

Armadillo-500 FX 液晶モデル スタートアップガイド

A542701-D00Z
A546701-D00Z

Version 1.2.0
2010/02/02

株式会社アットマークテクノ [<http://www.atmark-techno.com>]

Armadillo 開発者サイト [<http://armadillo.atmark-techno.com>]

Armadillo-500 FX 液晶モデル スタートアップガイド

株式会社アットマークテクノ

060-0035 札幌市中央区北 5 条東 2 丁目 AFT ビル 6F
TEL 011-207-6550 FAX 011-207-6570

製作著作 © 2008-2010 Atmark Techno, Inc.

Version 1.2.0
2010/02/02

目次

1. はじめに	7
1.1. 対象となる読者	7
1.2. 本書の構成	7
1.3. 表記について	8
1.3.1. フォント	8
1.3.2. コマンド入力例	8
1.3.3. アイコン	8
1.4. 謝辞	9
1.5. 安全に関する注意事項	9
1.6. 取り扱い上の注意事項	9
1.7. ソフトウェア使用に関する注意事項	10
1.8. 商標について	10
2. 作業の前に	11
2.1. 見取り図	11
2.2. 準備するもの	11
2.3. 接続方法	11
2.4. ジャンパピンの設定について	12
2.5. シリアル通信ソフトウェアの設定	12
3. 起動と終了	14
3.1. 起動	14
3.2. ログイン	17
3.3. 終了方法	18
4. デモアプリケーション	19
4.1. 機能選択	19
4.2. LED 制御	19
4.3. ジャンパー状態確認	20
4.4. キーボード動作確認	20
4.5. バックライト調整	21
4.6. 音再生	21
4.7. SSD 接続状態確認	22
5. 起動モードとブートローダの機能	23
5.1. 起動モードの選択	23
5.2. Linux カーネル起動オプションの設定	23
5.2.1. コンソールの設定	23
5.2.2. ルートファイルシステムの設定	24
5.2.3. その他の起動オプション	25
5.2.4. 起動オプションの設定例	25
6. コンフィグ領域 - 設定ファイルの保存領域	26
6.1. コンフィグ領域の読出し	26
6.2. コンフィグ領域の保存	26
6.3. コンフィグ領域の初期化	26
7. ネットワーク	28
7.1. ネットワークの設定	28
7.1.1. 固定 IP アドレスに設定する	28
7.1.2. DHCP に設定する	29
7.1.3. DNS サーバを指定する	29
7.1.4. 接続を確認する	29
7.2. ファイアウォール	29
7.3. ネットワークアプリケーション	30
7.3.1. TELNET	30

7.3.2. FTP	30
7.3.3. SSH	31
7.3.4. Web サーバ	31
7.3.5. NTP クライアント	31
8. ストレージ	32
8.1. ストレージとして使用可能なデバイス	32
8.2. ストレージの初期化とマウント	32
8.2.1. ディスクの初期化	32
8.2.2. ファイルシステムの構築	33
8.2.3. マウント	33
9. サウンド	34
9.1. コーデック仕様	34
9.2. OSS によるサウンド機能の実現	34
9.2.1. サウンドを再生する	34
9.2.2. 録音する	35
9.2.3. 音量を変更する	36
10. その他のデバイス	37
10.1. LED	37
10.1.1. ledctrl による制御	37
10.1.2. ledctrl 使用例	37
10.2. GPIO	38
10.2.1. Direction を INPUT にする	38
10.2.2. Direction を OUTPUT にする	39
10.2.3. INPUT 専用の GPIO	39

目次

2.1. 見取り図	11
2.2. 接続図	12
3.1. 起動ログ	14
3.2. 終了方法	18
4.1. デモアプリケーション - 機能選択メニュー	19
4.2. デモアプリケーション - LED 制御	19
4.3. デモアプリケーション - ジャンパー状態確認	20
4.4. デモアプリケーション - キーパッド動作確認	20
4.5. デモアプリケーション - バックライト調整	21
4.6. デモアプリケーション - 音再生	21
4.7. デモアプリケーション - SSD 接続状態確認	22
5.1. Linux カーネル起動オプションのクリア	23
5.2. コンソールの指定	23
5.3. ルートファイルシステムの指定	24
5.4. 起動オプション設定例 1	25
5.5. 起動オプション設定例 2	25
5.6. 起動オプション設定例 3	25
6.1. コンフィグ領域の読出し方法	26
6.2. コンフィグ領域の保存方法	26
6.3. コンフィグ領域の初期化方法	27
7.1. 固定 IP アドレス設定	28
7.2. DHCP 設定	29
7.3. DNS サーバの設定	29
7.4. 設定を反映させる	29
7.5. PING 確認	29
7.6. iptables	30
7.7. telnet	30
7.8. ftp	30
7.9. ssh	31
7.10. msntp	31
8.1. ディスク初期化方法	32
8.2. ファイルシステムの構築	33
8.3. マウント方法	33
9.1. サウンドの再生	35
9.2. サウンドの録音	36
9.3. 音量の変更	36
10.1. ledctrl コマンドフォーマット	37
10.2. ledctrl 使用例 1	37
10.3. ledctrl 使用例 2	38
10.4. ledctrl 使用例 3	38
10.5. ledctrl 使用例 4	38
10.6. GPIO : Direction を INPUT にする	38
10.7. GPIO : INPUT 時の入力状態を取得する	39
10.8. Direction を OUTPUT にする	39
10.9. OUTPUT 時の出力状態を変更する	39
10.10. GPIO のアクセス権限を確認する	39

表目次

1.1. 使用しているフォント	8
1.2. 表示プロンプトと実行環境の関係	8
1.3. コマンド入力例での省略表記	8
2.1. シリアル通信設定	12
3.1. シリアルコンソールログイン時のユーザ名とパスワード	17
5.1. 起動モード	23
5.2. コンソール指定に伴う出力先	24
5.3. ルートファイルシステムデバイス	24
7.1. 固定 IP アドレス設定例	28
7.2. telnet でログイン可能なユーザ	30
7.3. ftp でログイン可能なユーザ	30
7.4. ssh でログイン可能なユーザ	31
8.1. ストレージデバイス	32
9.1. 録音/再生ソース一覧	34
9.2. オーディオコーデックドライバの仕様	34
10.1. ledctrl : command	37
10.2. GPIO ファイルノード	38

1.はじめに

このたびは Armadillo-500 FX 液晶モデルをお求めいただき、ありがとうございます。

Armadillo-500 FX 液晶モデルは、中核機能を持った「Armadillo-500 FX」と液晶やタッチパネル、ユーザインターフェースを実現する「インターフェースボード」から構成されています。Armadillo-500 FX は、Freescale 社製 ARM11 プロセッサ「i.MX31」、DDR SDRAM、フラッシュメモリを高集積に配置した高性能小型 CPU モジュール「Armadillo-500」を中心に、パネルコンピュータとしての機能が凝縮されています。インターフェースボードは、「ユーザインターフェース」を実現する部分（LCD の種類、ボタンの数、タッチパネルの種類など）と「外部インターフェース」を実現する部分（USB ポートやオーディオ入出力、SD スロットなど）で構成されています。

Armadillo-500 FX をそのまま利用しインターフェースボードだけをカスタマイズ開発することで、パネルコンピュータ開発時のハードウェアに対する様々な要求に短期間で対応することが可能となります。

Armadillo-500 FX は、標準 OS に Linux を採用していますので、Linux の豊富なソフトウェア資産を利用することができます。また、C などのプログラミング言語を使用し、オリジナルのプログラムを作成して動作させることも可能です。カスタマイズ方法については、「Armadillo-500 FX ソフトウェアマニュアル」等を参照してください。

本書には、ご利用にあたっての注意事項や、ご購入時の状態で利用できるソフトウェアの機能について記載されています。Armadillo-500 FX 液晶モデルがお手元に届きましたら、ハードウェアの動作確認、及びデフォルトソフトウェアの使用方法について確認いただくようお願い致します。

