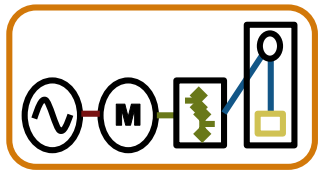
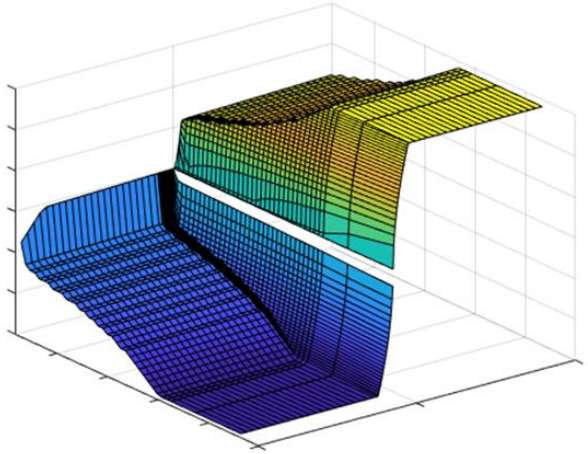
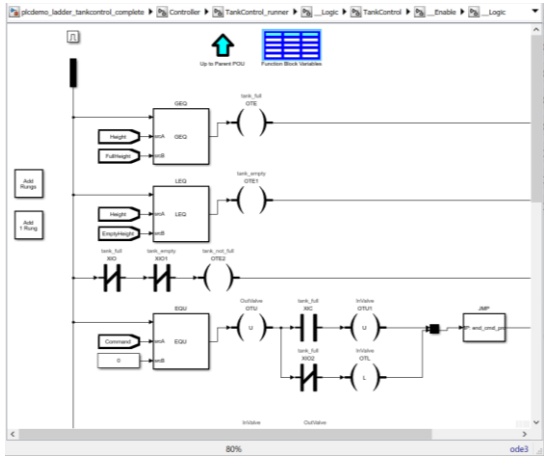
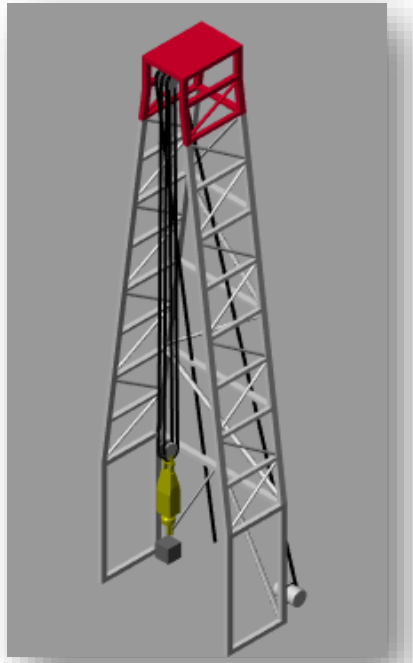
Digital Twin Applications for Oil/Gas Industry

Optimizing and Validating Controls for Drilling









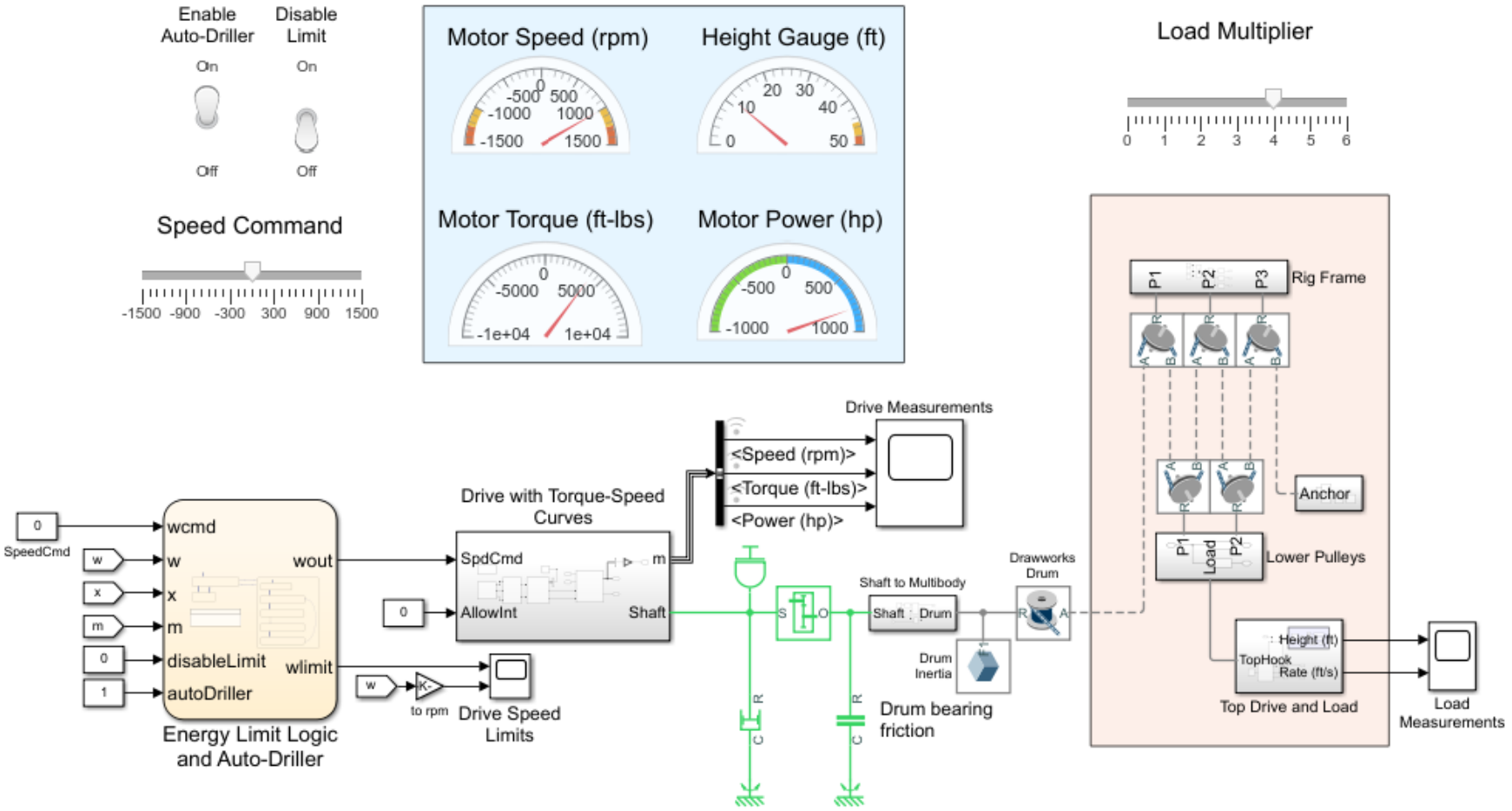
Jonathan LeSage, PhD

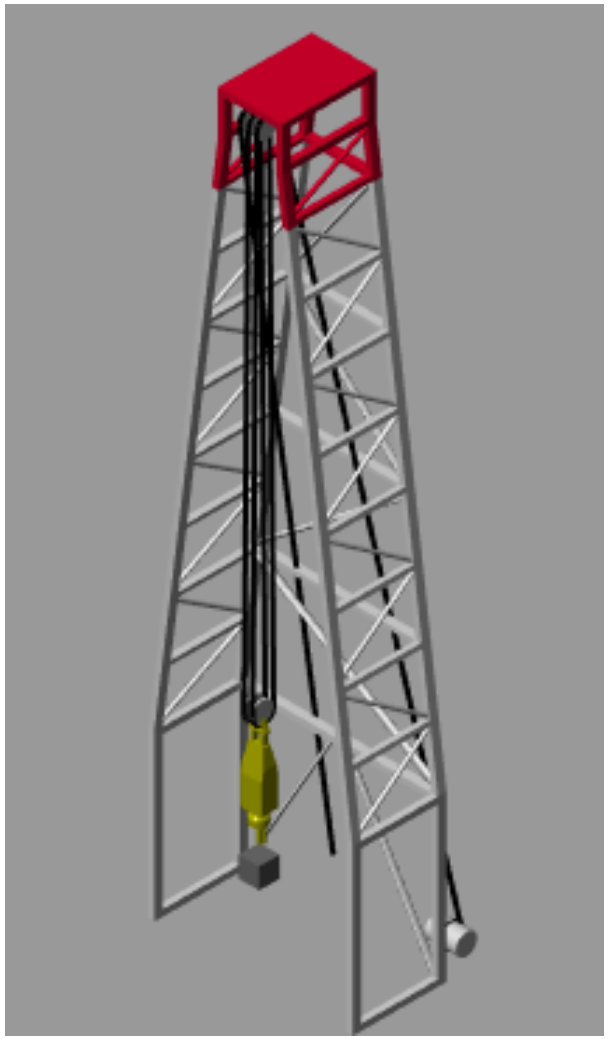
Senior Application Engineer – Energy and Automation

Agenda

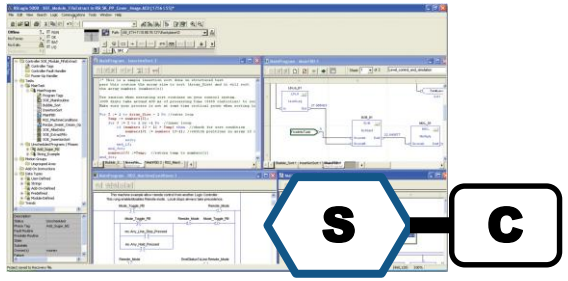
- Introduction to Modeling with Simulink and Simscape
- Leveraging Models
  - Calibrating Digital Twin with Field Data
  - Sizing Systems using Operation Data
  - Digital Twins for Anomaly Detection
- Optimizing and Validation of Controls
  - Optimizing Drilling Operations
  - Importing Existing PLC Ladders and C Software
  - Controller Error Checking and Test Vectors
- Hardware-in-the-loop Testing of PLCs

