POWERTOP

User’s Guide

Kristen C. Accardi
kristen.c.accardi@intel.com

Alexandra Yates
alexandra.yates@intel.com

PowerTOP is a Linux* tool used to diagnose issues related to power consumption and power management. PowerTOP has an interactive mode where the user can experiment with various power management settings for cases where the Linux distribution has not enabled these power saving settings.

Contents

1 Linux Power Management Concepts

We consider the concepts in this section to be the stepping stones for understanding PowerTOP's features.

1.1 Power Management

On mobile devices, power management (PM) is an important feature that requires refinement to maximize the time a computer can run from a full battery charge. Devices and the operating system interact to find a fine balance between idle and active power consumption.

Active power is known as the power consumed by the system during an active operation. A good example of this is the power consumed by the system when browsing a web page.

Idle power is known as the amount of power consumed when no user is present. Additionally, idle power can be the amount of power consumed when not all devices are currently needed. For instance, when editing a document, it is possible to power down sections of the platform, such as the SSD or Wi-Fi, to conserve battery life.

1.2 Advanced Configuration and Power Interface (ACPI)

Advanced Configuration and Power Interface (ACPI) is an open industry specification co-developed by Hewlett-Packard, Intel, Microsoft, Phoenix, and Toshiba.

ACPI establishes industry-standard interfaces, enabling OS-directed configuration, power management, and thermal management of mobile, desktop, and server platforms.

When first published in 1996, ACPI evolved an existing collection of power management BIOS code, Advanced Power Management (APM) application programming interfaces (APIs), PNPBIOS APIs, and Multiprocessor Specification (MPS) tables into a well-defined power management and configuration interface specification.

The specification enables new power management technologies to evolve independently in operating systems and hardware while making sure that they continue to work together. [1]

1.3 ACPI C-States

The ACPI system allows only four possible C-States; C0, C1, C2, and C3. As such, the platform BIOS typically maps an ACPI C-state to a specific Core C-state. For the exact mapping on a platform, please see the specific CPU BIOS Writers Guide for that platform.

1.4 OS request of C-states

For Linux based OSs, the OS can request C-states, either using the ACPI idle driver (acpi_idle) or through the use of the intel_idle driver.

To find out what idle driver Linux is using, one can issue:

```
$ cat /sys/devices/system/cpu/cpuidle/current_driver
intel_idle
```

The intel_idle driver can request each of the individual C-states available from the Intel® processor, while acpi_idle can only request the ACPI C-states.

1.5 Intel Idle Driver

The intel_idle driver is a CPU idle driver that supports modern Intel processors. The intel_idle driver presents the kernel with the duration of the target residency and exit latency for each supported Intel processor. The CPU idle menu governor uses this data to predict how long the CPU will be idle.

1.6 C-states

When a platform enters an idle state, the CPU can be commanded by the Operating System Power Management (OSPM) to enter a low power state. These states, known as CPU C-states, operate on a per-core basis, as requested by the Operating System Power Management (OSPM). Each state defines the degree to which a core can be put to sleep (powered down). C0 is used to indicate a fully operational and "awake" processor. All C-states (C1 – Cn) build upon the functionality of the previous level by adding different parts of the processor to power down.

1.7 Package C-states:

Package C-states, often referred to as PC-states or PCx, happen when all the platform cores agree to enter a specific C-state. The notation is for the highest possible C-state. In a four core system, if one core is at C3, and the others at C6, the Package C-state will be PC3.

1.8 Linux P-State Control - intel_pstates vs. cpufreq

Modern CPU architecture includes a feature to enable the operating system to scale the CPU frequency. Typically, this works by scaling the frequency down to save power, while scaling the frequency up on intense workloads. In some cases, where power is constrained, it may be worth clamping the maximum frequency a system can perform.

Starting with the 3.10 kernel, the Linux kernel has included the intel_pstate driver. This driver enumerates against the hardware capabilities of the CPU, instead of depending on the more limited ACPI enumeration. Because of this very basic change, the intel_pstate driver includes the control algorithms specific to a CPU. This causes the system to judge more accurately what P-state a platform should be in, while maintaining a superior battery life.

