

注意 rtc 的驱动以及默认支持，另外需要有纽扣电池才能实现断电之后实时时钟。

C 测试程序

RTC 测试程序

C 测试程序例程源码压缩包为 “iTOP-4412-miniLinux_c-RTC_v1.0.zip” 。

代码分析

下面主要来分析下 “iTOP-4412-miniLinux_c-LED_v1.0\rtc_test.c” 文件部分代码如下所示。

```
01 : fd = open("/dev/rtc1", O_RDONLY);
02 : retval = ioctl(fd, RTC_RD_TIME, &rtc_tm);
```

01 : open 函数中的第一个参数，执行打开设备节点 “/dev/rtc1” 的动作，获取句柄 fd，用于 ioctl 操作。

02 : 参数 fd 为 open 打开之后获取的句柄；参数 RTC_RD_TIME 为读取时间；参数 rtc_tm 表示对应时间结构体。存放时间变量。

例程测试

```
[root@iTOP-4412]#
[root@iTOP-4412]# mount /dev/sda1 /mnt/
[root@iTOP-4412]# cd /mnt/
[root@iTOP-4412]# ./r
rfid/      rtctest
[root@iTOP-4412]# ./rtctest
[ 228.109465] s5m_rtc_read_time: 2012/2/9 23:19:48(4)
RTC date/time: 9/2/2012 23:19:48
OS date/time(UTC): 9/2/2012 23:19:47
OS date/time(Local): 9/2/2012 23:19:47
```

如上图所示，./rtctest 读取时间。

1.在虚拟终端中使用 date 命令来查看和设置系统时间 查看系统时钟的操作： # date

设置系统时钟的操作： # date 100110202016.00

通用的设置格式： # date 月日時分年.秒

2.同步系统时钟和硬件时钟，使用 hwclock 命令来同步系统时钟和硬件时钟。

系统时钟和硬件时钟同步： # hwclock --systohc

表示 SYStem clock to Hardware Clock

```
[root@iTOP-4412]# date
Sat Oct  1 10:27:10 UTC 2016
[root@iTOP-4412]# date 100110202016.00
Sat Oct  1 10:20:00 UTC 2016
[root@iTOP-4412]# hwclock --systohc
[ 716.322596] s3c_rtc_settime() 2016-10-1 10:20:14
[ 716.325804] s5m_rtc_set_time: 2016/10/1 10:20:14(6)
[root@iTOP-4412]# date
Sat Oct  1 10:20:18 UTC 2016
```