# Table of Contents

## CHAPTER 1: ABOUT THIS MANUAL

1. **Manual Contents** .......................................................................................................................................................................................... 1
2. **Additional Resources** ........................................................................................................................................................................... 2
3. **Conventions** .................................................................................................................................................................................................. 3
   3.1 Typographical ................................................................................................................................................................................................ 3
   3.2 Content .................................................................................................................................................................................................. 3
   3.2.1 File names .......................................................................................................................................................................................... 5
   3.2.2 Physical orientation and Origin .......................................................................................................................................................... 5
   3.2.3 Part Pin Names .................................................................................................................................................................................. 5
   3.2.4 Schematic Clippings ....................................................................................................................................................................... 5
   3.2.5 Terminology .................................................................................................................................................................................................. 5

## CHAPTER 2: QUICK START GUIDE

1. **Provided Materials** .................................................................................................................................................................................... 6
2. **ESD Warning** .................................................................................................................................................................................................. 7
3. **Power-On Instructions** ............................................................................................................................................................................ 7
   3.1 Jumper setting ....................................................................................................................................................................................... 8
   3.2 Memory and heatsinks ........................................................................................................................................................................ 8
   4.3 **Install user clock oscillator** ............................................................................................................................................................... 9
   4.4 **Prepare configuration files** .......................................................................................................................................................... 9
   4.5 **Connect cables** ................................................................................................................................................................................ 10
   4.6 **View configuration feedback over RS232** ........................................................................................................................................ 11
   4.6.1 Interactive configuration ................................................................................................................................................................. 13
   4.6.2 Read temperature sensors ............................................................................................................................................................... 13
   3.7 **Check LED status lights** ................................................................................................................................................................. 14

## CHAPTER 3: CONTROLLER SOFTWARE

1. **USB Controller** ....................................................................................................................................................................................... 23
   1.1 **Menu Options** .................................................................................................................................................................................. 23
   1.1.1 File Menu ....................................................................................................................................................................................... 23
   1.1.2 Edit Menu ....................................................................................................................................................................................... 24
   1.1.3 FPGA Configuration Menu ........................................................................................................................................................... 24
   1.1.4 Settings/Info Menu ....................................................................................................................................................................... 25
   1.1.5 Configuration Register Map .......................................................................................................................................................... 26
2. **PCI AeTest Application** ............................................................................................................................................................................ 28
   2.1 **Functionality** ................................................................................................................................................................................... 28
   2.2 **Running AeTest** ............................................................................................................................................................................... 29
   2.3 **Compiling AeTest** ............................................................................................................................................................................. 29
   2.3.1 Compiling AeTest for DOS ............................................................................................................................................................ 29
   2.3.2 Compiling AeTest for Windows XP ................................................................................................................................................ 29
3. **Updating the Firmware** ........................................................................................................................................................................ 30
   3.1 **Updating the MCU (Flash) firmware** ........................................................................................................................................... 30
   3.2 **Updating the Cyclone (EEPROM) firmware** .................................................................................................................................... 31
List of Figures

Figure 1 DN7000K10PCI configuration controls ......................................................................................................................... 1
Figure 2 DN7000K10PCI configuration controls ......................................................................................................................... 2
Figure 3 FPGA Names ......................................................................................................................................................................... 8
Figure 4 RS232 Output ........................................................................................................................................................................... 13
Figure 5 Interactive Config Menu ......................................................................................................................................................... 13
Figure 6 Temperature Threshold Menu .............................................................................................................................................. 14
Figure 7 Configuration Status LEDs ..................................................................................................................................................... 14
Figure 8 DN7000K10PCI LEDs ............................................................................................................................................................. 17
Figure 9 USB Controller Window ......................................................................................................................................................... 19
Figure 10 USB Controller Log Output ............................................................................................................................................ 20
Figure 11 AETEST Main Menu ............................................................................................................................................................. 21
Figure 12 AETest Interconnect Menu ............................................................................................................................................... 22
Figure 13 Firmware update jumper ....................................................................................................................................................... 30
Figure 14 USB Controller Firmware Update Mode .......................................................................................................................... 31
Figure 15 Firmware Update Header ..................................................................................................................................................... 32
Figure 16 Programmer tool Window .................................................................................................................................................... 33
Figure 17 DN7000K10PCI Block Diagram .......................................................................................................................................... 34
Figure 18 Cyclone 2 IO Connections .................................................................................................................................................. 39
Figure 19 Cyclone 2 Configuration ...................................................................................................................................................... 40
Figure 20 DONE LEDs ........................................................................................................................................................................... 41
Figure 21 FPGA JTAG Header ............................................................................................................................................................ 42
Figure 22 TCK buffer ............................................................................................................................................................................. 42
Figure 23 FPGA A Configuration Bank ........................................................................................................................................... 43
Figure 24 Main.txt Commands ........................................................................................................................................................ 46
Figure 25 RS232 Buffer and Headers ............................................................................................................................................... 52
Figure 26 8442 Clock synthesizer ...................................................................................................................................................... 52
Figure 27 DN7000K10PCI clocking ................................................................................................................................................ 58
Figure 28 Main Bus read timing .......................................................................................................................................................... 100
Figure 29 Main Bus write timing ....................................................................................................................................................... 101
Chapter 1: About This Manual

This User Guide accompanies the DN7000K10PCI LOGIC Emulation Board. For specific information regarding the Stratix-II parts, please reference the datasheet.

Figure 1 DN7000K10PCI

1 Manual Contents

This manual contains the following chapters:

About This Manual

List of available documentation and resources available. Reader’s Guide to this manual

Quick Start Guide
Step-by-step instructions for powering on the DN7000K10PCI, loading and communicating with a simple provided FPGA design and using the board controls.

**Board Hardware**

Detailed description and operating instructions of each individual circuit on the DN7000K10PCI

**Controller Software**

A summary of the functionality of the provided software. Implementation details for the remote USB board control functions and instructions for developing your own USB host software.

**Reference Design**

Detailed description of the provided DN7000K10PCI reference design. Implementation details of the reference design interaction with DN7000K10PCI hardware features.

**FPGA Design Guide**

Information needed to use the DN7000K10PCI with third-party software, including Altera Quartus software, Certify, and Identify. Some commonly asked questions and problems specific to the DN7000K10PCI

**Ordering Information**

Contains a list of the available options and available optional equipment. Some suggested parts and equipment available from third party vendors.

## 2 Additional Resources

For additional information, go to [http://www.dinigroup.com](http://www.dinigroup.com). The following table lists some of the resources you can access from this website. You can also directly access these resources using the provided URLs.
### About This Manual

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description/URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserDN7000K10PCI User Guide</td>
<td>This is the main source of technical information. The manual should contain most of the answers to your questions</td>
</tr>
<tr>
<td>Dini Group Web Site</td>
<td>The web page will contain the latest manual, application notes, FAQ, articles, and any device errata and manual addenda. Please visit and bookmark: <a href="http://www.dinigroup.com">http://www.dinigroup.com</a></td>
</tr>
</tbody>
</table>
Most of your questions regarding usage and capabilities of the Stratix 2 devices will be answered here, including readback, boundary scan, configuration, and debugging |
| E-Mail                    | You may direct questions and feedback to the Dini Group using this e-mail address: support@dinigroup.com |
| Phone Support             | Call us at **858.454.3419** during the hours of 8:00am to 5:00pm Pacific Time. |
| FAQ                       | The download section of the web page contains a document called **DN7000K10PCI Frequently Asked Questions (FAQ)**. This document is periodically updated with information that may not be in the User's Manual. |

### 3 Conventions

This document uses the following conventions. An example illustrates each convention.

#### 3.1 Typographical

The following typographical conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning or Use</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Courier font</td>
<td>Messages, prompts, and program files that the system displays</td>
<td>speed grade: -100</td>
</tr>
</tbody>
</table>
| Courier bold    | Literal commands that you enter in a syntactical statement | ngdbuild  
design_name |
<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning or Use</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Garamond bold</strong></td>
<td>Commands that you select from a menu</td>
<td>File ➔ Open</td>
</tr>
<tr>
<td></td>
<td>Keyboard shortcuts</td>
<td>Ctrl+C</td>
</tr>
<tr>
<td><strong>Italic font</strong></td>
<td>Variables in a syntax statement for which you must supply values</td>
<td>ngdbuild design_name</td>
</tr>
<tr>
<td></td>
<td>References to other manuals</td>
<td>See the Development System Reference Guide for more information.</td>
</tr>
<tr>
<td></td>
<td>Emphasis in text</td>
<td>If a wire is drawn so that it overlaps the pin of a symbol, the two nets are <strong>not</strong> connected.</td>
</tr>
<tr>
<td><strong>Braces [ ]</strong></td>
<td>An optional entry or parameter. However, in bus specifications, such as bus[7:0], they are required.</td>
<td>ngdbuild [option_name] design_name</td>
</tr>
<tr>
<td><strong>Braces { }</strong></td>
<td>A list of items from which you must choose one or more</td>
<td>lowpwr = {on</td>
</tr>
<tr>
<td>**Vertical bar</td>
<td>**</td>
<td>Separates items in a list of choices</td>
</tr>
<tr>
<td><strong>Vertical ellipsis</strong></td>
<td>Repetitive material that has been omitted</td>
<td>IOB #1: Name = QOUT' IOB #2: Name = CLKIN'</td>
</tr>
<tr>
<td>-</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td><strong>Horizontal ellipsis</strong></td>
<td>Repetitive material that has been omitted</td>
<td>allow block block_name loc1 loc2 ... locn;</td>
</tr>
<tr>
<td>. . .</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Prefix “0x” or suffix “h”</strong></td>
<td>Indicates hexadecimal notation</td>
<td>Read from address 0x00110373, returned 4552494h</td>
</tr>
<tr>
<td><strong>Letter “#” or “_n”</strong></td>
<td>Signal is active low</td>
<td>INT# is active low fpga_inta_n is active low</td>
</tr>
</tbody>
</table>

DN600K10PCI User Guide  www.dinigroup.com  4
3.2 Content

3.2.1 File names
Paths to documents included on the User CD are prefixed with “D:\”. This refers to your CD drive’s root directory.

3.2.2 Physical orientation and Origin
By convention, the board is oriented as show on page 3, with the “top” of the board being the edge near SoDIMM sockets A, B and C. The “right” edge is near FPGA C, the “left” side is the side with the PCI bezel. “topside” refers to the side of the PWB with FPGAs soldered to it, “backside” is the side with the daughtercard connectors (solder side). The reference origin of the board is the centerline of the lower PCI bezel mounting hole.

3.2.3 Part Pin Names
Pin names are given in the form <X><Y>,<Z>; The <X> is one of: U for ICs, R for resistors, C for capacitors, P or J for connectors, FB or L for inductors, TP for test points, MH for mounting structures, BT for sockets, DS for diodes, F for fuses, HS for mechanicals, PSU for power supply modules, Q for discreet semiconductors, RN for resistor networks, X for oscillators, Y for crystals. <Y> is a number uniquely identifying each part from other parts of the same X class on the same PWB. <Z> is the pin or terminal number or name, as defined in the datasheet of the part. Datasheets for all standard and optional parts used on the DN7000K10PCI are included in the Document library on the provided User CD.

3.2.4 Schematic Clippings
Partial schematic drawings are included in this document to aid quick understanding of the features of the DN7000K10PCI. These clippings have been modified for clarity and brevity, and may be missing signals, parts, net names and connections. Unmodified Schematics are included in the User CD. Please refer to this document. Use the PDF search feature to search for nets and parts. A netlist is also included on the user CD. Use this document to create ucf files.

3.2.5 Terminology
Abbreviations and pronouns are used for some commonly used phrases.

**Bit Files** are the configuration bit stream files generated by the Altera Quartus software. These normally end in the .rbf file extension.

**MCU** is the Cypress FX2 Microcontroller, U29
Quick Start Guide

Congratulations on your purchase of the DN7000K10PCI LOGIC Emulation Board! You can begin by installing the software, or by powering on your DN7000K10PCI. If you wish to begin installation, please follow the installation instructions. The remainder of this chapter describes the contents of the box and how to start using the DN7000K10PCI LOGIC Emulation Board.

1 Provided Materials

Examine the contents of your DN7000K10PCI kit. It should contain:

- DN7000K10PCI board
- Two CompactFlash cards
- USB CompactFlash card reader
- RS232 IDC header cable to female DB9 and 6ft. DB9 serial cable
- USB cable
- CD ROM containing:
  - Stratix II Reference Design
  - User manual PDF
  - Board Schematic PDF
  - USB program (usbcontroller.exe, aetest_usb.exe)
  - PCI program (Aetestdj.exe)
  - Source code for USB program, PCI program and DN7000K10PCI firmware
2 ESD Warning

The DN7000K10PCI is sensitive to static electricity, so treat the PCB accordingly. The target markets for this product are engineers that are familiar with FPGAs and circuit boards. However, if needed, the following web page has an excellent tutorial on the “Fundamentals of ESD” for those of you who are new to ESD sensitive products:

http://www.esda.org/basics/part1.cfm

There are two large grounded metal rails on the DN7000K10PCI.

The DN7000K10PCI has been factory tested and pre-programmed to ensure correct operation. You do not need to alter any jumpers or program anything to see the board work. A reference design is included on the provided CD and CompactFlash card.

The 400-pin connectors are not 5V tolerant. Take care when handling the board to avoid touching the daughtercard connectors.

3 Power-On Instructions

The image below represents your DN7000K10PCI. You will need to know the location of the following parts referenced in this chapter.

To begin working with the DN7000K10PCI, follow the steps below:
3.1 Jumper setting
There are two jumper blocks on the DN7000K10PCI.

JP3 should be set according to whether the board is being operated in stand-alone mode, or is plugged into a PCI slot. Connect JP3.1-JP3.3 if the board is operated in stand-alone mode. Connect JP3.2-JP3.4 if the board is operated in a PCI slot. This is the default configuration.

JP1 sets the power requirement communicated to a host over PCI. This jumper should be left in its default state, JP1.1-J1.2, to indicate that the DN7000K10PCI uses the maximum PCI power, 25W.

3.2 Memory and heatsinks
There should be an active heatsink installed on each FPGA on the DN7000K10PCI and a fan over the power supply units. Stratix 2 FPGAs are capable of dissipating 15W or more, so you should always run them with heatsinks installed.

The DN7000K10PCI comes packaged without memory installed. If you want the Dini Group reference design to test your memory modules, you can install them now in the 1.8V DDR2 DIMM sockets.

The FPGA naming convention is U8 is “FPGA A”, U15 is “FPGA B” and U23 is “FPGA C”

The socket DIMM A is connected to FPGA A, DIMM B is accessible from FPGA B and DIMM C is accessible from FPGA C. The sockets can accept any capacity DDR2 Sodimm module. Note that DDR1 modules will not work in these slots since they are supplied with 1.8V power and DDR1 requires 2.5V power and a completely different pin-out.
3.3 Install user clock oscillator

X4, an IC socket can accept a 3.3V, single ended oscillator. This oscillator output is distributed synchronously to all FPGAs. If your user design requires a certain frequency, install an oscillator here. The frequency of this global clock network will be read and reported by the Dini Group reference design.

Alternatively, this clock tree can be sourced from the SMA J7.

3.4 Prepare configuration files

The DN7000K10PCI reads FPGA configuration data from a CompactFlash card. To program the FPGAs on the DN7000K10PCI, FPGA design files (with a .rbf file extension) put on the root directory of the CompactFlash card file using the provided USB card reader.

The DN7000K10PCI ships with a 32 MB CompactFlash card preloaded with the Dini Group reference design.

1. Insert the provided CompactFlash card labeled “Reference Design” into your USB card reader. Make sure the card contains the files:

   A_180.rbf
   B_180.rbf
   C_180.rbf
   main.txt

The files A-C_180.rbf are files created by the Altera Programming tool, part of the Quartus 2 package. The file main.txt contains instructions for the DN7000K10PCI configuration circuitry, including which FPGAs to configure, and to which frequency the global clock networks should be automatically adjusted. File names on the CompactFlash card must be 8 characters or shorter.

The file main.txt has the following contents
2. Insert the CompactFlash card labeled “Reference Design” into the DN7000K10PCI’s CompactFlash slot.

3.5 Connect cables
The configuration circuitry can accept user input to control FPGA configuration or provide feedback during the configuration process. The configuration circuitry IO can also be used to transfer data to and from the user design.

1. Use the provided ribbon cable to connect the MCU RS232 port (P2) to a computer serial port to view feedback from the configuration circuitry during FPGA configuration. Run a serial terminal program on your PC (On Windows you can use HyperTerminal Start->Programs->Accessories->Communications->HyperTerminal) and make sure the computer serial port is configured with the following options:
   
   • Bits per second: 19200
   • Data bits: 8
   • Parity: None
   • Stop Bits: 1
   • Flow control: None
   • Terminal Emulation: VT100

2. Use the provided USB cable to connect the DN7000K10PCI to a Windows computer (Windows XP is recommended).

3. Plug an ATX power supply into J3. If you want to use the DN7000K10PCI in a PCI slot, instead of connecting an ATX connector, plug the DN7000K10PCI into a +3.3V or +5.0V PCI slot. Do not connect J3 to an
external power supply if using the DN7000K10PCI in a PCI slot. Turn on
the ATX power supply or computer.

When the DN7000K10PCI powers on, it automatically loads Altera FPGA design files
(ending with a .rbf extension), found on the CompactFlash card in the CompactFlash
slot into the FPGAs. (Assuming the main.txt file on the card is found and contains
correct configuration instructions)

3.6 View configuration feedback over RS232
As the DN7000K10PCI powers on, your RS232 terminal (connected to P2) will
display useful information about the Configuration process.

