# Develop a brake-by-wire system for Level 4 (L4) autonomous trucks based on Model-Based Design (MBD)

Xiaoling Han

Sr. Director of Vehicle Control Integration and Sensors

# Content

**Basic introduction**

- Project introduction
- Problem Statement

**Solution and results**

- Solution
- Results
- Tools used

**Conclusion**

- Conclusion

# Company introduction: TuSimple is a global autonomous technology company revolutionizing the estimated $4 trillion global freight market.

**Project introduction: there are many critical drive-by-wire systems needed for Level 4 (L4) autonomous trucks and a fully redundant braking system is essential for a driver-less application.**

| | |
|---|---|
| Engine | Steering |
| Braking | Transmission |
| Lighting | Power |

**Problem statement: basic requirements of brake-by-wire are accuracy, low latency, safety, etc. Safety is the top priority.**

## Safety

- Avoid unintended deceleration (FP*)
- Avoid low/no braking force (FN*)
- …

## Performance

- Deceleration accuracy: <= 10%
- Settling time: <= 3s
- Command latency: <=20ms
- …

*** FP: False Position**
*** FN: False Negative**

## Project introduction: design a fully redundant L4 brake-by-wire system, including EE architecture, hardware and software.

- EE Architecture
- Hardware
- Software
- Simulation
- Testing

# Content

**Basic introduction**

- Project introduction
- Problem Statement

**Solution and results**

- Solution
- Results
- Tools used

**Conclusion**

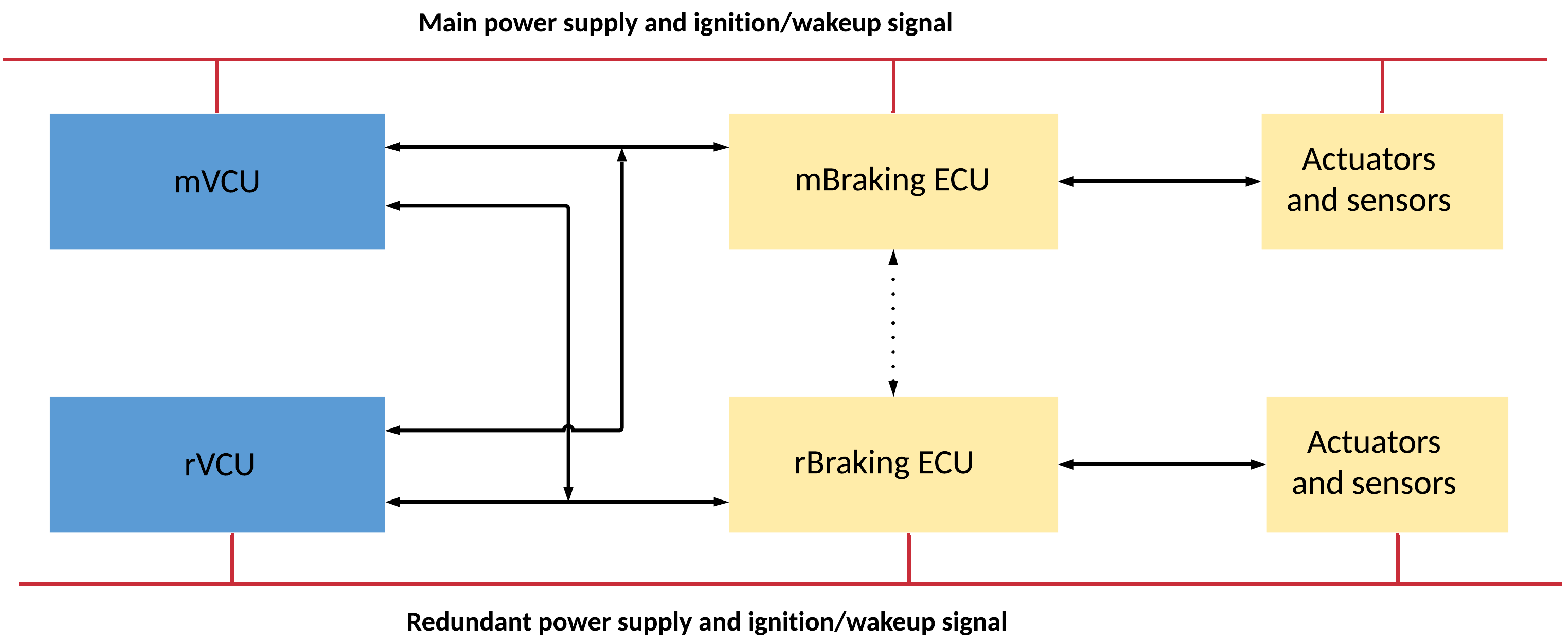
- Conclusion

**System analysis: based on HARA* analysis, FSR* and FMEA*, we can understand the safety goals, functional safety requirements, most critical functions in the braking system and provide related mitigation solutions and design.**

