

Armadillo-810 カメラモデル開発セット スタートアップガイド

A8101-D00Z

Version 1.0.1
2013/02/19

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

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

Armadillo-810 カメラモデル開発セット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
TEL 011-207-6550 FAX 011-207-6570

製作著作 © 2013 Atmark Techno, Inc.

Version 1.0.1
2013/02/19

目次

1. はじめに	8
1.1. 本書および関連ファイルのバージョンについて	8
1.2. 対象となる読者	9
1.3. 本書の構成	9
1.4. 表記について	9
1.4.1. フォント	9
1.4.2. コマンド入力例	9
1.4.3. アイコン	10
1.5. 謝辞	10
2. 注意事項	11
2.1. 安全に関する注意事項	11
2.2. 取扱い上の注意事項	12
2.3. ソフトウェア使用に関する注意事項	12
2.4. 書き込み禁止領域について	13
2.5. 電波障害について	13
2.6. 保証について	13
2.7. 輸出について	13
2.8. 商標について	13
3. 作業の前に	14
3.1. 準備するもの	14
3.2. 開発/動作確認環境の構築	14
3.2.1. ATDE5 セットアップ	15
3.2.2. 取り外し可能デバイスの使用	16
3.2.3. コマンドライン端末(GNOME 端末)の起動	17
3.2.4. シリアル通信ソフトウェア(minicom)の使用	18
3.3. インターフェースレイアウト	19
3.3.1. Armadillo-810 CPU ボード インターフェースレイアウト	19
3.3.2. Armadillo-810 拡張ボード 01 (A コネクタ用) インターフェースレイアウト	20
3.3.3. Armadillo-810 カメラモジュール 01 (B コネクタ用) インターフェースレイアウト	21
3.4. 組み立て	22
3.4.1. Armadillo-810 カメラモデルの組み立て	22
3.4.2. レンズの交換方法	25
3.5. 接続方法	25
3.6. ジャンパピンの設定について	27
3.7. スライドスイッチの設定について	27
3.8. vi エディタの使用方法	28
3.8.1. vi の起動	28
3.8.2. 文字の入力	29
3.8.3. カーソルの移動	29
3.8.4. 文字の削除	30
3.8.5. 保存と終了	30
4. 起動と終了	31
4.1. 起動	31
4.2. ログイン	35
4.3. 終了方法	36
5. 動作確認方法	37
5.1. USB ガジェット	37
5.1.1. UVC ガジェット	37
5.1.2. シリアルガジェット	39

5.1.3. イーサネットガジェット	40
5.2. ネットワーク	41
5.2.1. ネットワーク設定の変更方法	41
5.2.2. ファイアウォール	44
5.2.3. ネットワークアプリケーション	44
5.3. シリアル	47
5.3.1. シリアルコンソールとして使用する	47
5.3.2. ファイルを転送する	48
5.4. ストレージ	51
5.4.1. ストレージの使用法	51
5.4.2. ストレージのパーティション変更とフォーマット	52
5.5. LED	53
5.5.1. LED を点灯/消灯する	53
5.5.2. トリガを使用する	54
5.6. GPIO	55
5.6.1. 入出力方向を変更する	55
5.6.2. 入力レベルを取得する	56
5.6.3. 出力レベルを設定する	56
5.7. RTC	56
5.7.1. RTC に時刻を設定する	56
6. コンフィグ領域 – 設定ファイルの保存領域	58
6.1. コンフィグ領域の読出し	58
6.2. コンフィグ領域の保存	58
6.3. コンフィグ領域の初期化	58
7. ユーザー登録	60
7.1. 購入製品登録	60

目次

3.1. GNOME 端末の起動	17
3.2. GNOME 端末のウィンドウ	18
3.3. minicom 設定方法	18
3.4. minicom 起動方法	18
3.5. minicom 終了確認	19
3.6. Armadillo-810 CPU ボードのインターフェースレイアウト図	19
3.7. Armadillo-810 拡張ボード 01 (A コネクタ用)のインターフェースレイアウト図	20
3.8. Armadillo-810 カメラモジュール 01 (B コネクタ用)のインターフェースレイアウト図	21
3.9. Armadillo-810 カメラモデルの組み立て	22
3.10. コネクタ嵌合時の取扱い上の注意 1	23
3.11. コネクタ嵌合時の取扱い上の注意 2	23
3.12. コネクタ嵌合時の取扱い上の注意 3	23
3.13. コネクタ抜去時の取扱い上の注意 1	23
3.14. コネクタ抜去時の取扱い上の注意 2	24
3.15. コネクタ抜去時の取扱い上の注意 3	24
3.16. ねじ締め時の注意事項 1	24
3.17. ねじ締め時の注意事項 2	24
3.18. カメラレンズの交換	25
3.19. 接続図	26
3.20. スライドスイッチの設定	28
3.21. vi の起動	28
3.22. 入力モードに移行するコマンドの説明	29
3.23. 文字を削除するコマンドの説明	30
4.1. 起動ログ	31
4.2. 終了方法	36
5.1. guvcview を起動	37
5.2. guvcview のビデオウィンドウ	38
5.3. guvcview のコントロールウィンドウ	38
5.4. /dev/ttyACM0 を指定してシリアルターミナルを起動	39
5.5. /dev/ttyGSO 上でシリアルコンソールを起動	39
5.6. イーサネットガジェット認識時の ifconfig の出力例	40
5.7. イーサネットガジェットの通信確認	41
5.8. デフォルト状態の/etc/config/interfaces	42
5.9. 固定 IP アドレス設定	42
5.10. DHCP 設定	43
5.11. DNS サーバーの設定	43
5.12. 設定を反映させる	43
5.13. PING 確認	44
5.14. iptables	44
5.15. telnet でリモートログイン	45
5.16. ftp でファイル転送	46
5.17. Armadillo 上でアップロードされたファイルを確認	46
5.18. Armadillo トップページ	47
5.19. mount コマンド書式	51
5.20. ストレージのマウント	51
5.21. ストレージのアンマウント	52
5.22. fdisk コマンドによるパーティション変更	52
5.23. EXT3 ファイルシステムの構築	53
5.24. LED を点灯させる	53
5.25. LED を消灯させる	54

5.26. LED の状態を表示する	54
5.27. LED のトリガに timer を指定する	54
5.28. LED のトリガを表示する	55
5.29. GPIO の入力レベルを取得する	56
5.30. GPIO の出力レベルを設定する	56
5.31. システムクロックを設定	57
5.32. ハードウェアクロックを設定	57
6.1. コンフィグ領域の読み出し方法	58
6.2. コンフィグ領域の保存方法	58
6.3. コンフィグ領域の初期化方法	59

表目次

1.1. 使用しているフォント	9
1.2. 表示プロンプトと実行環境の関係	9
1.3. コマンド入力例での省略表記	10
3.1. ATDE5 の種類	16
3.2. ユーザー名とパスワード	16
3.3. 動作確認に使用する取り外し可能デバイス	17
3.4. シリアル通信設定	18
3.5. Armadillo-810 CPU ボードのインターフェース内容	19
3.6. Armadillo-810 拡張ボード 01 (A コネクタ用)のインターフェース内容	20
3.7. Armadillo-810 カメラモジュール 01 (B コネクタ用)のインターフェース内容	21
3.8. ジャンパの機能	27
3.9. 入力モードに移行するコマンド	29
3.10. カーソルの移動コマンド	29
3.11. 文字の削除コマンド	30
3.12. 保存・終了コマンド	30
4.1. 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	35
5.1. mDNS で設定されるホスト名	41
5.2. 固定 IP アドレス設定例	42
5.3. TELNET でログイン可能なユーザ	44
5.4. ftp でログイン可能なユーザ	45
5.5. シリアルデバイス	47
5.6. ストレージデバイス	51
5.7. LED クラスディレクトリと LED の対応	53
5.8. trigger の種類	54
5.9. Armadillo-810 拡張ボード 01 (A コネクタ用)の CON2 の GPIO ディレクトリ	55
5.10. direction の設定	56
5.11. 時刻フォーマットのフィールド	57

1. はじめに

このたびは Armadillo-810 カメラモデル開発セットをお求めいただき、ありがとうございます。

Armadillo-810 カメラモデルは、「Armadillo-810」と「Armadillo-810 拡張ボード 01 (A コネクタ用)」、「Armadillo-810 カメラモジュール 01 (B コネクタ用)」から構成されています。

Armadillo-810 は、Renesas 社製 Cortex-A9 プロセッサ「R-Mobile A1」、DDR3 SDRAM、フラッシュメモリを中心に、カメラインターフェース、シリアルポート、USB 2.0 デバイスポートなどを搭載し、且つ、拡張用コネクタには USB 2.0 ホストインターフェース、SD/SDIO インターフェース、SPI、GPIO などといった組み込みシステムに求められる機能を備える小型 CPU ボードです。

Armadillo-810 は、インテリジェントカメラのプラットフォームとして利用することを想定して設計されています。カメラから取得した画像を加工・解析を行い特定用途向けのデータに変換して送信することができます。Armadillo-810 カメラモデル開発セットには、シキノハイテック製カメラモジュールが搭載され、且つ必要なソフトウェアが同梱されていますので、ご購入後すぐにシステム開発をスタートすることができます。

Armadillo-800 シリーズは標準 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 言語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。ソフトウェアのカスタマイズ方法については、「Armadillo-810 製品マニュアル」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-810 カメラモデル開発セットがお手元に届きましたら、ハードウェアの動作確認、およ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。

尚、Armadillo-810 には、**ご購入ユーザーに限定して公開しているソフトウェアやハードウェア情報**があります。主な限定コンテンツを次に示します。

- ・ FSE
- ・ AV ミドルウェアコーデック
- ・ 拡張ボードの回路図

限定コンテンツを取得するには、「7. ユーザー登録」を参照してください。

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. 本書および関連ファイルのバージョンについて

本書を含めた関連マニュアル、ソースファイルやイメージファイルなどの関連ファイルは最新版を使用することをおすすめいたします。本書を読み始める前に、Armadillo サイトで最新版の情報をご確認ください。

Armadillo サイト - Armadillo-810 ドキュメント・ダウンロード

<http://armadillo.atmark-techno.com/armadillo-810/downloads>

1.2. 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だら幸いです。

1.3. 本書の構成

本書では、Armadillo-810 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

1.4. 表記について

1.4.1. 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

1.4.2. 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

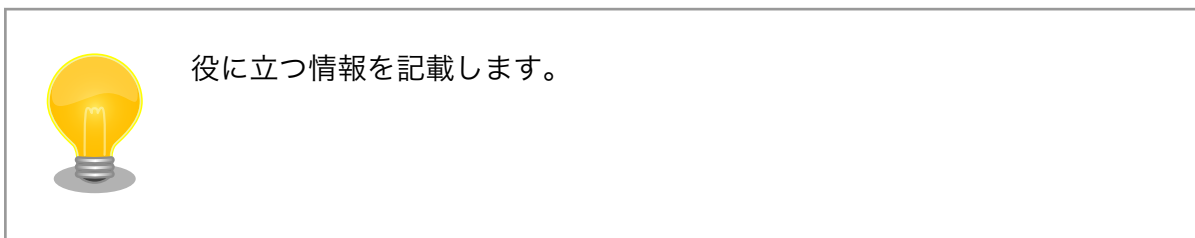
コマンド中で、変更の可能性のあるものや、環境により異なるものに関しては以下のように表記します。適時読み替えて入力してください。

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

1.4.3. 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。



1.5. 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。

2. 注意事項

2.1. 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。



- ・ ご使用の前に必ず製品マニュアルおよび関連資料をお読みにになり、使用上の注意を守って正しく安全にお使いください。
- ・ マニュアルに記載されていない操作・拡張などを行う場合は、弊社 Web サイトに掲載されている資料やその他技術情報を十分に理解した上で、お客様自身の責任で安全にお使いください。
- ・ 水・湿気・ほこり・油煙等の多い場所に設置しないでください。火災、故障、感電などの原因になる場合があります。
- ・ 本製品に搭載されている部品の一部は、発熱により高温になる場合があります。周囲温度や取扱いによってはやけどの原因となる恐れがあります。本体の電源が入っている間、または電源切断後本体の温度が下がるまでの間は、基板上の電子部品、及びその周辺部分には触れないでください。
- ・ 本製品を使用して、お客様の仕様による機器・システムを開発される場合は、製品マニュアルおよび関連資料、弊社 Web サイトで提供している技術情報のほか、関連するデバイスのデータシート等を熟読し、十分に理解した上で設計・開発を行ってください。また、信頼性および安全性を確保・維持するため、事前に十分な試験を実施してください。
- ・ 本製品は、機能・精度において極めて高い信頼性・安全性が必要とされる用途(医療機器、交通関連機器、燃焼制御、安全装置等)での使用を意図しておりません。これらの設備や機器またはシステム等に使用された場合において、人身事故、火災、損害等が発生した場合、当社はいかなる責任も負いかねます。
- ・ 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います。外来ノイズやサージ等により誤作動や故障が発生する可能性があります。万一誤作動または故障などが発生した場合に備え、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・ブレーカー等の保護回路の設置、装置の多重化等)に万全を期し、信頼性および安全性維持のための十分な措置を講じた上でお使いください。
- ・ 無線 LAN 機能を搭載した製品は、心臓ペースメーカーや補聴器などの医療機器、火災報知器や自動ドアなどの自動制御器、電子レンジ、高度な電子機器やテレビ・ラジオに近接する場所、移動体識別用の構

内無線局および特定小電力無線局の近くで使用しないでください。製品が発生する電波によりこれらの機器の誤作動を招く恐れがあります。

2.2. 取扱い上の注意事項

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

破損しやすい箇所	カメラモジュールは、破損しやすい部品になっています。無理に力を加えて破損することのないよう十分注意してください。
本製品の改造	本製品に改造 ^[1] を行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(SD/SDIO, USB)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には、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。
衝撃	落下や衝撃などの強い振動を与えないでください。

