DK2000
High-Performance Demo Kit Motherboard

www.maxim-ic.com

GENERAL DESCRIPTION

The DK2000 is a powerful, flexible platform for evaluating Dallas Semiconductor Telecom ICs and developing related firmware. The ICs are mounted on daughter cards specifically designed to plug into the DK2000's four connectors. The DK2000 provides a Motorola MPC8260 PowerQUICC II communications processor, L2 cache, DRAM, flash memory, various clocks and support logic, and an RS-232 interface to a host PC. As shipped from the factory, the processor runs general-purpose firmware that executes reads and writes to the daughter cards on behalf of PC-based demo software.

DEMO KIT CONTENTS

DK2000 Board
DK2000 Power Supply
One Daughter Card
CD-ROM
- ChipView Demo Software
- DK2000 Data Sheet
- DK2000 Schematics
- Configuration Files
- Definition Files
- Initialization Files

FEATURES

- Interfaces with Up to Four Daughter Cards Simultaneously
- Connects PC-Based Demo Software to the Telecom ICs Under Evaluation
- Provides Point-and-Click Access to All Telecom IC Registers and Features
- Demo Software User Interface can be Customized with Simple Text Edits
- Supports Development of Telecom Firmware Before Target Board Design is Complete
- 64MB of DRAM, 4MB of Flash Memory
- Supports 5V, 3.3V, and 2.5V Telecom ICs
- Hardware Support for Daughter Card-to-Daughter Card and Daughter Card-to-Processor TDM Data Streams
- Hardware Support for UTOPIA II Interface
- Side TIM Connector Provides Access to the PowerPC 60x Bus for High-Performance Applications
- Provides Several Connectors for In-System Programming of Board Components

