tttech auto
part of NXP

For further information, including
price and availability, contact
products@tttech-auto.com

TTTech Auto AG
Austria - HQ
Phone: +43 1 585 65 38-5000

Germany
Phone: +49 841 88 56 47-0

South Korea
Phone: +81 31 707 8895

China
Phone: +86 189 6476 2518

MotionWise
safety middleware
classic platform

The Software Platform that ensures a safe and deterministic execution environment for mixed-criticality software applications.

- Enables the safe and efficient distribution, execution and communication of mixed-criticality applications within and across SoCs (System on a Chip)
- Comes with a rich toolset and embedded functions
- Can host AUTOSAR Classic Applications
- Is designed for E/E architectures with centralized domain controllers, High Performance Computing (HPC) platforms, and zonal ECUs.
- Facilitates application integration and Basic Software (BSW) configuration through automation
- Designed for highest safety and security (ISO 26262 ASIL D and ISO/SAE 21434)

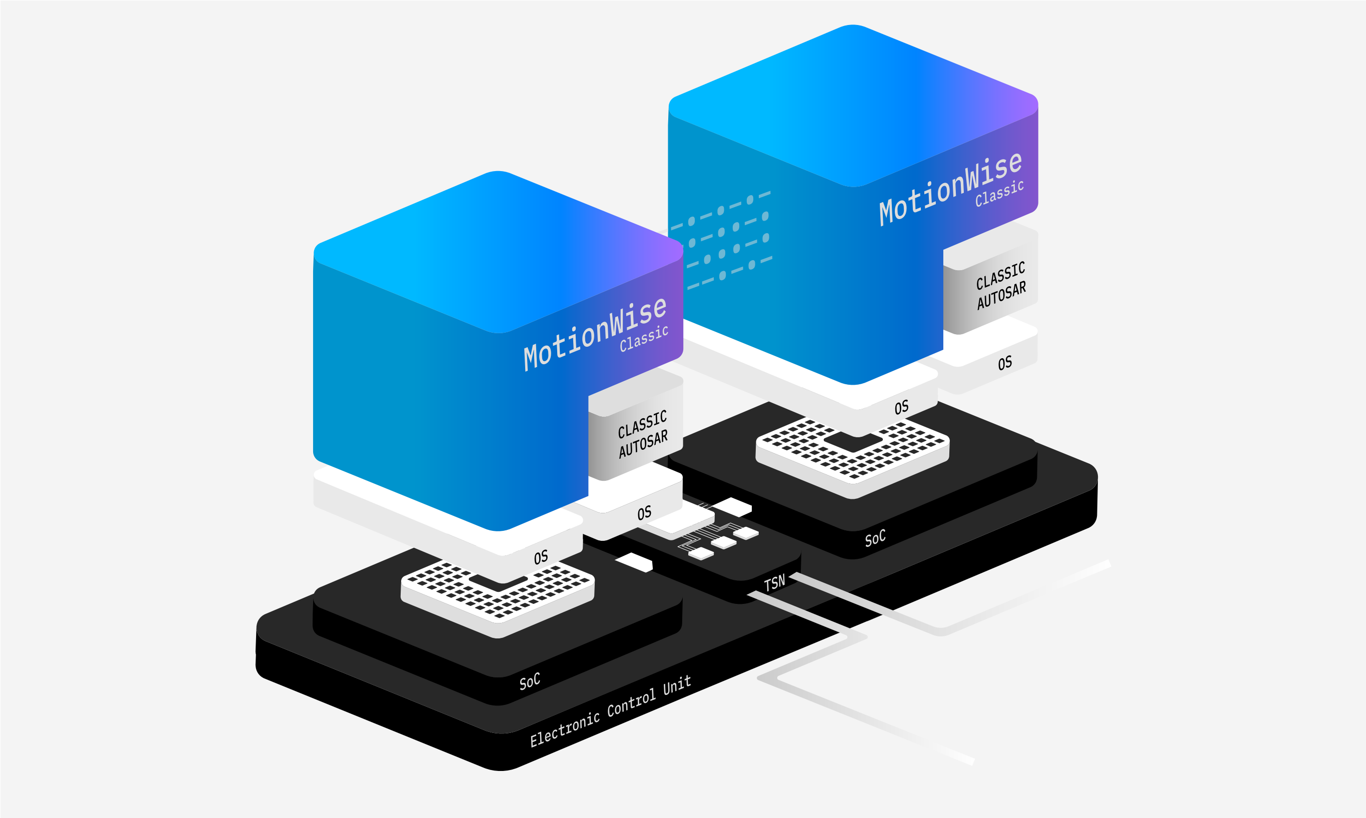

Figure 1 MotionWise deployed in a single ECU system

KEY BENEFITS

Reduce time and cost for software integration

MotionWise Safety Middleware provides tools to ensure the correctness and completness of all system inputs.