Digital Twin of Drawworks Drilling System





Design

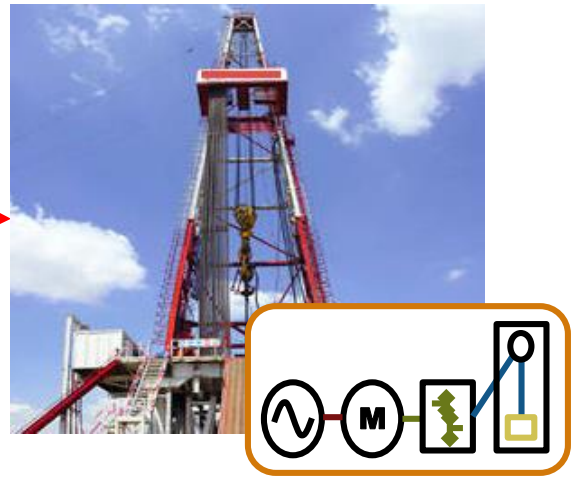


PLC Development Environment

Implement

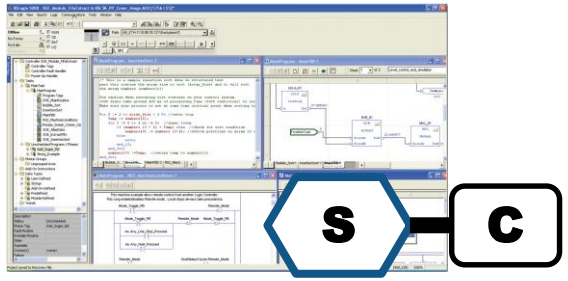
Industrial Controllers

Test



Equipment/Prototype

Design

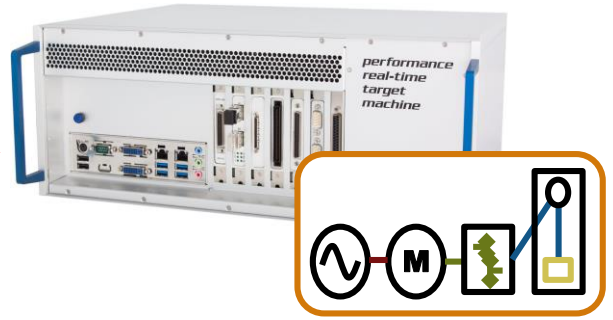


PLC Development Environment

Implement

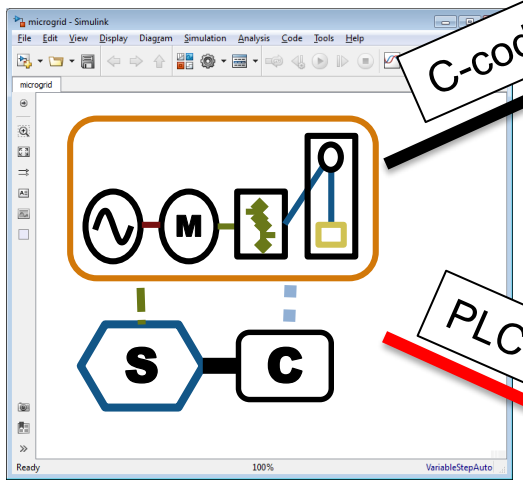
Industrial Controllers

Test



Equipment Simulation

Design



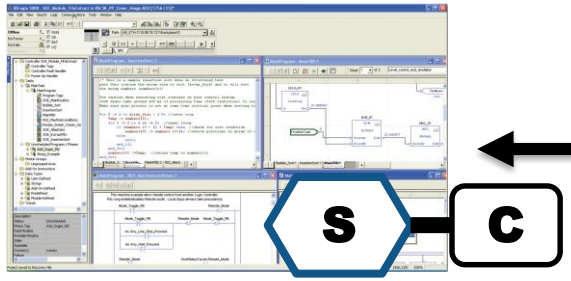
Desktop Simulation

C-code Generation

PLC Code Generation

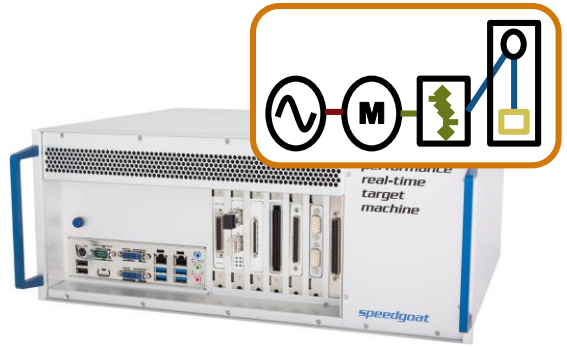
Implement

Real-time Interface



PLC Development Environment

Test



Equipment Simulation

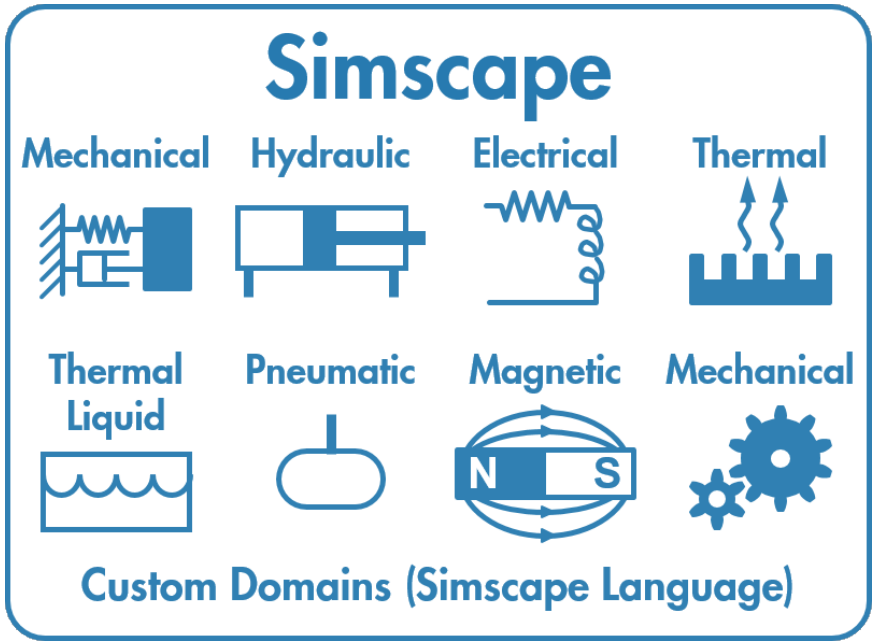
Bus

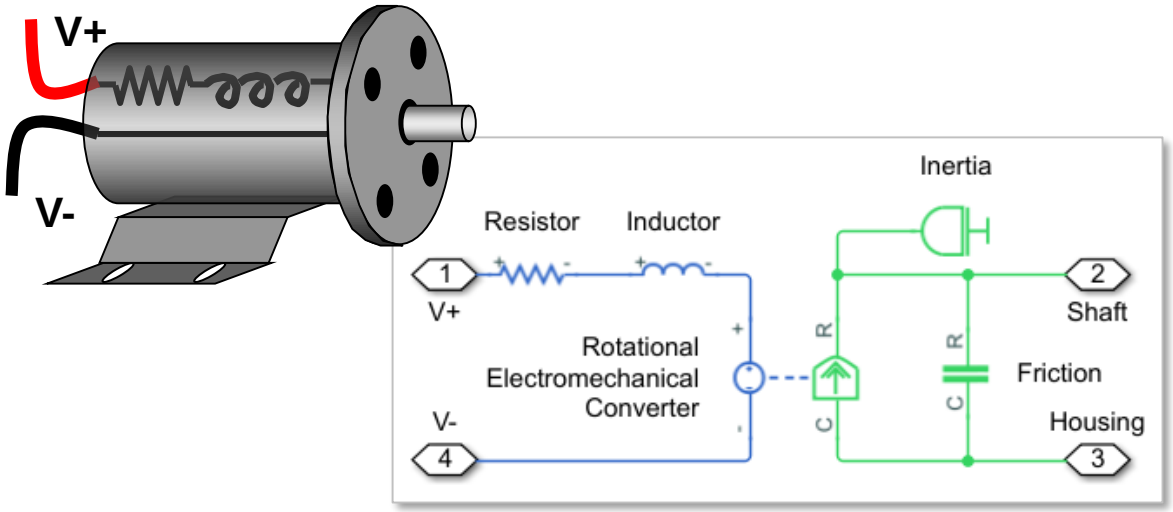
Industrial Controller

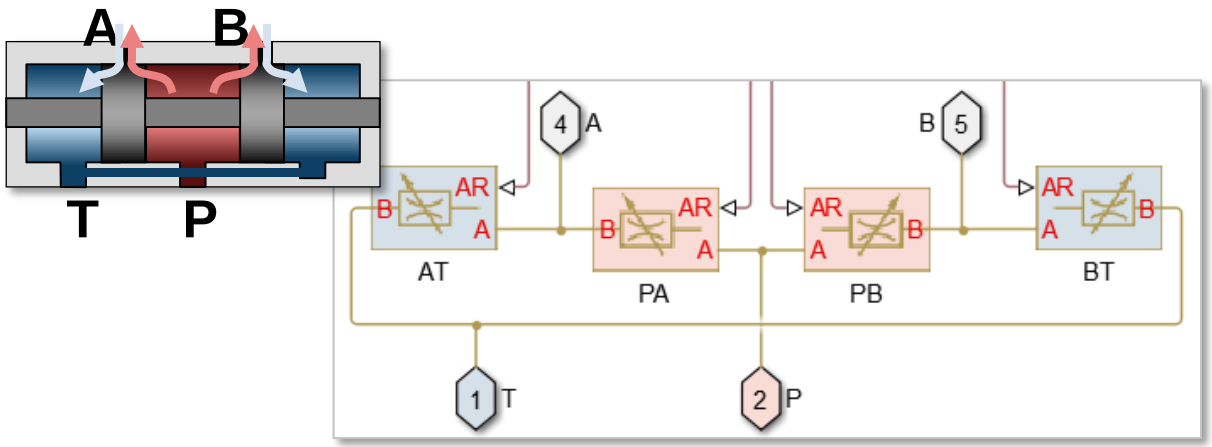
Agenda

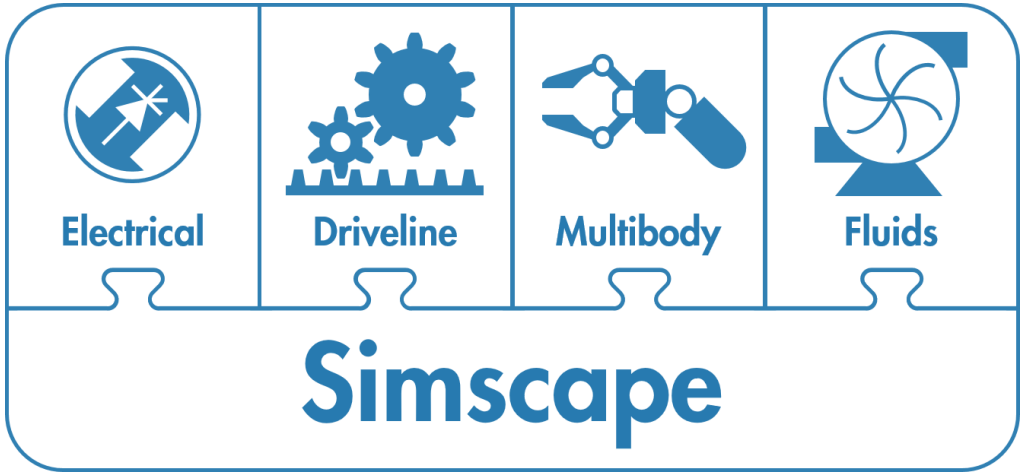
- Introduction to Modeling with Simulink and Simscape
- Leveraging Models
  - Calibrating Digital Twin with Field Data
  - Sizing Systems using Operation Data
  - Digital Twins for Anomaly Detection
- Optimizing and Validation of Controls
  - Optimizing Drilling Operations
  - Importing Existing PLC Ladders and C Software
  - Controller Error Checking and Test Vectors
- Hardware-in-the-loop Testing of PLCs

Introduction to Simulink and Simscape

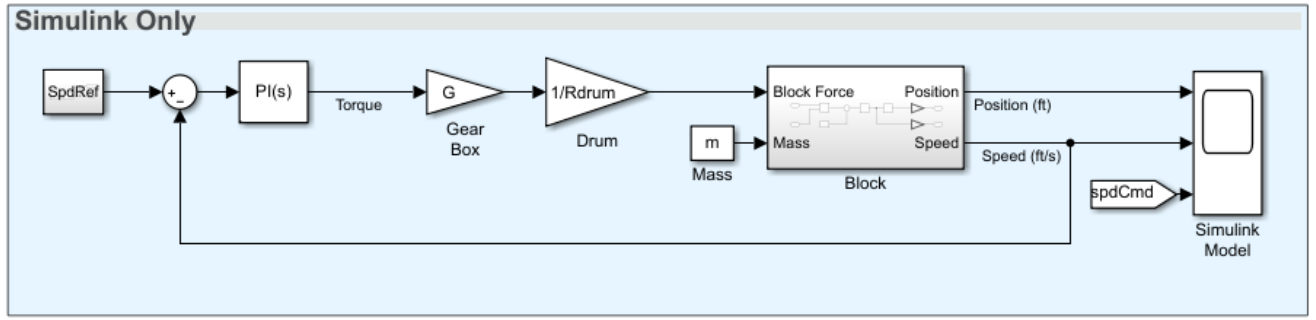


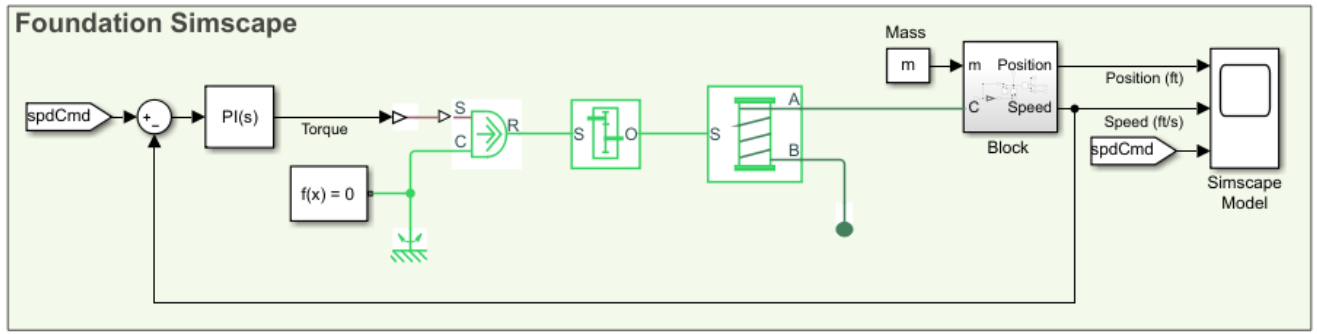


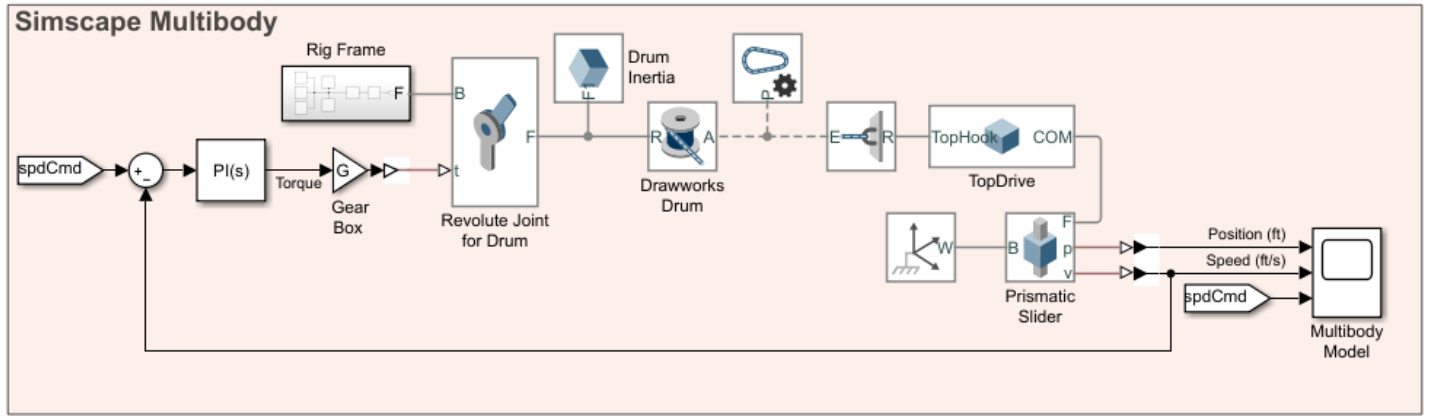




Introduction to Simulink and Simscape



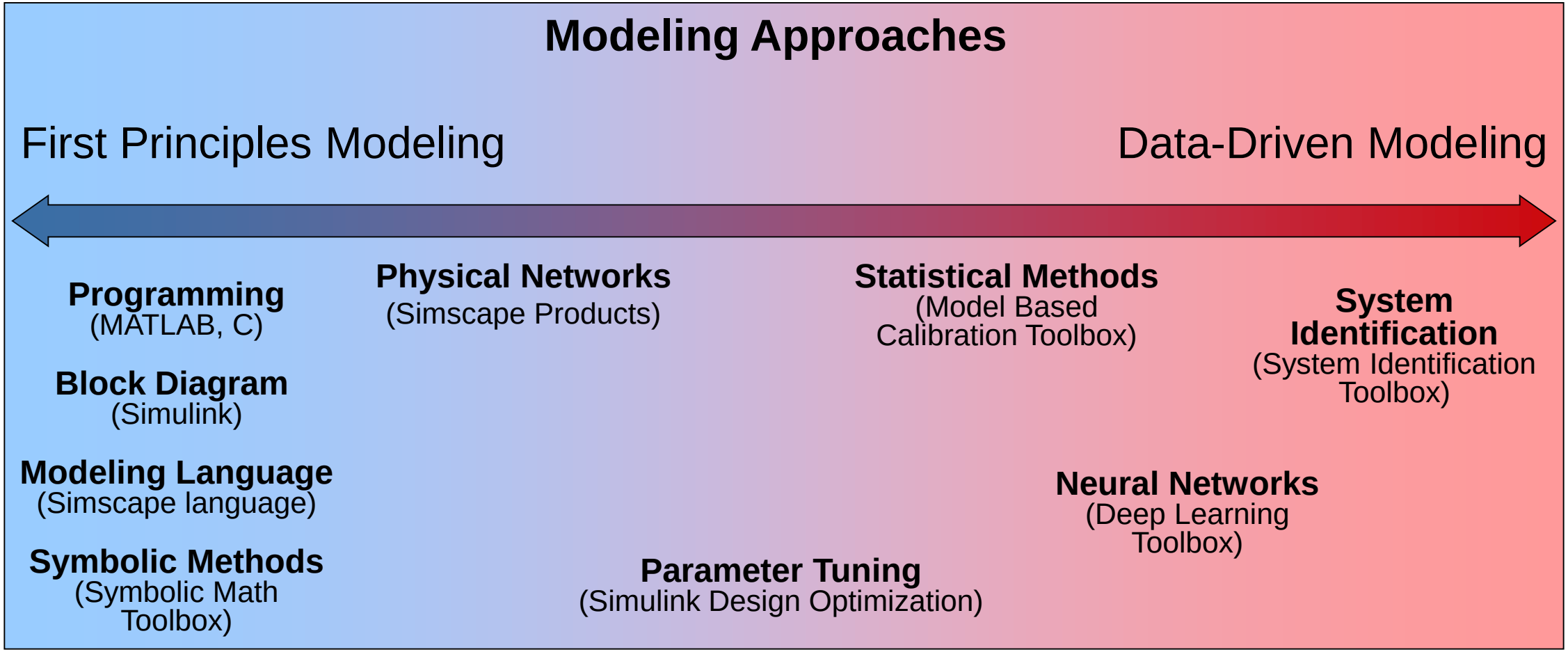




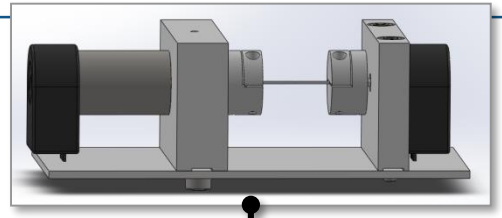
Agenda

- Introduction to Modeling with Simulink and Simscape
- Leveraging Models
  - Calibrating Digital Twin with Field Data
  - Sizing Systems using Operation Data
  - Digital Twins for Anomaly Detection
- Optimizing and Validation of Controls
  - Optimizing Drilling Operations
  - Importing Existing PLC Ladders and C Software
  - Controller Error Checking and Test Vectors
- Hardware-in-the-loop Testing of PLCs

Modeling Physical Systems with MathWorks Products



Modeling Approaches



First Principles

Data-Driven

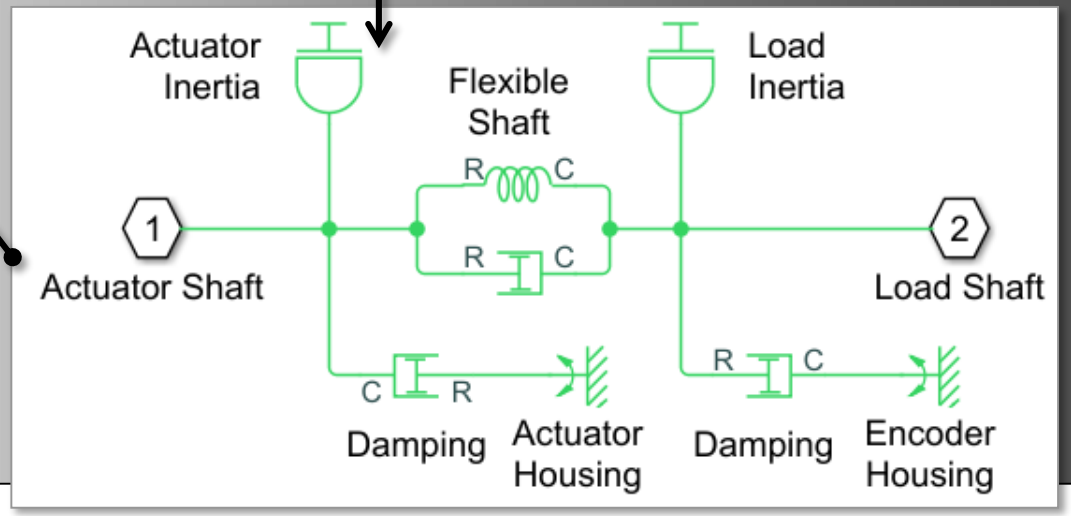
Programming

Block Diagram

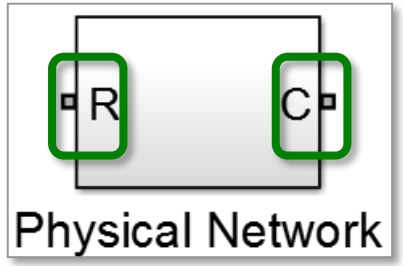
Modeling Language

Symbolic Methods

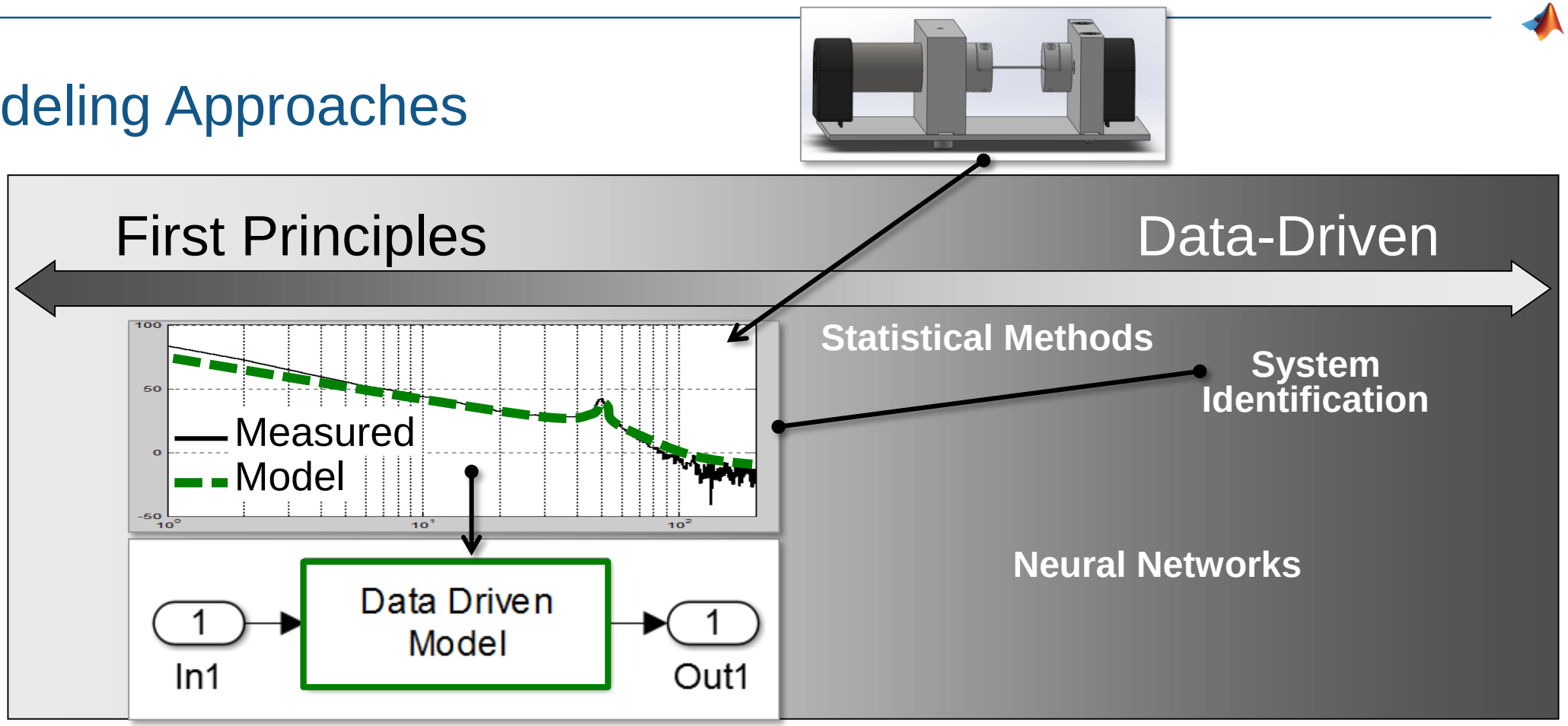
Physical Networks



- Purpose: Explore design or physical parameters
- Requirements:
  - Physics of system are well-known
  - Component-level models exist or can be created