ORDERING INFORMATION

PART	DESCRIPTION
DSDK2000	Motherboard

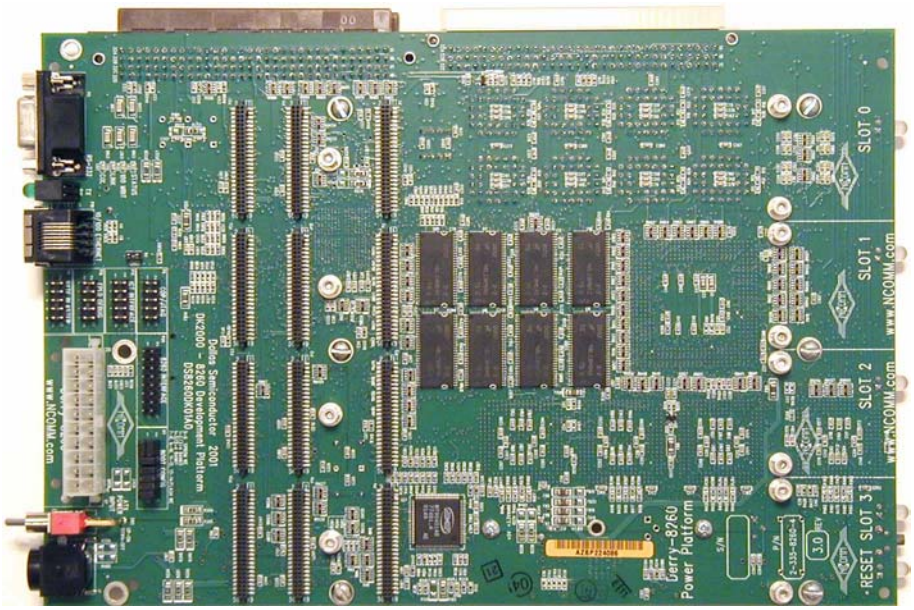


TABLE OF CONTENTS

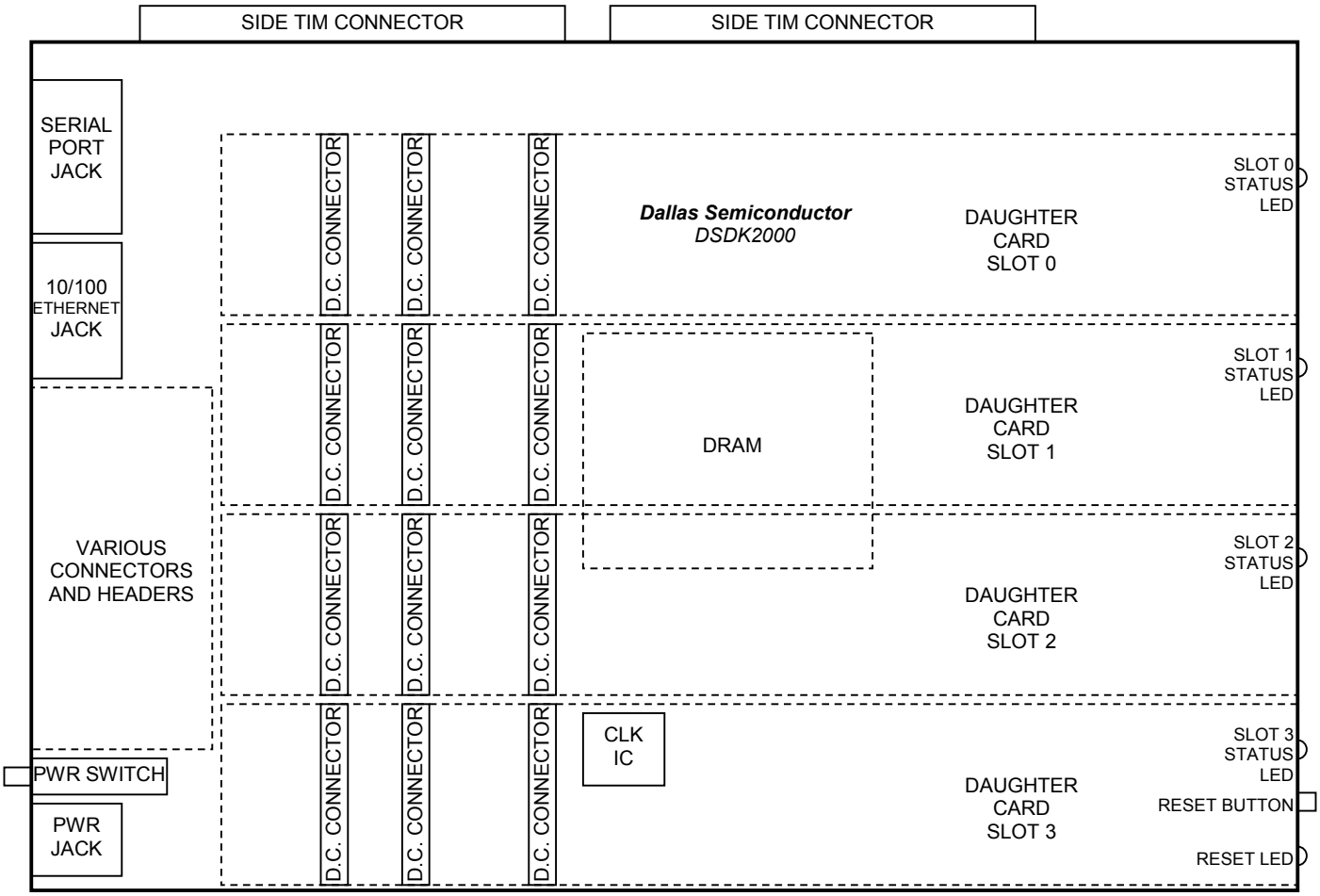
LIST OF FIGURES

LIST OF TABLES

BOARD FLOORPLAN

Figure 1 shows the floorplan of the top of the DK2000 board. Daughter cards attach to the daughter card connectors in the left-center of the board. The power-supply jack and power switch are in the lower-left corner of the board, while the serial port jack is in the upper-left corner. Each daughter card slot has a status LED on the right edge of the board. The board reset button and the reset status LED are situated in the lower-right corner. Most components—including the processor, L2 cache, flash memory, and programmable logic—are located on the bottom of the board.

Figure 1. Board Floorplan



INTRODUCTION

This document is divided into two main sections: *Basic Operation* and *Advanced Features*.

The *Basic Operation* section discusses how to:

- Set up the hardware and connect to a PC
- Install and run the ChipView demo software
- Use ChipView's Register View and Demo modes to interact with the daughter card hardware
- Select and use the definition and configuration files provided with the DK2000 and the daughter cards

The *Advanced Features* section discusses how to:

- Create and edit register definition (.DEF) files
- Create and edit register initialization (.INI) files
- Use Terminal Mode

In addition to these main sections, the *Appendix* provides hardware-related details that supplement the schematic. Only users with complex evaluation/development requirements need the information in the *Advanced Features* section and the *Appendix*.

BASIC OPERATION

Hardware Configuration

Connecting a Daughter Card. Plug the daughter card into one of the DK2000's connectors. The daughter card should be oriented as shown in Figure 1. Note that some daughter cards have two connectors while others have three. The third connector, which is optional, is for advanced features (UTOPIA bus, POS-PHY bus, etc). The DK2000 is compatible with both two-connector and three-connector daughter cards, and supports the advanced features available on the third connector. Note that daughter cards are not designed for hot insertion. Only connect daughter cards to the DK2000 platform with the power off.

Power Supply Connections. The DK2000 operates from an external power supply brick. Plug the brick into an AC power outlet and into the DK2000's power supply jack. The on/off switch for the board is next to the power supply jack. Just after power is applied to the board, the RESET LED glows green and the slot 1 LED glows red. A few seconds later, after the boot routine completes, slot LEDs turn green for slots where daughter cards are recognized and red where daughter cards are not present.

Connecting to a Computer. Connect a standard DB-9 serial cable between the serial port on the DK2000 and an available serial port on the host computer. The host computer must be a Windows®-based PC. Be sure the cable is a standard straight-through cable rather than a null-modem cable. Null-modem cables prevent proper operation.

Installing the ChipView Software

To install the demo software on the host PC, run SETUP.EXE on the demo kit CD-ROM (or from the .ZIP file downloaded from our website, www.maxim-ic.com/telecom). Follow the instructions given by the SETUP program. By default, SETUP installs the application software in "C:\Program Files\ChipView" and creates a shortcut in the ChipView program group.

Running the ChipView Software

Run the ChipView application. If the default installation options were used, click the Start button on the Windows toolbar and select Programs→ChipView→ChipView. The main menu window provides three options: Register View, Demo, and Terminal Mode. Register View mode and Demo mode are discussed in the following paragraphs. Terminal mode is discussed in the *Advanced Features* section.

Register View Mode

Register View provides an intuitive user interface for reading, writing, and viewing the IC registers on the daughter card. Register bytes and bits are displayed by name in an on-screen array. Values can be read or written with a click of the mouse. Figure 2 shows an example of the Register View window.

To go to Register View from the ChipView main menu window, follow these steps:

1) Push the Register View button in the main menu window. A popup window for COM port selection appears next. Select the appropriate port from the menu and click OK. Next, the Definition File Assignment window appears. This window has subwindows to select definition files for each of the four daughter card slots on the DK2000 board.

2) For each slot with a daughter card installed, select a definition file from the list shown, or browse to find a file in another directory. Typically, definition file names contain the device name, e.g., DS2155.def. Some daughter cards ship with multiple definition files. Refer to the daughter card data sheet for detailed information on the use of the various files.

3) Press the Continue button.

Windows is a registered trademark of Microsoft Corp.

The main part of the Register View window displays the register map. To select a register, click on it in the register map. When a register is selected, the full name of the register and its bit map are displayed at the bottom of the Register View window. Bits that are logic 0 are displayed in white, while bits that are logic 1 are displayed in green.

The Register View interface supports the following actions:

