



Documentation

In order to fulfill the requirements of a real-time system, a system must react to an external event like an interrupt within a defined time frame. Therefore several mechanisms, configurations and implementation rules have to be considered.

Technical Basics

Currently running tasks with low priority must be preempted to allow the real-time critical task to run. Preemption depends on the rules of the task's scheduling policy. Another important aspect in a real-time system is the guaranteeing the exclusive use of certain resources for real-time tasks.

[Read more about technical basics](#)

Technical details of PREEMPT_RT patch

The main aim of the PREEMPT_RT patch is to *minimize the amount of kernel code that is non-preemptible*¹⁾. Therefore several substitution mechanisms and new mechanisms are implemented.

[Read more about technical details](#)

HOWTOs

There are three categories of HOWTOs:

- [RTOS and RT application](#)
- [RT Debugging](#)
- [RT tools and utilities](#)

Publications

Several real-time related publications can help broaden your knowledge about real-time systems.

[Read more about publications](#)

Known Limitations

Some functionality is not available on RT because its implementation is not compatible with RT. Those limitations may apply to the kernel as a whole or may apply to specific architecture or platforms.

[Read more about known limitations](#)

¹⁾ Paul McKenney. A realtime preemption overview. <https://lwn.net/Articles/146861/> [<https://lwn.net/Articles/146861/>]