Requirement Modeling

of Distributed Automotive Control Systems

Presented by:

Nate Rolfes

Ford Motor Company

Material prepared for the Mathworks Automotive Conference, May 12, 2016


Use Caution. Back Up Slowly
00:30.81

Pro Trailer Backup Assist

Distributed Control System Overview

Powertrain System

- Throttle Speed Controller for Speed Limiting
- Gear Shift Lever for State Logic

Steering System

- Steering Torque Sensing
- Steering Controller Actuation and Logic

Brake System

- Wheel Speed Sensors for Odometry
- Accelerometers for Vehicle Dynamics State Estimation
- Braking Controller Actuation for Speed Limiting

Pro Trailer Backup Assist

Distributed Control System Overview

Camera System

- Rearview Camera for Trailer Angle Detection

Pro Trailer Backup Assist

Distributed Control System Overview

Camera System

- Rearview Camera for Trailer Angle Detection
- Lighting for Night Usage

Pro Trailer Backup Assist

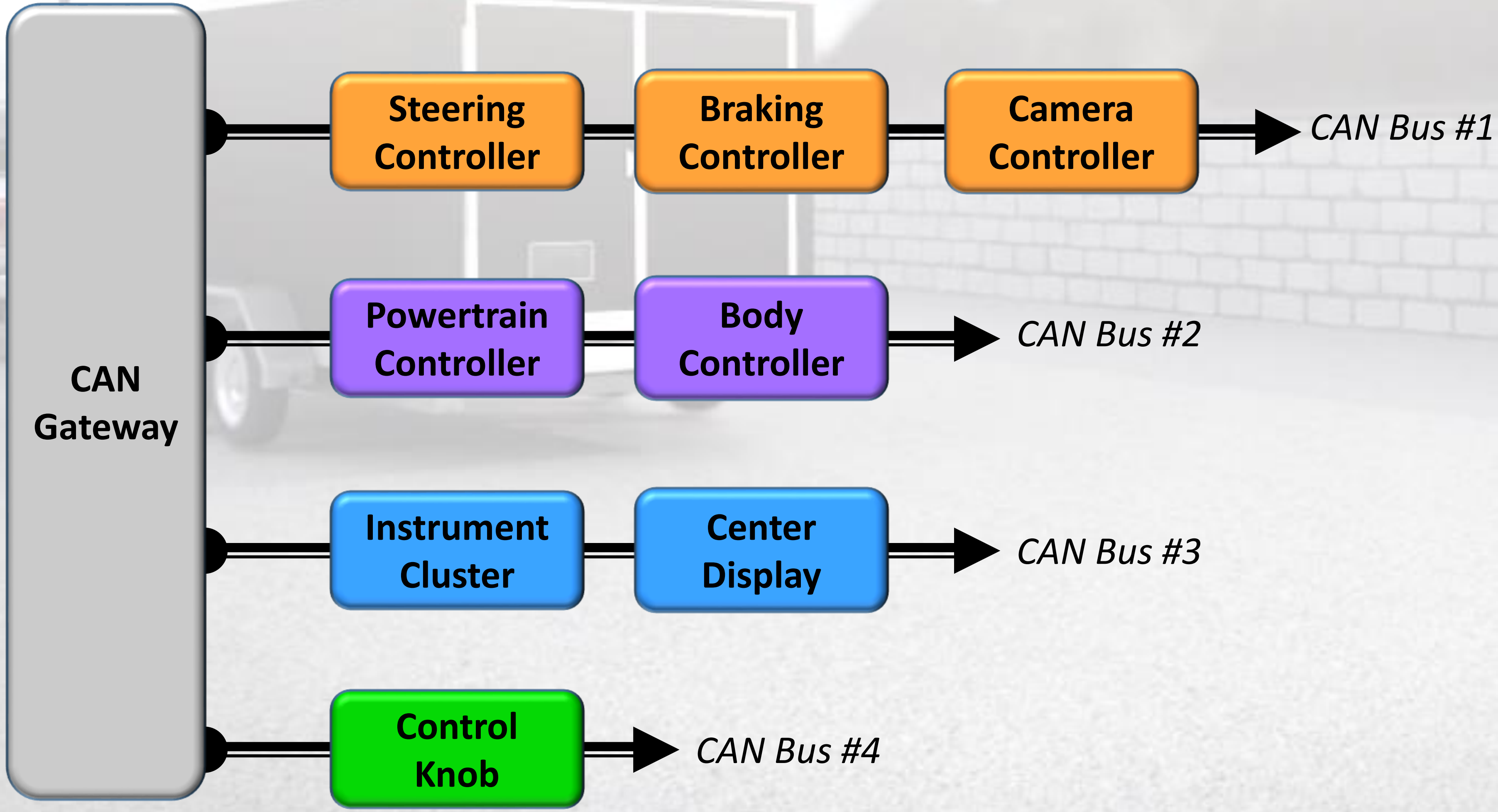
Distributed Control System Overview

HMI System: Driver Inputs

- Activation Switch and Control Knob
- Five-Way Buttons for Driver Inputs
- Cluster Display for Menu Selection and Instructions
- Center Console Display for Trailer View & Warnings

Challenges of Pro Trailer Backup Assist

- There is not a standalone "Pro Trailer Backup Assist" Module
- The feature is a Distributed Logic Control System containing Eight ECU's on four CAN buses connected through a CAN Gateway



Challenges of Pro Trailer Backup Assist

- The control logic is designed based on engineering considerations, e.g.
  - Optimizing and sharing new functionality
  - Leveraging and adapting carryover functionality
  - Minimizing communication bandwidth
  - Meeting program timing

Steering Controller

Braking Controller

Camera Controller

Powertrain Controller

Body Controller

Instrument Cluster

Center Display

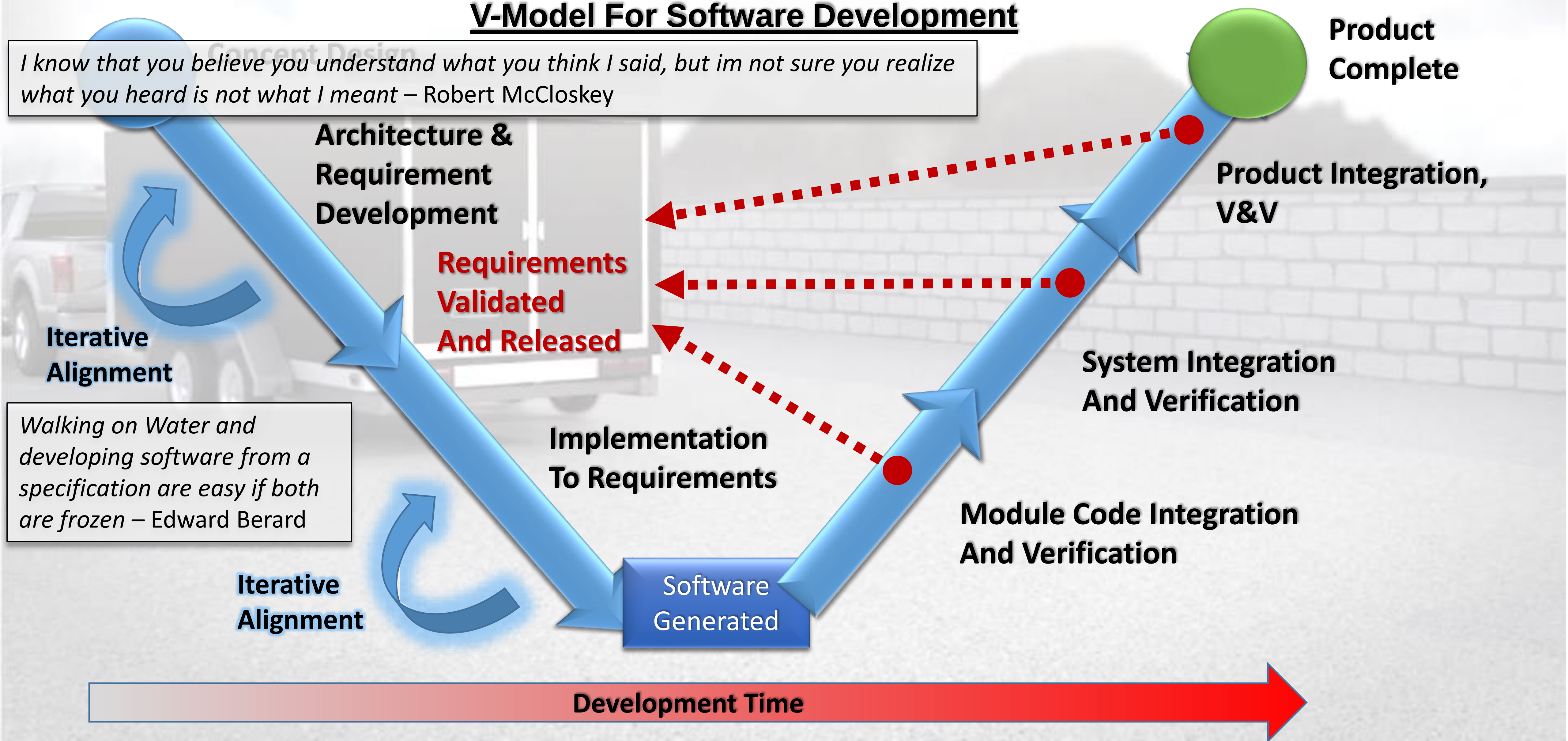
Control Knob

- Each component of the system must meet its intended **logical requirements** to prevent logic looping, locking, or exiting during operation.
- Requirements and priorities must be combined and integrated with existing functional requirements to ensure compatibility with other functional systems and interfaces.

Challenges of Pro Trailer Backup Assist

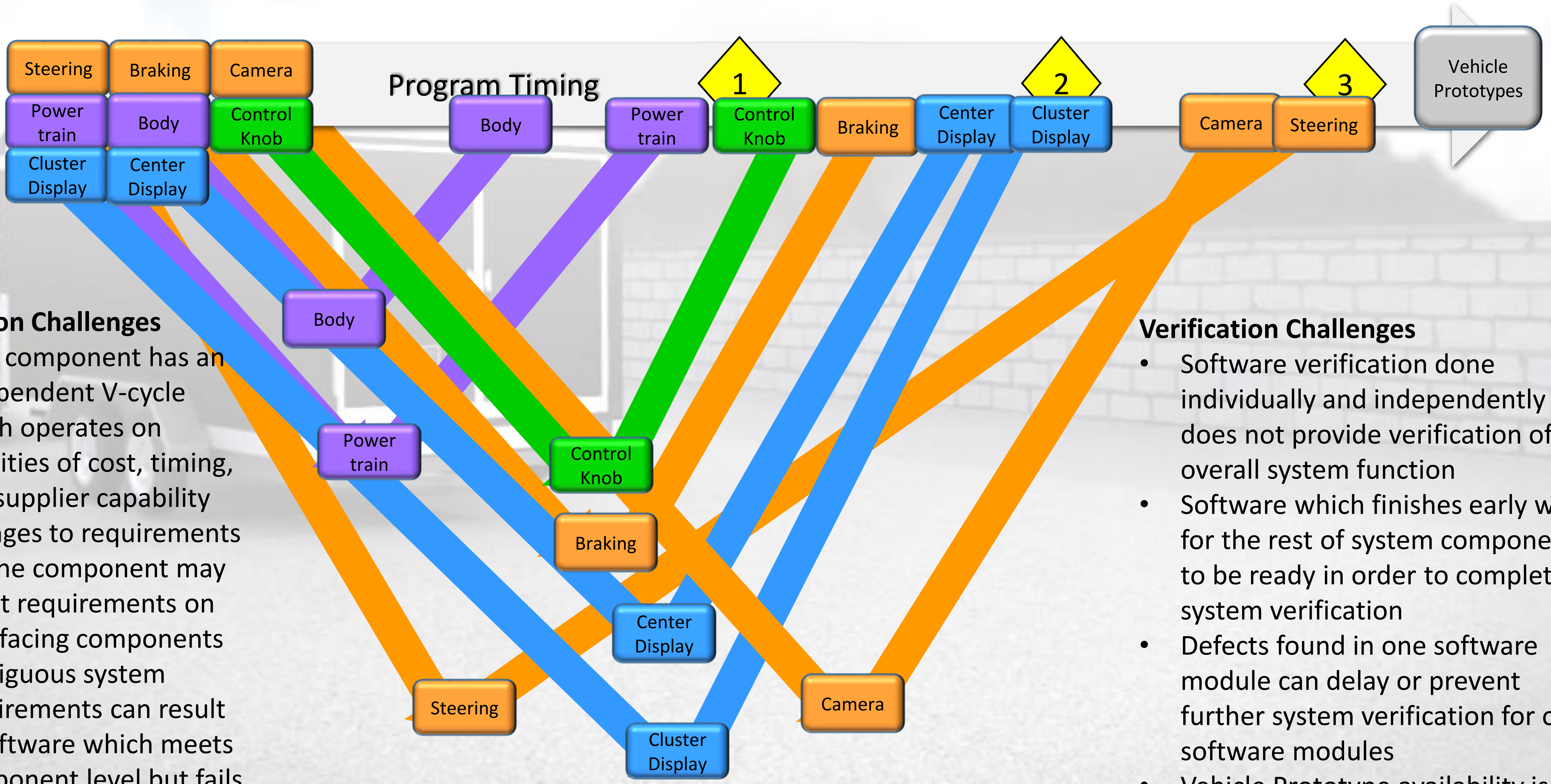
V-Model For Software Development

I know that you believe you understand what you think I said, but im not sure you realize what you heard is not what I meant – Robert McCloskey



