CPU Power Bench: An Automated Benchmark Tool for Power Estimation in Single-Board Computers - Extended Abstract

Houssam Kanso[1,*], Adel Noureddine[1] and Ernesto Exposito[1]

[1]Universite de Pau et des Pays de l'Adour, E2S UPPA, LIUPPA, Anglet, France

Abstract

CPU Power Bench is an automated benchmark tool designed to accurately generate power estimation models for various Single-Board Computers (SBCs) such as Raspberry Pi, Asus TinkerBoard, etc. This tool stresses the CPU with variable theoretical loads, then collects data on CPU cycles and measures the power consumed by the device for each load. CPU Power Bench automatically applies linear and polynomial regression algorithms on the collected data to generate power models, as we have observed a correlation between CPU utilization and power consumption in SBCs.

Keywords

Power Consumption, Measurement, Power Estimation, Automated, Single-Board Computers (SBCs), Tool

1. Introduction

Nowadays, single-board computers (SBCs) are widely used in various embedded systems and Internet of Things (IoT) applications due to their small size, low cost, and low power consumption. This type of device may be powered by batteries or other portable power sources. Power consumption is a critical factor in defining the overall performance and energy efficiency of these devices. Thus, measuring or estimating the power consumption of these devices is a crucial first step toward optimizing their energy usage.

Some state-of-the-art approaches involve the use of hardware meters (such as in [1, 2]). These are physical devices that can be attached to a computing device to measure its power consumption. Hardware meters can provide accurate power measurements. However, hardware-based approaches may require costly specialized equipment.

Other solutions are software-based, they are more accessible, less costly, and more scalable (such as in [3, 4]). However, they may not always be as accurate as hardware meters and can be affected by various factors, such as the device's workload or environmental conditions. As

In: B. Combemale, G. Mussbacher, S. Betz, A. Friday, I. Hadar, J. Sallou, I. Groher, H. Muccini, O. Le Meur, C. Herglotz, E. Eriksson, B. Penzenstadler, AK. Peters, C. C. Venters. Joint Proceedings of ICT4S 2023 Doctoral Symposium, Demonstrations & Posters Track and Workshops. Co-located with ICT4S 2023. Rennes, France, June 05-09, 2023.

*Corresponding author.

houssam.kanso@univ-pau.fr (H. Kanso); adel.noureddine@univ-pau.fr (A. Noureddine); ernesto.exposito@univ-pau.fr (E. Exposito)

https://www.noureddine.org/ (A. Noureddine); https://eexposit.perso.univ-pau.fr/ (E. Exposito)

0000-0002-4811-191X (H. Kanso); 0000-0002-8585-574X (A. Noureddine); 0000-0002-3543-2909 (E. Exposito)

a result, researchers have been working on improving the accuracy of software-based power estimation techniques by incorporating machine learning algorithms.

In this tool demo paper, we introduce CPU Power Bench an automated power consumption model generation tool. In Section 2, we present the energy and metrics benchmarking architecture. In Section 3, we present the relevance of our tool and its use in understanding the energy consumption of each device.

2. CPU Power Bench

CPU Power Bench is an automated benchmark to accurately generate a power model for single-board computers (Raspberry Pi, Asus TinkerBoard, etc.). Our tool's primary objective is to generate power estimation models for any SBC in a simple and automated manner. The tool runs three sets of scripts that stress the CPU with loads ranging from 0 to 100% and collects real-time CPU cycles. Simultaneously, it collects the power consumed by the device from the wattmeter. Then, it uses the theoretical and practical collected data to generate linear and polynomial power estimation models.

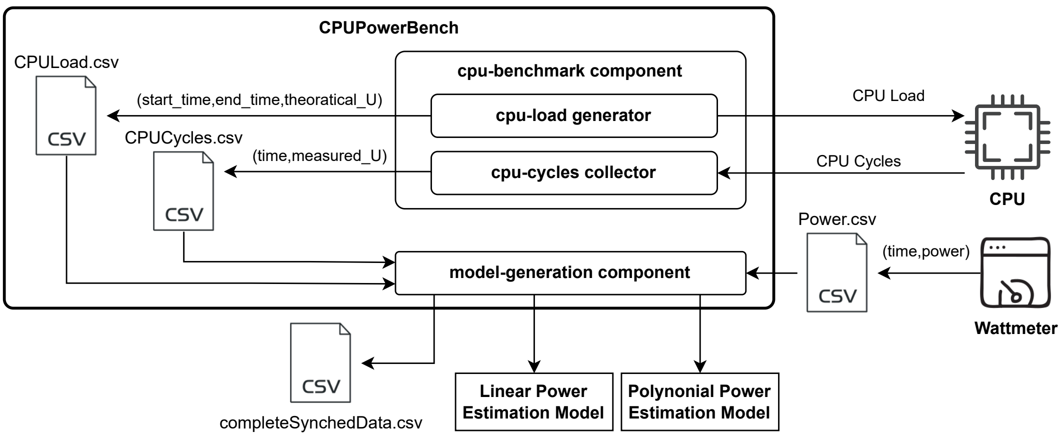

Figure 1: A general overview of CPU Power Bench

Concretely, our architecture, in Figure 1, is composed of three components: a cpu-load generator, a cpu-cycle collector, and a model-generation component. In the following, we detail each of these components.

