

BlueCat Linux Target Support Guide

DOC-427-01

For Intel StrongARM SA-1110 Boards

Product names mentioned in the *BlueCat Linux Target Support Guide for Intel StrongARM SA-1110 Boards* are trademarks of their respective manufacturers and are used here for identification purposes only.

Copyright ©1987-2001, 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 Target Support Guide for Intel StrongARM SA-1110 Boards* 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

CHAPTER 1	OVERVIEW	1
CHAPTER 2	DOWNLOADING AND BOOTING BLUECAT LINUX ON THE TARGET	3
	Prerequisites	3
	Downloading and Booting Overview	4
	Setting up Hardware	5
	Obtaining and Installing the Jflash Utility	6
	Obtaining Jflash	7
	Installing Jflash on the Host	7
	Linux Host	7
	Windows Host.	8
	Using i_osloader to Download a Demo System	9
	Downloading i_osloader	9
	Downloading showcase into Target Flash Memory	11
	Booting Options	12
	Booting a Demo System from Flash Memory	13
	Booting a Demo System over a Network	14
CHAPTER 3	KERNEL CONFIGURATION OPTIONS	17
CHAPTER 4	SUPPORTED DEMO SYSTEMS	29
	Demo Systems	29

developer Demo System	29
osloader Demo System	30
showcase Demo System	30
Downloading and Booting showcase with an FFS Root Filesystem	32
Modifying Target Board IP and Gateway Addresses	33

CHAPTER 5	SUPPORTED DEVICE DRIVERS	35
------------------	---------------------------------------	-----------

This *Target Support Guide* (TSG) provides information about the BlueCat Linux Target Support Package (TSP) for Intel StrongARM SA-1110 development boards.

Throughout this Target Support Guide (TSG), the TSP is referred to as the “SA-1110,” the board as the “SA-1110 development board,” or simply as the “target board.”

- **Chapter 1** is an overview of this TSG’s individual chapters.
- **Chapter 2** describes the BlueCat Linux downloading and booting procedures for the SA-1110 development board.
- **Chapter 3** provides configuration option information about the SA-1110 TSP’s default BlueCat Linux kernel.
- **Chapter 4** summarizes BlueCat Linux demo systems supported by the SA-1110 TSP.
- **Chapter 5** provides a list of SA-1110 TSP-supported device drivers, with important information about each of these.

Downloading and Booting BlueCat Linux on the Target

This chapter provides instructions for downloading a BlueCat Linux system from a cross development host into the SA-1110 development board's flash memory, and then booting the BlueCat Linux system on the target board.

Specifically, this chapter demonstrates downloading BlueCat Linux `i_osloader` onto the target board, which then can be used to download the `showcase` demo system. The same procedure can be used to download other custom embedded systems.

Prerequisites

This document is a guide to downloading and booting BlueCat Linux systems on the user's target board. Scenarios that use demo systems included in the BlueCat Linux distribution are presented. As such, the user is assumed to be familiar with the target board hardware and the manufacturer's documentation for it. The user must also have an understanding of system administration for the particular cross development host before BlueCat Linux and the SA-1110 TSP are installed.

Before installing and booting BlueCat Linux on an SA-1110 development board, it is assumed that the user has installed the default BlueCat Linux configuration and the SA-1110 Target Support Package (TSP) on the cross development host:

1. Install the BlueCat Linux ARM core on the cross development host, as detailed in the "Installing the Default Configuration" section in Chapter 1, "Installation" of the *BlueCat Linux User's Guide*

2. Install the SA-1110 TSP on the cross development host as detailed in the “Installing Support for Target Boards” section of Chapter 1, “Installation” in the *BlueCat Linux User’s Guide*
3. Activate support for the SA-1110 TSP as detailed in the “Activating Support for a Target Board” section of Chapter 1, “Installation” in the *BlueCat Linux User’s Guide* Use the following command:

```
$ . SETUP.sh assabet
```

Downloading and Booting Overview

The standard procedure for installing and booting a BlueCat Linux system on SA-1110 development boards consists of the following main steps:

- Obtaining and installing the `Jflash` utility provided by Intel
- Setting up hardware
- Downloading the BlueCat Linux `i_osloader` into the SA-1110 development board’s flash memory
- Booting an embedded system

These steps are performed sequentially:

1. Ensure that the CompactFlash Ethernet Communication card is installed on the target and connected to the cross development host. Ensure that the ISV Cable Base Station shipped with the target board is connected to the cross development host. See the “Using `i_osloader` to Download a Demo System.” section.
2. Obtain and install the `Jflash` utility. The `Jflash` utility is an SA-1110 flash memory programming utility provided by Intel. See “Setting up Hardware” on page 5 for a detailed description.
3. Download the BlueCat Linux `i_osloader` into the SA-1110 flash memory. See “Using `i_osloader` to Download a Demo System” on page 9 for a detailed description.
4. Use `i_osloader` to download the `showcase demo` system into target flash memory as a composite image.

5. Boot the embedded system from target flash memory or over a network using `i_osloader`.

The section “Booting Options” on page 12 provides examples of booting the `showcase` demo system supported in the SA-1110 TSP.