Walking on Water and developing software from a specification are easy if both are frozen – Edward Berard

Challenges of Pro Trailer Backup Assist



Validation Challenges

- Each component has an independent V-cycle which operates on priorities of cost, timing, and supplier capability
- Changes to requirements on one component may affect requirements on interfacing components
- Ambiguous system requirements can result in software which meets component level but fails the system level

Verification Challenges

- Software verification done individually and independently does not provide verification of the overall system function
- Software which finishes early waits for the rest of system components to be ready in order to complete system verification
- Defects found in one software module can delay or prevent further system verification for other software modules
- Vehicle Prototype availability is too late to resolve critical defects

Solutions of Pro Trailer Backup Assist

1. **REQUIREMENT MODELING**:
   - A modeling methodology for Requirements which captures and simulates the logical parts to ensure the distributed control logical design of requirements works as intended prior to release for software implementation
2. **DISTRIBUTED NETWORK SIMULATION**:
   - A simulation environment which can link multiple Controller modules, CAN Networks, Driver and Vehicle Interactions.
   - It can simulate both MIL (Virtual) and HIL (Hardware) in real-time and each controller can be switched in real-time to either the MIL or HIL version. It can test all systems together or target systems individually at the system engineer's discretion
3. **VALIDATION AND VERIFICATION TOOL:**
   - A tool that can work effectively throughout the Software V process to:
     - ✓Test and validate requirement models (Down the System V)
     - ✓Verify that software components and module outputs match the requirement model behavior (Up the System V)

What is Requirement Modeling?

TIME

Remembering the Apollo 11 Moon Landing With the Woman Who Made It Happen

Lily Rothman @lilyrothman | July 20, 2015

"...Part of what had made Hamilton's work so effective was that she tested everything so rigorously,
in a simulator that could demonstrate the "system of systems" at work,
and the relationship between the software, the hardware and the astronaut. "We couldn't run something up to the moon," she says. But they could run lots of tests on the ground.
Hamilton's team found that nearly three-quarters of them were interface errors, like conflicts in timing or priority..."

Margaret Hamilton

http://time.com/3948364/moon-landing-apollo-11-margaret-hamilton/

What is Requirement Modeling?

2013-01-2237

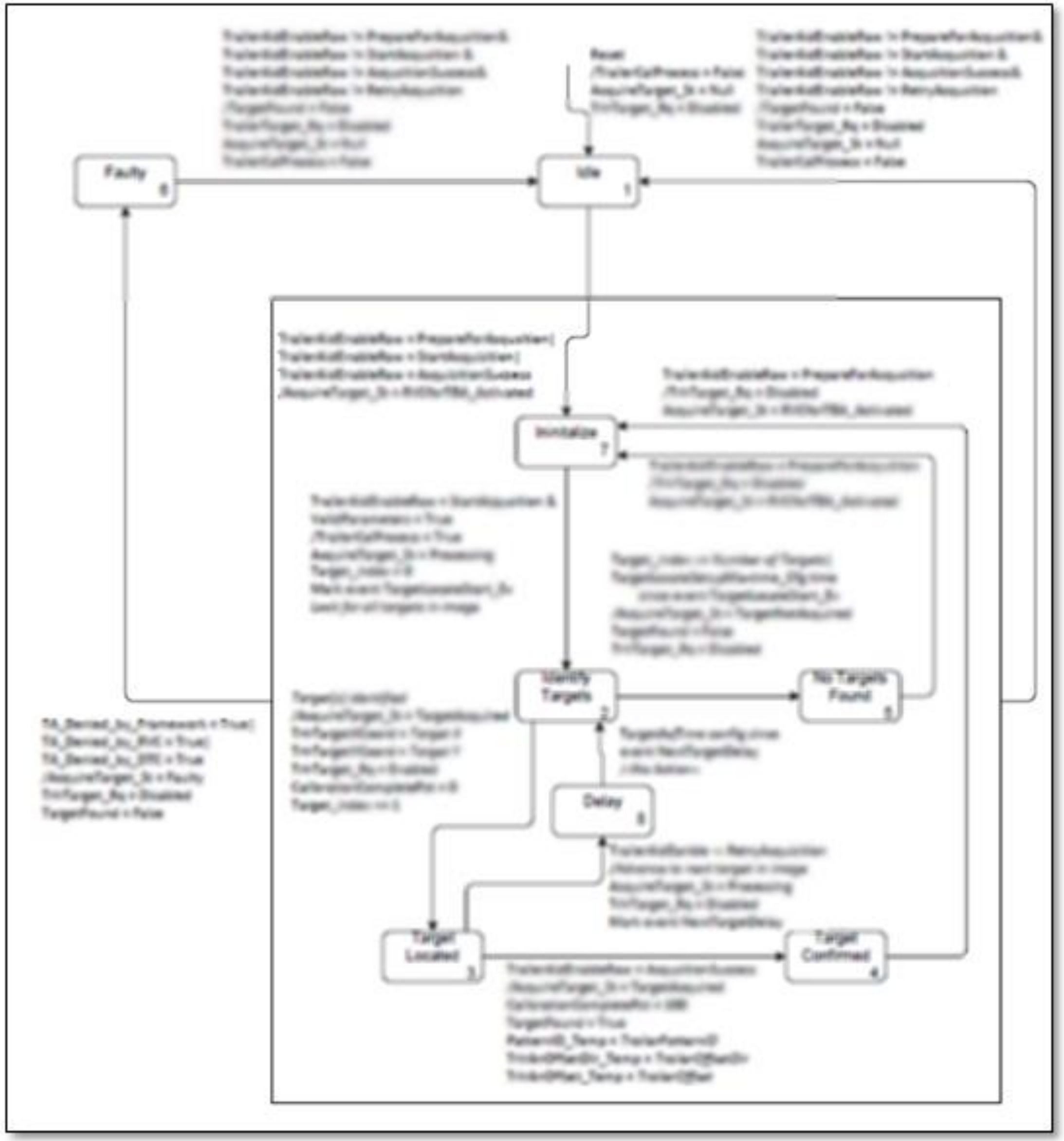
Requirements Modeling and Automated Requirements-Based Test Generation

John Lee and Jon Friedman
MathWorks

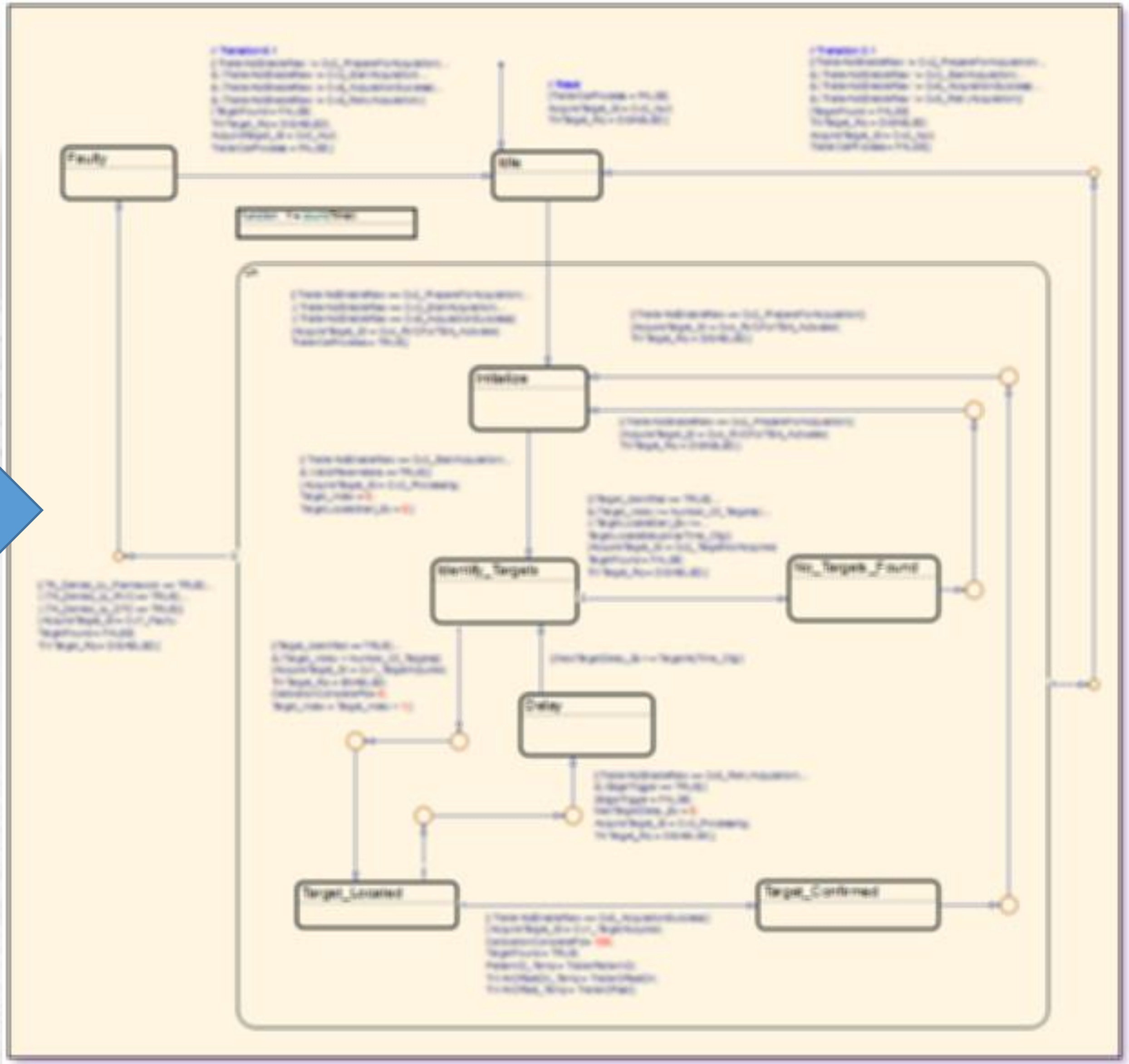
"...The goal of requirements models is **to capture the functional requirement in a clear, concise, analyzable and executable manner**, which is **typically not possible with natural language.**
The requirements models can then be used to **evaluate the interaction and compatibility of requirements from disparate sources**
as well as to develop tests and acceptance criteria (or expected outputs).
The use of the requirements models for test creation enables engineers to assess the completeness of the tests **using different notions of coverage on the requirements model**..."

Requirement Modeling Example

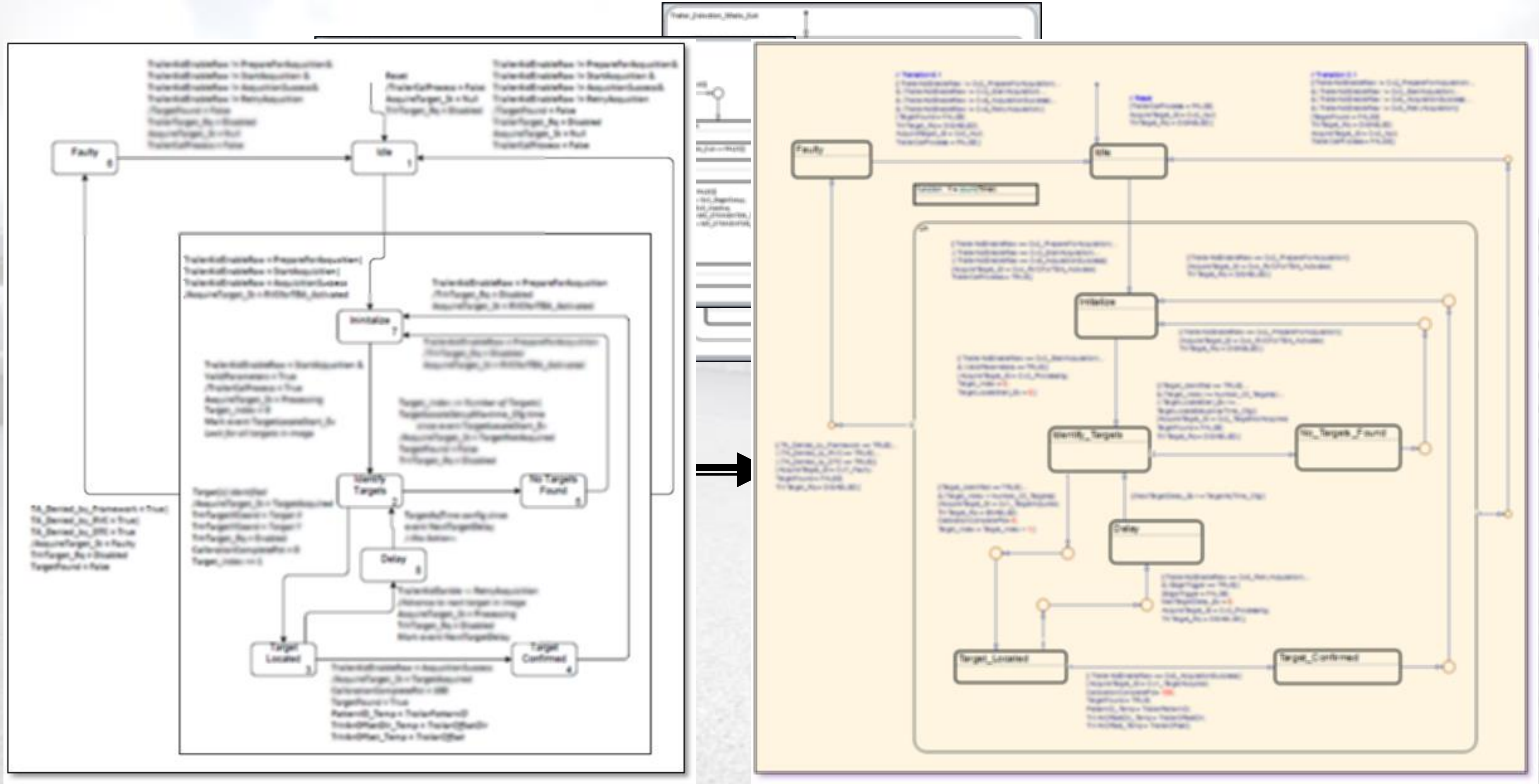
Paper Specification State Machine:



Stateflow Requirement Model:



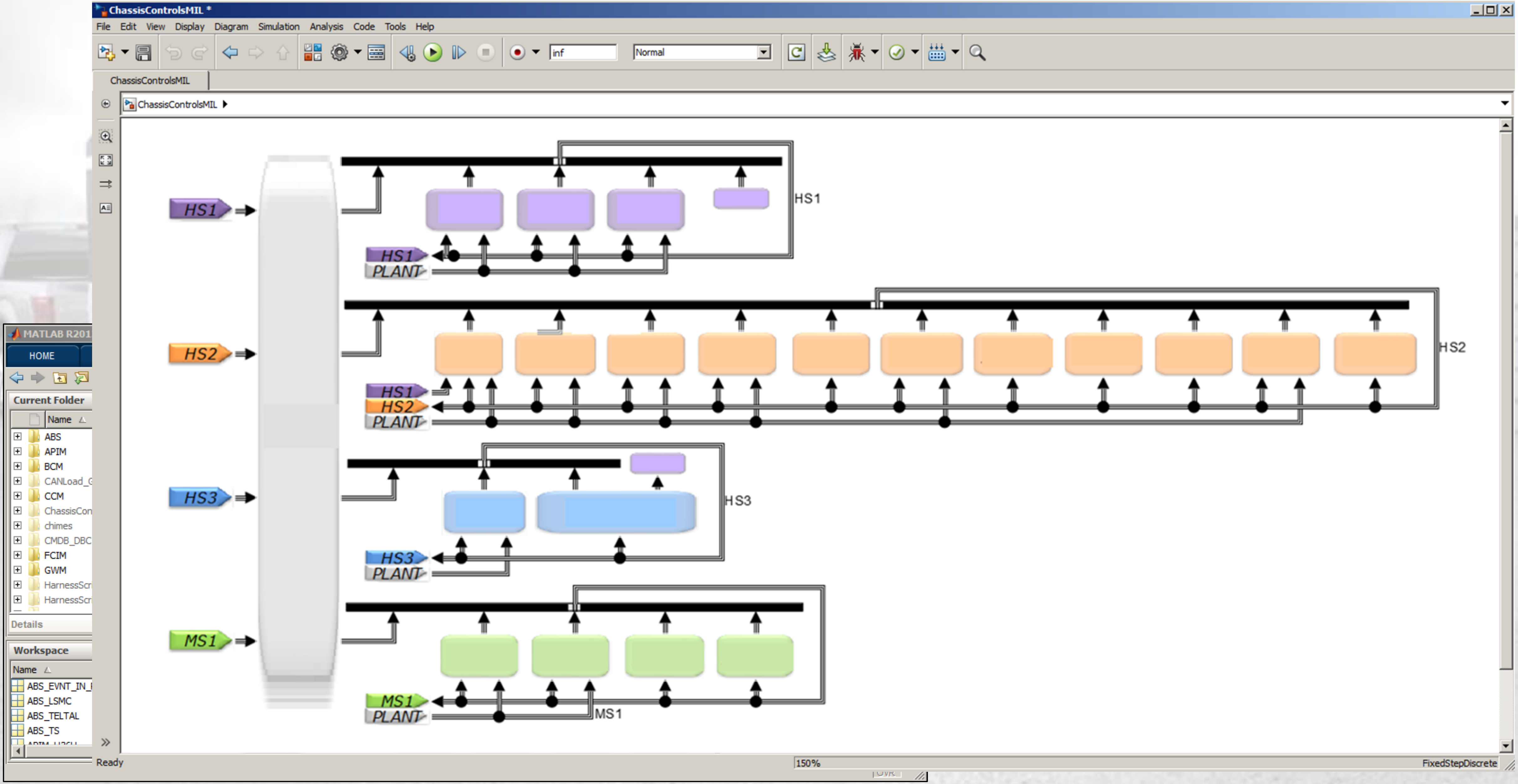
Requirement Modeling Example



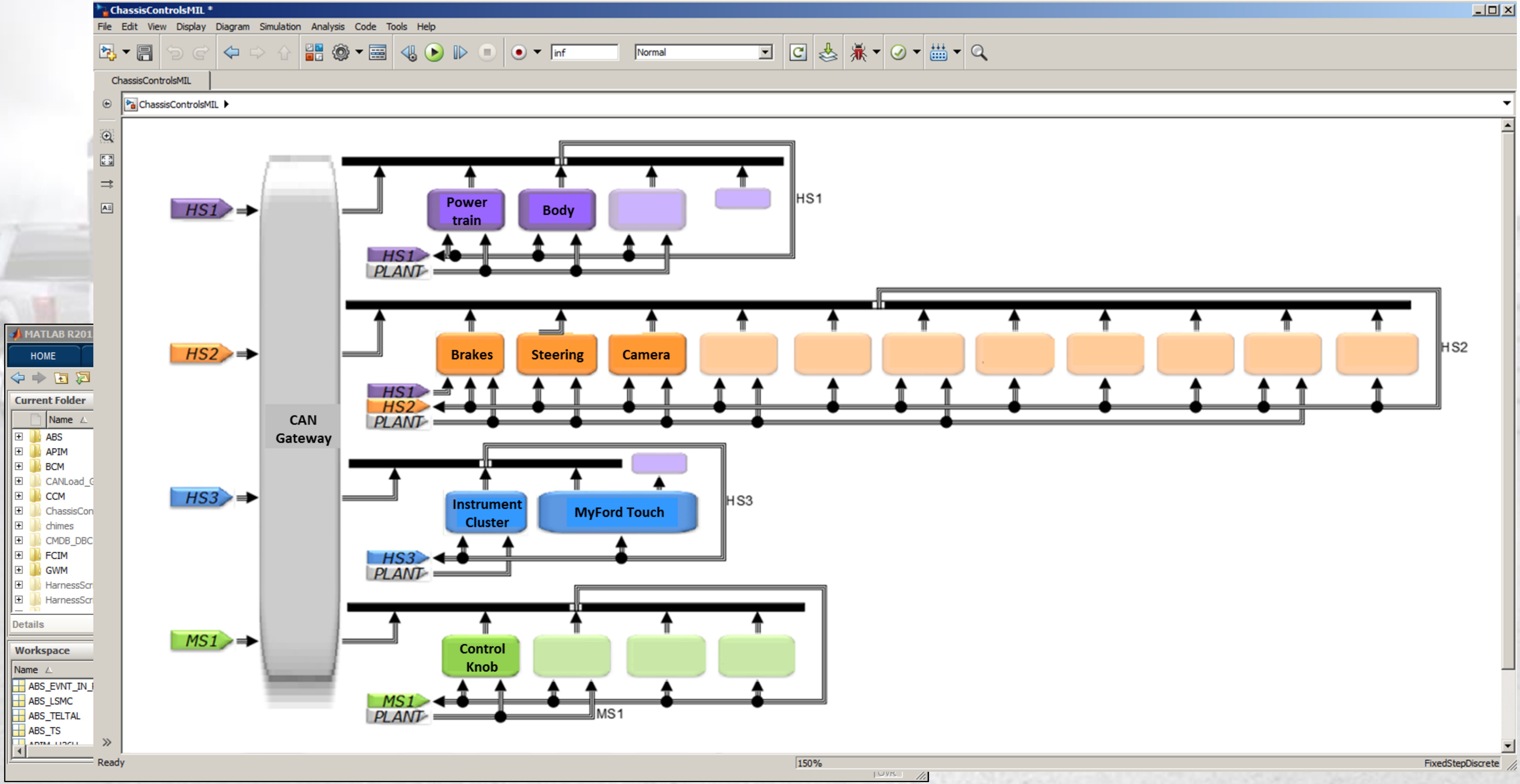
Solutions of Pro Trailer Backup Assist

1. **REQUIREMENT MODELING**:
   - A modeling methodology for Requirements which captures and simulates the logical parts to ensure the distributed control logical design of requirements works as intended prior to release for software implementation
2. **DISTRIBUTED NETWORK SIMULATION**:
   - A simulation environment which can link multiple Controller modules, CAN Networks, Driver and Vehicle Interactions.
   - It can simulate both MIL (Virtual) and HIL (Hardware) in real-time and each controller can be switched in real-time to either the MIL or HIL version. It can test all systems together or target systems individually at the system engineer's discretion
3. **VALIDATION AND VERIFICATION TOOL:**
   - A tool that can work effectively throughout the Software V process to:
     - ✓ Test and validate requirement models (Down the System V)
     - ✓ Verify that software components and module outputs match the requirement model behavior (Up the System V)

