ArduPower v2: Open and Modular Power Measurement for HPC Components

Daniel Bremer, Michael Kuhn and Mohammad Reza Heidari
Universität Hamburg
Hamburg, Germany
Email: {daniel.bremer, michael.kuhn, heidari}@informatik.uni-hamburg.de

Abstract—**Accurate power measurement is a prerequisite to energy optimization in High Performance Computing (HPC) systems. Fine temporal and spatial power profiling is also required to capture fast power fluctuations that can point to inefficiencies in applications, libraries or the operating system. Moreover, flexible deployment of powermeters is an important factor to apply power measurement in computing systems without breaking into the system circuitry or compromising the system warranty. However, it is usually a challenge to find a comprehensive solution in the market that meets all the requirements to study energy efficiency in HPC systems. In response to the various research needs, ArduPower aims at a modular internal wattmeter platform. It enables flexible deployment in computing systems, renders accurate power monitoring with a high sampling rate at component level and offers an on-the-fly power consumption analysis for given components. Its open design and low price lend it to an affordable option for fast deployment and further customization for specific needs. Our evaluations show that ArduPower is capable of delivering competitive results with commercial solutions which allows us to identify application phases within popular benchmark programs. This information can be exploited to tune for higher computational performance or energy efficiency in computing systems.**

Keywords–Internal Wattmeter; Power Consumption; Energy Efficiency; ArduPower; Modular Architecture.

I. Introduction

In many fields in research and engineering, High-Performance Computing (HPC) is required to process data or simulate processes or structures. While computation clusters were rated and evaluated by their performance and Floating Point Operations per Second (FLOPS) for a long time, today's evaluation processes also consider operating costs. Besides maintenance, a huge factor for these is energy consumption. This shift is also shown with the introduction of the Green500 list in 2009 which, opposing to the TOP500 list, ranks systems using their energy efficiency by considering watts per GFLOPS. While the TOP500 is dominated by Graphics Processing Unit (GPU) powered systems, the Green500 list places an ARM-based CPU in the top position, which only reaches place 159 in the TOP500 [1]. The ArduPower v2 platform, introduced in this paper, aims to enable cheap, plug and play power measurement for many systems, allowing for optimizations in energy efficiency in regards to hardware selection, as well as programming techniques.

Measurement of power consumption is done with wattmeters. The formula for electric power P is given by $P = U * I$, where U is voltage and I is the current. Normally, the voltage on a power delivery line is known relatively well and should be in a specified range, e.g., the ATX standard specifies $\pm 5\,\%$ for positive lines. Acquiring a value can be easily done by either consulting the standard's documentation or by probing with a voltmeter, which is connected in parallel, or an Analog-Digital-Converter.

The current, on the other hand, fluctuates with the current workload of the consumer circuit and often is an unknown value. To measure its value, most often Hall-effect sensors are used to determine the strength of the magnetic field induced by a current passing a conductor, which directly correlates to the current value. Measuring current can be done with a clamp meter, which is placed around a singular conductor and captures the induced magnetic field strength, or by placing an ammeter in series with the consumer.

Introducing a wattmeter for energy analysis can be done externally or internally. External wattmeters, such as ZES Zimmer LMG450 or similar, monitor power consumption between the power outlet and a server's power supply. Readings represent the power draw of a system as a whole, but only allow an overall estimation of real load. Factors offsetting the values are the efficiency of the power supply or sporadic, non-application related workloads. A detailed analysis often is hard and power fluctuations of single components can not be identified.

Internal wattmeters aim to provide power draw readings on a component level, differentiating between energy used by processors, mainboard, hard disk drives, etc. A finer level of analysis is possible as different application behaviors like computation bound or memory bound show on their corresponding channels and in the specific energy profiles. Results also can be transformed for system-level analysis by summing up all channels. Combining both variants, external and internal, the efficiency of power supply units can be evaluated. With ArduPower v2 an internal wattmeter is introduced. Evolved from the ArduPower platform, it delivers more features while maintaining the already proven concept.

II. Uses of Power Consumption Analysis

Building a computation cluster is a complicated task requiring optimization of multiple features. A naive approach to this can be simply to optimize available computational resources, like FLOPS. This might work for building a capable general purpose cluster, but often systems face specific workloads repeatedly, which might be a reference for tuning. For instance, systems running ligand configuration simulations often benefit from high core counts, as the tasks are worked on in a batch-like manner, therefore parallel and independent simulations are done. Engineering tasks, however, are regularly calculations-heavy and benefit from faster floating point arithmetics, that are found on higher clock-frequency CPU models.