2.3. 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	本製品の標準出荷状態でプリインストールされている Linux 対応ソフトウェアは、個別に明示されている（書面、電子データでの通知、口頭での通知を含む）場合を除き、オープンソースとしてソースコードが提供されています。再配布等の権利については、各ソースコードに記載のライセンス形態にしたがって、お客様の責任において行使してください。また、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む）は、現状有姿 (AS IS) にて提供します。お客様ご自身の責任において、使用用途・目的の適合について事前に十分な検討と試験を実施した上でお使いください。アットマークテクノは、当該ソフトウェアが特定の目的に適合すること、ソフトウェアの信頼性および正確性、ソフトウェアを含む本製品の使用による結果について、お客様に対し何らの保証も行いません。
--------------------	--

パートナー等の協力により Armadillo ブランド製品向けに提供されているミドルウェア、その他各種ソフトウェアソリューションは、ソフトウェア毎にライセンスが規定されています。再頒布権等については、各ソフトウェアに付属する readme ファイル等をご参照ください。その他のバンドルソフトウェアについては、各提供元にお問い合わせください。

^[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。

2.4. 書込み禁止領域について



EEPROM のデータは、本製品に含まれるソフトウェアで使用しています。正常に動作しなくなる可能性があるため、書込みを行わないでください。また、意図的に書込みを行った場合は保証対象外となります。

2.5. 電波障害について



Armadillo-810 は、クラス A 情報技術装置(申請中)です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。VCCI-A

2.6. 保証について

本製品の本体基板は、製品に添付もしくは弊社 Web サイトに記載している「製品保証規定」に従い、ご購入から 1 年間の交換保証を行っています。添付品およびソフトウェアは保証対象外となりますのでご注意ください。

製品保証規定 <http://www.atmark-techno.com/support/warranty-policy>

2.7. 輸出について

本製品の開発・製造は、原則として日本国内での使用を想定して実施しています。本製品を輸出する際は、輸出者の責任において、輸出関連法令等を遵守し、必要な手続きを行ってください。海外の法令および規則への適合については当社はなんらの保証を行うものではありません。本製品および関連技術は、大量破壊兵器の開発目的、軍事利用その他軍事用途の目的、その他国内外の法令および規則により製造・使用・販売・調達が禁止されている機器には使用することができません。

2.8. 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。™、®マークは省略しています。
- ・ SD、SDHC、microSD、microSDHC、SDIO ロゴは SD-3C、LLC の商標です。



3. 作業の前に

3.1. 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。

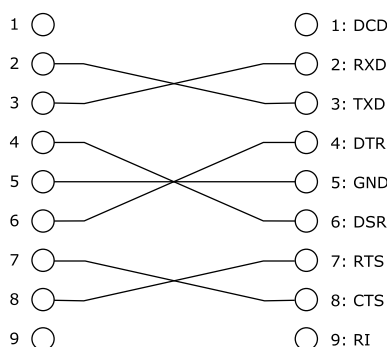
- 作業用 PC Linux または Windows が動作し、2 ポート以上^[1]の USB インターフェースを持つ PC です。作業用 PC には、「3.2. 開発/動作確認環境の構築」を参照して開発/動作確認環境を構築してください。
- SD カード SD スロットの動作を確認する場合などに利用します。
- USB メモリ USB ホストの動作を確認する場合などに利用します。
- シリアルクロスケーブル D-Sub9 ピン(メス - メス)のクロス接続用ケーブルです。シリアルポートの動作を確認する場合に利用します。



シリアルクロスケーブルの結線

シリアルクロスケーブルには様々な結線のものが存在します。本書ではハードウェアフロー(RTS/CTS)を使用しないため、互いのコネクタの TxD と RxD 同士および GND 同士が結線されているケーブルであれば利用可能です。

ハードウェアフローを使用する場合は、以下のようなインタリンク結線のシリアルケーブルをご利用ください。



3.2. 開発/動作確認環境の構築

アットマークテクノ製品のソフトウェア開発や動作確認を簡単に行うために、VMware 仮想マシンのデータイメージを提供しています。この VMware 仮想マシンのデータイメージを ATDE (Atmark Techno Development Environment) と呼びます。ATDE の起動には仮想化ソフトウェアである VMWare を使用します。

^[1]USB HUB を利用することもできます。



仮想化ソフトウェアとして、VMware の他に Oracle VM VirtualBox が有名です。Oracle VM VirtualBox には以下の特徴があります。

- ・ GPL v2(General Public License version 2)で提供されている^[2]
- ・ VMware 形式の仮想ディスク(.vmdk)ファイルに対応している

Oracle VM VirtualBox から ATDE を起動し、ソフトウェア開発環境として使用することができます。ただし、UVC ガジェットの動作確認など、本書に記載されている内容のうち一部について適用できない場合があります。十分ご注意くださいの上で作業してください。

ATDE は、バージョンにより対応するアットマークテクノ製品が異なります。Armadillo-810 に対応している ATDE は、ATDE5 (ATDE バージョン 5)です。

ATDE5 は Debian GNU/Linux 7.0(コードネーム wheezy)をベースに、Armadillo-810 のソフトウェア開発を行うために必要なクロス開発ツールや、Armadillo-810 の動作確認を行うために必要なツールが事前にインストールされています。

3.2.1. ATDE5 セットアップ

3.2.1.1. VMware のインストール

ATDE5 を使用するためには、作業用 PC に VMware がインストールされている必要があります。VMware 社 Web ページ(<http://www.vmware.com/>)を参照し、利用目的に合う VMware 製品をインストールしてください。



VMware は、非商用利用限定で無償のものから、商用利用可能な有償のものまで複数の製品があります。製品ごとに異なるライセンス、エンドユーザー使用許諾契約書(EULA)が存在するため、十分に確認した上で利用目的に合う製品をご利用ください。



VMware や ATDE5 が動作しないことを未然に防ぐため、使用する VMware のドキュメントから以下の項目についてご確認ください。

- ・ ホストシステムのハードウェア要件
- ・ ホストシステムのソフトウェア要件
- ・ ゲスト OS のプロセッサ要件

VMware のドキュメントは、VMware 社 Web ページ (<http://www.vmware.com/>)から取得することができます。

^[2]バージョン 3.x までは PUEL(VirtualBox Personal Use and Evaluation Licence)が適用されている場合があります。

3.2.1.2. ATDE5 の取得

「表 3.1. ATDE5 の種類」に示す ATDE5 のアーカイブのうちいずれか 1 つを作業用 PC にコピーします。ATDE5 のアーカイブは Armadillo サイト(<http://armadillo.atmark-techno.com>)または、開発セット付属の DVD から取得可能です。

表 3.1 ATDE5 の種類

ATDE5 アーカイブ	ベースの Debian GNU/Linux
atde5-[version]-amd64.zip	64-bit PC(「amd64」)アーキテクチャ用 Debian GNU/Linux 7.0
atde5-[version]-i386.zip	32-bit PC(「i386」)アーキテクチャ用 Debian GNU/Linux 7.0



作業用 PC の動作環境(ハードウェア、VMware、ATDE5 の種類など)により、ATDE5 が正常に動作しない可能性があります。VMware 社 Web ページ(<http://www.vmware.com/>)から、使用している VMware のドキュメントなどを参照して動作環境を確認してください。

3.2.1.3. ATDE5 の起動

ATDE5 のアーカイブを展開したディレクトリに存在する仮想マシン構成(.vmx)ファイルを VMware 上で開くと、ATDE5 を起動することができます。ATDE5 にログイン可能なユーザーを、「表 3.2. ユーザー名とパスワード」に示します^[3]。

表 3.2 ユーザー名とパスワード

ユーザー名	パスワード	権限
atmark	atmark	一般ユーザー
root	root	特権ユーザー



ATDE に割り当てるメモリおよびプロセッサ数を増やすことで、ATDE をより快適に使用することができます。仮想マシンのハードウェア設定の変更方法については、VMware 社 Web ページ (<http://www.vmware.com/>)から、使用している VMware のドキュメントなどを参照してください。

3.2.2. 取り外し可能デバイスの使用

VMware は、ゲスト OS (ATDE)による取り外し可能デバイス(USB デバイスや DVD など)の使用をサポートしています。デバイスによっては、ホスト OS (VMware を起動している OS)とゲスト OS で同時に使用することができません。そのようなデバイスをゲスト OS で使用するためには、ゲスト OS にデバイスを接続する操作が必要になります。



取り外し可能デバイスの使用方法については、VMware 社 Web ページ (<http://www.vmware.com/>)から、使用している VMware のドキュメントなどを参照してください。

^[3]特権ユーザーで GUI ログインを行うことはできません。

Armadillo-810 の動作確認をおこなうためには、「表 3.3. 動作確認に使用する取り外し可能デバイス」に示すデバイスをゲスト OS に接続する必要があります。

表 3.3 動作確認に使用する取り外し可能デバイス

デバイス	デバイス名
開発用 USB シリアル変換アダプタ(Armadillo-800 シリーズ対応)	Future Devices FT232R USB UART
Armadillo-810 USB インターフェース(CON4)	g_uvc_acm_ether
作業用 PC の物理シリアルポート	シリアルポート

3.2.3. コマンドライン端末(GNOME 端末)の起動

ATDE5 で、CUI (Character-based User Interface)環境を提供するコマンドライン端末を起動します。ATDE5 で実行する各種コマンドはコマンドライン端末に入力し、実行します。コマンドライン端末にはいくつかの種類がありますが、ここでは GNOME デスクトップ環境に標準インストールされている GNOME 端末を起動します。

GNOME 端末を起動するには、「図 3.1. GNOME 端末の起動」のようにデスクトップ左上のメニューから「端末」を選択してください。



図 3.1 GNOME 端末の起動

「図 3.2. GNOME 端末のウィンドウ」のようにウィンドウが開きます。



図 3.2 GNOME 端末のウィンドウ

3.2.4. シリアル通信ソフトウェア(minicom)の使用

シリアル通信ソフトウェア(minicom)のシリアル通信設定を、「表 3.4. シリアル通信設定」のように設定します。また、minicom の横幅を 80 文字以上にしてください。横幅が 80 文字より小さい場合、コマンド入力中に表示が乱れることがあります。

表 3.4 シリアル通信設定

項目	設定
転送レート	115,200bps
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

minicom の設定を開始するには、「図 3.3. minicom 設定方法」のようにしてください。設定完了後、デフォルト設定(df1)に保存して終了します。

```
[ATDE ~]$ LANG=C minicom --setup
```

図 3.3 minicom 設定方法

minicom を起動させるには、「図 3.4. minicom 起動方法」のようにしてください。

```
[ATDE ~]$ LANG=C minicom --noinit --wrap --device /dev/ttyUSB0
```

図 3.4 minicom 起動方法

minicom を終了させるには、まず Ctrl+a に続いて q キーを入力します。その後、以下のように表示されたら「Yes」にカーソルを合わせて Enter キーを入力すると minicom が終了します。

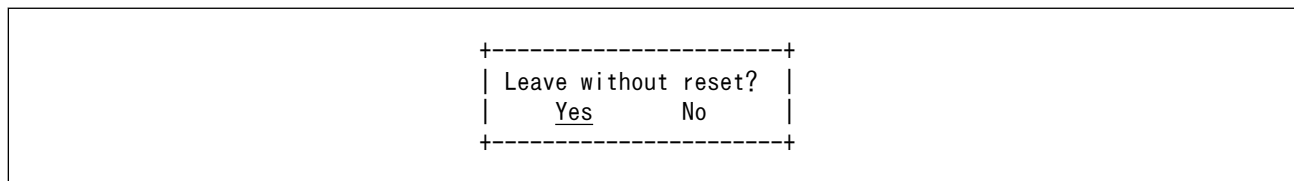



図 3.5 minicom 終了確認



Ctrl+a に続いて z キーを入力すると、minicom のコマンドヘルプが表示されます。

3.3. インターフェースレイアウト

Armadillo-810 及びカメラモデル開発セットに含まれる各基板のインターフェースレイアウトです。各インターフェースの配置場所等を確認してください。

3.3.1. Armadillo-810 CPU ボード インターフェースレイアウト

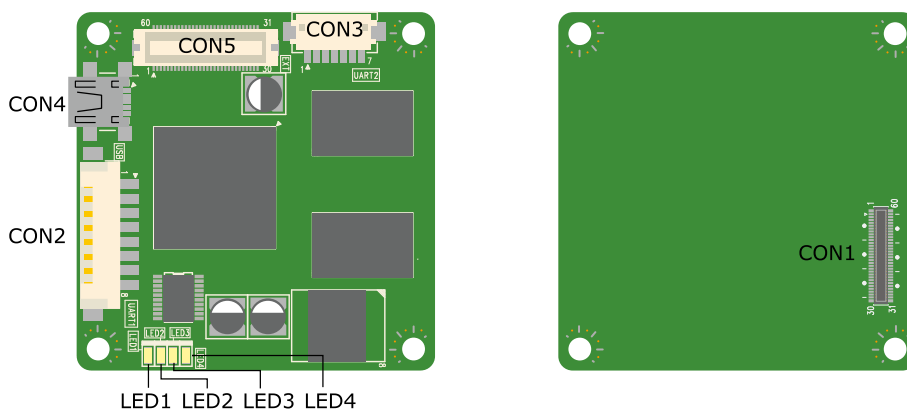


図 3.6 Armadillo-810 CPU ボードのインターフェースレイアウト図

表 3.5 Armadillo-810 CPU ボードのインターフェース内容

部品番号	インターフェース	形状	備考
CON1	拡張インターフェース 2 (B コネクタ)	BtoB コネクタ 60P(0.4mm ピッチ) DF40C-60DP-0.4V(51)/HIROSE	対向コネクタ例: DF40HC(4.0)-60DS-0.4V(51)/HIROSE 挿抜寿命:30 回
CON2	シリアルインターフェース 1	ピンヘッダ 8P(2mm ピッチ) DF3DZ-8P-2H(51)/HIROSE	信号レベル: RS232C 対向コネクタ例: DF3-8S-2C/HIROSE 挿抜寿命:50 回
CON3	シリアルインターフェース 2	ピンヘッダ 7P(1.25mm ピッチ) DF13A-7P-1.25H(51)/HIROSE	信号レベル: 3.3V CMOS 対向コネクタ例: DF13-7S-1.25C/HIROSE 挿抜寿命:50 回
CON4	USB インターフェース	USB mini B コネクタ	USB2.0 Device(High Speed 対応)