How to determine if you're using CPUFreq or intel_pstate:

```
cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_driver
```

The intel_pstate controls can be found at:

```
cd /sys/devices/system/cpu/intel_pstate
```

SYSFS entry	Valid Values	Description
no_turbo	0 – 1	Enables or disables the CPU Turbo feature. A value of 1 will disable turbo
max_perf_pct	0% - 100%	A percentage value of the maximum possible CPU speed. By default this is 100%. A value of 0% will set the system to LFM.
min_perf_pct	0% - 100%	A percentage value of the minimum possible CPU speed. By default this is 20%. A value of 100% will set the system to HFM.

2 PowerTOP Data Interpretation

2.1 Overview tab / Summary

The overview tab lists the top power consuming items which keep waking a processor from its idle state. When tuning applications and device drivers for power, the idea is to reduce the number of wakeups/second to maximize the system's power performance. This tab shows the usage, number of events, category, description, and power estimate of the most consuming power items in the system.

There are two requirements that must be met for the power estimate to show in this list.

1- Check that there are not files under

```
$ ls  /var/cache/powertop/
```

2- The system has to be running on battery power only. Not connected to the wall power.

3- Check that the Power Capping Framework and Running Average Power Limit are enabled. Most recent Linux distributions enable by default.

```
$ ls  /sys/class/powecap/intel_rapl
```

RAPL support allows PowerTOP to provide more accurate power measurements. If RAPL is not present then, change the RAPL setting on the linux .config file, recompile, and reinstall Linux kernel. Note that RAPLE was released on Linux 3.13 [2].

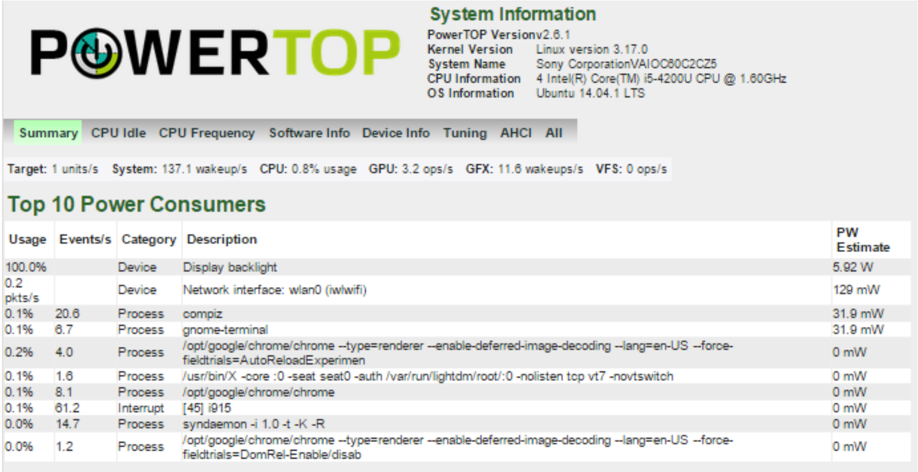
```
$vim .config
CONFIG_POWERCAP=y
CONFIG_INTEL_RAPL=y
```

4- PowerTOP needs to execute for a minimal number of times to allow a good fitness model calculation. After version 2.7 PowerTOP shows the required number of times to execute along with the number of measurements that already accumulated. The data is stored on saved_parameters.powertop and saved_results.powertop both under /var/cache/powertop/.

To speed things up one could execute a bash program that runs PowerTOP for the required number of times.

```
for (i=0; i < minimum_runs; i++)
    sudo powertop –time=10 --html
```