Besides clock speeds and thread counts, there are also architectural differences. Generally, CPU clusters are suited for every computational problem, but, in recent years, development of new programming techniques and hardware architectures

 ISBN: 978-1-61208-788-7

pushed the usage of GPUs and accelerator cards, which provide higher FLOPS than regular CPUs. Also, ARM's technology emerged, providing low-powered CPUs with high energy efficiency. Besides the hardware cost for a new system, energy cost accumulates over the time of usage, reaching a significant portion of the initial costs. HPC systems have a typical lifetime of around 5–6 years and energy costs over this period of time can easily add another 20 % of total procurement costs. These costs include electricity for powering the machine as well as costs for cooling. Due to the high costs of supercomputers, reducing energy costs even by small amounts can result in huge cost savings. In periods of inactivity, servers are typically not shut down but left idling, during which they still consume significant amounts of energy. While external and non-critical devices like GPUs can be turned off completely, CPUs can typically only be put into power-saving modes.

Power consumption analysis allows evaluating components on an energy level. When comparing configurations, a standardized set of programs can be run to model general cluster usage and energy consumption can be tracked. If the results show the efficiency of GPUs over CPUs for the desired use case, such a more specialized cluster could be taken into consideration. If computation speed can be sacrificed for energy efficiency, ARM-based CPUs could offer an alternative over regular x86 CPUs. Evaluating feature changes can be done as well. While GPUs often perform faster floating point arithmetics than CPUs, modern chips provide vectorization with AVX512 and therefore can heavily boost performance, and in some cases even outperform GPUs [2].

In operation, power consumption monitoring can be used to allocate resources to users. Traditional job managers allocate available nodes by considering real-world time slots, CPU time or a type of credit. Instead of providing a 6-hour time slot on a machine, the job manager could implement a power limit that could be set over a certain time span. This would motivate users to use more energy-efficient code, to maximize runtimes of programs, and be mindful when planning jobs. Managers like SLURM [3] support power management natively, which can manage nodes power caps and allows for accounting of consumed energy.

III. Related Work

Reliable power measurement is a prerequisite to energy efficiency as the key optimization goal in HPC systems. Despite considerable efforts, there is a wide spectrum of applications that yet demand more flexible power measurement in HPC systems. Power monitoring solutions are usually dictated by the applications' power profiles, which affect the required sampling rate, measurement accuracy, spatial granularity, scalability and instrumentation. More often than not, it is highly desirable to leverage existing flexibility that lends itself to a variety of applications and customizations [4].

External power measurement is a common practice to study power consumption at node level in computing systems. Power Supply Units (PSUs) and Power Distribution Units (PDUs) often provide the overall power consumption of a computing node via the Intelligent Platform Management Interface (IPMI) with a relatively low precision and sampling rate [5]. However, they fail to capture power traces that last only a few seconds [6]. Professional AC wattmeters, such as ZES Zimmer, offer accurate average power measurement externally at node level but their low sampling rate cannot reflect fast power fluctuations for detailed analysis [7]. The external power measurement approach usually suffers from a low spatial granularity that is an obstacle in the power analysis at component level.

Internal power measurement is, however, a way forward to achieve more flexible power monitoring as it usually offers higher spatial and temporal granularities. PowerInsight, designed by Sandia National Laboratories and Penguin Computing, is an internal wattmeter with 15 channels and provides instantaneous DC power measurement at a maximum total sampling rate of $1{,}000\,\mathrm{Sa/s}$ [8]. It offers a probe-based monitoring connected to a BeagleBone for data collection and forwarding. Its design supports standard power plugs as well as PCI-e risers to monitor devices such as GPUs. PowerMon2 is also an internal wattmeter that provides instantaneous power measurement at a maximum sampling rate of $1{,}024\,\mathrm{Sa/s}$ and supports up to 8 DC channels [9]. PowerPack is another internal power monitoring system that supports dynamic power management configurations [10]. It provides component-based power monitoring through a dedicated data acquisition mechanism with support for multicore processors and power-performance analysis affected by the DVFS mechanism.