部品番号	インターフェース	形状	備考
CON5	拡張インターフェース 1 (A コネクタ)	BtoB コネクタ 60P(0.5mm ピッチ) DF17(4.0)-60DS-0.5V(57)/HIROSE	対向コネクタ例: DF17(4.0)-60DP-0.5V(57)/HIROSE 挿抜寿命:50 回
LED1~LED4	ユーザー LED	LED(黄色、面実装)	

3.3.2. Armadillo-810 拡張ボード 01 (A コネクタ用) インターフェースレイアウト

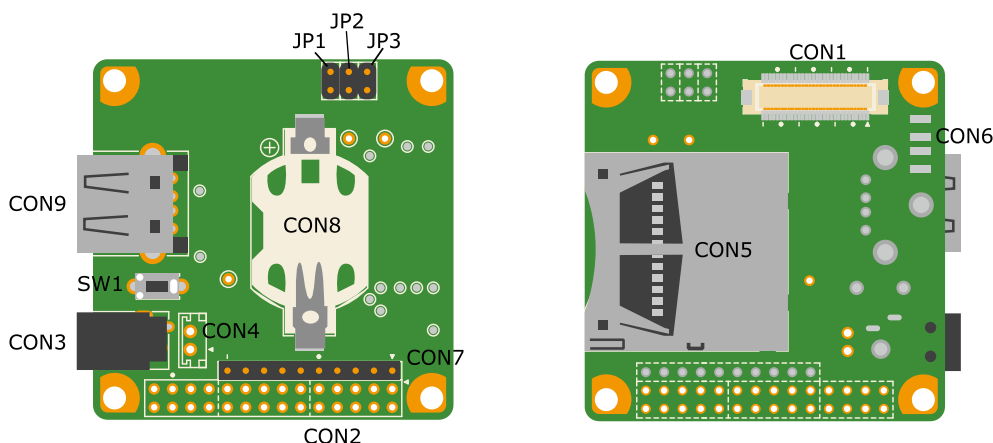


図 3.7 Armadillo-810 拡張ボード 01 (A コネクタ用)のインターフェースレイアウト図

表 3.6 Armadillo-810 拡張ボード 01 (A コネクタ用)のインターフェース内容

部品番号	インターフェース	形状	備考
CON1	Armadillo-810 接続インターフェース	BtoB コネクタ 60P(0.5mm ピッチ) DF17(4.0)-60DP-0.5V(57)/HIROSE	対向コネクタ例:DF17(4.0)-60DS-0.5V(57)/HIROSE 挿抜寿命:50 回 Armadillo-810 CPU ボードの拡張インターフェース 1 (A コネクタ)と接続
CON2	拡張インターフェース	ピンヘッダ 28P(2.54mm ピッチ)	コネクタ非搭載(搭載コネクタ例: A1-28PA-2.54DSA(71)/HIROSE)
CON3	電源入力 1	DC ジャック	対応プラグ: EIAJ#2
CON4	電源入力 2	ピンヘッダ 2P(2.5mm ピッチ)	コネクタ非搭載(搭載コネクタ例: B2B-EH/JST)
CON5	SD インターフェース	SD スロット SCDA9A0400/ALPS	信号線は CON7 と共通
CON6	Reserved	Pad	このインターフェースを使用する場合の動作は保証されていません
CON7	JTAG インターフェース	ピンヘッダ(10P) (2.54mm ピッチ)	信号線は CON5 と共通
CON8	RTC 外部バックアップインターフェース	電池ボックス	対応電池: CR2032
CON9	USB インターフェース	USB Type A コネクタ	USB2.0 Host(High Speed 対応)
JP1	起動モード設定ジャンパ	ピンヘッダ(2P) (2.54mm ピッチ)	オープン: OS 自動起動モード ショート: 保守モード
JP2	SD/JTAG 設定ジャンパ	ピンヘッダ(2P) (2.54mm ピッチ)	オープン: SD(CON5)有効/JTAG(CON7)無効 ショート: SD(CON5)無効/JTAG(CON7)有効
JP3	起動デバイス設定ジャンパ	ピンヘッダ(2P) (2.54mm ピッチ)	オープン: オンボードフラッシュメモリブート ショート: SD(CON5)ブート
SW1	リセットスイッチ	タクトスイッチ SKHLACA010/ALPS	

3.3.3. Armadillo-810 カメラモジュール 01 (B コネクタ用) インターフェースレイアウト

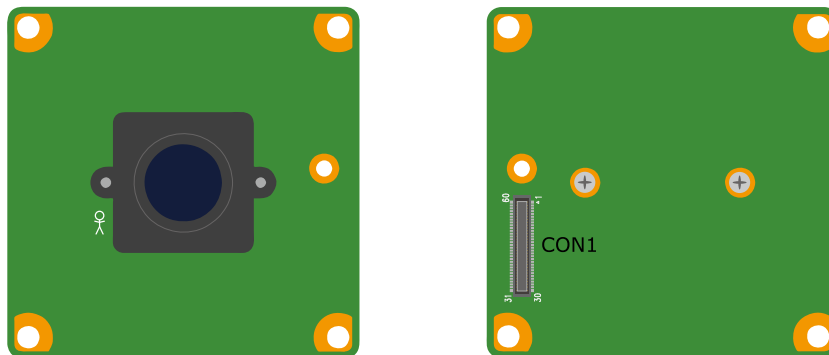


図 3.8 Armadillo-810 カメラモジュール 01 (B コネクタ用)のインターフェースレイアウト図

表 3.7 Armadillo-810 カメラモジュール 01 (B コネクタ用)のインターフェース内容

部品番号	インターフェース	形状	備考
CON1	Armadillo-810 接続 インターフェース	ピンソケット 60P(0.4mm ピッチ) DF40HC(4.0)-60DS-0.4V(51)/ HIROSE	対向コネクタ例: DF40C-60DP-0.4V(51)/ HIROSE 挿抜寿命:30 回 Armadillo-810 CPU ボードの拡張インター フェース 2 (B コネクタ)と接続

3.4. 組み立て

3.4.1. Armadillo-810 カメラモデルの組み立て

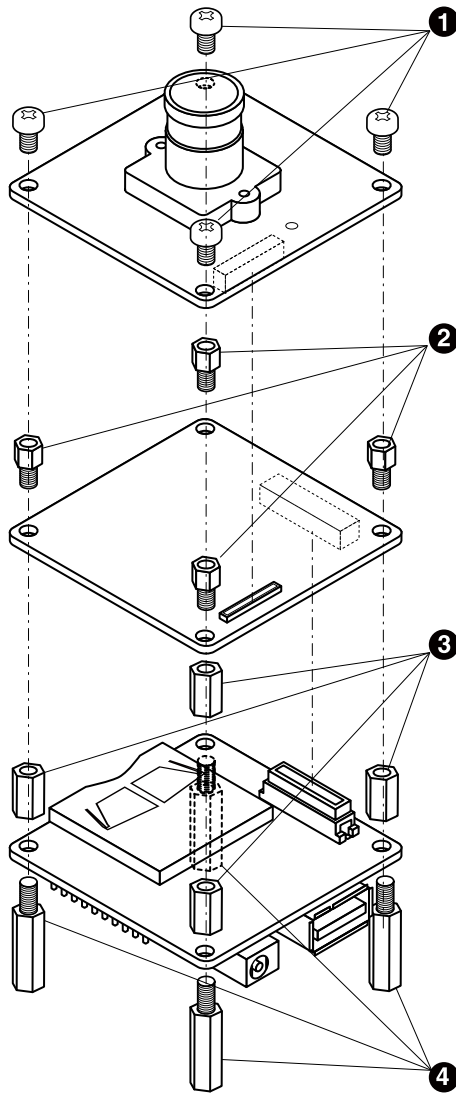


図 3.9 Armadillo-810 カメラモデルの組み立て

- ① なべ小ねじ(M3、L=4mm)
- ② 金属スペーサ(M3、L=4mm)
- ③ 金属スペーサ(M3、L=8mm)
- ④ 金属スペーサ(M3、L=15mm)



コネクタ嵌合時の取扱い上の注意

嵌合する際は、コネクタの中心をきちり合わせてください。

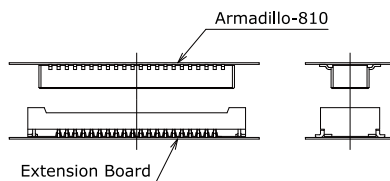


図 3.10 コネクタ嵌合時の取扱い上の注意 1

位置合わせをする際は、無理な力を加えることなく誘い込み口を探してください。無理な力を加えると、モールドの破損、削れが発生し、接触抵抗の不具合等に繋がる場合があります。

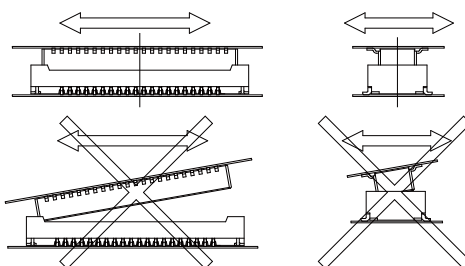


図 3.11 コネクタ嵌合時の取扱い上の注意 2

コネクタが誘い込まれると、コネクタ間の距離が近くなり、平行になって前後左右に動かなくなります。この状態からまっすぐに嵌合してください。

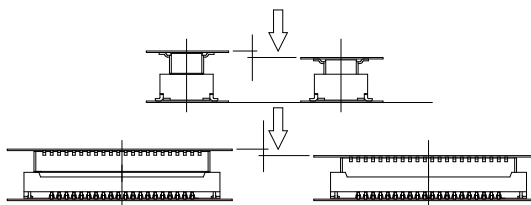


図 3.12 コネクタ嵌合時の取扱い上の注意 3



コネクタ抜去時の取扱い上の注意

コネクタは平行に抜去してください。

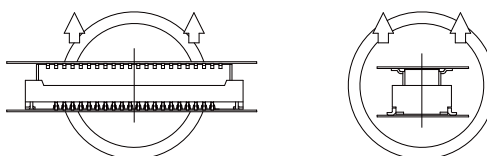


図 3.13 コネクタ抜去時の取扱い上の注意 1

平行に抜去することが困難な場合、コネクタ幅の狭い方向から斜めに抜去してください。

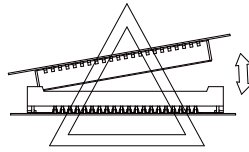


図 3.14 コネクタ抜去時の取扱い上の注意 2

コネクタが損傷する可能性が高いため、コネクタのコーナー方向や幅の広い方向から斜めに抜去しないでください。

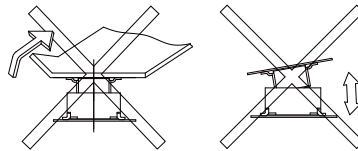


図 3.15 コネクタ抜去時の取扱い上の注意 3



ねじ締め時の注意事項

ねじは図のようにコネクタから遠い箇所から順に締めてください。先にコネクタに近い箇所のねじを締めると、コネクタに無理な力が加わり、コネクタ破損の原因となることがあります。

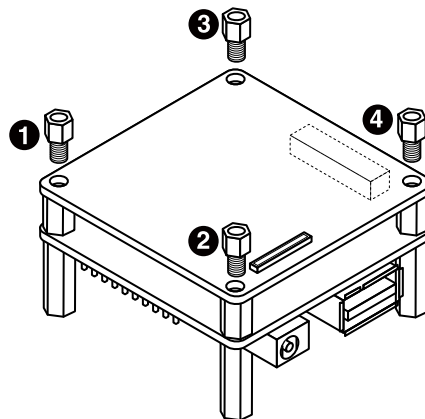


図 3.16 ねじ締め時の注意事項 1

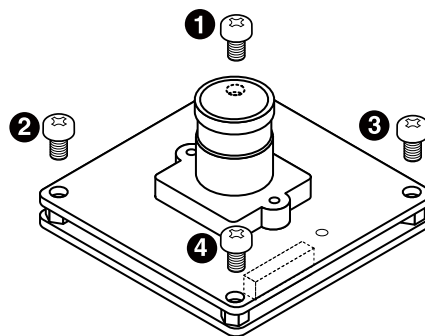


図 3.17 ねじ締め時の注意事項 2

3.4.2. レンズの交換方法

Armadillo-810 カメラモジュール 01 (B コネクタ用)には出荷時にカメラモジュール用レンズ(水平画角 79°)が取り付けられています。付属のカメラモジュール用レンズ(水平画角 120°)をご使用になりたい場合は、カメラレンズを交換して使用する事が可能です。

