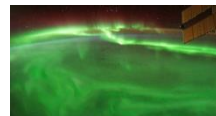
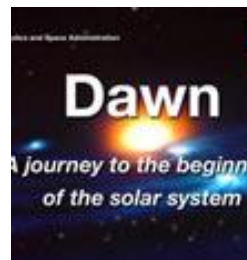


RTEMS介绍

rayx.cn@gmail.com

RTEMS简介

- ❑ 1988年为导弹控制系统开发
- ❑ 1994年开放源代码
- ❑ 2006年水星探测器，火星探测器
- ❑ 2007年火/木星间小行星探测
- ❑ 2008欧洲宇航局Herschel/Plank计划
- ❑ 2009 REXUS 5火箭
- ❑ 2009 RUAG 空间站管理控制单元
- ❑ 2010 THEMIS 卫星
- ❑ 2010 RTEMS RFS文件系统



RTEMS性能简介

- ❑ 功能完整的实时操作系统
- ❑ 性能达到商业级(参考vxWorks对比报告)
- ❑ 完全开放源代码
- ❑ 比GPL更灵活的Licence授权方式
- ❑ 不断增加新功能与新硬件支持，每4个月发布一个新版本
- ❑ 可配置/剪裁
- ❑ 支持多种文件系统：FAT, RFS, RAM, NFS, YAFFS, etc.
- ❑ 高性能，系统行为可预期
- ❑ 方便的开发环境
- ❑ 支持多种CPU
- ❑ 纳秒级时间精度(部分硬件平台)
- ❑ POSIX/ITRON等API
- ❑ 多处理器支持（同构/异构）
- ❑ 开发平台：Linux，FreeBSD, NetBSD, Windows(cygwin, MinGW), MacOS
- ❑ 开发语言：C, C++, Ada, Go, and Java，Lua, Python

支持的处理器1

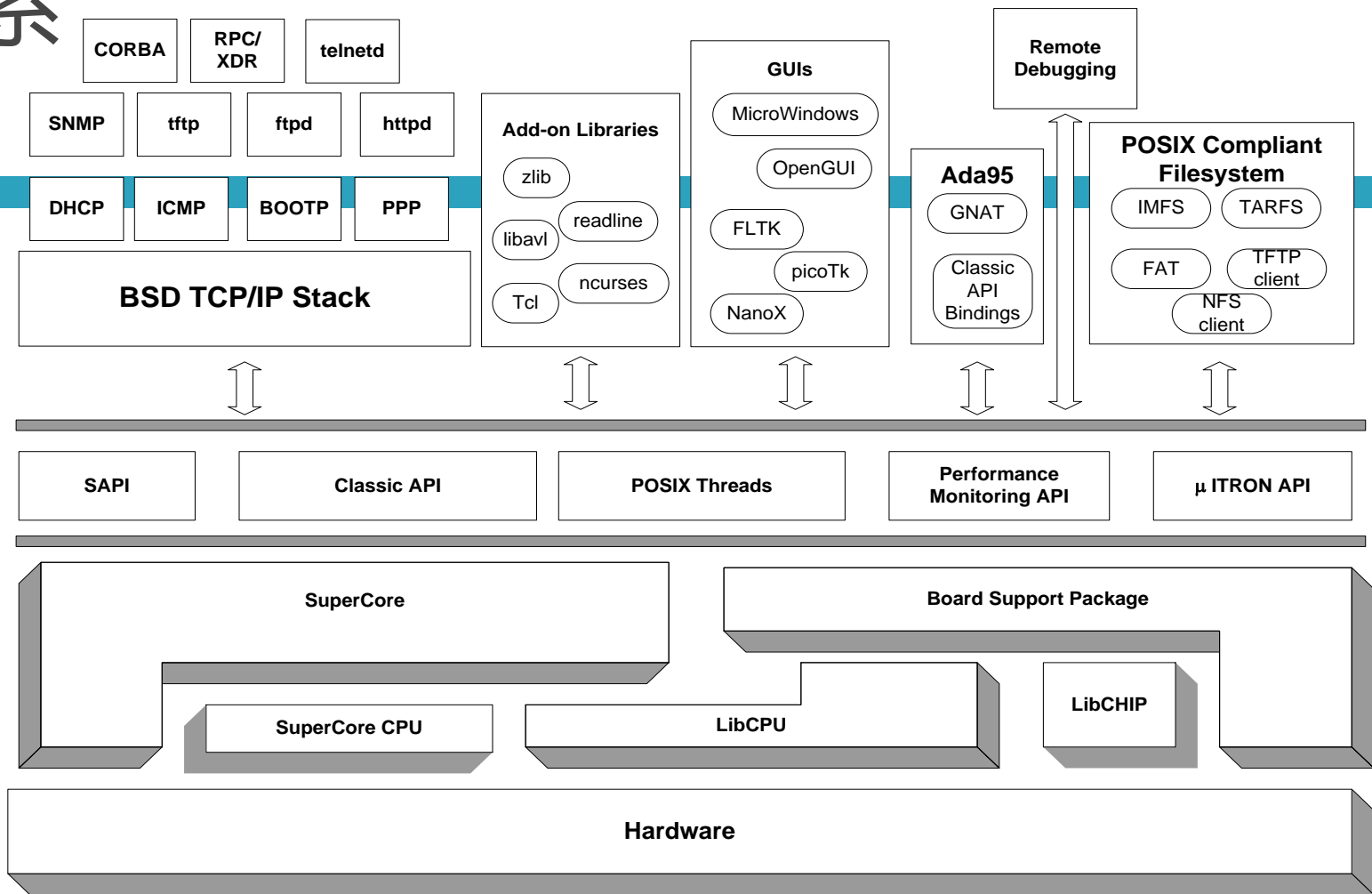
Architecture	Target CPU	4.6 Branch	4.7 Branch	4.8 Branch	4.9 Branch	4.10 Branch	CVS Head
Altera NIOS II	nios	No	No	Yes	Yes	Yes	Yes
ADI Blackfin	bfin	No	No	Yes	Yes	Yes	Yes
ARM with many CPU models	arm	Yes	Yes	Yes	Yes	Yes	Yes
ARM Thumb	arm	No	No	No	Yes	Yes	Yes
Atmel AVR	avr	No	Stub	Stub	Stub	No Interrupts	No Interrupts
AMD A29K	a29k	Yes	No	No	No	No	No
HP PA-RISC	hppa1.1	Yes	No	No	No	No	No
Intel/AMD x86	i386	Yes	Yes	Yes	Yes	Yes	Yes
Intel i960	i960	Yes	No	No	No	No	No
Lattice Mico32	lm32	No	No	No	No	Yes	Yes