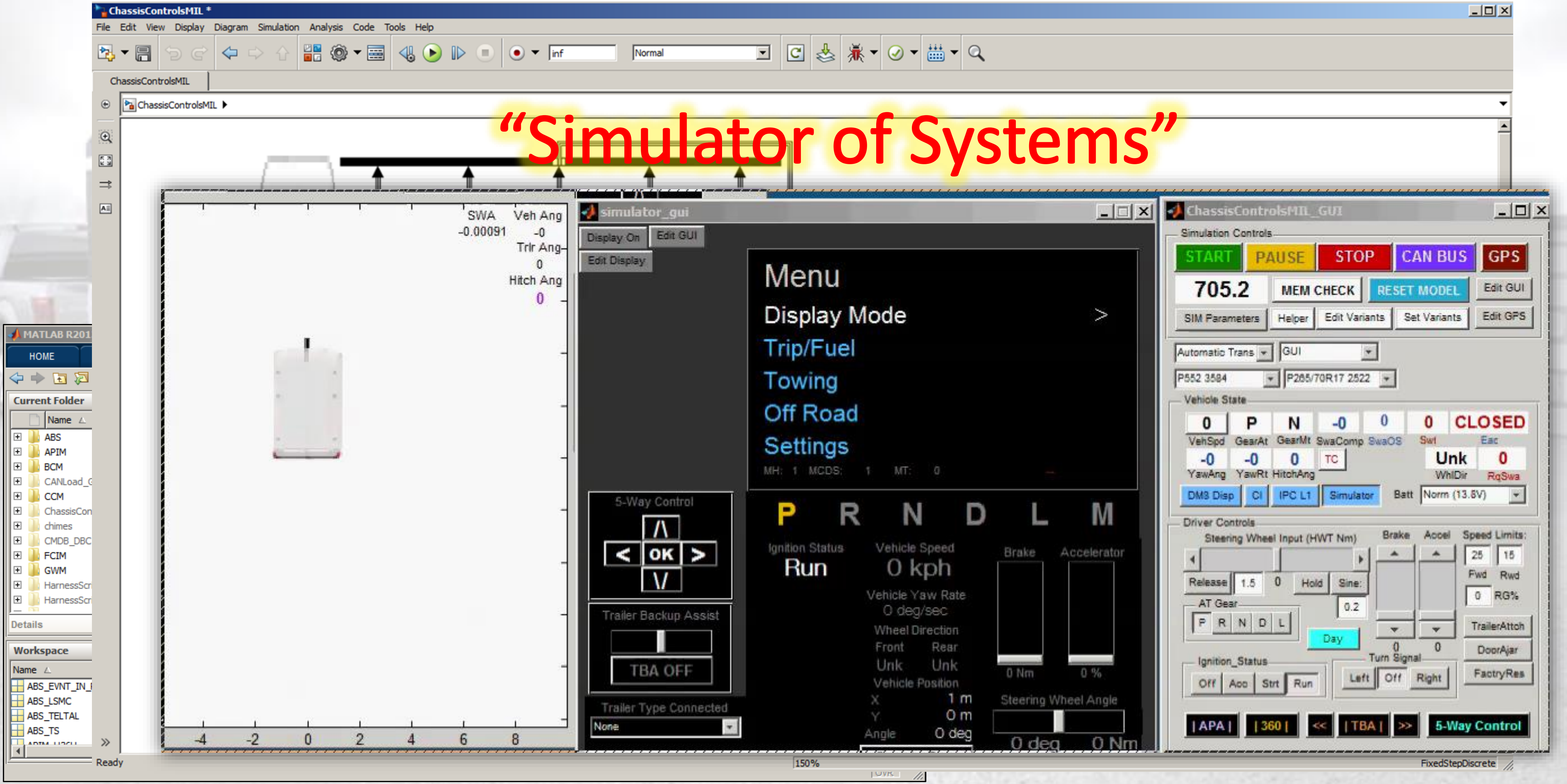
Distributed Network Simulation MIL



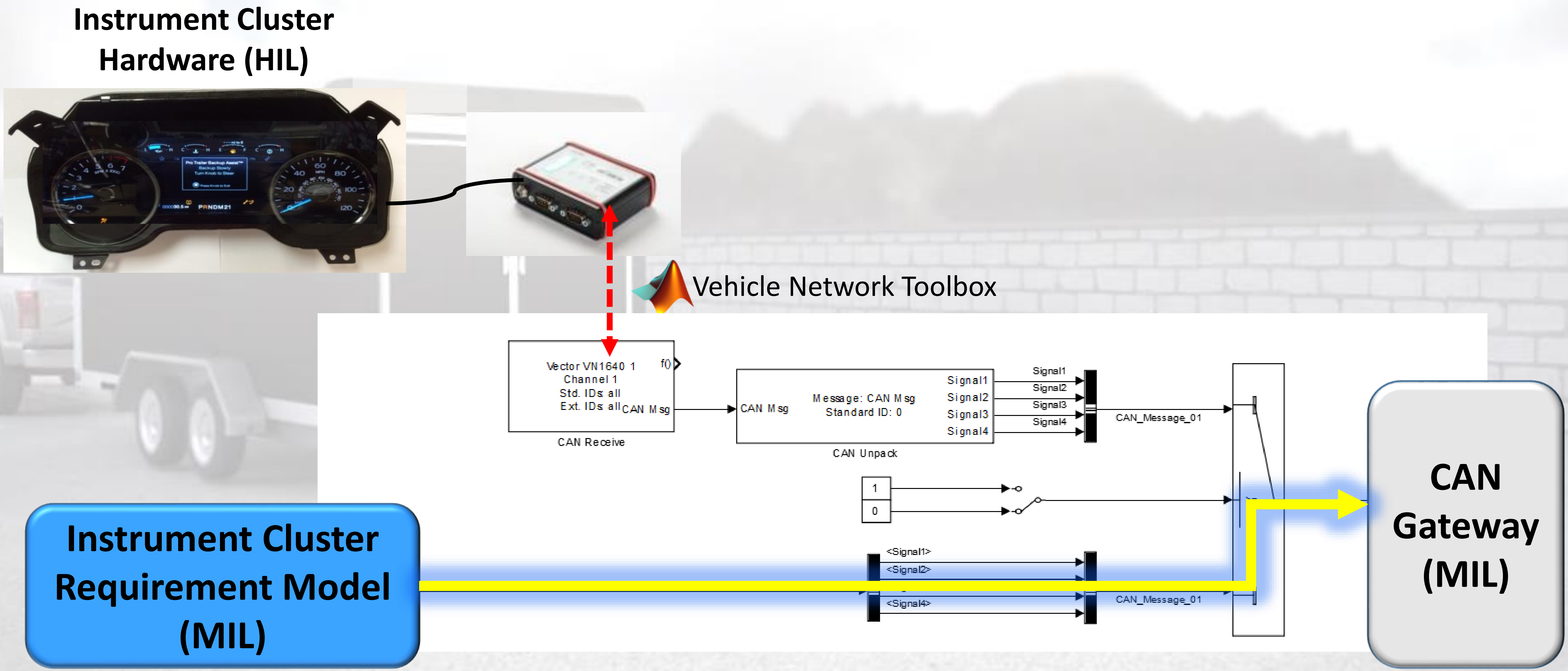
Distributed Network Simulation MIL



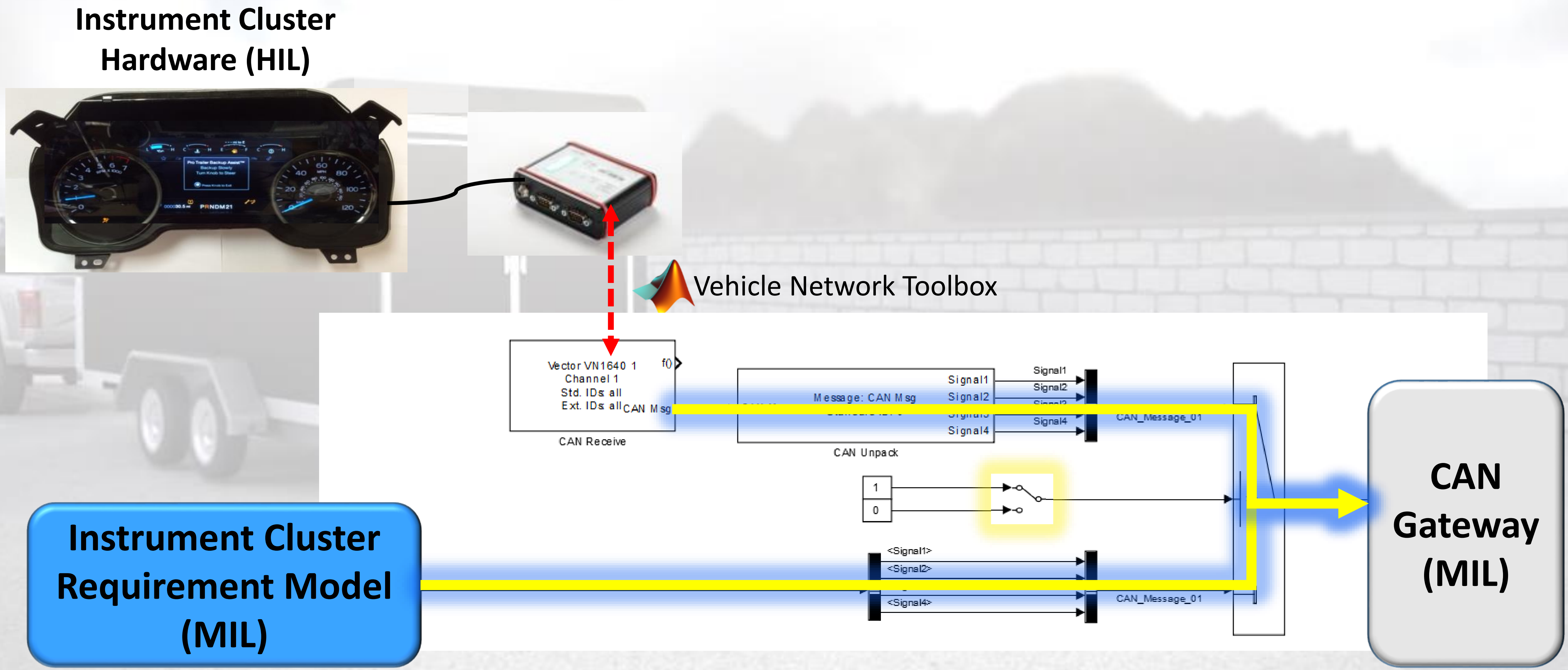
Distributed Network Simulation MIL



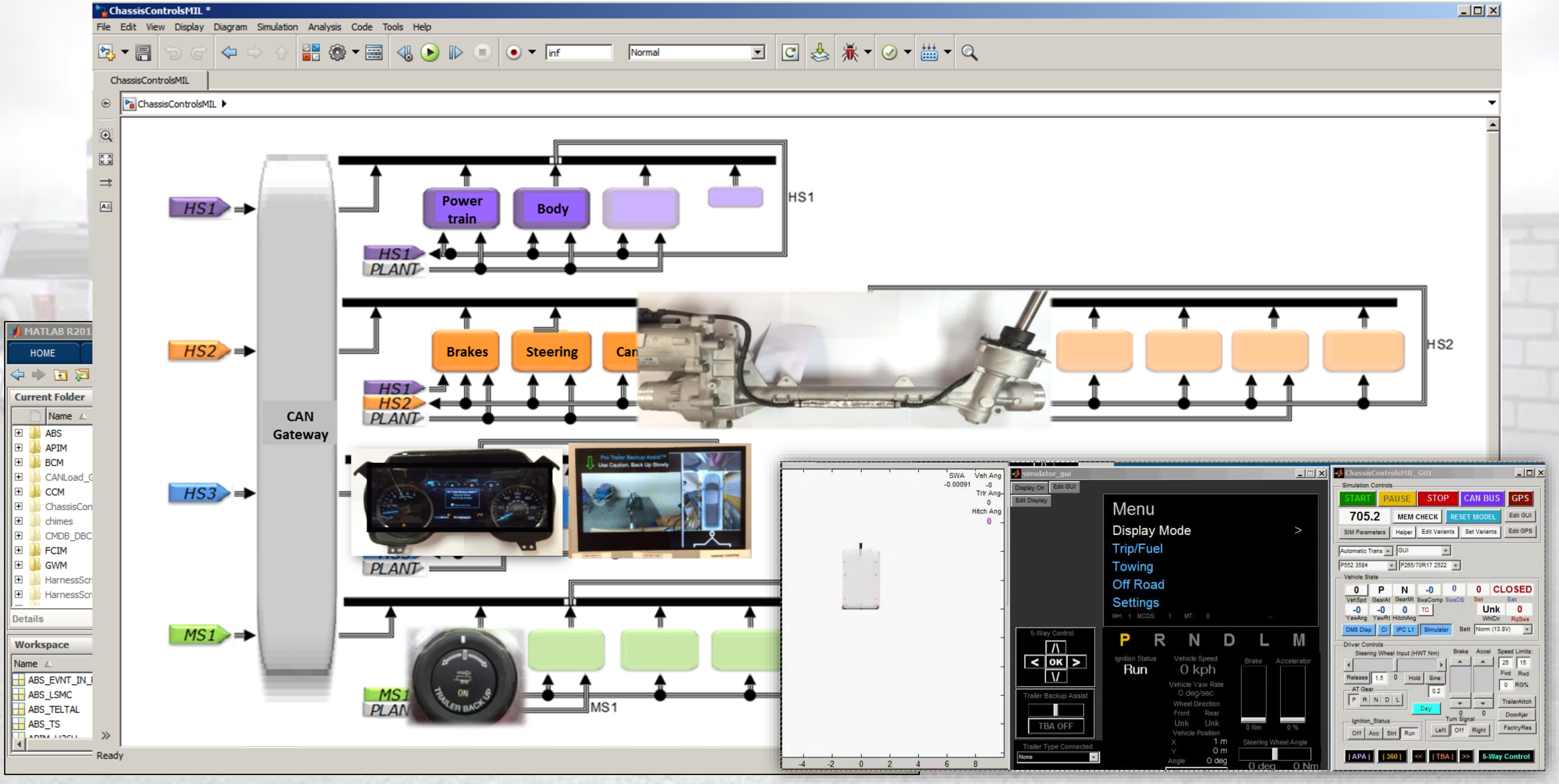
Adding Hardware using Vehicle Network Toolbox