交換時には、カメラモジュールを下向きにし、レンズを矢印の方向に回してホルダから外し、交換したいレンズを装着して下さい。

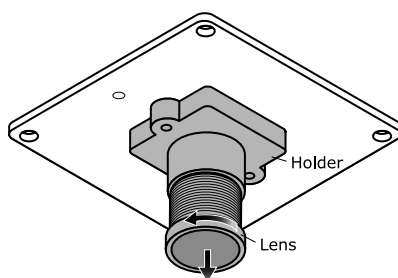


図 3.18 カメラレンズの交換



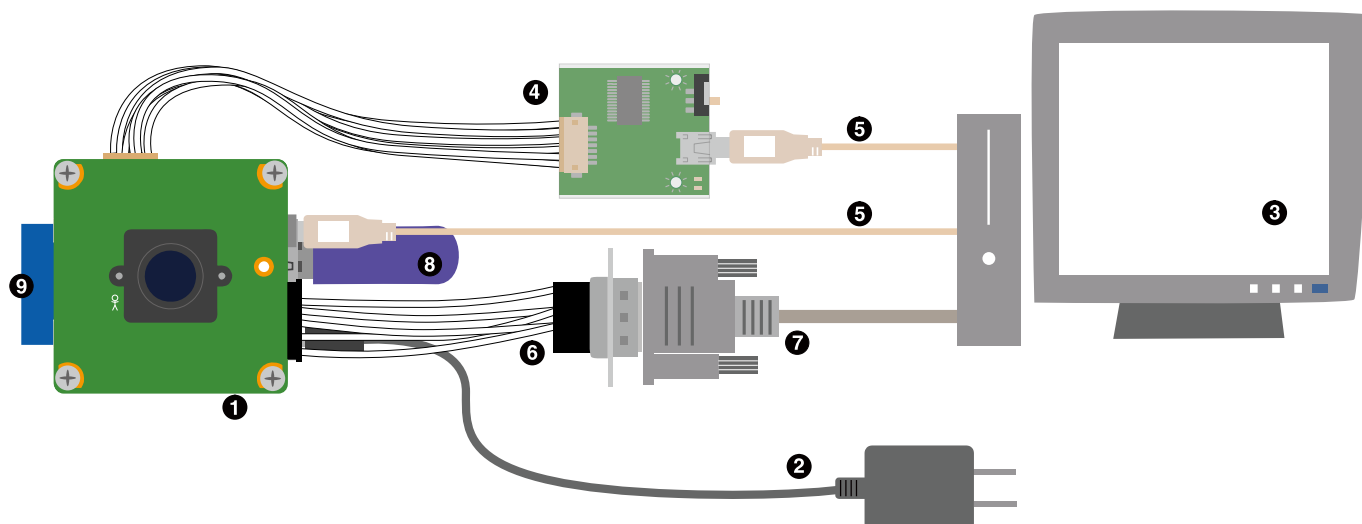
カメラレンズ交換時の注意事項

下記の項目に十分留意して作業を行ってください。下記の項目を守らず作業を行った場合、撮像素子やレンズが破損、またはゴミ等が付着し、取得画像にノイズとなって残る可能性があります。

- ・ レンズに直接触らないで下さい。
- ・ レンズホルダはカメラモジュールから外さないで下さい。
- ・ ピント調整の際はカメラの映像を確認しながら行ってください。
- ・ レンズ交換の際はゴミや埃が撮像素子に付着しないよう注意してください。
- ・ 埃が立たない場所で作業を行って下さい。
- ・ レンズ交換は埃やゴミが侵入しないよう短時間で作業を完了させて下さい。
- ・ 基板からのレンズ先端の高さは 18mm 以下にしないでください。18mm 以下までレンズを入れた場合、レンズがイメージセンサーに接触し、レンズおよびイメージセンサーが故障する恐れがあります。
- ・ レンズ交換作業による破損または故障につきましては、保証対象外となります。

3.5. 接続方法

Armadillo-810 と作業用 PC との接続例を「図 3.19. 接続図」に示します。



- ① Armadillo-810 カメラモデル
- ② AC アダプタ(5V/2.0A EIAJ#2)^[4]
- ③ 作業用 PC
- ④ 開発用 USB シリアル変換アダプタ(Armadillo-800 シリーズ対応)^[4]
- ⑤ USB2.0 ケーブル(A-miniB タイプ, 1.8m)^[4]
- ⑥ 7 ピン シリアルケーブル(Armadillo-800 シリーズ対応)^[4]
- ⑦ シリアルクロスケーブル
- ⑧ USB メモリ
- ⑨ SD カード

図 3.19 接続図



開発用 USB シリアル変換アダプタ(Armadillo-800 シリーズ対応)の取扱い上の注意

USB シリアル変換アダプタには電源投入順序があります。Armadillo-810 に接続する際は、以下の手順に従ってご使用ください。接続手順に従わない場合は、USB シリアル変換アダプタが故障する可能性がありますのでご注意ください。

1. 起動中の作業用 PC と USB シリアル変換アダプタを USB2.0 ケーブルで接続します。
2. Armadillo-810 のシリアルインターフェース 2 (CON3)に USB シリアル変換アダプタを接続します。
3. 上記接続を確認後、Armadillo-810 に電源を投入します。

^[4]Armadillo-810 カメラモデル開発セット付属品

また、Armadillo-810 に USB シリアル変換アダプタを接続した状態のまま、作業用 PC または USB シリアル変換アダプタから USB2.0 ケーブルを抜く場合や作業用 PC をシャットダウンする場合は、Armadillo-810 の電源が切断されていることを確認してから行ってください。

3.6. ジャンパピンの設定について

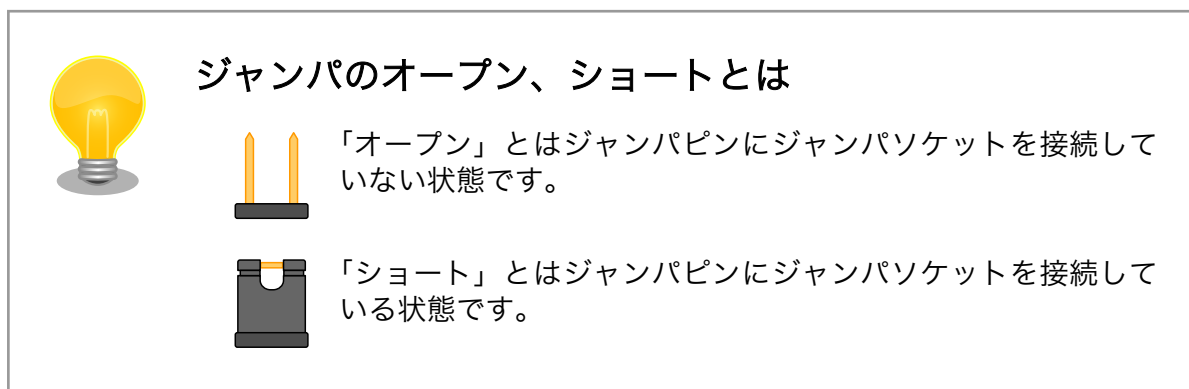
ジャンパの設定を変更することで、Armadillo-810 の動作を変更することができます。ジャンパの機能を「表 3.8. ジャンパの機能」に示します。

表 3.8 ジャンパの機能

ジャンパ	機能	動作
JP1	起動モード設定	オープン: OS を自動起動します。 ショート: ブートローダーを保守モードにします。
JP2	SD/JTAG 設定	オープン: SD インターフェースを有効化します。JTAG インターフェースは無効化されます。 ショート: JTAG インターフェースを有効化します。SD インターフェースは無効化されます。
JP3	起動デバイス設定	オープン: オンボードフラッシュメモリのブートローダーを起動します。 ショート: SD カードのブートローダーを起動します。

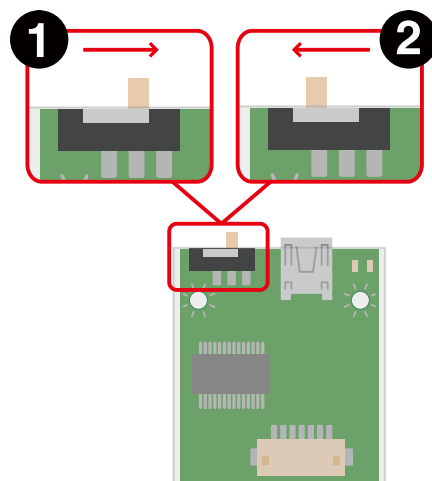
各ジャンパは必要に応じて切り替えの指示があります。ここでは、全てのジャンパをオープンに設定しておきます。

ジャンパピンの位置は「図 3.7. Armadillo-810 拡張ボード 01 (A コネクタ用)のインターフェースレイアウト図」で確認することができます。



3.7. スライドスイッチの設定について

開発用 USB シリアル変換アダプタ (Armadillo-800 シリーズ対応) のスライドスイッチには、Armadillo-810 拡張ボード 01 (A コネクタ用) の JP1 と同じ機能が割り当てられています。



- ① OS 自動起動モード
- ② 保守モード

図 3.20 スライドスイッチの設定

Armadillo-810 に Armadillo-810 拡張ボード 01 (A コネクタ用) を接続している場合は、常に「OS 自動起動モード」に設定してください。

3.8. vi エディタの使用方法

vi エディタは、Armadillo に標準でインストールされているテキストエディタです。本書では、Armadillo の設定ファイルの編集などに vi エディタを使用します。

vi エディタは、ATDE にインストールされてる gedit や emacs などのテキストエディタとは異なり、モードを持っていることが大きな特徴です。vi のモードには、コマンドモードと入力モードがあります。コマンドモードの時に入力した文字はすべてコマンドとして扱われます。入力モードでは文字の入力ができます。

本章で示すコマンド例は ATDE で実行するよう記載していますが、Armadillo でも同じように実行することができます。

3.8.1. vi の起動

vi を起動するには、以下のコマンドを入力します。

```
[ATDE ~]# vi [file]
```

図 3.21 vi の起動

file にファイル名のパスを指定すると、ファイルの編集 (*file* が存在しない場合は新規作成) を行いません。vi はコマンドモードの状態です。


3.8.2. 文字の入力

文字を入力するにはコマンドモードから入力モードへ移行する必要があります。コマンドモードから入力モードに移行するには、「表 3.9. 入力モードに移行するコマンド」に示すコマンドを入力します。入力モードへ移行後は、キーを入力すればそのまま文字が入力されます。

表 3.9 入力モードに移行するコマンド

コマンド	動作
i	カーソルのある場所から文字入力を開始
a	カーソルの後ろから文字入力を開始

入力モードからコマンドモードに戻りたい場合は、ESC キーを入力することで戻ることができます。現在のモードがわからなくなった場合は、ESC キーを入力し、一旦コマンドモードへ戻ることにより混乱を防げます。



日本語変換機能を OFF に

vi のコマンドを入力する時は ATDE の日本語入力システム(Mozc)を OFF にしてください。日本語入力システムの ON/OFF は、半角/全角キーまたは、Shift+Space キーで行うことができます。

「i」、「a」それぞれのコマンドを入力した場合の文字入力の開始位置を「図 3.22. 入力モードに移行するコマンドの説明」に示します。

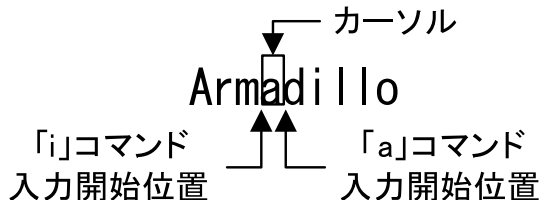



図 3.22 入力モードに移行するコマンドの説明



vi での文字削除

コンソールの環境によっては BS(Backspace)キーで文字が削除できず、「^H」文字が入力される場合があります。その場合は、「3.8.4. 文字の削除」で説明するコマンドを使用し、文字を削除してください。

3.8.3. カーソルの移動

方向キーでカーソルの移動ができますが、コマンドモードで「表 3.10. カーソルの移動コマンド」に示すコマンドを入力することでもカーソルを移動することができます。

表 3.10 カーソルの移動コマンド

コマンド	動作
h	左に 1 文字移動

コマンド	動作
j	下に1文字移動
k	上に1文字移動
l	右に1文字移動

3.8.4. 文字の削除

文字を削除する場合は、コマンドモードで「表 3.11. 文字の削除コマンド」に示すコマンドを入力します。

表 3.11 文字の削除コマンド

コマンド	動作
x	カーソル上の文字を削除
dd	現在行を削除

「x」コマンド、「dd」コマンドを入力した場合に削除される文字を「図 3.23. 文字を削除するコマンドの説明」に示します。

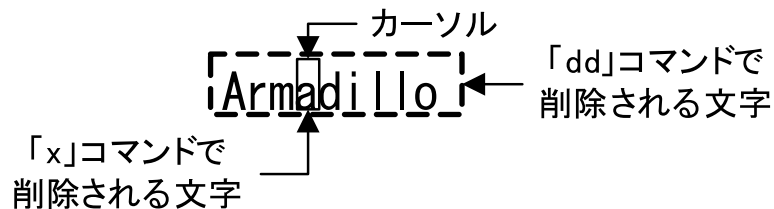


図 3.23 文字を削除するコマンドの説明

3.8.5. 保存と終了

ファイルの保存、終了を行うコマンドを「表 3.12. 保存・終了コマンド」に示します。

表 3.12 保存・終了コマンド

コマンド	動作
:q!	変更を保存せずに終了
:w [file]	ファイル名を file に指定して保存
:wq	ファイルを上書き保存して終了

保存と終了を行うコマンドは「:」(コロン)からはじまるコマンドを使用します。"."キーを入力すると画面下部にカーソルが移り入力したコマンドが表示されます。コマンドを入力した後 Enter キーを押すことで、コマンドが実行されます。