Watch the configuration status output

DINI GROUP FLP EEPROM VERSION 06
USB CONNECT NOT SET
Rebooting from flash...please wait
=.=. DN7000K10PCI MCU FLASH BOOT .=.=

-- FPGAS STUFFED --
A B C
RESETTING CF.....DONE

Calculating M/N values...
00
A 02 14

Calculating M/N values..
01
B 01 19

Calculating M/N values...
02
D 03 1C

--OPTIONS--
Message level set to default: 2
Sanity check is set to default: ON

Setting ACLK...
M VAL = 0x14 N VAL = 0x02
Setting BCLK...
M VAL = 0x19 N VAL = 0x01
Setting CCLK...
M VAL = 0x1C N VAL = 0x03

--OPTIONS--
Message level set to default: 2

****CONFIGURING FPGA: A*****
-- Performing Sanity Check on Bit File --
-- BIT FILE ATTRIBUTES --
FILE NAME: A_180.RBF

“Flash Boot” means the board is not in firmware update mode.

The global clocks (ACLK, BCLK, CCLK) are frequency-configurable. The
M binary sequence represents the multiplication applied to the
installed crystal. The N represents the division applied, U6, U14, U20,
U31 and the ICS8442AY datasheet.

The MCU is setting the clocks to their default values ACLK 200Mhz,
BCLK 108.8Mhz, CCLK 128Mhz, R1CLK (not available on
DN7000K10PCI), R2CLK (**DEFAULT**)”

The MCU detects which FPGAs are present

The MCU detects if a CompactFlash card is present

The MCU tries to access the CompactFlash card. If the MCU is not
successful in reading the files on the CompactFlash card, be sure you
have not formatted the card in Windows. Windows uses a non-
standard format for media cards and will make the card unreadable.
You can download a format utility from dinigroup.com to repair your
incorrectly-formatted SM card.

The MCU reads the contents of the file MAIN.TXT and executes
each instruction line.
Here the MCU is setting the clocks according to instructions in MAIN.TXT

The MCU is configuring FPGA A according to instructions in MAIN.TXT

The sanity check option reads the design (".rbf") file headers and verifies that the design is compiled for the same type of FPGA that the MCU detects on your DN7000K10PCI. If the design and FPGA do not match, the MCU will reject the file and flash the Error LED. You may need to disable the sanity check option (See Chapter X, section X) if you want to encrypt or compress your configuration files.

The MCU is configuring FPGA B according to instructions in MAIN.TXT

The MCU is setting the temperature threshold to cause a board reset.

Here is the MCU main menu

1.) Configure FPGAs using "MAIN TXT"
2.) Interactive configuration menu
3.) Check configuration status
4.) Change MAIN configuration file
5.) List files on Smart Media
6.) Display Smart Media text file
7.) Change RS232 PPC Ports

f.) Turn FANS off
g.) Display FPGA Temperatures

FPGA Temperature Alarm Threshold: 80 degrees C

-- TEMPERATURE SENSORS --
A YES
B YES
C YES

-- PERFORMING SANITY CHECK ON BIT FILE --
FILE NAME: B_180.RBF
FILE SIZE: 003A943B bytes
PART: 467010113152001
DATA: 2005/07/19
TIME: 17:05:01
Sanity check passed

-- PERFORMING SANITY CHECK ON BIT FILE --
FILE NAME: A_180.RBF
FILE SIZE: 003A943B bytes
PART: 467010113152001
DATA: 2005/07/25
TIME: 17:09:38
Sanity check passed

**CONFIGURING FPGA A******
**CONFIGURING FPGA B******

Here the MCU is setting the clocks according to instructions in MAIN.TXT

The MCU is configuring FPGA A according to instructions in MAIN.TXT

The MCU is configuring FPGA B according to instructions in MAIN.TXT

The MCU is setting the temperature threshold to cause a board reset.
QUICK START GUIDE

h.) Set FPGA Temperature Alarm Threshold
i.) Read Temperature Sensor Reg
j.) Write Temperature Sensor Reg

Options 8, 9, and A are only available when the DN7000K10PCI reference design is loaded. For more information on how the MCU communicates with the reference design, see Chapter X, The Reference Design.

You should see the DN7000K10PCI MCU main menu. If the reference design is loaded in the Stratix 2 FPGAs, then you should see the above on your terminal. Try pressing 3 to see if the configuration circuit was successful in programming the FPGAs.

ENTER SELECTION: 3

******************** CONFIGURATION STATUS *********************
FPGA B NOT configured

The easiest way to verify your FPGAs are configured is to look at DS18, DS14, DS16 located above each FPGA. When the green LEDs are lit, the FPGA under it is successfully configured.

3.6.1 Interactive configuration
If you want to put multiple designs on a single CompactFlash card, you can use the interactive configuration menu to select which .rbf file to use on each FPGAs. Select menu option 2.

ENTER SELECTION: 2
-=-= INTERACTIVE CONFIGURATION MENU -=-=

1) Select bit files to configure FPGA(s)
2) Set verbose level (current level = /)
3) Enable sanity check for bit files
M) Main Menu

Enter Selection:

3.6.2 Read temperature sensors
The DN7000K10PCI is equipped with temperature sensors to measure and monitor the temperature on the die of the Stratix FPGAs. According to the Stratix 2 datasheet, the maximum recommended operating temperature of the die is 85°C degrees. If the microcontroller measures a temperature above 85°C degrees, it will reset the DN7000K10PCI.
If you think your DN7000K10PCI is resetting due to temperature overload, you can use the temperature monitor menu to measure the current junction temperature of each FPGA.

```
ENTER SELECTION: g

-- FPGA TEMPERATURES (Degrees Celsius [+/- 4]) --
B 29
-- Set FPGA Temperature Alarm Threshold --
(degrees C, decimal values, range [1-127])
Old Threshold: 80
New Threshold: 85
Threshold Updated: 85 Degrees C
```

The Stratix 2 FPGA can operate as hot as 85 degrees Celsius before damaging the part, although timing specifications are not guaranteed. The MCU allows you to change the reset threshold, although we recommend improving your heat dissipation to maintain a low junction temperature.

### 3.7 Check LED status lights

The DN7000K10PCI has many status LEDs to help the user confirm the status of the configuration process.
1. Check the power voltage indication LEDs to confirm that all voltage rails of the DN7000K10PCI are present. From the top, the LEDs indicate the presence of –12V, +12V, +5V, +3.3V, +2.5V, and “ATX POWER OK” Green lit LED's on the voltage present LEDs indicate the rails are present (greater than 1.7V). A green lit “ATX power OK” indicates that the voltage monitors inside the ATX power supply are within acceptable operating ranges (5V, 3.3V and 12V). If this LED is not lit green, the DN7000K10PCI might not function properly.

2. Check the Configuration status LEDs. Under error conditions, all four red LEDs will blink.

3. Check the Cyclone FPGA status LED, DS10. This unreasonably bright LED indicates that the Cyclone 2 FPGA has been configured. If this LED is not lit soon after power on, then there may be a problem with the firmware on the DN7000K10PCI. If this LED remains off or blinking, it may indicate a power failure.

4. Check the FPGA A status LED, DS16 to the upper left of FPGA A. This blue LED is lit when FPGA A is configured and operational. This light should be on if you loaded the reference design from the CompactFlash card.

5. Check the FPGA B status LED, DS20 directly above FPGA B. This light should be lit blue if your DN7000K10PCI was installed with the FPGA B option, and the reference design is loaded.

6. Check the FPGA C status LED, DS26 to the upper left of FPGA C. This blue LED will light if you have the FPGA C option and the FPGA is configured.

7. Check the FPGA A User LEDs to the left of FPGA A. These should blink at a rate proportional to the ACLK frequency.

8. Check the FPGA C User LEDs to the right of FPGA C.

9. If you suspect one or more FPGAs did not configure properly, check the configuration circuitry’s status lights. If there has been an error, three of the LEDs will blink. If there has been no error, there should be two LEDs ON and two OFF. If there was an error, the best way to determine the cause of the error is to connect a serial terminal to the RS232 port (P2) and try to configure again. Configuration feedback will be presented over this port.

You should also notice the Fans mounted above the 3 Stratix 2 FPGAs and the Fan mounted above the power supplies spinning.
<table>
<thead>
<tr>
<th>Assembly Number</th>
<th>Signal</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS1 red</td>
<td>MCU_LED</td>
<td>This 4-LED block blinks an error code when configuration fails.</td>
</tr>
<tr>
<td>DS2 red</td>
<td>CFPGA_LED</td>
<td>This 4-LED block flickers with activity over USB, PCI, Main Bus or CompactFlash</td>
</tr>
<tr>
<td>DS3 red</td>
<td>PWR_OK</td>
<td>The ATX power supply is working properly</td>
</tr>
<tr>
<td>DS4 red</td>
<td>+2.5V</td>
<td>The +2.5 power rail is present.</td>
</tr>
<tr>
<td>DS5 red</td>
<td>+3.3V</td>
<td>The +3.3V power rail is present</td>
</tr>
<tr>
<td>DS6 red</td>
<td>+5.0V</td>
<td>The +5.0V power rail is present</td>
</tr>
<tr>
<td>DS7 red</td>
<td>+12V</td>
<td>The +12V power rail is present</td>
</tr>
<tr>
<td>DS8 red</td>
<td>-12V</td>
<td>The –12V rail is present</td>
</tr>
<tr>
<td>DS9, DS11, DS13 green</td>
<td>FPGA_LED_A</td>
<td>FPGA A user LEDs</td>
</tr>
<tr>
<td>DS10 really bright blue</td>
<td>CFPGA DONE</td>
<td>The Cyclone configuration FPGA is configured</td>
</tr>
<tr>
<td>DS12 red</td>
<td>+1.2V_A fail</td>
<td>The +1.2V power rail dedicated to FPGA A has failed</td>
</tr>
<tr>
<td>DS14 green</td>
<td>QL_LED_IN</td>
<td>There is data input activity over PCI</td>
</tr>
<tr>
<td>DS15 green</td>
<td>QL_LED_OUT</td>
<td>There is data output activity over PCI</td>
</tr>
<tr>
<td>DS16 blue</td>
<td>FPGA_A_DONE</td>
<td>FPGA A is configured</td>
</tr>
</tbody>
</table>
4 Using the Reference Design with the Provided Software

To communicate with the reference design (or user design) on the emulation board, the DN7000K10PCI provides three options out of the box.

- USB
- PCI
- RS232

The USB and PCI interfaces allow configuration of the FPGAs and bulk data transfer to and from the User design. The RS232 interface allows low-speed data transfers to and from the User design, and control and monitoring of the configuration process.

This section will get you started and show you how to operate the provided software. For detailed information about the reference design and implementation details, see Chapter 5, The Reference Design.

4.1 Operating the USB controller program

Use the provided USB monitoring software to verify that the design is loaded into the FPGAs.
1. Insert the CDROM that came with your DN7000K10PCI into the CDROM drive of your computer.

2. Connect the USB cable to your DN7000K10PCI and a Windows XP PC. (Before or after the DN7000K10PCI has powered on). This can be done even if the DN7000K10PCI is plugged into the PCI slot of this or another computer.

3. When you connect the USB cable to your PC for the first time, Windows detects the DN7000K10PCI and asks for a driver. The board should identify itself as a “DiNi Prod FLASH BOOT”. When the new device detected window appears, select the option "install from a list" -> select "search for the best driver in these locations". Select "include the location in the search" and browse to the product CD in “Source Code\AETEST_USB\driver\win_wdm" ->select "finish"

4. After Windows installs the driver, you will be able to see the following device in the USB section of Windows device manager: “DiniGroup DN7000K10PCI FLASH boot”.

5. Run the USB controller application found on the product CD in “Source Code\USBController\USBController.exe”.

If the board is not plugged in to USB, or the driver is not installed, then this message will appear. You may need to wait a few seconds. The DN7000K10PCI takes about 4 seconds to start up after power-on. During this time, the USB controller program will not detect the board.
6. This window will appear showing the current state of the DN7000K10PCI. Next to each FPGA a green light will appear if that FPGA is configured successfully. The above window shows the USB Controller connected to a DN7000K10PCI with a single FPGA in the B position. If you have the reference design loaded and a DDR2 SODIMM installed, you can use the USB Controller to run tests of the SODIMM. From the FPGA Memory menu, select Test DDR.

7. Clear the FPGAs of their configurations. Right-click on an FPGA and select from the popup menu, “Clear FPGA”. The green light above the FPGA on the GUI and on the board should stop shinning green.

8. Configure an FPGA using the USB Controller program. Right-click on an FPGA and select Configure FPGA via USB from the popup menu. The program will open a dialog box for you to select the configuration file to use for configuration. Browse to the provided user’s CD “USERCD: \BitFiles\7000K10PCI\MainTest\EPS180\A_180.rbf” If you are configuring an EPS130 or EPS90 devices you should select a rbf file from the EPS130 or EPS90 directories instead. If you are configuring FPGA B or FPGA C, you should select B_180.rbf or C_180.rbf instead.
QUICK START GUIDE

FPGA B cleared successfully.
FPGA A cleared successfully.
Doing a sanity check...Sanity Check passed. Configuring FPGA B via USB...please wait.
File
D:\dn_BitFiles\DN7000K10PCI\MainTest\ES180\B_180.rbf transferred.
Configured FPGA B via USB

9. The message box below the DN7000K10PCI graphic should display some information about the configuration process

10. Reset Fpgas

The USB Controller program also allows you to easily configure and transfer data to and from the user design on the emulation board. More information is provided in Chapter X, “The USB program”

4.2 Communicating with the Reference design

The reference design uses the MainBus interface. The MainBus interface uses the MB[0-37] signals that connect together among all three FPGAs. The user design is not required to use this interface. For details on the MainBus interface signaling, see Chapter 5, The Reference Design

When the DN7000K10PCI powers on, this interface is disabled by default. Click on the Enable USB communication button to restore the functionality of the reference design. (Memory tests, etc)

4.3 Communicating to the User Design over the Serial Port

Each FPGA on the DN7000K10PCI is connected to a RS232 header. This header is pinned out so that you can use the provided cable to connect to your computer’s serial port.

FPGA A – P5
FPGA B – P7
FPGA C – P8

The reference design implements a digital loopback on these connectors, using the same port settings as the MCU RS232 terminal (19200 bs). RS232 uses 12V signaling, so these headers are connected to the FPGA IOs through RS232 buffers.
4.4 Using AETEST to run hardware tests

AETest is the program that you can use to verify the hardware on the DN7000K10PCI, as well as to demonstrate the reference design function. The following instructions assume you have a PC running the Windows XP operating system. The user CD includes a Windows version of the AETest program. If you plan to use the DN7000K10PCI in stand-alone mode, connect the DN7000K10PCI to your WindowsXP computer and use aetest_usb in D:\aetest_usb\aeusb_wdm.exe. If the computer asks for a driver, click “Have Disk” and browse to D:\AETest_sb\driver\win_wdm\dndevusb.inf If you are going to use the DN7000K10PCI in a PCI slot, turn off the computer, insert the DN7000K10PCI into an unused PCI slot, and turn the computer on. If the operating system asks for a driver, click “Have Disk” and browse to D:\AETest\wdmdrv\drv\dndev.inf Then run the PCI version of the AETest application at D:\AETest\aetest\aetest_wdm.exe

4.4.1 Use AETest

The Aetest application should display its main menu.

![AETEST Main Menu](image)

Figure 11 AETEST Main Menu

Run one of the tests. Choose option A. Remember, the FPGA you test has to be loaded with the reference design, or the test will fail.
For more information on the AETEST program, see Chapter 3.

4.5 Moving On
Congratulations! You have just programmed the DN7000K10PCI and learned all of the features that you must know to start your emulation project. You can use the constraints file (.qsf) provided on the User CD with the reference design to start your own project. A netlist of the board is included in the User CD to help you interface to the DN7000K10PCI. All of the source code for the reference design in Verilog is included on the provided CD.
Controller Software

1 USB Controller

USBController application is used to communicate with the DN7000K10PCI.

All USBController source code is included on the CD-ROM shipped with the DN7000K10PCI. The USBController can be installed on Windows 98/ME/2000/XP. There is a command line version called AETEST_USB that can be installed on Linux and Solaris.

The USBController Application contains the following functionality:
- Verify Configuration Status
- Configure FPGA(s) over USB
- Configure FPGAs via CompactFlash card
- Clear FPGA(s)
- Reset FPGA(s)
- Set Global clocks frequency
- Update MCU FLASH firmware

The following function interface with the Dini Group reference design.
- Read/Write to FPGA(s) –
- Test DDRs/FLASH/Registers/FPGA Interconnect

1.1 Menu Options
1.1.1 File Menu
The File Menu has the following 2 options:

a. Open – opens a file with the selected text editor (notepad by default). To change the text editor see Settings/Info Menu section

b. Exit – Closes the USBController application
1.1.2 Edit Menu
The Edit Menu performs the basic edit commands on the command log in the bottom half of the USBController window.

1.1.3 FPGA Configuration Menu
The FPGA Configuration Menu has the following options:

1. Refresh – Redraws the board graphic. (If configuration status has changed)

2. Configure via USB (individually) – After selecting this option a window will pop and ask which FPGA you want to configure and then what rbf file you want to configure the selected FPGA with. The status of the FPGA configuration will detailed in the log window and the DN7000K10PCI will be updated after the rbf file has been transferred.

3. Configure via USB using file – This option allows the user to configure more than one FPGA over USB at a time. To use this option you must create a setup file that contains information on which FPGA(s) should be configured and what rbf files should be used for each FPGA. The file should be in the following format, the first character of each line represents which FPGA you want configured (a-f or A-F), this letter should be followed by a colon and then the path to the rbf file to use for this FPGA. The path to the rbf file is relative to the directory where this setup file is, or you can use the full path. Below is an example of an accepted setup file:

   A: A_180.rbf
   B: B_180.rbf
   C: C_180.rbf

   The clock setting commands available in the main.txt file can also be used in the setup file.

4. Configure via Media Card – This option configures the FPGAs using the CompactFlash card in the DN7000K10PCI’s CF slot. Please section Creating Configuration File “main.txt” for information on what files should be on the CompactFlash card to use this option.

