

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。

AVR[®] Prog

使用者の手引き



目次

1. 序説	3
<hr/>	
2. インストール	3
<hr/>	
3. Windows版AVRprog実行	4
3.1. デバイス選択	4
3.2. INTEL HEXファイル収納	5
3.3. デバイス書き込み	6
3.4. デバイス照合	7
3.5. デバイス読み戻し	7
3.6. 高度機能の使用	8
3.7. 施錠ビット書き込み	8
3.8. ヒューズビット書き込み	8
3.9. チップ消去卸	8
3.10. 状態部	9
<hr/>	
4. DOS版の実行	9
<hr/>	
5. 障害対策	9
5.1. 状態表示部	9
5.2. 何処で手助けを得るか	10
<hr/>	
付録	11





(訳注) 原書は古いものですが、操作部分の手引書として作成しました。従ってWindows版操作に関しては現状対応としていますが、基本的にその他の部分は原書に従っていますので、注意してください。また、DOS版プログラムは現在提供されていないため、対応する部分は基本的に省略しました。

1. 序説

この手引書はAVRprogと呼ばれるAVRプログラミング一括ソフトウェアの使用法を記述します。AVRprogはプログラミング機能を持つAtmelのAVRツールの全てを網羅します。

AVRprogはAT90S標準系列マイクロコントローラの全デバイスプログラミングできます。付加機能としてAtmelのAT89S系統マイクロコントローラの多くもプログラミングできます。

AVRプログラミング一括ソフトウェアは以下のプログラムから成ります。

- **AVRPROG.EXE** - Windows版
- **PROGF.EXE** - フラッシュメモリ書き込み用DOSプログラム
- **PROGE.EXE** - EEPROM書き込み用DOSプログラム
- **READF.EXE** - フラッシュメモリ読み出し用DOSプログラム
- **READE.EXE** - EEPROM読み出し用DOSプログラム
- **PROGFUSE.EXE** - ヒューズビット書き込み用DOSプログラム
- **PROGLOCK.EXE** - 施錠ビット書き込み用DOSプログラム

AVRprogはMicrosoft Windows下で走行します。加えてMS-DOSプログラム群全体でWindows版プログラムと殆ど同じ機能を持ちます。

AVRprogは現在、以下のツールとで使うことができます。

- AT90DEVBOARD - 第2世代開発基板
- AT90PROG1200 - AT90S1200プログラミング基板
- AT90ISP - AVR実装書き込み器

AVRprogは将来の以下のAVRツールでも動作します。

- AT90BASEPROG - AVR標準系列書き込み器

Windows版AVRprogはAVR Studioに含まれ、Toolsメニュー内で見つけることができます。

2. インストール

(訳注) 原書ではFD供給のAVRprogインストールが記述されていますが、AVRprogが現状独立した形で供給されていないので省略しました。現在はAVR Studioインストール時にWindows版AVRprog.exeが同時にインストールされます。これはインストールフォルダ下のAVRprogフォルダ内にあります。



3. Windows版AVRprog実行

AVRprogはAVRprog.exeのダブルクリックまたはAVR StudioのメニューバーからToolsメニューを選択して、AVRprogをクリックすることによって開始されます。

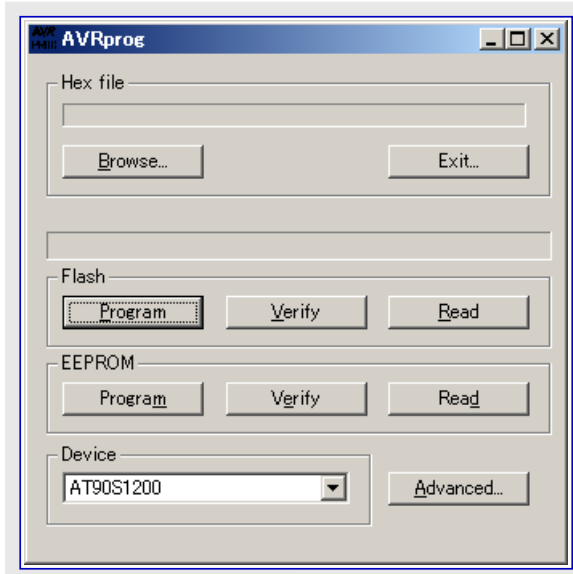
AVRprogの開始に先立って、プログラミング能力を持つAVRツールの1つが利用可能なシリアルポート(COM)に接続され、電源が入っている必要があります。有効なツールがどのシリアルポートにも見つからない場合、画面に右のメッセージが現れます。



AVRprogがどの支援基板もなぜ見つけれられないかは多くの理由があるかも知れません。これらは本手引書の最後の「障害対策」章で記述されます。

支援するAVRツールが電源ONで接続されているなら、画面に右のウィンドウが現れるでしょう。

このウィンドウは4つの部分に分けられます。上部はHex file欄と呼ばれ、AVRprogのファイル緩衝部内にIntel HEXファイルを収納するのに使われます。中間2つの部分はFlash欄(中間上)とEEPROM欄(中間下)と呼ばれます。これらは選択したデバイスのフラッシュメモリまたはEEPROMどちらかの書き込み、照合、読み出しに使われます。最後の部分はウィンドウ下部のDevice部です。これはプログラミングするデバイスの選択に使われるデバイス選択引き落とし一覧です。



