WinANSI
Windows ANSI Editor

Created by Tim Jones
AmSoft Development

About WinANSI

WinANSI came about while I was evaluating Russell Frey's PowerBBS for Windows. As a part of PowerBBS setup, you are allowed to edit the text files displayed by the BBS system to callers at various times. Mr. Frey defaulted the called editor to 'TheDRAW', which is an excellent ANSI editor if you don't mind leaving the Windows environment. Well, I didn't have 'TheDRAW' at the time, so I changed his default to Windows' NOTEPAD, only to be disappointed when it wouldn't let me enter a literal ESC character. After looking around for a week on various BBS's, I decided that I would have to do it myself if I were going to satisfy my need for a true Windows-based ANSI editor. WinANSI is the result of my work.

WinANSI provides a Windows 3.0 user with the ability to edit ANSI display screens without exiting Windows. While it is not as full featured as programs like TheDRAW, offering no animation or COM file creation, WinANSI does provide all of the standard ANSI escape sequences and line drawing capabilities required in most ANSI BBS screens. WinANSI does not provide an active editing environment as with TheDRAW. WinANSI allows you to manipulate the actual ANSI escape sequences and then view the results of your editing session in a separate window.

WinANSI is Shareware

WinANSI 1.2 is a Shareware product. Licensing of WinANSI is very simple; if you like WinANSI and wish to continue using it past an evaluation period evaluation period of thirty days, send US$15.00 to:

> Tim Jones
> AmSoft Development
> 3032 Higate Dr.
> Deltona, Fl 32738
> USA

Checks must be in US funds drawn on a US bank and made out to Tim Jones. Cashier Checks and Money Orders will be processed immediately. Others allow seven working days (the bank's rule, not mine) for the check to clear. Registration provides you with information on updates, the latest version on diskette (Specify low or high density and which disk size you require), information about other AmSoft products and services and a clear conscience. Also, please show where you obtained your copy of WinANSI in your correspondence.

WinANSI is Copyright © 1991, AmSoft Development. All rights reserved. You may distribute this shareware version via normal shareware channels (i.e., BBS's, shareware disk collections, etc.), but you are not allowed to sell WinANSI for profit.

PowerBBS is a shareware DOS or Windows BBS written by Russell Frey
PKZIP and PKUNZIP are registered trademarks of PKWare, Inc.
Windows and MS-DOS are registered trademarks and VisualBASIC is a trademark of Microsoft Corporation.
IBM and IBM PC are registered trademarks of International Business Machines Corporation

Section 1 - Installation of WinANSI

Installing WinANSI in Windows 3.0

In order to begin using WinANSI, you must properly install the parts that make up the whole. The package you received, if in **PKZIP** format, should contain the following files:

WINANSI.EXE	The WinANSI executable
WINANSIV.PIF	The PIF PIF file used to view the results of

your work

WINANSI2.WRI	This document file in Windows Write format
LINESAMP.TXT	Line Draw Sample ASCII Text
LINESAMP.ANS	Line Draw Sample with ANSI Color
BLOKSAMP.TXT	Block Character sample ASCII Text
README.TXT	Last minute information

You will also need the widely available VBRUN100.DLL Windows VisualBASIC Dynamic Link Library. It should be available on the BBS from which you downloaded the WINANSI.ZIP file. Look for a file called VBRUN100.ZIP or VB_DLL.ZIP.

After unZIPping the files, place the VBRUN100.DLL and WINANSIV.PIF files in a directory within your DOS path. Place WINANSI.EXE in a directory of your choice. If you are not in Windows, start Windows and select the group into which you wish to add WinANSI. Select New from the File menu and click Program Item Program Item (if not already selected). Enter the following information in the fields provided:

Description	**WinANSI 1.2**
Command Line	**{d:path\}WINANSI.EXE**

The {d:path\} are the drive and path into which you placed the WinANSI executable file. If you can't remember the directory in which you placed WINANSI.EXE, click on the Browse button to find it.

If you are familiar with the operation of the File Manager and the Drag and Drop method of adding files to your program groups, you may also add WinANSI in this manner. For more information on this operation, see the Windows User's Guide, page ... no, wait! they don't even acknowledge that drag and drop can be done. Boy howdy. Well here's what you do:

Open both Program Manager and File Manager on your desktop and arrange them so that you can see both. Open the group you wish to install WinANSI into. Select the "WINANSI.EXE" application icon in the File Manager by clicking it once and holding down the left mouse button. While holding down the mouse button, drag the pointer to the group window in the Program Manager and release the button. The WinANSI ICON should appear.