以降、本書では他の Armadillo シリーズにも共通する記述については、製品名を Armadillo と表記します。

1.1. 対象となる読者

- ・ ハードウェアの動作確認をされる方
- ・ ソフトウェアの基本的な使用方法の確認をされる方

上記以外の方でも、本書を有効に利用していただけたら幸いです。

1.2. 本書の構成

本書では、Armadillo の基本的な使用方法について記載されています。

以下に主な項目を挙げます。

- ・ 接続方法
- ・ 起動と終了
- ・ 各種設定方法
- ・ 各種アプリケーションの使用方法

1.3. 表記について

1.3.1. フォント

本書では以下のような意味でフォントを使いわけています。

表 1.1 使用しているフォント

フォント例	説明
本文中のフォント	本文
[PC ~]\$ ls	プロンプトとユーザ入力文字列
text	編集する文字列や出力される文字列。またはコメント

1.3.2. コマンド入力例

本書に記載されているコマンドの入力例は、表示されているプロンプトによって、それぞれに対応した実行環境を想定して書かれています。「/」の部分はカレントディレクトリによって異なります。各ユーザのホームディレクトリは「~」で表わします。

表 1.2 表示プロンプトと実行環境の関係

プロンプト	コマンドの実行環境
[PC /]#	作業用 PC 上の root ユーザで実行
[PC /]\$	作業用 PC 上の一般ユーザで実行
[armadillo /]#	Armadillo 上の root ユーザで実行
[armadillo /]\$	Armadillo 上の一般ユーザで実行
hermit>	Armadillo 上の保守モードで実行

コマンド中で、変更の可能性のあるものや、環境により異なるものに関しては以下のように表記します。適時読み替えて入力してください。

表 1.3 コマンド入力例での省略表記

表記	説明
[version]	ファイルのバージョン番号

1.3.3. アイコン

本書では以下のようにアイコンを使用しています。



注意事項を記載します。



役に立つ情報を記載します。

1.4. 謝辞

Armadillo で使用しているソフトウェアは Free Software / Open Source Software で構成されています。Free Software / Open Source Software は世界中の多くの開発者の成果によってなっています。この場を借りて感謝の意を表します。

1.5. 安全に関する注意事項

本製品を安全にご使用いただくために、特に以下の点にご注意くださいますようお願いいたします。



本製品には一般電子機器用(OA機器・通信機器・計測機器・工作機械等)に製造された半導体部品を使用しておりますので、その誤作動や故障が直接生命を脅かしたり、身体・財産等に危害を及ぼす恐れのある装置(医療機器・交通機器・燃焼制御・安全装置等)には使用しないでください。また、半導体部品を使用した製品は、外来ノイズやサージにより誤作動や故障する可能性があります。ご使用になる場合は万一誤作動、故障した場合においても生命・身体・財産等が侵害されることのないよう、装置としての安全設計(リミットスイッチやヒューズ・ブレーカ等の保護回路の設置、装置の多重化等)に万全を期されますようお願い申し上げます。

1.6. 取り扱い上の注意事項

本製品に恒久的なダメージをあたえないよう、取り扱い時には以下のような点にご注意ください。

- | | |
|--------------|--|
| 本製品の改造 | 本製品に改造 ¹ を行った場合、また CPU モジュール基盤の着脱を行なった場合は保証対象外となりますので十分ご注意ください。また、改造やコネクタ等の増設 ² を行う場合は、作業前に必ず動作確認を行ってください。 |
| 電源投入時のコネクタ着脱 | 本製品や周辺回路に電源が入っている状態で、活線挿抜対応インターフェース(LAN, USB, WSIM, SD, MIC, HP)以外へのコネクタ着脱は、絶対に行わないでください。 |
| 静電気 | 本製品には CMOS デバイスを使用していますので、ご使用になる時までは、帯電防止対策された出荷時のパッケージ等にて保管してください。 |
| ラッチアップ | 電源および入出力からの過大なノイズやサージ、電源電圧の急激な変動等により、使用している CMOS デバイスがラッチアップを起こす可能性があります。いったんラッチアップ状態となると、電源を切断しないかぎりこの状態が維持されるため、デバイスの破損につながる可能性があります。ノイズの影響を受けやすい入出力ラインには、保護回路を入れることや、ノイズ源となる装置と共通の電源を使用しない等の対策をとることをお勧めします。 |

¹ コネクタ非搭載箇所へのコネクタ等の増設は除く。

² コネクタを増設する際にはマスキングを行い、周囲の部品に半田くず、半田ボール等付着しないよう十分にご注意ください。

衝撃 落下や衝撃などの強い振動を与えないでください。

1.7. ソフトウェア使用に関する注意事項

本製品に含まれるソフトウェアについて 本製品に含まれるソフトウェア(付属のドキュメント等も含みます)は、現状のまま(AS IS)提供されるものであり、特定の目的に適合することや、その信頼性、正確性を保証するものではありません。また、本製品の使用による結果についてもなんら保証するものではありません。

1.8. 商標について

Armadillo は株式会社アットマークテクノの登録商標です。その他の記載の商品名および会社名は、各社・各団体の商標または登録商標です。

2.作業の前に

2.1. 見取り図

Armadillo-500 FX 液晶モデルの見取り図です。各インターフェースの配置場所等を確認してください。

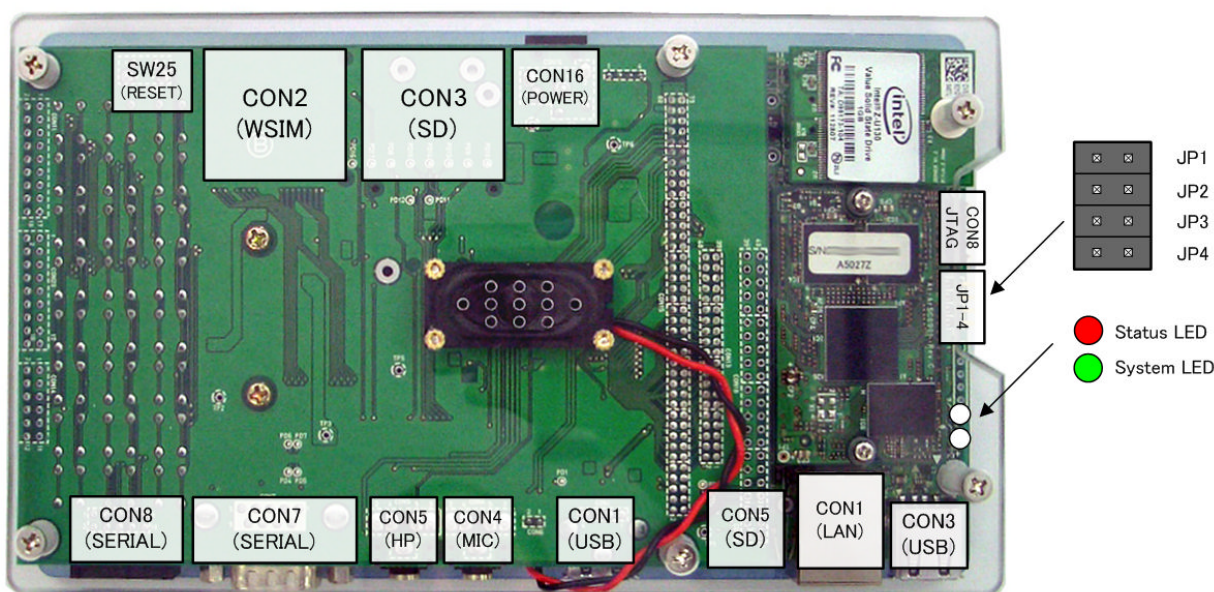


図 2.1 見取り図

2.2. 準備するもの

Armadillo を使用する前に、次のものを準備してください。

作業用 PC とシリアルクロスケーブル Linux または Windows が動作し、1 ポート以上のシリアルインターフェースを持つ PC と D-Sub9 ピン（メス - メス）のクロス接続用ケーブルです。作業用 PC にはシリアル通信ソフトウェア¹をインストールしてください。（Linux 用のソフトウェアは付属 CD の tool ディレクトリに収録されています）

