

# BlueCat Linux Board Support Guide

---

BlueCat Linux Release 5.1

DOC-0627-00

*for Motorola MPC5200 Lite5200 Boards*

Product names mentioned in the *BlueCat Linux Board Support Guide for Motorola MPC5200 Lite5200 Board* are trademarks of their respective manufacturers and are used here only for identification purposes.

Copyright ©1987 - 2005, LynuxWorks, Inc. All rights reserved.  
U.S. Patents 5,469,571; 5,594,903

Printed in the United States of America.

All rights reserved. No part of the *BlueCat Linux Board Support Guide for Motorola MPC5200 Lite5200 Board* may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photographic, magnetic, or otherwise, without the prior written permission of LynuxWorks, Inc.

LynuxWorks, Inc. makes no representations, express or implied, with respect to this documentation or the software it describes, including (with no limitation) any implied warranties of utility or fitness for any particular purpose; all such warranties are expressly disclaimed. Neither LynuxWorks, Inc., nor its distributors, nor its dealers shall be liable for any indirect, incidental, or consequential damages under any circumstances.

(The exclusion of implied warranties may not apply in all cases under some statutes, and thus the above exclusion may not apply. This warranty provides the purchaser with specific legal rights. There may be other purchaser rights which vary from state to state within the United States of America.)

---

# Contents

---

<b>PREFACE</b>	.....	<b>V</b>
	For More Information .....	v
	Typographical Conventions .....	vi
	Special Notes .....	vii
	Technical Support .....	vii
	How to Submit a Support Request .....	vii
	Where to Submit a Support Request .....	viii
<b>CHAPTER 1</b>	<b>OVERVIEW</b> .....	<b>1</b>
	Features Overview .....	1
	Kernel Version .....	1
	BlueCat Linux Cross-Development Tools .....	2
	The MPC5200 Lite5200 GPIO API .....	2
	User Space IRQ Delivery API .....	2
	Supported Hardware .....	2
	Available BlueCat Linux Development Tools .....	3
	Supported Cross-Development Hosts .....	3
<b>CHAPTER 2</b>	<b>DOWNLOADING AND BOOTING BLUECAT LINUX ON THE TARGET</b> .....	<b>5</b>
	Prerequisites .....	5
	Downloading and Booting Overview .....	6
	Setting up Hardware .....	6
	Connecting the Target Board Serial Port to the Host .....	6
	Connecting the Target Platform Ethernet Card to the Host .....	7
	Setting up the Motorola dBUG Firmware .....	7
	Downloading a BlueCat Linux System into Flash .....	8

Downloading a BlueCat Linux System into Flash Using the Motorola dBUG Firmware .....	8
Downloading a BlueCat Linux System into Flash Using the OS Loader .....	9
Booting a BlueCat Linux System from a Network .....	10
Booting a BlueCat Linux System from a Network using the Motorola dBUG Firmware .....	10
Booting a BlueCat Linux System from a Network using the OS Loader .....	12

---

**CHAPTER 3    KERNEL CONFIGURATION OPTIONS ..... 15**

---

<b>CHAPTER 4    SUPPORTED DEMO SYSTEMS..... 37</b>	
Demo Systems .....	37
developer Demo System .....	37
osloader Demo System .....	38
showcase Demo System .....	38

---

<b>CHAPTER 5    SUPPORTED DEVICE DRIVERS ..... 39</b>	
The MPC5200 Lite5200 GPIO API .....	40
User Space IRQ Delivery API .....	42

---

<b>CHAPTER 6    KNOWN PROBLEMS AND LIMITATIONS..... 45</b>	
Motorola MPC5200 Lite5200 Board Problems and Limitations .....	45
User Documentation Updates .....	46

---

# Preface

---

## For More Information

For more information on the features of BlueCat Linux, refer to the following printed and online documentation.

- *BlueCat Linux User's Guide*

This document contains information about installing, configuring, and using BlueCat Linux.

- Online information

The complete BlueCat Linux documentation set is available on the BlueCat Linux Documentation CD-ROM. Books are provided in both HTML and PDF formats.

Updates to these documents are available online at the LynuxWorks Website: <http://www.lynuxworks.com>.

Additional information about commands and utilities is provided online with the `man` command. For example, to find information about the GNU GCC compiler, use the following syntax:

```
man gcc
```

## Typographical Conventions

The typefaces used in this manual, summarized below, emphasize important concepts. All references to filenames and commands are case-sensitive and should be typed accurately.

### Kind of Text

### Examples

Body text; *italicized* for emphasis, new terms, and book titles

Refer to the *BlueCat Linux User's Guide*.

Environment variables, filenames, functions, methods, options, parameter names, path names, commands, and computer data

```
ls
-l
myprog.c
/dev/null
```

Commands that need to be highlighted within body text, or commands that must be typed as is by the user are **bolded**.

```
login: myname
# cd /usr/home
```

Text that represents a variable, such as a filename or a value that must be entered by the user, is *italicized*.

```
cat <filename>
mv <file1> <file2>
```

Blocks of text that appear on the display screen after entering instructions or commands

```
Linux version 2.4.10-1
(bin@build1) (gcc version
2.95.3 20010315 (release)) #5
Tue Dec 18 13:33:08 MSK 2001
Processor: Intel StrongARM-
IXP1200 revision 3
Architecture: Intel IXP1200
On node 0 totalpages: 32768
zone(0): 32768 pages.
zone(1): 0 pages.
zone(2): 0 pages.
```

Keyboard options, button names, and menu sequences

**Enter**, **Ctrl-C**

## Special Notes

The following notations highlight any key points and cautionary notes that may appear in this manual.

---

**NOTE:** These callouts note important or useful points in the text.

---



**CAUTION!** Used for situations that present minor hazards that may interfere with or threaten equipment/performance.

---

## Technical Support

LynuxWorks Support handles support requests from current support subscribers.

For questions regarding LynuxWorks products or evaluation CDs, or to become a support subscriber, our knowledgeable sales staff will be pleased to help you (<http://www.lynuxworks.com/corporate/contact/sales.php3>).

### How to Submit a Support Request

When you are ready to submit a support request, please include *all* the following information:

- First name
- Last name
- Your job title
- Phone number
- Fax number
- E-mail address
- Company name
- Address
- City, state, ZIP

- Country
- LynxOS or BlueCat Linux version you are using
- Target platform (for example, PowerPC or x86)
- Board Support Package (BSP)
- Current patch revision level
- Development host OS version
- Description of problem you are experiencing

## Where to Submit a Support Request

### By E-mail:

Support, Europe	tech_europe@lnxw.com
Support, worldwide except Europe	support@lnxw.com
Training and courses	USA: training-usa@lnxw.com Europe: training-europe@lnxw.com

### By Phone:

Training and courses	USA: +1 408-979-4353 Europe: +33 1 30 85 06 00
Support, Europe (from our Paris, France office)	+33 1 30 85 93 96
Support, worldwide except Europe and Japan (from our San José, CA, USA headquarters)	+1 800-327-5969 or +1 408-979-3940
Support, Japan	+81 33 449 3131

**By Fax:**

Support, Europe (from our Paris, France office)	+33 1 30 85 06 06
Support, worldwide except Europe and Japan (from our San José, CA, USA headquarters)	+1 408-979-3945
Support, Japan	+81 22 449 3803



The *BlueCat Linux Board Support Guide for Motorola MPC5200 Lite5200 Boards* provides information about the BlueCat Linux Board Support Package (BSP) for the Motorola MPC5200 Lite5200 board.

