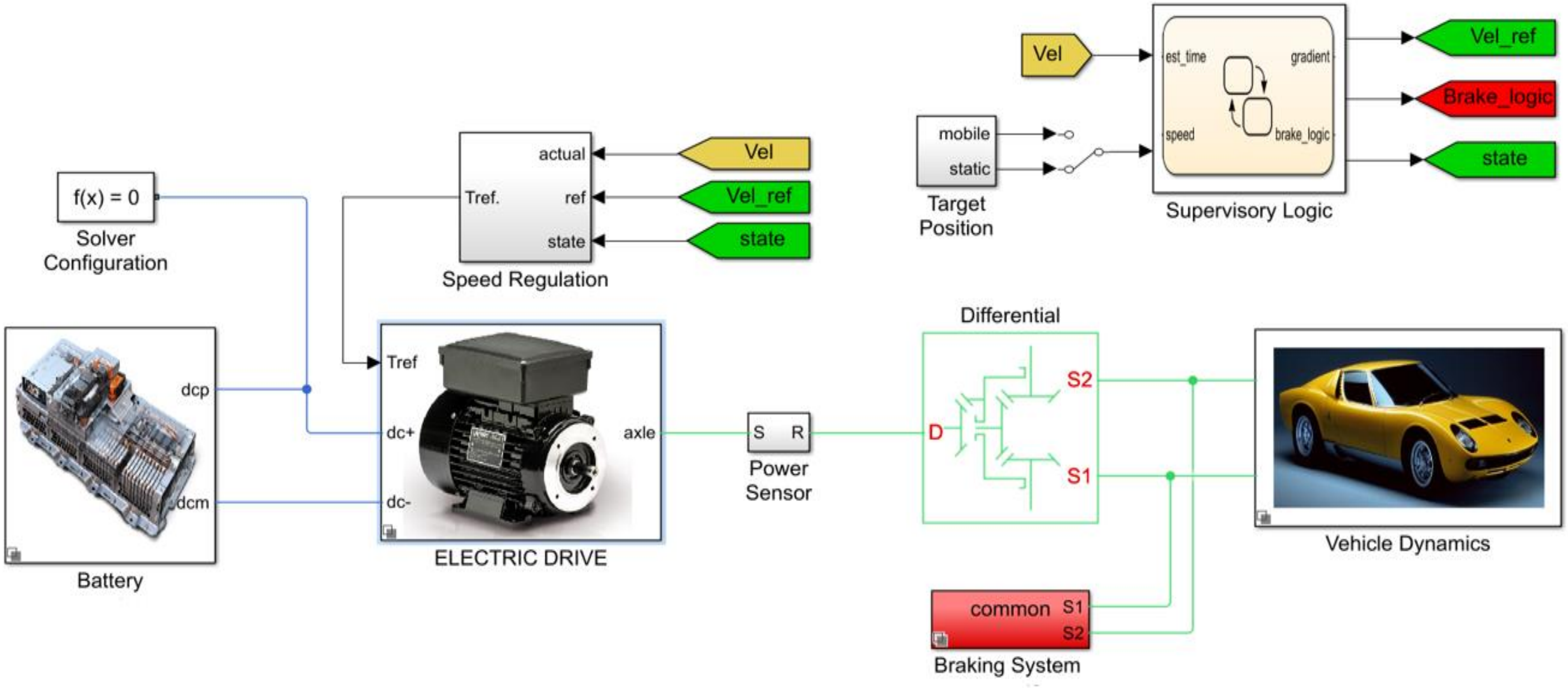
MATLAB EXPO 2018

Managing Performance and Safety in Multi-Domain Complex Systems

Juan Sagarduy

Managing Performance & Safety in Multi-Domain Systems



Performance
size-weight
autonomy

Electric Vehicle Active Safety

Safety
emergency response
fault management

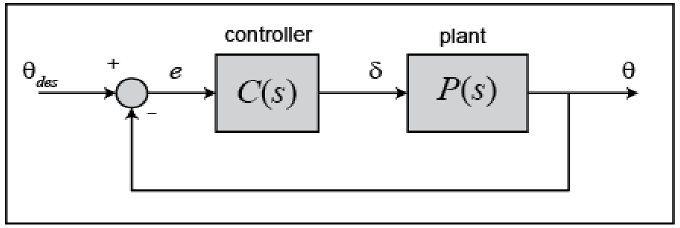
Managing Performance & Safety in Multi-Domain Systems

1. System design
(integration, optimization)

Managing Performance & Safety in Multi-Domain Systems

1. System design (integration, optimization)

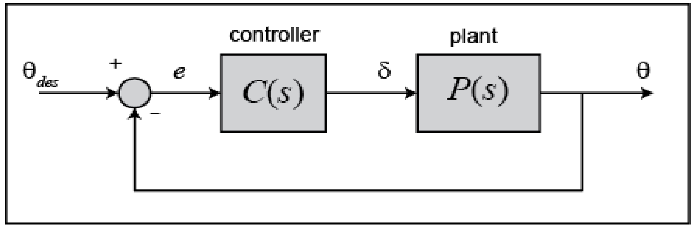
2. Development of control algorithms



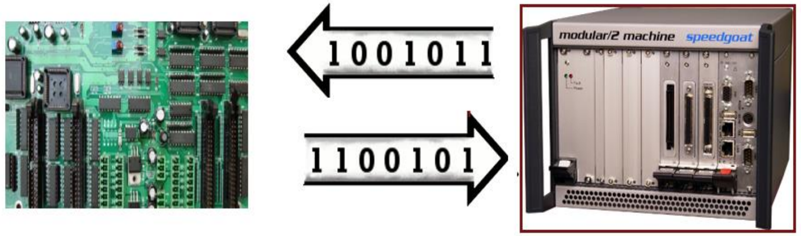
Managing Performance & Safety in Multi-Domain Systems

1. System design (integration, optimization)

2. Development of control algorithms



3. Hardware-based physical emulation (real-time testing)

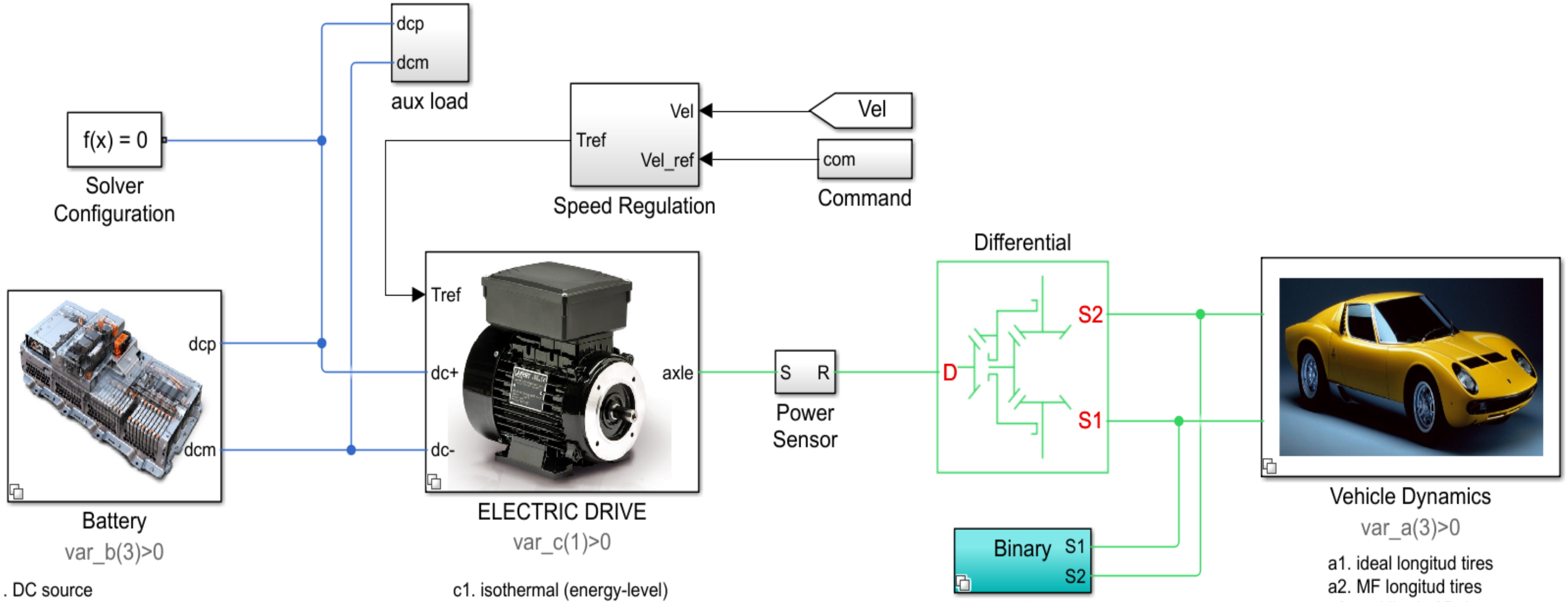


1. System Design

Explore – Integrate – Optimize

Active Safety – System Design

Step 1. Model configuration (no supervisory logic) to understand physical behaviour and add controllability