4. 起動と終了

4.1. 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 v3.2.0 (Armadillo-810/nor) compiled at 15:48:09, Feb 06 2013
Uncompressing kernel.....
.....done.
Uncompressing ramdisk.....
.....done.
Booting Linux on physical CPU 0
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 3.4-at1 (atmark@atde5) (gcc version 4.6.3 (Debian 4.6.3-11) ) #1 P
REEMPT Thu Feb 7 07:23:13 JST 2013
CPU: ARMv7 Processor [412fc093] revision 3 (ARMv7), cr=10c53c7d
CPU: PIPT / VIPT nonaliasing data cache, VIPT aliasing instruction cache
Machine: armadillo810
Memory policy: ECC disabled, Data cache writeback
bootconsole [early_ttySC2] enabled
Built 1 zonelists in Zone order, mobility grouping on. Total pages: 130048
Kernel command line: console=ttySC2,115200 earlyprintk=sh-sci.2,115200
PID hash table entries: 2048 (order: 1, 8192 bytes)
Dentry cache hash table entries: 65536 (order: 6, 262144 bytes)
Inode-cache hash table entries: 32768 (order: 5, 131072 bytes)
allocated 1048576 bytes of page_cgroup
please try 'cgroup_disable=memory' option if you don't want memory cgroups
Memory: 512MB = 512MB total
Memory: 489880k/489880k available, 34408k reserved, 0K highmem
Virtual kernel memory layout:
   vector   : 0xffff0000 - 0xffff1000   (  4 kB)
   fixmap   : 0xffff0000 - 0xfffe0000   ( 896 kB)
   vmalloc  : 0xe0800000 - 0xff000000   ( 488 MB)
   lowmem   : 0xc0000000 - 0xe0000000   ( 512 MB)
   pkmap    : 0xbfe00000 - 0xc0000000   (  2 MB)
   modules  : 0xbf000000 - 0xbfe00000   ( 14 MB)
     .text   : 0xc0008000 - 0xc04a5000   (4724 kB)
     .init   : 0xc04a5000 - 0xc04ca000   ( 148 kB)
     .data   : 0xc04ca000 - 0xc04ff8a0   ( 215 kB)
     .bss    : 0xc04ff8c4 - 0xc052cc74   ( 181 kB)
NR_IRQS:16 nr_irqs:16 16
intc: Registered controller 'r8a7740-intca' with 117 IRQs
```

```
intc: Registered controller 'r8a7740-intca-irq-pins' with 32 IRQs
intc: Registered controller 'r8a7740-intcs' with 33 IRQs
sched_clock: 32 bits at 128 Hz, resolution 7812500ns, wraps every 3489660920ms
Console: colour dummy device 80x30
 sh_cmt_simple.10: used as clock source
 sh_cmt_simple.14: used for clock events
 sh_cmt_simple.14: used for periodic clock events
Calibrating delay loop... 1576.53 BogoMIPS (lpj=6156288)
pid_max: default: 32768 minimum: 301
Mount-cache hash table entries: 512
Initializing cgroup subsys cpuacct
Initializing cgroup subsys memory
Initializing cgroup subsys devices
Initializing cgroup subsys freezer
Initializing cgroup subsys blkio
CPU: Testing write buffer coherency: ok
hw perfevents: enabled with ARMv7 Cortex-A9 PMU driver, 7 counters available
Setting up static identity map for 0x40391eb0 - 0x40391ee4
dummy:
NET: Registered protocol family 16
pfc: r8a7740_pfc handling gpio 0 -> 858
gpiochip_add: registered GPIOs 0 to 858 on device: r8a7740_pfc
CON5: STANDARD extension board found.
L310 cache controller enabled
l2x0: 8 ways, CACHE_ID 0x410000c7, AUX_CTRL 0x42440000, Cache size: 262144 B
hw-breakpoint: found 5 (+1 reserved) breakpoint and 1 watchpoint registers.
hw-breakpoint: maximum watchpoint size is 4 bytes.
bio: create slab <bio-0> at 0
sdhi0: 3300 mV
SCSI subsystem initialized
usbcore: registered new interface driver usbfs
usbcore: registered new interface driver usb
usbcore: registered new device driver usb
i2c-gpio i2c-gpio.2: using pins 99 (SDA) and 98 (SCL)
i2c-sh_mobile i2c-sh_mobile.0: Runtime PM disabled, clock forced on.
i2c-sh_mobile i2c-sh_mobile.0: I2C adapter 0 with bus speed 100000 Hz
i2c-sh_mobile i2c-sh_mobile.1: Runtime PM disabled, clock forced on.
i2c-sh_mobile i2c-sh_mobile.1: I2C adapter 1 with bus speed 100000 Hz
Linux video capture interface: v2.00
Advanced Linux Sound Architecture Driver Version 1.0.25.
Switching to clocksource sh_cmt_simple.10
 sh_cmt_simple.14: used for oneshot clock events
NET: Registered protocol family 2
IP route cache hash table entries: 4096 (order: 2, 16384 bytes)
TCP established hash table entries: 16384 (order: 5, 131072 bytes)
TCP bind hash table entries: 16384 (order: 4, 65536 bytes)
TCP: Hash tables configured (established 16384 bind 16384)
TCP: reno registered
UDP hash table entries: 256 (order: 0, 4096 bytes)
UDP-Lite hash table entries: 256 (order: 0, 4096 bytes)
NET: Registered protocol family 1
RPC: Registered named UNIX socket transport module.
RPC: Registered udp transport module.
RPC: Registered tcp transport module.
RPC: Registered tcp NFSv4.1 backchannel transport module.
Trying to unpack rootfs image as initramfs...
rootfs image is not initramfs (junk in compressed archive); looks like an initrd
Freeing initrd memory: 23536K
```



```
audit: initializing netlink socket (disabled)
type=2000 audit(0.500:1): initialized
VFS: Disk quotas dquot_6.5.2
Dquot-cache hash table entries: 1024 (order 0, 4096 bytes)
squashfs: version 4.0 (2009/01/31) Phillip Lougher
NFS: Registering the id_resolver key type
nfs4filelayout_init: NFSv4 File Layout Driver Registering...
msgmni has been set to 1002
Block layer SCSI generic (bsg) driver version 0.4 loaded (major 253)
io scheduler noop registered
io scheduler deadline registered
io scheduler cfq registered (default)
sh-dma-engine sh-dma-engine.0: Runtime PM disabled, clock forced on.
sh-dma-engine sh-dma-engine.1: Runtime PM disabled, clock forced on.
sh-dma-engine sh-dma-engine.2: Runtime PM disabled, clock forced on.
sh-dma-engine sh-dma-engine.3: Runtime PM disabled, clock forced on.
SuperH SCI(F) driver initialized
sh-sci sh-sci.0: Runtime PM disabled, clock forced on.
sh-sci.0: ttySC0 at MMIO 0xe6c40000 (irq = 80) is a scifa
console [ttySC2] enabled, bootconsole disabled
console [ttySC2] enabled, bootconsole disabled
sh-sci sh-sci.1: Runtime PM disabled, clock forced on.
sh-sci.1: ttySC1 at MMIO 0xe6c50000 (irq = 81) is a scifa
sh-sci sh-sci.2: Runtime PM disabled, clock forced on.
sh-sci.2: ttySC2 at MMIO 0xe6c60000 (irq = 82) is a scifa
sh-sci sh-sci.3: Runtime PM disabled, clock forced on.
sh-sci.3: ttySC3 at MMIO 0xe6c70000 (irq = 83) is a scifa
sh-sci sh-sci.4: Runtime PM disabled, clock forced on.
sh-sci.4: ttySC4 at MMIO 0xe6c80000 (irq = 89) is a scifa
sh-sci sh-sci.5: Runtime PM disabled, clock forced on.
sh-sci.5: ttySC5 at MMIO 0xe6cb0000 (irq = 90) is a scifa
sh-sci sh-sci.6: Runtime PM disabled, clock forced on.
sh-sci.6: ttySC6 at MMIO 0xe6cc0000 (irq = 22) is a scifa
sh-sci sh-sci.7: Runtime PM disabled, clock forced on.
sh-sci.7: ttySC7 at MMIO 0xe6cd0000 (irq = 23) is a scifa
sh-sci sh-sci.8: Runtime PM disabled, clock forced on.
sh-sci.8: ttySC8 at MMIO 0xe6c30000 (irq = 91) is a scifb
brd: module loaded
loop: module loaded
physmap platform flash device: 08000000 at 00000000
physmap-flash.0: Found 1 x16 devices at 0x0 in 16-bit bank. Manufacturer ID 0x00
0089 Chip ID 0x008965
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
Creating 6 MTD partitions on "physmap-flash.0":
0x000000000000-0x0000000040000 : "bootloader"
0x0000000040000-0x0000000080000 : "config"
0x0000000080000-0x00000000c0000 : "license"
0x00000000c0000-0x00000004c0000 : "firmware"
0x00000004c0000-0x00000008c0000 : "kernel"
0x00000008c0000-0x0000004000000 : "userland"
pegasus: v0.6.14 (2006/09/27), Pegasus/Pegasus II USB Ethernet driver
```

```
usbcore: registered new interface driver pegasus
usbcore: registered new interface driver asix
ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver
rmobile-ehci-driver rmobile-ehci-driver: R-Mobile EHCI
rmobile-ehci-driver rmobile-ehci-driver: new USB bus registered, assigned bus number 1
rmobile-ehci-driver rmobile-ehci-driver: irq 158, io mem 0xc6701000
rmobile-ehci-driver rmobile-ehci-driver: USB 2.0 started, EHCI 1.00
hub 1-0:1.0: USB hub found
hub 1-0:1.0: 1 port detected
ohci_hcd: USB 1.1 'Open' Host Controller (OHCI) Driver
rmobile-ohci-driver rmobile-ohci-driver: R-Mobile OHCI
rmobile-ohci-driver rmobile-ohci-driver: new USB bus registered, assigned bus number 2
rmobile-ohci-driver rmobile-ohci-driver: irq 158, io mem 0xc6700000
hub 2-0:1.0: USB hub found
hub 2-0:1.0: 1 port detected
Initializing USB Mass Storage driver...
usbcore: registered new interface driver usb-storage
USB Mass Storage support registered.
renesas_usbhs renesas_usbhs: Runtime PM disabled, clock forced on.
renesas_usbhs renesas_usbhs: gadget probed
renesas_usbhs renesas_usbhs: probed
  gadget: using random self ethernet address
  gadget: using random host ethernet address
usb0: MAC 6e:c9:e7:ed:79:4f
usb0: HOST MAC 4a:9a:9f:cc:ed:c4
  gadget: UVC Composite Gadget, version: 0.9.0
  gadget: userspace failed to provide iSerialNumber
  gadget: g_uvc_acm_ether ready
mousedev: PS/2 mouse device common for all mice
hub 1-0:1.0: over-current condition on port 1
rtc-s35390a 2-0030: rtc core: registered rtc-s35390a as rtc0
i2c /dev entries driver
sh_mobile_ceu sh_mobile_ceu.0: Runtime PM disabled, clock forced on.
soc-camera-pdrv soc-camera-pdrv.0: Probing soc-camera-pdrv.0
sh_mobile_ceu sh_mobile_ceu.0: SuperH Mobile CEU driver attached to camera 0
ov772x 1-0021: ov7725 Product ID 77:21 Manufacturer ID 7f:a2
sh_mobile_ceu sh_mobile_ceu.0: SuperH Mobile CEU driver detached from camera 0
sh_mobile_wdt sh_mobile_wdt.0: Runtime PM disabled, clock forced on.
device-mapper: ioctl: 4.22.0-ioctl (2011-10-19) initialised: dm-devel@redhat.com
sh_mobile_sdhi sh_mobile_sdhi.0: Runtime PM disabled, clock forced on.
sh_mobile_sdhi sh_mobile_sdhi.0: Platform OCR mask is ignored
sh_mobile_sdhi sh_mobile_sdhi.0: mmc0 base at 0xe6850000 clock rate 99 MHz
usbcore: registered new interface driver usbhid
usbhid: USB HID core driver
usbcore: registered new interface driver snd-usb-audio
ip_tables: (C) 2000-2006 Netfilter Core Team
TCP: cubic registered
NET: Registered protocol family 17
VFP support v0.3: implementor 41 architecture 3 part 30 variant 9 rev 3
registered taskstats version 1
rtc-s35390a 2-0030: setting system clock to 2000-01-01 00:00:00 UTC (946684800)
ALSA device list:
  No soundcards found.
RAMDISK: ext2 filesystem found at block 0
RAMDISK: Loading 23537KiB [1 disk] into ram disk... done.
VFS: Mounted root (ext2 filesystem) on device 1:0.
```

```
Freeing init memory: 148K
Mounting proc: done
Starting fsck for root filesystem.
fsck 1.25 (20-Sep-2001)
/dev/ram0: clean, 1207/1464 files, 21299/23537 blocks
Checking root filesystem: done
Remounting root rw: done
Mounting usbfs: done
Mounting sysfs: done
Mounting tmpfs on /dev: done
Cleaning up system: done
Running local start scripts.
Creating mtd devnode: done
Loading /etc/config: done
Starting udevd: done
Mounting devpts: done
Changing file permissions: done
Configure /home/ftp: done
Starting syslogd: done
Starting klogd: done
Mounting firmware on /opt/firmware: done
Mounting license on /opt/license: done
Mounting tmpfs on /tmp, /var/tmp: done
Mounting ramfs on /home/ftp/pub: done
Setting hostname: done
Starting basic firewall: done
Configuring network interfaces: done
Starting inetd: done
Creating avahi.services: done
Starting avahi.daemon: done
Starting lighttpd: done
Starting sshd: failed
(sshd: you will be available to use after run '/etc/init.d/sshd keygen')
Running local start script (/etc/config/rc.local).
Starting uvc-gadget: done

atmark-dist v1.31.0 (AtmarkTechno/Armadillo-810)
Linux 3.4-at1 [armv7l arch]

armadillo810-0 login:
```

図 4.1 起動ログ

4.2. 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 4.1. 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 4.1 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
guest	(なし)	一般ユーザ

4.3. 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.
Syncing all filesystems: done
Unmounting all filesystems: done
The system is going down NOW!
Sent SIGTERM to all processes
Sent SIGKILL to all processes
Requesting system halt
System halted.
```

