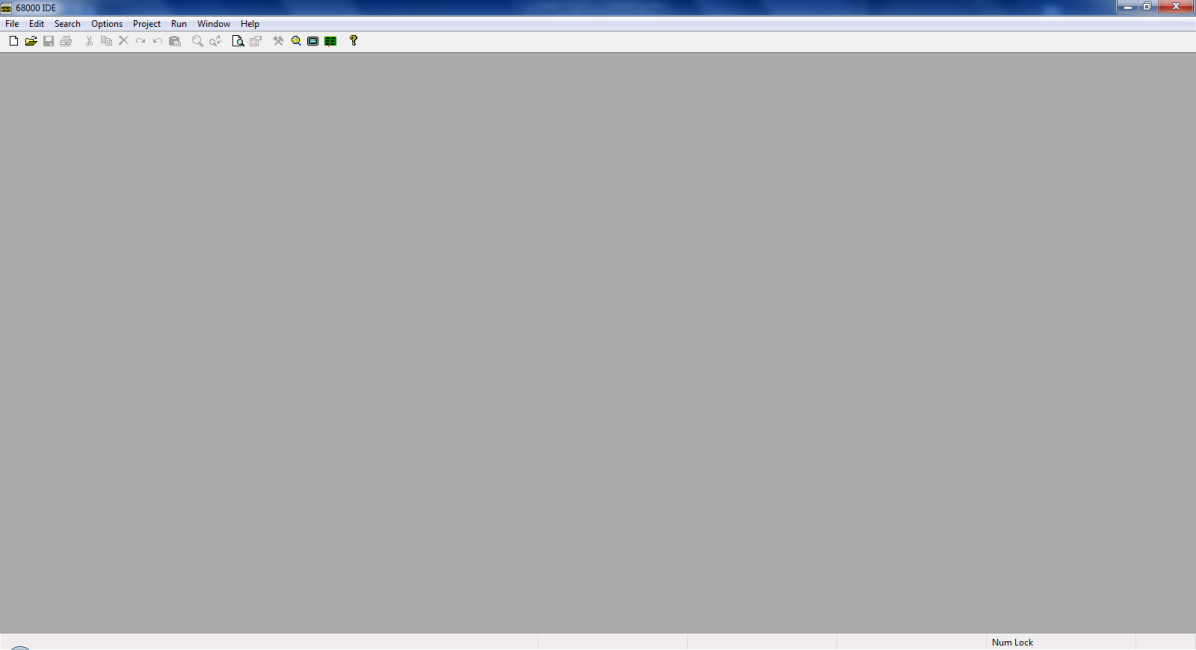
**Using IDE68k to write assembler programs for the 68000   
Soft Core Processor on the Altera Board**

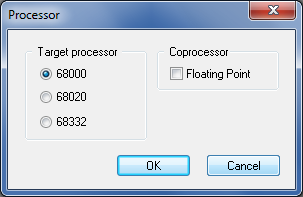
**Download, Install and Run**

* Download and Install the IDE68k (Integrated Development Environment for the 68k) software onto your PC and unzip it into the folder C:\IDE68k
* Start the IDE – the following screen appears.

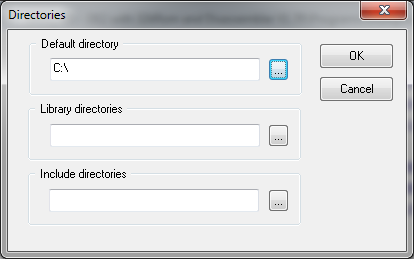


**Setting IDE Options**

Chose menu **Options->Processor** and chose **68000** as the target processor (see below). You only have to do this once, the first time you start IDE

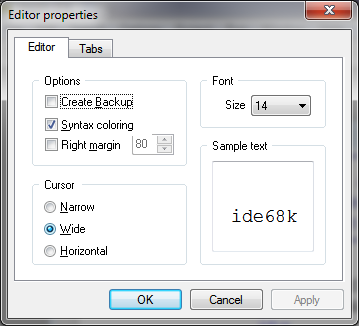


Chose menu **Options->Directories** and set the default directory (the library and include directories do no matter at this stage) - see below – **chose your own default directory name that is meaningful to you**. You only have to do this once, the first time you start IDE. The directory you specify here is where IDE will save your source files and any IDE generated files.