Setting up Hardware

Before downloading BlueCat Linux `i_osloader` into target flash memory, set up hardware as explained below and in Figure 2-1:

1. The SA-1110 development board and the cross development host need to be connected with the ISV Cable Base Station that is shipped with the board. The base station plug end of the ISV cable is connected to the DB9 serial and the DB25 parallel ports on the cross development host. The RJ-11 connector is not used.
2. Insert the Socket Communication CompactFlash Ethernet card (shipped with the SA-1110 board) into the development board and connect it to the cross development host.
3. Connect a headset/speaker system to J1 of the SA-1110 development board. This headset will be used to the MP3 demo application as part of the BlueCat Linux `showcase` demo system.

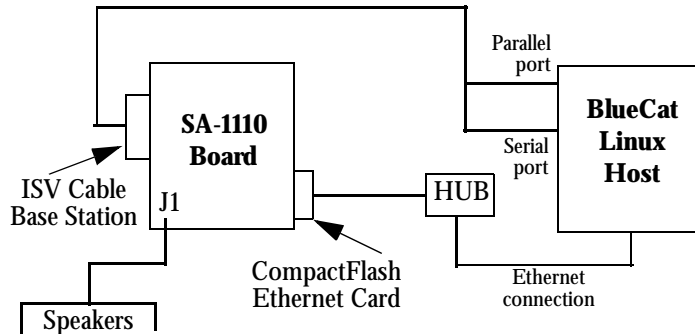


Figure 2-1: Hardware Connections

Obtaining and Installing the Jflash Utility

The `Jflash` utility provided by Intel is used to program a BlueCat Linux embedded system, such as `i_osloader`, into the SA-1110's flash memory. `Jflash` is used for programming on-board flash memory using a PC's parallel port. `Jflash` uses the existing cable shipped with the SA-1110 Microprocessor Development Board kit.

This software runs directly on Linux, Windows NT, and Windows 2000. If the `giveio.sys` device driver is installed, `Jflash` will also run on Windows NT and Windows 2000. Depending on the cross development host, the user can download `Jflash` for Windows or Linux from the respective internet site. See Table 2-1 for details.

Obtaining Jflash

Table 2-1 shows the internet sites from which the files necessary for installing the `Jflash` utility can be downloaded:

Table 2-1: Web Site Addresses

Host	Web Site Addresses
Linux	<code>ftp://ftp.arm.linux.org.uk/</code>
Windows	<code>http://developer.intel.com/</code> <code>ftp://ftp.freesoftware.com/</code>

Before installing `Jflash`, download the requisite files and place them in a directory on the hard disk.

Installing Jflash on the Host

The procedure for installing the `Jflash` utility is different for Windows and Linux cross development hosts:

Linux Host

Use the following procedure to install `Jflash` onto a Linux cross development host:

1. Unpack the downloaded file to a temporary directory, for example, by entering the following commands:

```
bash$ cd /tmp
```

```
bash$ tar xzf /path/Assabet-Jflash-v1.2-1.tgz
```

2. To build the executable image, enter the following commands:

```
bash$ cd Jflash-1.2-1
```

```
bash$ make -f Makefile.linux
```

NOTE: *The executable image must be built outside of the BlueCat Linux environment as the build process uses cross development host tools.*

3. Copy the executable image into the BlueCat Linux environment by running the following command:

```
bash$ cp Jflash-linux \  
/BlueCat_Linux_environment_path/cdt/bin/Jflash
```

Windows Host.

NOTE: *The name of the drive where Windows is installed is C: in this section. The user must replace all occurrences of C: by an actually used drive letter.*

Use the following procedure to install Jflash onto a Windows cross development host:

1. Run the SA1110JFlashKit_V12.exe file and follow the on-screen instructions. This file is a self-extracting archive file used to install the Jflash utility, and is started as an ordinary program.
2. To copy the Jflash utility to the BlueCat Linux environment enter the following commands:

```
bash$ cd "C:/Program Files/Intel \  
SA11X0 Software/JTAG Software"
```

3. Unzip the Jflash12.zip file. After the file has been unzipped, it will be seen to contain a file called Jflash.exe.
4. Copy Jflash.exe to the specified directory.

```
bash$ cp Jflash.exe \  
/BlueCat_Linux_environment_path/cdt/bin/
```

5. Install the low-level driver that permits Jflash.exe to access the parallel port. The SA-1110 development board is accessed by Jflash using the JTAG connector plugged to the parallel port.

```
bash$ cd "C:/Program Files/\  
Intel SA11X0 Software/JTAG Software"
```

```
bash$ ./instdrv giveio remove
```

```
bash$ ./instdrv giveio \  
'C:\Program Files\Intel SA11X0  
Software\JTAG Software\giveio.sys'
```

The `./instdrv giveio remove` command may print an error message indicating that the driver has not been previously installed. In case of occurrence of such an error, ignore it and proceed with the installation.

All commands must be executed as Administrator.

NOTE: *This step must be repeated after each restart of the host computer as the driver is not started automatically upon a reboot.*

Using i_osloader to Download a Demo System

This section and its subsections provide step-by-step detailed instructions for the process by which any supported demo system (e.g., `showcase`) or custom BlueCat Linux system can be downloaded into the target flash memory using `i_osloader`. Refer also to the *BlueCat Linux User's Guide* for information on downloading a BlueCat Linux system into target flash memory.

