

ELAN-104NC

Technical Manual For Embedded Linux

Product Information

Full information about other Arcom products is available by visiting our WebSite at:
www.arcomcontrols.com

Useful Contact Information

Customer Support Europe

Tel: +44 (0)1223 412 428
Fax: +44 (0)1223 403 409
E-mail: support@arcom.co.uk

Customer Support US

Tel: 816 941 7025
Fax: 816 941 7807
E-mail: support@arcomcontrols.com

Sales Offices

United Kingdom:

Arcom Control Systems Ltd
Clifton Road
Cambridge CB1 7EA,
UK
Tel: 01223 411 200
Fax: 01223 410 457

E-mail:
sales@arcom.co.uk

United States:

Arcom Control Systems Inc
13510 South Oak Street
Kansas City MO 64145
USA
Tel: 816 941 7025
Fax: 816 941 0343

E-mail:
icpsales@arcomcontrols.com

Sales Hotlines

Belgium:

Groen Nummer:
Tel: 0800 7 3192
Fax: 0800 7 3191

France:

Numero Vert
Tel: 0800 90 84 06
Fax: 0800 90 84 12

Germany:

Kostenlose Infoline:
Tel: 08001 824 511
Fax: 08001 824 512

Netherlands:

Gratis 0800 Nummer:
Tel: 0800 0221136
Fax: 0800 0221148

Italy:

Numero Verde:
Tel: 0800 790841
Fax: 0800 780841

Whilst Arcom's sales team is always available to assist you in making your decision, the final choice of boards or systems is solely and wholly the responsibility of the buyer. Arcom's entire liability in respect of the boards or systems is as set out in Arcom's standard terms and conditions of sale.

If you intend to write your own low level software, you can start with the source code on the disk, which is supplied. This is example code only to illustrate use on Arcom's products. It has not been commercially tested. No warranty is made in respect of this code and Arcom shall incur no liability whatsoever or howsoever arising from any use made of the code.



Contents

Contents	1
Revision History	1
Overview.....	2
Detailed software list	3
File system layout.....	4
Configuration files and boot scripts.....	5
Arcom Embedded Linux CD contents	6
Installing.....	7
Installing on a headed system	7
Installing on a headless system.....	7
Using the CD as a rescue disk	7
Device drivers	8
Flash	8
Raw access	8
Utilities.....	10
eraseall	10
flashboot	10
Examples:.....	11
Disabling or Bypassing the flash boot loader.....	12
Disabling and re-enabling the flash boot loader	12
Bypass the flash boot loader	12
Installing Linux on to a hard drive	13
Standard Linux install	13
Compiling a kernel.....	13
Enable Boot from new kernel.....	14
Install the support software.....	15
Configuring the hard drive system for flash or Hard drive boot	15
Testing applications on the flash file system.....	16
Bios update facility	17
Creating the Bios update floppy disk from a Windows system	17
Creating the Bios update floppy disk from a Linux system.....	17
Using the Bios Update Disk.....	17
Appendix A.....	18
Current limitations of JFFS	18
Hard links are not supported	18
Output of 'df' command is incorrect.....	18
Shared, writable mmappping of files is not supported	18
Delay during mounting	18
Appendix B.....	19

Revision History

Manual	Comments
Issue A	001010 Initial revision

Overview

Arcom's Embedded Linux is a standard Linux distribution specially reduced to fit on the on-board flash array. The distribution is based in part on Red Hat Linux 6.2 software. Key software includes

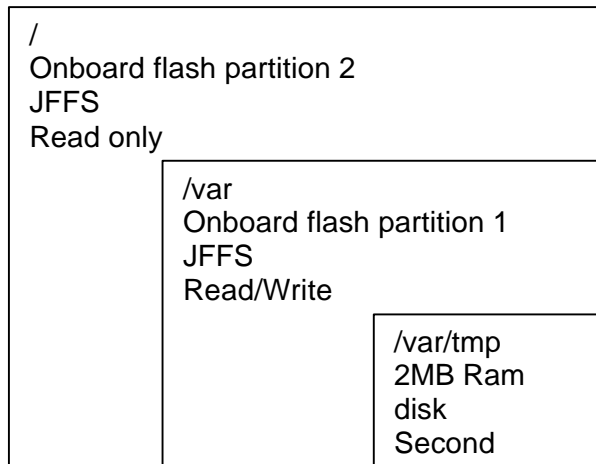
- Kernel 2.2.17
- GNU C library 2.1.3
- JFFS (Journalling Flash File System) installed on the onboard flash array