支持处理器2

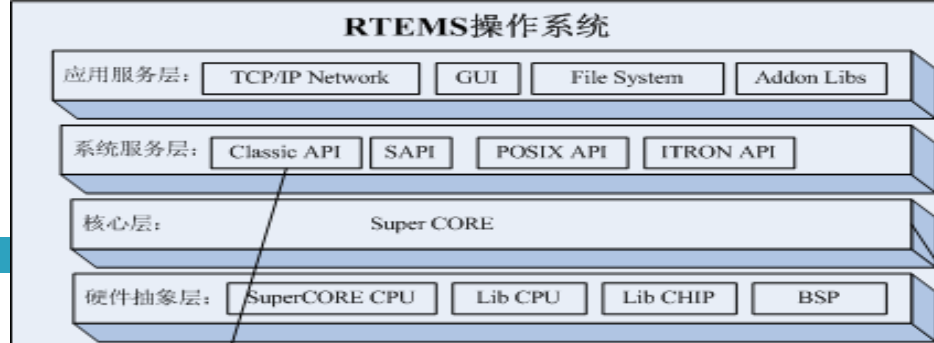
MIPS R3000 and R4000	mips	Yes	Yes	Yes	Yes	Yes	Yes
Motorola MC68xxx	m68k	Yes	Yes	Yes	Yes	Yes	Yes
Motorola MC683xx	m68k	Yes	Yes	Yes	Yes	Yes	Yes
Motorola Coldfire	m68k	Yes	Yes	Yes	Yes	Yes	Yes
OpenCores OR32	or32	Yes	No	No	No	No	No
PowerPC including 4xx, 5xx, 6xx, 7xx, 8xx, and 74xx	powerpc	Yes	Yes	Yes	Yes	Yes	Yes

Renesas H8/300	h8300	Yes	Yes	Yes	Yes	Yes	Yes
Renesas M32C	m32c	No	No	No	No	No Interrupts	No Interrupts
Renesas M32R	m32r	No	No	No	No	No Interrupts	No Interrupts
Renesas SH including SH1, SH2, SH3 and SH4	sh	Yes	Yes	Yes	Yes	Yes	Yes
SPARC	sparc	Yes	Yes	Yes	Yes	Yes	Yes
SPARC V9	sparc64	No	No	No	No	No	Yes
Texas Instruments C3x/C4x	tic4x	Yes	No	No	No	Yes	No

