ELECOMP CAPSTONE SUMMIT
FRIDAY, MAY 10, 2019

THE UNIVERSITY OF RHODE ISLAND

THINK BIG WE DO™

Gen-2 SPEC Tester

The Next Generation of System Power Extended Cycling (SPEC) Testing

Team Members: Alexa Darche (ELE), Tyler Kaiser (ELE), Zach Galipeau (ELE)

Technical Directors: Darryl Galipeau and Mark Rodrigues

Project Motivation

Testing a product for reliability is a vital step before it can enter production. That is why Infineon subjects their voltage converters to high current loads for months at a time. Since testing lasts so long, it must be low maintenance. In other words, “Set it and forget it.” Currently, Infineon has no ability to communicate with the devices under test (DUTs). There is also no ability to regulate the temperature of the system. The current solution is to use third party software to monitor the devices, but existing software is very expensive and cannot write to the DUTs. Since the new system would be an in-house program, it would save time and money over dealing with a vendor. The new program will be able to communicate with the DUTs, collect all the relevant data, and will be a significant cost reduction for Infineon.

Key Accomplishments

Hardware: To develop a temperature control system, an initial connection to the test system needed to be established. The test system has an I^{2}C bus which connects all the devices to a multiplexer header. Using a test Arduino sketch and fan controller board, we were able to see all the devices on the I^{2}C bus and develop our own code to read and write to the board. An Arduino was used to interface the testing board with the computer software. The test board and actual sites use more or less the same device addresses, so the information found using the Arduino test sketch could be used during the full system testing.

Site Communication: One of the first major milestones was establishing communication to the devices on the each site. This was done through Arduino using I^{2}C communication protocol. To do this, the address of each device must be set to a unique value. Since the sites all use identical device addresses, this meant that only one set of addresses needed to be stores on the Arduino. The device used only allowed for 9 addresses per site. To bypass this, a multiplexer was used to mirror 5 site addresses on two channels. The redundancy in device addressing made expanding the communication Arduino sketches fairly straightforward.

E-Fuse: The E-Fuse is designed to disconnect a site in the event of a voltage or current error to prevent overheating. This protects the system from damaging itself and is a crucial part of the temperature control system design. This was found to be controlled by a pin on the main board. When the GUI detects any type of fault the pin is set high, activating the E-fuse. The E-fuse is a pin on the site and is driven high to activate. Because there is no communication bus involved, an Arduino general programmable input/output (GPIO) pin is used to control this feature.

E-Load: The system itself is designed to stress test voltage converters under a high current load. This simulates years of use in only a few weeks. As of now, the load current is set manually with a potentiometer, which is tedious and takes valuable time out of the testing process. The GUI has the ability to talk with and directly program the electronic load. This gives the end user simple and intuitive control over each individual load board, saving time and improving efficiency.

Graphical User Interface: A fully customized graphical user interface (GUI) was delivered to the application engineers at Infineon. This GUI allows a user to specify a target temperature as well as seen the current thermal output of the testing rack. The user can view all 60 sites simultaneously, allowing for error detection and thermal verification, or can view each site individually, allowing for viewing logged data and error specifications. The Arduinos are completely handled by the GUI, from uploading the program to reading and writing information to the board. In addition, the load boards of each site can be controlled using the GUI. This results in an smooth experience for the end user and significantly more control than the previous interface.

Database: A SQLite database was used to log the information from the system and display to the user in the GUI. SQLite was chosen for its versatility and mobility, as a single .db file can be stored locally and accessed from the user interface. Eliminating the need for a seperate SQL manager, such as SQL Server Management Server made it easier for the Infineon application engineers to install and implement the GUI on their company computers

Anticipated Best Outcome

The goal was to have full thermal control of all 60 sites. In order to streamline more complex calculations, namely $R_{DS(ON)}$ testing, temperature should remain constant. This was not possible as the fan speed of each site was set to a constant value. The new system has a graphical user interface (GUI) where Infineon application engineers can specify a desired system temperature. The system relays this information to a fan controller on each site, which dynamically sets the fan speed to maintain this user specified temperature.

Project Outcome

The Anticipated Best Outcome was achieved. We were able to deliver a functioning system to Infineon that controls the thermals of the SPEC Testing system.