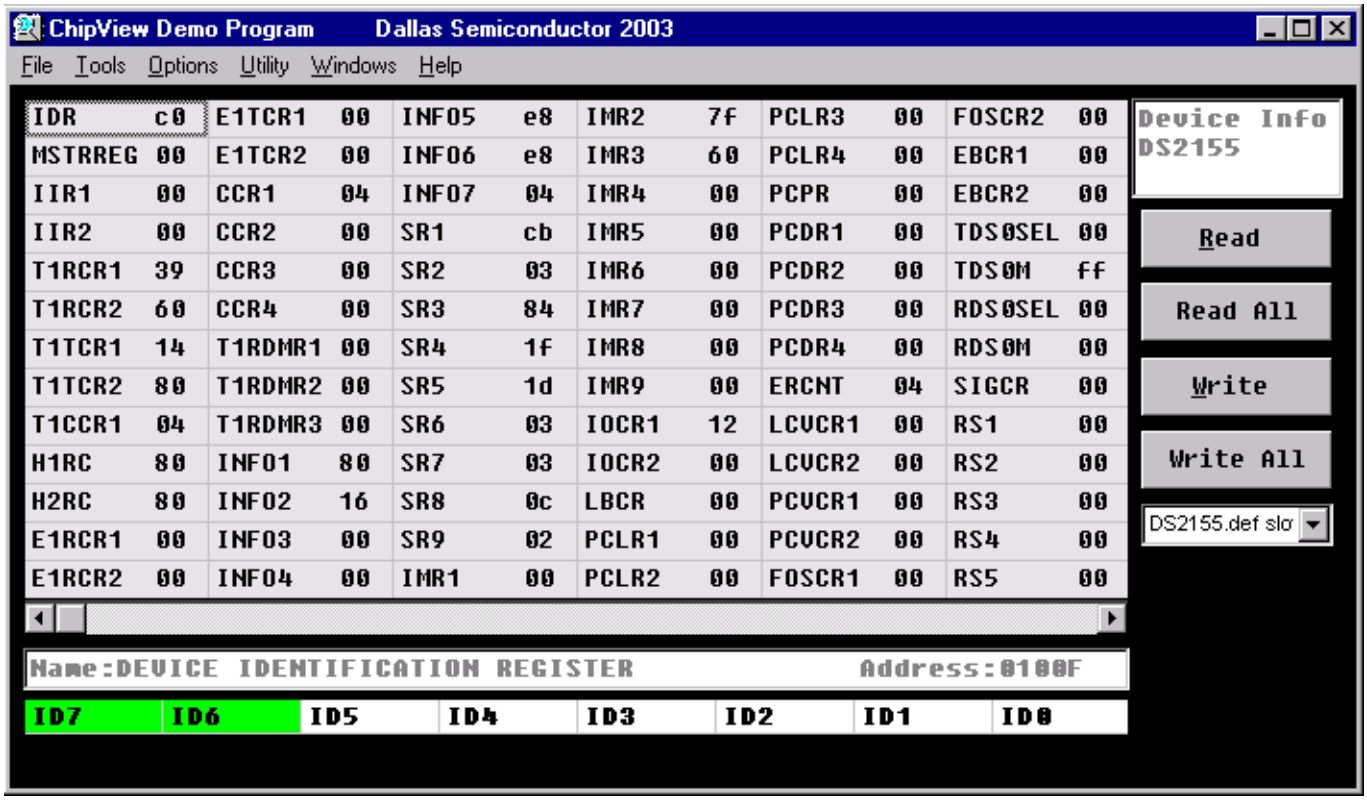
- **Toggle a bit.** Select the register in the register map and then click the bit in the bit map.
- **Write a register.** Select the register, click the Write button, and enter the value to be written.
- **Write all registers.** Click the Write All button and enter the value to be written.
- **Read a register.** Select the register in the register map and click the Read button.
- **Read all registers.** Click the Read All button.

When the Read or Read All buttons are selected, registers whose values have changed since last read are highlighted in green. This highlighting can be disabled by unchecking the Options→Highlight Changed Registers menu selection.

When multiple definition files are loaded at the same time, use the pulldown menu below the command buttons to switch between definition file views. Multiple definition files can be loaded at the same time for each daughter card slot to control different portions of the daughter card hardware. To load an additional definition file, select File→Definition File. In the Definition File Assignment window, select the appropriate file and press the Continue button. The Register View window now shows the view defined by the new definition file. See the *Advanced Features* section below for information about creating and editing definition files.

To go to Register View from other views, select Windows→Go to Register View.

Figure 2. Register View Window



Demo Mode

Demo mode provides an intuitive user interface for configuring daughter cards at a high level using option buttons and menu selections. Key status information is also displayed, such as LOS, OOF, and AIS. Figure 3 shows an example of the Demo window.

To go to the Demo window from the ChipView main menu window, follow these steps:

1) Push the Demo button in the main menu window. A popup window for COM port selection appears next. Select the appropriate port from the menu and click OK. Next, the Configuration File Assignment window appears. This window has subwindows to select configuration files for each of the four daughter cards slots on the DK2000 board.

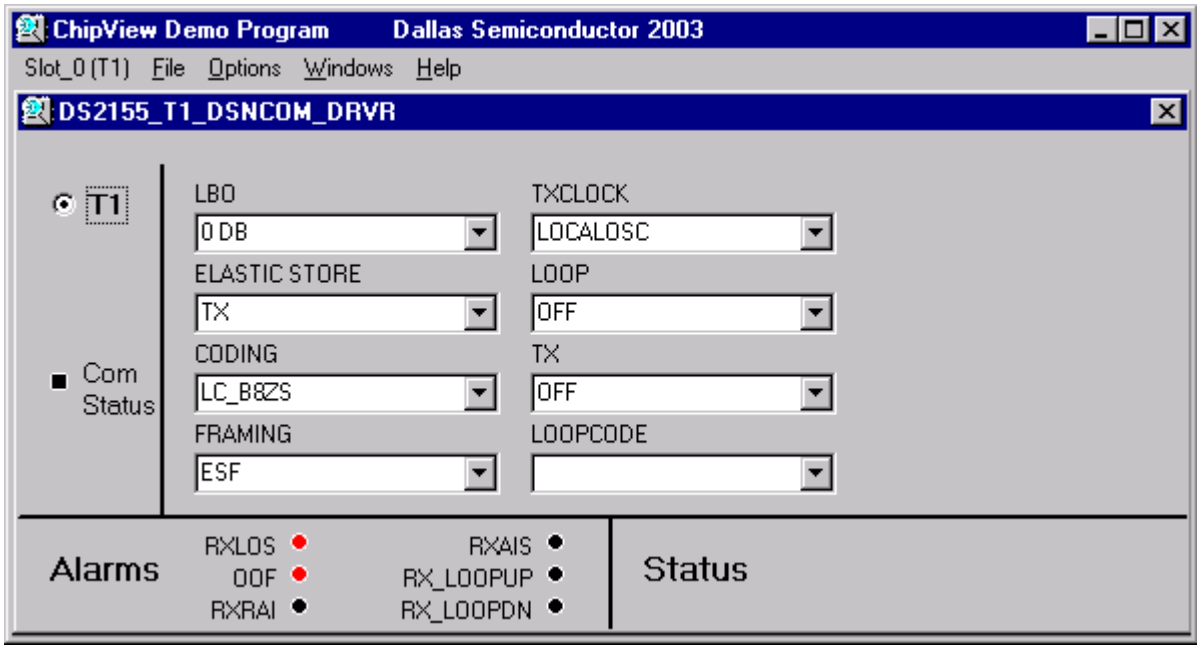
2) For each slot with a daughter card installed, select a configuration file from the list shown, or browse to find a file in another directory. Typically, configuration file names contain the device name, e.g., DS2155.cfg. Some daughter cards ship with multiple configuration files. See the daughter card data sheet for detailed information on the use of the various files.

3) Press the Continue button.

The Demo window shows various daughter-card-specific configuration menus and status indicators. See the daughter card data sheet for details on the specific menus, selections, and indicators used. The Com Status indicator, which is common to most configuration files, changes state approximately once a second when the daughter card software is communicating properly with the daughter card.

To go to the Demo window from other views, select Windows→Go to Demo.