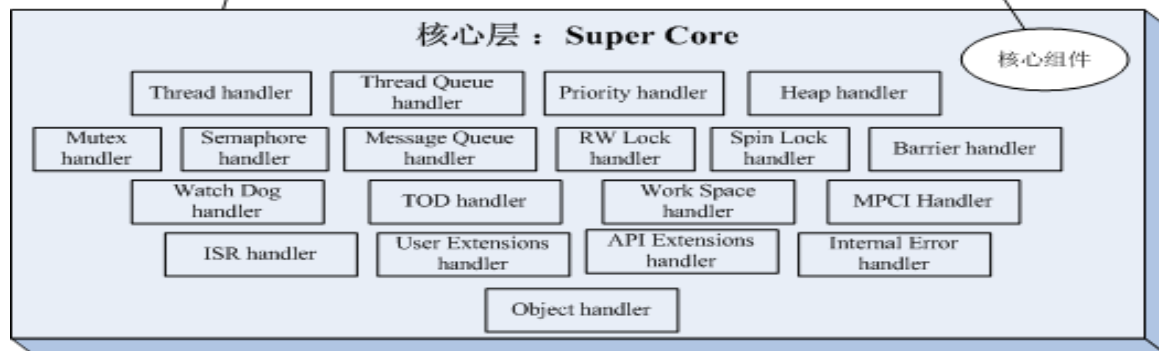
RTEMS系 统结构



RTEMS操作系统



核心层：Super Core



系统服务层：Classic API



版本更新

□ 4.8 (2007)

- 修改任务优先级后，优先级配置立刻生效(优先级队列改变)
- 纳秒时间戳 64bit 以及基于纳秒级别调度
- 代码可配置性。最小配置Kernel+执行程序elf从34K缩减到9K
- 优化启动时间
- 增加任务遍历API
- 增加NIOsII , Blackfin 支持
- Skyeye支持
- POSIX Barriers, Spinlock , Rwlock
- API增加 , 新的BSP....

□ 4.9 (2008 ~ 2011)

- 增加 API
- 改善系统可配置性
- 开发工具更新
- RTEMS SHELL脚本
 - 支持cpu, stack, memory workspace, ipconfig , route, ls, rm cp, mv, ln, mount etc
- Flash filesystem
- 网络优化
- 新的BSP
- 可以选择LWIP来取代BSD协议栈 , 用于小的嵌入式 设备

版本更新

□ 4.10 (2011 ~)

- BuildTool 更新
- 规范化ISR流程与接口
- 文件rename, state等操作放入标准API
- Mount 层次化。有利于支持更多文件系统
- POSIX pipe, getrusage
- Shell 支持login 用户鉴权, fdisk, 减少内存使用
- SuperCore时间戳
- 优先级数目可配置16~256
- RTEMS 文件系统RFS发布
- SIMD支持: SSE, AltiVec
- 最高优先级, 堆栈大小可配置
- 文件系统可以完全被禁用(最小RAM需求减少到2K)
- 新的BSP如lpc 32xx; lpc24xx, lm32, MPC55xx

□ 4.11

- SPARC V9 (sparc64)
- ARMCortex-M
- Earliest Deadline First (EDF) and Constant Bandwidth Server (CBS)
- Lm3s69xx BSP
- QorIQ BSP
- 动态加载(so/dll)
- USB 2.0Host
- GIT 迁移
- Posix time test
- Coverity 扫描
-

当前工作

- ❑ Cache 管理
- ❑ HyperVisor
- ❑ JAVA
- ❑ MMU 支持
- ❑ SMP支持(分布调度，更多处理器)
- ❑ ISO 9660文件系统
- ❑ Gprof
- ❑ Lua语言
- ❑ 动态二进制加载与动态库加载
- ❑ USB 2.0 host
- ❑ IPv6
- ❑