The Motorola MPC5200 Lite5200 board is based on a 400 MHz MPC603e processor core with an integrated double precision Floating Point Unit that operates at -40 to 85 degrees C. The MPC5200 was designed for fast data throughput and processing. The integrated BestComm DMA controller offloads the main MPC603e core from I/O intensive data transfers. An integrated Double Data Rate (DDR) memory controller accelerates data access with an effective memory bus speed of 266 MHz. A high-speed PCI interface backed by the BestComm DMA controller and DDR support enables high-speed data transfers in and out of the MPC5200. The MPC5200 is well suited for networking, industrial control, and automotive applications. The MPC5200 serves the processing-intensive network gateway, automotive, Internet access, industrial automation, image detection/analysis, and electronic/medical instrumentation markets.

Throughout this Board Support Guide (BSG), the BSP is referred to as the “lite5200” and the target board is referred to as the “MPC5200 Lite5200 board” or simply as the “target board.”

---

## Features Overview

The following sections describe the new features of this release.

### Kernel Version

BlueCat Linux release 5.1 is based on the Linux kernel version 2.6.7 available from [www.kernel.org](http://www.kernel.org).

## BlueCat Linux Cross-Development Tools

BlueCat Linux release 5.1 supports the following versions of the GNU toolchain:

- gcc version 3.2.2
- binutils version 2.13.1

---

## The MPC5200 Lite5200 GPIO API

The BlueCat Linux kernel for the MPC5200 Lite5200 board provides a set of services to control the GPIO pins of the MPC5200 processor.

---

## User Space IRQ Delivery API

The BlueCat Linux kernel for MPC5200 Lite5200 board provides the ability to handle the hardware interrupts in user space applications.

---

## Supported Hardware

Table 1-1 describes the hardware supported with this release. For available BlueCat Linux drivers, please see Chapter 5, “Supported Device Drivers.”

**Table 1-1: Hardware Supported**

Model	Description
Motorola MPC5200 Lite5200 board	<ul style="list-style-type: none"><li>• MPC603e PowerPC processor core with a double-precision FPU</li><li>• Big-endian</li><li>• 64 MB SDR and DDR SDRAM</li><li>• 16 KB instruction cache, 16 KB data cache integrated to the MPC603e PowerPC core</li><li>• Memory Management Units, one for each cache integrated to the MPC603e PowerPC core</li><li>• I2C (to 520 Kbps) for E2PROM (also available on two connectors)</li><li>• Single USB 1.1 (master-only compatibility)</li></ul>

**Table 1-1: Hardware Supported (Continued)**

Model	Description
	<ul style="list-style-type: none"> <li>• Multiple, reconfigurable GPIO</li> <li>• PCI interface with standard PC-type connector</li> <li>• Built-in BestComm DMA I/O subsystem that contains an intelligent DMA unit, which provides a frontline interrupt control and data movement via a separate peripheral bus to the on-chip peripheral functions</li> <li>• Built-in interrupt controller that manages 4 external interrupt request lines and 47 internal interrupt sources</li> <li>• Peripheral Serial Controller (PSC1) with transceiver</li> <li>• Dual MSCAN2.0 A/B built-in controller modules</li> <li>• Built-in Fast Ethernet controller (FEC) with a 10 Mbps and 100 Mbps IEE 802.3 MII and 10 Mbps 7-wire interface</li> </ul>

---

## Available BlueCat Linux Development Tools

Table 1-2 indicates the availability of BlueCat Linux development tools on the cross-development platforms listed for use with the lite5200 BSP.

**Table 1-2: BlueCat Linux Tools Availability**

Tool	Windows	Linux
CodeWarrior	N/A	N/A
SpyKer	N/A	N/A
VisualLynux	✓	N/A

---

## Supported Cross-Development Hosts

The BlueCat Linux development environment requires an installed, functional cross-development host with an Intel 386 or higher CPU. This host needs to be running one of the following development environments:

- Windows 2000/Pro with SP1 or later
- Windows XP

- PC running Red Hat Linux 8.0
- PC running Red Hat Linux 9

# *Downloading and Booting BlueCat Linux on the Target*

This chapter provides instructions for downloading a BlueCat Linux demo system from a cross-development host onto the target and then booting the demo system on the target platform.

---

## **Prerequisites**

This document is a guide to downloading and booting BlueCat Linux systems onto the user's target platform. Scenarios that use demo systems included in the BlueCat Linux distribution are presented. A basic familiarity with the target platform hardware and operation is required. The user must also have an understanding of system administration for the particular cross-development host on which the BlueCat Linux Core and the BSP are installed. It is assumed that the user has the manufacturer's documentation for the target board as well as system administration reference material for the cross-development host.

Before downloading and booting BlueCat Linux on the target board, it is assumed that the default BlueCat Linux PowerPC configuration and the lite5200 BSP have been installed on the cross-development host. This means that the user must:

1. Install the BlueCat Linux PowerPC Core onto the cross-development host, as described in the "Installing the Default Configuration" section in Chapter 1, "Introduction and Installation" in the *BlueCat Linux User's Guide*.
2. Install the lite5200 BSP onto the cross-development host as detailed in the "Installing Target Board Support" section of Chapter 1, "Introduction and Installation" in the *BlueCat Linux User's Guide*.
3. Activate support for the lite5200 BSP as detailed in the "Activating Support for a Target Board" section of Chapter 1, "Introduction and Installation" in the *BlueCat Linux User's Guide*.

---

## Downloading and Booting Overview

The procedure for downloading and booting a BlueCat Linux system on the Motorola MPC5200 Lite5200 target consists of the following main steps:

- Setting up hardware
- Setting up the Motorola dBUG firmware
- Downloading and booting a BlueCat Linux system from the target Flash memory or a network

Downloading and booting a BlueCat Linux system can be performed using either of the two OS boot loaders:

- Motorola dBUG firmware
- BlueCat Linux OS loader

The BlueCat Linux OS loader demo system currently includes the `osloader` Kernel Downloadable Image (KDI). `osloader` is the image with the base functionality of the BlueCat Linux OS loader configured in.

Please refer to Chapter 3, “Downloading and Booting BlueCat Linux” in the *BlueCat Linux User’s Guide* for a discussion of the OS loader.

---

## Setting up Hardware

### Connecting the Target Board Serial Port to the Host

The target board has one serial port. This port is used both by the Motorola dBUG firmware and the BlueCat Linux system console.

The serial port connected to the target serial port has a baud rate of 9600.

Throughout this chapter, the terminal window connected to the serial connector is referred to as the “Motorola dBUG firmware console” or the “BlueCat Linux console,” depending on the context.

## Connecting the Target Platform Ethernet Card to the Host

The Ethernet port on the target board is used to provide a standard network connection for the board and, in particular, to load BlueCat Linux embedded systems onto the board over a network. The Ethernet port on the MPC5200 Lite5200 board is used to connect to a LAN.

It is also required that the user set up networking on the host system. In particular, the user must choose a unique IP address for the development host as well as for the target board. These addresses are referred to as `<host_IP_address>` and `<target_IP_address>`, respectively. For more information on how to set up networking on the host, please refer to system administration reference material.

TFTP must be enabled on the host. For more information, refer to “Setting Up a TFTP Server” in Chapter 3, “Downloading and Booting BlueCat Linux” in the *BlueCat Linux User’s Guide*.

---

## Setting up the Motorola dBUG Firmware

To set up the Motorola dBUG firmware options for BlueCat Linux, perform the following steps:

1. Reset the target board.

The Motorola dBUG firmware boots up, and the prompt (dBUG>) appears on the Motorola dBUG firmware console.

2. Perform the following commands:

```
dBUG> set client <target_IP_address>
dBUG> set server <host_IP_address>
dBUG> set netmask <netmask>
dBUG> set filetype image
dBUG> store
```

where `<target_IP_address>` is the IP address of the target board, `<host_IP_address>` is the IP address of the host, and `<netmask>` is the network netmask.

---

## Downloading a BlueCat Linux System into Flash

This section provides instructions on how a BlueCat Linux embedded system can be downloaded into the target Flash memory using either the Motorola dBUG firmware or the BlueCat Linux OS loader. Refer also to the *BlueCat Linux User's Guide* for additional details about the BlueCat Linux OS loader.

