AVR C Tool Chain & IDE

Download & install the latest release of WinAVR first, then download & install AVR Studio.

Download & Install the latest demo version of Imagecraft ICCV7 for AVR.

You will use the WinAVR compiler which integrates with AVR Studio as a plugin.

You will need an AVR Dragon USB programmer (the department has lots, they are cheap). This can program any AVR chip and also debug chips which have less than 32K flash. AVR Studio integrates AVR Dragon as a programming and debugging tool. Three interfaces can be used, ISP or JTAG are commonly found on development boards. More details and header pin connections can be found from Atmel's AVR Dragon page.

NB – remember that for practical debugging you must be able to reprogram the AVR and that therefore it will be easier if the interface I/O connections are not used for anything else in your design – this means that programming leads can be connected continuously through the debug/test cycle.

Interface	header	program	Debug (in-circuit emulate)
ISP	6 way IDC	yes	No
JTAG	10 way IDC	yes	Yes <=32K flash
DW	3 way	no	Yes <=32K flash NB – must first set fuses via a programming interface

For problems, sample code, etc when using AVR check documentation on this site (LCD and serial link) if you need more information try www.avrfreaks.net and especially the tutorials. AVR is very widely used and you will find C sample code for many applications.
Note also the functions written in the GNU AVR libc library.

The **Imagecraft** demo compiler has an **application builder** which will generate sample C setup code for any number of AVR peripherals, including the system clocks, from a GUI. The generated code includes typical interrupt handlers if required. This is invaluable. The code can be copied (preview), inserted into an AVR Studio project C file, and modified further as required. Note that it should be put in a file separate from other WinAVR code due to incompatible macros. Make sure that you set the correct CPU (or at least a CPU with the same peripherals) in the application builder.

To understand fully the all the available setup options you will need to read the datasheet – but to use them you can just tick boxes in the application builder.

IMPORTANT HINTS using AVR Studio & WinAVR Tool chain

- Make sure the **I/O register window** is visible when using the debugger or working out I/O register coding – it is invaluable. It is found (obscurely) from **View->toolbars->I/O.**
- When debugging without AVR Dragon (& in circuit debugging) use simulator 2 normally, but if you have problems try simulator 1 as well.
- When debugging use –O1 to allow debug of c locals etc. For real code use –Os for minimum flash size.
- If RAM space gets used up consider using program space storage of data, particularly constant strings. See AVR libc **pgmspace.h** for discussion.
- If you run out of RAM/ROM consider moving to a bigger compatible microcontroller – 16K & 32K versions are both available and cheap. Beware moving above 64K ROM since 4 byte code pointers must then be used which is automatic but may create some obscure incompatibilities. Note that 64K ROM will mean AVR Dragon can't do in circuit debugging.

Other resources

AVR LCD Howto

AVR Serial Link Howto