Hex file欄とFlash欄の間は進捗バーと上図では見えない状態行があります。

ウィンドウ下側の右側にはAVRprogの高度機能を持つ新しいウィンドウを引き出す、Advancedと記された釦があります。これにはチップ消去、ヒューズビット書き込み、施錠ビット書き込み釦のような機能があります。このウィンドウは接続されたAVRツールに対するいくつかの情報も表示します。

3.1. デバイス選択

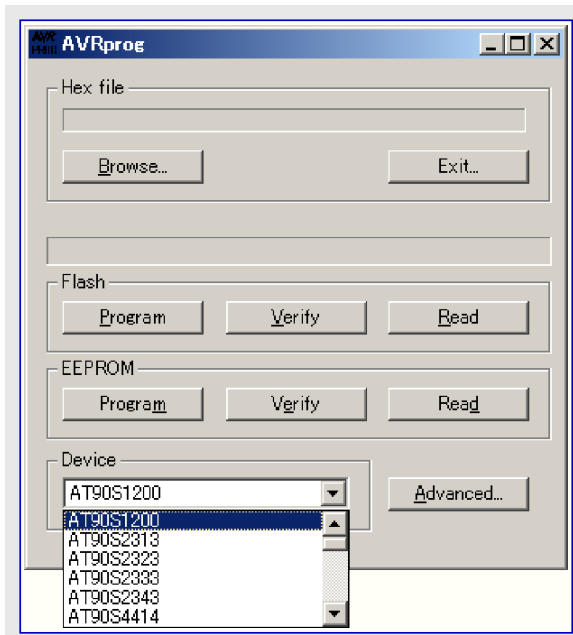
デバイスをプログラミングする最初の段階はデバイス選択引き落とし一覧からデバイスを選択することです。デバイス選択引き落とし一覧の内容は接続されたAVRツールのプログラミング能力に応じて変わります。AVRprogはシリアルポートに接続されたAVRツールを自動的に検知し、デバイス選択引き落とし一覧の内容をそのプログラミング能力に合わせて調整します。

AVR910書き込み器が接続されている場合、デバイス選択引き落とし一覧は右のように見えるでしょう。

AVR910は直列プログラミングだけが使えるので、直列プログラミング機能を持つデバイスだけが選択できます。

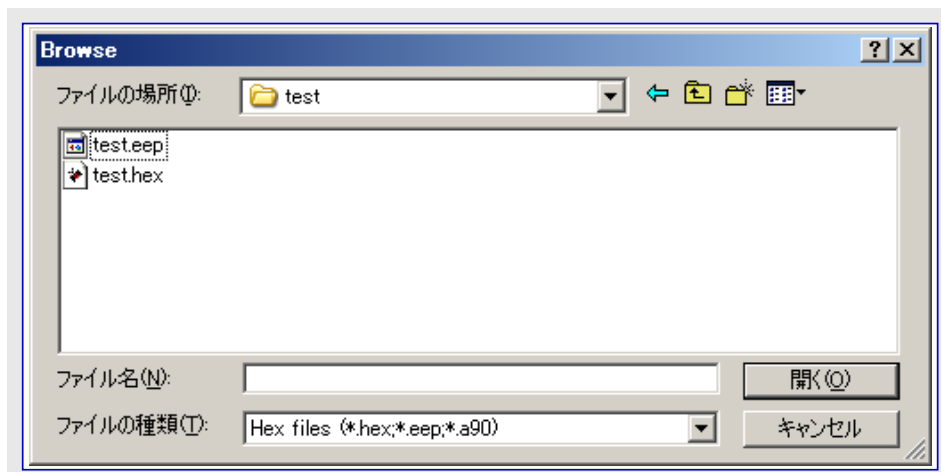
(訳注) 上記で原書のAT90DEVBOARDはAVR910書き込み器に変更しました。また原書の本位置に記述されているAT90PROG1200並列書き込み器の記述は省略しました。

マウス左釦のクリックによって、プログラミングしたいデバイスを選んでください。

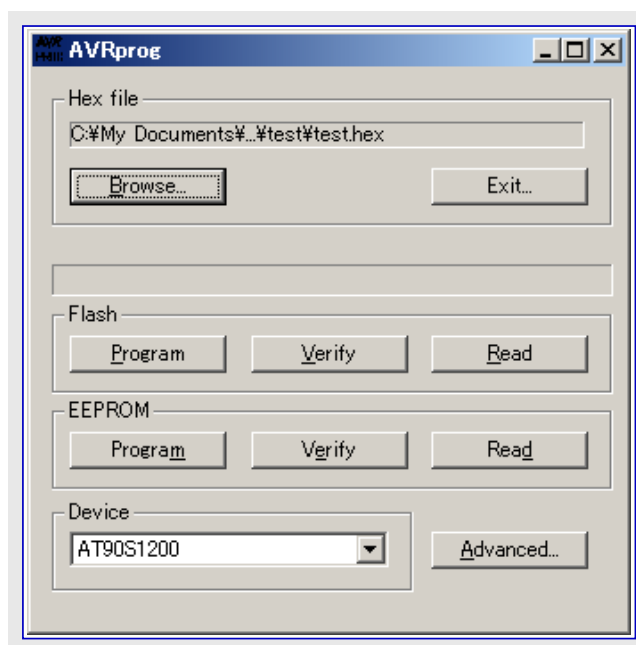


3.2. INTEL HEXファイル収納

次の段階は選択したデバイスのフラッシュメモリまたはEEPROM内に書き込むべきINTEL HEXファイル(内容)を収納することです。Browseダイアログを引き出すためにBrowse鈕をクリックしてください。画面は概ね次のように見えるでしょう。



