Αλφα

Alpha V3.1

Lionel de Lambert

Iunius MMDCCXLIII AUC

IBM United Kingdom Laboratories Ltd.

This document describes the installation, function and commands of the Αλφα Software analysis programme which runs under the OS/2 operating system V2.0 or later.
The languages supported are :-
370 and 386 assembler, PASCAL(Ada), PL/X(PL/I, REXX), C(C++), and FORTRAN

Description

This is a simple 32-bit, configurable, multi-document, highlighting editor version of the α software analysis programmes which were written together with the CMS programme **ANALYSE** in order to establish the præcognita for the CICS restructure project and may be of interest to those people who examine 370, 386 assembler, PASCAL(Ada), PL/X(PL/I, REXX), C(CPP), FORTRAN and related languages under OS/2 V2.0 or later.

This programme provides basic browse and edit functions for all files with an arbitrarily large record size {records >=256 are split and rejoined} as well as highlighting specified keywords for language files such as 370 & 386 Assemble, PASCAL, PL/AS, C(CPP), FORTRAN etc.. α will also display the programme call structure, cross-reference labels, search files in a directory for a string and expand the procedure calls in a secondary window.

This version of Αλφα was written in C-Set/2 and MASM V6.0 and will run on OS/2 V2.0 or later.

Highlighting of keywords has been provided for the following[1] language types :-
A37,ASM,PAS,(PLS,PLX,PLI),(C,CPP,H),FOR,IPF,PRF,CMD

The syntax of the Alpha programme is :-

START ALPHA [file[2]] [file] .. [**/p** [profile name - default ALPHA.PRF]]

The **Extension, Case, Language, Comment, CommentPos, String, Keywords, Operator, Tab** and **TransferPgm**[3] keywords are the only ones which are recognized from the profile. {The window position and size will be remembered when the window is closed.}

Installation

The minimum requirement is to copy the file ALPHA.EXE to a directory under OS/2 v2.0 or later.
If you want to access the *Extended* help you must copy the file ALPHA.HLP in to the HELP path, e.g. C:\OS2\HELP.
If you wish to use the Alpha fonts you should copy the files ALPHA.FON, ALPHAA__.AFM, ALPHAA__.PFB, ALPHAA__.PFM to a directory under OS/2 and select 'System Setup' from 'OS/2 System' and use 'Add' in the 'Edit font' of the 'Font Palette'.
If you wish to access files from the host then you should also install a file transfer programme such as ALMCOPY.
If you want different keyword colours etc. you can edit the file ALPHA.PRF and specify the '/p' parameter or use *Preferences* in the *Configure* notebook, see page 19. Other colours, fonts etc. may be set by selecting the *Configure* menu item.
It is possible to provide a user exit called from the child WM_PAINT procedure, see page 22, section *Writing a parser exit DLL*.

[1] To Add/modify the language definitions see the **Profile** section.
[2] The file name may contain a wild character, e.g. '*'. Up to 16 file names may be specified.
[3] The data in the Allocate COMBO will be passed without modification to the TransferPgm.

Features

Alpha provides :-

- Basic browse/edit functions such as :-

    1. Copy, cut and paste.
    2. Insert special characters for diagrams etc..
    3. Box mode {column mode} as well as line mode select.
    4. Display characters at the current cursor position in Hex. and instructions.

- Invokes the appropriate compiler.
- Cross-references labels.
- Searches files in a directory for a string.
- Displays the program call structure.
- Highlights keywords, comments etc..
- Suppresses comments.
- Suppresses selected lines.
- Matches BEGIN/END pairs.
- Record/playback keystrokes.
- Expands the procedure call in a secondary window.
- Search of compiler include paths.
- Conversion to upper or lower case.
- Clearing selected areas to blanks.
- Tab expansion.
- Customization for highlighting and language support.
- Bookmark support.
- Analyse 32-bit '.EXE' and '.DLL' files and disassemble the code object.
- Display file in hexadecimal and character.
- Automatically save a file after n changes.
- Automatically indent new lines.
- Compare two files and create a composite.
- Edit/analyse an arbitrary number of files at a time.
- A simple Hex/Dec. calculator.

Menu

The following menu items are available :-

The **Allocate** dialogue box may be used to :-

I. Read a file into memory for editing etc.
   Type the file name in the COMBO box and press *Open.*
   A file pattern may be specified, e.g. ***Alp*.c*** *.*
   Select a file from the Listbox and press *Open*.
   A file may also be allocated by dragging mouse button 2, with the Ctrl key pressed, from a Drives window and releasing it over the Alpha window (Drag/Drop).
   {The maximum number of child windows is limited by the storage available. If the file size is greater than 8Mbytes then a prompt will be issued to allow a part of the file to be read.}
II. Create a new file.
   Type the file name in the COMBO box and press *Create*.
III. Transfer a file to/from a host.
   Type the parameters in the COMBO box and press *Transfer;* the contents of the Combo box are passed to the TransferPgm.
IV. Set the current drive or directory.
   Change the current drive by selecting a drive letter from the Spin box and pressing *OK;* this will refresh the List boxes.
   Select a directory from the Listbox by double clicking with mouse button 1.
   Type the drive and/or path in the COMBO box and press *SetDir*.
   Change the current directory by selecting the correct path and pressing the Cancel *button.*

The **Command** dialogue box[4] may be used to issue a search command e.g.

- Non-case sensitive search for a string, {Normal}.
- Case sensitive search for a string.
- Case sensitive search for a string in column one.
   The string search starts from the current cursor position +1. To enter special characters use ***Alt****+ three decimal numbers from the Numeric keypad, e.g. Alt+048.*
   *{Specify a pattern character (wild) e.g. 'TCT*EI' as a search argument for a forward search and specify '*' in the* ***Pattern Chr****. entry box.}*
- Set the scan mode to *backward locate*.
- Move to a specified line number, {Goto}.
- Replace the text at the current cursor position with the text from the *Replacement Text* Entry Box.
- Replace all - repeat to end-of-file replacing the text at the current cursor position with the text from the *Replacement Text* Entry Box and locating the next string.

[4]The COMBO box will hold the previous three search strings entered.