代码获取

□ 源代码：

- <ftp://ftp.rtems.com/pub/rtems>

□ 工具链：

- <http://www.rtems.com/ftp/pub/rtems/linux/4.10/fedora/>

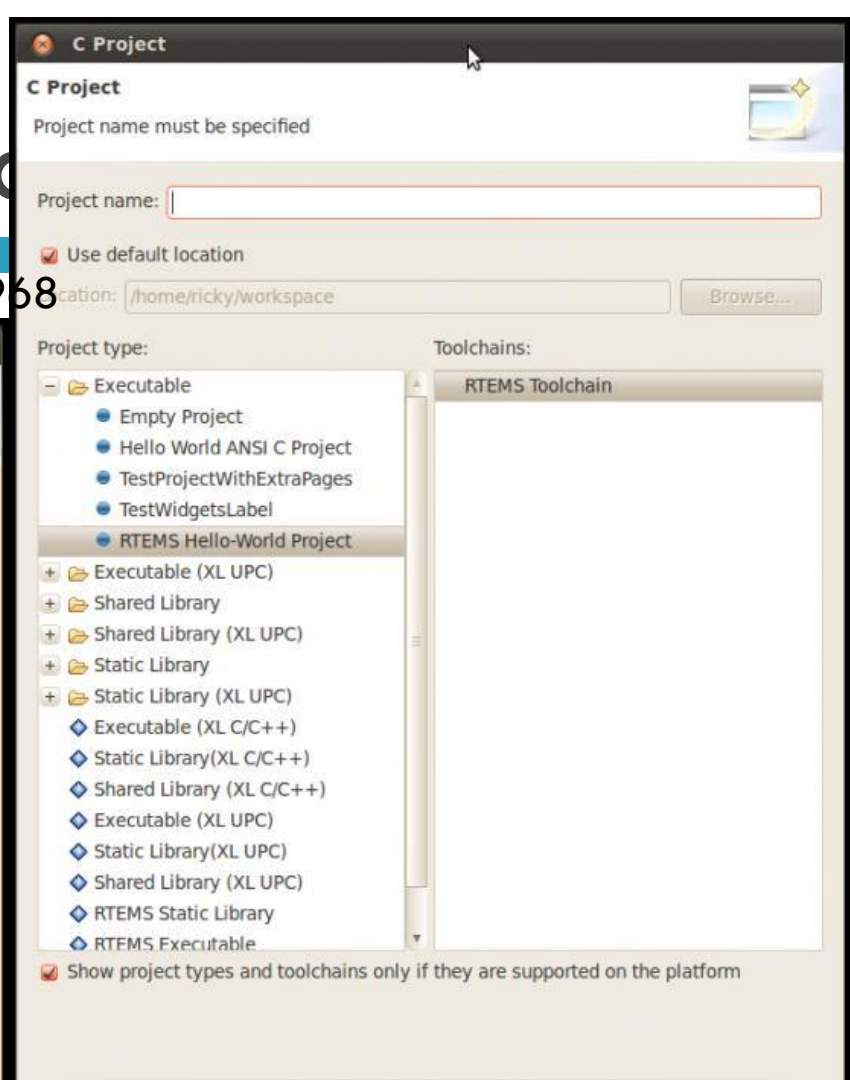
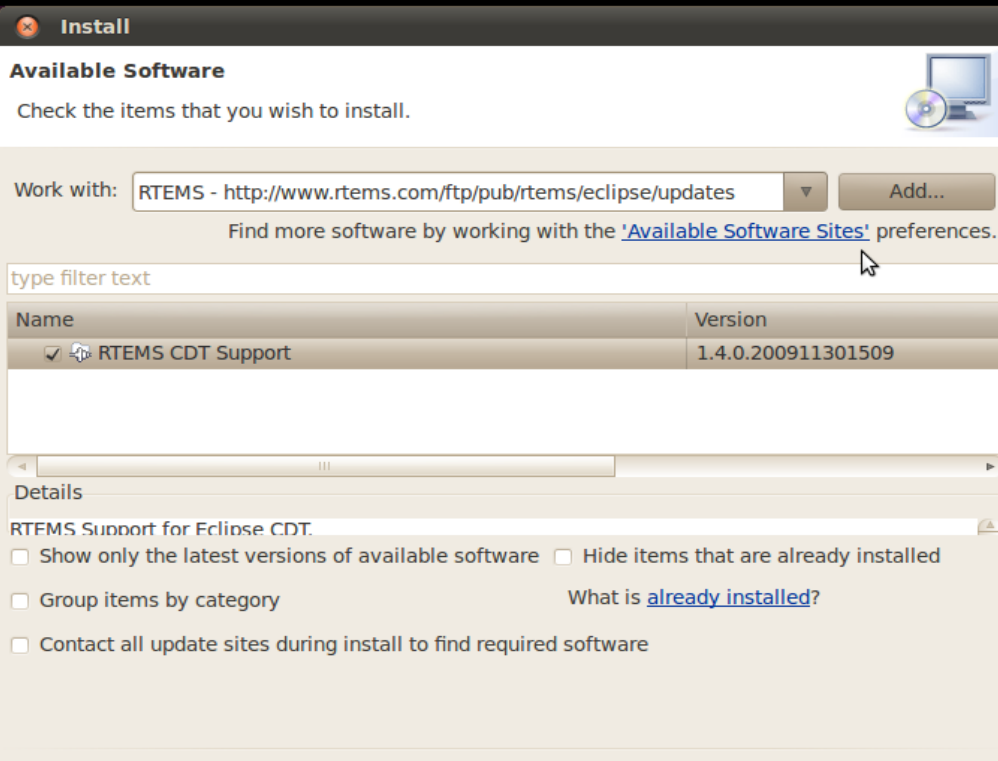
- <ftp://ftp.rtems.com/pub/rtems/SOURCES>

□ MinGW工具在Windows下安装

- 从RTEMS4.8开发版开始，RTEMS也开始支持Window下图形界面的安装包。该安装包需要和MinGW工具配合使用。

Eclipse 安装 (rickle)

<http://blog.csdn.net/rickleaf/article/details/6293968>



The screenshot shows the Eclipse IDE interface. The **Project Explorer** on the left shows a project named "Hello-World" with a source file "Hello-World.c" and a header file "system.h". The **Editor** in the center displays the contents of "Hello-World.c". The code defines a configuration, includes standard headers, and implements an `rtems_task Init` function. A typo in the `printf` statement is highlighted with a red squiggly line: `prlnftf` instead of `printf`. The **Console** at the bottom shows the output of a build attempt, which failed with a linker error due to the undefined reference to `prlnftf`.

```
#define CONFIGURE_INIT
#include "system.h"
#include <stdio.h>
#include <stdlib.h>

rtems_task Init(
    rtems_task_argument ignored
)
{
    printf( "\n\n*** HELLO WORLD TEST ***\n"
    printf( "Hello World\n" );
    prlnftf( "*** END OF HELLO WORLD TEST ***\n" );
    exit( 0 );
}
```