Detailed software list

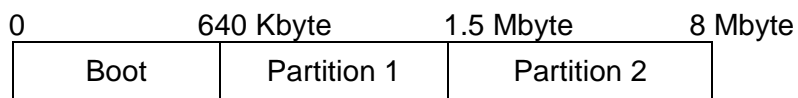
Component	Commands / Files Provided
GNU C library 2.1.3	libc.so.6, libnsl.so.1, libnss_files.so.2, libnss_dns.so.2, libresolv.so.2, libutil.so.1, libm.so.6, ld-linux.so.2, libdl.so.2, libcrypt.so.1
NCurses 4.0	libncurses.so.4.0
Bash 2.04	Complete with command line editing and history
ae 962	Text editor
BusyBox 0.47	basename, cat, chmod, chown, chgrp, chroot, chvt, clear, cp, cut, date, dd, deallocvt, df, dirname, dmesg, dtmp, du, dumpkmap, echo, false, fdflush, free, freeramdisk, grep, gunzip, gzip, head, hostid, hostname, id, kill, killall, length, ln, loadacm, loadfont, loadkmap, logger, logname, ls, makedevs, md5sum, mkdir, mkfifo, mknod, mktemp, more, mv, nslookup, ping, printf, ps, pwd, rm, rmdir, sed, setkeycodes, sleep, sort, sync, tail, tar, tee, test, touch, tr, true, tty, uptime, usleep, wc, which, whoami, uniq, uname, update, yes
TinyLogin 0.78	adduser, addgroup, deluser, delgroup, getty, login, passwd, su, sulogin
modutils 2.3.11	depmod, insmod, lsmod, modprobe, rmmod
sysvinit 2.78	halt, init, poweroff, reboot, shutdown
ftp client	ftp
linux-ftpd 0.17	FTP server
telnet	Client & server
thttpd 2.20b	Light-weight HTTP server
mount, umount stty, reset	
mkfs.ext2	Plus required libraries: libcom_err.so.2.0, libext2fs.so.2.4, libe2p.so.2.3, libuuid.so.1.2
syslogd, klogd	
networking utilities	ifconfig, route
Find	
Inetd	Tcpd
Ldconfig	

File system layout

The entire linux file system is spread across 3 separate partitions for added data security.



The onboard flash partitions are positioned as follows in the on-board flash array



Configuration files and boot scripts

Arcom Embedded Linux uses a System V type init process. Scripts are placed in /etc/init.d/ with symlinks for each runlevel in /etc/rc?.d/. The '?' maybe replaced with one of the following characters:

Character	Function	Description
S	Startup	Run once at boot time
0	Halt	Run on system shutdown
1	Single	Run on entering single user mode
2	Normal	
3	Normal	
4	Normal	
5	Normal	
6	Reboot	Run when rebooting

When runlevel changes, the K* scripts in the /etc/rc?.d/ directory corresponding to the new runlevel are executed first in alphanumerical order (with an argument of 'stop'). Then the S* scripts in the same directory are executed in alphanumerical order (with an argument of 'start').

Arcom specific (used in the boot scripts) configuration files are located in /etc/config

File	Description	Format
/etc/config/console/keymap	Default keymap loaded during boot	Symlink to the real keymap. (Usually under /usr/lib/kbd/keymaps).
/etc/config/network/hostname	Hostname	HOSTNAME = <i>hostname</i>
/etc/config/network/interfaces/*	Network interface configuration. One file per interface.	DEVICE = <i>network device name</i> IPADDR = <i>IP address</i> NETMASK = <i>netmask</i>
/etc/config/network/routes/gateway	Default gateway IP address	GATEWAY = <i>IP address</i>
/etc/config/hardware/serial/*	Serial port IRQ configuration.	DEVICE = <i>path to device node</i> IRQ = <i>IRQ</i>