5. Clear All FPGAs – This option will deconfigure all FPGAs.

6. Reset – This options sends an active low reset to all FPGAs on the signal called RESET_FPGASn which is connected to the following I/O pins:

   FPGA A: AN21
   FPGA B: AN21
   FPGA C: AN21
## 1.1.4 Settings/Info Menu

The Settings/Info Menu has the following options:

1. Set Global clock frequencies

The clocks on the DN7000K10PCI are automatically adjusted to the user’s desired frequency by reading the setup file on the CompactFlash card. If you wish to change the frequency after power-on, or do not want to use a CompactFlash card, you can set the frequency in the USB program.

- **ACLK)** ACLK is generated from a 25MHz crystal. Available frequencies are:

<table>
<thead>
<tr>
<th>31.25</th>
<th>34.375</th>
<th>37.5</th>
<th>40.625</th>
<th>43.75</th>
<th>46.875</th>
<th>50</th>
<th>53.125</th>
<th>56.25</th>
</tr>
</thead>
<tbody>
<tr>
<td>59.375</td>
<td>62.5</td>
<td>65.625</td>
<td>68.75</td>
<td>71.875</td>
<td>75</td>
<td>78.125</td>
<td>81.25</td>
<td></td>
</tr>
<tr>
<td>84.375</td>
<td>87.5</td>
<td>93.75</td>
<td>100</td>
<td>106.25</td>
<td>112.5</td>
<td>118.75</td>
<td>125</td>
<td></td>
</tr>
<tr>
<td>131.25</td>
<td>137.5</td>
<td>143.75</td>
<td>150</td>
<td>156.25</td>
<td>162.5</td>
<td>168.75</td>
<td>175</td>
<td>187.5</td>
</tr>
<tr>
<td>200</td>
<td>212.5</td>
<td>225</td>
<td>237.5</td>
<td>250</td>
<td>262.5</td>
<td>275</td>
<td>287.5</td>
<td>300</td>
</tr>
<tr>
<td>312.5</td>
<td>325</td>
<td>337.5</td>
<td>350</td>
<td>375</td>
<td>400</td>
<td>425</td>
<td>450</td>
<td>475</td>
</tr>
<tr>
<td>500</td>
<td>525</td>
<td>550</td>
<td>575</td>
<td>600</td>
<td>625</td>
<td>650</td>
<td>675</td>
<td>700</td>
</tr>
</tbody>
</table>

- **BCLK)** BCLK is generated from a 14.318 Mhz crystal. Supported frequencies are:

<table>
<thead>
<tr>
<th>32.22</th>
<th>34.01</th>
<th>35.80</th>
<th>37.58</th>
<th>39.37</th>
<th>41.16</th>
<th>42.95</th>
<th>44.74</th>
<th>46.53</th>
</tr>
</thead>
<tbody>
<tr>
<td>48.32</td>
<td>50.11</td>
<td>51.90</td>
<td>53.69</td>
<td>55.48</td>
<td>57.27</td>
<td>59.06</td>
<td>60.85</td>
<td>62.64</td>
</tr>
<tr>
<td>64.43</td>
<td>66.22</td>
<td>68.01</td>
<td>69.80</td>
<td>71.59</td>
<td>73.38</td>
<td>75.17</td>
<td>76.96</td>
<td>78.75</td>
</tr>
<tr>
<td>80.54</td>
<td>82.33</td>
<td>84.12</td>
<td>85.91</td>
<td>89.49</td>
<td>93.07</td>
<td>96.65</td>
<td>100.2</td>
<td>103.8</td>
</tr>
<tr>
<td>107.4</td>
<td>111.0</td>
<td>114.5</td>
<td>118.1</td>
<td>121.7</td>
<td>125.3</td>
<td>128.9</td>
<td>132.4</td>
<td>136.0</td>
</tr>
<tr>
<td>139.6</td>
<td>143.2</td>
<td>146.8</td>
<td>150.3</td>
<td>153.9</td>
<td>157.5</td>
<td>161.1</td>
<td>164.7</td>
<td>168.2</td>
</tr>
<tr>
<td>171.8</td>
<td>179.0</td>
<td>186.1</td>
<td>193.3</td>
<td>200.5</td>
<td>207.6</td>
<td>214.8</td>
<td>221.9</td>
<td>229.1</td>
</tr>
<tr>
<td>236.2</td>
<td>243.4</td>
<td>250.6</td>
<td>257.7</td>
<td>264.9</td>
<td>272.0</td>
<td>279.2</td>
<td>286.4</td>
<td>293.5</td>
</tr>
<tr>
<td>300.7</td>
<td>307.8</td>
<td>315.0</td>
<td>322.2</td>
<td>329.3</td>
<td>336.5</td>
<td>343.6</td>
<td>350.8</td>
<td>372.3</td>
</tr>
<tr>
<td>386.6</td>
<td>400.9</td>
<td>415.2</td>
<td>429.5</td>
<td>443.9</td>
<td>458.2</td>
<td>472.5</td>
<td>486.8</td>
<td>501.1</td>
</tr>
<tr>
<td>515.4</td>
<td>529.8</td>
<td>544.1</td>
<td>558.4</td>
<td>572.7</td>
<td>587.0</td>
<td>601.4</td>
<td>615.7</td>
<td>630.0</td>
</tr>
<tr>
<td>644.3</td>
<td>658.6</td>
<td>672.9</td>
<td>687.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **CCLK)** CCLK is generated from a 16.0 Fundamental crystal. Supported frequencies:

<table>
<thead>
<tr>
<th>32</th>
<th>34</th>
<th>36</th>
<th>38</th>
<th>40</th>
<th>42</th>
<th>44</th>
<th>46</th>
<th>48</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>52</td>
<td>54</td>
<td>56</td>
<td>58</td>
<td>60</td>
<td>62</td>
<td>64</td>
<td>66</td>
</tr>
<tr>
<td>68</td>
<td>70</td>
<td>72</td>
<td>74</td>
<td>76</td>
<td>78</td>
<td>80</td>
<td>82</td>
<td>84</td>
</tr>
<tr>
<td>86</td>
<td>88</td>
<td>92</td>
<td>96</td>
<td>100</td>
<td>104</td>
<td>108</td>
<td>112</td>
<td>116</td>
</tr>
<tr>
<td>120</td>
<td>124</td>
<td>128</td>
<td>132</td>
<td>136</td>
<td>140</td>
<td>144</td>
<td>148</td>
<td>152</td>
</tr>
<tr>
<td>156</td>
<td>160</td>
<td>164</td>
<td>168</td>
<td>172</td>
<td>176</td>
<td>184</td>
<td>192</td>
<td>200</td>
</tr>
</tbody>
</table>
CONTROLLER SOFTWARE

208 216 224 232 240 248 256 264 272
280 288 296 304 312 320 328 336 336
344 352 368 384 400 416 432 448 464
480 496 512 528 544 560 576 592 608
624 640 656 672 688

(2) Change Text Editor – This option allows the user to select a text editor to use (the default editor is notepad).

(3) FPGA Stuffing Information – This option will display the type of FPGAs that are stuffed on the DN7000K10PCI.

(4) MCU Firmware Version – This option will display the MCU Firmware version in the log window.

(5) BOARD/CYCLONE Version – This option will display the Board Version along with the Cyclone (Config Fpga) Version.

1.1.5 Configuration Register Map
The DN7000K10PCI firmware is updated constantly to add compatibility for new products and add features. The information in this section may change after this manual is printed. The memory space of the MCU is 16 bits wide.

This table describes registers within the Configuration FPGA that are accessible from the memory space of the MCU.

<table>
<thead>
<tr>
<th>REGISTER</th>
<th>ADDRESS</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>BITS_1</td>
<td>DF07</td>
<td>BIT7: mcu_fpga_config_rd</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT6: mcu_fpga_config_done</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT5: FPGA_ack</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT4: r_FPGA_PROGn,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT3-2: mcu_mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT1: mcu_sm_rdy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT0: mcu_reading</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT4: FPGA_DONE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT3 CPLD_idle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT2: SM_RDYBUSYn</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT1: FPGA_INITn</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIT0: mcu_encrypt</td>
</tr>
<tr>
<td>BITS_2</td>
<td>DF08</td>
<td>BIT0: mcu_encrypt</td>
</tr>
<tr>
<td>SM_SIGNALS</td>
<td>DF09</td>
<td>address register for upper FLASH/SRAM bits</td>
</tr>
<tr>
<td>MCU_XADDR</td>
<td>DF0A</td>
<td>address register for upper FLASH/SRAM bits</td>
</tr>
<tr>
<td>MCU_CNTL</td>
<td>DF0B</td>
<td>FPGA_select[3:0] = bits 3:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Selects which FPGA PPC is connected to PPC Port 1 and 2:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FPGAs are represented by 4 bits:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A = 000, B = 0x1, ..., l = 0x8</td>
</tr>
<tr>
<td>FPGA_SELECT</td>
<td>DF0C</td>
<td>Port 1 is lower 4 bits,</td>
</tr>
<tr>
<td>PPC_RS232_12SELECT</td>
<td>DF0D</td>
<td></td>
</tr>
</tbody>
</table>
### CONTROLLER SOFTWARE

<table>
<thead>
<tr>
<th>Register Name</th>
<th>Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>PPC_RS232_34SELECT</td>
<td>DF0E</td>
</tr>
<tr>
<td>FPGA_CNTRL</td>
<td>DF0F</td>
</tr>
<tr>
<td>FPGA_BE</td>
<td>DF10</td>
</tr>
<tr>
<td>FPGA_RD_DATA</td>
<td>DF11</td>
</tr>
<tr>
<td>FPGA_WR_DATA</td>
<td>DF12</td>
</tr>
<tr>
<td>FPGA_ADDR</td>
<td>DF13</td>
</tr>
<tr>
<td>FPGA_ERROR</td>
<td>DF14</td>
</tr>
<tr>
<td>GPIF_DATA</td>
<td>DF20</td>
</tr>
<tr>
<td>GPIF_ERROR</td>
<td>DF21</td>
</tr>
<tr>
<td>HOLD_DONES</td>
<td>DF22</td>
</tr>
<tr>
<td>STATES</td>
<td>DF2</td>
</tr>
<tr>
<td>FPGA_FREQ_H</td>
<td>DF24</td>
</tr>
<tr>
<td>FPGA_FREQ_SEL</td>
<td>DF25</td>
</tr>
<tr>
<td>FPGA_FREQ_L</td>
<td>DF26</td>
</tr>
<tr>
<td>MCU_STUFFING1</td>
<td>DF27</td>
</tr>
<tr>
<td>MCU_STUFFING2</td>
<td>DF28</td>
</tr>
<tr>
<td>ACLK_N_VAL</td>
<td>DF29</td>
</tr>
<tr>
<td>ACLK_M_VAL</td>
<td>DF30</td>
</tr>
<tr>
<td>BCLK_N_VAL</td>
<td>DF31</td>
</tr>
<tr>
<td>BCLK_M_VAL</td>
<td>DF32</td>
</tr>
<tr>
<td>DCLK_N_VAL</td>
<td>DF33</td>
</tr>
<tr>
<td>DCLK_M_VAL</td>
<td>DF34</td>
</tr>
<tr>
<td>BOARD_VERSION_DUP</td>
<td>DF46</td>
</tr>
<tr>
<td>CF_REG_OFFSET</td>
<td>DF6X</td>
</tr>
<tr>
<td>FPGA_COMMUNICATION</td>
<td>DF39</td>
</tr>
<tr>
<td>PENDING_CLKS</td>
<td>DF40</td>
</tr>
<tr>
<td>CHECKSUM</td>
<td>DF45</td>
</tr>
<tr>
<td>TEMP_SENSOR_A</td>
<td>DF50</td>
</tr>
<tr>
<td>TEMP_SENSOR_B</td>
<td>DF51</td>
</tr>
<tr>
<td>TEMP_SENSOR_C</td>
<td>DF52</td>
</tr>
<tr>
<td>SERIAL_NUM_ADDR</td>
<td>DFFA</td>
</tr>
<tr>
<td>SPARTAN_MKS_VERSIO_ADDR</td>
<td>DFFB</td>
</tr>
<tr>
<td>SPARTAN_VERSION_ADDR_DR</td>
<td>DFFD</td>
</tr>
<tr>
<td>BOARD_VERSION_NEW</td>
<td>DFFE</td>
</tr>
<tr>
<td>BOARD_VERSION</td>
<td>DFFF</td>
</tr>
</tbody>
</table>

Port 2 is upper 4 bits
Select which FPGA PPC(s) is connected to PPC
Port 3 and 4
bits[1:0] = 01 (write address), 10 (data write),
11 (data read)

bits[4:2] = r_fpga_state
bit5 = r_fpga_auto_inc
(when high, auto increments fpga_addr after access)
bit6 = r_fpga_rd_done
select byte in addr, read, and data bytes


Sets the divider value of GCLK0
Sets the multiplier of GCLK0
Sets the divider value of GCLK1
Sets the multiplier of GCLK1
Sets the divider value of GCLK2
Sets the multiplier of GCLK2
Sets the divider value of GCLK0
Sets the multiplier of GCLK0
Sets the divider value of GCLK1
Sets the multiplier of GCLK1
Sets the divider value of GCLK2
Sets the multiplier of GCLK2

Various CompactFlash card controls. Main.txt is read through these registers.
Disables Main Bus interface
Causes reprogramming of onboard synthesizers
A checksum of USB configuration data
Temperature of FPGA A
Temperature of FPGA A
2 PCI AETEST Application

AETEST utility program can test and verify the functionality of the DN7000K10PCI Logic Emulation board, and provide data transfer to and from the User design.

All AETEST source code is included on the CD-ROM shipped with your DN7000K10PCI Logic Emulation kit. AETEST can be installed on a variety of operating systems, including:

- DOS and Windows 95/98/ME using DPMI (DOS Protected Mode Interface)
- Windows 98/ME using a VxD driver
- Windows 2000/XP (Windows WDM)
- Windows NT
- Linux
- Solaris

2.1 Functionality

The AETEST utility program contains the following tests:

- PCI Test
- Memory Tests (SRAM & DDR)
- FLASH Test
- Daughter Card Test (with or without cables)
- BAR Memory Range Tests

AETEST also provides the user with the following abilities:

- Recognize the DN7000K10PCI
- Display Vendor and Device ID
- Set PCI Device and Function Number
- Display all configured PCI devices
- Various loops for PCI device-function and ID numbers
CONTROLLER SOFTWARE

- Write and Read Configuration DWORD
- Write DWORD, Read DWORD and Write/Read DWORD (Same Address)
- BAR Memory Fill, Write and Display
- Configure/Save BAR’s from/to a file

2.2 Running AETEST
The AETEST utility should now recognize the DN7000K10PCI with the DEVICE_ID of 0x1600 and its VENDOR_ID of 0x17DF.

2.3 Compiling AETEST
There are two versions of AETest, one that controls the DN7000K10PCI from a PCI bus, and the other that controls the DN7000K10PCI over a USB connection. The source for the PCI version is found on the User CD:
D:\aetest\The source for the usb version is found on the User CD:
D:\aetest_usb\

You will likely want to interface to your ASIC emulation project using PCI. You may want to start your controller software by modifying and recompiling AETEST.

2.3.1 Compiling AETest for DOS
The DOS version of AETest requires DJGPP. You can find it at http://www.delorie.com/djgpp/
Follow the installation instructions for DJGPP. The download comes with an utility to set your environment variables correctly.
D:\AETest\aetest
Contains the source code for AETest. Copy this directory to your hard drive. Open the file Makefile. This file must be edited to define which operating system you wish to target. Uncomment the line
#DESTOS = DOS_DJGPP
and the line
include Makefile.make
In a DOS shell, run make

2.3.2 Compiling AETest for Windows XP
AETest for Windows requires visual studio to compile.

Copy the directory on the User CD
D:\AETest\aetest
to your local machine. Open the file “Makefile” and uncomment the lines
CONTROLLER SOFTWARE

#DESTOS = WIN_WDM
and include Makefile.make, Run make.

3 Updating the Firmware

Dini Group may release firmware bug fixes or added features to the DN7000K10PCI. If a firmware update is released you will need to

There are two firmware files that Dini Group may release, the first is a Micro controller (MCU) software update that is stored in a flash memory. This update can be accomplished easily from within the USBController application.

The second update that may be required is a Cyclone FGPA core update. The configuration data for the Cyclone FPGA is contained in an Altera serial configuration PROM. This update can be accomplished with the Altera JTAG programming program (Altera USB Blaster cable) and Quartus Programmer.

3.1 Updating the MCU (flash) firmware

To protect against accidental erasure, the MCU firmware cannot be updated unless the board is put in firmware update mode during power-on. Find JP2 on the DN7000K10PCI.

Install a jumper across the pins of JP2. Power on the DN7000K10PCI.

Open the USB Controller program. If the 7000K10PCI powered on in firmware update mode, there will be an “Update Flash” button near the top of the USB Controller window. Click on this button.
When the Open... dialog box appears, navigate to the Firmware image file supplied by Dini Group. The file name should be “flash_flp.hex”. Press OK.

The USB Controller should freeze for about 10 seconds while the firmware update is taking place. When the download is complete, the Log window should print, “Update Complete”

Move Switchblock 1 # 3 to the OFF position to put the DN7000K10PCI back into normal operation mode. Power cycle the board.

3.2 Updating the Cyclone (EEPROM) firmware

The Cyclone firmware is programmed by connecting an Altera JTAG configuration cable to the Cyclone device through the Cyclone's firmware header.

Connect an Altera USB Blaster (or ByteBlaster) configuration cable to the USB port of your computer. When the USB Blaster cable has power, the status LED on USB Blaster turns amber.

Use a .1” IDC cable to connect the USB Blaster cable to the DN7000K10PCI connector P4.
Power on the DN7000K10PCI.

Open the Altera Programmer tool from within Quartus (Figure 16). If USB-Blaster is not selected already, then click on “Hardware Setup…” and select it. In the “Mode” drop-down box, select “Active Serial Programming”.