図 4.2 終了方法

SD カードなどのストレージをマウントしていない場合は、電源を切断し終了させることもできます。



ストレージ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。ストレージをアンマウントしてから電源を切断するようご注意ください。

5. 動作確認方法

5.1. USB ガジェット

Armadillo-810 を USB デバイスとして使用することができます。ここでは動作確認を ATDE5 で行うため、「3.2.2. 取り外し可能デバイスの使用」を参照して ATDE5 と Armadillo-810 CPU ボードの CON4 を USB2.0 ケーブルで接続する必要があります。

USB デバイスの機能は、Linux カーネルの USB ガジェットドライバによって提供されます。USB ガジェットドライバは様々な種類のもので用意されていますが、工場出荷イメージでは Armadillo-810 を USB 複合デバイス^[1]として使用することができる「UVC Composite Gadget」が有効になっています。

UVC Composite Gadget は以下に示す 3 種類の USB デバイス機能を持っています。各機能は同時に利用することができます。

UVC ガジェット

Armadillo をビデオ出力デバイス、ATDE をビデオ入力デバイスとして扱うことができます。V4L2 インターフェースを利用するキャプチャーアプリケーションなどで利用することができます。

シリアルガジェット

Armadillo と ATDE を、互いにシリアルデバイスとして扱うことができます。シリアル(tty)デバイスで通信するアプリケーションなどで利用することができます。

イーサネットガジェット

Armadillo と ATDE を、互いにネットワークデバイスとして扱うことができます。ソケットで通信するアプリケーションで利用することができます。

5.1.1. UVC ガジェット

Armadillo-810 の UVC ガジェットは、「UVC(USB Video Class)」として実装されています。ATDE が UVC ガジェットを認識すると、/dev/video0 というデバイスファイルが作成されます。アプリケーションは、このデバイスを介して画像を取得することができます。ここでは、「gview」というアプリケーションで動作の確認を行います。

5.1.1.1. 起動方法

「図 5.1. gview を起動」を実行すると、「図 5.2. gview のビデオウィンドウ」と「図 5.3. gview のコントロールウィンドウ」の 2 つのウィンドウが立ち上がります。ビデオウィンドウには、UVC ガジェットから取得した映像が表示され、コントロールウィンドウで解像度の切り替えなどを行うことができます。

```
[ATDE ~]$ gview -w 0
```

図 5.1 gview を起動

[1]複数の USB 機能を持ったデバイスのこと。



図 5.2 guvcview のビデオウィンドウ^[2]

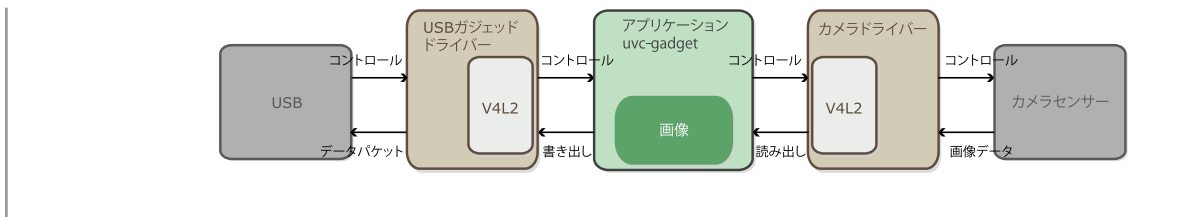


図 5.3 guvcview のコントロールウィンドウ



UVC ガジェットの機能は、UVC ガジェットドライバとカメラドライバで画像を転送するような形で実現しています。2つのドライバは「V4L2(Video for Linux 2)」という共通のインターフェースを持っています。V4L2を利用して画像を転送するのが「`uvc-gadget`」というアプリケーションです。UVC ガジェットの概念図を次に示します。

^[2]画面は、ハメコミ合成しています。



5.1.2. シリアルガジェット

Armadillo-810 のシリアルガジェットは、「CDC-ACM(USB Communication Device Class - Abstract Control Model)」として実装されています。Armadillo-810 では/dev/ttyGS0 を、ATDE では/dev/ttyACM0 を使用したシリアル通信を行うことができます。

シリアルガジェットの動作を確認するには、Armadillo-810 の/dev/ttyGS0 にシリアルコンソールを起動させます。ATDE のシリアル通信ソフトウェア(minicom)を用いることで、シリアルガジェット経由で Armadillo-810 にログインすることができます。

手順 5.1 CDC-ACM 通信確認手順

1. ATDE で minicom を起動します。シリアルデバイスには/dev/ttyACM0 を指定します。

```
[ATDE ~]$ minicom -o -w -D /dev/ttyACM0
```


図 5.4 /dev/ttyACM0 を指定してシリアルターミナルを起動

2. Armadillo で getty を起動します。シリアルデバイスには ttyGS0^[3]を指定します。/etc/inittab の設定を有効にするためには、プロセス ID が 1 である init プロセスに SIGHUP シグナルを送る必要があります。

```
[armadillo ~]# echo ::respawn:/sbin/getty -L 115200 ttyGS0 >> /etc/inittab
[armadillo ~]# kill -SIGHUP 1
```

図 5.5 /dev/ttyGS0 上でシリアルコンソールを起動

ATDE の minicom にログインプロンプトが表示されます。ユーザー「guest」でログインすることができます。



以下のように/etc/securetty に端末(シリアルデバイス)を登録すると、特権ユーザー「root」でログインすることが可能になります。


```
[armadillo ~]# echo ttyGS0 >> /etc/securetty
```

^[3]/dev/を指定する必要はありません。

5.1.3. イーサネットガジェット

Armadillo-810 のイーサネットガジェットは、「RNDIS(Remote NDIS)」として実装されています。Armadillo-810 と ATDE は、互いにネットワークインターフェース usb0 を使用したネットワーク通信を行うことができます。

Armadillo-810 と ATDE をイーサネットガジェットで接続すると、IPv4LL という機構を使ってリンクローカルアドレス^[4]が設定されます。



ATDE5 のネットワーク設定は、ネットワークマネージャーを利用せずに "/etc/network/interfaces" に基づいて設定されています。


ATDE でイーサネットガジェットを認識したかどうかは、"ifconfig" コマンドの出力結果をみると判断することができます。

```
[ATDE ~]$ LANG=C sudo ifconfig
(省略)
usb0      Link encap:Ethernet  HWaddr a2:e4:92:6f:c6:54 ①
          inet6 addr: fe80::a0e4:92ff:fe6f:c654/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:12 errors:0 dropped:0 overruns:0 frame:0
          TX packets:77 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:905 (905.0 B)  TX bytes:13943 (13.6 KiB)

usb0:avahi Link encap:Ethernet  HWaddr a2:e4:92:6f:c6:54 ②
          inet addr:169.254.9.18 Bcast:169.254.255.255 Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

- ① イーサネットガジェットを認識すると"usb0"が表示されます
- ② リンクローカルアドレスが割り当てられた場合に"usb0:avahi"が表示されます

図 5.6 イーサネットガジェット認識時の ifconfig の出力例



イーサネットガジェットのネットワークインターフェース(usb0)の MAC アドレスには、Linux カーネルが生成したランダム値が設定されます。この MAC アドレスのうち、特定の意味を持つ I/G(Individual/Group)および U/L(Universal/Local)ビットについてはランダム値ではなく固定の値が設定されます。I/G ビットは 0(ユニキャストアドレス)に、U/L ビットは 1(ローカルアドレス)に設定されます。

Armadillo-810 と ATDE をイーサネットガジェットで接続すると、マルチキャスト DNS(mDNS)という技術を利用してローカルネットワークで利用することができるホスト名が設定されます。mDNS により設定されるホスト名を次に示します。

^[4]IPv4LL によって割り当てられる特定のアドレス範囲(169.254.0.1~169.254.255.254)の IP アドレス。

表 5.1 mDNS で設定されるホスト名

ホスト	ホスト名
Armadillo-810	armadillo810-0.local
ATDE5	atde5.local



ATDE5 が所属するネットワーク内に複数の ATDE5 が存在する場合は、mDNS で設定されるホスト名が重複しないように "atde5-2.local" のようなホスト名が設定されます。

ATDE5 に mDNS で設定されたホスト名を確認するには、次のようにコマンドを実行します。

```
[ATDE ~]$ ps axu | grep avahi-daemon
avahi      2464  0.0  0.0 34156 1700 ?        S   12:40   0:00 avahi-
daemon: running [atde5-2.local]
(省略)
```



ネットワーク設定が完了すると、ネットワーク通信ができる状態となります。ping で通信させてみましょう。

```
[ATDE ~]$ ping armadillo810-0.local
```

図 5.7 イーサネットガジェットの通信確認

5.2. ネットワーク

イーサネットガジェットを利用することにより、ネットワーク機能を利用することができます。ここでは、ネットワークの設定方法やネットワークを利用するアプリケーションについて説明します。

5.2.1. ネットワーク設定の変更方法

Armadillo のネットワーク設定の変更方法について説明します。



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。

5.2.1.1. Armadillo にログインしてネットワーク設定を変更する

Armadillo 上の「/etc/config」以下にあるファイルを編集し、コンフィグ領域に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。

5.2.1.1.1. デフォルト状態のネットワーク設定

ネットワーク設定は、`/etc/config/interfaces` に記述されています。デフォルト状態では、次のように設定されています。

```
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

auto lo usb0
iface lo inet loopback
iface eth0 inet dhcp
iface usb0 inet manual ❶
    up ifconfig usb0 up
    post-up zcip usb0 /etc/zcip.script > /dev/null
    down ifconfig usb0 down
```

❶ イーサネットガジェット(usb0)用の設定は、リンクローカルアドレスを利用

図 5.8 デフォルト状態の/etc/config/interfaces

5.2.1.1.2. 固定 IP アドレスに設定する

「表 5.2. 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 5.9. 固定 IP アドレス設定」のように編集します。

表 5.2 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 usb0
iface lo inet loopback
iface eth0 inet dhcp
iface usb0 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 5.9 固定 IP アドレス設定

5.2.1.1.3. DHCP に設定する



イーサネットガジェットは、IPv4LL によってリンクローカルアドレスが設定されるため、通常 DHCP に設定する必要はありません。

DHCP に設定するには、vi エディタで/etc/config/interfaces を、次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 usb0
iface lo inet loopback
iface eth0 inet dhcp
iface usb0 inet dhcp
```

図 5.10 DHCP 設定

5.2.1.1.4. DNS サーバーを指定する

DNS サーバーを指定する場合は、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 5.11 DNS サーバーの設定



DHCP を利用している場合には、DHCP サーバーが DNS サーバーを通知する場合があります。この場合、/etc/config/resolv.conf は自動的に更新されます。

5.2.1.2. 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。次のように設定を反映させます。

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

図 5.12 設定を反映させる



ネットワークの設定をコンフィグ領域に保存し、Armadillo を再起動している場合には「図 5.12. 設定を反映させる」に示す操作は不要です。

同じネットワーク内にある通信機器と PING 通信を行います。

```
[armadillo ~]# ping 192.168.10.20
```

図 5.13 PING 確認

5.2.2. ファイアーウォール

Armadillo では、簡易ファイアー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 5.14. iptables」のようにコマンド実行してください。

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

図 5.14 iptables

5.2.3. ネットワークアプリケーション

工場出荷イメージで利用することができるネットワークアプリケーションについて説明します。



ATDE と Armadillo のネットワーク設定がデフォルト状態であることを想定して記述しています。ネットワーク設定を変更している場合は適宜読み換えてください。

5.2.3.1. TELNET

ATDE などの PC からネットワーク経由でログインし、リモート操作することができます。ログイン可能なユーザを次に示します。

表 5.3 TELNET でログイン可能なユーザ

ユーザ名	パスワード
guest	(なし)

TELNET を使用して ATDE から Armadillo にリモートログインする場合の例を、次に示します。

```
[ATDE ~]$ telnet armadillo810-0.local
Trying 169.254.245.125...
Connected to armadillo810-0.local.
Escape character is '^'.

atmark-dist v1.31.0 (AtmarkTechno/Armadillo-810)
Linux 3.4-at1 [armv7l arch]

armadillo810-0 login: guest ❶
[guest@armadillo ~]$
[guest@armadillo ~]$ su ❷
Password: ❸
[root@armadillo ~]#
[root@armadillo ~]# exit ❹
[guest@armadillo ~]$ exit ❺
Connection closed by foreign host.
[ATDE ~]$
```

- ❶ "guest"と入力するとログインすることができます。パスワードの入力は不要です。
- ❷ 特権ユーザーとなる場合には"su"コマンドを実行します。
- ❸ 特権ユーザーのデフォルトパスワードは"root"です。
- ❹ 特権トユーザーから guest ユーザーに戻る場合は、"exit"と入力します
- ❺ telnet を終了するにはもう一度"exit"を入力します

図 5.15 telnet でリモートログイン

5.2.3.2. FTP

ATDE などの PC からネットワーク経由でファイル転送することができます。次に示すユーザでログインすることができます。

表 5.4 ftp でログイン可能なユーザ

ユーザ名	パスワード
ftp	(なし)

ftp を使用して ATDE から Armadillo にファイルを転送する場合の例を、次に示します。

```
[ATDE ~]$ ls -l file
-rw-r--r-- 1 atmark atmark 1048576 Jan 1 12:00 file
[ATDE ~]$ ftp armadillo810-0.local
Connected to armadillo810-0.local.
220 localhost FTP server (GNU inetutils 1.4.1) ready.
Name (armadillo810-0.local:atmark): ftp
331 Guest login ok, type your name as password.
Password: ❶
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd pub ❷
250 CWD command successful.
ftp> put file ❸
local: file remote: file
200 PORT command successful.
150 Opening BINARY mode data connection for 'file'.
226 Transfer complete.
1048576 bytes sent in 0.14 secs (7399.5 kB/s)
ftp> quit ❹
221 Goodbye.
[ATDE ~]$
```