Arcom Embedded Linux CD contents

Directory Name	Description
Linux Specific	
utils	Embedded Linux utilitites
Linux Examples	Example code
install	Installation files
install/boot	Kernel and Boot loader Images
Boot	Boot disk image
Source	Arcom Embedded Linux Source
Board Specific	
Acrobat Reader	Acrobat Reader
Arcom Drivers	Drive for NVSRAM
Bios	Boot image and bios updating utilities
VIDEO	Mode switching utilities
Demos	Demo applications
Ethernet	RTL8019 Ethernet drivers
Examples	General example code
FlashFx	Datalight FlashFx drivers
Internet Explorer 5	MS IE5
Manual	Arcom Documentation
Reference	Board reference documentation
Romdos	Datalight ROM-DOS

Installing

Installing on a headed system

1. Ensure the display, keyboard and CD-ROM drive are correctly plugged in
2. Power on the board
3. Insert the CD and reboot
4. The CD will start and display some text and a prompt

Welcome to the Arcom Embedded Linux n.nn Installation CD

...

boot:

5. Press 'a' then 'enter' if the CD-ROM drive is configured as master or 'b' then 'enter' if the CD-ROM is configured as slave. If unsure check the jumper setting on the drive or choose 'b'. Note that after 30 seconds the CD will continue booting using a serial terminal as the console, reboot if this occurs.
6. After Linux has finished booting the installation program will start. Use TAB, ALT+TAB, and the cursor keys to select fields; space bar to toggle check boxes and press buttons.
7. A number of optional components can be selected. Do not select the X client and X server for the ELAN104-NC. Choose PPP for PPP dial-up support, if required.
8. Press the Next button to continue
9. Configuring the network: The hostname is required although you can accept the default. If you require an IP address you must also supply a netmask. The default gateway and nameserver are optional.
10. Press the Next button to continue
11. Ensure the "Boot from flash" checkbox is enabled if you wish to boot from the onboard flash. Add any extra kernel parameters if required though they will not normally be needed.
12. Press the Next button to continue

Installing on a headless system

1. Connect a serial terminal (VT100) to the first serial port (/dev/ttyS0, COM1). Configure the serial terminal to 9600 baud, no parity and 8 data bits. Setting up serial terminals is beyond the scope of this document (see the Serial Terminal HOWTO for details)
2. Ensure the CD-ROM is connected and configured as a slave device
3. Power on the board
4. Insert the CD
5. The CD should will boot and after 30 seconds or so Linux will start to boot displaying to the serial terminal
6. Follow from step 7 given in the section above.

Using the CD as a rescue disk

Should the flash become unbootable for some reason the CD can be used to reinstall the flash boot loader and fix problems with the installation. At the first screen of the installation program press the Back button. You'll be presented with a login prompt. Login as root (no password).

Install the boot loader with the flashboot command. The boot loader image is in /cd/install/boot/elan-nc and is stage2.rom. An optional splash screen bitmap can be found as /cd/install/boot/common/splash.bmp. The kernel image is /cd/install/boot/elan-nc/zImage.

```
flashboot -2 stage2.rom -s splash.bmp zImage root=1f02 ro
```

Mounting the flash partitions

```
mount -t jffs /dev/mtdblock1 /mnt/1
mount -t jffs /dev/mtdblock2 /mnt/2
```

Setup the network

```
ifconfig eth0 IP address netmask netmask
route add default gw IP address of default gateway
```

The CD contains most of the utilities that are installed (ftp client, ae text editor etc).

Device drivers

Flash

Raw access

Raw (byte) access to the flash is provided via the Memory Technologies Device (MTD) sub-system. This provides the following character devices (major number: 90)

Device	Minor number	Description	Read/write?
/dev/mtd0	0	1 st , (boot) partition	Read/write
/dev/mtd1	2	2 nd , (/var/) partition	
/dev/mtd2	4	3 rd , (/) partition	
/dev/mtdr0	1	1 st , (boot) partition	Read only
/dev/mtdr1	3	2 nd , (/var/) partition	
/dev/mtdr2	5	3 rd , (/) partition	

These devices can be read from and written to using standard utilities (dd, cat etc). In addition a utility (eraseall) is provided on the CD in directory /utils , which allows you to erase entire devices.

