MAY1733

TEAMNAME

Testing Environment for Accessing and Monitoring Networked Automation and Measurement Equipment

Team Members | Christian Hurst | Chris Little | Antonio Montoya | Braden Rosengren | Benjamin Wiggins

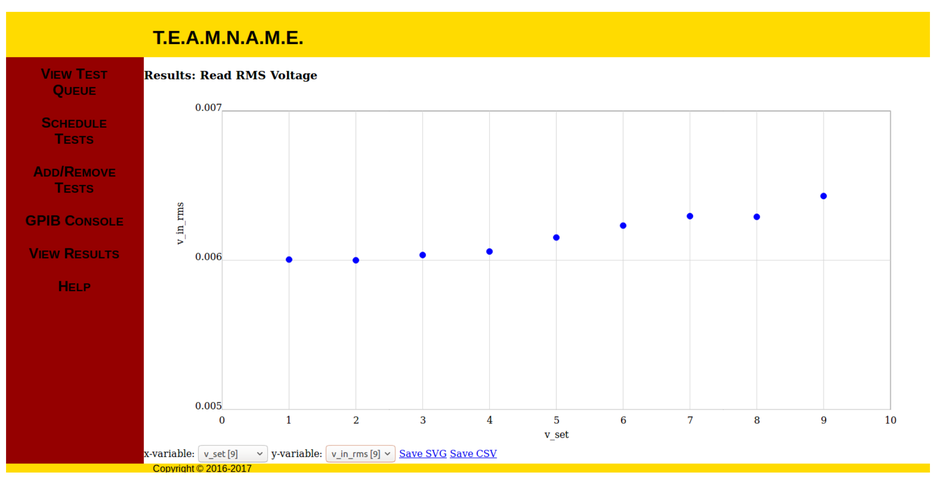
Faculty Advisor | Dr. Randall Geiger

Problem Statement

There are not many test equipment automation software solutions available to engineers. Those that are available are often expensive, unintuitive, and usually only support a limited set of devices or run only on specific platforms.

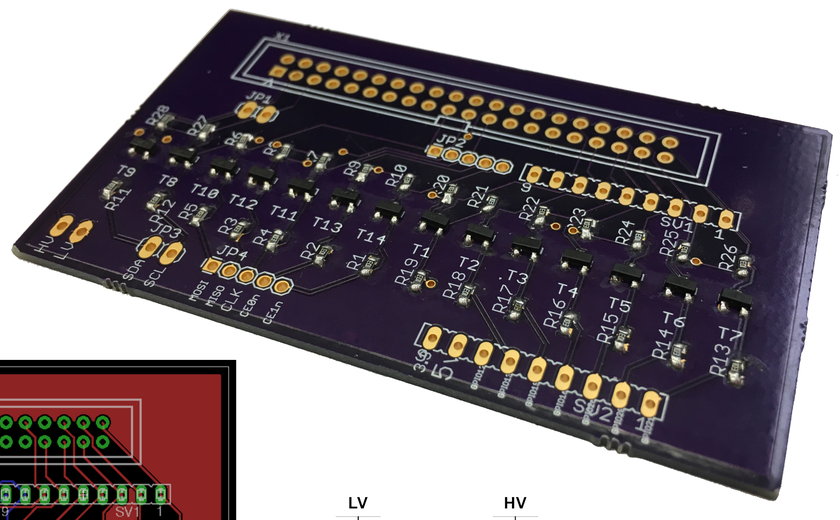
Solution

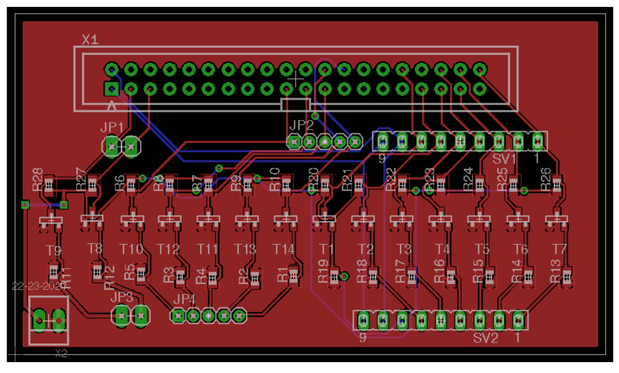
Our solution is to develop an open source platform that is easy to use, inexpensive, and can be easily expanded to use the full feature set of any piece of test equipment available to the user. By using inexpensive hardware and open source software, we are able to reduce costs for the end user and avoid proprietary software licenses. Our web-based interface allows users to connect from any platform over the local network and view their results or download them for further analysis.

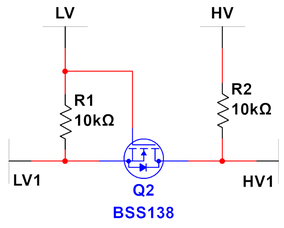


Hardware

Our hardware goal was to create a breakout board that would allow us communication through SPI, I2C, and GPIO to integrated circuits and silicon wafers that operate at a different voltage than the Raspberry Pi. This breakout board uses an array of variable level shifters to translate between the 3.3 V GPIO pins on the Raspberry Pi and the operating voltage of the IC being tested.







Testing

- Before the breakout board was designed, the level shifter circuit was verified through simulation. After verification and board design, the breakout board was verified through measurement to ensure it performed as expected.
- Our software was tested regularly throughout its design process. Individual features were tested upon implementation. Additionally, our software was systematically tested with actual run-time scenarios in order to look for points of failure.

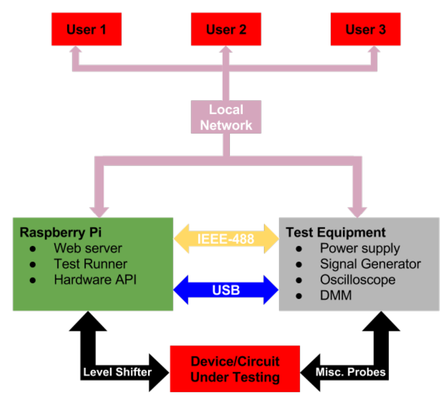
Project Goals and Requirements

In this project, we aim to:

- Develop remote hardware test automation and monitoring capabilities
- Create user-expandable API and development tools.
- Provide an inexpensive alternative to more costly automation solutions
- Demonstrate conceptual viability of standardized hardware API and testing system
- Present plan for creating an open source platform

Our final product must:

- Provide a web-based GUI to provide cross-platform functionality
- Interface with lab equipment using a variety of methods
- Initialize all required processes at startup (without user intervention)
- Include a variable logic level shifter to allow for digital communication between the Raspberry Pi and the device under test
- Include documentation to allow users to expand functionality



Software

- Hosts a web server allowing users to connect to TEAMNAME from over the local network
- Communicates with test equipment using GPIB commands over IEEE-488, USB, and Ethernet
- Automates the running of test procedures and saving of measurements for later viewing and retrieval
- Allows user to upload test procedures and download measurements easily over the network
- Open-source software reduces expenses and encourages community development
- User-extensible API and development tools allow a wide range of hardware support
- Web-based interface provides functionality for Windows, Mac and Linux clients