Downloading i_osloader

Use the following procedure to download `i_osloader` into the SA-1110 board's flash memory using the `Jflash` utility:

1. Power-up the SA-1110 development board.
2. Type the following commands in the BlueCat Linux environment to download `i_osloader` into target flash memory:

```
BlueCat:# cd \  
"$BLUECAT_PREFIX/demo/osloader"  
  
BlueCat:# Jflash i_osloader.kdi
```

3. After the `Jflash` utility been downloaded, reset the board by pressing the **Reset** button (S9). The following screen output appears:

```
Command line: root=101
Uncompressing Linux...done.
Now booting the kernel
Linux version 2.2.12-1 (root@bluecat) (gcc version
egcs-2.91.66 19990314/Linux 1
NetWinder Floating Point Emulator V0.94.1 (c) 1998
Corel Computer Corp.
Calibrating delay loop... 194.15 BogoMIPS
Memory: 30612k/32M available (872k code, 20k reserved,
508k data, 4k init)
DENTRY hash table entries: 2097152 (order: 12,
16777216 bytes)
Buffer-cache hash table entries: 32768 (order: 5,
131072 bytes)
Page-cache hash table entries: 8192 (order: 3, 32768
bytes)
POSIX conformance testing by UNIFIX
CompactFlash Card is detected
Linux NET4.0 for Linux 2.2
Based upon Swansea University Computer Society
NET3.039
NET4: Linux TCP/IP 1.0 for NET4.0
IP Protocols: ICMP, UDP, TCP
TCP: Hash tables configured (ehash 32768 bhash 32768)
Starting kswapd v 1.5
SA1100 serial driver version 4.26 with no serial
options enabled
ttyS00 at 0xf8010000 (irq = 15) is a SA1100 UART
ttyS01 at 0xf8050000 (irq = 17) is a SA1100 UART
RAM disk driver initialized: 16 RAM disks of 4096K
size
ne.c:v1.10 9/23/94 Donald Becker
(becker@cesdis.gsfc.nasa.gov)
NE*000 ethercard probe at 0x300: 00 c0 1b 00 bb 72
eth0: NE2000 found at 0x300, using IRQ 42.
RAMDISK: Compressed image found at block 3150484
Flash mapping for Assabet board initialized starting
at address 0xd0000000
Flash on the Assabet board: Found 2 x16 CFI devices at
location 0 in 16 bit mode
Assabet Flash MTD driver: Configuration of partitions
is :
Assabet Flash MTD driver: Configured 0 partitions
VFS: Mounted root (ext2 filesystem).
Freeing unused kernel memory: 4k init
BlueCat Loader Shell
>
```

Upon successful completion of the procedure, `i_osloader` will start from target flash memory automatically upon a board power-up/reset.

NOTE: *To enable access to the parallel port, the `Jflash` utility must be run as root on Linux or as Administrator on Windows NT/2000 (no Windows98-specific restrictions exist).*

Downloading showcase into Target Flash Memory

Use the following procedure to download the `showcase` demo system into the SA-1110 board's flash memory using `i_osloader` as a BlueCat Linux composite image composed of a kernel image and a compressed root filesystem image.

NOTE: *After instructions shown below have been completed, the `showcase demo` system will start upon a target board power-up/reset. To download and boot other embedded systems, the user needs to download `i_osloader` into target flash memory again.*

To install the `showcase` demo system into target flash memory as a BlueCat Linux image composed of the kernel and the compressed root filesystem image using `i_osloader`, use the following procedure:

1. Partition the target flash memory. Create a single partition located at the beginning of flash memory. For instance:

```
> exec flash_fdisk /dev/mtddchar0 0-15
```

This command creates a partition of size 4 MB at the beginning of target flash memory.

2. Set the environment variables to properly configure network access. For instance:

```
> set IP target_IP_address
```

```
> set HOST tftp_server_IP_address
```

```
> set IF eth0
```

3. Set the `FILE` environment variable so that it points to the `showcase` BlueCat Linux image. For instance:

Booting a Demo System from Flash Memory

At this point, the `showcase` demo system has successfully been downloaded into the target flash memory. It is then booted from flash as follows:

Reset the target board by entering the following command:

```
> reset
```

The following screen output appears while the target board is being reset. The boot sequence, when complete, validates that the demo system has booted from the target flash memory by displaying the `bash#` prompt:

```
Putting Restarting system.
Command line: ramdisk_size=32000 root=101
Uncompressing Linux...done.
Now booting the kernel
Linux version 2.2.12-1 (root@bluecat) (gcc version egcs-
2.91.66 19990314/Linux 1
NetWinder Floating Point Emulator V0.94.1 (c) 1998 Corel
Computer Corp.
Console: colour dummy device 80x30
Calibrating delay loop... 194.15 BogoMIPS
Memory: 29872k/32M available (896k code, 20k reserved, 524k
data, 16k init)
DENTRY hash table entries: 262144 (order: 9, 2097152 bytes)
Buffer-cache hash table entries: 32768 (order: 5, 131072
bytes)
Page-cache hash table entries: 8192 (order: 3, 32768 bytes)
POSIX conformance testing by UNIFIX
CompactFlash Card is detected
Linux NET4.0 for Linux 2.2
Based upon Swansea University Computer Society NET3.039
NET4: Linux TCP/IP 1.0 for NET4.0
IP Protocols: ICMP, UDP, TCP
TCP: Hash tables configured (ehash 32768 bhash 32768)
Starting kswapd v 1.5
Console: switching to colour frame buffer device 40x30
SA1100 serial driver version 4.26 with no serial options
enabled
ttyS00 at 0xf8010000 (irq = 15) is a SA1100 UART
ttyS01 at 0xf8050000 (irq = 17) is a SA1100 UART
UCB1200 generic module installed
ucb1200 touch screen driver initialized
```

```
UDA1341 audio driver initialized
RAM disk driver initialized: 16 RAM disks of 32000K size
ne.c:vl.10 9/23/94 Donald Becker (becker@cesdis.gsfc.nasa.gov)
NE*000 ethercard probe at 0x300: 00 c0 1b 00 bb 72
eth0: NE2000 found at 0x300, using IRQ 42.
RAMDISK: Compressed image found at block 3150508
Flash mapping for Assabet board initialized starting at
address 0xd0000000
Flash on the Assabet board: Found 2 x16 CFI devices at
location 0 in 16 bit mode
Assabet Flash MTD driver: Configuration of partitions is :
Assabet Flash MTD driver: Configured 0 partitions
VFS: Mounted root (ext2 filesystem).
Freeing unused kernel memory: 16k init
INIT: version 2.77 booting
INIT: Entering runlevel: 1
Network is configured as follows:

Target IP address: 216.100.252.144
Gateway IP address:

Apache server is started.

Microwindows demo is started.

bash#
```

