

AVR4903 : ASF – USB装置HIDマウス応用

要点

- USB 2.0適合
 - ・ USB第9章保証
 - ・ HID適合
 - ・ 低速(Low,1.5Mビット/秒)、全速(Full,12Mビット/秒)のデータ速度
- 標準USB HIDマウス実装
 - ・ XY移動
 - ・ 左、右、中央釦
 - ・ 輪スクロール
- 遠隔起動支援
- USBバス給電支援
- 実時間(OS適合、割り込み駆動)
- 8ビットと32ビットのATMEL® AVR® MCUを支援

1. 序説

この資料の狙いは新規または既存プロジェクトでUSBマウス装置応用を統合する簡単な方法を提供することです。

図1-1. USB装置HIDマウス応用



2. 略語

- ・ ASF : AVRソフトウェア枠組み(AVR Software Framework)
- ・ CD : 複合装置(複数のインターフェースを持つUSB装置)(Composite device)
- ・ FS : USB全速(Full Speed)
- ・ HID : 対人インターフェース装置(Human interface device)
- ・ HS : USB高速(High Speed)
- ・ LS : USB低速(Low Speed)
- ・ UDC : USB装置制御部(USB Device Controller)
- ・ UDD : USB装置記述子(USB Device Descriptor)
- ・ UDI : USB装置インターフェース(USB Device Interface)
- ・ USB : 万能直列バス(Universal Serial Bus)
- ・ SOF : フレーム開始(Start of frame)
- ・ ZLP : 0長パケット(Zero length packet)



ATMEL

マイクロコントローラ

応用記述

本書は一般の方々の便宜のため有志により作成されたもので、ATMEL社とは無関係であることを御承知ください。しおりのはじめにでの内容にご注意ください。

Rev. 8409A-08/11, 8409AJ1-03/14

3. 概要

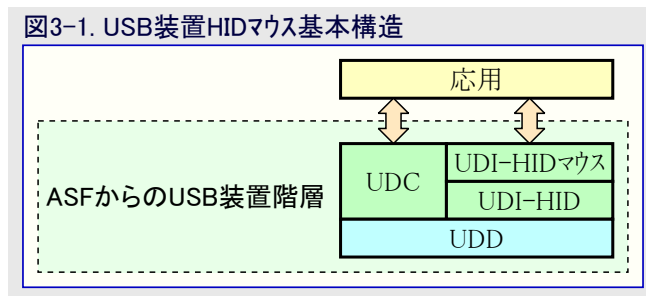
この資料はUSB装置HIDマウス応用を構築する時の必要条件の全ての形式に対する以下の4つの部分を含みます。

- **即時開始**
使用準備済みのHIDマウス装置例を開始する方法を記述します。
- **例の説明**
HIDマウス装置例を説明します。
- **USB装置マウスの構築**
プロジェクトにUSB装置マウスを追加する方法を記述します。
- **USB複合装置でのマウス**
複合装置プロジェクトにUSB装置マウスを統合する方法を記述します。

これらの部分の全てに対して、以下のようなHIDマウス応用の主な単位部構成を知ることが推奨されます。

- 使用者応用
- USB装置インターフェースHIDマウス (UDI-HIDマウス)
- USB装置インターフェースHID (UDI-HID)
- USB装置制御部 (UDC)
- USB装置ドライバ (UDD)

USB階層実装に関するもっと高度な情報については、「AVR4900:ASF-USB装置階層」応用記述を参照してください。



注: USB装置階層は`common¥services¥usb`ディレクトリに於いてASFで利用可能です。

4. 即時開始

USB装置マウス例はATMEL® AVR Studio® 5で利用可能です。ここでの例はマウスとU-ディスクを支援する複合装置を提供します。

AVR Studio 5は新規例プロジェクト(New Example Project)の作成を許します。例一覧に於いて、使用するATMEL基板に対応するUSB装置HIDマウス(USB device HID mouse example)を選択してください。