Battery
var_b(3)>0

b1. DC source
b2. thermal cooling - ideal power
b3. fluid cooling - ideal flow source
b4. fluid cooling - valve-based

ELECTRIC DRIVE
var_c(1)>0

c1. isothermal (energy-level)
c2. thermal (energy-level)
c3. advanced FEM (detailed)

Braking System
var_d(1)>0

d1. binary
d2. independent (actuator)
d3. common (actuator)

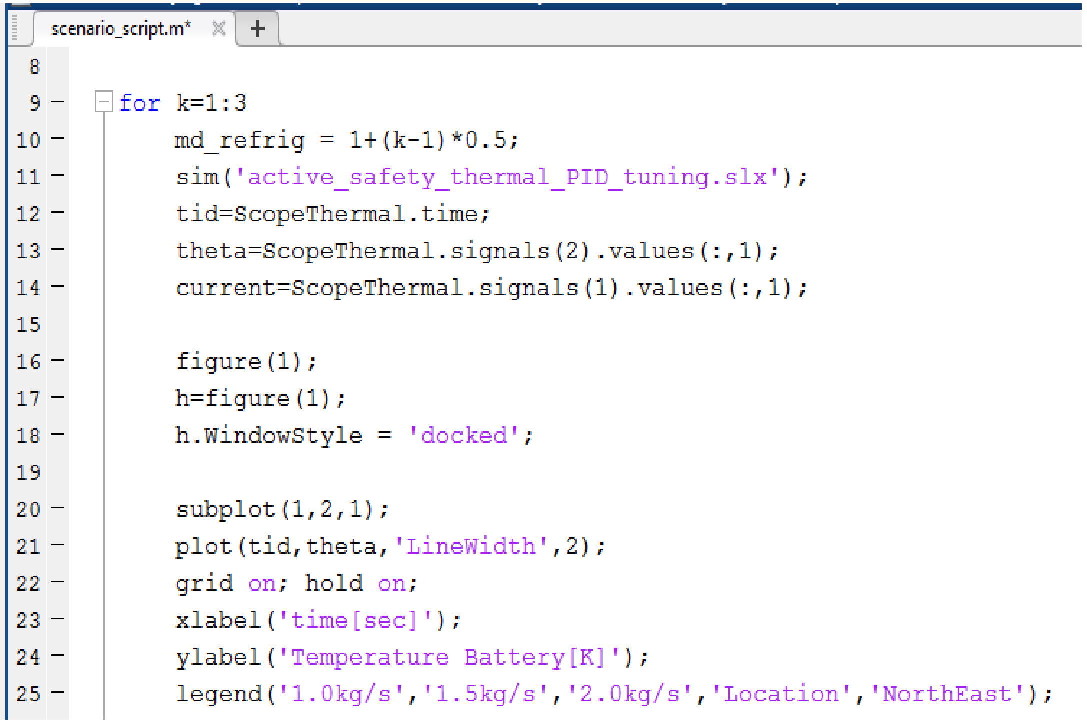
Vehicle Dynamics
var_a(3)>0

a1. ideal longitud tires
a2. MF longitud tires
a3. multibody 3D tires

** models available upon request*

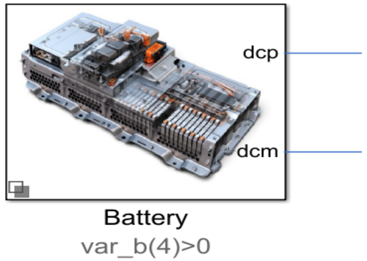
Active Safety – System Design

- Refine requirements
- Set-up test scenarios
- Report & align with others

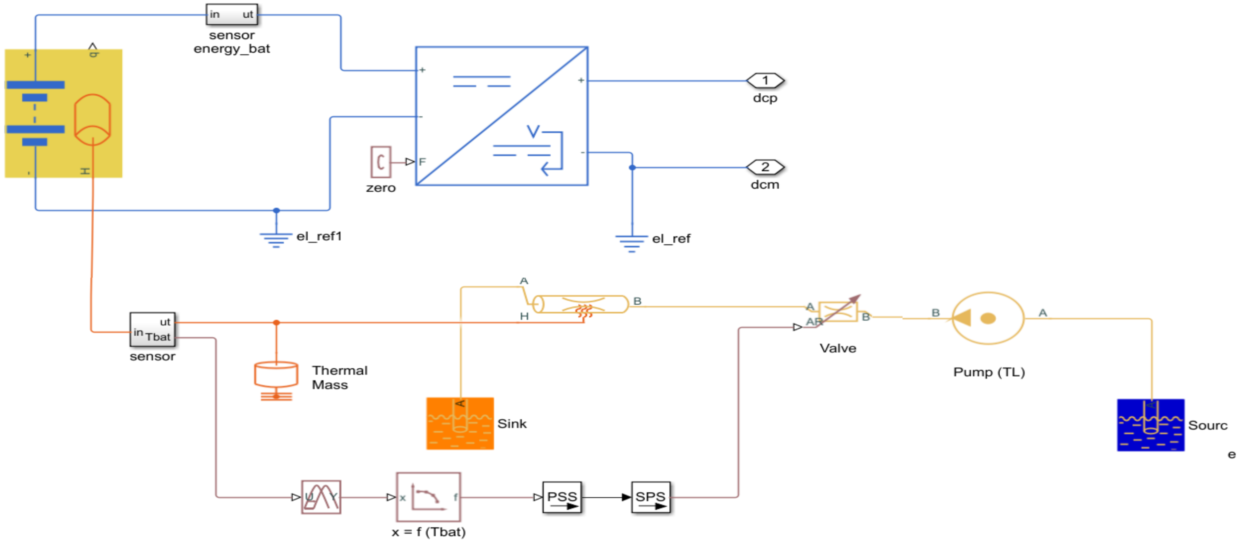


```

for k=1:3
    md_refrig = 1+(k-1)*0.5;
    sim('active_safety_thermal_PID_tuning.slx');
    tid=ScopeThermal.time;
    theta=ScopeThermal.signals(2).values(:,1);
    current=ScopeThermal.signals(1).values(:,1);

    figure(1);
    h=figure(1);
    h.WindowStyle = 'docked';

    subplot(1,2,1);
    plot(tid,theta,'LineWidth',2);
    grid on; hold on;
    xlabel('time[sec]');
    ylabel('Temperature Battery[K]');
    legend('1.0kg/s','1.5kg/s','2.0kg/s','Location','NorthEast');
```

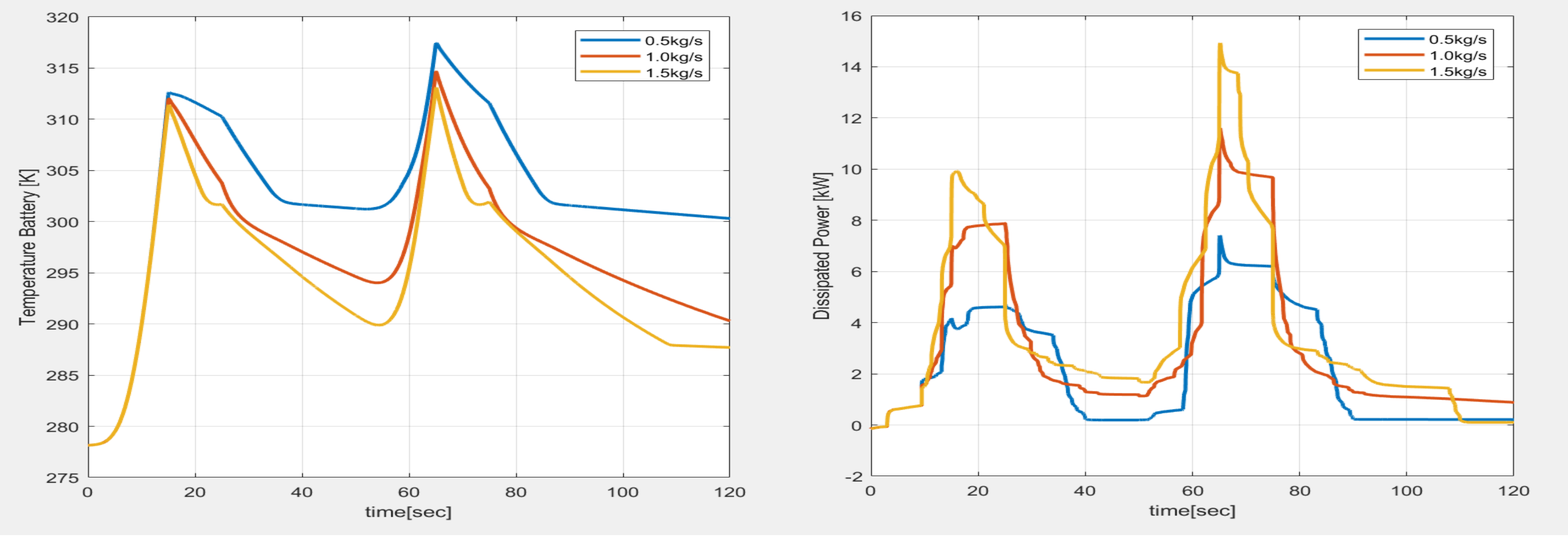
Active Safety >> System Design -> Battery Cooling



b1. DC source
b2. thermal cooling - ideal power
b3. fluid cooling - ideal flow source
b4. fluid cooling - valve-based