ネットワーク環境 Armadillo と作業用 PC をネットワーク通信ができるようにしてください。

2.3. 接続方法

Armadillo の接続例です。

¹Linux では「minicom」、Windows では「Tera Term Pro」などです。

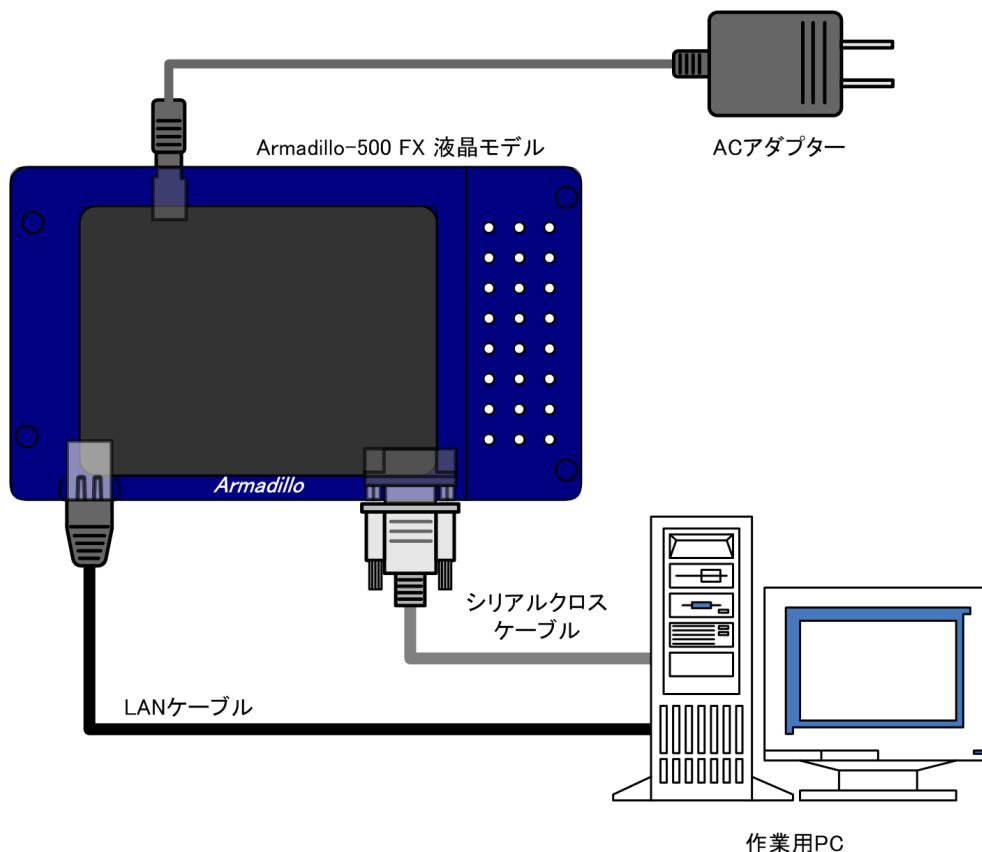


図 2.2 接続図

2.4. ジャンパピンの設定について

本書では JP2 と JP4 以外は操作することはありません。JP1 と JP3 は常にオープンに設定してください。JP2 と JP4 については、必要に応じて切り替えの指示があります。ここでは、オープンに設定しておきます。

