Build an Automated Test System with TestStand and the NI PXI Platform

Featuring: TestStand, PXI, LabVIEW, Switch Executive, and NI Modular Instruments

PXI

NATIONAL INSTRUMENTS

Today, We'll Explore:

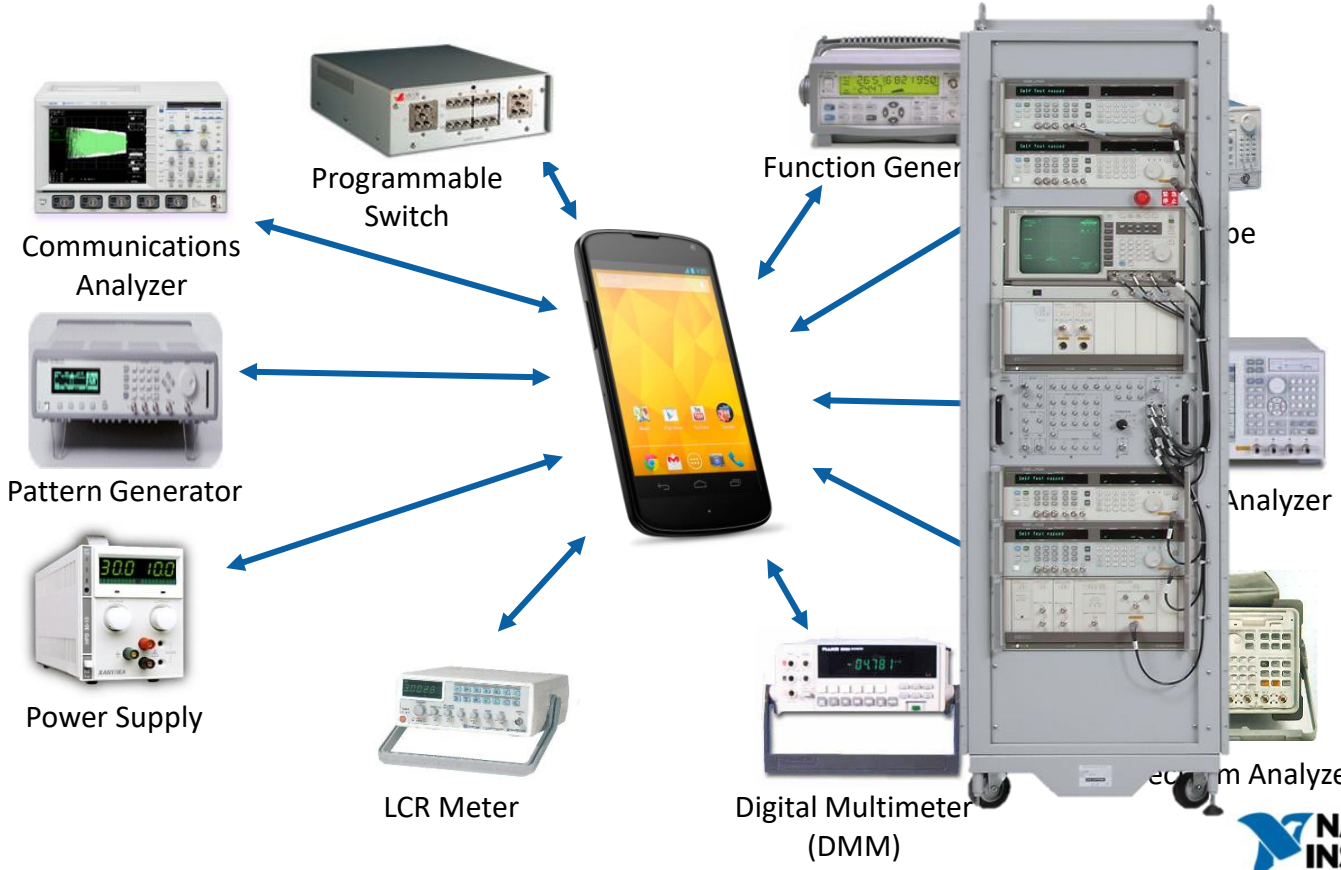
- The Challenges of Developing Test and Measurement Systems
- The NI Test Platform
- Hands-On Hardware
- Automated Test with TestStand
- Scaling Up Production with Parallel Testing
- Test System Deployment

The Origin of Test and Measurement

- Pen and paper approach
- Out-of-the-box software

Today's Designs: Converging Complexity

Today's Challenges: The Traditional Approach



Traditional Test Solutions

Challenging to develop and maintain

- Disaggregate hardware
- Closed software solutions
- Conflicting programming approaches
- Lack of tooling experience
- Limited analysis and visualization
- Hard to integrate new technologies

NATIONAL INSTRUMENTS

NI's Value Proposition

PXI Modules
>1,500 options from over 60 PXI vendors

Software
Flexible driver APIs, example code, soft front panels, and configuration

PXI Controllers
Performance embedded–-Windows or RT OS remote control via desktop or laptop

PXI Chassis
Options ranging from low-cost, 4-slot desktop to high-performance,18-slot rack-mount

Market Shift Toward PXI for Automated Test

Benchtop Test		Automated Test
Interactive use	⟷	Ease of automation
Time to first measurement	⟷	Speed of measurement
I/O connectivity	⟷	Integrated, modular I/O
Vendor-defined measurements	⟷	Software-defined measurements

Traditional Box Instrumentation

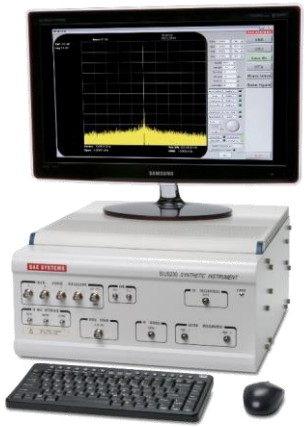
PXI Modular Instrumentation

BAE PXI-Based Synthetic Instrumentation

Traditional Rack and Stack

Replaced with:

PXI Modular Instruments

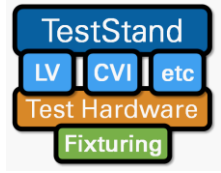


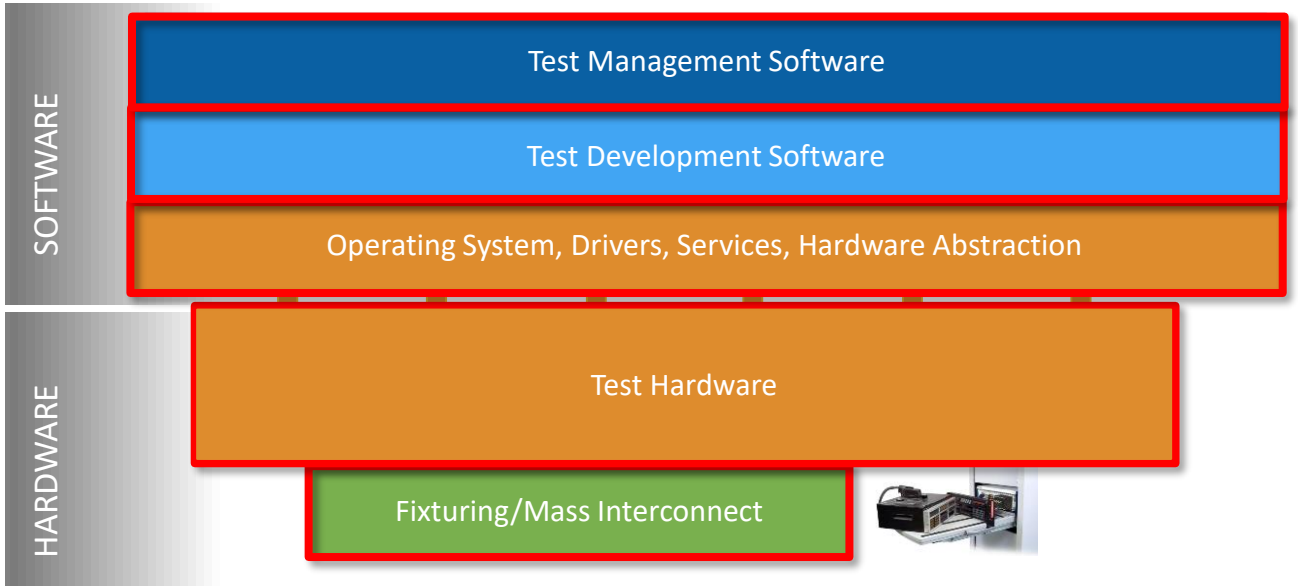
- Smaller
- Faster
- More Accurate
- Less Expensive

Introduction to the NI Automated Test Platform

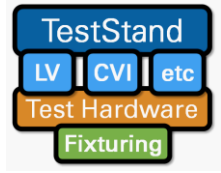
NATIONAL INSTRUMENTS

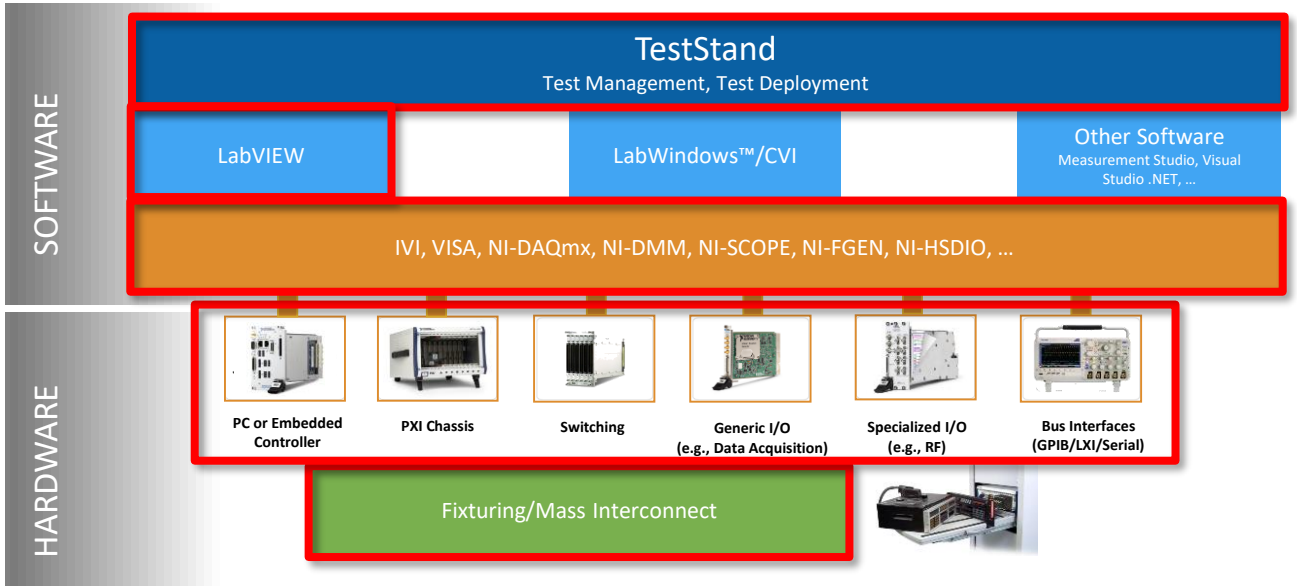
Components of a Test Platform





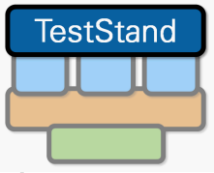
NI's Automated Test System Solution





NATIONAL INSTRUMENTS

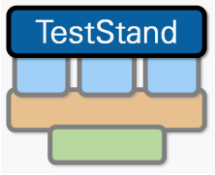
Common Needs



No matter the industry, every company has test needs for its products to ensure quality. A proper automated testing framework (either built in house or off the shelf) should:

