关于树莓派实时时钟遇到的问题

实时时钟设置参考链接

<http://www.raspberrypi-spy.co.uk/2015/05/adding-a-ds3231-real-time-clock-to-the-raspberry-pi/>

<http://blog.csdn.net/huayucong/article/details/50061025>

遇到问题：

开机运行指令

sudo echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-1/new\_device

导致硬件时钟重置成与树莓派系统时钟一样的初始值

树莓派开机启动时间为1979年1月1日

硬件时钟被重置为这个时间并且为不可读变量

解决办法

系统软件有bug，运行

sudo apt-get –y –force --fix-missing upgrade

对系统软件进行重新更新就可解决问题

cd .config/lxsession/LXDE-pi

sudo nano autostart

sudo apt-get remove fake-hwclock  
sudo rm /etc/cron.hourly/fake-hwclock  
sudo update-rc.d -f fake-hwclock remove  
sudo rm /etc/init.d/fake-hwclock

启动

update-rc.d hwclock.sh enable

sudo apt-get install fake-hwclock

再次停止

sudo apt-get --purge remove fake-hwclock

sudo systemctl disable hwclock-stop

参考链接

<http://henson.github.io/post/raspiclock/>

<http://www.elevendroids.com/2012/12/setting-up-hardware-rtc-in-raspbian/>

<https://www.raspberrypi.org/forums/viewtopic.php?f=91&t=67364>

# RealTimeOnRaspberryPi

关于树莓派实时时钟遇到的问题

实时时钟设置参考链接

http://www.raspberrypi-spy.co.uk/2015/05/adding-a-ds3231-real-time-clock-to-the-raspberry-pi/

http://blog.csdn.net/huayucong/article/details/50061025

http://henson.github.io/post/raspiclock/

http://www.elevendroids.com/2012/12/setting-up-hardware-rtc-in-raspbian/

https://www.raspberrypi.org/forums/viewtopic.php?f=91&t=67364

https://www.raspberrypi.org/forums/viewtopic.php?t=85683

遇到问题：

开机运行指令

sudo echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-1/new\_device

导致硬件时钟重置成与树莓派系统时钟一样的初始值

树莓派开机启动时间为1970年1月1日

硬件时钟被重置为这个时间并且为不可读变量

解决办法

系统软件有bug，运行

sudo apt-get –y –-force-yes --fix-missing upgrade

对系统软件进行重新更新就可解决问题，但后来在另一个树莓派上运行该指令以后结果发现又遇到另外一个问题

那就是rpi启动后的时间和最后关机时间相同，并且系统时间还会对RTC进行重置，使得RTC不能正常工作。

（Except that the rpi still powers up with the same time I turned it off, and the RTC does not appear to have advanced while the power was off. initially I suspected that the DS1307 oscillator was not running on battery (though a scope shows a 32khz wave), but now it appears that 'fake-hwclock' is superseding what's in the hwclock. I.e, during startup, fake-hwclock loads the time from /etc/fake-hwclock.data, and then sometime after that, hwclock -w is done (since the hwclock appears to be matching this wrong time by the time I log in). I've done an experiment where I manually put the /etc/fake-hwclock.data a half hour into the future - using a linux laptop while the rpi was off - and confirmed this.

In other words, I can power it off - change the /etc/fake-hwclock.data to some time in the future, then power it on before that time happens, and that time I selected will be the 'system time', and will also have been set into the RTC, as reported by hwclock -r. This will also occur if the recorded time is in the past (relative to when I do the power up) -- as is the case if I don't tinker with /etc/fake-hwclock.data at all.