### Downloading a BlueCat Linux System into Flash Using the Motorola dBUG Firmware

To download a BlueCat Linux embedded system into the target Flash memory using the Motorola dBUG firmware, perform the steps below. This section uses the `osloader` demo system as an example, but these instructions are applicable to any of the demo systems.

1. Copy the `osloader.kdi` file from the `$BLUECAT_PREFIX/demo/osloader` directory to the `/tftpboot` directory on the development host:

```
BlueCat:$ cp
$BLUECAT_PREFIX/demo/osloader/osloader.kdi \
/tftpboot/osloader.kdi
```

2. Reset the target board.
3. At the dBUG prompt, enter the following commands:

```
dBUG> dn osloader.kdi
dBUG> fe ff000000 ff14ffff
```

Type **YES** at the confirmation prompt.

```
dBUG> fp ff000000 ff14ffff 20000
```

As the result, the `osloader` demo is programmed into Flash.

To boot the `osloader` demo installed into the Flash memory, type the following command at the dBUG prompt:

```
dBUG> go ff000000
```

This command starts the `osloader` demo system programmed into Flash.

## Downloading a BlueCat Linux System into Flash Using the OS Loader

To download a BlueCat Linux embedded system into the target Flash memory using the BlueCat OS loader, perform the steps below. This section uses the `osloader` demo system as an example, but these instructions are applicable to any of the demo systems.

1. Copy the `i_osloader.kdi` file from the `$BLUECAT_PREFIX/demo/osloader` directory to the `/tftpboot` directory on the development host:

```
BlueCat:$ cp
$BLUECAT_PREFIX/demo/osloader/i_osloader.kdi \
/tftpboot/i_osloader.kdi
```

2. Copy the `osloader.kdi` file from the `$BLUECAT_PREFIX/demo/osloader` directory to the `/tftpboot` directory on the development host:

```
BlueCat:$ cp
$BLUECAT_PREFIX/demo/osloader/osloader.kdi \
/tftpboot/osloader.kdi
```

3. Reset the target board.
4. Boot the `i_osloader` demo from a network as described in “Booting a BlueCat Linux System from a Network” on page 10.

The BlueCat Linux OS loader prompt (`>`) appears on the console.

5. At the BlueCat Linux OS loader prompt, type the following commands:

```
> set IF eth0
> set IP <target_board_IP>
> set HOST <development_host_IP>
> set FILE tftp osloader.kdi
> exec flash_fdisk /dev/mtdchar0 0-20
> flash /dev/mtdchar1 erase
> reset
```

where `<target_board_IP>` is the IP address of the target and `<development_host_IP>` is the IP address of the development host.

The partition size specified in the `flash_fdisk` command depends on the size of the system and should be large enough to hold the BlueCat Linux demo KDI. For the example above, the following calculation

shows that a partition of size 0-20 is a correct parameter for the `osloader demo` system:

$$(20 - 0 + 1) * 64KB = 1344KB$$

where 64 KB is the size of the Flash sector. The calculated value 1344 KB is greater than the ROM requirement for the `osloader demo` (1202.5 KB). Refer to Chapter 4, “Supported Demo Systems” for details about the ROM requirements for all supported BlueCat Linux demo systems.

As the result, the `osloader demo` is programmed into Flash.

To boot the `osloader demo` installed into the Flash memory, type the following command at the dBUG prompt:

```
dBUG> go ff000000
```

This command starts the `osloader demo` system programmed into Flash.

---

## Booting a BlueCat Linux System from a Network

A BlueCat Linux demo system can be booted from a network using either the Motorola dBUG firmware or the BlueCat Linux OS loader.

### Booting a BlueCat Linux System from a Network using the Motorola dBUG Firmware

The Motorola dBUG firmware uses the TFTP network protocol to load BlueCat Linux images over a network. To boot the `osloader demo` system over a network using the dBUG firmware, perform the following steps:

1. Copy the `osloader.kdi` file from the `BLUECAT_PREFIX/demo/osloader` directory to the `/tftpboot` directory on the cross-development host.

```
BlueCat:$ cp $BLUECAT_PREFIX/demo/osloader/\
osloader.kdi /tftpboot/osloader.kdi
```

2. Reset the board.

The following information appears on the console:

```
MPC6XX Lite5200 16MB Firmware v2d.2a.1a (Build 18 on Dec 3 2003
18:02:05)
```

```
CPU: MPC5200 v1.2
Clocks: Core=396.00 MHz, SDRAM=132.00 MHz, IPBI=132.00 MHz, PCI=33.00
MHz
SDRAM: 00000000 - 03FFFFFF = 64 MBytes SDR
Flash: FF000000 - FFFFFFFF = 16 MBytes
```

```
MBIST: ok
```

```
Enter 'help' for help.
dBUG>
```

3. To download the `osloader` demo image, execute the following command:

```
dBUG> dn osloader.kdi

Address: 0x00020000
Downloading Image 'osloader.kdi' from 192.168.4.100
.....
1231361 bytes read via TFTP

dBUG>
```

---

**NOTE:** The user can set the default name for a BlueCat Linux image using the following commands:

```
dBUG> set filename <kdi_name>
dBUG> store
```

where `<kdi_name>` is the name of the BlueCat Linux image.

The `store` command saves the default name for a BlueCat Linux image in the EEPROM. The default name of the BlueCat Linux image can be omitted in the `dn` command.

---

4. To boot a BlueCat Linux system enter the `go` command:

```
dBUG>go
loaded at: 00800000 00927A00
zimage at: 008057B0 008D0F01
initrd at: 008D5000 00927A00
avail ram: 00400000 00800000

Linux/PPC load: hda=bswap hdb=bswap hdc=bswap hdd=bswap root=101
Uncompressing Linux...done.
Now booting the kernel
Linux version 2.6.7 (bin@build2) (gcc version 3.2.2) #3 Fri Dec 24
13:23:42 UTC4
On node 0 totalpages: 16384
DMA zone: 16384 pages, LIFO batch:4
Normal zone: 0 pages, LIFO batch:1
```

```
HighMem zone: 0 pages, LIFO batch:1
Built 1 zonelists
Kernel command line: hda=bswap hdb=bswap hdc=bswap hdd=bswap root=101
PID hash table entries: 512 (order 9: 4096 bytes)
Memory: 62528k available (1452k kernel code, 524k data, 72k init, 0k
highmem)
Calibrating delay loop... 263.16 BogoMIPS
Dentry cache hash table entries: 8192 (order: 3, 32768 bytes)
Inode-cache hash table entries: 4096 (order: 2, 16384 bytes)
Mount-cache hash table entries: 512 (order: 0, 4096 bytes)
NET: Registered protocol family 16
PCI: Probing PCI hardware
Initializing Cryptographic API
Serial: MPC52xx PSC driver
ttyS0 at MMIO 0xf0002000 (irq = 39) is a MPC52xx PSC
RAMDISK driver initialized: 16 RAM disks of 8192K size 1024 blocksize
eth0: Phy @ 0x0, type LXT971 (0x001378e2)
NET: Registered protocol family 2
IP: routing cache hash table of 512 buckets, 4Kbytes
TCP: Hash tables configured (established 4096 bind 8192)
NET: Registered protocol family 17
RAMDISK: Compressed image found at block 9044
Freeing BlueCat RFS memory: 330k freed
VFS: Mounted root (ext2 filesystem).
Freeing unused kernel memory: 72k init
BlueCat Loader Shell
>
```

## Booting a BlueCat Linux System from a Network using the OS Loader

To boot the `showcase` demo system over a network using the BlueCat Linux OS loader, perform the following steps:

1. Copy the `showcase.kernel` and `showcase.rfs` files from the `$BLUECAT_PREFIX/demo/showcase` directory to the `/tftpboot` directory on the cross-development host.
2. Boot the OS loader as described in “Booting a BlueCat Linux System from a Network using the Motorola dBUG Firmware” on page 10.
3. At the BlueCat Linux OS loader prompt (`>`), enter the following commands:

```
> set IF eth0
> set IP <target_board_IP>
> set HOST <development_host_IP>
> set KERNEL tftp showcase.kernel
> set RFS tftp showcase.rfs
> set CMD ramdisk_size=28472
> boot
```

where *<target\_board\_IP>* is the IP address of the target and  
*<development\_host\_IP>* is the IP address of the development host.