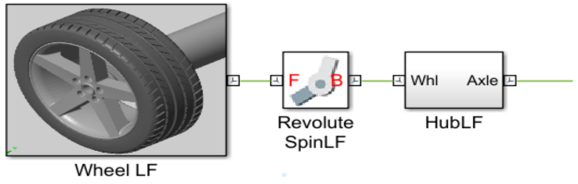


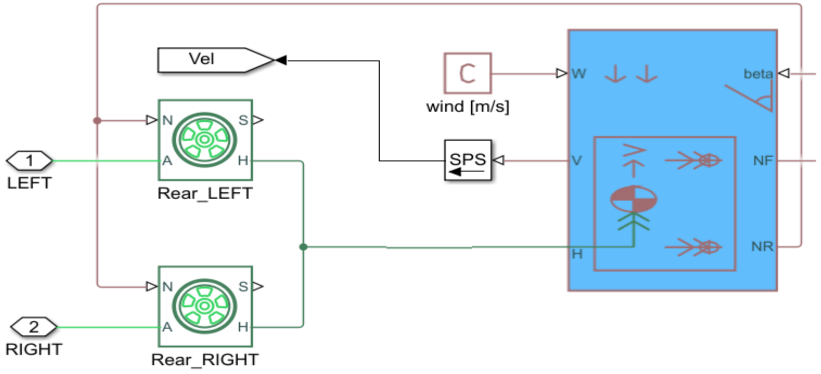
Battery
Temperature
(K)



Dissipated
Heat
(kW)

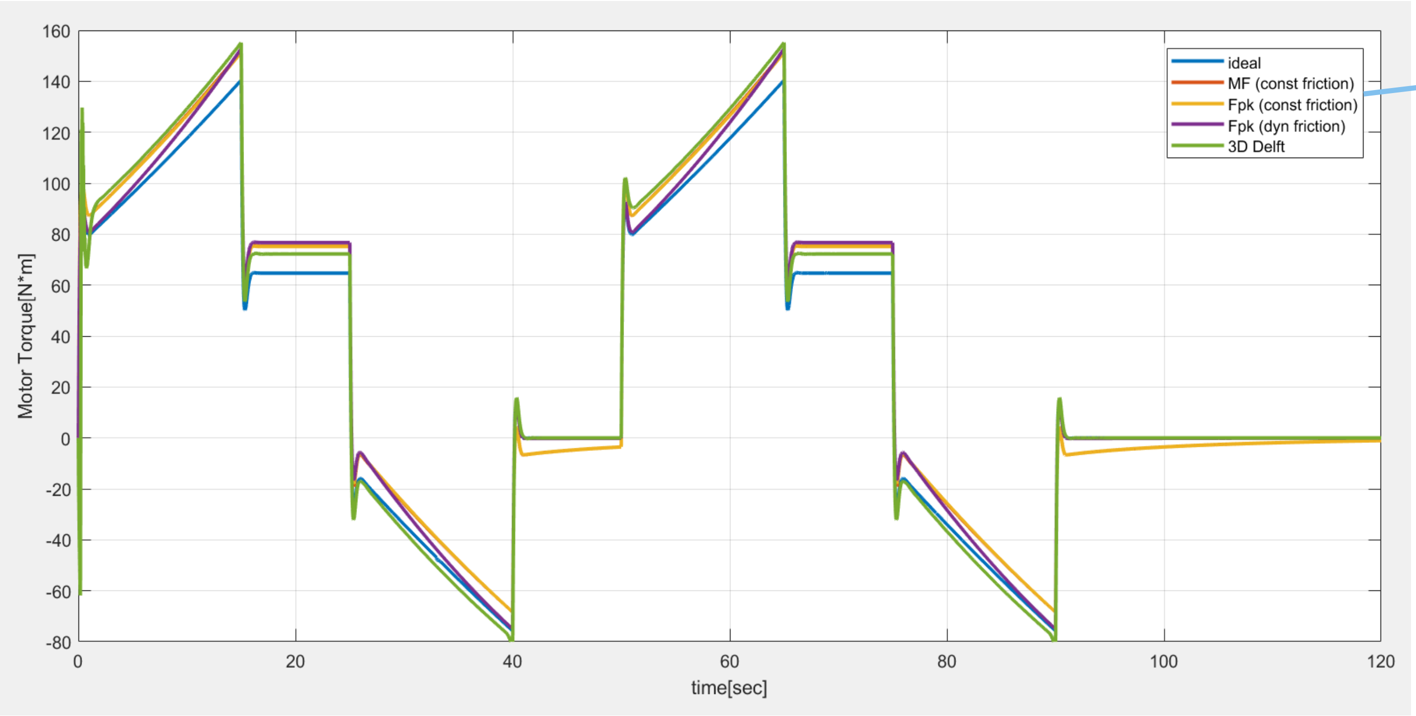
Active Safety >> System Design -> Vehicle Dynamics





a1. ideal longitud tires
a2. MF coefficients >> constant friction
a3. Fpk >> constant friction
a4. Fpk >> [pv]-dependent friction
a3. multibody 3D tires

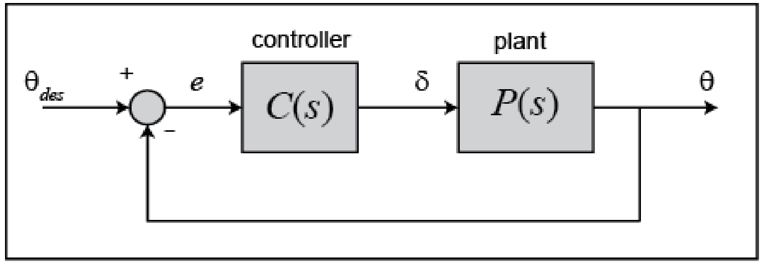
Tractive torque



Tire model assumption (model fidelity)

2. Algorithm Development

Regulate – Tune – Supervise

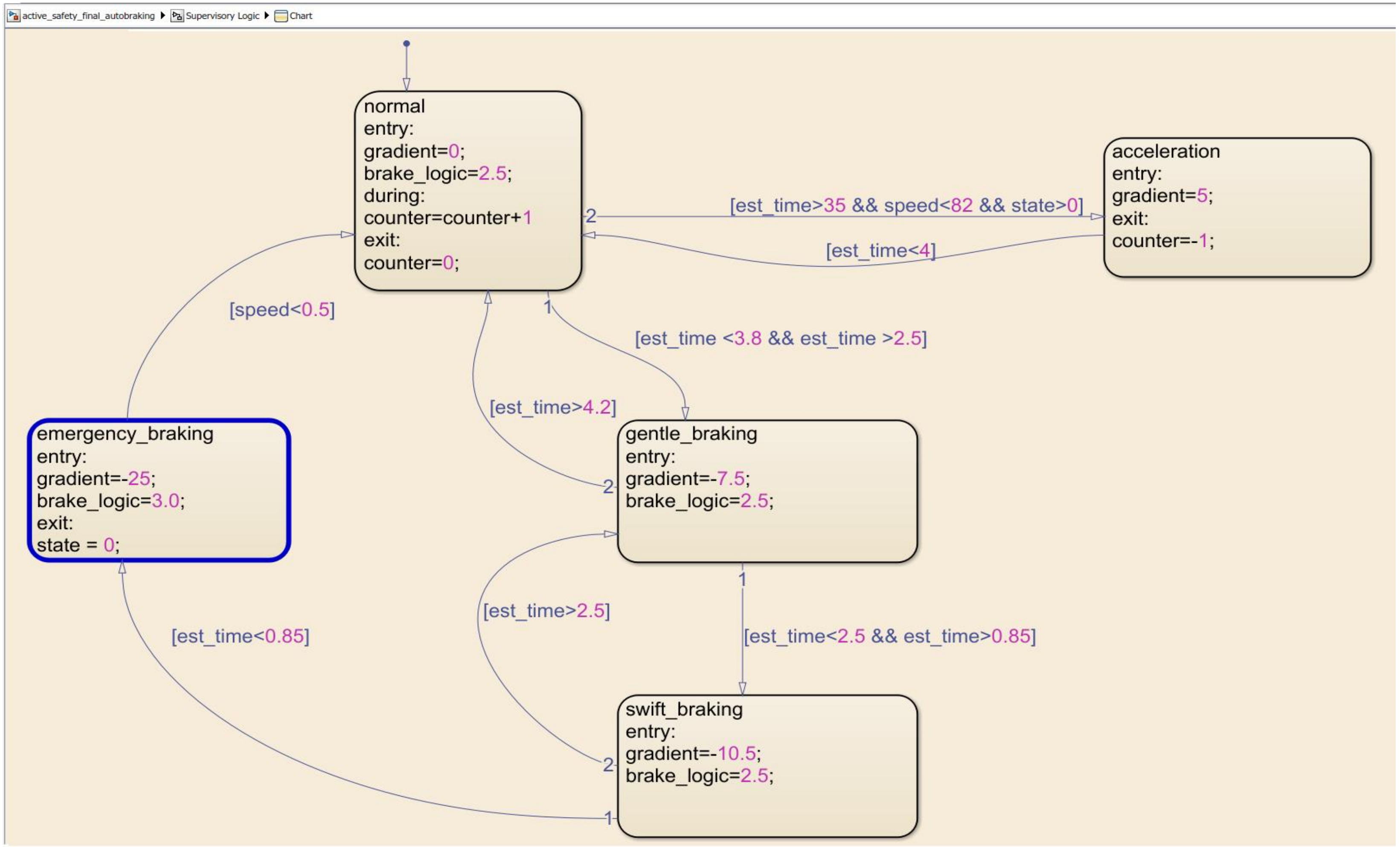


Active Safety – Algorithm Development

Structure and threshold values
are critical to supervisory logic design.
Physical state of all components is decisive.