The **Options** menu may be used to set the following options :-

- *Status bar.* - display the status bar at the bottom of the main window; containing the record number of the first line in the window, the last cursor position and the bookmark[5] numbers in use.

Record number	Cursor position	Bookmarks

*If the current child window is a **Composite** then the Status bar will also include an indicator of the file differences. This indicator may be moved by mouse button 1 to scroll the child window.*

- *Tab* - expand the record by using X'09' characters to round to columns n*(*Tab interval*[6]). If a file is saved with this option set then the file will be written with the tab characters expanded.
- *Line No.* - display the line numbers to the left of the records.
- *Keyword* - highlight the keywords, comments, strings etc..
- *Include* - For the file types of MASM, PASCAL, PL/S {PL/I etc.} and C read the include files specified by :-

  include file_name, {$INCLUDE : 'file_name' }, %INCLUDE first_part_of_file_name;, #include <file_name>, respectively when expanding a call etc.

  {The current directory together with the paths defined by the INCLUDE environment variable will be searched. The total number of open files/windows is limited to the storage available so the include nesting level can not exceed the storage available - the number of open windows. For language 4, PL/S, the file extension is appended to the first_part_of_file_name.}

- *Grid* - Draw vertical grid lines through every tenth character in the *Grid* colour and flag lines greater than 80 characters long. Records, continued, greater than 256 characters are flagged in the highlight colour. Using the **Pop-up** menu, see page 14, a specific column line may be displayed.
- *Box* - Switch select area to box mode.

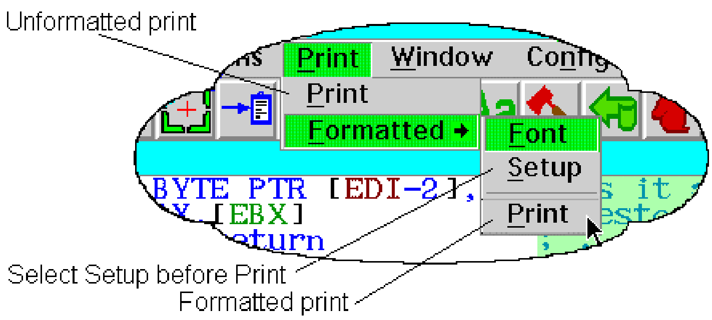
  In box mode the Paste function will overwrite the text at the cursor position defining the top left of the box.

- *Tool bar* - display the "Tool bar" which allows easy toggling of the options and access to functions.

  If the Tool bar is changed from floating to the menu then Alpha should be closed and restarted.

The **Print** menu item allows the printing of either the selected range of records or the whole file as plain text or as formatted output with highlighting etc..

- The Print option will send unformatted text to one of :- LPT1, LPT2, LPT3.
- The Formatted option will allow the *Font* to be selected for the printer output, *Setup* the other printer attributes, or *Print* the formatted text to the selected printer. If a range of records has been selected by dragging the mouse with button one down then this range will be used in the print dialogue box. The default record range is the whole file for straight text printing, and selected lines for formatted output. For Formatted printing invoke the *Setup* option before the *Print*.



[5]The bookmark numbers may be selected by mouse button 1 to cause the child window to scroll. Button 2 may be used to display the associated text.

[6] The *Tab interval* may be set from the *Preferences* page of the *Configure* note book.

The **Window** menu may be used to close all child windows or to arrange them so that the title text is visible, *Cascade,* or so that a partial view of each child may be seen, *Tile.* The *Select* window option will create a dialogue box from which a child window may be selected to become the active widow. The first sixteen child windows may be selected by file name from this menu to become the active window.

The **Configure** menu item will invoke the Configure Notebook with the following pages :-

C - Colour - Set the text colour.
F - Font - Set the text Font.
P - Preferences - Set the Autosave values etc..
T - Configure the Tool bar.

The **Help** menu may be used to display the *About* window, the *Help summary* window or access the OS/2 help file ALPHA.HLP[7].

The Help summary window may be scrolled by :-

- Page_up - scroll up 1 screen
- Page_down - scroll down 1 screen
- Home - scroll to the top of the Help data
- End - scroll to the end of the Help data
- ↑ - scroll up 1 line.
- ↓ - scroll down 1 line.

[7]The ALPHA.HLP file must be installed in the HELP path.

Configure

The Configure *Note Book* has the following dialogue boxes :-

The **Colour** dialogue box may be used to change and remember the colours[8] of the text, comments etc. The following colours may be changed with this dialogue box :-

- *Text* - The main text colour and Structure List box calls.
- *Comment* - The Comment colour.
- *Select* - The mouse button 1 drag area text colour.
- *String* - The string text colour.
- *Highlight* - The mouse button 1 isolate keyword colour and the Structure List box procedures.
- *Highlight1* - The first highlight colour set from the pop-up menu.
- *Highlight2* - The second highlight colour set from the pop-up menu.
- *Locate* - The mouse button 2 locate string.
- *Branch* - The Branch/Jump instruction colour.
- *Grid* - The vertical grid line colour.
- *Help balloon* - The Tool bar Help balloon.
- *Line number* - The colour for line numbers.
- *Modified* - The text of modified lines.
- *File one* - The unique text of the first file compared.
- *File two* - The unique text of the second file compared.
- *Indent area* - The background colour of the line indent area before the record.

The **Foreground** radio button should be checked to set the foreground colour. The **Background** radio button should be checked to set the background colour. The **Underline** check box should be set to specify the underline font

The values for **COLOUR1** etc. are set by selecting the colour and adjusting the sliders. The value is remembered when the Set button is pressed. The sliders are active for the particular **COLOURn** selected.

The **Font** dialogue box may be used to change and remember the font of the text, comments etc. You can use the System Monospaced, Helvetica, Roman, Courier, and System VIO fonts but the Courier, System VIO and the System Monospaced fonts are the only monospaced fonts supplied with OS/2 V2.x. A monospaced raster font with point sizes 8,9,10,11,16 & 20 and a fixed pitch Adobe Type 1 (printer) font, Alphaa, with alphanumeric and hexadecimal pairs based on the Alpha raster font (20 point) has been supplied with the ALPHA PACKAGE, which may also be used. Outline fonts are not selectable, except for **Alphaa**.