プロジェクトはどんな変更も必要なく、コンパイル、書き込み設定、走行だけがが必要です。

USB装置HIDマウス例は使用するハードウェアに依存して、外部電源と共に、またはなし(USBケーブル接続だけ)で走行できます。

4.1. 使用者インターフェース

使用者インターフェースはハードウェア上で利用可能なスイッチとLEDに依存します。

利用可能な2つのLEDでは次の通りです。

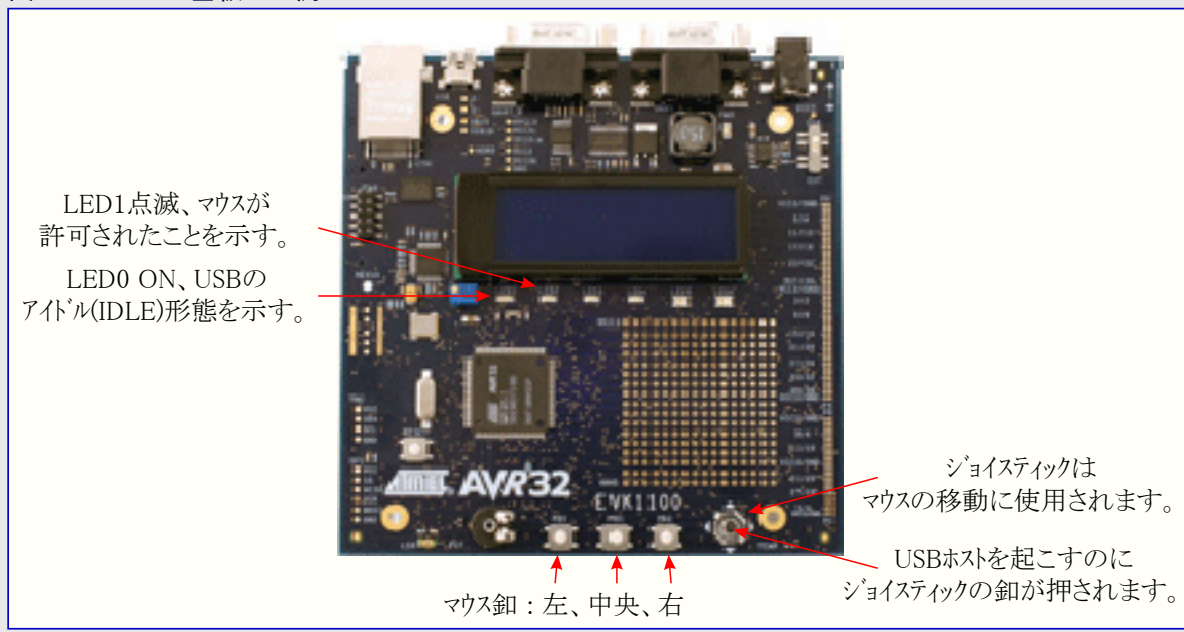
- LED0はUSB装置がアイドル形態の時にON、休止(サスペンド)形態でOFFです。
- LED1はマウス インターフェースがUSBホストによって許可されていることを示すのに点滅します。

スイッチまたは感知器を用いて以下の活動を行うことができます。

- XとY軸でのポインタ移動
- スクロール輪移動
- 左、右、中央の釦クリック
- 遠隔起動

使用者インターフェース記述(基板特有)は`ui.c`ソースファイルの最後で定義されます。このファイルはプロジェクトフォルダ内で利用可能です。

図4-1. EVK1100基板での例



5. 例の説明

ASFは様々なATMEL AVR製品に対してUSB装置マウス例を提供します。

これら全ての例は共通ファイルを共有します。

ファイルの説明と実行の動きが本章内の以下で記述されます。

5.1. 例の内容

表5-1はUSB装置HIDマウス例内に含まれる主なファイルの要約を紹介します。これらのファイルは図3-1. USB装置HIDマウス基本構造で記述される単位部と連携します。

表5-1. USB装置HIDマウス例のファイル

単位部	ファイル	ASFパス	説明
応用	main.c ui.c conf_usb.h	Exampleフォルダ*	主繰り返し マウス操作のためのハードウェア スイッチ構成設定 USB装置形態設定
UDI-HIDマウス	udi_hid_mouse.c/h udi_hid_mouse_desc.c udi_hid_mouse_conf.h	common¥services¥usb¥class¥hid¥ device¥mouse¥ common¥services¥usb¥class¥hid¥ device¥mouse¥	標準HIDマウス クラス実装 HIDマウス インターフェースを持つUSB装置用USB記述子 (USB複合装置に適用不可)
UDI-HID	udi_hid.c/h usb_protocol_hid.h	common¥services¥usb¥class¥hid¥ device¥ common¥services¥usb¥class¥hid	共通HIDクラス実装 HID規約定数
UDC	udc.c/h udc_desc.h udi.h udd.h usb_protocol.h usb_atmel.h	common¥services¥usb¥udc¥ common¥services¥usb¥	USB装置核 USB規約定数
UDD	usbb_device.c/h usbc_device.c/h usb_device.c/h	avr32¥drivers¥usbb¥ avr32¥drivers¥usbc¥ xmega¥drivers¥usb¥	USBドライバ*