Click on “Add File” button.

Please select the Cyclone Firmware provided by The Dini Group. This file should be named configFpga.pof.

Select “Program/Configure” box and Verify (option) then click “Start” to program EEPROM device. This should take several seconds. While programming the device, you will notice that the blue LED (DS11) is turned off. After finish programming, the blue LED (DS11) should turn on indicate that the part is done configuration.
Power cycle the DN7000K10PCI. The new firmware is now loaded. The Done LED (blue DS11) should be on (There is error if this LED isn’t on, please repeat the above steps). You can close the programmer tool and disconnect the USB Blaster cable.
Hardware

1 Overview

Below is a block diagram of the DN7000K10PCI

The following sections describe in detail each circuit on the DN7000K10PCI. Note that Schematics appearing in this section are illustrative and may have had details omitted or have been modified for clarity and brevity. If you need to probe, modify or design around the DN7000K10PCI you will need to examine the complete schematics. See the PDF schematic document on the User CD. An assembly drawing has also been provided to help you find probe points on the DN7000K10PCI.

DN7000K10PCI Functionality
The components and interfaces featured on the DN7000K10PCI include:

- PCI hosted logic prototyping system available with up to 3 Altera Stratix-II FPGA's (1508 BGA):
  - EP2S90-5, -4, -3 *
  - EP2S130-5, -4 *
  - EP2S180-5, -4
  * PCI and FPGA interconnect may be limited by FPGA selection.

- FPGA to FPGA interconnect LVDS differential
  - 300Mhz differential (DDR capable) FPGA-to-FPGA
  - Reference designs for integrated I/O SERDES
  - 10x pin multiplexing per LVDS pair with DPA support
  - FPGA A to B, B to C, A to C interconnect: 308 signals each (or >1500 signals each with multiplexing)
  - Greatly simplified logic partitioning

- Separate programmable clock synthesizers (ICS8442)
  - Global clocks (ACLK, BCLK, CCLK)
  - User configurable via CompactFlash, PCI, or USB

- Dedicated 64-bit/66MHz PCI Bridge - QL5064
  - No FPGA resources consumed for PCI protocol.
  - Maximum PCI data transfer rate: 533mb/s
  - Programmable Target Prefetching/Write Posting
  - PCI2.2 Compliant
  - Zero Wait State Master and Target Bursting
  - Five Independent Master DMA Channels:
    - Transmit
HARDWARE

- Receive
  - 1 -- Single PCI Access (SPCI)
    - DMA Chaining/Scatter Gather
    - Mailboxes, Interrupts
- Advanced FPGA configuration via PCI, USB2.0, or CompactFlash
  - Partial reconfiguration support on all FPGAs
- DDR2 SODIMMs -200MHz (x3)
  - 1 per each FPGA
  - 64-bit data width, 200MHz operation
  - PC2-3200/PC2-4200
  - Addressing and power to support 4GB in each socket
  - Verilog/VHDL reference design provided (no charge)
  - DDR2 SODIMM data transfer rate: 25.6Gb/s
  - Alternate SODIMM's available (consult factory):
    - QDR SSRAM
    - SSRAM pipeline/flowthrough, NoBL/ZBT
    - FLASH
    - Micron connectors
    - Micron RLDRAM
- FPGA configuration via PCI, CompactFlash and USB
- Status LEDs
- Standalone operation with off-the-shelf ATX power supply.
- Two, 400-pin expansion connectors with 348+ connections + clocks
2 Configuration Circuit

2.1 Overview
The goal of the configuration circuit on the DN7000K10PCI is to allow the user to configure his FPGAs using any host interface. The configuration system on the DN7000K10PCI allows configuration over PCI, USB, JTAG, or automatic configuration from a CompactFlash card.

The circuit is designed to provide an easy configuration solution that will work out-of-the-box for most users. For special configuration requirements, the configuration
circuitry is programmable. The verilog code for the configuration FPGA and the C code for the microcontroller are both provided on the reference CD. The C code for the PCI controller program and USB Windows GUI controller program are also included on the User CD.

2.2 The Cyclone 2 FPGA

The configuration circuitry of the DN7000K10PCI is built around an Altera Cyclone 2 FPGA. The TODO interface of the user FPGAs is connected directly to the general purpose IOs of the Cyclone 2, allowing the maximum flexibility of configuration. The Cyclone 2 also shares connectivity with the three user FPGAs over a 40-bit Main bus, allowing fast transfers from a computer to the user design over PCI or USB. The Cyclone 2 FPGA also provides IO expansion for the Cypress Microcontroller. The Cyclone 2 FPGA comes preloaded with a core that provides a way to program the Stratix 2 FPGAs over PCI, USB and CompactFlash.

The Cyclone FPGA is connected to the Cypress microcontroller’s address and data busses, and the control registers within the Cyclone 2 FPGA that control FPGA configuration are memory-mapped into the MCU’s address space.
Figure 18 Cyclone 2 IO Connections
2.2.1 Cyclone Configuration

The Cyclone 2 FPGA is configured from an Altera serial prom. The Cyclone’s configuration mode is hard-wired into Active Serial mode. After power up, the Cyclone automatically clocks an external PROM, U41, which programs the FPGA over the serial configuration data pin DIN.

A green LED, DS24, lights when the DONE pin is high. This signal is driven by the Cyclone 2 FPGA when it is configured and running.

Both the Cyclone and the serial prom are connected in a JTAG chain attached to J14. This header is used when performing firmware updates to update the PROM.

As soon as the Cyclone 2 FPGA is configured, it resets the Cypress microcontroller. Pull-downs on the PROG pin of FPGAs A B and C ensure that the FPGAs cannot be active unless the Cyclone 2 is successfully configured.

2.2.2 CompactFlash

The CompactFlash card interface is connected to the IOs of the Cyclone 2 FPGA.

2.2.3 PCI communication

To enable configuration over PCI, the Cyclone 2 is connected to a subset of the Quicklogic 5064 PCI interface. Due to the available number of IOs on the Cyclone 2 FPGA, only 32 of the 64 data bits in the Quicklogic interface are connected to IOs on the Cyclone 2. Also, the Cyclone 2 does not connect to any of the signal required for DMA operation over PCI.
From the PCI host perspective, the address range BAR0 is directed to the Cyclone 2 FPGA, and BAR1-BAR7 is directed towards the Stratix 2 FPGA A.

Since FPGA A and the Cyclone 2 both can access the Quicklogic 5064 back-end, only one must communicate with the QL5064 at a time. The signals SP_PCI_REQ and LX_PCI_ACK are used to control communication with the QL5064.

2.2.4 RS232
The DN7000K10PCI has four RS232 headers. One (P2) is used by the microcontroller unit to provide configuration feedback and control. Connect this to a computer terminal at 19200 bitrate, no parity, no handshaking.

2.2.5 IIC
There is a single IIC bus on the DN7000K10PCI connecting all IIC enabled chips on the board. On this bus are three MAX1617A temperature sensing chips (U3, U4, U24), and a serial eprom. The temperature sensors on the IIC bus are polled about once per second by the MCU to read the temperature of each FPGA. The temperatures can be read back over RS232, AETEST, AETEST_usb, or the Windows GUI.

2.3 Configuration Options
The DN7000K10PCI allows FPGA configuration from any of four methods. USB, PCI, JTAG or CompactFlash.

When a Stratix FPGA is configured, the DONE pin on the FPGA is pulled high. The DN7000K10PCI has a blue LED attached to the DONE signal of each to indicate the state of the DONE pin on the three Stratix 2 FPGAs and on the CycloneII configuration FPGA.

![DONE LEDs Diagram](image.png)
2.3.1 Jtag
Jtag is the only configuration method on the DN7000K10PCI that does not use the Stratix 2 Fast Parallel configuration interface. When programming the user FPGAs over a JTAG cable plugged into J6, the DN7000K10PCI configuration circuitry is not used.

A JTAG connection is required to use some Altera configuration features. Configuration over JTAG is slower than TODO SelectMap. You can still use the CompactFlash or USB interfaces to control clock settings if you plan to configure through JTAG.

To configure using JTAG, we recommend using Altera USB Blaster, or Altera ByteBlaster.

**JTAG ByteBlaster II or USB-Blaster**

*Figure 21 FPGA JTAG Header*

The JTAG signals TMS is bussed to all three Stratix 2 FPGAs. TDO connects to FPGA A, the TDO of FPGA A is connected to TDI of FPGA B, the TDO of FPGA B connects to the TDI of FPGA C and TDO of FPGA C is connected to the TDI of J13. TCK is buffered and passed to each FPGA in a point-to-point fassion.
If you ordered your DN7000K10PCI with one or more FPGAs not installed (Option FPGA A NONE, FPGA B NONE, or FPGA C NONE) then a bypass resistor is installed connecting the TDI pin to the TDO pin of the uninstalled FPGA. This is so the JTAG chain will remain intact when FPGAs are missing.

### 2.3.2 CompactFlash

When the DN7000K10PCI powers on, the microcontroller reads the contents of any CompactFlash card that is in the CompactFlash slot. The microcontroller by default opens a file on the root directory named “Main.txt” if it exists. This file contains instructions for the configuration circuitry to configure the Stratix 2 FPGAs.

To create a CompactFlash card to control the DN7000K10PCI configuration, insert the CompactFlash card into a card reader (provided) and connect it to a PC. Create a file on the root directory of the card and call it “Main.txt”

In main.txt, write a series of configuration commands, separated each by a new line. A valid command is one of the following:
// <comment>
FPGA A:<filename>
FPGA B:<filename>
FPGA C:<filename>
CLOCK FREQUENCY: <clockname> N <number> M <number>
SANITY CHECK: <yn>
VERBOSE LEVEL: <level>
MEMORY MAPPED 0x<SHORTADDR> 0x<BYTE>
MAIN BUS 0x<WORDADDR> 0x<WORDDATA>

<comment> can be any string of characters except for newline.
<filename> can be the name of a file on the root directory of the CompactFlash Card.
<number> can be any one or two digit positive integer in decimal
<clockname> can be [A,B,C] A is ACLK, B is BCLK, C is CCLK clock synthesizer.
<yn> can be the letter y or the letter n
<level> can be 0,1,2 or 3
<portnumber> can be 1,2,3, or 4. The DN7000K10PCI only has 1 user RS232 port (1) so 2-4 will cause no operation.
<fpganame> can be [A,B,C].
<SHORTADDR> is a 4-digit hex number (16 bits)
<BYTE> is a 2-digit hex number (8 bits)
<WORDADDR> 8-digit (32 bit) hex number representing a main bus address
<WORDDATA> 8-digit (32 bit) hex number containing data for a main bus transaction
The following table describes the function of each of the available main.txt commands.

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>// &lt;comment&gt;</td>
<td>The MCU performs no operation.</td>
</tr>
<tr>
<td>VERBOSE LEVEL: &lt;level&gt;</td>
<td>This command will set the amount of output the MCU will produce over the RS232 port during configuration. When level is set to 0, the MCU will produce only error output. Before this command is executed, the level is set to the default value 3.</td>
</tr>
<tr>
<td>FPGA A:&lt;filename&gt;</td>
<td>The FPGA “A” will be configured with the file named by &lt;filename&gt;</td>
</tr>
<tr>
<td>FPGA B:&lt;filename&gt;</td>
<td>The FPGA “B” will be configured with the file named by &lt;filename&gt;</td>
</tr>
<tr>
<td>FPGA C:&lt;filename&gt;</td>
<td>The FPGA “C” will be configured with the file named by &lt;filename&gt;</td>
</tr>
<tr>
<td>SANITY CHECK: &lt;yn&gt;</td>
<td>If &lt;yn&gt; is set to y, then the MCU will examine the headers in the .rbf files on the CompactFlash card before using them to configure each FPGA. If the target FPGA annotated in the .rbf file header is not the same type as the FPGA the MCU detects on the board, it will reject the file and flash the error LED. Before this command is executed, &lt;yn&gt; is set to the default value y. If you want to encrypt or compress your rbf files, you will need to set &lt;yn&gt; to n.</td>
</tr>
<tr>
<td>MAIN BUS 0x&lt;WORDADDR&gt;</td>
<td>Writes data in &lt;WORDDATA&gt; to the address on the main bus interface at &lt;WORDADDR&gt;. This command only makes sense in the context of the Dini Group reference design, unless your design implements a compatible controller on the main bus pins.</td>
</tr>
<tr>
<td>MEMORY MAPPED 0x&lt;SHORTADDR&gt; 0x&lt;BYTE&gt;</td>
<td>Writes to an address in the MCU memory space. The Programmer's guide in the Controller software section contains a memory map.</td>
</tr>
</tbody>
</table>
HARDWARE

CLOCK FREQUENCY: \(<\text{clockname}> \ N \ <\text{number}> \ M\> \text{<number>}

The MCU will adjust the clock synthesizer producing clock \(<\text{clockname}>\) to multiply its reference frequency by \(<M>\) and divide it by \(<N>\).

Note that the clock synthesizers have a limited bandwidth, and for clocks A B and C, the reference frequency \(* M\) must fall in the range 250Mhz-700Mhz.

The reference frequencies are
ACLK 25Mhz
BCLK 14.18Mhz
CCLK 16Mhz

An example main.txt file:

<table>
<thead>
<tr>
<th>VERBOSE LEVEL: 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>// This will prevent the MCU output over RS232 to speed up configuration</td>
</tr>
<tr>
<td>FPGA A:a.rbf</td>
</tr>
<tr>
<td>// this will load the configuration a.rbf into FPGA A</td>
</tr>
<tr>
<td>CLOCK FREQUENCY: A N 4 M 10</td>
</tr>
<tr>
<td>// This will cause Aclk frequency to be</td>
</tr>
<tr>
<td>// 25*10=250 / 4 = 62.5Mhz</td>
</tr>
<tr>
<td>MAIN BUS: 0x0000 0x0001</td>
</tr>
<tr>
<td>// Writes to a register in FPGA A.</td>
</tr>
</tbody>
</table>

Even if you are not planning to configure your Stratix 2 FPGAs using a CompactFlash card, you may want to leave a CompactFlash card in the socket to automatically program your global clock. (Clocks may also be programmed using the provided USB application, or over the MCU RS232 terminal.)

2.3.3 USB

The USB interface on the DN7000K10PCI is provided by the Cypress microcontroller unit. The Cypress microcontroller is programmed to interrupt when it receives a USB vendor request.

When the MCU receives over USB a Bulk Transfer type request, it does not interrupt. The raw data contained in the bulk transfer is driven out on the GPIF pins of the MCU to the Cyclone 2. This data is transferred synchronous to the signal MCU_IFCLK. As long as the signal GPIF_CTL is held high by the MCU, the Cyclone 2 clocks MCU_IFCLK to receive the USB data.

When data is written to the Cyclone 2 from a bulk transfer over the MCU’s GPIF interface, the Cyclone 2 either writes that data onto the Fast Parallel interface of the Stratix FPGAs, or onto the Main bus using the Main Bus interface described in the Reference Design chapter.
The control register FPGA_SELECT within the Cyclone 2 determine to which interface this data is routed to.

2.3.4 PCI
The PCI interface on the DN7000K10PCI is primarily used directly by the Stratix 2 FPGA. The Quicklogic 5064 implements the PCI interface and delivers the data directly over the QL interface to FPGA A. However, when the host machine makes a read or write transaction to the DN7000K10PCI on a BAR0 address over PCI, the data is instead delivered to the Cyclone 2 configuration FPGA. This allows the Cyclone 2 to configure FPGAs, change other configuration settings, or communicate to FPGAs B and C over PCI.

The program AETest, supplied on the User CD along with the AETest source code, allows the user to configure FPGAs and change configuration settings over PCI. When used with the Dini Group reference design, or if the user has created a compatible Main Bus interface, the AETest program can also communicate directly to the FPGA A, B and C designs.

2.3.4.1 Configuration

To configure an FPGA over PCI, the host program writes the following instructions to BAR0 of the DN7000K10PCI.

First, the host instructs the Cyclone 2 FPGA to “select” an FPGA for configuration. Write one word of data to BAR0, 0x0208. The data 0x11 represents FPGA A, 0x12 is FPGA B, 0x13 is FPGA C.

Next, the host instructs the Cyclone 2 FPGA to assert the PROGn signal (Reset) of the selected FPGA’s selectmap interface. This causes the FPGA to un-configure regardless of the reconfiguration setting made by the rbf file’s reconfigure setting. The PROGn signal must be asserted once before the FPGA is configured for the first time. Write a word to the address BAR0, 0x208. The data word 0x11 represents FPGA A, 0x12 is FPGA B, 0x13 is FPGA C.

When the FPGA is read to configure, it pulls the selectmap signal INTn low. The configuration process should pause until this occurs. To read the current value of INTn on the selected FPGA, read from the BAR0 address 0x208. Bit 5 represents the value of the selected INTn signal.

After the INTn signal is detected, the Host should de-assert PROGn (Reset). Write to BAR0 address 0x208 the data word representing the selected FPGA. 0x11 is FPGA A, 0x12 is FPGA B, 0x13 is FPGA C.

The configuration stream for the FPGA is then sent to BAR0, address 0x210, one byte at a time. Some time during the configuration stream byte loading process, a startup
sequence is sent to the FPGA and the FPGA becomes operational. This startup sequence is contained within the rbf file.

To determine if the selected FPGA is currently configured (i.e., configuration was successful), read from BAR0 address 0x208. The bit 5 contains the state of the DONE FPGA pin, the bit 6 contains the state of the FPGA INIT signal.

By convention, the host program should leave the Cyclone in the FPGA deselected state. To deselect the FPGA, write to BAR0, address 0x208 the data 0x10. (FPGA SELECT NONE)

2.3.4.2 Config Space

PCI can also be used to control other configuration functions on the DN7000K10, such as temperature sensors and clocks. This is done by altering the data in the XDATA memory space of the configuration MCU.