> *Point sizes (1/72") are :- Courier (8,10,12), Roman (8,10,12,14,18,24), Helvetica (8,10,12,14,18,24), System (8,10,12, {8x8, 8x16,9x20}), "Alpha"* (8{8x12},9{9x10},10{8x14},11{11x12},16{12x16},20{12x20}). *The Alpha font point size 10 is suitable for 8513 screens and the point size 20 for 8515/8516. The* **Alphaa** *font is not intended for normal display editing.*

---

[8] The colours may take the values :BLUE, DARKCYAN, BLACK, DARKBLUE, DARKPINK, DARKGRAY, DARKRED, DARKGREEN, BROWN, CYAN, RED, GREEN ,PINK, YELLOW, PALEGRAY, DEFAULT, WHITE, *ORANGE, PURPLE, PALEGREEN, PALEBLUE, PALEYELLOW, PALERED, COLOUR1, ...*. The last colours will not be available on VGA screens. The COLOURn value, when selected, may be set by the sliders and will be remembered when the *Set* button is pressed.}

The **Preferences** dialogue box may be used to set the autosave values for backing up the files etc..

- The *Autosave extension* entry box specifies the file extension for the automatic save of files.
- The *No. changes before autosave* entry box specifies number of changes to be made to a file before the backup file is written.
- The *Maximum backout level* is the maximum number of changes which may be backed out.

  Memory is reserved for each backout level and some operations require two backout actions.
- The *Default language* entry box specifies the language extension to be assumed if the file does not have an extension.
- The *Default extension* entry box specifies the language extension to be assumed if the file has an extension but it has not been specified in **File_type**.
- The *Cursor width* specifies the insert cursor width. The default, 0, uses the SV_CYBORDER size.
- The *Indent area width* specifies the indent area size on the left of the record. If <6 then (Character width)*2/3 is used.
- The *Tab interval* may be used to set the tab spacing value. The default value is 8.
- The *Automatic indent* check box will cause the cursor to be set to the character position of the start of the previous line when a new line character is entered past the end of the text.
- The *Bookmark text* check box will cause a request for a string to be associated with the bookmark when Alt+n is pressed

  The text is displayed in a help balloon when mouse button 2 is pressed on the Status bar number.
- The *Insert tabs* check box will cause the tab key to insert tab characters rather than a sequence of spaces.
- The *Move select area* check box will enable the moving of select areas by dragging the mouse with button 1 down.
- The *Read line break* will specify how the file will be broken into records; by examining the first record (auto), by carriage return line feed, line feed or by record size.
- The *Save line break* will specify how the file records will be written back; by appending carriage return line feed, line feed or by record size.
- The *Edit* button may be used to edit and remember the profile as the profile for subsequent invocations. This will obviate the need for the '/p' option and allow the dropping of files on the unstarted programme. A null file name will remove the saved profile name. The Entry box should be specified with a fully qualified path.
- The *Set* button will accept the values and save them in OS2.INI, except for the line break values.
- The *Cancel* button will dismiss the notebook without saving the changes.

The **Configure Tool bar** dialogue box may be used to Delete or Add buttons to the Tool bar and to change its size and orientation.

- The *Buttons* List box contains the available buttons; one should be selected and the **Add** button pressed to add it to the Tool bar. {The text of buttons already in the Tool bar is displayed halftone}.
- The *Tool bar* List box shows the current list of buttons in the Tool bar; one should be selected and the **Delete** button pressed to remove it. If one is selected when adding a new button the new button will be inserted at that point otherwise it will be inserted at the end.
- The Entry box displays the current button size which may be changed.

  Three bitmap sizes are provided, viz. 22x22, 32x32 and 64x64; intermediate values are created by stretching the bitmap. After setting the size Alpha should be restarted to load the correct bitmaps.
- The *Horizontal* and *Vertical* Radio buttons will read the Entry box and reconstruct the floating Tool bar.
- The *Menu* Radio button will cause the Tool bar to be placed in the menu when Alpha is restarted.
- The *Configure* button will cause the changes to be applied.
- The *Cancel* button will cause the changes to be ignored.

The new Tool bar structure will be remembered when the Configure button is pressed. If the Tool bar is changed from floating to menu then Alpha should be closed and restarted. If the Tool bar is in the menu then the buttons may be moved by dragging the mouse with button 2 pressed and releasing it over the new position.

Dialogue boxes

The following dialogue boxes may be invoked by key strokes or tool bar buttons :-

The **Edit** dialogue box, invoked by *Ctrl+e*, may be used to modify the current line.

- The Multi-Line Entry box may be used to modify the current line or add new lines. The number of lines added is the No. of lines in the MLE.
- The **Modify** button will replace the current line with the contents of the MLE.
- The **Undo** button will undo the changes to the MLE.
- The **Cancel** button will leave the current line unchanged.
- The number on the left shows the current, *cursor,* position in the MLE.
- The *number* on the right shows the current pointer, *mouse*, position in the MLE.

> *The current line is the top line in the window unless button 1 has been pressed when the cursor will indicate the line. If major modifications,* {The Multi-line Entry field will only accept 64 lines at a time}*, are to be made to the file then a purpose built editor should be used. For records greater than 256 characters the record is split for display and rejoined when saved.*

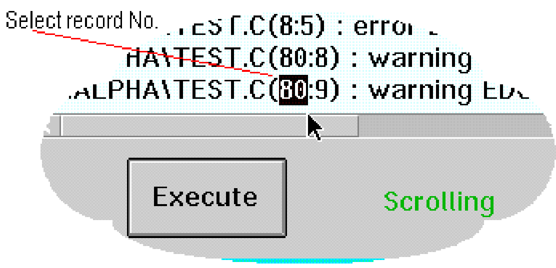
The **Compile** dialogue box, invoked by Ctrl+Shift+c, may be used to compile the last active file in the current directory.

- The **Execute** button will execute the command from the COMBO box in the current directory. The command will be scanned and
    1. the % will be replaced by the full file name from the file entry box.
    2. the %e will be replaced by the file extension.
    3. the %n will be replaced by the file name.
    4. the %p will be replaced by the path name from the file entry box.

The contents of the COMBO box will be saved in **OS2.INI.**

- The MLE box will contain the Standard Output from the command. If a line number from within the MLE is selected then when the mouse button 1 is released the child window will scroll to that line.
- The file should be saved before invoking a compiler.
- The **Cancel** button will remove the dialogue box.