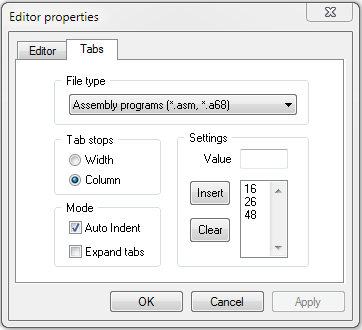


**Setting the Editor Properties**

Chose menu **Options->Editor** and set the properties as shown below

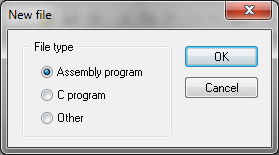


Click on the Tabs button below and set the following properties (make sure the File Type is set to Assembly Programs). Make sure the other settings are as shown below also. **IMPORTANT**: Set tabs at columns 16, 26 and 48



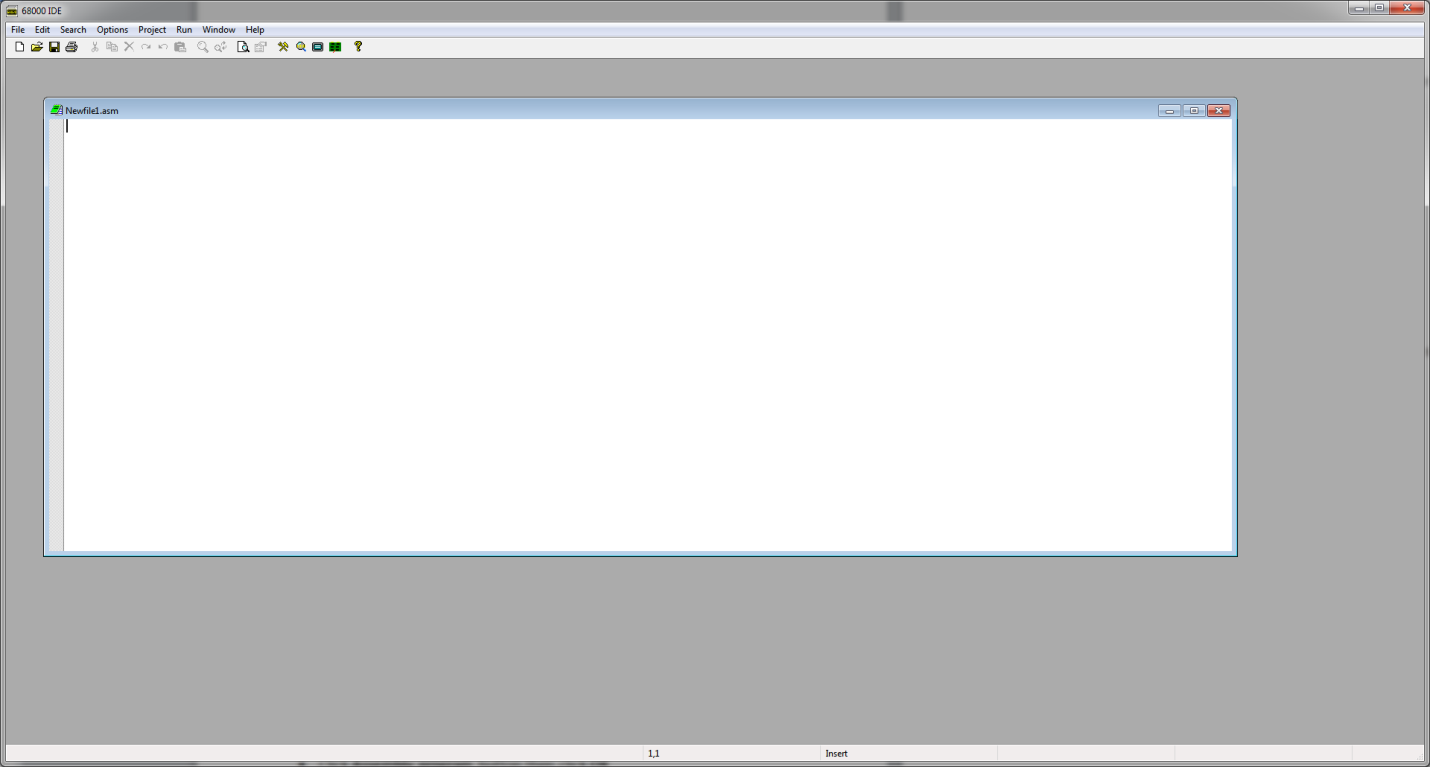
**Creating an Assembly Language Program**

* The IDE is very easy to use.
* Use the **File Menu** to create a new file.
* The following dialog box appears.