- Provide a flexible, open test architecture
- Eliminate programming of common test executive tasks
- Enable reuse of existing test code
- Simplify updates and long-term maintenance
- Accelerate test system development
- Ensure high test system performance

Role of Test Management Software



Test System Components

Operations **different** for each device tested:

- Calibration
- Configuring instruments
- Data acquisition
- Measurements
- Analyzing results
- Test strategies

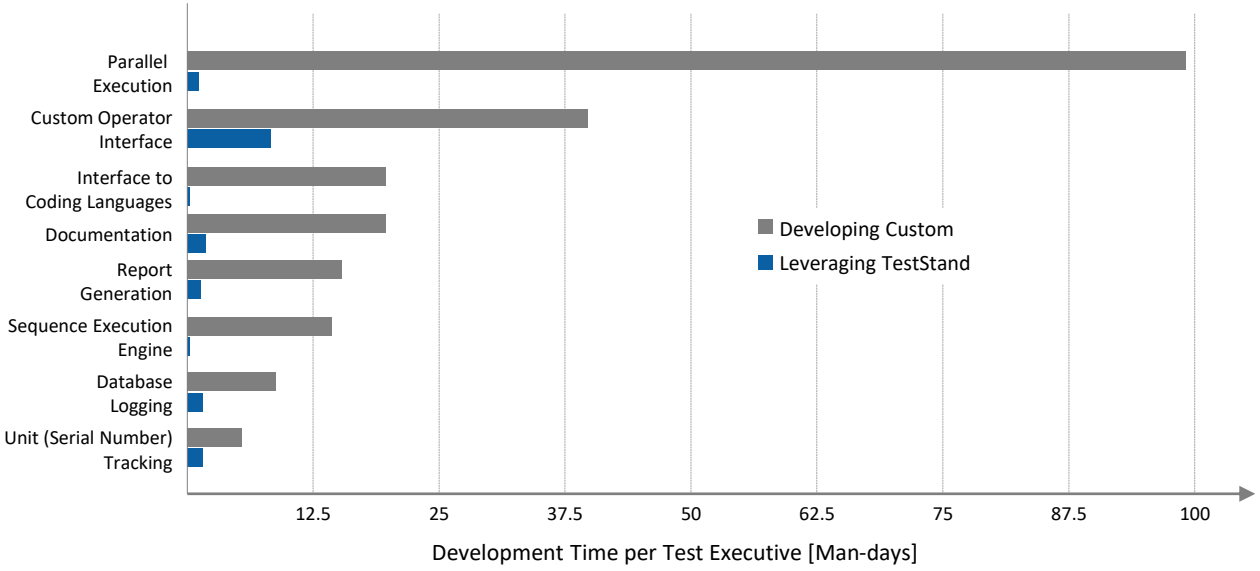
Operations **repeated** for each device tested:

- Operator interfaces
- User management
- Unit under test (UUT) tracking
- Test flow control
- Archiving results
- Test reports

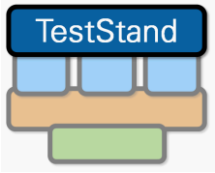
Test Management Software

NATIONAL INSTRUMENTS

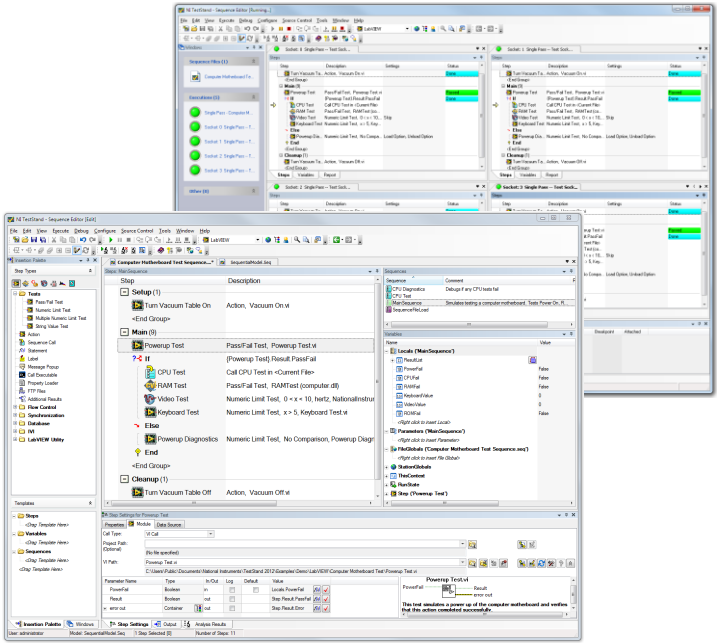
Increased Productivity with TestStand



TestStand Test Management Software



- Graphical development environment
- Automation of tests written in any language
- Multithreaded sequence execution
- ASCII, HTML/Web, XML, and ATML report generation
- Access, Oracle, SQL Server database connectivity



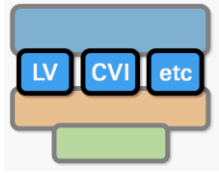
TestStand Test Management Software

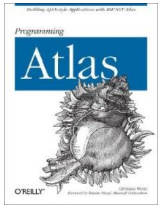
- Full-featured and Simple user interfaces are provided in LabVIEW, LabWindows™/CVI, C#, C++, VB .NET
- TestStand Deployment Utility builds a simple installer with necessary dependencies

NATIONAL INSTRUMENTS

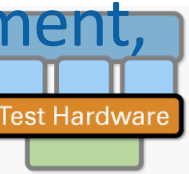
Choosing a Development Environment

- Faster/Easier Development and Debugging
  - Complete and intuitive graphical environment, hierarchal dataflow-oriented design
- High-Performance Execution
  - Efficient and fast executables, advanced multithreading capabilities, proven reliability
- Components
  - VIs, DLLs, COM, A reusable/standard ActiveX server and component capabilities
- Designed for Engineers
  - Third-party party instrument support with more than 10,000 instrument drivers, powerful visualization capabilities, and scientific analysis tools

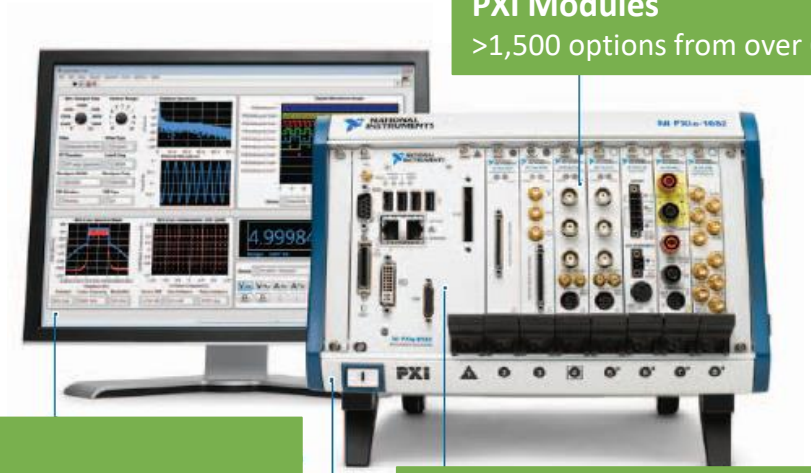




PXI--The Industry-Leading Platform for Test, Measurement, and Control



PXI Modules
>1,500 options from over 60 PXI vendors



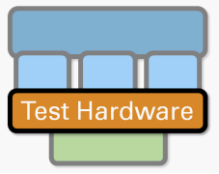
Software
Flexible driver APIs, example code, soft front panels, and configuration

PXI Controllers
Performance embedded–-Windows or RT OS remote control via desktop or laptop

PXI Chassis
Options ranging from low-cost, 4-slot desktop to high-performance,18-slot rack-mount

NATIONAL INSTRUMENTS

Modular Capability for Every Application

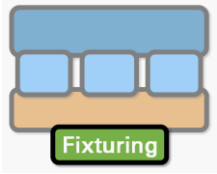


DAQ and Control	Instruments	Interfaces
Multifunction I/O	Oscilloscopes	GPIB, USB, LAN
FPGA	High-Speed Digital I/O	RS232/RS485
Digital I/O	Digital Multimeters	CAN, LIN, FlexRay
Analog Input/Output	Signal Generators	Avionics Buses
Vision and Motion	Switching	I2C/SPI
Counter/ Timer / Clock	RF Analyzers/Generators	Boundary Scan/JTAG
Sensor Measurements	Power Supplies	DeviceNet, PROFIBUS
Reconfigurable I/O	Dynamic Signal Analyzers	SCSI, Ethernet
Signal Conditioning	Source Measure Units	VXI/VME

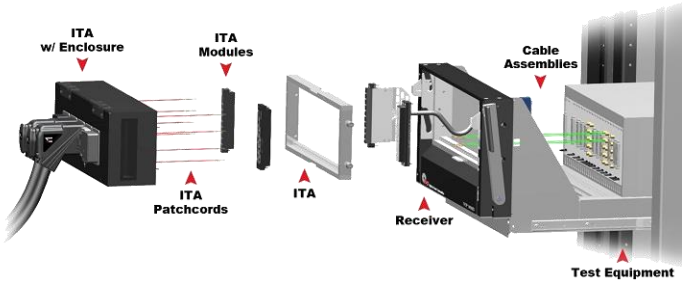
NI Offers 600+ PXI Products

NATIONAL INSTRUMENTS

Fixturing/Mass Interconnects



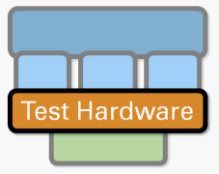
- Hardware designed for quick connection and disconnection with high signal reliability
- Very important for test systems with large numbers of test points



MAC-PANEL
Interconnect Solutions & Services

VPC Virginia Panel Corporation

NATIONAL INSTRUMENTS

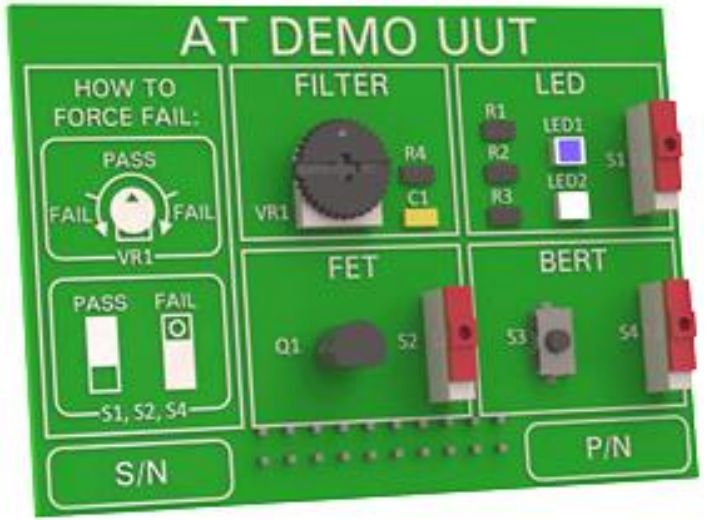


Intro to the Hands-On Hardware

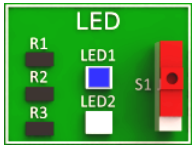
NATIONAL INSTRUMENTS

The Challenge

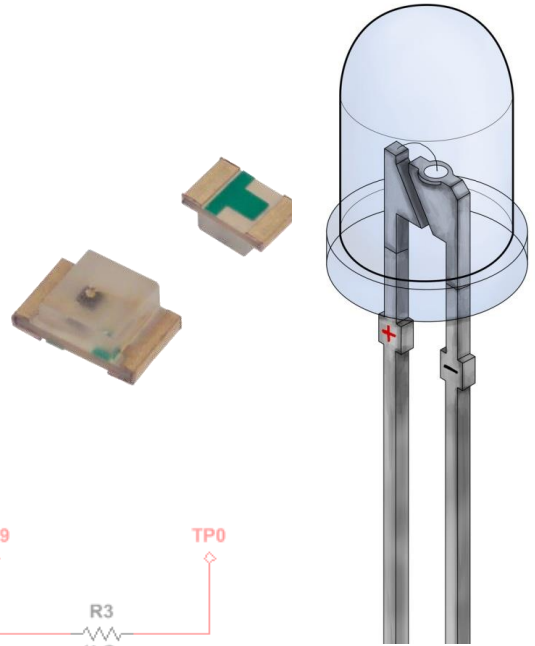
- 4 tests on each UUT
  - Low-pass filter test
  - LED tests
  - FET test
  - Bit error rate test (BERT)
- 4 UUTs in less than 20 seconds
- Rugged, modular, interchangeable UUTs