New HPC systems embed vendors' proprietary solutions to estimate energy consumption system-wide from node level to the component level. As a prominent example, the High Definition Energy Efficiency Monitoring (HDEEM) is a sophisticated approach, designed by Technische Universität Dresden and Bull S. A. S. [11]. Its goal is a fine-grained power measurement enabled by a sampling rate of $8{,}000\,\mathrm{Sa/s}$ over a fine spatial granularity, e.g., for per-CPU measurements. Cray XC30 system series also offers an integrated power measurement infrastructure at component level that includes blade and GPU measurements at $10\,\mathrm{Sa/s}$ [12]. IBM applies a similar integrated circuits approach to Power7 processors to monitor power consumption at a component level [13].

While the PowerInsight [8] approach is similar to ArduPower in regard to deployment, the feature sets differ vastly. Featuring a BeagleBone board, PowerInsight has potential to directly store captured data on a drive or stream it to another endpoint via Ethernet, but is limited in terms of connectivity, as the internal 7-channel ADC only is referenced to $1.8\,\mathrm{V}$, making it crucial to implement a dedicated ADC for $5\,\mathrm{V}$ references sensors on the "Power Cape" board. ArduPower, on the other hand, is better geared towards customization, featuring an Arduino Mega. This provides 16 native $5\,\mathrm{V}$ referenced ADC channels and more connectivity via GPIO pins, allowing implementation of SPI or serial buses for further devices, not just for power monitoring, but also for arbitrary sensors like thermal probes. Providing a richer interface for sensors, the automatic configuration feature enables plug and play installation, leading to fast deployment.

IV. The ArduPower Platform

The first implementation of ArduPower was presented by Dolz et al. and was based on a custom shield for the Arduino Mega platform, providing 16 inputs for power lines [14]. After a configuration process, the system could be connected to a power wiring harness inside a computation node, by cutting wires and rerouting them to the shield, and data was sent with an 8-bit serial interface to the monitoring node.

While the wattmeter itself worked flawlessly, the system required much manual configuration. Version 2 re-implements the idea of using Hall-effect sensors for current measurement and introduces new features, such as an automatic configuration of the device and an improved serial protocol. To enable fast setup and modularity, the platform was split into a collector unit, consisting of an Arduino Mega 2560 and a shield, which is used for internal line routing and automatic configuration circuitry. Moreover, the Hall-effect sensors were moved onto dedicated probe devices to modularize the approach.

Probes are designed to be female-male extension cables equipped with Hall-effect sensors. This allows integration into the existing wiring harness with no further manipulation. All installed probes then need to be connected with the collector unit by cable to be forwarded to the monitoring node.

A. Hardware

In the following, ArduPower v2's hardware design will be described in detail, which is illustrated in Figure 1.

Probes: Voltages on internal power lines are known to be either 3.3 V, 5 V or 12 V with an error of 5 %, as given by the ATX specification. Components with high power consumption like CPUs or GPUs normally use 12 V for power delivery, while Solid-State Drives (SSDs) and USB devices do not consume as much power and, therefore, mainly rely on lower voltages. The reason to use different voltage levels is to keep currents low. Both voltage and current are a linear factor in electric power, therefore, it is possible to transport 60 W of power with 5 V and 12 A or by using a voltage of 12 V and 5 A of current. The reasons to choose higher voltages are smaller wire diameters, reduced magnetic fields (induced by the Hall-effect) and safety.

As the voltages are known due to standardization, probes need to be put in series with the power supply and power consuming component. For simple connection, a probe is designed similar to an extension cable with a small circuit board interrupting the power carrying line. This circuit then contains an ACS723 Hall-effect sensor by Allegro and connections to provide power to the probe and return data to the collector unit. To allow for better accuracy, the bidirectional, 40 A-rated sensor for high power probes (e.g. PCI-e power to GPUs) and the bidirectional, 20 A variant for lower-powered devices were chosen. Connection pins on the probe are used for 5 V power in and GND for power delivery and for an analog signal from the ACS723 sensor which represents the current, a type pin to determine the used ACS723 variant and a pin carrying the line voltage to the ArduPower collector unit, that is used by the automatic configuration.

Shield: Arduino shields are circuit boards with a layout that can directly be stacked onto a device's pins and allow for hot-pluggable reconfiguration of functionalities. ArduPower's version 1 directly mounted the Hall-effect to this board, while version 2 moved the sensors to the probes and used the shield for internal wire routing. It also features an analog multiplexer chip, a voltage divider and connection pins to interface with the probes.