* Click on the **Assembly program** button then click **OK**

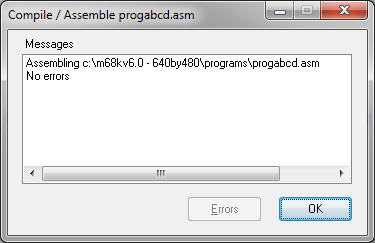
You will see a new file is created (see below)



**SAVE THE FILE with a more meaningful name. You can then type in your source code/assembly language programs**

**Assembling your Programs**

Press function key **F7** to build. The IDE is very fast, it has finished almost before your key press has been released. You should see a similar dialog box to the one below if your program assembled OK. If not, click on the errors button and then the individual errors themselves to locate and then correct the errors.



**Assembler Output Files**

Two other files will have been produced by the IDE (locate these and view them they should be in the default directory you set up with IDE at the start of this tutorial)

* A “.**masm**” file, an assembler listing file showing how your program has been converted to machine code.
* A “.**hex**” file which is the file we download to the DE2 board to download our program. This file is in the form of Motorola S-Records, a very popular standard. It looks something like this (*you will see this when you download the program to the DE2 board*)

S004000000FB

S21480000046FC00002E7C0084000023FCFFFFFFFFE0

S214800010008013EC42B9008013F0207C008013EC43

S214800020227C00860000203C0000040E670000084A

S21480003012D8538066FA4EB90080073A60C21039EB

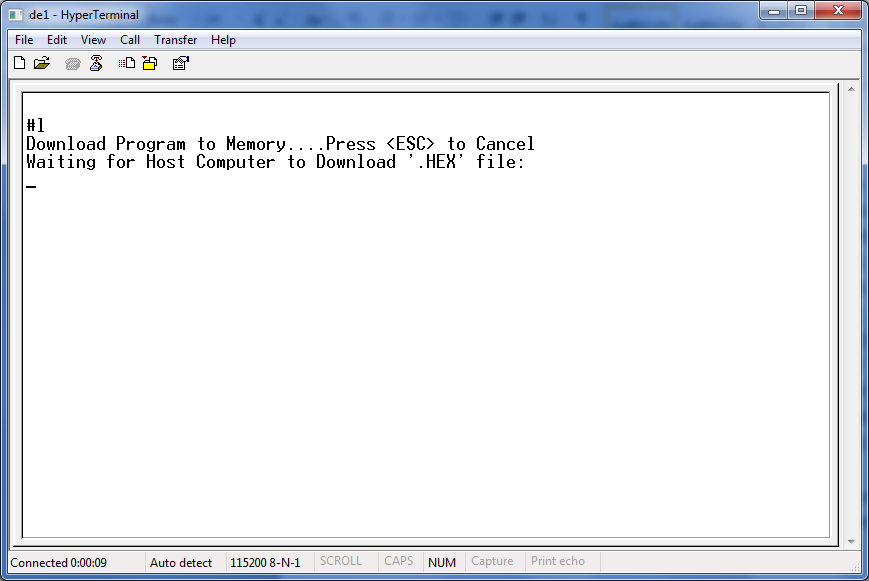
S8

S2 is the start of a record, followed by a 2 digit byte count (e.g. 14) followed by a 6 digit address (e.g.800000) followed by some data bytes, followed lastly by a two digit checksum at the end of the line

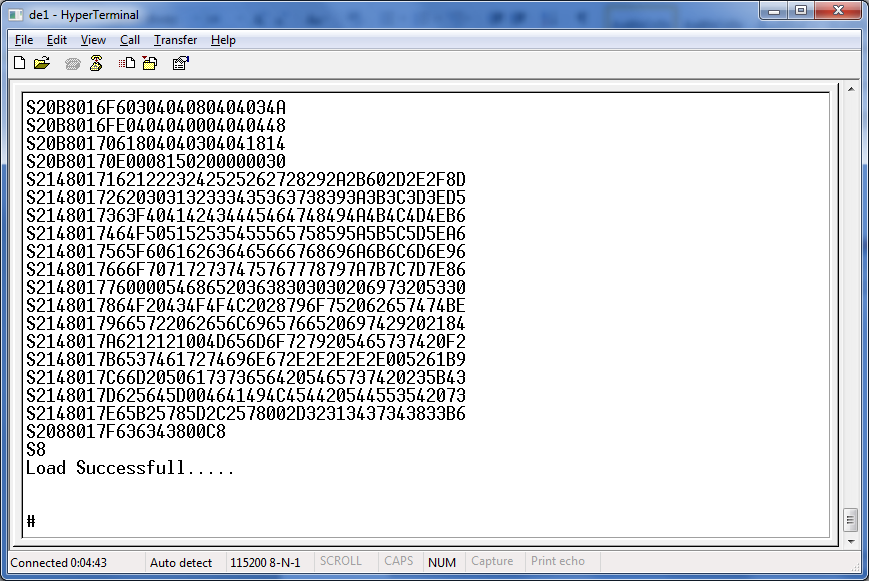
**Downloading the Program to the Board**

Using the Hyper terminal software, make sure you have communicated with the Altera board, i.e. have the “#” prompt and can communicate with the debug monitor (if you haven’t downloaded a working 68000 soft core processor design/system, e.g. the ‘MC68k.pof’ or ‘MC68k.sof’ file to the board you **won’t** get this prompt).

