MATLAB EXPO 2019

Introduction to Simulink and Stateflow

Tim Johns

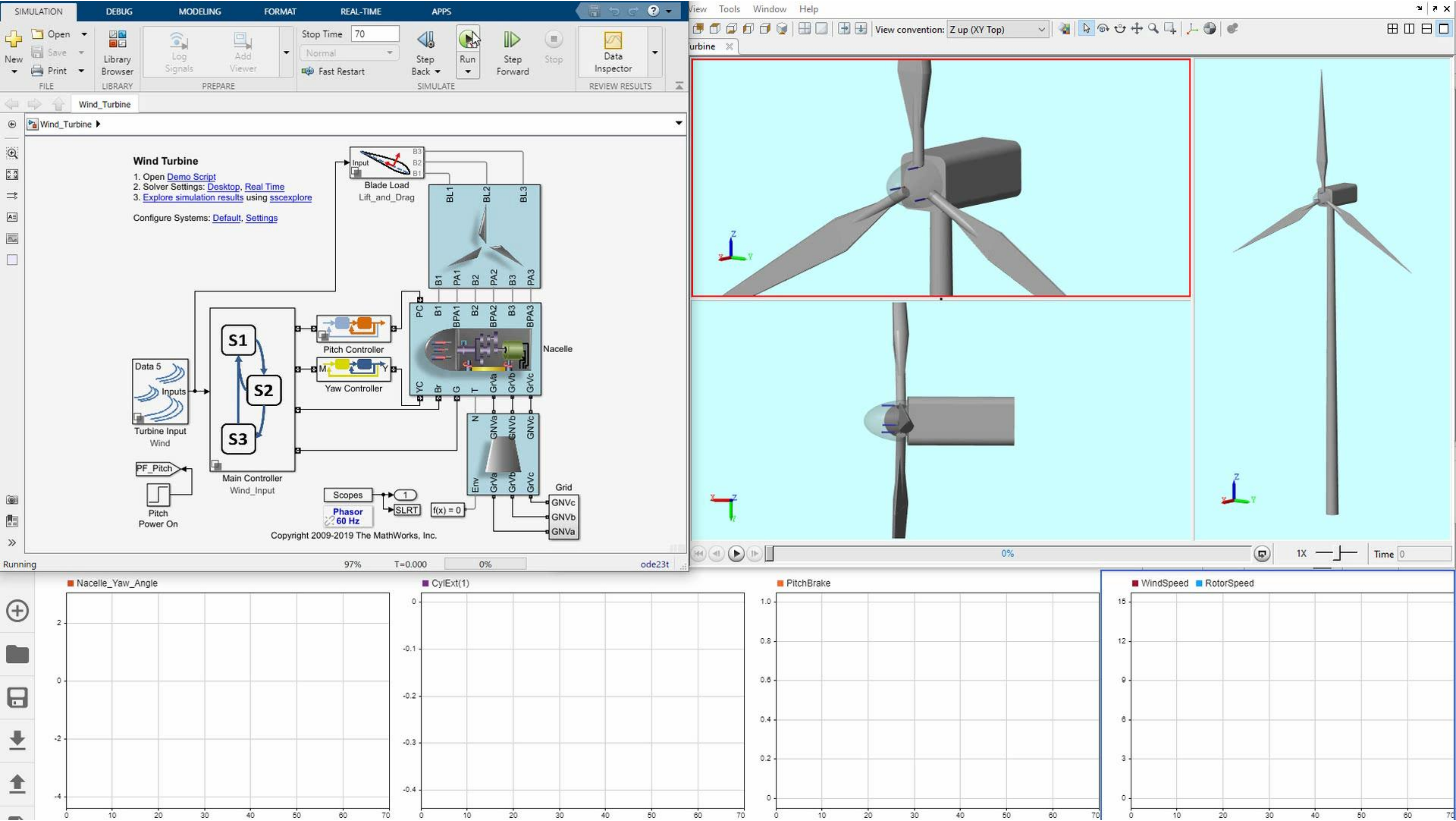
Talks earlier today

Our Journey Towards Model-Based Product Lines
Rolls-Royce

Matt Ley, Rolls-Royce

Controlling Complexity at McLaren Automotive with Model-Based Design
McLaren Automotive

Matthew Chave, McLaren Automotive Ltd.


SIMULATION
DEBUG
MODELING
FORMAT
REAL-TIME
APPS
New
Open
Save
Print
FILE
Library Browser
LIBRARY
Log Signals
Add Viewer
PREPARE
Stop Time 70
Normal
Fast Restart
Step Back
Run
Step Forward
Stop
SIMULATE
Data Inspector
REVIEW RESULTS
Wind_Turbine
Wind Turbine
1. Open Demo Script
2. Solver Settings: Desktop, Real Time
3. Explore simulation results using sscexplore
Configure Systems: Default, Settings
Blade Load
Lift_and_Drag
Pitch Controller
Yaw Controller
Nacelle
Data 5
Inputs
Turbine Input
Wind
S1
S2
S3
Main Controller
Wind_Input
PF_Pitch
Pitch
Power On
Scopes
SLRT
Phasor 60 Hz
f(x) = 0
Grid
GNVc
GNVb
GNVa
Copyright 2009-2019 The MathWorks, Inc.
Running
97%
T=0.000
0%
ode23t
View
Tools
Window
Help
View convention: Z up (XY Top)
1X
Time 0
Nacelle_Yaw_Angle
CylExt(1)
PitchBrake
WindSpeed
RotorSpeed

Topics

1. Why model a system?
2. Why use Simulink?
3. Simulink and Stateflow basics with a worked example

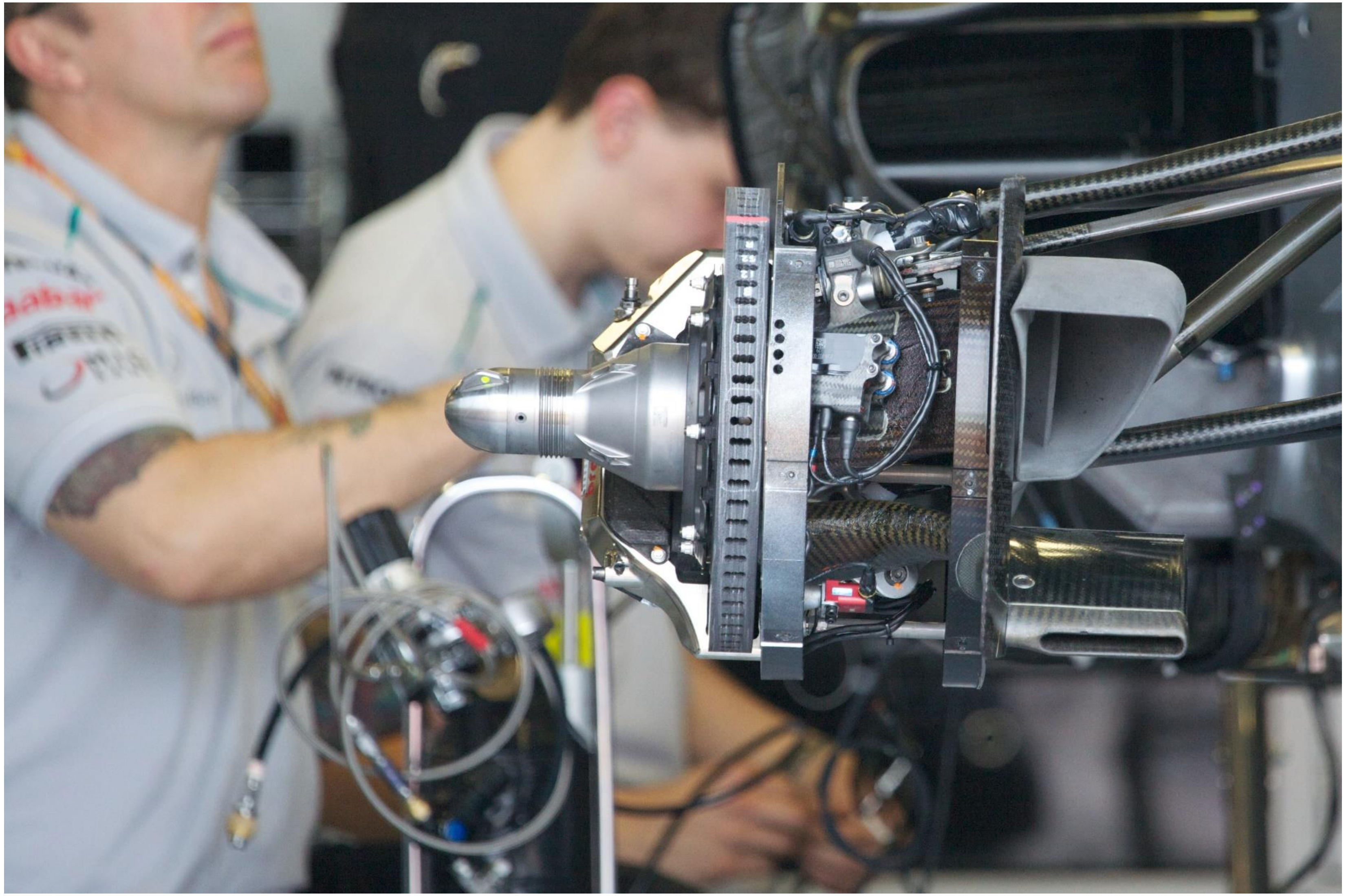
Why model a system?

