

Cyclone III FPGA Starter Kit

User Guide



101 Innovation Drive San Jose, CA 95134 www.altera.com

P25-36228-03

Document Version:1.2Document Date:July 2010

© 2010 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX are Reg. U.S. Pat. & Tm. Off. and/or trademarks of Altera Corporation in the U.S. and other countries. All other trademarks and service marks are the property of their respective holders as described at www.altera.com/common/legal.html. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and service as at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera. Altera customers are advised to obtain the latest version of device

specifications before relying on any published information and before placing orders for products or services.



Part Number UG-01018-1.2

Altera Corporation July 2010

ii Cyclone III FPGA Starter Kit User Guide



Contents

Chapter 1. Getting Started

Introduction	. 1–1
Before You Begin	. 1–2
Further Information	
Software Installation	1–2
Installing the Cyclone III FPGA Starter Kit	1–2
Installing the Quartus II Web Edition Software	1–4

Chapter 2. Development Board and Control Panel Setup

Development Board Setup	2–1
Requirements	
Powering Up the Development Board	
Installing the USB-Blaster Driver	
Control Panel Setup	
Configuring the FPGA Using the Quartus II Programmer	

Chapter 3. Using the Control Panel

-1
-1
-2
-2
3
-3
4
-5
-5
-5
-6
-6
7

Chapter 4. Measuring Power on the Cyclone III Starter Board

Introduction	4–1
Measuring Power	4–2
Changing the Example Design	4–3

Appendix A. Programming the Configuration Flash Device

OverviewA	.–1
Creating a Flash-Programmable POF File A	-1
Programming the Flash Device	-5

Additional Information

Revision History	. Info–i
How to Contact Altera	Info-ii
Typographic Conventions	Info-ii



1. Getting Started

Introduction

Welcome to the Altera[®] Cyclone[®] III FPGA Starter Kit, which includes a full-featured field-programmable gate array (FPGA) development board, hardware and software development tools, documentation, and accessories needed to begin FPGA development.

The development board includes an Altera Cyclone III FPGA and comes preconfigured with a hardware reference design stored in flash memory. You can use the development board as a platform to prototype a variety of FPGA designs.

The starter kit provides an integrated control environment that includes a software controller in a control panel application, a USB command controller, a multi-port SRAM/DDR SDRAM/flash memory controller, and example designs specified in Verilog code. You can use this design as a starting point for test designs.

This user guide addresses the following topics:

- How to set up, power up, and verify correct operation of the development board.
- How to install the Cyclone III FPGA Starter Kit.
- How to install the Altera[®] Quartus II Web Edition software.
- How to set up and use the control panel, a graphical user interface (GUI), to manipulate components on the board, implement applications.
- How to configure the Cyclone III FPGA.
- How to set up and run example designs.

For complete details on the development board, refer to the *Cyclone III FPGA Starter Board Reference Manual*.

Before You	Before proceeding, check the contents of the kit:
Begin	 Cyclone III FPGA Starter Development Board 12-V DC power supply USB cable

For the most up-to-date information on this product, visit the Altera website at www.altera.com/products/devkits/altera/kit-cyc3-starter.html.

Further Information

For other related information, refer to the following websites:

For More Information About	Refer To
Additional daughter cards available for purchase	www.altera.com/products/devkits/ kit-daughter_boards.jsp
Cyclone III handbook	www.altera.com/literature/lit-cyc3.jsp
Cyclone III reference designs	http://www.altera.com/products/devkits/altera/kit- cyc3-starter.html
eStore if you want to purchase devices	www.altera.com/buy/devices/buy-devices.html
Cyclone III Orcad symbols	www.altera.com/support/software/download/pcb/ pcbpcb_index.html
Nios [®] II 32-bit embedded processor solutions	www.altera.com/technology/embedded/ emb-index.html

Software Installation

This section describes the following procedures:

- "Installing the Cyclone III FPGA Starter Kit"
- "Installing the Quartus II Web Edition Software" on page 1–4

Installing the Cyclone III FPGA Starter Kit

The license-free Cyclone III FPGA Starter Kit installer includes all the documentation and design examples for the kit.

To install the Cyclone III FPGA Starter Kit, follow these steps:

1. Download the Cyclone III FPGA Starter Kit installer from the Cyclone III FPGA Starter Kit page of the Altera website. Alternatively, you can request a development kit DVD from the Development Kits, Daughter Cards & Programming Hardware page of the Altera website.

٦

2. Follow the on-screen instructions to complete the installation process.

The installation program creates the Cyclone III FPGA Starter Kit directory structure shown in Figure 1–1.

Figure 1–1. Cyclone III FPGA Starter Kit Default Installed Directory Structure



Table 1–1 lists the file directory names and a description of their contents.

Table 1–1. Installe	ed Directory Contents
Directory Name	Description of Contents
board_design_files	Contains schematic, layout, assembly, and bill of material board design files. Use these files as a starting point for a new prototype board design.
demos	Contains demonstration projects that may or may not contain up-to-date source code.
documents	Contains the development kit documentation.
examples	Contains the example design files for the Cyclone III FPGA Starter Kit
factory_recovery	Contains programming files for returning board to factory default condition.

