VMW Group Power Measurement Techniques

Vincent M. Weaver
University of Maine
Electrical and Computer Engineering
vincent.weaver@maine.edu

Abstract

This document gives a background on common ways of measuring power consumption in desktop and server computer systems, and describes how the VMW Group at the University of Maine makes power measurements.

1 Introduction

Power consumption of computing systems in an increasingly important metric. In embedded systems low power is important, especially in mobile devices such as cellphones. Large compute clusters in high-performance computing can consume as much power as small towns, so optimizing performance is important. Normal users care about their power bills and would like to use as little as possible.

Conducting actual power measurements of systems is tricky. Most machines do not include power measurement, and the ones that do are often low-resolution and of unknown quality. Custom instrumenting a system can be difficult and often involves a lot of intrusive changes to the system, up to and including cutting wires.

We look at various challenges we encountered when trying to instrument various server and desktop machines.

2 System-Wide Power Measurement

The easiest way to obtain power measurements is to use some sort of system-wide overall measurement, usually at the power-outlet level.

2.1 Watts Up Pro

The Watts Up Pro [14] (as seen in Figure 1) measures power at the wall outlet and allows logging it over USB. Its primary limitation is that it only updates at 1Hz so while it can give a rough overview of full-system power consumption it cannot provide detailed measurements.

Figure 1: WattsUpPro hooked to an ARM development board. TODO: get a better picture.

2.1.1 Software

There are various ways to gather that power measurements.

- **Windows**: There are various packages available from the product website. Past researchers in our group have used these tools to download .csv files for analysis, but I am not sure if any gave sufficient details on how they did this (TODO: check on this).
- **Linux**: The website provides documentation and some code to gather results with Linux. Sometimes the provided code will manage to lock up the device, requiring a power-cycle. I have developed some simple code that purely logs the time and power values. This can be obtained here: `git clone https://github.com/deater/uarch-configure.git` in the wattsup directory, the `wattsup-simple` tool.

- **PAPI**: There are plans for a PAPI component but that hasn't happened yet.

2.2 IPMI

Many server systems have embedded processors that monitor power usage at the power supply level. Under Linux you can use the `ipmitool` to look at this.

See Chad's Thesis [43] for some info on trying this on various of our servers. It possibly was not working all that well, but it merits another more detailed look.

2.3 Embedded Boards

We have a wide variety of ARM embedded boards (Raspberry Pi, Beaglebone, Pandaboard, Odroid, etc).

The easiest way to get power for these is often with something like the WattsUpPro.

Some boards, like the Raspberry Pi, run on USB power and so a USB interceptor (as described in Section **??**) can be used instead.

Some devices have test points where you can measure the current. Even better, the Beagleboard has such a test point with a sense resistor across it which is hooked to an internal ADC so you can measure the power internally. Philipp (cite?) measured this but found the results weren't really that great.

Devices like cellphones usually you have to intercept between the battery and the device. We have not done measurements like that yet.

See our MDPI Electronics paper and our website for a wide variety of power and FLOPS/W readings for a large number of embedded boards, gathered with the WattsUpPro.

3 Power Estimates

If actual power measurement is difficult, there are various ways to estimate power. A common way is to create some sort of model based on something that is easy to measure (such as hardware performance counters). There is a lot of related work on this (cite some of it).

In addition, some modern hardware supports estimating in hardware, as described below.

3.1 RAPL

Recent Intel chips support RAPL which provide the results from a power model running on a helper chip. Some Haswell server machines provide results that are actual measurements, not estimates.

TODO: describe Link to my RAPL pages Describe the 3 ways to read the values out on Linux Give examples on how to use the RAPL tool on Linux

3.2 AMD APM

AMD's APM (???) is similar to RAPL but with lots more limitations.

Our current set of Piledriver servers do not seem to support this well.

TODO: add more info

4 Detailed Power Measurement

Conducting detailed power measurement is tricky. In general you will need the following:

- A power sensor (usually extrapolated from a current sensor)
- An amplifier to amplify the signal
- Some sort of meter or analog/digital converter (ADC) to log the signal.

4.1 Sensor Types

4.1.1 Sense Resistors

You can calculate the power draw using sense resistors according to Ohm's Law. P=IV where P is Power, I is current and V is the Voltage. You can calculate the current using a sense resistor; also Ohm's Law: V=IR where V is voltage, I is current and R is resistance. So if you can measure the voltage drop across the resistor, you can calculate the current using I=V/R.

You should use as accurate a resistor as possible 1% or better.

