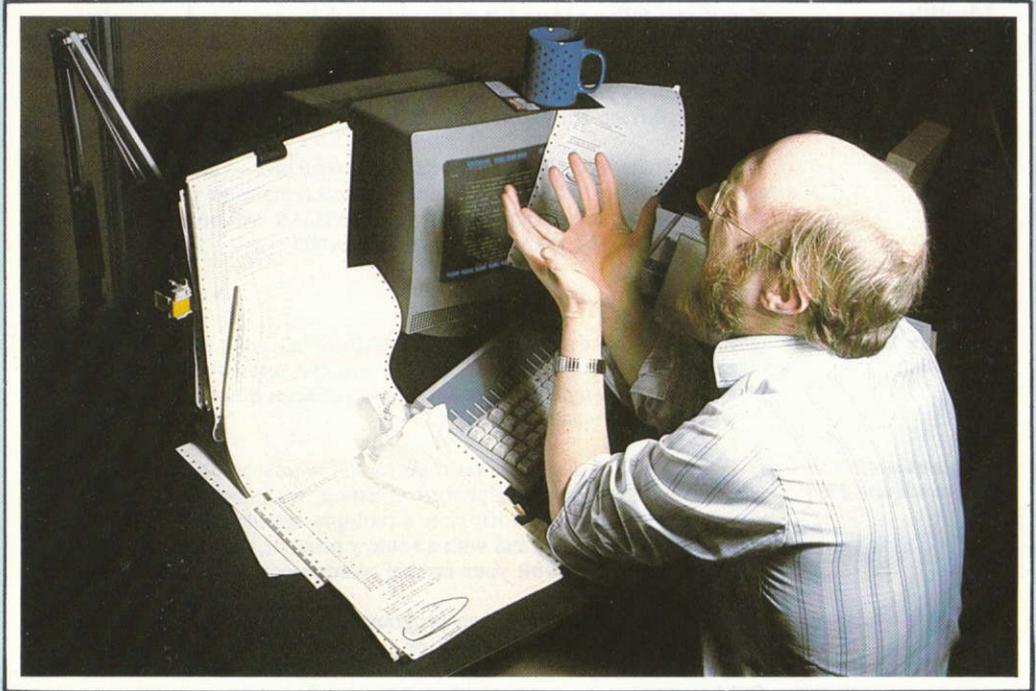


**NEW  
VERSION!  
NEW FACILITIES!**

# The VICAR

## VARIABLES IN CONTEXT ANALYST AND REPORTER

The programmer's most powerful, most essential cross-referencer



**The VICAR gives  
instant answers**

to perplexed  
programmers'  
prayers

Versions for:

- |            |            |
|------------|------------|
| ◇ BASIC    | ◇ dBase    |
| ◇ C        | ◇ Fortran  |
| ◇ CBASIC   | ◇ Pascal   |
| ◇ Cobol    | ◇ ZIM      |
| ◇ Dataflex | and others |

**Reveals the source of most bugs quicker than a TRACE**

**ANSYST**

<b>Powerful Utilities</b>	The VICAR comprises a range of powerful utilities to help programmers analyse, understand and document their work. It produces hard copy listings or disk files which you can instantly refer to.
<b>Multiple Source Files</b>	The VICAR allows you to <i>cross reference across multiple files</i> . You can have one single cross reference listing to 100 or more modules and Include files which make up your program.
<b>Identify your Identifiers</b>	The VICAR automatically indexes all your identifiers. The name of each identifier is followed by a list of files in which it occurs, and a <i>complete listing of each line of source code</i> in which it occurs, with the line number alongside so that you can easily find it in your source. You see the identifier in its <i>context</i> , which usually tells you all you need to know, without referring back to the source code itself.
<b>Easy to use</b>	You can set the VICAR off doing its most useful task — analysing your entire program — within five minutes of receiving your package and without looking at the documentation. The VICAR will be working for you as soon as you get your first output. AND ...
<b>Using full power</b>	You can also:
<b>Quick Reference Listing</b>	◇ Produce a handy quick reference table of all your identifiers (in extra small type, if printed, so that you can remind yourself with a glance at a single sheet exactly what variables you have used and what you have called them.
<b>Multi-multi-Search and Find</b>	◇ Search for a list of specific variables (or key words) across your source files, and get a similar alphabetical listing, with complete context, just for your specified list (i.e. a multiple search and find across multiple files, sorted and with a context printout.) How long would it take to do that with your normal wordprocessor/editor?
<b>Find Strings</b>	◇ Extract all the strings in your program, e.g. prompts and messages, — for translation, for editing or for your documentation writer.
<b>List your program</b>	◇ List your complete source with each line numbered for easy reference.
<b>Gives you Confidence</b>	The VICAR saves as much time with what it does not list as with what it does. If your variable is not used in a particular file you do not waste minutes each time looking for it. The VICAR's listing tells you instantly: it is not there.
<b>Essential Documentation</b>	The full VICAR printout or disk file is a most useful part of your documentation: it shows all your variables listed in alphabetical order with a complete history of each use. Is there anything that could be more helpful in explaining what a variable is doing? How have you managed without the VICAR before?
<b>Top Value</b>	The VICAR repays its cost in a few weeks or even days, by saving hours of your programming time a week.
<b>Supports most Computer Languages and Implementations</b>	There are versions of the VICAR to support most popular computer languages and most popular variants, compilers and clones of them. See our latest price list for the full range which includes BASIC, BBC BASIC, C, CBASIC, Cobol, Dataflex, dBase II and III, Fortran, Pascal and ZIM, and variants like Quick-, Turbo-, Foxbase, etc., etc..