ArduPower v2 implements an automatic configuration that collects information on the connected probes to determine their type and the voltage of the measured line. As for this, an analog voltage must be digitalized and each voltage has to be sensed by the Arduino's Analog-Digital-Converter (ADC). But, as the

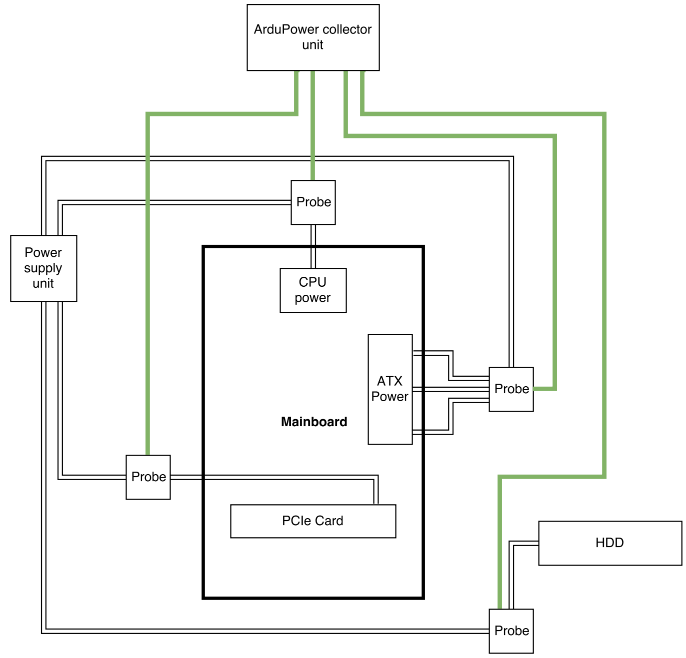

Figure 1. Implementing ArduPower v2 is done by placing probes between the power supply unit and each different consumer components, such as CPU, GPU or storage devices. Each probe then is connected to the ArduPower collector unit, which itself is connected to a monitoring node.

output signal from the probes also is an analog signal, this, too, has to be sent to the ADC, which maxes out the Arduino's capabilities, as the ADC only provides 16 inputs. An external multiplexer was built onto the shield to allow for input of 32 analog signals, enabling sensing all probed lines' voltages and the output values of the sensors.

Another crucial step identifying the line voltages on the probes was the introduction of a voltage divider. The Arduino Mega has a rated pin voltage of 5 V, while computer power lines carry up to 12 V and therefore would destroy the device. To circumvent this, a voltage divider was set up, remapping the 0 V to 12 V range to a 0 V to 5 V range, which can be read by the ADC.

Automatic Configuration: Previous versions of ArduPower required a configuration file for voltage information like probe voltage. Version 2 implements an automatic configuration feature. By forwarding the voltage of the probed line or another set voltage between 5 V and 12 V from the probe to the collector unit, the voltage can be automatically identified and used for power calculation. The user only must identify the measured line, e.g., whether a 12 V line is connected to an EPS connector, providing power to a CPU or a PCI device, like a GPU. This feature requires a combination of hardware and software. While a voltage divider and an additional analog multiplexer were introduced on the ArduPower shield, the control of the multiplexer is realized with software. Voltage measurement is handled by the Arduino's ADC with the `analogread` method, while voltage identification is done in software, as the voltage divider shifts the measured voltages into a 0 V to 5 V range.

Further optimization is done by only considering connected probes when performing a measurement step. Analog-digital-conversion is a relatively slow operation, therefore, only a rate of 9,600 Hz can be achieved, as each ADC cycle takes about 110 μs. With each start of a measurement, connected probes are stored and only the corresponding ports are then covered in the measurement cycle.

Deployment: Deploying the ArduPower wattmeter is a process of installing the probes between the components that should be power-traced and the PSU. For this, standard power headers such as ATX, PCI-e or EPS and a probe featuring the corresponding connector are required. Physically, no modifications have to be done other than adding the probes to the existing wire harness and routing wires from the probes to the collector unit and from this unit to an available USB port.

Server style mainboards are often fed with only a 12 V line providing most of the power, which then is transformed to the respective operating voltage of the component. Such boards are currently not supported due to the limitation of space in such chassis.
Larger scale deployments can be realized by installing probes into multiple nodes and feeding the probe's connections to either a dedicated ArduPower collector unit per node (similar to 1) or by connecting probes of multiple nodes to a single collector unit. Traces then can be collected by the profiled node itself or by a dedicated node that all collector units are connected to.

