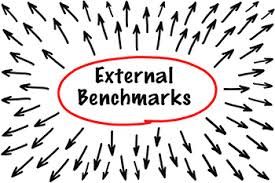
**Software 32**

So, I now can GOTO …..

Back in the dim and distant past, the first thing that magazines would do with the latest home computer is benchmark it.

PCW had it’s own set of benchmarks, which were BASIC programs that were run and measured to see how long they take. The simplest one was this :

100 FOR I = 1 TO 1000

110 NEXT I

I’ve actually found them here <http://www.geocities.ws/peterochocki/computers/pcwbm.html>

To give some idea ; a ZX81 4.5s, a Speccy 4.4s, a C64 1.2s,a BBC Micro 0.8s. The slowest machine would be a ZX81 in slow mode which would be about 18s or so (estimate).

The variation is quite surprising. The first thing on the list is a 68000 at 8Mhz (HP Integral PC) which takes 1.9s ; it is quite remarkable that the BBC Micro (6502 at 2Mhz) should be over twice as fast given that both are operating in floating point.

So, I have benchtested it. I can’t actually run BM1 but I can run Benchmark 2, which is the same thing with an IF … GOTO instead , in VTL-2,

100 A = 0

110 A = A + 1

120 # = (A < 1000) \* 110

Line 120: Returns 110 if A < 1000 and 0 otherwise ; # (GOTO) does not execute if its r-expr is zero.

Which runs in 46 seconds. This isn’t bad, really – it’s a slow processor, it’s running at a quarter of its actual speed anyway (just sticking an SC/MP 2 in and clocking it up to 4Mhz will get that down to 10 seconds) and it’s not written for efficiency by any means, it’s written more with clarity in mind (honest – the optimised one would look like spaghetti).

By comparison : ZX81 6.9s, Speccy 8.2s, C64 9.3s, Atari 800 7.3s

So not too bad really. If I conveniently forget they are all doing it in floating point and I’m doing it in 16 bit integer ☺

For depressing comparison ; 2400Mhz Celeron, running QBASIC, 0.011s. I have reduced this PC to a wreck of its original finery.