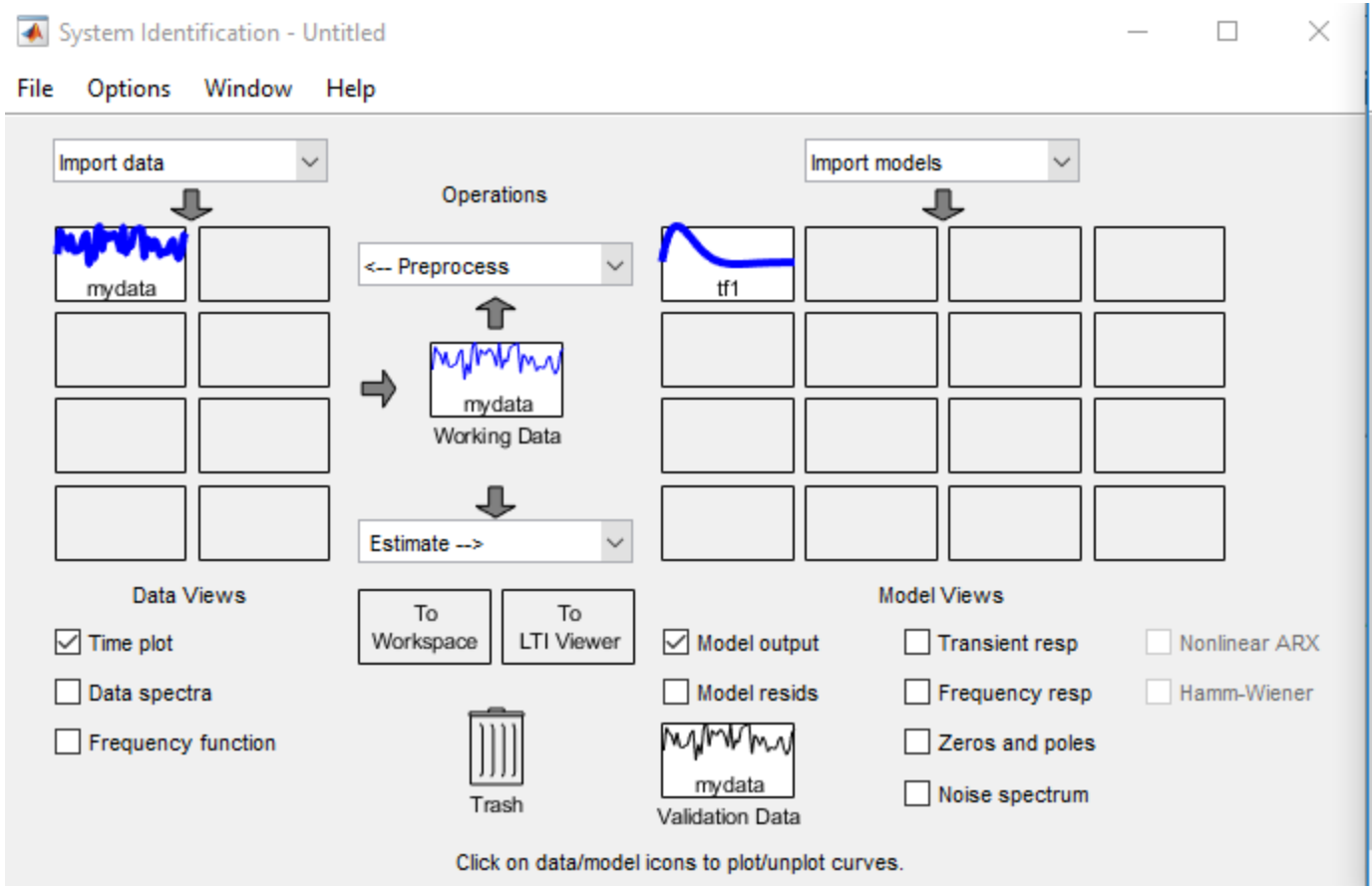


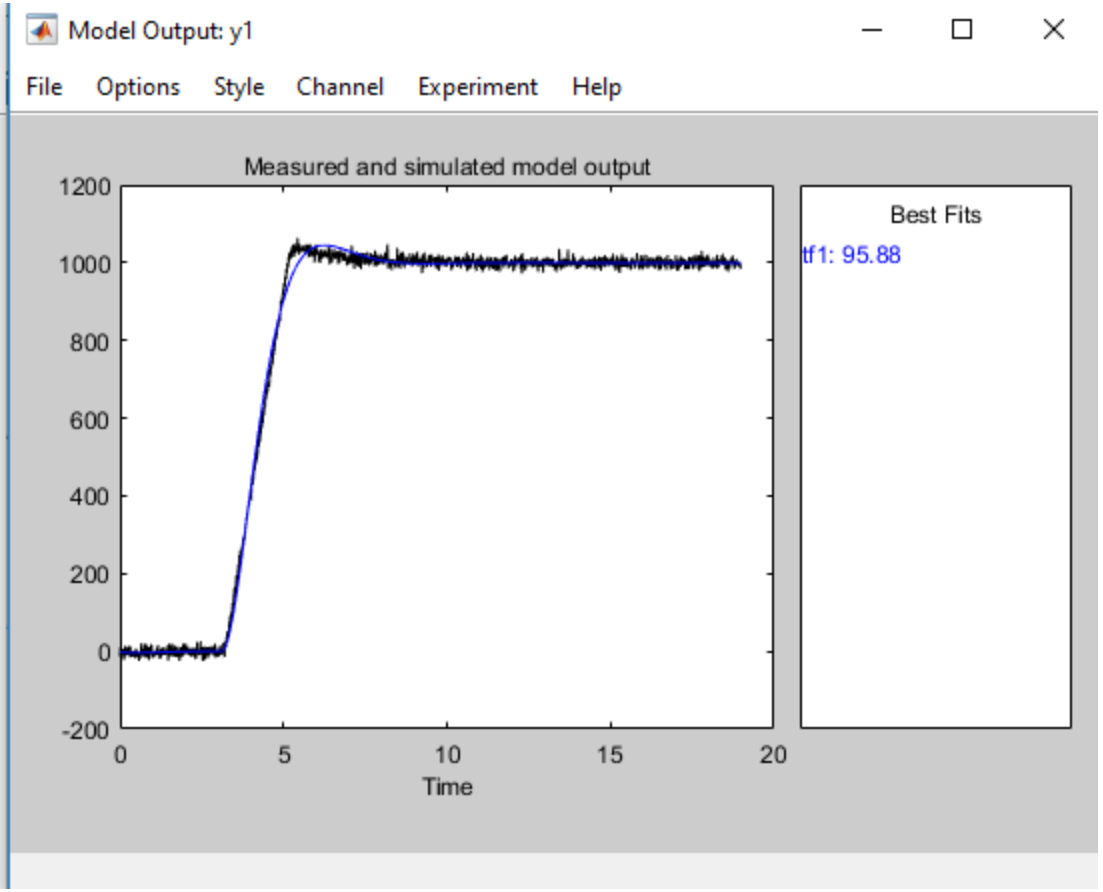
Modeling Approaches



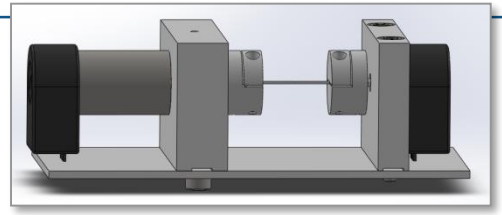
- Purpose: Model an existing design (real or virtual)
- Requirements:
  - Relevant set of measured data is available
  - Design and physical parameters will not be changed

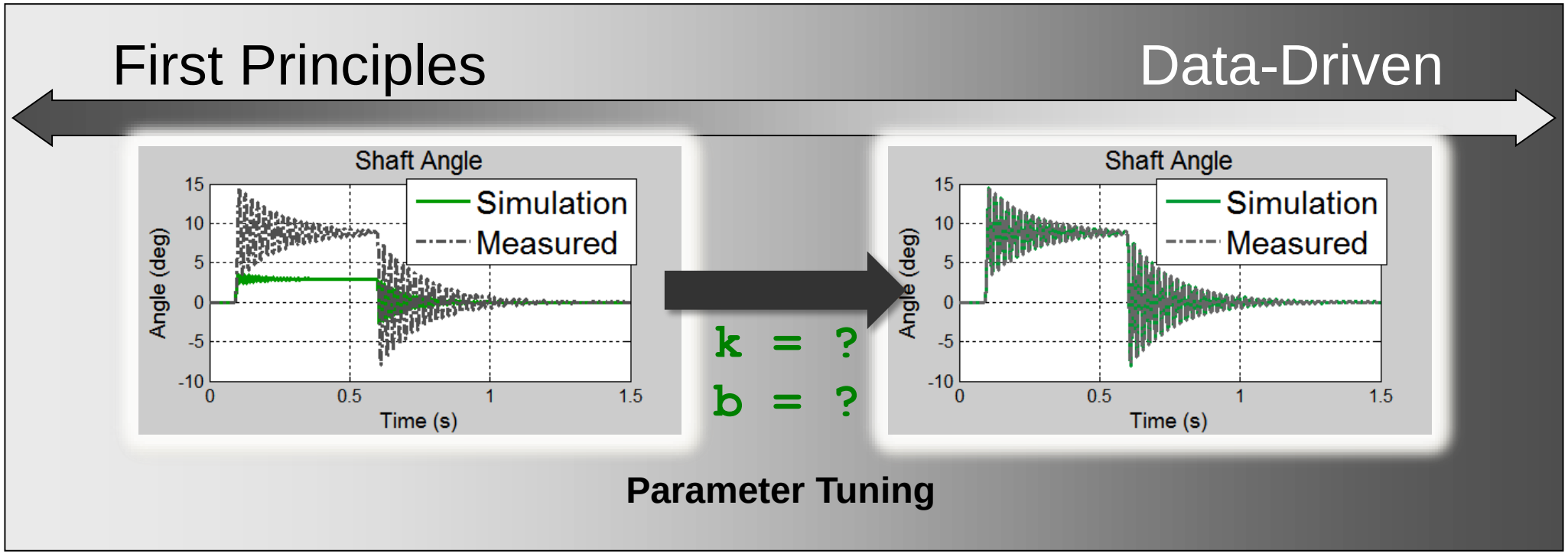
Purely Data-Driven Modeling





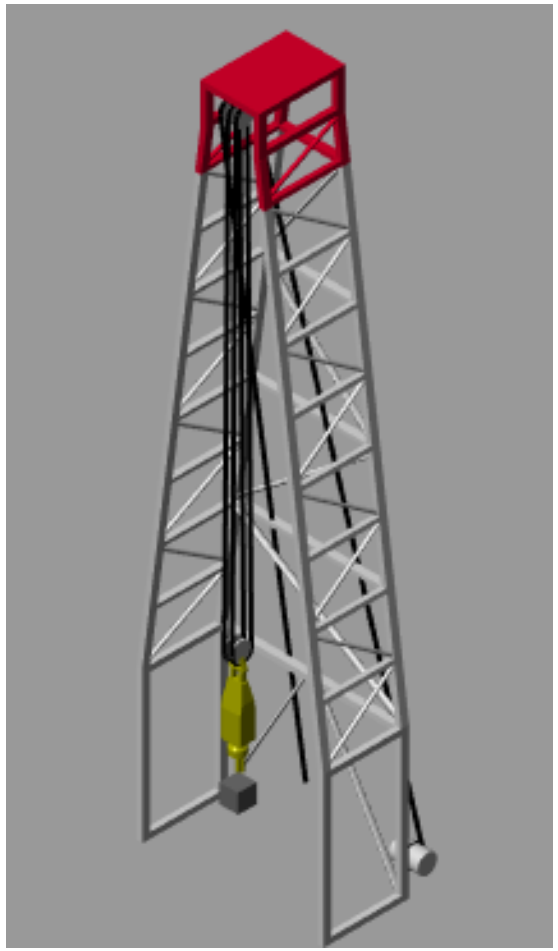
Modeling Approaches

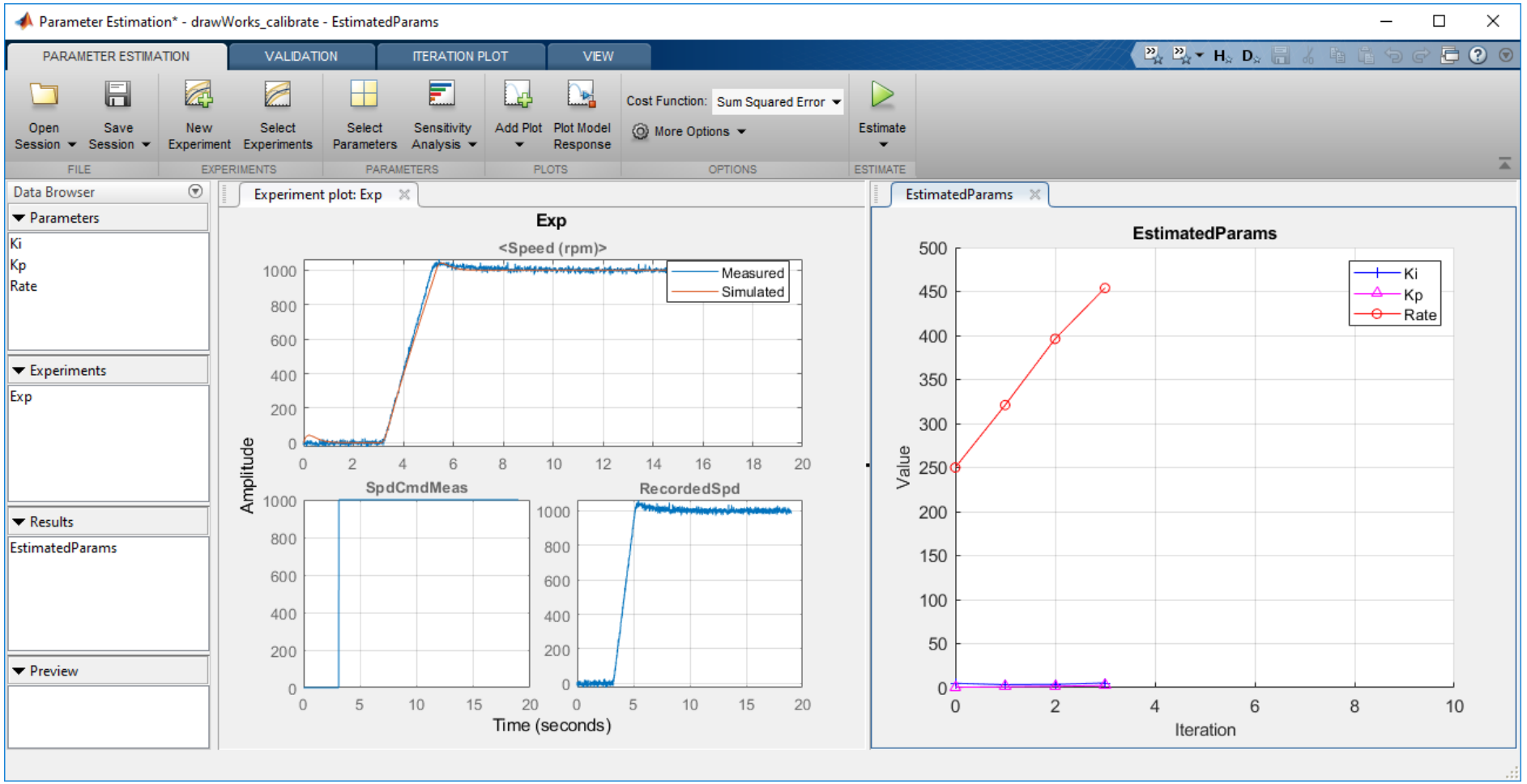




- Purpose: Ensuring parameter values are accurate
- Requirements:
  - Relevant set of measured data is available
  - Physically meaningful parameters can be automatically tuned