At this point, `showcase` automatically boots from the on-board flash memory every time the board is reset by first pressing the Reset button (S9).

NOTE: *After being downloaded into target flash memory, the showcase demo system will start on a board power-up/reset. To boot other embedded systems the user needs to download `i_osloader` to target flash memory again.*

Booting a Demo System over a Network

To boot a demo system or a custom embedded system from a cross development host over a network connection using `i_osloader`, use the following procedure:

1. Download `i_osloader` into the target flash memory, as described under “Downloading `i_osloader`.”
2. Boot the BlueCat Linux demo system of choice on the cross development host via a network connection using `i_osloader` as detailed in the “Booting BlueCat Linux over Network or Parallel Port” section in Chapter 3, “Downloading and Booting BlueCat Linux” of the *BlueCat Linux User's Guide*

Kernel Configuration Options

The SA-1110 TSP comes with a default BlueCat Linux kernel. This kernel has a number of configuration options. These options are detailed in the tables listed in Table 3-1: Kernel Configuration Options below:

Table 3-1: Kernel Configuration Options

Configuration Option	Table and Page Numbers
System and Processor Type	Table 3-2, page 18
Code Maturity-Level Options	Table 3-3, page 18
Loadable Module Support	Table 3-4, page 19
General Setup	Table 3-5, page 19
Plug and Play Support	Table 3-6, page 20
Block Devices	Table 3-7, page 20
Character Devices	Table 3-8, page 20
Video for Linux	Table 3-9, page 21
Joystick Support	Table 3-10, page 21
Floppy Tape Device Driver, Ftape	Table 3-11, page 22
Console Drivers	Table 3-12, page 22
Networking Options	Table 3-13, page 23
Amateur Radio Support	Table 3-14, page 23
IrDA Subsystem	Table 3-15, page 24
Network Device Support	Table 3-16, page 24
ARCnet Devices	Table 3-17, page 24
Ethernet (10 or 100 Mbit)	Table 3-18, page 25

Table 3-1: Kernel Configuration Options (Continued)

Configuration Option	Table and Page Numbers
Token Ring Devices	Table 3-19, page 25
WAN Interfaces	Table 3-20, page 25
SCSI Support	Table 3-21, page 26
Sound	Table 3-22, page 26
Filesystems	Table 3-23, page 26
Network Filesystems	Table 3-24, page 27
Partition Types	Table 3-25, page 27
Kernel Hacking	Table 3-26, page 28
LinuxWorks Messenger Support	Table 3-27, page 28
Modular Advanced Power Management	Table 3-28, page 28

Table 3-2: System and Processor Type

Option	Value	Description
CONFIG_ARCH_SA1100	Y	ARM system type
CONFIG_CPU_SA1100	Y	SA1100 processor type of CPU
CONFIG_SA1100_ASSABET	Y	SA1100-based board type
CONFIG_CPU_32	Y	32-bit processor used
CONFIG_ARM7	Y	Optimizes for ARM7 CPU

Table 3-3: Code Maturity-Level Options

Option	Value	Description
CONFIG_EXPERIMENTAL	N	Prompts for development and/or incomplete code/drivers
CONFIG_ALIGNMENT_TRAP	N	Enables kernel-mode alignment trap handle (Experimental)
CONFIG_TEXT_SECTIONS	N	Splits text into discardable sections

Table 3-4: Loadable Module Support

Option	Value	Description
CONFIG_MODULES	Y	Enables loadable module support
CONFIG_MODVERSIONS	N	Sets version information on all symbols for modules
CONFIG_KMOD	N	Kernel module loader support

Table 3-5: General Setup

Option	Value	Description
CONFIG_NET	Y	Networking support
CONFIG_ANGELBOOT	N	Loads kernel using Angel Debug Monitor
CONFIG_BLUECAT_THUMB	N	BlueCat Linux kernel support for THUMB binaries
CONFIG_BLUECAT_LOADER	N	BlueCat Linux OS loader
CONFIG_BLUECAT_IGNORE_PRINTK	N	BlueCat Linux ignore <code>printk</code>
CONFIG_BLUECAT_SMALL_FOOTPRINT	Y	BlueCat Linux small memory footprint
CONFIG_BLUECAT_RAM_PRESERVE	1	Part of RAM preserved after loading the kernel
CONFIG_CF_ASSABET	Y	Supports CompactFlash devices on SA-1110
CONFIG_BLUECAT_MEMSIZE	N	Memory sizing benchmark
CONFIG_SYSVIPC	Y	System VIPC
CONFIG_BSD_PROCESS_ACCT	N	BSD Process Accounting
CONFIG_SYSCTL	Y	Systl support
CONFIG_NWFPE	Y	Math emulation
CONFIG_BINFMT_AOUT	N	Kernel support for <code>a.out</code> binaries
CONFIG_BINFMT_ELF	Y	Kernel support for ELF binaries
CONFIG_BINFMT_MISC	N	Kernel support for MISC binaries
CONFIG_ARTHUR	N	RISC OS personality
CONFIG_PARPORT	N	Parallel port support