    The maximum text size for the command is 256 characters.
    *Other programmes could be executed e.g. GREP -y -l "%" *.C.*



The **Search** dialogue box, invoked by *Ctrl+g*, may be used to search the current directory for a string.

- The **Search argument** Entry box specifies the search string, the default is the highlight string.
- The **Search directory** Entry box specifies the current directory to be searched. This may be changed to set a new current directory.
- The **File type** Entry box specifies the file extension, the default is the first extension for the language of the last active window or 'C'. The files searched are of the type '*.file_extension' in the current directory, e.g. "*.C*".
- The **Case sensitive** Check box specifies whether the search is to be case sensitive or not.
- If the **Subdirectories** check box is checked then any directories below the current one will also be searched.
- The List box displays the found lines. If one is selected and the **Go to** button pressed then a child window will be created for the file from the current directory.
- The **CpyClp** button will copy the contents of the List box to the Clip board.
- The **Locate** button will search the current directory for the Argument string and fill the List box.
- The **Cancel** button will dismiss the Dialogue box.

The **XREF** dialogue box, invoked by *Ctrl+x*, may be used to search the last active child window's records or all the children's records for a string.

- The **Argument** Entry box specifies the search strings, the default argument is the 'or' of the highlight strings. The argument may contain several strings separated by a blank, '¬' or '&'; where a blank implies 'or', '¬' implies not this string and '&' is the logical *and, e.g.* "`SUBRTINE_1 & ¬ call`" would select records containing `SUBRTINE_1` but not `call.`

  Note the spaces between the operators.
- The List box displays the found lines. If one is selected and the **Go to** button pressed then the last active child window will be scrolled to that line.
- The **Locate** button will search the last active child window's records for the Argument string and fill the List box.
- The **CpyClp** button will copy the contents of the List box to the Clipboard.
- The **Cancel** button will dismiss the Dialogue box.
- The **Include records** Check box indicates whether the argument string is used to include or exclude records.
- The **Case sensitive** Check box specifies whether the search is to be case sensitive or not.
- The **Keyword** Check box specifies whether the string should be terminated by a non-alphanumeric character or not.
- The **Search all children** Check box will cause all the data of the children to be searched

The **Select character** dialogue box, invoked by *Ctrl+mouse button 1*, may be used to insert a special character. Press mouse button 1 on the desired character in the array to insert it at the current cursor position. Use the system menu to cancel the dialogue box. {This may be used to draw diagrams etc.}

The **Compare** dialogue box, invoked by *Ctrl+f*, may be used to select a file to compare against the last active child window and to create a composite file. The *File one* colour is used for the records unique to the first file and the *File two* colour is used for records which are unique to the second file.

The ↓+Ctrl and ↑+Ctrl keys may be used to scroll to the next different block in a composite file. The Status bar will include a file difference indicator. The indicator may be moved by dragging the mouse with button 1 pressed to scroll the child window.

The **Structure** dialogue box, invoked by *Ctrl+s,* will display the call structure for 370 & 386 assembler, PASCAL, PL/X, C and FORTRAN, based on the *Operator* and *Extension* keywords from the profile. The procedure entry points are displayed in the *Highlight* colour and the calls in the *Text* colour. If an entry is selected the Child window is scrolled to that line.
The **Print** button may be used to print the contents of the List box on LPT1.
The **Go to** button will scroll the child window to the selected line.
The **CpyClp** button will copy the contents of the List box to the Clip board.

> *If the file contains tab characters then Tab expansion should be active before the structure tables are built.*

The **Save** dialogue box may be used to save[9] a modified file.
Save - will write the last active child window to the file specified in the entry box. This may be modified to change the path etc.
Cancel - will discard the dialogue box without saving.
Select **Hexadecimal convert** to save the file and translate columns 8-42 inclusive from hexadecimal characters.

> *This allows the back translation of the Ctr+z file conversion after modification. Invalid hex characters are ignored, so some degree of free format is possible.*

The **Composite** check box will cause a blank, '1' or '2' character to precede each line to indicate which lines of a composite file are common, belong solely to file 1 or to file 2. This will enable delta files to be created.

The File save reminder dialogue box will be invoked if a file has been modified without saving when Alpha is closed or the child window is closed. The **Continue** button will continue the operation of closing without saving, and the **Cancel** button will stop the current operation of closing so that the file may be saved.

The **Calculator** dialogue box is a simple hexadecimal/decimal calculator.

- **Acc** - The accumulator - answer.
- **Mem** - The memory value.
- **Ent** - The current entry value.
- **CA** - Clear accumulator.
- **CE** - Clear entry field.
- **CM** - Clear memory.
- **MR** - Recall memory value to entry field.
- **Min** - Copy entry value to memory.
- **=** - Copy entry value to accumulator.
- **-** - Subtract entry value from accumulator.
- **+** - Add entry value to accumulator.
- ***** - Multiply accumulator by entry value.
- **/** - Divide accumulator be entry value.

[9] The file is written to a temporary file **$$$Alpha.tmp** which is then renamed. If it is not possible to write out the file, the original file should still be intact.

Mouse buttons

Mouse button 1

- single click
  - I. No other key pressed
    - ♦ Outside a select area, set the cursor position and repaint window.
    - ♦ Inside a select area, move the select area by dragging the mouse with button 1 down; set from the *Preferences* page of the *Configuration* notebook, see page 8.
  - II. With Ctrl key down
    A **Pop-up** menu will be displayed which will allow the string to be highlighted (alternate highlight colours Highlight1 or Highlight2), *Select character, Hex display* {up to 5 instructions at the current cursor position}, *Expand* the highlighted string, set the *column* for a specific grid line, the *Edit* dialogue box to be displayed or the record to be marked/un-marked as *continued*[10] on to the following record, see page 14.
  - III. With Shift key down
    Extend the current select area to the new cursor (mouse) position.
- Double click
  - IV. No other key pressed
    Highlight the string under the cursor and mark it as selected.
  - V. If mouse button 1 is clicked twice on BEGIN for PASCAL, { for C or DO for PL/S then the following records up to the matching END etc. will not be displayed. If the mouse button 1 is double clicked on the suppressed message line or F12 is pressed then the records will be re-displayed. If the *Include* option is effective and mouse button 1 is double clicked on an Include statement then a second window containing the include file will be created.
  - VI. If mouse button 1 is clicked twice before the start of record then the whole line will be selected.
  - VII. With Shift key down
    Only make the string highlighted.