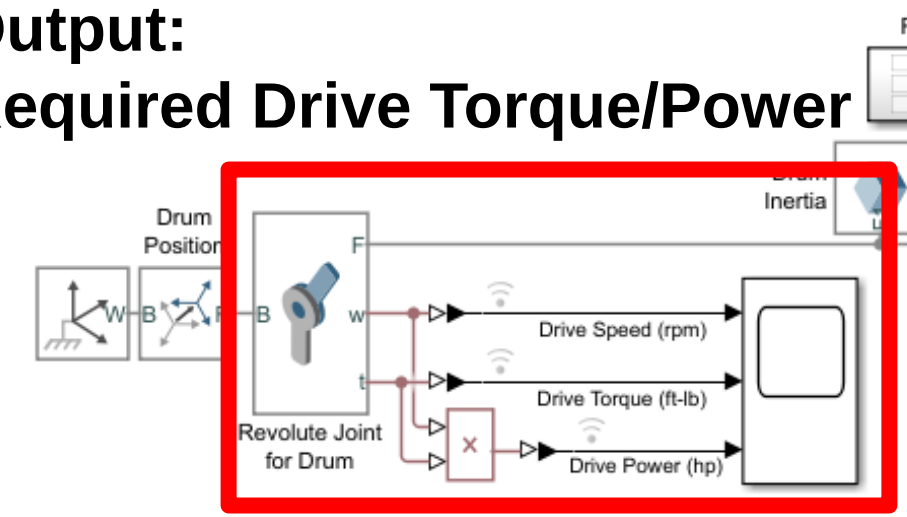
Calibrating Digital Twin – Using Field Data

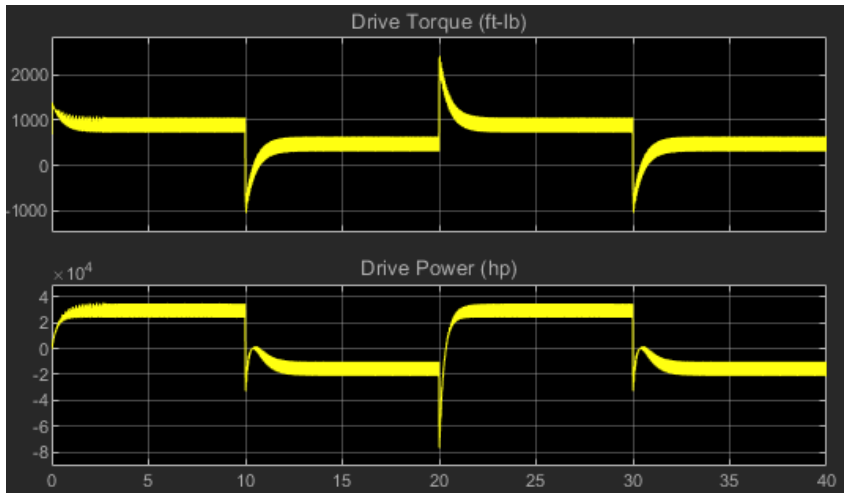


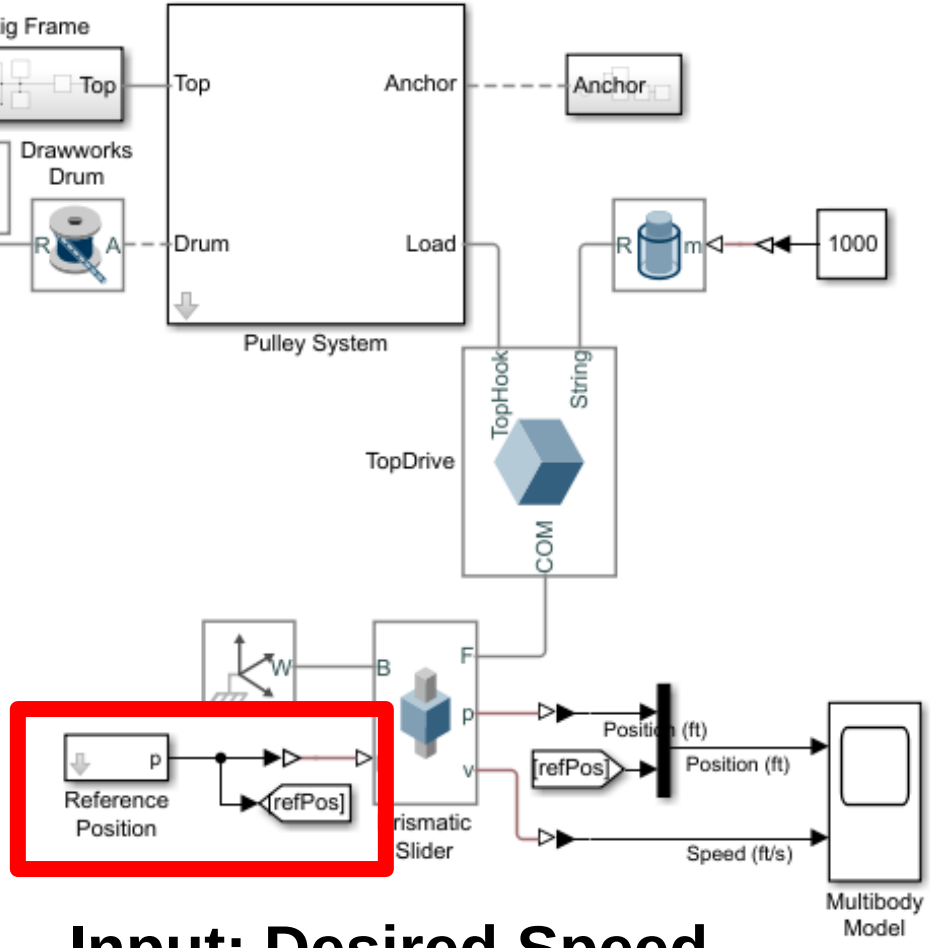


Sizing Systems using Operation Data

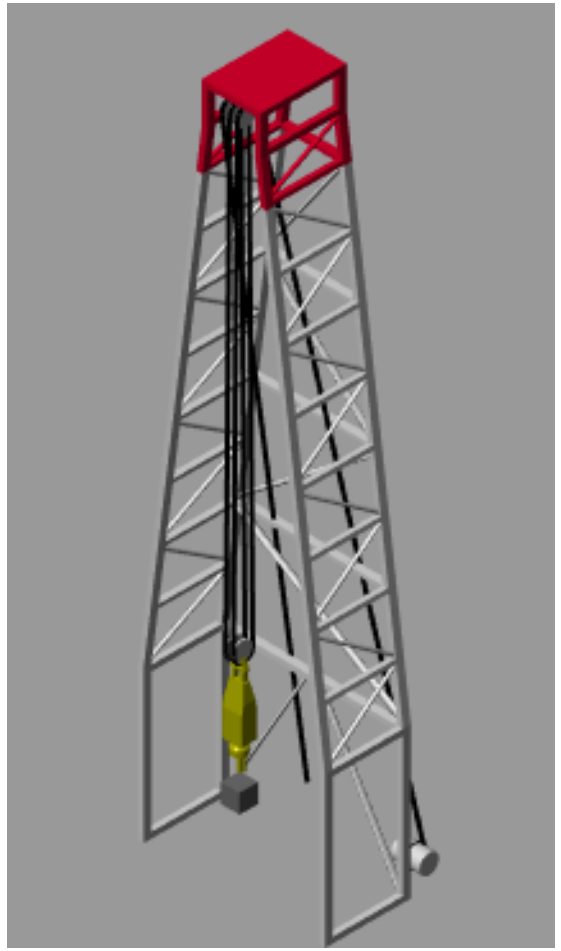
Output:
Required Drive Torque/Power



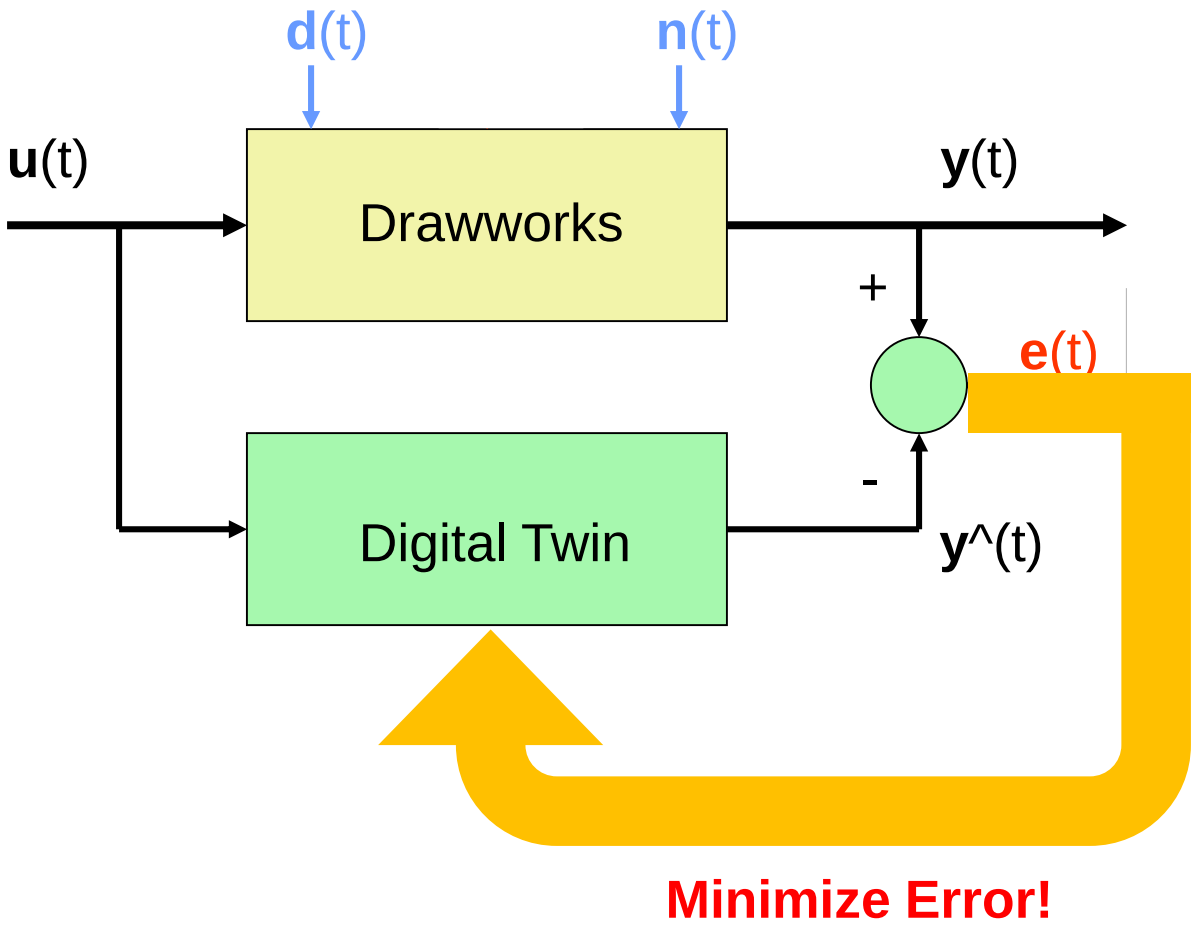


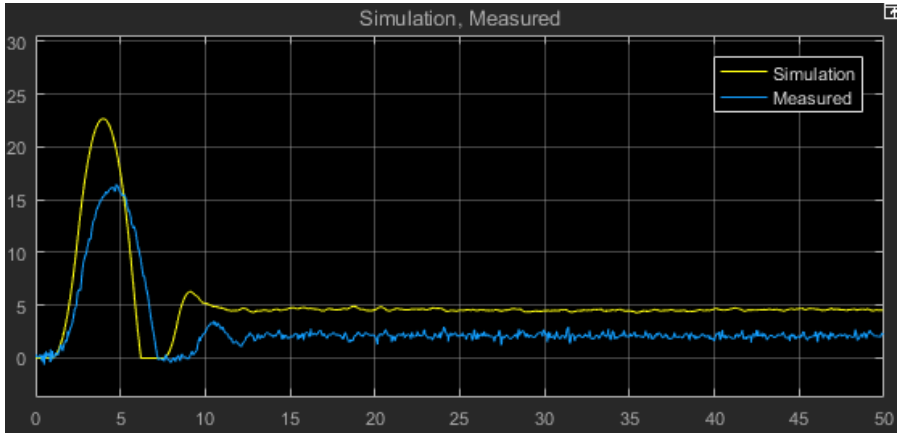


Input: Desired Speed

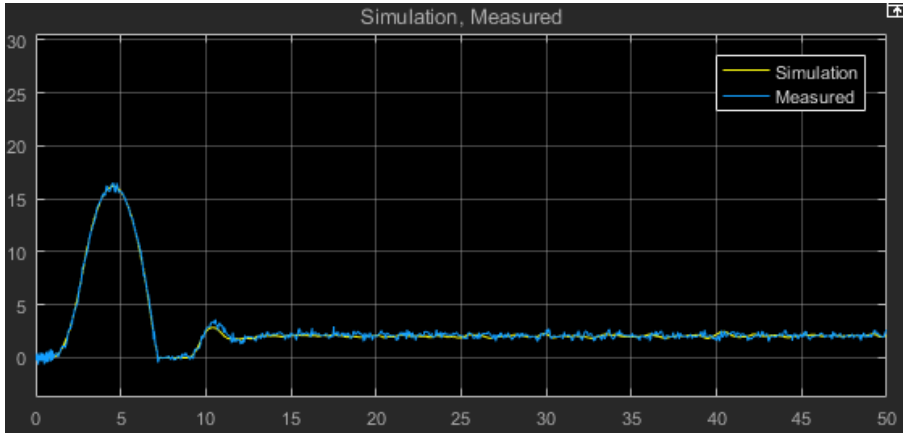


Use “Digital Twin” for Fault Detection

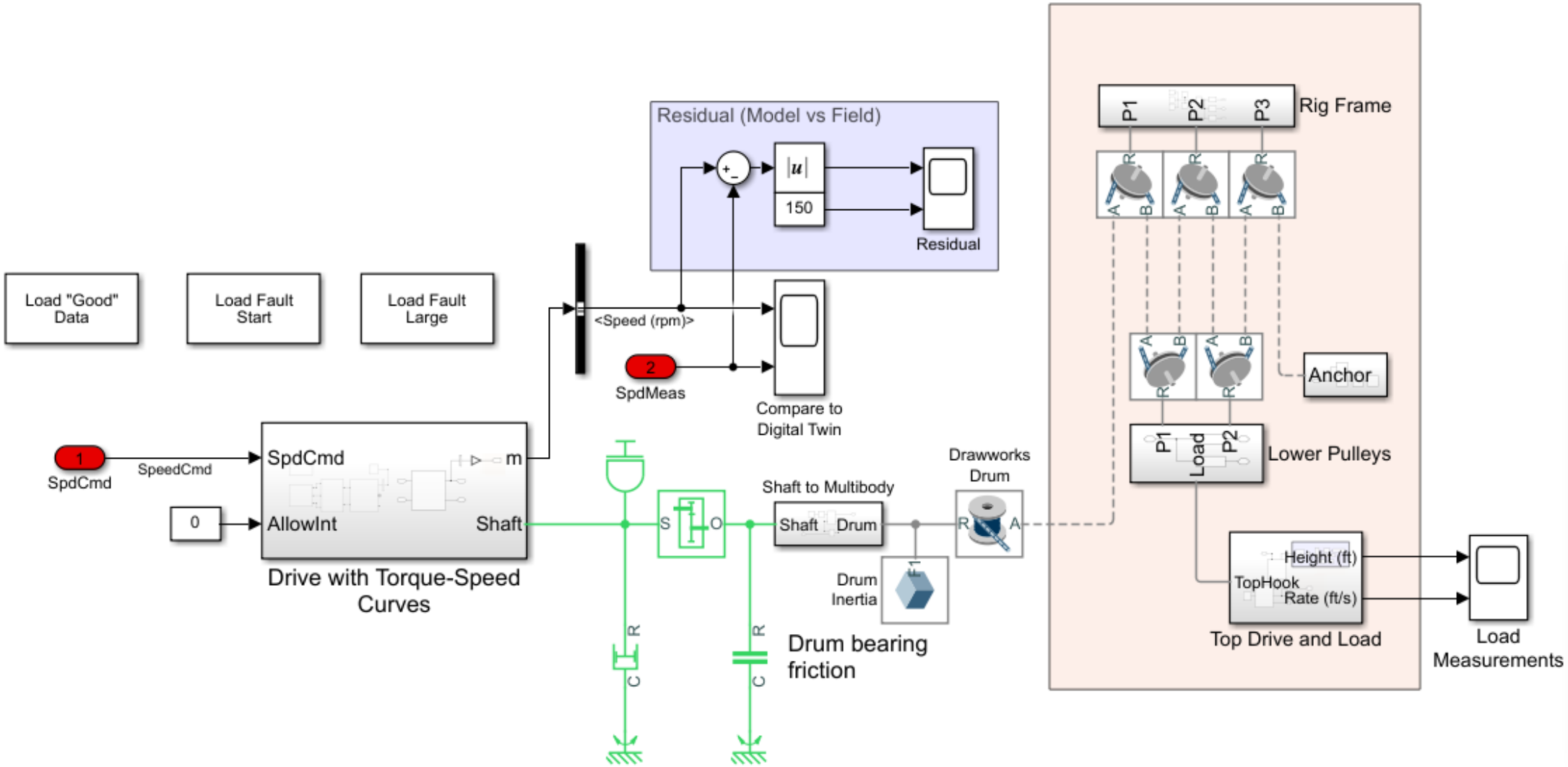




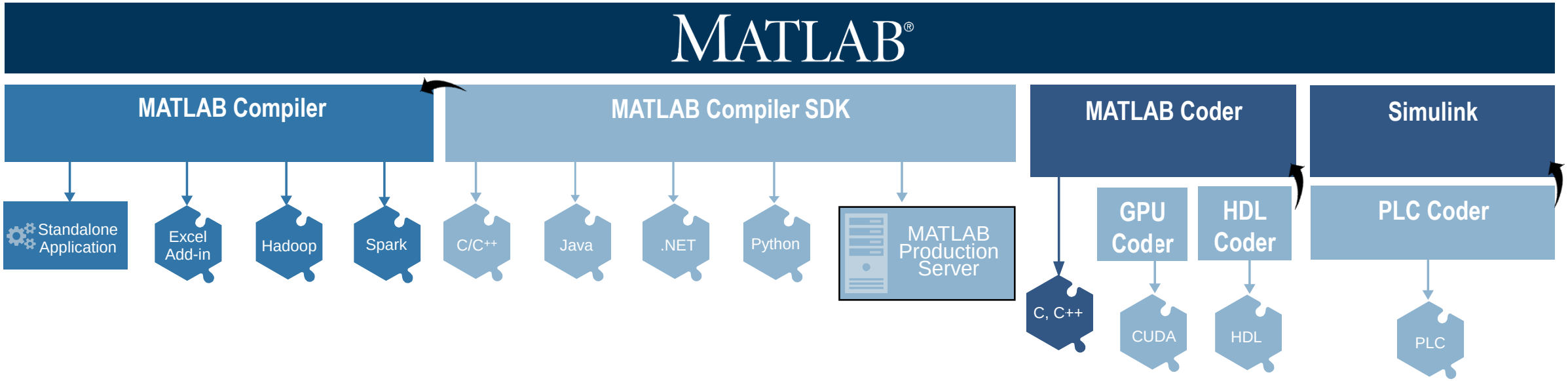
Optimization



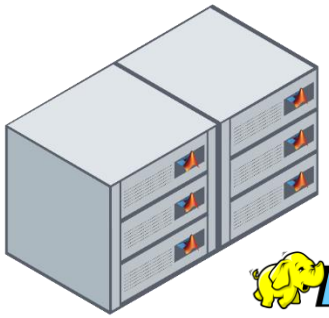
Fault Detection Application of Digital Twin



