Train Screenboard/Driver Display
Managing the complexity ahead

PikeOS®: the proven platform for a Safe & Secure Operation

- Graphical UI and real-time
- Allow mixed-criticality
- Safe GPU sharing

EN 50128 SIL 3/4 • IEC 61508 SIL 3 • Common Criteria EAL3+ • Trusted by leading OEMs & Tier-1s • Quality „Made in Germany“

www.sysgo.com
Train Screenboard/Driver Display
Managing the complexity ahead

Challenge

Driver display systems (or screenboards) are becoming more and more complex, while integrating highly critical with commodity functions. This trend will continue, leading to configurable multi-function displays (MFDs), as already seen in avionic cockpits. This requires the possibility to run safety critical UI components, which are to be certified, and in parallel enable feature rich non-critical commodity UI. The later requires the use of standard components, while focusing on a sophisticated UI representation (e.g. through 2D/3D acceleration and rendering through HW). Safety critical UI components will focus on a leaner approach in order to reduce risks and cost of the certification while using a pre-certified UI framework.

Solution

In order to have a cost-effective commodity UI implantation, the feature richness of today’s available UI frameworks have to be available. And the coexistence of UIs of different criticality requires a strict separation of functionality, while being able to share the same graphic HW.

• The pre-certified PikeOS RTOS/Hypervisor allows you to use commodity functionality from an embedded Linux (e.g. HTML5) side-by-side with a pre-certified UI framework.
• The UI framework can share the GPU, while access to the GPU is controlled and managed by PikeOS.
• While implementing a feature-rich UI, PikeOS enables the concurrent execution of real-time tasks on the same platform.

PikeOS Software Architecture

In order to have a cost-effective commodity UI implantation, the feature richness of today’s available UI frameworks have to be available. And the coexistence of UIs of different criticality requires a strict separation of functionality, while being able to share the same graphic HW.

• The pre-certified PikeOS RTOS/Hypervisor allows you to use commodity functionality from an embedded Linux (e.g. HTML5) side-by-side with a pre-certified UI framework.
• The UI framework can share the GPU, while access to the GPU is controlled and managed by PikeOS.
• While implementing a feature-rich UI, PikeOS enables the concurrent execution of real-time tasks on the same platform.

PikeOS Software Architecture

In order to have a cost-effective commodity UI implantation, the feature richness of today’s available UI frameworks have to be available. And the coexistence of UIs of different criticality requires a strict separation of functionality, while being able to share the same graphic HW.

• The pre-certified PikeOS RTOS/Hypervisor allows you to use commodity functionality from an embedded Linux (e.g. HTML5) side-by-side with a pre-certified UI framework.
• The UI framework can share the GPU, while access to the GPU is controlled and managed by PikeOS.
• While implementing a feature-rich UI, PikeOS enables the concurrent execution of real-time tasks on the same platform.

PikeOS Software Architecture

In order to have a cost-effective commodity UI implantation, the feature richness of today’s available UI frameworks have to be available. And the coexistence of UIs of different criticality requires a strict separation of functionality, while being able to share the same graphic HW.

• The pre-certified PikeOS RTOS/Hypervisor allows you to use commodity functionality from an embedded Linux (e.g. HTML5) side-by-side with a pre-certified UI framework.
• The UI framework can share the GPU, while access to the GPU is controlled and managed by PikeOS.
• While implementing a feature-rich UI, PikeOS enables the concurrent execution of real-time tasks on the same platform.

PikeOS Software Architecture

In order to have a cost-effective commodity UI implantation, the feature richness of today’s available UI frameworks have to be available. And the coexistence of UIs of different criticality requires a strict separation of functionality, while being able to share the same graphic HW.

• The pre-certified PikeOS RTOS/Hypervisor allows you to use commodity functionality from an embedded Linux (e.g. HTML5) side-by-side with a pre-certified UI framework.
• The UI framework can share the GPU, while access to the GPU is controlled and managed by PikeOS.
• While implementing a feature-rich UI, PikeOS enables the concurrent execution of real-time tasks on the same platform.

PikeOS Software Architecture

In order to have a cost-effective commodity UI implantation, the feature richness of today’s available UI frameworks have to be available. And the coexistence of UIs of different criticality requires a strict separation of functionality, while being able to share the same graphic HW.

• The pre-certified PikeOS RTOS/Hypervisor allows you to use commodity functionality from an embedded Linux (e.g. HTML5) side-by-side with a pre-certified UI framework.
• The UI framework can share the GPU, while access to the GPU is controlled and managed by PikeOS.
• While implementing a feature-rich UI, PikeOS enables the concurrent execution of real-time tasks on the same platform.