ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Achronix SPD-DEV-BRD-SPD60* at our website: ***Click HERE***

Bitporter User Guide

UG004 v1.1 – March 20, 2009

achronix
SEMICONDUCTOR CORPORATION

Copyright Info

Copyright © 2008–2009 Achronix Semiconductor Corporation. All rights reserved. Achronix, Speedster, and picoPIPE are trademarks of Achronix Semiconductor Corporation. All other trademarks are the property of their prospective owners. All specifications subject to change without notice.

NOTICE of DISCLAIMER: The information given in this document is believed to be accurate and reliable. However, Achronix Semiconductor Corporation does not give any representations or warranties as to the completeness or accuracy of such information and shall have no liability for the use of the information contained herein. Achronix Semiconductor Corporation reserves the right to make changes to this document and the information contained herein at any time and without notice. All Achronix trademarks, registered trademarks, disclaimers and patents are listed at http://www.achronix.com/legal.

Achronix Semiconductor Corporation

333 West San Carlos Street, Suite 1050
San Jose, CA, 95110
USA

Phone : 877.GHZ.FPGA (877.449.3742)
Fax : 408.286.3645
E-mail : info@achronix.com

Table of Contents

Preface ---------- iv

About this Guide ---------- iv

Reference Documents ---------- iv

Conventions Used in this Guide ---------- iv

Chapter 1 – Configuration Overview ---------- 1

Introduction ---------- 1

JTAG Configuration using the Bitporter Pod ---------- 2

Supported Operating Systems ---------- 2

Minimum Hardware Requirements ---------- 3

Board-Level Device Connections ---------- 3

Chapter 2 – Software and Driver Install ---------- 5

Introduction ---------- 5

Standalone STAPL Player and USB Driver Install ---------- 5

Windows ---------- 5

Linux ---------- 8

Chapter 3 – Connecting the Bitporter Pod ---------- 9

Introduction ---------- 9

Choosing a Connection Type — USB versus Ethernet ---------- 9

Connecting the Bitporter Pod ---------- 9

Connecting the Bitporter Pod via Ethernet (Windows and Linux) ---------- 13

Disconnecting the Bitporter Pod ---------- 14

Bitporter Connection Errors ---------- 14

"Attempting to auto-detect Bitporter pods (USB pods and Ethernet pods on this sub net)... found 0 pods." ---------- 14

"PROGRAM ERROR: Bitporter driver library not found - unable to program device." ---------- 14

"Port has not been opened" ---------- 15

"Unable to retrieve pathname for USB device" ---------- 15

"USB communication error" ---------- 15

"USB communication timeout" ---------- 15

"Unable to load hid.dll. One or more exported functions missing" ---------- 15

"Can't open setupapi.dll: <reason>" ---------- 15

"Can't open hid.dll: <reason>" ---------- 15

“USB read error. <reason>” - 15
“USB write error. <reason>” - 15
"No available Achronix Bitporter products with matching serial number found on USB port." 15
Ethernet Connection Errors: - 16
"No available Achronix Bitporter products with matching serial number found on local subnet." - 16
"No available Achronix Bitporter product found at supplied network address." - 16
"Network device is not an Achronix Bitporter pod." - 16
Verifying the Setup - 16
Bitporter Connectivity Self Test - 16
Bitporter-to-Target-Device Connectivity Test - 17
Handling Multiple Pods Connected to the Same PC - 17

Chapter 4 – Using the Achronix STAPL Player - 19

Introduction - 19
Command Syntax - 19
Bitporter Pod Naming Conventions - 20
Picking a STAPL Action (-a option) - 20
Procedures - 21
The Action read_idcode - 22
The Action program - 22
The Action verify - 23
The Action read - 23
The Action read_state - 25
The Action erase - 27
Querying the Availability of Connected Pods (-q option) - 28
Connecting to Specific Pods by Name (-p option) - 29
Configuring the Bitporter Pod's IP address (-i Option) - 30
Bitporter Pod MAC Addresses - 30
Checking the Bitporter Pod's Current Ethernet IP Configuration - 30
Configuration the Pod for DHCP (Dynamic IP Address) - 31
Configuring the Pod to Use a Static IP Address - 32
Programming a Device - 33

Revision History - 35

Preface

About this Guide

This guide provides details about device programming within the Achronix CAD Environment (ACE), starting with an existing bitstream through to downloading and device configuration.

This guide consists of the following chapters:

Chapter 1 – "Configuration Overview" provides an overview of configuring Achronix FPGAs.

Chapter 2 – "Software and Driver Install" details the installation of the Bitporter USB drivers and the Achronix STAPL player.

Chapter 3 – "Connecting the Bitporter Pod" covers connecting the pod to a target device and host PC.

Chapter 4 – "Using the Achronix STAPL Player" covers the syntax and usage of the STAPL player.

Reference Documents

ACE User Guide (UG001)

ACE Installation and Licensing Guide (UG002)

ACE Quick Start Guide (UG003)

EIA/JEDEC Standard 71 (JESD71), *Standard Test and Programming Language (STAPL)*

Conventions Used in this Guide

Item	Format	Examples
Command-line entries	Formatted with the Courier bold font face.	`$ Open top_level_name.log`
File Names	Formatted with the Courier font face.	`filename.ext`
GUI buttons, menus and radio buttons	Formatted with the Helvetica bold font face.	Click **OK** to continue. **File → Open**
Variables	Formatted with italic emphasis.	*`design_dir`*`/output.log`
Window and dialog box headings and sub-headings	Heading formatted in quotation marks.	Under "Output Files," select ...
Window and dialog box names	Initial caps.	From the Add Files dialog box, ...

Chapter 1 – Configuration Overview

Introduction

Achronix Semiconductor delivers the world's fastest commercially available FPGAs. Using CMOS process technology and conventional SRAM-based device configuration, Speedster™ FPGAs are capable of operating at 1.5 GHz of system throughput by using unique picoPIPE™ technology.

The embedded programming and configuration logic in the FPGA is designed to support a variety of programming options. **Figure 1-1** outlines the basic block diagram of the programming and configuration logic, including additional logic to implement security features. The configuration management unit controls the startup and shutdown sequence from configuration mode to the user mode and back. The configuration management unit includes the provisions for configuring the device with a secure bitstream using a 256-bit Advanced Encryption Standard (AES) algorithm in Cipher Block Chaining (CBC) mode. The device contains a small non-volatile memory for the storage of the required AES key.

This guide covers configuration via the JTAG port using the Achronix Bitporter pod.

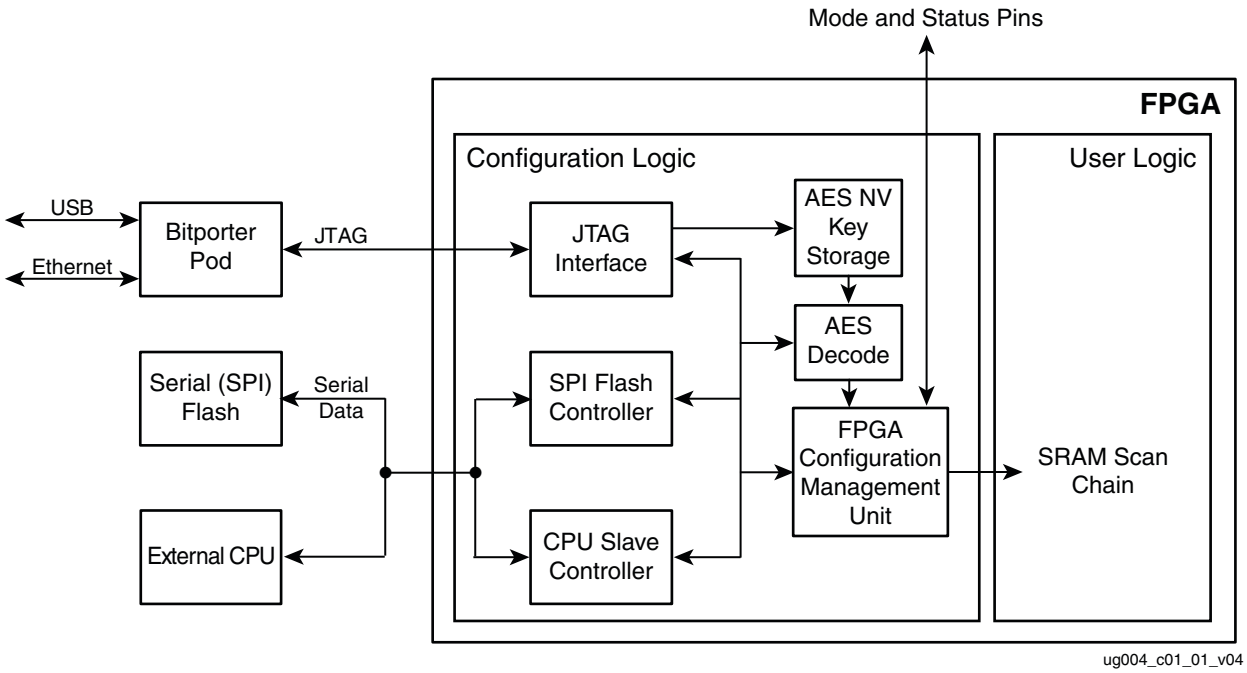


Figure 1-1: *Speedster Configuration Options*

JTAG Configuration using the Bitporter Pod

The Bitporter pod (**Figure 1-2**) connects between a host PC via either a 10/100 Ethernet or USB 2.0 connection and an EJTAG-compliant connector on the target system. When connected, the Bitporter pod supports device configuration and debug, along with flash memory programming.

