



Real Time Linux

Walter Fetter Lages

w.fetter@ieee.org

Universidade Federal do Rio Grande do Sul

Escola de Engenharia

Departamento de Engenharia Elétrica

ENG04008 Sistemas de Tempo Real



Introdução

- Linux é um kernel de sistema operacional de uso geral
 - Otimizado para desempenho médio para todos os processos
 - Resolução do timer
 - 100Hz no 80x86
 - 100Hz, 250Hz ou 1000Hz a partir da versão 2.6.x
 - 1024Hz no Alpha
 - Escalonamento não tempo real
- Real Time Linux
 - Adaptação do Linux para tempo-real
 - Alterações na resolução do kernel
 - Alterações no escalonamento



Variantes Real Time Linux

- Hard real time
 - RT-Linux
 - RTAI
- Firm real time
 - KURT
- Soft real time
 - Linux/RK
 - RED-Linux
 - SMART-Linux
 - ART Linux
 - Linux-SRT
 - Qlinux

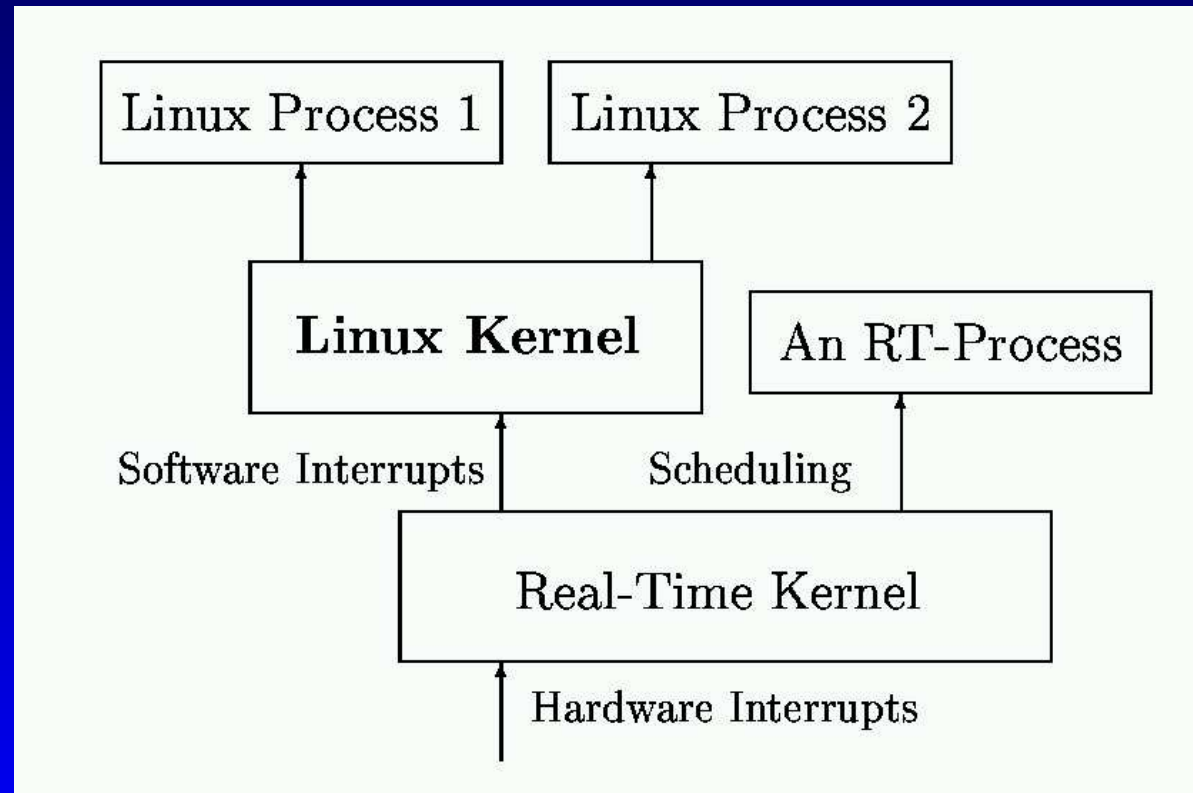


RT-Linux

- Desenvolvido por Victor Yodaiken e Michael Barabanov no NMT
- Executa o kernel do Linux como a tarefa de menor prioridade
 - Substitui os CLIs, STIs e IRETs do kernel padrão do Linux por macros (virtualização)
 - Kernel do Linux completamente preemptível
 - Tratadores de interrupção nunca são atrasados por tarefas não tempo real
 - As últimas versões suportam programação em tempo real ao nível de usuário (LXRT)

RT-Linux

- RT-Linux e tarefas de tempo real são módulos do kernel
- Escalonamento baseado em prioridades





Real Time Application Interface

- Desenvolvido por Paolo Mantegazza no Instituto Politécnico de Milão a partir do RT-Linux
- Utiliza o conceito de HAL e *pipeline* de interrupções
- Mais ênfase a aplicabilidade do sistema do que à índices de desempenhos teóricos
- Suporta programação em tempo real a nível de usuário (LXRT) e ponto flutuante desde as primeiras versões



RTAI

- Chaveamento entre RTAI e kernel padrão
 - Se o RTAI não está *montado* o kernel normal do Linux é utilizado
- Suporta SMP
- Interface estilo POSIX threads
- Diversos escalonadores disponíveis



KURT

- Desenvolvido por Douglas Niehaus na Universidade do Kansas
- Aumenta a resolução do timer do Linux utilizando UTIMER
 - 8254 programado para gerar interrupção no próximo instante de escalonamento (50us antes) ou no próximo jiffie
 - TSC e spin loop para obter maior resolução
- Escalonamento planejado
- Os serviços do kernel do Linux não tem temporização conhecida



Tabela de Escalonamento

```
struct rt_event
{
    unsigned long data_1;
    unsigned long data_2;
    unsigned long data_3;
    unsigned long module_id;
    unsigned long tenms;
    unsigned long usec;
}
```