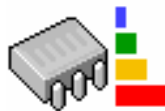


SPEEDY v5.2

June 6, 2005



Index

What is Speedy?	2
Contact Information	2
How to install Speedy?	3
How to use Speedy?	3
User Manual	3
CPU Frequency versus Timing	4
Performance increase with overclocking	5
Is Speedy efficient? (analysis for DragonBall processors based devices)	6
1 st case: simple overclocking	8
2 nd case: overclocking + additive accelerating features	9
What about new ARM devices running Palm OS5?	10



What is Speedy?

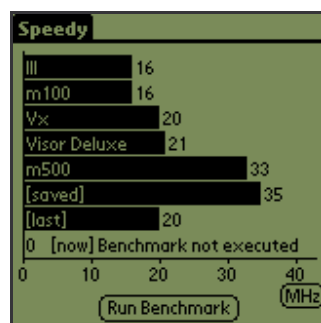
Speedy is a benchmark utility for Palm OS based devices.

Speedy performs three resource taking routines (calculation, memory access and graphical display) to measure the real performance of your device.

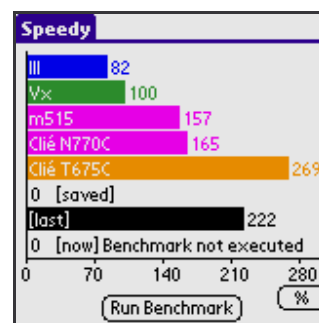
Speedy is especially useful in different cases:

- to check that your Palm OS based device behaves and performs like it should;
- to see the impact of CPU down/over-clocking upon your device's performance;
- to really compare different devices and benchmark them.

Your device's performance is ranked on several charts against baseline measurements of other common Palm OS compatible devices. Speedy reports both a CPU frequency and a relative index (100% is for the Palm Vx at 20 MHz).



CPU frequency view



Relative index view

Contact Information

- ✓ World Wide Web main download Site : <http://www.aldweb.com>
- ✓ Author e-mail : info@aldweb.com

Speedy is free to use. However, if you like it, please consider making a donation to show your support. Any amount will be highly appreciated.

To proceed, just connect to:

<http://give.aldweb.com/> (secured PayPal transaction)

For your information, the usual donation amount is of US\$ 10 = EUR 10

How to install Speedy?

There is nothing special to say here.

Speedy is a PRC file that is installed like any other Palm file using HotSync.

So, extract **Speedy.PRC** from the ZIP archive file.

Double-click on it and the Palm install tool will popup.

Speedy.PRC will be transferred to your Palm device next time you synchronize your Palm with your PC with HotSync.



Avoid installing the current version of Speedy over a previous one. I do not guarantee that it will work fine doing so. Please, uninstall any previously installed version of Speedy before installing this one.

✓ **Minimum Palm OS requirement for Speedy is version 2.0**

✓ **Speedy is Palm OS version 5 compliant**

How to use Speedy?

User manual

I'll be very quick in these explanations as Speedy behaves like all Palm OS based applications.

Speedy is very easy to use. Just press the **Run Benchmark** button to launch the benchmark routines. Result will be stored in the [now] line. Last result is stored in [last] line, so as for you to remember the previous benchmark result.

You can also save the [now] or the [last] result in the [saved] line.

By taping on any line, you can see the details of the benchmarks of the different devices.

By taping on 'MHz' (Frequency) or '%' (Relative timing) written at the right of the X axis, you switch between these two chart types.



Be careful not to tap with the stylus on the screen while the benchmark runs, otherwise the results will be depreciated by up to 10% (about 10% if you keep the stylus pressed against the screen).

CPU Frequency versus Timing

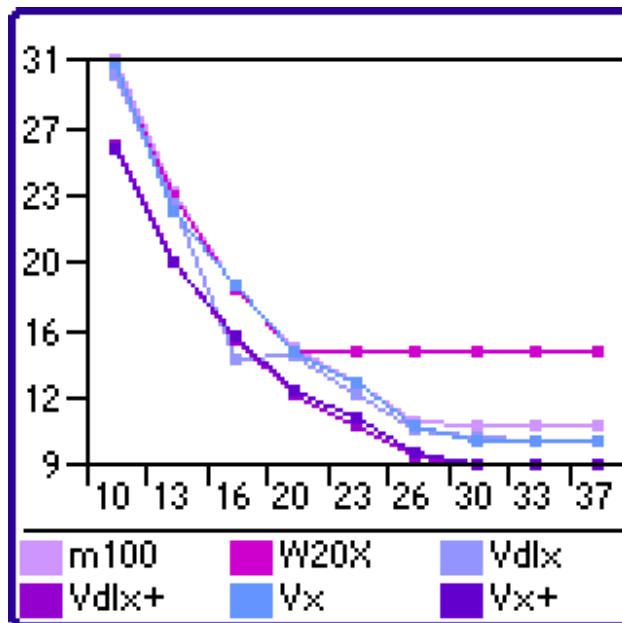
All collected datasheets and graphs shown here under are available in a QuickSheet file (<http://www.cesinc.com>) : **BenchSpeedy.PDB** (sent together with this Speedy.pdf file).