B. ArduPower v2 Firmware

Microcontrollers like the Atmel ATmega2560, which is the processor used on the Arduino Mega platform, not only can interact with hardware like Hall-effect sensors, but also can be flashed with a custom firmware that enables processing of data. While ArduPower provides an Arduino-compatible shield and probes to plug into a computer's wiring harness, a complementary firmware is required to control the serial communication with a monitoring node, optimize data for efficient transport and enable features like automatic configuration.

Serial Protocol: A standard serial interface uses a frame length of 8 bits for data. The result of the analog-digital-conversion has a precision of 10-bits. For better efficiency in serial communication, the channel-combination-protocol was developed for ArduPower v1. This protocol features a synchronization bit and splits data of two probes to fit into three serial frames. This protocol leaves one bit unused in every three frames and cannot fill 4 bits of a frame for every uneven number of probes, leaving

$$\lfloor n/2 \rfloor + (n \mod 2) * 4$$

bits unused per n probes. By altering the serial connection to use 6 data bits, a 10 bit result can be split into two 5 bit frames and therefore each probe requires two serial frames. This change uses 100 % of all available data bits while having a 4 bit shorter overall length if the number of probes is odd.

C. Collector Program

While the hardware collects data on power consumption, calculation of the actual current, storing of values and analysis are handled externally by either the analyzed node itself or by a dedicated monitoring computer. To retrieve data, a collector program is provided, which handles receiving data from the ArduPower unit over a serial connection and deciphering the protocol. Also, the automatic configuration results are stored and used to determine power consumption.

All measurement values sent by the ArduPower unit are raw 10 bit ADC values, representing the output voltage of the ACS723 sensors. This value first has to be used to derive the actual sensor's output voltage, which then must be mapped to a current. This is done by solving

$$I = \frac{\text{type}}{2} * \left(\frac{\text{analogReadVal} - 5\,\text{V}}{1{,}024\,\text{V}} - 2.5 \right)$$

where `type` is the sensor's range (e.g., 20 A or 40 A) and `analogReadVal` is received from the ArduPower unit. Furthermore, the program enables starting and stopping of read cycles on the ArduPower unit, can request the current sensor configuration and handles synchronization errors.

V. Evaluation

Evaluation is done by comparing the results of our approach to similar works. Song et al. profiled the HPC Challenge benchmark set and can be used for cross-comparison of power draw behavior [15]. Comparing against the now deprecated ArduPower v1 platform shows improvements of the new device and allows to check the accuracy of the gained results.

For testing, a Nehalem based Intel machine was used. The system was equipped with two Xeon X5560 CPUs with a rated Thermal Design Power (TDP) of 95 W each. The TDP provides the maximum heat output of a chip in reference to the base clock frequency, therefore it is not the maximum electrical power draw of a chip, which can be higher. Power draw increases quadratically with frequency, therefore resulting in a much higher power draw than in the product specification when turbo boosting.

Data was collected from ArduPower v2 using a standalone collector script. However, a custom collector for the Diamond daemon is also available in the paper repository at [16]. It also includes ArduPower v2's firmware, schematics and more.

When interpreting the following results, a preliminary remark must be kept in mind, namely, the expected behavior in a dual socket system would be nearly identical power draw on both CPUs. As the boards feature two EPS connectors for power delivery to the CPU, the wattage on both should be the same. All experiments show an offset between both EPS lines, which leads to the assumption that power delivery to the CPUs is asymmetrical on both connectors. The difference between both inputs is a constant of 20 W, therefore it is assumed that this is due to the mainboard design.

A. High-Performance LINPACK

Aiming to rate the peak performance R_{peak}, the High-Performance LINPACK Benchmark (HPL) was developed. The benchmark in its core is a solver for dense linear equations leveraging LU factorization with partial pivoting, which then results in a number of 64-bit floating point operations that can be performed per second (FLOPS). The most common implementation of the LINPACK Benchmark was developed by Petitet et. al. and leverages MPI for scalability [17].

The LINPACK benchmark works in a loop with calculation and communication phases. While a computation phase applies load to the CPU, in the communication phase, performance data is shared over MPI, resulting in waiting times in different processes, due to non-parallel execution times.