- *cpu-benchmark component:*
  - *cpu-load generator:* The benchmark produces a workload by applying various loads on the processor of the device. Our strategy is to apply a stress load on the CPU, spanning the entire load spectrum, starting from 0% to 100% with an increase of 5% in each step. The stress is applied for 60 seconds at each percentage interval, and we save the workload timestamp, storing all the data in a CSV file (CPULoad.csv).

We initially used the same stress command (or stress-ng) used in the literature [5] to specify a percentage CPU load. However, we noticed that the CPU load was inconsistent, with the actual CPU load altering between 0% or 100% in various time duration, rather than consistently stabilizing at the asked percentage load. Instead, we used a Python script, CPULoadGenerator [6], which consistently stressed the CPU at the asked percentage with a small degree of variation.

  – *cpu-cycles generator:* The data collector component is a program that collects CPU metrics. In particular, we collect CPU cycles from the Linux *proc* interface (`/proc/stat`). Then, we calculate the CPU utilization (ranging from 0 to 1) and save this data to another CSV file (CPUCycles.csv).

- *model-generation component:* Takes as input the CPUCycles.csv, CPULoad.csv, and Power.csv (data collected by the wattmeter) to synchronize the data into the completeSynchedData.csv. Then, it uses the same data to generate power estimation models (Linear and polynomial) using the scikit-learn machine learning library. It also calculates the best polynomial degree (ranging from 1 to 9) before generating the power estimation model based on the polynomial function.

CPU Power Bench was tested to generate power estimation models for a series of Raspberry Pis (Zero W, 1B, 1B+, 2B, 3B, 3B+, 4B, and 4B+), in addition to ASUS Tinker Board. It provided highly accurate power estimation models for each of these devices with error rates as low as 0.33% and up to 7.81% for linear models, and 0.3% up to 3.83% for polynomial models. CPU Power Bench has much lower error rates when compared to state-of-the-art approaches: 14.56% for PowerPi [3] and 40.76% for EMM [5].

3. Relevance and novelty

CPU Power Bench is a tool designed to generate accurate power estimation models in an automated way. It integrates all the required components into a single easy-to-use tool. The tool can be used to simulate a wide range of workloads on the device's processor, allowing researchers to analyze power consumption under different operating conditions. By stress testing the device, users can assess the power consumption of the device under heavy loads and determine the maximum power consumption that the device can handle. One of the key features of CPU Power Bench is its ability to implement power estimation models on a large scale of SBC devices, even if they are not connected to the internet. This feature is particularly useful for researchers and developers to test and analyze the power consumption of a variety of devices. Another advantage of CPU Power Bench is its architecture, which ensures compatibility with numerous devices.

Acknowledgments

The project leading to this publication has received funding from Excellence Initiative of Université de Pau et des Pays de l'Adour - I-Site E2S UPPA, a French "Investissements d'Avenir" programme.

Online Resources

The source code of the CPU Power Bench is available via CPU Power Benchmark GitHub Project (github.com/joular/cpupowerbench).

References

[1] F. Astudillo-Salinas, D. Barrera-Salamea, A. Vazquez-Rodas, L. Solano-Quinde, Minimizing the power consumption in Raspberry Pi to use as a remote WSN gateway, in: 2016 8th IEEE Latin-American Conference on Communications (LATINCOM), IEEE, Medellin, Colombia, 2016, pp. 1–5. doi:10.1109/LATINCOM.2016.7811590.

[2] ALCIOM, Powerspy2 - experts systèmes électroniques [online], Apr. 06, 2023. URL: https://www.alciom.com/en/our-trades/products/powerspy2/.

[3] F. Kaup, P. Gottschling, D. Hausheer, PowerPi: Measuring and modeling the power consumption of the Raspberry Pi, in: 39th Annual IEEE Conference on Local Computer Networks, IEEE, Edmonton, AB, 2014, pp. 236–243. doi:10.1109/LCN.2014.6925777.

[4] P. M. Sánchez Sánchez, J. M. Jorquera Valero, A. Huertas Celdrán, G. Bovet, M. Gil Pérez, G. Martínez Pérez, LwHBench: A low-level hardware component benchmark and dataset for Single Board Computers, Internet of Things (2023) 100764. URL: https://linkinghub.elsevier.com/retrieve/pii/S2542660523000872. doi:10.1016/j.iot.2023.100764.

[5] K. Kesrouani, H. Kanso, A. Noureddine, A Preliminary Study of the Energy Impact of Software in Raspberry Pi devices, in: 29th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises, Bayonne, France, 2020, pp. 231–234. doi:10.1109/WETICE49692.2020.00052.

[6] G. Carlucci, CPU Load Generator , 2017. URL: https://github.com/GaetanoCarlucci/CPULoadGenerator.