To test the results, double click the Palm Tree WinANSI Icon to start WinANSI. If this works, you are now ready to begin using WinANSI or you may continue with the next step, installing WinANSI as the editor for PowerBBS.

Installing and Using WinANSI in PowerBBS Config

The first step required to use WinANSI with PowerBBS PowerBBS is to install WinANSI as the default text editor in the PowerBBS config program. The instructions here relate to installation under "PowerBBSs CONFIG for CONFIG for Windows," not the DOS version. Start Power CONFIG and select option A, General Setup. Use the pagedown key to get to the screen titled "Options 3" and enter the full path and filename under which you installed WinANSI.

Next, WinANSI requires that you give information about where PowerBBS exists. You must edit your AUTOEXEC.BAT file and add a line that says:

        SET POWERBBS=d:\path\

Be sure to include the full drive and path information, including the final '\'. The next step requires that you create a text file in your PowerBBS directory called PBBSNODE.LST. This list consists of the names of the various POWRBBS.DAT files that you have assigned to your nodes. For instance, if you have a local node called PBBS0.DAT and COM1 and COM2 nodes called PBBS1.DAT and PBBS2.DAT respectively, you would create your PBBSNODE.LST file as follows:

        0,   PBBS0.DAT
        1,   PBBS1.DAT
        2,   PBBS2.DAT

Even though PowerBBS currently only supports COM1 through COM8, WinANSI provides support for up to twenty node configuration files (including the local node). If no POWERBBS environment variable or PBBSNODE.LST file is found, you will be asked for the location of the PowerBBS files. If you are not using PowerBBS, just click on the "No PowerBBS" button. You are now ready to begin using WinANSI to edit your text and ANSI display files.

To edit the selected file, use F1 for the ASCII text version of the file or F2 for the ANSI version. For some PowerBBS files, there is no ANSI equivalent to the text version. Please refer to the PowerBBS documentation for more information. If the file does not exist, WinANSI will warn you and ask if you would like to create it. Answer appropriately to begin your editing session or return to the PowerBBS CONFIG to select another file.

With release 1.2 of WinANSI, the only actual direct application communication between PowerBBS CONFIG and WinANSI comes from the WinANSI side, causing the PowerBBS CONFIG application to minimize itself when WinANSI is started and then Restore itself when WinANSI exits. Discussions between myself and Mr. Frey will hopefully permit the inclusion of two way communications using DDE (Dynamic Data Exchange) between WinANSI and PowerBBS CONFIG in the next release cycle.

Section 2 - Using WinANSI

Now that you have installed WinANSI, you can use it to edit both standard text files standard text files and ANSI text files. If you are using WinANSI with PowerBBS CONFIG, WinANSI will open with the selected file loaded. All you need to do now is make the changes and adjustments to the file and select Save, Save As or Exit from the File menu. If you started WinANSI from the Program Manager, you will be presented with a blank slate upon which your ANSI masterpiece may evolve.

Because most terminal emulation programs use the standard IBM PC ASCII character set, this is the character set that is used in WinANSI. Unfortunately, Windows does not allow any of it's text processing applications to utilize the Terminal font, so I am unable to show the actual character value/character combinations within this text.

WinANSI Key Definitions

Editing in WinANSI is as easy as typing typing in any normal text editor with a few exceptions. If you wish to place a {TAB}, you would use the key combination {CTRL-TAB}. WinANSI uses the DOS default tabstop setting of every eighth character and actually translates the {TAB} into the appropriate number of {SPACE} characters to take you to the next tabstop position.

If you wish to enter one of the special characters like the Sigma symbol, you would use the Windows mechanism of holding the {ALT} key down while typing the ASCII value for the character as shown in the appendix of the Windows manual (it's 228 for the Sigma symbol!). Upon releasing the {ALT} key, the appropriate IBM PC ASCII character will appear in your text.

To enter line draw characters, you could use the {ALT}-keypad method, but WinANSI allows you to use the keypad keys (with {NUMLOCK} on) to enter these characters. To see the key definitions, select Key Help from the Special menu. To enter the block draw characters from the IBM PC ASCII set, you would use the function keys {F1} through {F8}. Once again, the Key Help option will show you how these keys are assigned.

Developing ANSI Screens

Development of an ANSI screen should be performed in two steps. First you should create a standard ASCII version of your display and ensure that the file displays as you expect by selecting View File View File from the Special menu. Once the ASCII file is complete, you may now add ANSI escape sequences ANSI escape sequences to provide color image definition.