Installing the Quartus II Web Edition Software

The Quartus II Web Edition software provides the necessary tools for developing hardware and software for Altera FPGAs. Included in the Quartus II Web Edition software are the Quartus II software, the Nios II EDS, and the MegaCore[®] IP Library. The Quartus II software (including SOPC Builder) and the Nios II EDS are the primary FPGA development tools for creating the reference designs in this kit.

To install the Quartus II Web Edition software, follow these steps:

- 1. Download the Quartus II Web Edition software from the Quartus II Web Edition Software page of the Altera website. Alternatively, you can request a DVD from the Altera IP and Software DVD Request Form page of the Altera website.
- 2. Follow the on-screen instructions to complete the installation process.
 - If you have difficulty installing the Quartus II software, refer to Quartus II Installation & Licensing for Windows and Linux Workstations.

The Quartus II Web Edition software includes the following items:

- Quartus II software—The Quartus II software, including the SOPC Builder system development tool, provides a comprehensive environment for system-on-a-programmable-chip (SOPC) design. The Quartus II software integrates into nearly any design environment and provides interfaces to industry-standard EDA tools.
 - To compare the Quartus II subscription and web editions, refer to *Altera Quartus II Software—Subscription Edition vs. Web Edition.* The kit also works with the subscription edition.
- MegaCore IP Library—A library that contains Altera IP MegaCore functions. You can evaluate MegaCore functions with the OpenCore Plus feature to perform the following tasks:
 - Simulate behavior of a MegaCore function in your system
 - Verify functionality of your design, and quickly and easily evaluate its size and speed
 - Generate time-limited device programming files for designs that include MegaCore functions
 - Program a device and verify your design in hardware

The OpenCore Plus hardware evaluation feature is an evaluation tool for prototyping only. You must purchase a license to use a MegaCore function in production.



- For more information about OpenCore Plus, refer to *AN 320: OpenCore Plus Evaluation of Megafunctions*.
- Nios® II Embedded Design Suite (EDS)—A full-featured tool set that allows you to develop embedded software for the Nios II processor which you can include in your Altera FPGA designs.

Licensing Considerations

The Quartus II Web Edition software is license-free and supports Cyclone III devices without any additional licensing requirement. This kit also works with the Quartus II Subscription Edition software, after you obtain the proper license file. To purchase a subscription, contact your Altera sales representative.



2. Development Board and Control Panel Setup

Development Board Setup

The development board is preloaded with an example design to demonstrate the Cyclone[®] III device and board features. At power-up, the preloaded design also enables you to quickly confirm that the board is operating correctly.

Figure 2–1 shows the Cyclone III development board layout and components.



Figure 2–1. Cyclone III Development Board Layout and Components

Requirements

Before you proceed, ensure that the follwing items are installed:

- Altera[®] Quartus[®] II software on the host computer
- Cyclone III FPGA Starter Kit
- USB-BlasterTM driver software on the host computer. The Cyclone III FPGA starter development board includes an integrated USB-Blaster circuitry for FPGA programming.

Powering Up the Development Board

To power-up the development board, follow these steps:

- 1. Ensure that the ON/OFF switch (SW1) is in the OFF position (up).
- 2. Connect the USB-Blaster cable from the host computer to the USB-Blaster port on the development board.
- 3. Connect the 12-V DC adapter to the development board and to a power source.



Only use the supplied 12-V power supply. Power regulation circuitry on the board could be damaged by supplies greater than 12 V.

- 4. Press the power switch (SW1).
- 5. Confirm that all four user LEDs are ON.

Installing the USB-Blaster Driver

The Cyclone III FPGA development board includes an integrated USB-Blaster circuitry for FPGA programming. However, for the host computer and board to communicate, you must install the USB-Blaster driver on the host computer.

Installation instructions for the USB-Blaster driver are available on the Altera website at www.altera.com/support/software/drivers/ dri-index.html. On the "Altera Programming Cable Driver Information" page of the Altera website, locate the table entry for your configuration and click the link to access the instructions.

Control Panel Setup

Setting up the control panel involves the following:

- Configuring the FPGA
- Starting the control panel
- Power up the board and ensure that is is operational.

For more information about using the control panel, refer to the "Using the Control Panel" chapter.

Configuring the FPGA Using the Quartus II Programmer

You can use the Quartus II Programmer to configure the FPGA with a specific **.sof**. Before configuring the FPGA, ensure that the Quartus II Programmer and the USB-Blaster driver are installed on the host computer, the USB cable is connected to the development board, power to the board is on, and no other applications that use the JTAG chain are running.

To configure the Cyclone III FPGA, follow these steps:

- 1. Start the Quartus II Programmer.
- 2. Click Add File and select the path to the desired .sof.
- 3. Turn on the **Program/Configure** option for the added file.
- 4. Click **Start** to configure the selected file to the FPGA. Configuration is complete when the progress bar reaches 100%.



3. Using the Control Panel

Overview

The control panel consists of the following:

- The graphical user interface (GUI) application on the host computer
- The standard Nios II hardware design running on the board's Cyclone III FPGA device

After installing the Cyclone III FPGA Starter Kit, you can locate the control panel for the hardware and software in the *<kit path>\demos\control_panel* directory.