The formula which links frequency (F) and timing (T) is:

$$F = 1 / T$$

This formula was tested at different CPU clock speeds using AfterBurner hack (<http://www.palmgear.com>) on several Palm devices :

- Palm m100 (**m100**) – normal CPU speed is 16 MHz
- Workpad 20X, clone of the Palm III (**W20X**) – normal CPU speed is 16 MHz
- Visor Deluxe (**Vdlx**) – normal CPU speed is 16 MHz
- Visor Deluxe with with 0-ws Core, Fast Mpx and Fast Page options set. (**Vdlx+**)
- Palm Vx (**Vx**) – normal CPU speed is 20 MHz
- Palm Vx with with 0-ws Core, Fast Mpx and Fast Page options set. (**Vx+**)



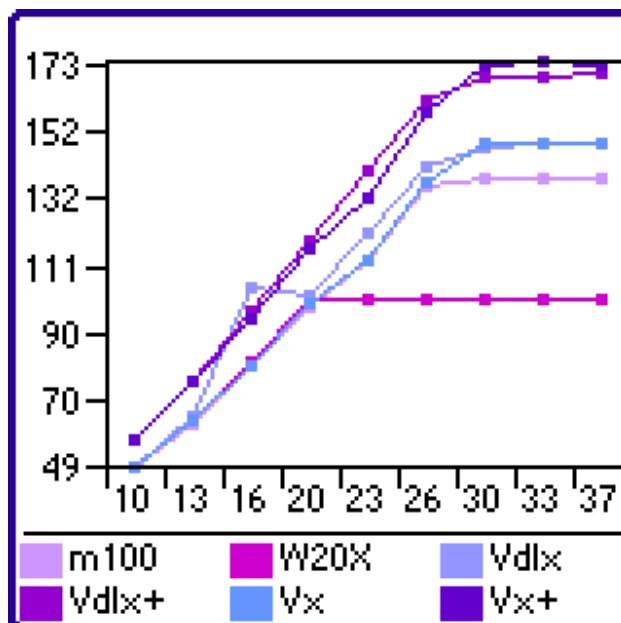
X axis: CPU speed (in MHz)

Y axis: Absolute speed (in seconds)

All curves follow the $F = 1 / T$ formula with 2 artifacts that can be easily explained:

- Visor Deluxe artifact:
At the normal clock speed of 16 MHz for this device, we can see on the graph that the value point is not normal.
This is because the Visor Deluxe has the three 0-ws Core, Fast Mpx and Fast Page options set by default. If you use AfterBurner at this speed, inactivating these three options will have the curve behave back at it should.
This is a wise way found by Handspring to accelerate this device without overclocking it!
- Workpad 20X artifact:
This older Palm device cannot stand more that a 20 MHz overclocking.

Performance increase with overclocking



X axis: CPU speed (in MHz)

Y axis: Relative speed (in %)

This graph clearly shows that, upon a given overclocking, the Palm devices do not speed up anymore. So, you do not need to overclock your Palm device above a given CPU speed. Speedy is very useful to find this overclock limit.

Is Speedy efficient? (analysis for DragonBall based devices) (in other words : how does Speedy compare to Benchmark v2.0 ?)

What a strange question, isn't it? But, you have to be aware that, as with any benchmarking program, the numbers returned by Speedy can be misleading if you do not understand what is being measured and how to interpret the results.

Indeed, the performance of a given application fully depends on the mix of operations (calculations, memory accesses, database accesses, display on screen...) performed by that application. All benchmarking utilities will use their own mix of operations and try to be as close as possible from the average mix of operations that most applications will have.

Speedy has been clearly and very carefully designed to perform three high resource taking routines (calculation, memory access and graphical display) to give numbers as close as possible to a standard Palm OS application's running speed, whatever the CPU frequency of the device is, down/over-clocked or not.

This being said, let's see how Speedy compares to another well-known benchmarking utility for Palm OS devices.

