

# System Sensors

Your system likely has many sensors built in for displaying useful information on internal hardware status. For example, the commands below will help in finding the path to system temperature sensors.

```
user@host ~ $: sensors -f
coretemp-isa-0000
Adapter: ISA adapter
Package id 0: +80.6°F (high = +176.0°F, crit = +212.0°F)
Core 0:      +73.4°F (high = +176.0°F, crit = +212.0°F)
Core 1:      +73.4°F (high = +176.0°F, crit = +212.0°F)
Core 2:      +69.8°F (high = +176.0°F, crit = +212.0°F)
Core 3:      +68.0°F (high = +176.0°F, crit = +212.0°F)

acpitz-acpi-0
Adapter: ACPI interface
temp1:      +82.0°F (crit = +221.0°F)
temp2:      +85.6°F (crit = +221.0°F)

nouveau-pci-0100
Adapter: PCI adapter
GPU core:   +0.97 V (min = +0.60 V, max = +1.27 V)
fan1:      691 RPM
temp1:      +89.6°F (high = +203.0°F, hyst = +37.4°F)
              (crit = +221.0°F, hyst = +41.0°F)
              (emerg = +275.0°F, hyst = +41.0°F)
power1:     36.13 W (crit = 275.00 mW)

asus-isa-0000
Adapter: ISA adapter
cpu_fan:    0 RPM
```

We can see that the CPU and GPU temperature sensors are known to our system as `coretemp-isa-0000` and `nouveau-pci-0100`, respectively. Run the command below to list the system path to all connected temperature devices by name, and cross-check these two outputs to gather the needed information for your sensors.

```
user@host ~ $:for i in /sys/class/hwmon/hwmon*/temp*_input; do echo "$(<$(dirname $i)/name): $(cat  
${i%_*}_label 2>/dev/n  
ull || echo $(basename ${i%_*})) $(readlink -f $i)"; done
```

```
acpitz: temp1 /sys/devices/virtual/thermal/thermal_zone0/hwmon0/temp1_input  
acpitz: temp2 /sys/devices/virtual/thermal/thermal_zone0/hwmon0/temp2_input  
coretemp: Package id 0 /sys/devices/platform/coretemp.0/hwmon/hwmon2/temp1_input  
coretemp: Core 0 /sys/devices/platform/coretemp.0/hwmon/hwmon2/temp2_input  
coretemp: Core 1 /sys/devices/platform/coretemp.0/hwmon/hwmon2/temp3_input  
coretemp: Core 2 /sys/devices/platform/coretemp.0/hwmon/hwmon2/temp4_input  
coretemp: Core 3 /sys/devices/platform/coretemp.0/hwmon/hwmon2/temp5_input  
nouveau: temp1 /sys/devices/pci0000:00/0000:00:01.0/0000:01:00.0/hwmon/hwmon3/temp1_input
```

## Displays

When attempting to manage displays, whether its the orientation or enabling / disabling, look to the man pages for `xrandr`. See the commands below for some examples.

```
# Output information on displays
```

```
xrandr
```

```
# List the output names for displays
```

```
xrandr --output
```

```
# Move DP-2 to the right of HDMI-1
```

```
xrandr --output DP-2 --right-of HDMI-1
```

```
`
```

```
#### Timezone
```

```
To see date / time, run `date`
```

To adjust local TZ settings, run `tzselect`. Pay attention to the final output of this tool as it will explain how to make your change permanent. For me, I had to add the following to the end of my `~/.profile` :

```
```bash
```

```
TZ='America/New_York'; export TZ
```

## Memory

Some useful commands to find information on memory usage -

```
# Output various memory details
cat /proc/meminfo

# Can be used with grep, awk, etc for more specific output..

# ex) Show MiB of memory available
grep -w MemAvailable: /proc/meminfo | awk '{print $2 / 1024 "MiB"}'
```

## Input Devices

Run the following to get information on input devices attached to the machine -

```
# In the output shown below, my keyboard is AT Translated Set 2 keyboard
xinput list

# Example output:
[ Virtual core pointer                id=2  [master pointer (3)]
 | ↳ Virtual core XTEST pointer        id=4  [slave pointer (2)]
 | ↳ Elan Touchpad                     id=10  [slave pointer (2)]
[ Virtual core keyboard              id=3  [master keyboard (2)]
 | ↳ Virtual core XTEST keyboard        id=5  [slave keyboard (3)]
 | ↳ Power Button                      id=6  [slave keyboard (3)]
 | ↳ Power Button                      id=7  [slave keyboard (3)]
 | ↳ Sleep Button                      id=8  [slave keyboard (3)]
 | ↳ TOSHIBA Web Camera - HD: TOSHIB   id=9  [slave keyboard (3)]
 | ↳ AT Translated Set 2 keyboard       id=11  [slave keyboard (3)]

# Test the device..
xinput test "AT Translated Set 2 keyboard"

# Example output:
key release 36
key press  40
dkey release 40
key press  50
key release 50

# The output above shows me pressing / releasing keys in real time.

# Exit with CTRL-C
```

## Power Supplies / AC Adapters

```
# List power supplies, AC adapters -
ls -l /sys/class/power_supply/

# Example output...
lrwxrwxrwx 1 root root 0 Mar 23 23:02 AC ->
```

```
.././devices/LNXSYSTM:00/LNXXSYBUS:00/PNP0A08:00/device:00/PNP0C09:00/ACPI0003:00/power_supply/AC
lrwxrwxrwx 1 root root 0 Mar 23 23:02 BAT0 ->
.././devices/LNXSYSTM:00/LNXXSYBUS:00/PNP0A08:00/device:00/PNP0C09:00/PNP0C0A:00/power_supply/BAT0
# Above, my battery is seen as BAT0, my AC port for charging is AC
```

---

Revision #6

Created 28 September 2019 10:18:15 by Shaun Reed

Updated 20 December 2021 22:14:22 by Shaun Reed