Use a small resistor. If your resistor is too big (or the current too high) the voltage drop will be high. For example, if you are measuring a device that draws 1A of current and you try to use a 1Ω resistor, the voltage drop across it will be 1V; if you only have 5V to start with dropping the voltage to 4V will probably make your circuit not work.

Also make sure the power rating of the resistance is high enough. Note that the power doesn't have to be for the full power of the circuit; use P=IV but in this case the power dissipated is current times the voltage *drop*, not the full voltage, so it will be many times less than the full power being measured.

Choosing a proper resistor size can be tricky, if too small it can be hard to measure the voltage without an amplifier, too big and the voltage drop will interfere with the circuit.

Be aware with small resistors that a regular volt meter will have trouble measuring it, so you sometimes will have to believe the resistance printed on the side rather than trying to measure it unless you have a specially calibrated DVM that can measure small resistances.

4.1.2 Hall-Effect Sensors

These can use magnetic fields to measure current and thus you do not have to worry about introducing voltage drops into the circuit.

We use the ACS715 (cite) on a braekout board from Pololu. To determine the current, you measure the voltage out, subtract 500mV and then it is 133mV per Amp.

Datasheet isn't really that helpful.

4.2 Sensor Boards

4.2.1 ATX: Motherboard Power

Most modern desktops use the ATX power connector which has 24 pins coming from the power supply providing power.

Ian and Juan were manually cutting and twisting the individual wires.

I've constructed a custom board to make this better, details soon.

Ian and Juan were not able to isolate which pins powered which parts of the motherboard. We have continuing work trying to figure this out.

4.2.2 CPU Power: P4 Connector

The most common way to measure CPU power is to take the P4 connector (an additional set of 12V lines coming from an ATX power supply) and measure that. These lines are usually fed into the CPU power conversion circuitry, but often other things get powered by it too so its not a perfect measure.

For example in Figure **??** shows a picture from one of our Haswell motherboards where it says the P4 connector powers things besides just CPU.

See related work for other groups trying to measure CPU power.

4.2.3 DRAM: DIMM riser card

We have one, the JET-5464 DDR3 DIMM Extender with Current measurement. `http://www.mfactors.com/jet-5464-ddr3-dimm-extender-with-current-measuring/`

It is hard to buy. Have trouble sourcing another, and a DDR4 one. We tried to make our own (much cheaper) but not work.

Have recently obtained DDR4, DDR3-SODIMM, and another DDR3. (TODO: types and info)

4.2.4 SATA interceptor

Spencer made one of these, with a sense resistor across the SATA lines. Measuring 12V can be tricky.

4.2.5 HP Server connector

The HP enterprise servers we use have custom connections between the power supply and motherboard (they are not standard ATX connectors).

See Chad's thesis [43] for the details on how he constructed intercept boards to measure the power.

Also note that we managed to fry our sandybridge-EP server while using these boards.

4.2.6 USB interceptor

Measuring the power consumption of USB devices is fairly straightforward as you just have to intercept the 5V line. This will probably get more difficult with the recently introduced USB-C standard.

In addition to measuring the power of USB devices (keyboards, mice, hard-drives, memory keys, etc) many embedded systems (such as the Raspberry Pi) are also powered by USB.

You can make your own board by splitting a USB cable, putting a sense resistor across 5V, and then measuring the drop. We have done this (TODO: include a picture).

You can also obtain various pre-made USB power meters. Often they just have displays, rather than any way of logging. We do have the adafruit "USB Power Gauge" `https://www.adafruit.com/products/1549`. It has a 0.1 ohm sense resistor, an INA169 current sensor, and an ATtiny85 microcontroller that monitors everything. There are LEDs which light up to indicate power (useless for our purposes and possibly counterproductive, but pretty). In a useful feature for us, it will log out the power over a serial connection at 9600 baud.

To do this, hook up a USB-serial connector. For the ones we have, hook the black wire to the ground pin and the white wire to the TX pin. Use minicom (on Linux) or putty (windows) to gather the results.

Limitations: it is unclear how often/regular the outputs happen? Also adafruit warns that the accuracy is at best +/-0.1V and 50mA due to noise, thermal changes, etc.

4.2.7 PCIe interceptor

Useful for modern disk and GPUs. We have obtained one but have not tried it yet.

4.2.8 GPU Power

Many devices (especially Nvidia ones) provide built-in power meters. We've used this in ECE574 (TODO: provide details here).

Also we can try using a PCIe interceptor as described above.

4.3 Amplifiers

Sometimes the output from the sensors is small enough that it is in the noise area of the A/D converter. For example, the DIMM measurement typically only varies between 1 - 15mV. To get a better signal, an amplifier can be used to amplify the signal. However care must be taken as the amplification stage can introduce its own source of error.

