

FLEXIBLE PRODUCTION TEST SOFTWARE DESIGN

The challenge with designing a flexible production test system is providing a software architecture that allows for quick test application development for a variety of products. To address this challenge, our engineers at G Systems used the NI TestStand and LabVIEW test sequences developed for one product and easily migrated those test sequences to support the development of many new products. By abstracting the instrument drivers and building configurable step types in NI TestStand, the system can support a variety of test hardware configurations and dynamic configuration of test parameters. The result was a flexible production test system that meets the ever-changing requirements of today's manufacturing environment.

THE CHALLENGE

Design and implement a custom functional test system to produce many different types of semiconductor hybrid products. The system needed to have a flexible test architecture, easy-to-use development environment, intuitive production graphical user interface (GUI), instrument and product interchangeability, reduced maintenance cost, and the ability to archive product test data.

THE SOLUTION

G Systems created a custom test executive with user-configurable step types. This architecture provides the user with a flexible method to quickly assemble and configure new custom test sequences, which allows production testing of many different families of hybrid semiconductor components. They also used the NI TestStand executive to execute the test sequences, perform test step pass/fail evaluation, and archive the final test results for future analysis.

OVERVIEW

This project involved replacing the manual process of testing hybrid components using bench instruments with an automated test system. For this system we selected NI TestStand and LabVIEW software as well as computer-based automated test equipment (ATE) as shown in Figure 1.

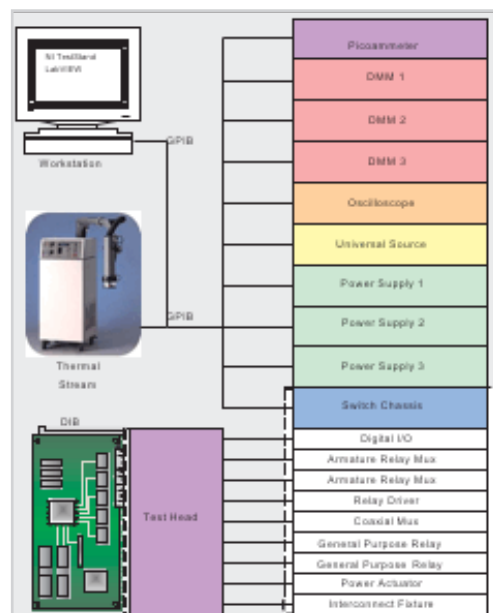


Figure 1. Test System Hardware Diagram

SOFTWARE ARCHITECTURE

The major components of the software architecture implemented by G Systems are shown in Figure 2. These components include the instrument abstraction layer, configurable instrument layer, sequence editor, NI TestStand engine, process model, and GUI.

INSTRUMENT ABSTRACTION

The instrument abstraction layer consists of several instrument classes for instrument interchangeability. Interchangeable virtual instrument (IVI) classes and drivers are employed. Where IVI drivers do not exist, a “wrapper” class calls existing plug-and-play instrument drivers, and abstracts instrument functionality for each class. The instrument classes automatically detect hardware models and provide a high-level interface for seamless code integration.

The configuration file structure provides instrument ID extraction through the instrument’s name (e.g., [DMM1], [PS1]). System instruments can be configured without modifying the low-level code by simply editing the configuration file. The instrument manager VI is the core component for instrument communication. All the instrument information is stored in the instrument manager’s internal cache shift registers.

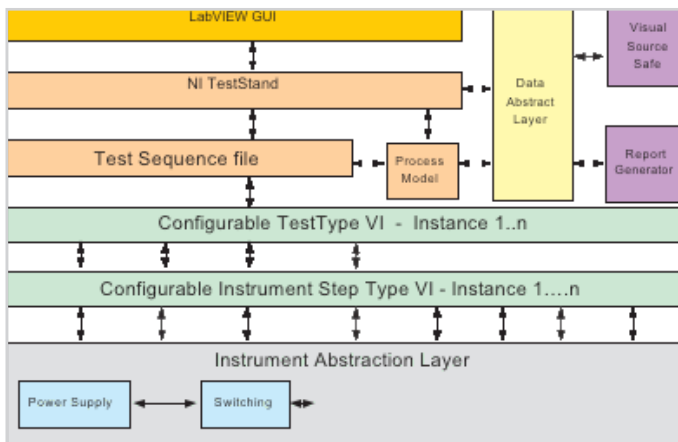


Figure 2. Software Component Architecture using NI TestStand and LabVIEW

CONFIGURATION TEST STEPS

The configurable test step enables the user to configure test parameters during sequence editing without additional programming. Configurable test steps provide the user with a GUI to configure the test parameters. When the user configures a test step they can set test parameters while visualizing how those parameters affect the pass/fail test status. Figure 3 shows the configurable instruments and test step types selection from the menu.