In addition to the standard C system calls (read(), write(), lseek() etc) a number of ioctl's are provided to obtain information or perform erases. You'll require the kernel and MTD sources to be correctly installed. Ensure that "linux/mtd/mtd.h" is #include'd.

ioctl() request	Third parameter	Description
MEMGETINFO	mtd_info_t *	Obtain device information
MEMERASE	erase_info_t *	Erase given region of flash

mtd_info_t and erase_info_t are as follows:

```
typedef struct {
    unsigned char type;    // Type of device. Always MTD_NORFLASH
    unsigned char flags;  // Device capabilities. Always MTD_CAP_NORFLASH
    unsigned long size;    // Total size of device in bytes
    unsigned long erasesize; // Size of an erase zone in bytes
} mtd_info_t;
```

```
typedef struct {
    unsigned long start; // Offset (in bytes) to start erase from
```

```
        unsigned long length // Total number of bytes to erase
    } erase_info_t;
```

Note that erases must start and end on erase zone boundaries.

Do NOT write to or erase the flash via the raw character devices if JFFS is used on that partition as this will cause data loss or corruption.

File system

In addition to raw access the flash can be accessed via a journalling file system -- the Journalling Flash File System (JFFS). This places a file system onto the flash which is transparent to the user. As supplied Arcom Embedded Linux is preinstalled onto the flash using JFFS.

Flash partitions with JFFS are mounted using a special pseudo-block device (major: 31)

```
mount -t jffs /dev/mtdblock1 mount point
```

The block devices are

Device	Minor number	Description
/dev/mtdblock0	0	1 st , (boot) partition
/dev/mtdblock1	1	2 nd , (/var) partition
/dev/mtdblock2	2	3 rd , (/) partition

The first partition (/dev/mtdblock0) does not have a JFFS partition when supplied, the user may put one on if the board is not required to boot from flash.

No special utility is required to make a JFFS file system. Simply erase the whole of the partition (with eraseall) and mount as normal. The file system will be initially empty.

JFFS partitions do not need to be fsck'ed (this is done when mounting). The /sbin/fsck.jffs supplied is a dummy, always succeeds and is solely to simplify the boot scripts.

Although JFFS is a journalling file system this does not preclude the loss of data. However, the filesystem will remain in a consistent state and always be mountable. Data can be lost if the board is powered down during a write.

Utilities

eraseall

The eraseall utility is found in the CD's /utils directory.

eraseall erases all of a given MTD device.

```
eraseall MTD char device
```

Take care when using this command as specifying the wrong device can cause data loss.

Example:

To erase all of the boot partition.

```
eraseall /dev/mtd0
```

flashboot

The flashboot utility installs or uninstalls the flash boot loader and kernel image, it is found in the CD's /utils directory.

The boot loader consists of 5 components

- stage 1 loader
- splash screen bitmap (optional)
- stage 2 loader
- kernel command line
- kernel image.

All components must be installed correctly for the boot loader to work.

The kernel image must be a zImage as bzImages are not supported. A custom, precompiled kernel image is supplied on the development kit CD as /install/boot/elan-nc/zImage.

If the standard partition/filesystem layout is used the kernel parameters must contain "root=1f02" or the root filesystem on the flash will not be mounted. You cannot use the device name.

The splash screen is either a 640 × 480, 800 × 600 or 1024 × 768, 256 color Windows format bitmap. It must be less than 96 Kibyte in size. RLE8 encoded bitmaps only are supported.

The stage 2 boot loader can be found on the development kit CD as
/install/boot/elan-nc/stage2.rom .

Usage:

```
flashboot [Option]... [Kernel image] [Kernel command line parameter]...
```

Options:

- 1. --stage1=LOADER
location of the stage 1 boot loader image. [default: /boot/stage1.rom]
- 2. --stage2=LOADER
location of the stage 2 boot loader image. [default: /boot/stage2.rom]
- d, --device=DEVICE
MTD char device to install boot loader to and image. This should always be the first MTD device or the boot loader will not work. [default: /dev/mtd0]

- k, --kernel-only
only install a kernel image and command line. The boot loader must already be installed or it will not boot.
- n, --no-erase
don't erase flash before installing - flash must already be erased or the images will not be written correctly.
- s, --splash-bmp=BMP
location of splash screen bitmap
- u, --uninstall
disable boot loader. This does not erase the flash and reinstalling will require erasing.
- help
display concise help and exit
- version
output version information and exit

Examples:

Reinstalling all of the boot loader

```
flashboot -2 stage2.rom -s splash.bmp zImage root=1f02 ro
```

Changing the command line (this requires reinstalling the kernel image)

```
flashboot -k zImage root=1f02 ro
```

Disabling the boot loader

```
flashboot -u
```

Re-enabling the boot loader after disabling

```
flashboot -2 stage2.rom -s splash.bmp
```

Disabling or bypassing the flash boot loader.