To write to the MCU address space, access the DN7000K10’s BAR0 at the address MCU_BAR_ADDR 0x258. Send a 32-bit word of data. This data is decoded as follows:

Bits 31-16: address in the XDATA space. (only addresses 0xDF00-0xDFFF reside in the Cyclone 2)

Bits 15-8: Ignored

Bits 7-0: The Data to write

To read from the MCU address space, access the DN7000K10’s BAR0 at the following 32-bit address:

Bits 31-24: The DN7000K10PCI's BAR0

Bits 23-16: the lower 8 bits of XDATA address you would like to read. This corresponds to addresses 0xDF00-0xDFFF of the XDATA address. (Only addresses 0xDF00-0xDFFF reside in the Cyclone 2)

Bits 15-0: 0x0260

2.3.4.3 Main Bus Space

PCI can also be used to send information to and from your Stratix 2 user design through the Cyclone 2 FPGA. Communication directly to the user design can also be accomplished from FPGA A by communicating directly with the QL PCI backend interface. This method is an order of magnitude faster, and allows the use of advanced PCI features like DMA. See Hardware: PCI interface. Communication with the FPGAs
HARDWARE

through the Cyclone occurs using the Main Bus interface. For information on the main bus interface see Reference Design: Main Bus Interface.

To write to the main bus interface, write to BAR0 address QLPCI_REG_MBADDR with the 32-bit value representing the main bus address you would like to write to. Then, write a second PCI write to address QLPCI_REG_MBWRDATA with 32-bit data representing the data that you would like to write to main bus. After the Cyclone 2 has received a write to both the MBADDR and MBWRDATA registers, it will write to the main bus interface.

To read from the main bus interface, first write to BAR0 address QLPCI_REG_MBADDR with the 32-bit value representing the main bus address you would like to read from. Then, read from BAR0, QLPCI_REG_MBRDDATA. The returned value will be the value read off the main bus at the selected address. When an error has occurred (No FPGA responded to the read request) the Cyclone will return the value 0xABCDABCD.

2.4 FPGA configuration Process

For information regarding the JTAG interface and configuration, See Altera publication TODO, Stratix 2 configuration guide.

When configuring over PCI, USB or CompactFlash, the FPGAs are configured over the Stratix 2 SelectMap bus.

All SelectMap signals are connected directly to the Cyclone2 FPGA. The Fast Parallel Configuration signals are:

- NSTATUS
- nCONFIG
- CONF_DONE
- DATA[7..0]
- nCE
- CLK
- CS
- NWS
- NRS
- NCS
- CLKUSR
- DEV_OE
- DEV_CLRn
- RunLU
- PLL_ENA
- D[0-7] Fast Parallel data signals.
HARDWARE

Config_n  Active low asynchronous reset to the configuration logic. This will cause the FPGA to become unconfigured. The documentation refers to this signal as PROGn

CONF_DONE  After the FPGA is configured, it is driven high by the FPGA.

INIT  Low indicates that the FPGA configuration memory is cleared. After configuration, this could indicate an error.

RDWR_B  Active low write enable. The documentation refers to this signal as RDWR

BUSY  When busy is high, the SelectMap configuration stream must stop until BUSY goes low.

CS_n  Chipselect. The documentation refers to this signal as CSn

CCLK  Signals D[0:7], DONE, RDWR_B and CS_B are clocked on CCLK

NCE  Left selected.

Each Stratix 2 FPGA has a complete set of SelectMap signals connected point-to-point to the Cyclone 2, except for FPGA B and C, who share signals D[0-7]. All signals are 2.5V CMOS signals except for D[0-7] of FPGA A (Signals SELECTMAP_3V_D[0-7]), which are 3.3V CMOS.

All commands required to configure a Stratix 2 FPGA are created and embedded in the .rbf files created by the Altera TODO program. The DN7000K10PCI does not interact with the SelectMap interface other than to reset the FPGA using the PROGn-INTn-PROGn reset sequence described in UG071, and to copy a rbf stream file unaltered to the FPGA over the data pins D[7-0]. Select map commands can be issued to the Stratix 2 FPGA from the host using the same interface used to configure and update FPGA.

After a Stratix 2 FPGA is configured, it asserts the signal DONE. On the DN7000K10PCI, these signals have an LED attached to each DONE signal placed near the upper corner of each FPGA.
If your Stratix 2 FPGA design is failing to produce the intended (or any) results, you should check the DONE light above the FPGA to make sure it is configured correctly. The design files created by Altera TODO software contain a CRC check, so if the Stratix 2 FPGA detects a CRC failure, there was a transmission error during configuration and the DONE light will not glow. The DN7000K10PCI microcontroller also checks the design files you send to make sure they are compiled for the FPGAs that are installed on your board. If they are not, then the microcontroller unit halts the configuration process. As a result, when the DONE light goes on, you will know that the configuration process was successful.

2.5 MCU
The operation of the Cyclone 2 is monitored and controlled by a Cypress CY7C68013 microcontroller. The microcontroller also has a USB 2.0 interface that can be used to monitor the board, control configuration, or transfer data to and from the user FPGA design. Basic operation can be controlled over an RS232 link from a computer terminal.

2.5.1 RS232
The primary method of user interaction with the DN7000K10PCI configuration circuitry is the MCU’s RS232 port (P2). The Cypress CY7C68013 has two RS232 pins that are buffered through a 12V voltage translation buffer for use with a standard computer serial port.
The RS232 port will be able to communicate with a standard PC serial port set to 19200 baud, 8 data bits, no parity, no handshaking. When you connect a computer terminal to the port and power on the DN7000K10PCI, the firmware loaded on the microcontroller unit will display a menu on the terminal. This menu will allow you to control the basic configuration options of the DN7000K10PCI including configuration, clock frequencies.

2.5.2 Clocks
The Cypress CY7C68013 is also responsible for configuring the global clocks of the DN7000K10PCI. The Cypress CY7C68013 MCU reads the file “main.txt” from the CompactFlash card in the socket (J2), and follows the users clock configuration commands.
The 3 ICS8442 clock synthesizers on the DN7000K10PCI used for generating the global clocks, ACLK, BCLK and CCLK, share a serial configuration bus connected to the MCU to program them. The ICS8442 frequency synthesizers are capable of multiplying and dividing the reference frequencies provided by their reference crystals. The MCU loads the user's desired multiplication “M” value, and division, “N” value into the settings registers in the ICS8442 chip.

2.5.3 LEDs
The MCU is connected to 4 red LEDs that are visible from outside the PC case when the DN7000K10PCI is plugged into a PCI slot. The LEDs flash a status code during and after configuration.

All four flashing LEDs means there has been an error configuring at least one FPGA.

2.5.4 Memory space
The XDATA memory space of the MCU is partitioned into four sections.

- 0x0000 - 0x1FFF  internal data/program memory
- 0x2000 - 0xCFFF  external SRAM
- 0xDFF0 - 0xDFFF  memory mapped registers (no external memory accesses)
- 0xE000 - 0xFFFF  reserved by MCU, RD/WR strobes not active in this region

The internal data memory region is mapped to an internal SRAM in the Cypress MCU. When the microcontroller code calls memory access from this region, the external Address and Data busses are not used. After power on reset, the MCU reads from the IIC Eprom connected to the MCU_EPROM signals and fills this internal memory before allowing the PC to run. The code in this section of memory contains core functions of the Dini Group firmware, like setting up the interrupt registers, communicating with USB, and allowing firmware updates.

The external SRAM is used for heap data.

The memory mapped register region (The DF region) contains registers in the Cyclone 2 FPGA that control FPGA configuration.

The program memory space of the MCU is directly mapped to the external Flash memory.

When the Cypress MCU is reset (which happens after the Cyclone 2 is configured), it loads its boot code into its 8kB of internal memory from a serial EEProm (U1). The code in the EPROM instructs the MCU to execute code located on the FLASH memory (U30). The code in the EEPROM and FLASH is located on the user CD.
Communication over the MCU memory bus to the Cyclone 2 is synchronized to the 24Mhz MCU_CLK (X1). For information regarding the timing of transactions on this bus, see the Cypress CY7C68013 user manual.

The Configuration FPGA is connected to the MCU_DATA[7:0] signals, the MCU_ADDR[15:0] signals and the MEM_OE signal, allowing it to decode address accesses of the MCU. The Configuration FPGA is programmed to respond to accesses in the XDATA address space in the address range of 0xDF00 to 0xDFFFF.

Communication over the MCU memory bus to the Config FPGA is synchronized to the 24Mhz MCU_CLK (X3). For information regarding the timing of transactions on this bus, see the Cypress CY7C68013 user manual.

The following registers implemented in the Configuration FPGA are accessible as part of the MCU’s XDATA address space.
<table>
<thead>
<tr>
<th>Register Name</th>
<th>XDATA Address</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPGA_SELECT</td>
<td>DF0C</td>
<td>FPGA_select[5:0] = bits 5:0</td>
</tr>
<tr>
<td>PPC_RS232_ABSELECT</td>
<td>DF0D</td>
<td>Not used on this board</td>
</tr>
<tr>
<td>PPC_RS232_CDSELECT</td>
<td>DF0E</td>
<td>Not used on this board</td>
</tr>
<tr>
<td>FPGA_COMMUNICATION</td>
<td>DF39</td>
<td>0x01 disable main bus</td>
</tr>
<tr>
<td>NEW_BOARD_VERSION</td>
<td>DFFE</td>
<td>Returns 0x0F</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This uniquely identifies a DN7000K10PCI</td>
</tr>
<tr>
<td>OLD_BOARD_VERSION</td>
<td>DFFF</td>
<td>Returns 0xF0</td>
</tr>
<tr>
<td>PENDING_CLKS</td>
<td>DF40</td>
<td>0x01 ACLK synthesizer will be reset</td>
</tr>
<tr>
<td>ACLK_N_VAL</td>
<td>DF29</td>
<td>Divider setting for the ACLK synthesizer</td>
</tr>
<tr>
<td>BCLK_N_VAL</td>
<td>DF31</td>
<td>Divider setting for the BCLK synthesizer</td>
</tr>
<tr>
<td>DCLK_N_VAL</td>
<td>DF33</td>
<td>Divider setting for the DCLK synthesizer (requires reset to take effect)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0x00 – Divide by 1</td>
</tr>
<tr>
<td>BCLK_M_VAL</td>
<td>DF32</td>
<td>Multiplier setting for the ACLK synthesizer (requires reset to take effect)</td>
</tr>
<tr>
<td>ACLK_M_VAL</td>
<td>DF30</td>
<td>Multiplier setting for the BCLK synthesizer (requires reset to take effect)</td>
</tr>
<tr>
<td>DCLK_M_VAL</td>
<td>DF34</td>
<td>Multiplier setting for the DCLK synthesizer (requires reset to take effect)</td>
</tr>
<tr>
<td>TSTEMPERATURES[0]</td>
<td>DF50</td>
<td>FPGA A temperature</td>
</tr>
<tr>
<td>TSTEMPERATURES[1]</td>
<td>DF51</td>
<td>FPGA B temperature</td>
</tr>
<tr>
<td>TSTEMPERATURES[2]</td>
<td>DF52</td>
<td>FPGA C temperature</td>
</tr>
</tbody>
</table>

These registers can be written to from the USB interface. See *USB Software: Programmers Guide*.

### 2.5.5 USB

The Cypress CY7C68013 has a built-in USB 2.0 interface. The USB type B connector on the DN7000K10PCI (J1) is connected directly to the USB pins on the Cypress MCU.
USB Interface Type B

The USB protocol is completed by the Cypress CPU.

The Cypress receives a 24Mhz clock from an oscillator (X3). The Cypress internally multiplies this clock to 480Mhz for USB 2.0 and 48Mhz for GPIF operation. The core runs at 24Mhz along with the external memory interface. Communication over this external memory interface is clocked using the MCU_IFCLK signal driven from the MCU at 48Mhz. (The Cyclone communicates over main bus with the Stratix 2 FPGAs using a separate 48Mhz oscillator (X1) and distributes this clock to each FPGA including itself)
2.5.6 CompactFlash

The CompactFlash card socket pins are bussed among the Cypress MCU GPIF pins, the Cyclone 2 FPGA IOs, and the CompactFlash card socket. After reset, the MCU uses this connection to look for and read the contents of the file main.txt on the CompactFlash card. The main.txt file contains instructions for configuring the user design into the three Stratix 2 FPGAs.

After reading the configuration instructions, the MCU reads the headers of the user’s FPGA design (“.rbf”) files and verifies that they target the correct type of FPGA that are installed on your DN7000K10PCI. This will prevent damage to the FPGA from an incorrect or corrupt .rbf file. This behavior can be turned off.

If this check is passed, MCU uses its memory mapped interface with the CycloneII to instruct the CycloneII to read the CompactFlash card and configure the Stratix 2 FPGAs over SelectMap bus.

3 Clocking

The clocking circuitry on the DN7000K10PCI is designed for high-speed operation. The flexible clock design should meet the most difficult clocking needs, allowing 8 totally asynchronous, controllable clock sources for each FPGA.

All clocks operating above 100Mhz are fully differential, LVDS signaled, low skew, low jitter clocks.
HARDWARE

From the above diagram, the global clocks are listed here.

ACLK, BCLK, CCLK. These global clocks are supplied by ICS8442 frequency synthesizers. They are configured from the MCU to output a user-specified frequency from 31 to 700Mhz. They are each distributed to FPGAs A B and C.

LVDS_CLK. This clock is a copy of the ACLK. It is buffered 1:8 by clock buffers and distributed to the DPA clock inputs on the FPGA. This clock can be used for High-speed DPA functions on the FPGA. Since each FPGA receives this clock with as much as 500ps of skew, this clock should not be used for synchronous IO.

CYCLONE_CLK. This clock is driven from the CYCLONE configuration FPGA, and can be adjusted by changing the Cyclone FPGA’s configuration stream. By default the FPGA outputs a 48Mhz clock. This clock is used by the reference design’s “Main Bus” interface.

DC_CLK- This clock is driven by either of the two daughtercard interfaces. A differential clock multiplexer selects from the two inputs. This setting can be changed using the USB Controller interface, or the CompactFlash main.txt interface.
3.1 Global Clocks

The three main global clocks are driven by ICS8442 clock synthesizers, each capable of producing frequencies of 700Mhz (or greater). The clock synthesizers can be programmed from a CompactFlash card, from the GUI application (See Chapter X, the USB Application) or left at their default values (ACLK 100Mhz, BCLK 57.2Mhz, CCLK 64Mhz).

Even if you are using PCI or USB as a host interface for the DN7000K10PCI, it may be convenient for you to create a CompactFlash card with a main.txt file in it to configure the clocks automatically every time the board powers on.

Each ICS8442 has an internal multiplication PLL that can operate between 250 and 700 Mhz. With 1, 2, 4, or 8x division on the output, the possible output frequencies are 31.25 – 700Mhz. VCO_SEL can be used to disable the PLL, so ACLK BCLK and CCLK can operate at or below their fundamental crystal frequencies - 25Mhz, 14.3Mhz and 16Mhz respectively.

The Serial configuration bus of the ICS8442 is connected to the Cypress MCU GPIF pins and controlled through software.

The crystal inputs are parallel resonant, fundamental mode.
3.2 LVDS Clocks

The 8442 outputs are connected to a 1:8 LVDS buffer, and distributed to the FPGAs. Aclk and Bclk are also distributed to the expansion headers as well.

3.3 User Clock

3.4 PCI Clock

4 Reset Topology

The DN7000K10PCI is protected from undervoltage and over temperature by a reset circuit. When the board powers on, a voltage monitor waits until all voltages are above their minimum voltage levels, then deasserts reset. The Cyclone 2 distributes the reset signal to all FPGAs and the Microcontroller unit, so until the Cyclone 2 is configured,
reset remains asserted. During this time, the Stratix 2 FPGA cannot be programmed via USB, CompactFlash, PCI or JTAG.

The user may also assert reset by pressing button S2, “Reset”. This will trigger the reset signal “SYS_RSTn” which is monitored by the Cyclone FPGA. When SYS_RST is asserted, the Cyclone FPGA resets the Stratix 2 FPGAs, causing them to lose their configuration data and deactivate. The Cyclone also causes a reset on the Microcontroller unit, which will cause the microcontroller to reload configuration instructions from the CompactFlash card, after reset de-asserts. USB contact will be lost with the USB host, and the DN7000K10PCI and host operating system will behave as if the board were just plugged in to USB.

There is a second button, S1. When this button is pressed, the signal “RESET_FPGAs” is asserted. On this schematic, this signal is also called, “NIOS reset”. This signal is sent to the Stratix 2 FPGAs on a user-accessible IO pin, and could be used by the user design as a reset signal (Does not cause loss of configuration data). This signal can also be activated from the USB Controller program, or the CompactFlash card.
The 5.0V level is not monitored. This is because a large number of ATX power supplies/hosts have poor 5.0V regulation. Since the DN7000K10PCI can operate with 5.0V at very low levels, this voltage is not monitored.

The same that the SYS_RSTn signal is held low by the LT2900 is 50us. The S2 “reset” button forces the voltage monitor to register a reset condition. The PWR_FAULT signal is driven by the 1.2V monitors. Each 1.2V rail has a dedicated voltage monitor, with wire-OR outputs onto the PWR_FAULT signal.

Each FPGA is also connected to a temperature monitor. The Stratix 2 FPGA can easily overheat if a heatsink and fan are not used. The recommended operating temperature for the Stratix 2 is 85 degrees C. The absolute maximum temperature for operation is 125 degrees C. If at any time the junction temperature of the Stratix 2 exceeds 85 degrees, the Microcontroller will reset the FPGAs, causing them to lose their configuration data. An overheating FPGA could be the result of a misconfiguration, a clock that is set incorrectly, or an inadequate heatsink unit. The heatsink and fan assembly that comes with the DN7000K10PCI is appropriate for dissipating the amount of heat energy available through a PCI slot without the auxiliary power connector (25W total for the card). If you are operating the DN7000K10PCI at very high speeds in stand alone mode and you are causing heat overload resets, you may need to install a larger heatsink, or increase the system airflow.
FPGA Temperature Monitor (Optional)

This circuit shows the MAX1617 temperature monitor. The IIC bus is connected to the Cypress microcontroller. There are three temperature monitors sharing the IIC bus, each device configured to a different address on the bus.

