* * * * * * * * * * *

"small" T_EX

* * * * * * * * * * *

Send submissions to:
Lance Carnes
163 Linden Lane
Mill Valley, CA 94941
(415) 388-8853

The number of small T_EX implementations is growing by leaps and bounds. In addition to TYX's T_EX-in-C, Tom Hickey's Apollo implementation, and my HP3000 version, there is an implementation on the HP1000 and a possible Apple LISA version

The people at TYX inform me that they are busy bringing up T_EX82 in a C version which will run on all of their machines (ONYX, PDP11-44, PLEXUS) as well as on 68000–based systems. My understanding is that this is not a WEB transport, but a rewrite in C (as was their T_EX80).

Tom Hickey has already transported T_EX82 to the Apollo, and reports that it was an easy task. Tom notes that the new version is faster, though this may be partially due to the new Apollo processor. He is using an Imagen printer, and a report on the activities at OCLC appears elsewhere in this issue.

T_EX82 is in the works for the HP3000. See the site report below. (Again this issue, this page was printed on the HP2680A Laser Printer.)

There is a new implementation of T_EX82 for the HP1000 from JDJ Wordware, Cupertino, CA. They have a driver for the Epson MX-80, as well as for the Imagen. See their report elsewhere in this issue.

Alas, there is still nothing for the 8-bit micros, but there is good chance the Apple LISA will soon support a version of T_EX. See David Fuch's article in this issue.

HP3000 SITE REPORT
Lance Carnes

T_EX82 is struggling to life on the HP3000! Transporting the WEB system was unbelievably painless, and bringing up T_EX82 is still before me. What follows is a brief glimpse of what has been done so far.

After unloading the WEB sources from tape, it took me exactly three tries to successfully compile and run TANGLE, and have it output itself in Pascal! Stunned, I decided to press my luck and try to bring up WEAVE. This came up the first time, mostly because the change file is almost identical to that for TANGLE.

Most of the problems encountered so far had to do with the change files. After trying to create them *ad hoc* based on the listings, it occurred to me combine all of the system dependent modules (identifiable by their index reference @ ↑system dependencies) as a first approximation change file. This works out well, since you need only read and modify the lines in this file.

T_EX82 should be up in the next few weeks, and I will be able to report on its performance at the July meeting.

TWO BUGS IN T_EX80-in-Pascal
(or FLOGGING A DEAD HORSE)
Lance Carnes

For those of you still hacking away at T_EX80 the following bugs have been identified. Both are caused by variables which contain uninitialized values.

The first bug occurs in the procedure hyphenate where exception lookup is done. In the August 1981 listing of T_EX, section 470, the following appears:

for $i := j + 1$ to n do
$\quad$ $truncword[i] :=$ *shortAsciiNull*;

And then in section 472 there appears:

for $i := 1$ to *hashlength* do
$\quad$ $hash := hash^*16 + truncword[i]$

The problem I experienced was that *hash*, which is computed from the first *hashlength* characters, became a negative number. It turned out this occurred because for $n <$ *hashlength* the last $hashlength - n$ places in the array *truncword* were not assigned a value. Since *truncword* is a local array, its initial value is just whatever garbage was left on the stack, and if the garbage happened to be a negative 32-bit integer, *hash* became negative also.

The fix for the bug is to replace n with *maximumDistinctionLength* in the code from section 470 shown above.

The second bug turned up when a source file had the following: \xdef\junkie{}. The symptom is an array index violation in Pascal runtime. The bug occurs in the procedure scantoks.

Refer to Section 194 in the August 1981 listing of T_EX. The array index violation occurred in the third to last line:

$link(q) := 0$; { delimit token list }

It turns out q is not set in the case of an empty definition, i.e. {}, and since it is a local variable, it just contains whatever garbage is on the stack.

The fix for this bug is to replace q with p. A workaround is to use \xdef\junkie{{}}.

The moral of this story is: if you are going to use a Pascal variable be sure you have previously assigned it a value.