

## Overview

This demonstration project describes how to create an embedded system and to utilize the lwIP to develop a Web Server application. This project was built on a Xilinx Spartan-6 device contained by the Digilent Atlys™ development board, using the Lightweight IP (lwIP) TCP/IP networking stack. The Xilinx® Software Development Kit provides lwIP software customized to run on Xilinx embedded systems containing either a PowerPC® or MicroBlaze™ processor.

This project is also an example of managing files by using the LibXil Memory File System.

## Package description

The “Atlys\_AXI\_Web\_Server\_Demo\_v\_1\_02” package contains five folders with the following description:

- **doc** – contains the current document.
- **memfs** – the contents of the MFS image. The image is generated using the *mfsgen* utility. Documentation on the LibXil Memory File System and *mfsgen* can be found in the Xilinx document titled *OS and Libraries Document Collection* (UG643).
- **ready\_for\_download** – the files needed for a pre-built design: the executable software file (*Atlys\_Webserver\_Demo.elf*) and the MFS image (*image.mfs*).
- **sw** – the software application files needed for Xilinx SDK.
- **hw** – the hardware design project created using Xilinx Platform Studio (XPS).

## Hardware and Software Requirements

- Digilent Atlys Development Board
- Two USB A to USB micro B cables (one for UART and one for programming)
- One Ethernet cable, for connecting the board to a Windows or Linux host.
- Terminal emulator application, such as AccessPort or HyperTerminal.
- Xilinx Platform Studio 14.3\*, for making the hardware platform
- Xilinx SDK 14.3\*, for running or making modifications to the software

**\*Note:** Newer versions of Xilinx EDK will likely work, but the procedures below may differ slightly

## Network Setup

1. An Ethernet Adapter on your PC is required.
2. In the Network Connections, in Control Panel, right click on your Ethernet Adapter and select Properties.
3. Then go to the properties of “Internet Protocol (TCP/IP)” and set the following:

IP address:	192.168.1.11
Subnet mask:	255.255.255.0
Default gateway:	192.168.1.1

You can leave the DNS addresses blank.

4. Your Ethernet adapter must be configured to auto-negotiate the communication speed. The procedure for doing this varies for different adapters, but typically it can be done from the Windows device manager by right-clicking your network adapter and selecting "Properties". You should then be able to set the link speed to "Auto negotiation" somewhere within the dialog box that opens. This is often the default setting, meaning that it is not likely any action will be required.

## Atlys Setup

1. Connect the PROG and UART ports to the PC using the two USB Micro cables.
2. Connect an Ethernet cable from the Atlys to the PC.
3. Connect a 5V power supply to J11 and power on the Atlys.

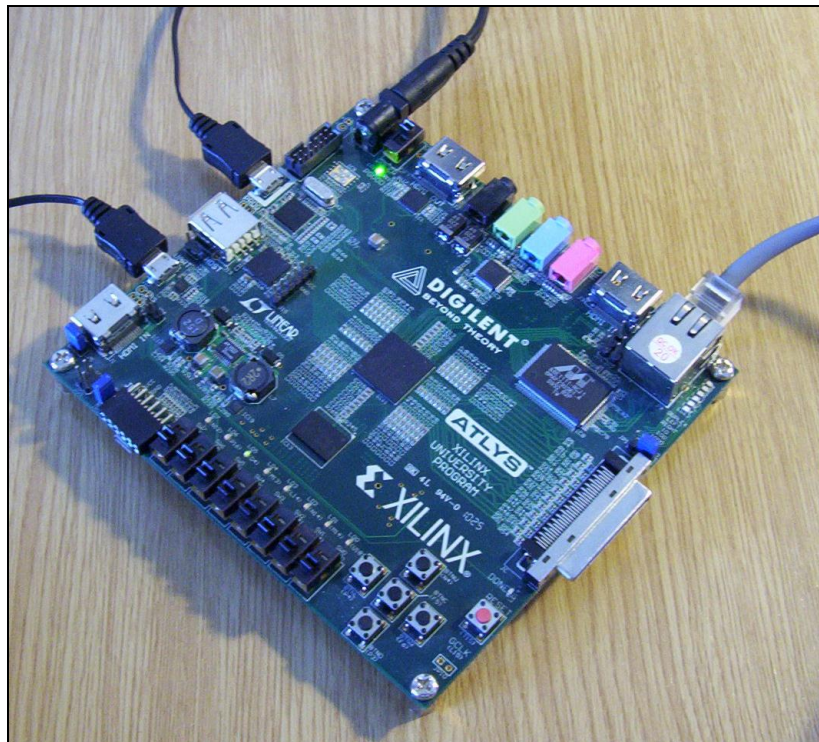


Figure 1. Atlys board setup

4. Open up the terminal emulator application on the attached computer. Connect to the COM port corresponding to the Atlys UART using a 9600 Baud, 8 data bits, 1 stop bit, and no parity.

