Simulating Reality: Unlocking Device Behavior Through Virtual Models

By

Sathvika Gambheerrao

Tata Institute of Fundamental Research, Hyderabad

Introduction To Virtual Simulations

Software device simulators are virtual models that replicate the functionality of devices in a software environment.

They are crucial for testing applications without the need for physical hardware.

Why Simulation?

Restricted Access to Physical Devices	Simulations allow testing and development when physical devices are unavailable or inaccessible, ensuring progress without delays.
High-Stakes Equipment	With devices worth millions of pounds, simulations help validate commands and prevent the risk of sending incorrect or harmful instructions, avoiding costly damages.
Pre-Delivery Testing	Simulations enable Factory Acceptance Tests (FAT) to be performed before device delivery, ensuring compliance with specifications and reducing deployment risks.

Why Simulation?

Offline or Non-Operational Devices	Simulations replicate device behavior even when the actual hardware is offline, enabling continuous testing and development
Limitations of Manufacturer-Provided Software	Manufacturers sometimes provide software for their devices, but it may be inefficient, limiting functionality or failing to meet the specific needs of the facility

Benefits Of Simulation

Cost Efficient

Reduces the need for expensive hardware

Risk Reduction

Allows for safe testing of hazardous scenarios

Speed

Accelerates the development and testing process

Challenges In Device Simulation

Accuracy	Development time	Firmware Update	Documentation
• Ensuring simulations accurately represent real-world scenarios.	• Simulating complex devices takes more time	• Needs updating the simulator every time firmware updates	• It is important to have proper documentation

Best Practices For Implementing Device Simulators

Define Clear Objectives	Understand what you aim to achieve with simulations.
Select Appropriate Tools	Choose simulators that best fit your needs.
Ensure Compatibility	Verify compatibility with existing systems.
Regular Updates	Keep simulators updated with the latest features and device models.

Simple Device Simulations

Simulators are often built for straightforward devices that operate in a question-and-answer format, allowing developers to send commands and receive predictable responses for testing and validation.

https://github.com/e9ctrl/vd

Complex Device Simulations

Complex Device Simulations with Lewis, a Python-based package

State Machine Support

- Simulates complex devices with multiple states (e.g., *idle*, *moving*, *error*).
- Enables accurate emulation of real-world device behavior.

Time-Based Simulations

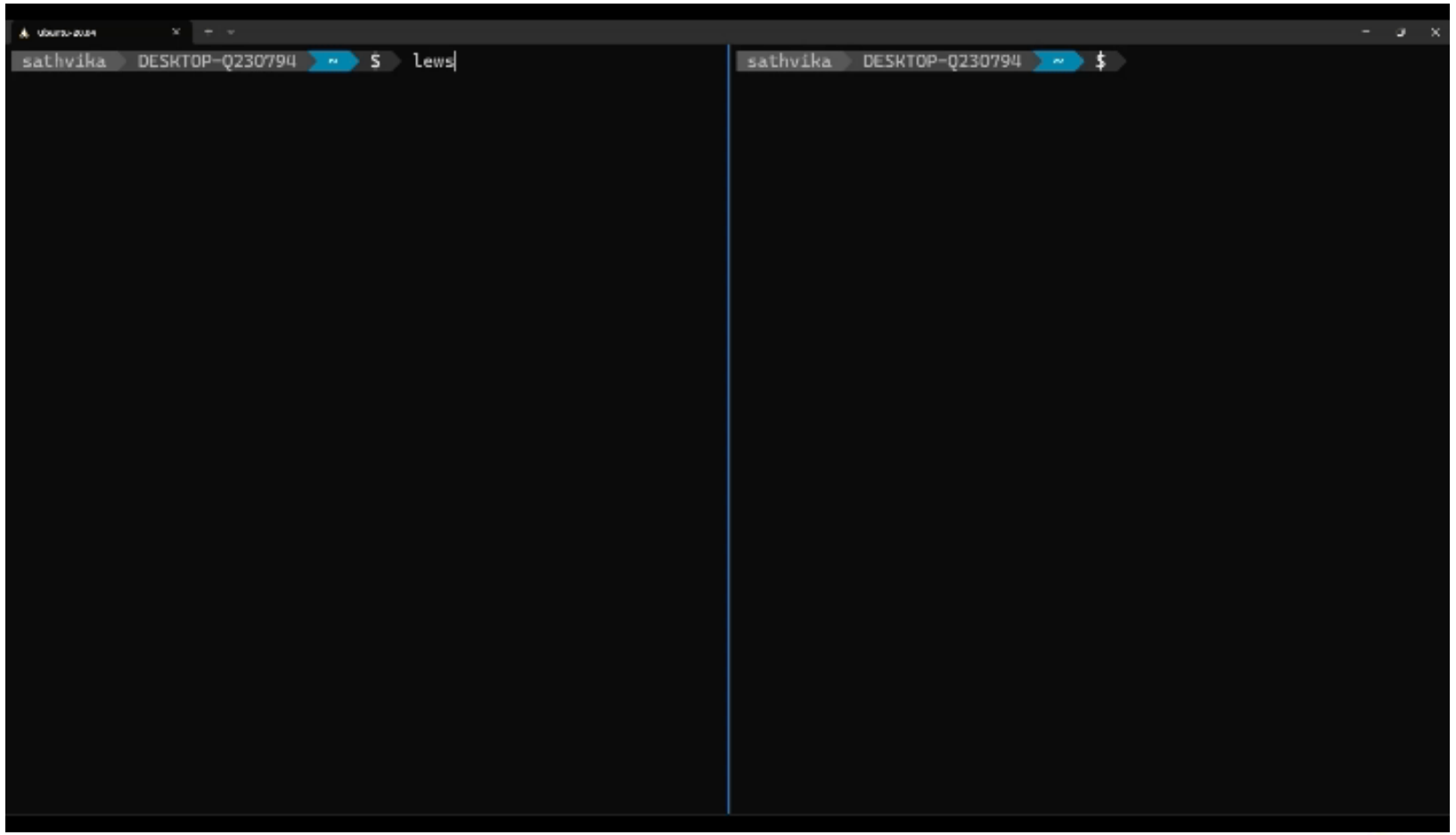
- Models time-dependent processes and device responses.
- Useful for scenarios like gradual state transitions or delay simulations.

Flexibility and Scalability

- Adapts to complex device protocols beyond simple question-answer interactions.

https://isiscomputinggroup.github.io/lewis

Motor Example Simulation Using Lewis



Conclusion: The Impact Of Virtual Models

Risk Reduction — Prevents potential damage to the equipment by allowing thorough testing of commands and operations in a virtual environment.

Speed — Enables software development and testing to proceed without waiting for physical devices.

Improved Collaboration — Allows teams to work remotely fostering better coordination and productivity.

EPIC
Extreme Photonics Innovation Centre

central laser facility

THANK YOU!

tifr

UKRI Science and Technology Facilities Council