4.3.1 MCP6044

The simplest amplification is via an op-amp, generally a differential amplifier circuit. We have been using MCP6044 op-amps [39]. Specifically the differential amplifier circuit described in the microchip documentation [3]. The circuit used is summarized in Figure **??**. We also try to use this in a single-supply setup as it can be difficult to obtain a negative supply, as most digital systems do not provide those voltages.

Our Raspberry Pi monitoring boards use the circuit shown, as well as 200M and 10k 1% resistors to generate a gain of 20 (actual measured gain is closer to 17). We have done some minimal validation of the outputs using a function generator and oscilloscope.

TODO: calculate the error bars.

Include circuit.

4.3.2 Instrumentation Amplifier

For some experiments we have been using an INA122 Instrumentation Amplifier [6]. An instrumentation amplifier is similar to a difference amplifier, but is made out of multiple op-amps to eliminate the need for worrying about the input impedeance.

The INA122 is nice in that it will run off a single supply, and the gain is configurable with an external resistor.

We've been using the INA122U package, which is surface mount, and then soldering them (I think in a toaster oven) to breakout boards.

Some things to watch for: you cannot drive the outputs rail-to-rail. The specs say the lowest you can reliably output is (V-)+150mV and the highest is (V+)-100mV. So with a single-supply 5V setup, you can only have reliable gain if the output is between 150mV to 4.9V. We actually ran into this problem measuring the values of the DRAM adapater (with 3.33mΩ resistance); if the gain was only 18x then the output was lower than 150mV and the actual gain measured was a factor of two too big. Setting the gain to 300x and having the output above 150mV made the gain behave as expected. This explains some of the issues in Chad's thesis [43].

We've currently been using just plain 5% resistors for setting the gain, ideally we'd get some fancier 1% or better in the proper values.

4.4 Power Logging Devices

Once you have the sensors set up, you want some way of logging the values returned. Ideally you could find a cheap device that could record many channels at high frequency. In practice things get expensive really fast if you want good results.

4.4.1 Measurement Computing A/D

We have a Measurement Computing USB-1208FS-Plus data aquisition device [37]. It can plug in over USB and provides a number of inputs and outputs; our primary interest is in the Analog inputs.

Software

Windows: Various tools that past students have used, need to get documentation.

Linux: This is way more difficult than it has any right to be. The way I currently do it is a custom piece of software I wrote (will post it soon) that uses the DaqFlex interface. It is based on the tool here `https://github.com/kienjakenobi/daqflex`.

Other approaches that are possible:

- MC themselves link to a guy in North Carolina that has a library for accesing the boards. `ftp://lx10.tx.ncsu.edu/pub/Linux/drivers/USB/` The sample code works and can dump some data, but there are no tools that use the library, just limited selftests. Building the code is a pain too as it requires various libs.

- There is also a C# Mono tool that uses the daqflex interface. `http://www.mccdaq.com/daq-software/DAQFlex.aspx` Cite the whitepaper. Interesting project here: `http://wind.cs.purdue.edu/doc/mcc_daq_testing.html` This is a special firmware on the boards that responds to a series of text commands to configure the board. Again, once you go through the complicated setup of installing mono yo can get some sample tools, but they give you a gui that displays current values but no good way to do long-term recording and output to CSV.

  In the past I hacked up the ainscan tool to dump the data, but can't find that work again.

- Finally, there are some C++ tools that talk to the daqflex interface. The one I've been using is a modified one from github (link) that will read the values and output to a CSV file. How to use DAQFlex directly from C++ `http://kb.mccdaq.com/KnowledgebaseArticle50047.aspx` and `https://github.com/kienjakenobi/daqflex`. I use something similar but written in C

4.4.2 Teensy

Chad developed a Teensy board that could for low-cost provide many A/D channels that can then be read via a USB/serial connection. Mao has continued this work. NEED MORE DETAILS.

4.4.3 SPI A/D Converter

The SPI A/D converter used in the Raspberry Pi cluster. More details.

4.4.4 Logging Voltmeter

The voltmeter in the lab can in theory measure voltages regularly and output over USB. I don't think anyone has tried it out yet.

4.4.5 Logging Oscilloscope

The oscilloscope can in theory also measure and log the values.

4.5 Pitfalls

4.5.1 Synchronizing Measurements

If measuring things on different machines need some way to synchronize measurements.

Using a serial port.

Using ntp.

5 Results

Plotting the results.

Usually I use C programs to parse the raw data, then generate plots using jgraph.

Some people use Octave/MATLAB but I wouldn't recommend that.