5.2. 例の動き

main.cとui.cのファイルがHIDマウス応用のユーザーインターフェースを実装します。

実装は以下のように3つの段階から成ります。

1. USB装置を開始します。

```
udc_start();
udc_attach(); // USBケーブルが接続された時に呼ばれなければなりません。
               // 接続されたケーブルはVBus事象経由で検知されます。
```

2. 呼び戻し経路で許可されるべきHIDマウス インターフェースを待ちます。

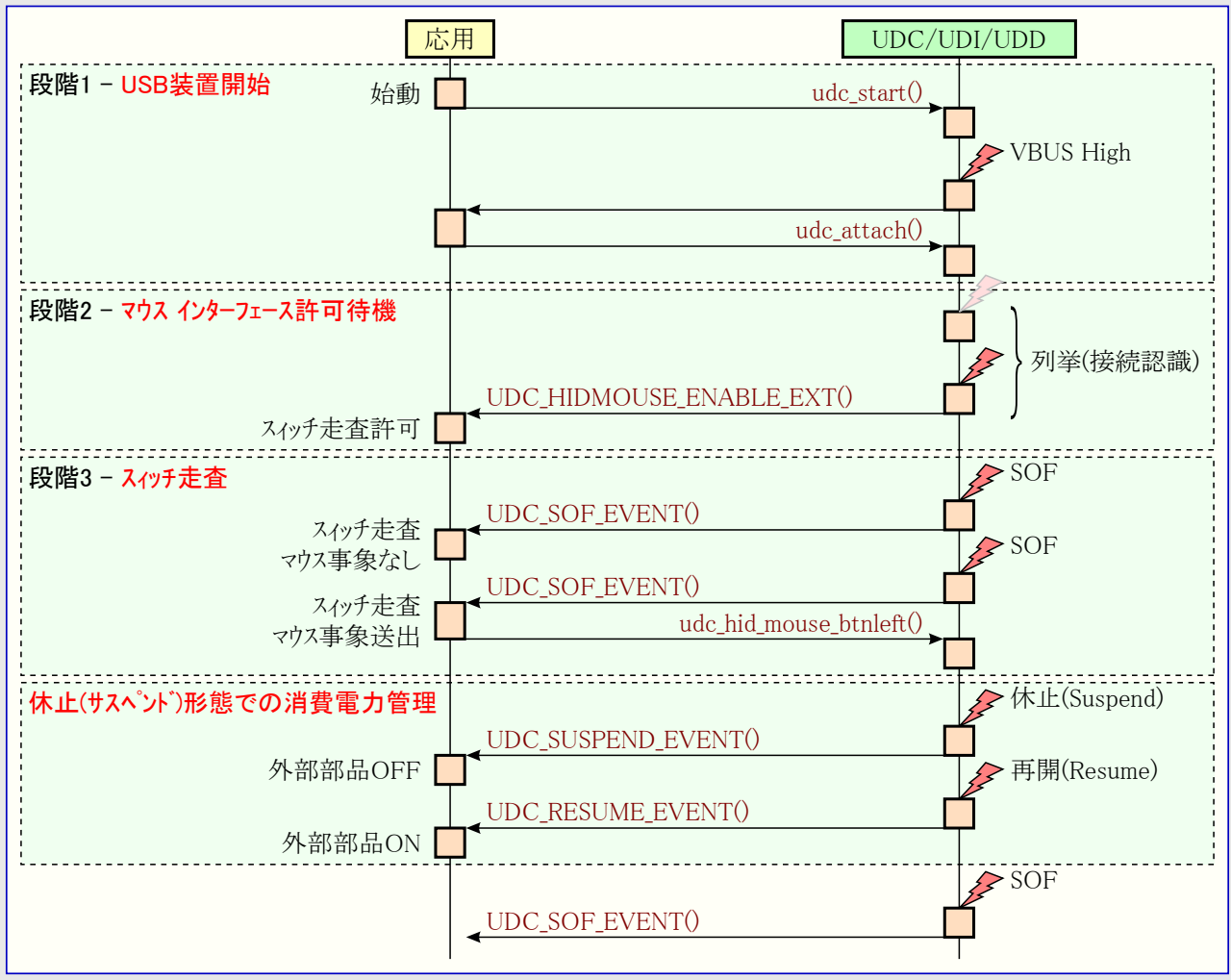
```
UDI_HIDMOUSE_ENABLE_EXT() // HIDマウス事象を認可します。
```