また、ジャンパピンの位置は「図 2.1. 見取り図」で確認することができます。

2.5. シリアル通信ソフトウェアの設定

シリアル通信ソフトウェアを起動し、シリアルの通信設定を、「表 2.1. シリアル通信設定」のように設定してください。



Armadillo-240 では、RS232C レベル変換アダプターを経由させる必要があります。

表 2.1 シリアル通信設定

項目	設定
転送レート	115,200bps

項目	設定
データ長	8bit
ストップビット	1bit
パリティ	なし
フロー制御	なし

3.起動と終了

3.1. 起動

Armadillo の電源を投入してください。次のように起動ログがシリアル通信ソフトウェアに表示されま

す。

```
Hermit-At v1.1.20 (Armadillo-500/fx) compiled at 20:30:31, Oct 14 2008
Uncompressing kernel.....done.
Uncompressing ramdisk.....done.
Linux version 2.6.26-at1 (2.6.26) (build@sv-build) (gcc version 4.1.2 20061115
(prerelease) (Debian 4.1.1-21)) #1 PREEMPT Wed Oct 22 20:49:03 JST 2008
CPU: ARMv6-compatible processor [4107b364] revision 4 (ARMv6TEJ), cr=00e5387f
Machine: Armadillo-500 FX
Memory policy: ECC disabled, Data cache writeback
CPU0: D VIPT write-back cache
CPU0: I cache: 16384 bytes, associativity 4, 32 byte lines, 128 sets
CPU0: D cache: 16384 bytes, associativity 4, 32 byte lines, 128 sets
Built 1 zonelists in Zone order, mobility grouping on. Total pages: 32512
Kernel command line: console=ttymx0,115200 usb-storage.delay_use=0
MXC IRQ initialized
PID hash table entries: 512 (order: 9, 2048 bytes)
MXC GPT timer initialized, rate = 60000000
Console: colour dummy device 80x30
Dentry cache hash table entries: 16384 (order: 4, 65536 bytes)
Inode-cache hash table entries: 8192 (order: 3, 32768 bytes)
Memory: 128MB = 128MB total
Memory: 107656KB available (3104K code, 198K data, 128K init)
Mount-cache hash table entries: 512
CPU: Testing write buffer coherency: ok
net_namespace: 480 bytes
NET: Registered protocol family 16
L2X0 cache controller enabled
CPU is i.MX31 Revision 2.0
Clock input source is 26000000
MXC GPIO hardware
Using SDMA I.API
MXC DMA API initialized
SCSI subsystem initialized
CSPI: mxc_spi-2 probed
usbcore: registered new interface driver usbfs
usbcore: registered new interface driver hub
usbcore: registered new device driver usb
MXC I2C driver
MXC I2C driver
```

```
clk: Unable to get requested clock: dfm_clk
NET: Registered protocol family 2
IP route cache hash table entries: 1024 (order: 0, 4096 bytes)
TCP established hash table entries: 4096 (order: 3, 32768 bytes)
TCP bind hash table entries: 4096 (order: 2, 16384 bytes)
TCP: Hash tables configured (established 4096 bind 4096)
TCP reno registered
NET: Registered protocol family 1
checking if image is initramfs...it isn't (bad gzip magic numbers); looks like an
initrd
Freeing initrd memory: 18801K
usb: Host 2 host (isp1504) registered
usb: Host 1 host (serial) registered
usb: OTG host (isp1504) registered
Low-Level PM Driver module loaded
NetWinder Floating Point Emulator V0.97 (double precision)
DPTC driver module loaded
msgmni has been set to 247
io scheduler noop registered
io scheduler cfq registered (default)
MXC/iMX Backlight Driver
Console: switching to colour frame buffer device 80x30
mxcfb: fb registered, using mode KYOCERA-VGA
Serial: MXC Internal UART driver
mxcintuart.0: ttymxc0 at MMIO 0x43f90000 (irq = 45) is a Freescale MXC
console [ttymxc0] enabled
mxcintuart.1: ttymxc1 at MMIO 0x43f94000 (irq = 32) is a Freescale MXC
mxcintuart.4: ttymxc4 at MMIO 0x43fb4000 (irq = 47) is a Freescale MXC
brd: module loaded
loop: module loaded
wsim_pm: W-SIM Power Management for i.MX31
PPP generic driver version 2.4.2
smc911x: Driver version 2008-06-02.
eth0: SMSC911x MAC Address: 00:11:0c:00:99:88
Driver 'sd' needs updating - please use bus_type methods
armadillo5x0-nor: Found 1 x16 devices at 0x0 in 16-bit bank
  Intel/Sharp Extended Query Table at 0x010A
  Intel/Sharp Extended Query Table at 0x010A
  Intel/Sharp Extended Query Table at 0x010A
  Intel/Sharp Extended Query Table at 0x010A
  Intel/Sharp Extended Query Table at 0x010A
Using buffer write method
Using auto-unlock on power-up/resume
cfi_cmdset_0001: Erase suspend on write enabled
armadillo5x0-nor: use default partitions(4)
Creating 4 MTD partitions on "armadillo5x0-nor":
0x00000000-0x00020000 : "nor.bootloader"
0x00020000-0x00220000 : "nor.kernel"
0x00220000-0x01fe0000 : "nor.userland"
0x01fe0000-0x02000000 : "nor.config"
fsl-ehci fsl-ehci.0: Freescale On-Chip EHCI Host Controller
fsl-ehci fsl-ehci.0: new USB bus registered, assigned bus number 1
fsl-ehci fsl-ehci.0: irq 36, io mem 0x43f88400
fsl-ehci fsl-ehci.0: USB 2.0 started, EHCI 1.00, driver 10 Dec 2004
usb usb1: configuration #1 chosen from 1 choice
hub 1-0:1.0: USB hub found
hub 1-0:1.0: 1 port detected
Initializing USB Mass Storage driver...
```

```
usbcore: registered new interface driver usb-storage
USB Mass Storage support registered.
MXC keypad loaded
input: mxckpd as /devices/virtual/input/input0
ads7846 spi3.1: touchscreen, irq 130
input: ADS784x Touchscreen as /devices/platform/mxc_spi.2/spi3.1/input/input1
rtc-s353xxa 1-0030: rtc core: registered S-353XXA as rtc0
rtc-s353xxa: S-353XXA Real Time Clock class driver, (C) 2008 Atmark Techno, Inc.
i2c /dev entries driver
MXC MMC/SD driver
mxcmci-0 found
mxcmci-1 found
Registered led device: status
usb 1-1: new high speed USB device using fsl-ehci and address 2
usb 1-1: configuration #1 chosen from 1 choice
scsi0 : SCSI emulation for USB Mass Storage devices
usbcore: registered new interface driver usbhid
usbhid: v2.6:USB HID core driver
SSI module loaded successfully
Advanced Linux Sound Architecture Driver Version 1.0.16.
ASoC version 0.13.2
wm8978: WM8978 Audio Codec v0.2
scsi 0:0:0:0: Direct-Access      Intel      Value SSD          2.00 PQ: 0 ANSI: 2
asoc: WM8978 HiFi <-> imx-i2s-1 mapping ok
sd 0:0:0:0: [sda] 2006016 512-byte hardware sectors (1027 MB)
sd 0:0:0:0: [sda] Write Protect is off
sd 0:0:0:0: [sda] Assuming drive cache: write through
sd 0:0:0:0: [sda] 2006016 512-byte hardware sectors (1027 MB)
sd 0:0:0:0: [sda] Write Protect is off
sd 0:0:0:0: [sda] Assuming drive cache: write through
sda: sda1
sd 0:0:0:0: [sda] Attached SCSI disk
sd 0:0:0:0: Attached scsi generic sg0 type 0
ALSA device list:
 #0: armadillo500fx (WM8978)
ip_tables: (C) 2000-2006 Netfilter Core Team
TCP cubic registered
NET: Registered protocol family 17
NET: Registered protocol family 15
VFP support v0.3: implementor 41 architecture 1 part 20 variant b rev 2
rtc-s353xxa 1-0030: setting system clock to 2000-01-02 00:43:28 UTC (946773808)
RAMDISK: ext2 filesystem found at block 0
RAMDISK: Loading 18801KiB [1 disk] into ram disk... done.
VFS: Mounted root (ext2 filesystem).
Freeing init memory: 128K
init started: BusyBox v1.00 (2008.10.22-11:49+0000) multi-call binary
Starting fsck for root filesystem.
fsck 1.25 (20-Sep-2001)
ext2fs_check_if_mount: No such file or directory while determining whether /dev/
ram0 is mounted.
/dev/ram0: clean, 866/1536 files, 17037/18801 blocks
Checking root filesystem: done
Remounting root rw: done
Mounting proc: done
Mounting usbfs: done
Mounting sysfs: done
Delayed device initialization: fsl-ehci fsl-ehci.1: Freescale On-Chip EHCI Host
Controller
```



```
fsl-ehci fsl-ehci.1: new USB bus registered, assigned bus number 2
fsl-ehci fsl-ehci.1: irq 35, io mem 0x43f88200
fsl-ehci fsl-ehci.1: USB 2.0 started, EHCI 1.00, driver 10 Dec 2004
usb usb2: configuration #1 chosen from 1 choice
hub 2-0:1.0: USB hub found
hub 2-0:1.0: 1 port detected
fsl-ehci fsl-ehci.2: Freescale On-Chip EHCI Host Controller
fsl-ehci fsl-ehci.2: new USB bus registered, assigned bus number 3
fsl-ehci fsl-ehci.2: irq 37, io mem 0x43f88000
fsl-ehci fsl-ehci.2: USB 2.0 started, EHCI 1.00, driver 10 Dec 2004
usb usb3: configuration #1 chosen from 1 choice
hub 3-0:1.0: USB hub found
hub 3-0:1.0: 1 port detected
done
Cleaning up system: done
Running local start scripts.
Starting udevd: done
Changing file permissions: done
Configure /home/ftp: done
Starting syslogd: done
Starting klogd: done
Loading /etc/config: done
Starting basic firewall: done
Setting hostname: done
Configuring network interfaces: info, udhcpc (v0.9.9-pre) started
eth0: SMC911x/921x identified at 0xc9000000, IRQ: 64
eth0: link down
debug, Sending discover...
eth0: link up, 100Mbps, full-duplex, lpa 0x45E1
debug, Sending discover...
debug, Sending select for 172.16.2.243...
info, Lease of 172.16.2.243 obtained, lease time 86400
done
Starting inetd: done
Starting sshd: done
Starting thttpd: done
Starting Xfbdev: done
Starting a500fx-demo: done
Mounting ramfs /home/ftp/pub: done

atmark-dist v1.21.0 (AtmarkTechno/Armadillo-500-FX.dev)
Linux 2.6.26-at1 [armv6l arch]

a500-fx login:
```

図 3.1 起動ログ

3.2. ログイン

起動が完了するとログインプロンプトが表示されます。「表 3.1. シリアルコンソールログイン時のユーザ名とパスワード」に示すユーザでログインすることができます。

表 3.1 シリアルコンソールログイン時のユーザ名とパスワード

ユーザ名	パスワード	権限
root	root	root ユーザ

ユーザ名	パスワード	権限
guest	(なし)	一般ユーザ

3.3. 終了方法

安全に終了させる場合は、次のようにコマンドを実行し、「System halted.」と表示されたのを確認してから電源を切断します。

```
[armadillo ~]# halt
[armadillo ~]#
System is going down for system reboot now.

Starting local stop scripts.
Exiting Syslogd!
Syncing all filesystems:           done
Unmounting all filesystems:       done
The system is going down NOW !!
Sending SIGTERM to all processes.
The system is halted. Press Reset or turn off power
MXC Watchdog shutdown
System halted.
```

図 3.2 終了方法

リムーバブルディスクを使用していない場合は、電源を切断し終了させることもできます。



リムーバブルディスクにデータを書き込んでいる途中で電源を切断した場合、ファイルシステム、及び、データが破損する恐れがあります。リムーバブルディスクをアンマウントしてから電源を切断するようにご注意ください。