I tried putting 'hwclock -s' in /etc/rc.local, but that runs way too late.）

由于树莓派内部并没有实时时钟，因此树莓派有一个fake-hwclock，它会保存系统最后一次关机的时间放在

/etc/fake-hwclock.data文件中，并且默认为格林威治时间

要想显示为本地时间，可以把 /sbin/fake-hwclock文件中的这一行

"date -u '+%Y-%m-%d %H:%M:%S' > $FILE"

改为

"date '+%Y-%m-%d %H:%M:%S' > $FILE"

即可.原因是"-u"会将时间转换为UTC时间

实时时钟的完整设置过程

1 使能I2C

sudo raspi-config -> Select “Advanced Options” -> Select “I2C” -> Select “Yes” -> Select “Ok” ->Select “Finish”

2 Install Utilities

sudo apt-get update

sudo apt-get install -y python-smbus i2c-tools

3 shutdown

sudo halt

4 check i2c enbale

lsmod | grep i2c\_

5 test hardware

sudo i2cdetect -y 1

6 DS3231 Module Setup

sudo nano /etc/modules

add contents as follow:

snd-bcm2835

i2c-bcm2835

i2c-dev

rtc-ds1307

then save.CTRL-X,Y and ENTER

7 I2C Device Setup

sudo nano /etc/rc.local

Add the following two lines before the exit 0 line :

echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-1/new\_device

hwclock -s

8 reboot

9 set Time

sudo hwclock --set --date="11/23/14 14:09:00"

10 remove fake-hwclock

sudo apt-get remove fake-hwclock

sudo rm /etc/cron.hourly/fake-hwclock

sudo update-rc.d -f fake-hwclock remove

sudo rm /etc/init.d/fake-hwclock

update-rc.d hwclock.sh enable

or

sudo apt-get --purge remove fake-hwclock

sudo systemctl disable hwclock-stop

11 test if succeed or not

如果上述方案不行则可以尝试下面一种方案

12 如果不行，也可以尝试修改 hwclock.sh

sudo nano /etc/init.d/hwclock.sh

找到 "case "$1" in"并修改

init\_rtc\_device()

{

[ -e /dev/rtc0 ] && return 0;

# load i2c and RTC kernel modules

modprobe i2c-dev

modprobe rtc-ds1307

# iterate over every i2c bus as we're supporting Raspberry Pi rev. 1 and 2

# (different I2C busses on GPIO header!)

for bus in $(ls -d /sys/class/i2c-adapter/i2c-\*);

do

echo ds1307 0x68 >> $bus/new\_device;

if [ -e /dev/rtc0 ];

then

log\_action\_msg "RTC found on bus `cat $bus/name`";

break; # RTC found, bail out of the loop

else

echo 0x68 >> $bus/delete\_device

fi

done

}

case "$1" in

start)

# If the admin deleted the hwclock config, create a blank

# template with the defaults.

if [ -w /etc ] && [ ! -f /etc/adjtime ] && [ ! -e /etc/adjtime ]; then

printf "0.0 0 0.0\n0\nUTC" > /etc/adjtime

fi

init\_rtc\_device

# Raspberry Pi doesn't have udev detectable RTC

#if [ -d /run/udev ] || [ -d /dev/.udev ]; then

#return 0

#fi

13 save

14 Update the real HW Clock and remove the fake

sudo update-rc.d hwclock.sh enable

sudo update-rc.d fake-hwclock remove

15 Now that real hardware clock is installed, remove the fake package and it’s crons:-

sudo apt-get remove fake-hwclock

sudo rm /etc/cron.hourly/fake-hwclock

sudo rm /etc/init.d/fake-hwclock

16 reboot and test

方案三

1 编辑/boot/config.txt文件

**sudo nano /boot/config.txt**

to edit the pi configuration and add whichever matches your RTC chip:

dtoverlay=i2c-rtc,ds1307

or

dtoverlay=i2c-rtc,ds3231

2 卸载fake hwclock

Disable the "fake hwclock" which interferes with the 'real' hwclock

sudo apt-get -y remove fake-hwclock  
sudo update-rc.d -f fake-hwclock remove

3 Run **sudo nano /lib/udev/hwclock-set** and comment out these three lines**:**

**#if [ -e /run/systemd/system ] ; then  
# exit 0  
#fi**

参考文献

https://learn.adafruit.com/adding-a-real-time-clock-to-raspberry-pi?view=all

硬件时钟修改指令：

sudo hwclock --set --date =”11/23/16 18:38:00”

硬件时钟读取和系统时钟对比指令：

sudo hwclock –r;date