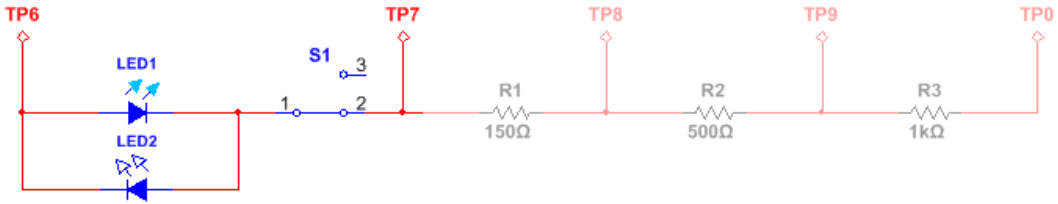


Testing an LED

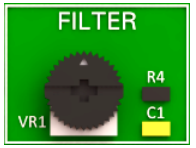


- Provide constant current source and measure voltage drop across leads
- LEDs on our UUTs should measure between 2.4 and 2.9 V
- DMM is used to source current and read voltage
- Switching is critical for connecting DMM to all test points

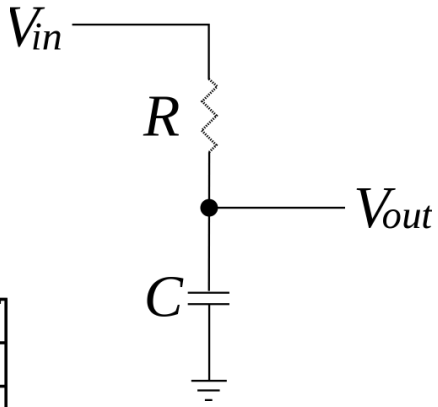


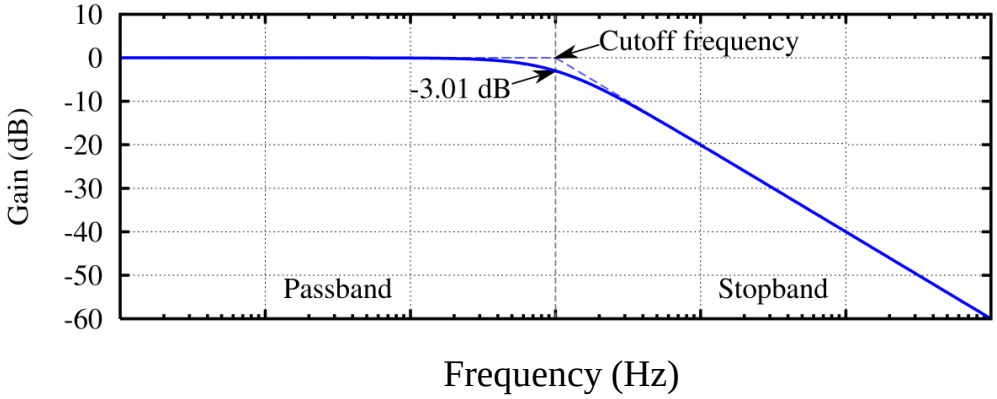


Testing a Low-pass Filter



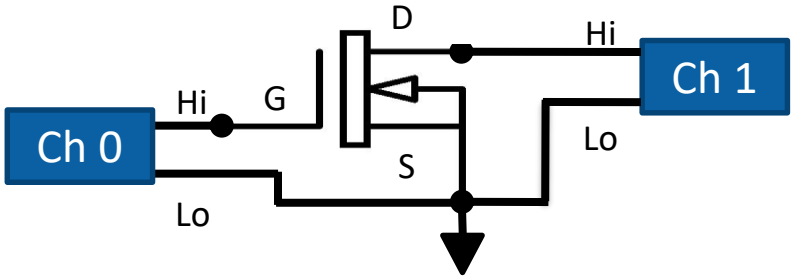
- Sine wave output ramped from 1 kHz to 1 MHz on V_{in} using a function generator
- We've replaced resistor, **R**, with a potentiometer, allowing resistances from 100 to 200 Ω
- V_{out} read by an oscilloscope

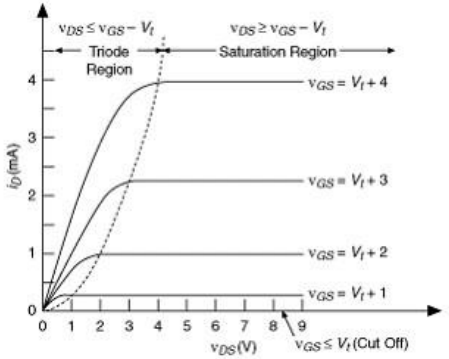


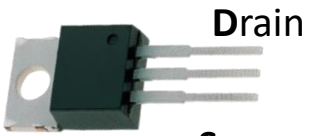


Testing a FET (Field Effect Transistor)

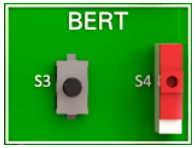
- Operates at low-current levels
  - Requires a high-precision voltage
- Ch 0 steps a gate voltage four times
- For every step of Ch 0, Ch 1 sweeps the drain voltage and measures the drain current
- We'll be using a Source Measure Unit (SMU)



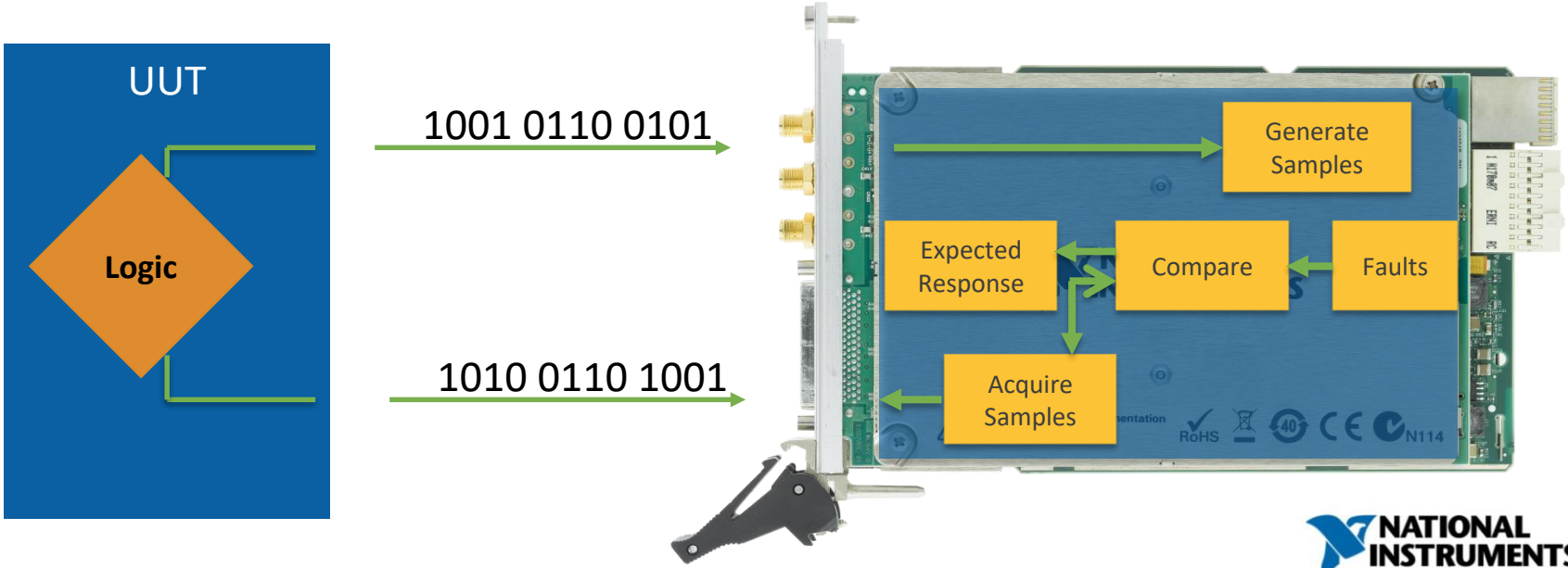




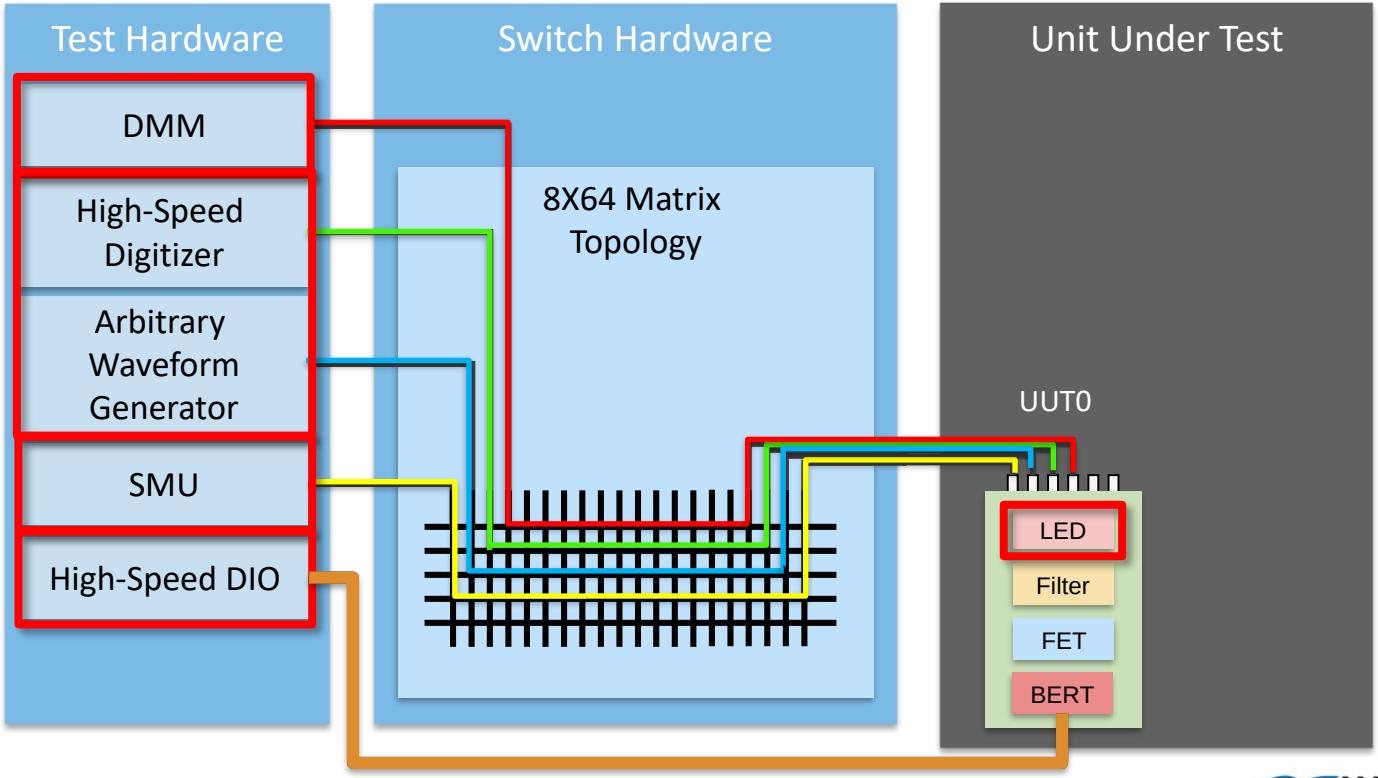
HSDIO Bit Error Rate Test (BERT)