4. デモアプリケーション

この章では、Armadillo の電源を投入すると液晶パネルで表示されるデモアプリケーションについて説明します。

4.1. 機能選択

デモアプリケーションの各種機能は、画面の下に並んでいるボタンを押して選択します。

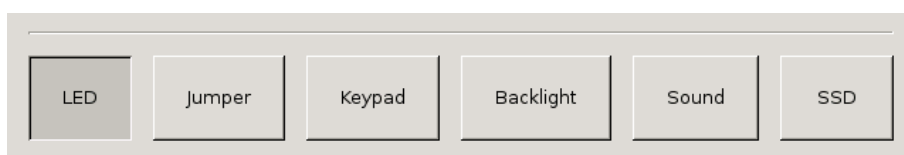


図 4.1 デモアプリケーション - 機能選択メニュー

4.2. LED 制御

「LED Status Control」画面では、Armadillo の Status LED¹を制御します。

「ON」ボタンを押すと LED が点灯します。「OFF」ボタンを押すと LED が消灯します。

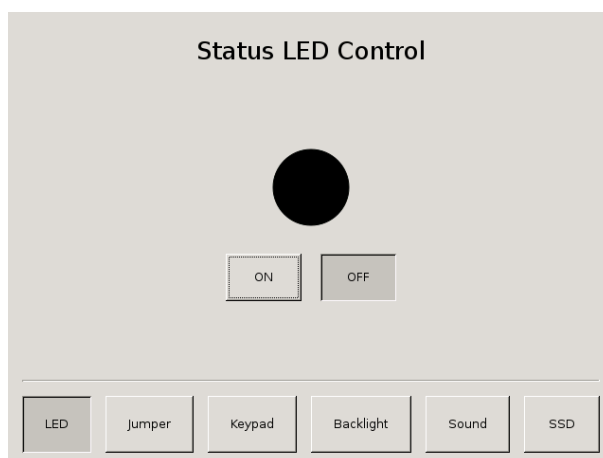


図 4.2 デモアプリケーション - LED 制御

¹LED の位置は「図 2.1. 見取り図」を参照してください。

4.3. ジャンパー状態確認

「Jumper 2 Status」画面では、Armadillo の JP2 の状態を確認できます。

JP2 がショートされている場合、画面の中央に「SHORTED」が表示されます。JP2 がオープンの場合、「OPEN」が表示されます。

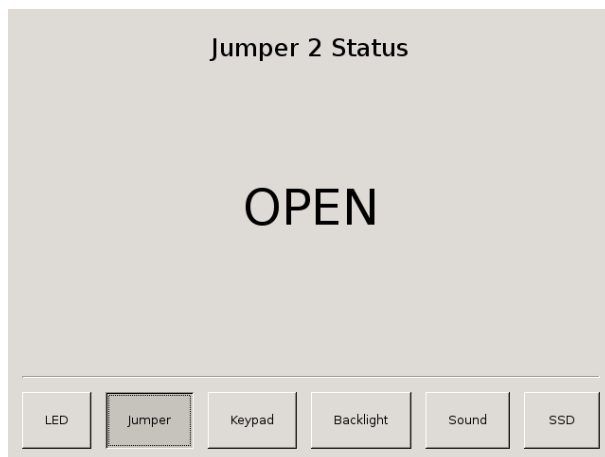


図 4.3 デモアプリケーション - ジャンパー状態確認

4.4. キーパッド動作確認

「Keypad Monitor」画面では、Armadillo のキーパッドの反応を確認できます。

キーパッドを押すと画面上の対応するボタンが押された状態になり、キーパッドを離すとボタンが元に戻ります。一度押されたボタンは、その後赤く表示されます。

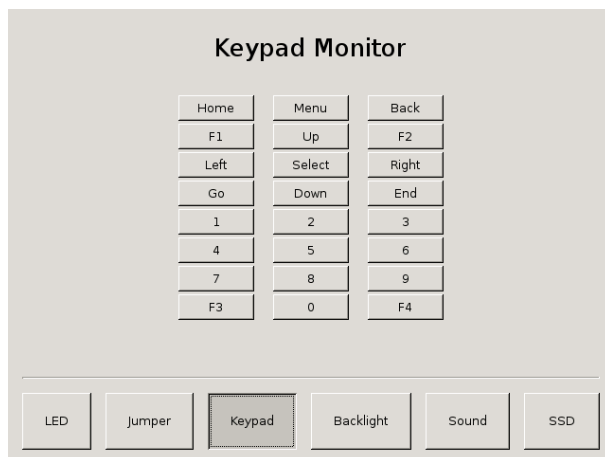


図 4.4 デモアプリケーション - キーパッド動作確認

4.5. バックライト調整

「Backlight Control」画面では、Armadillo の液晶パネルバックライトの明るさを調整できます。

スライダーでバックライトの明るさを調整します。左に移動させるとバックライトが暗くなり、右に移動させると明るくなります。

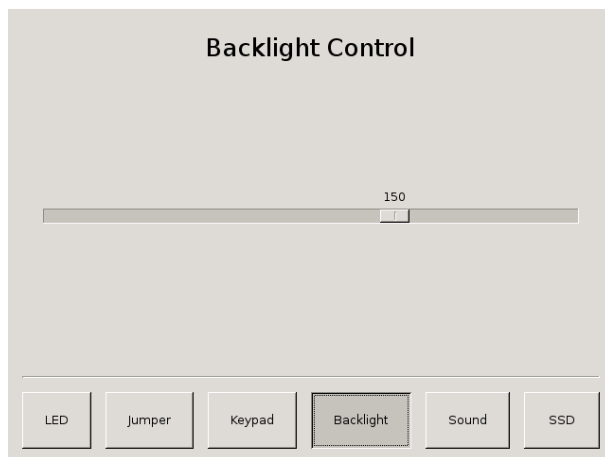


図 4.5 デモアプリケーション - バックライト調整

4.6. 音再生

「Sound Test」画面では、Armadillo のスピーカーから音を鳴らすことができます。

「Play」ボタンを押すと音が再生されます。

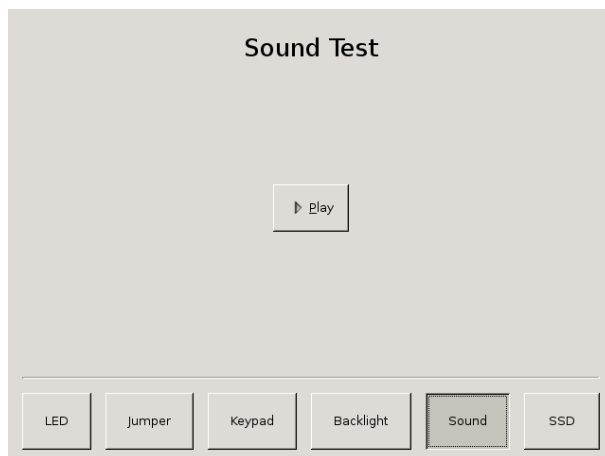


図 4.6 デモアプリケーション - 音再生

4.7. SSD 接続状態確認

「SSD Status」画面では、Armadillo の USB 接続 SSD の接続状態を確認できます。

USB 接続 SSD が接続されている場合、画面の中央に「Connected」が表示され、非接続の場合「Disconnected」が表示されます。

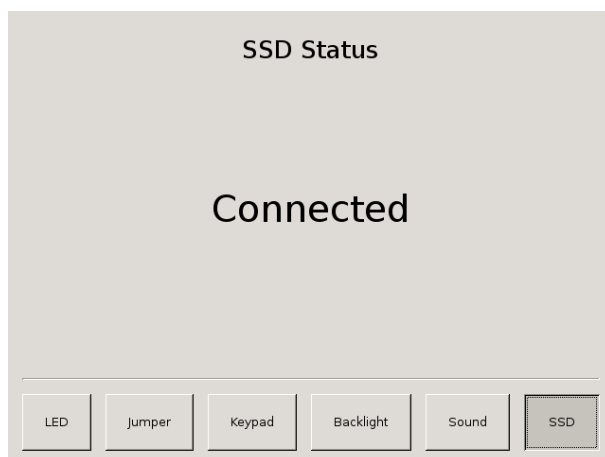


図 4.7 デモアプリケーション - SSD 接続状態確認

5. 起動モードとブートローダの機能

この章では、Armadillo の起動モードと採用しているブートローダ「Hermit-At」の起動設定機能について説明します。

起動モードには、オートブートモード、保守モード等、システム起動時に最初に動作するソフトウェアを選択することができます。

Hermit-At では、Linux カーネルを起動させる時の起動オプションの設定、クロックの設定等、システム起動時の初期設定を行うことができます。

5.1. 起動モードの選択

起動モードの設定は、JP4 の設定により決定されます。各起動モードは「表 5.1. 起動モード」のようになります。

表 5.1 起動モード

モード	JP4	説明
オートブート	オープン	電源投入後、自動的に Linux カーネルを起動させます。
保守	ショート	各種設定が可能な Hermit At コマンドプロンプトが起動します。