Note: *USB 1.0 and 1.1 are also supported, but discouraged for performance reasons.*



ug004_c01_02_v01

Figure 1-2: *Bitporter Pod*

The JTAG configuration flow is as follows:

- Generate a *design_name*.jam file from a placed-and-routed design within ACE.
- Connect the Bitporter pod to either the USB or Ethernet port of the host PC and to the JTAG port of the target Speedster device.
- *Play* the STAPL file to the FPGA using acx_stapl_player.exe, executed from the command line, or via the Download view within ACE (see "Playing a STAPL File" in *ACE User Guide* (UG001) for details).

Supported Operating Systems

JTAG programming is currently supported under the following operating systems:

- For acx_stapl_player with USB-connected Bitporter pods:
  - Red Hat Enterprise Linux Release 5, 32-bit
  - CentOS 5, 32-bit
  - Microsoft Windows XP Professional SP2 32-bit (other versions may work, but are untested)

- For `acx_stapl_player` with Ethernet-connected Bitporter pods:
  - Red Hat Enterprise Linux Release 5, 32-bit
  - Red Hat Enterprise Linux WS Release 4, 32-bit
  - CentOS 5, 32-bit
  - Microsoft Windows XP Professional SP2 32-bit (other versions may work, but are untested)

Note: *Other 32-bit versions of Linux may work, but are unsupported. No 64-bit operating systems are supported at this time.*

Minimum Hardware Requirements

- Pentium-class PC with a minimum of 128 MB of memory (512 MB for Windows Vista)
- Either a USB 2.0 port and/or an Ethernet connection for the Bitporter pod

Note: *USB 1.0 and 1.1 may be used but USB 2.0 is strongly recommended for performance reasons. In addition, a USB port connection is required to change the Ethernet configuration of a Bitporter pod (Bitporter pods are configured for DHCP by default — if this configuration is acceptable, no USB connection is needed).*

Board-Level Device Connections

Figure 1-3 details the board-level electrical connections to the JTAG header, and **Figure 1-4** provides the mechanical specifications.

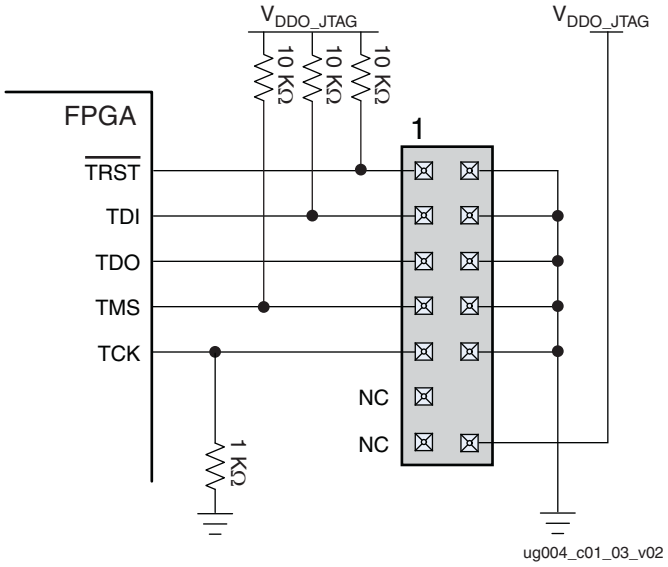


Figure 1-3: *JTAG Header Electrical Connections*

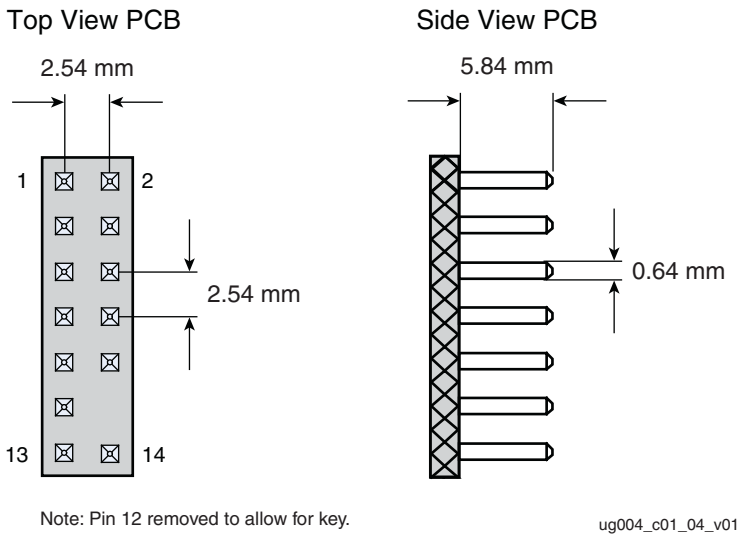


Figure 1-4: *JTAG Header Mechanical Specifications*

Chapter 2 – Software and Driver Install

Introduction

Prior to device configuration, both the STAPL player (`acx_stapl_player`) and, if required, the USB drivers for the Bitporter pod must be installed on the host system. Currently, the USB drivers and STAPL player are installed concurrently with the Achronix CAD Environment (ACE) software; however, they can be installed separately if desired.

Note: *No license file is needed when running the STAPL player from the command line.*

Standalone STAPL Player and USB Driver Install

For users wishing to install only the player, for example when using a separate host PC in a lab environment, the player and USB drivers can be installed separately. Users wishing to install the entire ACE software should refer to *ACE Installation and Licensing Guide (UG002).*

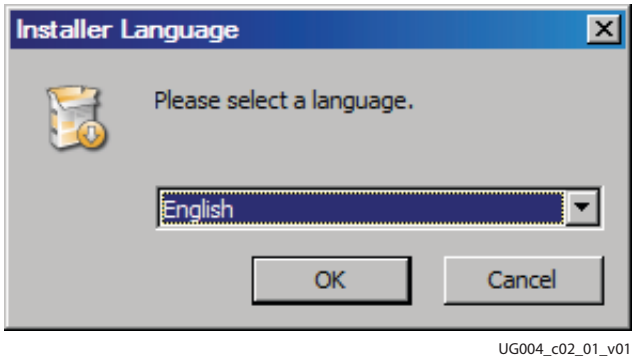
Important! *Disconnect any attached Bitporter pods from the host PC before installing the player and drivers.*

Windows

After locating the standalone STAPL player installer (`standalone_acx_stapl_player_installer.exe`) on the SDK60 CD, install the software to the desired installation directory.

Important! *It is not recommended to install both ACE and the standalone STAPL player on the same machine. Be sure to uninstall any previous installation of ACE before running the standalone STAPL player installer.*

1. Locate the standalone STAPL player installer `standalone_acx_stapl_player_installer.exe`) on the SDK60 CD. Double-click the install file to begin the process.
2. Select the desired language from the pull-down menu of the Installer Language dialog box (**Figure 2-1**).and click **OK** to open the Achronix STAPL Player Setup wizard.



UG004_c02_01_v01

Figure 2-1: *Installer Language Dialog Box*

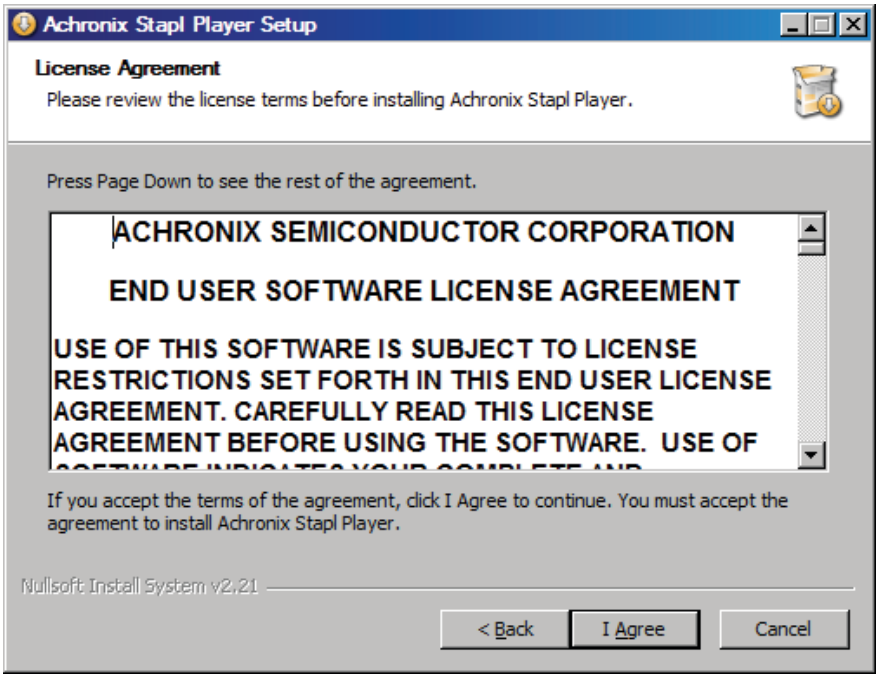
3. From the wizard, click **Next >** to open the License Acceptance dialog box. Click **I Agree** to continue.