| Functional Safety Requirement Worksheet | |
|---|---|
| System Element/Module: | Vehicle Control |
| FSR#: | FSR 2 |
| Functional Safety Requirement | Shall Avoid Sending Erroneous Deceleration Request |
| Functions Operating Mode | Operating |
| Module Primary Functions: | Send Engine Brake Torque Request to Engine, Send Foundation Brake Pressure to Brake |
| Element ASIL | ASIL D |
| Affected Higher FSR and/or Safety Goals/Numbers | N/A |
| Will the ASIL be Decomposed? | Yes - ASIL D = ASIL B(D) + ASIL B(D) |
| Processing/Cycle Time | 20 ms |
| What is the longest delay allowable for fault | 3 cycles in a row (60ms) |
| What is the longest delay allowable for completing the FuSA Mitigation | Reference ODD |

*** HARA: Hazard Analysis and Risk Assessment**
*** FSR: Functional Safety Requirement**
*** FMEA: Failure Mode and Effects Analyses**

**Solution: the redundant braking architecture includes dual ECU* (mBraking ECU* + rBraking ECU), dual braking VCU (mVCU + rVCU), dual power supply, dual ignition/wakeup signals.**

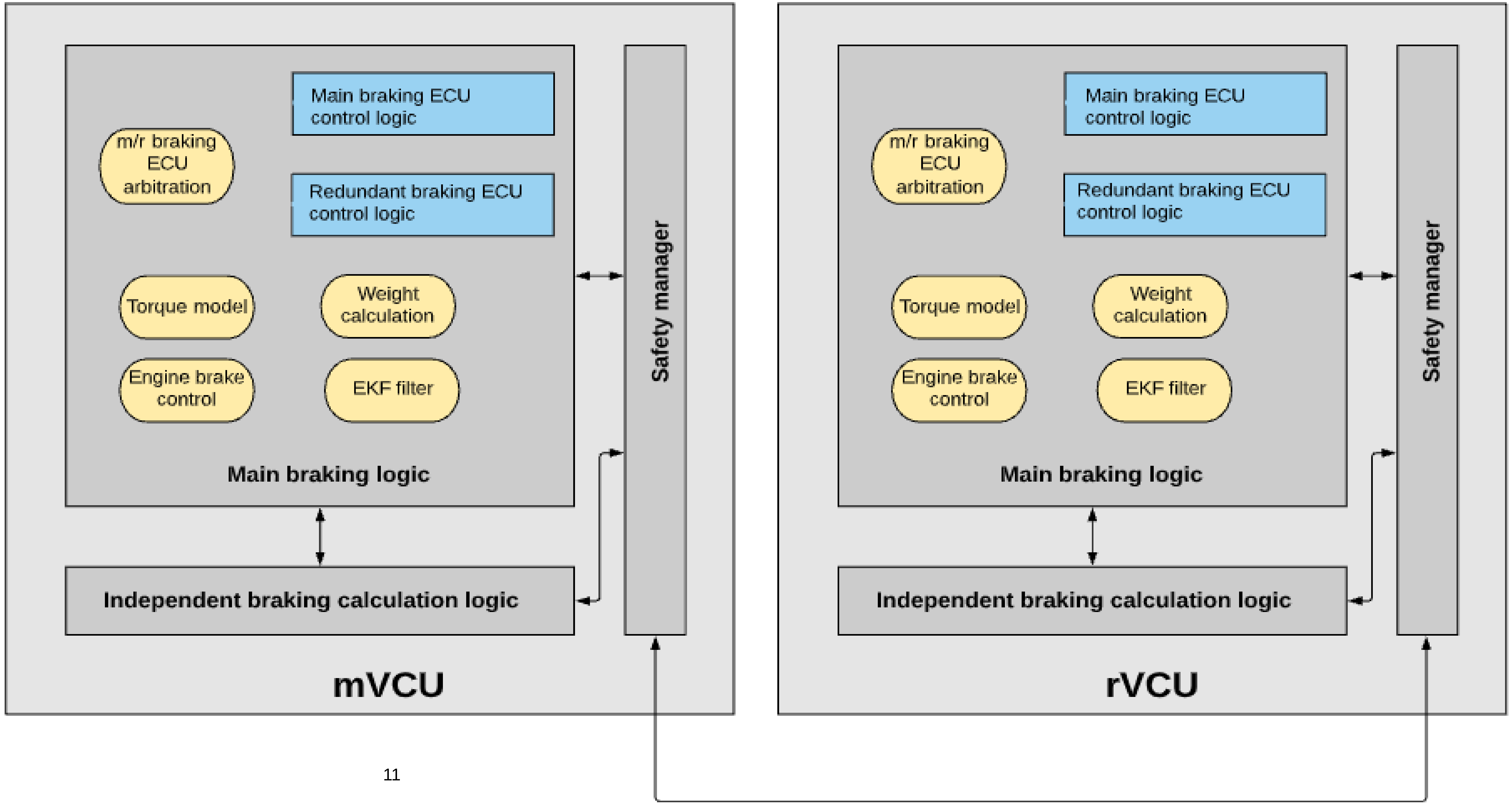



* VCU: Vehicle Control Unit
* ECU: Electronic Control Unit

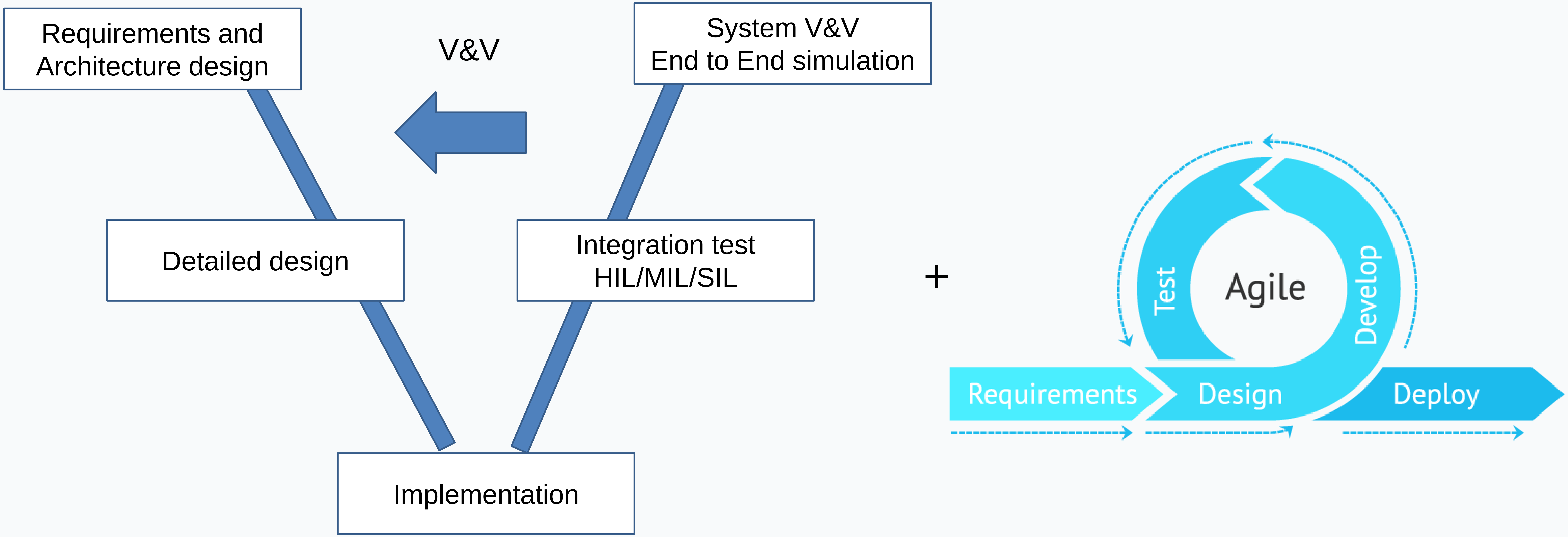
* m: main
* r: redundant

**Solution: we develop the VCU software based on the MBD, since this will help us to work on a virtual system by MIL/SIL (permits multiple design iterations without impacting real hardware that may be expensive); we use AUTOSAR architecture because this is helpful to put everything together like building LEGO blocks.**

## Solution: VCU software has symmetric design with both main braking logic and safety monitoring logic in each VCU. To coordinate the two VCUs to work together, arbitration logic is necessary.