HEXファイルが属するフォルダを位置するのに上部のファイルの場所引き落とし一覧を使い、それ(対象HEXファイル)のダブルクリックか、または選択と開く鈕の使用のどちらかによって1つを選択してください。ファイルが、HEX以外の拡張子の場合、対象の拡張子を変えるのにウィンドウ下部のファイルの種類引き落とし一覧を使ってください。ファイル(内容)が収納されてしまった後、AVRprogウィンドウは概ね次のように見えるでしょう。



(訳注) 原書本位置にはReload鈕記述がありますが、現在のAVRprogでは削除されてしまっていますので対応する記述を省略しました。

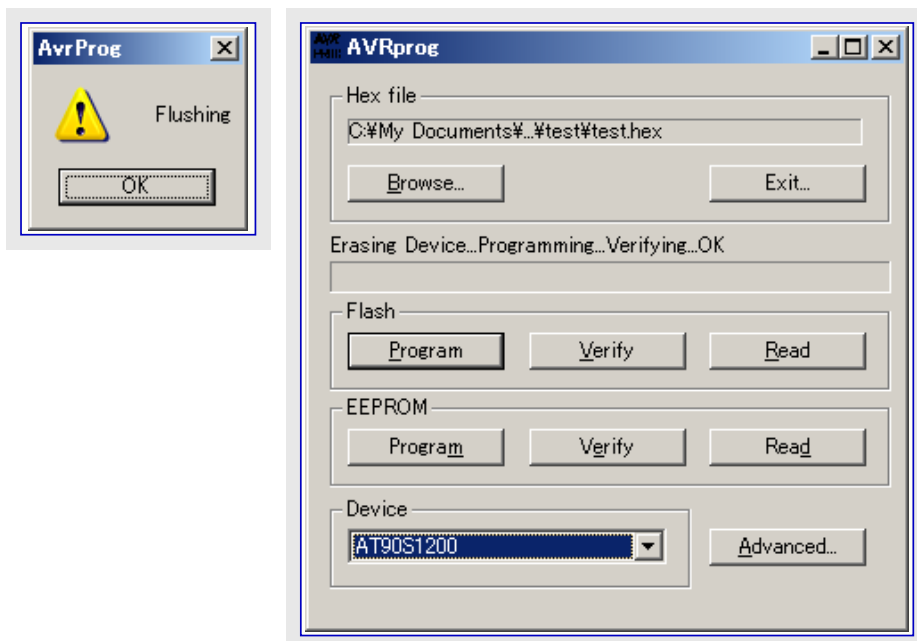
Browseダイアログが最後にファイルを収納したフォルダを記憶することに留意してください。同じフォルダから別のファイルが格納される場合、Browse鈕が押された時にそのフォルダが直ぐ利用可能です。

3.3. デバイス書き込み

プログラミング処理の最後の段階は収納したHEXファイル(内容)を選択したデバイス内に書き込むことです。**Flash**欄の**Program**鈕がクリックされると、開始されます。

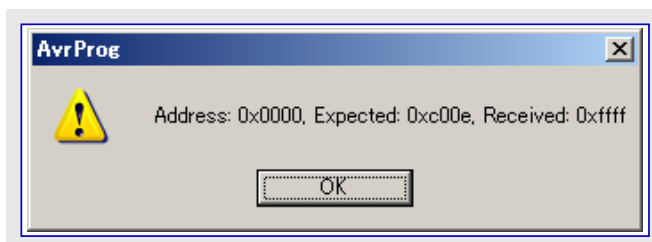
通信同期不調の場合、左下のメッセージボックスが現れます。ここで**OK**鈕をクリックすると、直前の残データを破棄し、書き込みが始まります。

収納したファイル(内容)が選択したデバイスのフラッシュメモリ内に書かれます。**AVRprog**ウィンドウは書き込み成功後、概ね右下のように見えます。



状態行が**Hex file**欄の真下に現れることに注意してください。これは最後に実行した動作の状態を一定時間表示し、その後消えます。書き込み中は進捗バーが有効です。