UG004_c02_02_v01

Figure 2-2: *Achronix CAD Environment Setup Wizard*

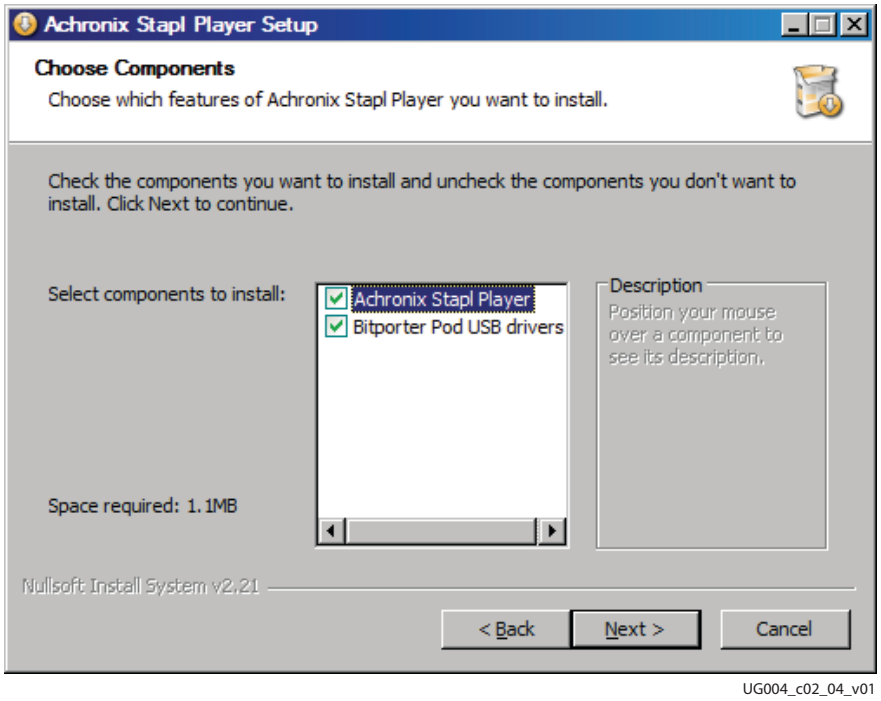
4. From the License Agreement dialog box (**Figure 2-3**), click **I Agree** after reviewing the license.



UG004_c02_03_v01

Figure 2-3: *License Agreement Dialog Box*

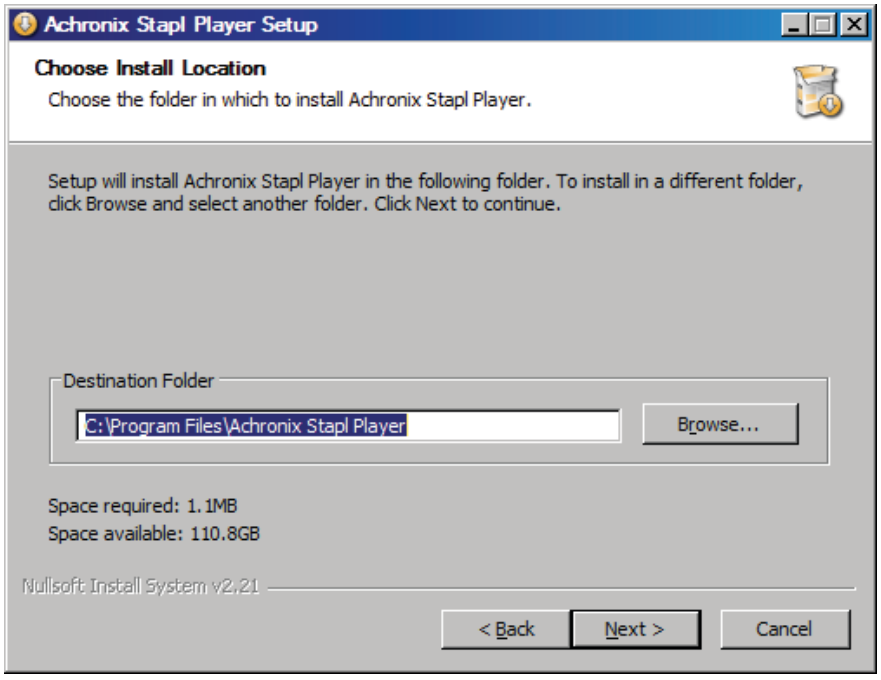
5. From the Choose Components dialog box (**Figure 2-4**), select the desired components to install. Click **Next >** to continue.



UG004_c02_04_v01

Figure 2-4: *Choose Components Dialog Box*

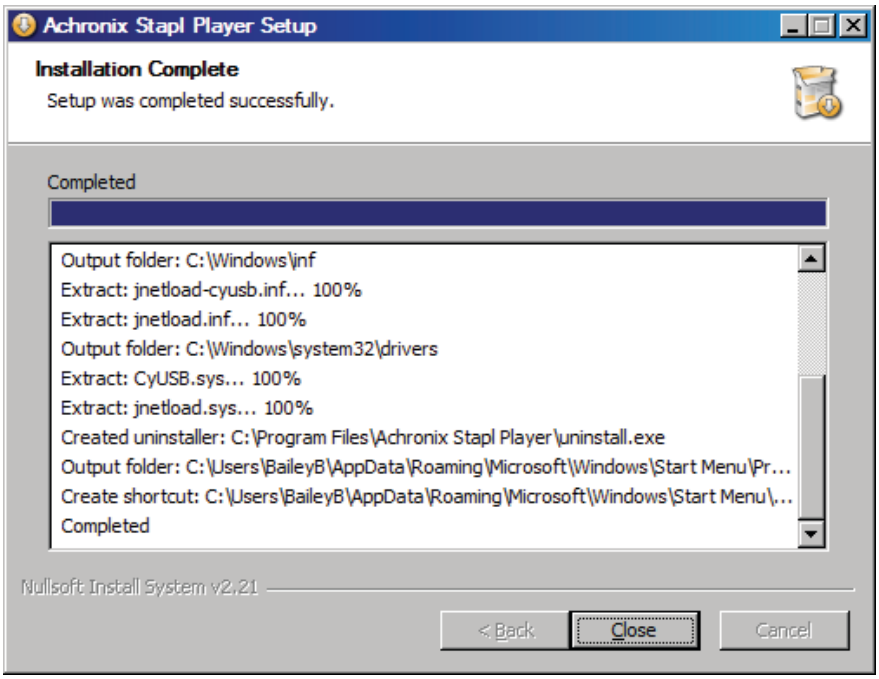
6. From the Choose Install Location dialog box (**Figure 2-5**), set the desired installed directory by either typing the path under Destination Folder or using the **Browse** button to navigate the desired location. Click **Next >** to continue.



UG004_c02_05_v01

Figure 2-5: *Choose Install Location Dialog Box*

7. From the Choose Start Menu Folder dialog box, select the desired folder and click **Install** to start file extraction and begins installation. Once the install completes, click **Close** from the final dialog box (**Figure 2-6**).



UG004_c02_06_v01

Figure 2-6: *Completing Install*

Linux

After locating the standalone STAPL player installer (`standalone_acx_stapl_player.tgz`) on the SDK60 CD, extract and install the software to the desired installation directory.

Note: *The archive contains no directory structure; therefore, the installation directory chosen must be the final desired location.*

1. Extract the software.

```
% mkdir <install_directory>
% cd <install_directory>
% tar -xzvf standalone_acx_stapl_player.tar
```

2. (Only necessary when USB Bitporter support is desired.) Run the Perl install script `./install_acx_bitporter_usb.pl`

```
# ./install_acx_bitporter_usb.pl
```

After the installation completes, the script ends with the following message:

```
Bitporter USB driver, version 2.4.2, installed.
```

Note: *Linux USB Bitporter connections are currently only supported on 32-bit CentOS 5 and RHEL 5. Other Linux release may work if they support udev, but are not supported.*

Chapter 3 – Connecting the Bitporter Pod

Introduction

Choosing a Connection Type — USB versus Ethernet

Ethernet

- Ethernet-connected Bitporter pods are visible from multiple PCs at once:
  - However, users can overwrite each other's programs. The Achronix acx_stapl_player/Bitporter programming system does prevent multiple users from programming the device simultaneously but does not ensure that the connected hardware is not already mid-test. Such test queueing is left up to the user.
  - Pod autodetect visibility is limited to the pod's local subnet. Beyond the local subnet, the pod's IP address must be known before a user can connect to it.
- Bitporter performance may be slower via Ethernet than via High-Speed USB, depending upon network configuration. On a congested network, programming via Ethernet can take up to 5x longer than via High-Speed USB.
- Before attempting Ethernet connection to a Bitporter pod, please consult the network administrator to ensure the Bitporter is allowed to connect to the necessary network.
- A USB connection is required to alter the Bitporter pod's Ethernet configuration, for example, selecting static IP over DHCP, changing the static IP, etc. (see command-line options `-id`, `-iq`, and `-is` in **Table 4-1**, page 19)

USB

- Bitporter performance may be faster than via Ethernet, depending upon network configuration.
- Only visible from the PC hosting the connection.

Connecting the Bitporter Pod

Important! *Never apply power to the target without the pod turned on. The Bitporter pod must always be powered if the connected target device is powered. Always supply power to the Bitporter pod first before supplying power to the target hardware. Always power down the target hardware before powering down the Bitporter pod. When power-cycling the pod, leave it turned off for at least 5 seconds.*

Also, the Bitporter pod is sensitive to Electrostatic Discharge (ESD). When operating the pod, ESD precautions must be observed to ensure proper function.

Connecting the Bitporter Pod via USB

Before connecting the Bitporter pod, please observe the following:

- Do not plug in the Bitporter USB cable until *after* the software installation (see **Chapter 2 – "Software and Driver Install"**).
- When using the USB interface, *do not* plug in the power cable. The power cable is only used for the Ethernet interface. When using the USB interface, all power to the device is provided through the USB cable.