Platform Architectural Diagram – Operationalizing Analytics



Desktop Users



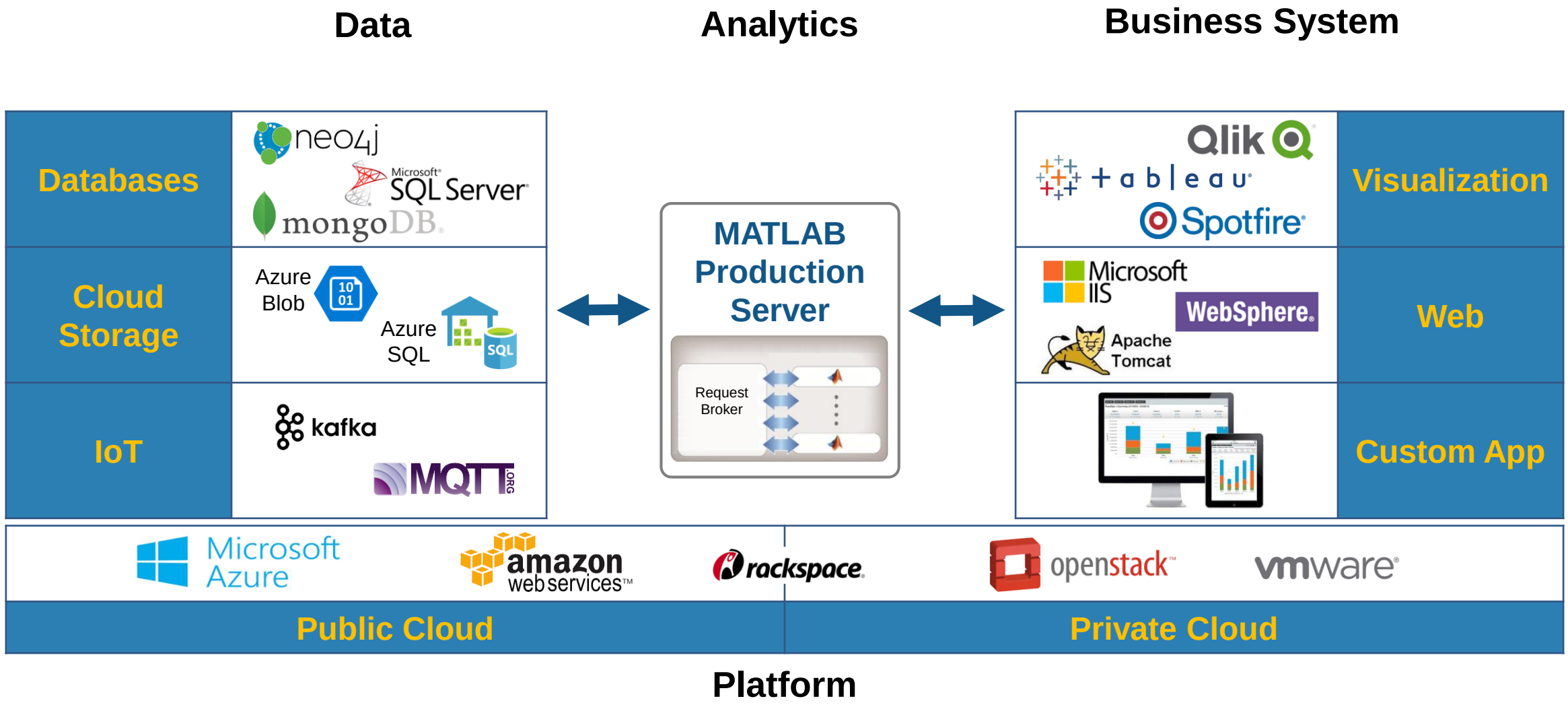
Enterprise IT Systems

Embedded Systems (Including Edge Devices)

- Microcontrollers
- NVIDIA GPUs
- DSP chips
- FPGAs
- ARM-based
- Low-cost:
  - Arduino
  - Raspberry Pi
  - BeagleBone

Denotes product dependency.

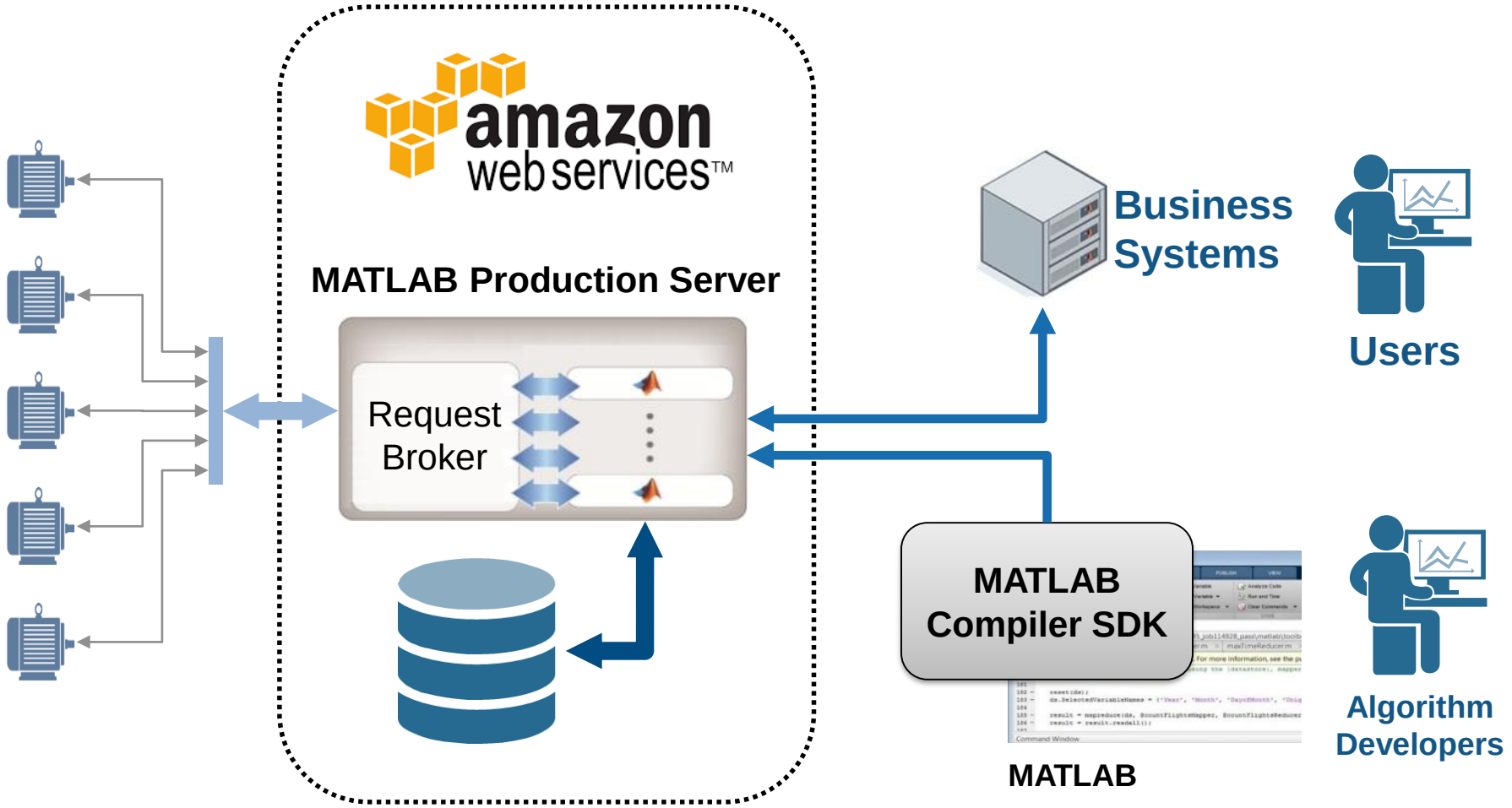
Technology Stack for Enterprise Integration



Customer Example: IoT Analytics on AWS

Industrial Air Compressors

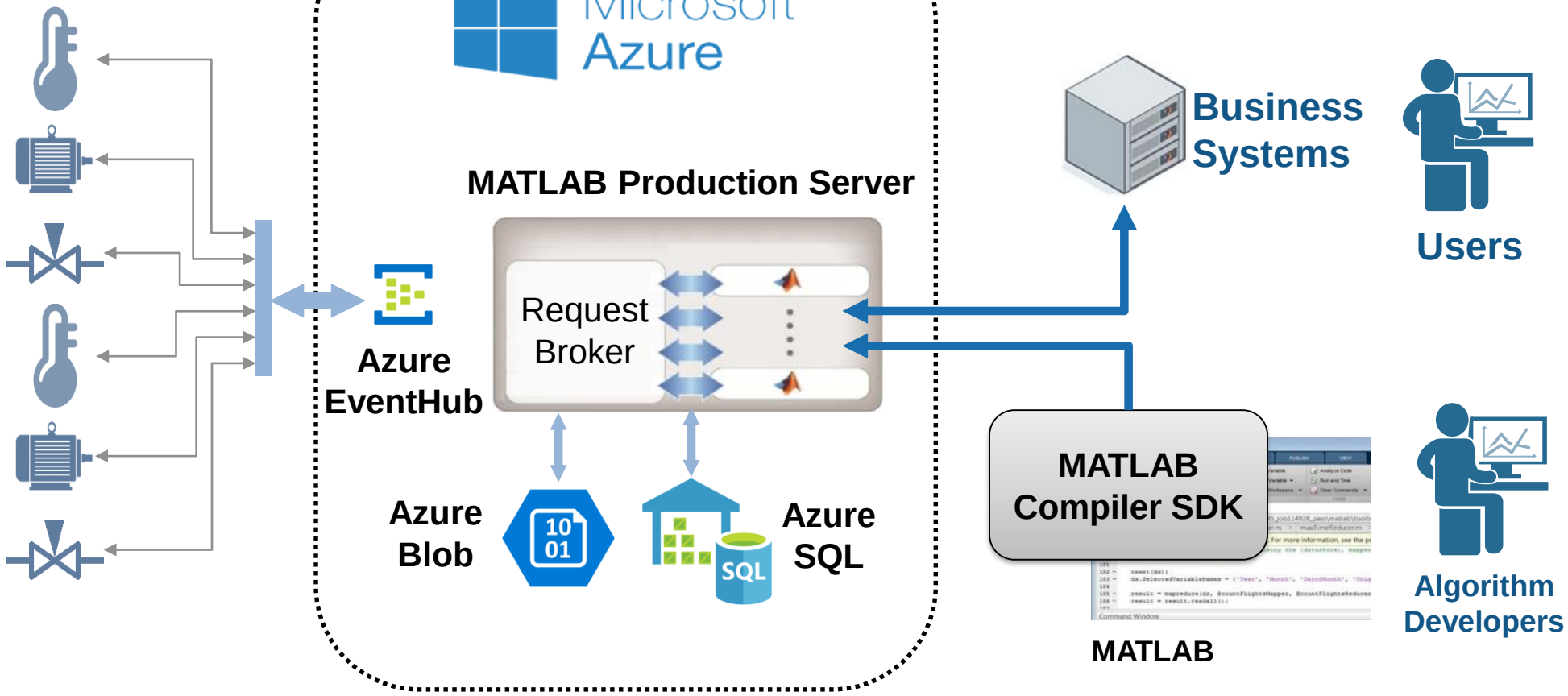
- Networked communication
- Embedded sensors
- Data reduction



Customer Example: IoT Analytics on Azure

Building/HVAC automation control system

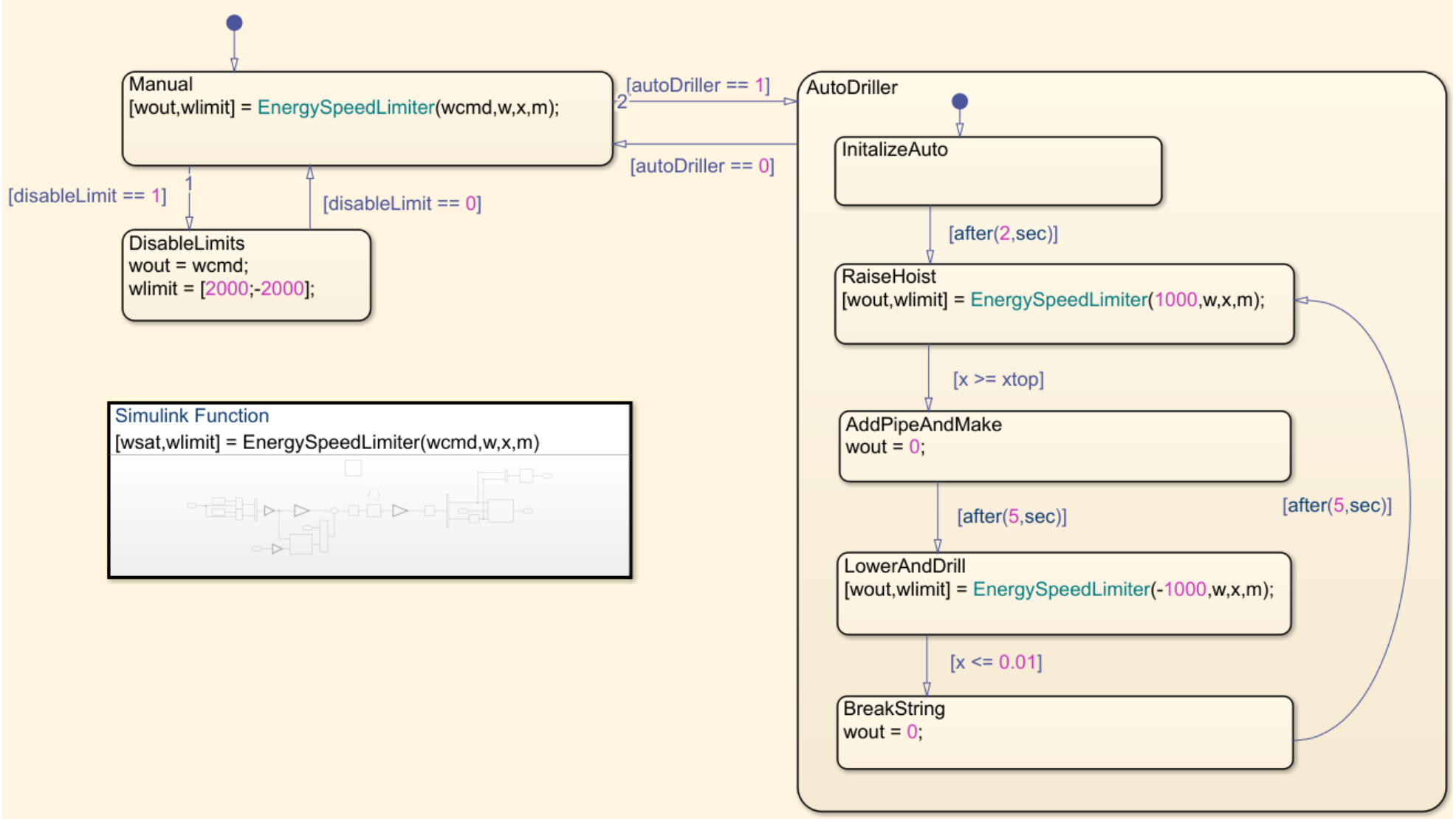
- Variety of sensors and controls
- Networked communication
- Data reduction