書き込みに失敗すると、次のようなメッセージボックスが現れます。



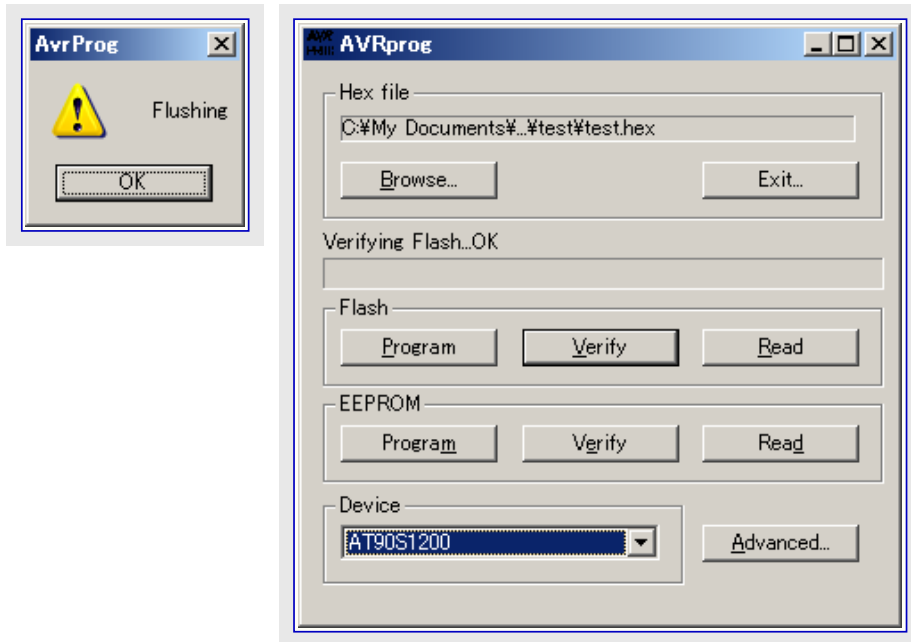
フラッシュメモリとEEPROMの両方が書かれる場合、フラッシュメモリが常に先で、EEPROMが後で実行されなければならないことに注意してください。これはフラッシュメモリ書き込みが常にチップ消去命令で始まり、EEPROM書き込みはそうでないからです。

3.4. デバイス照合

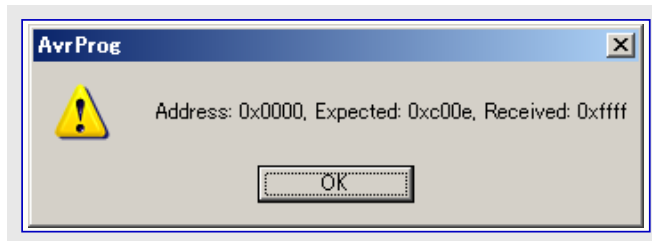
BrowseダイアログでAVRprogのファイル緩衝部に収納したHEXファイル(内容)に対してデバイス内容を照合するには、(Flash欄かEEPROM欄のどちらかの)Verify鈕を押してください。

通信同期不調の場合、左下のメッセージボックスが現れます。ここでOK鈕をクリックすると、直前の残データを破棄し、照合処理が始まります。

照合処理がOKなら、AVRprogウィンドウは概ね右下のように見えるでしょう。



照合処理が不正になったなら、次のようなメッセージボックスが現れます。



AVRprogウィンドウの状態行は消滅する前に、照合失敗(Verifying...Failed)を表示するでしょう。

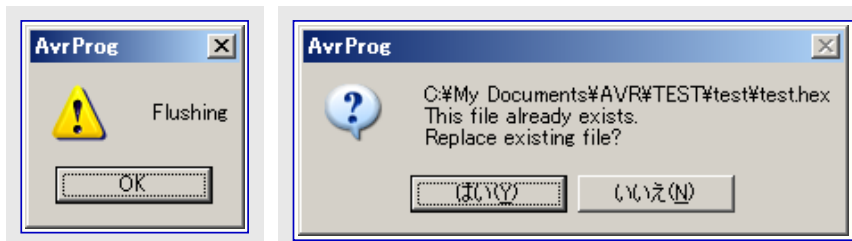
注: 照合失敗に対する最も共通する理由はデバイスの施錠ビットがプログラム(0)されてしまっていることです。これはデバイスに対する照合任意選択を禁止します。

3.5. デバイス読み戻し

デバイスの内容を読み戻すには(Flash欄かEEPROM欄のどちらかの)Read鈕を押してください。

通信同期不調の場合、左下のメッセージボックスが現れます。ここでOK鈕をクリックすると、直前の残データを破棄し、読み戻し処理が始まります。

この内容はHex file欄のファイル名に保存されます。ファイルが既存の場合、既存ファイルの置き換えを希望するかを問い合わせる警告メッセージボックスが表示されます。このメッセージボックスは概ね右下のように見えるでしょう。