In Figure 2, these phases can be seen as local minima. The total CPU power consumption peaks at close to 280 W. This

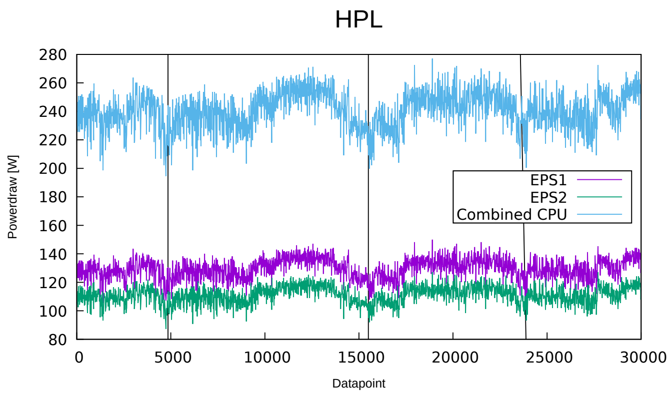

Figure 2. Excerpt of a HPL run measurement. The shown run time is 27.6 s with a total of 30,000 samples. Vertical lines mark the communication sections. Only every 15th sample is plotted for better visibility.

is much higher than the expected TDP of 190 W, but backed by the measurements of ArduPower v1, which also detects wattage peaks. The excess of 90 W therefore can be explained by the aforementioned Turbo Boost features, which overclock the CPU by 0.4 GHz over the base clock. Over the course of the run no thermal throttling could be observed, as this would be recognizable by valleys in the graph. In the future, information about application phases could be used to improve application performance, as well as energy efficiency. For instance, during computation phases, the network is typically idle and could be used for asynchronous I/O or other tasks.

B. MPI Fast Fourier Transform

Similar to the HPL, the MPI Fast Fourier Transform (FFT) benchmark is also contained in the HPC Challenge benchmark suite. It implements the fast Fourier transform operation over all MPI processes and can be used to identify scaling issues and MPI/communication-link limitations.

The MPI FFT's power draw graph is shown in Figure 3. The general course of power draw is a spike, which is a short computation or generation phase, followed by a plateau that represents MPI activity in communicating data. The measurements were done on a single node, which impacts the timings of these phases. As they are tightly linked to a single system's load, huge variation can be expected. The wattage in these phases is lower because of synchronization on MPI processes and memory copy operations that normally are not handled by the CPU. Again, information about computation and communication phases could help to use the system more efficiently. For instance, Turbo Boost could be selectively switched on for the short computation spikes.

C. High Performance Conjugate Gradients

While benchmarks such as the HPL test the performance of computation systems by producing huge arbitrary loads, the High Performance Conjugate Gradients benchmark proposed by Dongarra et. al. targets to model data access patterns as found in the real world by leveraging sparse matrix calculations and local Gauss-Seidel smoothing, besides other commonly used engineering methods [18]. The HPCG benchmark is known to consume much less power than the HPL, while maintaining a more stable draw [19]. In contrast to the HPL, this benchmark is memory bound, without the aim to calculate peak system performance. The results in Figure 4 are again confirmed with ArduPower v1, cross-validating the measurement.

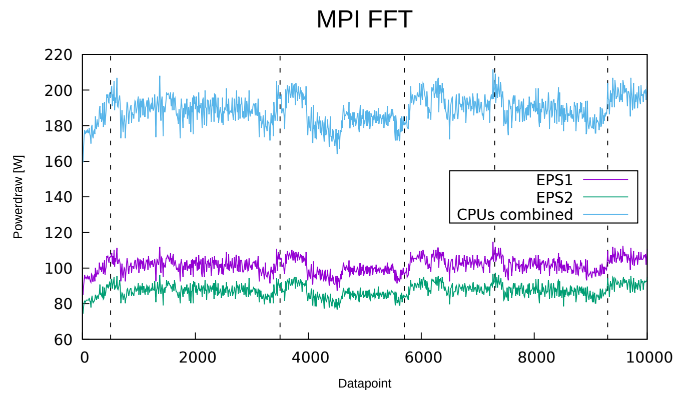

Figure 3. Power consumption of the MPI FFT test in the HPCC benchmark. Vertical lines signal peaks, which indicate computation/generation phases, followed by communication activity. The data is reduced to show every 10th sample.

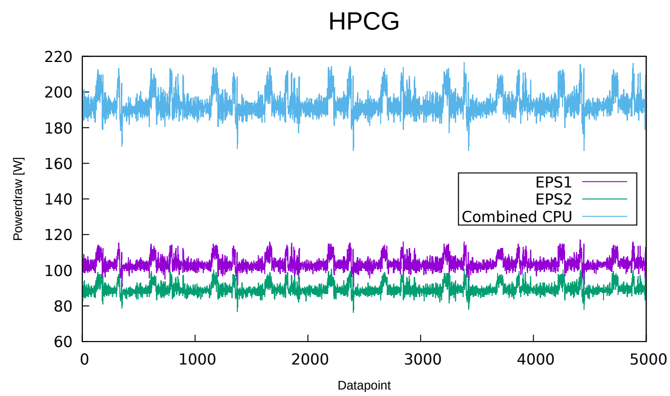

Figure 4. Excerpt of a HPCG power draw analysis. The shown duration is 4.6 s with a total of 5,000 samples. This plot shows every captured data sample.