C-Build [Hello-World]
./src/Hello-World.o: In function 'Init':
/home/ricky/workspace/Hello-World/RTEMS Executable Configuration/./src/Hello-World.c:34: undefined reference to `prlnftf'
collect2: ld 返回 1
make: *** [Hello-World.exe] 错误 1

Debug Console (GDB Hardware Debugging)

GDB Hardware Debugger (11/18/08 9:04 AM) (Suspended)

Thread [0] (Suspended)

- 5 testIO() C:\java\SDK_workspace\SerialRemoteDebug\src\SerialRemoteDebug.c:120 0x001001a0
- 4 Init() C:\java\SDK_workspace\SerialRemoteDebug\src\SerialRemoteDebug.c:157 0x00100291
- 3 _Thread_Handler() C:\opt\src\rtms\rtms-4.9.0\cpu\it\score\src\threadhandler.c:144 0x0012c7e3
- 2 <symbol is not available> 0x00000405
- 1 <symbol is not available> 0x00000000

C:\opt\rtms-4.9\bin\386-rtms4.9-gdb.exe (11/18/08 9:04 AM)

C:\java\SDK_workspace\SerialRemoteDebug\Debug\SerialRemoteDebug.exe (11/18/08 9:04 AM)

Variable Window:

Name	Value
devName	0x00167f2f
buffer	0x00167e14
fd	3
numBytes	33

0x167f2f "/dev/console"

SerialRemoteDebug.c

```

}

void testIO(char *devName) {
    char buffer[256];

#ifdef TEST_COM1
    printf("**** Simple COM1 Test (9600 SN1) ***\n");
#else
    printf("**** Simple Remote Debug Test ***\n");
#endif

    int fd = open(devName, O_RDWR | O_NOCTTY | _FNDELAY);

    int numBytes = write(fd, "Hello, I'm waiting for input...\r\n", 33);
    if (numBytes < 0) {
        printf("\nFailed to write to %s!\n", devName);
    }

    numBytes = read(fd, buffer, 255);
    if (numBytes < 0) {
        printf("\nFailed to read from %s!\n", devName);
    }
    else {
        buffer[numBytes] = 0; // terminate
        printf(buffer);
    }
}

```

Library List:

- # CONF
- stdio.l
- stdlib.
- string
- unistd
- fcntl.l
- errno.
- termic
- rtms
- rtms.
- bsp.h
- bsp/u
- bsp/tt
- Init(rt
- # CONF
- libchip
- libchip
- # CONF
- # CONF
- # CONF
- # CONF
- # CONF

