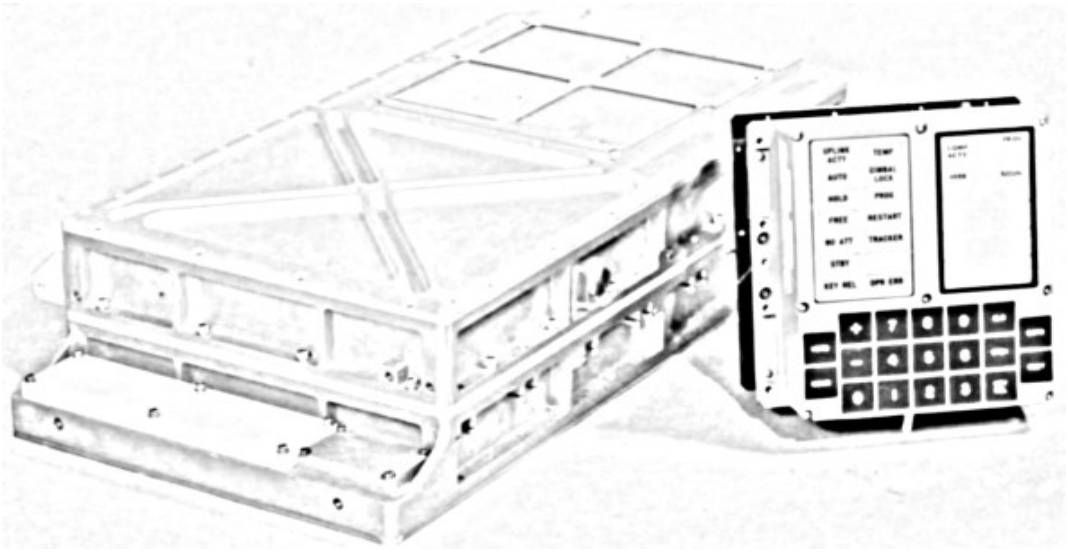
Syntax Highlighting
for
AGC Assembler Code



Written by Onno Hommes

December 31 2007

Table of Contents

AGC Syntax Highlighting

This section explains how to get AGC syntax highlighting enabled in several frequently used editors under Linux and Windows. Although only a select list is currently supported it is hoped for, that this list will stimulate others to add and submit additional highlight files (i.e. Highlight source and instructions on how to get the syntax recognition to work).

The following environments are currently supported:

- Kate (Linux)
- KWrite (Linux)
- KDevelop (Linux)
- Eclipse (Linux/Windows)
- TextPad (Windows)
- Vim (Linux)
- Gvim (Windows)

The next sections will explain how to enable syntax highlighting for each of these respective editors. Once enabled you'll be able to more quickly navigate and digest the wonders of the original AGC code Colossus and Luminary or write your own code from scratch for the Virtual AGC simulator.

In case where Linux is mentioned as the operating system in the list above in many cases if the tool works on other platforms it should work as well (e.g. Vim).

The files referenced in this document can be obtained by downloading the file *SyntaxAGC-<date>.zip*

The general folder layout of the SyntaxAGC distribution is to have a folder for each supported tool which makes locating the syntax files a breeze for your desired tool.

Enable Kate AGC Highlighting

To enable highlighting in Kate only a few simple steps have to be taken.

1. Find the file agc.xml in the kate folder in the SyntaxAGC distribution.
2. Copy the agc.xml to the katepart/syntax directory (make sure you have proper privileges)

   Example: cp agc.xml /usr/share/apps/katepart/syntax/.
3. Now start Kate

Doesn't get any easier than that. If you do not have root privileges you may want to add this file to your user files (*$HOME.kde/share/apps/katepart/syntax*)

Once you completed the above steps you also immediately have source highlighting in KWrite and KDevelop.

Enable TextPad AGC Highlighting

Enabling syntax highlighting in TextPad is quite straight forward. It requires very few steps to get this working:

1. Find the agc.syn file in the textpad directory in the SyntaxAGC distribution and copy this file to the TextPad (usually located in the “Program Files” folder Samples directory.
2. Launch TextPad
3. Add a new document class. You can find the menu option “New Document Class” on the Configure menu. For the document class add AGC as the name and “*.s” as the members. To finalize enable the syntax highlighting and select the agc.syn as your Syntax definition file.

That is all there is for TextPad. Now open a AGC assembler file and enjoy the highlighting.

Enable Eclipse AGC Highlighting

To enable highlighting in Eclipse you have several options: You can write your own development plug-in or use a more elegant more open eclipse plug-in editor that just requires a syntax description file. Here we will use the more simple and elegant approach using the editor called Colorer 5 which supports more than 100 programming/database/script languages. To get the AGC syntax highlighting in Eclipse using the editor plug-in Colorer follow the following steps:

1. Install Colorer in Eclipse. Just use the build in Find and install in Eclipse under the Help menu and add a new remote install site: http://colorer.sf.net/eclipsecolorer/
2. Verify the Colorer plugin is installed (usually a restart of Eclipse is required) by going to the Preferences menu option under the Windows menu. If you see a root node Colorer... in the preferences dialog then you have properly installed the plug-in.
3. Locate the eclipse subdirectory in the SyntaxAGC distribution.
4. Copy the contents of the eclipse directory into the *<path>/eclipse/plugins/net.sf.colorer_<version>/colorer/hrc* directory. This will add the files agc.ent.hrc and rare/agc.hrc to the hrc directory.
5. Now theoretically colorer allows you to use an include mechanism to get the agc.ent.hrc file included into the proto.hrc file (also in the hrc directory). However after several failed attempts to get this to work you may just want to add the contents of the agc.ent.hrc file to the proto.hrc file your self. Just insert the text next to any other prototype section (.e.g. Right under the XML comment: <!-- main languages -->
6. Now restart Eclipse
7. Now associate the .s file type with the colorer editor. Goto Preferences->General->Editors->File Associations. You will probably already see the .s type if not just add a new *.s file type and associate the Colorer editor.

That is it and now start creating your new AGC project or navigate and read the existing Colossus or Luminary code base.

Enable Vim AGC Highlighting

This section should work for both Vim on Linux distributions as well as on Windows (i.e. gvim). Execute the following steps to enable AGC syntax highlighting:

1. Locate the agc.vim file in the SyntaxAGC distribution
2. copy the agc.vim file to the vim syntax directory. On linux this would be something like:

   cp <path>agc.vim /usr/share/vim/vim71/syntax/.

   On windows copy the file into something like:

   C:\Program Files\Vim\vim63\syntax

   Obviously your installed versions could be different from the above examples and so must be appropriately adapted to fit your installed version
3. This could conclude this install but then you are left with executing the following manual steps for each loaded file:

   :set syntax=agc
4. Since this is quite frustrating you may want to update the filetype.vim file. So locate this file in your Vim install directory.
5. Add just before: “Assembly (all kinds)

   *au BufNewFile,BufRead *.s setf agc*

This completes the Vim install. Notice that because of the location of the “au” command you have overwritten the default behavior of the syntax highlighting of *.s files for general identified assemblers. If this something you don't want and you want automatic content recognition then you need to update the asm detection function in the vim files.