Thermal and Operational Characteristics of Several Models of Raspberry Pi Under Load

HDTodd@gmail.com, May, 2016

Summary

- Under normal operating conditions (ambient temperature about 22°C, standard clock rates, but regardless of load), Pi-1 and Pi-2 models would not benefit substantially from the addition of a heatsink or fan, whether or not the Pi is housed in a case.
- Under normal operating conditions (ambient temperature about 22°C, standard clock rates, steady-state loads < 1.0), a heatsink might extend the life of your Pi-3 at a minimal cost, but the cost and added operational complexity of a fan would probably not be justified.
- If your Pi-3 operates with a load of > 1.0 but < 3.0 for extended periods of time, a fan would likely be a justified investment to extend the lifetime of your Pi.
- If you expect to operate your Pi-3 with a load of > 3.0, you may need to install a fan to maintain the Pi-3 temperature below the (default) 85°C thermal trip-point, above which the processor speed (and performance) are throttled to avoid thermal damage to the processor. Depending on the case, without cooling, you may receive only 70%-85% of the theoretical capacity of your system.
- A Pi-3 with a heatsink operates at about 5-6°C <u>lower</u> in the aluminum MobileAppSystems Proto Armour case than in the acrylic C4Labs Zebra (old style) case. A Pi-3 with heatsink in the Proto Armour case did not experience CPU throttling due to heat when operating with all four cores active, but it did operate at about 80°C.
- A Pi-3 in a Zebra case with a heatsink and with the XWindows screen being displayed but with no processing activity – a situation that might apply for long periods if you just leave your system running while you're not using it – would "idle" at about 50°C.
- The Pi-3 has a multi-thread work capacity roughly 10x the Pi-1B.

Introduction

As a result of its success in the education and hobby marketplaces, the Raspberry Pi has evolved quickly to offer several different versions based on three models of Broadcom processors. The latest model, the Pi-3 has a 1.2GHz, 64-bit, quad-core processor and offers about 10x the processing power of the original Pi-1.

Monitoring my new Pi-3, I found the temperatures to be higher than I had anticipated (though well below manufacturer design specs, apparently). I investigated adding a heatsink. I found that other owners had also wondered if they should take steps to help cool their processors and had explored options. I was particularly intrigued by the work of Christopher Barnatt (see References), who showed the value of (and technique for!) adding a heatsink or fan to a Pi-3 system. [His work is a model for what I've done here.]

Barnatt's work still left me wondering if a fan would be cost-effective in my particular circumstance – I have a low load most of the time. So I thought it would be interesting to explore the thermal and operating characteristics of the Pi-3. And then curiosity led me to repeat similar tests on my Pi-1B and Pi-2B systems.

I had developed some tools to display weather-station data, from which I developed a set of tools to monitor and display Raspberry Pi CPU temperatures (available on GitHub; see References). Researching what others had published, I decided to take some steps to moderate my Pi-3 temperature and document what I learned along the way.

Though I'm a bit rusty at it, I tried to use my skills as a trained scientist (and educator) to approach this scientifically: document the details of the equipment under study, document the details of the testing procedure, and strive to make this an experiment that others could reproduce themselves, with their own particular situations.

This paper describes the experimental details, summarizes the results, and suggests some possible conclusions that might be deduced from the observations.

Experimental

Though the focus of this work was to investigate the thermal and operating characteristics of the Pi-3, it might be helpful for others to know similar characteristics of the other Pi models and to be able to compare the Pi-3 with the other models in similar situations. So while the heatsink experiments were on the Pi-3 only (subsequently justified by observations on the other models), the observations include data for other Pi models, too.

The models used in this study include:

- Pi-3; reboots off external, USB-connected SSD; single USB connection to USB hub; HDMI output; wireless Logitech 400 keyboard/trackpad connected through USB hub; WiFi enabled; Bluetooth disabled; Zebra Black Ice (plastic) case; initially without heatsink.
- Pi-2B; boot off internal microSD; Edimax WiFi USB adapter; accessed via SSH; MobileApps Systems' Proto Armour (aluminum) case;

- Pi-1B: boot off internal microSD; Edimax WiFi USB adapater; accessed via SSH; one in a Proto Armour case and one in a Zebra Black Ice case.