Note: *When the power cable is plugged into the Bitporter, the Bitporter ignores the USB interface and only utilizes the Ethernet interface.*

- Be sure to have the USB appropriate drivers installed. See **Chapter 2 – "Software and Driver Install"** for details.

Windows

To connect the Bitporter pod to the target hardware:

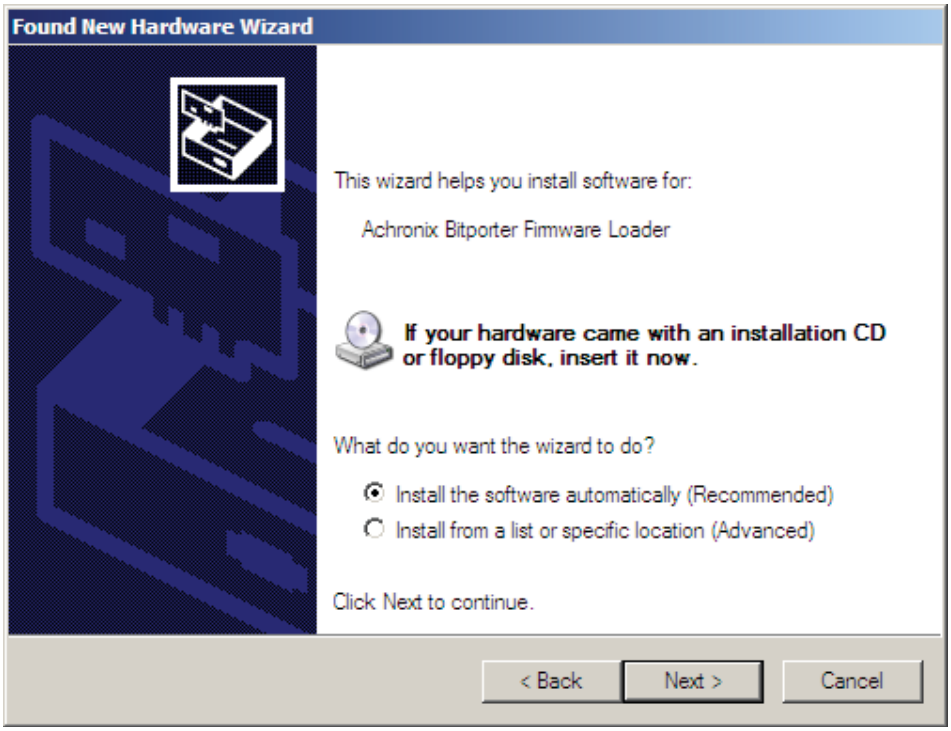
1. Turn off the power to the target hardware.
2. Connect one end of the JTAG flat ribbon cable to the target EJTAG connector. The red strip is pin 1.

   Note: *The key of the JTAG cable should point towards J52 (towards the bottom edge of the board).*
3. Connect the other end of the JTAG flat ribbon cable to the Bitporter pod. The plug is keyed.
4. Connect the USB cable to the host PC.
5. Connect the USB cable to the Bitporter.
6. When prompted by the Found New Hardware Wizard, click **Next >** (**Figure 3-1**).



Figure 3-1: *Found New Hardware Wizard*

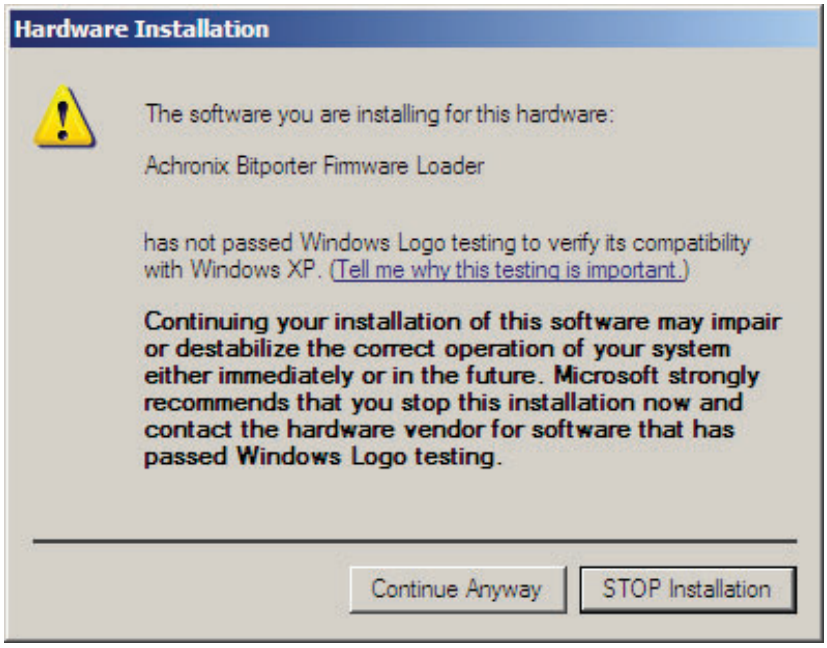
7. At the next window of the wizard (**Figure 3-2**), select **Install the software automatically (Recommended)**, followed by **Next >.**



ug004_c03_02_v01

Figure 3-2: *Found New Hardware Wizard*

8. After the wizard has found the Bitporter firmware loader, at the Windows Logo testing warning window, click **Continue Anyway**.



ug004_c03_03_v01

Figure 3-3: *Windows Logo Testing Warning Window*

9. At the final wizard window (**Figure 3-4**), click Finish to complete the driver install.

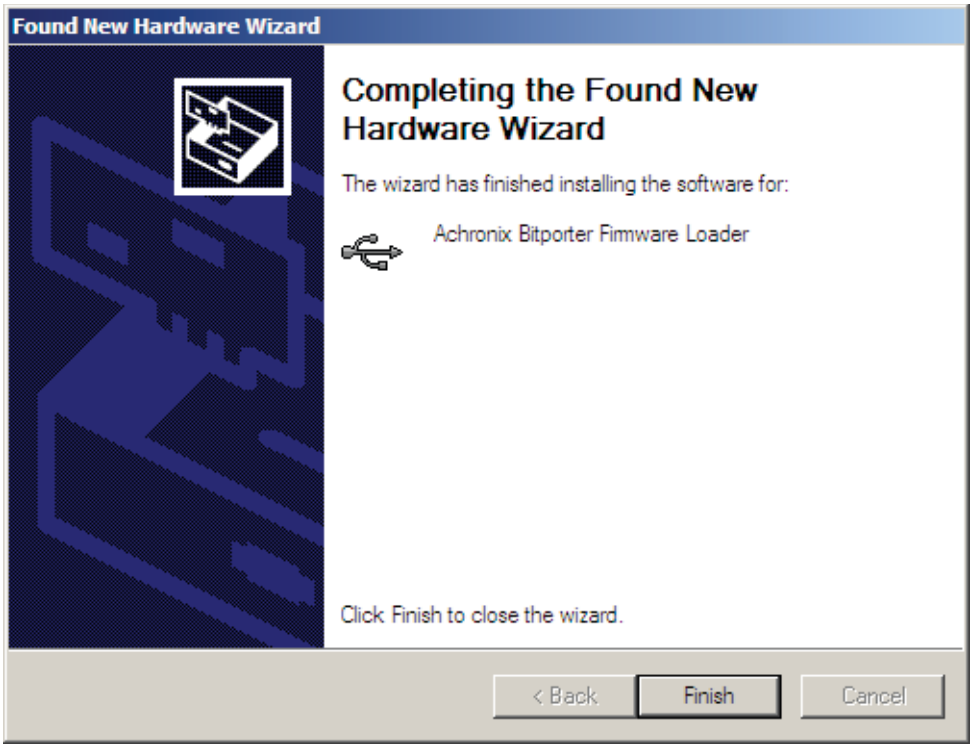


Figure 3-4: *Completing the Found New Hardware Wizard*

The wizard appears twice during the install: once for the Achronix Bitporter Firmware Loader and then again for the Achronix Bitporter Pod drivers. The first driver downloads firmware to the Bitporter pod; the second driver connects to the pod for JTAG operations.

If *not* prompted to install these drivers the first time the Bitporter is connected to a USB port, unplug the USB cable, reboot the host PC. Once the system has rebooted, plug in the USB cable again — at this point driver installation should begin.

Note: *During the driver installation, the Bitporter pod's power LED turns on and the COMM LED may flash. Once driver installation completes successfully, the power LED remains lit, and the COMM LED turns off.*

10. Turn on the power to the target hardware.
11. Continue to **"Verifying the Setup,"** on page 16.

Linux

1. Turn off the power to the target hardware.
2. Connect one end of the JTAG flat ribbon cable to the target EJTAG connector. The red strip is pin 1.

   Note: *The key of the JTAG cable should point towards J52 (towards the bottom edge of the board).*
3. Connect the other end of the JTAG flat ribbon cable to the Bitporter pod. The plug is keyed.
4. Connect the USB cable to the host PC.

5. Connect the USB cable to the Bitporter.

   Note: *During the pod initialization, the Bitporter pod's power LED turns on and the COMM LED may flash. Once pod initialization completes successfully, the power LED remains lit, and the COMM LED turns off.*

6. Turn on the power to the target hardware.
7. Continue to **"Verifying the Setup,"** on page 16.

Connecting the Bitporter Pod via Ethernet (Windows and Linux)

The Bitporter pod supports Ethernet connectivity to one or more PCs, supporting both static and dynamic (via DHCP) IP addressing. Before attempting Ethernet connection to a Bitporter pod, please consult the network administrator to ensure that:

- A DHCP server covers the network segment onto which the pod is attached. If no such DHCP server is in place, then consult with the system administrator to determine the static IP addressing settings to be explicitly assigned to the pod.
- The Bitporter pod is allowed to connect to the necessary network.

