Overview

Freescale OSEKturbo OS is the leading OSEK/VDX™ Real Time Operating System (RTOS) implementation available on the market today in terms of quality and performance. OSEKturbo fully conforms to the latest OSEK/VDX Operating System standard. The RTOS is fully configurable and statically defined within the CodeWarrior™ IDE environment. Only those functions that are required are included providing the user with a truly scaleable RTOS.

OSEKturbo OS is available for a wide range of 8-, 16- and 32-bit embedded microprocessors. The emphasis in the design of OSEKturbo OS was to meet demanding performance requirements, while maintaining the outstanding reputation for stability and quality (SEI CMM level 5) established in more than five years of automotive ECU development. The result is that Freescale OSEKturbo is one of the most widely used OSEK/VDX OS implementations on the market today. Although the OSEK/VDX specification was created by the automotive industry, the specification does not limit its use to automotive applications and it is increasingly being used in a wide range of embedded applications ranging from industrial control to telecom products.

Benefits

By adopting Freescale OSEKturbo and CodeWarrior Development Studio, embedded developers can:

> Reduce software integration time and the associated costs
> Re-use software modules among different application designs
> Reduce maintenance costs through the use of a high-level language and common platform
> Ensure a high-quality, stable software environment for their applications
> Ensure a highly efficient OS with extremely small memory footprint and outstanding performance
> Ensure an optimized, highly integrated and broadly supported development tool chain

OSEKturbo applications are written in C language using the OSEK Application Programming Interface (API) enabling portability and re-use of software.

Freescale OSEK tool support

Freescale offers a Builder tool for easy configuration of OSEKturbo applications. The Freescale OSEK Builder is a graphical Windows™ based application that enables the developer to easily configure a runtime system. The tool allows the developer to optimize the software to meet the specific application needs. A complete system description is generated in OSEK Implementation Language (OIL) format, allowing full portability of the application configuration. The developer may also choose to generate OS libraries from Builder.

Freescale also offers an OSEK plug-in to allow kernel-aware debugging in CodeWarrior Development Studio, enabling the developers to use the CodeWarrior IDE for the whole application development. It brings many advantages such as simplified project setup through the use of stationaries and samples. CodeWarrior tools support Freescale’s 8- to 32-bit microcontrollers reducing the number of tools that need to be re-learned for new projects on different architectures. OSEK System Generator (SysGen) is fully integrated in the build process and for the configuration of project settings. OSEK Builder can also be called as OSEK configuration editor. CodeWarrior Development Studio supports full OSEK OS kernel awareness during debugging. CodeWarrior Development Studio can also provide simulation support for development without hardware.

In addition Freescale has established strong partnerships with industry leaders providing deeper integration with a wide variety of partner tools including Model Based Simulation, Code Generation, Schedulability Analysis, popular CAN-based communications software and tools, and support for many 3rd Party OSEK kernel aware debuggers (emulators) and compilers.
OSEKturbo Performance Information

- Measurements done with Timer capture (MPC555: TB, time base register; HCS12DP256: TCNT, timer count register)
- Tasks activated consecutively, ISR that activates tasks as well
- 19 tasks, 3 ISRs, STANDARD status

OSEKturbo Performance Information

<table>
<thead>
<tr>
<th></th>
<th>MPC555 40MHz; BCC1</th>
<th>HCS12DP256 8MHz; BCC1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context switch between tasks (ActiveTask)</td>
<td>1.7µs</td>
<td>12.5µs</td>
</tr>
<tr>
<td>Terminate task with return to background task</td>
<td>0.9µs</td>
<td>6.6µs</td>
</tr>
<tr>
<td>Context switch from ISR (ActiveTask)</td>
<td>2.1µs</td>
<td>14.9µs</td>
</tr>
</tbody>
</table>

Typical Memory Requirements:

<table>
<thead>
<tr>
<th></th>
<th>MPC555</th>
<th>HCS12DP256</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total OS ROM</td>
<td>2304 bytes</td>
<td>776 bytes</td>
</tr>
<tr>
<td>OS data</td>
<td>108 bytes</td>
<td>60 bytes</td>
</tr>
<tr>
<td>OS code</td>
<td>2196 bytes</td>
<td>716 bytes</td>
</tr>
<tr>
<td>Total OS RAM</td>
<td>466 bytes</td>
<td>90 bytes</td>
</tr>
</tbody>
</table>

Freescale OSEKturbo OS Products

OSEKturbo releases exist for all OS (BCC1, ECC1, BCC2, ECC2) and COM (CCCA and CCB) conformance classes.

Supported Processors (check with your local Freescale sales office for latest list)
- Freescale 68HC08
- Freescale 68HC(S)12
- Freescale MPC5xx
- Freescale MPP5200
- Freescale DSP56800E
- Freescale S12X
- Freescale MPC55xx
- Freescale MAC7100

Technical Enhancements
- Timescale feature that enables cyclic task activations based on efficient alarms
- OSEKturbo highly optimized for speed and memory on each target
- ORTI (OSEK Runtime Interface) for kernel aware debugging
- Enhanced OSEK Builder for OIL configuration
- Multiple scheduling policies
- Event control for task synchronization
- Resource management based on OSEK priority ceiling protocol
- Counter management
- Efficient alarm management
- Optimal stack methods
- Stack monitoring
- Fast links for accessing system data structures
- Automatic exclusion of unused system services
- Specific hardware features such as memory bank switching and use of low power modes, interrupt handler and Floating Point registers
- Full internal OSEK communication for inter process communication
- Task management for activation and termination of tasks
- Interrupt management
- Error handling

Benchmark configuration:
- BCC1 with one task per priority
- Pre-emptive Scheduling
- No multiple activations
- 10 basic tasks + 1 initialization task
- 10 alarms with task activations
- 1 16-bit (1 ms)
- 1 ISR (in addition to system timer)
- No messages, no resources
- Standard status, debug level zero
- No hooks

Learn More: For more information about Freescale products, please visit www.freescale.com/codewarrior