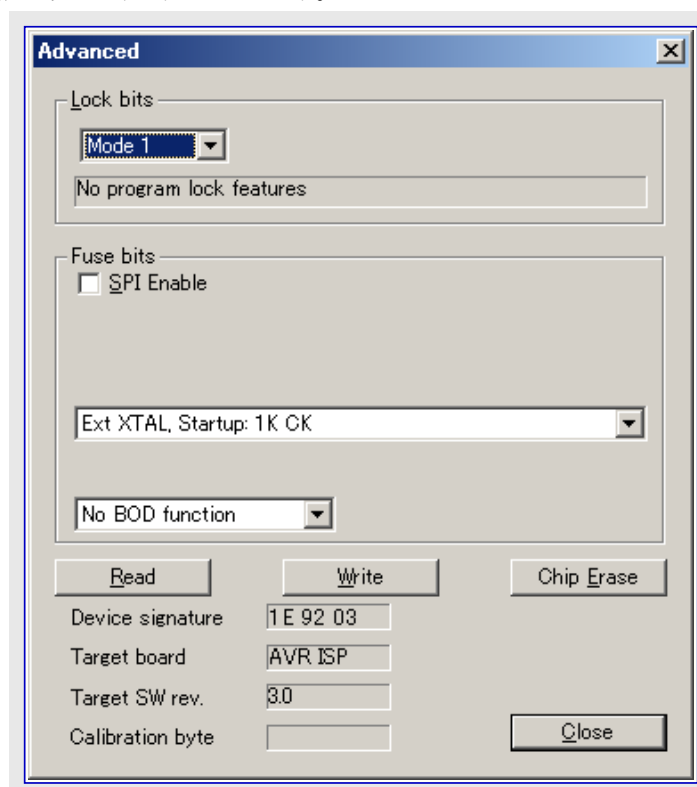


既存ファイルの置換を望まないなら、いいえをクリックし、Browseウィンドウを開いて新しい(別の)ファイル名を入力しなければなりません。現在位置とは別のフォルダのファイルへの書き出しを望む場合、ウィンドウ上部のファイルの場所引き落とし一覧を使うか、または完全なパスを含むファイル名を入力してください。



3.6. 高度機能の使用

AVRprogの高度機能を使うには、Advancedウィンドウを開くためにAdvanced鈕を押してください。このウィンドウは概ね次のように見えます。



上部は施錠ビットプログラミング用です。中間部はヒューズビットプログラミング用です。下部はデバイスと接続されたAVRツールについてのいくつかの状態情報を表示します。中間部にはチップ消去(Chip Erase)鈕もあります。これはデバイスプログラミング中、意図せず使われないよう、このウィンドウ内に配置されています。

3.7. 施錠ビット書き込み

施錠ビットを書くには希望する種別を選択し、Write鈕をクリックしてください。種別選択引き落とし一覧直下には施錠ビット種別の短い説明の状態行があります。施錠ビット種別に対する更なる説明については、それぞれのデバイスのデータシートをご覧ください。

施錠ビットを解除(1)する方法がチップ消去だけなことに注意してください。これはChip Erase鈕または、デバイスのフラッシュメモリの書き換えのどちらかによって行なえます。

AT90S1200の初期版(改訂A~C)に於いて、施錠ビットが直列プログラミング動作で書けないことに注意してください。これらのデバイスのどれかが選択された場合、Advancedウィンドウが開かれる時に施錠ビット選択が活性(有効)になりません。

3.8. ヒューズビット書き込み

ヒューズビットを読み書きするには、デバイスによって並列書き込み器が必要とされます。これらのデバイスでは、例えばAVRISPが接続されている場合、ヒューズビット選択が活性(有効)になりません。これはそのツールが直列プログラミングだけを実行し得るからです。

Advancedウィンドウ移行時、現在のヒューズビットの状態がデバイスから読まれ、ヒューズビット選択枠(Fuse bits欄)内に表示されます。強制的に状態ビットを読み出すには、Fuse bits欄下のRead鈕を使ってください。状態ビットを書くには希望する組み合わせを選択し、Read鈕傍のWrite鈕をクリックしてください。施錠ビットが先行して設定されている場合を除いて、施錠ビットも同時に書かれることに注意してください。ヒューズビットがチップ消去によって影響を及ぼされないことにも注意してください。これら(ヒューズビット)を変更する方法はWrite鈕で書き換えることだけです。

3.9. チップ消去鈕

この鈕がクリックされると、デバイスが消去されます。これはフラッシュメモリとEEPROMの内容を消去して同時に施錠ビットが解除(1)されます。ヒューズビットは本命令によって影響を及ぼされません。

3.10. 状態部

状態部の上部領域は現状デバイスの識票バイトを表示します。識票バイトの説明については、それぞれのデバイスのデータシートをご覧ください。

施錠ビットが設定(0)されていると、デバイスによって直列プログラミング動作で識票バイトが読み出せないことに注意してください。冷え列プログラミング動作が使われる場合、識票バイトは全ての施錠ビット種別に対して常に読むことができます。

状態部で現在どのAVRツールが接続されているかを表示します。これは以下などです。

AVR DEV – AT90DEVBOARD用表記

AVR PPR – AT90PROG1200基板用表記

AVR ISP – AT90ISP基板またはAVR910用表記

AVR BASE – AT90BASEPROG基板用表記

4. DOS版の実行

(訳注) 原書に於ける本節は現在供給されていないため省略しました。

5. 障害対策

できる限り異常のないプログラムを作るように心がけていますが、未だプログラムが解決できないいくつかの状態があります。以降の一覧は起こるかもしれない状況のいくつかを示します。AVRprogは明確な注記を除いて、Windows版とMS-DOS版の両方に対して使われます。

起動失敗

AVRprogが開始せず、“No supported boards found!”(支援基板が見つからない)メッセージが現れたなら、次の一覧を調べてください。

- ・対象基板に電源が印加されているか?
- ・対象の電源スイッチがONか? (注: 対象の電源LEDが点灯しているかを調べてください。)
- ・シリアルケーブルの両端が接続されているか?
- ・直通形式シリアルケーブルか? (注: 直通形式だけで動作し、逆接形式では動作しません。)
- ・複数のプログラムがシリアルポートにアクセスしていませんか?