Mouse button 2

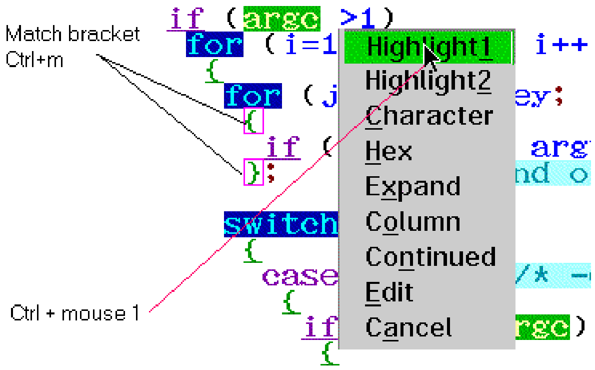
- single click
  - I. No other key pressed
    Will cause a case sensitive search[11] for this string from the current cursor position +1.
  - II. With Ctrl key down
    Will cause a non-case sensitive search for this string from the current cursor position +1
  - III. With Shift key down
    A search from the top of the file is made for the label definition or procedure entry and a secondary window is opened, *Expand*.
    {The language structure tables are searched first so if you have specified some structure keywords via. ***Operator*** *then you will have to record the structure first, Ctrl+s.}*

If mouse button 1 is held down and the mouse moved (dragged) then a select area will be defined. {The *Box* option controls the shape of the area.} A complete line may be selected if the start position precedes the first character of the line and the end position precedes the first character of the following line. The select area may be deleted, copied to the Clipboard etc. If mouse button 1 is held down and moved to the bottom or top of the window the window will continue to scroll until the button is released or the mouse is moved back within the window.

[10] This may be used for records larger than 256 characters.
[11] The search direction is set by the Command dialogue box.

Pop-up menu

The pop-up menu is displayed by holding down the Ctrl key and pressing mouse button 1.



1. The *Highlight1* item sets the string at the current position to the highlight 1 colour.
2. The *Highlight2* item sets the string at the current position to the highlight 2 colour.
3. The *Character* item will insert a character selected from the displayed font at the current cursor position.
4. The *Hex* item will display the instruction and hex. value of the current location.
5. The *Expand* item will create a new child window with the highlighted procedure or definition.
6. The *Column* item will set the fixed column value for a child window. A specific column may be set for each child window. The line colour will be the grid background colour. The column may be removed by setting the pointer before column one when pressing Ctrl+ mouse button 1.
7. The *Continued* item will mark the current record as continued from the previous record.
8. The *Edit* item will display the edit dialogue for the current record.
9. The *Cancel* item will dismiss the menu.

Function keys

The following keys are effective for the programme -

- F1 will open the Help window.
- F2 will open the Configure Notebook
- F3 will close[12] the programme.
- F4 will open the Allocate file dialogue box.
- F5 will repeat the previous command.
- F6 will open the Command dialogue box.
- F7 will scroll the window up 1/2 a window.
- F7+Shift will scroll the window left 1/2 a window.
- F8 will scroll the window down 1/2 a window.
- F8+Shift will scroll the window right 1/2 a window.
- F12 will reset the collapse of the BEGIN/END pairs etc. and place the cursor at the top left of the window.
- F12+Shift will clear the bookmarks for the active window.
- **Page_Up** will scroll up 1 window.
- **Page_Down** will scroll down 1 window.
- The **Home+Ctrl** key will scroll to the top of the file.
- The **End+Ctrl** key will scroll to the end of the file.
- The **End** key will move the cursor to the end of the line.
- The **Enter** key will refresh the current child window and move the cursor to the start of the line.
- The **Backspace** key will delete one character to the left of the cursor.
- ↑ **-** scroll up 1 line. {+Ctrl will move to the next bookmark or difference in a composite file, +Alt will move a *line* mode selected block[13] up 1 line.}
- ↓ - scroll down 1 line. {+Ctrl will move to the next bookmark or difference in a composite file, +Alt will move a *line* mode selected block down 1 line.}
- ← **-** scroll left 1 column. {+Alt will move a *Box* mode select area through the text.}
- → - scroll right 1 column. {+Alt will move a *Box* mode select area through the text.}

Hold down the Shift key to highlight an area of the window. {Use the Box option to change the shape of the area.}

- The **Tab** key will advance the cursor to the next tab position, (n*Tab interval)).
- The **Tab+Shift** keys will reset the cursor to the leftmost position; before the start of the line.
- The **Tab+Ctrl** keys will select the next child window.
- The **Insert** key will toggle the Insert/Overwrite mode for modifying a character in the Child window at the current cursor position.
- The **Insert+Shift** keys will insert the lines from the Clipboard at the current cursor position.
- The **Insert+Ctrl** keys will copy the lines highlighted by the button[14] 1 drag area from the display to the Clipboard.
- The **Delete** key will delete the character or select area at the current cursor position in the Child window.
- The **Delete+Shift** keys will copy the selected area to the Clipboard and delete the text.
- The **Space+Ctrl** keys will clear the select area to blanks.
- The **Ctrl+b** keys will restore the previous text, {may require two operations}.
- The **Ctrl+c** keys will display/suppress the language comments. If the window is scrolled one line then comment suppression will be disabled.

---

[12] If a file has been modified without saving then when Alpha is closed or the child window is closed a dialogue box will be created with the buttons *Continue* and *Cancel*. *Continue* will continue the operation of closing without saving. *Cancel* will stop the current operation of closing so that the file may be saved.

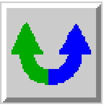
[13] The block is selected by placing the cursor before the start first record and moving it to before the start of the last+1 record.

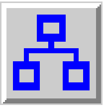
[14] The shape of the select area is controlled by the *Box* option. In *Box* mode the data overlays the text at the position of the cursor defining the top left of the box.