Modelling & simulation gives you **insight**

Application areas

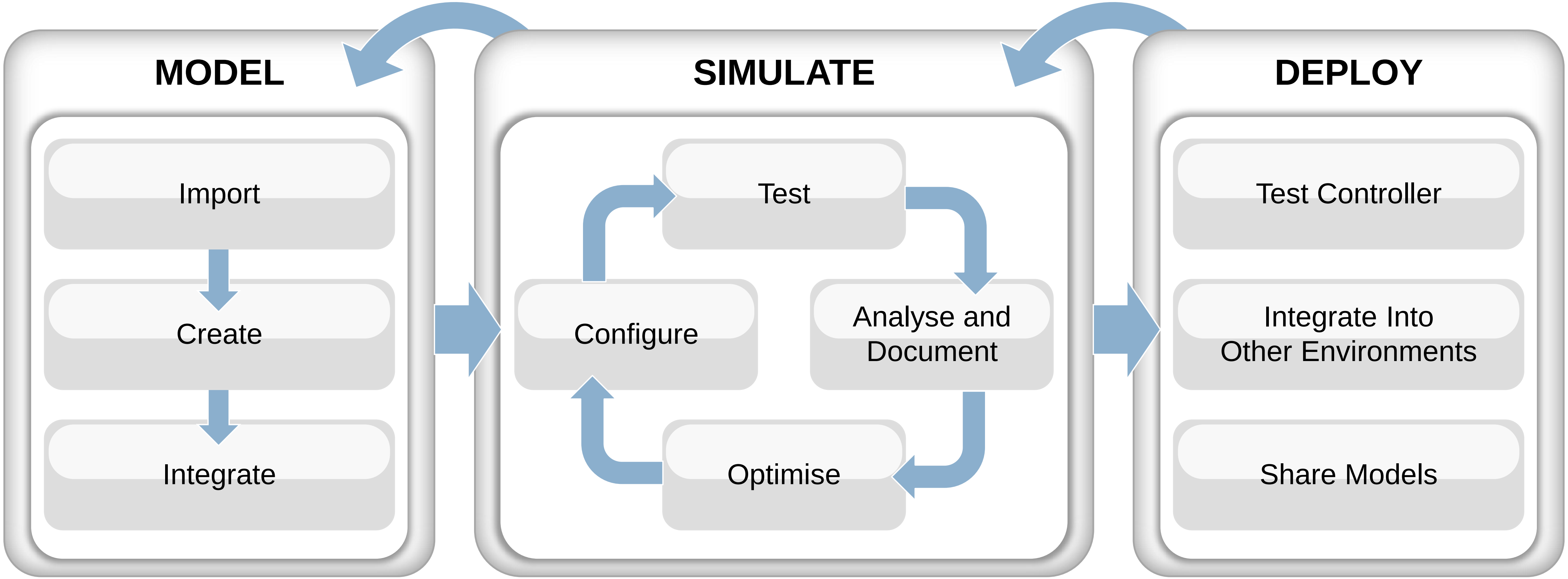
By Artes Max from Spain - Mercedes W10 / Valtteri Bottas / FIN / Mercedes-AMG Petronas Motorsport, CC BY-SA 2.0

By Krasimir Grozev - Own work, CC BY-SA 3.0

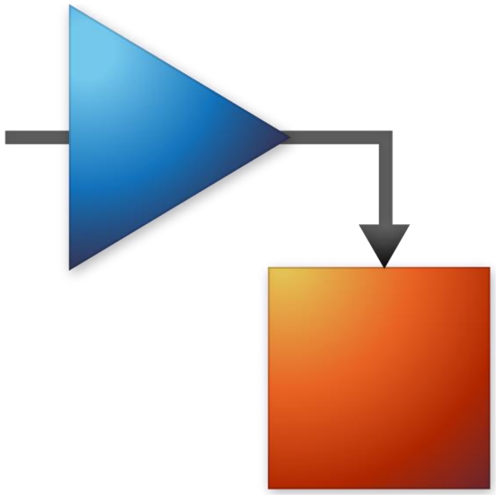


By Mark McArdle from Canada - Mercedes Brakes, CC BY-SA 2.0

- Hydraulic pressures and flows
- Control systems
- Heat dissipation
- Suspension motion

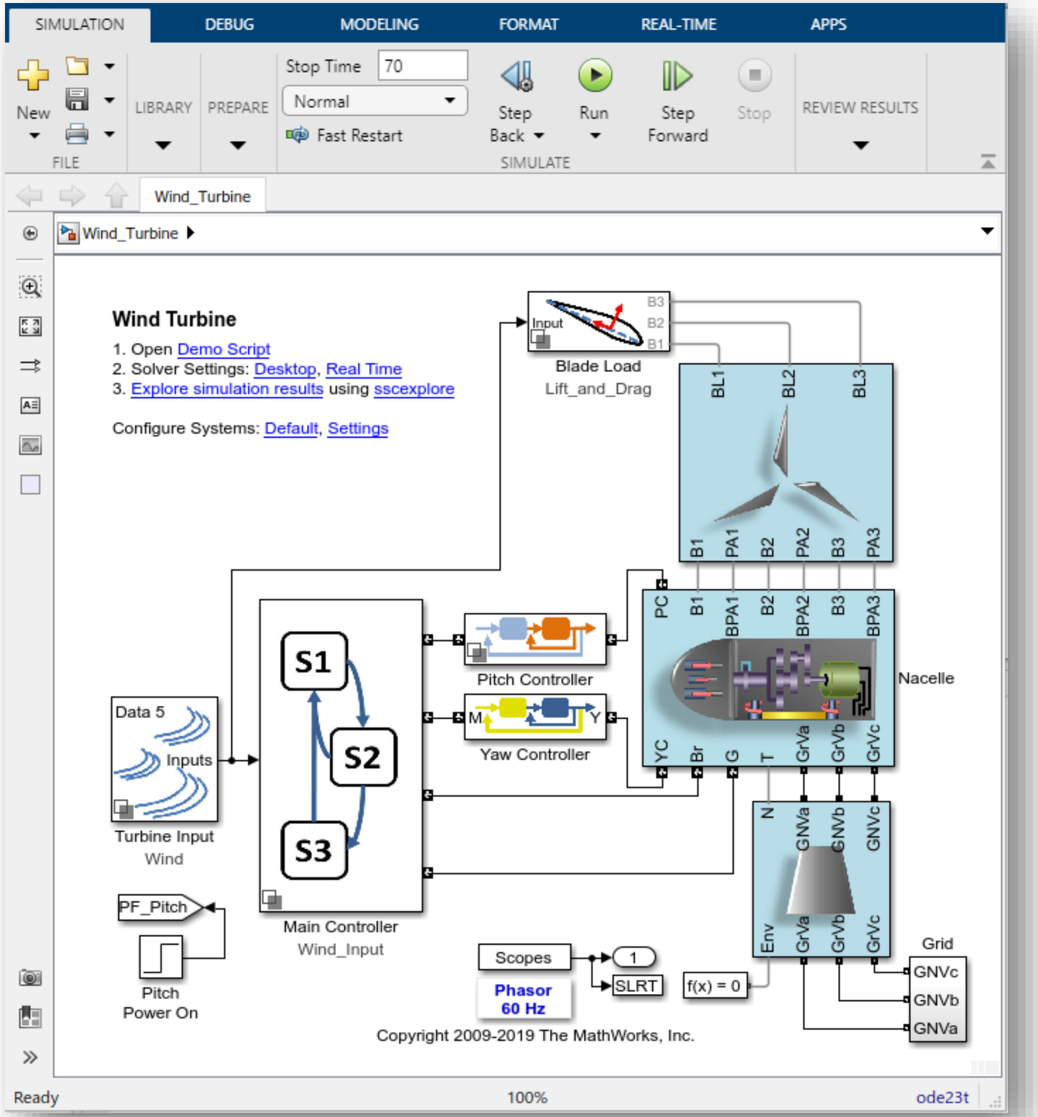

MODEL
Import
Create
Integrate
SIMULATE
Test
Configure
Analyse and Document
Optimise
DEPLOY
Test Controller
Integrate Into Other Environments
Share Models

What is **Simulink?**



What is Simulink?

1. Graphical programming language
2. Solve time-domain problems



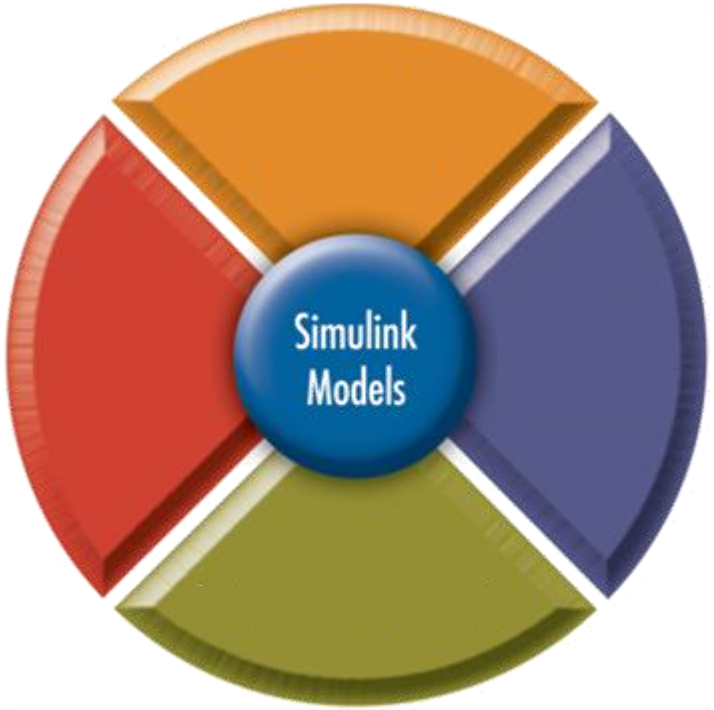
Why use **Simulink?**

Why use Simulink?