Windows環境での共通する問題は2つのプログラムが同時に同じシリアルポートを制御できないことです。DOS窓が開いてシリアルポートを使うプログラムが走行してしまった場合(例えそのプログラムが終了されてしまっても)、Windows版のAVRprogは動作しません。Windows版のAVRprogが開いたままで、DOS版がDOS窓で走ると、同じ状態が起こります。解決策はシリアルポートを使ったかもしれない、または使うかもしれない、全てのプログラム(とDOS窓)を(完全に)閉じることです。これが問題を解決するでしょう。

5.1. 状態表示部

AVRprogが“Programming failed!”(書き込み失敗)または“Verified failed!”(照合失敗)のどちらかのメッセージを表示する場合、以下の一覧を調べてください。

- ・対象プログラミングソケットにデバイスが装着されているか?
- ・対象プログラミングソケットにデバイスが適切に装着されているか?
- ・デバイスの全ピンが対象プログラミングソケットの内側か?
- ・対象プログラミングソケットに正しいデバイスが装着されているか?
デバイスが対象基板のプログラミングソケットに完全に挿入されていることを調べてください。
- ・デバイス選択で正しいデバイスが選択されているか?

注: 不正なデバイス選択でデバイス書き込みを試みると、対象基板のソフトウェアが暴走し、総体として固まります。AT90DEVBOARDでは黄LEDの常時点灯が見えるかもしれません。AT90PROG1200基板では緑LEDが常時点灯するかもしれません。これが起きた場合、対象基板上の各々のリセット釦を使い、再び試みてください。これが助けにならないなら、対象基板への電源をOFFし、AVRprogプログラムを閉じてください。DOS版がWindows環境のDOS窓で使われている場合、そのDOS窓を閉じてください。その後、対象基板への電源を再びONにし、AVRprogを再び開始してください。これがこの状況を解決するでしょう。

注: デバイスは悪いESD環境や悪い取り扱いのため、開発で損傷を受けるかもしれません。常に正しいESD予防策でデバイスを取り扱ってください。

不良デバイスと思ったなら、他の多くのデバイスが同じ動作となるかの調査を試みてください。同じ動作でなければ、不良デバイスでないかもしれません。同じ動作ならば、対象基板が損傷しているかもしれません。更に対象基板は正しいESD予防処置で取り扱うべきです。入力電力が対象基板の供給範囲を超えはいけません。これは基板への恒久的な損傷を引き起こすかもしれません。対象基板の短絡を避けることにも十分に注意してください。これも対象基板に恒久的な損傷を引き起こすかもしれません。

(訳注) 原書に於けるDOS版固有の記述は、現在DOS版が供給されていないために省略しました。



5.2 何処で手助けを得るか

AVRprogやAVR対象基板で何らかの問題がある場合、最寄のAtmel代理店に手助けに関してお問い合わせください。avr@atmel.comでAVR支援係にEメールを送ることもできます。使った対象基板、AVRprogソフトウェア版番号、問題についての全情報を含めてください。使っているパソコンと、どのオペレーティングシステムを使っているか(Windowsの種類)の情報も含めてください。





付録 各AVRデバイスに於けるプログラミング関係情報とAVRprog 1.40版での支援情報の一覧を以下に示します。

メモリ容量はバイト単位です。プログラミング方式の略号はLS=低電圧直列、HS=高電圧直列、HP=高電圧並列、BL=ブートローダを示します。(BL)は専用の保護された領域はありませんが、制限付きでブートローダが可能であることを示します。

基本的にAVRprogは実デバイスの確認を行いませんので、書き込み手順に相違がある初期の一部デバイスを除いて、ページを含めたメモリ容量が一致さえすれば、どのデバイス選択でも書くことが可能です。ロック/ヒューズビットについては直列プログラミング命令形式が一致さえすれば、ウィンドウ表示は異なりますが、目的デバイスでの対応位置を指定することによってメモリと同様に書くことができます。