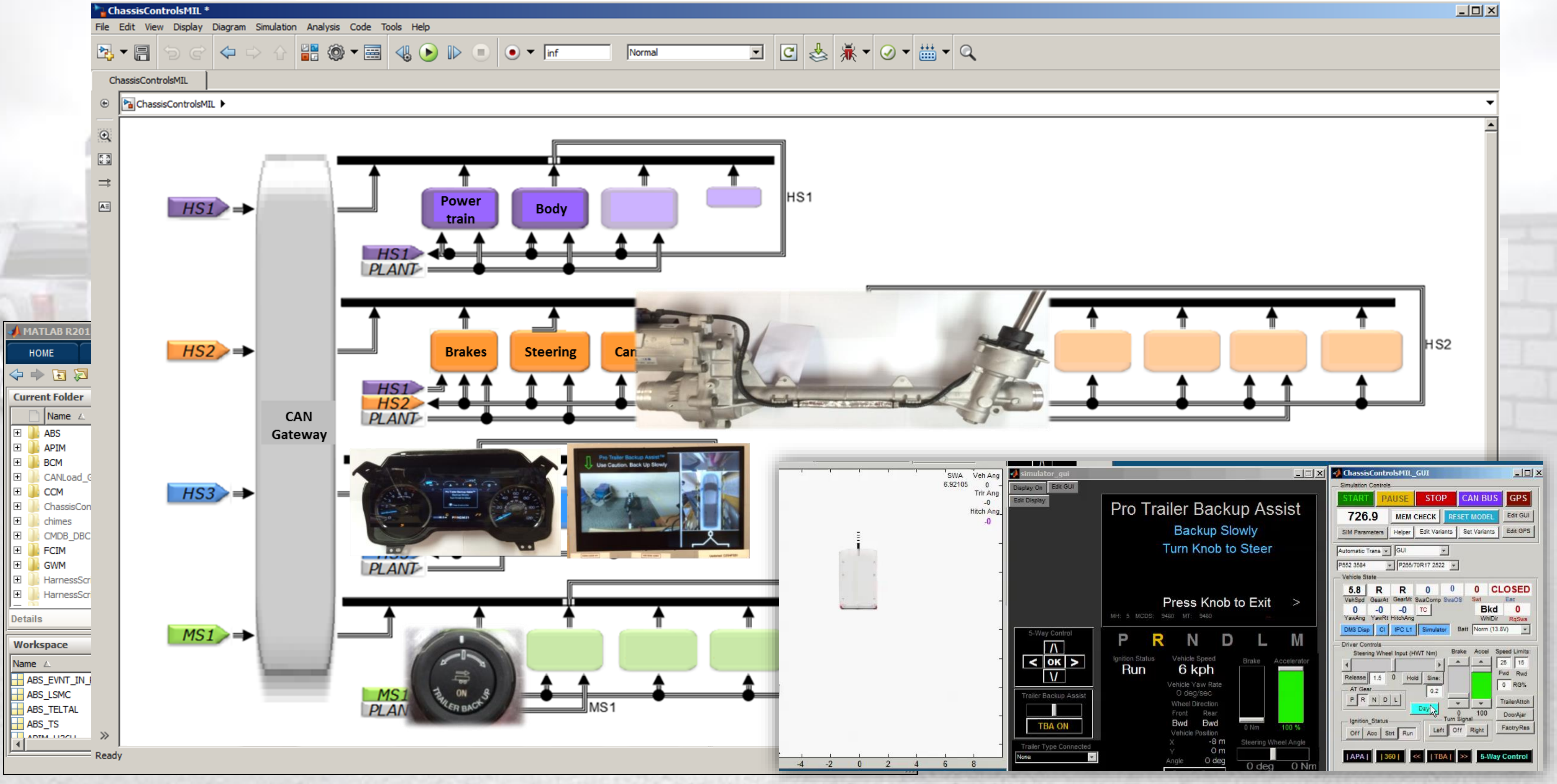
Adding Hardware using Vehicle Network Toolbox



Distributed Network Simulation MIL & HIL



Distributed Network Simulation MIL & HIL

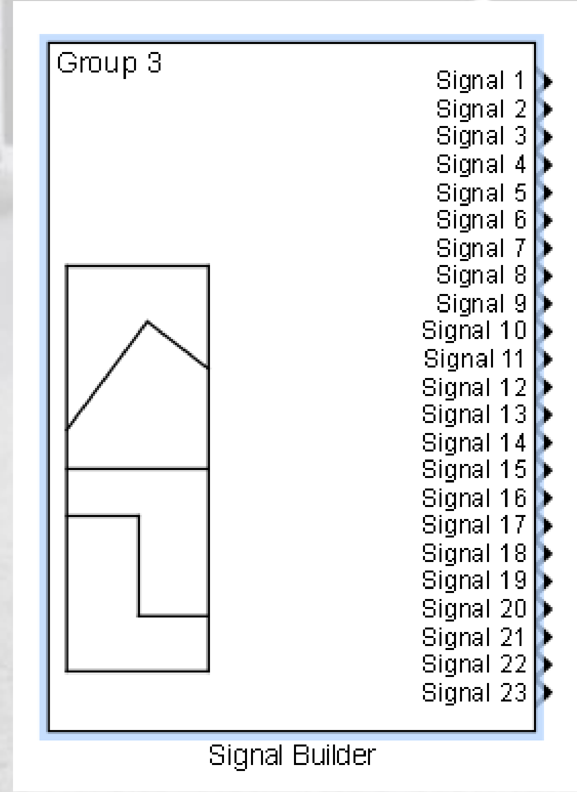


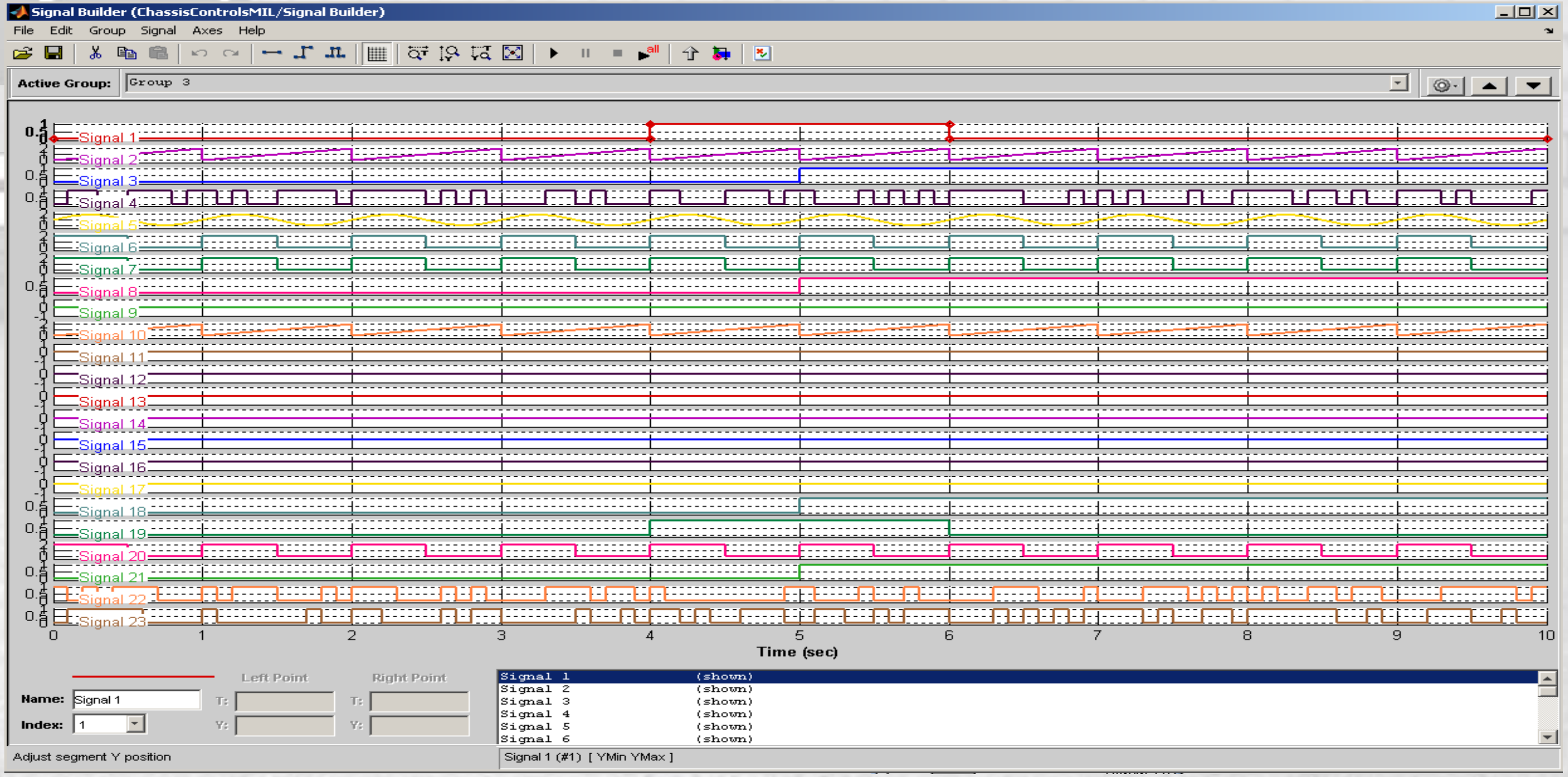
Solutions

1. **REQUIREMENT MODELING**:
   - A modeling methodology for Requirements which captures and simulates the logical parts to ensure the distributed control logical design of requirements works as intended prior to release for software implementation
2. **DISTRIBUTED NETWORK SIMULATION**:
   - A simulation environment which can link multiple Controller modules, CAN Networks, Driver and Vehicle Interactions.
   - It can simulate both MIL (Virtual) and HIL (Hardware) in real-time and each controller can be switched in real-time to either the MIL or HIL version. It can test all systems together or target systems individually at the system engineer's discretion
3. **VALIDATION AND VERIFICATION TOOL:**
   - A tool that can work effectively throughout the Software V process to:
     - ✓ Test and validate requirement models (Down the System V)
     - ✓ Verify that software components and module outputs match the requirement model behavior (Up the System V)

Validation and Verification Tool

- Most verification tools are not intended to handle "simulation of systems" with driver-in-the-loop. Most are designed and focused for unit-level testing and verification.
- Simulink Signal Generator tool works for unit testing of simple models and test cases with a limited and predictable number of inputs and outputs.
- It becomes cumbersome to modify and maintain Signal Generator for highly complex or distributed logic control models with hundreds of potential inputs and outputs





Validation and Verification Tool

- Based on previous experience and not-so-successful attempts with existing verification tools, I developed a unique verification tool that would integrate seamlessly into the Distributed Network Simulation environment.

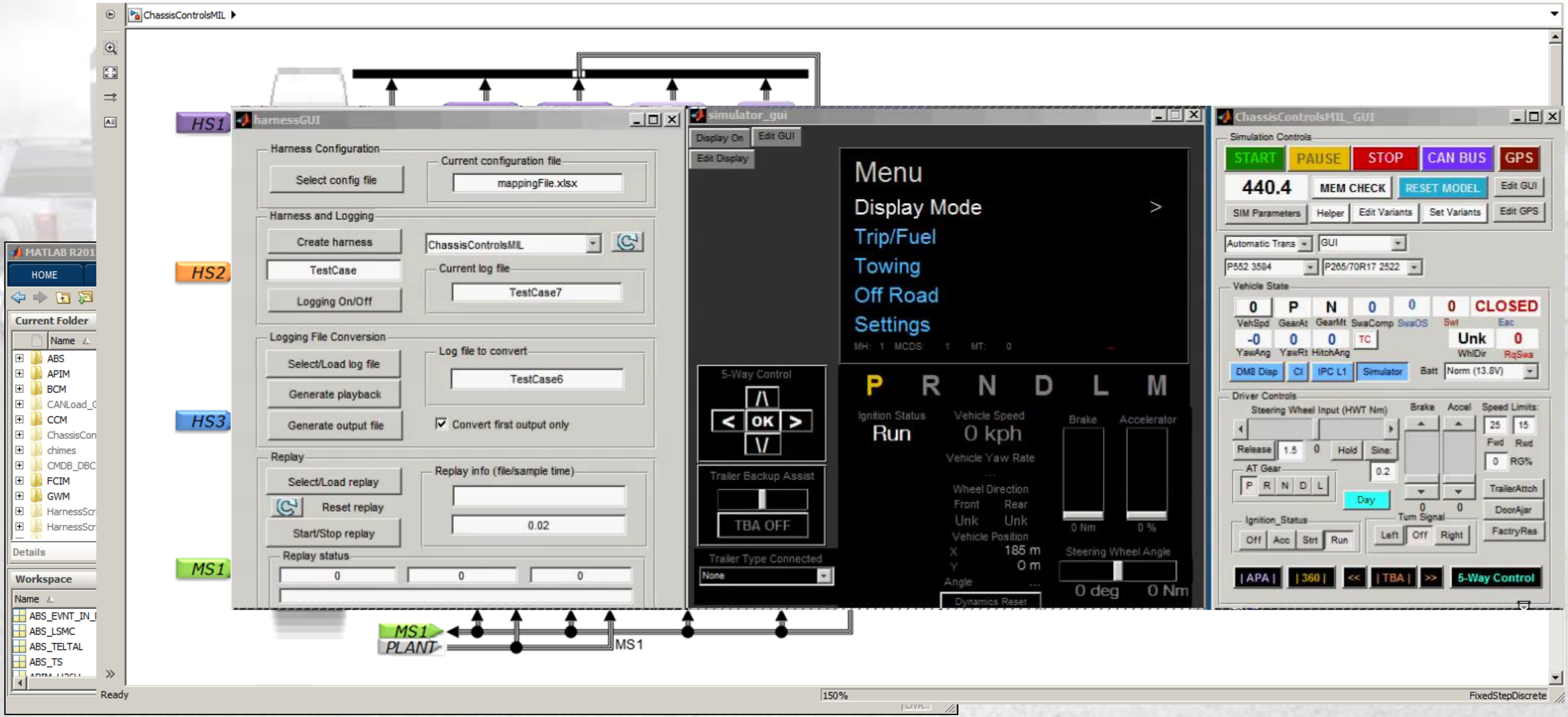
STEP 1: Define the Test Case

Simple Trailer Backup Assist Test Case

1. Driver activates Trailer Backup Assist (Press Button)
2. Driver begins to back-up trailer for a few seconds (Shift to Reverse, Accelerator Pedal)
3. Driver stops the vehicle (Depress Brake Pedal)
4. Driver deactivates Trailer Backup Assist (Press Button)

Validation and Verification Tool

STEP 2: Simulate and Record the Test Case



Validation and Verification Tool

STEP 3: Generate Test Case Replay Script and Master Report

TEST CASE SCRIPT

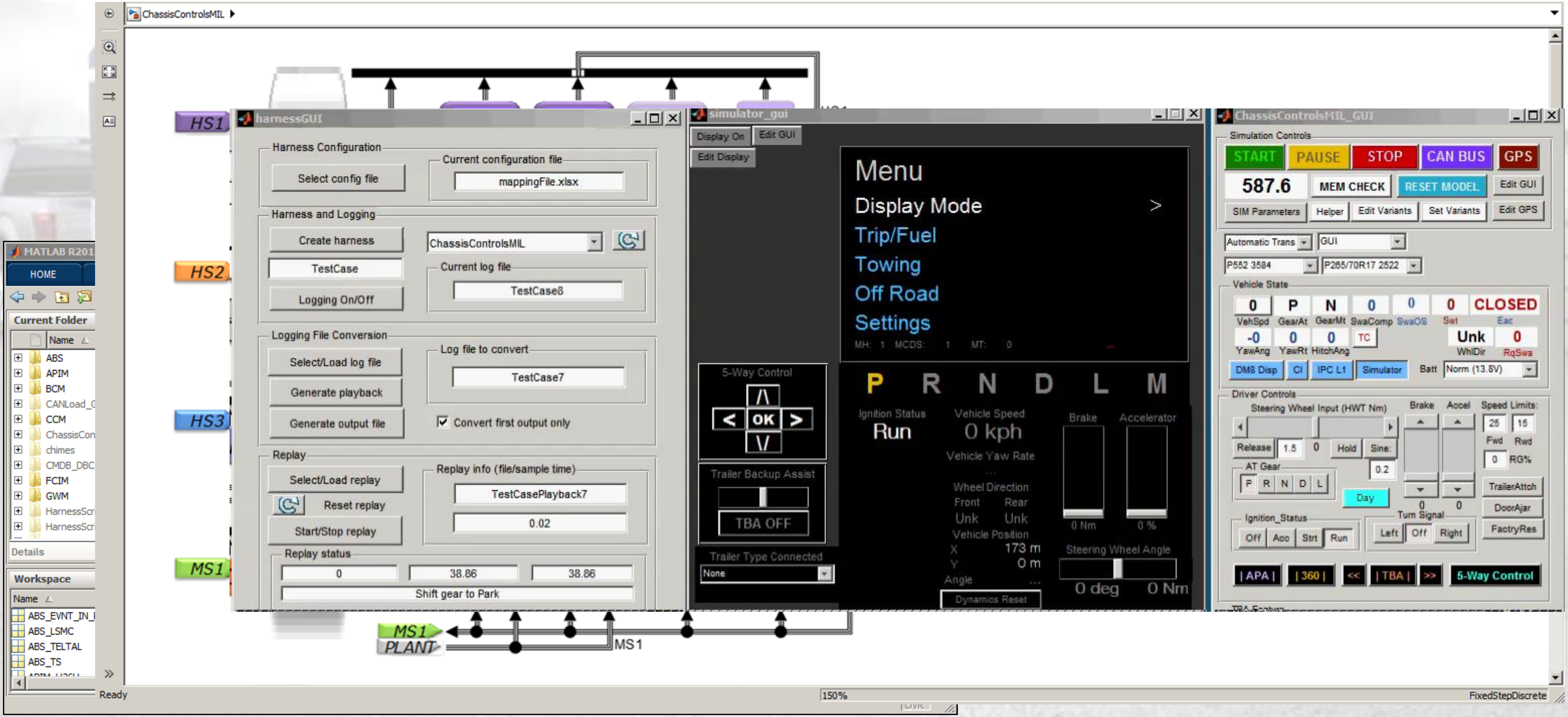
Row	Time	Driver Input
3	0	
4	2.82	TBA Button Pressed
5	3.14	TBA Button Not Pressed
6	4.26	Down Pressed
7	4.5	Down Not Pressed
8	5.54	Up Pressed
9	5.78	Up Not Pressed
10	6.86	Ok Pressed
11	7.06	Ok Not Pressed
12	16.94	Shift gear to Reverse
13	18.78	Accel pedal 25 %
14	19.14	Accel pedal 50 %
15	19.66	Accel pedal 75 %
16	20.98	Accel pedal 50 %
17	21.3	Accel pedal 25 %
18	21.72	Accel pedal 0 %
19	22.34	Brake Pedal 25 %
20	22.76	Brake Pedal 50 %
21	23.16	Brake Pedal 75 %
22	24.12	Brake Pedal 50 %
23	24.46	Brake Pedal 25 %
24	24.82	Brake Pedal 0 %
25	26.64	TBA Button Pressed
26	26.9	TBA Button Not Pressed
27	30.1	Ok Pressed
28	30.42	Ok Not Pressed
29	38.86	Shift gear to Park

TEST CASE MASTER

Row	Time	Driver Input	HMI Request	HMI Status	Camera Status	Setup Status	Steering Angle	Vehicle Speed
3	0		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
4	2.82	TBA Button Pressed	1 HMI	Inactive	Null	Inactive	0 deg	0 kph
5	3.14	TBA Button Not Pressed	1 HMI	Inactive	Null	Inactive	0 deg	0 kph
6	3.26		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
7	4.26	Down Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
8	4.5	Down Not Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
9	5.54	Up Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
10	5.78	Up Not Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
11	6.86	Ok Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
12	7.06	Ok Not Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
13	7.24		2 HMI	ActivateTba	Null	Inactive	0 deg	0 kph
14	7.32		2 HMI	ActivateTba	TbaActive	Inactive	0 deg	0 kph
15	7.46		2 HMI	DeactivateTba	TbaActive	Inactive	0 deg	0 kph
16	7.58		4 HMI	DeactivateTba	TbaActive	Inactive	0 deg	0 kph
17	7.84		4 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
18	15.46		14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
19	16.94	Shift gear to Reverse	14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
20	17.5		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
21	18.78	Accel pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
22	19.14	Accel pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0.004036 kph
23	19.66	Accel pedal 75 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	1.1564 kph
24	20.98	Accel pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	6.101 kph
25	21.3	Accel pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	5.9395 kph
26	21.72	Accel pedal 0 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	5.7276 kph
27	22.34	Brake Pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	5.4148 kph
28	22.76	Brake Pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	3.9163 kph
29	23.16	Brake Pedal 75 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0.98992 kph
30	24.12	Brake Pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
31	24.46	Brake Pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
32	24.82	Brake Pedal 0 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
33	26.64	TBA Button Pressed	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
34	26.9	TBA Button Not Pressed	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
35	27.02		13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
36	30.1	Ok Pressed	13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
37	30.42	Ok Not Pressed	13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
38	32.04		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
39	38.86	Shift gear to Park	1 HMI	Inactive	Null	Inactive	0 deg	0 kph

Validation and Verification Tool

STEP 4: Replay and Record the Test Case



Validation and Verification Tool

STEP 5: Compare Test Case Results against Master

TEST CASE MASTER

Time	Driver Input	HMI Request	HMI Status	Camera Status	Setup Status	Steering Angle	Vehicle Speed
0		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
2.82	TBA Button Pressed	1 HMI	Inactive	Null	Inactive	0 deg	0 kph
3.14	TBA Button Not Pressed	1 HMI	Inactive	Null	Inactive	0 deg	0 kph
3.26		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
4.26	Down Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
4.5	Down Not Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
5.54	Up Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
5.78	Up Not Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
6.86	Ok Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
7.06	Ok Not Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
7.24		2 HMI	ActivateTba	Null	Inactive	0 deg	0 kph
7.32		2 HMI	ActivateTba	TbaActive	Inactive	0 deg	0 kph
7.46		2 HMI	DeactivateTba	TbaActive	Inactive	0 deg	0 kph
7.58		4 HMI	DeactivateTba	TbaActive	Inactive	0 deg	0 kph
7.84		4 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
15.46		14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
16.94	Shift gear to Reverse	14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
17.5		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
18.78	Accel pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
19.14	Accel pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0.004036 kph
19.66	Accel pedal 75 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	1.1564 kph
20.98	Accel pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	6.101 kph
21.3	Accel pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	5.9395 kph
21.72	Accel pedal 0 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	5.7276 kph
22.34	Brake Pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	5.4148 kph
22.76	Brake Pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	3.9163 kph
23.16	Brake Pedal 75 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0.98992 kph
24.12	Brake Pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
24.46	Brake Pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
24.82	Brake Pedal 0 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
26.64	TBA Button Pressed	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
26.9	TBA Button Not Pressed	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
27.02		13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
30.1	Ok Pressed	13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
30.42	Ok Not Pressed	13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
32.04		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
38.86	Shift gear to Park	1 HMI	Inactive	Null	Inactive	0 deg	0 kph

TEST CASE ITERATIVE

Time	Driver Input	HMI Request	HMI Status	Camera Status	Setup Status	Steering Angle	Vehicle Speed
0		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
5.1	TBA Button Pressed	1 HMI	Inactive	Null	Inactive	0 deg	0 kph
5.42	TBA Button Not Pressed	1 HMI	Inactive	Null	Inactive	0 deg	0 kph
5.54		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
6.54	Down Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
6.78	Down Not Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
7.82	Up Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
8.06	Up Not Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
9.14	Ok Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
9.34	Ok Not Pressed	2 HMI	Inactive	Null	Inactive	0 deg	0 kph
9.72		2 HMI	ActivateTba	Null	Inactive	0 deg	0 kph
9.8		2 HMI	ActivateTba	TbaActive	Inactive	0 deg	0 kph
9.94		2 HMI	DeactivateTba	TbaActive	Inactive	0 deg	0 kph
10.06		4 HMI	DeactivateTba	TbaActive	Inactive	0 deg	0 kph
10.32		4 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
17.94		14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
19.22	Shift gear to Reverse	14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
19.78		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
21.06	Accel pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
21.42	Accel pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0.004036 kph
21.94	Accel pedal 75 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	1.1564 kph
23.26	Accel pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	6.101 kph
23.58	Accel pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	5.9395 kph
24	Accel pedal 0 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	5.7276 kph
24.62	Brake Pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	5.4148 kph
25.04	Brake Pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	3.9163 kph
25.44	Brake Pedal 75 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0.98992 kph
26.4	Brake Pedal 50 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
26.74	Brake Pedal 25 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
27.1	Brake Pedal 0 %	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
28.92	TBA Button Pressed	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
29.18	TBA Button Not Pressed	5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
29.3		13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
32.38	Ok Pressed	13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
32.7	Ok Not Pressed	13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
34.32		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
41.14	Shift gear to Park	1 HMI	Inactive	Null	Inactive	0 deg	0 kph

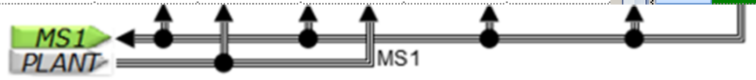
Validation and Verification Tool

STEP 5: Compare Test Case Results against Master

1											
2	T										
3	0										
4	2.8200000	TBA Button	11	1		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
5	3.1400000	TBA Button	11	0		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
6	3.2600000					2 HMI	Inactive	Null	Inactive	0 deg	0 kph
7	4.2600000	Down Pres	7	1		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
8	4.5	Down Not	7	0		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
9	5.5400000	Up Pressed	6	1		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
10	5.7800000	Up Not Pre	6	0		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
11	6.8600000	Ok Pressed	10	1		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
12	7.0600000	Ok Not Pre	10	0		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
13	7.2400000					2 HMI	ActivateTb	Null	Inactive	0 deg	0 kph
14	7.3200000					2 HMI	ActivateTb	TbaActive	Inactive	0 deg	0 kph
15	7.4600000					2 HMI	Deactivate	TbaActive	Inactive	0 deg	0 kph
16	7.5800000					4 HMI	Deactivate	TbaActive	Inactive	0 deg	0 kph
17	7.8400000					4 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
18	15.460000					14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
19	16.940000	Shift gear t	4	1		14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
20	17.5					5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
21	18.780000	Accel pedal	3	25		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
22	19.140000	Accel pedal	3	50		5 HMI	Inactive	TbaActive	Inactive	0 deg	0.004036 k
23	19.660000	Accel pedal	3	75		5 HMI	Inactive	TbaActive	Inactive	0 deg	1.1564 kph
24	20.980000	Accel pedal	3	50		5 HMI	Inactive	TbaActive	Inactive	0 deg	6.101 kph
25	21.300000	Accel pedal	3	25		5 HMI	Inactive	TbaActive	Inactive	0 deg	5.9395 kph
26	21.720000	Accel pedal	3	0		5 HMI	Inactive	TbaActive	Inactive	0 deg	5.7276 kph
27	22.340000	Brake Peda	2	25		5 HMI	Inactive	TbaActive	Inactive	0 deg	5.4148 kph
28	22.760000	Brake Peda	2	50		5 HMI	Inactive	TbaActive	Inactive	0 deg	3.9163 kph
29	23.160000	Brake Peda	2	75		5 HMI	Inactive	TbaActive	Inactive	0 deg	0.98992 kp
30	24.120000	Brake Peda	2	50		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
31	24.460000	Brake Peda	2	25		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
32	24.820000	Brake Peda	2	0		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
33	26.640000	TBA Button	11	1		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
34	26.900000	TBA Button	11	0		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
35	27.020000					13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
36	30.100000	Ok Pressed	10	1		13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
37	30.420000	Ok Not Pre	10	0		13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
38	32.040000					1 HMI	Inactive	Null	Inactive	0 deg	0 kph
39	38.860000	Shift gear t	4	0		1 HMI	Inactive	Null	Inactive	0 deg	0 kph