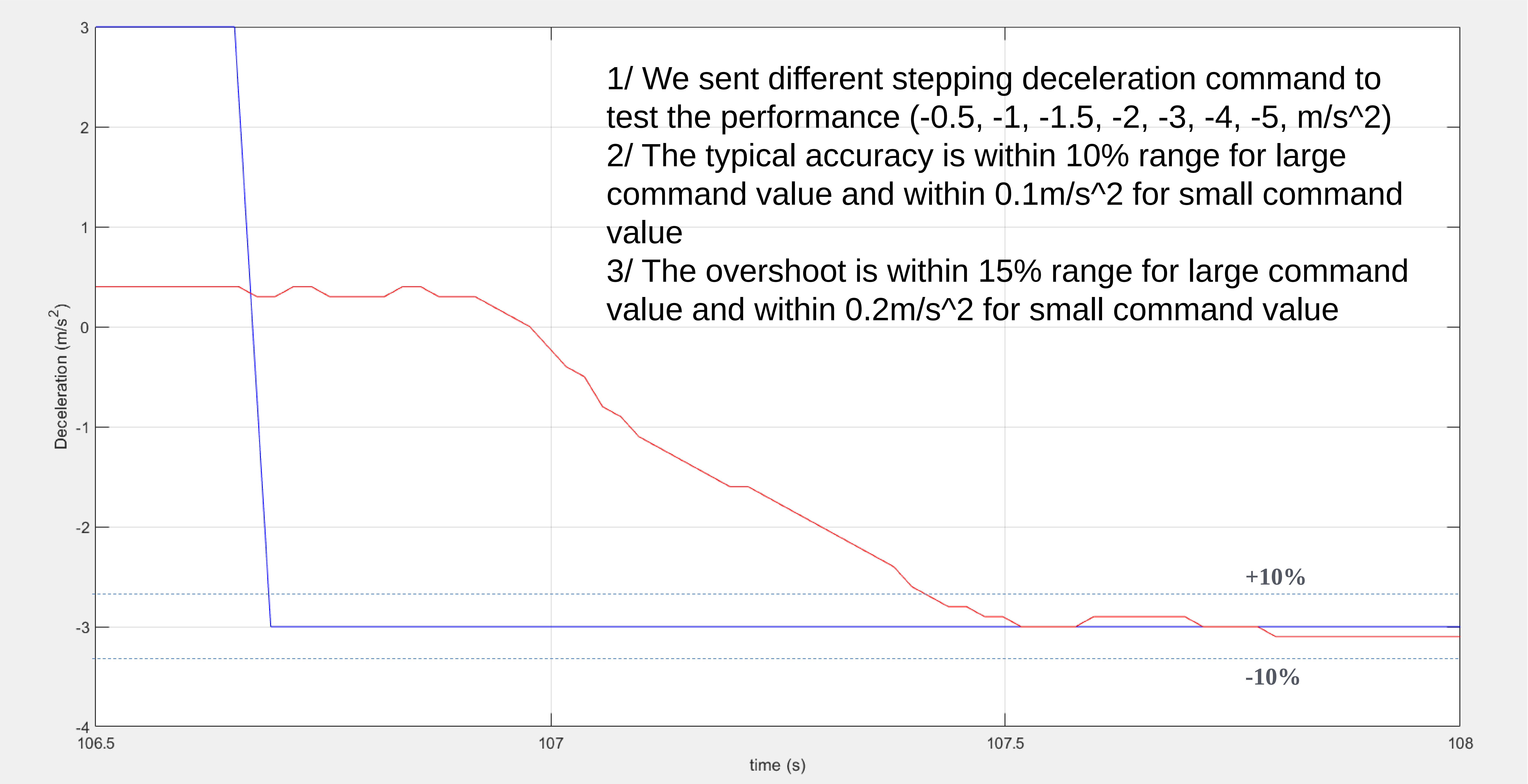


**Process: V cycle is the basic process that we are following during the MBD development. We are combing the sprint of Agile development as well into our process.**