Active Safety – Algorithm Development > Supervisory Logic

Structure and threshold values are critical to supervisory logic design.
Physical state of all components is decisive.



active_safety_final_autobraking
Supervisory Logic
Chart
[est_time>35 && speed<82 && state>0]
[est_time<4]
[speed<0.5]
[est_time <3.8 && est_time >2.5]
[est_time>4.2]
[est_time>2.5]
[est_time<0.85]
[est_time<2.5 && est_time>0.85]
1
2
1
2
1
2
1

```
normal
entry:
gradient=0;
brake_logic=2.5;
during:
counter=counter+1
exit:
counter=0;
```

```
acceleration
entry:
gradient=5;
exit:
counter=-1;
```

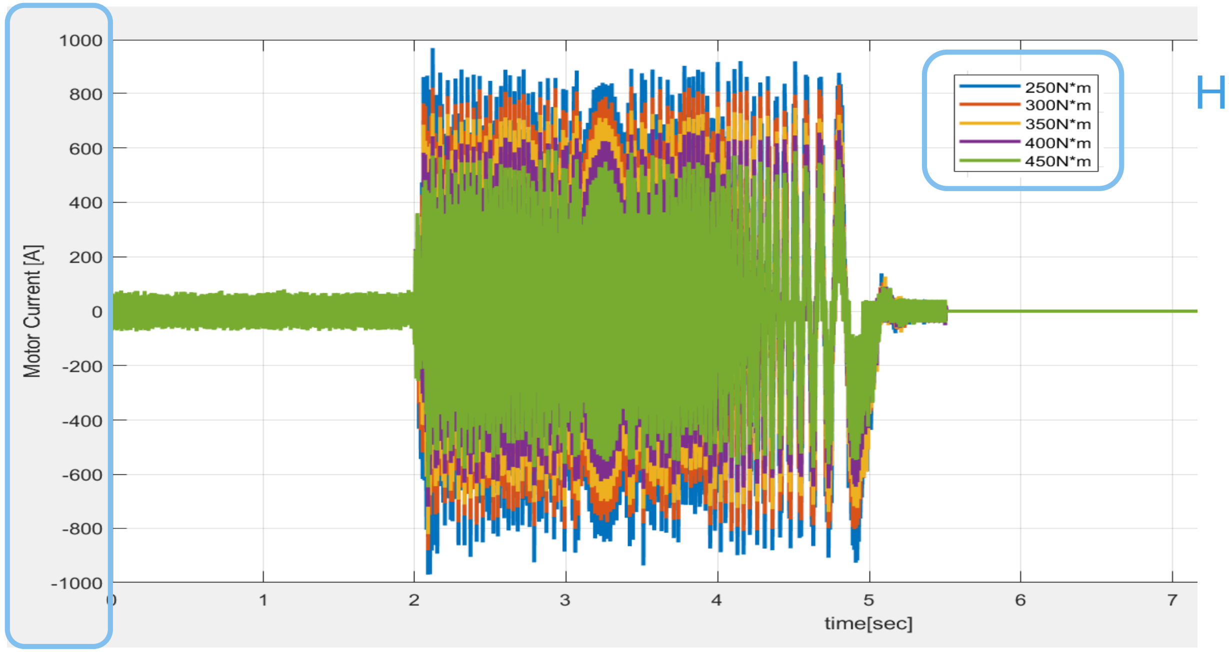
```
emergency_braking
entry:
gradient=-25;
brake_logic=3.0;
exit:
state = 0;
```

```
gentle_braking
entry:
gradient=-7.5;
brake_logic=2.5;
```

```
swift_braking
entry:
gradient=-10.5;
brake_logic=2.5;
```

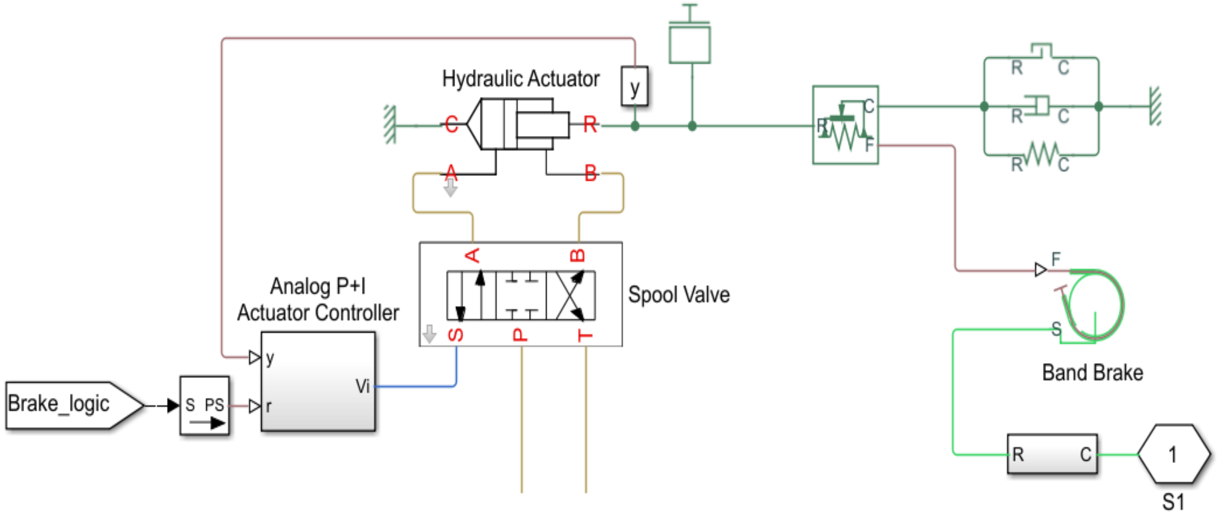
Active Safety – Algorithm Development -> braking

What is the trade-off
between hydraulic
and regenerative braking?



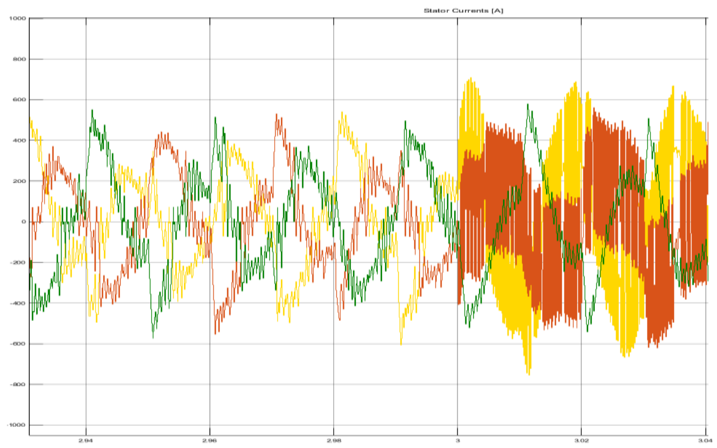
Hydraulic brake capacity

Regenerative motor currents

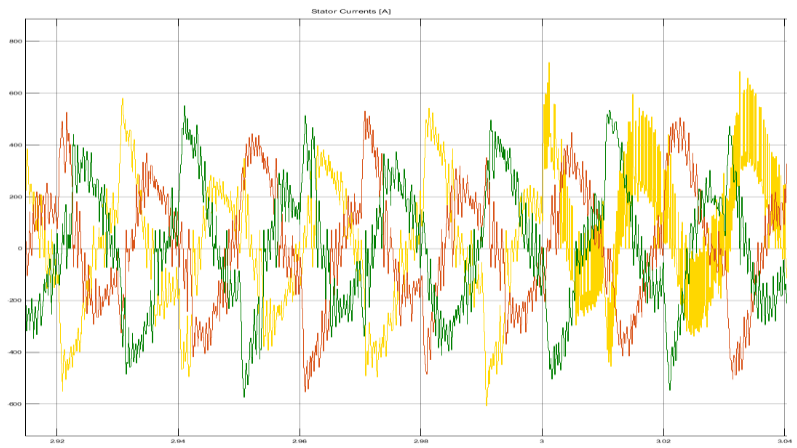


Are physical boundaries trespassed?
Is safety guaranteed?

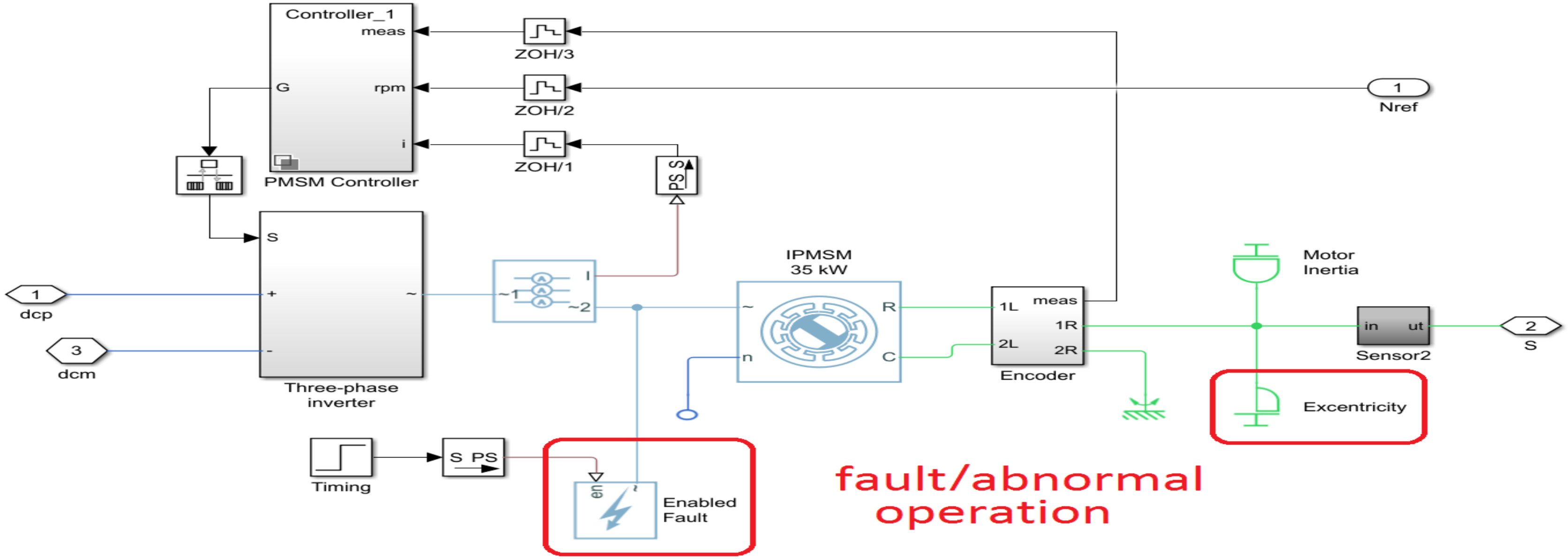
Active Safety - Prognosis > fault/degradation signature(s)



fault (a-b)

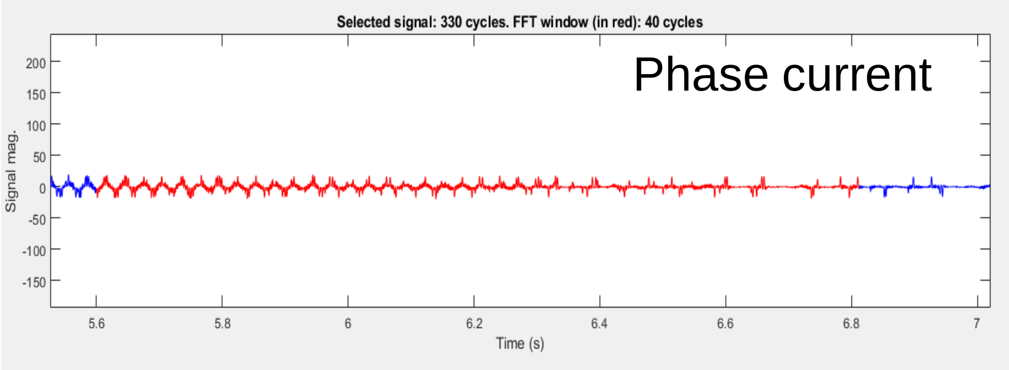


fault (a-ground)

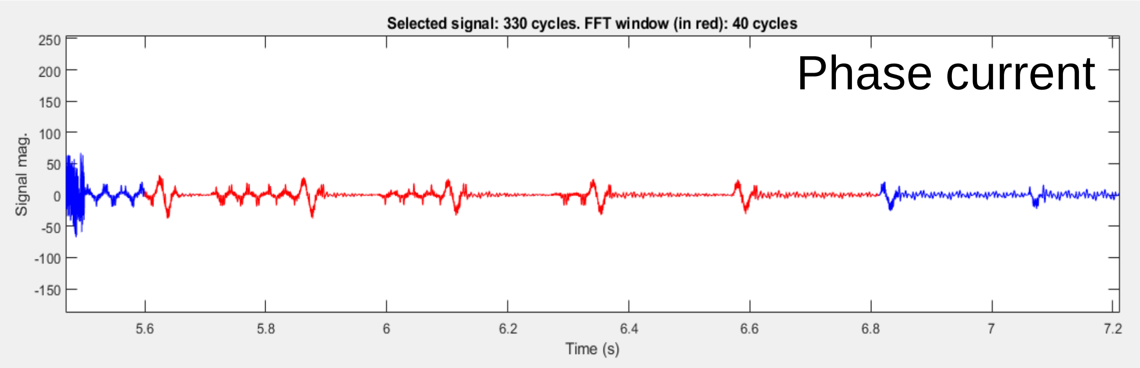


fault/abnormal operation

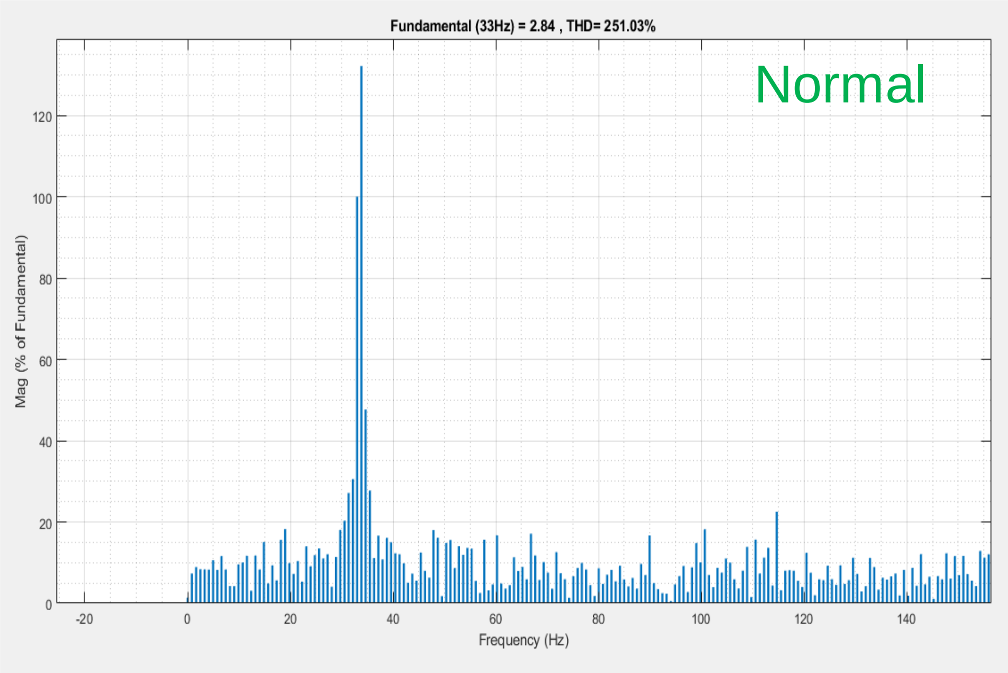
Active Safety - Prognosis > fault/degradation signature(s)



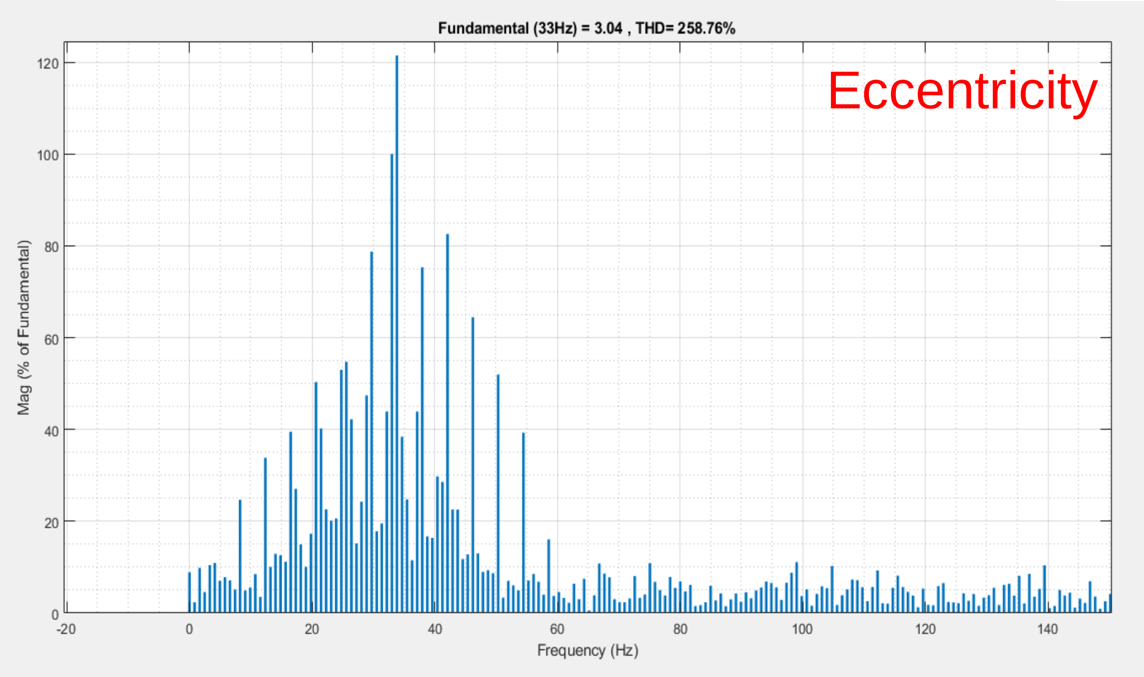
Phase current



Phase current

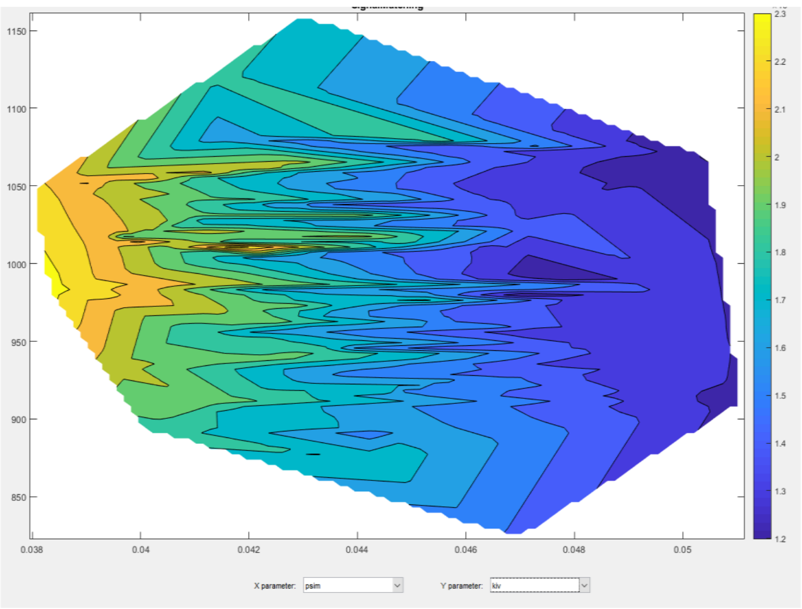


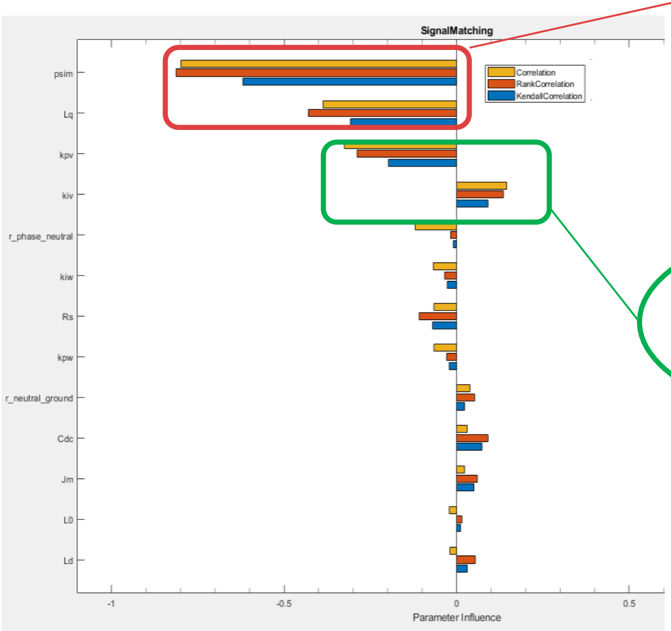
Normal



Eccentricity

Active Safety - Prognosis > Sensitivity & Calibration



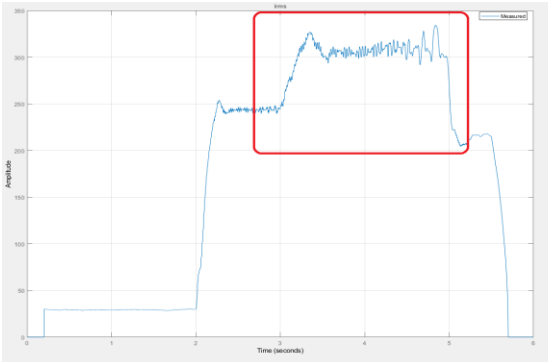


Motor flux constant & inductance highly influence current

agressive PI gains enhance the fault signature

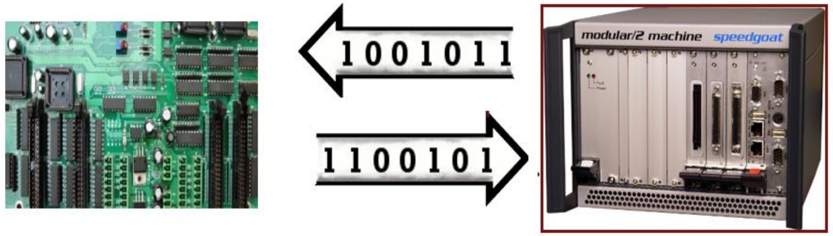
MonteCarlo simulations (Sensitivity Analysis with Parallel Computing)

How is the signature quantity i.e. current affected by deviations/uncertainties In physical/control properties?



3. Testing

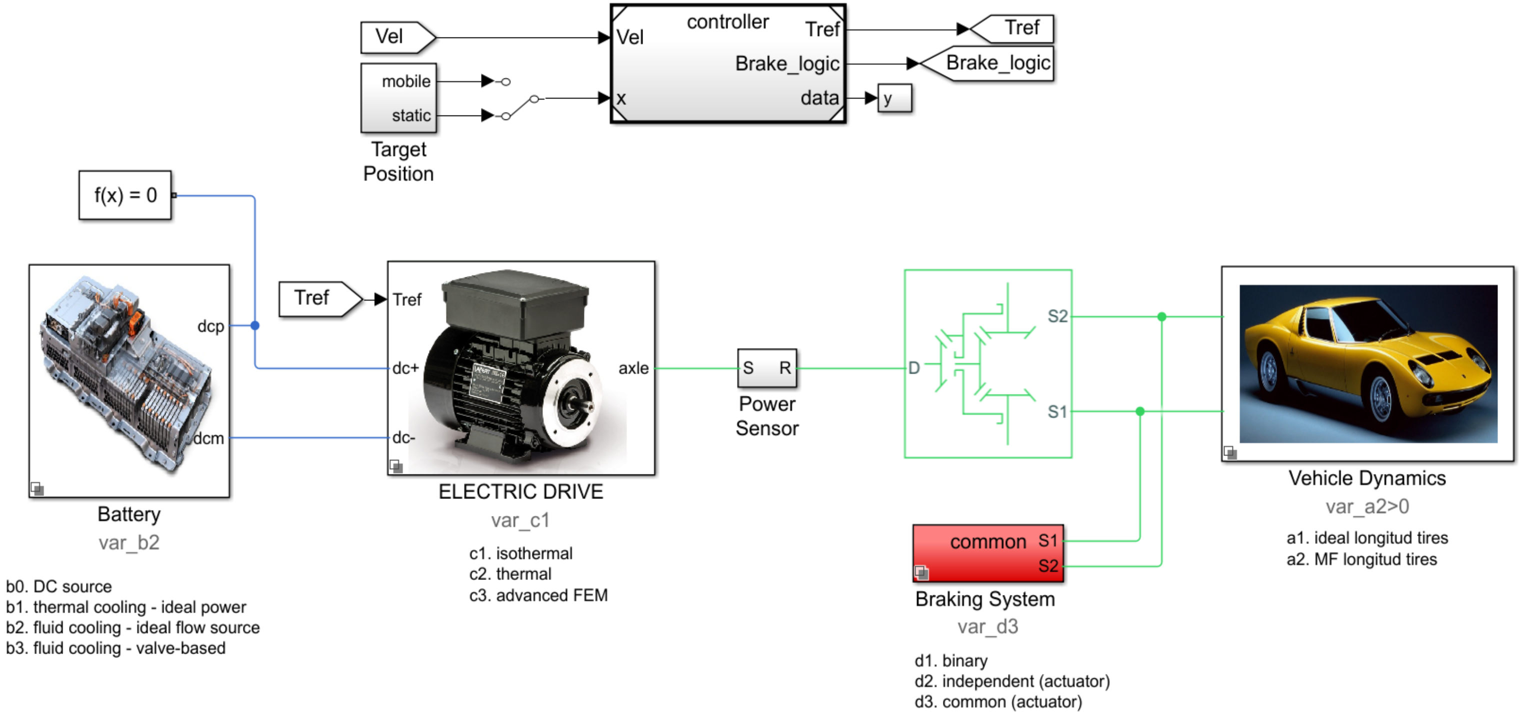
Verify your solution in real-time



Active Safety - Real-time Testing

Prepare model for real-time testing

a. golden reference >> controller (fixed-step) + Plant (variable-step)
b. controller + Plant (fixed-step solver)
c. optimal solution >> robustness vs. speed
d. test on hardware (Speedgoat)



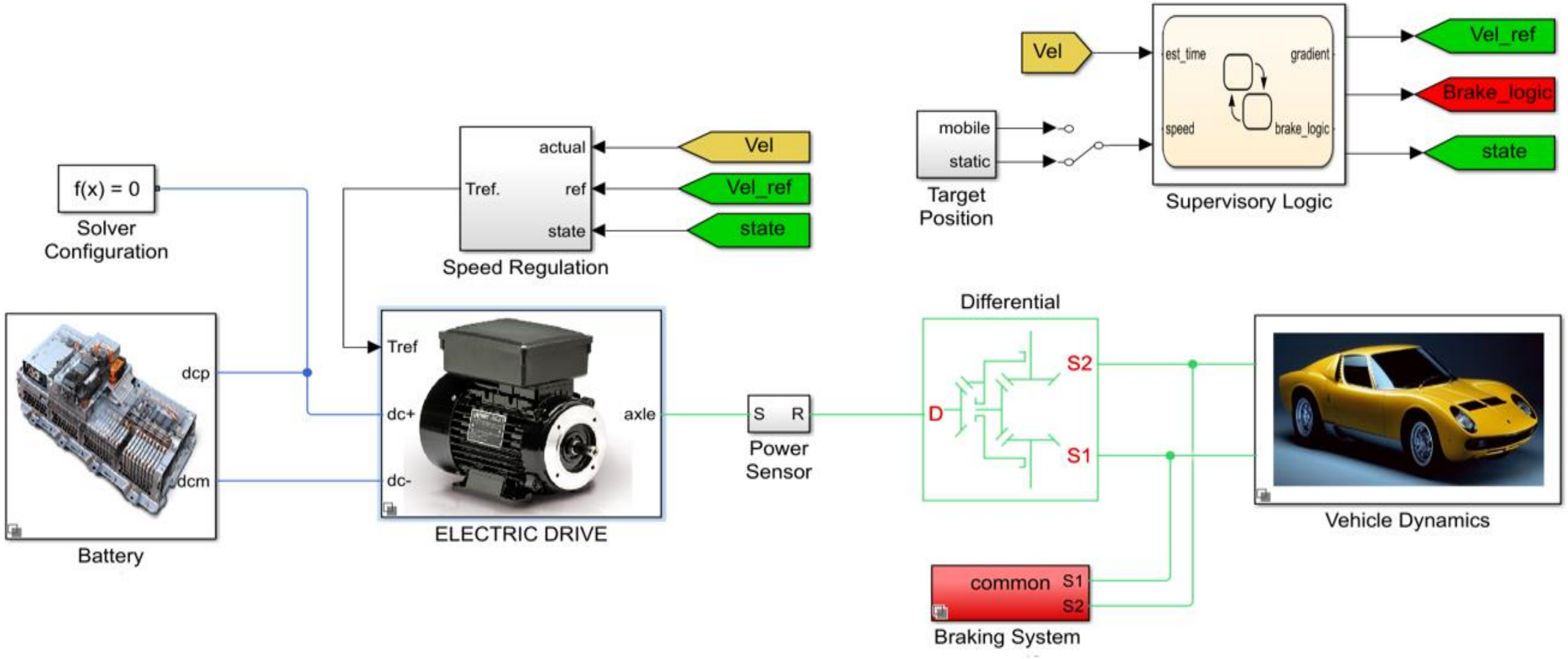
b0. DC source
b1. thermal cooling - ideal power
b2. fluid cooling - ideal flow source
b3. fluid cooling - valve-based

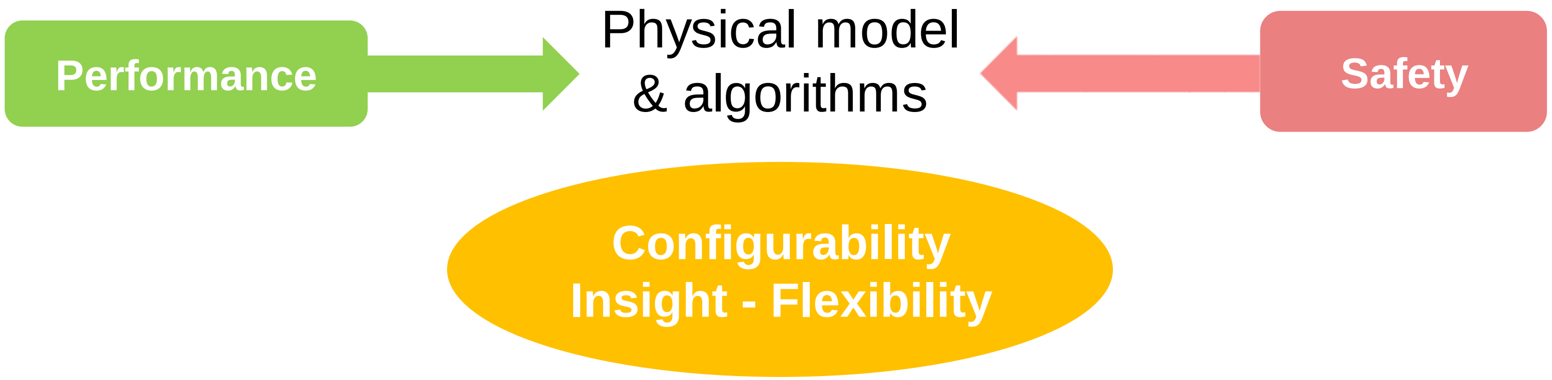
c1. isothermal
c2. thermal
c3. advanced FEM

d1. binary
d2. independent (actuator)
d3. common (actuator)

a1. ideal longitud tires
a2. MF longitud tires

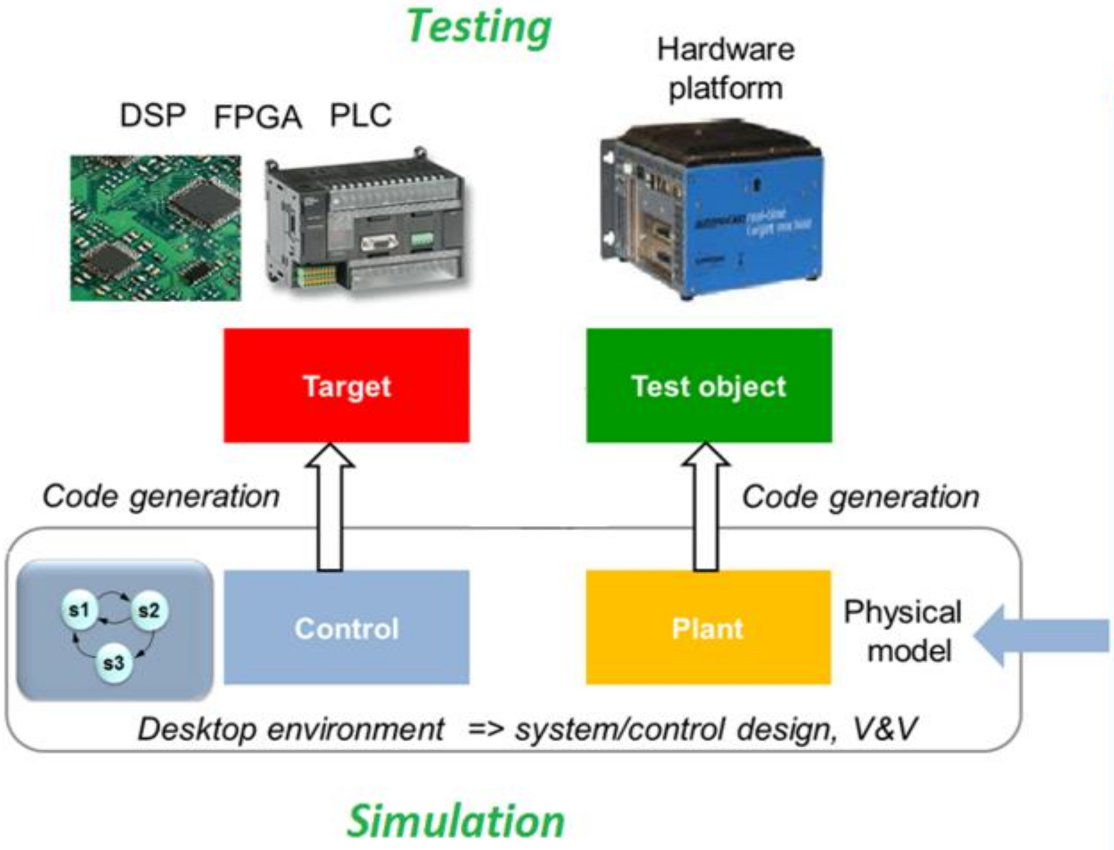
Conclusions


f(x) = 0
Solver Configuration
actual
Tref.
ref
state
Vel
Vel_ref
state
Speed Regulation
Vel
est_time
gradient
speed
brake_logic
Vel_ref
Brake_logic
state
mobile
static
Target Position
Supervisory Logic
Differential
dcp
dcm
Battery
Tref
dc+
dc-
axle
ELECTRIC DRIVE
S
R
Power Sensor
D
S2
S1
Vehicle Dynamics
common
S1
S2
Braking System


Performance
Physical model
& algorithms
Safety
Configurability
Insight - Flexibility

MATLAB/Simulink offers a unique environment for data analytics & embedded development

MATLAB
SIMULINK



Testing

DSP FPGA PLC

Hardware platform

Target

Test object

Code generation

Code generation

s1 s2 s3

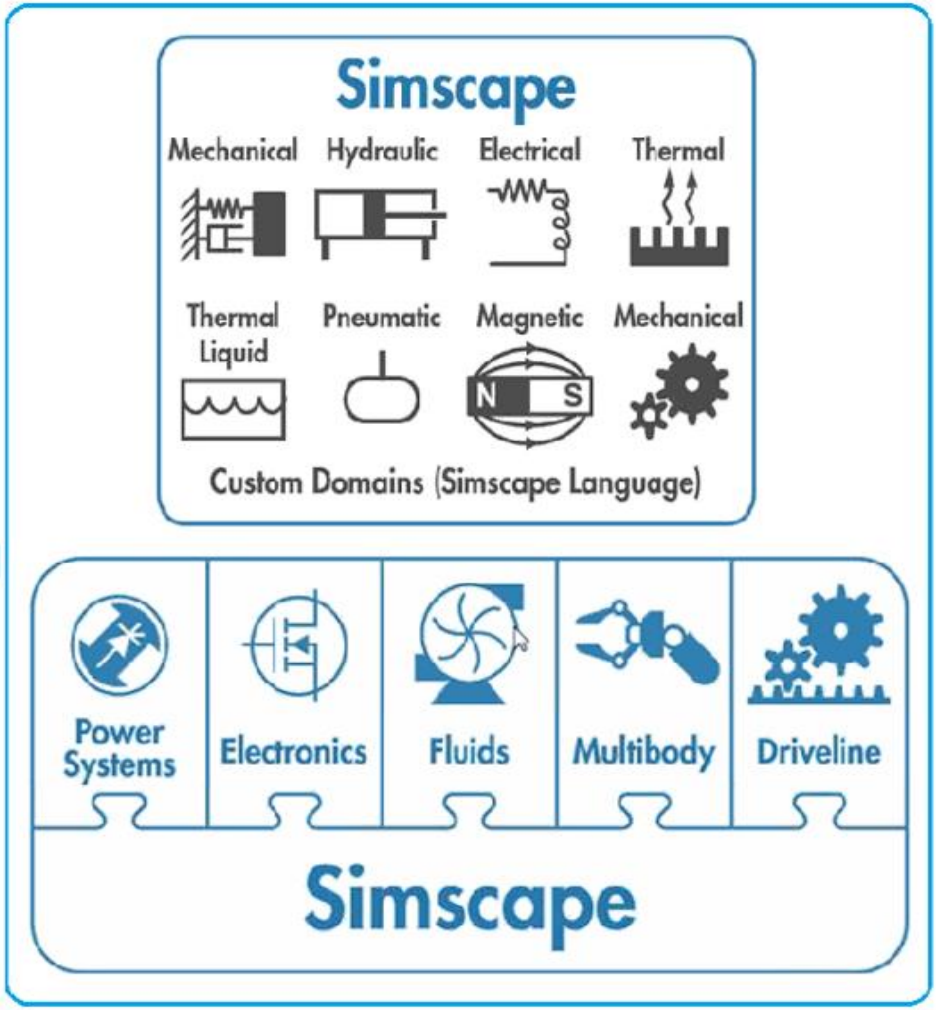
Control

Plant

Physical model

Desktop environment => system/control design, V&V

Simulation



Simscape

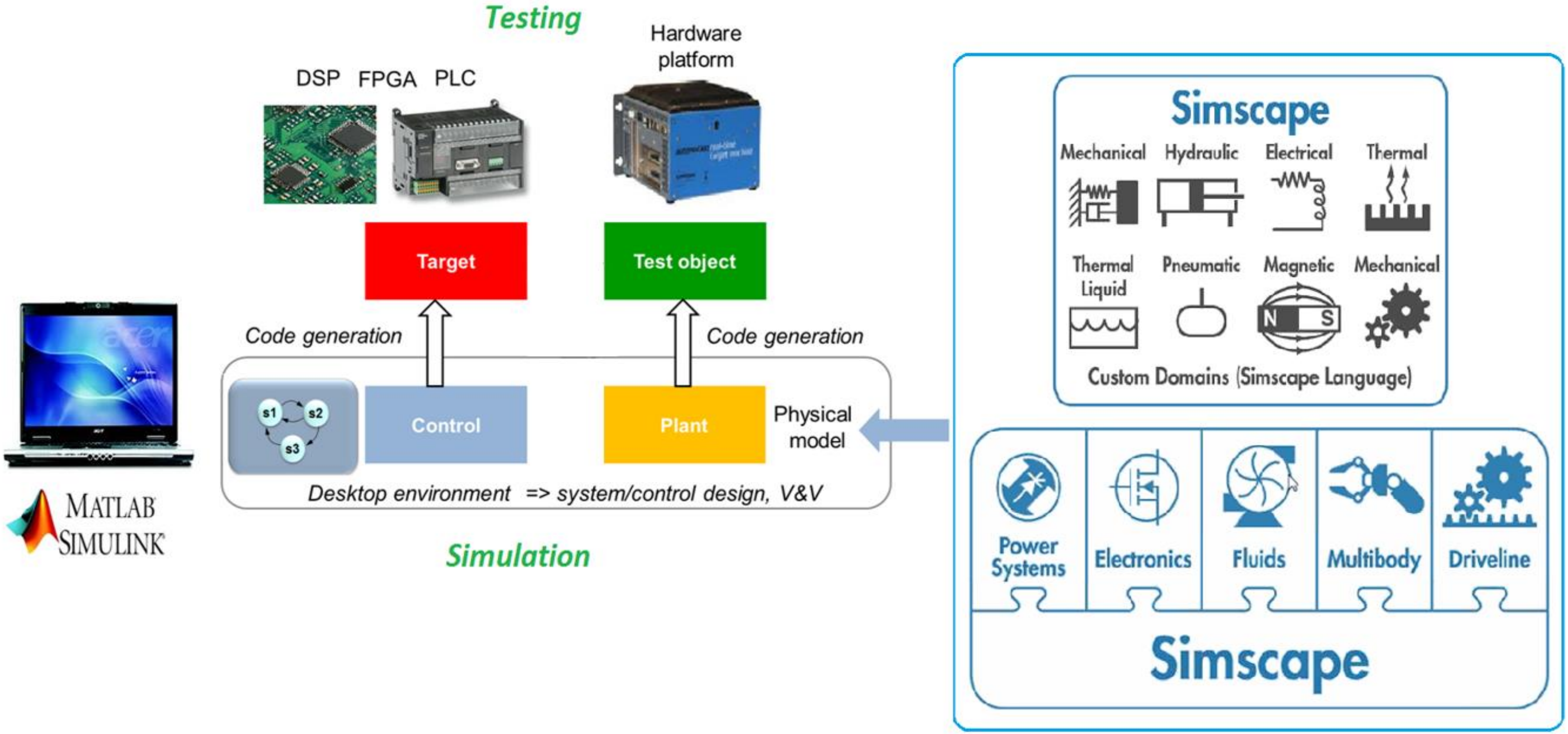
Mechanical
Hydraulic
Electrical
Thermal
Thermal Liquid
Pneumatic
Magnetic
Mechanical

Custom Domains (Simscape Language)

Power Systems
Electronics
Fluids
Multibody
Driveline

Simscape

Partnership with MathWorks
reduces risk and accelerates the adoption process


Testing
Hardware platform
DSP
FPGA
PLC
Target
Test object
Code generation
Code generation
s1
s2
s3
Control
Plant
Physical model
Desktop environment => system/control design, V&V
Simulation
MATLAB
SIMULINK
Simscape
Mechanical
Hydraulic
Electrical
Thermal
Thermal Liquid
Pneumatic
Magnetic
N
S
Mechanical
Custom Domains (Simscape Language)
Power Systems
Electronics
Fluids
Multibody
Driveline
Simscape