Figure 3. Demo Window



ADVANCED FEATURES

This section discusses several advanced features of the DK2000 platform. Many DK2000 users do not need to read this information. The DK2000 and Dallas daughter cards ship with full definition files for Register View mode and one or more configuration files for Demo mode. These files support most users very well without any need for customization. For users with more complex requirements, however, this section describes how to:

- Create and edit definition (.DEF) files
- Create and edit initialization (.INI) files
- Use Terminal mode

Creating and Editing Definition (.DEF) Files

Definition files are ASCII text files that specify register names, addresses, and bit fields and their arrangement in the Register View window. Dallas Semiconductor distributes full definition files with each daughter card. Any edits to the Dallas definition files should be made in copies of the files and not in the originals.

The text in Figure 4 is a definition file template. Only the REGISTER, DISPLAY, and END fields are required. Each field starts with the field name followed by a colon (i.e., “DEVICE:”) and ends with the next field name. The definition file fields are described in Table 1. All numbers are in decimal format, unless otherwise stated.

Figure 4. Definition File Template

```
REM: remark
DEVICE:
DSxxxx
OFFSET:
0x1000
LINKS:
1
filename
SETUP:
on
REG INI:
on
DSxxxx.INI
DEVICE ID:
on
address,rname,rtype,bus,ivalue,position,fullname,b7,b6,b5,b4,b3,b2,b1,b0,
REGISTER:
number of registers
address,rname,rtype,bus,ivalue,position,fullname,b7,b6,b5,b4,b3,b2,b1,b0,
DISPLAY:
number of columns
1,2,3,4,5,6,7,8,9,10,11,12,13,14,
END:
```

Table 1. Definition File Fields

FIELD	DESCRIPTION
REM	Used for remarks to document the definition file. Cannot be used inside another field.
DEVICE	*This field is not yet supported in the ChipView software.* The argument is a string of text that is displayed at the top of the Register View screen to help the user keep track of which definition file is currently in use.
OFFSET	When located outside the REGISTER field, the argument specifies a global address offset for all registers in the definition file. In some Dallas-made definition files OFFSET has two arguments. Older Dallas demo kit software selects the first argument. The ChipView software selects the last argument. When located inside the REGISTER field, the argument specifies a local address offset for all subsequent register listings. All register addresses following the local OFFSET field are offset by both the global and local offsets. The scope of the local offset is to the end of the REGISTER field or to the next local offset field. Arguments are in four-digit hexadecimal format of the form “0x0000.”
LINKS	Loads additional definition files. Used to accommodate more than one device on a piece of hardware or to split a large register set into smaller subsets. The first argument is a number from 1 to 10 specifying the number of definition files to link. Subsequent arguments are the filenames of the definition files being linked. The number of filenames must be equal to the number specified in the first argument. Linked definition files have all the functionality of the main definition file except that the LINKS field is ignored.
SETUP	*This field is not yet supported in the ChipView software.* Enables initialization register values. The argument must be either “on” or “off.” If the argument is “on,” ChipView initializes all registers with a zero and then the initial value specified in the REGISTER field. When SETUP is “on” the REG INI field is enabled.
REG INI	*This field is not yet supported in the ChipView software.* Specifies an initialization file for initializing register values. REG INI is only enabled if the SETUP field is “on.” The first argument must be either “on” or “off.” The second argument is a valid register initialization file (.INI file). If the SETUP and REG INI fields are both “on,” registers are initialized by the values in the initialization file.
DEVICE ID	*This field is not yet supported in the ChipView software.* Defines how to determine if the device is present on the target hardware. The first argument must be either “on” or “off.” The second argument is a valid register description (see the REGISTER field for format). If the first argument is “on” the ChipView software performs a device-check read/write sequence to the register specified in the second argument. If the device check fails, a Device Not Present error is displayed.
REGISTER	Describes the registers of the target hardware. The first argument is the number of registers (1 to 255). Subsequent arguments are comma-delimited strings with 14 subfields as follows: *address,rname,rtype,bus,ivalue,position,fullname,b7,b6,b5,b4,b3,b2,b1,b0*, The number of strings must be equal to the number of registers specified in the first argument. See Table 2 for subfield definitions.
DISPLAY	*This field is not yet supported in the ChipView software. Currently, registers are displayed 14 per column in the order listed in the REGISTER field.* Specifies how to display the registers on screen. The first argument is a number from 1 to 20 that states the number of columns to be displayed. Subsequent arguments are comma-delimited strings of numbers, where each number specifies a register definition. The first register definition in the REGISTER field is 0, the second is 1, and so on. The strings of numbers can be up to 14 numbers long. The number of strings must be equal to the number of columns specified in the first argument.
END	Specifies the end of the definition file. This field has no arguments.

Table 2. Register Subfield Definitions

SUBFIELD	DESCRIPTION
address	Register address. Hexadecimal format of the form 0x0000.
rname	Register name (acronym) that is displayed in the register map display area. A string of ≤ 7 characters.
rtype	Register type 0 = invalid – not displayed, read, written, or initialized 1 = read-only – cannot be written 2 = read/write – can be read and written 3 = status1 – read operation is preceded by a write of 0xFF 5 = error – cannot be written 6 = test – can be read and written 7 = status2 – read operation is are followed by a write of the value read
bus	This field should be always be "1" for definition files used with DK2000.
ivalue	Initial value written to the register during initialization if the SETUP field is "on." Two-digit hexadecimal format of the form "00."
position	Register position. Allows the user to sequentially number the register definitions for use in the DISPLAY field. These numbers are for the user only; this field is not read by the software. For proper use with the DISPLAY field, register definitions should be numbered consecutively starting from 0 with no missing or repeated numbers.
fullname	Full register name. A string of ≤ 50 characters that is displayed at the top of the bitmap display when the register is selected in the register map.
b7, b6, b5, b4, b3, b2, b1, b0	Bit names. Each is a string of ≤ 6 characters that is displayed in the bit map display.

Creating and Editing Initialization (.INI) Files

Register View mode provides an easy method for initializing an entire register set using initialization files. To initialize the register set from an initialization file, choose File→Register .INI File→Load .INI File. To save the state of a register set to an initialization file, choose File→Register .INI File→Build .INI File. Only the registers of the currently visible definition file are affected by these commands.

Terminal Mode

In addition to Register View mode and Demo mode, the ChipView software also offers Terminal mode, which gives direct access to the processor. The commands that can be entered from Terminal mode are listed in Table 3. The interface specifications are 38,400 baud, 8 data bits, 1 stop bit, no parity, no flow control, ANSI emulation. Locally typed characters are echoed by the DK2000, not the terminal software.