The design downloaded to the Cyclone III device implements a command controller that processes board commands sent over the USB-Blaster from the control panel. To perform the appropriate actions, the command controller communicates with the controller of the targeted board I/O device.

You can perform the following actions with the control panel:

- Light up LEDs
- Detect push button presses
- Read from and write to the DDR SDRAM, SRAM, flash memory, and on-chip RAM

The following sections describe how to perform the above actions with the control panel already open on the host computer. If not already open, launch the control panel as described in "Control Panel Start".

Control Panel Start

The Cyclone III development board is shipped with an example design stored in the flash memory which configures the Cyclone III FPGA upon power-up with the standard Nios II design.

For an older version of the Cyclone III development board shipped with the Cyclone III FPGA Starter Kit v7.1.0, v7.2.0, or 8.0.0 application, you must manually configure the **cycloneIII_3c25_start_niosII_standard.sof** into the FPGA before launching the control panel application. You can locate the source for the example design in the *<kit path>\examples\cycloneIII_3c25_starter_board_standard* directory.

To launch the control panel user interface, run the **control_panel.exe** program found in the *<kit path*>**demos****control_panel** directory (Figure 3–1).

Figure 3–1. Control Panel Window



LEDs and Buttons

Illuminating LEDs

To illuminate an LED, follow these steps:

- 1. The **LED & Buttons** tab should be visible when the application runs. If it is not visible, click the **LED & Buttons** tab (Figure 3–2).
- 2. Click on LEDs to individually turn on the LEDs.

Buttons Indicators

1. Press the push-button switches on the board. Notice that buttons on the GUI change accordingly.

Figure 3–2. Control Panel Window for LEDs and Buttons



DDR SDRAM/ SSRAM/On-Chip Controller

You can perform the following types of memory read/write operations with the control panel:

- Read from and write to the DDR SDRAM, SSRAM, or on-chip device
- Write entire contents of a file, to the DDR SDRAM, SSRAM, or on-chip device
- Read contents of the DDR SDRAM, SSRAM, or on-chip device, to a file

The following sections describe how to access the DDR SDRAM. You can use the same procedure to access the SSRAM.

Read/Write Data

To read from and write to the DDR SDRAM, follow these steps:

 Click the DDR SDRAM tab (Figure 3–3). The Address column indicates the hex address of the DDR SDRAM. The values inside the 0-3, 4-7, 8-B, and C-F columns are the DDR SDRAM contents in hex words format.

Figure 3–3. Control Panel DDR SDRAM Tab

	J4) - 32MB (16 x	DRAM Memory Flash	- //	.hip Memory 100 to 0x5FFFF1	T
art Address: 020	and the second second second	24) Add 835	range. 0x40000		Save File Load File
Address	0-3	4 - 7	8 - B	C - F	Data to Ascii-Text
2000000	00000000	00000000	00000000	00000000	
2000010	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFF	
2000020	00000000	00000000	FFFFFFFF	00000000	
2000030	00000000	00000000	00000000	FFFFFFFF	
2000040	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000050	FFFF0000	FFFF0000	FFFFFFFF	FFFFFFFF	
2000060	02020202	03030303	OOFFOOFF	OOFFOOFF	
2000070	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000080	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000090	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20000A0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000080	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20000c0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20000D0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20000E0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20000F0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000100	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000110	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000120	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000130	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000140	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000150	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000160	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000170	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000180	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
2000190	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20001A0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20001B0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20001C0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20001D0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	
20001E0	OOFFOOFF	OOFFOOFF	OOFFOOFF	OOFFOOFF	

2. To write a 32-bit word to the DDR SDRAM, click the desired location, enter the desired value in hex format, and press **Enter**.

Read from a File

To read the contents of a file and load it to the DDR SDRAM, follow these steps:

- 1. Click Load File.
- 2. Browse to **sample.txt** located in the **control_panel** directory and click **Open**. This step instantiates the DDR SDRAM controller and loads the text contents into the DDR SDRAM. Notice that the **Data to Ascii-text** column shows the DDR SDRAM contents in Ascii value.

Write to a File

To write the contents of the DDR SDRAM to a file, follow these steps:

- 1. Click Save File.
- 2. Enter the start and end addresses of the DDR SDRAM.
- 3. Choose a file name and click **Save**. This instantiates the controller to read the DDR SDRAM contents from the start address to the end address, and write the contents to a file.

Flash Memory Programmer

You can perform the following operations to read from and write to the board's flash memory with the control panel:

- Perform a CFI query of flash memory
- Erase select blocks of flash memory
- Write 32-bit hex word to flash memory
- Write a binary file to flash memory
 - Load the contents of the flash memory into a file



Do not exit from the control panel while erasing the flash memory.

Flash Memory Tab

To use the flash memory functions, click the **Flash Memory** tab (Figure 3–4).