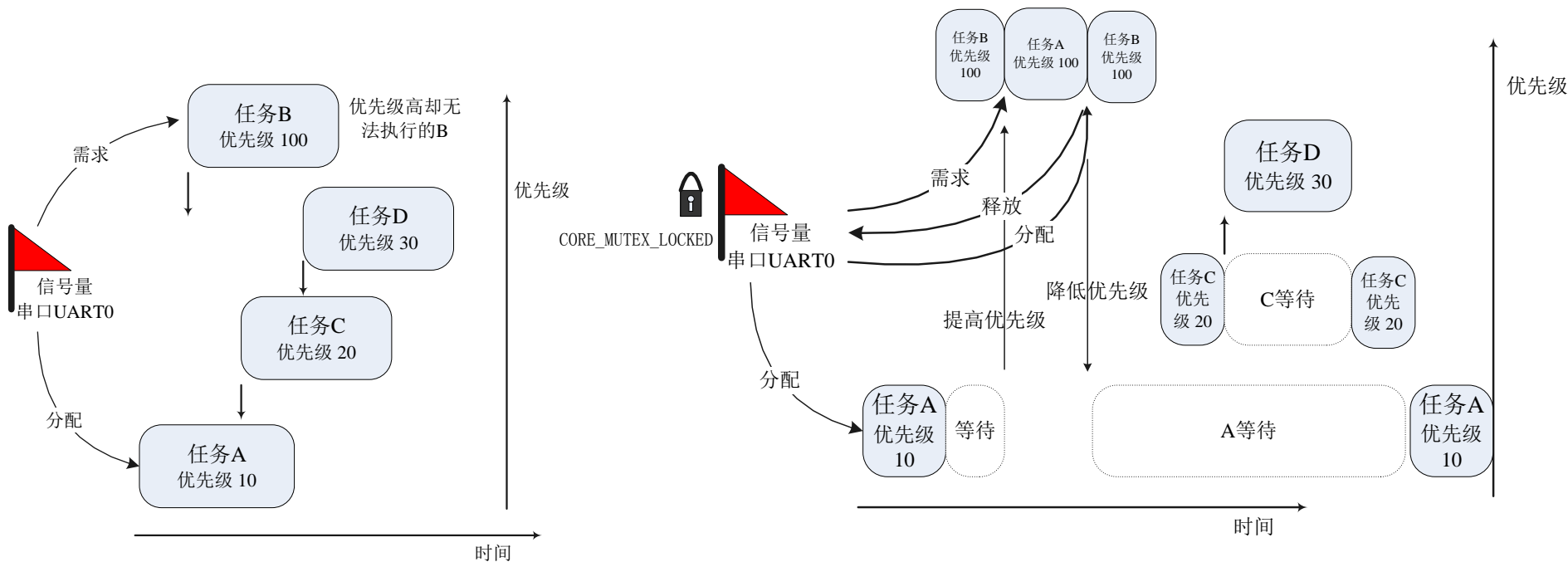
RTEMS系统覆盖测试

- RTEMS提供完善的测试脚本与测试用例
- 使用模拟器追踪指令执行情况并记录
- 对部分平台(ARM, Sparc/Leon, m68K等), 指令覆盖率达到100%

RTEMS Shell 动态执行和文件系统

- Shell支持超过70个指令
- 可以mount多种文件系统(FAT, NFS等)
- 可以获取系统性能数据（类似Top）
- 标准Posix指令，例如cp, rm, mkdir等
- 可使用cexp动态加载应用程序/函数