Table 3-6: Plug and Play Support

Option	Value	Description
CONFIG_PNP	N	Plug and Play support

Table 3-7: Block Devices

Option	Value	Description
CONFIG_BLK_DEV_FD	N	Normal PC floppy disk support
CONFIG_BLK_DEV_IDE	N	Enhanced IDE/MFM/RLL disk/ CD-ROM/tape/floppy support
CONFIG_BLK_DEV_HD_ONLY	N	Old hard disk (MFM/RLL/IDE) driver
CONFIG_BLK_DEV_LOOP	Y	Loopback device support
CONFIG_BLK_DEV_NBD	N	Network block device support
CONFIG_BLK_DEV_MD	N	Multiple devices driver support
CONFIG_BLK_DEV_RAM	Y	RAM disk support
CONFIG_BLK_DEV_INITRD	N	Initial RAM disk (initrd) support
CONFIG_BLUECAT_RFS	Y	BlueCat Linux RFS support
CONFIG_BLK_DEV_GENERIC_FLASH_DOC	N	M-Systems DiskOnChip
CONFIG_BLK_DEV_XD	N	XT hard disk support
CONFIG_PARIDE_PARPORT	Y	Parallel port IDE device support
CONFIG_BLK_CPQ_DA	N	Compaq SMART2 support
CONFIG_BLK_DEV_HD	N	Old hard disk support

Table 3-8: Character Devices

Option	Value	Description
CONFIG_VT	Y	Virtual terminal
CONFIG_VT_CONSOLE	N	Support for console on virtual terminal

Table 3-8: Character Devices (Continued)

Option	Value	Description
CONFIG_SERIAL	N	Standard/generic (dumb) serial support
CONFIG_SERIAL_EXTENDED	N	Extended dumb serial driver options
CONFIG_SERIAL_SA1100	Y	SA1100 serial port support
CONFIG_SERIAL_SA1100_CONSOLE	Y	Console on SA1100 serial port
CONFIG_FL_SA1100	N	Support for SA1100 frontlight regulation
CONFIG_UCB1200	Y	Generic support for UCB1200-based devices
CONFIG_UCB1200_SA1100	Y	Support UCB1200-based devices on SA1100
CONFIG_TS_SA1100	Y	Support for Philips UCB1200 Touch Screen
CONFIG_SERIAL_NONSTANDARD	N	Non-standard serial port support
CONFIG_UNIX98_PTYS	N	UNIX98 PTY support
CONFIG_MOUSE	N	Mouse support (not serial mice)
CONFIG_QIC02_TAPE	N	QIC-02 tape support
CONFIG_WATCHDOG	N	Watchdog Timer Support
CONFIG_NVRAM	N	/dev/nvram support
CONFIG_RTC	Y	Enhanced Real Time Clock support
CONFIG_RTC_GENERIC	Y	Support for generic 32-bit Real Time Clocks
CONFIG_DTLK	N	Double Talk PC internal speech card support

Table 3-9: Video for Linux

Option	Value	Description
CONFIG_VIDEO_DEV	N	Video for Linux

Table 3-10: Joystick Support

Option	Value	Description
CONFIG_JOYSTICK	N	Joystick support

Table 3-11: Floppy Tape Device Driver, Ftape

Option	Value	Description
CONFIG_FTAPE	N	Ftape (QIC-80/Travan) support
CONFIG_FT_NORMAL_DEBUG	Not Set	Controls the amount of debugging output of ftape
CONFIG_FT_STD_FDC	Not Set	Floppy drive controller for ftape

Table 3-12: Console Drivers

Option	Value	Description
CONFIG_VGA_CONSOLE	N	VGA text console
CONFIG_FB	Y	Support for frame buffer devices (Experimental)
CONFIG_DUMMY_CONSOLE	Y	Dummy console net driver support
CONFIG_FB_SA1100	Y	SA1100 LCD display support
CONFIG_FB_VIRTUAL	N	Virtual Frame Buffer support (Only For Testing!)
CONFIG_FBCON_ADVANCED	N	Advanced low-level driver options
CONFIG_FBCON_MFB	Y	Monochrome support
CONFIG_FBCON_CFB2	Y	2 bpp packed pixels support
CONFIG_FBCON_CFB4	Y	4 bpp packed pixels support
CONFIG_FBCON_CFB8	Y	8 bpp packed pixels support
CONFIG_FBCON_CFB16	Y	16 bpp packed pixels support
CONFIG_FBCON_CFB24	Y	24 bpp packed pixels support
CONFIG_FBCON_CFB32	Y	32 bpp packed pixels support
CONFIG_FBCON_FONTWIDTH8_ONLY	N	Supports only 8 pixel-wide fonts
CONFIG_FBCON_FONTS	N	Other font support
CONFIG_FONT_8x8	Y	VGA 8x8 font support
CONFIG_FONT_8x16	Y	VGA 8x16 font support