By default, all Bitporter pods are set up to use DHCP negotiation the first time they are plugged into an Ethernet network. Since the pod stores its Ethernet configuration in flash memory, the Ethernet configuration typically only needs to be performed once.

Note: *Only pods on the same subnet (without an intervening router) can be automatically detected by STAPL player (see the* `-q` *option,* **Table 4-1**, *page 19). Use the* `-p` *option to connect to pods on non-local subnets using the podname containing the IP address of the pod (see* **Table 4-1***). For ease of use, it is recommend that both the PC used to program the Achronix device and the Bitporter pod are placed on the same subnet.*

To connect to a Bitporter pod via Ethernet:

1. Turn off the power to the target system.
2. If the pod is connected to the PC via USB, disconnect the USB cable from the Bitporter.
3. Connect the JTAG flat ribbon cable to the target JTAG pins. The red strip is pin 1.
4. Connect the JTAG flat ribbon cable to the Bitporter pod. The plug is keyed.
5. Connect the power cable to the power outlet and the pod. The Bitporter pod's POWER LED should now be on. The pod's COMM LED should start blinking.
6. Connect the Ethernet cable to the Ethernet jack on the wall and the pod. Once the Bitporter pod's COMM LED stops blinking, the pod has successfully acquired an IP address.

   Note: *If the COMM LED does not stop blinking (it should take less than ten seconds), talk to the network administrator to verify that the device's MAC address has permission to connect to the local network, and that the device is connected to the proper Ethernet network jack.*

7. Turn on the power to the target hardware.
8. Continue to **"Verifying the Setup,"** on page 16.

The pod is now able to be referred to by name as `net<serial_number>` or `net<ip_address>`. The latter name must be used if pod communication is being attempted across subnets (see **Table 4-1**).

Disconnecting the Bitporter Pod

To end a programming session and disconnect the Bitporter pod from the target hardware:

1. Wait until `acx_stapl_player.exe` finishes running.
2. Turn off the power to the target hardware.
3. Disconnect the USB cable for USB-connected pods, or the power cable for Ethernet-connected pods.

Bitporter Connection Errors

Below is a listing of possible error messages that can occur during USB setup or pod operation:

"Attempting to auto-detect Bitporter pods (USB pods and Ethernet pods on this sub net)... found 0 pods."

During the operation of acx_stapl_player, if no USB pods are connected, or if all USB-connected pods are busy, the following error message sequence is displayed:

```
Achronix STAPL Player (acx_stapl_player) -- Version 0.X (Main) -- Build
22573*
-- Date 2009-03-04 07:08
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All rights
reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

****************************************
*  Attempting to find all reachable pods:  *
****************************************
Attempting to auto-detect Bitporter pods (USB pods and Ethernet pods on
this sub net)...
        ...found 0 pods.
INFO: No pods found during auto-detection phase.  (Auto-detection can only
find Ethernet-connected pods on the local subnet, and USB-connected pods
which are not currently in use.)
INFO: Please verify that your Bitporter cables are plugged in properly.  If
you are using Ethernet, please ensure that the power supply is plugged in to
the Bitporter pod.  If you are using a USB connection, do not connect the
power supply to the Bitporter pod.  For more information, please see the
Chapter "Connecting the Bitporter Pod" in the ACE Programming Guide (UG004).
```

Ensure that the Bitporter Pod is properly connected and is not busy with another process.

"PROGRAM ERROR: Bitporter driver library not found - unable to program device."

If during the operation of acx_stapl_player, the following error message sequence is displayed:

```
PROGRAM ERROR: Bitporter driver library not found - unable to program
device.
PROGRAM ERROR: Exiting with error code: -10
```

First verify that the host machine is running a supported operating system. For example, this error can occur when running 64-bit Linux distributions.

If the host system is running a supported operating systems, ensure that `libjnetserver.so` file in the same directory as the `acx_stapl_player` executable.

"Port has not been opened"

An attempt was made to access the USB device before it was opened. This error can occur if the device is power-cycled or unplugged without restarting the software.

"Unable to retrieve pathname for USB device"

This is an internal error. Please contact Achronix technical support.

"USB communication error"

A communication error not covered by another message was encountered. Check connections and POWER LED status.

"USB communication timeout"

No response was received from the Achronix device within a timeout period. Check connections and POWER LED status. This error can also occur if the unit is unplugged or powered down.

"Unable to load hid.dll. One or more exported functions missing"

The Windows library `hid.dll` could not be loaded because its interface was incompatible. This is an internal error. Please contact Achronix technical support.

"Can't open setupapi.dll: <reason>"

Windows could not open `setupapi.dll`. The error string returned from Windows follows.

"Can't open hid.dll: <reason>"

Windows could not open its USB driver file `hid.dll`. The error string returned from Windows follows.

"USB read error. <reason>"

The Windows driver returned an error from a read operation. The Windows error text follows. This error can occur if the unit is powered down or unplugged during operation.

"USB write error. <reason>"

The Windows driver returned an error from a write operation. The Windows error text follows. This error can occur if the unit is powered down or unplugged during operation.

"No available Achronix Bitporter products with matching serial number found on USB port."

In enumerating the devices attached to the USB port, at least one Achronix Bitporter pod was found but the pod with the specified serial number was not found or was already in use. Check the serial number (be sure to enter it exactly as shown on the serial number sticker),

then check the connection and be sure the POWER LED is illuminated before starting the software. See **"Bitporter Pod Naming Conventions"** in **Chapter 4 – Using the Achronix STAPL Player** for more details.

Ethernet Connection Errors:

There are a number of port-related error messages that can occur during the TCP/IP and Ethernet setup process and operation:

"No available Achronix Bitporter products with matching serial number found on local subnet."

The program attempted to connect to an Ethernet pod specifying a serial number (e.g. "net40XXX"), and no Achronix Bitporter pod with a matching serial number was found on the local subnet. Check the connection and ensure that the COMM LED has finished blinking (after power-up) before starting the software. If the pod's TCP/IP settings were manually configured, verify that the provided settings are correct. Verify with the System Administrator that the pod is on the same local subnet as the PC attempting to connect. See **"Bitporter Pod Naming Conventions"** in **Chapter 4 – Using the Achronix STAPL Player** for more details.

"No available Achronix Bitporter product found at supplied network address."

The program attempted to connect to an Ethernet pod specifying an IP address but no Bitporter pod was found at that address. Check the connection and ensure that the COMM LED has finished blinking (after power-up) before starting the software. If the pod's TCP/IP settings were manually configured, verify that the provided settings are correct. If the pod is configured to use DHCP, verify with the System Administrator that the network address is correct.

"Network device is not an Achronix Bitporter pod."

The program attempted to connect to a pod at the supplied address, but the device at the that location does not appear to be a Achronix Bitporter pod. Verify the IP address of the desired pod and that the correct address was specified.

Verifying the Setup

Bitporter Connectivity Self Test

To verify that the USB drivers, STAPL player and Bitporter pod are correctly installed:

1. Open a command prompt in the desired installation directory.
2. At the command prompt, run:

   `acx_stapl_player -q`

   The program returns a listing of all correctly connected and currently available pods (those not currently in use):

```
Achronix STAPL Player (acx_stapl_player) -- Version 2.1
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All rights
reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation
```

```
****************************************
* Attempting to find all reachable pods: *
****************************************
Attempting to auto-detect Bitporter pods (USB pods and Ethernet pods on
this subnet)...
        ...found 3 pods.
|==========|=================|
|   Pod    | (Ethernet only) |
|   Name   | Owner's IP addr |
|==========|=================|
| usb12345 |  <not tracked>  |
| net12346 | 192.168.1.123   |
| net12347 |   <available>   |
|==========|=================|
```

Refer to **"Querying the Availability of Connected Pods (-q option)"** in **Chapter 4 – Using the Achronix STAPL Player** for complete details.

Bitporter-to-Target-Device Connectivity Test

To verify the connection to the target device via the Bitporter pod:

1. Open a Windows command prompt and navigate to the STAPL player installation directory.
2. At the command prompt, run:

   `acx_stapl_player -aREAD_IDCODE read_idcode.jam`

   Note: *The file* `read_idcode.jam` *is on the SDK60 CD in the* `/Software` *directory.*

   After successful opening the Bitporter Pod, the program returns the device ID code of the target device:

   ```
   Entering ISP programming mode...
   Reading Device ID code...
   IDCODEVALID=0
   IDCODE=0011000 00000000 00001011 001000001
   Exiting ISP programming mode...
   Exit code = 0... Success
   ```

Handling Multiple Pods Connected to the Same PC

By default the Achronix STAPL Player assumes that it is operating in a single Bitporter pod environment. If this is true, no special actions by the user are necessary — when acx_stapl_player finds only one pod during the auto-detection phase, it uses that pod.

The Achronix STAPL Player can support multiple users sharing a collection or pool of Bitporter pods connected to a single PC via USB, and/or Bitporter pods connected to a collection of PCs via Ethernet. When multiple pods are found, but the desired pod is not specified with the `-p` command-line option, acx_stapl_player exits with an error, informing the user that they must specify by name, which pod(s) is/are to be used.