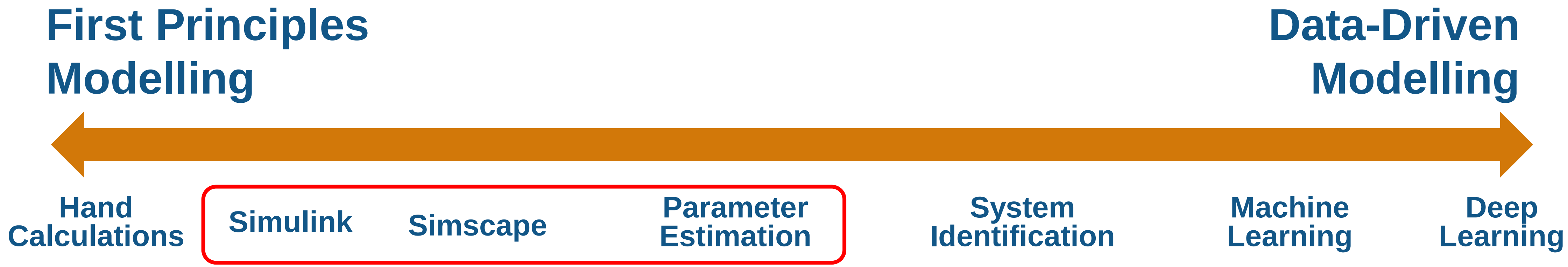
1. Integrated environment
2. Requirements to C code
3. Automate & analyse with MATLAB

Model Based Design with Simulink

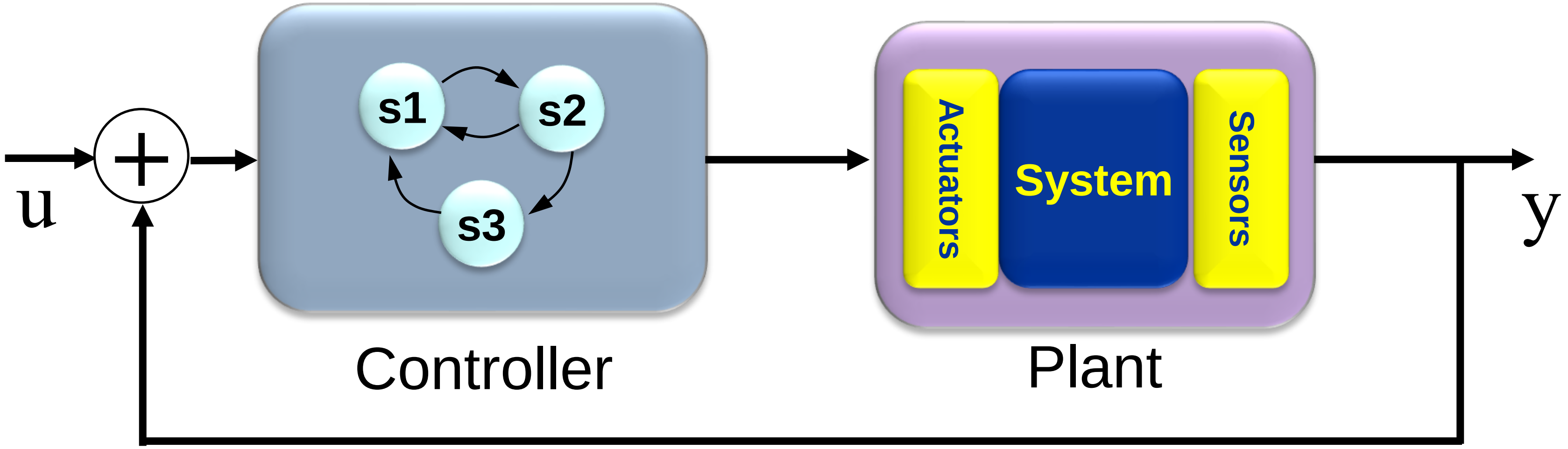
- Modelling and simulation
  - Multi-domain dynamic systems
  - Nonlinear systems
  - Continuous-time, discrete-time, multi-rate systems
- Plant and Controller Design
  - Select/optimise control architecture and parameters
  - Rapidly model “what-if” scenarios
  - Communicate design ideas
  - Embody performance specifications
- Implementation
  - Automatic code generation
    - Embedded systems, FPGAs, GPUs
  - Rapid prototyping for SIL, PIL, HIL
  - Verification and validation



Modelling systems

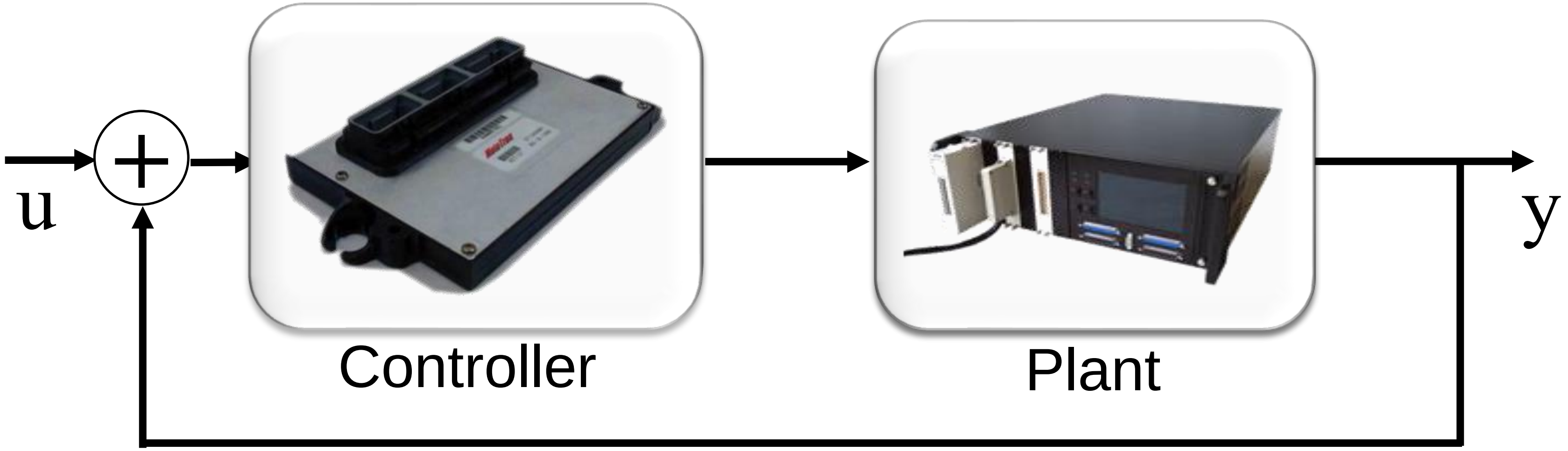


Optimise System-Level Performance



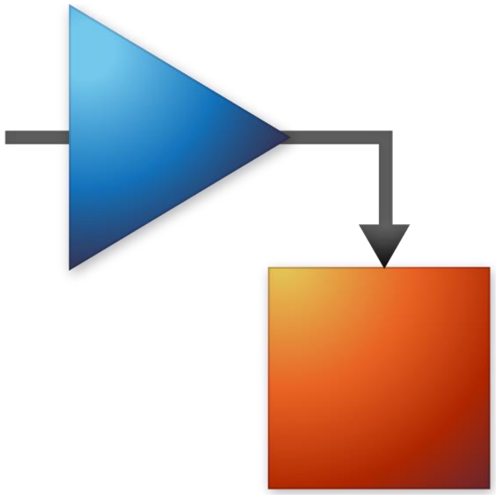
- Simulating plant and controller **in one environment** allows you to
  - Study closed-loop behaviour
  - Optimise system-level performance.
  - Automate tuning process using optimisation algorithms
  - Accelerate process using parallel computing

Detect Integration Issues Earlier



- Controls engineers and domain specialists can work together to **detect integration issues in simulation**
  - Convert plant models to C code for hardware-in-the-loop tests
  - Share models with other internal users
  - Share models with external users while protecting IP