Using LibreOffice and a spreadsheet you can also generate graphs, possibly quickly from a CSV file.

6 Future Work

We plan to continue trying to validate and measure the power on a wide variety of systems.

7 Related Work

TODO: List all papers by our group that have measured power.

Power is an important metric and researchers have been wanting accurate power readings for years. There are various ways to do this; the most common are estimation and instrumentation. In estimation the power usage is estimated based on some metric, usually the hardware performance counters. In instrumentation the system has probes added to measure actual power readings. There are also hybrid approaches, such as RAPL, where hardware readings and software estimates are combined in real time.

In this document we concentrate on looking at instrumenting various machines.

The total power in a system is made up of various components, including the CPU, the RAM, any disks, GPUs, network adapters, and any other I/O. Additional sources of power loss are power distribution, power-supply inefficiency, and cooling.

7.1 CPU Power Measurement

CPU power is difficult to measure directly as the lines connecting the power supply to the CPU are often not directly broken out and thus hard to access. For this reason it is often more common to estimate CPU power usage rather than directly measure it.

According to some Intel documentation [26], the 4-conductor auxiliary (P4) 12V ATX connector is dedicated to the CPU Voltage. Often other devices are powered from this line; we have a board that labels the connector as "CPU/USB/NIC" and others have found that it may include the DRAM usage.Measuring this line is the most common way of measuring power. A more destructive way of measuring power involves tapping into the voltage regulator circuitry or even cutting traces on the motherboard to insert sense resistors.

Porier et al. [45] describe the "Foxton Technology" on an Itanium processor that has an embedded processor with four A/D converters that measure temperature and power (using an on-die resistor) every $8\mu s$. The measurements are used for frequency scaling and are not exposed to the user.

Mesa-Martinez et al. [38] measure the temperature of a CPU in detail using an oil bath and an infrared camera. They then estimate the power consumption based on thermal measurements. They validate this against power readings taken with a multi-meter (compensating for a 10% loss in the CPU voltage regulator circuitry). Hamann et al. [22] do similar work with estimating power from temperature.

Eamsailzadeh at al. [15] measure 61 benchmarks across 8 processor types using the 12v line.

7.2 DRAM Power

The power consumption of main memory is of interest, although usually it is overlooked since it tends to be much smaller than the power used by the CPU. Power is not often measured; more common is to simulate it and use values from datasheets to provide a basis for the models. When actual power is measured it is usually done by modifying a DIMM extender board to have a sense resistor.

Gottscho et al. [18, 19] measure DRAM power using a 2Ω sense resistor and a digital multimeter sampling at 10 samples/second. Rahmati et. al [46] measure DRAM power using a 200Ω sense resistor and an A/D board.

The RAPL DRAM interface was first described by David et al. [10]. While concentrating on power-capping, they do describe in detail the underlying power model which presumably is similar to that found in modern Intel chips. A parametric model is built using genetic algorithms based on various inputs and the weights are calibrated by the BIOS as boot. They validated against real hardware using a DIMM riser card and a data acquisition board sampling at 100Hz. They found accuracy of 1% when using a Nehalem server system and a DDR3 1333 4GB memory module.

Khanna et al. [28] describe the weights used in RAPL DRAM measurements. They measure actual DRAM results using a riser with a 5mΩ sense resistor sampled at 100Hz. They find RAPL results within 2.3% of actual measurements.

7.3 GPU Power

On modern computing systems the graphics processor (or GPU) can use a large amount of power. This is especially true of machines used for gaming, as well as supercomputers where GPGPU work is done (large calculations done on the graphics device).

Measuring the power involves intercepting the PCIe power lanes and measuring the current. Some GPUs, such as high-end NVIDIA cards, support the NVML library [42] which can provide power measurements. Also Intel RAPL can also provide GPU power estimates for GPUs integrated into the CPU package.

In this work I do not measure the GPU power.

7.4 Hard Drive Power

Storage devices also consume power. In typical desktop systems the power usage is overshadowed by the CPU power, but in situations with large number of disks (such as a RAID arrays) the power consumption can be significant. Measuring the power often involves intercepting the power connectors as they come into the disk; hard drives are a lot easier to instrument for power than most of the other hardware components. It is also possible to estimate power usage. Often operating system metrics are used to determine when I/O happens, and models are constructed of the various moving parts in the drive based on actual measurements.

Hylick et al. [25] measure in detail power consumption of 10 different hard drives by placing .02Ω resistors on the power lines and logging 12-bit values at 1100Hz. Lee et al. [33] measure power consumption using Hall Effect sensors at 16-bit resolution. Yan et al. [51] measure power of hard drives using the DEEP/LEEP framework [48, 47] which provides samples at 10kHz.