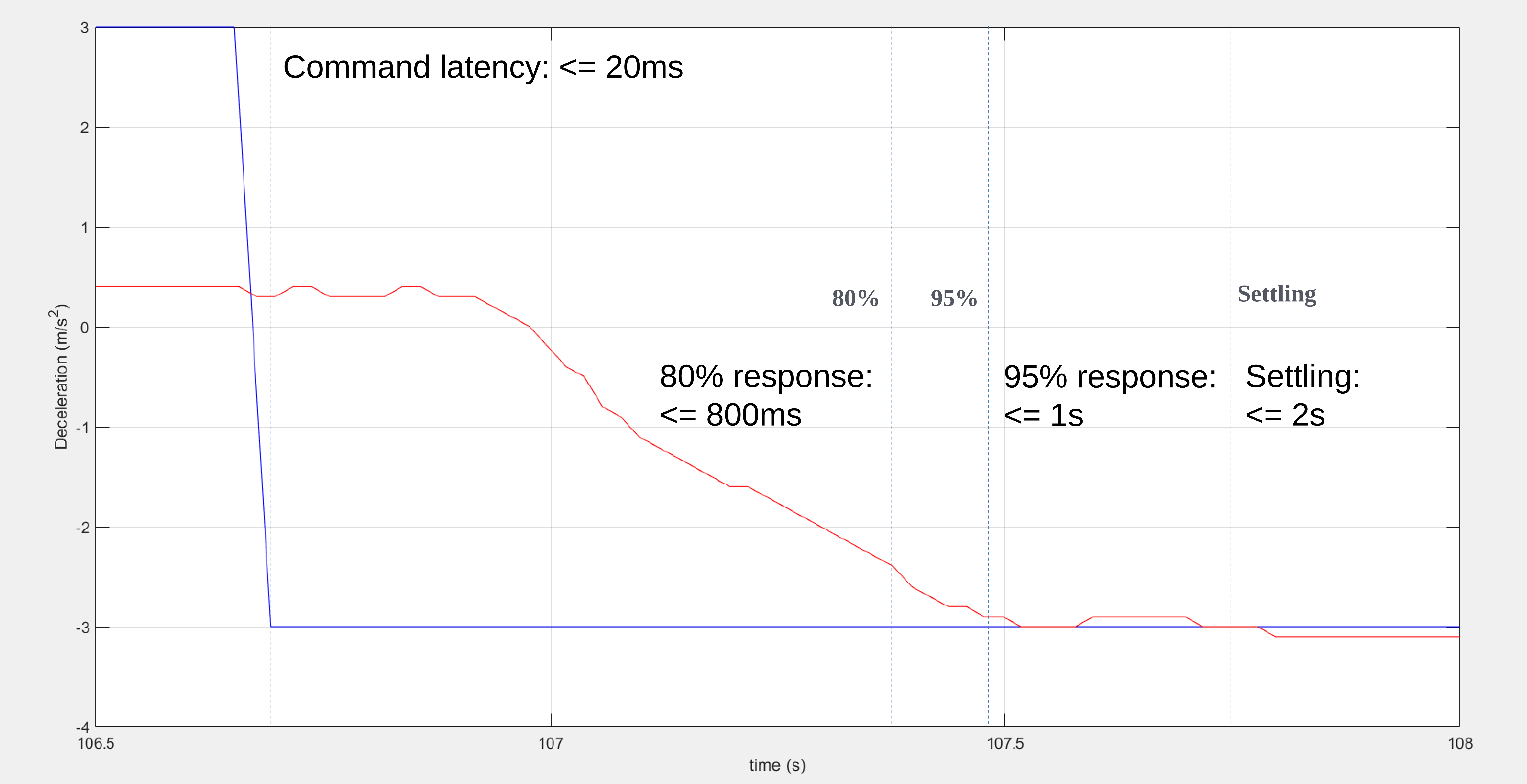



- **The above process can support quick iteration (1 epic release per month) + fast pace software feature release (patch release in 24 hours, feature release in 72 hours).**
- **MBD is essential for our VCU development since it allows us to collaborate and integrate easily.**

# Result: from the road testing, the control accuracy of the braking performance can meet our requirements within the +/-10% accuracy and less than 15% overshoot.