Using **Simulink**



From: *Bill*
To: *Tim*
Subject: *Motor model and control*

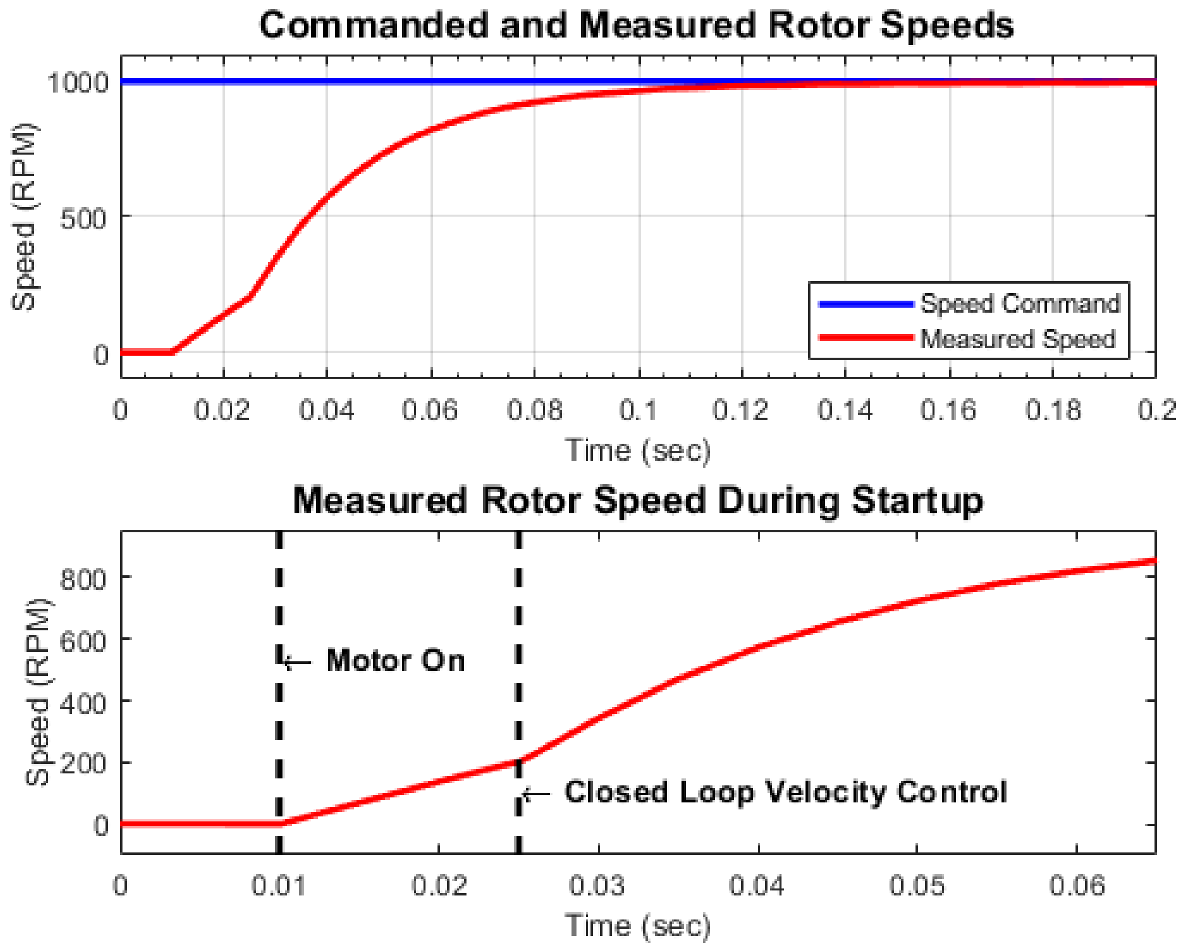
Hi Tim

We're adding an electric camera gimbal to our quadcopter. We need to understand how the motors will behave. We also need to develop a control system. Can you help?

Thanks

Bill

Closed-loop motor control



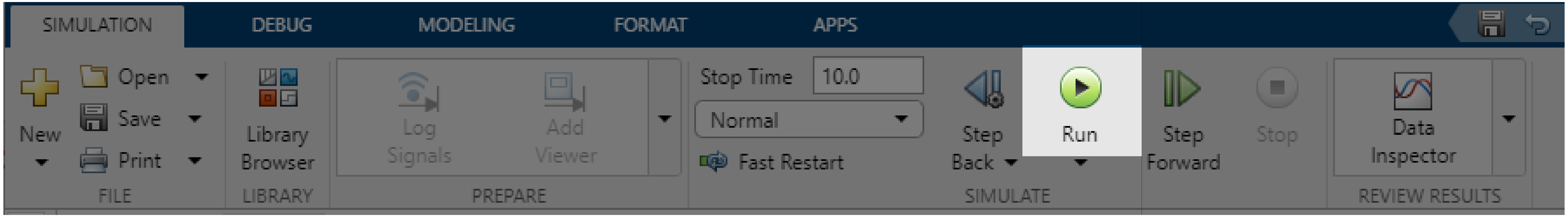
What questions might we want to answer?

1. Can I get the closed-loop response I need?
2. What current will my motor draw during operation?
3. Does my system still work if component values change?
4. What if…?

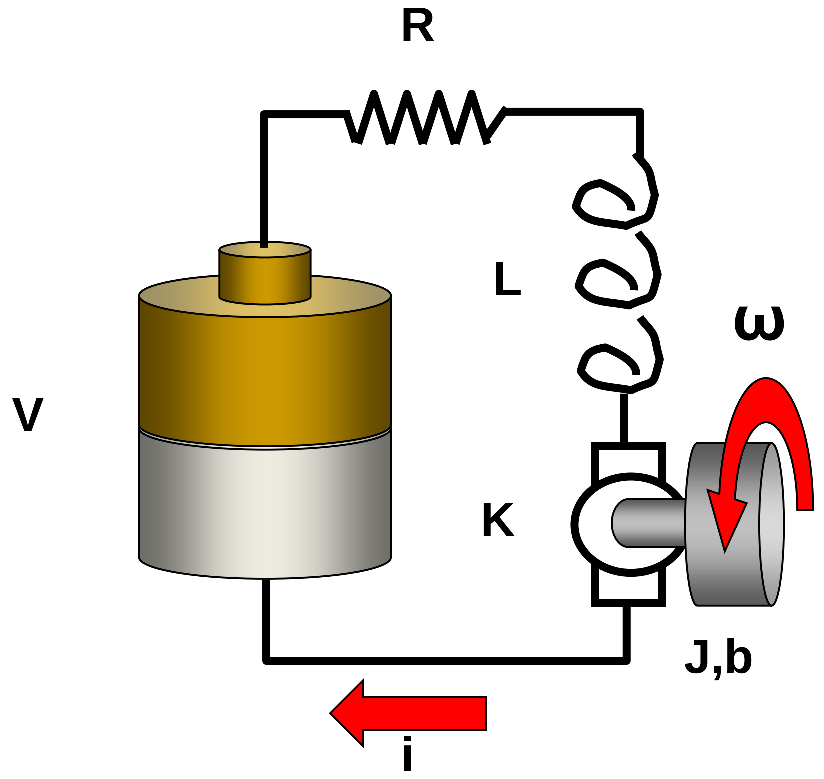
Steps in the process

1. Model the motor & speed controller
2. Refine the motor model using measured data
3. Model the supervisory logic
4. Deploy the control model to hardware

At each stage **simulate the model**



PID Control of a DC Motor

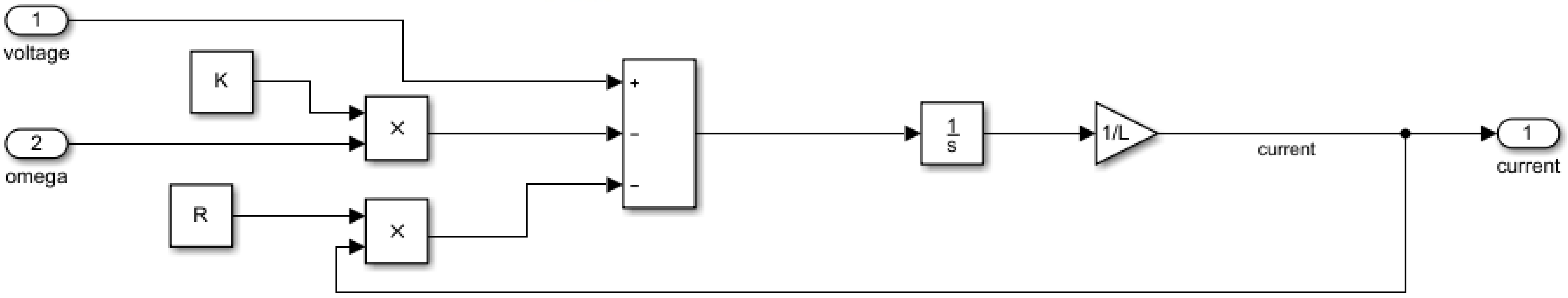


$$V = K \cdot \omega + i \cdot R + L \frac{di}{dt} \Rightarrow i = \frac{1}{L} \int (V - K \cdot \omega - i \cdot R) dt$$

$$-T = K \cdot i - b \cdot \omega - J \frac{d\omega}{dt} \Rightarrow \omega = \frac{1}{J} \int (T + K \cdot i - b \cdot \omega) dt$$

Modelling the electrical system

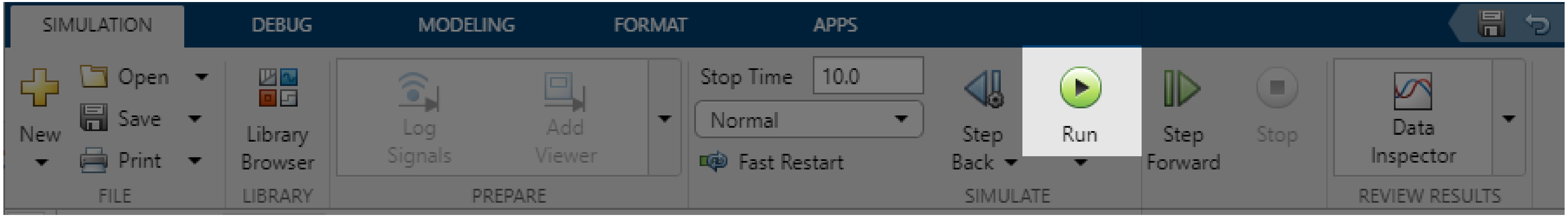
$$i = \frac{1}{L}\int (V - K\cdot\omega - i\cdot R)dt$$