In this work I do not measure hard drive power values.

7.5 Full-system Measurements

The easiest way to obtain full-system measurements is to measure the A/C power at the wall outlet. Off-the-shelf tools such as the WattsUpPro? [14] meter can yield a decent estimate for total system power usage but fail to expose subsystem measurements. The WattsUpPro also has poor temporal resolution, only recording one measurement per second.

Some server nodes provide similar low-frequency power measurements via the IPMI subsystem, which can often be accessed either remotely or using helper utilities as described in Section **??**.

I break the full-system measurement methods out into those of server systems and those of embedded systems.

7.5.1 Servers

The Powerpack [17] project describes a fully instrumented x86 cluster. PowerMon2 [4] involves small boards that can measure 8-channels of the ATX power supply at high resolution. Hackenberg et al. [20] describe many cluster measurement techniques, including RAPL.

LEAP [48] is most similar to my work. They instrument an Intel Atom low-end machine and gather power measurements of the CPU (by tapping into the power converter), DRAM (with a DDR2 extender and sense resistor), and hard drive (.01Ω sense resistor), and provide results over the serial port that can be time correlated with execution time. It uses a dedicated A/D converter for providing the measurements at 10kHz.

Cui et al. [9] instrument a full system. They measure disk (with a 0.02 ohm resistor and a 50x amplifier) at 20us resolution. They also measure the CPU via the 12V lines, DIMMs with an 0.02 ohm resistor, and the network and video cards with a PCI extender. They use a digital multimeter to gather results.

Power on servers can also be estimated. Economou et al. [13] look at performance in a server; they use micro-benchmarks to determine rough power consumption and then generate estimates for the various components in a system (CPU, RAM, hard-drive, etc.) Lee and Brooks [32] use machine learning to predict the power usage of a full system.

Bircher and John [5] generate models to estimate full system behavior using performance counters. They look at estimating Disk, DRAM, and I/O power usage based on various CPU counters and validate using various benchmark suites (SPEC2k, dbt-2, SPECjbb). They use sense resistors for instrumentation with the data recorded for later analysis, synchronized with a pulse over serial port.

Various papers look at the various ATX wires in an attempt to find out which powers what component [17, 16, 7]. Chen et al. [7] use a multimeter to measure CPU power (using the dedicated ATX line) and the "brown wires" on the ATX connector to measure memory and a WattsUpPro for full system power measurement. Feng et. al [16] measure the power consumption on a cluster. They instrument the machines at the ATX level with 0.1 Ohm resistors and logging multimeters. They find the CPU powered by 5V pins, memory and others through 3.3V pins, and the 12V pins primarily for fans. Castaño et. al [41] measure the 12V ATX power supply lines and use those to create a model for predicting full system behavior, including CPU and disk activity. Khoshbakht and Dimopoulos [29] investigate system power, measuring current on the ATX CPU power connector with an acquisition device sampling at 15k samples/second (which is then smoothed) and then comparing against the MARS86 simulator. Diouri et al. [12] validate a variety of power meters that measure A/C power (OmegaWatt, WattsUpPro?) and ones that measure ATX power lines (PowerMon2, NI and DCM). They use the pmlib framework. They find the external meters give similar results, but the internal ones vary with PowerMon2 giving different results than the others. Piga et al. [44] use hall-effect sensors to investigate in detail the power consumption of all of the ATX lines.

Mahesri and Vardhan [34] break down the power consumption of a laptop using an oscilloscope and a clamping current probe.

The IBM BlueGene series of supercomputers is based off of an embedded design and often ranks highly in the Top Green 500 supercomputers list [1]. The BluegGene P and Q systems provide an interface for measuring system power [52, 50, 23].

Laros et al. [30] utilize the i2c connection to the voltage regulator board on Cray computers to gather detailed power information.

Economou et al. [13] propose Mantis which is a detailed full-system power estimation environment that is based on results gathered from an instrumented server blade. They measure the various power planes, and split the 12V plane to add a sense resistor to enable splitting of memory and CPU values.

Hsu and Poole [24] describe power measurement details for supercomputing clusters starting with the utility connection down to the individual processors.

In this work not only do I measure the ATX power on desktop systems, but I also measure the power on the custom HP connector. I have not found any reference of others attempting to measure power on this common type of server power supply.

7.5.2 Embedded Systems

Power is not only important for servers, but also in embedded systems. In my work I restrict my measurements to desktop and server machines, but the same measurement tools I develop could be used on embedded systems.