Disabling and re-enabling the flash boot loader

The flash boot loader can be disabled using the eraseall utility and re-enabled using the flash boot Utility. See the Utilities section of this manual for detailed information on the use of these utilities

To disable the boot loader type

```
eraseall /dev/mtd0
```

Bypass the flash boot loader

If you want to bypass the flash boot loader then do the following

Reboot the system

Hold down the ALT key

When you see the following displayed on the screen

```
ALT-F1- Bypass flash boot  
ALT-F5 -Do not display splash screen
```

Hold down the F1 key

Release the F1 key then release the ALT key.

Installing Linux on to a hard drive

Before being able to boot from the installation CD it is advisable to disable the flash boot loader if it is enabled(see disabling flash boot loader section)

Standard Linux install

Install a standard Linux onto the hard drive.

Arcom recommends Red Hat Linux 6.2, although other distributions may be suitable.

1. For a detailed description of the Linux installation see the Red Hat Linux Installation guide.
2. We suggest the following
 - a. At the first screen press enter to select the first option
-To install or upgrade a system running Red Hat Linux 2.0 or later in graphical mode, press<Enter> key.
 - b. AT the screen that asks for a driver disk select cancel
 - c. Select the language, keyboard and mouse
 - d. At the install options screen select KDE workstation
 - e. We suggest that the following partitions are created when requested.
 - i. Swap partition 127Mbyte
 - ii. /boot partition 16 Mbytes
 - iii. /root partition Rest of Hard Drive
 - f. Select the appropriate time zone
 - g. Configure the login accounts as required
 - h. Select the correct monitor
 - i. The Graphics controller is a Chips and Technologies CT65550
 - j. The Installation should now begin.
 - k. At the end of the installation you will ask whether you want to create a boot disk.
This is recommended

4. Remove the CD to allow the Red Hat 6.2 installation to boot from the Hard drive.

For other Linux distributions install according to the instructions supplied with the distribution.

Compiling a kernel

To access the flash and other ELAN-104NC specific devices a new Linux kernel will need to be compiled.

An Arcom specific kernel source tree is supplied on the Arcom Development Kit CD in /source/linux-2.2.17-ael-1.01.tgz

Mount the CD

Mount /dev/hdb /mnt/cdrom

Copy the kernel tarball from the CD (to for example /usr/local/src/)

```
cp /mnt/cdrom/source/linux-2.2.17-ael-1.01.tgz /usr/local/src
```

Unpack the source code

```
cd /usr/local/src  
tar xzf linux-2.2.17-ael-1.01.tgz
```

Configure the kernel (the kernel configuration file elan-104nc+mtd.conf can be used as a starting point)

```
cd linux-2.2.17  
make menuconfig
```

Select Load an Alternate Configuration File from the Main menu

Build the dependencies by typing

```
Make dep
```

Build the kernel image by typing

```
Make clean  
make bzImage
```

The bzImage is used **only** for a hard drive installation. Arcom embedded Linux installed on the flash does not support bzImage it requires a zImage instead.

Build and install the modules

```
make modules  
make modules_install
```

Copy the kernel image and System.map file to appropriate places

```
cp arch/i386/boot/zImage /boot/vmlinuz-2.2.17  
cp System.map /boot/System.map-2.2.17
```

Enable Boot from new kernel

To enable booting from this new kernel the /etc/lilo.conf file will need to be modified. Add lines similar to

```
image=/boot/vmlinuz-2.2.17  
label=hdd  
read-only  
root=/dev/hda6
```

Consult the lilo.conf man page and distribution documentation if you need more details.

You then need to run lilo, this installs a boot loader that will be activated the next time you boot the system.

Run lilo by typing

```
lilo
```

Now reboot the system and at the Lilo prompt select hdd.

Install the support software

Included on the CD is the following support software

eraseall, checklib and flashboot.

These can be found in the utils directory on the CD. Copy these three files to the /usr/bin directory on the hard drive.

Also on the CD is a stage2.rom image This is in the install/boot/elan-nc directory. Copy this to /boot on the hard drive.