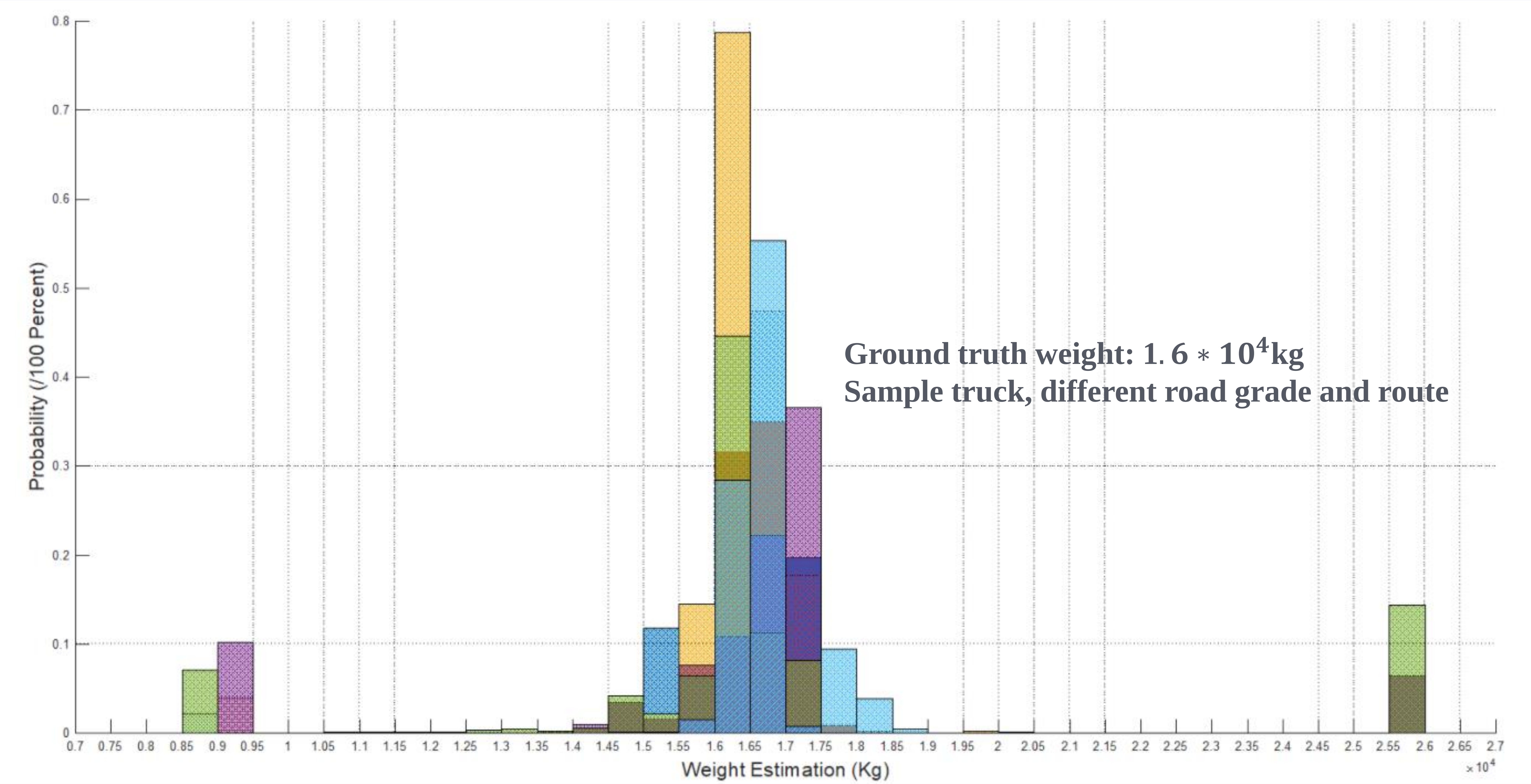


**Result: from the road testing, for most cases, the latency of the braking performance can achieve <=20ms for command latency, <= 800 ms for 80% of ADS* request, <1s for 95% ADS request, and the settling time is less than 2s**