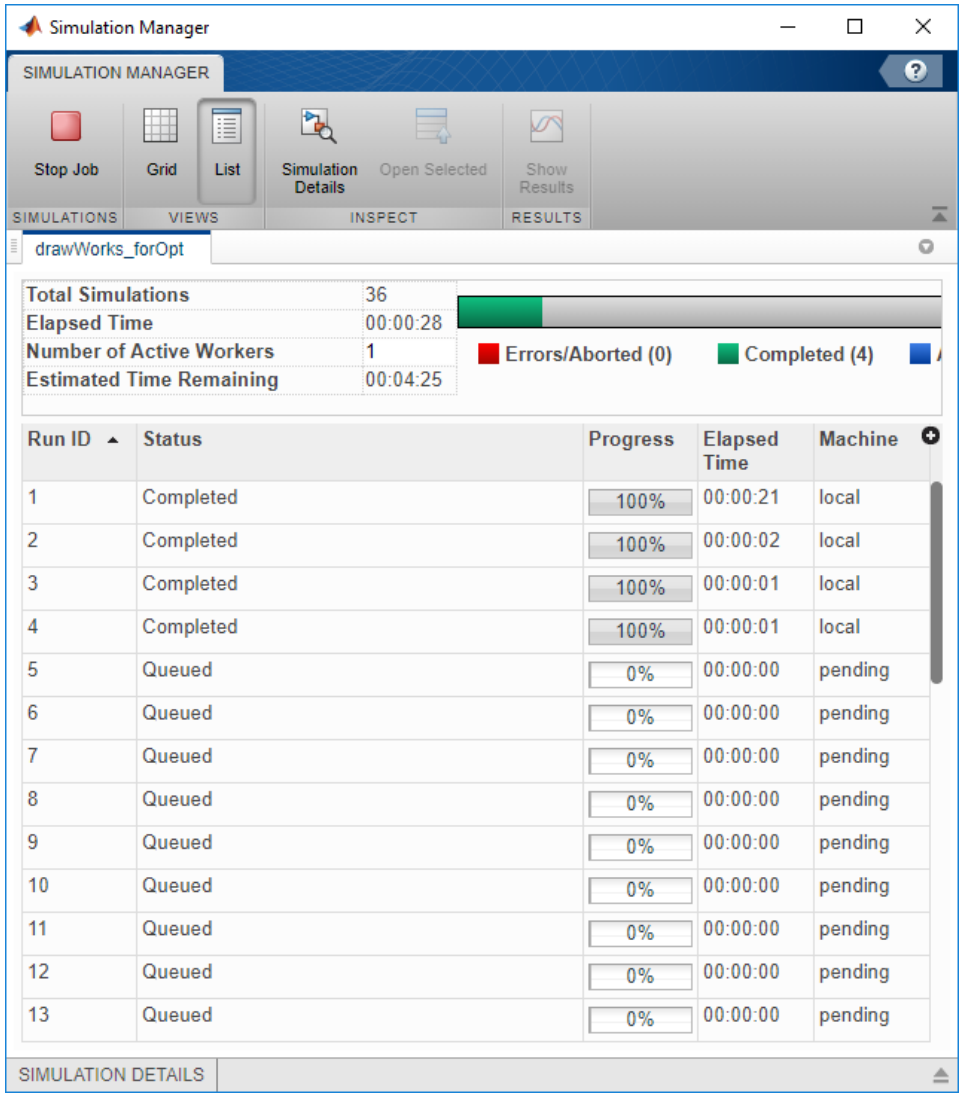
Agenda

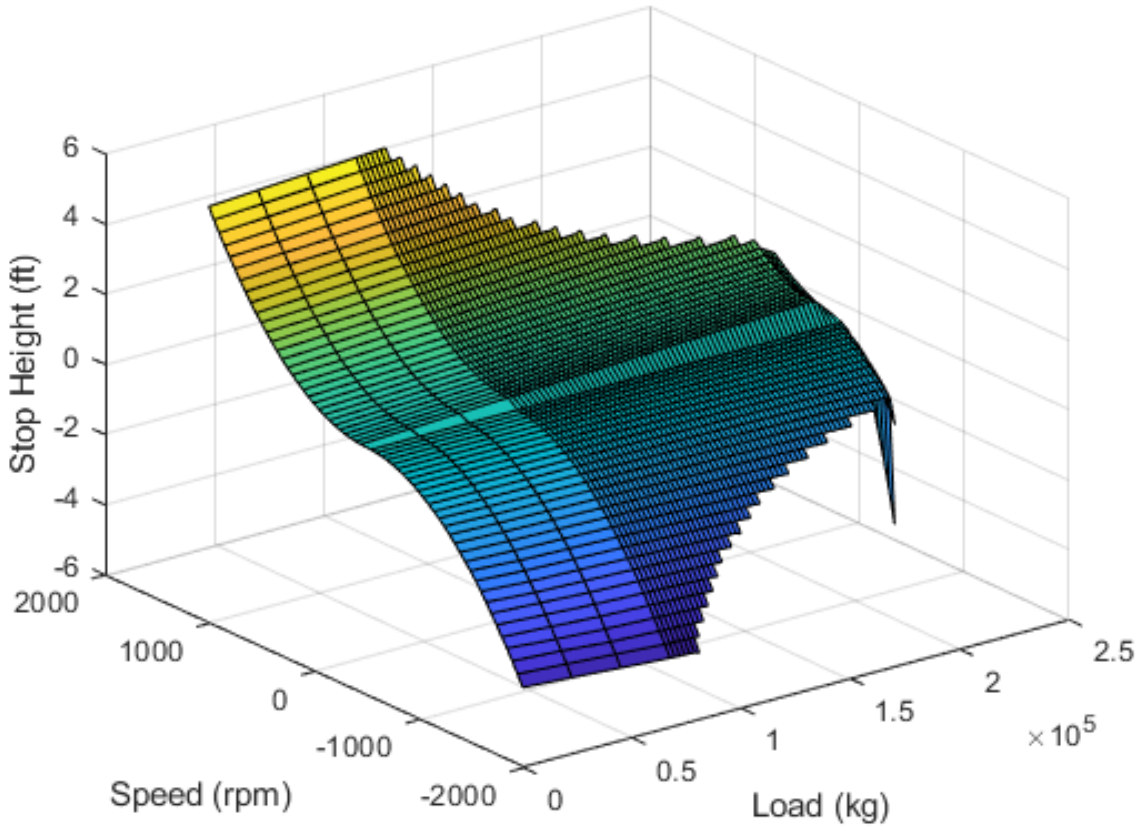
- Introduction to Modeling with Simulink and Simscape
- Leveraging Models
  - Calibrating Digital Twin with Field Data
  - Sizing Systems using Operation Data
  - Digital Twins for Anomaly Detection
- Optimizing and Validation of Controls
  - Optimizing Drilling Operations
  - Importing Existing PLC Ladders and C Software
  - Controller Error Checking and Test Vectors
- Hardware-in-the-loop Testing of PLCs

Optimizing Drilling Operations – First Principles

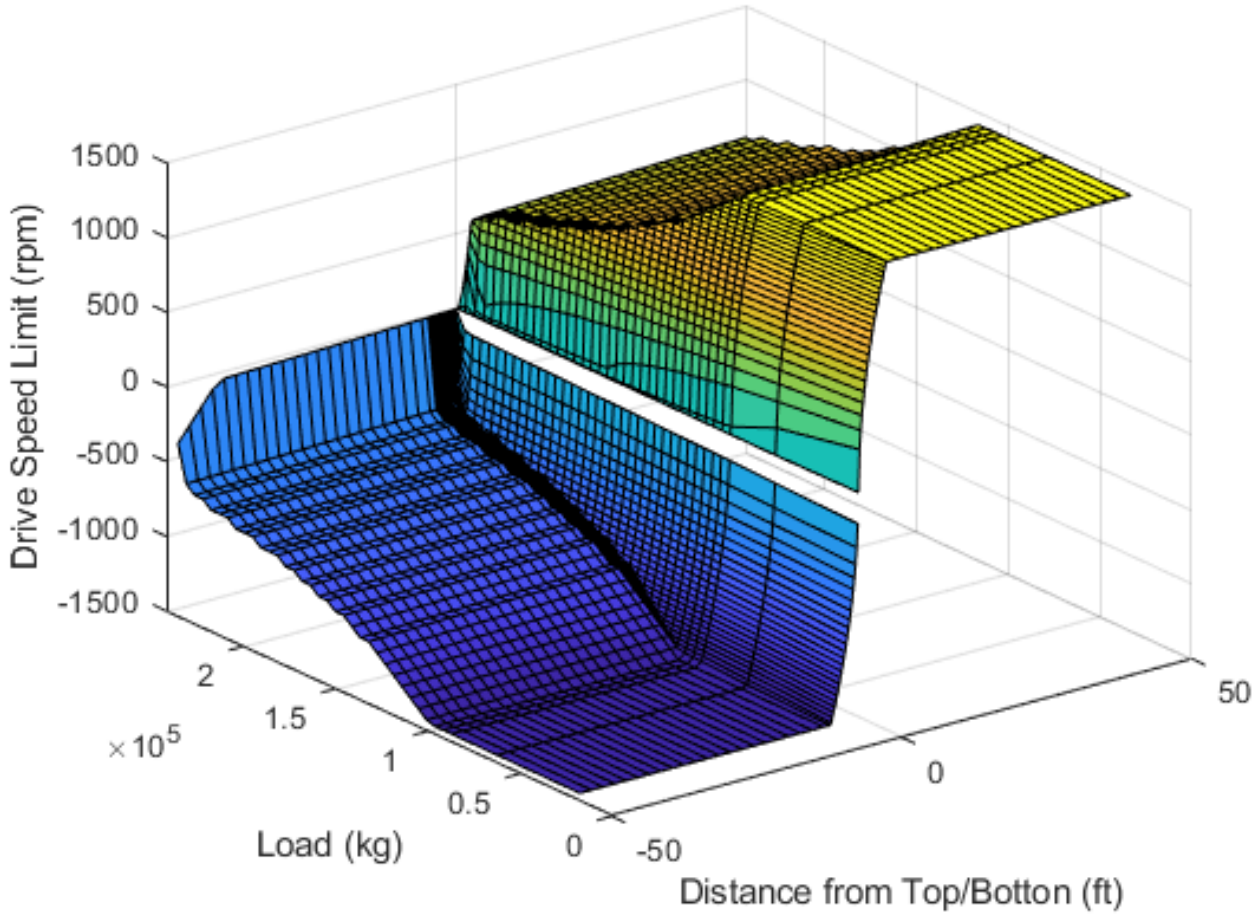


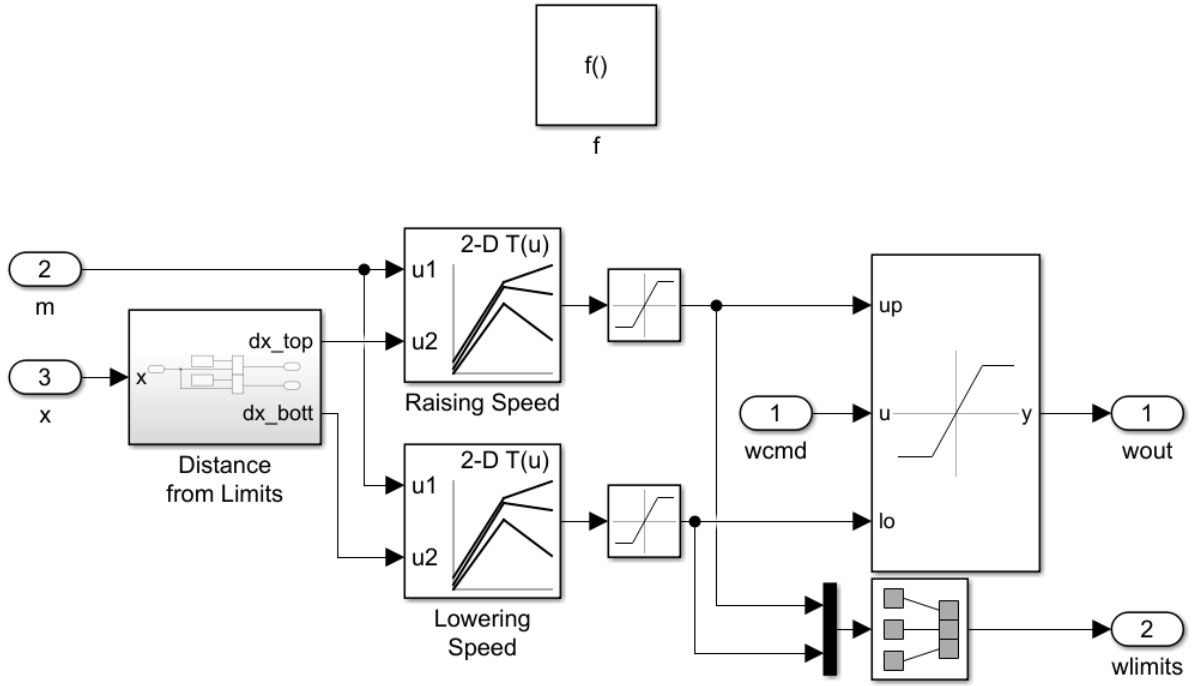
Optimizing Drilling Operations – Range of Operation Sweep



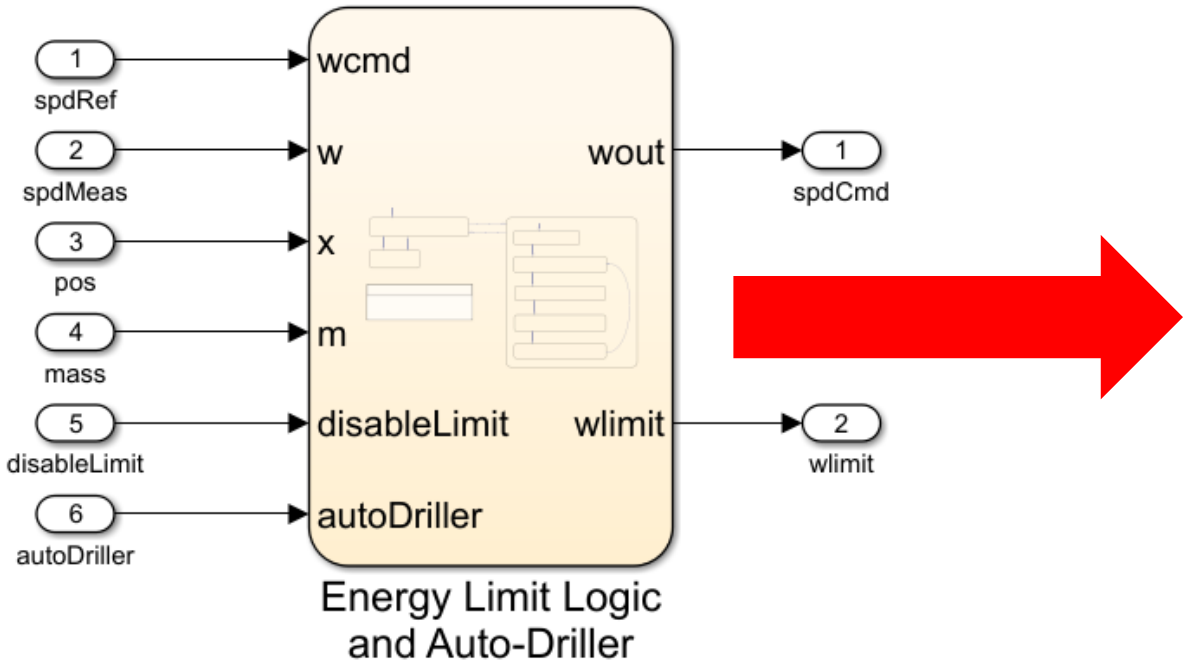


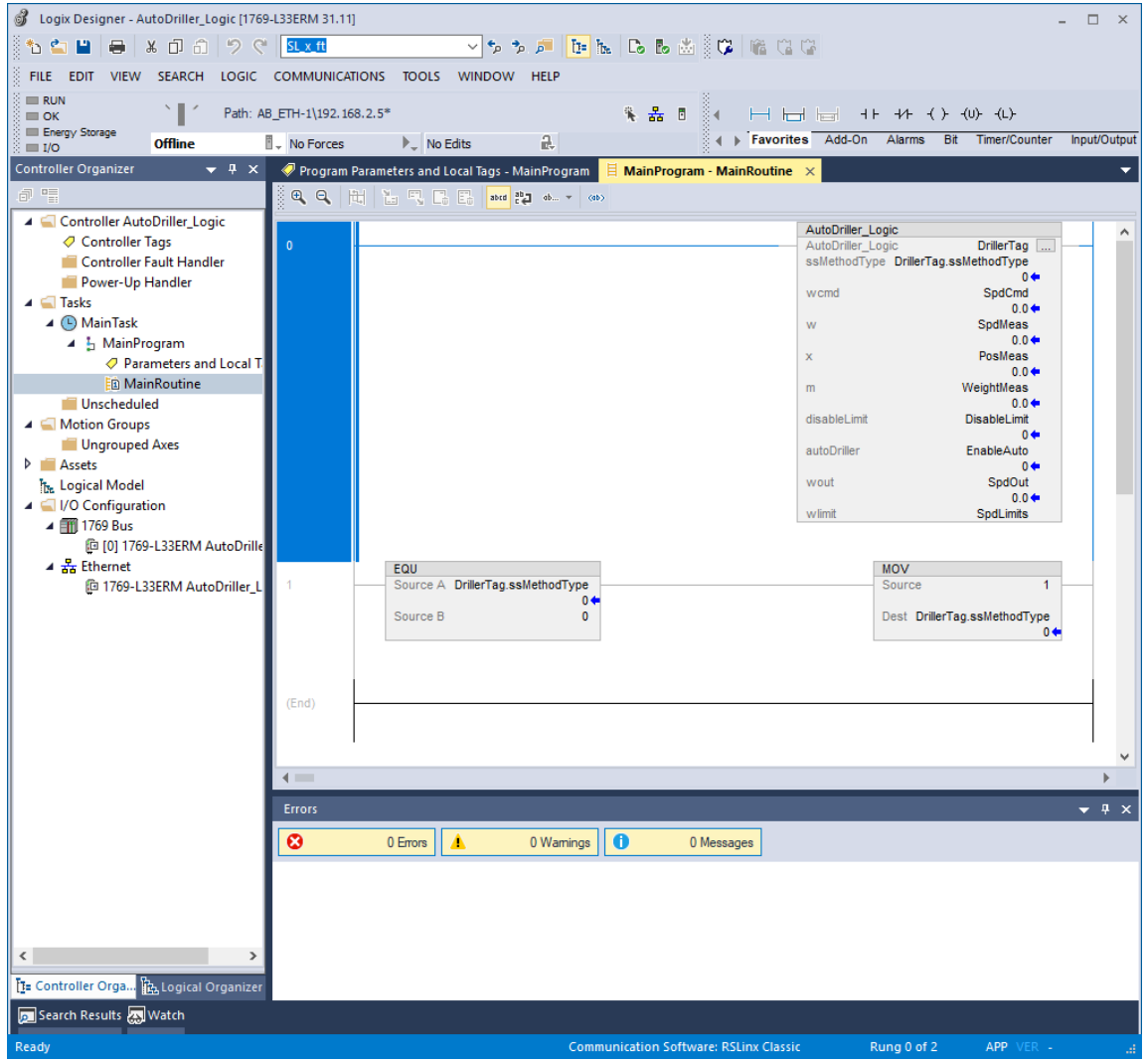
Optimizing Drilling Operations – Create Lookup Table