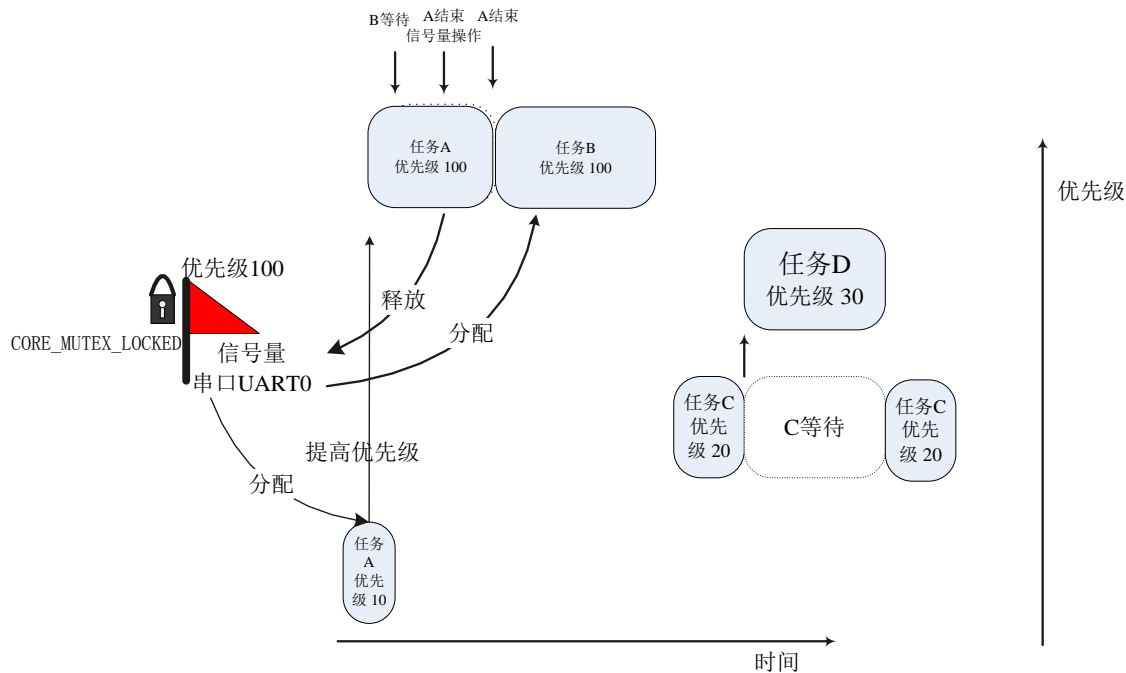
优先级继承



操作系统任务是可抢占的，但是对信号量这样的资源占用却是不可剥夺的，C和D将抢占后，B无法执行

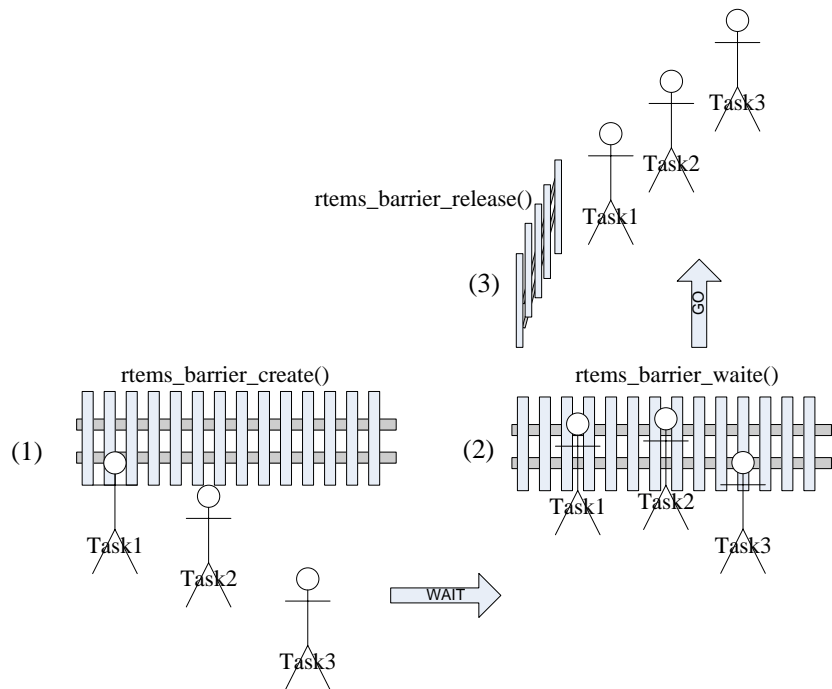
如果获得互斥资源的任务由低优先级变为高优先级，中等优先级的任务就不会抢占该任务

优先级置顶/天花板算法



资源具有优先级，其优先级与预计要求占用此资源的任务中的最高优先级相同。要求占用资源的任务的优先级在得到资源后，会动态提升任务自身的优先级，新的优先级是任务此时所占用的所有资源中最高的资源优先级

任务同步与互斥--Barrier



Page 10 of 10



覆盖测试

sparc/erc32 -O2, POSIX Enabled, Core Only (O2Pd)

Day	Time	Version	Uncovered Ranges	Covered Percentage	Uncovered Bytes	Total Bytes	Results
2011-12-13	19:06	4.10.2	42	98.83	864	73788	untarred tarball

sparc/erc32 -O2, POSIX Enabled, Developmental (O2PD)

Day	Time	Version	Uncovered Ranges	Covered Percentage	Uncovered Bytes	Total Bytes	Results
2011-12-13	19:06	4.10.2	754	53.21	105592	225684	untarred tarball

多处理器支持

□ MPCI

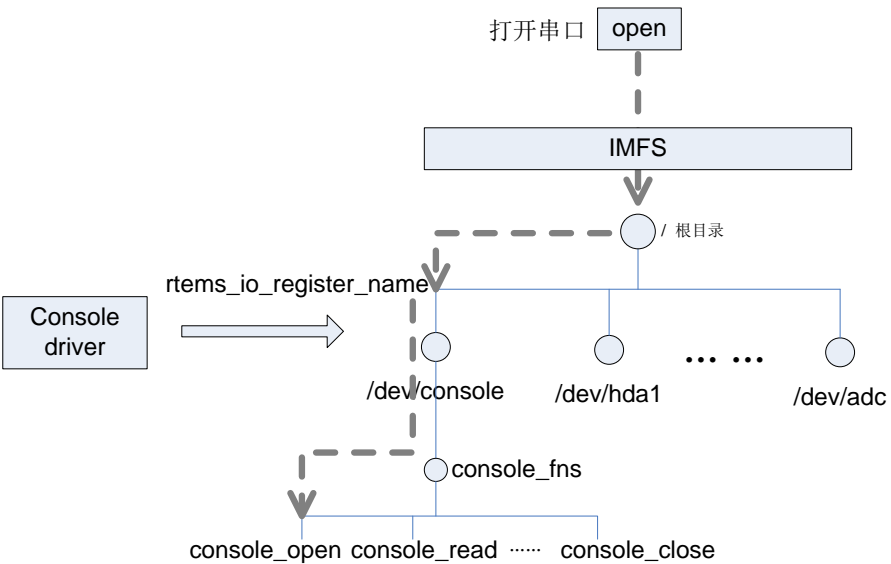
▣ 异构处理器通信

- 大部分RTEMS的API支持本地（缺省）和远程节点，例如
semaphore

□ SMP

- ▣ 目前支持大颗粒锁，支持x86, sparc/leon2/leon3 处理器

驱动程序流程



系统运行实时监控

□ Capture Engine

NAME	RPRI	CPRI	STATE	%CPU	%STK	FLGS	EXEC TIME
04010001	IDLE	255	255	READY	96.012%	0% a-----g	1
08010009	CPIt	1	1	READY	3.815%	15% a-----	0
08010003	ntwk	20	20	Wevnt	0.072%	0% at----g	0
08010004	CSr0	20	20	Wevnt	0.041%	0% at----g	0
08010001	main	250	250	DELAY	0.041%	0% a-----g	0
08010008	test	100	100	Wevnt	0.000%	20% at-T-+g	0
08010007	test	100	100	Wevnt	0.000%	0% at-T-+g	0
08010005	CSr0	20	20	Wevnt	0.000%	0% at----g	0
08010006	RMON	1	1	Wsem	0.000%	0% a-----	0