- ❶ ftp ユーザにパスワードが設定されていないため Enter キーを入力します。
- ❷ ファイル転送することができる pub ディレクトリに移動します。
- ❸ ファイルをアップロードします。ダウンロードする場合は"get"コマンドを使用します。
- ❹ ftp を終了する場合は"quit"と入力します。

図 5.16 ftp でファイル転送

ATDE から Armadillo にファイルをアップロードすると、/home/ftp/pub/ディレクトリ以下にファイルが作成されています。ダウンロードする場合も、同じディレクトリにファイルを配置してください。

```
[armadillo ~]# cd /home/ftp/pub/
[armadillo /home/ftp/pub]# ls
file
```

図 5.17 Armadillo 上でアップロードされたファイルを確認

5.2.3.3. HTTP サーバー

Armadillo では、HTTP サーバーが動作しています。ATDE などの PC の Web ブラウザから Armadillo の URL (<http://armadillo810-0.local/> または、[http://\[ArmadilloのIPアドレス\]/](http://[ArmadilloのIPアドレス]/) ^[5]) にアクセスすると、Armadillo のトップページ(index.html)が表示されます。

[5] Armadillo の IP アドレスが 192.168.10.10 の場合、<http://192.168.10.10/> となります。

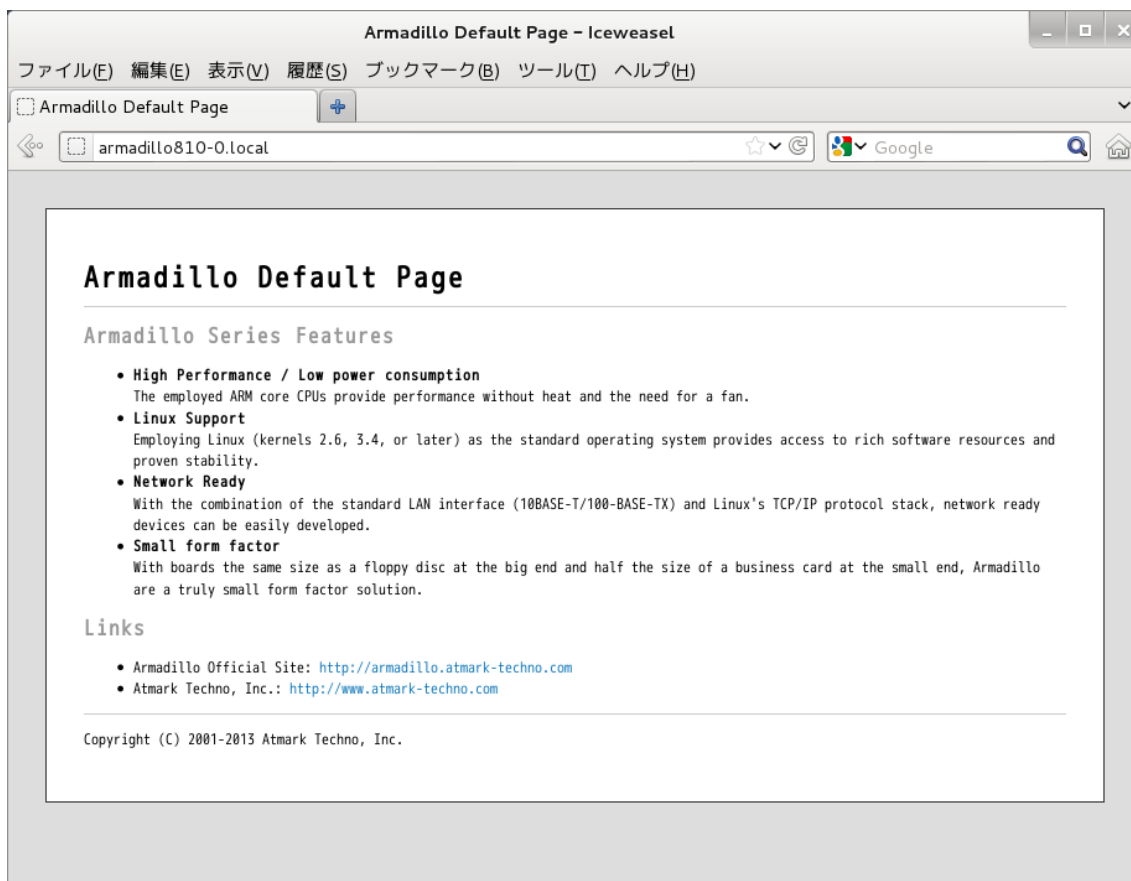


図 5.18 Armadillo トップページ

5.3. シリアル

Armadillo-810 でシリアルポートとして使用可能なデバイスを次に示します。

表 5.5 シリアルデバイス

インターフェース	デバイスファイル(ATDE5)	デバイスファイル(Armadillo-810)
シリアルインターフェース 1(CON2)	/dev/ttyS0	/dev/ttySC8
シリアルインターフェース 2(CON3)	/dev/ttyUSB0	/dev/ttySC2
USB インターフェース(CON4) ^[a]	/dev/ttyACM0	/dev/ttyGS0

^[a] 「5.1. USB ガジェット」 に示すシリアルガジェットを使用します。



「表 5.5. シリアルデバイス」 に示すデバイスファイルは、「3.5. 接続方法」に従って接続された場合のもので、異なる接続をしている場合は適宜読み替えてください。

5.3.1. シリアルコンソールとして使用する

シリアルの動作を確認するには、シリアルコンソールを起動させます。ATDE のシリアル通信ソフトウェア(minicom)を用いることで、シリアル経由で Armadillo にログインすることができます。



Armadillo-810 の工場出荷イメージでは、シリアルインターフェース 2(CON3)が標準でシリアルコンソールとして使用できるようになっています。

シリアルインターフェース 1(CON2)をシリアルコンソールとして使用する手順を次に示します。

手順 5.2 シリアルコンソールとして使用

1. ATDE で minicom を起動します。シリアルデバイスには /dev/ttyS0 を指定します。

```
[ATDE ~]$ minicom -o -w -D /dev/ttyS0
```

2. Armadillo で getty を起動します。シリアルデバイスには ttySC8^[6]を指定します。/etc/inittab の設定を有効にするためには、プロセス ID が 1 である init プロセスに SIGHUP シグナルを送る必要があります。

```
[armadillo ~]# echo ::respawn:/sbin/getty -L 115200 ttySC8 >> /etc/inittab  
[armadillo ~]# kill -SIGHUP 1
```

ATDE の minicom にログインプロンプトが表示されます。ユーザー「guest」でログインすることができます。



以下のように /etc/securetty に端末(シリアルデバイス)を登録すると、特権ユーザー「root」でログインすることが可能になります。

```
[armadillo ~]# echo ttySC8 >> /etc/securetty
```

5.3.2. ファイルを転送する

シリアルインターフェースを使用して、ファイルを転送することができます。ここでは、バイナリファイルの転送が可能な ZMODEM(バイナリ転送プロトコル)を使用する方法を説明します。

ここでは、ファイル転送を行うための作業ディレクトリ ~/minicom に移動して行います。

```
[ATDE ~]$ mkdir minicom  
[ATDE ~]$ cd minicom  
[ATDE ~/minicom]$
```

ファイルを転送するには、事前に「5.3.1. シリアルコンソールとして使用する」を参照して Armadillo にログインする必要があります。minicom を起動する前に、作業ディレクトリ ~/minicom に移動します。

^[6]/dev/を指定する必要はありません。

5.3.2.1. ファイルのアップロード

Armadillo から ATDE にファイルをアップロードする方法を説明します。

手順 5.3 Armadillo からファイルをアップロードする

1. "lsz"コマンドでファイル(send.bin)をアップロードします。"press any key to continue..."と表示されてから任意のキーを入力します。

```
[armadillo ~]$ ls
send.bin
[armadillo ~]$ lsz send.bin

+-----[zmodem download - Press CTRL-C to quit]-----+
|Receiving: send.bin
|Bytes received:  3072/  3072  BPS:10777
|Transfer complete
|  READY: press any key to continue...
+-----+

[armadillo ~]$
```

2. ATDE で minicom を起動したディレクトリ直下に、ファイルが転送されています

```
[ATDE ~/minicom]$ ls send.bin
send.bin
```



minicom の自動 ZMODEM ダウンロード機能により、Armadillo からアップロードしたファイルを自動的にダウンロードすることができます。

5.3.2.2. ファイルのダウンロード

Armadillo で ATDE 上のファイルをダウンロードする方法を説明します。

手順 5.4 Armadillo でファイルをダウンロードする

1. "lrz"コマンドを実行するとファイルのダウンロード待ち状態になります。

```
[armadillo ~]$ lrz
```

2. minicom で Ctrl+a に続いて s キーを入力し、ファイルのアップロードを開始します。
3. minicom でファイルをアップロードするプロトコルを選択します。「zmodem」にカーソルを合わせて Enter キーを入力します。

```

+-----[Upload]-----+
|zmodem
|ymodem
|xmodem
|kermit
|asci i
|hermit(bootloader)
|hermit(kernel)
|hermit(userland)
+-----+
    
```

4. minicom で Armadillo にアップロードするファイルを選択します。アップロードするファイルにカーソルを合わせて Space キーを入力し、ファイル選択を行った後に Enter キーを入力します。

```

+-----[Select one or more files for upload]-----+
|Directory: /home/atmark/minicom
| [..]
|  recv.bin
+-----+
          ( Escape to exit, Space to tag)
+-----+
          [Goto] [Prev] [Show] [Tag] [Untag] [Okay]
    
```

5. "press any key to continue..."と表示されてから任意のキーを入力します。

```

+-----[zmodem download - Press CTRL-C to quit]-----+
|Sending: recv.bin
|Bytes Sent:   3072   BPS:9707
|
|Transfer complete
|  READY: press any key to continue...
+-----+
[armadillo ~]$
    
```

6. Armadillo で lrz を起動したディレクトリ直下に、ファイルが転送されています

```

[armadillo ~]$ ls recv.bin
recv.bin
    
```

5.4. ストレージ

Armadillo-810 でストレージとして使用可能なデバイスを次に示します。

表 5.6 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 フラッシュメモリ	/dev/sd*[a]	/dev/sd*1
SD カード	/dev/mmcblk0	/dev/mmcblk0p1

^[a]USB ハブを利用して複数の USB メモリを接続した場合は、認識された順に sda sdb sdc ... となります。

5.4.1. ストレージの使用方法

ここでは、SD カードを例にストレージの使用方法を説明します。

Linux では、アクセス可能なファイルやディレクトリは、一つの木構造にまとめられています。あるストレージデバイスのファイルシステムを、この木構造に追加することを、マウントするといいます。マウントを行うコマンドは、mount です。

mount コマンドの典型的なフォーマットは、次の通りです。

```
mount -t fstype device dir
```

図 5.19 mount コマンド書式

-t オプションに続く device には、ファイルシステムタイプを指定します^[7]。FAT32 ファイルシステムの場合は vfat^[8]、EXT3 ファイルシステムの場合は ext3 を指定します。

device には、ストレージデバイスのデバイスファイル名を指定します。SD カードのパーティション 1 の場合は /dev/mmcblk0p1、パーティション 2 の場合は /dev/mmcblk0p2 となります。

dir には、ストレージデバイスのファイルシステムをマウントするディレクトリを指定します。

SD スロットに SD カードを挿入した状態で「図 5.20. ストレージのマウント」に示すコマンドを実行すると、/mnt ディレクトリに SD カードのファイルシステムをマウントします。SD カード内のファイルは、/mnt ディレクトリ以下に見えるようになります。

```
[armadillo ~]# mount -t vfat /dev/mmcblk0p1 /mnt
```

図 5.20 ストレージのマウント



FAT32 ファイルシステムをマウントした場合、次の警告メッセージが表示される場合があります。

```
FAT-fs (mmcblk0p1): utf8 is not a recommended IO charset for
FAT filesystems, filesystem will be case sensitive!
```

^[7]ファイルシステムタイプの指定は省略可能です。省略した場合、mount コマンドはファイルシステムタイプを推測します。この推測は必ずしも適切なものとは限りませんので、事前にファイルシステムタイプが分かっている場合は明示的に指定してください。

^[8]通常、購入したばかりの SD カードは FAT32 ファイルシステムでフォーマットされています。

これは無視して構いません。UTF-8 ロケールでは結局はファイル名の表示を正しく処理できないためです。

ストレージを安全に取り外すには、アンマウントする必要があります。アンマウントを行うコマンドは、`umount` です。オプションとして、アンマウントしたいデバイスがマウントされているディレクトリを指定します。

```
[armadillo ~]# umount /mnt
```