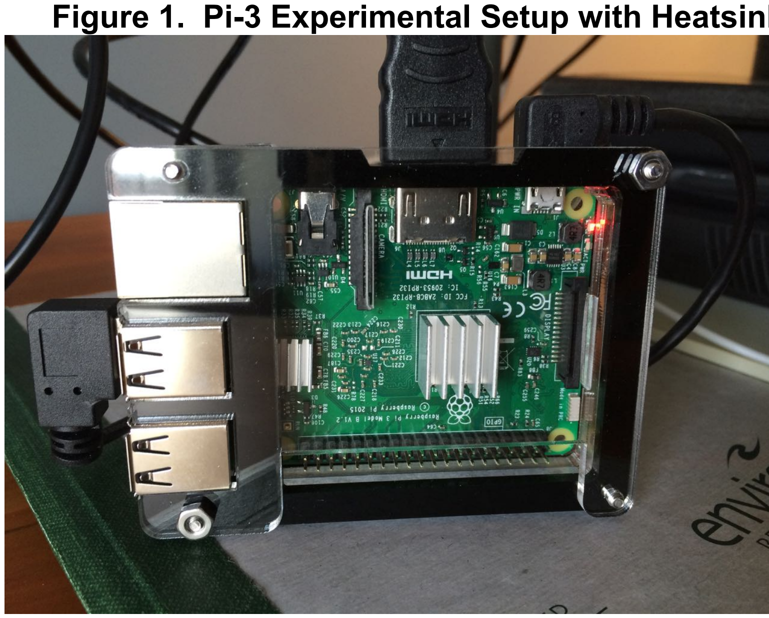
<u>None of these are overclocked.</u>

The Pi-1B processor has a single core; the Pi-2B and Pi-3 processors are quad-core, though with different architectures. Testing used up to 5 threads on the Pi-2B and Pi-3, to confirm that the processor cores were being fully utilized; only 2 threads were needed to confirm that the single core was being fully utilized (and that temperatures had reached maxima) on the Pi-1B.

The Pi's were connected in a manner that was intended to minimize current flow through the Pi board. The Pi-3 was connected to an HDMI display and (through the USB hub) wireless keyboard during the experiments. The other systems were accessed via SSH through their USB-based WiFi network connections. Each of the systems had only one USB device directly connected to the Pi board during the tests.

The Pi-3 had been installed in a Zebra Black Ice case. The temperature comparison studies were conducted with the case top on and off, but the Pi was left in the case (not tested as a bare board). All of the Pi-3 tests were conducted with the Pi standing on its side, as shown in the photo below, which also shows how the heatsink was subsequently installed for those tests:

Figure 1. Pi-3 Experimental Setup with Heatsink



All experiments were conducted in command-line mode, without XWindows running (except for the XWindows idling test), and no other programs were actively running (other than monitoring programs), though services such as Apache2 and MySQL were loaded and available (but dormant).

All systems were running Raspbian Jessie, 4.1.19-v7+. Experiments used sysbench v0.4.12 (installed via apt-get) to generate CPU load. The sysbench CPU test is a prime-number program that uses 64-bit integer arithmetic. In that sense, it is likely not be a good test of comparative CPU performance across Pi models, but it does generate a CPU load that causes the processor to heat up, which was the goal of these experiments.

The sysbench command line used to generate a single-thread job for the temperature studies was:

```
sysbench --test=cpu --cpu-max-prime=100000 --max-time=900 –batch
--batch-delay=60 --num-threads=1 run &
```

and modified accordingly to run 2, 3, 4, or 5 threads. The trailing "&" was used to run sysbench in the background and permit the terminal to be used for the manual monitoring progress. The sysbench command line used to generate a single-thread job for "work capacity" comparisons among systems was:

```
time sysbench --test=cpu --batch --batch-delay=60 --num-threads=1
--cpu-max-prime=20000 --max-requests=10000 run &
```

with the max-prime and requests values of (10000,20000) and (5000,10000), respectively.

Equilibration to steady-state load and temperature were monitored manually at 1-minute intervals (triggered when sysbench reported its 60-second periodic batch status) with the terminal command:

```
uptime; cat /sys/class/thermal/thermal_zone0/temp
```

All systems tested had the thermal trip-point set at the default Raspian setting of 85°C (`cat /sys/class/thermal/thermal_zone0/trip_point_0_temp`).