Typically on embedded systems it is hard to access the internal power lines (unless the board designer provided test points). Therefore you are usually limited to full-system measurements by tapping into the input power line. Some boards, such as the Beagleboard, provide a sense resistor connected to a built-in A/D converter allowing power measurements. However most embedded boards have a System-on-Chip (SoC) design where as much functionality as possible is included in a single chip, so breaking out detailed powered measurements is difficult.

Stanley-Marbell and Cabezas [49] compare Beagleboard, PowerPC, and x86 low-power systems for thermal and power. Aroca et al. [2] compare Pandaboard, Beagleboard, and various x86 boards and measure FLOPS/W. Jarus et al. [27] compare the power and energy efficiency of Cortex-A8 systems to x86 systems. Laurenzano et al. [31] compare Cortex A9, Cortex A15 and Intel Sandybridge and measure power and performance on a wide variety of HPC benchmarks.

Cloutier et al. [8] investigate power and performance of a wide variety of ARM boards as well as on a cluster built of Raspberry Pi nodes. These nodes are instrumented to measure power, by having a sense resistor across the input power which is amplified by an opamp and measured by an SPI A/D converter.

Molka et al. [40] use a ZES LMG450 power meter sampling at 10Hz to measure the power consumption of individual x86-64 instructions, although they are limited by extracting the CPU power from full-system measurements (they do not monitor the CPU power directly).

Demmel and Gearhart [11] validate two SandyBridge machines against RAPL Package with the STREAM [36] benchmark and a full-system wall power meter.

Hackenberg et al. [20] validate RAPL (and the similar AMD APM interface) on a variety of SandyBridge hardware. They measure both at the wall, as well as the CPU and motherboard level by intercepting the ATX power connectors. They find that RAPL accuracy can vary by workload, and that it can be confused when HyperThreading is enabled.

Mazous, Pradelle and Jalby [35] apply statistical validation to RAPL results compared to full system wall outlet measurements on IvyBridge and SandyBridge. They found some anomalies with the RAPL results when only exercising a single core or when operating at maximum frequency.

Hackenberg et al. [21] investigate RAPL on Haswell-EP processors. They find that the DRAM + Package RAPL results correlate well with total system power readings, but do not measure the individual actual power results for CPU or DRAM.

8 Conclusion

Power measurement is hard, but somebody has to do it.

References

[1] Top green 500 list: Environmentally responsible supercomputing. `http://www.green500.org/`.

[2] AROCA, R., AND GONÇALVES, L. Towards green data centers: A comparison of x86 and ARM architectures power efficiency. *Journal of Parallel and Distributed Computing 72* (2012), 1770–1780.

[3] BAKER, B. Using single supply operational amplifiers in embedded systems. Tech. Rep. AN682, Microchip, 2000.

[4] BEDARD, D., FOWLER, R., LINN, M., AND PORTERFIELD, A. PowerMon 2: Fine-grained, integrated power measurement. Tech. Rep. TR-09-04, Renaissance Computing Institute, 2009.

[5] BIRCHER, W., AND JOHN, L. Complete system power estimation: A trickle-down approach based on performance events. In *Proc. IEEE International Symposium on Performance Analysis of Systems and Software* (Apr. 2007), pp. 158–168.

[6] BURR-BROWN. *INA122: Single Supply, MicroPower Instrumentation Amplifier*, Oct. 1997.

[7] CHEN, H., WANG, S., AND SHI, W. Where does the power go in a computer system: Experimental analysis and implications. In *International Green Computing Conference* (July 2011), pp. 1–6.

[8] CLOUTIER, M., PARADIS, C., AND WEAVER, V. Design and analysis of a 32-bit embedded high-performance cluster optimized for energy and performance. In *First International Workshop on Hardware-Software Co-Design for High Performance Computing* (Nov. 2014).

[9] CUI, Z., ZHU, Y., BAO, Y., AND CHEN, M. A fine-grained component-level power measurement method. In *International Green Computing Conference* (July 2011).

[10] DAVID, H., GORBATOV, E., HANEBUTTE, U., KHANNA, R., AND LE, C. RAPL: Memory power estimation and capping. In *ACM/IEEE International Symposium on Low-Power Electronics and Design* (Aug. 2010), pp. 189–194.

[11] DEMMEL, J., AND GEARHART, A. Instrumenting linear algebra energy consumption via on-chip energy counters. Tech. rep., Electrical Engineering and Computer Sciences, University of California at Berkeley, June 2012.

[12] DIOURI, M., DOLZ, M., GLÜCK, O., LEFÈVRE, L., ALONSO, P., CATALÁN, S., MAYO, R., AND QUINTANA-ORT, E. Solving some mysteries in power monitoring of servers: Take care of your wattmeters! In *Energy Efficiency in Large Scale Distributed Systems conference* (Apr. 2013).