Table 3-13: Networking Options

Option	Value	Description
CONFIG_PACKET	N	Packet socket
CONFIG_NETLINK	N	Kernel/User netlink socket
CONFIG_FIREWALL	N	Network firewalls
CONFIG_FILTER	N	Socket filtering
CONFIG_UNIX	N	UNIX domain sockets
CONFIG_INET	Y	TCP/IP networking
CONFIG_IP_MULTICAST	N	IP: Multicasting
CONFIG_IP_ADVANCED_ROUTER	N	IP: Advanced router
CONFIG_IP_PNP	N	IP: Kernel level auto-configuration
CONFIG_IP_ROUTER	N	IP: Optimizes as router not host
CONFIG_NET_IPIP	N	IP: Tunneling
CONFIG_NET_IPGRE	N	IP: GRE tunnels over IP
CONFIG_IP_ALIAS	N	IP: Aliasing support
CONFIG_SYN_COOKIES	N	IP: TCP syncookie support (not enabled per default)
CONFIG_INET_RARP	N	IP: Reverse ARP
CONFIG_SKB_LARGE	N	IP: Allows large windows (not recommended if <16 MB of memory)
CONFIG_IPX	N	IPX protocol
CONFIG_ATALK	N	Appletalk DDP

Table 3-14: Amateur Radio Support

Option	Value	Description
CONFIG_HAMRADIO	N	Amateur radio support

Table 3-15: IrDA Subsystem

Option	Value	Description
CONFIG_IRDA	N	IrDA subsystem support

Table 3-16: Network Device Support

Option	Value	Description
CONFIG_NETDEVICES	Y	Network device support
CONFIG_DUMMY	N	Dummy net driver support
CONFIG_EQUALIZER	N	EQL (serial line load balancing) support
CONFIG_NET_SB1000	N	General Instruments Surfboard 1000
CONFIG_FDDI	N	FDDI driver support
CONFIG_PPP	N	PPP (point-to-point) support
CONFIG_SLIP	N	SLIP (serial line) support
CONFIG_NET_RADIO	N	Wireless LAN (non-ham radio)
CONFIG_NET_FC	N	Fibre Channel driver support
CONFIG_SBNI	N	SBNI 12-xx support

Table 3-17: ARCnet Devices

Option	Value	Description
CONFIG_ARCNET	N	ARCnet support

Table 3-18: Ethernet (10 or 100 Mbit)

Option	Value	Description
CONFIG_NET_ETHERNET	Y	Ethernet (10 or 100 Mbit)
CONFIG_ASSABET_CFPLUS	Y	Socket CF+ CompactFlash Ethernet card support
CONFIG_ARM_AM79C961A	N	AM79C961A support
CONFIG_NET_VENDOR_3COM	N	3COM cards
CONFIG_LANCE	N	AMD LANCE and PCnet (AT 1500 and NE2100) support
CONFIG_NET_VENDOR_SMC	N	Western Digital/SMC cards
CONFIG_NET_VENDOR_RACAL	N	Racal-Interlan (micom) NI cards
CONFIG_NET_ISA	N	Other ISA cards
CONFIG_NET_EISA	N	EISA, VLB, PCI and on board controllers
CONFIG_NET_POCKET	N	Pocket and portable adaptors

Table 3-19: Token Ring Devices

Option	Value	Description
CONFIG_TR	N	Token Ring driver support

Table 3-20: WAN Interfaces

Option	Value	Description
CONFIG_HOSTESS_SV11	N	Control Hostess SV-11 support
CONFIG_COSA	N	COSA/SRP sync serial boards support
CONFIG_SEALEVEL_4021	N	Sealevel Systems 4021 support
CONFIG_DLCI	N	Frame relay DLCI support

Table 3-21: SCSI Support

Option	Value	Description
CONFIG_SCSI	N	SCSI support

Table 3-22: Sound

Option	Value	Description
CONFIG_SOUND	Y	Sound support
CONFIG_SOUND_SA1100	Y	Support for audio output on SA1100 boards
CONFIG_SOUND_ES1370	N	Ensoniq ES1370-based PCI sound cards
CONFIG_SOUND_ES1371	N	Ensoniq ES1371-based PCI sound cards
CONFIG_SOUND_SONICVIBES	N	S3 SonicVibes-based PCI sound cards
CONFIG_SOUND_MSNDCLAS	N	Support for Turtle Beach MultiSound Classic, Tahiti, Monterey
CONFIG_SOUND_MSNDPIN	N	Support for Turtle Beach MultiSound Pinnacle, Fiji
CONFIG_SOUND_OSS	N	OSS sound modules

Table 3-23: Filesystems

Option	Value	Description
CONFIG_QUOTA	N	Quota support
CONFIG_AUTOFS_FS	N	Kernel automounter support
CONFIG_AFFS_FS	N	Amiga FFS filesystem support
CONFIG_HFS_FS	N	Apple Macintosh filesystem support (Experimental)
CONFIG_FAT_FS	N	FAT filesystem support
CONFIG_ISO9660_FS	N	ISO 9660 CD-ROM filesystem support
CONFIG_JOLIET	N	Microsoft Joliet CD-ROM extensions
CONFIG_MINIX_FS	N	Minix filesystem support

Table 3-23: Filesystems (Continued)