Configuring the hard drive system for flash or Hard drive boot

If you wish to you can configure the hard drive system to boot from flash root system or the hard drive root system.

Create the flash device files by typing

```
/usr/local/src/mtd/utils/MAKEDEV
```

Create the flash filesystem mount point by typing.

```
mkdir /flash
```

Add the following lines to /etc/fstab

```
/dev/mtdblock2    /flash    jffs    defaults    0 0
/dev/mtdblock1    /flash/var jffs    defaults    0 0
```

You can use the vi editor as follows

vi /etc/fstab to start editor

INSERT key will allow you edit the file

ESC key will exit the editing mode.

Exit vi by typing

```
:qa!
```

Mount the flash filesystems

```
mount /flash
```

```
mount /flash/var
```

Modify /etc/lilo.conf to allow the booting of the kernel on the flash by adding lines similar to (consult the lilo.conf man page and distribution documentation for more details)

```
image=/boot/vmlinuz-2.2.17
label=flash
```

```
read-only  
root=/dev/mtdblock2
```

```
Rerun lilo  
lilo
```

Reboot and chose (at the LILO prompt) hdd – root filesystem on the harddisk or flash – root filesystem on the flash memory.

Testing applications on the flash file system

Application can be tested in an environment similar to Arcom Embedded Linux by temporarily changing the root directory (and hence the visible files) by using the chroot command. Using the application dir as an example.

Copy the applications from the hard drive to the flash filesystem which is mounted on /flash

```
cp /usr/bin/dir /flash/usr/bin
```

Check all the required libraries are available using check-libs

```
check-libs -r /flash
```

Check-libs will say that libtermcap.so.2 is missing.

Strip and copy any missing libraries across.

```
cd /lib  
Strip libtermcap.so.2 -o /flash/lib/libtermcap.so.2
```

Change the root directory to /flash

```
chroot /flash
```

Run the application etc. to test

```
dir
```

Leave the chrooted environment

```
exit
```

Note that in the chrooted environment the /proc filesystem will not be automatically mounted and can be mounted with

```
mount /proc
```

Either reboot and choose flash at the LILO prompt, or ensure that the flash boot loader is enabled.

Bios update facility

The CD contains a disk image, which allows the Bios to either be put back to the normal ELAN-104NC bios, or to update to the bios required for Linux on the ELAN-104NC.

Creating the Bios update floppy disk from a Windows system

Place a new floppy disk in the floppy drive then run the makedisk batch file which is in the /bios directory on the CD.

Creating the Bios update floppy disk from a Linux system

Place a new floppy in the floppy drive of your Linux system, then type

```
dd if=BIOSBOOT.DSK of=/dev/fdo
```

Using the Bios Update Disk

Place the update floppy disk in the floppy drive of the Arcom Elan-104NC system.

Reboot the system from the floppy drive. If the flash boot loader is enabled then you can either disable the flash boot loader(See the Disabling and enabling Flash boot loader section), or Bypass the boot loader (see Bypass flash boot loader section)

Select 3 Install Datalight BIOS for Embedded Linux
Select 4 Install Standard Datalight BIOS.

If you experience any problems with this facility power down the system for approx 10 Seconds and then power up again. Repeat the previous section.

Appendix A

Current limitations of JFFS

Hard links are not supported

Hard links creation with the `link()` system call or the `ln` command will fail with `EPERM` (Operation not permitted). This can lead to some programs mysteriously failing with an unexpected or unrelated error. In particular a number of programs use hard links as a method to lock files before updating.

Possible solutions:

- Make the program use `/tmp` or `/var/tmp` (which is a ram disk with a Second Extended file system)
- Modify the program to lock files using the `flock()` or `fcntl()` system calls

No program supplied in the development kit attempts to create hard links on JFFS partitions.

Output of 'df' command is incorrect

The number of blocks used/available given in the output of the `df` command is incorrect. The number of used blocks may be greater than the actual number of blocks used.

'du' can be used instead.

Shared, writable mmaping of files is not supported

The `mmap()` system call will fail with `EINVAL` if `PROT_WRITE` and `MAP_SHARED` are both used. eg `mmap(0, length, PROT_READ|PROT_WRITE, MAP_SHARED, fd, 0)` will fail if `fd` is the descriptor for a file on a JFFS filesystem.