Let's take a look at a simple development process. For purposes of demonstration, we'll create a new Hello file for an imaginary BBS called "The Simple BBS" that requires a half screen display. Open WinANSI to a blank slate. Use the following keystrokes to create the initial ASCII screen:

```
{ENTER}
{CTRL-TAB}Welcome to:{ENTER}
{ENTER}
{CTRL-TAB}{CTRL-TAB}The Simple BBS{ENTER}
```

Your ASCII screen should now consist of four lines as shown below:

Welcome To:

The Simple BBS

To see what it will look like, select View File from the Special menu. Keep in mind that Windows doesn't currently support the Blink ANSI attribute. Your readers will see blinking, if they are using the standard DOS ANSI.SYS driver. Double Click the Control Box on the window to return to WinANSI. Now we'll add color to the display.

Position the cursor just to the left of Welcome and press the {ESCAPE} key. Click the Bright button under Intensity and then the Green button under Foreground Color and click Okay. You should now see an arrow followed by "[1;32;40m" and the rest of your original text. Now Position the cursor after the colon and press {ESCAPE} again. This time, just click on the Set Default button. You will see the arrow followed by "[0;37;40m" which will return the color scheme to the normal grey on black display.

Take a look at your file by selecting View File under the Special menu or pressing the {F12} function key. It should now have the "Welcome to:" text in bright green and "The Simple BBS" should be in normal grey.

You can provide additional color in much the same manner by placing the cursor before the section that we wish to add color to and pressing the {ESCAPE} key.

Using Line Draw Characters

In addition to standard text and ANSI colors, WinANSI allows you to enter the various line draw characters supported by the IBM character set. To use these characters, select the type of lines you wish to use using the "Line Draw Type" buttons at the top of the WinANSI display. Entering the characters is then performed by using the numeric keypad, numbers "0" through "9" and the "." key. To see the character assignments, select Key Help from the Special menu as described above.

Entering a box around the sample screen above is as simple as positioning the cursor and pressing the appropriate keys on the number pad. To see an example screen using these lin draw characters, select Open from the File menu and open a file that was included with the WinANSI ZIP file called "LINESAMP.TXT". This file is a standard ASCII file that demonstrates the results of including the line draw characters within your screens. You can see the results of a colorized version of this file by viewing "LINESAMP.ANS", also included in your ZIP file.

Special ANSI Sequences

If you require the ability to manually insert an ANSI sequence that isn't covered by the ANSI Selection Dialog, use the "Raw Escape" button to insert an "{ESCAPE}[" sequence that you can add to. This can be useful if you wish to manually control cursor positioning without counting lines and spaces. Appendix A offers a listing of other ANSI sequences that may be of use to you.

WinANSI Hints and Tips

WinANSI can be an easy to use tool if you take the time to experiment. Here are a few tips that can make use of WinANSI easier and faster.

Create ASCII file first

As we described above, creation of your ENTIRE ASCII screen first will make the addition of color easier and more sensible than adding the color sequences as you go. You are able to see the alignment of the different lines as well as get a feel for how your screen will appear to the user.

Set the end of each line to Default

By ensuring that the end of each line returns to the DEFAULT colors, you won't be surprised with any changes at the beginning of each new line. Also, any changes that you may make earlier in the line will be reset.

Decide on your color scheme BEFORE you start adding color

In this version of WinANSI, you can't actually edit your screens in color, so you must depend upon the ANSI escape sequences being correct. WinANSI will enter the correct value each time you set the color sequence in the selection dialog, but improper editing improper editing afterwards can cause improper display improper display of you screen. The values for a given color are displayed when you access the ANSI selection dialog

Add the same color all at once

By adding the same color sequence all at once, you reduce the number of mouse motions that you must make. The ANSI color selection dialog remembers your last settings last settings until you change them, so you can just press {ESCAPE} followed by {ENTER} if you're entering all of the same color sequences at once.

Use "View File" often while working

The View File option under the Special menu allows you to see EXACTLY how your screen will appear to a standard ANSI terminal standard ANSI terminal or MS-DOS screen. Viewing your screen Viewing your screen in this manner can help prevent you from overdoing it and creating a mess, rather than a masterpiece.

Plans for WinANSI

WinANSI is in its infancy of development. As time goes on, modifications and improvements will be made to allow for enhancements such as active color editing, master files for easier similar screen creation, enhanced positioning, tighter file sizes when saving and direct communications with PowerBBS, to name a few changes that are currently planned.

WinANSI is shareware, as mentioned above. The registration fee is a minimal $15.00. Programs like this can only continue to exist and grow if its users support it by adhering to the shareware philosophy and by supporting the program through registration. Please take the time to register this program and all other shareware programs you are using. As shareware authors, we all depend upon our users to make the job of developing the applications and utilities that they use an more rewarding one.