Once the system inputs are correct (scheduling attributes, communication matrix etc.) the entire configuration, generation and build process is automated and requires no manual intervention.

The platform can be easily integrated into the customer's CI/CD pipelines running the complete generation and build. In addition to the Platform components, this also includes all other integrated components such as the operating system (OS), communication middlewares and AUTOSAR Standard BSW stacks from established vendors.

Guarantees freedom from interference between mixed-criticality applications

The time-triggered scheduler and supervision mechanism guarantees freedom from interference for applications in the temporal domain.

The TSN protocol guarantees reserved bandwidth in the Ethernet network and no interference from lower criticality applications to safety critical applications.

The memory management system abstracts the underlying OS and hardware mechanisms in order to mediate the applications' access to the memory in a safe manner respecting their criticality level (QM to ASIL D) thus ensuring freedom from interference in the spatial/memory domain.

The “correct-by-design” system enables the early detection of errors and resource issues

With their "correct-by-design" approach, MotionWise scheduling solutions ensure that timing constraints and resource requirements for safety and time-critical workloads are met even in peak load scenarios.

This significantly reduces the effort required in the system verification and validation phase and prevents resource issues from occuring shortly before the start of production.

ECU ARCHITECTURE

The MotionWise Classic Platform supports both multi-SoC and multi-ECU topologies, which include at least one MCU as a safety island and multiple high performance MPUs for hosting complex processing workloads (e.g. sensor fusion, object detection and classification).

Each SoC can have one or more hosts. MotionWise distinguishes between two main types of hosts:

- Safety Host: A standalone microcontroller (MCU) or the safety island of a complex SoC. The Safety Host executes the AUTOSAR Classic software stack and an OSEK operating system. Examples of Safety Hosts are the Infineon Aurix TC39x or the M7 cores of an NXP S32G. A Safety Host contains the MotionWise Classic Platform. MotionWise requires that each ECU has at least one Safety Host.

- Performance Host: The performance CPU cores of a SoC. The Performance Host executes a POSIX-based operating system and provides more computation capabilities than a Safety Host. Examples of Performance Hosts are the A-cores of an NXP S32G or a Texas Instruments TDA4. The Performance Host contains the MotionWise Classic Platform. A single ECU can have 0 or more Performance Hosts.

Figure 2 shows a reference deployment of the in-vehicle stack features of the MotionWise Classic Platform. More details about the provided features can be found in chapter Features – In vehicle SW stack.