These commands load the `showcase` demo system from a network onto the target board and then automatically start it.



# Kernel Configuration Options

The lite5200 BSP comes with a default BlueCat Linux kernel. This kernel has a number of configuration options. This chapter details these options in the tables listed in Table 3-1: “BlueCat Linux Default Configuration for the lite5200 BSP Distribution” below. Boldfaced entries in the tables represent subordinate menus. Italicized entries represent comments.

**Table 3-1: BlueCat Linux Default Configuration for the lite5200 BSP Distribution**

Table Number and Configuration Parameter
Table 3-2: “Code Maturity Level Options”
Table 3-3: “General Setup”
Table 3-4: “Loadable Module Support”
Table 3-5: “Processor”
Table 3-6: “Platform Options”
Table 3-7: “Bus Options”
Table 3-8: “Advanced Setup”
Table 3-9: “Device Drivers”
Table 3-10: “File Systems”
Table 3-11: “MPC52xx CPM Options”
Table 3-12: “Library Routines”
Table 3-13: “Kernel Hacking”
Table 3-14: “Security Options”
Table 3-15: “Cryptographic Options”

**Table 3-2: Code Maturity Level Options**

Description	Setting
Prompt for development and/or incomplete code/drivers	Y
Select only drivers expected to compile cleanly	Y
Select only drivers that don't need compile-time external firmware	Y

**Table 3-3: General Setup**

Description	Setting
Support for paging of anonymous memory	Y
System V IPC	Y
POSIX message queues	is not set
BlueCat Linux OS loader support	is not set
BlueCat Linux ignore printk	is not set
Memory sizing benchmarks	is not set
BSD process accounting	is not set
Sysctl support	Y
Auditing support	is not set
Kernel log buffer size (16 => 64KB, 17 => 128KB)	17
Support for hot-pluggable devices	is not set
Kernel <code>.config</code> support	is not set
<b>Configure standard kernel features (for small systems)</b>	Y
<i>--- Configure standard kernel features (for small systems).</i>	
Load all symbols for debugging/ <code>kksymoops</code>	Y
Include all symbols in <code>kallsyms</code>	is not set
Enable futex support	Y
Enable eventpoll support	Y

**Table 3-3: General Setup (Continued)**

Description	Setting
No-op I/O scheduler	Y
Anticipatory I/O scheduler	Y
Deadline I/O scheduler	Y
CFQ I/O scheduler	Y
<b>CODETEST device driver configuration</b>	is not set
Optimize for size	is not set

**Table 3-4: Loadable Module Support**

Description	Setting
Enable loadable module support	Y
Module unloading	Y
Forced module unloading	is not set
Module versioning support (Experimental)	Y
Automatic kernel module loading	is not set

**Table 3-5: Processor**

Description	Setting
<b>Processor Type</b>	
6xx/7xx/74xx/8260	Y
40x	is not set
44x	is not set
POWER3	is not set
POWER4 and 970 (G5)	is not set
8xx	is not set

**Table 3-5: Processor (Continued)**

Description	Setting
AltiVec support	is not set
Thermal management support	is not set
CPU frequency scaling	is not set
IRQ to user delivery	Y

**Table 3-6: Platform Options**

Description	Setting
<b>Machine Type</b>	
CHRP/Power Mac/PReP	is not set
Amiga-APUS	is not set
Cogent-Willow	is not set
Force-PowerCore	is not set
Force-PowerPMC250	is not set
Galileo-EV-64260-BP	is not set
IBM-Spruce	is not set
Motorola-LoPEC	is not set
Motorola-MCPN765	is not set
Motorola-MVME5100	is not set
Motorola-PowerPlus	is not set
Motorola-PrPMC750	is not set
Motorola-PrPMC800	is not set
Motorola-Sandpoint	is not set
SBS-Adirondack	is not set
SBS-K2	is not set
SBS-Palomar4	is not set
Synergy-Gemini	is not set

**Table 3-6: Platform Options (Continued)**

Description	Setting
EST8260	is not set
SBC82xx	is not set
SBS8260	is not set
RPXSUPER	is not set
TQM8260	is not set
Freescale LITE5200/(IceCube)	Y
High resolution timer support	is not set
Symmetric multiprocessing support	is not set
Preemptible kernel	Y
High memory support	is not set
Kernel support for ELF binaries	Y
Kernel support for MISC binaries	is not set
Default boot loader kernel arguments	Y
Initial kernel command string	console=ttyS0, 9600 panic=25

**Table 3-7: Bus Options**

Description	Setting
Legacy <code>/proc/pci</code> interface	is not set
PCI device name database	is not set

**Table 3-8: Advanced Setup**

Description	Setting
Prompt for advanced kernel configuration options	is not set
<i>--- Default settings for advanced configuration options are used.</i>	

**Table 3-9: Device Drivers**

Description	Settings
<b>Generic Driver Options</b>	
Driver Core verbose debug messages	is not set
<b>Memory Technology Devices</b>	
Memory Technology Device (MTD) support	Y
Debugging	is not set
MTD partitioning support	Y
MTD concatenating support	is not set
RedBoot partition table parsing	is not set
Command line partition table parsing	is not set
<i>--- User Modules And Translation Layers</i>	
Direct char device access to MTD devices	Y
Caching block device access to MTD devices	Y
Flash Translation Layer (FTL) support	is not set
NAND Flash Translation Layer (NFTL) support	is not set
Inverse NAND Flash Translation Layer (INFTL) support	is not set
<b>RAM/ROM/Flash Chip Drivers</b>	
Detect Flash chips by Common Flash Interface (CFI) probe	Y
Detect non-CFI AMD/JEDEC-compatible Flash chips	is not set
Flash chip driver advanced configuration options	is not set

**Table 3-9: Device Drivers (Continued)**

<b>Description</b>	<b>Settings</b>
Support for Intel/Sharp Flash chips	is not set
Support for AMD/Fujitsu Flash chips	Y
Support for ST (Advanced Architecture) Flash chips	is not set
Support for RAM chips in bus mapping	is not set
Support for ROM chips in bus mapping	is not set
Support for absent chips in bus mapping	is not set
Older (theoretically obsoleted now) drivers for non-CFI chips	is not set
<b>Mapping Drivers for Chip Access</b>	
Support nonlinear mappings of Flash chips	Y
CFI Flash device in physical memory map	is not set
CFI Flash device mapped for Motorola Lite5200	Y
Total Flash size	0x01000000
PCI MTD driver	is not set
<b>Self-Contained MTD Device Drivers</b>	
Ramix PMC551 PCI Mezzanine RAM card support	is not set
Uncached system RAM	is not set
Test driver using RAM	is not set
MTD emulation using block device	is not set
<i>--- Disk-On-Chip Device Drivers</i>	
M-Systems Disk-On-Chip 2000 and Millennium	is not set
M-Systems Disk-On-Chip Millennium-only alternative driver	is not set
M-Systems Disk-On-Chip Millennium Plus	is not set
<b>NAND Flash Device Drivers</b>	
NAND device support	is not set
<b>Parallel Port Support</b>	
Parallel port support	is not set