Table 3. Terminal Mode Commands

COMMAND	FUNCTION
F	Display firmware version.
Help | ?	Display help text.
RB <address>	Read from byte at absolute address, no offset added to address. Example: RB 90000000 reads first address in CS9.
RW <address>	Read 16-bit word at absolute address, no offset added to address.
RL <address>	Read 32-bit long word at absolute address, no offset added to address.
Repeat <count> <command string>	Repeats the command given by <command string> the number of times specified by the <count> argument.
setDev <0–F>	Set default device number for use with the X command.
setSlot <0–3>	Set default slot number for use with the X command.
TimInfo [slot number]	Displays information about the attached daughter cards. If no slot number is given, TimInfo displays concise information about all four slots.
WB <address> <value>	Write byte to absolute address, no offset added to address. Example: WB 90000000 FF writes 0xFF to the first address in CS9.
WW <address> <value>	Write 16-bit word to absolute address, no offset added to address.
WL <address> <value>	Write 32-bit long word to absolute address, no offset added to address.

Table 3. Terminal Mode Commands (continued)

COMMAND	FUNCTION
X <addr> [, <endaddr>] [= <value>]	Read or write to daughter card slot addresses. The fifth hex digit of the address is the daughter card slot number. The fourth hex digit of the address is the device number. Addresses with fewer than four hex digits are added to the addresses of the default slot, as set by the setSlot command, plus the default device, as set by the setDev command. Examples: $ X 31020 = FF {write FFh to slot 3, device 1, address 020h} $ X 31999 {read slot 3, device 1, address 999h} FF {value stored in slot 3, device 1, address 999h} $ X 55 {read address 55h of default slot/device as set by setSlot and setDev} 32 {value stored in default slot/device, address 55h} $ X 20, 30 = 5 {write 05h to default slot/device, addresses 20 to 30h}
The following commands are used by Demo mode. They are not recommended for use in Terminal mode.	
CTRL <…> <slot>	The DK2000 firmware includes T1/E1 device driver code written by NComm. CTRL calls the TE1DCTRL device driver API with the indicated parameters (see T1/E1 driver code documentation for details). The slot number on the end is not passed through to the API but is simply used to determine which device driver to call. Example: CTRL 0 400 0 {resets span 0 of slot 0}
POLL <…> <slot>	The DK2000 firmware includes T1/E1 device driver code written by NComm. POLL calls the TE1DPOLL device driver API with the indicated parameters (see T1/E1 driver code documentation for details). The slot number on the end is not passed through to the API but is simply used to determine which device driver to call. Example: POLL 0 600 0 {polls for RLOS on span 0 of slot 0}

Additional Development Resources

The following resources are available for continued development using the DK2000:

NComm, Inc.
T1/E1 Trunk Management Software (TMS™)
www.ncomm.com

Wind River International
The DK2000 firmware uses the VxWorks real-time operating system (RTOS) from Wind River.
www.windriver.com

Electro Surface Technologies, Inc. (EST) (A division of Wind River)
The DK2000 is compatible with the VisionClick C/C++ source-level debugger/flash programming tool.
www.est.com

TMS is a trademark of NComm Inc.

APPENDIX

MPC8260 CPU and Memory Map

CPU Core. The DK2000 development platform is based on the Motorola MPC8260 PowerQUICC II processor. This processor integrates a PowerPC core, a system interface unit (SIU), and a communications processor module (CPM).

The DK2000 board is configured with a 66MHz oscillator providing the system bus clock and SIU clocks. The MPC8260 internally multiplies the system clock to 133MHz for the CPM and to 200MHz for the PowerPC processor core. The internal clock multiplier is determined during PORESET (power-on RESET) based on the states of RSTCONF, MODCLK [1–3], and the values in the hard-reset configuration word bits 28–31. The DK2000 board wires RSTCONF low, forcing the MPC8260 to read the hard-reset configuration word from the beginning of flash memory. DK2000 's default configuration word is configurable depending on your application. Refer to the Motorola MPC8260 PowerQUICC II User's Manual, Section 5.4.1 (www.motorola.com) for detailed information about reset configuration.

RESET CONFIGURATION BYTE	DEFAULT DK2000 VALUE
0	0x1E
1	0x82
2	0x83
3	0x45

SDRAM. The DK2000 development platform contains 64MB of SDRAM controlled by the MPC8260's internal SDRAM controller. The SDRAM is connected to the MPC8260's chip select 2 (CS2).

Level 2 Cache Control. To provide additional performance, the DK2000 board has been designed with the option for 256kB, 512kB, or 1MB of L2 cache. One, two, or four MPC2605s are used as the L2 cache. The MPC2605 can function in either copy-back mode or write-through mode. The L2 cache can be enabled and disabled though a register in the EPLD at CS11 + 0x01. This register controls four signals: L2_FLUSH_L, L2_MISS_INH_L, L2_TAG_CLR_L, AND L2_UPDATE_INH_L. The board powers up with the L2 cache disabled; software must configure the MPC8260 to work with the L2 cache before enabling it. See Table 7 for a description of the L2 cache control register.

FLASH—2 Banks. The DK2000 development platform has 4MB of flash memory organized into two banks. Each bank is organized as 512kB x 32, consisting of four Atmel AT49LV040 devices that are socketed for easy removal and external programming. Through jumper selection, either of the two flash banks can be configured as the boot ROM. The flash banks are controlled by the MPC8260's chip selects 0 and 1 (CS0 and CS1). The chip-select assignment for each bank is a jumper-configurable selection. The silk screening on the board next to the BOOT CONFIG header (P7) indicates which byte lane each FLASH device is attached to. See Figure 4.

EEPROM. A 16kb EEPROM is connected to the SPI™ port on the MPC8260. The EEPROM is organized as 2048 x 8.

Chip-Select Mapping

The MPC8260 has 12 chip-select outputs. The DK2000 board uses these chip selects as defined in Table 4.

CS0 and CS1. Chip selects 0 and 1 are connected to the two 2MB flash banks through jumper block P7. To connect CS0 to bank 0 and CS1 to bank 1, place a jumper across pins 1 and 2 and another jumper across pins 3 and 4. To connect CS0 to bank 1 and CS1 to bank 0, place a jumper across pins 1 and 3 and another jumper across pins 2 and 4. See Figure 5.

CS7. Chip select 7 addresses the STIM board ID and LED control register as shown in Table 5.

SPI is a trademark of Motorola, Inc.