Note: *The* `-q` *command-line option can be used to list the pods detected by acx_stapl_player (see* **Table 4-1**, *page 19).-*

Chapter 4 – Using the Achronix STAPL Player

Introduction

The Achronix STAPL Player (`axc_stapl_player.exe`) takes the bitstream output of the Achronix CAD Environment (ACE) in STAPL format, and then runs (or plays) the STAPL program. The program causes the Bitporter pod to pass these commands to the JTAG port of an Achronix FPGA. The STAPL player executable is run from a Windows or Linux command prompt.

Command Syntax

The STAPL player command syntax is:

```
acx_stapl_player [options] [filename]
```

The available command options are listed in **Table 4-1**.

Note: *The option names are not case sensitive.*

Table 4-1: *acx_stapl_player.exe Command Options*

Option	Description
`-h`	Displays the help message.
`-q`	Displays the name and state of all available USB-connected pods, and all Ethernet-connected pods regardless of availability (the pod's current owner's IP address is shown if the pod is not available). (1)(2) See also **"Querying the Availability of Connected Pods (-q option)," on page 28.**
`-a<action>`	Specifies an action name (see **Table 4-2**).
`-d<proc=1>`	Enables an optional procedure.
`-d<proc=0>`	Disables a recommended procedure.
`-d<var=val>`	Initializes STAPL variable (var) to specified value (val).
`-id`	Sets IPv4 configuration of a currently USB-connected pod to DHCP (dynamic IP) so that it may be connected via Ethernet. (3)
`-iq`	Displays the current IPv4 configuration of a connected pod. (2)
`-is<config>`	Static IP Configuration mode sets the static IPv4 configuration of a currently USB-connected pod so that it may be connected via Ethernet. The argument <*config*> is a comma-delimited list of configuration options — follow the s with a comma, then the desired IP address, a comma, then the subnet mask, a comma, then the default gateway. (3) Static IP example: -is,192.168.1.123,255.255.255.0,192.168.1.1 See also **"Configuring the Bitporter Pod's IP address (-i Option)," on page 30.**

Table 4-1: *acx_stapl_player.exe Command Options (Continued)*

Option	Description
`-l<filename>`	Sends program output to a log file as well as the console.

`-p<podname>`	Specifies a pod by name, which the tool should use to load the STAPL file. Switch can be followed with comma-separated list of pod names (no spaces allowed). If none of the specified pods are available, the program exits. For example, to connect to a USB-connected pod with the serial number of 40815:
	`-pusb40815`
	For an Ethernet-connected pod with the serial number of 40815 on the same subnet:
	`-pnet40815`
	For an Ethernet-connected pod not on the local subnet and at address 192.168.1.1:
	`-pnet192.168.1.1`
	See also **"Connecting to Specific Pods by Name (-p option),"** on page 29.

Notes: 1. Query pod mode only finds Ethernet pods on the current subnet. Pods on other subnets are not displayed. 2. No STAPL actions are run; program exits after query. 3. No STAPL actions are run; program exits after the pod (re)configuration is complete.	

Bitporter Pod Naming Conventions

Each Bitporter pod has a unique name which include a prefix to indicate the kind of connection used to communicate with the pod (for example, "usb" for USB-connected pods and "net" for Ethernet-connected pods) plus a suffix indicating either the pod's serial number or IP address.

For example, a pod with the serial number 12345 (the serial number is on a sticker located on the Bitporter), the pod is can be named:

- **usb12345 –** if connected via USB.
- **net12345** – if connected with Ethernet, connected on the subnet.
- **net192.168.1.123** – if connected via ethernet, whether on the local subnet or not.

Picking a STAPL Action (-a option)

When executing or playing a STAPL file, the user is required to choose an action to take, since most STAPL files contain numerous actions. When using the STAPL player, the user specifies the desired action with the `-a` option. For example, to program an Achronix device with design:

```
acx_stapl_player -aprogram <design_name>.jam
```

Note: *Only a single action can be specified and executed at a time.*

Table 4-2 lists the standard STAPL actions which can be found in STAPL files generated by ACE.

Table 4-2: *STAPL Actions Generated by ACE*

Action Name	Description
read_idcode	Returns the IEEE 1149.1 IDCODE of the target device.
program	Programs the device.
verify	Verifies the programming of the device.
erase	Performs a bulk erase of the device.
read	Reads the programming data of the device.
read_state	read_state polls the current state of the chip.
Notes: 1. All actions in this table are supported in ACE v2.1.	

Procedures

STAPL actions are made up of one or more STAPL procedure calls (**Table 4-3** and **Table 4-4**). Some of these procedures can be marked as OPTIONAL or RECOMMENDED, meaning the user can choose to include or exclude the specified procedure — RECOMMENDED procedures are called by default, but may be excluded by the user; OPTIONAL procedures are not called by default but may be included by the user. Refer to EIA/JEDEC Standard 71 (JESD71), *Standard Test and Programming Language (STAPL)* for complete details.

Note: *STAPL is case insensitive; the procedure names listed in* **Table 4-3** *and* **Table 4-4** *are in all capitals for readability.*

Table 4-3: *Standard STAPL Procedures used by ACE*

Procedure Name	Description
DO_READ_IDCODE	Returns the JTAG IDCODE.
DO_ERASE	Performs a bulk erase of the device.
DO_PROGRAM	Programs the device.
DO_VERIFY	Verifies the programming data of the device.(1)
DO_READ	Reads the programming data from the device.(1)
Notes: 1. Only covers core data excluding block RAMs, I/O, SerDes, etc.	

Table 4-4: *Additional (Non-Standard) Procedures used by ACE*

Procedure Name	Description
DO_ENTER_USER_MODE	Changes the mode of the device to User Mode.
DO_EXIT_USER_MODE	Changes the mode of the device out of User Mode.
DO_BRAM_VERIFY	Verifies the programming data of the block RAMs in the device.
DO_BRAM_READ	Reads the programming data from the block RAMs in the device.
DO_READ_STATE	Checks for synchronization state errors on the device.
DO_ENABLE_DETAILS_ON_FAILURE	If there is a failure detected during DO_READ_STATE, enabling this optional procedure produces detailed error information useful for debug.

Disabling a Recommended Procedure

The -d<proc=0> command-line option can be used to disable a recommended procedure. Multiple recommended procedures can be disabled within a single action, but each require its own command-line option to disable. For example, to disable the recommended procedure DO_VERIFY during programming for the file demo.jam, enter the following:

```
acx_stapl_player -aprogram -dDO_VERIFY=0 demo.jam
```

To disable both DO_VERIFY and DO_ENTER_USER_MODE:

```
acx_stapl_player -aprogram -dDO_VERIFY=0 -dDO_ENTER_USER_MODE=0
demo.jam
```

Enabling an Optional Procedure

The -d<proc=1> command-line option can be used to enable an optional procedure. Multiple optional procedures can be enabled, each requiring its own command-line option. For example, to enable the optional procedure DO_READ for the program action in the file demo.jam, enter the following:

```
acx_stapl_player -aprogram -dDO_READ=1 demo.jam
```

The Action read_idcode

The action read_idcode returns (to the command-line) the JTAG IDCODE reported by the connected device. For example, the expected output for this action when run against the SPD60-FBGA 1892 is:

```
IDCODE=00100000 00000000 00010110 01000001
```

Table 4-5: *Procedures Called During read_idcode*

Procedure name	Optional	Recommended
DO_ENTER_JTAG		
DO_READ_IDCODE		
DO_EXIT_JTAG		

The Action program

The program action erases the contents of the target device and then downloads the user's design.

Table 4-6: *Procedures Called During program*

Procedure Name	Optional	Recommended
DO_ENTER_JTAG		
DO_RESET_CHIP		
DO_ERASE		
DO_PROGRAM		
DO_VERIFY	✓	
DO_BRAM_VERIFY	✓	
DO_BRAM_READ	✓	
DO_READ	✓	

Table 4-6: *Procedures Called During program (Continued)*

Procedure Name	Optional	Recommended
DO_ENTER_USER_MODE		✓
DO_READ_STATE	✓	
DO_EXIT_JTAG		

The Action verify

The verify action verifies that the design contained in the STAPL file is the same design found in the target device, with results reported to the command-line. The verify action may be run/played when the device is in user mode, or when out of user mode. By default, only core configuration memory is verified, excluding block RAM.

Note: *Verification of I/O ring configuration memory is not supported at this time.*

Table 4-7: *Procedures Called During verify*

Procedure Name	Optional	Recommended
DO_ENTER_JTAG		
DO_EXIT_USER_MODE	✓	
DO_BRAM_VERIFY	✓	
DO_VERIFY		
DO_ENTER_USER_MODE	✓	
DO_EXIT_JTAG		

The Action read

The read action reads the entire contents of the core configuration memory (optionally the block RAM contents) and dumps the contents to the console.

Note: *Reading the I/O ring configuration memory is not supported at this time.*

Table 4-8: *Procedures Called During Read*

Procedure Name	Optional	Recommended
DO_ENTER_JTAG		
DO_EXIT_USER_MODE	✓	
DO_BRAM_READ	✓	
DO_READ		
DO_ENTER_USER_MODE	✓	
DO_EXIT_JTAG		

Core Configuration Memory Output

Each core configuration memory address frame contains 1024 bits of configuration data (32 words of 32 bits), manipulated by the STAPL player as eight 128-bit blocks for performance reasons.

Each 128-bit block is composed of four 32-bit words (**Figure 4-1**). The LSB of Word[0] is the LSB of the 128-bit chunk at offset 0 of a given address. The MSB of Word[3] is the MSB of the 128-bit chunk at offset 0 of a given address. The LSB of Word[4] is the LSB of the 128-bit chunk at offset 1 of a given address. The MSB of Word[31] is the MSB of the 128-bit chunk at offset 7 of a given address.