**Table 3-9: Device Drivers (Continued)**

Description	Settings
<b>Plug and Play support</b>	is not set
<b>Block Devices</b>	is not set
Normal floppy disk support	is not set
Compaq SMART-2 support	is not set
Compaq Smart Array 5xxx support	is not set
Mylex DAC960/DAC1100 PCI RAID Controller support	is not set
Micro Memory MM5415 Battery Backed RAM support (Experimental)	is not set
Loopback device support	is not set
Network block device support	is not set
Promise SATA SX8 (carmel) support	is not set
RAM disk support	Y
Default RAM disk size	8192
Initial RAM disk ( <code>initrd</code> ) support	is not set
BlueCat Linux RFS support	Y
Support for Large Block Devices	is not set
<b>ATA/ATAPI/MFM/RLL support</b>	
ATA/ATAPI/MFM/RLL support	is not set
<b>SCSI device support</b>	
SCSI device support	is not set
<b>Multiple device support (RAID and LVM)</b>	
Multiple devices driver support (RAID and LVM)	is not set
<b>Fusion MPT device support</b>	is not set
<b>IEEE 1394 (FireWire) support</b>	
IEEE 1394 (FireWire) support	is not set
<b>I2O device support</b>	

**Table 3-9: Device Drivers (Continued)**

Description	Settings
I2O support	is not set
<b>Macintosh device drivers</b>	is not set
<b>CAN support</b>	
Controller Area Network (CAN) support	Y
MSCAN support	Y
Debugging support for MSCAN driver	is not set
<b>Networking Support</b>	
Networking support	Y
<b>Networking options</b>	
Packet socket	Y
Packet socket: mmaped I/O	is not set
Netlink device emulation	is not set
UNIX domain sockets	Y
PF_KEY sockets	is not set
TCP/IP networking	Y
IP: multicasting	is not set
IP: advanced router	is not set
IP: kernel level autoconfiguration	is not set
IP: tunneling	is not set
IP: GRE tunnels over IP	is not set
IP: ARP daemon support (Experimental)	is not set
IP: TCP syncookie support (disabled per default)	is not set
IP: AH transformation	is not set
IP: ESP transformation	is not set
IP: IPComp transformation	is not set
The IPv6 protocol (Experimental)	is not set

**Table 3-9: Device Drivers (Continued)**

Description	Settings
<b>Network packet filtering (replaces ipchains)</b>	is not set
<b>SCTP Configuration (Experimental)</b>	
The SCTP protocol (Experimental)	is not set
Asynchronous Transfer Mode (ATM)	is not set
802.1d Ethernet Bridging	is not set
802.1Q VLAN support	is not set
DECnet support	is not set
ANSI/IEEE 802.2 LLC type 2 support	is not set
The IPX protocol	is not set
Appletalk protocol support	is not set
CCITT X.25 Packet Layer (Experimental)	is not set
LAPB Data Link Driver (Experimental)	is not set
Frame Diverter (Experimental)	is not set
Acorn Econet/AUN protocols (Experimental)	is not set
WAN router	is not set
Fast switching (read help!)	is not set
Forwarding between high speed interfaces	is not set
<b>QoS and/or fair queueing</b>	
QoS and/or fair queueing	is not set
<b>Network testing</b>	
Packet Generator (Use with Caution)	is not set
<b>Amateur Radio support</b>	
<b>IrDA (infrared) subsystem support</b>	
<b>Bluetooth subsystem support</b>	
Network device support	Y

**Table 3-9: Device Drivers (Continued)**

Description	Settings
Dummy net driver support	is not set
Bonding driver support	is not set
EQL (serial line load balancing) support	is not set
Universal TUN/TAP device driver support	is not set
<b>ARCnet devices</b>	
ARCnet support	is not set
<b>IBM On-chip net device</b>	
<b>Ethernet (10 or 100Mbit)</b>	
Ethernet (10 or 100Mbit)	Y
Generic Media Independent Interface device support	is not set
National DP83902AV (Oak Ethernet) support	is not set
Sun Happy Meal 10/100baseT support	is not set
Sun GEM support	Y
3Com cards	is not set
<b>Tulip family network device support</b>	
“Tulip” family network device support	is not set
HP 10/100VG PCLAN (ISA, EISA, PCI) support	is not set
EISA, VLB, PCI, and on-board controllers	is not set
<b>Ethernet (1000 Mbit)</b>	
Alteon AceNIC/3Com 3C985/NetGear GA620 Gigabit support	is not set
D-Link DL2000-based Gigabit Ethernet support	is not set
Intel PRO/1000 Gigabit Ethernet support	is not set
National Semiconductor DP83820 support	is not set
Packet Engines Hamachi GNIC-II support	is not set
Packet Engines Yellowfin Gigabit-NIC support (Experimental)	is not set
Realtek 8169 Gigabit Ethernet support	is not set

**Table 3-9: Device Drivers (Continued)**

Description	Settings
Marvell Yukon Chipset/SysKonnect SK-98xx support	is not set
Broadcom Tigon3 support	is not set
<b>Ethernet (10000 Mbit)</b>	
Intel PRO/10GbE support	is not set
S2IO 10Gbe XFrame NIC	is not set
<b>Token Ring devices</b>	
Token Ring driver support	is not set
<b>Wireless LAN (non-ham radio)</b>	
Wireless LAN drivers (non-ham radio) and wireless extensions	is not set
<b>WAN interfaces</b>	
WAN interfaces support	is not set
Fiber Distributed Data Interface (FDDI) driver support	is not set
High Performance Parallel Interface (HIPPI) driver support (Experimental)	is not set
Point-to-Point Protocol (PPP) support	is not set
Serial Line Internet Protocol (SLIP) support	is not set
Traffic Shaper (Experimental)	is not set
Network console logging support (Experimental)	is not set
<b>ISDN subsystem</b>	
ISDN support	is not set
<b>Telephony Support</b>	
Linux telephony support	is not set
<b>Input device support</b>	
Input devices (needed for keyboard, mouse, ...)	is not set
<i>---Userland interfaces</i>	
<i>--- Input I/O drivers</i>	

**Table 3-9: Device Drivers (Continued)**

Description	Settings
Gameport support	is not set
Serial I/O support	is not set
i8042 PC keyboard controller	is not set
<i>--- Input Device Drivers</i>	
<b>Character Devices</b>	
Virtual terminal	is not set
Nonstandard serial port support	is not set
<b>Serial drivers</b>	
8250/16550 and compatible serial support	is not set
<i>--- Non-8250 serial port support</i>	
Freescale MPC52xx family PSC serial support	Y
Console on a Freescale MPC52xx family PSC serial port	Y
Freescale MPC52xx family PSC serial port baud	9600
Unix98 PTY support	Y
Legacy (BSD) PTY support	Y
Maximum number of legacy PTY in use	256
QIC-02 tape support	is not set
<b>IPMI</b>	
IPMI top-level message handler	is not set
<b>Watchdog Cards</b>	
Watchdog Timer support	is not set
/dev/nvram support	is not set
Generic /dev/rtc emulation	is not set
DoubleTalk PC internal speech card support	is not set
Siemens R3964 line discipline	is not set
Applicom intelligent fieldbus card support	is not set

**Table 3-9: Device Drivers (Continued)**

Description	Settings
<b>Ftape, the floppy tape device driver</b>	
Ftape (QIC-80/Travan) support	is not set
/dev/agpgart (AGP support)	is not set
Direct Rendering Manager (XFree86 4.1.0 and higher DRI support)	is not set
RAW driver (/dev/raw/rawN) (Obsolete)	is not set
<b>I2C support</b>	
I2C support	Y
I2C device interface	is not set
<b>I2C Algorithms</b>	
I2C bit-banging interfaces	is not set
I2C PCF 8584 interfaces	is not set
<b>I2C Hardware Bus support</b>	
ALI 1535	is not set
ALI 1563	is not set
ALI 15x3	is not set
AMD 756/766	is not set
AMD 8111	is not set
Intel 801	is not set
Intel 810/815	is not set
ISA bus support	is not set
NVIDIA Nforce2	is not set
Parallel port adapter (light)	is not set
Intel PIIX4	is not set
S3/VIA (Pro)Savage	is not set
S3 Savage 4	is not set
National Semiconductor SCx200 ACCESS.bus	is not set