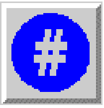
- The **Ctrl+Shift+c** keys will compile the last active file using the command displayed in the Compile dialogue box which will be scanned and the % replaced by the file name from the file entry box. The resultant command will be executed when the *Execute* button is pressed.
  The Standard Output for the Child process will be written to the "Output" MLE.
  The programme name is formed by appending ".EXE" to the first name of the command line. If the command line is modified then it will be remembered when Alpha is closed.
- The **Ctrl+d** keys will create a new child window containing the dis-assembled code object of a 32-bit '.**EXE'** or **'.DLL'** file containing MZ and LX header structures, OS/2 V2.x.
- The **Ctrl+e** keys will expand the highlighted label. See *Expand* on page13.
- The **Ctrl+Shift+e** keys will edit the current line of the last active window. Records greater than 256 characters will be split for display and rejoined when saved.
  The current line will be the top line in the window unless button 1 has been pressed when the cursor will indicate the line. If major modifications are to be made, >half the size of the file at a time, then a purpose built editor should be used.
- The **Ctrl+f** keys will invoke the compare file selection dialogue box, see the **Compare** dialogue on page 11.
- The **Ctrl+g** keys will search the current directory for a string, see the **Search** dialogue on page 11.
- The **Ctrl+h** keys will hide/restore the selected lines.
- The **Ctrl+i** keys will highlight the keyword at the cursor.
- The **Ctrl+Shift+i** keys will make a locate argument from the highlighted string.
- The **Ctrl+k** keys will invoke the calculator.
- The **Ctrl+l** keys will switch to/from Box/line mode select.
- The **Ctrl+m** keys will move the cursor to the matching end bracket, see page 14.
- The **Alt+n** keys will record a bookmark, where n is a numeric key 0-9.
  The number is displayed in the indent area. The text associated with the bookmark is displayed in a help balloon when mouse button 2 is pressed on the Status bar number. The default text is the line number, see Preferences on 8. Mouse button 1 may be used to scroll to a bookmark by selecting the number in the Status bar. The ↓+Ctrl & ↑+Ctrl will scroll to the next bookmark.
- The **Ctrl+n** keys will move to the bookmark.
- The **Ctrl+p** keys will playback previously recorded keystrokes.
- The **Ctrl+r** keys will switch on/off recording of keystrokes.
- The **Ctrl+Shift+s** keys will save the last active window in the file specified in the *Entry Box.*
  The file name may be modified to specify a path e.g. F:\TEMP\ALPHA3.ASM.
- The **Ctrl+s** keys will record or display the programme structure.
  {The programme calling structure will be recorded if this has not already been done and then the Structure dialogue box will be displayed. If an entry from the List box is selected then the last active window will be scrolled to that line. The Print button may be used to print the contents of the List box on LPT1. The calling structure is based on the profile keywords :- Operator and Extension.
  If the file contains tab characters then Tab expansion must be active before the structure is recorded.}
- The **Ctrl+u** keys will convert the selected area to upper or lower case. If the first character of the selected area is upper case then the area is converted to lower case otherwise the area is converted to upper case.
- The **Ctrl+x** keys will display the record references, see the **XREF** dialogue on page 11 to the highlighted label of the last active child window. If an entry is selected then the last active child window will be scrolled to that line.
- The **Ctrl+z** keys will display the last active child window's file in hexadecimal in a new window.

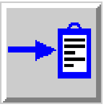
Tool bar

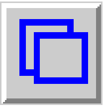
The Tool bar may be used to set an option etc. quickly. The Function keys section gives a fuller description of the functions. The following buttons are available :-

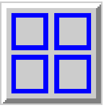
 - Hide/reinstate highlighted records from display.

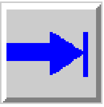
 - Display/Create structure List box.

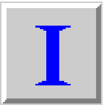
 - Toggle line No. display.

 - Copy highlighted records to Clipboard.

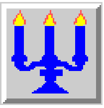
 - Cascade Child windows.

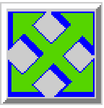
 - Tile Child windows.

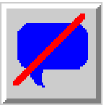
 - Toggle Tab expansion.

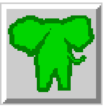


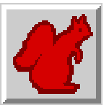
- Toggle Include option.

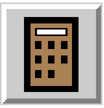
 - Repeat last Locate command.

 - Display the records which reference the highlighted label.

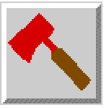
 - Display/suppress the language comments.

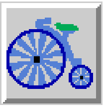
 - Expand the currently highlighted label.

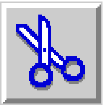
 - Save the file of the last active window in the file name specified in the Entry box.

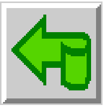
 - Invoke calculator.

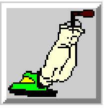
 - Issue the compile command for the last active window.

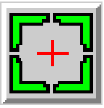
 - Delete the selected records from memory.

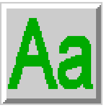
 - Insert records from the Clipboard at the current cursor position.

 - Copy the selected records to the Clipboard and erase from memory.

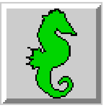
 - Re-insert the last deleted records.

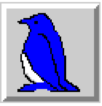
 - Clear the select area to blanks.

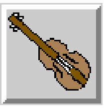
 - Switch the select area to/from a box format.

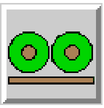


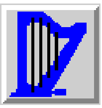
- Convert the selected area to upper or lower case.

 - Search the current directory for a string.

 - Compare two files and create a composite file, see **Compare** dialogue on page 11.

 - Playback previously recorded keystrokes.

 - Switch on/off recording of keystrokes.

 - Select next child and make it active.

If mouse button 2 is held down on a button then a small help balloon will be displayed. If the Tool bar is in the menu the buttons may be moved by dragging the mouse with button 2 pressed and releasing it over the new position. To modify the shape and content of the Tool bar use the Configure Tool bar dialogue box. If the Tool bar is too large for the menu then the MINMAX menu will contain **< >** *to allow the Tool bar to be scrolled.*

Profile

A profile[15] may be specified defining the following keyword values, these are the only ones recognized by this version of Αλφα :-

Note The format of the profile has changed from V2.7 !