Custom configurable steps were built for each instrument. Based on the test plan, several configurable test step types were developed for performing measurements. These custom configurable steps give the user the flexibility to script new test sequences for testing hybrid circuits.

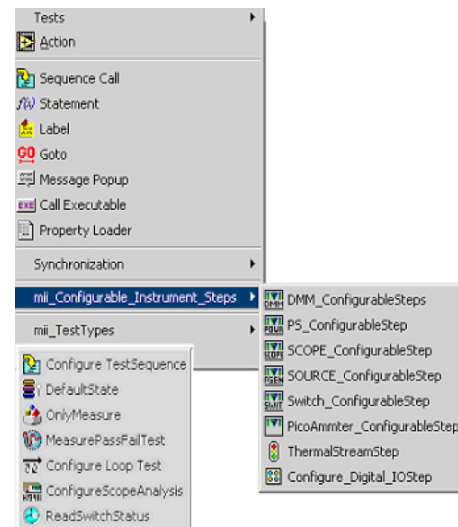


Figure 3. Custom Configuration Step Types

SEQUENCE EDITOR

For debugging and testing a new unit under test (UUT) model, the test engineer or technician uses the NI TestStand Sequence Editor (Figure 4) to execute the test sequence and process model. The Sequence Editor is easy to use during new product development because it offers the flexibility to change test parameters, modify execution flow, and view watch variables in a debug mode.

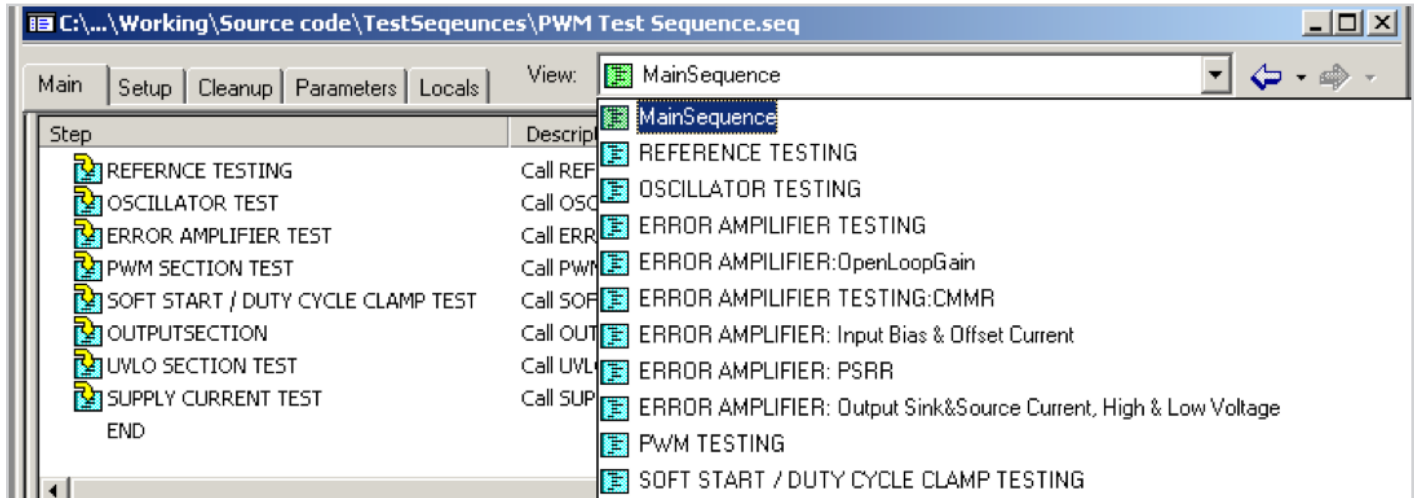


Figure 4. The Sequence Editor

NI TESTSTAND ENGINE

The Active X GUI controls the NI TestStand engine for loading sequence files and executing tests, and it provides centralized run-time memory space for shared data. Using NI TestStand API calls (UI messages), data is transferred between NI TestStand and the custom operator interface. The NI TestStand property loader loads the test limits and parameters based on the temperature at which the component is being tested.

PROCESS MODEL

The test system process model behavior is subdivided into five main subsequences. During Model PreUUTLoop, the process model identifies, initializes, and configures each instrument. Model PreUUT sets the temperature, makes the connections to the UUT, and performs the recurring initialization tasks. MainSequence performs the main sequence of test steps while evaluating results against test limits. Model PostUUT increments the temperature index and is called during shutdown to close instrument handles and perform other necessary shutdown functions.