- Generate a known stimulus and compare it to an expected response
  - Ideal for testing the logic of your UUT or the quality of your transmission lines



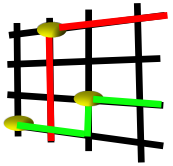
Switch System Block Diagram



Note: Connections are not actually single wire

Switch Executive

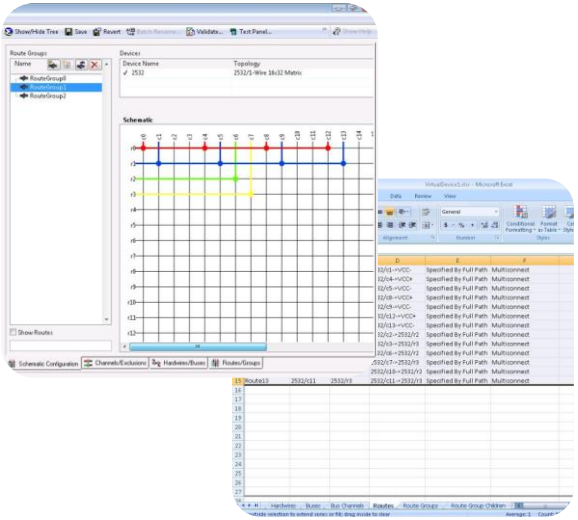
Intelligent Switch Management Software



- Easy channel aliasing
- Automatic routing
- Interactive debug panel
- Simplified ADE integration
- Integrated TestStand deployment

+

Graphical Configuration



Excel Integration

NATIONAL INSTRUMENTS

Instructor Demo

Switch Executive Demonstration

Objective

Time to complete: 10 minutes

Understand how to use Switch Executive to incorporate switching into a test system.

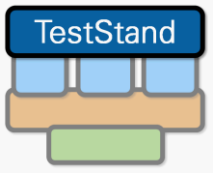
Exercise 1

Explore the Hands-On Hardware

Objective

Time to complete: 20 minutes

Configure the switch to connect to the UUT in Socket 0 and run the provided LabVIEW VIs to verify that the UUT and modular instruments respond as expected.

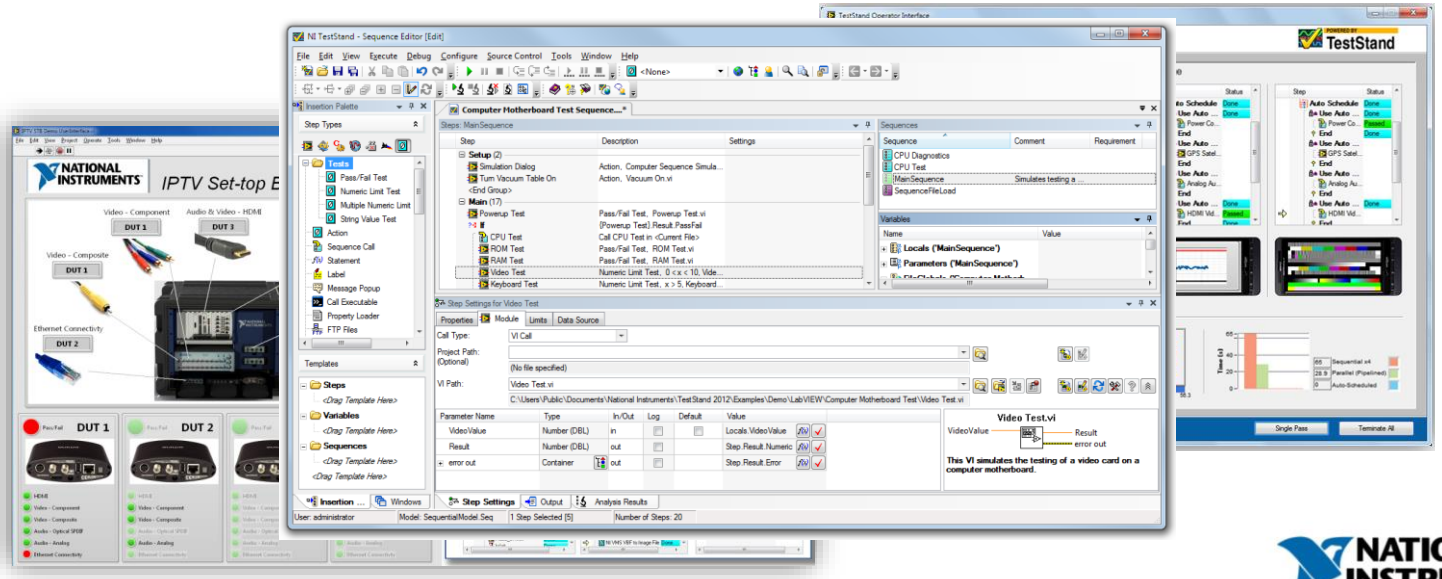


Automated Test with TestStand

NATIONAL INSTRUMENTS

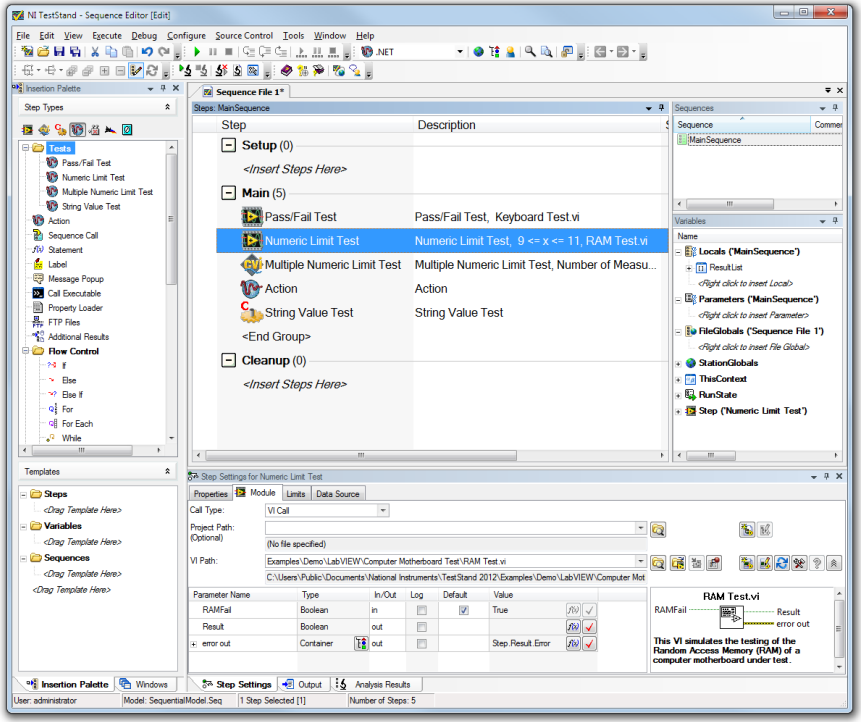
TestStand Components

- 2 Primary Stages of a Project
  - Development--Sequence Editor
  - Deployment--User Interface(s)

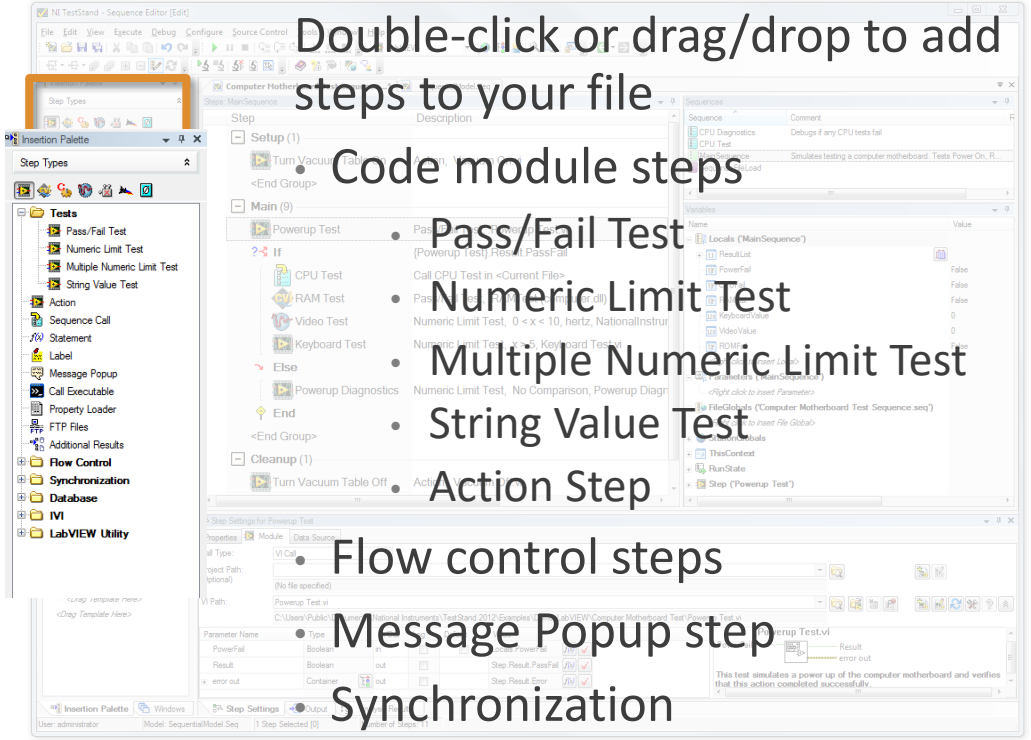


TestStand Sequence Editor

- Create
- Edit
- Manage
- Execute
- Debug
- View reports
- Deploy
- Create user profiles
- Customize