Table 4. Chip Selects and Memory Map

CHIP SELECT	FUNCTION/DEVICE	STARTING ADDRESS
CS0	Flash Bank 0 (Boot ROM)	0x10000000
CS1	Flash Bank 1	0x20000000
CS2	SDRAM	0x00000000
CS3	Daughter Card Slot 0	0x30000000
CS4	Daughter Card Slot 1	0x40000000
CS5	Daughter Card Slot 2	0x50000000
CS6	Daughter Card Slot 3	0x60000000
CS7	STIM ID Control	0x70000000
CS8	STIM Device Control	0x80000000
CS9	Peripherals	0x90000000
CS10	Unused (pin assigned to different function)	
CS11	EPLD	0xB0000000

Note 1: CS0 and CS1 can be swapped with jumper settings.
Note 2: The Dallas Semiconductor device is typically located at the daughter card address + 0x1000. Example: For a daughter card in slot 2, the device address begins at 0x50001000 (0x50000000 plus the daughter card offset of 0x1000).

Table 5. Chip Select 7 Mapping

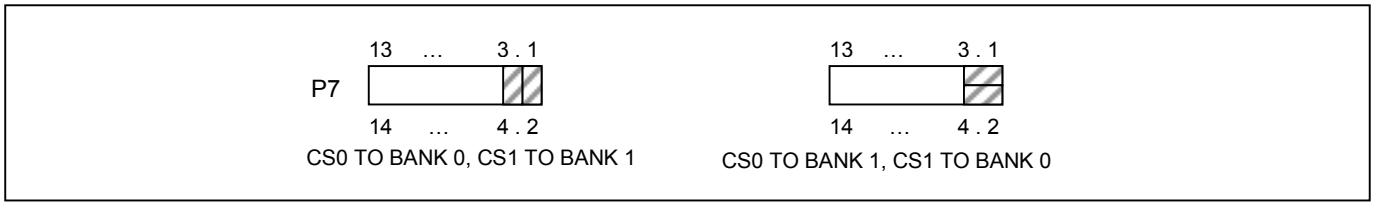
ADDRESS OFFSET	BIT 0	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7
0x00	STIM ID							
0x02	STIM LED		N/A	N/A	N/A	N/A	N/A	N/A

The **STIM ID** register contains the ID of the STIM board. Value is 0xFF if no STIM card is present.

The **STIM LED** register controls the LED on the STIM as follows:

00 LED is OFF
01 LED is ON and is RED
10 LED is ON and is GREEN
11 LED is OFF

Figure 5. Flash Bank Configurations



CS9. Chip select 9 is connected to the EPLD and is used to address several devices. The EPLD decodes CS9 accesses according to the Table 6.

Table 6. Chip Select 9 Mapping

ADDRESS OFFSET	BIT 0	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7
0x0000	Slot 1 Daughter Card ID				Slot 0 Daughter Card ID			
0x1000	Slot 3 Daughter Card ID				Slot 4 Daughter Card ID			
0x2000	Slot 3 LED		Slot 2 LED		Slot 1 LED		Slot 0 LED	
0x3000	FPGA							

The **Slot Daughter Card ID** registers are used to determine the types of daughter cards present in the system. Typical software implementation reads the daughter card ID values and assigns device drivers based on the values. Daughter card IDs that are the value 0xE indicate an extended daughter card ID. The extended daughter card ID can be read at offset 0 of the slot address. A value of 0xF indicates that a daughter card is not present in the slot.

The **Slot LED** registers control the LEDs on the daughter cards as follows:

00 LED is OFF
01 LED is ON and is RED
10 LED is ON and is GREEN
11 LED is OFF

The **FPGA** register holds the address of the FPGA on the board. This is reserved for future use. The FPGA is unpopulated on most boards.

CS11. Chip select 11 is used by the EPLD. The current EPLD implementation has the registers defined in Table 7.

Table 7. Chip Select 11 Mapping

ADDRESS OFFSET	BIT 0	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7
0x00	EPLD Version Number							
0x01	UPDATE	TAG_CLR	FLUSH	MISS_INH	Unused			

UPDATE: Reset to 0. When 0, inhibits updating of the L2 cache.
TAG_CLR: Reset to 0. When 0, clears the L2 cache Tag.
FLUSH: Reset to 0. When 0, flushes the L2 cache.
MISS_INH: Reset to 0. When 0, forces all L2 cache accesses to miss.

MPC8260 I/O Pin Mapping

The MPC8260 has 120 I/O pins that can be configured for special-purpose or general-purpose I/O. The DK2000 development platform takes advantage of as much of the I/O capability as possible. Table 8 shows how the I/O pins are connected on the DK2000 board. Table 9 and Table 10 add details on some of the special-purpose pin assignments.

Table 8. MPC8260 I/O Pin Assignments

PORT A		PORT B		PORT C		PORT D	
PIN	CONNECTOR	PIN	CONNECTOR	PIN	CONNECTOR	PIN	CONNECTOR
0	N.C.	0	N/A	0	Ethernet	0	N/A
1	N.C.	1		1	STIM	1	
2	STIM	2		2	STIM	2	
3	STIM	3		3	STIM	3	
4	N.C.	4	Ethernet	4	Ethernet	4	Daughter Card 3
5	N.C.	5	Ethernet	5	Ethernet	5	N.C.
6	Daughter Card 0	6	Ethernet	6	Daughter Card 0–3	6	N.C.
7	Daughter Card 0	7	Ethernet	7	Daughter Card 0–3	7	Daughter Card 0–3
8	Daughter Card 0	8	Ethernet	8	Ethernet	8	RS-232
9	Daughter Card 0	9	Ethernet	9	Daughter Card 0–3, Daughter Card	9	RS-232
10	Daughter Card 0–3	10	Ethernet	10	Ethernet	10	Daughter Card 1
11	Daughter Card 0–3	11	Ethernet	11	Ethernet	11	Daughter Card 1
12	Daughter Card 0–3	12	Ethernet	12	Daughter Card 0–3	12	Daughter Card 1
13	Daughter Card 0–3	13	Ethernet	13	Daughter Card 0–3	13	Daughter Card 1
14	Daughter Card 0–3	14	Ethernet	14	Daughter Card 0–3	14	I^2C™
15	Daughter Card 0–3	15	Ethernet	15	Daughter Card 0–3	15	I^2C
16	Daughter Card 0–3	16	Ethernet	16	Ethernet	16	Daughter Card 2
17	Daughter Card 0–3	17	Ethernet	17	Ethernet	17	Daughter Card 0–3
18	Daughter Card 0–3	18	Ethernet	18	Daughter Card 0–3, STIM	18	Daughter Card 0–3
19	Daughter Card 0–3	19	Ethernet	19	Daughter Card 0–3, STIM	19	Daughter Card 0–3
20	Daughter Card 0–3	20	STIM	20	Daughter Card 0–3	20	Daughter Card 0–3, STIM
21	Daughter Card 0–3	21	STIM	21	Daughter Card 0–3	21	Daughter Card 0–3, STIM
22	Daughter Card 0–3	22	STIM	22	Daughter Card 1	22	Daughter Card 0–3, STIM
23	Daughter Card 0–3	23	STIM	23	Daughter Card 1	23	Daughter Card 3
24	Daughter Card 0–3	24	STIM	24	Daughter Card 3	24	Daughter Card 3
25	Daughter Card 0–3	25	STIM	25	Daughter Card 3	25	Daughter Card 3
26	Daughter Card 0–3	26	STIM	26	Daughter Card 2	26	Daughter Card 2
27	Daughter Card 0–3	27	STIM	27	Daughter Card 2	27	Daughter Card 2
28	Daughter Card 0–3	28	Byte Blaster	28	STIM	28	Daughter Card 2
29	Daughter Card 0–3	29	Byte Blaster	29	STIM	29	Daughter Card 0–3
30	Daughter Card 0–3	30	Byte Blaster	30	Daughter Card 1	30	N.C.
31	Daughter Card 0–3	31	Byte Blaster	31	Daughter Card 1	31	N.C.