Figure 3–4. Control Panel Flash Memory Tab

000	CFI Query	Reset	Frase Block	
			LI USE DIOLK	
0-3	4 - 7	8 - B	C - F	Data to Ascii-Text
54686973	20697320	61207361	6D706C65	This is a sample
20746578	74206669	6C652070	726F7669	text file provi
64656420	666F7220	74686520	6578616D	ded for the exam
706C6520	6F662077	72697469	6E672061	ple of writing a
2066696C	6520746F	206D656D	6F727920	file to memory
7573696E	6720416C	30217261	20437963	using AlOira Cyc
6C6F6E65	20494949	20537461	72746572	lone III Starter
20486974	20436F6E	74726F6C	2050616E	Kit Control Pan
656C2061	70706069	63617469	6F6E2E20	el application.
49662074	6865206D	656D6F72	79207772	If the memory wr
69747465	6E20746F	20697320	466C6173	itten to is Flas
68207468	6E652074	68652076	6F6C6174	h thne the volat
696C6520	636F6E74	656E7473	2077696C	ile contents wil
6C207065	72736973	74207570	6F6E2070	l persist upon p
6F776572	20637963	6C696E67	20746865	ower cycling the
20626F61	72642C20	486F7765	00000000	board, Howe
74686973	2066696C	65206361	6E206265	this file can be
20657261	73656420	62792063	6C69636B	erased by click
696E6720	74686520	45726173	65206275	ing the Erase bu
74746F6E	2E204F74	68657220	6D656D6F	tton. Other memo
20606160	2020606C	20206560	61726120	`a` `l e`ara
7460616C	63204064	64612061	20436120	t`alc @dda a Ca
64686420	20494140	20006061	20706572	dhd IA@ .`a per
20426164	20406F66	64224D64	20406068	Bad @ofd"Md @`h
206C2061	60706869	00002061	616E2020	l a`phi aan
41202064	68642065	64206F62	61206170	A dhd ed oba ap
20602064	68206420	20692020	44686163	`dh d i Dhac
68206420	68642074	68652060	6E646174	h d hd the `ndat
69646520	60206264	646C7020	20656864	ide `bddlp ehd
64206064	62206060	64206160	62622060	d `db ``d a`bb `
64206460	20612062	6C606C61	20204865	d d` a bl`la He
	20746578 6455420 70666520 20666960 20466960 20486974 65622061 49662074 6572061 65776572 204267468 69207468 69207468 69207468 69207468 69207468 69207468 692074572 00506160 744566820 20426164 20622061 41202064	$\begin{array}{cccccc} 20746578 & 74206669 \\ 6455420 & 66567220 \\ 706c6520 & 66662077 \\ 706c6520 & 6760416 \\ 6455420 & 6720416 \\ 73736956 & 6720416 \\ 6c676855 & 20494949 \\ 536c2051 & 70706c59 \\ 856c2051 & 70706c59 \\ 89620746 & 68552050 \\ 69747455 & 68207466 \\ 685207466 & 68552074 \\ 696c6520 & 63569674 \\ 6207466 & 68552074 \\ 6207665 & 72736973 \\ 67776572 & 20637963 \\ 20626761 & 72642020 \\ 74686973 & 2066696 \\ 2066510 & 2020605 \\ 7466673 & 206696 \\ 2066510 & 2020405 \\ 7466673 & 206646 \\ 2060516 & 6320406 \\ 20662164 & 2040676 \\ 20662164 & 2040676 \\ 20662046 & 68642074 \\ 20662046 & 68642074 \\ 2062064 & 68264207 \\ 68206420 & 68642074 \\ 2042066 & 6220606 \\ \end{array}$	$\begin{array}{c} 20746578 & 74206669 & 6c52070 \\ 6455420 & 6667220 & 7468520 \\ 706c5520 & 6672077 & 72697469 \\ 2066696c & 6520746F & 2069569 \\ 70736968 & 6720416C & 30217261 \\ 6c676865 & 20494949 & 20537461 \\ 6c67686 & 205206 & 6550672 \\ 6c7776572 & 20637963 & 6c696867 \\ 72736973 & 74207370 & 6c696676 & 6520361 \\ 206520 & 6356420 & 665520 & 65562720 \\ 2066567 & 74586420 & 662792063 \\ 6c696672 & 74586520 & 45727657 \\ 20665670 & 74586420 & 662792063 \\ 6c686420 & 20290464 & 64612061 \\ 20426164 & 2049474 & 206051 \\ 20426164 & 20496766 & 64224064 \\ 20662164 & 204906766 & 64224064 \\ 20662164 & 204906766 & 64224064 \\ 20662616 & 65200665 & 642260661 \\ 202662064 & 682642074 & 6655220 \\ 20602616 & 682642074 & 6652206 \\ 68206420 & 68642074 & 665206 \\ 68206420 & 2069661 \\ 2062064 & 682642074 & 665206 \\ 68206420 & 206961 \\ 20626264 & 68264274 & 6652206 \\ 68206420 & 2069616 \\ 20206264 & 682642074 & 665206 \\ 68206420 & 206961 \\ 2062616 & 6220606 & 64226066 \\ 2020606 & 68206420 \\ 20602640 & 682642074 & 665206 \\ 68206420 & 006616 \\ 20206264 & 68206420 & 2065206 \\ 68206420 & 006661 \\ 20620640 & 68264074 & 6652060 \\ 68206420 & 006616 \\ 62206064 & 6220606 & 6420676 \\ 68206420 & 006616 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 00002061 \\ 000002061 \\ 00002061 \\ 00002061 \\ $	20746578 74206669 6c652010 726F7669 6455420 6667220 74666520 65786165 706c6520 6667220 74666520 65786165 706c6520 66662077 72697469 65872061 706c6520 6762077 72697469 65872061 70736968 67204162 30217261 20437963 73736968 67204162 30217261 724637863 6768652 20494949 20537461 72746572 20485974 20436766 6452016 656672 9462074 6865200 6556672 79207772 67747465 68207468 20697320 466c6173 69207468 68652074 68552076 67662173 69207468 68652074 68552076 67662173 69207468 2062774 686572 20776962 620706572 20637963 6C696672 20746865 726572 20637963 6C596672 20746865 746869732 2065466 22017570