5.2. Linux カーネル起動オプションの設定

Linux カーネル起動オプションを変更することで、コンソールや、ルートファイルシステム等の様々な種類の設定を変更することができます。ここでは、Armadillo に関係のある代表的なオプションについて説明します。

また、これらの設定は、Hermit-At の `setenv` 機能を使用します。`setenv` で設定されたパラメータはフラッシュメモリに保存され再起動後も設定が反映されます。

設定されたパラメータをクリアするには、`clearenv` を使用します。

```
hermit> clearenv
```

図 5.1 Linux カーネル起動オプションのクリア

5.2.1. コンソールの設定

起動ログの出力コンソールを変更するには、下記のように `console` パラメータにコンソール指定子を設定します。

```
hermit> setenv console=ttymxc0
```

図 5.2 コンソールの指定

設定によるログの出力先は、「表 5.2. コンソール指定に伴う出力先」のようになります。

表 5.2 コンソール指定に伴う出力先

コンソール指定子	起動ログ出力先
ttymxc0	CON7
ttymxc4	CON8
null	なし
その他 (tty1 等)	指定するコンソール

5.2.2. ルートファイルシステムの設定

ルートファイルシステムとしてマウントするファイルシステムイメージの場所や、マウントするファイルシステム等を設定します。各パラメータの意味は以下を参照してください。

root

使用するルートファイルシステムのパーティションを指定します。

rootfs

使用するルートファイルシステムのタイプを指定します。

rootwait

ルートファイルシステムがアクセス可能になるまで待機します。

```
hermit> setenv root=/dev/sda1 rootfs=ext3 rootwait
```

図 5.3 ルートファイルシステムの指定

5.2.2.1. ルートファイルシステムイメージの場所

ファイルシステムイメージの場所を設定する場合は、イメージが存在するパーティションを設定します。各デバイスのパーティションノードの例を、「表 5.3. ルートファイルシステムデバイス」に示します。指定がない場合（デフォルト）は、RAM ディスク (/dev/ram0) が指定されます。

表 5.3 ルートファイルシステムデバイス

デバイス名	デバイスノード	先頭パーティションノード
RAM ディスク	/dev/ram	/dev/ram0
USB メモリ(SSD など)	/dev/sd*	/dev/sd*1
MMC/SD カードディスク	/dev/mmcblk*	/dev/mmcblk*p1

5.2.2.2. ルートファイルシステムタイプ

特異なファイルシステムを使用する場合は、ファイルシステムタイプを指定します。指定がない場合は、ext2、ext3、msdos、vfat のいずれかでマウントされます¹。

¹Linux カーネルが標準でサポートするファイルシステムの場合は、特に指定する必要はありません

5.2.2.3. ファイルシステム待機

Linux カーネルは、指定するルートファイルシステムが存在するデバイスの認識が完了していなければ、ルートファイルシステムをマウントすることはできません。ドライバのロードタイミングやデバイスに依存する時間等、デバイスの認識時間は様々な要素で変動します。

ここで指定することができるのは、ルートファイルシステムがアクセス可能になるまで待機するオプションとなります。指定がない場合（デフォルト）は、待機なしとなります。

5.2.3. その他の起動オプション

本書で紹介したオプション以外にも様々なオプションがあります。詳しくは、Linux の解説書や、Linux カーネルのソースコードに含まれるドキュメント（kernel-source/Documentation/kernel-parameters.txt）等を参照してください。

5.2.4. 起動オプションの設定例

- ・ コンソールを CON8 にする場合

```
hermit> setenv console=ttymxc4
```

図 5.4 起動オプション設定例 1

- ・ コンソールを液晶パネルにする場合

```
hermit> setenv console=tty1
```

図 5.5 起動オプション設定例 2

- ・ コンソールを CON7 へ出力、ルートファイルシステムを USB 接続 SSD のパーティション 1 にする場合

```
hermit> setenv console=ttymxc0 noinitrd root=/dev/sda1 rootwait
```

図 5.6 起動オプション設定例 3

6. コンフィグ領域 – 設定ファイルの保存領域

コンフィグ領域は、設定ファイルなどを保存しハードウェアのリセット後にもデータを保持することができるフラッシュメモリ領域です。コンフィグ領域からのデータの読出し、またはコンフィグ領域への書込みは、flatfsd コマンドを使用します。

6.1. コンフィグ領域の読出し

コンフィグ領域を読み出すには以下のコマンドを実行します。読み出されたファイルは、「/etc/config」ディレクトリに作成されます。

```
[armadillo ~]# flatfsd -r
```

図 6.1 コンフィグ領域の読出し方法



デフォルトのソフトウェアでは、起動時に自動的にコンフィグ領域の読出しを行うように設定されています。コンフィグ領域の情報が壊れている場合、「/etc/default」ディレクトリの内容が反映されます。

6.2. コンフィグ領域の保存

コンフィグ領域を保存するには以下のコマンドを実行します。保存されるファイルは、「/etc/config」ディレクトリ以下のファイルです。

```
[armadillo ~]# flatfsd -s
```

図 6.2 コンフィグ領域の保存方法



コンフィグ領域の保存をおこなわない場合、「/etc/config」ディレクトリ以下のファイルへの変更は電源遮断時に失われます。

6.3. コンフィグ領域の初期化

コンフィグ領域を初期化するには以下のコマンドを実行します。初期化時には、「/etc/default」ディレクトリ以下のファイルがコンフィグ領域に保存され、且つ「/etc/config」ディレクトリにファイルが複製されます。

```
[armadillo ~]# flatfsd -w
```

図 6.3 コンフィグ領域の初期化方法

7. ネットワーク

この章では、ネットワークの設定方法やネットワークを使用するアプリケーションの使用方法について説明します。

7.1. ネットワークの設定

Armadillo 上の「/etc/config」以下にあるファイルを編集し、コンフィグ領域を保存することにより起動時のネットワーク設定を変更することができます。コンフィグ領域の保存については、「6. コンフィグ領域 – 設定ファイルの保存領域」を参照してください。

また、出荷時のネットワーク設定は、DHCP となっています。



ネットワーク接続に関する不明な点については、ネットワークの管理者へ相談してください。

7.1.1. 固定 IP アドレスに設定する

「表 7.1. 固定 IP アドレス設定例」に示す内容に設定変更するには、vi エディタで /etc/config/interfaces を、「図 7.1. 固定 IP アドレス設定」のように編集します。

表 7.1 固定 IP アドレス設定例

項目	設定
IP アドレス	192.168.10.10
ネットマスク	255.255.255.0
ネットワークアドレス	192.168.10.0
ブロードキャストアドレス	192.168.10.255
デフォルトゲートウェイ	192.168.10.1

```
[armadillo ~]# vi /etc/config/interfaces
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

auto lo eth0
iface lo inet loopback
iface eth0 inet static
    address 192.168.10.10
    netmask 255.255.255.0
    network 192.168.10.0
    broadcast 192.168.10.255
    gateway 192.168.10.1
```

図 7.1 固定 IP アドレス設定

7.1.2. DHCP に設定する

DHCP に設定するには、vi エディタで `/etc/config/interfaces` を、「図 7.2. DHCP 設定」のように編集します。

```
[armadillo ~]# vi /etc/config/interfaces
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

auto lo eth0
iface lo inet loopback
iface eth0 inet dhcp
```

図 7.2 DHCP 設定

7.1.3. DNS サーバを指定する

DNS サーバを指定する場合は、vi エディタで `/etc/config/resolv.conf` を編集します。

```
[armadillo ~]# vi /etc/config/resolv.conf
nameserver 192.168.10.1
```

図 7.3 DNS サーバの設定

7.1.4. 接続を確認する

ここでは、変更した IP 設定で正常に通信が可能か確認します。

まず、設定を反映させます。設定後、コンフィグ領域を保存し再起動した場合は必要ありません。

```
[armadillo ~]# ifdown -a
[armadillo ~]# ifup -a
```

図 7.4 設定を反映させる

同じネットワーク内にある通信機器と PING 通信を行ってみます。

```
[armadillo ~]# ping 192.168.10.1
```

図 7.5 PING 確認

7.2. ファイアウォール

Armadillo では、簡易ファイアウォールが動作しています。設定されている内容を参照するには、「図 7.6. iptables」のようにコマンド実行してください。

```
[armadillo ~]# iptables --list
```