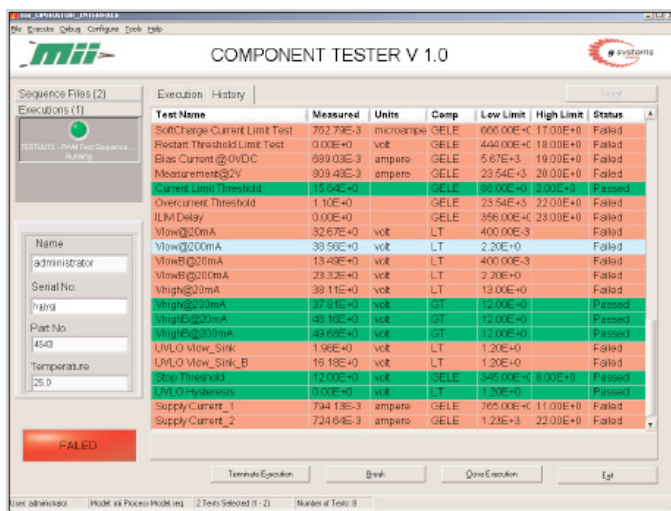


Figure 5. LabVIEW Test Executive GUI

PROCESS MODEL POST RESULTS LIST ENTRY

In this call back, data saving is implemented. This call back is triggered only when the main step “record results property” is enabled. A precondition is set to log the data when the data logging function is enabled and the step type is a custom numeric limit or a pass/fail test.

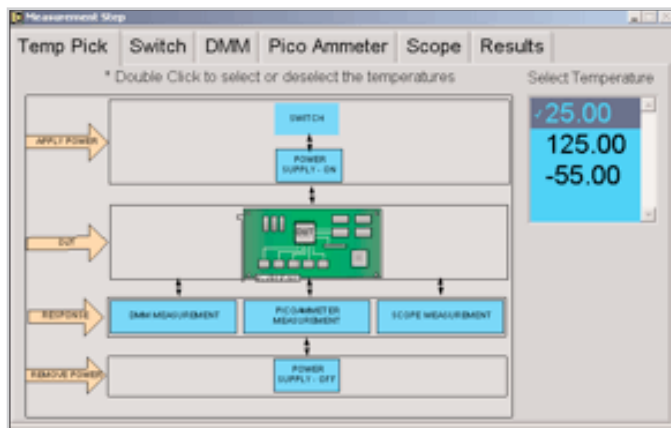


Figure 6. Configurable Measurement Step

CONFIGURATION ENTRY POINT

The Configuration Entry Point is a LabVIEW GUI that allows the user to enter the part and lot number, limit file and data saving directory paths, test temperature selection to be executed, and the data saving enable-disable option.

THE GRAPHICAL USER INTERFACE AND PERFORMING MEASUREMENTS

G Systems developed a custom GUI using LabVIEW to display information important to the customer. (Figure 5).

Many of the hybrid test system component sections were tested for proper functionality at different temperatures by the ATE system. Specific measurements included reference, oscillator, output, error amplifier, pulse-width modulation, under voltage lock out, and soft start section testing.

Comparison Parameter	Manual Testing	Automated Testing
Architecture	No architecture, no reusability	Custom configurable steps provide the flexibility to quickly script new test sequences for new hybrid devices. The tests developed for one device are easily used for testing many other devices.
Accuracy and Repeatability	Test accuracy and repeatability dependent on the production operator.	Data measured using the new functional test system is repeatable and eliminates test operator process errors.
Skills Required for Testing	A skilled technician required to perform test.	A production operator can perform all required test.
Test Time Reduction	Manual lab environment, very time consuming	10X increase in production throughput

Table 1. System Performance Comparison for Manual vs. Automated Testing

Some of the specific tests that can be performed include measuring the input bias/offset current and voltages, common mode rejection ratio, power supply rejection ratio, clock high and low, thresholds, rise time, fall time, TON, and TOFF. An example of a configurable measurement step is shown in Figure 6.

THE BENEFITS OF OUR AUTOMATED TEST SYSTEM

Using NI TestStand and LabVIEW software, G Systems engineers quickly created a user-configurable hybrid semiconductor functional

test system. This custom system solution saved our customer more than \$100,000 compared to buying an entry-level commercially available mixed signal tester. The system greatly reduces the product test cycle time versus the previous method of manual test. The system also provides a flexible test architecture, easy-to-use development environment, intuitive production GUI, instrument and product interchangeability, custom configurable instruments and test steps, reduced maintenance costs, and the ability to archive product test data.

To see how G Systems can help on your next project, download our [capabilities brochure](#).



G Systems, L.P.

1240 E. Campbell Road, Ste. 100 Richardson, TX 75081-1935

Phone: 972-234-6000 **Fax:** 972-234-6018

Online: www.gsystems.com **Sales:** sales@gsystems.com **Information:** info@gsystems.com

CAGE Code: 3HPP5 **NAICS:** Primary 541330, Secondary 541511 334515 334519 335999

Product and company names listed are trademarks or trade names of their respective companies.