5 Power

The DN7000K10PCI gets its power from the 5.0V and 3.3V rails of the PCI card edge connector. It can also be operated in stand-alone mode with a 20-pin ATX power supply connector.

The PCI slot is capable of sourcing

The main rails of the DN7000K10PCI are:

- 1.2V – This is the main power supply rail used for the internal digital logic of Stratix 2 FPGAs.

- 1.8V – This is used for IO signaling and interal logic of DDR2 SDRAM memory.

- 2.5V – This is used to power FPGA interconnect with low-power LVDS. It is also used as the analog power supply on the Stratix 2 FPGAs.

- 3.3V – This voltage supplies the LVDS clock distribution trees. It is also used to power the LVTTL interfaces of the Cypress microcontroller, and Quicklogic 5064 PCI bridge.
- 5.0V – This voltage is used to supply power to the 1.2, 2.5 and 1.8V switching power supplies. It also powers the FPGA cooling fans, some Gigabit optical modules, and the PCI signaling. If the PCI slot isn’t providing enough power, then a Hard Drive 4-pin power cable can be connected to the board (from the same ATX power supply) to reduce the voltage droop on 5V. Please note that the board is capable of exceeding the 25W limit of the PCI connector (depending on the desity of the FPGAs utilized, and the operating frequency).

The DN7000K10PCI also has these secondary rails:

- 0.9V – This voltage is used to terminate the SSTL18 signaling of the DDR2 memory module.

- +12V – This rail is passed directly from the PCI edge connector and ATX power connector to the Micropax expansion header. See Chapter X, Section X, Expansion Headers. Note that the fuse between +12V and the expansion headers is not installed on the board.

- -12V – This rail is passed directly from the PCI edge connector and ATX power connector to the Micropax expansion header. See Chapter X, Section X, Expansion Headers. Note that the fuse between -12V and the expansion headers is not installed on the board.

There are test points for measuring the voltage levels of each rail near the top left of the DN7000K10PCI. Each rail is monitored by a voltage monitor circuit, and will cause a reset if any of the primary supplies drop 5% or more below their setpoints.

There are also LEDs next to each testpoint to indicate the presence of each voltage rail. These LEDs do not indicate that a rail is within 5% of its setpoint, only that the rail is present and above 1.6V. A power OK led shows the status of the ATX power supply’s PWR_OK signal. If this LED is lit, then +5.0V and +3.3V (and +12V –12V) are within 5% of their setpoints.

**Voltage Indicators**

![Voltage Indicators](image-url)
5.1 Switching power supplies

The main power rails for the Stratix 2 FPGAs are produced on board with three 20A switching power supplies, one for each of 1.8V, and 2.5V. The 1.8V power rail supplies the DDR2 SODIMM modules and the IO power for the FPGA pins that are connected to that module. The 2.5V power is used to supply IO current for the LVDS interconnect between FPGAs.

Switching Power Supply 1.8V @ 20A

The DN7000K10PCI is shipped with a fan mounted above the power supplies to help keep them cool. If you need to remove this fan, the DN7000K10PCI will function properly without it.

Each of the primary power rails (3.3, 2.5, 1.8, 1.2A, 1.2B, 1.2C) is monitored for undervoltage. If the voltage monitor circuit detects a low voltage, it will hold the board in reset until the supply is back within 5% of its setpoint.

Reset Circuit

Note: Set +5V to trip at 3.0V
The VCCINT power pins of the Stratix 2 FPGAs are supplied by 1.2V switching power supplies. Each FPGA has a dedicated switching power supply.

**VCCINT Switching Regulator 2.5V @ 10A**

![Diagram of VCCINT Switching Regulator](image)

Note: Add provision for a fan to be mounted over the PSU’s, COFAN USA P/N P-410H03.

Figure 37

These smalled power units are capable of outputting 10A each.

### 5.2 Secondary Power Supplies

The secondary power supplies are derived from a primary supply.

#### 5.2.1 DDR2 Termination Power

DDR2 memory modules use the SSTL18 signaling standard. Properly terminating SSTL18 requires a termination power supply of 0.9V. Since as much as 1.6 Amps of termination current are needed, a switching power supply is required.

![Diagram of DDR Switching Power Supply VTT - 0.9V @ 3A](image)

The ML6554 produces up to 3A of the required 0.9V termination power rail along with a stable 0.9V reference voltage supply.
5.3 Heat dissipation
Stratix 2 FPGAs are capable of drawing incredible amounts of current from their 1.2V and 2.5V power supplies. According to Altera online power estimator tool, a fully utilized FPGA running at 300Mhz can draw more than TODO of power. With this much power used in each FPGA, the DN7000K10PCI can dissipate 75 or more Watts of heat. For all but the most trivial designs, a heatsink must be used with the Stratix 2 FPGA. The DN7000K10PCI comes with a forced air heatsink rated at 2 degrees per Watt. Since the maximum operating junction temperature of a Stratix 2 FPGA is 85 degrees, assuming an ambient temperature of 50 degrees (the inside of your computer case) the most amount of energy dissipated by the FPGA using the standard fan is 85 – 30 / 2 = 27.5W. This should be sufficient for most applications. If you intend to operate the Stratix 2 FPGA at very high speeds, or are getting overheating issues with your design, you will need to install a larger heatsink.

Figure 39 The FPGA Temperature monitor circuit. The MAX1617’s I2C bus is connected to the Cypress MCU

Figure 40 Cooling fan power connector. The FAN_AONn signal allow the USB controller to turn on and off the Cooling fans
6 FPGA interconnect

The DN7000K10PCI was designed to maximize the amount of interconnect between the two primary Stratix 2 FPGAs A and B. This interconnect was routed as tightly coupled differential LVDS to provide the best immunity to power supply and crosstalk noise so that your interconnect can operate at the full switching speed of the output buffers. Following Altera recommendations, the interconnect on the DN7000K10PCI was designed to operate at 1Gb/s for every LVDS pair. (TODO Note 1Gb/s operation requires the fasted speed-grade part –12) In order to achieve such breakneck speeds, you will need to operate the busses of signals using a source-synchronous clocking scheme. The interconnect signals on the DN7000K10PCI have been optimized to operate in “lanes” There are 7 lanes between FPGAs A and B, three between B and C and two between FPGAs A and C. Each lane has a differential LVDS source-synchronous clock in each direction.

Figure 41
Clocking incoming data at high speeds required the use of the each input’s delay buffer to align each bit. The incoming clock needs to be adjusted and used to clock the inputs within its lane. This process can be automated by the use of the new Stratix 2 feature IDELAYCTL.

For detailed description of the required user design to achieve 1Gbs operation, see Altera Application note XAPP704, “High Speed SDR LVDS Transceiver”.

Synchronus clocking and single-ended signaling are still possible on the DN7000K10PCI, you are not required to use highspeed serial design techniques. Single ended interconnect is recommended for signaling below 133Mhz. Because of the DN7000K10PCI’s excellent low-skew clocking network, global synchronous clocking should work fine for your interconnect at speeds lower than 300Mhz. The source synchronous clock signals can also be used as single ended or differential interconnect, or to forward clocks from one FPGA to another.

The total interconnect counts between FPGAs

- A to B 304
- B to C 304
- A to C 304

7 Memory interface

There are three standard 200-pin DDR2 SODIMM module sockets on the DN7000K10PCI. These sockets are supplied with 1.8V power and keyed for use with DDR2 SDRAMs. One socket is connected to FPGA A, B and C. For IOs connected to the DDR2 sodimm, use the SSTL18 IO standard.
Figure 42
7.1 Clocking
Clock signals for the SODIMM interface are driven directly from 1.8V differential IO on the Stratix 2 FPGA.

7.2 Serial Presence Detect.
The EEPROM on the SODIMM is accessible by the FPGA.

8 Daughtercard Headers
The daughtercard interface provides 186 signals from a host board to a daughter card for each of the two daughtercard connectors. These signals are high-speed, flexible and differentially capable.

The daughtercard interface is built around a 400-pin FCI “MegArray” connector. This BGA array of pins is designed for high-speed high-density board-to-board connections.

The “Plug” of the system is located on the host, and the “receptacle” is located on the expansion board. This selection was made to give a greater height selection to the daughter card designer.

The user typically designs his own daughter card with interfaces required for the emulation project. The Dini Group is happy to review daughter card designs intended for interface to a Dini Group host for possible compatibility problems. A daughter card designer should use the schematic of the host board (provided with the board) to verify his design. The PCB libraries and schematic libraries of an example daughter card are provided here

http://dinigroup.com/DNMEG_Obs.php

8.1.1 Banking System
To allow flexibility to the daughter card designer, the daughtercard interface is divided into three “Banks”. Each bank can have its own VCCO power, VREF (threshold voltage) and source-synchronous IO clocking. Each bank contains 62 user IO signals (can be used as 31 differential pairs). Eight of these signals can be used as a source-synchronous clock into the host FPGA. Four of these signals can be used as a reference voltage to the host fpga for standards requiring a reference voltage.

On the DN7000K10PCI, banks B0 and B1 are combined into one large bank. These must share VCCO voltages!
The banks are named B0, B1 and B2, and every user IO on the header interface corresponds to one of these banks. The signals name given to each user IO pin contains either “B0”, “B1”, or “B2” in the name.

Other connections on the daughter card connector system (not correlated to a bank) include three dedicated, differential clock connections for inputting global clocks from an external source, power connections, bank VCCO power, a buffered power on reset signal.

8.2 Daughter Card Electrical
The connector itself is capable of as high as 10Gbs transmission rates using differential signaling. All signals on the host are all routed as differential, 50-Ohm transmission lines, with means to properly terminate. All signals are routed against a ground plane, so for the best signal integrity, should be routed against a ground plane on the daughter card with excellent IO voltage bypassing close to the terminus. When signals are used differentially, the trace impedance is 100-ohms.

Signals on the host are not length-matched, except for each signal in a differential pair, which are. Differential pairs are routed in parallel, but not closely coupled to make single-ended signaling possible. Using the IDELAY and ODELAY (Stratix devices adaptive delay) elements on the FPGA, the skew between mismatched signals can be corrected.

Most pins are assigned in a GSG pattern to minimize crosstalk. Some signals are arranged in a GSSG pattern. (Pins in column E and F)

8.2.1 Daughter Card Signals
Bank 0 (VCC0) includes the signals BL[0-31].
Bank 1 (VCC01) includes B1L[0-31].
Bank 2 (VCC02) includes signals B2L[0-31]

On the DN7000K10PCI, banks B0 and B1 are combined into one large bank. These must share VCCO voltages!

Special purpose pins are described below.

8.2.2 Clock Outputs
These signals are used for sending a clock signal (differential) from the daughter card to the host.

The pair GCAP/GCAN (Pairs E1, F1) can be used as a SSTL18, SSTL25, LVDS differential pair, or GCAP can be used single-endedly as LVCMOS25. These signals only connect to the FPGA associated with the connector.
The pair GCBP/GCBN (E3, F3) can be used as a SSTL18, SSTL25, LVDS differential pair, or GCAP can be used single-endedly as LVCMOS25. These signals only connect to the FPGA associated with the connector.

The pair GCCP/GCCN (Pair E5, F5) must be used as an LVDS pair. This differential clock is buffered and distributed to every FPGA on the host. This clock is de-skewed on the host through a zero-delay buffer. There may be special frequency requirements or settings for the buffers to work.

### 8.2.3 User IO

User IO signals connect directly to a general-purpose IO site on the host FPGA. These signals can be used as any drive standard supported by the host FPGA. Each bank must share VCCIO and VREF requirements. For example, if bank B0 is supplied 1.8V by the daughter card on ALL of the VCCIO_B0 pins, and supplies 0.9V to ALL of the VREF_B0 pins, then the daughter card may use each of the user IO pins as 1.8V SSTL inputs or outputs, LVDS outputs, 1.8V LVCMOS inputs or outputs, since all of these signal standards’ requirements are met.

User IO signals can be used single-endedly, or differentially. Differential pairs are pre-selected. Each set of signals whose signal names differ only by a “p” or “n” in the signal name can be used as differential pairs. For example, the signals B2_L4p and B2_L4n can be used as a differential pair. These two signals are matched in length.

### 8.2.4 IO Clock

Some of the signals connected to the daughter card expansion headers are “clock-capable”; the inputs on the host FPGA can be used for source-synchronous clocking. On Virtex-4 devices, these pins have “CC” in the name. The CC pins on each bank are suitable for IO, source-synchronous clocking for all signals on that bank. A CC clock can only be used to clock signals on the bank associated with it.

If source-synchronous clocking is not required, these signals can be used as “User IO”.

### 8.2.5 VREF

Pins declared as “VREF” pins by Xilinx have a defined placement on the daughter card pin out to allow the daughter card to define a logic threshold as required by some standards. If you want to use a standard with a VREF (SSTL15, SSTL15, SSTL18, HSTL15, HSTL18, HSLVCMOS33) the daughter card should supply this reference voltage on these pins. For optimal performance, capacitors should be installed on the host board on these signals near the host’s FPGAs. Space provisions for these capacitors have been provided.

If VREF is not required by the intended signaling standard, then these signals can be used as “User IO” without restriction.
8.2.6 Power
The +3.3V, +5.0V and +12V power rails are supplied to the Daughter card headers. Each pin on the MegArray connector is rated to tolerate 1A of current without thermal overload. Most of the power available to daughter cards through the connector comes from the two 12V pins, for a total of 24W. The host provides a fuse on each of these rails.

8.2.7 IO Power
The signaling standard of the daughter card is left undefined by the host. For a standard to be used, the daughter card should supply power to the VCCIO pins of the daughter card connector. The pins are connected directly to the VCCIO power pin on the host FPGA. Each of the three banks on the daughter card has two VCCIO pins. Each of the three banks can have its VCCIO set independently, but both pins on a bank must be the same voltage. The daughter card should be able to supply enough current to the host FPGA to power the entire bus. The daughter card designer will need to calculate the host FPGA’s current requirements.

On the DN7000K10PCI, banks B0 and B1 are combined into one large bank. These must share VCCO voltages!

When the daughter card does not supply a voltage the host, the host will power these pins at 1.2V with a minimal current capacity. The daughter card can overdrive this voltage safely.

8.2.8 Reset
The reset signal (Active low) is an open-drain, buffered copy of the reset signal on the host. This signal is asserted when the host power is not within tolerance. The signal must be pulled up on the daughter card with a resistor. When the reset signal is active, the FPGAs on the host will not be configured.
The RSTn signal to the daughter card is an open-drain, buffered copy of the SYS_RSTn signal. This signal causes the entire DN8000K10 to reset, losing all FPGA configuration data and resetting the configuration circuitry.

8.3 Pin out
The following lists are the pin assignments to the MEG Array connector. The pins are labeled as in the FCI connector part drawing.

8.3.1 Clock and Reset Signals

<table>
<thead>
<tr>
<th>Pin</th>
<th>Signal</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>GCAP</td>
</tr>
<tr>
<td>E1</td>
<td>GCAN</td>
</tr>
<tr>
<td>E3</td>
<td>GCBP</td>
</tr>
<tr>
<td>F3</td>
<td>GCBN</td>
</tr>
<tr>
<td>E5</td>
<td>GCCP</td>
</tr>
<tr>
<td>F5</td>
<td>GCCN</td>
</tr>
<tr>
<td>J2</td>
<td>RSTn</td>
</tr>
</tbody>
</table>

8.3.2 Power Signals

<table>
<thead>
<tr>
<th>Pin</th>
<th>Voltage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>+12V</td>
</tr>
<tr>
<td>K1</td>
<td>+12V</td>
</tr>
<tr>
<td>C1</td>
<td>+5V</td>
</tr>
<tr>
<td>H1</td>
<td>+5V</td>
</tr>
<tr>
<td>B2</td>
<td>+3.3V</td>
</tr>
<tr>
<td>D2</td>
<td>+3.3V</td>
</tr>
<tr>
<td>G2</td>
<td>+3.3V</td>
</tr>
</tbody>
</table>

8.3.3 Ground Signals

<table>
<thead>
<tr>
<th>Pin</th>
<th>Signal</th>
</tr>
</thead>
<tbody>
<tr>
<td>A10</td>
<td>B3</td>
</tr>
<tr>
<td>A12</td>
<td>B1</td>
</tr>
<tr>
<td>A14</td>
<td>B13</td>
</tr>
<tr>
<td>A16</td>
<td>B11</td>
</tr>
<tr>
<td>A18</td>
<td>B5</td>
</tr>
<tr>
<td>A2</td>
<td>B9</td>
</tr>
<tr>
<td>A22</td>
<td>B7</td>
</tr>
<tr>
<td>A24</td>
<td>B23</td>
</tr>
<tr>
<td>A26</td>
<td>B21</td>
</tr>
<tr>
<td>A28</td>
<td>B15</td>
</tr>
<tr>
<td>A30</td>
<td>B19</td>
</tr>
<tr>
<td>A34</td>
<td>B27</td>
</tr>
</tbody>
</table>
8.3.4 Bank B0 IO
Pins shaded yellow are “CC” pins. These pins are the same for the P10 header.