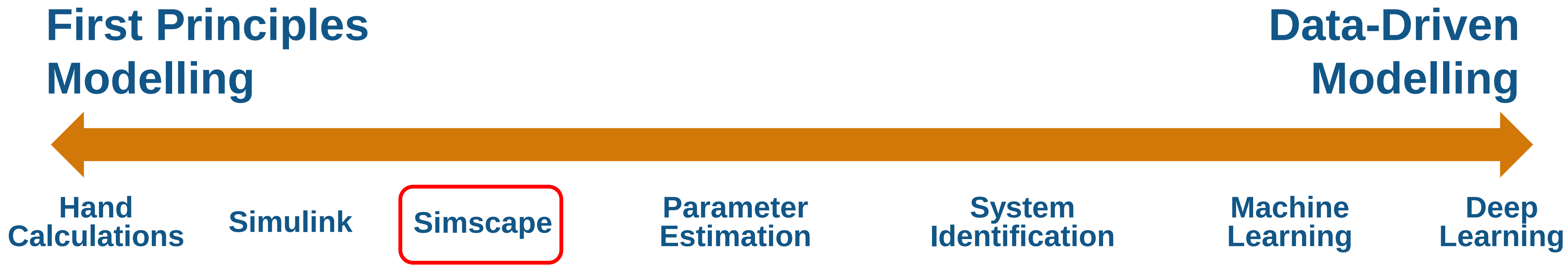
Steps in the process

✓ Model the motor & speed controller
2. Refine the motor model using measured data
3. Model the supervisory logic
4. Deploy the control model to hardware

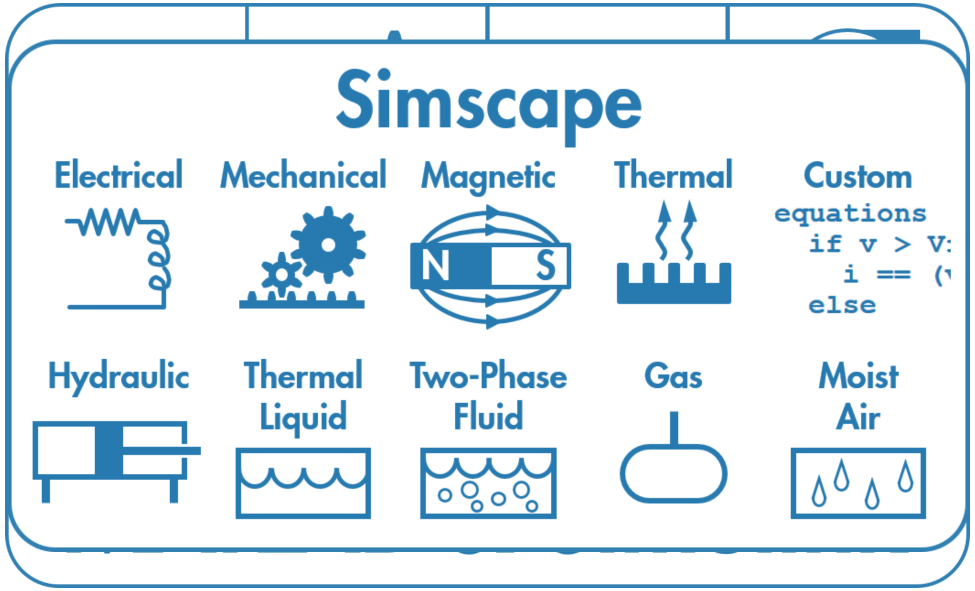
At each stage **simulate the model**



Modelling systems

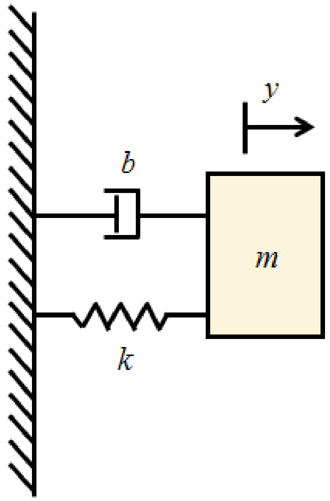
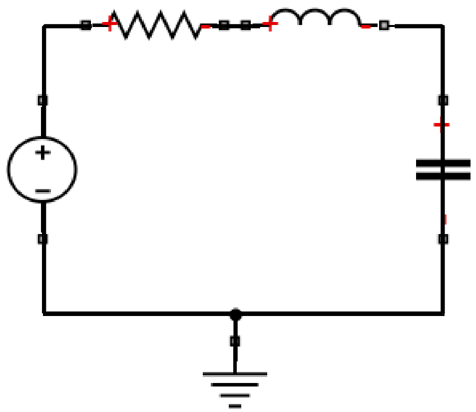


What is Simscape?



- Multi-domain physical modeling (acausal)
- Work with components, not equations
- Models reflect structure of physical system

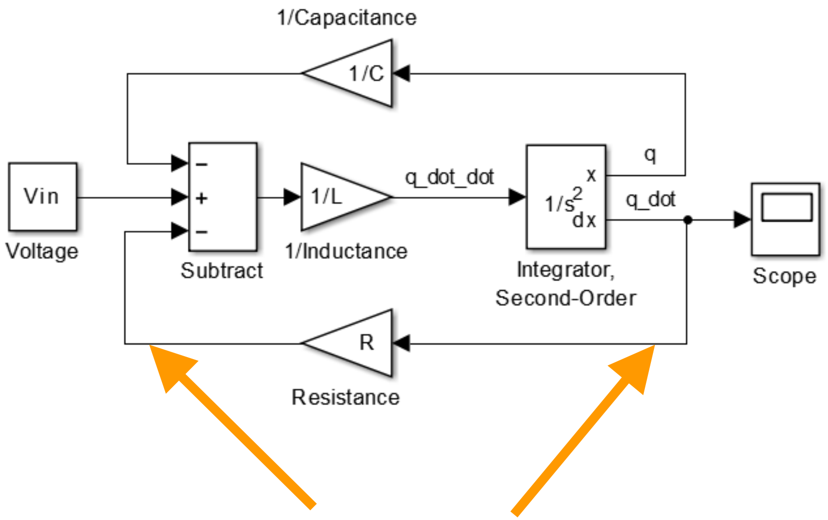
Simulink vs Simscape

	Diagram	Equation
Mass-spring-damper		$m\ddot{y} + b\dot{y} + ky = 0$
RLC circuits		$L\ddot{q} + R\dot{q} + \frac{1}{C}q = V_{in}$

Simulink

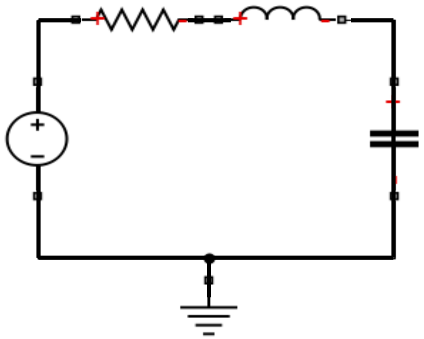
$$L\ddot{q} + R\dot{q} + \frac{1}{C}q = V_{in}$$

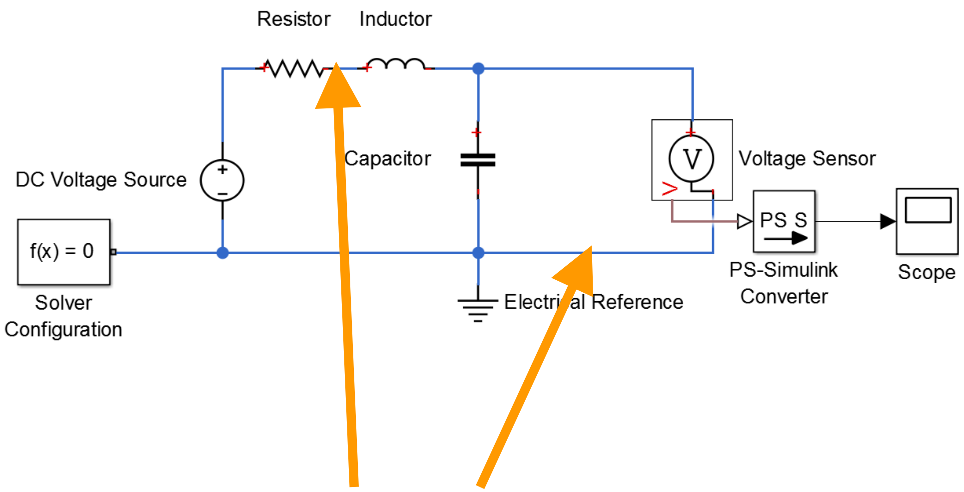
$$\ddot{q} = \frac{1}{L}\left(V_{in} - R\dot{q} - \frac{q}{C}\right)$$