Sequence Editor–Insertion Palette

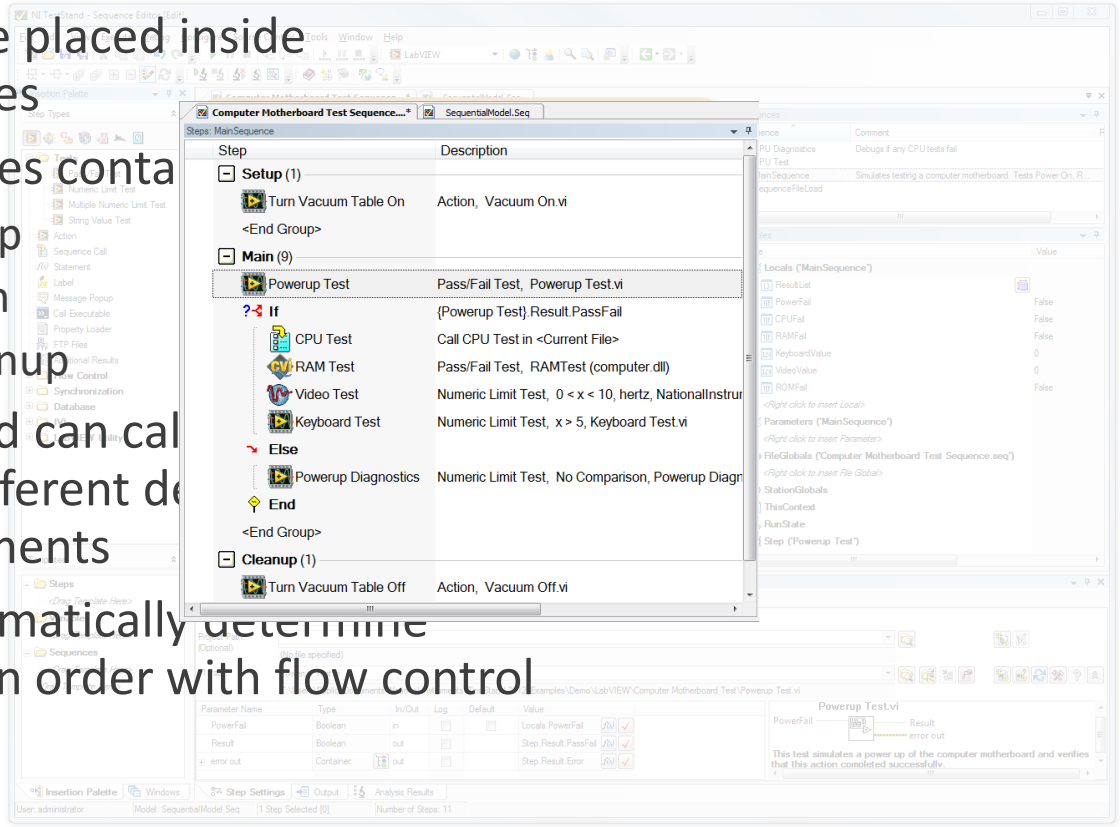


- Double-click or drag/drop to add steps to your file
- Code module steps
- Pass/Fail Test
- Numeric Limit Test
- Multiple Numeric Limit Test
- String Value Test
- Action Step
- Flow control steps
- Message Popup step
- Synchronization

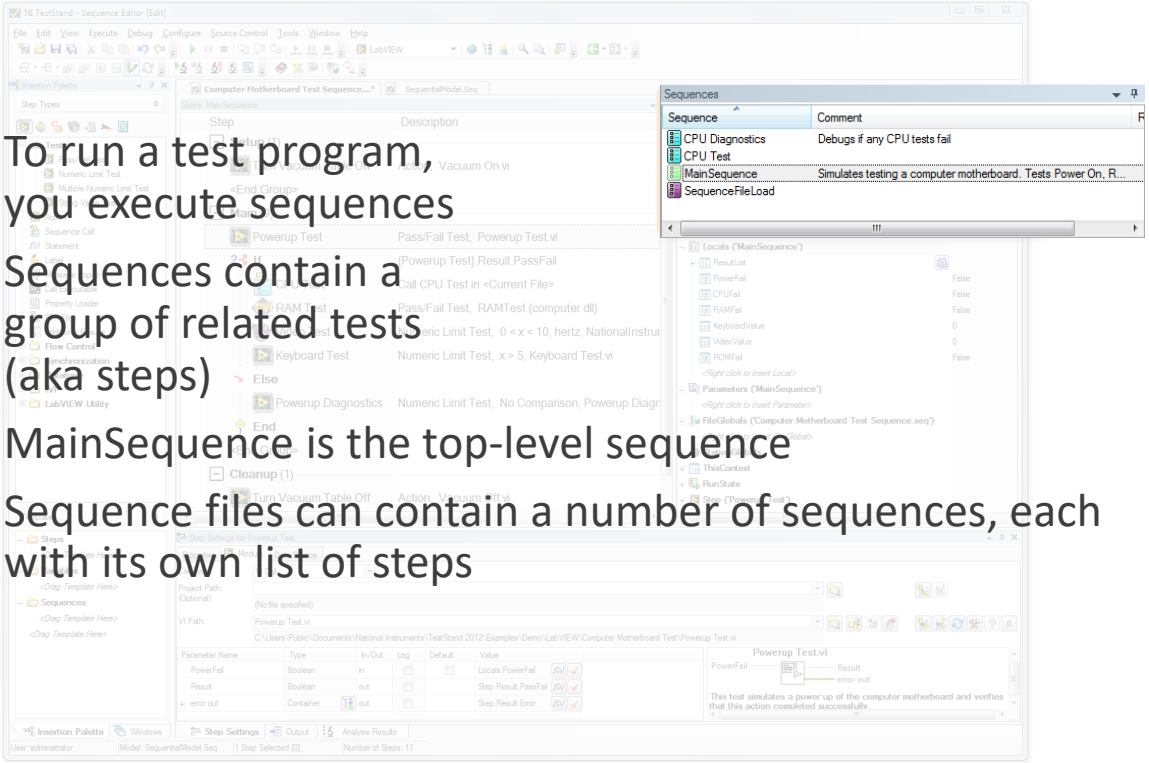
NATIONAL INSTRUMENTS

Sequence Editor–Steps Pane

- Steps are placed inside sequences
- Sequences conta
  - Setup
  - Main
  - Cleanup
- TestStand can cal
  many different de
  environments
- Programmatically determine execution order with flow control steps



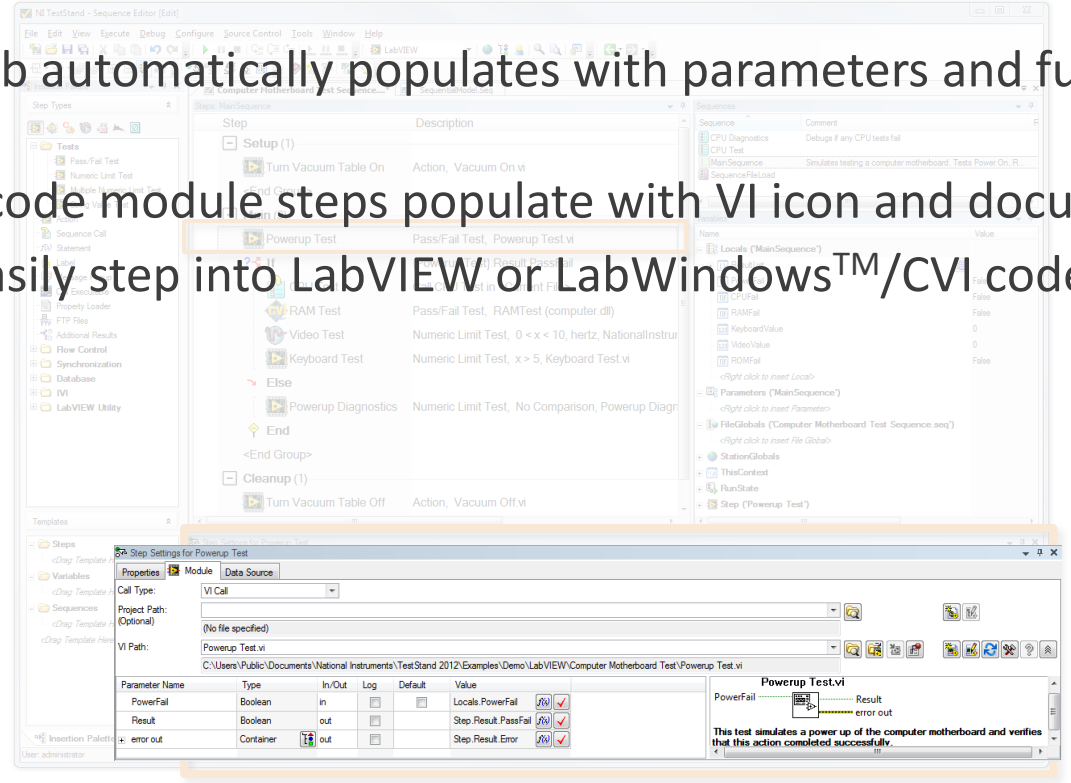
Sequence Editor--Sequences Pane



- To run a test program, you execute sequences
- Sequences contain a group of related tests (aka steps)
- MainSequence is the top-level sequence
- Sequence files can contain a number of sequences, each with its own list of steps

Sequence Editor--Step Settings--Module Tab

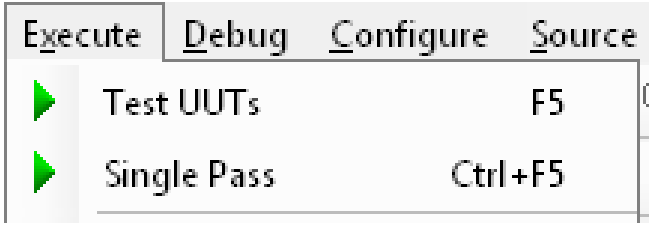
- Module tab automatically populates with parameters and functions of code module
- LabVIEW code module steps populate with VI icon and documentation
- You can easily step into LabVIEW or LabWindows™/CVI code with Edit button



NATIONAL INSTRUMENTS

Sequence Editor–Single Pass vs. Test UUTs

- Single Pass
  - Executes test sequence once
  - Often used during test development
- Test UUTs
  - Loops through test sequence for multiple UUTs
  - Often used for production as units roll off of assembly line
  - Shows serial number dialog



Instructor Demo

Introduction to the TestStand Environment

Objective

Time to complete: 15 minutes

Run simulated mobile device test sequence.

Exercise 2

Introduction to the TestStand Environment

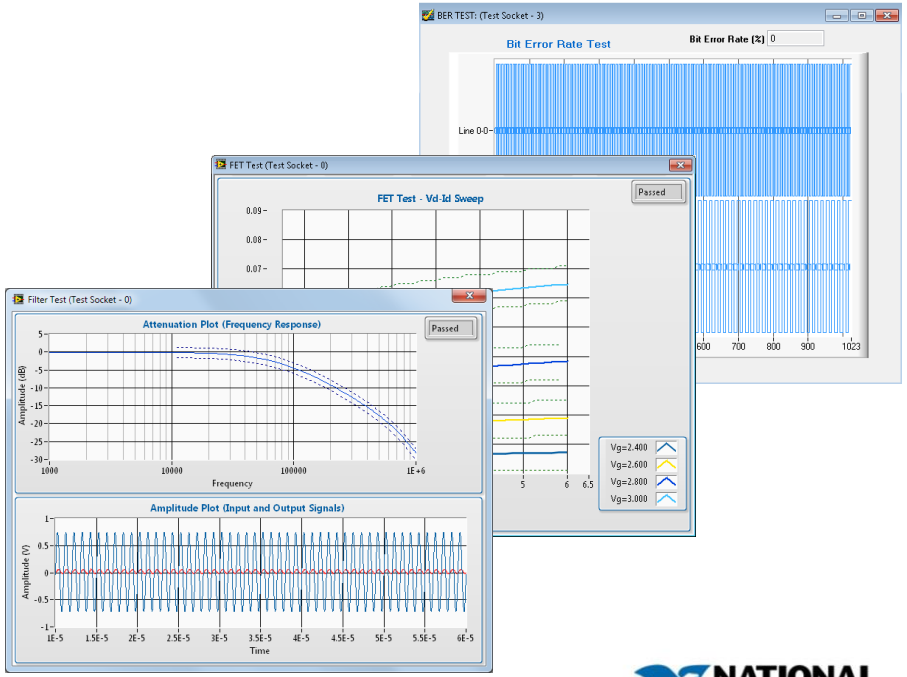
Objective

Time to complete: 15 minutes

Run simulated mobile device test sequence.

Hands-On Test System Creation

- Scenario: Tests are provided by your engineering team
  - Filter Test
  - FET Test
  - BER Test
  - LED Tests
- Easily integrated into TestStand



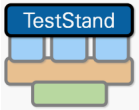
Exercise 3

Adding Tests Using the TestStand Sequence Editor

Objective

Time to complete: 25 minutes

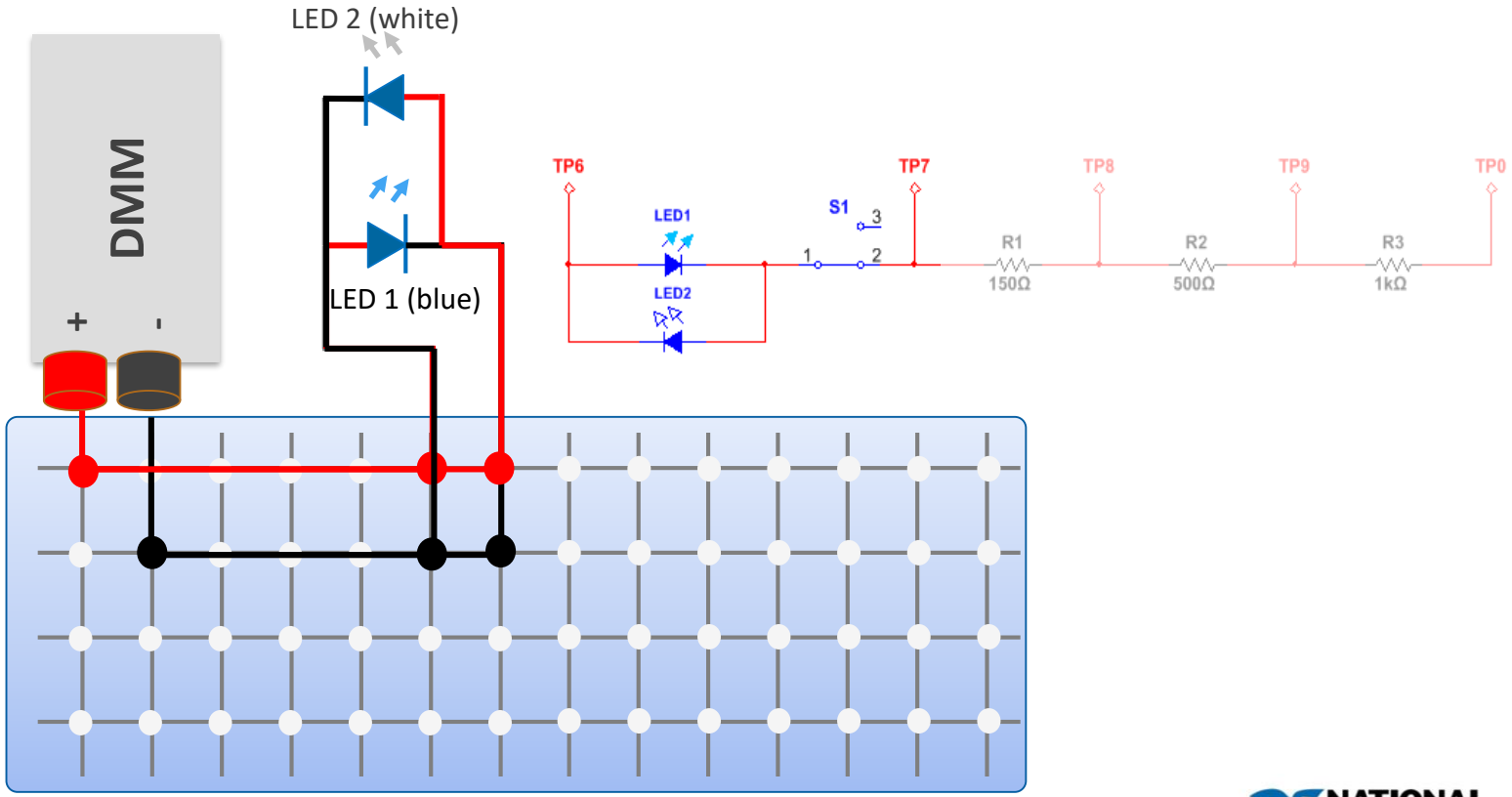
Create steps to test the filter, LED, FET and transmission lines on one UUT.



Scaling up Production with Parallel Testing

NATIONAL INSTRUMENTS

Increasing Hardware Utilization with Switching

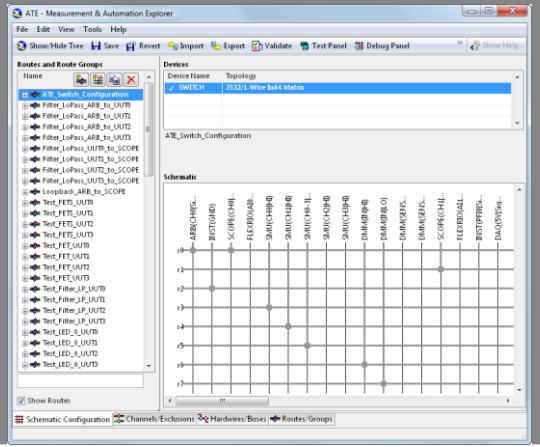


PXI-2532 8x64 Matrix

Switch Automation with TestStand

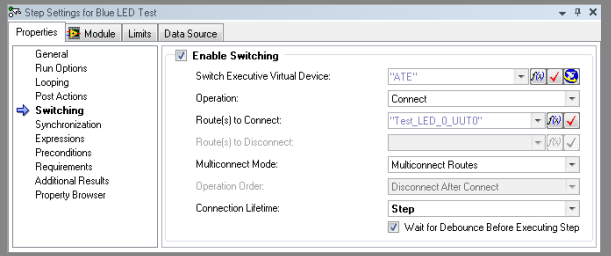
Step 1

Create or import switch system configuration file into NI Switch Executive



Step 2

Configure the Switching property in NI TestStand to call preconfigured routes and route groups



Exercise 4

Integrating Switching into the Sequence using Switch Executive

Objective

Time to complete: 10 minutes

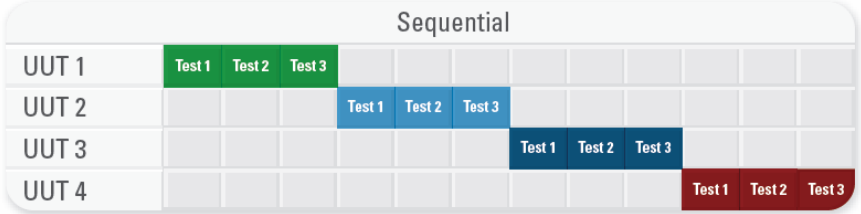
Create a new step for testing the second LED and add switching operations to the test steps for the two LEDs, filter, and FET on one UUT.

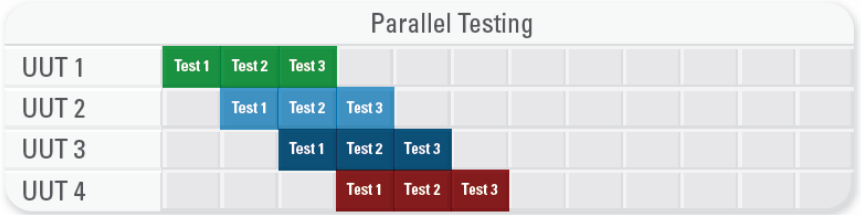
Sequential vs. Parallel vs. Auto-scheduled Execution

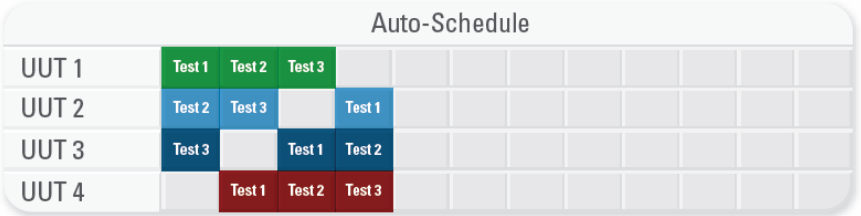
- Simple 1 UUT test fixture
- Extremely simple code structure
- 12 time blocks for 4 units

- Multiple UUTs in parallel
- Switching allows hardware to be shared
- ~30–60% decrease in testing time

- Same HW setup as above
- TestStand auto-schedules resources
- ~15–20% further decrease in testing time

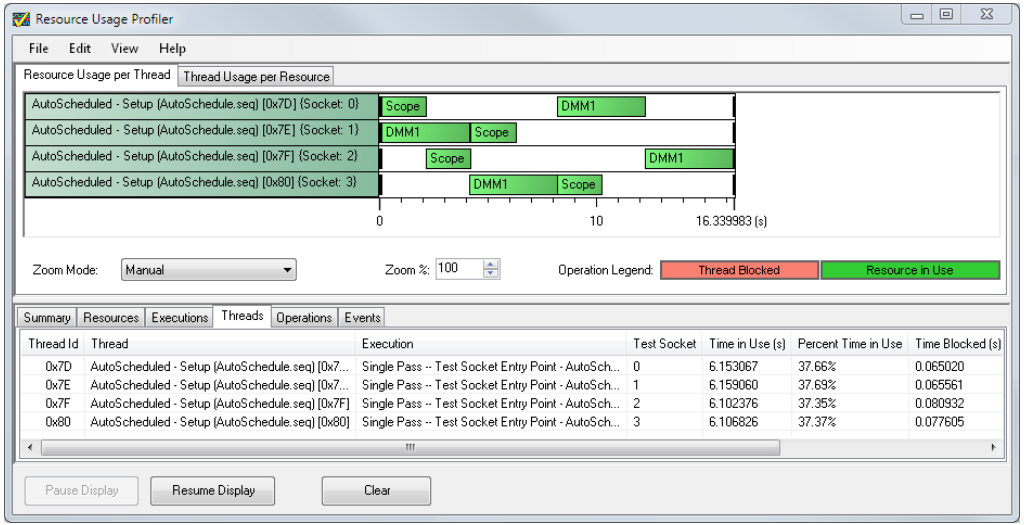






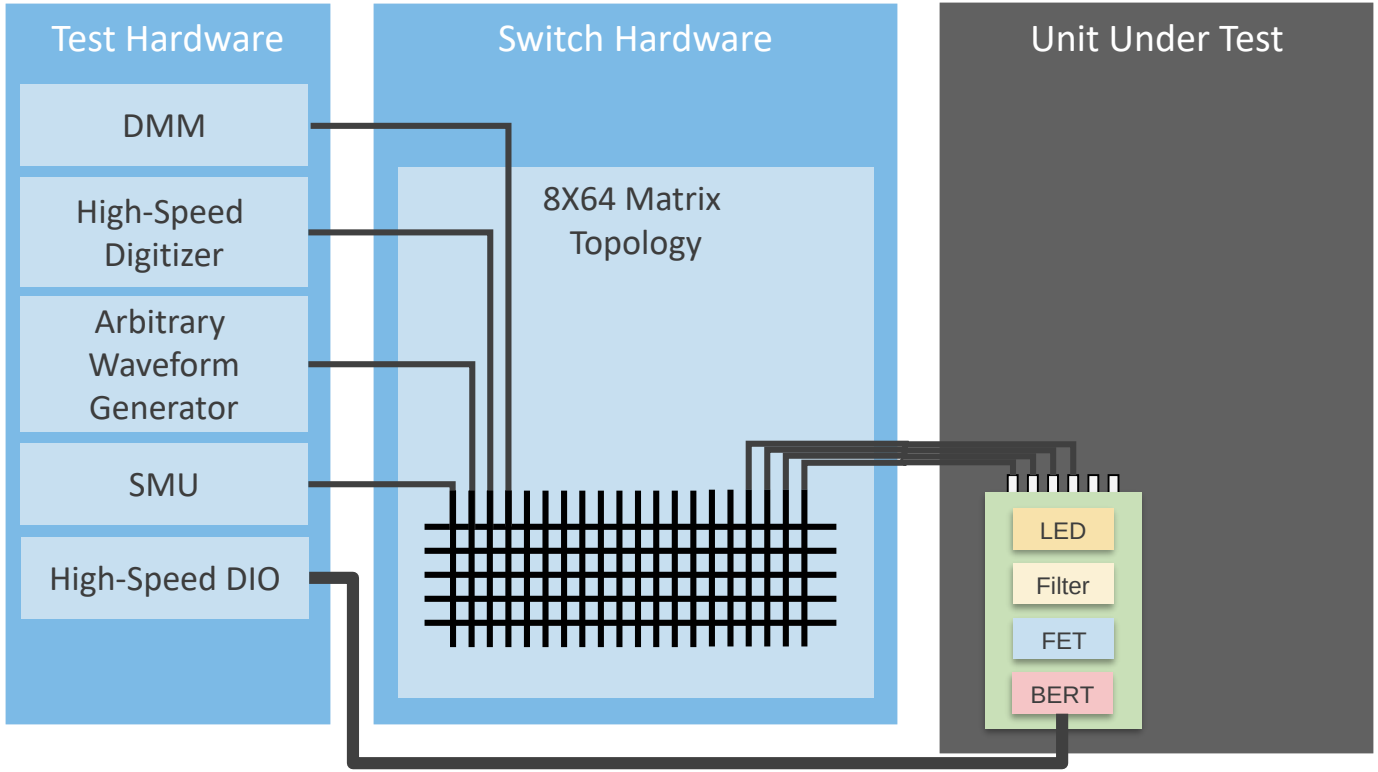
Execution Profiler

- Provides immediate visualization and performance statistics for all current executions, threads, and resources
- Used to optimize test sequences by profiling step execution time
- A fully optimized test should take 4X the longest test time

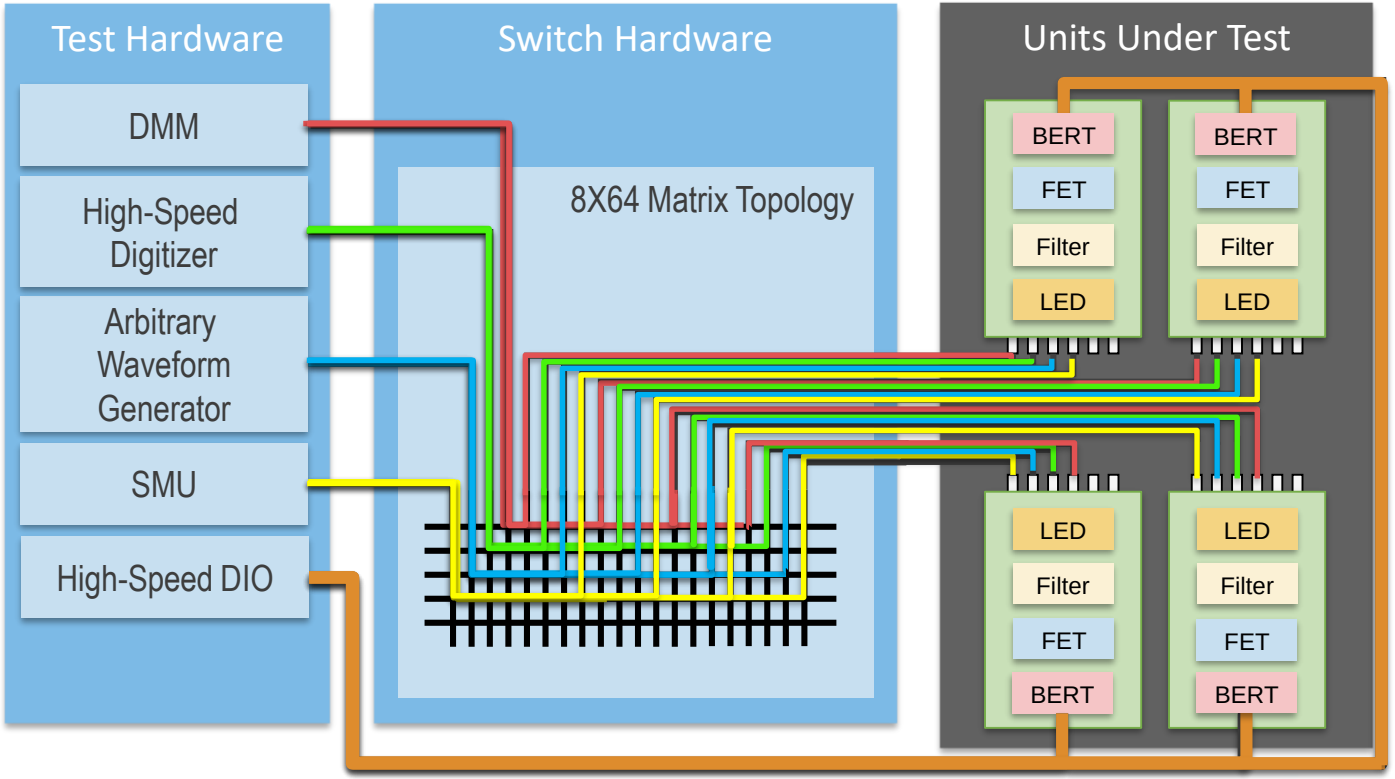


Auto-Scheduled Execution

Switch System Block Diagram



Switch System Block Diagram



Exercise 5

Performing Auto-Scheduled Parallel Testing

Objective

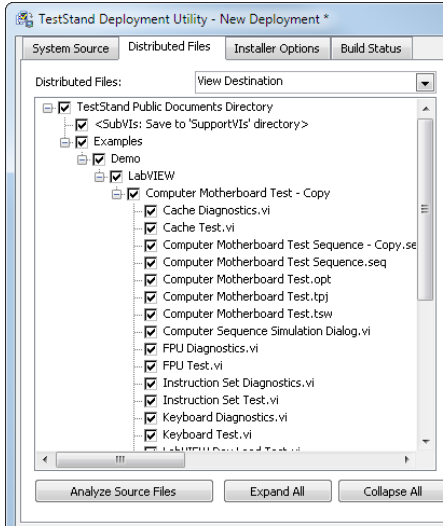
Time to complete: 15 minutes

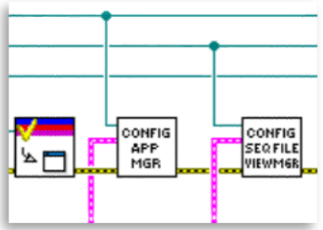
Utilize the TestStand Auto-Scheduling steps to scale up the testing from 1 UUT to all 4 UUTs in parallel.

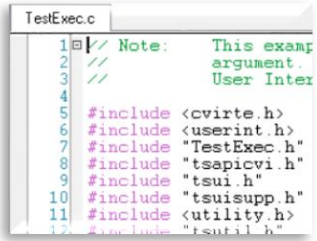
Test System Deployment

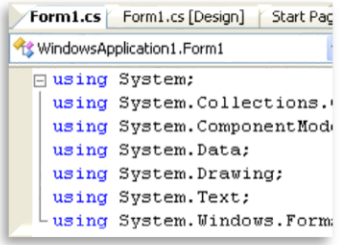
TestStand Deployment

- TestStand Deployment Utility
  - Analyzes test sequences and creates distributable installer
- Full-Featured and Simple UIs written in five development languages for easy customization
  - NI LabVIEW
  - NI LabWindows™/CVI
  - C#.NET
  - VB.NET
  - C++