Benchmark v2.0 (<http://www.quartus.net>) was written and compiled by Neal Bridges on-board a Palm IIIc using Quartus Forth in 1999, in order to be a freeware demonstration program of Quartus Forth's ability to produce compact, fast, stand-alone applications.

Benchmark has been until now and the arrival of Speedy almost the ONLY benchmarking utility for the Palm OS platform and it therefore became the widely accepted standard benchmarking tool for this platform.

I have written Speedy because it was a good opportunity for me to understand how to program a benchmarking tool. I then discovered that I had built what I believe to be a more reliable benchmark tool for the Palm OS based devices, in the way that Speedy seemed to give numbers closer to the real speed of the devices. My objective here is not to criticize Benchmark v2.0 as it has been very useful in two ways : prove that Quartus Forth is a good development language (and believe me, it is!) and give quite a good benchmarking tool for Palm OS devices two years back from now.

My objective is only to try to explain you why according to me the newer program Speedy (written with another very good development language being HSPascal) is just a little bit more accurate in its results than Benchmark.

For doing so, I timed a few programs' execution on a Palm Vx at three speeds using the AfterBurner hack (<http://www.palmgear.com>) :

- 10 MHz
- 20 MHz
- 30 MHz

The timed programs were:

- **SCells**, a modified version of my Cells freeware program (source code available both for SCells and Cells) with automatic timing of one cells board evolution.
- **SMatches**, a modified version of my Matches freeware program (source code available both for SMatches and Matches) with automatic timing of a few games played by two computer players.
- **SNumerus**, a modified version of my Numerus freeware program (source code available both for SNumerus and Numerus) with automatic timing of hundreds of calculations.
- **PrefMngr** (<http://members.xoom.com/PixIL/Pilot>) with manual timing.
- **PC-cillin** (<http://www.antivirus.com/wireless>) with manual timing.
- **InoculateIT** (<http://www.ca.com>) with manual timing.

Note: SCells, SMatches and SNumerus derivate from versions of the current Cells, Matches and Numerus programs which are Palm OS v5 compliant.

These programs can run for a period of time with no user interaction (no ticks), a condition needed for timing execution speed.

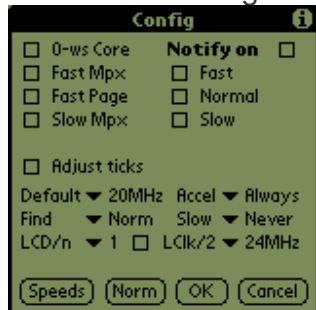
Of course, automatic timing as reported by the Palm is much more precise than manual timing which implies coordination between my eye, my brain and my hand to start and stop the timer!

The use of a shareware program like SysInfo (<http://www.aldweb.com>) was needed to disable the auto-off timer of the Palm device, as some programs execute in more than 3 minutes (which is the maximum auto-off timer that can be entered in the Palm Preferences).

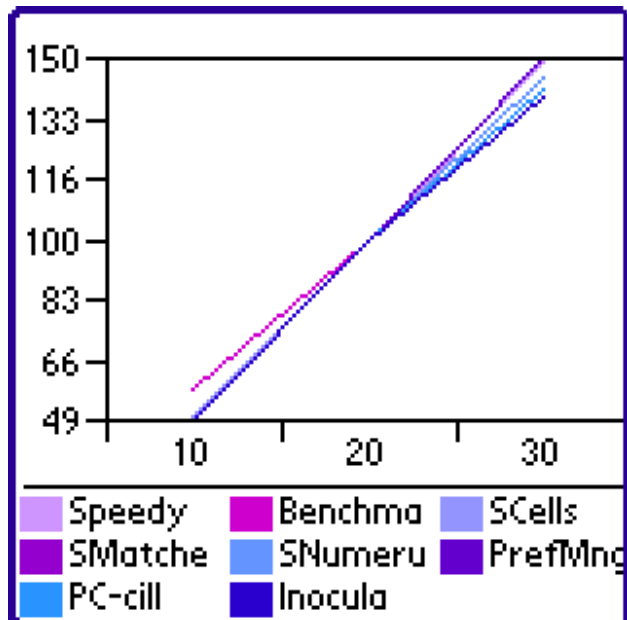
All collected datasheets and graphs are available in a QuickSheet file (<http://www.cesinc.com>) : **BenchApplis.PDB** (sent together with this Speedy.pdf file).

