International Journal of Emerging Trends in Computer Science and Information Technology
ISSN: 3050-9246 | https://doi.org/10.63282/3050-9246.IJETCSIT-V5I4P121
Eureka Vision Publication | Volume 5, Issue 4, 187-191, 2024

Original Article

Hardware-in-the-Loop Power Profiling Automation for Consumer Streaming Devices: A Multi-Lab Framework for Regulatory Compliance Validation

Rajasekhar, Sunkara
Independent Researcher, USA.

Abstract - Consumer streaming devices must comply with a growing set of regional energy regulations. In the United States the relevant rules come from the Department of Energy and the California Energy Commission. In the European Union the relevant rule set is the Ecodesign framework for off mode, standby mode, and networked standby. Canada, India, and Japan have their own national programs that reference the same underlying IEC measurement methodology. Each regulation prescribes a set of operating states to measure, and each release of a device requires that the measurements be repeated. Doing this work manually does not scale with the number of devices in the portfolio and the cadence of releases. This paper describes a hardware-in-the-loop power profiling automation framework deployed across two geographically distributed lab locations, onboarding thirty-three streaming stick devices for unattended twenty-four-hour, seven-day-a-week power profiling. The framework integrates programmable AC power meters, programmable USB switching, and infrared remote simulation through a Python orchestration layer. It drives the device under test through the operating states required by each regulation, captures the measurements, and produces the records used in compliance submissions. Empirical results from production deployment show a sixty percent reduction in new product profiling time, from approximately twenty-eight hours to between twelve and fourteen hours per device, a fifty percent reduction in sustenance profiling cycles, and seventy-two percent test case automation across a thirty-nine-case regulatory compliance suite. The paper details the framework architecture, the calibration discipline, and the lessons learned from cross-lab deployment, and provides a reference design for test engineering teams facing similar multi-jurisdictional compliance demands.

Keywords - Power Profiling, Energy Compliance, Hardware-In-The-Loop, HIL, Test Automation, Python, Streaming Devices, Fire TV, US DOE, CEC, EU Erp, Ecodesign, Nrcan, BEE, IEC 62301, Multi-Lab Deployment.

1. Introduction

Each new release of a consumer streaming device is subject to energy compliance measurement under a portfolio of regional regulations. The portfolio currently includes the US Department of Energy and California Energy Commission requirements, the European Union Ecodesign framework that succeeded Commission Regulation (EC) No 1275/2008 and was replaced by Regulation (EU) 2023/826, Natural Resources Canada requirements, the Bureau of Energy Efficiency program in India, and Japanese energy standards. The regulations prescribe operating states such as active power, active standby, and passive standby, and they reference IEC 62301 for the underlying measurement methodology and IEC 62087 for on-mode methodology where relevant.

Performing these measurements manually does not scale. A typical streaming stick has multiple operating states to measure, multiple firmware variants per release, and multiple regulatory jurisdictions whose requirements differ in detail. A single engineer working through the matrix manually for a single device can spend close to a working week per release. With multiple devices in the portfolio and multiple releases per year per device, the manual approach pushes against the available engineering capacity and concentrates schedule risk in a small number of senior engineers.

This paper describes a hardware-in-the-loop power profiling automation framework that was built to address this scaling problem. The framework was deployed across two geographically distributed lab locations and onboarded thirty-three streaming stick devices for unattended operation. It drives the device under test through the regulatory states using programmable instrumentation, captures the measurements, and produces the records used in compliance submissions. The framework is structured so that the same Python orchestration code drives all benches, regardless of the device being measured or the lab the bench is located in.

The rest of the paper is organized as follows. Section 2 covers the prior manual process and its limitations. Section 3 describes the framework architecture. Section 4 describes the cross-lab deployment. Section 5 covers the measurement protocols implemented in the framework. Section 6 presents the empirical results from production deployment. Section 7 discusses lessons learned. Section 8 concludes.

