

A GLOSSARY

Application Programming Interface (API)

A library of C/C++ functions and assembly macros that define VDK services. These services are essential for kernel-based application programs. The services include interrupt handling, thread management, and semaphore management, among other services.

Context Switch

A process of saving/restoring the processor's state. The scheduler performs the context switch in response to the system change.

A hardware interrupt can occur and change the state of the system at any time. Once the processor's state has changed, the currently running thread may be swapped with a higher priority thread. When the kernel switches threads, the entire processor's state is saved and the processor's state for the thread being switched in is restored. This process is known as a context switch.

Critical Region

A sequence of instructions, execution of which cannot be interrupted or swapped out. To ensure that the execution of a critical region is uninterrupted, all interrupt service routines must be suspended before calling the critical region. Once the critical region routine has been completed, ISRs are enabled.

Event

A signal, similar to a semaphore, used to synchronize multiple threads in a system. An event is a logical switch, having two binary states (available/true and unavailable/false) that control thread execution. When an event becomes available, all pending (waiting) threads in the wait list are set to be ready-to-run. When an event is available and a thread pends on it, the thread continues running and the event remains available.

For error handling purposes, threads can specify a timeout period when pending on an event. A certain time period can be set for posting an event at a given time, thus allowing threads to be made ready to run periodically.

An event is a code object of global scope, so any thread can pend on any event. Event properties include the EventBit mask, EventBit value, and combination type. Events are statically allocated and enumerated at run-time. An event cannot be destroyed, but its properties can be changed (see [LoadEvent\(\)](#)).

EventBit

A flag set or cleared to post the event. The Event is posted (available) when the current values of the system EventBits match the EventBit's mask and EventBits' values defined by the Event's combination type.

There is one and only one EventBits word in a system. It matches the width of the processor's data bus: sixteen bits for ADSP-219x DSPs; thirty-two bits for ADSP-21xxx, and ADSP-TSxxx DSPs.

Device Driver

A user-written routine that abstracts the hardware implementation from the application code. User code accesses device drivers through a set of Device Driver APIs.

Interrupt

An external or internal condition detected by the hardware interrupt controller. In response to an interrupt, the kernel processes a subroutine call to a predefined Interrupt Service Routine (ISR). Interrupts have the following specifications:

- **Latency** — interrupt disable time—the period between the interrupt occurrence and the first ISR's executed instruction.
- **Response** — interrupt response time — the period between the interrupt occurrence and a context switch.
- **Recovery** — interrupt recovery time — the period needed to restore the processor's context and to start the return-from-interrupt (RTI) routine.

Interrupt Service Routine (ISR)

A routine executed as a response to a software interrupt or hardware interrupt. VDK supports nested interrupts, meaning that the kernel recognizes other interrupts and/or services interrupts with higher priorities while executing the current ISR. VDK ISRs are written in assembly language. VDK reserves the timer and the lowest priority (idle) interrupt.

Real-Time Operating System (RTOS)

A software executive that handles DSP algorithms, peripherals, and control logic. The first release of Analog Devices RTOS comprises the following components: kernel, communication manager, support library, and device drivers. RTOS enables structured, scalable, and expandable DSP application development while hiding OS complexity.

Round-Robin Scheduling

A scheduling scheme where all threads at a given priority are given processor time automatically in fixed duration intervals. Round-Robin priorities are specified at build time.

Pre-emptive Kernel

VisualDSP++ Kernel is a priority-based kernel in which the currently running thread of the highest priority is pre-empted, or suspended, to give system resources to the new highest priority thread.

RTOS Kernel

The main module of any Real-Time Operating System. The kernel loads first and permanently resides in the main memory. Typical kernel manages other modules of the Real-Time Operation System. Kernel typical services include context switching and communication management between OS modules.

Scheduler

A kernel component responsible for scheduling of system threads and interrupt service routines. VDK is a priority-based kernel in which the highest priority thread is executed first.

Semaphore

A signal (similar to an event) used to synchronize multiple threads in a system. A semaphore is a logical switch, two states of which (available/true and unavailable/false) control any thread execution. Unlike an event, whose state is automatically calculated, a semaphore is directly manipulated via `PendSemaphore()` or `PostSemaphore()`. Posting a semaphore takes a deterministic amount of time and may incur a context switch.

Signal

A method of communicating between multiple threads. VDK supports three types of signals: semaphores, events, and device flags.

System Configurator

The System Configuration control is accessible from the **Kernel** tab on the **Project Window**. The **Kernel Window** provides a graphical representation of the data contained in the `Vdk.h` and `Vdk.cpp` files.

Thread

A kernel system component that performs a predetermined function and has its own share of system resources. VDK supports multithreading, a run-time environment with concurrently executed independent threads.

Threads are dynamic objects that can be created and destroyed at runtime. Thread objects can be implemented in C, C++, or assembly language. A thread's properties include an ID, priority, and current state (wait, ready, run, or interrupted). Each thread maintains its own C/C++ stack.

Unscheduled Region

A sequence of instructions whose execution can be interrupted, but cannot be swapped out. The kernel acknowledges and services interrupts when an unscheduled region routine is running.

VisualDSP++ Kernel (VDK)

RTOS kernel from Analog Devices. VDK is a part of VisualDSP++. The kernel is integrated with the Integrated Development and Debugging Environment (IDDE) and assembler, compiler, and linker programs into the DSP development tool chain.

VDK is supported on ADSP-219x, ADSP-21xxx, and ADSP-TSxxx processors.

Ticks

The system level timing mechanism. Every system tick is a timer interrupt.