For the Pi-3 temperature equilibration data collected for the graphical display, I used the RPiTempLogger program available from GitHub (see References) and apache2 with Google Graphs and PHP code (from the package) to display the results graphically. The RPiTempLogger.c code was modified to sample at 15-second intervals (rather than 300-second intervals in the package as distributed) to present the data visually with finer granularity.

The RPiTempLogger program uses the same source for its temperature reading as the terminal command above (/sys/class/thermal/thermal_zone0/temp). [This appears to yield the same result as the `vcgencmd measure_temp` command that Barnatt uses.]

The process for each set of measurements to measure and record the increase in temperature under load and monitor the Pi-3 for temperature equilibration is as follows:

0. **Preparation:** Install tools (sysbench; edit RPiTempLogger.c with interval changed to 15 sec, make, make install; sqlite3 & Apache2 if not already installed & configured to support RPiTempLogger); reboot to make sure

logger is running; verify that RPiTempLogger is creating entries in the database in `/var/databases/MyPiTemps.db` .e.g.,

```
$sqlite3 /var/databases/MyPiTemps.db
sqlite> SELECT * FROM PiCoreTemps WHERE DateTime>datetime('now','-24 hours');
.quit
$
```

Make sure there are some recent entries for temperature in that database to verify that the logger is operating correctly. Then clear the database by renaming that db file so the logger will start a new one, cleanly. Shutdown the Pi and let it cool to room temperature.

1. Power up and boot up the Pi; verify that RPiTempLogger is running
2. Issue the command to start the CPU test with a request for one execution (representing a load of 1.0):
   ```
   sysbench --test=cpu --cpu-max-prime=100000 --max-time=900 \
   --batch --batch-delay=60 --num-threads=1 run &
   ```
3. As sysbench issues its 1-minute updates, type in the command (the up-arrow bash recall command is very handy here!):
   ```
   uptime; cat /sys/class/thermal/thermal_zone0/temp
   ```
   Monitor the increase in load as it approaches the number of threads you've asked for, and monitor the temperature to see that it is increasing. The temperature should converge to a fairly constant value after about 10 minutes. The sysbench command runs for 15 minutes to ensure that the temperature has stabilized, but if you find that the temperature isn't stable after 15 minutes, you may need to rerun the test (be sure to restart the database!) with a longer execution time.
4. When the first pass (1 thread) has finished, you do not need to cool the system down if you're increasing the number of threads. Just issue the command to run again with 2 threads (load=2):
   ```
   sysbench --test=cpu --cpu-max-prime=100000 --max-time=900 \
   --batch --batch-delay=60 --num-threads=2 run &
   ```
5. Repeat 2-3 again with –num-threads=n for as many runs as you want <u>but</u> if you decrease the number of threads relative to the immediately preceding run, allow the system to idle for a few minutes until the load and temperature (reported by the commands in 3 above) drop to near-idle conditions again.
6. When you've finished the sequence of runs for one testing setup, make a copy of the database file for safe keeping at this point; it's easiest to do the following:
   ```
   sudo cd /var/databases/
   sudo cp MyPiTemps.db Pass<n>-MyPiTemps.db
   ```
   where <n> is a different digit for each pass with a new setup (with/without heatsink, with/without case, etc.).
7. You can check the graphical display of the data by providing "localhost" as your URL to a browser.
8. Shutdown; disconnect power; let system return to room temperature while you set up for the next pass, with different physical setup or CPU, then repeat starting at 1.
9. After completing all the tests for a particular system,

a. kill RPiTempLogger
b. Copy the `/var/databases/MyPiTemps.db` file to a different place for safe keeping.
c. Edit the RPiTempLogger.c code to return the interval to 300; make; make install; reboot

(Questions, clarification, etc? Please email me to the address above with specific items that aren't clear.)

Observations

Ambient room temperature was consistently about 20°C-22°C during these tests. No compensation was made for slight changes in room temperature.

