

このソフト(avrdudewriter.exe)について：

これはavrdude.exeを使って、AVRの各ヒューズ値、ロック値及びメモリーの内容を読み書きするソフトです。

はじめに：=====AVRにHEXファイルを書けるようになりたいと思うと、次のようなことが心配になる。

- 1) Arduino Unoは持っているが市販の書込装置を持っていない！。
- 2) avrdude.exeを扱えるか？。
- 3) MS-DOSを詳しく知らないで、パスの設定、コマンドプロンプト画面でのディレクトリ移動、コマンドや引数の使い方や操作ができるか？。

その様な不安が無くても、コマンドプロンプト画面での操作は結構手間取る。

そこで、初心者でも使い易い次のような使用(仕様)を想定してVisual Studioでこのソフトを作ってみた。

- 1) avrdudeで動作していること、動作させている方法が見えるように。
- 2) avrdudeのコマンド、引数はこのソフトだけで作成、かつマウスで操作可能。
- 3) 書込装置はArduino UnoをISP化したArduino as ISPとする。
- 4) パスの設定は不要。
- 5) ディレクトリーの移動は不要。
- 6) コマンドプロンプト画面はソフトから起動。
- 7) コマンドプロンプト画面でのキーボード操作は不要(Enterキーは必要)。

必要なもの(使用したもの)：

Windows 11
Arduino IDE 1.8.13
Arduino Uno R3(但し：純正品)を書込装置(Arduino as ISP)にして使用する(*1)
AVR/ATMEGA328P
市販のブートローダー書き込みシールド
または同等品、例えば：

Arduino Uno用シールドボード(PCB/DIY)やブレッドボード、16MHz水晶、ZIFソケット、LED、抵抗、コンデンサー等の部品および配線材

(*1) Arduino Unoを書込装置(Arduino as ISP)にする方法は別途参照。

(*2) CH340を使っている互換品Arduino Unoについて。

ドライバーをインストールし直して使用している。

起動して最低限のコマンド(ヒューズ部、メモリー部、オプション部は全て空白)を実行すると次のエラーが表示される。(全てではなく、一部のみ記載した)

Double check connections and try again, or use -F to override this check.

オプション部に-Fを入力してコマンドを実行すると正常に動作するようになるが、一度USBを取り外して再度接続すると、再度同じエラーが表示される。

インストール/アンインストール：

圧縮ファイルを書込可能フォルダー(例えばuserのDocument内)に解凍またはその結果を移動し、avrdudewriter.exeファイルを起動する。
アンインストールはフォルダごと削除する。

1. 初期設定(default.ini)画面説明：=====使用する各種パラメーターを登録する。

avrdudewriter.exeを起動すると、初期設定がされていなければ、この画面が表示される。

既に初期設定されている場合は、この画面は表示されず、メイン画面が表示される。

またこの画面は、メイン画面のメニューからも表示可能。

1.1) avrdude部

ボタン①をクリックすると表示されるファイルダイアログ画面で、Arduinoフォルダー内にあるavrdude.exeファイルを選択すると、②にその絶対パスが登録される。

ボタン③をクリックすると表示されるファイルダイアログ画面で、Arduinoフォルダー内にあるavrdude.confファイルを選択すると、④にその絶対パスが登録される。

二つのファイルが正しく登録されていないとメイン画面は正常に動作しません。

1.2) 引数部

後で追加登録・変更可能なので、最初は規定値のまま使用する。

登録はavrdude.exeのフォーマットに従ってください。

空欄は無視されますのでメイン画面では表示されません。

例えばpartsnoで次のように登録した場合"m328p, ?, (空欄), m168p, t2313"

メイン画面では"m328p, ?, m168p, t2313"と表示されます。

尚、次の引数は1番目(一番左)に登録したものがメイン画面で自動的に選択されます。

```
programmer:stk500v1
baudrate  :19200
partno    :m328P
```

必要なら登録し直してください。

port: 登録する場合は大文字(例えばCOM3, COM4)で登録してください。

(接続されているポートを自動検出するので、通常は登録する必要はありません)

1. 3) モード部

疑似darkmodeの設定変更は次回起動時に有効となる。

1. 4) 変更後は保存して終了する。

2. メイン画面説明 : =====

2. 1) avrdude部 :

初期設定が正しくされていれば設定したavrdude.exeとavrdude.confの絶対パスが表示される。

2. 2) connection部 :

初期設定画面で1番目(一番左)に登録したものが自動的に選択されます(除くport)。
ですから、初期値では次の通りになります。

programmer:stk500v1が選択され表示される。

port : 下記参照。

baudrate :19200選択され表示される。

partno :m328P選択され表示される。

2. 2. 1) portについて :

USBを接続しないでavrdude.exeを起動した場合 :

ドロップダウンをクリックしてもコンボボックスには何も表示されません。

(初期設定で何も登録されていないため)

USBを接続して①のDetectボタンを押すと再度自動検出を行い検出したものを
コンボボックスに登録するのでドロップダウンをクリックして選択する。

USBを接続してavrdude.exeを起動した場合 :

ドロップダウンをクリックすると自動検出されたものだけが登録されているので選択する
重要！！！！！！

接続されているポートは自動検出するので、通常は登録する必要はありませんが、
登録する場合は大文字(例えばCOM3, COM4)で登録してください。

例え登録していても自動検出されたportが同じなら再登録(二重登録)しません。

自動検出でエラーが発生したことは無いが、その場合はportを登録して対応願います。

2. 2. 2) その他 :

programmerの名前はそれぞれの製品によって決められている。

ARDUINO ISPとかARDUINO AS ISPという名前では動作しない。

次の値を入力または登録・選択してコマンドを作成・実行(コマンド部参照)すると、
利用できるprogrammer一覧が表示される。

-c ?

or

-c?

partnoの名前はそれぞれの製品によって決められている。

ATMEGA328Pの場合はm328pになる。

次の値を入力または登録・選択してコマンドを作成・実行(コマンド部参照)すると
利用できるpartno一覧が表示される。

-p ?

or

-p?

avrdude.exeのコマンド一覧

次の値を入力または登録・選択してコマンドを作成・実行(コマンド部参照)すると
コマンド一覧が表示される。

-?