D. ArduPower v2 Performance

Redesigning the ArduPower platform was done with the goals of increased usability, higher modularity for any given use case and increased performance. To increase throughput of captured data, a new protocol based on 6-bit serial communication was implemented. The sampling speed is mostly determined by the analog-digital-conversion. The new serial protocol on the other hand only increases performance marginally, as the effect only is noticeable with odd probe counts. The achievable sample rate with a varying number of connected probes is shown in Figure 5.

VI. Conclusion

ArduPower v2 is a modular and open design for an internal wattmeter that allows power measurements on a component level. We make use of an Arduino Mega 2560, as well as custom-built shield and software to enable high-frequency measurements using probes. In contrast to coarse-grained external solutions, this allows us to capture and correlate the power impact of individual application phases. This information can then be used to improve the energy efficiency of the overall software stack. ArduPower v2 is an evolutionary improvement of our previous design that delivers verified results while providing a richer feature set than its predecessor.

Since we follow an open approach, we make all necessary schematics, firmware, scripts, etc. available in a repository at

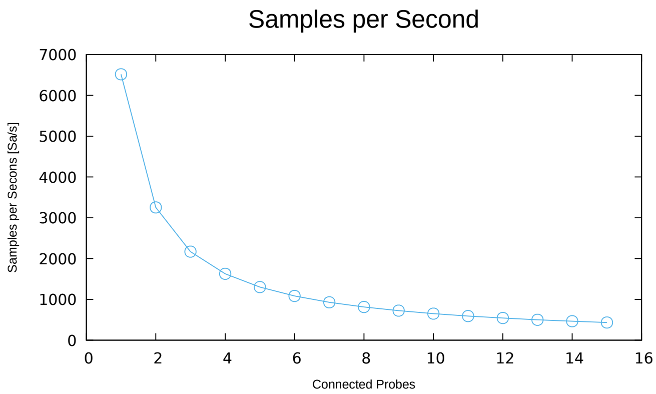

Figure 5. ArduPower v2 sample collection performance. The graph shows the achieved samples per second for each probe configuration. The rapid falloff is due to rising analog-digital-conversion impact on the cycle duration.

[16]. Interested parties can use the provided information to build their own version of ArduPower and use it for fine-grained power measurements. We welcome collaboration to improve both the hardware architecture itself, as well as its integration into the software stack. ArduPower v2 should be usable for computer clusters of any size and can be integrated without having to modify existing hardware. It can therefore provide a convenient building block for research on performance analysis and energy efficiency across many different fields.

VII. Future Work

Being built around modularity, the ArduPower platform can be extended to provide even more insights. In its current state, the focus is on internal power measurement, but a probe for mains electricity power consumption can be implemented to enable efficiency rating of power supply units and easier overall power consumption figures. Currently, there is no possibility to capture power draw from the mainboard to a component. Sophisticated adapters for memory slots, PCI-e or even M.2/U.2 slots, can be made to break down power consumption in even more detail. The open, modular design can also be used for analysis of different performance factors. Temperature probes tracking thermals and cooling capacity, anemometers for airflow analysis or even security mechanisms like intrusion detection can be developed and connected with via the simple five-pin interface. Besides analog values, as shown in the project, also digital communication, such as UART/USART or One-Wire, can be realized. Firmware modification could even allow SPI by providing a clock signal on the voltage return line.

Moreover, ArduPower could be used to provide additional functionality within existing software, such as SLURM. Due to its open design and modest costs, even clusters not equipped with vendor-specific solutions could be upgraded using ArduPower to enable fine-grained power measurements. In contrast to the first version of our approach, ArduPower v2 does not require cutting cables and can thus be integrated and removed easily. While ArduPower v2 still must be embedded into the actual system, its small form factor fits even into relatively small cases.

Due to the high resolution of ArduPower v2's captured measurements, it becomes possible to observe even small variations in power consumption, such as during waiting phases making use of spinlocks. Moreover, the data would make it possible to automatically determine even short application phases. This information could then be used to tune the underlying hardware appropriately.