I ran a number of tests to determine optimum test conditions, determine the accuracy of the temperature sensor and reproducibility between units of the same model, etc., and answered a few questions along the way:

- **How long should a temperature-maxing test run?** A test run on the Pi-3 was made with sysbench max-time=600 to determine if that was sufficient time for the CPU temperature to reach its maximum: it was close to a converged maximum temperature (see graph) but I wasn't fully confident that it had completely converged. To ensure that temperatures had stabilized, subsequent tests on the Pi-3 used a sysbench max-time=900 setting to allow the CPU to reach its maximum steady-state temperature under load.
- **How accurate are the Pi-1B CPU temperature sensors?** With an ambient room temperature of 22°C, two different Pi-1B's that had been powered down for at least 24 hours were powered up and booted up, and their temperatures were read within 1 minute of power up. They reported temperatures of 30.4°C and 29.9°C, with loads of 1.83 and 1.77. We might draw two conclusions from this:
    - It appears that the Pi-1B temperature sensor may report a temperature that is as much as 8°C above the actual temperature.
    - Temperature readings between different Pi-1B's appear to be consistent within about 1°C (based on only two sample points)
- **How accurate is the Pi-2B CPU temperature sensor?** With an ambient room temperature of 22.4°C, a cold-booted Pi-2B reported a temperature of 31.5°C within 1 minute of power-up, with load 1.31. It appears that the Pi-2B thermal sensor may read as much as 9°C above the actual temperature of the device.
- **How accurate is the Pi-3 CPU temperature sensor?** Despite the ambient room temperature of 22-23°C, the Pi-3 CPU temperature was reported to be 35-38°C as the first temperature read within 1 minute after powering up and booting from a room-temperature-equilibrated state (powered off for at least 24 hours). Even allowing for some immediate

heat buildup, it appears that the Pi-3 thermal sensor may read as much as 16°C above actual temperature of the device.

- **What are the temperatures for the various models at idle?** Each of the systems was booted and allowed to stabilize until the load was reported to be approximately 0.0, then the reported CPU temperature was recorded (again: no XWindows, no active running processes). The "idle" temperatures reported were:
    - Pi-1B: 37.9°C
    - Pi-2B: 34.7°C
    - Pi-3: 46.2°C
- **How hot would the Pi-3 become if I left it on but not running anything?** For the Pi-3 with a heatsink in place and in a Zebra Black Ice case, with XWindows running to offer a graphical interface, but with no active processes, the load was reported to be 0.15-0.25 and the CPU temperature was reported to be 49°C. So it appears that a Pi-3 left for long periods of time with XWindows running but no other active processes would "idle" at roughly 50°C, though (again) if the sensor is reporting 16°C too high, that "idle" temperature might actually be only about 34°C.

To confirm that sysbench generates a CPU load that would stress the 4-core Pi-3 under load and increase its temperature, two runs were made, with the Pi-3 in the case but before the heatsink was installed, with num-threads=1 and =2:

- Num-threads=1 generated a steady-state load of 1.0 @ 58.5°C
- Num-threads=2 generated a steady-state load of 2.0 @ 70.3°C