表A. デバイス関連仕様とAVRprog支援関係

デバイス名	識別符号	識票バイト	フラッシュメモリ容量 (ページ)	EEPROM容量 (ページ)	プログラミング方式	備考
AT89C1051	\$80	\$1E,\$12	1K (-)	- (-)	LS	
AT89C2051	\$81	\$1E,\$21	2K (-)	- (-)	LS	
AT89S8252	\$86	\$1E,\$72	8K (-)	2K (-)	LS	
AT89S53	\$87	\$1E,\$53	12K (-)	- (-)	LS	
AT90S1200	\$10	\$1E,\$90,\$01	1K (-)	64 (-)	LS,HP	改訂A
	\$11					改訂B
	\$12					改訂C
	\$13					改訂D
AT90S2313	\$20	\$1E,\$91,\$01	2K (-)	128 (-)	LS,HP	
AT90S2323	\$48	\$1E,\$91,\$02	2K (-)	128 (-)	LS,HS	
AT90S2343	\$4C	\$1E,\$91,\$03	2K (-)	128 (-)	LS,HS	
AT90S2333	\$34	\$1E,\$91,\$05	2K (-)	128 (-)	LS,HP	
AT90S4414	\$28	\$1E,\$92,\$01	4K (-)	256 (-)	LS,HP	
AT90S4434	\$6C	\$1E,\$92,\$02	4K (-)	256 (-)	LS,HP	
AT90S4433	\$30	\$1E,\$92,\$03	4K (-)	256 (-)	LS,HP	
AT90S8515	\$38	\$1E,\$93,\$01	8K (-)	512 (-)	LS,HP	
AT90S8535	\$68	\$1E,\$93,\$03	8K (-)	512 (-)	LS,HP	
AT90C8535	\$70	\$1E,\$93,\$04	8K (-)	512 (-)	HP	
AT90C8544	\$71	\$1E,\$93,\$0?	8K (-)	512 (-)	HP	
AT90CAN32	\$--	\$1E,\$95,\$81	32K (256)	1K (8)	LS,HP,JTAG,BL	
AT90CAN64	\$--	\$1E,\$96,\$81	64K (256)	2K (8)	LS,HP,JTAG,BL	
AT90CAN128	\$--	\$1E,\$97,\$81	128K (256)	4K (8)	LS,HP,JTAG,BL	
AT90PWM2/3	\$--	\$1E,\$93,\$81	8K (64)	512 (-)	LS,HP,dW,BL	
AT90PWM2B/3B	\$--	\$1E,\$93,\$83	8K (64)	512 (-)	LS,HP,dW,BL	
AT90PWM1	\$--	\$1E,\$93,\$83	8K (64)	512 (-)	LS,HP,dW,BL	
AT90USB64x	\$--	\$1E,\$96,\$8?	64K (256)	2K (8)	LS,HP,JTAG,BL	
AT90USB128x	\$--	\$1E,\$97,\$82	128K (256)	4K (8)	LS,HP,JTAG,BL	
ATtiny19	\$58	\$1E,\$90,\$0?	1K (-)	- (-)	LS,HS	
ATtiny10	\$51	\$1E,\$90,\$03	1K (-)	- (-)	HS	
ATtiny11	\$50	\$1E,\$90,\$04	1K (-)	- (-)	HS	
ATtiny12	\$55	\$1E,\$90,\$05	1K (-)	64 (-)	LS,HS	
ATtiny15	\$56	\$1E,\$90,\$06	1K (-)	64 (-)	LS,HS	
ATtiny13	\$--	\$1E,\$90,\$07	1K (32)	64 (4)	LS,HS,dW,(BL)	
ATtiny22	\$--	\$1E,\$91,\$06	2K (-)	128 (-)	LS,HS	
ATtiny28	\$5C	\$1E,\$91,\$07	2K (-)	- (-)	HS	
ATtiny25	\$--	\$1E,\$91,\$08	2K (32)	128 (4)	LS,HS,dW,(BL)	
ATtiny26	\$5E	\$1E,\$91,\$09	2K (32)	128 (4)	LS,HP	
ATtiny2313	\$--	\$1E,\$91,\$0A	2K (32)	128 (4)	LS,HP,dW,(BL)	
ATtiny24	\$--	\$1E,\$91,\$0B	2K (32)	128 (4)	LS,HS,dW,(BL)	
ATtiny261	\$--	\$1E,\$91,\$0C	2K (32)	128 (4)	LS,HS,dW,(BL)	
ATtiny45	\$--	\$1E,\$92,\$06	4K (64)	256 (4)	LS,HS,dW,(BL)	
ATtiny44	\$--	\$1E,\$92,\$07	4K (64)	256 (4)	LS,HS,dW,(BL)	
ATtiny461	\$--	\$1E,\$92,\$08	4K (64)	256 (4)	LS,HS,dW,(BL)	
ATtiny85	\$--	\$1E,\$93,\$0B	8K (64)	512 (4)	LS,HS,dW,(BL)	
ATtiny84	\$--	\$1E,\$93,\$0C	8K (64)	512 (4)	LS,HS,dW,(BL)	
ATtiny861	\$--	\$1E,\$93,\$0D	8K (64)	512 (4)	LS,HS,dW,(BL)	

[次頁へ続く](#)