I^2C is a trademark of Philips Corp. Purchase of I^2C components of Maxim Integrated Products, Inc., or one of its sublicensed Associated Companies, conveys a license under the Philips I^2C Patent Rights to use these components in a I^2C system, provided that the system conforms to the I^2C Standard Specification as defined by Philips.

Table 9. MPC8260 I/O Pin Assignments for TDM Connections

NAME	SCHEMATIC A1–D1 SLOT 0–2	SCHEMATIC D2 SHARED	TDM A1 SLOT 0	TDM B1 SLOT 1	TDM C1 SLOT 2	TDM D1 SLOT 3	TDM A2 STIM	TDM C2 STIM	TDM D2 STIM
RXCLK	NIMX_2	SNIM_B7	PC31	PC23	PC27	PC25	PC19	PC29	PA3
TXCLK	NIMX_4	SNIM_B6	PC30	PC22	PC26	PC24	PC18	PC28	PA2
RXD	NIMX_6	SNIM_B5	PA8	PD12	PD27	PD24	PD21	PB26	PB22
TXD	NIMX_7	SNIM_B4	PA9	PD13	PD28	PD25	PD22	PB27	PB23
RSYNC	NIMX_8	SNIM_B3	PA6	PD10	PD26	PD23	PD20	PB24	PB20
TSYNC	NIMX_9	SNIM_B2	PA7	PD11	PD16	PD4	PC9	PB25	PB21

Note: TDM D2 is shared between daughter card slots and STIM connector.

Table 10. MPC8260 I/O Pin Assignments for UTOPIA Bus

NAME	UTOPIA FUNCTION
PA31	TXENA
PA30	TXCLAV0
PA29	TXSOC
PA28	RXENA
PA27	RXSOC
PA26	RXCLAV0
PA25	TXD0
PA24	TXD1
PA23	TXD2
PA22	TXD3
PA21	TXD4
PA20	TXD5
PA19	TXD6
PA18	TXD7
PA17	RXD7
PA16	RXD6
PA15	RXD5
PA14	RXD4
PA13	RXD3

NAME	UTOPIA FUNCTION
PA12	RXD2
PA11	RXD1
PA10	RXD0
PC21	TXCLK
PC20	RXCLK
PC15	TXADDR0
PC14	RXADDR0
PC13	TXADDR1
PC12	RXADDR1
PC7	TXADDR2/TXCLAV1
PC6	RXADDR2/RXCLAV1
PD29	RXADDR3/RXCLAV2
PD19	TXADDR4/CLAV3
PD18	RXADDR4/RXCLAV3
PD17	RXPRTY
PD7	TXADDR3/TXCLAV2
EPLD (Generated in Logic)	TXPRTY

MPC8260 Interrupts

The MPC8260 supports eight external interrupt lines. The interrupts are connected to the system as described in Table 11.

Table 11. MPC8260 Interrupt Connections

INTERRUPT	CONNECTION
IRQ0	FPGA, STIM
IRQ1	FPGA, Daughter Card 1, STIM
IRQ2	FPGA, Daughter Card 2
IRQ3	FPGA, Daughter Card 3
IRQ4	FPGA, STIM
IRQ5	FPGA, Daughter Cards 0–3
IRQ6	FPGA
IRQ7	STIM

In general, the daughter cards use interrupts IRQ1, IRQ2, IRQ3, and IRQ5. IRQ5 is bused among all four daughter cards but is not used by daughter card 1, daughter card 2, or daughter card 3. The STIM card uses IRQ4 for its main interrupt.

Daughter Card Interface Pin Assignment

The DK2000 has four daughter card interfaces each consisting of three 50-position connectors: PX, P1X, and P2X (where X = 1 to 4 = slot number + 1). Table 12 shows the pin assignment for these connectors.

Table 12. Daughter Card Connector Pin Assignment

PX CONNECTOR		P1X CONNECTOR		P2X CONNECTOR	
PIN	NAME	PIN	NAME	PIN	NAME
1	+5V	1	+5V	1	TXADDR[3]/TXCLAV[2]
2	GND	2	GND	2	TXADDR[4]/TXCLAV[3]
3	NIMX_0	3	SNIM_NX_0	3	TXADDR[1]
4	LA31	4	SNIM_B0	4	TXADDR[2]/TXCLAV[1]
5	NIMX_1	5	SNIM_NX_1	5	GND
6	LA30	6	SNIM_B1	6	TXADDR[0]
7	NIMX_2	7	SNIM_NX_2	7	TXDATA[6]
8	LA29	8	SNIM_B2	8	TXDATA[7]
9	NIMX_3	9	SNIM_NX_3	9	TXDATA[4]
10	LA28	10	SNIM_B3	10	TXDATA[5]
11	NIMX_4	11	SNIM_NX_4	11	TXDATA[2]
12	LA27	12	SNIM_B4	12	TXDATA[3]
13	NIMX_5	13	SNIM_NX_5	13	TXDATA[0]
14	LA26	14	SNIM_B5	14	TXDATA[1]
15	NIMX_6	15	LA21	15	TXPRTY
16	LA25	16	SNIM_B6	16	TXENA
17	NIMX_7	17	LA20	17	TXSOC
18	LA24	18	SNIM_B7	18	TXCLK
19	NIMX_8	19	LA19	19	GND
20	LA23	20	NIMD15	20	GND
21	NIMX_9	21	LA18	21	TXADDR[3]/TXCLAV[2]
22	LA22	22	NIMD14	22	TXADDR[4]/TXCLAV[3]
23	NIMX_10	23	+3.3V	23	TXCLAV[0]
24	RHWL	24	NIMD13	24	TXADDR[2]/TXCLAV[1]
25	NIMX_11	25	CLK16384MHZ	25	GND
26	GND	26	GND	26	GND
27	NIMX_12	27	+3.3V	27	RXADDR[3]/RXCLAV[2]
28	BWE0L	28	NIMD12	28	RXADDR[4]/RXCLAV[3]
29	NIMX_13	29	LA17	29	RXADDR[1]
30	HRESETL	30	NIMD11	30	RXADDR[2]/RXCLAV[1]
31	NIMX_CSL	31	LA16	31	GND
32	CPUCLK5	32	NIMD10	32	RXADDR[0]
33	NIMX_ID0	33	LA15	33	RXDATA[6]
34	NIMD7	34	NIMD9	34	RXDATA[7]
35	NIMX_ID1	35	LA14	35	RXDATA[4]
36	NIMD6	36	NIMD8	36	RXDATA[5]
37	NIMX_ID2	37	LA13	37	RXDATA[2]
38	NIMD5	38	FPGAOEL	38	RXDATA[3]
39	NIMX_ID3	39	LA12	39	RXDATA[0]
40	NIMD4	40	BWE1L	40	RXDATA[1]
41	CLK44736MHZ	41	LA11	41	RXPRTY
42	NIMD3	42	GND	42	RXENA
43	N.C.	43	+2.5V	43	RXSOC
44	NIMD2	44	CLK1544MHZ	44	RXCLK
45	IRQ5L	45	+5V	45	GND
46	NIMD1	46	GND	46	GND
47	IRQ	47	CLK20MHZ	47	RXADDR[3]/RXCLAV[2]