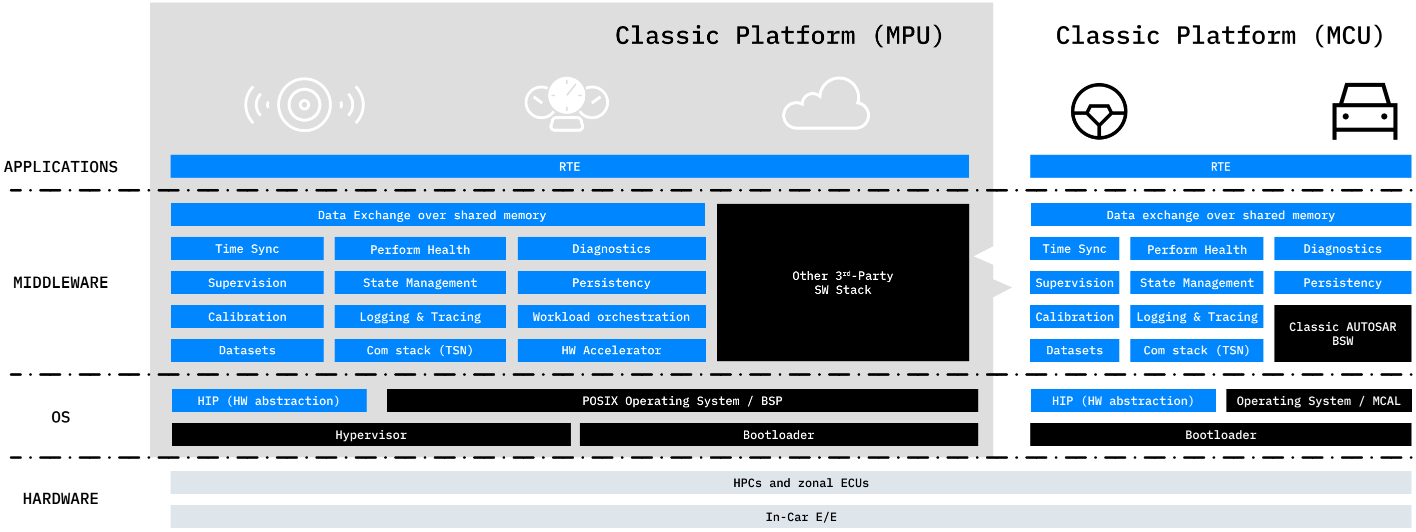

Figure 2 Layered Block diagram of the reference MotionWise Classic Platform

Typical deployment scenarios of Domain Control Units (DCU) and High-Performance Compute (HPC) are

- 1 µc+ 1 µP
- 1 µc+ 2 µPs
- single µP

Communication between the hosts can be enabled via Ethernet, shared memory or both.

FEATURES - IN VEHICLE SW STACK

MotionWise Classic Platform   integrates some parts of MotionWise Schedule product. The following features:

- Global scheduling
- Time-triggered scheduling
- Computation chains
- Task monitoring
- Time synchronization

are described in the MotionWise Schedule Product Brief.

Communication

MotionWise Classic Platform enables safe and deterministic communication between applications running on different platforms on a single ECU or across multiple ECUs.

The communication stack provides the following capabilities and properties:

- standardized communication APIs according to AUTOSAR Classic (RTE),
- high-performance inter-process communication via shared memory on both the MotionWise Classic Platform and the MotionWise SOA Platform. This provides zero-copy implementation, data transparency, data allocation, and a complete memory abstraction, which meets ASIL-D according to ISO 26262
- signal-based communication via standardized protocols over CAN and Ethernet
- service-oriented communication over Ethernet and allows third-party SOME/IP stack integration
- end-to-end communication protection for safety-critical data
- precisely scheduled network traffic, deterministic communication, co-existence of safety-critical and best effort traffic by leveraging the TSN capabilities

Safety Monitoring and Management

The safety mechanisms monitor the states of applications, hosts and SoCs. All safety mechanisms of the platform are qualified to ASIL-D according to ISO26262. The following mechanisms are the building blocks of the safe system:

- Host supervision - checks the health status of all hosts on a MotionWise ECU
- Computation chain monitoring - supervises all computation chains in the system, including chains that span multiple hosts or SoCs
- Health management
    - central error reporting mechanism available to the entire platform
    - initiates user-defined reactions to execution errors that occur during runtime
    - keeps system and hardware resources available for critical applications
    - controls the transition to a degraded mode.

System State Management

MotionWise Classic Platform can manage the complete system lifecycle from ECU startup to ECU shutdown, to ensure stable and consistent host and ECU states and state transitions.

The platform provides a default state machine with predefined states required by the MotionWise classic platform. The ECU state machine provides extensive customization options, such as custom ECU states or transition events.

Information about the state of system and host states, as well as active platform functions and safety mechanisms are made available, to all hosted applications via the available MotionWise APIs.

Startup Configuration Framework

The Startup Configuration Framework allows the initialization of individual platform components as and customer applications via configuration files. The configuration files can be updated without the need for a complete software update.

This allows the application behavior to be re-configured and re-calibrated during startup, for example by changing end- of-line parameters and system/ECU variants.

The startup configuration files can be uploaded via normal, remote, or over-the-air software update mechanisms.

Development Mode Switch

MotionWise Classic Platform allows the system to be executed in either in a debugging mode or in a series production mode, even after the platform software has been compiled. Depending on the execution mode, certain development features are activated, and safety features are deactivated.

The exact test or debug features that are activated/deactivated can be configured. MotionWise provides an API for applications to retrieve the current mode.

Persistent Data Storage and Management

The MotionWise Safety Host offers a solution for writing data to the host’s Non-volatile Memory (NvM). This data is safely and securely stored between drive cycles, even in the event of host reset or power loss.

The persistent storage of data allows for internal consistency checks, wear leveling, and retention support. Through flash abstraction layers and file systems, MotionWise abstracts all communication between applications and the underlying storage mechanism.

XCP Calibration

The MotionWise Safety Host offers a solution for online reconfiguration of the runtime behavior of the in-vehicle stack via Ethernet and CAN. A development PC can have read/write access to the contents of the MotionWise application API memory space during ECU runtime.