This will provide



System Information

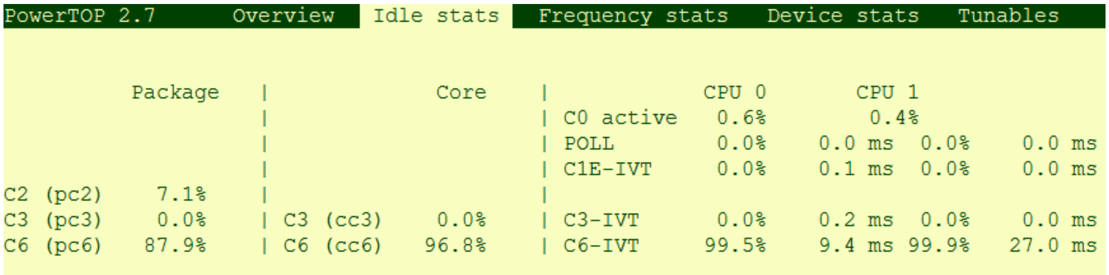
PowerTOP Version	v2.6.1
Kernel Version	Linux version 3.17.0
System Name	Sony CorporationVAIOC60C2CZ5
CPU Information	4 Intel(R) Core(TM) i5-4200U CPU @ 1.60GHz
OS Information	Ubuntu 14.04.1 LTS

Target: 1 units/s System: 137.1 wakeup/s CPU: 0.8% usage GPU: 3.2 ops/s GFX: 11.6 wakeups/s VFS: 0 ops/s

Top 10 Power Consumers

Usage	Events/s	Category	Description	PW Estimate
100.0%		Device	Display backlight	5.92 W
0.2 pkts/s		Device	Network interface: wlan0 (iwlwifi)	129 mW
0.1%	20.6	Process	compiz	31.9 mW
0.1%	6.7	Process	gnome-terminal	31.9 mW
0.2%	4.0	Process	/opt/google/chrome/chrome --type=renderer --enable-deferred-image-decoding --lang=en-US --force-fieldtrials=AutoReloadExperimen	0 mW
0.1%	1.6	Process	/usr/bin/X -core :0 -seat seat0 -auth /var/run/lightdm/root/:0 -nolisten tcp vt7 -novtswitch	0 mW
0.1%	8.1	Process	/opt/google/chrome/chrome	0 mW
0.1%	61.2	Interrupt	[45] i915	0 mW
0.0%	14.7	Process	syndaemon -i 1.0 -t -K -R	0 mW
0.0%	1.2	Process	/opt/google/chrome/chrome --type=renderer --enable-deferred-image-decoding --lang=en-US --force-fieldtrials=DomRel-Enable/disab	0 mW

2.2 Idle Stats tab



```
PowerTOP 2.7      Overview   Idle stats   Frequency stats   Device stats   Tunables

          Package    |            Core     |           CPU 0       CPU 1
                     |                     | C0 active   0.6%        0.4%
                     |                     | POLL        0.0%    0.0 ms  0.0%    0.0 ms
                     |                     | C1E-IVT     0.0%    0.1 ms  0.0%    0.0 ms
C2 (pc2)    7.1%     |                     |
C3 (pc3)    0.0%     | C3 (cc3)   0.0%     | C3-IVT      0.0%    0.2 ms  0.0%    0.0 ms
C6 (pc6)   87.9%     | C6 (cc6)  96.8%     | C6-IVT     99.5%    9.4 ms 99.9%   27.0 ms
```

The idle stats tab presents the CPUs and GPUs currently loaded in the system in relationship with their C-states. Intel platforms show the list of C-states, broken up by CPU, by core, and by package. Simplistically, a core can contain one or more CPUs, A package contains one or more cores. To reach a system idle state, a deep package C-state must be attained. This happens when all CPUs and GPUs reach the deep idle state.

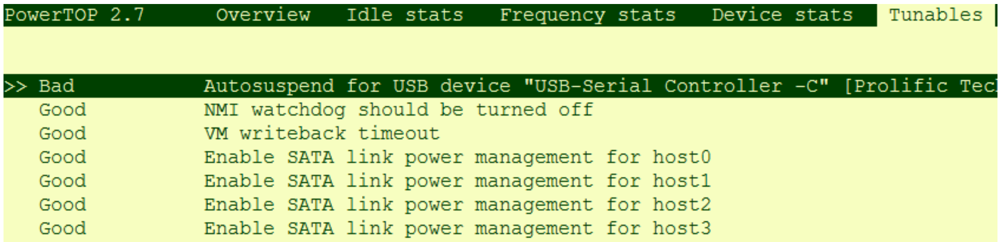
2.3 Frequency Stats tab

The frequency stats tab presents the P-states of a system in relationship with the idle state. To see the different P-states supported on an Intel processor, it is recommended to run the Intel idle driver. Additionally, the governor has to be set to a different setting other than performance, or PowerTOP will show only the maximum performance run by the CPU.