2. The Prior Manual Process

Before the framework was built, power profiling on streaming sticks was performed manually. An engineer would set up the device on a bench, drive it into the relevant operating state using the physical remote, read the power meter, and record the value in a spreadsheet. For state transitions that required disconnecting a USB peripheral, the engineer would physically unplug the peripheral and reconnect it on the appropriate cue. For states that required waiting for the device to settle into the state, the engineer would wait. The process was reliable when performed carefully, but it was slow, and the cost of a single mistake was redoing the affected state.

The cycle time of a new product profiling pass was approximately twenty-eight hours per device. The cycle time of a sustenance pass on an already-onboarded device was shorter but still substantial. The compliance suite included thirty-nine distinct test cases derived from the union of the relevant regulatory requirements. The manual process completed all of these, but it did so by occupying engineering time that could not be used for other work, and it required serial scheduling because the device count per engineer was limited by the engineer's available attention.

3. Framework Architecture

The framework is built around the idea that the bench is the unit of automation. Each bench is a self-contained physical setup that can run unattended once it has been configured. The framework provides the orchestration code that drives a bench, and the database and reporting layer that aggregates results across benches.

3.1. The Bench

Each bench includes a power meter, a programmable USB switch, an IR emitter, the device under test, and the host computer that runs the orchestration code. The power meter is selected to provide zero point zero one watt resolution, which is the resolution required by IEC 62301 for sub-watt measurements. The selection criteria and cost profile of this bench are described in detail in companion work; briefly, the bench is built around commodity instrumentation that costs approximately six hundred US dollars per bench, in contrast to reference benches that cost five thousand to eight thousand US dollars.

3.2. The Orchestration Layer

The orchestration layer is written in Python and runs on the host computer attached to each bench. It exposes drivers for each piece of instrumentation on the bench and for the device under test. The drivers normalize the heterogeneous control surfaces of the underlying hardware into a small set of operations that the higher-level test code uses, such as set the power state, read the steady-state power, connect or disconnect a USB peripheral, and send a sequence of remote button presses. The higher-level test code is written against this abstraction, not against the specific hardware, which is what allows the same test to run on any bench.

3.3. Test Case Library

Each regulatory measurement state is represented as a test case in the framework. The test case carries the sequence of operations needed to bring the device into the state, the criteria for confirming that the state has been reached, the measurement protocol to apply, and the form in which the measurement is recorded. The thirty-nine test cases in the regulatory compliance suite are individual entries in this library.

3.4. Results Storage and Reporting

Results from each run are stored in a central database. The database is the source of truth for compliance submissions. Reports are generated against the database for each release and contain the per-state measurements for each device in the release. The reports are structured to align with the documentary requirements of each jurisdiction so that the same database supports submissions to the US, EU, Canada, India, and Japan.

3.5. Scheduling and Unattended Operation

The framework runs tests according to a schedule rather than on demand. Each bench polls the schedule, picks up the next test case it is responsible for, runs it, and writes the result. Unattended operation is possible because every step of the test, including state transitions and peripheral changes, is driven by programmable instrumentation rather than by a human.

4. Cross-Lab Deployment

The framework is deployed across two geographically distributed lab locations. Thirty-three streaming stick devices were onboarded across the two labs.

4.1. Why Two Labs

Cross-lab deployment serves two purposes. The first is capacity. Concentrating all benches in a single location creates an upper bound on capacity determined by the size of the room and the density of benches that can be installed. The second is redundancy. A single lab is a single point of failure for the compliance workflow. With two labs running the same framework against the same database, the loss of one lab is recoverable rather than blocking.

4.2. Uniformity of the Bench Across Labs

The framework requires that the bench in each lab be a faithful reproduction of the bench in the other. The bill of materials is the same. The driver code in the orchestration layer is the same. The reference instrument used for calibration in each lab is from the same model family. This uniformity is what allows results captured in one lab to be combined with results captured in the other lab without a per-lab correction factor. Cross-lab uniformity was an explicit deployment goal, not an emergent property.