Table 12. Daughter Card Connector Pin Assignment (continued)

PX CONNECTOR		P1X CONNECTOR		P2X CONNECTOR	
PIN	NAME	PIN	NAME	PIN	NAME
48	NIMD0	48	CLK3088MHZ	48	RXADDR[4]/RXCLAV[3]
49	+5V	49	+5V	49	RXCLAV[0]
50	GND	50	GND	50	RXADDR[2]/RXCLAV[1]

Side TIM Pin Assignment

A side TIM (STIM) interface is provided to expand the capabilities of the DK2000 design. The STIM interface provides the entire PowerPC 60x bus from the MPC8260, along with three TDM channels and various system-generated clocks. The interface is composed of two 96-position DIN connectors. Table 13 shows the pin assignment for this connector.

Table 13. STIM Connector Pin Assignment

PIN	J3 CONNECTOR			J4 CONNECTOR		
	ROW A	ROW B	ROW C	ROW A	ROW B	ROW C
1	D31	+5V	D15	D47	L2A6	D63
2	D30	+5V	D14	D46	L2A5	D62
3	D29	L2A31	D13	D45	L2A4	D61
4	D28	L2A30	D12	D44	L2A3	D60
5	D27	L2A29	D11	D43	L2A2	D59
6	D26	L2A28	D10	D42	L2A1	D58
7	D25	L2A27	D9	D41	L2A0	D57
8	D24	GND	D8	D40	PB21	D56
9	D23	L2A26	D7	D39	PB20	D55
10	D22	L2A25	D6	D38	PB23	D54
11	D21	L2A24	D5	D37	PB22	D53
12	D20	L2A23	D4	D36	PA2	D52
13	D19	L2A22	D3	D35	PA3	D51
14	D18	GND	D2	D34	PC3	D50
15	D17	L2A21	D1	D33	PC2	D49
16	D16	L2A20	D0	D32	PC1	D48
17	CLK16384MHZ	L2A19	PC29	TT2	+3.3V	TT4
18	BALE	L2A18	PC28	TT1	+3.3V	TT3
19	PB25	L2A17	PB26	TT0	+3.3V	GND
20	PB24	GND	PB27	TSIZ3	WE4L	CLK1944MHZ
21	RAMPSDA10	L2A16	RAMCAS	TSIZ2	WE5L	GND
22	RAMWEL	L2A15	RAMGL4L	TSIZ1	WE6L	GND
23	CS8L	L2A14	RAMPSDAMUX	TSIZ0	WE7L	CLK3088MHZ
24	CS7L	L2A13	CLK1544MHZ	TSL	+2.5V	GND
25	RHWL	L2A12	TAL	TBSTL	+2.5V	CLK20MHZ
26	OEL	GND	TEAL	BGL	+2.5V	GND
27	+5V	L2A11	HRESET	BRL	PD21	GND
28	+5V	L2A10	IRQ7L	DBBL	PD22	GND
29	WE0L	L2A9	IRQ4L	DBGL	PC19	CLK44736MHZ1
30	WE1L	L2A8	IRQ1L	GBL_L	PC18	GND
31	WE2L	L2A7	IRQ0L	MISC4	PD20	CLK6312MHZ
32	WE3L	GND	CPUCLK3	CACHE0	PC9	GND

Ethernet Interface

The DK2000 development platform includes a fast Ethernet port. The Ethernet port consists of a Level One LXT970 Ethernet PHY connected to FCC3 of the MPC8260. The configuration of the Ethernet port is accomplished using the general-purpose I/O pins of the MPC8260, making the DK2000 adaptable to the network environment. Table 14 shows the I/O pins and their functions with the LXT970.

Table 14. Ethernet I/O Pin Assignments

PIN	FUNCTION	PIN	FUNCTION
PB14	TX_EN	PB8	RXD0
PB7	TXD0	PB9	RXD1
PB6	TXD1	PB10	RXD2
PB5	TXD2	PB11	RXD3
PB4	TXD3	PB12	CRS
PB15	TX_ERR	PB19	MDIO
PC16	TX_CLK	PB18	MDC
PB13	COL	PC0	CFG0
PB17	RX_DV	PC4	CFG1
PB16	RX_ER	PC5	FDE
PC17	RX_CLK	PC8	MDDIS
PC11	PWRDWN	PC10	TRSTE

The Ethernet interface is not currently supported in software.

Debug Interface

The DK2000 platform provides two debug connectors to satisfy debug and development needs. Connector P9 is a standard JTAG/COP interface to the MPC8260 as defined by Motorola. Connector P20 is a Vision Probe/Vision ICE connector, as defined by Wind River.

UPDATES AND ADDITIONAL DOCUMENTATION

Software updates, IC data sheets, and daughter card documentation are available on our website, www.maxim-ic.com/telecom.

TECHNICAL SUPPORT

For additional technical support, please e-mail your questions to telecom.support@dalsemi.com.

SCHEMATICS

The installation program for the ChipView software also loads a .PDF file containing the DK2000 schematics. To access this file, click the Start button on the Windows toolbar and select: Programs→ChipView→DK2000 Schematics.