Connections represent signal transmissions

Simscape





Connections represent physical connections

Simscape Products

Electrical

- Actuators
- Semiconductors
- Motors & generators
- Transformers

Driveline

- Gears, belts
- Tyres, engines
- Clutches
- Transmissions

Multibody

- 3D Dynamics
- Kinematics
- Force Analysis
- CAD Import

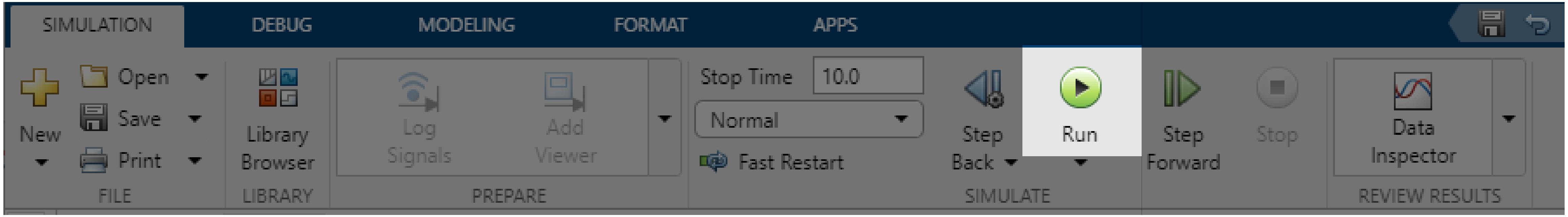
Fluids

- Hydraulic actuation
- Heating and cooling
- Fluid transportation
- Fluid power

Steps in the process

✓ Model the motor & speed controller
2. Refine the motor model using measured data
3. Model the supervisory logic
4. Deploy the control model to hardware

At each stage **simulate the model**



From: *Bill*
To: *Tim*
Subject: *Calibrating motor model*

Hi Tim

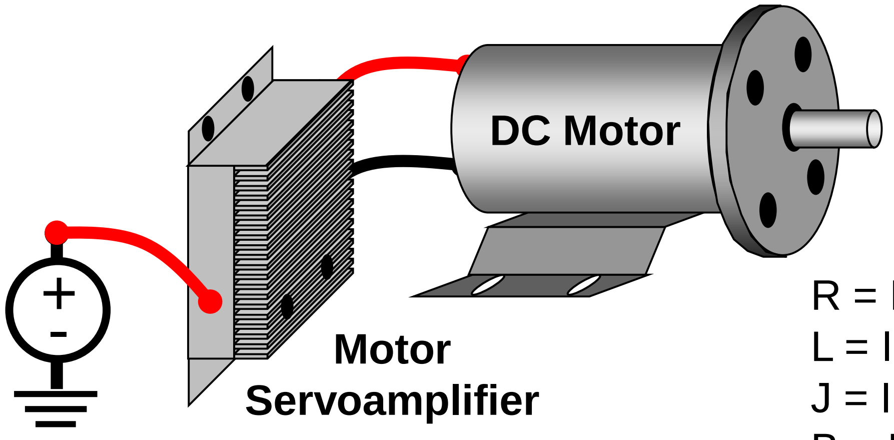
I tried to run the motor model you sent, but the outputs didn’t match up at all with the motor in our lab. Please can you send a model that’s more useful!

Thanks

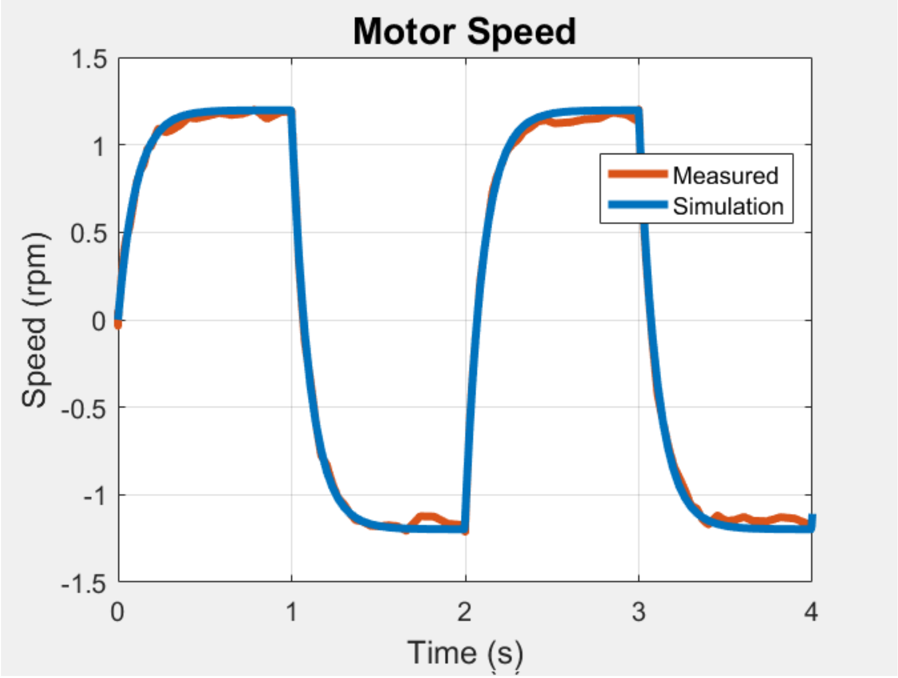
Bill

Estimating parameters using measured data

Model:



R = Resistance
L = Inductance
J = Inertia
B = Friction
K = Back EMF Constant



Problem: Simulation results do not match measured data

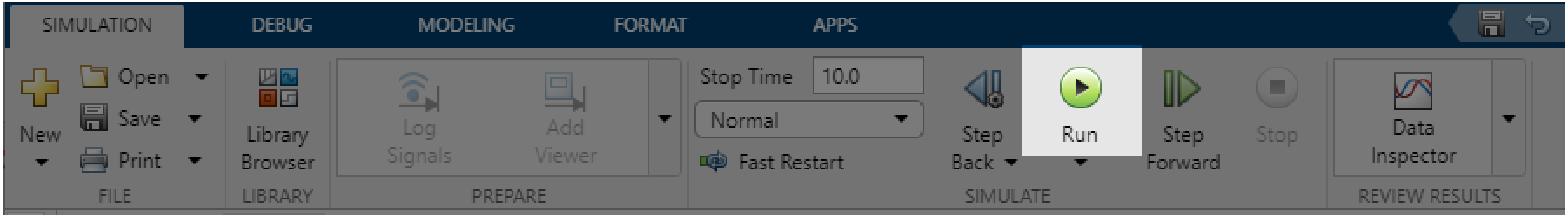
Solution: Use Simulink Design Optimization to automatically tune model parameters

Steps in the process

- ✓ Model the motor & speed controller
- ✓ Refine the motor model using measured data

3. Model the supervisory logic
4. Deploy the control model to hardware

At each stage **simulate the model**



From: *Bill*
To: *Tim*
Subject: *Motor controller calibration*

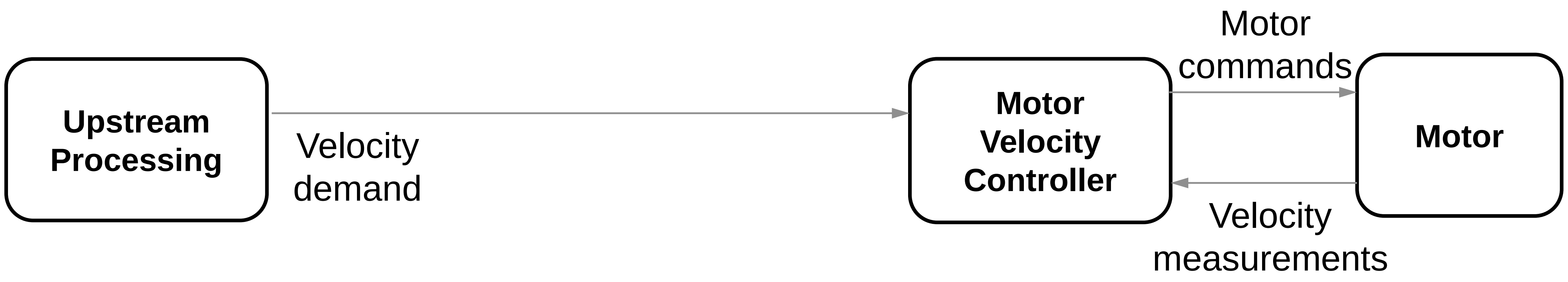
Hi Tim

When the system is turned on, the motor could be in any position. Could you add a calibration step so that it always starts in the same place?