4.3. Onboarding a New Bench

Onboarding a new bench follows a procedure that takes the bench from delivery to first compliance run in a bounded amount of time. The procedure includes the hardware setup, the installation of the orchestration code on the host computer, the validation of the bench against the local reference instrument, and the registration of the bench with the central scheduling and results database. The procedure is documented to the extent that an engineer who has not previously onboarded a bench can complete the onboarding without supervision.

5. Measurement Protocols

This section describes the measurement protocols the framework implements. The protocols are derived from the relevant regulations and from the IEC standards those regulations reference.

5.1. Active Power

Active power is measured while the device is performing its main function. The framework drives the device into the active state using a defined sequence of remote button presses, waits for the device to settle, and records the steady-state power. The settling criterion is the stability criterion prescribed by IEC 62301, applied to the average power over the last third of the measurement window. The measurement window length is also taken from the relevant regulation.

5.2. Active Standby

Active standby is the state in which the device is not performing its main function but can be brought back to active without user intervention beyond a remote command. The framework drives the device into this state from active by sending the relevant remote command, waits for the device to enter the state and stabilize, and then records the steady-state power. The transition itself is verified through the ADB control channel before the measurement is recorded.

5.3. Passive Standby

Passive standby is the state in which the device is in an off-like state but still drawing power because it remains plugged in. The framework drives the device into this state through the appropriate sequence of operations, waits for the device to settle, and records the steady-state power. The measurement target is well below one watt and the zero point zero one watt resolution of the meter is what makes the measurement meaningful.

5.4. USB Peripheral State Transitions

Several test cases require the device to be measured with a USB peripheral connected or disconnected. The framework uses the programmable USB switch to perform these state changes on cue. The switch is driven through the orchestration layer, so the test case carries a single instruction to change the peripheral state, and the underlying driver handles the hardware specifics.

5.5. Recording and Provenance

Each measurement is recorded with the device serial number, the firmware build identifier, the bench identifier, the lab location, the test case identifier, the operating state, the measurement value, the units, and the timestamp. The provenance is what makes the record useful for compliance, because compliance submissions need to trace each value back to the conditions under which it was measured.

6. Empirical Results

This section reports the empirical results from production deployment of the framework on Fire TV streaming stick product lines.

6.1. New Product Profiling Time

New product profiling time was reduced by approximately sixty percent, from approximately twenty-eight hours per device to between twelve and fourteen hours per device. The reduction comes primarily from running multiple test cases unattended in sequence, including the state transitions that previously required physical engineer intervention.

6.2. Sustenance Profiling Time

Sustenance profiling, which is the re-measurement of an already-onboarded device after a firmware update, was reduced by approximately fifty percent. The reduction is smaller in proportion than for new product profiling because sustenance profiling did not include the per-device setup overhead that new product profiling did, so there was less initial overhead to remove.

6.3. Automation Coverage

Across the thirty-nine-case regulatory compliance suite, the framework achieved seventy-two percent test case automation. The remaining twenty-eight percent of cases are either too rare to justify the engineering investment of automating, or require a physical interaction with the device that the current bench cannot perform. These remaining cases are run manually.

6.4. Device Count and Lab Locations

Thirty-three streaming stick devices were onboarded across two lab locations. The combined throughput of the two labs is what supports the release cadence of the device portfolio. Concentrating the same number of benches in a single lab would have required a larger room and would have removed the redundancy provided by the cross-lab deployment.

6.5. Metrics Validation as a Companion Workstream

A separate Python automation framework was built to validate device metrics across releases. The metrics framework audits and consolidates the metrics suite and is now adopted as the standard process across all releases. The metrics work is complementary to the power profiling work and shares the same orchestration patterns.

7. Lessons Learned

7.1. Cross-Lab Uniformity Is Not Free

Achieving cross-lab uniformity required explicit attention. Small differences in the bench, the reference instrument, or the host computer configuration produce small differences in results. The framework treats cross-lab uniformity as an invariant and re-validates each bench against the local reference periodically to detect drift early.