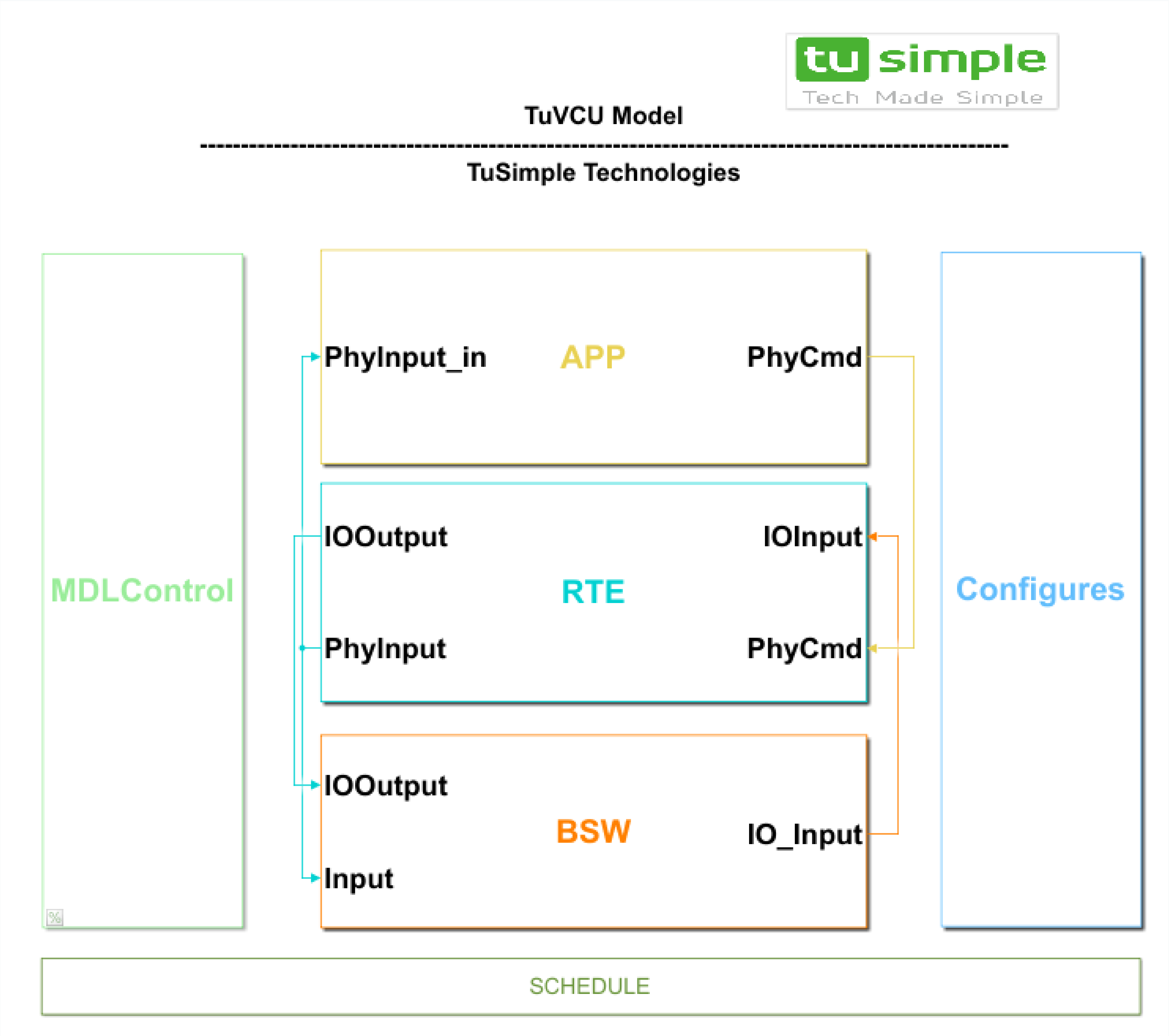

*** ADS: Autonomous Driving System**

# Result: from the road testing, the accuracy of torque model is in +/-10% range and the weight/COG estimation is in +/-10%.




*** COG: Central Of Gravity** **Different color stands for different trips**

**Developing tools: MATLAB/Simulink/Stateflow with many in house design scripts (variant control, compile control, Model Reference control, etc). We use Embedded Coder to generate code automatically because it makes the Simulink models of the control system the "single source of truth".**