Some application software will assume that such a system call will always succeed and will fail mysteriously (possibly with a segmentation fault or similar).

If the file is not required not persist across reboots then a work around is to use a symlink to the `/var/tmp` directory. This is on the ram disk which is an ext2 file system which support all `mmap()` operation.

Delay during mounting

There is a delay of approx. 20s when mounting the JFFS filesystems. This is due to the comprehensive integrity check which is performed each time the f/s is mounted. Future releases from the MTD/JFFS development team (www.linux-mtd.infradead.org) will substantially reduce this delay.

Appendix B

Sources for the software contained in Arcom Embedded Linux.

Linux kernel 2.2.17

- <ftp://ftp.kernel.org/pub/linux/kernel/2.2/linux-2.2.17.tar.gz>
- + elan-a20.patch (available from Arcom)
- + MTD from CVS (15 September 2000)
<http://www.linux-mtd.infradead.org/>

Bash 2.04

- <ftp://ftp.gnu.org/pub/gnu/bash/bash-2.04.tar.gz>

modutils 2.3.11

- <ftp://ftp.kernel.org/pub/linux/utils/kernel/modutils/v2.3/modutils-2.3.11.tar.gz>

TinyLogin 0.78

- <ftp://opensource.lineo.com/tinylogin/tinylogin-0.78.tar.gz>
- + patches which have been incorporated into CVS
<http://opensource.lineo.com/cvs.html>

BusyBox 0.47

- <ftp://opensource.lineo.com/busybox/busybox-0.47.tar.gz>
- + patches which has been incorporated into CVS
<http://opensource.lineo.com/cvs.html>

linux-ftpd 0.16

- <ftp://ftp.uk.linux.org/pub/linux/Networking/netkit/linux-ftpd-0.16.tar.gz>

netkit-ftp 0.16

- <ftp://ftp.uk.linux.org/pub/linux/Networking/netkit/netkit-ftp-0.16.tar.gz>

thttpd 2.19

- <http://www.acme.com/software/thttpd/thttpd-2.19.tar.gz>

sysvinit 2.78

- <ftp://sunsite.unc.edu/pub/Linux/system/daemons/init/sysvinit-2.78.tar.gz>

ae 962

- ftp://ftp.debian.org/debian/dists/potato/main/source/base/ae_962.orig.tar.gz
- + patched with ftp://ftp.debian.org/debian/dists/potato/main/source/base/ae_962-26.diff.gz
- + compiled without SLang support and with ncurses

loadkeys 1.03wip + keymaps

- <ftp://ftp.kernel.org/pub/linux/utils/kbd/kbd-1.03wip.tar.gz>

pppd 2.4.0

- <ftp://linuxcare.com.au/pub/ppp/ppp-2.4.0.tar.gz>

syslinux 1.48

- <ftp://ftp.kernel.org/pub/linux/boot/syslinux/syslinux-1.48.tar.gz>

eraseall 0.1.0

- MTD CVS (15 September 2000)
<http://www.linux-mtd.infradead.org/>

kannel 10.3

<http://www.kannel.org>

Contact Arcom if modifications are required to the following software: flashboot, check-libs, Stage 1 flash boot loader (stage1.rom), arcom_setup.o, BIOS images and BIOS update utilities.

All other software taken from the Red Hat 6.2 distribution

<http://www.redhat.com/>

FAQS

Question I have Linux pre-loaded onto an ELAN-1-4NC board.
How do I boot from an attached floppy or CD-ROM drive ?

Answer During the LINUX boot process, you need to hold down the ALT key and then press the F1 key. This will bypass the LINUX BIOS extension and allow any attached drive to boot in the normal way.

Question **How do I remove LINUX from a configured ELAN_104NC board ?**

Answer At the LINUX prompt type :
dd if=/dev/zero of=/dev/mtd0 count=1
This will destroy the start of the BIOS extension as stored in flash. When the system is rebooted, the LINUX BIOS extension will not be found so any attached bootable media will be used instead.

Question **I need to remotely login to our ELAN-104NC board and run without video. I can remotely login, but switching to root has been disabled. How do I allow remote logging in as root ?**

Answer You need to log in locally as root and type:
chmod u+s /bin/tinylogin
You should then be able to login remotely as a normal user and su to root.