## Errata to So you went and bought a vector computer

## Stewart A. Levin

In SEP-44 I wrote a short article discussing how to write FFT algorithms for a vector computer such as the Convex C-1 here at SEP. In the appendices I gave some simple Fortran codes reprinted by permission of North-Holland from an excellent article by Paul Swarztrauber in "Parallel Computing". I also supplied some comparative timings of them for our Convex. Since then I've discovered a typographic mistake and a Fortran compiler problem that invalidated several of the timings. The corrections follow.

The declaration in Figure 7 for subroutine CANGLE was in error. Argument N should have been M. This invalidated my timings for the Pease long-vector FFT because this typo caused CANGLE to reset the vector length N to 1. In addition to this error on my part, several of the subroutines had been improperly optimized by the compiler I used while making the timings. Since then, Convex has released new versions (1.6 and 1.7) of their Fortran compiler which fixed these problems. The table below is the result of retiming the 1024 complex test problem after the abovementioned corrections. For the Pease long-vector algorithm the additional 13 msec is for table generation with CANGLE. I emphasize these timings are for the simple Fortran codes I published in SEP-44. Convex has recently provided SEP with a hand-coded, optimized variant of the Pease algorithm that runs in 3 msec or five times faster than the corresponding Fortran timing given below.

Algorithm	Sort	Combine	Total
Cooley-Tukey	18	56	74
CT/loop invert	18	17	35
Gentleman-Sande	19	67	86
Pease	19	66	85
Pease/long vec	19	8+13	40
" + loop inv	7	8 + 13	28
Stockham			66
" + loop inv			20