Figures

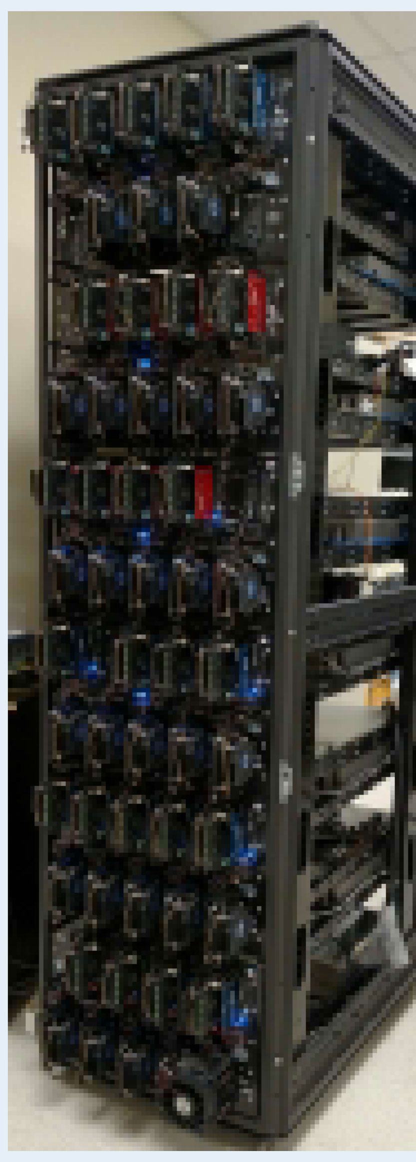

Fig. 1: SPECTester system tower at Infineon Technologies lab in Warwick, RI. The system features 60 individual sites for testing phase cards under varying loads.

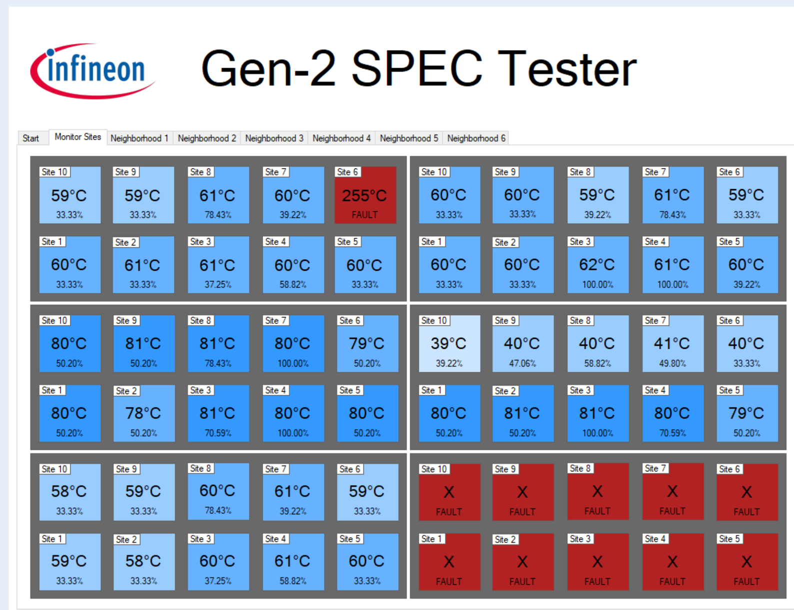

Fig. 2: GUI Panel View used for displaying the status of each site in an overview.

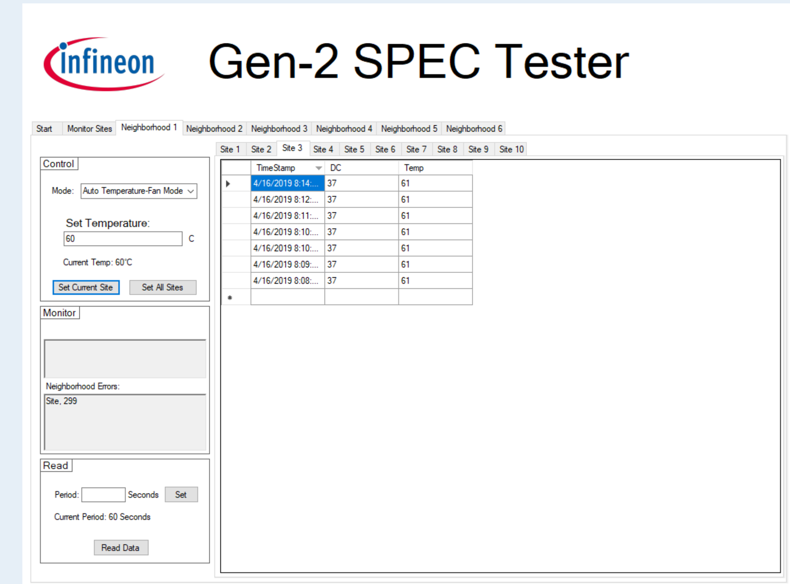

Fig. 3: GUI Site View developed by team. Displays the current and logged information of a given site, as well as any faults that may have occurred. Allows for control of individual thermal systems by setting a target temperature or fan speed.