Data Collection and Logging

The MotionWise Safety Host provides an API for all hosted applications to log debugging messages both in the development phase and in a series production contexts. This API can be called up to the ASIL-D context. The log messages can be mirrored to a development PC via Ethernet or UART.
Profiling

MotionWise Classic Platform provides two options for system profiling: a MotionWise-specific tool chain (only available in MotionWise Safety Host), and integration with Gliwa T1 libraries. Profiling is possible for all MotionWise Classic Platform hosts, applications, and runnables. It can be enabled and disabled on a per-entity basis.

Profiling

MotionWise Classic Platform provides two options for system profiling: a MotionWise-specific tool chain (only available in MotionWise Safety Host), and integration with Gliwa T1 libraries. Profiling is possible for all MotionWise Classic Platform hosts, applications, and runnables. It can be enabled and disabled on a per-entity basis.

TOOLS

The MotionWise toolset consists of software applications used to configure, generate and test the behavior of the in-vehicle stack and to assist application SW developers in creating and testing their applications.

The MotionWise toolset includes:

1. MotionWise Creator
2. Software Development Kit (SDK)

MotionWise Creator

MotionWise Creator is a powerful tool that simplifies software integration for SDVs by automating configuration processes, reducing cognitive load on integrators, and enabling faster incremental release cycles.

By providing early validation, efficient automation, and a streamlined workflow, it minimizes errors and accelerates development. Its flexibility and cloud-based infrastructure ensure that integrators can adapt to evolving use cases and market demands, making MotionWise Creator an essential solution for modern software-defined vehicles. It consists of frontend tools (client) and a cloud-based backend. MotionWise Creator is a headless application that can be connected to any CI/CD pipeline(s) via its command-line interface.

Additionally, it offers a graphical user interface (GUI) for user-friendly configuration tasks, such as scheduling applications and creating computation chains. It can also integrate with an IDE, allowing for seamless development and debugging in a familiar environment.

The MotionWise Creator user can customize the error response and reporting behavior of the MotionWise in-vehicle stack and enables the user to configure various MotionWise functionalities. MotionWise Creator also validates the customer's system model against the MotionWise requirements provided in the System Definition Guide and provides early and comprehensive feedback of the system model including alerts for deviations from the modeling methodology and guidelines, as well as errors in the model that may result in inappropriate system behavior.
Once the system definition inputs are semantically correct and match the selected HW platform, no manual intervention is required. The complete generation and build process is automated in the MotionWise Creator back-end, which runs in the cloud.

The provided automation framework supports the configuration of integrated standard SW components or stacks (e.g. Classic AUTOSAR BSW or a SOME IP stack) from selected established vendors. The framework is designed to support seamless extensions tailored to the specific requirements of the system

MotionWise Software Development Kit (SDK)

MotionWise SDK is a toolset that supports the application developer in developing and testing applications for the Safety Host directly on a PC without the need for the MotionWise ECU.

MotionWise SDK includes:

1. Software in the Loop
The Software in the Loop environment enables the development and testing of applications on a PC. Platform communication interfaces between software components are simulated. Test vectors can be injected to simulate the independent behavior of the application at runtime without the need for the HW platform. This allows the same application code to be compiled, linked and executed on both the PC and the physical ECU.

2. Remote access tools
Remote access tools allow the user to monitor and re-configure network traffic (RTE) on hosts executing the MotionWise Classic Platform. The user can perform standard tasks such as logging, tracing, and configuring the routing table.

Supported third-party components

MotionWise Classic Platform supports a variety of widely adopted third party components.

It integrates well with standard automotive software stacks, as shown in the table below.

COMPATIBILITY	MICROCONTROLLER / MCU	MICROPROCESSOR / SOC
AUTOSAR COMPATIBILITY	Vector MICROSAR Classic stack	non-AUTOSAR based POSIX systems
OPERATING SYSTEM	Vector MICROSAR OS	QNX OS 7.1 or QNX OS 8.0* or Linux
SEMICONDUCTORS	▪ ARM Cortex M7 (NXP S32G2 / G3) ▪ R5F (e.g. TI TDA4) ▪ TriCore (e.g.: TC39x)	ARMv8 and x86-64, e.g.: ▪ ARM Cortex A53/A55/A72 (e.g. S32G, HR J5, TDA4) ▪ Kryo Gen6 (e.g. Qualcomm SA8xxx)
PROFILING	▪ GLIWA T1	▪ GLIWA T1

* *The features, marked with the asterisk are planned for 2025*