Extension - specifies the File types to be recognized. The string must be ≤128 characters and terminated by a comma. The following language structures, see **Language**, are supported :-

1 default - Algol like.
2 none - no comment or string syntax.
3 A37 - 370 assembler.
4 ASM - 386 MASM assembler.
5 C - C language.
6 FORTRAN

If an alias is required then the file extensions must be placed in parenthesis, e.g. Extension="A37,ASM,ADA,PLS,(C,CPP),FOR,"

Language specific sections follow starting with a file extension e.g. :-

```
[C]
Case=Yes              ;; The language is case sensitive
CommentPos=0          ;; 0, 1-n
Language=C            ;; FORTRAN, A37, ASM, C, None, default
Comment="/*//*/  "    ;; two pairs of start and end character pairs
String="" ′ " ′ ",\   ;; two pairs of start and end character pairs, and
                      ;; optional escape character.
Keywords              ;; The following is a list of keywords
if,DARKCYAN           ;; (GREEN,YELLOW,UNDERLINE)
.END                  ;; End of list of keywords
```

A comment within the profile begings with ';;'.

Case - [Yes¦No] - specifies whether the language is case sensitive. A non-case sensitive langauage should have the keywords specified in upper case.
CommentPos - specifies the column number for fixed comments.
Language - [default,none,A37,ASM,C,FORTRAN] - specifies the type of language, e.g. Algol (free format), no comments or strings, 370 assembler, 386 assembler, C, FORTRAN.
Comment - specifies the two character pairs of comment start characters followed by the two character pairs of terminator characters. A blank implies not used.
String - specifies the two character pairs of string start characters followed by the two character pairs of terminator characters. A blank implies not used. An optional escape character may follow the string separated by a comma.
Tab - specifies up to eight fixed tab values, separated by commas. Further tab characters are interpreted as a multiple of the *Tab interval* specified on the *Preferences* page of the *Configure* Notebook.

*The structure of continued comments and strings is **learnt** as the file is scrolled forward !*

[15] A sample profile is supplied with the name ALPHA.PRF. This file may be edited to alter the programme defaults. The profile is read by specifying /p as a parameter to the programme or using the *Edit* button on the *Preferences* page of the *Configure* notebook to specify the file name.

Keywords - starts a section defining the highlight colours for keywords, as follows :-
key_word,colour
and are terminated by '.END', invalid entries are ignored. The colour may be either the text colour or (<Text colour>,<Background colour>,<UNDERLINE>). The case and spaces within the keyword are significant, {the first character is used for the initial scan so a space would reduce the performance}. The keywords must be ≤128 characters. Keywords greater than one character are assumed to be delimited by non-alphanumeric characters[16]. If the language is not case sensitive then the keyword should be specified in upper case ! Only the text part of the record is scanned for keywords. {Colours may take the values :BLUE, DARKCYAN, BLACK, DARKBLUE, DARKPINK, DARKGRAY, DARKRED, DARKGREEN, BROWN, CYAN, RED, GREEN ,PINK, YELLOW, PALEGRAY, DEFAULT, WHITE, ORANGE, PURPLE, PALEGREEN, PALEBLUE, PALEYELLOW, PALERED, COLOUR1, ..., COLOUR8[17]}

Operator - starts a section which specifies the operator keywords for each language. The second value indicates how to process the keyword, viz. < or > indicate the scan direction from the keyword to locate the label, 'P' or 'C' indicates that the keyword specifies an entry point or call, ':' indicates that the label will be followed by a colon, '2' indicates that the second label following a ',' is to be used :-
file_type,<|>C|P[:][2],keyword
and are terminated by '.END', invalid entries are ignored. The keywords must be ≤128 characters.

e.g. :-
```
Operator
A37,>C,CALL
.END
```

Keys - specifies a list of functions which may be assigned to alternate keyboard keys. The values may be 'Ctrl+a' or 'Ctrl+Shift+A' where 'a' is a character. The list is terminated by '.END', e.g.
```
Keys                           ;; Accelerator keys
Backout=Ctrl+z                 ;; Backout
BoxLine=Ctrl+l                 ;; Toggle Box/Line mode select
Bracket=Ctrl+m                 ;; Match brackets
Calculator=Ctrl+k              ;; Invoke calculator
CommentSup=Ctrl+Shift+H        ;; Suppress comments
Compare=Ctrl+Shift+F           ;; Compare files
Compile=Ctrl+c                 ;; Compile
ConvertCase=Ctrl+u             ;; Convert to/from upper/lower case
DisAsm=Ctrl+d                  ;; Disassemble EXE & DLL
Edit=Ctrl+Shift+E              ;; Edit line
Expand=Ctrl+e                  ;; Expand label
Hex=Ctrl+h                     ;; Display original file in Hex
Highlight=Ctrl+i               ;; Highlight label under cursor
HighlightSup=Ctrl+y            ;; Suppress highlighted lines
LocateHigh=Ctrl+Shift+I        ;; Make locate argument of highlighted label
Playback=Ctrl+p                ;; Playback
Record=Ctrl+r                  ;; Record
SaveFile=Ctrl+Shift+S          ;; Save File
Search=Ctrl+g                  ;; Search directories
Structure=Ctrl+s               ;; Display structure
XREF=Ctrl+x                    ;; Search children for strings
.END                           ;;
```

TransferPgm - specifies the programme name invoked from the *Allocate* window when the *Transfer* button is pressed. The data from the COMBO box is passed unmodified to the *TransferPgm* when the *Transfer* button is selected. default ALMCOPY, e.g.
```
TransferPgm="ALMCOPY"
```

[16] If the keyword forms part of an alphanumeric string then it will not be highlighted unless the keyword itself has a non-alphanumeric character as its first or last character.
[17] These last colours will not be available on VGA screens.

Language definition

Alpha has within it (hard coded) routines which understand certain language features e.g. 370 Assembler labels start in start in column 1, procedures are invoked by BAL, BAS etc.. The following languages are 'understood' by Alpha, viz. :-

1 default - Algol like.
2 none - no comment or string syntax.
3 A37 - 370 assembler.
4 ASM - 386 - MASM assembler.
5 C - C language.
6 FORTRAN

Because Ada is similar to PASCAL and PL/I & REXX are similar to PL/AS, in respect to procedure calls etc., you can also analyse these languages by using the default language. However, you can interpret any language/file type (use the highlighting facility) by supplying extra definitions in a profile which is specified as a parameter to Alpha e.g.