3. スイッチを走査してどれかが操作されている場合にマウス事象を送ります。

```
If (switch_left_down) udi_hid_mouse_btnleft(HID_MOUSE_BTN_DOWN);
```

助言: 実装を簡単化するため、1ms毎にスイッチを走査するのにSOF事象が使用されます。

図5-1. 例の動きの流れ



6. USB装置マウスの構築

USB装置マウス単位部はATMEL AVR Studio 5で利用可能で、AVR Studio 5プロジェクトに持ち込む(インポート)することができます。

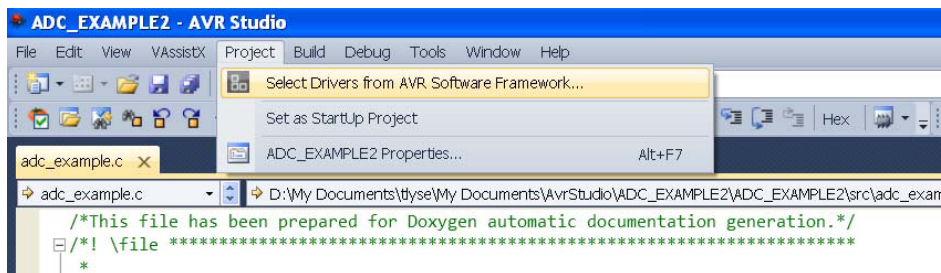
この項はプロジェクトにUSB装置マウスを追加する方法を記述します。

1. USBマウス単位部をインポートしてください。
2. 私的なUSBパラメータを形態設定してください。
3. USB装置を走らせるためにUSBルーチンを呼んでください。

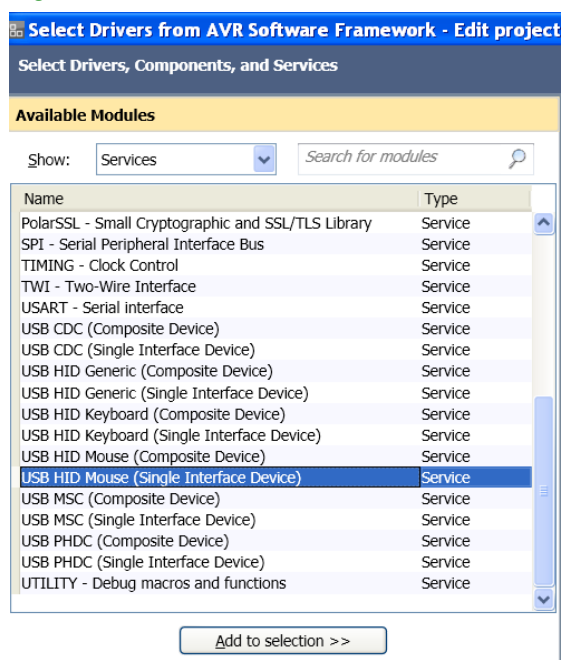
6.1. USB単位部のインポート

USB HIDマウス単位部をインポートするには、下の指示に従ってください。

1. あなたのプロジェクトを開く、または作成してください。
2. **Project**メニューから、“**Select Drivers from AVR Software Framework**”を選択してください。



3. **Services**を選んで、**USB HID Mouse (Single Interface Device)**を選択して“**Add to selection**”鈕を押してください。



