

Build Notes

To build the PWM_Controller (Windows XP)

- Download and install the IDE (visit <http://arduino.cc/en/> & click *Download*)
- Connect your Arduino Uno to a convenient USB port. If the *new device Wizard* appears, point it to the drivers subfolder in the installed package.
- Decompress the MW download package and move or merge its *Arduino* folder into your *My Documents\Arduino* folder.
- Open the *PWM_SpeedCtrl* sketch from the IDE or by double-clicking the *PWM_SpeedCtrl.ino* file from the Windows Explorer.
- Click the *Upload* button on the IDE. Comm errors are likely an incorrect COM port.

Note: The sketch files created by the IDE are plain ASCII text, but use Unix-style line breaks (a single LF) instead of DOS/Windows-style breaks (a CR/LF pair). The Notepad application in Windows XP will not display these correctly. Use the Arduino IDE or Wordpad instead.

Here are a few comments on the Arduino IDE:

- The sketch folder name must match the INO filename exactly, and is case sensitive.
- Use the Verify function (test compile) often, especially if you're new to C syntax.
- A simple typo like a missing semicolon will create a large number of errors.
- The cause of a reported error is frequently on a preceding line of code.
- In searching for compilation errors, try commenting-out your new code and re-enable small sections of it as you verify (recompile).
- The autoformat function is invoked when you type ctrl-T, and can be annoying when it happens by accident, especially if you don't like what it does to your sketch.
- The built-in documentation does not always specify the data types used in library functions. You may have to search the header files (*.h) to get this information.
- Many of the data types supported by C are not implemented as you might expect. For example, a *float* and a *double* may be the same (a float).
- The Arduino documentation does not do a good job of describing the distinction between pre-vs-post increment/decrement operators. The syntax “++i” first increments i and then evaluates the expression containing it. The syntax “i++” first evaluates the expression and *then* increments i.
- Conditional expressions followed by a single statement do not require it to be enclosed by curly braces { }. A block of multiple statements, of course, do need the curly braces.
- The *ternary* operator (syntax with ? :) is a compact way to assign a variable one of two values depending on a condition. It's similar to an if/then/else clause.

Build Notes

Here is a quick overview of how the PEM Controller code operates, and what routines are invoked by others. Despite the many lines of code, it's architecturally fairly simple.

