

RTA-OS

NXP S32K ARM7V with the Green Hills Compiler

Port Data Sheet

RTA-OS is the ETAS Classic AUTOSAR OS implementation. RTA-OS supports a wide variety of microcontroller/compiler combinations (RTA-OS ports). This port data sheet describes the support for the NXP S32KV ARM7V (S32K2xx) with the Green Hills compiler

Supported Devices

RTA-OS supports the Cortex-M7 cores on the following variants of the NXP S32K microcontrollers:

- S32K2xx

Toolchain support

This port supports the following compilers:

- Green Hills v2018.1.4

Interrupt model

On the S32KARMV7/GHS port, RTA-OS supports 15 levels of Category 1 and Category 2 ISRs, plus two fixed interrupts (HardFault and NMI) and user level.

Memory model

On the S32KARMV7/GHS port, RTA-OS uses the standard flat memory model, following standard EABI.

Memory overhead of RTA-OS

Object	RAM (bytes)	ROM (bytes)
Task	0	20
Cat 2 ISR	0	8
Resource	4	8
Alarm	12	2
Counter	4	20
Schedule Table	16	16
Expiry Point	0	4

Performance

The following gives the key RTA-OS kernel performance data measured in CPU cycles.

Action	Exec time	Ref
Pre-emption	73	A
Normal Termination	25	B
Task Switch	40	C
ChainTask	121	D
WaitEvent	256	E
SetEvent	287	F
Schedule	69	G
ReleaseResource	67	H
Cat 2 ISR Entry Latency	127	I
Cat 2 ISR Exit Latency – interrupted task	104	J
Cat 2 ISR Exit Latency – task switch	73	K
Cat 1 ISR Latency	39	L

RTA-OS

NXP S32K ARMv7 with the Green Hills Compiler

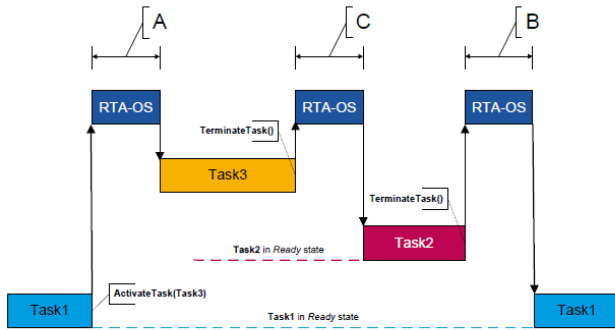


Figure 1 - Task1 is preempted by Task3, followed by a task switch and then normal termination of Task2

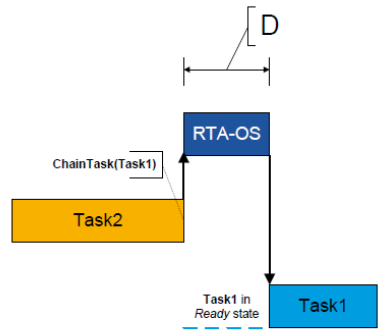


Figure 2 - Task2 chains Task1

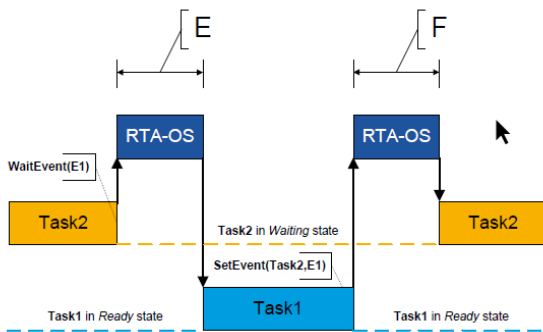


Figure 3 - Task2 waits for an event set by Task1

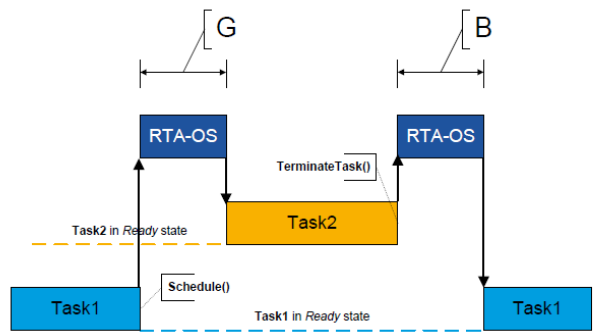


Figure 4 - Task1 allows cooperative scheduling by Task2

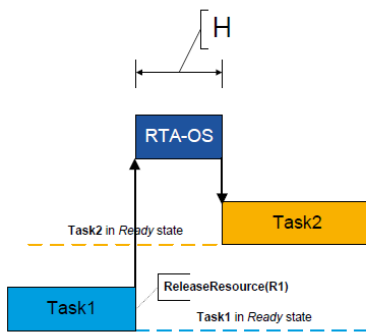


Figure 5 - Task1 releases a resource

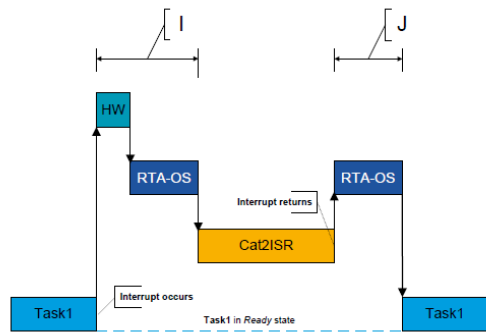


Figure 6 - Category2 ISR entry and exit latency

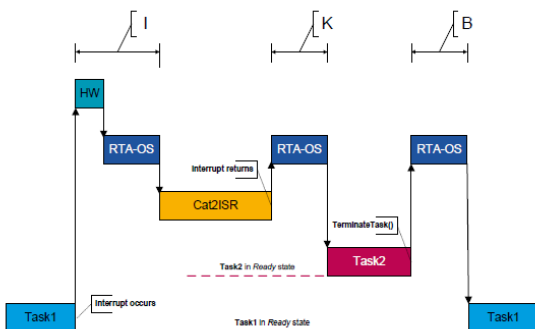


Figure 7 - Category2 ISR switches to Task2

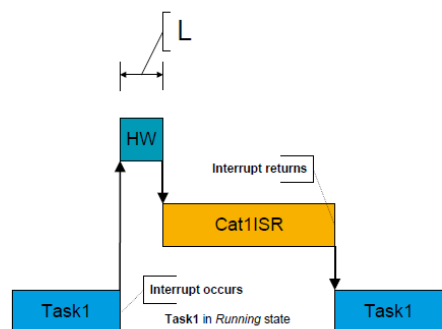


Figure 8 - Category1 ISR entry latency