## Using the Pre-Built design

If you wish to use the already built Hardware Platform then please jump to the [Compiling and Running the Software](#) section of this document.

## Creating the Hardware Platform

1. Make sure that this package has been placed in a location that contains no spaces in the file path. Xilinx Platform Studio (XPS) does not allow projects to be built from locations containing spaces.

2. Open **hw/system.xmp** in XPS.
3. If using a version of XPS newer than 14.3, allow the tools to backup and migrate the project. Follow the wizard that opens, using the default selections.
4. Once the project has opened, click **Project** → **Export Hardware Design to SDK....**
5. Ensure “Include bitstream and BMM file” is checked, and then click **Export & Launch SDK**.
6. Wait for the build to complete (this can take ~30 minutes, depending on your system). Once completed, SDK will automatically open.
7. In the “Project Explorer” pane, right-click the hardware platform project (it will have a name ending in **hw\_platform**) and select **Rename....** Change the name to **hw\_platform**. This is required for the projects to import correctly in the next section.

## Compiling and Running the Software

1. Open Xilinx SDK and create a new Workspace in an empty folder. Skip this step if you completed the previous section and SDK is already open.
2. From the top menu click **File** → **Import....** Select **General** → **Existing Projects into Workspace** and click Next.
3. In the “Select root directory” option browse to the **sw** directory in the downloaded package. If you created the Hardware Platform from scratch then unselect “hw\_platform”. Check the “Copy projects into workspace” option and click Finish. The project should try to build automatically, but result in several errors because the project is not correctly pointed to the BSP.

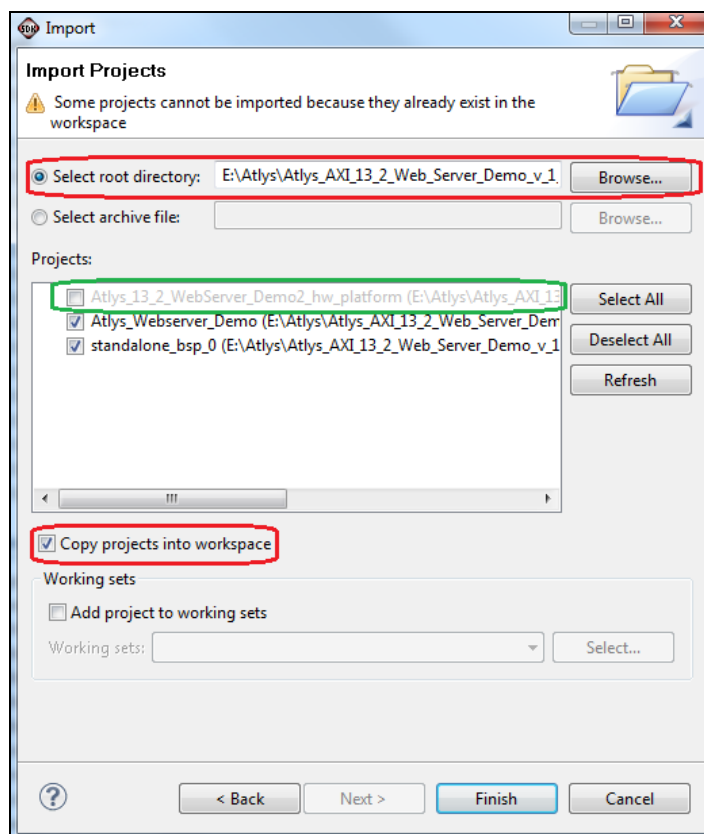


Figure 2. Importing the Software Application

4. Under the “Project Explorer” pane, right-click “Atlys\_Webserver\_Demo” and select **Change Referenced BSP**. Select “standalone\_bsp\_0” and press OK. The project should automatically build again. If the project still does not build successfully, then clean the project by clicking **Project** → **Clean...** and selecting “Clean all projects”. The project should then automatically clean and rebuild successfully.
5. From the top menu of SDK click on **Xilinx Tools** → **Program FPGA** → **Program**. Once complete, the DONE LED on the Atlys should illuminate.
6. Click on **Xilinx Tools** → **XMD Console** which will open the XMD console. Execute the following commands:

- a. Connect Microblaze to the debugging module:

```
connect mb mdm
```

- b. Change to the **ready\_for\_download** folder using the `cd` command
- c. Download the MFS image:

```
dow -data image.mfs 0xC5000000
```