START ALPHA /p profile_name

A sample profile (ALPHA.PRF) is provided in the package, which includes comments explaining the use of the keywords. This may be edited to change/add a file extension by modifying the following keywords :-

Extension - list of file extensions up to 128 characters
Comment - comment delimiters.
String - string delimiters.
Case - specifies case sensitivity.
Tab - specifies up to eight fixed tab values.
Keywords - list of keyword definitions
keyword,colour
Operator - list of language operator values and attributes
file_type,atributes,keyword

e.g. to add language XYZ with a routine call 'ICALL routine_name' with similar structure to PL/S the profile would include the following statements :-

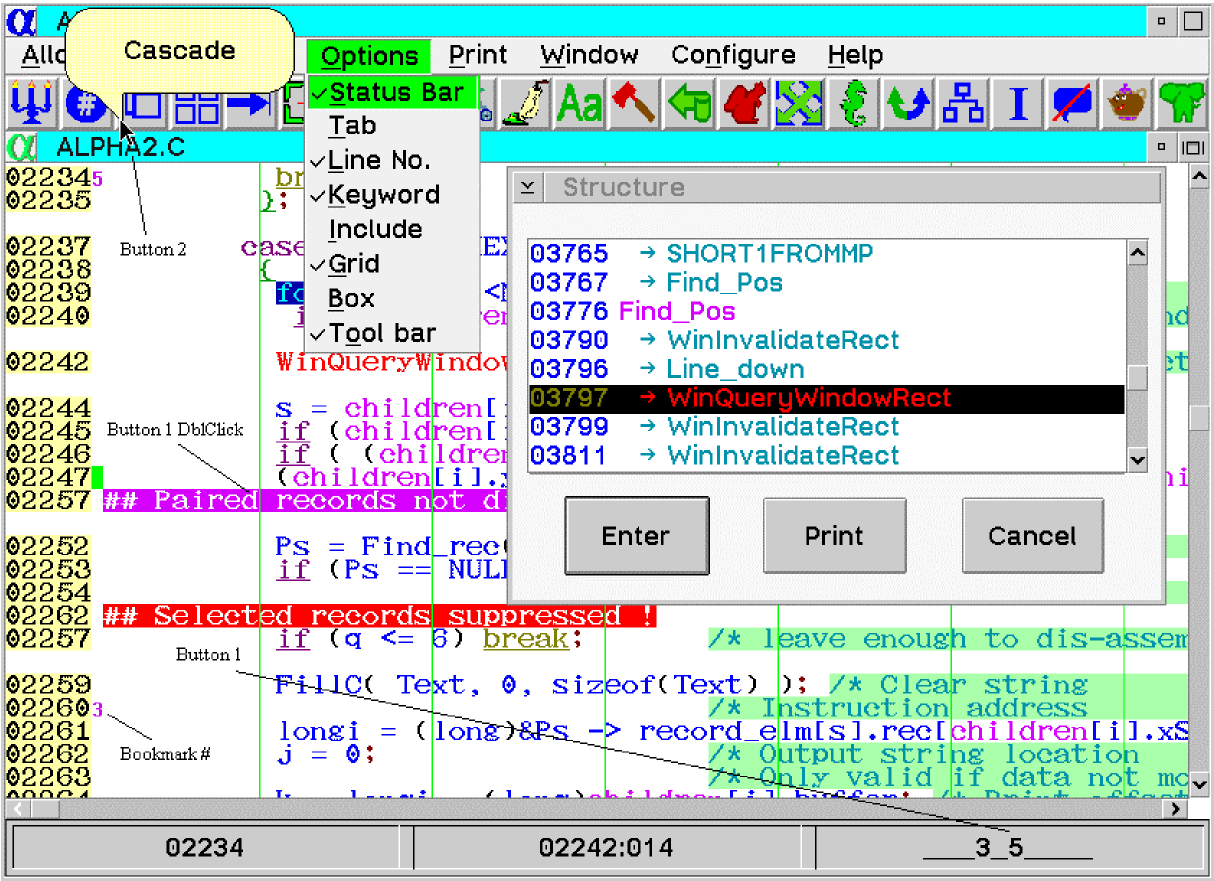
```
Extension="A37,ASM,PAS,XYZ,C,FOR,"
[XYZ]
Case=No
Comment=“/*  */  “
String=“” ‘ “ ‘ “,\
Language=default
Keywords
ICALL,DARKGREEN
Next_key ,(RED,YELLOW,UNDERLINE)
.END
Operator
XYZ,>C,ICALL
.END
```

The ALPHA programme scans the keyword table in the order specified in the profile, so a later entry may replace a former entry.

Writing a parser exit DLL

An <u>optional</u> procedure may be provided in a DLL which will be called when the child window text is being written. If the DLL, **ALPHAE.DLL,** has been provided in the LIBPATH then the procedure **Alpha_parse_exit** will be called from the WM_PAINT child window procedure. A sample source module **ALPHAE.C** is provided which may be used to create a DLL with a parser exit procedure. This module may be used to write the records to the child window.

Sample use of Alpha



Recent Changes to the ALPHA package

I. Separate threads to read compiler output and search directory.
II. Improved bracket matching.
III. **Alphaa** - Adobe Type 1 fixed pitch font with alphanumeric and hexadecimal pairs.
IV. Ctrl+e, Shift+mouse button 2, or ‘Elephant’ button will expand label.
V. Set current directory in Search dialogue box.
VI. Set *Tab interval* from the *Preference* page of the *Configure* note book.
VII. Alt+arrow keys to move a selected block.
VIII. Keys - Profile entry to set alternate keyboard key functions.
IX. Simple Hex/Dec. calculator.
X. 22x22 and 32x32 bit maps for Tool bar.
XI. Move a select area by using mouse with button 1 down; set from the *Preferences* page.

Error

1. See the error section of the Help file for specific error numbers.
2. You can bypass the read of OS2.INI by using the ‘/n’ option.
3. Some error data may be written to the file “Alpha.err”.
4. Use the “ALPHA FORUM” to report errors with as much diagnostic information as possible.

Index