[13] ECONOMOU, D., RIVOIRE, S., KOZYRAKIS, C., AND RANGANATHAN, P. Full-system power analysis and modeling for server environments. In *Proc. Workshop on Modeling, Benchmarking, and Simulation* (June 2006).

[14] ELECTRONIC EDUCATIONAL DEVICES. Watts Up PRO. `http://www.wattsupmeters.com/`, May 2009.

[15] ESMAEILZADEH, H., CAO, T., YANG, X., BLACKBURN, S., AND MCKINLEY, K. Looking back on the language and hardware revolutions: Measured power, performance and scaling. In *Proc. 16th ACM Symposium on Architectural Support for Programming Languages and Operating Systems* (Mar. 2011).

[16] FENG, X., GE, R., AND CAMERON, K. Power and energy profiling of scientific applications on distributed systems. In *Proc. 19th IEEE International Parallel and Distributed Processing Symposium* (Apr. 2005), p. 34.

[17] GE, R., FENG, X., SONG, S., CHANG, H.-C., LI, D., AND CAMERON, K. PowerPack: Energy profiling and analysis of high-performance systems and applications. *IEEE Transactions on Parallel and Distributed Systems 21*, 6 (May 2010).

[18] GOTTSCHO, M., KAGALWALLA, A., AND GUPTA, P. Analyzing power variability of ddr3 dual inline memory modules. Tech. rep., University of California, Los Angeles, 2011.

[19] GOTTSCHO, M., KAGALWALLA, A., AND GUPTA, P. Power variability in contemporary DRAMs. *IEEE Embedded Systems Letters 4*, 2 (2012), 37–40.

[20] HACKENBERG, D., ILSCHE, T., SCHOENE, R., MOLKA, D., SCHMIDT, M., AND NAGEL, W. E. Power measurement techniques on standard compute nodes: A quantitative comparison. In *Proc. IEEE International Symposium on Performance Analysis of Systems and Software* (Apr. 2013).

[21] HACKENBERG, D., SCHÖNE, R., ILSCHE, T., MOLKA, D., SCHUCHART, J., AND GEYER, R. An energy efficiency feature survey of the Intel Haswell processor. In *Proc of the 11th Workshop on High-Performance, Power-Aware Computing* (May 2015).

[22] HAMANN, H., LACEY, J., WEGER, A., AND WAKIL, J. Spatially-resolved imaging of microprocessor power SIMP: hotspots in microprocessors. In *In Proc. 10th Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronics Systems* (May 2006), pp. 121–125.

[23] HENNECKE, M., FRINGS, W., HOMBERG, W., ZITZ, A., KNOBLOCH, M., AND BÖTTIGER, H. Measuring power consumption of IBM Blue Gene/P. *Computer Science – Research and Development 27* (Nov. 2012), 329–336.

[24] HSU, C.-H., AND POOLE, S. Power measurement for high performance computing: State of the art. In *International Green Computing Conference* (July 2011).

[25] HYLICK, A., SOHAN, R., RICE, A., AND JONES, B. An analysis of hard drive energy consumption. In *Proc. IEEE 16th International Symposium on Modelling Analysis and Simulation of Computer and Telecommunication Systems* (Sept. 2008), pp. 1–10.

[26] INTEL. Voltage regulator-down (vrd) 11.0 processor power delivery design guidelines for desktop lga775 socket. http://www.intel.com/content/dam/doc/design-guide/voltage-regulator-down-11-0-processor-power-delivery-guide.pdf, Nov. 2006.

[27] Jarus, M., Varette, S., Oleksiak, A., and Bouvry, P. Performance evaluation and energy efficiency of high-density HPC platforms based on Intel, AMD and ARM processors. *Energy Efficiency in Large Scale Distributed Systems* (2013), 182–200.

[28] Khanna, R., Zuhayri, F., Nachimuthu, M., Le, C., and Kumar, M. Unified extensible firmware interface: An innovative approach to DRAM power control. In *Proc. International Conference on Energy Aware Computing* (Nov. 2011).

[29] Khoshbakht, S., and Dimopoulos, N. Relating application memory activity to processor power. In *Proc. International Conference on Parallel Processing* (Oct. 2013), pp. 849–857.

[30] Laros III, J., Pedretti, K., Kelly, S., Vandyke, J., Vaughan, C., and Swan, M. Investigating real power usage on red storm. In *Cray Users Group Meeting* (2009).