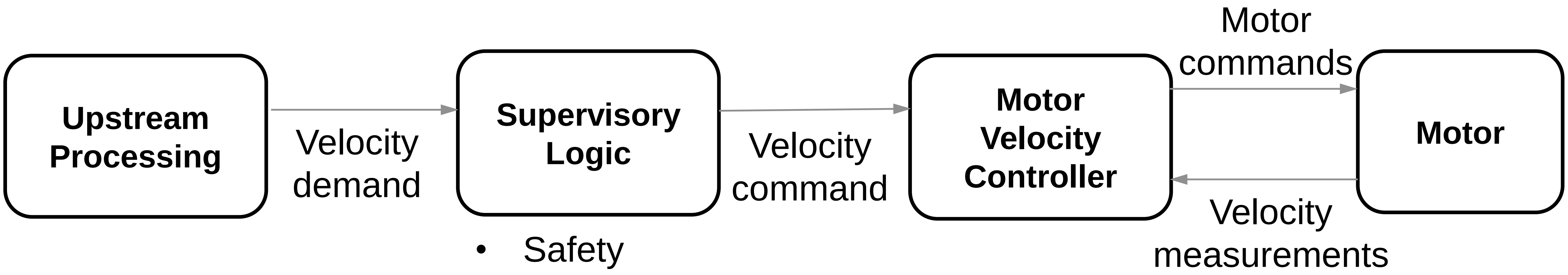
Thanks,

Bill

System Overview



System Overview



- Safety
- Modes
- Smoothing

Supervisory logic

- Receive input:
  - velocityDemand: demanded motor speed

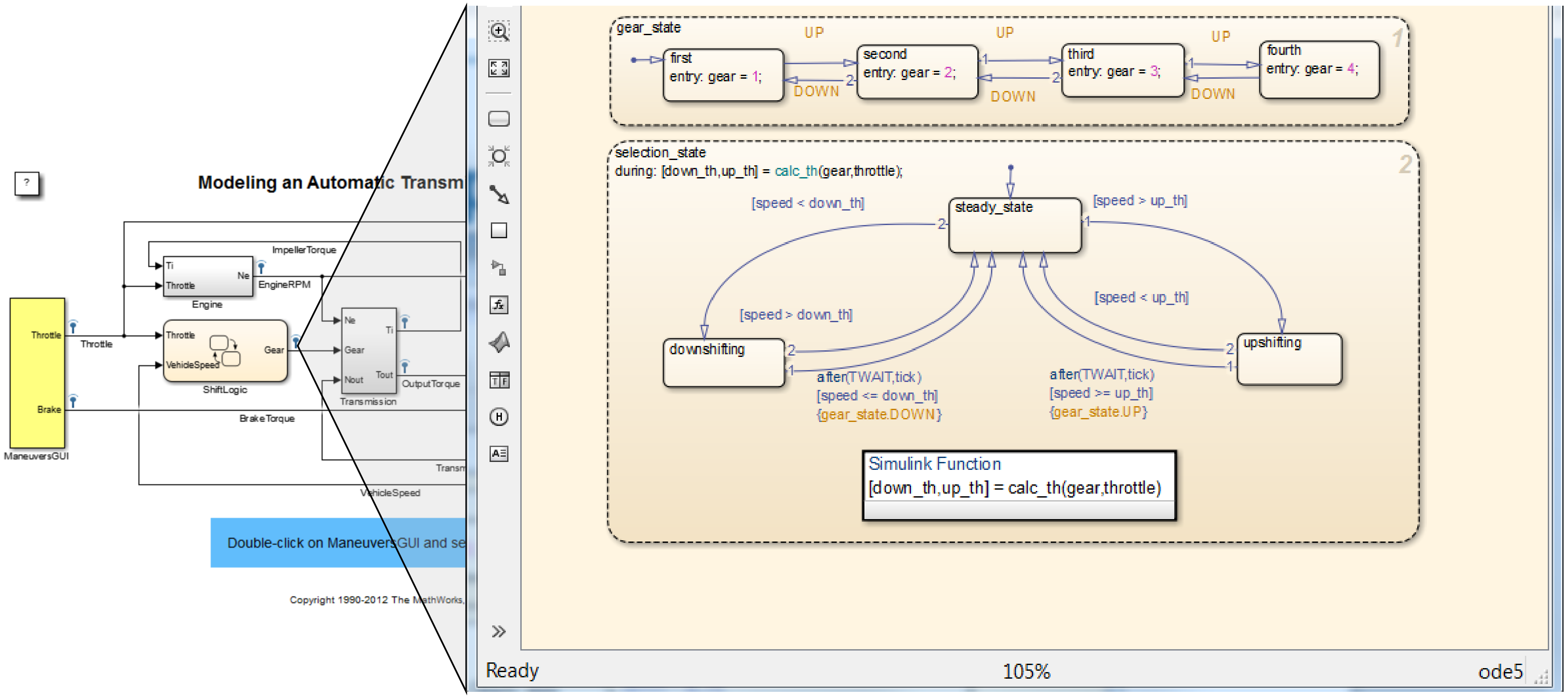
- Send three outputs:
  - motorOn: whether motor should be enabled or not
  - commandType: current operating mode
  - velocityCommand: demanded motor speed

- When motor is enabled, run required motor calibration steps:
  - Set mode to calibration
  - Run at 25 rad/s for 1 second
  - Stop motor for 1 seconds
  - Set mode to velocity control

How can we implement this?

Stateflow

- State machines and flow charts
- Combine control, supervisory, and mode logic



From: *Bill*
To: *Tim*
Subject: *URGENT: Unexpected motor controller behaviour*

Hi Tim

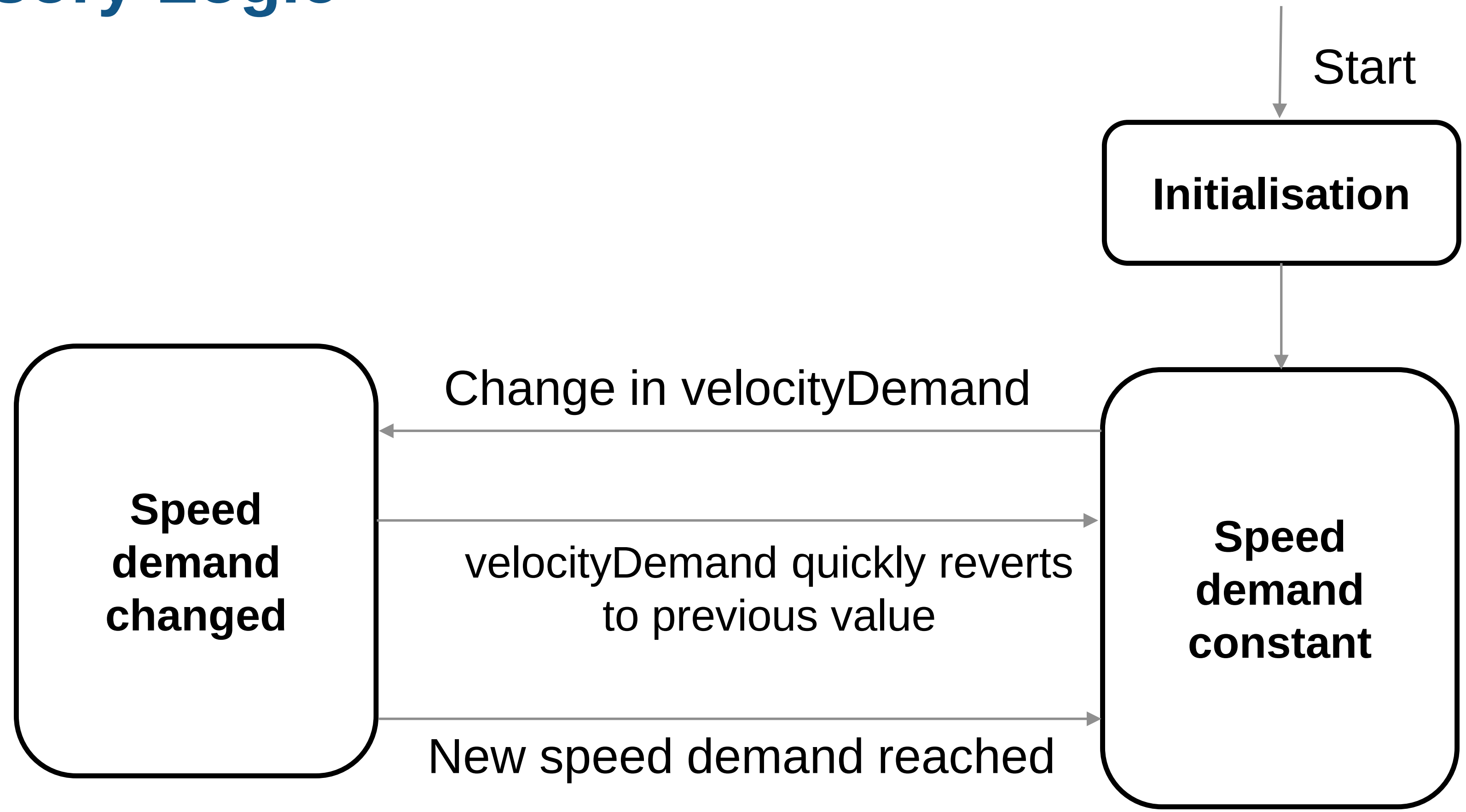
While testing out the motor control algorithms today, the motor suddenly slowed down before spinning back to its demanded speed. Please can you look into it asap and send us an updated algorithm.

Thanks
Bill

Supervisory Logic

- Receive input
- Send outputs
- Run required motor calibration steps
- Sanitise the input demand to ensure smooth operation of motor

Supervisory Logic



How can we implement this?

From: *Bill*
To: *Tim*
Subject: *URGENT: New motor speed requirement*

Hi Tim

We’ve just heard from the motor supplier that there’s a potential design issue if the motor speed changes too quickly. Please can you change the supervisory logic to avoid us running into this problem.