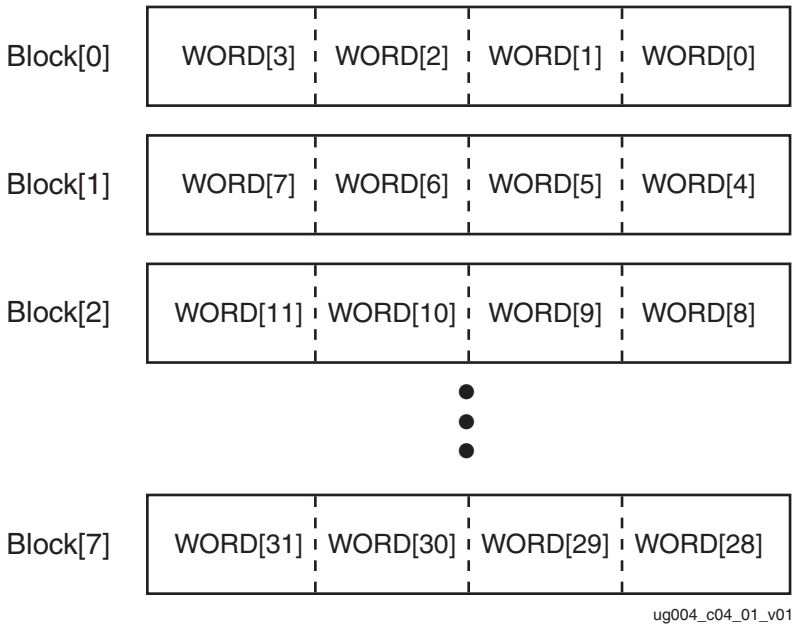


Figure 4-1: *Core Configuration Memory Output*

Successful Output Example

The example below shows truncated results for a single core configuration memory address read (actual results would contain on the order of 36,735 addresses).

Note: *The results are redirected to the file* `quickstart.log`*, for ease of later reading.*

Command entered:

```
acx_stapl_player.exe quickstart.jam -aREAD > quickstart.log
```

The command returns:

```
Achronix STAPL Player (acx_stapl_player) -- Version 0.X (Main) -- Build
15556* -- Date 2008-07-07 08:49
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All rights
reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

Entering ISP programming mode...
Reading the programming data of the device...
Reading core configuration memory contents:
core address:
                                       00 0000
    0000 0003 0000 0002   0000 0001 0000 0000
    0000 0007 0000 0006   0000 0005 0000 0004
    0000 000B 0000 000A   0000 0009 0000 0008
    0000 000F 0000 000E   0000 000D 0000 000C
    0000 0013 0000 0012   0000 0011 0000 0010
    0000 0017 0000 0016   0000 0015 0000 0014
    0000 001B 0000 001A   0000 0019 0000 0018
    0000 001F 0000 001E   0000 001D 0000 001C
Exiting ISP programming mode...
Exit code = 0... Success
```

Failed Output Example

If there are any regions of memory encountered in the read which fail their read request, the program reports the read address, which 32-bit word from that address had the failure, and then exits immediately.

Command entered:

```
acx_stapl_player.exe quickstart.jam -aREAD > quickstart.log
```

The command returns:

```
Achronix STAPL Player (acx_stapl_player) -- Version 0.X (Main) -- Build
15556* -- Date 2008-07-07 08:49
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All rights
reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

Entering ISP programming mode...
Reading the programming data of the device...
Reading core configuration memory contents:
ERROR: Hardware indicates invalid read data at word: 0 of core memory
address:
                                      00 0000
  Invalid word follows:
                                   0 0000 0000
PROGRAM ERROR: Exit code = 12... Read failure
```

The Action read_state

The read_state action allows the user to poll the live state of the device. If any of the bits are high, the Action reports READ_STATE_FAIL. If all the bits are low, the Action reports READ_STATE_PASS.

Pass Example

Command entered:

```
acx_stapl_player primitivedff.jam -aREAD_STATE
```

The command returns:

```
Achronix STAPL Player (acx_stapl_player) -- Version 2.1
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All rights
reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

Entering JTAG programming mode...
Number of devices detected on scan chain: 1
Reading chip state...
    READ_STATE_PASS: no errors found.
Exiting JTAG programming mode...
Exit code = 0... Success
```

Default Fail Example

Command entered:

```
acx_stapl_player primitivedff.jam -aREAD_STATE
```

The command returns:

```
Achronix STAPL Player (acx_stapl_player) -- Version 2.1
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All rights
reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

Entering JTAG programming mode...
Number of devices detected on scan chain: 1
Reading chip state...
    READ_STATE_FAIL: errors found!
Exiting JTAG programming mode...
Exit code = 0... Success
```

Detailed Fail Example

Command entered:

```
acx_stapl_player primitivedff.jam -aREAD_STATE -
dDO_ENABLE_DETAILS_ON_FAILURE=1
```

The command returns:

```
Achronix STAPL Player (acx_stapl_player) -- Version 2.1
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All rights
reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

Entering JTAG programming mode...
Number of devices detected on scan chain: 1
Reading chip state...
    READ_STATE_FAIL: errors found!
        error at bit: 0
        error at bit: 1
        error at bit: 2
        error at bit: 3
        error at bit: 4
        error at bit: 5
        error at bit: 6
        error at bit: 7
        error at bit: 8
        error at bit: 9
        error at bit: 10
        error at bit: 11
        error at bit: 12
        error at bit: 14
        error at bit: 15
        error at bit: 16
        error at bit: 17
        error at bit: 18
        error at bit: 19
```

```
        error at bit: 20
        error at bit: 21
        error at bit: 22
        error at bit: 23
        error at bit: 24
        error at bit: 25
        error at bit: 26
        error at bit: 27
        error at bit: 28
        error at bit: 29
        error at bit: 30
        error at bit: 31
        error at bit: 32
        error at bit: 33
    asyncerr[1991..1919] =
                      00   0000 0000 0000 0000
    asyncerr[1919..1792] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[1791..1664] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[1663..1536] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[1535..1408] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[1407..1280] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[1279..1152] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[1151..1024] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[1023..896] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[895..768] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[767..640] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[639..512] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[511..384] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[383..256] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[255..128] =
     0000 0000 0000 0000   0000 0000 0000 0000
    asyncerr[127..0] =
     0000 0000 0000 0000   0000 0003 FFFF DFFF
Exiting JTAG programming mode...
Exit code = 0... Success
```

The Action erase

The erase action resets the configuration memory in the device to their default values, including block RAM and I/O configuration bits. There are no expected failure modes for this Action.

Querying the Availability of Connected Pods (-q option)

The Bitporter pod query mode is used when multiple Bitporter pods are visible to the host running acx_stapl_player. This mode returns the current state of all connected pods (whether in use or not).

Important! *It is recommended that this query mode not be the only means used to synchronize multiple users' attempts to use multiple pods — users still need to ensure that a given target board is not being used through another non-Bitporter interface before using the Bitporter attached to that target board.*

To see a list of the pods currently auto-detected by acx_stapl_player, run the following at the command-line:

```
acx_stapl_player -q
```

This command returns to the command-line a table of the currently detected pods, using the following criteria:

- **USB-connected Bitporter pods:** Pods connected to the host are listed by name in the first column only if they are currently not in use by another instance of acx_stapl_player. Pods connected to the host via USB which are already in use are not be listed. Pods connected to other hosts via USB are not listed, regardless of usage state.
- **Ethernet-connected Bitporter pods:** All Bitporter pods connected on the same subnet as the host are listed by name in the first column. The second column contains the owner of each pod found. A listing of `<available>` means the pod currently has no owner (and is available for immediate use). A listing displaying an IP address means the pod is currently in use by a host located at that IP address.

For example, if two pods with serial numbers 12345 and 12346 are connected to the same host via USB, and two additional pods with serial numbers 12347 and 12348 are connected via Ethernet on the same subnet, then run the acx_stapl_player in query mode:

```
acx_stapl_player -q
```

The command returns:

```
Achronix STAPL Player (acx_stapl_player) -- Version 1.5
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All rights
reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

****************************************
* Attempting to find all reachable pods: *
****************************************
Attempting to auto-discover Bitporter pods (USB pods and Ethernet pods on
this subnet)...
        ...found 3 pods.
|==========|=================|
|   Pod    | (Ethernet only) |
|   Name   | Owner's IP addr |
|==========|=================|
| usb12345 |  <not tracked>  |
| net12346 | 192.168.1.123   |
| net12347 |   <available>   |
|==========|=================|
```

In this example, pod 12345 is connected via USB, and is currently not in use. Pod 12346 is visible via Ethernet, but is already in use by a host at IP address 192.168.1.123. Pod 12347 is also visible via Ethernet, currently has no owner, and is thus available for use. Pod 12348 is not listed, which means it must also be connected via USB, but it is in use. (USB probes in use are not listed.)

In this case, the user can run acx_stapl_player to connect to either usb12345, net12347, via the `-p` command-line option.

Connecting to Specific Pods by Name (-p option)

When multiple pods are visible to the acx_stapl_player, the user must specify which pod the acx_stapl_player should use to communicate with a device. The desired pod is specified with the `-p` command-line option, where the user specifies a podname or a list of potential podnames to be used. To determine which pods are currently available for use, the `-q` option is used.