CFI Query

The common flash interface (CFI) flash memory devices conform to basic flash commands. The most basic command is Query which switches the device into a ROM table mode so that features of the flash device are determined by reading values from the table.

To perform a CFI query using the host application, click **CFI Query**. Notice that the memory table displays contents that correlate with the table contents as described in the device datasheet.

To put the flash device back in user mode, press **Reset** on the control panel.

Read/Write Data

To read from and write to the flash memory, follow these steps:

		Address range:				
				Contraction of Contractory Contractory		
Start Address: 000	000000	CFI Query	Reset	Erase Block		
Address	0 - 3	4 - 7	8 - B	C - F	Data to Ascii-Text	
00000000	54686973	20697320	61207361	6D706C65	This is a sample	-
00000010	20746578	74206669	6C652070	726F7669	text file provi	
00000020	64656420	666F7220	74686520	6578616D	ded for the exam	
00000030	706C6520	6F662077	72697469	6E672061	ple of writing a	
00000040	2066696C	6520746F	206D656D	6F727920	file to memory	
00000050	7573696E	6720416C	30217261	20437963	using AlO!ra Cyc	-
00000060	6C6F6E65	20494949	20537461	72746572	lone III Starter	
00000070	204B6974	20436F6E	74726F6C	2050616E	Kit Control Pan	-
00000080	656C2061	70706C69	63617469	6F6E2E20	el application.	-
00000090	49662074	6865206D	656D6F72	79207772	If the memory wr	
0A000000	69747465	6E20746F	20697320	466C6173	itten to is Flas	-
00000080	68207468	6E652074	68652076	6F6C6174	h thne the volat	-
000000c0	696C6520	636F6E74	656E7473	2077696C	ile contents wil	
000000000	6C207065	72736973	74207570	6F6E2070	l persist upon p	-
000000E0	6F776572	20637963	6C696E67	20746865	ower cycling the	
000000F0	20626F61	72642C20	486F7765	00000000	board, Howe	-
00000100	74686973	2066696C	65206361	6E206265	this file can be	-
00000110	20657261	73656420	62792063	6C69636B	erased by click	-
00000120	696E6720	74686520	45726173	65206275	ing the Erase bu	
00000130	74746F6E	2E204F74	68657220	6D656D6F	tton. Other memo	-
00000140	20606160	2020606C	20206560	61726120	`a` `l e`ara	-
00000150	7460616C	63204064	64612061	20436120	t'alc @dda a Ca	-
00000160	64686420	20494140	20006061	20706572	dhd IA@ .`a per	-
00000170	20426164	20406F66	64224D64	20406068	Bad @ofd"Md @`h	-
00000180	206C2061	60706869	00002061	616E2020	l a`phi aan	-
00000190	41202064	68642065	64206F62	61206170	A dhd ed oba ap	-
000001A0	20602064	68206420	20692020	44686163	`dh d i Dhac	-
00000180	68206420	68642074	68652060	6E646174	h d hd the `ndat	-
000001c0	69646520	60206264	646C7020	20656864	ide bddlp ehd	
000001D0	64206064	62206060	64206160	62622060	d 'db 'd a bb '	-
000001E0	64206460	20612062	6C606C61	20204865	d d` a bl`la He	-
						-
			ADTERA		Ex	

Figure 3–5. Control Panel Flash Memory Tab

- 1. Click **Erase Block** to perform a block erase of the flash memory. The **Address** column indicates the hex address of the flash memory. The values inside the **0-3**, **4-7**, **8-B**, and **C-F** columns are the flash memory contents in hex words format.
- 2. To write a 32-bit word to the flash memory, click the desired location, enter the desired value in hex format, and press **Enter**.



4. Measuring Power on the Cyclone III Starter Board

Introduction

One of the main features of the Cyclone[®] III device is its low power consumption. You can measure the power of the 3C25 device on the Cyclone III starter board under various conditions with an example design provided with the kit.

The power example design allows you to control the amount of logic utilized in the FPGA, the clock frequency, the number of I/Os being used, and measure the effect on the power to the Cyclone III device. Because the Cyclone III starter board has only four buttons and four LEDs, interaction with the board is minimal as defined below.

Table 4–1 describes the functionality of the four input buttons that control the power example design.