6.2. USB形態設定

全てのUSB階層形態設定は応用単位部内の`conf.usb.h`ファイルに格納されます。これらの形態設定は簡単で、どんな特別なUSBの知識も必要ありません。

これらはUDC,UDI,UDDの各USB単位部に対して1つの形態設定項目があります。

UDC形態設定のありそうなことは「**AVR4900:ASF-USB装置階層**」応用記述の「**7.1.1. USB装置形態設定**」で記述されます。

UDD形態設定のありそうなことは「**AVR4900:ASF-USB装置階層**」応用記述の「**7.1.3. USBドライバ形態設定**」で記述されます。

マウス インターフェースのUDIはどの形態設定も必要ありません。

6.3. USB実装

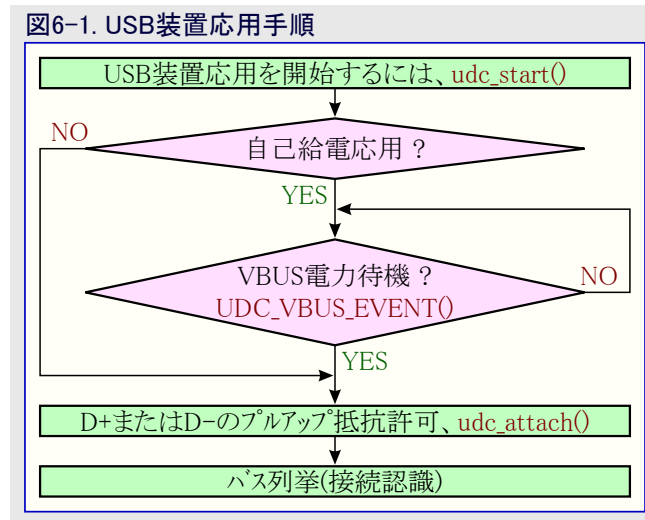
この項は追加してUSB装置マウス応用を走らせるためのソースコードを記述します。

実装は以下の3つの段階から成ります。

1. USB装置を開始します。
2. ホストによってHIDマウスが許可されるのを待ちます。
3. 感知器を走査して、何かあれば、USBマウス事象を送ります。

6.3.1. USB装置制御

USB装置応用を開始するには2つの関数呼び出しだけが必要とされます(図6-1をご覧ください)。



注: 新規プロジェクトの場合、USB階層は割り込みを許可してクロックと休止管理部(sleepmgr)のサービスを初期化することが必要です。
例:

```

<conf_usb.h>
#define UDC_VBUS_EVENT(b_vbus_high) ¥
    vbus_event(b_vbus_high)

<主Cファイル>:
main() {
    // 割り込み認可
    irq_initialize_vectors();
    cpu_irq_enable();
    // 休止管理部サービス初期化
    sleepmgr_init();
    // クロックサービス初期化
    sysclk_init();
    // USB階層装置許可
    udc_start();
}

vbus_event(b_vbus_high) {
    if (b_vbus_high) {
        // USB装置接続
        udc_attach();
    } else {
        // USB装置切断
        udc_detach();
    }
}
  
```

6.3.2. USBインターフェース制御

装置列挙(USB装置の検出と識別)後、USBホストは装置形態設定を開始します。装置からのUSBマウスインターフェースがホストによって受け入れられると、USBホストはこのインターフェースを許可してUDL_HIDMOUSE_ENABLE_EXT()呼び戻し関数が呼ばれます。

USB装置が切断、またはUSBホストによってリセットされると、USBインターフェースは禁止され、UDI_HIDMOUSE_DISABLE_EXT()呼び戻し関数が呼ばれます。

従って、これらの関数でマウスによって使用される感知器を許可/禁止することが推奨されます。

例:

```

<conf_usb.h>
#define UDI_HIDMOUSE_ENABLE_EXT() ¥
    mouse_enable()
#define UDI_HIDMOUSE_DISABLE_EXT() ¥
    mouse_disable()

<主Cファイル>:
mouse_enable() {
    // 感知器許可、感知器走査処理開始
    ~
    return true;
}
mouse_disable() {
    // 感知器禁止、感知器走査処理停止
    ~
}

```

6.3.3. USBマウス制御

表6-1.で記述されるUSB HIDマウス関数は応用にマウスの制御を許します。

表6-1. UDI HIDマウス関数

宣言	説明
udi_hid_mouse_moveScroll(int8_t pos)	スクロール輪での値を送信
udi_hid_mouse_moveY(int8_t pos_y)	マウスポインタでのY軸値を送信
udi_hid_mouse_moveX(int8_t pos_x)	マウスポインタでのX軸値を送信
udi_hid_mouse_btnmiddle(bool b_state)	"中央クリック"事象を送信
udi_hid_mouse_btnright(bool b_state)	"右クリック"事象を送信
udi_hid_mouse_btnleft(bool b_state)	"左クリック"事象を送信

例:

```

<感知部Cファイル>:
sensor_process() {
    if (is_sensor_move_up())
        udi_hid_mouse_moveY(sensor_get_move_value());
    if (is_button_left_press())
        udi_hid_mouse_btnleft(HID_MOUSE_BTN_DOWN);
    if (is_button_left_release())
        udi_hid_mouse_btnleft(HID_MOUSE_BTN_UP);
    ~
}