[31] Laurenzano, M., Tiwari, A., Jundt, A., Peraza, J., Ward Jr., W., Campbell, R., and Carrington, L. Characterizing the performance-energy tradeoff of small ARM cores in HPC computation. In *Proc. of Euro-Par 2014* (Aug. 2014), pp. 124–137.

[32] Lee, B., and Brooks, D. Accurate and efficient regression modeling for microarchitectural performance and power prediction. In *Proc. 12th ACM Symposium on Architectural Support for Programming Languages and Operating Systems* (Oct. 2006), pp. 185–194.

[33] Lee, D., O'Sullivan, M., and Walker, C. A framework for measuring the performance and power consumption of storage components under typical workload. *GSTF Journal on Computing 1*, 2 (Feb. 2011).

[34] Mahesri, A., and Vardhan, V. Power consumption breakdown on a modern laptop. In *Proc. of the 4th International Workshop on Power-Aware Computer Systems* (Dec. 2004), pp. 165–180.

[35] Mazouz, A., Pradelle, B., and Jalby, W. Statistical validation methodology of CPU power probes. In *Proc. of 1st International Workshop on Reproducibility in Parallel Computing* (Aug. 2014).

[36] McCalpin, J. STREAM: Sustainable memory bandwidth in high performance computers. `http://www.cs.virginia.edu/stream/`, 1999.

[37] Measurement Computing. *USB-1208FS-Plus Analog and Digital I/O User's Guide*, 2014.

[38] Mesa-Martinez, F., Nayfach-Battilana, J., and Renau, J. Power model validation through thermal measurements. In *Proc. 34th IEEE/ACM International Symposium on Computer Architecture* (June 2007), pp. 302–311.

[39] Microchip. *MCP6041/MCP6042/MCP6043/MCP6044 Data Sheet*.

[40] Molka, D., Hackenberg, D., Schone, R., and Muller, M. Characterizing the energy consumption of data transfers and arithmetic operations on x86-64 processors. In *Proc. International Green Computing Conference* (Aug. 2010), pp. 123–133.

[41] no, M. C., Catalán, S., Mayo, R., and Quintana-Ortí, E. Reducing the cost of power monitoring with DC Wattmeters. *Computer Science – Research and Development 30*, 2 (May 2015), 107–114.

[42] NVIDIA. *NVML Reference Manual*, 2012.

[43] Paradis, C. Detailed low-cost energy and power monitoring of computing systems. Master's thesis, University of Maine, Aug. 2015.

[44] Piga, L., Bergamasci, R., Azevedo, R., and Rigo, S. Power measuring infrastructure for computing systems. Tech. Rep. IC-11-09, Universidade Estadual de Campinas, Mar. 2011.

[45] Poirier, C., McGowen, R., Bostak, C., and Naffziger, S. Power and temperature control on a 90nm Itanium ®-family processor. In *Proc. IEEE International Solid State Circuits Conference* (2005), pp. 304–305.

[46] Rahmati, A., Hicks, M., Holcomb, D., and Fu, K. Refreshing thoughts on DRAM: Power saving vs. data integrity. In *Workshop on Approximate Computing Across the System Stack* (Mar. 2014).

[47] Singh, D., and Kaiser, W. Energy efficient network data transport through adaptive compression using the DEEP platforms. In *IEEE 8th International Conference on Wireless and Mobile Computing* (Oct. 2012), pp. 253–260.

[48] Singh, D., Peterson, P., Reiher, P., and Kaiser, W. The atom LEAP platform for energy-efficient embedded computing: Architecture, operation, and system implementation. Tech. rep., University of California, Los Angeles, Dec. 2010.

[49] Stanley-Marbell, P., and Cabezas, V. Performance, power, and thermal analysis of low-power processors for scale-out systems. In *Proc of IEEE International Symposium on Parallel and Distributed Processing* (May 2011), pp. 863–870.

[50] Wallace, S., Vishwanath, V., Coghlan, S., Lan, Z., and Papka, M. Measuring power consumption on IBM Blue Gene/Q. In *Proc. 9th Workshop on High-Performance Power-Aware Computing* (May 2013), pp. 853–859.

[51] Yan, J., Lonappan, C., Vajid, A., Singh, D., and Kaiser, W. Accurate and low-overhead process-level energy estimation for modern hard disk drives. In *IEEE International Conference on Green Computing and Communications* (Aug. 2013), pp. 171–178.

[52] Yoshii, K., Iskra, K., Gupta, R., Beckman, P., Vishwanath, V., Yu, C., and Coghlan, S. Evaluating power monitoring capabilites on IBM Blue Gene/P and Blue Gene/Q. In *IEEE International Conference on Cluster Computing* (Sept. 2012), pp. 36–44.