Option	Value	Description
CONFIG_NTFS_FS	N	Windows NT NTFS support (read only)
CONFIG_HPFS_FS	N	OS/2 HPFS filesystem support (read only)
CONFIG_PROC_FS	Y	/proc filesystem support
CONFIG_ROMFS_FS	N	ROM filesystem support
CONFIG_EXT2_FS	Y	Second extended filesystem support
CONFIG_SYSV_FS	N	System V, Version 7 and Coherent filesystem support
CONFIG_UFS_FS	N	UFS filesystem support
CONFIG_BLUECAT_FFS	Y	BlueCat Linux Flash File System support

Table 3-24: Network Filesystems

Option	Value	Description
CONFIG_CODA_FS	N	Coda filesystem support (advanced network filesystem)
CONFIG_NFS_FS	N	NFS filesystem support
CONFIG_LOCKD	N	lockd support
CONFIG_SMB_FS	N	SMB filesystem support (to mount WfW shares, etc.)
CONFIG_NCP_FS	N	NCP filesystem support (to mount NetWare volumes)

Table 3-25: Partition Types

Option	Value	Description
CONFIG_BSD_DISKLABEL	N	BSD disklabel (BSD partition tables) support
CONFIG_MAC_PARTITION	N	Macintosh partition map support
CONFIG_SMD_DISKLABEL	N	SMD disklabel (Sun partition tables) support
CONFIG_SOLARIS_X86_PARTITION	N	Solaris (x86) partition table support
CONFIG_NLS	N	Native language support

Table 3-26: Kernel Hacking

Option	Value	Description
CONFIG_FRAME_POINTER	N	Compiles kernel with frame pointer (useful for debugging)
CONFIG_DEBUG_ERRORS	N	Verbose kernel error messages
CONFIG_DEBUG_USER	N	Verbose user fault messages
CONFIG_DEBUG_INFO	N	Includes debugging information in kernel binary
CONFIG_MAGIC_SYSRQ	N	Magic SysRq key
CONFIG_BLUECAT_KDBG	N	Includes <code>kdbg</code> kernel debugger

Table 3-27: LynuxWorks Messenger Support

Option	Value	Description
CONFIG_BLUECAT_IOPMAN	N	Enables LynuxWorks IOP Manager support
CONFIG_BLUECAT_MSNG	N	Enable Messenger support

Table 3-28: Modular Advanced Power Management

Option	Value	Description
CONFIG_BLUECAT_APM	N	Modular Advanced Power Management support

Supported Demo Systems

This chapter provides information about BlueCat Linux demo systems supported by the SA-1110 Target Support Package (TSP).

Demo Systems

Table 4-1 shows the demo systems supported in the SA-1110 TSP standard distribution, boot devices supported by each demo system, and their RAM and ROM requirements:

Table 4-1: Demo Systems Supported by SA-1110 TSP

Demo System	Default Boot Devices Supported	ROM Requirements	RAM Requirements
developer	Flash Network using OS loader	2769 KB	12288 KB
osloader	Flash Network using OS loader	753 KB	2560 KB
showcase	Flash Network using OS loader	3032 KB	13312 KB

developer Demo System

The `developer` demo system includes functionality of the `shell`, `ftp`, `ping`, `gdb`, and `vl_demo` demo systems. Refer to Chapter 4 of the

BlueCat Linux User's Guide for a description of `developer` and its constituent demo systems.

osloader Demo System

`osloader` is the BlueCat Linux OS loader used to boot a BlueCat Linux embedded system on the target board. Refer to Chapter 4 of the *BlueCat Linux User's Guide* for details.

showcase Demo System

This section provides information on the `showcase` demo system. The `showcase` demo system showcases certain features specific to the SA-1110 development board, and configures an Apache web server. A generic description can be found in Chapter 4 of the *BlueCat Linux User's Guide*. Board-specific features are described below.

DEMO

An Apache web server and a simple MicroWindows MP3 player demonstration

SYNOPSIS

This demo system starts and configures the Apache HTTP daemon turning the target board into a Web server. At the same time, a simple MicroWindows MP3 application is started.

REQUIREMENTS

Storage	Medium
RAM	Medium
Network	Yes
Disk	None
Kernel Option	The choice of <i>kernel options</i> for this demo system depends on the root filesystem used:

- If the root filesystem is an FFS image programmed into target flash memory, then the following two kernel options must be specified:

```
assabet_part_conf="\8-39\" root=1f01
```

- If the root filesystem is an RFS image, then the `ramdisk_size=32768` option must be specified.

DESCRIPTION

The system boots up in single-user mode. `init` starts `bash` without a login prompt. `bash` automatically runs the `/.bashrc` script file, which executes commands to bring up the network interface and set up the kernel routing table. The network configuration is displayed on the terminal connected to the serial line. `.bashrc` starts the `httpd` daemon and a MicroWindows MP3 demo application.

The MicroWindows MP3 demo application draws a window with a picture on the screen and starts the MP3 demonstration. If the user touches the application window, the demo flips from one picture to another, while stopping and restarting the MP3 application. When the demo finishes playing the MP3 file, it pauses for 5 seconds before replaying the file from the beginning. Pressing the Close button of the application window causes the demo to quit. Also, the application window can be moved around the screen by dragging the top bar.

After the demo system is booted the Apache server is accessible from any networked machine using the IP address entered by the user, and serves the `index.html` page located in the `showcase` subtree in the `demo` directory.