1st case: simple overclocking

AfterBurner configuration:



Results, all brought back to be at 100% for Palm Vx at 20 MHz (standard speed) with no additive acceleration option (0-ws Core, Fast Mpx and Fast Page) set.



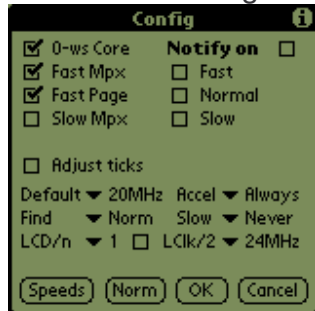
X axis: CPU speed (in MHz)

Y axis: Relative speed (in %)

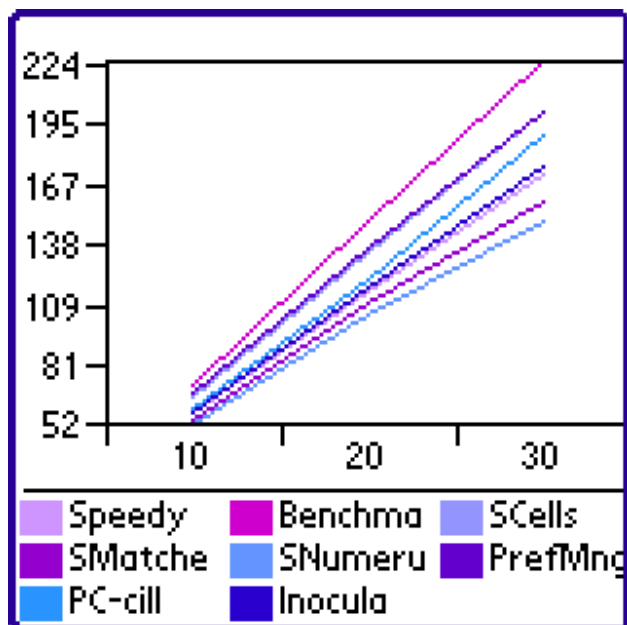
The results are very clear: both Speedy and Benchmark give accurate results.

2nd case: overclocking + additive accelerating features (0-ws Core, Fast Mpx and Fast Page)

AfterBurner configuration:



Results, all brought back to be at 100% for Palm Vx at 20 MHz (standard speed) with the 0-ws Core, Fast Mpx and Fast Page options set.



X axis: CPU speed (in MHz)

Y axis: Relative speed (in %)

According to the programs' use of calculations, memory accesses or screen display, the execution speeds can be very different from one program to another. If Speedy is about in the middle of the range thus giving an average execution speed, we can see on the graph that Benchmark results in the highest line, thus giving a much too positive trend.

This means that Benchmark is very "enthusiastic" and overestimates the speedup due to the set up of the 0-ws Core, Fast Mpx and Fast Page accelerating options, whereas Speedy will give a more accurate idea of the real gain of these options.

What about new ARM devices running Palm OS5?

Benchmark results for the new ARM devices as returned by Speedy are to be considered with care.

Speedy was initially designed to work with the former generation of Palm OS based devices running a DragonBall processor.

In its current version, Speedy has been upgraded to take into account the specificities of the new ARM based devices and to re-evaluate the accuracy of the results returned.

But, when the benchmark would run during 15 seconds for a 20 MHz DragonBall device, it runs in less than one second for the new high speed ARM devices. As a consequence, precision is depreciated with the new devices.

For example, running Speedy on a Palm Vx (with a 20 MHz DragonBall EZ processor) and on a Palm Tungsten C device (with a 400 MHz ARM processor) will provide these kinds of results:

- Palm Vx
 - 15.02 seconds ~ 20 MHz
 - 15.04 seconds ~ 20 MHz
- Palm Tungsten C
 - 0.75 second ~ 400 MHz
 - 0.77 second ~ 390 MHz

As you can see, a 0.02 second benchmark speed difference will return with a 0 MHz difference for the Palm Vx and with a 10 MHz difference for the Palm Tungsten C. This is no big deal, the Palm Tungsten C really is a fast device ☺

If we now compare Benchmark 2.0 and Speedy again, but for the new ARM devices, we get these results for the Palm Tungsten C:

- Speedy: 2000% / Palm Vx = 20 times faster than Palm Vx
- Benchmark: 539% / Palm Ille ~ 592% / Palm Vx ~ 6 times faster than Palm Vx

Since the Palm Tungsten C runs a 400 MHz processor and the Palm Vx a 20 MHz processor, we could roughly expect the Palm Tungsten C to run approximately 20 times faster than a Palm Vx. This is the result returned by Speedy ☺