Once a pod name or list of pod names is known, they can be specified on the command-line as a comma-separated list immediately following the `-p` on the command-line. When multiple pod names are listed, acx_stapl_player attempts to connect to them in the specified order and uses the first available specified pod. The player reports which pod is being used.

For example, to read the JTAG IDCODE of the Achronix device connected to the first available pod via pods 12348, 12345, and 12347, in that order, use the following command:

```
acx_stapl_player -pusb12348,usb12345,net12347 -aREAD_IDCODE quickstart.jam
```

The command returns:

```
Achronix STAPL Player (acx_stapl_player) -- Version 1.5
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All rights
reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

Attempting to auto-discover Bitporter pods (USB pods and Ethernet pods on
this subnet)...
        ...found 3 pods.
|==========|=================|
|   Pod    | (Ethernet only) |
|   Name   | Owner's IP addr |
|==========|=================|
| usb12345 |  <not tracked>  |
| net12346 | 192.168.1.123   |
| net12347 |   <available>   |
|==========|=================|
Attempting to connect to user-specified pod(s):
WARNING: User-specified pod "usb12348" not found/not connected.
Successfully opened Bitporter pod usb12345.

Entering ISP programming mode...
Reading Device ID code...
IDCODEVALID=1
IDCODE=00100000 00000000 00010110 01000001
Exiting ISP programming mode...
Exit code = 0... Success
```

The STAPL Player attempts to connect to the pods in the order listed, then plays the specified Action using the first successful connection. Unsuccessful connection attempts are reported as WARNINGS.

Configuring the Bitporter Pod's IP address (-i Option)

The user may query a pod's current Ethernet IP configuration with the `-iq` option, set the configuration to dynamic/DHCP with the `-id` option, or set a static IP with the `-is` option.

Bitporter Pod MAC Addresses

When adding devices to a network, network administrators frequently need to know the MAC address of the device. The Bitporter pods are each shipped with documents describing their MAC address. If the MAC address documentation for pod is unavailable, the pod's MAC address can be calculated by one of the following methods:

- **Automatic MAC address calculation tool**

  Achronix has provided a web page which customers can use to look up their pod's MAC address based upon the pod's serial number (the serial number is a five digit decimal number printed on a label stuck to the pod). The MAC address calculation website is: https://secure.achronix.com/bitportermac.php

- **Manual MAC address calculation**

  The MAC address can also be manually derived from the pod's serial number.

  MAC address = Manufacturer (prefix) : serial number (suffix) = 00:0F:75:00:30:39

  Where:

  Manufacturer = 00:0F:75

  Serial number = 12345 (decimal) = 0x3039 (hex) = 0x003039 = 00:30:39

Checking the Bitporter Pod's Current Ethernet IP Configuration

Note: *This option works whether the pod is connected via USB or via Ethernet.*

Simply run acx_stapl_player with the `-iq` option. If multiple pods are connected, the pod name must be specified with the `-p` option.

Example

To query the IP configuration of a USB pod with serial number 54321:

```
acx_stapl_player.exe -iq -pusb54321
```

The command returns:

```
Achronix STAPL Player (acx_stapl_player)
(c) Copyright 2007-2009 Achronix Semiconductor Corp.  All rights reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

*****************************************
* Checking current Ethernet configuration: *
*****************************************
Attempting to auto-discover Bitporter pods (USB pods and Ethernet pods on
this subnet)...
        ...found 1 pods.
|==========|=================|
|   Pod    | (Ethernet only) |
|   Name   | Owner's IP addr |
|==========|=================|
| usb54321 |  <not tracked>  |
|==========|=================|
```

```
Attempting to connect to user-specified pod(s):
Successfully opened Bitporter pod usb54321.

Current Bitporter IP Configuration:
        DHCP = on
```

Configuration the Pod for DHCP (Dynamic IP Address)

By default, Bitporter pods are configured to obtain their IP address via DHCP.

Note: *At this time, there is no way to ask the Bitporter pod for its current IP address. If a DHCP-configured pod's IP address needs to be known (for example, to connect to a pod on a different subnet), The recommend method is to use the MAC address to lookup the IP in the network's routing table. A future version of the acx_stapl_player will support querying a named pod for its current IP address.*

Reconfiguring from Static IP to DHCP

Changing the Ethernet IP configuration of a pod requires a USB connection. It is not possible to change the IP configuration of a pod that is currently connected via Ethernet.

Note: *Always consult the network administrator to verify that the pod will be allowed to acquire an IP address via DHCP.*

To change the IP configuration:

1. Either disconnect the pod from the target or power-down the target. The pod must always be powered when the connected target is powered.
2. Disconnect the Ethernet cable from the pod.
3. Disconnect the power cable from the pod.
4. Connect the USB cable to the host and the pod.
5. On the host, run acx_stapl_player with the `-id` option.

For example, to configure a pod with serial number 12345 to obtain its IP address via DHCP, enter the following command:

```
acx_stapl_player.exe -id -pusb12345
```

The command returns

```
Achronix STAPL Player (acx_stapl_player)
(c) Copyright 2007-2009 Achronix Semiconductor Corp.  All rights reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

****************************************
* Checking current Ethernet configuration: *
****************************************
Attempting to auto-discover Bitporter pods (USB pods and Ethernet pods on
this subnet)...
        ...found 1 pods.
|==========|=================|
|   Pod    | (Ethernet only) |
|   Name   | Owner's IP addr |
|==========|=================|
| usb12345 |  <not tracked>  |
|==========|=================|
Attempting to connect to user-specified pod(s):
Successfully opened Bitporter pod usb12345.
```

```
Setting new network configuration...
        Success!
Updated Bitporter IP Configuration:
        DHCP = on
```

Configuring the Pod to Use a Static IP Address

Reconfiguring the pod to use a static IP requires contacting the network administrator to determine the required information:

- The pod's new static IP address, example: 192.168.1.123
- The network mask, example: 255.255.255.0
- The gateway IP address, example: 192.168.1.1

Note: *Changing the Ethernet IP configuration of a pod requires a USB connection. It is not possible to change the IP configuration of a pod that is currently connected via Ethernet.*

To change the pod to use a static IP address:

1. Either disconnect the pod from the target or power-down the target. The pod must always be powered when the connected target is powered.
2. Disconnect the Ethernet cable from the pod.
3. Disconnect the power cable from the pod.
4. Connect the USB cable to the PC and the pod.
5. (Optional) Verify that the stapl player can see the pod over the USB connection: `acx_stapl_player -q`
6. On the host, run acx_stapl_player with the `-is,<pod IP>,<network mask>,<gateway IP>` option.

For example, to configure a pod with serial number 12345 to use the static IP 192.168.1.123, enter:

```
acx_stapl_player.exe -is,192.168.1.123,255.255.255.0,192.168.1.1 -pusb12345
```

This command returns:

```
Achronix STAPL Player (acx_stapl_player)
(c) Copyright 2007-2009 Achronix Semiconductor Corp.  All rights reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

****************************************
* Checking current Ethernet configuration: *
****************************************
Attempting to auto-discover Bitporter pods (USB pods and Ethernet pods on
this subnet)...
        ...found 1 pods.
|==========|=================|
|   Pod    | (Ethernet only) |
|   Name   | Owner's IP addr |
|==========|=================|
| usb12345 |  <not tracked>  |
|==========|=================|
Attempting to connect to user-specified pod(s):
Successfully opened Bitporter pod usb12345.
```

```
Setting new network configuration...
        Success!
Updated Bitporter IP Configuration:
        DHCP = off
        ip = 192.168.1.123
        mask = 255.255.255.0
        gateway = 192.168.1.1
```

Programming a Device

Once the user has generated a bitstream for the design, the following steps are used to program the device:

1. Connect the Bitporter pod to the target systems per **Chapter 3 – "Connecting the Bitporter Pod"**.
2. Open a command prompt of the host system (**Start** → **Run**, then enter **CMD**, and click **OK**; or **Start** → **All Programs** → **Accessories** → **Command Prompt**).
3. Locate the directory containing the programming file, and then download the bitstream:

   acx_stapl_player -aprogram *<design_name>*.jam

   The program returns:

```
Achronix STAPL Player (acx_stapl_player) -- Version 1.5
(c) Copyright 2007, 2008 Achronix Semiconductor Corp.  All
rights reserved.

contains elements of Jam STAPL Player Version 2.5 (20040526)
Copyright (C) 1997-2004 Altera Corporation

Attempting to auto-discover Bitporter pods (USB pods and
Ethernet pods on this subnet)...
        ...found 1 pods.
|==========|=================|
|   Pod    | (Ethernet only) |
|   Name   | Owner's IP addr |
|==========|=================|
| usb12345 |  <not tracked>  |
|==========|=================|
Successfully opened Bitporter pod usb12345.

Entering ISP programming mode...
Performing bulk erase on the device...
Programming the SPD60-FBGA1892 device...
Starting user mode...
Exiting ISP programming mode...
Exit code = 0... Success
```

Revision History

The following table lists the revision history of this document.

Version	Revision
1.0	Initial release version.
1.1	• Document title changed from *ACE Programming Guide* to *Bitporter User Guide*. • Added support for Ethernet-connected Bitporter pods. • Added support for Linux installations. • Added the following sections to **Chapter 4 – "Using the Achronix STAPL Player"**: **"Bitporter Pod Naming Conventions,"** on page 20, **"Connecting to Specific Pods by Name (-p option),"** on page 29, **"Configuring the Bitporter Pod's IP address (-i Option),"** on page 30, **"The Action read,"** on page 23, **"The Action read_state,"** on page 25, and **"The Action erase,"** on page 27.

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP