

## HOW TO USE MODEL-DRIVEN DEVELOPMENT FOR SMART HOSPITAL BOOMS AND SENSORLESS MOTOR CONTROL

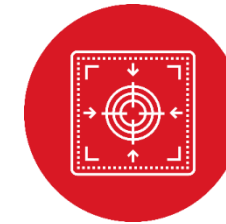


Embedded Software Engineer  
+31 88 334 33 23  
sjoerd.rozendal@kendrion.com

*Let's Connect!*



High System Efficiency



High-precision applications

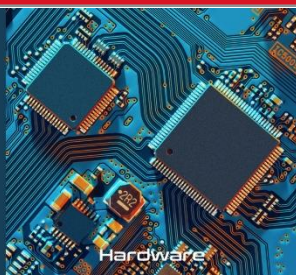


Working in Harsh Environments

*Learn more about Power & Motion Control*



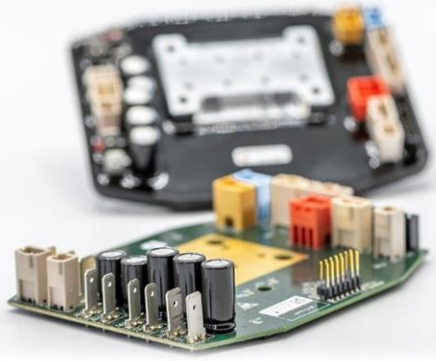
**D&E**  
EVENT



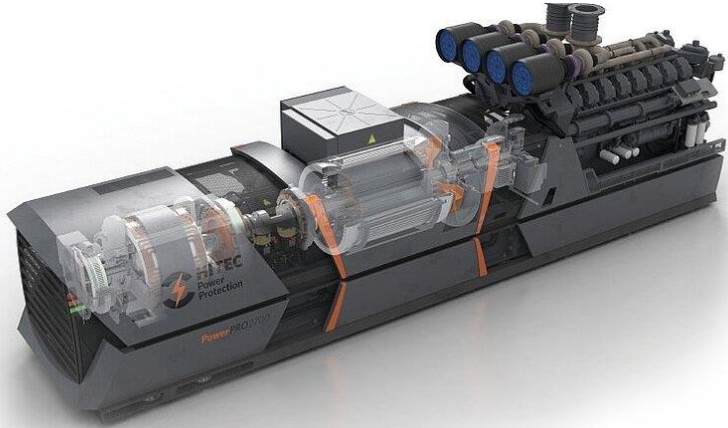
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch





# DEVELOPMENT SERVICE MISSION-CRITICAL APPLICATIONS



# 3T





D&E  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congressentrum 's-Hertogenbosch

# ABOUT 3T

## Our mission:

We continuously invest in our expertise in electronics and embedded systems to enable our clients offering the best possible product to the market.

-  100+ innovators
-  40+ years
-  Enschede, Eindhoven & Drachten
-  Since 2021 part of Kendrion



**D&E**  
EVENT

Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch



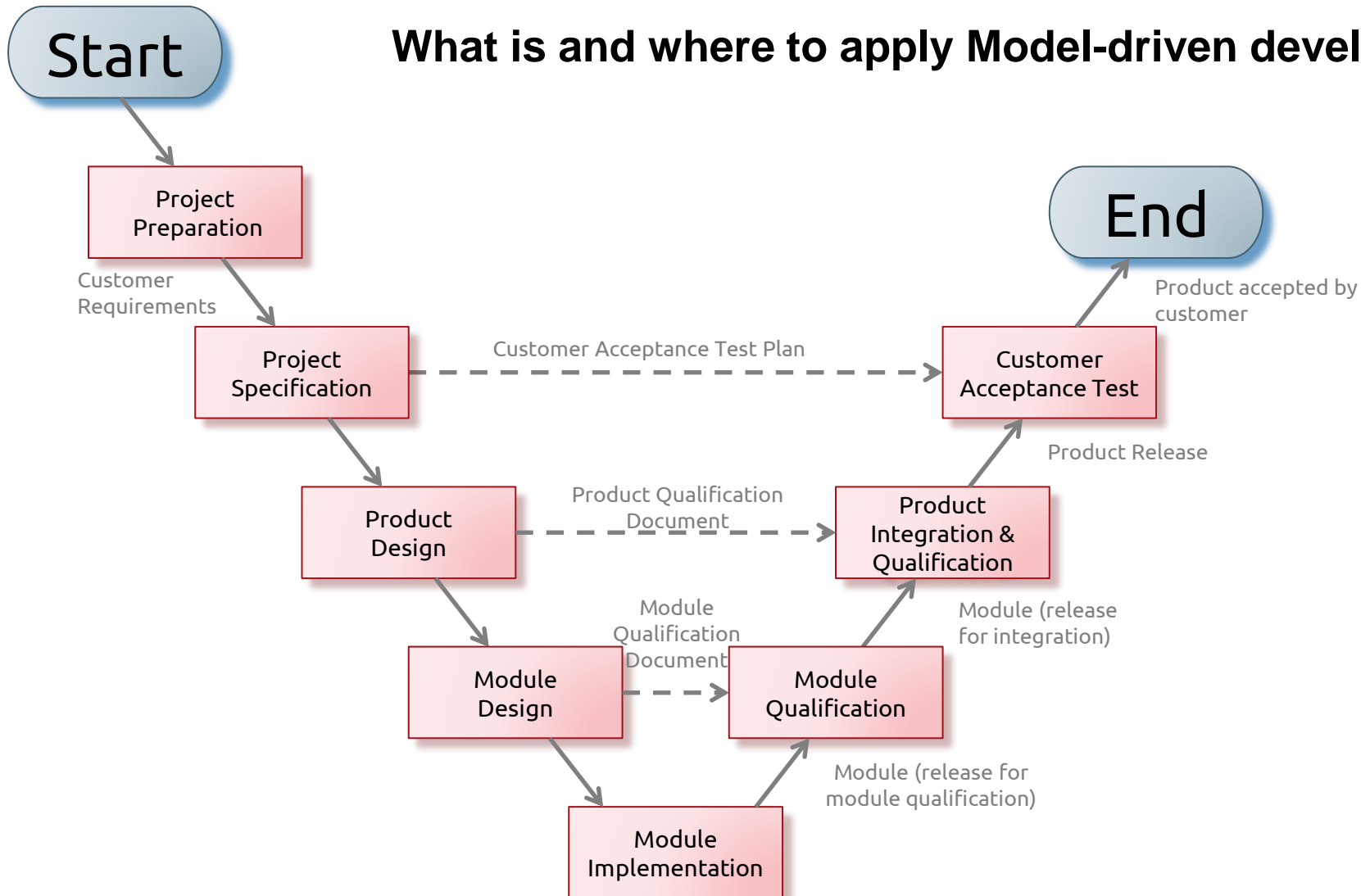


# Demonstration of MDD and Code Generation by two project examples.



# THE DEVELOPMENT PROCESS

## What is and where to apply Model-driven development (MDD)



MDD as a framework to manage:

- Requirements
- Complexity
- Failure analysis

**3T**

D&E  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch





# Project: Skytron Freedom

**3T**

**D&E**  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congressentrum 's-Hertogenbosch

# PROJECT: SKYTRON FREEDOM

## The Challenge

- Easy positioning and movement of arm
  - Drift due to mechanical bending
  - Interacting with devices on arm cause movement
  - Accidental impacts may have large consequences
- Intentional friction in joints
  - Takes quite some effort to move arm around
  - Wear on friction mechanism



**3T**

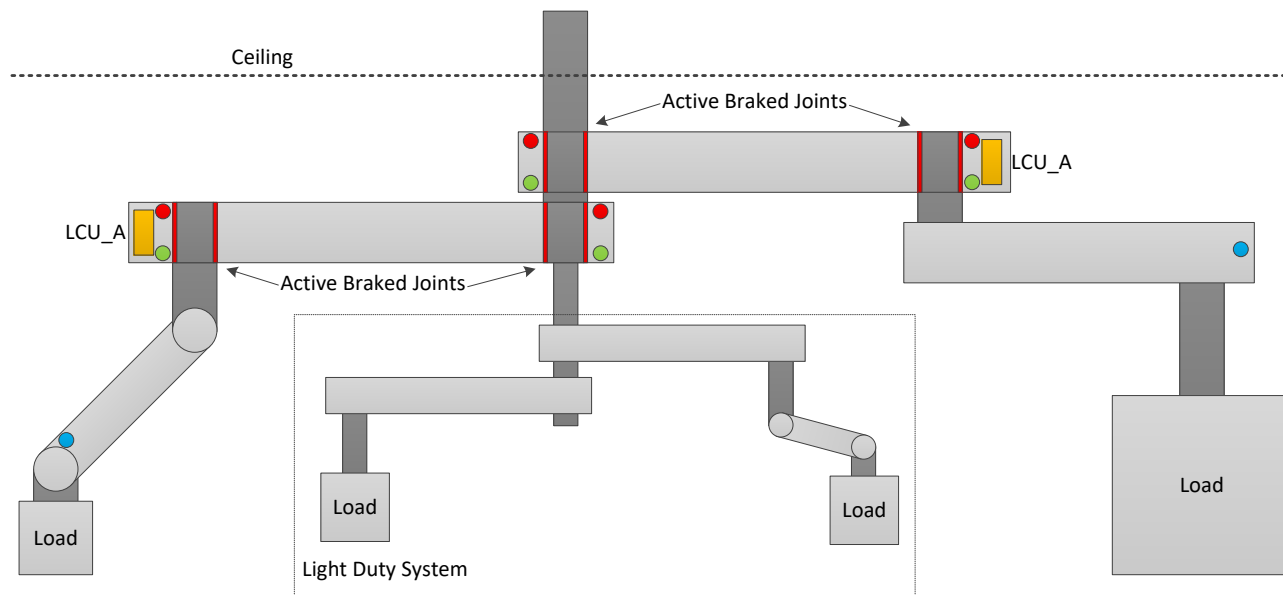
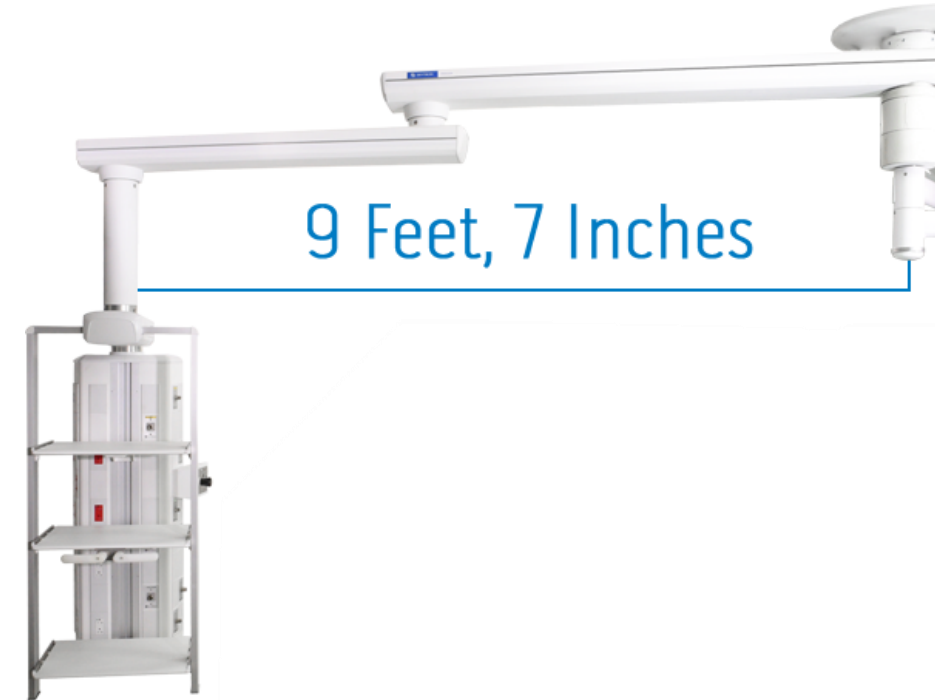
**D&E**  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch

# PROJECT: SKYTRON FREEDOM

## The Solution:

- Smart Braking System
  - Easy to move when needed
  - Stable when stationary
  - No special user interaction needed
  - Sensor system to detect user intention



# 3T

D&E  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch

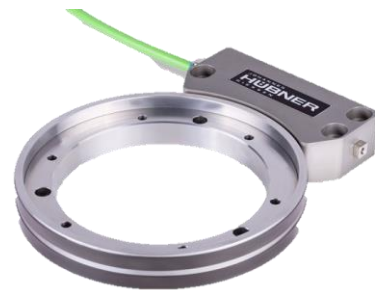
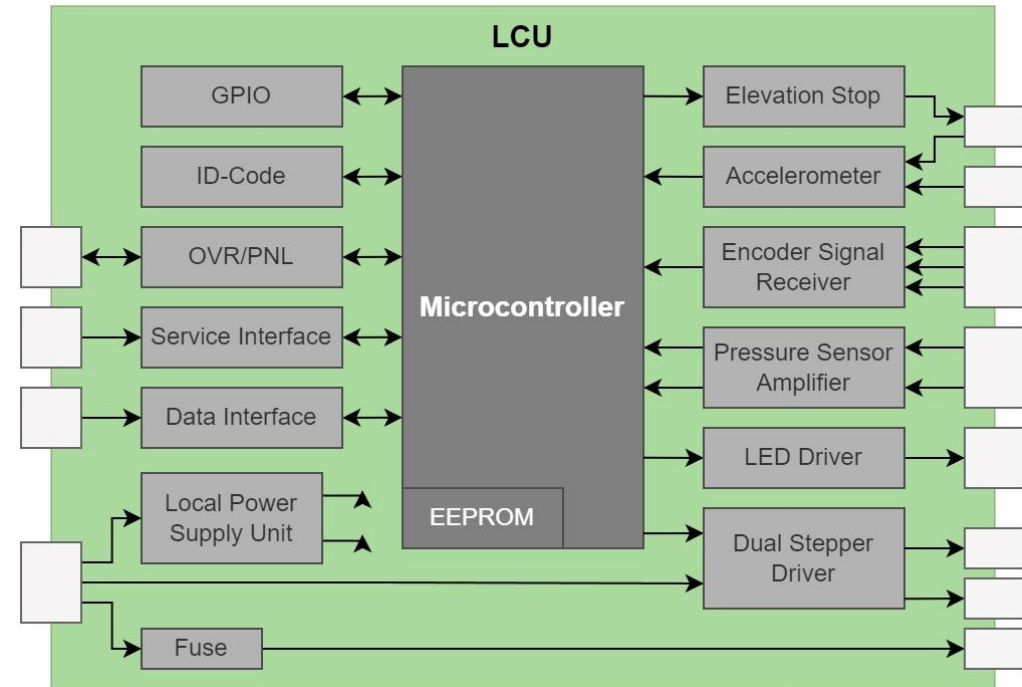


# PROJECT: SKYTRON FREEDOM

## Specification

- Ensure all hardware interfaces are accessible and controllable
- The desired “Excellent user experience” of the end-product

Mechanical design of brakes and arm were not available at the start of the project.



# 3T

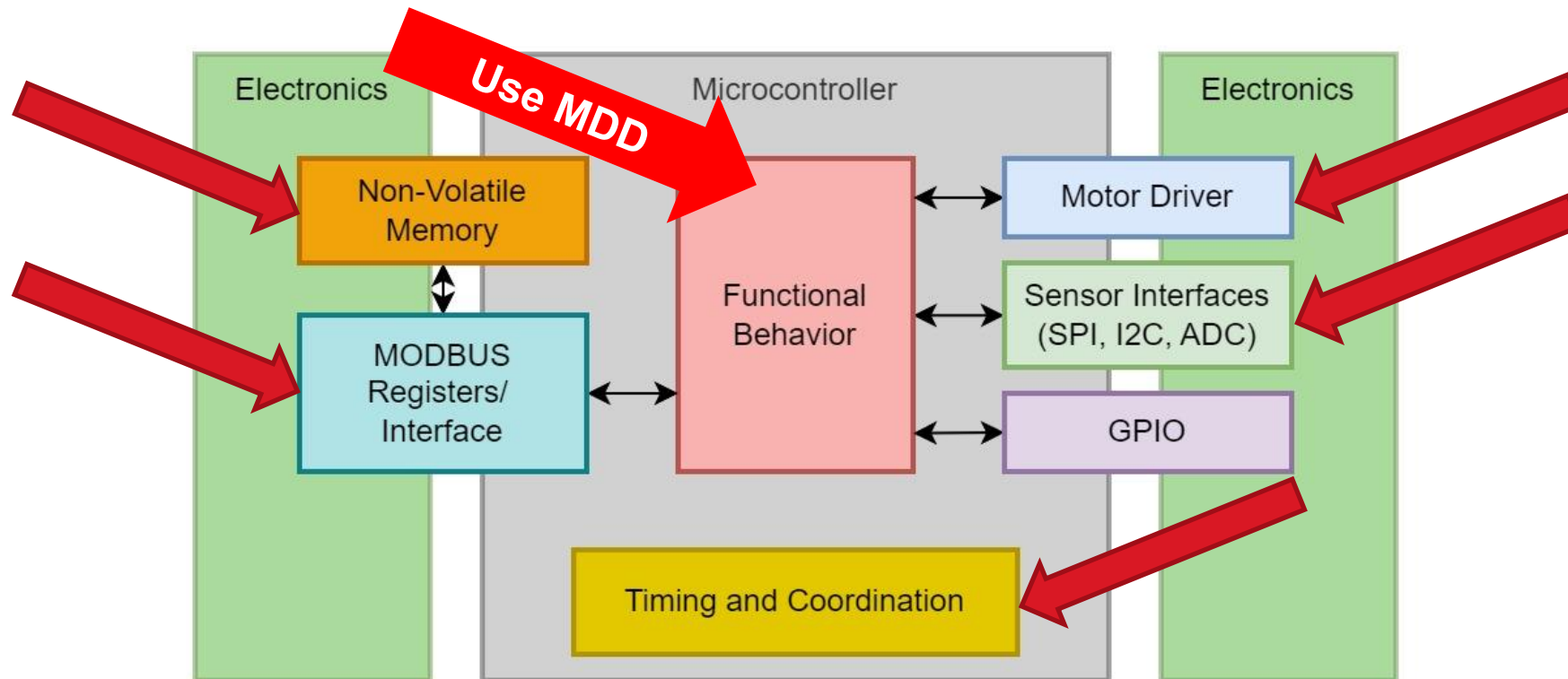
D&E  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch

# PROJECT: SKYTRON FREEDOM

## Development Strategy: Reduce development time and risk

- Use of existing design experience for electronics and software
- Use MDD for functional behavior design and implementation



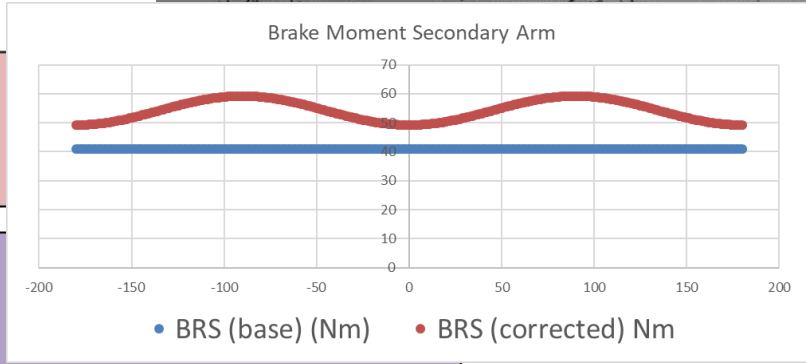
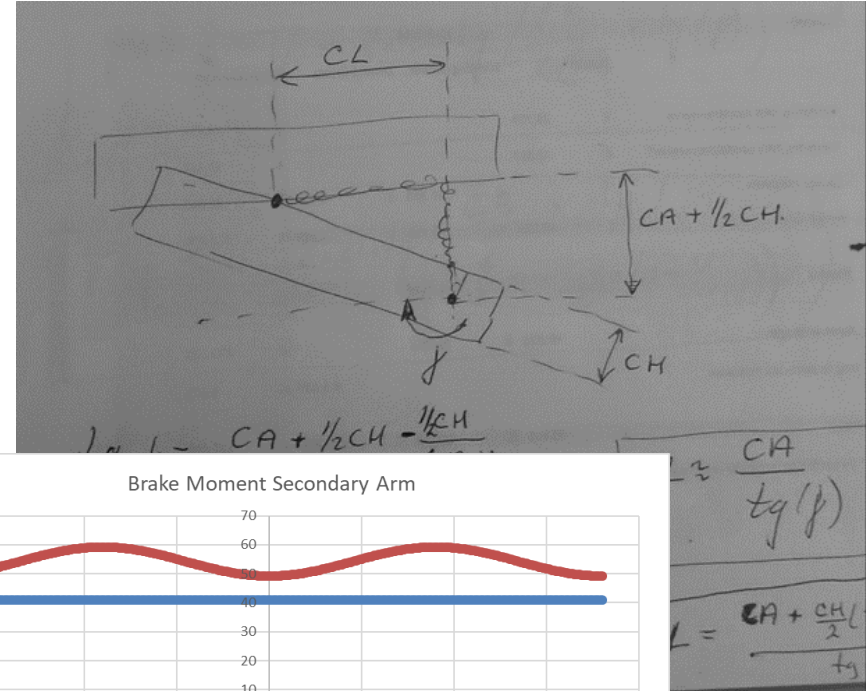
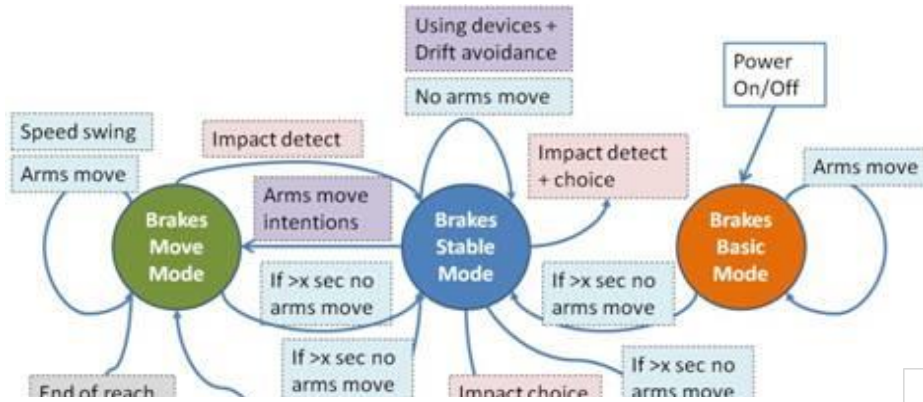
# 3T

D&E  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch

# PROJECT: SKYTRON FREEDOM

## Customer's Language



### Aanpassen Brake Moment:

**BRP-mimp:** Brake = ALS(IMDM=true; BRP-brake-max; Brake (last))

**BRP-stab:** Brake = ALS(STAB=true; BRP-stab; Brake (last))

### Aanpassen Brake Moment:

**BRS-mimp:** Brake = ALS(IMDM=true; BRS-brake-max; Brake (last))

**BRS-stab:** Brake = ALS(STAB=true; BRS-stab; Brake (last))

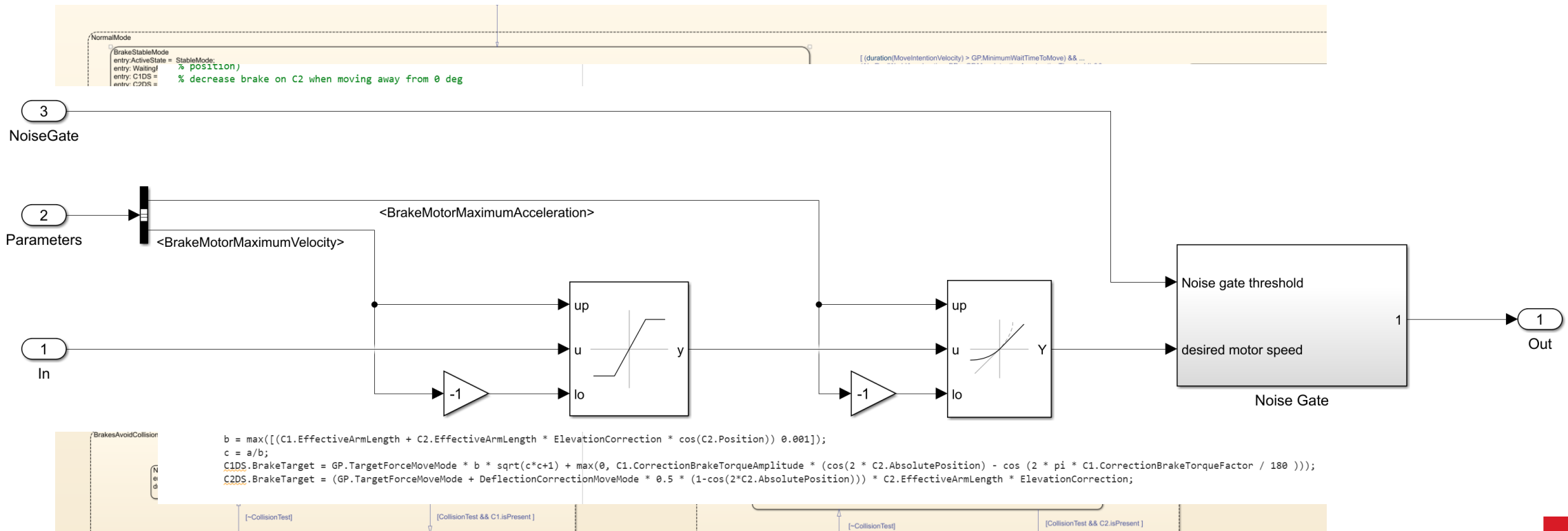
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
	(mg)	(mg)	(mg)	(rad)	(rad)	(rad)	(rad/msec)	(rad/msec)	(rad/msec)		MOVE	MOVE	ROTATE	ROTATE		STABLE
	Ax	Ay	Az	Ha	Hb	Hc	Ha	Hb	Hc		intent	active	pendant	carrier		pendant
47	3.267	-1.930	0.000	-1.300	-1.981	-1.539	3.728	22.371	22.371							JA
48	3.118	-1.782	0.000	-1.300	-1.981	-1.545	3.728	26.099	22.371							JA
49	2.970	-1.633	-0.015	-1.300	-1.969	-1.545	0.000	33.556	26.099				JA			
50	3.118	-1.782	0.000	-1.300	-1.969	-1.551	0.000	37.284	29.827				JA			



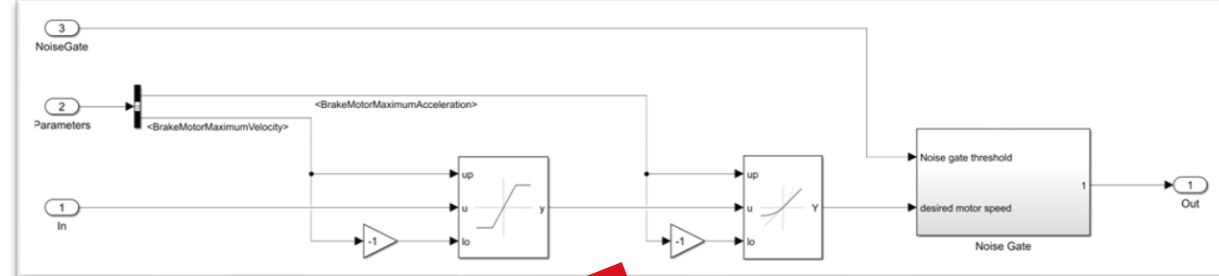
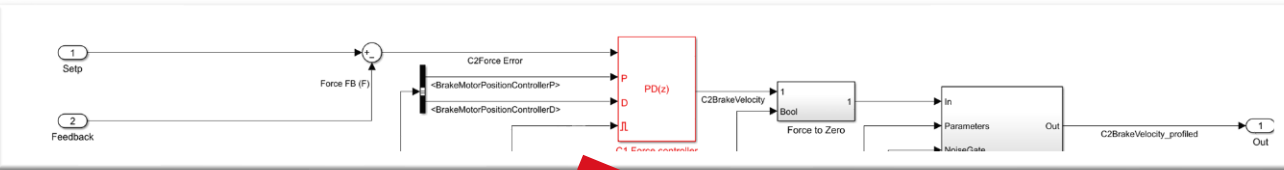


# PROJECT: SKYTRON FREEDOM

## Translation to model



# PROJECT: SKYTRON FREEDOM



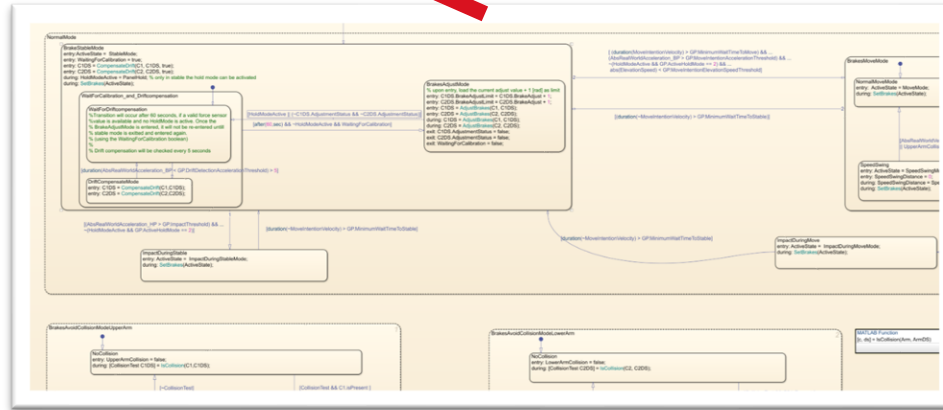
The Design of the Functional Behavior

```

% decrease brake on C1 when moving away from @ deg
if C2.Velocity * C2.AbsolutePosition > 0
    DeflectionCorrectionMoveMode = 135/250 * LoadParams.Weight;
else
    DeflectionCorrectionMoveMode = -1 * 135/250 * LoadParams.Weight * GP.TargetForceMoveMode / 200;
end

switch(inp)
case (InitMode, InitBrakesMode, BasicMode)
    a = C2.EffectiveArmLength * ElevationCorrection;
    b = C1.EffectiveArmLength;
    c = a/b;
    C1D5.BrakeTarget = 1.2 * GP.TargetForceStableMode * C1D5.TargetForceCompensation * b * sqrt(c*c+1);
    C2D5.BrakeTarget = 1.2 * GP.TargetForceStableMode * C2D5.TargetForceCompensation * C2.EffectiveArmLength * ElevationCorrection;
case StableMode
    a = C2.EffectiveArmLength * ElevationCorrection * sin(C2.Position);
    b = max([(C1.EffectiveArmLength + C2.EffectiveArmLength * ElevationCorrection * cos(C2.Position)) @.001]);
    c = a/b;
    C1D5.BrakeTarget = GP.TargetForceStableMode * C1D5.TargetForceCompensation * b * sqrt(c*c+1);
    C2D5.BrakeTarget = (GP.TargetForceMoveMode + DeflectionCorrectionMoveMode * 0.5 * (1-cos(2*C2.AbsolutePosition))) * C2.EffectiveArmLength * ElevationCorrection;
case MoveMode
    a = C2.EffectiveArmLength * ElevationCorrection * sin(C2.Position);
    b = max([(C1.EffectiveArmLength + C2.EffectiveArmLength * ElevationCorrection * cos(C2.Position)) @.001]);
    c = a/b;
    C1D5.BrakeTarget = GP.TargetForceMoveMode * b * sqrt(c*c+1) + max(@, C1.ConnectionBrakeTorqueAmplitude * (cos(2 * C2.AbsolutePosition) - cos(2 * pi * C1.ConnectionBrakeTorqueFactor / 180)));
    C2D5.BrakeTarget = (GP.TargetForceMoveMode + DeflectionCorrectionMoveMode * 0.5 * (1-cos(2*C2.AbsolutePosition))) * C2.EffectiveArmLength * ElevationCorrection;

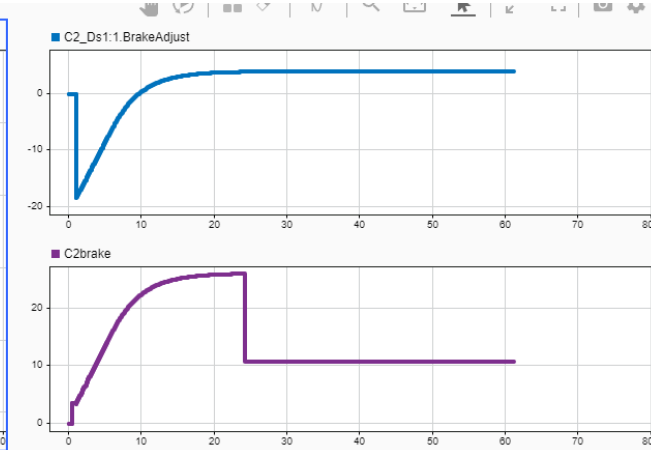
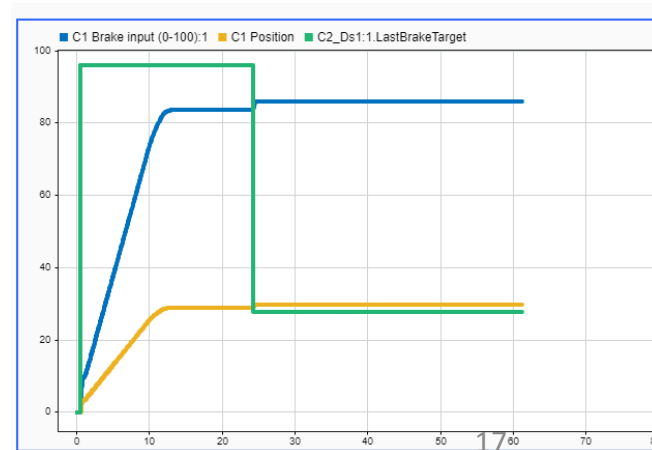
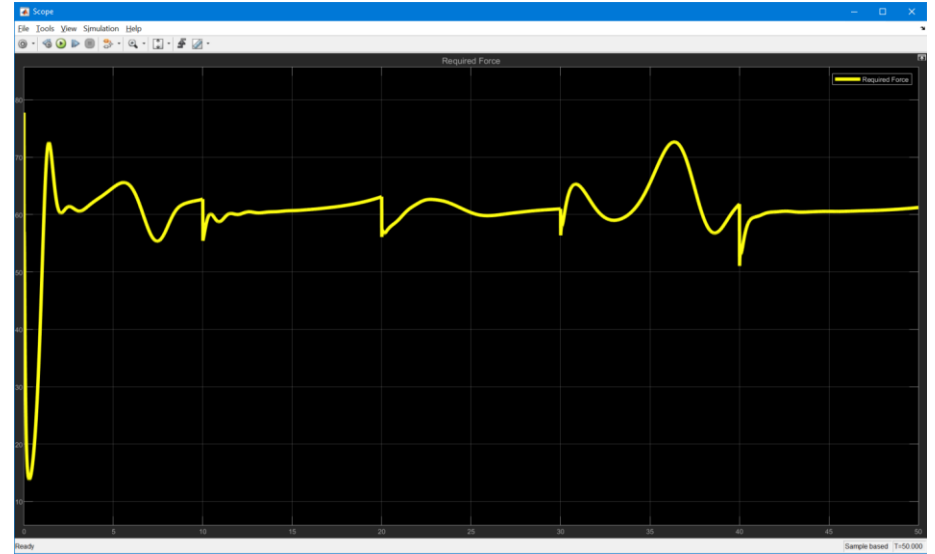
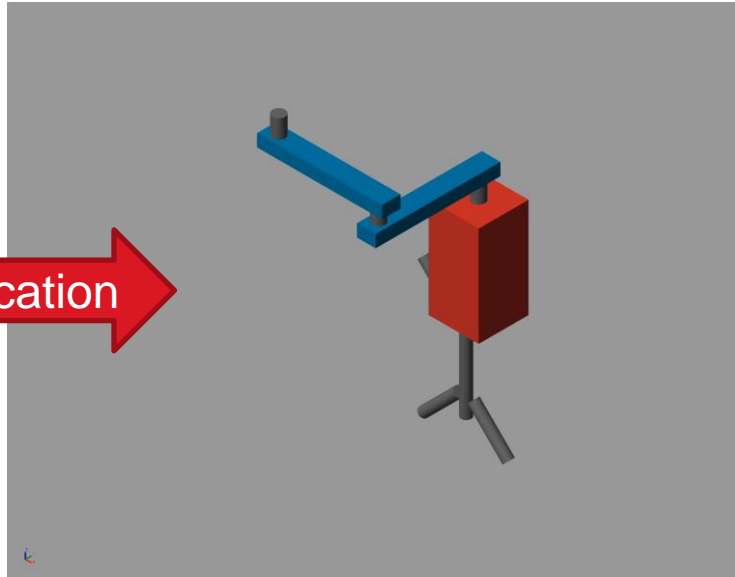
```



# PROJECT: SKYTRON FREEDOM

## Verification

DESIGN



3T

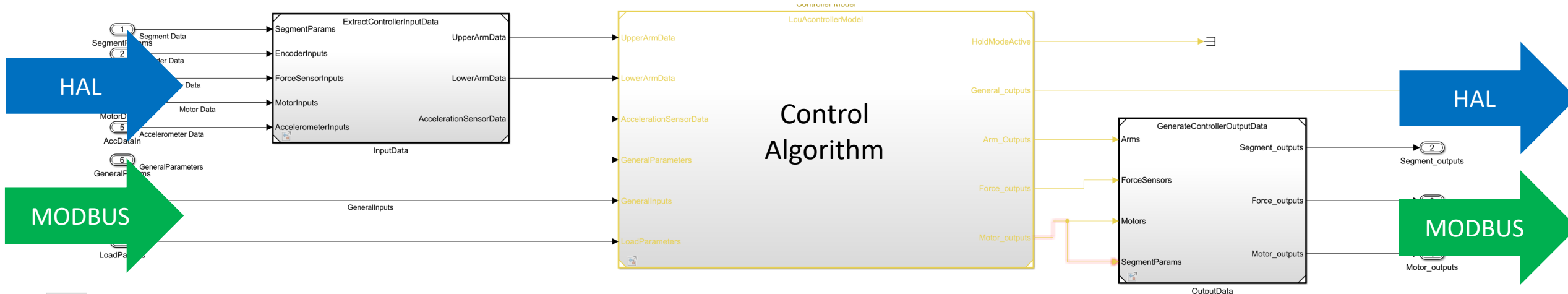
D&E  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congressentrum 's-Hertogenbosch

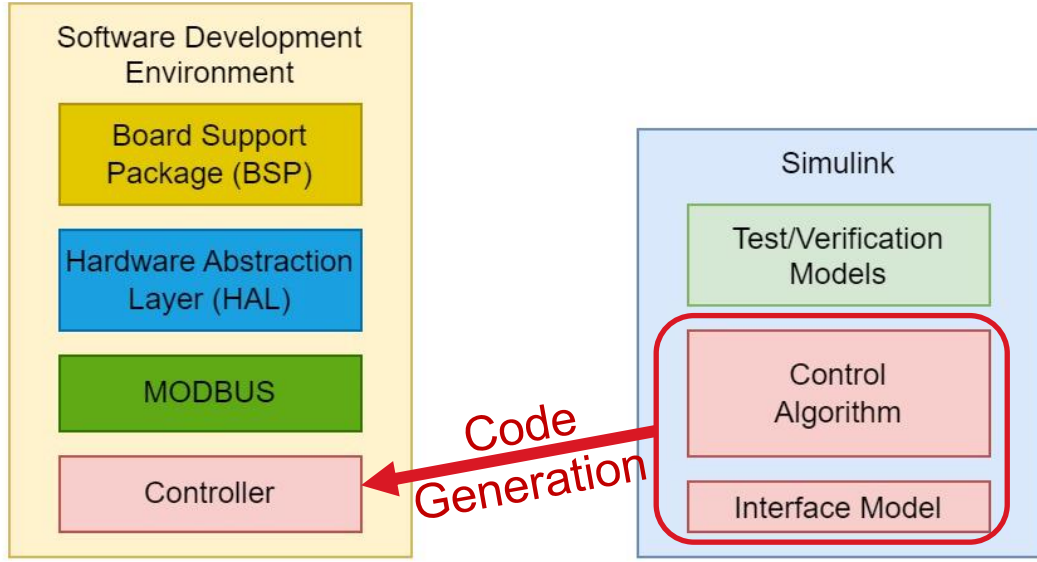


# PROJECT: SKYTRON FREEDOM

## Translation to model



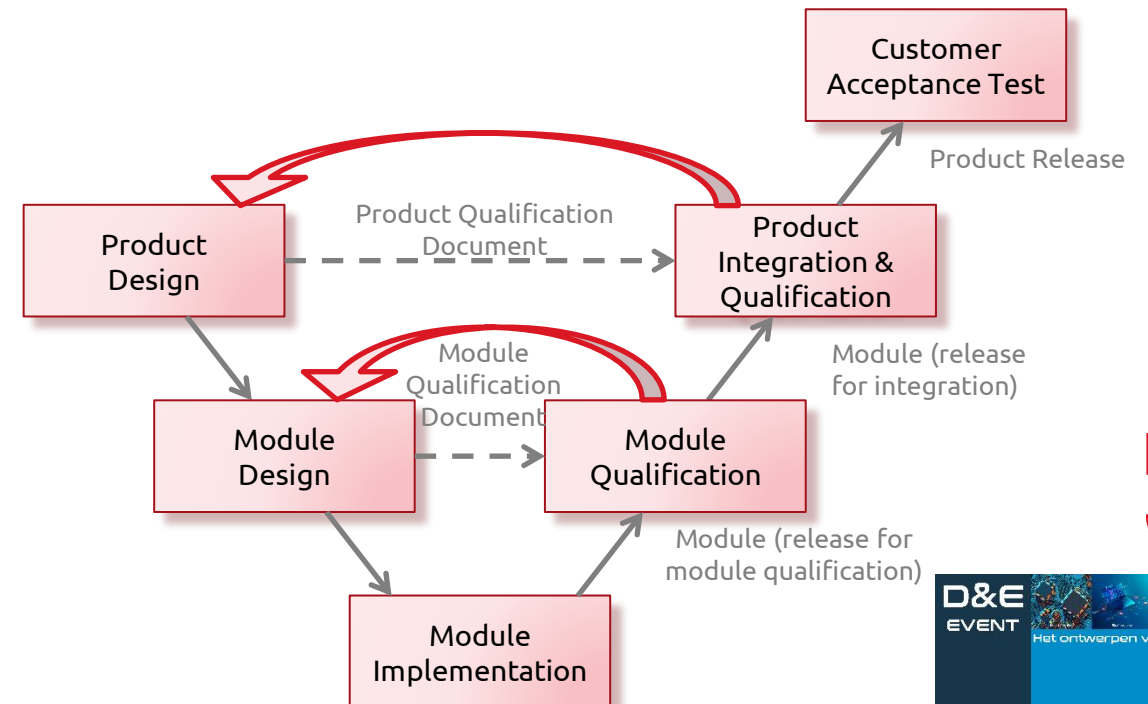
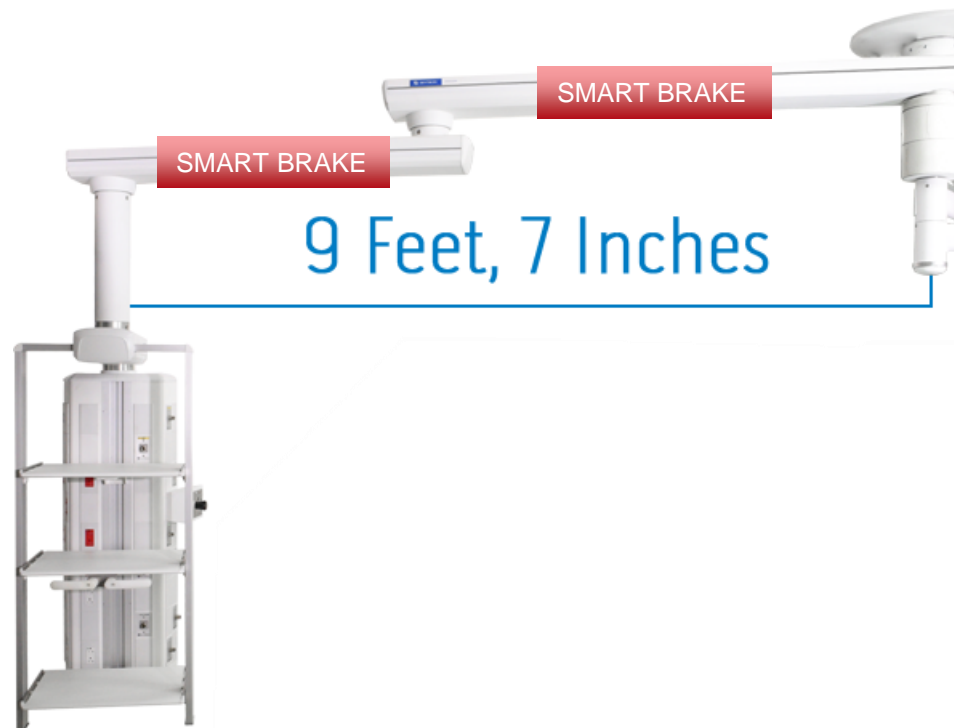
The screenshot shows a code editor window titled 'LcuAController'. The left sidebar displays a 'Content' pane with a tree view including 'Summary', 'Subsystem Report', 'Code Interface Report', 'Traceability Report', 'Static Code Metrics Report', 'Code Replacements Report', and 'Coder Assumptions'. Under the 'Code' section, files like 'ert\_main.cpp', 'LcuAController.h', and 'LcuAController\_types.h' are listed. The main editor area shows C++ code for the 'LcuAController.cpp' file, including comments and function definitions like 'void ModelClass::initialize()'. The code includes references to 'rtb\_ControllerModel' and 'rtb\_ControllerModel'.



# PROJECT: SKYTRON FREEDOM

## Integration

- Comparison between model and mechanical system
  - Optimize system and model
- Rapid design iterations to achieve the “*Excellent User Experience*”



# 3T

D&E  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch

# KEY TAKEAWAYS

## Project

## Customer Challenge

## Strategy

## Results

Co - development



- System is mechanically sub-optimal.
- Ambiguous requirements

- Use models to verify understanding of requirements.
- Use MDD as a framework, design methodology and a common language.

- Reduced development costs and time-to-market
- Autogenerated 17k+ lines of source code in 4 minutes.
- Shortened design iterations: day(s) instead of weeks.

# 3T

D&E  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congressentrum 's-Hertogenbosch

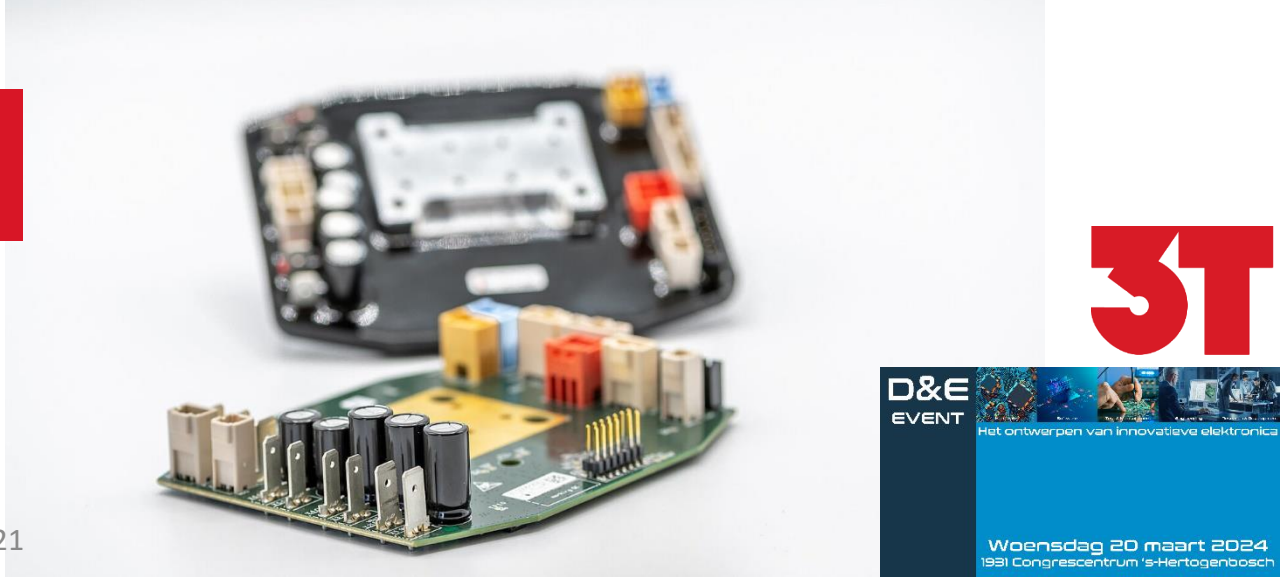




Racing Car by Solar Team Twente, picture © Solar Team Twente



# Project: Sensorless Motor Control



# 3T

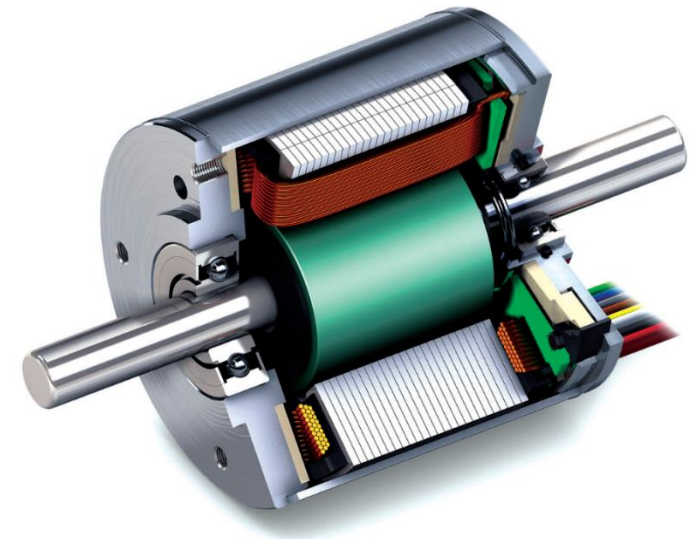
**D&E**  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congressentrum 's-Hertogenbosch

# PROJECT: SENSORLESS MOTOR CONTROL

## The challenge

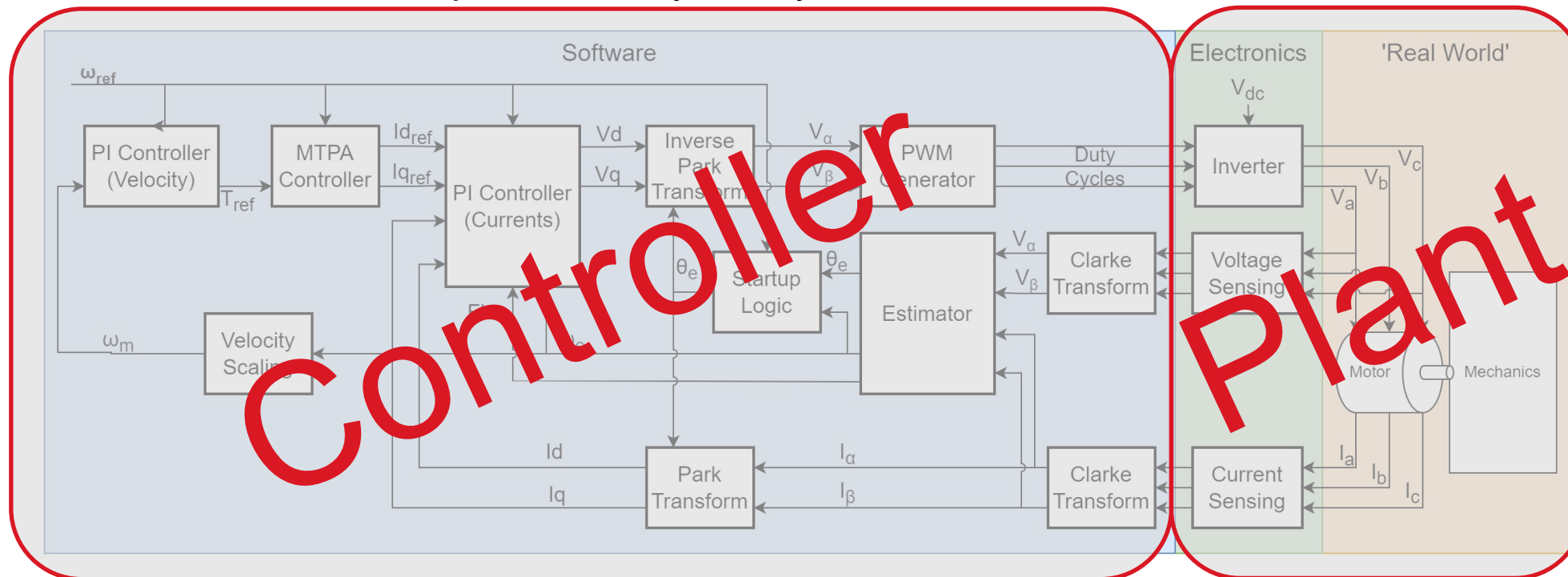
- Develop, evaluate and optimize high performance sensorless motor control solutions.
- Feasibility studies:
  - Electrical/Mechanical system
  - Execution timing (PIL)
- Make control algorithm target independent.



# PROJECT: SENSORLESS MOTOR CONTROL

## The Solution

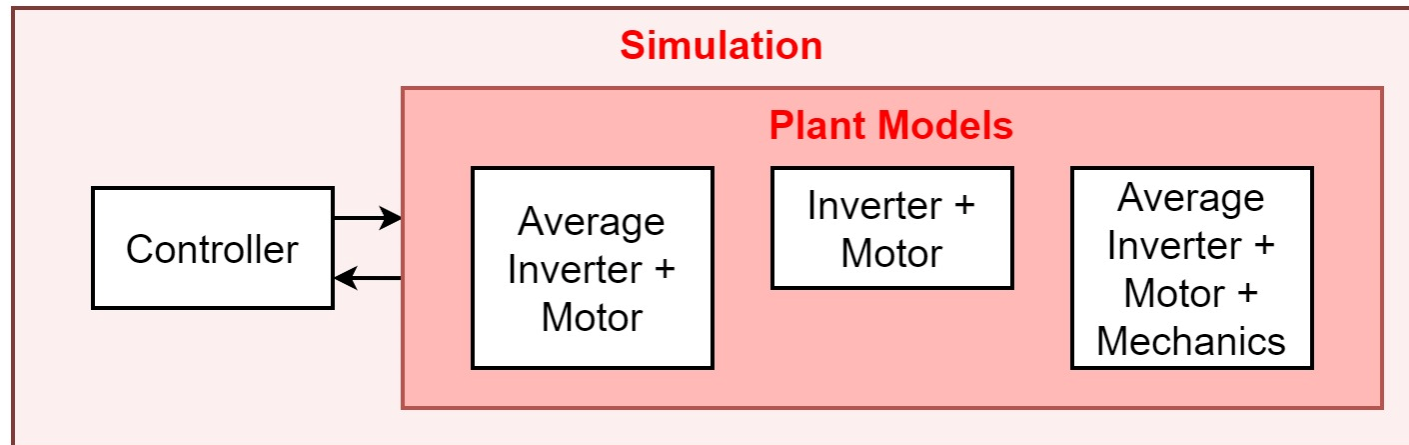
- Use Simulink to create multi domain models with clear interfaces.
  - Motor Control Blockset for Control Algorithm
  - Simscape Electrical for Motor Drive Electronics and Motor model
  - Simscape Multibody for System Mechanics



# PROJECT: SENSORLESS MOTOR CONTROL

## Development Strategy: Reduce development time and risk

- Design and validate the controller for different operating conditions.
  - Startup behavior
  - Inverter behavior
  - Mechanical behavior



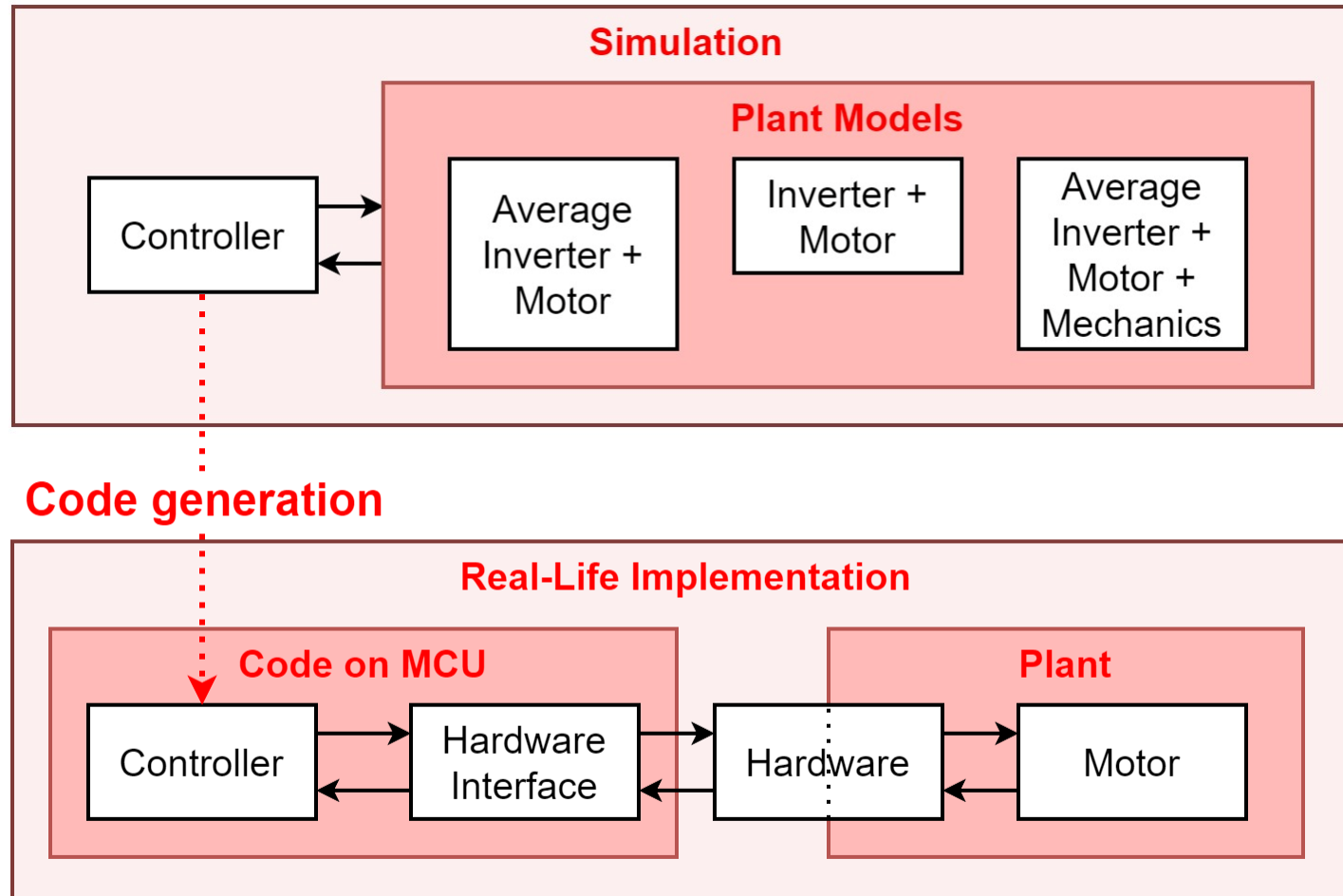
3T



# PROJECT: SENSORLESS MOTOR CONTROL

Development Strategy: Reduce development time and risk

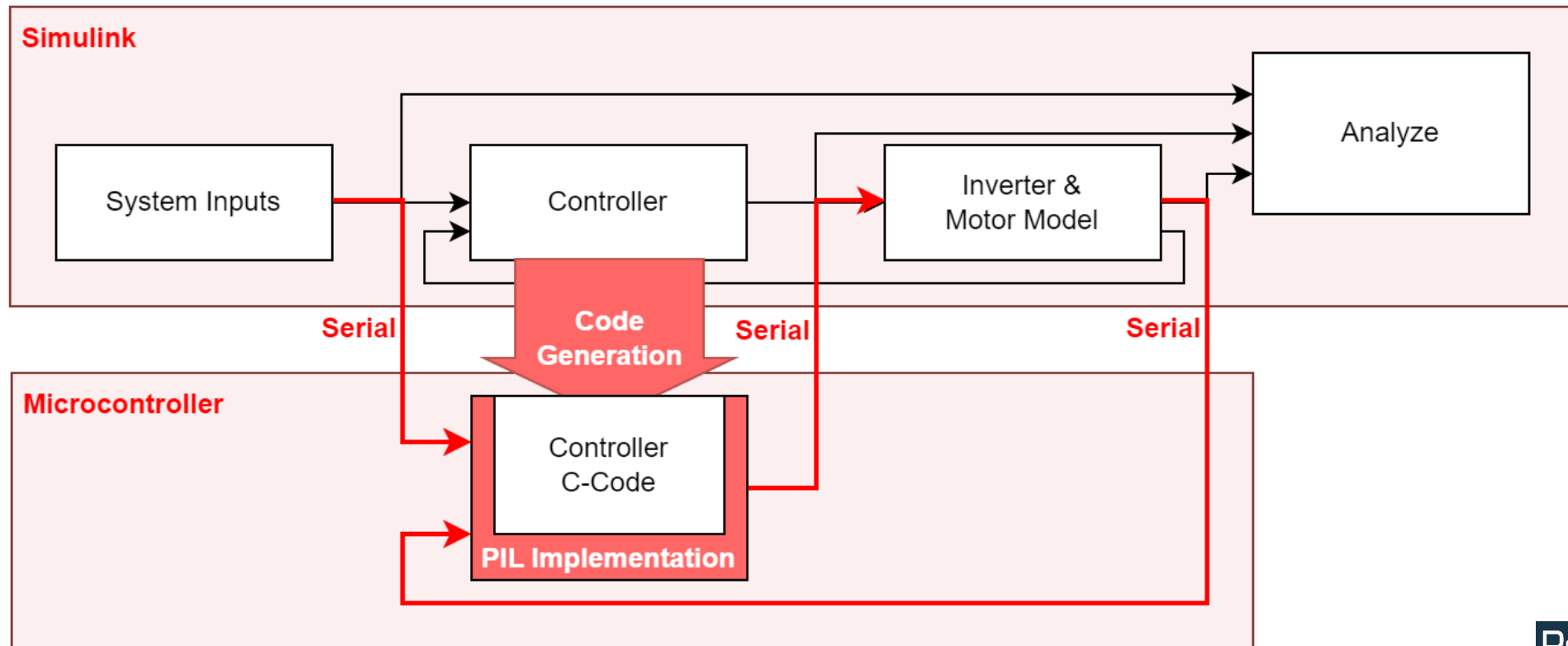
- Use Embedded Coder to generate code.



# PROJECT: SENSORLESS MOTOR CONTROL

## Example Feasibility study: Execution timing

- Processor in the Loop



3T

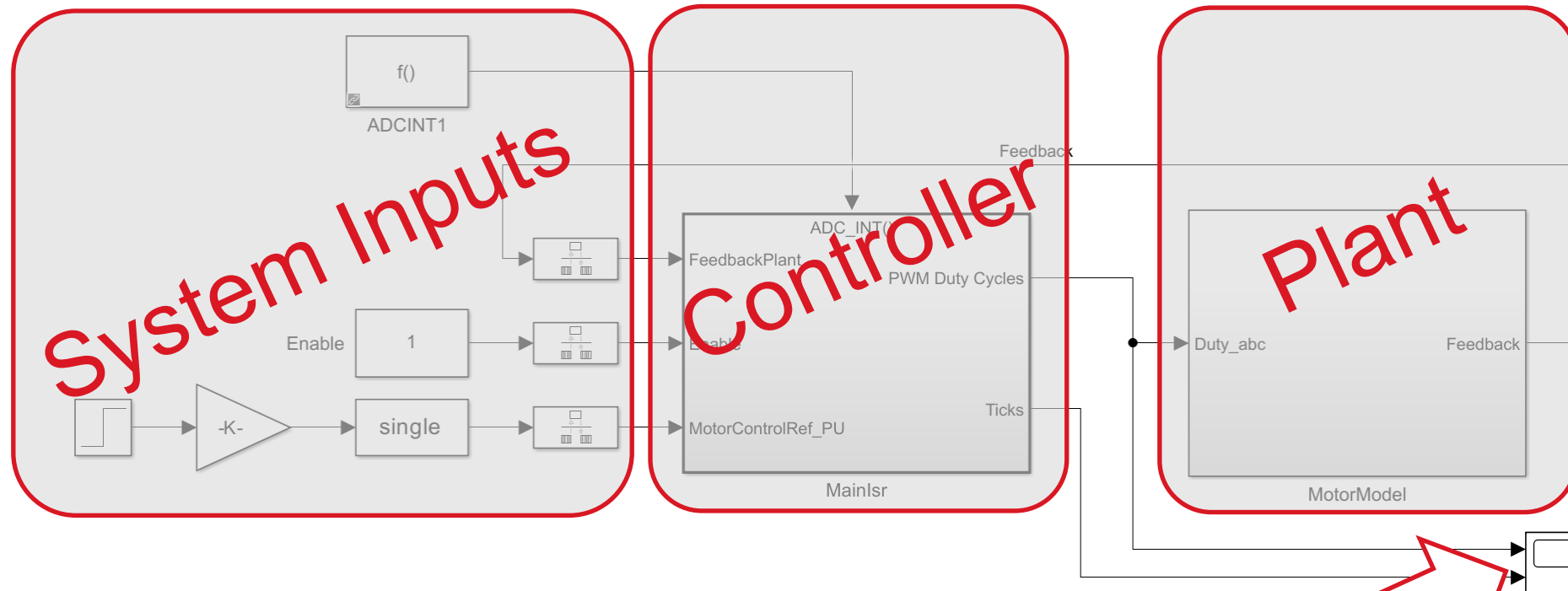
D&E  
EVENT  
Het ontwerpen van innovatieve elektronica

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch

# PROJECT: SENSORLESS MOTOR CONTROL

## Example Feasibility study: Execution timing

- Processor in the Loop

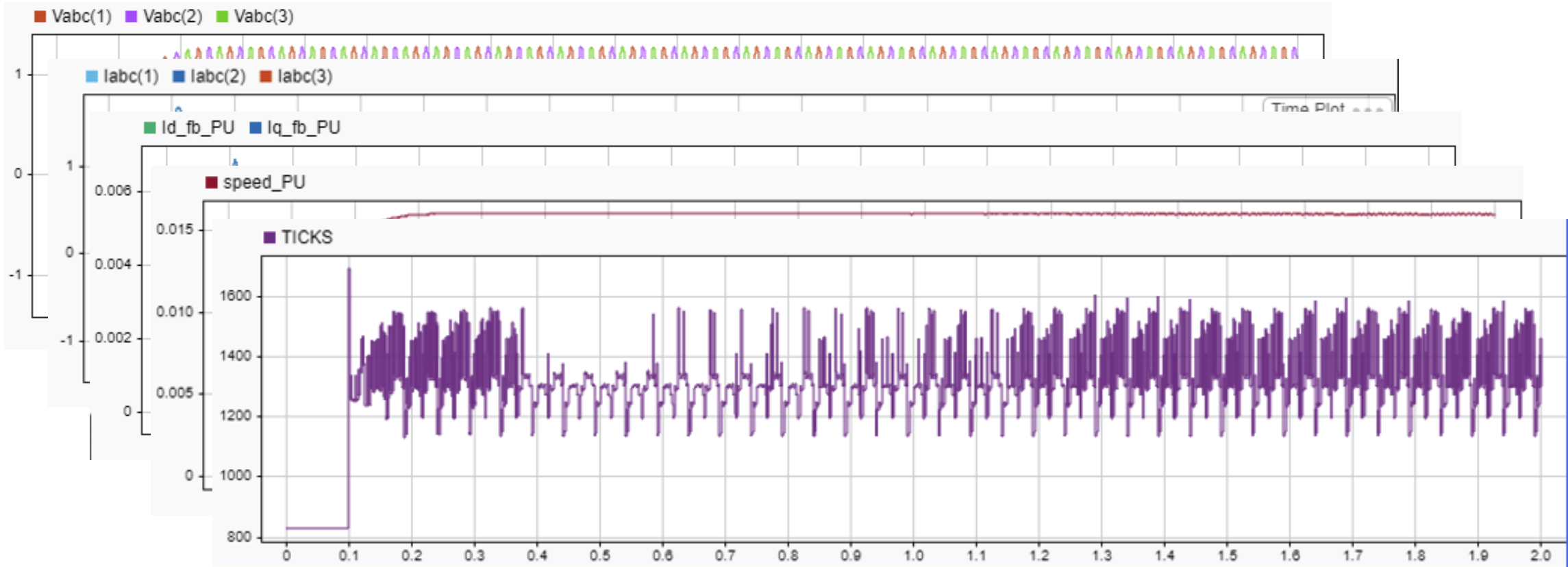


Scope for Analysis

3T

# PROJECT: SENSORLESS MOTOR CONTROL

## Example Feasibility study: Execution timing

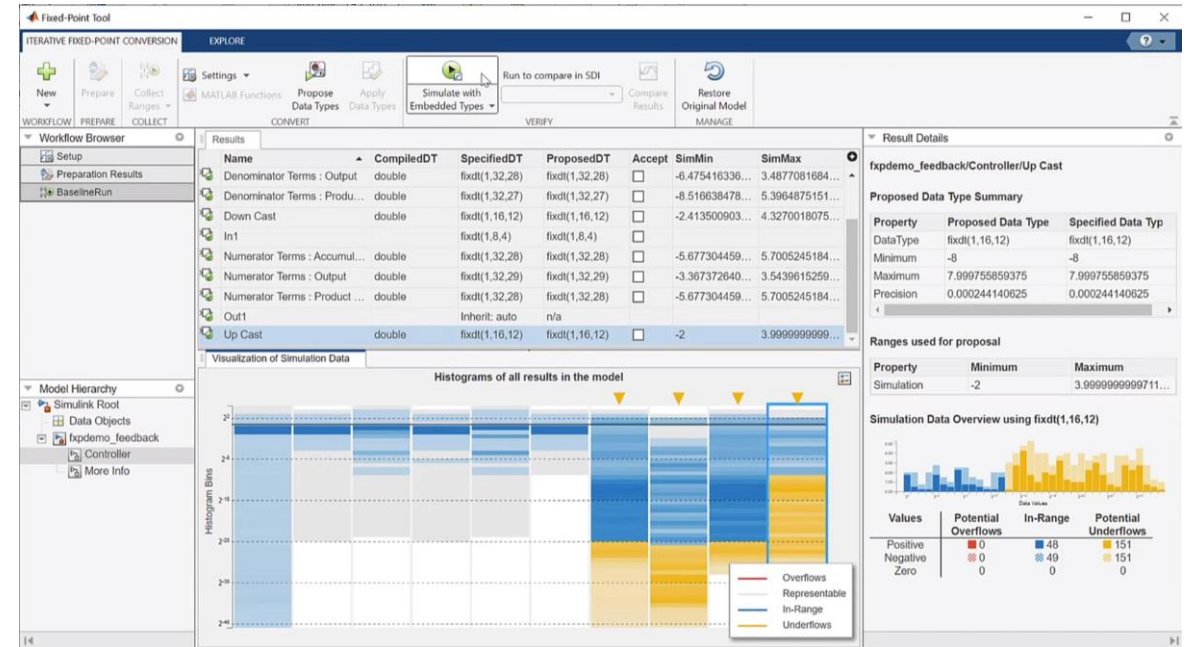




# PROJECT: SENSORLESS MOTOR CONTROL

## Future Plans

- Convert base datatype to fixed-point using Fixed-Point Designer Toolbox.
- Optimize computationally intensive math functions (Accuracy vs execution efficiency).



# KEY TAKEAWAYS

## Project

## Customer Challenge

## Strategy

## Results

Co - development

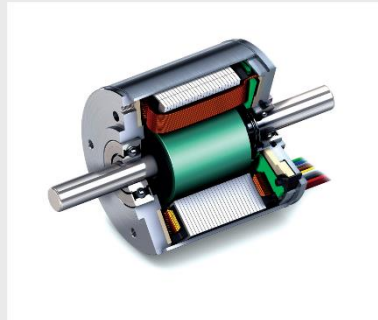


- System is mechanically sub-optimal.
- Ambiguous requirements

- Use models to verify understanding of requirements.
- Use MDD as a framework, design methodology and a common language.

- Reduced development costs and time-to-market.
- Autogenerated 17k+ lines of source code in 4 minutes.
- Shortened design iterations: day(s) instead of weeks.

Innovation & support



- Development of high-performance control algorithms is costly as it is target-dependent.
- Feasibility studies of mechanical systems.

- Use Simulink to create multi domain models with clear interfaces.
- Use Embedded Coder to generate target independent code.

- Met code execution time requirement of  $<30 \mu\text{s}$ .
- Answered feasibility questions in weeks instead of months.
- Achieved better software quality / decreased errors.

# 3T

**CHALLENGING TASKS ARE OUR SPECIALTY.  
3T IS WHERE TECHNOLOGY AND QUALITY MEET.**

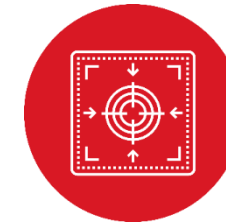


*Let's  
Connect!*

**Marcel van Wilpe**  
Business Development  
+31 53 851 66 26  
marcel.vanwilpe@kendrion.com



**High System  
Efficiency**



**High-precision  
applications**

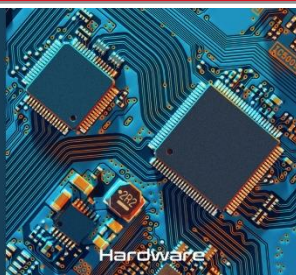


**Working in Harsh  
Environments**

*More  
about 3T*



**D&E  
EVENT**



**Het ontwerpen van  
innovatieve elektronica**

Woensdag 20 maart 2024  
1931 Congrescentrum 's-Hertogenbosch