- ?(この場合は動作しない)

2. 3) fuse部 :

2. 3. 1) efuse値を読んで画面に表示する場合は①のコンボボックスでefuse:r:con:hを選択する。

④は空白のまま変わらず。

conは出力先としての画面を表す。以下同じ。

尚、コンボボックス上でconをefuse.txtに書き換えてefuse:r:efuse.txt:hにするか

efuse:r:efuse.txt:hを登録して選択するとStartupPath(後述)内のefuse.txtファイルにその値が書き込まれる。以下同様。

hfuseの場合はhfuse:r:con:hを選択する。

lfuseの場合はlfuse:r:con:hを選択する。

lock場合はlock:r:con:hを選択する。

2. 3. 2) efuseに値を書き込む場合は①のコンボボックスでefuse:w:0x##:mを選択する。

②に書き込む値を16進数2桁(大小文字どちらでもよい)で入力して③setボタンをクリックする。

④には①の##部分が②で入力した値に変わった引数が表示される。以下同様。

hfuseの場合はhfuse:w:0x##:mを選択して同じ操作を行う。

lfuseの場合はlfuse:w:0x##:mを選択して同じ操作を行う。

lockの場合はlock:w:0x##:mを選択して同じ操作を行う。

lock値はこのコマンドだけでは指定した値に変更できない場合がある。

例えばこのコマンドでデフォルト値の0x3Fを0x0Fには変更できますが、

その反対を行うとエラーを表示し変更できません。

-e optionで初期化すると0x0Fが0x3Fになります。

尚、誤った値の変更設定はATMEGA328Pが使用できなくなる場合が生じます。

2. 4) memory部 :

flashメモリーに書き込む場合(HEXファイルを書き込む場合がこれに相当する)は

①のコンボボックスでflash:w:<filename>:iを選択する。

ボタン②をクリックすると表示されるファイルダイアログ画面で、書き込む内容が保存されているファイルを選択すると、その絶対パスが③に表示される。

④には①と②から作成された引数が表示される。

尚、ボタン②をクリックしないで③に直接出力先を書くことも可能。

例えば、conとかflash.txtとか。

その場合には画面またはStartupPath(後述)に出力される。

2. 5) option部

その他に必要な引数を記入または登録したコンボボックスから選択する。

2. 6) command部 :

これまでに設定した各値を結合して最終コマンド・引数を作成する。

ボタン①をクリックすると、これまでに指定した各値を全て反映したコマンド及び引数が②に作成・表示される。

ボタン③をクリックすると作成されたコマンド・引数が全て選択され、クリップボードにコピーされる。

ボタン④をクリックするとコマンドプロンプト画面が表示されるので、画面上で右クリック(またはCtrl+v)するとコピーした内容が貼り付けられ、

Enterキーを押すとコマンドが実行され、結果が表示される。

Wait for exit cmd.exe⑤にチェックをしていた場合は、コマンドプロンプト画面を閉じないとメイン画面での操作を行うことはできない。

2. 7) 重要 ! ! ! ! !

最終commandはavrdude部、connection部、option部、memory部、fuse部で入力された各コマンドを順番に結合しただけで、その順番に不都合があったとしてもそれを検出できません。

例えば次の2種類の処理が同時に存在していた場合、

①メモリの或る場所の値を或る値に変更する。

②内容を保護するためにメモリを書込み禁止にする。

この場合は処理の順番が②、①になってしまっていては意味がありません。

その場合は先ず①の処理を先に行い、その後に②を処理をするように複数回に分けて処理しなければなりません。

3. その他 : =====

3. 1) avrdude部とconnection部だけが正しく入力された状態でcommand部の操作を全て行うと、コマンドプロンプト画面にはfuse値(efuse, hfuse, lfuse)が表示される。

3. 2) optionで-vを指定すればより詳細な内容が表示される。

3. 3) 全てのコンボボックス及び一部のテキストボックスでは、その値を変更できるが

(例 2. 3. 1)/2. 4)、その際には各部で正しく反映されているかを確認してください。

3. 4) 各書き込み時には不適切な設定をしていないか十分注意してください。

3. 5) StartupPathとはアプリケーションを開始した実行可能ファイルがある場所です。

メイン画面のメニュー操作で表示して確認できる。

3. 6) config. ini ファイルのパスは初期画面のメニュー操作で表示して確認できる。

3. 7) bootloader書き込み済みAVRにこのソフトでHEXファイルを書き込むとbootloaderは削除されるので、必要な場合は再度bootloaderを書き込まなければなりません。

添付 : ArduinoISP Writer.zip

(圧縮ファイルの内容)

avrdudewriter.exe

readme.txt

documents(folder) :

doc_01_avrdudewriter使用説明.pdf

doc_02_使用事例.txt

doc_03_bootloaderについて.txt

form_01_スクリーンショット(メイン画面).png

form_02_スクリーンショット(初期設定画面).png

form_11_メイン画面説明.png

form_12_初期設定画面説明.png

form_20_Arduino UNO BOOTLOADER Shield.pdf

form_30_Connection Diagram.png

以上