The `showcase` demo system can be downloaded into target flash memory either as a BlueCat Linux composite image composed of kernel and root filesystem images, or as a BlueCat Linux image containing a kernel image and an FFS image mounted as root. The former case is detailed under “Downloading showcase into Target Flash Memory” in Chapter 2, and the latter in “Downloading and Booting showcase with an FFS Root Filesystem” below.

Downloading and Booting showcase with an FFS Root Filesystem

To download the `showcase` demo system into target flash memory using `i_osloader` with the FFS image mounted as a root filesystem, use the following procedure:

1. Rebuild the `showcase` BlueCat Linux image so that it does not include the compressed root filesystem image.

Also, the kernel command line in the BlueCat Linux image must have the `assabet_part_conf` and `root` options.

The target flash memory partition configuration string must be in accordance with the argument supplied to `flash_fdisk` in Step 2 below.

The following command issued while in the `$BLUECAT_PREFIX/demo/showcase` directory rebuilds the BlueCat Linux image and sets the appropriate kernel command line:

```
BlueCat:$ echo \  
"assabet_part_conf=\"8-39\" root=1f01"|\  
mkboot -m -k showcase.kernel -c stdin \  
showcase.kdi
```

2. Partition the SA-1110 board's flash memory. Create two partitions, one for the kernel image and the other for the root filesystem image. For instance:

```
> exec flash_fdisk /dev/mtdchar0 0-7:8-39
```

This command creates a first partition of size 2 MB for the kernel at the beginning of flash memory, and a second partition of size 8 MB for the root filesystem image right after the first partition.

3. Set the environment variables to properly configure network access. For instance:

```
> set IP 172.17.3.8  
> set HOST 172.17.0.1  
> set IF eth0
```

4. Set the `FILE` environment variable so that it points to the `showcase` FFS root filesystem image. For instance:

```
> set FILE tftp /tftpboot/showcase.jffs
```

5. Burn the FFS image into target flash memory:

```
> flash /dev/mtdchar2 erase
```

6. Set the `FILE` environment variable so that it points to the `showcase` BlueCat Linux kernel image. For instance:

```
> set FILE tftp /tftpboot/showcase.kdi
```

7. Burn the BlueCat Linux kernel image into target flash memory:

```
> flash /dev/mtdchar1 erase
```

8. Reset the target board:

```
> reset
```

Upon successful completion of the procedure, the `showcase` demo system will start from target flash memory automatically.

NOTE: *Mounting the `showcase` FFS image as a root filesystem takes about 20 seconds during the kernel bootstrapping process.*

Modifying Target Board IP and Gateway Addresses

The `showcase` demo system includes an Apache web server feature. The Apache web server is “a robust, commercial-grade, featureful, and freely-available source code implementation of an HTTP (web) server.” See <http://www.apache.org> for further information.

Before installing the BlueCat Linux `showcase` demo system, users who wish to take advantage of the `showcase` Apache web server feature may wish to define their own target board IP, rather than use the default address (216.100.252.144) defined in the `showcase` demo system’s `.bashrc.assabet` file.

To change the `showcase` default IP address for the Apache web server BlueCat Linux web pages, open the `showcase` demo system’s

`.bashrc` `assabet` file with any text editor (such as `vi`), and perform the following steps:

1. Change the default IP address (216.100.252.144) to a new user-selected target board IP address by entering a new address at the following line:

```
TARGET_IP=target_IP
```

2. Set a gateway IP address by entering a value at the following line:

```
GATE_IP=gateway_IP
```

3. Rebuild the `showcase` demo system by entering the following command under the `$BLUECAT_PREFIX/demo/showcase` directory:

```
BlueCat:$ touch showcase.spec
```

```
BlueCat:$ make rootfs
```

```
BlueCat:$ make kdi
```

Supported Device Drivers

Table 5-1 shows the device drivers supported by the SA-1110 TSP and details important information about them:

Table 5-1: Device Drivers Supported by the SA-1110 TSP

Hardware Device	Device Drivers	Location in Source Tree	Kernel Configuration Options	Notes
UART	serial_sa1100.c	drivers/char	CONFIG_SERIAL_SA1100 CONFIG_SERIAL_SA1100_CONSOLE	
Touch Screen Phillips UCB1300	ts_sa1100.c ts_sa1100.h	drivers/char include/linux	CONFIG_TS_SA1100	
LCD Frame Buffer	fb_sa1100.c	drivers/video	CONFIG_FB_SA1100	
LCD FrontLight	fl_sa1100.c fl_sa1100.h	drivers/char include/linux	CONFIG_FL_SA1100	
Digital Audio Interface Phillips UDA1341	audio_sa1100.c	drivers/sound	CONFIG_SOUND_SA1100	
Real Time Clock	rtc_generic.c rtc_generic.h	drivers/char include/linux	CONFIG_RTC	

Table 5-1: Device Drivers Supported by the SA-1110 TSP (Continued)

Hardware Device	Device Drivers	Location in Source Tree	Kernel Configuration Options	Notes
Ethernet CompactFlash card Socket CF+	ne.c 8390.c 8390.h	drivers/net drivers/net drivers/net	CONFIG_ASSABET_CFPLUS	To use the Ethernet card, CompactFlash support must be enabled using the CONFIG_CF_ASSABET option.
NOR Flash	assabet.c	drivers/mtd	CONFIG_MTD_ASSABET CONFIG_MTD_ASSABET_PART	