

LOW COST PROGRAM GIVES VZ200/300 FULL LEVEL II BASIC

Ever wished that your little VZ200 or VZ300 would run full Microsoft Level II BASIC instead of just a stripped-down version? You needn't wish any longer thanks to an enterprising local programmer.

Jim Rowe

REMEMBER STEVE OLNEY? If you're a VZ200 or VZ300 owner and BASIC programmer, you should. We've published at least three of his articles so far, mainly on resurrecting dormant functions and statement keywords in VZ BASIC. One was in the March '84 issue, another in October '84 and the last in May '85.

Steve's a very knowledgeable guy when it comes to the VZ200/300, in terms of both software and hardware. He's spent quite a lot of time burrowing into its little secrets, and probably knows as much about it as anyone in Australia.

I know that sounds a bit like paeaning in his pocket, but I've just been trying out the latest fruit of his labours. And this time it's not just an article showing you how to restore a few more missing functions to VZ BASIC. It's a machine language utility program that restores pretty well the whole blinking lot for you — instant Level II BASIC! Hence my little paean of praise.

Steve calls his new utility Extended BASIC Version 2.2, or 'EXBSV2.2' for short. It is available on either cassette tape or disk, to suit both basic and expanded VZ systems. It is also compatible with both the VZ200 and VZ300, and with the current Disk BASIC (V1.2 DOS).

You load EXBSV2.2 into your VZ before you load in anything else. It is only about 1600 bytes long (about 1.5K) and is fully self-locating, finding the top of available RAM and installing itself there. At the same time it lowers the BASIC 'top of RAM' pointer to prevent any other programs from being loaded over it.

As part of the installation it patches itself into ROM BASIC, in much the same way that Disk BASIC does, to become

transparent to the user. All that you're aware of is that the RAM is now about 1.5K smaller than before — plus, of course, the fact that your trusty VZ now responds to no less than 25 new BASIC commands!

Of these 25 new commands, 23 are basically resurrected Level II commands that have been sleeping there all the time in the VZ's ROM, quietly waiting for EXBSV2.2 to sound the trumpet. They're listed in the table. The other two are extras — a bonus that Steve Olney has thrown in for good measure. And very handy they are too: MERGE, to allow you to combine programs and routines, and RENUM to let you rationalise and tidy up a program whose line numbers have become a mess after a lot of editing and patching (or after using MERGE).

All of the 25 new commands are fully functional, and when used in a program can be LISTed — at least on any machine with EXBSV2.2 loaded. All but two of them will even RUN on a VZ which doesn't have EXBSV2.2 loaded! The two exceptions are ON and ERROR, which arise because of a conflict in token codes (normal VZs use the normal ERROR token for the added command SOUND).

Even here Steve Olney has provided an answer, for those who really do want the Level II programs they generate to be capable of running on plain-vanilla VZs (how helpful can the guy get?). He's done this by providing the listing of a short BASIC routine which you can MERGE into the top of your programs after they're finished and debugged. You then use it to convert your finished programs

When it has finished, you DELETE the routine itself (notice that?) and CSAVE

the converted program. It won't LIST properly any more, but it will now RUN on a VZ without EXBSV2.2 installed. There's just one tiny catch: you can't use the construct 'IF <expression> THEN ERROR <n>' in any program that you want to convert in this fashion. You can only use ERROR in the 'ON ERROR GOTO' construct. Not a serious limitation, but worth remembering.

But back to EXBSV2.2 itself. Normally you'd expect to load this into your VZ every time you turn it on, which is easy enough and only takes a couple of seconds with the disk system. And with the utility installed, all of the new commands are at your disposal.

It's great to be able to use direct commands like DELETE, AUTO, TRON and TROFF, RENUM and MERGE. How did we ever get along without DELETE? It's so damn useful — not to say virtually essential when you want to scrub a whole range of program lines.

Then into the actual programming. It's really good to be able to use double-precision constants and variables again. Plus to be able to define variables as integer, single, double or string type using DEFINT, DEFNSG, DEFDBL and DEFSTR. It's also much neater to be able to use ON-GOTO and ON-GOSUB, instead of a flock of IF-THENs. Not to mention being able to use ERROR, ERR and ERL. It's nice to be able to use RESUME and RANDOM, too.

Of course there's also FIX, FRE, and MEM — plus familiar old mates like CINT, CSNG and CDBL, POS and STRING\$ (handy in setting out screens, that one — I missed it). And of course the very versatile VARPTR. Wheee! Makes

you feel a bit like Uncle Scrooge let loose in the Mint (well almost).

All of the new commands and functions seem to work perfectly. I certainly couldn't find any bugs, anyway — if there are any, they're pretty well hidden. From a functional point of view, my VZ now behaves like any other Level II machine.

So thanks to EXBSV2.2, Steve Olney's little genie, you can now trundle out all those old TRS80/System80 programs and get them running on your trusty VZ. The graphics will need a few mods, of course, but the programs themselves will be fine.

And the cost of this magic ute? A mere \$15 for the tape version, or \$22 for the disk version. Both prices include packing and postage, and EXBSV2.2 comes complete with a set of driving instructions. You couldn't get much better value for money — obviously Steve Olney is not out to rip anyone off.

I've only got one complaint. Couldn't he have given it a name that's easier to pronounce and type, like 'Jeannie'? Try typing EXBSV2.2 all the way through a review, and you'll know what I mean!

Still, whatever he cares to call it, it's a utility that almost every VZ programmer

TABLE 1. WHAT EXTENDED BASIC PROVIDES

System Commands:

AUTO	automatic line numbering for program entry
DELETE	delete a line or group of lines
TRON	enable trace function (for debugging)
TROFF	disable trace function
MERGE	merge tape program with program in memory
RENUM	renumber program lines

BASIC Statements:

DEFINT	define variable as an integer
DEFNG	define variable as single precision
DEFDBL	define variable as double precision
DEFSTR	define variable as string type
ERR	error code
ERL	line in which error was deleted
ERROR	used to simulate an error condition
ON-GOTO	branch to one of several line numbers depending upon the value of an expression
ON-GOSUB	branch to one of several subroutines depending upon the value of an expression
RANDOM	reseed random number generator
RESUME	continue program execution after error handling

BASIC Functions:

CINT	convert variable to an integer
CSNG	convert variable to single precision
CDBL	convert variable to double precision
FIX	return truncated integer part of a number
FRE	returns the amount of free memory remaining
MEM	returns the amount of free memory remaining
POS	returns the current screen cursor position
STRING\$	returns a string of specified length
VARPTR	locates a variable in memory

is going to want. And at this stage you can only get it direct from Steve Olney at ~~200 Terrace Road,~~ North Richmond, NSW

2754. I only hope that his local post office is prepared for the onslaught.

WRITE TO:

TAPE \$ 15:00
DISK \$ 22:00

S. OLNEY
P.O. BOX 125
NORTH RICHMOND
NSW 2754.

REPRINTED BY KIND PERMISSION OF ELECTRONICS
TODAY
INTERNATIONAL