Table 4–1. Four l	Table 4–1. Four Input Button Functionality							
Button	FPGA Pin	Туре	Description					
1	F1	Reset	Resets the demo to the beginning, node i_nrst.					
2	F2	Toggle	Advances the example design to the next higher frequency, node <code>i_nfreq_next</code> .					
3	A10	Toggle	Advances the example design to the next higher resource utilization, node_i_nperc_next.					
4	B10	Press and Hold	Enables the outputs to toggle, node i_noutput_ena.					

Tables 4–2 and 4–3 describe how the LEDs indicate the example design's current power state.

Table 4–2. LEDs Power State (Frequency)							
Dioplaya	L	EDs	State	Clock Frequency			
Displays	MSB	LSB	Sidie	(MHz)			
Frequency	LED2	LED1	00	0			
			01	33			
			10	67			
			11	100			

Table 4–3. LEDs Power State (Resources)							
Dianlava	LE	Ds	State	% of Design Hand			
Displays	MSB	LSB	State	% of Design Used			
Resources	LED4	LED3	00	25%			
			01	50%			
			10	75%			
			11	100%			

The design used for power measurement is a replicated set of randomly filled ROMs that feed a multiplier block and a shift register that is fed by a signal that changes every clock cycle. Tables 4–2 and 4–3 show the power state which represent the percent of the full design used. As compiled, this full design uses:

- Logic elements: 22,493/24,624 (91%)
- Combinational functions: 1,961/24,624 (8%)
- Dedicated logic registers: 21,133/24,624 (86%)
- Total registers: 21,133
- Total pins: 73/216 (34%)
- Total memory bits: 524,288/608,256 (86%)
- Embedded Multiplier 9-bit elements: 128/132 (97 %)
- Total PLLs: 1/4 (25%)

Measuring
Power

The example design is located in

<*kit install*>**examples****cycloneIII_3c25_start_power_demo**. Configure the FPGA with the **.sof** found in the directory.

The input clock (i_clk PIN_B9) is the 50-MHz oscillator on the board, which generates the input clock for the reference design through a PLL

For more information on configuring the FPGA, refer to "Configuring the FPGA Using the Quartus II Programmer" on page 2–3.

Current sense resistors $(0.010 \Omega \pm 1\%)$ are installed at locations JP6 (FPGA core power) and JP3 (FPGA I/O power + other device I/O power). With a digital multimeter set to mV measurement range, the resistor at location JP6 measures the core power. The resistor at location JP3 measures the I/O power. To measure the current being used in various configurations, use the following steps:

To obtain the power (P) in milliwatts, measure *<Measured Voltage>* (the voltage across the sense resistors at JP6 or JP3) in mV and calculate the nominal power using the equation:

P = 100 x <*Measured Voltage*> x <*Supply Voltage*>

where *<Supply Voltage>* is 1.2 V for JP6 and 2.5 V for JP3.

You can use the four input buttons to advance through the various power state as outlined in Table 4–2. Notice how current increases as frequency and resource usage increase.

You can also measure the I/O power consumed by measuring the voltage across sense-resistor JP3 when Button 4 is pressed and held. Because this 2.5-V power rail is shared with other devices, there is a nominal 100 mW that must be subtracted from the calculated I/O power to obtain the FPGA I/O power.

The number of I/O pins used is controlled by the resource state (shown in Tables 4–2 and 4–3). For each increment in resources, 16 additional I/O pins are added (refer to Table 4–4).

Table 4–4. I/O Pin & Resource State					
LED4/LED3	Number of I/O Pins				
00	16				
01	32				
10	48				
11	64				

Similarly, the toggle-frequency of these I/O pins is set by the overall design frequency (refer to Table 4–1).

Changing the Example Design

The source code for the Cyclone III power example design is also provided so you can use it as a starting point for your own measurements. You can adjust the number of outputs by changing parameter NUM_OUTPUTS_PER_STAMP. The default is 16, which for four resource percentage steps equates to 16 x 4 = 64.

The appropriate pins to be used as outputs are pre-assigned to the HSMC connector (J1). If you would like to look at more than the 76 I/Os available on J1, you need to make the appropriate pin assignments.



Appendix A. Programming the Configuration Flash Device

Overview	The Intel® P30 flash device uses active parallel flash configuration to configure the Cyclone® III device on power up. The Cyclone III Starter Board has a factory default configuration programmed into the P30 flash; however, after developing your own project, you may want to replace this factory default configuration with your own. This appendix describes how to reprogram the Intel P30 flash device.
Creating a Flash- Programmable POF File	 After a Quartus II compilation, a Programmer Object File (.pof) is created. Before you can program this file into the Intel P30 flash device on the Cyclone III development board, you must modify the .pof by performing the following steps: 1. Choose Convert Programming File from the File menu. The Convert Programming Files window opens (refer to Figure A–1).

