# Macintosh OS X でAVR-GCC と GNUアセンブラを使うには

CrossPack-AVRというパッケージをダウンロードし，インストールすれば準備完了です．ユーティリティにターミナルがあるので，それでLinuxと同じようにプログラミングと，プログラムのアップロードが行えます．

　デモ用サンプルディレクトリ，あるいは   
/usr/local/CrossPack-AVR/etc/templates/TemplateProject   
をコピーして中で作業すればよろしい．

接続すると，「新しいネットワークインターフェースが検出されました」ダイアログが出るが，ネットワークインターフェースではないのでキャンセルします．Arduino IDEか，コマンドls /dev/tty.usbserial-\* で，シリアルポート名を確認してください．プログラムのアップロードの前にリセットボタンをおす必要はありません．

# WindowsでAVR-GCCとGNUアセンブラを使うには

　Windows XPで説明します．それ以降のWindowsではドライバのインストールは結構自動的に行われるらしい．

（０） 使うPCは， Arduinoをつないでスケッチがアップロードできるようになっていること．Arduinoソフトウェアを起動してArduinoをつなぎ，Tools/Serial PortメニューのリストでArduinoを接続するときに使うポートＣＯＭ番号を確かめておく．  
※USB仮想シリアルポートのドライバソフト（USB Serial ConverterとUSB Serial Portの二つ）がインストールされている必要があるため．インターネットへ接続できるなら， Webで解説ページをみつけ（たとえばGoogleでArduino, Windows, インストールの３語で検索）インストールと動作確認を済ませておく．PC/Windowsにくわしいなら，FDTI社のサイトからUSB仮想シリアルポート・ドライバだけインストールしてもよい．

（１） WinAVR をダウンロードしてインストールする．（たとえばGoogleでwinavr, インストールの２語で検索してインストール方法解説ページを見なさい．）

（２） \WinAVR-xxxx\sample （xxxxはバージョンによって変わる．）を，ディレクトリ（フォルダ）ごとコピーする．この中にあるMakefileを使うため．

（３） そのディレクトリにアセンブラのプログラムファイルmain.S を作る．実験で書いたプログラムをそこにコピーして使うのがいい．

（4） テキストエディタ（なんでもよい）でMakefile の，下のリストにある部分を書き直す．保存の時に形式を変えたり，拡張子をつけたりしないように気をつける．なおserial port の名前は，（０）で確かめておいたものに書き換える．

（5） Windowsのスタートメニューから，コマンドプロンプト（いわゆるDOS窓）を選んでひらく．（もしcygwinを使っているならそのシェルでよい．）上記ディレクトリに移動cdして，make （実際は make all ）すると，コンパイルされ，アップロード用の機械語のファイルmain.hexができる．

（６） プログラムを書き込むには，Arduinoが接続された状態でmake programを実行して，ただちにArduinoのリセットボタンを押し，はなす．（このリセット動作は1〜2秒以内に行う必要がある．）うまくいくと，Arduinoボードにプログラムがアップロードされ，さまざまなメッセージとともに進行状況をあらわすための ##### がたくさん出るのでわかる．  
コネクション失敗などのエラーが出て，うまくいっていないようなら（エラーメッセージは読みなさい），一通り確認して，この（６）の手順をやりなおす．

🖑注意 make cleanを実行すると，main.Sは消されてしまう！（main.cから自動生成されたファイルと勘違いされるため．）

## Makefileの要修正箇所

F\_CPU = 16000000

MCU = atmega328p

AVRDUDE\_PROGRAMMER = stk500v1 -b 57600

AVRDUDE\_PORT = com3