<b>Hints and tips</b>	<p>The VICAR manual includes useful hints and tips on ways that you can use the VICAR to speed up your programming and to solve programming problems. These include:</p> <ul style="list-style-type: none"> <li>◇ What parts of my program are affected by this new bit of code? (How to get a quick printout of every line that might be affected).</li> <li>◇ What lines of my program do I have to change to convert to a new version of my language? (How to get a quick printout of the lines that might cause problems.)</li> </ul>
<b>New facilities</b>	<ul style="list-style-type: none"> <li>◇ support for sub-directories and wild-cards</li> <li>◇ even faster operation</li> <li>◇ more efficient use of memory and disk</li> <li>◇ more helpful user interface</li> <li>◇ command line control option</li> <li>◇ extra context option (neighbouring lines)</li> </ul>
<b>Multi-Users</b>	Our very favourable site licensing terms make the VICAR ideal to use in education or throughout a business.
<b>Support</b>	Our software is designed not to need support: so we offer a free hotline service for those rare occasions when it does.
<b>Updates</b>	Updates are available at low prices to registered users.
<b>Guarantee</b>	Try the VICAR for yourself. Mail order purchases have a no hassle, <b>money-back guarantee of satisfaction</b> and we encourage dealers to offer one too.
<b>Other Languages</b>	Iansyst produces new versions of the VICAR as the demand arises. We happily work with language developers to support their languages, and we produce versions on (low cost) commission for new languages, new computers and operating systems. Our priorities are dictated by <i>your</i> demands.
<b>Technical Requirements</b>	<p>Operating systems: MSDOS, PCDOS, CP/M-86, Concurrent DOS, CP/M-80, CP/M Plus or true compatibles Enquire about others.</p> <p>RAM needed: At least 128k (16-bit systems); 64k (8-bit systems)</p> <p>Source code size: Virtually unlimited</p> <p>Disk: At least 1 floppy disk. A Winchester disk is advisable for large programs.</p> <p>Printer: Optional No installation required No copy protection</p>
<b>Other products</b>	<p>Note: features vary on some installations and some language versions.</p> <p>IANKEY typing trainers help computer users of all kinds to get the most from their computers. For example, the <b>Typing Crash Course</b> gives a quick grounding in the basics of touch typing, and contains plenty of practice material to work up to speeds of 40 words a minute. The unique <b>"Two Fingers" to Touch Typing Conversion Course</b> converts experienced two finger typists to fluent touch typing in four intensive sessions; you <i>do not</i> have to start again from scratch, and your speed hardly drops while you learn.</p> <p><b>PEP — the Personal Excellence Package</b> — provides various means of assessing and improving your mental performance: IQ tests, personality tests and mental exercises.</p> <p>Vicar, Iankey, Typing Crash Course, "Two Fingers" and PEP are Trademarks of Iansyst. All other Trademarks acknowledged.</p>

# PROGRAMMERS —

## LET THE VICAR SOLVE YOUR PROBLEMS!

Do you ever wonder ...

- ◇ why does that variable have this value?  
Where is the bug coming from?
- ◇ where have you used that identifier which you need to change?
- ◇ what else will happen if you change it?  
Do you just try to "suck it and see" or do you know what the result will be?
- ◇ what was the exact label you used before?  
Was it TimeNumber or TimeNo or what?

Let the VICAR give you answers! Instantly!

Are you ...

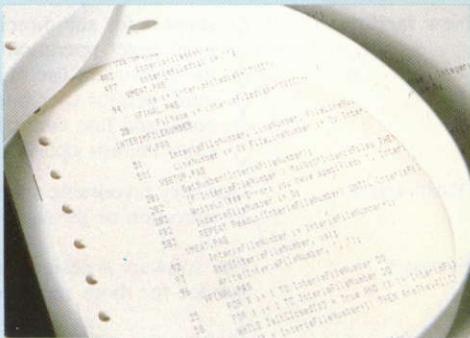
- ◇ writing large computer programs?
- ◇ translating from one computer language to another?
- ◇ updating to a new version of a language?
- ◇ working with several programmers on one project? Or trying to maintain standards?
- ◇ in a mess? (Someone else's, of course)
- ◇ working on a program you wrote long ago?
- ◇ introducing a new procedure or variable to replace an old one?

If so, you especially need the VICAR.

"... easy to use and the output an invaluable aid to debugging, software documentation, and quite a variety ... of other aspects of system development. As the manual says, worth its weight in gold!"  
Nick Plant, Computercraft.

"On the first day of using the VICAR it solved a problem and so paid for itself on that day."  
John McFall, Newcastle Microcomputer Centre.

"I found this a helpful tool to have around the office ... I think the outlay on Vicar is well worth while."  
Program Now



A visual history of each identifier

Programmers do it quicker with the VICAR!

The VICAR saves you hours in understanding your program and in debugging changes. It gives a very comprehensive super-cross-reference.

- ◇ It deals with all your source files in one go.
- ◇ It gives the complete context where each identifier occurs. In most cases you can see what your program is doing without even looking up the source code. The VICAR provides instant answers time after time.

You'll bless the VICAR!

Your Dealer: