
NATIONAL
INSTRUMENTS™
MathWorks®

Best Practices for Integrating NI Tools & MathWorks Software

Tarek Safwan

Senior Field Marketing Engineer, National Instruments

NI and MathWorks are pleased to announce a collaboration between the two companies to provide our mutual customers with an experience that boosts their productivity.

May 2019

What This Means

- We see our tools as complimentary
- More openness in both tool chains for integration with the other
- Better technical support for users

Seamless Integration Promotes Collaboration

- Shared goal of ensuring seamless integration between NI and MathWorks toolchains
- You can use the right tool for each stage in your workflow

For example

- In MATLAB, you can use many kinds of NI hardware directly
- In LabVIEW NXG and LabVIEW 20xx you can use code written in MATLAB

Agenda

- Automated Measurements & Data Analysis
- Hardware in the Loop Test & Rapid Control Prototyping
- Wireless Communications

Automated Measurements & Data Analysis

MathWorks & NI Software Interfaces: MATLAB & LabVIEW

- LabVIEW 2019:
  - **MATLAB Script node**
  - System Exec VI
  - Call Library Function node (w/ MATLAB® Compiler SDK™ or MATLAB® Coder™)

- LabVIEW NXG:
  - Shared Library Interface (w/ MATLAB Compiler SDK or MATLAB Coder)
  - **Interface for MATLAB**

MATLAB Script Node in LabVIEW 20xx

- MATLAB® and LabVIEW must both be installed
- Full access to MATLAB built-in functions, objects, and data types
- Calls the MATLAB software to execute scripts or functions (IDE to IDE communication)

Tektronix®

Example customer application using MATLAB Script Node in LabVIEW

Tektronix Application Note - Automating Bit Error Rate Measurements of Complex Modulated Optical Signals

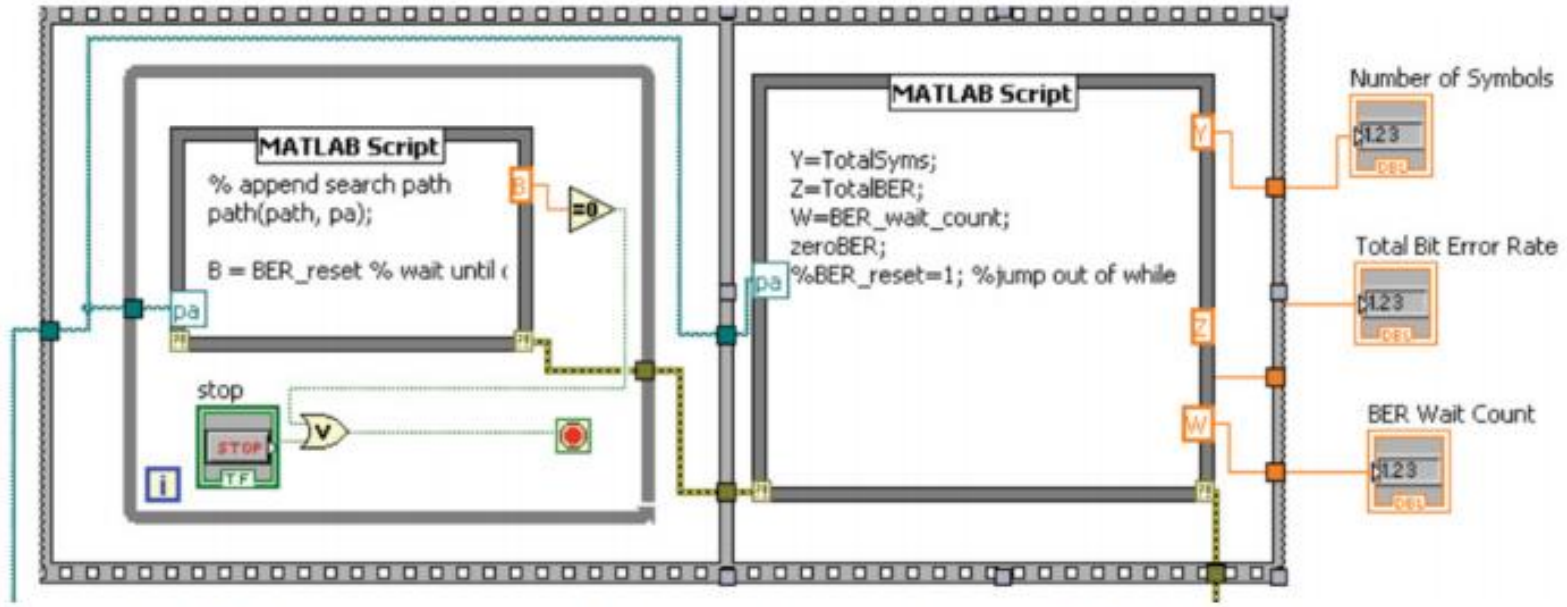
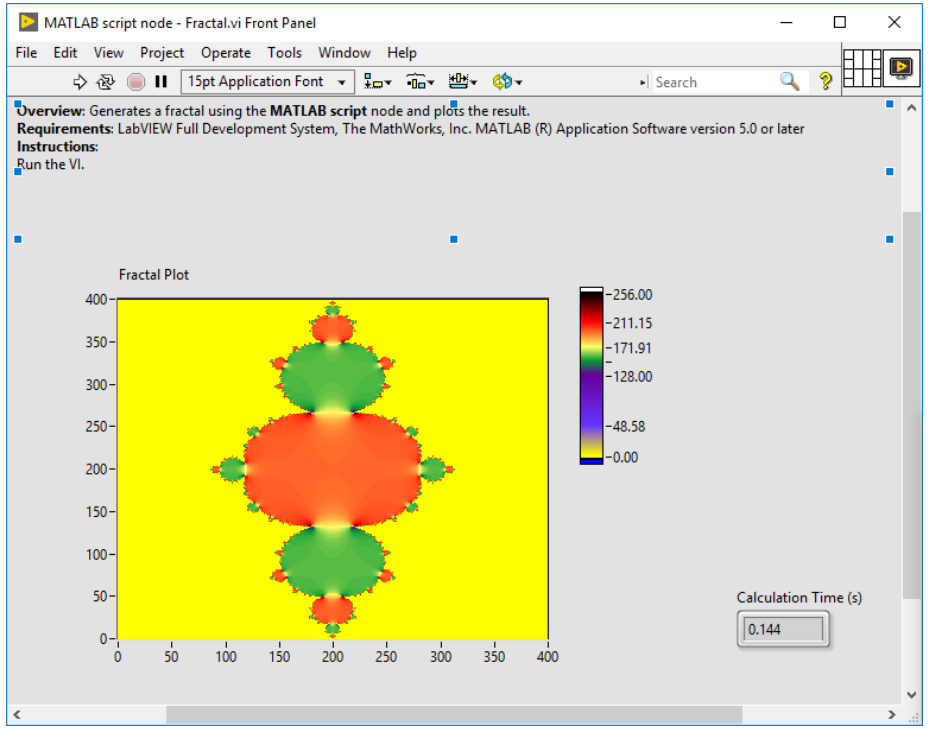
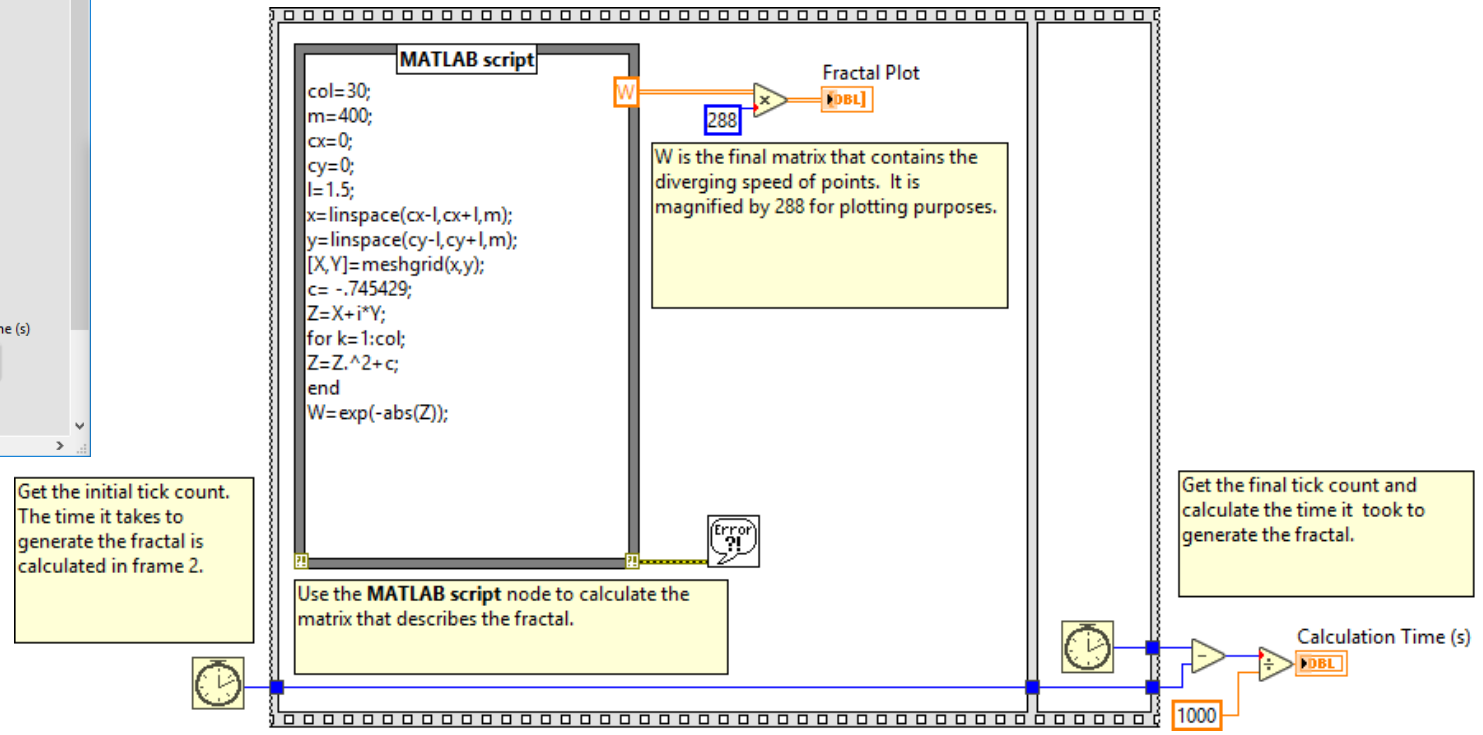


Figure 5. Details of LabVIEW Matlab Script Node to acquire BER information from Matlab. The left-hand frame waits until BER_reset is set to 0 (in the OM4106D engine file, illustrated above); it then passes control to the right-hand frame, which reads the variables and resets the BER_reset flag via the zeroBER.m command.

NATIONAL INSTRUMENTS™

Demonstration

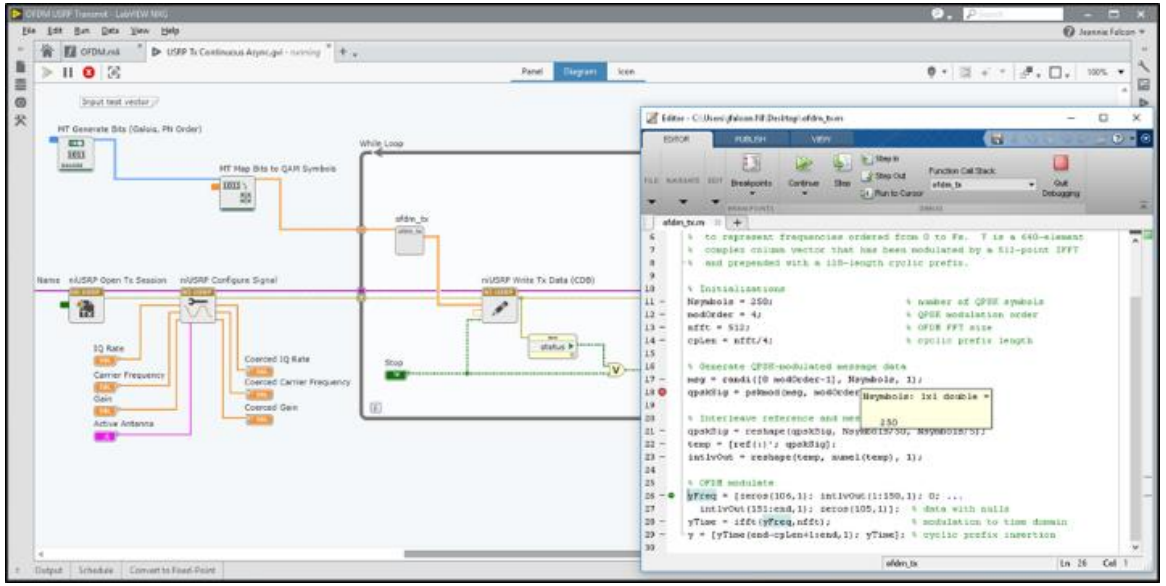




NATIONAL INSTRUMENTS

Interface for MATLAB ® in LabVIEW NXG 3.0

- Import .m files with the Interface for MATLAB ® document and configure inputs/outputs
- Drop it in your diagram as a subVI node like any other function call
- Debug and modify the MATLAB code while running the VI



NATIONAL INSTRUMENTS

Capabilities of Different Software Interfaces

Options	LabVIEW Versions	Capabilities
Interface for MATLAB®	NXG	MATLAB® communication, can pass multiple data types
MATLAB Script node	20xx	MATLAB communication, can pass multiple data types
Call Library Function Node	20xx	Works with your DLL and can run on real-time targets
Shared Library Interface	NXG	Works with your DLL
Model Interface Toolkit	20xx	Works with your model DLL, can have MATLAB Function Block in your Simulink model
VeriStand	-	Configure real-time test applications with your Simulink and Simscape™ models

Interfacing MathWorks Products to NI Hardware

Instrument Control Toolbox™
for MATLAB & Simulink

Image Acquisition Toolbox™
for MATLAB & Simulink

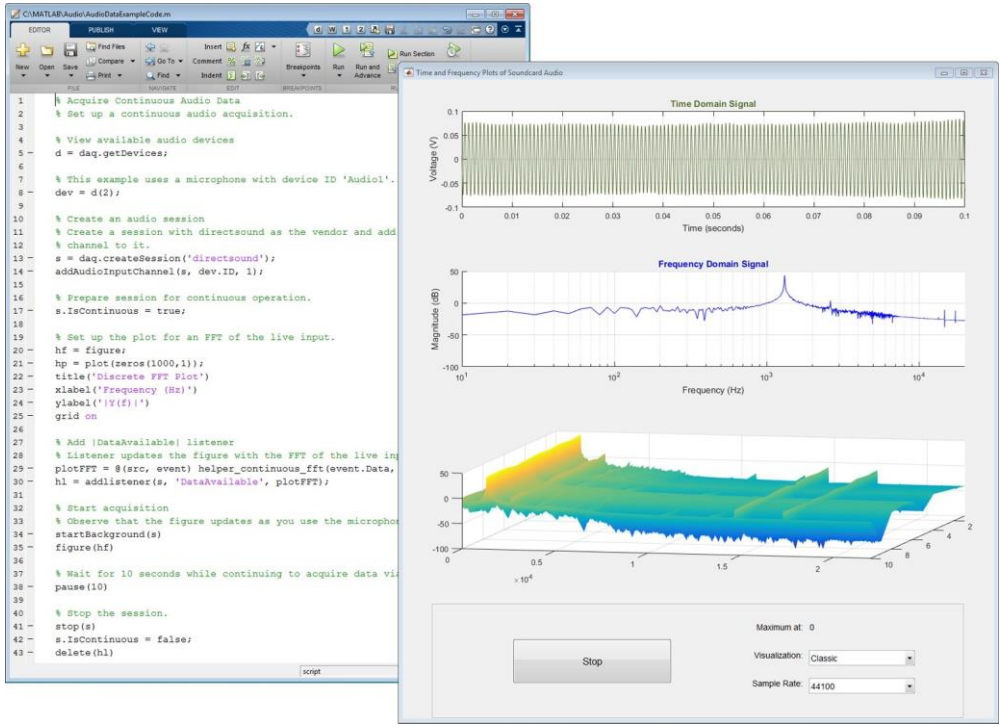
Data Acquisition Toolbox™
for MATLAB & Simulink
Hardware includes:
NI, Measurement Computing, & Digilent™

Plus interfaces to NI-XNET, Ettus Research™ & BEEcube™ hardware

NATIONAL INSTRUMENTS

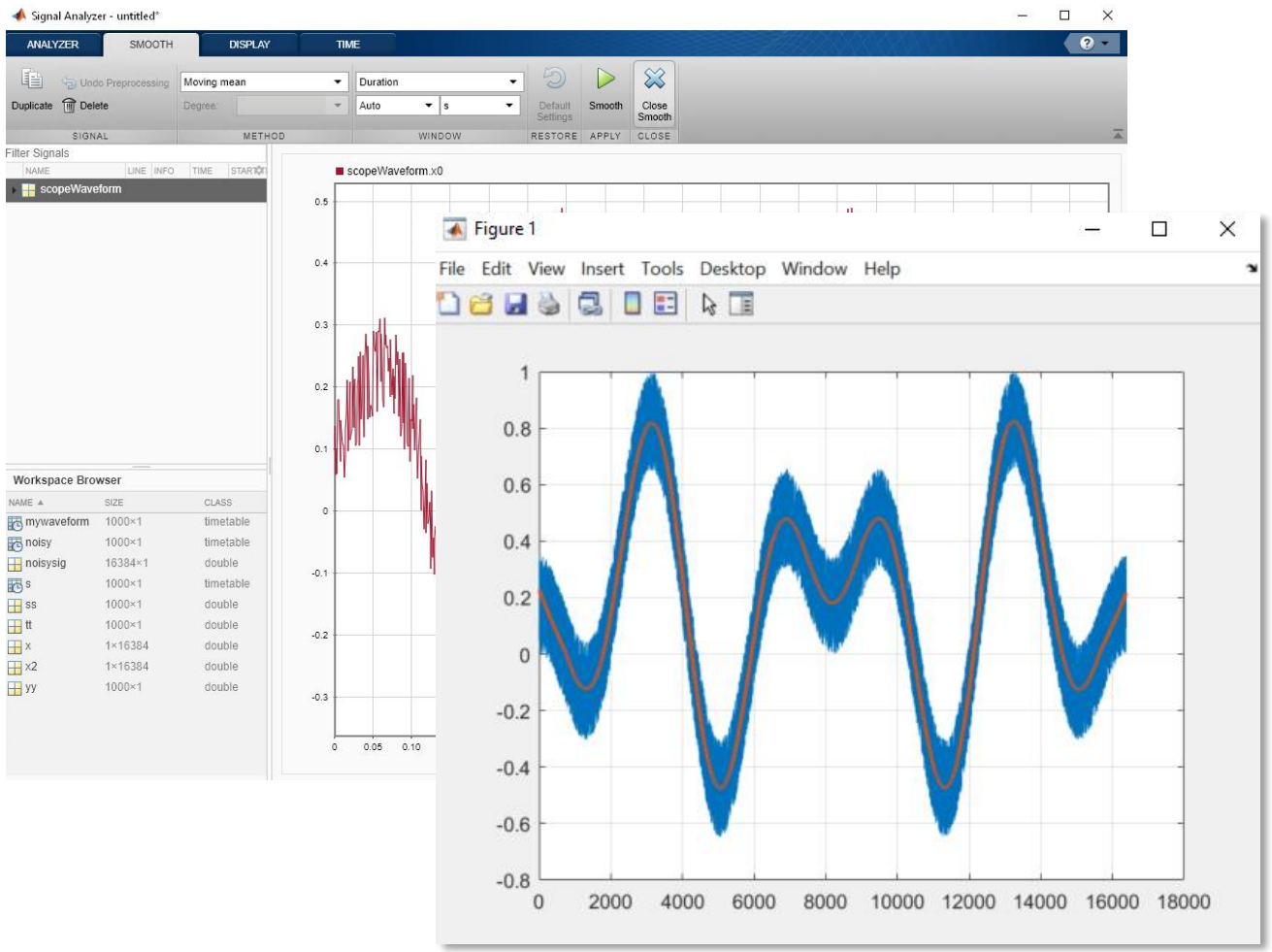
Data Acquisition Toolbox

- **Connect to data acquisition cards, devices, and modules**
- **Software** - MATLAB and Simulink with NI-DAQmx
- **Hardware buses** - USB, PCI, PCI-Express®, PXI, and PXI-Express devices
- Support for analog input, analog output, counters, timers, and digital I/O
- Live acquisition of measured data
- Hardware and software triggers for control of data acquisition
- Device-independent software interface



Instrument Control Toolbox

- **Control and communicate with test and measurement instruments**
- **Software:** MATLAB & Simulink
- Test & Measurement Tool
- IVI and VXI*plug&play*
- NI-VISA commands
- **Modular instruments drivers:**
  - NI-DCPower
  - NI-DMM
  - NI-FGEN
  - NI-SCOPE
  - NI-SWITCH



NATIONAL INSTRUMENTS

Demonstration

- Thunderbolt connection from laptop to PXI
- Hardware support packages: NI-FGEN and NI-SCOPE
- NI MAX
- MATLAB
  - Hardware discovery for FGEN and SCOPE
  - Waveform generation in MATLAB
  - Soft front panel for NI SCOPE

NATIONAL INSTRUMENTS

MATLAB R2019a

HOME | PLOTS | APPS | EDITOR | PUBLISH | VIEW

Search Documentation | Eric

New | Open | Save | Find Files | Compare | Print | Go To | Find | Insert | Comment | Indent | Breakpoints | Run | Run and Advance | Run Section | Advance | Run and Time

FILE | NAVIGATE | EDIT | BREAKPOINTS | RUN

\\fs-57-ah\vmgr$ ▸ home07 ▸ ewetjen ▸ Documents ▸ MATLAB ▸ Examples ▸ R2019a ▸ deeplearning_shared ▸ DeepLearningSpeechRecognitionExample

Current Folder

Name

- weightedClassificationLayer.m
- test1.m
- streaming_commands.png
- splitData.m
- speechSpectrograms.m
- simpleacq_ICP.m
- simpleacq.m
- oscilloscopeApp.mlapp
- noisysine.m
- NIWeek_Demo.m
- multifreq.m
- GenerateSignal.m
- fgen_test.m
- fgen_mdd.m
- detectspeech_stack
- detectspeech_daq_mic.m
- detectspeech3m.m

oscilloscopeApp.mlapp (App)

No details available

Editor - \\fs-57-ah\vmgr$\home07\ewetjen\Documents\MATLAB\Examples\R2019a\deeplearning_shared\DeepLearningSpeechRecogniti...

+23 | GenerateSignal.m | cleansig.m | Oscilloscope.m | StateNotConnected.m | InstrumentAdaptorFactory.m

```
7 -  connect(ictObj);
8
9    %% Create a noisy signal with 3kHz and 5kHz components
10 - x = linspace(-4*pi, 4*pi,16384);
11 - x2= linspace(-1*pi, 1*pi,16384);
12 - noisysig = (sin(x).*sin(x2)) + 0.5*rand(size(x));
13
14   % x = linspace(-2*pi, 2*pi,16384);
15   % noisysig = sin(x) + 0.3*rand(size(x));
16   % normalize the waveform so values are between -1 to + 1
17 - if  max ( abs (noisysig )) ~= 0
18 -      noisysig =  ( noisysig./ max (noisysig ))';
19 - end
20   %% Configure FGEN for output of custom waveform
21
22   % These values are defined in the driver's header file 'niFgen.h'
23 - NIFGEN_VAL_OUTPUT_FUNC = 0;
```

Command Window

New to MATLAB? See resources for Getting Started.

fx >>

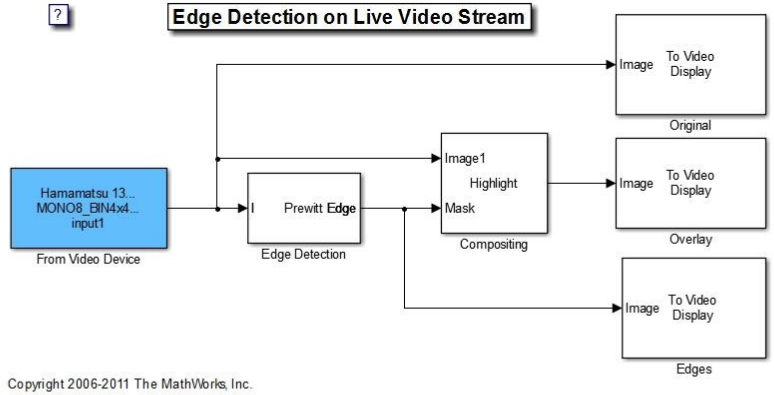
Workspace

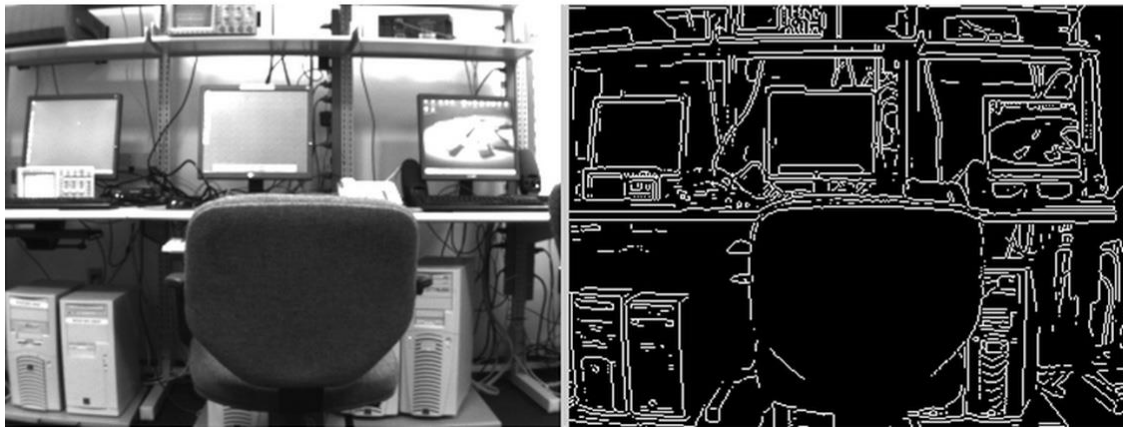
Name	Value
ChannelNa...	'0'
ictObj	1x1 icdevice
mywavefo...	1000x1 timet...
NIFGEN_A...	0.5000
NIFGEN_A...	0
NIFGEN_A...	1000
NIFGEN_A...	0
NIFGEN_V...	0
NIFGEN_V...	102
noisysig	16384x1 dou...
resourceID	'DAQ::Fgen'
x	1x16384 dou...
x2	1x16384 dou...

script | Ln 12 Col 49

Image Acquisition Toolbox

- **Acquire images and video from industry-standard hardware**
- **Software:** MATLAB and Simulink, Image Processing Toolbox™, NI-IMAQ™
- **Hardware:** Camera Link frame grabbers for PCI, PXI, and PXIe
- Hardware triggering and synchronization for multiple devices
- Image Acquisition app for rapid configuration, acquisition, and live previewing

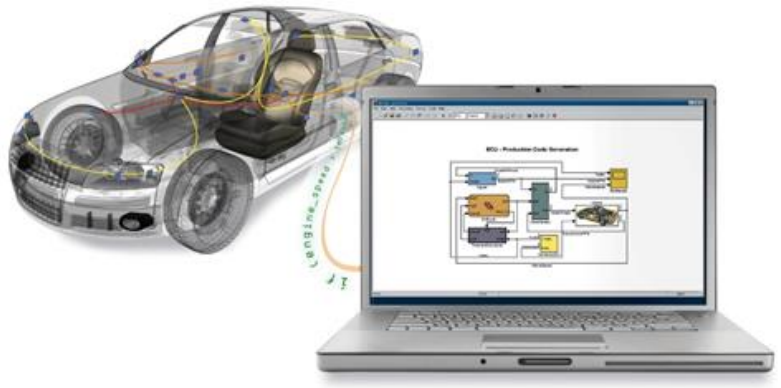




NATIONAL INSTRUMENTS

Vehicle Network Toolbox

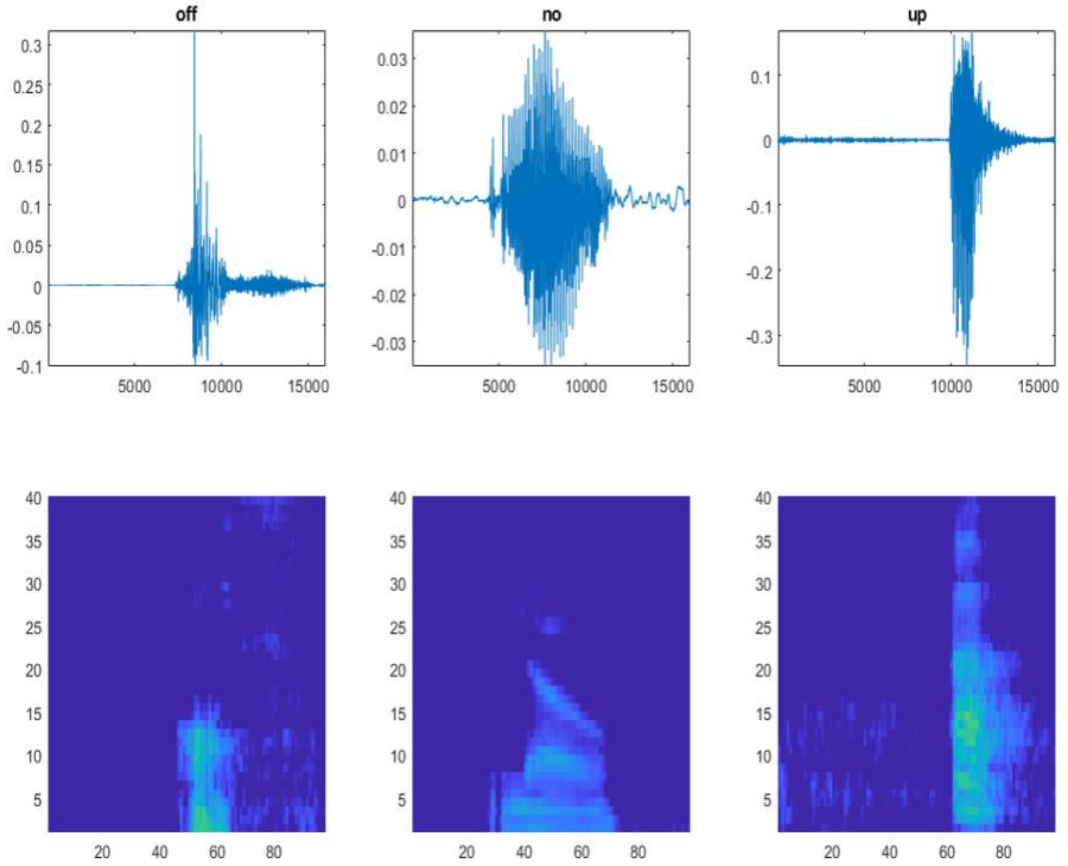
- **Communicate with in-vehicle networks using CAN, CAN FD, J1939, and XCP protocols**
- **Software:** MATLAB and Simulink, Vehicle Network Toolbox™, NI-XNET™
- **Hardware buses** - USB, PCI, and PXI devices
- Model Based Development and testing of embedded systems
- Log and record messages for later analysis and replay
- Simulate message traffic on a virtual CAN bus or connect to a live network or ECU
- Work with industry standard files like A2L, MDF, DBC and CDF files



NATIONAL INSTRUMENTS

Audio Recognition

- Acquire data from microphone
- Classify speech using pre-trained neural network
- Use mel spectrogram as feature for classification
- Predict command spoken based on a probability threshold



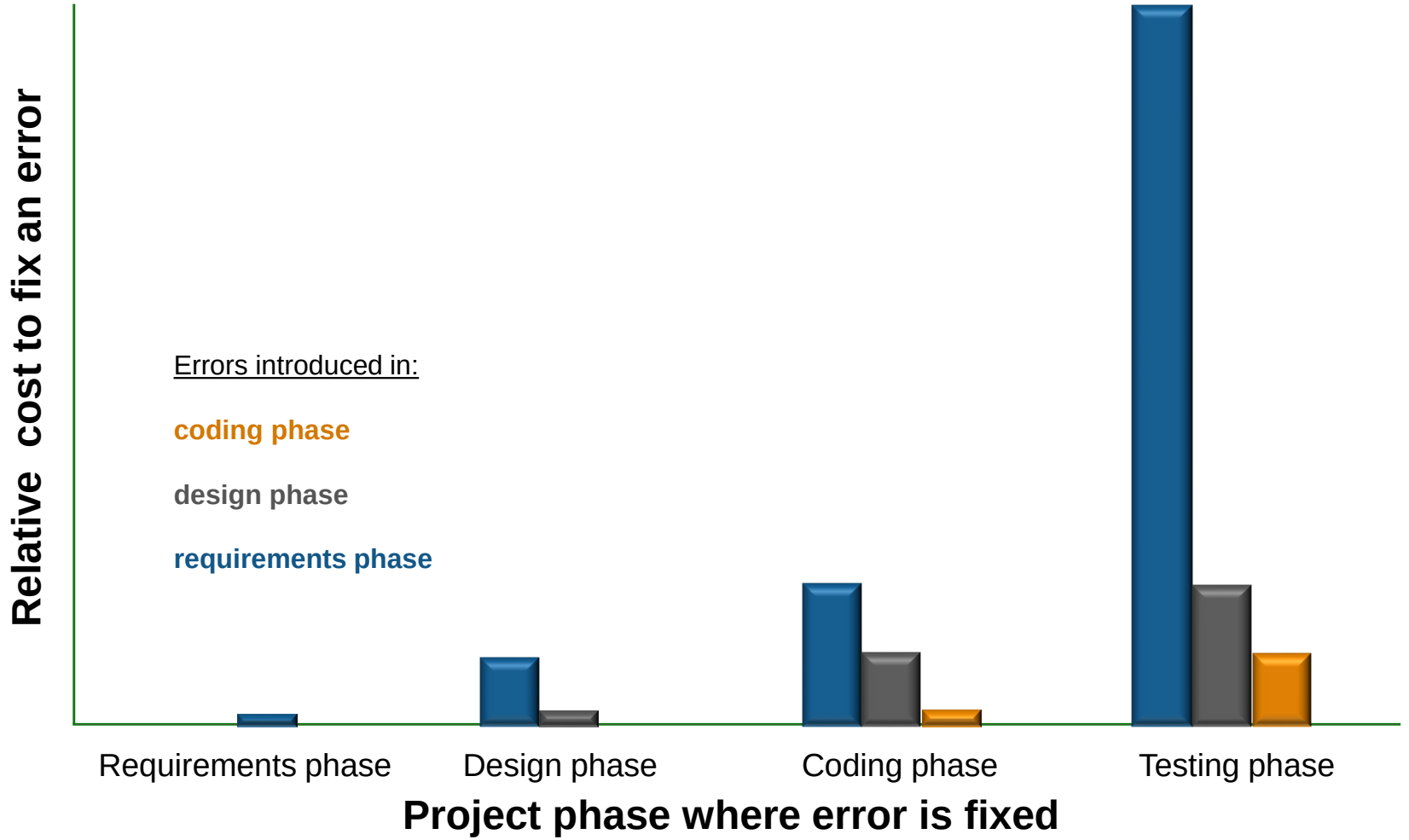
NATIONAL INSTRUMENTS

HIL & Rapid Control Prototyping

This Section Will Cover

- Model-Based Design
- MathWorks software for HIL and RCP with NI hardware
- Integration of compiled models with NI VeriStand
- Guidance for model development and solver

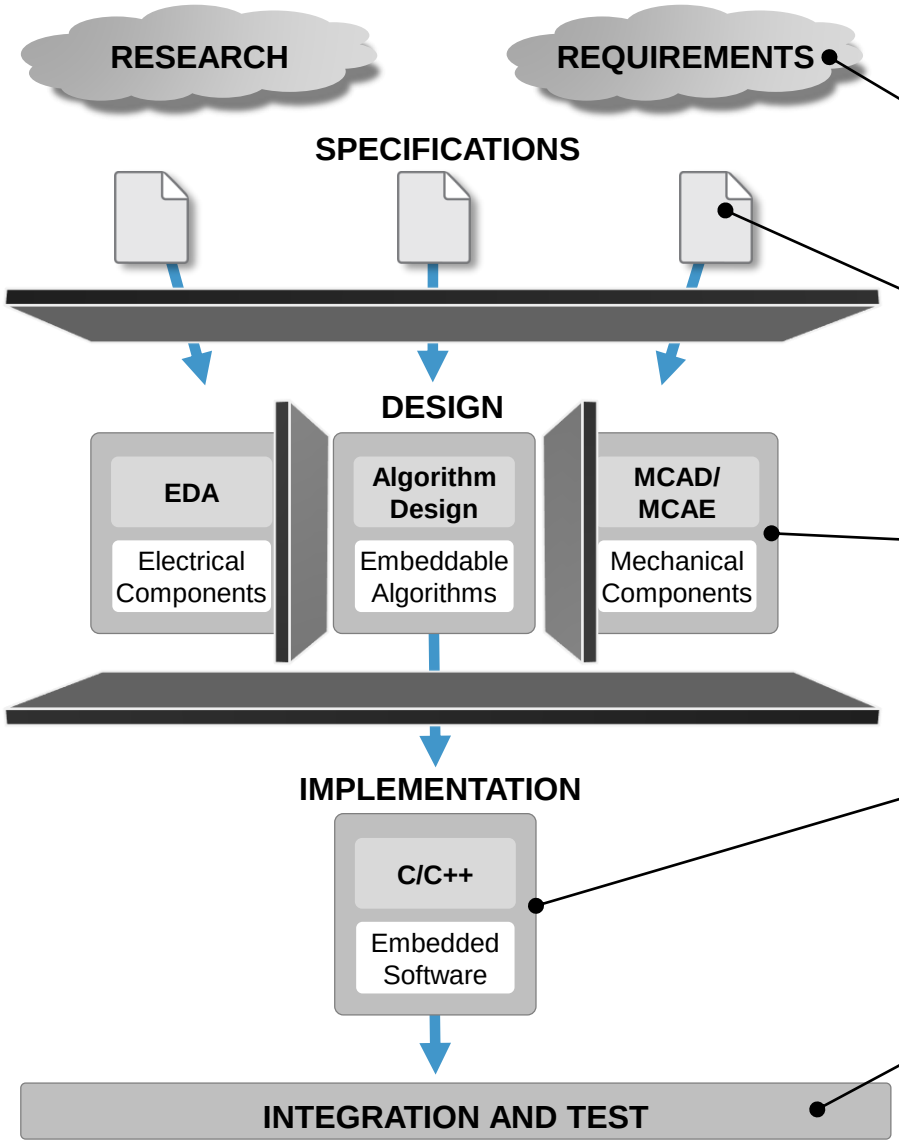
The Importance of Testing During the Design Phase



NATIONAL INSTRUMENTS

Source: Return on Investment for Independent Verification & Validation, NASA, 2004.

Traditional development process



Requirement Documents
- Difficult to analyze
- Difficult to manage as they change

Paper Specifications
- Easy to misinterpret
- Difficult to integrate with design

Physical Prototypes
- Incomplete and expensive
- Prevents rapid iteration
- No system-level testing

Manual Coding
- Time consuming
- Introduces defects and variance
- Difficult to reuse

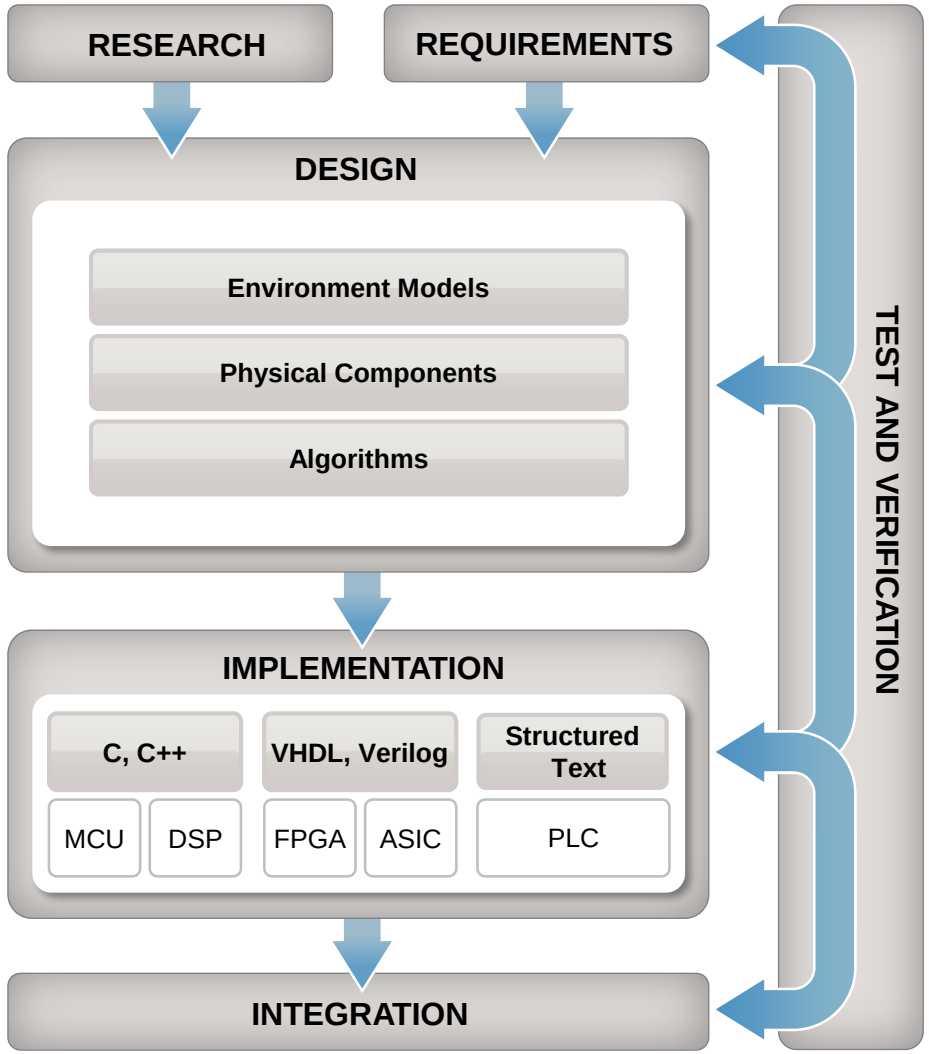
Traditional Testing
- Design and integration issues found late
- Difficult to feed insights back into design process
- Traceability

NATIONAL INSTRUMENTS

Model-Based Design with MathWorks & NI tools

MathWorks software:
- MATLAB
- Simulink
- Stateflow
- Simscape

NI software and hardware for HIL
- VeriStand
- CompactRIO, PXI
- and more



Both tool chains used together

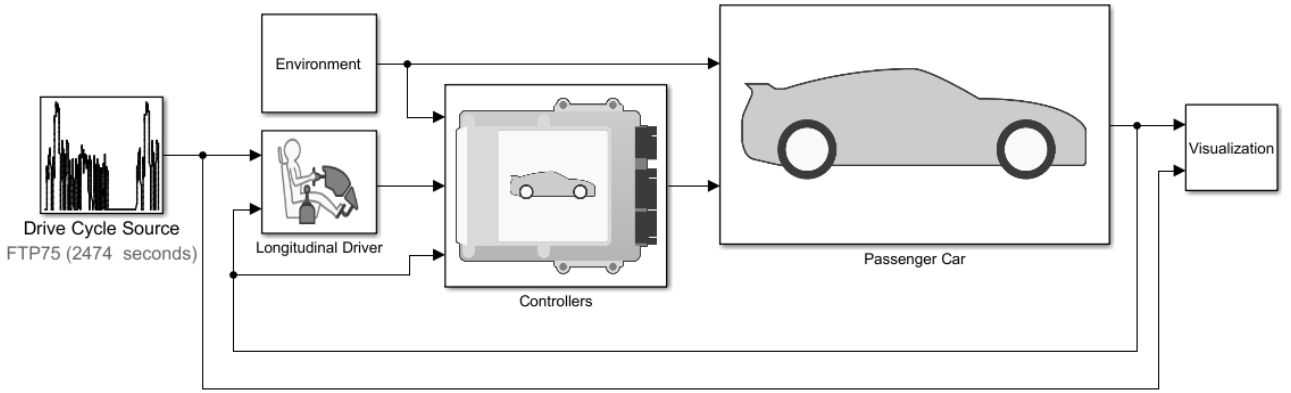
Applications:
- HIL testing
- Rapid Prototyping
- Monitoring

NATIONAL INSTRUMENTS

Core MathWorks products

SIMULINK®

Simulation and Model-Based Design



Model and simulate your system

- Use one multi-domain environment
- Model the system under test and the plant
- Simulate closed-loop system behavior

Test early and often

- Test your system under all conditions
- Validate your design with real-time testing
- Trace from requirements to design to code

Automatically generate code

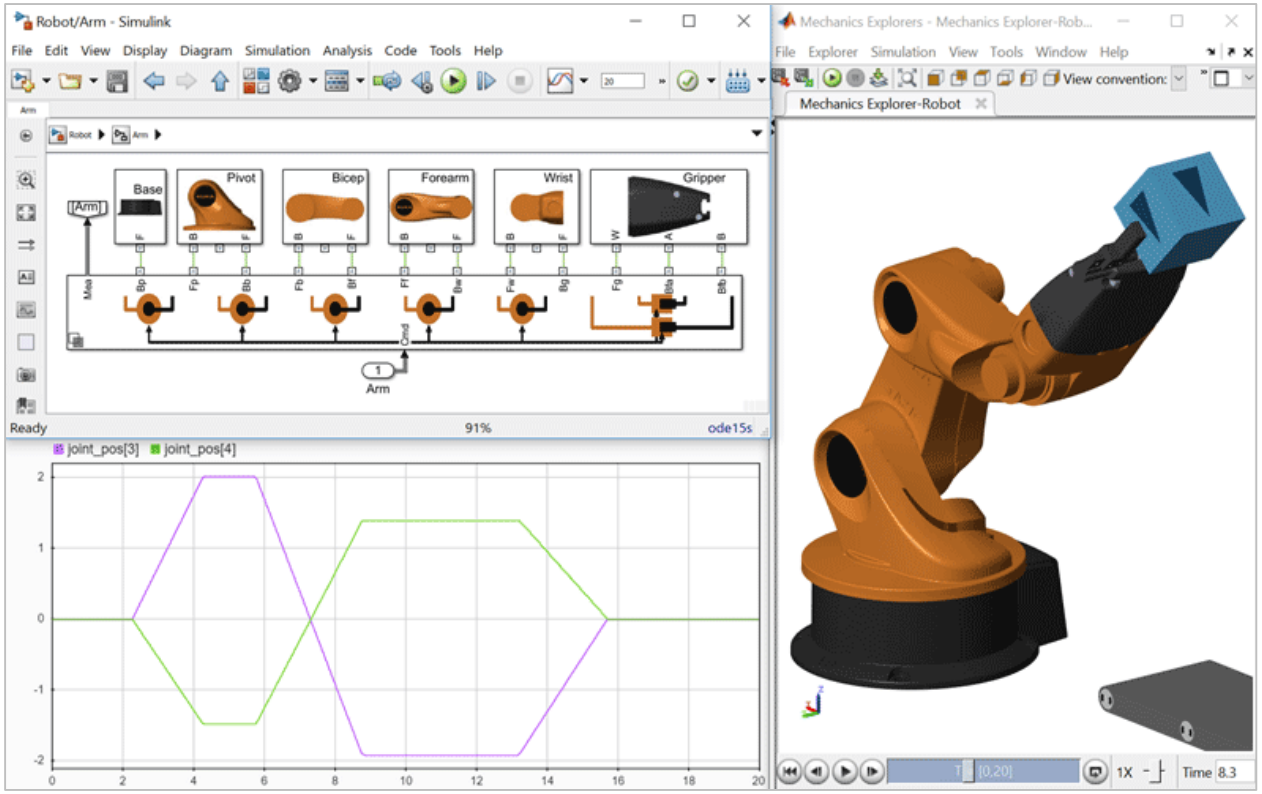
- Generate production-quality C and HDL code
- Deploy directly to embedded processors or FPGA's/

NATIONAL INSTRUMENTS™

Core MathWorks products

Simscape™

Model and simulate multidomain physical systems



Enables physical modeling (acausal) of multidomain physical systems

- Assemble a schematic
- Equations derived automatically
- Leverage MATLAB and Simulink

With Simscape you can:

- Refine requirements for system
- Discover integration issues early
- Design control systems and logic
- Optimize system-level performance
- Test embedded software without hardware prototypes

NATIONAL INSTRUMENTS

Core MathWorks products

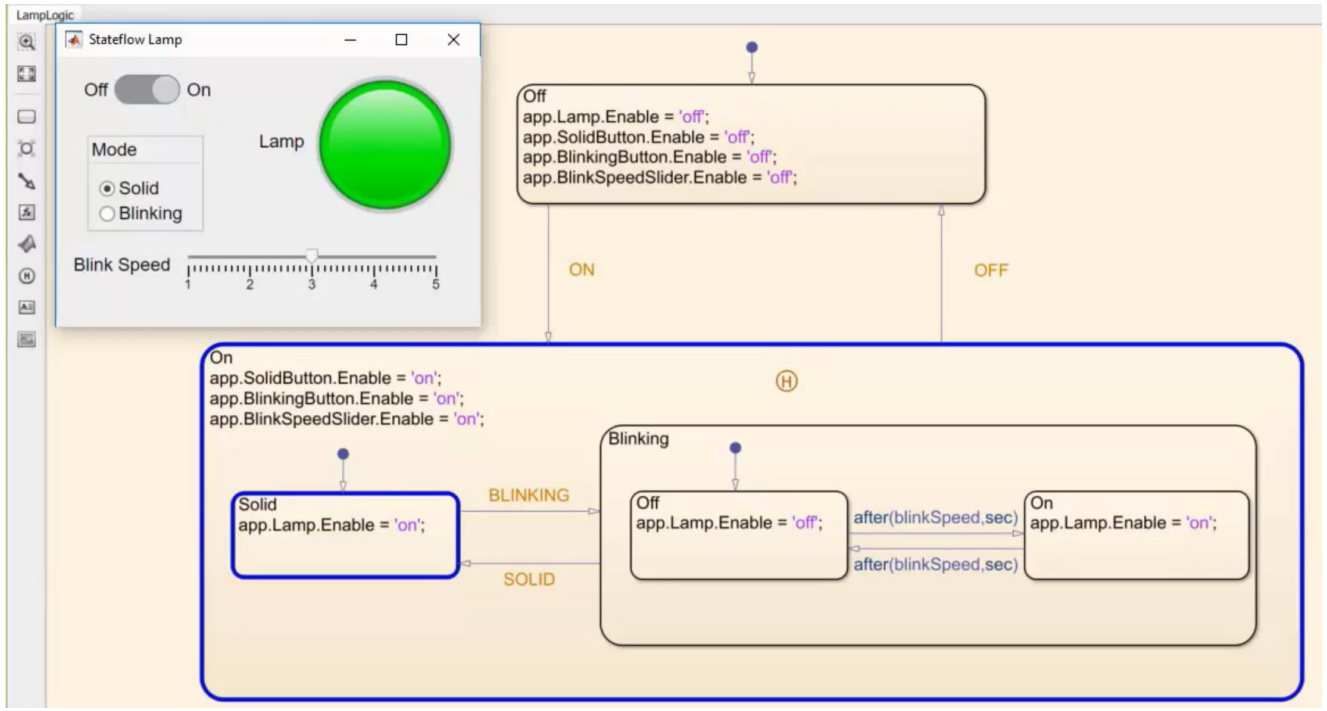
Stateflow®

Model and simulate decision logic

Design control logic
- Design state machines
- Create flow charts
- Model logic with tables

Execute and debug charts
- Visualize system behavior with animations
- Step through chart execution in detail

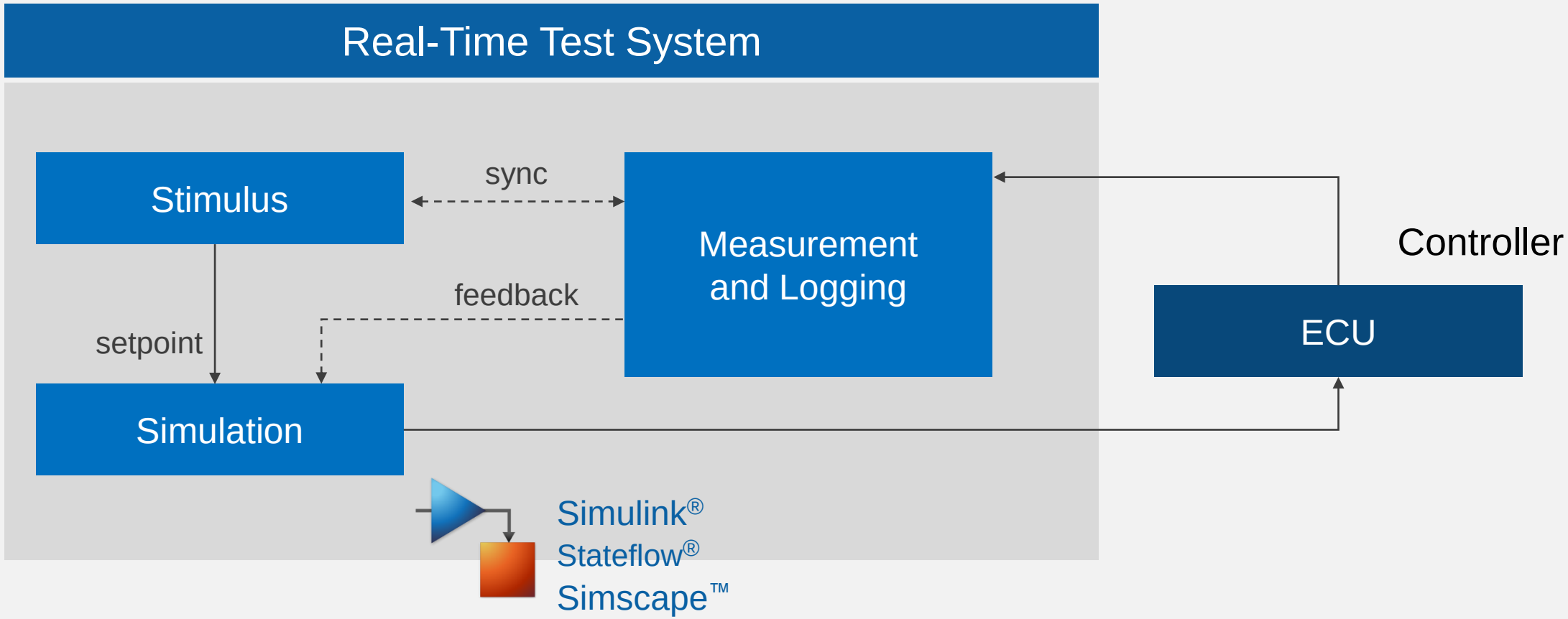
Automatically generate code
- Generate production-quality C and HDL code
- Deploy directly to embedded processors or FPGA's/ASIC's



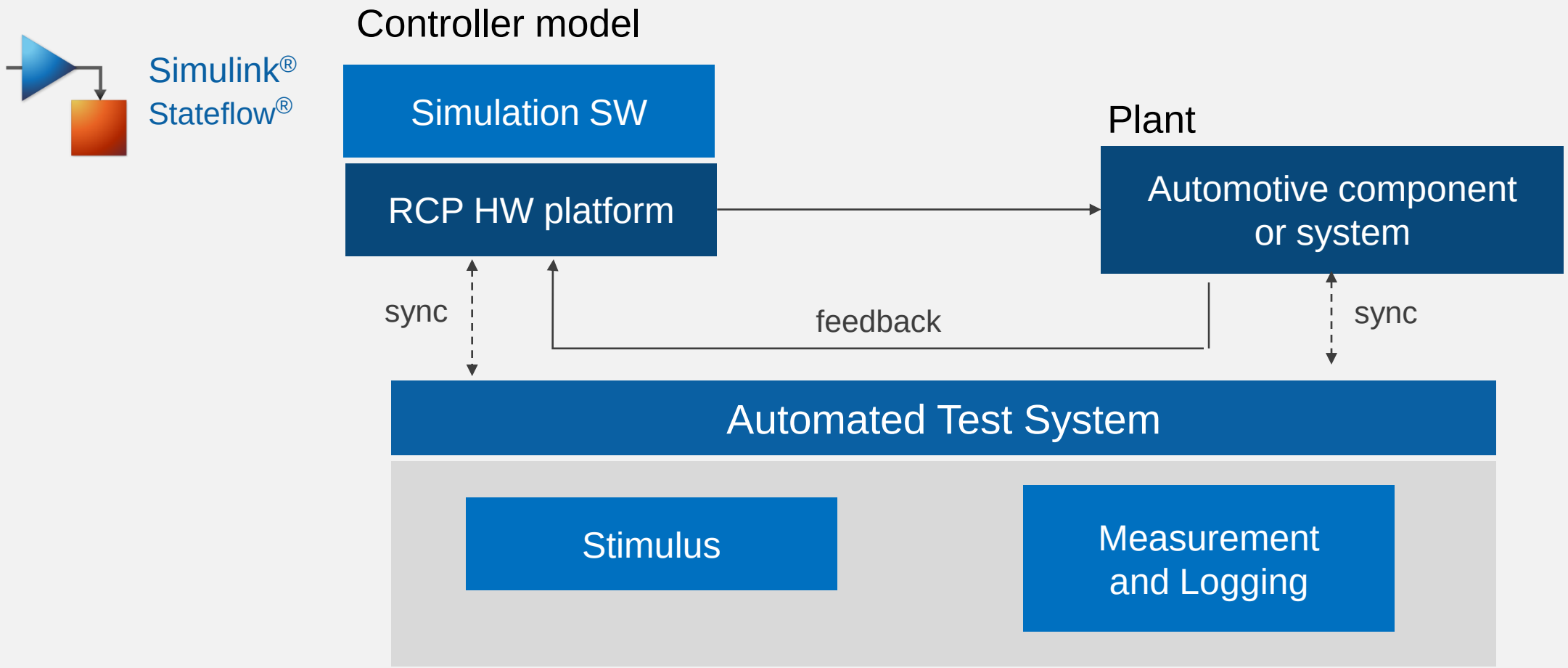
NATIONAL INSTRUMENTS

HIL Software Architecture – Automotive Example

Plant model



RCP Software Architecture – Automotive Example



NI VeriStand

Embedded Test Software Functionality

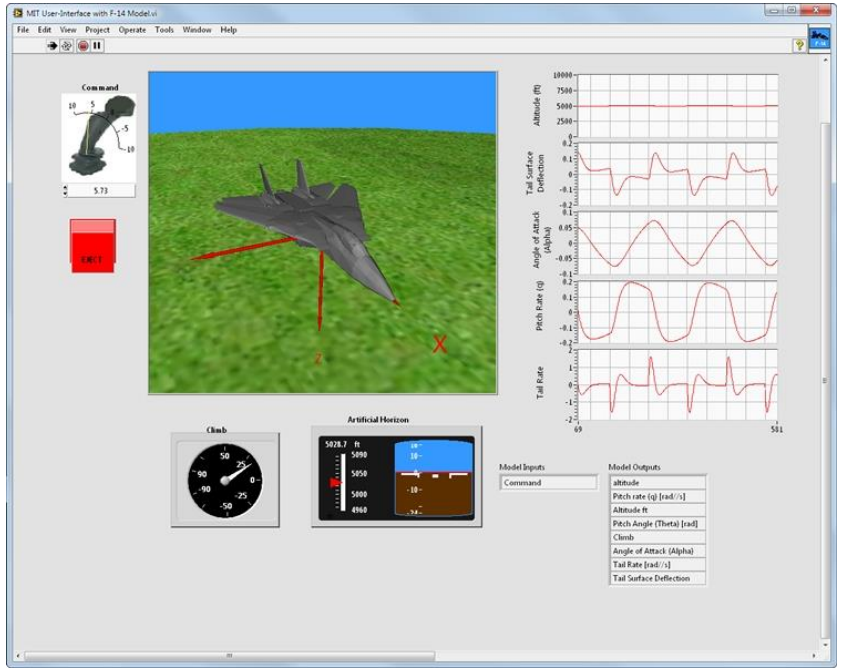
- RT Stimulus Generation
- Hardware I/O
- Alarming
- Deterministic Model Execution
- Multichassis Synchronization
- Closed-Loop Control
- Data Logging
- Test Automation
- Calculated Channels
- User Account Management
- Multichassis Data Sharing
- Scaling and Calibration

NATIONAL INSTRUMENTS

LabVIEW Model Interface Toolkit

Integrate 3rd party simulation models in LabVIEW

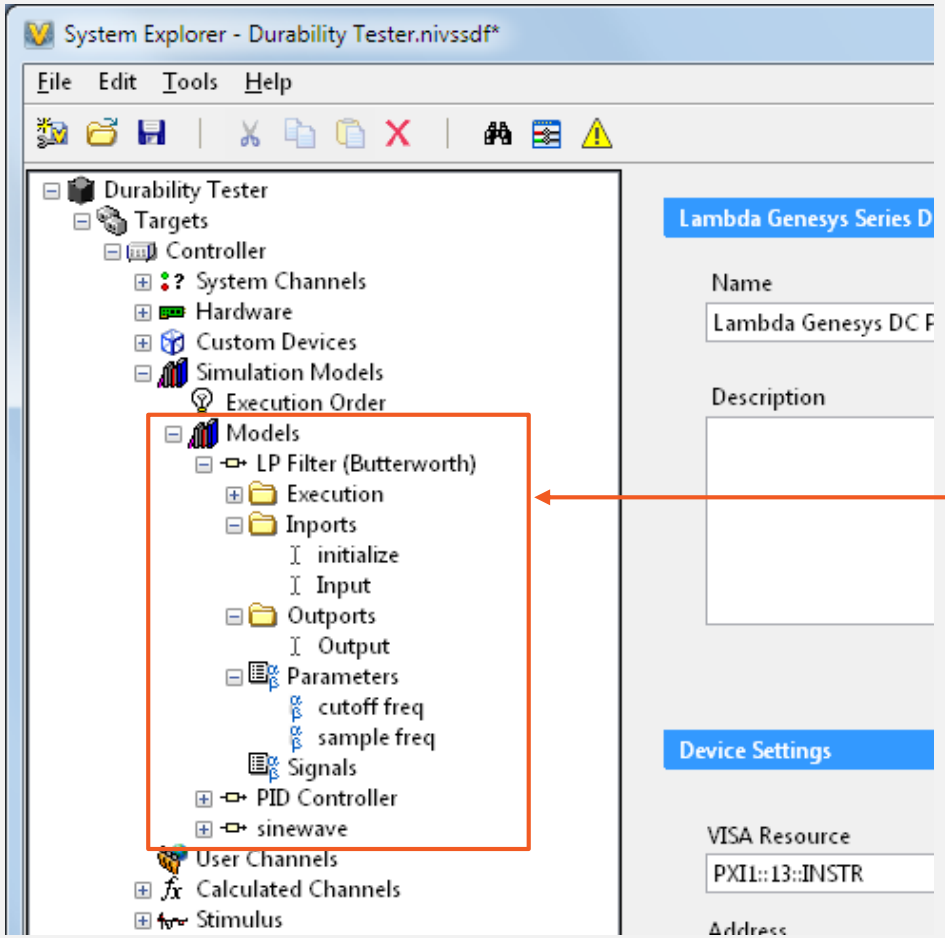
- Connect Simulink® & Simscape™ simulation models to NI hardware using Simulink® Coder™
- Can call MATLAB® through MATLAB function block in Simulink model
- Run multiple models simultaneously
- Combine models from multiple simulation environments
- Co-simulation: can include fixed-step discrete-time and continuous-time solvers for ODEs



LabVIEW Model Interface Toolkit page

NATIONAL INSTRUMENTS

VeriStand Model Integration



Import models from MathWorks software for real-time execution of:

- System simulations
- Closed-loop controllers
- Signal analysis algorithms

NATIONAL INSTRUMENTS

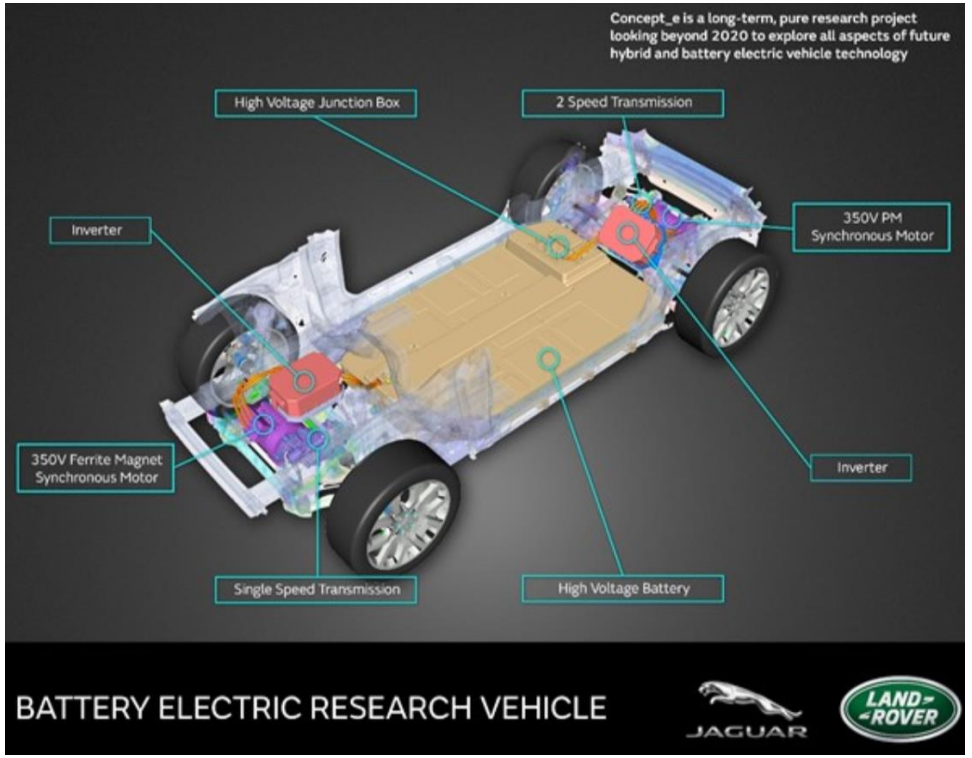
BMS HIL Test System Helps Jaguar Land Rover Shorten Time-to-Market for Hybrid and Electric Vehicles

Challenge

Creating a system to rapidly design and validate battery management system (BMS) firmware algorithms on prototype hardware, and safely test the unit with various battery chemistries, fault scenarios, and drive profiles

Solution

A hardware-in-the-loop (HIL) test platform based on MathWorks Simulink, NI PXI, LabVIEW, VeriStand, DIAdem, EtherCAT hardware, and Bloomy's Battery Simulator 1200 instruments to simulate a 24-cell advanced-chemistry, hybrid- and electric-vehicle battery



NATIONAL INSTRUMENTS

Rapid Control Prototyping (RCP) with MathWorks, NI, and Quanser

- Simulink, Stateflow, Simulink Coder and NI software (VeriStand or LabVIEW) and hardware
- Simulink, Stateflow, Simulink Coder and NI hardware (no VeriStand or LabVIEW):
  - Quanser (myRIO, select CompactRIO configurations, other possible)
  - Your custom C/C++ solution with Simulink Coder

QUARC/NI cRIO System(s)

QUARC Host:
Controller design

QUARC Target:
Code running here

MATLAB® & SIMULINK®
QUARC®

Host PC

Network

NI cRIO-9063

Control Signal(s)

cRIO Module Configuration(s):
Fixed FPGA Personalities

Feedback signals: Sensor(s), etc.

NATIONAL INSTRUMENTS

Wireless Design and Test

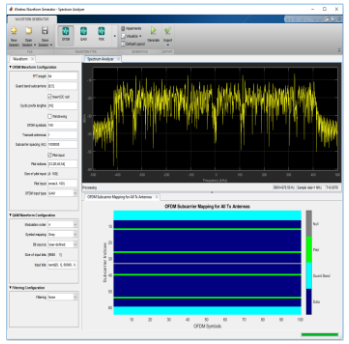
Better Workflows for Wireless Design and Test

Improving productivity through automation, reuse, and defined interfaces

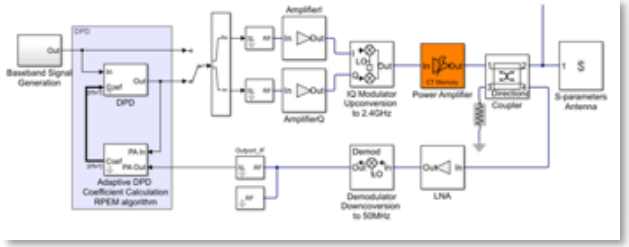
MathWorks Software

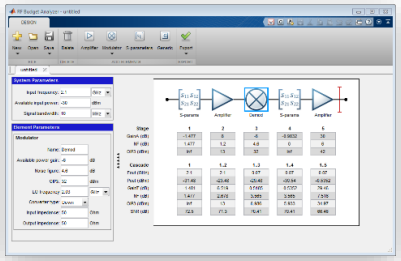
```
% Establish the number of component carriers.
numCC = length(NDLRB);

% Create transmission for each component carrier
enb = cell(1,numCC);
for i = 1:numCC
    enb{i} = lteRMCDL('R.5');
    enb{i}.NDLRB = NDLRB(i);
```



MATLAB® & SIMULINK®



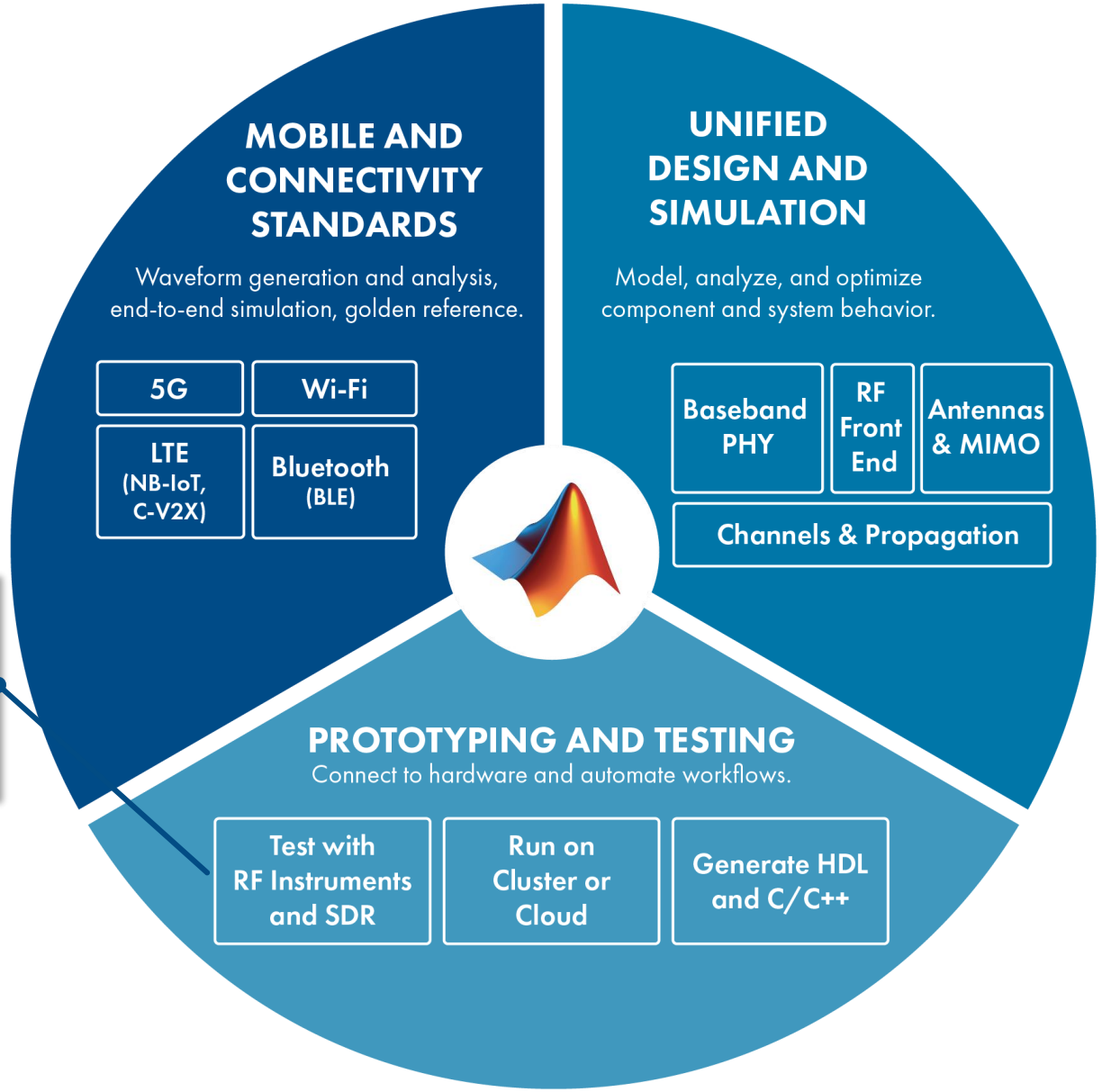


NI Hardware and Software

NATIONAL INSTRUMENTS

Wireless Communications Development

With MATLAB, Simulink, and NI Tools

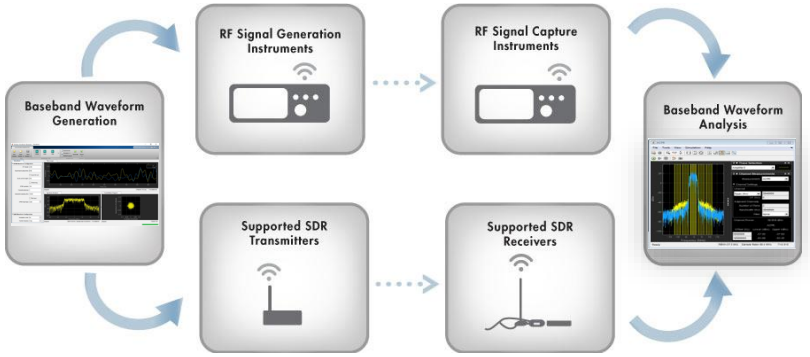


NI SDR and Instruments

NATIONAL INSTRUMENTS

Available Now: MathWorks Support for NI Test Hardware

Generate and Analyze Waveforms



Ettus USRP

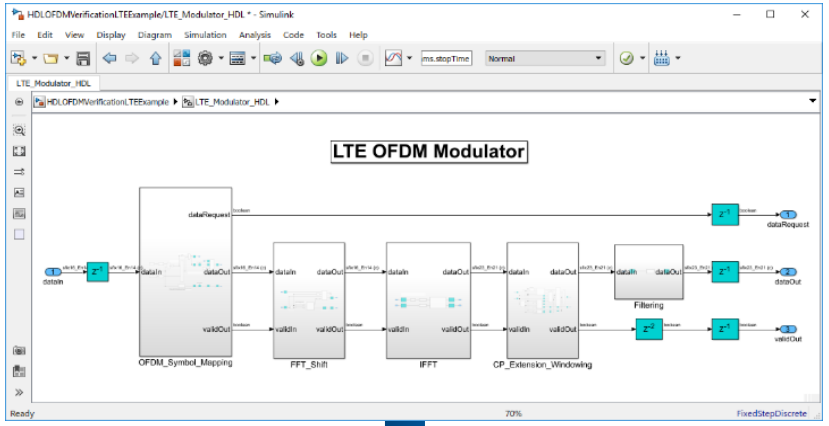
NI USRP

NI PXI

Support Packages for USRP (Communications Toolbox)

Instrument Control Toolbox

Prototype Algorithm IP



Ettus Embedded Series USRP

- HDL Coder
- Embedded Coder

NATIONAL INSTRUMENTS

Application Case Studies

Research prototyping

Application: Modulation Classification with Deep Learning

Configuration:

- MATLAB, Communications Toolbox, Deep Learning Toolbox
- NI USRP Radio

Design validation

Application: PA / DPD Design and Device Characterization

Configuration:

- MATLAB, Simulink, Communications Toolbox, RF Blockset
- NI PXI Test Instrument, RFmx software

NATIONAL INSTRUMENTS

Case Study: Modulation Classification with Deep Learning

- Deep learning and machine learning applications in Wireless Communications
  - Waveform classification
  - Channel estimation and equalization
  - Adaptive DPD
- Challenges
  - Domain-specific algorithm development and simulation
  - Workflow from simulation to testing on real-world signals
- Workflow with MATLAB and NI Tools
  - Design, simulate, and analyze in MATLAB
  - Live signal testing using MATLAB and USRP

NATIONAL INSTRUMENTS

Deep Learning Workflow

Access and Explore Data → **Label and Preprocess Data** → **Develop Predictive Models** → **Test and Integrate Models**

Files

Databases

Sensors

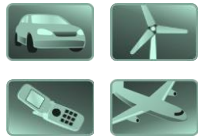
What to do when no pre-labeled data exits?

Desktop Apps

Enterprise Scale Systems

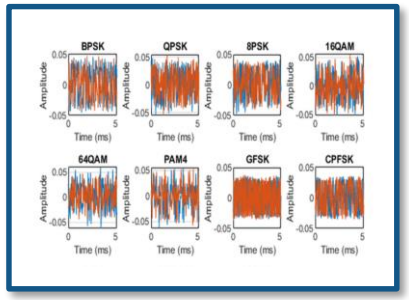
Java
MATLAB
Excel
.NET
C/C++
dll
.exe
Python

Embedded Devices and Hardware

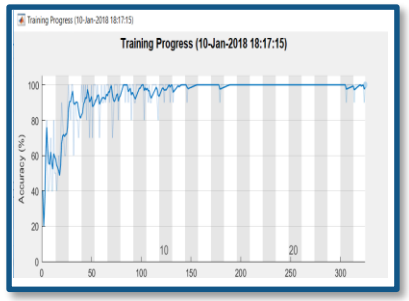


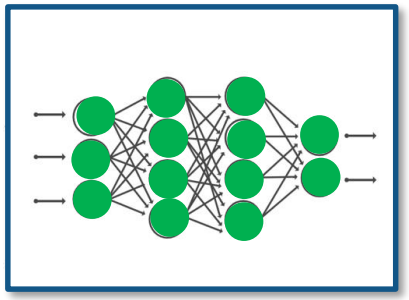
Model Development with Synthetic Signals

Train



- Noise
- Multipath fading
- Frequency offset
- Sampling rate offset

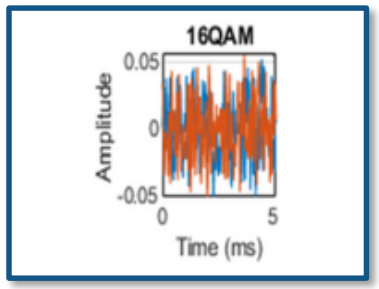


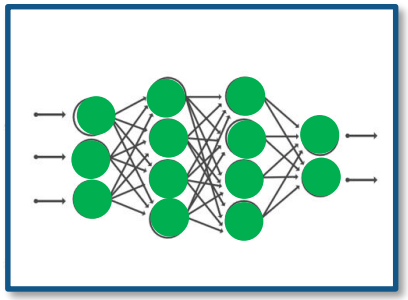


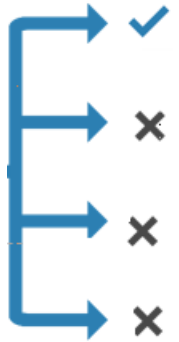
Synthesize labeled data → Add impairments → Train Network → Trained Model

Test

New Signal → Pre-Trained Model → Classification



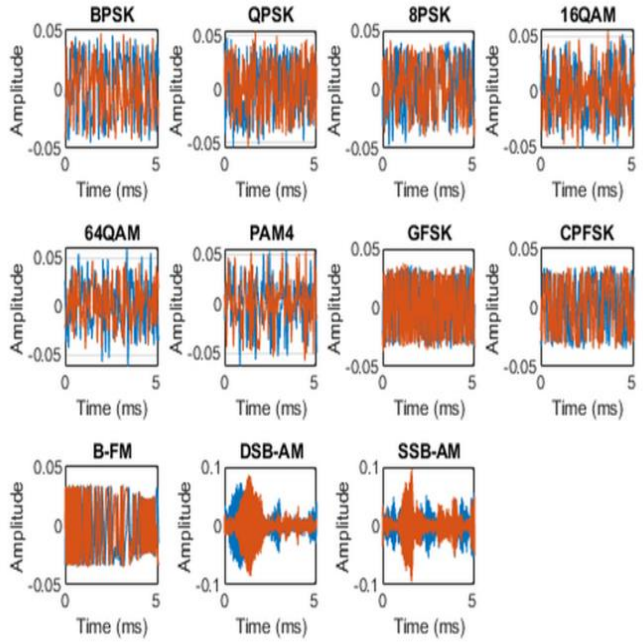


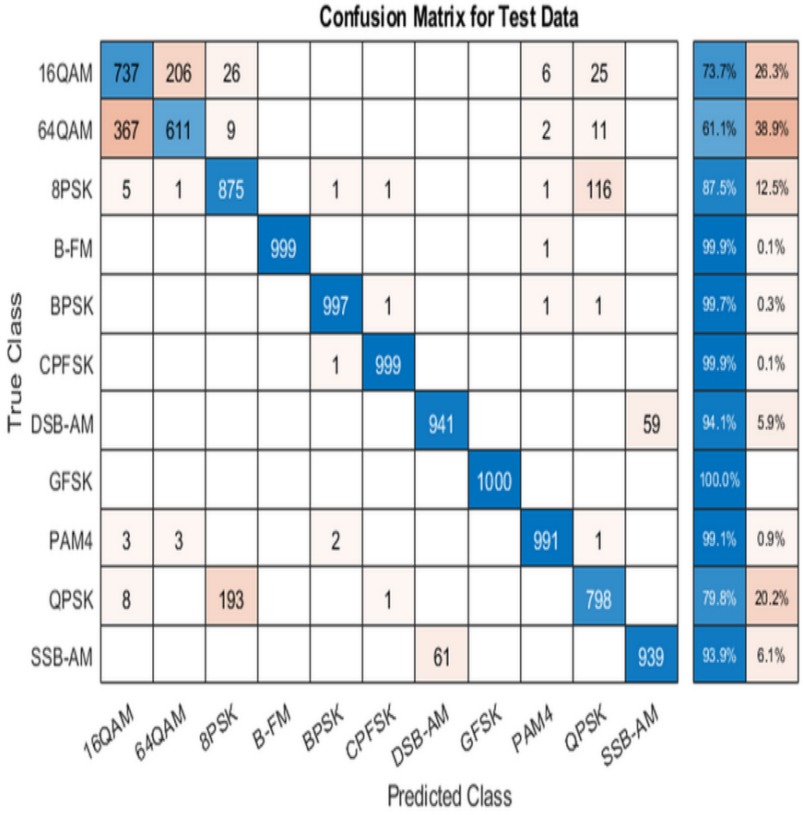


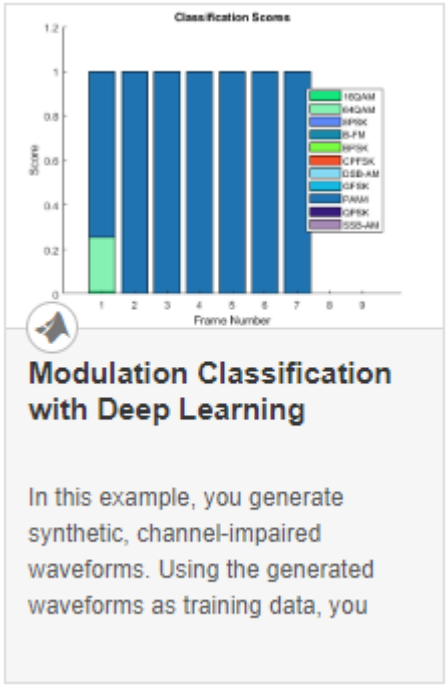
NATIONAL INSTRUMENTS

Wireless Modulation Classification with Deep Learning

- Generate synthetic modulated signals
- Apply channel impairments
- Train a CNN to classify modulation types



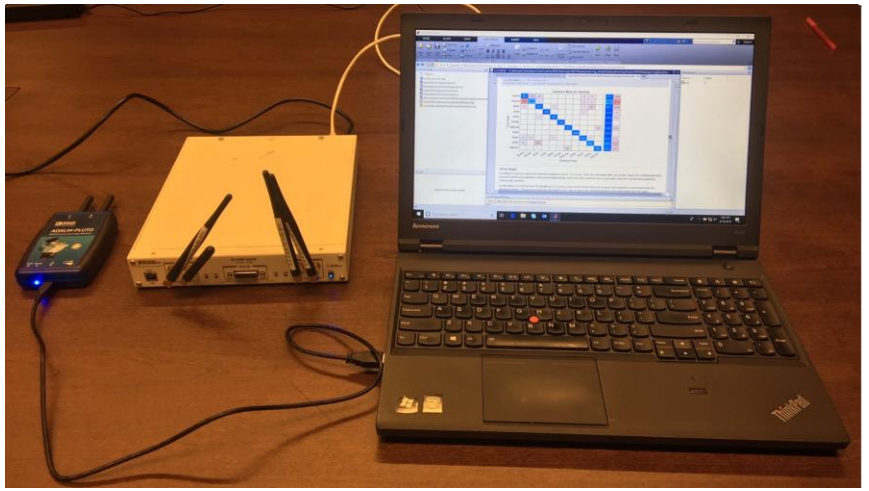


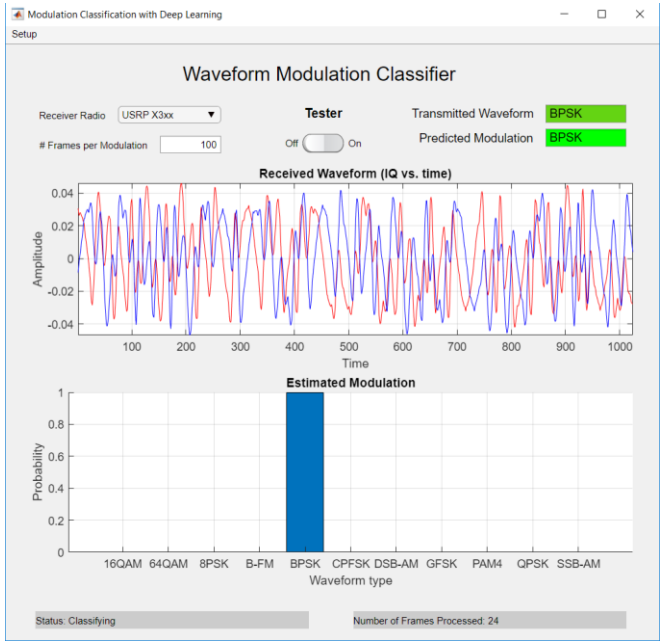


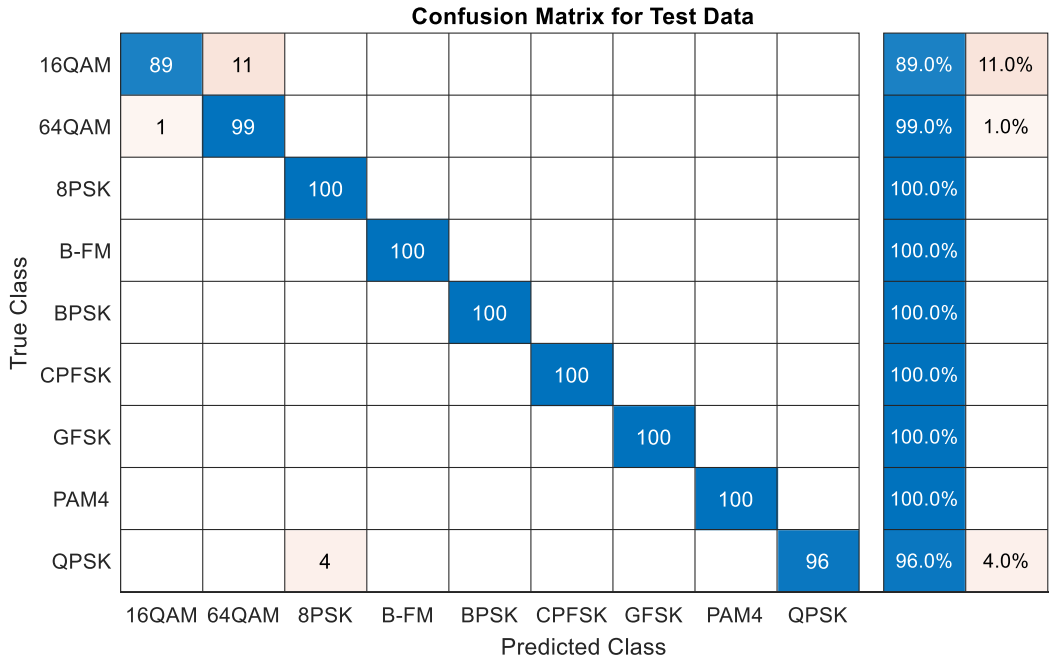
NATIONAL INSTRUMENTS

Over-the-air Test with NI SDR

- Generate OTA test signals using any source
- Connect MATLAB to NI SDR to receive signals
  - USRP-2943 or Ettus X310
- Process real-time data in MATLAB





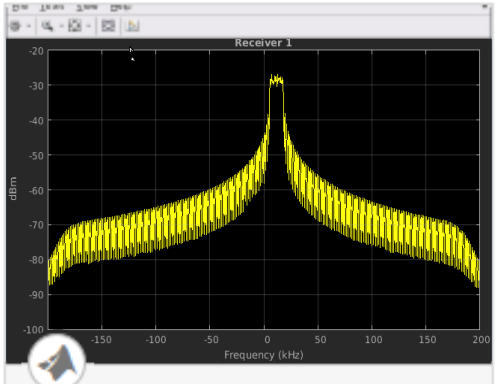


NATIONAL INSTRUMENTS

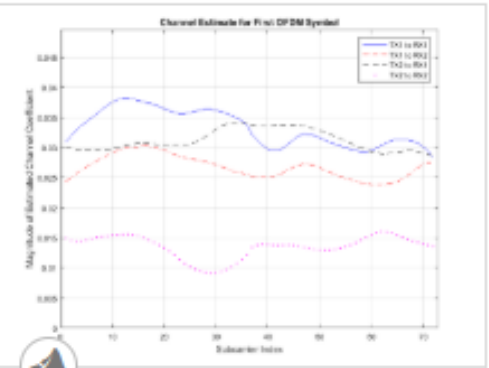
MATLAB and USRP: Reference Examples

- Test and analyze algorithms and standards-based designs
- Transmit/receive live signals
- Prototype custom designs (E310)

Learn more about
USRP Support Package for Communications Toolbox

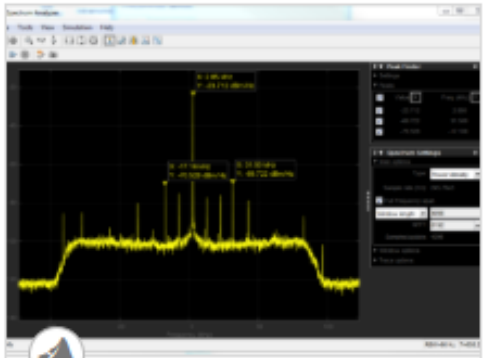


Multi-User Transmit Beamforming with USRP® Hardware

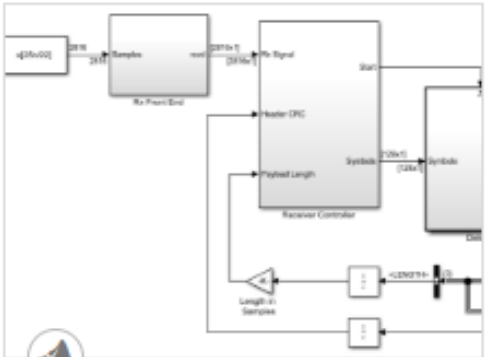


LTE Cell Search, MIB and SIB1 Recovery with Two Antennas

Transmission and Reception of an Image Using LTE Toolbox and a...



Frequency Offset Calibration Receiver Using USRP® E310



IEEE 802.11 WLAN - Beacon Frame Receiver with USRP® Hardware

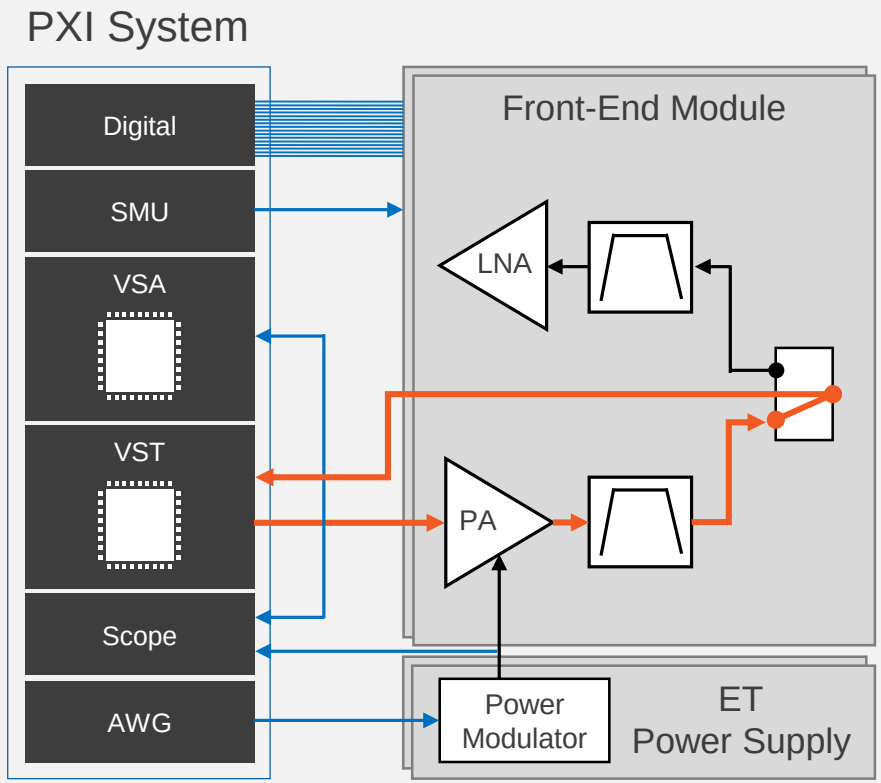
NATIONAL INSTRUMENTS

5G PA/FEM Characterization

NI Front-End Module Test With DPD

- VST with 1 GHz instantaneous generation and analysis bandwidth
- Free NI-RFmx SpecAn with LUT, MPM, and GMP DPD models
- Free RFIC Test Software with DPD automation examples

1. Generate reference waveform and acquire distorted waveform
2. Create predistortion model by comparing reference waveform to distorted waveform
3. Apply DPD to reference waveform using predistortion model
4. Generate predistorted waveform and make measurements

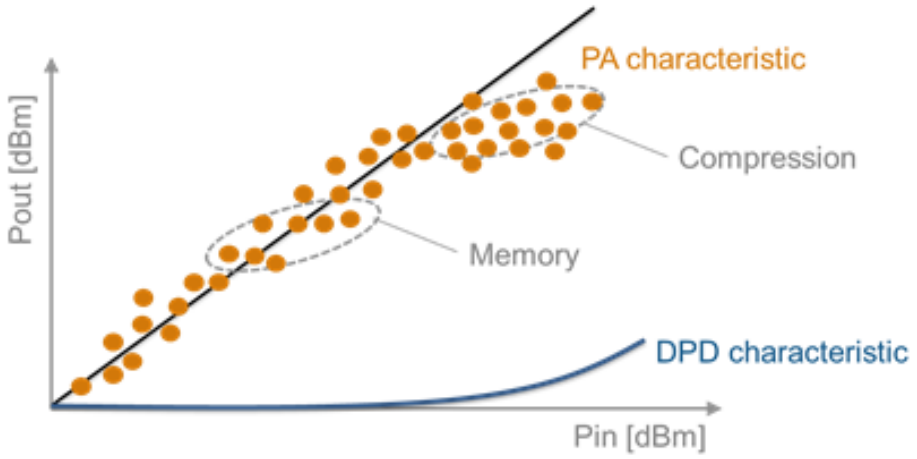


NATIONAL INSTRUMENTS

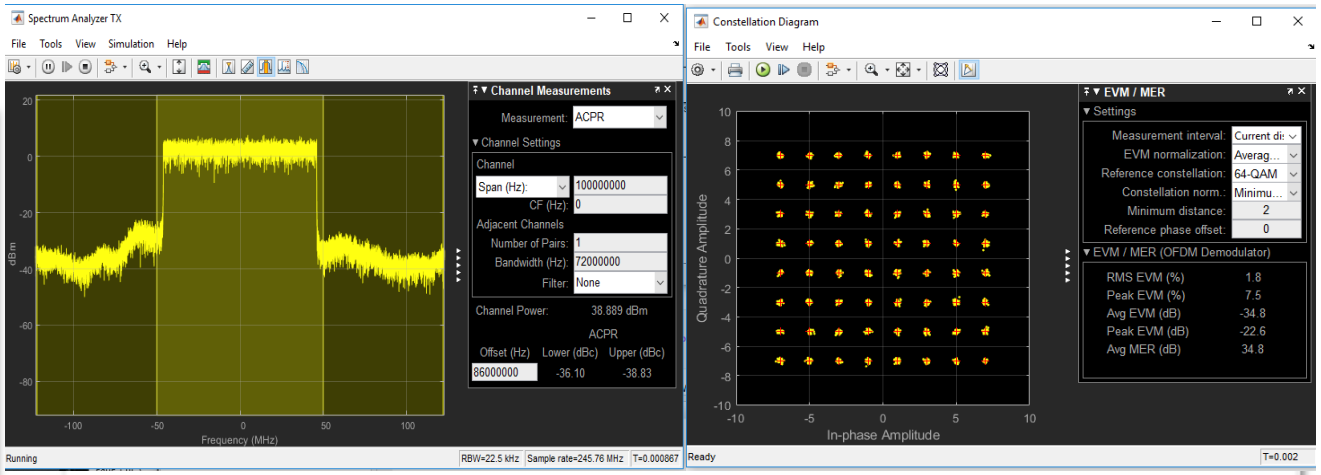
MathWorks

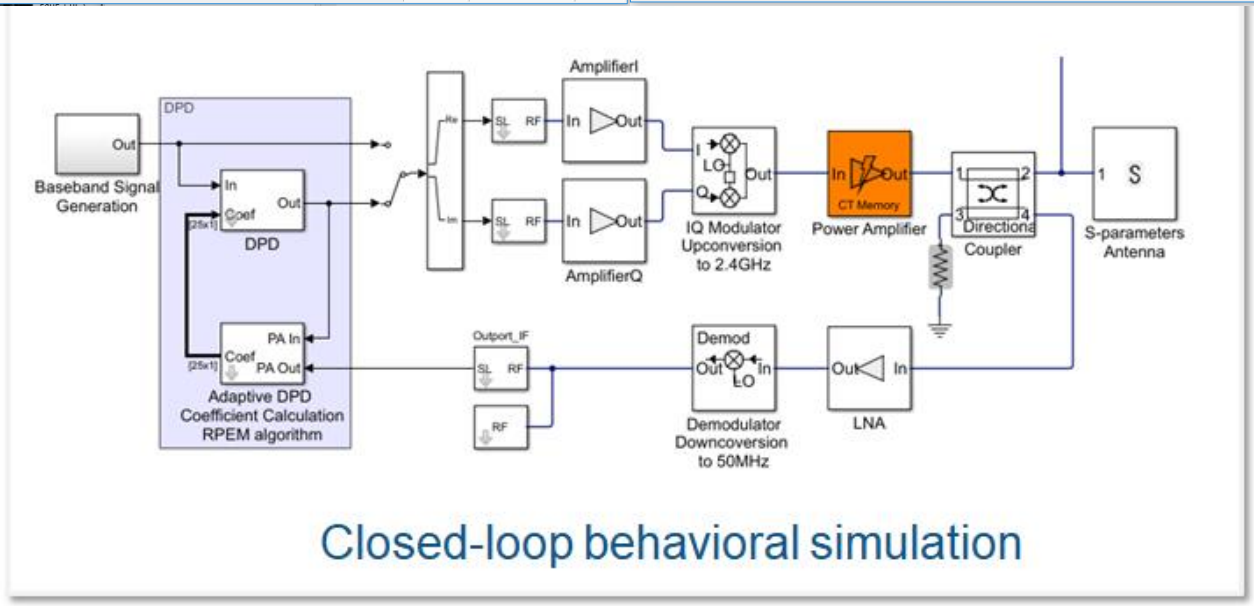
Design of PA Linearization with DPD

With MATLAB and Simulink



Characterize power amplifier (PA) model using measured data



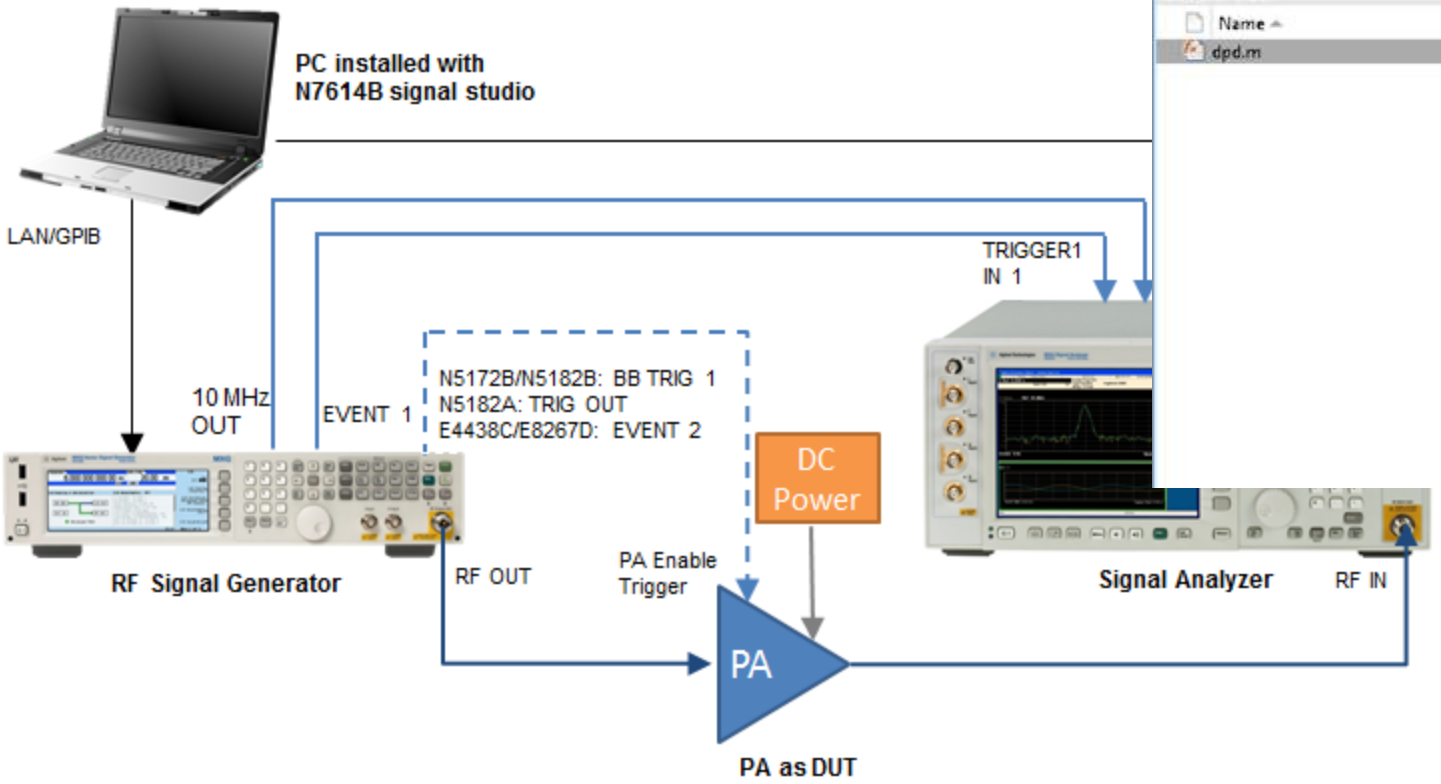


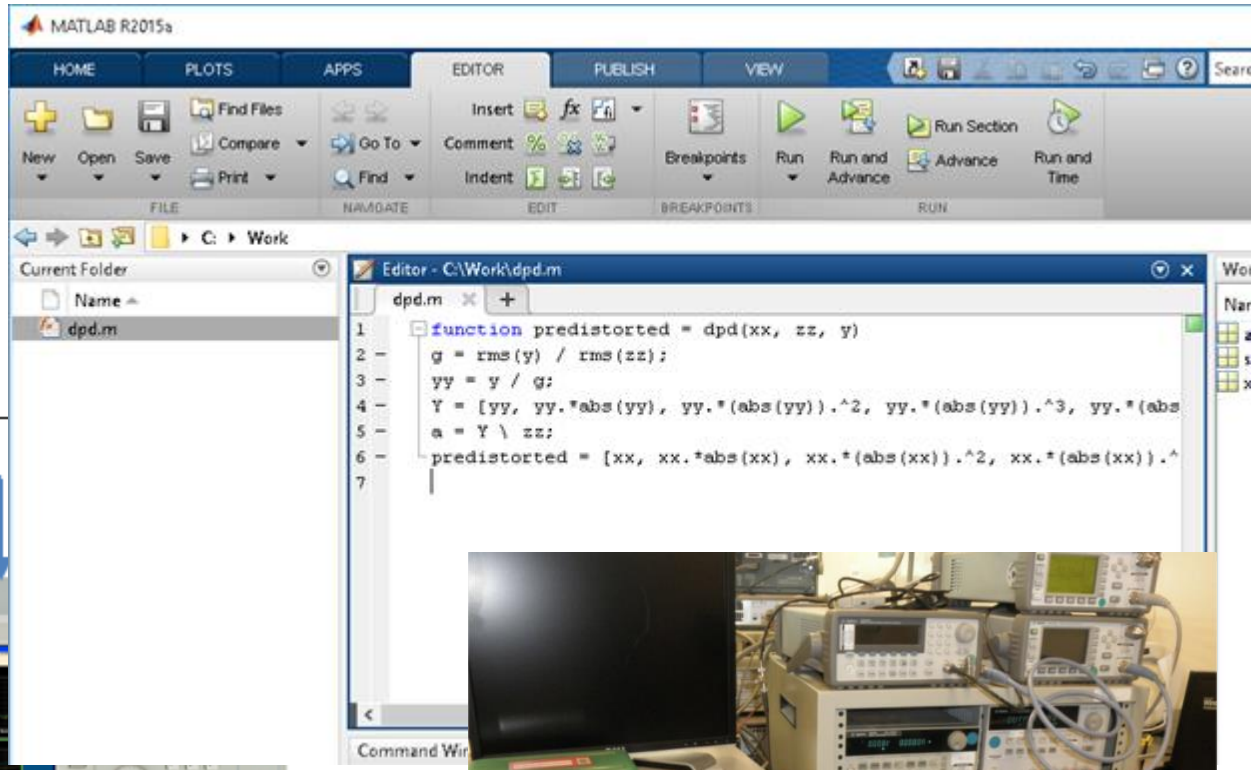
Closed-loop behavioral simulation

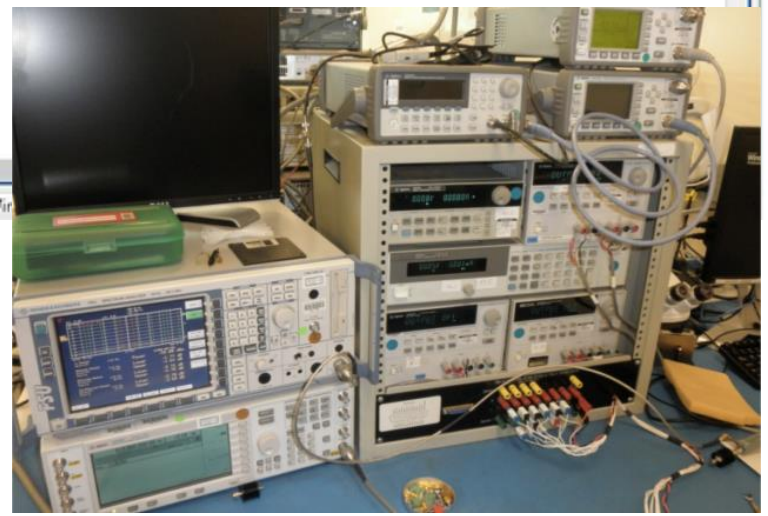
NATIONAL INSTRUMENTS

Traditional T&M Setup for MATLAB based PA characterization w/ DPD algorithm running in MATLAB

- Familiar user experience for many engineers
- Slower measurement speed, Large physical footprint
- Expensive to upgrade or replace – even Software
- Difficult to synchronize for ET & DPD
- Tradeoffs between speed and accuracy



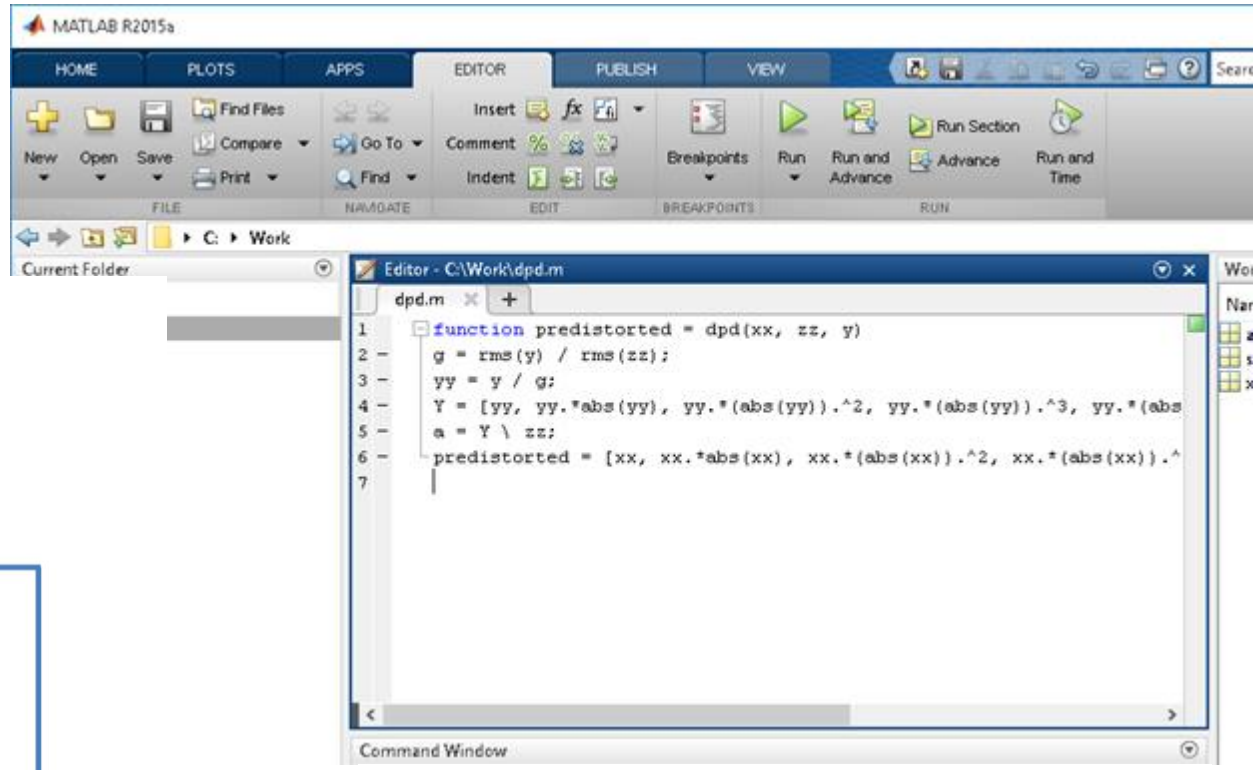


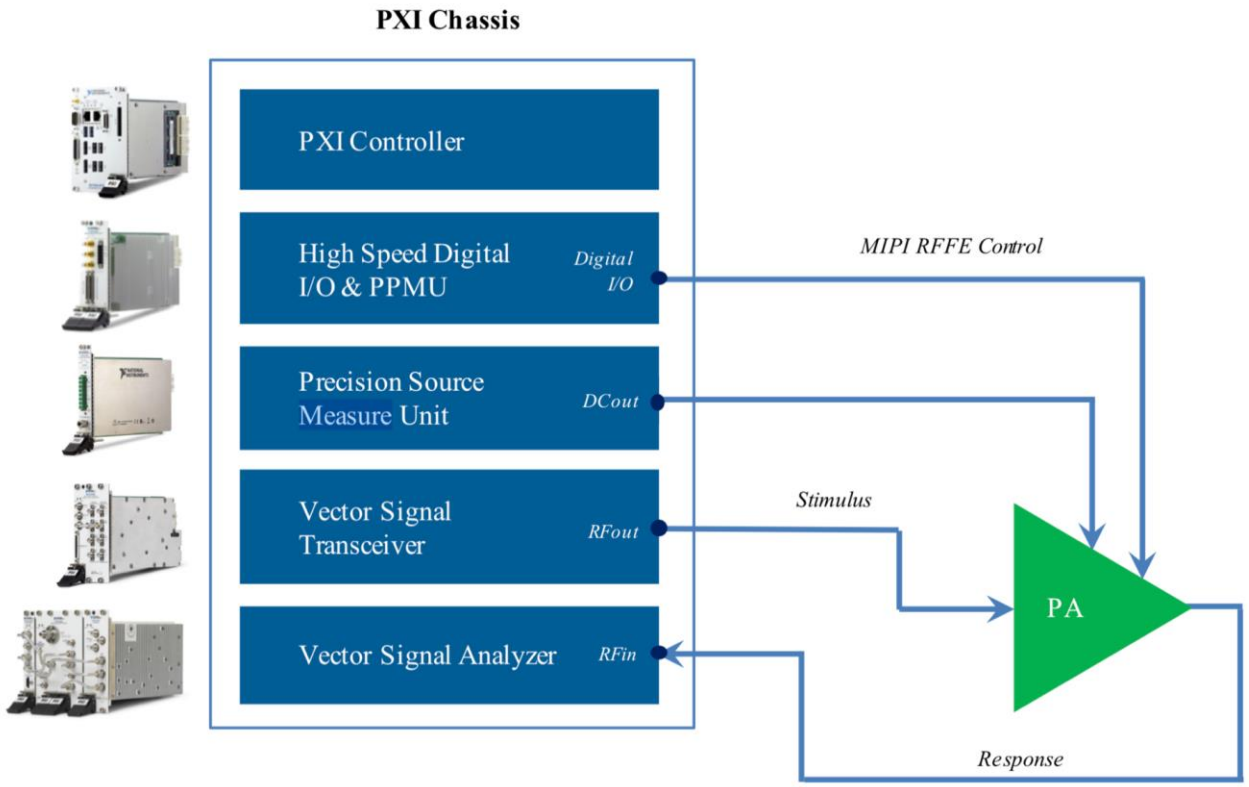


NATIONAL INSTRUMENTS

NI PXI Setup for MATLAB based PA characterization w/ DPD & ET algorithm running in MATLAB

- Similar user experience as box-instruments
- Faster and FPGA-accelerated measurement speed, at a fraction of the physical footprint
- Modularity for incremental upgrades
- Native synchronization technologies at sub nanosecond accuracy
- R&D grade measurement accuracy with production test speed





NATIONAL INSTRUMENTS


NATIONAL
INSTRUMENTS™