Figure A–1. Convert Programming Files Window

D 약 당행 중 사용 등 사용 가 reject Navigator Entry	- X / I I I I I I I I I I I I I I I I I I	
Completion Herarchy	Convert Programming Files Specify the kyot files to convert and the type of programming file to generate. You can also most type file is chromoton that are the conversion takes information created here for have one. Conversion stells Files Open Conjection Schip Data. Save Conversion Selap	1
Henorchy Prime of Design Units	Odput programming like poer Programming like poer Programming like poer Ordingstation device: EPC16 Mode: Total Passave Senail Plagame: Crides #00/pasame.throwodput.like pod device.cdl device.cdl Plagame: File poer Fil	rsion 8.0
E Concile Design P- Analysis 1.5yrthesis	Input Riss to convert	Documentation
Type Message		

- 2. Select the following settings:
 - **Programming File Type:** Programmer Object File (.pof)
 - Configuration Device: CFI_128MB
 - Mode: Active Parallel
 - File Name: Type the name of the flashable .pof to write

If you choose to overwrite the existing **.pof**, a warning message occurs.

3. Under Input file to the convert, select Configuration Master under SOF Data. Refer to Figure A-2.

Before moving to the next step, ensure that the setting for the **Configuration Device** is CFI_128MB.

Figure A-2. Input File to Convert

File/Data area	Properties	Start Address	Add <u>H</u> ex Dat
SOF Data	Page 0	<auto></auto>]
Configuration Master			Add Sof Data
High Byte (D[158])			
Low Byte (D[70])			Add File
			<u>R</u> emove
			Down
			Properties
		General	te Close

- 4. Click Add File.
- 5. Choose the **.sof** you want to convert and click **OK**.
- 6. Select **SOF Data** and click **Properties**. The **SOF Data Properties** window appears.
- 7. Select and type the following settings as shown in Figure A–3:
 - **Pages:** 0
 - Address mode for selected pages: Start
 - Start address (32-bit hexadecimal): 0x020000

Figure A–3. SOF Data Properties

□6 □7	
Selected pages comment: Page_()
Address mode for selected pages	
Start	•
Start address (32-bit hexadecima	ox020000
End address (32-bit hexadecimal	OXFFFFFFFF

- The flash address 0x20000 is the default starting address from which the Cyclone III device starts loading configuration data.
- 8. Click **OK**. Figure A–4 shows the updated **Convert Programming Files** window.

Figure A–4. Updated Convert Programming Files Window

ools	<u>W</u> indow				
		onvert and the type of progr file information from other file			created here for
	Conversion setup files				
	Open Cony	zersion Setup Data	<u>s</u>	ave Conversion Setup	»
	Output programming file				
	Programming file type:	Programmer Object File ((.pof)		-
	Options	Configuration device:	CFI_128MB - M	ode: Active Parallel	
	File name:	C:/temp/cyclonelll_3c2			
	Advanced	Remote/Local update dif			
		IV MEMORY MAD FILE			
		Memory Map File			
	Input files to convert	y♥ Memory Map rile			
	File/Data area	j⊻ memory map rilê	Properties	Start Address	Add <u>H</u> ex Data
	File/Data area ⊡ SOF Data		Properties Page 0	Start Address 0x00020000	
	File/Data area SOF Data Configuration	Master	Page 0		Add <u>H</u> ex Data Add <u>S</u> of Data
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0		Add Sof Data
	File/Data area □ SOF Data □configuration	Master _3c25_start_my_first_fpg 58])	Page 0		
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0		Add Sof Data
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0		Add <u>S</u> of Data Add <u>File</u> <u>R</u> emove
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0		Add <u>S</u> of Data Add <u>File</u>
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0		Add <u>S</u> of Data Add <u>File</u> <u>R</u> emove
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0		Add <u>S</u> of Data Add <u>E</u> ile <u>R</u> emove
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0		Add <u>S</u> of Data Add <u>E</u> ile <u>R</u> emove
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0		Add Sof Data Add Eile Remove Up Down Properties
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0		Add Sof Data Add Eie Remove Up
	File/Data area SOF Data 	Master _3c25_start_my_first_fpg 58])	Page 0	<u>b</u> c00020000	Add Sof Data Add Eile Remove Up Down Properties

- 9. Click **Generate**. If you are overwriting the input **.pof** you will receive a warning asking if you want to overwrite it. Click **Yes** to overwrite the file or enter a different filename. When the Quartus II software finishes converting the file, you can use the converted **.pof** to program the on-board parallel flash device.
 - The Quartus II software also generates a MAP file, which can help you debug issues with locations in the flash device.

Programming the Flash Device

Altera recommends that you do not overwrite the factory hardware and factory software images unless you are an expert with the Altera tools or deliberately overwriting the factory design. If you unintentionally overwrite the factory image, perform these flash programming instructions using the **cycloneIII_3c25_start_niosII_standard.pof** found in the **factory_recovery** directory for the object file in step 9.

To program the flash device, follow these steps:

- 1. Open the Quartus II Programmer.
- 2. Click **Auto Detect** from the button list to the left of the programming file list area.
- 3. Select the detected Cyclone III 3C25 device.
- 4. Choose **Attach Flash Device** (Edit menu). The **Select Flash Device** window opens.
- 5. Turn on the **Flash Memory** and **CFI_128MB** options (refer to Figure A–5).