## Model development

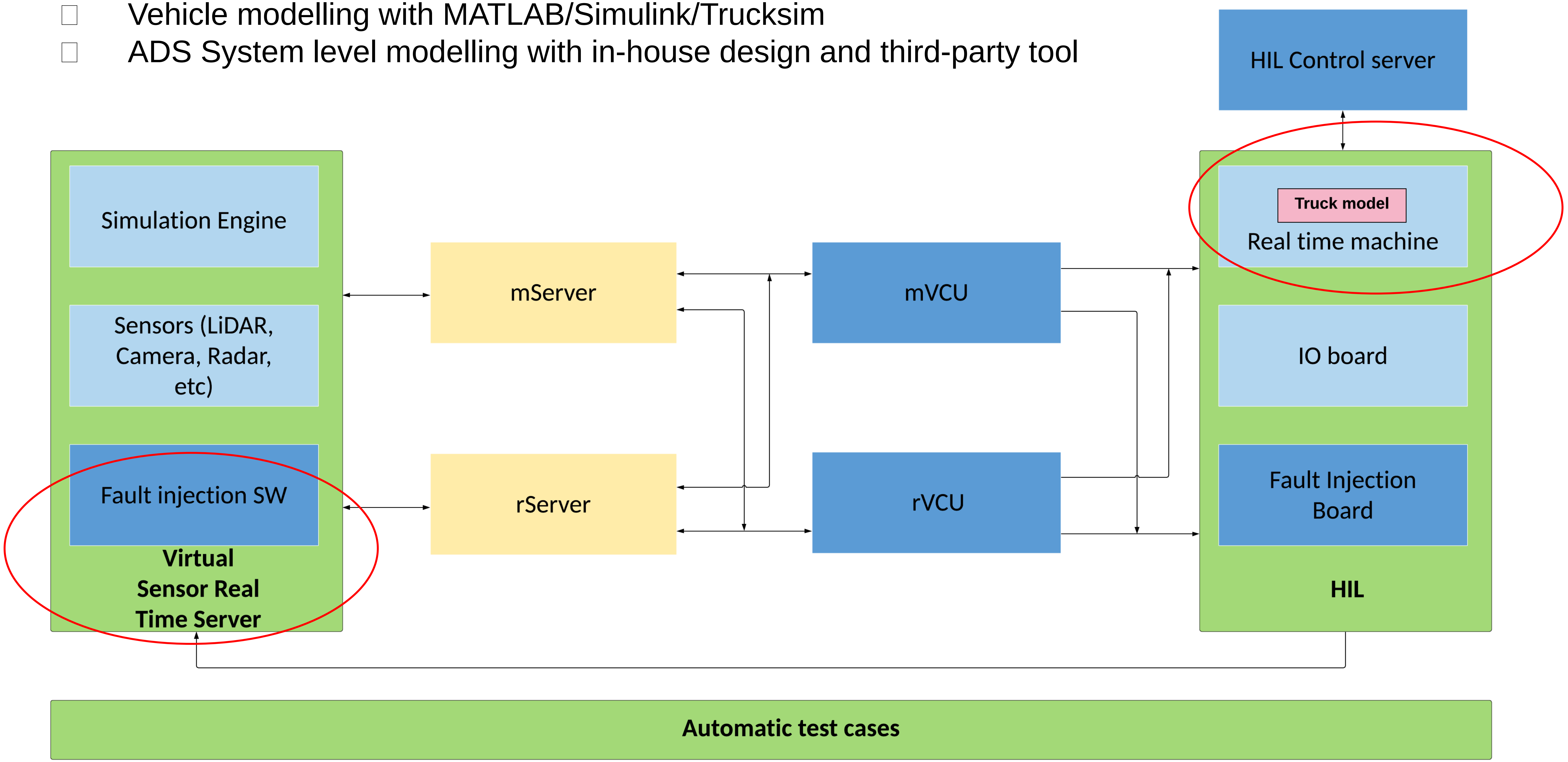
- Simulink is a powerful tool to design the VCU control algorithm
- Model Reference helps the team to collaborate together

## Version and variants control

- MATLAB is a powerful tool to control the variants by customized scripts (we have many versions of vehicle platform with need to have different calibration, software, and configurations)
- MATLAB works well with Github and it makes the version control and team collaboration easier

## Tools: the end-to-end simulation including virtual sensor is a powerful tool for us to generate thousands of critical test cases to test the control system performance and simulate over 200 possible ADS failures. It helps to catch bugs and can cover more test cases than possible with driving a truck on a road

- Vehicle modelling with MATLAB/Simulink/Trucksim
- ADS System level modelling with in-house design and third-party tool



# Content





18

# Conclusion: it's very efficient to design the brake-by-wire control software for a L4 autonomous truck based on MBD. Great tool chain based on MBD to support the development.

Redundant brake-by-wire control is a critical system for L4 vehicle

MBD is essential and efficient to design the brake-by-wire control SW

- Improve the feature release time from 2 weeks to 72 hours
- With customized dbc message builder, can add 1 message in less than 1 minute (based on Vehicle Network Toolbox)
- Reduce the software variants from 5 to 1

There is a full set of tools based on MBD enabling autonomous truck development (great eco-system)

- Developing (architecture design, control software development, code generation, etc.)
- Testing suite (HIL, MIL, SIL, end to end, etc.)
- Version control
- Data analysis and annotation

tusimple

# THANK YOU!

**We are still on the way to the production. MBD helped us a lot in the POC(Proof of Concept)/Prototype/A sample phase, and there will be more challenges on the way to full autonomy.**
**We believe MBD will help us more on our journey, such as ADS power system control, fail operational / fail safe system development, etc.**
**Open for any comments and suggestions.**