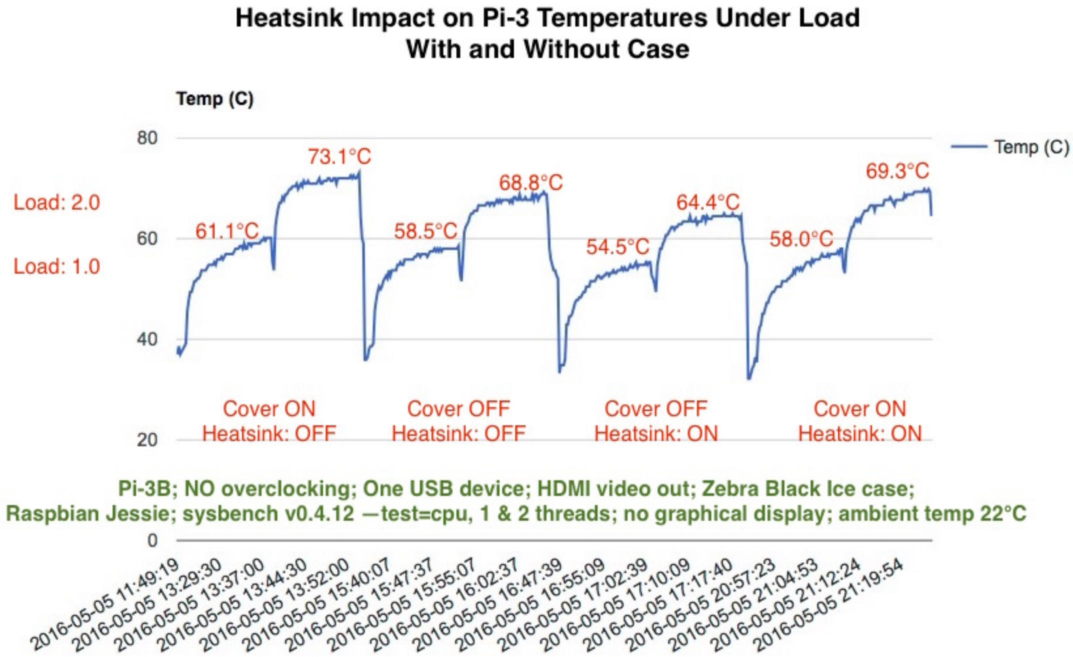
These results demonstrated that the sysbench thread-count generated a corresponding value for CPU load and that adding a second thread – and engaging a second core in the case of Pi-3 and Pi-2B – really did add a second CPU task to the system load.

With the tools calibrated and multi-core operation verified to function as expected, the load-generated heating of the Pi-3 was tested in four configurations:

- Without + With case cover
- Without + With heatsink

The Pi-3 was tested with the sysbench command with 1 & 2 threads in each of those configurations, and the temperature at the resulting maximum equilibrium temperature was recorded. Figure 2 shows the temperatures as the Pi-3 reached its maximum temperature in each of those cases.

Figure 2.



The equilibrated temperatures (averaged, rather than the peaks shown in the figure), under load, for the four different cases are summarized in Table 1.

Table 1. Impact of Heatsink and Case upon Pi-3 CPU Temperatures

Pi-3 CPU temperatures	Case Uncovered		Case Covered	
	No Heatsink	Heatsink	No Heatsink	Heatsink
1 thread	58.5°C	54.8°C	60.1°C	56.9°C
2 threads	68.8°C	64.5°C	72.5°C	69.8°C

The results here show that a case can raise the operating temperature of the Pi-3 by 2-5°C, and the addition of a heatsink can lower its temperature by 3-4°C. They also show that the Pi-3 heats quickly as additional cores are activated.

To analyze the active-core-count heating dependency of the Pi-3 and to compare heating among the models of Pi's, the sysbench command was used with num-threads=1..5 (Pi-2B, Pi-3) or 1..2 (Pi-1B) to determine the thermal characteristics of each of the models under load. The resulting data are presented in Table 2:

Table 2. Reported CPU Temperature as a Function of Load

Threads	Load	**Pi-1B**, Zebra case, no heatsink	**Pi-2B**, Proto Armour case, no heatsink	**Pi-3**, Zebra case, with heatsink	**Pi-3**, Proto Armour case, with heatsink
0	0.0	38.5°C	34.7°C	46.2°C	41.9°C
1	1.0	41.2°C	40.6°C	59.6°C	53.7°C
2	2.0	41.7°C	44.9°C	69.2°C	64.5°C
3	3.0	–	48.7°C	79.0°C	72.5°C
4	4.0	–	53.0°C	82.7°C[1]	80.6°C
5	5.0	–	53.0°C	82.2°C[1]	81.7°C[2]

[1] The yellow overheating-warning box appeared in the upper right corner of the attached display shortly after beginning the 4-thread test and continuing in the 5-thread test.

[2] A faint yellow overheating-warning box appeared in the upper right corner of the attached display several minutes into the 5-thread test.

By extrapolation, in the Zebra case the Pi-3 temperature might have been expected to reach 89°C with all four cores running, but it only reached 82°C. The sysbench program also reports the number of “events” in each of its runs, and for the Pi-3 case, the reported values were:

Table 3. Work Delivered by the Pi-3 as a Function of Load.