At the debug monitor prompt ‘#’, enter the command ‘**L**’ for load. The board will then wait for you to download (see below).



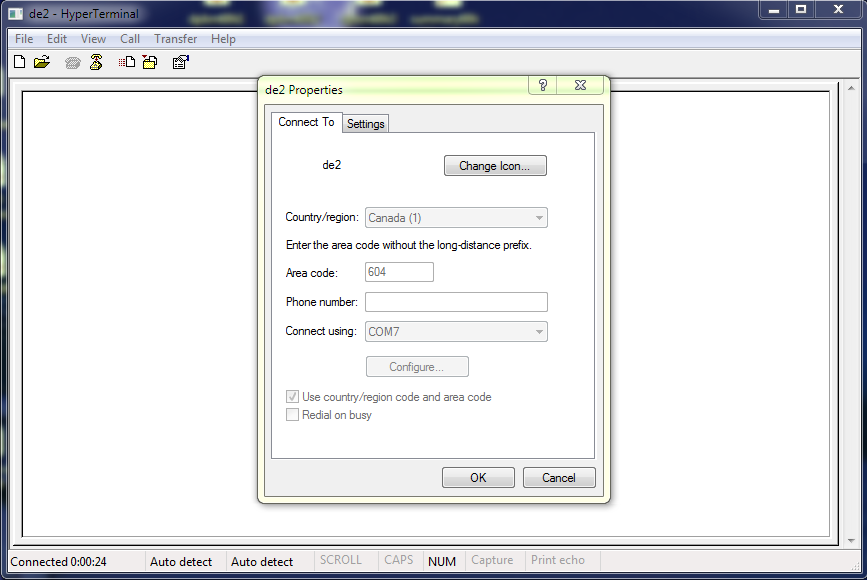
Click on the Hyperterminal “**Transfer**” menu and select “**Send Text File**”. Locate the file in the dialog box. You are looking for a file with a “**.HEX**” file name extension. Once you have found it, click OK and you should see a screen similar to that below (the data will vary) as the S-Records are downloaded. You should see the message “**Load Successful**” at the end, if all went well.



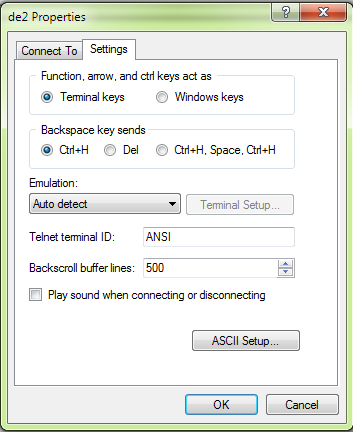
**NOTE**: The vast majority of laptops will work fine, **BUT**, if you experience problems downloading your program, then first try a different USB port on your laptop (USB 3 ports seem to have more problems than USB 2 due to their higher speed). If that fails experiment with some changes to the communication speed from within the **HyperTerminal** program as follows.

In **HyperTerminal**, click File->Properties, the following screen appears

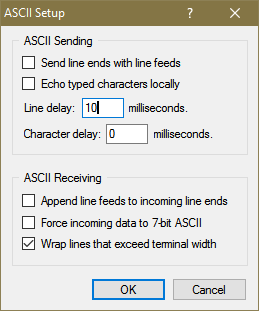
In the **HyperTerminal program**, click File->Properties, the following screen appears



Click on the Settings tab, the following appears



Click the ASCII Setup button, the following windows appears



Experiment with the value of the Line Delay (e.g. 10) to pause transmission after each line of text is sent – try different values until you find one that works. You unfortunately will have to enter this again **each time** you start HyperTerminal (unless you buy the HyperTerminal program which allows you to save the settings).

**Running your Program on the Board**

* At the “#” debug prompt, type ‘g’ for go.
* Press the Blue reset button (Key0) on the board at any time to stop your program and reset the 68000 processor on the board. This will return you to the debug prompt ‘#’.