**Table 3-9: Device Drivers (Continued)**

Description	Settings
SiS 5595	is not set
SiS 630/730	is not set
SiS 96x	is not set
VIA 82C586B	is not set
VIA 82C596/82C686/823x	is not set
Voodoo 3	is not set
MPC107/824x/85xx/52xx	Y
<b>Hardware Sensors Chip support</b>	
Analog Devices ADM1021 and compatibles	is not set
ASUS ASB100 Bach	is not set
Dallas Semiconductor DS1621 and DS1625	is not set
FSC Hermes	is not set
Genesys Logic GL518SM	is not set
ITE IT87xx and compatibles	is not set
National Semiconductor LM75 and compatibles	is not set
National Semiconductor LM78 and compatibles	is not set
National Semiconductor LM80	is not set
National Semiconductor LM83	is not set
National Semiconductor LM85 and compatibles	is not set
National Semiconductor LM90 and compatibles	is not set
Maxim MAX1619 sensor chip	is not set
VIA686A	is not set
Winbond W83781D, W83782D, W83783S, W83627HF, ASUS AS99127F	is not set
Winbond W83L785TS-S	is not set
Winbond W83627HF, W83627THF, W83637HF, W83697HF	is not set
<b>Other I2C Chip support</b>	

**Table 3-9: Device Drivers (Continued)**

Description	Settings
EEPROM reader	Y
Philips PCF8574 and PCF8574A	is not set
Philips PCF8591	is not set
Epson 8564 RTC chip	is not set
I2C Core debugging messages	is not set
I2C Algorithm debugging messages	is not set
I2C Bus debugging messages	is not set
I2C Chip debugging messages	is not set
<b>Misc devices</b>	is not set
<b>Multimedia devices</b>	is not set
Video for Linux	is not set
<b>Digital Video Broadcasting Devices</b>	
DVB for Linux	is not set
<b>Graphics support</b>	
Support for frame buffer devices	is not set
<b>Sound</b>	
Sound card support	is not set
<b>USB support</b>	
Support for host-side USB	Y
USB verbose debug messages	is not set
--- <i>Miscellaneous USB options</i>	
USB device file system	is not set
Enforce USB bandwidth allocation (Experimental)	is not set
Dynamic USB minor allocation (Experimental)	is not set
--- <i>USB Host Controller Drivers</i>	
EHCI HCD (USB 2.0) support	is not set

**Table 3-9: Device Drivers (Continued)**

Description	Settings
OHCI HCD support	Y
UHCI HCD (most Intel and VIA) support	is not set
<i>--- USB Device Class drivers</i>	
USB Bluetooth TTY support	is not set
USB Modem (CDC ACM) support	is not set
USB Printer support	is not set
USB Mass Storage support	is not set
USB Human Interface Devices (HID)	is not set
USB Human Interface Device (full HID) support	is not set
<i>--- Input core support is needed for USB HID input layer or HIDBP support.</i>	
<b>USB HID Boot Protocol drivers</b>	
<i>--- USB Imaging devices</i>	
USB Mustek MDC800 Digital Camera support (Experimental)	is not set
<i>--- USB Multimedia devices</i>	
DABUSB driver	is not set
<i>--- Video4Linux support is needed for USB Multimedia device support.</i>	
<i>--- USB Network adaptors</i>	
USB CATC NetMate-based Ethernet device support (Experimental)	is not set
USB KLSI KL5USB101-based Ethernet device support	is not set
USB Pegasus/Pegasus-II-based Ethernet device support	is not set
USB RTL8150-based Ethernet device support (Experimental)	is not set
Multipurpose USB Networking Framework	is not set
<i>--- USB port drivers</i>	
<b>USB Serial Converter support</b>	
USB Serial Converter support	is not set

**Table 3-9: Device Drivers (Continued)**

Description	Settings
<i>--- USB Miscellaneous drivers</i>	
EMI 6 2m USB Audio interface support	is not set
EMI 2 6 USB Audio interface support	is not set
Texas Instruments Graph Link USB (aka SilverLink) cable support	is not set
USB Auerswald ISDN support (Experimental)	is not set
USB Diamond Rio500 support (Experimental)	is not set
USB Lego Infrared Tower support (Experimental)	is not set
USB LCD driver support	is not set
USB LED driver support	is not set
Cypress USB thermometer driver support	is not set
USB PhidgetServo support	is not set
<b>USB Gadget Support</b>	
Support for USB Gadgets	is not set

**Table 3-10: File Systems**

Description	Setting
Second extended file system support	Y
Ext2 extended attributes	Y
Ext2 POSIX access control lists	is not set
Ext2 security labels	is not set
Ext3 journalling file system support	is not set
Reiserfs support	is not set
JFS file system support	is not set
XFS file system support	is not set
Minix file system support	is not set
ROM file system support	is not set

**Table 3-10: File Systems (Continued)**

Description	Setting
Quota support	is not set
Kernel automounter support	is not set
Kernel automounter version 4 support (also supports v3)	is not set
<b>CD-ROM/DVD File Systems</b>	
ISO 9660 CD-ROM file system support	is not set
UDF file system support	is not set
<b>DOS/FAT/NT File Systems</b>	
DOS FAT file system support	is not set
NTFS file system support	is not set
<b>Pseudo File Systems</b>	
/proc file system support	Y
sysfs file system support	Y
/dev file system support (Obsolete)	is not set
/dev/pts Extended Attributes	is not set
Virtual memory file system support (former shm file system)	is not set
<b>Miscellaneous File Systems</b>	
ADFS file system support (Experimental)	is not set
Amiga FFS file system support (Experimental)	is not set
Apple Macintosh file system support (Experimental)	is not set
Apple Extended HFS file system support	is not set
BeOS file system (BeFS) support (read-only) (Experimental)	is not set
BFS file system support (Experimental)	is not set
EFS file system support (read-only) (Experimental)	is not set
Journalling Flash File System (JFFS) support	Y
JFFS debugging verbosity (0 = quiet, 3 = noisy)	0
Journalling Flash File System v2 (JFFS2) support	Y

**Table 3-10: File Systems (Continued)**

Description	Setting
JFFS2 debugging verbosity (0 = quiet, 2 = noisy)	0
JFFS2 support for NAND Flash (Experimental)	is not set
Compressed ROM file system support	is not set
FreeVxFS file system support (VERITAS VxFS™-compatible)	is not set
OS/2 HPFS file system support	is not set
QNX4 file system support (read-only)	is not set
System V/Xenix/V7/Coherent file system support	is not set
UFS file system support (read-only)	is not set
<b>Network File Systems</b>	
NFS file system support	Y
Provide NFSv3 client support	Y
Provide NFSv4 client support (Experimental)	is not set
Allow direct I/O on NFS files (Experimental)	is not set
NFS server support	is not set
Secure RPC: Kerberos V mechanism (Experimental)	is not set
SMB file system support (to mount Windows shares, etc.)	is not set
CIFS support (advanced network file system for Samba, Windows, and other CIFS-compliant servers)	is not set
NCP file system support (to mount NetWare volumes)	is not set
Coda file system support (advanced network file system)	is not set
Andrew File System (AFS) support (Experimental)	is not set
<b>Partition Types</b>	
Advanced partition selection	is not set
<b>Native Language Support</b>	
Base native language support	is not set

---

**Table 3-11: MPC52xx CPM Options**

Description	Setting
<i>--- Use USE Motorola BestComm API Implementation.</i>	
MPC52xx FEC Ethernet	Y
Use MDIO for PHY configuration	Y
Generic PHY support	Y
Support for LXT971 PHY	Y
Support for MPC52xx GPIO control	Y

**Table 3-12: Library Routines**

Description	Setting
CRC32 functions	Y
CRC32c (Castagnoli et al.) Cyclic Redundancy-Check	is not set

**Table 3-13: Kernel Hacking**

Description	Setting
Kernel debugging	Y
Debug memory allocations	is not set
Magic SysRq key	is not set
Spinlock debugging	is not set
Sleep-inside-spinlock checking	is not set
Include kgdb kernel debugger	is not set
BlueCat Linux kernel debugger	is not set
Include xmon kernel debugger	is not set
Include BDI-2000 user context switcher	is not set

**Table 3-13: Kernel Hacking (Continued)**

Description	Setting
Compile the kernel with debug info	is not set
Support for early boot texts over serial port	is not set

**Table 3-14: Security Options**

Description	Setting
Enable different security models	is not set

**Table 3-15: Cryptographic Options**

Description	Setting
Cryptographic API	is not set

This chapter provides information about BlueCat Linux demo systems supported by the lite5200 BSP.