Threads	**Load**	**Pi-3 Temp**, Zebra case, with heatsink	**Number of sysbench events**	**% of Events vs Projected from 1-thread**
1	1.0	59.6°C	1318	100.0%
2	2.0	69.2°C	2643	100.3%
3	3.0	79.0°C	3961	100.2%
4	4.0	82.7°C[1]	4486	85.1%[1]
5	5.0	82.2°C[1]	4478	84.9%[1]

The work delivered is a linear function of #threads for 1-3 threads. Extrapolated to the 4th thread, the system only delivers about 85% of the expected work “events”. It appears that although the thermal-limiting trip point is set at 85°C, and even though the Pi-3 temperature sensor may be reporting temperatures as much as 16°C higher than actual temperatures, the Pi-3 system is throttling the CPU speeds when the temperature is reported to be about 82°C.

The result is that at full load, with heatsink in place, in a case, but with no fan, the Pi-3 system in the Zebra case delivers only about 85% of its capacity.

Finally, with the Pi-3 work data available, it might be interesting to see comparable data for the other systems to determine the single-thread capacity of each of the three models. This is not a comprehensive benchmark, since the sysbench CPU test involves just 64-bit integer calculations, but the comparison gives a rough estimate of the single-thread capacity of the three systems.

The following presents the time required to execute 10,000 “requests” (in sysbench terms), with max-prime=20000, for each of the three systems:

Table 4. Comparison of Work Capacity of Three Models of Pi.

Pi Model	Time to perform 10000 events, 1 thread max-prime=20000 max-requests=10000
Pi-1B	1363 sec
Pi-2B	769 sec
Pi-3	477 sec
Pi-3, 4 threads	120 sec

The single-stream performance of the Pi-3 is about 3x the Pi-1 for this particular type of computational problem. But the last row in Table 4 demonstrates that for problems that can use all four cores on the Pi-3 concurrently, it is, indeed, over 10x as fast as the Pi-1B, as the Pi Foundation claimed in its announcement of the Pi-3 model.

Conclusions

All the conclusions regarding thermal characteristics are subject to the caveat that it appears that none of the thermal sensors in the Pi models are accurate and that the Pi-3, in particular, seems to report temperatures about 16°C higher than they actually are. If that’s true in all Pi-3 units, not just the one tested here, then it suggests that the real operating temperature of the Pi-3 is about 66°C when CPU throttling begins under heavy load, and it appears that that would be a safe operating temperature. [More research by someone with an external thermal probe and thermal paste would be useful here.]

1. While the operating temperatures of the Pi-1B and Pi-2B were relatively low, even under load, the Pi-3 appears to run at a higher temperature even with no load. And under load, temperatures can exceed the thermal trip point, past which the CPU speed appears to decrease in order to reduce temperatures. The result is that the system protects itself against overheating, but the system’s work capacity is reduced by at least 15% (depending on case, heatsink) as a result.

2. The addition of a heatsink reduces the Pi-3 CPU temperature by 3°C-4°C, and for loads that are routinely <1.0, that may be sufficient to prolong the life of the Pi-3. Above loads of 1.0, a fan might provide enough additional cooling to prolong the life of the Pi-3 to justify the cost and operational nuisance (power drain, noise, etc.). Above loads of 3.0, a fan would be

needed to cool the Pi-3 enough to maintain its performance at 100% of theoretical capacity.

3. Most hobbiest and educational users probably don't need a fan, but the small additional cost of a heatsink would likely be justified. Any such cost, or the cost of adding a fan, must be balanced against the cost of replacing a Pi-3 ($35) and its anticipated lifetime (possibly 2-4 years before replacing with a new model at the same cost).

4. Pi-1B and Pi-2B users probably need neither a heatsink nor a fan.

5. The 4-core Pi-3 really does offer about 10x the work capacity of the 1-core Pi-1 and about 1.6x the capacity of the 4-core Pi-2B.

References

1. RPiTempLogger: https://github.com/hdtodd/RPiTempLogger
2. Christopher Barnatt, **Explaining Computers**, Raspberry Pi-3 CPU Temperature Tests and Heatsink (video): https://www.youtube.com/watch?v=e6okZKRwnTQ
3. Christopher Barnatt, **Explaining Computers**, Raspberry Pi 3: Fan and Cooling Tests, https://www.youtube.com/watch?v=5Ud-grj4Zl0