```

7. USB複合装置でのマウス

本章は「AVR4902:ASF-USB複合装置」応用記述と共にUSB複合装置を作成するためのHIDマウス情報を提供します。

7.1. USB形態設定

6.2.項で記述されたUSB形態設定に加えて、以下の値がconf_usb.hファイルで定義されなければなりません。

- USB_DEVICE_EP_CTRL_SIZE : エンドポイント制御容量、これは以下でなければなりません。
 - 低速(Low Speed)に対して8
 - 全速(Full Speed)装置に対して8,16,32または64 (RAMを節約するために8が推奨されます。)
 - 高速(High Speed)装置に対して64
- USB_HID_MOUSE_EP_IN : HIDマウス インターフェースによって使用されるINエンドポイント番号。
- USB_HID_MOUSE_IFACE_NUMBER : HIDマウス インターフェースのインターフェース番号。
- USB_DEVICE_MAX_EP : 応用での総エンドポイント数。これはHIDマウス インターフェース用の1つのエンドポイントを含めなければなりません。

7.2. USB記述子

conf_usb.hファイルで定義される複合装置のUSB装置記述子はHIDマウス インターフェース記述子を含めなければなりません。

```
//! 複合インターフェース記述子の構造体定義
#define UDI_COMPOSITE_DESC_T          ¥
    udi_hid_mouse_desc_t udi_hid_mouse;  ¥
    ~

//! 全速(Full Speed)用複合インターフェース記述子を満たす。
#define UDI_COMPOSITE_DESC_FS          ¥
    .udi_hid_mouse                      = UDI_HID_MOUSE_DESC, ¥
    ~

//! 高速(High Speed)用複合インターフェース記述子を満たす。
#define UDI_COMPOSITE_DESC_HS          ¥
    .udi_hid_mouse                      = UDI_HID_MOUSE_DESC, ¥
    ~

//! インターフェース記述子に対応するインターフェースAPIを満たす。
#define UDI_COMPOSITE_API              ¥
    &udi_api_hid_mouse,                 ¥
    ~
```

8. 目次

要点	1
1. 序説	1
2. 略語	1
3. 概要	2
4. 即時開始	2
4.1. 使用者インターフェース	2
5. 例の説明	3
5.1. 例の内容	3
5.2. 例の動き	4
6. USB装置マウスの構築	4
6.1. USB単位部のインポート	5
6.2. USB形態設定	5
6.3. USB実装	5
6.3.1. USB装置制御	6
6.3.2. USBインターフェース制御	6
6.3.3. USBマウス制御	7
7. USB複合装置でのマウス	7
7.1. USB形態設定	7
7.2. USB記述子	8
8. 目次	8



Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131
USA
TEL (+1)(408) 441-0311
FAX (+1)(408) 487-2600
www.atmel.com

Atmel Asia Limited

Unit 01-5 & 16, 19F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
HONG KONG
TEL (+852) 2245-6100
FAX (+852) 2722-1369

Atmel Munich GmbH

Business Campus
Parking 4
D-85748 Garching b. Munich
GERMANY
TEL (+49) 89-31970-0
FAX (+49) 89-3194621

Atmel Japan

141-0032 東京都品川区
大崎1-6-4
新大崎勸業ビル 16F
アトメル ジャパン合同会社
TEL (+81)(3)-6417-0300
FAX (+81)(3)-6417-0370

© 2011 Atmel Corporation. 全権利予約済

ATMEL®、ATMELロゴとそれらの組み合わせ、AVR®、AVR Studio®、それとその他はATMEL Corporationの登録商標または商標またはその付属物です。他の用語と製品名は一般的に他の商標です。

お断り: 本資料内の情報はATMEL製品と関連して提供されています。本資料またはATMEL製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。ATMELのウェブサイトに表示する販売の条件とATMELの定義での詳しい説明を除いて、商品性、特定目的に関する適合性、または適法性の暗黙保証に制限せず、ATMELはそれらを含むその製品に関連する暗示的、明示的または法令による如何なる保証も否認し、何ら責任がないと認識します。たとえATMELがそのような損害賠償の可能性を進言されたとしても、本資料を使用できない、または使用以外で発生する(情報の損失、事業中断、または利益と損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してATMELに責任がないでしょう。ATMELは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。ATMELはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、ATMEL製品は車載応用に対して適当ではなく、使用されるべきではありません。ATMEL製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

© HERO 2014.

本応用記述はATMELのAVR4903応用記述(doc8409.pdf Rev.8409A-08/11)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。