図 7.6 iptables

7.3. ネットワークアプリケーション

ここでは、出荷時に収録されているソフトウェアのうちネットワークに関するアプリケーションの操作方法を説明します。

7.3.1. TELNET

7.3.1.1. TELNET サーバ

他の PC からネットワーク経由でログインし、リモート操作が可能となります。「表 7.2. telnet でログイン可能なユーザ」に示すユーザでログインすることができます。

表 7.2 telnet でログイン可能なユーザ

ユーザ名	パスワード
guest	(なし)

7.3.1.2. TELNET クライアント

telnet を使用して、他の PC にリモートログインすることができます。telnet 使用するには、「図 7.7. telnet」のようにコマンドを実行します。

```
[armadillo ~]# telnet 192.168.10.1
```

図 7.7 telnet

7.3.2. FTP

7.3.2.1. FTP サーバ

他の PC からネットワーク経由でファイルの転送ができます。「表 7.3. ftp でログイン可能なユーザ」に示すユーザでログインすることが可能です。

表 7.3 ftp でログイン可能なユーザ

ユーザ名	パスワード
ftp	(なし)

7.3.2.2. FTP クライアント

ftp を使用して、他の PC とファイル転送ができます。ftp を使用するには、「図 7.8. ftp」のようにコマンドを実行してください。

```
[armadillo ~]# ftp 192.168.10.1
```

図 7.8 ftp

7.3.3. SSH

7.3.3.1. SSH サーバ

他の PC からネットワーク経由でログインし、安全にリモート操作が可能となります。「表 7.4. ssh でログイン可能なユーザ」に示すユーザでログインすることができます。

表 7.4 ssh でログイン可能なユーザ

ユーザ名	パスワード
guest	(なし)

7.3.3.2. SSH クライアント

ssh を使用して、安全に他の PC へリモートログインすることができます。ssh を使用するには、「図 7.9. ssh」のようにコマンドを実行します。

```
[armadillo ~]# ssh user@192.168.10.1
```

図 7.9 ssh

7.3.4. Web サーバ

Armadillo では、Web サーバが動作しています。PC などの Web ブラウザから Armadillo の URL (http://[Armadillo の IP アドレス])¹ にアクセスすると、「/home/www-data」以下のディレクトリをブラウズすることが出来ます。

7.3.5. NTP クライアント

Armadillo では、SNTP (Simple Network Time Protocol) クライアントが使用できます。SNTP に対応しているタイムサーバから時刻情報を取得することができます。

「図 7.10. msntp」のようにコマンドを実行します。

```
[armadillo ~]# msntp -r 192.168.10.1
The time correction is 17180483.054 +/- 0.075+0.001 seconds
Do you want to correct the time anyway? y
```

図 7.10 msntp

¹Armadillo の IP アドレスが 192.168.10.10 の場合、http://192.168.10.10/ となります。

8. ストレージ

8.1. ストレージとして使用可能なデバイス

Armadillo では、「表 8.1. ストレージデバイス」に示すデバイスをストレージとして使用することができます。

表 8.1 ストレージデバイス

デバイス種類	ディスクデバイス	先頭パーティション
USB メモリ(SSD など)	/dev/sd*	/dev/sd*1
MMC/SD カード	/dev/mmcblk*	/dev/mmcblk*p1

8.2. ストレージの初期化とマウント

USB 接続 SSD (以降、SSD と表記) を例にとり、ストレージの初期化とマウント方法について説明します。

8.2.1. ディスクの初期化

ここでは、SSD にパーティションを 1 つ作ります。

```
[armadillo ~]# fdisk /dev/sda

Command (m for help): d
No partition is defined yet!

Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-1324, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-1011, default 1011):
Using default value 1011

Command (m for help): p

Disk /dev/sda: 1027 MB, 1027604480 bytes
32 heads, 62 sectors/track, 1011 cylinders
Units = cylinders of 1984 * 512 = 1015808 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            1         1011     1002881   83  Linux

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
```



```
sd 1:0:0:0: [sda] 2007040 512-byte hardware sectors (1028 MB)
sd 1:0:0:0: [sda] Write Protect is off
sd 1:0:0:0: [sda] Assuming drive cache: write through
sda: sda1
sd 1:0:0:0: [sda] 2007040 512-byte hardware sectors (1028 MB)
sd 1:0:0:0: [sda] Write Protect is off
sd 1:0:0:0: [sda] Assuming drive cache: write through
sda: sda1
Syncing disks.
```

図 8.1 ディスク初期化方法

8.2.2. ファイルシステムの構築

初期化したディスクのパーティションにファイルシステムを構築します。ここでは、SSD のパーティション 1 (`/dev/sda1`) に EXT3 を構築します。

```
[armadillo ~]# mke2fs -j /dev/sda1
mke2fs 1.25 (20-Sep-2001)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
125440 inodes, 250720 blocks
12536 blocks (5%) reserved for the super user
First data block=0
8 block groups
32768 blocks per group, 32768 fragments per group
15680 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376

Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 35 mounts or
180.00 days, whichever comes first. Use tune2fs -c or -i to override.
```

図 8.2 ファイルシステムの構築

8.2.3. マウント

ファイルシステムを作成後、「図 8.3. マウント方法」のようにコマンドを実行する事で、ext3 ファイルシステムを使用して `/dev/sda1` を `/mnt` にマウントします。

```
[armadillo ~]# mount -t ext3 /dev/sda1 /mnt
```

図 8.3 マウント方法

9. サウンド

Armadillo-500 FX 液晶モデルはオーディオコーデックを搭載しており、録音及び再生をおこなうことができます。

9.1. コーデック仕様

録音及び再生のソースとして「表 9.1. 録音/再生ソース一覧」に示す入出力を使用することができます。

表 9.1 録音/再生ソース一覧

ソース	インターフェースボード コネクタ	備考
マイク入力	CON4	デフォルト録音ソース
スピーカ出力	CON6	デフォルト再生ソース
ヘッドホン出力	CON5	

オーディオコーデックドライバは「表 9.2. オーディオコーデックドライバの仕様」に示す値に設定することができます。

表 9.2 オーディオコーデックドライバの仕様

チャンネル数	1 (monoral)/2 (stereo)
サンプリングフォーマット	Signed 16 bit, little-endian/Signed 20 bit, little-endian/Signed 24 bit, little-endian
サンプリング周波数	48000/44100/32000/22050/16000/11025/8000 Hz

9.2. OSS によるサウンド機能の実現

Linux でサウンド機能を実現するには、ALSA¹と OSS²の二つの方法があります。OSS は、ALSA の OSS エミュレーション³によって実現されます。OSS エミュレーションには、以下の二つの方法があります。

- ・ カーネルレベル OSS エミュレーション (snd-ocm-oss, snd-mixer-oss, snd-seq-oss を使用)
- ・ alsa-oss パッケージによるユーザ空間 OSS エミュレーション (aoss スクリプトを使用)

Armadillo-500 FX 液晶モデルのデフォルト設定では、カーネルレベル OSS エミュレーションによる OSS 機能を提供しています。

OSS プログラミングについては、「Open Sound System Programmer's Guide」⁴をご参照ください。

9.2.1. サウンドを再生する

OSS を使用したサウンドの再生には **vpplay** コマンドを使用します。

オプションの詳細は以下を参照してください。

¹Advanced Linux Sound Architecture <http://alsa.sourceforge.net>

²Open Sound System <http://developer.opensound.com/>

³<http://alsa.opensrc.org/OssEmulation>

⁴<http://www.opensound.com/pguide/oss.pdf>

-r

再生するファイルフォーマットに raw を指定します。

-w

再生するファイルフォーマットに wav を指定します。

-v

再生するファイルフォーマットに voc を指定します。

-S

ステレオ再生します。未指定の場合はモノラル再生になります。

-s

サンプリング周波数を指定します。300 以下を指定した場合は 1000 倍されます。未指定の場合は 8000 になります。

-b

サンプルサイズ(bit)を指定します。8 または 16 が指定可能です。未指定の場合は 8 になります。

-t

再生時間を秒単位で指定します。未指定の場合はファイルの最後まで再生されます。

filename

再生するサウンドファイル名を指定します。

```
[armadillo ~]# vplay -r -S -s 8000 -b 16 -t 5 filename
```