References

[1] "Green 500 November 2019," Nov 2019, retrieved: April, 2020. [Online]. Available: https://www.top500.org/green500/lists/2019/11/

[2] D. N. S. S. Liyanage, G. V. M. P. A. Fernando, D. D. M. M. Arachchi, R. D. D. T. Karunathilaka, and A. S. Perera, "Utilizing Intel Advanced Vector Extensions for Monte Carlo Simulation based Value at Risk Computation," in International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland, ser. Procedia Computer Science, vol. 108. Elsevier, 2017, pp. 626–634. [Online]. Available: https://doi.org/10.1016/j.procs.2017.05.156

[3] "Slurm Workload Manager." [Online]. Available: https://slurm.schedmd.com/

[4] Diouri et al., "Assessing Power Monitoring Approaches for Energy and Power Analysis of Computers," Sustainable Computing: Informatics and Systems, vol. 4, no. 2, 2014, pp. 68–82.

[5] Intel, Hewlett-Packard, NEC, and Dell. IPMI Specification V2.0 Rev. 1.1. Retrieved: April, 2020. [Online]. Available: http://www.intel.com/content/www/us/en/servers/ipmi/ipmi-second-gen-interface-spec-v2-rev1-1.html (2013)

[6] D. Watts, P. Ainsworth, S. Aithal, B. Choi, and M. Rodriguez, "IBM Systems Director Active Energy Manager Installation and User's Guide," Oct 2019.

[7] Z. Electronic Systems, "4 Channel Power Meter LMG450," User Manual, 2009.

[8] J. H. Laros, P. Pokorny, and D. DeBonis, "PowerInsight - A commodity power measurement capability," in 2013 International Green Computing Conference Proceedings, June 2013, pp. 1–6.

[9] D. Bedard, M. Y. Lim, R. Fowler, and A. Porterfield, "PowerMon: Fine-grained and Integrated Power Monitoring for Commodity Computer Systems," in Proceedings of the IEEE SoutheastCon 2010 (SoutheastCon). IEEE, 2010, pp. 479–484.

[10] R. Ge et al., "PowerPack: Energy Profiling and Analysis of High-Performance Systems and Applications," IEEE Transactions on Parallel and Distributed Systems, vol. 21, no. 5, May 2010.

[11] D. Hackenberg et al., "HDEEM: High Definition Energy Efficiency Monitoring," in 2014 Energy Efficient Supercomputing Workshop, Nov 2014, pp. 1–10.

[12] S. J. Martin and M. Kappel, "Cray XC30 Power Monitoring and Management," in Cray User Group Conference Proceedings, 2014.

[13] M. Knobloch, M. Foszczynski, W. Homberg, D. Pleiter, and H. Böttiger, "Mapping fine-grained power measurements to HPC application runtime characteristics on IBM POWER7," Computer Science - Research and Development, vol. 29, no. 3, Aug 2014, pp. 211–219. [Online]. Available: https://doi.org/10.1007/s00450-013-0245-5

[14] M. F. Dolz, M. R. Heidari, M. Kuhn, T. Ludwig, and G. Fabregat, "ArduPower: A Low-Cost Wattmeter to Improve Energy Efficiency of HPC Applications," in Sixth International Green and Sustainable Computing Conference, IGSC 2015, Las Vegas, NV, USA, December 14-16, 2015. IEEE Computer Society, 2015, pp. 1–8. [Online]. Available: https://doi.org/10.1109/IGCC.2015.7393692

[15] S. Song, R. Ge, X. Feng, and K. W. Cameron, "Energy Profiling and Analysis of the HPC Challenge Benchmarks," Int. J. High Perform. Comput. Appl., vol. 23, no. 3, August 2009, pp. 265–276. [Online]. Available: http://dx.doi.org/10.1177/1094342009106193

[16] "ArduPower Github Repository." [Online]. Available: https://github.com/wr-hamburg/energy2020-ardupower

[17] A. Petitet, R. Whaley, J. Dongarra, and A. Cleary, "HPL – a Portable Implementation of the High-Performance Linpack Benchmark for Distributed-Memory Computers," 01 2008.

[18] M. A. Heroux and J. Dongarra, "Toward a New Metric for Ranking High Performance Computing Systems," 6 2013.

[19] "HPCG Performance Efficiency on VE at 5.99%," Nov 2019, retrieved: April, 2020. [Online]. Available: https://sx-aurora.github.io/posts/hpcg-tuning/