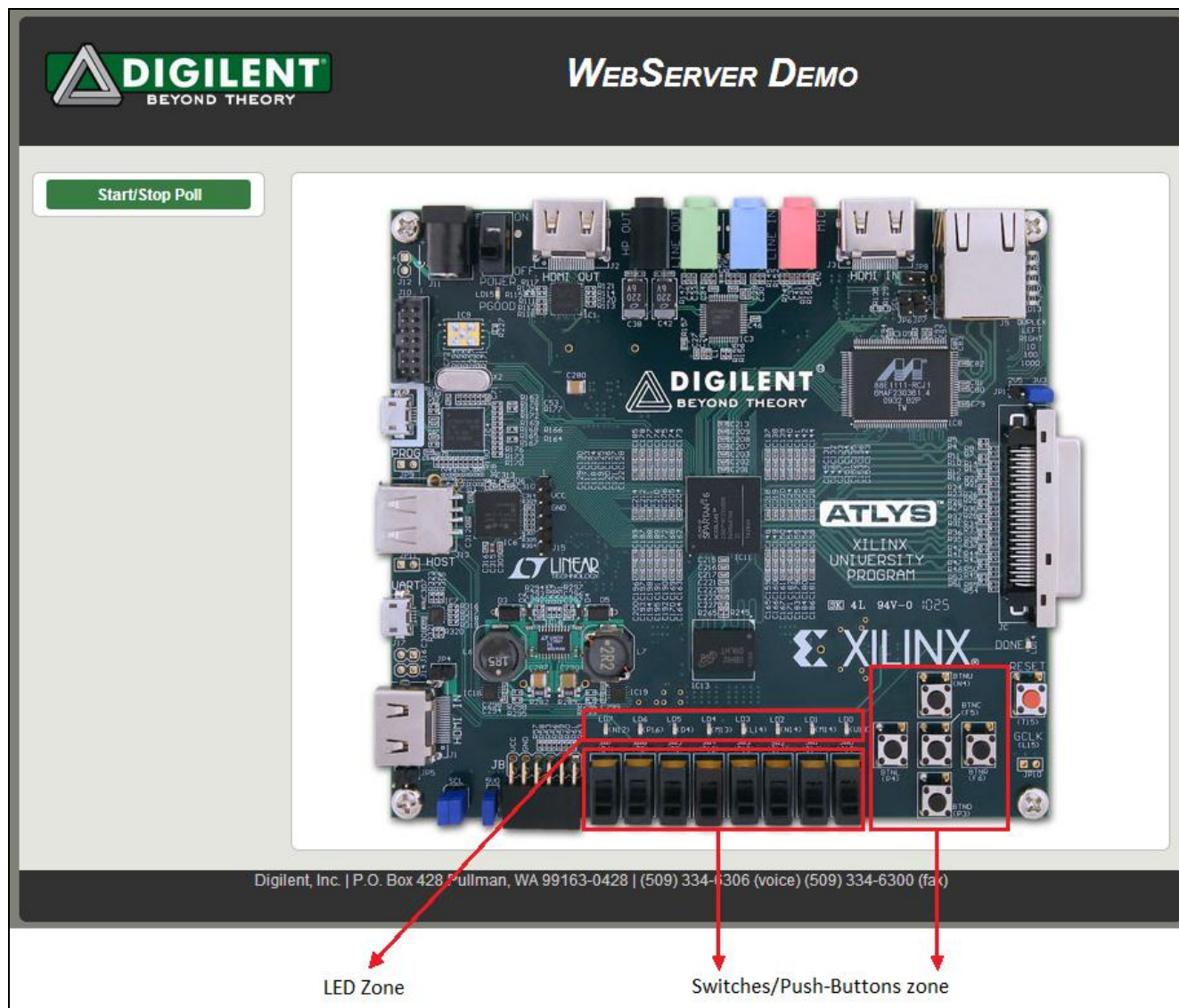
This will take a couple minutes. Once completed, Done will be printed to the XMD console.

7. From the top menu select **Run** → **Run Configuration...**
8. Create a new run configuration by right clicking the “Xilinx C/C++ ELF” tab on the left pane.
9. In the “Project” region, browse to the *Atlys\_Webserver\_Demo* project.
10. In the “C/C++ Application” region, Click “Search Project...” and select *Atlys\_Webserver\_Demo.elf*.
11. Click **Apply**, and then **Run** to run the executable. Once complete, you should begin to see feedback at the attached terminal.

## Interacting with the Web Server

On the attached computer, open a web browser and point it to the following IP address: **192.168.1.10**. The following Web Page should appear:





**Figure 3. Web Server demonstration web site**

It has the following regions:

- The LED zone* – by clicking over one of the LED's, the corresponding LED will light up on the Atlys board.
- The switches/push-buttons zone* – by modifying one of the switch's/push-button's state, the corresponding switch/push-button will also light up on the Web page.

## Using the Switches and Push-Buttons

- Click once on the **Start/Stop Poll** button to start the polling of the Switches/Push-Buttons.
- Change the state of the Switches on the Atlys board; press the Push-Buttons (for at least 500 ms).
- Click again on the **Start/Stop Poll** button to stop polling GPIO's.