## Demo Systems

Table 4-1 lists the demo systems supported in the lite5200 BSP distribution, the boot devices supported by each demo system, and their respective RAM and ROM requirements.

**Table 4-1: Demo Systems Supported by lite5200 BSP**

Demo	Boot Devices Supported by Default	ROM Requirements	RAM Requirements
developer	Network (using Motorola dBUG firmware) Network (using BlueCat Linux OS loader) ROM/Flash (using Motorola dBUG firmware) ROM/Flash (using BlueCat Linux OS loader)	5266.5 KB	24144 KB
osloader	Network (using Motorola dBUG firmware) ROM/Flash (using Motorola dBUG firmware)	1202.5 KB	5690 KB
showcase	Network (using Motorola dBUG firmware) Network (using BlueCat Linux OS loader) ROM/Flash (using Motorola dBUG firmware) ROM/Flash (using BlueCat Linux OS loader)	3915.0 KB	15217 KB

### developer Demo System

The `developer` demo system is a package consisting of the functionalities of `shell`, `ftp`, `ping`, and `gdb`. For descriptions of `developer` and its

components, refer to Chapter 4, “BlueCat Linux Demo Systems” in the *BlueCat Linux User’s Guide*.

### **osloader Demo System**

`osloader` is the BlueCat Linux OS loader system used to boot a BlueCat Linux system on the target board. Refer to Chapter 4, “BlueCat Linux Demo Systems” in the *BlueCat Linux User’s Guide* for details.

### **showcase Demo System**

The `showcase` demo system starts and configures the Apache HTTP daemon, turning the target board into a Web server. Refer to Chapter 4, “BlueCat Linux Demo Systems” in the *BlueCat Linux User’s Guide* for details.

---

## CHAPTER 5 *Supported Device Drivers*

Table 5-1 lists the device drivers supported by the lite5200 BSP and provides important information about them.

**Table 5-1: Device Drivers Supported by the lite5200 BSP**

Hardware Device	Device Drivers	Location in Source Tree	Kernel Configuration Options
MPC52xx PSC	<code>mpc52xx_uart.c</code>	<code>drivers/serial</code>	<code>CONFIG_SERIAL</code> <code>CONFIG_SERIAL_MPC52xx</code> <code>CONFIG_SERIAL_MPC52xx_CONSOLE</code>
Fast Ethernet Controller	<code>fec.c</code>	<code>arch/ppc/5xxx_io</code>	<code>CONFIG_FEC_ENET</code> <code>CONFIG_USE_MDIO</code> <code>CONFIG_FEC_LXT971</code>
BestComm DMA Controller	<code>sdma.c</code>	<code>arch/ppc/5xxx_io/bestcomm/</code>	<code>CONFIG_BESTCOMM_API</code>
CAN Controller	<code>*.c</code>	<code>pcan/drivers</code>	<code>CONFIG_CAN_BUS</code> <code>CONFIG_PCAN_MPC5200</code>
Flash Memory	<code>lite5200.c</code>	<code>drivers/mtd/maps</code>	<code>CONFIG_MTD</code> <code>CONFIG_MTD_PARTITIONS</code> <code>CONFIG_MTD_COMPLEX_MAPPING</code> <code>CONFIG_MTD_CFI</code> <code>CONFIG_MTD_CFI_AMDSTD</code> <code>CONFIG_MTD_LITE5200</code> <code>CONFIG_MTD_LITE5200_LEN</code>
I2C	<code>i2c-mpc.c</code>	<code>drivers/i2c/busses</code>	<code>CONFIG_I2C</code> <code>CONFIG_I2C_MPC</code>
USB	<code>ohci-*.c</code>	<code>drivers/usb/host</code>	<code>CONFIG_USB</code> <code>CONFIG_USB_OHCI_HCD</code>
User space IRQ delivery	<code>irq.c</code>	<code>arch/ppc</code>	<code>CONFIG_USER_IRQ</code> <code>CONFIG_SYSCTL</code>
GPIO	<code>gpio.c</code>	<code>arch/ppc/5xxx_io</code>	<code>CONFIG_GPIO</code>
PCI	<code>mpc52xx_pci.c</code> <code>mpc52xx_pci_ub.c</code>	<code>arch/ppc/syslib</code>	<code>CONFIG_PCI</code>

## The MPC5200 Lite5200 GPIO API

The BlueCat Linux kernel for the Lite5200 board provides a set of services to control the GPIO pins of the MPC5200 processor.

Table 5-2 describes a correspondence between the Linux driver constants and the Lite5200 GPIOs.

**Table 5-2: Correspondence Between the Linux Driver Constants and Lite5200 GPIOs**

Linux Driver Constants	Lite5200 GPIOs
MPC52xx_PIN_IR_USB_CLK	IR_USB_CLK LED D17
MPC52xx_PIN_PSC_1	PSC6_1 LED D16
MPC52xx_PIN_IR_TX	IRDA_TX LED D15
MPC52xx_PIN_PSC_0	PSC6_0 LED D14
MPC52xx_PIN_TIMER4	TIMER_4
MPC52xx_PIN_TIMER5	TIMER_5
MPC52xx_PIN_TIMER6	TIMER_6
MPC52xx_PIN_TIMER7	TIMER_7
MPC52xx_PIN_TIMER3	TIMER_3
MPC52xx_PIN_PSC3_0	PSC3_0
MPC52xx_PIN_PSC3_1	PSC3_1
MPC52xx_PIN_PSC3_2	PSC3_2
MPC52xx_PIN_PSC3_3	PSC3_3
MPC52xx_PIN_PSC3_4	PSC3_4
MPC52xx_PIN_PSC3_5	PSC3_5
MPC52xx_PIN_PSC3_6	PSC3_6
MPC52xx_PIN_PSC3_7	PSC3_7

**Table 5-2: Correspondence Between the Linux Driver Constants and Lite5200 GPIOs (Continued)**

Linux Driver Constants	Lite5200 GPIOs
MPC52xx_PIN_PSC3_8	PSC3_8
MPC52xx_PIN_PSC3_9	PSC3_9

To control a GPIO of the target, use the `mknod` command to create a special device node with major number 10 and minor number 223:

```
bash# mknod /dev/gpio c 10 223
```

An interface is implemented as a set of `ioctl()` commands for this device file. Each command takes the `mpc52xx_gpio_ctl` structure as a parameter:

```
struct mpc52xx_gpio_ctl {
    u32 pin;
    u32 value;
    u32 timeout;
};
```

The `pin` field is the pin number. The `value` field is `ioctl`-specific. The `timeout` field is the timeout in 1/1000 seconds. The `value` and `timeout` fields are not used in some `ioctl()` commands.

The `ioctl()` commands listed in the following table are defined.

**Table 5-3: Description of the `ioctl()` Command**

ioctl Function	Description
MPC52xx_GPIO_IOCTL_CONFIG	Enable or disable GPIO, and select I/O direction for GPIOs, whose direction can be configured. The value member of the argument structure should be <code>MPC52xx_GPIO_ENABLE_IN</code> , <code>MPC52xx_GPIO_ENABLE_OUT</code> , or <code>MPC52xx_GPIO_DISABLE</code> . To enable the GPIO for input, either enable the GPIO for output or disable the GPIO.
MPC52xx_GPIO_IOCTL_OUT	Set the output signal. The value member 0 is for 0 signal and 1 is for 1.