7.2. Unattended Operation Magnifies Small Defects

Defects that are invisible in attended manual operation become visible in unattended automation. An engineer who is present can intervene when the device does not enter the expected state. An unattended bench cannot. The framework therefore had to be hardened against the long tail of failure modes that the manual process simply absorbed.

7.3. The Schedule Is the Right Unit of Throughput

Treating the schedule, not the test, as the unit of throughput is what makes the framework efficient. A bench that is always working on the next scheduled test is fully occupied. A bench that waits for an engineer to start the next test is not.

7.4. Reports Are Part of the System

The reports generated from the database are not an afterthought. They are the artifact that the compliance team uses to make submissions. The structure of the reports is part of the system design and not a downstream convenience.

7.5. Onboarding Engineers Pays Back Quickly

The investment in writing the bench onboarding procedure paid back faster than initially expected. An engineer who has not previously onboarded a bench can complete the procedure in a bounded amount of time, which removes the bottleneck that would otherwise concentrate onboarding on a small number of senior engineers. The procedure is treated as a living document and is updated whenever an onboarding session surfaces a step that the document did not cover. This continuous improvement of the procedure keeps the cost of adding a bench predictable as the portfolio grows.

7.6. Manual Cases Are Not Failures

The twenty-eight percent of test cases that remain manual are not a failure of the framework. They are cases for which the engineering investment of automation does not pay back, either because they are run too rarely or because they require a physical interaction that the current bench design cannot perform. Treating these cases as expected residual work, rather than as automation gaps to close at any cost, keeps the framework's roadmap aligned with where the actual value is.

8. Conclusion

Power profiling for energy compliance on consumer streaming devices is a workload that benefits substantially from hardware-in-the-loop automation. The framework described in this paper reduced new product profiling time by approximately sixty percent and sustenance profiling time by approximately fifty percent, achieved seventy-two percent test case automation across a thirty-nine-case regulatory compliance suite, and supported thirty-three devices across two geographically distributed lab locations. The architectural decisions that made this possible, in particular the bench as the unit of automation, the uniform

Python orchestration layer across benches, the schedule-driven unattended operation, and the centralized database with provenance, transfer to other consumer electronics test workloads with similar characteristics. Test engineering teams facing similar compliance demands can apply the same patterns with attention to the specifics of the regulations and devices they cover.

Acknowledgments

This work was performed in the context of power and thermal profiling for Fire TV streaming stick product lines. The author thanks the cross-functional teams responsible for the device platforms, the lab infrastructure, and the compliance submissions.

References

[1] Commission Regulation (EC) No 1275/2008, Ecodesign requirements for standby and off mode, and networked standby, electric power consumption of electrical and electronic household and office equipment.
[2] Commission Regulation (EU) 2023/826, replacing Regulation (EC) No 1275/2008.
[3] IEC 62301, Household electrical appliances: Measurement of standby power.
[4] IEC 62087, Methods of measurement for the power consumption of audio, video, and related equipment.
[5] US Department of Energy, Test Procedure for Television Sets, 10 CFR Part 430 Appendix H.
[6] California Energy Commission, Appliance Efficiency Program regulations.
[7] Natural Resources Canada, Energy efficiency regulations for consumer electronics.
[8] Bureau of Energy Efficiency, India, Standards and Labeling Program.
[9] Beyer, B., Jones, C., Petoff, J., and Murphy, N. R. Site Reliability Engineering: How Google Runs Production Systems. O'Reilly Media, 2016.
[10] Humble, J. and Farley, D. Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation. Addison-Wesley Professional, 2010.
[11] Myers, G. J., Sandler, C., and Badgett, T. The Art of Software Testing, 3rd Edition. Wiley, 2011.
[12] United States Environmental Protection Agency. ENERGY STAR Program Requirements for Set-Top Boxes Eligibility Criteria, Version 4.1, 2017.
[13] International Energy Agency. More Data, Less Energy: Making Network Standby More Efficient in Billions of Connected Devices. IEA, 2014.
[14] Python Software Foundation. The Python Language Reference, Release 3.x. Python.org, 2023.