Optimizing Drilling Operations – Generating PLC Structured Text

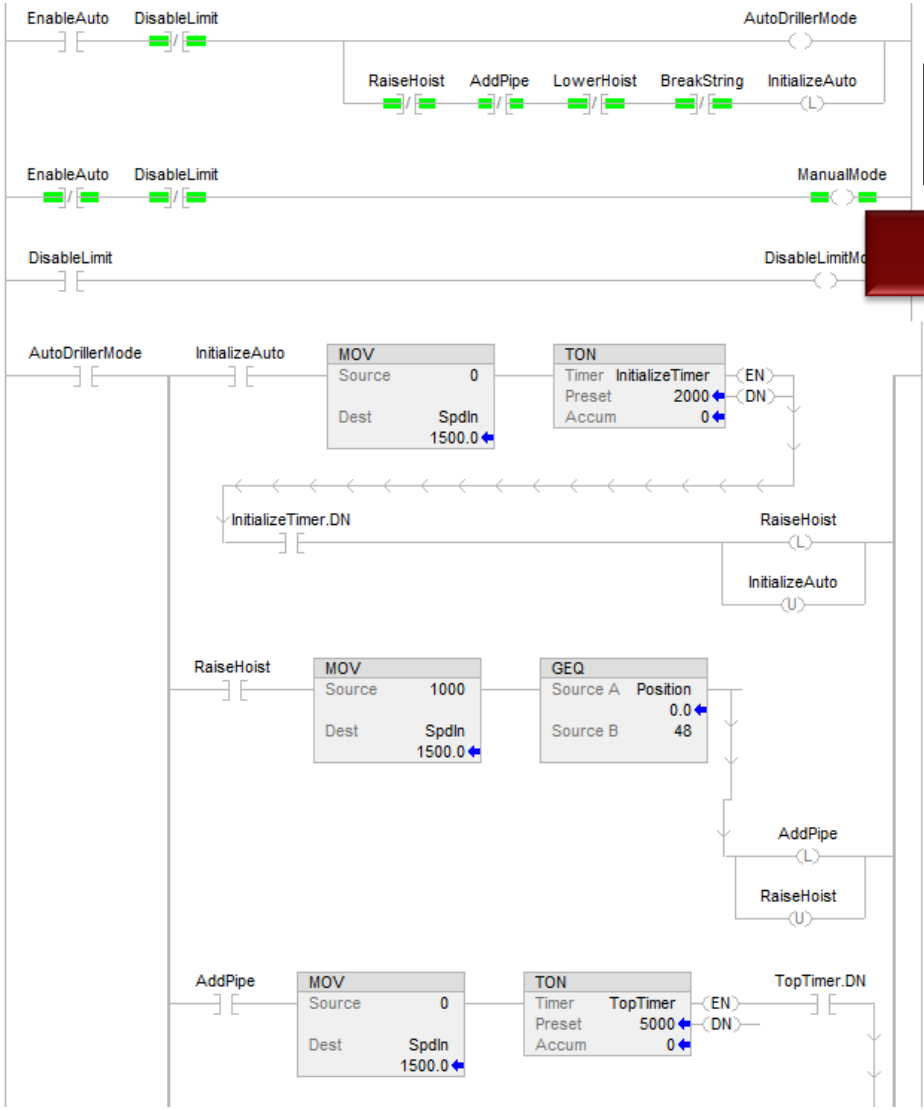




Agenda

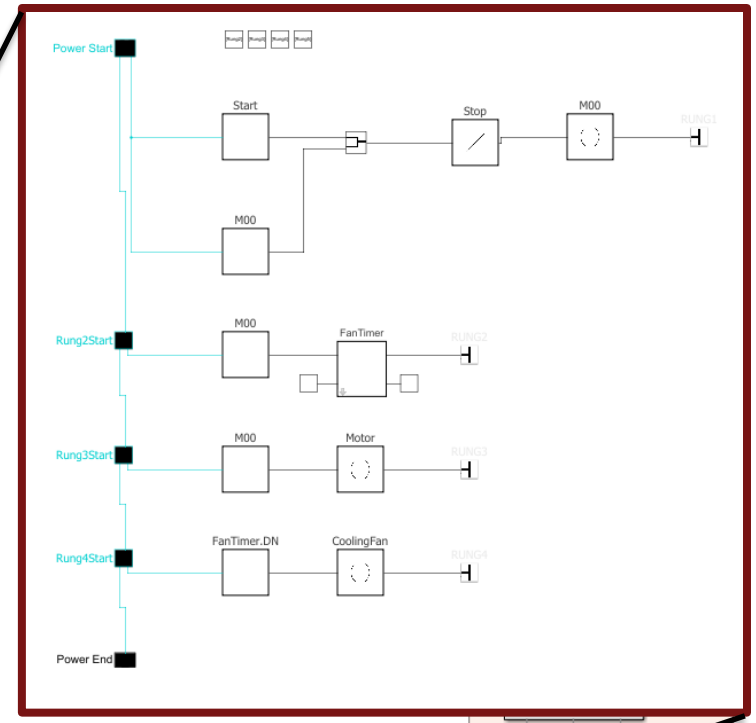
- Introduction to Modeling with Simulink and Simscape
- Leveraging Models
  - Calibrating Digital Twin with Field Data
  - Sizing Systems using Operation Data
  - Digital Twins for Anomaly Detection
- Optimizing and Validation of Controls
  - Optimizing Drilling Operations
  - Importing Existing PLC Ladders and C Software
  - Controller Error Checking and Test Vectors
- Hardware-in-the-loop Testing of PLCs

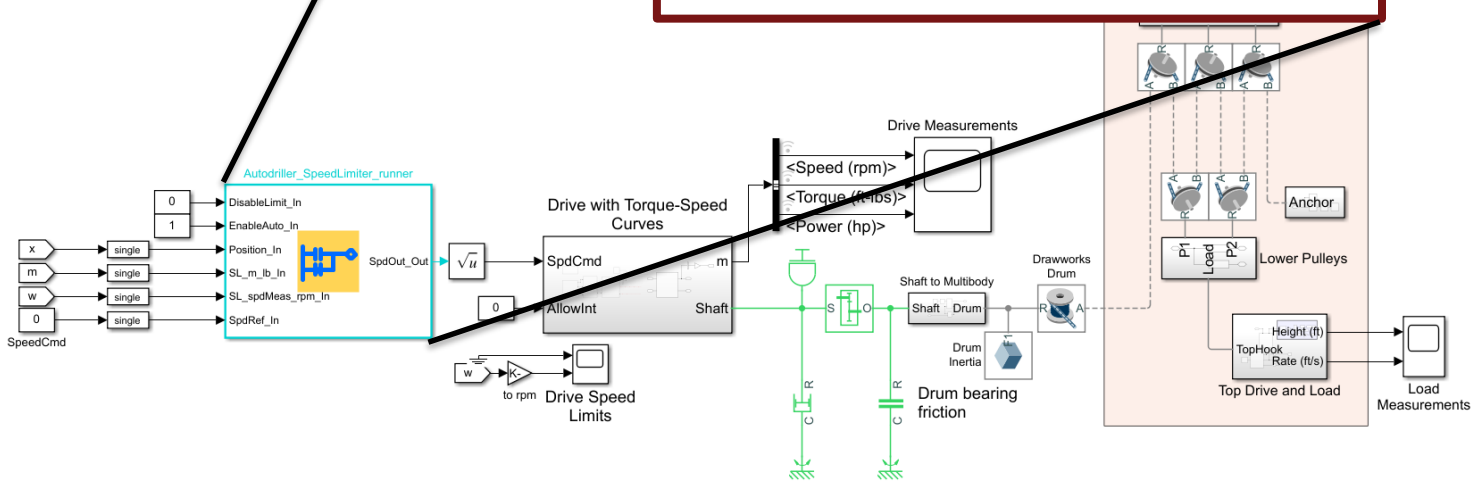
Leveraging Legacy PLC Ladders for Digital Twins



Export L5X Instruction

“plcimportladder”

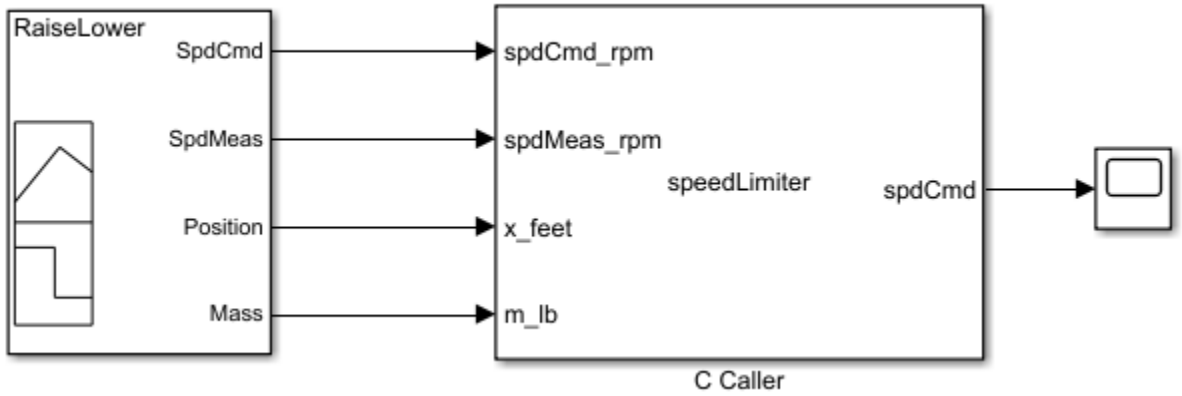


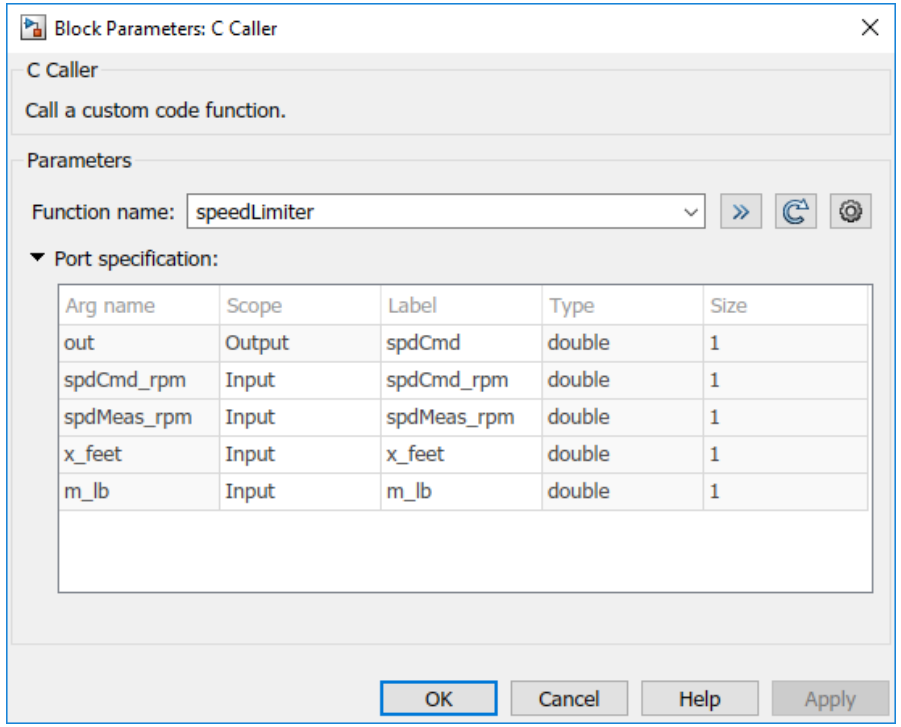


Leveraging Legacy C Software for Digital Twins

speedLimiter.c

```c
#define _USE_MATH_DEFINES
#include <math.h>
double speedLimiter(double spdCmd_rpm, double spdMeas_rpm, double x_feet, double m_lb)
{
  double xTop = 48*0.3048;  // Top Height for Block [m]
  double xBott = 0;         // Bottom Height for Block [m]

  double Pmax = 1500*746;   // Maximum Drive Power [W]
  double Tmax = 8000*1.36;  // Maximum Drive Torque [N-m]

  double Rd = 0.25;         // Drum Radius [m]
  double G = 10;            // Gear Box Ratio
  double Np = 5;            // Number of Pulleys

  double g = 9.81;          // Gravity [m/s^2]
  double Fmax, spdMax, spdMin, vMax, vMin;

  // Convert from US units to SI units
  double spdCmd = spdCmd_rpm*M_PI/30;
  double spdMeas = spdMeas_rpm*M_PI/30;
  double m = m_lb*0.45359237;
  double x = x_feet*0.3048;

  // Compute Maximum Line Force
  Fmax = (G*(Np-1)/Rd)*fmin(Tmax,Pmax/fabs(spdMeas));

  // Maximum Block Velocity
  vMax = 2*g*(x-xTop) - 2*(x-xTop)*Fmax/m;
  if (vMax < 0)
  {
    vMax = 0;
  }
  vMax = sqrt(vMax);

  // Minimum Block Velocity
  vMin = 2*g*(xBott-x) - 2*(xBott-x)*Fmax/m;
  if (vMin < 0)
  {
```

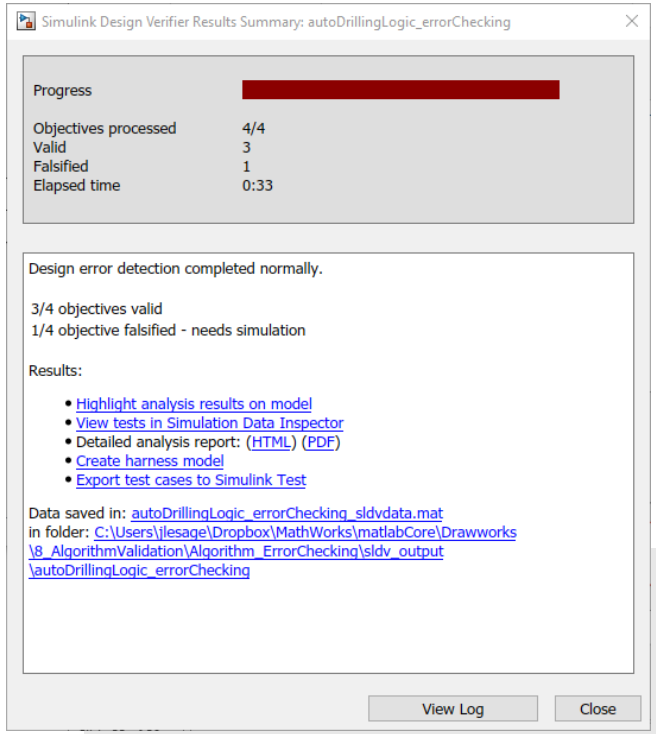




Agenda

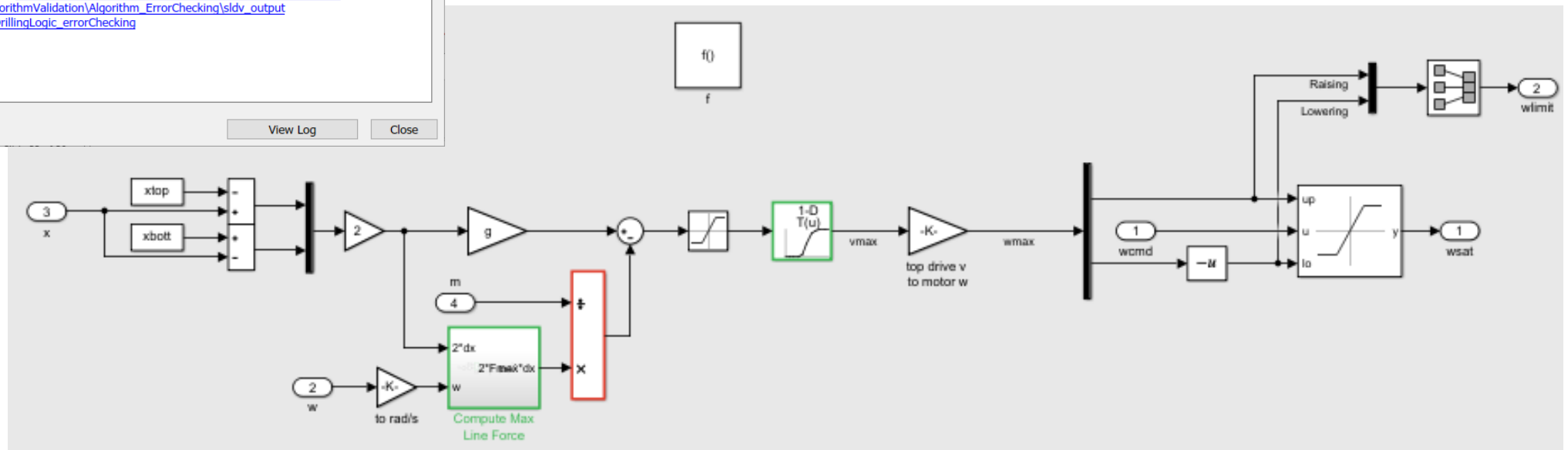
- Introduction to Modeling with Simulink and Simscape
- Leveraging Models
  - Calibrating Digital Twin with Field Data
  - Sizing Systems using Operation Data
  - Digital Twins for Anomaly Detection
- Optimizing and Validation of Controls
  - Optimizing Drilling Operations
  - Importing Existing PLC Ladders and C Software
  - Controller Error Checking and Test Vectors
- Hardware-in-the-loop Testing of PLCs