**Table 5-3: Description of the `iotctl()` Command (Continued)**

iotctl Function	Description
MPC52xx_GPIO_IOCTL_IN	Get the input signal. The <code>iotctl</code> result 0 is for 0 signal and 1 is for 1.
MPC52xx_GPIO_IOCTL_WAIT	<p>Wait for signal change on the pin. The <code>timeout</code> member is a timeout in msecs. The <code>value</code> field specifies a condition for signal change.</p> <p>The possible conditions are:</p> <ul style="list-style-type: none"> <li>• MPC52xx_GPIO_WAIT_ANY—Any input transition causes an event.</li> <li>• MPC52xx_GPIO_WAIT_RISING—An event occurs at input rising edge.</li> <li>• MPC52xx_GPIO_WAIT_FALLING—An event occurs at input falling edge.</li> <li>• MPC52xx_GPIO_WAIT_PULSE—An event occurs at any input pulse (that is, at 2nd input edge).</li> </ul> <p>The result is 0 on time-out or msecs remained to time-out if interrupt occurs.</p>

## User Space IRQ Delivery API

The BlueCat Linux kernel for the Lite5200 board provides the ability to handle the hardware interrupts in user space applications.

An interrupt is controlled via file operations on a pseudo-file in the `/proc` file system. What follows is the list of system calls to control hardware interrupts in the user space applications:

- `open (char * name, int flags, int mode)`

An `open` system call is used to request handling of a particular IRQ in the application. The `name` parameter is the filename `/proc/irq/<irq_num>/irq`, where `<irq_num>` is the IRQ number. The `flags` parameter must be `O_RDWR`, and the `mode` parameter must be 0.

The `result` value is a file descriptor associated with the interrupt.

- `read(int fd, void * buf, int size)`

A `read` system call enables the interrupt and makes the application wait until the next IRQ occurs. The `fd` parameter is the file descriptor obtained by the `open` call. The `buf` parameter is the pointer to the integer variable. The `size` parameter is 4.

The return value is 4 if interrupts occur. In this case, the supplied integer variable is the count of interrupts. If the UNIX signal occurs during the interrupt wait, the return value is negative, and the `errno` variable is set to `ERESTARTSYS`.

- `ioctl(int fd, int command, void * arg)`

An `ioctl` system call is an extension of the `read` system call. The `ioctl` returns the same set of parameters as the `read` call and additionally the timestamp of the interrupt. It is recommended that users use `ioctl` instead of the `read` system call. The `fd` parameter is the file descriptor obtained by the `open` call. The `command` parameter must be `USER_IRQ_WAIT_IOCTL`. The `arg` parameter is the pointer to the `user_irq_info` structure. The return value is 0 if interrupts occur. In this case, the `user_irq_info` structure is filled in as follows:

- The `count` field is the count of interrupts.
- The `real_time` field is the `REALTIME` clock's timestamp of the last interrupt.
- The `real_time` field is the `MONOTONIC` clock's timestamp of the last interrupt.

If the UNIX signal occurs, the return value is negative and the `errno` variable is set to `ERESTARTSYS`.

- `write(int fd, void * buf, int size)`

The `write` system call is used to enable/disable interrupts explicitly or to reset the interrupt counter.

The `fd` parameter is the file descriptor obtained by the `open` call. The `buf` parameter is the pointer to the integer variable. If the variable is negative, the interrupts will be disabled explicitly. If the variable is zero, the interrupts will be enabled explicitly. If the variable is positive, the value of the variable will be subtracted from the current interrupt counter. The `size` parameter is 4.

It is not recommended that users use the `write` system call during normal interrupts handling.

- `close(int fd)`

The `close` system call disables the interrupt and breaks the association of the interrupt with the file descriptor. The `fd` parameter is the file descriptor obtained by the `open` call.

This chapter describes known problems and limitations of this release.

---

## **Motorola MPC5200 Lite5200 Board Problems and Limitations**

The following are known problems and limitations of this release:

- Modification of the file system stored in a RAM disk does not persist across unmounting/mounting in BlueCat Linux 5.1. This limitation is due to a defect in the Linux kernel 2.6, described by official kernel maintainer Andrew Morton ([www.lkml.org](http://www.lkml.org)):

*“Because the kernel considers the ramdisk as being ‘memory backed’ it doesn’t do writeback into the blockdev pagecache. If you remove the memory-backed flag, ramdisk contributes to dirty memory in undesirable ways. That memory-backed flag is too overloaded and needs to be split up. It’s something I need to fix, but nobody seemed to be hurting from it up to now so I figured it could wait until after 2.6.0.”*

## User Documentation Updates

- Chapter 4, “BlueCat Linux Demo Systems” in the *BlueCat Linux User's Guide* provides incorrect values for the Storage and RAM requirements for the `developer` demo system. The correct requirements for `developer` are:
  - Storage: Medium
  - RAM: Large
- The “Booting from RAM using an NFS Server” section in Chapter 3, “Downloading and Booting BlueCat Linux” in the *BlueCat Linux User's Guide* contains incorrect commands. Please use the following commands to boot a BlueCat Linux system on the target board from an NFS server.

```
> set IP <target_IP_address>
> set HOST <host_IP_address>
> set IF <ethernet_interface>
> set KERNEL nfs /nfsboot <name>.kernel
> set NFS nfs /nfsboot <name>.rfs
> boot
```

- The “Mounting a Root File System from NFS” section in Chapter 3, “Downloading and Booting BlueCat Linux” in the *BlueCat Linux User's Guide* contains incorrect commands. Please use the following commands to boot a BlueCat Linux kernel that mounts an NFS-based file system as the root file system.

```
> set IP <target_IP_address>
> set HOST <host_IP_address>
> set IF <ethernet_interface>
> set KERNEL tftp /tftpboot/<name>.kernel
> set CMD console=ttyS1 root=/dev/nfs rw \
nfsroot=<host_IP_address>:/nfsboot \
ip=<target_IP_address>:<host_IP_address>:::::off \
panic=1
> boot
```

- The information in the *BlueCat Linux User's Guide* that states that `make xconfig` is supported on Linux hosts only is out of date.

`make xconfig` is now also supported on Windows hosts.

Ensure that Microsoft Visual Studio 6.0 or higher is installed on the cross-development host and that the Visual Studio environment tools

have been set up to allow invocation of the Microsoft Visual Studio tools in command line mode.

Then, to install the Qt library on the Windows host, go to [www.trolltech.com](http://www.trolltech.com) and download the Qt software for Microsoft Visual Studio C++ Windows users. To install the Qt library, follow the instructions provided with the Qt software.

`$BLUECAT_PREFIX/usr/src/linux/scripts/kconfig/Makefile` (the `Makefile` for the Windows host) uses the `QTLIBS` environment variable to list the Qt libraries that are needed to link with the `qconf` executable used to implement `make xconfig` on the Windows host. The following default definition is used:

```
QTLIBS = qt-mteval323.lib qtmain.lib
```

This definition specifies that libraries from the Qt 3.2.3 evaluation version for Windows are needed to link with the `qconf` executable.

If the Qt version installed on the host differs from the Qt 3.2.3 evaluation version, the `QTLIBS` definition must be changed to specify the correct list of libraries. This can be done either by manually editing `$BLUECAT_PREFIX/usr/src/linux/scripts/kconfig/Makefile` to modify the `QTLIBS` definition or by defining the `QTLIBS` environment variable using the **Properties->Advanced->Environment Variables** wizard in the context menu of the **My Computer** icon on the Windows desktop. The second approach allows the user to avoid changing the `Makefile` every time BlueCat Linux is reinstalled.