8.3.5 Bank B1 IO

Pins shaded yellow are “CC” pins. These pins are the same for the P10 header.
<table>
<thead>
<tr>
<th>Letter</th>
<th>Code</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>A19</td>
<td>DC_B1P7</td>
<td>AH17</td>
</tr>
<tr>
<td>B20</td>
<td>DC_B1N7</td>
<td>AJ17</td>
</tr>
<tr>
<td>C19</td>
<td>DC_B1P8</td>
<td>AL15</td>
</tr>
<tr>
<td>D20</td>
<td>DC_B1N8</td>
<td>AM15</td>
</tr>
<tr>
<td>E19</td>
<td>DC_B1P28</td>
<td>AM17</td>
</tr>
<tr>
<td>F19</td>
<td>DC_B1N28</td>
<td>AN17</td>
</tr>
<tr>
<td>H19</td>
<td>DC_B1P9</td>
<td>AT13</td>
</tr>
<tr>
<td>G20</td>
<td>DC_B1N9</td>
<td>AU13</td>
</tr>
<tr>
<td>K19</td>
<td>DC_B1P10</td>
<td>AV13</td>
</tr>
<tr>
<td>J20</td>
<td>DC_B1N10</td>
<td>AW13</td>
</tr>
<tr>
<td>A21</td>
<td>DC_B1P11</td>
<td>AH18</td>
</tr>
<tr>
<td>B22</td>
<td>DC_B1N11</td>
<td>AJ18</td>
</tr>
<tr>
<td>C21</td>
<td>DC_B1P12</td>
<td>AL16</td>
</tr>
<tr>
<td>D22</td>
<td>DC_B1N12</td>
<td>AM16</td>
</tr>
<tr>
<td>E21</td>
<td>DC_B1P29</td>
<td>AN18</td>
</tr>
<tr>
<td>F21</td>
<td>DC_B1N29</td>
<td>AP18</td>
</tr>
<tr>
<td>H21</td>
<td>DC_B1P13</td>
<td>AT14</td>
</tr>
<tr>
<td>G22</td>
<td>DC_B1N13</td>
<td>AU14</td>
</tr>
<tr>
<td>K21</td>
<td>DC_B1P14</td>
<td>AV14</td>
</tr>
<tr>
<td>J22</td>
<td>DC_B1N14</td>
<td>AW14</td>
</tr>
<tr>
<td>A23</td>
<td>DC_B1P15</td>
<td>AG19</td>
</tr>
<tr>
<td>B24</td>
<td>DC_B1N15</td>
<td>AH19</td>
</tr>
<tr>
<td>C23</td>
<td>DC_B1P16</td>
<td>AK17</td>
</tr>
<tr>
<td>D24</td>
<td>DC_B1N16</td>
<td>AL17</td>
</tr>
<tr>
<td>E23</td>
<td>DC_B1P30</td>
<td>AP19</td>
</tr>
<tr>
<td>F23</td>
<td>DC_B1N30</td>
<td>AR20</td>
</tr>
<tr>
<td>H23</td>
<td>DC_B1P17</td>
<td>AT15</td>
</tr>
<tr>
<td>G24</td>
<td>DC_B1N17</td>
<td>AU15</td>
</tr>
<tr>
<td>K23</td>
<td>DC_B1P18</td>
<td>AV15</td>
</tr>
<tr>
<td>J24</td>
<td>DC_B1N18</td>
<td>AW15</td>
</tr>
<tr>
<td>A25</td>
<td>DC_B1P19</td>
<td>AN19</td>
</tr>
<tr>
<td>B26</td>
<td>DC_B1N19</td>
<td>AM19</td>
</tr>
<tr>
<td>C25</td>
<td>DC_B1P20</td>
<td>AL18</td>
</tr>
<tr>
<td>D26</td>
<td>DC_B1N20</td>
<td>AM18</td>
</tr>
<tr>
<td>E25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H25</td>
<td>DC_B1P21</td>
<td>AR16</td>
</tr>
<tr>
<td>G26</td>
<td>DC_B1N21</td>
<td>AT16</td>
</tr>
<tr>
<td>K25</td>
<td>DC_B1P22</td>
<td>AU16</td>
</tr>
<tr>
<td>J26</td>
<td>DC_B1N22</td>
<td>AV16</td>
</tr>
<tr>
<td>A27</td>
<td>DC_B1P23</td>
<td>AG20</td>
</tr>
<tr>
<td>B28</td>
<td>DC_B1N23</td>
<td>AH20</td>
</tr>
<tr>
<td>C27</td>
<td>DC_B1P24</td>
<td>AU17</td>
</tr>
<tr>
<td>D28</td>
<td>DC_B1N24</td>
<td>AT17</td>
</tr>
<tr>
<td>H27</td>
<td>DC_B1P25</td>
<td>AV17</td>
</tr>
<tr>
<td>G28</td>
<td>DC_B1N25</td>
<td>AW17</td>
</tr>
<tr>
<td>K27</td>
<td>DC_B1P26</td>
<td>AN15</td>
</tr>
</tbody>
</table>
8.3.6 Bank B2 IO

Pins shaded yellow are “CC” pins. These pins are the same for the P10 header.

<table>
<thead>
<tr>
<th>Pin</th>
<th>Description</th>
<th>Slot</th>
</tr>
</thead>
<tbody>
<tr>
<td>E27</td>
<td>DC_B2P25</td>
<td>AV23</td>
</tr>
<tr>
<td>F27</td>
<td>DC_B2N25</td>
<td>AW23</td>
</tr>
<tr>
<td>A29</td>
<td>DC_B2P1</td>
<td>AN24</td>
</tr>
<tr>
<td>B30</td>
<td>DC_B2N1</td>
<td>AP24</td>
</tr>
<tr>
<td>C29</td>
<td>DC_B2P2</td>
<td>AP22</td>
</tr>
<tr>
<td>D30</td>
<td>DC_B2N2</td>
<td>AP23</td>
</tr>
<tr>
<td>E29</td>
<td>DC_B2P26</td>
<td>AU24</td>
</tr>
<tr>
<td>F29</td>
<td>DC_B2N26</td>
<td>AV24</td>
</tr>
<tr>
<td>H29</td>
<td>DC_B2P3</td>
<td>AJ23</td>
</tr>
<tr>
<td>G30</td>
<td>DC_B2N3</td>
<td>AK23</td>
</tr>
<tr>
<td>K29</td>
<td>DC_B2P4</td>
<td>AL24</td>
</tr>
<tr>
<td>J30</td>
<td>DC_B2N4</td>
<td>AM24</td>
</tr>
<tr>
<td>A31</td>
<td>DC_B2P5</td>
<td>AN25</td>
</tr>
<tr>
<td>B32</td>
<td>DC_B2N5</td>
<td>AR25</td>
</tr>
<tr>
<td>C31</td>
<td>DC_B2P6</td>
<td>AU23</td>
</tr>
<tr>
<td>D32</td>
<td>DC_B2N6</td>
<td>AT24</td>
</tr>
<tr>
<td>E31</td>
<td>DC_B2P27</td>
<td>AV25</td>
</tr>
<tr>
<td>F31</td>
<td>DC_B2N27</td>
<td>AW25</td>
</tr>
<tr>
<td>H31</td>
<td>DC_B2P7</td>
<td>AJ24</td>
</tr>
<tr>
<td>G32</td>
<td>DC_B2N7</td>
<td>AK24</td>
</tr>
<tr>
<td>K31</td>
<td>DC_B2P8</td>
<td>AL25</td>
</tr>
<tr>
<td>J32</td>
<td>DC_B2N8</td>
<td>AM25</td>
</tr>
<tr>
<td>A33</td>
<td>DC_B2P9</td>
<td>AP26</td>
</tr>
<tr>
<td>B34</td>
<td>DC_B2N9</td>
<td>AR26</td>
</tr>
<tr>
<td>C33</td>
<td>DC_B2P10</td>
<td>AT25</td>
</tr>
<tr>
<td>D34</td>
<td>DC_B2N10</td>
<td>AU25</td>
</tr>
<tr>
<td>E33</td>
<td>DC_B2P28</td>
<td>AV26</td>
</tr>
<tr>
<td>F33</td>
<td>DC_B2N28</td>
<td>AW26</td>
</tr>
<tr>
<td>H33</td>
<td>DC_B2P11</td>
<td>AJ25</td>
</tr>
<tr>
<td>G34</td>
<td>DC_B2N11</td>
<td>AK25</td>
</tr>
<tr>
<td>K33</td>
<td>DC_B2P12</td>
<td>AM26</td>
</tr>
<tr>
<td>J34</td>
<td>DC_B2N12</td>
<td>AN26</td>
</tr>
<tr>
<td>A35</td>
<td>DC_B2P13</td>
<td>AP27</td>
</tr>
<tr>
<td>B36</td>
<td>DC_B2N13</td>
<td>AR27</td>
</tr>
<tr>
<td>C35</td>
<td>DC_B2P14</td>
<td>AT26</td>
</tr>
<tr>
<td>D36</td>
<td>DC_B2N14</td>
<td>AU26</td>
</tr>
<tr>
<td>E35</td>
<td>DC_B2P29</td>
<td>AV27</td>
</tr>
<tr>
<td>F35</td>
<td>DC_B2N29</td>
<td>AW27</td>
</tr>
<tr>
<td>H35</td>
<td>DC_B2P15</td>
<td>AK26</td>
</tr>
<tr>
<td>G36</td>
<td>DC_B2N15</td>
<td>AL26</td>
</tr>
<tr>
<td>K35</td>
<td>DC_B2P16</td>
<td>AM27</td>
</tr>
</tbody>
</table>
8.3.7 VCCIO Signals

The daughtercard attached to the DN7000K10PCI is expected to drive a voltage onto these pins. This voltage is used by the FPGAs on the DN7000K10PCI as IO power. The daughtercard should allow enough current consumption through these pins for one bank on the FPGA to operate. Typically this is about 1 Amp.

VCCO corresponds to bank B0, VCC1 corresponds to bank B1, VCC2 corresponds to bank B2.

On the DN7000K10PCI, banks B0 and B1 are combined into one large bank. These must share VCCO voltages!
### 8.4 VCCO bias generation

Since a daughter card will not always be present on a daughter card connector, a VCCO bias generator is used on the motherboard for each daughter card bank to keep the VCCO pin on the FPGA within its recommended operating range. The VCCO bias generators supply +1.2V to the VCCO pins on the FPGAs, and are back-biased by the daughter card when it drives the VCCO rails.

The daughter card should over-drive this power supply with the voltage it requires for IO on each bank. The bias-generating regulator will shut off and not interfere with the daughter card's impressed voltage.

The VCCO voltage impressed by the daughter card should be less than 3.75 to prevent destruction of the Virtex 4 IOs connected to that daughter card.

![Diagram of VCCO bias generation](image)

**Figure 44**

### 8.5 Daughter card Mechanical

Daughter card expansion headers are located on the bottom side of the PWB. This is done to eliminate the need for resolving board-to-board clearance issues, assuming the daughter card uses no large components on the backside.
Each host with a daughter card interface makes certain minimal provisions for daughter cards. Enough space is reserved for each daughter card plug to accommodate the following hypothetical daughter card. (The DNMEG_OBS400 conforms to these dimensions)

Note that the components on the topside of the daughter card and DN8000K10 face in opposite directions.
At least four mounting positions are provided for each header in a standard location, as shown above.

Boards that have multiple daughter card connectors next to each other (horizontally) use a standard spacing of 78.25mm (3.08in) from pin A1 to pin A1. (and aligned vertically)

The connectors used in the expansion system on the host are FCI MEG-Array 400-pin plug, 6mm, part #84520-102. A suitable mating connector for use on a daughter card would be FCI part 74390-101 (lead-free 74390-101LF). This provides the minimum board-to-board spacing. Other spacings are possible with different connectors on the daughter card.
HARDWARE

8.5.1 Insertion and removal
Due to the small dimensions of the very high speed MegArray connector system, the pins on the plug and receptacle of the Meg Array connectors are very delicate.

When plugging in a daughter card, make sure to align the daughter card first before pressing on the connector. Be absolutely certain that both the small and the large keys at the narrow ends of the Meg Array line up BEFORE applying pressure to mate the connectors!

Figure 47

Place it down flat, then press down gently.

Figure 48
The following two excerpts are taken from the FCI application guide for the Meg Array series of connectors.

A part can be started from either end. Locate and match the connector's A1 position marking ("?") for both the Plug and Receptacle. (Markings are located on the long side of the housing.) Rough alignment is required prior to connector mating as misalignment of >0.8mm could damage connector contacts. Rough alignment of the connector is achieved through matching the Small alignment slot of the plug housing with the Small alignment key of the receptacle housing and the Large alignment slot with the Large alignment key. Both connector housings have generous lead-in around the perimeter and will allow the user to blind mate assemble the connectors. Align the two connectors by feel and when the receptacle keys start into the plug slots, push down on one end and then move force forward until the receptacle cover flange bottoms on the front face of the plug.

Dec 09, 2004

Like mating, a connector pair can be unmated by pulling them straight apart. However, it requires less effort to un-mate if the force is originated from one of the slot/key ends of the assembly. (Reverse procedure from mating) Mating or un-mating of the connector by rolling in a direction perpendicular to alignment slots/keyes may cause damage to the terminal contacts and is not recommended.

8.6 Standard Daughter cards
The Dini Group has some general-purpose daughter cards available. See the Dini Group website for more information.

8.6.1 DNMEG_OBS400
The breakout daughter card provides 48 signals on Mictor connector, 124 signals on .1” pitch headers arranged for differential signaling, 14 differential coax cable connection (intended for use with the DN8000K10 rocketIO headers) and global clock inputs.
8.6.2  **DNMEG_OBS300**
The DNMEG_OBS300 is identical to the DNMEG_OBS400 except that the receptacle is a 300-pin instead of a 400-pin connector. This can only be used with DC0, DC1, DC2, DC3, DC4, DC9 on the DN8000K10.

8.6.3  **DNMEG_S2GX**
This board provides a Stratix 2 GX FPGA, capable of 6.5Gbs serial data transmission. And some memory options.

The DNMEG_S2GX also has a daughter card header configured as a pass-through from its own daughter card receptacle.

8.6.4  **DNMEG_ADC**
This board provides a high-speed ADC and DAC, 1Gb Ethernet, Virtex-4 SX55 FPGA, and Memory options.

8.6.5  **DNMEG_DVI**
This board provides dual DVI-D input and output and memory options.

8.6.6  **DNMEG_ARM (coming soon)**
Arm processor
9 RS232

Each FPGA is connected to an RS232 header. The header is designed to be used with the provided RS232 cables. They can be connected to a computer through the serial port.

The RS232 signals connected to the FPGA are buffered through a 12V RS232 level-translator. The IOs for use with RS232 are:

- G22 RX
- F22 TX

The headers are located next to each FPGA.
### 10 LEDs

#### Config Status LEDs

<table>
<thead>
<tr>
<th>DS1</th>
<th>CFPGA_LEDn0</th>
<th>RED</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFPGA_LEDn1</td>
<td>RED</td>
<td></td>
</tr>
<tr>
<td>CFPGA_LEDn2</td>
<td>RED</td>
<td></td>
</tr>
<tr>
<td>CFPGA_LEDn3</td>
<td>RED</td>
<td></td>
</tr>
</tbody>
</table>

FPGA configuring activity

Main Bus over PCI activity

Configuring using PCI activity

<table>
<thead>
<tr>
<th>DS2</th>
<th>MCU_LEDn0</th>
<th>RED</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCU_LEDn1</td>
<td>RED</td>
<td></td>
</tr>
<tr>
<td>MCU_LEDn2</td>
<td>RED</td>
<td></td>
</tr>
<tr>
<td>MCU_LEDn3</td>
<td>RED</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DS14</th>
<th>QL_LED_QLOUT</th>
<th>GREEN</th>
</tr>
</thead>
</table>

PCI output activity

<table>
<thead>
<tr>
<th>DS15</th>
<th>QL_LED_QLIN</th>
<th>GREEN</th>
</tr>
</thead>
</table>

PCI input activity

<table>
<thead>
<tr>
<th>DS11</th>
<th>CFPGA_DONE</th>
<th>BLUE</th>
</tr>
</thead>
</table>

Configuration FPGA is working

<table>
<thead>
<tr>
<th>DS16</th>
<th>FPGA_CFDONE_A</th>
<th>BLUE</th>
</tr>
</thead>
</table>

FPGA A is configured

<table>
<thead>
<tr>
<th>DS17</th>
<th>FPGA_CFDONE_B</th>
<th>BLUE</th>
</tr>
</thead>
</table>

FPGA B is configured

<table>
<thead>
<tr>
<th>DS26</th>
<th>FPGA_CFDONE_C</th>
<th>BLUE</th>
</tr>
</thead>
</table>

FPGA C is configured

#### Power status LEDs

<table>
<thead>
<tr>
<th>DS3</th>
<th>-12V</th>
<th>RED</th>
<th>-12V power is on</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS4</td>
<td>+12V</td>
<td>RED</td>
<td>+12V power is on</td>
</tr>
<tr>
<td>DS5</td>
<td>+5V</td>
<td>RED</td>
<td>+5V power is on</td>
</tr>
<tr>
<td>DS6</td>
<td>+3.3V</td>
<td>RED</td>
<td>+3.3V power is on</td>
</tr>
<tr>
<td>DS7</td>
<td>+2.5V</td>
<td>RED</td>
<td>+2.5V power is on</td>
</tr>
<tr>
<td>DS13</td>
<td>+1.2VA</td>
<td>RED</td>
<td>+1.2V power for FPGA A is on</td>
</tr>
<tr>
<td>DS21</td>
<td>+1.2VB</td>
<td>RED</td>
<td>+1.2V power for FPGA B is on</td>
</tr>
<tr>
<td>DS25</td>
<td>+1.2VC</td>
<td>RED</td>
<td>+1.2V power for FPGA C is on</td>
</tr>
<tr>
<td>DS8</td>
<td>PWR_OK</td>
<td>RED</td>
<td>+1.2V ATX power supply is on</td>
</tr>
</tbody>
</table>

#### User LEDs

<table>
<thead>
<tr>
<th>DS12</th>
<th>FPGA_LED0_A</th>
<th>GREEN</th>
<th>User controlled FPGA pin G21</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS9</td>
<td>FPGA_LED2_A</td>
<td>GREEN</td>
<td>User controlled FPGA pin F20</td>
</tr>
<tr>
<td>DS10</td>
<td>FPGA_LED1_A</td>
<td>GREEN</td>
<td>User controlled FPGA pin E20</td>
</tr>
<tr>
<td>DS20</td>
<td>FPGA_LED0_B</td>
<td>GREEN</td>
<td>User controlled FPGA pin G21</td>
</tr>
<tr>
<td>DS19</td>
<td>FPGA_LED1_B</td>
<td>GREEN</td>
<td>User controlled FPGA pin F20</td>
</tr>
<tr>
<td>DS18</td>
<td>FPGA_LED2_B</td>
<td>GREEN</td>
<td>User controlled FPGA pin E20</td>
</tr>
</tbody>
</table>
HARDWARE

DS24 FPGA_LED0_C GREEN User controlled FPGA pin G21
DS23 FPGA_LED1_C GREEN User controlled FPGA pin F20
DS22 FPGA_LED2_C GREEN User controlled FPGA pin E20

11 PCI interface

11.1 PCI edge connector

The edge connector of REVISION 1: Revision 1 of the DN7000K10PCI can only be operated in a 3.3V PCI slot.