表A(続き). デバイス関連仕様とAVRprog支援関係

デバイス名	識別符号	識票バイト	フラッシュメモリ 容量 (ページ)	EEPROM 容量 (ページ)	プログラミング 方式	備考
ATmega48(P)	\$--	\$1E,\$92,\$05	4K (64)	256 (4)	LS,HP,dW,(BL)	
ATmega83	\$65	\$1E,\$93,\$05	8K (64)	512 (-)	LS,HP,BL	ブートローダ
	\$67					
ATmega8515	\$3A	\$1E,\$93,\$06	8K (64)	512 (4)	LS,HP,BL	ブートローダ
	\$3B					
ATmega8	\$76	\$1E,\$93,\$07	8K (64)	512 (4)	LS,HP,BL	ブートローダ
	\$77					
ATmega8535	\$69	\$1E,\$93,\$08	8K (64)	512 (4)	LS,HP,BL	
ATmega88(P)	\$--	\$1E,\$93,\$0A	8K (64)	512 (4)	LS,HP,dW,BL	
ATmega161	\$60	\$1E,\$94,\$01	16K (128)	512 (-)	LS,HS,BL	ブートローダ
	\$61					
ATmega163	\$64	\$1E,\$94,\$02	16K (128)	512 (-)	LS,HS,BL	ブートローダ
	\$66					
ATmega16	\$74	\$1E,\$94,\$03	16K (128)	512 (4)	LS,HP,JTAG,BL	ブートローダ
	\$75					
ATmega162	\$--	\$1E,\$94,\$04	16K (128)	512 (4)	LS,HP,JTAG,BL	
ATmega169(P)	\$78	\$1E,\$94,\$05	16K (128)	512 (4)	LS,HP,JTAG,BL	ブートローダ
	\$79					
ATmega168(P)	\$--	\$1E,\$94,\$06	16K (128)	512 (4)	LS,HP,dW,BL	
ATmega165(P)	\$--	\$1E,\$94,\$07	16K (128)	512 (4)	LS,HP,JTAG,BL	
ATmega164P	\$--	\$1E,\$94,\$0A	16K (128)	512 (4)	LS,HP,JTAG,BL	
ATmega323	\$--	\$1E,\$95,\$01	32K (128)	1K (-)	LS,HP,JTAG,BL	
ATmega32	\$72	\$1E,\$95,\$02	32K (128)	1K (4)	LS,HP,JTAG,BL	ブートローダ
	\$73					
ATmega329(P)	\$--	\$1E,\$95,\$03	32K (128)	1K (4)	LS,HP,JTAG,BL	
ATmega3290(P)	\$--	\$1E,\$95,\$04	32K (128)	1K (4)	LS,HP,JTAG,BL	
ATmega325(P)	\$--	\$1E,\$95,\$05	32K (128)	1K (4)	LS,HP,JTAG,BL	
ATmega3250	\$--	\$1E,\$95,\$06	32K (128)	1K (4)	LS,HP,JTAG,BL	
ATmega406	\$--	\$1E,\$95,\$07	40K (128)	512 (4)	HP,JTAG,BL	
ATmega324P	\$--	\$1E,\$95,\$08	32K (128)	1K (4)	LS,HP,JTAG,BL	
ATmega604	\$42	\$1E,\$96,\$01	64K (256)	2K (-)	LS,HP	
ATmega64	\$45	\$1E,\$96,\$02	64K (256)	2K (8)	LS,HP,JTAG,BL	ブートローダ
	\$46					
ATmega649(P)	\$--	\$1E,\$96,\$03	64K (256)	2K (8)	LS,HP,JTAG,BL	
ATmega6490(P)	\$--	\$1E,\$96,\$04	64K (256)	2K (8)	LS,HP,JTAG,BL	
ATmega645(P)	\$--	\$1E,\$96,\$05	64K (256)	2K (8)	LS,HP,JTAG,BL	
ATmega6450(P)	\$--	\$1E,\$96,\$06	64K (256)	2K (8)	LS,HP,JTAG,BL	
ATmega640	\$--	\$1E,\$96,\$08	64K (256)	4K (8)	LS,HP,JTAG,BL	
ATmega644(P)	\$--	\$1E,\$96,\$09	64K (256)	2K (8)	LS,HP,JTAG,BL	
ATmega103	\$41	\$1E,\$97,\$01	128K (256)	4K (8)	LS,HP	
ATmega128	\$43	\$1E,\$97,\$02	128K (256)	4K (8)	LS,HP,JTAG,BL	ブートローダ
	\$44					
ATmega1280	\$--	\$1E,\$97,\$03	128K (256)	4K (8)	LS,HP,JTAG,BL	
ATmega1281	\$--	\$1E,\$97,\$04	128K (256)	4K (8)	LS,HP,JTAG,BL	
ATmega2560	\$--	\$1E,\$98,\$01	256K (256)	4K (8)	LS,HP,JTAG,BL	
ATmega2561	\$--	\$1E,\$98,\$02	256K (256)	4K (8)	LS,HP,JTAG,BL	





本社

Atmel Corporation

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

国外営業拠点

Atmel Asia

Unit 1-5 & 16, 19/F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
Hong Kong
TEL (852) 2245-6100
FAX (852) 2722-1369

Atmel Europe

Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-Yvelines
Cedex
France
TEL (33) 1-30-60-70-00
FAX (33) 1-30-60-71-11

Atmel Japan

104-0033 東京都中央区
新川1-24-8
東熱新川ビル 9F
アトメル ジャパン株式会社
TEL (81) 03-3523-3551
FAX (81) 03-3523-7581

製造拠点

Memory

2325 Orchard Parkway
San Jose, CA 95131, USA
TEL 1(408) 441-0311
FAX 1(408) 436-4314

Microcontrollers

2325 Orchard Parkway
San Jose, CA 95131, USA
TEL 1(408) 441-0311
FAX 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3
France
TEL (33) 2-40-18-18-18
FAX (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards

Zone Industrielle
13106 Rousset Cedex
France
TEL (33) 4-42-53-60-00
FAX (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR
Scotland
TEL (44) 1355-803-000
FAX (44) 1355-242-743

RF/Automotive

Theresienstrasse 2
Postfach 3535
74025 Heilbronn
Germany
TEL (49) 71-31-67-0
FAX (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Biometrics

Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex
France
TEL (33) 4-76-58-47-50
FAX (33) 4-76-58-47-60

文献請求

www.atmel.com/literature

© Atmel Corporation 1998.

Atmel製品は、ウェブサイト上にあるAtmelの定義、条件による標準保証で明示された内容以外の保証はありません。本製品は改良のため予告なく変更される場合があります。いかなる場合も、特許や知的技術のライセンスを与えるものではありません。Atmel製品は、生命維持装置の重要部品などのような使用を認めておりません。

本書中の®、™はAtmelの登録商標、商標です。
本書中の製品名などは、一般的に商標です。

© HERO 2020.

この使用者の手引きはAtmelのAVRprog英語版使用者の手引き(改訂1021A-01/98)の翻訳日本語版使用者の手引きです。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。

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。