図 5.21 ストレージのアンマウント

5.4.2. ストレージのパーティション変更とフォーマット

通常、購入したばかりの SD カードや USB メモリは、一つのパーティションを持ち、FAT32 ファイルシステムでフォーマットされています。

パーティション構成を変更したい場合、`fdisk` コマンドを使用します。`fdisk` コマンドの使用例として、一つのパーティションで構成されている SD カードのパーティションを、2 つに分割する例を「図 5.22. `fdisk` コマンドによるパーティション変更」に示します。一度、既存のパーティションを削除してから、新たにプライマリパーティションを二つ作成しています。先頭のパーティションには 100MByte、二つめのパーティションに残りの容量を割り当てています。先頭のパーティションは `/dev/mmcbk0p1`、二つめは `/dev/mmcbk0p2` となります。`fdisk` コマンドの詳細な使い方は、man ページ等をご参照ください。

```
[armadillo ~]# fdisk /dev/mmcbk0
```

```
The number of cylinders for this disk is set to 62528.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
 1) software that runs at boot time (e.g., old versions of LILO)
 2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): d
Selected partition 1

Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-62528, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-62528, default 62528): +100M

Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (3054-62528, default 3054):
```

```
Using default value 3054
Last cylinder or +size or +sizeM or +sizeK (3054-62528, default 62528):
Using default value 62528

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

Calling ioctl() to re-read partition table.
mmcblk0: p1 p2
mmcblk0: p1 p2
Syncing disks.
```

図 5.22 fdisk コマンドによるパーティション変更

FAT32 ファイルシステムでストレージデバイスをフォーマットするには、mkfs.vfat コマンドを使用します。また、EXT2 や EXT3 ファイルシステムでフォーマットするには、mke2fs コマンドを使用します。SD カードのパーティション 1 を EXT3 ファイルシステムでフォーマットするコマンド例を、次に示します。

```
[armadillo ~]# mke2fs -j /dev/mmcblk0p1
```

図 5.23 EXT3 ファイルシステムの構築

5.5. LED

Armadillo-810 の LED は、LED クラスとして実装されています。LED クラスディレクトリ以下のファイルによって LED の制御を行うことができます。LED クラスディレクトリと LED の対応については、「表 5.7. LED クラスディレクトリと LED の対応」を参照してください。

表 5.7 LED クラスディレクトリと LED の対応

LED クラスディレクトリ	説明	デフォルトトリガ
/sys/class/leds/LED1/	LED1 (黄)	none
/sys/class/leds/LED2/	LED2 (黄)	none
/sys/class/leds/LED3/	LED3 (黄)	none
/sys/class/leds/LED4/	LED4 (黄)	none

以降の説明では、任意の LED を示す LED クラスディレクトリを"/sys/class/leds/[LED]"のように表記します。


5.5.1. LED を点灯/消灯する

LED クラスディレクトリ以下の brightness ファイルへ値を書き込むことによって、LED の点灯/消灯を行うことができます。brightness に書き込む有効な値は 0~255 です。

brightness に 0 以外の値を書き込むと LED が点灯します。

```
[armadillo ~]# echo 1 > /sys/class/leds/[LED]/brightness
```

図 5.24 LED を点灯させる



Armadillo-810 の LED には輝度制御の機能が無いため、0 (消灯)、1～255 (点灯)の 2 つの状態のみ指定することができます。

brightness に 0 を書き込むと LED が消灯します。

```
[armadillo ~]# echo 0 > /sys/class/leds/[LED]/brightness
```

図 5.25 LED を消灯させる

brightness を読み出すと LED の状態が取得できます。

```
[armadillo ~]# cat /sys/class/leds/[LED]/brightness
0
```

図 5.26 LED の状態を表示する

5.5.2. トリガを使用する

LED クラスディレクトリ以下の trigger ファイルへ値を書き込むことによって LED の点灯/消灯にトリガを設定することができます。trigger に書き込む有効な値を次に示します。

表 5.8 trigger の種類

設定	説明
none	トリガを設定しません。
mmc0	SD カードのアクセスランプにします。
timer	任意のタイミングで点灯/消灯を行います。この設定にすることにより、LED クラスディレクトリ以下に delay_on, delay_off ファイルが出現し、それぞれ点灯時間, 消灯時間をミリ秒単位で指定します。
heartbeat	心拍のように点灯/消灯を行います。工場出荷イメージでは設定することができません。
default-on	主にカーネルから使用します。起動時に LED が点灯します。工場出荷イメージでは設定することができません。

以下のコマンドを実行すると、LED が 1 秒点灯、500 ミリ秒消灯を繰り返します。

```
[armadillo ~]# echo timer > /sys/class/leds/[LED]/trigger
[armadillo ~]# echo 1000 > /sys/class/leds/[LED]/delay_on
[armadillo ~]# echo 500 > /sys/class/leds/[LED]/delay_off
```

図 5.27 LED のトリガに timer を指定する

trigger を読み出すと LED のトリガが取得できます。"[]"が付いているものが現在のトリガです。

```
[armadillo ~]# cat /sys/class/leds/[LED]/trigger
[none] mmc0 timer
```

図 5.28 LED のトリガを表示する

5.6. GPIO

Armadillo-810 の GPIO は、generic GPIO として実装されています。GPIO クラスディレクトリ以下のファイルによって GPIO の制御を行うことができます。GPIO クラスディレクトリと GPIO の対応を次に示します。

表 5.9 Armadillo-810 拡張ボード 01 (A コネクタ用)の CON2 の GPIO ディレクトリ

ピン番号	GPIO ディレクトリ
CON2 1 ピン	/sys/class/gpio/gpio66/
CON2 2 ピン	/sys/class/gpio/gpio67/
CON2 3 ピン	/sys/class/gpio/gpio68/
CON2 4 ピン	/sys/class/gpio/gpio69/
CON2 5 ピン	/sys/class/gpio/gpio70/
CON2 6 ピン	/sys/class/gpio/gpio71/
CON2 7 ピン	/sys/class/gpio/gpio72/
CON2 8 ピン	/sys/class/gpio/gpio73/
CON2 9 ピン	/sys/class/gpio/gpio74/
CON2 10 ピン	/sys/class/gpio/gpio75/
CON2 13 ピン	/sys/class/gpio/gpio160/
CON2 14 ピン	/sys/class/gpio/gpio161/
CON2 15 ピン	/sys/class/gpio/gpio159/
CON2 16 ピン	/sys/class/gpio/gpio162/
CON2 18 ピン	/sys/class/gpio/gpio7/
CON2 19 ピン	/sys/class/gpio/gpio8/
CON2 20 ピン	/sys/class/gpio/gpio10/
CON2 21 ピン	/sys/class/gpio/gpio11/
CON2 22 ピン	/sys/class/gpio/gpio20/
CON2 23 ピン	/sys/class/gpio/gpio12/
CON2 24 ピン	/sys/class/gpio/gpio5/
CON2 25 ピン	/sys/class/gpio/gpio13/
CON2 26 ピン	/sys/class/gpio/gpio9/

以降の説明では、任意の GPIO を示す GPIO クラスディレクトリを"/sys/class/gpio/[GPIO]"のように表記します。

5.6.1. 入出力方向を変更する

GPIO ディレクトリ以下の direction ファイルへ値を書き込むことによって、入出力方向を変更することができます。direction に書き込む有効な値を次に示します。

表 5.10 direction の設定

設定	説明
high	入出力方向を OUTPUT に設定します。入力レベルの取得/設定を行うことができます。入力レベルは HIGH レベルになります。
out	入出力方向を OUTPUT に設定します。入力レベルの取得/設定を行うことができます。入力レベルは LOW レベルになります。
low	out を設定した場合と同じです。
in	入出力方向を INPUT に設定します。入力レベルの取得を行うことができますが設定はできません。

5.6.2. 入力レベルを取得する

GPIO ディレクトリ以下の value ファイルから値を読み出すことによって、入力レベルを取得することができます。"0"は LOW レベル、"1"は HIGH レベルを表わします。入力レベルの取得は入出力方向が INPUT, OUTPUT のどちらでも行うことができます。

```
[armadillo ~]# cat /sys/class/gpio/[GPIO]/value
0
```

図 5.29 GPIO の入力レベルを取得する

5.6.3. 出力レベルを設定する


GPIO ディレクトリ以下の value ファイルへ値を書き込むことによって、出力レベルを設定することができます。"0"は LOW レベル、"0"以外は HIGH レベルを表わします。出力レベルの設定は入出力方向が OUTPUT でなければ行うことはできません。

```
[armadillo ~]# echo 1 > /sys/class/gpio/[GPIO]/value
```

図 5.30 GPIO の出力レベルを設定する

5.7. RTC

Armadillo-810 拡張ボード 01 (A コネクタ用)には、カレンダー時計(Real Time Clock)が実装されています。電源を切断しても一定時間(平均 300 秒間、最小 60 秒間)時刻を保持することができます



長時間電源が切断されても時刻データを保持させたい場合は、Armadillo-810 拡張ボード 01 (A コネクタ用)の CON8(RTC 外部バックアップ)にバッテリー(対応電池: CR2032)を接続します。

5.7.1. RTC に時刻を設定する

Linux の時刻には、Linux カーネルが管理するシステムクロックと、RTC が管理するハードウェアクロックの 2 種類があります。RTC に時刻を設定するためには、まずシステムクロックを設定します。その後、ハードウェアクロックをシステムクロックと一致させる手順となります。

システムクロックは、date コマンドを用いて設定します。date コマンドの引数には、設定する時刻を [MMDDhhmmCCYY.ss] というフォーマットで指定します。時刻フォーマットの各フィールドの意味を次に示します。

表 5.11 時刻フォーマットのフィールド

フィールド	意味
MM	月
DD	日(月内通算)
hh	時
mm	分
CC	年の最初の 2 桁(省略可)
YY	年の最後の 2 桁(省略可)
ss	秒(省略可)

2013 年 1 月 23 日 4 時 56 分 00 秒に設定する例を次に示します。

```
[armadillo ~]# date ❶
Sat Jan 1 09:00:00 JST 2000
[armadillo ~]# date 012304562013.00 ❷
Wed Jan 23 04:56:00 JST 2013
[armadillo ~]# date ❸
Wed Jan 23 04:56:00 JST 2013
```

- ❶ 現在のシステムクロックを表示します。
- ❷ システムクロックを設定します。
- ❸ システムクロックが正しく設定されていることを確認します。

図 5.31 システムクロックを設定

システムクロックを設定後、ハードウェアクロックを hwclock コマンドを用いて設定します。

```
[armadillo ~]# hwclock ❶
Sat Jan 1 00:00:00 2000 0.000000 seconds
[armadillo ~]# hwclock --utc --systohc ❷
[armadillo ~]# hwclock --utc ❸
Wed Jan 23 04:56:10 2013 0.000000 seconds
```

- ❶ 現在のハードウェアクロックを表示します。
- ❷ ハードウェアクロックを協定世界時(UTC)で設定します。
- ❸ ハードウェアクロックが UTC で正しく設定されていることを確認します。

図 5.32 ハードウェアクロックを設定

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. ユーザー登録

アットマークテクノ製品をご利用のユーザーに対して、購入者向けの限定公開データの提供や大切なお知らせをお届けするサービスなど、ユーザー登録すると様々なサービスを受けることができます。サービスを受けるためには、「アットマークテクノ ユーザーズサイト」にユーザー登録をする必要があります。

ユーザー登録すると次のようなサービスを受けることができます。

- ・ 製品使用や部品などの変更通知の閲覧・配信
- ・ 購入者向けの限定公開データのダウンロード
- ・ 該当製品のバージョンアップに伴う優待販売のお知らせ配信
- ・ 該当製品に関する開発セミナーやイベント等のお知らせ配信

詳しくは、「アットマークテクノ ユーザーズサイト」をご覧ください。

アットマークテクノ ユーザーズサイト

<https://users.atmark-techno.com/>

7.1. 購入製品登録

ユーザー登録完了後に、購入製品登録することで、「購入者向けの限定公開データ^[1]」をダウンロードすることができるようになります。

Armadillo-810 購入製品登録

<https://users.atmark-techno.com/armadillo-810/register>

Armadillo-810 の購入製品登録を行うには、Armadillo-810 から取り出した「正規認証ファイル」をアットマークテクノ ユーザーズサイトからアップロードする必要があります。次のように Armadillo-810 から正規認証ファイル(board-info.txt)を取り出してください。

手順 7.1 Armadillo-810 から正規認証ファイルを取り出す

1. ATDE で minicom を立ち上げて、Armadillo-810 にログインします。

```
[ATDE ~]$ LANG=C minicom --noinit --wrap --device /dev/ttyUSB0

armadillo810-0 login: root
Password:
[root@armadillo810-0 (ttySC2) ~]#
```

2. "get-board-info-a810"コマンドを実行して正規認証ファイル(board-info.txt)を作成します。

[1] 拡張ボードの回路図データや、ミドルウェアパッケージなど

```
[root@armadillo810-0 (ttySC2) ~]# get-board-info-a810
[root@armadillo810-0 (ttySC2) ~]# ls
board-info.txt
[root@armadillo810-0 (ttySC2) ~]#
```

3. "lsz"コマンドで正規認証ファイルを ATDE に転送します。"press any key to continue..."と表示されてから任意のキーを入力します。

```
[root@armadillo810-0 (ttySC2) ~]# lsz board-info.txt

+-----[zmodem download - Press CTRL-C to quit]-----+
|Receiving: board-info.txt
|Bytes received: 775/ 775 BPS:10318
|Transfer complete
| READY: press any key to continue...
+-----+

[root@armadillo810-0 (ttySC2) ~]#
```

4. minicom を終了させると ATDE に正規認証ファイルが転送されています。

```
[ATDE ~]$ ls board-info.txt
board-info.txt
```

取り出した正規認証ファイルを「Armadillo-810 購入製品登録」ページの「正規認証ファイル」欄に指定し、アップロードしてください。

改訂履歴

バージョン	年月日	改訂内容
1.0.0	2013/02/14	・ 初版発行
1.0.1	2013/02/19	・ 「開発用 USB シリアル変換アダプタの取扱い上の注意」を追加

Armadillo-810 カメラモデル開発セットスタートアップガイド
Version 1.0.1
2013/02/19

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 TEL 011-207-6550 FAX 011-207-6570