![Figure 52]

Do not attempt to plug a PCI card in backwards if it does not fit.

11.2 The Quicklogic 5064

In order to provide a highspeed easy-to-use interface for your design, the DN7000K10PCI comes equipped with a PCI bridge, a QuickLogic 5064.

To implements a PCI interface, you should use the PCI module provided on the user CD. A document describing this module is also on the user CD, QL5064_Interface_Module.doc. The rest of this section describes to function of the PCI hardware.
11.3 Stratix 2 FPGA Communication
When the board is in a PCI slot, whether or not FPGA A is configured, the device will communicate with the host. This is because the QL5064 is always active. The QL5064 will report a Vendor ID: 17DF, and a Device ID: 1864.

The QL5064 will define BARs 0 through 7.

BARs 1-5 are reserved for user communication with the FPGA.

11.4 Cyclone 2 Communication
The QL5064 will also define BAR 0. This range is reserved for the Dini Group reference design, configuration, and testing purposes. The only reason you should use BAR 0 is to communicate over the Main Bus interface.
The above diagram shows the signal connections between the Quicklogic, Cyclone and Stratix FPGA. Data from BAR 1-5 is transferred from the QL5064 to the Stratix FPGA over the 64-bit QLDATA_IN and QLDATA_OUT busses.

Data on BAR 0 is transferred from the QL5064 to the Cyclone FPGA over the lower 32 bits (QLDATA_IN0-31, QLDATA_OUT0-31).

To access the “MainBus” interface, follow these steps:

1) Write the 32-bit (4 byte) “MainBus” address you wish to access to BAR 0, offset 0x240. This sets the Cyclone’s main bus address register. This also causes a “main bus address latch” to occur over main bus.

2) To read from main bus at this address read 4 bytes from BAR 0, offset 0x250. This will cause the Cyclone to read from main bus and return the 32-bit result. After this is done, the Cyclone will increment the address in its address register, so consecutive addresses can be read by reading from this offset multiple times. It will then cause a “main bus address latch” to occur.

3) To write to an address on main bus, write 4 bytes to BAR 0, offset, 0x248. After a write is completed, the Address register increments by 1., then, a “main bus address latch” will occur.

11.5 PCI clocking
All communication to the Quicklogic 5064 chip is synchronized with a 75Mhz oscillator. The 75Mhz PCI UCLK is delivered to FPGA A, the Cyclone 2 FPGA and the Quicklogic 5064.
11.6 JTAG
The PCI connector’s JTAG signals are looped back to bypass the DN7000K10PCI when it is plugged into the PCI slot.

11.7 PCI Power
In most applications, the DN7000K10PCI can draw its power from the PCI slot. The PCI specification requires that the motherboard provide 25W of 5V power for the DN7000K10PCI to use (Most motherboards provide well in excess of this amount, by supplying the power for PCI cards directly from the ATX power supply).

You should not plug an external power supply into the power connector, J3, while the DN7000K10PCI is plugged into a PCI slot. This will cause large currents between the power supplies of the motherboard and the aux power supply.

11.8 PCI Signaling
The DN7000K10PCI is keyed to work in 3.3V motherboards.

To allow universal (3.3V or 5.0V) PCI IO, the DN7000K10PCI uses the PCI bus’s VIO pins to detect the IO levels used by your motherboard. Most motherboards use 5V signal levels on the PCI bus, but many servers, and all 64bit PCI slots require 3.3V signaling levels.

The DN7000K10PCI can be used in a PCI or PCI-X slot operating at 33Mhz or 66Mhz. It can also be used in 100Mhz and 133Mhz busses, although the DN7000K10PCI will cause the entire bus to operate at 66Mhz.

12 Mechanical
The dimensions of the PWB are 312mm long by 135mm tall, plus a 8.25mm PCI edge connector. This is taller than the PCI specification allows, although the DN7000K10PCI fits easily inside most ATX computer cases.
The topside clearance with the factory installed active heatsinks is 23mm. This leaves just enough room for airflow if the adjacent PCI slot is left unoccupied, or the DN7000K10PCI is the last PCI card in the row. The factory-installed fans can be removed if you do not require high-power operation, allowing the DN7000K10PCI to meet the PCI height restriction. You should leave the heatsinks in place. The back (solder) side clearance is 3.5mm. This exceeds the PCI specification by 1.5mm.

If it is required that the DN7000K10PCI use only one PCI slot, the fan can be removed from the active heatsink assembly, as long as sufficient airflow is provided. Most PC cases do not provide sufficient airflow for high-power applications. You can leave the PC case opened to decrease the effective ambient temperature. It may not be possible to operate the DN7000K10PCI at full power while meeting the PCI physical dimensions.

The board plugs into any PCI or PCI-X* slot with 5V or 3.3V keying, 32-bit or 64-bit slot widths. 33Mhz or 66Mhz. (*100 and 133Mhz busses will automatically negotiate down to 66Mhz when a DN7000K10PCI is connected.
Chapter 5: Introduction to the Reference Design

This chapter introduces the DN7000K10PCI Reference Design, including information on what the reference design does, how to build it from the source files, and how to modify it for another application.

1 Exploring the Reference Design

1.1 What is the Reference Design?
The reference design is a fully functional Stratix 2 FPGA design capable of demonstrating most of the features available on the DN7000K10PCI. Features exercised in the reference design include:

- Access to the DDR2 SDRAM Modules At 200Mhz
- UART Communication
- FPGA Interconnect
- Interaction with the Configuration FPGA and MCU
- Access to external LEDs
- Daughter Card Header example.
- USB memory map to DDR2 memory.
- PCI memory map to DDR2 memory.
- Pin-multiplexed FPGA interconnect using LVDS at 600Mbs per signal pair

All source code for the reference design is included on the CD and may be used freely in customer development. Precompiled rbf files for the most common stuffing...
options are also included and can be used to verify board functionality before beginning development. A build utility, described in the section Compiling The Reference Design, can be used to generate new rbf files, or to generate rbf files for less common configurations of the DN7000K10PCI.

The reference design was created using

Here are the default main.txt file lines.

```
verbose level: 2
sanity check: y
clock frequency: A 100 MHz // Used only for LVDS_Intercon test
clock frequency: B 200 MHz // Used in MainRef for DDRII clock
clock frequency: C 100 MHz // Checked in MainRef for valid operation

fpga A: A_180.rbf
fpga B: B_180.rbf
fpga C: C_180.rbf
```

## 2 Reference Design Memory Map

The Dini Group reference design memory maps the main features of the DN7000K10PCI to the host interfaces: USB, PCI and RS232.

### 2.1 Main Bus interface

The Dini Group reference design memory maps the main features of the DN8000K10 to the USB interfaces.

The Main Bus interface is used to access the reference design memory map. Addresses are 32-bits. Each address contains a 32-bit word. The first 4 bits in the Address is used by the Dini Group reference design to distinguish FPGAs.

<table>
<thead>
<tr>
<th>FPGA</th>
<th>Address</th>
<th>Register Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPGA A</td>
<td>0x08000000</td>
<td>IDCODE</td>
<td>returns 0x05000117</td>
</tr>
<tr>
<td>FPGA A</td>
<td>0x08000001</td>
<td>DDR2HIADDR</td>
<td></td>
</tr>
<tr>
<td>FPGA A</td>
<td>0x08000002</td>
<td>SCRATCH</td>
<td>scratch space for test</td>
</tr>
<tr>
<td>FPGA A</td>
<td>0x08000004</td>
<td>INTERCONTYPE</td>
<td></td>
</tr>
<tr>
<td>FPGA A</td>
<td>0x08000005</td>
<td>DDR2SIZE</td>
<td>size of the ddr2</td>
</tr>
</tbody>
</table>
FPGA A 0x08000010 PCI_FEEDBACK
FPGA A 0x08000011 STARTBITS
FPGA A 0x08000012 CLK_DC_FAIL    Reserved for daughercard test
FPGA A 0x08000013 DC_0P_FAIL    Reserved for daughercard test
FPGA A 0x08000014 DC_0N_FAIL    Reserved for daughercard test
FPGA A 0x08000015 DC_1P_FAIL    Reserved for daughercard test
FPGA A 0x08000016 DC_1N_FAIL    Reserved for daughercard test
FPGA A 0x08000017 DC_2P_FAIL    Reserved for daughercard test
FPGA A 0x08000018 DC_2N_FAIL    Reserved for daughercard test
FPGA A 0x08000020 MBCLK_CNT    Clock counter for 48Mhz
FPGA A 0x08000021 ACLK_CNT    Clock counter for ACLK
FPGA A 0x08000022 BCLK_CNT    Clock counter for BCLK
FPGA A 0x08000023 CCLK_CNT    Clock counter for CCLK
FPGA A 0x08000024 CYCLONE_CNT Clock counter for
FPGA A 0x08000025 DCARD_CNT    Clock counter for MEGclk
FPGA A 0x0C000000 ABP0 OUT W; the output state of FPGA
FPGA A 0x0C000004 ABP0 OE    W; The output enable of each
FPGA A 0x0C000008 ABP0 IN    The input state of each
FPGA A 0x0C00000C ABP0 Name    “ABP0” (ascii)
FPGA A 0x0C000010 ABP1 OUT W; ABP1 IO output values
FPGA A 0x0C000014 ABP1 OE W; Output enable of ABP1
FPGA A 0x0C000018 ABP1 IN R; ABP1 input values
FPGA A 0x0C00001C ABP1 Name “ABP1” (ascii)
FPGA A 0x0C000XX0 BUS XX OUT    XX can be 0-21 hex. Output
FPGA A 0x0C000XX4 BUS XX OE    XX can be 0-21 hex. OE
FPGA A 0x0C000XX8 BUS XX IN    XX can be 0-21 hex. The
FPGA A 0x0C000XXC BUS XX Name    The name of the bus XX
FPGA B 0x10000000 DDR2 space… Mapped to DDR2
FPGA B 0x17FFFFFF … …interface
INTRODUCTION TO THE REFERENCE DESIGN

FPGA B 0x18000010 PCI.Feedback
FPGA B 0x18000011 STARTBITS
FPGA B 0x18000012 CLK_DC_FAIL  Reserved for daughercard test
FPGA B 0x18000013 DC_0P_FAIL  Reserved for daughercard test
FPGA B 0x18000014 DC_0N_FAIL  Reserved for daughercard test
FPGA B 0x18000015 DC_1P_FAIL  Reserved for daughercard test
FPGA B 0x18000016 DC_1N_FAIL  Reserved for daughercard test
FPGA B 0x18000017 DC_2P_FAIL  Reserved for daughercard test
FPGA B 0x18000018 DC_2N_FAIL  Reserved for daughercard test
FPGA B 0x18000020 MBCLK_CNT  Clock counter for 48Mhz
FPGA B 0x18000021 ACLK_CNT  Clock counter for ACLK
FPGA B 0x18000022 BCLK_CNT  Clock counter for BCLK
FPGA B 0x18000023 CCLK_CNT  Clock counter for CCLK
FPGA B 0x18000024 CYCLONE_CNT Clock counter for
FPGA B 0x18000025 DCARD_CNT  Clock counter for MEG clk
FPGA B 0x18000026 EXT_CNT  Clock counter for SMA clock

FPGA B 0x1C000XX0 BUS XX OUT  XX can be 0-21 hex. Output status of IOs on bus XX.
FPGA B 0x1C000XX4 BUS XX OE  XX can be 0-21 hex. OE status of IOs
FPGA B 0x1C000XX8 BUS XX IN  XX can be 0-21 hex. The input values
FPGA B 0x1C000XXC BUS XX Name  The name of the bus XX (schematic)

FPGA C 0x20000000 DDR2 C space… Mapped to DDR2 SODIMM…
FPGA C 0x27FFFFFF … interface
FPGA C 0x28000010 PCI.Feedback
FPGA C 0x28000011 STARTBITS
FPGA C 0x28000012 CLK_DC_FAIL  Reserved for daughercard test
FPGA C 0x28000013 DC_0P_FAIL  Reserved for daughercard test
FPGA C 0x28000014 DC_0N_FAIL  Reserved for daughercard test
FPGA C 0x28000015 DC_1P_FAIL  Reserved for daughercard test
FPGA C 0x28000016 DC_1N_FAIL  Reserved for daughercard test
FPGA C 0x28000017 DC_2P_FAIL  Reserved for daughercard test
FPGA C 0x28000018 DC_2N_FAIL  Reserved for daughercard test
FPGA C 0x28000020 MBCLK_CNT  Clock counter for 48Mhz
FPGA C 0x28000021 ACLK_CNT  Clock counter for ACLK
FPGA C 0x28000022 BCLK_CNT  Clock counter for BCLK
FPGA C 0x28000023 CCLK_CNT  Clock counter for CCLK
FPGA C 0x28000024 CYCLONE_CNT Clock counter for
2.2 Using the Reference Design

The reference design is controlled from logic in the Cyclone FPGA. To access data in the FPGAs while running reference design tests, the cyclone logic accesses a memory-mapped data bus. This bus is implemented using the DN7000K10PCI’s main bus signals, connecting all 3 FPGAs and the Cyclone configuration FPGA. As well as being used by the GUI application’s built-in hardware tests, this data interface can be accessed directly using the GUI’s read and write address menu options. Many users eventually implement this interface in their designs for conviniece. The source code for the USB GUI application is provided.

The waveforms for a read and write transaction on the Main Bus interface is shown below. All transactions are initiated by the Cyclone configuration FPGA. USB_CLK is fixed at a free-running 48Mhz. RD, WR, ALE are driven by the Cyclone. DONE, VALID are driven by the FPGA. The bus AD[31:0] is bi-directional.

The following waveform shows a Main Bus read operation. It is initiated by the Cyclone asserting RD.

![Main Bus read timing](image)

The following diagram shows a write operation. It is started by the Cyclone asserting WR.
2.3 Compiling the Reference Design
This section deals with the source code to the Reference Design, which can be found on the CD-ROM. All file references are with respect to the root directory of the Reference Design source code (/source/FPGA). Files that are specific to the DN7000K10PCI design are found in the DN7000K10PCI subdirectory, whereas general application code is found in the common subdirectory.

2.3.1 Altera Quartus
A sample Project Navigator project is located at ‘DN7000K10PCI/implement/fpga.npl’. For information on using Altera Quartus, see the chapter Design Guide in Chapter 3.

3 Getting More Information

3.1 Printed Documentation
The printed documentation, as mentioned previously, takes the form of a Stratix 2 datasheet and a DN7000K10PCI User Guide.

3.2 Electronic Documentation
Multiple documents and datasheets have been included on the CD.

3.3 Online Documentation
There is a public access site that can be found on the Dini Group web site at http://www.dinigroup.com/.

- Probe the clock your design uses. Is it running?
• Check the pinout in your constraint file against the schematic, pinlist, or the provided example constraints file.
• Check the place-and-route report file to make sure that 100% of your IOBs (IO block) used have LOC (location) constraints. There is never a time that a port should not be constrained.
• Use the .PAD report to make sure your constraints were applied correctly. The place and route tools can override constraints in ways you might not expect.
• Double-check that the connections match between your FPGA pins and the daughtercard pins.
• Make sure that none of the other FPGAs are driving MB pins.
• Make sure that the "Unused IOBs" option in the Quartus settings is set to "Float" and not "Pulldown" or “Drive low”
Chapter 6: Ordering Information

1. Part Number - DN7000K10PCI

1 FPGA Options

1.1 FPGA A, B, C

Select an FPGA part to be supplied in the A position. This FPGA is connected to the PCI bus, an expansion header, and can source global clocks. The –12 speed grade is required for full speed operation (1Gbs/pair) of the interconnect between fpgas.

- NONE
- EP2S90-5, -4, -3 *
- EP2S130-5, -4 *
- EP2S180-5, -4

*When and EPS90 or EPS130 is selected for FPGA A, the FPGA A may not use the QL interface to PCI. The “Main Bus” PCI interface can still be used.

*When EP90 is selected for FPGA A, B or C, the number of available differential interconnect between FPGAs is reduced.

2 Optional Equipment

The Dini Group supplies standard daughtercards and memory modules that you can use with the DN7000K10PCI.

- DNMEG_OBS-400 – This standard daughtercard provides 112 FPGA signals to tenth-inch headers, 48 signals to 2 Mictor 38-pin headers.

- SRAM module for use in the 200-pin SODIMM sockets of the DN7000K10PCI. QDRII, 300Mhz 64x2Mb
• SRAM module for use in the 200-pin SODIMM socket. 64x2Mb Standard SDR SRAM. Pipelined or Flowthrough, NoBL available

• RLDRAM module for use in the 200-pin SODIMM socket. 64x16Mb, 300Mhz DDRII

• Flash module for use in the 200-pin SODIMM header.

• Micror module for use in the 200-pin SODIMM header. (2 Micror 38 connectors for use with logic analyzer.

You may also want to obtain from a third party vendor

• 200-pin DDR2 SDRAM SODIMM(s)

• Altera USB Blaster cable