2.4 Device Stats tab

The device stats tab presents the list of devices in the system that consume the most power.

2.5 Tunables tab



```
PowerTOP 2.7      Overview   Idle stats   Frequency stats   Device stats   Tunables |

>> Bad           Autosuspend for USB device "USB-Serial Controller -C" [Prolific Tec
   Good          NMI watchdog should be turned off
   Good          VM writeback timeout
   Good          Enable SATA link power management for host0
   Good          Enable SATA link power management for host1
   Good          Enable SATA link power management for host2
   Good          Enable SATA link power management for host3
```

This tab lists the devices that are present on the system. Devices that are not tuned for power management have the “Bad” label associated. Otherwise, the label reads “Good”.

In this tab, one can tune the system to be power friendly by toggling each item from bad to good. When toggling the item, the screen shows the command used to tune the system. Usually, this is a shown as a change on sysfs.

Keep in mind, these settings are not permanent. The system will reset back to "Bad" power performance after restarting the machine. To permanently set the system to the optimal power management settings, you can add .bashrc to the command(s) listed when toggling the setting from bad to good. Make sure you test each setting separately because they can introduce instabilities on your system.

If all settings work on the system, you can simplify the list of commands by using:

```
$powertop –auto_tune.
```

For more information on Linux kernel power management features, please see: https://www.kernel.org/doc/Documentation/power/

3 PowerTOP's Features

3.1 Interactive mode

```
$sudo powertop
```

This is the default mode to execute PowerTOP. It opens an n-curses interface that hosts PowerTOP's interactive mode. Notice that PowerTOP needs super user privileges. To navigate, use the tab key to visit the different menu items. Use the arrow keys to navigate vertically on a page and the enter key to change configuration of the items under Tunables.

3.2 HTML mode

```
$sudo powertop --html[=FILENAME]
```

This feature executes PowerTOP and stores the result in the powertop.html file, if no name is given. This mode is often used to send emails investigating power management issues.

3.3 CSV mode

```
$sudo powertop --csv[=FILENAME]
```

This feature executes PowerTOP and stores the result in the powertop.csv file, if no name is given. This mode is often used to send emails investigating power management issues.

3.4 Debug mode

```
sudo powertop --debug
```

The debug setting runs the power measuring algorithm for each device for 750 iterations. It then applies the least square algorithm to identify the fitness of the measurements. It prints the list of devices with their respective score. The scores are used to calculate the overall power usage for each device.

3.5 Calibrate mode

```
sudo powertop --calibrate
```

Calibrate mode measures power for a set of runs using different idle settings for USB devices, radios, backlight, wifi, disk, and the CPU. This feature is useful when identifying the most optimal brightness setting on your laptop computer screen when using only battery power.

3.6 Auto-Tune mode

```
sudo powertop --auto-tune
```

This feature sets all tunable options to their GOOD setting without additional user intervention. Keep in mind that this settings will be reset after you boot the system.

3.7 Workload mode

```
sudo powertop --workload[=WORKLOAD]
```

This mode is used to execute a workload and identify the power consumption of the system during the execution of the workload. This is a useful feature when running benchmarks. The WORKLOAD is the binary file used to execute the workload.

3.8 Extech mode

```
sudo powertop --extech[=DEVNODE]
```

This mode allows you to execute PowerTOP when the system is connected to an Extech power analyzer.

3.9 Options

These options can be used with most of the previous modes.

--version	Prints PowerTOP's version information.
--time[=x]	Generates a report for 'x' amount of seconds.
--iteration[=x]	Executes PowerTOP tests for 'x' number of iterations.
--quiet	Suppresses stderr output.
--help	Print this help menu.

4 References

[1] ACPI Reference Manual http://www.acpi.info/
[2] Linux Kernel Release Notes http://kernelnewbies.org/Linux_3.13