Figure A–5. Select Flash Device

Select Flash Device	Device name CFI_128MB CFI_256MB CFI_256MB CFI_32MB CFI_512MB CFI_512MB CFI_64MB CFI_96MB	New Import Export
		Eemove Check
	ОК	Cancel

6. Click OK.

- 7. In the Quartus II Programmer, select the CFI_128MB device.
- 8. Click Change File from the button list at the left of the programming file area.
- 9. Select the converted .pof that you generated in the previous section.
- 17 To restore factory flash contents, choose cycloneIII_3c25_start_niosII_standard.pof located in the factory_recovery directory as your converted .pof.
- 10. Turn on the **Program/Configure** option for all devices shown in the Programmer.
 - [F
 - Turning on the option for the .pof enables all three options, which is what you want to do (refer to Figure A–6).

Figure A–6. POF Options

🏓 Start	File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine	Security Bit	Erase	ISP CLAMP
📲 Stop	Factory default PFL image	EP3C25	00000000	FFFFFFF	✓						
== 0.0p	C:/temp/cyclonelll_3c25	. CFI_128MB	0014E1AD								
Auto Detect	L Page_0				V						
X Delete											
🌽 Add File											
👺 Change File											
🗳 Save File											
😂 Add Device											
📫 Up											
🔑 Down											

- 11. Click Start. The Programmer loads the special flash programming hardware into the FPGA, which allows the Programmer to communicate with the flash device. The Programmer sends the .pof to the flash device via the flash programming hardware. The Quartus II Message window displays the bank addresses as they are erased and then written.
- 12. To configure the Cyclone III 3C25 with your design from the on-board flash device, either push the reconfiguration button or turn the Cyclone III Starter Board off and then on again.



Additional Information

Revision History

The table below displays the revision history for the chapters in this user guide.

Chapter	Date	Version	Changes Made
All	July 2010	1.2.0	 Removed "Licensing the Quartus II Software". Updated Figure 1–1 on page 1–3. Updated "Installing the Quartus II Web Edition Software" on page 1–4. Updated "Installing the Cyclone III FPGA Starter Kit" on page 1–2. Updated "Further Information" on page 1–2. Updated Copyright information.
All	March 2010	1.1.0	 Updated the directory structure in Figure 1–1. Updated "Control Panel Start" section and Figure 3–1. Updated "LEDs" section and Figure 3–2. Updated "DDR SDRAM/SSRAM Controller and Programmer" section and Figure 3–3. Updated "Flash Memory Programmer" section and Figure 3–4.
1, 2, 4	June 2008	1.0.1	 Updated directory structure figure and installed directory contents table. Updated the control panel user interface executable file name. Updated the kit directory path. Updated the configuration SOF file name. Updated kit's example design file name.
All	April 2007	1.0.0	First publication.

How to Contact Altera

For the most up-to-date information about Altera products, refer to the following table.

Contact Note (1)	Contact Method	Address
Technical support	Website	www.altera.com/support
Technical training	Website	www.altera.com/training
	Email	custrain@altera.com
Product literature	Website	www.altera.com/literature
Non-technical support (General)	Email	nacomp@altera.com
(Software Licensing)	Email	authorization@altera.com

Note to Table:

(1) You can also contact your local Altera sales office or sales representative.

Typographic Conventions

This document uses the typographic conventions shown below.

Visual Cue	Meaning	
Bold Type with Initial Capital Letters	Command names, dialog box titles, checkbox options, and dialog box options are shown in bold, initial capital letters. Example: Save As dialog box.	
bold type	External timing parameters, directory names, project names, disk drive names, filenames, filename extensions, and software utility names are shown in bold type. Examples: f _{MAX} , \qdesigns directory, d: drive, chiptrip.gdf file.	
Italic Type with Initial Capital Letters	Document titles are shown in italic type with initial capital letters. Example: <i>AN 75: High-Speed Board Design.</i>	
Italic type	Internal timing parameters and variables are shown in italic type. Examples: t_{PIA} , $n + 1$.	
	Variable names are enclosed in angle brackets (< >) and shown in italic type. Example: <i><file name=""></file></i> , <i><project name=""></project></i> . pof file.	
Initial Capital Letters	Keyboard keys and menu names are shown with initial capital letters. Examples: Delete key, the Options menu.	
"Subheading Title"	References to sections within a document and titles of on-line help topics are shown in quotation marks. Example: "Typographic Conventions."	

Visual Cue	Meaning	
Courier type	Signal and port names are shown in lowercase Courier type. Examples: data1, tdi, input. Active-low signals are denoted by suffix n, e.g., resetn.	
	Anything that must be typed exactly as it appears is shown in Courier type. For example: c:\qdesigns\tutorial\chiptrip.gdf. Also, sections of an actual file, such as a Report File, references to parts of files (e.g., the AHDL keyword SUBDESIGN), as well as logic function names (e.g., TRI) are shown in Courier.	
1., 2., 3., and a., b., c., etc.	Numbered steps are used in a list of items when the sequence of the items is important, such as the steps listed in a procedure.	
••	Bullets are used in a list of items when the sequence of the items is not important.	
\checkmark	The checkmark indicates a procedure that consists of one step only.	
	The hand points to information that requires special attention.	
CAUTION	A caution calls attention to a condition or possible situation that can damage or destroy the product or the user's work.	
WARNING	A warning calls attention to a condition or possible situation that can cause injury to the user.	
4	The angled arrow indicates you should press the Enter key.	
••••	The feet direct you to more information on a particular topic.	