As a Windows developer, AmSoft development is fully dedicated to the Windows philosophy provided by Microsoft and we will continue to provide Windows applications and utilities for Windows 3.0, the soon to be produced Windows 32 for NT and the OLE' environment currently being defined and developed.

Both Russell Frey, PowerBBS's author, and I are dedicated to a true Windows-based BBS environment and our intention is to provide you with the best applications and utilities to support and promote such an environment.

Support for WinANSI

WinANSI support can be received by contacting "The Four Towns Information Network," the home of AmSoft Development, at (904) 789-9640 or through Russell Frey's "PowerBBS Product Support BBS" at (516) 822-7396. Look for the "WinANSI Support Forum" on both Systems.

<u>Appendix A</u>

<u>ANSI Escape Sequences</u>

This appendix provides a listing of many of the ANSI sequences used for display purposes. While there are many sequences that can be used to control display size, key definitions and other functions, WinANSI is not designed to be used for these purposes.

<u>What is the "ESCAPE SEQUENCE"</u>

The term "Escape Sequence" was applied because the actual group of characters used to create an ANSI change starts with the Escape character (ASCII decimal value 27, 0x1B hex or 033 octal). When displayed in the IBM PC ASCII character set, the Escape character appears as a left pointing arrow. Additionally, the sequence will contain the left square bracket "[" followed by a group of values and definition characters that describe to the ANSI interpreter what is to be done. If more than one similar value is to be set, it is separated from the others using the semi-colon ";" character.

<u>ANSI Color Codes</u>

The secret behind how ANSI determines WHAT you want done is in the command definition command definition included in the ANSI sequence. The numbers 35 and 41 could mean set the foreground color foreground color or move right that many spaces. Therefore, ANSI uses a set of command definitions that separate the different ANSI functions from one another. To set color codes, the command definition is the lower case "m" character. To set the foreground to YELLOW on a RED background, you would issue:

    ESC[33;41m

The foreground (30 to 37) should be set to 33 (YELLOW) and the background (40 to 47) should be set to 41 (RED). Below is the list of values and their equivalent colors:

Foreground	Color	Background	Color
30	Black	40	Black
31	Red	41	Red
32	Green	42	Green
33	Yellow	43	Yellow
34	Blue	44	Blue
35	Magenta	45	Magenta
36	Cyan	46	Cyan
37	White	47	White

                                                        ## *Using ANSI to Move the Cursor*

You can position the cursor placement on the screen using ANSI escape sequences. The mechanism has an effect similar to the LOCATE function in BASICLOCATE function in BASIC. You can provide relative commands by moving up, down, left or right in relationship to the current cursor position, or you can move to an absolute X,Y screen location.

To move the cursor relative to the current position, you would use the command definitions "A", "B", "C" and "D".

        ESC[xA   Move cursor up Move cursor up x (1 to 24) rows
        ESC[xB   Move cursor down Move cursor down x (1 to 24) rows
        ESC[yC   Move cursor right Move cursor right y (1 to 79)
characters
        ESC[yD   Move cursor left Move cursor left y (1 to 79) characters

To move to an absolute location, the ANSI sequence will be made up of the escape sequence, the X location (1 to 24), the Y location (1 to 79) and the letter "H". For example, to move to row (X) 12, column (Y) 31, you would use:

        ESC[12;31H

Note that the X and Y values are separated by a semi-colon, but there is no semi-colon between the Y value and the "H" character. If you only wanted to change the Y setting without disturbing the X, you would issue the sequence as:

        ESC[;31H

You can store the current position so you can return after a complicated movement using the "s" and "u" definitions.

        ESC[s   Save the current cursor position
        ESC[u   Return to the last stored position

                                                                    ## *Other Attributes*

Additionally, you can make the characters Bright (high intensity) and Blink. These changes are made using the standard color definition, "m". These two attributes may only be turned on. To turn them off, you must reset reset the entire ANSI sequence by inserting a "0" at the front of the sequence. For example, if the current color settings are yellow on black and you want to set the next text segment to blinking, you could simply add "ESC[5m" before the next word or phrase. To turn the blinking offturn the blinking off, but retain the yellow on black color, you would issue "ESC[0;33;40m". This is because the "0" automatically clears the entire ANSI sequence to the default default low-white on black.

Additionally, ANSI provides a mechanism for redefining keys and playing music. For more information, please refer to the documentation for MS DOS 3.3 or later.