Validating New (and Existing) Logic before Deployment

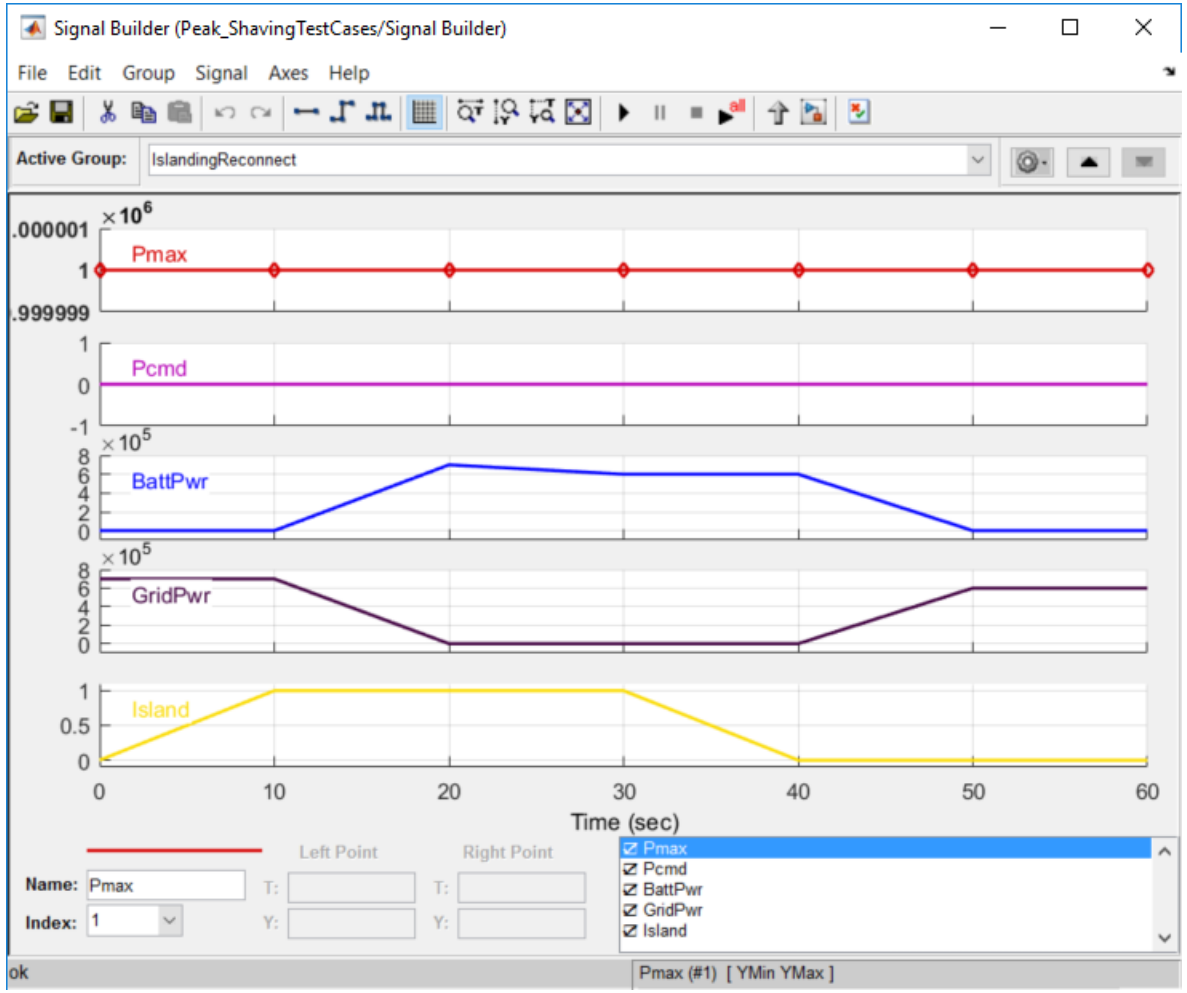


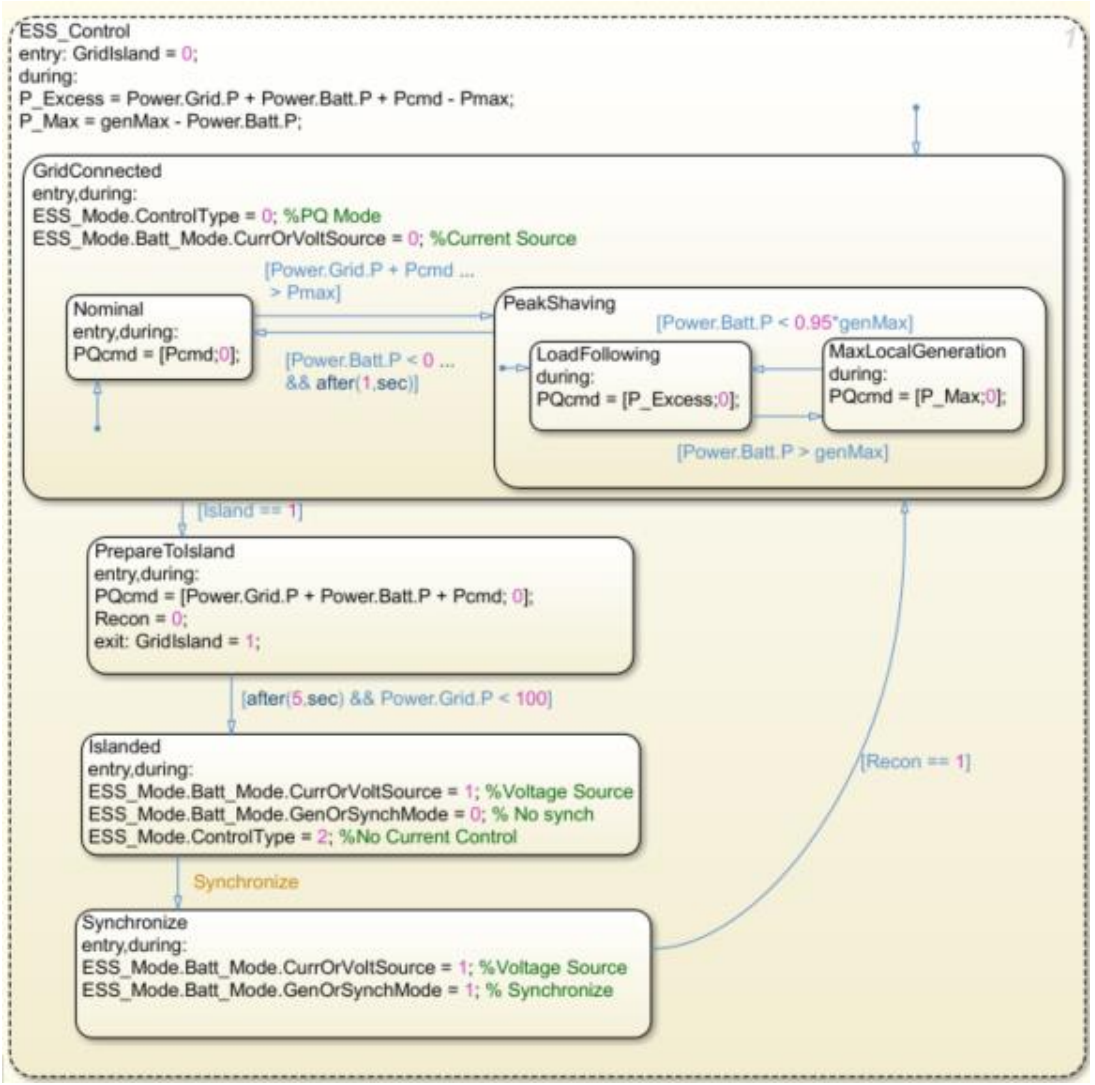
Detect hard-to-find design errors before simulation:

- Dead logic
- Division by zero
- Range violation
- Integer overflow
- Assertion violation
- Out of bound array access



Rigorous Testing of Controls before Deployment



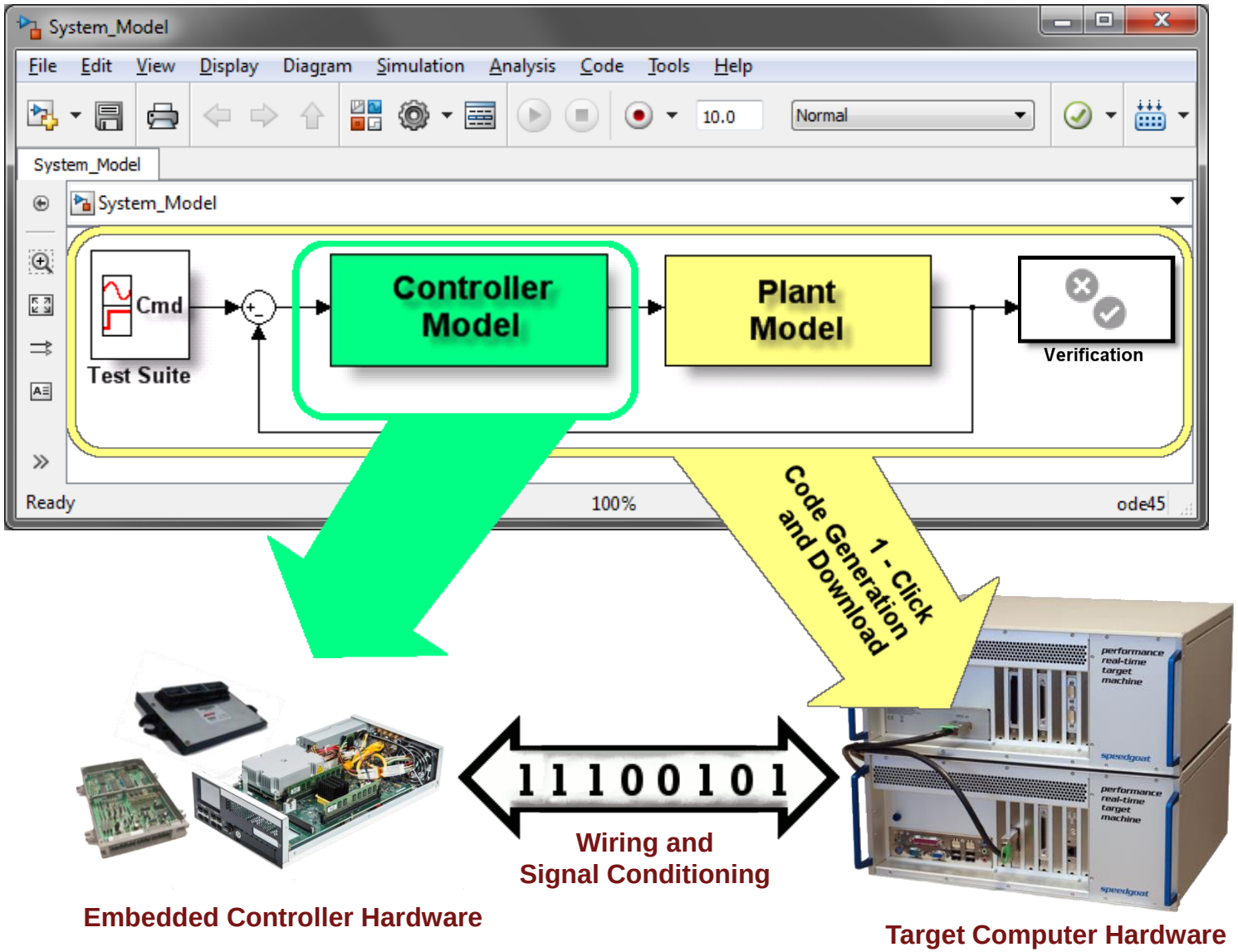


Agenda

- Introduction to Modeling with Simulink and Simscape
- Leveraging Models
  - Calibrating Digital Twin with Field Data
  - Sizing Systems using Operation Data
  - Digital Twins for Anomaly Detection
- Optimizing and Validation of Controls
  - Optimizing Drilling Operations
  - Importing Existing PLC Ladders and C Software
  - Controller Error Checking and Test Vectors
- Hardware-in-the-loop Testing of PLCs

Real-Time Simulation and Testing Tasks:

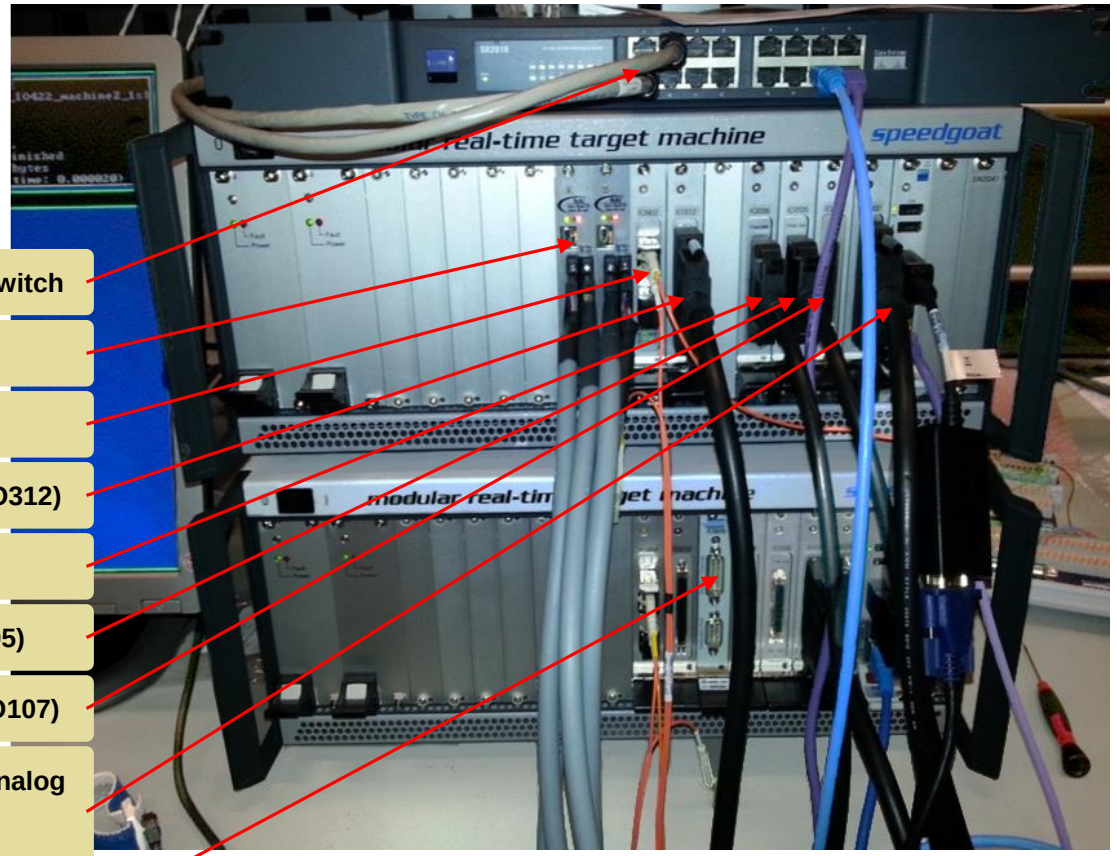
Hardware-in-the-loop (HIL) Simulation



Real-Time Simulation and Testing Example

Hardware-in-the-loop Simulation/Testing

- Hardware Under Test
  - Full authority digital engine control (FADEC)
- Simulation
  - Aircraft Engines



- Development/target computer Ethernet switch
- 6 LVDT Simulation channels (IO422)
- Shared/Reflective Memory (IO902)
- FPGA 16 Encoder Emulation channels (IO312)
- 32 24V digital input channels (IO206)
- 32 24V/0.5A digital output channels (IO205)
- 16 DIFF 16-bit analog output channels (IO107)
- 32 SE/16 DIFF 16-bit analog input, 4 SE analog output, 8 TTL digital input, 8 TTL digital output channels (IO102)
- RTD simulation (IO926)

Fixed-Function I/O Modules

Powerful “as is” functionality

IO Type	Functionality
Analog	High-resolution, high-speed, simultaneous sampling, BNC and XLR panels, …
Digital	TTL/LVCMOS, RS422/RS485/LVDS, 06-48V, low/high side, opto-coupled, …
Serial	RS232, RS422, RS485, SDLC, HDLC
Ethernet-based	EtherCAT, EtherNet/IP, Modbus TCP, POWERLINK, real-time UDP, …
Protocols	CAN, SAE J1939, LIN, Profibus, Modbus, SPI, I2C, SSI, ARINC-429, MIL-STD-1553, FlexRay, ...
Video	CameraLink, USB WebCam
Audio/Speech	Audio/Speech optimized analog IO modules
Shared Memory	Reflective Memory for high speed data transfer in multi-processor systems
Various	LVDT/RVDT, Synchro/Resolver, reed relays, programmable resistors, external signal conditioning modules (current to voltage, voltage to current, temperature, …)

- Delivery includes I/O cables, terminal boards, test models, and Simulink driver blocks
- 3 years of warranty, and long-term availability (7+ years for most I/O modules)

Real-Time Simulation and Testing Tasks:

Rapid Control Prototyping