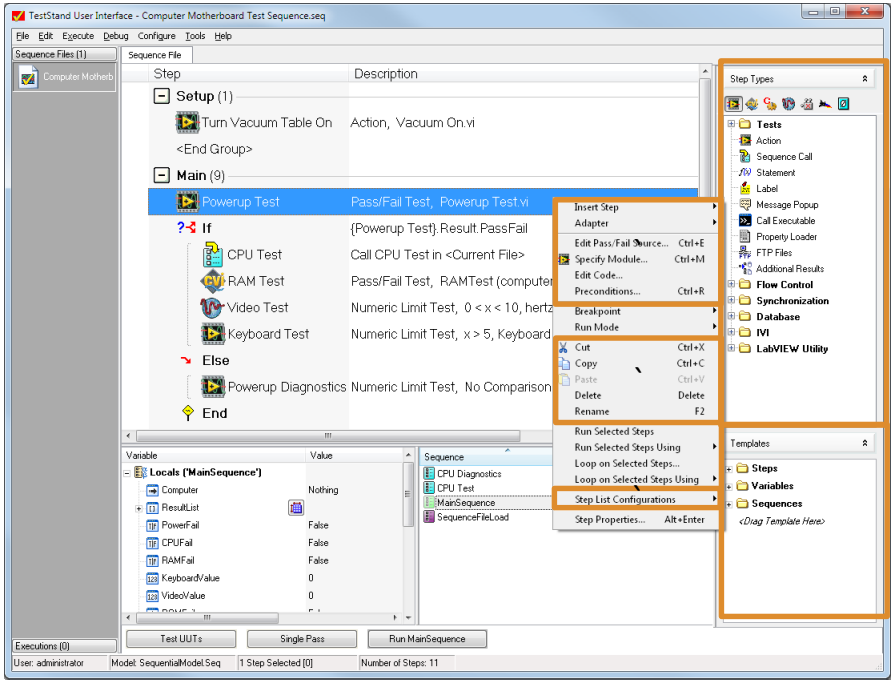


NATIONAL INSTRUMENTS

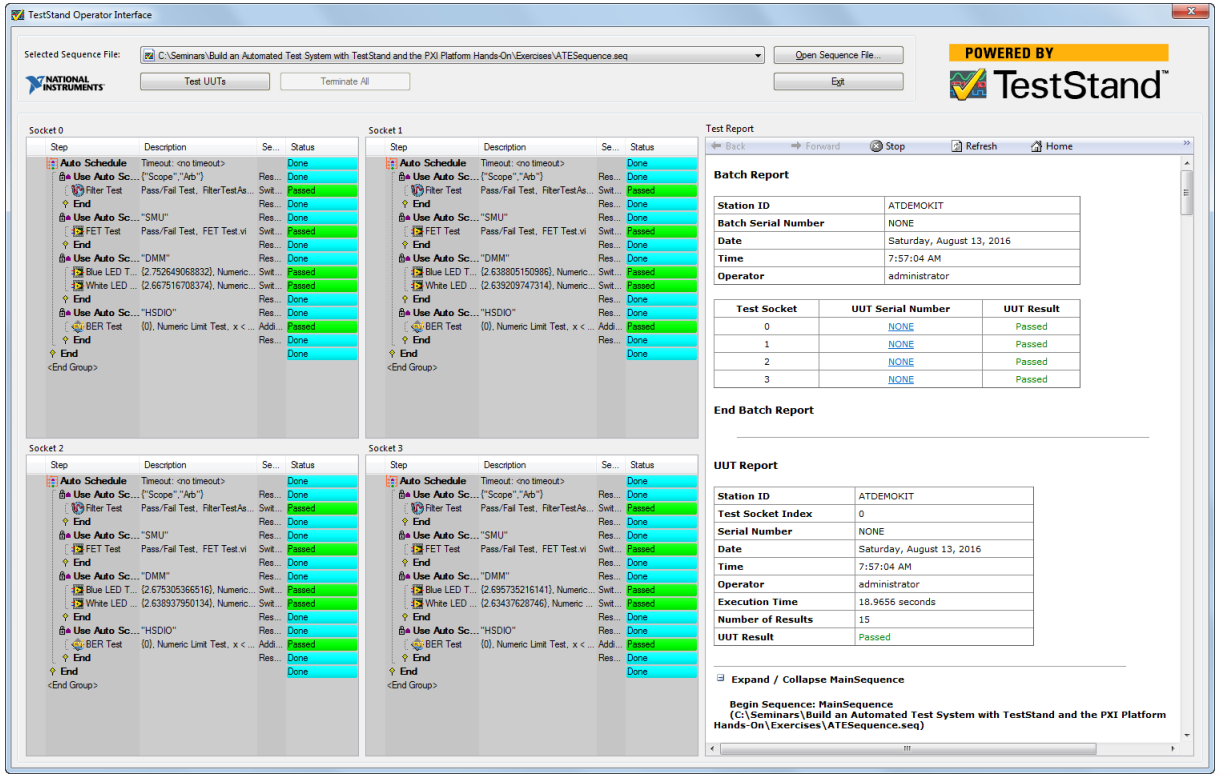
Provided TestStand User Interfaces

Hotkey:
Ctrl+Alt+
Shift+Insert

Default Full UI—Operator Mode



Our Customized LabVIEW User Interface



Instructor Demo

Execute Project with Customized TestStand User Interface

Objective

Time to complete: 5 minutes

Understand the use of potential customizations to TestStand User Interfaces.

Exercise 6

Execute Project with Customized TestStand User Interface

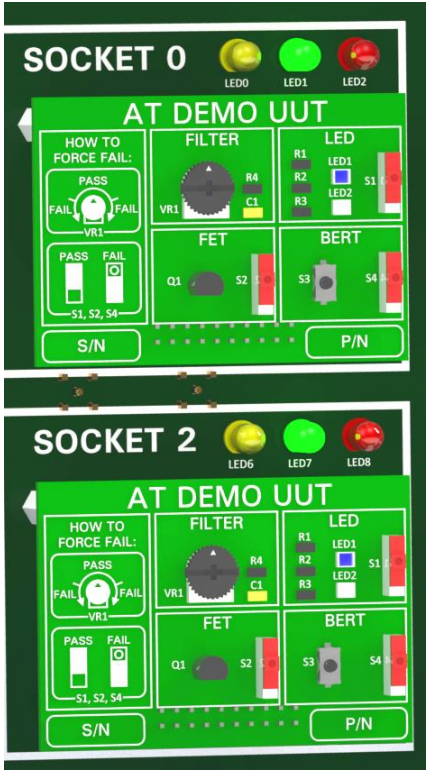
Objective

Time to complete: 5 minutes

Understand the use of potential customizations to TestStand User Interfaces.

New Test Requirement (optional)

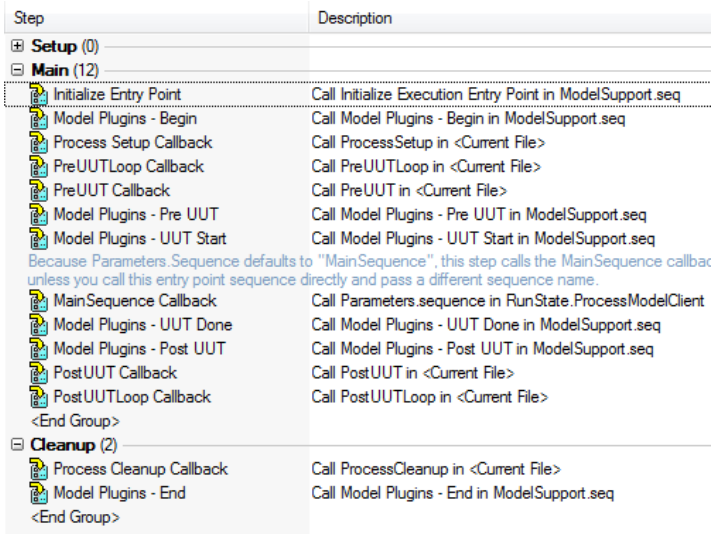
- There are 3 LEDs on each test socket (yellow, green, and red)
- Use provided API in TestStand to show the operator whether the UUT is testing (yellow), has passed (green), or has failed (red)
- Why shouldn't we add this code directly to MainSequence?
  - This adds clutter and hides purpose of test code
  - Socket LED control and UUT LED testing could be easily confused by operators



NATIONAL INSTRUMENTS

TestStand Execution Architecture–Process Model

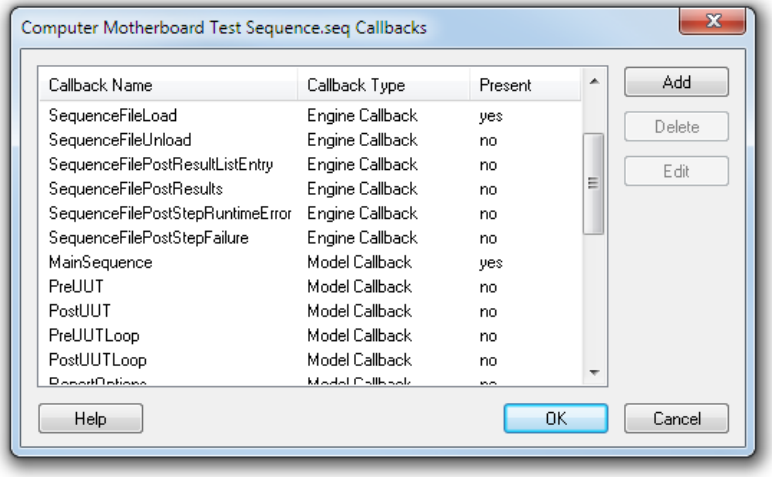
- Sequence file that contains all code not specific to one type of UUT
  - Reporting
  - Database logging
  - Serial number scanning
  - Reporting information to a custom user interface
- Default process model is SequentialModel.seq
- Whole companies or groups often standardize on a customized process model



SequentialModel.seq

Solution for LEDs–Callbacks

- Sequences that a client file can override to modify behavior at a particular time
- Engine callbacks
  - Predefined by the TestStand Engine
  - Ex: SequenceFileLoad to launch an FYI dialog describing the purpose of a sequence file
- Process model callbacks
  - Sequences marked as a “callback” in process model can be overridden by sequence files using that process model
  - Ex: PreUUT can be overridden to replace serial number scanning from process model with intelligent camera-based vision recognition system



Exercise 7

Overriding Callbacks

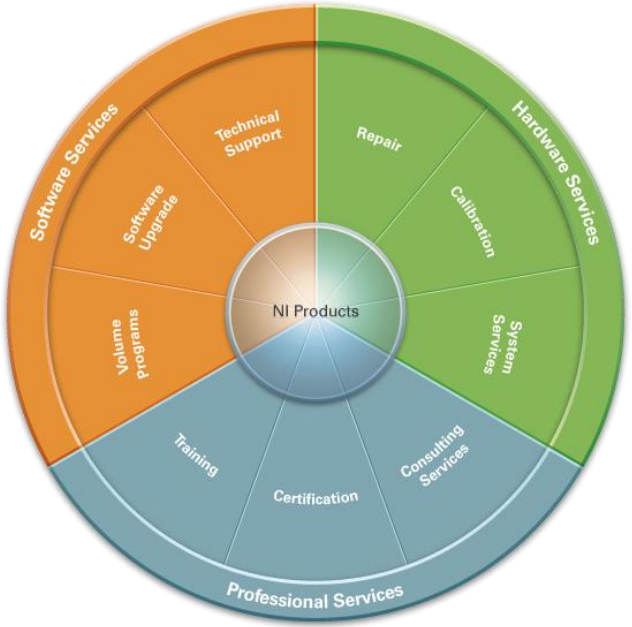
Objective

Time to complete: 15 minutes

Override the PreUUT, PostUUT, and PostUUTLoop callbacks to change the status LEDs for each UUT to yellow, green, and red, depending on whether the unit is testing, passed, or failed.

National Instruments Services and Support

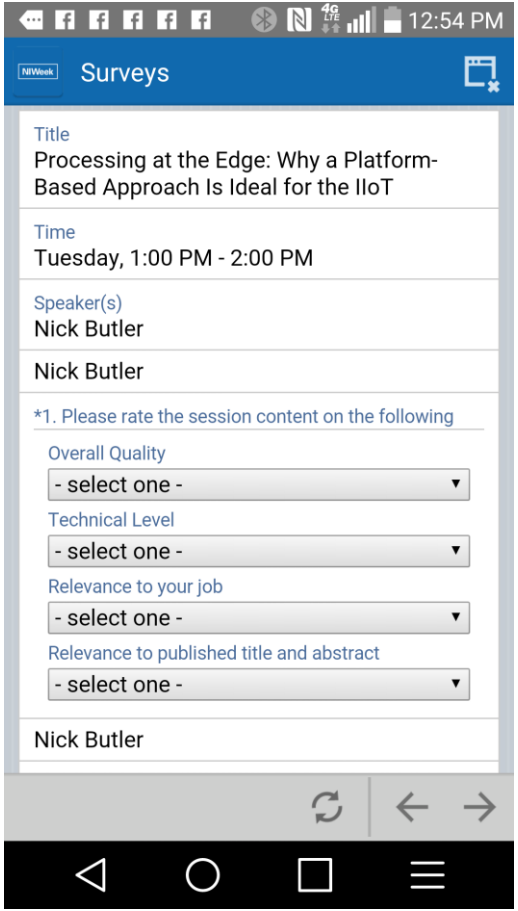
- Technical Support
  - Web support resources
  - Application engineers worldwide
  - Premier support
- Instructor Led Training
  - Developing Test Programs Using TestStand
  - Architecting Test Systems Using TestStand
  - Thousands of engineers trained each year
- Certification
  - Certified TestStand Developer (CTD)
  - Certified TestStand Architect (CTA)



ni.com/services

Visit ni.com/automatedtest

- Access test development resources
  - System design templates
  - Reference guides
  - Optimization strategies
- Read case studies
  - Explore business and technology impact
- Learn about the products
  - PXI, LabVIEW, TestStand, and more
- Configure a system or get a quote



Before you go, take the survey.

NATIONAL INSTRUMENTS

Stay Connected During and After NIWeek

ni.com/niweekcommunity

facebook.com/NationalInstruments

twitter.com/niglobal

youtube.com/nationalinstruments

Please provide feedback on this session via the NIWeek Mobile App