4	5.1000000	TBA Button	11	1		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
5	5.4200000	TBA Button	11	0		1 HMI	Inactive	Null	Inactive	0 deg	0 kph
6	5.5400000					2 HMI	Inactive	Null	Inactive	0 deg	0 kph
7	6.5400000	Down Pres	7	1		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
8	6.7800000	Down Not	7	0		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
9	7.8200000	Up Pressed	6	1		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
10	8.0600000	Up Not Pre	6	0		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
11	9.1399999	Ok Pressed	10	1		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
12	9.3400000	Ok Not Pre	10	0		2 HMI	Inactive	Null	Inactive	0 deg	0 kph
13	9.7200000					2 HMI	ActivateTb	Null	Inactive	0 deg	0 kph
14	9.8000000					2 HMI	ActivateTb	TbaActive	Inactive	0 deg	0 kph
15	9.9400000					2 HMI	Deactivate	TbaActive	Inactive	0 deg	0 kph
16	10.060000					4 HMI	Deactivate	TbaActive	Inactive	0 deg	0 kph
17	10.320000					4 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
18	17.940000					14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
19	19.220000	Shift gear t	4	1		14 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
20	19.780000					5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
21	21.060000	Accel pedal	3	25		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
22	21.420000	Accel pedal	3	50		5 HMI	Inactive	TbaActive	Inactive	0 deg	0.004036 k
23	21.940000	Accel pedal	3	75		5 HMI	Inactive	TbaActive	Inactive	0 deg	1.1564 kph
24	23.259999	Accel pedal	3	50		5 HMI	Inactive	TbaActive	Inactive	0 deg	6.101 kph
25	23.580000	Accel pedal	3	25		5 HMI	Inactive	TbaActive	Inactive	0 deg	5.9395 kph
26	24	Accel pedal	3	0		5 HMI	Inactive	TbaActive	Inactive	0 deg	5.7276 kph
27	24.620000	Brake Peda	2	25		5 HMI	Inactive	TbaActive	Inactive	0 deg	5.4148 kph
28	25.040000	Brake Peda	2	50		5 HMI	Inactive	TbaActive	Inactive	0 deg	3.9163 kph
29	25.440000	Brake Peda	2	75		5 HMI	Inactive	TbaActive	Inactive	0 deg	0.98992 kp
30	26.400000	Brake Peda	2	50		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
31	26.740000	Brake Peda	2	25		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
32	27.100000	Brake Peda	2	0		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
33	28.920000	TBA Button	11	1		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
34	29.180000	TBA Button	11	0		5 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
35	29.300000					13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
36	32.379999	Ok Pressed	10	1		13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
37	32.700000	Ok Not Pre	10	0		13 HMI	Inactive	TbaActive	Inactive	0 deg	0 kph
38	34.320000					1 HMI	Inactive	Null	Inactive	0 deg	0 kph
39	41.139999	Shift gear t	4	0		1 HMI	Inactive	Null	Inactive	0 deg	0 kph

ABS_EVNT_IN_I
ABS_LSMC
ABS_TELTAL
ABS_TS



Ready

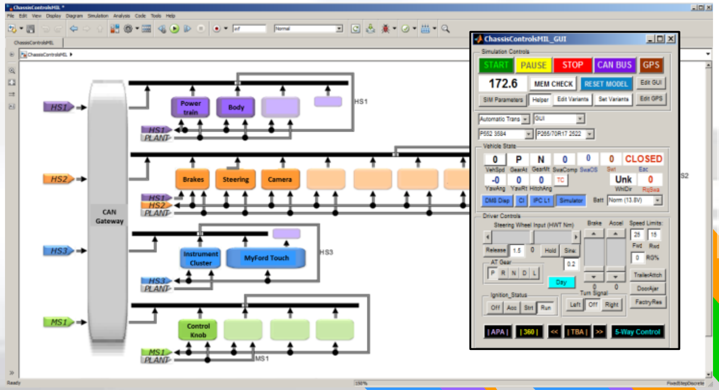
150%

FixedStepDiscrete

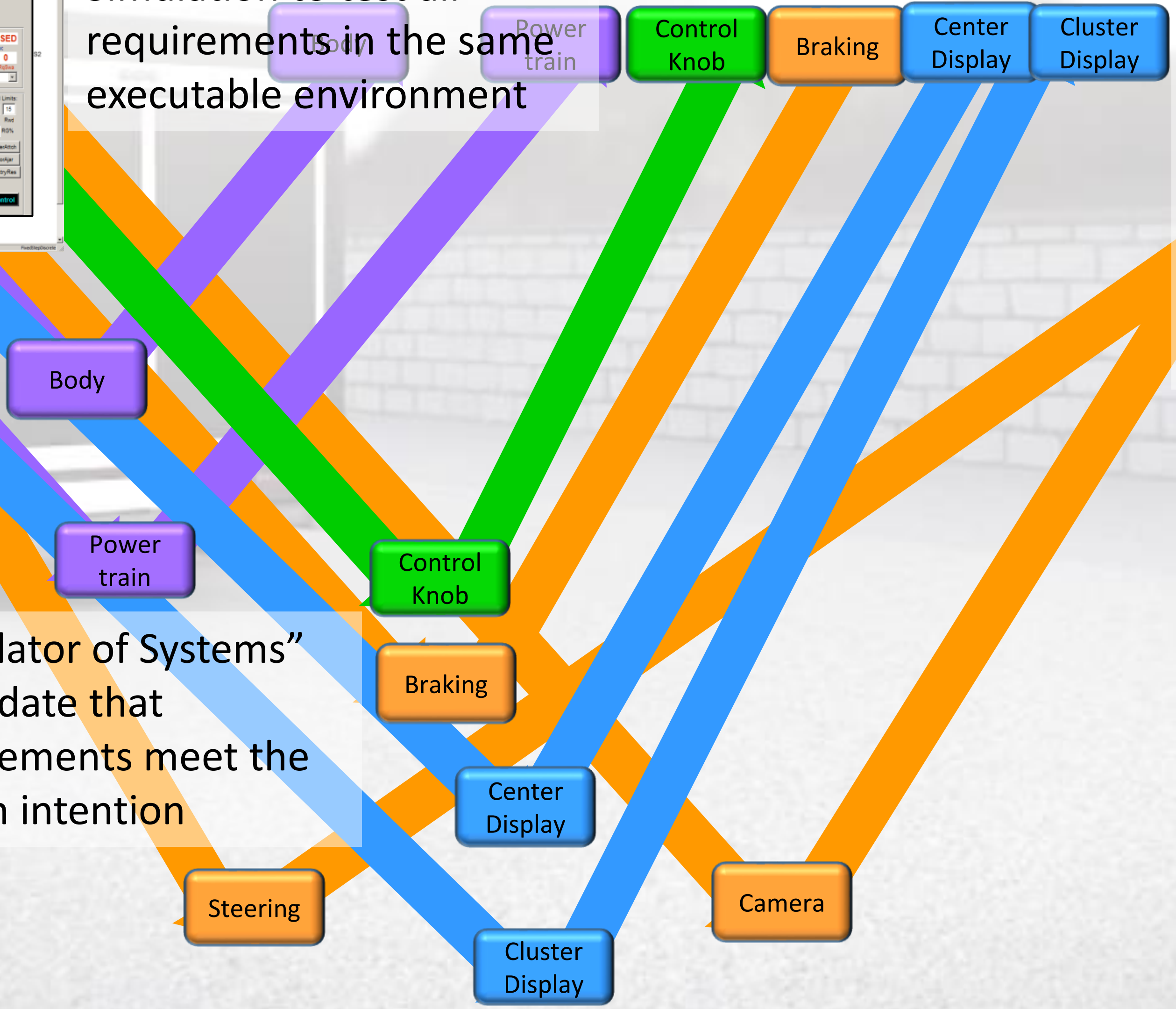
Validation and Verification Tool

- Recording a Master Test Case and generating a Master Test Report creates a document which captures the system outputs based on the Requirement Models.
- The Master Test Case behavior can be replayed repeatedly to verify the system for new software releases of each module for any MIL/HIL configuration of the Distributed Network Simulation.
- Master Test Reports can be provided to engineers and suppliers to define how their module should react in the system. They can be customized and targeted towards specific modules so that only the relevant test data is generated in the report.
- Iterative Test Reports can be compared against the Master to exactly identify logical defects within the context of Simulation Time and Driver Actions.
- Test Reports can be configured to include any system inputs, outputs, or parameters that exist in the simulation environment
- Can be used in conjunction with Coverage Tool to track coverage metrics.
- In conjunction with Vector CANape (CAN and Video logging), all test case data can be logged into a single synchronous timeline for evidence and review.

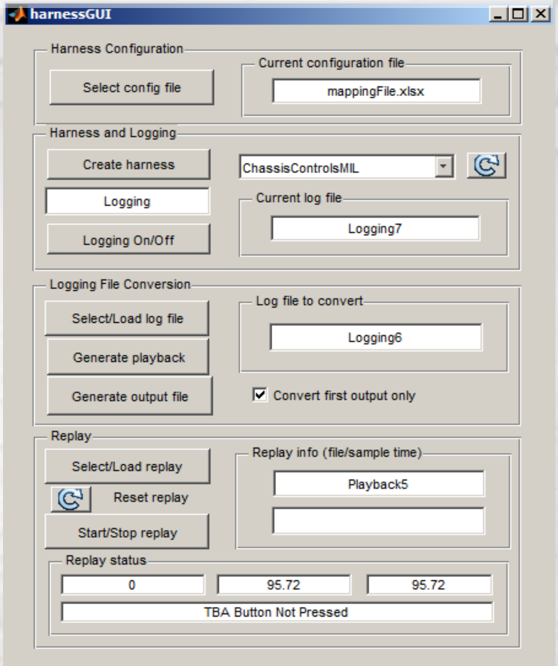
Solutions Overview

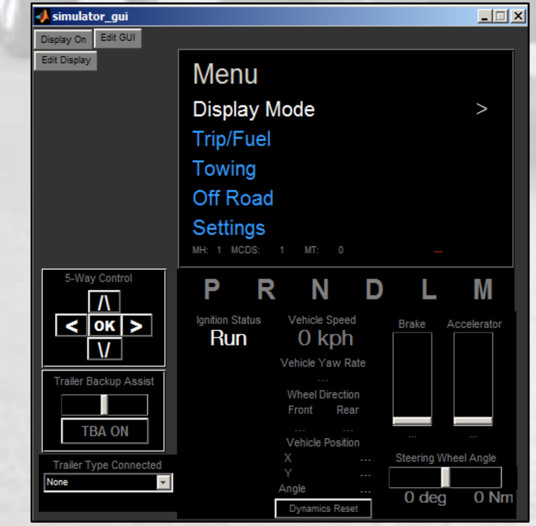


Requirement Model Simulation to test all requirements in the same executable environment



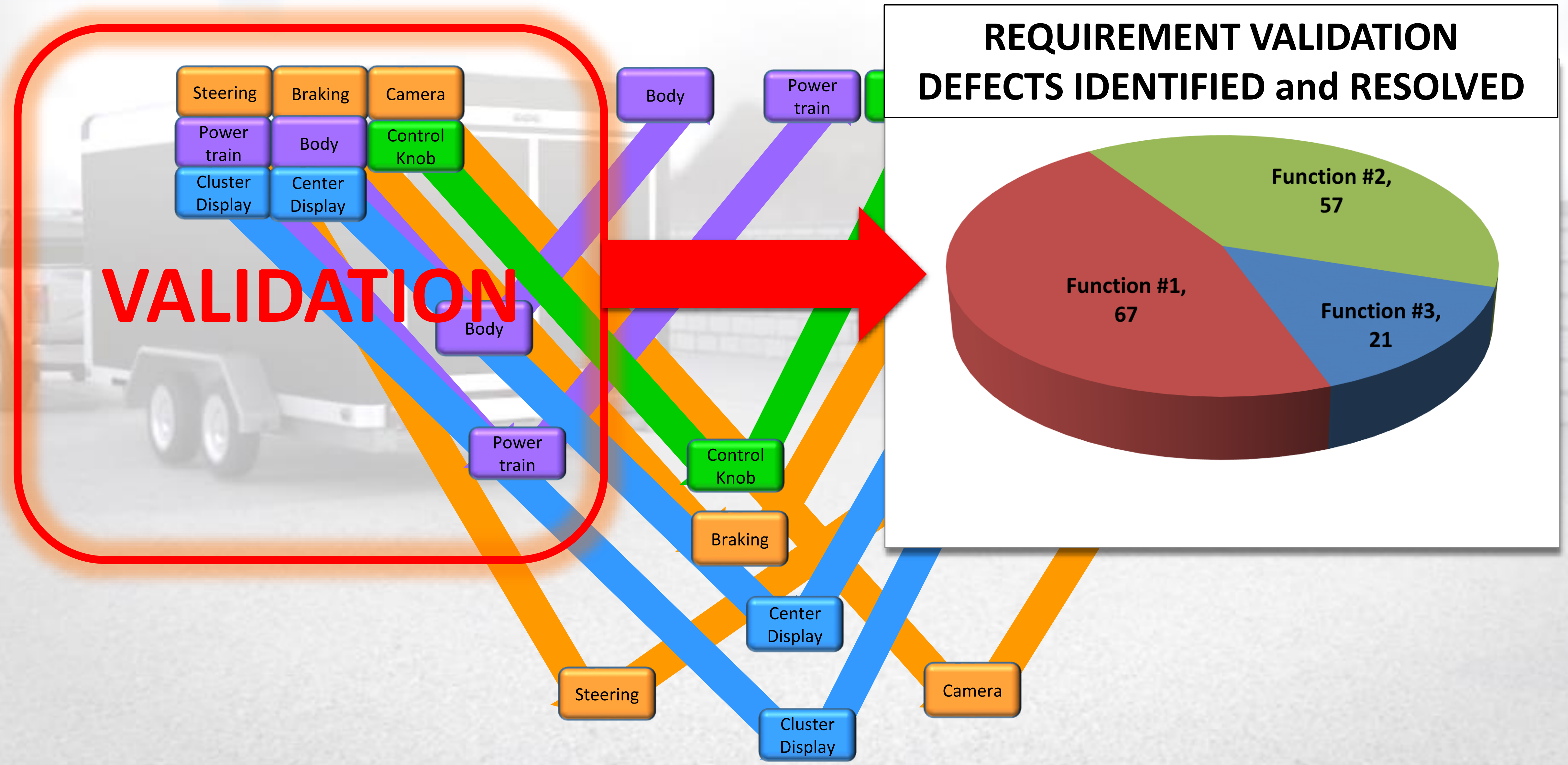
Validation and Verification Tool to record Master Test Cases and generate repeatable test scripts which can be replayed in any MIL/HIL integration configurations





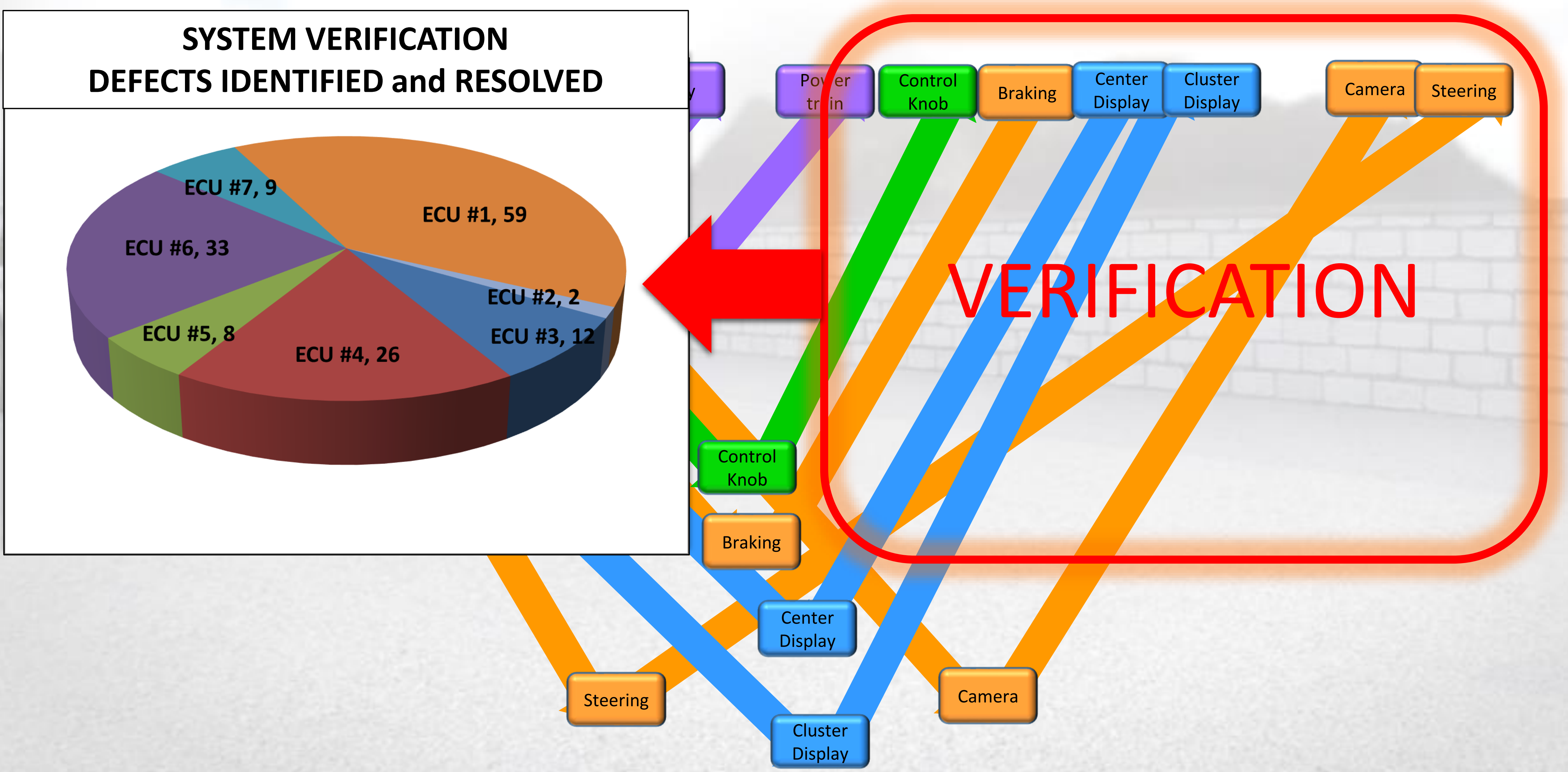
"Simulator of Systems" to validate that requirements meet the Design intention

Results



Steering
Braking
Camera
Power train
Body
Control Knob
Cluster Display
Center Display
VALIDATION
Body
Power train
Body
Power train
Control Knob
Braking
Center Display
Steering
Cluster Display
Camera
REQUIREMENT VALIDATION
DEFECTS IDENTIFIED and RESOLVED
Function #2, 57
Function #1, 67
Function #3, 21

Results


SYSTEM VERIFICATION
DEFECTS IDENTIFIED and RESOLVED
ECU #1, 59
ECU #2, 2
ECU #3, 12
ECU #4, 26
ECU #5, 8
ECU #6, 33
ECU #7, 9
Control Knob
Braking
Center Display
Cluster Display
Camera
Steering
VERIFICATION
Control Knob
Braking
Center Display
Steering
Cluster Display
Camera

What's Next?

Validation

Concept → MIL

Emphasis:

Readability

Output:

Requirements

Verification

SIL → HIL

Emphasis:

Testability, Traceability

Output:

Prototypes

Implementation

MIL → SIL

Emphasis:

Efficiency, Compliance

Output:

Software

What's Next?

Validation

Concept → MIL

Emphasis:

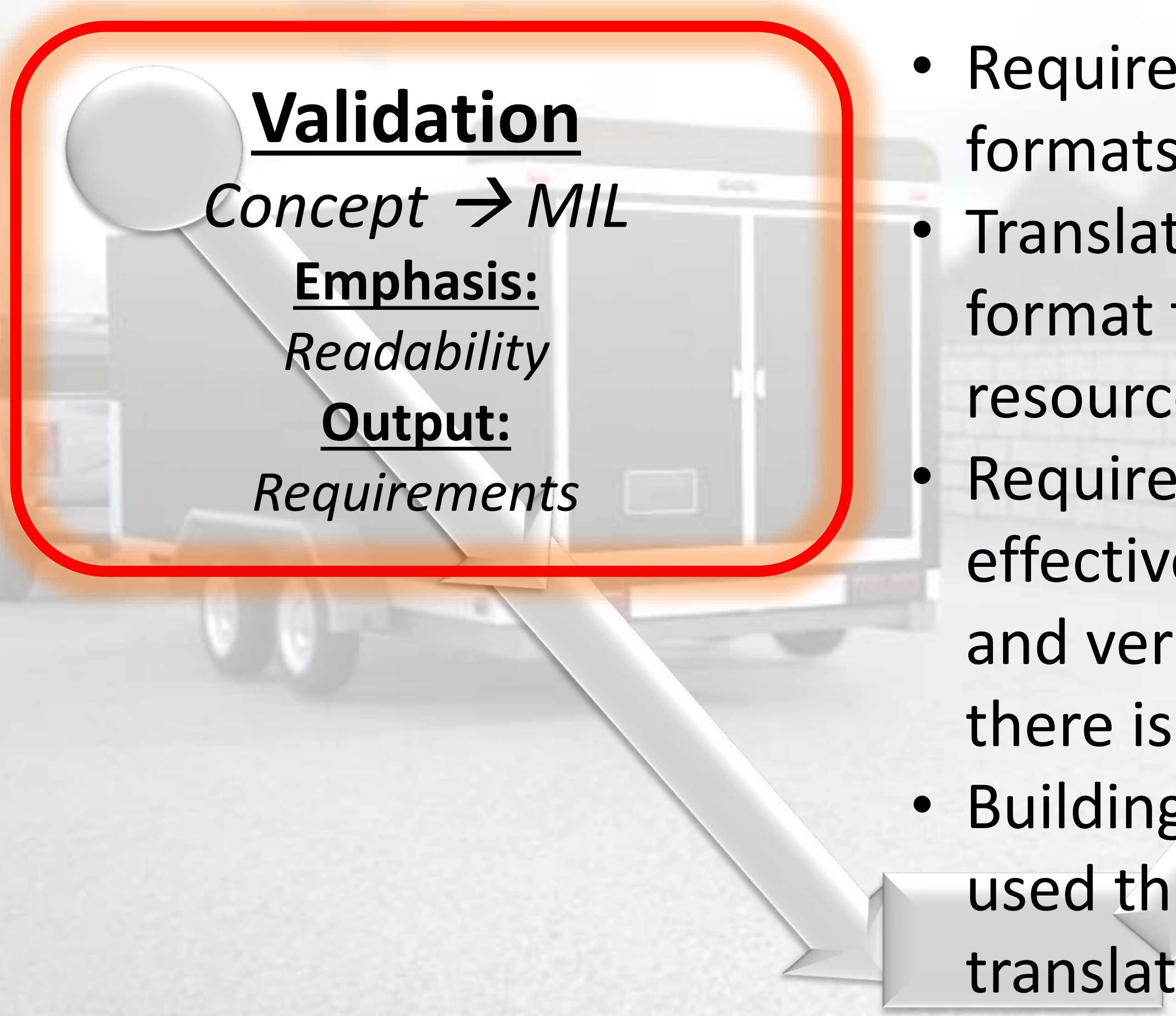
Readability

Output:

Requirements

- Requirement Validation step is often skipped, overlooked, or misunderstood.
- Requirement Validation skillsets and tools are undeveloped and unrecognized
- Few tools exist to simulate and validate requirements.
- An ideal tool would provide the ability to simulate and generate requirements from a model the same way that tools exist to generate, test, and verify code and hardware from a model.

What’s Next?



- Requirement “modeling” is also done in formats that are non-executable.
- Translation from one tool, language, or format to another takes significant time and resource and introduces errors in translation.
- Requirement Modeling in Matlab is uniquely effective and efficient when code generation and verification is already done in Matlab – there is no translation needed!
- Building an executable model that can be used throughout the System V without translation is a **HUGE** efficiency gain and the essence of Model-Based Design.

Thank you for your time and attention! ☺

Acknowledgements

John Lee	MathWorks
Dr. Darrel Recker	Ford Motor Company
Brad Hochrein	Ford Motor Company

Comments and Feedback can be directed to:

Nate Rolfes

nrolfes@ford.com