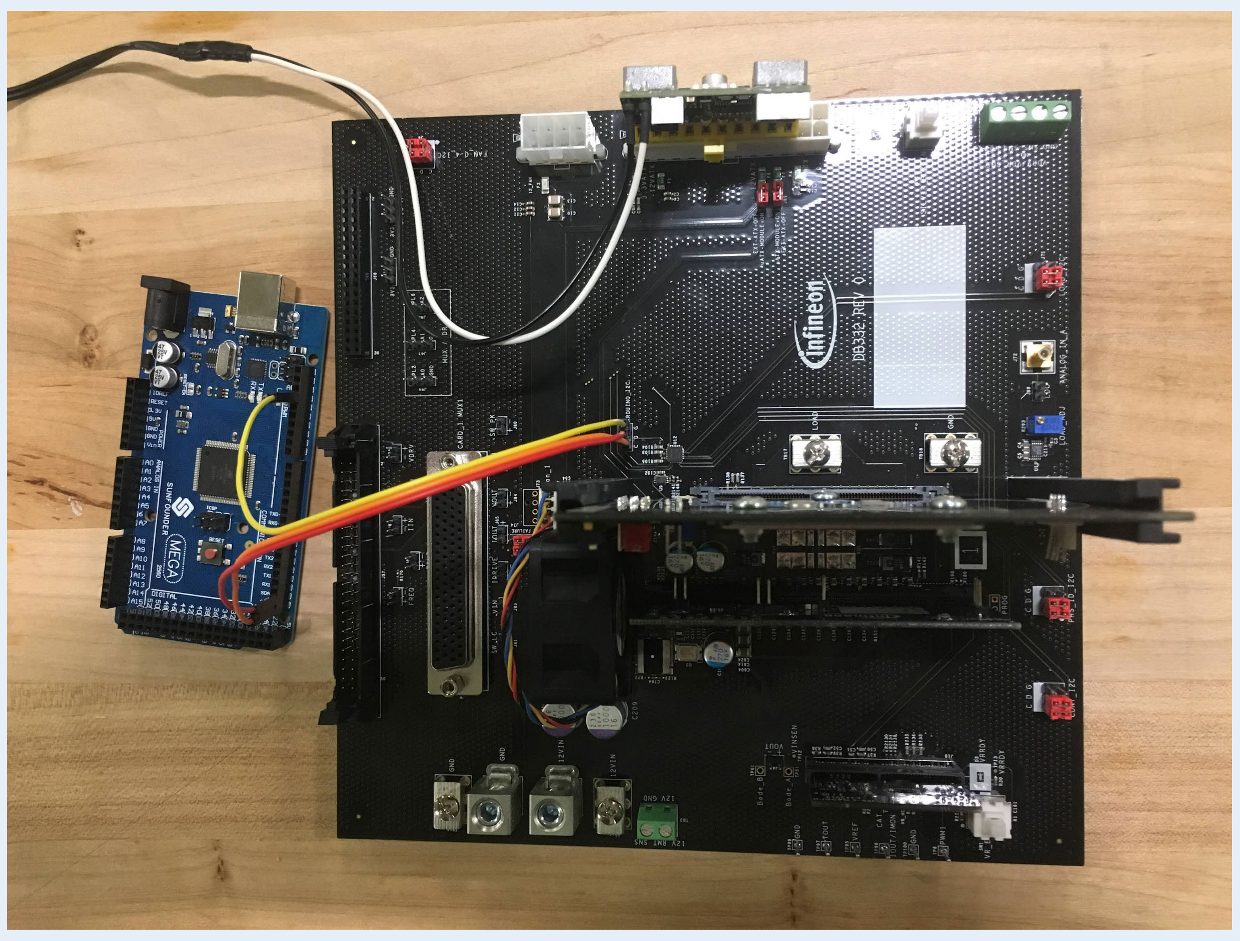

Fig. 4: A single site which was used to design and debug the system. An Arduino was used to interface the board to the GUI.