図 9.1 サウンドの再生

9.2.2. 録音する

OSS を使用したサウンドの録音には **vrec** コマンドを使用します。

オプションの詳細は以下を参照してください。

-r

録音するファイルフォーマットに raw を指定します。

-w

録音するファイルフォーマットに wav を指定します。

-v

録音するファイルフォーマットに voc を指定します。

-S

ステレオ録音します。未指定の場合はモノラル録音になります。

-s

サンプリング周波数を指定します。300 以下を指定した場合は 1000 倍されます。未指定の場合は 8000 になります。

-b

サンプルサイズ(bit)を指定します。8 または 16 が指定可能です。未指定の場合は 8 になります。

-t

録音 時間を秒単位で指定します。未指定の場合は無制限になります。

filename

録音データを出力するファイル名を指定します。

```
[armadillo ~]# vrec -r -S -s 8000 -b 16 -t 5 filename
```

図 9.2 サウンドの録音

9.2.3. 音量を変更する

OSS を使用して再生/録音時の音量を変更するには **mixer** コマンドを使用します。

[device]には、変更対象のデバイス名を指定します。有効なデバイス名は、pcm2(ヘッドホン出力)、phout(スピーカ出力)、ogain(出力倍率)、igain(入力倍率)です。

[value]には、音量を 0-100 の間で指定します。

```
[armadillo ~]# mixer [device] [value]
```

図 9.3 音量の変更

10. その他のデバイス

10.1. LED

10.1.1. ledctrl による制御

ledctrl では、Armadillo の赤い (status) LED を制御することができます。ledctrl は、「図 10.1. ledctrl コマンドフォーマット」のように使用します。

```
[armadillo ~]# ledctrl [ledname] [command]
```

図 10.1 ledctrl コマンドフォーマット

ledname には、status を指定します。

command には、LED に対しての制御方法を指定します。command 一覧は、「表 10.1. ledctrl : command」のとおりです。

表 10.1 ledctrl : command

command	動作
on	LED を点灯させます。
off	LED を消灯させます。
blink_on [interval_ms]	LED の点滅を開始します。点滅間隔は、interval_ms で指定します。デフォルトでは 200ms の点滅間隔となります。 また、点滅を開始する 1 つ前の状態を保存し、blink_off 時に状態を元に戻すことができます。
blink_off	LED の点滅を終了します。
status	現在の状態を表示します。表示する項目は、brightness、delay_on、delay_off となります。

10.1.2. ledctrl 使用例

- ・ 点灯させる

```
[armadillo ~]# ledctrl status on
```

図 10.2 ledctrl 使用例 1

- ・ 消灯させる

```
[armadillo ~]# ledctrl status off
```

図 10.3 ledctrl 使用例 2

- ・ 点滅を開始させる

```
[armadillo ~]# ledctrl status blink_on 500
```

図 10.4 ledctrl 使用例 3

- ・ 点滅を終了させる

```
[armadillo ~]# ledctrl status blink_off
```

図 10.5 ledctrl 使用例 4

10.2. GPIO

Armadillo の GPIO は、`/sys/devices/platform/armadillo5x0_gpio.0/ports` の下にあるファイルを読み書きすることにより、簡易的に制御することができます。

制御できる項目は、「表 10.2. GPIO ファイルノード」のとおりです。

表 10.2 GPIO ファイルノード

ファイル名	説明
gpio*_dir	GPIO の Direction を変更することができます。0 を書き込むと INPUT に、1 を書き込むと OUTPUT になります。それ以外は未定義となります。 読み出した場合は、現在の状態を取得できます。
gpio*_data	Direction が INPUT の場合は、GPIO の現在の入力レベルを読み出すことができます。書き込みは無視されます。Direction が OUTPUT の場合は、GPIO の出力レベルを設定できます。読み出した場合は、現在の出力レベルを取得できます。読み書きする値は、0 で LOW レベルを表し、1 で HIGH レベルを表します。それ以外は未定義となります。

10.2.1. Direction を INPUT にする

GPIO3_0 (CON11 1 ピン) を入力にする場合は、「図 10.6. GPIO : Direction を INPUT にする」のようにコマンドを実行します。

```
[armadillo ~]# echo 0 > /sys/devices/platform/armadillo5x0_gpio.0/ports/gpio3_0_dir
```

図 10.6 GPIO : Direction を INPUT にする

現在の GPIO3_0 の入力状態を取得するには、「図 10.7. GPIO : INPUT 時の入力状態を取得する」のようにコマンドを実行します。

```
[armadillo ~]# cat /sys/devices/platform/armadillo5x0_gpio.0/ports/gpio3_0_data
0
```

図 10.7 GPIO : INPUT 時の入力状態を取得する

10.2.2. Direction を OUTPUT にする

GPIO3_0 (CON11 1 ピン) を出力にする場合は、「図 10.8. Direction を OUTPUT にする」のようにコマンドを実行します。

```
[armadillo ~]# echo 1 > /sys/devices/platform/armadillo5x0_gpio.0/ports/
gpio3_0_dir
```

図 10.8 Direction を OUTPUT にする

HIGH レベルを出力するには、「図 10.9. OUTPUT 時の出力状態を変更する」のようにコマンドを実行します。

```
[armadillo ~]# echo 1 > /sys/devices/platform/armadillo5x0_gpio.0/ports/
gpio3_0_data
```

図 10.9 OUTPUT 時の出力状態を変更する

10.2.3. INPUT 専用の GPIO

OUTPUT 設定ができない GPIO は、Direction ファイルのアクセス権限が読み込み専用になります。アクセス権限は「図 10.10. GPIO のアクセス権限を確認する」のようにコマンドを実行して確認します。

```
[armadillo ~]# cd /sys/devices/platform/armadillo5x0_gpio.0/ports
[armadillo /sys/devices/platform/armadillo5x0_gpio.0/ports]# ls -l gpio3_0_dir
-rw-rw-r-- 1 root root 4096 Oct 18 07:23 gpio3_0_dir
[armadillo /sys/devices/platform/armadillo5x0_gpio.0/ports]# ls -l gpio1_8_dir
-r--r--r-- 1 root root 4096 Oct 18 07:23 gpio1_8_dir
```

図 10.10 GPIO のアクセス権限を確認する

改訂履歴

バージョン	年月日	改訂内容
1.0.0	2008/10/21	・ 初版発行
1.0.1	2009/01/09	・ 「6. コンフィグ領域 – 設定ファイルの保存領域」を修正 ・ 「9. サウンド」を追加 ・ 「図 5.3. ルートファイルシステムの指定」誤記修正
1.1.0	2009/03/18	・ 「図 2.1. 見取り図」を差し替え ・ 「1. はじめに」、「3.2. ログイン」、「3.3. 終了方法」、「2.5. シリアル通信ソフトウェアの設定」構成を変更 ・ 誤記、表記ゆれ修正
1.1.1	2009/07/17	・ 「5. 起動モードとブートローダの機能」の表記を変更 ・ 本文のレイアウト統一 ・ 誤記、表記ゆれ修正 ・ 「図 7.4. 設定を反映させる」のコマンドオプションの誤記を修正
1.1.2	2009/07/29	・ 製品保証に関する記載を http://www.atmark-techno.com/support/warranty-policy に移動(2009/08/03 適用)
1.2.0	2010/01/29	・ 表のレイアウト統一 ・ guest ユーザパスワードの誤記を修正 ・ 表紙に A546701-D00Z を追加 ・ 「1.6. 取り扱い上の注意事項」誤記修正 ・ 「図 2.1. 見取り図」部品番号の誤記修正

Armadillo-500 FX 液晶モデルスタートアップガイド
Version 1.2.0
2010/02/02

株式会社アットマークテクノ

060-0035 札幌市中央区北 5 条東 2 丁目 AFT ビル 6F TEL 011-207-6550 FAX 011-207-6570
