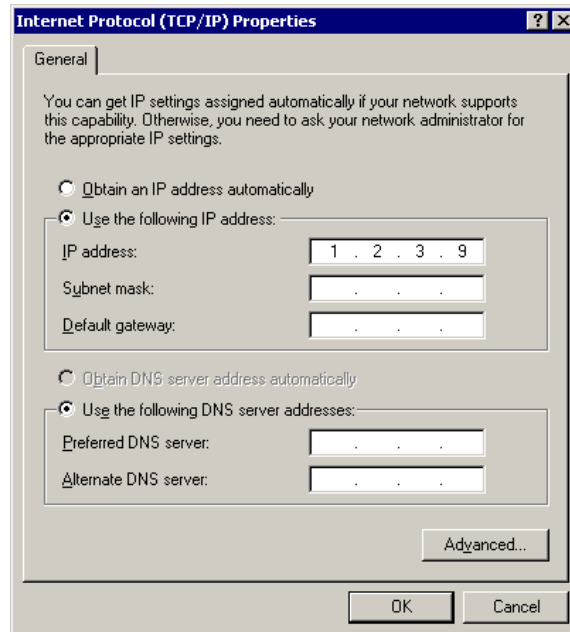


Figure 4. Internet Protocol (TCP/IP) Properties

- After you've added a valid IP address, **Select OK** at the IP Properties window
- If you are prompted that a subnet mask is missing, **Select OK**. **Select OK** again and again.
- At this point in XPS, the code should have already been recompiled and the bitstream should have been updated and downloaded to the board by XPS. Check in XPS to make sure that iMPACT has run and the device is configured. The DONE light should be illuminated on the board.
- In XPS, Run **Tools > XMD**. This will load the application code via the MicroBlaze Debug Module (MDM) through XMD. In the XMD console, you should see "Web Server Running", as shown below in Figure 5.

```

c:\EDK_Gm\bin\nt\xmd.exe
Xilinx Microprocessor Debug (XMD) Engine
Xilinx EDK 6.2.2 Build EDK_Gm.13.6
Copyright (c) 1995-2002 Xilinx, Inc. All rights reserved.

Load MHS file <xload mhs system.mhs> for MicroBlaze FastDow

Processors in System (C:\working\Gm_webserver\memec_v2p\SP2\MicroBlaze\system.mh
s)::
    Microblaze-MDM(1) : mblaze

JTAG chain configuration
-----
Device   ID Code      IR Length  Part Name
1        0123e093      10         XC2UP4
Assuming, Device No: 1 contains the MicroBlaze system
Connected to the JTAG MicroBlaze Debug Module (MDM)
No of processors = 1

Connecting to XMD stub..
XMD communication stub initialized. Version No: 2
Communicating with XMD stub on target board
JTAG Uart opened Successfully
Connected to MicroBlaze "stub" target. id = 0
Starting GDB server for "stub" target (id = 0) at TCP port no 1234

Downloading Web Server Using MicroBlaze FastDow

Program Executable Instruction Section(s):
Section(0): 0x80f00000-0x80f0650c
WARNING:: Unable to Check Memory for MB with Stub
Either MHS/MSS is not loaded or MSS does not contain Microblaze with Xmd
Stub Peripheral
WARNING:: Unable to Check Memory for MB 0
Program Data Memory Section(s):
Section(0): 0x80f0650c-0x80f06e7e
Section(1): 0x80f06e80-0x80f070d0
Section(2): 0x80f070d0-0x80f074b4
Section(3): 0x80f074b8-0x80f08828
WARNING:: Unable to Check Memory for MB with Stub
Either MHS/MSS is not loaded or MSS does not contain Microblaze with Xmd
Stub Peripheral
WARNING:: Unable to Check Memory for MB 0
Web Server Running ...

```

Figure 5. XMD “Web Server Running”

Note: XMD is the debug engine for embedded systems. It includes a TCL environment that allows users to fully create custom debug tools. After launching XMD, it will source xmd.ini if it is present in the EDK project directory. The xmd.ini file contains a list a TCL commands to be run every time XMD is run.

- Open an HTML browser. Make sure that your browser's internet settings should be : "Direct Connection to the Internet" and point to the URL: <http://1.2.3.4:8080> where '1.2.3.4' is the IP you specified for the web server.
- The webserver demo should appear in your browser. Follow the instructions to read the dip switch values and write to the LEDs.

Porting Design to Memec 2VP4 Rev 2, Rev 3 or Rev 4

- UCF Change:

```
NET "ETH_RXD<2>"      LOC = "G20";    #LIOA15, Rev1
#NET "ETH_RXD<2>"      LOC = "F11";    #LIOA15, Rev2,3,4
NET "ETH_MDIO" LOC = "H19";          #LIOA13, Rev1
#NET "ETH_MDIO" LOC = "E11";          #LIOA13, Rev2,3,4
```