□ Monintor Engine

□ rtems \$ *task*

ID	NAME	PRIO	STAT	MODES	EVENTS
WAITID	WAITARG	NOTES			

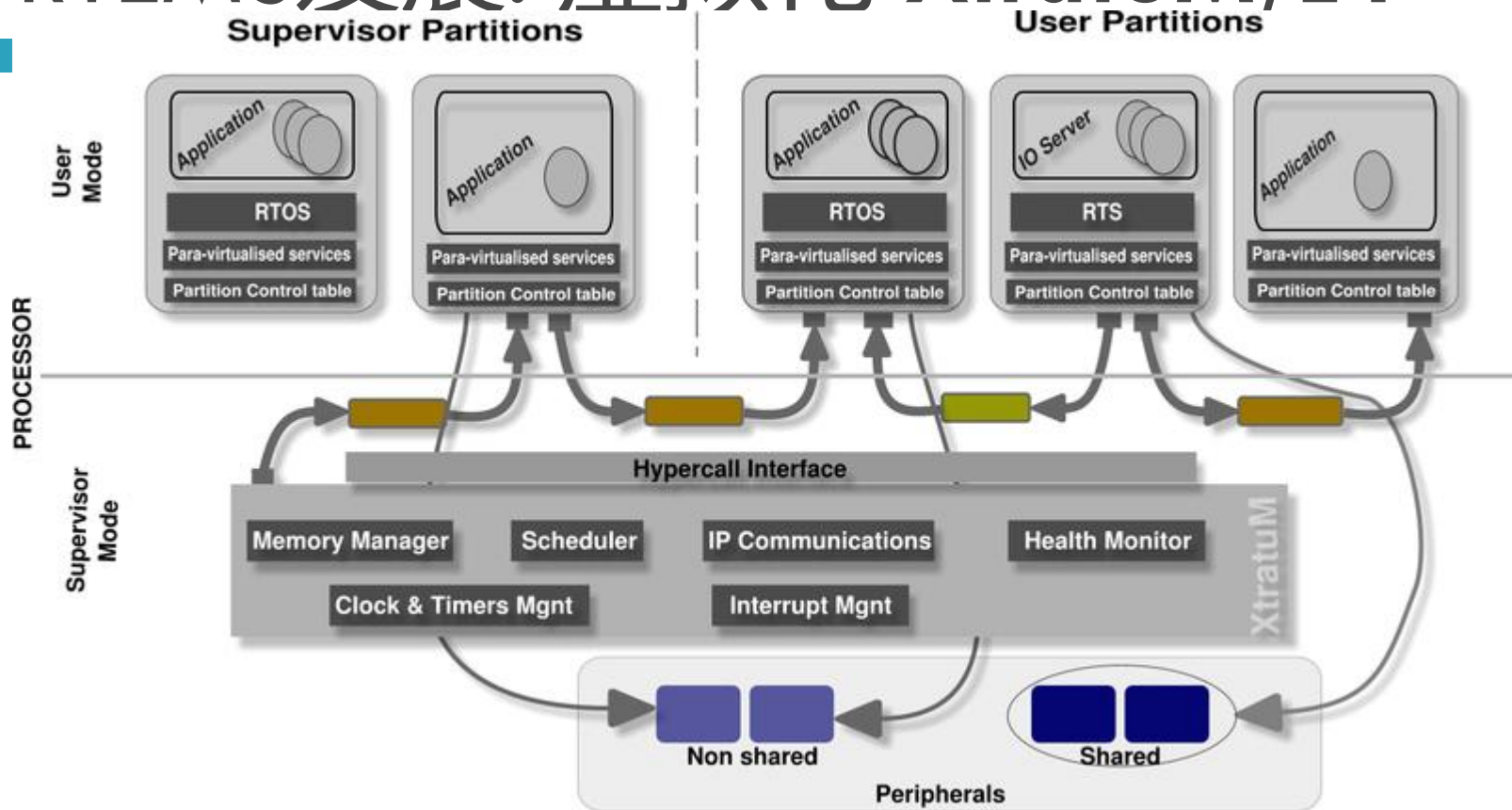
0a010002	TA1	1	DELAY	P:T:nA	NONE
0a010003	TA2	1	DELAY	P:T:nA	NONE
0a010004	TA3	3	DELAY	P:T:nA	NONE
0a010005	TA4	4	DELAY	P:T:nA	NONE
0a010006	TA5	5	DELAY	P:T:nA	NONE
0a010007	RMON	1	READY	P:T:nA	NONE
1a010007	0x1				

Cexp调试

- Cexp> printf("Hello world")
- Cexp> some_variable = 0xdeadbeef
- Cexp> some_double_variable = *(double*)&some_variable
- Cexp> a=printf, a && a("The square root of 2 is %g\n",sqrt(2.0))

- .o 动态库加载
- 类型转换
- 函数调用
- 变量值

RTEMS发展: 虚拟化 XtratuM/L4



RTEMS Hypervisor