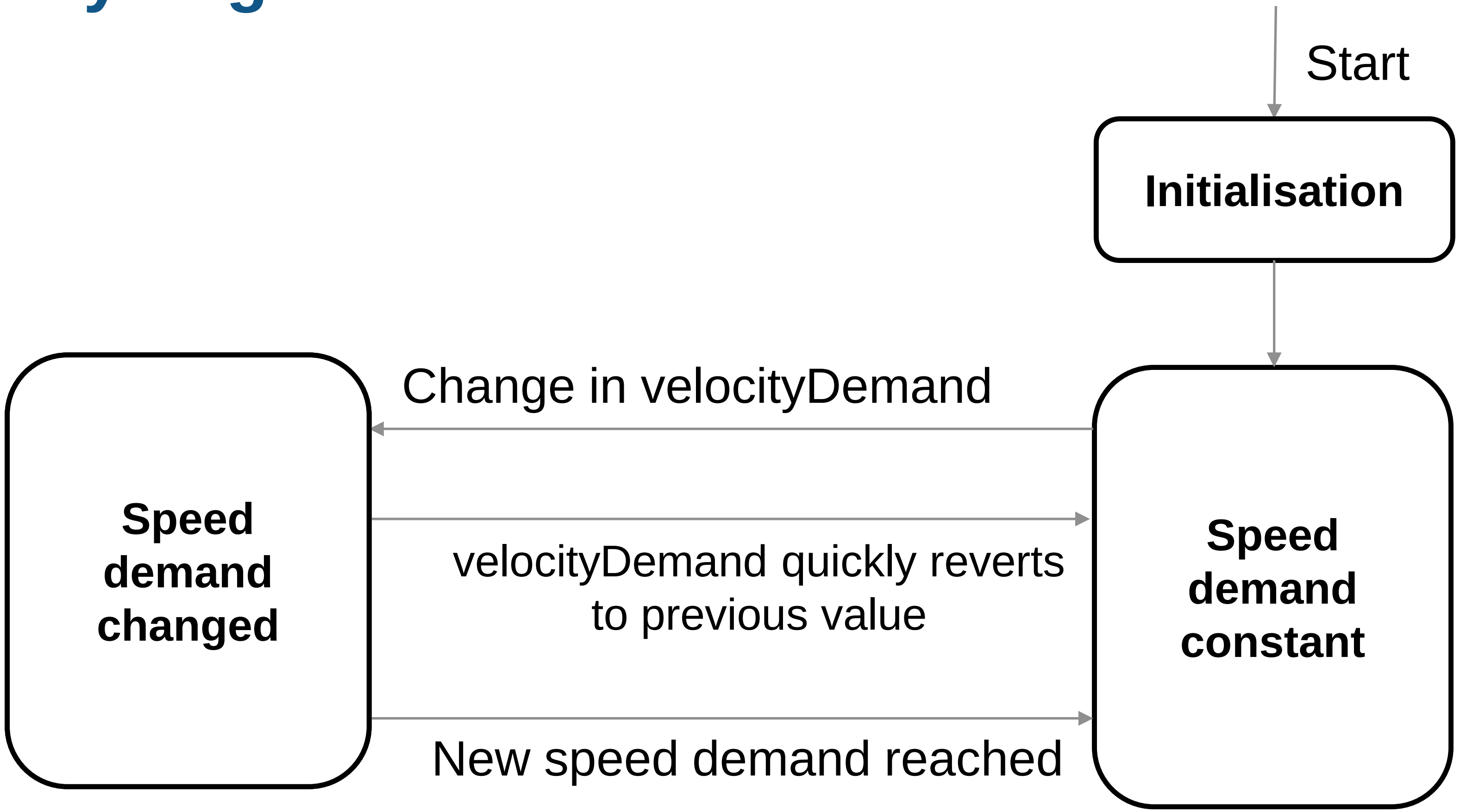
Thanks,

Bill

Supervisory Logic

- Receive input
- Send outputs
- Run required motor calibration steps
- Sanitise the input demands to ensure smooth operation of motor
- Safely transition between large changes in input demands

Supervisory Logic



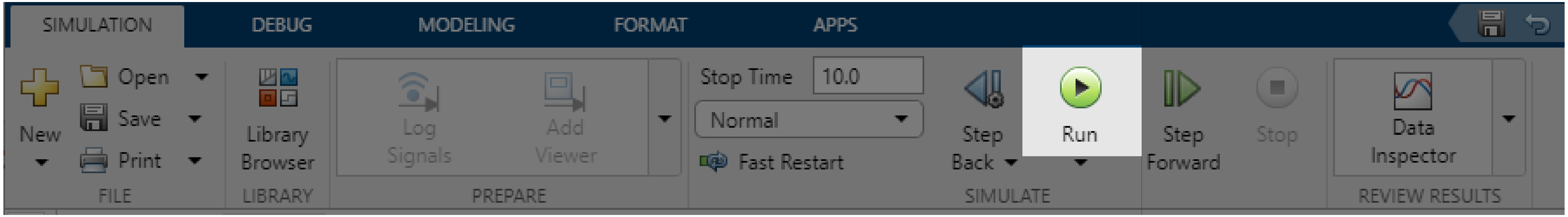
How can we implement this?

Steps in the process

- ✓ Model the motor & speed controller
- ✓ Refine the motor model using measured data
- ✓ Model the supervisory logic

4. Deploy the control model to hardware

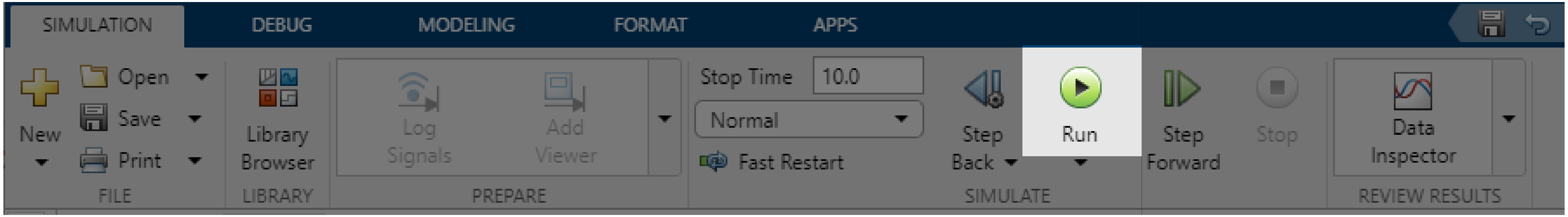
At each stage **simulate the model**



Steps in the process

- ✓ Model the motor & speed controller
- ✓ Refine the motor model using measured data
- ✓ Model the supervisory logic
- ✓ Deploy the control model to hardware

At each stage **simulate the model**



From: *Bill*
To: *Tim*
Subject: *Project signed off*

Hi Tim,

Thanks for all your hard work on this project, the customer has just signed it off! We’ve made great progress and I look forward to working with you again.

Bill

From: *Tim*
To: *Bill*
Subject: *Re: Project signed off*

Thanks Bill. Why don’t you try the free Simulink Onramp training so that you can do it yourself next time? Or there’s a 2 day training course in Cambridge in December!

Tim

Conclusions

1. Modelling and simulation give you **insight** to:
   - Make **smarter decisions**
   - Make them **earlier**

2. **Simulink** allows you to model the **complete system** in a **single environment**

Embraer Speeds Requirements Engineering and Prototyping of Legacy 500 Flight Control System

Challenge

Accelerate development of the **flight control system** software for the Legacy 500 midsized business jet

Solution

Use **Simulink** to **model the system** and the **aircraft dynamics**, run requirements-based test simulations, and speed the delivery of mature, internally validated software requirements

Results

- **Development time reduced by at least 6 months**
- Delays due to requirements issues minimised
- Models reused for real-time testing

The Embraer Legacy 500.

"Modeling with Simulink is instrumental to our team's ARP 4754 work, specifically validating system-level requirements, developing requirements-based tests, and defining low-level software requirements that our supplier uses to produce DO-178 Level A flight code using Simulink and Embedded Coder."

- Rodrigo Fontes Souto, Embraer

Link to user story

Virgin Orbit Simulates LauncherOne Stage Separation Events

Challenge

Simulate separation events for LauncherOne spacecraft to ensure sufficient clearance between separating structures

Solution

Use MATLAB, **Simulink**, and **Simscape** Multibody to **model structural components**, automate Monte Carlo simulations, and analyse and visualize results

Results

- **Simulations completed 10 times faster**
- Simulation set up times cut by up to 90%
- Hardware designs informed by simulation results

Virgin Orbit's LauncherOne vehicle assembled (top), with exploded view showing the fairing, payload, and first and second stages (bottom).

> *"MATLAB and Simulink saved us about 90% on costs compared with the alternative we considered while giving us the coding flexibility to develop our own modules and fully understand the assumptions being made, which is essential when reporting results to other teams."*
>
> *- Patrick Harvey, Virgin Orbit*

Link to user story

Bombardier Transportation Implements Model-Based Design to Accelerate Rail Propulsion System Development

Challenge

Reduce embedded control software lead times for rail propulsion systems

Solution

Use **Model-Based Design** to validate requirements, verify designs, and generate production code for embedded targets

Results

- **Costs reduced by 45%, lead time by 35%**
- Adoption of new workflow streamlined
- Code implementation delays minimised

Bombardier train used in Germany.

"Our Simulink models serve as a single, cohesive source of our design—including documentation and implementation—which lowers costs and makes changes easier to implement. As we add more automated tests, we reduce certification costs, as well."

- Claes Lindskog, Bombardier Transportation

Link to user story

What next?

Simulink Onramp

- Free!
- 2.5 hours
- Self-paced online

Simulink for System and Algorithm Modelling

- 2 days
- Instructor-led
- Next runs in Cambridge in December